

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.



[12] 发明专利说明书

专利号 ZL 02824587.3

G06F 17/30 (2006.01)
G06F 12/00 (2006.01)
G06F 15/80 (2006.01)
G06F 15/16 (2006.01)
H04M 3/42 (2006.01)
H04L 12/28 (2006.01)

[45] 授权公告日 2009年6月24日

[11] 授权公告号 CN 100504853C

[22] 申请日 2002.10.24 [21] 申请号 02824587.3
[30] 优先权

[74] 专利代理机构 北京市柳沈律师事务所
代理人 邵亚丽 马莹

[32] 2001.10.24 [33] US [31] 60/386,487
[86] 国际申请 PCT/US2002/034309 2002.10.24
[87] 国际公布 WO2003/036521 英 2003.5.1
[85] 进入国家阶段日期 2004.6.9

[73] 专利权人 BEA 系统公司
地址 美国加利福尼亚州

[72] 发明人 丹尼尔·塞尔曼 罗伯特·伯格曼
爱德华·K·奥尼尔

[56] 参考文献
EP1130513A2 2001.9.5
US5907705A 1999.5.25
CN1252565A 2000.5.10
审查员 颜世莹

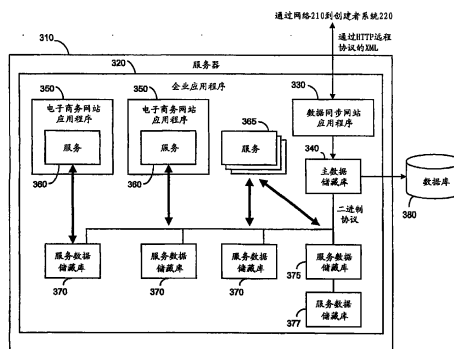
权利要求书4页 说明书9页 附图6页

[54] 发明名称

同步应用程序数据并将该数据部署到远程服务器的方法

[57] 摘要

在一个或多个服务器(230、240、250)上运行的企业应用程序(320)所用的应用程序数据可与从其他地方部署的应用程序数据进行同步。应用程序数据可通过 HTTP 以 XML 文件传输来使与网络可用的服务器的同步更容易。多个服务数据储藏库(370、375)可与在企业应用程序中运行的主数据储藏库(340)中驻留的应用程序数据进行同步。通过使用代理数据储藏库也提供了服务器到服务器的数据同步功能。应用程序数据文件可从创建者系统(220)部署到远程服务器。新创建应用程序数据文件可使用源码控制系统在创建者系统(220)中划分版本,并且与为将应用程序数据部署到产品服务器作准备的多阶段测试过程一起,由创建者可选择性地部署到各种服务器去。



1. 一种用于同步应用程序数据的方法，包括：

在数据同步网络应用程序处接收同步请求，所述数据同步网络应用程序运行在网络服务器上；

由数据同步网络应用程序发送新应用程序数据至在所述网络服务器上的、根据所述同步请求待同步的主数据储藏库，其中所述主数据储藏库与所述新应用程序数据同步；

将所述新应用程序数据从所述主数据储藏库发送至连接至所述网络服务器的关系数据库管理系统；

当所述主数据储藏库被同步时，由所述主数据储藏库通知在所述网络服务器上的服务数据储藏库；以及

在接收来自所述主数据储藏库的通知之后，由所述服务数据储藏库将在所述服务数据储藏库中的数据与在所述主数据储藏库中的所述新应用程序数据同步。

2. 如权利要求1所述的方法，还包括：

当在所述服务数据储藏库中的数据与在所述主数据储藏库中的所述新应用程序数据同步时，由所述服务数据储藏库通知所述主数据储藏库；以及

在所述主数据储藏库中保持指示已经将所述服务数据储藏库同步的时间的日志。

3. 如权利要求1所述的方法，其中所述服务数据储藏库包括第一数据储藏库，所述方法进一步包括：

当在所述第一数据储藏库中的数据与所述新应用程序数据同步时，由所述第一数据储藏库通知在所述网络服务器上的第二数据储藏库；以及

在从所述第一数据储藏库接收所述通知之后，由所述第二数据储藏库将存储在所述第二数据储藏库中的数据与所述新应用程序数据同步。

4. 如权利要求1所述的方法，还包括：

在所述主数据储藏库接收所述新应用程序之后，从所述主数据储藏库发送所述新应用程序数据至在所述网络服务器上运行的代理数据储藏库；

从所述代理数据储藏库向运行在分离的网络服务器上的第二数据同步网络应用程序转发所述新应用程序数据；

由所述第二数据同步网络应用程序向在所述分离的网络服务器中的、待与所述新应用程序数据同步的第二主数据储藏库发送所述新应用程序数据；

向在使用第二主数据储藏库的所述分离的网络服务器中的服务数据储藏库通知第二主数据储藏库收到所述新应用程序数据；以及

在从第二主数据储藏库接收所述通知之后，将在位于所述分离的网络服务器中的所述服务数据储藏库中的数据与在位于所述分离的网络服务器的、使用所述服务数据储藏库的第二主数据储藏库中的所述新应用程序数据同步。

5. 如权利要求 4 所述的方法，其中：

所述代理数据储藏库运行在第一企业应用程序中；并且

所述第二数据同步网络应用程序和第二主数据储藏库运行在第二企业应用程序中。

6. 如权利要求 1 所述的方法，还包括：

在所述通知和同步步骤之后，由所述服务数据储藏库轮询所述主数据储藏库，以确定所述主数据储藏库是否已经接收附加新应用程序数据；以及

当所述轮询结果指示所述主数据储藏库已经接收附加新应用程序数据时，由所述服务数据储藏库将在所述服务数据储藏库中的数据与在所述主数据储藏库中的所述附加新应用程序数据同步。

