

基于 Browser/WebServer 模式的电子表单系统

Electronic Bill System Based on Browser/ Web Server Model

徐 鹏 王克宏

(清华大学计算机系 北京100084)

Abstract The application prospects of electronic data interchange technology are approved by all circles. The authors explain the framework and detail implementation technology of electronic bill system based on Web which is implemented in Java, and describe in detail the management and creating system of electronic bill. Finally, the importance and wide foreground of this system are given.

Keywords Electronic commerce, Network computing, Electronic data interchange, Java, Electronic document, Internet, Browser, Web server and PDF specification

一 引言

90年代初,一种新型的计算机网络应用技术——电子数据交换 EDI(Electronic Data Interchange)以其特有的简洁、高效、安全和迅捷特性引起世界各国的高度重视,被认为是提高工作效率、服务质量和企业竞争能力的强有力的手段^[1]。EDI旨在实现表单传送的电子化,所以有人称 EDI 为无纸化贸易。使用电子表单的同时仍然需要纸张表单辅助,只是纸张表单从以前的主要或唯一的地位,下降到次要和辅助的地位。也就是说,EDI最重要的意义不在于节约纸张,而在于其快速、避免重复劳动、提高效率、节约成本等方面,因此 EDI 技术的实质是强调快速传输(比如从邮寄的几天变成几分钟甚至实时)、节约劳动(不必反复打印和录入表单),从而提高效率和节约成本。

随着 Internet 和 Intranet 的高度发展,EDI 必然随之普及。也就是说,EDI 在从传统的客户/服务器(Client/Server)模式,向浏览器/网络服务器(Browser/WebServer)模式转换。后者将取代传统的 EDI 成为主要的 EDI 模式,这是由 Web 的巨大影响力所造成的。基于网络和客户/服务器结构的 E-Form 系统是在企业的局域网或 Internet 网范围内实施电子化表单设计、集中管理、分发、填写、收集处理工作的系统。它包括了电子表单的设计工具、客户端应用程序、管理工具以及在网络上的分发工具、数据上报工具、电子表单与数据库的操作定义、打印输出管理系统、电子表单对业务流程(表单处理流程)的定义。

标准化 EDI 已经成为全球电子商务的关键技术,实现了世界范围内电子商务文件的传递。先进的 EDI 技术具有开放性和包容性,在开发 EDI 网络应用程序

中,无需改变现行标准,而只需要扩充标准。随着 Internet、Intranet 和电子商务技术的发展,当这个商业社会环境都实现了表单的电子化,在一张表单中就可以定义全球范围内的审批和支付过程。

二 电子表单生成及管理系统 XINCA 的研究与实现

在电子商务飞速发展的时代,电子表单系统成为了整个 EDI 系统中必不可少的一个组成部分。随着 Internet 的发展,不同的公司可以将公司承接项目的数据存储在公司数据库中,由于公司的职员分布在不同的办公室,甚至在出差时可能远在异国他乡,因此希望能够实现通过网络对这些数据进行可视化的实时访问,更新和维护。另外,由于这些项目信息经常需要显示或打印成表单的形式使用,而数据库的显示及打印效果并不尽如人意,所以要求把数据处理成某种电子表单的形式,以便公司职员能够通过网络调用这些表单。随着在线交易系统逐渐增多,不论是商家之间进行正常的贸易,还是普通客户与网络商店进行的在线购物,都需要有一个进行交易的凭证。电子表单在其中也发挥了其他方式所不可取代的作用。

电子表单系统包括三个部分:硬件系统、翻译软件和传输系统^[2]。其中后两者对于众多软件开发人员来说,意义更加重大。特定的处理程序需要将用户接口中提供的特定内容根据客户的需要,翻译成所需的电子文档格式,供客户浏览;同时通过网络,将翻译好的电子文档传送到客户端。电子表单管理制作系统正是在这样的背景下产生的。它应用于包括 Internet 和 Intranet 在内的网络环境中,其功能是由用户通过网络对存储于数据库中需要在表单上体现的数据进行管

理,对生成表单的外观进行设计,并最终生成满足用户要求的电子表单。

Java 语言是伴随 Internet 而形成、发展、壮大的。这种语言自身具有的诸多优势和功能,使其成为网络计算环境下应用程序设计的首选语言^[1]。同时 Java 软件公司与数据库和数据库工具的开发商一起建立了一种独立于 DBMS 的机制,使得开发者不必考虑所用的特定数据库而编写客户端应用程序。JDBC 是一种支持基本 SQL 功能的通用低层应用程序接口,它在不同的数据库功能模块的层次上提供了统一的用户接口。Java 技术与电子表单系统的结合可以说是相得益彰!

