

摘要

在世界各国积极倡导的“信息高速公路”的五个应用领域中，电子政务被列为第一位。政府信息化是社会信息化的基础，电子政务已成为世界新一轮公共行政管理改革和衡量国家以及城市竞争力水平的标志之一。与信息技术发达的国家相比，我国的电子政务无论是在理论研究还是实际应用方面均存在着较大的差距。尤其是在电子政务的建设中，面临着如何将政府现有的各个应用系统进行有效地集成来搭建电子政务体系架构的难题。

本文依托台湾、大陆地区几个大型项目，有着明确的应用背景、实用背景和实际基础。研究之目的在于解决并实现将政府的现有应用系统集成为电子政务的应用架构，以后保护政府已有的投资，加快政府进行电子政务建设的步伐。

本论文的主要贡献在于以下几点：

1. 从理论上对建立一个完整的分布式、跨平台的电子政务架构进行了分析，在此基础上提出了基于 J2EE 集成平台 UniFlow 工作流管理系统，作为电子政务架构基础平台的方案；
2. 在 UniFlow 工作流管理系统的集成设计中，作者从状态图及有限状态机理论中引申设计出了适合于电子政务架构基础的基于事件驱动的状态图（State Diagram）模型，克服现有的建模方法不能描述复杂业务过程处理的缺陷；UniFlow 的组织架构采用新型的矩阵制组织结构，将政务部门的静态部门树型结构和动态工作组模式相结合，使组织模型和过程模型灵活地集成起来，克服了传统组织模型的单一、缺乏弹性等缺陷。
3. 提出基于 UniFlow 的信息资源集成设计方案，采用先进的 XML 技术，解决了离散的信息资源所形成的信息孤岛现象。
4. 提出了基于 UniFlow 工作流管理系统平台的电子政务门户（GIP，Government Information Portal）架构集成方案：通过对 DocMan 公文管理系统、E-Form 电子表单系统和 Ucan 报表系统的集成分析和集成设计，最终实现了基于 UniFlow 并以 GIP 作为统一入口的电子政务应用架构，
5. 就基于 UniFlow 工作流管理系统的电子政务应用架构尚需解决的一些问题进行了分析并提出了自己的一些看法。

关键词：电子政务，工作流，架构，集成，电子表单，GIP，UniFlow，E-Form，Ucan

Abstract

E-Government Affairs ranks the first among 5 applications of "Information Superhighway" which countries all over the world advocate actively today. The government information is the basis of the social information. The E-Government Affairs has been one of the marks which is the new round commonality administration management in the world and scale the country or city' competition power. Compared with the information developed countries, the E-Government Affairs in our country have obvious differences whether in theory research or the practice application. Especially, we are facing with how to integrate each application system to set up the E-Government Affairs architecture. The dissertation based on several important projects in Taiwan and mainland areas has definite application background and practical background. The motive of this dissertation focus on resolving and realizing to integrate the application systems in existence as the E-Government Affairs architecture. Not only is the old investment protected, but also dose step of the E-Government Affairs construction quicken.

The main contributions of the dissertation as following:

1. To set up the distributed and different platform E-Government Affairs architecture is analyzed. Then, the UniFlow Workflow Manage system as the E-Government Affairs architecture's base platform based on J2EE is put forward.
2. The author designed the State Diagram which based on the event driven which is suit for E-Government Affairs architecture which is cited from State Diagram and the finite State Machine theory. Thus, the limitation of the present modeling methods which is unable to describe complex operation process can be gotten over. The UniFlow's organization architecture adopts new matrix organization architecture and combined with the static department tree structure and the dynamic workgroup pattern. So, the organization model and the process model are integrated neatly.
3. The information resource integration scheme which adopt the advanced XML technology based the UniFlow is bring forward and the "information isolated island" is solved.
4. The E-Government Affairs GIP based on the UniFlow Workflow Management System is put forward. The GIP scheme is realized by the integration of the DocMan, the E-Form and the Ucan.
5. Some problems in the E-Government Affairs system have been discussed and some new opinions is supposed and analyzed.

Key words: E-Government Affairs, Workflow, Architecture, integration, GIP, UniFlow, E-Form, Ucan

第一章 概述

§ 1.1 引言

人类进入 21 世纪以来，信息网络化正在延伸到社会的每一个角落，这种趋势已成为一种不可抗拒的时代潮流。全球性的网络化、信息化进程深刻地改变了人类的生存方式：一方面，Internet 技术应用以及电子商务的飞速增长给人们生活工作的各个层面带来了深刻的影响；另一方面，信息网络时代的到来给国家政府部门的工作提出了快捷、高效的要求。网络对于政府来讲不仅是一种高效的通信手段，而且是一座沟通政府与社会各界的桥梁。从全球范围来看，推动政府部门办公网络化、自动化、电子化，实现信息共享已是大势所趋。

在各国积极倡导的“信息高速公路”的 5 个应用领域中，电子政务 (e-Government Affairs) 被列首位。而针对电子政务的概念，目前有很多种说法，例如：电子政府、网络政府、政府信息化管理等，人们常常将电子政府和电子政务相混淆。所谓“电子政府 (e-Government)”，简单地说，就是政府职能上网，即政府为了在工作和管理方式上适应网络信息时代，在对政府传统的组织结构进行重组和流程再造的基础上，利用现代计算机和通信网络技术，将政府部门内、外部的管理和服务的职能放到网络上完成。而所谓的电子政务是政府机构应用现代信息和通信技术，将管理和服务通过网络技术进行集成，在互联网上实现政府组织结构和工作流程的优化重组，超越时间和空间及部门之间的种种限制，面向社会提供优质和全方位的、规范而透明的、符合国际水准的管理和服务。可见，真正的电子政务绝不是简单的“政府上网工程”，更不是为数不多的网页型网站系统。严格地说，它是更广义上的政府办公自动化或者理解为面向社会的政府办公自动化，其本质是“以网络为工具，以用户为中心；以应用为灵魂，以便民为目的”。

因此，从广义的角度上说，电子政务实际上是一个系统工程，它的内容十分广泛，国内外有着不同的内容规范。根据我国政府所规划的项目来看，电子政务主要包括这样几个方面：政府间的电子政务 (G2G)；政府对企业的电子政务 (G2B)；政府对公民的电子政务 (G2C)。归纳起来，电子政务所强调的是：各个政府职能部门之间、政府与公众、政府与企业的协同工作；在保证安全性的前提下具有开放性；注重集成政府的各种业务和资源。

据统计，一个国家政府所拥有的信息资源占整个社会资源的 80%。由此可见，政府的信息化建设对社会信息资源的开发与利用起着主导作用。目前，信息网络技术的发展使得政府机构拥有、生产、使用与传送信息的方式都发生了深刻的变化，这些都对政府的管理水平和服务功能提出了更高的要求。而电子政务的目的正是政府机构应用现代信息和通信技术，将管理和服务通过网络技术进行集成，并在互联网上实现政府组织结构和工作流程的优化重组，超越时间、空间与部门分割的限制，全方位地向社会提供优质、规范、透明、符合国际水准的管理和服务。

总而言之，电子政务的发展不但使政府能够有效地驾驭行政管理的职能并促使社会经济从整体上发展，而且能够充分地推进全社会民主化进程。

§ 1.2 国内外有关电子政务的发展及研究现状

一、发达国家电子化政务发展及研究现状

美国在克林顿执政期间就已经开始积极倡导和推动电子政务的建设。目前，美国国家政府以及所有的州级政府全部实现上网，而且几乎所有的市级政府部门也已经建立起自己的网

站。美国政府网站的内容非常丰富，据初步统计，目前已能够提供 27 种功能。使普通用户可以快捷获取各种有关信息。美国在 2003 年 10 月以前实现政府办公的无纸化作业，以使美国公民与其政府的互动关系实现电子化。

据英国国家统计局的报告显示，目前英国的成年网民中，有 18% 的人在使用政府机构网站获取服务或官方文件等信息，政府机构网站的总数已达 1000 多个，每星期的访问请求超过了 2000 万。2000 年 12 月开通的“英国在线”，已经整理了 1000 多个政府机构网站的信息，帮助用户迅速、方便地寻找所需网站，以获取关于就业、理财、旅行、生活等方面的政府信息与服务。

新加坡从 1981 年开始发展电子政务，目前其电子政务的发达程度已备受世人瞩目，许多国家都效仿新加坡的电子政务发展模式。新加坡的电子政务应用每年可为其政府节省大量的资金。据了解，到 2001 年 7 月，新加坡电子政府已可以为其公民提供 200 项以上的电子政务服务。

为了加速政府信息化的进程，日本政府于 2000 年 3 月正式启动了“电子政务工程”，其主要内容是通过各网络系统办理各种申请、申报、审批等手续，实施政府网上采购计划，该工程现已全面投入使用。目前，日本政府可在网上办理申报、申请、审批等 3000 多项业务，政府网上采购计划也将全面实现。另外，日本政府将在 2005 年前使政府各部门的主要业务全部通过互联网进行，这无疑标志着日本政府将全面进入办公电子化、无纸化的时代。

综观发达国家电子政务发展的状况，我们可以看到：电子政务的发展应与政府改革紧密地结合；在电子政务开发中，国家政府需要制定了统一的规划和技术标准，以此来规范电子政务的发展；注重实际应用，并把为企业、公众服务、实现资源共享放在重要地位；在具体实施方面，发达国家普遍实行分阶段实施的策略，由简单到复杂，由易到难。

二、我国电子化政务发展及研究现状

早在 1992 年，国务院办公厅就提出建设全国行政首脑机关办公决策服务系统的目和具体实施方案，并在全国政府系统推行办公自动化。2001 年，国务院办公厅又制定了全国政府系统政务信息化建设的 5 年规划，对我国政府信息化的指导思想、方针、政策等做出了明确规定。

然而，与其他发达国家的电子政务相比较，我国的电子政务才刚刚起步，电子政务的理论和政策研究应用还处于探索阶段。一方面，我们的政府信息化还处于初级发展阶段，立法滞后致使电子交易、电子签名和电子支付等合法性缺乏法律依据；另一方面，我们与主要信息技术发达国家的差距很大，缺乏统一的建设依据和安全标准，难以统一规划。但随着信息技术的不断发展，我国对推进电子政务、加快政府信息化建设高度重视。针对我国目前的现实情况，如何解决跨平台、跨异构的应用软件系统来建立电子政务应用架构是实施电子政务的主要瓶颈之一。

§ 1.3 研究的背景

云集软件网络(中国)有限公司是台商在中国大陆投资最大的软件公司，该公司一直从事于政府企事业应用软件的开发，公司在不同时期先后开发出了 UniFlow 工作流管理平台、E-Form 电子表单系统、Ucan 报表系统。其中，UniFlow 工作流管理系统，是作为 Oright 办公自动化软件的核心平台，它主要包含以下功能模块：流程定义与优化，工作流引擎，任务管理，流程监控四个核心部分；E-Form 是一个安装在服务器上，用来建立任意的 WEB

表单并能自动处理表单数据的电子表单系统；Ucan 表单系统为一种电子表单设计定制工具，在图形化界面下，提供一系列表单制作元件，使程序员及用户能轻松制作出符合业务需求的电子表单。为了顺应时代的潮流，公司决定将过去和现在正在开发的产品由我们进行集成开发出电子政务应用产品。因此，本论文的研究与以下几个项目有关：

1. 工作流管理系统 UniFlow
2. Web 化电子表单 E-Form
3. 一种快速报表工具 Ucan
4. 云集软件网络（中国）有限公司之产品 DocMan 公文管理制作系统
5. 台湾阳明交通大学公文管理系统
6. 台湾行政院主计处公文管理系统

§ 1.4 研究内容及各个章节组织

一、研究内容

本文较为深入地探讨并设计实现了基于 WEB 的分布式电子政务集成架构：将各个应用系统（UniFlow、E-Form 和 Ucan）集成在以电子政务门户（GIP）作为统一入口、以 UniFlow 工作流管理系统为统一平台的集成体系，通过各类应用系统彼此作用、相互链接，形成了一个有机的数据、信息流处理体系。

二、各章节组织

第一章 概述

第二章 电子政务应用架构的分析

从电子政务的应用出发，提出了电子政务架构集成的原则、目标，并从技术理论上对应用系统集成方法进行了分析。

第三章 电子政务的基础平台—UniFlow 工作流管理系统的工作流设计与实现

本文通过对工作流和工作流管理系统的有关概念分析，提出了将 UniFlow 工作流管理系统作为电子政务架构的基础平台的方案，然后对 UniFlow 工作流管理系统及其相关的模块进行集成分析、设计，实现了 UniFlow 的集成。

第四章 电子信息资源库的集成

本文通过对政府现有的资源状况进行分析，采用电子政务各个应用系统的程序代码跨数据库平台和电子政务各个应用系统的信息转换方法，提出基于 UniFlow 的电子政务系统信息资源集成设计方案。

第五章 电子政务门户(GIP)集成的设计与实现

通过对 E-Form 电子表单应用系统和 DocMan 公文管理系统的集成必要性进行分析，设计并实现了将 E-Form 电子表单应用系统和 DocMan 公文管理系统集成在 UniFlow 工作流管理系统基础平台上的 GIP（政府信息门户）。

第六章 总结与展望

对论文进行了全面的总结，并针对论文的不足之处提出了自己的一些观点和今后所要进行深入研究的内容。

第二章 电子政务架构分析

§ 2.1 电子政务应用分析

根据政府机构的业务形态来看，我们通常把电子政务应用分为三个应用领域：（1）政务信息查询：面向社会公众和企业组织，为其提供政策、法规、条例和流程等诸多方面的查询服务；（2）公共公务办公：借助互联网实现政府机构的对外办公，如申请、申报等，可以提高政府的运作效率，增加透明度；（3）政府办公自动化：以信息化手段提高政府机构内部办公的效率，如公文报送、信息通知和信息查询等。其业务模型可以用下图表示。

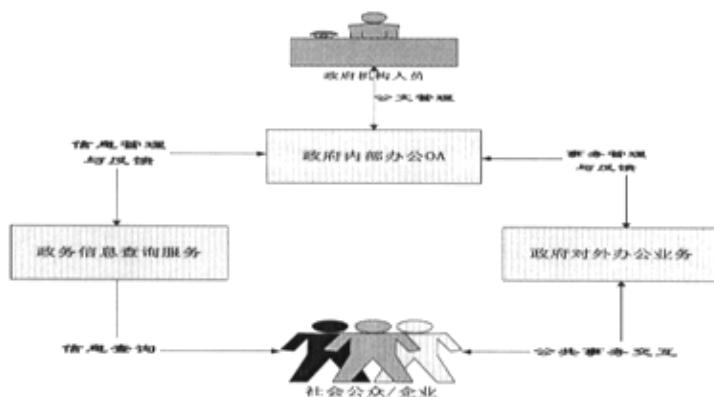


图 2.1 电子政务系统应用模型

从上图 2.1 可知，电子政务应用主要存在着三种信息流：

政务办公信息流、公共事务信息流和咨询政务信息的信息流；其中前两种信息流属于业务信息，而最后一个数据信息。所以，作为一个完整的电子政务系统，它具有以下的特征：首先，电子政务基于网络并符合 Internet 技术标准，并且是一个跨机构的、一体化的、支持前台（政府信息门户网站）和后台（包括政府内部管理信息系统、电子办公系统、数据库、安全平台和业务平台以及决策支持系统等）无缝集成的智能化综合系统；其次，它具有各种应用支撑系统，能在互联网上全面实现以公民为核心进行公共管理事物传递服务的网上自动化办公系统，并满足信息交流的实时双向性要求，同时能传递高度个性化信息服务的政府信息门户 GIP（Government Information Portal），GIP 是“政府门户网站”或“一站式”服务网站；最后，它是一个能够实现政府资源计划（GRP），支持政府科学决策，体现协同管理和客户关系管理（CRM）的信息管理系统。不仅如此，它还是一个和数字签名、公钥基础设施相配套的，能保证信息交流、服务传递以及在线支付安全的可靠系统。

由此可见，统一的电子政府希望能把政府的服务全部融入到整个国家网络中，实现一种无缝集成的“一站式”服务。一站式服务框架包括工作流引擎、通用政务构件、个性化管理模块、客户端模块，以及服务集成模块等。通过对具体政务应用资源的集成，对公众提供统一、标准、方便、快捷的电子政务服务。

§ 2.2 电子政务架构分析

为了对各个应用系统进行有效地集成以建立电子政务应用架构，就要从以下六个方面进行分析。

一、建立架构集成目标

根据目前电子政务技术的发展及政府部门的实际业务及今后拓展的需要，应该采用浏览器/服务器与客户机/服务器体系结构相结合的多层应用结构，使用当前先进的系统平台、面向对象的开发方法、可视化的开发工具，开发出一套较为完善的适用于多方需求并通过网络收集、发布各种信息的集成方案。

电子政务集成目标有：

- (1) 先进性和成熟性：采用目前比较成熟的先进技术和产品（如微软公司的开发平台、J2EE 和 Java），不但能满足现在的业务需求，同时还能适应将来的业务发展。
- (2) 开放性和标准性：采用有关的国际标准、国家标准和行业规范。
- (3) 时效性和共享性：能够及时、准确、动态地更新和反映客户新闻简报的信息，并提供信息资源的共享和传递。
- (4) 可操作性和可伸缩性：系统软件安装简单、操作方便，具有良好的操作界面，尽可能考虑今后业务的发展。
- (5) 能在已有构件的基础上生成应用程序，实现“零代码编写”的理想目标。

二、设立架构集成的原则

电子政务架构的设计要力求做到完备性、灵活性、可扩展性、可理解性，同时抽象能用于不同的场合；用户能轻松地添加和修改功能，定制框架；用户和框架的交互清晰，文档齐全。

为此，我们对电子政务架构集成设立了如下原则：

- (1) 实现面向产品化、实用性的构件库系统，并具开放性、可扩展性。
- (2) 支持异构环境中的系统、构件的互连和通信。
- (3) 实现新旧系统的兼容性。
- (4) 提供一致的接口分配。
- (5) 遵循重要构件标准（如 CORBA）。
- (6) 构件具有透明本地化、平台无关性特点。
- (7) 系统的配置、数据交换基于 XML 和 Java 的标准化格式。
- (8) 支持个性化信息服务的定制和可重构性。

三、架构集成的分析

架构（Architecture）即软件系统的体系结构，它是软件设计中非常重要的一个环节。本质上，架构是一种抽象，也就是说，架构设计摒弃了具体的细节，仅仅抓住软件最高层的概念，也就是最上层、优先级最高、风险最大的那部分需求。在软件开发的过程中只要需求和架构确定之后，这个软件就基本上可以定型了，而架构设计的实质是一种权衡。一个应用系统特别是电子政务系统的架构有许多种解决方案，然而我们在其中要确定唯一的架构方案，这就意味着我们要在不同的矛盾之间进行权衡来做出选择。在电子政务应用架构的集成设计中，我们面对很多的矛盾：开放和集成，一致性和特殊化，稳定性和扩展性等等。任何一对矛盾的解决都依赖于我们对软件的不同期望：一方面我们希望软件稳定运行的要求，另一方面却影响了我们对软件易于扩展的期望；一方面我们希望软件简单明了，另一方面却增加了

设计的复杂度。所以，我们对于架构设计优劣的标准只能是根据需求的侧重点，进行权衡而做出合理的抉择。

架构设计要从系统的全局出发，着重考虑系统间的关系和设计系统间的接口。电子政务作为一个体系，在其开发过程中，我们所面临的任务是要把政府不同时期或不同领域的应用系统进行集成。然而，从概念上来看，“集成”与“连接”（point-to-point）不同，它不是简单地把两个或多个单元点到点地连在一起。事实上，“集成”是将原来没有联系或联系不紧密的单元组成为有一定功能的、紧密联系的新系统，而两种或多种功能的集成包含着两种或多种功能之间的相互作用。另外，集成是属于系统工程中的系统综合、系统优化范畴。电子政务中的集成是把人、政务办公系统和信息管理系统三者紧密结合起来，组成一个统一的整体，使整个政府机关范围内的工作流、物流和信息流都保持通顺流畅和相互有机联系。这里的集成要从整个政府机关的服务目标和内外环境出发进行优化组合，应摆脱过去人为分工、部门界限和职能范围所带来的束缚，这是实现工作流、物流和信息流畅以及政府高效运行的前提。其中，工作流的集成要消除多余的环节，须从工作流的组织上进行优化、协调，包括进行并行作业；信息的集成要消除信息流各环节上人工重复输入信息以及输出数据的泛滥，使人们及时得到准确的信息，保持整个系统内数据的一致性和完整性。

所以，在电子政务应用架构的设计上，我们主要采用两种关键技术：（1）基于 J2EE 的标准规范开发：基于 MVC 框架设计模式，Java Bean/EJB 的业务逻辑封装，Jsp/Servlet 的表现逻辑。（2）基于 XML 的数据标准：XML 格式数据的导入导出；XML 格式的系统配置；XML 格式的数据交换：XML 的数据呈现。

考虑到电子政务系统结构非常的复杂，因此在系统结构的处理上，我们采用了层模式作为系统的基本结构。在每个层，我们还定义了几个子模块来处理特定的问题，将复杂的功能有序的组织起来。一般地，电子政务系统由公文管理、项目审批、行政事务、决策信息、项目稽查、安全管理等子模块构成，这就要求电子政务的开发应该是基于构件式的开发模式来实现系统功能。按照一定的标准编制成各类具有特定功能的构件集合，而构件之间要进行无缝集成。所以，针对电子政务应用系统的逻辑综合架构，我们在集成过程中采用基于中间件集成的方法，中间件起到提供一个应用程序间协调点的作用：中间件提供通用接口，所有集成应用可以用接口相互传递消息，每个接口定义了一个由另一个应用程序提供的业务过程；中间件能够执行复杂的操作—交换、聚集、路由、分离和转换消息—当数据在应用到应用传递时。不仅如此，我们在构件的基础上采用 Java 技术。其原因是：首先，Java 促进了软件构件技术的开放与标准化。其次，Java 全力支持 Internet。另外，Java 提供了更高级的安全保障机制，它能让用户在每一层上设置安全保障，对过程进行更强有力的控制。Java 的软件构件称为 JavaBean，或者简称 Bean。按照 Javasoft 给出的定义，Bean 是能够在构造工具中进行可视化操作的可重用软件。JavaBean 的组件模型包含组件和容器两个基本要素，这一思想在 ActiveX/DCOM 技术中同样存在。作为一种典型的组件模型，JavaBean 具有属性、方法、事件、自我检查、定制和永久性等 6 个方面的特征。其中前 3 种特征（属性、方法、事件）是面向对象的组件必须满足的基本要求，属性和方法保证 Bean 成为一个对象，而事件可以描述组件之间的相互作用以及组件与容器之间相互感兴趣的事情。通过事件的生成、传播和处理，构件相互之间关联在一起，共同完成复杂的任务。后三种特征（自我检查、定制和永久性）主要侧重于对 JavaBeans 组件性质的刻画。最后，在客户端采用前端浏览器界面，业务逻辑采用 servlet，配合 JSP 编写，浏览器到服务器的数据采用集中处理，具体的方法是在业务逻辑和前端浏览器之间采用 FC（Front Control）模式，接受前端浏览器传送过来的数据，并指派给相应的业务逻辑处理。数据的合法性检验分为两个部分：和业务逻辑无关的基本合法性验证在前端使用 Java Script 处理，和业务逻辑相关的合法性验证在业务逻辑层处理，可以使用一个集中的 servlet 专门处理错误情况。

