

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.  
G06F 9/44 (2006.01)



# [12] 发明专利申请公开说明书

[21] 申请号 200510089509.9

[43] 公开日 2006年4月5日

[11] 公开号 CN 1755619A

[22] 申请日 2005.7.22

[21] 申请号 200510089509.9

[30] 优先权

[32] 2004.9.30 [33] US [31] 10/954,328

[71] 申请人 微软公司

地址 美国华盛顿州

[72] 发明人 J·P·罗森伯格 R·-C·肖

[74] 专利代理机构 上海专利商标事务所有限公司  
代理人 张政权

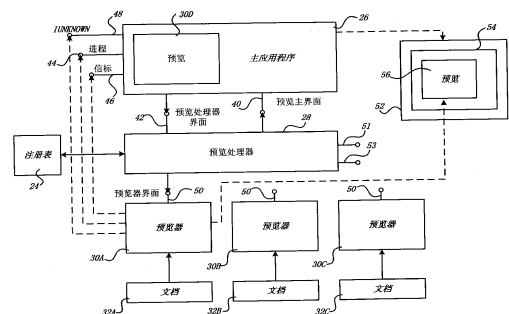
权利要求书 4 页 说明书 11 页 附图 5 页

## [54] 发明名称

提供文档预览的方法、系统和装置

## [57] 摘要

提供了生成文档预览的方法和装置。依照该方法，预览处理器提供到一个或多个预览器的界面。预览器是能够为一个或多个文件类型生成视觉预览的软件组件。预览处理器可操作地为文档生成一预览而标识适当的预览器以响应来自主应用程序为文件生成预览的请求。预览处理器接着会调用预览器并指令预览器载入文档并在由调用主应用程序提供的窗口的次窗口中显示预览。



1. 提供文档预览的方法，所述方法包括：  
提供能够被一个或多个主应用程序调用的预览处理器，所述预览处理器用于标识和载入预览器以生成主应用程序所标识的文档的视觉预览；  
在预览处理器处从主应用程序接收请求以生成文件的视觉预览；  
标识能够生成文档预览的预览器以响应接收请求；以及  
将来自预览处理器的请求发送到预览器，指示预览器载入所述文件并生成所述文档的视觉预览。
2. 如权利要求1所述的方法，其特征在于，所述来自主应用程序的请求还包含对于一窗口的句柄，应该在所述的窗口中生成预览，其中所述预览器还用于在由所述主应用程序标识的窗口的次窗口中显示预览。
3. 如权利要求2所述的方法，其特征在于，所述标识能够生成文档预览的预览器包括请求所述主应用程序标识所述预览器。
4. 如权利要求2所述的方法，其特征在于，所述标识能够生成文档预览的预览器包括请求所述主应用程序标识对应于所述预览器的类标识符。
5. 如权利要求2所述的方法，其特征在于，所述标识能够生成文档预览的预览器包括搜索注册表的内容以标识所述预览器。
6. 如权利要求2所述的方法，其特征在于还包括：  
确定浏览器是否成功地载入所述文档；以及  
从所述预览处理器返回出错消息到主应用程序以响应确定所述预览器没有成功地载入所述文档。
7. 如权利要求6所述的方法，其特征在于还包括在预览处理器处接收来自所述主应用程序的请求以显示对应于所述出错消息的消息并显示响应于所述请求的消息。

8. 如权利要求 7 所述的方法，其特征在于还包括：  
在预览处理器处从主应用程序接收指向进程界面的指针；  
将所述指向进程界面的指针从所述预览处理器传递到所述预览器；以及  
经由进程界面，从预览器发送进程消息到主应用程序。

9. 如权利要求 7 所述的方法，其特征在于还包括查询所述主应用程序生成文档预览所需的附加文件的位置。

10. 如权利要求 7 所述的方法，其特征在于还包括：  
在预览处理器处从所述主应用程序接收指向信标界面的指针；  
将指向信标界面的指针从预览处理器传递到预览器；以及  
查询所述主应用程序以获取载入在所述信标界面上生成文档预览所需的支持文件的许可。

11. 计算机可读介质，具有存储于其上的计算机可执行指令，当计算机执行所述可执行指令时会造成计算机将预览处理器界面展现给一个或多个主应用程序，所述预览处理器界面可操作用于展现将一个或多个用于标识和载入用于生成文档预览的预览器的方法。

12. 如权利要求 11 所述的计算机可读介质，其特征在于，所述预览处理器界面包括用于初始化预览处理器的初始化方法。

13. 如权利要求 12 所述的计算机可读介质，其特征在于，所述预览处理器界面包括用于标识和载入所述预览器的载入方法。

14. 如权利要求 13 所述的计算机可读介质，其特征在于，标识所述预览器包括由所述主应用程序调用被展现用于获取所述预览器的身份的界面。

15. 如权利要求 14 所述的计算机可读介质，其特征在于，标识所述预览器包括由所述主应用程序调用被展现用于获取所述预览器的类型标识符的界面。

16. 如权利要求 14 所述的计算机可读介质, 其特征在于, 标识所述预览器包括搜索注册表以获取注册为所述文档类型的预览器的预览器。

17. 如权利要求 14 所述的计算机可读介质, 其特征在于, 所述预览处理器界面包括用于显示所述预览的显示方法。

18. 如权利要求 14 所述的计算机可读介质, 其特征在于, 所述预览器可操作用于展现包括用于显示预览的显示方法的预览界面, 所述预览处理器可操作用于在所述预览器界面上调用所述显示方法以响应接收在所述预览处理器界面上对所述显示方法的调用。

19. 用于提供文档预览的方法, 所述方法包括在主应用程序处展现预览主界面, 所述预览主界面包括获取预览器方法, 预览处理器可以通过该方法为文档类型获取预览器身份, 获取预览器类标识符方法, 所述预览处理器通过该方法可以为文档类型预览器获取类标识符, 获取界面方法, 所述预览处理器通过该方法可获取指向其它由主应用程序提供的界面的指针, 以及完成载入方法, 所述预览处理器通过该方法可通知所述主应用程序要预览的文档已经被载入。

20. 如权利要求 19 所述的方法, 其特征在于还包括在所述预览处理器处展现预览处理器界面, 所述预览处理器界面包括用于初始化所述预览处理器的初始化方法、用于指令所述预览处理器标识预览器和将要预览的文件载入到预览器的载入方法、用于显示预览的显示方法、用于指令所述预览处理器显示预览的窗口已经被改变的窗口改变方法以及用于撤销对所述预览处理器的操作初始化的撤销初始化方法。

