



[12] 发明专利说明书

专利号 ZL 200580023470.9

[45] 授权公告日 2009 年 10 月 7 日

[11] 授权公告号 CN 100547545C

[22] 申请日 2005.6.8

[21] 申请号 200580023470.9

[30] 优先权

[32] 2004.7.13 [33] US [31] 10/889,777

[86] 国际申请 PCT/EP2005/052632 2005.6.8

[87] 国际公布 WO2006/005655 英 2006.1.19

[85] 进入国家阶段日期 2007.1.12

[73] 专利权人 国际商业机器公司

地址 美国纽约

[72] 发明人 A·波洛佐夫

[56] 参考文献

CN1432248A 2003.7.23

US2003/0154239A1 2003.8.14

US6011918A 2000.1.4

WO95/04968A1 1995.2.16

A bytecode translator for distributed execution of "legacy" Java software. TATSUBORI M ET AL. ECOOP 2001. OBJECT. ORIENTED PROGRAMMING. 15TH EUROPEAN CONFERENCE., Vol. 2072 . 2001

审查员 欧阳琦

[74] 专利代理机构 北京市中咨律师事务所

代理人 于 静 李 峰

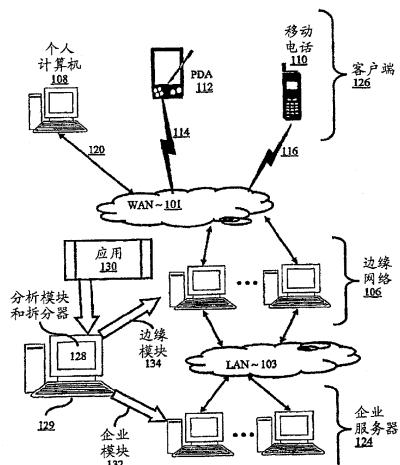
权利要求书 3 页 说明书 17 页 附图 11 页

[54] 发明名称

用于网络边缘计算的应用拆分的方法和系统

[57] 摘要

用于网络边缘计算的应用拆分包括：识别应用拆分点；识别针对所述应用拆分点的拆分类型；根据所述拆分类型将所述应用拆分成子部件；以及将子部件移动到边缘网络。识别应用拆分点可包括识别调用 EJB 的应用的类。识别应用拆分点可包括识别更新企业数据库的应用的类。识别应用拆分点可包括识别调用 JCA 功能的应用的类。识别针对所述应用拆分点的拆分类型可包括识别针对所述应用拆分点的思洛拆分类型。识别针对所述应用拆分点的拆分类型可包括识别针对所述应用拆分点的 EJB 拆分类型。



1. 一种用于网络边缘计算的应用拆分的方法，所述方法包括：

识别应用拆分点；

识别针对所述应用拆分点的拆分类型，包括：识别针对所述应用拆分点的思洛拆分类型、EJB 拆分类型、或者正面拆分类型，其中所述思洛拆分类型包括：通过 JDBC 调用来更新企业数据库的类的应用拆分点、产生 HTTP 调用的类的应用拆分点、或潜在应用拆分点列表上的所有剩余未分配的类的应用拆分点，其中所述正面拆分类型包括：不止一次识别为潜在拆分点的类的应用拆分点；

根据所述拆分类型将所述应用拆分成子部件，包括：

检验将被包括在子部件中的每个类是可串行化的；

检验用于维护有状态数据的类的出现；以及

如果出现用于维护有状态数据的任意类，则插入胶合类和对胶合类的调用以确保所述有状态数据在需要时同时在边缘模块及其相关企业模块中成为可用，其中所述胶合类用于从所述边缘模块向所述其相关企业模块传送所述有状态数据；以及

将子部件移动到边缘网络。

2. 根据权利要求 1 所述的方法，其中，识别应用拆分点还包括识别调用 EJB 的应用的类。

3. 根据权利要求 1 所述的方法，其中，识别应用拆分点还包括识别更新企业数据库的应用的类。

4. 根据权利要求 1 所述的方法，其中，识别应用拆分点还包括识别调用 JCA 功能的应用的类。

5. 根据权利要求 1 所述的方法，其中：所述应用拆分点包括调用 EJB 的调用分级中的类；所述拆分类型包括 EJB 拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的类拆分成客户端 RMI 模块和服务器 RMI 模块；以及，将子部件移动到边缘网络还包括将

所述客户端 RMI 模块移动到所述边缘网络。

6. 根据权利要求 1 所述的方法，其中：所述应用拆分点包括调用 EJB 的调用分级中的类；所述拆分类型包括 EJB 拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的类拆分成客户端 web 服务模块和服务器 web 服务模块；以及，将子部件移动到边缘网络还包括将所述客户端 web 服务模块移动到所述边缘网络。

7. 根据权利要求 1 所述的方法，其中：所述应用拆分点包括更新企业数据库的调用分级中的类；所述拆分类型包括思洛拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的所有类作为一个子部件聚合成思洛模块；以及将子部件移动到边缘网络还包括将所述思洛模块移动到所述边缘网络。

8. 根据权利要求 1 所述的方法，其中：所述应用拆分点包括调用 EJB 和调用 HTTP 功能的调用分级中的类；所述拆分类型包括思洛拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的所有类作为一个子部件聚合到思洛模块中；以及将子部件移动到边缘网络还包括将所述思洛模块移动到所述边缘网络。

9. 根据权利要求 1 所述的方法，其中：所述应用拆分点包括调用 JCA 功能和调用 HTTP 功能的调用分级中的类；所述拆分类型包括思洛拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的所有类作为一个子部件聚合到思洛模块中；以及将子部件移动到边缘网络还包括将所述思洛模块移动到所述边缘网络。

10. 根据权利要求 1 所述的方法，其中：所述应用拆分点包括被不止一次列为潜在拆分点的调用分级中的类；所述拆分类型包括正面拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的类拆分成客户端 web 服务模块和服务器 web 服务模块；以及，将子部件移动到边缘网络还包括将所述客户端 web 服务模块移动到所述边缘网络。

11. 一种用于网络边缘计算的应用拆分的系统，所述系统包括：
用于识别应用拆分点的装置；

用于识别针对所述应用拆分点的拆分类型的装置，该装置还包括：用于识别针对所述应用拆分点的思洛拆分类型的装置、用于识别针对所述应用拆分点的 EJB 拆分类型的装置、用于识别针对所述应用拆分点的正面拆分类型的装置，其中所述思洛拆分类型包括：通过 JDBC 调用来更新企业数据库的类的应用拆分点、产生 HTTP 调用的类的应用拆分点、或潜在应用拆分点列表上的所有剩余未分配的类的应用拆分点，其中所述正面拆分类型包括：不止一次识别为潜在拆分点的类的应用拆分点；

用于根据所述拆分类型将所述应用拆分成子部件的装置，包括：

检验将被包括在子部件中的每个类是可串行化的装置；

检验用于维护有状态数据的类的出现的装置，其中所述有状态数据可用于在多个子部件中被维护；以及

如果出现用于维护有状态数据的任意类，则插入胶合类和对胶合类的调用以确保所述有状态数据在需要时成为可用的装置，其中所述有状态数据可用于在多个子部件中被维护；以及

用于将子部件移动到边缘网络的装置。

12. 根据权利要求 11 所述的系统，其中，用于识别应用拆分点的装置还包括用于识别调用 EJB 的应用的类的装置。

13. 根据权利要求 11 所述的系统，其中，用于识别应用拆分点的装置还包括用于识别更新企业数据库的应用的类的装置。

14. 根据权利要求 11 所述的系统，其中，用于识别应用拆分点的装置还包括用于识别调用 JCA 功能的应用的类的装置。

15. 根据权利要求 11 所述的系统，其中：所述应用拆分点包括调用 EJB 的调用分级中的类；所述拆分类型包括 EJB 拆分类型；用于根据所述拆分类型将所述应用拆分成子部件的装置还包括用于将所述调用分级中的类拆分成客户端 RMI 模块和服务器 RMI 模块的装置；以及用于将子部件移动到边缘网络的装置还包括用于将所述客户端 RMI 模块移动到所述边缘网络的装置。

用于网络边缘计算的应用拆分的方法和系统

技术领域

本发明的领域是数据处理，或者更具体地是用于网络边缘计算的应用拆分的方法、系统和产品。

背景技术

因特网和万维网主要以客户端 - 服务器模型运转用于数据通信。被传输的内容越来越复杂，并且客户端越来越多样化。高效地将复杂、多变的内容递送到越来越多样化的客户端设备正变得越来越困难。所述增长的需求导致越来越多用于处理每个新类型的数据的新设备、新服务器、新客户端。

在概念性客户端 - 服务器模型中，所述客户端从所述服务器请求信息，并且所述服务器以所述客户端所请求的格式发回信息。由于需要仅将注意力集中到在“终端”的处理，所以该内容递送系统曾经被认为是理想的。然而，现实已要求更多，并且响应于此，新的服务已出现。在所述客户端附近和边缘递送点的高速缓存使运转更快，并且减轻其网络负载。网关（例如用于无线设备的那些）帮助多组客户端进行连接。内容递送网络（CDN）更好地组织了用于所述客户端的因特网内容的庞大阵列。所有这些服务已相互独立地增长，然而，其使得扩展和新服务对于销售商特定和难以管理。随着多样化客户端设备的演进，内容提供商被迫以针对每种设备、个人数字助理（PDA）、个人计算机（PC）、蜂窝式电话、膝上型计算机、电子书阅读器和其它设备的每种必要格式提供数据。服务器被迫处理可用格式和所述客户端可以接受的格式之间的区别。

网络边缘体系结构提供一种用于解决由于“网络内”服务而已增长的

复杂度的方式。术语“网络边缘”通常用于描述因特网中的实体的物理或网络边界。例如，网络边缘可以是企业网络的内联网或者因特网服务提供商（“ISP”）的接入点的集合。“网络边缘”可以还指内容递送的区分点，例如国家的边界，其中，在该点将数据翻译成合适的语言或格式是有益的。所述网络边缘是内容递送路径上内容处理可能发生的点。