7. 如权利要求 1 所述的方法，还包括：

在所述服务数据储藏库中保持指示已经将所述服务数据储藏库与在所述主数据储藏库中的数据同步的时间的日志。

8. 如权利要求 1 所述的方法，其中：

所述同步步骤由 J2EE 企业应用程序执行。

9. 如权利要求 1 所述的方法，其中：

所述新应用程序数据由在第一企业应用程序中运行的服务使用。

10. 如权利要求 9 所述的方法，其中：

在与所述新应用程序数据同步之前，所述数据同步网络应用程序在第二企业应用程序中运行。

11. 如权利要求 1 所述的方法，其中：

所述新应用程序数据定义了所述数据同步网络应用程序的业务逻辑。

12. 如权利要求 1 所述的方法，其中：

所述新应用程序数据包括至少一个 XML 文件。

13. 如权利要求 1 所述的方法，其中：

所述新应用程序数据包括适合于通过 HTTP 传输的应用程序数据文件。

14. 如权利要求 1 所述的方法，其中：

所述方法作为准备将所述新应用程序数据部署到产品服务器的测试过程的一部分执行。

15. 一种用于通过网络部署应用程序数据到远程网络服务器的方法，包括：

由数据同步网络应用程序提交新应用程序数据到在所述网络服务器上的主数据储藏库并且根据同步命令同步所述新应用程序数据；

将根据所述同步命令同步的所述新应用程序数据的子集从所述主数据储藏库发送至在所述网络服务器上的第一服务数据储藏库；

由所述第一服务数据储藏库通知在所述网络服务器上的第二服务数据储藏库由所述第一服务数据储藏库接收所述新应用程序数据中的至少一部分；以及

在从所述第一服务数据储藏库接收所述通知之后，由所述第二服务数据储藏库将在所述第二服务数据储藏库中的数据与在所述第一服务数据储藏库中的所述新应用程序数据的子集同步。

16. 如权利要求 15 所述的方法，还包括：

在所述通知和同步步骤之后，由所述第二服务数据储藏库轮询所述主数据储藏库，以确定所述主数据储藏库是否已经接收附加新应用程序数据；以及

当所述轮询结果指示所述主数据储藏库已经接收附加新应用程序数据时，由所述第二数据服务储藏库将在所述第二服务数据储藏库中的数据与所述附加新应用程序数据同步。

17. 如权利要求 15 所述的方法，其中：

所述新应用程序数据包括至少一个 XML 文件。

18. 如权利要求 15 所述的方法，其中：

所述新应用程序数据包括适合于通过 HTTP 传输的应用程序数据文件。

19. 如权利要求 15 所述的方法，其中：

所述方法作为准备将所述新应用程序数据部署到产品服务器的测试过程

的一部分执行。

20. 一种用于同步应用程序数据的方法，包括：

由第一数据同步网络应用程序发送新应用程序数据至在第一网络服务器上的第一主数据储藏库；

由第一主数据储藏库发送所述新应用程序数据到在第一网络服务器上的代理数据储藏库；以及

从代理数据储藏库转发所述新应用程序数据到运行在第二网络服务器上的第二数据同步网络应用程序；

由所述第二数据同步网络应用程序向在所述第二网络服务器中的、待与所述新应用程序数据同步的第二主数据储藏库发送所述新应用程序数据；

向在使用第二主数据储藏库的分离的网络服务器中的服务数据储藏库通知第二主数据储藏库收到所述新应用程序数据；以及

在从第二主数据储藏库接收所述通知之后，将在位于所述分离的网络服务器中的所述服务数据储藏库中的所述数据与在位于所述分离的网络服务器的、使用所述服务数据储藏库的第二主数据储藏库中的所述新应用程序数据同步。

21. 如权利要求 20 所述的方法，其中：所述代理数据储藏库运行第一企业应用程序并且所述第二数据同步网络应用程序运行第二企业应用程序。

22. 如权利要求 20 所述的方法，其中：

所述新应用程序数据包括至少一个 XML 文件。

23. 如权利要求 20 所述的方法，其中：

所述新应用程序数据包括适合于通过 HTTP 传输的应用程序数据文件。

24. 如权利要求 20 所述的方法，其中：

所述方法作为准备将所述新应用程序数据部署到产品服务器的测试过程的一部分执行。

同步应用程序数据并将该数据部署到远程服务器的方法

版权通告

本专利文本所公开内容的一部分包含有受版权保护的材料。版权所有人不反对任何人对专利文档或专利公开内容进行如专利和商标局中的对专利文件和专利公开的翻印复制，但除此之外将保留所有的各种版权权利。

相关申请的参照引用

本申请声明要求在2001年10月24日提交的美国临时专利申请No.60/386,487(律师案号No. BEAS1112US0)的权利，并在此引用作为参考。

发明背景

1. 技术领域

本发明涉及部署和同步数据的技术。

2. 背景技术

基于网络的软件应用程序已作为强大的工具出现，以同样的方式提供有用的服务给卖主和顾客。电子商务网络应用程序可配置为依照业务逻辑而运作来实现有助于电子商务的各种过程。这种应用程序也可提供与远程互联网客户交互的服务和/或应用程序。

Java2 平台企业版(J2EE)，来自于加州圣克莱拉的 Sun 微系统公司，有助于在多层企业应用程序环境中运行的电子商务网络应用程序的开发。这种网络应用程序的运作由用于配置网络应用程序的数据(“应用程序数据”)来控制。因此，通过更新与应用程序关联的应用程序数据，可修改由网络应用程序所提供的功能。

遗憾的是，从新的或更新的应用程序数据到 J2EE 网络应用程序的部署和同步工作是麻烦的过程。新创建的应用程序数据通常依照数据库所使用的特定模型存储在数据库中。如果开发者试图部署这种应用程序数据到运行中的产品服务器，数据在转换成适合传输到服务器的形式之前必须先从数据库

中提取出来。通常都需要脚本来从数据库导出应用程序数据到文件系统中。导入应用程序数据到运行中的服务器也需要脚本。因为这些原因，维护应用程序数据的不同版本也会成为一种笨拙和难做的努力。