2.1 浏览器/Web 服务器结构模型

为了规范化电子商务的过程和信息服务形式,人们常常在企业商务站点中设置许多表单。通过表单在网络上的相互传递来达到网络商务单证交换的目的,在 Web 站点设计过程中,表单是通过屏幕设计的结构和内容来完成的。随着计算机硬件性能的提高和价格的大幅度下降,图形化界面和鼠标点击的操作方式大大缩短了人机的距离,而现代计算机网络拉近了人与人之间的距离。因此,可以将以前由客户通过业务员输入或传达的信息,直接由有能力的客户输入和浏览。如顾客通过网络自行提交订货单、定票单,查看订货、订票情况,以减少中间环节、降低差错、提高效率、节省人力。它实际上是将业务信息系统延伸到客户端,使客户也能够加入到业务 workflow 操作中来。

EDI 系统传统的传输过程可以划分为几步:

- 1) 数据被内部应用程序从内部系统提取出来;
- 2) 数据被转换为某种数据交换双方或多方都认同的标准格式;
- 3) 数据被传送到对方,这一步分为数据从本地机传出、在线路上传递以及被对方接收三个部分;数据交换的双方可能直接连接,也可能通过电话线和调制解调器,或 ISDN 网和 Internet 等方式连接;
- 4) 数据被转换为对方内部使用的格式;
- 5) 数据被对方的内部应用程序自动处理。

传输过程中,计算机的种类、应用程序的类型和连接方式都可以是任意的,对正常的传输没有影响。而且由于在传输过程中无需再输入,使出错率有可能降为零,因此大大节省了传输时间和经费。

目前生成的表单是通过电子邮件以报文的形式来传送的,电子邮件方式的表单传送也不能够满足信息时代用户的需求,其主要缺陷有两点:一是必须获得客户端的电子邮件地址方能实现传送;二是普通的电子邮件为纯文本,使用的信息方式少,信息量小,且安全性差,因此要选择一种特定的文件格式,并在生成系统中按照这种格式来生成最终的电子表单文件供用户浏

览。由于网上交易的时效性非常强,要求实时生成表单并将其传送给客户,因此必须通过系统软件方式自动生成结果文件并将表单中需呈现的数据添加到文件中去。

由于 Internet 的发展和电子表单管理制作系统在网络上应用,因此我们所设计的电子表单系统应当充分利用 Web,并应当弥补传统表单生成系统的不足。我们利用 Java 语言编写 Applet 程序,并将其嵌入网页中。当用户访问此网页时,程序类文件随着主页下载到用户本地机上,并被浏览器解释执行。但是我们发现如果单独使用 Applet 程序会出现一些缺陷:

- 实现表单文件自动生成功能的软件包不属于 Java 语言提供的基本类库,因此浏览器访问主页时,如果想正确解释执行程序,需要先从 Web 服务器下载此软件包,然后才能够使用它的方法。由于表单生成软件包的功能复杂,包含了较多的类文件,即使打包成 JAR 文件以后,文件还是比较大,所以下载此软件包会影响程序初始执行的速度;

- 程序将电子表单做成特定文档格式的文件,该文件需要保存起来以便由浏览器的辅助应用程序打开它供用户观看。然而 Applet 受安全性的限制,不能够对本地的文件系统进行读写操作,所以对程序来说是致命的缺陷;

- 在 Applet 中使用 JDBC 对数据库进行操作是可行的,但是它不能够装入和运行包含本地方法的驱动程序,对于通过 socket 直接与数据库通信的驱动,只能要求数据库在产生 Applet 的服务器上。有时还会涉及防火墙的问题,而且数据库管理员也不易于对该程序访问数据库时出现的错误即时进行监控或改正;