所述网络边缘体系结构是用于执行内容递送过程的一部分的技术的聚集，其中，所述内容递送过程用于在客户端和服务器之间传输数据。其体系结构是开发用于执行内容递送功能的软件和硬件的统一方案。网络边缘体系结构允许例如数据代码转换的操作。内容提供商可以不管客户端设备的类型而以单一、标准格式提供数据，并且网络边缘体系结构使得提供适合所述设备的用于呈现的格式更容易。例如，内容提供商可以不管可请求该内容的客户端设备的类型而以单一格式存储内容，并且所述网络边缘技术可以针对所述接收设备重新格式化所述内容。运行在边缘服务器或边缘网络上的基于规则的服务基于针对各种设备的标准化协议来进行适当的数据转换。

考虑电影递送的例子。通过使用网络边缘技术，消费者以适合于其设备和其个人品味的格式观看电影。内容提供商不再必须担心将观看电影的设备，因为数据转换的负担从内容提供商和服务器处移除。作为代替，数据转换被合并到所述网络边缘。

在传统客户端/服务器体系结构中，用户信息和客户端设备接口必须在所述客户端设备或所述内容服务器上进行管理。即，所述用户被要求重新输入数据，并且所述服务器被要求处理客户端多样性问题，而不论所述客户端设备是蜂窝式电话、PDA、膝上型计算机、个人计算机等。

从提供商到客户端的内容传输和重新组成以几个步骤发生。客户端的请求根据所述服务器中保留的已知信息加上从所述客户端输入的另外所需的数据进行修正，即个性化。例如，所述客户端可能正在使用蜂窝式电话来接入万维网或电子邮件服务。这要求所述服务器发回仅适合该蜂窝式电话接口的内容，并且通过语音菜单或键盘菜单进行处理以获得用户特性。

在所述网络边缘体系结构中，边缘服务器修正所述客户端的请求以指定所述合适的蜂窝式电话接口。然后，所述边缘服务器与内容服务器通信以填入用户数据，由此简化了用户交互。然后，所述边缘服务器修正去往所述内容服务器的请求，用以获得随后被所述边缘服务器重新组合以适合所述客户端设备的数据。所述合适的内容然后以适合所述客户端设备的方式被下载到所述客户端设备中。

所述网络边缘服务器驻留于所述客户端和所述内容服务器之间，并且用作同时所述客户和所述内容提供商的代理。当所述客户端发出对于内容的请求时，网络边缘服务器典型地基于一组预定义的规则根据所述客户的信息修正所述请求。所述边缘服务器然后将所述修正的请求发送到所述内容服务器，其中，所述发送可能具有如之前所指出的到所述内容服务器的某种交换。所述内容服务器被解除了修正所述请求和重新组合所述内容的负担。所需要的内容的任意转换将在使用标准协议的网络边缘服务器处发生。通过该方式，网络边缘体系结构简化了所述内容提供商所要求的用于解决不同设备所需要的不同格式的信息。

网络边缘体系结构表示从传统客户端 - 服务器模型向更高效的客户端 - 边缘 - 服务器模型的不断增长的转移。当内容级服务在所述网络边缘被执行时，内容提供商可以提供个性化的、可扩展的服务。通过使用网络边缘体系结构，所述边缘起到当前被强制在客户端和服务器二者上的作用。在传统的客户端 - 服务器体系结构中被完全安装在内容服务器或应用服务器上的应用软件现在可以被部署到所述边缘网络。另外，在传统客户端 - 服务器体系结构中被完全安装在内容服务器或应用服务器上的应用软件现在可以在所述内容服务器和所述边缘之间被拆分，并且在许多情况下必须被拆分。使用自动化工具来识别应用应当怎样被准备部署到所述边缘以及在必要时怎样拆分所述应用将是有益的。

发明内容

用于网络边缘计算的应用拆分的方法、系统和计算机程序产品被提供。

实施例包括：识别应用拆分点；识别针对所述应用拆分点的拆分类型；根据所述拆分类型将所述应用拆分成子部件；以及将子部件移动到边缘网络。许多实施例包括确定考虑到其管理的数据的情况下应用太大而不适合所述边缘网络。

在典型实施例中，拆分所述应用包括取决于用户偏好拆分所述应用。在许多实施例中，识别应用拆分点包括识别调用 EJB 的应用的类。在许多实施例中，识别应用拆分点包括识别更新企业数据库的应用的类。在许多实施例中，识别应用拆分点包括识别调用 JCA 功能的应用的类。

在许多实施例中，识别针对所述应用拆分点的拆分类型包括识别针对所述应用拆分点的思洛（silo）拆分类型。在典型实施例中，识别针对所述应用拆分点的拆分类型包括识别针对所述应用拆分点的 EJB 拆分类型。在许多实施例中，识别针对所述应用拆分点的拆分类型包括识别针对所述应用拆分点的正面（facade）拆分类型。

在一些实施例中，所述应用拆分点包括调用 EJB 的调用分级中的类；并且所述拆分类型包括 EJB 拆分类型。在一些所述实施例中，根据所述拆分类型将所述应用拆分成子部件包括将所述调用分级中的类拆分成客户端 RMI 模块和服务器 RMI 模块；以及将子部件移动到边缘网络包括将所述客户端 RMI 模块移动到所述边缘网络。

在一些实施例中，所述应用拆分点包括调用 EJB 的调用分级中的类，并且所述拆分类型包括 EJB 拆分类型。在一些所述实施例中，根据所述拆分类型将所述应用拆分成子部件包括将所述调用分级中的类拆分成客户端 web 服务模块和服务器 web 服务模块；以及将子部件移动到边缘网络包括将所述客户端 web 服务模块移动到所述边缘网络。

在一些实施例中，所述应用拆分点包括更新企业数据库的调用分级中的类，并且所述拆分类型包括思洛拆分类型。在一些所述实施例中，根据所述拆分类型将所述应用拆分成子部件包括将所述调用分级中的所有类作为一个子部件聚合到思洛模块中；以及，将子部件移动到边缘网络包括将所述思洛模块移动到所述边缘网络。

在一些实施例中，所述应用拆分点包括调用 EJB 和调用 HTTP 功能的调用分级中的类，并且所述拆分类型包括思洛拆分类型。在一些所述实施例中，根据所述拆分类型将所述应用拆分成子部件包括将所述调用分级中的所有类作为一个子部件聚合到思洛模块中；以及将子部件移动到边缘网络包括将所述思洛模块移动到所述边缘网络。

在一些实施例中，所述应用拆分点包括调用 JCA 功能和调用 HTTP 功能的调用分级中的类，并且所述拆分类型包括思洛拆分类型。在一些所述实施例中，根据所述拆分类型将所述应用拆分成子部件包括将所述调用分级中的所有类作为一个子部件聚合到思洛模块中；以及将子部件移动到边缘网络包括将所述思洛模块移动到所述边缘网络。

在一些实施例中，所述应用拆分点包括被不止一次列为潜在拆分点的调用分级中的类，并且所述拆分类型包括正面拆分类型。在一些所述实施例中，根据所述拆分类型将所述应用拆分成子部件包括将所述调用分级中的类拆分成客户端 web 服务模块和服务器 web 服务模块；以及将子部件移动到边缘网络包括将所述客户端 web 服务模块移动到所述边缘网络。

从本发明的第一方面来看，其提供一种用于网络边缘计算的应用拆分的方法，所述方法包括：识别应用拆分点；识别针对所述应用拆分点的拆分类型；根据所述拆分类型将所述应用拆分成子部件；以及将子部件移动到边缘网络。

优选地，本发明提供一种方法，其中，识别应用拆分点还包括识别调用 EJB 的应用的类。

优选地，本发明提供一种方法，其中，识别应用拆分点还包括识别更新企业数据库的应用的类。

优选地，本发明提供一种方法，其中，识别应用拆分点还包括识别调用 JCA 功能的应用的类。

优选地，本发明提供一种方法，其中，识别针对所述应用拆分点的拆分类型还包括识别针对所述应用拆分点的思洛拆分类型。

优选地，本发明提供一种方法，其中，识别针对所述应用拆分点的拆

分类型还包括识别针对所述应用拆分点的 EJB 拆分类型。

优选地，本发明提供一种方法，其中，识别针对所述应用拆分点的拆分类型还包括识别针对所述应用拆分点的正面拆分类型。

优选地，本发明提供一种方法，其中：所述应用拆分点包括调用 EJB 的调用分级中的类；所述拆分类型包括 EJB 拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的类拆分成客户端 RMI 模块和服务器 RMI 模块；以及将子部件移动到边缘网络还包括将所述客户端 RMI 模块移动到所述边缘网络。

优选地，本发明提供一种方法，其中：所述应用拆分点包括调用 EJB 的调用分级中的类；所述拆分类型包括 EJB 拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的类拆分成客户端 web 服务模块和服务器 web 服务模块；以及将子部件移动到边缘网络还包括将所述客户端 web 服务模块移动到所述边缘网络。

优选地，本发明提供一种方法，其中：所述应用拆分点包括更新企业数据库的调用分级中的类；所述拆分类型包括思洛拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的所有类作为一个子部件聚合成思洛模块；以及将子部件移动到边缘网络还包括将所述思洛模块移动到所述边缘网络。

优选地，本发明提供一种方法，其中：所述应用拆分点包括调用 EJB 和调用 HTTP 功能的调用分级中的类；所述拆分类型包括思洛拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的所有类作为一个子部件聚合成思洛模块；以及将子部件移动到边缘网络还包括将所述思洛模块移动到所述边缘网络。

优选地，本发明提供一种方法，其中：所述应用拆分点包括调用 JCA 功能和调用 HTTP 功能的调用分级中的类；所述拆分类型包括思洛拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的所有类作为一个子部件汇聚成思洛模块；以及将子部件移动到边缘网络还包括将所述思洛模块移动到所述边缘网络。

优选地，本发明提供了一种方法，其中：所述应用拆分点包括被不止一次列为潜在拆分点的调用分级中的类；所述拆分类型包括正面拆分类型；根据所述拆分类型将所述应用拆分成子部件还包括将所述调用分级中的类拆分成客户端 web 服务模块和服务器 web 服务模块；以及将子部件移动到边缘网络还包括将所述客户端 web 服务模块移动到所述边缘网络。

从本发明的第二方面来看，其提供了一种用于网络边缘计算的应用拆分的系统，所述系统包括：用于识别应用拆分点的装置；用于识别针对所述应用拆分点的拆分类型的装置；用于根据所述拆分类型将所述应用拆分成子部件的装置；以及用于将子部件移动到边缘网络的装置。

