

2007年11月CSAI辅导班内部论文范文

论文范文总共79篇，都是从希赛的网上辅导平台

论分布式数据库的集成

摘要:

本文讨论了某公司发货系统的分布式数据库集成解决方案。该公司由于业务的发展,要在另三个城市设立货仓进行发货。为此,需要增加原先的 MIS 系统实现这一功能。公司委任我作为项目经理完成系统的设计和开发的工作。我经过分析,使用了 Sybase 的分布式数据库技术。我设计的这个系统是采用典型的 C/S 结构,但客户端连接服务器的网络采用电话线拨号,速度有限,传统 Windows 界面的客户端应用程序相应速度比较慢。于是我采用了优化数据库结构的方法,把数据分两部份存放,基础数据放客户机,销售资料主要采用键码放服务器,应用程序再现数据时从服务器取键码,到客户机取对应的解释。由于键码的数据量少,网络传输便快。在构建这个分布式数据库系统的过程中,我着重研究并解决了数据同步和事务协调的问题,到得了良好的应用效果。

正文:

2004 年 3 月,由于公司业务的发展,要求在其它三个城市设立货仓,处理发货业务。公司本部运行着一套用 Sybase 数据库的 MIS 系统可以实现发货,该系统用的是 C/S 结构。由于客户端连接服务器的网络采用电话拨号,所以直接把客户端软件直接安装在外地访问本部数据库,速度很慢。于是,公司成立了一个项目,专门解决这个问题。在这个项目中,我担任项目经理。经过对现有系统的分析,我们决定利用 Sybase 提供的技术,采用分布式数据库集成的方法来改造目前的系统使之能适应新的需要。项目分三个阶段进行,一是进行需求分析,确定要增加的功能。二是进行系统设计,改变后数据分布如何,系统架构如何。最后是实现和测试,上线。整个项目历时从分析到实现历时三个月,最后于 2004 年 6 月份系统成功上线。

在分析阶段时我发现由于客户端地域的分散,遍及三个省境内,连接服务器数据库的网络采用电话拨号方式,速度有限,在使用客户端应用程序时感觉界面速度很慢。我经过分析,认识到许多操作都要从服务器中取数据,速度慢就慢在数据访问上。服务器是没有瓶颈的,问题出在网络速度上。出于成本和业务量方面的考虑,公司不会用专线连接,只能是电话拨号。这时只能改变目前软件的实现方法,来适应这种低速网络的使用模式。

经和项目组的人员一起探讨,结合关系数据库的知识,我认识到,应用程序的每一次数据库操作,都要访问多个相联的表,其中,有销售订单表和物料基础数据表/客户资料表/货仓的基础数据等。销售订单表中存放着出销售的订单编号,成品编号等,数据量少。而基础数据表就则放着成品的相关信息,有大量的数据。如果考虑把销售订单放在服务器,基础数据放在客户端,当应用程序中访问数据时,总是从服务器上存取销售订单,从客户端提取成品/订单的详细信息。由于订单的数据量少,便减少了网络上传递的数据量,从而提高了界面的响应速度。

把数据分散存放只是工作的第一步,接下来要考虑应用程序怎样访问这种分布式数据。开发应用时,如果每一功能都针对两个数据库进行,就带来了许多麻烦。所以,我通过研究 Sybase 的分布式数据库技术,决定采用 CIS(组件集成服务)部件,来合并两个数据库成一个统一的分布式数据库。应用程序只要连接一个数据库,就可以透明统一访问到两个数据库中的数据。

该技术具体实施方法是:在客户端数据库中建立一个对服务器数据库的远程访问服务名,包含访问地址,登录用户名,登录密码等关键的连接信息;前且对服务器中销售订单建立一个本地代理表。结构和服务器中远程表完全一样,它是访问服务器中会员资料的中转和代理。客户端应用程序访问本地代理销售资料表时,实际上是通过预先定义的远程访问服务名中包含的连接信息到服务器中对应的实际销售资料表中访问数据。这种访问对于客户端完全透明,感觉不到是从物理上独立的两个服务器中存服数据。所以,这种数据库结构是典型的分布式数据库。

部署这种分布式数据库不是难事,只要在客户端和服务服务器上安装 12.0 版本以上的数据库服务器,在客户端服务器上建立远程服务名和代理表即可。由于 Sybase 数据库的安装支持脚本方式,在客户端应用程序的标准安装过程中,嵌入 Sybase 数据库的安装和配置脚本,就自动化地完成了所有工作。

在实际使用该分布式数据库系统的过程中,遇到了几个问题,第一,数据同步。客户端基础数据不是绝对静态的,也有变化,因此在服务器要设置一个统一的基准,称为主点数据。客户端总是要复制使用,称为复制点数据。如何及时感知到服务器端主点数据的变化,有效率地复制到客户端,是个难题。Sybase 针对这种应用场合,提供了复制服务器技术,但为了避免过于复杂,我们采用实际应用程序来管理同步。当服务器端主点数据有了更改时,保存一个相应的标识和时间戳,客户端应用在登录服务器时,检查这些标识,一检测到了数据有更新,就首先下载,然后再进入系统正常使用。这种方法实现起来,增加了额外的开发量,且不能判别绕过应用程序对数据的直接修改,但是,是最简单和有效的方法。

第二个问题是事务协调问题。物理上独立的两个数据库,在协同操作时,如果服务器正好停机或者网络故障,完整的一个事务没能完成,就会“事务崩溃”。虽然 Sybase CIS 内嵌了两阶段提交技术,能够自动恢复。但是应用程序在这种情况下,敏感性不够,操作界面会无端凝固,影响了使用的方便性。我针对 PB 对连接判断和感知,用了一个小小编程技巧,使应用程序能够及时感知到数据库连接故障,及时停止和恢复事务,使操作界面表现友好灵活。

在具体的应用中,我们在三个城市安装了增强的客户端应用程序,同时安装了 Sybase 数据库。初始化时,把基础数据放从公司本部的数据库导入客户端的数据库中。用户在外地进行发货时,先拨号上网,然后启动客户端程序。在登录过程中,客户端程序会检查服务器上的标识和时间戳检查这些主数据是否有更新,如果有就先下载,下载完成后进入系统正常使用。在服务器更新的数据比较多的情况下,下载的时间会比较长,这时如果遇到急需发货,则会影响到货物不能及时发出去。为了解决这个问题,我设置了在每天凌晨的某个时刻自动登录和启动客户端程序,在下载更新数据完成后自动关闭。这样可以把一部份数据更新的内容放在非工作时间内完成,减少了发货登录的时间。用户登录后开始进行发货操作。输入销售订单,通过本地代理表系统自动到服务器获取该销售订单的数据,如发货的数量,客户编号等。而一些基础数据则可以直接从本地的数据库中得到,如销售产品的描述,客户的地址/电话/传真,发货的库位等。完成出货动作后,会自动更新服务器的库存。而这一更新通过提交事务在后台进行,不影响前台的操作。所以,对用户来讲,能够进行正常的操作,不会因为速度慢而进行不下去。

在当今的信息社会里,互联网带来了相互连通的方便,而且知识爆炸,数据的分布式访问是个必然的趋势。目前新起的 XML 技术,提供了各种平台数据库之间的一个公共数据访问标准,可以用来构建更加灵活,适应性更强的分布数据库技术。将 XML 用在分布式数据库中,将是未来的一个趋势。

论软件工程度量

摘要

随着计算机科学技术的不断进步，在教育系统中越来越多的学校开始应用教育管理系统来实施日常的教育教学工作。我有幸参与了公司《***数字校园》——中小学学校管理信息系统软件项目的设计开发全过程，并担任项目管理任务。

本文结合本人工作的实际经历，通过对学校管理信息系统软件项目实施过程进行描述，讨论了软件工程活动中，为了把握软件工程过程的实际情况和所生成产品的质量，需要对进度、质量、人员等方面进行度量。文中着重分析了在本项目中使用度量方法改进开发过程的实施手段。

正文

早在 2000 年教育部就颁布了在中小学实施“校校通”工程的通知，决定从 2001 年起，用 5 年到 10 年左右的时间，在全国中小学基本普及信息技术教育，全面实施“校校通”工程。经过几年的建设，很多学校的网络硬件环境已经形成，但真正在学校教学、研究和管理中发挥作用的高质量软件产品并不多见。我们公司以前曾经为某个学校开发过 c/s 结构的教育管理系统，当看到目前这个教育信息化的大趋势后，就想以一款技术含量高、可扩展性强并符合国家标准和规范的校园网应用软件《***数字校园》产品进入教育市场。

项目确立之初，我们与两所学校签订了开发意向，并要求在半年内拿出运行可靠的软件产品。作为运行在网络系统高速公路上的“车”的应用系统，主要是为校园网群不同层次提供应用性支撑软件系统，两所学校现阶段的要求是建立学校网络管理信息系统和通信信息服务系统。于是确定《***数字校园》的功能以学校管理信息系统和信息服务系统为主框，其中办公管理信息系统覆盖教育教学管理的各方面，包括：学校基本信息管理、基本教务管理、计划管理、学籍管理、备课管理、考试管理、成绩管理、排课系统、学生管理、教职工管理、教学计划管理、图书信息管理、仪器设备管理、房产管理等；信息服务系统是为丰富校园的文化生活服务的，包括：学校教育信息服务、校园风貌、教育园地、电子邮件、BBS 学生讨论区、教师科研信息等版块。

由于系统的最终交付日期已经确定，如果进度安排落空，会导致市场机会的丧失，用户会不满意，甚至还会要求赔偿经济损失，从而导致成本的增加。因此在项目真正实施前，我们就确定了以进行有效的项目估算、产生高质量的系统和按时交付软件产品为目标的软件项目管理策略。公司在早期做过 c/s 结构的教育管理系统，对于目前要开发系统的项目任务有较清楚的认识，并能快速地建立任务之间的相互关系。根据以前项目的开发过程建立的度量基线，我们可以较准确地估算各个任务的工作量，分配人员和其他资源，制定进度时序。

下面就谈谈在这个中小学学校管理信息系统项目的资源管理中如何使用的度量方法改进开发过程的。

在此项目中我们采取了以下主要措施和方法：

一、在软件计划和需求分析阶段，综合考虑利用可复用资源，合理估算项目工作量，准确度量任务权重，成功降低软件生产成本

在本系统的定义阶段，我们把估算项目工作量作为此阶段的主要度量元素，通过综合考虑利用可复用资源，准确核算任务权重，从而尽量精确地估算出项目工作量，将项目的开发周期尽可能限制在可实现的最短时间内，以达到降低软件生产成本的目的。

我们在项目工作量及技术难度的估算上遵循面向对象技术原则。

本系统采用了面向对象的分析、设计等一系列面向对象技术，在系统工作量估算方面根据功能点进行。将每个功能模块逐步分解，直至基本模块为止。由于公司以前完成过基于 c/s 结构的教育管理系统的成功案例，因此有不少供我们参考的数据和资料。对于本系统的第一部分学校办公管理信息系统我们就是借鉴以前基于 c/s 结构的教育管理系统。c/s 结构的教育管理系统是利用不同的客户端版本，达到区分多用户访问权限的目的，但综合所有客户端版本，其在基本功能、数据库结构设计和前台操作，如对数据库进行添加、删除、修改、查询等操作方面都与新项目管理信息系统部分大体相同。正因为如此，有大量的资源可供我们复用，许多基本功能模块仅作少量修改就能够被复用到新系统中。在工作量的估算上也有很好的比照借鉴作用。以前的数据对工作量的估算是一个重要的参考，为项目进度安排提供了依据。

在技术上，我们重点考虑在基于 B/s 结构的系统中多用户访问及权限控制模块的设计和项目第二部分信息服务系统的实现。由于软件工作量的估算首先从软件项目范围抽出软件功能，接着给出为实现每一软件功能所必须执行的一系列软件工程任务，如需求分析、设计、编码和测试等。所以对于上述含有技术难度的任务，我们利用历史数据中对不同复杂性和开发特性的功能模块，分别按最佳地、可能地、悲观的三种情况给出的估计值计算加权平均，得出较为准确的任务权重度量值。

根据以上分析我们制定了一个详细的项目进度表并定义了相应的里程碑，限定了项目要在六个月的时间内全部完成。这比我们最初的预计提前了两个月，大大降低了项目开发成本。

二、在软件设计编码阶段，注重人员安排，确保产品质量

人力资源在考虑各种软件开发资源时，人是最重要的资源。在安排开发活动时必须考虑在开发过程各阶段中对各种人员的需要。在软件项目设计编码阶段，我们把人员调配确定为此阶段的主要度量元素，按照人员自身能力状况合理安排，保证了软件产品的高质量。

本系统在软件需求分析和设计阶段，有两名公司的高级技术人员参与对软件系统进行定义。他们利用自己丰富的实践经验，为系统提出了许多可行的能够高效实现预定任务的方案，同时在技术难度相对大一点的部分由他们从设计方面把关。而在项目实施阶段他们也负责指导相关人员对关键任务的理解，并随时监控程序员在具体编码中是否真正实现了他们的设计思想。这一系列的工作安排，确保了我们的软件产品在关键技术问题上从未出错。尤其是高级技术人员设计的由超级用户实施的权限控制模块，可以为不同的用户设置每个模块的不同访问权限，也可以在权限分配中对权限相同的用户进行批量处理。产品交付使用后，用户对于灵活方便的权限控制模块非常满意。

由于有可复用的资源和高级技术人员的指导，项目开发中大量的编码和调试工作，是由初级技术人员来实现。根据进度表中各个里程碑的划分，管理人员可以方便地度量每个编程人员的实际工作情况，同时按照工作时间安排进行软件测试和验收，从而确保开发的进度和软件的质量。

三、在软件开发全过程，随时掌握项目进行情况，必要时调整进度表

在为项目开发制定的详细进度表中，确切地定义了每一项任务的工作量、开始时间、持续时间和结束时间。我们让开发小组的每一位成员认真了解自己所承担任务的时间表，并根据自己的任务制定相应的工作计划。

我们公司里有一个用来管理员工每日工作情况的日志管理系统。为了使教育管理信息系统项目能够顺利完成开发任务，我们要求参与此项目开发的人员在当日工作结束后都要按照实际工作情况填写它。例如，一个开发人员今天的工作是对某个功能模块进行编程，在日志文档中会包含：开发的模块名称、开发目标、使用的方案、新建或修改的文件名称、遇到的困难、解决的方法以及计划的完成情况等。同时每周要对自己的工作进展做出结论，如：某某任务已完成 80%，能够按照计划时限完成本项任务，并向项目组汇报。

开发小组的每个成员都要对自己做出的结论负责，这样管理人员就可以做到随时了解项目进度，为调整项目计划提供客观基础。项目小组会在每个里程碑处以会议的形式对本阶段的工作进行确认、总结，对本阶段的进展情况做出结论，并决定是否调整下一阶段的进度计划。

本系统在整个项目开发过程中，由于总体工作量估算基本准确，关键任务及人员使用定位合理，使得开发任务按期高质量的完成了。但是在我们的工作中也有失误的地方。系统特点决定了项目需要完成不少网络界面设计，因为公司美工人员不足，一个美工同时兼顾几个项目，造成项目组与美工人员沟通不畅，某些界面设计无法达到要求，险些影响整个项目进度。在管理人员发现情况后及时外聘美工，补足人员调配缺陷才使任务顺畅进行。

通过以上的分析和讨论我们可以知道，要想真正在软件生产中实现工程进度的可控性，就必须在软件项目计划阶段准确确定软件范围、合理调配所需资源，从而获得更接近实际值的软件任务成本和工作量估算。软件工程度量的估算虽然不可能与工程实际进度完全一致，但在不断发现导致合理估算的信息的过程中，是可以逐步达到计划目标的。软件工程的度量随着项目的进展定期更新，从而将软件工程的进度限定在可控范围内，就能使得软件生产任务按照预定计划实现。

论软件过程的度量

摘要:

本文从软件错误数、缺陷数的角度讨论了软件质量度量的方法、作用和意义。

2004 年 2 月我作为某大学网络学院技术部的技术骨干，参加了《网络学院 2004 版远程教育平台》的开发，在该项目中我担任项目负责人。该平台涉及分布全国的重多学生、教学站、教学中心，大量教学资源以及教师、教务管理等多方面，该平台的建设直接关系到学院今后的高效管理、持续发展，因此对我们的软件提出了很高的质量要求。

在该项目中，我启用了软件质量度量，主要计算软件开发各阶段的错误数、缺陷数、错误清除率，本文重点介绍了需求分析阶段我们按功能点计算错误数的方法，以及计算结果对我们系统的影响。

对于软件质量度量我们的经验还十分缺乏，本文作者也特别希望大家提出更好的建议帮助我们改善度量方法。

正文:

本人所在单位为某综合性大学网络学院技术部，从 2001 年建院开始我们技术部陆续为学院自主开发或外购了多种软硬件系统，这些系统的开发是不同时间、由不同开发人员用不同开发工具、使用不同数据库运行在不同平台上的，比如部分是：asp/Sql server2000/Windows，部分是 Php/Mysql/Linux 等，子系统之间数据割裂、交叉、重复现象严重，代码经过几年的修改再修改已经变的很难修改；服务器的负荷也已经很难满足要求，网络高峰期特别是考试前后很难访问。因此，学院决定于 2004 年 2 月立项开发《网络学院 2004 版远程教育平台》，在该项目中，由我担任项目负责人。

该远程教育平台分 2 期工程，采用增量模式开发。一期工程主要包括：学生管理、教务管理、教务管理、教学资源管理、网络题库考评、虚拟社区、毕业论文管理等，其中教学资源管理中的流媒体课件系统、虚拟社区中的电子白板利用原有外购产品，此次只进行升级。二期工程主要包括 OLAP 联机分析系统，对学生地区分布、年龄分布、专业分布、成绩分布，教学资源质量、教师工作质量等进行分析和评估；建立自适应学习平台、研究性学习平台等。

本远程教育平台是一个庞大的体系，是学院高效管理、持续发展的基石。平台涉及遍布全国各地的 3 万多学生、15 个省自治区级的教学站、教学站下设的 200 多个教学中心；教学资源涉及 15 个专业近 400 门课程，学校近 400 名教师以及我们网络学院各管理部门（比如财务部、学生管理部、教学资源管理部、教师管理部等）。本平台的开发，学院领导给予了高度重视，提出该平台至少应满足今后四到五年的规模发展需要；新的教学技术教学理念的实施应能与本系统无缝集成。

作为本项目的负责人，我深感责任重大。反思我们前几年软件开发管理力度不大造成一定程度上开发的混乱，我参观学习了本市两家取得 CMM4 级认证的软件开发，我决定在本项目的开发中启用软件质量度量，主要度量软件各阶段的错误率、缺陷率，以及上一阶段对下一阶段缺陷的转播、缺陷的放大。鉴于我们进行软件度量的经验缺乏，我们聘请了一名 CMM4 级软件公司的资深项目管理专家做咨询顾问，请他每周至少一次来学院指导工作。

我们采取的基本度量方案是：度量软件的错误数、缺陷数、计算错误清除率。具体实施中，我设计了一个错误汇总表和几个分级明细表。在汇总表中，我将纵坐标设计为软件开发各阶段：计划阶段、需求阶段、设计阶段、代码阶段和测试阶段、使用维护阶段；横坐标设计为：本阶段发现的错误数、下阶段发现的错误数(此数据作为本阶段的缺陷数)，错误清除率。汇总表主要从宏观的角度看我们软件的整体质量。汇总表的数据来自多个明细表。

我在这里重点介绍我们需求分析阶段错误数的统计，以及通过错误数统计、分析对我们软件开发的促进和提高。我们都知道软件的质量首先是满足各类需求，不满足需求的软件就谈不上质量。因此我们对需求分析给予了高度重视。我设计了如下需求分析阶段错误数统计表：纵坐标是各功能点以及子功能点，功能点和子功能点分级展开；横坐标由各功能点简单说明（该项链接到需求规格说明书）、业务需求、功能需求、性能需求、安全需求、数据需求、错误描述、评估人、评估时间等项组成，其中业务需求、功能需求、性能需求、安全需求、数据需求均由两栏组成，一栏是错误数、一栏是缺陷数，缺陷数由后续开发阶段填写。

这个需求错误评估表做出来之后，我们技术部做了大量工作来填充这个表。首先是功能点的划分。我们采用 UML 统一建模工具，用 ROSE 2003 建模系统各部分。我们将该系统的参与者分为：网络学院院长、院各管理部门负责人及管理人员、网站数据管理员、教学站以及教学中心负责人及管理人员、教师、学生等，根据不同参与者确定了大量用例，并对每个用例编写了：用例目的、前置条件、后置条件、限制条件、基本流程、替换流程等；在这些用例的基础上，我们划分出了功能点以及子功能点。

建立了各功能点之后，最重要的是评估。学院请来了本系统各类有代表性的用户，比如学生人数最少的、最多的、离学院距离最近的、最远的教学站以及教学中心负责人，进原“网络教室”、原论坛最活跃的、最不活跃的、成绩最好的、最不好的学生，最热门专业、最冷门专业以及公共课的教师等等组成的用户代表团，还请来了安全技术专家、服务器方案专家、教育管理专家、教育技术专家等组成的专家团，与我们学院各部门管理人员一起组成的评审团对我们的需求进行评估打分。在评估之前，我们技术部讲解了我们的项目计划、需求说明，有的需求用以前系统的功能演示，有的需求用我们设计的界面原型演示，使每个参与人员都充分了解本系统。做好这些准备工作之后，我们要求每个参与人员认真评估需求，特别是与自己相关的那部分，对于需求错误一定要 给予详细的错误描述；对于希望补充的功能，可以另外说明。

技术部通过 2 周时间的分析、整理这些需求错误统计表，得到了很多有价值的信息。既有比较重要的功能遗漏：比如考试场次安排。以前我们的考试场次安排通过手工进行，只考虑一些粗略的因素，比如考试人数、课程数等。但实际情况远比这个复杂，这次的监考教师、教学站负责人都提出了这个要求，由软件系统综合考虑实考人数、补考人数、座位数、考场数、监考教师数、试卷数等安排考试场次。还有一些错误是我们技术部预先想不到的，比如：通过边远地区学生和教学站的调查，发现这些地区很多电脑配置很低，运行的是 IE4.0 甚至是 4.0 以下，上网是电话拨号；他们看不到我们技术人员用脚本做出来的漂亮页面，看到的是很混乱的页面；流媒体课件无法点播，其它图片下载也非常的慢。而且这部分学生大约占总数的 10%。这些错误对我们技术部分析人员触动很大，我们的系统必须给予正确的应对，比如我们准备今后在制作流媒体课件的同时，制作部分教学光盘分发到边远地区，我们的界面设计人员一定要在多个浏览器下测试效果。

以上只对软件需求阶段错误数统计进行了介绍。事实上，我们通过对整个项目各个阶段错误数以及缺陷数的统计分析，大大改进了我们的软件质量；最明显是我们的新系统 2005 年 2 月投入运行后，各方面的反应都很好。

软件项目度量对我们来说，还明显的经验不足。我们的软件质量度量中，还有一些重要的影响因素没有考虑和计算。比如错误的权重：修改错误的时间、人力、资源；修改错误时引起的其它错误；上阶段未发现错误对下阶段开发的影响没有量化等等。希望今后的软件开发过程中我们改进这些问题，并引入更先进的度量方法。

论软件的性能优化设计

摘要:

本人 2004 年有幸参加了中国石油集团的高性能数控测井系统项目的开发研制工作。该系统是在当前测井成套测井装备的基础上, 为了满足高精度, 高性能, 高效率的要求开发的测井系统。该系统由井下成套仪器, 测井遥测系统, 测井地面系统, 测井软件系统, 测井解释评价系统等子系统组成。本人在其中主要是负责测井软件系统的分析、设计以及部分开发任务。作为整个系统控制核心的测井软件如何才能保证有整个系统的高性能和高可靠性呢? 本文从系统优化、程序设计优化两个方面来详细讨论如何提高整个测井软件系统的性能。其中系统优化主要是通过调节软件运行环境来优化软件性能, 程序设计优化主要从程序架构设计、语法、 内存管理、输入输出等方面来讨论如何采取措施提高软件的性能。

正文:

随着当前石油测井技术的发展, 为了能更快, 更好的得到储层地层信息, 解决目前国内测井系统不统一, 测井精度不高, 效率低下的缺点, 2004 年 1 月中国石油集团公司科技局成立了高性能数控测井系统项目, 目的是为国内测井行业提供一个从井下到地面以及解释评价的整套测井系统。系统的设计目标是一次测井, 取得所有合格资料, 并且能保证 60 井次的免维修率。整个系统由井下成套仪器, 测井遥测系统, 测井地面系统, 测井软件系统, 测井解释评价系统等子系统组成。我主要是负责测井软件系统的分析, 设计和部分开发工作。

整个测井软件系统完成三个主要任务: 测井数据的采集、 测井数据的工程值计算、测井过程的监控。对应于这三个任务, 整个系统分为三个系统层: 数据驱动层(简称前端), 数据处理层(主控)和表象管理层(后端)。数据驱动层的主要目的是封装地面系统和井下遥测系统, 为数据的上传和命令下达提供统一的接口。数据处理层的主要目的是按每种仪器的刻度算法计算测井原始数据。表象管理层则负责测井数据的表象处理, 主要是曲线和图像的显示与绘图。系统前端运行在 Vxworks 系统上, 主控程序和后端显示程序运行在 Window2000 系统中。前端主要是采用 Tornado2.0 开发, 主控程序和后端显示程序采用 VisualC++2003 开发。

由于测井软件系统负责着整个井下仪器的数据采集、仪器状态控制和测井资料获取。所以对测井软件系统的性能要求是非常高的。这主要表现在以下方面:

采集数据的完整性和正确性要求。由于目前井下仪器主要是通过 100K 的 CTS 遥测传输数据。要求每 80 毫秒采集一个完整的数据帧, 由于仪器算法的特殊性, 要求不能丢掉一帧, 也不能采集错误的数据。这就要求数据驱动层有非常好的实时性和高可靠性。

数据显示和打印的实时性要求。由于井下地质情况复杂, 为了保证测井安全以及测井数据质量, 需要把测井的数据用曲线和图像的形式实时的显示和打印出来, 以便于监控测井过程中井下的各种情况。

在软件分析设计和开发中, 我们主要是系统优化和程序设计优化两个方面采取措施来提高软件的性能并取得了比较好的效果。下面我主要从这两方面进行讨论:

一、系统优化

系统优化主要是从测井软件运行的系统环境角度方面采取措施来提高软件的性能。由于 Window2000 操作系统的分时多任务系统, 不能保证在 80ms 时间间隔内测井系统所要求的数据的完整性和正确性, 所以我们采用 VxWorks 实时操作系统做为前端操作系统, 运行数据采集程序, 保证测井数据完整和正确。再通过网络连接到主控程序处理数据。为了保证系统网络数据传输的性能, 采用千兆网, 连接主控和前端。同时为了提高数据 IO 的响应速度, 以及数据的可靠, 采用 SCSI 接口的磁盘阵列, 提高数据的冗余度来保证系统数据的可靠性。同时, 主机系统采用 Intel 公司 P4EE 结构的多线程处理器, 以及 512M 的 DDR 内存, 来满足数据吞吐的要求。

二、程序设计优化

程序设计优化主要是程序设计优化主要从程序架构设计, 语法算法和编译, 内存管理, 输入输出, 软件测试等方面采取措施提高软件的性能。

1、程序架构设计

由于优化性能并不是个局部的过程, 整个程序的性能提高必须在程序设计的开始阶段就给予考虑和规划, 以设计良好的软件架构保证系统的良好性能。本系统采用三层系统结构, 各层之间采用网络连接, 实现松散的耦合。即使后端显示程序出现错误, 也可以保证测井数据的完整和正确。同时引入组件对象的开发思想, 把每个仪器的算法及控制都做成组件, 并进行单独的测试, 从而保证组件的正确性。采用这种办法不仅可以保证错误不会在系统的扩散, 方便系统的调试。同时也可以利用以前成熟的仪器算法和控制代码, 不会引入新的风险, 提高系统的成熟度和可靠性。

2、语法算法优化和编译优化

在代码开发阶段, 可以通过 c++语法特点来优化系统, 实现更好的性能。例如程序中大量的简单函数, 为了提高运行效率, 一般都采用内联函数的方式实现。在循环语句中, 对各种情况进行度量主要通过把临时变量提出到循环体外, 以及设置 break 和 continue 语言减少循环次数。在对大内存对象的参数传递上, 例如仪器服务表结构, 基本都是利用引用和指针实现, 减少系统在堆栈上的内存分配。在仪器的刻度计算中经常要用到各种各样的算法, 采用在保证测井精度的原则下, 根据测井原理优化算法, 尽量把一些递归和非线性的算法, 转换为非递归的简单运算。同时为了充分发挥 Intel CPU 的功能, 我们选择 Intel 的 C++编译器, 做为系统的编译器。Intel 编译器对 C++错误检查比较严格, 这也保证了程序运行时的错误减少。在开发库的选择上, 我们也采用了精练的 ATL 库开发仪器组件, 使用 WTL 库开发用户的界面, 同时使用 STL 库进行数据结构的实现。WTL 库的使用使得我们可以抛弃传统的 MFC 动态库, 减少程序的内存占用, 也可以减少 MFC 常见的内存泄漏。STL 是经过考验的 C++标准库, 实现的代码精练高效, 管理方便。经过我们测试, 同样的数据结构如队列、链表等, STL 的实现比 MFC 的实现, STL 的速度大约快 10%。

3、内存管理的优化

内存管理对系统也很重要, 在测井后端显示程序中, 由于要分配大量的小对象, 例如曲线和矩形等, 通过分析性能测试数据, 发现大量的小内存释放和分配, 成为后端显示程序的速度瓶颈, 所以我们采用自己的内存管理机制来减少系统的调用。首先根据测试数据度量, 分配一大块内存给应用程序, 然后通过管理这块内存, 来满足程序对小内存块的请求和释放。如果内存不足的话, 可以另外分配大块内存进行扩充。通过采用自己的内存管理机制后, 再运行测试程序, 发现程序的运行速度几乎提高了两倍。

4、输入输出优化

输入输出在系统的应用非常的多, 特别是本系统采用三层网络的结构, 其中的网络输入输出更加的重要。为了系统取的最好的性能, 我们对前端和主控之间, 主控和后端显示之间的网络通讯采用的都是多线程异步通讯的方式。与以前采用同步的方式对比, 系统的速度提高了一倍, 而且网络的故障也减少了很多, 基本没有发生网络通讯中断, 和数据掉帧的现象。在本系统中还有一个比较关键的技术难点, 就是在如何在显示测井曲线和图像时同步连续的打印出显示的曲线。由于程序的实时性要求, 程序在网络通讯和数据计算上已经消耗了大部分的资源。在这方面我们主要采取的技术手段的同步缓存显示数据, 然后在安排一个单独的线程专门负责测井曲线的打印。经过测试, 打印和显示之间只有一米左右的延迟。在 200:1 的通常测井显示比例下, 这点延迟是在允许范围内的。

5、软件测试

测试并不能直接提高软件的性能, 但是测试是提高软件性能的有效手段。一个好的测试工具更能提高工作的效率和质量。在本系统的开发过程中, 我们主要是使用 Rational 公司的 Purityplus 工具进行测试。特别是用其中的 Quantify 工具进行系统性能测试, 可以精确到代码行。通过它可以快速发现系统的性能瓶颈在哪儿, 在哪儿耗时特别多, 以及整个程序的运行时间。大部份的优化措施都是在测试过程中发现问题, 然后在进行具体的优化。

在采用以上各种优化措施后, 软件系统的整体运行效率提高了 50%左右, 今年上半年, 整个系统完成了系统联调。在下井实验过程中, 在连接井下仪器、地面系统、和测井软件系统的基础上, 软件运行可靠, 保证了 24 小时不断电的情况下, 数据帧掉帧率为 0, 误码率为 0 的好效果, 得到用户的认可。

在整个系统的开发过程中, 特别是对软件的性能优化, 使我认识到软件开发中采用不同的设计和措施对软件的性能影响是很大的。好的测试工具对性能的提升有很大的帮助。在目前对软件性能要求越来越严格的要求下, 只有在重视测试方法和测试工具使用的情况下, 采用恰当的优化方法才能提高系统的性能。

论软件设计模式的应用

摘要:

本人 2004 年有幸参加了中国石油集团的高性能数控测井系统项目的开发研制工作。该系统是在当前测井成套测井装备的基础上, 为了满足高精度, 高性能, 高效率的要求开发的测井系统。该系统由井下成套仪器, 测井遥测系统, 测井地面系统, 测井软件系统, 测井解释评价系统等子系统组成。本人在其中主要是负责测井软件系统的分析、设计以及部分开发任务。设计模式是前人设计面向对象软件的经验总结和, 在软件设计中灵活的使用设计模式可以极大的提高系统的稳定性, 可扩展性, 以及良好的可维护性。本文描述了在测井软件系统开发过程中, 如何分析和发现相关模式, 以及如何选择和应用设计模式, 特别是介绍了 MVC 模式在软件框架和相关系统模块中的应用和使用效果。在文章的最后, 讨论了在实际项目开发中, 设计模式应用的有关想法和教训。

正文:

随着当前石油测井技术的发展, 为了能更快, 更好的得到储层地层信息, 解决目前国内测井系统不统一, 测井精度不高, 效率低下的缺点, 2004 年 1 月中国石油集团公司科技局成立了高性能数控测井系统项目, 目的是为国内测井行业提供一个从井下到地面以及解释评价的整套测井系统。系统的设计目标是一次测井, 取得所有合格资料, 并且能保证 60 井次的免维修率。整个系统由井下成套仪器, 测井遥测系统, 测井地面系统, 测井软件系统, 测井解释评价系统等子系统组成。我主要是负责测井软件系统的分析, 设计和部分开发工作。整个测井软件系统完成三个主要任务: 测井数据的采集、测井数据的工程值计算、测井过程的监控。测井数据采集主要是采集井下仪器通过测井遥测系统传输的测井数据, 并保证数据的完整性, 正确性。测井数据工程值计算主要是把采集的数据根据不同仪器刻度计算方法进行工程值的计算。测井过程监控主要是把计算的测井数据用曲线和图像的方式实时的显示在屏幕和打印成图, 由测井操作员进行实时监控。

设计模式是前人设计面向对象软件的经验总结和, 在软件设计中灵活的使用设计模式可以极大的提高系统的稳定性, 可扩展性, 以及良好的可维护性。在测井软件系统框架进行分析和设计时, 考虑如何提高系统的稳定性、可扩展性和可维护性时, 我们采用了 MVC 设计模式。

MVC 模式构架包括三个部分: 模型 (Model)、视图 (View)、控制 (Control)。模型主要是对系统的数据和逻辑运算的封装。它独立与系统的界面和 I/O。视图把表示模型的数据和逻辑关系用特定的形式展示给用户。控制处理用户和软件之间的交互操作, 当模型的数据有所变化时, 控制负责通知视图做出相应的更新。模型、视图、控制的相互分离有利于模块之间内聚性的提高, 耦合更加松散。一个模型可以对应多个视图, 由控制来传播模型的变化从而更新视图。

MVC 模式如何在测井软件系统实现, 我们主要是从如下四个方面进行:

一、分析系统功能, 分离功能模型

首先根据系统的主要任务进行系统的模块分解。根据测井软件系统数据采集、数据转换和测井监控三个主要任务, 把系统分为三个模块对应于 MVC 模式的三个部分。其中模型 (Model) 对应于数据的采集和工程值的计算。测井视图 (View) 对应于测井监控功能。测井模型所要实现的功能包括: 测井数据的采集、数据的刻度计算、数据的存储、数据的操作。测井数据的采集负责硬件平台的初始化, 下井仪器的初始化, 井下仪器数据的中断相应, 数据帧的采集, 数据帧的重组等。数据的刻度计算主要是根据不同的仪器实现数据的刻度计算, 包括刻度系数表的获取、刻度计算、深度延迟计算等。数据存储主要是原始数据的存储和测井数据的存储。这里我们采用的是测井公用的 XTF 格式做为数据存储格式。数据的操作是视图和模型之间数据交互的接口。它主要是提供数据输入和输出功能。

二、视图的设计与实现

视图主要是提供测井数据的图形显示。通过调用模型中的数据操作方法, 提取测井数据, 根据不同的测井数据提供曲线、波列、图像等多种表现形式。在本系统的实现中, 为了提高数据采集的稳定性和程序的健壮性, 采用进程间通讯的方式。就是说视图的实现本身一个独立的程序。它与模型之间的通过 TCP/IP 网络进行通讯。视图主要包括数据源、数据表象对象、绘图打印模块等部分组成。数据源负责得到模型 (Model) 的数据, 然后把数据分配给每个数据表象对象。数据表象对象是个有层次的类家族, 其基类是绘图类 (CDrawObj), 所有的数据表象包括道 (CDrawTrack)、曲线 (CDrawCurve)、波列 (CDrawWave)、图像 (CDrawImage), 数值对象 (CDrawData) 等都是从其派生的。最后有绘图打印模块提供管理, 负责视图的区域更新, 数据表象的绘制和打印等功能。

三、控制的设计与实现

控制主要功能是提供用户的输入输出反馈, 同时监控模型的数据变化, 通知视图进行更新。由于控制和视图的耦合非常的紧密, 在架构实现中, 控制和视图是在一个应用程序中实现的。控制主要分为井下仪器控制和视图控制两个部分。其中井下仪器控制主要是由操作人员根据视图中的曲线和图像信息, 对仪器发出的状态控制命令, 以保证测井过程中数据和仪器的安全。视图控制则是操作人员对视图显示参数的调整, 包括鼠标的响应和键盘的响应以及用户对测井原始图的特殊要求如道大小, 曲线位置的摆放, 颜色的调整等。

四、使用可动态添加算法模型

由于每次测井作业中下井仪器串的仪器种类和仪器的数量都是变化的, 为了能更好的抽象出实际的测井模型, 提高系统的灵活性, 在模型中数据刻度计算部分, 我们采用的动态添加的方式。我们把不同测井仪器的刻度算法封装到动态连接库, 然后根据测井作业的不同, 调用用不同的仪器动态库中的刻度算法。由于视图和控制与模型之间的松耦合, 当用户添加算法模块, 视图与控制基本不要修改。

在采用 MVC 模式的软件框架后, 整个系统分为两个部分, 数据采集管理器和数据实时浏览器。数据采集管理器对应于模型 (Model) 的实现, 数据实时浏览器对应于视图 (View) 和控制 (Control) 的实现。我们采用的是 Visual C++ .net 基于 Window2000 平台来进行系统开发。采用 MVC 模式给我们带来了如下好处:

1、由于模型 (Model) 与视图 (View) 和控制 (Control) 之间的松耦合, 使得我们非常容易就实现了一个模型运行同时建立多个视图。这在调试仪器时非常有用, 当硬件人员调试仪器时直接连接网线就可以一边看仪器一边看数据。不再需要象以前必须到地面系统控制室查看数据了。

2、适合多硬件平台的跨接。由于不同的硬件平台上采集数据的方式都不同, 有的系统采用的是 PCI 总线, 有的是 USB 接入, 有的是 ISA 卡接入。由于模型 (Model) 和视图 (View) 的松耦合, 当要移植到不同的硬件平台上是我们只有修改相应的模型 (Model), 有可以实现对不同硬件平台的支持。

3、良好的可维护性和扩展性。由于采用 MVC 模式, 系统模块功能划分明确, 代码实现也相对容易。代码的错误不会在系统中扩散, 同时由于可以动态添加仪器算法模块, 当用户添加新仪器时, 不需要更改系统程序, 只有添加仪器动态库 DLL 就可以了。

在整个系统的开发中, 我们还应用了一些别的模式, 有些模式是在进行系统设计时, 就考虑到而特意实现的, 有些模式是在采用别的方法实现后, 效果不太理想, 在代码重构时引进的。在应用设计模式进行系统设计和开发后, 整个系统各个模块之间逻辑变的相对独立, 耦合也很松散, 结构的扩展性良好。而且使得代码的重用的程度变好, 减少了错误的发生和错误在代码中的扩散。但是在实际应用模式的过程中, 我还发现模式应用的经验越丰富, 模式应用的就越好。有时在采用何种模式时, 有几种模式方案可以采用, 但是具体采用那个模式就需要不断的尝试, 看看模式是否满足实际的需要。特别要注意的是不能为了设计模式进行设计, 也就是过分设计的问题。这样会导致设计过于复杂, 偏离程序设计简约够用的基本原则。

目前设计模式在软件开发中的应用正引起广大开发人员的注意, 各大软件开发商也在软件开发工具中提供了有关设计模式的自动应用的工具, 相信设计模式会越来越多应用于软件的设计和开发中。

论软件的可维护性设计

摘要:

随着软件大型化、复杂化的发展,软件维护所耗费的资源越来越多,软件可维护性设计日益得到重视。我单位近几年开发综合业务 ATM 交换机,用户对交换机的可维护性要求很高。我参加了该项目并负责软件的维护性设计工作。根据当前工作中在维护性设计中的不足。通过在各个软件开发阶段注重软件可维护性的应用,规范文档,使用 CASE 工具管理软件版本和成立软件可维护性设计小组等方面,为软件的可维护性设计提供了帮助,并最终开发出具有良好可维护性的交换机软件。但是由于初次实施这方面的工作,大家思想上认识不够,许多操作不习惯,并且单位里不具备专用的测试软件和其它 CASE 工具,在一定程度上制约了软件可维护性的实施。

正文:

经过一系列的需求分析、设计、编码和测试之后,软件正式交付用户使用。至此,软件变进入维护期。软件维护的工作量特别大,随着时间的推移,软件维护对开发商带来的成本压力也越来越大。许多软件开发商要把 70%的工作量用在维护已有的软件上,平均来说,大型软件的维护成本是开发成本的 4 倍左右。因此,在开发软件时,就应该考虑到可维护性问题,进行软件的可维护性设计。

2002 年底,我单位开始为某集团开发综合业务 ATM 交换机。该交换机支持多种业务应用,包括话音、IP 数据、图像和视频等;用户可通过维护台或网管对交换机进行配置和管理;由于特殊的应用,用户对该交换机提出了很高的要求,并且提出要求产品交付使用之后,我单位要有很好的服务支持,鉴于将来要大批量生产交换机,软件的可维护性设计被提上日程。我有幸参加了该项目,并负责软件的维护性设计工作。

在以前的课题中,也曾提到过要进行软件的可维护性设计,但在真正实施过程中,还存在诸多问题,主要表现在:

(1) 在软件开发过程中对可维护性设计考虑欠缺,导致软件设计从根本就忽视了可维护性的重要性。

(2) 软件设计文档不规范,内容不一致。在维护阶段出现问题,根据文档不易得到有帮助的信息,难以定位错误的类型和根源。

在本交换机软件的设计过程中,我们通过注重软件可维护性的开发过程,规范文档,使用 CASE 工具管理软件版本和成立软件可维护性设计小组等方面进行软件的可维护性设计,最终开发出具有良好可维护性的交换机软件。

一、注重可维护性的开发过程

在整个交换机软件的开发过程中,从软件易于理解、易于测试、易于修改的角度出发,提高软件的可维护性。

在需求分析阶段,和用户进行充分的交流和协商,对将来要改进的和可能要修改的部分进行明确。由于该交换机所涉及的业务种类广泛,并且综合了话音、IP 和网管等多种技术,任何一种技术实现的功能不完善或者扩展性不好,都不会让用户满意。但是,另一方面,又考虑到用户需求和功能需求并非容易获取,所以通过和用户定期交流,举办各种形式的讨论等方式尽可能了解当前的需求和以后需要扩展的需求信息,由专人整理记录这些信息,作为以后的跟踪内容。即使在其它设计阶段对需求的临时变动,也要在这个记录中体现。

在设计阶段,交换机软件被划分为不同模块进行设计,并遵循“高内聚、低耦合”的设计原则;这些工作便于将来软件维护工作的进行。同时也已考虑到,对可能要扩展的地方,预留出充足接口。在一些模块中,如网管模块中,根据功能,尽可能使用面向对象的设计方法,以便维护时的修改和升级。

在编码阶段,我和小组成员制定了统一的编码规范,经过半天的培训,强化编码人员对注释的使用,并强调要保证注释的质量,对有可能出现误解的地方,注释的要详细。并且,每个文件都要注明编写者,生产日期和版本号。

在测试阶段,测试组成员已经负责进行测试,我们小组这时的工作是根据测试报告,对照测试大纲和用例设计,对当前的测试进行总结,比如,何种测试用例发现何种错误,最常见的错误,如何从测试结果判断是哪种错误,该错误所在的模块是什么。在相关人员修改错误时,记录排错时的思路 and 过程。特别是,根据这些总结,我们编写了“ATM 交换机软件故障解析”,这篇总结在后来的维护阶段被证明是最受欢迎的文档之一。

在维护阶段,制定严格的管理要求。每一次维护工作之后,都要按照配置关联,同步更新维护有关的系统文档和用户文档,包括维护需求、源代码、注释、设计文档、测试文档和用户使用手册等,保证系统的一致性。维护中所进行的修改要专人记录,生成“ATM 交换机维护更新”文档,做为内部文件存档。同时把一些内容扩充到“ATM 交换机软件故障解析”中。在用户使用时,做好用户的培训工作,初期由专人和用户一起操作交换机,直到能熟练操作,以免用户使用交换机时产生不满。

二、规范文档

交换机交付用户使用之后,除了在培训时所了解的内容之外,为了让用户对交换机软件能够更好的理解和使用,向用户提供了多种随机文档,包括功能说明,安装文档,用户使用手册,参考手册,管理员指南等。在文档的编写过程之前,我们编写了“ATM 交换机软件文档编写规范”,对文档格式和一些必要内容进行了规范,保证各文档的风格一致,内容一致。对于一些用户使用中容易出错的地方,比如配置某种功能时,在用户使用手册示例说明。在具体编写文档时,根据设计人员的反馈信息,也及时调整了文档编写规范。

在设计开发过程中,对某个问题进行修改,或者功能增删,要充分考虑到问题所涉及的不同文档,保证前后文档在该问题的一致性。对于所修改的部分,要填写“更改单”,需要写明更改人,更改理由,更改所影响的程序和文档,更改日期,批准人。采用 CASE 工具在这一方面也起到了事半功倍的作用。

三、使用 CASE 工具管理软件版本

在软件的设计编码过程中,尤其是在调试阶段,会不断的生成新的程序版本。为了有效的管理版本问题,采用 Ration 公司的 ClearCase 工具,由专人负责进行管理,从而保证软件版本的一致性。

四、成立软件可维护性设计小组

为了有效的对软件可维护性设计进行管理,成立了软件可维护性设计小组。我担任小组组长,明确了维护性设计的工作内容和各人的责任,针对不同的模块,又确定四个责任人。在运作过程中,组长对软件开发阶段所需进行的工作进行协调,各负责人对维护性设计所涉及的变动控制进行维护。

因为交换机软件的各个模块开发时间有穿插,因此,对开发过程中出现的一些问题,包括技术方面和管理方面的问题,我们都及时进行了记录,对后面开发的软件模块进行指导,避免了同样问题的再次发生。现在这份文档已经成为单位新课题启动时的“必修”文档。

按照上面的思路,经过两年多的工作,我们已完成了交换机软件的开发,新的软件运行良好,交付用户后,用户很满意,受到了业务部门和技术部门高层的赞许。尤其是我们所总结的“ATM 交换机软件故障解析”和“ATM 交换机软件文档编写规范”等文档,对单位其它课题也起到了很好的指导和规范作用。并且,在提高软件可维护性的同时,也提高了软件产品的质量,我自己的开发管理水平也得到了很大的提升。单位的高层领导对我们制定的规范和做法也表示认可,正式在其它课题中推广。

然而,由于初次在整个软件开发过程中进行可维护性设计,还有许多要改进的地方。许多情况下,现有的可维护性设计措施理论性太强,具体实施时可参考的依据少,比如我们测试小组如何更好的与各课题开发组间进行协调工作,感觉还是有不尽人意的地方。在测试阶段,单位条件所限,没有采用专业的测试软件进行测试,主要靠人工根据经验生成测试用例,测试力度不够,隐含的问题较多,这也不利于今后的维护工作。

软件可维护性设计是一个长期的工作过程,我希望今后能够不断的充实自己在这方面的知识,提供整体能力,为单位软件可维护性设计方面提供新的、好的思路。

论行业应用软件系统的开发规划

摘要:

本文通过地震基本参数快速获取与发布系统论述了行业应用软件系统的开发规划。本人去年有幸参加了《地震基本参数快速获取与发布系统的研制》项目的开发，该系统按功能被分为三个子系统：地震实时波形数据监控子系统；地震事件处理子系统；数据发布子系统。该系统具有地震行业的专业背景，需要地震监测中心（用户单位）、地震监测专家和项目开发人员间良好的分工协作；与常规的通用应用软件开发过程相比，具有明显的差异，主要表现在系统的分析、设计、实现、测试和维护均带有明显的行业特点。行业应用软件的成功开发必须遵循行业规范，正确使用专业术语，及时与专家和用户进行交流。

本人在项目中主要负责需求分析和软件系统设计，在实践中我发现，行业应用软件系统的成功开发是从其规划到诞生起就应从行业中汲取各种营养，才能长出行业特征，具有行业味道，满足行业应用的需求。

正文:

本人去年参加了《地震基本参数快速获取与发布系统的研制》项目的开发。在项目中担任组长，负责项目的规划、分析、设计。由于防震减灾是公共安全的重要内容之一，该项目主要针对大地震的震前预警和震后消息的快速发布，是防震减灾工作的关键业务系统，是地震应急工作的支撑技术系统之一，对公共安全具有重要意义。项目范围包括：地震数据采集、传输、存储、处理、发布；该系统是一个硬、软件相结合的信息系统；硬件包括：拾震器（地震波形采集器）、数据传输设备（基带 modem 和多路复用器）、多用户串口卡、服务器、手机。软件可以按功能分为三个子系统：地震实时波形数据监控子系统；地震事件处理子系统；数据发布子系统。

由于地震行业的专业性强，由开发人员现学领域知识是不现实的，这就需要用户和行业专家的协助。

聘请行业专家做项目组的顾问是应用软件开发过程中较好的策略。行业专家主要是作为领域知识源，提供技术文件和已有应用。在地震系统的这个项目中，我们聘请了地震监测系统的资深专家作为项目的顾问，获取到《地震数据分类和共享指南》等技术文件和部分以前编写的源代码，该源代码是多个独立运行的、凌乱的、较小的系统，经过整理后可以用到新系统中，这为领域模型和设计模型的建立打下了良好的基础。

用户是应用软件目前和未来的需求提供者，其对项目成败的重用性是显而易见的。OOA 过程是从对系统将被使用的方式的理解开始的。在需求诱导时需要从用户的观点对系统建模，需要项目组和用户共同确定使用场景和定义系统的功能和运行需求；需要用户提供清楚而无二义性的终端用户和系统如何相互交互的描述；需要用户提供确认测试的基础。在地震项目中，我们与用户充分合作，请用户给我们讲解地震常识和本项目涉及的专用仪器的功能，将用户作为项目组成员，作为每个阶段的技术评审的主要成员。

开发者是应用软件系统的建设者，需要从用户方进行需求诱导，建立领域分析模型；需要将分析模型变换为设计模型，建立软件的构造蓝图；需要完成编码和测试。由于应用软件具有行业背景，在建立分析模型是需要从各种渠道了解行业知识；在建立设计模型是需要使用逆向工程技术从现有的系统中得到关键的算法。在进行测试时需要使用用户提供的具有行业特点的真实的测试数据。开发人员需要随时在涉及到行业知识时向用户或行业专家请教，必须克服在开发一般应用软件时的问题讨论在相对封闭的范围内进行的特点。

与常规的通用应用软件开发进行比较，专业背景的应用软件开发过程及各个环节的主要差异有：

（1）在计划阶段，具有行业背景的应用软件系统与普通应用软件的目标与动机一般是不同的，普通应用软件以给本公司或部门带来经济上的收益为目的，而行业应用软件以更好的完成某项政府职能为目的。地震行业的应用系统开发目的是为了有效的减少地震灾害造成的损失。系统方案的制定必须符合行业的特征。地震具有突发性、不可抗拒性、影响面广、广大群众积极参与、数据量非常大、算法复杂等特征，在系统方案制定时必须针对这些特点进行。在地震行业的应用系统中，从拾震器（地震波形采集器）、数据传输设备（基带 modem 和多路复用器）、多用户串口卡、服务器、手机到通信信道均实现冗余备份，当任一环节出现故障时，均能由热备件自动接管任务，应用软件必须支持这种功能。这就增加了应用软件开发的复杂性。

（2）在需求分析阶段，行业背景的应用软件系统与普通应用软件相比一般需求比较明确、相对固定和有章可寻。必须获得行业的各种技术规范、数据共享资料及现行系统的运行数据。在此基础上划出应用软件的作用范围和与周围环境进行数据交换的接口，在类的划分时需要从行业术语中选择类的名称，类的属性需要在专业知识中有明确的定义，类的方法也需要使用行业内的主流算法；在地震行业的应用系统开发中，分析模型的建立需要从行业中得到基本素材，从行业知识和管理中得到优化，产生出行业专家认可的类-责任-协作者模型和对象-关系模型。

（3）在设计阶段，行业背景的应用软件系统与普通应用软件相比，从用户界面设计、数据管理、任务管理到控制机制均要表现出行业特征。我们在开发地震应用软件时，邀请有专业知识用户参加到项目组实施的关键环节中，将设计模型在地震行业进行各种地震知识的解释。确保设计模型与领域模型保持一致。在描述每个子系统并将其分配处理器或任务时，子系统设计需要由行业中完成特定的功能的“层次”来确定；地震实时波形数据监控子系统需要完成各监控点的数据采集、地震事件识别和生成地震事件数据地震；地震事件处理子系统需要完成大数据量的传递、进行各种复杂计算和生产地震的各类基本参数；数据发布子系统需要完成在有新震情时第一时间以 web 或手机短信的方式对相关用户群进行数据广播；各子系统间通过分布式数据库完成数据复制和保证数据一致性。数据和任务管理的设计策略需要参考与《地震数据分类和共享指南》相类似的行业标准；用户界面设计也要根据地震数据的特点和物理意义来展示数据；为系统设计具有行业特点的控制机制，在地震应用软件中，我们采用地震触发作为系统的控制机制；通过创建每个操作的过程表示和类属性的数据结构而完成对象设计，数据结构和算法必须符合行业需要；使用对象间的协作和对象关系完成消息设计。在评审设计模型时需要行业专家和用户的参与，从行业的角度提出修改意见。

（4）在集成测试阶段，行业背景的应用软件系统与普通应用软件相比，一般更强调进行 β 测试和进行新旧系统对比，需要使用行业的数据作为测试数据，测试案例需要由用户和专家参与制订，行业数据需要由用户和专家提供，测试结果需要由用户和专家确认和解释。由于地震数据具有突发性，当地震发生时系统的数据量非常大，因此系统测试需要在真实的数据环境中进行。我们在测试地震应用软件时使用的是真实的地震数据，从数据的采集、传输、存储、处理和显示等的各个环节均使用的是地震事件数据，根据测试结果和用户意见进行修改。

从上面的几个主要差别可以看出，具有行业背景的应用软件的开发需要行业中的各类标准、专业知识、现有相关系统和人员参与到开发过程中，要使行业特点与信息技术有机结合，在项目管理上还需要包含行业内的相关的人和物。

在地震应用软件的开发过程中，人员、设备、资料等的方方面面的资源均是由项目组与地震系统共享的，在知识和思想上充分融合，经过双方的共同努力不仅开发出了满足地震行业需要的应用软件，还丰富了双方的知识面，拓宽了眼界。

本人在项目中主要负责需求分析和软件系统设计，在实践中我发现，行业应用软件系统的成功开发是从其规划到运行的全过程中均应从行业中汲取各种营养，才能长出行业特征，具有行业味道，满足行业应用的需求。

论企业信息化战略规划技术

摘要:

本文通过建立就业服务系统的信息规划,讨论了企业信息化战略规划技术。企业信息化战略规划在企业信息化建设中占有非常重要的地位,是企事业信息化建设成功与否的关键。我们采用的方法是,成立信息化建设领导小组,由主要领导担任组长。一、了解就业服务系统的战略目标,使信息化的战略目标和就业服务系统的战略一致。二、统一部署,进行资料收集,除考察现有软件系统外,还要考察手工操作,形成资料,制作功能表。三、根据收集的资料,我们采用自顶向下规划与局部设计相结合的方法制定出了信息系统的框架。四、加强培训,提高员工素质。

本人在此次战略规划中担任技术层面的工作。

正文:

企业信息化战略规划,在企业信息化建设中占有非常重要的地位,是企业信息化建设成功与否的关键。

本人有幸参加了市就业服务系统的信息化建设规划,担任系统分析和需求分析等相关工作,主要负责技术层面的问题。

随着社会的发展,加强信息化建设成为各单位的首要任务。为了建设“数字城市”,加强就业培训的管理工作,构建信息化的培训管理格局,我们开始着手进行全市的就业培训信息化建设,根据省里的要求,做到同人同城同库,实现就业培训的“实名制”。

目前,就业服务系统的信息化建设缓慢,各单位各自为政,分别开发了不同的信息管理系统,这些系统的开发,没有统一的标准,系统之间互相不“说话”,无法实现数据共享,浪费了巨大的人力物力。为了从根本上改变目前的混乱状况,在上级领导的支持下,在 2003 年我们开发了就业服务系统的管理软件,实现了资源共享。

就业服务系统管理软件涉及到就业服务系统几个相关单位,包括培训、考核、职业介绍、就业、财政等单位。就业服务系统管理着全市失业人员的所有信息,要求系统具有从失业人员登记,发布培训信息,用工信息,到学员参加考核,顺利就业为止等一系列功能。

为了使就业服务系统的信息化建设达到预期的目的,我们进行了战略规划,成立了就业服务系统信息化建设领导小组,由主管就业服务系统业务的领导担任组长,主抓此项工作。具体实施的步骤如下:

一、我们根据全系统发展计划的文件,全面了解本系统的任务,目标和长期发展的战略规划。我们和上层领导座谈,认真了解全系统的任务和目标,及时,方便,准确,快捷为下岗失业人员服务,是我们工作的宗旨。认真阅读相关资料,准确理解长期发展的战略规划。

根据需求做好设计,形成报告,请有关的业务专家来进行讨论,并参考就业服务系统的发展计划,检讨我们制定的信息化战略规划是否符合战略发展计划。

二、由信息化小组招集各单位各部门的负责人开会,统一布置了全面进行信息化建设的事宜,要求各部门配合信息化小组的工作,把全系统的信息化建设工作做好。

随后我们紧锣密鼓的展开了调研活动。资料收集是非常重要的,没有详细真实的第一手资料,要想进行战略规划就是纸上谈兵,建设空中楼阁。

首先,我们考察现有的软件系统。

由于本系统有独立功能的业务部门,他们已有现成的信息系统。为了了解各业务部门现有软件,我们制定了各业务部门软件系统调查报告。该报告主要要求各部门填写现行系统的现状,主要内容包括:软件运行的操作系统,使用的数据库管理系统,主要的数据库和表以及表结构;现行系统运行所占用的空间大小,主要表的大小,软件的业务流程,基本框架结构等内容。

其次,对各单位的手工业务活动,进行调查。

我们主要采用了“请进来,走出去”的方法进行工作。

我们根据各单位提交的业务报告,请相关的业务骨干,进行讲解。并对不理解的地方进行提问,从而把业务流程“吃透吃准”。为进一步形成功能表打下坚实的基础。

我们到各单位进行走访,现场观摩和实习操作,对各业务部门的流程有了更进一步的理解。在进行走访活动之前,我们详细研究了各单位的资料,对其中描述的不清楚或者不明确的地方,做了重点标注,并记录下来,以备走访时使用。除了观摩和操作外,对业务人员和技术人员,按照事先准备好的问题进行采访。

第三,形成资料,制作功能表。

根据各单位提交的报告和我们走访的结果,对形成的文档资料,我们进行了整理。为了清楚的表达各业务段的流程,我们使用 Visio 制作了各种业务流程图,并用 Excel 制作了所需的所有表格,对表格的名称进行了统一,把各单位五花八门的表名进行了修改。例如:有单位把一月一报的表格叫月报,其中的内容为每日的内容,而对每月的累计报表,也叫月表;我们把表名进行了修改,规定:含有每日数据的报表叫日报表,含有每月数据的表格叫月报表。如果有几个月表,则能合并的就合并,不能合并的,则叫月表一,月表二,或根据报表的主要内容进行相应的命名。表格名称统一以后,便于交流,省去了许多解释的时间。

我们根据各部门的业务活动制定出了各部门功能表。

文档制作完成之后,我们进行审查,这次是由我们来讲,各业务单位听,对我们理解不对的地方进行修正。如此反复几次后,可以基本确保我们获得资料的正确性。

三、根据采集到的资料,我们采用自顶向下规划与局部设计相结合的方法制定出了信息系统的框架。将整个就业服务系统的业务分为相对独立的几个部分,失业人员管理系统,培训管理系统,鉴定考核管理系统,就业管理系统等。对于一些具体的业务采用局部设计的方法进行,如:报名,分配培训机构等。

我们设计了人员档案主题数据库,主要用于管理失业人员的各种情况,如:就业,培训,考核,技能等级,家庭住址,所在街道等信息。此数据库由各管理系统进行维护,保证数据库信息的及时、准确。

四、加强培训,提高人员素质。

在我们的调查过程中,发现各单位使用计算机软件系统的人员素质较低,对信息化几乎没有什么概念。开发出来的信息系统,需要使用的是“人”,没有人员素质的提高,盲目的进行信息化建设,也是很难成功的。

为此,我们组织了多期电脑扫盲班,提高班,加强了员工对信息化建设的认识,以及进行信息化建设所能带来的各种优点。通过培训,提高了员工的操作水平,为我们与各单位业务人员进行沟通起到了良好的作用。

总之,就业服务系统的信息化建设起步晚,底子薄,经过信息化建设规划后,为开展信息化建设指明了方向,为就业服务系统的信息化建设起到了良好的作用。

目前,本系统各单位已经使用了新的软件系统,结束了各单位各自为政的历史。但这仅仅是完成了信息化战略的第一步,为实现数据仓库,建立决策支持系统提供了基础。

在现阶段,本单位的信息化也仅仅是在业务方面进行的。其它方面的信息化建设依然开展不够。例如内部办公系统仅有和上级部门相联的一个办公系统,内部办公效率不高,手工操作还是比较多。对已有系统的利用率还不是很高,领导决策也没有完全借助计算机的帮助。计算机的使用大部分还停留在打字机的水平。要解决这些问题,我认为主要是领导对信息化建设要有足够的重视(特别是一把手),不能仅仅停留在口头上,要有实际的措施。比如领导要参与培训和计算机的使用,并增加一些物质鼓励。在已有信息系统的基础上,经过若干年的数据积累,构建数据仓库,决策支持系统。建立内部办公系统,邮件系统,使各部门员工自觉自愿的使用计算机来进行工作,尝到信息化建设带来的好处。

论软件产品线技术

摘要

根据公司软件系统开发的需要，我们在软件的开发过程中引入了软件产品线技术，成立了基于软件产品线的项目组。本人有幸参加了该项目，并在其中担任软件分析与设计、软件产品线核心资源开发的工作。

在软件产品线的开发过程中，我们使用了 ROSE 建模工具，有效地完成了产品线中核心资源和产品的建模分析与设计实现；我们使用了国际标准 POSC 数据模型框架，有效地解决了数据的多样性与可扩展性，实现了统一开放的测井数据访问系统；建立了统一的可扩展的地质绘制组件和统一公用的数据处理模块。最终圆满的完成了公司产品线的建立和各子系统的开发。

正文

我公司是中国西部最大的石油测井服务公司，主要从事测井工程服务、测井仪器与测井软件的研发工作，在公司内部已经开发了多套测井数据软件，其中包括测井综合地面监控采集软件、测井数据解释处理系统、测井实时快速处理与解释系统、生产测井解释系统，分别由多个项目组分别开发。

其中，测井综合地面监控采集软件主要配合公司的综合地面系统，通过与地面硬件系统的前端计算机的通讯，实现对井下测井仪器状态的实时监视和控制、测井数据的实时采集和显示。整个软件分为通讯模块、数据采集模块、实时解码模块、实时简单处理模块、实时显示模块、实时出图模块和主控界面模块等。

测井数据解释处理系统主要完成测井数据的测后处理，包括测井数据的解码、数据编辑、环境校正、常规处理、成像处理、成果图绘制、解释报告生成等功能，整个软件按分层架构开发，包括系统底层（包括数据底层、绘图底层、通讯层）、业务逻辑层（包括各种测井数据处理模块）和应用软件层（包括数据解码程序、数据编辑程序、深度校正程序、综合绘图程序、解释报告生成程序等）。它主要是针对勘探开发测井中产生的新井的测井数据处理，所涉及到的数据主要是按深度采样的连续的测井曲线数据，而成果图主要是一种按深度比例的连续的测井图。

测井综合地面监控采集软件和测井数据解释处理系统均以推出了多个版本，进行了大范围的生产运用。而测井实时快速处理与解释系统和生产测井解释系统是我们正在准备开发的系统，测井实时快速处理与解释系统主要通过通讯卫星与在野外进行工作的测井车上的测井综合地面监控采集软件进行数据通讯，准实时的获得测井数据，然后由在解释中心的测井解释专家根据邻井资料和区块经验进行实时分析，整个平台分为实时数据通讯模块、实时解码模块、实时处理模块、实时深度校正模块、实时环境校正模块和实时绘图模块等，而生产测井解释系统主要完成石油开采后期动态监测时的测井数据的数据处理，测井数据主要是离散的测井曲线数据，在数据绘制上和处理上与测井数据解释处理系统类似。

在以上的各个项目中本人主要负责了测井数据解释处理系统的软件分析与设计、软件构建的工作。对于以上的各个系统，如果用抽象的视角去观察，我们可以发现，其实他们处理的对象是一样的都是测井数据，在功能上都是获得数据后，处理数据，然后按规定的绘图格式展现给测井工程师，所不同时各系统的侧重点不一样，测井综合地面监控采集软件主要是完成对测井数据的采集工作，和通过数据的曲线图形方式监视井下仪器的工作状态和测井数据的正确性，对系统的实时性要求比较高；测井数据解释处理系统主要完成测井数据的测后处理和测井数据的多角度观察分析，对实时性和可靠性要求不是很高；生产测井解释系统与测井数据解释处理系统基本层次结构相似，只是数据的格式和显示方式不一样；测井实时快速处理与解释系统对实时性要求稍高一些。通过以上对系统的分析可知，在以往的开发中我们各个项目组的人员实际在数据的存取、测井曲线的显示、测井数据的处理方面我们做了大量了重复性工作，各个项目组都形成了自己的一套软件架构，在资源上造成浪费、且代码质量不能够得到有效保证。因此，由于数据存储和处理的一致性和绘图方式的一致性，我们可以引入产品线技术，有效的提高公司的软件系统的复用性和可扩展性，同时能够有效提高公司软件的开发效率和质量。

我们给上级领导提出了合并测井综合地面监控采集软件项目组和测井数据解释处理系统软件项目组，成立公司的软件研发部，并将测井实时快速处理与解释系统和生产测井解释系统交由软件研发部来进行开发，在软件研发部，依托测井综合地面监控采集软件项目和测井数据解释处理系统软件项目的软件开发经验建立公司的软件产品线，这个建议得到了上级部门的一致肯定，最终成立了软件研发部，其下根据软件产品线的要求设立核心底层组、产品组。

核心资源的建立。核心资源就是在测井行业软件中能够共享的部分。数据存储与访问部分、通用测井数据处理方法模块部分和测井曲线绘图部分，在我公司已建立和即将建立的各系统中都是高度共享的。由于我们的核心开发小组的成员就是原来各系统开发的核心成员，对原系统都有充分的理解，加上成立统一的核心小组后，更方便程序开发人员的之间的交流与沟通，我们很快就核心资源的需求达成了共识，对于需求的分析和核心资源的设计，由于核心资源涉及到的系统边界较多，特别是测井数据的存储与访问接口，因此我们使用了 Rose 建模工具，通过对原有系统的逆向工程和 UML 建模分析，理出了核心资源的接口设计和核心资源子系统的划分和子系统内各设计类，最终有利于我们建立合理的核心资源软件架构。

在数据存储与访问部分，由于涉及的测井数据格式较多，难以形成统一的数据存储格式和访问接口。通过采用国际石油软件组织的 POSC 行业标准有效的解决了这一问题，使用 POSC 的 Epicentre 数据模型实现测井数据的统一化存储，而 POSC 的 DEA 标准描述了数据的存取交换的方法，使用 DEA 使我们的系统具备了数据访问的标准接口。另外由于采用了国际通行的 POSC 标准，使我们的软件具备了开放性，能够支持满足 POSC-Epicentre 数据模型的所有数据格式和对外提供基于 POSC-DEA 的标准数据访问接口，使得我们能够方便实现与第三方软件的数据交换。

测井数据处理部分，根据处理的通用性，将处理包分为通用数字信号处理包和测井专用处理软件包，将数据滤波等通用数字信号处理方法放到一个包中，将测井行业专用的 POR 处理方法等放到一个包中，实现了方法模块的高效复用。同时使用简单工厂模式实现了处理模块的动态加载。另外在构建测井数据处理构件时，由于测后处理和测时处理模块的处理方式不一样我们很难进行统一接口的设计，测时处理主要是基于深度点的单点处理模式，而测后处理是基于深度段的处理方式，测后处理由于处理的复杂性可能会用到历史数据，在实现处理模块核心资源时，我们以单点处理模块为主，使用模块聚合方式实现了基于深度段的处理模块。而核心绘图资源的建立过程中也遇到了同样的问题，主要是实时系统和测后解释系统之间的绘图方式的不同，一个时基于深度点的行模式一个基于深度段的绘图模式，数据对象绘制图件较多，实时系统注重绘图的实时性，测后处理系统更关注绘图的效果。但是在图头的绘制上具有相同性对实时性要求不高，且图头的绘制要求基本一致，因此通过建立了一套图头绘制子系统和两个版本的绘图子系统，以应对不同的时间响应需求和绘图质量。

通过使用软件产品线技术，最终有效地提高了公司软件生产效率和质量、缩短了开发时间、降低了总体开发成本。但是也遇到了一些问题：1）核心人员原先都没有从事软件产品线的开发，往往只关注自己熟悉领域的软件设计，没有全局的观念，为此我们通过加强团队人员间的沟通，互相了解彼此的领域，使开发人员的全局观念加强，另外通过软件架构设计培训，提高软件人员的架构观念；2）没有合适的构件集成工具，处理流程的建立主要依靠代码来实现，因此在未来的开发迭代周期里有必要开发集成工具。

论软件产品线技术

摘要:

本人作为某软件公司负责人之一,通过对位于几个省的国家甲级、乙级、丙级设计院的考查和了解,我决定采用软件产品线方式开发系列《设计院信息管理平台》产品。

该产品线开发主要有如下功能:1)知识、资源管理平台;2)内部管理平台;3)基于WEB的办公平台。软件产品线最难开发的是核心资产的分析与建模,如何从用户重多不同需求中抽象出共性的东西,如何使得核心资产通过继承、参数化等方式能够组装成用户实际需要的产品,我以及我们公司的系统分析员做了大量工作。概括起来我们用了如下三种方式:

(1)加强核心资产开发的灵活度,部分产品作成用户能够自定义的功能,彻底免除产品化时的问题,但这样做难度很大,实施周期长;

(2)说服客户微调管理模式,以使得管理适应我们的产品线功能。这样做无疑是“双赢的”,但是实际不见得会行得通;

(3)根据流程最长的需求开发所有需要的构件,构件间的接口做成松藕合。演化成产品时进行构件组装。

正文:

本人所在软件公司于2003年6月承接了某《化工乙级设计院管理平台》工作,主要功能包括:1)知识、资源管理平台;2)内部管理平台;3)基于WEB的办公平台。该项目于2004年2月成功投入运行,得到客户的肯定。该项目的开发成功以及通过与该设计院员工的接触,对我触动很大。我作为公司主要负责人与其他负责人研究后,决定进入“设计院信息工程”这样的一个行业领域做深度软件产品开发。理由有如下几点:

1)设计院是一类知识技术密集型单位,员工素质较高,管理比较规范,有完善的硬件设施——性能良好的局域网平台,具备了实施信息化管理的基础;

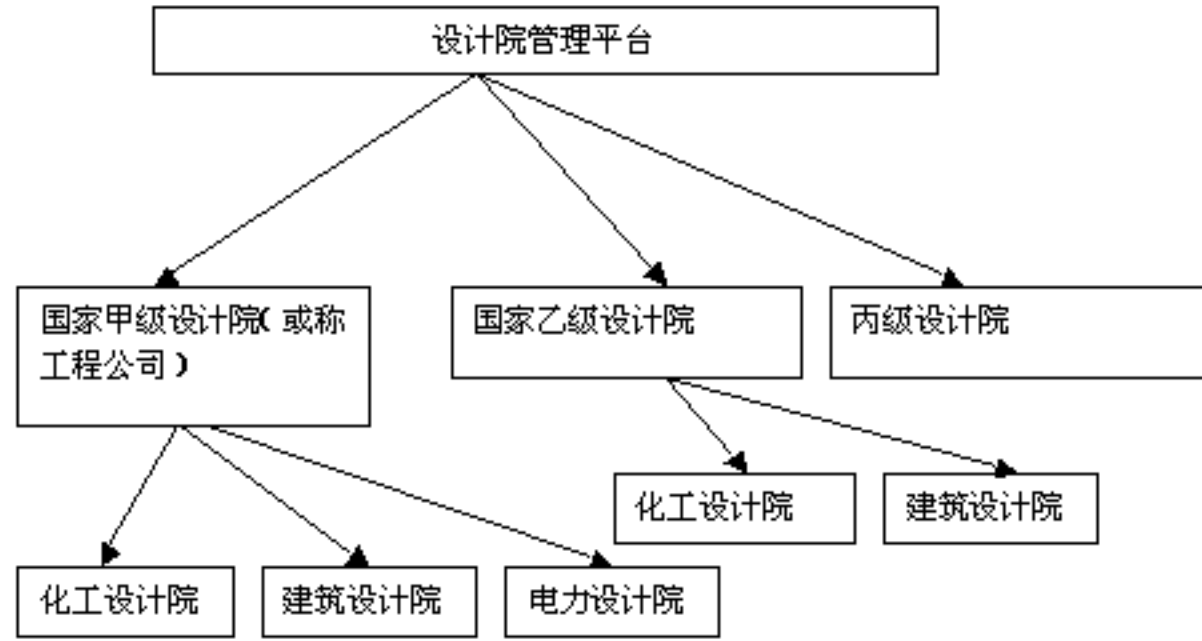
2)设计院的图形工作站所用的二维、三维软件或其它工程计算软件都用正版软件且价格比较昂贵,高层领导他们有支付软件费用的观念;

3)设计院开发一个项目,从可行性分析到项目实施、项目交付、到项目后期支持往往需要几年的时间,这期间产生巨量的、非常重要的各类文档(比如可行性分析报告、进度计划、项目条件、变更记录、会议纪要、计算书)和大量工程图,在整个项目开发期间这些资料要提供给项目开发人员查阅或使用、修改,这些资料的安全性、完整性都是至关重要的,这些资料既是知识、技术的承载体、又是法律的承载体;这些资料保存年限是30年或更长。目前设计院大都设立专门的部门,但基本都是采用手工管理。

4)设计院的客户在项目开发期间以及项目上线以后一段时间都希望得到良好的技术服务。技术支持既包括现场技术服务,也包括远程服务。现在的远程技术支持是很被动的接听客户电话、传真或EMAIL,没有建立专门的平台也没有进行规范管理。

有了对设计院内部运作的一般了解后,我们将已经开发成功的某《化工乙级设计院管理平台》精简做成演示原型,我带着这个原型以及对《设计院信息管理平台》的设想来到本市各类设计院:省级建筑设计院、市级建筑设计院、省级电力设计院、省级化工设计院、大型企业内部设计院等大小20家,征求他们的意见和建议,了解他们对该管理平台的认可度。我感到很欣慰的是80%的设计院高层领导对我们的产品设想有兴趣,并提出了针对自己单位的需要扩充或压缩的功能。随后的一个多月我带着补充修改后的原型产品,来到另外几个省会的设计院征求意见,结果:1)大部分的设计院目前没有完善的管理平台;2)如果我们真能设计出满足他们需要的产品,他们有兴趣购买。

通过持续2个月的调查我们公司于2004年5月决定投入这个行业的软件开发,因为要开发大量类同的软件产品,公司决定采用软件产品线开发。我们的对该系列产品的设想是:



这样划分的理由是:不同级别的设计院规模大小、管理模式、职能部门、待管理的资源区别非常大,国家甲级设计院除了设计以外,还能做技术咨询、工程总承包、建设监理、设备材料供货等工作,承接国际性工程。而丙级设计院一般隶属于相关厂矿企业,只能做本单位的项目开发,所以管理规模、管理资源相对较少,乙级设计院则介于两者之间。在同级别的设计院中,则因为设计院的性质不同,开发所采用的步骤和侧重点不同。该系列产品主要功能:

1)知识、资源管理平台:管理内容包括工程图纸、各类文档(可行性研究报告、项目计划、条件变更单、项目会议纪要、来往传真等)、标准图库、标准及规范、招投标书、合同等;2)内部管理平台:包括项目管理、人力资源管理、供应管理、客户关系管理、员工绩效管理等3)基于WEB的办公平台:用于对客户的远程技术支持、保持与关键设备供应商、国家级研究机构的快速联系、快速获取新技术新行业信息、推广公司形象。

确立了产品线开发的基本思路以后,我采用一种动态的组织结构进行开发,初始阶段将公司主要技术力量集中起来进行核心资产开发;计划当我们的核心资产开发已经有几个实际产品之后,逐步将重点放到产品开发上并不断演化我们的核心资产。我决定核心资产的开发先重点考虑“乙级设计院”级别,这样做的好处:将来开发丙级设计院产品,只需做适当的功能裁剪;另外避免了开发甲级设计院管理平台的技术风险和项目风险。