21. 如权利要求 20 所述的方法, 其特征在于还包括在所述预览器处展现预览界面, 所述预览界面包括用于接收指令以载入要预览的文档的载入方法、用于显示指令以显示预览的显示方法以及用于接收预览所显示的窗口已被改变的指示的窗口改变方法。

22. 如权利要求 21 所述的方法，其特征在于，所述预览器界面上的载入方法被所述预览处理器调用以响应在所述预览处理器界面上接收调用所述载入方法。

23. 如权利要求 21 所述的方法，其特征在于，所述预览界面上的显示方法被所述预览处理器调用以响应在所述预览处理器界面上接收调用所述显示方法。

24. 如权利要求 21 所述的方法，其特征在于，所述预览界面上的窗口改变方法被所述预览处理器调用以响应在所述预览处理器界面上接收调用所述窗口改变方法。

## 提供文档预览的方法、系统和装置

### 相关申请的交叉引用

本专利申请与同时提交题为“Method And Computer-Readable Medium For Previewing And Performing Actions On Attachments To Electronic Mail Messages”（代理人案号为 60046.432US01）的美国专利申请有关，后者通过引用被特别地包含于此。

### 背景技术

当使用具有现代图形操作系统的计算机系统时，通常期望浏览文件的预览。文档预览是对文件的只读重现，它模仿当用与文件相关联的应用程序打开该文件时会看到的东西。一些应用程序允许用户看到文件的静态图像，但是静态图像不是用户可以翻页通过或导航的真实预览。其他程序需要用户启动与该文件相关联的应用程序来观看其内容。例如，在电子邮件（“e-mail”）客户端应用程序中，有时期望看到附加在 e-mail 上的一个或多个文件的预览。然而当前的 e-mail 应用程序需要用户启动与所附文档相关联的应用程序来浏览该文件。这个过程对用户来说会是耗时且令人沮丧的。

也期望在各种环境中浏览文档预览。例如，可能期望在应用程序的环境中浏览文档预览，诸如上述的 e-mail 客户端应用程序。也可能期望在操作系统的环境中浏览文档预览，诸如图形桌面外壳。例如，当用户利用外壳导航含有文档的文件夹时，用户可能期望快速浏览文档预览而无需启动相关联的应用程序。类似地，可能期望快速用于打开文件的对话框环境中快速浏览文档预览。在这种方式中，用户能够浏览文档预览并快速地检查文件内容而无需打开与文档相关联的应用程序。然而，为了预览使用当前系统的文件，每个应用程序和操作系统必须为预览文档实现它们自己的代码。

也可能期望提供在各种环境中快速预览文档而不影响计算机系统的响应性的装置。此外，在预览文档时遇到的任何错误不应该影响主操作系统或应用程序的性能或稳定性。本发明的各种实施例是关于这些和其它考虑而作出的。

## 发明内容

依照本发明的实施例，用于提供文档预览的方法、系统和装置解决了上述和其它的问题。本发明的实施例提供了操作系统程序和应用程序可以利用以提供对文件的快速预览而无需启动与文档相关联的应用程序的通用装置。本发明的各个实施例提供的装置允许对预览翻页和导航。实施例也提供了不会严重地影响计算机系统性能和稳定性的预览设施。

依照本发明的一个方面，提供了生成文档预览的方法。依照该方法，提供了展现可由期望显示文档预览的主应用程序调用的预览处理器界面的预览处理器。例如，电子邮件客户机应用程序会包括一主应用程序，并且可以调用预览处理器界面来显示作为一电子邮件消息的附件的文档预览。通过预览处理器界面，预览处理器可操作用于标识用于显示文档预览的预览器，用于加载预览器并指示预览器建立和显示预览。

为了建立和显示预览，主应用程序会调用预览处理器界面，请求载入要预览的文件。请求会包括到该文件的路径和到预览应该被显示的窗口的句柄。响应于接收请求，预览处理器可以标识适当的预览器来为文档生成预览。可以通过查询主应用程序所提供的可选界面、通过搜索经注册的预览器的注册表或通过其它方法来标识预览器。

一旦为文档类型标识了预览器，预览处理器会调用由预览器展现的界面以指示预览器载入文档。预览处理器也会调用预览界面以在主应用程序所标识的窗口的次窗口中示出文档预览。会在独立的处理空间中执行预览器，以不影响主应用程序的性能。为了类似的原因，预览处理器和预览器之间的通信会在背景线程上发生。

一些可任选界面会由主应用程序实现以为预览器所使用。特别地，会由主应用程序提供信标界面，预览器会通过信标界面询问主应用程序以允许载入生成文档预览所需的支持文档。会由主应用程序实现进度界面，预览器会通过进度界面来提供进度指示符，诸如载入文档和生成预览的进度。也会提供界面，预览器通过该界面获取定位生成文档预览所需的附加文件的界面。

本发明的实施例会被实现为计算机进程、计算机系统或制造物品，诸如计算机程序产品或计算机可读介质。计算机程序产品可以是计算机系统可读的计算机存储介质并编码用于执行计算机进程的计算机程序指令。计算机程序产品也可以是在计算机系统可读的载波器上的传播信号并解码用于执行计算机进

程的计算机程序指令。

通过阅读一下详细描述和查看相关联的附图，表征本发明的这些和各种其它特征以及优势将变得显而易见。

#### 附图说明

图 1 是说明本发明的各个实施例中所使用和提供的计算机系统的计算机系统构架图；

图 2 是说明本发明的实施例所提供的各个软件组件和界面的各方面的软件构架图；以及

图 3—5 是说明由本发明的实施例所提供的用于提供文档预览的各个软件组件操作的流程图。

#### 具体实施方式

将参见附图描述本发明的各个方面，其中相同的号码表示相同的元件。特别地，图 1 和相应的讨论意在提供对实现本发明实施例的适当计算环境的简要、一般描述。虽然会在编程模块的一般环境中描述本发明（其中会结合在个人计算机上的操作系统上运行的程序模块来执行本发明），本领域的技术人员会认识到本发明也可以结合其它类型的计算机系统和编程模块来实现。