- Applet 任务过多,负担重,其程序维护困难。

基于以上原因,我们考虑使用浏览器/Web 服务器结构。

客户端借助于 Web 浏览器(例如 Netscape 和 Microsoft IE),其程序是一个 Applet,它被嵌入网页之中,由用户使用浏览器访问主页时可以下载到本地机运行。客户端程序一方面负责与用户进行交互,生成图形化用户界面,接收用户的输入,用户可以进行的操作包括管理数据库数据,设置电子表单外观的属性,以及查阅电子表单(最好是能够直接在 Web 浏览器中对生成的电子表单文件进行浏览),另一方面,客户端程序与服务器端程序进行网络通讯,在两端之间传输数据。它向服务器端提出请求,并传送相应的参数,然后它从服务器端接收相应的结果。

服务器端程序是一个一般的应用程序(Application),它由管理员在服务器上使用 JDK 来运行。一方面服务器端程序负责在 Web 服务器上最终完成各种

操作,包括对数据库进行读写操作,完成管理数据库中数据的功能;以及对文件系统进行访问,并利用 XIN-CA PDF 软件包生成最终的电子表单文件。另一方面,服务器端程序监听客户端请求,接收客户端的参数,完成要求的操作,然后把结果返回客户端。浏览器/Web 服务器结构解决了单独利用 Applet 程序所产生的问题,运行 Applet 不必下载 PDF 软件包,程序运行效率高;JDBC 由可信任的应用程序运行,使数据库系统可以分布在远程服务器上;制作的 PDF 文件被保存在服务器上,由浏览器通过网络访问;服务器端程序分担了实际电子表单管理和制作的任务,Applet 程序的负担减轻了。

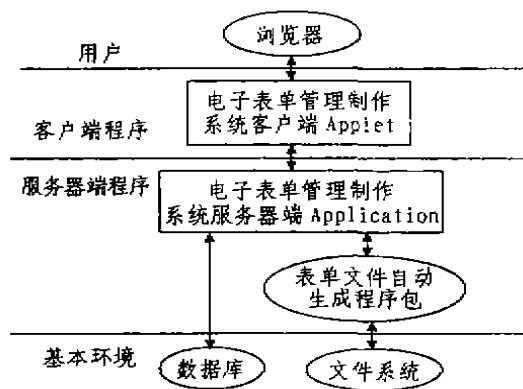


图1 基于浏览器/Web 服务器模式的电子表单系统

图1展示了电子表单管理制作系统程序的基本结构。

当然,浏览器/Web 服务器结构也有自身的一些缺点。由于客户端需要与服务器端进行数据交换(即客户端将用户请求和输入的参数传送给服务器端,服务器端将处理之后的结果数据传回客户端),所以相对于单独 Applet 的实现方式来说,多了网络数据传输这一步骤,减慢了执行速度,也加大了编程工作的复杂度。但是这些缺点一般不足以对程序造成大的损失,尤其对于企业内部局域网来说,一般网络传输的速度很快,系统执行速度不会受到较大的影响。

2.2 客户端程序模块的划分与实现

电子表单自动管理和生成系统的客户端是 Applet,它具有三种模块类型:图形界面接口模块、网络通讯模块和客户端管理模块。

1. 图形界面接口模块 此模块面向用户,功能是负责与用户进行交互,生成图形化用户界面,接收用户输入。我们可以通过 Java 语言提供的丰富的组件类型和完善的布局管理器对用户界面进行设计。在事件处理程序中可以使用多线程技术,用 AWT 线程响应事

件并获取参数,用另外一个专门的线程来完成具体事件的处理过程。这样 AWT 线程可以很快返回闲置状态,以备下一个事件使用,在用户看来系统可以实现对事件的实时响应了^[4]。此外在图形界面接口模块中还包括数据库管理模块和电子表单文件设置模块,其中数据库管理模块负责为用户查询、删除、新建或修改数据库记录并进行可视化显示;电子表单文件设置模块负责设置决定表单文件外观和内容的属性,例如文件的页面大小、字体信息和文本内容等属性。这样,用户就可以根据自己的需要选择获取的信息内容并设置最终电子表单文件的基本格式。