此外，在新应用程序数据可被移动到运行中的产品服务器之前，可能需要部署应用程序数据到多个服务器以遵循多阶段测试过程。这种应用程序数据迁移可能会是麻烦的手工过程，在应用程序数据从一个服务器迁移到另一个服务器并且涉及多个人时会变得愈加容易出错。因此，在 J2EE 平台环境中部署和同步应用程序数据的现有技术很大程度上是专用的和不方便的。

发明内容

本发明，粗略地说，公开了同步运行在一个或多个服务器上的企业应用程序所用的应用程序数据。例如，在一个实施例中，接收应用程序数据并与数据库和数据储藏库同步。在另一个实施例中，提供了应用程序数据部署方法，允许应用程序数据的创建、提交到源码控制系统、以及发送到其数据储藏库与数据同步的远程服务器。

在另一个实施例中，通过使用代理数据储藏库来在企业应用程序之间同步应用程序数据。在另一个实施例中，通过使用一个数据储藏库询问另一个来在数据储藏库之间同步应用程序数据。

系统和计算机可读介质也提供作为这些方法的实现部分和变体。如本公开内容中所述，也可能有其他许多实施例。

附图说明

图 1 展示了如本发明的实施例所述的用于创建应用程序数据文件和划分其版本的系统的概念图。

图 2 展示了如本发明的实施例所述的用于同步数据的系统的高层方框图。

图 3 展示了如本发明的实施例所述的运行着具有可被同步的数据的企业应用程序的服务器的方框图。

图 4 是展示了如本发明的实施例所述的用于同步数据的过程的流程图。

图 5 展示了如本发明的实施例所述的运行着企业应用程序的多个服务器的方框图，该企业应用程序带有通过使用代理数据储藏库可被同步的数据。

图 6 展示了如本发明的实施例所述的运行着企业应用程序的一群服务器的方框图，该企业应用程序带有可被管理服务器同步的数据。

具体实施方式

图 1 展示了如本发明的实施例所述的用于创建应用程序数据文件和划分其版本的系统的概念图。创建者 110 是具有创建可被一个或多个网络应用程序和/或服务所使用的应用程序数据的职责的人。在各种实施例中，创建者 110 可以是业务线(line-of-business)经理、业务工程师、开发人员或负责创建应用程序数据的其他人。应用程序数据可被用于配置网络应用程序来依照创建者 110 所定义的业务逻辑运作。例如，创建者 110 创建的应用程序数据可包括比如活动定义的业务政策文档、计划说明书、业务规则集、门户定义和/或网络应用程序使用的其他类型数据。

为了产生应用程序数据，创建者 110 可与各种软件工具建立联系，比如控制中心 120。在一个实施例中，控制中心 120 是来自加州圣约斯的 BEA 系统公司的电子商务控制中心工具。创建者 110 使用控制中心 120 创建的数据可作为一组应用程序数据文件存储在本地层次文件系统中。图 1 展示了存储在层次文件系统中的应用程序数据文件 130 的例子。应用程序数据文件 130 可根据需要创建它们的网络应用程序来组织。在一个实施例中，由控制中心 120 创建的应用程序数据存储在遵循由需要创建(author)这些数据的网络应用程序所解释的 XML(可扩展标记语言)模型的 XML 文件中。因为 XML 可被跨越多种平台的各种系统所读取和解释，使用 XML 应用程序数据要优于数据库面向脚本方法。

创建者 110 和控制中心 120 交互所产生的应用程序数据文件 130 可被检入源码控制系统 140。源码控制系统 140 允许应用程序数据文件 130 的不同版本可为测试、部署、同步和/或其他用途而很容易地取回。例如，如果网络应用程序部署了有错的应用程序数据，源码控制系统 140 允许创建者 110 来取回应用程序数据的先前版本和重新部署数据到网络应用程序。可当并行创建应用程序数据时，使用源码控制系统 140 也避免多个创建者覆盖彼此的文件。在一个实施例中，源码控制系统 140 是本领域中所知的任何适用的源码控制系统。

在创建系统数据后，典型地要遵循多阶段核准过程来测试新创建的应用

程序数据和其他创建者所创建的其他数据和/或应用程序之间的交互。例如，在应用程序数据部署到产品服务器之前，会遵循独立的开发和质量保证阶段。这些阶段帮助确保在产品服务器上运行的网络应用程序将可靠地运作。

在网络应用程序的这种协作开发中，软件开发者会很希望同步跨越企业应用程序界限来将应用程序数据与远程服务器同步。此后进一步描述的各种数据同步过程利用(leverage)了由 HTTP(超文本传输协议)提供的优势来使可通过互联网访问的远程服务器上的应用程序数据同步变得更容易。如本发明的特定实施例所述，新创建的应用程序数据可与一个或多个正运行的远程服务器同步。通过识别服务器所关联的 URL(统一资源定位符)，应用程序数据可远程地与经过 HTTP 通过互联网可访问的任何服务器同步。

图 2 展示了如本发明的实施例所述的用于同步数据的系统的高层方框图。图 2 的系统可用于在将网络应用程序最终部署到产品服务器上之前协作开发和测试它们。在一个实施例中，图 2 的系统实现为使用来自加州圣约斯的 BEA 系统公司的 BEA WebLogic 平台的一个或多个组件。如图 2 中所示，多个服务器与网络 210 通讯。在一个实施例中，图 2 的系统实现为点对点网络。图 2 的每个服务器与可存储运行在服务器上的网络应用程序和/或服务所用的应用程序数据的数据库相关联。如图 2 所示，在多阶段应用程序数据核准过程中，每个服务器可用于不同的用途。例如，这些用途可包括：开发(服务器 230 和数据库 235)、质量保证(服务器 240 和数据库 245)、产品化(服务器 250 和数据库 255)和其他(未显示)。

图 2 的每个服务器通过网络 210 能够接收来自于创建者系统 220 的应用程序数据。接收自创建者系统 220 的应用程序数据可与此后进一步描述的各种服务器上运行的网络应用程序所用的应用程序数据同步。在一个实施例中，创建者系统 220 是图 1 中展示的系统。

