

研究、探讨、传播

www.WebSphereChina.net 总第2期

WebSphere 中国

IBM WebSphere行业解决方案的专业纸质读物

封面报道

IBM WebSphere 产品新特性



免费读物 审核赠阅



服务为本 / 资源为根
效率为主 / 客户为尊

<<<详情点击: www.digitalchina.com



中国最大的整合IT服务提供商神州数码公司为您量身定制解决方案。

作为IBM公司最重要的合作伙伴，神州数码系统科技公司IBM事业部整合集团各项资源，本着以客户为中心，以服务为导向的宗旨，为合作伙伴提供基于IBM p系列产品、IBM存储产品、IBM软件全线产品的最便捷的解决方案，最全面的项目方案咨询，最专业的产品技术培训。

了解更多IBM产品信息及解决方案，请致电：

北京：010-62694234 / 广州：020-87582246-8372 / 成都：028-85459888-1978 / 上海：021-22019408 / 武汉：027-87222817 / 西安：029-88342433
福州：0591-83306303 / 济南：0531-82382114 / 沈阳：024-23582662 / 郑州：0371-65825391 / 合肥：0551-2834200 / 长沙：0731-4161486-1361
杭州：0571-87362179 / 长沙：0731-4161486-1361 / 哈尔滨：0451-86299696-3066 / 南京：025-83351122-1219 / 深圳：0755-82966699-8501

一个企业和一座城市的智慧未来

2029年7月1日，星期一。

我依然在7点钟起床，开始新的一天的工作。石嘴山，这座崛起于山水之间的新兴工业城市，大都市的繁荣大幕早已开启。

打开电视，美丽的“卡通小秘书”端庄走来，问候之后，告诉我的是昨天设定的日程安排。紧接着，我所要去往的地方，预计到达时候的天气情况，一目了然。喝下一杯牛奶的工夫，我所要经过的所有路线，适时路况信息图包括对路况发展的预测，已经传输到我的手机。当然，这杯牛奶，我的手机也早已经完成对它的安全和营养检测。

也在这个时间内，“我的医生”已经向最近一家医院的计算机中心发出了信号。设置在服装中的感应器把监测到的体温、汗腺分泌、脉搏和心跳频率等数据自动与医院数据库中所储存的我的“生理正常值”相比较，“我的医生”清楚地告诉我，今天健康状况良好，还没有忘记祝福我：一天好心情！

这，也许只是一个普通市民，在若干年后一个平常的星期一。然而在今天，却是我们的一个梦想，一个追求。对于IBM——作为一家世界领先的提供信息科技和服务的公司，却是一个值得挑战的新课题——建设智慧城市。

我们欣喜地看到，IBM正在接受挑战。那么，智慧城市对市民而言到底以为着什么？它能给我们带来什么样的智慧体验？

我理解，首先，应该是对人本身的服务，包括对生命、财产、身体的安全和保障服务。“智慧的医疗”，正是向IBM提出的第一个挑战。未来，能否通过医疗服务的电脑化和系统化全方位最大化信息的收集和储存，通过互联互通的信息系统使各医疗单位有效地实现无缝信息共享，而借助智能的医疗系统还能为病患者全面提升服务的质量和速度，进而实现

以人为本的医疗服务体系，都是我们今天面对拥挤的医院、不平衡的医疗条件所想到的。

其次，就是对跟人有关内容的服务，最典型的的就是交通服务的改善。今天，我们每一个人可能都曾经或者正在承受堵车的困扰。很多人都在设计我们未来所愿意接受的“畅行无阻”。IBM“智慧的交通”提出要实时进行跨网络交通数据分析和预测，避免不必要的浪费，最大化交通流量、持续进行数据分析和建模，改善交通流量和



基础设施规划、大幅降低碳排放量、能源消耗和各种污染物排放。可我必须要指出的是，面对中国经济的发展大势，汽车保有量的大幅增长却是我们必须面对的物理基础。

再者，就是要给市民一个可持续发展的城市。增长 GDP 的同时不要牺牲城市对居民提供的服务的质量，而且这个增长必须是高效率的，这自然要求我们要着力进行城市产业的改造以至再造，把以传统的制造业为主的 GDP 增长转到服务导向的经济发展。还回到交通，它不是简单的不要堵车，还要考虑诸如企业要运输原料、物资的时候在路上耽误的时间越长效率越低，同时又制造很多的污染，消耗很多能源，在城市里面产生了很多热。我很欣赏 IBM 对“智慧的供应链”的理解和发展，通过绿色供应链解决方案、供应链网络优化平台和碳足迹追踪系统，分析供应链体系结构跟物流体系结构，通过数据分析优化找出更好的供应链和物流设计方式。从长远看，还可以通过全面掌握不同行业的碳足迹，可以更好地计划和管理企业、城市的碳排放量，通过碳排放激励措施机制，促使城市重工业、高能耗产业向清洁型企业积极发展，或者在规划新城区建设的时候，就能有意识地设计产业布局，构筑城市可持续核心竞争力。

毋庸置疑，智慧的城市应当有一个更好的服务公民和服务企业的“智慧政府”。传统政府是按业务、管理职责分别设定，各司其职。新一代政府转型中，目标用一个字讲叫做“e-government”，不是电子化而已，是通过“e”的转型，让一个地方政府或者市政府，以公民为中心调整政府的机构，理顺政府的业务，在部门之间实现共享协同作业，对服务对象实现一站式或者一条龙的服务。而“智慧的政府”意味着变等待服务请求为主动的连续的服务，最大化纳税人的利益。智慧的政府要精简业务流程和降低服务成本，变成一个一站式的面向市民、服务公民的体制，政府按照自己的业务板块化作业。经济发展和稳定是一大块，公共安全是一块，社会服务是一块，教育、社保和市政建设是几大板块，不同城市有不同的做法。支撑这个板块的支撑层有法律框架、市政建设，再到底层其中一大块就是信息的基础架构。

当然，要实现这个美好的“梦”，信息化人才储备，包括最大限度调动人的积极性和主观能动性，是每一个地方政府从这一刻开始就要行动的事情！

让我们，共同努力！

宁夏回族自治区石嘴山市副市长 **张柏森**

www.WebSpherechina.net



IBM MQ V7 及其新特性

WebSphere MQ作为在市场上领导地位的消息整合中间件产品，提供了灵活和可靠的方案来解决系统整合的需求，到目前已经度过了它15岁生日，IBM推出了最新版本MQ V7；MQ为您的SOA提供通用的信息传递主干，使用通用接口将一切连接到一起，以使信息可以自由流通。

WAS 7系统管理的惊鸿一瞥



我从2000年开始使用WAS，第一个项目是农行网上银行，WAS 3.0，那个时候WAS的管理界面是Java客户端。WAS给人的感觉是系统管理方面显得比较笨重，不过在WAS 5开始，这方面的改进非常显著。这不，让人期待的WAS 7发布了。系统管理方面又有不少新特性，让我们看看到底有哪些改进。

专家专栏 P34

卷首语

- 1 一个企业和一座城市的智慧未来

新闻

- 4 IBM计划投资1亿美元研发新手机服务
4 IBM在华成立全球铁路创新中心
5 IBM软件助力打造“智慧的电信”
5 智慧的连接业务与技术 让SOA落地

封面报道

- 7 IBM MQ V7及其新特性
8 IBM WAS V7及其新特性
10 IBM WPS V6.2及其新特性

解决方案

- 13 电信人力资源管理系统优化解决方案
15 金仕达数字化校园解决方案

21 浙江移动网络投诉

处理综合平台二期工程

25 金证基金CRM系统KingCRM

解决方案

28 适用于统一信息平台项目

WebSEAL性能调优解决方案

31 人员整合与协作解决方案

专家专栏

34 WAS 7系统管理的惊鸿一瞥

36 WebSphere MQ For AIX和

HP-UX安装配置说明

40 Ajax：超越桌面的Web用户体验

SOA产品实验室

- 42 手把手进行产品实验室，亲手体验搭建SOA的激动过程

博客

- 44 白话解说Spring容器设计理念
46 自动重启WebSphere Application Server
49 WebSphere Troubleshooting 用工具分析GC Log

有问必答

- 53 应用程序的“类装入和更新检测”

WebSphere 中国
IBM WebSphere行业解决方案的专业纸质读物

电话：010-68867107-806
地址：北京市石景山区石景山路23号中融大厦530室(100043)
邮件：chao_huang@twigroup.com.cn
广告：010-68867107-806 联系人：黄超

主编：黄超
专家顾问：寇卫东 毛新生
甘奎 黄河
张东焕
设计：肖晔(杞人)

《WebSphere 中国》是一本审核赠阅的免费读物，由WebSphere开发与应用网站发起、协作创立并完成的第一本专注于研究、探讨、传播IBM WebSphere行业解决方案的专业纸质读物，服务于广大的中国IBM WebSphere用户。
申请免费阅读，请访问：www.WebSpherechina.net

把最新的菜上来，不差钱！

请您先从这道菜开始吧！



插图 / 纪人

IBM MQ V7 及其新特性

寇卫东、刘洪宇 / 文

WebSphere MQ作为在市场上领导地位的消息整合中间件产品，提供了灵活和可靠的方案来解决系统整合的需求，到目前已经度过了它 15 岁生日，IBM 推出了最新版本 MQ V7；MQ 为您的 SOA 提供通用的信息传递主干，使用通用接口将一切连接到一起，以使信息可以自由流通。允许信息在所有应用与技术服务之间流通，从而提供端到端事务处理能力，并提供更高的可用性、可伸缩性和安全性。

MQ 7.0 新增优势大致包括：对发布 / 预订和 Java 消息服务 (JMS) 的消息传递的增强；发布 / 订阅、JMS、MQ Client 等的增强等。从对业务至关重要的关键企业级的应用到部门级的最小应用，MQ V7 都可以提供当前最高水平的可靠性、安全性、以及可扩展性。MQ V7 为 SOA 应用程序提供了敏捷、可靠的基础，对以百万计的开发者 and 架构师而言，他们可以在 MQ V7 基础上建造基于 SOA 的应用、服务之间消息的传递。

本文主要涉及了 MQ V7 的新特性，并简单分析一下 MQ V7 的新特性带来了哪些实际的好处。

1. 功能的增强

MQ V7 在 MQ V6 的基础上做了较大改动。MQ V7 不仅增强了发布/订阅功能，提供了一种新的发布/订阅引擎并整合在队列管理器中，还提供了对 Web2.0 的整合。

Web 2.0 支持

Web 2.0 试图通过不太复杂的技能及提供更加简便的环境视图。使用 4 个简单的动词通过基于 REST 的 Web 服务提供对资源的访问，仍然需要安全、可靠地访问并交付业务信息。MQ V7 提供了 Web 2.0 整合，将 MQ V7 资源表示为一组 URI，这是在 Web 2.0 中表示服务的通用方式。

与 WSRR 集成

MQ V7 与 IBM WebSphere Service Registry and Repository 完全集成，使 MQ V7 应用更加具有可见性，最大限度实现在整个企业范围内的重用。即使是低级别 MQ V7 构件，例如队列管理器、队列、通信挑战和通用功能，在注册库中也保持可见。

发布 / 订阅增强

在 MQ V6 或之前的版本中对发布/订阅的支持不是产品本身内置的，而是通过 MQ Support Pac 的形式提供的，而 MQ

V7.0 为我们提供了一个新的内置于 MQ 队列管理器之中的内置发布 / 订阅引擎，从而取代了 MQ V6 的 Support Pac，实现了队列管理器自身就可以管理发布 / 订阅工作，而不需要依赖于外部单独的部件；而且，从编程接口而言，发布 / 订阅接口也成为了 MQI 内置的一部分。

2. 性能方面的提升

MQ V7 在内核方面做了大量的优化，尤其是 JMS 和发布 / 订阅。

JMS 性能提升

MQ V7.0 中的一些功能有助于 JMS 应用程序的改善。消息预读功能大量提高非持久性消息的吞吐量，尤其是对于目的队列中有大量的消息需要迅速的处理。在 MQ V7 里，JMS 选择器性能通过服务器端的选择器匹配也得到了改善，消除了因客户端选择器匹配而带来的性能瓶颈。

发布 / 订阅性能提升

MQ V7.0 优化了发布/订阅消息传递的吞吐量。发布 / 订阅功能已经被整合进入队列管理器中，从而简化了应用编程；在 MQ V7 版本以前，发布/订阅不是本地

MQI 接口的一部分，应用需要通过一个队列接口连接到运行在队里管理器之外的代理上来进行通讯；在新版本里，新的 MQI 接口已经支持本地集成的发布 / 订阅功能；因此不需要象以前一样在队列管理器和独立的发布和订阅组件中进行两次队列持久性发布。通过对持久性发布和订阅消息传递的消息日志记录进行优化，吞吐量可以获得大量的提升。

3. 更智能的管理

MQ Explorer 可以支持在 x86 上的 Linux 和 Microsoft Windows 机器上对 MQ 进行远程配置。MQ V7 在 MQ Explorer 图形界面上做了几点改进；首先不管是新手还是初步接触的用户，通过使用欢迎页面都能获得一个很好的入门；其次在 MQ Explorer 管理视图中，您能够监控和管理基础设施的运行状况。例如队列管理器、队列、信道、正在处理的信息以及群集等，都可通过 Explorer 工具查看和管理。此外 MQ7 还新添加了导入和导出设置的功能，包括连接信息，本地的队列管理器信息。■



IBM WAS V7 及其新特性

寇卫东、刘睿 / 文



寇卫东 寇卫东教授，IBM 软件集团两岸三地大中华区总工程师（总架构师），国际知名专家，美国马里兰大学兼职教授，中国通

信学科第一位博士、第一位博士后，有接近30年国内外IT行业经验，曾在北美西门子、美国AT&T、加拿大IBM、瑞典林彻平大学、加拿大滑铁卢大学、香港大学等境外的公司和大学工作17年，设计和领导开发大型软件系统，教授研究生课程，曾任首席工程师、首席研究员、常务副总监等职，曾6次担任国际会议大会主席，出版了7部英文专著、拥有20余项美国、加拿大、中国等国家专利、发表了100余篇论文；回国后，曾任国家重点实验室主任、计算机学院院长，1992年当选为纽约科学院院士；2004年荣获中国政府授予国际专家的最高荣誉，并获温家宝总理等国家领导人接见、颁奖、合影、宴请。

IBM WebSphere Application Server 7.0（在下文中简称WAS V7）构建在早期版本的强大和稳定的核心之上，并反映了IT技术创新的最新发展。通过不断引进和支持最新的技术、标准和编程模型，WAS V7使应用程序与业务完美结合。WAS V7全方位支持 Java EE 5 标准，使用者可以充分利用简化了的 Java EE 5 编程模型，快速开发企业应用，降低开发成本。WAS V7更是在 Java SE 6 运行环境上打造的，它提供了一些新的应用编程接口，以增加开发人员的工作效率。

WAS V7在支撑强健和敏捷业务应用方面独具三大优势：简化开发工作，高性能以及智能的管理。从对业务至关重要的关键企业级的应用到部门级的最小应用，WAS V7都可以提供当前最高水平的可靠性、安全性、以及可扩展性。WAS V7为面向服务的体系结构（Service Oriented Architecture, SOA）应用程序提供了敏捷、可靠的基础。对以百万计的开发者 and 架构师而言，他们可以在 WAS V7 基础上建造、重用、运行、整合、管理 SOA 的应用和服务。

本文的重点不在于讲解 Java EE 5、EJB 3.0、Java SE 6、SOAP 1.2、MTOM、OASIS、Portlet 268 等新引进的标准和协议，而且以一般应用开发、维护以及架构设计人员的视角，来简单分析 WAS V7 的新特性带来了哪些实际的好处和不同。

1. 简化开发模型

WAS V7 增强了对标准化的支持、对新技术的支持，以及对开发框架的选择，这些开发框架简化了编程模型，从而提高了开发人员的效率。WAS V7 提供了对 Java EE 5 规范的完全支持，通过使用注释（annotation），EJB 和 Web 服务繁琐的通用代码被完全的屏蔽。Java Persistence API (JPA) 进一步简化了 EJB 开发，它支持使用带注释的 POJO 来创建实体 Bean。开发人员无须关心细节就能充分享受 EJB 和 Web 服务的强大功能。这种方式符合流行的被称为依赖项注入或反向控制的编程模式。而 AOP 的热门产品 Spring 也得到 WAS V7 的认证。Java EE 5

认证、EJB 3.0 支持、JPA、以及 JDK 6.0 提供了简化了的编程模型。这些编程模型可以用来建造可复用的持久对象，Web 服务支持，包括 JAX-WS, SOAP 1.2, MTOM, XOP, WS-ReliableMessaging, WS-Trust, WS-SecureConversation, WS-Policy, 以及 Kerberos Token Profile, 从而简化了在混合环境下的互操作性。

其它强大的开发特性还包括：

- 支持 Web 服务的各种最新标准。
- 支持跨企业应用程序的 HTTP Session。

2. 更高效的引擎

WAS V7 提供了一个快速、可靠、安



刘洪宇 IBM 中国公司软件部高级工程师。2006 年加入 IBM 公司软件部，主要负责 IBM WebSphere 家族相关产品。刘洪宇为中国移动通信山西分公司、陕西电信、甘肃烟草、贵州公安、山东公安、陕西省高等法院、陕西省政府、新疆塔里木油田等客户的开放系统上提供技术支持，并参与相关 IT 项目建设和软件架构规划。



刘睿 IBM 中国公司软件部高级工程师，联机交易系统专家。刘睿一直为中国人民银行、中国农业银行、中国工商银行、中国银行、中国移动通信广东分公司、华为公司在开放系统的 IBM WebSphere CICS、WebSphere MQ 等系统提供技术顾问支持，还多次举办应用程序开发讲座。刘睿还是开源和免费软件的作者，其有广泛影响力的作品包括 EasyCICS 和 Regular Statement String 等。刘睿的个人网站是 <http://www.Irsolution.com>。

全、开扩展、预备好了的环境来支撑开发和提高应用效率，使得业务可以以较少的资源取得更多的结果。WAS V7 通过部署、Web 服务、以及 EJB2 的增强，以较少的能量消耗处理器完成以前版本所做的同样工作，从而优化了运行性能。这方面的新技术还包括：

- WAS 的 OSGi 技术和智能预置功能可以仅选择加载应用程序需要的运行时功能模块，从而显著减少内存占用空间和应用程序服务器启动时间。

- IBM Java SE 实现引入的共享类缓存又经过改进，大大减少了加载时间。

- 64 位 WAS JVM 中使用压缩引用 (compressed reference) (或指针压缩)，明显减少了 64 位 JVM 的进程内存占用空间。

3. 更智能的管理

智能的管理就是要对可能出现的关系重大的情况有准备、有对策。WAS V7 提供了一个简化的基础设施，灵活和有效的应用控制机制和运行效率，因此企业可以通过基础设施和应用智能化对变化的业务需求及时做出反映。

WAS ND 中的灵活管理特色改善了管理的效率，实现了 WAS Base 和 WAS Express 的远程生产资源管理。新的管理工具使得安全管理更上一层楼。新的 Websphere 业务水平应用 (WBLA) 能扩展到 Java EE 5 之外，大大改善了多组件应用管理。

管理组件与应用程序服务器运行时分离，通过称为管理代理的单独 (基本) WAS 管理进程。系统管理员只需记住一组远程管理端口 (管理代理的管理端口)，而不是每个应用程序服务器的管理端口。

除了以上三大特点之外，WAS V7 在集成能力、支持 SOA 架构、以及安全方面也非常有特色：

● 更强大的集成能力

WAS V7 包括一个 WebSphere MQ JMS JCA 1.5 资源适配器，这意味着 WebSphere MQ 中的消息 MDB 现在可以使用更通用的“激活规范”，而不是老式的侦听器端口。以前版本的 WebSphere，访问 EJB 的远端客户程序必须运行在应用程序客户机容器里，这样造成了一般的系统根本无法访问 WebSphere 的 EJB。现在 WAS V7 提供了 IBM Thin Client for EJB，使这个问题迎刃而解。

● 更好的面向服务架构

服务治理是面向服务架构 (SOA) 的重要方面。WAS V7 提供了丰富的 Web 服务策略集。这些策略包括用于业务相关的断言和 QoS 策略。另外 WebSphere Service Registry and Repository 可以发现 WAS V7 JAX-WS 策略集。在服务的构建方面，WAS V7 通过 Feature Pack for Web 2.0 来提供强大的 Web 2.0 方面的支持能力。

● 更完备的安全体系

单个 WebSphere Application Server 单元中支持创建多个安全域，每个安全域可以具有自己的用户群体。

可以选择生成 WAS 管理操作的安全审核记录。

WAS V7 提供了对 Kerberos 身份验证的支持。

总之，WAS V7 作为 IBM 新一代面向服务的应用服务器的核心基础，无论从功能性、可靠性、还是可操作性上，都不愧为企业级用户的首选产品。■

IBM WPS V6.2 及其新特性

寇卫东、范迪 / 文



范迪 现任IBM公司软件部高级技术顾问。多年来一直从事 WebSphere 中间件及其相关产品的技术支持和软件架构工作。通过

IBM SOA 架构师认证，是国内最早接触和从事 SOA 相关技术支持和推广工作的技术专家之一。在 SOA、BPM、EAI 等领域的设计和开发方面有着丰富的经验，并致力于将其在 IBM 的众多合作伙伴及企业客户中推广普及。曾参与过中国石油、中国网通、中国电信、中国移动、人民银行、交通银行、国家发改委、中国国际航空公司、中国远洋进出口公司、国家税务总局、华为公司等众多大型项目的技术支持和方案设计工作。

企业的业务流程管理 (Business Process Management, BPM) 需要一个功能强大的软件平台来支撑业务流程管理的应用程序。WebSphere Process Server (以下简称 WPS) 就是这样一个软件平台。WPS V6.2 是目前的新版本，它增强了灵活性和对流程实例的控制、简化和改进了应用程序部署过程，以及支持最新技术和标准。本文将介绍 WPS V6.2 的这些新特性。

1. 增强灵活性和对流程实例的控制

如何增强灵活性，使得用户能够定义和测试他们的业务流程，从而缩短需求定义与获得可执行业务流程之间的时间间隔？如何支持新的直接部署场景？如何对于复杂的问题确定场景，可以引入信息技术，同时仍然维持可跟踪性和治理？在早期版本的 WebSphere BPM 堆栈中，流程生命周期一直是从 WebSphere Business Modeler 中的业务建模开始，然后是导出到 WebSphere Integration Developer 进行服务组装，最后是将流程模块部署到 WPS 中。WPS V6.2 对此作了革命性的改变：我们不再需要这么繁琐了，因为 WPS V6.2 支持新的直接部署场景。WebSphere Business Modeler 的用户可以将模块直接部署到 WPS 运行。用户能够定义和测试他们的业务流程，从而缩短需求定义与获得可执行业务流程之间的时间间隔。对于复杂的问题确定场景，可以引入信息技术，同时仍然维持可跟踪性和治理。

WPS V6.2 增强了业务空间功能。WebSphere 提供支持的业务空间是 WebSphere BPM 产品套件中的每个产品附带的一个 Web 2.0 应用程序。用户可以使用称为小部件的应用程序创建自己的空间。WPS V6.2 在工作流图中添加了查看业务流程或任务历史记录的功能，从而

增强了用于工作列表和任务管理的小部件。对于使用特别子任务的动态流程场景，增强了小部件使用户可以查看、创建、修改、取消或验证状态。

WPS V6.2 包括了许多对人工任务的改进。在早期版本中，流程中的步骤可以是内联人工任务，或者是外部参与任务。参与任务具有不能访问流程上下文的缺点，从而限制了可以使用此类任务的场景。WPS V6.2 可以将流程上下文传播到参与任务，从而使业务流程能够管理该任务的生命周期，并使得该任务能够使用替换。任务属性中有一个选项使用户可以将任务的生命周期与调用业务流程绑定在一起。在此场景中，该任务称作是流程的“子任务”。当将人工任务与流程的生命周期绑定时，终止流程也就终止了该人工任务。