总而言之，为了实现一个好的架构设计，在电子政务应用架构的设计中力求做到：

(1) 代码复用：尽可能地对已有的代码甚至大粒度构件进行重用，不但可以提高软件的开发效率，而且可以降低成本。这样，把对于数据库的操作用 DAO 作为一个类；把具体功能的实现用 JavaServlet 类；对于一些公用的字符、日期等的转换作为公用类来进行调用。

(2) 透明设计：把软件功能的具体实现细节隐藏起来，仅把客户需求的接口呈现给客户。一个具体的例子是我们使用 JSP 的 tag 技术（如 JavaBean）来代替 JSP 的嵌入代码。

(3) 易于扩展：在电子政务的各个应用系统界面中，我们都留有接口（如组织管理系统界面等），以便于将来的软件扩展。

(4) 简洁明了：复杂的架构不论是测试还是维护都是困难的。因而，我们在电子政务的架构设计中，在能够满足目的下尽可能做到架构的简单明了。这就要求我们采用适当的设计模式。

(5) 高效安全：作为电子政务应用架构，考虑到信息访问量因素，为了保证系统的高效安全，我们在对于数据库的操作上采用了连接池技术。

四、架构集成方案的设计

针对公司的总体规划，结合国外的先进技术和经验以及国内的具体现实情况，我们所采用的集成方案为：建立以 GIP 为统一入口、以 UniFlow 工作流管理系统为统一平台，将 E-Form 电子表单系统、Ucan 报表系统及 DocMan 公文管理系统（在应用集成模块中）进行集成的方案如图 2.2 所示：

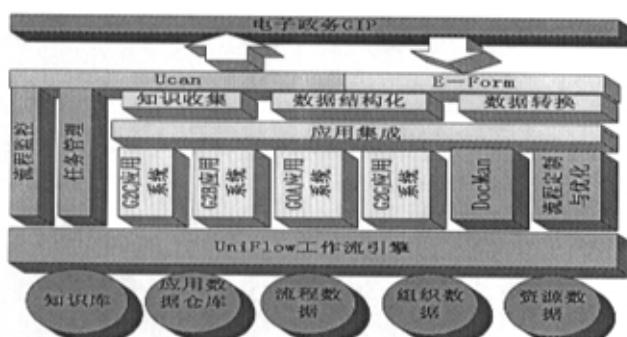


图 2.2 电子政务应用架构

其中，UniFlow 工作流管理系统的集成将在第三章论述，GIP、E-Form 电子表单系统、Ucan 报表系统和 DocMan 公文管理系统的集成将在第五章讨论。

针对该集成方案，我们采用下面的设计方法。

(1) 通过对系统界面的集成，每个用户注册得到统一用户码，作为其登录的通用账号，可以登录查询整个事务流程，审批的进展状态和提醒通知公告服务。政府对内可以提供适应各部门电子办公的接入系统，对外统一窗口提供各种职能服务。民众可以在网站了解或办理部分或所有服务事项。

(2) 通过对 UniFlow 工作流管理系统的集成设计，各个部门、各个人之间完全通过计算机进行自动化的协同工作；使用 Internet 的邮件服务及时发邮件通知用户处理，在政府部门间及时的传达信息；系统提供的系统管理功能可以根据需要对人员、权力等功能进行配置，包括工作流的设置、权力的授予和撤销。这样大大提高了工作的顺利展开和改进。

(3) 通过对电子信息资源的集成，使得各种协议、信息格式和接口标准符合国际标准，

来实现跨平台运行的体系架构，可以运行在各种操作系统平台上，如 UNIX、Windows NT 等平台。

(4) 采用高度模块化设计和各子系统之间留有接口，对各个应用系统的集成，为系统扩充集成留有余地。通过系统构件的分层，将频繁变动得业务逻辑层分离出来，实现通用类构件的完全复用。并且在各个模块之间设计统一的接口，当某一模块业务逻辑改变时，使系统之间的影响最小，使系统实现即插即用，让系统容易升级。为此将我们将电子政务系统构件定义为四个层次：系统构件层，指系统平台本身所提供的类库。通用模块构件层，它提供系统中所需使用的通用函数类，如字符串处理、数据库连接等类函数和业务处理过程相对不变的类。业务构件层，它提供领域内所需使用的类库和业务处理服务，在该层的函数调用或继承通用构件层的函数。表现层（JSP 页面层），它实现客户端界面，并调用业务层所提供的服务。

综上所述，我们开发的电子政务系统采用软构件设计方法，把应用系统划分为若干个功能模块，把程序生成问题转化为软构件的设计、处理问题。对于不同的应用系统，通过分析发现它们之间的共性与个性，把共性部分作为标准软构件，把用户要求存入规则库。根据软构件集成算法，将软构件连接成完整集成体系。

五、架构方案的实现

根据设计的架构集成方案，我们使用先进的 UML（统一建模语言）标准和风格进行系统分析和设计，包括界面设计、流程设计、建模规范、编码规范、持久层设计和测试数据。整个方案的实现过程如表 1.1 所示。

表 1.1 电子政务应用架构的实现过程

集成过程	集成工作流程	集成任务	集成工作的里程碑
需求	Define the E-Government Affairs System	1. Capture a Common Vocabulary 2. Find Actors and Use Cases 3. Prioritize Use Cases 4. Review the Requirements	1. Change Request 2. Glossary 3. Iteration Plan [provided by PM] 4. Risk List [provided by PM] 5. Review Record 6. Supplementary Specification
	Refine the E-Government Affairs System Definition	1. Detail a Use Case 2. Structure the Use-Case Model 3. Review the Requirements	7. Use-Case Model (Actor, Use-Case, Use-Case Package) 8. Use-Case Modeling Guidelines [provided] 9. Use-Case Priority List 10. Vision [provided by PM]
分析	Define an Initial the E-Government Affairs Architecture	1. Architectural Analysis 2. Review the Architecture	1. Change Request 2. Data Model 3. Deployment Model 4. Design Guidelines [provided]
	Analyze Behavior	1. Model the User Experience 2. Use-Case Analysis 3. Review the Analysis	5. Design Model (Analysis Class, Design Class, Framework)

设计	Refine the E-Government Affairs Architecture	1.Identify Design Mechanisms 2.Identify Design Elements 3. Incorporate Existing Design Elements 4.Describe Concurrency and Distribution 5. Review the Architecture	Component, Design Package, Design Subsystem, Interface, Use-Case Realization) 6.Reference Architecture [provided] 7. Review Record 8.Software Architecture
	Detail the Design	1. Use-Case Design 2. Subsystem Design 3.Component Design 4.Class Design 5.Database Design 6. Review the Design	Document 9.User-Experience Guidelines [provided] 10. User-Experience Model (Navigation Map, Screen, Use-Case Storyboard)
实现	Structure the Implementation Model	1.Structure the Implementation Model	1. Change Request 2.Implementation Guidelines [provided] 3.Implementation Model (Implementation File, Implementation Directory) 4.Programming Guidelines
	Implement Design Elements	1.Implement Design Elements 2.Perform Unit Tests 3. Review the Implementation	5. Review Record 6.Test Guidelines [provided] 7. Test Results 8. Test Script

六、架构的性能优化

针对该电子政务架构的设计方案，为了保证架构的使用性能，我们采取以下两方面的措施。

1. 高性能的数据库连接

考虑到数据库的访问会通过网络实现，连接数据库的性能对整个系统的影响非常大。在数据库应用程序中，频繁对数据库进行连接，将消耗掉大量资源，系统性能会急剧下降，甚至有可能成为整个系统性能的瓶颈。为此，我们采用数据库连接池技术，把数据库的连接进行专门处理，并采用后台线程建立联接的机制，可以使数据库的访问达到最佳状态。其原理是：预先准备一个连接池，并且由所有处理请求的 servlet 共享；通过这种方法，servlet 可能就不用等待数据库建立连接，如果在数据库访问请求时有可用的连接资源，就可以直接使用池中的连接而不必等待 java 虚拟机再建立一次连接。另外，封装多种数据库（SQL Server, Oracle 等）的连接。

同时，还需要设计两个相关的类。

(1) 管理类 DBConnectionManager 支持对一个或多个由属性文件定义的数据库连接池的访问，客户程序可以调用其中的 getInstance()方法访问本类的唯一实例。其中的内部类 DBConnectionPool 定义了一个连接池，它能够根据要求创建新连接，直到预定的最大连接数为止；在返回连接给客户程序之前，它能够验证连接的有效性。

(2) 数据库连接类 DbConn 实现从连接池里获得对数据库的连接，并且进行查询、更

新、事务处理、释放连接等操作并采取相应的措施：首先，在每次不需要该连接时必须调用 close()方法，并把该连接还给连接池；其次，当查询次数有可能超过 300 次时，调用方法 executeQuery1(String sql)；最后，对于 sqlserver2000 中进行事务处理时，不能调用方法 executeQuery1(String sql)。

2. 高效灵活的应用逻辑

我们对电子政务的具体应用进行了深度抽象，并且采用统一的数据库数据传输结构和统一的顶层接口，使应用逻辑有很强的灵活性。此外，在应用层中，利用对象缓冲和超时刷新管理，即可使访问数据库的次数大大减少，又不会耗费太大的系统内存资源。

§ 2.3 小结

本文首先从电子政务的实际应用出发，从整体上对电子政务应用系统进行了详细地分析并提出了电子政务应用架构的方案。首先，从理论上提出了电子政务集成的原则和目标。其次，将政府的各个应用系统集成为以 GIP 作为统一入口，建立在以 UniFlow 工作流管理系统为基础平台之上架构。最后，从技术理论上对电子政务应用架构的集成方案及其实现的过程进行了分析和研究，并对整个系统的性能优化提出了解决措施。

第三章 UniFlow 工作流管理系统的集成设计与实现

§ 3.1 引言

据统计，在政府的日常活动中，70%以上属于流程类活动，即：大量的政府业务是按照某种固定的程序进行处理。我们把一项事务经过一个环节处理后再转往下一个环节的连续步骤称作流程。而工作流（过程模型、过程定义或工作流模型）就是针对这些业务过程进行分析、评价或优化，最后借助于计算机自动处理。工作流管理软件的目的就是规范流程，最大限度的提高工作效率。

在电子政务应用中，公文流转的载体是工作流，政府机关的各种表单也是随着工作流而进行运转和分发的。由此可见，工作流管理系统（以下简称 WFMS）是电子政务架构中的基础，政府部门的其它应用系统都依赖于工作流。可见，WFMS 的本质是电子政务应用的底层架构，即它作为电子政务产品的其它应用模块的基础运行和开发环境。

显然，要解决电子政务体系这种频繁变化、复杂协同的办公问题，关键就是搭建能够满足电子政务体系所需要的 WFMS 平台。我们在进行电子政务应用架构的设计时，把 UniFlow 工作流管理系统作为基础平台。

§ 3.2 工作流技术及 WFMS 概况

一、工作流和 WFMS 的关系

工作流管理联盟（WFMC）权威机构认为，工作流是一类能够完全或者自动执行的经营过程，它根据一系列过程规则、文档、信息或任务能够在不同的执行者之间进行传递和执行。即：一个工作流包括一组相互关联的任务（Task）和它们相互顺序的关系，还包括流程及任务的启动和终止条件，以及对每个任务的描述。而 WFMS 则是以计算机支持的分布式、协同工作的自动化来实现工作业务流程的软件环境。工作流是对一整套规则与过程的描述，以便管理在协同工作进程中的信息流通与业务活动。其目标不仅仅是处理过程，也不仅仅是把事物从一个地方流向另一个地方，而是管理那些引导作业环境如何运作的规则与过程。WFMS 履行了这个关键的角色并为规则与过程自动化提供了有效的实现平台，还给予业务逻辑一个易操作易控制的界面。

二、WFMS 的功能划分

在 WFMS 中，我们首先应该建立一个人事组织子系统，其中的人员按照其所处的角色（职务或者工作权限）来架构的。将日常工作进行逻辑上的划分或分割若干子工作（或步骤），并定义其处理的顺序。进而考虑到每个步骤根据不同的条件会产生相应的结果。然后对在此期间所产生不同的状态进行定义。并根据各个步骤的不同需求，分别建立各阶段所需要的表单、文件等。一般地，WFMS 应该具有如下三个基本的相关功能模块：工作流程的建立、工作流程的运行控制和工作流程的信息交互。其中各个模块又包含其数据系统、接口。各模块的功能如表 3.1 所示。

表 3.1 WFMS 的功能划分

工作流管理系统	序号	功能模块名称	功能概述
	1	工作流程的建立	通过分析、建模及系统定义手段将现实世界的业务过程转化成某种能够被计算机处理的形式化表示：图形或符号。其实质就是确定对于工作流程进行控制所用的各个参数，即工作流程的过程定义。而过程定义是由过程定义工具和过程定义数据所组成的。
	2	工作流程的运行控制	对于工作流程的状态进行转换控制；通过对流程的定义进行解释，建立流程的实例，并在相关状态下调用流程所需的资源。一般是通过工作流引擎(Workflow Engine)提供这些服务。Workflow Engine 是 WFMS 的核心软件组元。它的功能包括：解释过程定义、创建过程实例并控制其执行、调度各项活动、为用户工作表添加工作项、通过应用程序接口（API）调用应用程序、提供监督和管理功能等。工作流执行服务可以包括多个引擎，不同工作流引擎通过协作工作执行工作流。
	3	工作流程的信息交互	是指在工作流程实例运行中，系统同用户或者其它应用模块进行信息交互的过程。如：用户的读写操作、系统间的信息交互处理等。

§ 3.3 UniFlow 的集成

§ 3.3.1 UniFlow 的概况

电子政务的其它应用系统以 UniFlow 作为基础平台，它们都是在 UniFlow 工作流管理平台的基础之上开发和运行的，如图 3.1 所示。

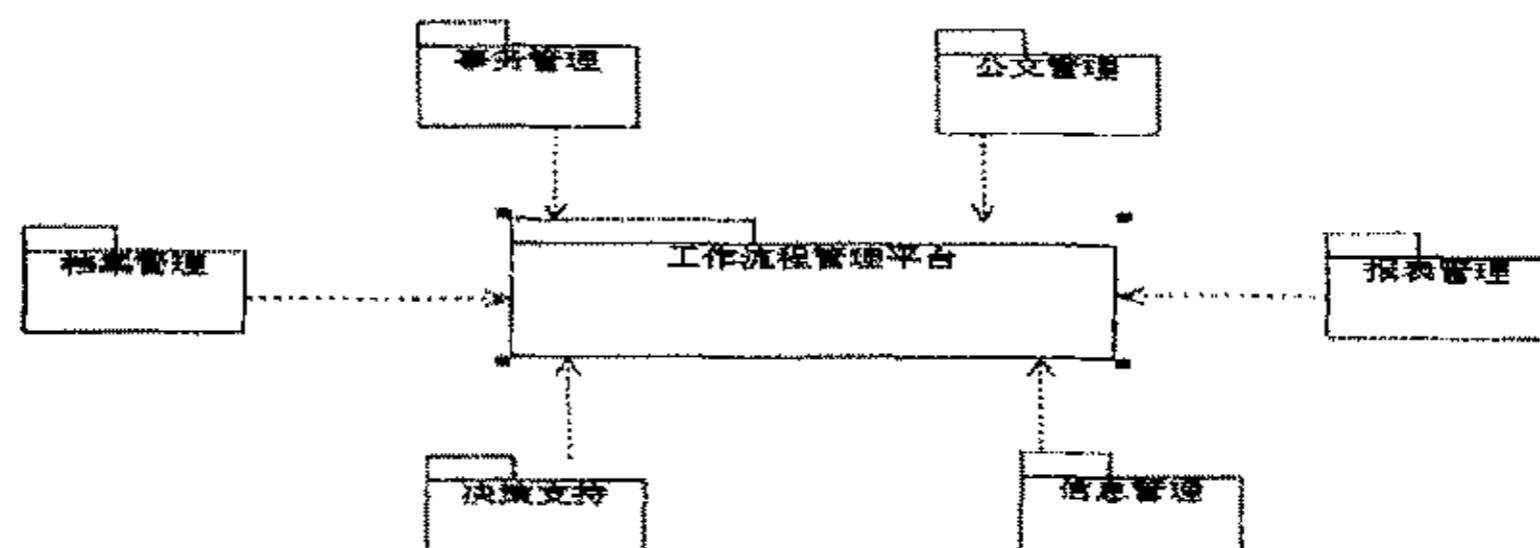


图 3.1 UniFlow 工作流平台处于核心位置

因此，UniFlow 的集成必然涉及到下面的技术领域：首先，我们要面对工作流的建模技术，即对基于电子政务应用架构的工作流模型的集成进行分析和设计；其次，要解决好基于分布式的资源分配、任务调度分配；最后，利用中间件和 J2EE 技术，把工作流模型和其它应用模块进行有效地集成并，实现一个高效、安全的 WFMS。

为此，我们参照了 WfMC 标准，把流程定义工具作为重点，而流程的导航作为核心。根据上面的 WFMS 的三个基本的功能模块，我们将 UniFlow 工作流平台系统功能模块在此基础上拓展为 6 大模块，它们分别是：工作流元模型包、工作流定义工具、工作流管理工具、工作流引擎、工作流监控

工具和工作流任务管理器。各包的关系如图 3.2 所示。

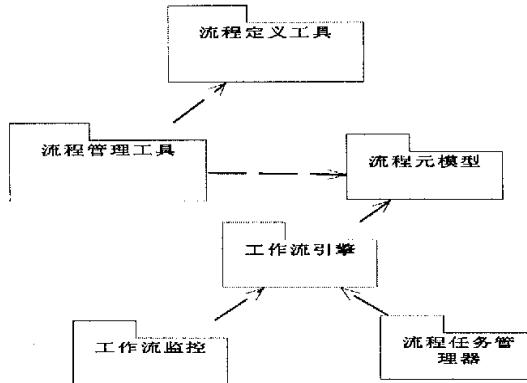


图 3.2 UniFlow 工作流平台系统模块划分

其中，各模块的功能如下：

1. 工作流元模型 (flowModel): 实现对工作流元模型各个元素的统一管理，能提供给工作流定义工具方便的接口，使工作流定义工具能够生成业务流程。
2. 工作流定义工具(flowTools): 调用工作流元模型提供的功能，为用户生成业务流程提供方便。
3. 工作流管理工具(flowMan): 实现业务流程的分类、状态和权限管理，并可以便捷地调用工作流定义工具对业务流程进行新增、删除和修改。
4. 工作流引擎(flowEngine): (1) 为过程和活动的执行进行导航。根据过程定义和工作流相关数据，为过程实例的运行进行导航。(2) 与外部资源进行交互完成各项活动。
工作流执行服务通过两种途径完成与外部资源和用户的交互：客户应用接口和直接调用应用接口方式。
5. 工作流任务管理器(flowTask): 分配流程实例处理的任务、实现任务的办理；提供代办事宜列表，方便用户任务的查询。
6. 工作流监控(flowMonitor):
(1) 实现对所有业务流程的跟踪，监控业务流程的运行，对业务流程的历史信息进行管理。(2) 根据流程的运行期限，实现流程运行的稽催管制并通报任务项的完成。(3) 对任务进行重新分配（设置代理人）。(4) 并案、拆案管理。

§ 3.3.2 UniFlow 的集成设计

在 UniFlow 中，工作流模型是该系统的基础，它是政府机关依靠 WFMS 以电子公文的运行方式来处理特定的工作业务的模式。工作流模型所描述的是一个简单或复杂的业务过程：这个业务是由什么样的角色在什么时候如何去处理，在处理的过程中需要什么数据资源并产生什么样的数据结果。通过对 WFMS 的研究分析，我们知道 UniFlow 工作流模型建立在过程模型的基础上，需要将描述企业或机关人员的组织模型、描述企业或机关资源的资源模型以及提供信息定义的工作流相关数据加以集成，使其成为一个完整的、具有多视图特性的模型。因此，我们将 UniFlow 工作流模型分为以下四部分：工作流之过程定义、工作流之组织架构、工作流之资源管理和工作流之相关数据系统。其含义如表 3.2 所示。

表 3.2 UniFlow 工作流模型的组成

工 作 流 模 型	序号	名称	含义
	1	工作流之过程定义	用来定义工作流模型的过程逻辑，包括组成工作流的所有状态及状态之间的依赖关系。它是整个工作流模型的基础与核心。
	2	工作流之组织架构	用来定义政府机关及其人员、角色的组织管理架构，它包括几种不同形式的组织元素，以及每种组织元素内部的递阶层次关系。组织模型的主要任务是为企业或机关人员执行工作流提供柔性的组织定义，为过程模型提供参与角色。
	3	工作流之资源管理	用来定义政府机关资源的组织机构，包括含有几种不同形式的资源容器元素以及容器内部的递阶层次关系，直至最终的原子级资源个体。资源模型的主要任务是为政府机关人员执行工作流提供参与的文件、表单资源。
	4	工作流之相关数据系统	用来定义工作流执行过程中需要用到的数据，包括简单数据类型与复杂的企业对像，即：WFMC 所提出的工作流参考模型中的工作流相关数据。工作流相关数据主要用于各种条件的判断，以实现工作流引擎对不同状态下活动的选择性路由。工作流相关数据为工作流的执行提供了数据信息。在 UniFlow 中，工作流相关数据采用对工作流程进行属性定义的方式实现，并作为工作流过程模型的一部分。

§ 3.3.2.1 工作流过程模型的集成设计

一、工作流过程模型分析与研究

工作流过程模型是整个 WFMS 的基础，它的确定性保证了系统内各组成部分之间交互的一致性，也决定了一个 WFMS 集成实施的诸多环节。在信息管理自动化的环境下架构电子政务，工作流模型采用简单、直观、又具有较强描述能力的面向政府应用的模型。因此，为了克服现有的建模方法缺陷，我们在 UniFlow 平台的分析过程中，从状态图及有限状态机理论中引申设计出了适合于电子政务架构基础的基于事件驱动的状态图（State Diagram）模型。

1. 状态机(State Machine)

所谓的状态机是由一系列状态、状态转移和事件组成，它依靠状态的转移过程来明确地表达出政府业务处理的行为过程，即状态机表达了一个对象类（工作表单类或公文类）的一个有序的状态及其之间的转移过程，而状态机模型就要采用一个对象类对应一个工作流程的机制。因而，我们把状态机分为状态、转移、事件、活动和动作 5 个元素，它们的含义如表 3.3 所示。

表 3.3 状态机的组成及其含义

状 态 机	序号	元素名称	含义描述
	1	状态 (State)	表示一个模型元素在生存期的一种状况，如：满足某些条件，进行某些活动，或等待某些事件的出现等。一个状态就是在一个有限的时间段内存在。状态图中的状态可以是简单状态或组合状态(Composite State)。包含有子状态的状态称为组合状态。
	2	转移	表示一个模型元素的不同状态之间的联系。在事件的触发下，一