服务器工具 260 和数据库工具 265 也在图 2 中展示。这些工具允许分别对在产品服务器 250 和产品数据库 255 上的数据进行实时修改。在一个实施例中，服务器工具 260 是基于 JSP(Java 服务器端页面)的工具，允许直接地施加变化到产品服务器 250 去。

图 3 展示了如本发明的实施例所述的运行着有可被同步的数据的企业应用程序的服务器的方框图。服务器 310 可为图 2 中展示的能够接收来自于创建者系统 220 的应用程序数据的任何服务器。如图 3 中所示，多个网络应用

程序和服务驻留在服务器 310 上运行的企业应用程序 320 中。电子商务网络应用程序 350 包括依赖于在服务数据储藏库 370、375 和/或 377 中存储的应用程序数据的服务 360。在企业应用程序 320 环境中运行的附加服务 365 也可依赖于在服务数据储藏库中存储的应用程序数据。在一个实施例中，服务数据储藏库是作为对于已被部署的服务类而言重要的、在数据项的内存中缓冲的 Java 对象。

数据同步网络应用程序 330 也运行在企业应用程序 320 中并且负责执行如本发明所述的数据同步操作。如图 3 所示，数据同步应用程序 330 可通过网络 210 与图 2 的创建者系统 220 通讯。企业应用程序 320 中的主数据储藏库 340 可与数据同步应用程序 330、数据库 380 和数据储藏库 370 及 375 通讯。

数据同步应用程序 330 可拦截通过网络 210 从创建者系统 220 发送的同步请求。在一个实施例中，这些同步请求遵循远程协议通过 HTTP 作为 XML 发送。当数据同步应用程序 330 接收到来自创建者系统 220 的同步请求时，它引导接收自创建者系统 220 的应用程序数据到作为应用程序数据的运行时间表示的主数据储藏库 340 去。数据同步应用程序 330 也引导所接收的应用程序数据到作为应用程序数据的持久性存储位置的数据库 380 去。在一个实施例中，数据储藏库 340 以企业 Java Beans (EJB) 存储到来的应用程序数据。在另一个实施例中，数据库 380 是关系数据库管理系统。

在一个实施例中，数据同步应用程序 330 包括一组 JSP，它们能够报告：主数据储藏库 340 的内容、与主数据储藏库 340 成功同步的数据储藏库和在企业应用程序 320 中每个数据储藏库的内容。

图 3 的服务数据储藏库中的每个都包含了存储在主数据储藏库 340 中的应用程序数据的子集。服务 360 和 365 使用服务数据储藏库中的数据来执行商业任务。例如，规则服务可使用数据来评定顾客是否适合特定的顾客区段 (segment)。因为图 3 的服务数据储藏库只维护在主数据储藏库 340 中数据的子集，每个服务能够解析服务所需要的应用程序数据的一个小的特定集合，而不需要维护在主数据储藏库 340 中维护的所有应用程序数据的副本。

每次主数据储藏库 340 同步时，它通知数据储藏库 370 和 375。这些数据储藏库于是将这些数据子集与主数据储藏库中的数据进行同步。在一个实施例中，应用程序数据遵循二进制协议在企业应用程序 320 的组件之间通讯。

运行在服务器 310 上的多个企业应用程序(未显示)也可使用运行在每个企业应用程序中的独立数据同步应用程序进行同步。代理数据储藏库可用来辅助这种同步。

在某些情况下,主数据储藏库 340 不直接将数据与数据储藏库同步。而是,使用多个数据储藏库来形成通知链。参见图 3,数据储藏库 375 和 377 建立了通知链。当数据储藏库 375 接收到来自主数据储藏库 340 的更新后的应用程序数据时,它通知数据储藏库 377。数据储藏库 377 于是可与数据储藏库 375 所接收的应用程序数据进行同步。因此,储藏库 375 和 377 都可以不需要在主数据储藏库 340 发送更新后的应用程序数据到这些储藏库的条件下进行同步。

图 4 是展示了如本发明的实施例所述的用于同步数据的过程的流程图。当应用程序数据创建者系统 220 与图 2 的任何服务器进行同步时,可通过从创建者系统 220 发送同步请求到待同步服务器的 URL 来发起同步过程,在那里请求会被数据同步应用程序 330 拦截。在各种实施例中,创建者 110 可使用控制中心 120 的图形界面或 Java 命令行发起同步过程。

在本发明的各种实施例中,可使用不同的同步模式。在从客户刷新模式中,同步所有用于特定应用程序的应用程序数据。使用这种模式,所有数据从数据储藏库中清除,然后将所有相关应用程序数据文件从创建者的本地文件系统发送到数据同步网络应用程序。这种模式有助于将内存中数据与持久性数据进行同步,或者从同步错误中恢复。在非客户(one-way-from-client)模式中,只更新那些已被删除、更新或者在创建者的本地文件上创建的应用程序数据文件。于是,将在创建者系统和待同步服务器之间的数据传输量减到最少。

当在步骤 410 中的同步过程初始化时,待同步应用程序数据将通过网络 210 从创建者系统 220 发送到服务器 310(步骤 420)。在一个实施例中,应用程序数据使用 POST 命令通过 HTTP 作为 XML 文件发送。应用程序数据由数据同步应用程序 330 接收。数据同步应用程序 330 于是将数据库 380 和主数据储藏库 340 与新接收的应用程序数据进行同步(步骤 430)。作为步骤 430 的结果,数据库 380 包含以用于持久性存储的数据库格式来存储的更新后的应用程序数据,并且主数据储藏库 340 包含以 EJB 运行时间格式存储的更新后的应用程序数据。

