[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 说 明 书

专利号 ZL 200410007617.2

[45] 授权公告日 2007 年 12 月 26 日

[11] 授权公告号 CN 100357887C

[22] 申请日 2004.2.24
[21] 申请号 200410007617.2
[30] 优先权
[32] 2003. 2.24 [33] US [31] 10/373，362
[73] 专利权人 微软公司
地址 美国华盛顿州
[72] 发明人 M·罗卖林－斯托批格男
M·A·F·卡尔布西
[56] 参考文献
US2002/0156815A1 2002.10.24
EP1220113A2 2002.7.3
CN1371049A 2002.9.25
CN1393795A 2003.1.29
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 张政权

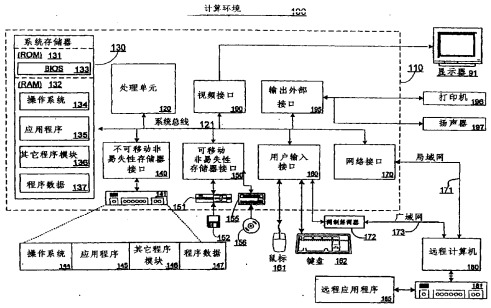
权利要求书 3 页 说明书 17 页 附图 9 页

[54] 发明名称

用于生成网络内容的基础结构

[57] 摘要

用于表示网络内容的结构，将那个内容的各种组成模块化。“线框”定义了一页内容的空间区域。区域被称为“槽”。“视图”定义了在线框中的槽和生成内容对象之间的映像或绑定。根据被绑定到槽的对象，绘制工具用内容填充槽。槽可被绑定到控制、可扩展式样语言(XSL)标记、或其他视图。控制是生成内容以填充槽的可执行或可解释编码。XSL 处理器也可根据 XSL 标记生成这样的内容。此外，视图可以被递归地使用，所以槽的内容可由另一个视图生成。控制可从配置文件中接收参数，所以给定的控制可产生给定原始内容段的不同变体。



1．一个用于生成内容项目的系统，其特征在于，所述系统包括：

用于定义内容项目的多个区域的第一线框装置；

用于提供第一组内容生成对象的装置；

用于将第一组内容生成对象映像到由第一线框定义的区域中的装置；以及

绘制工具装置，对于由第一线框定义的每个区域，所述绘制工具装置用于：

使得内容根据由视图映像到区域的内容生成对象被生成；以及

将生成的内容插入区域；

其中由视图映像到区域的内容生成对象包括可扩展式样语言标记，以及绘制工具装置存在于服务器中。

2．如权利要求 1 所述的系统，其特征在于，内容项目包括超文本标记语言或扩展超文本标记语言网页。

3．如权利要求 1 所述的系统，其特征在于，由视图映像到区域的内容生成对象是包括可执行或可解释编码的控制，生成的内容由所述可执行或可解释编码的执行生成。

4．如权利要求 3 所述的系统，其特征在于，还包括：

用于提供配置文件的装置，

其中由可执行或可解释编码生成的内容至少部分地基于所述配置文件中包含的数据。

5．如权利要求 3 所述的系统，其特征在于，视图规定了模式，在模式中可执行或可解释编码要运行，并且由可执行或可解释编码生成的内容至少部分地基于模式。

6．如权利要求 1 所述的系统，其特征在于， 由视图映像到区域的内容生成对象包括第二视图，第二视图通过将第二线框与第二组内容生成对象联系起来规定内容，生成的内容通过绘制工具装置计算第二视图生成。

7．如权利要求 1 所述的系统，其特征在于，至少一个由第一线框定义的区域不与任何内容生成对象一起被第一视图映像，并且绘制工具装置使得内容项目能够反映在所述由第一线框定义的区域之一中的内容缺失。

8．一个用于生成内容项目的计算机执行方法，其特征在于，包括：

访问规定一组线框的槽与一组内容生成对象之间的联系的视图，线框的槽

定义了内容项目的区域；以及

对于每个槽，根据内容生成对象，将内容插入所述槽之一中，视图将内容生成对象与所述槽之一联系起来，其中内容生成对象包括可扩展式样语言标记，以及其中视图被服务器中的绘制工具使用。

9．如权利要求 8 所述的方法，其特征在于，内容项目包括超文本标记语言或扩展超文本标记语言网页。

10．如权利要求 8 所述的方法，其特征在于，内容生成对象包括可执行或可解释编码，视图将内容生成对象与所述槽之一联系起来，并且其中被插入所述槽之一的内容通过执行所述可执行或可解释编码被生成。

