



[12] 发明专利说明书

专利号 ZL 200410008243.6

[45] 授权公告日 2007 年 10 月 3 日

[11] 授权公告号 CN 100340985C

[22] 申请日 2004.3.1

[21] 申请号 200410008243.6

[30] 优先权

[32] 2003. 2. 28 [33] US [31] 10/377,258

[73] 专利权人 微软公司

地址 美国华盛顿州

[72] 发明人 B·M·琼斯 M·萨维奇

R·A·利特尔 M·穆雷

R·科里 M·塔科

[56] 参考文献

WO02/063494A2 2002.8.15

CN1136847A 1996.11.27

WO03/014965A2 2003.2.20

CN1128372A 1996.8.7

审查员 解欣

[74] 专利代理机构 上海专利商标事务所有限公司

代理人 陈斌

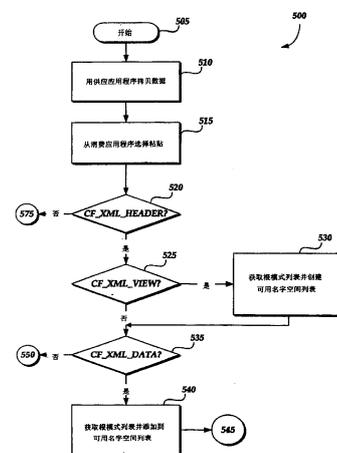
权利要求书 5 页 说明书 19 页 附图 6 页

[54] 发明名称

提高计算机软件应用程序粘贴功能的方法和系统

[57] 摘要

本发明提供了提高计算机软件应用程序可用的粘贴功能的方法和系统，用于粘贴数据到计算机生成的文档。从一个文档选择数据用以粘贴到另一个文档。选取的数据保存到一个存储位置，如剪贴板。随着选择的数据，与之相关的信息或数据也被保存，用于向消费或粘贴软件应用程序提供与所选数据相关的数据类型的信息。相应的，为了保持粘贴数据中数据的结构和格式与粘贴之前应用到该数据的结构和格式一致，消费应用程序可获取一个名字空间和相关的资源，如 XSLT 转换文件，用来将选择的数据从与第一文档关联的一种数据类型转换成第二种数据类型，以便粘贴选择的数据到第二文档。



1. 一种提高计算机软件应用程序粘贴功能的方法，其特征在于，该方法包括：

由供应应用程序打开第一文档；

从第一文档选择数据，用于由消费应用程序粘贴到第二文档；

保存所选择的数据到一存储位置；

保存与所选择的数据相关联的数据到所述存储位置，用于向消费应用程序提供关于所选数据的信息；

如果与所选择的数据相关联的数据标识与所选择的数据相关的第一名字空间，标识与所选择的数据关联的一个或多个数据类型，向消费应用程序提供所述第一名字空间；

选择一个或多个数据类型中的一个用于粘贴所选择的数据到所述第二文档；以及

获取与所选择的数据类型相关联的一个资源用于准备数据以由消费应用程序粘贴到第二文档。

2. 如权利要求1所述的方法，其特征在于，在获取与所选择的数据类型相关联的一个资源用于准备数据以由消费应用程序粘贴到第二文档的步骤之前，解析与所选择的数据相关联的资源的可用名字空间列表；以及如果所述可用名字空间列表标识与所选择的数据相关联的资源，则将所述资源提供给消费应用程序。

3. 如权利要求2所述的方法，其特征在于，

如果所述可用名字空间列表不标识与所选择的数据相关联的资源，那么解析与所选择的数据相关联的资源的名字空间库；以及

如果该名字空间库标识与所选择的数据相关联的资源，那么将所述资源提供给消费应用程序。

4. 如权利要求3所述的方法，其特征在于，

如果多个名字空间由与所选择的数据相关联的数据标识，则获取与所述多个名字空间中的一个相关联的第二资源，用于准备数据由消费应用程序粘贴到第二文档。

5. 如权利要求4所述的方法，其特征在于，所述方法还包括：

在依照第一资源粘贴所选择的数据到第二文档和依照第二资源粘贴所选择的数据到第二文档之间，提供用户可选的选择。

6. 如权利要求4所述的方法，其特征在于，所述方法进一步包括：

依照从多个可用名字空间中的一个选择的一个或多个资源中的一个，向

用户提供可选的选择，用于粘贴所选择的数据到第二文档，每个可用名字空间都与一个或多个资源相关联。

7. 如权利要求 5 所述的方法，其特征在于，所述保存所选择的数据到一存储位置包括以可扩展标记语言 XML 格式保存所选择的数据。

8. 如权利要求 7 所述的方法，其特征在于，所述第一和第二资源包括可扩展格式页语言转换 XSLT，用于将所选择的数据从与第一文档相关联的第一数据类型转化为第二数据类型，用以粘贴所选择的数据到第二文档。

9. 如权利要求 8 所述的方法，其特征在于，如果没有资源与所选择的数据类型相关联用以准备数据以由消费应用程序粘贴到第二文档，则将所选择的数据粘贴到第二文档而不用提高的粘贴功能。

10. 如权利要求 9 所述的方法，其特征在于，所述保存与所选择的数据相关联的数据到所述存储位置包括以 CF_XML_HEADER 格式保存数据到所述存储位置。

11. 如权利要求 10 所述的方法，其特征在于，如果所述第一名字空间标识为与所述第一文档相关联，则向与所选择的数据关联的数据提供信息使消费应用程序能够在可用名字空间列表或名字空间库中定位第一名字空间。

12. 如权利要求 11 所述的方法，其特征在于，所述向与所选择的数据相关联的数据提供信息使消费应用程序能够在可用名字空间列表或名字空间库中定位第一名字空间的步骤包括：为第一名字空间提供版本标识符，为第一名字空间提供统一资源标识符，及为第一名字空间提供文件大小。

13. 权利要求 12 所述的方法，其特征在于，如果所述多个名字空间被标识为与第一文档相关联，则向与所选择的数据相关联的数据提供信息使消费应用程序能够在可用名字空间列表或名字空间库中定位所述多个名字空间。

14. 如权利要求 13 所述的方法，其特征在于，所述向与所选择的数据相关联的数据提供信息使消费应用程序能够在可用名字空间列表或名字空间库中定位所述多个名字空间的步骤包括：为所述多个名字空间的每一个提供版本标识符，为所述多个名字空间的每一个提供统一资源标识符，以及为所述多个名字空间的每一个提供文件大小。

15. 如权利要求 10 所述的方法，其特征在于，如果所述第一名字空间被标识为与第一文档关联，则依照所述第一名字空间向消费应用程序提供与所述第一文档关联的任何数据类型。

16. 如权利要求 15 所述的方法，其特征在于，依照所述第一名字空间与所述第一文档关联的数据类型以 CF_XML_VIEW 格式被提供给消费应用程序。

17. 如权利要求 13 所述的方法，其特征在于，如果所述多个名字空间被标识为与第一档关联，则依照所述多个名字空间向消费应用程序提供一个或多个与第一文档相关联的数据类型。

18. 如权利要求 17 所述的方法，其特征在于，依照所述多个名字空间与所述第一文档相关联的一个或多个数据类型以 CF_XML_DATA 格式被提供给消费应用程序。

19. 如权利要求 1 所述的方法，其特征在于，所述供应应用程序是字处理应用程序。

20. 如权利要求 1 所述的方法，其特征在于，所述供应应用程序是电子表格应用程序。

21. 如权利要求 1 所述的方法，其特征在于，所述供应应用程序是幻灯片展示应用程序。

22. 如权利要求 1 所述的方法，其特征在于，所述消费应用程序是字处理应用程序。

23. 如权利要求 1 所述的方法，其特征在于，所述消费应用程序是电子表格应用程序。

24. 如权利要求 1 所述的方法，其特征在于，所述消费应用程序是幻灯片展示应用程序。

25. 如权利要求 1 所述的方法，其特征在于，所述保存所选择的数据到一存储位置包括保存所选择的数据到剪贴板。

26. 一种提高计算机软件应用程序粘贴功能的方法，其特征在于，该方法包括：

由供应应用程序打开第一文档；

依照标记语言将结构应用到所述第一文档；

从所述第一文档选择数据，用于粘贴到第二文档；

保存所选择的数据到一存储位置；

保存与所选择的数据相关联的数据到所述存储位置用于向消费应用程序提供关于所选择的数据的信息；

由消费应用程序打开第二文档，选择消费应用程序粘贴功能以将所选择的数据粘贴到所述第二文档；

由消费应用程序读取与所选择的数据相关联的数据；

响应于消费应用程序读取与所选择的数据相关联的数据，确定是否有一个或多个名字空间与所选择的数据相关联，定义对于应用到所选择的数据上的结构所许可的数据内容、数据类型和数据结构；

由消费应用程序获取与一个或多个名字空间中的一个相关联的资源用于粘贴所选择的数据到第二文档；以及

根据所述资源粘贴所选择的数据到第二文档。

27. 如权利要求 26 所述的方法，其特征在于，在由消费应用程序获取与一个或多个名字空间中的一个相关联的资源用于粘贴所选择的数据之前，

确定在可用名字空间列表中是否包含消费应用程序使用的一个或多个名字空间中想要的一个；

如果可用名字空间列表包含消费应用程序使用的一个或多个名字空间中想要的一个，则选择所述想要的名字空间；以及

向消费应用程序提供所选择的名字空间。

28. 如权利要求 27 所述的方法，其特征在于，如果所述可用名字空间列表不含有消费应用程序使用的一个或多个名字空间中想要的一个，则确定名字空间库是否含有一个或多个资源用于可供消费应用程序使用的一个或多个名字空间中的一个；

如果所述名字空间库含有用于一个或多个名字空间的一个或多个想要的资源可供消费应用程序使用，则从名字空间库中选择所述想要的资源；以及向消费应用程序提供所选择的名字空间和资源。

29. 如权利要求 28 所述的方法，其特征在于，所述方法进一步包括：提供用户可选的选择，用于依照所选择的名字空间粘贴所选择的数据到第二文档。

30. 如权利要求 29 所述的方法，其特征在于，所述标记语言是可扩展标记语言 XML。

31. 如权利要求 30 所述的方法，其特征在于，所述资源是可扩展格式语言转换 XSLT 文件，用于将所选择的数据从与所述第一文档关联的第一数据类型转换成第二数据类型，以便将所选择的数据粘贴到第二文档。