通常，程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、组件、数据结构等。此外，本发明的技术人员会理解也可以用其它计算机配置实现本发明，所述计算机配置包括手持式设备、多处理器系统、基于多处理器或可编程消费电子产品、微型计算机、大型计算机等。本发明还可以在分布式计算环境中实现，其中在分布式计算环境中任务由通过一通信网络链接的远程处理设备执行。在分布式计算环境中，程序模块可以位于包含存储器设备的本地和远程计算机存储介质中。

参见图 1，将描述在本发明的各个实施例中所使用的计算机 2 的说明性计算机构架。图 1 中所示的计算机构架说明了通用桌上型或膝上型计算机，包括中央处理器 5（“CPU”）、系统存储器 7、包括随机存取存储器 9（“RAM”）和只读存储器（“ROM”）11 和将存储器耦合到 CPU 5 的系统总线 12。包含有助于例如在启动过程中在计算机内部的元件之间传输信息的基本例程的基本输入/输出系统存储在 ROM 11 中。计算机 2 还包括用于存储操作系统 16、应用程序和其它程序模块的大容量存储设备



14, 这会在下文中详细描述。

大容量存储设备 15 通过连接到总线 12 的大容量存储控制器(未示出)连接到 CPU 5。大容量存储设备 14 和器相关联的计算机可读介质为计算机 12 提供了非易失性存储。虽然这里所包含的对计算机可读介质的描述指大容量存储设备, 诸如硬盘或 CD-ROM 驱动器, 本领域的技术人员应该理解计算机可读介质可以是计算机 2 能够访问的任何可用介质。

通过举例而非限制, 计算机可读介质可以包括计算机存储介质和通信介质。计算机存储介质包含以任一方法或技术实现的用于信息存储的易失性和非易失性介质、移动和不可移动介质, 所述信息诸如计算机可读指令、数据结构、程序模块或其它数据。计算机存储介质包含但不限于: RAM、ROM、EPROM、EEPROM、闪存或其它固态存储器技术、CD-ROM、数字通用盘(“DVD”)或其它光学存储器、磁带盒、磁带、磁带存储器或者其它磁存储设备、或者任何其它能够用来存储所需信息并且能够由计算机 2 访问的介质。

依照本发明的各种实施例, 计算机 2 会在使用通过诸如 Internet 等网络 18 到远程计算机的逻辑连接在网络环境中操作。计算机 2 可以通过连接到总线 12 的网络接口单元 20 连接到网络 18。应该理解网络接口单元 20 也可以被用于连接到其它类型的网络和远程计算机系统。计算机 2 也可以包括用于接收和处理来自多个其它设备的输入的输入/输出控制器 22, 所述其它设备诸如键盘、鼠标或电子笔(图 1 中未示出)。类似地, 输入/输出控制器 22 会提供到显示屏、打印机或其它类型的输出设备的输出。

如上面简要提到的, 多个程序模块和数据文件会存储在计算机 2 的大容量存储设备 14 和 RAM 9 中, 包括适用于控制网络个人计算机操作的操作系统 16, 诸如来自 Washington Redmond 的微软公司的 WINDOWS XP 操作系统。大容量存储设备 14 和 RAM 9 也可以存储一个或多个程序模块。特别地, 大容量存储设备 14 和 RAM 9 会存储主应用程序 26。主应用程序 26 包含期望向用户提供文档 32 预览的任何类型的程序。依照本发明的实施例, 主应用程序 26 包括来自微软公司的 OUTLOOK 个人信息管理器应用程序。可以通过本发明的各个实施例为附在 e-mail 消息上的文件生成和显示预览。然而, 应该理解, 主应用程序 26 可以包含文字处理器应用程序、电子数据表应用程序、演示应用程序、操作系统 16 或任何其它类型的程序。这些程序会使用上述装置在例如文件打开对话框、操作系统外壳或任何其它的位置中提供文档预览。

大容量存储设备 14 和 RAM 9 也被用于存储预览处理器程序 28。预览处理器 28 是用于提供文档预览的装置。如下面要详细描述，预览处理器 28 实现诸如主应用程序 26 可以调用以生成文档预览的界面。由预览处理器 28 提供的界面将文档预览的提供从主应用程序 26 抽象出来。

依照本发明的实施例，预览处理器 28 会使用一个或多个预览器程序 30 来生成实际的文件 32 预览。预览器会向预览处理器 28、主应用程序 26 或操作系统 16 注册为特定类型文件的预览器。有关经注册的预览器的信息由预览处理器 28、主应用程序 26 或操作系统来维护，所述信息是关于经注册的预览器的身份和位置。操作系统 16 会在注册表 24 中存储这一信息。应该理解主应用程序 26 自身会被注册为不同类型的文档的预览器。关于主应用程序 26、预览处理器 28 和预览器 30 的附加细节将关于图 2-5 在下面描述。

现在参见图 2，将提供关于本发明的实施例所使用的各种软件组件的附加细节。如图 2 所示，主应用程序 26 通过使用预览处理器 28 所实现的预览处理器界面 42 与预览处理器 28 通信。反过来，预览处理器 28 会经由主应用程序所实现的预览主界面 40 与主应用程序通信。预览处理器也会通过使用预览界面 50 与预览器 30A-30D 通信。应该理解主应用程序 26 会实现预览器 3D 并以这里所描述的关于预览器 30A-30C 相似的方式与预览处理器 28 通信。也应该理解预览处理器 28 会从注册表 24 检索以标识适当的预览器及其位置。

如上简述的，预览器 30A-30C 的每一个用于生成一个或多个文档类型的预览。文档 32A-32C 一般存储在预览器 30A-30C 可以访问的位置中的大容量存储设备 14 上。当预览处理器 28 发出载入请求时，这个位置被传递给预览器 30A-30C。这个过程会在下面详细描述。

预览器 30A-30C 会通过使用一个或多个可选界面直接与主应用程序 26 通信。特别地，进程界面 44 会被用于向主应用程序 26 提供进程指示符。进程指示符于是会被显示给用户以提供例如预览显示进程的图形指示。信标界面 46 会被预览器 30A-30C 用于获取主应用程序 26 的允许以载入与文档相关联的支持文件。例如，如果要被预览的文档是超文本链接标示语言(“HTML”)文档，会需要载入图形和其它参考文件以提供预览。在载入参考文件之前，预览器会请求主应用程序 26 授权通过信标界面 46 来载入参考文件。IUnknow 界面 48 会被预览器 30A-30C 用于提供执行各种功能的界面，诸如定位生成文档预览所需的支持文件。如本发明的技术人员所公知的，IUnknow 界面 48 让客户端获取