建模我们采用ROSE 2003,对需求分析的评估则采用RequisitePro,软件开发平台选用J2EE,数据库选用ORACLE,这样做主要考虑J2EE的平台无关性。在开发平台的选择中我们曾经考虑选用.NET,因为.NET开发相对简单和快捷,但.NET限制服务器在WINDOWS平台上运行,考虑我们今后客户中的“甲级设计院”,服务器一般都采用UNIX系统,为便于今后产品的移植,我们选择了J2EE。版本控制工具选用ClearCase。

核心资产开发最主要的是如何从众多不同需求中抽象出相同部分,并进行概括或分类。比如我们对员工绩效管理分析时就发现,相同的部分是:输入均为图纸量、其它文档、项目总利润、职工职称等,输出则为与项目相关的奖金。但是对奖金的算法,基本每个单位有自己的一套公式:有的对图纸量分专业计算复杂度系数;有的按本专业计算复杂度(比如关键技术、新技术或普通技术);有的按复用度(该图纸与原有图纸的相似度)计算折扣;有的不分职称只计算单纯的图纸量;有的则考虑职称或工龄;有的将计算书折算为图纸进行考核;有的专门对计算书或其他文档考核、有的计算考虑加班有的不考虑。

面对我们从20来个设计院收集不同的考核方法,我以及我们的系统分析员确实伤透了脑筋。如果全部采用抽象类或接口来实现,那么抽象的层次肯定比较多,产品开发人员今后很难理解核心构件的意图,而通过组装类构造的话,组装类的开发工作量比较大。而最大的问题则是如何适应用户考核方法的变更。我们最终的做法是:收集和整理20来个设计院的考核方案中所有考虑因素的全集,对这些全集进行分类,然后使用MVC设计模式设计成用户自己可以控制的控件,在控件间设定一定的逻辑表达,由用户自定义计算公式计算奖金。这种设计虽然核心资产的开发周期相对较长,但我觉得是值得的。因为大大缩短了今后产品的开发周期,并减少了产品定制时的错误。

本次核心资产开发中公用部分较多的是对“知识、资源管理平台”的开发,虽然最初我们也了解到不同设计院管理方法也有所不同,我们在研究几家管理比较好的设计院方案后,拿出了一套比较规范的管理流程方案,这几个设计院表示愿意改进他们的管理措施以适应我们的方案。这一点是我本次产品线开发最值得骄傲的地方,实现了“双赢”——客户改进了他们的管理,我们节约了开发成本。

我觉得最为困难的是在众多不同需求中如何抽象出相同部分,而对不同部分进行灵活的组装或构造比如本次设计的核心模块:“项目管理”的流程:项目开发→可行性研究报告→可行性评审→初步设计→初步设计评审→施工图设计→施工图交底→项目后期支持;但有的项目由生产厂家进行可行性研究,那么同样省略的就有可行性研究评审;还有的小型项目,不需要做初步设计,项目开工后直接进入施工图设计。这种情况下我根据流程最长的需求开发所有需要的构件,构件间的接口做成松藕合。演化成产品时进行构件组装。

因为目前我们通过核心资产演化的产品还不多,所以很难评价核心资产开发的整体质量。这个问题的解决有赖于更多实际产品的开发以及更多的用户反馈。本次软件产品线开发,我觉得有如下一些收获:1)对我们公司员工的素质是一个大锻炼,2)在开发过程中我们也与设计院一些高层领导及部门领导结下了深厚的友谊,加深了对该类技术密集型单位的认识,非常便于我们今后推广最终产品;3)通过核心资产的更多开发,我们今后产品开发的工程量、产品开发质量都会有数量级的改善。

论软件产品线技术

摘要

本人在测井行业的一个国有企业软件开发部工作，从 2002 年初开始，我陆续参加了多个测井软件开发项目，这些项目都是测井行业资料处理解释软件，具有很强的行业特征，其开发方向和应用范围都非常相似，从“测井资料处理集成软件”项目，开始我实施了软件产品线技术，虽然在开始阶段，由于经验不足和管理不善，遇到了一些问题，但是随着逐步实施，都得到的纠正和有效控制，目前这几个软件项目都非常顺利的完成，实施工期明显缩短，极大的提高了产品质量，本文就在这些项目中为什么实施软件产品线？在实施过程中遇到哪些问题？产品线开发支持工具选用情况和产品线实施带来的益处等进行论述，并分析总结在目前本单位产品线技术应用中存在的不足。

正文

本人在测井行业的一个国有企业软件开发部工作，在近几年来内，我陆续参加了几个的软件开发，2002 年初参加“测井资料处理集成软件”项目，此项目的目的是集成本单位重点实验室研究成形的测井资料处理方法，提供一体化的应用，服务于各个测井公司，我负责底层平台设计和开发，2003 年初参加“阵列感应实时分析软件”项目，主要针对单位自行研制测井仪器-阵列感应仪器 MIT 的现场资料实时处理，总体负责软件的设计和实现；2003 年中，参加国家级项目“山地地震和矢量地震勘探研究”的“成像勘探开发处理软件”子项目，本项目的目的是实现国内勘探解释方法和仪器的大整合，“成像勘探开发处理软件”子项目完成成像勘探部分的软件开发，我负责成像勘探部分软件架构设计和分析实现；2004 年初，参与总公司科技局“测井综合应用平台”项目，性质与“测井资料处理集成软件”类似，但集成范围扩大到整个测井行业，我参与了总体架构组，并完全负责底层平台的设计。

这几个项目都是测井行业资料处理解释软件，具有很强的行业特征，其开发方向和应用范围都非常相似。几个项目表现了很多共性特征，如数据录入接口方面，都采用测井行业标准的数据格式 CIF、LIS 和 XTF 等；数据表现方面，采用图件可视化方式；数据管理方面，除了第二个实时解释项目采用文件系统外，其他均采用数据库存储，使用测井国际化标准组织 POSC 的 EPICentry 规范，不仅如此，处理流程也大致相似，首先获取数据、进行数据转换，按照用户指定的方法进行资料处理，然后以形象化的方式反馈给用户一些处理特征，进行处理结果输出，各种资料归档等。除了这些共性外，几个项目也表现了一些个性特征，如“阵列感应实时分析软件”要求的实时性比较高，性能要求较高，采用文件系统，“测井资料处理集成软件”项目用于实验室方法集成，分析功能和研究性质略强于其他三个项目等，虽然存在个性，但是我认为几个项目更多地表现为一致性。

在这几个项目中我都实施了软件产品线技术。主要原因有以下几个：1、本部门长期承接测井行业软件开发，开发范围清晰，且容易界定，在多个连续的项目中，一直存在一些共性开发。在早期项目中，由于构架和开发接口不统一，造成重复开发的弊端，实施产品线可以有效节省人力、物力和财力。2、测井行业目前出现的软件都是层次架构，80%采用采用 C/S 模式，主要应用在测井资料处理上，其余采用 B/S 模式或者两种组合方式进行简单数据查询和处理结果显示，有利于产品线架构设计。3、由于行业特点，开发的软件很难适合所有的测井公司，而多数情况是量身定做，因此同一个项目可能出现多种软件版本，如华北油田专版，重点挂接生产测井解释功能，而长庆油田对斜井校正等方法较为重视，采用软件产品线可以方便解决这些问题。4、同时，测井行业中，方法的更新较快，但是一般的软件使用的方法都相同，如“测井资料处理集成软件”项目和“测井综合应用平台”项目使用同样的软件处理方法和图件绘制方式（测井行业特征，方法永远采用最新的），一旦解释方法和绘制方式改变，所有的软件要求能统一改变。

我在接手“测井资料处理集成软件”项目后，与其他开发人员向单位建议了软件产品线的开发技术，得到同意并开始实施。由于是新项目开始，而且软件规模不算很大，因此，我们直接采用了革命的方式，项目直接实施全新的软件产品线方式，我将团队分为两路，一路总结分析旧项目的软件，一路进行当前项目的分析界定，同时聘请行业专家和技术专家结合本项目从大局进行规划，首先进行行业分析，对行业分析结果与单位的规划进行整合，最后在形成的领域架构基础上进行当前系统的需求分析和设计界定，其中开发重点在核心资源库上，然后当前项目的软件直接复用资源库上的产品线构架和构件。这几个项目完成后，本单位的核心资源库得到了丰富和充实，由这几个项目直接构成的核心资源包括两套产品线架构，分别适用于测井资料处理和测井资料检索管理两种开发类型；数据格式转换构件组，能够进行十几种测井原始数据进行转换和录入；测井图件绘制构件组，数据按照四十多种图形方式显示在通用的桌面系统、浏览器，并能以位图、矢量元文件、SVG 和 CGM 进行输出；还有文件系统和数据库操作标准和一些商业构件库和中间件等。

几个项目完成的非常顺利，除了第一个项目的出现短暂延期外，其他项目大幅度缩短了开发周期，而且软件维护和版本升级非常方便，但在实施过程中我们也遇到了一些问题：1、在第一个项目实施期间，经验不足，人员分工与过程组织不合理，核心构件与应用系统开发混淆，导致了开发设计的反反复复，延迟工期。2、部分核心资源设计不合理，某些构件在老项目方便使用，但没有充分考虑新项目的特点，在新项目无法充分利用，如果直接修改冲突构件满足新项目，老项目的维护和更新又无法与新项目配套，在一段时间内存在同时维护多套多版本构件的情况。3、资源管理不利，因为产品线实施需要非常强大的构件管理和配置管理，我的四个项目针对不同油田的应用每个项目都生成了多个软件产品，维护还包括了核心资源维护和应用系统维护，实施前期经常出现管理不善带来的负面影响。

到目前，我们逐步解决了这些不利情况，主要是加强人才培养、资源管理和组织管理。在资源管理上，我们主要通过自动化工具来解决。目前支持产品线的工具不多，大多是构件的管理和组装工具，在系统分析设计和建模方面一直采用 Rational 的 ROSE，软件配置管理采用 CVS，核心资源管理方面从初始的利用 CVS 转变到北大青鸟的公共软件构件库管理系统，它是采用了 B/S 的多层体系结构，分布式的应用架构，基于 EJB 技术，具备较强的灵活性和扩充性，支持刻面分类等多种分类模式，并提供基于角色的用户管理机制，使系统具有灵活的权限分配和安全的控制方式。在选择这些工具的选择上最重要的是核心资源管理工具选择，因为其他工具相对成熟且熟悉，而核心资源库是产品线最重要的部分，并须能进行统一的管理和识别，如关键词分类、枚举分类和刻面分类方法等，因此我们选择了北大青鸟的公共软件构件库管理系统来进行核心资源库的维护，它不仅满足上述要求，而且完全支持 UDDI V2.0 标准规范，而且支持构件统一、描述、发现和集成。

产品线技术在本单位不断扩充与发展，到目前为止，形成了采集、通讯、解释处理等多方面的产品线架构和构件库，利用这些资源，可以方便快速地“组装”一个测井实用软件系统。通过实施软件产品线，本单位测井软件的开发模式逐渐规范，不同项目复用同一个或多个产品线架构，做到组件复用的最大化，实施工期明显缩短，极大的提高了产品开发过程和产品质量。虽然有了几年的技术和资源积累，我认为在本单位的产品线开发上还存在一些不足，如在领域工程方面，缺乏具备深厚行业经验的领域分析人员，需要在不断的实践培养过程中，辅以本行业外部专家的咨询和辅导；在核心资源方面，缺乏高效的管理，虽然使用了较为方便的构件管理工具和其他辅助工具，但是其智能化欠缺，不是很成熟，同时构件的组装上，还是采用人工代码方式，缺少智能组装工具应用；还有就是管理和过程方面，传统单兵作战思想影响依然深远，国有企业开发人员或多或少有一定技术封闭性，不利于核心资源积累，而且向团队开发和多部门协同开发转变存在一个阶段，需要更好的管理和更多时间的磨合。

论基于构件的软件开发

摘要:

基于构件的软件开发是提高软件生产效率和软件产品质量的有效途径, 本文结合我们的实践, 以“在线学习支持服务平台”项目为例, 讨论了基于构件的软件开发的技术的应用。由于我校现有的各级软件系统都是基于微软 Windows 系列平台, 因此我们确定使用微软的 COM 组件技术来开发该平台, 并介绍了该平台所使用的几种 COM 组件, 主要采用 VB6 语言编写通用模块并生成 DLL 文件及注册成为 COM 程序, 客户端用 ASP 语言来实现并通过 ADO 技术来调用 SQL SERVER 2000 和 COM 组件。

在本项目中的开发过程中, 我担任了系统设计工作。

正文:

“在线学习支持服务平台”是面向我校开放教育学生进行远程学习教学辅导, 经过多年的远程教育模式的探索, 确立了成熟完善的远程教育教学模式—利用先进的网络数字信息技术, 为广大的学生提供开放的教育平台和最优秀的教育资源, 突出个性、学生自主学习的教学。

“在线学习支持服务平台”是一个综合性的在线式基于 WEB 的远程教学平台, 存储着核心信息数据, 提供网上课程、信息发布、查询、BBS、VOD 视频点播等教学服务, 该系统的开发技术主要集软件复用、企业级应用程序开发等技术于一体的“基于构件的软件开发”。系统运行于 WINDOWS SERVER 2000 , 用 SQL SERVER 2000 为后台数据库, 用 ASP+IIS5.0 来架构网站。

由于 COM 组件既可以被嵌入动态 Web 页面, 还可以在 LAN 或桌面环境的 VB、VC 等应用中使用。另外该组件之间是彼此独立的。当应用需求发生变更时, 可能需要更换中间层的个别 COM 组件, 但并不影响其他组件的继续使用。组件具有若干对外接口(属性和方法), 可以根据不同的应用需求, 有选择地使用不同的接口。即使不再使用某些接口时, COM 组件本身仍然可继续使用。同一 COM 组件可以在不同的应用环境中重复使用。因此, 结合我校的实际情况, 我校现有的各级软件系统都是基于微软 Windows 系列平台, 且开发人员对 COM 组件技术也较熟悉, 对开发语言 VB6 也很熟悉, 因此我们确定使用微软的 COM 组件技术来开发该平台。

该平台采用 B/S 结构进行设计, 把整个系统分为三个层: 数据库层, 应用逻辑层, 用户界面层。用户界面是浏览器(如 IE 等), 并通过 ASP 语言来实现同应用逻辑层构件交互。应用逻辑层负责事务处理, 应用逻辑层用主要通过使用 COM 组件方式来实现, 数据库层用 SQL Server。

我们依据平台的主要功能, 在平台开发中, 如果采用传统的方法来开发, 则每实现一个功能都要编写同样的代码, 为了节省开发时间和提高维护效率, 我们把共用的代码模块都做成组件, 例如我们把记录操作(如记录的删除、增加、修改等)、数据库操作、查询做成用户管理组件, 把用户身份认证和用户类型识别做成用户管理组件, 把所有实现与数据库的连接做成连接组件, 把用户的错误操作、与系统的交互出错等做成错误处理组件。对于各组件我们采用 VB6 语言进行编写并生成 DLL 文件, 通过注册成为 COM 程序, 供各个组件调用。在数据库连接方面, 我们采用了 ADO 技术。由于 ADO 采用了 OLE-DB 技术, 使能访问各式各样的数据并提高了访问性能。

在该平台的开发过程中, 主要设计和实现了以下一些 COM 组件:

(1) 用户管理组件, 包括身份认证功能

我们主要定制 COM 组件用户管理组件 UserCheck.dll 进行用户管理处理。该组件主要完成两个功能: 一是身份认证功能, 主要是提供用户登录时验明身份, 保证应用的安全性。二是根据用户所输入的账户名确定该用户的类别。

因此, 该组件具有三个接口, 每个接口代表组件的某个属性或方法。对用户的登录请求做出相应的处理: 如果是学生登陆则转入学生学习平台, 如果是教师登陆则转入教师平台, 如果是管理员登陆则转入管理员平台。

(2) 查询和提交信息组件

我们主要定制 COM 组件 QuerySys.dll 进行查询和提交信息处理。该组件主要完成两个功能: 一是供学生用于查询学生成绩和查询课程信息; 三是提交学生注册信息。

该组件具有两个接口, 每个接口代表组件的某个属性或方法。如果学生的请求是查询功能(QueryInfo 方法), 则将查询信息请求做出相应的处理, 并将查询结果集返回给出学生。如果学生的请求是提交注册信息(Submitinfo 方法), 则将提交信息请求做出相应处理, 并将信息提交提示返回给用户。

(3) 连接组件

我们主要定制 COM 组件 Conector.dll, 该组件主要完成与数据库的连接。该组件具有一个接口, 那就是确定数据源, 以便自动连接后台数据库。

(4) 错误处理组件。

我们主要定制 COM 组件 Cerror.dll, 该组件主要确定错误类集, 该组件具有一个接口, 主要是输出错误信息, 方便用户排错。

我们把编译好的组件, 将其在 MST 中注册, 并将其分布在服务器上, 这样就可以在设计平台过程中进行调用这些组件了。在本系统中, 我们通过以下几种方式把组件集成到系统中来:

一是连接集成, 即我们将组件直接嵌入 ASP 主页中, 即在 ASP 脚本中通过 SET 对象名=Server.CreateObject(“类名”)来引用, 使此二进制组件可以运行于服务器端。

二是容器集成, 即如果一个组件需要调用另一个组件时, 就在需调用的组件中引用另一个组件的方法。例如在使用查询和提交信息组件时就需要先调用连接组件。

我们结合连接集成和容器集成两种方式来组装系统, 以登陆界面为例, 在客户端我们只提供两个输入项和一个提交信息的功能按钮, 主要通过 ASP 来实现。在服务器端, 主要根据用户输入的信息来进行相应的处理, 这就要调用各种组件。如果学生以错误的学号和用户名登陆, 则系统调用用户管理组件、错误处理组件和连接组件, 返回非法用户的信息。如果学生以合法的身份登陆进入学生平台, 这就要调用用户管理组件、连接组件。如果教师以合法的身分登陆进入教师平台, 也要调用用户管理组件、连接组件。

目前, 该平台运行收到良好的效果, 我们采用 COM 组件技术进行开发, 减少了重复输入代码的工作, 缩短软件的开发周期。同时, 在进行系统维护时, 我们只关心组件的接口参数, 而不用再考虑组件内部的具体实现, 提高了系统的可维护性。在以后的工作中, 如果我们要扩展某些功能时, 也可以重复利用这些组件, 提高了系统的可复用性。目前该平台运行存在的缺点是: 由于在 ASP 中运行的 COM 组件是二进制代码, 当 COM 组件工作出错时, ASP 不能指出 COM 组件发生错误的具体位置, 只能简单显示对象创建不成功。这样就给我们在调试该平台过程中增加了难度。

论基于构件的软件开发

摘要

去年初，单位承担了新立的“测井资料处理与解释集成软件”项目，目的是集成目前国内零散的测井解释方法，我有幸参加该项目，并负责软件系统平台设计和部分开发工作，在项目的实施过程中，我充分进行基于构件的软件开发，复用成熟的商业构件和本单位的构件资源库，同时考虑了本项目开发资源的进一步复用，形成了绘制组件包，数据交换组件和数值计算组件包等。基于构件开发，大大提高了软件的质量，缩短软件的开发周期。开发的软件目前在石油测井几个油田现场使用，并得到用户的好评。本文就在本项目中如何进行基于构件开发进行描述，并在复用构件的使用和丰富方面谈一些自己看法。

正文

2004年3月，我有幸参加了单位软件开发项目——“测井资料处理与解释集成软件”，并负责系统平台设计和部分开发工作，本项目是由中国石油天然气集团公司科技部的测井软件类项目，在项目立项之前，国内测井行业为了提高石油解释的分析和决策能力，在解释方法方面广泛与各大高校进行合作，经过长时间的积累，形成的非常多而且实用的方法和处理模型，并开发了简单的使用系统，但这些应用非常零散，不统一，无法协调工作，处理效率低，多个解释系统之间没有必然联系，而且数据格式多种多样，不利于测井的精细解释和综合应用。本项目的目的是为了综合决策能力，提高石油解释的准确率，考察了解国内流行的测井资料处理解释方法，将成形和半成形的解释方法进行筛选和整理，集成到统一的系统平台上，增强一体化处理功能和商业化能力，并有偿反馈给石油测井现场用户使用。

在本项目立项前，单位承担了较多的软件开发项目，软件开发方法一直在改进更新。单位早期的软件以数值计算为主，主要采用面向过程开发方法，但随着软件的复杂度增加，可视化要求和开发效率，在面向过程方法不能满足要求的情况下，开发方法转为面向对象方法，实现界面编程，可视化操作交互，开发效率得到大幅度提升，但是这种方法也存在一些不足，无法有效实现资源复用，多个项目之间的资源复用仅限于部分开发过程和偶尔可以复用的模块类，新项目很少能借鉴老项目，即使存在明显类似的项目也无法有效复用，出现了多个项目组软件界面不统一，操作习惯不一致等情况，因为新项目只注重眼前效率和操作适应性，根本没有考虑以后的复用，导致重复开发，浪费人力物力。因此，在面向对象开发方法的基础上进行改进，采用基于构件的开发方法，在每一次项目中，尽可能地考虑以后可重用的部分，以构件形式进行设计开发，虽然在单部分的开发工作可能增加，但是多项目综合考虑，大大提升软件的开发效率。而且，本单位大部分项目为测井类软件项目，在软件的设计思路和最终成品上看，很多部分非常相似甚至雷同，几个连续项目存在联系，基于构件开发的软件方法在本单位更能体现作用。

在“测井资料处理与解释集成软件”项目的设计和实施过程中，我使用了基于构件开发的方法，充分利用公认的构件技术-COM/DCOM 和 ADO 等，使用已有的商业构件-Xtreme 和 Esstentails 等，复用单位的构件资源-数据转换组件，并开发了新的构件，如绘制组件包，数据交换组件和数值计算组件包等。

本软件系统以测井解释分析为主，主要包括图件绘制、用户交互操作、数值处理和数据存储等几个部分，我将软件设计为 C/S 四层体系架构。客户端应用程序采用 Microsoft Visual C++ 开发，本单位已经购买了 Xtreme 的使用权，Xtreme 是基于 VC 的 MFC 框架的界面构件，它是微软的合作伙伴，能经过简单的几步向导就能产生美观的软件界面，它生成的界面具有一致性，对于测井解释集成应用，统一的界面可以降低软件的学习难度，增强软件的易用性，使用 Xtreme，我非常方便地实现了界面设计和开发工作。

图件绘制基于 GDI 和 GDI+ 开发，每个绘制对象采用单独的一个类进行封装，绘制的载体对象派生于 MFC 的窗口类，其他对象派生于同一个基类，基类设计足够的消息响应函数，对象采用重载虚函数或者操作自动定向，实现鼠标响应，属性修改和递归型子对象管理。绘制载体类和基类封装在一起，形成绘制核心库，其他对象单独以 MFC 扩展动态库实现，一个或者几个对象设计成一个扩展对象库，如果要增加和删除某个对象，只需修改相应的扩展动态库。这样，只需要绘制核心库与必要的绘制对象扩展库，就可以提供相应的绘制功能，而且绘制的对象可以定制并进行二次开发。在某些绘制图件的设计中，考虑到算法的难度，直接利用较为成熟的绘制构件-Gigasoft Pro Essentials，如等值线图，区块三维分布图和多井连通图等图件的绘制算法，核心绘制部分直接由 Essentials 完成，用户交互响应由对象封装。

测井解释的核心部分是解释方法，每种测井项目有多种解释方法，而每种解释方法又有多个解释模块，因此解释方法的设计是相对复杂的。在考虑到测井解释的流程化处理和模块动态增减的要求，采用 COM 组件技术进行了模块设计。例如常规测井解释项目，包含 POR、CLASS、SAND 和 CRA 等多种解释方法，而这些解释方法都包含几个解释过程，如 POR 包含泥质含量计算，渗透率计算、孔隙度计算和含水饱和度计算，将上面的每个过程设计为功能完整封装的 COM 组件，当用 POR 解释方法进行资料处理时，依次调用处理过程完成解释。由于采用组件方法设计，方法模块可以动态添加，如果在 POR 的基础上需要添加含油饱和度计算模块，只需要设计相应模块，动态添加到 POR 的管理队列中。在解释方法分解设计的同时，将多种相同的数值计算方法进行提取抽象，形成满足自己需求的数值计算组件包，方便多种解释方法的复用，并为新的解释方法研究提供强大的计算能力。

在软件设计开发中，数据的存储最为繁琐，由于测井解释面对多种数据格式，如国内 Forward 软件使用 wis (WellLogging Information Sysetm 测井信息系统) 数据文件，国外 GeoFrame 使用的 Lis (Log Information Standard)、eXpress 使用的 XTF (eXchange Tape Format 可交换带格式) 格式和 DPP 使用的 Clis 格式等，这些测井软件使用相当广泛，如果本项目软件能进入市场，这些格式都需要兼容，但是将这些格式都作为应用对象很难实现或者不可能实现。在数据储存方面，我直接采用 Sql Server 数据库存储，存储结构设计采用国际标准结构，国际石油标准化组织 POSC 制定了一套 EPICENTRY 石油数据存储标准，这种结构被石油系统广泛采用，可以存储石油系统多行业数据，如测井、录井、钻井和开发类数据等，直接采用已有标准可以满足复杂的存储要求，避免重新设计格式的存储能力限制。数据访问采用 ADC 数据访问组件，实现数据通讯和交换。在数据库的数据录入方面，我充分利用了本单位其他项目开发的数据格式转换组件，用于支持多种测井数据的解码和编码，由于以往项目的数据操作是基于文件系统 (CIF 格式)，内部封装了固定 CIF 文件格式转换功能，我将组件进行了适当修改，将基于 CIF 文件格式的操作改为内存操作方法，实现向数据库录入。

在数据库和解释模块之间，我设计了数据交换组件，因为测井解释的特殊性，一次解释需要获得数据非常多，一次解释处理至少需要二三条曲线，测井采样间隔一般为 0.01 米左右，2000 米井段每条曲线传输数据量就有 100 多 K，如果需要处理成像数据，数据量更大，可能有几十兆，甚至上百兆，由解释模块或测井图件直接采用 ADO 进行数据库访问，每次解释会多次访问数据库，势必会降低软件的执行效率和处理的稳定与连续性。数据交换组件将一次解释数据全部读出，打成固定格式的数据包 GDS (Global Data Storage)，GDS 可以以文件形式存放和内存形式交换，应用模块只处理 GDS 数据包，完成各种应用后，按照同样的格式传回数据库，优化处理性能，而且基于 GDS 构件的应用还可以扩展开来，GDS 不一定只与我们设计的数据库进行交换，如果需要增加其他存储方式数据库，只要知道详细的存储结果，开发访问组件，GDS 就能与其相联进行信息处理，新的数据格式也是一样，只要提供了访问组接口，也能够进行测井解释。

开发过程中，我充分利用已有的构件，也充分地考虑了本项目开发资源的进一步复用，开发了绘制组件包、数据交换组件和数值计算组件包等构件，完善单位的可供复用的构件库，积累和固化知识财富。到目前为止，绘制组件包和数据交换组件已经在新项目测井数据采集平台使用，效果非常好。

基于构件的软件开发，可以缩短软件产品的开发周期，提高软件质量，但是如果构件选择不当或使用方法不当，也会带来一些麻烦，在构件的使用上，我也走了一些弯路，如在使用 Xtreme 上，由于购买了它的源代码，曾将某些绘制模块建立在其源代码级别的紧密耦合，当 Xtreme 提供的新的版本后，这些紧密耦合的模块无法适应新版本，如果想继续应用，必须再进行同样的开发工作，最后我放弃了某些紧密耦合的功能，将绘制交互与 Xtreme 完全隔离开，后续的版本升级再没有带来任何问题，类似的教训还有很多，总结分析后，我认为在基于构件的开发中主要注意一些问题：如项目开发不要与已有构件紧密耦合，防止构件升级带来的影响；构件的使用一定要进行严格的测试与审核；不要勉强应用某些构件，如使用巨大构件的其中某个微小部分，虽然功能很适合应用，但可能影响软件效率；新项目必须注意构件库的积累，只有这样才能充分体现基于构件开发的便利；新构件功能一定要完整，文档及接口定义明晰，否则难以复用。

在本项目的开发工作完成后，相比以往的开发项目，开发周期明显缩短，软件的质量明显提高，用户软件维护要求也减少了，而且其扩展性非常强大，目前该解释软件已经在华北油田、长庆油田和土哈地质研究院进行了现场应用，得到了用户的一致认可。通过这次案例，我认为在软件系统的复杂性不断增长的情况下，基于构件的开发能有效提高软件质量、积累和固化知识财富，并有效地缩短软件产品的开发周期，提高了软件生产效率。

论软件项目计划的制定

摘要:

本文讨论了一个作者参与的软件项目的项目计划制订的若干问题。项目所开发的产品是一种智能电子教学设备,该设备可以实时同步地将用户在硬件端的书写内容显示在计算机屏幕上,并可以保存、编辑、打印用户输入的数据,联网的计算机也可以实时观看用户的书写过程,并且用户还可以通过投影在硬件端的 PC 机画面交互操作 PC 机。

作者是该项目的软件开发组负责人兼软件架构师。作者针对项目计划的制定采取了:分而治之,逐步求精,经验数据三个主要策略,从而得到较好的效果。

正文:

2002 年 6 月,作者所在公司启动了一个项目,该项目开发出来的产品是一种智能教学设备,该设备可以实时同步地将用户在硬件端的书写内容显示在计算机屏幕上,用户可以保存、编辑、打印通过硬件端输入到计算机的书写内容,联网的计算机也可以实时观看用户的书写过程。另外,用户还可以通过投影在硬件端的 PC 机显示画面交互地操作 PC 机。

作者有幸全程参与该项目的开发,并且担任了项目 PC 机软件开发组的负责人兼软件构架师的角色。对于这种实时通信且具有联网功能的软件项目,我认为首先需要制定一个良好的项目计划,才可以保证项目开发的成功。

总结这次项目的经验,我认为行之有效的策略有三个,分别是分而治之、逐步求精、经验数据。下面就结合这三个策略详细讨论本次项目计划的制订。

一、分而治之

将一个过于复杂的问题分解成若干复杂度不那么高的小问题来依次解决,这种方法人类已经采用了几千年。这里我们也可以用于项目计划的制定。因为整个考虑项目的方方面面来制定计划其复杂度已经超过了人类处理问题的能力。为了解决这个问题,可以将整个项目分解为一些更小的组织体,逐一进行处理,这项工作也就是项目管理中的 WBS(工作分解结构)。

比如针对这次项目中采取的 RUP 开发过程模型,我在完成需求管理计划时我就将计划内容分解成初始、细化、构建、移交四个阶段来分别制定,最后合到一块儿就是完整的需求管理计划。

除了按时间段分解的角度来制定项目计划,我制订软件开发计划时同时按照了 RUP 过程方法的工作流的概念来分解项目计划的制定工作,根据每个工作流在四个阶段业界通用的工作量估计来制定计划,安排工作人员以及相应的软件资源。因为软件开发计划涉及到多个工作流,我认为以这种方式分解是合理的。同时因为本项目的特点,我省略了业务建模工作流,这是因为这次的产品是以硬件为主,软件为辅的消费类产品,所以业务建模不是那么必要了。

以不同的方式分解项目,可以从多个不同的角度来制定整个项目计划,有利于全面、深入地了解项目,避免“瞎子摸象”的情况发生。

二、逐步求精

计划工作其实是一种管理未来、管理未知的工作,而未来是变化莫测的,还存在许多自身无法掌握的因素,因此存在很大的难度。而解决这一困难的法宝就是逐步求精。按照先框架后细节,先粗后细地进行项目的计划。

比如在这个项目中,在接受这个项目后就开始了做了一个初步计划,这个计划的内容主要是做出时间上的安排。因为打算在 2003 的 5 月需要用这个项目的产品申请国家中小企业创新基金的支持,所以完成时间就定在了 2003 年的 4 月,预留一个月用于写申请报告。总的时间进度确定后,大概分配了三个时间段:系统工程分析、软件开发模型确定、软件产品制造时间段、项目总结。

等到确定这次项目后的 RUP 开发模型后,就可以继续对项目计划进行第二次求精了。其实 RUP 过程中出体现了逐步求精的理念,比如在初始与细化两个阶段都要产生出项目计划的制品。这样我就可以在这个两个阶段对项目计划逐步求精,比如在初始阶段只是将我需要完成的项目计划分为了需求管理计划、软件开发计划、实施计划,然后在细化阶段我再具体地制定每类计划的详细内容。

比如在初始阶段时架构设计考虑以 MFC 为平台,根据这个决定软件开发计划的制定是比较粗略的,在细化阶段架构设计进一步详细,这时已经清楚各个模块和 MFC 的 Doc/View 主结构的接口定义,以及各模块之间的接口定义,这时我就可以根据所需开发的模块制定计划。比如这时我就计划了特效界面模块开发分两次迭代,第一次迭代计划一个月时间,第二次迭代两周时间,第一次迭代需要完成放大和缩小、树形选择、缩略显示等主要的界面效果,第二次迭代的主要任务是根据用户反馈进行修改调整。

三、经验数据

要制定一个好的计划离不开精确的估算。不过项目计划是在项目开发的早期制定的,而在早期要完成精确的估算是非常困难的。要解决这个问题的关键就在于“经验数据”。由于整个软件产业都还十分年轻,经验数据的积累都普遍不足,才导致这一现象的出现。

但是因为这次项目开发的产品在国内还没有开发过,在加上公司没有积累深厚系统的项目历史数据。针对面临的困难,我选用了 FP 功能点分析做为项目主要的估算方法。因为 FP 方法中有大量项目经验数据可以从网络上获得,同时其数据功能 ILF、EIF,以及事务功能 EI、EO、EQ 的计算对经验数据依赖不强,只需对概念理解正确一般就可以正确估算了。在估算成本的时候,因为公司以前的生产率数据是以 LOC 为单位的,我利用软件工程书籍中的“逆火”经验数据,将 LOC 转换为功能点单位,当然,这里必然导致一些误差。为了降低估算误差,最后使用 Delphi 专家分析法对估算结果进行了调整。

Delphi 方法是一种集策法,也就是通过多名专家对估计值的不断校正的方法。当然,请专家增加了项目成本,不过最后得到高质量的项目计划还是值得的。比如,在某专家的建议下我们改变了自行开发网络层组件的计划,而是采购现有的完全可以解决项目需求的成熟的中间件产品,这个策略的调整在后来证明是正确的。一开始犯错误的原因是由于我们网络开发经验不足把用户需求想复杂了。

最后谈一下使用的工具软件。在制定项目计划过程中我采用了 Microsoft 的 Project 2003 绘制甘特图。因为项目的进度安排是和项目中每个人都是息息相关的,所以在做甘特图前我首先征集了大家对文字和条形图效果的意见,然后按大家的意见进行了美化,比如用鲜艳的颜色标识关键任务,放大任务摘要信息,突出里程碑信息等。这在有些项目管理者看来似乎是小事,不过我认为一个赏心悦目的甘特图可以带给观看者好的心情,而好的心情可以大大提高工作效率。

同时,考虑创新基金支持的项目在交互期限上有很大压力,所以在定义甘特图任务的依赖关系时我采取了业界惯用的“时间盒”的技术,也就是在每个任务的任务信息对话框中“前置任务”一栏中的“延隔时间”我填入 5%—15%,也就是说当任务完成 90%左右时就可以结束转而执行下一个任务。因为本项目中的所有人员几乎是全程参与,所以我不是很担心每个任务遗留的少量问题在下一阶段没有负责人去解决。

配合 Project 2003 使用的估算软件是 Software Productivity Research 的 KnowledgePlan。这款工具软件的最新版加强了对 Microsoft Project 2003 以及 RUP 开发模型的支持,而且其中的 Project Template 功能允许用户采用自己定制的 WBS 来进行估算,这些因素使得 KnowledgePlan 对本项目的项目计划成功制定带来很大的帮助。

在上述三个策略的指导下,以及合适工具的辅助下,使最后形成的计划有效地指导了后期的开发活动。项目开发出来的产品通过了专家的鉴定,获得了国家中小企业创新基金的支持。

项目完成后发现的问题是早期计划的估算结论偏差还是较大,看来还是受到缺乏经验数据或者经验数据不够精确的影响,所以在以后的工作中需要开展有效的度量的工作,为公司积累覆盖面广且尽量精确的经验数据。

论信息系统监理中的组织协调

摘要:

2003 年 7 月我所在的单位承接了**市城市综合地下管线信息系统建设项目的监理, 监理工作的范围包括: 系统设计; GIS 数据库系统和地下管线综合管理系统的开发; 地下管线勘测数据加工入库的质量控制。我作为总监理工程师代表参与了该监理项目。

在监理的过程中我感到良好的组织协调是实现“四控三管”等监理目标的重要方法和手段。应当在项目的监理规划中细致地分析项目运作的模型, 准确地定义各个项目干系人的角色和活动, 以便让各个干系人都清楚项目的运作过程, 并且认同监理的工作。在此基础上, 我们参照 UML 的方法分析项目活动过程, 本着公平、公正、独立的原则在参与项目的各方之间组织协调, 加强信息管理和信息交流, 顺利地完成了监理工作。

正文:

城市地下管线是城市的重要基础设施。运用地理信息系统(GIS)等信息技术实现对地下管线的现状以及其建设和维护情况的动态管理, 对整个城市的建设和发展有着重要的意义。2003 年 7 月我所在的单位承接了**市综合地下管线信息系统建设项目的监理, 我作为总监理工程师代表参与了该监理项目。

该项目是**市地下管线普查和地下管线管理机制建设的一部分, 是一个大型的项目。项目内容包括: 编制勘测规程和数据标准、建设地下管线数据库系统、开发地下管线综合管理信息系统、开发数据处理模块和 CAD 插件发、地下管线勘测成果数据的处理和入库。该项目采用 GIS 技术建设城市地下管线的三维空间数据库, 要求实现动态管理和决策支持等功能。监理工作的范围包括: 系统设计; GIS 数据库系统和地下管线综合管理系统的开发; 地下管线勘测数据加工入库的质量控制。

该项目采用螺旋式的过程模型。首先勘测 0.6 平方公里范围的数据作为实验数据。分析实验结果制定详细设计方案和一系列的规程及数据标准, 并开发系统原型。其次, 按照定义的勘测规范和数据标准重新勘测 1 平方公里的测试范围, 评估测试原型。并且改进原型开发试用系统。然后, 在 5 平方公里范围内试用, 完善整个地下管线的管理业务流程。信息系统通过确认测试后, 展开全市范围的地下管线普查勘测, 并完成数据处理和入库的工作。

参与该项目除了发包方、承建方、监理方, 还有软硬件平台建设的分包方、管线勘测单位、管线数据使用单位、数据库基础资料提供单位、管线建设单位等多个干系人。他们各自处于不同的专业和行业环境, 完成不同的工作内容, 但是每一个干系人的作为和态度都对项目的开发具有重要的影响。在项目的每一个迭代阶段中都必须充分地调动各方的资源, 组织良好的多方协调配合, 才能保证项目的顺利开展。这其中监理就起到了重要的组织协调作用。可以说在多方之间良好地组织协调, 是促进项目的进展、实现项目目标的最重要的方法。因此, 我在项目的启动阶段就充分地认识到组织协调的重要性和存在的困难, 并采取了如下的措施。

一、参照 UML 的方法分析项目活动过程

由于该项目不仅仅是软件开发, 而且还包括有应用集成、工程勘测、数据生产等内容。并且在项目的开发过程中要新建一套政府对地下管线实施管理的机制和数据动态更新的业务流程, 其中涉及到多方利益关系的协调。因此项目的管理过程具有复杂性和独特性。为了应对这一情况, 我们在编制监理规划的时候就认为必须详细地分析整个项目的管理和实施的过程, 根据项目的特点建立一套项目活动过程的模型, 解释项目各方干系人在项目活动中的相互关系和行为方式。这一模型最重要的要求和作用是要让各方都能够充分地理解模型表达的内涵, 进而接受和确认这种关系和行为要求。以便建立起共同的项目目标和协调一致的行为准则。

我们参照了 UML 的分析方法和表达方法, 设计项目管理的模型。比如, 使用顺序图和协作图表示各方之间消息传递的顺序和关系; 使用一系列带泳道的活动图表示出各项活动的过程中各方之间的联系和流程; 最重要的是, 建立信息管理和文档管理的模型, 通过类图定义了针对 GIS 工程的各种技术文档和管理文档的属性、功能要求, 设计这些文档在本项目过程中产生和使用的活动图, 以及相互联系。

在监理规划对项目管理模型分析的基础上, 我们得以在监理实施计划中制定具体的信息管理、信息通信、配置管理、组织协调的有效机制。

二、建立有效的信息管理机制

信息管理机制是保证我们设计的项目管理 UML 模型得以实现的重要手段。我们规定三方的工作人员都要按照 UML 模型中的要求编制文档, 并提交给信息管理员编号集中归档。每一个项目阶段或者里程碑的检查验收都以是否完成文档编制要求作为第一指标。未能完成文档要求的不予签署监理意见, 总监还要追求专业监理工程师和监理员的责任。

另外, 我们也制定了信息管理员需要及时发布的信息和文档类别。包括监理周报、会议纪要、监理专题报告、项目变更文件等文档由信息管理员立即发布到三方的各个小组联系人的邮箱中。

我们使用 Microsoft Project 建立了项目管理的网站, 作为资源共享和信息通信的平台。三方以及工程勘测单位都可以及时地发布消息, 并且了解项目的进展情况, 增进沟通协调。

在本项目中, 工程勘测单位与信息系统承建方之间没有直接的合同关系, 各自只需要对发包方负责, 但是在实际的开发中, 两者之间的联系是非常关键的环节。因此我们需要着力协调两方之间的沟通和通信。在信息传递的活动图上, 勘测单位也是其中一个执行者。一些重要的环节, 比如数据标准和原型的评测, 更需要两方密切合作。因此我们在发包方的授权下, 协调监督两方之间的协调配合。

一、本着公平公正独立的原则组织协调

我们一方面必须以科学、规范的方法组织项目的监理工作, 以得到项目有关各方的认可。还必须以公平、公正、独立的原则在多方之间组织协调, 以我们诚恳、守信的处事态度和实际的行动获取合作方的信任和支持。

在**市还没有启动本项目的时候, 地下管线的管理实际上是采用了一种不规范的流程。某工程勘测院就处在流程的一个环节中。项目实施过程中, 按照新的管理模式, 该工程勘测院应当完全退出管理的环节, 并移交所有的历史资料。这时该工程勘测院对管理模式以及本项目提出了一些意见和建议。但是遭到发包方的反对, 引发了矛盾。我们为了促成项目的成功, 从技术的角度独立公正地分析了这些意见和建议, 有理有据地指出其中合理的和不合理的部分, 为发包方提供了参考意见并从中协调。发包方最终接受了部分的建议解决了矛盾。

在监理过程中的组织协调包括有项目参与人之间的协调和项目外部因素的协调, 有合同因素的协调和非合同因素的协调。比如以上的例子中的协调工作就超出了监理合同的范围。但是很多意外的因素都有可能影响项目的质量和效果, 因此必须加以积极的引导。

我们规定每周组织一次监理会议, 会上主要讨论监理报告、研究项目实施中的问题。并且要求专业监理工程师在会上报告与发包方和承建方人际沟通的情况, 评价项目管理的过程是否按照我们设计的模型运作。

在本监理项目中, 由于采用了以上的方法, 使得发包、承建、监理三方以及工程勘测单位等形成了较好的合作关系, 使项目得以顺利完成, 并且达到预定的质量和进度要求。

论软件开发成本管理

摘要

2004 年 8 月, 我作为项目经理开始参与某某银行授信业务系统的开发项目, 主要工作职责为需求分析、系统设计和项目管理。系统基本功能包括: 业务操作、业务提醒、基础资料、查询统计和权限管理等五个模块。系统采用 Struts+Hibernate 主流 Web 应用框架, 实现 Web 应用程序服务器 WebSphere 与协作应用程序服务器 Lotus Domino 的高度集成。

项目的成功很大程度上归功于在项目过程中各个阶段对进度和成本的有效管理和控制。本文以该项目为例, 结合作者实践, 讨论了信息系统项目中的成本管理问题, 主要通过计划在计划阶段做好工作量估算, 有效管理和控制风险因素, 在实施阶段进行成本跟踪和控制等方法来有效管理和控制项目成本。实施结果....

正文

2004 年 8 月, 我作为项目经理开始参与某某银行授信业务系统的开发项目, 主要工作职责为需求分析、系统设计和项目管理。当然也做一些编码工作, 主要是基础性公用代码和关键核心代码的编写与维护。授信是指银行以自身信用向客户提供贷款(包括项目贷款)、担保、开票信用证、汇票承兑等业务, 授信业务是商业银行资金运作中最为重要的业务之一。开发授信业务系统, 提高授信业务的管理水平和运行效率、充分利用共享的信息资源、减小各种风险、运用各种科学的金融分析模型指导业务开展具有十分重要的意义。系统基本功能包括: 业务操作、业务提醒、基础资料、查询统计和权限管理等五个模块。系统全面实现授信业务的网上操作, 实现流程的上报, 审批和管理, 大大提高了授信业务工作效率。提供了强大的业务查询和统计功能, 便于对授信业务工作的管理和监督。其中业务操作模块实现授信业务工作流程, 主要包括正常类授信业务申报、问题类授信业务申报、特殊类授信业务申报和授信后监控业务等工作流程。

系统采用 Struts+Hibernate 主流 Web 应用框架, 开发工具采用 WebSphere Studio Application Developer 5.0 (WSAD 5.0), WSAD5.0 集成并扩展了 Eclipse 2.0 的功能。硬件配置方面: IBM P610 小型机用于安装 WebSphere 5.0, DELL 服务器用于安装 Domino R6 和 SQL Server 2000。实现 Web 应用程序服务器 WebSphere 与协作应用程序服务器 Lotus Domino 的高度集成, 并使用 Single Sign On(SSO)实现单点登陆。总体架构思想, 将表单数据的生成和分析采用关系型数据库来实现, 通过 WebSphere 架构实现业务逻辑的处理, 而表单的审核流程由 Domino 进行驱动。将基于业务为主的 J2EE 服务系统和基于协作为主的 DOMINO 流程处理系统有效的结合起来, 确保整个业务流程的有效运行和各种数据查询分析统计的有机结合。

由于考虑到银行帐户年度等因素, 客户要求系统在 2004 年 12 月底前交付, 项目开发周期为 4 个月。项目人员配备情况, 项目经理 1 人, 开发人员 4 人, 测试人员 3 人, 界面美工人员 1 人, 项目行政秘书 1 人, 配置管理人员 1 人, 质量管理人员 1 人。其中开发人员小张来自某某银行科技处。项目行政秘书、配置管理、质量管理等人员为兼职人员, 为多项目共享。由于公司属于大型软件企业, 在项目基础设施方面包括开发服务器、开发机、测试服务器、配置管理服务器、开发工具等配备状况较好。

软件成本管理是软件项目管理的一个重要组成部分, 也是一个十分容易被忽视但却又是十分重要的内容。成本管理的目的是通过执行项目成本管理过程和使用一些基本项目管理工具和技术来改进项目成本绩效。项目组整体上把按进度和预算交付项目作为我们最大的挑战, 因此我们十分重视对项目进度和成本的控制和管理。该项目中我们借助项目管理软件 Microsoft Project 2003 来辅助进度和成本的计划和管理。我们主要通过计划在计划阶段做好工作量估算, 有效管理和控制风险因素和在实施阶段进行成本跟踪和控制等方法和策略来有效管理和控制项目成本。

1、计划阶段做好活动历时(工作量)估算

项目需求分析阶段结束, 《软件需求说明书》得到客户正式签字确认后, 我们开始创建工作分解结构 WBS 和制定详细项目进度计划。我们认为工作量估算是成本估算的基础, 对于项目成本管理十分关键。由于对代码行(LOC)估算、功能点(FP)估算等估算方式研究不是很深入, 工作量估算主要采用基于公司项目历史绩效数据库和个人经验的估算方法。对于部分涉及流程的活动单位一般比较难一次性把握其活动的历时, 事实上流程调试的工作量在页面基本功能(增加/删除/修改)的 3 倍工作量以上。例如业务操作模块——问题类授信业务申报——问题类客户行动计划申请流程页面提交工作量为 2 日/人, 而流程调试需要涉及 20 多个角色和 8 条路径。对于估算把握不是很好的任务, 我们一般通过提供一个乐观估算 A、悲观估算 B、正常估算 M 进行 3 次估算然后利用 PERT 公式 $[1(4*M+A+B)/6]$ 计算取整。每项活动我都先确定具体人员, 然后需要对活动本身进行详细分析, 必要时查看公司项目历史绩效数据库。最后需要为各项活动建立了依赖关系, 明确各项活动的前置任务, 活动开始时间和结束时间。总体上讲活动历时估算工作量较大, 我花费了数个工作日。

项目组人员流动率较低, 在 J2EE 和 Struts 架构下的 WEB 应用开发已经有一定的项目积累和团队合作基础。如项目组自行开发了功能完善的 Struts-config.xml 统一维护工具, 实现了 FormBean 和 ActionBean 方便管理。有大量可供复用的东西, 如公共基础代码包, 权限管理模块等。这些也是在我们工作量估算中需要考虑的因素。

2、有效管理和控制风险因素

项目中我们对项目风险进行了必要的管理, 以避免风险事件的发生引发项目成本增加或超支。公司项目管理部门提供了风险管理计划的模板和风险事件列表模板。为了让项目组整体在各个阶段保持良好的风险意识, 我尝试采用了“十大风险事项跟踪”, 把项目中各主要风险事项按照排名张贴在公告栏上。由于当时有部分未明晰的需求包括: ①问题类客户行动计划申请流程; ②查询统计部分需求; ③客户方面可能提出的新需求。需求和范围界定不清、计划不充分、用户参与不足、缺乏领导支持、技术问题等为我们项目计划阶段主要风险事件。事实表明, 这种做法效果是非常明显的。特别是客户方面, 我定期把风险事件列表 Email 给客户方项目负责人方某。为了能尽快落实未明晰的需求部分, 我与客户方主要项目负责人方某进行了面对面的沟通。通过一番利弊关系的陈述, 达成尽快明晰悬留部分需求的共识。需求问题很快得到解决。项目组整体信心十足, 积极性和责任感增加。公司领导方面对项目组也表现出特别的关心, 特别是公司赵总开始频繁出现在项目组的每周进度评审会议上, 他们也开始担心因为对项目支持不够而导致项目的失败。

3、实施阶段进行成本跟踪和控制

实施阶段需要进行成本的跟踪和控制。Project 2003 中需要设定各项资源(人员)的工时标准费率, 即人员每小时的工作成本。项目组成员每周五下班前通过内网 B/S 项目管理信息系统 PMIS 提交《项目周报》, 把各自本周内完成的任务进度情况和下周任务计划进行汇报。报告要求按百分比严格量化任务完成情况, PMIS 只提供具体百分比的选择。项目经理(我)把各项任务实际完成数据输入到进度计划中, Project 2003 自动成本统计表, 清楚显示任务基准和实际成本信息。通过查看跟踪甘特图就可以较好把握项目总体的进度绩效。

授信业务系统在 2004 年 12 月下旬正式上线, 提前 1 周完成了项目。目前系统运行正常, 受到客户方各有关部门的一致好评, 对项目满意度较高。项目的成功很大程度上归功于在项目过程中各个阶段对进度和成本的有效管理和控制。没有成本管理, 项目也可能成功。但没有成本管理的项目, 对于项目管理质量、时间、成本三大目标的实现是具有巨大风险。

论企业级信息系统项目管理体系的建立

摘要

2004 年 8 月，我作为项目经理开始参与某某银行授信业务系统的开发项目，主要工作职责为需求分析、系统设计和项目管理。系统基本功能包括：业务操作、业务提醒、基础资料、查询统计和权限管理等五个模块。系统采用 Struts+Hibernate 主流 Web 应用框架，实现 Web 应用程序服务器 WebSphere 与协作应用程序服务器 Lotus Domino 的高度集成。本文结合作者项目实践，以该项目为例，讨论了如何建立企业级信息系统项目管理体系的问题，第一阶段工作重点为普及项目管理知识，第二阶段工作重点为建立项目管理体系，推行“项目经理负责制”。后继阶段的工作重点为项目管理改进与提高。项目管理体系的确立为项目提供了良好的支持环境。

正文

2004 年 8 月，我作为项目经理开始参与某某银行授信业务系统的开发项目，主要工作职责为需求分析、系统设计和项目管理。当然也做一些编码工作，主要是基础性公用代码和关键核心代码的编写与维护。授信是指银行以自身信用向客户提供贷款（包括项目贷款）、担保、开票信用证、汇票承兑等业务，授信业务是商业银行资金运作中最为重要的业务之一。开发授信业务系统，提高授信业务的管理水平和运行效率、充分利用共享的信息资源、减小各种风险、运用各种科学的金融分析模型指导业务开展具有十分重要的意义。系统基本功能包括：业务操作、业务提醒、基础资料、查询统计和权限管理等五个模块。系统全面实现授信业务的网上操作，实现流程的上报，审批和管理，大大提高了授信业务工作效率。提供了强大的业务查询和统计功能，便于对授信业务工作的管理和监督。其中业务操作模块实现授信业务工作流程，主要包括正常类授信业务申报、问题类授信业务申报、特殊类授信业务申报和授信后监控业务等工作流程。

系统采用 Struts+Hibernate 主流 Web 应用框架，开发工具采用 WebSphere Studio Application Developer 5.0 (WSAD 5.0)，WSAD5.0 集成并扩展了 Eclipse 2.0 的功能。硬件配置方面：IBM P610 小型机用于安装 WebSphere 5.0，DELL 服务器用于安装 Domino R6 和 SQL Server 2000。实现 Web 应用程序服务器 WebSphere 与协作应用程序服务器 Lotus Domino 的高度集成，并使用 Single Sign On(SSO)实现单点登陆。总体架构思想，将表单数据的生成和分析采用关系型数据库来实现，通过 WebSphere 架构实现业务逻辑的处理，而表单的审核流程由 Domino 进行驱动。将基于业务为主的 J2EE 服务系统和基于协作为主的 DOMINO 流程处理系统有效的结合起来，确保整个业务流程的有效运行和各种数据查询分析统计的有机结合。

当时公司为某大型软件（上市）公司在该省的分公司和研发中心，成立不到一年。企业管理和项目管理总体上比较混乱，项目管理中出现了很多问题如项目管理缺乏系统性，项目管理体系不健全，项目风险防范意识缺乏，不注重项目经验总结，项目组之间缺乏合作与交流等。项目组间人员流动频繁，甚至出现分配下去的任务完成一半就改派到一个新的项目组的情况。部分人员同时参与两个或多个项目，双重甚至多重领导。用我们比较形象的说法是“消防队一一救火”，管理随意性大，员工整体缺乏稳定的工作环境。为改变这一糟糕的现状，我向公司高层领导提出了建立和完善企业项目管理体系的建议，逐步建立与推行“项目经理负责制”。

对项目管理知识体系的良好把握和过去几年项目实践中积累的丰富经验，使我很快说服了公司领导赵总。当天召开了一次由公司项目经理以上职位人员参加的高效率会议，赵总主持了会议。会上大家总结了公司项目管理中存在的一些问题，对在公司内建立和推行项目管理体系有了很好的认识，很快达成了一致意见。最后落实了人员和责任，由公司技术总监陈某总体负责该项工作，小刘负责“项目经理负责制”相关制度和文档的起草，我负责该项工作的进度计划控制和人员安排等。高层领导的支持和参与对项目管理体系在公司内的成功建立意义重大。

第一阶段工作重点为普及项目管理知识，通过周末集中培训在公司范围内普及项目管理知识。考虑到成本问题，我们采用外派培训的方式。由于我以前参加过 PMI 的项目管理知识培训与认证又是当初工作的提议人，所以就被选中参加了一次为期 3 天的软件企业项目管理方面的专业培训。当然对我来说外派回来任务很重，要对全公司的人员进行 2 天的培训。我对公司研发人员和行政人员进行了分析，发现很多研发人员进入公司前是又项目管理意识的，但慢慢就被整个混乱的组织环境同化。我对培训内容做了 5 天充分的准备，采用多选择图片、多举例子、多讲案例的方式。尽可能增加课程的实质性内容，以项目范围管理、时间管理、费用管理、质量管理为重点，也涉及风险管理、人力资源管理、采购管理、沟通管理和项目整体管理等知识领域。事实证明那次培训是很成功的，公司员工参与程度比较高，很多员工当场提出了很多实际问题和一些好的建议。在我看来，课程最后的一段时间就是一次集思广益的“头脑风暴会议”。

第二阶段工作重点为建立项目管理体系，即“项目经理负责制”。公司整体有了基本的项目管理知识，就正式开始推行“项目经理负责制”。首先是对组织机构进行了调整，我们采用了项目型组织结构，以避免之前双重甚至多重领导而导致管理纷乱的局面。项目经理对项目组人员控制权加大，项目组人员受项目经理单独领导，明确规定部门经理不能对各项目组人员随意调动。如测试部门人员安排到项目的测试阶段，受项目经理的全权控制，测试部经理对相关人员没有控制权。部门存在的意义只是方便相关人员技术交流和探讨。公司设立了项目管理办公室（PMO），主要职能包括：①提供公司项目管理的支撑环境；②公司内多项目的管理与监督；③项目管理制度、规范、文档模板等的建立与维护；④人员绩效考核与绩效数据库的建立与维护；⑤提高公司整体的项目管理能力。我作为项目管理办公室成员主要职责为各项目管理制度、规范、文档模板等的建立与维护，我把项目组中采用的比较完整的《项目管理计划》、《工作分解结构 WBS》《进度计划》等文档模板作了适当的修订供公司各项目组共享。PMO 参考软件产品生命周期模型—ISO 15504 建立了项目过程模型，对模型中主要过程包括软件开发过程、支持过程、组织过程进行了完整的定义。确立了项目组成员、项目经理、项目管理层（PMO）、项目客户四级项目管理控制体系，明确项目范围、进度与费用控制权限在不同角色之间的分配。

后继阶段的工作重点为项目管理改进与提高。公司在形成基本的项目管理体系后，也特别注重实际项目经验及项目历史数据的积累，及时优化项目管理流程与规范。聘请了外部的项目管理顾问帮助诊断现行项目管理中存在的问题并提供了有价值的意见。

建立了项目管理体系后，公司总体项目管理状况有了较大的改善。各项目组团队凝聚力较强，混乱状态逐渐消失。企业级项目管理体系的建设需要一个逐步完善的过程，企业在关注项目管理体系建设的同时，应注重项目管理知识的普及、人员培养与项目经验积累，逐步提高组织的项目管理水平和能力，最终提高项目管理绩效。

论项目中的质量管理与企业质量管理的异同

摘要

2004 年 8 月，我作为项目经理开始参与某某银行授信业务系统的开发项目，主要工作职责为需求分析、系统设计和项目管理。系统基本功能包括：业务操作、业务提醒、基础资料、查询统计和权限管理等五个模块。系统采用 Struts+Hibernate 主流 Web 应用框架，实现 Web 应用程序服务器 WebSphere 与协作应用程序服务器 Lotus Domino 的高度集成。本文结合作者项目实践，以该项目为例，讨论了信息系统项目中的质量管理问题，主要包括配置管理、技术评审和软件测试等质量保证方法和策略。项目的成功很大程度上归功于在项目过程中各个阶段对质量的有效管理和控制。文章最后结合 ISO 和 CMM 分析了信息系统开发中的质量管理与企业质量管理联系与区别。

正文

2004 年 8 月，我作为项目经理开始参与某某银行授信业务系统的开发项目，主要工作职责为需求分析、系统设计和项目管理。当然也做一些编码工作，主要是基础性公用代码和关键核心代码的编写与维护。授信是指银行以自身信用向客户提供贷款（包括项目贷款）、担保、开票信用证、汇票承兑等业务，授信业务是商业银行资金运作中最为重要的业务之一。开发授信业务系统，提高授信业务的管理水平和运行效率、充分利用共享的信息资源、减小各种风险、运用各种科学的金融分析模型指导业务开展具有十分重要的意义。系统基本功能包括：业务操作、业务提醒、基础资料、查询统计和权限管理等五个模块。系统全面实现授信业务的网上操作，实现流程的上报，审批和管理，大大提高了授信业务工作效率。提供了强大的业务查询和统计功能，便于对授信业务工作的管理和监督。其中业务操作模块实现授信业务工作流程，主要包括正常类授信业务申报、问题类授信业务申报、特殊类授信业务申报和授信后监控业务等工作流程。

系统采用 Struts+Hibernate 主流 Web 应用框架，开发工具采用 WebSphere Studio Application Developer 5.0 (WSAD 5.0)，WSAD5.0 集成并扩展了 Eclipse 2.0 的功能。硬件配置方面：IBM P610 小型机用于安装 WebSphere 5.0，DELL 服务器用于安装 Domino R6 和 SQL Server 2000。实现 Web 应用程序服务器 WebSphere 与协作应用程序服务器 Lotus Domino 的高度集成，并使用 Single Sign On(SSO)实现单点登陆。总体架构思想，将表单数据的生成和分析采用关系型数据库来实现，通过 WebSphere 架构实现业务逻辑的处理，而表单的审核流程由 Domino 进行驱动。将基于业务为主的 J2EE 服务系统和基于协作为主的 DOMINO 流程处理系统有效的结合起来，确保整个业务流程的有效运行和各种数据查询分析统计的有机结合。

在该项目中，我们主要通过配置管理、技术评审和软件测试等质量保证方法和策略来保证系统的质量。

1、加强配置管理

配置管理是项目中实施软件质量管理的关键，公司为项目组配备了一名兼职配置管理员辅助我完成配置管理工作。同时根据公司项目管理部门的要求项目设立了变更控制委员会 CCB，主要由技术总监、我（项目经理）、用户代表、质量管理员、配置管理员等 5 人组成。在公司里变更控制委员会是处于尝试阶段新生事物，所以具体的配置管理和变更控制主要由我和配置管理员完成。公司主要采用 Visual SourceSafe 6.0(VSS 6.0)作为配置管理的工具，主要考虑到 VSS 操作简单，界面友好，版本控制功能强大和比较常用等原因。项目一启动，我们就在公司配置管理服务器上建立了项目开发库，开发库主要供项目组开发人员使用，变更比较频繁。我为项目组每个成员建立了访问帐户和权限，项目开发库主要由我负责管理和控制。配置管理员负责管理和控制项目受控库，用于保存项目各阶段结束时发布的阶段性文档。如需求分析阶段结束《软件需求规格说明书》，概要设计阶段《概要设计说明书》等。

2、定期开展技术评审

项目各阶段进行必要的技术评审活动，如在软件需求分析阶段结束后进行软件需求评审，以确保在《软件需求规格说明书》中所规定的各项需求的合适性。详细设计阶段制定了 java 文件包的命名规则，如问题类客户行动计划申请相关类包含于 com.opr.question.plan 以下，其下统一细分为 Dao、Action、ActionForm 相关类。这些规则必须在编码阶段得到很好的遵循和执行。评审活动一般由项目经理（我）和质量管理人员分别进行。项目组每周五下午召开进度审查会议，对本周各任务完成情况进行总结。对已完成的任务进行验收，验收包括对功能和代码质量的总体把关，对出现进度延迟的任务进行分析及时解决问题。如评审中发现小部完成的正常类授信业务——完全现金保证额度申请功能没有采用项目组规定的统一的页面处理方式，即先提交数据后根据返回的状态标识或异常情况统一关闭提交页面或返回到错误提示页面。这种页面处理方式的优点是当用户数据提交不成功时，已经输入信息的不会丢失，同时给予必要的提示。实践表明，即使在数据提交前页面校验很完整的情况下，也存在一定出现提交不成功事件发生的概率。特别是在财政、金融、银行等行业的应用系统中一般页面输入数据量较大，出现数据提交不成功而需要第二次重新输入将严重影响用户对系统的兴趣和态度。我们认为健壮性和可使用性是比较关键的质量指标。

3、充分做好测试工作

测试作为保证软件质量的重要手段，在该项目中地位非常明显。编码阶段我们主要进行单元测试，采用编码人员自己测试和项目组内部交叉测试向接合的方式。如正常类授信业务模块与问题类授信业务模块的编码人员相互交换进行测试。交叉测试的优点在于能够发现很多编码人员自己测试自己的程序不能发现的问题，所谓“旁观者清”！集成测试和系统测试由测试小组单独负责，主要采用测试管理工具 IBM Rational TestManager 和有“最佳功能测试工具”之称的功能测试工具 Rational Robot。测试人员负责测试脚本的编写，Rational Robot 根据事先记录的页面数据输入（或操作），执行脚本自动实现功能测试。测试人员把测试中发现的 Bug 和问题提交到 TestManager，由 TestManager 负责统一管理和控制。项目经理（我）对各 Bug 进行分类并设定修复优先级，分配给具体项目组编码人员进行修复。编码人员修复 Bug 后，通过 TestManager 反馈修复信息给测试人员，测试人员进行测试直到没有发现问题。TestManager 可以用来从测试计划、测试设计、测试实现、测试执行和结果分析等多个方面进行测试管理，管理和控制功能将大，还提供图形化查询和统计图表。对于基于 B/S GUI 技术的应用程序自动进行功能和性能测试，Rational Robot 是主流的和推荐的选择。TestManager 和 Rational Robot 等测试工具的综合运用大大提高了项目组的测试效率，有效和有力保证的软件系统的质量。

授信业务系统在 2004 年 12 月下旬正式上线，提前 1 周完成了项目。目前系统运行正常，受到客户方各有关部门的一致好评，对项目满意度较高。项目的成功很大程度上归功于在项目过程中各个阶段对质量的有效管理和控制。没有质量管理，项目也可能成功。但没有质量管理的项目，无论从质量、时间、成本任何一个方面看都是有巨大风险的。

信息系统开发中的质量管理与企业质量管理联系比较密切但两者的区别也是很明显的，两者不可简单的互相替代。前者属于项目管理层面，而后者属于企业管理层面。信息系统开发中的质量管理重点是通过执行必要的质量管理过程如质量计划、质量保证和质量控制等来保证项目最终交付的成果对客户需求和期望的一致性，主要涉及软件研发过程中的细节内容，如必要的配置管理、单元测试活动等。如 CMM 强调的是软件开发过程的管理。而企业质量管理的任务是企业级质量管理体系的建立、管理与维护，主要涉及企业质量规范、制度的制定和宏观支撑环境等内容。如 ISO 9001 侧重于“机构保证在设计、开发、生产、安装及服务过程中与指定的要求一致”。CMM 和 ISO 9001 两者的最大相似之处在于两者都强调“该说的要说到，说到的要做到”。对每一个重要的过程应形成文件，包括指导书和说明，并检查交货质量水平。两者都强调持续改进。

论软件开发的风险管理

摘要:

本文讨论了某公司实施 SAP 系统的风险管理。该公司原先运行着一套 ERP 系统，现在要转到 SAP 上，需要完成新系统的流程的重新定义，数据的切换，用户的培训等工作。项目要求在 11 个月的时间内完成。实施一个大型的 ERP 系统有着各种的风险，这些风险如果不加分析和控制，将会给整个项目造成致命的影响。我作为项目经理，主要从控制进度风险，人员流动风险和系统功能风险三个方面去进行风险的管理。最后这三方面的风险都得到了有效的控制，从而使项目顺利完成。

正文:

2003 年 1 月，我参与了西门子集团下某公司的 SAP 系统的实施，提任项目经理。该公司之前运行着另一套 ERP 软件：QAD 的 MFG/PRO 系统。由于集团总部的要求，要用 SAP 系统替换原先的 MFG/PRO 系统，并且要在 2003 年 11 月前完成。整个项目完成以下阶段，首先是项目的引进，包括成立项目小组，由顾问对项目小组成员进行初步的培训，让小组成员对 SAP 的标准流程有个大概的认识。接下来是要分模块进行讨论，制定出各模块的实施蓝图 (blueprint)。该公司实施了以下的模块：SD (销售与分销)，MM (物料管理)，CO (成本控制)，QM (质量管理)，PP (生产控制)，FI (财务核算)，CO (成本控制) 等。在 Blueprint 完成后，由顾问根据定下的流程配置一个测试的系统，用户在该测试环境下进行练习和测试。测试完成后就是数据的准备和切换了，要从 MFG/PRO 系统把需要的数据下载下来然后你上传到 SAP 系统。完成数据的切换，SAP 系统正式上线，同时不再使用原先的系统。

因为整个项目要在 11 个月的时间内完成，时间是非常紧迫的。如何在如此短的时间内使项目能顺利进行，控制各种可能出现的风险是必要的。为此，在项目的初始阶段，我召开了小组成员开会，专门针对项目的风险进行了讨论。会上，大家把想得到的风险都提了出来。经过分析筛选，我最后确定了三个重点进行控制的风险，并采取了相应的措施进行控制。

1、控制进度风险

ERP 的实施是一个大型的项目，涉及到企业的流程改造和其它方方面面的东西。而该项目的上线时间不能改变，所以，项目进度在这里是个潜在的风险，如果不能如期上线，则公司将的运作将会受到重大的影响。为此，我在项目启动后，召集了项目小组成员开会制定项目计划。我首先用 Microsoft Project 制定了项目的总体计划，在这个总体计划中，明确了各个阶段的任务和完成时间。如什么时候完成设计蓝图 (Blueprint)，什么时候进行关键用户的培训，什么时候进行测试和练习，什么时候进行数据的切换等。总体计划制定出来后，各顾问在总体计划下制定各模块的实施计划，把每个模块在每一阶段的问题细化。各模块的实施计划要在总体计划的基础上进行，在任务和时间上不能滞后于总体计划。无论是总体计划和实施计划，都要求明确各步完成的时间，要精确到哪一天，而不能用模糊的描述，比如“3 月初完成 Blueprint”这样的低描述是不允许的。为保证项目照进度进行，每周五都要召开项目会议，检讨项目的进展情况，发现有超期的任务，分析原因，及时解决。在进度的控制方面，还要应付突发的事件造成的影响，及时地调整计划以适应新的情况。2003 年 4 月—5 月，由于受到“非典”的影响，外部顾问不能出差来我公司，这使项目的进度受到了很大的影响。为了把这种影响降到最低，我即时调整了项目计划，把这段时间安排为关键用户的培训和对系统的熟悉。因为之前顾问已对关键用户进行了一些培训，所以用户对 SAP 系统有了一定的认识。通过上机的操作，更进一步了解系统。有问题我们通过 E-Mail 与顾问联系。通过这种方法，把原本以后进行的用户练习提到了前面。虽然没有顾问的现场指导，但通过自己的摸索对系统的印象更加深刻，为之后的工作打好的基础。

2、控制人员流失的风险

在实施 SAP 过程中，有两种可能会导致人员的离职，一是工作繁重乏味，压力大；二是积累了一定的 SAP 经验后找工作相对比较容易，这时如果有其它更好的机会，员工会考虑跳槽。而项目小组人员流动将会对项目造成很大的影响，甚至导致项目的失败。所以如何控制这些风险，是作为项目经理要考虑的问题。为此，我首先与人力资源部一起，制定了一套有效的激励机制。包括，把参加 SAP 项目作为年终的一个绩效考核内容；因为项目需要加班加点的，公司免费提供晚餐；设立项目基金，对表现突出的小组成员进行物质上的奖励；在整个公司的范围内大力宣传 SAP，让项目小组成员感觉到 SAP 的重要性和实施项目的价值等。其次，为了避免出现某个项目小组成员离职导致项目不能进展的情况出现，我在每个模块都安排了两个人员负责，一个是主负责，一个是次要负责。这样，可以减少人员流失造成的损失。在这里，我没有采用让项目小组成员签合同的方法，即实施项目后要在公司工作多少年。我觉得这种方法会给小组一种压抑的感觉，而公司主要要靠企业文化来吸引员工。事实证明，我所采用的措施是有效的：在整个项目的实施过程中，没有一个小组成员退出或是离职。在项目运行一年多来，只有两个当时的关键用户离职，而他们离职后后备人员可以马上顶上来，对系统没有造成什么影响。

3、控制系统的功能

有人戏称 SAP 是“Stop All Production”，这也从一个方面反映了实施 SAP 的风险所在。而造成上了系统后停产的一个原因就是系统功能不能满足物流和生产的需要，这也是我担心的一个问题。因为切换后旧系统不能再使用，如果这时候新系统满足不了需要，就真的会造成停产。为此，我在项目中采用了演化型的原型开发方法，用演化型的开发方法，可以让用户针对已配置好的原型进行测试，发现不能实现的功能及时提出来，改进后再测试，再改进。在这里，测试工作显得很重要。为此，我强调要一定要做好测试工作。在系统测试阶段，我把所有的项目成员集中在会议室中进行系统测试。在测试中，我要求用真实的数据，模拟真实的环境进行。系统测试通过后，我还特别做了一次上线前的演习，即把所有相关的数据都导致 SAP，配置一个上线后要用的系统，在此系统上进行操作。这次演习成功后，坚定了大家使用系统的信心。系统上线后也没有出现什么大的问题。

通过以上措施，使把 SAP 的主要风险基本上都控制在萌芽状态，项目没有因为这些风险受到影响，最后项目如期上线，受到了管理层和用户的肯定。

在项目进行过程中，有些风险并没有事先预计出来。比如说顾问的问题。有的顾问水平高，但项目也多。用在我们这个项目上的时间就比较少，这给项目带来了不利影响。为此，我通过和该项顾问所在的公司签定合同，注明顾问在我们公司的工作时间，否则将属违约，通过这一方式使顾问的管理得到改善。

论信息系统开发的变更控制

摘要

2004 年 8 月，我作为项目经理开始参与某某银行授信业务系统的开发项目，主要工作职责为需求分析、系统设计和项目管理。系统基本功能包括：业务操作、业务提醒、基础资料、查询统计和权限管理等五个模块。系统采用了 Web 应用程序服务器 WebSphere 与协作应用程序服务器 Lotus Domino 的高度集成的技术构架，基于 Struts 架构实现 MVC 模式。信息系统项目由于目标不明确、需求变化频繁等原因，开发过程中发生变更在受难免。为了保证项目有序进行，实施强有力的变更控制和管理非常必要。本文对项目中采用的变更管理方法、措施和工具 VSS，以及变更控制的主要步骤进行了较详细的介绍。最后对项目变更管理过程的一些实际问题进行了总结。

正文

2004 年 8 月，我作为项目经理开始参与某某银行授信业务系统的开发项目，主要工作职责为需求分析、系统设计和项目管理。当然也做一些编码工作，主要是基础性公用代码和关键核心代码的编写与维护。授信是指银行以自身信用向客户提供贷款（包括项目贷款）、担保、开票信用证、汇票承兑等业务，授信业务是商业银行资金运作中最为重要的业务之一。开发授信业务系统，提高授信业务的管理水平和运行效率、充分利用共享的信息资源、减小各种风险、运用各种科学的金融分析模型指导业务开展具有十分重要的意义。系统基本功能包括：业务操作、业务提醒、基础资料、查询统计和权限管理等五个模块。系统全面实现授信业务的网上操作，实现流程的上报，审批和管理，大大提高了授信业务工作效率。提供了强大的业务查询和统计功能，便于对授信业务工作的管理和监督。其中业务操作模块实现授信业务工作流程，主要包括正常类授信业务申报、问题类授信业务申报、特殊类授信业务申报和授信后监控业务等工作流程。

授信系统采用了 Web 应用程序服务器 WebSphere 与协作应用程序服务器 Lotus Domino 的高度集成的技术构架，并使用 Single Sign On(SSO)实现单点登陆。总体架构思想，将表单数据的生成和分析采用关系型数据库来实现，通过 WebSphere 架构实现业务逻辑的处理，而表单的审核流程由 Domino 进行驱动。将基于业务为主的 J2EE 服务系统和基于协作为主的 DOMINO 流程处理系统有效的结合起来，确保整个业务流程的有效运行和各种数据查询分析统计的有机结合。系统基于 Struts 架构实现 MVC 模式。

考虑到银行帐户年度等因素，客户要求系统在 2004 年 12 月底前交付，项目开发周期为 4 个月。由于我和项目组部分成员都有一些失败项目的经历，所以整体上对项目管理比较重视。需求分析阶段我们的一贯目标是采用文字或图形化（该项目中主要采用 Rational rose 和 Visio 工具）的方式清晰正确地描述了至少 80%的实际需求，并在完成系统设计完成或编码阶段开始前明确剩余需求，高质量的需求分析为项目范围管理奠定了基础。基于对客户需求的良好把握，综合使用类比法、自上而下法和自下而上法创建高质量的工作分解结构（WBS）。通过工作分解结构把项目范围分解开来，使项目组人员对项目一目了然，能够使项目的概况和组成明确、清晰、透明、具体。

进入项目实施编码阶段，可能引发范围变更的需求因素主要为部分未明晰的需求包括：①问题类客户行动计划申请流程：由于该流程涉及风险处、授信处、风审会和贷审会等部门，对于流程是否要简化和参考总行的相关做法存在较大分歧，没有能短时间达成一致意见。②查询统计部分需求：该部分需求系辅助性功能需求，需要综合和汇总全分行各部门及下属支行用户的需求。③客户方面可能提出的新需求。

变更是信息系统开发项目的一个突出特点，也是信息系统项目最为普遍的特点。变更管理的目的是对于复杂、无形的软件，防止在其多变的情况下失控，出现混乱，最终造成工作的损失。因此在该项目中，我们十分重视对项目变更的控制和管理，项目中主要采用了配置库、变更控制委员会、变更评审会议等方法 and 措施。