优选地，本发明提供一种系统，其中，用于识别应用拆分点的装置还包括用于识别调用 EJB 的应用的类的装置。

优选地，本发明提供一种系统，其中，用于识别应用拆分点的装置还包括用于识别更新企业数据库的应用的类的装置。

优选地，本发明提供一种系统，其中，用于识别应用拆分点的装置还包括用于识别调用 JCA 功能的应用的类的装置。

优选地，本发明提供一种系统，其中，用于识别针对所述应用拆分点的拆分类型的装置还包括用于识别针对所述应用拆分点的思洛拆分类型的装置。

优选地，本发明提供一种系统，其中，用于识别针对所述应用拆分点的拆分类型的装置还包括用于识别针对所述应用拆分点的 EJB 拆分类型的装置。

优选地，本发明提供一种系统，其中，用于识别针对所述应用拆分点的拆分类型的装置还包括用于识别针对所述应用拆分点的正面拆分类型的装置。

优选地，本发明提供一种系统，其中：所述应用拆分点包括调用 EJB 的调用分级中的类；所述拆分类型包括 EJB 拆分类型；用于根据所述拆分类型将所述应用拆分成子部件的装置还包括用于将所述调用分级中的类拆分成客户端 RMI 模块和服务器 RMI 模块的装置；以及用于将子部件移动

到边缘网络的类还包括将所述客户端 RMI 模块移动到所述边缘网络的装置。

从本发明的第三方面来看，其提供了一种可载入到数字计算机的内部存储器中的计算机程序产品，其中，所述计算机程序产品包括当所述产品运行在计算机上时用于执行以实现如上面所描述的本发明的软件代码部分。

附图说明

下面仅借助于例子并参考附图详细描述了本发明的实施例，在附图中：

图 1 阐明了根据本发明的实施例的用于网络边缘计算的应用拆分的示例性系统的系统图；

图 2 阐明了示出示例性软件开发系统的框图；

图 3 阐明了示出用于网络边缘计算的应用拆分的示例性方法的流程图；

图 4 阐明了示出用于识别应用拆分点的示例性方法的流程图；

图 5 阐明了示出用于识别针对应用拆分点的拆分类型的示例性方法的流程图；

图 6 阐明了示出针对 EJB 拆分类型的用于网络边缘计算的应用拆分的示例性方法的流程图；

图 7 阐明了示出针对 EJB 拆分类型的用于网络边缘计算的应用拆分的另一示例性方法的流程图；

图 8 阐明了示出针对思洛 (silo) 拆分类型的用于网络边缘计算的应用拆分的示例性方法的流程图；

图 9 阐明了示出针对思洛拆分类型的用于网络边缘计算的应用拆分的另一示例性方法的流程图；

图 10 阐明了示出针对思洛拆分类型的用于网络边缘计算的应用拆分的另一示例性方法的流程图；以及

图 11 阐明了示出正面拆分类型的用于网络边缘的应用拆分的示例性

方法的流程图。

具体实施方式

本发明在本说明书中很大程度上是就用于网络边缘计算的应用拆分的方法这一方面进行描述。然而，本领域的技术人员将认识到，包括用于根据所述公开的方法运转的合适编程装置的任意计算机系统也落在本发明的范围内。合适的编程装置包括用于指挥计算机系统执行本发明的方法的步骤的任意装置，其包括例如：包括耦合到计算机存储器的处理单元和算术-逻辑电路的系统，其中，所述系统具有在计算机存储器中进行存储的能力，所述计算机存储器包括被配置为存储数据和程序指令的电子电路，其中，所述程序指令是用于由处理单元执行的、被编程的本发明的方法的步骤。

本发明还可以体现为一种计算机程序产品，例如磁盘或其它记录媒体，其用于与任意合适的数据处理系统一起使用。计算机程序产品的实施例可以通过使用用于机器可读信息的任意记录媒体被实现，其中，所述记录媒体包括磁性媒体、光学媒体或其它合适的媒体。本领域的技术人员将立即认识到，具有合适的编程装置的任意计算机系统将能够执行如体现为程序产品的本发明的方法的步骤。本领域的技术人员将立即认识到，尽管本说明书中所描述的示例性实施例的多数是面向被安装在计算机硬件上并在其上执行的软件，然而，作为固件或作为硬件被实现的可选实施例也理当在本发明的范围内。

用于网络边缘计算的应用拆分

从图 1 开始参考附图对用于网络边缘计算的应用拆分的方法、系统和产品进行描述。图 1 阐明了根据本发明的实施例的用于网络边缘计算的应用拆分的示例性系统的系统框图。图 1 的系统包括通过广域网 (101) 连接到被配置作为边缘网络 (106) 的几个边缘服务器的几个示例性客户端设备 (126)。所述边缘服务器又通过局域网 (103) 连接到代表企业服务器 (124) 的几个应用和内容服务器。

图 1 的例子中的所述客户端 (126) 包括：通过有线连接 (120) 连接到广域网 (101) 的个人计算机 (108)、通过无线连接 (114) 连接到广域网 (101) 的无线的、使能网络的个人数字助理 (“PDA”) (112)，以及通过无线连接 (116) 连接到广域网 (101) 的移动电话 (110)。

图 1 的所述系统包括：正在开发系统 (129) 上被开发的用于拆分成两个模块的应用 (130)、用于安装到所述边缘网络 (106) 上的边缘模块 134 和用于安装到所述企业网络 (124) 上的企业模块 (132)。开发系统 (129) 是这样的计算机，所述计算机具有例如 Java_{TM} 软件开发工具包 (“SDK”) 的软件开发环境，并且还具有分析模块和拆分器 (128)，其一般通过识别应用拆分点、识别针对所述应用拆分点的拆分类型、根据所述拆分类型将应用拆分成子部件以及将某些子部件移动到边缘网络来运转。

为进一步说明，图 2 阐明了示出这样的示例性软件开发系统的框图，所述系统一般通过识别应用拆分点、识别针对所述应用拆分点的拆分类型、根据所述拆分类型将应用拆分成子部件以及将某些子部件移动到边缘网络来运转。图 2 的所述示例性系统包括 JDBC_{TM} 分析模块 (202)，该模块一般运转为将应用拆分点识别为调用 Java 数据库连接 (“JDBC”) 功能以更新企业数据库的类。对 JDBC 功能的所述调用可以例如表示为对被命名为 ‘executeUpdate’ 或 ‘executeInsert’ 的 Java 成员方法的调用。

图 2 的所述示例性系统还包括 EJB_{TM} 分析模块 (204)，该模块一般运转为将应用拆分点识别为调用企业 Java 柄 (“EJB”) 的类。调用 EJB 的类可以被识别为例如使用对 EJB Properties() 构造器、InitialContext() 构造器等的调用以构造会话 EJB 的类。

图 2 的示例性系统还包括 JCA_{TM} 分析模块 (206)，该模块一般运转为将应用拆分点识别为调用 Java 构造器体系结构 (“JCA”) 功能的类。调用 JCA 功能的类可以被识别为例如使用对 JCA ConnectionFactory()、createInteraction() 方法、Interaction.execute() 方法等的调用的类。

在图 2 的例子中，分析模块 (202、204 和 206) 将拆分点识别为潜在的拆分点的列表。针对每个如此识别的拆分点，拆分类型分析模块 (208)

识别出拆分类型。所述拆分类型分析模块一般运转为针对作为调用 EJB 功能的类的拆分点识别 EJB 拆分类型。所述拆分类型分析模块运转为针对作为通过例如 JDBC 调用来更新企业数据库的类的拆分点识别思洛 (Silo) 拆分类型。所述拆分类型分析模块运转为针对作为产生 HTTP 调用的类的拆分点识别思洛拆分类型。所述拆分类型分析模块运转为针对作为被所述分析模块 (202、204 和 206) 不止一次识别为潜在拆分点的类的拆分点识别正面 (facade) 拆分类型。

图 2 的示例性系统包括拆分器 (210)，其是一般通过在所识别的拆分点根据针对每个拆分点的拆分类型拆分所述应用来实现创建所述应用 (130) 的子部件的工作的软件模块。对于 EJB 拆分类型，拆分器 (210) 可以将应用 (130) 拆分成客户端 RMI_{TM} (“Java 远程方法调用”) 模块和服务器 RMI 模块，将所述客户端 RMI 模块移动到边缘网络作为 EJB 边缘模块 (216)，并且将所述服务器 RMI 模块移动到企业网络作为 EJB 企业模块 (218)。可选地，对于 EJB 拆分类型，拆分器 (210) 可以将应用 (130) 拆分成客户端 web 服务模块和服务器 web 服务模块，将所述客户端 web 服务模块移动到边缘网络作为 EJB 边缘模块 (216)，并且将所述服务器 web 服务模块移动到企业网络作为 EJB 企业模块 (218)。类似地，对于正面拆分类型，拆分器 (210) 可以将应用 (130) 拆分成客户端 web 服务模块和服务器 web 服务模块，将所述客户端 web 服务模块移动到边缘网络作为正面边缘模块 (212)，并且将所述服务器 web 服务模块移动到企业网络作为正面企业模块 (214)。对于所述思洛拆分类型，拆分器 (210) 可以将所述调用分级中的所有类作为一个子部件汇聚到思洛模块 (220) 中。所述拆分器 (220) 可以将所述思洛模块 (220) 移动到边缘网络或移动到企业网络。

为进一步说明，图 3 阐明了示出用于网络边缘计算的应用拆分的示例性方法的流程图，其中，所述方法包括识别 (302) 应用拆分点 (310)。图 4 阐明了示出用于识别应用拆分点的示例性方法的流程图。在图 4 的例子中，识别应用拆分点通过识别 (402) 调用 EJB 的应用的类来实现。如

上面所指出的，调用 EJB 的类可以通过其对 EJB 成员方法的使用被识别，其中，所述对 EJB 成员方法的使用例如是对 EJB Properties()构造器、InitialContext()构造器等的调用。在图 4 的所述方法中，识别应用拆分点还可以通过识别 (404) 通过例如对例如 ‘executeUpdate’ 或 ‘executeInsert’ 的 JDBC 方法的调用来更新企业数据库的应用的类来实现。在图 4 的所述方法中，识别应用拆分点可以通过识别 (405) 调用 JCA 功能的应用的类来实现。调用 JCA 功能的类可以被识别为例如使用对 JCA ConnectionFactory()、createInteraction()方法、Interaction.execute()方法等的调用的类。