WPS V6.2 有一个新功能：支持人工任务的历史记录日志。用户可以在管理控制台中使用针对 BPC 容器和 HTM 配置的面板启用人工任务历史记录。启用以后，用户就可以查看各种任务状态的历史记录，以及执行任务的人员和时间。这样业务人员可以查看任务历史记录。然而，人工任务历史记录不是像公共事件基础结构 (Common Event Infrastructure) 那样的审核机制；而是其生命周期与人工任务绑定在一起。删除任务以后，任务历史记

录也就随之删除了。

以前版本WPS V6.0.2 引入了一个用于页面流的新功能,其中单个用户可以执行流程中的多个连续任务; completeAndClaimSuccessor API 实现了此功能。许多页面流流程也是由将要执行工作的用户启动的。为了支持此场景, WPS V6.2 包括了一个名为 initiateAndClaimFirst 的新 API, 它可以启动流程实例, 然后申领第一个工作项, 全都在一个 API 调用中完成。

在大容量环境中, 用户可能需要在成千上万个任务中挑选出要处理的正确任务。可以使用查询, 并指定该任务的属性, 例如状态、描述和流程数据。尽管该查询可能非常复杂, 但是将用户界面保持快速和响应灵敏是非常重要的。为了支持这样的场景, WPS V6.2 允许用户创建查询表。有一个作为 Support Pac 提供的 Eclipse 插件用于创建 XML 文件, 然后可以使用名为 manageQueryTables.py 的 wsadmin 脚本导入该 XML 文件。此脚本创建优化的 SQL 以用于查询, 而不需要在数据库中创建额外的视图或表。使用此功能, 即使大型查询通常也只需很短的时间即可处理完成。

2. 简化和改进应用程序部署过程

首先通过若干新功能和选项, WPS 增强和改进了安装部署。WPS V6.2 包括完整版本的 WebSphere Application Server Network Deployment 以及 WebSphere Application Server Feature Pack for Web Services。在为 WPS 创建概要时, 有一个用于概要扩充的新选项允许用户选择 Web Services Feature Pack。用户可以创建 Installation Factory 集成安装包以及脚本功能来配置生产环境, 也可以使用一个新的安装验证工具来验证服务器的成功配置。如果安装期间出错, 错误定位机制可以帮助用户找出问题所在。在 IBM z/OS 平台上, WPS 安装的改进是通过减少生成的身份验证别名的数量来完成的。用户现在可以使用 zPMT 配置工具来创建扩充响应文件。最后, WPS 新版本也改进了数据定义语言的生成, 使得其使用现在变得更容易了。

在平台流通性方面, WPS V6.2 包含了若干更新以便与最新的平台保持一致。它目前支持 WAS V6.1, 它也支持将 Windows Vista 用作运行平台, 同时还支持 IBM z/OS 和 z/OS E V1.9 或更高版本。WPS V6.2 还支持 zFS, 用户可以充分利用本机 z/OS 工具, 它还支持 IBM IMS V10。

3. 支持最新技术和标准

WPS V6.2 包含 WebSphere Application Server Feature Pack for Web Services, 而这个 Feature Pack 支持 WAS V6.1 发布以后新出现的 Web 服务标准, 例如, Web Services Reliable Messaging、Web Services Addressing、SOAP Message Transmission Optimization Mechanism 等等。

WPS V6.2 通过添加对 WS-I Reliable Secure Profile、SOAP 1.2 和 WS-Reliable Messaging 的支持, 改进了互操作性。通过对 JAXWS 2.0、JAXB 2.0、SAAJ 1.3 和 StAX 1.0 的支持, 改进了 Web 服务绑定。而服务发现目前支持基于 JAX-WS 和 JAXB2 的 Java 服务, 以及对数组的支持。

当与调用 Web 服务的流程一起工作时, WPS V6.2 支持 Web 服务策略集, 简化了管理。与必须为每个服务调用定义 QoS 限定符不同, 用户可以指定策略集。已有的 Default 策略集包括 WS-Reliable Messaging、WS-Security、WS-Transaction、WS-Addressing、HTTP Transport 和 SSL Transport。

WPS V6.2 符合美国联邦政府的联邦桌面核心配置所定义的安全设置。FIPS 140-2 加密功能则由作为基础的 WebSphere Application Server Network Deploy 环境所管理。

对 WS-BPEL 2.0 支持在 WPS V6.2 中得到了增强。有一个新的构造来支持 RepeatUntil 循环。While 循环在某个条件为 True 时运行, 而 RepeatUntil 循环则至少运行一次, 然后继续循环直至某个条件为 True。

在早期版本的 WPS 中, 所使用的 WS-BPEL 1.1 样式的 XPath 变量引用需要 GetVariableData() 方法。而在 WPS V6.2 中, 新的增强支持通过 \$variable 表示法使用 WS-BPEL 2.0 样式的变量引用。与使用 bpws:getVariableData('customer', '/' / name') 不同, 用户现在只需使用 \$customer/name。

总之, WPS V6.2 这个新版本增强了灵活性和对流程实例的控制、简化和改进了应用程序部署过程, 并且支持最新技术和标准。它也添加了对多个平台的较新版本、以及对 WebSphere Adapters V6.2 的支持, 更新了集成后的 WebSphere ESB 以支持策略驱动的多个中介, 也包括支持与 WebSphere Service Registry and Repository 的集成。■





电信人力资源管理系统优化解决方案

戴科勉 / 文

电信人力资源系统(KPI)是电信一站式服务平台系统 (EIAC) 上的一个功能块。目前该系统已投入生产使用, 满足电信 4 万多名正式员工的月度考核, 绩效沟通等使用要求。

在硬件方面分别是两台 IBM p550 (was) 和两台 IBM p560(oracle)服务器, 操作系统皆是 Aix5.3。

软件架构: IHS+WebSphere集群, 2个节点, 6个 server, 数据库使用 Oracle RAC 集群

系统所遇到的性能问题, 主要表现为以下几点:

- WAS 服务器 Cpu100%
- 内存溢出
- 系统响应时间长
- 数据库服务器 cpu100%
- 系统出现宕机或者服务异常

性能问题成因

● 数据库资源管理不当是造成 J2EE 应用运行效率低下的主要原因

● 功能算法不当也是造成 J2EE 应用运行效率低下的主要原因

● 外资源未及时释放是系统宕机的主要原因

● BUG 的影响

● 其它原因, 不合适的业务流程, 网络问题, 数据库或应用服务器未调优, 意想不到的原因。

系统优化是一个过程, 从应用服务器主机到数据库服务器主机, 从应用服务器 (Websphere) 配置到数据库 (Oracle) 参数调优, 到最后的应用程序优化都是一个过

程, 随着系统功能的增加, 数据量的增加, 使得系统的性能逐步下降, 甚至开始调整的一些参数都需要根据系统使用情况, 重新设置。以下是电信人力资源系统的一个调优的简单介绍:

整个调优主要分为三个阶段, 如下:

1. 环境配置调优

- 1). 系统软件架构与漏斗模型
- 2). IHS+Was 集群 +2 节点 +6Server
- 3). ORACLE RAC 集群
- 4). 操作系统 (AIX) 参数优化
- 5). IHS 参数优化
- 6). WAS 参数优化
- 7). ORACLE 参数优化

2. 通过压力测试与监控调优

- 1). 监控部署
- 2). 压力测试与分析
- 3). Was 参数调整 (线程池调整)
- 4). 数据源变更 (C3p0 变更为 was 数据源)

5). 缓存组件变更 (Ehcache 与 Memcache)

6). 将 AIO 方式修改为 NIO 方式

7). 去除节点自动同步

8). 增加 jvm 启动配置

3. 应用程序调优

- 1). SQL 优化
- 2). 程序逻辑优化 (设计优化)
- 3). 缓存使用优化
- 4). Open session in view filter
- 5). 批量提交
- 6). 数据库索引优化

7. 存储过程优化

8. OSCACHE 组件引入

调优经验分享:

寻找平衡点:



为了让系统能在一个时间段内处理更多的请求, 我们需要分配各个管道的资源请求, 换句话说也就是寻找系统的平衡点, 合理的设置各个管道的参数, 让每个管道的请求都能快速被处理, 各个资源都能充分利用起来。

我们需要根据系统实际应用情况寻找系统的平衡点呢? 分别设置 IHS, 线程池, 数据库连接池等连接参数, 充分利用各个池的资源, 使得系统可以快速响应更多的并发请求, 且充分利用每台服务器的资源。

并发数通常用来衡量系统的性能, 但并发数的概念如按词义去解释, 我们很容易理解为在同一瞬间, 系统同时处理多个请求。但服务器主机cpu资源有限, 所以同时处理请求不可能太多, 但客户需求上又明确指出系统需要达到并发几千的请求, 这样是否目标太高我们无法实现呢?

并发用户数或简称并发数

用户需求规格说明书中描述的并发数指标通常被理解为, 支持若干人同时的访问或同时在线。

但实际情况下, 同时通过页面请求同时到达服务器的请求往往有限, 同一时间

点，绝大多数用户从发送请求到请求返回，多数处于等待状态。

因此并发用户数的定义是为了衡量并发用户对系统的负载。

我们调优过程中，并发用户数，是指某一时间内，对应用系统发起请求并处于等待处理的用户数。

确认了并发数的概念，也确认了我们的目标，剩下就是如何去寻找我们系统所能承受的并发数问题了。

系统业务不同，实际应用给系统的所带来的压力也不同。以写操作为主的请求，压力通常都体现在数据库端；而以读为主的操作请求，压力则体现在应用服务器端，甚至对网络带宽都有一定的要求。但一个业务系统是不是仅仅只是读操作或者写操作为主？

实际应用上，常常出现多种的情况。其中读和写的操作可以是并行，也可以是读为主，或者写为主，在这里我们可以单方面的通过种种的途径去针对性能调优读



或写这个操作过程，但如何让系统更好的运行，满足生产使用的要求，这是我们不能忽略的问题。寻找出各个管道参数配置的平衡点很关键，所以我们在系统调优前要了解系统的实际应用情况，定制出不同的测试场景，通过压力测试，寻找出最合适系统参数配置，合理分配每个池的资源，寻找出一个合适的平衡点。

压力测试经验分享：

压力测试的数据采集是系统优化过程中必不可少的一个环节，我们可以根据调优前系统的性能状况与调优后系统的性能状况，来判断与抉择该功能块是否需要调优，之前调优所制定的方案是否成功。

每一次的系统调优，我们都会使用 LoadRunner 工具进行压力测试，并且采集以下数据：

1、cpu，在压力测试情况下，每台服务器的cpu使用情况，包括应用服务器与数据库服务器

2、内存，在压力测试情况下，每台服务器的内存使用情况，包括应用服务器与数据库服务器

- 3、吞吐率，每秒完成事务数
- 4、完成事务数
- 5、事务响应时间
- 6、事务出错率

常见压力测试方法：

- 功能压力测试
- 极限测试（最大并发数）
- 疲劳测试（系统稳定性测试）

定制压力测试计划，可以让这个压力测试过程事半功倍，其中计划主要包含以下几步：

- 1、压力测试用例定制
- 2、压力测试数据采集的方法

3、压力测试数据的分析方法

系统异常处理经验分享

信息收集

● 收集各个服务器信息(cpu,内存,io,网络io等)；

1、通过 nmon 工具收集 cpu，内存，磁盘 io，网络 io 等使用情况；

2、使用 vmstat 命令检查 cpu 的详细信息使用情况；

● 收集日志信息；

● 收集 /WebSphere/App*/profiles/AppSrv*/logs/ 的日志信息；

● logs/server 目录下的日志信息；

● SystemOut.log、SystemErr.log、native_stderr.log、native_stdout.log；

● ffdc 目录下的日志信息；

● 内存溢出时在默认节点目录下 javacore、heapdump 日志信息；

● 如无内存溢出，可以通过 kill -3 pid 强制生存 javacore 文件；

● 另外可以通过系统参数配置，或者用 wsadmin 命令生产 heapdump 文件；

● 检查系统各个功能点，确认是局部问题还是整个系统的问题；

文档形式收集系统的情况，出错页面，各个常用功能快的访问情况；

● 收集数据库信息；

可以通过 Statspak 与 AWR 工具进行数据库的数据收集，根据报告对了解整个 oracle 的使用情况与性能情况。

以上是电信人力资源管理系统监控优化的部分经验分享，希望能为大家带来帮助。

调优是一个过程，要让系统能更好的运行，不单单是配置参数修改，而是要多方面的配合，因此我认为调优是一直贯穿于整个系统的生命周期。■



摄影 / 杞人

金仕达数字化校园解决方案

复旦金仕达 林强 / 文

校园信息化面临的挑战



以整合为核心，建立一体化的数字化校园

- 数据整合:** 建立数据交换平台和公共数据平台，实现学校信息资源共享和利用
- 应用整合:** 通过公用的技术平台和工具，实现应用系统之间的业务协同和统一服务；
- 流程整合:** 通过全局服务模型开发和管理工具，实现应用流程优化和整合；

集中与分散相适应的建设模式

- 集中建设模式:** 基础平台和全局性应用系统采用集中建设模式，把数字化校园的核心内容掌握在自己手中；
- 分散建设模式:** 部门级应用系统采用分散建设模式，选择最满足学校需求的产品和能提供最佳服务的厂商，并通过系统整合技术融入到数字化校园整体框架；

金仕达数字化校园建设理念

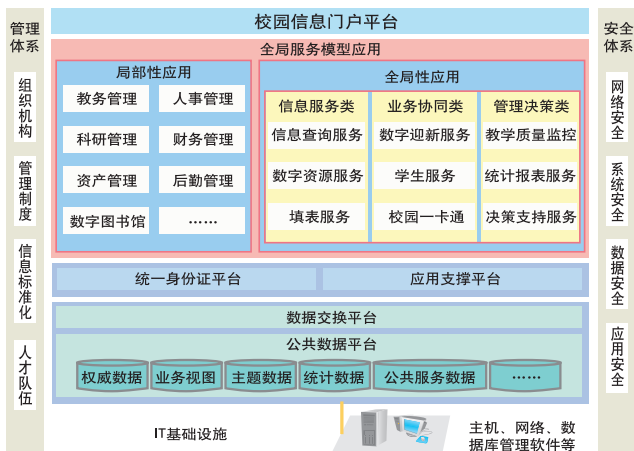
SUNGARD 金仕达针对我国前期数字化校园建设存在的问题，结合公司在高校信息化建设领域的实践成果，以及在金融、企业和政府领域的信息化建设的经验，提出了“数字化校园”总体解决方案。

突出业务协同和面向服务的应用体系建设

- 业务协同:** 加强部门间的管理协同，细化管理粒度，提高管理效率；
- 面向服务:** 应用系统不止面向管理人员，同时为师生提供服务的平台。

金仕达数字化校园总体框架

SUNGARD 金仕达基于 SOA 体系架构的全局服务模型来构建数字化校园总体架构,将学校的资源、数据、信息和应用流程,按照基于服务的方式整合起来,使它们之间彼此互相关联、数据共享、融通。总体框架如下图所示:



整个框架分为五个层次:

- 基础设施层:** 由主机、网络、系统软件、数据库软件等组成,是学校信息化的基础。
- 共享数据层:** 对应用服务层的数据进行抽取、综合、归类和抽象,形成全局性信息,支持全校性的共享、分析和互操作。通过数据交换平台,使各应用系统能在数据层面形成一体,实现数据共享,消除信息孤岛。
- 应用支撑层:** 统一和规范基础服务工具,例如表单工具、报表工具、分析工具、工作流服务工具等,统一技术标准,实现流程的整合和优化。提供统一的身份管理、认证服务和权限管理平台,实现用户、认证和权限的统一管理。
- 应用服务层:** 由学校各部处和院系的各种应用系统(例如教务、科研、人事等)和覆盖全校的全局性应用服务(例如填表服务、归档服务、迎新服务)组成,覆盖了学校在教学、科研、管理、服务等各个领域的业务、管理和决策,是学校信息化工作环境。
- 服务展现层:** 实现浏览、查询、搜索、互动等一站式的服务,实现单点登录,体现平台共享、个性化、交互式的特点。

金仕达数字化校园解决方案特点

● 先进的总体架构

金仕达充分参考了国内外高校信息化发展的现状和趋势,结合 SOA 体系架构的理念,提出了全局服务模型架构,既能够满足目前的数字化校园建设需要,同时能适应未来的扩展。

● 金融级别的数据交换平台

金仕达数据交换平台是真正意义上的数据交换产品,支持系统间的异步、同步、定时、实时、增量、批量数据交换;能够方便的实现复杂网络的互联,完成高强度的加密,解决多个校区、不同网络之间的系统通讯问题,保障数据安全可靠的传输;同时,图形化的设计界面和全局监控使维护和管理更为简单。

● 灵活的核心数据模型设计策略,建立学校信息化标准

金仕达参照教育部标准和学校需求,建立学校信息化标准。同时,金仕达采用元数据管理思路,制定灵活的数据模型设计策略,使业务系统变化不会对核心数据模型产生大冲击。

● 金融级高性能和高可靠性,企业级可扩展性的公共数据平台

金仕达公共数据平台采用多种数据校验方法,保证数据的一致性,利用数据库技术、CACHE 技术和并行处理技术保证数据处理系统性能。公共数据平台与金仕达数据交换平台的无缝连接,保证数据采集的可扩充;同时,建立了统一的数据服务机制,保证向应用系统提供数据服务的可扩充。

● 功能完整的统一身份认证平台

金仕达统一身份认证平台实现统一用户管理、统一认证服务和统一权限管理,同时提供全面的操作审计和分级授权。

● 成熟的商用门户平台

金仕达采用国际知名厂商的成熟的商用门户产品,结合金仕达在教育行业的经验和技術积累,为用户提供功能强大、可靠的信息门户。

● 加强横向协同、深化纵向管理、突出服务的应 用产品

金仕达一卡通、金仕达学生综合服务解决方案等应用产品,强调优化业务流程,加强部门间的业务协同,提高部门间的沟通效率;加强纵深的管理,减少管理复杂度,提高部门的管理效率;同时,突出产品的服务意识,提高全体师生的满意度。

金仕达数字化校园解决方案—数据整合服务

公共数据平台

通过公共数据平台构建数字化校园全局共享数据集,形成数

摄影/纪人



数模型、按主题整合的各业务系统的主题数据模型、全校统一的标准代码数据模型。

● 数据处理及服务

对数据交换平台交换来的数据进行加载、整合；对数据进行分析、挖掘，形成中间统计层；对外提供统一的数据服务。**数据处理及服务。**

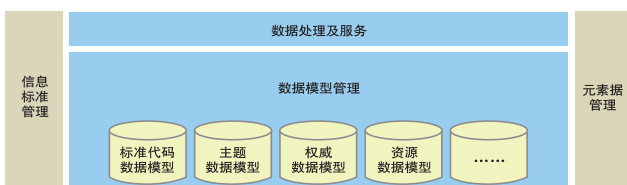
对数据交换平台交换来的数据进行加载、整合；对数据进行分析、挖掘，形成中间统计层；对外提供统一的数据服务。

数据交换平台

通过数据交换平台系统，为不同的应用系统在异构复杂环境

下提供了实时数据通信平台,突破了横亘于异构不兼容应用或数据库之间的障碍,实现了不同应用间的松散耦合联接。

总体框架



功能介绍

● 信息标准管理

对公共数据平台所采用的学校执行标准、所参考的国标、部标以及业务系统数据字典的对应关系进行管理,并对变化情况进行跟踪。

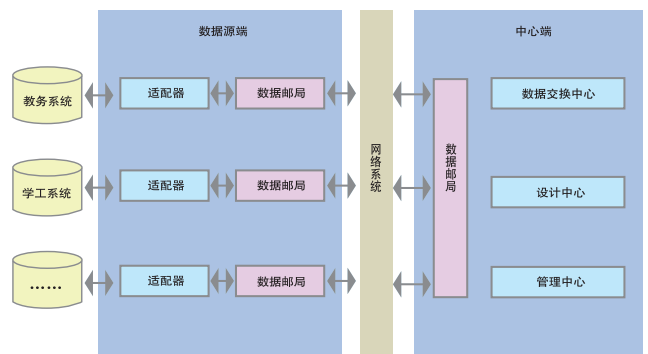
● 元数据管理

对公共数据平台的数据中心中的核心数据模型、应用数据模型、数据模型映射、数据对象进行管理和跟踪。

● 数据模型管理

数据模型是信息共享和公共数据库平台的核心,它对数据源系统中所采集的数据进行重新组织,同时建立如下几种数据模型:面向全局的权威数据模型、以文件等形式存储和管理的资源

总体框架



功能介绍

● 适配器

适配器工作在应用系统端,负责该系统与其他应用系统的数据交换。

● 数据交换中心

数据交换中心在系统中,起到数据统一交换的作用:各应用发送到中心的数据,由交换中心根据预先配置的交换规则交

换到目标系统中。

● **设计中心**

实现图形化的交换设计功能，简捷的鼠标拖拽操作方式。

● **数据邮局**

在与业务无关的通讯中间件基础上，实现数据的可靠传输。

● **管理中心**

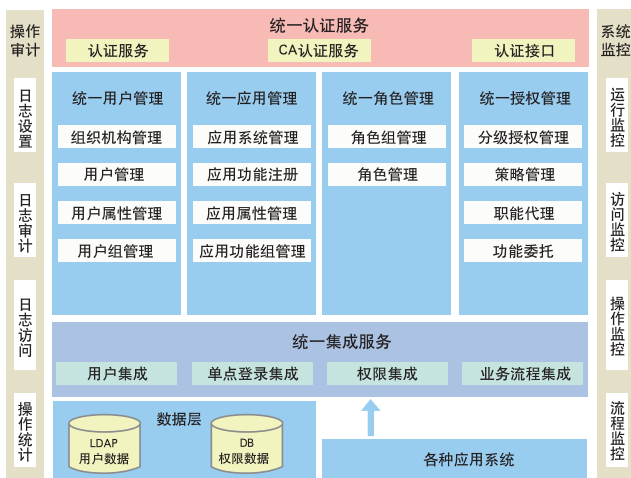
管理中心子系统是“数据交换平台”的主要支持系统，包含系统监控、日志管理、认证管理、数据维护等功能。

金仕达数字化校园解决方案－应用整合服务

统一身份认证平台

统一身份认证平台，作为网络应用的授权和访问控制中心，实施由用户到应用的安全管理。它为数字化校园中的所有应用系统提供统一的用户管理、认证服务和权限管理，实现各应用系统的“集中认证”、“统一授权”。

总体框架



功能介绍

● **统一用户管理**

管理数字化校园中使用统一身份认证系统的所有业务应用系统的全部用户。

● **统一角色管理**

管理统一身份认证系统本身和数字化校园中所有应用系统中需要使用到的角色信息。

● **统一应用管理**

管理各个应用系统的应用功能点和应用权限。

● **统一授权管理**

实现用户与角色、角色与功能的对应管理，实现分级授权、功能代理、权限委托等多种权限分发管理。同时，实现权限策略的定制和调用，便于实现与应用流程的紧密结合。

● **统一认证管理**

为各数字化校园中各应用系统提供统一的认证服务。

● **统一认证服务**

为各数字化校园中各应用系统提供统一的认证服务。

● **统一集成服务**

实现业务系统的用户、认证和权限整合，将相关系统整合到统一身份认证平台中来。

● **操作审计**

记录各种操作行为，通过多种提醒方式，便于管理和审计。

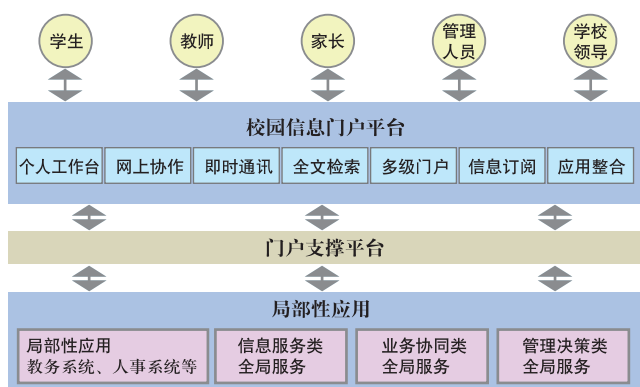
● **系统监控**

对平台运行情况、访问情况、操作情况和业务集成情况进行全面监控。

校园信息门户平台

综合信息服务门户是访问全局服务模型的唯一入口，它是学校信息资源的综合展现。它在表现层整合各个应用系统，为学校用户提供一站式的信息服务。通过门户建设，为全校师生的工作生活带来便利。