给定对象上到其它界面的指针。

为了提供预览，主应用程序 26 将处理传递到预览处理器 28 的窗口 52。预览处理器接着在主应用程序创建的窗口内创建子窗口(或次窗口)。当预览器已被标识和实例化时，对于窗口的句柄接着被传递到适当的预览器 30A—30C。为了在窗口中再现预览 56，预览器在窗口 52 中创建次窗口 54。因为为预览指示的用户输入会被预览器接收，预览器会用于处理某些类型的输入并将其它类型的处理传递到预览处理器 28，随后到主应用程序 26。在这种方式中，预览器接收到的实际意在主应用程序 26 的输入会由主应用程序 26 处理。

再参见图 2，将提供关于预览处理器界面 42、预览主界面 40 和预览器界面 50 的附加细节。预览处理器界面 42 提供“初始化”方法，主应用程序 26 在创建预览处理器 28 的实例后调用所述“初始化”方法。初始化方法允许预览处理器 28 在窗口 52 上将自己初始化。传递到初始化方法的参数是主应用程序 26 的实例、到预览主界面 40 的指针以及会被用于定义预览处理器 28 的操作的一个或多个标记。也会在界面 42 上提供设置窗口方法，所述界面取得预览应该在其中被再现的窗口 52 的句柄。

预览处理器界面 42 也提供“载入”方法。当接收到生成文档预览的请求时，由主应用程序 26 调用载入方法，所述文档预览诸如用户选择要预览的一个或一组文件。当调用该方法时，预览处理器 28 保存要预览的文件的名称并调用后台线程以预览该文件。后台线程接着确保实际上是有文件要预览、获取文件的扩展名并试图为该扩展名定位预览器 30A—30C。如果定位了预览器，就使用该预览器预览文件。载入方法将到要预览的文件的完整路径来作为参数。下面将描述关于载入方法的操作的附加细节。

预览处理器界面 42 也包括“显示”方法。通过调用载入方法将文件成功地载入到预览器中后，主应用程序 26 调用显示方法。如果主应用程序 26 要将文件再现到预览中，它就调用显示方法。预览处理器界面 42 也包括“显示出错消息”方法。如果载入或显示方法返回出错代码且主应用程序 26 要预览处理器 28 显示代码的出错消息就调用显示出错消息方法。显示出错消息获取不能被载入或显示为参数的文件的出错代码和文件名。

预览处理器界面 42 也包括一“窗口已改变”方法。这一方法在被传到设置窗口方法的窗口 52 以某些方式改变时被调用。例如，如果窗口 52 的大小改变，主应用程序 26 可以调用该方法来通知预览处理器 28 有关于这一变化。然

后，预览处理器 28 可以调用适当的预览器来更新子窗口 52 和预览 56 的大小。预览处理器界面 42 也包括一“未初始化”方法，该方法在没有其余要被预览的文档时被调用。未初始化方法解除对预览处理器 28 的初始化。

预览主界面 40 是可任选的调回界面，主应用程序 26 实现所述所选调回界面以允许预览处理器 28 使用对主应用程序 26 可用的资源。例如，如果主应用程序 26 已经载入特定文档类型的预览器，不是让预览处理器 28 试图自己载入预览器，而是将界面传递到载入的预览器对主应用程序 26 是有用的。

预览主界面 40 提供“获取预览器”方法。在预览处理器 28 试图为文件定位预览器时，预览处理器 28 会调用这个方法。通过获取预览器方法，预览处理器 28 能够为该文件类型询问主应用程序 26 预览器的身份。获取预览器方法将要预览文件的扩展名作为输入并为文件类型提供指示预览器身份的输出参数或空。这个方法在例如主应用程序 26 已经载入预览器或主应用程序 26 提供预览器 30D 自身时会是有用的。

如果获取预览器方法不返回一预览器，预览处理器 28 会利用由预览主界面 40 提供的“获得预览器类型 ID”方法。通过获取预览器类型 ID 方法，主应用程序 26 能够将它要为给定扩展名的特定文件所使用的任何预览器的类型 ID 传递到预览处理器 28。例如，主应用程序 26 会要指定特定预览器被用于特定的可扩展标记语言(“XML”)文件。获取预览器类型 ID 方法将要预览的文件的扩展名作为输入并为文件类型提供指示预览器类型 ID 的输出参数或空。

预览主界面 40 也提供“获取界面”方法。当指示预览处理器 28 载入文件，它尝试获取到信标界面 46、进程界面 44 和 IUnknown 界面 48 的指针以传递到预览器供载入和显示时使用。获取界面方法被用于这个目的。获取界面方法将所请求的界面的身份作为参数，并且如果返回就将指向所请求界面的指针作为输出参数返回。

预览主界面 40 也提供“完成载入”方法。当预览器完成载入一文件，会调用这个方法让主应用程序 26 知道会采取的下一步骤。例如，主应用程序 26 会在预览处理器界面 42 上调用显示方法以响应接收对完成载入方法的调用。或者，如果文档不能被载入，会以出错消息调用完成载入方法。在这种情况下，主应用程序 26 会在预览处理器界面 42 上调用“显示出错消息”方法。

预览界面 50 是要实现的预览器 30A-30D 的基本界面。通过实现这个界面，会调用预览器载入文件、再现该文件到次窗口、如果需要提供内部文件导航以

及当预览结束时卸载该文件。预览器会在后台线程上被调用，并且也应该试图尽可能快地载入和再现因为用户可能在预览处等待。

预览器界面 50 展现“载入”方法，所述载入方法被调用以初始化特定文件上的预览器。载入方法将要预览的文件的路径和到信标界面 46、进程界面 44 和 IUnknown 界面 48 的可选指针作为参数。这些界面可以被用在上述的方式中。

当载入文件以响应对载入方法的调用时，预览器不应该调用任何用户界面，诸如进程对话等等。文件不应该被写入锁定且预览器不应该以任何方式改变或更新文件。如果文件是密码保护的，预览器也应该无声地失效。预览器也不应该执行包含在文件中的任何宏。预览器也应该被设计成可以调用载入方法一次且可以用潜在的不同窗口多次调用“显示”方法(下面论述)。载入方法应该执行任何在准备要被再现的文件中所涉及的耗时的工作，因此“显示”方法可以尽可能快。如果用户试图取消预览，载入方法应该停止载入文件并尽可能快地返回。