2. 网络通讯模块 由于此系统基于浏览器/Web 服务器结构,因此作为客户端的 Applet 必须通过这一模块与服务器端进行数据交换。通过 socket 通讯技术的使用,把从图形化用户界面模块传递的用户输入的数据传送给服务器,再接收服务器端的结果数据,传递给图形化界面接口模块并显示给浏览者^[4]。对于 Applet 程序来说,目前它只能与其所在服务器上的应用程序进行通讯连接,而不能与其他主机连接,这种限制对于浏览器主机来说是安全的。

由于在程序设计过程中已经设置好了数据传输的顺序,对需要传输的数据类型也很清楚,因此可以利用 Java 语言提供的多种输入/输出流方式(例如 ObjectInputStream 和 ObjectOutputStream),按照数据本来的类型进行传输,而不需要将其转换为字符串再送往输出流,然后再由服务器端从输入流得到数据后再转换为原类型。这样可以提高整个系统程序的执行效率。

3. 客户端管理模块 此模块负责对整个客户端程序的功能进行协调控制。

2.3 服务器端程序模块的划分与实现

电子表单自动管理和生成系统的服务器端是普通的应用程序,它按功能划分为四个功能模块:

1. 网络 I/O 操作模块 实现与客户端进行输入输出流的数据交换功能。此系统应用于 Internet,必须能够同时为多用户提供服务,因此在网络 I/O 操作模块中我们使用了多线程技术,每一个独立的线程均实现了 socket 通讯、输入/输出流数据传输和服务器服务功能。

2. 数据库操作模块 本模块根据网络 I/O 操作模块从客户端接收的数据,产生相应的 SQL 命令,实现对数据库进行读写的操作,包括查询、删除、插入和更新记录数据。我们使用了 JDBC 来访问数据库,通过它可以充分利用 Java 语言的特点并实现不同数据库的统一访问接口,可移植性强。同时使用 Application 作为客户端 Applet 访问数据库的中间层,这样可以提高访问效率。

3. 电子表单文件制作模块 传统的表单均是基于纯文本的,而这种方式在信息化社会中已经远远不能够满足用户的需求。目前最重要的问题是要选择一种最佳的文件格式供所生成的电子表单使用,并将用户在电子表单中所需要呈现的数据按照特定格式文件的规范写入并生成最终的电子表单。

通过对一个电子表单的基本功能进行分析,可以进一步确定在电子表单自动生成系统中对功能模块的设计方向。构成电子表单的基本要素包括

- 待填写的“表单字段”。这是表单中最重要的部分,这是通过填写表单字段,表单才能够完成对业务处理的定义,表单字段应当规定填写内容的形式,如文字的字体、大小、图片的大小、色彩等。

- 线框和表框,表单的线框在物理位置、虚实和粗细的控制上有严格的要求。表单的表框在复杂的表头的设计和表体、表头合一上有较严格的要求。

- 固定的文字说明。表单中的陈述信息需要使用大量的文字说明,而一个 PDF 的页面空间有限。

- 固定的图片、徽标:表单中的题头、题尾、装饰等都会用到徽标、装饰线等贴图,这些贴图的大小、比例都可以随意调整。

- 选择框:这是表单中“表单字段”的另外一种形式,选择框往往被编成各种不同的组。根据选择性的不同,成组的选择框可以是多选一(单选框),也可以是多选多(复选框)。

将目前国际流行的几种文档格式比较之后,我们认为 PDF 是电子表单首选的文档格式。PDF (Portable Document Format),即可移植的文档格式。用户可以以一种简便、可靠的方式来浏览 PDF 格式的电子文档,同时这种文档格式还独立于生成它的应用软件、硬件和操作系统,因此这种文档具有可移植性^[5]。按照 PDF 格式生成的电子表单具有如下几个优点:

- PDF 文件是与平台无关的,所以它适合于网络环境的需要,对相同的表单内容,用户无论是在微机还是在工作站上,无论使用 Windows、Unix 还是 Macintosh 操作系统,都希望看到同样的效果,而 PDF 文件可以满足这个要求;

- PDF 文件结构紧凑,并使用了数据压缩技术,因此生成的文件较小。这样在网络环境下使用时,传输量少,传输速度快,这一点对于当前网络条件并不十分理想的中国来说优势体现得更加明显;