11．如权利要求 10 所述的方法，其特征在于，可执行或可解释编码在确定所生成内容期间考虑包含在配置文件中的数据。

12．如权利要求 10 所述的方法，其特征在于， 可执行或可解释编码在确定所生成内容期间考虑在视图中规定的模式。

13．如权利要求 8 所述的方法，其特征在于，内容生成对象包括子视图，视图将内容生成对象与所述槽之一联系起来，并且方法还包括：

计算子视图以生成被插入到所述槽之一中的内容。

14．一种用于生成网页的方法，其特征在于，方法包括：

接收第一视图，视图规定了第一线框与第一组内容生成对象之间的对应关系，第一线框包括网页的多个区域，每个区域或者是空的或者是对应于第一组内容生成对象之一，第一组内容生成对象包括可扩展式样语言标记，它向可扩展式样语言处理器规定将被生成或检索的内容，以及其中网页是在服务器上被绘制的；

对于第一线框中的每个所述区域：

如果所述区域之一对应于一内容生成对象，那么在所述区域之一中，根据所述内容生成对象，将内容加至网页；

如果所述区域之一不对应于一内容生成对象，那么使得网页被构造以反映所述区域之一中内容的缺失。

15．如权利要求 14 所述的方法，其特征在于，这组内容生成对象包括一个或多个：

可执行或可解释编码，它们操作来生成内容；

第二视图，它规定了第二线框与第二组内容生成对象之间的对应关系。

16. 如权利要求 15 所述的方法，其特征在于，可执行或可解释编码生成的内容是包含在配置文件中的数据的函数。

17. 如权利要求 15 所述的方法，其特征在于，可执行或可解释编码生成的内容是在第一视图中为可执行或可解释编码规定的模式的函数。

18. 一种帮助内容项目的生成的方法，其特征在于，包括：

使用第一线框的第一标识符，第一线框定义内容项目的多个区域；

对于由第一线框定义的至少一个区域，绑定关联于所述区域之一的内容生成对象，所述绑定步骤进一步包括：

通过关联于所述区域之一的第二标识符识别所述区域之一，

通过关联于所述内容生成对象的第三标识符识别内容生成对象，其中内容生成对象包括可扩展式样语言标记，它向存在于服务器的可扩展式样语言处理器规定将被生成或检索的内容。

19. 如权利要求 18 所述的方法，其特征在于，所述第一、第二和第三标识符的每一个包括一文本串。

20. 如权利要求 18 所述的方法，其特征在于，内容生成对象包括生成将被插入进所述区域之一的内容的可执行或可解释编码。

21. 如权利要求 18 所述的方法，其特征在于，内容生成对象包括将第二线框的区域与一组内容生成对象联系起来的第二视图。

用于生成网络内容的基础结构

发明的领域

本发明主要涉及计算领域，尤其是提供可被用来动态地创建网络内容，如由入口站点和搜索引擎创建的网页，的基础结构。

发明的背景

传统网页包括 符合标记语言的内容，标记语言诸如超文本标记语言（HTML）或扩展超文本标记语言（XHTML），它适合于由网络浏览器表现。一些网页是静态的——例如，一页永恒的、不变的 HTML 或 XHTML 内容可被存储在文件中，文件可被下载到用户机器并且显示在浏览器上。然而，大多数商业网络内容，如由搜索网站、入口站点、电子商务网站提供的内容，不是静态的，而是动态生成的，以至于网站对于用户能够个性化，或响应来自于用户的一些输入而生成。

动态生成的内容由专门为此目的设计的程序生成。这样的程序是可执行或可解释编码的模块，根据一套规则或过程模块创建内容。例如，当用户向搜索引擎网站发送一个查询时，搜索引擎查阅各种各样的源（例如，已知网页的数据库）以产生结果，然后搜索引擎网站处的内容生成程序生成包括结果的一页 HTML 内容，并且向用户发送那页。清楚地，这样的结果页面必须被动态地生成，因为结果页面的内容将根据用户提交什么查询而改变。

虽然在动态生成网络内容上现存软件是有效的，但是当存在对程序需要产生的内容的创造性变化时，这样的软件感到灵活性的缺乏。例如，搜索引擎可使用显示搜索结果为搜索引擎定位的网页标题清单的程序。然而，如果搜索引擎的操作者希望引入一个表示这些结果的新方法（例如，用已定位网页的缩略图加强清单），一般，完成这个改变的唯一方法是重写程序（或，至少，增加支持缩略图包含的程序的编码）。

大多数网络内容可视作由较小内容段的模块化“积木”构建。例如，搜索结果页面由各种各样的独立内容段组成（例如，标识、版权启事、最新搜索的结果、用于输入另一个搜索查询的搜索框、广告等等）。理论上，网络内容的

模块化性质建议单个程序可驱动内容生成处理，通过由内容设计者在运行时期提供的模块化积木构建内容。内容的创造性改变不使这样的程序的改变成为必需，因为程序可简单地针对使用不同的积木来产生不同内容。然而，在不改变软件本身可对内容做出什么类型的创造变化的方面，传统内容生成软件是十分有限的。