如上所述，预览器界面 50 也展现“显示”方法。调用该方法来使预览器将传递到载入方法的文件再现到特定的窗口。这是独立于载入方法的方法，允许浏览器载入文件而无需其也将文件展现至窗口。显示方法将窗口的句柄取作为参数，预览应该在所述窗口中被呈现。

预览界面 50 也展现“窗口改变”方法。调用该方法使得预览者知道当前使用的窗口由于某种原因被改变了。一般调用该方法以响应对预览处理器界面 42 上窗口改变方法的调用。预览器界面 50 也展现“卸载”方法。调用该方法使得预览者知道它应该卸载经由载入方法载入的文件。关于由预览处理器界面 42、预览主界面和预览界面 50 提供的各种方法的使用和操作的附加细节将在下文中参见图 3-5 详细描述。

依照本发明的实施例，预览处理器 28 也会支持 IService Provider 界面 51 或 IOle CommandTarget 界面 53。界面 51 是预先存在的界面，允许组件展现非相关的功能。通过界面 51，主应用程序 26 能够询问当前预览器是否支持某些类型的功能。界面 53 是另一个预先存在的界面，允许调用程序(主应用程序 26)向客户机(预览器)发出命令。预览处理器 28 将这个界面实现为到当前预览器的传递通路。

现在参见图 3，将描述说明性例程 300，说明了提供文档预览的过程。当

读到这里所示的例程的讨论时，应该理解本发明的各个实施例的逻辑操作被实现为(1)在计算系统上运行的一系列计算机实现的动作或程序模块和/或(2)计算系统内互相连接的机器逻辑电路或电路模块。所述的实现是取决于实现本发明的计算系统的性能要求的一种选择。相应地，图 3-5 所示的逻辑操作以及这里所描述的组成本发明的实施例是指不同的操作、结构设备、动作或模块。本领域的技术人员应该认识到这些操作、结构设备、动作和模块可以用软件、固件、专用数字逻辑以及它们和任何组合来实现，而不背离这里提出的权利要求中所述的本发明的精神和范围。

例程 300 在操作 302 处开始，其中确定是否接收到生成一个或多个文档的请求。举例而言，这种请求可以由用户选择带有一个或多个附件的 e-mail 消息作出。或者，这种请求可以在编程上由操作系统或其它类型的程序实现。如果没有接收到预览请求，例程 300 返回到操作 302，在其中作出另一个这样的确定。如果接收到请求，例程 300 从操作 302 继续到操作 304。

在操作 304 处，主应用程序 26 创建预览处理器 28 的一个实例。例程 300 接着继续到操作 306，其中主应用程序 26 在预览处理器界面 42 上调用初始化方法。预览处理器 28 以所述方式将其自身初始化以响应接收到对初始化方法 26 的调用。设置窗口方法也可以用窗口的句柄来调用，预览应该在所述窗口中生成。

例程 300 从操作 306 继续到操作 308，其中主应用程序 26 在预览处理器界面 42 上调用载入方法。预览处理器 28 试图为要预览的特定类型的文件表示预览器、载入该预览器并指示预览器载入文件以响应接收到对载入方法的调用。这个过程在操作 310 处发生。下面会关于图 4 描述说明行例程 400，所述例程 400 用于定位预览器和载入文件。

例程 300 从操作 310 继续到操作 312，其中确定是否表示出预览器且能够载入特定的文件。如果预览器不能载入文件，例程 300 转移到操作 316，其中通过用出错消息在预览主界面 40 上调用完成载入方法返回错误消息到主应用程序 26。主应用程序 26 会在预览处理器界面 42 上调用显示出错消息方法以响应这一错误。例程从操作 316 继续到操作 318，在操作 318 处结束。

在操作 312 处，如果确定预览器能够载入文件，那么例程 300 继续到操作 314，其中会用成功消息在预览主界面 40 上调用完成载入方法。主应用程序 26 会在预览处理器界面 42 上调用显示方法以响应接收这一消息。预览处理器 28

会在预览界面 50 上调用显示方法以造成预览器在次窗口 54 中显示预览来响应。当不再需要预览时，会调用各种撤销初始化方法来撤销对预览器和预览处理器 28 的初始化。下面将关于图 5 描述说明性例程 500 以对这些元件撤销初始化。例程 300 从操作 314 继续到操作 318，在操作 318 处结束。

现在回到图 4，将描述说明性例程 400，用于为文档标识和载入适当的预览器。例程 400 在操作 402 处开始，其中为与预览器通信创建后台线程。使用后台线程使得在预览器变得不响应的时间中可以创建和使用新的后台线程。这也允许主应用程序 26 在预览处理器 28 等待预览器载入和呈现文件时保持响应。这允许预览处理器 28 处理用户决定它们不要预览它们已经请求预览的文件的情况。例如，如果用户选择电子表格文件，并且当预览器载入文件时，用户决定他们实际上希望预览一图像，它们能够在主应用程序 26 中选择图像。预览处理器 28 能够快速确定现有的后台线程是否处于能够载入图像预览器的状态。如果不是，预览处理器 28 能够为图像预览器载入后台线程并无需等待电子表格文件完成载入就能载入图像的预览。

例程 400 从操作 402 继续到操作 404，其中确定所需的预览器是否已经被载入。为了改进性能，预览处理器 28 会在需要预览器后将预览器留在存储器中一段时间。在这种方式中，预览器有效地被高速存缓且在对预览器后续的请求中会节省载入时间。在操作 406 处，如果预览器已经被载入，例程 400 继续到操作 408 处，其中当前被载入的预览器指定为要预览的文件的预览器。如果所需的浏览器没有被载入，例程 400 从操作 406 转移到操作 422 处。

在操作 422 处，搜索注册表 24 以确定预览器将其自身注册为特定文档文件类型的预览器。如果在注册表中标识了预览器，那么例程 400 转移到操作 426，其中在注册表中标识的预览器被指定为文件的预览器。如果在注册表中没有发现预览器，那么例程 400 继续到操作 410。