- PDF 文件功能强大,显示效果精美。缩略图、概要图的使用更有利于用户对文件整体有全面的了解;超链接的使用便于用户对感兴趣的内容进行浏览;对文本、图形、图像,甚至音频和视频的支持,使得 PDF 文件具有多媒体化的优点,表单的全部内容都可以用

合适的 PDF 功能来实现;

- PDF 文件随机访问的特点满足用户快速浏览文件的要求。

另外,与其他几种常见的文档格式相比,PDF 格式的文档也具有其他格式文档无法取代的优势,下面我们将 PDF 与其他几种主要的文档格式进行比较:

- TXT 文件:具有跨平台的特点,但其内容仅仅局限于文字,在网络时代这种文档远远不能够满足用户的要求。

- DOC 文件:打印效果很好,但与平台相关。由于是 Microsoft 公司的产品,因此只能够在 Windows 操作系统下使用,而不能在 Unix 等其他操作系统上应用。

- HTML 文件:与平台无关,但显示和打印效果不好,文本、图像无法做到精确定位,而且在文件中无法进行绘图。

- PostScript 文件:与 PDF 文件很相似,但性能不及 PDF 文件,而且这是一种编程语言,不容易掌握。

- PDF 文件:与平台无关,适合网络使用。文件结构紧凑,并使用了数据压缩技术,在网上使用时,传输速度快。此外这种文件的功能强大,显示效果精美。缩略图、概要图的使用有利于用户对文件整体进行了解;超链接的使用便于用户对感兴趣的内容进行浏览;对文本、图形、图像,甚至音频和视频的支持,使得 PDF 文件具有多媒体化的优点,电子表单的全部内容都可以用合适的 PDF 功能来实现。

但是 PDF 自身也存在一定的问题,例如用户无法自动生成一个 PDF 文件,无法将 PDF 格式的商务报表与数据块结合在一起。正是为了弥补上述这些不足之处,我们设计 XINCA 电子表单自动生成系统。XINCA 系统由 XINCA PDF 软件包和 XINCA Studio 可视化开发工具组成, XINCA PDF 软件包适用于任何能够运行 Java 虚拟机的计算机,在这样的计算机上您需要安装一个支持 Java 的网络浏览器(例如 Netscape Navigator 或 Microsoft Internet Explorer)以及一个 PDF 浏览器(Adobe 公司免费为全球用户提供了 Acrobat Reader)。这个软件包的主要用途是根据需要自动生成各种基于 PDF 格式的电子报表和其他电子文档,在所生成的 PDF 文件中可以包容文字、图形、图像、声音等多种媒体信息,并可以实现链接、压缩、加密等处理。在 PDF 软件包的详细设计中,我们针对文字、图形等不同对象的属性设置缺省的属性取值,这样如果被添加的对象其属性与缺省值相同的话,就可以不用调用相应设置属性的方法了,从而简化了程序设计人员的编程工作。同时利用 Java 语言自身的多态特性,使得方法的使用更加灵活,因为我们所设计的软件

包是供 Java 开发人员使用的,因此在设计上必须考虑与标准类库中的组件设计方法保持一致。针对以上所提到的各种组件,我们应当提供与标准接口一致的编程接口,包括类、成员变量以及方法等。这样开发人员使用起来就比较熟悉并且易于操作。不同的企业可以根据自己的需要,利用 PDF 软件包设计出不同种类的电子表单填写程序,服务器端的程序将根据用户填写的内容和企业表单的模式,自动调用 PDF 软件包实时生成 PDF 电子表单文件并通过网络传输给客户,从而有助于实现正规化、安全的实时在线交易系统和在线数据查询系统。

PDF 软件包是以代码方式呈现给用户的,所以表单设计人员必须通过手工编写不同的代码来完成电子表单设计工作,同时对我们前面所介绍的 PDF 软件包的结构(包括各种类方法的调用以及接口的定义等)应当十分熟悉,另外对电子表单设计的修改也是比较麻烦的。因此,为了满足各种电子表单生成的需要,以及提高系统的易用性和直观性,我们需要使实现电子表单设计功能的系统具有可视化功能。通过 XINCA Studio 可视化开发工具,用户可以通过简单的操作来完成电子表单的设计,并且用户可以做到所见即所得(WYSIWYG),从而可以很大程度上提高设计效率,并减少设计人员的培训时间。可视化表单生成工具的推出,能够为用户提供更为完善的电子商务解决方案,帮助用户高效率地使用简单操作即可完成电子表单的设计功能。