如此前所述，图3的各种应用程序和服务会依赖于驻留在服务数据储藏库中的应用程序数据。从而，每个服务数据储藏库可订阅接收更新后的应用程序数据。为了更新在数据储藏库370和375中的应用程序数据，主数据储藏库340通知服务数据储藏库370和375中的每一个：以这些储藏库已订阅接收的任何更新后的应用程序数据（步骤440）。如前所述，服务数据储藏库377通过通知链接接收它的通知。主数据储藏库340也通知运行在其他服务器上的企业应用程序所用的代理数据储藏库（未显示），如此后进一步描述。在服务数据储藏库已被通知更新后的应用程序数据之后，每个储藏库将自己与它所订阅的主数据储藏库中的特定应用程序数据进行同步（步骤450）。代理数据储藏库也转发在步骤440中发送的任何通知到它们的远程数据同步网络应用程序，以同步在服务器310上的其他企业应用程序（未显示）中运行的储藏库或其他远程服务器（步骤460），如此后进一步描述。

服务数据储藏库370和/或375也可周期性询问主数据储藏库340来检查应用程序数据是否已更新。如果检测到更新后的数据，则在主数据储藏库340中维护的应用程序数据可与服务数据储藏库370和/或375同步。

主数据储藏库340维护着描述每次更新的日志和包括了对哪个数据储藏库已成功更新和哪个没有的描述。数据同步应用程序330可通过HTTP或Java命令解释器返回状态消息到创建者系统220来标识哪个应用程序数据文件已成功同步。图3的每个数据储藏库也可维护描述当前内容的日志。

如图3中所示，应用程序数据可在创建者系统和运行在单个服务器上的企业应用程序之间同步。如本发明的各种实施例所述，多个服务器也可通过在图4的步骤410中发起的单个同步过程同步。通过使用在图5中阐述的附加部件，可辅助这种多个服务器的同步。

图5展示了如本发明的实施例所述的运行着企业应用程序的多个服务器的方框图，该企业应用程序带有通过使用代理数据储藏库可被同步的数据。企业应用程序620和660分别运行在服务器610和650上。运行在企业应用程序620中的主数据储藏库630可被数据同步应用程序（未显示）同步，如此前所述，该数据同步应用程序与创建者系统（未显示）通讯。主数据储藏库630与也在企业应用程序620中运行的代理数据储藏库640通讯。代理数据储藏库640可订阅，以接收在主数据储藏库630中存储的更新后的应用程序数据。

数据同步网络应用程序680是运行在远程服务器650上的数据同步应用

程序的实例。但是，数据同步应用程序 680 并非直接从创建者系统 220(类似于图 3 的数据同步应用程序 330)接收更新，而是接收从代理数据储藏库 640 发送的更新。远程主数据储藏库 670 是用于企业应用程序 660 的主数据储藏库，该企业应用程序依照由数据同步应用程序 680 接收的应用程序数据进行更新。远程主数据储藏库 670 可传递更新后的应用程序数据到在如此前讨论的在企业应用程序 660 中运行的其他代理和/或服务数据储藏库(未显示)。应该理解，在图 5 的服务器上可出现附加部件(未显示)，比如在图 3 和/或图 6 的服务器中阐述的一个或多个部件。

在操作中，主数据储藏库 630 从与诸如图 2 的创建者系统 220 一样的创建者系统(未显示)通讯的数据同步网络应用程序接收更新后的应用程序数据。代理数据储藏库 640 订阅，以接收来自于主数据储藏库 640 的更新后的应用程序数据。因此，主数据储藏库 630 接收的更新后的应用程序数据被传递到代理数据储藏库 640。代理数据储藏库 640 于是从服务器 610 传递应用程序数据到运行在服务器 650 上的数据同步应用程序。在一个实施例中，应用程序数据由代理数据储藏库 640 遵循二进制协议接收。从而，在以 XML 通过 HTTP 传递应用程序数据到在远程服务器 650 中的数据同步应用程序 680 之前，代理数据储藏库 640 将应用程序数据编制成 XML 格式。

当数据同步应用程序 680 接收更新后的应用程序数据时，数据同步应用程序 650 开始用更新后的应用程序数据更新远程主数据储藏库 670。如图 3 中所示的持久性存储数据库(未在图 5 中显示)也被同步，并且更新后的应用程序数据则转发到企业应用程序 660 的所有数据储藏库(未显示)，该数据储藏库订阅以接收更新后的应用程序数据。

于是，通过订阅代理数据储藏库来接收来自于主数据储藏库的更新，运行在远程服务器上的企业应用程序中的应用程序数据可用由主数据储藏库接收的应用程序数据来同步。也可用代理跨越运行在相同服务器上的不同企业应用程序来同步数据。此外，通过经由代理链接远程服务器到附加远程服务器，可同步多个服务器的链。这些准则也可进一步用于对跨越如图 6 中阐述的服务器群集的应用程序数据进行同步。

图 6 展示了如本发明的实施例所述的一群服务器的方框图，该服务器运行着具有可被管理服务器同步的数据的企业应用程序。在管理服务器 710 上的企业应用程序 720 的数据同步网络应用程序 730 和主数据储藏库 740 可与

群集 790 的被管理服务器 760 通讯。如前所述, 在远程服务器 760 上运行的企业应用程序 770 的数据储藏库(未显示)中驻留的应用程序数据可与在主数据储藏库 740 中存储的更新后的应用程序数据进行同步。主数据储藏库 740 可传递更新后的应用程序数据到代理数据储藏库(未显示), 代理数据储藏库于是传递数据到在企业应用程序 770 上运行的数据同步网络应用程序(未显示)。因此, 多个服务器 760 可被从创建者系统(未显示)接收更新后的应用程序数据的单个管理服务器 710 同步。应该理解, 在图 6 的服务器上可出现附加部件(未显示), 比如在图 3 和/或图 5 的服务器中阐述的一个或多个部件。

如果可行, 本发明可使用硬件、软件或硬件和软件的组合来实现。符合本发明的软件, 比如程序代码和/或数据, 可存储在一个或多个计算机可读介质上。在不脱离本发明的精神的情况下, 如果也可行, 此处阐述的各种硬件组件和/或软件组件可组合成包括软件、硬件或两者的复合组件。类似地, 在不脱离本发明的精神的情况下, 如果可行, 此处阐述的各种硬件组件和/或软件组件可分解成包括软件、硬件或两者的子组件。此外, 如果可行, 应该想到, 软件组件可实现为硬件组件, 和反之亦然。而且, 如果可行, 这里阐述的各种步骤可组合成复合步骤和/或分解成子步骤。也应该想到, 这里阐述的软件组件可使用一个或多个一般用途或特定用途的计算机和/或计算机系统、网络和/或其他来实现。