	(Transition)	一个状态可以转移到另一状态。转移用实箭头线表示，箭尾连接出发状态，即源状态，箭头连接目标状态。在箭头线上可以表示与该转移有关的选项：事件、条件或动作。当处于源状态的对象接收到一个事件，并且条件得到满足时则执行相应的动作，并从源状态转移到目标状态。如果事件缺省，一旦条件满足则自动实现转移。
3	事件 (Event)	是一个有意义的出现(Occurrence)的说明。该出现在某个时间和空间点触发，并且立即触发一个状态的转移。例如，一个信号、对一个操作的调用、一个对象的创建或销毁、超时、某个条件的改变等都是事件的例子。
4	活动 (Activity)	是在状态机中进行的一个非原子的执行，它由以序列的动作组成。
5	动作 (Action)	是一个可执行的原子计算，它导致状态的变更或返回一个值。

2. 状态图是状态机的图形化的直观表示。它表现了一个业务流程的生存历史，显示触发状态转移的事件和因状态改变而导致的动作。一个工作流的状态图分为状态(起始状态、中间状态和结束状态)、转移、判定和同步4种图形化表示。

(1) 状态

状态分为起始状态、中间状态和结束状态，这三种状态的含义及其图形化表示如表3.4所示。

表3.4 状态的分类、含义及图形

状态名称	状态的图形化表示	状态的含义	注解
起始状态		起始状态是流程的起点(状态的初始源)，而不是转移的目标。在流程启动时，也需要完成某种特定的任务来激活整个工作流的运转。	一个流程被激活的方式有：(1)人工激活，即人为的信息数据输入或产生；(2)定时或限时激活；(3)外界消息激活。
中间状态 (一般状态)		中间状态由状态名称、状态内部转移和嵌套状态图组成。一个工作流的状态分为起始状态、中间状态和结束状态。在内部转移中规定了动作或活动，它们将响应对象所接收到的事件，但是不改变对象自身的状态。	在内部转移中规定了动作或活动，他们将响应对象所接收到的事件，但是不改变对象自身的状态。状态图中的状态可以是简单状态或组合状态。包含有子状态的状态称为组合状态。
结束状态		终结状态是模型元素的最后状态，代表一个状态图的终止点，是转移的最终目标，常用圆中套一个实心圆表示。	

(2) 转移(Transition)

转移用实心箭头线表示，箭尾连接出发状态，即源状态，箭头连接目标状态。在箭头线上可以表示与该转移有关的选项：事件、条件(Condition)或动作。当处于源状态的对象接收到一个事件，并

且在满足一定的条件时就执行相应的动作：工作流就从源状态转移到目标状态。如果事件缺省，一旦条件满足则自动实现转移。如图 3.3 所示。

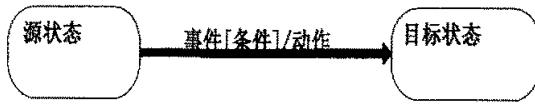


图 3.3 状态的转移

(3) 判定(Decision)

判定处于状态图中的一个特定位置，作为工作流走向的判断：随着此条件下的取值不同而发生不同的走向（即分支），判定用空心小菱形表示。一般情况下的判定，只有一个导入转移和两个导出转移，而导出转移的条件为布尔表达式，如图 3.4 所示。

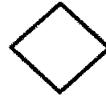


图 3.4 判定

(4) 同步(Synchronization)

同步可视化地定义了并发工作流的分支（Fork）与聚合（Join）。如表 3.5 所示。

表 3.5 同步

名称	图形化表示	含义
分支 (Fork)	<pre> graph LR A([源状态]) --> B[] B --> C1([目标状态1]) B --> C2([目标状态2]) </pre>	分支是指所处理的业务需要多部门（多任务）分开处理。它是一个源状态分为两个或两个以上的目标状态。而源状态进入目标状态（1、2）又可分为三种情形：And（同时并行）、Or（仅选其一）和 Discriminator（条件判断）
聚合 (Join)	<pre> graph LR A1([源状态1]) --> B1[] A2([源状态2]) --> B1 B1 --> C([目标状态]) </pre>	聚合是两个或两个以上的源状态连接为一个目标状态。其含义是：目标状态任务的执行有待于源状态（1、2）均完成时，才可进行。

二、工作流过程模型的集成

根据以上分析，我们将一个流程抽象为一个系统，并将该流程系统上升为一个流程系统类。对于一个流程系统类，分析其对流程运转有意义的属性，这些属性即成为工作流程相关数据。根据这些关键属性，确定流程系统类的各个状态，并分析出在流程各个状态下所能接收的事件以及流程在某一状态条件下接受事件触发后的转移方向，从而将实际的业务流程抽象为 UniFlow 工作流程。在流程实例运转过

程中，流程系统总是处于某一个状态。在特定状态条件下，流程系统通过事件驱动实现流程的转移，从而达到流程流转的目的。

1. 模型元素的集成分析

UniFlow 的理论依据是状态图与有限状态机模型。为了便于集成，有必要先对模型元素进行分析。由于 UniFlow 的工作流模型的表现形式是一个状态图，并展现为有向图模式。其模型元素分析如表 3.6 所示。

表 3.6 UniFlow 工作流模型元素

工作流模型元素	编 号	元素名称	图标	元素分析
	1	开始状态		开始状态是状态图的起始点，代表流程系统的初始状况，也是业务流程的唯一入口点，它没有前驱节点。
	2	结束状态		结束状态是状态图的结束点，代表流程系统的结束状态，也是业务流程完成的标记，它没有后继节点。
	3	一般状态		一般状态是状态图的某一特定状态，代表业务流程系统某个特定条件下的表现形式。它是业务流程系统的一个原子状态，是不可以再分的。它至少有一个前驱节点和后继节点，在事件的触发下根据转移定义实现状态之间的迁移。
	4	组合状态		组合状态也是状态图的某一个特定状态，但是组合状态必须包含一个或多个状态图。即组合状态是可以再分的，非原子状态。作为组成业务流程状态图的基本单元，一般状态是不可以再分的原子状态，一旦我们的流程比较复杂，涉及的环节比较多，那么状态的数量也将大大增加，影响用户对主要业务流程状态的总体把握。常用的解决办法就是采用状态嵌套子流程的方式实现工作流程定义的自顶向下。另外，在某些业务流程的特定状态下，往往可以将该状态下所要办理的工作任务同时分给不同的业务人员进行并行办理，而这些并行的流程之间往往又需要协调（相互等待）运行。为此，我们定义了该状态类型，以便方便用户对业务流程的定义，并使得 UniFlow 流程模型具备并发、同步、协调的功能。
	5	会办状态		会办状态也是状态图的某一个特定状态，是我们为 UniFlow 工作流模型所设计的特有状态形式。该状态形式包含且必须包含一个子状态图，该子状态图可以根据流程条件，任意地对子状态图循环（顺会）或并发（并会）的运行。我们设计该状态的目的是为了解决电子政务事务处理过程中相互协作办理的问题。会办状态需要一个或多个会办人员进行处理。
	6	事件		事件是一个业务流程系统在某一个状态下可以接收的有意义的出现（Occurrence）。流程系统根据触发的事件类型执行响应的动作，即：当流程运行过程中需要激活其他流程时所采用的触发机制。根据事件的触发机制，我们把事件类型分为下面四种：(1)自动触发事件 (2)人工触发事件 (3)消息触发事件 (4)时间触发事件

	7	转移		转移是一个状态图在特定状态下接收事件后进行状态变迁的依据，表示业务流程系统在特定状态下接收事件后的转移形式。
	8	判定		判定是一个状态图在特定状态下接收事件后进行状态变迁的依据，属于条件转移，它包含真值转移与假值转移两种类型。可以使用嵌套的形式构造任意复杂的条件转移形式。
	9	子流程		一个子流程对应一个状态图，它可以由任意的 UniFlow 模型元素组合而成并被组合状态或会办状态组成。
	10	流程属性	无	流程属性是 UniFlow 工作流程定义的流程相关数据。

2. 模型元素的集成设计

根据对模型元素的分析，我们将模型元素的集成为以下 7 种基类：业务流程类、状态类、事件类、转移类、判定类、属性类和注释类。其中，一般状态类、组合会办状态类、开始状态类、结束状态类均是状态类的导出类；组合会办状态又派生出组合状态与会办状态；业务流程类是由其他元素类组合而成的，而组合会办状态类又由一个或多个业务流程类聚合而成。一个业务流程类必须包含一个开始状态和一个或者多个结束状态，也可以包含任意多个其他元素及属性。开始状态类可以拥有一个或多个事件出口且至少有一个事件出口，但没有转移入口。结束状态类可以拥有一个或多个转移入口且至少有一个转移入口，但没有事件出口。其他状态类型，可以拥有一个或多个转移入口和事件出口，且至少必须分别有一个转移入口和事件出口。一个事件类可以绑定一个转移类或判定类。一个判定类包含两个出口（条件值真或假，判定条件的取值依赖于业务流程属性值）每个出口可以绑定另一个判定类或转移类。每一个元素都可以拥有一个或多个注释。模型元素集成后的类图 3.5 所示。

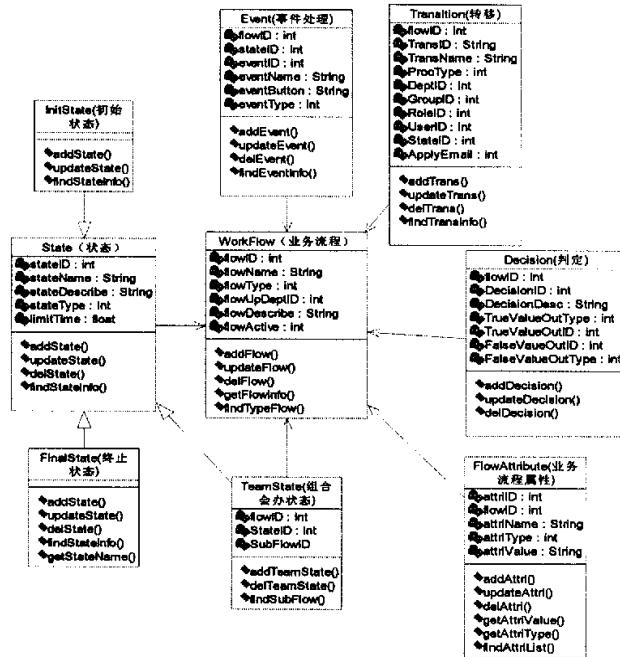


图 3.5 UniFlow 元模型类的集成

3. 业务流程的集成

在 UniFlow 平台的分析研究过程中, 我们从状态图及有限状态机理论中引申设计出了适应于电子政务等所需的、基于事件驱动的日常办公流程的状态图模型, 该模型可以实现业务流程需要的如下功能, 见表 3.7 所示。

表 3.7 业务流程的功能

序号	功能名称	功能含义
1	串行事务	对于串行事务的实现, 只需将业务流程根据需要分为各个状态, 并将这些状态通过转移连接起来即可。
2	并发事务	对于事务并发的实现, UniFlow 模型通过在组合状态下嵌套任意多个并发的子流程得以实现, 当流程转移到组合状态时工作流引擎自动激活组合状态下的各个子流程, 使组合状态下的子流程并发地运行。对于同一个流程的并发执行, 我们采用会办状态的并会得以实现。
3	并发协调	指并发流程间的相互等待。UniFlow 模型是通过在组合状态下定义多个子流程来实现事务并发的。对于事务的并发协调, UniFlow 模型是通过对组合状态下的子流程的相关状态设定同步关联, 工作流引擎作对应控制而实现的。
4	并发同步	对于并发同步的实现, UniFlow 模型是通过对组合状态下子流程的结束等待来实现的。即: 在流程运行过程中激活了其他流程, 而本流程需要被激活流程的返回结果, 或根据被激活流程设置的流程相关数据, 在进行下一步状态选择时, 所采用同步等待机制。
5	事务无条件转移	对于事务无条件转移, UniFlow 模型可以简单地采用一般的转移连接弧, 连接两个状态而实现。
6	事务条件转移	对于事务条件转移, UniFlow 模型可以通过判定和无条件连接弧的组合使用得以实现。
7	事务循环执行	对于事务循环执行, UniFlow 模型可以通过在状态图中设定闭环转移实现事务循环执行功能; 或者也可以采用会办状态顺会得以实现。
8	事务会办(顺会或并会)	对于事务会办处理, UniFlow 模型专门设定了一个会办状态来实现, 并包含了顺会和并会功能。

三、工作流定义模型

工作流的生成是工作流建模的核心, 它根据用户在建模工具界面的输入产生可被工作流引擎解释执行的工作流。在工作流生成设计过程中我们参考了 WfMC 的规范, 结合实际需要进行了设计。工作流定义模型设计: 首先要定义一个元模型 (元模型指描述模型的模型), 它是用来描述工作流模型内在联系的模型和工作流模型内部包含的各个对象及其之间的关系、对象属性, 其目的是为了有利于建立可以在多个工作流产品之间交换信息的模型; 其次要定义一套可以在 WFMS 之间、管理系统与建模工具之间交互过程模型定义的接口, 该接口称为 WFMS 接口。

1. UniFlow 工作流元模型的集成

工作流包含诸如工作流名称、所属类别、过程启动、终止条件、系统安全、监控和控制信息等一系列基本属性。这个过程模型反映了业务过程的目的, 即这个过程要实现的目的和最终达到的目标是什么。如图 3.6 所示。

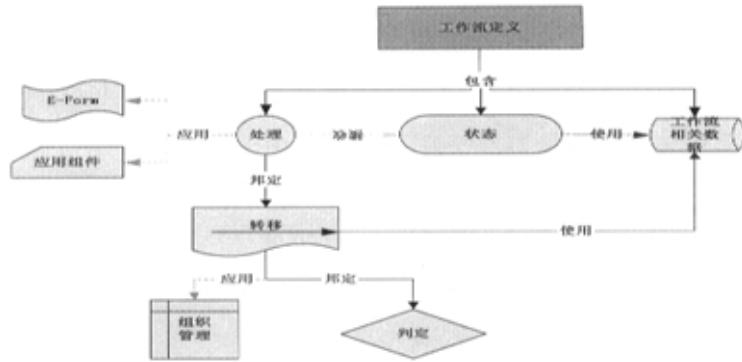


图 3.6 UniFlow 工作流元模型的集成

其中,工作流相关数据即流程属性, 它决定了流程的状态及状态的可接收事件和转移目标。

状态表示一个工作流在生存期的一种状况, 如: 满足某些条件, 或等待某些事件出现等。在特定转态下的可接收事件, 又由状态满足的条件决定。

处理(事件)表示了一个状态下的可接收事件, 它可以调用 E-Form 或应用组件进行作业, 完成一定的功能。一个处理绑定了一个转移。

转移是在事件触发下根据流程的属性实现流程状态迁移。转移可以联合判定实现复杂的状态迁移, 并应用组织数据实现任务的指派。

2. WFMS 接口设计

WFMS 接口描述了工作流定义输入与输出接口, 它在建模方面主要覆盖了以下几个功能:

- (1) 通信建立: 建立各参与运行系统之间的通信连接, 并在完成协作后断开通信连接。
- (2) 工作流模型操作: 对过程模型的名称进行检索并完成对过程模型对象的读/写等。
- (3) 工作流模型对象操作: 完成创建、检索和删除对象的操作, 并可以完成创建、设置和删除对象属性的操作。

§ 3.3.2.2 组织管理模块的集成设计

所谓的组织是为了优化管理和生产而建立的人员集合, 这个集合具有一定的内部层次和隶属结构, 具有一定的职能和责任, 并需要在指定时间内完成指定任务。每一个组织有一个最高管理决策者, 组织成员具有一定的职责、权力和技能, 成员之间根据组织的架构就产生了一定的权力和义务关系。

而在 WFMS 中, 组织架构是政府进行信息决策、任务执行和监督等重要活动的载体。它揭示了政府部门人员的管理层次, 反映了人与工作、决策、责权之间的联系。对于组织结构进行合理的架构和管理是提高政务活动运作效率和安全性的基础。一方面, 对于 WFMS 中所涉及到的敏感业务处理数据, 我们应对不同的部门或者角色来分配不同的管理和处理权限; 另一方面, 组织架构不但是工作流任务分配的依据, 同时也是电子政务架构中其它应用模块的基础数据。由此可见, 组织架构在电子政务系统中具有重要的地位。

一、组织架构模型集成分析

在进行 UniFlow 集成时, 针对组织管理模块, 我们选用矩阵式组织架构。首先, 它可以灵活的建立动态的面向任务的工作组; 其次, 它是一个富有弹性的网状组织结构, 有利于业务流程重组与信息共享; 另外, 它还能够使政府对于外界环境做出快速的适应, 从而有利于提高政府行政绩效。它把按职能划分的部门和按项目划分的小组(项目组、群组或工作组)结合起来组成矩阵, 同一名管理人员

在与原有职能保持组织和业务联系的同时可以参加多个工作组的工作。

利用矩阵制组织结构，我们将政务部门的静态部门职能结构用传统的树形结构形式表示，而将按项目划分的组织结构都采用动态工作组的形式表示，这样就使得电子政务的组织架构根据实际的应用而变化，从而更具有灵活性。其中的工作组是一个动态的概念，指为了完成某项任务动态组成的人员组合。因此工作组是一个跨部门的组织，并直接面向任务，直接对客户（任务提出者）负责。

为了在政府部门上下级之间，人员之间形成责权关系，在我们引入角色和职务这两个概念，并将每一个具体的人员赋予相应的角色和职务。职务以行政职责为前提，代表了管理上的等级关系，如书记，市长，总经理、部门经理等。在实际工作中任务的分配是动态的，并不总是固定在具体的人员上，因而角色和职务都是抽象的概念，直接面向业务过程：角色指具有某些技能，能够完成某项任务的人员的总称，如收件人等；职务指具有某个技能集合能够完成某项任务的人员的总称。以“角色”和“职务”为过渡，可以将组织职能模型和过程模型灵活地集成起来。

二、组织架构的集成

根据组织对象数据的一般结构，我们抽取了一个具有基本的标识属性、描述属性、行为属性和关系属性的组织对象一般类。并在此基础上派生出3个子类：层次结构类（OrgConstants）、行为者类和技能类，如图3.7所示。

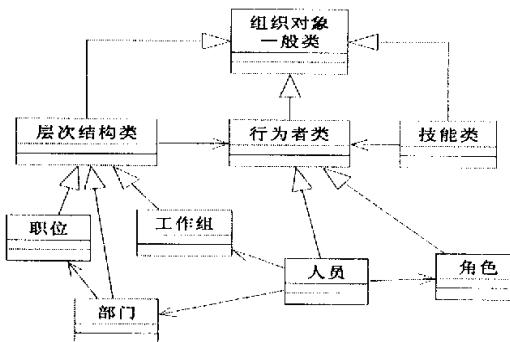


图3.7 组织结构的类

1. 层次结构类

层次结构类代表有层次性的企业或机关单位组织对象，它又派生为：部门、职位、工作组。如前所述，部门表示静态树形结构上的节点，工作组表示横向动态形成的人员组合。而职位是以行政责任为前提的概念，它代表人员在管理上的上下级关系。

2. 行为者类

行为者类代表具有行为能力的企业或机关组织中的对象，它又派生为：人员类和角色类。

3. 技能类

技能类代表政府单位人员的工作能力。它既包括政府单位中人员所具有的工作能力，也可以包括完成工作任务需要的工作能力。

4. 人员类

人员类代表政府单位中的人员，它管理人员对象的相关信息。

5. 角色类

角色类以技能（集合）为主要定义依据，同时包括对职务的要求。

将这些类进行打包集成为组织管理模块，如图3.8所示。在工作流的建立过程中，可以根据实际的业务模型选择相应的人员，这时可以调用包中相应的类来执行。



图 3.8 组织管理模块的集成

§ 3.3.3 UniFlow 集成的实现

通过以上的分析，我们可知 UniFlow 系统作为电子政务架构的基础平台，通过工作流系统（包括建模工具、管理工具、用户界面、工作流机以及相应的支撑数据库）实现了平台应用协作功能；并以平台的代理服务作为底层的信息基础结构，实现了平台的应用协作功能。整个工作流实施体系是分布于全企业范围的信息环境下的，各组成模块均是以独立的应用程序形式面向企业不同层次的使用者。

将 UniFlow 的各模块和组织管理模块进行集成后如图 3.9 所示。

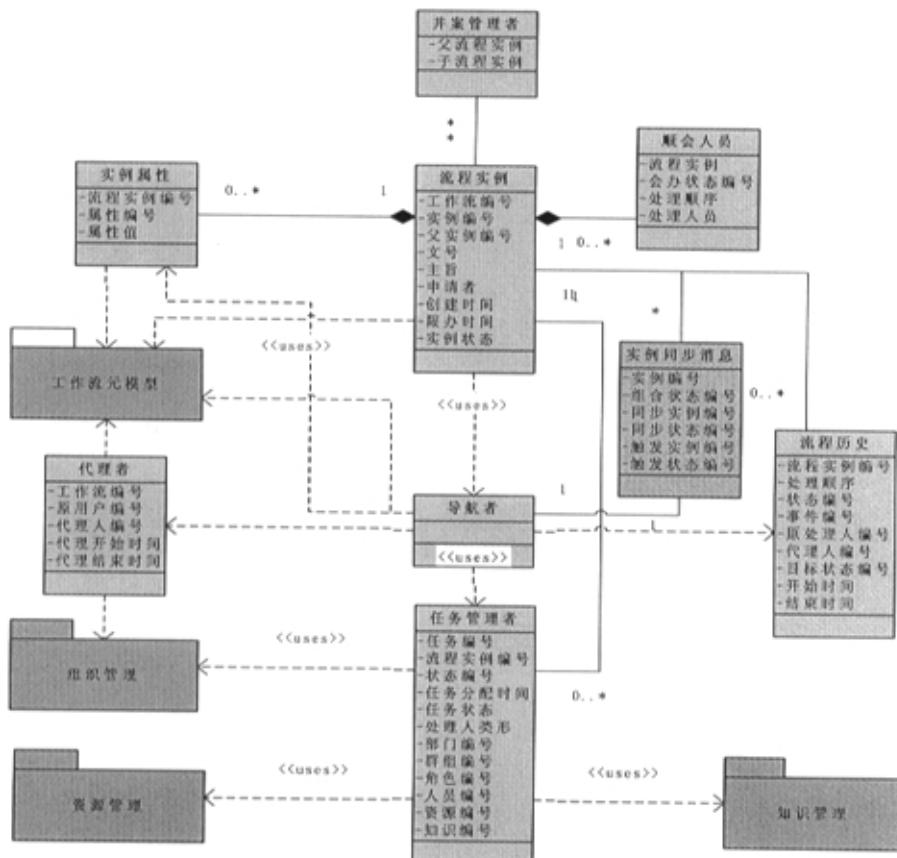


图 3.9 UniFlow 的集成

由图 3.9 可以看出，整个系统运作过程如下：(1) 工作流模型的实例化与提交；(2) 工作流机启动活动实例并推动过程，同时申请相应的企业资源与人员；(3) 活动的参与者通过自己的用户界面与工作流机交互，包括通知工作流机活动的开始、结束以及获取内容等操作；(4) 管理人员通过工作流管理工具对企业所有运行的工作流进行监控，并处理相关的异常情况；(5) 集成平台的运控代理为整个工作流实施系统提供底层通讯支持，包括屏蔽异构环境和消息传递等。