32. 如权利要求 29 所述的方法，其特征在于，所述保存与所选择的数据相关联的数据到所述存储位置用于向消费应用程序提供关于所选择的数据的信息的步骤包括：识别所述一个或多个名字空间中的任何一个是否与第一文档相关联。

33. 如权利要求 32 所述的方法，其特征在于，所述识别所述一个或多个名字空间中的任何一个是否与第一文档相关联的步骤包括：向消费应用程序提供信息用于在可用名字空间列表或名字空间库中定位所述一个或多个名字空间的任何一个。

34. 一种从拷贝应用程序粘贴数据到消费应用程序的方法，其特征在于，所述方法包括：

应用第一可扩展标记语言 XML 元素到第一文档的一区域的开头，该区域被选择用于从拷贝应用程序进行拷贝，并用于粘贴到消费应用程序；

应用第二 XML 元素到所述选择的区域的末尾；

在与所述选择的区域相关的首部中提供信息，所述信息包括可用 XML 名字空间的标识符和对应于每个可用的名字空间的剪贴板格式标识符的指针；

在首部中提供关于每个可用名字空间的文件大小的信息；以及

提供一个或多个剪贴板上可用的名字空间以使消费应用程序能够从一个或多个名字空间中进行选择，用于粘贴从拷贝程序得到的数据。

提高计算机软件应用程序粘贴功能的方法和系统

(1) 技术领域

本发明涉及提高计算机软件应用程序现有的粘贴功能的方法和系统，用于将数据粘贴到计算机生成的文档。

(2) 背景技术

计算机软件应用程序使用户能够创建各种文档，协助他们工作、学习和休闲。比如，字处理应用程序使用户能够创建信件、文章、书籍、备忘录、等等。电子表格应用程序使用户能够存储、操纵、打印、显示各种包括文字与数字的数据。这些应用程序有着许多众所周知的长处：包括丰富的编辑、格式、打印和计算功能。

众多软件应用程序拥有的一个普通而且有用的功能是从一个给定的文档、电子表格、幻灯片或其他计算机生成的文档中剪切或拷贝数据随后进行一个粘贴操作，将剪切或拷贝的数据粘贴到文档中想要的位置的功能。通常，用户从第一个或称为供应应用程序，如字处理应用程序，剪切或拷贝包括文本、图像或包括文字与数字的数据，然后使用第二或称为消费应用程序，如电子表格应用程序来粘贴这些剪切或拷贝的数据。比如，为了将一组数字粘贴到字处理应用程序，用户可能想要从电子表格应用程序文档中剪切或拷贝这些数字，而在所述字文档中，用户正在准备信件发给客户或同事。典型地，在剪切或拷贝时，数据缓存在某个存储位置中以备在粘贴操作过程中进行后续的粘贴。

当从供应应用程序剪切或拷贝下数据，然后通过第二消费应用程序粘贴到另一个不同的文档而供应应用程序和消费应用程序是两个不同的软件应用程序时，通常第一应用程序提供的许多特性，即使不是全部，包括格式特性在内，都因为第二或消费应用程序不包含提供包括格式在内的专门特性所需的功能而在粘贴过程中丢失了。比如，用户从电子表格应用程序文档拷贝了一组数据，然后粘贴到字处理文档中，粘贴到该字文档的数据可能就不带任何电子表格应用程序中存在的格式。即，如果电子表格应用程序中的数据显示成多行多列，那粘贴到字处理应用程序的数据可能只是简单的没有任何格式的数字序列。

为了满足对软件应用程序更高的功能要求，软件开发者已经开始使用标记语言，诸如可扩展标记语言，以使用户能够注释软件应用程序文档，给文档

提供除了负责创建该文档或与之相关的可视格式的软件应用程序的常规功能之外的有用结构。比如，用户可能想通过自己的字处理应用程序创建一模板文档用来准备给出版商写稿。通过将结构应用到文档，收到文档的出版商可以通过处理文档来使用该文档结构所定义的数据。不幸的是，在从含有结构注释的文档剪切或拷贝数据时，这些结构注释通常都会在粘贴操作中丢失，特别是在由一个不同的消费应用程序进行该项粘贴操作的时候。即使消费应用程序被设计成能够理解和使用标记语言，定义标记语言结构注释的模式或规则，供应应用程序用的与消费应用程序用的可能也会有极大的不同。

本发明正是基于上述和其他方面的考虑而诞生的。

(3)发明内容

发明概述

本发明提供用于提高计算机软件应用程序粘贴功能的方法和系统，用于粘贴数据到计算机生成的文档。如通常所述，从第一应用程序选取数据用以粘贴到第二应用程序文档。选择的数据及其相关信息，诸如名字空间列表等一起被存储到一个存储位置。当选择的数据被粘贴到第二即消费应用程序文档时，消费应用程序检查与所选数据关联的可用名字空间列表，看数据的各种不同表示能否能为粘贴应用程序所理解。粘贴应用程序也可查看是否有像可扩展标记语言（XSL）转换文件这样的资源，用于将所选择的数据转换成更易被第二或消费应用程序使用的格式。寻找支持可用名字空间的资源的一种方法是由消费应用程序检查名字空间库寻找一可用的资源，用于转换数据给消费应用程序使用。如果找不到资源，名字空间不能被粘贴应用程序作本机理解，则根据默认的粘贴资源将数据粘贴到第二应用程序文档。粘贴应用程序也可以有不考虑名字空间而处理任何类型数据的默认方法。

更具体地，本发明提供一种提高计算机软件应用程序粘贴功能的方法。通过供应应用程序打开第一文档，根据诸如可扩展标记语言（XML）等标记语言将结构应用到该第一文档。也有可能第一应用程序只是用作以一种有意义的方式来显示现存的数据。从第一应用程序选择数据用于粘贴到第二应用程序。所选的数据被存储到一存储位置，与之相关的信息也被存储到该位置，用以向消费应用程序提供所选数据的信息。这些信息主要包括名字空间列表。这些名字空间标识拷贝应用程序用来表示数据的不同方法。一个例子就是带有金融数据的电子表格文档。提供的一个名字空间可能用于描述电子表格，另一个名字空间用于描述金融数据。

经消费应用程序打开第二文档，在消费应用程序中选择粘贴功能以将所选择的数据粘贴到该第二文档。消费应用程序读取与所选数据的相关信息。作

为响应，消费应用程序确定是否有一个或多个标识相关数据类型的名字空间能被消费应用程序作本机理解。如果名字空间超过一个，应用程序或者让用户来选择使用哪一个名字空间及相关的资源，或者使用消费应用程序文档中已经存在的最适合该数据的名字空间及相关资源。呈现的选择可能是选择使用哪个名字空间，或对于给定的名字空间使用哪个资源。消费应用程序可能获取一个或多个资源文件，比如与名字空间相关的 XSLT 转换文件，用于在消费应用程序不能理解该名字空间时粘贴所选数据到第二文档。在消费应用程序选择了一个或多个名字空间中的一个用于粘贴所选的数据到第二文档之后，所选的数据就根据消费应用程序最想要的方法粘贴到第二文档。有时使用的其它资源可能是 XSLT 文件，用于将所选的数据转换成第二或消费应用程序可使用的数据类型或格式。

依照本发明的一个方面，在为一个或多个名字空间获得资源用于粘贴数据之前，先要确定可用的名字空间列表中是否包含消费应用程序可理解的一个或多个名字空间中想要的一个。如果可用的名字空间列表包含消费应用程序可理解的一个或多个名字空间中想要的一个，则选择该想要的名字空间，并将它提供给消费应用程序。如果该可用的名字空间列表不包含消费应用程序可理解的一个或多个名字空间中想要的一个，则需要确定一名字空间库中是否包含能够帮助消费应用程序理解供其使用的一个或多个名字空间的资源。如果名字空间库中包含一个能够帮助消费应用程序理解供其使用的一个或多个名字空间的资源，则从名字空间库中选择该想要的资源，并将它提供给消费应用程序帮助它使用该一个或多个名字空间。

与所选数据相关的信息被保存到一存储位置以提供给消费应用程序，它可包括一个指示，用来说明与所选数据相关的一个或多个名字空间。对于和所选数据相关的每一个名字空间，与所选数据相关的信息包括任何相关的名字空间的信息，包括该名字空间的版本标识符、用于定位名字空间的统一资源标识符和该名字空间的文件大小。

根据本发明的另一方面，提供一种方法用于从一拷贝应用程序粘贴数据到一个消费应用程序。第一可扩展标记语言 (XML) 元素被应用到第一文档的区域的开头。该区域被选择用于拷贝应用程序的拷贝和消费应用程序的粘贴。第二 XML 元素被应用到所选区域的末尾。信息在与选择区域相关的首部 (header) 中提供。该信息包括可用 XML 名字空间的身份标识和对应于每个可用名字空间的指向剪贴板格式标识的指针。在文件首部提供关于每个可用名字空间文件大小的信息。剪贴板上提供一个或多个可用的名字空间，使消费应用程序能够从中选择，用于粘贴从拷贝应用程序得到的数据。

这些和其它的特征和优点构成了本发明的特征，将在阅读、浏览下面的详细说明和相关图表后更加清楚。可以理解的是，上述的概要描述和接下来的详细描述对本发明只是示列和解释说明，而不是对权利要求书中的本发明的限制。

(4)附图说明

图 1 所示是提供本发明的示例操作环境的计算机、相关外设和网络设备的块图。

图 2 所示是一简化的块图，说明了供应应用程序和消费应用程序之间的交互，其中数据由供应应用程序进行剪切或拷贝，由消费应用程序进行粘贴。

图 3 所示是一简化的块图，说明了依照本发明的示例实施例，向用户提供增强的粘帖功能的用户接口。

图 4 所示是软件应用程序粘帖和消费从供应应用程序剪切或拷贝得到的数据的屏幕显示。

图 5 和图 6 所示是依照本发明实施例用于提高消费应用程序粘帖功能的方法的流程图。

(5)具体实施方式

以下参照上述的附图们对本发明实施例进行描述，在所有的几幅图中类似的数字标示类似的部分或部件。本发明涉及提高计算机软件应用程序粘帖功能的方法和系统，用于粘帖数据到计算机生成的文档。用户从诸如电子表格的第一应用程序选择数据，粘帖到诸如字的第二处理应用程序。粘帖过程可能是以传统的剪切/拷贝和粘帖操作的形式进行，或以拖和放的形式将选择的数据从第一应用程序文档粘帖到第二应用程序文档。依据本发明的实施例，第一或供应应用程序以可扩展标记语言（XML）表示（representation）的形式将选择的数据写入到一存储位置，如剪贴板。数据写入到存储位置的格式包括所选择的数据和与之相关的一个或多个名字空间的信息。与所选数据相关的可用名字空间可标识出与所选数据相关的数据格式化的类型。例如，标识出的可用名字空间可包括超文本标记语言（HTML）、简历文档或字处理器 XML。HTML 名字空间标识可标识所述数据能被组织成 HTML。简历文档名字空间可标识所述数据能够依照简历 XML 模式文件进行组织。字处理器 XML 名字空间可以标识所述数据能够依照 XML 的字处理器版本格式化。当然，这些名字空间只是可与所选数据相关联的众多类型名字空间的样例。