上述公开内容不是要将本发明局限在所公开的特定应用程序领域的精确形式里。应该想到, 根据本公开内容的本发明的各种替代实施例和/或修改都是可能的。

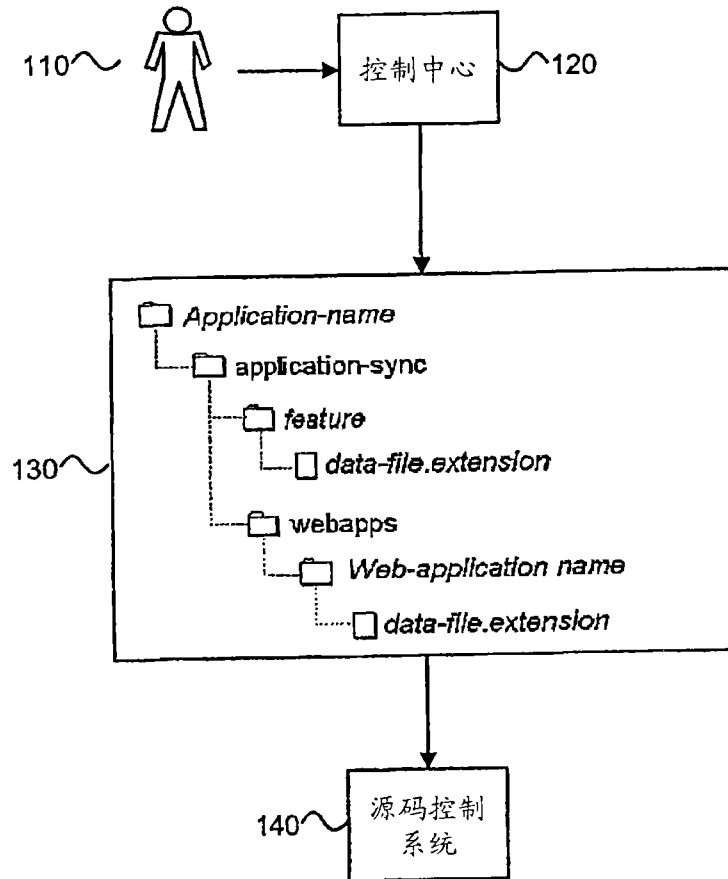


图 1