总体框架



功能介绍

● **个人工作台**

综合信息服务门户允许用户将个人的日常工作任务和所需信

息统一集成到门户中,使门户成为个人工作的电子桌面,用户可以根据需要自己定义自己的工作台的界面和风格。

● 网上协作

综合信息服务门户具有极强的协作性,通过整合数字化校园中的各种基础服务,提供校园的协作及交互功能,包括文档共享、知识搜索、办公协同、专题论坛等。

● 即时通讯

提供校园内即时通讯的功能,使门户成为互动、沟通的平台。

● 全文检索

建立校园内搜索引擎和索引引擎,实现校园信息资源的有效和快速检索。不仅可以对网页进行全文检索,同时支持对文件、关系型数据库的全文检索。

● 多级门户

允许学校的各级组织和个人建立自己的子门户,用于各种信息发布和交流。

● 信息订阅

在信息门户相关的内容源上添加或更新信息时,信息订阅可以通过 E-Mail、即时消息、短消息等向用户进行通报或者发送提醒。

● 应用整合

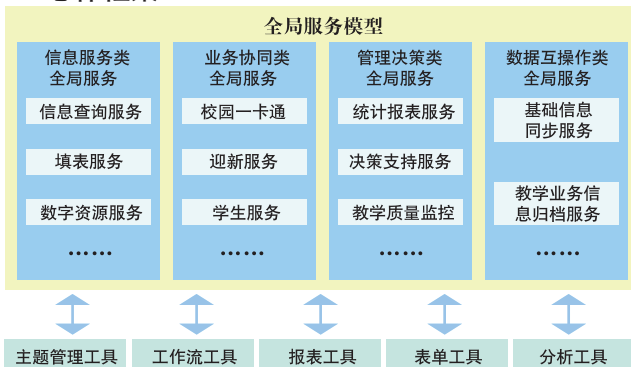
综合信息服务门户实现应用业务系统的整合,将相关系统整合到统一门户中来。

金仕达数字化校园解决方案－流程整合服务

应用支撑平台

为了支持全局服务模型服务和应用的建设,需要一个功能强大、完备的支撑平台来开发和管理这些全局服务模型。

总体框架



功能介绍

● 主题管理工具

实现查询主题的定义,基于公共数据平台,定义查询主题,用于数据查询、发布和服务。

● 表单工具

支持表单的定义、生成、部署和管理。表单工具实现跨部门的简单应用。

● 报表工具

实现各种统计分析报表的定义和生成,能根据全局服务需求,基于公共数据平台,自动生成统计分析图表。

● 分析工具

实现多维分析,为领导决策提供服务。

● 工作流工具

实现业务流程的定义和配置,结合其它工具实现全局性业务流程的整合和优化。

金仕达数字化校园解决方案－应用体系建设 学生综合服务解决方案

学生综合服务解决方案提供学生从入学到毕业离校成为校友,整个过程中涉及到所有与学生相关业务信息化,主要包括七大系统:

- 数字迎新系统
- 学生工作管理系统
- 住宿管理系统
- 教学管理系统
- 就业管理系统
- 离校管理系统
- 校友管理系统

其它管理类应用解决方案

其它管理类应用解决方案涵盖了人力资源管理、科研管理、教学质量管理等部门级业务管理。

- 教学质量管理系统
- 人事管理系统
- 科研管理系统

金仕达数字化校园解决方案－系统运行环境

基础平台运行环境

软件环境

操作系统

- 服务器端: AIX, Solaris, HP-UX, Linux

- 客户端: Windows2000/XP/2003

数据库

- DB2, Oracle

应用中间件

- Websphere, Weblogic, Oracle Application Servers, Tomcat

应用系统运行环境

软件环境

操作系统

- 服务器端: AIX, Solaris, HP-UX, Linux

- 客户端: Windows2000/XP/2003

数据库

- DB2, Oracle

应用中间件

- Websphere, Weblogic, Oracle Application Servers, Tomcat

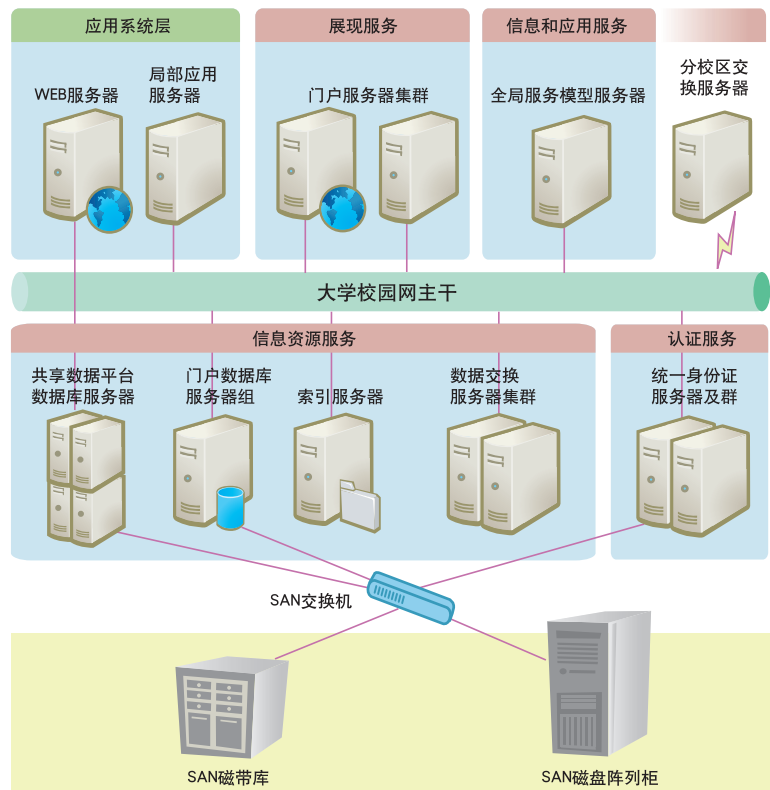
成功案例

- 上海交通大学
- 同济大学
- 上海财经大学
- 上海海事大学
- 上海中医药大学



金仕达数字化校园解决方案－网络结构

数字化校园网络结构





浙江移动网络投诉处理综合平台二期工程

杭州东方通信软件技术有限公司 / 文

一、网络投诉处理综合平台思路和目标

1、投诉处理平台二期建设思路

东方通信在浙江移动网络投诉处理业务需求的指导下,结合自身公司实力和第三方合作伙伴 IBM 公司的共同努力,顺利完成了投诉处理综合平台的研发工作,为浙江移动在网络投诉处理能力提升过程中提供了有力的技术支撑。在投诉处理综合平台的二期建设过程中,东方通信将总结一期项目建设经验的基础上,继续有效贯彻中国移动投诉前移的指导思想,实现网络投诉处理

的两个前移目标: 客服前移和监控前移,为浙江移动网络投诉处理水平更上一个台阶提供有力的基础。因此,浙江移动网络投诉处理综合平台二期建设的总体思路是:

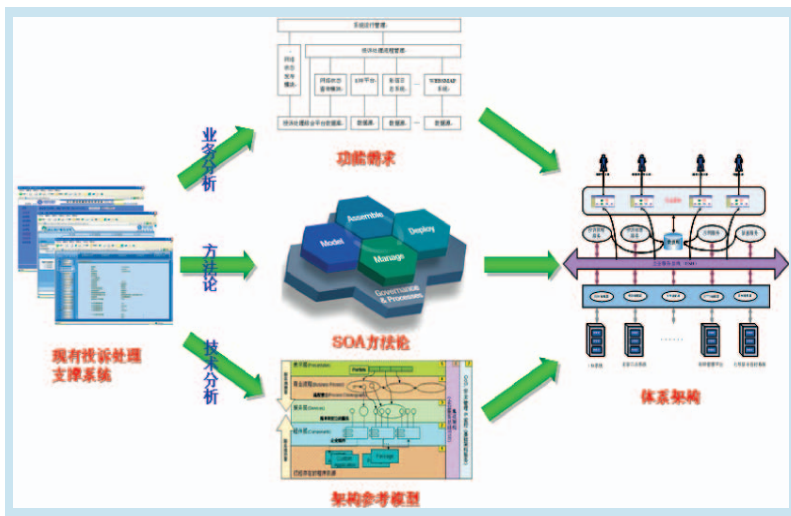
(1) 首先,在浙江移动网络投诉综合平台一期建设稳定运行的基础上,通过梳理和固化常用的网络投诉处理经验,增加集成客服部门、监控部门、网管各专业室、数据业务中心和地市公司等部门和单位最为关注的投诉处理 IT 支撑系统;

(2) 其次,重点解决与 IMEP 投诉工单流程无缝衔接,实现

投诉处理自动化、投诉工单智能预处理和自动化回复,达到投诉问题一键通的解决效果,实现投诉处理综合平台作为“支撑工具”向“关键支撑系统”的定位转变;

(3)再次,二期项目工程建设过程中将优先针对一期建设所存在的不足之处进行提升和解决,增强系统的可靠性和稳定性,提高系统查询性能和速度,为系统向客服一线人员的推广使用提供有力的保障。

东方通信在浙江移动投诉处理综合平台的二期项目工程建设过程中,将协同浙江移动进行网络投诉处理业务需求的整理,梳理最为常用的网络投诉处理向导,调研可集成的网络投诉处理支撑IT系统,同时结合业界先进的SOA方法论和系统架构参考模型,提出贴切实际应用的系统体系架构,保障投诉处理综合平台的可用性和可扩展性,为二期项目工程建设的顺利执行奠定一定的理论技术基础(如下图所示)。



2、投诉处理平台二期建设目标

浙江移动网络投诉处理综合平台二期工程的建设目标即在一期实现的网络投诉处理综合平台上,进行系统优化升级,改进系统整合方式,提高系统性能,增加整合其他投诉相关专业系统,开发面向一线投诉处理人员的使用界面。主要包括:

(1)根据投诉业务需求在一期SOA架构基础上增加集成其他相应投诉处理IT支撑系统,其中包括PMS平台、EMS系统、卓越网络平台、IMEP工程模块、指挥调度系统、手机定位平台、手机证券管理系统、手机报系统等等;

(2)提供与IMEP系统的电子工单交互接口,实现网络投诉

处理工单流转的无缝衔接;

(3)实现投诉处理指导流程基于一定的处理规则进行投诉工单自动预处理,实现投诉处理智能化,达到投诉处理一键通效果;

(4)在一期的基础上增强投诉问题统计分析,通过统计结果分析网络当前存在的问题;

(5)在一期的基础上优化基于SOA的服务整合策略,增加数据处理中心,将常用服务数据保存到网络投诉处理综合平台数据库中,以增加系统响应速度;

(6)优化前期平台实现的安全管理、日志管理、数据备份和恢复、系统基础数据配置等基本功能,提高性能;

(7)二期将在一期稳定运行的基础上推广到客服一线人员和地市维护人员。

通过网络投诉处理综合平台二期的建设,所带来的影响主要包括以下几个方面:

1)将浙江移动网络投诉处理综合平台推广到客服一线人员和地市维护人员,实现真正的投诉处理前移,进而缩短了投诉处理时间,提高了用户满意度,同时减少了流转到网络部门的投诉数量;

2)实现网络投诉处理的自动化和智能化,减轻了投诉人员的工作量,提高了工作效率;

3)集成不同的投诉专业数据,并在这些数据之上进行了加工处理,实现了不同专业投诉数据的共享,增加了投诉现象关联分析能力;

4)实现与IMEP网络投诉处理工单的无缝衔接,通过投诉工单智能预处理和自动化回复,方便了网络投诉处理人员进行一站式管理,增强了投诉

处理人员工单和工作量的管理;

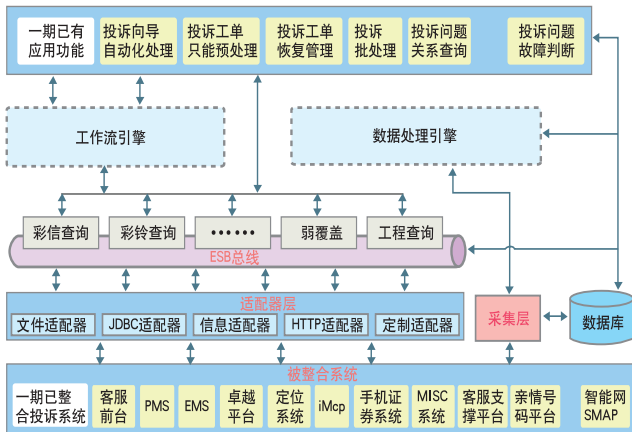
5)通过开发一些辅助工具如建言献策和状态管理等等,增强了系统应用的友好性,改进了系统与投诉处理人员的互动性。

二、网络投诉处理综合平台二期建设方案

1、系统体系架构

浙江移动网络投诉处理综合平台一期采用基于SOA的架构设计,基于J2EE技术,使用Websphere Application Server作为应用集成平台和Websphere Message Broker中间件作为企业服务总线,实现了12类投诉处理系统的三种整合方式(单点

登陆、普通整合和深度整合)。二期在保留原有体系结构的基础上,增加了数据处理引擎模块和与其他一些应用组件模块,网络投诉处理综合平台的系统框架图如下所示:



系统框架图

浙江移动网络投诉处理综合平台二期将继续继承一期系统的可移植性和扩展性,在系统架构上采用组件化的设计理念,并以横向组件插入的方式镶嵌到整体系统中,对整体系统架构不会做大的调整和变化。

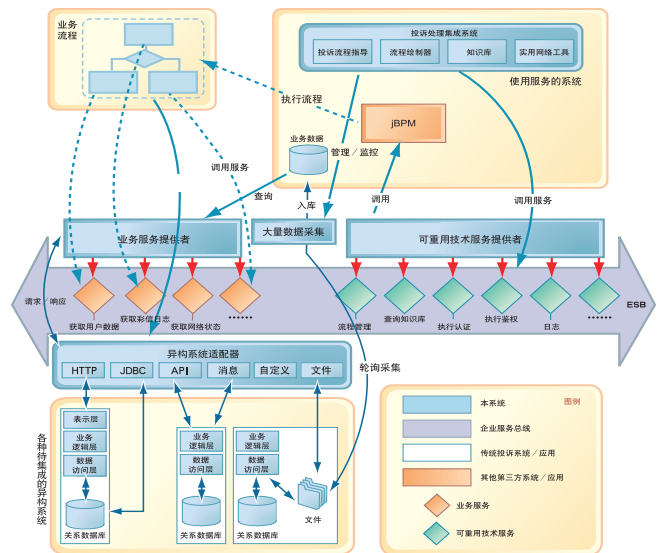
2、系统整合策略

2.1 系统整合策略分析

浙江移动网络投诉处理综合平台一期基于SOA方法论进行系统整合,通过适配器实现整合系统和平台间数据访问的透明性,在对这些数据业务进行封装,然后把封装好的服务通过ESB总线向上层应用或第三方应用提供服务。详细系统结构图如右上图所示:

浙江移动网络投诉处理综合平台以企业服务总线为中心,主要包含以下组成部分:

- 企业服务总线: 作为数据转换、交互、路由中心。
- 传统系统和应用: 各种待整合的传统投诉系统和应用。
- 适配器: 完成从传统系统和应用到ESB的数据桥接作用。
- 业务服务提供者: 通过ESB,从传统投诉系统中获取数据,实现各投诉流程的业务服务。
- 前置采集: 对于日志文件等大数据量的投诉数据,采用采集机预先采集的方式。
- 采集数据集中处理: 对前置采集的数据进行集中的预处理与入库,供业务服务提供者使用。
- 网络投诉处理综合平台: 通过使用ESB提供的服务,综



合各投诉支撑系统所能实现的功能,提供一个统一的处理入口。

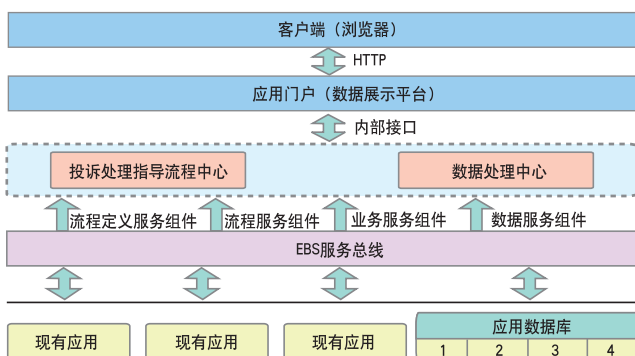
* 可重用技术服务: 实现各种可重复使用的技术服务,供投诉处理综合平台调用,也可供未来的其他系统使用。

浙江移动网络投诉处理平台一期围绕要求从整体的角度对投诉处理进行统一集中的管理目标,初步实现了一个适应由于网络故障、施工工程等等因素而导致的网络投诉快速解决方案。方案基于SOA方法论进行投诉专业支撑系统整合,其中实施策略又分为单点登陆、普通整合和深度整合三种。方案根据投诉处理向导节点需求,分别将不同的专业系统功能抽取为服务,通过ESB总线将不同的服务提供给上层应用使用。考虑到浙江移动网络投诉处理综合平台一期里的向导虽不同于普通的工作流程(用于协同不同部门、不同人员之间的工作流),是对投诉处理人员工作思路的一个整理和体现,但从实现上来看也是根据投诉处理向导的各个步骤,将分布于不同投诉支撑系统的功能组织起来。这样整合系统的关键之一是如何梳理并建立投诉处理向导,基于向导,统一管理投诉处理数据,并驱动底层数据集成,形成统一集中的投诉数据中心,为投诉应用提供统一的数据接口。网络投诉处理综合平台二期工程建设中,将利用一期建设中的系统整合经验,针对不同的被整合投诉处理IT支撑系统,分别提供相应的整合策略和系统整合适配器。

2.2 基于流程组件的面向服务的整合

在面向服务的设计中,不能基于业务实体设计服务。相反,每个服务都是管理一组业务实体中的操作的完整单元。例如,彩

铃查询服务将响应来自任何其他系统或需要访问用户号码相关彩铃信息的请求。彩铃服务可以处理查询彩铃定制信息请求以及删除铃声。彩铃服务拥有所有与它管理的铃声有关的数据，并且能够代表调用方进行其他服务查询，以提供统一的彩铃服务视图。这意味着服务是一个管理器对象，它创建和管理它的一组组件。因此如何实现服务组件以达到最大化的重用是面向服务进行整合的关键。考虑到浙江移动网络投诉处理综合平台整合的服务组件是为投诉处理向导服务的，因此服务组件直接面向是向导节点，向导节点通常对应单个服务组件或复合服务组件进行功能处理。因此设计如下基于流程组件的面向服务的整合策略示意图如下：



a) 投诉处理指导流程中心

投诉处理指导流程针对的是一个相对比较单一的任务，不同于IMEP里面的工作流，基本上由一位人员在较短时间内完成，通常表现的是一个思路的分析过程，通过相关步骤来定位分析投诉问题，保证大部分问题的正确解决，因此，投诉处理流程中心虽不是一个工作流，但也为投诉问题处理提供投诉处理指导流程建模、分析、管理等全生命周期的管理职能，负责服务组件在不同的流程节点进行分配，并且通过流程节点驱动数据中心进行数据的抽取与转移。

数据处理中心对外提供的的数据抽取接口和服务组件提供的发布消息接口挂在流程的各个不同节点上，将每个环节要完成的核心业务逻辑与为支持集成而由系统自动完成的数据抽取、消息发布等逻辑隔离开来，从而实现集成服务组件的接口通用性和可重性，增强了系统的适应投诉处理流程变化能力。

b) 数据处理中心

数据处理中心主要完成对经过SOA整合的数据进行二次加工功能，在数据的传递过程中，数据中心还将通过数据获取与发

送接口完成数据格式的转换、打包、拆包等步骤。数据中心通过ESB服务总线将应用提供的数据收集上来进行整合分析转换进行二次发布，使得投诉数据在更高层次上为投诉业务进行服务。

3、系统整合方案

根据当前各类不同投诉处理系统的实际使用情况以及整合难度，投诉处理综合平台对各支撑系统的集成整合分成两类：深度整合和单点登录集成。

3.1 单点登录方案

投诉处理综合平台提供到目标投诉处理系统的链接及单点登录，用户可以在平台中启动并切换到对应的投诉处理系统。

浙江移动投诉处理综合平台的单点登录体现在两个方面：一是投诉处理综合平台作为统一信息平台的一个门户接入，投诉处理综合平台的用户可以通过平台单点登录进入；其次用户登录投诉处理综合平台后，可以直接访问投诉处理综合平台整合的各投诉处理系统，无须重新输入另外的用户名和口令。

3.2 深度整合方案

深度整合就是对原投诉系统的功能在投诉处理综合平台上重新实现，并提供一些新功能的增加，对原系统中有缺陷的功能模块做一些修改。它是系统整合中最深层次的整合。通过深度整合，可以实现系统整合后系统显示风格的统一，用户认证、授权的统一，摒弃掉其他系统业务模块层次的变动对本系统的影响。

3.3 基于ActiveX控件的嵌入式整合方案

对于整合系统中存在的业务非常复杂、数据量大的一些C/S系统，应用开发商需对整合的模块进行ActiveX组件化，平台把这些ActiveX组件放在应用服务器上，客户端调用时就直接把ActiveX控件下载到客户端，并在浏览器中展现和使用。

3.4 基于平台定制Web服务调用方案

平台提供一组Web服务接口给被整合系统调用，被整合系统通过参数的形式把数据传给服务，服务再对这些数据进行处理和保存，实现了应用的数据整合。平台再根据这些数据进行业务服务包装。

3.5 系统整合适配器

东信提供了面向标准协议的Adapter，如Web Services、HTTP、COM、CORBA、JDBC、EJB、JText、email、MQ、JMS等，能够更方便地实现和多种标准协议的连接和互通，实现数据抽取、应用整合的功能。此外，针对特殊接口的被整合系统，可单独定制相应的适配器。■



金证基金 CRM 系统 KingCRM 解决方案

深圳金证科技股份有限公司 / 文

基金市场已经成为我国金融领域里的重要组成部分。随着基金市场的蓬勃发展，众多基金公司纷纷成立，开放式基金的发行速度也越来越快，销售渠道和销售手段越来越丰富，单只基金的规模也越来越大，持有基金的客户也变得越来越多，开放式基金已经得到了广大投资者的认可，全民购买基金的投资趋势逐步形成。

然而在这种竞争日渐激烈的市场环境下，对基金公司本身而言却要面临许多挑战，要想在残酷的竞争中立于不败之地，在有限的市场里发展更多的客户，那么在变化的市场中不断改革创新营销和服务手段，进行有效的客户关系管理就是基金公司需要关注的首要问题。

金证基金 CRM 系统 KING CRM/FUND 已经为许多基金公司提供了理想的答案。

产品概述

金证基金 CRM 系统 KING CRM/FUND 1.0 是一套低成本、高性价比的营销服务解决方案。系统架构成熟稳定、功能完备、灵活性高、扩展性好，具有投资少、实施周期短、易维护的特点。

企业级应用

KING CRM/FUND 1.0 是面向基金公司的企业级应用系统，能够与基金公司现有及将来其他业务系统方便对接。该系统集中部署一套，可以满足基金公司总部、分公司、各片区使用。

功能全面

KING CRM/FUND 1.0基于市场营销的理念,提供了基金公司在市场营销过程中的完善的客户管理、渠道管理、产品管理、市场管理、销售管理、客户服务、决策分析、知识管理、作协同、绩效考核等整套功能。

灵活的树状结构与菜单分组

KING CRM/FUND 1.0采用树状层次构造,支持销售渠道、部门、人员、地域的树状管理,用户可无限的自定义深度和节点;支持菜单分组,不同权限的角色进入系统后根据授权动态生成菜单;基于角色分配功能权限。

多视角的资源整合

KING CRM/FUND 1.0从数据中心层面,围绕客户、渠道、产品等构建;从功能层面,围绕营销理念组织系统;从用户层面,按用户使用者的角色对功能集合进行组织。

个性化的首页展示

KING CRM/FUND 1.0根据不同用户的工作特点、性质及其不同的访问视角,定义不同的角色,从而拥有不同的首页展示。用户进入系统时,系统主动呈现给用户最为关心的和经常使用的一些信息和链接,便于用户使用自身关注的常用功能。

单点登录

KING CRM/FUND 1.0提供从OA系统登录本系统的支持。用户在登录OA后,可以点击OA中的相应链接,直接登录本系统,无须再输入用户名和密码,从而达到一次登录,可同时访问OA与本系统的目的。而且可两个系统使用相同的用户名、密码,使用者无须再为需记住多套用户名、密码而烦恼。

先进的技术架构

KING CRM/FUND 1.0采用先进的B/S体系的J2EE架构以适应基金营销服务工作分散和移动性需要,同时J2EE架构还方便系统的跨平台使用,零客户端安装(只要客户机上有IE等浏览器即可访问)大大减轻了系统维护的负担。

√客户层: IE 浏览器

√WEB层: JSP/Servlet

√应用层: WEBSHERE/WEBLOGIC/TOMCAT等?