当通过选择第二程序的粘帖功能或通过“放”数据的方法将所选的数据粘帖到第二应用程序文档时，第二或消费应用程序从所选数据提供的信息中检测出格式化为 XML 的数据正在被粘帖。基于所选数据提供的信息，消费应用程序检查与将要粘帖的数据相关的所有可用名字空间。例如，上述的例子中，消费应用程序检测出所选数据已经关联了 HTML、简历文档或字处理器 XML

的名字空间。相应地，消费应用程序查看可用名字空间列表寻找资源来帮助粘贴选择的数据。例如，如果供应应用程序是电子表格应用程序，消费应用程序查看可用名字空间列表寻找 XSLT 文件，用于将电子表格 XML 格式的数据转换成消费应用程序可用、可消费的格式。如果消费应用程序是字处理器，那么它将查找 XSLT 转换文件，将电子表格数据转换成字处理应用程序可消费的数据。例如，如果在电子表格应用程序中数据被组织成两行三列的形式，消费应用程序使用的 XSLT 转换文件会将这些数据转换成字处理应用程序可消费的数据以保持它两行三列的格式。如果没有本发明的功能，数据会依照消费应用程序默认的粘贴功能进行粘贴，因此可能失去了供应应用程序的格式信息。结果，依据本例，相对于两行三列的格式，该数据会被这样粘贴到字处理应用程序，即作为原始数据在工作区中从左到右地排列。

如果对于消费应用程序标识了多个可用名字空间，那么消费应用程序可从中选择最有可能向其提供想要的粘贴功能的名字空间。或者，向消费应用程序的用户提供用户接口，使用户能够在应用程序可理解的这些可用名字空间的可用转换文件中进行选择。如果消费应用程序在可用名字空间列表中没有找到资源，它将去查找名字空间库寻找可用资源，以帮助消费应用程序在粘贴之前进行数据的转换。如果名字空间库中存在多个资源，将提供用户以接口，使用户能够在可用的资源中进行选择。如果在可用名字空间列表或可用名字空间库中都没有找到可用的资源，那么消费应用程序会依照默认的功能来粘帖数据，如上所述。

操作环境

图 1 和接下来的讨论试图对可能实现本发明的适合的计算环境做一个简单概要的描述。虽然本发明被描述成在通常的运行于个人电脑操作系统上的应用程序上下文环境中，但本领域的熟练技术人员都知道到本发明也可结合其它的程序模块一起来实现。通常，程序模块包括例程、程序、组件、数据结构等等，它们运行特殊的任务或实现特殊的抽象数据类型。而且，本领域内的熟练技术人员将理解，本发明可用其它计算机系统配置实施，包括手持装置、多处理器系统、基于微处理器或可编程的消费者电子器件、手机、小型机、大型机等。本发明也可用于分布式计算环境，在此环境下，依靠通过通信网络连接的远程处理装置来执行任务。在分布式计算环境中，在本地和远程的内存存储装置中都可放置程序模块。

参照图 1，实现本发明的一个示例系统包括常规的个人电脑 20，具有处理单元 21，系统存储器 22，连接系统存储器和处理单元 21 的系统总线 23。系统存储器 22 包括 ROM 24 和 RAM 25。基本输入/输出系统 26 (BIOS) 存储在 ROM 24 中，所述基本输入/输出系统包括例如在启动时帮助在个人电脑的

元件之间进行信息传递的基本例程。个人电脑 20 还包括如硬盘驱动器 27 和磁盘驱动器 28 来读写可移动硬盘 29、光盘驱动器 30 来读取 CD-ROM 盘 31 或读写其它光盘介质。硬盘驱动器 27、磁盘 28 和光盘驱动器 30 各自通过硬盘驱动接口 32、磁盘驱动接口 33、光盘驱动接口 34 连接到系统总线 23。驱动器和相关的计算机可读介质为个人电脑 20 提供了非易失的存储。虽然上述的计算机可读介质指的是硬盘、可移动磁盘和 CD-ROM 盘，但本领域的熟练技术人员将可以理解其它类型的计算机可读介质，如盒式磁带、闪存卡、数字视频光盘、Brnoulli 磁带 (cartridges) 等等，也可用在示例的操作环境中。

许多程序模块可存储在驱动器和 RAM 25 中，包括操作系统 35、一个或多个应用程序 200、220、程序数据，诸如可用名字空间列表 225 和其它程序模块 (未示出)。

用户可通过键盘 40 和定点 (pointer) 装置，如鼠标输入命令或信息到个人电脑。其它输入装置 (未示出) 包括麦克风、操纵杆、游戏手柄、卫星电视天线、扫描仪等等。这些和其它的输入装置通常经连接到系统总线的串行端口 46 连接到处理单元 21，但也可经其它接口进行连接，如游戏端口或通用串行总线。监视器 47 或其它类型的显示设备同样经接口连接到系统总线 23，如视频适配器 48。除了监视器，个人电脑通常包括其它外围输出装置 (未示出)，如扬声器或打印机。

个人电脑 20 可使用到一个或多个远程计算机如远程计算机 49 的逻辑连接而在网络化环境下进行操作。远程计算机 49 可以是服务器、路由器、对等设备或其它普通的网络节点，虽然图 1 中只示出了存储设备 50，但通常它包括所述的个人电脑 20 的许多个或所有的元件。服务器 49 和存储器 50 保存应用程序，如应用程序 20，和数据存储，如名字空间库 230。图 1 描述的逻辑连接包括局域网 51 和广域网 52。这样的网络环境在办公室、企业范围的计算机网络、内联网和因特网中都很常见。

在 LAN 网络环境中使用时，个人电脑 20 通过网络接口 53 连接到 LAN 51。在 WAN 网络环境中使用时，个人电脑 20 通常包括调制解调器 54 或其它在 WAN 52 如因特网上建立通信的装置。调制解调器 54 可为内置或外置的，它通过串行端口接口 46 连接到系统总线 23。在网络环境中，所述的关于个人电脑 20 的程序模块或其中的部分模块存储在远程存储装置中。可以理解的是，所示的网络连接是示例，也可能其它方法来建立通信连接。

操作

图 2 所示是一简化的块图，说明了供应应用程序和消费应用程序之间的交互，其中数据通过供应应用程序进行剪切或拷贝，通过消费应用程进行粘贴。如图 2 所示，示出了供应应用程序 200，剪切或拷贝的数据在其中保存以供消

费应用程序使用。供应应用程序可以是包括字处理、电子表格、幻灯片展示等应用程序在内的任何能够在其中剪切或拷贝数据并用于后续的粘贴的软件应用程序。一旦用户使用供应应用程序 200 来选择数据，进行剪切或拷贝的操作 205，该操作包括从供应应用程序剪切或拷贝所选的数据。一旦剪切或拷贝了选择的数据，那么该数据就被送到存储位置 210，诸如图 2 所示的粘贴板。存储位置 210 保存剪切或拷贝的数据，用于根据用户的选择由原来的供应应用程序 200 或消费应用程序 220 进行后续的粘贴。

依照本发明的实施例，剪贴板 210 是一个视窗操作系统所维护的存储资源的示例。剪贴板存储剪切或拷贝的数据的副本，后续的粘贴操作从剪贴板 210 传递数据到消费程序。正如本领域熟练技术人员所熟道的，倘若第二或消费应用程序 220 能够读取供应应用程序 200 生成的数据，则如上所述，剪贴板 210 可使数据从一个应用程序传递到另一个应用程序。为了确保消费应用程序能够读取、理解和使用供应应用程序 200 剪切或拷贝的数据，通常这些剪切或拷贝的数据以一种通用的格式，如超文本标记语言的格式保存到剪贴板 210，以便在这种格式下数据最有可能为大量的不同种类的消费应用程序 220 所消费。其它类型的格式包括丰富文本格式，或者数据被保存为非格式化的数据。在第一应用程序进行拷贝时，所有这些格式被告知（advertised）剪贴板。然后，由粘贴应用程序决定它将要求使用哪个可用的格式。

一旦启动消费应用程序 220 用于粘贴从供应应用程序 200 剪切或拷贝的数据，选择粘贴操作 215 以粘贴剪切或拷贝的数据到消费应用程序文档中想要的位置。必须理解，剪切/拷贝操作 205 和粘贴操作 215 可能以拖放操作的形式结合起来，从供应应用程序文档拖出所选的数据放到消费应用程序。如果剪切或拷贝的数据依照一定数量的通用格式类型经供应应用程序进行了格式化处理，如上所述，则消费应用程序如所料想能够粘贴剪切或拷贝的数据，但是供应应用程序提供给数据的许多特征，包括特殊的格式，经常会丢失。例如，如果用户在字处理供应应用程序 200 中剪切或拷贝表格形式的数据，之后将数据粘贴到电子表格消费应用程序 220，原先表格形式的数据简单地插入到电子表格内的单个单元，致使电子表格消费应用程序 220 不能如用户所愿对该数据进行操作。

随着可扩展标记语言（XML）等标记语言的出现，通常供应应用程序 200 准备的文档已经使用标记语言的结构进行了注释，以便给文档有用的结构用于管理、呈现和操纵包含在文档中的数据。正如本领域熟练技术人员所熟知的，不同的软件应用程序包含有不同的“本机”程序用于读取、理解和使用诸如 XML 的标记语言。即，相对于消费应用程序 220，供应应用程序 200 可能被设计成提供不同的或较多的 XML 结构和功能。因此，如果从一个供应应用程序 200 创建的 XML 格式的文档剪切或拷贝数据用于另一个不同的消费应

用程序的消费，多数关联于该数据 XML 格式的结构和功能在消费应用程序粘贴数据到由消费应用程序所创建和操作的文档中时会丢失。

以下是一个可应用于由字处理供应应用程序 200 用户书写的文章上的 XML 结构的实例。

XML 结构实例

```
<article>
<title>EXTRA EXTRA</title>
<body>
  <summary>Today in New York...</summary>
  <main>Today in New York...</main>
</body>
  *
  *
  *
</article>
```