公司为项目组配备了一名兼职配置管理员辅助我完成配置管理工作。同时根据公司项目管理部门的要求项目设立了变更控制委员会 CCB，主要由技术总监、我（项目经理）、用户代表、质量管理员、配置管理员等 5 人组成。在公司里变更控制委员会是处于尝试阶段新生事物，所以具体的配置管理和变更控制主要由我和配置管理员完成。公司主要采用 Visual SourceSafe 6.0(VSS 6.0)作为配置管理的工具，主要考虑到 VSS 操作简单，界面友好，版本控制功能强大和比较常用等原因。项目一启动，我们就在公司配置管理服务器上建立了项目开发库，开发库主要供项目组开发人员使用，为项目组每个成员建立了访问帐户和权限等，项目开发库主要由我负责管理和控制。配置管理员负责管理和控制项目受控库，用于保存项目各阶段结束时发布的阶段性文档。

当我完成项目管理计划好，配置管理员基于项目管理计划和公司项目管理部门文档维护的模板完成了《配置管理计划》，主要包括了配置项计划、《配置项管理表》、《变更请求申请表》、基线计划、交付计划和备份计划等。提交项目管理部门审批同意，然后召开了一次由项目组全体人员参加的配置管理专题会议，明确工作流程，落实责任人等相关事项。

由于每年最后几个月为银行业务旺季和客户方部分人员交叉出差等原因，需求分析阶段的遗留问题一直没有得到解决。需求说明书也一直没有得到正式签字确认，为了不影响项目总体进度，同时考虑到主要的需求是确定的，项目组直接进入了分析设计阶段和后继的编码阶段后。为了能尽快落实未明晰的需求部分，我与客户方主要项目负责人方某进行了面对面的沟通。通过一番利弊关系的陈述，达成尽快明晰悬留部分需求的共识。当然主要由方某负责内部协调，统一风险处、授信处、风审会和贷审会等部门对问题类客户行动计划申请流程的意见。第二周星期一上午刚上班，就接到了方某的电话，告诉我各部门对问题类客户行动计划申请流程已经达成了一致意见，查询统计部分需求也整理完毕。根据客户方面提供的意见，我对《软件需求说明书》作了必要的修订，并提交客户方项目关键负责人方某正式签字确认。《软件需求说明书》的变更相应的也带来了《概要设计说明书》、《详细设计说明书》和《数据库设计说明书》等关键性项目文档的变更，并对开发库和受控库内相应文档进行了版本升级。

2004 年 11 月中旬，当主题编码阶段完成进入测试阶段。客户方提出了新的需求，由于在正常类授信业务申报、问题类授信业务申报、特殊类授信业务申报和授信后监控业务等 20 多个工作流程中都涉及客户资料（如《企业法人营业执照》复印件、《税务登记证》复印件、《组织机构代码证》复印件等）的上传，客户要求系统中增加一个功能实现对扫描仪的直接调用。由于考虑到该功能最终用户（各支行客户经理）处扫描仪型号不统一等问题，项目组没有批准这一变更，当然向客户提供了充分的理由。

事实证明，需求分析阶段工作必须要做到位，就好像摩天大楼的地基一定要保质保量。需求分析图形化工具如 Rational rose 和 Visio 等的采用，以及原型方法和快速原型方法的采用将有效地帮助项目组整体对项目范围和产品需求的把握，从而从项目早期就开始尽可能避免不必要的变更。

变更控制委员会存在一些缺点：比如变更的决策过程可能要花费较长的时间。变更控制委员会通过一次会议可能无法做出决策。因此我们对该过程进行了简化，以便可以就次要的项目变更做出快速决策。公司项目管理部门采用了“48 小时政策”，可以由项目小组核心成员在其技能和权限范围内对关键决策或变更达成共识。受决策影响或变更影响严重的人员只有 48 小时向高层管理人员汇报并获得批准。如果某些原因，相关高层经理可以在 48 小时内修改决策，否则项目团队的决策则视为被批准。

VSS 的项目组织管理使得项目组的协调变得简单容易且很直观，当一个和一组文件发放给另一个成员、小组、Web 站点或是任何其他地址，VSS 确保他们之间的真正共享及所选的一组文件的不同版本的安全性。

授信业务系统在 2004 年 12 月下旬正式上线，提前 1 周完成了项目。目前系统运行正常，受到客户方各有关部门的一致好评，对项目满意度较高。项目的成功很大程度上归功于在项目过程中各个阶段对变更的有效管理和控制。没有变更管理，项目也可能成功。但没有变更管理的项目，无论从质量、时间、成本任何一个方面看都是有巨大风险的。

论软件需求分析方法和工具的应用

【摘要】

本文讨论《企业人事信息系统》项目的需求分析方法与工具的选用。该系统的建设目标是帮助该企业管理好企业内部的人员和人员的活动，人事信息管理指的是企业员工从招聘面试到离职退休的全过程，涉及的主要活动包括面试、报到、培训、升职、离职或其他的人事变动，也包括电子化考勤、工资性收入的计算与分发、使用其他公司资源的有关记录（如宿舍、保险、证件办理等等）。此外，本系统也涉及到企业在全国各地的人事信息管理，企业的组织架构的设置，级别与职务管理，人力申请直至人力需求报表，从而形成一个对企业真正有用的人事信息管理应用系统。在本文中首先讨论了选用面向对象方法与工具的主要理由与策略，进一步通过一个简例说明该方法与工具使用的效果，也讨论了使用多种工具与方法在需求分析中的必要性，最后简要小结了选用正确工具与方法的意义和作用。

在项目开展期间，我担任了系统分析、系统设计与数据库管理等大量工作。

【正文】

人事信息管理系统是一个有着广泛应用面的实用性系统，但是，我国各个企业有着自身的体制、机制、特点与不同的要求；在开发这类系统时，系统需求分析是极为重要的一环。在整个分析过程中，我们都采用了面向对象的分析方法，这是因为我们在近几年的实践中已坚信这种方法能够更加有效地表达和描述现实世界。软件要具有适用性和扩展性，就必须更接近于现实世界本身的发展规律。

以一个简单的例子来看，假设要求设计关于引进人才评估的一个系统，按我们过去的做法，先会要求提供给我们一份相关的引进人才评估表，然后依葫芦画瓢地设计相应的表单与界面，在短期来说，这样做是简便而实用的，但并不能够符合现实世界的长远目标，这套设计方法不具有扩展性，因为任何一份评估表的结构都会有可能产生许多改变的。要用面向对象的方法，可以从中提取出表类型、表结构、评分方法以及能考虑继承等各方面的要素，这样就可以保证软件的通用性，可配置性与可维护性。

在工具的选择过程中，我们选择了现在已十分流行的 Rational 系列，包括 Rational Rose、RUP、SoDA 等，为什么选取这个系列工具呢？这是基于我们对软件需求分析目标的看法，我们认为需求分析应当能正确地回答如下的几个关键性问题：

- (1) 用户的需求是否已详尽地被考虑到了？
- (2) 用户理解或明白我们所描述的内容吗？
- (3) 分析是否会和设计相脱节？
- (4) 程序员能明白我们的分析与设计要求吗？等等。

以下对上述几个问题逐一简要地加以说明：

- (1) 详尽地获取用户的需求。

用户的需求可分为显式的需求与隐性的需求，用户的倾向往往只顾及到当前的与明显的需求。要达到需求理解的全面性，不仅仅只是依靠有效的用户谈话和调查，因为我们所面对的用户需求往往会有些片面的，采用 Rational Rose（基于 UML）提供的用例，以及多种图的联合使用，可以使我们发现其中的遗漏。

- (2) 使用户能充分地理解我们的表示方法，能够真正明白我们描述的内容。

软件需求分析规格说明书通常会冗长而枯燥的，一般的用记不容易深入理解，这样就削弱了分析的正确性。通过支持面向对象及 UML 语言的 RationalRose 可以更好地和用户交流，让用户了解系统的运作方式甚至细节的操作。

- (3) 使分析和设计两个阶段互相联系与贯通。

这是我们选择面向对象的方法及 RationalRose 工具的重要原因，系统分析要向用户描述的不仅仅是用户的需求，而且包括解决方法，解决方法当然应包括设计（程序）、数据库与系统配置，我们当然不希望用户得到的是一个与需求规格说明不相同的软件，也不可能要求程序员完成一个不可胜任的任务。然而我们在以前的多项工作中经常发现这类情节，因为系统分析与设计相互脱节，导致一头扎在分析中不顾设计有关的事宜。

分析与设计的脱节，还不利于设计规格说明的评估，因为分析往往会脱离现实，导致缺乏评估的依据。

因为不可能成功地完成设计而使分析需要重来，就会造成巨大的浪费与损失。一个好的工具可以使分析与设计更紧密地连结起来，甚至于一一对应。面向对象的分析方法使对象之间相对而言有独立性，减少了任何影响到全局的改动，能避免因需求变化而导致全盘皆动的被动局面。

- (4) 使程序员明白我们的设计。

一个好的设计应该让程序员感到清晰明白，更少疑问。一个疑问很多的设计加上沟通不畅，绝对会出现在应用环境下所不需要的另一个软件，所以设计规格说明书务必清楚、形象与明确，当然 RationalRose 有足够的图形与其他形式，能使程序员更加明确，甚至能细做到每一个语句（事实上如果使用 VB，程序架构都有可能直接生成了）。

- (5) 选择 UML 可能会有更多的理由。

比如用户文档的编写、数据库设计，我们都需要做到有延续性，有自动化支持和具有质量上的保证。

所以，我们选用了以上的方法和工具。

在分析中，面对考勤班次的问题时，由于过去一直使用纸卡方式考勤，使用户对班次形成了固定的概念，而现在的许多考勤软件也采用多次刷卡的方法来形成一天的记录。经过面向对象的分析可以发现，事实上每天的上班记录是由多个时段所形成的，时段的多少在各个公司，各个工种与部门都不尽相同，每个时段可能有不同的属性，时段与时段组合可形成为班次，这更适合于现实的情况，使之能更加灵活与更有扩展性。其实，在天与天之间也都有相互之间的关系。在这一点上，我们又发现必须在考勤与薪金工资中加入与 MRP 中相似的期段 (Periods) 的基本概念，比如可以称之为考勤期段，允许为用户更加方便地设置考勤期段，可能使之不一定与自然年月日相同等等。

Rational Rose 使我们更方便地把上面的想法在类上去实现，更进一步地设计好我们的高效率的数据库。

当然，使用单一的一个工具去完成一个中大型的应用系统的需求分析，是不可能成功的，因为社会在发展，用户的需求也在改变，如何把握住用户的需求是需要时间的，而面向对象的方法有时也会忽略外在的与表层的要求，不仅仅是要获得关键的需求，其他更多的需求往往要等到用户在使用后才知识，然而等到用户使用是不现实的，作为原型开发模型中的原型也是收集用户需求，描述与解释需求的一类相当有效的方法与工具。

在我们的开发过程中，为了更好地让用户了解我们的系统和我们的设计方案，让用户在见面会上更有方向性与针对性，我们首先用 Access 开发出原型，让用户先试用。这样，我们在真正的分析与设计时就能更加符合用户的要求。

总之，软件需求分析方法和工具的使用，对我们软件开发过程影响是很深远的，选用高效能的正确的方法与工具，可以使我们的软件更加正确地反映现实需求，更加具有可用性、可扩展性和可维护性，降低了软件项目的风险。

论需求分析方法和工具的应用

【摘要】

本文以某个 IT 产品销售公司的信息系统项目的开发为背景, 讨论了一个信息系统需求分析的整个过程, 其重要特征是: 所涉及的项目是原有系统的一个升级替换版本。因此, 需求分析过程不同于建立一个全新的系统, 大体上可分为三个阶段: (1) 实施逆向工程获得对系统的初步了解; (2) 在第 1 步与客户交流, 最终获得基本需求, 交由客户评审补充; (3) 在第 2 步的基础上开发原型, 利用原型与客户交流, 最终获得基线需求。针对上述三个阶段, 本文论述了所使用的分析方法与工具以及所遇到过的一些典型问题和措施, 最后对需求分析中使用的工具, 谈一些自己的初步体会。

【正文】

我于 1998 年 8 月至 2000 年 7 月参加了某个大型集团的企业信息系统的开发工作, 该大型集团的业务主要涉及到 IT 类产品的进销存。本人在项目中负责系统分析的工作, 该集团企业原先已委托某个电脑公司开发过一套 IT 类产品管理系统, 但是该老系统存在两个主要的问题: (一) 系统运行速度非常慢, 如商品销售开单时, 从确定开单到开单完成有时需要 1~2 分钟左右的响应时间, 让客户无法忍受。(二) 系统数据不准确, 经常出现实物库存与电脑库存严重不相匹配的情况, 使销售数据的统计产生一些混乱, 有关财务的数据因此无法有效使用, 只能采用人工录入方式补充进行。在这种情况下, 该集团的总经理决定参考原有系统重新开发一个系统, 以便解决原系统所存在的上述两个难以克服的难题。注: 原系统要用 PB6.5 开发, 数据库采用 SYBASE, 服务器采用 Windows 2000 Server, 客户端采用 Windows 98。程序架构采用的是传统的 C/S 结构。

鉴于该集团业务操作复杂, 流程多, 涉及人员多等特点, 以及项目完成时间短, 经费有限, 人员有限等限制约束条件, 再考虑到必须避免前一系统出现过的结构混乱与难于维护等问题, 我们决定要对原系统的需求做一个比较彻底的和切实可行的分析, 由于原有系统已经开发了近两年, 并且客户也有了一定的使用经验, 业务基本流程本身也并没有太大的变化, 因此, 我们把需求分析的过程分为三步: (1) 分析原有系统的结构, 主要是数据库结构和程序结构, (2) 在获得第 (1) 步结果的基础上写出基本需求, 交由客户评审补充, (3) 在第 (2) 步的基础上开发原型, 利用此原型与客户交流, 从而获得最终可用的需求结果。下面按上述三步分别加以论述。

第一步是实施逆向工程, 获取原有系统的基本需求。

由于原有系统在功能上大体上能基本满足客户的需求, 并且两年多的开发中也积累了不少经验, 因此, 从中可以获得一些有益的参考, 也可以避免多走弯路。在这一阶段, 我们采用的主要工具是 PB 自带的 Power Designer 和 PB Documents; 前者主要用来分析数据库结构, 后者主要用来分析程序结构, 便于开发人员与高级用户理解程序, 采用这两个工具的原因是: 原系统过于庞大, 模块多, 数据库模式多, 表格量很大, 仅靠人工的方法很难从中获得一个比较完整的、明确的系统结构以及整体构成, 而且原有系统未能提供一套正确完整有效的设计文档, 于是我们只能依靠工具辅助来进行。在使用 Power Designer 分析数据库, 并且 PB Documents 分析原程序中的 PBL 以后, 我们对原系统的结构有了一个初步的了解, 再结合对原系统的使用, 基本明确了功能与流程的需求, 并在此基础上用人工录入方式, 产生了初步需求的自然语言文档。这里指出, 使用 Power Designer 的一个不足之处是: 如果一个表中的字段过多, 而且又同时依赖多个表时, 输出的表格相关图形很复杂, 有很多交叉, 且难于调整, 不方便阅读及打印。

第二步是在第一步的基础上进行的, 即写出系统基本需求, 交由客户评审和补充。

通过第一步的逆向工程, 我们获得了系统的基本需求。为了充分记录需求的变化及需求之间的依赖关系, 我们决定选用 Rational 公司的 Requisite Pro 作为我们的需求管理工具, Rational 公司有一整套用于需求管理的工具, 功能非常强大, 包括 Requisite Pro、Clear Quest 等等, 这些需求分析工具可以对需求进行全面的管理, 包括记录需求的变化情况, 需求之间的依赖关系等等, 但是, 我们考虑到 Rational 的一套工作全面实施会非常昂贵与复杂, 需要非常强的项目管理能力才能全面实施, 因此, 我们只要用了其中最简单的一部分功能, 那就是记录需求变更, 记录需求之间的依赖关系, 其他跟 RUP 关的功能都给略去了。之所以这样做, 主要是考虑到项目的经费人、人力以及国内软件开发的实际情况, 正如前面所说, 我们根据自己的理解并写出基本需求后, 交由客户做评审并做适当补充, 我们将经过补充整理后的需求作为正式需求记录入 Requisite Pro 所维护的数据库中, 并对各个需求进行分类, 设定优先级等, 这些工作完成后, 就可以从数据库中直观地了解客户到现在为止提出了哪些需求, 哪些需求是必须优先考虑的, 哪些是难度较大的等等。在这个过程中, 我们遇到了一些问题, 譬如: 用户对我们用自然语言书写的需求文档有许多地方不理解, 往往在花了较长时间阅读之后, 仍不明白我们所描写的需求过程与他们所完成的业务之间的对应关系, 另外是由于首次采用 Requisite Pro 进行需求管理, 在类型划分, 属性值的在确定上, 部分开发人员没有经验, 造成了不少反复, 对于前者, 我们在方法是想办法增加一些示意图, 将大的流程分解为小流程, 再与客户反复交流与沟通, 最终达到双方理解一致的目的。对第二个问题, 则参考了一些例子, 再结合实际中属性的使用情况, 给予取舍或者选择, 经过这一阶段工作, 我们建立了基本的需求库, 定出了基本需求规格说明。

第三步则是在第二步的基础上建立起原型, 利用原型与客户进行更深入的交流, 通过交流修改相应的需求。

在这一阶段的工作是在对第二步任务进行报告交流的基础上进行的, 我们用 PB 开发了一个原型系统, 就具体的业务流程与客户进行交流与沟通, 通过原型, 客户发现了许多我们与他们的理解相互不协调的地方, 我们在修改需求的同时, 也在 Requisite Pro 需求数据库中记录不修改的历史。事实证明, 这种记录历史的作用是很有效的, 如曾经有客户在两个不同的时间对同一需求提了相反的需求, 我们根据历史记录很快证实了该客户的提法有错误, 在事实面前无需再作争论, 同时利用 Requisite Pro, 我们还发现了一些需求相互之间有矛盾, 经过这一阶段工作, 我们终于获得了经过用户认可的需求基线, 即是可用于下一步进行详细设计的基线需求。

在这个项目中, 我们利用了 Power Designer、PB Documents 等逆向工程分析工具和 Requisite Pro 需求管理工具, 这些工具的使用, 使我们提高了工作效率, 起到了一定的辅助作用。但是, 就需求分析工具方面而言。我们觉得国内应用得还是太少了, 这一方面是因为对需求分析不够重视, 另一方面是因为管理水平还达不到相应的层次。Rational 公司的一整套需求分析工具, 其功能是非常强大的, 国外已在普遍地使用, 在国内也逐渐开始普及, 特别是那些通过 CMM 二级以上评审的单位, 都必须使用工具对需求进行管理。在本项目中, 我们仅仅利用了 Requisite Pro 功能的一些小方面, 已经体会到该工具对于项目管理的诸多好处。如果一个有实力的公司能够全面实施 RUP, 那么需求管理这个老大难的问题会变得不再那么棘手了, 项目的质量也会得到相应的提高。目前国内由于 CMM 热潮的兴起, 已经逐渐重视需求分析, 也逐渐使用需求分析工具, 这是非常可喜的, 当然, 更希望在不久的将来, 能用上国产的需求分析工具, 那时我们的软件产业也许会真正地腾飞了。

论软件需求分析方法和工具的应用

【摘要】

本文以某通信公司的业务报表系统开发为例，讨论了软件需求分析工具与方法的选择。我们认为，软件需求分析是软件工程中重要的一步，直接关系到后继工程进行以及最终的产品能否满足用户的需求，因此在整个工程中起着关键性的作用。采用适当的工具，有可能显著减少需求阶段的错误，也可大幅度提高需求分析的质量和工作效率。当然工具的选择应当与实际的项目相结合，充分地发挥工具的作用。本文结合我们工作的实际经历，简要讨论了开发系统时选用的工具及其应用，选用时所考虑的原则以及所碰到的问题。在文中也结合多种开发方法（即传统的瀑布法、信息工程法、面向对象的方法）的比较，指出各种方法的不足之处，说明我们所采用的工具对软件需求分析所起的作用，以及相应产生的效果。

【正文】

我在某市一家通信公司工作，作为一名技术骨干，受领导委托，参与了开发本公司的业务报表系统，我担任系统的需求分析，总体设计和部分代码的编写工作。

我所在的企业作为一家通信运营公司，分为总部、省级公司和地市级分公司三级，各级公司之间都有数据报表的要求，但是，每一个地市分公司因所处的地方不同，经营环境不同，所面临的问题也不一样，因此形成了各各具特色的数据报表（除地市分公司向省公司汇报的之外）。公司又分设了许多部门，这些部门也都会需要数据，作为分析决策的依据。因此，了解各个部门的需求就成了业务报表系统的关键。

在调研的过程中，我选用了一种工具叫 PlayCASE，可以从网上免费下载，有很强的功能。下面就介绍一下，在需求分析阶段，我是如何使用这一工具的。

第一步，了解业务组织结构。公司内部的数据实际上是在部门之间流动的。业务部门需要知道在本地覆盖区内各基站的话务量、当天的话务量（即话务量的时空分布）。财务部门需要知道本月各类用户的话费收入、预交款收入、与其他电信运营商的网间结算等。计划部门需要各部门的分析数据。计费部门需要提供本月的账单统计数据、话单统计数据分布（比如分别按照基站分布、时段分布以及按用户类别分布）、预交款统计数据、当前的欠费总额分布、催缴情况等等。这些部门时常为了数据而产生了大量无谓的争议。在使用 Play CASE 工具时，先要将这些部门录入到 Play CASE 的“业务部门”中，构成了一个信息源的接收点（或发送点）；而 Play CASE 通过图示表示了这些部门的关系，并转换成了相应的软件结构。实际上，这是一种系统建模的方法，即把业务系统中的各个组织，转变为软件功能中的各个结构。这样，在需求分析阶段，明确哪些部门需要数据，从而保证了需求分析对整个公司的全面性，而不会忽略掉某一个部门，导致需求分析的不完整。

第二步，了解各个业务部门中的业务流程，使之通过 PlayCASE 转换成软件的运行过程，这是一种动态建模的方法。在上一步的基础上，追踪各个部门的行为，录入到 PlayCASE 中，并以形式化的语言描述各过程。对于复杂的过程，该工具不提供了进一步细化的方法，并且形成了业务流程图和业务状态图。根据这些流程图、状态图与实际业务部门的业务相结合比较，还是较为吻合的。在此步的实施过程中，运用了动态建模技术，使各部门业务流程的情况在软件的运行过程反映出来，从而保证了需求分析阶段中运行过程的描述能真实地反映实际情况，防止在后继的程序编写过程中，可能会经常发生的一类情况：程序员因为没有理解业务流程而出现“闭门造车”的现象，从软件的功能角度上保证了软件的正确性。

第三步，将业务数据转变为软件数据，这一步工作实际上就是收集各部门所需要的数据。分析各部门需要的数据都有哪些；以及数据是如何转换的，这可以归入“功能建模”的范畴。将这些相应数据录入到 PlayCASE 中，选定所属的部门。这时就自动地建立了 DFD 图（数据流程图），数据字典，省去了人工建立时的很大麻烦。

第四步，将业务上的数据关系转 为成软件中的数据关系，这里采用了面向对象的方法，把业务部门所需要的数据看作一个实体，部门间的数据关系就是实体之间的关系，比如：经营部门所需要的用户资料、用户话费，实际上就是用户这一实体与账单这一实体间的关系。Play CASE 提供了构件（不过我觉得是部件更为合适一些），来表示对应的数据，并提供了三种构件的表示关系即组装关系、分类关系与相连关系。这三类关系基本上反映出了现实世界中的业务数据之间的关系。例如现实世界中的用户资料与用户话费，在 Play CASE 中，可将用户构件与账单构件和相连关系表示，这种方法，实际上是借鉴了 OOA 面向对象的分析方法中的类、聚集、继承、封装等概念，能较好地反映出现实中的业务；同时，这一步的工作也为总体设计中数据库的概念模式设计奠定了很好的基础。

经历了上述四个步骤以后，利用 Play CASE 工具自动行成了软件需求规格说明书、初步的 DFD 图 and 业务流程图，为下一步的总体设计打好了基础。

使用 Play CASE 工具，使需求分析既能继承传统的结构化分析方法，又能吸收面向对象设计方法的优点。比如能把业务流程转变成为运行过程，业务组织转变成了软件的结构等都体现了这一点。而在运行过程中，对复杂过程的细分以及追踪则反映了传统方法中的自上到下分解的分析思想，这对于解决复杂系统的分析是很有帮助的。

通过使用，我觉得这个工具还是很不错的，因为它实际将以下四个方面的问题结合起来了：软件、业务、开发人员和用户。对于用户而言，PlayCASE 用图形化的方式显示出业务流程，使用户了解业务在软件中的运行过程，提供了将来验收软件时的依据，对于开发人员来说，使开发人员能更清楚地了解业务流程，不会再发生“因为不理解用户的需求而出现的闭门造车情况，从而导致开发出来的产品不符合用户需要”的现象。因此，PlayCASE 所自动提供的需求说明书能够很好地沟通用户与开发人员之间的理解，使他们都能对需求有共同的理解。

使用 Play CASE 工具后，使我们的需求分析取得了很好的效果，不但能自动地提供许多结果，如需求说明书等；还使需求的质量有限很大的提高受到领导的赞扬（领导不是学计算机的，但对公司的业务十分熟悉）；在后继的设计与维护工作中，我们感到工作似乎轻松了很多。

当然，该软件工具也有不足之处，一个突出问题是灵活性不够，一旦公司的部门或者组织机构发生变化时，整个设计都要重新来过。因此，在改进的过程中，我们在第一步过程预留了好多个虚拟的部门，以备将来进一步的扩充或者变动。

论软件需求分析方法和工具在 IC 行业内部的 CAD 应用

【摘要】

本文通过一个集成电路设计有关的软件项目，讨论了该项目的主要特点和本人所担任的工作，着重讨论了在项目需求分析过程中采用的具体方法和工具以及选用的理由。

由于项目的专业领域特殊性，分两类不同的需求讨论了需求分析中遇到的问题及解决方法，在这个过程中给出了对选用的具体工具和方法的效果的描述。接着本文讨论了对使用方法的改进的一些想法以及具体的实现过程。最后提出了我对需求分析的某些看法，强调了与客户沟通的重要性。

【正文】

近年，我一直从事某企业中有关 IT 项目的开发，有一个系统是用于计算机辅助电路设计的，包括了从上流设计到下流设计的所有流程，如用于可设计百万门数量级的逻辑门电路。有关方面把电路中路径的提取、过滤以及表示的某软件开发任务交给我公司，我有幸担任了该部分的需求分析以及设计。

我所设计部分为一单独可启动的软件，主要是解析文件中的连线路径，以列表视图和用直方图等把它们显示出来，还可以执行诸如查找与过滤等功能。

委托方对此提供了很初步的需求说明，把一些基本功能及性能要求描述了一下。我在需求分析时的工作主要有两点：第一，对该软件的界面等详细需求要自己重新进行分析提取。第二，对于已提供的功能要求需要深化和细化，以形成真正完整的需求分析文档。

在接到需求分析任务后，我分析了一下所要完成的工作。发现由于是专用领域的软件，对专业领域要求相当高，所以准备把此项目分成两部分：

(1) 界面所受专业领域影响几乎没有，但由于全部没有任何要求，反而会感到风险和改动可能是最大的。

(2) 功能方面由于委托方的许多功能都可以调用相应模块来得到，并且已有了相应的书面的简单需求，相应来说只是完成深化，对界面，我采用了部分 RUP 的思想迭代与渐进。而对功能需求采取了分层细化，每细化一层就要求委托方确认、修改和补充。

首先把风险较大的部分完成，这是现代软件开发的基本常识。我选择先进行界面的需求分析，第一步是根据功能描述抽取出逻辑模型，并使逻辑模型与界面元素及功能一一对应，大体上决定了界面应有的功能，然后根据该界面功能描述，确定具体的控件，这时，我参考了委托方已初步完成的主窗口的界面布局及控件的使用规律，然后根据需要完成的功能从 Qt（由于要支持 Windows 和 Unix 双平台，所以控件库采用 Qt）的类库中选择相应的控件，在提取和抽象逻辑模型时，我采用了 Rose2000 中的用例图，即以 USE-CASE 图来描述与外部的关系。之所以采用 Rose，我是基于以下的原因：第一，在已开发的部分中，委托方统一要求我们使用 Rose 进行类和顺序图等的设计和代码生成。第二，Rose 提供了标准的图来描述系统与外部的关系，在全球范围已是一种标准结构。第三，使用上的方便性。我用 Rose 的 USE-CASE 图，理清了我们的软件窗口与委托方主窗口以及外部角色（操作者）之间的相互关系。

在确定了界面元素后，考虑到文档的可理解性不是很强，我采用 Visio 2000 把界面的外观绘制出来，写上了基本的控件作用，随后送给委托方评审，幸运的是除了几个小功能的修改，委托方基本批准了我的方案。

下面的工作是为控件的行为及状态变化制定相应的状态迁移图，我选用的工具仍是 Rose，我用了状态图和时序图，把重要的控件状态变化及相应顺序进行了描述，随后的几天把相应的 DOC 文档建好写明，基本上界面设计就完成了。

下面的需求是针对功能需求的。虽然委托方技术部门有初步的需求文档，但由于领域的专门化不对，我不清楚其中复杂的路径提取关系及较深入的专业术语，一直有一种举步维艰的感觉。只能采用分层细化的原则，从最初的几条深入一层变成十几条，这样的话，不会一下子碰到太深的专业问题，可以循序渐进从委托方与文献的解答中不断学习，深化自己对专业领域的了解，这样在设计中自己始终是层层推进的。不至于一下碰到无法逾越的专业障碍。

在这一阶段的开发中，由于一直是与自己不熟悉的专业领域打交道，所以我觉得一些辅助设计工具似乎无法发挥应有的功能。在这期间，对我帮助最大的应是公司物 E-mail 系统，所有不清楚的问题提出，以及对问题的解答都通过它进行周转。换句话说，在需求分析阶段，它起到了一个与客户的交流沟通和客户需求的提取作用。所以，我认为在这一阶段，E-mail 系统是对我帮助最大的工具，其次是 Excel，我用它建立了问题跟踪图表，对每一个提出的问题，均需要记录上去，把问题结果（可分为已清楚、仍不太清楚、不清楚、尚未回答）均记录下来，根据这些表，我可以很好地了解自己作中的核心问题，并有了解决它的方向，提高了工作效率。

每进行一层的细化，我都把结果交付委托方审核，由他们进行提出何时能终止细化，大约在八层细化后，对方认为已达到效果，确认可以结束，至此，分析工作全部完成，项目的需求分析基本成功了。

在这次需求分析中，我认为取得成功的原因主要是方法和工具选择得正确。在界面设计中采用了流行的辅助工具，对需求及逻辑模型的建立提供很大的帮助，可以更方便帮助自己理清思路。选用了迭代法，把一些错误的影响在功能分析和界面分析的不断迭代过程中加以改正，在后期，以功能需求为主时，我主要依赖的是沟通工具和表格工具，这也说明辅助工具不是万能的，需求分析的关键之关键，应是客户的交流与沟通。

通过这次案例，我认为在软件的需求分析工作中，方法的重要性应远超过工具的使用，应当首先确定分析中的风险，把风险分类，用不同的方法去解决各类风险，而工具的选择不仅是要看影响力和名气，而是要真正为我所用，应把握其精髓，即是此工具到底可以对开发有什么帮助，而不是仅限于如何使用，我认为在需求分析中工具的作用不外乎两个：一是实际系统与环境模型等的抽象工具，二是需求表达工具，第一类的代表是 Rose，第二类的代表是 Word, WPS, Visio 等，在这次项目中由于地理上的限制还用到了沟通工具，Web 浏览与 E-mail 服务系统。

论 Java 技术在因特网平台上的应用

【摘要】

根据某类企业的迫切需求，我所在的信息技术公司组织了一个企业资源计划（ERP）项目的开发，希望推进我国 ERP 应用的发展，也希望更深入有效地运用 Java 技术。该项目的内容涉及到某类行业的企业生产经营的全过程，其基本目标是为了提高企业的劳动生产率，增加企业的利润，优化配置企业的资源，使企业的整体运营水平能上一个台阶。这是一个基于 Java 技术的 Intranet 典型应用项目。

在该项目中，我承担项目负责人的重要职责，比如在项目的准备阶段，我曾组织了对项目组的成员进行该类企业业务流程方面的培训，在项目需求分析和设计阶段，我着重考虑了架构好系统的框架和原型，为项目组及其他分析员进行下一步的细化分析奠定了坚实的基础。同时我还组织好项目总体组，把握住各模块之间的接口分析，保持各个分析员之间实现密切的沟通。在系统的开发阶段，做好开发、测试方面的协调和同步工作，保证系统的可靠性，在系统的实施阶段能够顺利地推进项目，此项目开发后的应用已得到了用户们的一致好评。

【正文】

与国际上 ERP 项目的广泛应用相比，我国的 ERP 应用水平尚有相当大的差距。根据某类企业的实际迫切需求，我公司组织了对一类 ERP 产品的开发，我有幸参与了该项目的分析与设计，开发的成果是一个典型的 Java 技术应用于 Intranet 的实际项目。

在选择具体的技术方案时，我们曾经进行了认真的思考和研究。对于选择普遍采用的微软模式的平台方案，还是跨平台式的 Java 方案，我们曾举棋不定，这是因为微软的 VB + ASP 已成为大家在校长时间工作后认可而熟悉了的方案。而 Java 由于其环境要求高与执行效率低的老大难问题，成为我们担心害怕的重要因素。但是 Java 的跨平台特性越来越成为人们的关注点，尤其是许多大中型的企业，他们现有的网络系统都是基于多种平台的，对跨平台的要求和呼声极为强烈，而对软件公司来说，软件的跨平台特性有可能会节约开发成本，降低维护量，也能获得更多客户的认可。综合考虑了诸多市场行情与行业发展因素，最终决定一定要用 Java。所幸的是现在 Java 用于因特网的开发隔壁已经越来越便利了。

目前 Java 在因特网上的开发技术已呈百花齐放之势态，有最初的 Java Servlet，有与数据库联系在一起的 SQL-J，还有可与 ASP 和 PHP 相媲美的 JSP。尤其是 JSP 技术的迅速发展，使得 Java 的网络应用不再是少数人的专利，JSP 以其执行的高效性和使用的方便性，已成为近年来大家首选的因特网开发技术，JSP 是一种页面开发技术，它以 Java 为其服务器端语言，结合 Java Script 作为其客户端语言，能方便地实现页面的表示。

选择好了后端的 Java 和前端的 JSP，还有一项重要的任务，那就是前后的联接。由于 JSP 主要用于页面表现，需要表现的内容要封起来，这样，为了保证主要商务逻辑的安全性，我们采用了 Java Bean 作为桥梁，即客户端 JSP 通过其中 Java Bean 的使用，完成主要的商务逻辑功能。在后台，将 Bean 构造好，形成一个强大的 Bean 库，再由前台 JSP 进行使用。

在进行 Java Bean 的规划时，我们下决心作出很大的投入，因为这些不仅是我们当前项目中所需急用的，而且还应成为公司长期积累使用的一个强大的资源库，能实现一定程度的资源共享和软件复用，为其他项目开发打好基础。因此，此次规划的目标是形成公司 Java 技术的 Java Bean 的平台库。

我们根据 Java Bean 所体现的类的用途，将这些类分成几个层次。最底部的一层就是参数化类的构造，这一层的类所实现的主要功能包括通用访问机制，对数据库等其他层次访问接口和公共处理系统等。中间一层是实体类的构造，这些实体类包括与数据信息相关的结构及其处理方法，其中的重点是包含了一些重要的商务逻辑的处理。这一层类与系统各部分相关，并且其安全性要求很高，直接影响到系统主要功能的体现因为系统的主体是对一些逻辑进行处理，这就要求这层实体类的规划需要十分认真，做到细节准确。最上面的一层可以称为接口类，这一层类主要用于实现底层的类与前台之间的关系。也只有这层类才能由前台 JSP 进行 Java Bean 调用而加以使用，只有这层具有开放性，这一层类除了上述的接口功能外，还应当有一项重要的实用内容，即包括用于实现前台 JSP 的页面自动构造程序。

这里所说的页面自动构造程序可以认为是本系统的一个重要特点，目的是为了用户可以方便地自定义界面，而不需要由程序员修改程序，这样能够极大地满足了用户的要求。页面自动构成程序的主要内容包括对界面元素的定制与修改、位置的修改、动作的触发、行为的控制以及报表设计和计算汇总等功能。页面自动构成程序的设计主要采用上述的接口类 JSP 相结合的方式，用类实现元素的定制、控制及关联，并将重要信息加以保存，以利于用户的多次反复修改。该自动构造程序提供了强大功能，已成为我们的一个独立产品。能应用于各个项目的界面制作，实现了我们原先制定的共享资源的目标。

前台 JSP 的应用中，做到了尽可能最简化的程度，这样可以提高系统的安全性。当然在我们的系统中，还存在一些客户端控制比较复杂的情况，为保护这段比较复杂的控制脚本，我们采取了用 Servlet 的方法，保护这段脚本，从而保证了一定程度的安全性。

在系统的登录过程中，我们采取了相当严格的登录键检查操作，和用户没有供应商提供的相应的键，就无法通过验证而进入系统。对于试用版的用户则提供了一种有效期限约束。这些加密或安全措施，通过在 Java Bean 中封装了严格而有强大功能的加密算法，在客户端申请验证后才能准予通过。

在使用这套技术方案的过程中，我们曾经遇到过许多的困难。比如：前面曾提到过要求 JSP 中代码能够尽量简化，以提高安全性。由 JSP 中仍有一些容易让人可能猜测到处理方法的语句及处理的过程，为进一步提高安全性，我们通过查阅大量的网上资源，才形成了一套较好的措施，比如制作 JSP 的标记库，将有可能被猜测的处理进一步加以规划，对应地生成一套行之有效的实用标记库，这样就又增加了一道很有效的防护墙，大幅度地提高了安全保密性，并且使页面结构的分离达到了一定的水准。又如：在对数据的处理上，刚开始时也总是遇到系统运行会变得越来越慢的情况，最后追查其原因，发现原来是数据的连接过多，我们及时地采用了数据连接池等技术解决了此类问题。

该系统采用 Java 平台，提供了深入地使用 JavaBean 和 JSP 的方案，其效果是相当显著的，在用户真实使用环境中受到了一致好评，运行也较为稳定。由于采用了统一而方便的页面自动构造程序，用户的界面非常友善，并且可以按用户需求进行定制，满足了用户的适应性需求。而在我们公司的内部，也开始建立了一套基于此平台的资源库，成为公司的今后开发使用的宝贵财富。

必须指出的是，在此系统中，还存在着很多的不足，比如实体类的组装程度尚不尽如人意，根据多种商务逻辑的一些共同点，可以进一步加以抽象封装，使这部分内容能满足多种系统对类似逻辑的处理过程。我将会在今后的工作中进一步加强各方面的分析能力，带领团队不断地超越现在的层次与水准，加强我们的队伍建设，希望有更多优秀的软件产品上写着 Made In China。

Java 技术在因特网平台上的应用

【正文】

数据通讯是不前十分活跃与热门的计算机与信息技术的应用领域。某大型通信公司开发了其业务的主要支撑平台，在这里，我们简称之为“通信信息服务平台”，用于在全国与全球一工展数据业务的需要。该平台是一个典型的 Java 技术应用于 Internet 的项目。

作为信息技术公司中的一名技术骨干，我有幸参加了该系统的分析与设计工作，承担了相当多的 Java 应用开发任务。此系统中的软件部分大多由 Java 来实现，在全 系统中我们是这样来用 Java 构架系统的：

- (1) 本系统可分为 4 层，分别是 Browser、表示层、中间件层和数据层。
- (2) 表示层用 Java 中的 JavaScript 来实现页面输出。
- (3) 中间件层用 Java 来实现 CORBA，即实现 Component（构件）主要实现业务逻辑析封装与复用。
- (4) 数据层主要是数据库和存储过程的实现。

我们在应用 Java 技术时，所采用的技术和策略可大致上归纳为以下 5 个方面：

(1) 使 Java Script 尽量简单，因为 Java Script 在我们系统中是放在服务器端执行的，该语言是通过一个解释器解释执行的，相对速度很慢，我们要用了两台 HP 前置机来运行 JavaScript，但是其运行速度还是不理想，所以我们在设计中把 Java Script 仅用来显示从中间件层所得到的数据，生成动态页面。在最初的设计中表示层（Java Script）曾承担了一些业务逻辑处理操作，导致效率不理想，因此，我们不得不尽量地减少 Java Script 的程序量。

(2) 用 Java 实现 CORBA 时，应尽量考虑共享和复用，在本系统中，最初的设计是让 Java 在实现 Component 时，只是执行一些数据库表的操作，导致表示层的负载较大。后来，我们重新设计时，总结归纳了所有的 Use Case，找出了其中可供共享和复用的接口，把相同的业务逻辑操作封装到一个接口中去。因为 Java 的执效率比 Java Script 要高，因此提高了系统效率。

(3) 在别的项目中，我们曾大量地使用过 Java 中的 JSP 技术和 Servlet 技术，一般人可能不能区分这两种 Java 技术的区别。为了得到系统的一些执行速率的数据，我们采用了一个著名的压力测试软件——Load Runner 来测试这两种技术的差别。测试表明：用 JSP 和 Servler 完成同样的一个操作，并且保证是在相同的测试环境中（相同服务器、压力测试工作站与数据库环境），得到测试数据却有着很大差别，JSP 完成一个操作的平均执行时间大致会是 Servlet 程序的两倍。在一个企业级应用项目中，这可能是一个很关键的瓶颈。因此，我们得出的结论是：在可能的条件下，尽量地多使用 Servlet。当然，与 Servlet 相比，JSP 编程快速，修改方便，在访问量不是很大的应用场合下也是可以接受的。

(4) 使用 Java 作为整体解决方案时，应尽量使用相同版本的 JDK。在用 Java 作为编程语言的项目中，几乎大多要遇到“汉字”问题，即 Java 在没有经过转换的情况下，在输出汉字时，很可能会出现乱码。采用不同版本的 JDK，解决的方案是不一样的，比如 V1.2.2 版本的 JDK 和 V1.3 版本的 JDK 解决方法就会有一些不一样，把 V1.2.2 的 Java 程序放在 V1.3 的 JDK 中，就不能顺利输出汉字了。其根本原因在于 Java 使用了 Unicode 编码，和我们中国的国标编码不一样。所以在这个意义上一些人竭力鼓吹的“一次编写，到处运行”似乎不一定能在所有的场合都行得通。

(5) 使用 Java 时，应尽量遵从软件规范。在 Java 中有一个 JVM 的概念，即在 Java 虚拟机中使用了一个垃圾收集器，专门用来回收内存。但是该垃圾收集器在给编程人员带来方便的同时，也隐埋下了隐患。在程序设计中，并不能强制执行垃圾收集器，所以，开发人员不能确定某对象是否已释放，常常让编程人员养成依赖自动收集的坏习惯，因此我们要求：在 Try,Catch 之后必须明确要求回收内存（当然，也只能是通知垃圾收集器来回收垃圾），这样可以有效地提高系统稳定性。

以上这些实用性的技术与策略，是我们在实践中的一些实际体会，仅供各位开发人员根据实际情况参考。

当然，在使用 Java 作为解决方案时，也会遇到很多让我们头疼的问题，这些问题导致同时执行的并发性比较差，系统速度慢等等。归纳起来看，我们曾遇到过的主要具体的问题有：

- (1) 用 Java 来实现 CORBA 中的 Component，有时效率会比较低。
- (2) 用 Java 来建立数据库连接往往会比较慢。

(3) 用 JSP 编程时容易导致系统信息的扩散。比如：如果有黑客攻击一台运行 JSP 程序的服务器，他可以故意地输入一些非法字符或异常信息给 JSP 程序，于是程序执行将出现异常。这时，就会在页面上打印出相应的错误信息。很不幸的是，这些信息极有可能暴露出这台服务器的 JDK 的版本号与路径信息等内容。这往往容易让黑客们有机可乘，有可能去抓住系统的漏洞。

在发现了这些问题后，我们经过仔细研究，找出了一些解决办法。比如：

(1) 既然用 Java 实现 Component 比较慢，我们就尽量减少 Component 所执行的业务逻辑量。争取把能够放在存储过程中实现操作，尽可能在存储过程中加以实现。众所周知，数据库的存储过程操作，比起在 Java 程序中执行数据库操作要快得多。

(2) 既然用 Java 建立数据库连接比较慢，我们就可以把数据库连接封装成连接池（Connect Pool），从而能非常有效地提高系统效率。我们也曾经用“Load Runner”作过压力测试，使用连接池比不使用连接池的速度要快上 3~5 倍。

(3) 为了对付 JSP 程序与 Servlet 程序会打印出异常信息的问题。我们曾查阅了很多 JSP 或 Servlet 的资料，最终是毫无头绪。但是我们可以换另一种思路，即是不从程序下手，而从 Web Server 着手，我们可以把 Apache 配置成为使这类异常信息不再打印出来，而是使之仅出现一个通用的异常说明的页面，这样，就能十分有效地解决这个问题。

在我们使用 Java 作为编程语言的这么多项目中，绝大多数是比较成功的。Java 语言作为一种快捷、稳定的计算机语言，开发基于因特网应用的项目大多是相当稳定和比较适用的。

在我个人看来，Java 的应用前景十分光明，大体上可以着眼于以下方面：

- (1) 在因特网上将会有更加广泛的应用。
- (2) 在嵌入式设备中，Java 也大有用武之地。比如：在最新推出的 Java 技术中，Java 已经进入了手机领域。
- (3) Java 程序大多以线程运行，占用资源少，会逐步代替 ASP 与 CGI 程序。根据第三方测试表明：JSP 程序比 ASP 程序要快 2 倍以上。用 JSP 代替 ASP 应是大势所趋。
- (4) Java 在无线互联网中的应用将会更加广泛。Java 支持 WAP，可以方便地用 Java 开发 WAP 程序，实现 WAP 应用。
- (5) Java 与 XML 的无缝连接使 Java 在数据传输和异构网络通信方面有着很大的优势。

就我个人而言，我将会在相当长一段时期内致力于 Java 在无线互联中的应用，为我国的移动通信事业开发出更多的优秀实用的项目。

论 Java 技术在因特网平台上的应用

【摘要】

因特网上应用的日益普及与深化，为 Java 技术的运用提供了广阔的活动舞台，也大大推进了 Browser/Server 模式的企业内联网应用与网络计算。

作为某信息公司中的技术骨干，我有幸承担了某银行信贷管理与查询系统等的开发任务，独立地完成了其中的系统设计、类设计、部分开发及测试工作。

整个系统完全按照 J2EE 的标准来设计。前台界面应用了 JSP 技术，控制部分采用了 Servlet 来开发，业务逻辑应用了 EJB 技术来封装，应用服务器用了支持 J2EE 标准的 BEA 公司的 Weblogic，后台的数据库选用的是 Informix7.3，目的是为了与银行中其他业务系统数据库保持一致。在硬件平台上，我们选用的是 HP 公司的某台中型服务器机器，操作系统是 HP-UX。

该系统界面运用的是 IE，它不仅兼容性较好，而且已为广大用户所熟悉。系统运行后，各个支行都普遍反映界面友善，功能强大，开发的效果令人满意。

【正文】

在银行应用中私人的储蓄、企业的会计、国际业务、信贷、财务管理都十分重要的，它们构成银行的基础业务系统。我从事开发的信贷业务更是银行利润来源的重要部分，与储蓄，对公等以交易事务为主的业务模式所不同的是，尽管信贷也是交易，但需要更多其他辅助信息的支持。如客户基本资料，在本行内业务发生状况、信用等级、是否有逾期贷款未能归还等。单个支行的有关业务人员及分行管理人员都希望能方便及时地了解这些信息，传统的基于终端的用户界面难以传递这么多信息给用户，所以我们决定采用基于浏览器 IE 的用户界面，一方面 IE 使用方便，不需要专门培训，另外它是与 Windows 操作系统捆绑在一起的，也可节省前台费用，在开发技术上有 ASP、JSP 可供选择。

由于考虑到 Java 技术在 Internet 上的迅速发展，J2EE 更是提出了全新的用语言来统一平台的思路，于是我们决定采纳 J2EE 标准，并选用了 JSP。在设计上，基本上是采用了一个交易画面对应于一个 JSP 程序，充分发挥 JSP 动态处理页面的长处。

为了使设计有更好地可扩展性、灵活性与逻辑性，能为以后扩展奠定坚实的基础，我采用了 (Model, View, Controller) 的 MVC 设计模式，View 全部由 JSP 实现，而 Controller 则是设计了一个 Servlet 程序，它负责处理前台浏览器传送来的所有请求，并按事先定义好的路径/程序关系，分发给相应的 JSP 程序去处理，由于 Servlet 本来就是为 Java 服务器端编程来设计的，因此由它来负责服务器端的处理是相当合适的。

在开始设计时，我运用了构件技术，由 EJB 承担起设计模式的 Model 角色，具体的贷款开户，放款，结算逾期贷款，归还贷款等交易都对应一个具体的 EJB，为了将这些处理逻辑与相应的数据操作分离开，能更加便于维护，我将处理业务的 EJB 设计成 Session Bean，而为每个 Session Bean 再配备一个相对应的 Entity Bean，用于访问后台的数据库，贷款管理中很重要的一点是进行查询，我按照需求分析的结果，为每类查询都设计了相对应的 Bean，其目标是尽可能地提高查询的速度。

在对数据库的存取中，我本来的设计应用 Informix JDBC 所带的 Driver Manager，这样，在存取数据库中的 Bean 中就要把 Driver 及 Server 写入，后来考虑到应尽量提高应用的平台独立性，在参阅了 J2EE 中 JDBC 部分的说明后，改用了 Data Resource 的处理方法，这样，即使以后数据库换成 Oracle 或其他产品，程序也不用修改，只需要在配置时进行变动即可。

在这次信贷管理系统的开发过程中，Java 的平台无关性优势，在开发人员从事开发的活动中体现得淋漓尽致。由于经费相对紧缺，我们的开发环境是各个项目组共用一台 HP 机器，虽然每个开始小组都搭建了自己的环境，但项目一多，特别是遇上结息与批量测试等场合，机器就显得不堪重负，使开发与测试工作的效率大为下降。我们小组由于采用的是 Java 技术，大家可以在自己的 NT 机器上搭建相同的环境。这样一来，大家平时的开发工作，包括 JSP，Servlet, EJB 的程序，都可以在本地完成，只是到测试或展现阶段才需放到 HP 开发机器上进行。

以前我们开发的 Web 应用，往往只是应用了部分的 Web 技术，如采用 Apache WebServer、ASP 开发语言等。整个体系的集成与组合往往不够理想，这次由于我们采用的一整套符合 J2EE 标准的组件，整个系统的协同性与一致性非常之好。再加上有一个支持 J2EE 的应用服务器——BEA Weblogic，以往我们做得不理想的复杂配置，模块间的连结，如今都用不到再操心了，只需在图形化的配置工具中，输入系统所需要配置，如路径与实际应用程序的关系，组件中的 EJB 引用，Data Resource 的属性等；全部配置完成后，Weblogic 会替我们完成项目的部署，并将这一切有关的程序都封装起来。

原来，我们开发小组的文档编制任务显得非常之繁重，因为整个系统既有交易部分，又有管理查询部分，交易、数据与源程序都很多，为了解决这个问题，我们直接应用了 Java 源程序中的 Javadoc 导出文档，这样不仅文档美观，而且能够保持与源程序的一致性，实乃一石二鸟之举。

整个项目完成后，用户使用下来都觉得界面友好，操作简便。但是我心里知道，这个系统还有很多可以加以改进的地方。

首先，基于 Java 系统的开发需要资金较多的投入，由于该系统受到经费的限制，只申请到一台生产用机，这样，Web Server、Application Server、DB Server 只能被挤放在一起。虽然 Weblogic 能实现部分负载平衡，但在将来的业务发展时，这样的分布肯定不是最理想的。好在我们在设计时已经考虑过尽量有良好的扩展性，在以后条件许可时，只需进行在不同机器之间的进一步部署即可，应用程序大体上无需改动。

其次，在设计上，可以采用 UML 的产品，如 Rational Rose，另一方面，Rational Rose 具有自动代码生成功能，也可以大大节省开发的成本。

最后，目前的信贷管理系统相对用户数目量不多，当推广类似系统需要拥有大批用户时，基于 Java 的系统的响应时间与系统分布都会有效为突出的矛盾出现。

以上这些，都是我在今后的系统设计与开发中需要加以注意的地方，也是运用 Java 技术应当努力的方向。

论改进 Web 服务器性能的有关技术

【摘要】

基于 Web 技术的数据库应用是当前应用的一个热点，在用户数目与通信负荷很大的场合，提高 Web 服务器性能是一个迫切的课题。本文从笔者参与某个银行系统项目开发的经历出发，阐述了提高 Web 服务器的性能应渗入到项目论证、选型、开发、运行和管理的各个环节，只有各个环节都能充分考虑到性能与质量的需要，系统的性能才是真正可保证的和可扩充的。

文章从系统的实际运行与相应的经验出发，阐述了性能改进方面的一些具体措施。

比如：在本文中讨论了 Web 服务器平台的选型考虑；Web 服务器的配置管理；应用系统本身的优化与预先设计系统时可扩性的性能保障等具体内容。

通过技术上的分析与改进，综合性地运用多类措施与手段，在实际系统中，Web 服务器运行的性能得到了一定程序的保证。

【正文】

我所在的单位是把目标定位于金融领域开发 IT 应用的一家信息技术公司。随着金融电子化建设和商业银行之间市场竞争的加剧，各主要商业银行不断通过信息技术提供新的金融产品，并且希望整合市场渠道。比如主要的商业银行不断推出形形色色的网上银行服务。在这种背景下，本人参与了开发新一代网上银行产品，涉及提供网上个人理财服务、网上外汇买卖服务、网上企业服务等具有市场竞争力的产品。作为项目开发的组织者之一和主要的技术骨干，在整个项目开发过程中始终要处于第一线，从而在改进 Web 服务器性能、提高整个网上银行系统性能方面收获良多，在本文中简要讨论如下，希望与读者们共享经验。在 Web 服务器配置与优化方面，我有如下几方面主要的体会：

第一方面是 Web 服务器选型考虑。

在 Web 服务器选型及网上银行搭建这初，我们就已充分考虑整个网上银行的性能及可扩展性问题，这一考虑为该系统的稳定性及扩展能力方面打下了坚实的基础。

某银行原有的一些网上产品由于开发较早，故而采用的是老式的 HTTP Server + CGI 程序调用的方式。这时，每一客户请求需要对应于后端系统的系统进程来运行 CGI 程序来处理，系统的开销相当大，系统的扩展能力也很差，性能已不能满足业务处理的需要，故而在为此银行系统具体选型的时候，我们一开始就否决了这种方案。

通过市场上同类产品的比较选择，我们选择了国际商业机器有限公司 IBM 的 Web Sphere 产品系列作为该行网上银行系统的建立平台。作出这样选择是因为 Web Sphere 基于使 HTTP Server 和应用服务器相分离的整体架构，同时支持 JSP、Servlet 和企业级 Java Bean 等轻量级线程规范，所有的请求对应于应用服务器上的处理线程，系统的开销低，效率非常高，同时 Web Sphere 整个体系结构相当的灵活，为适应扩展需要可以作不同的横向和纵向扩展，从而可以满足各银行未来的扩展需要。

正是因为在一开始选型的时候我们就已考虑到未来的扩展需要，整个系统在接下来的几次性能改进方面，我们大体上都能相对顺利地达到了预期目标。

第二方面是 Web 服务器的性能配置。

在一开始系统上线的时候，由于系统的负荷不是很大，为了节省系统总拥有成本 TCO 投资，我们在一台较低配置的 IBM RS6000 上投产了该系统。整个系统的 HTTP 服务器、应用服务器、通信服务器等均位于该台机器上，由于初始投产时用户不多，所以系统的性能基本上能令人接受。

但随着业务的发展和用户访问量的增大，我们发现该服务器的响应变慢，系统的 CPU 利用率和内外存交换显著增大。经过跟踪，我们发现关键原因之一是系统的内存不足的缘故。由于网上银行服务器把大量用户的会话信息保存在内存中供给应用系统使用，当内存不足时，大量 Session 信息被迫交换至硬盘，大量 CPU 时间消耗在等候内外存的交换上，系统效率迅速下降。

鉴于这种情况，我们把该服务器的内存由 2GB 扩充为 4GB，同时相应调整用户会话信息的保存时间，这样整个系统的效率又回到较为理想的状况。

由于新应用的不断投产及数据库操作的日益增加，我们后来逐渐监控到系统的数据库处于繁忙状态，系统的错误日志也记录下了供应用服务器使用的数据库连接处出现资源不足的情况。在这种背景下，我们认为整个系统由于硬件配置所限，应该进行横向扩展，因此我们把数据库服务器分离出来，配置到另一较高性能的服务器上，相应定义的数据库资源也大幅增加，这样整个系统的性能又处于较为理想的状况。

第三方面是对应用系统进行相应的优化以提高性能。

Web 服务器配置及相应的硬件扩展不失为解决系统性能问题的一条捷径，但应用系统的优化也是应该重点加以考虑的，毕竟它能够在投入较少的情况下提高系统的运用效率。

在开发的初期，我们就已经十分注意系统的利用效率，比如提醒程序员尽量不要利用用户会话信息（Session）来传递大的对象，对于内存要注意回收等。同时，通过内部的交流会推广与介绍一些小的，有用的编程技巧来提高开发人员的水平，通过代码的抽查，希望能在早期就发现问题等。

在系统运行期间，我们通过监控发现，应用服务器所基于的 Java 虚拟机，其内存堆的空闲空间有不断下降的趋势，每隔若干天导致空间消耗殆尽，无法分配新对象空间，从而导致系统重启。在排除了系统本身问题的原因外，我们确定为应用系统的开发有问题。通过从网上下载 IBM 公司检测 Java 虚拟机的相关工具对 JVM 进行监控后终于发现系统内部存在着不能回收内存的对象，再通过查找相应的程序发现在该程序中有“环状”的对象引用，从而导致对象使用后不能被垃圾收集器所回收。这个问题的解决过程虽然十分艰苦，但由于该问题不能通过升级硬件或增加资源配置而得到根本解决，会给系统带来很大的隐患。所以，整个过程的分析与解决是完全值得的，更何况通过查找故障原因的过程，给整个项目组上了生动的一堂软件质量保证课，对项目组的质量意识起了很大的促进作用。

所以说改进 Web 服务器的性能并不单纯是系统管理方面的工作，它渗透到开发以及系统运行第一系列环节中。

第四方面预先考虑未来的扩展与性能需要。

随着系统的发展及成熟，考虑到用户访问量的不断上升，为了预留系统的发展空间，我们最近又对整个系统作了一个系统性的升级。通过引入多台 HTTP 服务器及应用服务器并行工作提高整个系统吞吐量及单点故障克服能力。由于在一开始选型的时候就已经充分考虑到动态负载均衡及横向扩展方面的需要，这一项的升级无需对整个系统的体系结构作根本的变革，对应用程序来说，更是没有造成任何影响。

整个项目历时近两年，从这两年的系统情况来看，整个系统是成功的。根据我亲身的经历，系统性能并不单纯是系统运行与管理阶段的问题，而是渗透在项目论证、开发以及运行的各个阶段。只有各个阶段都能充分考虑性能方面的需要，在实际运行时，整个系统的性能才可能真正有保障。在技术方面来看，可以综合利用选型评估、硬件扩展、应用优化和系统配置优化等一系列的手段；比如在硬件扩展方面，又可以分为主要部件扩容，纵向升级、横向升级等方面。在我们的项目实践中，曾综合地利用了上述的各种手段。比如某银行的整个系统日访问量不足 1 万至现在的每日超过 10 万次以上的点击的发展情况来看，整个系统的性能保障及提高方案是比较成功的。

改进 Web 服务器性能的有关技术

【摘要】

一个大中型的图书馆信息系统涉及到许多方面的技术与方案，本文着重讨论与 Web 服务器性能有关的一些内容。

本人有幸作为项目负责人之一参与了某大型图书馆数字化信息系统的设计和基于 Web 应用软件的开发工作。由于在数字化图书馆信息系统中流通着的大多是数字化的索引、文摘、全文、图像或音频视频等多媒体信息，对 Web 服务器性能有着较高的要求。

结合实际工程经验，本文将从硬件实现手段（缓存服务器、均衡负载设备、Web 双机镜像、CPU 和网卡的提升、网络带宽扩充）和软件实现手段（三层 C/S 软件结构设计、应用程序部署）等两个大方面论述如何提高 Web 服务大路的性能，以便使用户能够更快捷、高效、安全地使用应用系统。

【正文】

随着 Intranet 信息技术的发展，图书馆为了更好地发挥其图书流通、资料检索和学术交流的职能，图书馆的数字信息化工程也势在必行。某图书馆为了尽快地步入世界先进图书馆的行列，已经启动了一部分的数字图书馆工程。

该数字图书馆工程主要包括对外信息 Web 发布系统，交互式检索网，后台馆藏信息管理系统、多媒体资料采集制作以及外 VOD 点播系统等。本人有幸作为项目负责人之一，参与了整个数字化信息系统的总体设计，并参与了基于 Web 的一些应用（如对外信息发布系统、图像/全文混合系统、VOD 点播系统的）开发。

某图书馆数字化信息系统从网络环境上讲，主要划分为多个网段：（一）Intranet 接入部分，采用 2M 的 DDN 专线；（二）公共网段（非军事区），主要包括前台发布数据库服务器、Web 服务器、E-mail/FTP/DNS 服务器、检索服务器及 SAN 网络区域存储设备；（三）是内部局域网，包括内网 Web 服务器、后台馆藏数据库服务器、OA 服务器等。（四）是 VOD 点播专用网，包括音频视频点播服务器等。由于制定了严格的网络级和应用级访问权限，通过具有三层交换能力的高性能交换机和安全授权认证系统等，有效地控制了访问权限，确保了数据的安全性和完整性。考虑到经费和人员素质及今后的维护管理运营等方面，操作系统采用 WindowsNT 平台，服务器选用 DELL 高端的系列，数据库采用 IBM 的 DB2。主干网为千兆快速交换式以太网，局域网百兆到桌面，VOD 点播网十兆到桌面。

在该网络环境下应用主要分为三大部分：（一）对外 Web 发布系统、对外图书辅助检索系统；（二）后台馆藏信息管理系统和图像/全文混合检索系统；（三）VOD 点播系统。由于绝大部分应用采用 Browser/Server 方式结构，最终用户在本机只需安装 IE 或者 Netscape Web 浏览器，在后台数据库服务器的支持下通过网页方式请求和访问各类应用服务。另外，由于在图书馆信息系统中流通的多为索引、摘要、全文或音频视频等多媒体信息，对 Web 服务器性能与网络带宽等有更高的要求。

通过不断地试验和实践，我们发现从以下几个方面可以相对有效地提升 Web 服务器性能：

（1）缓存服务器和均衡负载设备使用可以缓解访问瓶颈，提高网络带宽、实现均衡负载。

缓存服务器也称为 cache 服务器，可以存储 cache 静态的内容如网页、多媒体点播资源和会议实况（已压缩的、有一定格式要求的）等。此外，目前美国 cashflow 缓存服务器，已经可以存储 cache 数据库、ASP 等动态内容。cache 服务器通常放到防火墙之外，外网 Web 服务器之前，因此 Internet 用户点击网页不再直接访问网站 Web 服务器，而是访问 cache 服务器。

由于 cache 服务器具有多个 CPU 和高速大容量 I/O 通道，独立的 OS，因此能大大缓解 Internet 访问瓶颈，而且也具有一定的抗黑客攻击的能力。

目前某图书馆采用这种方式，把大数据量的静态图片、点播资源、虚拟三维应用等都事先置放在 cache 服务器中，即使现今只有 2M internet 的接入带宽，以上应用的播放速度和效果仍能让用户满意。

另外一种方式采用均衡负载设备或 Web 双机镜像。这种方式通过负载均衡的方法达到 Web 访问性能最优。Web 双机镜像是较早以前流行的方式，虽能使系统可靠性提升，但由于双机总是在互相询问对方状态，将会影响一定的访问性能。均衡负载设备是独立于 Web 服务器的硬件，它和 Web 服务器及网站中其他服务器接在同一交换机上，通过负载调度程序为各个服务器分配工作量，从而，能达到充分利用资源，提高访问性能的目的。只是由于某图书馆目前对外发布资源相对仍较少，只有用了三台 Web 服务器，因此目前的均衡负载设备作用还不显著。

（2）从 Web 服务器的配置来看，Web 服务器自身 CPU 个数及速度、网卡数量、Web 服务器与防火墙的位置关系等，都会影响到 Web 服务器的性能。

从 Web 服务器硬件本身来讲，CPU 个数的增加、网卡个数的增加、I/O 信道的扩展无疑可以直接地提高 Web 服务器性能。此外，由于千兆口的防火墙目前较少且费用较高，如果把 Web 服务器放置防火墙之后，一定会大大影响 Internet 访问性能。某图书馆采用 IDS（入侵侦测）+Web 服务器（服务器防火墙，较低端，不会影响流量）+应用服务器+数据库服务器（防火墙，高端），分层次的安全模式，既保证了系统的安全模式，既保证了系统的安全性，又提升了网络访问性能。

另外，某图书馆还采用了 SAN 网络区域存储来提高服务器访问速度。

（3）三层 C/S 软件结构设计和应用程序的适当部署也会提高 Web 服务器的性能。

将业务逻辑、通用访问接口与数据等相互分离、分别置放于 Web 服务器、应用服务、数据库服务器上，通过过程功能逻辑的合理部署，也能大大改进 Web 服务器性能。

一般的原则是，Web 服务器只需接受 Internet http 访问请求，使 Web 只有最少的任务，把实际处理交给各个应用服务器处理，然后返回结果给 Browser。某图书馆采用这种方式专门开发了搜索引擎应用服务器和混合检索应用服务器等，达到了良好的应用效果。

事实上，Web 服务器的性能提升还存在很多手段和方法，比如 CPU 与存储之间关系，Web 交换机等等，有待于我们进一步的实践、分析和讨论。

论实时控制系统与企业信息系统的集成

【摘要】

近年来，在应用需求的强大驱动下，我国通信业有了长足的进步。现有通信行业中的许多企业单位，如电信公司或移动集团，其信息系统的主要特征之一是对线路的实时监控要求很高，数据量庞大，如何将实时控制与信息系统集成在一起便成为系统实施的一个关键部分。

在参与了某个通信公司的一套网管系统以及决策支持系统的设计后，我们分析了两者的集成与应用工作，深切地感受到有一个良好的设计策略以及重视所选用的工具是一个关键。这个项目主要是对下属各分站的子网以及有关链路的连通情况进行实时监控、实现报警、路由控制和授权等功能，其关键在于提供一个实时显示情况地地图界面，并将数据汇总和组织，建立起数据仓库以及进一步实施数据挖掘分析，从而能支持企业的决策分析。我作为设计人员之一，着重在本文中讨论控制系统与信息系统集成时的策略。

【正文】

众所周知，通信行业需要有一整套监控通信网络手段，其工作特点是涉及到的各分站与基站的在地理位置上的分布性，更加需要有在更高一级提供检测不同分站链接情况的手段。一般来讲，由于数据都是海量的，所以如何将整个网络系统所得的数据及时处理，以便和决策部门的分析相结合，也成为迫切需要解决的重要课题。简言之，分布性、实时性以及数据海量性是解决整个系统设计和集成的核心问题。

首先，让我们来讨论一下“网管监控系统”。由于我参与设计与开发的这个系统并不是位于基层的分站，其定位在将不属各分站的主机通信数据（包括数据流量、链路负荷、通往其他结点即主机的连通情况等）加以收集，所以对于具体通信事务的底层操作要求并不很高。

考虑到上述原因，我们采用了一个理理信息系统开发平台 Mapinfo 并采用 Delphi 编程，后台用 SQL Server 数据库（这里由于考虑到决策所需要用到的 Microsoft 公司的 OLAP Service）。在分析和计划之前，我们先对 ITU801 标准做了详细的探讨，这只是一个有关子网和链路定义以及分层等描述的标准，在听取了许多分站人员的建议后，将 MAPINFO 公司提供的—个相关的 MAP X 的 Active X 控件嵌入到 Delphi 程序中，利用 MAPX 中提供的丰富的类以及操作，比如 Object、Layer 等实现网管界面，并且加入了子网和链路的概念，对属下的分站可以随意地组合成为不同子网，而且实现了放大与缩小的功能，大致可以将整个地区的分站集中在一张地图中，能显示在屏幕上，这时，只是显示出各个分站在概要，小到可以显示出某台主机的机柜、机框直到插件板（因为这些都要实时监控）。我们采用了分层的方法来实现以上缩放。对于一些静态的数据，如分站，主机的位置等则先用 Mapinfo 公司提供的一套编制地理信息的工具（MAP X 是其提供给编程工具的一个 Active X 控件）做成静态的层次图放置于数据库中。

我们新做成的这套系统通过与各分站的专用线路加以连接，能实时地得到数据，显示于地图上，反映出各站、各子网、各链路的实时状态，并能将控制命令传回分站（如强制链路中断、路由转换等）。

现在，让我们来讨论其中最为关键的问题，即是要将实时控制系统与企业信息系统加以集成，我们的设想和体系结构大体上可以用一条简图表示（此处暂略）。

在这个体系结构中，由各分站保留着详细的数据，网管系统则在一定时间间隔内将汇总到的数据作少量统计，抽取其中需要保存的内容放入数据库，如每每分钟流量，某分站与其他分站每分钟通信流量，在该分站中某个链路的负荷（这些链路有可能是动态分配的，也可能是固定分站之间的通信链路），尽管如此，数据仍然是海量的，因此，如果要把这些数据都直接送到各个决策部门，比如送给市场部门是不现实的。所以，我们在数据库的基础上建立了数据仓库，确定了客户、时间、通信量、计费 and 故障等几个数据仓库的主题，每隔一定时间对数据库中的原始数据进行清理与抽取等预处理工作，建立好数据仓库。这里的预处理包括了许多方面的内容，比如有建立计算时间，但是无计费的（计费值为零）的数据，应视为建立失败的无数数据，需要予以剔除，某些企业租用的是专用线路按月计费，中间的通信因此无计费的一些有关记录也应剔除等。

在预处理之后，再利用 OLAP SAervice 的分析将数据融合与汇总，按照决策部门的需要提供相应数据（比如：市场部门需要每一分站的收益，客户分布情况以及客哀悼费用等）。这些都可以由 OLAP Service 对数据作预先处理，此时处理完的数据在逻辑上是以立方体（CUBE）形式存在的，其占用的存储空间便能显著地降低，如 1999 年 8 月有 2000 万条通讯记录，即使形成作为备份的文本都需要 4G 空间，经过 OLAP Service 处理后仅需 200M 左右空间，因此，经处理后的数据主要存放于另外的相关部门的机器中，而不能与主服务器放在一起。

最后，再来讨论由决策人员所使用的系统，由于这些部门并不分散，我们就没有采用 OLAP Service 的 Web 发布方案，采用 Delphi 编制了访问 OLAP Service 的客户端软件，用了 OLAP Service 提供的、Cube Browser 控件，用相似于网页的界面提供了数据立方体的各种操作，如上钻（观察角度从月转到季度甚至年），切片，旋转等操作，为了便于输出打印数据，还内嵌了 Microsoft 的 Excel 数据透视表，可以将 Cube Browser 上所看到的数据转化为 Excel 的表格形式，或者转换成饼形图、柱形图和曲线图等，比如可以观察每天 24 小时通信流量的分布曲线图，可以发现在夜间 12 点以后明显能信流量减少，而决策部门便可制定某些优惠或减价措施吸引更多客户在 12 点之后使用网络。

另外，在采用 OLAP Service 中的数据挖掘功能时，其中提供的两类算法分别是基于决策树的分类和基于决策树的聚类，市场部门的聚类算法将客户根据费用情况加以聚集，以期发现处于同一消费水平的客户的共同特征，便于制定政策、吸引客户。这方面的努力我们将会进一步持续进行，以保证有足够的海量数据而发现其中的规律。

整个系统运行后，其数据采集，数据处理等一系列工作都由程序定期地自动进行，该系统应用已有一段时间，受到了不少好评。当然，也发现了其中有不少问题，比如：主服务器数据库的容量问题，主站与分站的通信效率问题，还有在网管系统中，网络故障的确定还不够细致，需要由分站再具体化加以确定，决策系统与网管系统之间还缺少直接通信手段等，这些都有待于进一步的解决与改进。

实时控制系统与企业信息系统集成化是推动从事生产制造、测量与监控等业务的企事业单位真正迈向信息化，提高工作效率的一个重要动力。如果是大型企业，更需要有一整套的系统，支持 Web 发布，智能查询，自动识别如用于故障预测和数据挖掘等技术，从而能够将底层的实时监控与高层的决策更好地集成在一起。展望其前景，无疑是十分美好的，但是我们认为相应的工作量很大，在技术上仍然需要有所提高和有所突破。

论实时控制系统与企业信息系统的集成

【摘要】

本文以一个信息化改造项目的为例讨论了实时系统与信息系统的集成。我曾参加了一个中等规模的现代化生产企业的数字化改造项目，该企业拥有 4 座自动化连续式工作的窑炉，以及 8 座自动化间隙式工作的窑炉以及多台半自动的中大型辅助机器。该企业希望能将这些设备实现数字化，并且重点要建立起一个中央监控室，能实现对设备的运行状态参数的监督和记录两大任务，前者用于防止意外事故，后者可用于向该企业决策人员和技术开发部门提供信息。

通过我们的开发组与该企业相关人员一起努力，分四个步骤共同完成了这一工作。

第一步是实现设备状态参数的数字化输出；第二步是建立中央监控室的监督和记录功能；第三步健全监控室的控制功能及相应信号的输出；第四步则是实现生产设备自动化控制的数字信号接入功能。

我在其中的主要工作有三个方面：

- (1) 作为公司开发组和企业间联络桥梁；
- (2) 负责确定该项目中各部分之间的分工，在发生冲突或出现问题时提出相应的具体解决办法；
- (3) 帮助解决与协调在工作过程中出现的各种困难。

【正文】

现代化企业发展生产与提高效率的根本途径之一是加速信息化的进程。在所从事的专业生产领域中，我参与开发项目的这家企业可以认为已经具有相当程度的现代化的基础了，比如它已拥有 4 条自动化连续式工作的窑炉、8 座自动化间隙式工作的窑炉和多台半自动的中大型辅助机器。但是这些设备的自动化控制在改造前还主要依靠模拟量控制，也不具备信息与数据的记录、汇总与分析功能。该企业一方面出于对今后发展的需要，希望记录下这些设备在工作过程中连续的状态参数的变化情况，有运行的日志与历史记录，以提供其技术开发部门，作为产品质量改进研究中的参考；进一步还可提供给企业管理部门决策分析时的参考。另一方面，企业希望能够对设备生产状态有全面的监督和一定的紧急控制与应度的能力，能对生产设备的操作意外和设定不当，或者发生突然的未预料到的事件，防止造成事故与损失。

我们根据该企业的要求，结合项目的资金、时间、人员等现实状况，再三考虑了该企业的经营情况、产品的市场和前景、项目开发所面临的风险等诸多因素，经过仔细分析，得出了如下的 4 条意见：

(1) 由于资金的限制，切实地在相应各个环节上节约成本是相当重要的，因此要尽可能地在原有设施与条件的基础上进行改造，而不是进行根本性的替换；

(2) 此企业需要的是“实时控制系统和企业信息系统的初步集成”，而不是一个功能相当丰富和完善的系统，该企业现阶段既不具备开发这样一个系统的能力和条件，也不具备管理维护和应和高级集成系统的相关人员，所以，项目的目标应当切合于目前条件下企业的总体要求。这样既有利于控制成本，也有利于减少项目风险；

(3) 由于该企业的生产情况和资金、人员的限制，项目必须分阶段地进行。大体上可划分为如下四个阶段：①实现设备状态参数的数字化输出；②建立中央监控室的监督和记录功能；③健全中央监控的控制功能和相应信号的输出；④实现生产设备自动化控制的数字信号接入功能。

(4) 参与本项目涉及到的双方的大多数人员都不精通对方的专业领域，因此必须在加强互相沟通的同时，确定明确的分工关系。

上述四条意见在经过双方的磋商与研究后，获得了双方全体项目参与人员的一致认同，成为这个项目开发过程中双方必须理解与遵循的准则。

在第一阶段，我们开展了对半自动的中大型辅助机器的自动化改造。事实上，该企业早在这类打算，并且已做了相应的技术储备，因而这一部分的工作由该企业自身的技术人员全权负责并加以实施。项目中所涉及到的所有自动化生产设备都已具有依据状态参数模拟信量进行控制的能力，对于所采集到的状态参数模拟量，企业曾计划采用一类以模拟信号远程地传至中央监控室，再进行模数转换的方案。此方案对企业来说实现比较简单，但存在着成本较高、远传过程易受到干扰等不利因素。随着模数转换设备成本的显著下降和可靠性提高，经我们建议和双方讨论，企业有决心在生产设备的控制设备上就地实现现场模数转换，再远传数字信号至监控室，这一工作同样地由熟悉这项技术的企业技术人员实行。

第二阶段的工作主要由我方开发组成员负责。我们将人员大体上分为 3 组，第一组主要是根据企业长期累积资料以及公开发表的相关技术，建立起一个合理有效地模型，其中包括诸如数据采样记录的间隔时间，不同生产阶段的数据处理时所采用的数学模型等数据处理的相关内容；第二组负责监控记录软件的输入输出接口，用户图形界面的选定和设计等软件外围功能的实现；第三组则集中力量编写一简单实用、针对性强和小巧的相关数据记录的专用数据库。这一阶段是控制质量和成本的关键性阶段。出于对成本的考虑，以及根据数据的流量不很大，对数据的实时性处理要求不是很高（通常情况下，设备的实时控制仍由原来的自动化系统所承担）的实际情况，中央监控室采用了一套有双机备份的服务器作为数据处理用的服务器，另一套同样有双机备份的服务器作为数据库服务器，并且没有使用价格昂贵的商用数据库，而采用了由自己开发的一个经济实用的专用数据库。