√数据库: Oracle

√商业智能: BO+ 水晶报表

设计原则

√以投资者为中心

√以金融工具为主轴

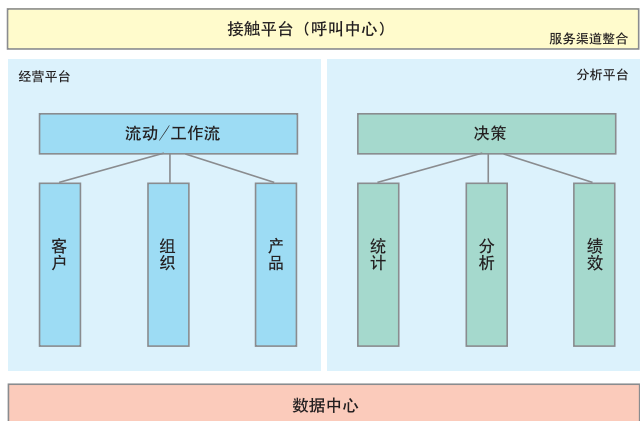
√以基金公司组织结构为载体

√以业务流程控制为手段

√以绩效考核为目的

√以决策分析为核心

总体模型



系统组成

KING CRM/FUND 1.0由部署于J2EE架构上的WEB应用系统与后台的数据采集系统两大部分构成。WEB应用系统包括数据库服务器、应用服务器/WEB服务器、IE浏览器客户端;数据采集系统为后台支撑系统,主要负责从多个采集数据源(包括TA系统、客服系统、资讯系统、OA系统等)采集、清理数据,以满足应用系统运行所需。

数据中心

KING CRM/FUND 1.0运行的基础数据主要来自基金公司的TA系统数据库,也可根据具体业务需要从财务系统、OA系统、原有客服系统等采集数据,集中管理。数据采集模块负责把各采集的数据按时/按条件采集、整理并保存到系统运行的数据库中。

系统功能

客户管理

客户信息集成、客户分类、客户分级、组合查询、客户分配与客户关怀等。

渠道管理

树状渠道组织结构、渠道信息集成、渠道分析、渠道分配与渠道客户关怀等。

市场管理

媒体管理、合作伙伴管理、市场推广活动管理、客户问卷调查、交易调查、竞争信息管理、宣传资料管理和礼品管理等。

销售管理

销售机会管理、客户需求管理、决策树管理、投资建议书管理、合同管理与销售总结等。

客户服务

服务频道管理、发送对象分组管理，支持对邮件、短信的立即或定时发送等。

决策分析

客户构成分析、保有份额分析、交易分析、贡献分析、盈亏分析、净值分析等。

知识管理

文档库管理、基金产品管理、信息动态等。

工作协同

日程管理、工作简报、工作周报、会议纪要、通知管理等。

绩效考核

销售数据申报、申报数据审核、奖励计算、奖励结转等。

KING CRM/FUND 1.0 特色和优势

特色	优势
完善的系统设计	
模块化的设计	系统提供的应用模块，用户可以有选择地运用，每个软件之间相互独立，模块接口开放、明确，模块的增加和对模块的修改不应对其他模块产生影响，并能不影响系统运转的情况下做到模块更新、模块加载，模块之间的接口明确，开放，规范。
标准化和开放性	系统设备平台、系统软件、应用软件的建设遵循业界统一标准，具有开放性，保证所选不同型号的硬件设备能相互兼容，并且最大可能利用客户现有设备资源，尽量不增加主机设备投入。
支持多种数据来源	支持分布式数据管理，支持多数据源间的访问连接，支持文本文件方式的数据源以及数据库作为数据源。
支持多种数据发布接口	提供信息发布 API 接口支持多种方式发布信息。
功能全面	基于市场营销的理念，提供了基金公司在市场营销过程中的完善的客户管理、产品管理、市场管理、销售管理、客户服务、决策分析、知识管理、工作协同等整套功能。

多视角的数据展示	从数据中心层面，围绕客户、渠道、产品等构建；从功能层面，围绕营销理念组织系统；从用户层面，按用户使用者的角色对功能集合进行组织。
树型组织结构管理	采用树状层次构造，支持销售渠道、部门、人员、地域的树状管理，用户可无限的自定义深度和节点，并有直观展示
技术领先	
J2EE 架构	先进灵活的 J2EE 架构
多数据来源	可以从不同的数据源中进行数据采集整理，以插件技术及灵活的配置支持新的数据源采集
报表引擎	采用通用报表设计与引擎，进行灵活定制报表展现
多重系统安全机制	
数据加密	中间加密传送，关键数据加密存放，保证数据机密
权限管理	菜单权限和数据权限分别控制，对用户可访问的功能与数据进行多级控制



平台和运行环境

支持数据库

Oracle 8i/9i

支持多种操作系统

Windows、UNIX、Linux、AIX、Solaris 等

支持多种应用服务器

Websphere、Weblogic、Tomcat/JBoss 等

目前已经采用该系统的用户

南方基金、嘉实基金、融通基金、鹏华基金、天弘基金、诺安基金、长城基金、天相投资顾问公司。 ■

适用于统一信息平台项目 WebSEAL 性能调优解决方案

环亚时代 蒲志林 / 文

一、概述

1、WebSEAL 概述

TAM(IBM Tivoli Access Manager for e-business)是广泛用于电子商务、企业门户以及分布式应用程序的统一安全管理解决方案，功能非常强大。WebSEAL是TAM的前端Web服务器，它能为后端受保护的Web对象提供SSO(Single-Sing-On单点登录)，并将后端Web应用服务器资源合并到其安全策略中。

WebSEAL通常作为反向Web代理，从Web浏览器接收HTTP/HTTPS请求，并转交给后端的Web服务器，后端Web服务器返回的内容通过WebSEAL传递给Web浏览器。它替后端服务器把守着大门，所有需要通过WebSEAL的请求由Police Server评估，以确定是否授权用户访问所请求的资源。

WebSEAL提供以下功能：

支持多种认证方法

内置式和插件式体系结构都提供支持多种认证机制的灵活性。

接受HTTP和HTTPS请求

通过WebSEAL联结技术集成和保护后端服务器资源

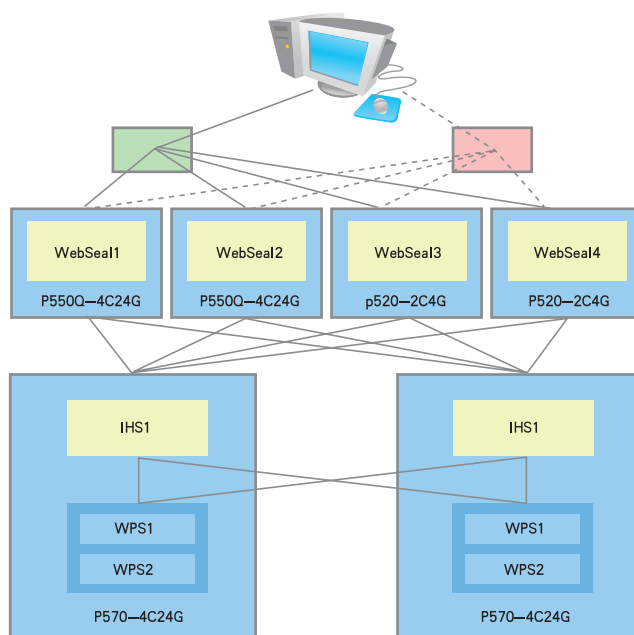
管理用于本地和后端服务器Web空间的细粒度访问控制

所支持的资源包括：URL、基于URL的正则表达式、CGI

程序、HTML文件、Java servlet和Java类文件。

作为反向Web代理执行

提供SSO单点登录

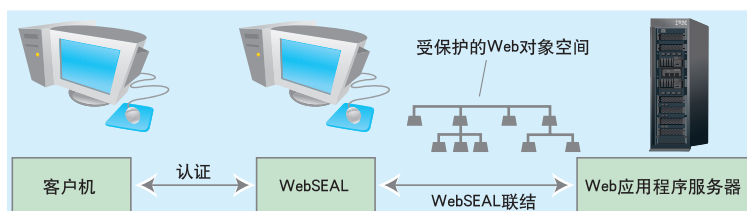


2) 四台P550服务器做WebSEAL服务器，主机配置4CPU（双核1.9G）24G内存，硬件性能非常强；

3) 两台Portal服务器，2*2群集（两个节点，每个节点跑两个Portal Server服务）；

4) 用IHS做软负载。

从上面的架构图上可以看出，WebSEAL是整个系统的入口，整个平台的认证以及资源访问都需要经过WebSEAL，它的性能直接决定了整个平台的性能表现，对WebSEAL的性能调优意义重大。



2、系统结构

为了更好地演示调优过程，以某移动公司的统一信息平台为例，平台的系统架构图如右上图：

1) 用F5四层交换机做负载均衡，双机主从备份；

二、调优前系统表现

整个平台安装完成后，没有对WebSEAL进行调优，对平台进行了压力测试，并割接了部分用户进行试运行，试运行期间的性能表现如下：

1) 登录时间长：门户用户从登录到首页全部展现出来，响

应时间 15-20 秒（基本没有什么并发压力的情况下）；

2) WebSEAL 服务器 CPU 占用率极其脆弱，波动很大：

在没有任何压力情况下，1 个用户登录，WebSEAL 的 CPU 马上下降 10-15%，有时能从空闲 90% 一下子下降到空闲 0%

3) 同时 5 个用户登录，WebSEAL CPU 马上出现 100%，并一直保持不下，响应时间越来越长，直到系统宕机。

三、WebSEAL 调优

1、操作系统调优

操作系统操作系统目前的参数值列表：`#ulimit -a`

```
time(seconds) unlimited file(blocks) unlimited data(kbytes)
unlimited stack(kbytes) 4194304 memory(kbytes) unlimited
coredump(blocks) unlimited nofiles(descriptors) unlimited
```

```
#vi /etc/security/limits
```

```
data=-1
```

```
rss=-1
```

```
fsize = -1
```

```
# export LDR_CNTRL=MAXDATA=0x80000000
```

调整 TCP 参数：

```
# no -algrep tcp_nodelayack
```

```
tcp_nodelayack = 0
```

```
# no -p -o tcp_nodelayack=1
```

注：tcp_nodelayack 提示 TCP 发送一个立即确认信息，而不是延迟确认信息。在高并发和网络交互非常频繁的环境中调整这个参数非常有效，在某些复杂的网络环境中可以极大地提高网络性能。如果您更改了这个参数却没有提高性能，那么建议将该参数还原为先前的值。



2、WebSEAL 应用调优

主要对以下文件进行操作：

参数	缺省值	调优值	备注
worker-threads	50	500	不能调的过大，300-800 为佳，通过以下命令查看当前工作进程数： pdadmin -a sec_master -p password server task webseal_server stats get pdweb.threads
user-and-group-in-same-suffix		Yes	
policy-cache-size	32768	65536	
max-entries	4096	10000	
cache-enabled		yes	
cache-user-size	256	2560	
cache-user-expire-time	30	3600	
cache-group-membership		yes	
cache-group-size		64	
cache-group-expire-time		300	
cache-use-user-cache		yes	
user-session-ids	no	yes	设为 yes 可以用命令清除用户 session
max-size		20000000	日志文件大小：20M
requests	yes	no	去掉 request 日志
referers	yes	no	去掉 referer 日志
agents	yes	no	去掉 agent 日志
logaudit	yes	no	去掉审计日志
io-buffer-size	4096	8192	去掉 agent 日志
Ltpa_time	3600	7200	去掉审计日志

注：修改前注意备份，需要重启 WebSEAL 服务。

四、调优后系统表现

系统整体调优后，性能指标如下：

- 1) 注册用户数：1 万
- 2) 全天活动用户数：7566
- 3) 全天累计登录人次：25719
- 4) 每秒最大登录人次：10
- 5) 每分钟最大登录人次：151



6) 按小时统计用户登录人次: 早上 8: 30 上班, 8: 00-9: 00 是系统登录访问高峰期, 早高峰登录人次 5709, 最高能到 6500, 过半的用户是在这段时间登录系统的。

1	2009-06-08 00	24
2	2009-06-08 01	2
3	2009-06-08 02	7
4	2009-06-08 03	1
5	2009-06-08 04	1
6	2009-06-08 06	19
7	2009-06-08 07	742
8	2009-06-08 08	5709
9	2009-06-08 09	2797
10	2009-06-08 10	2259
11	2009-06-08 11	2116
12	2009-06-08 12	514
13	2009-06-08 13	000
14	2009-06-08 14	1564
15	2009-06-08 15	2811
16	2009-06-08 16	2201
17	2009-06-08 17	2090
18	2009-06-08 18	1454
19	2009-06-08 19	443
20	2009-06-08 20	172
21	2009-06-08 21	146
22	2009-06-08 22	74
23	2009-06-08 23	38

7) 用户从登录到首页全部展现出来, 响应时间 3 秒左右;

8) DB2/IDS 服务器: CPU 占用率不超过 5%, 内存占用率基本无变化, 比较平稳; DB2 连接数很稳定, 保持在 70-120, 不超过 150; LDAP 两个服务器的连接数加起来不超过 50 个。

9) Portal 服务器: CPU 占用率不超过 30%, 平均 10-15%, 内存占用率基本无变化, 比较平稳。

10) Webseal 服务器: CPU 占用率峰值 50%, 平均 15-25%, 内存占用率比较平稳, 无太大变化。

采集日期: 2009-06-08 周一 (平时登录人数比此略低 5-10%)

五、总结

1) WebSEAL 默认版本 6.0.0.0 不稳定, 建议打补丁到最新稳定版本 (2009.06.09 WebSEAL 稳定版本是 6.0.0.22, 最新稳定版本可以查看 TAM 的 Buglist)。

2) WebSEAL 服务器硬件配置最佳方案, 按优先级排列:

???独立服务器: 采用性能一般的独立的刀片机或者 PC Server, 2CPU 4G 内存的配置即可

性能超强的服务器划分独立的 LPAR 逻辑分区: 每个 LPAR 按 2CPU 4G 内存的配置划分

一个物理主机上安装多个 WebSEAL: 用不同的 IP 或者端口提供服务

第一种方案最优, 性价比和稳定性俱佳。

3) WebSEAL 的性能跟很多因素都有紧密的关联关系: 网络、操作系统、硬件配置、LDAP Server、应用服务器性能等, 调优需要从多方面综合考虑

4) 在高负载大并发的应用环境中, 对 WebSEAL 性能影响较大的参数有:

Webseal 服务器的 tcp_nodelayack 内核参数

此参数调为 1 能提高网络传输性能, 对于这种高吞吐量的网络请求有比较明显的效果此参数调整后 Webseal CPU 占用率明显下降, 并行支撑能力成倍提升

Webseal 内存参数

Webseal 是 C 的内核, 调整 JVM 参数对它是无效的, 默认内存分配 256M, 远远不能满足系统需求, 把 LDR_CNTRL 调为 2G 后 CPU 占用率波动降低很多, CPU 占用率总体也有所下降。

Webseal 日志参数

关闭 Webseal 的 referers、agent、requests、logaudit 日志, 还有 trace 日志, 并发高的访问系统, 写日志会占用很高的系统资源。

应用响应时间

Junction 链接的应用资源如果很长时间没有返回结果, 就会一直占着 WebSEAL 的链接线程资源, 对 WebSEAL 的 CPU 和内存资源都消耗很大

5) WebSEAL 的参考性能指标:

一个 2CPU 4G 的 PC Server, 假设上面只跑了一个 WebSEAL 的服务, 所有的参数均已调到最佳值, 按照河南移动或者北京联通的业务规模 (注册用户 1 万, 全天累计登录人次 2.5 万人次, 早高峰累计登录 5 千人次, 活动用户比例 3/4), 这样的一台 WebSEAL 服务器大约可以支撑 3000-4000 千注册用户的访问量。■

注: 这个数值属于经验估计, 每个项目由于硬件配置和业务规模不一样, 这个值会有比较大的出入, 建议通过压力测试实际评测。

人员整合 与协作解决方案

神州数码系统科技公司 · IBM 产品本部供稿

- 展示界面可操作性差？
- 资源分散？
- 用户体验差？
- 无法实现资源重用？
- 人员协作差？
- 知识资源无法实现共享？
- 无法快速响应业务的变化？
- 服务水平低？

面对不断增长的海量的信息，迅猛发展的信息技术，您的企业是否遇到过上述问题？IBM 人员整合与协作解决方案帮助您扫除困扰。

通过IBM的人员整合与协作解决方案，您可以构建一个统一的、动态的工作环境，实现人员间的互动——包括员工与员工之间、上下级之间以及跨部门之间的协作，提高工作效率的同时，增进客户与员工、员工之间以及其他员工上下级之间的关系。它提供全面集成的协作功能以及世界一流的人员集成平台。集成协作环境系统架构如下图：

底层数据源构成了人员整合与协作的数据信息支撑环境，包括一切运营管理过程中的关键数据，这些数据主要存储在关系数据库（IBM DB2）、复合型数据库（Lotus Notes）和文件系统等。应用系统需要对这些数据和信息以某种方式进行分类存储和管理，并根据数据和信息之间的关联关系把它们按照更高的层次需要，全新地组织和存储起来，以便于访问和重复利用。

在中间层是企业应用平台，用以支持应用层的功能实现。企业应用平台基于IBM标准的电子商务基础服务技术（J2EE/Web Service/LDAP/EAI等），可以提供与其它系统相应的接口，通过Web Service/XML进行数据的交换。协同办公应用平台 Lotus Domino提供了企业级应用扩展能力和高可靠性的平台服务。通过基于Domino平台的特定协作组件扩展的协作服务层可以为企业应用和员工提供全方位的协作式工作环境（电子邮件、即时消息、 workflow、项目协作、文档服务、搜索引擎等），可以在上层



漫画/杞人

的应用中利用这些基础服务，实现真正的协作互动。

重点解决了以下问题：

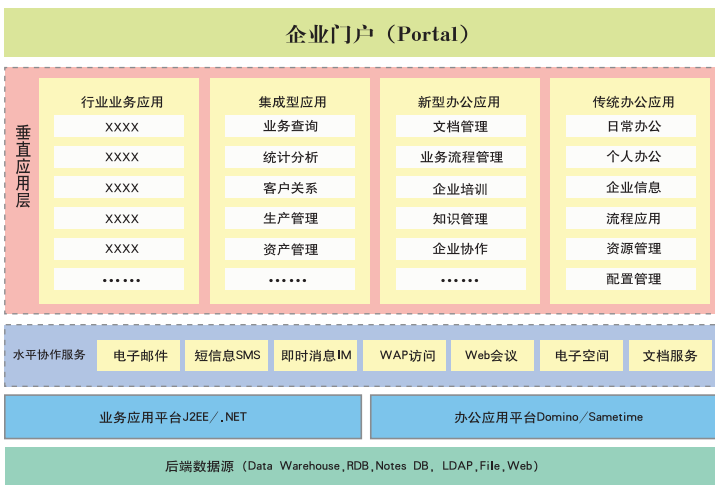
- Web 实时互动协作
- 网络化文档资源管理
- 内容编目和信息检索
- 可定制的流程管理
- 各系统之间的集成性

最上层是企业门户，实现用户界面层的整合，可以帮助企业将人员、流程和信息链接在一起。门户是通过统一的、标准的用户界面，连接到广泛的应用、内容和数据的集成方法。用户通过浏览器能够找到相关的人、应用程序和内容。用户根据自己的角色能够查看到相应的资源，并且透过门户可以访问到多种异构数据源，包括关系数据库、多维数据库、文档管理系统、电子邮件系统、新闻和各种文件系统。

实现如下功能：

- 对企业信息的分类和连接
- 根据用户工作角色不同提供不同类型信息的导航

- 用户可以根据自己的需要定制自己的起始界面
 - 提供友好的、易于使用的用户工作界面
- 解决方案架构及核心技术
此解决方案基础架构如下所示:



1. 人员整合技术——门户 (WebSphere Portal)

门户技术是 IBM 按需应变工作平台的核心组成部分, IBM WebSphere Portal 提供了可扩展的框架, 使企业各种应用、内容、人员和流程可以实现互动。自助服务特性使最终用户可以个性化或组织他们自己的门户网站视图, 管理自己的配置文件, 以及发布并与其他同事共享文件。WebSphere Portal 还提供了其他服务, 例如: 单点登录, 安全性, 文件管理, Web 内容发布, 搜索, 个性化, 协作服务, 企业应用集成, 移动设备支持和站点分析等。WebSphere Portal 是业界综合性最强的门户网站解决方案, 已经成为了事实上的电子商务体系结构标准。主要技术如下:

与后台系统整合技术。Portlet 是门户网站的核心。"Portlet" 指门户小应用程序, 它是可以重复使用的组件, 提供对应应用、基于 Web 的内容及其他资源的访问功能。通过 Portlet 可以访问网页、Web 服务、应用和合成内容。在解决方案中提供了多个层次的支持: (i) 提供就绪的 Portlet 组件, 实现对已有系统的快速集成; (ii) 提供 Portlet 模板, 对于标准技术的应用 (如 JSP、Servlet、HTML、XML), 实现 Portlet 的快速开发; (iii) 提供面向业务人员的 Portlet 开发工具, 在生产环境中, 无需编码, 可以让业务人员通过图形化界面工具, 开发出 Portlet (数据库、

JCA 连接、Domino 等); (iv) 另外, 还提供了完整的 J2EE 集成开发环境, 提供相关的开发向导, 方便、快捷地实现门户小服务程序 (Portlet) 的开发; 提供集成测试环境, 支持本地和远程单步调试, 以方便开发人员, 节省开发周期。

在用户操作界面上实现各种应用间的互操作性 (C2A)。在界面上点击一个应用 (Portlet) 中的数据内容, 数据可以自动传送到其它的应用系统 (Portlet) 中去进行相关的处理。

2. 协作技术

- **工作流定制:** 使组织中的各项工作均处在有效的管理和监控之下; 并对各种知识信息进行积累, 为各级领导的决策提供有效的支持; 进而建立企业知识管理协作平台。

- **文档管理:** 文档管理功能为用户提供一个安全的文档容器, 提供文档分类、文档搜索、全文检索、版本控制、编辑锁定等功能, 还与系统集成, 与公文流转流程结合, 产生新的应用, 为下一步知识管理打下良好的基础。

- **Web 内容管理:** 创建、批准、管理和发布 Web 内容。该过程的步骤包含定义内容类型、角色、出版物选项、目的地规范和工作流进程。

- **人员协作功能:** 发挥人员及其个人专长的力量, 将其作为组织中最宝贵的资产。

- **团队工作空间:** 为企业节省培训费用和缩短培训周期, 立即提高工作效率。

- **实时协作:** 与现有应用整合, 使在不同工作地点的人员进行有效及时的沟通, 快速找到相应的专家, 快速的与客户和合作伙伴联系, 召开 Web 会议。可以拓展已有的电子邮件的沟通交流的方式, 大大提高员工的工作效率和响应能力。

- **远程教学功能:** 可以让员工实现随时随地学习, 使组织提高即时应答能力, 并支持员工积极培养技能以提高工作满意度。

- **企业移动办公技术:** 随时随地随意访问企业内部信息系统、实现日常移动办公是目前最受欢迎的无线/移动协作方式。

- **集成其他数据库:** 通过建立跨平台综合信息查询引擎, 提供企业系统的综合信息查询, 体现系统信息无缝衔接的特点, 实现异种数据库的信息组合、模糊查询功能。

- **搜索引擎技术:** 提供联邦搜索异种数据源的搜索功能。利用企业已有的熟悉用户界面, 通过单一访问的入口, 可以快速有

效的访问企业相关数据和信息。

3. 系统安全性

提供了完善的安全管理功能,包括用户身份的管理、认证、授权、单点登录...

