

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 17/30 (2006.01)

G06F 9/44 (2006.01)



[12] 发明专利说明书

专利号 ZL 200510125312.6

[45] 授权公告日 2008 年 11 月 5 日

[11] 授权公告号 CN 100430940C

[22] 申请日 2005.11.15

[21] 申请号 200510125312.6

[30] 优先权

[32] 2004.11.18 [33] US [31] 10/992,523

[73] 专利权人 国际商业机器公司

地址 美国纽约

[72] 发明人 詹姆斯·M·麦卡德勒

[56] 参考文献

CN1409841A 2003.4.9

US2002/0174174A1 2002.11.21

审查员 赵颖

[74] 专利代理机构 中国国际贸易促进委员会专利
商标事务所

代理人 付建军

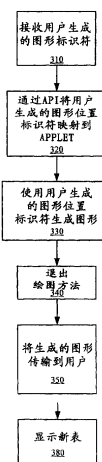
权利要求书 2 页 说明书 6 页 附图 3 页

[54] 发明名称

编程门户应用程序的方法和系统

[57] 摘要

提供了用于实现从客户端到服务器的动态图形接口的操作的系统和方法。该方法包括在门户接收多个用户生成的绘画位置标识符，将用户生成的绘画位置标识符通过 Applet 应用程序编程接口 (API) 映射到 Applet 功能，利用所述用户生成的绘画位置标识符使用绘图方法生成绘画。在生成绘画之后，绘画被返回到与用户生成的绘画位置标识符关联的用户。用户生成的绘画位置标识符可以是鼠标点击、敲击以及通过笔进行的平板 PC 图形指示中的一个或多个。门户通过 Applet API 启用图形接口，Applet API 被存储在 Java 存档文件 (JAR) 中。JAR 文件能使多个用户查看绘画，而无需本地文件。



1. 一种实现从客户端到服务器的动态图形接口的方法，该方法包括：

在 Portlet 接收多个用户生成的绘画位置标识符；

将用户生成的绘画位置标识符通过 Applet 应用程序编程接口 API 映射到 Applet 功能；以及

通过 Applet 功能利用所述用户生成的绘画位置标识符使用绘图方法生成绘画。

2. 根据权利要求 1 所述的方法，进一步包括将绘画返回到与用户生成的绘画位置标识符关联的用户。

3. 根据权利要求 1 所述的方法，其中用户生成的绘画位置标识符是鼠标点击、敲击以及通过笔进行的平板 PC 图形指示中的一个或多个。

4. 根据权利要求 1 所述的方法，其中，所述 Portlet 通过 Applet API 启用图形接口。

5. 根据权利要求 1 所述的方法，其中，Applet 是使用 Java 编程语言命令来编写的。

6. 根据权利要求 1 所述的方法，其中，Applet API 被存储在 Java 存档文件 JAR 中。

7. 根据权利要求 1 所述的方法，其中，映射包括将 Applet 方法映射到 Portlet 方法。

8. 根据权利要求 6 所述的方法，其中，Applet API 允许多个用户查看绘画，而无需本地文件。

9. 一种计算机系统，包括：

处理器；

连接到处理器的存储器；

通过处理器可操作的 Portlet，所述 Portlet 被配置为接收多个用户生成的绘画位置标识符；

连接到所述 Portlet 的 Applet 应用程序编程接口 API, Applet API 被配置为通过所述 Portlet 映射用户生成的绘画位置标识符; 以及