第三阶段可以看成是第二阶段的自然延伸，在第二阶段成功的基础上，利用第二阶段模块处理后所获得的数据，依据设备的多种临界指标，进行相应的判断，允许在紧急情况下，发出相应的警报，并同时依据设备本身的相应紧急情况处理办法，发出控制信号加以处理实现。这一阶段的关键有两方面内容：一个问题是要求数据转换设备拥有相对较高的可靠性与可用性，另一个问题是要注意做好与自动化设备原有控制系统的自我保护功能的配合协调工作。

第四阶段则仍然由该企业的技术人员为主实施，在实现过程中主要是解决好第三阶段所遇到的上述两个关键问题。对于第一个问题，使用了更好地设备和部件来实现数模转换和动态控制；对于第二个问题，则在控制设备中设立了优先级判断，使自我保护装置的启动优先级离开中央监控室（由于自我保护启动速度更快，但是功能较弱）而加以解决。

从总的项目实施进行上来看，一、四两个阶段相连贯，二、三两个阶段相连贯，而它们之间则可并行地进行，从而满足了时间进度上的要求。

今后，本项目所采用的这类技术可能要走向全自动化。项目中涉及到的数据量将会更大得多，实时性要求也会更高。我们应注意使现有成熟的商业系统与产品如何应用到其中去，使之能尽快地满足企业的要求，节约成本，并且减少开发的危险。

论实时控制系统与企业信息系统的集成

【摘要】

本文通过“工控组态软件”项目的开发，着重讨论实时系统与信息系统的集成。近年来，国内外的组态软件取得了很大的发展，已广泛应用于企业生产。组态软件以实时数据库作为核心技术，综合了工控、网络、图形处理与数据库访问接口等技术，是技术含量较高的一类软件产品，具有良好的应用前景和市场潜力，因此，有多家信息技术公司都在开发工业组态软件。

我有幸参与了该项目，在该项目中担当了分析与设计的一部分任务，该软件采用 Windows 2000 操作系统，主要采用 VC6.0 进行开发。以下本文将从我所开发的组态软件的特征、软件的体系结构设计、实时数据库设计、可扩充性与可维护性设计以及项目实施管理等几方面加以论述。

【正文】

工业控制组态软件在工业界有着相当广泛的应用，此类软件允许用户在图形界面下对控制系统的各种采样点、过程输出点、设备、生产车间、控制回路、文件报警、生产报表、控制策略、网络设备和生产工艺画面进行定义与组态。使用该类软件时，用户甚至可以不写一程序就能够构成自己的控制系统，有些功能强大的组态软件还可提供与网络、Internet、数据库访问接口等的连接功能，使现场控制系统能相对方便地和企业的信息管理系统加以集成，某信息技术公司决定开发新的具有一定通用性的工业组态软件，作为技术骨干，我在该项目中担当了分析与设计的一部分任务，该软件采用了 Windows 2000 操作系统，主要采用 VC6.0 进行开发。

本文将从我们所开发的组态软件的基本特征、软件的体系结构设计、实时数据库设计、可扩充性与可维护性设计以及项目实施管理等几方面加以论述。

1、我所从事开发的组态软件的基本特征

通过分析国内外的组态软件的特点和当前的技术发展情况，我认为我们着手开发的组态软件应当突出下述三个特征。

(1) “实时与可靠”是此类软件赖以生存应用前提，但是目前还是有很多的组态软件做不到这一点。

(2) 具备良好的网络连网能力与分布功能。

(3) 有效地采用 ODBC (开放的数据库连接)，便于和其他信息系统集成。

这个项目在技术上，应着重于组态软件的体系结构设计与时时数据库的设计，需求分析则应着重分析国内外同类软件的功能，通过比较与鉴别，才能产生真正优秀的软件。

2、组态软件的系统体系结构

本软件采用的是三层体系结构，设计结构时要具有开放性和良好的可扩充性。

(1) 软件的底层是硬件访问控制层。这一层所采用的是前几年才推出来的 OPC (OLE for Process Control) 技术，采用该技术的好处是 OPC 是微软参与制定的标准接口技术，有众多的硬件厂商支持，所采用的 OLE 技术使软件具有良好的适应性和扩展能力。

(2) 中间层是实时数据库。该层是整个系统的核心，在设计上除了具有一般实时数据库具有的特性之外，应当为应用层提供了两类接口：一是应用编程接口 API (比如以 DLL 的方式实现)，二是 ODBC 接口，该接口使系统具有很好的开放性，便于系统集成。

(3) 上层是应用程序层。在该层通过 ODBC 接口访问实时数据库，可以通过 SQL 语句查询数据库的数据。

3、本项目涉及到实时数据库设计

在设计时，我们着重考虑了以下的四个方面：

(1) 实时数据库的基本功能：实时数据库完成实时数据库的采集、输出、报警文件等的管理，也进行历史数据的管理。

(2) 实时性设计：由于本系统所采用的操作系统是 Windows 2000，它的实时性较差，因此要求任务管理定时器必须具有良好的实时性，在系统设计时，我们采用了抢占式服务的高精度定时器，在一定程度上保证了系统具有良好的实时性。

(3) 任务调度：其目标主要是使系统在各时间段达到较理想的负荷任务的均衡性。

(4) ODBC 接口设计：即开发相应的驱动程序，实现 ODBC 功能，使之完全遵守 SQL 约定，这样能允许应用程序的开发手段和开发工具多样化，允许可以采用 VC、VB 或 Delphi 等作为开发语言，也使数据库具有很好的开放性。但 SQL 语句不能实现数据发生时间方面的选择，影响了实时性，因此，系统自动给每个数据库加上时戳，SQL 可通过时戳进行时间控制来选择 (读取) 数据，从而满足了实时性方面的基本要求。

4、本系统的可扩充性与可维护性设计

组态软件综合了多种技术，其体系结构与数据结构都较为复杂，再加上我们又希望能适应的实际应用场景有着复杂多变性，因此要求系统必须具有良好的可扩展性与可维护性，以满足功能与性能上不断变化的要求。在系统的设计技术上，我们大量地采用组件技术，如 OPC、COM/DCOM 与 3D 图形控件等，组件技术的采用使系统具有了良好的可扩展性与可维护性，降低了系统的复杂度。而且也使我们较方便地获得第三方支持，例如，请经验丰富的图形处理专家编写图形处理控件，就能加快软件开发的进度。

5、本项目中软件项目实施和管理

组态软件的需求在当前工业控制领域中是较成熟的，基本能满足一般用户的功能上需求，通过比较多家组态软件，可以发现：在它们之间有 80% 的功能是相同的或雷同的，由于我们项目开发的起步较晚，在自控领域里，我们处于劣势，因此我们提出了“重技术分析，轻需求分析”的思路，即把重点放在组件设计与体系结构的实现上。

在人员的配备上则根据组态软件的技术组成特点，组织一批在自控、网络、组件、实时系统设计和硬件上各有所长的 VC 高手组成一支精干高效的队伍。

在开发进度上则反复强调“质量第一，进度第二”的原则。

在我们的项目实施中，可靠性作设计的首要原则，要求项目组成员养成良好的编程序习惯，每天必须完成认真的工作日志，每周要写工作总结，完成一段程序代码之后，即应自己先进行从里到外的测试，只有从基础抓起，才能保证组态软件的质量。

通过本项目的开发成功，我深切地体会到要使组态软件在企业实时控制与信息系统集成中发挥其应有的作用，必须注意以下各点：先进的体系结构，支持 ODBC 的实时数据库；强大的网络功能；功能日益强大的脚本语言等。我期待着本人通过在这个领域中的辛勤耕耘，将会结出更多更丰硕的 IT 成果。

论软件质量保证的方法和策略

【摘要】

软件质量的优劣已经直接关系到软件项目的成败。这里总结了作者在软件系统开发中为了保证软件质量而采用的方法和策略，主要包括：坚决贯彻工程化开的原则；重视文档建设，以及文档和代码的一致性以及版本控制和跟踪管理；重视工程每一阶段的质量控制，包括计划、评审、变更管理、责任制的明确；重视人员的管理机制；以及在软件质量的量化控制方面的努力。软件工程的科学理论在不断地发展，要提高软件质量，需要从整体的软件过程和管理模式入手，采用科学的工具。

【正文】

软件作为一种产品，现在已经有了规范化的方法以工程化的思想去“生产”，它再也不是个人智慧的小产物，它经常是以若干人年的大投入去生产，而且它运作的性能和可靠性往往有着至关重要的意义和后果。所以提高软件质量已经成为整个软件开发过程中的关键问题之一。

我曾经主持开发了一个大型连锁经营的贸易企业的 MIS 系统的开发，参加了从最初的需求分析到最终的运行维护的全过程，主要担任开发方的项目负责人，负责项目开发的管理，人员调配，进度控制以及与客户方的联系和沟通工作。一个大型的 MIS 系统的开发，要保证最终的软件产品成功地实际运行，达到需求的目标。首先我就坚定一个信念，要以工程化规范的 SQA 来把握每一个步骤，自始至终，绝不马虎。最终我们的系统成功地替换掉旧系统，至今已经基本正常地运行。

这里在保证软件质量方面有一些做法总结如下：

1. 坚决以工程化的方法，按部就班地开展每一步工作，绝不急于求成。从最初的需求获取阶段开始，首先是双方决策人员和技术人员的沟通，尽量在战略目标上统一，在总体的资金投入、人员投入和系统目标上达成一个尽量明确的共识；然后进行可行性研究（在资金、时间和人员的限制之下是否可以达到系统目标）；在此基础上再进行需求分析和开发。经过不断的修正，得到一个适当的原型。然后开始总体设计和详细设计，以及代码设计和各级测试。先产生一个可试运行的产品，再不断地完善它。在每一个步骤中，都保证每个步骤的工作质量。

2. 重视每一个步骤中的文档建设，以及文档版本管理和文档与代码的一致性管理。这里我们采用了需求变更矩阵来控制并记录文档的变更路线和一致性。每一个步骤的文档和代码都有明确的版本代号，每一个版本的文档和代码的变更原因、提出变更的人员、对于变更请求的讨论意见记录、最后是否采纳变更请求都有相应的记录。如果采纳，则选择实现的相应措施、变更将要影响到的模块有哪些、相关的人员有哪些，全部都要记录在案；然后每个相应模块的实现情况以及相应的后续跟踪情况全部记录在案。对于那些已经接受的功能请求，它的完成情况、可能出现的潜在问题全部都在专门的会议上作确定，并有详细的文档记录。每一项决定的负责人都有记载。对于文档和代码的版本都有一个矩阵来跟踪和管理。

3. 重视工程每一阶段的质量控制。每个阶段都有详细的计划，包括此阶段的目标，工作内容，参加人员，每个人员的工作内容和目标，以及每个阶段的时限，每个人员的每项工作的时限。一般计划到每人每天。每三天有一次小组碰头会，总结计划完成情况，及时发现问题，尽量解决。影响较大的问题就要及时召开项目会议，作讨论，给出解决的方案。每周有一个开发会议，作周汇报，及时审核计划的完成情况。在有突发事件时，要求当天召集人员解决，绝不“留待以后”，造成祸患。每个阶段完成时的阶段总结是非常重要的，要求开发方和客户方的领导和技术人员都要参加，尤其对于有变更和有分歧的问题，务必要达成明确的共识。这样，每个阶段每个人员的计划、目标、工作内容，出现的问题，解决的情况，还可能会有潜在问题以及给予的评价全部都有详细明确的记录以供跟踪。这样一切都尽量做到有章可循，并且在有人员变动的情况下，可以把对开发工作的影响程度降低到最小。

4. 重视人员的管理。软件产品毕竟是由人生产出来的。人员管理直接关系到软件本身的质量。我要求每个人都要就自己的工作负责任。人员管理按照总负责人到项目负责人，到下层的开发人员，协调人员划分，既有从上到下的责任层次，又按照系统本身的功能模块分块。每一个人既对上级领导负责，又对横向的分块的项目组长负责。责任明确到人，到天，按照完成情况进行考核。以人员的责任感来保障软件的质量。

5. 在保障软件质量的量化方面做努力。尽量采用当前先进的软件质量度量方式，以科学的方法试题和检测软件的质量，而不仅仅是从定性的方面来衡量。无论是需求开发的产品，还是设计阶段的产品，当然也包括代码，以及相应的文档，每个阶段的产物都要在阶段总结的时候给予定量的质量考核和评估，然后对质量评估的结果给予分析和评价。对于不达标的阶段产品要给予处理意见，力争做到步步为营。

为了提高软件质量，与那种无章可循、个人说了算、没有严格的进度控制、没有严格的质量审查的老观念作斗争，我在这次系统开发的整个过程中，自始至终以工程化的 SQA 的观点出发，使得这次系统开发无论是资金分配、进度把握、系统功能的变更控制；以及人员调配都优于以往的系统，最终能够按协议中确定的目标和资源限制成功地完成。不过，软件质量在开发的各阶段的量化上还是需要进一步采用先进的方法，努力在软件过程管理上按照 CMM 的规范，从而有效地提高软件的质量。而且还要在采用相应的先进管理工具方面做努力。我们已经开始使用 Rational 公司的软件工程管理套件，争取在过程管理和质量控制方面更上一层楼。

利用测试提高软件质量

【摘要】

本文以某证券公司 Internet 应用开发中的软件质量保证实施的过程，讨论了项目实施过程中为保证软件质量所采用的主要技术和方案。

众所周知，测试是提高软件质量的一个重要途径，也是提高软件系统可靠性的重要方法。自中国加入 WTO 之后，该证券公司加强自身在 Internet 领域的竞争力，决定升级其软硬件系统平台。其软件平台是 J2EE 为构架的应用，升级的前提是尽可能和系统原有的软硬件平台兼容，最大限度地保护企业原有的投资，升级的目标是用户提供一个更加快速的交易平台，该平台具有更大的系统吞吐量，并发用户数量和更高的效率。为此我们制定了以下六个步骤完成对系统的测试：测试计划的制定、测试数据的准备、测试环境的建立和测试前数据的备份、测试、测试记录和结果分析、测试验收和试运行。

我在项目中担任了测试计划的制定，测试具体操作的进行等工作，下面就以这两个方面在正文中展开阐述。

【正文】

对以证券行业为代表的关键领域而言，其基本特点是：实时性强、交易不允许停机、瞬时数据量大。如果出现服务器或柜台机停机，或者数据丢失，则无论在声誉或经济上都将造成巨大的损失。所以在系统测试之初，我们就把确保系统的可靠性、安全性和效率放在了首位。证券企业的业务流程、工作方式有其自身的特点，因此为了确保软件质量，我们要求测试人员对相关的业务流程要了如指掌，这样在确定故障点时才能“胸有成竹”。

新的升级平台包含了硬件平台，而证券企业内部的网络测试工具和技术力量较薄弱，为此我们将硬件平台的测试任务让省外某知名网络测试实验室来负责，实施从网卡、交换机、路由器、外接设备、防火墙等全线产品测试，并且还对网络数据流量模型、网络性能以及优化、网络安全性等方面进行了较全面的测试评估。而基于 J2EE 构架的软件平台的测试则由我们企业内部人员和聘请的专家顾问完成，主要测试 EJB 的性能和负载、Web 站点的网页链接、Web 站点安全性等诸多方面的问题。

通过仔细分析业务流程和企业自身的特点，我们根据相应的环境确定了软件测试的工作量；再由测试要素确定测试用例和测试脚本。这里所说的测试要素是指诸如系统环境、类的接口和类型，程序的规模和复杂度，程序语言选择等方面的内容。测试范围主要是软件平台。在实践过程中，我们将测试人员的队伍不仅着眼于测试工程师中，而且将相关业务人员、监督人员也邀请到我们的测试行列中。这样做的原因是，业务员是功能模块的最终使用者，他们可以参与到测试数据和测试用例的准备工作中，并且可以分析相应的测试结果，发现一些和业务相关但被测试人员忽略的问题。而监督人员则是部门负责人，他们能够在测试进行时协调好相关部门人员的关系，并监督测试工作进展。如同 ERP 系统要由一把手来协调上下关系一样，测试工作也要领导的支持才能顺利完成相关进度规定的任务，为测试和软件质量保证打下良好的基础。

在具体测试过程中，我们分阶段分步骤地采用了一些先进的工具和方法策略。为我们测试成功奠定了基础。

软件测试指出我们选用的是 JUnit 测试工具，用于测试 EJB 模块组件，主要任务是检查 EJB 是否能完成其所对应的逻辑业务功能。在测试过程中，我们发现所需使用的测试用例中不仅包含普通类型的数据，而且还包含对象类型的数据，对象类型的数据输入仅靠人工来进行是非常低效的。为解决这个问题，我们测试小组成员经过和软件开发人员探讨，决定将美四小组提出的“设计模式”中的生成器（Builder）模式应用到测试中去。具体的做法是，使用导向器（Director）将要生成的对象类型数据——TestCaseObject 所拥有的属性和数据从文件中用 Director 类中的 readFromFile() 方法读出，放到 ArrayList 中，然后传递给类 Builder 中的子类——builderTestCaseObject 中去，由该子类生成相应的对象并返回给 Director。由于对象的创建过程代码由 Director 类负责，而生成由 Builder 的具体对应的子类负责，这样我们就可以重用对象类型的数据。这个程序不长，但将测试人员从繁重的手工机械重复的录入工作中解放出来，并且将软件开发复用领域里的模式引入到测试领域，激励了测试人员的积极性和创造力。

在完成初步的单元测试之后，我们又使用了美国 ISA 的 Panorama 软件工程和自动化测试工具。一个好的测试工具不仅能发现软件中的错误，而且能够定位这些错误，而修正这些错误并确保这类错误不再引入则是测试人员所渴望的。我们面对的是一个有近千个 EJB 组件的软件框架，每个组件的方法调用或多或少和其他组件有着联系。我们若改动一个方法调用、全局或者局部变量，势必将引起方法调用和变量在其他模块中的不一致性。如果仅靠人工搜索定位语句统一更改将是十分枯燥的工作，同时会产生较大的反复性。Panorama 在这一方面相当的出色，测试人员不仅能够精确得到导致软件故障的语句代码，更能从 Panorama 测试工具中得到修正后的方法调用会影响到哪一个位置的一份清单。这样我们测试人员可以提交一份相当详细的测试结果报告给软件开发人员修正，这样就确保同类型的问题在软件构架的大范围下统一解决，保证回归测试中不再出现同类问题，减少了反复性。而在测试用例方面，我们使用 Panorama 的测试覆盖功能从类、函数、块、段、边界循环条件、判定等诸因素多层次地确定测试覆盖能力。我们在实践中发现，面向对象系统测试用例大约只有 10%~15% 是效的，当测试覆盖在段中达到 50%~60% 时，随机生成的所有测试用例都是重复的。而利用 Panorama 提供的最小化测试用例，则有助于我们在修改模块之后，避免重新生成新的测试用例这一重复性劳动，由 Panorama 自动调用先前的最小化测试用例来进行回归测试。实践证明这是非常有效的。

基于站点的性能、负载测试，我们使用了 WebLoad 测试工具，对 EJB 分别进行了性能和负载测试。这两个测试看上去相似，但有区别。性能测试是由测试工具模拟单机检查 Bean 的响应性能如何，即从开始调用 Bean 或 Bean 中的方法到调用结束所花的时间。测试中我们发现一些 Bean 的响应性能较差，在经过了代码整理或优化算法后性能往往有了显著的提高，因此这一阶段的修正主要是代码的审查。测试中发现的一些问题，诸如一些没有必要创建的对象被创建了，一些线程要等待其它线程结束才能结束等，这些问题都由测试人员仔细记录到测试文档中，交由软件开发人员进一步处理。对于负载测试，使用测试工具模拟多用户访问站点，以检测站点所能容纳的极限访问量，或检查在用户开发数目增大的情况下，系统的事务处理能力、吞吐量、内存利用率等数据的变化及其原因。测试之初，系统不可避免地存在响应较慢的问题，我们知道这是由于客户机或线程争用系统资源（如代码中的资源，数据库后台资源等）而导致的。我们采用逐个隔离、逐一排除的方式找出平均响应最慢的组件以确认其为瓶颈。此类测试的体会是在访问之初，不要将测试的时间作为衡量的基准，因为此时 Web 服务器中的内容没有缓存到内存，且测试响应时间要多项取平均值作为标准。限于篇幅，Web 站点网页链接测试，安全测试不再叙述。

此次软件测试，对确定软件质量保证起了很大的作用，也使我受益匪浅，让我对质量的概念有了深层次的认识。但同时，我也认为自己还有很多领域和知识点要继续研究和实践。随着 UML 的流行，面向对象的测试要求也越来越高，一些新的领域，如软件复用领域等对测试提出了新的要求和挑战。理论和实践结合，不断探索和突破自己，应该是每一个工程技术人员内在的要求。

应用 CMM 保证软件质量

【摘要】

本文论述了如何在一个规模较大的网上管理系统的开发中结合CMM二级的框架要求，以及软件工程学的质量保证策略进行项目的软件质量保证工作。

本项目的特点有：

(1) 开发人员多，有40人左右；

(2) 采用面向对象分析与建模技术，JAVA语言，WebLogic应用服务器等以前项目中未采用过的开发模式和技术。因此不确定性因素很多，急需采用有效的质量保证策略。

公司为了提高软件开发能力，已经于近期全面引入了当今软件界正在流行、且行之有效的CMM质量保证体系，并在顾问公司和主评审员的帮助下，由公司的SEPG结合公司实际，制定了初步的规范体系和模板文件，并决定将本项目作为试点项目。

本文详细论述了作为SEPG负责人之一，并且担任本项目质量保证人员的笔者，是如何在本项目中有效推行CMM二级质量保证措施的，并指出了其中的经验教训和有关的建议。

【正文】

本项目是一个面向政府管理部门，全市房地产企业和个人用户的网上管理系统，它既是一个电子商务项目，又是一个电子政务项目。本系统采用了B/S结构，融合了政府部门和房地产企业的内部网上管理系统于一身，同时作为全国建设系统信息化的一部分而实现Inter-net平台上的上下集成。它还提供信息发布、房屋交易等电子商务功能。

本项目对系统的安全以及可靠性等方面有着较高要求，公司决定采用三层架构模式的J2EE环境作为运行环境。另外，本项目参与人员众多，面临着新技术、工期紧等影响软件质量的不利因素，对软件质量保证工作提出了很高的要求。本人作为公司CMM实施工作的负责人之一，以质量保证人员的角色参与到项目的开发和管理工作中，主要负责质量保证策略的建立以及实施工作。

一、以CMM二级理论为指导，采取措施保证开发过程与开发规范的符合性，以过程质量的提高来保证产品的高质量

1、建立起明确的权责制度，减少因权责不明而产生的混乱

为确保质量保证人员有独立的途径向公司反映开发中的问题，同时为了避免质保经理与项目经理、软件配置经理之间过分隔阂，本人在项目启动阶段，起草了《项目管理人员责任书》，并由相关人员评审通过；并提议公司设立了高级经理交涉，以快速处理纠纷。

为防止开发人员与项目管理人员在规范化开发过程中过度依赖规范而主动性不高，以及可能产生的相互埋怨，制定并实施了《项目开发人员守则》，由各项目级每个成员在参加项目之初进行签名式确认，以解决开发过程中应规范滞后和实施不力而引起的混乱问题。

2、以制定的《质量保证计划》为纲，全程监控各开发工作的过程建立和符合性问题

在项目启动阶段，就依据CMM二级要求和公司发布的项目开发规范，制定并基线化《XX项目质量保证计划书》，在计划书中详细地制定了质量保证工作的内容和进度安排。

计划书中主要有职责、培训工作、检查评审及组织工作等四方面的内容。

职责方面详细说明本人作为质量保证人员在项目中的全责以及主要活动，澄清了与各开发角色的关系，主要起到项目成员监督质量保证人员的作用。

培训方面指明了为有效推行CMM质量体系而进行的有关培训，有CMM基础理论方面的，也有本项目特色的规范方面的培训内容。

检查评审方面指出了要检查评审的过程及提交产品，并列举了相应的通过准则，即CHECK-List。比如要评审的过程有项目规划阶段；检查项目经理和配置经理是否按有关规范制定了各自的计划书；项目组的技术评审活动是否符合评流程和规范；风险分析过程和任务分解过程是否符合规范的执行。对提交的工作产品，如需求文档和设计文档，是否经过了正式技术评审并基线化。这些都指明了切入时间和建议人员。

组织工作方面指明QA（质保人员的简称）在开发过程要做的组织工作，如技术评审工作、测试工作、估计和工作细分等工作。这些组织工作主要是为了协助项目经理开展工作并能有效且及时地获得第一手质量方面的资料。

对项目开发过程中的跟踪和检查，主要采取了现场参与、分析项目成员日报和周报、个别交流以及项目周例会的形式。

二、以RUPCN和软件工程方面的理论为指导，制订了行之有效的技术规范文件

CMM质量体系更多的关注软件开发过程方面的事情，也就是建议由谁在什么时候做哪些工作，但没有指明各个工作如何开展，也就是偏重于管理，偏轻于技术指导。为了避免在框架方面很有效，但应实现细节不明确而出现的“一条腿走路”的现象，特在项目启动之初就制定了要建立的技术规范，如需求文档编写指南、界面设计规范等。这些都列入QA的计划中了。

考虑到本项目采用了面向对象的分析和设计技术，急需UML和Rose技术方面的指导性文档，因此将与UML高度相关的RUPCN体系为主要参考，其他软件工程理论为辅助参考，组织资源开发人员制定了各个开发阶段的规范性和指导性文档。

实践证明，项目组成员有了CMM过程方面的规范，又有了指导开发工作的详细技术文档后，开发质量有了质的提高。主要体现在以下几点：

1、各个过程的效率提高了，从而保证了各个交互成功的质量。

比如，人员的选择，时间的必备，开展的流程方面真正按照CMM的建议做了，这些都保证了开发过程的高效。一个显著的例子就是技术评审过程。如果选择的人员资格不够，所花的评审时间很少，开展的时机不对，就很难保证提交成果的质量。

2、人员之间的交流变得有效，自信心更强了。

三、不足亟待改进之处

由于是初次按照CMM的要求在项目中实践QA工作，因此不可避免地有一些不足之处，主要有：

- 因培训工作的不及时、不全面，导致QA实施工作遇到了一些麻烦。
- 质量度量数据的收集和分析工作做得还不够。仅仅有少量的度量数据，分析工作也很少。总的原因是缺少一个有效的度量数据收集和分析机制。

总之，这次项目的质量保证工作基本达到了CMM二级的框架要求，证明了过程质量是产品质量的重要因素，为本项目顺利地、高质量地完成做出很大的贡献。但也有很多教训值得吸取，需要在培训工作和QA度量数据收集和利用方面下更大气力。

护士工作网站系统中的质量保证

【摘要】

软件的质量就是软件的生命，是软件项目得失成败的关键。本文从一个真实的开发项目《护士工作网站系统》中描述软件质量保证的方法和策略。本文认为保证软件质量主要狠抓三个方面：有效控制软件开发的过程；内部有角色职责的组织；对开发过程和开发人员的恰当管理。这三个方面需要有机结合才能有效地保证软件质量。护士工作网站系统是医院信息管理系统的重要部分，本人在总结以前的经验教训的前提下，根据以上三方面工作，有效地控制了软件的质量，成功地使系统上县，得到用户的肯定。

【正文】

本人所属的公司是开发医院信息管理系统的，我担任该项目的负责人，具体工作包括开发人员的管理、系统分析及软件质量控制。

在系统开发的初期，我们开发了门诊系统，其中包括挂号、收费、门诊药房等子系统。我们当中不乏技术过硬的开发人员，但当系统上线时，软件的质量并不理想，其中包括一些明显的错误。如挂号重号、收费累计出错等；而且文档不齐全，造成软件维护比较困难。之后，我们开发护士工作站时，本公司几个技术骨干总结了以往的经验教训，得出以下结论：

- 1、在开发的过程中缺乏对开发人员的控制，如技术控制、文档控制、测试的控制以及版本控制。
- 2、开发人员的角色功能没有细分，即各开发人员职责不明确。如需求分析、设计、编码和测试的角色没有详细划分。

针对以上结论，本人提出，要保证软件的质量必须要做到以下两方面：加强对开发过程的控制；要有清晰的人员分工。这是根据ISO标准和CMM的指导精神，软件的质量存在于软件开发过程的质量，对软件生产过程进行改进，就是对软件质量的改进；提高对软件开发过程的控制能力，就能控制软件的质量。基于这个原理，本人公布了以上两项措施，得到广泛的支持，并共同制定了具体的实施方案和人员组织方法。

首先定义出开发中最主要的过程：系统的计划、需求分析、系统设计（包括概要设计和详细设计）、编码、系统测试，并对每个过程参考软件工程继续筛选，以适应本公司的实际需要，确定每个过程的具体工作内容。

其次确定了软件开发的具体开发方法，如采用面向对象的方法和结构化的方法。最后确立了面向对象的开发方法，并对每个开发过程定义了相应的具体文档，如系统计划必须提交《系统开发计划进度》，需求分析过程必须提交《软件需求规格说明书》等。并定义了文档的格式和必须提交的内容。

最后对每个过程发生相应的人员职责，即指定人员，人员的过程中扮演的角色及相应的工作要求，人员需要的相关知识及具体工作内容。

但制定这些实施方法和规章制度并不是一朝一夕可以完成的，我们总结了以往失败的经验教训，进行了认真的思考和论证，历时三个月，而这这套方案和制度制订出来，并实际实施我们开发的住院系统中，取得较好的效果。以下以开发护士工作站为例说明运用这套方法取得的效果。

我们在实施过程中改变了我们的方法，就是加强了对开发人员的管理。在开发的初期，我们和医院方面取得联系，并初步了该系统的功能目标，根据本公司的实际情况制定了系统开发计划。然后按照我们制定的过程，需求分析——系统分析——编码——测试——实施——维护进行开发。在整个过程中，我们得到所有的文档，并根据文档反映的内容对每个过程进行了检查。基本上可以按照我们预定的目标进行。虽然最后进度有所延误，软件的实施过程中还有错误，但所有这些都在我们的控制范围之内。

在实施这个质量过程中，我认为比较有独到之处的是加强了对人的管理。其实，开始实施的时候，我并未十分注意。在需求分析的时候，系统分析员虽然按照预定的方法和内容进行了工作，但对文档不够重视而且没有得到用户的确定后就马上进行设计，这样在设计过程中遭到用户异议，并造成了设计人员的概念模糊。我们马上回到需求分析过程，对分析员的工作认真检查，对文档进行严格的审核，终于完成了一份较完整的《软件需求规格说明书》。自此以后我对每个开发人员和每个过程进行认真管理和检查，确保文档的质量和人员的工作质量。最后软件完成后，其质量也得到了用户的肯定。由此我得出了以下体会：别人的工作质量控制也应该是过程质量控制的一个重要组成部分，而且是关键。开发人员的工作质量有了保证，过程质量也有根本的保障。光有制定的方案是远远不行的，关键是人的实施，把人实施过程的质量控制好了，过程中产生的文档质量也有了保障，软件质量也相应地有所保证。总之，软件质量在于过程质量，过程质量在于工作人员的工作质量，这是我在实施软件质量过程中最大的体会。

软件质量保证是一个很大的问题和实践方向，这是我在实施过程中的一些体会和方法，还有许多改进和完善地方，在以后的工作中，只要抓原则，逐步实施，我相信我们开发的系统有更进一步的质量提高。

论中间件在 SIM 卡应用开发中的作用

【摘要】

我曾于近期参与过一个基于SIM卡应用的开发项目，并在项目中担任系统分析的工作。在分析过程中，我们依据面向对象方法对系统进行了划分，其中许多的模块已有成熟的中间件产品可供使用。我们选择中间件产品的标准是：功能、性能、封闭性、独立性、可扩充性、是否标准化等（或是否是主流产品）、跨平台性、话语言性，以及中间件的大小和价位，并且面向对象的优于面向功能的。

由于复用了大批的中间件，使得项目工作量大大减少，开发周期明显缩短，并且在项目的编码部分，我们也尽量制作自己的中间件，以便在以后复用，或以恰当的形式销售，增加公司的利润。

自己设计中间件，首先要做领域分析，以确定其价值，并依据上述原则开发。产品形成后，要由管理系统负责中间件产品的存储、描述、查询、发布以及动态维护、版本控制等。

【正文】

中间件技术是近几年随着软件工程技术的发展，网络应用的普及而迅速发展起来的，其在软件开发中最突出的贡献在于资源复用。随着分散的、作坊式的、独立的软件开发方式向集中地、大型的、协作式的软件开发方式的转换，软件的划分向着高内聚、低耦合的方向前进。类似的，重复性的、独立的软件部分被抽取出来，被做成模块、对象以及中间件。我们利用中间件拼装产品，共享着人力资源和时间资源，减少了重复劳动，减少浪费，增加了价值。由于中间件的出现，软件开发可以变得十分专业化。例如通信中间件、数据库中间件，……，他们屏蔽了许多繁琐的细节，使得应用程序开发人员质量与优化业务逻辑。各种不同的劳动量得以在自己熟悉的领域得到发展，不用分散精力，这就是分工带来的优势，专业化带来的好处。如今，在计算机软件科学中，这就是中间件带来的好处。

我曾经参与过的某个项目在这方面带给我深深地感触。这是一个关于SIM卡的应用，其功能要求是：在SIM卡上存储信息并与计算机内的信息同步；利用SIM卡的信息实现网络安全控制。

我作为系统分析人员，采用面向对象方法进行分析，这里只作简要概述。

在该项目中，我抽象出的一个核心对象是“信息对象”。通过读取SIM卡/计算机内信息，可构造出该对象实例，通过将该对象实例更新入计算机/SIM卡完成信息同步的功能。而实现网络安全控制的功能则通过验证对象属性来完成。

其中读出SIM卡、写入SIM卡的这部分，已经存在大量的中间件产品，我们选择了微软公司的系列产品，带给我们极大的便利。因为我们不再花费精力去考虑如何编写代码以及与SIM卡读写设备驱动程序进行通讯，如何将信号转换为二进制数，或转化为字符。它使我们摆脱了对设备物理细节的考虑。

此外，在读写计算机内数据时，利用了数据库中间件，此处选择的是微软的ODBC、ADO系列。在进行用户界面设计时，也应用了许许多多各具体特色的图形控件。

而在实现SIM卡网络安全控制时，选择了SUM公司的基于JAVA的SIM应用套件、SSL网络安全协议的辅助构造我们的项目。

在这样的选择之后，遗留给我们的仅是一些信息管理以及在各中间件进行协调、调度的工作，整个项目因此而显得异常轻松。

如果在没有中间件可供使用，一切需要从零开始的境遇中，这样一个小项目将会变得多么可怕！除了劳动力资源的严重消耗，还带来了许多隐患，例如与别人的产品的协作性问题。因为如果对硬件设备的处理不同，网络传输的各自构件的协议不同，根本不存在互通性。

因此中间件在软件模块标准化、软件复用性、跨异构系统协作方面也功不可没！

在我们选择中间件产品的时候，最先考虑的是功能性。因为只有项目缺少的功能，才有被使用的价值。而且中间件产品性能也是可以影响项目的因素。此外中间件是否采用标准和通用的接口形式，其封闭性、独立性、扩充性、跨平台跨语言的能力，也是影响我们项目的稳定性、可扩充性的关键因素。另外产品的大小、运行环境以及价位也是我们考虑的因素，这影响到项目的成本和效益。

如今的主流开发技术都支持面向对象和中间件（组件、控件、服务）。大家都意识到了复用的重要性。如果由我来决策，我会选择在开发项目同时开发中间件产品。虽然这样的开发模式在单个项目中所耗资源较大，因为要考虑中间件的各种特性，如独立性、通用性等。但就长远目光来看其复用带来的成本节约以及项目周期短、开发模式标准化、通用化等利益，将极大地提升企业利润。

为了能够开发出有价值的中间件并同时进行高效地利用，我们注意以下方面：

一、在决策开发中间件时，一定要做好领域分析。在相关领域内通用的中间件才有可复用性，才有价值。

二、对于中间件的策划，要注意其独立性，封闭性（接口清晰、标准），最好是以对象的形式开发，提供继承功能，可以提高中间件的可扩充性。

三、最好做到跨平台、跨语言，以增强通用性。

除了这些原则，对于中间件产品有也应有相应的管理工具，应建立系统对中间件产品进行存储以及描述、查询、定位该产品，对于产品的升级也应用CUS等版本控制工具来管理。此外，发布产品也应遵循业界的标准。

当前的Web Service其实也是一种广义的中间件，公司应尽早投入到该领域，现有产品有微软的.NET，IBM的WebSphere等。

这些方案的益处是能够提高产品的可复用性，应用Web Service技术则提供了更远大的网络应用发展空间。

论虚拟现实技术的应用与发展

【摘要】

本文介绍了虚拟现实技术的主要技术和应用，并结合作者本人参加的国家重点实验室的自然科学基金项目，重点讨论了基于图像的建模与绘制在虚拟现实中的重要地位与应用，详细分析了近15年来国际上基于图像的建模与绘制技术的发展状况与当今世界上最先进的研究热点，并详细阐述了作者在国家重点实验室精心基于图像绘制技术科研工作中所参与的系统开发。文章最后，说明了作者所参加的科研项目的开发成果，并简单介绍了作者在试验项目中所发表的论文，对未来几年中虚拟现实技术的发展趋势作出了较为科学的预测并提出了作者本人独到的见解。

【正文】

虚拟现实技术是当前科学研究的热点，同时也是国家大力支持发展的核心技术之一。所谓虚拟现实，就是通过计算机等虚拟手段来构造一个非常接近真实世界的虚拟环境，它主要体现在视觉、触觉、嗅觉等方面的逼真模拟来达到高度的沉浸性，也就是如我们通常所说的“身临其境”。它涉及的领域非常广泛，包括仿真技术、多媒体技术、传感器技术以及计算机图形学等多个科学领域。同时，随着科技的发展特别是网络、计算机硬件等领域技术的完善，虚拟现实也得到了越来越多的应用。包括军事、医疗、航空航天、旅游业、建筑业、娱乐业等众多行业中，虚拟现实技术都具有广阔的应用前景。

我作为浙江大学CAD/CG国家重点实验室的一名科研人员，有幸参加了多个关于虚拟现实方面的科研项目，其中国家自然科学基金项目《虚拟现实中的基于图形的建模与绘制技术》中，我投入了大量的精力，进行了广泛的学习与探索，参读了国外很多相关领域的文献，特别是近15年来SIGGRAPH大会中的精采论文，并与实验室的其他人员进行了大量的讨论，和国内外领域中的专家学者进行了多方面的交流，取得了很多实质性的进展。同时，我结合该课题在理论的探索中进行了大量的编程模拟实验，开发了一些虚拟现实系统。

我们知道，虚拟现实的一个最主要的方面就是视觉效果上的真实性，例如虚拟漫游系统。可以说，视觉上的问题解决了才能进行考虑触觉、嗅觉等方面，否则就是本末倒置，因而目前世界上大部分的科研工作都是基于视觉上的真实性进行研究的。同时，随着网络技术的发展，网络的带宽不断增加，其于Web方式的虚拟现实系统越来越成为主流，而且在美国很多先进的企业中已经得到了广泛的应用。而网络上虚拟现实的一个关键性问题就是速度，因为网络带宽相对于本地运行来说速度还是很小的，那么如何解决虚拟现实中国形的绘制速度，以达到网络显示的实时性就是一个非常重要的课题。我们也了解到，传统的建模是基于几何的，这导致当场景较为复杂时计算量过大，特别是要处理物体表面材质时，几何贴图一来真实性不高，二来速度过慢，几乎没有达到网络实时性的可能。对于大部分P C机客户，更是不能在网上欣赏到逼真的虚拟现实场景。因此，世界上一些科研人员就提出了使用基于图像的建模与绘制的方法。因为图像是真实世界的拷贝，逼真度高，而计算量又很小，可以满足普通P C机的用户以及当今网络带宽的需要，很好地达到虚拟现实场景的实时绘制。

基于图像的绘制技术(image-based rendering, IBR)最早由Apple Company的Eric Chen提出，他早在80年代初就在SIGGRAPH上发表论文介绍了他的图像拼接理论，次年又基于该理论开发出了Apple著名的Quick Time VR系统，并在当年的SIGGRAPH上发表论文详细地介绍了该系统，在当时的计算机领域引起了极大的轰动。该系统的主要手段是用多点拍摄的8张图片，进行无缝拼接算法整合成一张360°的环视图像。我在实验室进行相关的科研时，参与了一个类似的系统开发，当然，时隔多年，我们又在原来的Eric Chen的理论基础上，结合最新的科研成果，使用更好的算法开发我们的系统。关于该系统的开发方法，本文后面将有详述。在Eric Chen之后，又有很多世界顶级的科研人员不断在SIGGRAPH上发表论文，以向世人告知他们的最新研究成果，现在IBR已经成为每年SIGGRAPH中一个重要议题，每年都有4~6篇论文发表。例如基于特殊点之间进行转换的Morphing技术，基于全光场理论的光场建模绘制技术，基于图像拼图的拼接技术，基于人机交互的辅助技术等，都在一段时间内取得了巨大的成功。特别是拼图技术和全光场理论带来的各种基于光场的技术，在最近的3年内主导了国际上IBR领域甚至虚拟现实领域的主要发展方向。

我在实验室从事相关领域的科研工作之时，主要是进行了大量的理论学习与研究，当然也开发了一些虚拟现实系统。例如本文前面提到的类似于Quick Time VR的系统。在此系统的开发中，基础的理论已经比较成熟，而且已经有产品化的实物供我们参考。我们的主要工作就是结合理论，使用逆向工程的方法，导出软件系统的设计思路，并融合后来几年中国际上对Quick Time VR系统提出的改进论，开发出一套自主知识产权的虚拟现实系统。

我们首先以理论为基础，结合逆向工程工具导出原始的设计模型，加以改进，采用合理的分层结构重新设计了系统。在讨论完系统高层模型后，我们的人员需要分成3个组来进行开发。一组开发算法，一组开发系统界面，一组开发底层控制部分。我被分在系统界面开发组。虽然不是核心算法组，我们在对待界面交互设计时仍然进行了大量的分析与策划。我们抛弃了传统的MVC模型来工作，而是采用了国际上最先进的TaskMaster模型来作为我们的界面部分的驱动模型。TaskMaster模型是MVC模型的改进，它使用有限自动机来控制状态驱动，将原来的MVC模型作了更加彻底的分离，使得软件系统的各个模块之间独立性更强，增加了内聚，减少了耦合。它将系统界面对用户的响应分成多个Task，每一个Task使用一个有限自动机来驱动，有限自动机接收消息，执行Implementation，并完成状态转换。同时，TaskMaster模型不但结构清晰，还具有代码自动生成工具，使用该工具，可以快速自动地实现自动机模型。总之，使用TaskMaster模型，我们解决了传统MVC模型的不足，取代了MFC消息响应机制的不完善性，大大加速了系统的开发。后来的开发成果也证明了我们使用较少的人力物力开发出了优质、美观、高效、高可靠性及高移植性的系统(TaskMaster模型很容易移植到Linux系统下，并且Linux系统下也有TaskMaster自动代码生成工具。)

另外，在开发过程中，我们的测试文档也做得相当成功。坚持使用了单元测试、集成测试，确认测试每一个步骤都到位，特别是单元测试和集成测试，我们小组做了大量的工作。先是个人测试，然后进行组内交换测试，最后全组讨论，再进行集成测试。文档也和测试一样，是每一天都跟着来的。

在项目管理上，我采用了Microsoft Source Soft来进行控制。因为小组有6人，大家分别在不同的机器上开发，进度不同，加上我们小组内模块相关性高，传统分开开发最后集成的方法肯定不行，而使用了Source Soft之后，管理开发就轻松许多，并且大大减少了最后进行模块集成的工作量。

总之，在本次科研活动中，我们不但顺利、高效地完成了模拟系统的开发任务，而且大量实践了软件工程方法，引进了国际最优先的TaskMaster模型等，得到了很好的效果。更重要的是，我们出色地完成了实验室的国家自然科学基金项目，为实验室争得荣誉，为国家填补了科技空白。我个人而言，不但在组织本小组进行系统开发的过程中，实践了项目管理、进度管理、质量管理、人员分配、合作能力、文档管理以及测试管理等软件工程方法，并引进了国际先进的开发模式，取得了良好的实践效果；同时，本人还有一篇关于基于图像的绘制技术的论文被第八届联合国际计算机大会录取。

当然，我在科研工作中以及相关系统开发中也遇到一定的困难，比如模型选择、人员分配等都走了一些弯路。而且TaskMaster模型虽然清晰高效，但代码量比传统MVC要大得多，文件数也多，给管理造成了一定的困难。

再回到虚拟现实领域，我觉得目前在视觉方面已经较为成熟，例如IBR领域，未来的扩展需要依靠硬件辅助计算、网络分布式、人工智能技术的结合。而整个虚拟现实领域将更多地开始从事触觉和嗅觉等方面的研究，而且为了达到更真实的效果和更高的实时性，我预计将会有更多的整合人机界面学、人工智能、分布式计算、数据库等技术，共同来创造一个完美的虚拟现实世界。

论软件项目的进度管理

摘要

本文讨论了《电力行业工作票、操作票系统》的项目管理，在本项目中我作为项目负责人，承担了项目管理工作。

在本项目管理中，我主要采用了面向对象技术同传统技术相结合的原则，在估算项目的工作量这方面尤为突出，面向对象技术对传统技术有所改进，传统技术能弥补面向对象技术的不足。

本文从合理的估算项目的工作量及技术难度；识别关键任务；随时了解项目进度，必要时调整进度表等方面讨论了《电力行业工作票、操作票系统》项目管理的基本活动与方法，有效地控制开发进度，确保项目如期按质量完成。本系统在电力系统已经运行，状况良好，受到一致好评。

正文

2003 年 2 月，我参加了《电力行业工作票、操作票系统》的开发，担任项目管理工作。电力系统有关部门在对电力设施进行检测、维修、试验等一系列活动时应按照我国电力行业相关标准进行工作，《电力行业工作票、操作票系统》就是按照国家有关标准及电力行业操作规程设计的仿真系统。工作人员在施工前按照工作流程在此仿真系统上进行操作，严格遵守电力设施的逻辑闭锁关系，顺序执行。有效地防止不规范操作，确保电力设施及现场工作人员的安全，提高安全意识。

本系统由系统图编辑平台和工作票、操作票签发系统两大部分组成，其中系统图编辑平台主要是编辑变电站、用电系统及变电站控制系统图，每一个电力设施对应一个对象，在系统图上都有相对应的部分，系统图真实地反映电力设施的布局及相互关系，生动形象又合乎技术标准，同时为第二部分提供操作对象。工作票、操作票签发系统主要是在系统图的基础上进行点击操作，每次点击对应一个对象即一个电力设施，根据电力设施的逻辑闭锁关系自动生成相应的工作票或操作票或提示操作不规范。

在本系统的开发过程中，我通过合理的估算项目工作量及技术难度；识别关键任务；随时了解项目进度，必要时调整进度表等方面对项目进行管理，确保本系统如期按质量完成。

1、合理的估算项目工作量及技术难度

我们在项目工作量及技术难度的估算上采用面向对象技术同传统技术相结合的原则。本系统采用了面向对象的分析、设计等一系列面向对象技术，在本系统工作量的估算上根据功能点进行估算。将每个功能模块逐步分解，直至基本模块为止。我们将系统分为系统图编辑与工作票、操作票签发两个大的功能分别进行估算。系统图编辑部分主要是一个图形编辑系统。一种电力设施对应一个类，电力设施的技术参数及其操作对应相应类的属性和方法，电力设施图是由线段、圆、曲线、折线、多边形等基本图形组成，这些基本图形分别对应一个类，这些类又继承一个最基本的类。系统图编辑部分的工作量也就是这些类的实现，工作票、操作票签发部分用到了编辑平台的系统图，因此由大量的功能可以复用，这部分的功能划分同系统图编辑部分一样也是采用类作为基本结构，这样就比较准确的进行工作量的估算。

同时我们开发的这个系统是基于 c/s 结构的，由于 c/s 结构的系统我们公司有不少成功的案例，因此有不少的案例供我们参考。对于本系统的第二部分我们就是借鉴以前我们做过的基于 c/s 结构的系统，基于 c/s 结构的系统的框架基本上是一致的，数据库的设计、前台操作如对数据库进行添加、删除、修改、查询等一系列活动大体相同。正是如此，有大量的东西可供我们复用，如权限控制模块我们就是复用以前的案例，仅作少量修改。在工作量的估算上也有很好的借鉴作用。这对工作量的估算也是一个重要的参考，为工作进度安排提供了依据。

在技术上，我们重点考虑本系统与其他 c/s 结构的系统的不同之处，相同或相似之处我们认为没有技术难点。系统编辑平台主要是绘图，我们知道 MFC 的绘图功能确实强大，但是过于繁琐，功能封装不是十分完美，我们采用了 Form++ 这个 MFC 扩展类库，这个扩展类库对图形操作封装得很好，大大降低了系统图编辑部分的难度，在界面设计上我们采用了 BCG 这个扩展类库，使得 VC 应用程序界面设计得如同 Delphi 等工具一样完美。同时减少了工作量，在工作安排上，技术难度相对大一点的部分我们安排经验丰富的程序员，同时也同其他工作组的成员商讨技术细节问题，同他们进行技术探讨。这样不至于因为某一技术细节而影响整个工程进度。

根据上述分析我们制定一个详细的进度表并定义相应的里程碑。

2、识别关键任务

系统图编辑部分是整个系统的基础，因为工作票、操作票签发部分是建立在该部分的基础之上，系统图编辑部分直接影响到整个项目。因此该部分是整个系统的关键部分，在这部分中每种电力设施所对应的类及其父类的定义是关键，因为所定义的类必须完整、准确地反映该电力设施的技术参数和操作。

工作票、操作票签发部分，是用户明确提出的要求实现的功能，直接面对用户，这部分的成功与否直接影响到该系统的质量，因此也是不容忽视的。

如果上述两部分任务的进度受到影响，则整个项目的完成将受到威胁。因此是本项目的关键任务。在进度控制时我们将其作为重点对象进行控制。

3、随时了解项目进度，必要时调整进度表

在确定项目开发计划时，我们制定了详细的进度表。我们在确定每一项任务时都确定该任务的工作量、开始时间、持续时间、结束时间。同时让每个小组成员知道自己所承担任务的时间表，小组成员根据自己的任务制定自己的详细工作计划。

工作日志是了解每个小组成员工作情况的很好的方式，我们要求每个小组成员对自己的工作都要做工作日志，对自己每天的工作做详细记录。每周对自己的工作进展做出结论，向项目组汇报。在做结论时，不得使用“差不多”、“大概”、“完成了 90%”…等模糊字眼。而是采用某任务“已经全部完成”、或者“90%的工作全部完成”或者“再过 1 天全部完成”…等方式。每个小组成员对自己做出的结论负责，这样可以做到随时了解项目进度，为调整项目计划提供客观基础。

同时我们在项目进度计划中根据项目设计定义了相关的里程碑，在每个里程碑我们都采取小组会议形式对本阶段的工作进行确认、总结，对本阶段的进展情况做出结论，并决定是否调整下一阶段的进度计划。

在系统图编辑部分我们认为各电力设施所对应的类（包括其父类）定义完成为一个里程碑，每个类是否具备了相对应的电力设施的技术参数及操作是该里程碑的标准，这些类（包括其父类）的实现完成又为一个里程碑，……整个系统图编辑部分完成也是一个里程碑。每个里程碑的标准在系统设计时已经定义好。

结束语

《电力行业工作票、操作票系统》目前已经开发完毕，运行状况良好，受到一致好评。在本系统开发的整个过程中采用了面向对象技术同传统技术相结合的原则，因为小组成员的各有特长，面向对象技术不是每个小组成员都熟练掌握，加之面向对象技术在我们公司还不是很成熟，必须有一个过渡，不能一下子转型，因此采用这种策略符合我们公司的现实情况。

由于项目进度管理得当，项目按期完成，我们小组赢得公司的好评，其他小组也研究我们的管理方式。当然项目管理方式多种多样，根据项目不同、人员不同管理模式应做调整而不是一成不变。适合本项目的管理模式才是最好的模式，先进的管理方法在不同的项目组中取得的效果是不同的，这有待于我们去研究，探索，实践，总结。

论软件项目的进度管理

摘要

2001 年 10 月，我参与 x x 市农村信用联社综合业务信息系统的开发工作，该系统面向交易、以客户为中心、采用大会计模式，并以会计核算为核心，实现本外币合一，将柜员管理与柜面业务有机地结合起来，以业务种类码来划分整个业务系统。该项目分为二期：第一期为期 14 个月，已于 2002 年底完成；第二期为期一年，尚在开发中。

在该系统的开发中，我主要担任可行性分析、需求分析、概要设计和第一期项目的项目管理工作。我采用 PERT 计划评审技术，标识关键任务的同时，允许一些任务并行进行；着重考虑人员在整个项目开发过程中的安排；重视文档规范、命名规范，为减少员工之间的通讯障碍，还启用了 Notes 系统；跟踪和控制项目计划的执行等措施，保证了项目按期完成。但是，由于对人员流动估计不足，复用力度不够，给项目的进度管理带来了一定的困难。在以后的项目中，我们把限制人员流动率将作为一项合同约束。

正文

2001 年 10 月，我参加了 xx 市农村信用联社综合业务信息系统（以下简称 xx 系统）的分析和开发，该系统由我联社与乙集团公司联合开发，系统采用 C/S 体系结构，主机采用两台 IBM RISC/6000 S 系列机，操作系统为 IBM 的 AIX 4.3.3，数据库选用 IBM 的 DB2 UDB 6.1，磁盘阵列选用 IBM 的 7133 SSA 串行存储体系结构，前台为 SCO UNIX 5.0.5，中间件采用 IBM Txseries CICS，编程语言采用 SQL 和 C。

xx 系统面向交易、以客户为中心、采用大会计模式，以县级联社为单位设立一个帐务系统，改变原来的储蓄、对公等总帐分开的做法，并以会计核算为核心，把整个帐务系统融合在一起，实现本外币合一，将柜员管理与柜面业务有机地结合起来，以业务种类码来划分整个业务系统，打破传统的业务品种和业务部门的界限。

xx 系统分两期实现，第一期为期 14 个月，已于 2002 年年底完成并投入运行，主要完成农村信用社柜面业务，包括储蓄（对私）、对公、信贷、系统内电子联行、通存通兑、与人行的天地对接、中间业务等；目前系统内共设机构 87 个，柜组 415 个，业务操作员三千余名。第二期为期一年，尚在开发中，主要为卡业务（包括与人行的一卡通、自办农信卡业务等）及客户自助业务（包括 ATM、电话银行、网上银行等）。

在 xx 项目中，作为联合开发的甲方主要代表，我担任项目的可行性分析、需求分析、概要设计和一期项目的项目管理，跨地区通存通兑接口模块的详细设计与实现等工作。

软件开发项目进度管理是软件开发项目管理的一个重要内容，有效的进度管理是保证软件开发项目如期完成的重要环节。在 xx 一期项目软件的开发过程中，为保证软件按时完成，我采用 PERT 计划评审技术及一系列的方法和策略。

1、采用 PERT 计划评审技术标识关键任务

采用 PERT 计划评审技术标识关键任务。xx 项目计划中规定了一期项目的交付期限为 2002 年年底。整个一期项目长达 14 个月。在一期项目的开发过程中，采用的是“改进型瀑布模型”，我们从可行性分析结果出发，使用快速原型方法来补充和完善需求说明，还对信贷部分的需求进一步细化。从设计阶段起的各阶段基本采用了传统的开发方法，各阶段的结束标志比较明显。所以在软件的开发过程中，我采用了 PERT 计划评审技术对开发过程中的各关键任务加以标识，允许关键任务以外的其他任务在机动期内伸缩。而关键任务的伸缩不得超过一周。当遇到关键任务延期时，我召集大家寻找原因，并由主要责任人签字。把这种责任作为业绩考核的一部分与收入挂钩。

在标识关键任务的同时，根据 PERT 图，允许某些任务的并行。在概要设计阶段完成并通过评审后，允许各子系统在详细设计阶段及实现阶段任务上的并行进行。我把系统划分成储蓄对公、信贷、联行三个子系统，在概要设计阶段的任务一完成，就将开发人员分成三个小组，分别进行上述三个子系统的详细设计与实现。实现了在这两个阶段上任务的并行，也确保了项目的如期完成。

2、着重考虑人员在整个项目开发过程中的安排

着重考虑人员在整个项目开发过程中的安排。我从各县联社抽调业务骨干，作为业务分析人员，提出并确认需求。考虑到本项目完成后的维护需要，在乙方开发人员大部分撤离后，维护任务将主要由我社一方承担，所以从概要设计阶段开始，我方派出了三名高级程序员，分别参与储蓄对公、信贷、联行子系统的设计，继而参与详细设计、编码实现，也为后来的维护作准备。

3、减少开发人员之间的通讯障碍，提高生产率

减少开发人员之间的通讯障碍，提高生产率为了确保项目的如期完成，我们事先规定了文档编写规范、命名规范，重视文档的编写、保管等工作。重视文档与设计的一致性，先修改文档，再修改程序，不至于因为文档与设计的不一致而影响工期，对跨越里程碑的文档修改设置严格评审。为了减少开发人员之间的通讯障碍，还启用了 Notes 系统，开发人员可以通过内部 Mail 进行交流，及时沟通，减少误解。

4、跟踪和控制项目计划的执行

跟踪和控制项目计划的执行。为确保本项目的如期完成，在项目开始后，我们定期召开各组长会议，让组长们报告各自小组的进展情况及遇到的问题。通过交流开发过程中遇到的问题，共同探讨解决方法。我对照计划，跟踪实际执行情况，如果项目进展顺利，在预算范围内，我会适当放松一些控制；但只要一有问题，我会尽可能帮助排解。例如，在编码及单元测试时，因为时间比较紧，尤其信贷组，按揭贷款和跨社多头贷款控制的编码过程中遇到些阻挠，测试也不够充分，加之个别人员流动，我只好调整计划，适当调整人员，还让信贷组开发人员加班加点。有时我也找个别开发人员交谈，以了解他们对开发进展情况的评价，以及他们个人遇到的一些问题。尽我所能帮他们解决问题。

在本项目的开发过程中，所有由于以上采取了以上的技术和方法，在很大程度上保证了项目的如期交付（一期项目按计划已于 2002 年年底交付并投入运行）。其中采用 PERT 计划评审技术，标识各阶段的关键任务，允许一些任务的并行执行是比较成功的，划分的储蓄对公、信贷、联行三个组在详细设计与编码阶段的并行执行，缩短了整个项目的开发周期。重视文档的编写，效果也比较好，内部 Notes 的启用减少了通讯障碍，提高了生产率。安排本系统员工的积极参与也是成功的一面，参与过需求分析和确认测试的我方人员在系统上点过程中对培训辅导帮助很大，而参与概要设计起各阶段任务的三名高级程序员更为系统投入运行后的维护提供了方便。一方面，这些人员参与了从概要设计阶段起的各个阶段，熟悉了整个的实现过程；另一方面，这些员工熟悉业务，在后来的改正性、完善性维护方面起了很大的作用。

但是对人员的流动把握不足，尤其对乙方人员的流动更显得无可奈何。很大程度上带走了经验，也影响了进度。同时，也没有充分考虑到本系统内参与设计人员的流动。项目初期，本想增加候补人员的，但考虑到那样做还得培训，同时也增加通讯、交流的时间，担心反而会影响进度，所以未安排充分的候补人员。致使人员离去时只好再作调整，显得有些被动。所以对于人员的流动，现已作为一项约束，限制人员流动率不得高于某个百分点而在二期的开发合同中明确规定，相信会带来好处。

另外，在项目设计过程中未充分考虑代码的复用，复用程度比较低，三个组间复用的只是本项目公用的如记帐部分等，还有就是各自组内的代码积蓄了。开发效率比较低，也正因此，不但延长了编码时间，而且还必须加强测试。如果能加大复用力度，在保证项目的进度和系统的质量方面会更完善些。

论软件过程的改进

摘要

2001 年 8 月，我参加了公司一个对原有产品进行升级的项目，担任项目经理一职。这个项目是在原有的产品上进行升级，系统架构由二层改为三层，并在功能上对原有系统进行扩充。同时考虑到以前我们做开发时，软件过程不规范，原有系统在设计、开发、维护、实施上存在一些弊端，开发出来的软件总存在这样那样的问题。我们在这个项目进行过程中，对软件过程进行了改进。以前我们做开发时，软件过程不规范，开发出来的软件总存在这样那样的问题，在这个项目里，我们努力做好项目规划、加强版本控制，采取了一些质量控制手段，如同行评审、专家评审、加强测试等，使软件的质量得到保证，同时也降低了开发成本。现在看来，这个项目在一定程度上是取得了成功，但还是存在一些问题，如采用工具方面，过程改进制度化方面，过程标准化与工程进度方面的矛盾等都有不如意的地方，后来公司都对这些问题有所认识，也在不断地进行改进，公司的软件过程已日渐成熟，并于去年顺利地通过了 CMM3 认证。

正文

软件过程的改进是一项需要长期不断地进行下去的工作。以我们公司来说，作为一家软件公司，过去对软件的开发过程并不重视，接到项目后首先考虑的是如何用尽量短的时间来完成，而忽略了软件的质量，结果也往往是欲速则不达，甚至因为产品的不成熟使工期一拖再拖。

在 2001 年 8 月，我们要对公司原有的《城镇职工基本医疗保险》产品进行升级。这个产品是为了配合国家的公费医疗制改革，由公费医疗改为基本医疗保险而开发的。产品由各地劳动部门下属的医疗保险管理中心组织建设，中心机房设在医保中心，通过城域网连接到各定点医疗机构（医院和药店），参保职工可持医保卡在定点医疗机构直接消费，由医疗机构为病人垫付费用，事后再与医保中心结算。原产品为两层 C/S 结构，各定点医疗机构的前台为脱机消费，定时传送数据。这次产品升级首先将系统架构由原来的两层 C/S 升级为三层 C/S，中间层采用比较流行的 Weblogic 或 SilverStream 做应用服务器，用 J2EE 实现业务逻辑，并采用标准的程序接口与各前台程序连接，以方便与第三方软件与我们的系统集成。网络传输也由原来的定时传送改为实时连接，软件功能上也需要突破由于原产品设计时没考虑周全而带来的限制，如原系统为实现脱机消费，采用了 IC 卡带金方式，但系统设计时没有考虑到 IC 卡本身可能存在质量问题，而经常发生医保个人帐户透支问题，在新系统中我们就采用了个人帐户在医保中心统一管理，IC 卡只是作为一个身份识别的工具来解决这个问题。因此在项目启动时，对这个产品的定位就是一个全新的产品，而不是对原有产品进行修改。在这个项目里，我担任了项目经理一职。

1、做好项目规划

在项目的规划阶段，我们意识到公司原有的软件过程存在很大的弊端，首先，原来的软件过程中，设计与开发职责不分，甚至存在分析、设计、开发、测试全由一个人承担的做法，这样做不但是对人力资源的浪费，同时软件质量也得不到保证。开发和测试由一人承担，不利于测试出软件中存在的错误，整个过程由一个人来做，做出来的软件究竟对不对，没有一个说法，只有到最后程序拿给用户去用时问题才能暴露出来。再者在这样的过程中，开发人员往往会忽略文档的重要性，这对后期的维护也会带来一些问题。针对这一点，我们首先将项目组分为设计、开发、测试三个组，设计和开发组由系统总设计师负责，测试组有一个专门的组长。设计组负责软件的分析与设计，形成设计文档，设计文档首先要做同行评审，评审内容一般是文档的规范性以及对开发人员的指导性方面，同行评审后由系统总设计师来做专家评审，评审的内容是设计是否符合业务需求。开发组负责根据设计人员的设计文档编写出代码，代码编写出来要通过同行评审，评审内容是代码的编写是否符合编码规范、是否具有可读性和可维护性。测试组负责根据需求和设计文档编写测试用例，并对开发出来的代码进行测试。通过这样的改进，我们充分调动了各员工的积极性，也明确了各自的责任，使得整个过程处于受控状态。

2、加强版本控制

在原来的软件过程中，我们对软件的版本控制不严密，没有采用必要的工具，而是完全由版本控制员手工进行操作，且版本控制员还要兼一部分开发任务。在这种情况下，版本控制经常出问题，有时同一代码被不同的人员同时修改，有时将本应发给甲用户的程序发给了乙用户，又或者开发人员自以为自己手上的代码是最新的，而出现已改过的 BUG 又重复出现的现象。这样做的另一个问题是版本的历史很难追踪，由什么人在什么时候做了什么样的修改完全没法掌握。在这个项目里，我们意识到这一点，首先，设立了专门的版本控制人员，同时使用了 ClearCase 版本控制软件，所有对文档和代码的修改必须先从版本控制服务器上 Check Out，改完后再 Check In。这样做就杜绝了版本的覆盖问题，而且版本历史也是一目了然，任何修改都会形成日志，这也为问题责任的追究提供了依据。

3、加强测试工作

在这个项目里，我们特别加强了测试人员的作用。在这之前，公司也设立过测试部，但由于存在部门之间的沟通问题，测试部很难参与到项目中来，即使参与进来也发挥不了应有的作用，测试部曾一度被撤消。这一次参与测试的是新成立的测试部，而测试人员加入到项目组，业务上测试组是受项目经理领导，人事上仍受测试部领导并考核。这样做，首先消除了测试与开发之间的沟通隔阂，而测试人员也少了其他项目的打扰，可以专心只为一个项目做测试。而以前出现的因部门间隔不让测试人员参与直接由开发人员自己测试的情况也就不存在了。

由于以前的软件过程存成那么多的问题，使我们的产品不是一个成熟的产品，不成熟的产品后期施工的成本是很高的，因为存在太多的问题，维护人员要做大量的维护，而前期开发并没有留下什么文档，也给后期的维护带来很多困难，维护人员每修改一段代码首先需要读懂原来的程序，往往读不懂时就直接在原来的程序上加上一段通过设置条件来跳过原来的代码，这样使得程序越来越难读懂，问题也越改越多。这样的产品拿到一个点去施工时往往需要二个月甚至更长的时间。在这次的升级中，由于采用了较好的软件过程，产品的成熟度得到了很大的提高，而设计文档也是我们这一次重点控制的对象。这样的产品为后期的施工提供了很好的条件，现在，产品在一个点的实施时间可以缩短到四十天以内，大大地减少了施工成本。

而好的设计文档也为产品的本地化修改提供了好的条件，维护人员读懂设计文档比读懂程序要容易得多，在这样的基础上做修改出现的问题也越来越少。

尽管在这个项目里我们做了这么多的改进，但也存在不少的问题，首先我们使用的 ClearCase 版本控制软件存在问题，这个软件要求所有开发人员将自己的机器加入到由服务器控制的域里，否则，就只能取到版本快照而不能进行版本更新。由于这样做，域管理员具有比本机超级用户更高的权力来控制每台机器，使得开发人员不愿意这样做，于是出现了多人用服务器超级用户远程控制服务器来取版本的现象，使得版本的责任追究出现问题。而我们使用的 ClearCase 版本不支持 Windows XP，也使这个版本控制软件的使用出现了问题。

另外，我们的软件过程制度化方面也没做好，在项目的早期，各项工作流程都被很好的执行，各种文档也非常完整。由于我们这一次的升级只是针对的整个产品的一个部分进行的，在这之后我们又对这个产品进行了一次更大的升级，使得我们的产品能覆盖更大的范围。但后面的这次升级由于规模比这一次大，人员也大量的增加了。而新加入进来的人员并没有很好地进行规范培训，好的软件过程标准也没有形成有效的制度，再加上项目工期非常紧，包括同行评审、专家评审这样的流程都开始有些流于形式甚至被忽略。开发组编码时也没有完全按制定的规范进行。因此，产品质量上就出来了一些反复。我们这个产品是个可分可合的产品。因些在后来的产品实施上出现了这样一种情况：如果一个点只实施前一次升级的那部分，施工难度很小，能在短期内完工，本地化开发工作也很好完成。而要全面实施整个产品的话，工期就会被拖得很长，本地化开发工作也存在很大的问题。

针对出现的这种情况，我们公司意识到了软件过程改进的重要性，针对版本控制软件问题，我们改用了功能虽然没有 ClearCase 强，但更适合于我们的 VSS。而在制度化方面，更是下了大力气，从印度请来了专家为我们的改进做参谋，现在公司的情况已有很大改观，各项制度已不再流于形式，而公司更是在并在去年年底顺利地通过了 CMM3 的认证。软件过程改进的路还很长，但有一点是不变的，只有通过软件过程的改进，我们的产品才能不断地走向成熟。也只有产品成熟了，我们才能在竞争中永远立于不败之地。

应用CMM改进银行软件过程

摘要

我在某银行软件开发科室工作，长期从事商业银行中间业务应用的开发。在以往的中间业务产品开发过程中，存在许多需求不明确；开发人员有章不循，依靠个人智慧，开发处于低水平的重复状态；项目的进度、质量难以保证等问题。

我们从2001年底开始着手解决中间业务产品的整合和综合发展问题，经过项目分析和论证，我们提出了要开发一套具有良好扩展性、通用的中间业务平台系统。为了避免在新平台的开发过程中继续出现上述问题，作为该项目的主要负责人，我在中间业务开发团队引入CMM，采取了一些有效的策略和方法，主要包括：进行CMM培训和咨询工作；重新制定中间业务开发规范；做好需求管理；加强过程的跟踪和监督等。我们成功实施了银行中间业务软件过程的改进，开发了高质量的中间业务平台。CMM是全面持续的过程改进，笔者从定量过程管理和人员组织管理方面，提出了今后的努力方向。

正文

中间业务是指银行利用自己的网点、设备、网络、信息等优势，以中间人的身份接受客户委托，提供各类金融服务并收取相应手续费的业务。作为银行业务新的利润增长点及重要的吸收存款手段，中间业务正日益受到国内各商业银行的重视。近年来，我所在银行根据业务发展的需要，通过陆续推出代发工资、代收电话费、代收水电费、代收学费、行政事业代收费等一系列代收代付的中间业务，在提高自身服务水平的同时，增强了在同业中的竞争能力。

一、中间业务开发过程中存在的问题

随着中间业务品种的增多和市场对中间业务要求的提高，中间业务系统越来越庞大与复杂，尽管我们中间业务开发团队不乏技术过硬的开发人员，还是经常出现项目不能按时完成；新的中间业务项目上线时，软件质量不理想，其中包括一些明显的错误，如客户不能交费、重复扣或少扣客户的费用，客户交费后打印不了发票等；而且文档不齐全，造成维护比较困难。作为中间业务开发团队的主要负责人，我召集了几名技术骨干总结了以往的经验教训，找出了存在的一些主要问题：

1、随意性决策较多。往往软件产品从立项阶段开始就成了“实验田”——软件产品做与不做，什么时候交付等多是凭个人的主观意愿，没有参考以往经验，也没有充分考虑资源的有效投入，导致开发的产品维护成本较高，“打补丁”现象较多，用户使用不方便。

2、有章不循。尽管我们科技部门比较注重规章制度的建设，也针对软件开发过程制定了大量的程序性文件，但由于没有严格的后续跟踪与监督机制，使软件开发的质量控制大打折扣。

3、依赖个人智慧。我们开发团队中不乏一些具有相当才干的项目经理和骨干人员，他们为研发银行软件产品做出了较大贡献。然而，由于他们的经验没有被很好地总结、归纳，并上升为整个开发团队的财富，致使有的项目负责人一走，开发质量就会下降，这说明软件产品的研发依赖于个人而不是整个开发团队。

4、“可视性”低。软件开发过程不透明，上级不了解下面在做什么，无法实时监控项目进展；下面的开发人员“报喜不报忧”。由此导致产品缺陷被慢慢积累起来，等看到结果为时已晚。

二、应用CMM改进中间业务平台开发过程

由于传统的中间业务系统产品开发平台多样化、缺少统一规范、产品分散、功能不足，大大影响了开发效率和增加了维护难度。为解决传统中间业务开发模式存在的诸多弊端，我们从2001年底开始着手解决中间业务产品的整合和综合发展问题，经过项目分析和论证，我们提出了要开发一套具有良好扩展性、通用的中间业务平台系统。为了避免在新平台的开发过程中继续出现上述问题，作为该项目的主要负责人，我考虑在中间业务开发团队引入CMM，试图把个人的脑力劳动规范为有纪律的智力产品。

1、培训与开发过程规范的制定

在项目启动前，我们请了一家知名的培训机构，培训了CMM的基础理论，并结合一些软件开发公司实施CMM的案例讲解，促进了团队成员对CMM的认识和对软件过程实行规范管理的理解和支持。

然后，我们分析开发部已有的规章制度存在的问题和缺陷，重新制定了中间业务开发规范。

首先定义出软件开发中的主要过程，参考CMM并针对中间业务开发的实际需要，确定了每个过程的具体工作内容。

其次确定了软件开发的具体方法，我们确立了面向对象的开发方法，并对每个开发过程定义了相应的具体文档，如系统计划阶段必须提交《系统开发计划》，需求分析阶段必须提交《软件需求规格说明书》。并定义了文档的格式和必须提交的内容。

最后，对每个过程划分相应的人员职责，即按照工作要求，进行人员分工，并确定具体的工作内容。

2、努力做好需求管理

在银行的软件研发过程中，需求管理是一个“老大难”问题。技术部门抱怨业务部门提出的需求不清晰，甚至提不出需求；业务部门抱怨技术部门对需求理解有偏差，开发的产品与初衷相去甚远。从CMM的角度来说，业务部门提出的需求只是顾客需求，而在顾客需求中既有技术层面，也有非技术层面的，即便是技术层面的需求，也并非面面俱到都要开发，比如一些技术上不可行或者资源要求不能满足的需求就必须剔除。只有适合软件产品研发的需求才会被最终制作成规格说明。

在中间业务平台需求的制定过程中，我采取了以下措施：

（1）组成业务部门与技术部门共同参与的需求组，需求提出后经过上级领导的评审和业务部门确认才实施开发。

（2）制定技术部门必须和业务部门商定需求变更流程和跟踪监督的机制，需求变更如何提出、由谁进行评审、由谁决定做与不做、对项目进度及对资源的影响如何等都应事先考虑。

3、加强过程的跟踪和监控

软件开发过程中，我们按照软件开发过程规范严格实施。在项目启动的初期，开发人员要花很大一部分时间写文档资料，工作压力比以前大多了，工作流程的改变，也导致一段时间内效率降低。大家逐渐习惯后，感觉文档是研发人员劳动成果最好的记录，工作比以前清晰，规范的文档减少了对某个人的依赖，使软件研发过程的上下环节紧密衔接。在整个过程中，我们得到所有的文档，并根据文档的内容对每个过程进行了检查。在进度控制方面，我首先制定了系统开发计划，要求开发人员填写工作量周报，并据此绘制项目进度图，随时了解项目进展，并适当调配人手，整个项目比计划略早完成。

在质量保证方面，银行软件产品质量对业务的开展尤为重要，它涉及银行的资金安全和自身信誉，因此必须确保软件开发产品质量。现在我国各银行并没有成立单独的质量保证机构，对质量的控制主要掌握在项目研发人员手中，为此CMM建议成立专门的SQA组，检查软件开发过程标准、规程的合理性。鉴于人手问题，我们没有成立专门的SQA小组，采用测试小组兼做SQA工作，对项目组的监督“对事不对人”，并定期公布监督结果。

三、经验和不足之处

经过近一年工作，我们完成了中间业务平台的开发，新的中间业务平台运行良好，受到了业务部门和技术部门高层的赞许。在该项目的开发中，由于引进了CMM，加强了软件过程管理，很好地解决了以前存在的四个问题。中间业务产品的开发目标更明确，提高了产品开发过程的可视性和可控性，从而在提高产品开发能力的同时提高了产品的质量；我的开发管理水平也得到了很大的提升。高层领导尤其对我们的开发流程和开发规范表示认可，并决定在整个开发部门推广。我将这次软件过程改进的成功实践归功于两个原因：一是基于我们长期中间业务开发的基础上总结出来的经验和教训，二是CMM原则与我们团队和项目本身的特点的有效结合。

然而，由于我们初次应用CMM，还有许多需要改进的地方：

1、我们仅运用了CMM的基本原则，整个软件过程的管理基本是定性的。过程改进的趋势是定量管理，必须采用合适的工具，如采用项目管理工具MS PROJECT控制项目进度，使用RATIONAL ROSE对需求分析和系统设计进行辅助设计，可预测和监控整个过程性能和产品质量，及时纠正偏差。

2、鉴于人手的问题，我们团队成员往往身兼设计、开发、测试、SQA数职，组织结构不合理、人员分工不清显然制约过程改进的实施，我们最近打算成立一个独立于开发部的质量控制部，分清设计、测试和质量保证的责任、界限和权限，使其在更高的层次上控制软件质量。

软件过程的改进是一个很大的课题和实践方向，在今后的工作中，我们将争取向CMM更高的级别努力。

论软件开发平台的选择与应用

摘要

我校教务管理系统是在根据我校原来的教务管理系统不再适应现在发展的要求而立项开发的。目的是为教务工作有关部门提供优质、高效的业务管理和事务处理，建立完备、可靠和开放性的教务管理系统。

我有幸参加了新的教务管理系统的开发，担任项目管理、系统分析与设计等工作。本文结合工作的实际经历，简要讨论了软件开发平台的选择与应用。在软件开发平台的选择与应用过程中，我们本着平台的开放性、分布性和平台无关性的原则，根据我校的具体情况，通过对目前两种主流平台：J2EE 和 .NET 的比较分析，和体系结构、应用平台的无缝集成、开发成本及易开发性的思考与研究，选择了 .NET 作为开发平台。使用 Microsoft 全新的集成开发环境 Visual Studio.NET，采用 ASP.NET、Web Service、ADO.NET 和 XML 等技术进行系统开发。