然而，在电子政务的架构中，对于 UniFlow 的调用者来说，它不需要知道过多工作流系统实现的细节，它只关心感兴趣的接口。这样，目标类对调用者来说就是一个黑盒子。如果接口是稳定的，那么，实现再怎么扩展，对客户来说也不会有很大的影响。所以，我们专门设计了 UniFlowWrapper 类，屏蔽了 UniFlow 系统的工作流的具体实现细节。电子政务系统的其它功能模块在需要时，通过调用 UniFlowWrapper 类来实现其功能。UniFlowWrapper 集成设计如图 3.10。

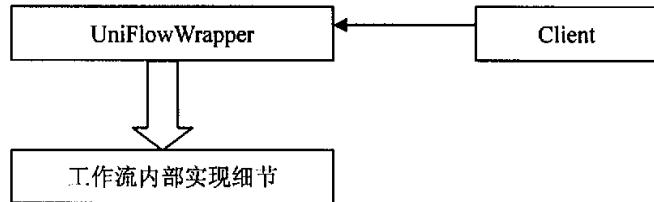


图 3.10 UniFlowWrapper 类的设计

显然，在技术上，根据 UniFlow 系统的实际运作情况，UniFlow 应采用先进网络计算架构及二层 B/S 架构，这样将使用户界面、业务逻辑、存储机制分开，让系统更容易维护及管理。而且，UniFlow 要以目前最流行和最具发展潜力的 J2EE 平台和 Web 技术作为系统的底层支持，实现系统的异构、分布，使 UniFlow 在使用上不受操作系统及数据库的限制，在操作系统方面 UniFlow 支持 Windows 系列、Netware 系列、Unix 系列、Linux 系列操作系统。在数据库方面，UniFlow 支持 Oracle、DB2、SQL Server、Sybase、Informix、MySQL 等关系数据库系统。在 Web Server 方面，UniFlow 需支持 IBM WebSphere、WebLogic、Tomcat、JBoss 等 Web 服务器系统。在应用集成方面，UniFlow 采用 EJB 来封装工作流系统中的主要组件；UniFlow 用 XML、HTML、Applet、JSP、Servlet 来作为客户端界面。

由于我们在电子政务系统的架构的集成中始终都采用的是基于组件化的设计思想，UniFlow 也不例外。该系统在应用功能增加时，只要增加相应的新组件。这样就可使之适应以后业务扩展的需要。因此，在应用部署上，UniFlow 分为三部分（如图 3.11）：客户端应用，工作流服务器和数据库管理系统。客户端应用：主要通过浏览器（Browser）实现功能操作界面，如工作流定义、过程控制、过程状态查询、任务项管理、过程监控、数据处理、应用管理、组织资源管理等功能。在实现上主要是采用 HTML、XML、Applet、Servlet、JSP 技术。

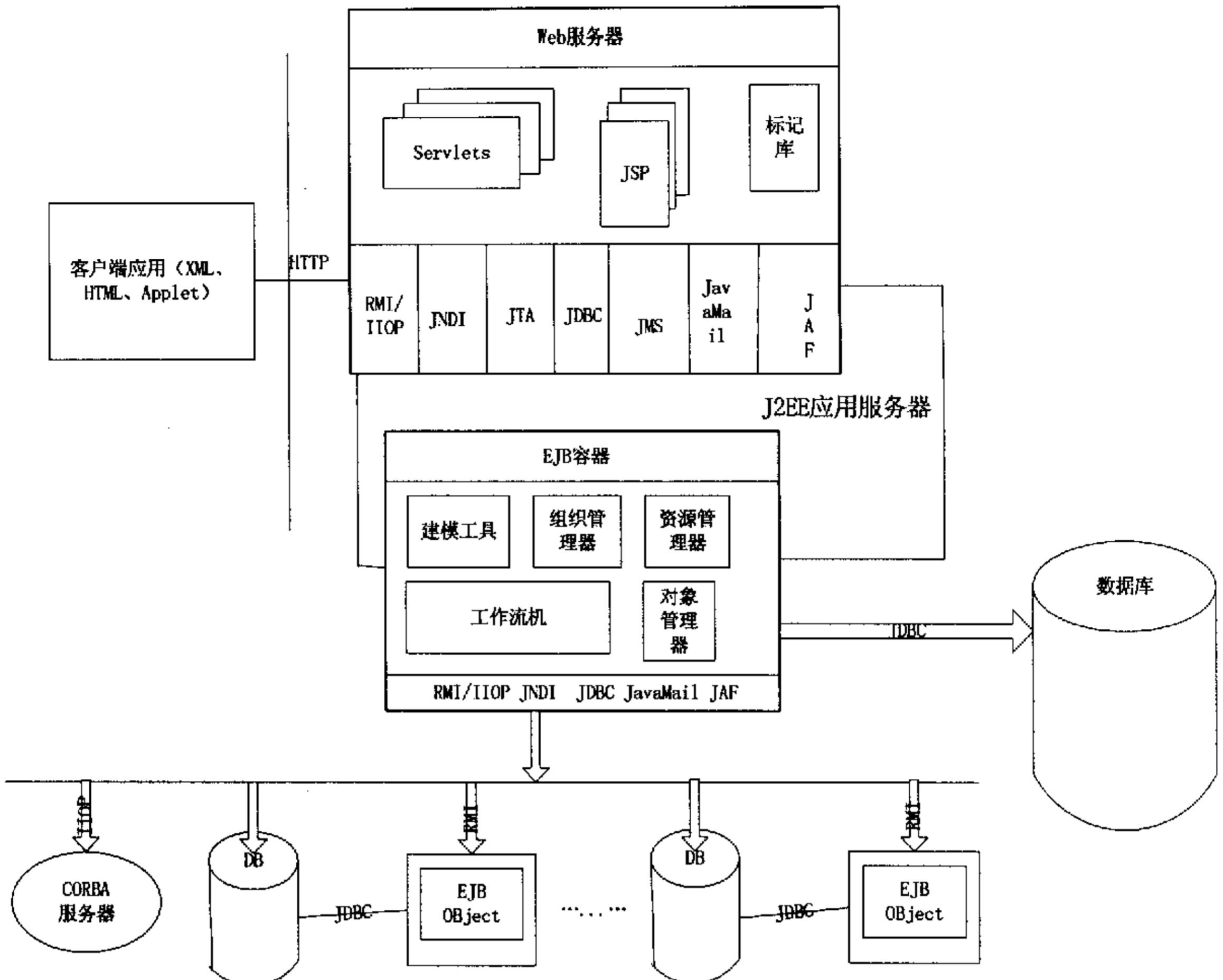


图 3.11 UniFlow 架构图

工作流服务器：UniFlow 采用网络计算的观念并采用 J2EE 技术，政府部门可以根据需要采用一台或多台异构的、分布的应用服务器实现系统功能，并可将执行任务需要的应用组件分布于各工作流执行服务器。用户通过 Browser 与 Web Server 沟通使用系统功能，后台大量的计算工作由 EJB 容器中的 EJB 组件和分布于各工作流机中的 EJB 对象协调完成。服务器之间与使用者之间都是以 TCP/IP 协议沟通，使用可以透过应用程序连接界面连接自己所需的工具与相关的服务器。

数据库管理系统：UniFlow 可以根据政府的实际需要选用一台或多台数据库系统，所选数据库系统可以是异种数据库系统。建议在总部采用 1 台 Oracle 数据库服务器存储 UniFlow 的各种数据，如工作流控制数据、相关数据、应用数据等。各下级机关也可以拥有自己的 Web Server 和相应的数据库，存储本级机关和下级机关的应用数据。

§ 3.4 小结

本文首先对工作流和 WFMS 及其关系进行了分析，提出了将 UniFlow 工作流管理系统作为电子政务架构的基础平台。其次，对 UniFlow 的集成进行了分析、设计和实现：将 WFMS 的 3 个基本功能模块拓展为 6 个功能模块进行了详细的功能集成分析；对 UniFlow 的工作流过程模型和组织管理模块进行了集成设计；通过对 UniFlow 系统的各模块和组织管理模块的集成、UniFlowWrapper 类的设计和对 UniFlow 架构的设计实现了对 UniFlow 的集成。

第四章 电子信息资源的集成

§ 4.1 政府的信息资源现状概述

目前，我国的绝大多数电子政务系统是以某一地区、行业或某一厂家的操作系统与系统集成技术为基础，使各系统自成体系，所带来的许多弊病严重影响电子政务的正常发展。这些弊病包括：资源分散、信息重复建设；数字资源种类繁多、应用系统各异；信息数据格式和存储方式多样，资源管理不统一等等。其结果导致了电子政务信息孤岛问题日益严重；另外，由于系统平台缺少功能强大的信息发布和检索工具，致使现在的资源和系统管理复杂化，从而制约了信息服务的功能和质量。这种信息孤岛使得原本一个主题的信息，被分成不相连、不相关的数据组织形式和存储地点，比如在政府部门里，公务员（主题）的信息可能分别存储在人事、财务等部门。而在行业系统中，这种情况更明显。这样，在原有这种信息孤岛式的系统下，就要不断地开发各种接口，如果组织规模较大，接口的数量就显得相当可观。这些接口的开发和维护的费用，以及系统的不稳定性，显然是相当惊人的。

因此，如何对电子政务架构中的应用系统和数据资源进行集成来架构一个统一电子政务综合处理平台、达到实现资源全面集成和共享，成为我国提高电子政务整体应用水平的关键环节。

§ 4.2 基于 UniFlow 的电子政务信息资源集成设计

§ 4.2.1 集成分析

要将云集软件公司在不同的时期所开发的各种政府办公应用系统架构在电子政务体系中，同样存在着信息资源集成的问题。针对这种现状，我们在进行电子政务架构设计时，将 UniFlow 作为电子政务架构的基础平台，并对数据信息资源的集成采取如下的策略。

一、资源与需求相结合

根据存储管理和发布方式的不同，基于 UniFlow 的电子政务系统目前的资源可以分为带标引属性的目录文件、多媒体文件、关系数据库数据等。这些数据重要性强、数据量巨大、关键数据多，存储和资源调用类型多样，资源集成具有很大难度。因此，对于目录文件库资源，我们彻底抛弃了原操作系统目录结构的管理方式，代之以集中的 UniFlow 数据库管理方式。对于其他数字资源，则本着集中、统一、实效的原则，相应地转移到 UniFlow 平台上，或将 UniFlow 和现有管理软件相结合。利用检索技术，该方案提供了前端的全文检索功能，具体内容包括以下二点：首先，针对带标引属性类文件，通过目录树的结构来组织和管理，目录或子目录表明该文件所属的类别。文件为纯 HTML 格式，但是在正文中带标引属性。目录结构较大，层次很多。UniFlow 针对数字图书馆解决方案将开发标引信息提取和转换入库工具。其次，对于多媒体文件，则通过目录树的结构来组织和管理，目录或子目录表明该多媒体文件所属的类别，在每个目录下有一个目录文件，文件中的每一行是该目录下的多媒体文件的标题或描述信息，实现对视频文件描述信息的检索和视频点播等。

二、强化信息检索服务

信息检索是电子政务应用系统信息服务的重要内容。我们通过 UniFlow 集成了各种数字资源，不仅大大加强了单个数字资源库的检索功能，包括全文检索、组合检索、二次检索、智能检索等等，而且还利用 UniFlow 实现了各种资源的跨库联合检索，用户可以通过一个检索入口在数个资源数据库中进行统一检索，从而带来了极大的方便。

利用 UniFlow 关系数据库网关，电子政务应用系统中的信息实现了集成与统一检索功能。这个系统的功能是集成各种各样异构系统的信息资源库。这些信息资源库目前处于信息孤岛的状况，通过我们的资源库系统能有效地重建成同构的、统一通过浏览器进行访问的信息资源库系统。

§ 4.2.2 应用系统的程序代码跨数据库平台

一、概述

我们知道，电子政务的各个应用子系统集成常常是很复杂的。这是因为它们所应用的关系数据管理系统之间有太多的结构差异。一般，用来描述 Oracle 结构的用词和术语通常与该词在 Microsoft SQL Server 中的意思大相径庭。而且 Oracle 和 SQL Server 都对 SQL-92 标准做了许多自有的扩展。尽管从一个应用程序开发人员的观点来看，Oracle 和 SQL Server 是以相似的方法来管理数据的。但在 Oracle 和 SQL Server 之间有着重大的内部区别。我们如果对于数据库的管理和设计得当，完全可以把这些区别对电子政务的集成的影响减到最小。

我们面临的最重要的信息资源集成问题是执行 SQL-92 语言标准和每一个关系数据管理系统提供的语言扩展。有一些开发人员只使用标准的 SQL 语法，以便尽可能地保持程序代码的普遍性。通常，这种方法把程序代码限制在 SQL-92 标准的登录级别上，而这个级别是被许多的数据库产品实现了的，包括 Oracle 和 SQL Server。这种方法将会产生一些不必要的程序代码复杂性而且还会对程序的性能造成很大的影响。例如，Oracle 的 DECODE 函数是一个非标准的 SQL 扩展。Microsoft SQL Server 的 CASE 表达式是一个超越了登录级别的 SQL-92 扩展，而且在所有其它的数据库产品中都没有实现。Oracle 的 DECODE 和 SQL Server 的 CASE 都是可选的，我们可以不用这两个函数而实现它们的功能，但这需要从关系数据管理系统中提取更多的数据。同时，对 SQL 语言的程序扩展也会造成困难。Oracle 的 PL/SQL 和 SQL Server 的 Transact-SQL 语言在函数上是相似的，但是在语法上不同，在两种数据库和程序扩展中间没有明确的对称性。因此，我们可能会决定不使用像程序和触发器这样的存储的程序，这样所造成的损失是很大的，因为它们提供了别的任何方式都无法实现的性能和安全性上的优点。

二、程序代码跨数据库平台

程序代码跨数据库平台主要表现在数据表字段的数据类型的选用。

Oracle 数据库之字段主要的类型有：VARCHAR2、NUMBER、VARCHAR、DATE、INTEGER、BLOB、CLOB。而数据库 SQL Server 之字段主要类型有：INT、Datetime、VARCHAR、BINARY、FLOAT、NUMERIC 等。

通过分析比较多种数据库系统（Oracle 和 SQL Server 等），为使程序支持多种数据库的需要，我们使用指定数据库字段类型：长整形、字符串型和二进制型。做法如下：

- (1) 浮点型字段类型用字符串型代替。
- (2) 日期型用长整形代替，如果精确到天，则用一个字段 8 位长整形替代，如 20020203，如果精确到分钟，则用 2 个字段 8 位长整形替代，第一个是精确到

分钟，结构同上，第二个采用 24 时制并且精确到 0.01 秒，如 21533203。

- (3) 取消自动递增的字段，因为不是每个数据库系统都支持递增类型字段
- (4) 不同的数据库采用不同的建表 SQL 语句和数据初始化 SQL 语句。
- (5) SQL 语句采用通用语句，尽量不用数据库函数。
- (6) 新增一个处理创建表和修改表的转化方法，主要是转化数据字段类型，输入参数为 SQL 语句和数据库类型编号，输出为相应数据库的正确 SQL 语句。输入的 SQL 语句只包含三种规定的数据类型，即长整形、字符串型和二进制型。通过语句中的这两项规定，相同的建表语句就可以转化为对应不同的数据库类型的 SQL 语句，以达到跨数据库平台的目的。

(7) 考虑多数据库的事务处理的可行性，根据多数据库更改数据库连接程序，如果不能使用事务处理的数据库，禁止使用事务处理。

3. SQL 函数的使用

通过对 Oracle 和 MSSQL 数据库的比较，我们采用其中通用的 SQL 函数，如：数学函数中的 COUNT (*) 等；字符串函数、日期时间函数、操作符及 SQL 基本语句。

4. SQL 语句的综合处理

由于原有的政府应用系统所用的数据库不尽相同，为了达到电子政务系统的跨数据库能力，我们在集成电子政务架构的过程中专门归纳了对数据库操作的 SQL 语句：

- (1) 基本的数据操作 Insert, Update 语句及一般从用户表中检索数据的 Select 语句。

例：Insert into tDoc (DocID) Values(1)

Update tDoc set DocName = 'Test' where DocID = 2

像这种类型的 SQL 语句可以经过解析转换后输出，可通用方法设为 AnsiString ConvertSql(AnsiString sSql)，大致如图 4.1 所示。

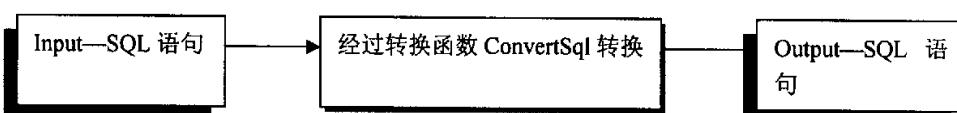


图 4.1 基本的数据操作语句处理

其中，Input 的 SQL 是某一特定类型数据库的 SQL 语句，经过处理转换后，输出的是与数据库无关的 SQL 语句。所以可能在转换函数中，要进行 SQL 语句的解析，获取字段的数据类型，根据数据库类型，进行不同的处理。

- (2) 从用户表中选取数据：获取数据表，字段信息等 SQL 语句。

TStringList *GetFieldNamesByTable (AnsiString sTableName)，通过一表名获取此表的所有字段名列表；

AnsiString GetFieldTypeByFieldName (AnsiString sTableName, AnsiString sFieldName)，通过表名及字段名获取字段类型；

Bool GetFieldNullableByFieldName(AnsiString sTableName, AnsiString sFieldName)，通过表名及字段名获取当前字段是否可为空；

Int GetFieldScaleByFieldName(AnsiString sTableName, AnsiString sFieldName)，若字段类型为数值型，则获取该字段的小数位数通过表名及字段名获取当前字段的精度。

以 Oracle 为例：

Select DATA_SCALE from USER_TAB_COLUMNS where TABLE_NAME = 'tDoc'
and COLUMN_NAME = 'DocID'

像这种类型的 SQL 语句不同上述第一种类型的处理，而应输出方法。

所以可做一个类用于封装这些信息。另外，上述只是当前涉及到的情形，更多的我们将对其进行不断地完善。

§ 4.2.3 应用系统的信息显示格式转换

在电子政务的集成设计中，由于有些应用系统是在繁体下开发的而有些是在简体系统下开发的；另外在日历记时上，台湾的政府机关常常采用的是民国纪年法，这和公元纪年法存在着矛盾(公元年=民国年+1911)。这些不可避免地导致信息转换问题。

针对信息展示的乱码问题及时间显示问题，我们在电子政务的集成过程中专门设计了相应的公用组件 ExtendString 类，如图 4.2 所示。

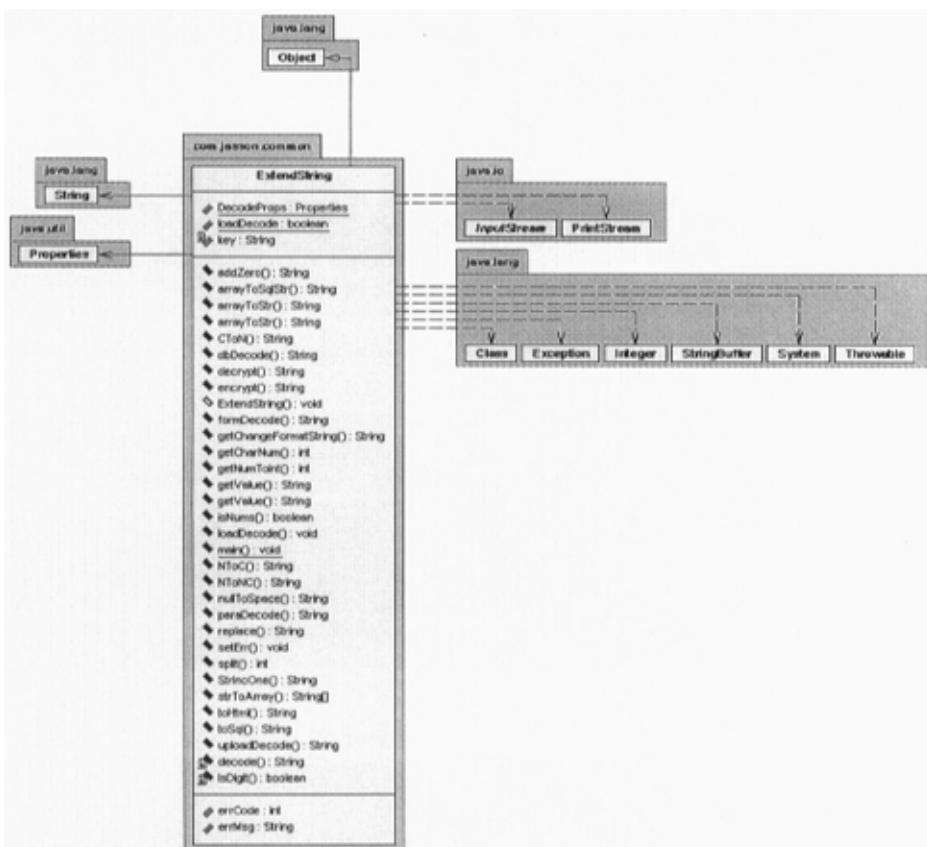


图 4.2 公用组件 ExtendString 类的设计

Jsp 页面在对于相关的信息进行处理时，只需调用 ExtendString 类相应的方法进行处理即可。

同样，我们针对日历转换问题也做了同样的处理：专门设计了日期转换类 ExtendCalendar，见图 4.3 所示。

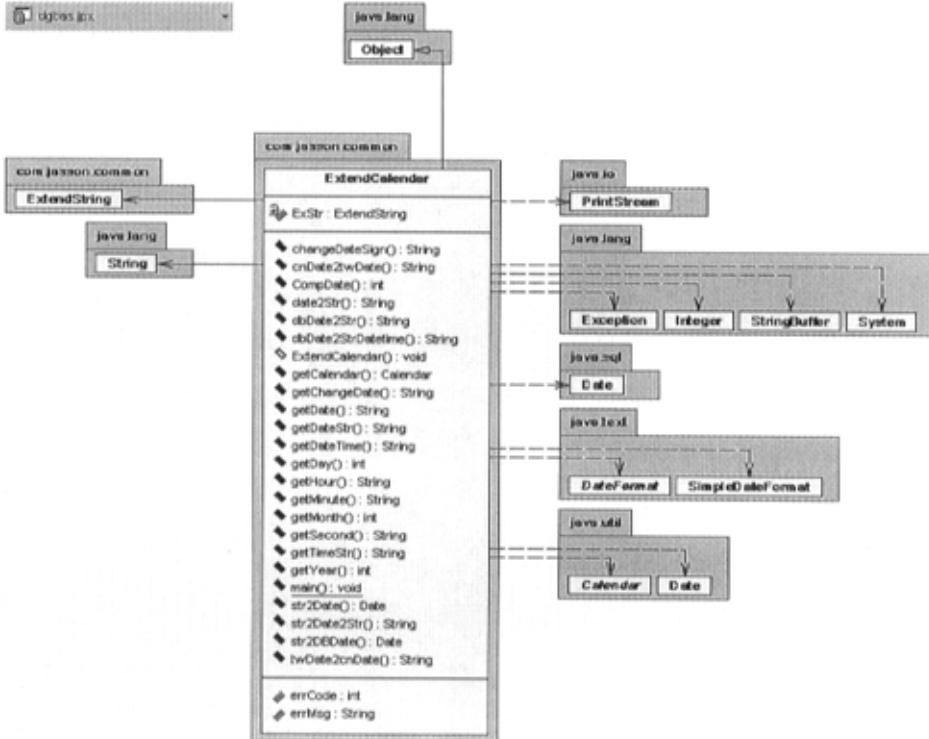


图 4.3 公用组件 ExtendCalendar 类的设计

§ 4.2.4 异构数据源的集成

从上面的分析可以知道，在政府的各个应用系统中，不可避免地存在着异构数据源：

(1) 各种系统所导致的数据源异构。数据源所依赖的应用系统、数据库管理系统乃至操作系统之间的不同构成了系统异构。

(2) 数据源的存储模式异构。一般的存储模式包括关系模式、对象模式、对象关系模式和文档嵌套模式等几种，其中关系模式为主流存储模式。需要注意的是，即便是同一类存储模式，它们的模式结构可能也存在着差异。例如 Oracle 所采用的数据类型与 SQLServer 所采用的数据类型并不是完全一致的。