4. 系统扩展性

使整个系统在满足现有的业务要求的同时,支持今后业务的扩展性,包括多个方面需要进行扩展,如:系统服务能力的扩展及新功能的扩展。

5. 网站分析

可捕捉和分析 Web 站点数据,从而提供关于访问者流量、访问者行为、站点使用、站点内容和站点结构的有用报告。您可从预定义的报告元素中构建报告,也可收集特定的信息来量身定制属于您的报告。

6. Web Service 技术

提供集成多个 Web Services 提供者的功能,以及通过 Web Services 发布集成的信息。通过对该技术的支持,实现企业各应用系统之间有效的整合。

人员整合与协作解决方案的价值

集成协作环境改进工作效率,提高劳动生产率,同时通过优化"员工与员工"和"员工与业务"的业务流程,改善"员工与客户"和"员工与上级"的业务关系。

建立动态办公平台的意义在于以下几个方面:

- **工作方式的改变:** 实现员工、企业、客户、供应商之间简单、统一和个性化的沟通,建立更紧密的合作平台。

- * **工作效率的提高,运行成本的节省:** 使员工摆脱企业中与自身不相干的复杂流程,把复杂的流程简单串联起来,从而提高企业内部的工作响应速度、积极性、工作效率和劳动生产率,节省运行成本。

- **整合的平台:** 建立集成的组织门户或平台,把各个孤立分散的机构网站集成为完整统一的高效的集合,同时关注有增加价值的活动。

- **端到端的解决方案:** 通过分析业务,快速改进信息流和协作方式,建立一个端到端行业范围的结构。

- **系列化的产品和一条龙的服务:** 公开的、基于标准的解决方案融合了 IBM Lotus 和 WebSphere 门户系列软件,以及 IBM 的业务咨询和应用管理服务的实施队伍。

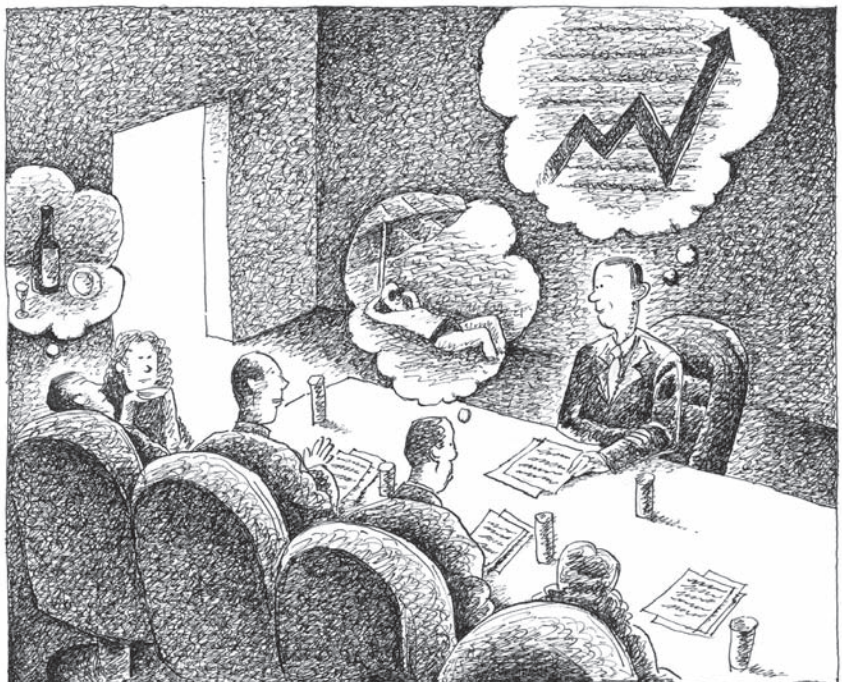
此系统起到"知识管理"的平台作用。面向知识管理的系统逐步把知识管理原则与实践融入每个员工的日常工作中去,它不仅模拟和实现了工作流的自动化,更模拟和实现了工作流中每一个单元和每一个工作人员运用知识的过程。

扩展系统"电子学习中心"的作用,建立学习型的组织。此系统以知识管理为核心,围绕实现知识管理的信息基础策略、信息共享策略、信息挖掘策略、信息展示策略、信息推荐策略等各个方面多层次多方式的发展。

硬件产品: IBM System P570、DS4800 磁盘阵列

操作系统: AIX 5L

软件产品: WebSphere Portal Server、WebSphere Application Server、Lotus Workplace、Lotus Domino/Notes、Lotus Sametime、Lotus e-Learning ■



漫画/杞人

WAS 7 系统管理的惊鸿一瞥

富通东方技术总监 张东焕 / 文



我从 2000 年开始使用 WAS，第一个项目是农行网上银行，WAS 3.0，那个时候 WAS 的管理界面是 Java 客户端。WAS 给人的感觉是系统管理方面显得比较笨重，不过在 WAS 5 开始，这方面的改进非常显著。这不，让人期待的 WAS 7 发布了。系统管理方面又有不少新特性。让我们看看到底有哪些改进（我选了几个我认为在系统管理方面能带来显著便利的新特性）。

1、管理代理 (Administrative Agent)

还是讲一个故事，前一段时间银联数据使用 WAS，采用的是应用服务器节点的类型。当 WAS 接受大量的应用请求时，系统出现 CPU 使用过高的情况，这时想查看 WAS 的性能信息，但是用 WAS 管理控制台无法登陆，因为作为应用服务器节点类型，管理和应用都使用同一个应用服务器，共同使用一个 Web 容器，应用将线程池占满了，WAS 管理控制台也就无法登陆了。现在好了，有了新的特性，管理代理。

WAS 7 引入了新的可选管理服务器，称为管理代理，以通过以下方式处理此类挑战：

多个独立应用程序服务器可以向单个管理代理注册，一经注册，即可通过单个管理代理监视和控制应用程序服务器。

管理代理成为了用于管理同一计算机上的多个独立应用程序服务器的中心入口点。如果用户管理的拓扑包含单台计算机上的大量独立应用程序服务器，则这种方法是极为重要的。

所注册的服务器的管理服务被合并到管理代理中，从而减少了以前的重复服务的内存占用空间。

管理代理执行的管理功能在所注册的应用程序服务器之间保持隔离，从而使得用户能够针对所管理的服务器集合中的特定服务器执行操作。

图 1. 管理代理

管理代理旨在作为补充 WAS 基

本拓扑的选项，其中的独立应用程序服务器继续为应用程序请求提供服务。只有服务器中的管理服务才合并到管理代理中。对于每个向管理代理注册的 WAS 基本概要，将在管理代理中创建一个管理子系统，以表示该概要的新管理入口点。

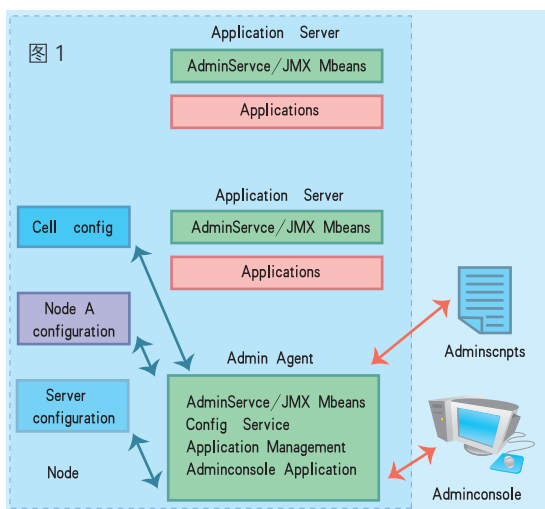
2、作业管理器 (Job Manager)

前一段时间去过华夏银行，他们的生产及测试环境共使用十多套 WAS，他们自己有时也不明白到底有多少，所以管理的时候需要登录每个机器上或者各自的 WAS 管理控制台，管理起来是相当的不容易。每个 WAS 做了哪个操作，结果如何，都没有记录，很不方便。现在有了新的特性，作业管理器。

单个作业管理器可以管理大量作为托管节点向其注册的 WAS 基本版本和 Network Deployment 拓扑。在此模型中，与

直接针对独立服务器和部署管理器调用管理操作不同，而是将这些操作作为管理作业提交给作业管理器。然后托管节点按预定义的间隔获取这些管理作业。在 WAS 7 中，作业管理器支持将服务器、集群和应用程序的配置和控制（启动/停止）操作以及执行 wsadmin 脚本作为管理任务。

通过指定作业激活或过期时间、重复执行模式、作业完成时的电子邮件通知等等，可以对作业提交过程进行自定义。



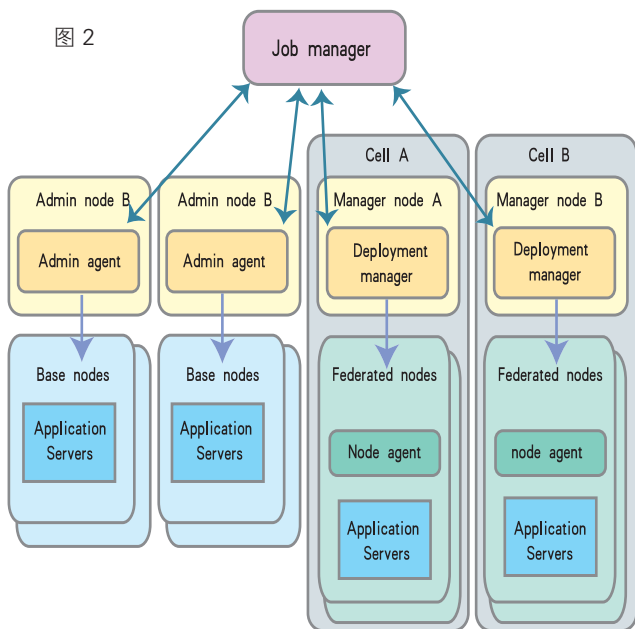


图 2. 作业管理器

这种新的灵活管理模型提供了许多优点：

作业管理器旨在对 Network Deployment 和基本版本拓扑形成补充，不需要对现有的节点进行重新配置。单个作业管理器可以管理数百个节点，与 Network Deployment 拓扑中由部署管理器拥有托管节点不同，给定的节点可以由多个作业管理器管理。

作业管理器管理的拓扑保持它们的自主性，包括它们的安全配置，从而可以使用现有的管理流程直接对其进行管理，例如脚本或管理控制台。

这种模型支持跨多个定义为一个组的托管节点执行协同管理操作。

异步作业提交模型可以促进地理上分散并且只能通过低带宽、高延迟网络进行访问的节点的管理。

3、业务级应用程序 (Business-Level Application, BLA)

以前将应用组装在 EAR 文件中，但是有很多共享库没有地方统一定义，我们需要在每个节点上部署这些共享库，并在通过 CLASSPATH 让应用去识别。现在有了业务级应用程序，管理整个应用变得更加轻松了。

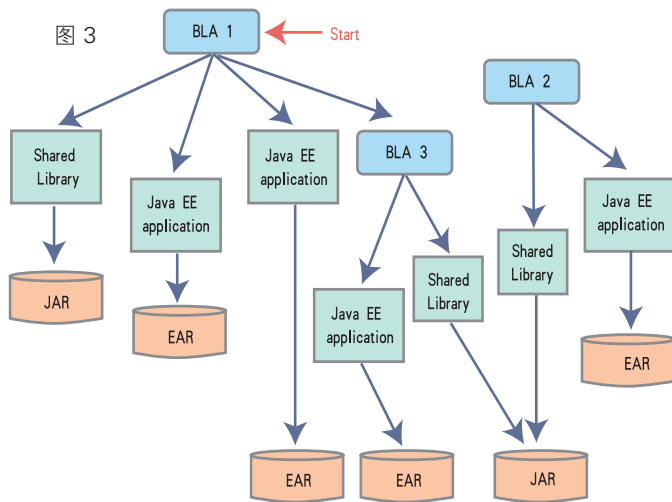


图 3. 业务级应用程序

在 WAS 7 中，BLA 组合可以包括一个或多个 Java EE EAR 文件和 Java EE 应用程序所依赖的共享库。BLA 定义为其构成部分提供了操作语义。例如，对某个 BLA 调用启动操作将导致启动该 BLA 中定义的所有企业应用程序和共享库。此外，BLA 体系结构支持按引用的递归组合，从而促进 BLA 和各个所部署构件的层次结构组装。此机制支持为特定的业务用途而定义某个 BLA，同时允许将其完整地包括在许多其他组合中。

BLA 定义将应用程序二进制与它们在特定部署实例中的配置分离，从而促进应用程序二进制跨多个应用程序定义的重用。可以多次配置单个 EAR 文件或库 JAR，使其成为一个或多个 BLA 的一部分。

通过其可扩展的部署模型，BLA 体系结构提供了对 WebSphere 应用程序部署的重要增强：共享库管理。

4、基于属性的配置

我们都知道从 WAS 5 开始，WebSphere 配置参数存储在一组配置文件中，大部分是 XML 格式。虽然可以使用 wsadmin 脚本或者控制台进行配置参数的修改，但这些相对来说有些复杂。我们常常需要从开发环境复制某些参数配置到生产环境中，或者从某一生产环境复制某些参数到另外一个生产环境中去。这种方式不是很方便。现在有了基于属性的配置，这类工作变得容易了。

WAS 7 提供了称为基于属性的配置的重要管理功能，可以简化将管理自动化的体验。使用此功能，管理员可以将

WebSphere 配置提取到属性文件中，属性文件以简单的 <property=value> 格式列出配置参数。所提取的配置的简单格式使得读取和操作它变得非常容易。还可以将属性文件反馈到 WebSphere 管理中，以便将其内容应用于现有的配置。

基于属性的配置功能是通过一组可以通过 wsadmin 脚本执行的管理命令来公开的。基于属性的配的主要功能包括：

能够创建或删除配置构件，以及读取或操作现有的配置。

预览模式，藉此生成报告，指示如何将属性文件中的数据应用于 WebSphere 配置而不实际更改配置。

变量支持，允许将属性文件中的值定义为可在应用属性文件时进行解析的变量，从而促进给定属性文件跨多个配置的重用。■

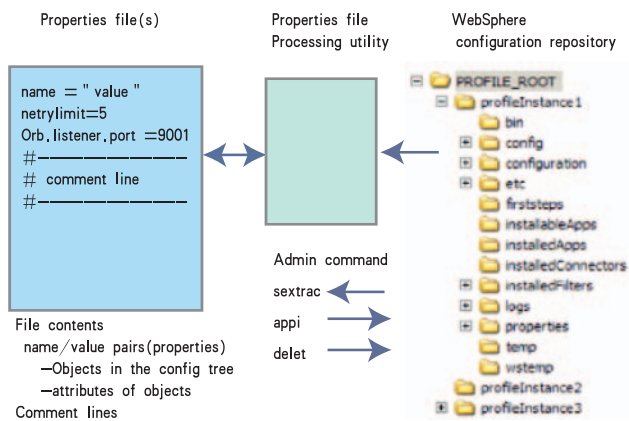


图 4. 基于属性文件的配置

专家专栏

WebSphere MQ For AIX 和 HP-UX 安装配置说明

中联集团系统软件技术部总监 甘 荃 / 文



概述

WebSphere MQ 支持的 UNIX 平台有 AIX、Linux、Solaris 和 HP-UX。由于篇幅有限，本文主要对 AIX 和 HP-UX 平台的安装步骤进行描述。另外 HP-UX 平台有 PA-RISC 平台和 IA64 平台，目前 IA64 平台比较流行，所以本篇文章仅以 HP-UX IA64 平台上安装 Websphere MQ 6.0 为例，结合个人项目经验，详细描述产品安装和配置的全过程，并针对在 HP-UX 11.31 平台上的安装注意事项进行了重点说明。

安装 WebSphere MQ 之前，建议做如下准备：

- 检查硬件和软件需求
- 创建 WebSphere MQ 文件系统
- 创建 mqm 组 and 用户

在安装过程中分为安装准备，调整核心参数(针对 HP-UX)，安装软件，安装补丁，版本验证。

软件安装完成后进行配置 MQ，然后

根据项目需要继续完成高可用性模式配置。

一、安装环境

在安装前要检查机器的硬件配置及操作系统是否达到相关要求，否则可能引起安装失败。WebSphere MQ 对于机器的环境要求并不高，普通的 PC 机都可以安装。如果有网络通信，则需要有相应的网卡并进行通信配置，比如 IP 地址。对于单机环境，可以用自环方式配置通信，

比如配置虚拟 Loopback 网卡。

针对操作系统，建议登陆 IBM 的官方网站确认最新的需求信息，如下链接：

<http://www-01.ibm.com/software/integration/wmq/requirements/index.html>

在一个项目中笔者在安装 MQ v6.0 for HP 11.31 的时候只参考安装手册，所以安装无法继续，后来通过查看官方网站的最新系统需求和注意事项，问题得以解

决,所以建议详细阅读有关的Note。例如针对 MQ v6.0 for HP 11.31 的安装系统需求如下 URL。

http://www-01.ibm.com/support/docview.wss?rs=171&uid=swg27006792
最近一次在安装 MQ v6.0 for Linux Suse v10的遇到了安装成功,但后来无法创建队列管理器问题,后来通过查看相关链接,安装了 MQ v6.0 for Linux Suse v10。

1. 硬件需求

● IBM:

WebSphere MQ for AIX,Version 6.0 可以运行在任何支持 AIX5L V5.2 或者 AIX 5L V5.3 的操作系统并能运行 64 bit 程序的 IBM 或其他厂商的服务器。

● HP:

WebSphere MQ for HP-UX, Version 6.0 可以运行在任何 Intel IA64(IPF) V2 以上版本的机器。

2. 操作系统

支持 HP-UX 版本:

- HP-UX 11i V2 (11.23) for IPF
 - HP-UX 11i V3 for IPF
- 支持 AIX 版本:
- AIX5L V5.2 (Plus ML3)
 - AIX V5.3

3. 通讯协议

- TCPIP
- SNA LU6.2

4. 软件安装空间需求

在 HP-UX 上软件安装在 /opt 目录下,需要的空间如下:

软件组件	空间大小(MB)
WebSphere MQ Server Installation	325
IBM Global Security Kit V7(64-bit)	14

在 AIX 上软件安装在 /usr 目录下,需要的空间如下:

软件组件	空间大小(MB)
WebSphere MQ Server Installation	325

二、WebSphere MQ 安装步骤

1. 准备工作

1.1 创建组和用户

WebSphere MQ 产品安装时,需要创建一个mqm组和一个mqm用户,要求mqm用户的主组是mqm。为了方便系统管理,建议创建mqm组和mqm用户时,自定义组ID和用户ID。如果是在高可用环境下,两台服务器创建的mqm组和mqm用户的ID必须要一样。

●注意:建议将root用户增加到mqm组中。

组名	组ID	用户
mqm	405	root,mqm

用户名	组ID	主组	组集	主目录	管理员用户
mqm	406	mqm	mqm	/home/mqm	false

1.1.1 AIX 平台

建议使用如下命令创建mqm组和mqm用户。

```
mkgroup id=405 mqm
mkuser id=406 pgrp=mqm
groups=mqm home=/home/mqm mqm
```

1.1.2 HP-UX 平台

建议使用如下命令创建mqm组和mqm用户。

```
> 创建mqm组
#groupadd -g 405 mqm
或者用SAM工具。
> 创建mqm用户
#useradd -u 406 -g mqm -d /home/mqm -m mqm
或者用SAM工具,增加mqm到Group SET中。
```

1.2 创建文件系统

如果MQ环境是单机环境则在本地磁盘创建如下文件系统即可;如果需要做成

高可用环境,那么需要在共享磁盘上创建一个共享vg,例如vgmq,本文以创建一个vgmq为例。

1.2.1 AIX 平台

WebSphere MQ产品的安装目录是/usr/mqm,工作目录是/var/mqm,安装目录和工作目录不允许修改。GSKit必须安装在缺省位置。

在AIX平台的vgmq创建如下lv和文件系统:

卷组	设备名	大小	文件系统	属主
vgmq	/dev/mqmlv	1024M	/var/mqm	mqm:mqm
vgmq	/dev/mqmloglv	1024M	/var/mqm/log	mqm:mqm

```
mklv -y mqmlv -t jfs2 vgmq 1G
crfs -v jfs2 -d /dev/mqmlv -m /var/mqm -A yes
mount /var/mqm
```

```
mklv -y mqmloglv -t jfs2 vgmq 1G
crfs -v jfs2 -d /dev/mqmloglv -m /var/mqm/log -A yes
mount /var/mqm/log
```

1.2.2 HP-UX 平台

WebSphere MQ产品的安装目录是/opt/mqm,工作目录是/var/mqm,安装目录和工作目录不允许修改。GSKit必须安装在缺省位置。

在HP-UX平台的vgmq创建如下lv和文件系统:

卷组	设备名	大小	文件系统	属主
vgmq	/dev/vgmq/lvmqm	1024M	/var/mqm	mqm:mqm
vgmq	/dev/vgmq/lvmqmlg	1024M	/var/mqm/log	mqm:mqm

命令参考如下:

```
lvcreate -L 1024 -n lvmqm /dev/vgmq
/sbin/mkdir -p -m u=rwx,g=rx,o=rx /var/mqm
/usr/sbin/mkfs -F vxfs /dev/vg00/
```

```
rlvmqm
/usr/sbin/mount -F vxfs -e /dev/vg00/
lvqm /var/mqm

lvcreate -L 1024 -n lvqm /dev/vgmq
/sbin/mkdir -p -m u=rwx,g=rx,o=rx /var/
mqm/log
/usr/sbin/mkfs -F vxfs /dev/vg00/
rlvmqmlog
/usr/sbin/mount -F vxfs -e /dev/vg00/
lvqmlog /var/mqm/log
```

2. 安装 WebSphere MQ for AIX

2.1 安装准备

使用如下命令：

```
#df -k
```

确认下列文件系统已经被 mount 上：

```
/var/mqm 和 /var/mqm/log
```

2.2 安装软件

把 WebSphere MQ 6.0 for AIX 的介质 ftp 到服务器上的一个临时目录，例如 /tmp/mqm，然后开始安装。如果是 tar.Z 文件，先用 uncompress 解压缩 Z 文件，然后用 tar xvf 命令解压缩 tar 文件，使用 root 用户执行安装。

```
# smitty install_latest
```

单击列出以显示软件的输入设备或目录，选择包含安装映像的位置。

使用要安装的软件字段以获取可用文件集的列表，然后选择要安装的文件集。

```
mqm.base
mqm.client
mqm.server
mqm.man.en_US.data
mqm.java
```

修改 ACCEPT new license agree-ments Yes。

2.3 安装补丁

首先将安装包 6.0.1-WS-MQ-AixPPC64-FP0001.tar 解包，然后安装

```
#tar -xvf 6.0.1-WS-MQ-AixPPC64-FP0001.tar
```

```
#cd 6.0.1-WS-MQ-AixPPC64-FP0001
```

```
#smitty update_all
```