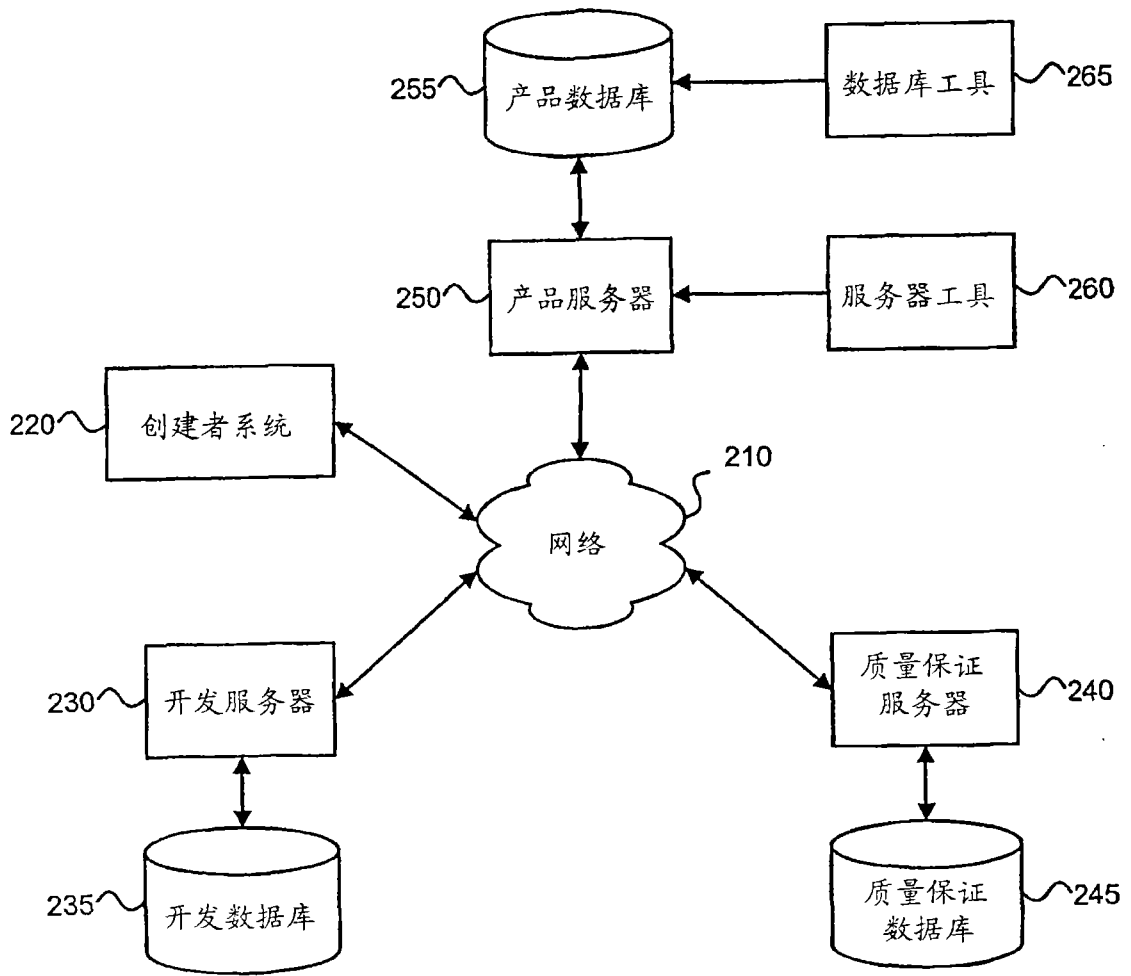


图 2

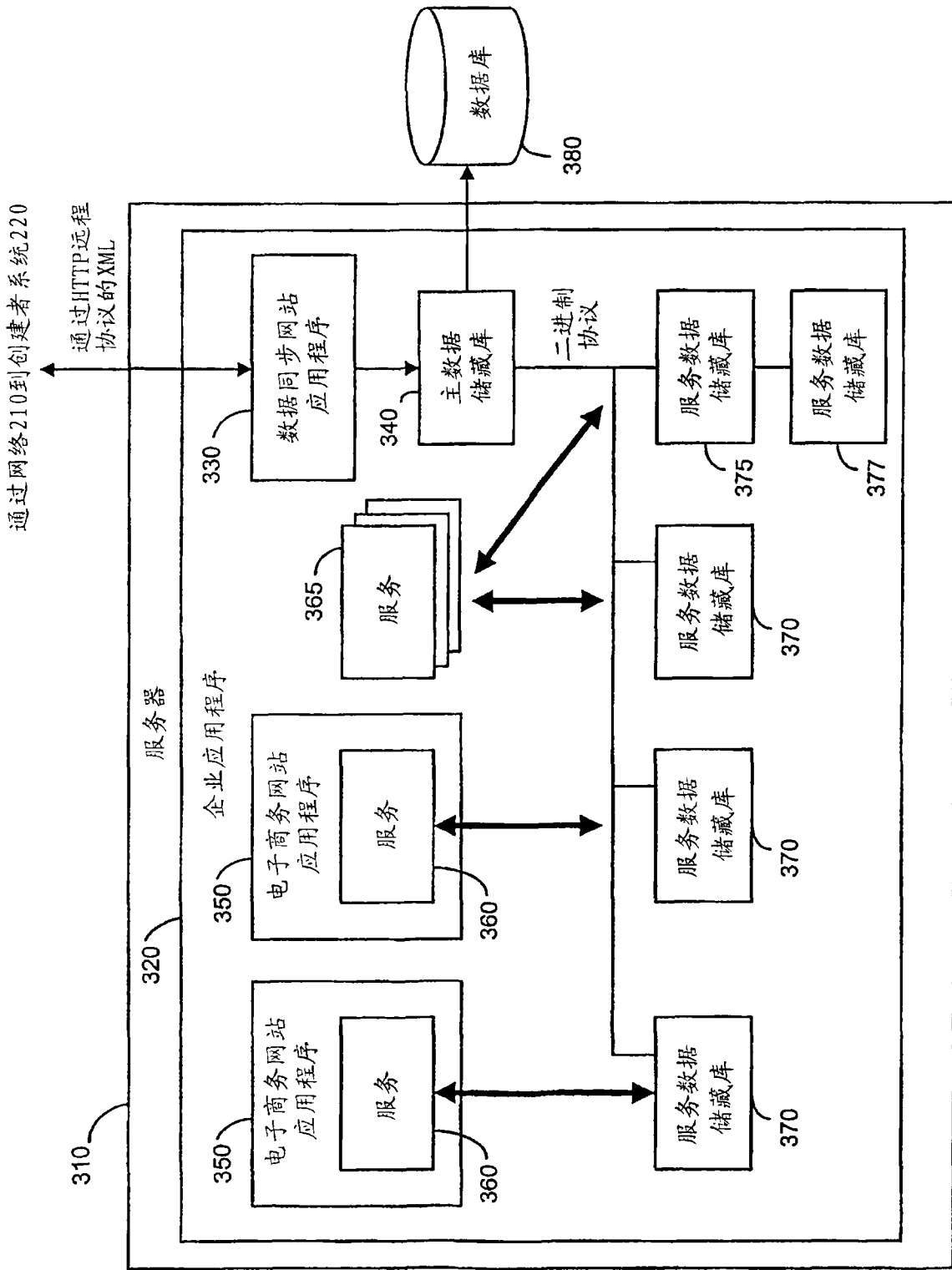


图 3

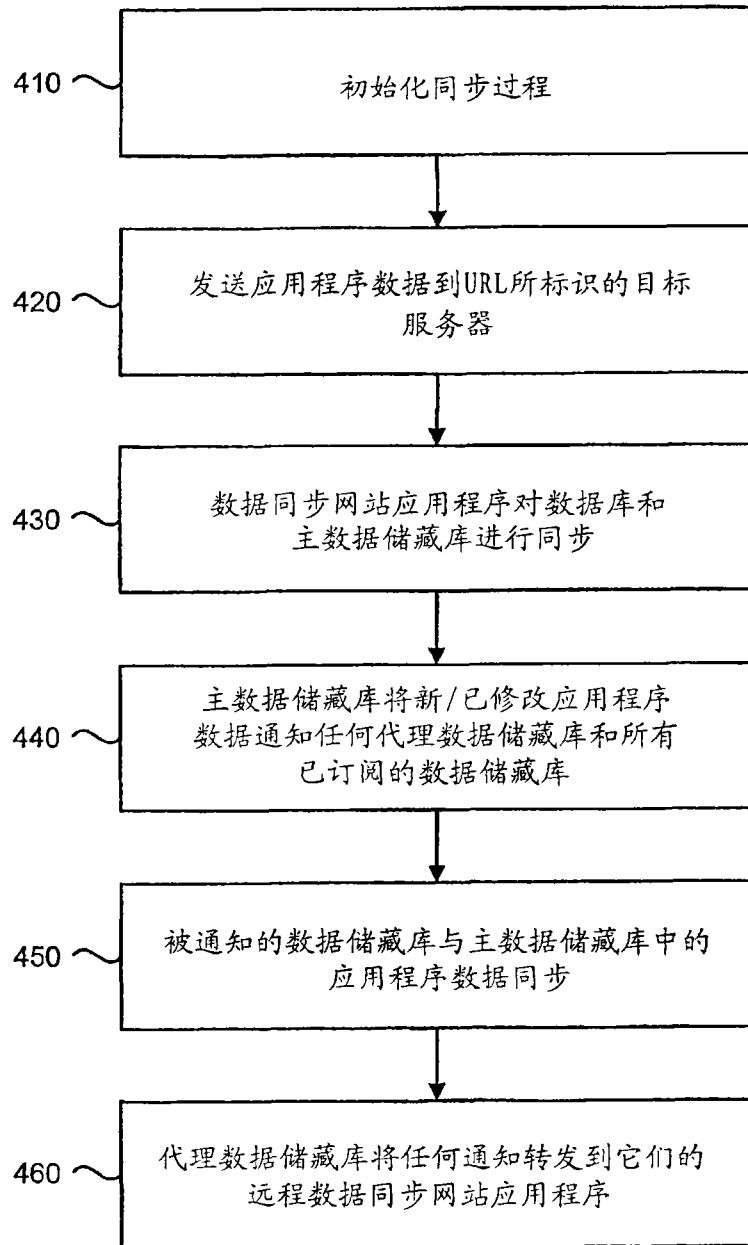