在操作 410 处，预览处理器 28 在预览主界面 40 上调用获取预览器方法。如上所述，获取预览器方法允许预览处理器 28 从主应用程序 26 请求预览器的身份。在操作 412 处，如果确定主机已经标识出预览器以响应对获取预览器方法的调用，例程 400 转移到操作 414，其中所标识的预览器被指定为文件的预览器。如果主应用程序 26 没有指定预览器，那么例程 400 继续到操作 416。

在操作 416 处，在预览主机界面 40 上调用获取预览器类 ID 方法。如果主应用程序 26 响应调用返回预览器的类 ID，那么方法 400 从操作 418 转移到操

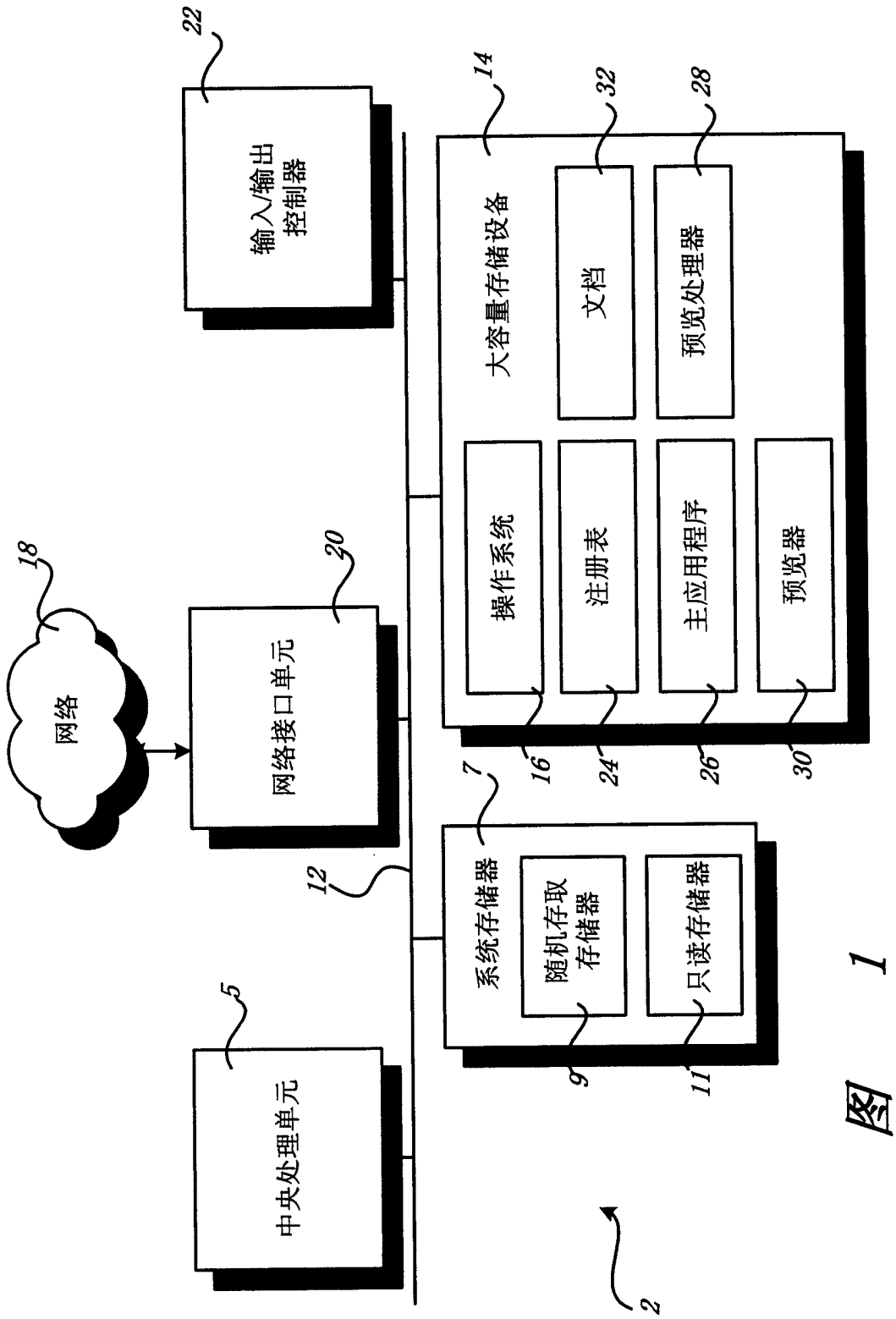
作 414, 其中由类 ID 标识的预览器被指定为当前文件的预览器。否则, 例程 400 从操作 418 处继续到操作 428 处, 在其中返回错误消息。

例程 400 从操作 408、414、420 和 426 继续到操作 430。在操作 430 处, 所指定的预览器被载入为后台线程上的进程外对象。例程 400 接着继续到操作 432, 其中预览器尝试载入文件并建立预览。例程 400 接着继续到操作 434, 其中确定文件是否被成功地载入。如果文件没有被成功载入, 例程 400 从操作 434 转移到操作 438, 在其中返回出错消息。如果文件被成功载入, 例程 400 继续到操作 436, 在其中返回成功消息。例程 400 从操作 436 和 438 继续到操作 440, 在其中返回。

现在参考图 5, 将描述说明性例程 500, 用于卸载预览器和预览处理器 28。例程 500 在操作 502 处开始, 其中预览处理器 28 在预览器上调用卸载方法。预览器卸载文件并撤销对自身的初始化以响应接收到这样的调用。例程 500 接着继续到操作 504, 其中在预览处理器界面 28 上调用撤销初始化方法。这个调用造成预览处理器对自身撤销初始化。例程 500 从操作 504 继续到操作 506, 在其中结束。

基于上述讨论, 应该理解本发明的各个实施例包括提供文档预览的方法、系统、装置和计算机可读介质。上述技术规范、例子和数据提供了对制造和使用本发明的组成部分的完整描述。由于可以不背离本发明的精神和范围作出本发明的许多实施例, 本发明驻留在这里所附的权利要求书中。