3. 安装 WebSphere MQ for HP-UX

3.1 安装准备

使用如下命令：

```
#df -k
```

确认下列文件系统已经被 mount 上：

```
/var/mqm 和 /var/mqm/log
```

3.2 调整核心参数

WebSphere MQ 使用了 semaphores 和 Shared memory，缺省 kernel 配置是不足够的，在安装之前需要根据服务器的配置情况增大相应的参数值，下表中为 WebSphere MQ 推荐的最小配置值。

shmmax	536870912
shmseg	1024
shmmni	1024
semaem	16384
semvmx	32767
semmns	16384
semmni	1024 (semmni < semmns)
semmnu	16384
semume	256
max_thread_proc	66
maxfiles	10000
maxfiles_lim	10000
nfile	10000

注意：核心参数修改完毕后，需要重新启动服务器。

3.3 安装软件

把 WebSphere MQ 6.0 for HP 的介质 ftp 到服务器上的一个临时目录，例如 /tmp/mqm，然后使用 root 用户开始安装。

- 首先执行 /tmp/mqm/mqlicense.sh，接受 license 许可，或者执行 /tmp/mqm/mqlicense.sh -accept。

- 如果是 HP-UX11.23 平台，则可以使用如下方法安装：

```
#swinstall -s /tmp/mqm/p600-100-051021.v11
```

使用空格键选中需要安装的软件，然后在 Action 菜单中选择 Mark for install 到上一层菜单中，在 Action 菜单中选择 Install(analysis)，选 Ok 开始安装。

- 如果是在 Itanium 的 HP-UX11.31 平台，需要使用 '-x allow_incompatible=true' 参数：

```
#swinstall -s /tmp/mqm/p600-100-051021.v11 -x allow_incompatible=true
```

如果安装 WebSphere MQ 组件，选择 MQSERIES，然后把光标从 Actions 移到 Install。在安装日志中将看到关于 MQ 是不兼容的警告信息，请忽略然后选择 OK 进行安装。

安装完成后，你会发现不能使用 MQ 命令，这时需要运行如下命令：

```
#swconfig -x allow_incompatible=true
MQSERIES
```

3.4 安装补丁

```
#cd /usr/hpinstall/patch_mq
```

```
#tar xvf 6.0.2-WS-MQ-HpuxIA64-FP0005.tar.Z
```

使用 swinstall 进行补丁安装。

注意：

如果是 HP 11.31 平台则需要如下方法进行安装：

使用 swinstall 安装 MQ6.0.1 GA

然后运行如下命令：

```
#swconfig -x allow_incompatible=true
MQSERIES
```

使用 swinstall 升级到 MQ 6.0.1.1。

使用 swinstall 升级到 MQ6.0.2.3。

3.4 验证 WebSphere MQ 版本

使用如下命令验证 WebSphere MQ 产品版本，使用 mqm 用户登录，然后执行 dspmqver 命令：

```
#su - mqm
$dspmqver
```

三、WebSphere MQ 配置步骤

在 WebSphere MQ 的配置过程中，我们使用控制命令(control command)来创建、启动、停止和删除队列管理器。而创建 MQ 的侦听程序、队列、通道和进程等对象则需要使用 MQSC 命令。

建议参考《System Administration Guide》和《Script (MQSC) Command Reference》文档。

1. 创建队列管理器

使用 mqm 用户登录执行命令：
crtmqm -lc -lf 16384 lp 2 ls 1 -q QMTEST

其中：lf 表示日志文件大小,单位: 页 (4K)；

lp 表示主日志的数目；

ls 表示从日志的数目；

q 表示缺省队列管理器；

QMTEST 表示队列管理器名。

2. 配置 MQ 监听端口

在文件 /etc/services 增加如下行(确保 1414/tcp 尚未被其它应用所占用)。

```
MQSeries 1414/tcp #for MQ Listener
使用 MQSC 命令定义侦听程序，例
```

如：

```
def listener(ltcp) trptype(tcp) control
(QMGR) port(1415) replace
```

注：在 AIX 平台还可以在 /etc/inetd.conf 文件中增加如下行来创建 MQ 的侦听程序。在 WebSphere MQ6.0 以后版本，我们推荐采用 MQSC 命令(DEF LISTENER)来创建侦听程序。

```
MQSeries stream tcp nowait root /opt/
mqm/bin/amqcrsta amqcrsta -m
QMTEST
```

3. MQ 的启动与关闭

停止队列管理器：

```
endmqm QMTEST
```

启动队列管理器：

```
strmqm QMTEST
```

显示队列管理器状态：

```
dspmq 显示现有的各个 qm 的状态
```

4. 定义 MQ 对象

创建队列管理器后，需要在队列管理器里面创建队列通道等对象,我们可以使用 MQSC 命令创建 MQ 对象，如下列表所示：

● 本地死信队列

```
def ql(DEADQ) defpsist(yes) maxdepth
(20000) replace
```

● 定义接收队列

```
def ql(XX) defpsist(yes) maxdepth
(20000) replace
```

● 设置队列管理器属性

```
alter qmgr deadq(deadq)
```

```
alter qmgr ccsid(819)
```

● 定义传输队列

```
def ql(TO.QMB) usage(XMITQ) defpsist
(YES) maxdepth(500000) trigger trigtype
(FIRST) trigdata(QMA.QMB) initq
(SYSTEM.CHANNEL.INITQ) replace
```

● 定义远程队列

```
def qr(QRB) rname(QLB) rqmname
(QMB) xmitq(TO.QMB) defpsist(yes) re-
place
```

● 定义发送通道

```
def chl(QMA.QMB) chltype(SDR)
discint(1800) conname('192.168.2.1
(1414)') xmitq(TO.QMB) trptype(tcp)
shorttrty(10000) shorttmr(30) longtmr
(300) longrty(99999999) batchsz(50)
replace
```

● 接收通道

```
def chl(QMB.QMA) chltype(rcvr) trptype(tcp)
使用 MQSC 命令编写队列管理管理器
器中所需的对象，然后保存文件，例如
crtobj.tst。
```

然后使用如下命令创建对象，确保队列管理器已经启动，并用 mqm 用户登录执行：

```
$runmqsc QMTEST <crtobj.tst >
crtobj.log
```

执行完毕，请检查 crtobj.log 日志文件，确保没有任何错误信息。

推荐参看上期《WebSphere 中国》的“WebSphere MQ 消息通道的配置和维护介绍”中的“附录：MQ 对象配置参考”。

五、配置高可用性模式

如果需要把队列管理器配置成高可用性模式，有如下注意事项：

1) 在共享卷组(vgmq)中创建 /var/mqm 和 /var/mqm/log 文件系统，

2) 在两台服务器上都需要创建 mqm 组和 mqm 用户，并且在两台服务器的组 ID 和用户 ID 都需要相同。

3) 在两台服务器上都需要安装 WebSphere MQ 软件。■

Ajax: 超越桌面的 Web 用户体验

IBM 中国开发中心 毛新生、李文兵 / 文

随着 Web 时代的到来, Web 2.0 越来越多地成为人们关注的焦点。在 Web 2.0 的诸多特征中, 一个很重要的元素是对于用户体验的改善。用户体验是指用户在使用产品或者服务过程中建立起来的心里感受。从技术角度来讲, 用户体验通常由可用性、信息架构以及用户界面组成。用户体验对于用户以及产品开发都非常重要, 它不仅能够提高用户的使用效率, 吸引更多用户群, 而且可以降低软件的开发和维护成本。

相对于 Web 2.0 式的 Web 应用, 传统的 Web 应用遵循一种请求/响应模式, 每次请求都会重新加载整个页面, 这样导致了用户在交互过程中经常处于等待状态, 严重影响了用户体验。此外, 传统的 Web 应用很少考虑用户使用过程中的交互性, 因此用户难以得到桌面应用式的用户体验。为了解决这样的问题, RIA(Rich Internet Application), 即丰富互联网应用程序便应运而生, 其最大的优点在于将桌面应用程序的用户交互体验与传统 Web 应用的部署灵活性结合起来。在 RIA 的各种相关技术中, Ajax (Asynchronous JavaScript and XML) 是一种被广泛使用的技术。Ajax 这个名词的首次出现是在 2005 年 2 月 18 日由 Jesse James Garrett 发表的一篇题为《Ajax: A New Approach to Web Applications》的文章中提出的。事实上, Ajax 并没有创造出某种具体的新技术, 它所使用的所有技术都是在很多年前就已经存在了, 然而 Ajax 提出以一种新方式来驾驭所有旧的技术, 使得古老的 Web 应用焕发了新的活力。下面列举了 Ajax 中所涉及的基本技术:

- * 超文本标记语言 (Hypertext Markup Language, HTML): 用以展现网页的内容和结构。

- * 层叠样式表 (Cascading Style Sheets, CSS): 用以展现网页内容的样式 (或装饰)。

- * 文档对象模型 (Document Object Model, DOM): 一种提供给网页脚本语言 (例如, JavaScript) 动态修改网页内容与结构的 API。

- * XMLHttpRequest 对象: 为浏览器与服务器之间的交互提供便利。浏览器端通过 JavaScript 脚本调用 XMLHttpRequest 对象向服务器端发出请求。

- * JavaScript: 在浏览器中执行的一种编程语言。它将所有的 Ajax 技术黏合在一起。JavaScript 可以侦听浏览器中发

生的事件 (例如单击一个按钮), 并使用 XMLHttpRequest 请求服务器以对事件作出反应, 然后根据返回的结果修改 DOM 树。

以上技术除了 XMLHttpRequest 以外, 所有的技术都是目前已经广泛使用的 Web 标准技术。而 XMLHttpRequest 虽然尚未被 W3C 采纳, 事实上已经成为一个约定俗成的标准。几乎所有主流的浏览器都支持这项技术。它也是 Ajax 概念的核心价值所在, 即提供客户端对服务器端的异步请求。Ajax 综合所有成熟的技术, 并提出自己独特的概念, 极大地改善了传统用户体验的缺陷, 因此 Ajax 是目前 Web 开发中最符合标准的技术。上述所有技术都已经可以在浏览器中使用, 因此用户仅需要一个浏览器而不需要安装任何额外的软件, 这对于 Ajax 技术的普及、降低部署维护的成本是非常重要的。

Ajax 的核心工作原理通俗来讲, 相当于在客户端和服务器之间加了一个中间层, 使用户的操作与服务器响应异步化。用户提交请求或者数据时, 通过 Ajax 发送服务器进行后台处理, 这期间客户端可以进行其他的操作。当服务器将数据返回时, Ajax 再决定如何处理这些数据。下图展示了 Ajax 与传统 Web 的交互模型比较示意图。



下面举一个简单的例子来演示如何具体使用 Ajax。这个例子中, 我们希望用户输入城市名称时, Ajax 会在后台获取该城市的邮编, 帮用户自动填入到表单中。下面是启动 Ajax 的 HTML 片段:

```
<form>
  <p>City: <input type="text" name="city" id="city" size="25"
onChange="callServer();" /></p>
  <p>Zip Code: <input type="text" name="zipCode"
id="zipCode" size="5" /></p>
</form>
```

这样在用户输入一个新的城市名称时, callServer() 方法就会被触发, 进而使用 Ajax 来调用服务。下面来看如何来使用 Ajax:


```
function callServer() {
    //创建一个XMLHttpRequest对象。为了例子的简单性，这里没有考虑浏览器之间的兼容性问题。您可以在 developerWorks 上获取更加完整的示例。
    var xmlhttp = new XMLHttpRequest();
    // 从 HTML 中获取城市的名称
    var city = document.getElementById("city").value;
    // 构建需要访问的服务的 URL
    var url = "/scripts/getZipCode.php?city=" + escape(city);
    // 开启一个连接服务的连接
    xmlhttp.open("GET", url, true);
    // 设置回调函数，这样当结果返回时，使用该函数对结果进行处理
```

```
    xmlhttp.onreadystatechange = updatePage;
    // 发送 Ajax 请求
    xmlhttp.send(null);
    接下来需要提供函数来处理服务器的响应结果。从上面的例子中可以看出，只需要提供一个回调函数就可以了。
```

```
function updatePage() {
    if (xmlhttp.readyState == 4) {
        var response = xmlhttp.responseText;
        document.getElementById("zipCode").value = response;
    }
}
```

是否感觉它们是很普通的代码，那就对了。Ajax 程序就是这么简单，只需要了解 JavaScript 以及相关的 Web 标准，您现在就可以编写 Ajax 的程序了。

虽然 Ajax 中的数据交换格式指的是 XML，但是它并没有限制数据交换的格式。而现在开发者经常为 XML 的解析伤透脑筋，无论是服务器端生成或处理 XML，还是客户端用 JavaScript 解析 XML，都常常需要编写复杂的代码，极低的开发效率。JSON (JavaScript Object Notation) 为 Web 应用开发者提供了另一种数据交换格式。它是一种轻量级的数据交换格式，非常适合于服务器与 JavaScript 的交互。通过 JSON，Ajax 可以更加方便快捷地处理各种复杂的数据结构。

Ajax 不仅满足了市场上对于丰富表现力与交互性的 Web 客户端的需求，并且无需安装。然而 Ajax 并不是在该领域中唯一的技术，相关的技术还有 Adobe Flex，JavaFX 以及 Microsoft Silverlight 等等。但是这些技术通常需要在浏览器上安装相关插

件，对于没有安装相关插件的浏览器会导致 Web 应用不可用。当然在某些情况下，Ajax 也不一定是最合适的选择，需要根据具体情况进行选择。

Ajax 正在给企业 Web 应用带来深刻的影响。Ajax 通过改善 Web 应用的用户体验从而提升了客户的满意度，进而增加了产品或服务的销售和利润。Web 应用的架构风格也因为 Ajax 而进行着改变，Ajax 将 Web 应用从传统的服务器端进行服务访问的形式转变成客户端来直接对服务进行访问。从而客户端将承担了更多的职责，一些数据和逻辑的处理在客户端完成，而浏览器成为了基于开放标准和编程模型的应用平台。

然而，在使用 Ajax 进行 Web 应用开发中，同样也面临着很多的挑战。例如浏览器之间的兼容性问题、JavaScript 缺乏组件和包模型、HTML 标准控件也比较少等等，于是大量的 Ajax 框架涌现出来，以帮助人们更加容易地编写基于 Ajax 的 Web 应用。其中经常被使用的包括 Dojo、jQuery、Prototype、Ext、YUI 等等，它们提供了丰富的工具包来简化 Ajax 编程、对 DOM 以及 CSS 操作、以及大量交互性良好的控件和各种特效的支持等。通过它们可以提高企业 Web 应用的开发效率。

针对 Ajax，IBM 提供了一系列的 Ajax 开发工具包。Dojo 是社区主流的 Ajax 框架 (<http://www.dojotoolkit.org>)，它是开放源代码的，并且基于开放的 Web 标准，例如 HTML、JavaScript 和 CSS。它很好地屏蔽了浏览器之间的差异，并且提供了模块化的包管理机制以及灵活的 widget 支持。IBM 不仅通过对 Dojo 的支持来提供 Ajax 支持，同时在 Dojo 的基础上扩展了一些 IBM 客户端 JavaScript 库来简化和不同协议、数据格式之间的操作。例如 SOAP JavaScript 库、Atom JavaScript 库、Gauge 小部件以及 OpenSearch JavaScript 库等。不仅如此，IBM 还提供了 Ajax proxy、JSON4J、Ajax messaging 等帮助用户更加快速地构建 Web 2.0 式的 Web 应用，从而降低开发成本以及加快产品的开发速度。

IBM 大量的 Web 2.0 产品也广泛地采用了 Ajax 技术。WebSphere sMash 是一个轻量级的、敏捷的 Web 开发平台，它同样提供了很好的对 Ajax 的支持。IBM Mashup Center 同样是使用 Ajax 技术来构建的 Mashup 平台，用以帮助企业用户快速构建定制化的 Web 应用。WebSphere Business Space 也通过使用 Ajax 技术实现了一个基于浏览器的界面，用户可对来自 WebSphere Business Process Management 产品的内容进行定制。

现在 Ajax 已经在 Web 应用中随处可见，并且它还将继续影响着下一代 Web 应用。让我们继续关注它、拥抱它吧！■



手把手进行产品实验室， 亲手体验搭建 SOA 的激动过程

如果您仍然觉得 SOA 的概念大于实际，理论与应用脱节？那么，通过 SOA 产品实验室体验活动，您的这种想法马上就会烟消云散。SOA 产品实验室体验活动是由 IBM 资深工程师精心准备的实验和课件，将带您在计算机上一步步搭建起模拟的业务应用场景下的产品实施架构，让您触摸真实的面向服务架构！

那么，SOA 产品实验室体验活动究竟是怎么一个活动，能够吸引如此多的人来参与？在体验的过程中，我们又将接触到那些 WebSphere 产品呢？此外，通过参加 SOA 产品实验室体验活动，我们又将获得哪些知识，有怎样的感受呢？

用户谈参加 SOA 产品实验室体验活动的感受

中国电子科技集团第十五研究所 陈华雄

非常荣幸能够参加由 IBM 公司举办的一年一度的 IBM Impact SOA 高峰论坛。其实，参加本次 IBM Impact SOA 高峰论坛，主要是奔着下午的 SOA 产品实验室体验活动去的。因为，在未参加之前，就看了很多 SOA 产品实验室体验活动的介绍，对其中的实验主题都非常感兴趣。于是，就欣然报名前往了。

相对其它分会场，SOA 产品实验室的教室显得相当安静。平时我们工作使用的都是 MB，想比较一下与 Data Power 的区别，故选择参加了“快速安全的 SOA 整合关键就在 Data Power”实验。每个实验室只有二十个座位，但每个实验室活动报名的人却超过四十人，所以自己早早去占位置。实验室整整齐齐地放了二十台机器，每个座位前面都有一本本实验的操作教材，教材主要是课程的详细操作步骤和操作说明，内容从浅入深说明 Data Power 的使用。内容从操作界面到简单使用，还有和 SOA 产品联合使用的操作。每个参加实验室活动的人可以按照教材上面写步骤来进行 step by step 的操作，实现每个课时的目标。“快速安全的 SOA 整合关键就在 Data Power”实验的老师来自台湾，经验丰富。他讲述 Data Power

的特点、功能、组成和课程的目标等，讲述过程中插入很多个人经历、体会和应用例子，使技术的内容听起来并不枯燥。

然后是上机操作时间，按照教材上写的步骤操作，有任何问题都能及时找老师解决。虽然教材内容写的步骤很详细，但是操作起来还是经常有这样或那样的问题。学员们遇到问题，老师都会耐心解答，即使是相同的问题。老师除了解答操作过程的问题，也会跟大家谈谈其它的话题，例如，使用经验，Data Power 跟其它的产品的比较等。

总的来说，参加 SOA 产品实验室体验活动在各个方面我都非常满意，因为我从中学到的不只是产品的使用，更重要的能了解从事 Data Power 工作的思维及解决问题的方法，技术专家手把手的传授使自己获益良多。此外，本次 SOA 产品实验室的体验印象非常深刻，跟技术专家面对面、点对点的交流的机会在日常工作中非常少，自己收获颇丰，希望能有更多这样的机会。

北京数字证书认证中心有限公司 张雪彬

一直对 SOA 很感兴趣，在 WebSphere 开发与应用网站看到有关 SOA 产品实验室体验活动的介绍，甚是欣慰，因为终于有机会实实在在地与 SOA

SOA 产品实验室体验活动简要介绍

实验类别	实验名称	实验简介	覆盖产品
动态业务流程管理	动态的账户验证 业务流程管理	本实验会通过从业务建模到业务实现到业务监控和动态配置等一个完整的 SOA 业务流程生命周期场景来实现一个银行业中常见的用户验证流程。 通过完成本实验，用户可以体验和了解 IBM SOA 和 WebSphere BPM 完整的产品线和解决方案开发生命周期。	WDPE; Modeler + WID + WPS + Monitor + WSRR
	通过 ILog 实现业 务流程中的动态 策略管理	本实验会通过实现一个保险业常见的报价查询场景来实现如何通过 ILog 业 务规则管理系统进行灵活动态的业务流程配置。 通过完成本实验，用户可以体验和了解通过 IBMLog 业务规则管理系统增强 业务流程的灵活性和可管理性。	WPS + Ilog(JRule)
应用整合 和连接	Be Smart – 智慧的销售管理	本实验会通过一个典型的端到端的应用系统集成解决方案，以企业服务总线 (ESB)为基础，完成对销售管理、人力资源及财务系统的整合，并实现了对整 个销售行为的监控和管理。 通过完成本实验，用户可以体验和了解 IBM 系统连接相关产品强大的功能和 易用性。	WMB + MQ + Adapter + Monitor
	快速安全的 SOA 整合关键 就在 Data Power	本实验会通过实际操作来让用户灵活、动态的配置系统安全策略，通过对硬 件的配置实现端到端的加密 / 解密过程。 通过完成本实验，用户可以体验和了解到 IBM DataPower 设备强大的安全保护功 能和灵活的配置功能。	DataPower
电子商务	通过 WebSphere Commerce 实现多渠道销售	本实验通过将网上销售、电话销售、移动电子商务销售渠道的整合，实现一个 典型的零售业销售场景。 通过完成本实验，用户会亲身体验 WebSphere ommerce 强有力的多渠道 C 销 售整合的解决方案，同时也将感受到更加新颖的、用户体验更强的 Web2.0 网上零售商店。	Commerce
	零售行业中的 内容管理中心	本实验会通过使用 WebSphere Commerce 内容管理工具进行商品管理和市场营 销管理的场景来实现电子商务站点商品编辑，市场营销管理的流程。 通过完成本实验，用户可以体验到 WebSphere Commerce 内容管理工具强大的业务管 理功能和 Web2.0 方便、快捷的界面效果。	Commerce

产品进行近距离的接触。活动当天，自己首先了解了大会的整体情况，之后就直奔主题为“快速安全的 SOA 整合关键就在 DataPower”的产品实验室教室。先说一下大会的组织，非常的专业，也很人性化，可以说是每个细节都很到位。

很感谢组织者，能亲自参与关于 DataPower 的实践，来自台湾的老师给我们上了一堂实践课，老师的水平非常高，我在短短的时间内学到了很多的东西。通过“快速安全的 SOA 整合关键就在 DataPower”实验，了解到我们只需对 DataPower 做设定而不是需要编程，通过 WEBGUI，就可以简单、迅速的设定物件进而完成部署任务。通过 DataPower 提供的丰富的使用者界面，用户能够十分容易地管理由资源控制创造出的不同服务领域。

总之，DataPower 对于 SOA 的支持非常好，功能非常强大，非常感谢组织者给予的自己这次学习的机会。

清华大学 Web 与软件技术研究中心 赵洁

2009 年 6 月 10 日，我与另一位研究中心的同事一起参加了 IBM

Smart SOA 高峰论坛会议，能参加这样的盛会我感到非常的荣幸，此次盛会带给我们视觉、听觉、和知识领域的全新体验。

由于我们现在的研究项目中涉及到业务规则的抽取和业务流程的管理，于是报名参加了“动态的账户验证业务流程管理”和“通过 Ilog 实现业务流程中的动态策略管理”两个实验。每个实验 1.5 个小时，实验过程中，IBM 工作人员送教材、讲解、演示，耐心的回答我们提出的每个问题，有时他们也会提出小问题，一步步引导我们深入理解实验的基本原理和软件的使用方法。通过亲自动手改动业务流程，更新规则内容，我很轻松的理解了很多业务流程管理和业务规则管理的知识。同时，体验到了 IBM Ilog 和 WebSphere 产品功能的强大和易用。一个下午很快过去了，时间虽然短暂，实验内容非常丰富和精彩。

真的很感激 IBM 公司为客户提供的这样一个亲身体验的实验机会，让我受益匪浅。我相信参会的很多人都会记住这次的盛会和实验的丰富内容。在此深深感谢帮助我的 IBM 工作人员，会议和实验的成功是他们辛勤工作的见证。希望将来能够有更多的机会与他们一起学习交流。◆