再次参考图 3：图 3 的所述方法还包括识别 (304) 针对所述应用拆分点的拆分类型 (312)。图 5 阐明了示出用于识别针对应用拆分点的拆分类型的示例性方法的流程图。在图 5 的所述例子中，识别针对所述应用拆分点的拆分类型包括识别针对所述应用拆分点的思洛拆分类型 (504)。在图 5 的所述例子中，识别针对所述应用拆分点的思洛拆分类型 (504) 可以通过从潜在拆分点 (406) 的列表和所述应用 (130) 中识别调用 HTTP 功能的所述应用中的类并且记录 (506) 所述被识别的类的调用分级来实现。HTTP 功能的例子包括 HttpServletRequest 类、HttpServletResponse 类和 HttpSession 类中的 Java 成员方法。

在图 5 的所述例子中，识别思洛拆分类型还可以通过从潜在拆分点 (406) 的列表和所述应用 (130) 中识别更新企业数据库的类来实现。更新企业数据库的类可以例如通过对例如 ‘executeUpdate’ 和 ‘executeInsert’ 的 JDBC 成员方法的调用被识别。在图 5 的所述方法中，识别针对所述应用拆分点的拆分类型包括识别 (508) 针对所述应用拆分点的 EJB 拆分类型 (510)。在图 5 的所述方法中，识别 EJB 拆分类型可以通过从潜在拆分点 (406) 的列表和从所述应用 (130) 源代码中识别 (508) 调用 EJB 的类来实现，其中，所述调用 EJB 的类例如由对 EJB Properties()构造器、InitialContext()构造器等的调用来指示。

在图 5 的所述方法中，识别针对所述应用拆分点的拆分类型包括识别

针对所述应用拆分点的正面拆分类型（514）。在图 5 的所述方法中，识别针对所述应用拆分点的正面拆分类型（514）可以通过从潜在拆分点的列表中识别（512）被不止一次列为潜在拆分点的类来实现。

图 5 的所述方法还包括将所述潜在拆分点列表上的所有剩余未分配的类分配（518）到所述思洛拆分类型（504）。即，在贯穿被列为潜在拆分点的类中扫描例如图 2 中标号（208）处所示的拆分类型分析模块的过程中，将所述拆分类型‘思洛’分配给还没被识别为 EJB 拆分类型或正面拆分类型的每个拆分点。

根据本发明的一些实施例实现了所有三种拆分：思洛拆分、正面拆分和 EJB 拆分。在所述实施例中，调用 HTTP 功能或更新企业数据部并且还调用 EJB 或被不止一次列为潜在拆分点的拆分点可以被归为思洛拆分类型和 EJB 拆分类型或正面拆分类型。即，一个拆分点可以被分配多于一个拆分类型。为避免所述拆分类型之间的冲突，所述实施例可以有利地允许每个拆分点仅一个拆分类型，其通过从所述思洛拆分类型（504）中删除（516）还被识别为 EJB 拆分类型（510）或正面拆分类型（514）的任意拆分点来实现。

再次参考图 3：图 3 的所述方法还包括根据所述拆分类型将所述应用拆分（306）成子部件（314）以及将子部件移动（308）到边缘网络（106）。对于 EJB 拆分类型，图 3 的所述方法包括将所述应用（130）拆分成客户端 RMI_{TM}（“Java 远程方法调用”）模块和服务器 RMI 模块，以及将所述客户端 RMI 模块移动到边缘网络和将所述服务器 RMI 模块移动到企业网络。可选地，对于所述 EJB 拆分类型，图 3 的所述方法包括将所述应用（130）拆分成客户端 web 服务模块和服务器 web 服务模块、将所述客户端 web 服务模块移动到边缘网络以及将所述服务器 web 服务模块移动到企业网络。类似地，对于正面拆分类型，图 3 的所述方法包括将所述应用（130）拆分成客户端 web 服务模块和服务器 web 服务模块、将所述客户端 web 服务模块移动到边缘网络以及将所述服务器 web 服务模块移动到企业网络。对于所述思洛拆分类型，图 3 的所述方法包括将所述调用分级中的所

有类作为一个子部件聚合到思洛模块中，以及将所述思洛模块移动到边缘网络或企业网络。

在图 3 的所述方法中，拆分(306)所述应用包括取决于用户偏好(303)拆分所述应用。所述用户偏好的例子包括：

- 不拆分 - 如果整个应用及其数据库将适合所述边缘网络
- 仅按照思洛拆分类型识别和拆分
- 仅按照正面拆分类型识别和拆分
- 仅按照 EJB 拆分类型识别和拆分
- 识别所有拆分类型，但仅按照思洛拆分
- 识别所有拆分类型，但仅按照正面拆分
- 识别所有拆分类型，但仅按照 EJB 拆分
- 识别所有拆分类型，并且按照优先权拆分：思洛/正面/EJB

所述第一示例性用户偏好详述为“不拆分 - 如果整个应用及其数据库将适合所述边缘网络”。在如果所述整个应用及其数据库将适合所述边缘网络的所述例子中，所述应用不被拆分，并且所述整个应用被移动到所述边缘网络。图 3 的所述方法因此还包括确定考虑到其管理的数据的情况下所述应用是否太大而不适合所述边缘网络。确定考虑到其管理的数据的情况下所述应用是否太大而不适合所述边缘网络的一种方法通过比较边缘网络的可用数据存储容量和所述应用的数据存储需求以及由所述应用管理的数据库中的数据来实现。

上面列出的所述用户偏好（“识别所有拆分类型，并且按照优先权拆分：思洛/正面/EJB”）是指：如果多于一个拆分类型出现，则所述应用取决于出现的拆分类型的优先权被拆分，在此情况下，思洛拆分类型优于正面拆分类型，并且正面拆分类型优于 EJB 拆分类型。即，如果所有拆分类型都出现，则仅按照思洛拆分。如果仅正面和 EJB 拆分类型出现，则仅按照正面拆分。如果仅 EJB 拆分类型出现，则按照 EJB 拆分。

根据所述拆分类型将所述应用拆分(306)成子部件(314)还有利地包括对将被包括在子部件中的每个类是可串行化的检验(未示出)。多数

Java 类是可串行化的一一除非，例如，成员方法例如在其调用参数中采用了例如字符或整数的本地数据类型。检验子部件中的每个类是可串行化因此可以通过扫描对本地数据类型的引用的应用源代码、报告对所述本地数据类型的发现以及将其转换成 Java 类和对 Java 对象的引用来实现。

另外，根据所述拆分类型将所述应用拆分（306）成子部件（314）还可以有利地包括胶合插入（glue insertion）。即，将被移动到边缘网络的子部件可以包括将跨多个请求/响应交换而被有状态地维护的数据。所述有状态数据可以在例如 Java HttpSession 对象中被维护。当应用功能在所述边缘网络和所述企业网络之间拆分时，所述有状态数据可以有益地同时在所述边缘服务器和所述企业服务器上被维护。用于从边缘模块向相应的企业模块传送所述有状态数据的软件被称为“胶合（glue）”。在本发明的典型实施例中，根据所述拆分类型将所述应用拆分（306）成子部件（314）还有利地包括：例如，检验例如包含 HttpSession 对象或对 HttpSession 对象的引用的类的维护所述有状态数据的类的出现，以及插入一个或更多胶合类或对胶合类的调用，以确保所述有状态数据在需要时同时在所述边缘模块及其相关企业模块中成为可用。

为进一步说明，图 6 阐明了示出针对所述 EJB 拆分类型的用于网络边缘计算的应用拆分的示例性方法的流程图。即，在图 6 的所述例子中，所述应用拆分点（310）被识别为调用 EJB 的调用分级中的类，并且所述拆分类型（312）被识别为 EJB 拆分类型。在图 6 的所述方法中，根据所述拆分类型将所述应用拆分（306）成子部件（314）通过将所述调用分级中的类拆分成客户端 RMI 模块（602）和服务器 RMI 模块（604）来实现。在图 6 的所述方法中，将子部件移动（308）到边缘网络通过将所述客户端 RMI 模块（602）移动到所述边缘网络（106）来实现。

为进一步说明，图 7 阐明了示出针对所述 EJB 拆分类型的用于网络边缘计算的应用拆分的另一示例性方法的流程图。即，所述应用拆分点（310）被识别为调用 EJB 的调用分级中的类，并且所述拆分类型（312）被识别为 EJB 拆分类型。在图 7 的所述方法中，根据所述拆分类型（312）将所

述应用 (130) 拆分 (306) 成子部件 (314) 通过将所述调用分级中的类拆分成客户端 web 服务模块 (702) 和服务器 web 服务模块 (704) 来实现。在图 7 的所述方法中，将子部件移动 (308) 到边缘网络 (106) 通过将所述客户端 web 服务 (702) 模块移动到所述边缘网络 (106) 来实现。

为进一步说明，图 8 阐明了示出针对所述思洛拆分类型的用于网络边缘的应用拆分的示例性方法的流程图。在图 8 的所述例子中，所述应用拆分点 (310) 被识别为更新企业数据库的调用分级中的类，并且所述拆分类型 (312) 被识别为思洛拆分类型。在图 8 的所述方法中，根据所述拆分类型 (312) 将所述应用 (130) 拆分 (306) 成子部件 (314) 通过将所述调用分级中的所有类作为一个子部件聚合到思洛模块 (802) 中来实现。在图 8 的所述方法中，将子部件移动 (308) 到边缘网络 (106) 通过将所述思洛模块 (802) 移动到所述边缘网络 (106) 来实现。

为进一步说明，图 9 阐明了示出针对所述思洛拆分类型的用于网络边缘计算的应用拆分的另一示例性方法的流程图。在图 9 的所述例子中，所述应用拆分点 (310) 被识别为调用 EJB 和调用 HTTP 功能的调用分级中的类，并且所述拆分类型 (312) 被识别为思洛拆分类型。在图 9 的所述方法中，根据所述拆分类型将所述应用 (130) 拆分 (306) 成子部件 (314) 通过将所述调用分级中的所有类作为一个子部件聚合到思洛模块 (802) 中来实现。在图 9 的所述方法中，将子部件移动 (308) 到边缘网络 (106) 通过将所述思洛模块 (802) 移动到所述边缘网络 (106) 来实现。