正文

随着我校规模的不断扩大，计算机科学技术的进步，我校原来的教务管理系统已不适应现在发展的要求。以前单机版的 VFP 教务管理软件，被分散的安装于全校的 14 个系部和教务处，各系部之间、系部与教务处之间信息不能共享，而且对教学计划、教学考核等功能不完善或根本不支持。教务处是学校主管教育教学工作的职能部门，也是学校领导在教学业务方面的参谋机构。教务工作直接影响学校教育教学改革和教育教学质量。因此，学校决定由教务处立项重新规划建设教务管理系统。整个系统包括教学计划子系统，教学资源子系统，网上选课子系统，智能排课子系统，教学考核子系统，学生学籍子系统，学生成绩子系统，教学实践子系统，教材管理子系统等。

我有幸参加了新的教务管理系统的开发，担任项目管理、系统分析与设计等工作。由于我校分南北两个校区，教务处和 14 个系部分布较散，另外随着 Internet 的迅速发展，部分信息需要通过网络向全校师生及外部用户发布，例如网上选课信息、学生基本信息及成绩等。基于传统的 C/S 模式体系可维护性和发布性差等原因难以满足新系统的要求，有效的采用基于 B/S 体系的 Web 应用能很好地解决这方面的问题。

基于互联网的应用要求软件平台具有开放性、分布性和平台无关性。从而相继出现了 RPC、COM 和 CORBA 等技术，但这些技术在实际应用中存在着许多不足和局限。它们的特定协议难以通过防火墙，因而不适应于 Web 上的应用开发。为了进一步开发基于 Web 的应用，出现了 Sun 公司的 J2EE 和 Microsoft 公司的 .NET 两种主流的软件开发平台。

在 J2EE 和 .NET 两者之间进行选择时，我们曾举棋不定。随着面向对象技术的兴起，Java 语言应用的迅速发展，以 Java 为程序设计语言的 J2EE 具有平台无关性。同时 J2EE 已成为 Web 应用开发的标准平台。以及它的相关技术 EJB、JSP、Java Servlet 等的迅速发展，J2EE 平台已成为 Java 技术企业级应用的理想平台。但我校原有的大部分操作系统、数据库和 Web 服务器都是采用 Microsoft 的系列产品，并且在 Microsoft 系列产品的使用和开发方面积累了较丰富的经验。.NET 支持多种程序设计语言如：C++、VB.NET、ASP.NET、C# 等，实现了语言互用性。而 J2EE 只能使用 Java，这是 J2EE 所不及的。并且 Visual Studio.NET 使得 .NET 的开发较 J2EE 的易用性好。

在最后具体的软件开发平台与应用的技术方案选择时，我们采用了 .NET 开发平台。其原因主要基于对开发平台选择原则如下的认真思考和研究。

一、体系结构方面的考虑

随着 Internet 的迅速发展，传统的 C/S 体系结构已显示出了它在异构的、分布式的网络环境中的不足。可维护性和可安装性差、并且不利系统扩展。从而新的体系结构 B/S 模式迅速发展了起来。B/S 模式有利于系统的扩展性、维护性。

在校园网发展逐步完善的今天，考虑到教务管理系统安装、维护的方便和部分信息的向外发布，以及传统的 C/S 模式技术的成熟性。我们采用了 C/S 和 B/S 相结合的模式。.NET 开发平台正是为进一步开发基于 Web 的应用而出现的，是一组用于建立 Web 服务器应用程序和 Windows 桌面应用程序的软件组件。.NET 支持多种编程语言，使各种语言可以自由地在整个 .NET 平台内互用，很好的发挥各种语言的特性。例如：我们对 C/S 结构程序使用执行效率高的 VC++.NET 和快速开发的 VB.NET，B/S 结构程序使用 .NET 专门为 Web 应用定制的 ASP.NET 和 C#。再加上功能强大的 Visual Studio.NET 集成开发环境，.NET 为 C/S 和 B/S 相结合的模式提供了很好的解决方案。

基于此原因，我们把系统的教学计划子系统，智能排课子系统，教学考核子系统，教学实践子系统，教材管理子系统等设计为 C/S 结构，网上选课子系统，学生学籍子系统，学生成绩子系统，教学资源子系统等设计为 B/S 结构。

二、应用平台的无缝集成性

由于我校使用的操作系统都是 Microsoft 的 Windows 系列，同时 .NET 是与 Windows 操作系统紧密捆绑在一起，使得 .NET 在 Windows 上的应用开发更为容易。并且以前的数据库是 VFP 的。考虑到数据的平稳过渡以及我们对数据库的熟悉程度，再加上 .NET 提供的 ADO.NET 数据访问组件是对 ADO 的改进，分为三组：ODBC、OleDb、SqlClient。其中 SqlConnection 是专门为 SQL Server 设计的，性能明显优于其它的数据访问组件。我们在新系统的后台数据库服务器的选择上采用了 Microsoft SQL Server2000。

选择 Microsoft 的操作系统 Windows，数据库服务器 SQL Server2000 和开发平台 .NET 应用平台，充分利用无缝集成平台的优势，使 .NET 应用开发更容易，运行更可靠、更安全。

这是 J2EE 所不及的。

三、节约开发成本

由于我校在以前的信息化建设过程中培养了一批经验较丰富的 C++、VB、ASP 等开发人员，熟悉 Windows 上的开发，同时在数据库管理系统 Microsoft SQL Server 上的设计与开发方面有一定的经验。而在 Java 开发方面的经验相对不足。如果我们选择 J2EE 则意味着开发人员资源的浪费，并且要重新培养 Java 开发人员，并且新培养的开发人员由于缺乏经验，很难保证开发效率和质量。选择 .NET，则我们的开发人员便能轻松的转变到 C++、VB.NET、ASP.NET 及 C# 的开发当中来。再加上以前在 Windows 上的开发经验，最终有利于我们的开发速度加快，质量提高，从而很好的节约了开发成本。

四、易开发性

就 .NET 开发平台的容易使用性来看，在如下几点得到了很好的体现。

1. .NET 的重要部件 ASP.NET 使 Web 应用程序的开发和部署更为容易。.NET 相对较新，它拥有 Java 所缺乏的改进，例如，ASP.NET 使开发者可以用比 Java 开发者在 J2EE 平台上更少的代码来实现 Web Services。在教务管理系统中，我们对学生成绩查询、学生网上选课和教学资源调配等交互较多的逻辑模块，都设计成 Web service 结构的中间组件。轻松的节省了花在用户界面编程上的开发时间、同时 Web service 完全可以在应用程序集成等场合下被重用。
2. ADO.NET 数据访问组件较以前的 ADO 更方便的访问各种类型的关系数据库和非关系数据库，获取本地和远程数据源，并对 XML 提供了强大的支持。这对以后的后台数据库的扩展也提供了很好的支持。
3. Visual Studio.NET 为 .NET 提供了一个统一的集成开发环境及工具，大大提高了开发者的效率；集成了多种语言支持；简化了服务器端的开发；提供了高效地创建和使用网络服务的方法等等。

通过对上述四个主要方面的思考，所以我们选择了 .NET 平台作为开发与应用。

目前软件开发平台主要向 Web 的应用方向发展。由于 Web 的应用是基于分布和异构的网络环境的，所以要求开发平台应具有开放性、分布性和平台无关性。现在流行的软件开发平台主要有 Microsoft 的 .NET 和 Sun 的 J2EE。在实际应用当中，我们具体选择那种方案应根据具体情况而定，很多情况可能会综合使用两种开发平台。在适应技术发展的过程中，快速的跟上新技术是必须的。

论软件开发平台的选择与应用

摘要

我所在的单位是国内主要的商业银行之一，作为单位的主要技术骨干，2003 年 1 月，我主持了远期结售汇系统的开发，该系统是我行综合业务系统 XX2000 的一个子系统，在选择 IBM 的 VISUALAGE FOR JAVA 作为软件开发平台的过程中，我主要遵循了保证与原有系统的互用性；保证开发人员快速而准确开发出应用程序；延长应用的生命力的原则。在选定了软件开发平台后，我还采用了三层 C/S 的软件体系结构；利用 CICS 的 SWITCH 组技术；使用标准、开放的规范等措施，保证了远期结售汇系统的按计划完成并且顺利投产，确保了该系统的开放性、互用性、可用性、先进性和稳定性，特别是我所选用的 VISUALAGE FOR JAVA 开发平台和为了保证系统开放性和先进性所采用的措施得到了领导和同事的一致认同和称赞。目前，但是，我也看到了近期软件开发平台的主流是 SUN 公司的 SUN ONE 和 MICROSOFT 的 .NET 两大阵形，我认为，而软件开发平台将朝着集成性、开放性、平台无关性的方向发展。

正文

我所在的单位是国内主要的商业银行之一。众所周知，银行的业务存在一个“二八定理”：即银行的百分之八十的利润是由百分之二十的客户所创造。为了更好的服务大客户，适应我国对外贸易的蓬勃发展态势，促进我国对外贸易的发展，2003 年 1 月，我行开展了远期结售汇业务。

所谓的远期结售汇就是企业在取得中国外汇管理局的批准后，根据对外贸易的合同等凭证与银行制定合约，银行根据制定合约当天的外汇汇率，通过远期汇率公式，计算出交割当天的外汇汇率，并在那天以该汇率进行成交的外汇买卖业务。远期结售汇系统是我行综合业务系统 XX2000 的一个子系统，它主要包括了联机部分、批量部分、清算部分和通兑部分，具有协议管理、合约管理、报价管理、外汇敞口管理、帐务管理、数据拆分管理、报表管理、业务缩微和事后监督等功能。

我作为单位的主要技术骨干之一，主持并参与了远期结售汇系统的项目计划，需求分析，设计，编码和测试阶段的工作。在选择使用 VISUALAGE FOR JAVA 作为软件开发平台的过程中我主要采用了以下几个原则：

1、保证与原有系统的互用操作性和兼容性。由于远期结售汇是我行综合业务系统的一个子系统，必须保证与原有业务系统的互用性，以确保我行业务的正常开展。我行以前的联机系统一直都是使用 IBM 的 VISUALGEN 2.0 来开发的，编译后生成 COBOL 程序，由于 COBOL 特别适合用来表示商业逻辑，所以在我行的联机系统取得了很大的应用。我主要选择使用 VISUALAGE FOR JAVA 开发平台的 VISUALGEN 部分的功能，程序编译后也是生成 COBOL 程序，确保了与原有系统的互操作性性和兼容性。这样，以前开发的应用程序可以不经任何修改，方便快捷地移植到 VISUALAGE FOR JAVA 开发平台上，这样既保护了以前的投资，又消除了因为移植而可能导致的错误，为我行的安全生产打下了良好的基础。

2、保证开发人员快速而准确开发应用程序。首先，VISUALGEN 2.0 是我行一直使用的开发平台，外汇部的开发人员能够用它快速而准确地开发出联机系统的应用程序，而 VISUALAGE FOR JAVA 的 VISUALGEN 部分和 VISUALGEN 2.0 只有很小的区别，只要进行很短时间的简单培训，开发人员便可以快速而准确地得开发应用程序；其次，VISUALGEN 2.0 只支持 IBM 的 OS/2 操作系统，而大部分的文档，办公软件和 NOTES 等都是安装在 WINDOWS 操作系统之上的，这样开发程序的时候需要在两个操作系统之间不断地切换，不但浪费了开发人员的时间，降低开发效率，还给开发工作带来了很多的不便，容易造成错误，比如开发人员在开发的过程中需要对照详细设计文档的时候，需要切换到 WINDOWS 操作系统，当数据比较多时，切换的次数也多，这样便浪费了开发时间，容易造成错误。而 VISUALAGE FOR JAVA 则支持 WINDOWS 操作系统，大大地方便了开发人员，提高了开发效率，减少了开发错误，得到了领导和开发人员的一致好评。最后，VISUALAGE FOR JAVA 集成了很多软件管理的技术，增强了对项目的管理，比如它支持团队开发，良好的团队开发管理为外汇部的开发工作带来了很大的方便，使得各个部分有条不紊地进行。又比如它良好的版本管理功能，使得开发工作可以更加高效而准确地进行，极大地支持了开发工作的顺利进行。

在选定了 VISUALAGE FOR JAVA 作为远期结售汇系统的开发平台之后，我还采用了以下几个措施来保证系统的开放性和先进性：

1、采用三层的 C/S 软件体系结构。根据银行联机系统对开放性、可用性和伸缩性的要求特别高的特点，我采用了三层 C/S 软件体系结构，三层 C/S 软件体系结构是目前比较先进和流行的软件体系结构，按照它所建立的分布式应用系统有很好的开放性、可用性、伸缩性和互用操作性，很好地适应了银行应用系统的特点，其中：表示层我采用了我行自行研发的 CITE 字符终端，由于字符终端在运行速度上比图形终端更快，很好地适应了我行对联机系统高速的要求；中间层我采用了 IBM 的 CICS TRANSATION SERVER (以下简称 CTS)，用来连接 CITE 和数据层，这样做的好处是屏蔽了通信，安全，线程管理等最重要而又最耗时的功能开发，使得开发人员可以专注于应用逻辑的开发，缩短了开发周期，减少了开发、维护和集成费用，增加了开发的成功率，极大地增强了系统的开放性，互用性和安全性；数据层采用了 IBM 的 DB2 UDB7.1 数据库，它的数据库连接池、方便快捷的索引等众多的技术能很好地适应了强调吞吐量和速度的银行系统。

2、利用 CICS 的 SWITCH 组技术。我行今年实现数据集中，全国设了南北两个数据中心，分别位于 XX 和 XXYY 两地，而两个数据中心的数据传输显得尤其重要。比如我行的外汇报价管理采用总行集中统一报价，各个地区再根据各自的业务量大小设置相应的优惠点数来进行远期结售汇的外汇买卖。由于总行在北方中心，这样就涉及到两个数据中心通信的问题，为了实现两个数据中心通信的负载均衡和容错，我采用了 CICS 的 SWITCH 组技术，所谓的 CICS 的 SWITCH 组技术就是在南北中心之间架设一组的 CICS TRANSATION SERVERCTS，每个 CICS TRANSATION SERVERCTS 都是相同的，起到了连接南北中心的作用，消除了单点故障。该 SWITCH 组不但具有开放性：每个 CICS TRANSATION SERVERCTS 与具体的应用逻辑无关，意味着所有的应用逻辑也可以共有该 SWITCH 组，而且具有先进性：为了简化系统的设计，缩短通信时间，我采用了简单的负载均衡算法，比如这次分配给第 N 个 CICS TRANSATION SERVERCTS，下次则分配给第 N+1 个；为了实现更好的容错，我采用了当某个 CICS TRANSATION SERVERCTS 失效的时候，把它正在处理的事务转到另一个 CICS TRANSATION SERVERCTS，大大增强了系统的可用性。

3、使用标准、开放的规范。每个公司为了提高自己的竞争力，都有自己专用的标准和规范，IBM 公司也不例外，为了保证系统的开放性，我采取了一些与规范有关的措施：比如在开发的过程中我要求开发人员尽量使用标准的 SQL 语句，以增加系统开放性；使用标准的通信协议 TCP/IP，以确保与其他子系统的通信。

由于采用了 VISUALAGE FOR JAVA 开发平台和以上保证系统措施开放性和先进性的措施，远期结售汇系统最后按照计划完成并顺利投产，不但取得了良好的社会效益和经济效益，而且我选用的开发平台和采取的措施得到了领导和同事的认同和称赞。

在总结经验和不断学习新技术的同时，我也看到了近期软件开发平台的发展趋势和主要特点：近期的软件开发平台朝着集成性、开放性、分布性和平台无关性发展，主流软件开发平台是 SUN 公司的 SUN ONE 和 MICROSOFT 公司的 .NET 开发平台，并有两强并立之势。结合我行综合业务系统的特点在充分分析比较了两个平台的优点缺点之后我发现：SUN 的 SUN ONE 开放平台最大的优点是操作系统的可移植性，对于我行 UNIX、OS/390 和 WINDOWS 操作系统共存的局面非常有利，缺点是只能用 JAVA 来开发程序，而 JAVA 对于商业逻辑的开发并不如 COBOL 那样简单易用，而且我联机系统大多数都是用 COBOL 语言来开发或编译成 COBOL 程序，不能保护原有的投资；MICROSOFT 的 .NET 的优点是开发、运行程序的成本更低，由于 .NET 的语言是完全中性的，所以可以使用 COBOL 程序来编程，保护了原有的投资，伸缩能力更强，互操作性性更强，这些特性对于我行系统都特别重要；缺点是只能在 WINDOWS 操作系统下运行，对于我行多操作系统并存的局面又很不利。为了适应软件开发平台的演变，我将在以后工作中不断地加强学习，了解、掌握开发平台的知识，努力适应发展趋势，为我国的软件事业贡献自己的微薄力量。

论软件开发平台的选择与应用

摘要

2003 年上半年, 我作为某电子通信公司的软件开发人员, 参与了一款基于嵌入式 linux——uclinux 的手持式无线网络通信产品的软件开发工作。该产品提供为无线传输优化的 TCP/IP 协议服务, 具有自主组网、多跳路由的功能, 集成语音通话、数据传输、实时图像传输等应用, 可广泛应用于野外通讯、数据采集等领域。在项目之始, 我们根据综合考虑性能、功能、技术可行性以及成本效益的原则, 确定了以 uclinux 为软件开发平台, 并在其应用过程中, 通过积极参与开源运动, 解决了不少技术难题, 保持了系统的先进和稳定性。在此过程中, 我作为项目的软件设计和实现人员之一, 参与了选择软件平台的决策过程以及之后的具体应用。

正文

我们说沙滩上建不起摩天大厦, 这是强调地基对于建筑物的重要性, 在软件开发过程中, 开发平台的作用就和地基类似, 它对软件的性能、功能、质量、稳定性和可扩展性都有至关重要的影响。因此, 选择和应用合适的软件开发平台, 是每个软件开发组织要面临的挑战之一。作为一个嵌入式软件设计开发人员, 2003 年, 我参与了一款手持式无线网络通信产品项目(下面简称无线项目)的研发工作, 同样也经历了一次这样的挑战。

由于嵌入式软件和硬件平台的密切关系, 有必要先简单介绍一下无线项目的硬件情况, 由于有大量音频、视频编码运算, 主处理器是美国 Analog Device 公司(下简称 AD 公司)的一款 DSP+RISC 架构的数字信号处理器 BF-21533, 外扩 32M SDRAM + 16M Flash, 外围器件有彩色 LCD, 键盘, 音频、视频以及射频芯片等。相应的软件方面涉及到从底层硬件驱动、开发平台移植、核心路由模块到顶层应用等多层次的开发工作, 不仅系统功能复杂, 而且其中有一些实时的应用需求, 在性能上也有较高的要求, 整个项目开发难度较大, 因此我们希望找到一个合适的软件开发平台, 充分利用硬件资源, 并且减少不必要的开发工作, 使精力能够集中到公司核心的业务上来。

我和另外一个软件设计人员经过和硬件设计人员的协商, 初步拟定了三个候选方案, 一个是采用 AD 公司提供的 Visual DSP++ IDE, 另一个是采用风河公司的 Vxworks, 再有一个是采用迅速发展的嵌入式 linux——uclinux。对这三个方案, 我们分析其优缺点, 综合考虑功能、性能、可行性以及成本效益, 最终选定了第三种方案。

首先从功能方面看, 这是我们重点考虑的方面。Visual DSP++ IDE 中集成的 VDK 功能最弱, 只提供了简单的任务调度功能, 不支持界面显示和 Flash 文件系统, 这意味着我们要充分发挥硬件的显示和存储功能, 就必须自己在 VDK 上开发相应的功能模块, 这必将带来大量的额外工作, 使得我们不能集中精力在自己的优势上, 从这一点上考虑, 我们就首先将 VDK 置于淘汰之列。再看 Vxworks, Vxworks 功能强大, 稳定性高, 风河公司在其基础上已经开发出各种板级支持包和大量的可复用的功能模块, 开发者采用类似于构件编程的方式, 可以快速开发出高质量的产品。我们初步估计了一下, 如果采用基于 Vxworks 的平台上进行开发, 不仅 LCD 显示和 flash 文件系统的开发工作可以省略, 甚至如果采用通用音频、射频编码芯片, 它们的驱动也不用开发, 整个软件项目的进度可以加快不少, 因此采用 Vxworks 方案无疑是诱人的。最后看 uclinux, 它也有我们所需要的界面显示和文件系统等功能, 但是缺乏来自平台供应商的技术支持, 应用资源没 Vxworks 齐全, 开发的工作量肯定要比 Vxworks 大一些, 不过 uclinux 也具有 Vxworks 所没有的优点, 它是开放源代码的, 可以对其源码进行局部修改以适应所开发软件的需要, 这一特点对于我们的核心多跳路由算法特别有用。从这些方面考虑, Vxworks 和 uclinux 都各有千秋, 不易取舍。

其次从性能方面来看。嵌入式系统对性能的评价主要从代码空间、数据空间、执行速度、中断响应速度等方面进行考虑。在这方面, 作为硬件厂商提供的嵌入式内核, VDK 优势明显, 其内核体积小, 所需内核代码、数据空间只有十几 K 字节, 同时厂商对其实行的硬件优化, 使得响应速度快, 实时性高; 而 uclinux 和 Vxworks 所需的代码、数据空间是以兆字节来计算的, 执行和响应速度也不及 VDK; 如果 uclinux 和 Vxworks 进行比较, Vxworks 实时性要占优一些, 因其是专门的嵌入式抢占式内核, 而 uclinux 只是通用 linux 内核稍作修改而来的, 且为非抢占式。但由于无线项目的硬件平台性能较为强大, 代码数据空间稍大和相应时间稍长这些都不是特别棘手的问题,

再次从可行性方面考虑。由于 VDK 方案所需开发工作量太大, 在项目规定的时间内难以完成, 因此又一次首先被淘汰; Vxworks 同样也存在可行性问题, 主要是我们采用的处理器 BF-21533 由于推出时间不长, Vxworks 对其不支持, 这意味着如果我们要使用采用 Vxworks, 就必须重新选择处理器, 这增加了硬件设计的工作量和修改的风险; 相反, AD 公司为发挥 BF-21533 的强大性能, 对 uclinux 进行了移植工作, 并且推出了相应的 toolchain 编译工具, 这意味着无需重新选择处理器也能进行 uclinux 平台上的开发, 这一点促使我们开始倾向于 uclinux 平台。

最后从成本效益方面考虑。VDK 和 uclinux 都是免费的, 而 Vxworks 作为商业软件, 价格不菲, 除去开发工具的费用, 它的按模块收费和形成产品后按件收费的特点使得不仅开发费用较高, 而且产品成本也会较高。这再一次将我们推向了 uclinux 平台。

综合上面四个方面的考虑, 我们最终选择了 uclinux 开发平台。

在应用 uclinux 平台的过程中, 我们充分利用 linux 的硬件平台无关的特点, 将系统划分成硬件相关和硬件无关两部分, 硬件相关的比如移植工作和各种 driver 的开发则在具体硬件平台上开发, 硬件无关的部分比如音频视频编码和路由算法则采取灵活的措施, 先在 windows 平台下开发测试, 然后在装有相同 linux 版本的 PC 机上进行必要的移植修改。这样做的目的是使得各部分软件可以并行开发, 不至于受硬件方面的影响, 同时也是针对 linux 平台下开发工具和测试工具比较缺乏的弱点, 充分利用公司在 windows 平台下的积累的工具资源, 从而提高软件质量。

在保持系统的先进性和开放性方面, 我们采取从哪里来, 就到哪里去的方法, 积极参与开源运动, 充分发挥开源运动的力量。

首先我们加入了 uclinux 网站上将 uclinux 移植到 BF-21533 处理器的项目组, 不仅关注和及时更新 uclinux 在 BF-21533 上的移植的最新版本、工具和补丁, 而且在相关新闻组上通过电子邮件的方式与来自全世界的开发者交流, 讨论在开发中遇到的各种问题及解决方法, 通过这种方式, 我们成功解决了一些棘手的问题比如系统的 boot load, 同时我们也将发现的 bug 即时反映给 AD 公司和相关开源项目, 帮助完善 uclinux 在 BF-21533 上的移植。

其次, 我们将简化后的系统公开, 在网上建立开源项目, 发布硬件设计图和在其上移植的 uclinux 内核、工具、部分硬件驱动以及相关指引, 使得有兴趣的人可以组建和我们类似的平台, 项目发布后的近一年中, 收到了很多开发者的电子邮件, 或者指出硬件设计上的瑕疵和修改意见, 或者指出某些驱动程序上的 bug 和设计不合理的地方, 有的甚至将自己修改后的系统设计图和相关驱动寄来供我们参考, 通过这些热心人的帮助, 我们发现了不少测试时没有发现的 bug, 提高了软件质量, 使得我们的 uclinux 平台日趋稳定, 同时也对平台的扩展也有了一些的想法和把握, 这些都为我们将来在其基础上进行新的项目开发提供了有利的条件。

目前, 在嵌入式软件开发平台占主流的是以 vxworks、wince 为代表的商业产品, 但以 linux 为代表的开源运动也将这一潮流带到了嵌入式开发这一领域, 显示了它的影响力, 而且影响力有越来越大的趋势, 这种趋势的驱动力不仅来自 linux 内部, 而且来自于外部厂商的支持, linux 2.6 已经正式支持不少嵌入式处理器, vxworks 开发商风河公司也已经舍弃了敌视 linux 的态度, 转而和 redhat 公司合作开发嵌入式 linux 平台。相信未来在嵌入式开发领域 linux 必将大有作为。

论软件开发平台的选择和应用

摘要

本文从一个行业 MIS 系统的开发实践, 讨论了软件开发平台的选择和应用。首先, 作者从项目的实际情况确定了软件开发平台的一些原则: 技术成熟兼一定先进性、高效集成的开发工具、开方人员熟练掌握等, 随后就系统平台、软件开发平台、数据库平台的选择作了详细论述。之后, 作者就开发过程中就保持系统开放性, 对数据导入导出、与 P3 软件集成、WEB 查看权限采取了相关措施; 就保持系统先时性, 提到了多种软件技术合成及 VPN 技术两项措施。最后, 作者对近期商业应用软件开发平台的主流--微软的 .net 及 J2ee 进行了介绍, 比较了其优缺点, 对今后本部门在软件开发平台的发展方向作了一定的评估。

正文

我公司是大型电源建设项目的专业建设公司, 曾以总承包方式承建设了多个电厂, 在工程建设过程中逐步建立了一系列完整、科学的工程管理体系, 与此相应的是逐步建立电力建设项目工程管理信息系统 (简称电建 MIS), 原版本的不足之处是应用模块不多, 且多以简单文本为主。

2002 年, 浙江省某规模为 4 台 60 万千瓦机组的发电厂二期工程建设上马。该工程作为一个大型能源投资项目, 将有力地拉动地方经济的增长, 并且将为浙江和华东地区的经济和社会发展提供强大的能源支持和保障。工程由多方共同出资建设, 由我公司承担工程总承包建设任务。借此工程建设的契机, 我公司决定重新开发电建 MIS 3.0 版。主要模块拟包括办公自动化、施工总平管理、合同管理、物资管理、质量管理、安全管理、图档管理、公用信息管理、综合查询 (包括 WEB 查询)、系统维护, 基本涉及我项目部的各个职能部门。

软件采用二层 C/S 与三层 B/W/S 相结合的方式。其中, B/W/S 方式用于查询和浏览, C/S 方式用于主要数据录入和维护, 采用 C/S 和 B/S 结合的混合体系结构, 较好地满足系统功能的需求, 并符合可持续发展的原则, 使系统有较好的开放性和易扩展性。

软件采用我公司和外部软件公司合作开发的形式, 版权由我公司独家所有。因多种原因, 与我们合作开发的软件公司有 2 个。我是软件开发的负责人。

在开发平台的选择上, 我们考虑了本 MIS 项目的特点: 该项目要求在规定的时间内 (6 个月) 内完成, 并要求有较高的质量, 且要为以后的进一步开发提供基础。

基于此, 我们对选择开发平台的原则达成共识:

- 一、技术上成熟且具有一定的先进性;
- 二、有高效、集成的开发工具;
- 三、应为开发人员熟练掌握。
- 四、软件平台提供商对该软件平台的后续支持能力。

首先, 在系统平台的选择上, 有两点考虑: 1、公司现有的各级软件系统都是基于微软 Windows 系列平台的, 且公司没有在日后使用其它平台的打算; 2、微软的 Windows 平台完全能满足开发、运行该类 MIS 系统的要求。因此确定新开发的 MIS 系统也就是基于此平台的。

对二层 C/S 在开发平台的选择, 考虑采用微软的 VB6.0 或 Sybase 的 Powerbuilder8.0。前者是公司前两版 MIS 系统的开发平台, 后者则更为现有的开发人员所熟悉。两者在开发上技术都很成熟。在开发界面的亲和性上, VB 做得较好; 而对从数据库设计到编程的全过程而言, Powerbuilder 具有更好的集成性, 用其集成工具 Powerdesign 设计的逻辑数据库可以很方便地生成物理数据库。从当时情况看, 无论是微软, 还是 Sybase, 对各自软件平台的后续支持能力都较好。最后综合各种情况, 选择了 Powerbuilder, 其最重要的一点是开发人员的熟悉程度。

对三层 C/S 在开发平台的选择, 考虑采用的比较方案是微软的 IIS+ASP 组合及 Apache+PHP 组合。开发人员的经验还是在 IIS+ASP 上, 对于 Apache+PHP 少有实践, 基于此, 选用第一种方案。

在数据库平台的选择上, 有微软的 SQL Server 和甲骨文的 Oracle 可以选择。前者是前两版 MIS 的数据平台。而 Oracle 数据库是目前公认的最优秀的大型数据库, 和微软的 SQL SERVER 相比, 它具有更好的稳定性和安全性---这通常是企业用户最关心的两个特性。开发人员对两种数据库都有相应的开发经验。最终选用了 Oracle 数据库平台。

在开发中考虑了系统的开放性和先进性。在开放性方面, 有以下措施。

1. 针对原有大量已积累数据 (大多是 EXCEL 格式的) 的导入, 及系统中数据可能的导出, 设计了专门的导入导出模块。使得操作人员一方面从繁重的初始数据录入工作中解脱出来, 一方面随时可以导出系统中的数据作个别分析。并预留了与公司总部及其它协作单位进行数据交换的接口。

2. MIS 系统与 P3 软件 (大型项目管理软件) 要求做到集成。为此设计了专门的模块, 通过 ODBC 方式进行后台的数据交换, 使得两套软件做到无缝集成: 在 P3 软件中操作的结果可以在 MIS 软件中反应出来, 而在 MIS 软件中相应模块中输入的数据也能在 P3 软件中反应出来。这样做的好处是大多数工程师无须操作 P3 软件即可输入有关工程进度的数据, 并通过 P3 软件的功能自动计算出相应的数据并得到横道图; 而少数工程师仍可在 P3 环境工作, 而工作结果同样反应在 MIS 中。

3. B/S 部分的使用对象不仅仅局限于项目部的员工, 还包括公司本部人员及众多协作单位的人员。而由此相关的权限及安全问题必须考虑, 即不同身分的人员能查看的权限是不一样的。ASP 在此方面的功能相对较弱, 为此在 B/S 的服务器端专门设计了基于 COM 技术的身份认证 ActiveX 组件。

在先进性方面, 有以下措施。

1. 采用了多种软件技术的合成, 如在 MIS 系统中合成了工作流技术, 在工作流技术中调用了微软 OFFICE 的宏技术。工作流技术上考虑与系统更好地集成, 采取了基于数据库的方案, 自行开发了相关的模块。其最终效果是在 MIS 中实现了办公自动化的功能, 极大地方便了使用者。

2. 采用了先过的 VPN 技术, 使得部分外出员工可以通过 Internet, 借助 VPN 访问内部的 MIS 系统, 从而完成远程办公。

系统运行后收到了良好的效果, 用户对该软件的满易度较前两版有较大的提高。该系统随后应用于多个电厂建设项目的管理中。

近期商业应用软件开发平台的主流越来越集中在微软的 .net 及 J2ee, 其共同特性是更多地支持面向 Internet 的开发。 .net 的优势是支持多种开发语言, 开发人员容易掌握, 产品集成好, 总体成本较低, 不仅在开发的时候容易介入, 在日后维护时也更能掌握主动权, 其缺点是事实上只支持 Windows 平台。J2ee 的优点是跨平台, 可以选择多家公司的相关产品, 但开发相对复杂, 且只支持 JAVA 语言。Powerbuilder 虽然有最新的面向分布式及三层 B/s 结构的新版本推出, 但终不及前两者。对于象我们在非软件企业的 IT 部门, 在开发新的软件系统时, 如采取自行开发或与其它软件公司合作开发的方式, .net 应该是一个比较好的选择。

论软件开发平台的选择与应用

摘要

本文讨论选择新软件开发平台用于重新开发银行中间业务系统。银行中间业务系统是指银行通过与企事业单位、机关团体的合作，为客户提供金融服务的系统。X 省农行银行的原中间业务系统软件开发平台是以 UNIX 系统为操作系统，使用的数据库是 Sybase，采用二层的 C/S 结构，使用 DB—Library，T—SQL 编程。随着业务的不断发展和软件开发维护工作的剧增，该软件开发平台表现出工作效率低，开放性差，开发出的产品不易管理等突出性的问题。为了解决原软件开发平台的不足之处和基于该银行长远发展目标的要求，我们引入新的软件开发平台 OSP PrePbranch。在文中阐述了选择软件开发平台的原则：要求开放性好，可复用性高、开发出的软件易于管理、风险可控、技术能与发展主流趋势相一致并易于掌握，并总结了应用新软件开发平台开发银行中间业务系统所带来的优势。

正文

银行的中间业务系统是指银行依托自身的电子化平台、网点网络和人员等资源，通过与企事业单位、机关团体的合作,为客户提供代收代付、资金转帐结算、业务代理和咨询、增值理财等金融服务的平台。X 省农业银行的中间业务系统主要包括由五大类业务组成，即批量业务，如代发工资、国库统发工资、批量代收代付；单笔缴费业务，如柜台代收学费、行政管理费等；通信类业务，主要是指要与外单位进行实时通信的业务，如固定电话缴费系统，手机缴费系统等；增值服务业务，如代客理财、银证资金转帐和结算；其他业务，如业务咨询和代理等。

该行原中间业务系统是 C/S 结构，采用 DB-Library/C 编程，使用的数据库是 Sybase，以 UNIX 作为操作系统。该开发平台在开发中间业务系统是工作效率比较低，开发周期比较长，复用性差。针对上述开发平台的实际情况和存在的主要问题我们引入新的国内某公司的基于中间件技术的软件开发平台“OSP PrePbranch”用于重新开发该银行的中间业务系统。在整个项目期间我参与了软件开发平台的选型，系统的分析，开发和设计，测试的工作。

在选择 OSP PrePbranch 平台时，我们主要遵循了以下原则：利用选择的平台开发出的新系统能和基础业务系统有机集成的原则，选择的开发平台易于掌握的原则，选择的开发平台开放性好、复用率高的原则，选择的平台易于管理的原则。

应用新软件开发平台开发出来的系统要能够和基础业务系统有机集成是引入软件开发平台的基本条件，它也是选择软件开发平台的原则之一。OSP PrePbranch 软件开发平台是一个独立的开发平台，使用该软件开发平台开发的中间业务系统能方便通过基础系统的接口直接调用，不需要另外开发接口调用中间业务系统，而且 OSP PrePbranch 厂商进行了现场演示和提供了正式的书面文档。这样，通过简单的调用能实现基础业务系统和中间业务系统的有机集成。

选择的开发平台要遵循易掌握性原则。引入一项过于复杂或难以掌握的软件开发平台是毫无意义的。基于中间件技术的 OSP PrePbranch 软件开发平台采用的操作系统是 unix，使用 C/C++和嵌入式的 T—SQL 语言为开发工具。C/C++、SQL 它们都是该银行软件工程师比较熟悉的编程语言，因此该软件开发平台易于掌握。使得该银行选用该软件开发平台在较短的时间就能掌握它，应用它。

选择的开发平台要遵循开放性好、复用率高的原则。OSP PrePbranch 系统提供的每个原子交易都定义了规范的接口说明，调用时入参、出参的类型、个数、顺序方面都有明确的定义，使用的场合范围；如文件的上传/下传服务就有详细的定义，上传文件名，传输方式，存放的目录等方面都有详细的说明。在开发时只要根据实际情况和应用说明文档就可以实现透明调用了。OSP PrePbranch 软件开发平台在应用层提供了网关（GateWay）服务，使得中间业务系统与协作单位的通信不需要银行开发人员自己开发，而由 OSP PrePbranch 直接提供；OSP PrePbranch 还提供了通信的包格式的定义，通信连接方式的定义，通信格式加包头，通行打包，通信格式去包头，通信格式解包等原子交易。

除了上述 OSP PrePbranch 系统本身具有良好的开放性，提供大量可复用的原子交易外。在此次开发新系统时根据银行中间业务系统的特点还自制作了大量可复用的原子交易，如帐户的入帐、帐户明细的查询、帐户性质的判断、卡折之间的转帐、各人帐户和对公帐户之间的转帐等，把这些模块加载到 OSP PrePbranch 系统的构件库中，与 OSP PrePbranch 本身的原子交易一起提供服务。对自制的原子交易也提供了规范的接口说明，参数调用说明，应用场合等规范性文档。它们和 OSP PrePbranch 软件开发平台本身直接提供的原子交易对于所有的程序员是透明的，程序员使用时只要查阅规范文档就可以直接使用，而根本不需要了解它们的具体实现的细节。有了这的开发平台，在开发新系统时只要根据不同的业务品种编写必需银行核心功能模块，然后从 OSP PrePbranch 构件库中选择所需的原子交易就可以快速开发出金融交易。这样提高了软件的复用率，该变了过去重复开和重复测试的大量工作，极大地缩短了开发地周期，大大加快了研制进程。因此，无论 OSP PrePbranch 软件开发平台本身，还是后继的开发都保持了良好的开发性，较高的复用率。

选择的开发平台要遵循易于管理的原则。该银行中间业务系统改造之前客户端前台软件升级时，采用手工作业方式，工作量大，不便于管理。新系统中把各网点终端信息收集在一起，保存中间业务系统应用数据库中；若软件需升级更新时，把升级软件放在中间业务平台的指定目录下，每天网点开启中间业务系统时，前台软件自动执行版本更新升级程序，保证网点及时、自动地从后台取回最新版本软件并投入生长运行。因此，新系统极大地减少了软件升级所带来地工作量，使得中心机房可对各网点运行的软件进行统一升级、版本控制、集中管理、方便管理。

由于选用的软件开发平台 OSP PrePbranch 是基于中间件技术的，它由前台开发子系统，主控守护进程 ubridge，后台开发子系统和数据库连接与管理子系统组成的。中间件技术是当前信息技术发展的一种主流技术，因此 OSP PrePbranch 软件开发平台在相当长的一段时期内保持技术领先的优势。

OSP PrePbranch 软件开发平台具有良好的开放性，复用性高，它所采用的是先进的中间件技术。特别是 OSP PrePbranch 系统的开放性好，可复用性高方面在应用该软件开发平台开发中间业务系统时，可以方便的使用 OSP PrePbranch 提供的开放的接口，大量现成的原子交易，在具体开发时只要根据业务需要编写必须的核心代码就可以快速开发出金融交易，缩断了软件开发的周期，提高了工作的效率，提高了软件的复用性，加快了研制开发的进程。使得该银行能在较短的时间内开发出新的金融产品，提高了银行的市场竞争力。

任何一种软件开发平台的选用都具有一定的风险，特别是在金融业中引入新的软件开发平台要谨慎从事。首先，它是一个基于中间件技术的成熟的产品，在此之前已经在国内金融业有多个成功应用的先例；其次，该软件开发平台大部分是采用 C/C++语言开发的，它易于掌握，因此就不存在选用一个不易掌握或不能掌握的平台；再次，该软件开发平台实施成本低。因此我们从软件开发平台的成熟性，易掌握性，实施成本等多方面进行了论证和考察，它所带来的风险是在可以控制的范围之内的。

当然，我们在选择 OSP PrePbranch 软件开发平台主要是基于该银行的实际情况的需要。在不同的应用领域有不同的选择，如企业应用集成可以选择 EJB 规范的 J2EE 平台，它具有开放性，平台应用的无关性等特点。但是无论那种软件开发平台，它们都应该朝着开放性、平台无关性、可移植性、能够实现异构集成等方向发展和演变，同时它们应该代表着先进的技术发展方向。

论软件三层结构的设计

摘要

我所在的单位是国内主要的商业银行之一，作为单位的主要技术骨干，2003 年 1 月，我主持了远期结售汇系统的开发，该系统是我行综合业务系统 XX2000 的一个子系统，由于银行系统对安全性，可靠性，可用性和响应速度要求很高，我选择了三层 c/s 结构作为该系统的软件体系结构，在详细的设计三层结构的过程中，我采用了字符终端为表示层，CICS TRANSATION SERVER 为中间层，DB2 UDB 7.1 为数据库层，并采用了 CICS SWITCH 组，并行批量的办法来解决设计中遇到的问题，保证了远期结售汇系统按计划完成并顺利投产，我设计的软件三层结构得到了同事和领导的一致认同和称赞。但是，我也看到在三层结构设计中存在一些不足之处：比如中间层的负载均衡算法过于简单，容易造成系统负荷不均衡，并行批量设计不够严谨，容易造成资源冲突等。

正文

我所在的单位是国内主要的商业银行之一。众所周知，银行的业务存在一个“二八定理”：即银行的百分之八十的利润是由百分之二十的客户所创造。为了更好地服务大客户，适应我国对外贸易的蓬勃发展态势，促进我国对外贸易的发展，2003 年 1 月，我行开展了远期结售汇业务。

所谓的远期结售汇就是企业在取得中国外汇管理局的批准后，根据对外贸易的合同等凭证与银行制定合约，银行根据制定合约当天的外汇汇率，通过远期汇率公式，计算出交割当天的外汇汇率，并在那天以该汇率进行成交的外汇买卖业务。远期结售汇系统是我行综合业务系统 XX2000 的一个子系统，它主要包括了联机部分、批量部分、清算部分和通兑部分，具有协议管理、合约管理、报价管理、外汇敞口管理、帐务管理、数据拆分管理、报表管理、业务缩微和事后监督等功能。

我作为单位的主要技术骨干之一，主持并参与了远期结售汇系统的项目计划、需求分析、设计、编码和测试阶段的工作。由于银行系统对安全性，可靠性，可用性和响应速度要求很高，我选择了三层 c/s 结构作为该系统的软件体系结构，下面，我将分层次详细介绍三层 c/s 软件体系结构的设计过程。

1、表示层为字符终端。我行以前一直使用 IBM 的 VISUALGEN 2.0 附带的图形用户终端来开发终端程序，但在使用的过程中，分行的业务人员反映响应速度比较慢，特别是业务量比较大的时候，速度更是难以忍受。为此，我行最近自行开发了一套字符终端 CITE，它采用 VISUAL BASIC 作为开发语言，具有响应速度快，交互能力强，易学，编码快和功能强大的特点，在权衡了两者的优点和缺点之后，我决定选择字符终端 CITE 作为表示层。

2、中间层为 CICS TRANSATION SERVER (CTS)。首先，我行与 IBM 公司一直保持着良好的合作关系，而我行的大部分技术和设备都采用了 IBM 公司的产品，其中包括了大型机，由于 CICS 在 IBM 的大型机上得到了广泛的应用，并在我行取得了很大的成功，为了保证与原来系统的兼容和互用性，我采用了 IBM 的 CTS 作为中间层，连接表示层和数据库层，简化系统的设计，使开发人员可以专注于表示逻辑和业务逻辑的开发工作，缩短了开发周期，减少开发费用和维护费用，提高了开发的成功率；其次，对于中间层的业务逻辑，我采用了我行一直使用的 VISUALAGE FOR JAVA 作为开发平台，它具有简单易用的特点，特别适合开发业务逻辑，可以使开发人员快速而准确地开发出业务逻辑，确保了远期结售汇系统的顺利完成；

最后，由于采用了 CTS，确保了系统的开放性和互操作性，保证了与我行原来的联机系统和其他系统的兼容，保护了我行的原有投资。

3、数据层为 DB2 UDB7.1. 由于 DB2 在大型事务处理系统中表现出色，我行一直使用 DB2 作为事务处理的数据库，并取得了很大的成功，在 DB2 数据库的使用方面积累了自己独到的经验和大量的人才，为了延续技术的连续性和保护原有投资，我选择了 DB2 UDB7.1 作为数据层。

但是，在设计的过程中我也遇到了一些困难，我主要采取了以下的办法来解决：

1、CICS SWITCH 组。众所周知，银行系统对于安全性，可靠性，可用性和响应速度要求很高，特别是我行最近进行了数据集中，全国只设两个数据中心，分别在 XX 和 YY 两个地方，这样对以上的要求就更高了，为了保障我行的安全生产，我采用了 CTS SWITCH 组技术，所谓的 CICS SWITCH 组，就是一组相同的 CTS，每个 CTS 上都有相同的业务逻辑，共同作为中间层，消除了单点故障，确保了系统的高度可用性。为了简化系统的设计和缩短通讯时间，我采用了简单的负载均衡算法，比如这次分配给第 N 个 CTS，下次则分配给第 N+1 个 CTS，当到了最后一个，就从第一个开始；为了更好地实现容错，我采用了当第 N 个 CTS 失效的时候，把它正在处理的业务转到第 N+1 个上面继续处理，这样大大增加了系统的可用性，可以为客户提供更好的服务；此外，我还采用了数据库连接池的技术，大大缩短了数据库处理速度，提高了系统运行速度。

2、并行批量。银行系统每天都要处理大量的数据，为了确保白天的业务能顺利进行，有一部分帐务处理，比如一部分内部户帐务处理，或者代理收费业务和总帐与分户帐核对等功能就要到晚上批量地去处理，但是，这部分数据在数据集中之后就显得更加庞大，我行以前采用串行提交批量作业的办法，远远不能适应数据中心亿万级的数据处理要求，在与其他技术骨干讨论之后，并经过充分的论证和试验，我决定采用了并行批量的技术，所谓的并行批量，就是在利用 IBM 的 OPC (Tivoli Operations, Planning and Control) 技术，把批量作业按时间和业务处理先后顺序由操作员统一提交的基础上，再利用 DB2 的 PARTITION 技术，把几个地区分到一个 PATITON 里面分别处理，大大提高了银行系统的数据处理速度，确保了远期结售汇系统三层结构的先进性。在并行批量的设计过程中，我考虑到批量作业有可能因为网络错误或者资源冲突等原因而中断，这样在编写批量程序和作业的时候必须支持断点重提，以确保生产的顺利进行。

由于我软件三层结构设计得当，并采取了有效的措施去解决设计中遇到的问题，远期结售汇系统最后按照计划完成并顺利投产，不但保证了系统的开发性开放性、可用性和互用性，取得了良好的社会效益和经济效益，而且我的软件三层结构设计得到了同事和领导的一致认同与称赞，为我行以后系统的开发打下了良好的基础。

在总结经验的同时，我也看到了我在软件三层结构设计中的不足之处：

首先，负载均衡算法过于简单，容易造成系统的负荷不均衡：由于每个业务的处理时间不一样，有的可能差距很远，简单的顺序加一负载分配算法就容易造成负载不均衡，但是如果专门设置一个分配器，则增加了一次网络通讯，使得系统的速度变慢，这样对响应速度要求很高的银行系统来说也是不可行的，于是我决定采用基于统计的分配算法，即在收到请求的时候，根据预先设定的权值，按概率，直接分配给 CTS。

其次，由于批量作业顺序设计得不够严谨等各种原因，容易造成资源冲突：在远期结售汇系统运行了一段时间之后，数据中心的维护人员发现了，系统有的时候会出现资源冲突现象，在经过仔细的分析之后，我发现，由于每天各个业务的业务量大小不一样，顺序的两个作业之间访问同一个表的时候便会产生资源冲突，另外，在 OPC 作业运行的过程中，操作员提交的其他作业与这个时间的 OPC 作业产生也有可能产生资源冲突。对于第一种情况，可以在不影响业务的情况下调整作业顺序或者对于查询作业运用 DB2 的共享锁的技术，而第二种情况则要制定规范，规定在某时间断内不允许提交某些作业来解决。为了更好地开展系统分析工作，我将在以后的工作实践中不断地学习，提高自身素质和能力，为我国的软件事业贡献自己的微薄力量。

论软件三层结构的设计

摘要

随着市场的建立和发展, 卫生行业面临了很多问题, 一些制约卫生事业发展的矛盾和问题日益显现, 因此, 国家卫生部要求各医院采用信息化管理。前不久, 我所在的部门承担了一个医院管理系统的设计和开发, 医院希望以此来转变医院现有的运行机制, 提高服务质量。该系统除了目前常见的计费系统、电子病历外, 还包括门诊医生工作站、住院医生工作站、护士工作站等分系统。考虑到需要通过 Intranet 实现功能, 并有部分的 Internet 功能, 本项目平台最后采用了 Java 平台。我在项目中主要负责项目的前期规划, 即选择合适的开发方案, 并建立部分的数据流, 在系统实施过程中推动其顺利前进。此系统开发成功后投入运行, 获得医院相关工作人员的好评。

正文

前不久, 我所在的部门承担了一个医院管理系统的设计和开发, 医院希望以此来转变医院现有的运行机制, 提高服务质量。客户是一个市级医院, 医院很早就开始从事信息化管理, 但主要是针对计费这一块, 后来, 对其进行了改进, 加入对病人电子病历的管理和采集, 这样当病人二次就诊时, 可以很容易地得到病人的既往病史。但随着系统的运行, 院方希望对现有系统进行改进, 为了更好得为病人服务, 医院考虑加入一些其它的分系统, 比如门诊医生工作站、住院医生工作站和护士工作站等等。因此我所在的部门承接了该 HIS 的开发, 开发的成果是一个典型的 Java 技术在 Intranet 上的应用。

在开发前期, 首先要设计出详细的系统功能规范, 这一部分所花费的时间很少, 因为卫生部在 2002 年曾经颁发了一个有关医院管理系统功能规范的通知, 我们参考了该规范, 很快确定了各分系统以及每个分系统的基本功能。但在选择合适的系统平台上有一番讨论, 考虑到医院原有系统在某些地方运行良好, 是否有必要将原有系统淘汰重新设计, 另外新的分系统到底采用何种平台结构也是需要考虑的问题。

医院原有的计费系统和电子病历系统数据流向范围比较固定, 主要集中在交费处和挂号处, 一旦引入了新的系统, 必然要将数据流向医院的各个部门。医院的 Intranet 已经实施, 因此首先考虑采用 B/S 架构体系, 旧系统的数据模型尽可能保留。在系统的软件平台上, 我们考虑使用 Java 平台, 可以让数据在整个系统安全、有效地流动; 另外现在也有很多的 HIS 系统可供我们参考, 虽然往往是单机版的系统, 但其中的数据模型有很好的参考价值。医院的现有网络系统和操作系统多种多样, 这就要求我们选择的软件平台必须具有开放性、平台无关性。而在不同的系统上安装相应的 Java 客户端虚拟机并不困难, 最后, 在项目组的讨论和征求客户意见下, 项目组采用了此方案。

在项目中, 我们这样设计 Java 架构系统, 将系统分为三层:

(1) 表示层采用 Jsp 实现页面输出, 这也是用户直接访问层, 表示层接受来自网络浏览器的 HTTP 请求, 然后返回给客户端浏览器可以显示的 HTML 页面;

(2) 中间件层用 Java 实现对数据库的访问, 考虑到数据的分布特点, 我们使用了数据库连接池技术;

(3) 数据库层用 SQL Server 实现数据库的管理和存储过程。

JSP 以其执行的高效性和使用的方便性, 已成为近年来大家首选的因特网开发技术, JSP 是一种页面开发技术, 它以 Java 为其服务器端语言, 结合 Java Script 作为其客户端语言, 能方便地实现页面的表示。选择 Jsp 作为前台语言, 是考虑到它的平台无关性, 能够兼容其他的操作系统和数据。利用 Jsp 可以将 HTML 文件很方便地发送到客户端 Web 浏览器, 同时也可以支持一些非 HTML 格式的文档发送。我们在网站中的用户层页面设计中, 广泛地参考了现有的一些成功案例, 在节约设计和开发的费用的同时也得到了用户的认可。比起单纯的 Servlet 技术, Jsp 在页面元素上也更为丰富, 在这里, 我们为今后客户页面定制做了部分尝试, 设计了一个比较简单的模块, 用户可以通过选择控件来动态生成页面提供打印, 当然在这部分我们设计中, 可提供的数据库字段是固定有限的, 灵活性没有很好体现。

我们大量工作主要放在中间件层, 由于系统中的医生工作站分系统是主要部分, 因此数据库的负载成为一个不得不考虑的问题。综合考虑, 我们决定采用数据库连接池技术, 因为基本上每个医生都对应一个客户终端, 无论是病人资料的查询或录入, 或者医嘱的处理, 都涉及到对数据库的频繁读写, 服务器对于数据库连接的频繁打开和关闭必然导致性能下降。一方面, 我们预先考虑数据库的连接量, 在系统初始阶段建立相应的存储空间, 当数据库连接打开和关闭时都对该连接池进行处理; 另一面, 我们也使用了高速缓存技术, 对某些固定的 SQL 查询结果, 例如药品查询、药性禁忌等, 将结果采用缓存存储, 以加快对数据库的访问, 降低了服务器端的负载, 提高性能。值得一提的是, 为了尽可能得避免纠纷, 医院要求采用医生签名制度, 我们设计了电子签名, 采用加密算法保证各环节产生数据的有关人员不能抵赖。

数据库层我们选择了 SQL Server, 程序员比较熟悉此平台的开发和设计。在前期, 我们到医院做了大量的走访工作, 了解整个数据流, 同时广泛参考现有的系统, 虽然对医院行业不是很熟悉, 但数据层开发遇到的问题的并不是很大。在概要设计阶段, 我们使用了一些计算机辅助开发工具, 这也加快了详细设计的进程。比如使用了 Sybase 的 PowerDesigner 工具, 在概要设计中规划了 E-R 图和各个分系统以及分系统之间的数据流图, 然后让工具直接生成后台数据库中的基本表结构, 大大提高了开发效率。

整个项目实施完成后, 医院相关工作人员反映良好, 但其中也暴露出了一些问题, 值得改进。

首先, 用 JSP 编程时容易导致系统信息的扩散。如果有人恶意攻击服务器, 程序执行将出现异常。这时, 就会在页面上打印出相应的错误信息。这些信息会暴露出这台服务器的路径信息。这个漏洞往往会被别人利用, 程序员对此也一头雾水, 下一步我们考虑从服务器入手, 采用通用的异常说明界面, 解决该问题。

其次, 在设计上, 可以使用更多的辅助开发工具和建模工具, 比如 Rational Rose, 可以利用它的代码自动生成功能, 来大大提高开发速度; 在用户界面定制方面, 我们希望通过不断的实践, 加强其灵活性, 尽可能把可供选择的字段扩大到整个数据库。

最后, 大量使用 Java 技术, 势必对服务器的负载过大, 在今后的开发中, 除了现有的数据库连接池和缓存技术, 我们考虑使用更多的数据库脚本来替代部分 Java 代码, 来高速实现逻辑, 相较而言, 数据库脚本的执行速度较有优势, 结合结果缓存, 应该对服务器的性能有较大的提高。

上面是我在今后的系统设计和开发中需要注意和加以改进的地方, 也是我今后应该努力的方向。

论软件三层结构的设计

摘要

随着中间件与 Web 技术的发展,三层或多层分布式应用体系越来越流行。在这种体系结构中,将应用功能分成表示层、功能层和数据层三部分。

本人在去年参加了一个备件流程管理项目的开发,在此项目中担任需求分析和结构设计等工作。结合需求分析结果和该单位的实际情况,在该项目中我们采用 C/S 和 B/S 的混合模式,客户端使用的是 Delphi 和 FrontPage 进行开发,中间件我们采用的是 COM+,使用 Delphi 进行开发,后台使用 SQL Server 数据库。本文详细描述三层结构的设计过程,重点讨论中间件的设计过程和在设计实施过程中碰到的一些问题以及解决的方法,文章最后说明了采用三层结构带来的效果,以及可以改进的地方。

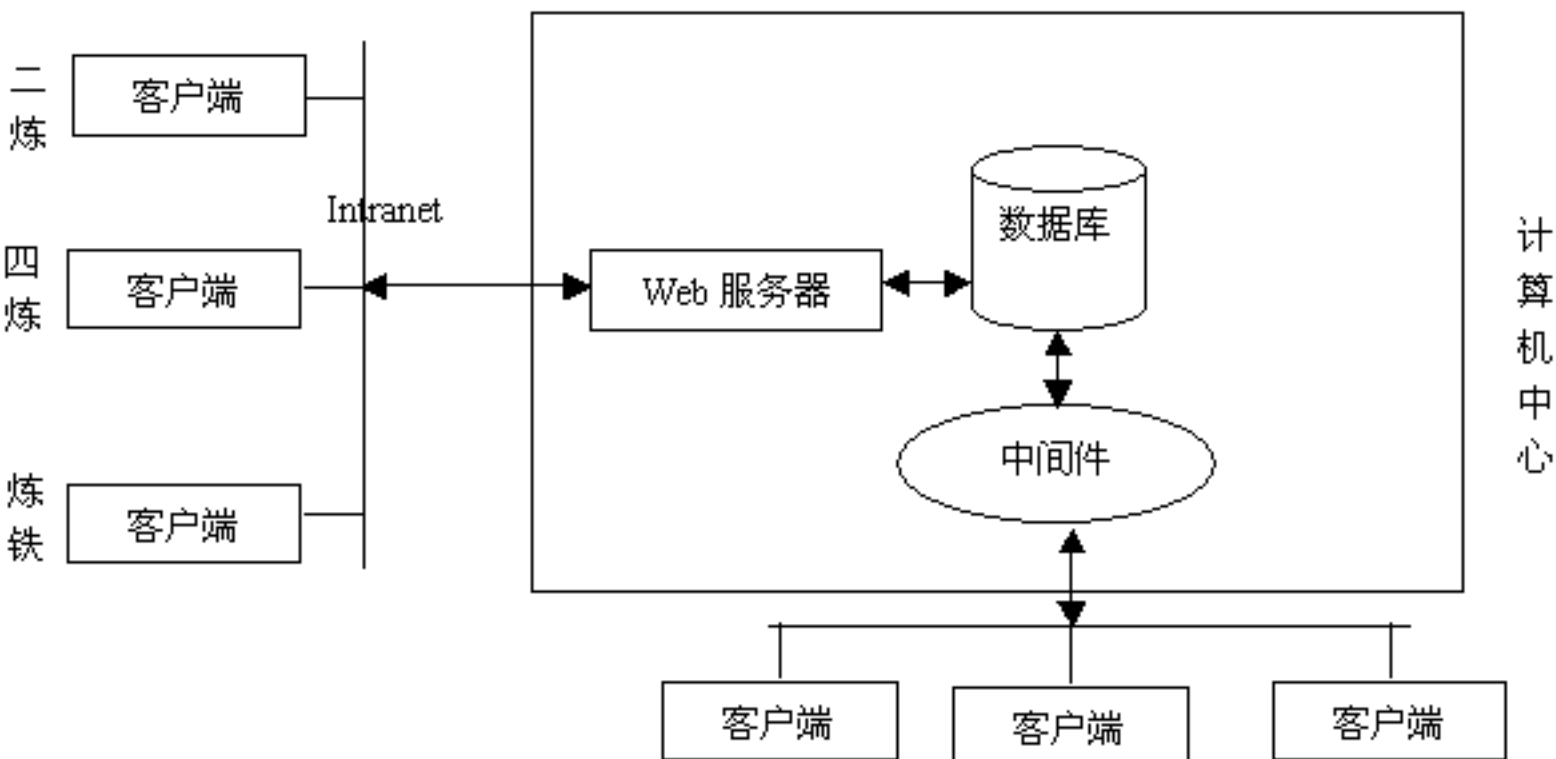
正文

2003 年 8 月,我参加了某钢铁公司备件采购项目,主要负责该项目的需求分析和结构设计等工作。该钢铁公司原先的备件采购全部是人工完成,手续复杂,办事效率低下,更重要的是无法提供及时有效的信息供企业领导参考。该公司一方面需要通过此项目来缩短采购时间,提高办事效率,另一方面该公司已经建成比较完善的局域网,需要在内部网上公开采购结果,并要做出统计分析,供全公司员工和领导查询参考。该项目涉及部门主要包括该集团公司下的仓储公司,供应公司,招标公司,劳企部等。

根据我们做出的需求分析以及各种体系结构的优缺点,我决定采用 C/S 和 B/S 混合的体系结构来开发。对于仓储公司等部门的需求,需要对数据进行更新处理,采用 C/S 结构可以更快更好的开发且数据处理速度更快,可以更好的满足要求。对于领导和员工的查询需求,我们采用 B/S 模式。采取这样的结构可以很好的满足用户需求,且容易开发和维护。由于都是在 windows 平台上使用,因此在开发工具的选择上,我们使用 Delphi 来开发仓储公司等部门的客户端和中间件,使用 FrontPage 来开发网页,连接在其内部网上提供查询服务。中间件我们采用的是 COM+,进行逻辑处理,数据层使用 SQL Server。

以下详细介绍三层结构的功能分配和物理分布,描述三层结构设计的过程,讨论在设计实施过程中碰到的一些问题以及解决的方法,文章最后说明采用三层结构带来的效果,以及可以改进的地方。

对于客户端,B/S 结构仅提供查询功能,使用 Internet Explorer,各个子公司、二级部门都可以通过内部网使用。C/S 结构提供日常操作和管理界面,承担着整个系统的数据录入及数据维护工作,使用 Delphi 开发,它是系统数据的入口,使用频繁,安装在仓储公司、供应公司等单位;中间件和数据库以及 Web 服务器都放在集团公司的计算机中心,便于维护管理。中间件负责根据客户端要求从数据库中取得数据,并在进行处理后提交到客户端显示;后台使用 SQL Server 数据库,数据集中在数据库服务器进行管理,方便数据管理和分析,保证数据安全。其物理分布如下图:



对于 C/S 和 B/S 结构,我们分别使用不同的工具来开发客户端。C/S 结构的客户端我们使用的是 Delphi 来开发,对于 B/S 的客户端我们使用的开发工具是 FrontPage,采用 VBScript 脚本语言来开发。客户端只用来显示结果,所有的逻辑处理我们全部放在中间件中。在客户端只需引用中间件提供的接口即可,这样一个方面使逻辑处理集中,便于维护,另一个方面只向客户端提供处理后的数据,可以减少网络流量,加快反应速度。

对于中间件的设计是我们工作的重点。Microsoft 的 MTS/COM+不但能够稳定地执行应用系统的企业对象,以服务客户端的请求,更能够提供在分布式环境和异质数据库之间保护数据的事务管理能力。这让分布式应用系统能够稳定、可靠地在复杂的环境中正确地执行。再加上 MTS/COM+能够有效地利用各种系统资源,增加中介软件的执行效率,因此使用 MTS/COM+作为中介软件的应用系统能够提供合理的执行效率。

经过考虑,我们决定使用 COM+作为中间件来开发。如何设计出合理的中间件关系到项目的成败。根据项目的特点,我决定根据不同的部门以及各个部门的需求来开发 COM+组件。因为各个部门有不同的数据表,中间件主要处理的是每个部门各自数据处理和内部网上的查询以及统计分析处理,所以为每个部门设计了数个 COM+组件,分别用来处理数据维护,查询操作以及统计分析工作。Delphi 提供了 MTS 的向导可以帮助程序员开发 COM+应用系统,程序员可以直接在 COM+数据模块中放入 ADOExpress 组件来存取数据库,提高了程序员的工作效率。

但在开发过程中,也出现了不少的问题。比如在仓储公司维护更新工作的设计中,使用到多个数据库表,有些程序员直接使用 MTS/COM+数据模块,然后在 MTS/COM+数据模块中加入 TADOConnection 连接数据库,再分别使用 TADOQuery/TADOTable 和 TDataSetProvider 组件连接到这些数据表。

从功能上说没有什么问题,但我发现这样设计存在一些问题。其中最大的问题就是执行效率的问题。这可以从数个不同的角度来看,首先程序员把所有的数据存取组件放在一个 MTS/COM+数据模块中,因此当客户端建立这个 MTS/COM+数据模块时需要花费许多的激活时间。第二个问题是如果客户端只需要使用其中的一个数据表的数据,那么客户端仍然需要花费多余的时间建立不相关的对象。最后,对于 MTS/COM+提供的 Pooling 机制而言,这样设计系统架构也是不好的,程序员应该尽量利用 MTS/COM+提供的数据库连接 Pooling 的功能。因此最好是把每一个 TADOConnection 的连接放在不同的 MTS/COM+数据模块中,因为这些数据表都位于一个相同的数据库中。

为了解决这些问题,我重新设计了这个架构。为充分利用 MTS/COM+的数据库连接 Pooling 功能,并且避免客户端建立不必要的对象,我使用多个独立的 MTS/COM+数据模块,其中分别放入 TADOConnection、TADOQuery 和 TDataSetProvider 以连接到不同的数据表。这样的设计虽然比直接使用单一的 MTS/COM+数据模块来得麻烦,但是无论在执行效率、系统的延展性以及资源的共享性上都比单一的 MTS/COM+数据模块好得多。

为了更好的设计开发中间件,我对使用 COM+做中间件来开发应用系统需要注意的一些问题做了总结:

- 1、对于客户端而言,应该尽早取得需要使用的 MTS/COM+对象,并且在最后使用完毕之后再释放取得 MTS/COM+对象。
- 2、避免激活不必要的事务 Context。对于 MTS/COM+对象而言,应该让事务管理越晚发生越好,并且在执行完必要的工作之后立刻调用 SetComplete/SetAbort 和 EnableCommit/DisableCommit 方法释放占据的资源。尽量把相关的 MTS/COM+对象放在同一个套件组件中,尽量减少不同套件组件之间 MTS/COM+对象的调用。
- 3、在 MTS 中使用 STA 线程模型的对象,在 COM+中使用 Neutral/Rental 线程模型的对象。
- 4、尽量利用 MTS/COM+对于数据库的 Pooling,而不要使用主从架构的方式来利用数据库 Pooling。通过使用这些方法在开发过程中我们大大改进了系统的执行效率。

数据层使用一台数据库服务器,通过企业局域,分布在仓储公司等单位的工作站利用客户端软件进行数据录入及维护工作,在各个二级分厂、单位的客户端都可以通过内部网进行查询浏览。我们采用的是 SQL Server 数据库,功能强大,使用方便,完全满足系统的要求。

最后通过项目组成员的努力,我们按期完成了任务。采用这样的体系结构,完全满足要求,并且安全可靠、容易维护、扩展方便、结构模块化、易操作。经过用户一段时间的使用,基本上没有什么问题。在后期的维护中,我们对中间件和客户端分别做了少许修改,但二者之间没有互相影响,这些充分体现出多层结构的优越性。当然在其中也存在一些问题。比如在系统的执行效率和延展性上,给人的感觉是多层应用系统的执行速度有点慢,还有组件的设计上还存在些问题。这些问题在了解 COM+的设计方法和系统需求后才能更好的解决。

XML 在网上银行中的应用

摘要

网上银行是指在 Internet 上提供银行服务,即银行的客户无须到银行柜台办理业务,可以在家庭、办公室等能够连入 Internet 的任何一处,登录到银行的网站进行交易。这是一种崭新的银行运营模式,具有方便快捷、成本低廉、不受时间地点限制等优点。

本文通过论述的项目是某银行行网上银行系统的 1.0 版本到 2.0 版本的升级和改造,论述了 XML 在 Internet 中的应用。我有幸参加了这个项目,承担在该项目中担当了部分的分析与设计的部分任务。

系统的 1.0 版本存在诸如交易超时、作业比较慢、不能满足客户个性化、技术相对落后等缺点。在 2.0 该项目版本的设计和开发过程中,我们基于 JAVA 技术,采用 J2EE 构架,使用应用了 XML 作为数据交换的标准,。在后台,基于业务数据建立了 XML 数据库,存放签约客户的历史数据,同时在 Web 服务端,我们也应用了 XML,读取 XML 数据库中的数据,同时给客户提供了“个性化”的服务。这些技术的采用,解决了 1.0 版中存在的问题。但是,因为 XML 是一种新的标准,有些地方还不是很完善,在 J2EE 架构下,如何使用 XML 是我们应该一直关注的问题,本文就该问题也有所论述。

正文

2002 年 3 月,我参与了的银行某网上银行系统的升级和改造工作。该系统采取总行、分行两层结构,总行网银中心连接各一级分行,提供信息服务、客户服务、帐务查询和实时交易等功能。网银中心与客户通过 Internet 相连,与分行业务主机通过城市综合网相连。网上银行的客户使用 Browser (浏览器)通过 Internet 连接到网银中心并且发起网上交易请求;网银中心验证后将交易请求返回;分行业务主机完成交易处理,返回处理结果给网银中心;网银中心对交易结果进行再处理后返回相应的信息给客户。

原有系统无论从业务上,还是从技术上通过运行一段时间,存在这很多问题,客户和业务人员意见也很多,诸如部分时间交易超时,作业业务比较慢,还有无法满足客户的某些个性化的要求,基于 ASP+VB 的应用也不合时宜,这些都是系统需要改造的原因。

新系统整个网上银行应用的开发,全部基于 JAVA 技术,数据的交换采用标准的 XML 协议,应用开发采用 Websphere Studio+Visual Age for Java 等工具进行,采用符合国际 J2EE 标准,系统采用了业界领先的中间件产品(BEA WebLogic)建立网上网上银行系统的交易平台,该系统的主控程序应用 JSP 和 servlet 编写,很好地发挥了多线程机制,大大提高了系统的性能;主要业务逻辑采用 EJB 技术实现,模块化结构利于新业务的开发与布部署;数据库访问符合 JDBC 标准,利用 WebLogic 的 Jdbc pool 提高了数据库访问的效率。

在这里主要谈谈 XML 在该项目中的应用。

XML 是一种具有描述数据功能的语言,它十分适合作为知识表示语言或作为组件及文件格式的表示方法。它还可以让数据在不同的来源中,根据通用的语法规则来处理。

而 Java 则是用于 Internet、适合于分布式环境、提供了一个跨平台的语言。XML 和 Java 相结合主要原因是基于 XML 的语法提供了一种灵活的、标准的、健壮的 Java 编程方案。

在该项目中这里,我们统一了 XML 的报文标准,列举了 XML 实例,规定了 XML 头部,XML 根节点,XML 二级节点及具体的报文字段。

在接口中的 XML 报文遵循如下约定:XML 头部如实例所示,不得改变;XML 根结点、二级节点如实例,不得改变;报文必须是包含‘\0’作为结束符的字符串;接口格式说明中的字段即指实例中具体的报文字段这一级。对于接口文件,规定所有发送、接收的文件名均为调用方确定,均带文件路径。

接下来谈谈原有系统存在的问题:

因网上银行业务的特殊性,Web 客户端需要连接多个业务种类,多种数据库,跨平台,跨数据库,环节多,这就是原有系统交易缓慢的重要原因之一,而且随着可提供新业务的种类的增多,问题暴露的更加明显;其次 Web 客户端的查询往往对各个应用系统的服务器的负载产生影响,影响了其他业务;同样,对 Web 客户无法提供个性化服务,B2B,B2C 提供的信息和帐页千篇一律,根本无法按照其要求定制;客户端无法进行一些运算,一些简单的比如”“还款试算”等还要通过后台应用服务器来进行运算,加重了主机负担。

在具体应用中,比如查询某客户的所有业务是困难的,这具体体现在,如果某签约客户想查询或执行信用卡的历史交易,必须向后台相应的业务的数据库服务器提交数据查询请求,而此时的后台服务器又往往又是银行的实时业务处理机,在高峰期正忙,并且历史交易的数据往往由于年终转换的原因,存在于其它数据表甚至其他数据库和服务器,都给查询带来了困难,只能作到有限的查询,或提供的数据项有限,或甚至根本无法提供,比如有关储蓄业务的有关明细帐务已接近 3000 万条,在高峰期间在其中取几条记录是困难的。另一方面,由于网上银行涉及多个对私和对公的业务,甚至包括资金的清算,一笔交易要跨越多个业务的服务器,这又存在跨不同种类的数据库的问题。

所以,将历史数据分离及整和是必然的,我们也曾考虑过将数据进行归类,建立一个类似"历史数据服务器"(或是数据仓库)上,但该历史数据仍需建立在某种数据库上如 INFORMIX-ONS,仅作到了数据的集合,没有治本,投入大,费时。

所以为了解决原有系统中存在的问题,我们在新系统中我们建立专用的转换服务器,作为"缓存",目的仅是为了"脱离"原有的依赖,减少联机处理,这样,有关历史的交易就不用分别直接连到所需业务的数据库或服务器,我们采用 XML 格式进行中间的转换。这种所谓的 XML 数据库其实就是文档的集合。我们用了一台 IBM PC SERVER 来存储 XML 数据,具体就是用 UNIX 下的文件系统来存储和检索。将部分数据转换成 XML 文件,包括所有以已签约的客户的历史明细帐及全部卡号(未来新开户时就不用再去信用卡主机进行校验了),同时 XML 文件的存储按文件系统,并对 B2B 和 B2C 进行了分类。

这里我们使用了 Informix Web DataBlade 工具,按照我们自己定义的数据结构进行了批量转换,基本与网银后台数据库服务器的数据库表结构相符,这样做的目的是为了便于更新,且对原有有关调用数据库的数据逻辑改动不大,将数据转换成了 XML 文件格式,对于其他非 Informix 数据库的数据,我们则采取了先将数据倒入导入至 Informix 数据库,再进行这种转换。

有关当日的账务,我们在日终将当日发生的流水帐进转换,存入了数据库,作为了历史交易的追加。在这里 Informix Web DataBlade 可以通过一个简单的 SQL 接口产生动态的 XML 数据和文档,用于日终更新。这样实际就上在数据库与应用服务器之间采用 XML 作为信息缓存。

建好了 XML 数据库,那么又如何来应用呢?接下来我们谈谈这方面的问题。

首先读取 XML 数据。具体的方法为:一个页面通过服务器端对象与 XML 数据源相连,将信息转换成数据抽象,接着用 JSP 元素显示数据。这样我们就使用了 XSLT 转换程序转换 XML,。在这里应用到的 XSLT 技术,XSLT 是 W3C 小组制定的一个转换语言规范,它可以用来将 XML 数据转换成 HTML、PDF 或其它 XML 格式。具体过程就是首先定义了 XSLT 模板,然后进行转换,除了模板需要定义外,这个解析的过程是还是比较容易的。其实 XSLT 与 XML 的关系,就好象 SQL 与表格化的数据一样。只是在 JSP 中,需用 scriptlet 或自定义标签中编程激活一个 XSLT 处理器来进行转换。在 XML 数据源上使用一个转换程序,或者是抽取数据或者是创建新的格式,这个转换程序可以使用许多不同的机制来实现并且通过自定义标签来访问它。

接下来 XML 应用就是使同样的数据可以以不同的浏览方式出现在浏览器中,而这些数据并不需要从再次从 Web 服务器上下载。其实这类应用的就是早期的 Web 上的另我们头疼的"动态表格"。

对于 B2B 的用户,可根据企业要求的,制定不同的往来对帐单,约十多种,以满足其对帐的需要,对其所属职工的集体办理的有关代发工资,缴存公积金,还个人贷款的有关帐务,都有良好的支持,在单位就可了解到,随时的变动,并可随时打印,不用再跑银行进行询问,取帐页。将来甚至直接提供信息化较高的企业 XML 数据,这也是未来交换数据的标准,也是应用 XML 的美好憧憬.对于 B2C 的个人用户,可根据其所开办的业务的不同,来对其提供"个性化"的服务,可提供该人名下的所有帐户的信息,如存贷款信息,各种缴费的信息,并可根据其需求方便的进行排序,筛选,组和,打印。甚至可以自己定制格式,改变了以往显示单一,多业务罗列,操作复杂的状况。Java 提供的 JSP 为多种基于 Web 的用户产生基于 XML 的标记语言的问题,也就是我们要达到的"个性化"界面。

在具体应用中,我们是用从 JSP 页面产生的 XML, XML 在 Web 界面层的应用得益于 JSP 技术的发展。同样,也有两种方法,一种直接将 XML 数据源集成到 JSP 的界面中去的方法是,将 XML 加载到 JavaBeans 组件中,然后在 JSP 中直接引用这些 JavaBeans 组件。最大好处是使我们的程序代码集中在一个地方(对 Java 技术而言,一般是指在“类”中),清晰,易于管理和修改。另一种方法是直接将 XML 数据转换成 Web 显示内容的另一种方法是使用 XSL 和 XSLT,将 XML 数据映射成 HTML(或 WML 等)的逻辑由 XSL 样式表(XSL StyleSheet)来定义。样式表描述了每个特定 XML 数据实体应该怎样转换成界面数据实体(如 HTML 表格、内联标记等),采用一套自定义的 JSP 标记并引用某个 XSLT 处理程序,也就是前面提到的模板。

通过比较综合了这两种方案方法,我们作出了选择,选择了后者,XSLT 方案的伸缩性要好一些,而且具有更好的可管理性。在这种情形下,我们的转换逻辑是编写在一个 XSL 样式表中,而不是在 Java 代码中。这意味着当需要修改界面时,大多数情况下只是编辑样式表或者 HTML,代码不受影响,不涉及程序编译的问题,业务人员经过简单的培训就可以修改,这对我们很重要。

在网银项目中由于 XML 的应用,解决了我们许多历史问题,也使银行的 B2B 和 B2C 业务进行了有效的整合,更好的为客户服务。

尽管 XML 是未来网上数据格式的标准,可在具体应用中还存在许多问题:

尽管我们由于网上银行所要求的数据格式不是很多,可建立 XSLT 的模板并不是一件容易的事,不是很好写,主要是由于表中嵌套太多,尽管是文本形式,阅读和修改很实际上是复杂的。