绘图功能, 该绘图功能被配置为通过 Applet 利用所述用户生成的绘画位置标识符使用绘图方法生成绘画。

10. 根据权利要求 9 所述的计算机系统, 其中, 所述 Portlet 通过 Applet API 启用图形接口。

11. 根据权利要求 9 所述的计算机系统, 其中, Applet API 被存储在 Java 存档文件 JAR 中。

12. 根据权利要求 9 所述的计算机系统, 其中用户生成的绘画位置标识符是鼠标点击、敲击以及通过笔进行的平板 PC 图形指示中的一个或多个。

13. 根据权利要求 9 所述的计算机系统, 其中, 绘画是通过画图方法来进行的。

14. 根据权利要求 9 所述的计算机系统, 其中, Applet API 被配置为将 Applet 方法映射到 Portlet 方法。

编程门户应用程序的方法和系统

技术领域

一般来说, 本发明涉及编程代码, 具体来说, 涉及使用 Java 小程序应用程序编程接口 (API) 的编程门户应用程序的系统和方法。

背景技术

自从 Java 编程在 1995 年开始, Java 小程序在计算机编程行业是已知的。Java 小程序通常从网站部署, 以独立于用户的本地文件与用户的浏览器应用程序进行操作。Java 小程序仍在继续流行, 因为程序员可以只用编码所需的五个主要方法来相对轻松地编写代码。Java 小程序已经被诸如 servlet、Java Server Page (JSP) 和 portlet 之类的更加健壮和复杂编程序模块取代。servlet 是对服务器的扩展, 其增强了服务器的功能。servlet 的最常见的用途是通过提供动态 Web 内容来扩展 Web 服务器。它们是在 Web 服务器上运行并建立 Web 页面的程序。servlet 可以动态地建立 Web 页面。快速建立 Web 页面是有用的, 因为 Web 页面可以基于由用户提交的数据。此外, 数据会频繁地变化, 页面可以动态地建立页面, 以便可以查看当前信息。Java Server Page (JSP) 是允许您将常规静态 HTML 与动态生成的 HTML 混合的技术。由 CGI 程序建立的许多 Web 页面大部分是静态的, 动态部分限于几个小的位置。但大多数 CGI 变体, 包括 servlet, 使您通过程序来生成整个页面, 即使其大多数总是相同的。JSP 允许单独地创建两个部分。

Portlet 是向门户用户显示相关信息的可重复使用的 Web 组件。Portlet 可以被用来实现电子邮件、天气、讨论论坛和新闻。Portlet 标准可使开发人员创建可以被用来在任何门户中支持标准的 portlet。

portlet、servlet、等等所存在的问题是，存在在服务器上难以实现的功能，如使用诸如 drawRect 等等之类的标准图形方法动态地创建图形，此后，将图形编码为 .jpg 文件等等的能力。 Portlet 和 servlet 不允许有效的存储，以便 Web 服务器可以轻松地定位这样的文件。当前方法不会轻松地允许这样的文件被动态地创建或允许多个并发用户使用相同的 portlet 来访问这样的文件。如此，门户应用程序有必要用将允许程序员更加轻松地编写可以在门户服务器内运行的基于图形的应用程序的技术来实现。

发明内容

提供了一种系统和方法，它们用于向最终用户提供编写数据包并上传程序以在他们自己的可与其他最终用户共享的门户服务器的门户环境内运行的能力。程序是作为小程序来实现的并作为 Java 存档 (JAR) 格式化文件来封装，这些文件可以被限制到在有限大小的工作空间内编写文件，以访问任何托管的数据库（如果授予适当的访问权限的话）。根据实施例，程序可以作为可以显示在最终用户门户上的 .jpg 文件或作为可扩展的矢量图类型文件来在服务器上创建图像内容。在一个实施例中，鼠标点击可以是转换为作为用于生成新图像的鼠标坐标发送回来 SOAP 消息或操作 URL。

附图说明

当结合下列附图考虑所说明的实施例的下列详细描述时，可以对本发明作出更好的理解，其中：

- 图 1 是支持所声明的主题的示范性系统体系结构的方框图；
- 图 2 是支持所声明的主题的门户服务器系统的方框图；以及
- 图 3 是显示根据本发明的实施例的方法的流程图。

具体实施方式

虽然是对编程语言进行特定参考而进行描述的，但是，当前发明

的系统和方法可以对于诸如口头语言等等之类的任何解释语言来实现。图 1 显示了示范性体系结构 100，其中，实现了根据本发明的系统。那些精通计算技术的人将认识到，除了下面将描述的应用程序和体系结构外，所说明的实施例与各式各样的应用程序和体系结构相关。此外，本发明的功能可以以软件、硬件或软件和硬件的组合来实现。硬件部分可以使用专门的逻辑来实现；软件部分可以存储在存储器中或存储在记录介质中，并由诸如微处理器之类的适当的指令执行系统来执行。

在本文档的上下文中，“存储器”或“记录介质”可以是包含、存储、传递、传播或传输程序和/或数据以便供指令执行系统、设备使用的任何装置。存储器和记录介质可以是，但不限于，电子、磁性、光学、电磁、红外线或半导体系统、设备。存储器和记录介质还包括，但不限于，下列各项：便携式计算机磁盘、随机存取存储器（RAM）、只读存储器（ROM）、可擦与可编程只读存储器（EPROM 或闪速存储器），以及便携式光盘只读存储器或在其上面可以存储程序和/或数据的其他合适的介质。