为进一步说明，图 10 阐明了示出针对所述思洛拆分类型的用于网络边缘计算的应用拆分的另一示例性方法的流程图。在图 10 的所述例子中，所述应用拆分点 (310) 被识别为调用 JCA 功能和调用 HTTP 功能的调用分级中的类，并且所述拆分类型 (312) 被识别为思洛拆分类型。在图 10 的所述例子中，根据所述拆分类型 (312) 将所述应用 (130) 拆分 (306) 成子部件 (314) 通过将所述调用分级中的所有类作为一个子部件聚合到思洛模块 (802) 中来实现。在图 10 的所述例子中，将子部件移动 (308) 到边缘网络 (106) 通过将所述思洛模块 (802) 移动到所述边缘网络 (106) 来

实现。

为进一步说明，图 11 阐明了示出针对所述正面拆分类型的用于网络边缘的应用拆分的示例性方法的流程图。在图 11 的所述例子中，所述应用拆分点（310）被识别为被不止一次列为潜在拆分点的调用分级中的类，并且所述拆分类型（312）被识别为正面拆分类型。在图 11 的所述方法中，根据所述拆分类型（312）将所述应用（130）拆分（306）成子部件（314）通过将所述调用分级中的类拆分成客户端 web 服务模块（702）和服务器 web 服务模块（704）来实现。在图 11 的所述方法中，将子部件移动（308）到边缘网络（106）通过将所述客户端 web 服务模块（702）移动到所述边缘网络（106）来实现。

从前面的描述中将理解，在不脱离本发明真正精神的情况下，修正和改变可以在本发明的各种实施例中产生。本说明书中的描述用于仅说明目的，并且将不被以限制性的意义理解。本发明的范围仅受权利要求的语言的限制。