另外,这种基于文件系统的后台历史数据按 XML 存储的方案还有待探讨,尽管 XML 的确与数据库有相似之处,但对于索引、安全机制、数据完整性等特性是不具备的。在实际应用中,我们目前由于签约的客户数量不是很多,不足万人,尽管我们只应用了一台服务器来存储数据,目前还没有什么问题,但随着签约客户的增多,不知会不会有问题。看来 XML 还是要发展,想取代数据库还是需要一段时间的。

还有,我们未将 XML 封装进 JavaBean,却将部分应用逻辑由 XSLT 来实现,这种方法是否合适还有待探讨,从某种程度来说是不符合惯例的,不仅仅是 Web 服务器负载的问题,它给编程人员和维护人员带来的问题有些还是不可预计的。

我想,未来 XML 会成为数据交换的标准的,我们也考虑逐步将我们现有的数据转换成 XML 格式。接下来,我们还将在这些数据的基础上,建立一个统一的,用 XML 实现通用 WEB 报表的系统,在这方面已有了成功的案例。

现在,大部分数据库支持 XML 格式的数据查询和转换,包括 SQL Server 2000,ORACLE,IBM DB2 等大型关系数据库,估计会越来越多,我想以上问题也会很快得到解决的。

论 XML 技术在 Internet 平台上的应用

摘要

2002 年 10 月, 我参与了一个三层在线商城的项目开发, 该项目整合了来自不同商家的信息, 方便在线用户的查询和购买。

在该项目中, 我担任系统分析的工作。在分析设计过程中, 我借鉴了 XML 成熟的技术, 采用 Java 语言, 整个系统由三层组成。在数据层, 对于不同的数据库, 最后都以 XML 数据的形式来实行数据间的转换和处理。在业务逻辑层, 在联机会话的持续时间内, 用户的帐户数据在内存中以 XML DOM 形式表示, 在表示层, 所以给用户的信息首先都封装成 XML 数据, 然后用服务器或者客户机附带的 XSLT 转换, 根据浏览器的性能将 XML 数据转换为 HTML 在前端显示。

在设计过程中, 如设计 XML 的各个基本元素, 我应用域分析的方法, 在采用 XML DOM 形式的时候, 分析比较了其他的形式, 在将 XML 转换为 HTML 的设计中, 引用了 XSLT。

正文

随着 Internet 信息技术的发展, 我所在的公司准备开发一个网上商城, 这样各个商家就可以把自己的产品信息在这个网上商城中发布, 并且提供了在线购买。也就是开发一个电子商务平台, 在这个平台上, 整合了来自不同商家的信息, 方便在线用户的查询和购买。有点类似现实中的商城, 为各个商家提供地方, 方便消费者购买。

本人有幸成为这个项目的系统分析员之一, 参与了这个系统的设计, 并且对系统中的关键实现技术也进行了一一验证。整个网上商城系统由三层组成。在数据层, 由于各个商家自己有着不同的数据库来存储自己信息, 为了实行信息在同一平台上的共享和处理, 我们采用 XML 数据的形式来实行数据间的转换和处理。在业务逻辑层, 在联机会话的持续时间内, 用户的帐户数据在内存中以 XML DOM 形式表示, 在表示层, 所有给用户的信息首先都封装成 XML 数据, 然后用服务器或者客户机附带的 XSLT 转换, 根据浏览器的性能将 XML 数据转换为 HTML 在前端显示。

在数据层, 我们面临的第一个的问题就是如何统一认识将要采用的 XML 数据的元素。刚开始是由一个人来定义 XML 数据的元素, 但是后来发现, 这样定义处理的 XML 元素很难获得别人的认同, 并且对于不同的商家, 所定义的 XML 元素不具有代表性。于是, 在争取到领导的支持后, 把以前的 XML 元素设计推倒重来, 而且借鉴了国外关于类似设计所采用的域分析的方法, 该方法就是一个用于确定网上商城这个域的术语, 范围, 共性和变性的过程。就这样为了寻求一个统一的 XML 元素的定义, 我们成立了一个小组来进行的网上商城的域分析, 这个小组有商家代表, 数据库设计员, 参与多个项目的有丰富经验的程序员和一个专门指导该组域分析的大学专家组成。

在小组会议的开始, 我们首先达成一个共识, 那就是需要采用一直标准术语来避免交流中产生的误解。于是, 我们在刚开始的一周内先确定的关于网上商城的一下公用术语, 比如 store (商城), shop (店面), ware (商品) 等, 作为 XML 最基本的元素。

在小组讨论的过程中, 有成员建议对于域分析, 分多个阶段进行, 每个阶段提交域分析报告, 比如第一阶段的域分析报告主要为标准术语以及各个商家信息共性和可变性的表格, 第二阶段的域分析报告, 对于域分析中的商家可变性的东西进行详细说明。我们采纳这个建议, 获得了很好的效果。就这样, 通过 3 周的域分析, 我们小组最终提交一份完整的关于 XML 网上商城元素的详细表格, 由于这些 XML 元素是由各个部分和商家代表讨论处理的, 所以很快被整个项目组成员所接收, 并且这些 XML 元素在后来的开发中证明是非常完整, 能够清楚地反映数据的结构, 大大提高了整个的系统的开发效率。

在业务逻辑层, 由于所有数据库的信息都被转换为 XML 数据结构, 所以在处理数据库信息的时候还必须对 XML 数据进行 XML 语法分析, 并且将分析出来的结果送往程序。在这儿我们考虑采用 XML DOM (文档对象模型) 来保存这些语法分析出来的 XML 数据。比如在联机会话的持续时间内, 一个用户的帐户数据首先从传统的关系型数据库中读取出来, 转换为 XML 数据结果, 并且通过一个 XML 语法分析器, 将 XML 数据转换成为 DOM 对象保存在内存中, 程序通过 Java 的 DOM API 访问这些对象。在这儿为了更好地选择处理 XML 的技术和方法, 我还比较其他两种分析 XML 数据并且保存的方法。

比较了几种方法, 发现 DOM 有个缺点就是当它保持的数据非常多的时候, 将大量占用内存的存储空间。但是, 使用 DOM API 也有一个明显的好处就是简单, 它可以通过 Java 程序直接使用一些方法调用 DOM 树上的数据。为此在设计的时候, 尽可能地重复调用已经存储在内存中的 DOM 对象上面的数据, 避免对于相同的数据有多个 DOM 对象存在。在编码过程中面临的问题不是很多, 关键是让程序员熟悉 DOM API 的各种调用方法。

在表示层, 考虑到将大量的运算负荷分布在用户端, 既用户可以根据自己的需求选择或者制订不同的应用程序以处理数据, 我们设计把所有给用户的信息首先都封装成 XML 数据, 然后用服务器或者客户机附带的 XSLT 转换, 根据浏览器的性能将 XML 数据转换为 HTML 在前端显示。这样的话, 服务器只需要准备一次的 Web 内容, 尽可能完善, 准确地将数据封装到 XML 文件中, 而 XML 的自解释性可以使得用户端在接收到 XML 文件的同时也理解数据的逻辑结构和含义, 再通过转换程序, XML 可转换为用户所需要的个性的多样的 HTML 显示方式。在设计中我们所选取的 XSLT 是一种用于操作 XML 文档的高级语言, 就像 SQL 是操作关系表的高级语言一样。

一个 XSLT 规范本事就是一个 XML 文档, 我们通过它的规范, 描述出各个用户可以选择的各个 HTML 显示模板, 这样客户端程序只需选择好模板, 再加上接收到的 XML 数据文件, 就可以方便地生成自己个性化的 HTML 文件, 并且在浏览器中显示出来。当然在开发中遇到了不少问题, 首先由于 XSLT 是一门起点比较高的语言, 也比较烦琐, 掌握起来比较慢, 使得对程序员的要求比较高。这样导致使用 XSLT 语言编写的 HTML 文件的模板过于冗长, 作为设计师我一方面是希望模板程序尽可能地包含 HTML 页面所有的功能和美观, 一方面又希望模板程序尽可能地短少, 简洁, 便于修改。

为了达到两种选择之间的平衡, 以达到最佳效果, 我对于基本模板的设计, 采用有经验的程序员进行设计, 并且要求尽可能多地写全文档, 并且频繁地开展小组会议, 对页面设计和美工人员详细讲述模板各个部分和功能。

在 Internet 平台上采用 XML 技术, 明显的效果之一对于不同数据库的支持, 通过将各种数据转换为 XML 文件, 可以实行了数据间的转换, 共享和处理。效果二是支持了用户的个性化服务, 支持用户在不同的客户端可以个性地选择显示界面。

由于 HTML 在许多复杂的 Web 应用中遇到了问题, 为了彻底解决这些问题, 必须采用功能强大的 XML 来代替 HTML 作为 Web 页面的书写工具, 而 XML 的广泛使用, 必定能够推动 Web 的不断发展, 开创 Web 应用的新时代。对于 XML 技术在 Internet 平台上的应用, 我更关注它的扩展性, 既让 XML 包含更加丰富更完整的数据信息, 目前公司所接触的首先是软件模型的交换和模型信息的保存, 打算在 XMI (XML Metadata Interchange) 展开工作。XMI 作为一种试图通过 XML 语言为程序员和其他用户提供一种交换元数据信息的标准途径, 是作为 MDA 模型驱动的模式交换的基础, 是非常具有意义的工作。

图书馆网络应用体系安全设计

摘要

某某大学图书馆从 85 年引进日本富士通的管理系统开始, 历经近 20 年的信息化建设后, 逐步形成了拥有一定硬件规模、软件资源和一批专业技术人才的现代化图书馆。而自从我校进入“211”以来, 建立数字化的图书馆就成为了我们工作的重中之重。我校数字化建设的主要内容是建立基于千兆主干网络的、提供多种网络服务的网络应用体系。项目建设完成后, 我馆成为该省规模最大的开放式的数字化文献提供和建设中心, 为本省的科技、文化、教育事业的发展提供了强大的资源保障。

2001 年 3 月, 我参加了某大学数字图书馆建设, 担任技术部主管职务, 负责理解学校和省教委对该项目的要求, 根据技术先进、成本适中、充分满足要求为原则, 进行应用需求分析, 对整个网络进行设计, 提出设计方案。由于该项目规模比较大, 提供的服务比较多, 作为校园网络的中心和提供网络服务的核心部门, 我馆图书馆的许多业务需要在网上展开, 各种特色服务所用的平台也是五花八门。这对安全方面的设计提出的较高的要求。我们通过采用保障 Internet 接入的安全、保证干路畅通和合理划分子网、保证软件系统的安全、健全管理机制等措施来设计图书馆网络。这种切合实际、低成本、高技术的设计方案实施后的网络, 其安全性大大加强了, 同时网络的性能并未受到太多的影响。

正文

2001 年 3 月, 我参加了某某大学数字 211 建设图书馆数字化建设部分(分为第一、第二、第三期工程)。第二期工程总额 5500 万元。该工程建设项目分为网络部分、软件部分以及资源建设部分。该项目的目标是, 将地处该城市的东、西、南、北端的各分校区和总校区的网络联连接成为一个畅通的宽带、高速、高性能的校园网; 建立一个文献信息中心, 能为读者提供电子期刊服务、图书书目数据服务、图书光盘点播、视频点播服务, 并且能够拥有部分自己的特色数据资源; 建立一个网络中心提供 Web 服务、邮件服务、全校的办公自动化服务, 它既是全校的应用服务中心、公共数据存储中心, 又是全校网络的管理中心。项目从实现的功能、设备配置的先进性、网络带宽的规模看, 该项目要达到国内一流、省内第一的水平。项目建成后, 该数字图书馆能承担全省的文献检索与开发的重任。我在该项目中担任技术部主管, 其任务是充分理解学校和省教委对该项目的要求, 根据技术先进、成本适中、充分满足要求为原则, 提出和需求分析书和项目计划书; 监督光纤的铺设和楼宇布线; 设备的货到验收; 网络调试(交换机调试、地址分配、访问控制配置、设备网管模块的配置、网络管理平台的配置); 网络应用系统的建立; 数据库的采购; 建立特色的数字化资源等等。进行应用需求分析, 对整个网络进行设计, 提出设计方案。

由于该项目规模比较大, 提供的服务比较多, 各种特色服务所用的平台也是五花八门。同时, 影响网络安全的因数比较多, 通常有人的因数、自然因数、病毒因数。如果对这些主要因数防范不得力, 将影响网络硬件、网络数据传输、数据服务器的安全。因此对其安全设计, 单一的技术或者设备保护难以保证本校网络的安全, 效果往往不理想; 所以我们采用多层次的防护体系来保证网络的畅通和网络数据的安全。我们根据业务数据的流动历经的重要环节, 来提出具体的安全方案。

1 保证接入的安全

俗话说病从口入, 同样一些常有的攻击往往来自外网, 同时使用图书馆所购买的外文资料比如 Web Science、EBSCO 等数据库需要访问外网; 同时, 外网的授权用户需要能够访问我们的数据。所以必须保证接入部分的稳定与安全。在外网联接上, 我们租用 2 条光纤线路(100M)分别接入中国电信和 XXX 大学, 使用 Cisco 的 7500 系列路由器接入, 然后用一台 Sun spark 操作系统为 Solaris 的服务器做防火墙将外网和内网划分开; 同时用其做策略路由服务器, 使用校园网内的用户可以快捷的访问电信网络和教育网络的资源。在外部用户访问上, 我们首先用 Sun spark 服务器防火墙进行 IP 地址过滤掉非法的 IP 地址, 然后通过用户名+密码模式登陆, 才能通过防火墙访问我们内部资源。

2 保证干路畅通和划分子网

网络干路是指各楼的骨干网、骨干网汇接形成的交换中心及联结 Internet 的接入线路。它的不畅通必然导致大规模的网络瘫痪, 外部付费用户访问不到内部资源, 这将造成极坏的影响。

在光纤干路上我们采用光纤连接主校区的各大教学楼、南院校区和 XXX 校区, 主干路使用 3 对光纤做冗余。由于其它分校区离主校区比较远, 光纤连入费用较高, 我们采用拨号接入的方式联结。我们采用 3ComRAS1500 作为拨号服务器, 分配 32 个校内电话号码给该服务器, 用户只需要拨这 32 个其中一个, 通过认证后, 就可以动态分配一个校内的合法 IP 来访问校内的资源。在干路上, 安全的隐患往往来自非技术因素, 由于本市修建大学城, 道路施工比较多, 我们的光纤被人为的挖断; 由于, 某路段起火烧掉部分光纤。对于这种情况, 我们采取紧急修理, 租用吉通的 2M 微波通讯线路和将策略路由指向电信线路的办法来保障网路的畅通。

由于本校应用点多, 地理位置也十分分散, 各大单位为相对独立的机构, 所以需要进行子网划分以便管理。

我们使用两台 3Com 的 CB9000 中心交换机连接各教学单位过来的光纤, 并且将华中分配的 16 个 C 类地址分成为 32 个 Vlan。子网的划分使安全性得到了很大的提高, 同时各单位的网络独立的虚网。同时, 每个教学单位指定专人负责网络地址的分配, 这样盗用 IP 地址的情况大大的减少了。

我们采用 3Com 的 Transcend 作为网管平台, 实施对网络干路 and 各个子网的监控和进行网络优化。当网络有人恶意下载资源和蠕虫病毒群发邮件时, 我们可以立即找出攻击源而进行相应的措施。

3 保证软件系统的安全

由于操作系统是软件系统的基础, 所以, 保护好操作系统是必须的, 对于 Windows 系统(提供 CNKI 和 VIP 镜像等服务)保护, 我们主要采取取消 Guest 帐号、取消不必要的服务(如远程注册表操作)安装诺顿防火墙和相应的杀毒软件、定时查看有无异常的程序运行等方式来保证其安全。对于 Linux 或者 Unix(提供邮件、网络计费、主页、论坛等服务), 我们关闭许多不必要的端口、定时对系统打包升级等措施来保证其安全。

对于数据库系统的保护, 主要集中在保证数据的完整性、正确性和安全性方面。在数据存储介质方面, 我们采用磁盘阵列、NAS、San 结合 RAID5(无独立校验磁盘的奇偶校验磁盘阵列)方式存储数据。

4 健全的管理机制

管理安全数字图书馆的工作状态在很大程度上取决于是否有良好的管理机制。如果制度合理、管理得当、执行得力就能有效的预防和控制安全事故的发生。反之, 则会引发各种安全隐患。

我们规定工作人员每日必须检查服务器的日志, 通过日志可以清楚的看到有无外来人员登陆服务器, 并且做了何种修改。对重要的数据服务器, 每日必须做异地数据备份。同时管理员的密码必须达到一定的长度并且每周建议修改一次。

一般而言, 网络安全是网络服务效率的保障。没有了数据安全, 网络服务成为无源之水; 没有网络系统的安全, 网络服务将成为无本之木。当然没有效率的网络传输是得不偿失的。所以, 我们要合理的规划好网络及软、硬件的合理利用。

在接入上, 由于全校有 3, 000 多台机器也要通过该做防火墙的服务器接入 Internet, 这将给该服务器造成较大的负担, 同时数据传输量大时, 该服务器处理速度相对将成为瓶颈。我们采取单独使用两台 Spark 主机做代理服务器, 代理服务器同时启动用户认证服务, 由于代理服务器的缓存保证了出校的流量大大减少, 同时认证功能使得非法用户无法登陆网络。这样的冗余设计使得网络的安全性能和效能得到了大幅度的提高。

在子网划分上, 连接各教学单位过来的光纤的 2 台中心 3Com CB9000 交换机之间用两根光纤作一个串口, 这样跨子网的访问无需通过交换能力不强的 Cisco 路由器 7500, 子网间传输数据的瓶颈问题消失了。

在数据存储上, 由于我们提供 CNKI 和 VIP 等服务, 这些服务器的访问量、存储的数据量都十分庞大, 单个的数据检索查询都会占据大量的 CPU 时间。我们使用 SAN 和 NAS 存储大量的数据, 当用户提出一个检索的申请时, 检索服务将在服务器上运行, 当用户提出一个下载的服务时候, 下载的服务由 SAN 和 NAS 去完成, 这样使得 SAN 和 NAS 的效能比磁盘阵列要高。由于 SAN 和 NAS 都可以做 RAID1~5, 所以数据安全性能比普通的硬盘要高。采用新技术和冗余使得我们的网络服务效率和安全性能得到了大幅度的提高。

网络的安全管理不是一个硬件和软件的引进就能得到大幅度的提高的, 他是一个长期的、动态的过程。本项目的网络系统还有许多问题, 一次网络发生了大规模的网络风暴, 从子交换机扩散到中心交换机, 主机与主机之间的数据传输丢包率达到了 75%几乎是不通了, 最后我们发现一个施工的单位将一根漏电的线缆搭在网路上。但是为什么会造成跨虚网的网络风暴, 从技术上我们无法解决, 只能是严格检测各子网和主干网络是否达到屏蔽的要求, 并且将电路和网路尽量分开。

论计算机网络的安全性设计

摘要

我在一家证券公司信息技术部门工作，我公司在 9798 年建成了与各公司总部及营业网点的企业网络，并已先后在企业网络上建设了交易系统、办公系统，并开通了互联网应用。因将对安全要求不同、安全可信度不同的各种应用运行在同一网络上，给黑客的攻击、病毒的蔓延打开了方便之门，给我公司的网络安全造成了很大的威胁。作为信息技术中心部门经理及项目负责人，我在资金投入不足的前提下，充分利用现有条件及成熟技术，对公司网络进行了全面细致的规划，使改造后的网络安全级别大大提高。本文将介绍我在网络安全性和保密性方面采取的一些方法和策略，主要包括网络安全隔离、网络边界安全控制、交叉病毒防治、集中网络安全管理等，同时分析了因投入资金有限，我公司网络目前仍存在的一些问题或不足，并提出了一些改进办法。

正文

我在一家证券公司工作，公司在 98 年就建成了与各公司总部及营业网点的企业网络，随着公司业务的不拓展，公司先后建设了集中报盘系统、网上交易系统、OA、财务系统、总部监控系统等等，为了保证各业务正常开展，特别是为了确保证券交易业务的实时高效，公司已于 2002 年已经将中心至各营业部的通讯链路由初建时的主链路 64K 的 DDN 和备链路 33. .3KPSTN，扩建成主链路为 2M 光缆作为主链路和 256K 的 DDN 作为备链路，实现了通讯线路及关键网络设备的冗余，较好地保证了公司业务的需要。并且随着网上交易系统的建设和网上办公的需要，公司企业网与互联网之间建起了桥梁。改造前，应用系统在用户认证及加密传输方面采取了相应措施，如集中交易在进行身份确认后信息采用了 Blowfish128 位加密技术，网上交易运用了对称加密和非对称加密相结合的方法进行身份认证和数据传输加密，但公司办公系统、交易系统、互联网应用之间没有进行安全隔离，只在互联网入口安装了软件放防火墙，给黑客的攻击、病毒的蔓延打开了方便之门。

作为公司信息技术中心运保部经理，系统安全一直是困扰着我的话题，特别是随着公司集中报盘系统、网上交易系统的建设，以及网上办公需要，网络安全系统的建设更显得尤为迫切。但公司考虑到目前证券市场疲软，竞争十分激励，公司暂时不打算投入较大资金来建设安全系统。作为部门经理及项目负责人，我在投入较少资金的前提下，在公司可以容忍的风险级别和可以接受的成本之间作出取舍，充分利用现有的条件及成熟的技术，对公司网络进行了全面细致的规划，并且最大限度地发挥管理功效，尽可能全方位地提高公司的网络安全水平。。在网络安全性和保密性方面，我采用了以下技术和策略：1、将企业网划分成交易网、办公网、互联网应用网，进行网络隔离。2、在网络边界采取防火墙、存取控制、并口隔离等技术进行安全控制。3、运用多版本的防病毒软件对系统交叉杀毒。4、制定公司网络安全管理办法，进行网络安全集中管理。

一、网络安全隔离

为了达到网络互相不受影响，最好的办法是将网络进行隔离，网络隔离分为物理隔离和逻辑隔离，我主要是从系统的重要程度即安全等级考虑划分合理的网络安全边界，使不同安全级别的网络或信息媒介不能相互访问或有控制的进行访问。针对我公司的网络系统的应用特点把公司证券交易系统、业务办公系统之间进行逻辑分离，划分成交易子网和办公子网，将互联网应用与公司企业网之间进行物理隔离，形成独立的互联网应用子网。公司中心与各营业部之间建有两套网络，中心路由器是两台 CISC07206，营业部是两台 CISC02612，一条通讯链路是联通 2M 光缆，一条是电信 256K DDN，改造前两套链路一主一备，为了充分利用网络资源实现两条链路的均衡负载和线路故障的无缝切换，子网的划分采用 VLAN 技术，并将中心端和营业部端的路由器分别采用两组虚拟地址的 HSRP 技术，一组地址对应交易子网，一组地址对应办公网络，形成两个逻辑上独立的网络。改造后原来一机两用（需要同时访问两个网络信息）的工作站采用双硬盘网络隔离卡的方法，在确保隔离的前提下实现双网数据的安全交换。

二、网络边界安全控制

网络安全的需求一方面要保护网络不受破坏，另一方面要确保网络服务的可用性。将网络进行隔离后，为了满足网络内的授权用户对相关子网资源的访问，保证各业务不受影响，在各子网之间采取了不同的存取策略。

(1)、互联网与交易子网之间：为了保证网上交易业务的顺利进行，互联网与交易子网之间建有通讯链路，为了保证交易网不受互联网影响，在互联网与中心的专线之间安装了 NETSCREEN 委托防火墙，并进行了以下控制：a)、只允许股民访问网上交易相应地址的相应端口。b)、只允许信息技术中心的维护机地址 PING、TELNET 委托机和路由器。c)、只允许行情发送机向行情主站上传行情的端口。d)、其他服务及端口全部禁止。并且在互联网和交易网之间还采用了 SSL 并口隔离，进一步保证了交易网的安全。

(2)、交易网和办公网之间：对于办公网与交易网之间的互访，采用 CISC02501 路由器进行双向控制或有限访问原则，使受控的子网或主机访问权限和信息流向能得到有效控制，主要采用的策略主要是对具体 IP 进行 IP 地址与 MAC 地址的绑定。

(3)、办公子网与互联网之间：采用东大 NETEYE 硬件防火墙，并进行了以下控制：a)、允许中心上网的地址访问互联网的任何地址和任何端口。b)、允许股民访问网上交易备份地址的 8002 端口。c)、允许短消息访问公司邮件 110、25 端口，访问电信 SP 的 8001 端口。d)、其他的都禁止。

三、病毒防治

网络病毒往往令人防不胜防，尽管对网络进行网络隔离，但网络资源互防以及人为原因，病毒防治依然不可掉以轻心。因此，采用适当的措施防治病毒，是进一步提高网络安全的重要手段。我分别在不同子网上部署了能够统一分发、集中管理的熊猫卫士网络病毒软件，同时购置单机版 KV3000 和瑞星防病毒软件进行交叉杀毒；限制共享目录及读写权限的使用；限制网上软件的下载和禁用盗版软件；软盘数据和邮件先查毒后使用等等。

四、集中网络安全管理

网络安全的保障不能仅仅依靠安全设备，更重要的是要制定一个全方位的安全策略，在全网范围内实现统一集中的安全管理。在网络安全改造完成后，我制订了公司网络安全管理办法，主要措施如下：1)、多人负责原则，每一项与安全有关的活动，都必须有两人或多人在场，并且一人操作一人复核。2)、任期有限原则，技术人员不定期地轮岗。3)、职责分离原则，非本岗人员不得掌握用户、密码等关键信息。4)、营业部进行网络改造的方案必须经过中心网络安全小组审批后方可实施。5)、跨网互访须绑定 IP 及 MAC 地址，增加互访机器时须经过中心批准并进行存取控制设置后方可运行。6)、及时升级系统软件补丁，关闭不用的服务和端口等等。

保障网络安全性与网络服务效率永远是一对矛盾，在计算机应用日益广泛的今天，要想网络系统安全可靠，势必会增加许多控制措施和安全设备，从而会或多或少的影响使用效率和使用方便性。如，我在互联网和交易网之间设置了放防火墙的前提下再进行了 SSL 并口隔离后，网上交易股民访问交易网的并发人数达到一定量时就会出现延时现象，为了保证股民交易及时快捷，我只好采用增加通讯机的办法来消除交易延时间题。

在进行网络改造后，我公司的网络安全级别大大提高。但我知道安全永远只是一个相对概念，随着计算机技术不断进步，有关网络安全的讨论也将是一个无休无止的话题。审视改造后的网络系统，我认为尽管我们在 Internet 的入口处部署了防火墙，有效阻挡了来自外部的攻击，并且将网络分成三个子网较减少了各系统之间的影响，但在公司内部的访问控制以及入侵检测等方面仍显不足，如果将来公司投资允许，我将在以下几方面加强：

- 1、在中心与营业部之间建立防火墙，通过访问控制防止通过内网的非法入侵。
- 2、中心与营业部之间的通讯，采用通过 IP 层加密构建证券公司虚拟专用网（VPN），保证证券公司总部与各营业部之间信息传输的机密性。
- 3、建立由入侵监测系统、网络扫描系统、系统扫描系统、信息审计系统、集中身份识别系统等构成的安全控制中心，作为公司网络监控预警系统。

论计算机网络的安全性设计

摘要

在计算机与网络技术飞速发展的今天，医院信息系统的建设已经成为医院现代化管理的重要标志，同时也是医院管理水平的一种体现。尤其是医疗保险制度的改革，与医院信息系统形成了相互促进的态势，我国很多医院都建立了自己的信息系统。由于行业性质的缘故，医院信息系统必须 7 X 24 小时不间断运转，因此对网络系统的安全性和可靠性有很高的要求。本文通过一个医院信息系统项目，阐述了医院计算机网络的安全性设计方面的一些具体措施，并就保障网络的安全性与提高网络服务效率之间的关系，谈了自己的一点体会。

正文

我于 2001 年 4 月至 2003 年 10 月参加了某医院的医院信息系统的建设工作，在项目中，我担任了系统分析与系统设计工作。医院信息系统是指利用计算机软硬件技术、网络通讯技术等现代化手段，对医院及其所属各部门对人流、物流、财流进行综合管理，对在医疗活动各阶段中产生的数据进行采集、存贮、处理、提取、传输、汇总、加工生成各种信息，从而为医院的整体运行提供全面的、自动化的管理及各种服务的信息系统。由于行业性质的缘故，医院信息系统必须 7 X 24 小时不间断运转，因此对网络系统的安全性和可靠性有很高的要求，在该项目的系统设计阶段，我们就将网络系统的安全性作为一个重要部分考虑在内。由于该信息系统是建立在一个物理上与公众网完全隔离的局域网基础上的，所以我们并没有过多地考虑防御来自外部入侵者的威胁方面的安全问题，我们认为该系统的安全核心一是保证信息系统的正常运行，二是保证数据的安全，也就是说该医院网络信息系统的安全可以分为信息系统安全和数据安全。下面就我们在这两方面所采取的措施加以论述。

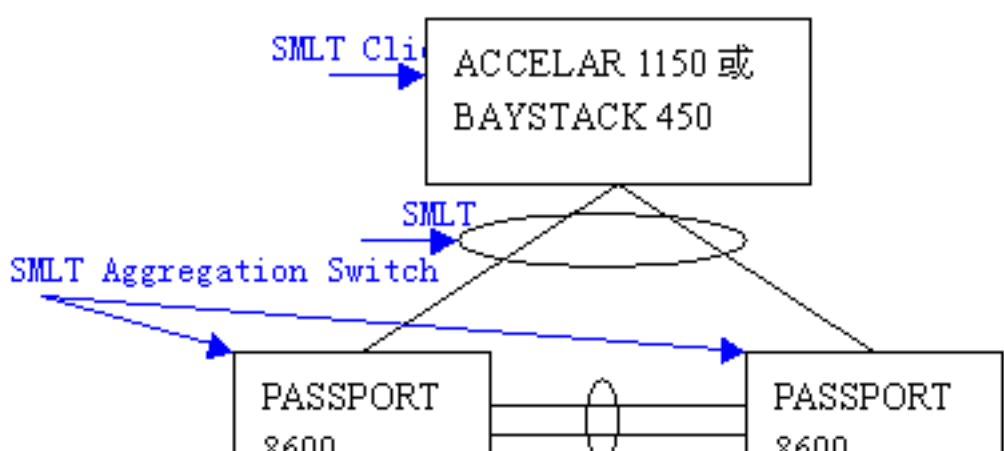
信息系统安全

信息系统安全涉及网络安全、服务器组的安全、供电安全、病毒防范等。

1、网络安全

对于医院的业务局域网，威胁网络安全的主要因素有：网络设计缺陷、网络设备损坏、非法访问等。经过充分调研，认真分析，结合该医院的实际情况，我们设计了一个主干为三层路由千兆交换以太网的网络方案。

我们采用具有三层路由功能的两台核心交换机 NORTEL PASSPORT 8600、两台具有三层路由功能的 NORTEL ACCELAR 1150 交换机和千兆级光纤组成网络主干，边沿交换机为 BAYSTACK450。本方案我们采用 SMLT (Split Multi-Link Trunking) +VRRP (Virtual Router Redundancy Protocol) 技术。NORTEL 公司的 MLT (Multi-Link Trunking) 是一种允许多条物理链路模拟成一条逻辑链路的聚合链路协议，它通过将两个交换机之间（或交换机与服务器之间）的两条或以上的物理传输链路虚拟为一条逻辑上的传输线路进行数据传输，进而可以成倍地提高两个交换机之间（或交换机与服务器之间）的数据传输带宽，同时提供了传输链路的冗余备份。当构成虚拟传输链路的几条物理链路有一条由于端口或传输介质本身失效时，不会影响数据的正常传输，所受到的影响仅仅是虚拟链路的传输带宽。SMLT，分离的多链路聚合主干，同 MLT 相比，SMLT 在构成上，不再是两个交换机之间，SMLT 的一端是一个支持 MLT 的交换机，而另一端则是由两个交换机通过 IST (Inter Switch Trunk，是连接两台聚合交换机以实现信息共享，使两台聚合交换机能作为一台逻辑交换机运转的点对点链路) 形成的一个逻辑上的交换机。MLT 交换机分别与这两个 SMLT 交换机连接，因此，SMLT 在增加带宽的同时，可以提供最高级别的可靠性——交换机级别的可靠性。两个 SMLT 交换机不论是端口失效还是端口模板失效，甚至是交换机失效都不会影响数据的正常传输，避免了单点失效对网络正常连通带来的影响。同时，传输负载由两个交换机来均衡完成，可以大幅度提高网络主干的传输性能。SMLT 体系结构由 SMLT Aggregation Switch、IST (Inter Switch Trunk) 和 SMLT Client 构成，其结构图如下：



在没有使用 SMLT 的情况下启动虚拟路由冗余协议 (VRRP)，通常只有主交换机进行数据包的转发，如果主交换机出了故障，备用交换机会自动顶替主交换机，完成数据包的转发工作；使用 SMLT，使得 VRRP 的性能得到扩展，除了主交换机进行数据包的转发外，备用交换机也进行数据包的转发，主交换机和备用交换机互为备份并互相侦听，这样既可以实现流量的负载均衡，也可以实现故障恢复，避免单点失效。为了避免边沿交换机出现单点失效，我们采用了堆叠技术，把若干台 BAYSTACK450 用堆叠电缆堆叠起来，在堆叠的某些交换机上加装光纤模块，由这些光纤端口捆绑成一条逻辑链路上联到网络主干，这样就算堆叠中的某台交换机损坏了，整个堆叠还可以正常工作。特别地，门诊收款处和门诊药房是医院的窗口单位，为了避免由于门诊楼交换机堆叠中的某台交换机出现了故障而导致门诊收款系统和门诊发药系统瘫痪，我们把门诊收款工作站和门诊发药站分散地接到堆叠中的七台交换机中。

在防止非法访问方面，我们采用了密码管理、权限设置、虚拟子网 (VLAN) 的划分等措施。

2、服务器组的安全

服务器是全院计算机网络的大脑和神经中枢，保证服务器可靠长期有效的运行是网络信息系统安全的一个特别重要的问题。

由于本方案中的应用程序是采用安全性较高的三层体系结构，所以服务器组包括域控制器、应用服务器和数据库服务器。

域控制器我们采用了两台稳定性较好的 IBM xseries 230 服务器，一台做主域控制器，另一台做备份域控制器，这样既可以实现登陆验证的负载均衡，又可以避免域控制器的单点失效问题。

应用服务器部分我们采用了六台 HP 380G3 服务器和一台 F5 BIG-IP5000 控制器。BIG-IP 控制器是针对企业本地网站或数据中心的一种产品。它能够提供高可用性和智能负载平衡功能。六台 HP 380G3 服务器通过 F5 BIG-IP5000 控制器连接到核心交换机 PASSPORT 8600，F5 BIG-IP5000 控制器可以持续监视六台 HP 380G3 服务器，以确保服务器运行正常，然后再自动将输入的服务请求路由到六台中可用性最高的服务器。这样连接，只要有一台 HP 380G3 服务器不出现故障，中间层应用程序便可以正常运行。这样设计既可以实现中间层应用程序的负载均衡，同时在 F5 BIG-IP5000 控制器不出现故障的前提下，又避免了应用服务器的单点失效问题。

数据库服务器部分我们采用了一台稳定性较高、存储性能较好的 HP DL760 G2 服务器、一台 HP DL580 G2 服务器和一台 HP MSA1000 光纤磁盘阵列柜。两台服务器分别通过光纤通道连接到磁盘阵列柜，组成存储局域网 (SAN)。本方案采用了微软的群集技术，实现了 Active/Passive 双机热备份模式，HP DL760 G2 做主数据库服务器，HP DL580 G2 做备份数据库服务器，在主服务器发生故障的情况下，备份服务器将自动在 30 秒内将所有服务接管过来，从而保证了数据库服务器的正常运行。在磁盘阵列柜，我们安装了 5 块 146G 的 SCSI 服务器硬盘，其中 4 块硬盘做 RAID5，一块硬盘做 Hot spare，这样可以保证阵列柜在两块硬盘发生故障时，系统还可以正常运行。

3、供电安全

由于医院许多大型诊疗仪器设备启动时有瞬间高压、高磁场等，会对计算机产生影响因此我们要求院方做到中心机房的电源专线专供，同时采用功率足够大的 UPS。

4、病毒防范

我们通过设置 VLAN 和要求院方安装网络版杀毒软件来防范病毒。

数据安全

我们采用数据备份来保证数据安全。

本方案我们采用冗余备份策略。1、利用 Veritas Backup Exec 9.1 软件将数据备份到磁带库中。Veritas Backup Exec 能为跨网络的服务器和 workstation 提供快速可靠的备份和恢复能力。我们利用 Veritas Backup Exec 的作业管理功能设置备份定时任务，每天进行一次数据库数据完全备份，每三个月进行一次系统的灾难备份。灾难备份能在数据库服务器崩溃时，避免重装系统，利用最新的数据备份使系统尽快恢复到运行状态。2、利用 MS SQL SERVER 2000 自身的备份功能，每天定时自动地进行一次数据完全备份，备份数据存放到另一台数据备份服务器 HP ML570 中，同时在 HP ML570 中设置一定时任务，对每天的备份数据进行一次完整性检测，这样可以保证备份数据是完整、可用的。

通过数据备份，能使医院在破坏数据的灾难事件中造成的损失降到最低。

计算机网络安全是一个系统工程，除了采用保障网络安全的技术外，还要加强安全教育和制度管理，因此我们强烈要求院方重视对各级计算机操作人员进行计算机网络安全的教育，并制定较为完善的计算机网络管理制度，如严禁非操作人员使用电脑；计算机中心指定专职系统管理员掌握服务器密码，每次更新或升级计算机软件必须有两人同时在场，并做好记录等。

在整个项目方案中，我们用于保障计算机网络系统安全的措施主要是设备的冗余、链路的冗余。采用冗余措施，除了可以避免单点失效问题，还可以增加网络带宽和实现业务流量的负载均衡。因此，保障计算机网络的安全性不仅可以保证网络服务的持续不中断，还可以更好地提高网络服务效率。

整个项目完成至今近一年，从系统运行的情况来看，整个方案是合理的，高可靠性的，院方也感到很满意。当然，方案中也有不足的地方，如 F5 BIG-IP5000 控制器存在单点失效问题；随着院方的网上挂号等业务的开展，本方案中没有考虑到内网与公众网相连的安全措施等。

论新技术的引进

摘要

根据国家税务总局对税务系统内所有系统进行集成与整合的需求，我所在的开发单位组织了全国金税工程防伪税控系统网络版的升级开发工作。该项目工程浩大，要求在具有严格的安全、可靠性能的基础上，将基于 DOS 操作系统、Foxpro 数据库的原单机版防伪税控子系统集成到基于网络的、大型数据库的“集中存储、分布操作”的分布式系统中来，并实现与基于 AIX 等操作系统和 Oracle 数据库的稽核协查等其他应用系统的数据共享和互操作。在项目中，我担任项目主管，主要负责系统规划和组织实施工作。我在将近一年的可行性研究、需求分析、系统研发与试点工作中，通过引进面向对象设计方法、采用 B/S/S 三层体系结构、利用群集实现负载均衡等新技术，使该项目取得了圆满成功，受到了用户的一致好评。但是现在看来，由于新技术的使用，怎样实现软件开发公司对新技术的渗透、怎样开发自主知识产权的中间件等问题，需要我们在今后系统开发中做进一步探索。

正文

2003 年元月，我作为项目主管，有幸参与了全国税务系统金税工程防伪税控系统的升级开发工作。防伪税控系统主要由基于增值税发票的企业开票、企业发行、报税、认证、发票发售等五大子系统组成，系统组成模块如图 1 所示。具体流程是：企业在当地税务机关通过企业发行系统取得用于开具增值税发票的相关设备（金税卡与 IC 卡）和权限，再到发票发售系统领取增值税发票，通过企业开票系统开发票后产生发票明细和申报纳税数据存入 IC 卡中，然后、企业持 IC 卡到税务机关由报税系统进行报税、并将取得的可抵扣的发票数据通过认证系统进行认证后，经过 Internet 上传到税务机关，所有数据经由税务机关处理后，传递到其它系统进行处理。系统要求具有严格的安全、可靠性能，必须建成基于网络的，分布式实时数据库处理系统，达到数据共享的目的。在此次开发过程中，我特别强调在认真细致的需求分析的基础上，最大限度地引进新技术新方法的思想。

一、采用面向对象分析方法

防伪税控系统网络的应用将极为广泛，涉及全国所有的有税局和一般纳税人企业，具有基于增值税发票的开票、认证、报税、传递等业务操作，面对如此复杂的系统，怎样从中理出头绪来，最大限度满足用户要求，实现整个开发流程的“无缝”连接，需求分析是最重要的环节。我在认真研究和分析旧单机版开发文档后，提出采用面向对象的开发方法。系统的主要业务范围是增值税专用发票的防伪与税控，通过对整个系统流程的分析，我抽象出“发票、操作员、安全卡”等类，从类出发，建立对象，再通过对象类之间的继承、聚合关系、消息和关联，分别建立开票、认证、开票、企业发行等子系统和共用的系统管理等模块，并通过操作员类中权限属性实现各子系统的权限控制。通过面向对象分析方法，不仅提高了系统的开发效率，而且提高了软件的复用性和可维护性。

在工具的选择过程中，我们选择了现在十分流行的系统 Rational 系统系列工具，包括 Rational Rose、Rup、Soda 和 Requisite Pro 等。特别的，从防伪税控系统的对于增值税发票的防伪和税控功能可以看出，整个系统具有很长的生命周期，必须面对税务系统因为经济发展而出现的多变的需求变更，并且，又必须服务于税务系统其它的如征收管理、稽核协查等应用系统。所以，要求系统具有很好的可维护性、可扩展性。而公司在原有单机版的升级上，因为没有统一的、规范的开发文档而吃尽了苦头。所以，我决定采用 Requisite Pro 作为我们现在和未来的系统需求管理工具。我们在对经过详细的用户调查后，按照我们的基本理解写成了基本需求，交给用户进行评审和补充，再形成正式的需求录入到 Requisite Pro 中去，并记录需求的变化情况、需求之间的依赖关系，实现需求的全面管理。公司决策层在认真听我的汇报后，表示了很高的兴趣并给出支持，使我更加增强引入新技术进行开发的决心。

但是，新技术并不意味着是最好的或是最适合的。在引用新技术中，我特别注意它的负面。面向对象分析方法虽然有效地表达和描述了现实世界，但有时也会忽略外在的表层的需求，有些关键需求要等到用户使用后才会提出，然而等到用户使用后再维护是不现实的，作为原型开发模型中的原型也是收集用户需求，描述与解释需求的一类相当有效的方法工具。在分析过程中，为了更好让用户了解我们的系统，更完整提出需求，我沿用了公司熟用的原型开发方法。通过利用 Access 开发出系统原型，让用户试用，取得很好的效果。所以在新技术引进过程中，我们熟悉的优秀的开发思想与方法不能丢，这是我们要特别注意的地方。

二、采用 B/S/S 三层软件体系结构

瘦客户端设计表示层。进行系统设计时，采用客户/服务器（C/S）模式还是采用浏览器/应用服务器/数据库服务器的（B/S/S）模式，成为我们开发小组成员争论的焦点。我认为，采用哪种系统体系结构相当重要，决策时，不但要考虑系统运行成本，还要考虑系统运行的稳定性、可靠性、可维护性以及业务需求变动时的可扩展性。新系统将所有数据集中到市级税务机关（参看图 1），如果将业务处理逻辑分散到客户端或数据库服务器，虽然可以低成本运行，但是，我们分析得到，同一时刻，区县级税务机关和企业到服务器的连接可能达到 30-1000，在发达地区可能更多，这显然击中了 C/S 模式的要害，并且，税务系统的需求经常变更，所以，我们决定采用应用服务器集中处理实现业务逻辑。事实证明，这个决策是正确的。新系统在沿海五省四市试运行，不仅稳定、可靠，易维护，而且，客户端采用 IE 浏览器，受到了维护人员和广大操作者的好评。

业务层引入 WEBLOGIC 应用服务器。众所周知，WEBLOGIC 是用于开发、集成、部署和管理大型分布式 WEB 应用、网络应用和数据库应用的 Java 应用服务器。可以说，采用 Weblogic 的主要理由有：首先是税务现有应用系统的异构的数据库平台如用于（如用于 CTAIS 征管系统的 Sybase、稽核协查系统的 Oracle、其它的 SQL Server）；再有，对多种操作系统的支持，包括 NT 系列、AIX、SCO 等；并且，税务的系统需求经常变化，比如在我們的开发过程中，因为国家税务总局推出“一窗式”管理，税局要求我们的报税系统有向征管系统传出某种格式数据的功能，我们利用 WEBLOGIC 对 XML 的支持，很快地解决了这个需求。总之，这要求系统具有很好的可移植性和可伸缩性，应用服务器能使程序员从常规设计中脱离出来，集中精力组织和优化业务逻辑。我特别注意了服务器的选型，主要从开发效率、可复用性、可伸缩性和可扩展性等几个方面考虑。

三、利用工作站实现负载均衡

企业将取得的可抵扣的增值税发票经过认证系统认证后，形成加密的认证数据，经过互联网传到税局的外网 WEB 服务器，再由 WEB 服务器传到内网的应用服务器进行解密和税局再认证。对于一个市级税局，这种发票数据可能非常庞大，如果由应用服务器信中进行解密，很显然，时间上是不允许的，并且，可能造成应用服务器的死机。我提出采用解密工作站队列处理解密工作，让应用服务器统一分配各工作站的工作量。这样，大大地减少了应用服务器的工作量。

综上所述，由于采用了面向对象的系统分析方法，B/S/S 系统体系结构、解密工作站实现负载均衡等新技术，提高了整个系统的开发效率和质量，使新系统按计划完成，且具有很好的安全性、可靠性、可维护性、可扩展性和外部接口。但是，由于新技术的采用，也使我们产生了一些新的问题。

怎样让员工理解、支持和掌握新技术。引进新技术，将打破开发人员习惯的开发方法和程序设计方式，这种习惯的打破，往往要付出更多的精力和时间，如果靠权力或压力来解决只能是一种表面现象，我认为，关键是要营造一种引进新技术进行软件开发的文化氛围，让软件公司开发人员学习并引进新技术到我们的开发中来形成一种习惯，怎样营造这种氛围，是我们管理领导层常思考的问题。

怎样开发自主知识产权的中间件。我们的应用服务器采用的是 BEA 公司的 WEBLOGIC 中间件，虽然，它提高了我们新系统的开发效率和安全性和平台无关性，但是，也大大提高了我们的软件开发成本。中间件在我国具有很大的市场和利润空间，我想基于中间件的开发，首先要把握它的市场价值，然后在开发上注重它的通用性。

论软件测试方法和工具的选用

摘要

本文通过某省征管信息系统，讨论了软件测试方法和工具的选用。该项目的目标是建设一个功能完备、监控严密、安全稳定的新一代税收征管信息系统，系统业务功能多，系统复杂，性能要求高。项目采用了 J2EE 应用开发平台，经历了多个测试阶段，根据系统的特点，文章重点讨论了我们在单元测试和性能测试中所采用的测试策略和测试工具，单元测试中划分了类的优先级，使用 Junit 测试框架；性能测试中确定关键用例，使用 Rational Test Manager 2003 工具测试。

作为项目负责人之一，我有幸参与了该项目的分析和设计，也参与了测试的整个过程。

正文

税收是国民经济的重要环节。今天，在业务需求和信息技术发展的双重交互推动下，税收征管业务的信息化应用在全球以突飞猛进的速度得到应用。然而，受限于工程进度的要求、IT 技术的发展和观念认识的差距，早期的信息化建设通常缺乏全局规划。为此，2002 年 10 月，某省税局决定在原有的信息化建设成果的基础上，充分发挥目前先进的计算机网络和通信技术的强大处理能力，构建全省集中的税收征管信息系统。工程的目标是集中存储和处理全省征管业务的基础数据，实现信息的高度共享和一致，全面推进信息技术在税收工作中的应用。

作为项目负责人之一，我有幸参与了该项目的分析和设计。

该系统建立在 J2EE 框架基础之上，其 BLH (BizLogicHandler) 是 J2EE 框架中的 Domain 层的实现；BPO (Business Persistence Object) 是 DAO 与 BO 的结合体，对应 J2EE 中 Persistence 层的实现；Web 层开发采用 Struts 框架。系统业务功能多；渠道多，后台前置系统多，系统复杂；系统性能要求高，并要求系统稳定，质量要求高；采用很多新技术，项目建设时间紧。

根据本项目的这些具体特点和要求，我们在 IBM 完全生命周期测试模型的基础上，根据项目时间要求、具体情况和成本效率进行了裁减，形成本项目的测试策略、总体测试计划和详细测试计划。对于应用系统的测试，我们划分为单元测试、部件组合测试、功能测试、性能测试和验收测试。其中重点关注了单元测试和性能测试，下面分别介绍。

单元测试阶段，我们采用开发人员自己写测试代码、小组内同级审查和测试组抽查相结合的测试策略。要求单元测试应紧接在编码编译通过之后，鼓励进行测试先行（即先编写测试用例，然后用测试驱动代码的实现）。

单元测试工具采用 junit 测试框架。因为，我们的开发语言是 Java，开发工具采用的是 Jbuilder，而 junit 是当前 java 自动化单元测试的实际标准，jbuilder 对 junit 提供了很好的支持。

对 Action 部分使用 StrutsTestCase 进行单元测试，StrutsTestCase for Junit 是对标准 Junit 中 TestCase 的扩展，可以对 Struts framework 的测试提供方便。我们使用了其中的 Mock object 方法，测试 Action objects、mappings、form beans 以及 forwards declarations，它不需要 servlet 引擎及 web application container 的环境，而且 StrutsTestCase 提供了许多“validation methods”，方便测试案例的编写。我们采取的原则是，尽可能的把逻辑代码从 jsp/servlet/action 中移出，使用 Junit 作单元测试。该系统单元测试中面临两个脱离，脱离 BizDelegate（封装了对 Session Façade 的调用过程，降低 Application 层与 Services 层的耦合性）对 action 进行测试，脱离 BPO 对 BLH 进行单元测试，为此我们使用 EasyMock 技术，为一个接口创建一个模仿对象，将模仿对象作为参数来调用域代码，具体为测试者提供了抽取方法和工厂方法。

因为该项目涉及的用例非常多，我们对其进行了优先级的划分。在单元测试的时候，我们同样对类进行了优先级的划分，依据是：类对应的用例的优先级、对系统是否产生严重后果、潜在的用户数量和使用频度、信息的价值。对于不同优先级的类采用不同的测试方法。高优先级的类，一定要有相应的测试类，以黑盒测试为主，对典型错误进行一定的白盒测试的方法，每一步修改都需要进行回归测试；中优先级的类，根据实际情况决定是否进行测试类，对核心的采取运行测试，其余的通过同级审查；低优先级的类，采用静态测试，所有类通过开发人员自检和同级审查。

为了保证测试的质量，我们测试之初就设置了专门的测试小组。在单元测试阶段，该小组监控所有的测试活动和任务的执行情况，对测试的总体进行跟踪、控制和报告。对于类的提交，我们制定了严格的审核过程。首先，开发人员测试审查自己的类；然后小组内审查人员审查相应的类，打上已审查标记；最后，测试小组审核和抽查已审查的测试类和代码；测试小组还需要根据审核和抽查情况进行统计分析，调节测试资源分布。

在性能测试阶段，我们分为四个阶段来实施：启动阶段、准备阶段、实施阶段和分析阶段。测试工具采用 Rational Test Manager 2003，测试环境包括 Local computer 和 Test agent，Local computer 作为测试平台的控制主机，负责整个测试的计划、设计、实现、执行和评估，作为 Test agent 的机器，统一接受由 Local computer 发出的脚本指令信息，在一台机子上可以模拟多用户访问系统，并将执行结果报告给 Local Computer，最后由 Local computer 生成统计报告。在测试中我们也发现响应时间慢的问题，在经过对服务器的调优，以及相应部分的代码优化、SQL 优化之后，性能得到明显改善。

下面简单介绍性能测试中我们对遇到的问题所采取的策略：

（1）该系统采用的是 J2EE 架构的一种模式，GUI 客户端直接和服务器连接，采用的是 BEA 公司独有的 T3 协议，而目前自动化测试工具能够录制和回放脚本的大都是基于 HTTP 协议的浏览器客户端方式。对此，我们采取自动录制和手工编写脚本相结合的方式，对于浏览器客户端的测试，采用自动测试工具录制脚本；对于 GUI 客户端的测试，用 JAVA 配以性能测试工具提供的 API 包，手动或半手动编写测试脚本。

（2）该系统业务功能繁多，测试需要准备的数据量大，而测试时间短。我们分析出业务具有代表性的重要和关键用例，并且利用开发过程已有的客户端程序，减少测试脚本的开发量。

（3）该系统渠道多，与外部系统接口复杂，而且系统采用多家公司产品，如果出现问题分析和定位困难。对此，我们利用性能测试检验客户和系统之间的交互，包括浏览器和 GUI 客户端等方式的连接。同时在进行性能测试的时候，将内部各种系统，与其连接的各外部系统的日志和监控工具全部打开，记录各部分的处理过程，这样当发现性能问题时，便能及早的定位瓶颈出现的位置；测试环境准备和测试时，请相关厂家的工程师提供现场支持，进行性能监控和问题分析。

由于采用了适当的测试方法、测试策略和测试工具，总体来看，我们的测试取得了不错的效果，有力地保证了项目的质量。但也有不足的地方，具体存在以下几个方面：

（1）开发人员的测试观念还不够强。虽然我们制定了良好的单元测试策略，但开发人员并没有很好地执行，以至于在以后阶段的测试和运行中受害不浅。

（2）每种测试之前，我们都组织力量充分准备了测试案例，但是在测试数据的准备上，由于系统的复杂性等多方原因，有些数据准备的不太完备。

要解决以上问题，我认为首先还是要树立开发人员的测试理念，只有从具体的开发人员做起，才能真正提高测试的质量，其次还要坚决贯彻执行项目中确立的测试方法和策略。

我们从实践中领会到，测试确实可以在保证软件质量方面起到很大的作用。但同时我们也认识到，测试中还有很多领域和知识点需要继续研究和实践，新技术的发展对测试也提出了新的要求和挑战，我们将在测试领域不断探索，不断实践。

论 ERP 系统测试方法和工具的选用

摘要

本文讨论了某报业的 ERP 系统项目的测试方法和工具的选用。

该系统是为了引进先进的管理理念，全面提升企业的管理水平为目标而开发实施的。系统包括企业财务会计、管理会计、生产管理、采购、库存、发行及广告管理等功能，全面实现报业经营管理的集成。我们的测试包括单元测试、集成测试、确认测试和系统测试，根据不同阶段分别进行。在本文中首先讨论了采用的测试方案，分析了如何使用工具来完成各种测试以及产生的效果，然后详细描述了权限测试。最后，讨论了存在的不足之处以及改进的办法。

在本项目的实施开发过程中，我担任了系统的蓝图设计、系统配置和测试工作。

正文

作为一家大型的媒体行业，某报业的各项经营业务飞速发展，但同时也存在经营业务缺乏一个统一、集成的平台，各项业务信息难以集成的状况，同时原有的管理系统已无法满足线的是的需要，不利于领导的决策，制约了企业的发展。如果不建立与企业规模相适应的管理信息系统，就很难在这个竞争日趋激烈的时代获胜。因此，为了引进先进的管理理念，全面提升企业的管理水平，进一步提高竞争力，决定实施 ERP 系统项目。

经过较长时间的调研和论证，我们决定引进 SAP 公司的 R/3 系统。报业不同于传统的制造业，有特定的经营方式，有自己独特的业务、客户、供应商；有独特的企业文化、政策规范和经营理念，还有一些重要的成功经验。这些都是无法与线涉及到 ERP 软件中去的。因此，在引进 R/3 时，我们考虑将企业的“个性”与 ERP 软件的“共性”结合起来，走二次开发的道路。我们请专业顾问公司协助完成实施，二次开发基本上由自己的队伍来完成。

SAP 的实施项目是个复杂的系统工程，因此我们采用三个阶段逐步实施的方法。第一阶段以财务管理为主，并与其它系统进行整合，实现 SAP 系统与业务运作系统无缝的集成。系统涵盖企业的财务会计、管理会计、生产管理、采购、库存、发行及广告等部门的业务管理。我有幸参与了该系统的开发与实施，担任蓝图设计、系统配置和测试工作。

ERP 系统相当复杂灵活，因此在系统正式上线前做好测试，选择合适的测试工具，尽可能多的发现各种错误和缺陷，对系统的正常运行至关重要。在项目准备阶段，我们制定了详细的测试计划，确定测试的进度安排、使用的工具以及这些测试的内容和目的。对标准系统的测试和二次开发部分方案有所不同。

我们的测试包括单元测试、集成测试、确认测试和系统测试，根据不同阶段分别进行。在测试前做好各种测试用例。用例清单中包括：交易代码（在 R/3 系统中，每个功能分配一个交易代码，用户输入后进入界面）、步骤说明、测试数据（输入数据）、期望结果、实际结果和测试状态。

根据系统特点，我们建立了生产系统和测试开发系统，所有设置、开发和测试（除系统测试）都在测试开发系统进行，确定没问题再通过传输系统传入生产系统。两个系统具有相同的环境配置。

在测试开发系统中我们建立了三个“client”（是 SAP 系统中的最高等级），分别为 100、200、300，测试流程是：在 100 中配置并测试业务场景和案例；在 200 中配置，并生成传输请求传到 300，在 300 中测试业务场景和案例。

我们在集成测试和系统测试中的压力测试中采用了 SAP ABAP/4 开发工作台提供的计算机辅助测试工具（CATT）。CATT 具有良好的集成功能和报表功能，使用 CATT 可以减少人工测试的次数，由系统自动完成测试，并使测试更加系统化（即事先定义测试过程中的输入值以及预期的测试结果），从而提高系统的质量。所有测试都有日志，根据所使用的测试模块和测试过程的不同，日志中记录按层次结构组织起来，显示成为带节点的结构。日志可以从客户端下载保存，便于以后查阅。

在集成测试中，我们在 CATT 中根据实际需要，编制测试过程，测试过程包括任意数量的测试模块。例如为了测试一个发行过程，我们建立了一个测试过程，包括创建客户、创建物料（报纸）、报数、生成标准销售订单、发货和开票过账几个测试模块。定义好参数和变量。其中客户和物料数据在系统外根据测试用例建立文本文件。测试时系统读取文本文件中的数据，每一步生成的数据作为下一模块的输入数据。最后系统生成日志，记录了详细结果。用此结果与预期的结果比较，查找问题。

在压力测试中，我们在文本文件中装入大量数据让系统处理，查看其性能下降的幅度是否在允许的范围内。

另外，我们在测试中比较有特点的是权限测试。ERP 系统是一个高度集成的系统，从生产、销售到财务，所有的处理和数据都集中管理，因此合理的授权管理是系统安全性实施的一个重要部分，保证合法用户拥有需要的权限，避免用户使用不应该拥有的权限使用系统的功能或查看数据。系统采用授权参数文件生成器来进行授权管理，授权参数文件将针对交易代码及组织结构两个层次。我们对生产系统中的所有用户进行授权，对所有模块及功能进行授权。上线前所有用户参数文件将传至生产系统。

我们制定了有效的权限测试策略，每个用户帐号在授权的组织结构中测试能否正常使用授权的交易代码对应的功能；对于重要的交易和数据着重进行是否有错误授权的测试，主要分两种：对同一交易，不同组织权限不同，所以特别测试跨组织结构的权限，防止某组织的用户访问其他组织的数据；有些交易，如某些订单的创建，操作人员可创建订单，但必须由领导确认释放，特别测试确认释放的权限是否正确授权。对公司的重要统计数据，只能领导有查看分析的权限，各部门只能看到本部门的数据，所以要特别测试这些部门的用户是否误分配了权限。

我们利用 CATT 管理工具中的统计功能查看状态分析报表，以及图形化的测试历史记录。并将所有测试日志下载到文件中保存下来，特别是有问题的地方，详细登记问题出处、问题原因和解决方法等。最后根据测试的各种资料完成测试报告，对测试进行概要说明，列出测试的结论，并作为软件配置保存。

通过各阶段的测试，发现了系统存在的缺陷、局限性，对比较严重的缺陷及时作了修改，对系统影响不大的缺陷作了修改安排。保证了系统正式上线后的稳定运行。另外，对系统性能影响的局限在测试报告中作了详细说明。例如，我们通过压力测试，发现当销售订单很多时，一些统计报表的查询性能会下降，而根据实际业务的情况，在 3 年的时间订单数会接近测试时的数。因此，我们建议系统使用到一定阶段，应该做归档处理，这为之后的维护人员对系统的维护提供了很大的帮助。

当然，还存在需改进的地方。比如我们在确认测试时，用户用日常工作中出现的一些特殊数据来测试，发现了功能上一个小问题。虽然影响不大，但也说明我们在单元测试和集成测试中选择的用例还不够，还应该更接近实际，选择一些更真实案例来测试。另外，我们只用到 CATT 的部分功能，如果更好的利用它，将会使测试更加便利有效。因此，在以后的项目中要特别注意充分利用工具来提高工作效率。

论嵌入式实时软件测试方法和工具的选用

摘要

本文以航天嵌入式实时软件测试的过程，讨论了项目开发中测试所使用的方法和工具。在该项目中，我担任测试经理工作。

嵌入式实时软件因其特殊性，在测试时有一定的困难。目前国内和军队已就软件产品的质量要求及其测试制定了多项标准，但缺乏符合国内标准的本土化的测试工具，因而在工具的选取上，主要考虑的是符合国际上航天或军用行业标准的测试工具，及软件测试工具增强软件测试的自动化程度，和对文档支持的自动化程度。

在编码过程中，我们选用 PC-Lint 作为代码静态分析工具，辅助编译器进行代码审查。在进行单元和集成测试时，我们使用了 LDRA 公司的 Testbed/TBrun 工具。

我认为，提高测试技术并不单指使用了测试工具进行测试，测试工具的使用可以提高效率，但过多的强调工具的使用反而会使项目陷入困境。

正文

众所周知，在航天领域安全是最关键的问题，软件的微小瑕疵就可能造成天文数字的巨额财产损失，甚至对国家安全造成严重威胁。这就使保证软件的质量，保证软件的高度可靠性，面临巨大的挑战。

我们单位是航天科技集团某院下属的一个研究所的软件中心，主要从事航天器上的嵌入式实时软件的研制。我们的软件是典型的嵌入式强实时软件，基于桌面 PC 为宿主机，x86 系列的 386EX 为目标机。软件计算量大，任务间时序要求严格，逻辑条件控制复杂，属于大型嵌入式软件。随着软件工程思想的引入，CMM 和 RUP 模型的推行，测试技术和测试方法也逐渐被重视起来。近年我一直从事软件测试的工作，在项目中担任测试经理角色一职，主要负责测试计划的制定、编写测试用例和组织软件测试。

当前我们大部分的测试还处于半手工的方式，没有形成一套自动化系统的软件测试标准和规范。在软件审查中测试用例简单，不全面。为了尽可能多地找出程序中的错误，生产出高质量的软件产品，我把测试工作重点放在加强对测试的组织和管理上。

按照软件工程的思想，中间过程产品的高质量，才能导致最终产品的高质量，软件开发过程中越到后期发现的错误，需要修复而付出的代价就越高。为了确保软件的质量，较理想的做法是对软件的开发过程，按软件工程各阶段形成的结果，分别进行严格的测试。保证每个开发过程以最小的缺陷进入下一阶段的开发。为此我们为测试活动的定义了以下的过程：制定测试计划、设计用例、实施测试、执行测试、评估测试。并使这些过程协同作用、互相促进，主要目标是在设定的条件限制下，尽可能发现和排除软件缺陷。

在组织软件测试上，我们定义了以下的规程。

- 1、开发人员负责自己开发模块的代码静态分析；
- 2、开发人员组成小组，相互之间进行代码检视；
- 3、开发人员负责自己开发模块的单元测试；
- 4、开发部内部集成测试；
- 5、独立的测试部门进行系统集成测试；
- 6、系统测试，较全面的对最终目标系统进行功能验证，做尽可能多地覆盖；

在具体测试过程中，我们分阶段分步骤地采用了一些先进的工具和方法策略。为我们测试成功奠定了基础。

目前国内和军队已就软件产品的质量要求及其测试制定了多项标准，但缺乏符合国内标准的本土化的测试工具，因而在工具的选取上，主要考虑的是符合国际上航天或军用行业标准的测试工具，及软件测试工具增强软件测试的自动化程度，和对文档支持的自动化程度。在编码过程中，我们选用 PC-Lint 作为代码静态分析工具，辅助编译器进行代码审查。通过 PC-Lint 我们能发现一般的语法错误，检查出那些通过了编译，虽然符合语法要求但不易发现的潜在错误。如没有被适当检验的数组下标，未初始化的变量，使用空指针，冗余的代码等等。PC-Lint 可以添加用户自定义的检查规则，用来适应新的编码规范。此外还支持命令行和 MAKEFILE 方式，可以集成到 UltraEdit 和 Tornado2 中开发环境中，或固化到软件开发测试流程中去，方便开发人员的使用。功能强大，易于使用，性价比高是我们选择 PC-Lint 的主要原因。我们发现利用 PC-Lint 进行代码静态分析，在程序动态测试之前发现编码错误，可以防止低级错误引入到后期的开发中。通过实践证明这样可以有效的降低软件除错的成本。

在进行单元和集成测试时，我们使用了 LDRA 公司的 Testbed/TBrun 工具。我们测试的是航天器的控制软件，语句逻辑控制结构复杂，进行白盒测试时，测试的输入量大，是一件十分枯燥且重复性的工作。通过 TBrun 提供的灵活的测试数据输入方法，测试数据可按范围、最大、最小及等步长或不等步长方式输入，实现了自动化的环境建立和测试输入。TBrun 还提供了可视化管理用例的方法，我们发现这些方法极大的提高了软件测试人员的积极性，使测试效率有了大幅度提高。

在测试中我们会常常遇到这样一个问题：在测试初期，一般主要针对功能点语句设计用例，以检查所期望的功能是否已经实现，这时覆盖率迅速增加，但一般达到 70%左右的覆盖率后，要再提高覆盖率是十分困难的，因为新的测试往往覆盖了相同的测试路径。这时需要对测试策略做一些改变：从功能性测试转向结构化测试。也就是说，针对没有执行过的路径，构造适当的用例来覆盖。Testbed 有效的解决了这个问题，Testbed 通过提供调用图与控制流程图，显示被测系统的调用关系及每个子程序的控制流程，在测试阶段，实时显示测试覆盖率，提供检查分析关键因素和优化测试路径的必要信息，帮助用户分析未测试的代码，实现以最少测试用例即可达到覆盖率要求。

Testbed 支持自动隔离测试，可自动产生驱动和桩模块。在我们按组合集成策略进行渐增式集成测试时，在程序结构的高层使用自顶向下的策略，在下面的较低层试用自底向上的策略时，自动产生桩模块和驱动模块，让我们把主要精力放在了测试用例的设计上。此外，被测代码修改后对测试用例自动验证，使得回归测试非常的轻松。实践证明了好的测试工具对于激励测试人员的积极性和创造性，提高测试效率是非常有效的。

嵌入式实时系统的性能测试，我们使用了 CodeTest。CodeTest 在真实软硬件环境下使用非采样性测试，可以精确计算出每个函数或任务的执行时间或间隔，所以其测试具有很高的可靠性。我们仿真实时系统并按照外部事件的序列检查其行为，使用类似于等价类划分的技术，对事件如中断，控制信号和数据分类测试，并在测试了每种事件后，以随机顺序和概率将事件传给系统，检查系统行为是否有行为错误。实现了以往难以进行的系统行为测试。在隔离了任务内部和系统行为错误后，我们用不同的数据率和处理负载来测试系统中通信的同步和互斥任务，确定任务时序是否正确，根据 CodeTest 测出的时间作为对应用程序优化的依据，使开发人员可以有针对性地优化某些关键性地模块或任务，以及改善整个软件的总体性能。

总之，软件测试方法和工具的使用，可以提高测试技术水平，对软件质量保证起很大的作用。但是提高测试技术并不单指使用了测试工具进行测试，测试工具的使用可以提高效率，但过多的强调工具的使用反而会使项目陷入困境。通过这次软件测试，对我启示很大。觉得目前我们的测试活动中存在以下的不足和可改进措施。

让软件测试走向规范，应用过程方法和系统方法来建立软件测试管理体系。其主要目标是让测试活动中的过程协同作用、互相促进，从而使它们的总体作用大于各过程作用之和。缺乏对测试工作进行度量，对测试效果进行评估，原因是缺少一个有效的度量数据收集和分

论分布式数据库的设计与实现

摘要

分布式数据库系统把应用所需的数据存放在多个数据库服务器上，完成某个数据操作要涉及到访问多个服务器，这适用于某种特定需要的应用。我在主持设计开发的一个 MIS 系统中，为了达到了在低速网络通道下有效提高应用程序性能的目的，使用了 Sybase 的分布式数据库技术。我设计的这个系统是采用典型的 C/S 结构，但许多客户端连接服务器的网络采用电话线拨号，速度有限，传统 Windows 界面的客户端应用程序相应速度比较慢。考虑到 B/S 结构也避免不了大量数据从服务器端传输到客户端，我认为 WEB 界面并不能有效解决这个问题，所以采用了优化数据库结构的方法，把数据分两部分存放，基础数据放客户机，会员资料主要采用键码放服务器，应用程序再现数据时从服务器取键码，到客户机取对应的解释，由于键码的数据量少，网络传输便快。在构建这个分布式数据库系统的过程中，我着重研究并解决了数据同步和事务协调的问题，取得了良好的应用效果。我认为，分布式数据库系统的技术在 Internet 时代正当其道，大有发展前景。

正文

分布式数据库系统把数据存放在多个数据库服务器上，当应用提取所需数据时，要访问多个服务器，综合多点数据才能完成。

分布式数据库技术在很多场合得到了应用。譬如某企业随着业务量的扩大，原有数据库服务器已经达到了容量和性能极限，如果不希望丢弃原有投资，可以建立另外一套新的数据库，跟原有的系统组成一个分布式数据库系统，给应用提供透明统一的数据访问。还有，如果某企业分成多个业务部门，而且地域分散，可以在某个部门放置单独的数据库服务器，用于存放该部门最常用的数据，而部门和部门之间相互引用的数据可以通过分布式数据库技术来方便地完成。分布式数据库不是简单地把集中数据库分散实现，而是针对某种特定应用需要而诞生，它必然具有自己特有的性质和特征，需要上面做许多的工作，来满足应用的要求。

我在设计、开发一个 MIS 系统时，针对应用的需要而引入分布式数据库技术，取得了良好的效果。

该系统针对会员资料的管理而设计，用于管理会员入会、缴纳会费、申请资助、办理资助审批、关系转移、退会和注销手续等业务流程。分三个级别的应用权限——基层单位级、总公司级和集团公司级，各个级别只能操作各自范围内的业务数据。该系统采用典型的 C/S 结构，后台数据库采用 Sybase，前端应用采用 PB 开发工具来设计标准的 Windows 操作界面。我在其中任系统分析和数据库设计的角色，担任了调查业务需求、业务建模和数据库建模、数据库设计以及指导应用程序测试、优化系统和应用的性能等等一系列工作。

由于客户端地域的分散，遍及多个省境内，许多使用该系统的基层单位连接服务器数据库的网络采用电话线拨号方式，速度有限，在使用客户端应用程序时感觉界面速度很慢。经过分析，认识到许多操作都要从服务器中取数据，速度慢就慢在数据访问上。服务器是没有性能瓶颈的，问题出在网络速度上。不可能要求众多使用客户改善和升级他们的网络，只能充分挖掘软件的潜力，来适应这种低速网络的使用模式。

经探讨，结合关系数据库的知识，认识到，应用程序的每一次数据库操作，都要访问多个相关联的表，其中，有会员资料表和基础数据表，会员资料表中存放许多的键码值，在基础数据表中有键码相应的解释。键码值的数据量比较少，而基础数据是静态的，几乎不会更改。如果考虑把会员资料放在服务器上，基础数据放在客户端，当应用程序中访问数据时，总是从服务器上存取会员资料，从客户端提取会员资料中键码的相应解释。由于键码的数据量少，便减少了网络上传递的数据量，从而提高了界面的响应速度。

同时考虑到基层单位总是操作自己所属的部分会员，增删转移操作少，会员列表比较固定，而每一项业务操作都涉及到要从会员列表中查找定位到某个会员，所以会员列表是最常访问的数据项。把会员列表从会员资料数据中抽取出来，也放置在客户端，这样，便进一步改善了性能。

把数据分散存放只是工作的第一步，接下来要考虑应用程序怎样访问这种分布式数据。开发应用时，如果每一功能都针对两个数据库进行，就带来了很多麻烦。所以，我们研究了 Sybase 的分布式数据库技术，决定采用了 CIS（组件集成服务）部件，来合并两个数据库成一个统一的分布式数据库。应用程序只要连接一个数据库，就可以透明统一访问到两个数据库中的数据。

该技术具体实施方法是，在客户端数据库中建立一个对服务器数据库的远程访问服务名，包含访问地址、登录用户名、登录密码等等关键的连接信息；并且对服务器中会员资料数据表建立一个本地代理表，结构和服务器中远程表完全一样，它是访问服务器中会员资料的中转和代理。客户端应用程序访问本地代理会员资料表时，实际上是通过预定义的远程访问服务名中包含的连接信息到服务器中对应的实际会员资料表中访问数据。这种访问对于客户端完全透明，感觉不到是从物理上独立的两个服务器中存取数据。所以，这种数据库结构是典型的分布式数据库。

部署这种分布式数据库不是难事，只要在客户端和服务器上安装 12.0 版以上的数据库服务器，在客户端服务器上建立远程服务名和代理表即可。由于 Sybase 数据库的安装支持脚本方式，在客户端应用程序的标准安装过程中，嵌入 Sybase 数据库的安装和配置脚本，就自动化地完成了所有工作。

在实际使用该分布式数据库系统的过程中，遇到了几个问题。第一，数据同步。客户端基础数据不是绝对静态的，也有变化，因此在服务器端要设置一个统一的基准，称为主点数据。客户端总是复制使用，称为复制点数据。如何及时感知到服务器端主点数据的变化，有效率地复制到客户端，是个难题。Sybase 针对这种应用场合，提供了复制服务器技术，但为了避免过于复杂，我们实际采用应用程序来管理同步。当服务器端主点数据有了更改时，保存一个相应的标识和时间戳，客户端应用在登录服务器时，检查这种标识，一检测到了数据有更新，就首先下载，然后再进入系统正常使用。这种方法实现起来，增加了额外的开发量，且不能判别绕过应用程序对数据的直接修改，但是，是最简单和有效的方法。

第二个问题是事务协调问题。物理上独立的两个数据库，在协同操作时，如果服务器正好停机或者网络故障，完整的一个事务没能完成，就会“事务崩溃”。虽然 Sybase CIS 内嵌了两阶段提交技术，能够自动恢复。但是应用程序在这种情况下，敏感性不够，操作界面会无端凝固，影响了使用的方便性。我们针对 PB 对于连接的判断和感知，用了一个小小编程技巧，使应用程序能够及时感知到数据库连接故障，及时停止和恢复事务，使操作界面表现友好灵活。

以上遇到的这些问题，都找到了解决办法。分布式数据库技术的应用并不是非常复杂，它往往为解决特定问题、满足特定需要而被采纳，使用得当，会给应用带来了许多便捷。

在当今信息社会里，互联网络带来了相互连通的便捷，而且知识爆炸，数据的分布式访问是个必然趋势。潮流兴起的 XML 技术，提供了各种平台数据库之间的一个公共数据访问标准，可能会用来构建更加灵活、适应性更强的分布式数据库技术。

论分布式数据库的设计和实现

摘要

本文论述《金蚕工程》的分布式数据库的设计和实现。该项目的设计目标是实现企业间茧、丝等的合同交易（交易规则和期货交易一样）、实时行情和成交数据的发布、茧丝质押和质押数据的发布。所有功能均要求既能在企业局域网交易大厅和 Internet 上进行，许多功能又要在苏州和成都分中心进行。系统从设计时就基于分布式数据库应用的可用性和可靠性作为系统一个关键目标。为了达到系统的上述要求，系统分别从数据库设计、应用数据集成和测试以及分布式数据库部署等做了大量工作。针对上述各部分，本文论述了分布式数据库的设计和实现及遇到过的典型问题和措施，最后对系统改进，谈一些自己的体会。

在项目开展期间，我担任了系统分析、系统设计与关键模块的编程等大量工作。该项目在 2002 年通过了浙江省软件评测中心的评测。

正文

我于 2000 年底到 2002 年 6 月组织了中国茧丝绸市场由国家经贸部资助的《金蚕工程》一、二期项目的开发。由于原有系统存在如下问题：（一）原有系统采用传统 C/S 结构，客户端运行在 DOS 平台，后台使用 Foxpro 数据库。系统处理慢，前后台数据一致性，并且 Foxpro 服务器在交易高峰经常莫名其妙的死锁和 Down 机。（二）市场是中国最大的茧丝交易市场，由于会员单位不断增加，原有远程电话拨入设备严重不够用，系统又不能通过 Internet 连接访问，进行合同交易和行情的及时查询，严重限制了市场的发展。（三）原有系统只部分实现了合同交易和资金清算，功能和企业发展不相适应。（四）由于会员在地区上分布不均匀，为了更好地服务于会员要求在满足一定条件下建立分中心交易系统。在这种情况下，市场决定开发基于 Internet B/S 结构和基于 Xnetserver 通讯中间件（类似 IBM MQSeries 的通讯中间件）的 4-Tier 结构的综合软件，分中心采用基于分布式数据库地应用系统。一期项目包括：基于 4-Tier 的合同交易系统和资金清算系统，基于 Internet 的行情查询和信息发布的企业网站。二期项目包括：茧丝质押、仓库管理系统、Internet 质押查询系统和分中心交易和资金结算系统。