(3) 各种信息资源所导致的语义冲突。由于信息资源之间存在着语义上的区别，而这些语义上的不同可能引起各种矛盾，从简单的名字语义冲突（不同的名字代表相同的概念），到复杂的结构语义冲突（不同的模型表达同样的信息）。语义冲突会带来数据集成结果的冗余，干扰数据处理、发布和交换。

针对上述的数据源异构问题，在进行电子政务应用的集成过程中，我们采用基于 XML 的三层 C/S 模型：把 XML 作为一种结构化信息交换的表示方法，负责所有与数据源的通信，根据最终用户的要求，聚集和组织从多个远程数据库服务器上获取数据，并用 XML 形式交互式的把数据源返回的消息传递给客户机。其过程详述如下：

首先，从数据层获取数据生成 XML 文档；由于 XML 能够使不同来源的结构化的数据很容易的结合在一起，可以在中间层服务器上对从后端数据库和其它应用处来的数据进行集

成。虽然每个数据库描述数据都是不同的，但 XML 可以自己定义文件标签，例如：通过 XML，公文就可以很容易以标准的方式按照名称、发文机关、收文机关或其他的标准进行分类。

其次，XML 数据的发送与刷新：利用 XML 开放的、基于文本的格式，可以将它通过 HTTP 像 HTML 一样传送。同时，中间层应用服务器将支持 XML 更新功能，将数据服务器上数据的变化及时地传递给客户，反之亦然。因此，中间层能够从客户端得到更新的数据，并把数据传送到数据储存服务器上。

最后，XML 数据在表示层的显示：XML 描述的数据发到表示层后，能够用多种方式显示。XML 定义的数据，其显示与内容是分开的，因此允许对同一数据指定不同的显示方式，使数据更合理地表现出来。CSS 和 XSL 为数据的显示提供了公布的机制，本地的数据能够以客户配置、使用者选择或其他标准决定的方式动态地表现出来。

因此，我们利用中间件（DAO 组件）集成各种异构数据库，这样不需要改变原始数据的存储和管理方式。DAO 组件位于各种异构数据库系统（数据层）和应用程序（应用层）之间，向下协调各数据库系统，向上为访问集成数据的应用提供统一数据模式，和数据访问的通用接口。各数据库的应用仍然完成它们的任务，DAO 组件系统则主要集中为异构数据源提供一个高层次检索服务。一方面，我们为中间件系统选择一种全局的数据模式。这种全局模式满足：（1）能够描述各种数据格式，无论其是结构化的还是半结构化的，无论其是否支持所有的查询语言还是简单的文本查询。（2）易于发布和进行数据交换，集成后的数据可以方便的以多种格式发布和便于应用交换数据。另一方面，利用 XML 技术，并将 XML 技术与全局数据模式相结合使异构数据源集成成中间件系统能更好地适应于开放、发展环境（例如，企业的动态联盟环境）中的数据集成。如图 4.4 所示。

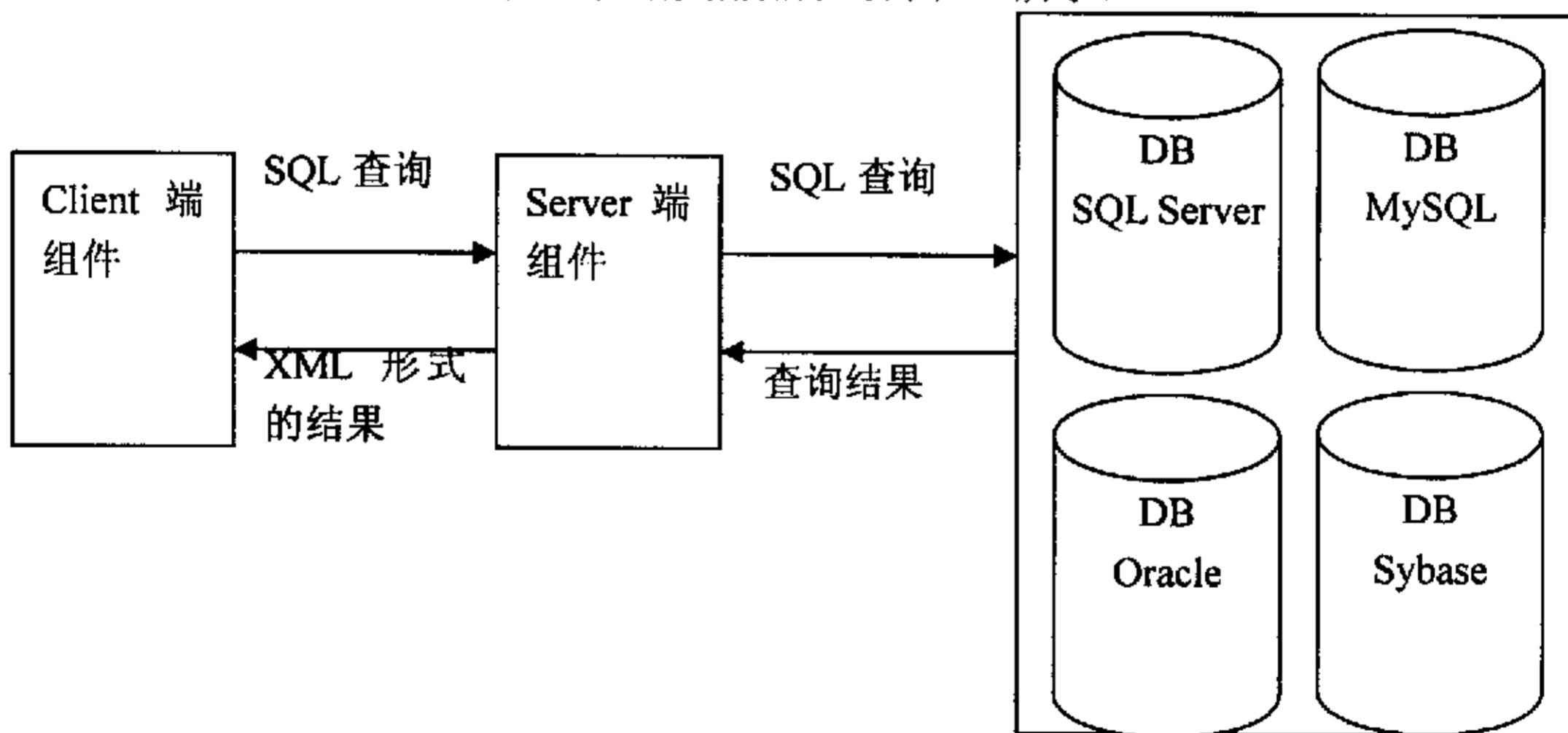


图 4.4 异构数据源的集成

§ 4.3 小结

本章通过对政府部门所存在的信息资源孤岛现象的分析，指出集成电子政务架构内各应用子系统和数据资源的重要性。并针对云集公司的各个应用系统所存在的信息资源集成问题所进行的分析研究，从电子政务各应用子系统的程序代码跨数据库平台、电子政务各应用子系统的信息显示格式转换和政府异构数据源三个方面对电子政务的信息资源进行了集成设计。

第五章 电子政务门户（GIP）的集成设计与实现

§ 5.1 电子政务门户（GIP）概述

门户（Portal）又称为入口，一般指集中了大量信息和服务的大型网站。电子政务门户（以下简称 GIP）是指政府对公众和企业信息服务和政务服务的入口网站，它集成了电子政务服务的应用功能。通过 GIP，一方面，政府可以为公众和企业提供及时、丰富的政务信息和所需的政务服务；另一方面，公众和企业可以随时随地查阅政府公布的各种政策信息，包括各种申报，甚至参政议政。可以说，GIP 就是政府“对外宣传的窗口、对外合作的窗口、为人民服务的窗口、接受人民监督”的窗口。它是电子化的单一窗口，民众或企业进入政府入口网站后，可以轻松按图索骥，连接到各政府相关部门，以取得所需的信息或完成洽办的事务。其模型如图 5.1 所示。

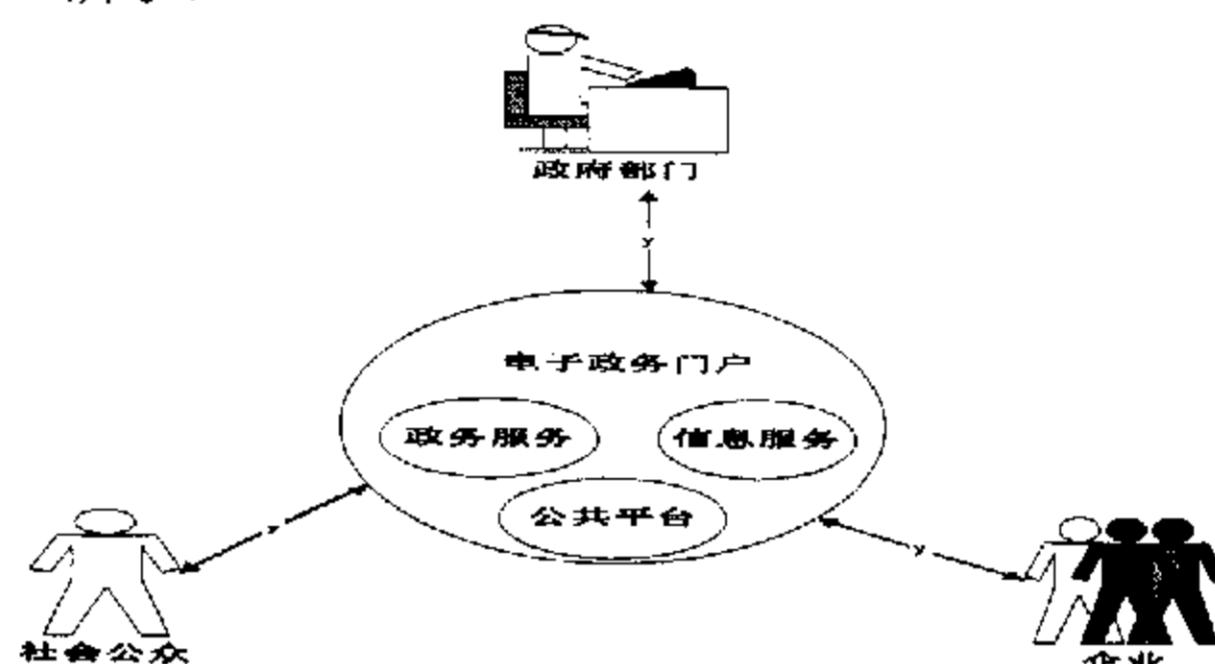


图 5.1 GIP 示意图

可以看出，GIP 主要服务内容为信息查询、网络申办和政府与民众双向沟通三大功能。其中，在信息查询方面：提供先进网络搜寻引擎，检索各机关网站文件；分类目录查询各机关网站；提供各机关通讯录资料；透过 GIP，可对民众进行政策宣导，以增进民众对政府施政的了解及支持。在双向沟通方面，提供公共论坛、网络民意调查、电子民意信箱索引系统、政府新闻网及机关每日活动行事历，扩大民众参与公共事务及反映民意。

总之，GIP 和电子政务的架构有着密切的关系，电子政务是政府实现政务公开的一种重要形式和发展方向，从公众的角度来看，对电子政务的关注主要是面向公众的信息和服务，作为电子政务系统架构的核心，电子政务的内容和功能是通过 GIP 的形式表现出来的，用户可以个性化地访问政府信息与服务。因此一个优秀的 GIP 网站是电子政务取得成效的基础，也是“电子政府”的核心组成部分和对外宣传窗口。

§ 5.2 世界发达国家和地区的 GIP 发展概况

许多发达国家和地区的政府为了提供人民不受时间地点限制的便利服务，进而提高政府行政效率，降低政府运作的成本，因此将 GIP 作为电子政务建设的重点。

新加坡财政部与信息通讯发展局于 1999 年推出电子化政府入口网站—eCitizen(<http://www.ecitizen.gov.sg>)，其中包括 170 项电子化政府服务。2001 年 10 月，将电子化政府服务增加至 560 项，分为 14 个类别，包括家庭、购屋、教育、健康、就业、商业、体育、旅游、选举和国防等。

加拿大政府认为，由政府率先 e 化将可加速民间企业 e 化的脚步，进而促进国家经济的

成长，因此加拿大积极进行电子化政府建设。加拿大于 2001 年 1 月推出政府入口网站 <http://www.canada.gc.ca>。加拿大政府还计划于 2004 年时让所有的政府信息和服务都可透过网络取得或申请，使国民和外国人不受时间和地点的限制，随时随地都可与加拿大政府取得联系。

香港政府以建构、经营、移转(Build、Operation、Transfer, BOT)的方式，鼓励民间企业积极参与公共服务电子化系统的建置计划，藉由民间企业建置并营运政府入口网站，以刺激各界进行运用。香港政府期望在 2003 年以前，将 90%适合于网络上提供的服务电子化。由康柏(COMPAQ)、和记黄埔(Hutchison Whampoa)和 Asia Global Crossing 合资组成「ESD 服务有限公司」，其中康柏是 ESD 电子基础设施、服务和解决方案的主要供货商。ESD 公司于 2002 年推出「ESD life 生活易」 (<http://www.esdlife.com>) 网站。

2000 年 9 月美国政府成立政府入口网站 FirstGov.gov (<http://www.FirstGov.gov>)，通过政府入口网站中的搜寻引擎，民众可轻松查询超过 2,700 万网页的政府信息。美国政府的线上服务包罗万象，国民可通过各种政府网站取得服务，例如税务(irs.gov)、教育(students.gov)、资深国民 (seniors.gov)、医疗保健(medicare.gov)等。此外，美国政府着手推动「政府书面文件消除计划」(Government Paperwork Elimination Act, GPEA)，预计于 2003 年将政府公文档案全面无纸化，而以电子文件取代，并采用电子签章。

台湾电子化政府入口网站由行政院研考会负责推动，2002 年 3 月正式推出「我的 e 政府 · 电子化政府入口网」 (<http://www.gov.tw/>)，作为政府网络服务的单一窗口，提供各种便捷的目录服务、生活信息及网络申办等服务。「我的 e 政府」网站提供近 1,100 项窗体下载及近 200 项网络申办服务，民众可透过分类目录检索各机关 4,800 笔以上网站信息，3,000 笔以上各机关通讯名录资料、政府重要施政信息和基本国情信息等。

§ 5.3 我国政府网站建设现状

与信息技术发达国家相比，我国的电子政务尚处于起步阶段。我国政府网站还不能提供“一站式”服务。从国外政府网站的“一站式”服务模式来看，网站所有服务内容和功能都是以用户为中心进行设计和分类的。而我国政府网站的服务模式仍然是以部门来分类服务项目，要办理多个事项就必须分别访问多个政府直属部门站点，这就增加了办事环节，影响了工作效率，也不便于公众充分了解政府网站的服务内容和信息。另外，我们常常看到，有相当的政府网站有名无实，信息更新不及时；有的没有深入链接，甚至变为“死网”。民众或企业根本就无法获得快捷、方便的服务。这样的政府网站实际上只是政府的摆设，它并没有起到作为政府入口网站应有的价值。

不仅如此，由于各地政府甚至同一地方的政府各个职能部门所用的应用系统都有所不同。这些应用系统都是政府在不同时期根据当时的具体需求而定制的，如具体行业的办公、财务应用软件等，所以导致的政府应用系统长期处于孤立状态而未能进行有效的集成。至于政府网站的所谓 GIP “一站式”单点登录就无从谈起。因此，加快开发我国的政府入口网站就显得尤为重要。

§ 5.4 基于 UniFlow 的 GIP 集成设计与实现

§ 5.4.1 GIP 的集成目标

我们所进行的 GIP 集成设计目标就是将原有的各种应用系统集成在电子政务应用架构的 UniFlow 基础平台上，从而实现政府“办公自动化、信息网络化、资源集成化、业务流程化、决策智能化、界面个性化、模块组件化、接口标准化”的目标。

为此，我们在进行 GIP 的集成时，首要的任务是透过软件设计结合管理政府机关人员清单，根据其角色的不同对各项应用系统使用权设定，进而达到建立电子政务平台管理基础，并在此基础上建立单一签入与系统权限管理制度，最终具备下列系统功能：

1. 账号管理

- (1) 许可管理账号（含新增、删除、修改、停用），并提供软件及数据权限设定。
- (2) 构建政府组织架构图，可供窗体流程连结，并需可搭配职务权限设定。

2. 单一入口权限管理机制

- (1) 建构于单一入口平台上之应用系统，需可依平台账号建构关联之管理设定软件使用权管理接口（AP 管理权限查询）。
- (2) 管理接口需可供后续开发平台的系统权限导入设定接口。
- (3) 使用权限设定可设定平台使用者账号应用系统的权限，例如：可使用、不能使用、停用等。

§ 5.4.2 应用系统的集成设计

一、E-Form 电子表单系统集成到电子政务架构中概述

首先，政府部门作为一个特殊的服务性行业，常常离不开政务处理过程中信息的收集、传递、整理、查询和统计分析。基于此，表单就应运而生。在过去，纸面表单的处理、保管和查找的成本都很高。如今，随着计算机技术和网络技术的发展，在政府的各种办公自动化系统中出现了电子表单，它突破了时空的限制，通过工作流系统就可以提交、传递，完成相应的流程数据处理，使信息的传递和数据的处理获得了最大的自由度和效率。

其次，云集公司所开发原有的针对 OA 产品的 E-Form 电子表单系统（以下简称 E-Form）不能实现跨平台的要求，表现在以下三个方面：首先，系统采用 Borland C++ Builder(C++) 开发，基于 Windows API；其次，数据库相关的操作采用 ActiveX / COM 结构，以及 BCB 的 MIDAS 的多层次数据库应用解决方案；最后，系统文件操作结构。窗体文件的上传采用 Socket 流的机制，路径参照 COM 应用程序。而下载则采用 HTTP 机制（Windows API），路径参照一指定的 HTTP URL。

在 UniFlow 工作流系统中，要运行一个工作流程，就必然离不开电子表单。另外，E-Form 系统只是一个工具软件，它只有集成在 UniFlow 系统中才体现其应用价值。因此，作为一个完整的电子政务应用架构的 GIP，就要有效地将电子表单加以集成。而 E-Form 集成到电子政务架构中应该具有以下的功能：

- (1) 系统需可跨平台，未来可移植至 Linux、Unix、NT、Windows 2000 操作系统上。
- (2) 提供符合 XML 格式的表格及数据的表单。
- (3) 表单结合签核机制：窗体产生者所产生之窗体须结合在线签核系统，提供审核机

制，并记录所有在线签核历史资料。

(4) 表单可设定其使用期间。

(5) 经核定后的表单则不允许修改。

(6) 回收汇总资料后，对可统计的问题有统计、打印、报表及提供各式数据汇出等功能。

(7) 表单管理者可管理所有表单，使用者并可有个人模板表单。

(8) 表单签核流程可更动设计。

(9) 授权及指定代理人的功能。

(10) 需提供个人签核文件流程现况查询。

(11) 提供弹性表单流程，即使用者可快速设定传递的对象，可顺会或平行分会，并可随政府部门的需求改变流程，无需重新撰写或修改程序，让信息系统能快速响应部门内外环境的改变。

(12) 提供政府办公流程所需的表单。如出差请假单、加班申请单、物品请领单等。

二、DocMan 公文管理系统集成到电子政务架构中的概述

公文是政府赖以办公表达其意图必不可缺的媒介，当今处于网络化、信息化和知识化的时代，为了进行快捷高效处理日益增多的大量纸本化公文，政府部门迫切地需要一个能够安全迅捷地自动化处理公文的应用管理系统。这是公文电子化的必然趋势。

DocMan 公文管理系统（以下简称 DocMan）作为电子化公文的管理系统，公文采用相同的规范；一旦公务员成功登录系统后，系统具有显示当前人员待处理公文并对于即将逾期办理的公文进行稽催管制等等，这对于政府的公务活动具有极其重要的意义：首先，政府部门实行无纸化办公可以节约大量的人力和物力。其次，实行电子化公文操作，使公文的处理具有高效性、快捷性和安全性，电子化公文在网络上的传输省去了纸本化公文在政府实体间的周转，并大大减少了人为的失误。第三，电子化公文的实现有利于体现政府处理公文过程中的清正廉洁，因为政府公务员始终面对的是电子公文而不是传递公文的人。第四，DocMan 便于政府部门对于公文进行管理。

电子政务作为政府部门的应用架构同样要把 DocMan 加以集成并达到以下的功能：

(1) 公文处理时的权责做到认职不认人。

(2) 系统应采模块化设计，所有的处理服务模块要开发为跨平台的组件，由前端表现层（presentation layer）调用后执行。此组件可定义作业模式，由使用者视需要设定参数。

(3) 实现基于电子公文的实时性。当有新传送的公文时，使用者若在线，画面信息可立即自动更新，并显示来文提示信息，此实时通知和及时更新的作业模式采用主动置放的技术（Web Broadcasting）。

(4) 每一份公文的处理，系统要自动检查该使用者目前可处理的选项作业，而且仅显示可执行的作业项目，以避免人为操作失误。

(5) 为充分达到计算机化控管的目标，公文处理流程每一个环节，包括并案数据的清单、会办单位设定顺序、公文承办单位、承办人员、签收时间、送件时间及创稿档案等信息，均应有明确的纪录，并可查阅公文处理流程。

(6) 要有弹性代理人制度设计，可依需要设定代理人代为处理公文，且公文流程须记录由何人代理。

(7) 每一操作画面皆需提供在线辅助说明(Online Help)，此说明采用 HTML 格式。在现有系统架构下开发厂商须提供未来与其它应用程序集成的接口与信息交换的标准。