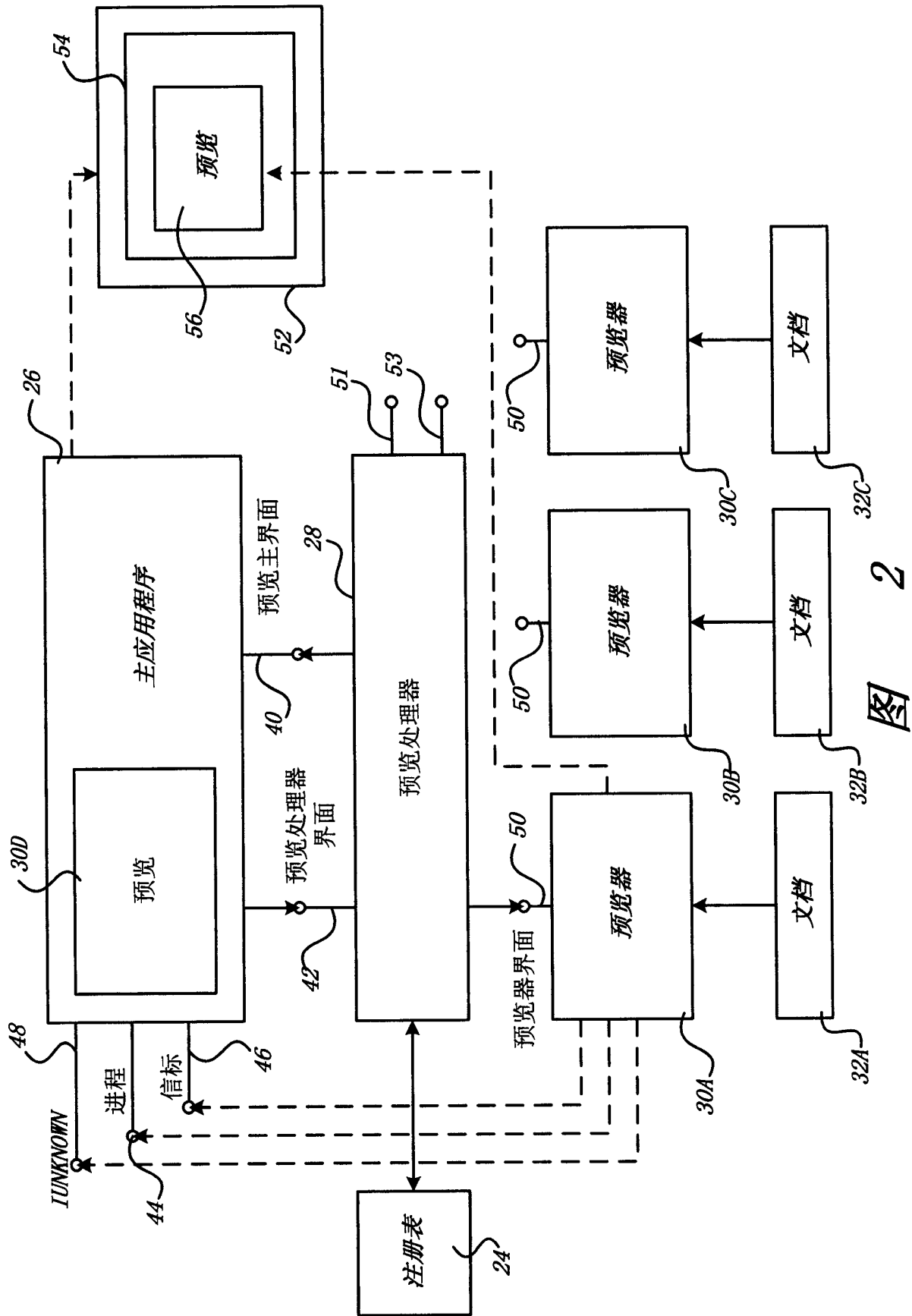


图 2

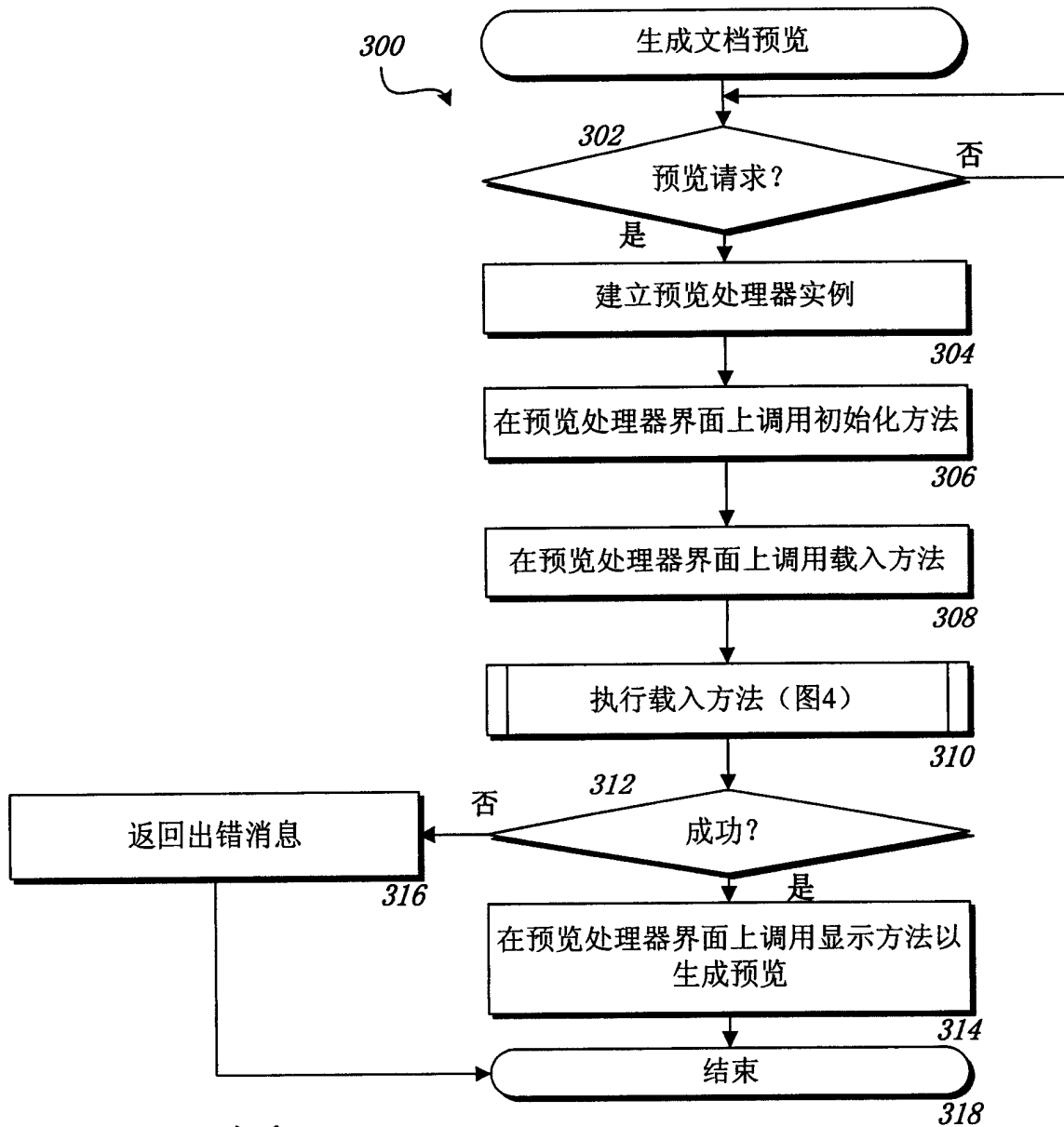


图 3