鉴于该项目业务比较复杂，流程比较多，系统要兼顾企业交易大厅会员、远程 Internet 交易的会员和分交易中心会员，项目完成时间短等特点，为了既要项目的按期投产又要实现基于分布式数据库的分中心交易和资金结算系统，我从数据库选型、数据库设计、应用数据集成和测试以及分布式数据库部署如下工作。

（1）数据库选型。现在的主流数据库一般都可以按分布式进行部署，例如：SQL Server、Oracle、Informix 和 DB2 等。由于一期项目的数据库平台是选择了 Oracle 7.0，经过了近一年的运行，其数据自动备份、多用户并发处理性能、磁盘表空间管理、Web 数据发布等表现良好。我公司又由于开发过基于该平台的分布式应用系统，公司对其分布式实施也有相当的经验，所以经我们开发部相关人员的讨论，我决定在各分交易中心继续采用 Oracle 数据库来实现，其分布功能主要基于 S Q L * N E T 来完成。

（2）分布式数据库设计。由于各交易品种的每节交易价格由市场的交易大厅决定，所以各分中心必须在每节交易开始前取得相关的交易数据。各分交易中心必须保留其会员已经交易成功的买入和卖出交易数据，并根据会员需要进行资金结算。经对分交易中心充分业务需求分析后，我觉得分交易中心的主要功能还是和主市场交易中心（简称：主中心）功能是一致的，只不过分交易中心的会员数比较少和无需保留各节的历史交易行情，所以，分中心的主要数据库中的各表还是照搬主市场数据库的各表，分中心只保留各自的会员基础数据和成交数据，保留会员历史交易数据，只保留当前的各节的交易行情，不保留历史行情。主中心必须保留各分中心的各项明细数据，主中心也必须及时计算分中心会员的汇总资金，所以主中心必须增加分中心汇总资金表。

（3）数据集成。为了保证分中心及时下单，必须保证每节的交易行情要及时的传送到各交易分中心。节处理最后一步必须把下一节的交易行情传送给各交易分中心。同时下发给分中心的数据还有，分中心本节各会员的成交数据。由于主中心保留各分中心的各项数据，所以，分中心有新的会员加入必须同时更新分中心和主中心数据库；分中心会员资金有变动时也必须同时更新主数据库。由于各会员只能在当日交易结束后才能进行数据平仓，所以分中心平仓数据汇总后同意发送到主中心进行平仓。在上述设计的数据库到数据库操作的数据完整性和一致性由数据库的二阶段提交（2 Phase Commit）来保证。由于 Oracle 是一个比较成熟的数据库平台，又由于每次事务都不大，所以运行到现在也没有发现数据不一致现象，没有出现“单边帐”，数据处理也比较快。为了进一步保障各中心数据的一致性，我还做了一个同步程序，由主中心发起，对主中心和分中心数据进行比较。该程序可定期进行。

（4）测试。由于一期项目已经运行了相当长的一段时间，系统已经相当稳定了。由鉴于 2 Phase Commit 出色表现，所以基于分布式的测试没有觉得和集中数据库有太大的区别。唯一的感受是在第一次建立远程数据库连接时，连接速度比较慢（有时超过 20 秒）。为了提高测试速度，我在同一个局域网中安装了三套 Oracle，分别充当主中心和两个分中心进行测试。

（5）部署分布式数据库。和集中时式的数据库不同，采用分布式系统，必须配置 S Q L * N E T。其他的 Oracle 组件和子系统安全同本地安装数据库一样。

分布式数据库在逻辑上属于同一系统，使得应用不必关心远程数据库的物理位置，就可以像访问本地数据库一样访问远程数据库。采用分布式数据库可充分提高系统的处理能力、均衡网络负载，所以该分布式数据库应用方案在银行代收、代付业务处理、保险代理点等单位 and 机构中大量采用。但由于近些年来通信基础建设比较迅速，特别是 2M、10M 和 100M 光纤的推广（100M 广电月租费仅为 800 元），又为了节省下属网点的机房费用，各单位又不约而同的采用的集中方案。

【缺点和需要改进的地方】

（1）分中心或主中心由于不明原因而 Down 机后，由于 Oracle 有自动恢复功能，可能造成对方数据库会收到过时的垃圾数据。该问题现在通过手工方式解决。

（2）由于分中心没有收到下一节的交易行情，可能会造成分中心会延时交易。分中心的主控系统可通过主动拉取数据来解决。

【结束语】

《金蚕工程》项目是成功的，年交易量已经有 100 多亿人民币。通过该项目的顺利投入运行，中国茧丝绸市场不仅牢牢把握住了中国的茧丝行情，随着我国进入 WTO，中国茧丝绸市场在世界上也有举足轻重的分量！

总之，中国的软件产业应该走自己的路，只有通过“已过程为核心，以度量为基础，已人为本”的管理政策，根据我国的国情、软件业和公司的现状，只有通过“干”，才能有实绩，才能实现“以信息化带动工业化的大思路”！

论分布式数据库的设计与实现

摘要

本文通过 xxx 高速公路收费系统（以下简称收费系统），来论述分布式数据库的设计与实现。收费系统是我公司近年来接的较为大型的项目，管理结构为三层结构：公司级、收费中心级、收费站级，各级之间即可独立的完成自身业务，又有自上而下的管理关系。收费中心、收费站均为三层 C/S 结构，公司级采取 B/S 结构。该系统的数据库也按照三层来设计，收费站存放本站的所有流水数据，收费中心存放所有数据，公司本部存放查询用汇总数据，收费站与收费中心使用事务复制来同步数据，而收费中心与公司本部使用快照复制来同步数据，并且使用分级的方法来测试收费站、收费中心与公司本部之间的数据同步。

在本项目的开发过程中，我担任了数据库的设计工作。

正文

2000 年 10 月—2001 年 12 月我公司开发了高速公路收费系统（以下简称收费系统），收费系统项目从管理层面分为三层结构：公司级、收费中心级和收费站级。

公司本部：供领导、运营部、财务部等业务部门了解业务情况、检查工作。B/S 结构各部门通过 Web 服务器查询数据库服务器，而公司数据库服务器定时要求中心数据库服务器复制汇总数据。

收费中心：收费系统的管理中心下达管理制度，管理数据到收费站，接收统计收费站的收费数据，上报汇总数据到公司，负责日常的管理工作。

收费站：具体进行收费的单位，收费车道的数据通过通信系统实时上传到收费站数据库保存、分类、汇总，并且实时传送收费中心下达的数据库管理，并通过通信子系统下载到车道收费机上具体实施。

系统采用三层 C/S 与 B/S 的混合结构，收费中心与收费站为三层 C/S 结构，而公司级为 B/S 结构。我在项目中担任了数据库的设计工作，负责数据库的设计、测试及实施。

数据库设计

此收费系统的结构较为复杂，分为公司级、收费中心、收费站三级管理结构，既可独立工作，又有管理的联系。数据实时传送到收费站数据库服务器，再实时传送到收费中心数据库服务器。在数据库设计方面我们按物理的分布也分为三层结构。

在收费站，根据系统的需求分析的结果，一辆车通过收费站时产生的最基本的数据有：通过日期、时间、车型、收费类型及收费金额，因为收费标准不轻易改变，考虑到我们采用的是专用的车道收费机，存储量较小，所以收费金额项在此处不计入数据库，上传至收费站数据库服务器后，可以用车数乘以此类车型的收费标准而得到。当车道收费机上传数据到数据库时，还要加上工班、车道及收费员信息，保证数据的唯一性，所以我们把日期、时间、工班、车道、收费员、收费类型、车型设为组合主键，为车辆流水数据。收费员下班后还要上缴实收金额，因此还要保存实收金额，包括日期、工班、收费员及实际收费金额，为工班收费数据。在此基础上，分析、汇总数据，得到以下几类数据：

业务类型数据：车辆流水数据、工班收费数据、车道开通情况、收费员上班情况；

扩展的数据：为了查询、打印的方便、高效，流水表经过分类汇总，产生了以车道分类统计的车流量表（日、月）、以收费员来分类统计的收费金额表（日、月）及不收费车辆统计表；

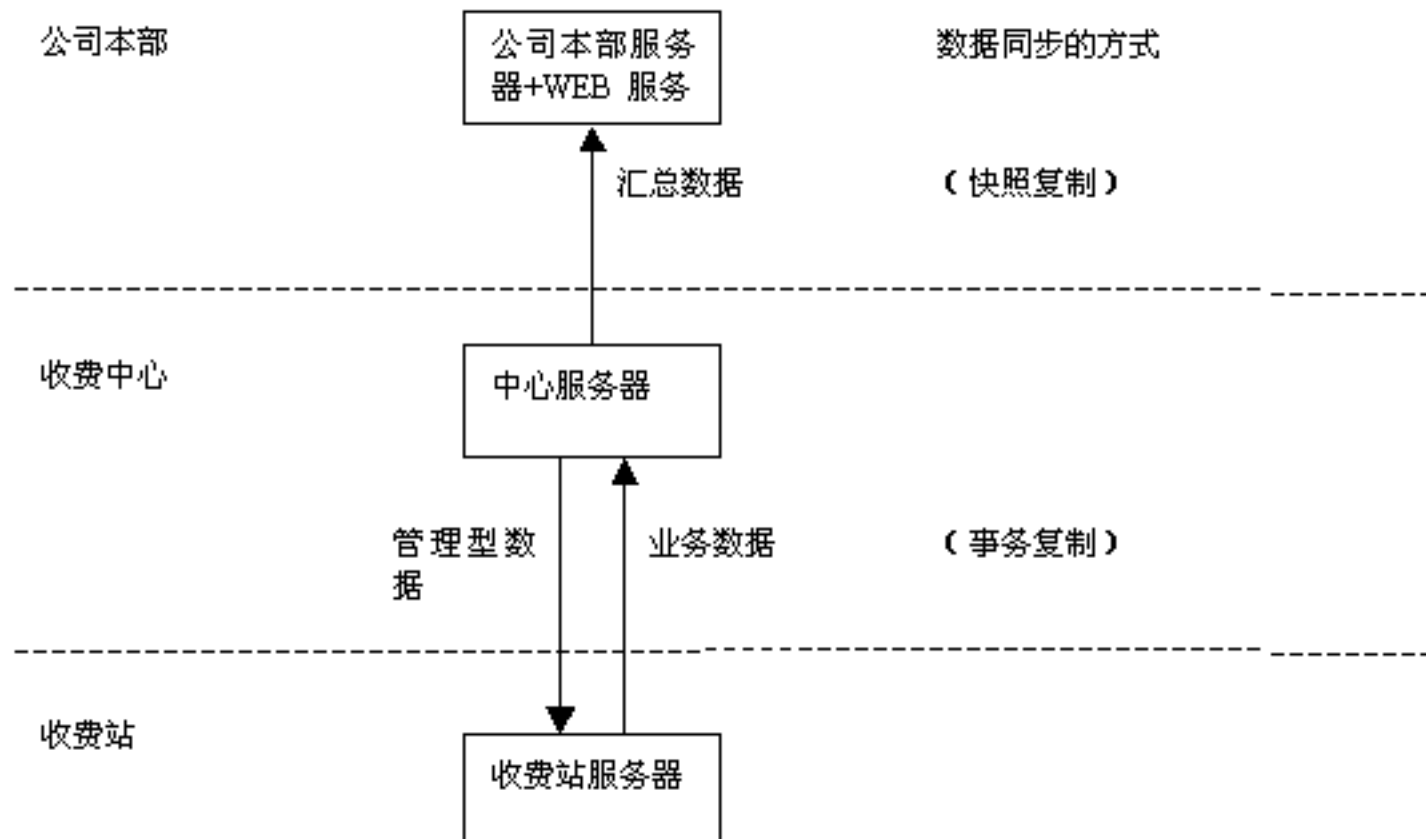
管理类型数据：收费标准、收费员信息、收费站信息、车道信息、工班信息、收费类型信息；从全局应用的角度出发，各收费站存放本站的数据，收费中心的数据库则存放所有数据，并对数据进行完整性和一致性的检查，这种做法虽然有一定的数据冗余，但在不同场地存储同一数据的多个副本，能提高系统的可靠性、可用性，使系统易于扩充，也提高了局部应用的效率，减少了通讯代价，同时也使得各处理机之间的相互干扰降到最低。

公司总部存放所有收费站的汇总数据，通过浏览器进行查询。

数据的分布

（1）在收费中心数据库服务器与收费站数据库服务器的数据关系中，由于收费站的数据是收费中心数据的子集，我们采用了水平分片的方式，通过并运算实现关系的重构。

（2）在收费中心数据库服务器与公司总部数据库服务器的数据关系中，数据是按照其应用功能来划分的，所以我们采用了垂直分片的方式。



数据分布在多个地点，为了保证数据的一致性及完整性，我使用了事务复制和快照复制两种数据同步的方式，在收费中心与收费站之间使用事务复制，而在收费中心和公司总部之间使用快照复制。

对于业务类型的数据，收费站在本地存放收费车辆的实时数据，而用户也要求收费中心也要实时的收费车辆数据，延迟不超过 2 秒，所以我采用事务复制进行业务数据的同步，收费站只需将更新的数据发送到收费中心的数据库即可。具体过程如下，把收费站的数据库作为出版者和分发者，收费中心的数据库作为订阅者，对收费站的数据建立快照代理，并在分发数据库中记录同步状态的信息。每一个使用事务复制的收费站数据库均有自己的日志读取代理，运行在分发者上并连接出版者。分发代理的任务是将分发数据库中保持的事务任务直接推动到订阅者。当推订阅被创建时，每个为立即同步而建立的事务出版物通过自己的分布代理运行在分发者上并与订阅者相连。

而管理型的数据是在收费中心设置，虽然修改的频度不是很大，但在修改后即要发生作用，所以也采用事物型复制，收费中心为出版者，收费站为订阅者。将管理型的数据发布到各收费站。

由于公司总部不需要实时更新，所以收费中心数据库服务器与公司总部服务器之间的数据同步设置为快照复制，公司总部数据库中建立收费中心表的快照，对这些数据的修改在收费中心进行，把收费中心数据库服务器设置为出版者，出版物为汇总数据，公司总部数据库服务器设置为订阅者，快照代理将准备包含有被出版数据表的结构与数据的快照文件，在分发者上存储这些文件，并在分发者的分发数据库中记录同步任务。在收费中心数据库服务器上建立一个作业，设置为每天早上 8 点更新一次（收费日的天为 8 点到第二天 8 点），公司总部就能得到截止到前一天所有收费站的汇总数据。

测试

数据库设计好了，就要对它进行测试。我们的测试策略为分步测试：首先测试收费站数据的正确性及完整性，在多台收费机上同时输入几组车型、收费类型的数据，查询数据库的流水数据是否正确，再看汇总数据是否正确；收费站正确后，再测试收费中心的数据的正确性、完整性及延时；再测试公司总部的数据正确性、完整性。

一开始我们把所有的表都建立在一个出版物中，但在测试中我们发现，由于收费站一个表的错误而造成的复制的中断，经常要重新配置复制，所有的表都要重新选择及设置一遍，非常繁琐，我们采取一个表建立一个出版物的方法来简化操作，只要恢复出错的复制，其他的复制仍然能正常执行，而且哪一个表的复制发生中断也很明确，不仅简化了操作，也加快了处理时间，就是日后用户维护起来也简单明了。

在设计过程中，基于查询及安全性的需要，我们大量的使用了视图，第一加快了查询速度；第二也防止了人为因素造成的数据的更改。

总结

按照以上的设计方案实施后，完全满足高速公路系统对数据实时性和完整性的要求，系统目前只在内部使用，而以后全省的高速公路收费系统实行联网，在外网上发布信息，那时数据的安全性及查询的响应速度将是我们考虑的重点。

论基于 WEB 的系统测试策略

摘要

我所在的公司是某省电信公司。2003 年初，公司构建了互联网宽带增值业务计费平台。该系统的目标是利用电信营运网络、用户资源和品牌优势，为各 ICP 和 ISP 提供统一门户、统一认证和统一支付的营运平台。系统业务功能多，与传统的电信业务系统的接口多，性能和安全性要求高。系统选用了微软公司的 .NET 开发平台，经历了多个测试阶段。根据系统的特点，本文重点讨论了我们在单元测试、功能测试和性能测试中采取的措施和策略，以及如何保证测试的充分性，并简要分析了基于 Web 的系统测试与传统的软件测试的不同之处。

我作为主要的技术负责人之一，参与了系统的技术选型、方案设计、需求分析和系统设计等工作，也参与了系统的测试过程。

正文

随着互联网的发展，一方面 ICP（内容提供商）和 ISP（服务提供商）有好的内容和服务，却苦于没有收费渠道；另一方面，用户希望使用有价值的内容和服务，却没有便捷的支付手段。互联网从免费转向有偿服务已成为一种趋势和必然。为此，我所在的某省电信公司于 2003 年初构建了互联网宽带增值业务计费平台，其目标是，利用电信营运网络、用户资源和品牌优势，为各 ICP 和 ISP 提供统一门户、统一认证和统一支付的营运平台，创造互联网用户、ICP/ISP 和电信营运商的多赢价值链。

系统选用微软公司的 .NET 开发平台，采用了 B/S 三层架构设计：表示层、业务逻辑层和数据层。表示层负责处理系统与互联网用户、ICP/ISP 以及业务管理人员的交互接口；业务逻辑层负责系统处理业务逻辑的所有 COM+ 组件组成；数据层是存放用户信息、业务信息和计费信息的关系数据库，采用微软公司的 SQL SERVER。系统业务功能多，与传统的电信业务系统的接口多，性能和安全性要求高，项目建设时间紧。

根据系统的具体特点和应用要求，我们仔细分析了业务流程和系统的应用环境，确定系统的测试要素，并据此编制项目测试计划、测试用例和测试脚本，将系统的测试划分为单元测试、部件组合测试、功能测试、性能测试、安全性测试和验收测试，重点关注了单元测试、功能测试和性能测试。下面就此作详细介绍。

首先，我们考虑项目测试组的成员不仅限于测试工程师，还请到了相关业务人员加入到测试队伍中。业务人员可以从实际业务的角度参与到测试数据和测试用例的准备工作中，并帮助分析测试结果，发现与实际业务相关而测试人员容易忽略的问题。

由于系统复杂、业务功能和接口繁多，因此我们从一开始就非常重视单元测试。在单元测试阶段，我们采取了开发人员自己编写测试用例、小组人员交叉评审和测试组抽查相结合的策略，以避免测试用例的片面性，同时要求单元测试应紧接在编码编译通过之后。

单元测试我们采用了 NUnit 自动化测试框架，用它在 .NET 类上创建和执行自动的单元测试。我们知道，.NET 引进了一个新的程序开发的概念——Attributes(属性)，让开发人员可以在程序代码之上再加入 Metadata(元数据)。Attributes 主要使用在 documenting your code(注释你的程序代码)，也可以用来提供有关 Assembly 的额外信息，NUnit 中的 Test Runner 会扫描已经编译好的程序代码，并且从 Attribute 里面知道哪些 Classes 是 Test Classes，哪些 Methods 是需要执行的 Test Methods。然后，Test Runner 使用 .NET 的 Reflection 技术来执行这些 Test Methods。这大大减轻了测试人员的工作压力，也有效地保证了单元测试的可靠性，为后续测试的顺利进行奠定了坚实的基础。

系统的功能测试，我们主要考虑了页面链接测试、表单提交测试和数据库测试。

在系统提供的统一门户上，有互联网用户的网上家园和用户定制服务，也有电信营运商提供的宽带增值业务，更多的是合作 ICP/ISP 的产品和服务。为了用户能通过统一门户查询、购买和使用各项服务，就必须保证站点所有链接的正确有效。我们采用了以前开发互联网智能搜索引擎项目的一个衍生产品——链接自动测试工具，它能自动检查、分析站点内所有页面链接，并生成一份详细的清单，报告链接页面是否存在、哪些页面未作链接等。这基本上不会带给测试人员工作量。

对于表单提交测试和数据库测试，我们采取了运行测试的方法。测试之前，依据系统需求分析，对每一个交互功能页面设计测试用例，重点关注关键页面的测试用例，如用户注册、登录、支付、服务目录等页面。测试针对提交操作的完整性来进行，以校验提交给服务器的信息正确性，同时检查数据库存取时数据的一致性。例如：用户密码要求长度不低于 8 位的非数字字符和数字组合，并区分大小写。测试时我们分别用 8 位以下、8 位及 8 位以上的非数字字符、数字字符及其混合型进行测试。另外，如果表单只允许接受某些指定值，或使用了默认值，还必须检验指定值和默认值的正确性。

对系统的性能测试，我们采用了 Webstress 测试工具，从连接响应速度和负载两个方面来进行。在连接响应速度测试时，通过测试工具模拟单机访问，我们发现某些组件和页面响应速度较慢。经过分析，我们进行了相应部份的代码优化、SQL 查询优化，并对应用服务器调优，采用了数据库连接池技术等，系统性能得到了明显改善。负载测试时，通过测试工具模拟多用户访问系统，检查系统所能承受的极限访问量。测试中，我们将系统各个部份的日志和监测工具全部打开，在并发用户数逐渐加大的过程中，记录系统在事务处理、CPU 负载、内存使用等方面的变化情况，借助这些数据查找并分析系统的性能瓶颈，采取针对性的改善措施。

由于系统功能多，与外部系统有大量的接口，系统建设时间紧张，而系统又是为互联网用户和合作 ICP/ISP 提供统一门户和计费服务。测试是保证系统质量的重要手段，测试的质量直接关系到项目的成败。为了使系统在功能、性能、兼容性、安全性以及页面风格和整体布局等方面有更进一步的测试，保证系统在投入运营前测试的充分，我们将系统部署到实际生产环境，接入了多家合作 ICP/ISP 的服务，同时在门户上开通了用户在线反馈系统，通过举办有奖专项活动鼓励用户使用系统并反馈系统的错误和缺陷，同时安排专人响应并整理用户的反馈，及时交由开发人员修改。实践证明，这是非常有效的方法。

从系统的实践中，我体会到基于 Web 的系统测试比传统的软件测试要困难得多。基于 Web 的系统在网络环境、用户范围、使用环境等方面都比传统的软件系统复杂，在测试时必须更多的关注系统的并发性、兼容性、安全性和可用性。同时，基于 Web 的系统比传统的软件有更短的发布周期，测试人员和测试管理人员面临着从测试传统的 C/S 结构和框架环境到测试快速改变的 Web 应用系统的转变。

目前，软件开发正在向基于 Web 的应用发展和转变，这对测试工作提出了新的要求和挑战。测试技术还有很多领域和知识点需要我们去研究、探索和实践，并在实践中不断总结和提高。

企业应用集成的实践

摘要

为了向铁路各部门用户提供高可用、整合的信息，受铁道部科技司委托，我单位承担了“信息应用集成的研究”这一项目的研究和开发工作。本人作为该项目的负责人之一，担任了方案设计师的职务。该项目的目标是实现现有系统中的数据共享，有机的结合相关联的数据，搭建统一的使用平台，为今后铁路信息化更大规模应用集成建立可行、可靠的依据。本文主要从以下三个方面描述作者在该项目中的工作：确立企业应用集成的解决方案；选择应用集成规范和平台；在现有的铁路信息系统中选择二至三个，对应用集成的方案进行实践，分析并改进方案。

正文

近年来，铁路信息化建设取得了突飞猛进的发展，特别是正在建设实施中的“铁路运输管理信息系统（以下简称 TMIS）”工程。TMIS 是实现铁路运输管理现代化的一项重要工程，覆盖了铁路运输组织管理的各个环节。从功能上来看，TMIS 以货运管理为核心，分为列车预报系统、货票系统、车号自动识别系统、货运营销和技术计划系统等多个子系统。从组织结构上看，TMIS 由铁道部、14 个铁路局、50 个铁路分局和 2000 多个车站四级组成。目前，TMIS 已经成为一个由多个子系统构成的、庞大的多级分布式应用系统。

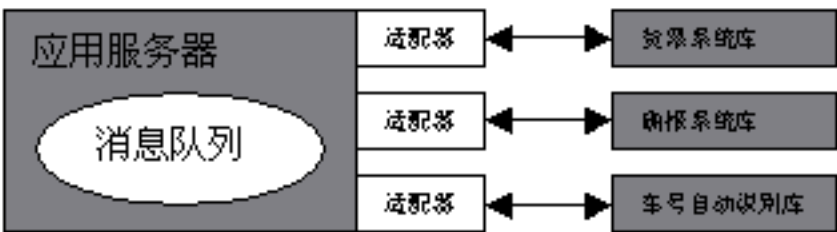
在部、局、分局的数据库中，已经建立了能够实时接收数据的货票库、预报库和车号自动识别库。货票库主要用于货物运费清算、预报库主要用于实时掌握调度行车信息、车号自动识别库主要用于车辆位置查询管理。为了实现列车、机车、车辆的实时追踪管理，必须共享各子系统数据；同时为了实现各级子系统的应用集成，必须选择一个开放式的应用集成规范和平台。为此受铁道部科技司委托，我单位承担了“信息应用集成的研究”这一项目的研究和开发工作。由于在 TMIS 工程建设中的多年工作经验，特别是在货票子系统中担任系统设计师的工作经验，本人有幸成为“信息应用集成的研究”项目的负责人之一，担任了项目方案设计师的职务。

经过分析，我对该项目的目标总结如下：为实现现有各系统中的数据共享，有机的结合相关联的数据，应确立企业应用集成的解决方案；为实现各级子系统的集成，建立统一的应用平台，应选择一个开放式的应用集成规范和平台；为建立今后铁路信息化大规模企业应用集成可行可靠的依据，应在现有的铁路信息系统中选择二至三个，对应用集成的方案进行实践。

下面我就从三个方面对企业应用集成的实践进行描述。

一、确立企业应用集成的解决方案

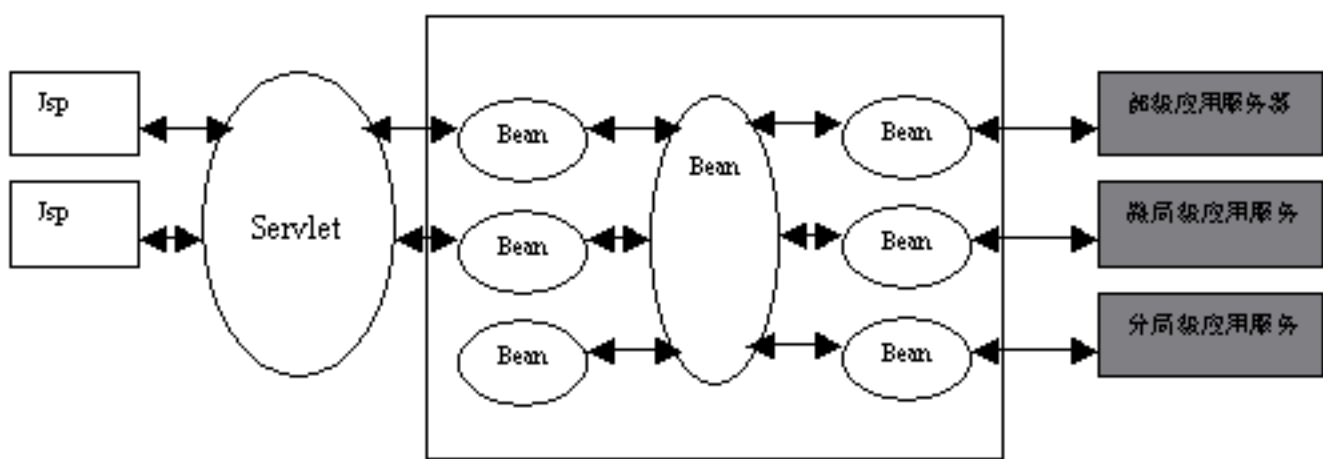
从集成的深度上来说，本方案包含了数据的集成，同时也包含了应用系统的集成。企业应用集成常用的方法有：基于客户端/服务器的方法、基于消息代理的方法、基于应用服务器的方法等。在这里我选择了基于应用服务器的方案，即通过建立独立的底层架构来连接企业的异构系统、应用数据等。如图所示：不同的子系统通过不同的适配器连接到应用集成服务器，应用集成服务器内部通过消息传递实现不同应用之间的交流。



二、选择应用集成规范和平台

从集成的广度上来说，本项目涉及部门之间的系统集成，同时也涉及了企业不同级（部、局、分局、站段）的系统集成。为了保证各级系统的数据一致，加强各级之间的交流，应该从整体来考虑企业的集成方案。较好的方法是选择一个开放的集成平台，把不同级的应用纵向的结合起来，通过统一的门户信息网站向外发布信息。

目前，开放式企业应用集成规范和平台的主流技术有两种：微软公司的 COM+规范和 Windows DNA 平台；SUN 公司的 EJB 规范和 J2EE 平台。由于 J2EE 架构的与操作系统无关的特性，为了更好的支持现有的不同级的系统，我选择了 EJB 和 J2EE 的组合。如图所示：系统采用 MVC 的设计模式。Jsp 所在为表示层，提供数据的展示功能；Servlet 所在为控制层，处理表示层与业务层的关系；业务层由多种 Bean 组成，Session Bean 处理交互相关的信息，Transaction Bean 处理具体的业务逻辑；Entity Bean 与数据层交互。



三、应用集成方案的实践

为了实现列车、机车、车辆、货物的实时追踪管理，需要在现有的几个系统中提供近乎实时的系统集成和数据集成。这些系统包括：货票系统提供了货物的名称、重量、装货车辆、发送人、收货人等信息；车号自动识别系统提供了车辆的位置等信息；预报系统提供了列车的车次、机车号、车辆号等信息。应用服务器通过编写各自适用的适配器连接这些系统。

应用服务器上有一个匹配作业，当车号自动识别数据入库时启动的该作业，通过车号和通过时间，在预报库中匹配该车所在的列车、机车等信息，然后在货票库中匹配货物的名称、收发货人等信息，匹配后的信息写入动态追踪库，至此一次作业完成。同一时间段内，应用服务器上有可能有 n 个等待匹配的作业，依次放在消息队列中，通过编码实现的应用集成程序统一调度，消息队列的实现选用 IBM 的 MQ series。

企业不同级（部、局、分局）的系统之间的数据更新不同步，造成各级之间的矛盾，通过数据同步服务器的加入，实现三级数据的有机整合，保证数据的一致性。

数据同步后写入中央数据库，由三台集群的 web 服务器向外发布信息。同时满足铁道部、路局、分局用户的使用，包括货票信息应用、预报信息应用、车号自动识别信息应用和动态追踪综合应用。编码工作量相对较少，系统响应时间快。

企业应用集成后，实际的效果证明本项目的系统方案设计到达了项目目标的要求，横向能够集成不同系统的相关数据，纵向能够提供统一的应用平台。并且扩展方便，通过编写新的适配器，可以集成新的应用系统；通过广域网和局域网的建设，能够加入更多的用户。同时，集成后的新增的动态追踪系统也成为越来越多用户的热点应用。

应用系统集成后，信息通过相对统一的界面向外发布。由于用户的部门和级别不同，因此主要关心信息的角度各不相同，所关心信息的内容也有所不同，用户可能需要多次选择或点击才能得到想要的信息。引入 XML 技术，实现个性化的展示层，方便用户的使用，是我在该项目中的下一个工作目标。

论依照 CMM3 级要求的软件过程裁剪

摘要:

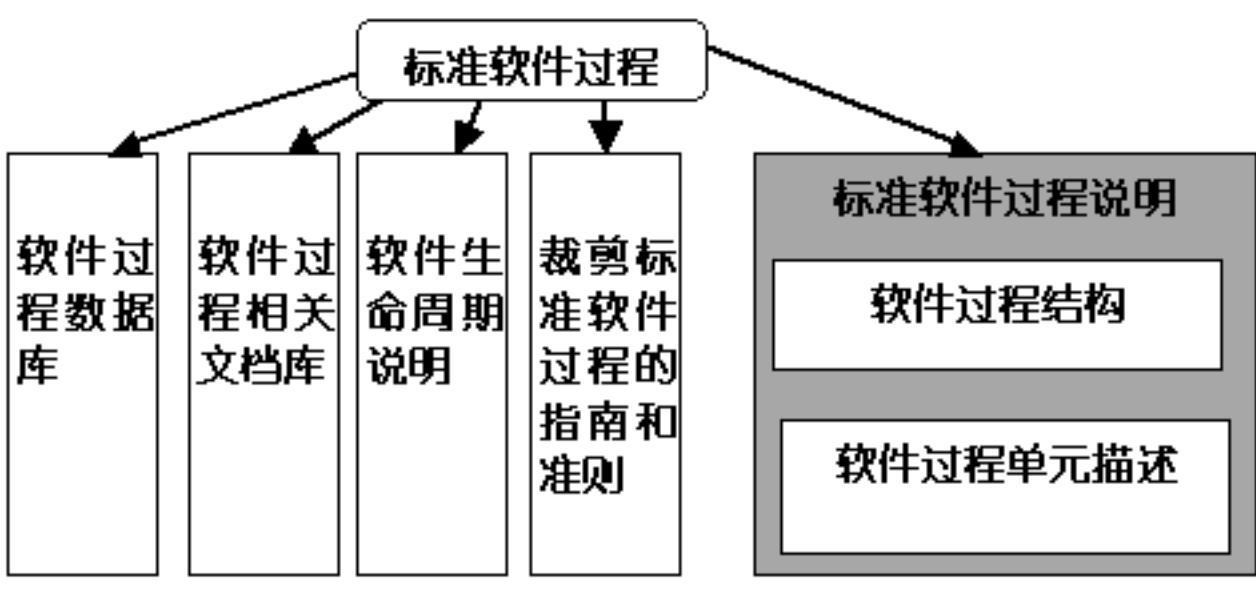
2004 年 3 月,我参加了国际联运过境运输管理系统的开发,担任项目经理一职,同时兼任系统分析师。本文通过该系统讨论依照 CMM3 级要求对软件过程进行裁剪的方法。

国际联运过境运输管理项目是一个开发时间为 1 年,软件开发人员为 6 人的中等开发规模项目,被公司指定为 CMM3 试点项目。我依照项目特点与 CMM3 要求对该项目进行了裁剪,裁剪后的软件过程包括:选择迭代增量型与瀑布模型结合软件周期模型;项目组由系统工程组、软件工程组、SQA 组、SCM 组、系统测试组构成;项目总体计划包含的内容与制定方法;过程数据的采集方式;SCM、SQA 活动方式等。本文最后讨论了该软件过程在执行时的效果与不足。

正文:

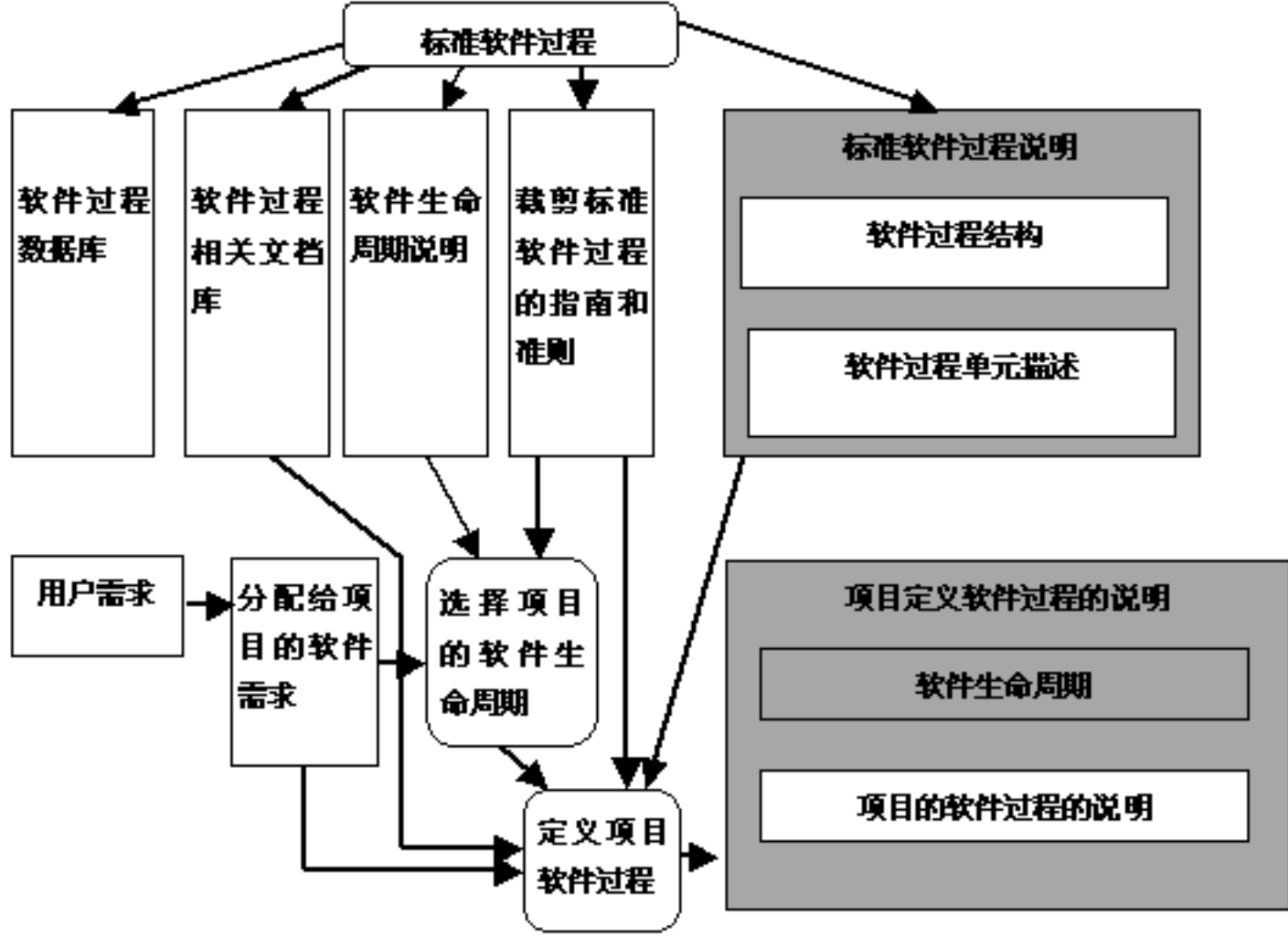
2004 年初,我公司受铁道部国际合作司委托开发国际联运过境运输管理系统,该系统建成后能在铁路国际联运制票点实现国际联运过境货票填制、数据上报,在联运信息采集点实现其它联运信息采集上报,在路、分局国际联运处、铁道部国际合作司实现联运信息综合应用等功能。由于我曾参与铁路现付货票系统的设计与开发工作,积累了相关业务知识,以及具备了软件开发管理经验,因此被公司指派为此项目的项目经理,负责对项目开发进行管理,同时担任系统分析师职责。公司正进行 CMM3 评审准备,并预计于 2005 年初进行 CMM3 正式评审,由于该项目开发时间为 2004 年初至 2005 年初,计划开发时间为 1 年,软件开发人员 6 人,属于中等开发规模项目,因此又被公司指定为 CMM3 试点项目。

按照公司规定,所有 CMM3 试点项目都必须遵守公司 CMM 项目管理体系要求。公司对这一级标准过程规定了五方面的内容,如图一所示。



图一：公司 CMM3 级标准软件过程示意图

结合公司对 CMM 项目的要求,我制定了本项目的裁剪方法,即:首先根据用户需求产生分配给项目的软件需求,然后根据公司标准软件过程要求中的第三方面和第四方面确定软件的生命周期模型,接下来结合第二方面和第五方面为本项目过程裁剪,定义本项目的软件过程,编制本项目软件过程说明。如图二所示。



图二：本项目过程裁剪方法示意图

下面我将从软件生命周期的选择、项目组的组成、项目总体计划的内容与制定方法、过程数据的采集方式、SCM 与 SQA 活动方式等 5 个方面论述对本项目的过程裁剪。

一、软件生命周期的选择。

对国际联运过境运输管理系统进行初步系统分析后,我了解到系统的功能需求可以分为三部分,第一部分:过境货运制票部分,该部分与国内现付货票系统处理流程非常相似,需求已经相对稳定,用户希望能尽快实现;第二部分:联运信息采集部分,该部分存在各路局目前的统计口径没有统一的问题,因此还需要等用户进一步研讨后确定;第三部分:联运信息综合应用部分,该部分未来统计数据存在变化性,统计报表增加、修改、删除的可能性非常大。

根据需求分析后项目可能遇到的变更情况,结合以往的开发经验,我为本项目确定了迭代增量型与瀑布模型结合的软件开发周期模型。软件开发周期共包含三次迭代过程,分别为:过境货运制票部分,联运信息采集部分,联运信息综合应用部分。过境货运制票部分由于需求相对稳定,并且是用户希望尽早实现的功能,属于第一次迭代过程;联运信息采集部分由于还需要进一步确定,属于第二次迭代过程;联运信息综合应用部分由于变化的可能性最大,属于最后一次迭代过程。每次迭代过程中又遵循瀑布周期模型的原则,分别定义需求、设计、编码、测试里程碑,以文档作为驱动,每个里程碑都需要进行严格的正式评审。

二、项目组的组成。

项目在合同中规定开发时间为 1 年,对项目进行评估后,公司决定为此项目安排 6 名专职开发人员,另配 1 名公司级 SQA 人员、2 名公司级测试人员。结合公司 CMM 项目对过程的要求,本项目组在公司 SEPG 组及其它过程支持组的协助下组建,包括:系统工程组、软件工程组、SQA 组、SCM 组、系统测试组。系统工程组、软件工程组由软件开发人员组成;SQA 组由公司 SQA 成员组成;SCM 组有一名软件开发人员兼任;测试组由公司测试组成员与软件开发人员兼任组成。与项目组相关的其它组还包括:公司 SEPG、公司 SCCB、技术委员会、培训组等。



图三：项目组成示意图

三、项目总体计划的内容与制定方法

根据公司软件标准过程说明,项目总体计划在该项目立项初期依照项目截止时间进行初步估计,在需求分配以后进行细化,并在第一次迭代中《需求规格说明书》正式评审时进行项目总体计划正式评审,且总体计划中的活动时间依据项目进度计划时间进行制定。CMM3 级要求项目进度计划的制定需要在估计了项目规模、工作量的基础上进行,在本项目的规模、工作量估计上我采用了专家估计的方法,具体做法是:邀请多名有开发经验的专家对关键技术文档页数、代码行数、测试用例个数进行背对背估计,将估计结果用 6σ 方法进行分析,当结果偏差超过接受阈值(本项目接受阈值设定为 30%)时,对不符合项重新进行估计,直至全部接受。

四、过程数据的采集方式

CMM3 要求对软件过程数据定时收集与测量。为收集项目的实际数据,公司正在使用一套自主开发的项目管理工具,使用该工具对软件过程数据定时收集与测量的方法如下:首先由项目经理填写项目任务分解表(WBS),系统自动生成项目组成员每日工作计划;然后项目组成员依照工作计划填报每日工作量,每周工作产品规模;此外,项目经理、SCM 员、SQA 员、系统测试员,负责收集管理活动工作量,填写项目管理活动测量表、SCM 活动测量表、SQA 活动测量表、系统测试活动测量表。这些收集的过程数据,最终都进入公司软件过程数据库。

五、SCM 与 SQA 活动方式

为保证项目在开发阶段和里程碑之后都能对开发产品进行版本控制,本项目的配置管理库中设置了三个子库,分别为开发库、审核库和基线库。SCM 活动的具体做法是:开发过程中产生的工程产品由开发产品责任人放入公司配置管理开发库,里程碑提交正式评审时由 SCM 员根据评审申请将相关工程产品移入审核库,评审通过后再移入基线库。进入基线库产品的任何变更必须经过变更流程,由 SCCB 进行变更控制。SQA 员对软件工程产品是否满足模板要求进行审核,并在每月第二周进行软件管理活动的评审,依照具体情况开具不符合报告,跟踪不符合项的解决情况至问题关闭。

以上是我在国际联运过境运输管理系统项目中根据公司 CMM3 级要求对软件过程进行的裁剪。在裁剪的过程中,我得到了 SEPG、SQA 组、测试组、培训组等公司过程支持组以及其它项目组的帮助,裁剪后的项目软件过程在项目开发执行中是令人满意的。项目一直处于受控状态,进度、成本、质量都得到一定的保证,更为公司的软件过程库提供了宝贵的真实的数据。最后,我对实现裁剪后软件过程中实施的效果谈一点看法。

在本项目过程中产生的管理活动、管理文档非常繁多,由此花费的工作量、成本较大,我个人认为项目组人员 10 人以下的开发项目可以引入敏捷开发思想的软件开发过程,通过灵活的开发方法、有效的沟通完成项目目标,这样的效率可能会大于来自参考公司过程库中历史同类项目经验数据所带来的好处。

论软件需求分析方法和工具的选用

【摘要】

本文通过一个集成电路设计有关的软件项目，讨论了该项目的主要特点和本人所担任的工作，着重讨论了在项目需求分析过程中采用的具体方法和工具以及选用的理由。

由于项目的专业领域的特殊性，分两类不同的需求讨论了需求分析中遇到的问题及解决方法；在这个过程中给出了对选用的具体工具和方法的效果的描述。接着本文讨论了对使用方法的改进的一些想法以及具体的实现过程。最后提出了我对需求分析的某些看法，强调了与客户沟通的重要性。

【正文】

近年，我一直从事某企业中有关 IT 项目的开发，有一个系统是由于计算机辅助电路设计的，包括了从上流设计（注：日式说法）到下流设计（注：日式说法）的所有流程，如用于可设计百万门数量级的逻辑门电路。有关方面把电路中路径的提取、过滤以及表示的某软件开发任务交给我公司，我有幸担任了该部分的需求分析以及设计。

我所设计部分为单独可启动的软件，主要是解析文件中的连线路径，以列表视图和用直方图等把它们显示出来，还可以执行诸如查找与过滤等功能。

委托方对此提供了很初步的需求说明，把一些基本功能及性能要求描述了一下。我在需求分析时的工作主要有两点：第一，对该软件的界面等详细需求要自己重新进行分析提取。第二，对于已提供的功能要求需要深化和细化，以形成真正完整的需求分析文档。

在接到需求分析任务后，我分析了一下所要完成的工作。发现由于是专用领域的软件，对专业领域要求相当高，所以准备把此项目分成两部分：

（1）界面所受专业领域影响几乎没有，但由于全部没有任何要求，反而会感到风险和改动可能是最大的。

（2）功能方面由于委托方的许多功能都可以调用相应模块来得到，并且已有了相应的书面的简单需求，相应来说只是完成深化。对界面，我采用了部分 RUP 的思想迭代与渐进。而对功能需求采取了分层细化，每细化一层就要求委托方确认、修改和补充。

首先把风险较大的部分完成，这是现代软件开发的基本常识。我选择先进行界面的需求分析。第一步是根据功能描述抽取出逻辑模型，并使逻辑模型与界面元素及功能一一对应，大体上决定了界面应有的功能，然后根据该界面功能描述，确定具体的控件，这时，我参考了委托方已初步完成的主窗口的界面布局及控件的使用规律，然后根据需要完成的功能从 Qt（由于要支持 Windows 和 Unix 双平台，所以控件库采用 Qt）的类库中选择相应的控件。在提取和抽象逻辑模型时，我采用了 Rose 2000 中的用例图，即以 USE-CASE 图来描述与外部的关系。之所以采用 Rose，我是基于以下的原因：第一，在已开发的部分中，委托方统一要求我们使用 Rose 进行类和顺序图等的设计和代码生成。第二，Rose 提供了标准的图来描述系统与外部的关系，在全球范围已是一种标准结构。第三，使用上的方便性。我用 Rose 的 USE-CASE 图，理清了我们的软件窗口（日式说法，即：沟通人员）与委托方主窗口（日式说法，同上）以及外部角色（操作者）之间的相互关系。

在确定了界面元素后，考虑到文档的可理解性不是很强，我采用 Visio 2000 把界面的外观绘制出来，写上了基本的控件作用，随后送给委托方评审，幸运的是除了几个小功能的修改，委托方基本批准了我的方案。

下面的工作是为控件的行为及状态变化制定相应的状态迁移图，我选用的工具仍是 Rose，我用了状态图和时序图，把重要的控件状态变化及相应顺序进行了描述，随后的几天把相应的 DOC 文档建好写明，基本上界面设计就完成了。

下面的需求是针对功能需求的。虽然委托方技术部门有初步的需求文档，但由于领域的专门化不对，我不清楚其中复杂的路径提取关系及较深入的专业术语，一直有一种举步维艰的感觉。只能采用分层细化的原则，从最初的几条深入一层变成十几条。这样的话，不会一下子碰到太深的专业问题，可以循序渐进从委托方与文献的解答中不断学习，深化自己对专业领域的了解，这样在设计中自己始终是层层推进的，不至于碰到无法逾越的专业障碍。

在这一阶段的开发中，由于一直是与自己不熟悉的专业领域打交道，所以我觉得一些辅助设计工具似乎无法发挥应有的功能。在这期间，对我帮助最大的应是公司的 E-Mail 系统，所有不清楚的问题的提出，以及对问题的解答都通过它进行周转。换句话说，在需求分析阶段，它起到了一个与客户的交流沟通和客户需求的提取作用。所以，我认为在这一阶段，E-Mail 系统是对我帮助最大的工具，其次是 Excel，我用它建立了问题跟踪图表，对每一个提出的问题，均需要记录上去，把问题结果（可分为已清楚、仍不太清楚、不清楚、尚未回答）均记录下来，根据这些表，我可以很好地了解自己工作中的核心问题，并有了解决它的方向，提高了工作效率。

每进行一层的细化，我都把结果交付委托方审核，由他们进行提出何时能终止细化，大约在八层细化后，对方认为已达到了效果，确认可以结束。至此，分析工作全部完成，项目的需求分析基本成功了。

在这次需求分析中，我认为取得成功的原因主要是方法和工具选择得正确。在界面设计中采用了流行的辅助工具，对需求及逻辑模型的建立提供很大的帮助，可以更方便帮助自己理清思路。选用了迭代法，把一些错误的影响在功能分析和界面分析的不断迭代过程中加以改正。在后期，以功能需求为主时，我主要依赖的是沟通工具和表格工具，这也说明辅助工具不是万能的，需求分析的关键之关键，应是与客户的交流与沟通。

通过这次案例，我认为在软件的需求分析工作中，方法的重要性应远超过工具的使用，应当首先确定分析中的风险，把风险分类，用不同的方法去解决各类风险，而工具的选择不仅是要看影响力和名气，而是要真正为我所用，应把握其精髓，即是此工具到底可以对开发有什么帮助，而不是仅限于如何使用。我认为在需求分析中工具的作用不外乎两个：一是实际系统与环境模型等的抽象工具，二是需求表达工具。第一类的代表是 Rose，第二类的代表是 Word, WPS, Visio 等，在这次项目中由于地理上的限制还用到了沟通工具，Web 浏览与 E-Mail 服务系统。

最后我还是总结一下，在需求分析中工具方法都只是辅助项目成功的因素，真正的决定因素还是一“与客户的沟通”。

基于 Web 的医院管理系统的开发

【摘要】

随着市场的建立和发展，卫生行业面临了很多问题，一些制约卫生事业发展的矛盾和问题日益显现，因此，国家卫生部要求各医院采用信息化管理。前不久，我所在的部门承担了一个医院管理系统的设计和开发，医院希望以此来转变医院现有的运行机制，提高服务质量。该系统除了目前常见的计费系统、电子病历外，还包括门诊医生工作站、住院医生工作站、护士工作站等分系统。考虑到需要通过 Intranet 实现功能，并有部分的 Internet 功能，本项目平台最后采用了 Java 平台。

我在项目中主要负责项目的前期规划，即选择合适的开发方案，并建立部分的数据流，在系统实施过程中推动其顺利前进。此系统开发成功后投入运行，获得医院相关工作人员的好评。

【正文】

前不久，我所在的部门承担了一个医院管理系统的设计和开发，医院希望以此来转变医院现有的运行机制，提高服务质量。客户是一个市级医院，医院很早就开始从事信息化管理，但主要是针对计费这一块，后来，对其进行了改进，加入对病人电子病历的管理和采集，这样当病人二次就诊时，可以很容易地得到病人的既往病史。但随着系统的运行，院方希望对现有系统进行改进，为了更好得为病人服务，医院考虑加入一些其它的分系统，比如门诊医生工作站、住院医生工作站和护士工作站等等。因此我所在的部门承接了该 HIS 的开发，开发的成果是一个典型的 Java 技术在 Intranet 上的应用。

在开发前期，首先要设计出详细的系统功能规范，这一部分所花费的时间很少，因为卫生部在 2002 年曾经颁发了一个有关医院管理系统功能规范的通知，我们参考了该规范，很快确定了各分系统以及每个分系统的基本功能。但在选择合适的系统平台上有一番讨论，考虑到医院原有系统在某些地方运行良好，是否有必要将原有系统淘汰重新设计，另外新的分系统到底采用何种平台结构也是需要考虑的问题。

医院原有的计费系统和电子病历系统数据流向范围比较固定，主要集中在交费处和挂号处，一旦引入了新的系统，必然要将数据流向医院的各个部门。医院的 Intranet 已经实施，因此首先考虑采用 B/S 架构体系，旧系统的数据模型尽可能保留。在系统的软件平台上，我们考虑使用 Java 平台，可以让数据在整个系统安全、有效地流动；另外现在也有很多的 HIS 系统可供我们参考，虽然往往是单机版的系统，但其中的数据模型有很好的参考价值。医院的现有网络系统和操作系统多种多样，这就要求我们选择的软件平台必须具有开放性、平台无关性。而在不同的系统上安装相应的 Java 客户端虚拟机并不困难，最后，在项目组的讨论和征求客户意见下，项目组采用了此方案。

在项目中，我们这样设计 Java 架构系统，将系统分为三层：

(1) 表示层采用 Jsp 实现页面输出，这也是用户直接访问层，表示层接受来自网络浏览器的 HTTP 请求，然后返回给客户端浏览器可以显示的 HTML 页面；

(2) 中间件层用 Java 实现对数据库的访问，考虑到数据的分布特点，我们使用了数据库连接池技术；

(3) 数据库层用 SQL Server 实现数据库的管理和存储过程。

JSP 以其执行的高效性和使用的方便性，已成为近年来大家首选的因特网开发技术，JSP 是一种页面开发技术，它以 Java 为其服务器端语言，结合 Java Script 作为其客户端语言，能方便地实现页面的表示。选择 Jsp 作为前台语言，是考虑到它的平台无关性，能够兼容其他的操作系统和数据。利用 Jsp 可以将 HTML 文件很方便地发送到客户端 Web 浏览器，同时也可以支持一些非 HTML 格式的文档发送。我们在网站中的用户层页面设计中，广泛地参考了现有的一些成功案例，在节约设计和开发的费用的同时也得到了用户的认可。比起单纯的 Servlet 技术，Jsp 在页面元素上也更为丰富，在这里，我们为今后客户页面定制做了部分尝试，设计了一个比较简单的模块，用户可以通过选择控件来动态生成页面提供打印，当然在这部分我们设计中，可提供的数据库字段是固定有限的，灵活性没有很好体现。

我们大量工作主要放在中间件层，由于系统中的医生工作站分系统是主要部分，因此数据库的负载成为一个不得不考虑的问题。综合考虑，我们决定采用数据库连接池技术，因为基本上每个医生都对应一个客户终端，无论是病人资料的查询或录入，或者医嘱的处理，都涉及到对数据库的频繁读写，服务器对于数据库连接的频繁打开和关闭必然导致性能下降。一方面，我们预先考虑数据库的连接量，在系统初始阶段建立相应的存储空间，当数据库连接打开和关闭时都对该连接池进行处理；另一面，我们也使用了高速缓存技术，对某些固定的 SQL 查询结果，例如药品查询、药性禁忌等，将结果采用缓存存储，以加快对数据库的访问，降低了服务器端的负载，提高性能。值得一提的是，为了尽可能得避免纠纷，医院要求采用医生签名制度，我们设计了电子签名，采用加密算法保证各环节产生数据的有关人员不能抵赖。

数据库层我们选择了 SQL Server，程序员比较熟悉此平台的开发和设计。在前期，我们到医院做了大量的走访工作，了解整个数据流，同时广泛参考现有的系统，虽然对医院行业不是很熟悉，但数据层开发遇到的问题的并不是很大。在概要设计阶段，我们使用了一些计算机辅助开发工具，这也加快了详细设计的进程。比如使用了 Sybase 的 PowerDesigner 工具，在概要设计中规划了 E-R 图和各个分系统以及分系统之间的数据流图，然后让工具直接生成后台数据库中的基本表结构，大大提高了开发效率。

整个项目实施完成后，医院相关工作人员反映良好，但其中也暴露出了一些问题，值得改进。

首先，用 JSP 编程时容易导致系统信息的扩散。如果有人恶意攻击服务器，程序执行将出现异常。这时，就会在页面上打印出相应的错误信息。这些信息会暴露出这台服务器的路径信息。这个漏洞往往会被别人利用，程序员对此也一头雾水，下一步我们考虑从服务器入手，采用通用的异常说明界面，解决该问题。

其次，在设计上，可以使用更多的辅助开发工具和建模工具，比如 Rational Rose，可以利用它的代码自动生成功能，来大大提高开发速度；在用户界面定制方面，我们希望通过不断的实践，加强其灵活性，尽可能把可供选择的字段扩大到整个数据库。

最后，大量使用 Java 技术，势必对服务器的负载过大，在今后的开发中，除了现有的数据库连接池和缓存技术，我们考虑使用更多的数据库脚本来替代部分 Java 代码，来高速实现逻辑，相较而言，数据库脚本的执行速度较有优势，结合结果缓存，应该对服务器的性能有较大的提高。

上面是我在今后的系统设计和开发中需要注意和加以改进的地方，也是我今后应该努力的方向。

财务数据仓库系统的设计与实现

【摘要】

近年来，数据仓库技术在信息系统的建设中得到了广泛应用，有效地为决策提供了支持。2004 年 6 月，本人所在单位组织开发了财务管理决策系统，该系统主要是使高层领导掌握企业的经营状况及进、销、存情况，分析市场趋势。

本文通过对财务数据的分析，结合数据仓库开发原理，完成对财务数据仓库的数据组织，介绍了财务数据仓库的设计和实现方法方法。财务数据仓库的设计步骤主要是遵循数据库设计的过程，为分概念模型的设计、逻辑模型设计、物理模型设计和数据仓库生成等几个阶段。

目前，该项目已顺利上线，领导反映良好。在该项目中，本人担任系统分析师职务，主要负责系统架构设计和数据仓库的设计工作。

【正文】

2004 年 6 月，我所在的单位为了快速适应市场的变化，使高层领导及时掌握企业的经营状况及进、销、存情况，分析市场趋势，决定开发财务数据仓库系统。在该项目中，本人担任系统分析师职务，主要负责系统架构设计和数据仓库的设计工作。

在这个系统的设计过程中，我们遵循了数据库设计的过程，整个财务数据仓库的设计步骤如下：

- (1) 概念模型的设计；
- (2) 逻辑模型设计；
- (3) 物理模型设计；
- (4) 数据仓库生成。

1、概念模型的设计

进行概念设计所要完成的主要工作有：

(1) 决策需求分析：对于数据仓库系统而言，决策者最为迫切的需求在于，更加准确的掌握企业的经营状况及进、销、存情况，包括分析进货趋势，分析销售市场波动趋势，分析企业存货情况，分析市场经营状况发展趋势。所要求的操作数据库的数据有商品进货数据、商品销售数据、商品库存数据、顾客信息和销售商信息。

(2) 确定系统的主题域及内容在上述需求分析的基础之上，我们可以确定企业财务仓库系统的 3 个主题，分别是商品、顾客和销售商。如图 所示。

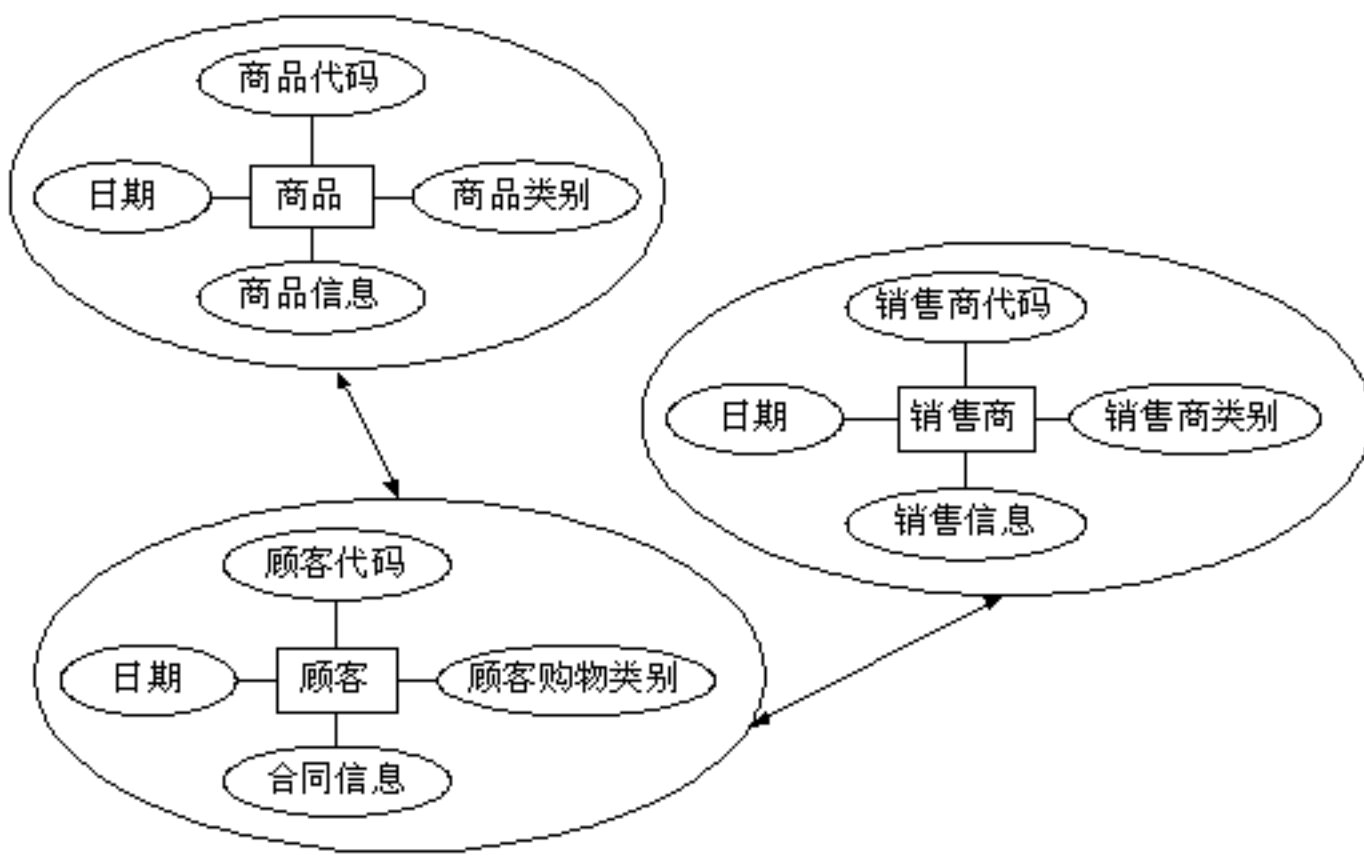


图 财务数据仓库分析模型图

2、逻辑模型设计

商品、顾客、销售商是财务数据仓库的 3 个主题，是其经营运作 3 个框架。商品主题描述企业商品分类及销售情况；顾客主题描述了企业对顾客进行分类及有关顾客合同的管理情况；销售商主题描述了企业销售人员销售商品及销售地区情况。其中商品主题作为中心，将这 3 个主题联系起来。它们的内容列出如下：

(1) 商品：商品固有信息(商品代码、商品名称、商品类别等)；商品库存信息(商品代码、库房号、库存量、日期等)；商品销售信息(商品代码、顾客代码、销售日期、销售单价、销售数量等)；

(2) 顾客：顾客固有信息(顾客代码、顾客名称、地址号、电话等)；顾客合同信息(顾客代码、合同代码、起始日期、终止日期、数量、价格等)；顾客购货信息(顾客代码、商品代码、单价、数量、日期等)；

(3) 销售商：销售商固有信息(销售商代码、销售商品、销售商品名、销售商地址等)；销售商地区信息(销售商代码、销售地区名、电话等)。

以“商品”为主题可以看到，首先，在从面向应用到面向主题的转变过程中，丢弃了原来不必要、不适合分析的信息，如各类领料单、出库单、入库单等；其次，在原有数据库模式中，有关商品的信息被分散在各个子系统中，如：商品销售信息存放于数据管理子系统，商品库存信息存放于商品库管理子系统中等，根本没有形成一个有关商品的完整一致的描述。而向主题的数据组织形式所实现的就是要形成商品一致的信息集合，以便在此基础上针对“商品”这一分析对象进行分析处理。

3、物理模型设计

(1) 确定数据的存储结构

通过定义功能/数据的交叉参照图，决定谁需要访问哪个范围的数据据。每个数据仓库

实施的最初阶段，必须标明最终用户的词汇表，定义恰当的商业术语，与底层数据联系起来。

由于数据仓库本身通常是面向主观意识的，基于最终用户的需求，创建数据仓库的第一步是识别和分析有关的内部数据和外部数据源。

数据模型在将内部和外部操作数据转换和集成到数据仓库里的过程中起着关键性的作用。在这个阶段，系统分析师必须收集信息，实施从数据源到数据模型的逻辑转换，**确定数据按简要或详细的程度保存在数据仓库中。**

创建词汇表、清除数据是产品数据与仓库数据之间的转化基础。词汇表是关于数据的数据。对于决策支持分析员来说，它是一个确定数据位置、理解计算法则的商业定位指导。清除数据和过滤数据包括转换数据、巩固数据和通过应用一致的命名法则协定解决在操作数据库中数据不一致的问题。

从长远的角度优化数据仓库，系统必须是灵活的，可扩充的，模块化的，以便有足够的

能力去适应系统的不断增长。

(2) 确定索引策略

由于数据仓库的数量比较大，因此在库结构设计时将每一个子库设计为树型索引结构，初始结点为所要决策的主题，其中间结点即为不同优先级的与该主题有关的查询角度与层次，而最终结点为经过预处理的有定义的数据集合。

4 数据仓库的实现

(1) 接口设计接口设计

在财务数据仓库中有几种方式：菜单式、问答式和图形式。接口设计涉及如下几个模式：输入响应模块、输出模块、人机对话管理模块和外围设备。由于时间和篇幅有限，这里就不详细介绍这些方式和模块了。

(2) 数据的采集

接口设计结束后，下一步工作就是将数据采集到数据仓库中。数据采集是将原始数据从多个传统事务处理数据库系统中提取出来进行清洗，集成等有关处理，使之符合数据仓库环境中对数据质量的要求后再装载入数据仓库中。

数据采集的主要工作有确定数据源的次序、元数据的管理、粒度划分、数据分割和数据的定期维护。数据在装入数据仓库底层的目标数据库后，还要完成的任务包括数据的定期清理、刷新、重建索引、初步划分主题等工作。

目前，该系统已经上线并顺利运行了一段时间，达到了开发的目标，为高层领导决策提供了可靠的数据来源和支持，得到了单位领导的一致好评。

数据仓库技术的发展包括数据抽取，存储管理，数据表现和方法论等方面。

在数据抽取方面，未来的技术发展将集中在系统集成化。它将互连、转换、复制、调度、监控纳入标准化的统一管理、以适应数据仓库本身或数据源可能的变化，使系统更便于管理和维护。

在数据管理方面，未来的发展将使数据库厂商明确推出数据仓库引擎，作为服务器产品与数据库服务器并驾齐驱。在这一方面，带有决策支持扩展的并行关系数据库将最具发展潜力。

在数据表现方面，数理统计的算法和功能将普遍集成到联机分析产品中，同时与 Internet/Web 技术紧密结合，推出适用于 Intranet 和终端免维护的数据仓库访问前端。

数据仓库实现过程的方法论将成为数据库设计的一个明确分支，并将成为管理信息系统设计的必备。

证券系统网络的安全性

【摘要】

我在一家证券公司信息技术部门工作，我公司在 1998 年建成了与各公司总部及营业网点的企业网络，并已先后在企业网络上建设了交易系统、办公系统，并开通了互联网应用。因将对安全要求不同、安全可信度不同的各种应用运行在同一网络上，给黑客的攻击、病毒的蔓延打开了方便之门，给我公司的网络安全造成了很大的威胁。

作为信息技术中心部门经理及项目负责人，我在资金投入不足的前提下，充分利用现有条件及成熟技术，对公司网络进行了全面细致的规划，使改造后的网络安全级别大大提高。

本文将介绍我在网络安全性和保密性方面采取的一些方法和策略，主要包括网络安全隔离、网络边界安全控制、交叉病毒防治、集中网络安全管理等，同时分析了因投入资金有限，我公司网络目前仍存在的一些问题或不足，并提出了一些改进办法。

【正文】

我在一家证券公司工作，公司在 1998 年就建成了与各公司总部及营业网点的企业网络，随着公司业务的不拓展，公司先后建设了集中报盘系统、网上交易系统、OA、财务系统、总部监控系统等等，为了保证各业务正常开展，特别是为了确保证券交易业务的实时高效，公司已于 2002 年已经将中心至各营业部的通讯链路由初建时的主链路 64K 的 DDN 和备链路 33.3K PSTN，扩建成主链路为 2M 光缆作为主链路和 256K 的 DDN 作为备链路，实现了通讯线路及关键网络设备的冗余，较好地保证了公司业务的需要。并且随着网上交易系统的建设和网上办公的需要，公司企业网与互联网之间建起了桥梁。

改造前，应用系统在用户认证及加密传输方面采取了相应措施，如集中交易在进行身份确认后信息采用了 Blowfish 128 位加密技术，网上交易运用了对称加密和非对称加密相结合的方法进行身份认证和数据传输加密，但公司办公系统、交易系统、互联网应用之间没有进行安全隔离，只在互联网入口安装了软件放防火墙，给黑客的攻击、病毒的蔓延打开了方便之门。

作为公司信息技术中心运保部经理，系统安全一直是困扰着我的话题，特别是随着公司集中报盘系统、网上交易系统的建设，以及网上办公需要，网络安全系统的建设更显得尤为迫切。但公司考虑到目前证券市场疲软，竞争十分激励，公司暂时不打算投入较大资金来建设安全系统。

作为部门经理及项目负责人，我在投入较少资金的前提下，在公司可以容忍的风险级别和可以接受的成本之间作出取舍，充分利用现有的条件及成熟的技术，对公司网络进行了全面细致的规划，并且最大限度地发挥管理功效，尽可能全方位地提高公司的网络安全水平。

在网络安全性和保密性方面，我采用了以下技术和策略：

- （1）将企业网划分成交易网、办公网、互联网应用网，进行网络隔离。
- （2）在网络边界采取防火墙、存取控制、并口隔离等技术进行安全控制。
- （3）运用多版本的防病毒软件对系统交叉杀毒。
- （4）制定公司网络安全管理办法，进行网络安全集中管理。

1、网络安全隔离

为了达到网络互相不受影响，最好的办法是将网络进行隔离，网络隔离分为物理隔离和逻辑隔离，我主要是从系统的重要程度即安全等级考虑划分合理的网络安全边界，使不同安全级别的网络或信息媒介不能相互访问或有控制的进行访问。

针对我公司的网络系统的应用特点把公司证券交易系统、业务办公系统之间进行逻辑分离，划分成交易子网和办公子网，将互联网应用与公司企业网之间进行物理隔离，形成独立的互联网应用子网。公司中心与各营业部之间建有两套网络，中心路由器是两台 CISCO 7206，营业部是两台 CISCO 2612，一条通讯链路是联通 2M 光缆，一条是电信 256K DDN，改造前两套链路一主一备，为了充分利用网络资源实现两条链路的均衡负载和线路故障的无缝切换，子网的划分采用 VLAN 技术，并将中心端和营业部端的路由器分别采用两组虚拟地址的 HSRP 技术，一组地址对应交易子网，一组地址对应办公网络，形成两个逻辑上独立的网络。改造后原来一机两用（需要同时访问两个网络信息）的工作站采用双硬盘网络隔离卡的方法，在确保隔离的前提下实现双网数据的安全交换。

2、网络边界安全控制

网络安全的需求一方面要保护网络不受破坏，另一方面要确保网络服务的可用性。将网络进行隔离后，为了能够满足网络内的授权用户对相关子网资源的访问，保证各业务不受影响，在各子网之间采取了不同的存取策略。

（1）互联网与交易子网之间：为了保证网上交易业务的顺利进行，互联网与交易子网之间建有通讯链路，为了保证交易网不受互联网影响，在互联网与中心的专线之间安装了 NETSCREEN 委托防火墙，并进行了以下控制：

- 只允许股民访问网上交易相应地址的相应端口。
- 只允许信息技术中心的维护机地址 PING、TELNET 委托机和路由器。
- 只允许行情发送机向行情主站上传行情的端口。
- 其他服务及端口全部禁止。并且在互联网和交易网之间还采用了 SSL 并口隔离，进一步保证了交易网的安全。

（2）交易网和办公网之间：对于办公网与交易网之间的互访，采用 CISCO 2501 路由器进行双向控制或有限访问原则，使受控的子网或主机访问权限和信息流向能得到有效控制，主要采用的策略主要是对具体 IP 进行 IP 地址与 MAC 地址的绑定。

（3）办公子网与互联网之间：采用东大 NETEYE 硬件防火墙，并进行了以下控制：

- 允许中心上网的地址访问互联网的任何地址和任何端口。
- 允许股民访问网上交易备份地址的 8002 端口。
- 允许短消息访问公司邮件 110、25 端口，访问电信 SP 的 8001 端口。
- 其他的都禁止。

3、病毒防治

网络病毒往往令人防不胜防，尽管对网络进行网络隔离，但网络资源互防以及人为原因，病毒防治依然不可掉以轻心。因此，采用适当的措施防治病毒，是进一步提高网络安全的重要手段。我分别在不同子网上部署了能够统一分发、集中管理的熊猫卫士网络病毒软件，同时购置单机版 KV3000 和瑞星防病毒软件进行交叉杀毒；限制共享目录及读写权限的使用；限制网上软件的下载和禁用盗版软件；软盘数据和邮件先杀毒后使用等等。

4、集中网络安全管理

网络安全的保障不能仅仅依靠安全设备，更重要的是要制定一个全方位的安全策略，在全网范围内实现统一集中的安全管理。在网络安全改造完成后，我制订了公司网络安全管理办法，主要措施如下：

- （1）多人负责原则，每一项与安全有关的活动，都必须有两人或多人在场，并且一人操作一人复核。
- （2）任期有限原则，技术人员不定期地轮岗。
- （3）职责分离原则，非本岗人员不得掌握用户、密码等关键信息。
- （4）营业部进行网络改造的方案必须经过中心网络安全小组审批后方可实施。
- （5）跨网互访须绑定 IP 及 MAC 地址，增加互访机器时须经过中心批准并进行存取控制设置后方可运行。
- （6）及时升级系统软件补丁，关闭不用的服务和端口等等。

在进行网络改造后，我公司的网络安全级别大大提高。但我知道安全永远只是一个相对概念，随着计算机技术不断进步，有关网络安全的讨论也将是一个无休无止的话题。审视改造后的网络系统，我认为尽管我们在 Internet 的入口处部署了防火墙，有效阻挡了来自外部的攻击，并且将网络分成三个子网较减少了各系统之间的影响，但在公司内部访问控制以及入侵检测等方面仍显不足，如果将来公司投资允许，我将在以下几方面加强：

- （1）在中心与营业部之间建立防火墙，通过访问控制防止通过内网的非法入侵。
- （2）中心与营业部之间的通讯，采用通过 IP 层加密构建证券公司虚拟专用网（VPN），保证证券公司总部与各营业部之间信息传输的机密性。
- （3）建立由入侵监测系统、网络扫描系统、系统扫描系统、信息审计系统、集中身份识别系统等构成的安全控制中心，作为公司网络监控预警系统。

保障网络安全性与网络服务效率永远是一对矛盾，在计算机应用日益广泛的今天，要想网络系统安全可靠，势必会增加许多控制措施和安全设备，从而会或多或少的影响使用效率和使用方便性。如，我在互联网和交易网之间设置了放防火墙的前提下再进行了 SSL 并口隔离后，网上交易股民访问交易网的并发人数达到一定量时就会出现延时现象，为了保证股民交易及时快捷，我只好采用增加通讯机的办法来消除交易延时问题。

电子报税系统的质量保证

【摘要】

2000 年 5 月,我担任某省国税局电子申报系统开发部的负责人,主要负责系统规划与组织实施开发。该系统工程浩大,业务需求多变,外部运行环境复杂,具有严格的安全、稳定、可靠性能要求,是一个基于网络的,大型数据库实时分布式系统。该系统主要由电子申报与电子完税两部分组成,电子申报部分由微机点对点申报、电话语音申报、Internet 网上申报、专用报税机申报四个子系统组成,并统一在电子申报服务器端软件模块下,电子申报服务器作为总个系统核心模块,还与银行前置机、税务各征管应用系统进行数据交换。我在将近两年的可行性研究、需求分析、系统研发与试点工作过程中,我们采用了 UML/Rational 软件工程方法、系统研发与试点同步进行的软件开发模式,而且还提出了独有的“流程编辑器”和“万能造表”的设计理念,有效地来提高了软件开发的质量和与效率。但仍然存在不能快速切换应用系统,进行故障屏蔽或系统恢复的容错性要求,缺少适应系统研发与试点同步进行的系统测试方法与工具等问题。