白话解说 Spring 容器设计理念

李俊杰 / 文

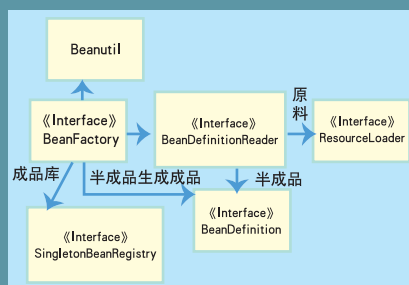
Spring是为了解决企业应用程序开发复杂性而创建的开源框架，书店上关于Spring的书籍汗牛充栋，网上相关的文章连篇累牍，其中有很多写的很不错的，有入门例子的，有问题解决方案的，有环境设置的，有源代码分析的，有spring与其他开源系统集成的，不一而足。本文通过生活白话，不拘泥于Spring源代码和专业术语的束缚，不拘泥于具体的实现细节，类比介绍Spring容器的宏观的设计理念。

让 Spring 容器走下神坛

Spring 容器提供 Spring 框架的基本功能，是工厂模式的实现。换句话说，Spring 就是Bean的工厂，管理Bean的生命周期。那如何来设计呢？Spring源代码虽不说是浩如烟海，也让人头晕目眩，如右图所示，我们让Spring容器走下神坛，既然是面向接口编程，用接口来描述框架，不必拘泥于具体实现可能更清晰些。

任何的设计都是这样，有原料，有入口，有加工，也有出口。既然Spring是Bean的容器，是Bean的工厂，那么生产Bean流程是：

- 1) 原料：bean 的配置文件；
- 2) 进料：通过 ResourceLoader 把 xml 文本文件读入；
- 3) 初加工：通过 BeanDefinitionReader 把原料加工成半成品 BeanDefinition；
- 4) 精加工：Bean 的生产车间 BeanFactory 把半成品 BeanDefinition 加工成 Bean；
- 5) 入库：成品库 SingletonRegistry 保存成品 Bean，及 Bean 的标签 (beanId)！你也知道这就是 Map 的工作了。



IOC 是 Spring 核心？关键？

Spring 容器使用控制反转 (IOC) 模式将应用程序的配置和依赖性规范与实际的应用程序代码分开，控制反转 (IOC) 模式是基于 java 反射的基本原理。

通常的软件设计，肯定把 IOC 作为容器的核心，其他外围的功能如 bean 的 Scope 都是基于 IOC 展开，而 Spring 设计却只把它放如工具类 BeanUtil，这正是 Spring 的高明之处，虽然 IOC 是 Spring 的创建 bean 的基础，但创建一个 bean 需要判断是否静态工厂创建，是否有含参数的构造方法，而这些参数也可能是其他对象的实例，创建 bean 的初始化方法等等，这些不仅仅是 bean 的 IOC 所能涵盖的，并且在创建一个 bean 的过程中

可能需要多次迭代调用 IOC 程序。

综上所述，IOC 虽然是 Spring 的创建 Bean 的基本原理，但仅是及其重要的一部分，并不是 Spring 容器的核心，是 Spring 创建 bean 的比不可少的工具。IOC 是 Spring 容器的关键，而不是核心。

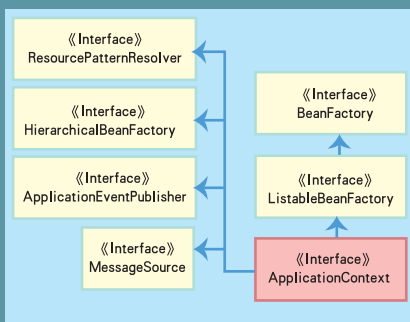
Interface 分级设计

接口的分级设计，是针对不同的客户给予不同级别的产品。不同等级的接口关注点也各有不同，就像产品也分为多个级别，如高级产品、中级产品、初级产品分别对应于不同的客户群，或者如同银行的客户渠道有银行卡，柜台，网银，手机终端，ATM 机等等。同样接口也分为内部接口和外部接口，就如同有些接口可能永远不被外部用户所知，仅仅是内部使用，就像工厂内部多个车间之间的内部产品。

org.springframework.beans.factory.BeanFactory 是 Spring 的顶级 Interface，其中只有 getBean 系列，containsBean，isPrototype，isSingleton，getType 等成品 Bean 的基本方法。

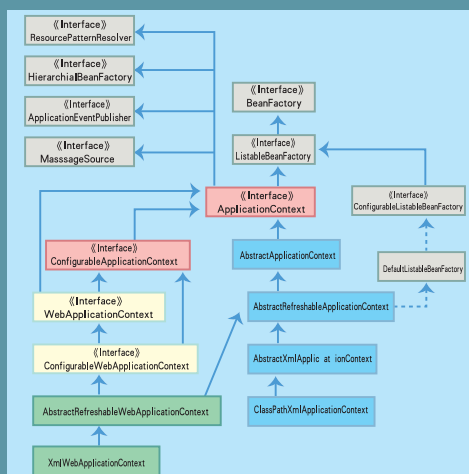
org.springframework.beans.factory.ListableBeanFactory 是 Factory 的管理方法如 containsBeanDefinition，getBeanDefinitionCount，getBeanDefinitionNames，getBeanNamesForType 系列，getBeansOfType 系列，ListableBeanFactory 是针对 BeanDefinition 半成品的。

而 ApplicationContext 接口除了集成 ListableBeanFactory 接口外，还继承了 ResourcePatternResolver, HierarchicalBeanFactory, ApplicationEventPublisher, MessageSource 的功能，如在资源处理（国际化处理），事件传递，及各应用层之间的 context 实现。同样是对外的接口，Spring 更建议采用 ApplicationContext 的原因也就在这儿，ApplicationContext 更像是针对VIP客户的产品。



Interface 扩展设计

Spring设计是开放性的设计思路,是值得我们做架构设计人员学习的,如右图所示:



总结如下:

- 1) 不同的包下面放不同的东西,上层包放接口,下层包放实现的抽象类及实体类。如context.support包下面放实现context包下面的接口的抽象类及实现类。
- 2) 为适应 WEB 级别的 Application-Context,创建了WebApplicationContext的接口,该接口继承了 ApplicationContext,又

添加了 WEB 层的个性化方法如 getServlet-Context 及属性。现在终于明白了在接口间的“extends”的含义,绝对是对上层接口的“扩展”,扩展的目的是适应更具体化的环境。就如同因为国家大,情况复杂,中央的政策相对都是比较抽象的,而地方政策必须结合地方的特色,既要满足中央政策的要求,即实现中央政策的接口,又要根据本地特殊情况,扩展个性化的政策,即地方政策接口,从而更具有可操作性。

3) 在 org.springframework.web.context.supportAbstractRefreshable

Web ApplicationContext 类是既实现了 ConfigurableWebApplicationContext 接口,即实现 WEB 的个性化特色,又继承了 org.springframework.context.support.AbstractRefreshableApplicationContext类。借助了已有的类,实现了架构上的复用,这就像制定政策既要满足地方上的特色,又可以参照其他地区现成的政策经验。在实际工作中,也会常常有这样的设计。 ■

com.springframework.beans.factory (淡黄色背景)	BeanFactory ListableBeanFactory ConfigurableListableBeanFactory DefaultListableBeanFactory	
com.springframework.context (棕色背景)	ApplicationContext ConfigurableApplicationContext	
org.springframework.context (黄色背景)	.support	AbstractApplicationContext AbstractRefreshable ApplicationContext AbstractXmlApplicationContext ClassPathXmlApplicationContext
org.springframework.web.context (绿色背景)	WebApplicationContext ConfigurableWeb ApplicationContext	
org.springframework.web.context (浅蓝色背景)	.support	AbstractRefreshableWeb ApplicationContext XmlWebApplicationContext

WebSphere 时区问题

现有服务器时区和日期时间都对,在日志记录中WebSphere时区不对,应该怎么个调整?

版本为WebSphere 6.0、操作系统为 windows 2003。

答:

应用服务器:选择你的服务→
java 和进程管理:进程定义→
其它属性:java 虚拟机→

通用 java 参数:加入 -Duser.timezone=GMT+08 即可。



自动重启 WebSphere Application Server

李 彧 / 文

自动重启应用服务器不见得是每个系统必须的,假设一个系统应用不存在内存泄漏、数据库连接泄漏之类的情形的话。

应用服务器产品本身所具备的集群技术足以保证系统的高可用性,因为集群能够进行负载均衡和失败转移,即使集群的一两个成员停止在理论上以及在短时间内应该对业务不会造成过大冲击。

因此,自动重启应用服务器是在特殊需求下的命题作文,不具有普遍性。在此提出解题思路和实现方法之一,仅作为参考。

一、需求背景

某企业级应用系统作为核心系统之一,在后端的数据库服务器设置的数据库连接数是数以千计的,甚至高达 5000,开发团队对于核心代码的代码走查和性能测试还是常抓不懈的,但是对于使用频率低的应用代码缺少重视,以及新开发的紧急需求因时间紧代码走查、性能测试不到位,凡此种种,都有可能埋伏有内存泄漏和数据库连接持有不释放的隐患,积少成多,在数天之后即可对后端的数据库造成压力,数据库连接数达到甚至超过预设阈值,此时不是应用服务器重启,就是数据库重启,在这种情形下,应用服务器动作影响面比数据库小得多,因此就提出了应用服务器自动重启的特殊需求。

二、设计思路

1、停止 WAS

1.1 读取企业应用服务器实例列表。读取 server 列表是至关重要的一个环节。在这里我们使用 WAS 提供的命令 `serverStatus.sh` 来获取 server 列表。但是在对 `serverStatus.sh -all` 结果进行文本处理的时候,问题随之而来,如果操作系统默认的字符集不支持中文的情形下,与 server 相近的分隔符分号有中文全角字符双字节(;)和英文半角单字节(;)的两种截然不同的表示法。如何巧妙构造处理中英全半角表示的分号分隔符将是解决问题的关键。

如下为操作系统默认支持语言分别为中英环境下的真实实例

1.1.1 在中文环境下,以全角字符双字节(;)分号为分隔符:

```
[root@test1 bin]$ sh serverStatus.sh -all
```

```
ADMU0118E: 日志文件无法写到位置
```

```
/opt/IBM/WebSphere/AppServer/logs/serverStatus.log, 请使用  
-logfile 指定不同的位置
```

```
ADMU0503I: 正在检索所有服务器的服务器状态
```

```
ADMU0505I: 在配置中找到服务器:
```

```
ADMU0506I: 服务器名: interface
```

```
ADMU0506I: 服务器名: nodeagent
```

```
ADMU0506I: 服务器名: test2web1
```

```
ADMU0506I: 服务器名: test2ejb1
```

```
ADMU0506I: 服务器名: WebServicecest
```

1.1.2 在英文环境下,以半角单字节(;)分号为分隔符:

```
[root@test2 bin]$ sh serverStatus.sh -all
```

```
ADMU0118E: The log file cannot be written to location
```

```
/opt/IBM/WebSphere/AppServer/logs/serverStatus.log; please  
specify a
```

```
;different location with -logfile
```

```
ADMU0503I: Retrieving server status for all servers
```

```
ADMU0505I: Servers found in configuration:
```

```
ADMU0506I: Server name: interface
```

```
ADMU0506I: Server name: nodeagent
```

```
ADMU0506I: Server name: test1web1
```

```
ADMU0506I: Server name: test1ejb1
```

```
ADMU0506I: Server name: WebServicecest
```

针对中英文语言环境下分号分别为中文字符双字节与英文单字节的困扰,给出统一的一种可行的解决方法如下:

```
# 定义 WAS 的安装路径
```

```
Basepath="/IBM/WebSphere/AppServer"
```

```
# 获取 WAS 的 server 实例列表
appsvrlist=$(Basepath/bin/serverStatus.sh -all|sed 's/
ADMU05061://g'|grep -v nodeagent|sed 's/ //g'|sed 's/: /
:g'|awk -F ':' '{ORS=" "}{print $2}')
```

其中获取 server 列表的脚本命令解释如下：

- a、使用 serverStatus.sh -all 命令获取所有的；
- b、根据关键字 ADMU05061 搜索适配行；
- c、在结果集中过滤掉 nodeagent 行，以防将 nodeagent 进程误停；
- d、在结果集中将单字节的空字符串 (" ") 过滤掉；
- e、在结果集中将双字节全角分号 (": ") 替换为单字节半角分号 (":");
- f、根据统一的单字节半角分号为分隔符获取 server 列，然后 server 之间使用空格打印输出 server 列表。

1.2 依次停止企业应用服务器实例。使用 stopServer.sh 命令。

2、启动 WAS

2.1 读取企业应用服务器实例列表。

2.2 依次启动企业应用服务器实例。使用 startServer.sh 命令。

3、设定自动重启频度和时间

3.1 让业务决定重启策略。一般原则下，根据 7*24 不间断运行的要求下，是不大可能停机自启的。但是在没有办法的情形下，业务和 IT 部门经过协商也是可以相关 作出一定的妥协和让步。如可以将重启频度策略为一周一次，重启时间策略定在业务最闲、使用最少的短暂时间如在凌晨六点五分自停止 WAS，六分十分钟内自启动 WAS。

3.2 设定定时重启。在 unix 下，可以通过 crontab 进行设置。如上述提到的一周一次，重启时间，包括自停止和自启动两个时间点，假设为周六凌晨的六点五分停，六分十分启。

```
5 6 * * 6 command line
```

```
10 6 * * 6 command line
```

三、实现脚本

1、自动停止 WAS 脚本

```
#!/bin/sh
#stopwas4linux.sh
```

```
#carpnet
#
# 用法: stopServer <server> [options]
#options: -nowait
#-quiet
#-logfile <filename>
#-replacelog
#-trace
#-timeout <seconds>
# -statusport <portnumber>
#-conntype <connector type>
#-port <portnumber>
#-username <name>
#-password <password>
#-help
#
Basepath="/IBM/WebSphere/AppServer"
tmpappsvrs=$(Basepath/bin/serverStatus.sh -all|sed 's/
ADMU05061://g'|grep -v nodeagent|sed 's/ //g'|sed 's/: /
:g'|awk -F ':' '{ORS=" "}{print $2}')
```

```
nullstr=$(echo $tmpappsvrs|sed /[[[:space:]]*/d')
if [ "$nullstr" = "" ]
then
echo "[date] 服务器实例列表还是为空: $nullstr, 异常退出"
exit 0
fi
if [ "$tmpappsvrs" != "" ]
then
echo "[date] 开始读取 'hostname' 的应用服务器 WAS"
echo "[date] 所有实例为: $tmpappsvrs"
echo "[date] 结束读取 'hostname' 的应用服务器 WAS"
echo "[date] 开始停止 'hostname' 的应用服务器 WAS"
count=0
for srv in $tmpappsvrs
```



```
do
count='expr $count + 1'
echo "[date] 读取即将停止的应用服务器WAS实例($count)
[$srv]"
$Basepath/bin/stopServer.sh $srv -quiet &
done
# 停止进程过程需要一定的时间
sleep 200
echo "[date] 结束停止 'hostname' 的应用服务器 WAS"
fi
exit 0
```

2、自动启动 WAS 脚本

```
#!/bin/sh
#startwas4linux.sh
#carpnet
#
# 用法: startServer <server> [options]
#options: -nowait
#-quiet
#-logfile <filename>
#-replacelog
#-trace
#-script. [<script. filename>] [-background]
#-timeout <seconds>
#-statusport <portnumber>
#-help
#
Basepath="/IBM/WebSphere/AppServer"
tmpappsrvs=' $Basepath/bin/serverStatus.sh -all|sed 's/
ADMU05061: //g'|grep -v nodeagent|sed 's/ //g'|sed 's/: /:
/g'|awk -F ' ' '{ORS=" "}{print $2}'
nullstr='echo $tmpappsrvs|sed /[[space:]]*/d'
if [ "$nullstr" = "" ]
then
echo "[date] 服务器实例列表还是为空: $nullstr, 异常退出
```

```
"
exit 0
fi
if [ "$tmpappsrvs" != "" ]
then
echo "[date] 开始读取 'hostname' 的应用服务器 WAS"
echo "[date]?? 所有实例为:$tmpappsrvs"
echo "[date] 结束读取 'hostname' 的应用服务器 WAS"
echo "[date] 开始启动 'hostname' 的应用服务器 WAS"
count=0
for srv in $tmpappsrvs
do
count='expr $count + 1'
echo "[date]读取即将启动的应用服务器 WAS实例($count)
[$srv]"
$Basepath/bin/startServer.sh $srv -quiet &
done
# 启动进程过程需要一定的时间
sleep 200
echo "[date] 结束启动 'hostname' 的应用服务器 WAS"
fi
exit 0
```

四、应用成效

在生产环境下,操作系统默认字符集无论是中文还是英文,通过设置定时任务,自动定时运行以上脚本,能够如你所求正常的、自动停/启 WAS,而无需人工干涉。

五、结论

自动重启WAS的脚本关键是如何获取server列表,可通过WAS本身提供的命令serverStatus.sh -all后加以过滤来实现,然后通过停止、启动命令stopServer.sh、startServer.sh循环调用server列表批量停启server;再辅以操作系统crontab命令定时策略设定自动停启周期和时间,这样,通过以上三个步骤,就能够达到WAS停启的自动化目标。■

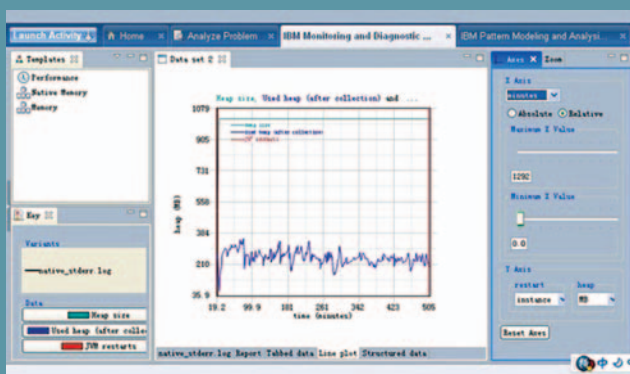
WebSphere Troubleshooting 用工具分析 GC Log

熊志辉 / 文

要进行 gc performance tuning, 不得不对 gc log 进行分析。之前说到了“人肉”的方法, 总觉得不够形象, 无法让不了解的开发人员抑或是技术负责人有个直观的了解, 所以本文介绍几个分析 GC log 的工具。

首先需要下载 IBM Support Assistant, 下载之后就可以从 Update-Tools add on 中下载我们需要的工具了, ISA 使用方法。ISA 把所有的工具集成在一个界面内, 省去了设置启动参数的麻烦, 同时能保持最及时的更新。分析垃圾回收日志, 我主要用“The IBM Monitoring and Diagnostic Tools for Java? 碟 Garbage Collection and Memory Visualizer”和“IBM Pattern Modeling and Analysis Tool for Java Garbage Collector (PMAT)”这两个工具。我会用实际例子来说明如何使用这个工具。

用 Garbage Collection and Memory Visualizer 载入 native_stderr.log, 首先你会看到下图:



这是一个500分钟的垃圾回收曲线图, 可以观察到一天以内的大致情况。总的来说, 蓝色的 Used heap (after collection) 运行在“平行通道”内, 没有走“上升通道”(炒股的朋友应该知道上

升通道的图形是咋样的)。所以在 Report 这个标签内, 可以看到“The memory usage of the application does not indicate any obvious leaks.”。

Report 中的 Summary 是需要关注的, 它向我们显示了 GC 发生的次数, 所用的 policy (optthruput), 平均停顿时间 248ms, 平均间隔时间 3.37 分钟, 还有垃圾回收的速率 (垃圾产生多并非不好, 反而是吞吐率高的一种表现)。

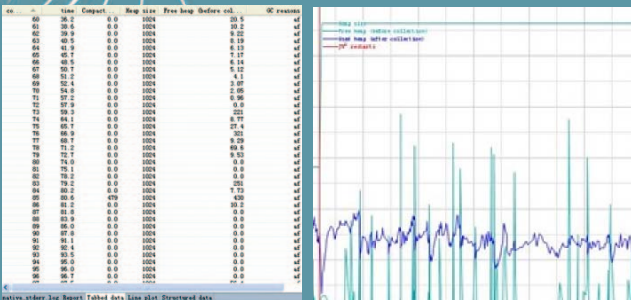
让我们再切回 Line plot 视图, 现在可以框选某一个时间段进行放大, 同时在右边 Axes 中选择 X 轴的坐标系, 默认的是相对时间, 以分钟为单位, 适用于你的应用程序总在启动 N 个小时后出现问题。如果是每天固定时间发生性能问题, 那么应该选用绝对时间。

Summary

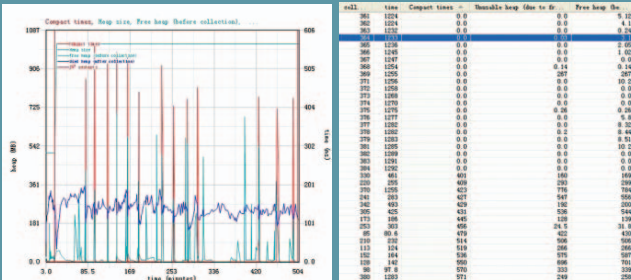
Allocation failure count	384
Concurrent collection count	0
Forced collection count	0
Full collections	0
GC Mode	optthruput
Largest memory request (bytes)	1484368
Mean garbage collection pause (ms)	248
Mean heap unusable due to fragmentation (MB)	38.8
Mean interval between collections (minutes)	3.37
Number of collections	384
Proportion of time spent in garbage collection pauses (%)	0.12
Proportion of time spent unpaused (%)	99.9
Rate of garbage collection	195.814 MB/minutes
Total amount tenured (MB)	95646

默认的曲线开启了 Heap size 和 Used heap (after collection), 你可以根据需要, 在 VGC Pause Date 和 VGC Date、VGC Heap Data 中勾选你需要查看的曲线。比如你觉得程序响应时间很长, 那么可以勾选上 Intervals between garbage collection triggers 和 Pause time, 看看上一条曲线是否和下面一条靠的“太近”。

通过在 Tabbed data 查看表格模式的回收信息 (同样可以勾选上节提到的内容来增加列), 我们看到 GC reason 都是 af。通过查看 Free heap before collection, 我们发觉有些 af 发生的时候, 堆中剩余空间其实还很多。



我覆盖上 Compact times 曲线，发现十分吻合，原来这时候是堆中的碎片太多，所以导致即使剩余空间很大，但是没有连续空间分配给新的请求，这时候就发生了压缩（Compact）。加了Unuseable heap due to fragmentation的第二副图可以更明显的看到这一点。进一步的，可以根据第一列的 collection id 去查看引起压缩的 requested_bytes，是否需要比较大的空间（比如大于64K），然后结合一下所有 requested_bytes 中大对象的数量，确定是否需要调整 LOA 大小。



你会发现有些曲线是灰色不能选择状态，那是因为根据我现在的 gc policy，关于gencon 模式下的 nursery 和 tenured 曲线在这个模式下是没有的。

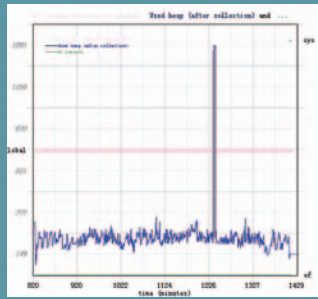
总的来说，运用不同的曲线组合，合适的缩放比例，不同的日志显示方法对照，我们可以验证gc是否符合我之前说过的一些“最优原则”——回收的间隔是否大大大于持续时间，整个回收时间是否在整个运行时间的5%以下，压缩的次数是否仅占一小部分（太多的话要考虑设置 -Xk -Xp 和调整 LOA）……Report 标签中的 Tuning recommendation 是你需要参考的内容，但不要一上来就根据它的提示去优化。

下一个例子就是一个有问题的垃圾回收日志。

整个垃圾回收曲线，当中有一个 Excessive GC 引发的 OutOfMemory

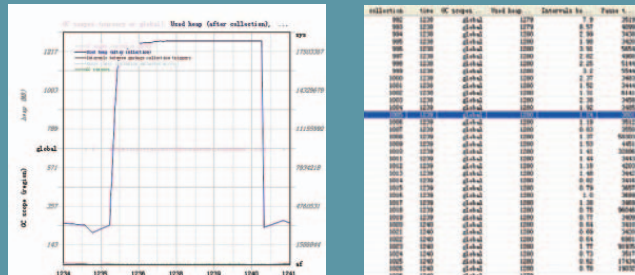
Excessive GC Time and OutOfMemoryError

The concurrent collector will throw an OutOfMemoryError if



too much time is being spent in garbage collection; if more than 98% of the total time is spent in garbage collection and less than 2% of the heap is recovered, an OutOfMemoryError will be thrown. This feature is designed to prevent applications from running for an extended period of time while making little or no progress because the heap is too small.