(8) 提供公文本文及附件储存于个人计算机、网络磁盘驱动器，并记录公文稿的送件人、文别、公文主旨、建立时间、附件数、附件档案。

(9) 数据可储存至服务器数据库中，可依公文属性查询自己的文件，并提供依建档时间范围、更新时间范围、全文检索主旨找出自己的档案和列出清单。

(10) 提供汇入汇出其它公文制作系统产生之标准交换用 xml 档案，并可汇入汇出文字文件。

(11) 需提供电子签章及流程控管与现有公文制作系统结合之规划

三、Ucan 报表系统集成到电子政务架构中的概述

所谓的报表就是把一系列相关的数据信息放在具有固定格式的固定位置上的表格栏位中并输出给用户。这些数据常常是通过直接查询、或将查询到的数据进行某种运算得到的统计。在电子政务应用系统中，报表系统将分散的政务数据信息集中起来，进行重新加工整理，为用户提供全方位的数据信息服务。而且报表工具是实现信息共享、传输和发布的主要方式：一方面，对用户来说，他所需要的是直观的数据库信息，而不仅仅是查看单个的数据；另一方面，对于政府部门的领导或有关人员来说，报表能够方便地进行行政管理和有关决策的重要依据。

Ucan 报表系统（以下简称 Ucan）是云集所开发的 WEB 化智能快速报表工具。作为电子政务应用系统就必然离不开报表工具。所以我们在进行电子政务系统架构的设计时，把 Ucan 集成在 UniFlow 工作流管理系统的基础平台上。

四、GIP 架构概述

从 GIP 架构到每一个应用程序都采用目前信息系统集成领域中先进的面向对象的模块化分布式结构设计，高层对象之间将采用统一协议联结调用和消息传递。这一结构可以最大限度地满足和适应政府的发展要求，通过系统定制选择、修改或增加应用逻辑对象，来满足新的政府需求，并迅速、合理、低成本地完成政府信息系统的开发和维护。

在将以上应用软件系统集成在 UniFlow 基础平台的过程中，我们利用先进的 UML 建模技术，紧紧围绕 Use Case，以 Use Case 为中心进行拓展。体系集成思想采用“系统架构的 4+1 视图”。如图 5.2 所示。

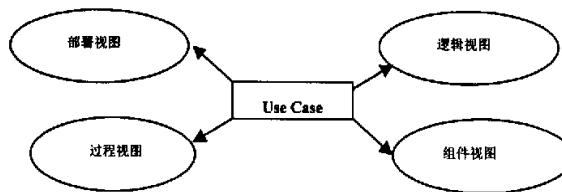


图 5.2 系统架构的 4+1 视图

基于上面的思想，我们在进行基于 UniFlow 基础平台之上的电子政务系统集成中，整体结构采用 MVC 模式：M（Model）采用 DAO 技术或者 JavaBean 组件技术对数据库进行操作；V（View）采用 Jsp 技术作为用户的表现层；C（Control）利用 Servlet 作为业务逻辑控制层。这样，应用与数据分离、编程代码与数据分离。业务逻辑是对业务实体进行组织的过程。分层只是将系统进行有效组织的方式。

在确定了系统框架和统一开发方针后，系统构件的开发就应充分利用已有框架所提供的服务和工具；并力求实现大粒度构件重用。通过系统构件的分层，将频繁变动得业务逻辑层分离出来，就实现了通用类构件的完全复用。并且由于接口的同一性，当某一模块业务逻辑改变时，使系统之间的影响最小。使系统实现即插即用，让系统容易升级。为此将 GIP 构件模型定义为四个层次。如图 5.3 所示。

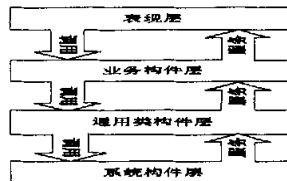


图 5.3 电子政务应用系统的层次架构方案

其中，系统构件层是指系统平台本身所提供的类库；通用类构件层提供系统中所需使用的通用函数类，如：字符串处理，数据库连接等类函数和业务处理过程相对不变的类，在本系统中，工作流平台、组织管理等通用组件将处于该层；业务构件层的作用是提供领域内所需使用的类库并进行业务处理服务，在该层的函数调用或继承通用类构件层的函数；表现层用来实现客户端界面，并调用业务层所提供的服务。

§ 5.4.2.1 E-Form 的集成设计与实现

一、E-Form 系统概述

电子表单系统是一个安装在服务器上，用来建立任意的 WEB 表单并能自动处理表单数据的表单系统，它可随意控制不同用户的权限，实现资料的保密性、安全性，可用于局域网或互联网，非常适合于单位内部或网站填写各种表单资料，建立内容丰富、查询方便的资料库。而在电子政务应用系统中，一般要求电子表单系统具有如下的功能：

- (1) 自动生成表单
- (2) 自动处理表单数据
- (3) 管理表单

E-Form 的作用就是为用户提供自定义表单工具，在电子政务应用系统中供用户自定义表单，结合电子政务中的工作流程定义功能，向用户提供自定义工作流，扩展应用系统的功能。工作表列出了与业务过程的参与者相关的一系列工作项，工作表处理程序则对用户和工作表之间的交互进行管理。工作表处理程序完成的功能有：支持用户在工作表中选取一个工作项，重新分配工作项，通报工作项的完成，在工作项被处理的过程中调用相应的应用程序等。为了达到这样的目的，我们将 E-Form 表单系统细分为三个子系统：表单设计制作、表单使用定制和表单浏览填写。各子系统间的关系如图 5.4 所示。

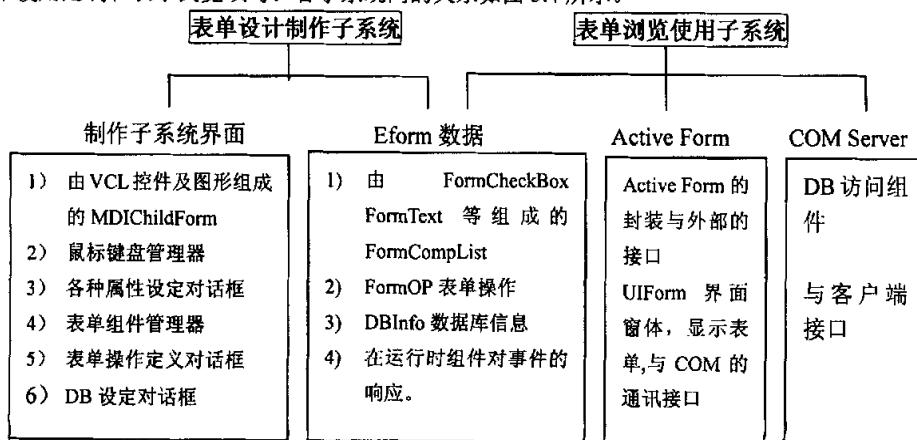


图 5.4 E-Form 各子系统间的关系

1. 表单设计制作子系统

表单设计制作子系统（eformMake.exe）指用户在图形化界面下，通过插入多种元件（如线条，文字，单选按钮，输入框等）及设置各种元件的属性及操作（如背景色，字体，尺寸、对应库表的字段等）来设计制作一种表单（如请假单），并保存为文件。eformMake.exe 运行在 windows 平台下，用户在图形化界面上进行编辑操作，最后生成电子表单的模板，用来制作表单，并将设计制作好的表单保存为文件（*.efm）。

在该系统的集成过程中，我们采用的一个重要的思想就是界面与模板数据分开。界面负责显示表单，负责响应用户的鼠标、键盘事件来编辑表单。而模板数据即 Eform 则负责保存我们关心的电子表单的结构及其属性等。

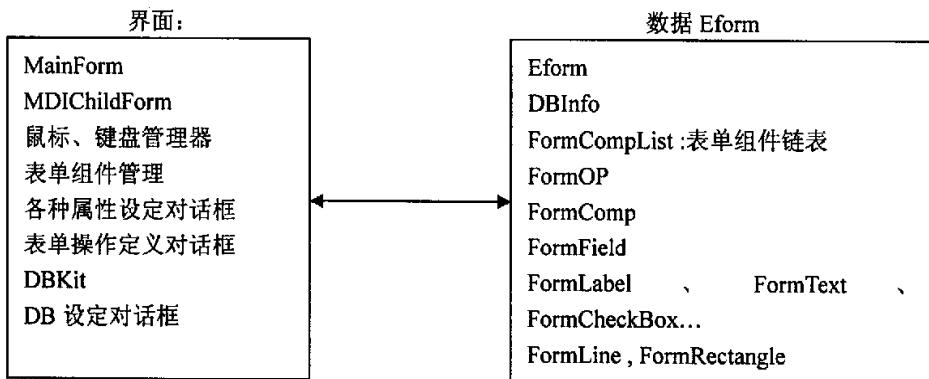


图 5.5 界面与数据相分离

其中，Eform 类存放表单的结构数据，MDIForm 界面则据 Eform 的数据来显示界面（外观），用户在 MDIForm 界面上操作，根据操作来修改 Eform 的数据。

2. 表单定制子系统

表单定制子系统是指表单设计好后，定义工作流的设计人员设置表单在一个工作流中每一个节点处的属性及在该节点处填写并提交表单后表单对应的方法，从而使表单与工作流程协同工作，完成一种工作流的定制。在一个工作流的不同节点，或在不同的使用场合（如查阅历史表单记录等），表单的某些特性有时会不同，主要体现为三点：某个表单域的读写权限，某个表单域是否隐藏，表单在用户按提交按钮时所对应的表单操作 FormOP。如一张请假单，申请人填写时前三个域可写（请假事由，时间等），后两个域隐藏（审批意见等），而在审批时，前三域可读，后两个域可写。在查阅历史时，所有域都只读。表单定制使用子系统就是解决这一问题的。解决方案时，用户根据需要设定表单中某些域的属性，并另存为一个文件，上传到服务器上。举例如下：

文件名	注解
vacation.efm	标准请假单：
vacation.efm.001	申请人填写时用，请假事由、时间、请假人三栏位可写，其它隐藏....
Vacation.efm.002	审批时用，前三栏位只读，审批人，审批意见栏位可写，提交时....
Vacation.efm.003	查阅时用，审批意见隐藏，其它栏位只读.....

这样，用户 IE 中浏览使用表单时，根据需要，使用不同的表单文件名（定制好的表单），即可满足需要。该子系统结构如下：

MainForm：主窗体，有菜单 FileOpen , FileSaveAs, FileExit

CustomizeDlg：设定对话框，传入 Eform 指针，取得其属性，设置对话框中控件属性，用户确定后，则将用户设置好的属性写入 Eform。

3. 表单浏览使用子系统(Active X + COM)

表单浏览使用子系统是指最终用户在浏览器（IE）中，阅读或填写某种表单（如用户填写一张请假单或部门主管阅读雇员填写的请假单等）。最终用户在 WEB 浏览器中使用电子表单，用户打开嵌有 Active Form 的 Web 页面时，Active Form 会下载到用户的浏览器端，用户指定要打开的表单文件后，Active Form 会解释表单文件，据表单文件显示设计制作好的表单，这样用户就可以在 WEB 页面中浏览填写 Active Form 中的表单。用户填写完后，按提交按钮，则由页面中的 JavaScript/VBscript 调用 Active Form 提供的提交方法，将表单中所填写数据写入到 DB 中。对 DB 的操作由 COM Server 完成，我们在此采用 DCOM 的解决方案，Active Form 作为 Client 端与 COM Server 通讯，而 COM Server 是三层结构中的 Application Server 层，它用 ADO 技术与数据库相连接，并访问 DB。

二、E-Form 的集成分析

E-Form 作为一种电子表单设计工具，在图形化界面下，提供一系列表单制作元件，使程序员及用户能轻松制作出电子表单。表单只是一种数据的载体。表单的设计工具相当于一个 class builder 工具，设计好的表单相当于一个类，表单的设计工作也即定义类的属性及方法。而表单的使用相当于对类的一个实例的属性的读、写（get/set）以及对类的方法的调用。为保证表单的通用性，设计好的表单作为一个类（包），外部调用程序与它的交互主要是通过读、写表单的属性及调用表单的方法。因此，在进行系统集成时，我们做到：

首先，我们采用逐步推进的策略，使系统先实现需求中列出的优先级别为 1 的功能。这些功能都是 E-Form 集成在电子政务架构中所具有的最基本的功能。根据设计文档划我们把改系统划分为几大模块，并根据开发成员的特点及擅长分配模块进行编码。当然在此编码前，为了便于系统的集成开发，在电子政务系统的架构和模型均已搭建好的情况下，各成员只须在此基础上编码。在每隔一个阶段，我们就会把每个成员的代码集成到一个系统中进行编译测试并出新的版本，在新版本的基础上，再进行编码，同时备份旧版本以备系统恢复。

其次，在系统实现优先级别为 1 的功能后，将系统挂接到 UniFlow 系统中进行测试，在挂接的过程中不断修改 Bug 同时完善系统，并从用户角度看 E-Form 须完善的功能并为下一阶段的开发提供一些建议。

最后，为了确保系统的开发进度和结果，在整个系统的开发过程中，制定开发计划并追踪计划。

三、E-Form 系统的集成设计

UniFlow 调用程序通过读写表单的属性或调用表单的方法来使用表单，E-Form 应能电子政务系统很好地结合。所以，E-Form 系统就要集成在 UniFlow 工作流平台系统中要能做到：创建一个流程时，在任何一个流程节点中能够选择已有的和创建新的表单，同时对该表单能够进行编辑；此表单能够随着该流程运转。要达到电子表单系统的这个最基本的应用，这就涉及到流程节点如何调用 E-Form 系统接口的问题，即：E-Form 系统如何和宿主系统集成的问题。

首先，E-Form 系统应能实现跨平台。为此，我们所采用的方法是：在物理上，增加一个 EForm 中间层服务器。EForm 客户端和中间层服务器只能运行于 Windows，而 WEB 应

用服务器及数据库服务器可跨平台。如图 5.6 所示。

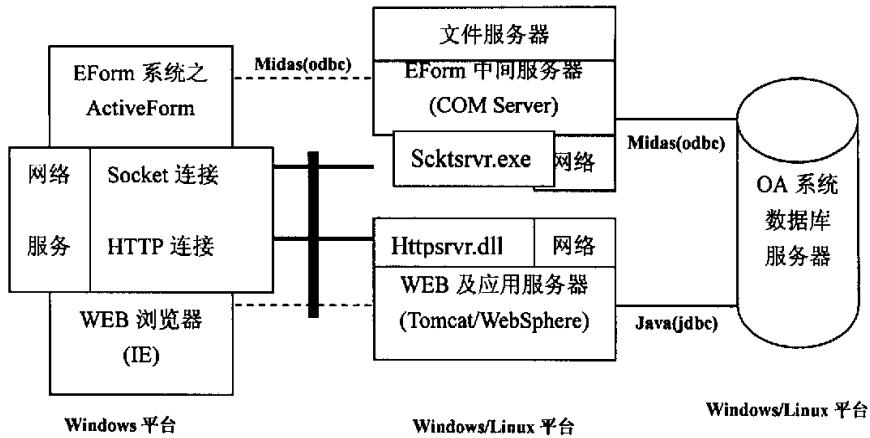


图 5.6 物理上增加中间层应用服务器时 EForm 的系统架构图

其次，我们专门设计了 E-Form 系统接口，如图 5.7 所示。

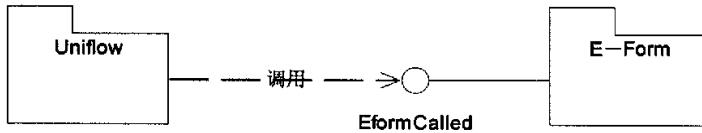


图 5.7 E-Form 系统和 UniFlow 系统的关系
上图中 EformCalled 类的详细设计如图 5.8 所示。

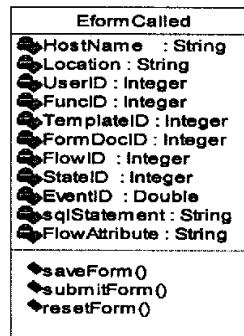


图 5.8 E-Form 系统的接口设计

EformCalled 类的说明：

1. EformCalled 类的参数

- 1) HostName: 部署 COM 服务器的机器名称, 例 “UCAN-GROUP”
- 2) Location: 放置表单模板文件的父目录的 URL 地址, 例如
“HTTP://UCAN-GROUP:8080/EFORM”

- 3) UserID: 当前用户的 ID 号
- 4) FuncID: 系统功能的编号,如表单定制,表单浏览等。在本系统中 1 表示表单管理,2 表示表单定制,3 表示表单使用
- 5) TemplateID: 表单模板 ID 号;
- 6) FormDocID: 表单实例 ID 号;
- 7) FlowID: 流程流号;
- 8) StateID: 流程状态节点编号;
- 9) EventID: 状态事件编号;
- 10) sqlStatement: 用于在表单和宿主系统之间传递数据的 SQL 语句;
- 11) FlowAttribute: 用于在表单和宿主系统之间传递数据的 Attr 列表。

参数传递情况一览如表 5.1。

2. EformCalled 接口类的方法

- 1) SaveForm(): 保存 E-Form

仅临时保存 E-Form 的数据信息(表单定制信息、表单用例数据)并不提交。即产生用于与宿主系统(OA、UniFlow 等)数据交换的 SQL 语句以及向宿主传递 FlowAttribute。

- 2) SubmitForm(): 提交 E-Form

提交 E-Form 的数据信息(表单定制信息、表单用例数据),外部在调用该方法完毕后即将 E-Form 数据更新到 DB 中。

- 3) ResetForm(): 重置 E-Form

将 E-Form 的数据信息(表单定制信息、表单用例数据),恢复到最初调用时的状态,类似于一般 HTML 表单中的 Reset 按钮,外部在调用该方法完毕后即将 E-Form 数据恢复到最初调用时的状态。

表 5.1 系统外部接口参数一览

子系统名称	传入参数	备注
模板管理	新增模板 HostName, Location, FuncID = 1	
	编辑模板 UserID,	
	删除模板 FuncID, TemplateID, SqlStatement:	
表单定制	新增表单定制 HostName, Location, FuncID = 2	
	修改表单定制 UserID,	
	删除表单定制 FuncID, TemplateID, SqlStatement, FlowID, StateID, EventID	
流程属性定制	同上	FuncID = 21
表单使用	新增表单用例 HostName, Location, FuncID = 3	
	修改表单用例 UserID,	
	删除表单用例 FuncID, TemplateID, SqlStatement, FlowID, StateID, EventID, FormDocID, FlowAttribute	

3. EformCalled 类中的 SQL 语句

SQL 语句用于在 E-Form 系统与宿主系统之间传递数据（如表单定制信息和表单用例数据），以在宿主系统中实现事务处理的一致性。SQL 语句（Insert 或 Update）中用指定的分隔字符来分隔界定所包含的字段名和字段。其中字段名前后由字符（ASCII 值 14）分隔，而字段值前后由字符（ASCII 值 15）分隔。

1) 传出 SQL 语句

在业务流程中，对于电子表单仅临时保存 E-Form 的数据信息（表单定制信息、表单用例数据）并不提交。

2) 传入 SQL 语句

仅临时保存 E-Form 的数据信息（表单定制信息、表单用例数据）并不提交。

4. FlowAttribute 字符串

用于 E-Form 系统与 UniFlow 之间传递属性数据，使 UniFlow（宿主）流程实现的属性数据能够适时更新。FlowAttribute 字符串中用指定的分割字符来分割各个不同的属性组。在属性组与组之间用 ASCII 码值 14 的分割。在 AttrID 与 AttrValue 之间用 ASCII 码值 15 分割。

5. 在 Applet 中调用 E-Form

一个大型系统中，采用多种平台技术开发是经常发生的，如宿主系统是采用 Java 等开发。因此本系统也考虑到了 Java Applet 的调用接口。

1) 定制流程属性

<http://UniFlow:8080/UniFlow/eform/CusomizeFlowAttribute.jsp?TemplateID=37&FlowID=1>

2) 定制表单

<http://UniFlow:8080/UniFlow/eform/CusomizeEFormVisible.jsp?TemplateID=37&FlowID=1&StateID=1&EventID=1>

四、E-Form 系统的集成难点及解决方案

在工作流设计过程中，在一个工作流中，需明确在某一节点使用哪一种表单，在此节点需对该表单的哪些元件的 visible 及 ReadOnly 属性进行设置，在此节点表单填写提交后调用哪个方法进行 DB 访问(存盘，实际上也即对表单命令按钮的 Action 属性的设置)。电子政系统对电子表单的调用有下列几种方案：

1. 程序中显式的访问表单的属性及调用表单的方法，例如

```
form1->Text1->visible=false;  
form1->saveButton->Action=method1  
form1->show()
```

此种方法不太适合在电子政务系统中用。

2. 在工作流程设计程序中，通过在工作流程设计界面上的操作，将工作流—节点—表单—表单中某些表单域的属性 (visible,ReadOnly)—表单在此节点提交后对应的操作 (SaveButton->Action 属性) 等上述参数存放到 DB 相应的库表中，由一个程序读取参数表来实现工作流。

此种方法可在电子政务系统中实现，但工作由 UniFlow 完成，一个问题是参数表的合理设计及驱动程序的合理设计，难度较大。另一个问题是工作流程设计的界面(供用户设定参数保存参数的界面)需进一步的探讨。

3. 设计好的表单如请假单保存在文件里，由一个 Active Form 控件来解释它。工作流程的设计过程中，在某一节点可能需设置表单域的若干属性(部分域的 ReadOnly, Visible,

SaveButton 按钮的 Action 属性), 可否将这些属性值写在表单文件里, 然后另存为一个文件(不同的后缀)如“请假单 1”, 这样工作流设置程序只要将工作流名—节点—对应的电子表单名(“请假单 1”)保存在 DB 的参数表中即可。此种方案的可行性需论证, 另外 WEBOA 工作流设置程序的界面需考虑, 同时表单属性设置对话框的界面也需进一步考虑。

§ 5.4.2.2 DocMan 的集成设计与实现

一、公文管理概述

1. 公文及公文管理

公文, 是行政机关在行政单位中形成的具有法定效力和规范体式的文书, 是依法进行政务管理和公务活动的必不可少的媒介。传统的政务办公一般都是以纸张公文作为媒介在政务机关之间进行流转传送的。然而, 随着计算机网络化和信息化技术的发展和普及, 行政机关的工作方式发生了翻天覆地的变化: 政府部门的公文趋于电子化, 从而达到政府部门的无纸化办公, 使公文传送的时效性适应现代化的节奏。为此, 行政机关日益迫切地需要一套完整的公文管理系统, 来满足公文自动化处理的需要, 以便提升公文办理绩效。显然, 一个完整的公文管理系统就是面向行政事业单位的日常公文处理的运用系统。