图 1 是支持所声明的主题的示范性系统体系结构 100 的方框图。系统 100 包括计算设备 101，在该示例中是个人计算机（PC）。显示器 103、键盘 105，以及鼠标 107 连接到 PC 101。显示器 103、键盘 105 和鼠标 107 向用户提供了与 PC 101 进行交互的装置。PC 101 中还包括中央处理单元（CPU）109 和记录介质，或数据存储 111。那些精通计算技术的人应该熟悉 PC 101 以及相关的组件 103、105、107、109 和 111。

PC 101 通过连接 117 连接到因特网 115。具有数据存储 121 的数据服务器 119 也连接到因特网 115。PC 101 的用户可以通过因特网 115 访问各种信息源，如数据存储 119 上的源文件 123。

PC 101 可以被配置为处理来自客户端应用程序的请求的 Web 服务器，也可以被配置为向连接的 Web 服务器发送请求的客户端。

根据实施例，PC 101 可以被配置为包括根据实施例的门户应用

程序的 Web 服务器。具体来说,请参看图 2,根据实施例,Web 应用程序服务器 200 被配置为允许程序员使用在 Web 应用程序服务器 200 上运行的小程序 210 作为 portlet 220 的小程序接口。portlet 220 被配置为调用小程序 210。图 2 还显示了被配置为提供最终用户可以在其上面绘画的帆布/图形 240 的门户容器 230。门户/小程序接口 240 被配置为帆布,该帆布操作以允许用户通过跟踪用户可以点击的位置传递鼠标坐标,例如,后者被传递到服务器以允许服务器端小程序通过 Applet API 作出响应。从而,Applet API 创建图形门户。在一个实施例中,提供了基于 SVG 的图形,以便在返回的 Web 页面中生成 SVG 标记。

门户容器 230 可以被配置为与 Java™ 存档 (JAR) 文件 250 进行操作。JAR 格式文件 250 可使用户将多个文件束存储在单个存档文件中。JAR 文件 250 被配置为包含与根据本发明的小程序和应用程序关联的类文件和辅助资源。JAR 文件 250 可以允许用户对 JAR 文件的内容进行数字签名。

根据实施例,小程序 210 被捆绑在 JAR 文件 250 中,小程序的类文件和关联的资源在单个 HTTP 事务中被下载到浏览器中。JAR 文件格式 250 定义扩展的封装,以便添加功能。存储在 JAR 文件存储在数据包可以可选地被密封,以便数据包可以实施版本一致性。将数据包密封在 JAR 文件内假设,可以在相同 JAR 文件中查找该数据包中定义的所有类。以前的解决方案要求用户用在 JAR 文件中编码的 APPLET(Web 页面,具有 APPLET 标记)来访问 Web 页面。Web 页面是由浏览器读取的。JAR 文件被下载到浏览器高速缓存中,并在浏览器沙箱内执行,以防止任何文件在浏览器机器上发生改变。

实施例假设,门户/小程序接口 240 被配置为要与小程序 210 可操作的 applet API,并与 Web 浏览器一起工作。这里根据实施例的 Applet API 可以由 Applet 类的 java.applet 数据包和 AppletContext 接口来提供。Applet API 240 可以被配置以允许加载

相对于小程序（或它在其中运行的页面）的 URL 指定的数据文件；显示短的状态字符串；允许文档显示在浏览器上；查找其他小程序；播放声音；以及检索 <APPLET> 标记中的用户指定的参数。如此，实施例允许门户程序员在通过门户-到-小程序接口适配器编写门户应用程序时选择 APPLETT API。适配器可以被配置为向小程序提供图形，以便由程序员在绘画上使用。

例如，当执行 `paint()` 时，适配器将数据转换为 .JPG 或 .SVG，将它传输到浏览器窗口，以便与其他门户内容一起显示。如此，门户小程序在应用程序 Web 服务器 200 上运行，而不是在浏览器的计算机上运行。