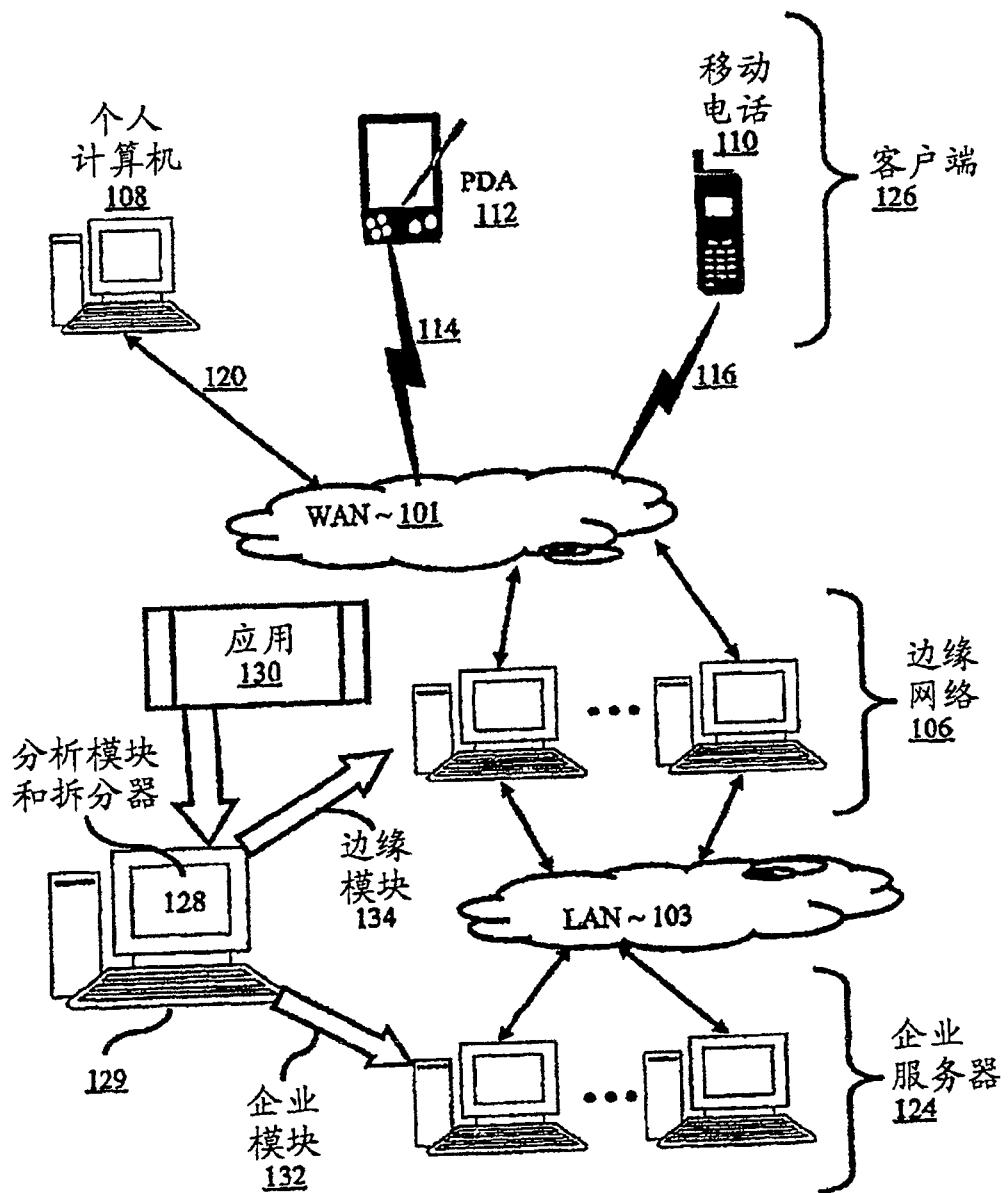


图 1

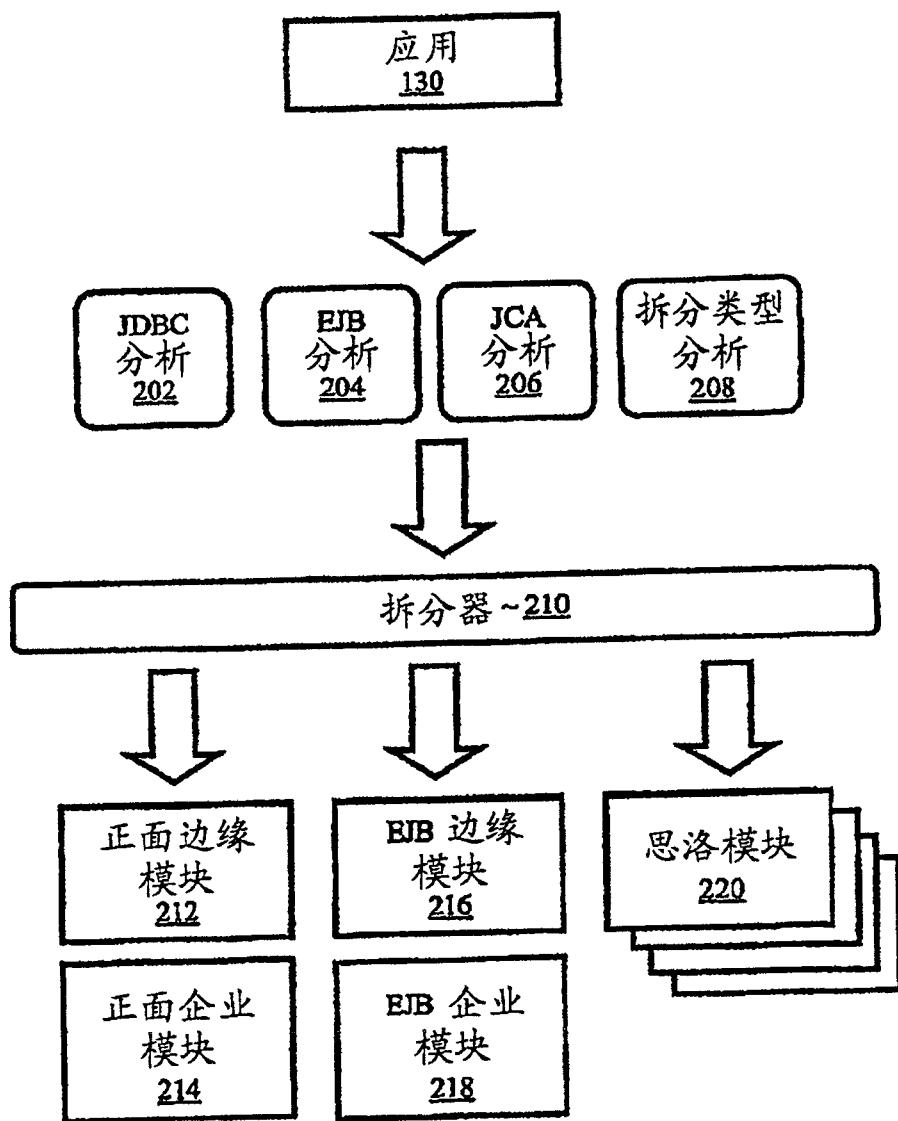


图 2

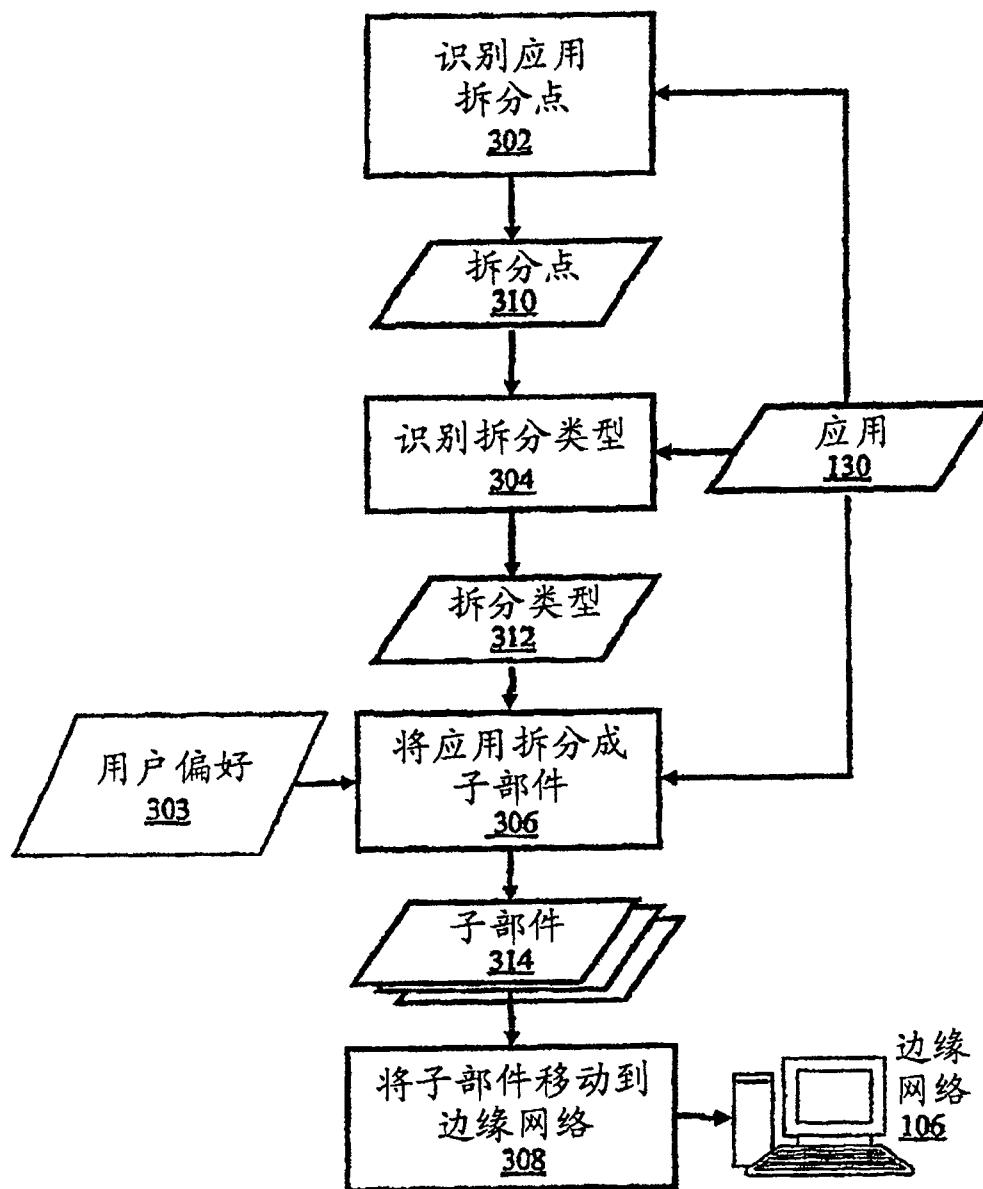


图 3

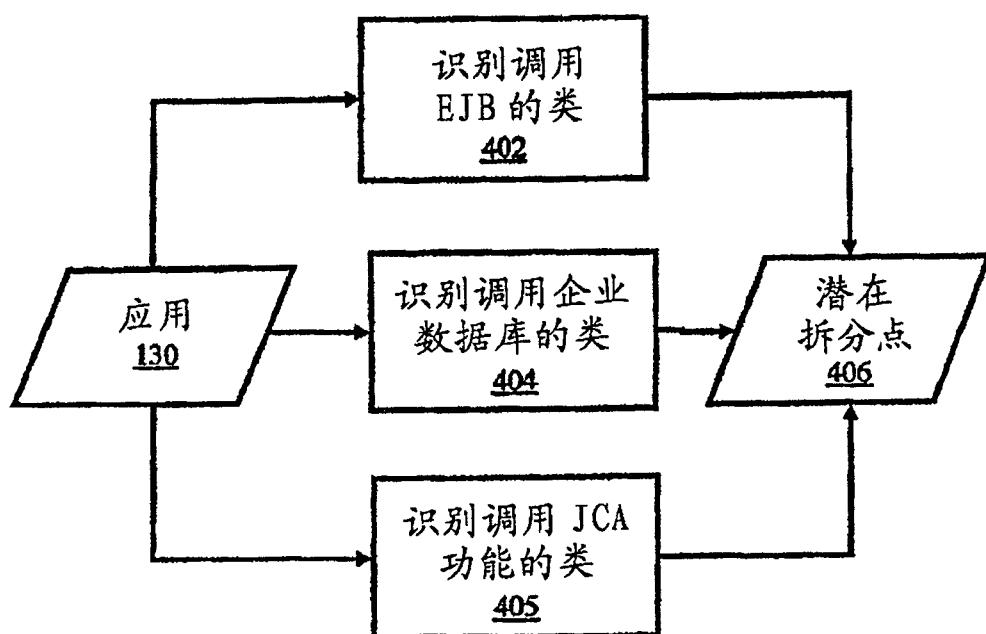