DocMan 主要包括: 公文制作、传递交换、流程管理、档案管理, 系统支持模块; 以及对 DocMan 作支持的工作流平台和组织管理模块。

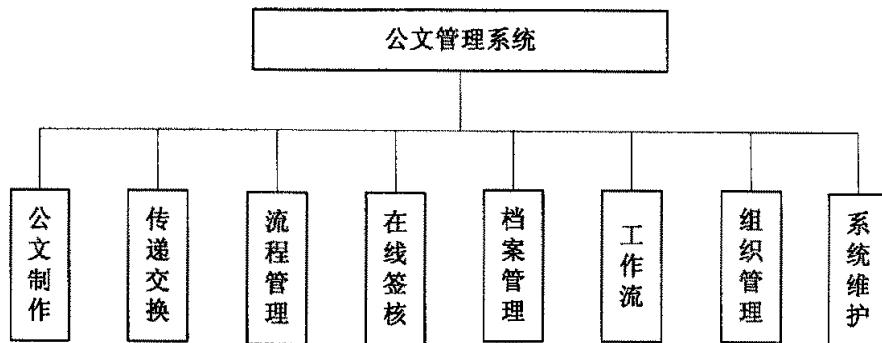


图 5.9 DocMan 公文管理系统

2. 政府机关处理公文的流程

根据政府机关的工作特点和性质, 一般来说, 公文在政府机关之间的流程如图 5.10 所示。

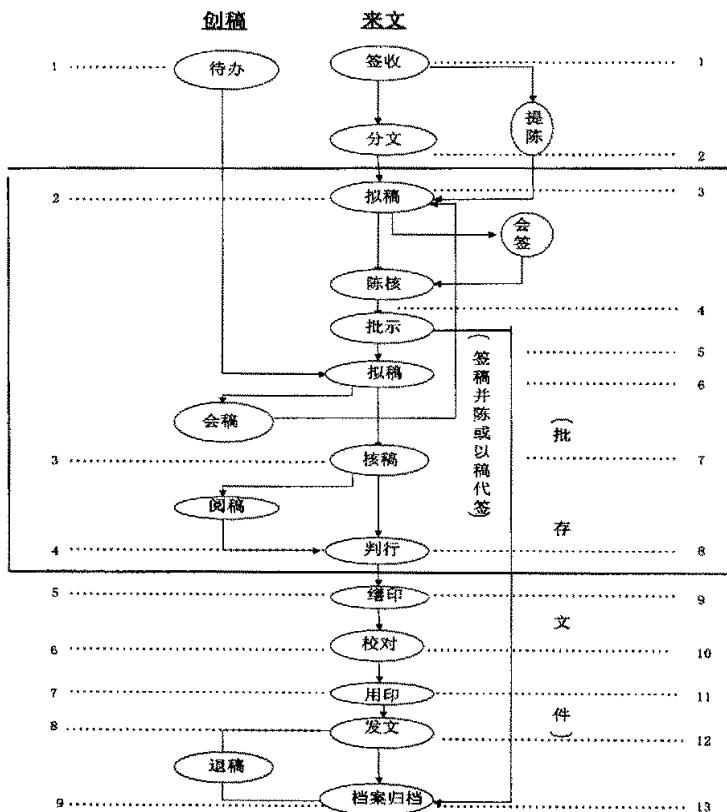


图 5.10 政府机关对于公文的处理流程

由上图可知，从公文产生的方式来看，它可分为发文处理流程和收文处理流程，它们的具体流转过程分别如上图的左右部分。

二、基于 UniFlow 的 DocMan 公文管理系统集成设计与实现

1. DocMan 的集成分析：

(1) 运行环境的要求

第一、尽可能的支持流行的 Web 服务器(如：IBM WebSphere4.0、BEA WebLogic5.0、Tomcat 4.0 等)；

第二、尽可能的支持流行的数据库管理系统(如：Oracle 8i、SQL Server、SyBase、Informix、DB2 等)；

第三、尽可能的支持流行的操作系统(如：Win32 系列、Unix 系列、Linux 系列)；

第四、尽可能的支持流行的浏览器(如：IE 5.0、NetScape)；

第五、尽可能的减轻用户购买其他第三方软硬件的负担。

(2) 非功能性需求及优先级的集成

一般来说，功能需求决定业务架构、非功能需求决定技术架构，变化案例决定架构的范围。其中，非功能需求定义了一些性能、效率上的一些约束、规则。而我们的技术架构要能够满足这些约束和规则。

因此，针对系统的非功能需求，我们对于架构的集成采用：

第一、保证系统的可移植性。在集成时首先要保证整个系统能在上述的环境中运行。

第二、界面友好性。

第三、安全性。

第四、模块化，层次化。

第五、可扩展性。作为一个新的开发产品，它应该是一个可持续发展的。因此，我们所开发的产品必须具有很好的可扩展性。

第六、健壮性。系统的开发过程中，应该始终保证我们的系统是可靠的、健壮的。

第七、效率性。系统的效率性一直是我们开发的系统中较头疼的问题；因此在系统设计、实现过程中，应该时刻考虑系统的效率性。

第八、文档的完备性。在系统开发过程中，应该始终保证，系统具有完整的开发文档，使产品具有更高的可维护性。

（3）DocMan 与 UniFlow 平台之关系

UniFlow 工作流管理平台作为我们电子政务架构中的公文管理应用系统的基础平台，一方面，UniFlow 为 DocMan 的公文流程提供了业务建模和信息建模，而 UniFlow 的过程建模工具则为工作流管理系统定义机关组织模型、工作流程、资源设备等基础模型。另外，DocMan 中的公文流程的转移是通过 UniFlow 平台中的工作流机和许多 API 接口来实现的。另一方面，诸如 DocMan 中的公文编辑等复杂的应用功能是通过开发新的 EJB 组件来实现的，并提供给 UniFlow 基础平台进行调用。

2. DocMan 的集成设计

在进行 DocMan 集成设计时，我们充分地利用 UML 技术，将 DocMan 的架构按照 5 个模型来集成。

（1）用例模型

用例模型是指在 GIP 架构设计中，根据电子政务需求挖掘系统功能所涵盖的主要用例。DocMan 的功能是从下面的用例图 5.11 中获得的。

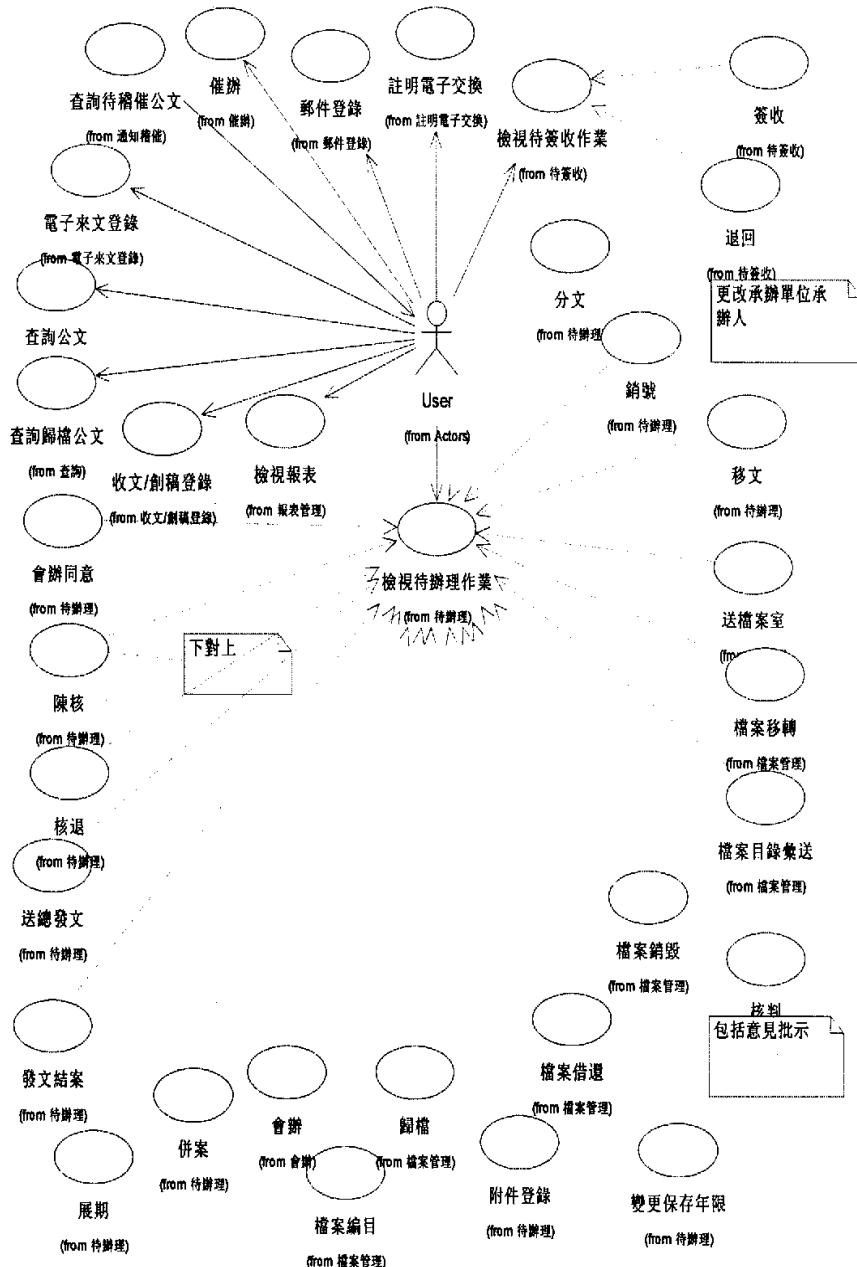


图 5.11 DocMan 的用例图

公文管理模块的角色是建立在电子政务架构的组织管理模块基础之上，各个部门的人员在系统中所担任的角色在组织管理中均已得到设定，各角色的关系如图 5.12 所示。

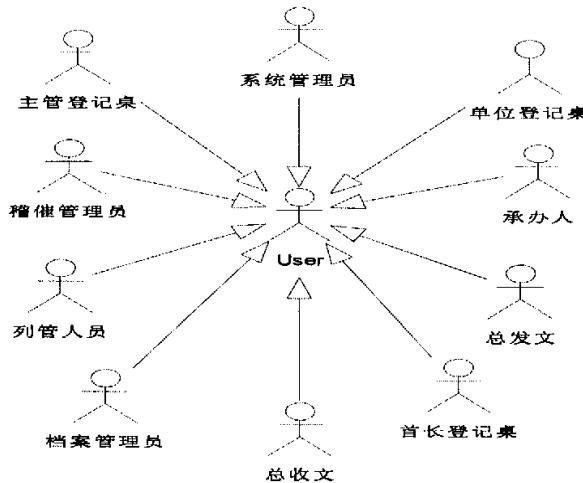


图 5.12 DocMan 的角色

(2) 设计模型

设计模型描述 DocMan 的逻辑结构。设计模型首先设计出系统架构的概要，然后提出系统的关键结构和行为模型元素。

▲DocMan 的架构概要

DocMan 有两个主要的结构，系统的逻辑分解为业务组件及其组件内的层和来自于架构机制的用例实现结构。其中，DocMan 的业务组件集成如图 5.13 所示。

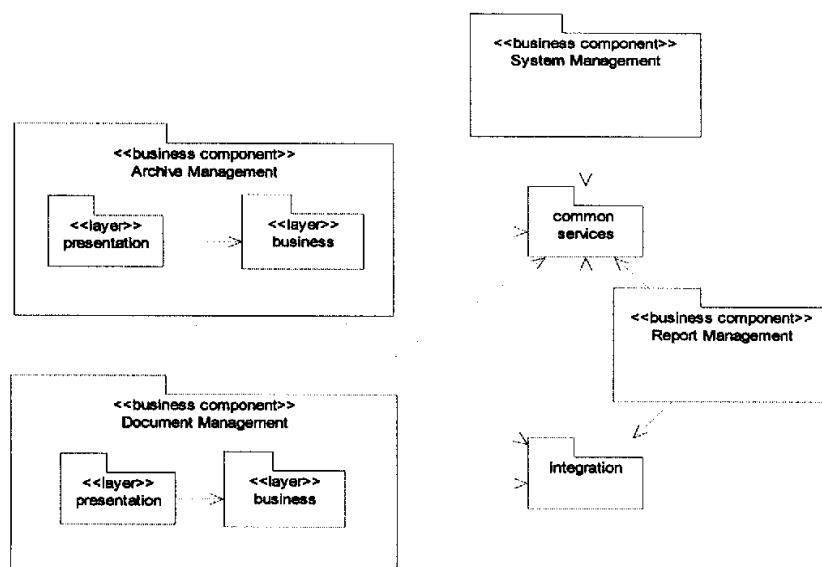


图 5.13 DocMan 的业务组件装配关系

▲DocMan 的集成设计：

所谓的机制是设计用来解决在系统中重复地出现的共同问题。在 DocMan 中所使用的机制就是下面的 MVC 架构模式：Servlet 作为 Control 来处理所有用户的请求和交互以及业务逻辑；DocMan 中的大部分业务逻辑被封装在数据访问对象的内部和集成层的元素中，它们充当 Model。所有的工作流逻辑被设计成 UniFlow 平台和 API，它们同样处在集成层；View 则用 JSP 来实现。

DocMan 的公文管理模块的设计模型如图 5.14 所示。

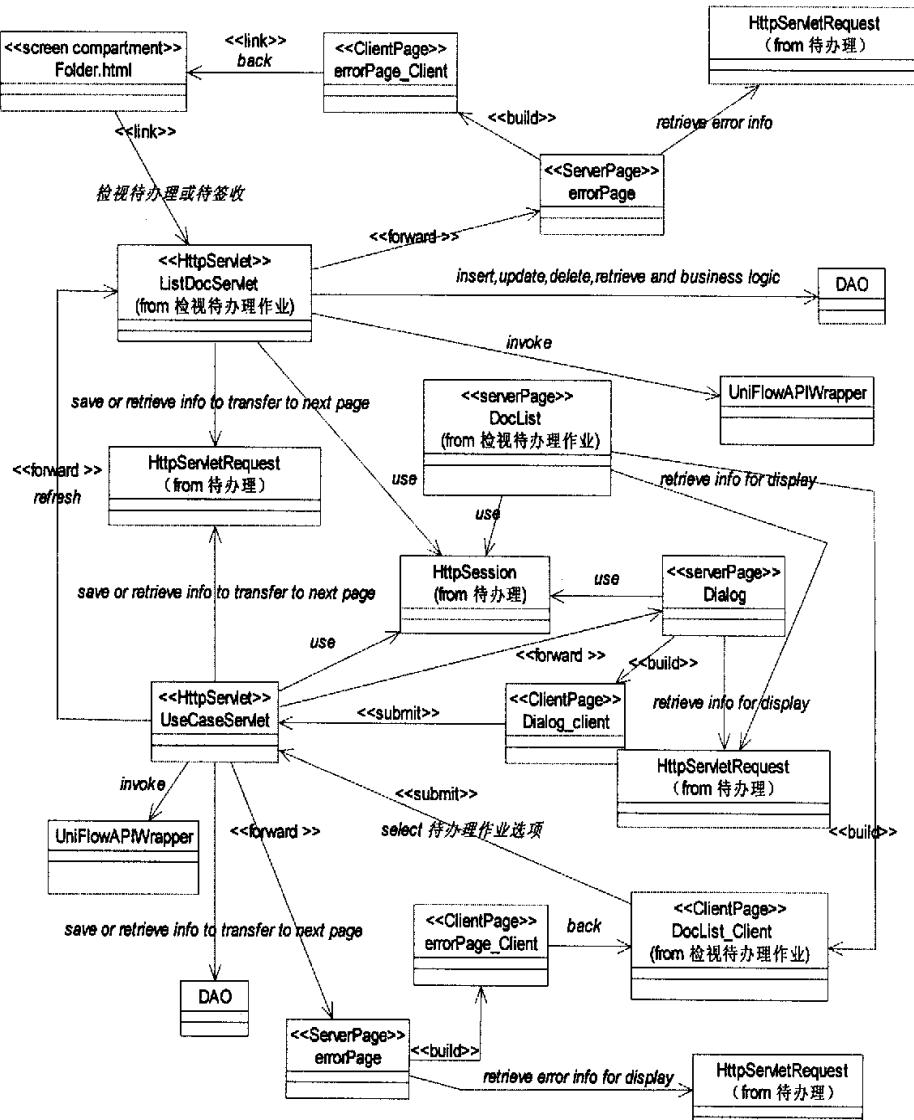


图 5.14 公文管理模块的设计模型

档案管理模块的设计模型如图 5.15 所示。

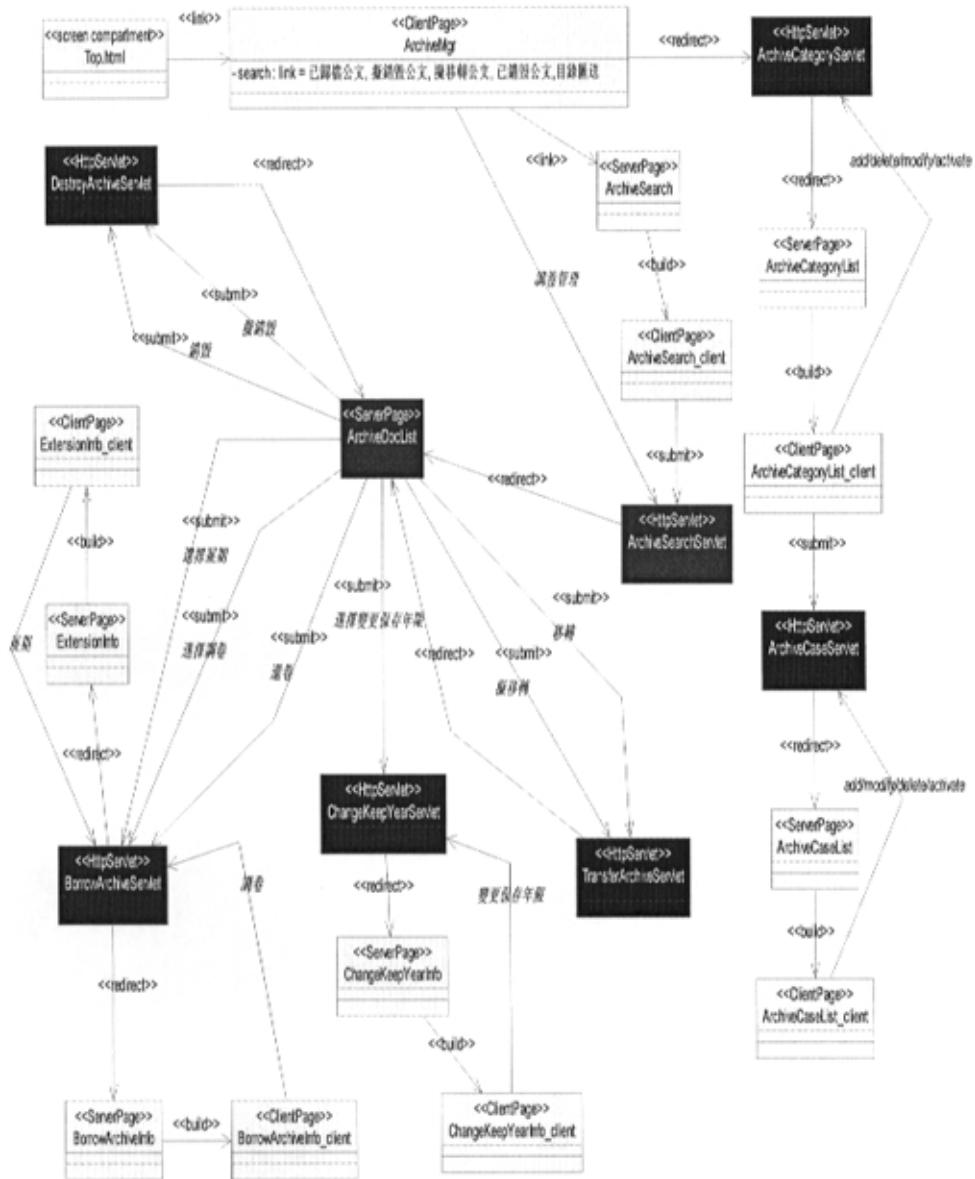


图 5.15 档案管理模块的设计模型

报表管理模块的设计模型如图 5.16 所示。

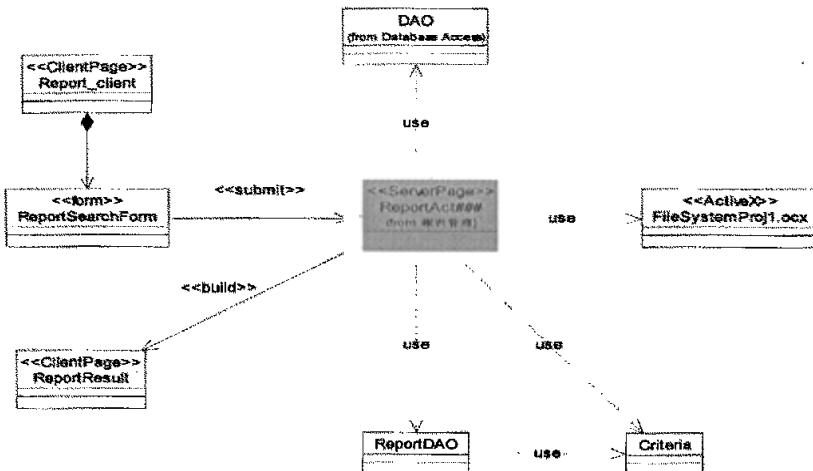


图 5.16 报表管理模块的设计模型

(3) 集成的实现模型

实现模型是指 DocMan 的模型元素的设计，即各层的类图设计。在类图设计的过程中，我们在 DocMan 的架构设计上，对类的设计原则采用下面的方法。

首先，我们对系统的功能进行了充分的分解，分解到细粒度，将每个类设计成尽可能地小。其方法就是对类地命名上采用动宾词组命名法，以此到达类的细化。采用这样的功能分解方法可以到达以下的目的：其一，每个类和方法的精度都提高了，从而达到类的重用性之目的；其二，功能分解还能实现透明的目标，因为我们使用了功能分解的规则之后，每个类都有自己的单独功能，这样，我们对一个类的研究就可以集中在这个类本身，而不用牵涉到过多的类。

其次，我们对于公文流程定义为：公文流程本身只有一个流程，是通过我们在数据库中设定相应数据，生成公文过程模型。供公文应用处理层调用。公文流程中各任务处理程序：公文流程系统根据公文过程模型的定义，针对各个任务（ProcessID）进行相应处理。并把公文应用数据与工作流控制数据之间相互结合较为紧密以此进行集成。

最后，我们在不同类间采用不同的耦合度。尽管人们常常希望类间的耦合度愈低愈好，但是系统间不可能总是低耦合的，否则系统将毫无用处。因此，我们要根据实际客观地对待类间地耦合性。对此我们原则是耦合性地高低和需求的稳定性程度成正比。对于稳定性高的需求，不容易发生变化的需求，我们可以把各类设计成紧耦合的（如 UniFlowWrapper 类），因为这样可以提高效率，而且我们还可以使用一些更好的技术来提高效率或简化代码，例如 Java 中的内部类技术。然而，如果需求极有可能变化，我们就需要充分的考虑类之间的耦合问题，我们可以想出各种各样的办法来降低耦合程度，如：增加抽象的层次来隔离不同的类。这个抽象层次可以是具体的类，也可以是接口，或是一组的类（例如 Beans）。

总之，架构越简单，其稳定性就越好。我们尽可能使所设计的类小一些；尽可能的类的方法短一些；尽可能的让类之间的关系少一些。架构的简单将会带来架构的稳定。

3. DocMan 中的公文传输机制

公文的传输作为 DocMan 中的最基本的功能，就是公文在政府部门之间或部门内部进行公文的处理。然而，在不同的部门可能存在着不同的系统平台，为了达到信息的共享和交互，就必须进行公文的传递交换。实质上，公文的传递交换是实现机关之间及机关内部电子行文、