具体来说，Applet API 240 进一步可使 portlet 220 接收用户操作，并将它们传递到 Applet 210 上，并提供交互图形。作为响应，Applet 210 根据绘图方法或绘画的其他适当方法执行绘画。Applet 210 退出绘图方法，以使 portlet 220 将生成的一切返回到用户。如此，Applet API 240 运行，以将门户方法映射到 Applet 210 方法。

现在请参看图 3，流程图显示了根据实施例的方法。区块 310 用于在门户接收用户生成的绘画位置标识符，这些位置可以是鼠标点击、敲击以及通过笔等等进行的平板 PC 图形指示。区块 320 用于将用户生成的绘画位置标识符通过 JAR 文件内的 Applet API 映射到小程序功能。区块 330 使用用户生成的绘画位置标识符生成绘画。区块 340 用于退出小程序中的绘图方法。区块 350 将生成的绘画传输到用户。

在一个实施例中，门户网站可以被配置为使用多个门户，这些门户可以用来提供 Web 页面的某些部分。根据实施例，自定义门户被配置为 Applet API 以使用户通过诸如 `Applet.paint()` 方法之类的方法在用户可查看的图形区域进行绘画。可以根据实施例来对网站进行配置，以提供门户适配器以将门户 API 重新映射到 Applet API，以便小程序能够在门户环境内运行。如此，当用户从客户端机器上传数据时，数据流到门户服务器，并从门户服务器到门户，并从门户到包

装的小程序。

一旦小程序接收用户输入数据，Applet 可以在图形上进行绘画。然后，门户可以将图形保存在 JPG 文件或以其他适当的文件格式进行保存。此后，门户可以被配置为对于包含 .JPG 文件（例如，基于 .SVG 绘画命令的文件）的 Web 页面传输通用资源定位器 (URL)。门户服务器可以被配置为将门户 Web 页面组合在一起，并将组合的 Web 页面传输到用户。

虽然是参考特定实施例特别显示和描述本发明的，但是那些精通本技术的普通人员将理解，在不偏离本发明的精神和范围的情况下，可以在形式上和细节上进行前述的及其他变化，包括但不限于，以相同或不同的顺序执行的较多的、较少的或修改的元素和/或较多的、较少的或修改的步骤。