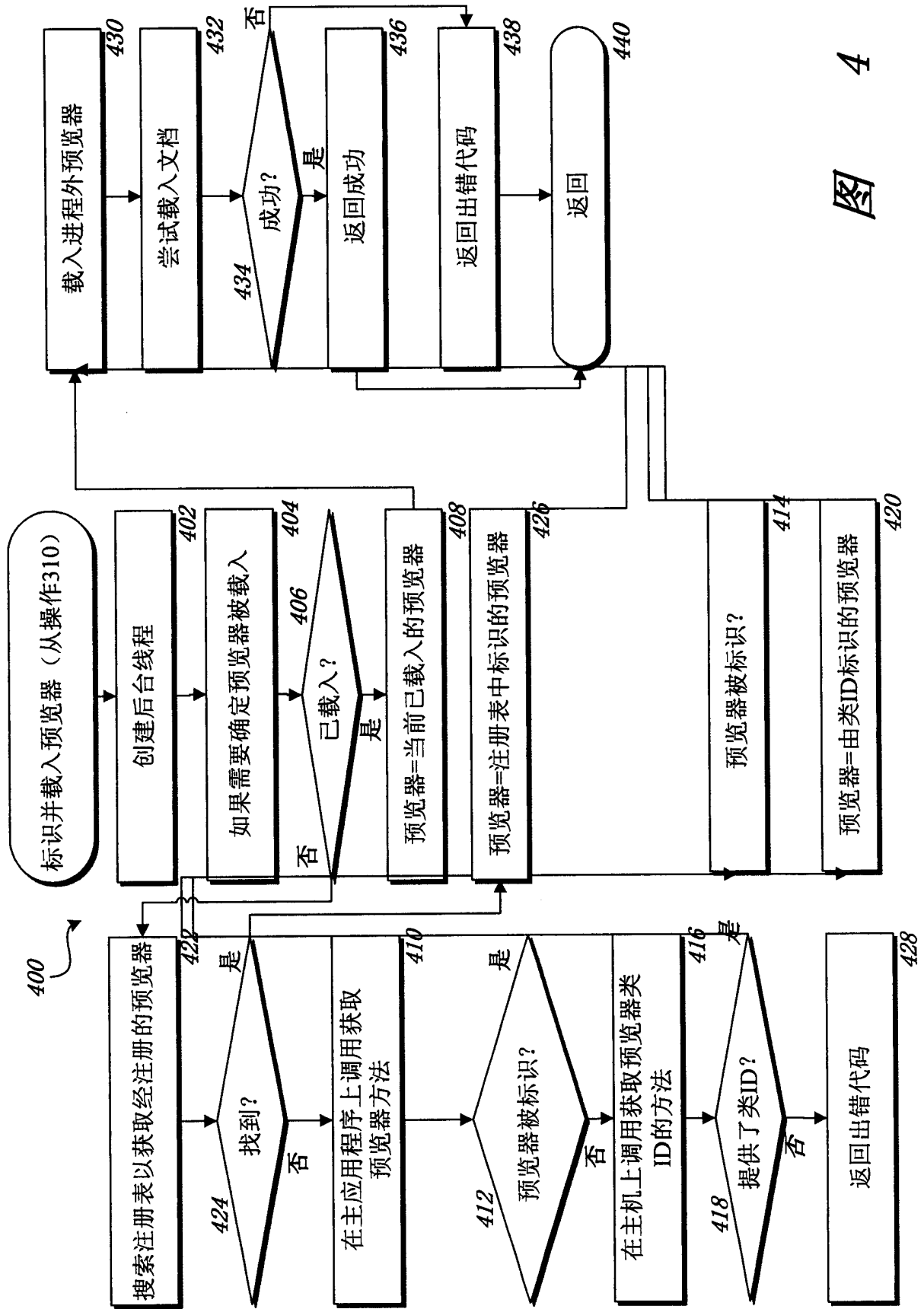


图 4

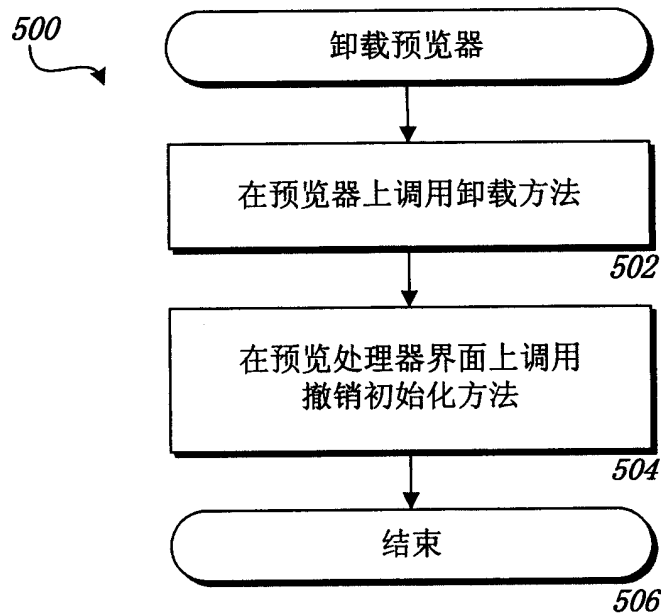


图 5