如 XML 结构实例所示，例举了许多 XML 元素来注释该“article”文档。例如，在文档的开始和末尾有<article>标签来定义该文档为“article”文档，其它多种元素如<title>和<body>包含在“article”结构内。在<body>元素内是它的两个子元素<summary>和<main>。该文档的 XML 注释使得用户能够定义文档的部分为特定类型的数据和数据结构。例如，用户可定义在<title>元素内文档的部为包含“title”类的的数据并具有“title”指定许可的结构。

为了给文档提供一套语法和数据类型的规则以控制给定文档诸如上面例举的“article”文档所可能包含的数据类型和数据结构，XML 模式被附加或关联到该文档用以提供控制每个 XML 元素和标签的规则，而使用这些元素和标签用户可以注释给定的文档。例如，“article”文档可有一个附加或关联的模式，诸如“article-schema.xsd”用于提供所容许的 XML 元素集，诸如<article>元素、<title>元素、<body>元素等等。该模式包括控制次序的规则，根据该次序将元素应用于文档，还包括与应用到该文档的个别元素相关的特殊规则。例如，附加或关联到“article”文档的模式可以规定与特定元素，如<date>元素（未示出）相关联的数据必须包括一个日元素，随后是一个月 h 元素，再后是一个年元素。另外，该模式可以要求<date>元素必须紧靠在<title>元素之前。

如本领域熟练技术人员所理解的，XML 模式的开发者确定 XML 元素的名字和这些元素所容许的相关数据类型和数据结构。于是，所有根据给定的模式用 XML 结构来注释的文档的用户都可使用 XML 结构中包含的数据，而不需要考虑该文档的总体类型和结构。例如，如果上述的“article”文档被送到该文档的出版商，该出版商可能会开发出分析该文档的软件应用程序，以便

在文档中来查找特定数据类型供出版商使用。例如，出版商可能只想发布文章的标题作为整篇文章将来出版的一则广告。标识 XML 的方法是通过名字空间。名字空间提供了一个标识（ID），从而 XML 的任何消费者将会知道他们消费的是何种类型的 XML。通过查看名字空间，消费应用程序将会知道创建该 XML 文件使用了哪个模式。使用文档中指定的名字空间，出版商将知道与 XML 元素<title>相关的数据是依照控制文档的模式被准备的。因此，出版商可开发软件应用程序或 XSLT 转换文件用于定位<title>元素并提取相关的数据，用于插入到出版商自己的文档（包括用某种特殊的方法来格式化文档）作为整篇文章将来出版的一则广告。

接着这个例子，许多不同的出版商可订阅同样的名字空间，以记录与“article”文档相关的规则，以便每个出版商可从文档的作者接收到这个“article”文档，并且依照组织该数据的 XML 元素来使用包含在文档中的数据。即，第一出版社可能只对包含在<summary>元素中的数据感兴趣，而第二出版商只对抽取<title>元素中的数据感兴趣。依照 XML 结构，通过使用自己的软件应用程序或 XSLT 转换文件来定位想要的元素，则每个出版商可不考虑文档的其它方面或元素而抽取想要的元素。之所以能够做到这一点，是因为文档的每个用户遵循着添加到或关联到该文档的名字空间中指定的数据类型和数据结构的规则。

如果消费应用程序确实找到了可用于可用名字空间之一的 XSLT 转换文件，则所选的数据经过 XSLT 转换文件的转换，消费应用程序将该结果作为粘贴功能的一部分来接收。或许有这种情况，即消费应用程序注意到了转换文件，它消费可用名字空间，自己对该数据进行转换。例如，如上所述，如果供应应用程序是一电子表格应用程序，所选的数据是两行三列的格式，那么用于转换电子表格数据供字处理应用程序消费的 XSLT 转换文件能够，例如使该数据得以转换，以便字处理应用程序能粘贴成两列，每列三项，而不是没有格式的原数据或电子表格所使用的表格形式。应当理解，第一或供应应用程序文档可根据传统的 XML 模式进行标记。例如，数据可依照“股票价格模式”（“stock price schema”）进行标记。在这种情况下，如果从电子表格应用程序拷贝了选择的数据粘贴到字处理应用程序，例如，在将选择的数据写入内存用于粘贴到字处理应用程序时，“股票价格模式”相关的名字空间和电子表格可视布局相关的名字空间都将被关联到所选的数据。

在识别出“股票价格模式”后，消费应用程序参考可用资源列表，探知所选数据的每个单元格是公司股票的符号，对于每个公司有公司的相关描述。因此，消费应用程序将在可用名字空间列表或名字空间库中查找诸如 XSLT 转换文件的资源来指导消费应用程序怎样粘贴标记的数据。提供给消费应用程序用户的用户接口可提供选项“粘贴公司概况”。如果用户选择了“粘贴公

司概况”，与粘贴功能相关的 XSLT 转换文件可使消费应用程序能够粘贴从供应应用程序选择得到的公司股票符号，同时粘贴经用户选择的粘贴功能从与所选数据关联的可用名字空间中引入公司概况。这种解决方法优于之前的替换方案，该方案是选择电子表格名字空间，采用与电子表格中相同的形式来粘贴数据。

如熟悉可扩展标记语言的人所了解，XML 是通过将元素和属性名称关联到由统一资源标识符所标识的名字空间的方法来限定 XML 文档所使用的元素和属性名称。XML 名字空间是名字集，由统一资源标识符参照标识，名字作为元素的类型和属性名称用于 XML 文档。单个 XML 文档可包含为多软件模块定义和使用的元素和属性。例如，依照本发明的实施例，单个 XML 文档，诸如字处理文档可包含为多软件模块定义和使用的元素和属性。例如，一个字处理文档可有为 HTML 处理模块、字处理应用程序 XML 处理模块定义和使用的元素和属性，或者该文档可包含为与其关联的一个或多个模式文件定义、使用或关联的元素和属性。例如，将元素和属性关联到字处理文档使其与一个模式文件相关联，所述模式文件与简历、法律文件等等文档关联。因此，单一文档，诸如典型的字处理文档可以有一个名字空间标识与每个不同软件模块关联的元素类型和属性名称，所述模块可消费或使用从文档得到的数据。接着上面列出的例子，字处理文档可包含一个与 HTML 处理模块关联的名字空间、一个与字处理 XML 处理模块关联的名字空间和一个与简历或法律文档模式文件关联的名字空间。

依照本发明的实施例，当选择了数据用于粘贴到第二个应用程序文档时，与之相关的标识名字空间的信息被提供给所选择的数据，用于通知第二或消费应用程序与数据相关的名字空间，以便第二或消费应用程序可根据其编程语言对数据关联的元素类型和/或属性进行操作。例如，如果第一应用程序是一个字处理应用程序，第二应用程序是一个电子表格应用程序，电子表格应用程序可使用由数据相关的名字空间标识的信息来确定通过使用与元素类型和属性关联的 XSLT 转换文件所选的数据将被最优地粘贴到第二应用程序文档，所述元素类型和属性与一个给定的名字空间相关。例如，如果一个给定的名字空间标识了与包含在所选数据中的操纵数据相关的元素类型和属性，而其它名字空间关联用于显示所选的数据，第二应用程序即本例中的电子表格应用程序将选择与操纵所选数据相关的名字空间，而不是与显示相关的名字空间。

在消费应用程序检查了从第一应用程序文档选择的 XML 数据相关的名字空间之后，依照本发明的实施例，为了得到最优消费，消费应用程序查看可用名字空间列表或名字空间库，寻找消费应用程序可用的资源用于转换所选的数据。例如，如果第二程序是一个试图粘贴从字处理应用程序选择的数据

的电子表格应用程序，则该第二应用程序将查看可用名字空间列表或名字空间库以获取一个可被第二应用程序使用的 XSLT 转换文件，用于转换字处理器的 XML 数据到电子表格的 XML 数据以供电子表格应用程序作最优消费。如本领域熟练技术人员所了解，可扩展样式表语言（XSL）包括用于说明数据格式的 XML 词汇表。XML 用 XSL 信息文件来指定 XML 文档的样式，描述文档是怎样从一个 XML 文档转换成另一个 XML 文档或其它类型的文档如 HTML 文档的。

仍旧参照图 2，名字空间可被添加到供应应用程序准备的文档，或被维持在该文档可访问的另外一个位置，如名字空间库 230 中。为定位和/或标识该名字空间，文档可包含一个文件路径指针或唯一名字空间标识符（例如，统一资源标示符或统一资源名称）。有关名字空间库 230 的详细描述，可参见 2002 年 6 月 27 日申请的题为“System and method for providing namespace related information”的美国专利申请，序列号为 10/184, 190，在此将其全文作为参照引用。有关从名字空间库 230 下载一个名字空间供供应应用程序 200 和消费应用程序 220 使用的方法的详细描述，可参见 2002 年 5 月申请的题为“Mechanism doe Downloading Software Components from a Remote Application for Use by a Local Software Application”的美国专利申请，序列号为 10/164, 260，在此将其全文作为参照引用。

一旦用户下载或通过其它方式得到一个名字空间及其相关的用户文档信息，该名字空间可被装载到一个包括该名字空间及其资源，以及其它已经装载的名字空间的名字空间列表中。该列表 225，由用户计算机 20 维持或用户可通过分布式计算环境从远程位置，如服务器 49 进行访问。可用名字空间列表包括与供应应用程序中数据相关的名字空间，该名字空间包括使消费应用程序能够定位和得到资源诸如用于帮助消费应用程序转换数据供其消费的 XSLT 转换文件的信息。

考虑用户使用字处理供应应用程序 200 剪切或拷贝表格式数据的实例。在剪切或拷贝数据前，依照供应应用程序 200 的本机 XML 编程语言，用 XML 注释来组织数据。如果通过粘贴操作 215，消费应用程序 220 要求粘贴从剪贴板 210 剪切或拷贝的数据，则消费应用程序 220 将访问可用名字空间列表 225 查看是否有可用的名字空间或对应于其中一个名字空间是否有合适的 XSLT 转换文件，用于增强消费应用程序 220 的本机 XML 的编程，以便消费应用程序 220 以与在供应应用程序 200 中所应用到数据上的同样的格式（如表格的形式）粘贴从字处理供应程序 200 中剪切或拷贝的数据。例如，在可用列表 225 中找到诸如“wordprocessor-to-spreadshheet-schema.xsl”这样的资源文件（该例中是 XSLT 转换文件）供消费应用程序将可用的字处理器名字空间转换到被其理解的电子表格名字空间。因此，当消费应用程序得到附加 XSLT 转换文件，