图 4

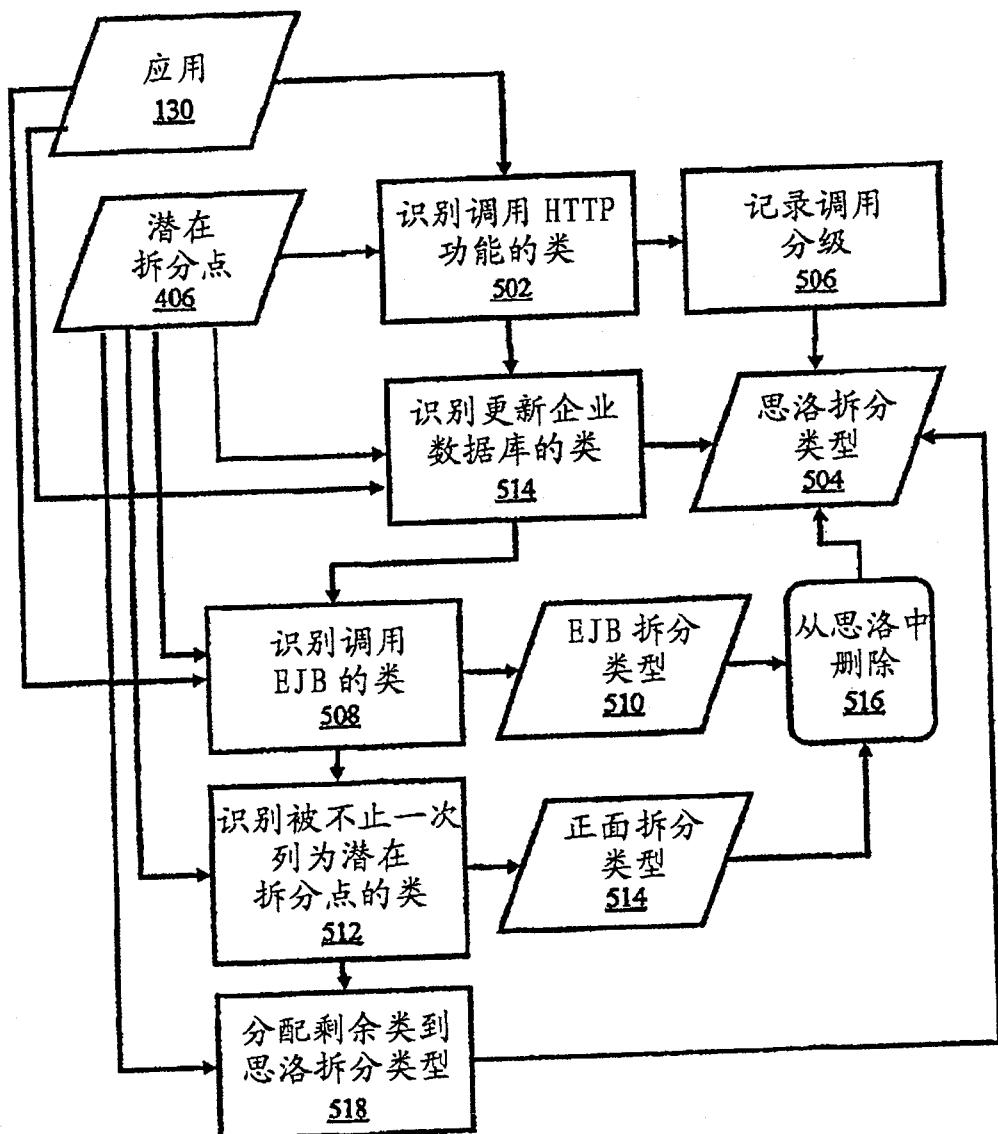


图 5

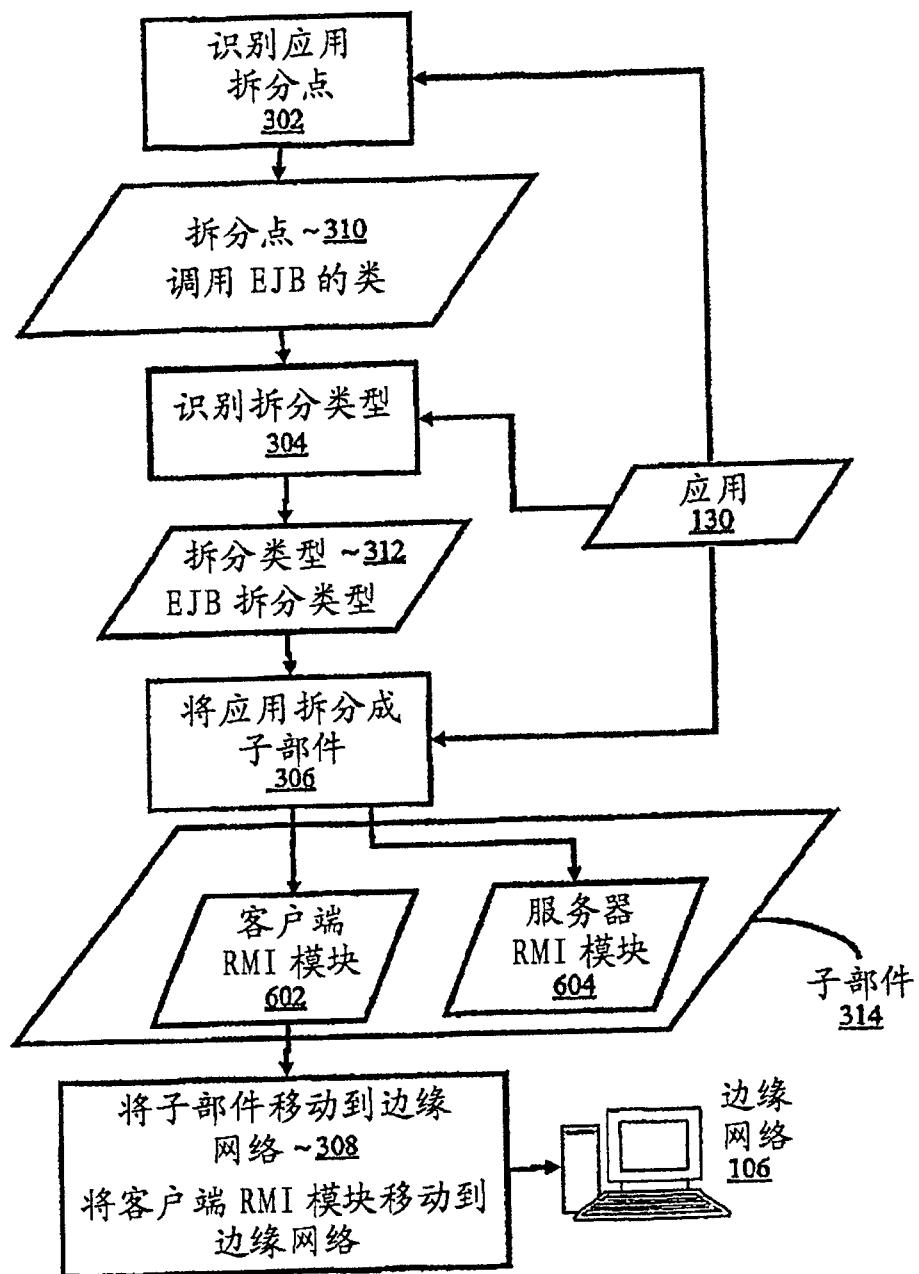


图 6

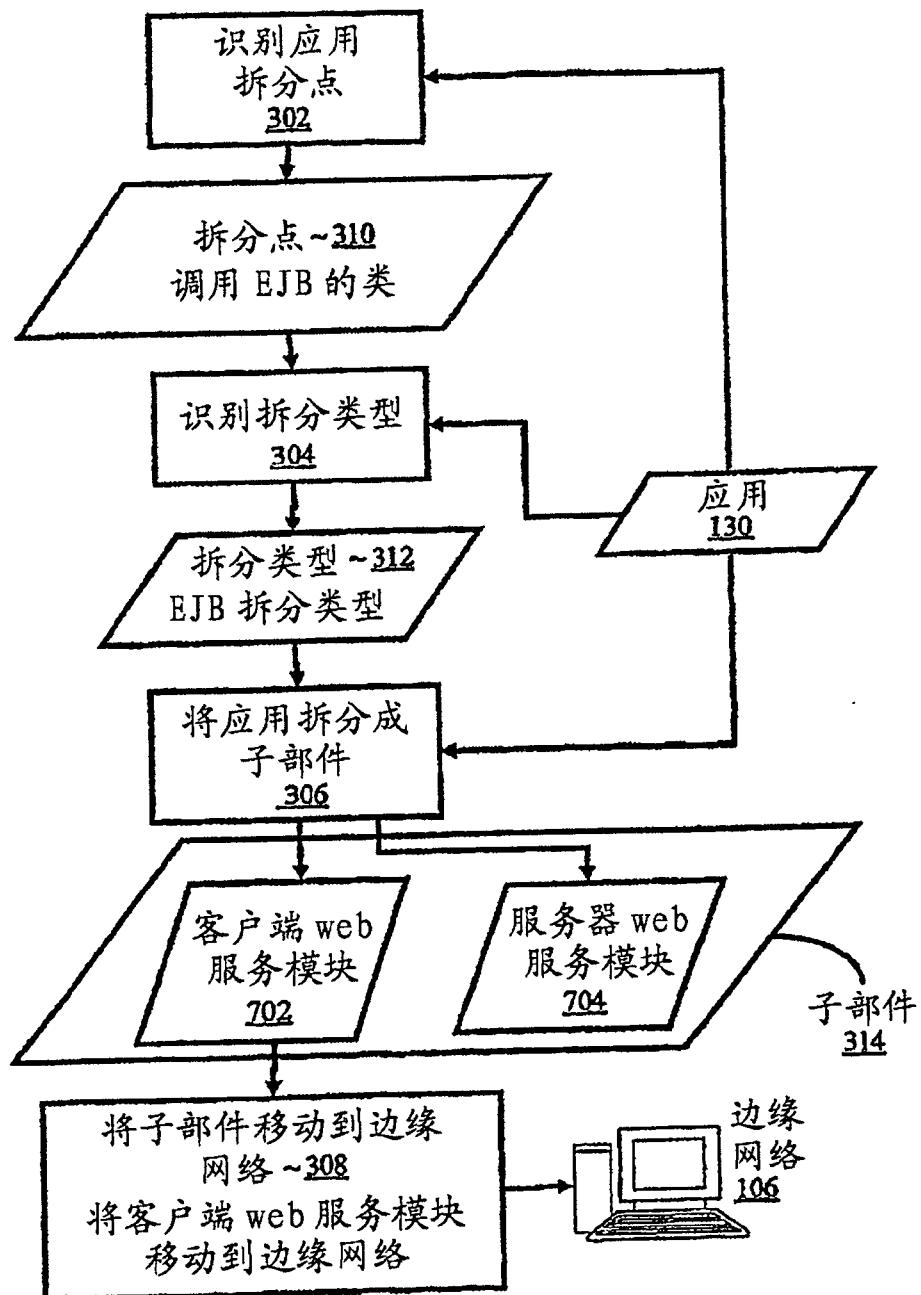


图 7