放大出问题的地方的曲线



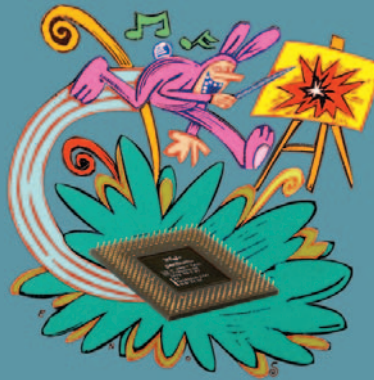
根据 ID 查看原始文件

```
<af type="tenured" id="1030" timestamp="Tue Nov 04 14:57:26 2008" intervalsms="11,440" >
  <minimum requested_bytes="216" />
  <time exclusiveaccessms="0.322" />
  <tenured freebytes="0" totalbytes="1342177280" percent="0" >
  <soa freebytes="0" totalbytes="1342177280" percent="0" />
  <loa freebytes="0" totalbytes="0" percent="0" />
</tenured>
```

```

<gc type=" global" id=" 1031" totalid=" 1031"
intervalms=" 11.837" >
  <compaction movecount="3301258" movebytes="196674784
" reason=" low free space (less than 4%)" />
  <refs_cleared soft=" 0" weak=" 15" phantom=" 7" />
  <finalization bjectsqueued=" 40" />
  <timesms mark="1753.108" sweep="10.302" compact="
1858.390" total=" 3622.032" />
  <tenured freebytes=" 519040" totalbytes=" 1342177280"
percent=" 0" >
    <soa freebytes="519040" totalbytes="
1342177280" percent=" 0" />
    <loa freebytes=" 0" totalbytes=" 0"
percent=" 0" />
  </tenured>
</gc>
<warning details=" excessive gc activ-
ity detected" />
  <tenured freebytes=" 518352"
totalbytes=" 1342177280" percent=" 0" >
    <soa freebytes="518352" totalbytes="
1342177280" percent=" 0" />
    <loa freebytes=" 0" totalbytes=" 0" percent=" 0" />
  </tenured>
<time totalms=" 3623.033" />
</af>
<af type=" tenured" id=" 1031" timestamp=" Tue Nov 04
14:57:30 2008" intervalms=" 3.715" >
  <minimum requested_bytes=" 120" />
  <time exclusiveaccessms=" 0.232" />
  <tenured freebytes=" 0" totalbytes=" 1342177280"
percent=" 0" >
    <soa freebytes=" 0" totalbytes=" 1342177280" percent="
0" />
    <loa freebytes=" 0" totalbytes=" 0" percent=" 0" />

```



```

</tenured>
<gc type=" global" id=" 1032" totalid=" 1032"
intervalms=" 4.385" >
  <compaction movecount="3170275" movebytes="192337800
" reason=" compact to aid heap contraction" />
  <contraction type=" tenured" amount=" 67108352"
newsize=" 1275068928" timetaken=" 0.001" reason=" excess
free space following gc" />
  <refs_cleared soft=" 0" weak=" 0" phantom=" 0" />
  <finalization bjectsqueued=" 9" />
  <timesms mark="245.373" sweep="12.
272" compact=" 482.748" total=" 745.486
" />
  <tenured freebytes=" 1022923992"
totalbytes=" 1275068928" percent=" 80" >
    <soa freebytes=" 1022923992"
totalbytes=" 1275068928" percent=" 80" /
>
    <loa freebytes=" 0" totalbytes=" 0"
percent=" 0" />
  </tenured>
</gc>

```

考虑到垃圾回收频率也较高,每次请求的空间都不大,所以我讲垃圾回收模式改为了gencon,之后没有OutOfMemory发生。不过我那份回收日志没有留下来,否则应该在文件Compare file中导入修改后生成的日志,进行比较,确定修改的结果。

注意事项: 对于 IBM JDK,直接勾选“详细垃圾回收”即可,但是对于HP 或者Sun JDK,最好使用-Xverbosegclog:file_name(生成能解析的XML格式)来指定一下,否则默认文件中会包含其它信息,工具讲无法以图像显示,还要自己修改文件,比较麻烦的。

IBM Pattern Modeling and Analysis Tool for Java Garbage Collector (PMAT)这个工具大同小异,个人使用它只是觉得提供的优化建议可以和上述工具互补,比如它会计算一个适合的-Xk -Xp值出来。■

一、WAS 配置 ND 集群后重启后服务器不能正常启动 (DM、NodeAgent 和 server 三者之间的启停关系)



WAS 配置 ND 集群后重启后服务器不能正常启动

AppSrv01\logs 下的 SystemErr.log 日志

```
[09-6-12 9:41:20:312 CST] 0000000a SystemErr R com.ibm.ws.exception.
```

```
RuntimeException: com.ibm.ws.exception.RuntimeError: com.ibm.ejs.EJSEException:
```

Could not register with Location Service Daemon, which could only reside in the NodeAgent. Make sure the NodeAgent for this node is up an running.; nested exception is:

```
org.omg.CORBA.ORBPackage.InvalidName: LocationService:org.omg.CORBA.
```

```
TRANSIENT: java.net.ConnectException: Connection refused: connect:host=dfpv10697, port=9900 vmcid: IBM minor code: E02 completed: No
```

```
at com.ibm.ws.runtime.WsServerImpl.bootServerContainer(WsServerImpl.java:194)
```

```
at com.ibm.ws.runtime.WsServerImpl.start(WsServerImpl.java:133) .....
```

答

日志信息 “Could not register with Location Service Daemon, which could only reside in the NodeAgent. Make sure the NodeAgent for this node is up an running”, 可以看出是 nodeAgent 没有启动导致不能启动服务器。

DM、NodeAgent 和 server 三者之间的启停关系如下:

1. DM 与 NodeAgent

两者的启停没有顺序关系。可先启动 (或停止) DM, 再启动 (或停止) NodeAgent, 也可以先启动 (或停止) NodeAgent 再启动 (或停止) DM。

2. NodeAgent 与 Server

(1) 必须先启动 NodeAgent, 才能启动 server。

(2) 停止 NodeAgent 不会影响 server 的状态, 即执行 stopNode.sh 命令前, 若 server 是运行状态, 在执行 stopNode.sh 命令后, server 不会停止, 而是仍然运行。

(3) server 的启动状态还可以通过参数配

置设定。

参数设置位置为 “应用程序服务器 == > server1 ==> 服务器基础结构 ==> Java 和进程管理 ==> 监视策略”

A. “自动重新启动”: “自动重新启动” 设置打勾, 无论下面的设置如何, 都会自动重启的。比如当 WAS server 因某种原因宕掉, 或者直接用 kill -9 杀掉之后, 它会自动重新启动的。意指除了 stopServer 正常指令之外的一些意外导致失败的情况下, 系统会自动重启

B. “节点重新启动状态”, 此处有三个选项为: STOPPED、RUNNING、PREVIOUS。缺省值是 STOPPED。

如果此属性设置为 STOPPED, 则 Node Agent 启动时不启动该服务器。

如果此属性设置为 RUNNING, 则 Node Agent 启动时始终启动该服务器。

如果此属性设置为 PREVIOUS, 则仅当该服务器在 Node Agent 停止时正在运行, Node Agent 启动时才会启动该服务器。◆

三、经常有 heapdump.***.phd 和 javacore.***.txt 文件产生



发现在 ApSrv01 下面有 heapdump.***.phd 和 javacore.***.txt 文件产生, 有时会

很快将空间充满, 请问是什么原因呢? 如何解决比较好?

答

生成这两个文件, 说明有 OutOfMemory 发生, 可以查看 SystemErr.log 确定发生的原因, 是参数设置有问题, 还是程序有内存泄漏。

“首先查看 systemerr.log 确定内存溢出的原因, 是 PermSize 太小, 还是 Native thread 无法创建, 前者用 MaxPermSize 调整 (IBM JDK 没这个参数), 后者调小最大堆大小或者 Xss 调整每个线程分配内存的大小。

如果是常见的堆的溢出, 确保 OutOfMemory 时能生成 heapdump 文件, 用 Dump analyzer 或者 MDD4J 分析 dump 文件, 找到堆中占用空间总数最大的 (或者数量最多的) 对象。然后调整堆范围到一个比较小的区间, 比如 256M ~ 384M, 重新启动服务器, 在运行 1 小时手动做一次 heapdump, 运行 4 小时后做一次 heapdump, 运行 8 小时做一次 (间隔仅作参考)。然后分析一下三者的区别, 看看哪个对象数量增长很多, 占用空间增加很大。结合 OutOfMemory 时候的分析, 应该能锁定问题的源头。”◆

二、如何备份部署管理器



如何备份部署管理器, 是将整个目录拷贝下来, 以后新装并替换目录吗? 还是只要备份一些文件, 具体是哪些?

答

DM 是 WAS 中的单点故障, 但由于 DM 启停状态不会影响应用服务器的运行, 所以我们一般只需做好 DM 的备份就行。可以将整个目录打包拷贝下来, 以后新装并替换目录。◆

四、应用程序的“类装入和更新检测”



集群环境下，设置应用程序的“类装入和更新检测”，勾选“更新应用程序文件时重新装入类”，然后确定，保存（保存时，会去同步节点）。保存好了，结果发现应用程序自动停止了，状态变为“×”了。马上启动应用程序，结果提示不能启动，说有的节点没有同步，可再点“企业应用程序”，确实发现应用程序已经启动了。

有3个问题：

1. 为什么应用程序会自动停止，是否正常。
2. 为什么明明报错，结果应用程序还是启动了（我访问应用程序没问题的）。
3. 要是上面的“类装入和更新检测”没有勾选中“更新应用程序文件时重新装入类”，怎样让 Websphere 服务器重新加载已修改的 jsp 页面，删除缓存可以达到目的不？

答

马上启动应用程序，结果提示不能启动，说有的节点没有同步，是因为节点还没有完全同步完成而报错，很正常应当好理解这个错误。

1. 为什么应用程序会自动停止，是否正常。

当修改应用程序的配置后，会自动进行停止，然后再重启，是正常的。

2. 为什么明明报错，结果应用程序还是启动了（我访问应用程序没问题的）。

是报还没有集群间同步完成的错误，后面它还是会执行重启应用的线程，因此会启动成功就正常访问。

3. 要是上面的“类装入和更新检测”没有勾选中“更新应用程序文件时重新装入类”，怎样让 Websphere 服务器重新加载已修改的 jsp 页面，删除缓存可以达到目的不？

3.1 在部署期间，可以在“JSP 重新装入选项”面板上启用 JSP 重新装入功能。

3.2 部署后，也可以通过修改 企业应用程序 > XXX > Web 模块的 JSP 重新装入选项，来达到加载更新 JSP 的目的。

一般来讲是根据 classes 的时间戳来进行判断是否更新的类，如果时间戳不一致就会进行重新加载。而且这个更新是有前提条件的。与操作系统的文件更新有关，与更新的设置也有关。

比如：在某些情况下 jar 可增不可减不可替，jar 与 classes 的加载顺序不同。◆



五、WAS 完全卸载问题

答

通常情况下，推荐使用 WebSphere 提供的卸载程序来卸载 WebSphere，而不要使用操作系统的卸载软件的功能来卸载 WebSphere，因为那样步骤比较复杂，并且有可能由于漏掉某些步骤而导致不能完全卸载干净。卸载 WebSphere 之前，需要停止所有 WebSphere 相关的进程，最好查看系统中的进程确保没有 java 进程运行，另外，如果使用了 Web 服务器，确保停止了 Web 服务器。然后运行 WebSphere 的卸载程序，卸载程序命令在 UNIX 系统中为 uninstall，Windows 中为 uninstall.exe，存在于 WebSphere 的安装目录下的 _uninst 目录中。例如，在 UNIX 系统中，将目录转至 install_root/_uninst 目录 (install_root 为 WebSphere 的安装目录)，运行 uninstall 命令，根据图形向导来卸载 WebSphere。◆



六、忘记了 WebSphere 控制台登陆密码，怎么办？

答

管理控制台密码遗忘有两种补救措施：

方法一：命令行——从 \$WAS_HOME/profiles/xxx 概要文件名/bin 目录下，运行 wsadmin -conntype NONE 。当 wsadmin 的命令行窗口出现之后，运行 securityoff 。上述操作在应用服务器启动或停止的状态都能发出。再次启用 WAS 时，就是停用管理安全性的状态了。

· \cells\xxx 下的 security.xml，把第一个 enable 改成 false 就取消安全性了。
· 详细说明 <http://www-01.ibm.com/support/docview.wss?uid=swg21105430>

方法二：修改配置文件——修改 \$WAS_HOME/config 后别忘了设置新的密码然后再次启用。◆

七、Was 6.1 集群中有一个节点显示的状态是“未同步”



发现 was6.1 集群中有一个节点显示的状态是“未同步”，使用了以下方法，但未凑效：

1. 选中未同步的节点，点击“同步”。
2. 选择所有集群下的节点，点击“全部再同步”。
3. 重新启动该未同步的节点。
4. 重新启动集群。

报错信息如下：

```
[6/16/09 8:52:56:014 CST]
00000033 NodeSync E ADMS0005E:
The system is unable to generate syn-
chronization request: javax.
management.JMRuntimeException:
ADMN0022E: Access is denied for the
getRepositoryEpoch operation on
ConfigRepository MBean because of in-
sufficient or empty credentials.
```

```
NodeSync.getCellRepositoryEpoch
(NodeSync.java:399)
    at com.ibm.ws.management.sync.
NodeSyncTask.doSync(NodeSyncTask.
java:261)
    at com.ibm.ws.management.sync.
NodeSyncTask.run(NodeSyncTask.java:
181)
    at java.lang.Thread.run(Thread.java:
810).
    at com.ibm.ws.management.
connector.soap.SOAPConnectorClient.
handleAdminFault(SOAPConnectorClient.
java:811)
    at com.ibm.ws.management.
connector.soap.SOAPConnectorClient.
invokeTemplate(SOAPConnectorClient.
java:779)
    at com.ibm.ws.management.
connector.soap.SOAPConnectorClient.
invoke(SOAPConnectorClient.java:575)
    at com.ibm.ws.management.
connector.soap.SOAPConnectorClient.
invoke(SOAPConnectorClient.java:402)
    at $Proxy0.invoke(Unknown Source)
    at com.ibm.ws.management.
AdminClientImpl.invoke(AdminClientImpl.
java:205)
    at com.ibm.ws.management.sync.
```

答

检查了<AppServer profile root>/config/cells/<cellname>/security.xml 和 <DMGR profile root>/config/cells/<cellname>/security.xml 两个文件，发现两个文件是一致的。

费了好大的劲，终于找到原因了。原来是因为ND服务器的时间与那个同步不过来的节点服务器的时间不一致导致的，将二者时间同步之后，重启集群及节点，问题解决。◆



八、websphere 全局安全性设置

答

websphere v6 安装完成后，登录到管理控制台 http://localhost:9060/admin/，此时可以用任何名称(包括空，而且不用输入密码)就登陆到管理控制台进行管理操作。此时首页提示“未启用安全性”。

如果想锁定管理控制台，只允许某个用户行使管理权限，可以登陆到控制台(http://localhost:9060/ibm/console)，开启安全性设置。

(1)登陆到管理控制台，进入“系统管理—控制台设置—控制台组”，添加控制台组，就是安装前创建的组 linux 上为 wsadmin, windows 上为 wasadmins，记得要保存设置。

(2)进入“系统管理—控制台设置—控制台用户”，

添加控制台用户，就是前面创建的websphere 用户。linux 上为 wsadmin, windows 上为 wasadmin. 保存设置。

(3)到“安全性—全局安全性”里进行安全性设置。找到“用户注册表—本地OS”进行用户注册。服务器用户标识：wsadmin (和第二步中设置的用户名要一样)。服务器用户密码：系统中创建 wsadmin 用户时指定的用户密码，保存设置。

(4)现在要启用全局安全性了。选中“启用全局安全性”复选框，切记要把“强制 Java2 安全性”给勾掉即不选中。系统默认启用全局安全性时会启用强制 Java2 安全性，如果没有设置 Java2 安全性(jvm 授权)，这里却启用了强制 java2 安全性，服务可能会启动不了。

重新启动服务器，登陆控制台 http://localhost:9443/ibm/console，全局安全性成功启用。◆



九、修改机器名后 WebSphere、DB2 不可使用

答

SQLLIB\DB2 目录下 db2nodes.cfg 文件中的下面一行：

```
0servername servername 0
```

没错，如果修改了机器名，可在 listener.ora 文件修改文本，

```
# LISTENER.ORA
Network Configuration
File:X:\Oracle\Ora81\
NETWORK\ADMIN\l
istener.ora ◆
```



DeveloperWorks 中国网站 WebSphere 专区

WebSphere 与 SOA 新手入门

顾名思义,SOA是一种体系结构——以业务为中心的IT体系结构。SOA的构件是服务,后者代表离散的、可重复的、可重用的业务任务。您能够以多种方式组合和重新组合这些任务,以构建集成多种来源的功能来支持不同业务需要的应用程序。通过本专题,您将了解 WebSphere 产品如何帮助您进行 SOA 项目从建模到部署的各个实现阶段,以适应面向服务的体系架构解决方案。

更多信息请访问:

ibm.com/developerworks/cn/websphere/newtosoa/

WebSphere 与 SOA 专区

DeveloperWorks WebSphere SOA 专区为您提供了有关使用 WebSphere 产品对面向服务的体系结构(SOA)进行建模、组装、部署和管理的最新技术信息。

更多信息请访问:

ibm.com/developerworks/cn/websphere/zones/soa/

WebSphere Web services 新手入门

Web 服务是一个软件接口,它描述了一组可以在网络上通过标准化的 XML 消息传递访问的操作。它使用基于 XML 语言的协议来描述要执行的操作或者要与另一个 Web 服务交换的数据。有几种 WebSphere 产品提供了 Web 服务功能。其中最主要的是 WebSphere Application Server 以及 WebSphere Application Server Network Deployment 的 Web 服务网关组件等。本专题为您提供关于 WebSphere 如何支持 Web 服务的简要概述。

更多信息请访问:

ibm.com/developerworks/cn/websphere/zones/webservices/wswebsvc.html

WebSphere Web services 专区

DeveloperWorks WebSphere Web services 专区提供了大量文章、教程、代码示例

等最新技术资源,以帮助您使用有关 WebSphere 产品创建、测试、注册、部署、管理和使用 Web 服务。

更多信息请访问:

ibm.com/developerworks/cn/websphere/zones/webservices/

WebSphere 与 SOA 认证专栏

想成为“IBM 认证业务分析师”、“WebSphere MQ 解决方案设计师”或者“SOA 解决方案设计师”吗?获得 IBM 专业认证是每一个技术爱好者的实力的证明。在本专栏中,我们把与 WebSphere 和 SOA 相关的 IBM 认证教程、资源汇总在一起,便于您浏览、学习。这些信息将不断更新、丰富,以协助您顺利通过 IBM 认证。

更多信息请访问:

ibm.com/developerworks/cn/websphere/zones/certification/

构建 SOA 组合业务服务专题

本专题通过一个银行业的示例向您全面地介绍了如何构建 SOA 组合业务服务以及相关方面的知识。在构建 SOA 组合业务服务的过程中,涉及到了很多 IBM 的产品,比如 WebSphere Process Server, WebSphere Integration Server, WebSphere Portlet 等,几乎包含了 IBM 所有与 SOA 相关的产品和解决方案。

更多信息请访问:

ibm.com/developerworks/cn/webservices/lp/composite/

IBM 业务流程管理期刊

IBM 业务流程管理期刊致力于为业务人员和 IT 专业人员提供关于 IBM 业务流程管理(BPM, Business Process Management)工具与技术的最新信息。该期刊分为特色文章、问与答、专家访谈等栏目,内容上涵盖了 IBM 业务流程管理核心产品的功能介绍,最佳实践,常见问题的解答集锦等方面

的内容。

更多信息请访问:

ibm.com/developerworks/cn/websphere/bpmjournal/

IBM WebSphere 业务流程管理专区

IBM WebSphere 提供了对业务流程进行建模、组装、部署和管理所需要的工具。本专区将为您提供业务流程管理相关的技术文章、教程、多媒体课堂等最新技术资源。

更多信息请访问:

ibm.com/developerworks/cn/websphere/zones/bpm/

IBM WebSphere BPM V6.1 入门

使用 IBM BPM (Business Process management) 产品,您可以轻松地协调、管理和监视您的业务流程、人工活动和业务活动。本文通过示例向您介绍了如何使用 IBM BPM 产品对使用 Lotus 表单的业务流程进行建模,如何为该流程指定业务度量,如何组装可部署的业务流程,如何生成业务客户端,以及如何构建监视模型。

更多信息请访问:

ibm.com/developerworks/cn/websphere/techjournal/index.html

IBM WebSphere 开发者技术期刊

IBM WebSphere 开发者技术期刊是 DeveloperWorks 站点 WebSphere 专区的核心刊物。每月推出一期,邀请 WebSphere 专家撰写,内容涵盖前瞻性的 WebSphere 技术讨论、核心产品的最佳实践、专家就读者提问的解答集锦等。栏目包括:“特色文章”、“系列连载”、“权威支持”、“专家评论”、“EJB 倡导者”和“专家访谈”。文章力求精益求精,旨在为 WebSphere 爱好者提供一个学习、交流的高质量的平台。

更多信息请访问:

ibm.com/developerworks/cn/websphere/techjournal/index.html



读者反馈

www.WebSpherechina.net



我们希望了解你的需求，
请简单的完成下面问题并传真给我们。

传真：010-68867107-805

印象最深的栏目：_____

希望了解更多信息的栏目：_____

最想从本刊中了解的内容：_____

对 WebSphere 这一技术的哪些侧重点感兴趣：_____

本期喜爱的文章：_____

本期不喜爱的文章：_____

请剪下并传真此页给我们。欢迎您成为 WebSphere 开发与应用网站会员，在收到您的信息并审核通过后，我们将免费为您邮寄杂志。同时您可以登录 WebSphere 开发与应用网站：www.WebSpherechina.net，与更多 WebSphere 专家在线互动交流。

姓名：_____ 性别：男 女 电子邮件：_____ 电话：_____ 传真：_____

单位：_____ 部门：_____ 职位：_____

地址：_____ 邮编：_____

地址：北京市石景山区石景山路23号中融大厦530室 (100043) 广告：010-68867107-806 邮件：chao_huang@twtgrou.com.cn





SunGard是世界领先的软件和IT服务企业，服务全球70多个国家，客户超过25,000位，包括世界上最大的25家金融服务企业。

SunGard为金融服务、高等教育和公共部门提供软件和流程解决方案，亦提供灾难恢复服务、IT管理服务、信息连续性咨询服务和业务连续性管理软件。

SunGard年收入超过50亿美元，位列财富500强第472位，在福布斯私有企业排行榜中位列私有软件和服务企业第一名。《Continuity, Insurance & Risk》杂志已史无前例的第五次将SunGard评为年度服务供应商。

更多信息，请登陆www.sungard.com。

- 全球第三大应用软件企业 (*Datamonitor, 2009)
- 经纪商及其他金融机构每天利用SunGard系统处理的交易达500万笔
- SunGard系统目前为25万亿美元的投资资产提供日常管理服务
- 资本市场领域供应商第一名 (*Source: American Banker/FinTech 100)

公司历史

SunGard成立于1982年，由Sun Oil Company以杠杆收购方式收购。公司于1986年在纽约股票交易所上市。2005年8月，银湖合伙公司 (Silver Lake Partners) 控制的私募股权基金公司通过杠杆收购以115亿美元代价将该公司转为私有。

一个空间，多种交流

WebSphere开发与应用社区

<http://www.webspherechina.net>

WebSphere开发与应用社区
是一个非盈利的用户组织，
为大家提供交流学习、
职业发展、
商务合作的中文协作平台。