消费应用程序的本机 XML 编程得到增强,使得消费应用程序(如,电子表格应用程序)现在能够依照从可用名字空间列表得到的名字空间转换来使用从供应应用程序得到的数据。即,为了处理从供应应用程序传来的数据,通过从可用名字空间列表得到 XML 名字空间,消费应用程序接收以与供应应用程序创建和处理该数据同样的方式处理从供应应用程序接收到的数据而所需要的语法和数据规则。

依照本发明的实施例,如果依照一个给定的 XML 名字空间使用供应应用程序来组织数据,用户经由消费应用程序依照一个选定的第三方 XSLT 转换文件可将剪切或拷贝的数据粘贴到该消费应用程序。例如,消费应用程序 220 的一个用户接收到从上述“article”文档剪切的数据,消费应用程序 220 的用户可在消费应用程序中查看可用名字空间列表的资源以确定是否有与多种出版商相关联的转换文件可用于依照一特定出版商的要求来组织该文档。然后,该消费应用程序 220 的用户在可用名字空间列表 225 的资源中查找能够使消费应用程序 220 依照想要的名字空间来使用该数据的转换文件。如果想要的资源不在可用名字空间列表 225 中,用户可查看本机或远程的名字空间库 230 来寻找所提供的名字空间的附加资源供消费应用程序使用以对从供应应用程序 200 得到的数据加以利用。

图 3 所示是一简化的块图,说明了依照本发明的示例实施例向用户提供增强的粘贴功能的用户接口。一旦消费应用程序 220 的用户选择了消费应用程序 220 的粘贴操作,可以向用户提供多种粘贴选项。例如,用户可选择简单粘贴操作将从供应应用程序剪切或拷贝的数据依照默认的格式,如 HTML 格式粘贴到消费应用程序正在使用的文档上。如果想要依照本发明实施例的增强的粘贴功能,用户可选择备用粘贴功能,如图 3 示出的特殊粘贴功能。

特殊粘贴用户接口 310 表明了用户可用的粘贴选项列表。例如,用户可选择以丰富文本格式(RTF)粘贴数据,或者用户可选择依照 HTML 格式 320 粘贴文档。如用户接口 310 所示,用户可选择依照 XML 格式 325 来粘贴数据而不使用本发明增强的粘贴功能。即,消费应用程序 220 依照其本机 XML 编程语言粘贴剪切或拷贝的数据。如果用户想要依照某个在此所述的增强功能来粘贴数据,用户可以选择用户接口 310 的“其它”按钮来启动一个可用模式用户接口 350。通过可用模式用户接口 350,用户可以选择可用列表 355 来查看在可用名字空间列表 225 上哪些名字空间及相关的 XSLT 文件可用,如以上参照图 2 所述。因此,特定转换文件的选择向用户正在使用的消费应用程序提供了所选 XSLT 文件增强的功能,用于对从供应应用程序剪切或拷贝的数据进行操作,如同该数据已被供应应用程序所使用或格式化一样。

如以上参照图 3 所述,如果可用名字空间列表中没有想要的可用名字空间或其它资源,为了提供从本机或远程名字空间库 230 可得到的更大的可用数

据视图列表，应用程序也可启动名字空间特征 370。如果用户通过用户接口 370 从名字空间库 230 中选择一个可用视图，则带有可用粘贴功能更新列表的特殊粘贴用户接口 310 就被提供给用户，包括由所选名字空间激活的粘贴功能。应当理解，某种情况下的视图可能是名字空间及其相关 XSLT 转换的任何组合。XSLT 转换将把可用名字空间转换成一个消费应用程序可理解的本机名字空间。不同的名字空间通常代表了不同类型的数据，不同的 XSLT 代表对不同类型数据的不同的视图。如图 3 所示，从名字空间库用户接口 370 中得到的“出版商 A”XSLT 380 的选择使得特殊粘贴用户接口 310 增加(populated with)了对“出版商 A”的粘贴功能 390。因此，为了使用下载的“出版商 A”的转换文件 380 使消费应用程序 220 能够依照出版商 A 提供的 XSLT 文件粘贴从供应应用程序 200 得到的数据，用户可选择“出版商 A”的粘贴功能 390。该转换被标识为与将出版商 A 名字空间转化成消费应用程序可理解的名字空间相关联。如本领域熟练技术人员所理解，参照图 2 和图 3 在此描述的具体的 XSLT 文件只是样例，任何转换或解释特定的名字空间的不同 XSLT 文件或其它方法都可提供给消费应用程序 220，用以向供应应用程序 200 中剪切或拷贝的数据提供增强的粘贴功能。

图 4 所示是软件应用程序粘贴和消费从供应应用程序得到的数据的屏幕显示。屏幕照 400 显示了一个示例字处理应用程序，用户正在其中制作文档 410。依照本发明的实施例，用户已经将数据粘贴到了一张从电子表格供应应用程序 200 拷贝过来的表内。在粘贴数据到文档 410 的操作中，呈现一对话框 420 向用户显示三种不同的数据格式，包括“八月历史”(Eight Month History) 格式 425，“项目财务总结”(Projected Financial Summary) 格式 430 和“工业分析”(Industrial Breakdown) 格式 435。应当理解，对话框 420 是图 3 所示的对话框 310 的替换。根据图 4 中的数据实例，用户已经选择了“八月历史”格式 425，该格式是可用名字空间列表 225 中的一个名字空间的视图，来将从供应应用程序(如电子表格应用程序)得到的数据在在文档 410 中依照记录了(dictating)“八月历史”的 XML 结构的 XSLT 文件进行格式化。应当理解，诸如“eightmonthhistory-shema.xml”的转换资源可从能使消费应用程序支持 XML 结构来提供“表 1”格式化的可用名字空间列表或名字空间库 225 得到，如图 4 所示。假如用户已经从供应应用程序选择了剪切和拷贝关于“工业分析”格式的信息，则用户可选择“工业分析”格式 435 以便将从供应应用程序剪切或拷贝的数据可依照由供应应用程序对数据进行组织的格式在消费应用程序中进行格式化。

依照本发明的实施例，当有数据以 XML 的格式剪切或拷贝并保存到剪贴板 210 用于后续消费应用程序 220 的粘贴时，可使用三个或更多的剪贴板格式来帮助消费应用程序确定怎样根据本发明的功能来处理剪切或拷贝的数

据。这三种格式包括 CF_XML_HEADER、CF_XML_VIEW 和 CF_XML_DATA。如上所述，包括供应应用程序 200 和消费应用程序 220 在内的许多应用程序支持由这些应用程序所创建的 XML 标记结构和文件。但是，如上所述，通常一个应用程序的 XML 支持和另一个应用程序的 XML 支持是不兼容的，结果是，在将数据从一个应用程序粘贴到消费应用程序文件时，格式和其它 XML 相关的特性丢失了。另外，如上所述，XML 模式也作为由第三方提供的供多种应用程序使用的工具而存在，但它却不被应用程序的本机 XML 编程语言支持。

CF_XML_HEADER 格式（“HEADER”）包含有关拷贝应用程序有哪个名字空间可用的信息。所有的粘贴应用程序可在 CF_XML_HEADER 中查看、决定是否有一个或多个名字空间可供其消费。HEADER 还标识了可以在哪里找到不同的名字空间，以便在确定想要消费的一个或多个名字空间时，粘贴应用程序可以回来查找得到所要的数据。HEADER 可以识别为包含想要的一个或多个名字空间之一处是 CF_XML_DATA 格式。这些信息对消费应用程序很有用，因为消费应用程序能够确定希望消费哪种格式。这有助于解决性能问题，因为在这些具体格式被要求之前，拷贝应用程序不需要用不同的格式来提供所有的数据。在余下部分的描述中，我们假定有两个格式可用。

第一个粘贴板格式是具有一个或多个名字空间的 CF_XML_VIEW，它通常更多的是格式标记而不是数据标记（诸如“SpreadsheetML”；或 documentML）。第二个粘贴板格式是含有一个或多个名字空间的 CF_XML_DATA，它没有定义格式或视图，而是定义了其下置的数据。通过涉及电子表格程序的一个实例，很容易看出这两个格式的活动。如果该电子表格程序支持 XML 结构，那么可以想象定义金融数据的名字空间可被应用到该电子表格用以标识不同的单元格及它们同金融数据的相关程度。当电子表格应用程序进行一项拷贝操作时，它将有两种表示数据的方法。其第一种表示方式就如同它出现在电子表格中，这要使用电子表格标记语言。该电子表格标记语言定义在电子表格中看到的一般情况，如格式、布局等等。这些数据将被放置到 CF_XML_VIEW 剪贴板格式，因为它最好地定义了数据“看上去”怎样。如果粘贴数据的人更多地关心外观和布局，而不是潜在的数据意义，那么这对他们非常的有用。

第二种在剪贴板上表示数据的方法是使用金融数据的名字空间。该名字空间不含涉及数据外观或布局的信息，代之以描述数据本身的意义。该数据将放置到 CF_XML_DATA 剪贴板格式。因为布局信息的不可得，消费应用程序需要自己确定怎样最好地显示这些数据。例如，在粘贴数据到字处理应用程序的例子中这可能是更想要的格式。原因是金融数据在电子表格中的显示方式可能与字处理应用程序中的显示方式完全不同。字处理应用程序需要查看金融数据名字空间，找到一些视图信息，如 XSLT 转换，来帮助字处理应用程