在电子表单文件制作模块的设计中需要特别注意的是,由于服务器文件系统空间有限,因此程序不能无限地创建新文件。当文件创建到一定数目时,就需要覆盖以前的文件,文件数目的上界值由程序管理员按照访问请求数量来确定。服务器程序在一般情况下并不为每个客户保留他们创建的表单文件。每次表单文件生成后,文件存储地址及名称被返回客户端,浏览器直接开始下载结果文件。服务器本地会对此文件进行备份,保证用户观看。

4. 服务器管理模块 负责对整个服务器端的功能进行协调控制。在系统实际运行过程中,本模块另一个重要任务是输出错误提示信息。因为服务器端程序在后台运行,不影响任何其他程序,也不占用桌面窗口,因此服务器端程序通常没有自己的界面,也无法把运行中产生的信息放入窗口构件来显示。一般情况下信息输出方式有两种:通过标准输出系统进行输出和通过文件进行输出。标准输出系统的输出很方便,且不占用文件系统空间,但是可显示的信息量一般较小,而且不能够长期保存以备查看;而将信息写入特定的文件则可以弥补前一种方式的缺陷,程序维护人员可以

根据文件中的详细信息很方便地对系统进行管理,但是这样就需要占用一定的磁盘空间。综合考虑两种方式的利弊,我们可以同时在系统中分别利用两种输出方式。将用户请求登录等一般信息保存到文件中以便管理员进行系统维护;对于数据库查询错误、电子表单文件生成错误等信息,一方面通过标准输出系统显示警告提示,一方面向文件写入具体的错误信息,使管理员可以及时、正确地对错误进行处理。

三 应用前景

传统预印纸张表单的生成、处理、流转方式是企业业务处理中错误、缓慢、昂贵的重要原因。目前,全世界占公司文件 80% 的预印表单,每年将耗资 60 亿美元的印刷费和 1200 亿美元的存储、分发、处理费用。激烈的市场竞争使业务种类的改进和扩展速度加快,对预印表单输出提出了挑战。因此电子表单自动生成系统被认为是替代纸质表单的一项重大技术。电子表单技术风靡西方发达国家,在企业业务中发挥了重要的作用。利用这种模型设计出的电子表单系统不但可以实现电子交易中定单等各种表单的填写、生成和传送过程,而且可以应用于各种 Web 站点(例如,目前此系统已经应用于日本多家电视台的 Web 站点,当用户希望查询特定时间段的电视节目时可以通过网页输入查询日期,这样服务器可以根据查询条件搜索数据库,并将查询结果直接生成一个 PDF 文件传送给用户)。目前,包括日本三菱公司、日本 NTT、SONY 公司在内多家著名的跨国企业都已经应用 XINCA 系统实现了企业内部及企业与用户之间的电子表单自动生成和传送功能。在上海证券交易所上海公司定期报告定期发布系统中,利用 XINCA 系统实现上市公司网上披露文件的自动生成功能。同时 IBM 公司的 WebSphere 大型电子商务项目和 Oracle 公司的软件产品中也应用此系统实现了电子票据的自动生成和管理功能。

参考文献

- 1 吴国新,吉逸. EDI 技术研究及其实现. 计算机研究与发展, 1998(2)
- 2 Margherio L. The Emerging Digital Economy. 1998
- 3 Java Programming Guide, JavaSoft Corp., Sun Microsystems
- 4 王克宏. Java 语言编程技术. 清华大学出版社, 1997. 9
- 5 Bienz T, Cohn R, Meehan J R. Portable Document Format Reference Manual Version 1.2. Adobe Systems Incorporated, 1996. 11
- 6 Xu Peng, Wang Kehong. The Design of Multi-Language PDF Package. International Symposium on Future Software Technology, 1998. 10