【正文】

2000 年 5 月,我开始担任某省国税局电子报税系统(以下简称 HNNT-EF)开发部的负责人,主要工作是对 HNNT-EF 进行规划并实施开发。HNNT-EF 系统分为电子申报与电子完税两部分,电子申报部分能够按税务局给定的内容和格式提供各种税种的申报表格和数据,为方便纳税人,还提供给各类型的纳税人使用“多元化”的申报纳税设备与手段,如“电话语音申报”、“微机点对点申报”、“Ineternet”等。纳税人利用各种终端,将自己的申报数据在经系统逻辑审核后传递到税务申报中心进行处理。电子完税系统是税务局、银行和国库间的税款划缴专用系统,应具有安全保障机制和可靠性措施,保证传递与存储税款信息安全可靠。而且,系统要求要具有较高的灵活性,能适应各种数据库开发软件接口,能连接多种数据库。我带队在经过将近一年的可行性研究、总体规划、可行性形究、需求分析后,于 2001 年 6 月正式立项建设。我主要负责 HNNT-EF 服务器端软件设计,服务器端软件是整总个系统的核心程序,包含四个功能模块,其中税局模块实现税局工作人员的申报功能控制和二次开发,接线模块用于电话、传真机、电脑及及 Internet 的接入服务、后台处理实现催报等功能、流程解释器主要用于解释二次开发的流程。

选择 UML/Rational。从项目开始建设起后,我给公司提出放弃用 Word、用 Visio 来写文档和设计结构图的原始做法,我拿着系统需求分析报告,向公司负责人提出如下看法:随着经济的发展,国地两税的业务需求将会不断发生变化,引发电子申报系统的申报格式与内容发生变化,系统的可维护性要求极高;并且,不断产生新的业务需求,决定系统要求有很好的可扩展性;电话语音申报子系统、Ineternet 申报子系统等决定需要进行分布式的小组开发,更需要清晰,易理解和共享的文档管理,如果系统不能满足这些需要,不采用完善的软件工程方法,公司将耗费大量的人力、财力去进行系统的后期维护。我建议公司采用 Rational Rose 进行需求分析与设计,并提出,在公司财力允许充许的情况下,逐步采用 Rational RequestPro 来管理需求,采用 Rational ClearCase 来记录变更需求。虽然,公司后来只采用了 Rational Rose 进行系统分析与设计,但是,这在提高软件开发质量与效率起到了很大的作用。回顾软件开发过程,我总结出有以下优点:详尽地获取用户需求,这是软件质量保证的关键。基于 UML 的 Rational Rose 提供的用例,交互图、顺序图的使用,使人们更容易发现其中的遗漏;使用 UML/Rational Rose 描述需求分析说明书,以“看图讲故事”的形式向用户讨论需求,用户更容易理解系统的操作,尽早发现隐性需求;并且,统一的系统分析设计语言,使程序员更容易理解系统分析师的设计意图,避免程序产生功能上的错误;这给系统的后期维护带来极大的便利,因为图形总比文本容易理解,规范的文档与版本控制便于程序员快速理解设计代码。所以,选择 UML/Rational,对提高系统的可维护性,可理解性,可扩展性,起到很大的作用。

采取系统研制与试点推广同步进行的工作开发模式。HNNT-EF 在该省两市进行了系统的总体规划、可行性研究、需求分析等前期工作后,我带领开发组提出“应用与完善同步,积极稳妥推进”的开发模式,制定了相应的短、中、长期技术方案各及各阶段的具体目标以及对应措施。因为 HNNT_EF 的服务端软件除作为电话语音申报、Internet 申报等子系统的统一处理平台外,还必须与银行基于 Oracle/Oracle 的前置机系统,税局基于 Sybase 的征管系统进行数据交换。象银行的前置机,系统要求银税联网的连接成功率大于>99%,数据准确率为 100%,平均响应时间少于<10 秒,我们在该省两个市进行试点过程中,发现大约有 2%的纳税户在进行电话申报纳税时因为响应时间过长而中断,通过大量的研究发现,这是银行前置机接口软件有问题,从而较快地得到解决,从这一实例可发现,边研发边试点,并跟据实际的应用情况对系统进行不断调整各完善,极大地提高了软件测试的效率和准确性,能较好地达到了预期的试点目标和应用效果。并且,这样与”用户打成一片“的开发模式,更容易使用户产生共鸣,尽早发现系统的隐性需求,最大地提高了系统的可用性。这为面对复杂的外部运行环境,采取系统研发与试点同步运行的开发模式积累了经验。当然,这种开发模块并不是通用的,需要有良好的关系处理和特殊的应用需求。

“流程编辑器”与“万能造表”的设计理念。在前期的需求分析中,我发现税务业务组人员经常提及申报表(内容、格式、审核公式)变动的说法,我经过分析得到证实,比如国家适应税率就经常更改,这就得修改申报表程序,这种后期经常修改源代码是软件开发的大嫉,我在经过大量的技术论证后,我提出了“万能造表“的概念,其原理是:将软件实现与表格彻底分离开来,建立申报表模板(与 Word 文字处理模板类似)的概念,与申报表程序分离开来。这样,如果税局的申报表发生了变动,我们不必修改源代码,只要按照规格修改申报表模块,并且,系统实现了将模板通过远程网络分发到纳税户微机上,实现模板的自动更新,这样,大大增加了系统的可维护性,做做到纳税户客户端的零维护,并且这种修改可由税务局相关工作人员进行,减少了公司的维护工作量。另外处,我考虑到,税务局随着经济发展,不但业务处理流程经常发生变动,而且还要不断时地增加业务处理子系统,怎样利用先期的开发成果与经验,较快速度地开发新的业务处理子系统,并在电子申报服务器端得到统一呢?我当时提出了与“编译原理”类似的流程编辑器。所谓流程编辑器是在 VC 语言的开发平台上,通过扩展功能函数,开发出针对数据库管理、串行通讯功能等组成的可视化的流程编辑语言,并以此为核心,开发出可视化的流程编辑器。这样,系统的处理流程可根据业务变化进行自定义,毋需重复写源代码,方便系统的维护和升级,并保证系统的稳定可靠。

完备的外部鉴定与审计。经过两年多的开发和试点工作,HNNT-EF 基本实现了技术开发和试点的目标和任务,为进一步完善和提高系统的技术应用水平,我提出了请该省信息厅进行外部鉴定与审计工作,进一步确定系统的先进性、可推广性,实用性。同时,通过对提供的项目资料是否齐全且合乎规范的审查,给各阶段系统开发人员按照软件工程开发方法,文档为先的开发思想形成一种压力。

综上所述,面对 HNNT-EF 复杂多变的用户需求与外部环境,采用先进的软件工程方法规范我们的开发过程、研发与试点同步进行的开发模式、独特的“流程编辑器“与”万能造表“设计理念等思想,保证了 HNNG-EF 系统开发的高效率与高质量,但是,在系统研发与试点过程中出现的一些问题,也值得我们思考:

软件开发的容错性。系统必须具有快速切换应用系统,进行故障屏蔽或系统恢复的容错性要求,这一点,我们在研发试点过程中有所忽视。HNNT-EF 系统的电话申报方式在某市开通使用的第二个征收期,由于使用户数由 800 户突增到 4000 户,突出地暴露了银行的税款扣缴系统处理并发请求的能力较差的缺陷,由此引发了 2002 年 4 月 10 下午系统在 2 个多小时无响应的事故。面对这种事故的发生,HNNT-EF 竟然竟然没有故障处理和应急措施,最可怕的是,这将大大降低纳税户的信任度。而造成的原因不得而知。为了使软件能带故障运行,我考虑采用中央存储库和对服务器进行备份的方案。但是,怎样发现外部系统带给软件的影响,我在这方面的经验很少。

缺少与系统研发与试点同步进行的系统测试的方法与工具。特别是缺少压力测试方法和自动测试工具。我原来思考引进 XP 开发思想,先开发测试程序,后开发软件模块,但因实施环境的复杂而放弃,这是一个我有待进一步学习研究的问题。

远程虚拟实验室的建设

【摘要】

本文首先概述了虚拟现实技术的产生、特征和应用,研究了我国计算机辅助教学的发展现状和存在的问题,以及虚拟现实技术在计算机辅助教学中应用的重要意义。

然后通过实例,介绍了虚拟现实技术在计算机辅助教学中的应用。2002年,本人所在的单位组织开发了一个用于远程教学的计算机辅助教学(CAI)系统。在该项目的开发中,我担任技术总监的职务,主要负责系统架构设计和方案选型的工作。

在该系统中,我们将虚拟现实的理论和技术引入到教学过程中。这样,不仅可以拓宽多媒体计算机的应用,为办学兴教节约大量资金,同时提高教学水平。

【正文】

计算机辅助教学就是利用功能强大的多媒体计算机作为媒体,开展教与学的一种教育培训手段。就全球范围看,我国的计算机辅助教学研究起步较晚,是在80年代后期展开研究的,但发展很快。2002年,本人所在的单位组织开发了一个用于远程教学的计算机辅助教学(CAI)系统。在该项目的开发中,我担任技术总监的职务,主要负责系统架构设计和方案选型的工作。

在该系统中,我们将虚拟现实的理论和技术引入到教学过程中。这样,不仅可以拓宽多媒体计算机的应用,为办学兴教节约大量资金,同时提高教学水平。我们以电子表格软件Excel为蓝本,运用虚拟现实的理论,设计开发了具有虚拟机制的课件“跟我学习”。

虚拟现实是一种支持多维空间信息技术的关键技术,是计算机技术的又一研究和应用热点。虚拟现实技术的主要内容是:实时三维图形生成技术、多传感器交互技术和高分辨率显示技术。

虚拟现实的理论和技术的应用非常广泛,特别是近年来三维图形、声音定位与合成、传感器、环境建模等技术的日趋成熟,为虚拟现实技术的研究提供了物质和技术基础。除了有许多专用的虚拟现实装置和系统投入使用,越来越多的研究人员将虚拟现实技术与各自领域的研究机制相结合,促进了高新技术的发展。目前虚拟现实技术应用已在通讯、医学、交通、商业和娱乐等方面初见成效。比如在娱乐领域,利用虚拟现实技术创造的虚拟环境,如星球大战,太空舰队等,不仅逼真,而且生动,具有极强的趣味性。虚拟现实技术与通讯技术相结合已经显示出诱人的前景。

计算机辅助教学就是利用功能强大的多媒体计算机作为媒体,开展教与学的一种教育培训手段。就全球范围看,我国的计算机辅助教学研究起步较晚,发展从浅到深,从研究到应用普及,从教育到培训,表明了CAI在我国广阔的前景。我国的计算机辅助教学虽然发展较快,但长期以来,在教育这个大舞台上,具有强大功能的计算机只能扮演电子黑板或电子图书的角色。辅助教学软件设计多以课本内容为剧本,顺序地在计算机上展示书本内容;而且表现形式简单,图形少且单调;动画和声音几乎没有,更不用说实践环节的联系了,基本上就是教材或习题集的翻版。这是CAI技术发展中的一个大误区。学习与应用是紧密关联的。要想掌握一门课程,特别是实践性强的课程,除了必须学习一定的理论知识以外,更重要的是实习。研究设计CAI课件的主要目的之一是实现个别化教学。为不同目的和起点的用户提供一个边学边做的实践学习环境。

虚拟现实技术即直接把计算机合成的信息通过声音、图像等多种信息作用于人的各种感官,包括听、看、嗅、触等感官,从而在人的周围形成一种人造空间,达到一种超模拟效果。这种效果也是教育与培训最为重要的,它无疑会极大的提高教育与培训的效果,特别是在学习迁移方面尤为如此。可以毫不夸张地说,虚拟现实技术就是为教育培训而创建的。利用虚拟现实技术可以使得对宇航员、飞机和舰船的驾驶员、指挥员以及各种复杂系统的操作人员的仿真培训更加生动、逼真,培训内容更丰富,效果更佳。在医学领域中利用虚拟现实技术开展外科手术的操作培训和实验,不但节约了大量的资金,而且比用活体进行培训和实验更加安全和方便。现在国际上正在研究一种利用虚拟现实技术进行遥控手术的技术,它的实施既可以作为医疗手段,也可以作为培训手段。此外,在幼儿的启蒙教育中,虚拟现实技术被用来作为开展寓教寓乐教育基本方式。

虚拟现实环境是一种和谐的人机界面。利用虚拟现实技术的合理、夸张和虚构的特点,使我们能够去完成传统的教育培训任务,能够去解决前人所未能解决的问题,能够去创造前所未有的教学环境。比如,虚拟实验、虚拟测试、虚拟培训等等。将虚拟现实的理论和技术引入教学过程中,不仅可以拓宽多媒体计算机的应用,为办学兴教节约大量资金,同时提高教学水平。

笔者以电子表格软件Excel为蓝本,运用虚拟现实的理论,设计开发了具有虚拟机制的课件“跟我学习Excel”,为学习者提供了一个多功能的虚拟学习环境。用户在学习过程中,不但能够学习Excel97的各种编辑的理论和方法,而且能够身临其境地进行实际操作。该软件集文字、图表、图形、图像、动画和声音于一体,引入人工智能和超链接机制,人机界面友好,为用户提供了一个边学边做的开放式学习环境。

该课件由以下六个模块组成:

第一部分“学习向导”。主要向用户介绍各个模块的功能和课件的使用方法,并给出几点学习建议。

第二部分“认识Excel”配合轻松的画面,简单扼要地向用户介绍Excel的性质,功能和主要应用范围。

第三部分“理论学习”。根据Excel的主要功能特点,结合文字,图形等的显示,按类分块的进行理论学习指导。还可以结合以后的模块进入演示或练习过程。

第四部分“实例演示”。结合Excel的主要功能进行实例操作演示。用户可以任意选择观看虚拟教师的操作演示。

第五部分“实战练习”结合Excel的主要理论和功能设计练习,并提供虚拟现实的操作系统,用户可以任意选择练习。

第六部分“测试训练”。综合Excel的理论和功能,配以虚拟现实的操作系统,设置不同类型的测试练习为用户选做。

虚拟现实技术日趋完善和提高,预期该领域未来的发展趋势为:

(1)硬件技术的发展。虚拟现实所要求的超级计算、图形图像处理、图像投影及交互等虚拟环境构建硬件技术将得到持续的发展,成本大幅度下降,从而促进虚拟现实技术的应用和普及。

(2)开放平台虚拟现实技术的发展。利用Intel微机硬件平台+Linux自由软件平台+中低档图形、数字投影设备构建经济型中低档VR系统,是虚拟现实技术的一个重要发展方向和现实途径。

(3)协同分布式虚拟现实技术的发展。虚拟现实系统已经由单机系统发展到分布式虚拟现实系统,现在人们正在向支持协同工作的分布式虚拟现实系统即协同虚拟现实(CVR)系统发展。

(4)可视化区域网络技术的发展。可视化区域网络(VAN)的目标是实现全球化用户用各种设备对可视化系统的交互访问以及可视化协同工作。VAN由高性能的可视化服务端(负责数据的存储管理、可视化计算等任务)、宽带网络和客户端组成。

(5)多感知能力的发展。未来理想的虚拟现实系统将提供人类所具有的一切感知能力,包括视觉、听觉、触觉,甚至味觉和嗅觉。

ERP 在商业管理中的应用

【摘要】

ERP 已突破了原来只管理企业内部资源的方式，把客户需求和企业内部的制造活动，以及供应商的资源融合在一起，更适合市场经济的需要。

我在一个上市公司的信息中心工作，2002 年 4 月，公司高层决定实施 ERP 系统，并委任我为实施部经理。对于如何系统地实施 ERP 系统，我们经过反复研究、权衡，最终决定采用“商品化软件+二次开发”的方式进行建设。在具体实施时，主要考虑以下问题：系统结构、系统集成、可扩展性、操作系统。

由于采取了合理的开发方式，且领导重视，真抓实干，系统顺利地上线运行，并取得了预期的效果。我认为，ERP 的发展趋势主要是：系统更加柔性化，应用范围广泛化，功能深度化，对供应链管理提供更好的支持，朝着 Internet 技术靠拢等。

【正文】

ERP 虽然是以 MRP II 为基础发展而来的一种管理思想，但是在功能和技术上已超越了传统的制造业并延伸到其他各行各业。在管理理念上，ERP 已突破了原来只管理企业内部资源的方式，把客户需求和企业内部的制造活动，以及供应商的资源融合在一起，更适合市场经济的需要。

ERP 在企业中的应用涉及到企业的生产经营机制、传统管理方法和管理基础等一系列实际问题，因此引进 ERP 系统的企业普遍出现了实施难度大、周期长、反复多、见效慢等现象。

我在一个上市公司的信息中心工作，该公司成立于 1996 年 4 月，由原来的国营大中型企业改制而成。公司以百货、钟表、眼镜、金银首饰等名、特专业商店和老字号企业为龙头，在全国各地开设专业特色分店、专柜。1998 年进一步调整经营结构，资本运行向优质企业靠拢。公司的十几家子公司的地理位置比较分散，而且子公司性质复杂，包括酒店、超市、房地产等行业，业务复杂、业务量大、跨地区经营是公司面临的最大问题。

2002 年 4 月，公司高层决定实施 ERP 系统，并委任我为实施部经理。公司所需要的 ERP 系统是一种能更好地满足企业多行业、跨地区、多供应商和销售渠道经营管理模式的信息管理系统，这种系统应该充分利用先进的计算机技术，为用户提供灵活性和可配置功能。

对于如何系统地实施 ERP 系统，我们经过反复研究、权衡，最终决定采用“商品化软件+二次开发”的方式进行建设，即购买符合要求的商品化软件，再与专业软件公司合作，根据实际需求进行二次开发。我们根据当时的计算机技术、行业特点等进行了可行性论证，最后达成一致意见，确定了“技术要顺应潮流，投资要受到保护，管理软件要强壮，合作公司要有实力”四项基本原则，保证 ERP 系统的顺利应用。

经过初步调研，我们决定 ERP 系统主要由以下几大部分组成。

(1) 供应商关系管理模块。本模块可以方便地加强与供应商的交流，降低公司的采购成本，它让供应商能查询与公司的业务往来情况和对帐单。

(2) 客户关系管理模块。为帮助公司最大程度地实施以客户为中心的战略思想，为客户提供更快更好的服务，本模块对客户信息进行收集与分析，提供客户注册、客户查询、客户社区、精品推荐、营销指南、服务宝典等功能。

(3) 销售管理模块。此模块用于管理销售业务，可以打印报价单、销售定单、发票等多种单据。

(4) 采购管理模块。本模块提供跟踪请购单和采购定单，定单收货确认与定单列印等一系列功能，帮助公司执行采购、安排生产计划、对于有效降低库存成本，保持产供销平衡，提高预测、决策的能力具有重要意义。

(5) 库存管理模块。本模块具备库存与仓库管理的多种功能，提供各式库存报表及产品目录，并与销售管理模块、应收款管理模块、采购管理模块与应付款管理模块紧密结合，共享库存资料。

(6) 应付款管理模块。本模块着重于公司业务预算和应付款项的管理和控制，便于平衡收支，提供安排付款，列印电脑支票，手写支票及更新库存数量，并自动生成财务凭证，更新到总帐中等多种功能。

(7) 应收款管理模块。本模块通过对客户、销售和应收款进行智能化的管理和控制，使得公司能够及时全面了解资金回收和应收款情况，此外还提供了多方面的销售、客户、产品和服务等分析报告。

(8) 总帐管理模块。本模块是一个可快速审核与分析总帐，自动完成从凭证汇总到报表生成的一系列财务核算过程。

(9) 网上商店。本模块将公司所售商品一一陈列在网上，让客户可以随时随地直接在網上快速浏览并选购商品，所生成的网上定单能够自动转入销售管理模块进行处理。

(10) 决策分析系统。本模块可按照用户需求，统计打印出各种财务、库存、应收、应付等统计报表，为管理决策提供依据。

基于以上需求，我们开始挑选合作伙伴。对上海、北京、深圳等一些软件公司分别进行了考察，最后选定了北京某集团软件公司作为实施 ERP 的合作伙伴。

在具体实施时，我们主要考虑了以下问题：

(1) 系统结构。我们在设计 ERP 系统结构时采用了三层机构，整个系统分为表示层、应用层和数据库层。表示层、应用层与数据库层分离。这样一来，用户可以从数据库服务器中装载可执行的应用，从而简化对应用的管理，同时在需要时，每个服务器还可以请求它自己的可执行模块。同时，为了确保 ERP 实施以后拥有足够的性能，项目实施小组在规划 ERP 项目时认真了解了网络的速度并且详细理解应用结构的网络性能需求，在此基础上决定公司总部和分公司之间的通信采用拨号方式。

(2) 系统集成。由于子公司多，且采用了来自多个不同厂商的管理信息系统，不同的数据库和硬件平台。我们将这些分离的 ERP 子系统相互集成起来，同时还将现有的应用同已有的应用和决策支持系统集成起来，最大限度地发挥了 ERP 项目的作用。在具体集成方法上，我们采用在一个单一的操作系统和数据模型上运行多个应用，减少了集成相关应用的复杂性。

(3) 可扩展性。因为我们的公司发展比较快，所以，在实施 ERP 时，我们充分考虑了系统的可扩展性。

(4) 操作系统。我们在实施 ERP 项目时考虑到操作系统的可移植性，因为有些服务器只能在某些硬件平台上实施。现在最常用的两种操作系统就是 Unix 和 Windows NT。Unix 宝刀不老，保持它在高端市场的领先地位。因为它能够支持高可伸缩的数据库服务器，使用基于 Unix 的方案将提供更大的可伸缩性，所以，我们选择了 Unix。

由于采取了合理的开发方式，且领导重视，真抓实干，系统顺利地上线运行，并取得了预期的效果。系统实施后，将公司的内部资源和外部资源整合在一起，有效地安排的产、供、销活动，满足公司利用全社会一切市场资源快速高效地进行生产经营的需求，实现了对公司的动态控制和各种资源的集成和优化，提升了公司的管理水平，进一步提高了效率和获得市场竞争优势。

通过 ERP 的实施，我认为，ERP 的成功确实是“三分靠技术，七分靠实施”，ERP 作为一个概念系统，管理到了企业的方方面面，这不只是一个简单软件系统所能解决的，它与企业的整个意识、理念、管理水平都有着密切的联系，需要一个循序渐进的过程。

ERP 也随着管理思想、技术的提高而随之发展。我认为，ERP 的发展趋势主要是：系统更加柔性化，从而更灵活地适应企业变化的需求；应用范围广泛化，渗透到了各行各业；功能深度化；对供应链管理提供更好的支持；朝着 Internet 技术靠拢；组件技术应用更加深入；安全保密性更高；成本更低、易用性更好；以及系统自助化。

嵌入式操作系统在控制系统中的应用

【摘要】

与通用操作系统相比，嵌入式系统由于其代码小，高度自动化，稳定，响应速度快等特点使之应用领域越来越广泛。我所在的单位是国内主要的医疗仪器研发单位，作为单位的主要技术骨干，2003年5月，我主持了 CTS—XXXX 的软件系统的开发，该系统是我单位的新一代数字 B 超诊断仪器的软件控制系统。在选择嵌入式操作系统时，我综合考虑了功能要求、开发难度、研发生产成本、运行稳定性、兼容性五个方面，在 windows ce、windows xp enbad、linux、自行开发操作系统等方案中选择了 linux 2.4 内核作为单片机的操作系统。

该系统及其硬件系统已于 2003 年 10 月底顺利完成并在全中国秋季医疗仪器博览会上展出，获得同行和客户的认同与称赞。但是，在开发过程中，由于对 linux 开发不够熟悉、与 linux 配套的应用软件 and 软件开发包较少等原因，为开发增加了不少难度。

【正文】

随着应用的范围不断扩大、需求日益复杂化、计算机技术不断发展，嵌入式系统的软、硬件资源也比以往更多起来，嵌入式系统开始逐渐承担更多更复杂的工作。比如，智能移动终端、机顶盒等等。这时，监控程序就不能满足需要了，嵌入式操作系统应运而生了。

通用操作系统由于是面向所有的计算机用户，而且所使用的内存、外存等硬件资源非常丰富，所以往往设计的比较庞大、复杂，功能上几乎无所不包，尽可能的适应所有用户的需求。嵌入式操作系统相比较于普通操作系统而言，只相当于最基本的内核部分，或者说只具备操作系统最基本的功能，比如中断处理、进程调度、存储管理等。这也是与嵌入式应用的体积小、重量轻的特点相适应的。一般来说，嵌入式操作系统具有如下特点：

(1) 稳定性：嵌入式系统常常需要在几乎没有人工干预的情况下完成高度专业化和实时的工作，这就要求嵌入式操作系统能必须性能非常稳定。

(2) 精炼高效：嵌入式系统中的硬件条件一般比较苛刻，可用资源很少，要在如此紧张的资源下很好地完成预定功能，要求嵌入式操作系统必须尽可能的精练、高效。而一些嵌入式环境中用不到的复杂功能则应该取消，以减小系统开销。

(3) 可裁剪：嵌入式操作系统作为操作系统，当然具有一定的通用性，能够适应多种平台和多种硬件环境。但是为了适应不同用户的不同应用环境而又能够避免冗余，就需要具有很好的可裁剪性能，使用户可以根据需要对各个组件进行裁剪和优化，得到一个有针对性的操作系统。

(4) 代码固化：嵌入式系统没有外存，其操作系统的代码需要固化在 ROM 中，系统加电后自动运行。

此外，嵌入式操作系统还常常具有其他一些特点。包括：启动速度快；具有实时性；能够适应多种处理器平台；一般提供配套的开发环境等等。

2003 年，我单位开发了新一代的数字 B 超诊断仪器 CTS—XXXX，作为单位的技术骨干之一，我主持并参与了 CTS—XXXX 软件系统开发的项目计划、需求分析、设计和测试阶段的工作。在选择嵌入式操作系统时，我综合考虑了功能要求、开发难度、研发生产成本、运行稳定性、兼容性等五个方面，在 Windows CE、Windows XP Enbad、Linux、自行开发操作系统等方案中选择了 linux 2.4 内核作为单片机的操作系统。

1、功能要求

首先是图像的实时显示，CE 在图像显示上显然比较薄弱，但是如果通过额外加入辅助硬件也可以达到要求，XP Enbed 和 linux 可以各自搭配 Dirext draw 和 Open GL，应付图像的实时显示绰绰有余，自行开发操作的系统功能比较简单，同样需要加入辅助硬件才可以达到要求。DICOM 3.0 方面，目前已经有了与 CE、XP Enbed 和 Linux 配套的开发包，自行开发操作系统只能通过 Dicom 标准里的算法自行开发或者购买辅助芯片来实现。工作站和数据库方面，CE、XP Enbed、Linux 显然都很容易实现，自行开发造作系统可以通过开发简单的文本数据库系统来实现数据库。CE、XP Enbed、Linux 都可以支持多种存储媒体，自行开发操作系统仍然需要开发存储媒体支持，并且还要符合现行各种媒体存储标准。四种方案对复杂的计算测量都可以实现。如果要考虑到以后加入三维成像功能，ce 和自行开发操作系统还是要加入辅助硬件才能实现。

2、开发难度

在开发难度上，CE 的图像显示由于要由硬件辅助，所以软件的开发难度降低，虽然增加了硬件的开发难度，但是由于以前单片机+程序的时候，图像实时显示也是由硬件承担，积累了丰富的开发经验，所以整体开发难度反而降低，而且由于 CE 的开发工具和软件包非常多，所以 CE 的开发难度是四个方案中最低的。XP Enbed 的难度主要是在 Direct draw 上，虽然微软一直推崇 DiretX、Direct draw，但是实际情况却是 Open GL 无论在网络资源、开发人员还是开发培训都要比 Direct draw 容易获得。Linux 的难度是对内核的裁剪以及开发习惯上。自行开发操作系统明显是难度最大的，不必赘述。

3、研发成本和生产成本

研发成本与开发难度密不可分，CE 的研发成本最低，自行开发操作系统难度最大，研发成本当然最高。XP Enbed 和 Linux 由于各自需要进行开发培训，二者研发成本不会相差很多，都居于中间。

在生成成本方面，CE 和 XP Enbed 都需要支付版权费用，CE 和自行开发操作系统都要加入辅助的硬件，而 Linux 既不需要支付版权费用也不需要额外的辅助硬件，所以生产成本是最低的。

4、运行稳定性

医疗仪器是否能正确稳定运行关乎患者的生命安全，CE、XP Enbed 都是微软公司专门用于嵌入式开发的操作系统，其稳定性和微软桌面操作系统系列不可同日而语，在工控开发同行中享有不错的口碑，Linux 其桌面操作系统就非常稳定，所以它的内核的运行稳定性更加不必质疑，在工控开发同行中一样享有不错的口碑，自行开发操作系统由于开发难度大，而且开发经验不足，所以难以保证产品运行的稳定性。

5、兼容性

实现同一功能的元器件，国产进口的价格相差很大，不同厂商之间价格相差很大，而且有的时候还会遇到器件停产和限制出口的问题，兼容性其实也是隐形的生成成本，所以，在选择元器件的时候，当功能性能稳定性符合要求的情况下我们会优先选择国产元器件。而目前我们使用的单片机处理器是日立公司的 SH4 处理器，可是如果国产的“龙芯”在功能性能稳定性上能达到我们的要求，我们会改用国产芯片来替代进口芯片。如果将来采用“龙芯”的话，CE 和 XP Enbed 就难以保证可以支持，自行开发的操作系统当然必须经过修改才可以支持不同的芯片，而 Linux，由于开放源代码，所以我们可以对其源代码进行修改以支持不同的芯片，而且，在北京和东北的一些高校实验室里，已经在这方面有了初步的结果。

该系统及其硬件系统已于 2003 年 10 月底顺利完成并在全中国秋季医疗仪器博览会上展出，获得同行和客户的认同与称赞。但是在开发过程中也遇到不少问题。

首先是 Linux 的开发经验不足，特别是 Linux 内核的裁剪和 Linux 驱动程序的开发，在我单位开发历史上是一片空白，各种网络资源也相对缺乏，为了让开发人员更快入手，缩短项目开发时间，在我的建议下，单位以不菲的价格聘请了工业控制界的其他公司、国外知名培训机构和高校教授为我们进行培训，在项目结束后，我还组织了参与本项目的软件人员根据本次的开发文档和开发经验，编写了单位内部的嵌入式 Linux 开发教材，防止日后因为人员的流动造成经验的流失。

然后是 Linux 的配套软件太少，比如书写文档的办公软件，各种分析设计的辅助软件，各种自动测试软件，版本控制工具软件，这些软件并不是没有，但使用的效果不够理想，而且有一些我们在 Windows 下习惯使用的软件并没有 Linux 的版本，使用同类软件的话需要花费额外的学习时间。于是我们都采用在 Windows 上安装 VM—WARE 仿真 Linux 的运行，通过虚拟网络将 Linux 和 Windows 互联，在 Linux 仅进行程序的运行调试和编译，其他工作都在 Windows 下进行。

最后是可用软件开发包极为缺乏，除了 C 语言的标准运行库和 Open GL，其他任何过程函数都需要动手编写，我们通过修改一些以前在 C51 下编写的函数和雇请软件公司为我们编写小型的过程模块，以减少开发工作量，确保的按时完成任务。

最后是可用软件开发包极为缺乏，除了 C 语言的标准运行库和 Open GL，其他任何过程函数都需要动手编写，我们通过修改一些以前在 C51 下编写的函数和雇请软件公司为我们编写小型的过程模块，以减少开发工作量，确保的按时完成任务。

以 Linux 内核作为嵌入式操作系统的作法在工业控制领域已经初露锋芒，由于其功能强大、生产成本低廉、可靠稳定、兼容性，必将拥有广阔的前景。希望能够在工控开发领域普及 Linux，增加 Linux 的交流与培训，而且，在开发过程中遵循开放源代码的思想，将自己开发和改进的应用软件 and 软件开发包、类库共享出来，共同促进以 Linux 为代表的免费、开放源代码的嵌入式操作系统的发展。

论企业应用集成

摘要:

本文讨论了某公司的应用系统集成项目。某公司为了应对市场变化的需要，决定把公司几个主要的应用系统 ERP 系统，PDM 系统，E-mail 系统集成在一起，系统集成完成后，ERP 系统可以与 PDM 系统交换数据，大大减少了重复工作。通过 ERP 系统与 E-mail 系统的集成，可以把 ERP 出来的报表自动发送给相关人员。我作为该项目的主要负责人之一，担任了系统分析和设计的工作，通过分析需求，设计三层体系结构，选择合适的平台等措施使项目能顺利完成。在项目实施过程中，我发现 XML 作为新的 WEB 开发语言，应是今后选择的一个方向，并且企业应用集成是一个不断发展的过程。

正文:

企业的应用集成 (EAI) 是指在企业范围内将多个应用系统的过程，软件，标准和硬件集成起来，使其成为无缝运作的整体。目前企业应用集成正越来越受到人们的重视。我在 2004 年 1 月参加了公司应用系统集成项目，作为项目的主要负责人，我担任了系统分析和设计的工作。项目首先是和相关部门的用户一起讨论系统集成内容，用户希望完成系统集成后能提供什么样的功能和服务。确定了这些需求后，就进行软件开发平台的选择和集成方案的设计。最后进行开发测试。测试完成后上线正式使用。整个项目用了五个月的时间完成，在 2004 年 6 月交付使用。

该企业的信息化程度比较高，主要的应用系统有：ERP 系统，实施了物流和财务模块，把公司的采购，财务集成到了一起；PDM 系统，产品的设计和开发在该系统上进行；E-mail 系统，主要是收发内部和外部的电子邮件。但是，随着企业的发展和市场竞争的激烈，问题也逐渐暴露出来。因为三个系统是独立的，没有数据的交换和共享，这样，大量相关的数据不得不重复输入。象 PDM 系统负责产品数据的维护。当一个产品的 BOM (Bill of material, 物料清单) 成熟后，要把该 BOM 导入到 ERP 系统。因为两个系统没有关联，这样就需要安排人员专门负责数据的录入，往往要加班加点，且容易出错，很不利于产品快速地生产并推向市场。为此，要实行应用集成，把企业的这些“信息孤岛”联系起来实现信息的交流。我在分析了企业应用系统的现状后，分以下几个步骤来实现该企业应用集成。

一、仔细研究现有的系统之间的关系，确定要集成的内容并考虑实现的方法

首先是 ERP 系统和 PDM 系统的数据的交换，要求在 PDM 系统开发出来的 BOM 下达后，要自动导入 ERP 系统。因为两者针对的是 BOM，结构是一致的，实现两者的数据集成，只需定义统一的数据接口和格式即可。其次是 E-mail 系统和 ERP 系统的集成。用户希望有 ERP 系统批准的采购订单能自动发送给供应商而不必手工要从 ERP 系统中下载下来整理后再发给供应商。驻外地的销售人员要求能每天收到关于成品库存的 E-mail。通过分析，发现公司用的 E-mail 系统是 Microsoft Exchange Server。而 Exchange 提供了与外界的 API 接口，只要照接口规定的格式填写，如发件人，收件人，信件标题，内容等，进行填写的文件，转送到了 Exchange Server 后，Exchange Server 就可以把将其发送到指定的收件人邮箱。利用这一功能，可以在 ERP 上开发接口程序，下载符合要求的文件到 Exchange Server 上实现 E-mail 自动发送。

二、设计企业应用集成的体系结构

为了避免传统的点到点的系统集成的缺点，我提出了三层的系统集成体系结构，即把企业应用系统分成表示层，中间层和企业信息系统层（包括数据系统），企业信息系统层由该企业的 ERP 系统，PDM 系统和 E-mail 系统组成，它们通过一个面向消息的中间件来实现数据的交换。中间层主要实现应用的业务逻辑和各种服务支持，它可以访问企业信息系统层运行且与应用相联系的数据和函数。比如前面说的 ERP 系统与 PDM 的数据交换，ERP 系统与 E-mail 系统的接口就在中间层实现。数据从底层进入中间层，在这一层完成数据交换和集成的功能。这样通过中间层就可以实现三个系统数据的共享。客户层则包括不同类型的客户端应用，之前三个系统都是 C/S 结构，各自有自己的客户端程序运行在个人计算机上，集成后，要开发出基于 WEB 的客户端程序，统一界面，方便访问。实现 B/S 与 C/S 共享。这样，通过中间层实现不同系统的协调工作和对企业各种事务活动的支持，屏蔽了低层的接口和技术细节，使数据能方便地交流。

三、选择合适的应用集成平台

目前，可作为开放式企业应用集成的规范和平台的主流技术有两种：一种是微软公司的 COM+规范和 Windows .NET 平台，另一种是 SUN 公司的 EJB 规范和 J2EE 平台。在平台的选择上我进行了反复的比较，最后选择了 J2EE 平台。因为 J2EE 平台的开放性与支持异构性，可移植性，支持的广泛性。对企业现有遗产系统的继承性和技术优势等。更重要的是它具有跨平台的功能。公司的 ERP 系统是运行在 HP Unix 服务器上的，PDM 系统用的则是 Microsoft NT 操作系统，E-mail 是 Microsoft 的 Exchange Server，这些应用使用不同的操作系统和平台，而微软的 .NET 只能用在 Windows 的操作系统的。所以基于公司的实际，J2EE 平台是合适的选择。选择 J2EE 平台存在的问题在于，一是成本相对比较高。二是公司的 IT 人员对 JAVA 还不熟练。为此，我在确定选用该平台后，先了解该平台的软件构成，只购买了我们需要的开发软件，从而节省了成本。对 JAVA 则安排了几次培训，使 IT 人员能很快地上手用 JAVA 开发程序。

四、应用集成方案的实践

在 J2EE 平台上，我通过 JAVA 在业务逻辑层的开发，实现了以下的业务流程。首先，在 PDM 系统的产品结构成熟后将会有有一个下达的动作，然后 BOM 数据通过中间层的处理自动导入到 ERP 系统，这能过之前已定义好的接口规范和数据格式可以很空易地实现。ERP 接收到产品结构数据后，即可以运行 MRP 产生采购订单和生产订单，投入产品的生产。此外，通过业务逻辑层，在 ERP 系统上下达 PO 后，采购订单的数据会自动传到 Exchange 服务器，然后通过 E-mail 自动发送给供应商。产品库存的数据也可以从 ERP 系统传到 E-mail 发给驻外地的销售人员。该应用系统集成后，用户不必再加班加点地在 ERP 系统中维护产品数据，并且保证了 ERP 的产品结构与 PDM 中的是一致的。由于实现了 ERP 与 E-mail 系统的集成，用户也不用担心哪张 PO 没有发送给供应商了。

项目上线后，满足了用户的需求，运行半年多来，系统基本稳定，大大提高了工作效率，得到用户和管理层的肯定。

但也存在着一些问题。在项目实施的后期，我们发现用 JAVA 编写代码效率比较低，运行速度慢，并且取数据还是不太方便。于是引入了 XML 技术进行数据的抽取，组织和表现，结果大大增加了开发的效率。通过该项目我认识到 XML 将会是今后 WEB 开发和实现企业集成的主要技术之一。此外，现在仅是实现了 PDM 系统和 ERP 系统的集成，之后随着企业的发展，ERP 将运行在更开放的平台上。会从企业的内部应用转为一个连接到 WEB 上的分布式应用系统。即 ERP/II，这将是以后的发展方向。

论企业应用集成

摘要:

2004 年 10 月,我参加了***车站综合信息平台项目的开发,承担项目的方案设计任务。该项目力图通过对车站现有信息子系统的集成,以达到共享各子系统的数据库,优化企业运输作业流程,提高企业经营管理水平之目的。

本文结合笔者的实践,以该综合信息平台建设项目为例,讨论了企业应用集成技术。在本着集成、开放标准、管理配套的原则下,提出了基于 Java 技术的 J2EE 应用服务器作为统一的应用集成平台,以集成适配器作为系统集成架构模式的总体设计思路,并着力介绍了该项目关键部件——集成适配器的构建过程。还就项目的具体实施作了详细叙述。最后,提出了企业应用集成的持续性,并确定了下一步集成的目标。

正文:

2004 年 10 月,我单位承接了***车站综合信息平台的建设任务。由于我具有多年的铁路行业软件项目开发经验,所以我有幸被单位指定为该项目的负责人,主要负责项目的方案设计工作。

该车站是一等客货运编组站,车站所在地是矿区,有规模不等的国有煤矿及个体煤矿数十个,车站主要以煤炭运输为主。近些年来,随着铁路 TMS 系统(铁路运输管理信息系统)建设的逐步深入,该车站建立了若干相应的应用子系统,主要有列车确报系统、车站现车系统、货票制票系统、车号红外线自动识别系统、货运计划系统、货运安全系统及货车轨道衡计重系统等。车站希望通过综合信息平台的建设达到以下几个目标:(1)实现各子系统间数据共享。(2)能够实时地向企业客户(货主)发布请车、承认车及货物运价调整等相关信息。(3)通过各子系统的应用集成,使车站运输作业流程得以优化。

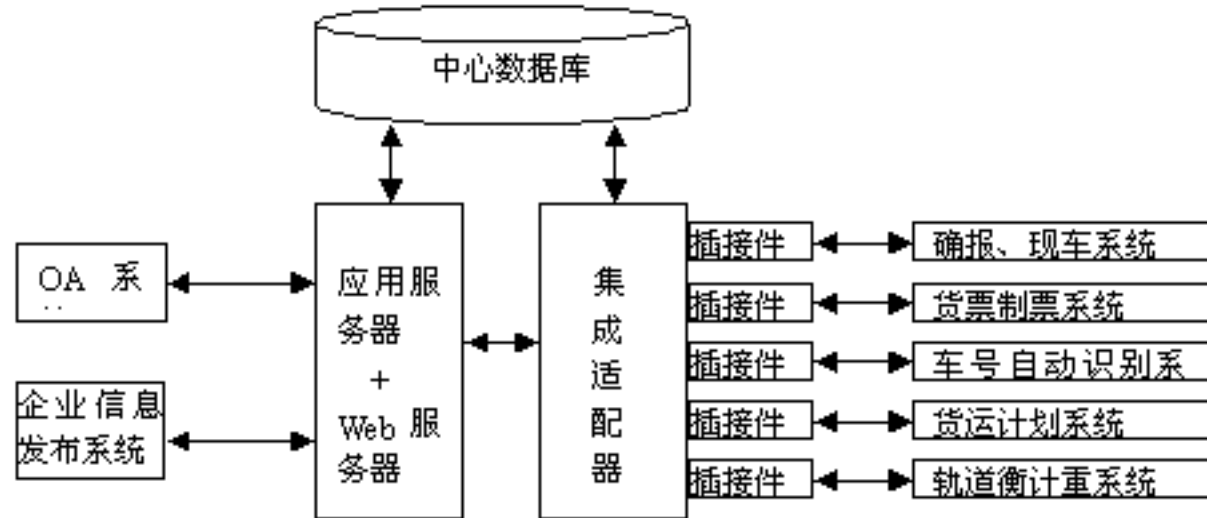
在设计综合信息平台建设总体方案时,我充分考虑了下面三个原则:

集成原则:综合信息平台最主要的目标是整合车站的信息资源,在考虑 最大化集成各个信息子系统时,应该避免产生新的“信息孤岛”。

开放标准原则:在综合信息平台建设时,应该站在整个系统宏观的高度,采用开放的标准和统一的架构来集成各信息子系统,避免各子系统“点到点”的低效落后的集成方式,以利于将来其他新的系统能够便利、无缝的整合。

管理配套原则:建设综合信息平台的目的是通过信息共享,达到业务流程的优化,以提高车站各方面整体管理水平。因此项目设计之时,应充分考虑企业管理方面的需求。

在遵循上述原则的基础上,经过对企业需求的认真分析,结合当今成熟的 EAI 技术,我提出了以基于 Java 技术的 J2EE 应用服务器作为统一的应用集成平台,以集成适配器作为系统集成架构模式的总体设计方案。如下图所示:



本设计方案从集成的广度来说,既包括了数据的集成,也包括了应用的集成;在集成的方法论方面来讲,大部分系统采用可白盒集成方法,少数系统采用的是黑盒集成方法。

根据综合信息平台的总体设计方案,集成适配器是该信息平台的最关键部分,它负责不同子系统之间数据的采集、转换和交流。因此,集成适配器的选型或设计的合理与否,是项目成败的关键。由于车站各信息子系统存在操作系统平台和数据库的异构性,无法从现有的中间件中找到完全适合的产品,因此我们决定自行开发此中间件。由于铁路各车站的业务领域和业务流程存在高度相似性,所以,此中间件有较高的复用价值。

对于集成适配器的设计,我采用了一种“可插拔”的设计理念。即为每个需要集成的子系统单独设计一个插接件,该插接件负责为与之相连的子系统提供数据及应用接口。各插接件通过 XML 格式的装配文件,自由组装到集成适配器这个容器中。集成适配器为所有插接件提供一个统一的调度模块,来协调和指挥所有插件,使之能够协同运作。

在集成适配器的开发中,我选用了开源的集成开发环境 Eclipse 作为开发平台。对于集成适配器调度模块的开发,我们采用了 Eclipse 提供的 Jobs API,Jobs API 封装了 JDK (Java Development Kit Tools) 的定时及同步方面底层 API,降低了编程的复杂性,提高了开发效率。在插接件的开发过程中,我们充分利用 Eclipse 的高度可扩展特性,在因特网上搜集该项目可用的插件,以这些插件为扩展点,来扩展我们自己的插接件。最后我们利用 Eclipse 提供的 RCP (Rich Client Platform) 技术,集成我们的功能插件,并生成独立于 Eclipse 平台的、可独立执行的集成适配器。

各信息子系统具体集成过程及效果如下所述:

(1) 轨道衡计重系统

该车站的轨道衡设在矿区与车站之间,距车站站场大约 2 公里左右,从矿区到车站的重车要经过轨道衡检斤,对于超重货车,要通知车站相关部门处理。由于该系统是个独立的单机系统,计重软件采用的是 VFP 编制,数据库是单机的 DBF。因此,我们首先从网络方面进行了集成:利用一对网桥将该系统接入车站 TMS 系统。重车通过轨道衡会产生结构为:(车辆顺位、车号、车种、铁重、计重)数据,与轨道衡相连的插件会通过 JDBC--ODBC 桥,将数据转换为两个不同格式的副本,一份写入中心数据库,一份作为矿区站发来的确报报文,写入确报系统的到确报库中。

通过该系统的集成,轨道衡工作人员免去了每天去车站递送过衡报表的工作,车站相关部门可随时通过浏览器查询过衡数据,并能方便的生成各种统计数据;车站的车号员通过查收确报报文,即可掌握过衡列车的组成内容,减去了每列车都要到现场抄车号,回去录入的工作流程,大大减轻了劳动强度,提高了工作效率。

(2) 货票制票系统

该系统的集成我们采用了批量文件传输(FTP)方式。货票制票机在制票完成后,会自动向分局传输货票数据。因此我们在货票传输地址表中增加一个条目,使货票数据在传送分局的同时也传送到集成适配器,与之相应的插件将到达的 80 列格式货票数据进行解析,形成结构为:(票号、车号、车种、计重、运价、发货人、收货人)的数据,利用 JDBC,一份写入中心数据库,另一份通过车号与现车系统的车辆库匹配,写入相应信息。

通过该系统的集成,减轻了编制出发列车确报的工作量,因为输入车号后,与该车号相关的货票信息会自动生成,免去了去查阅货票步骤。另外,该系统的集成,使货主通过因特网查询货票信息成为了可能。

其他的信息子系统如确报、现车系统、货运计划系统、车号识别系统等等,也都通过各自的接口插件进行了数据或应用的集成,在此不再详述。

(3) OA 系统和企业信息发布平台

通过 TMS 各子系统集成而建立的中心数据库,是办公自动化和企业信息发布平台的数据基础。通过设在货服大厅的大屏幕显示系统,能够实时动态地发布与货运业务相关的系统,方便了货主,提高了办公的透明度。

通过项目组成员努力工作,车站高层领导的高度重视,以及车站相关人员的通力合作,历时四个月,该车站的综合信息平台初步完成,达到了建设之初的需求目标,得到车站的一致好评。系统到目前为止运行稳定。

我们应该知道,企业应用集成是一个持续集成过程,是一项长期、不断进行的工程,不能指望短时间内达到深度集成。随着铁路信息化建设的深入,还会有新的应用需要整合。我们下一步的目标是建立该企业的企业门户,使我们前期的后台整合成果,能够在 Internet 上得以展现。

论基于 Web 服务的企业应用集成

摘要

去年,我所在的公司承担了一项为 X 卷烟厂建立企业应用集成的项目。主要目标是将该厂原使用的办公系统、人力资源系统、生产管理系统等系统进行集成,使之成为一个无缝连接的整体。我有幸参与了该项目,并担任主要的规划设计工作。

针对该项目,我们综合运用了 J2EE 平台构架、Web service、IBM WebSphere 等技术,来实现基于 Web Service 的企业应用集成,其效果相当显著。而且如果今后有新的系统加入,也能够很容易地添加。用户在使用该系统后,生产、经营效率有了大幅度的提高,得到了用户的一致好评。

正文

去年,我公司承担了 X 卷烟厂的企业应用集成项目。该卷厂是一家大型国有企业,经过多年的发展和积累,形成一套行之有效的管理机制,并实施了一系列的管理、生产自动化系统,如 LOTUS 的 Domino 办公自动化系统、人力资源管理系统、CRM 客户关系管理系统、用友财务软件等商品化软件,也有该厂与一家软件公司合作开发的生产管理软件(具有源代码)。这些软件实施于不同时间,也为各部门的生产、管理效率的提高起到了巨大的推动作用。但是各个分散独立的系统也给企业的发展造成了瓶颈,生产第一线的数据不能随时反馈给决策层领导,供应链、销售链不能与生产管理系统衔接,只能用手工方式将数据输入生产管理系统,不仅速度慢,而且经常容易出错。

随着企业的发展,这些系统也成为了一个个信息孤岛,给企业的进一步发展造成了瓶颈。为彻底消除这些限制企业发展的因素,该厂决定实施企业应用集成,将各个独立的系统集成起来,成为一个无缝连接的整体,使各种信息在整个企业范围内流动起来,促进企业的进一步发展。

我公司成立了项目组,我有幸参与了该项目,并担任主要的规划设计工作。

根据以往项目的经验,我们首先进行了技术方案的选择。目前企业应用集成目前通行的做法有三种:数据层集成、业务层集成和表示层集成。数据层集成需要修改大量的应用程序,而使用的系统中一些是商业软件,我们无法得到源代码。而表示层集成只是集成了各个应用程序的显示界面,就好像只是做一层外装修,无实质性的改变,而且有可能成为整个集成系统的瓶颈。业务层集成则克服数据层、表示层集成的这些缺点,而程序代码的修改量不大,且只是针对没有 API(应用程序接口)的应用程序。因此,我们根据企业目前系统使用的状况及我们以往开发类似项目的成功经验,经项目组讨论决定采用业务层集成方式来完成该项目。

我们采用了 J2EE 的平台构架,并为企业将来在企业间和电子商务的进一步集成,采用了基于 Web 服务的集成方案。Web 服务具有各个应用程序之间松散耦合、位置透明、协议独立的特点,某个应用程序的增减变动不会影响到其它应用程序的运行,并且它所使用的 SOAP、XML 等协议和技术也不被某一家公司所控制。

选定了实现平台,我们开始对各个系统进行封装、组合。各个系统的接口的异构性带来了访问上的麻烦。为了屏蔽接口上的差异性,我们采用了将这些接口映射为统一的虚拟组件。虚拟组件一边通过 J2EE 低层接口显示现有应用程序的功能,另一边通过现有应用程序接口 API 和它们进行连接。这样用户就可以用统一的方式访问虚拟组件而获得应用程序的功能。我们通过采用 J2EE 的 JCA 方式实现虚拟组件对各个应用程序的访问,为每一个应用程序定制一个资源适配器,并插入 J2EE 的应用程序服务器中。资源适配器含有一个指定的原系统库(用 Java 写的或带有自身接口单元的库),这个库同它描述的原系统具有连通性。通过 J2EE 平台的 JCA 这样一种方式,我们向用户提交了一个统一的调用接口。

解决了接口问题,接下来就要将各应用程序提供的服务通过 WSDL 描述服务,并且注册到企业内部的 UDDI 中心。我们采用了 Apache axis 1.0 来开发 Web 服务,IBM 的 WebSphere 作为消息服务器。根据该厂的要求,将各应用程序提供的服务分为 5 个种类来提供 Web 服务,分别是销售、人事、生产、供应、办公五个部分,再将各应用程序提供的 Web 服务具体细分,如生产部分提供了物料需求计划服务、库存服务、产品目录更新服务等。并 WSDL 实用程序生成代理代码,发布 Web 服务程序;在统一的服务发布服务器上,通过 UDDI 服务实现发现 Web 服务功能,便于查询系统各模块提供的 Web Service 服务。服务请求者按照标准接口构造 XML 格式的 SOAP Request 并通过 HTTP 方式向 UDDI 注册中心发出请求。当有一个新的物料需求计划生成时,我们通过使用 JMS 中间件来发布物料需求计划,使得订阅该主题的供应商得到消息。

企业信息门户 EIP 我们采用了 IBM WebSphere Portal Server 来实现。对用户帐号采用统一管理,统一授权,用户登录系统时采用统一认证服务调用接口以实现用户身份的认证过程,并通过单点登录 SSO,从根本上改变了现有各应用程序帐号之间进行漫游,不需要进行重新登录,给用户提供了极大的便利,并减少了不法用户利用认证攻击应用程序的可能,极大地提高了系统的安全性和可用性。

在系统实施过程中,我们也遇到了一些困难。

生产部门使用的生产管理系统是由该厂与一家软件合作开发的,没有预留 API,我们便对该系统进行了改造,由于该软件公司已找不到,而且该系统文档难以找到,因此我们对接口改造会对原应用程序造成的兼容性、稳定性的问题估计不足,改造完成后只进行了接口改造的回归测试,使得该应用程序一直不稳定。于是我们重新对该系统进行了深入透彻的了解,决定采用 CORBA 规范的接口定义语言 IDL 描述接口,而 J2EE 与 CORBA 又有极大的相关性,便于今后的封装、组合。接口改造完成后,我们进行了反复的回归测试和稳定性测试,才使系统又恢复原来的运行水平。

还有就是与各应用程序供应商的协调,大家都站在各自的立场上,设想对方的系统接口应该如何改,以适应自己的接口。经过我们和该厂的努力,最终大家都同意以最有利于集成系统的角度进行改造。

在本项目中,我们综合运用了 J2EE 平台构架、Web service、IBM WebSphere 等技术,来实现基于 Web Service 的企业应用集成,其效果相当显著。而且如果今后有新的系统加入,也能够很容易地添加。用户在使用该系统后,生产、销售、供应、办公成为一体化的无缝连接的一个整体,生产、经营效率有了大幅度的提高,得到了用户的一致好评。

我从实践中领会到,企业应用集成是一项技术复杂,工程繁杂的工程,但也是企业目前最急需解决的问题。同时我也认识到,企业应用集成还有很多领域和知识点需要继续研究和实践,新技术的应用也带来了新的要求和挑战,我将在企业应用集成领域不断探索,不断实践。

论软件需求分析方法和工具的应用

【摘要】

本文以某通信公司的业务报表系统开发为例，讨论了软件需求分析工具与方法的选择。我们认为，软件需求分析是软件工程中重要的一步，直接关系到后继工程进行以及最终的产品能否满足用户的需求，因此在整个工程中起着关键性的作用。采用适当的工具，有可能显著减少需求阶段的错误，也可大幅度提高需求分析的质量和工作效率。当然工具的选择应当与实际的项目相结合，充分地发挥工具的作用。本文结合我们工作的实际经历，简要讨论了开发系统时选用的工具及其应用，选用时所考虑的原则以及所碰到的问题。在文中也结合多种开发方法（即传统的瀑布法、信息工程法、面向对象的方法）的比较，指出各种方法的不足之处，说明我们所采用的工具对软件需求分析所起的作用，以及相应产生的效果。

【正文】

我在某市一家通信公司工作，作为一名技术骨干，受领导委托，参与了开发本公司的业务报表系统，我担任系统的需求分析，总体设计和部分代码的编写工作。

我所在的企业作为一家通信运营公司，分为总部、省级公司和地市级分公司三级，各级公司之间都有数据报表的要求，但是，每一个地市分公司因所处的地方不同，经营环境不同，所面临的问题也不一样，因此形成了各具特色的数据报表（除地市分公司向省公司汇报的之外）。公司又分设了许多部门，这些部门也都会需要数据，作为分析决策的依据。因此，了解各个部门的需求就成了业务报表系统的关键。

在调研的过程中，我选用了一种工具叫 PlayCASE，可以从网上免费下载，有很强的功能。下面就介绍一下，在需求分析阶段，我是如何使用这一工具的。

第一步，了解业务组织结构。公司内部的数据实际上是在部门之间流动的。业务部门需要知道在本地覆盖区内各基站的话务量、当天的话务量（即话务量的时空分布）。财务部门需要知道本月各类用户的话费收入、预交款收入、与其他电信运营商的网间结算等。计划部门需要各部门的分析数据。计费部门需要提供本月的账单统计数据、话单统计数据分布（比如分别按照基站分布、时段分布以及按用户类别分布）、预交款统计数据、当前的欠费总额分布、催缴情况等等。这些部门时常为了数据而产生了大量无谓的争议。在使用 Play CASE 工具时，先要将这些部门录入到 Play CASE 的“业务部门”中，构成了一个信息源的接收点（或发送点）；而 Play CASE 通过图示表示了这些部门的关系，并转换成了相应的软件结构。实际上，这是一种系统建模的方法，即把业务系统中的各个组织，转变为软件功能中的各个结构。这样，在需求分析阶段，明确哪些部门需要数据，从而保证了需求分析对整个公司的全面性，而不会忽略掉某一个部门，导致需求分析的不完整。

第二步，了解各个业务部门中的业务流程，使之通过 PlayCASE 转换成软件的运行过程，这是一种动态建模的方法。在上一步的基础上，追踪各个部门的行为，录入到 PlayCASE 中，并以形式化的语言描述各过程。对于复杂的过程，该工具不提供了进一步细化的方法，并且形成了业务流程图和业务状态图。根据这些流程图、状态图与实际业务部门的业务相结合比较，还是较为吻合的。在此步的实施过程中，运用了动态建模技术，使各部门业务流程的情况在软件的运行过程反映出来，从而保证了需求分析阶段中运行过程的描述能真实地反映实际情况，防止在后继的程序编写过程中，可能会经常发生的一类情况：程序员因为没有理解业务流程而出现“闭门造车”的现象，从软件的功能角度上保证了软件的正确性。

第三步，将业务数据转变为软件数据，这一步工作实际上就是收集各部门所需要的数据。分析各部门需要的数据都有哪些；以及数据是如何转换的，这可以归入“功能建模”的范畴。将这些相应数据录入到 PlayCASE 中，选定所属的部门。这时就自动地建立了 DFD 图（数据流程图），数据字典，省去了人工建立时的很大麻烦。

第四步，将业务上的数据关系转 为成软件中的数据关系，这里采用了面向对象的方法，把业务部门所需要的数据看作一个实体，部门间的数据关系就是实体之间的关系，比如：经营部门所需要的用户资料、用户话费，实际上就是用户这一实体与账单这一实体间的关系。Play CASE 提供了构件（不过我觉得是部件更为合适一些），来表示对应的数据，并提供了三种构件的表示关系即组装关系、分类关系与相连关系。这三类关系基本上反映出了现实世界中的业务数据之间的关系。例如现实世界中的用户资料与用户话费，在 Play CASE 中，可将用户构件与账单构件和相连关系表示，这种方法，实际上是借鉴了 OOA 面向对象的分析方法中的类、聚集、继承、封装等概念，能较好地反映出现实中的业务；同时，这一步的工作也为总体设计中数据库的概念模式设计奠定了很好的基础。

经历了上述四个步骤以后，利用 Play CASE 工具自动行成了软件需求规格说明书、初步的 DFD 图 and 业务流程图，为下一步的总体设计打好了基础。

使用 Play CASE 工具，使需求分析既能继承传统的结构化分析方法，又能吸收面向对象设计方法的优点。比如能把业务流程转变成为运行过程，业务组织转变成了软件的结构等都体现了这一点。而在运行过程中，对复杂过程的细分以及追踪则反映了传统方法中的自上到下分解的分析思想，这对于解决复杂系统的分析是很有帮助的。

通过使用，我觉得这个工具还是很不错的，因为它实际将以下四个方面的问题结合起来了：软件、业务、开发人员和用户。对于用户而言，PlayCASE 用图形化的方式显示出业务流程，使用户了解业务在软件中的运行过程，提供了将来验收软件时的依据，对于开发人员来说，使开发人员能更清楚地了解业务流程，不会再发生“因为不理解用户的需求而出现的闭门造车情况，从而导致开发出来的产品不符合用户需要”的现象。因此，PlayCASE 所自动提供的需求说明书能够很好地沟通用户与开发人员之间的理解，使他们都能对需求有共同的理解。

使用 Play CASE 工具后，使我们的需求分析取得了很好的效果，不但能自动地提供许多结果，如需求说明书等；还使需求的质量有限很大的提高受到领导的赞扬（领导不是学计算机的，但对公司的业务十分熟悉）；在后继的设计与维护工作中，我们感到工作似乎轻松了很多。

当然，该软件工具也有不足之处，一个突出问题是灵活性不够，一旦公司的部门或者组织机构发生变化时，整个设计都要重新来过。因此，在改进的过程中，我们在第一步过程预留了好多个虚拟的部门，以备将来进一步的扩充或者变动。

论分布式数据库的设计与实现

摘要

本文通过 xxx 高速公路收费系统（以下简称收费系统），来论述分布式数据库的设计与实现。收费系统是我公司近年来接的较为大型的项目，管理结构为三层结构：公司级、收费中心级、收费站级，各级之间即可独立的完成自身业务，又有自上而下的管理关系。收费中心、收费站均为三层 C/S 结构，公司级采取 B/S 结构。该系统的数据库也按照三层来设计，收费站存放本站的所有流水数据，收费中心存放所有数据，公司本部存放查询用汇总数据，收费站与收费中心使用事务复制来同步数据，而收费中心与公司本部使用快照复制来同步数据，并且使用分级的方法来测试收费站、收费中心与公司本部之间的数据同步。

在本项目的开发过程中，我担任了数据库的设计工作。

正文

2000 年 10 月—2001 年 12 月我公司开发了高速公路收费系统（以下简称收费系统），收费系统项目从管理层面分为三层结构：公司级、收费中心级和收费站级。

公司本部：供领导、运营部、财务部等业务部门了解业务情况、检查工作。B/S 结构各部门通过 Web 服务器查询数据库服务器，而公司数据库服务器定时要求中心数据库服务器复制汇总数据。

收费中心：收费系统的管理中心下达管理制度，管理数据到收费站，接收统计收费站的收费数据，上报汇总数据到公司，负责日常的管理工作。

收费站：具体进行收费的单位，收费车道的数据通过通信系统实时上传到收费站数据库保存、分类、汇总，并且实时传送收费中心下达的数据库管理，并通过通信子系统下载到车道收费机上具体实施。

系统采用三层 C/S 与 B/S 的混合结构，收费中心与收费站为三层 C/S 结构，而公司级为 B/S 结构。我在项目中担任了数据库的设计工作，负责数据库的设计、测试及实施。

数据库设计

此收费系统的结构较为复杂，分为公司级、收费中心、收费站三级管理结构，既可独立工作，又有管理的联系。数据实时传送到收费站数据库服务器，再实时传送到收费中心数据库服务器。在数据库设计方面我们按物理的分布也分为三层结构。

在收费站，根据系统的需求分析的结果，一辆车通过收费站时产生的最基本的数据有：通过日期、时间、车型、收费类型及收费金额，因为收费标准不轻易改变，考虑到我们采用的是专用的车道收费机，存储量较小，所以收费金额项在此处不计入数据库，上传至收费站数据库服务器后，可以用车数乘以此类车型的收费标准而得到。当车道收费机上传数据到数据库时，还要加上工班、车道及收费员信息，保证数据的唯一性，所以我们把日期、时间、工班、车道、收费员、收费类型、车型设为组合主键，为车辆流水数据。收费员下班后还要上缴实收金额，因此还要保存实收金额，包括日期、工班、收费员及实际收费金额，为工班收费数据。在此基础上，分析、汇总数据，得到以下几类数据：

业务类型数据：车辆流水数据、工班收费数据、车道开通情况、收费员上班情况；

扩展的数据：为了查询、打印的方便、高效，流水表经过分类汇总，产生了以车道分类统计的车流量表（日、月）、以收费员来分类统计的收费金额表（日、月）及不收费车辆统计表；

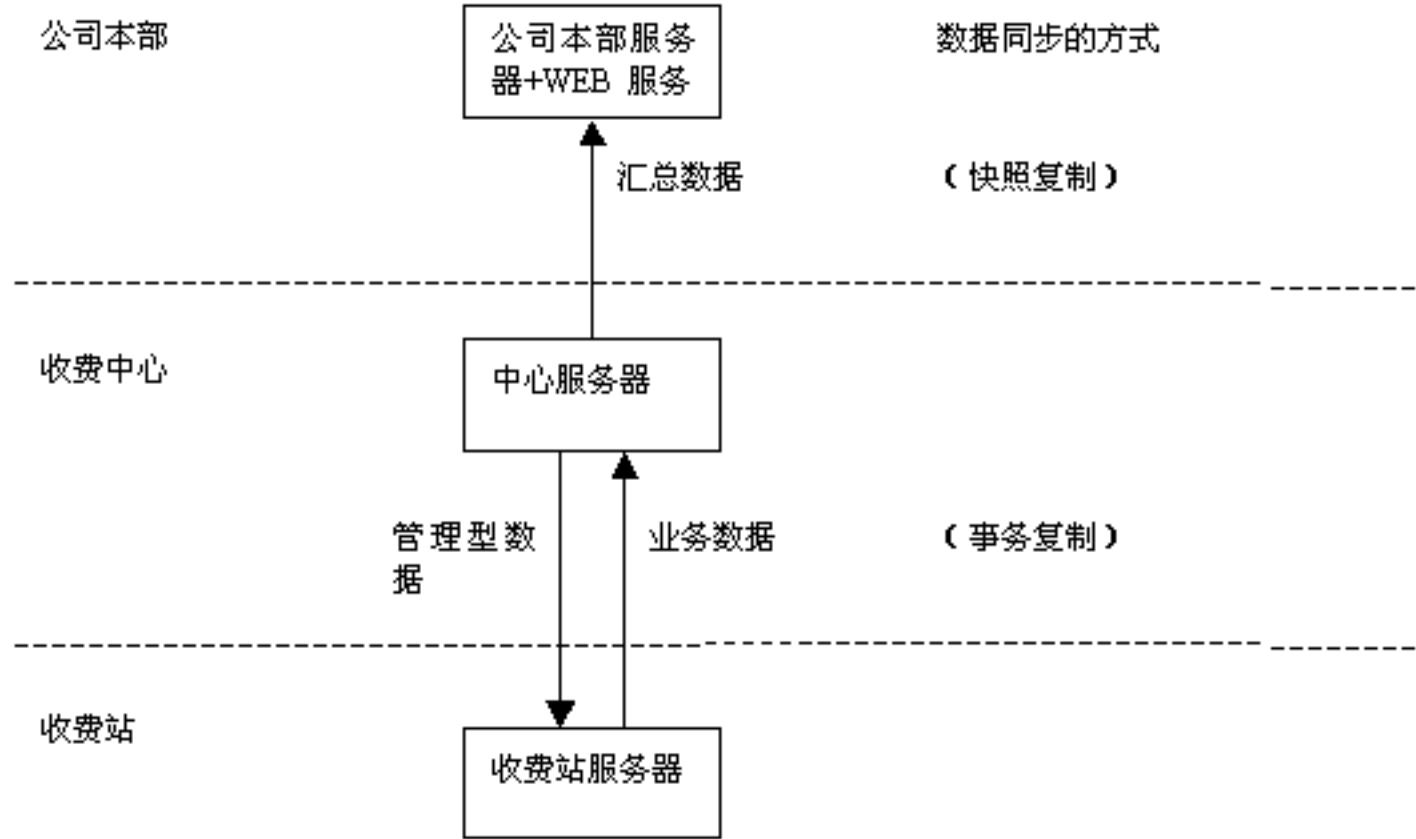
管理类型数据：收费标准、收费员信息、收费站信息、车道信息、工班信息、收费类型信息；从全局应用的角度出发，各收费站存放本站的数据，收费中心的数据库则存放所有数据，并对数据进行完整性和一致性的检查，这种做法虽然有一定的数据冗余，但在不同场地存储同一数据的多个副本，能提高系统的可靠性、可用性，使系统易于扩充，也提高了局部应用的效率，减少了通讯代价，同时也使得各处理机之间的相互干扰降到最低。

公司总部存放所有收费站的汇总数据，通过浏览器进行查询。

数据的分布

（1）在收费中心数据库服务器与收费站数据库服务器的数据关系中，由于收费站的数据是收费中心数据的子集，我们采用了水平分片的方式，通过并运算实现关系的重构。

（2）在收费中心数据库服务器与公司总部数据库服务器的数据关系中，数据是按照其应用功能来划分的，所以我们采用了垂直分片的方式。



数据分布在多个地点，为了保证数据的一致性及完整性，我使用了事务复制和快照复制两种数据同步的方式，在收费中心与收费站之间使用事务复制，而在收费中心和公司总部之间使用快照复制。

对于业务类型的数据，收费站在本地存放收费车辆的实时数据，而用户也要求收费中心也要实时的收费车辆数据，延迟不超过 2 秒，所以我采用事务复制进行业务数据的同步，收费站只需将更新的数据发送到收费中心的数据库即可。具体过程如下，把收费站的数据库作为出版者和分发者，收费中心的数据库作为订阅者，对收费站的数据建立快照代理，并在分发数据库中记录同步状态的信息。每一个使用事务复制的收费站数据库均有自己的日志读取代理，运行在分发者上并连接出版者。分发代理的任务是将分发数据库中保持的事务任务直接推动到订阅者。当推订阅被创建时，每个为立即同步而建立的事务出版物通过自己的分布代理运行在分发者上并与订阅者相连。

而管理型的数据是在收费中心设置，虽然修改的频度不是很大，但在修改后即要发生作用，所以也采用事物型复制，收费中心为出版者，收费站为订阅者。将管理型的数据发布到各收费站。

由于公司总部不需要实时更新，所以收费中心数据库服务器与公司总部服务器之间的数据同步设置为快照复制，公司总部数据库中建立收费中心表的快照，对这些数据的修改在收费中心进行，把收费中心数据库服务器设置为出版者，出版物为汇总数据，公司总部数据库服务器设置为订阅者，快照代理将准备包含有被出版数据表的结构与数据的快照文件，在分发者上存储这些文件，并在分发者的分发数据库中记录同步任务。在收费中心数据库服务器上建立一个作业，设置为每天早上 8 点更新一次（收费日的天为 8 点到第二天 8 点），公司总部就能得到截止到前一天所有收费站的汇总数据。

测试

数据库设计好了，就要对它进行测试。我们的测试策略为分步测试：首先测试收费站数据的正确性及完整性，在多台收费机上同时输入几组车型、收费类型的数据，查询数据库的流水数据是否正确，再看汇总数据是否正确；收费站正确后，再测试收费中心的数据的正确性、完整性及延时；再测试公司总部的数据正确性、完整性。

一开始我们把所有的表都建立在一个出版物中，但在测试中我们发现，由于收费站一个表的错误而造成的复制的中断，经常要重新配置复制，所有的表都要重新选择及设置一遍，非常繁琐，我们采取一个表建立一个出版物的方法来简化操作，只要恢复出错的复制，其他的复制仍然能正常执行，而且哪一个表的复制发生中断也很明确，不仅简化了操作，也加快了处理时间，就是日后用户维护起来也简单明了。

在设计过程中，基于查询及安全性的需要，我们大量的使用了视图，第一加快了查询速度；第二也防止了人为因素造成的数据的更改。

总结

按照以上的设计方案实施后，完全满足高速公路系统对数据实时性和完整性的要求，系统目前只在内部使用，而以后全省的高速公路收费系统实行联网，在外网上发布信息，那时数据的安全性及查询的响应速度将是我们考虑的重点。

改进 Web 服务器性能的有关技术

【摘要】

一个大中型的图书馆信息系统涉及到许多方面的技术与方案，本文着重讨论与 Web 服务器性能有关的一些内容。

本人有幸作为项目负责人之一参与了某大型图书馆数字化信息系统的设计和基于 Web 应用软件的开发工作。由于在数字化图书馆信息系统中流通着的大多是数字化的索引、文摘、全文、图像或音频视频等多媒体信息，对 Web 服务器性能有着较高的要求。

结合实际工程经验，本文将从硬件实现手段（缓存服务器、均衡负载设备、Web 双机镜像、CPU 和网卡的提升、网络带宽扩充）和软件实现手段（三层 C/S 软件结构设计、应用程序部署）等两个大方面论述如何提高 Web 服务大路的性能，以便使用户能够更快捷、高效、安全地使用应用系统。

【正文】

随着 Intranet 信息技术的发展，图书馆为了更好地发挥其图书流通、资料检索和学术交流的职能，图书馆的数字信息化工程也势在必行。某图书馆为了尽快地步入世界先进图书馆的行列，已经启动了一部分的数字图书馆工程。

该数字图书馆工程主要包括对外信息 Web 发布系统，交互式检索网，后台馆藏信息管理系统、多媒体资料采集制作以及外 VOD 点播系统等。本人有幸作为项目负责人之一，参与了整个数字化信息系统的总体设计，并参与了基于 Web 的一些应用（如对外信息发布系统、图像/全文混合系统、VOD 点播系统的）开发。

某图书馆数字化信息系统从网络环境上讲，主要划分为多个网段：（一）Intranet 接入部分，采用 2M 的 DDN 专线；（二）公共网段（非军事区），主要包括前台发布数据库服务器、Web 服务器、E-mail/FTP/DNS 服务器、检索服务器及 SAN 网络区域存储设备；（三）是内部局域网，包括内网 Web 服务器、后台馆藏数据库服务器、OA 服务器等。（四）是 VOD 点播专用网，包括音频视频点播服务器等。由于制定了严格的网络级和应用级访问权限，通过具有三层交换能力的高性能交换机和安全授权认证系统等，有效地控制了访问权限，确保了数据的安全性和完整性。考虑到经费和人员素质及今后的维护管理运营等方面，操作系统采用 WindowsNT 平台，服务器选用 DELL 高端的系列，数据库采用 IBM 的 DB2。主干网为千兆快速交换式以太网，局域网百兆到桌面，VOD 点播网十兆到桌面。

在该网络环境下应用主要分为三大部分：（一）对外 Web 发布系统、对外图书辅助检索系统；（二）后台馆藏信息管理系统和图像/全文混合检索系统；（三）VOD 点播系统。由于绝大部分应用采用 Browser/Server 方式结构，最终用户在本地只需安装 IE 或者 Netscape Web 浏览器，在后台数据库服务器的支持下通过网页方式请求和访问各类应用服务。另外，由于在图书馆信息系统中流通的多为索引、摘要、全文或音频视频等多媒体信息，对 Web 服务器性能与网络带宽等有更高的要求。

通过不断地试验和实践，我们发现从以下几个方面可以相对有效地提升 Web 服务器性能：

（1）缓存服务器和均衡负载设备使用可以缓解访问瓶颈，提高网络带宽、实现均衡负载。

缓存服务器也称为 cache 服务器，可以存储 cache 静态的内容如网页、多媒体点播资源和会议实况（已压缩的、有一定格式要求的）等。此外，目前美国 cashflow 缓存服务器，已经可以存储 cache 数据库、ASP 等动态内容。cache 服务器通常放到防火墙之外，外网 Web 服务器之前，因此 Internet 用户点击网页不再直接访问网站 Web 服务器，而是访问 cache 服务器。

由于 cache 服务器具有多个 CPU 和高速大容量 I/O 通道，独立的 OS，因此能大大缓解 Internet 访问瓶颈，而且也具有一定的抗黑客攻击的能力。

目前某图书馆采用这种方式，把大数据量的静态图片、点播资源、虚拟三维应用等都事先置放在 cache 服务器中，即使现今只有 2M internet 的接入带宽，以上应用的播放速度和效果仍能让用户满意。

另外一种方式采用均衡负载设备或 Web 双机镜像。这种方式通过负载均衡的方法达到 Web 访问性能最优。Web 双机镜像较早以前流行的方式，虽能使系统可靠性提升，但由于双机总是在互相询问对方状态，将会影响一定的访问性能。均衡负载设备是独立于 Web 服务器的硬件，它和 Web 服务器及网站中其他服务器接在同一交换机上，通过负载调度程序为各个服务器分配工作量，从而，能达到充分利用资源，提高访问性能的目的。只是由于某图书馆目前对外发布资源相对仍较少，只有用了三台 Web 服务器，因此目前的均衡负载设备作用还不显著。

（2）从 Web 服务器的配置来看，Web 服务器自身 CPU 个数及速度、网卡数量、Web 服务器与防火墙的位置关系等，都会影响到 Web 服务器的性能。

从 Web 服务器硬件本身来讲，CPU 个数的增加、网卡个数的增加、I/O 信道的扩展无疑可以直接地提高 Web 服务器性能。此外，由于千兆口的防火墙目前较少且费用较高，如果把 Web 服务器放置防火墙之后，一定会大大影响 Internet 访问性能。某图书馆采用 IDS（入侵侦测）+Web 服务器（服务器防火墙，较低端，不会影响流量）+应用服务器+数据库服务器（防火墙，高端），分层次的安全模式，既保证了系统的安全模式，既保证了系统的安全性，又提升了网络访问性能。

另外，某图书馆还采用了 SAN 网络区域存储来提高服务器访问速度。

（3）三层 C/S 软件结构设计和应用程序的适当部署也会提高 Web 服务器的性能。

将业务逻辑、通用访问接口与数据等相互分离、分别置放于 Web 服务器、应用服务、数据库服务器上，通过过程功能逻辑的合理部署，也能大大改进 Web 服务器性能。

一般的原则是，Web 服务器只需接受 Internet http 访问请求，使 Web 只有最少的任务，把实际处理交给各个应用服务器处理，然后返回结果给 Browser。某图书馆采用这种方式专门开发了搜索引擎应用服务器和混合检索应用服务器等，达到了良好的应用效果。

事实上，Web 服务器的性能提升还存在很多手段和方法，比如 CPU 与存储之间关系，Web 交换机等等，有待于我们进一步的实践、分析和讨论。