序以正确的方式来显示数据。

通过将不同的名字空间拆分成不同的剪贴板格式将更加高效，因为消费应用程序可能不需要或不同时需要供应应用程序的“VIEW”或“DATA”格式。即，如果基于用户选择的粘贴功能，消费应用程序确定不需要“VIEW”或“DATA”格式中包含的信息，那么消费应用程序不会从供它使用的这些格式装载数据。如果消费应用程序是和供应应用程序同样类型的程序（如，字处理应用程序），消费应用程序不需要另外的粘贴功能来进行用户选择的这个粘贴操作。CF_XML_HEADER 格式包含有关“VIEW”和“DATA”格式的版本和名字空间的信息。HEADER 格式也包括有关每个 VIEW 和 DATA 格式文件的文件大小的信息。下面是一个与用第三方“Get A Job.com”XML 模式进行标记从字处理文件拷贝来的数据片段相关联的 CF_XML_HEADER 格式实例。

```
<cfh:header cfh:Version="1.0" xmlns:cfh="urn:schemas-microsoft-com:office:clipboard:header">
  <cfh:data cfh:bytes="00000001343" >
    <cfh:Item cfh:name="D1" cfh:bytes="00000001343" >
      <cfh:URI cfh:value="urn:schemas-getajob-com:resume" cfh:root="Yes"
        />
    </cfh:Item>
  </cfh:data>
  <cfh:view cfh:bytes="00000003612">
    <cfh:Item cfh:name="V1" cfh:bytes="00000003612">
      <cfh:URI cfh:value="urn:schemas-microsoft-com:office:字.1.0"
        cfh:root="Yes" />
      <cfh:URI cfh:value="urn:schemas-microsoft-com:schemaLibrary"/>
      <cfh:URI cfh:value="urn:schemas-microsoft-com:AML"/>
    </cfh:Item>
  </cfh:view>
</cfh:header>
```

CF_XML_VIEW 数据格式（“VIEW”）向消费应用程序提供关于供应应用程序的本机 XML 编程语言的信息。通过读取包含在“VIEW”格式中的数据，消费应用程序依照自身的 XML 编程语言确定在粘贴操作的过程中，它是否能够完全使用供应应用程序的 XML 编程语言，而不需要获取额外 XSLT 转换文件。下面是一个可与从供应应用程序剪切或拷贝的数据相关联的

CF_XML_VIEW HEADER 格式的实例。

```
<cb:view xmlns:cb="urn:schemas-microsoft-com:office:clipboard:view"><!--
Root element for the ClipBoard-->
```

```
<cb:item cb:name="V0" cb:URI="The URI of the root schema">
```

注：依照一实施例，这里可以有数据。因为该数据处于 `cb:start` 标签之前，消费应用程序知道该数据不是所拷贝的一部分，但它仍然可能提供有关整个来源的性质方面的有趣信息。

```
<cb:start/><!--A single tag to be placed inline signifying the start of the
fragment-->
```

注：依照一实施例，这里可以有数据。因为该数据处于 `cb:start` 标签之后，消费应用程序知道该数据不是所拷贝的一部分。

```
<cb:end/><!--A single tag to be placed inline signifying the end of the
fragment-->
```

注：依照一实施例，这里可以有数据。因为该数据处于 `cb:end` 标签之前，消费应用程序知道该数据不是所拷贝的一部分，但它仍然可能提供有关整个来源的性质方面的有趣信息。

```
</cb:item>
```

```
<!--It is possible to place more than one if the pasting app decides to:-- >
```

```
<cb:item cb:name="V1" cb:URI="The URI of th root schema">
```

```
<cb:start/><!--A single tag to be placed nline signifying the start of the
fragment-->
```

```
<cb:end/><!--A single tag to be placed inline signifying the end of the
fragment-->
```

```
</cb:item>
```

```
</cb:view>
```

CF_XML_DATA 数据格式（“DATA”）向消费应用程序提供与供应应用程序中剪切或拷贝的数据相关联的 XML 名字空间的信息，这些 XML 名字空间不是供应应用程序本机 XML 编程语言的一部分。例如，“DATA”的 HEADER 向拷贝应用程序提供有关与供应应用程序用户创建的模式相关联的 XML 名字空间的信息，或是提供与第三方模式关联的 XML 名字空间的信息，如为了给用户消费或表示用于组织数据的出版商或股票市场公司模式，如上所述。下面是一个可由消费应用程序接收用以提供与供应应用程序剪切或拷贝的数据相关联的非本机 XML 名字空间的信息的 CF_XML_DATA 格式的实例。

```
<cb:data xmlns:cb="urn:schemas-microsoft-com:office:clipboard:data"><!--
```

Root element for the Clipboard-->

```
<cb:item cb:name="D0" cb:URI="The URI of the root schema">
```

注：这里可以有数据。因为该数据处于 `cb:start` 标签之前，消费应用程序知道该数据不是所拷贝的一部分，但它仍然可能提供有关整个来源的性质方面的有趣信息。

```
<cb:start/><!--A single tag to be placed inline signifying the start of the
fragment-->
```

注：这里可以有数据。因为该数据处于 `cb:start` 标签之后，消费应用程序知道该数据不是所拷贝的一部分。`<cb:end/><!--A single tag to be placed inline signifying the end of the fragment-->`

注：这里可以有数据。因为该数据处于 `cb:end` 标签之后，消费应用程序知道该数据不是所拷贝的一部分，但它仍然可能提供有关整个来源的性质方面的有趣信息。

```
</cb:item>
```

```
<!--It is possible to place more than one if the pasting app decides to:-- >
```

```
<cb:item cb:name="D1" cb:URI="The URI of th root schema">
```

```
<cb:start/><!--A single tag to be placed nline signifying the start of the
fragment-->
```

```
<cb:end/><!--A single tag to be placed inline signifying the end of the
fragment-->
```

```
</cb:item>
```

```
</cb:data>
```

图 5 和图 6 是依照本发明实施例的流程图，说明了用于提高消费应用程序粘贴功能的方法。该方法 500 从开始步骤 505 起始，然后转到步骤 510，供应应用程序中的数据被剪切或拷贝并保存到剪贴板 210 用于消费应用程序 220 的后续粘贴。在步骤 515 处，为了将剪切或拷贝的数据粘贴到正在使用的消费应用程序 220 内的一个文档，用户从消费应用程序选择粘贴操作。在步骤 520 处，消费应用程序确定是否有 `CF_XML_HEADER` 格式与剪切或拷贝的数据相关联。如果没有可用的 `CF_XML_HEADER` 格式，该方法前进到图 6 的步骤 575，数据被粘贴而不使用任何增强的粘贴功能。应当理解，如上所述，如果消费应用程序有附加的 XML 模式和/或数据结构可用，则该信息就会通过 `CF_XML_HEADER` 格式被提供给消费应用程序。

如果在 520 步提供“HEADER”格式，则该方法前进到步骤 525，确定该数据是否包括 `CF_XML_VIEW` 格式。如果是，那么该方法前进到步骤 530，消费应用程序创建首部 HEADER 中规定的可用名字空间列表作为可用名字空

间用于粘贴剪切或拷贝数据。例如，如果供应应用程序是一个电子表格应用程序，消费应用程序可能需要查询名字空间库以寻找可用的转换文件或其它资源用于支持消费应用程序（例如，字处理器）依照供应应用程序的 XML 功能粘贴剪切或拷贝的数据。查找名字空间库的原因是消费应用程序并不理解这个可用的名字空间，需要找到关于怎么处理它的附加信息。如果 XML 名字空间库不在可用名字空间列表中，该方法前进到步骤 550，消费应用程序检查名字空间库 230 寻找附加资源用于支持粘贴从供应应用程序剪切或粘贴的数据。

如果在步骤 535 提供 CF_XML_DATA 格式，该方法前进到步骤 540，消费应用程序查询可用名字空间列表获取与供应应用程序创建的或从供应应用程序外部的第三方资源接收的名字空间相关联的转换文件和/或其它的资源，如上所述。应当理解，参照步骤 530 和 540 描述的查询可用名字空间列表的操作可以通过提供用户接口来完成，如以上参照图 3 所述。

在图 6 的步骤 545 处，确定消费应用程序是否理解为从供应应用程序剪切或拷贝的数据作标识的名字空间和关联的转换文件。例如，如果消费应用程序是一个字处理应用程序，消费应用程序可能不需要转换文件形式的附加资源来提供与剪切或拷贝的数据关联的 XML 格式。如果是这样，该方法前进到步骤 575，消费应用程序粘贴剪切或拷贝的数据而不使用粘贴功能的增强。

如果消费应用程序不理解与所选数据关联的名字空间及相关的资源，或者如果名字空间或资源对消费应用程序都不可用，那么该方法前进到步骤 550，用户可用名字空间库 230 来选择与包含在从供应应用程序得到的 CF_XML_DATA 格式中的数据关联的附加名字空间及资源。如果从供应应用程序得到的 CF_XML_DATA 格式标示与剪切或拷贝的数据相关联的第三方名字空间，诸如一个给定出版商的 XSLT 文件，那么消费应用程序依照这个标示的名字空间粘贴数据所需要的 XSLT 文件可从名字空间库 230 中得到并用于消费应用程序。在步骤 555，确定用户是否选择了任何可用的转换文件供消费应用程序使用。如果不是，该方法前进到步骤 560，用户接口 310 的粘贴功能由默认的功能填入，如图 3 所述。

如果资源，诸如 XSLT 转换文件，位于可用名字空间列表或名字空间库，那么该方法前进到步骤 565，粘贴用户接口 310 填入由用户选择的附加功能，如以上参照图 3 所述。在步骤 570，确定是否有用户选择默认的通用粘贴功能或是否有用户选择了特殊粘贴功能。如果用户想要使用一个特殊粘贴功能，该方法前进到步骤 580，用户选择这个特殊粘贴功能。该方法前进到步骤 575，在此依照用户选择的粘贴功能粘贴数据。如果在步骤 570 用户不想选择特殊粘贴功能，该方法前进到步骤 575，如用户所选，数据被粘贴而使用任何增强的粘贴功能。该方法在步骤 595 终止。

如此所述，提供了提高计算机软件应用程序粘贴功能的方法和系统，用于粘贴数据到计算机生成的文档。对于本领域内的熟练技术人员而言，在本发明中可以做各种修改或变化而不脱离本发明的精神实质和范围。研读在此揭示的本发明的说明书和对本发明加以实践，本发明的其它实施例对本领域内的熟练技术人员是显而易见的。