图 4

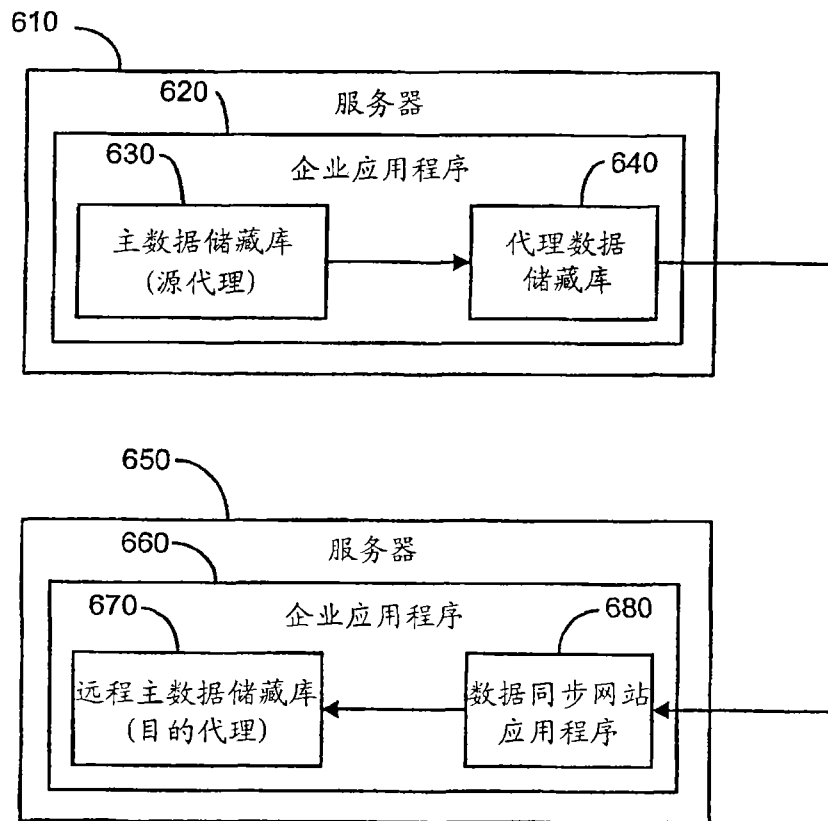


图 5

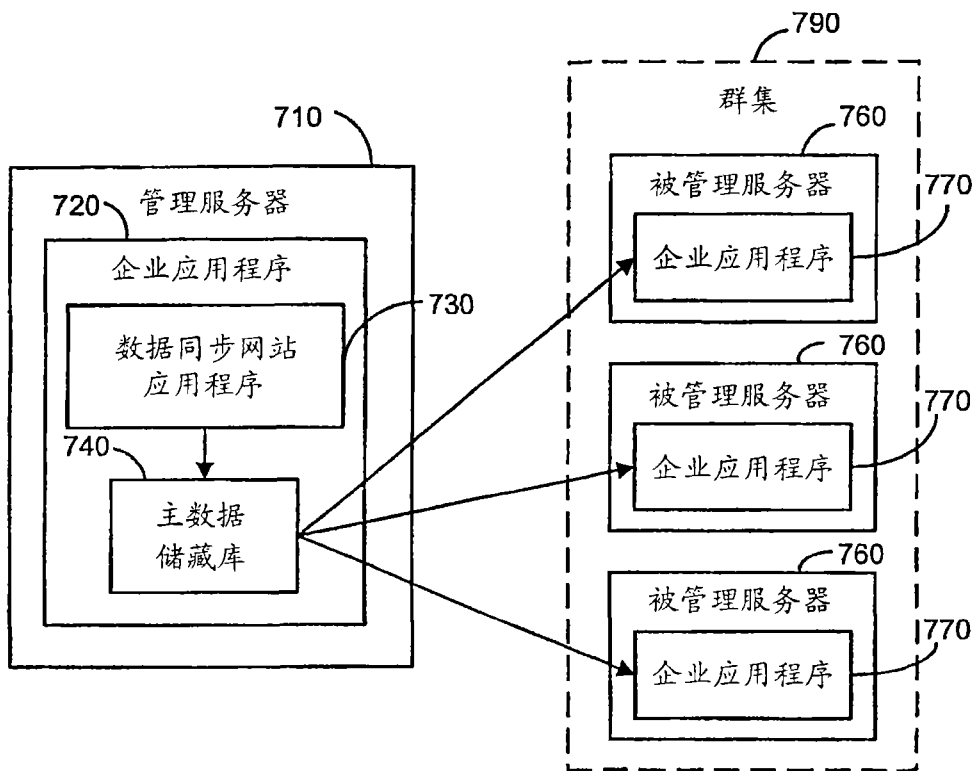


图 6