图1

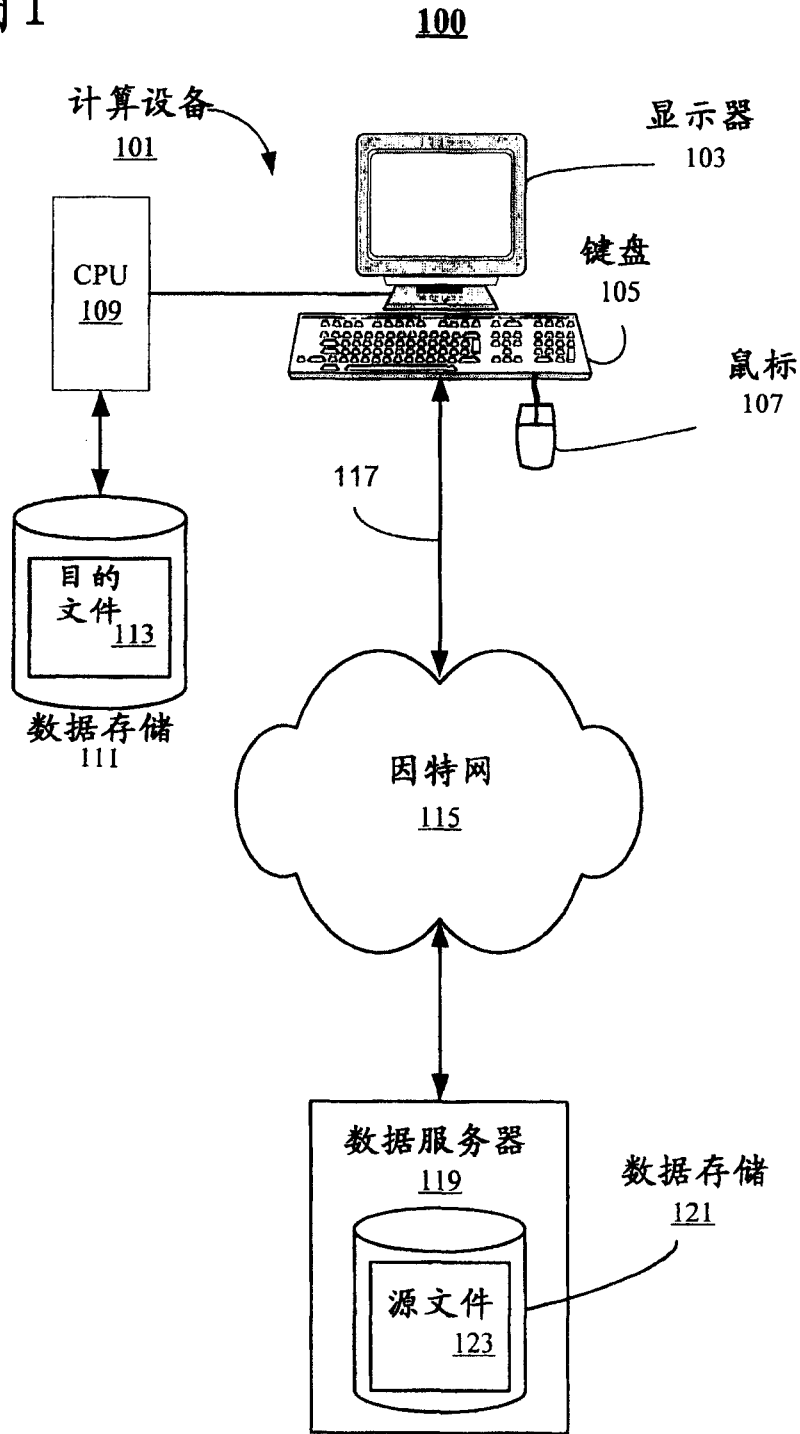


图2

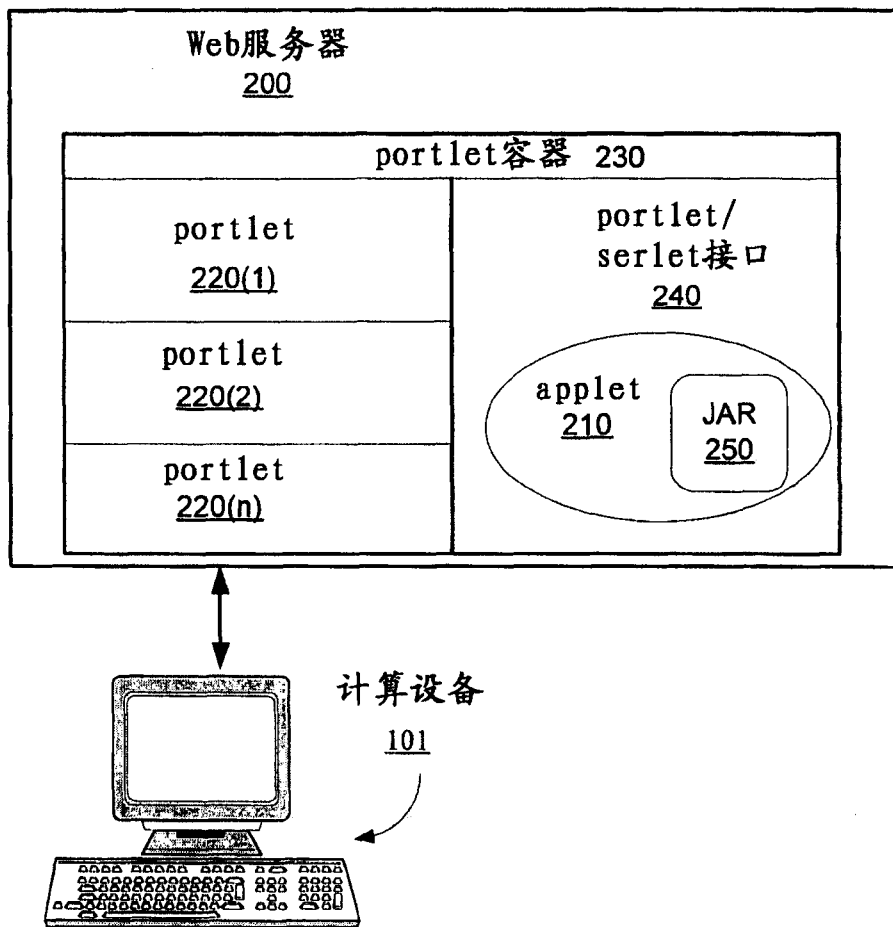


图 3