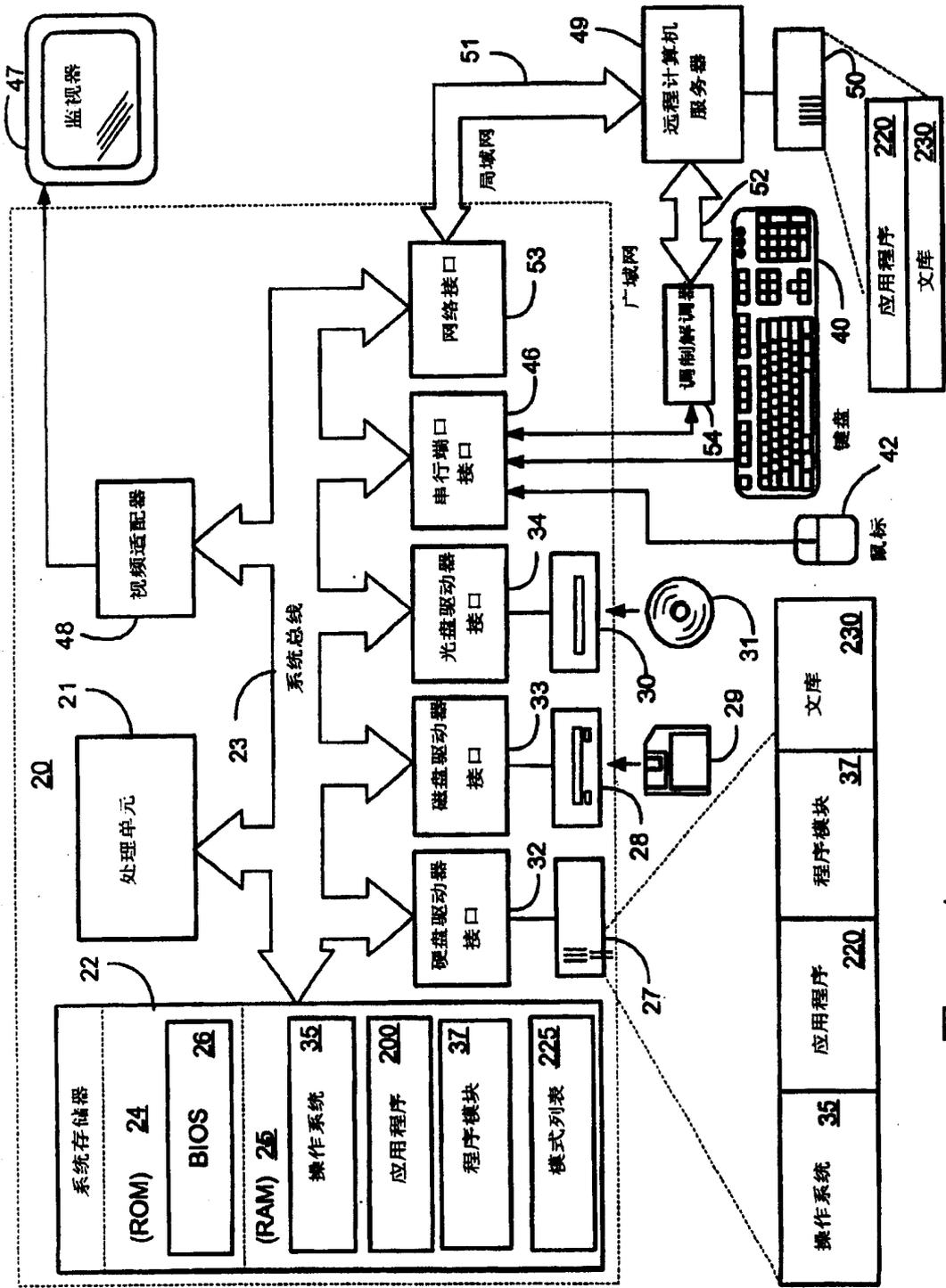


图 1

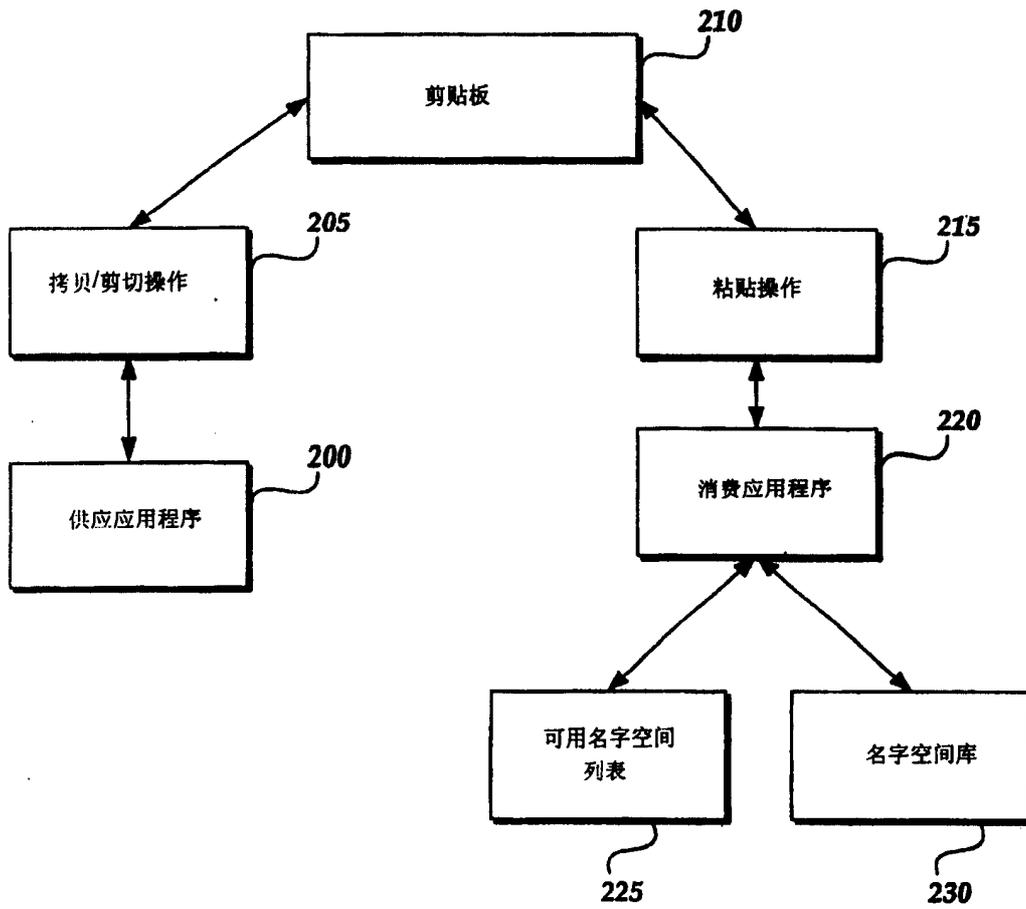


图 2

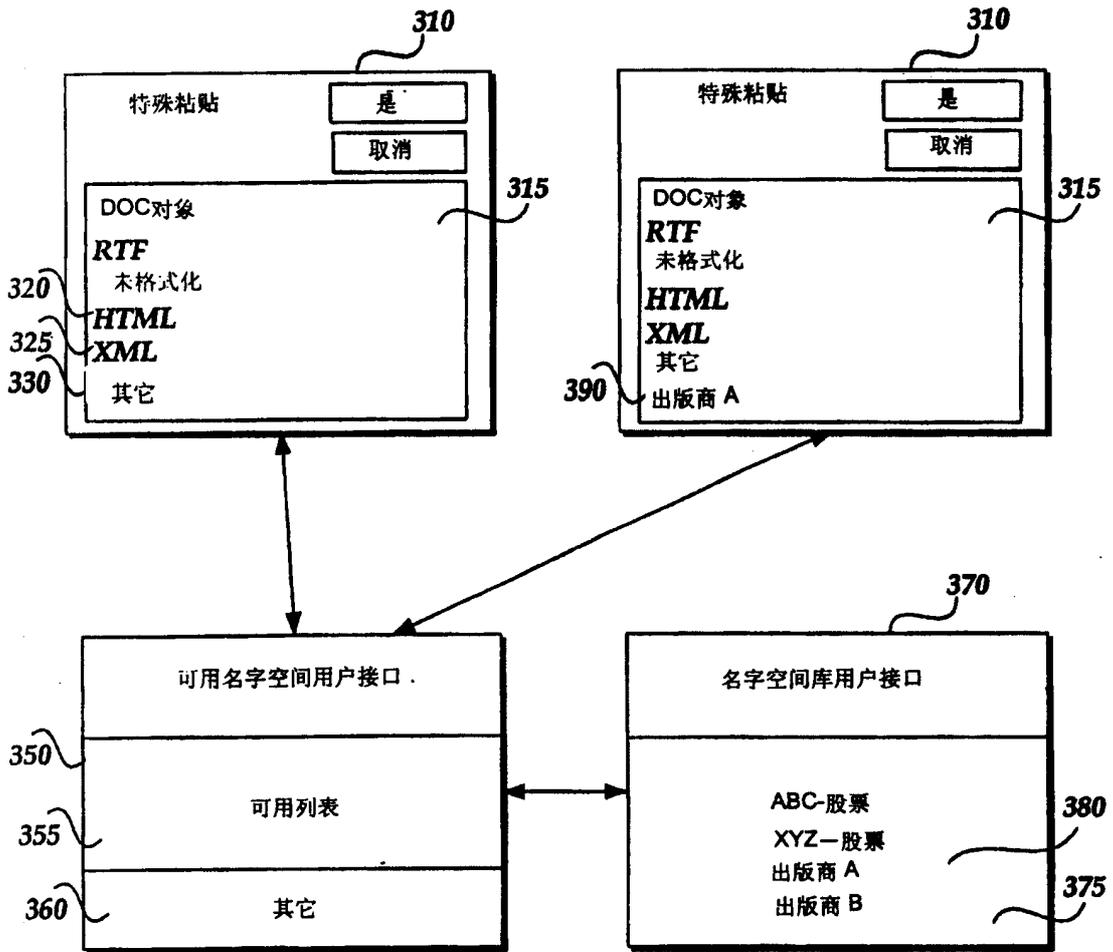


图 3