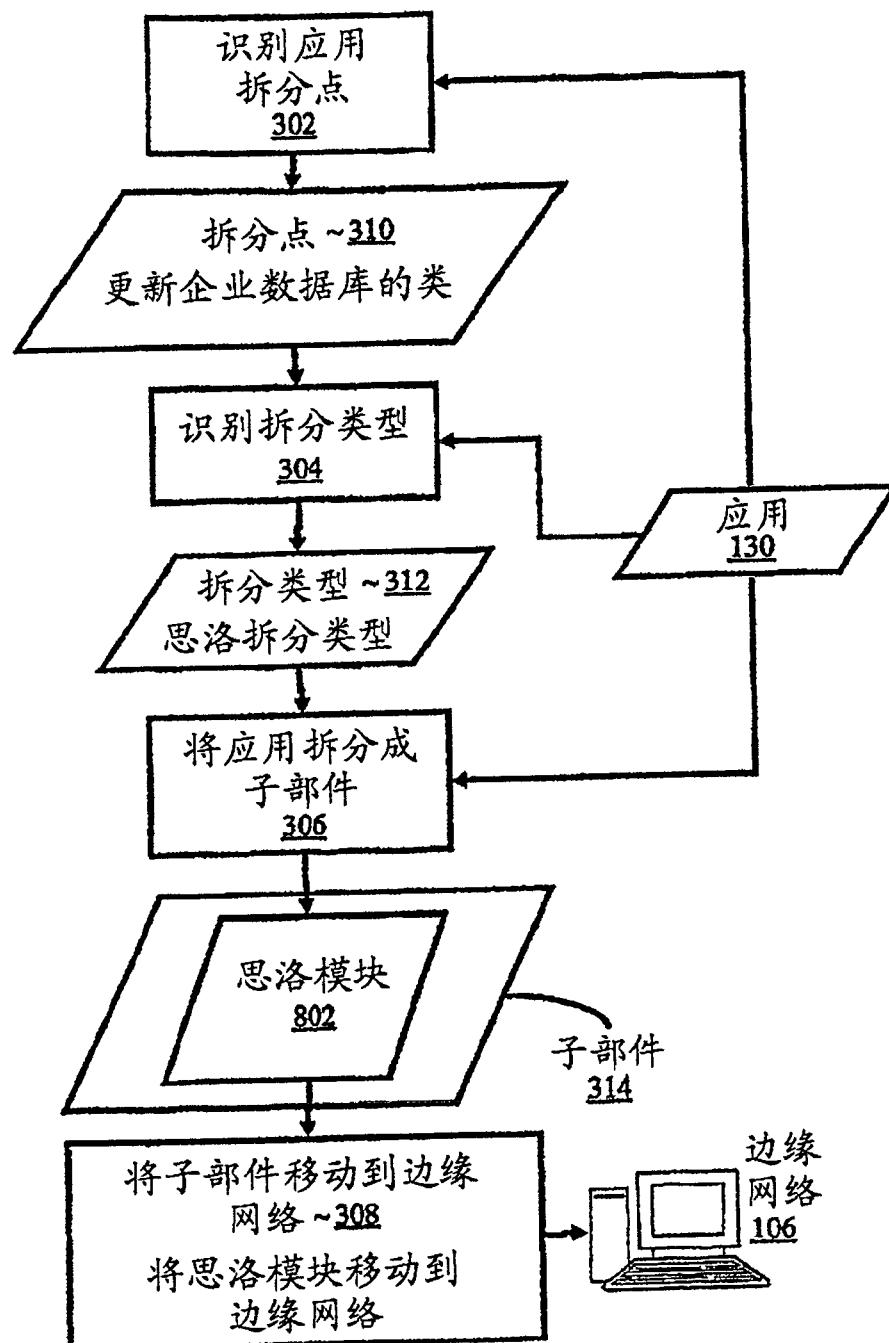


图 8

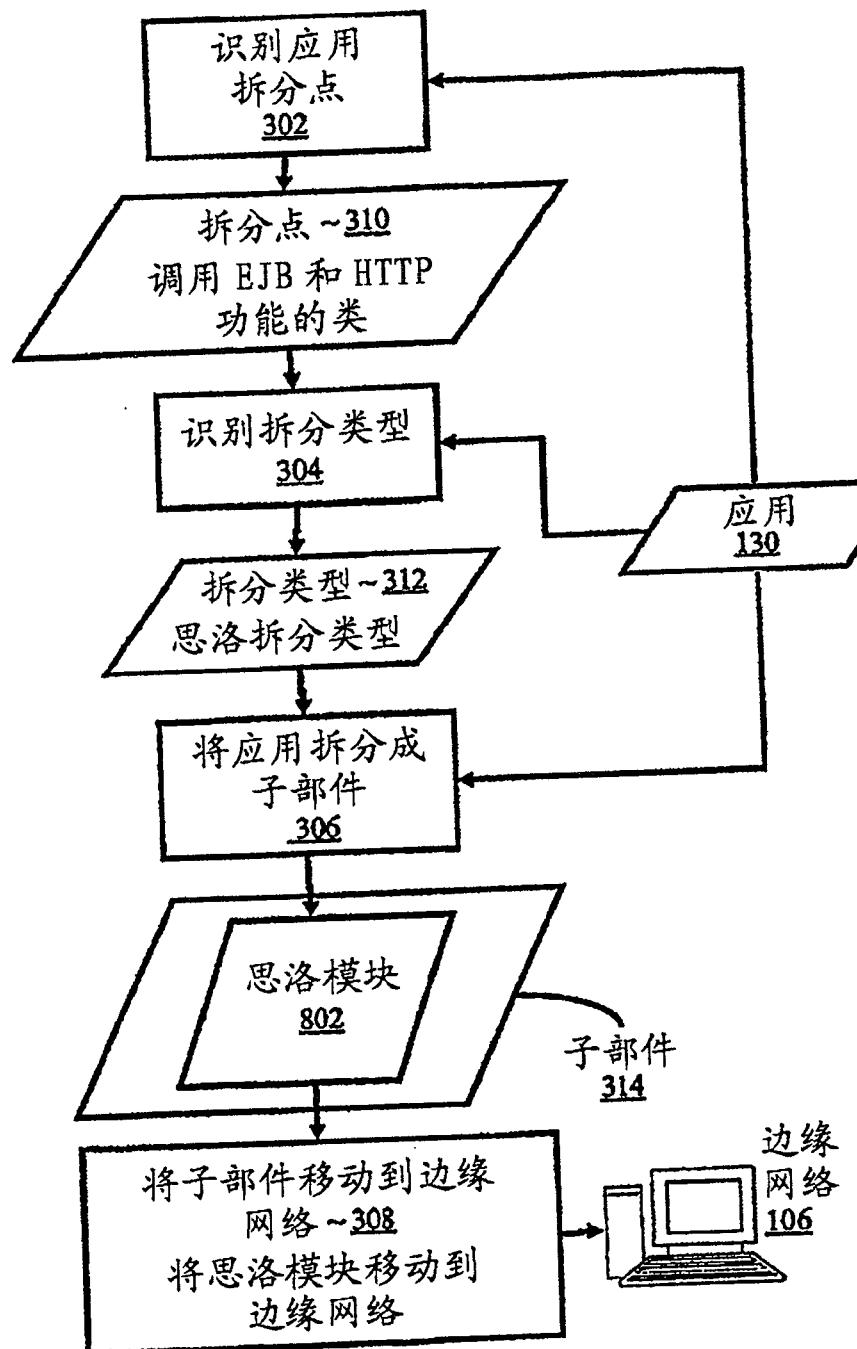


图 9

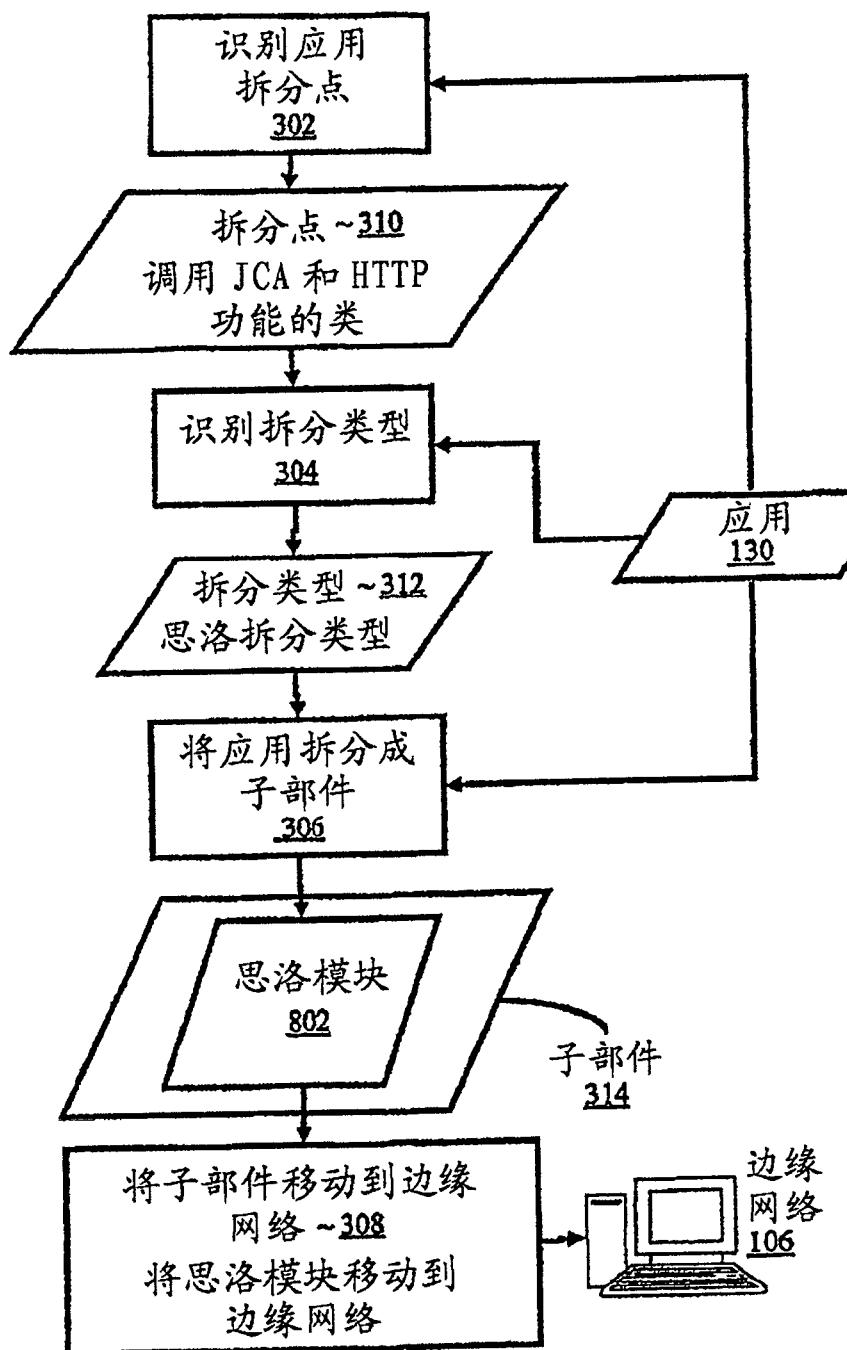


图 10

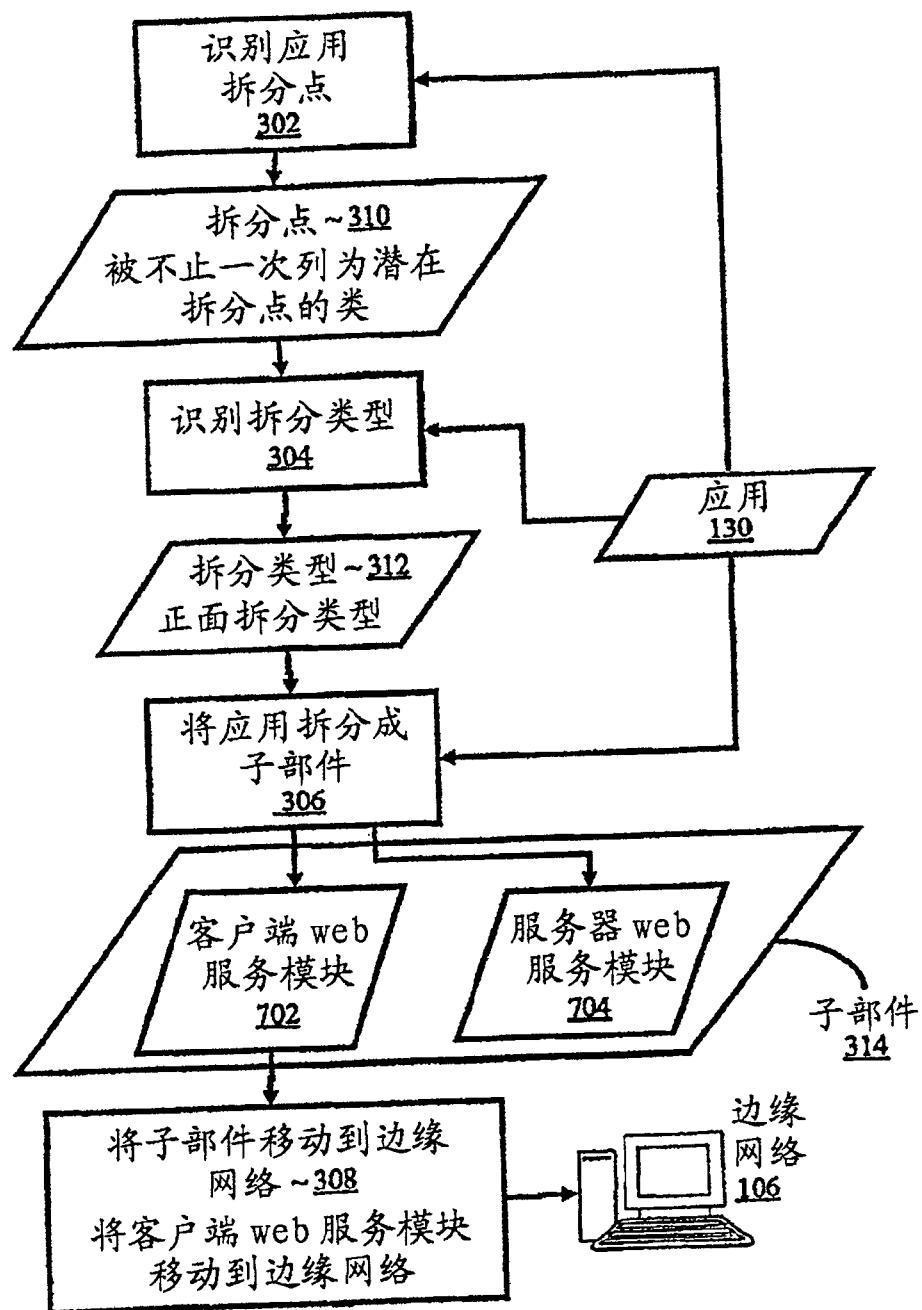


图 11