信息共享、分发公文等接口的功能子系统。显然，公文传递交换子系统的内容主要包含：传递交换格式规范定义、传递交换服务器规范定义、公文接收处理、公文发送处理等内容。公文的传递过程就是电子公文从发送端经过规范定义经过交换服务器最终到达公文的接收端。

在 DocMan 中，我们采用 XML 技术对电子公文进行规范化定义，从而达到传递交换的目的。

（1）电子公文的 XML 格式规范化定义

为了进行公文的传递交换，就必须对公文的格式进行规范化定义。公文的数据元素（Data Elements）是公文的最小的不可再分的信息单位，是一类数据的总称。不言而喻，公文的数据元素标准是电子政务建设中最基本的标准之一，这个标准在公文的信息资源规划中应该被确定为简明的数据元素命名标准、标识标准和一致性标准。所以，我们要将公文的各个属性作为公文的数据元素进行统一定义，使得各个政府部门都要遵从公文的统一格式。

然而，公文的属性有许多种，为了便于在 XML 格式文件中描述，我们将公文的数据元素分为：基本数据元、交换数据元和内部数据元。其中基本数据元不具有特定的含义：如“字”、“日期”等；交换数据元是指与公文电子交换、传输处理有关的数据元素：如“来文日期”、“发文字号”等；内部数据元公文管理或者机关内部使用的数据元。

在使用 XML 技术交换数据时，正确编写的 XML 文档可以有两种形式：它可以是格式正确的或者是有效的。如果一个 XML 文档在编写时遵从了 XML 标准中所列的规则，那么它就被认为是格式正确的。如果一个 XML 文档在编写时还遵循了一个标准的文件数据格式或组织架构，那么它就被认为是有效的。我们在公文的电子交换中采用了标准的 XML 语法，并对应了上述的三种数据元定义了共同传输文件格式，包括：“基本标签”、“交换用标签”及“内部用标签”，进而定义出各种规格公文类型内容结构。

（2）电子公文的交换技术

我们采用了前置处理器（FEP）与公文电子交换中心传递技术实现公文在机关之间的流转。机关公文经公文电子交换中心传递，并由中心提供公文传递通信纪录。

前置处理器与公文电子交换中心传递公文时，其交换信息采用两种方式传送，这两种方式分别是：公文、附件等资料量大的信息采用 SMTP 方式传输；控制信息则使用 FTP 方式传输，利用双向传输方式达到立即确认的目的。公文电子交换中心的 FTP 服务器结合安全控制软件一起工作。当使用者的连接程序没有依据规范执行或资料不符时，将立即被断线。具体如图 5.17 所示。

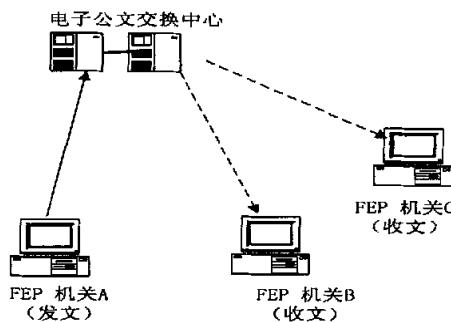


图 5.17 采用 FEP 电子公文交换方式

在上图中，发文机关将公文透过 FEP 传送至各机关统一公文电子交换中心进行发文，而收文机关则通过 FEP 至各机关统一公文电子交换中心进行收文。

§ 5.4.2.3 Ucan 的集成设计与实现

一、Ucan 报表概述

Ucan 以政府现有各种信息系统（包括内部和外部）的数据为基础，通过其强大的数据挖掘技术，以数据智能的形式为管理者服务。Ucan 主要是为了完成对各业务系统的数据查询、统计、挖掘、分析，并最终以报表的形式展现给用户。所以，Ucan 应该包含两个部分：报表设计和查询统计。为了更好地实现报表系统的通用性、可扩展性以及模块化结构设计的需要，我们将 Ucan 报表分为三个模块：报表设计模块、报表产生模块和报表处理模块。这三个模块是根据报表的产生过程而区分的，如图 5.18 所示：

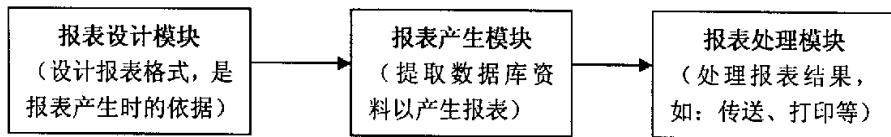


图 5.18 Ucan 报表的三个模块

二、基于 UniFlow 的 Ucan 集成设计与实现

在进行电子政务架构的设计过程中，Ucan 是作为 GIP 管理系统的一个子系统，由该管理系统调用 Ucan 来产生报表。因此需要把两个不同的系统进行紧密的集成。其中，报表的数据来源就是 UniFlow 工作流管理系统中的工作流程所产生的数据表。具体如下：

- 1) 采用执行档后台调用的方式进行，也就是执行 Ucan，并给予参数以控制报表的产生。
- 2) 当然是以组件的方式直接并入到管理系统中，这种组件结构，也比较适用于网际网络上的分布式架构或 N 层式架构。

集成的另一个重点便是报表参数，也就是在产生报表前，临时置入报表格式中的定义，例如日期范围等。要适用于基于 Web 的应用，也就是以网页的方式将报表呈现在屏幕上，方便报表参数的更改。这里采用了 HTML 中交互对象，如 List Box、Combo Box、Radio Box、Edit Box 等。如图 5.20 所示。

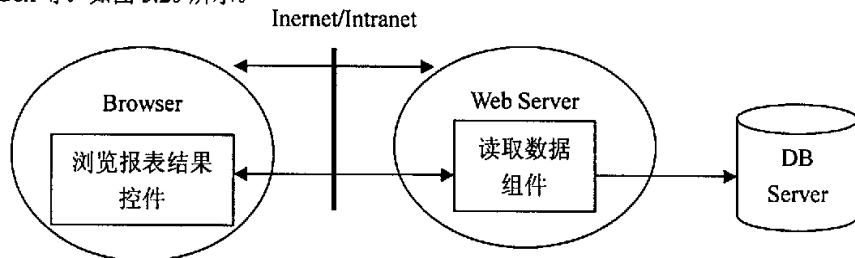


图 5.20 Ucan 系统 WEB 应用之浏览报表结果

由上图我们可以看出，用户根据其角色所赋予的权限进行有关报表作业，他需要在 Client 端输入有关报表的关键字，由 Server 端的读取数据组件按照这些关键字对有关的数据表进行模糊查询，并执行有关的运算，然后将其结果传送给用户。Ucan 对于数据库的处理方式，是透过系统自行组成的 SQL 语句，经由 COM 组件送至数据库系统执行。因此，对报表格式套数据生成报表结果的业务逻辑的实现如下所示：

(1) 组合 SQL 语句

分析所有报表节区中表格部件的单元格之数据来源设置，分组信息以及其他相关表达式

（包括条件表达式），得到所涉及的数据库、数据表、字段的信息。由此可以组合出 SQL 语句。其中的 SQL 语句基本语法为：

Select <数据域>, <计算公式> as <栏名> from <资料表> where <筛选条件> order by <群组字段> [desc]

(2) 套入数据

由组合 SQL 语句得到数据记录集，将数据套入到相应单元格中，并处理有关表达式，得到报表结果。套入数据的层次为：数据集、分组、记录、字段。

(3) 对于存在多个数据库来源的处理

```
SELECT [ALL | DISTINCT] select_list
    [INTO [new_table_name]]
    [FROM {table_name | view_name}[(optimizer_hints)]
    [{, {table_name2 | view_name2}[(optimizer_hints)]}
    [..., {table_name16 | view_name16}[(optimizer_hints)]]]
    [WHERE clause] [GROUP BY clause] [HAVING clause] [ORDER BY clause]
    [COMPUTE clause] [FOR BROWSE]
```

如果目标 Table 或 View 存在于不同的数据库中，则：

table_name | view_name = [[database.]owner.]{table_name. | view_name.}

4) 套入数据时相关部件间的交互

相互部件间的交换如图 5.21 所示。

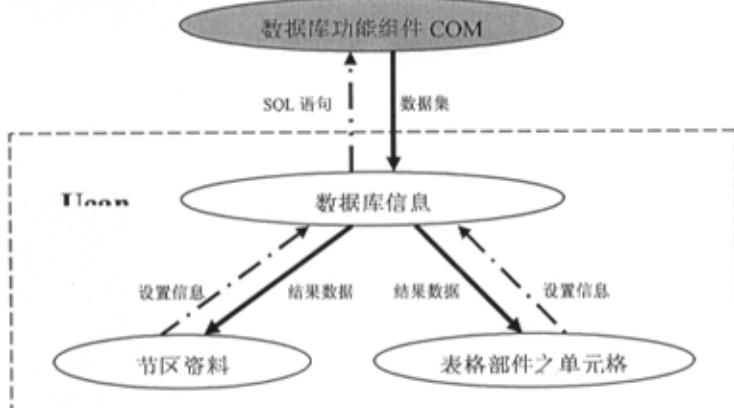


图 5.21 套入数据时相关部件的交互

§ 5.5 小节

本文首先通过对 GIP 的概念和国内外的 GIP 应用进行了较为详细的分析，提出了基于 UniFlow 工作流管理系统平台的 GIP 架构集成方案。通过对各系统的集成设计，实现了将 E-Form 电子表单系统、DocMan 和 Ucan 集成在 UniFlow 工作流管理平台之上的电子政务门户。其中 E-Form 系统作文电子政务应用架构的信息输入方式，为用户提供自定义表单工具，在电子政务应用系统中供用户自定义表单，结合电子政务中的工作流程定义功能，向用户提供自定义工作流，扩展应用系统的功能；DocMan 作为电子政务应用架构的信息处理和信息加工系统，根据工作流的过程定义，对政府业务进行处理电子化处理；Ucan 则是作为电子政务的信息输出方式，它是集成在 UniFlow 工作流管理系统基础之上，对 DocMan 的数据查询、统计、挖掘、分析，并最终以报表的形式展现给用户。

第六章 总结与展望

20世纪90年代以来，伴随着信息技术、特别是网络技术的飞速发展，信息化成为各国普遍关注的一个焦点。在国家信息化体系建设中，政府信息化又成为整个信息化中的关键。为此，各国都在注重发展电子政务应用技术，以提高本国政府的政务工作效率和加强本国政府的服务职能。

近几年来，随着政府信息化建设取得了重要进展，我国对电子政务的应用进行了大力倡导。然而，由于我国的电子政务起步较晚，整体水平仍然较低，各个地区、部门的发展也极不平衡，仍处于电子政务的起步阶段。其中一个较为突出的问题就是缺乏整体规划和统一标准。我国虽然在这方面做出过一些具体规定，但尚没有制定出政府信息化的中长期总体规划，特别是有关统一技术标准方面几乎还处于各自为战的状态。与此相联系，由于我国的地区分割、部门分割现象严重，如何进行电子政务的架构以集成政府现有的各种应用系统和各种信息资源必然成为政府信息化和电子政务发展中的关键所在。本论文就是在这样的背景下进行研究的。

§ 6.1 论文总结

本文通过对国内外的电子政务发展、应用研究概况进行了分析，结合云集（中国）网络有限公司的政府办公软件，采用纯 Java 技术的 B/S 模式并严格遵循 J2EE 标准，设计并实现了以 UniFlow 的工作流管理系统为基础平台、以 GIP 为统一入口并集成了政务应用系统（E-Form 电子表单系统、DocMan 公文管理系统和 Ucan 报表系统）的基于 WEB 的分布式、跨平台电子政务应用架构。

首先，根据电子政务的实际应用，在总体上，从技术理论的角度提出了电子政务集成的原则、目标、架构方案及其实现的过程，并对整个系统的性能优化提出了解决措施。

其次，通过对工作流技术和工作流管理系统的分析，参考了 WfMC 的规范，结合实际需求对电子政务应用系统的基础平台—UniFlow 工作流管理系统进行了功能的拓展和划分，并从状态图及有限状态机理论中引申设计出了适合于电子政务架构基础的基于事件驱动的状态图（State Diagram）模型，以此来对工作流过程模型进行集成设计；并对组织管理模块进行了集成设计，从而实现了 UniFlow 工作流管理系统平台的集成。为了使 UniFlow 系统能够和电子政务的其它应用系统进行有效地集成，我们设计了 UniFlowWrapper 类来屏蔽了 UniFlow 系统的工作流的具体实现细节。电子政务系统的其它功能模块在需要时，通过调用 UniFlowWrapper 类来实现其功能。真正地体现了面向对象的思想。

再次，针对政府的各个应用系统所产生的“信息孤岛”现象，我们从以下方面提出了电子政务系统信息资源整合方案：政府各个应用系统的程序代码跨数据平台；各个应用系统的信息转换；政府异构数据源的集成方案。从而达到电子政务系统的信息资源的集成。

最后，通过对电子政务门户（GIP）的分析，我们设计、实现了 E-Form 电子表单应用系统、公文管理系统和 Ucan 报表系统并将其集成在基于 UniFlow 工作流管理系统为基础平台的电子政务应用架构。其中，E-Form 电子表单系统是一个安装在服务器上，用来建立任意的 WEB 表单并能自动处理表单数据的表单系统，它可随意控制不同用户的权限，实现资料的保密性、安全性，可用于局域网或互联网，非常适合于单位内部或网站填写各种表单资料，建立内容丰富、查询方便的资料库。而公文管理系统采用 Model-View-Control 架构模式：Servlet 作为 Control 用来处理所有用户的请求和交互以及业务逻辑。公文管理系统中的大部分业务逻辑被封装在数据访问对象的内部和集成层的元素中，它们充当 Model；所有的

工作流逻辑被设计成 UniFlow 平台和 API，它们同样处在集成层；View 则用 JSP 来实现。从而实现了公文管理系统和 UniFlow 平台的集成。Ucan 报表系统用来将政府各业务系统的数据查询、统计、挖掘、分析，并最终以报表的形式展现给用户。

§ 6.2 论文不足与展望

本文是建立在 UniFlow 工作流管理系统、Web 化电子表单 E-Form、Ucan 快速报表工具、DocMan 公文管理制作系统、台湾阳明交通大学公文管理系统和台湾行政院主计处公文管理项目、产品和研究课题基础之上，在实际的应用中取得了良好的效果。在经过以后的不断完善，基于 UniFlow 工作流管理平台的电子政务应用架构必将在电子政务产品的市场上产生良好的效应。

但是，由于我国对于电子政务的解决方案上尚未制定出统一的标准，就如何针对政府现有的各种应用系统进行有效地集成还处于探索之中。所以基于 UniFlow 平台的电子政务架构还有待于进一步地研究和改善。具体表现如下：

1. UniFlow 工作流平台存在的不足

首先表现为当工作流实例在运行过程中，可能出现多个活动同时访问共享资源的情况，各个活动在操作上会相互重叠，为了保证系统正常运行，必须进行并发控制。

其次表现在处理大批量的业务时（如每分钟百笔以上），系统出现性能严重下降。

针对这些问题，我们在以后的工作中需要继续研究，使得 UniFlow 工作流管理平台得到不断地完善。

2. 电子政务的安全问题

由于电子政务的应用必然要涉及到国家政府的敏感数据，由此产生的数据安全问题尤为突出。由于本论文主要针对电子政务的架构设计进行讨论，而对于电子政务的安全问题较少涉及。电子政务的安全涉及到：(1)、网络的安全性 (2) WEB 系统访问的安全性；应用服务器访问的安全性 (3) 数据库系统访问的安全性。这些问题是我们今后的工作中所要研究的。

3. 电子政务的其它应用系统没有详细的讨论

一方面，由于我们的研究时间有限；另一方面，由于政府部门的应用系统繁多，实在难以逐一集成。因此，本论文没有对于电子政务的其它应用系统集成进行更多的涉及。针对这个问题，我们在下一步的研究工作中，还要将政府部门一些具有代表性的应用软件加以集成。

4. 对于电子政务应用架构，没有涉及到网络的架构。

我们在进行电子政务的架构设计时，主要针对软件系统体系架构进行设计。严格的电子政务架构必须考虑网络架构的设计，这也是本文今后将要考虑和研究的重点之一。

总而言之，面对我国目前在电子政务建设中所普遍存在的如何将原有的政府应用系统进行集成的问题，针对基于 UniFlow 工作流管理平台的电子政务应用架构的研究，我们在今后的工作中还有待于更深入地研究和开发。

参考文献

1. 厦门云集软件网络（中国）有限公司 ORight 软件及技术资料
2. 台湾华苓科技股份有限公司，AGENTFLOW 软件及技术资料
3. 《快速排版报表制作系统》，台湾捷成科技股份有限公司
4. 《Ucan 系统快速报表制作系统》，云集软件网络（中国）有限公司
5. 《(台湾)文书处理手册》，(台湾)行政院秘书处，2001.6
6. 《(台湾)文书流程管理手册》 (台湾)行政院研考会 2002.12
7. 《台湾文书及档案管理电脑化作业规范》，(台湾)行政院秘书处，2002.4
8. 《捷成公文制作系统》，台湾捷成科技股份有限公司
9. 《云集公文制作系统》，厦门云集软件网络（中国）有限公司
10. Kwan, M. M. and Balasubramanian, P. R., "Dynamic workflow management :a framework for modeling workflows." HICSS97
11. “8848 电子政务解决方案”，http://www.8848.net/solution_government.htm
12. 范玉顺、吴澄，“基于工作流的 CIMS 应用集成支持系统研究”，<http://www.simflow.net>
13. 《电子政务中空间数据的整合之道》 蔡晓兵 2003-7-15
14. 《电子政务系统参考模型》田景熙 (tian@js.cei.gov.cn)
15. 电子数据交换平台——电子政务数据交换需求和 Sybase 解决方案
<http://www.sybase.com.cn>
16. 《UML 与系统分析设计》 张龙详 人民邮电出版社
17. 《推动电子化政府建设》 台湾行政院研究发展委员会
18. 《公文管理系统规格书》 台湾行政院研究发展委员会
19. 《电子化/网络化政府大型行政信息系统整合要求》 台湾行政院研究发展委员会
20. 《公文交換作業》 台湾行政院研究发展委员会
21. 《文书制作规范》 台湾行政院研究发展委员会
22. 《使用 J2EE 整合企业应用——学习 EAI 及用 Java 整合你的应用》

Abraham Kang 著 刘建华 编译

23. 《浅谈各国政府电子化入口网站》 <http://www.find.org.tw>
24. 《E-government Strategy Update - December 2001》 <http://www.e-government.govt.nz>
25. Quick Report 文档资料, <http://www.altecdata.com>
26. Aveysinghe G, Phalp K. Combining Process Modeling Methods[J]. Information and Software Technology, 1997, 39(2):107-124
27. Hartel P ,Denker G .Information Systems Modeling with TROLL Formal Methods at Work [J]. Information Systems,1997,22(2/3): 79-99
28. 施伯乐等编著,《数据库系统教程》, 高等教育出版社, 1999. 12
29. 张维民等编著,《系统建模技术与应用》, 电子工业出版社, 1997. 10
30. 《Java 编程思想》(美) Bruce Eckel 著 候捷 翻译 , 机械工业出版社
31. Sandra E. Eddy, 《XML 可扩展标记语言命令祥解》, 电子工业出版社, 2001.6
32. 陶冶、范玉顺、罗海滨 , “分布式工作流系统的可扩展性和柔性研究”,
<http://www.simflow.net>
33. 蒋慧等编写,《UML 设计可行技术》, 北京电子出版社, 2001. 1
34. Erich Gamma 等著,《Design Pattern》, 机械工业出版社, 2000. 9
35. B. McLaughlin, Java & XML, O' Reilly, 2001.
36. Avraham Leff, James T Rayfield , “Web-Application Development Using the Model/ View/Controller Design Pattern” , Enterprise Distributed Object Computing Conference, 2001. Proceedings. Fifth IEEE International , 2001.
37. KURT A. GABRICK, DAVID B. WEISS, J2EE and XML Development, Manning Publications Co., 2002
38. M. Grand, Patterns in Java, John Wiley & Sons, 1998.
39. Brett McLaughlin, All about JAXP—Sun's Java API for XML parsing,
<http://www-900.ibm.com/developerWorks>
40. Benoit Marchal, XML by Example, Second Edition, QUE publishing company, 2002
41. E. Gamma, R. Helm, R. Johnson, and J. Vlissides, “Design Patterns Elements of Reusable Object-Oriented Software” , Addison-Wesley, 1995.

致 谢

首先感谢我的导师杨志义教授。在我攻读硕士学位期间，杨老师坚实的理论知识、丰富的实践经验、严谨的治学态度和忘我的工作精神无不给予我以巨大的感触和帮助。在课题的研究期间，杨老师无微不至地对我进行指导，期间他对我的课题研究提出了许多珍贵的建议，拓宽了我的视野，丰富了我的思路，加深了我的思考。不仅如此，杨老师还在生活中给我以热心的帮助和极大的关怀，使我学到了许多人生之道。在此我谨向我的导师表示最真挚的谢意！

感谢周兴社教授、赵政文教授、蔡皖东教授、刘明杰教授、谷建华教授、韩云霞副教授、尹信琴副教授、牛振喜副教授、陈世亮博士、商存慧讲师，他们无私的帮助和关怀使我在求学阶段受益匪浅。

感谢云集（中国）网络有限公司的董事长蒋为华先生、经理聂炎明先生、葛志春先生、刘育强先生、吴志松先生、黄涛先生和台湾的 Gene 系统分析员。在我进行课题研究期间，他们都曾给予我许许多多的帮助！尤其是聂炎明先生在选题和分析研究过程中对我进行悉心的指导。

感谢我的同课题的王玥同学和邓万宇同学。在攻读硕士学位和课题研究期间，他们都能给我以无私的帮助和关怀。特别是同他们进行探讨问题时，他们敏锐的观察力和聪明才智使我获得极大的启发。

感谢王备战副教授及其夫人楼润瑜高级工程师。在我攻读硕士学位和课题研究期间，他们都在生活和学习中对我倾注以无微不至的关怀和帮助。他们的帮助，使我终身难以忘怀！

特别要感谢我的父母、兄弟姊妹、我的妻子和儿子，他们正直的人格、勤劳朴素的品质和无微不至的关怀是我学习的动力！他们都给我许许多多的关怀！

我无法一一列举所有帮助我的人，我谨向他们表示真诚的感谢！

西北工业大学 学位论文知识产权声明书

本人完全了解学校有关保护知识产权的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属于西北工业大学。学校有权保留并向国家有关部门或机构递交论文的复印件和电子版。本人允许论文被查阅和借阅。学校可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。同时本人保证，毕业后结合学位论文研究课题再撰写的文章一律注明作者单位为西北工业大学。

保密论文待解密后适用本声明。

学位论文作者签名: 王伟

04年3月1日

指导教师签名: 李晓文

04年3月1日

西北工业大学 学位论文原创性声明

秉承学校严谨的学风和优良的科学道德，本人郑重声明：所呈交的学位论文，是本人在导师的指导下进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容和致谢的地方外，本论文不包含任何其他个人或集体已经公开发表或撰写过的研究成果，不包含本人或他人已申请学位或其它用途使用过的成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。

本人学位论文与资料若有不实，愿意承担一切相关的法律责任。

学位论文作者签名: 王伟

04年3月1日