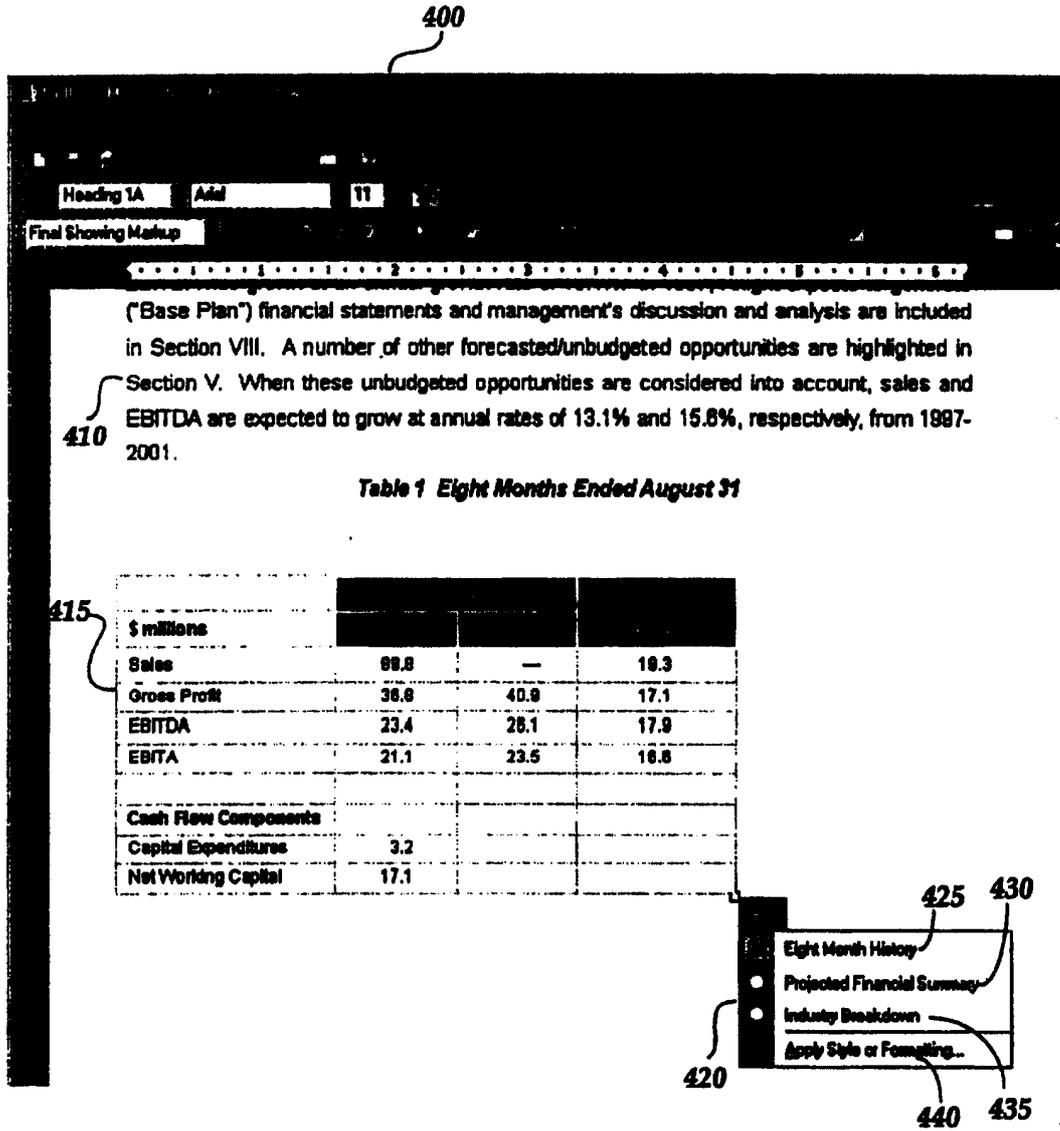


图 4

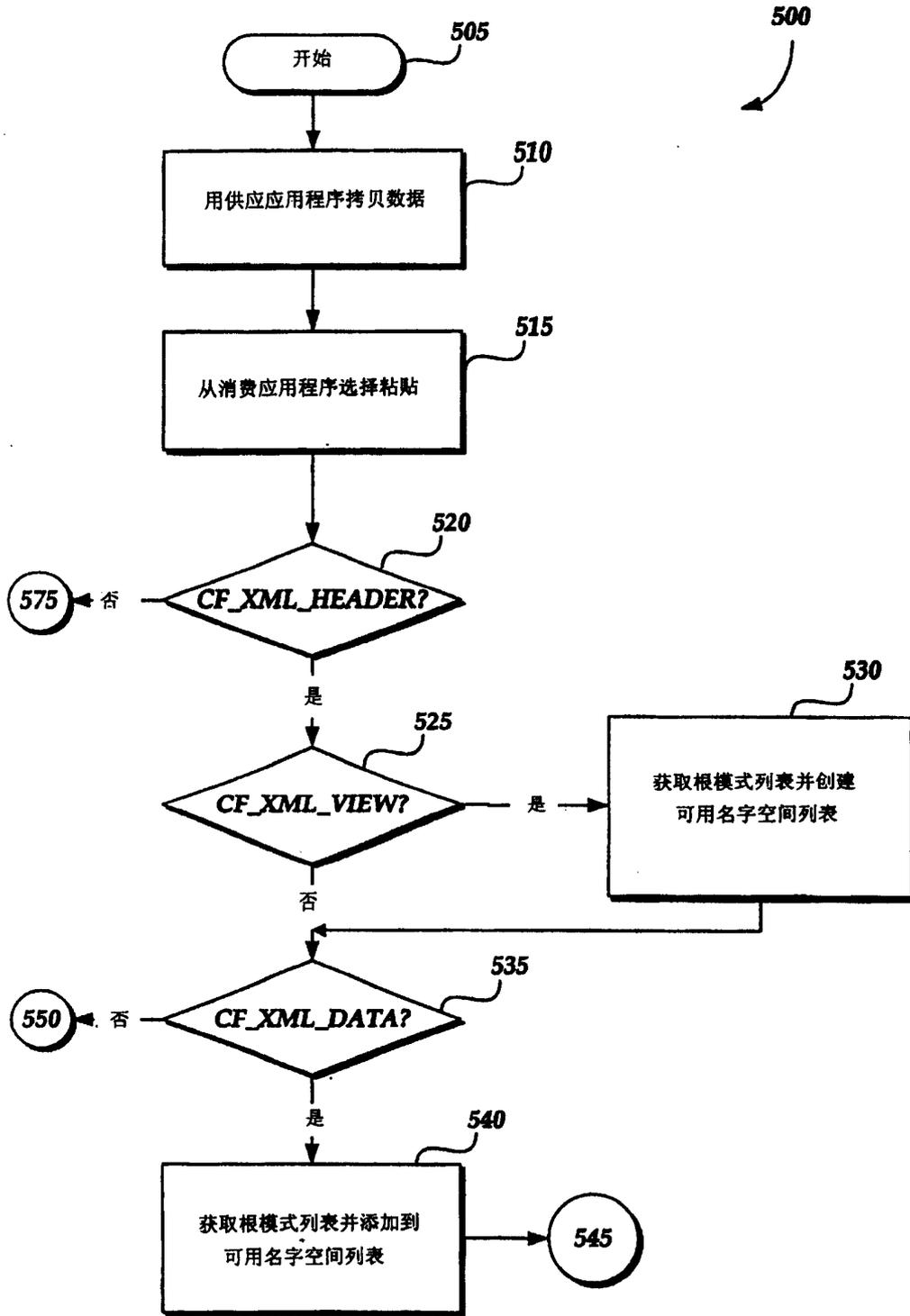


图 5

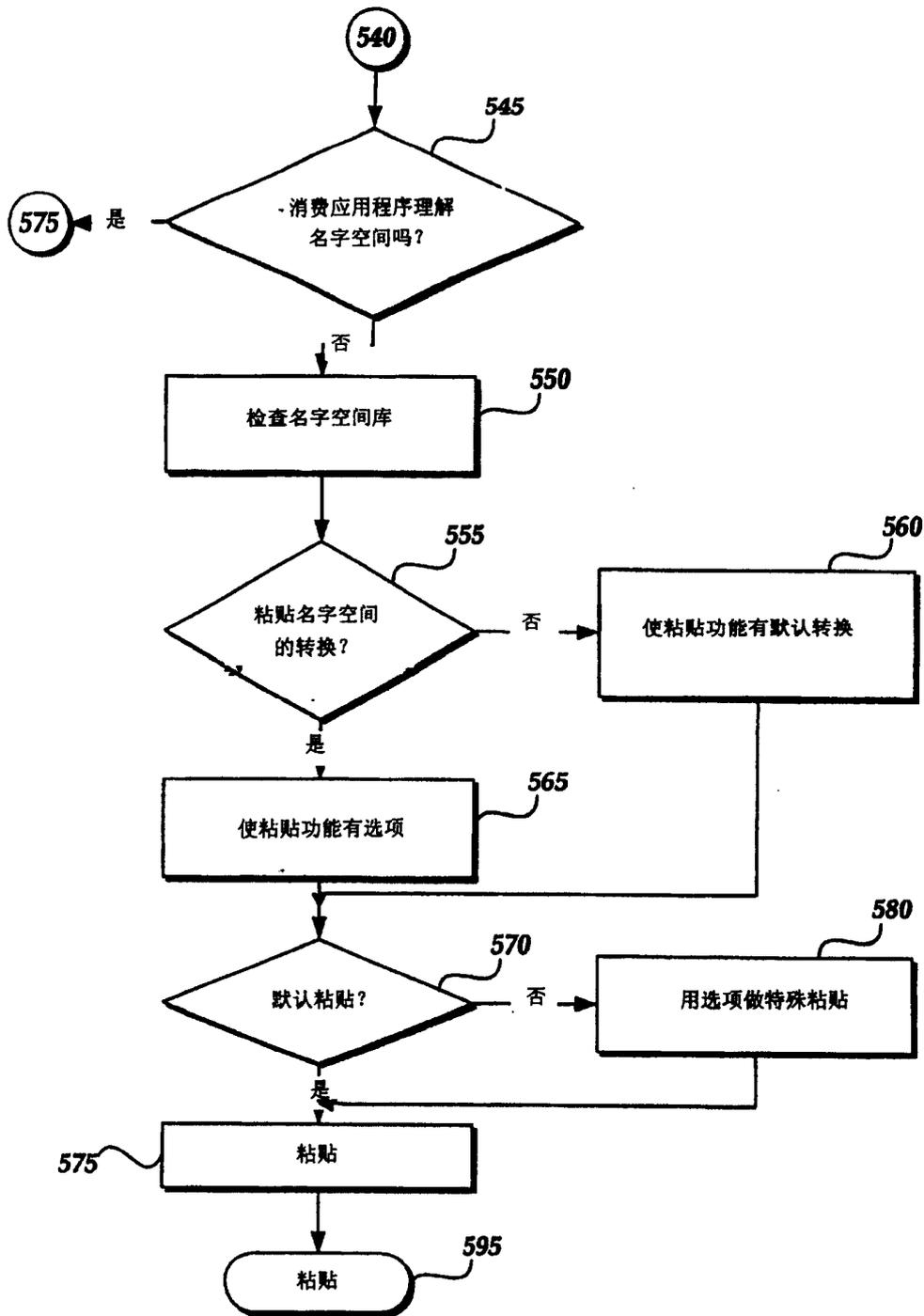


图 6