以上述观点，有必要有个克服先前领域的弊端的系统。

发明概述

本发明提供用于生成内容，如网页的基础结构。基础结构基于一页内容可从更小的部件构建的想法，这些部件可独立地设计、完成和修改。本发明使得这些更小部件，以及它们拟合在一起的方式，能够在运行时期被明确规定给内容生成软件。

依照发明，内容的创建由“线框”、“视图”和内容生成对象明确规定。线框是定义一内容段的空间区域的数据结构。例如，线框可将网页定义为具有四个长方形区域，从页面的顶部至底部垂直地运行。这些区域的每个称为“槽”。“视图”是将线框中每个槽绑定到内容生成对象，或一连串的内容生成对象（或，另一可选择地，指定特定的槽为空）的数据结构。

为了表现视图，绘制工具访问视图中确定的线框，并且对于那个线框中的每个槽，使得内容生成对象被绑定到那个槽中以产生内容。绘制工具然后将由对象产生的内容放入槽中。对于线框中的每个槽这个处理被重复。绘制工具的输出是一内容段（如 HTML 网页），内容包含由在线框的合适区域中的每个内容生成对象产生的内容。

内容生成对象包括“控制”、可扩展式样语言（XSL）标记以及视图。控制是生成将被放入槽中的 HTML 或 XHTML 内容的可执行或可解释编码。XSL 标记是可被 XSL 处理软件用来生成 HTML 或 XHTML 内容的数据。由于要在内容产生中计算视图结果，视图本身就是内容生成对象。因此，内容可从实际上是最高层视图的子视图、子子视图等等的视图构建。当绘制工具遇到被绑定到子视图中的槽时，绘制工具可递归地调用自己以根据那个视图为槽生成内容。

依照本发明的特点，根据配置文件中包含的参数，控制可展示不同的行为。因此，根据在配置文件中什么参数被提供，给定控制可被设计来产生一内容段的不同变体。

本发明的其他特点如下所述。

附图的概述

结合附图阅读，以上概述，以及以下较佳实施例的详细描述将被更好地理解。为了说明本发明，图中示出的是本发明的用作范例结构；但是，本发明不被限制于所揭示的特定的方法和手段。图中：

图 1 是用作范例的计算环境的框图，发明的方面在其中可被实现；

图 2 是用作范例的网络浏览器的用户界面的框图；

图 3 是第一用作范例的线框的框图；

图 4 是第一用作范例的控制的框图；

图 5 是第一用作范例的视图的框图，视图将控制绑定到第一用作范例的线框中；

图 6 是第二用作范例的线框的框图；

图 7-8 是用作范例的可扩展样式语言（XSL）标记的框图；

图 9 是第二用作范例的控制的框图；

图 10 是第二用作范例的视图的框图；

图 11 是第三用作范例的视图的框图；

图 12 是依据本发明的方面被生成的用作范例的内容的框图；

图 13 是依据本发明的方面用于生成内容的用作范例的基础结构的框图；

图 14 是依据本发明的方面用于生成内容的用作范例的处理的框图。

发明的详细说明

综述

许多商业网络内容从更小的内容段被构建，内容以定义的方式图示地被拟合在一起。本发明提供了使得内容提供者根据这些更小段能够明确规定内容如何被构造的基础结构。基础结构包括根据内容提供者的规范动态地生成内容的绘制工具。

用作范例的计算环境

图 1 展示了用作范例的计算环境，本发明的方面在其中可被实现。计算系统环境 100 仅仅是合适的计算环境的一个例子，并且不被试图建议作为发明使用或功能的范围的限制。计算环境 100 应被解释为对于任何部件之一或部件组

合既不具有任何依赖性，也不具有要求，这些部件在用作范例的操作环境 100 中被说明。

本发明可与许多其他通用或专用计算系统环境或配置一起运行。适合和本发明一起使用的众所周知的计算系统、环境、和/或配置的例子包括，而非限制，个人计算机、服务器计算机、手持式或膝上型设备、多处理系统、基于微处理机的系统、机顶盒、可编程用户电子装置、网络 PC、小型计算机、大型计算机、嵌入式系统、包括任何以上系统或设备的分布式计算系统等等。

本发明可用由计算机执行的计算机可执行指令的一般上下文来描述，如程序模块。一般地，程序模块包括完成特定任务或实现特定抽象数据、类型的例程、程序、对象、部件、数据结构等。本发明也可在分布式计算环境中被实践，其中任务由远程处理设备完成，远程处理设备通过通信网络或其他数据传送媒体被链接。在分布式计算环境中，程序模块或其他数据可位于包括存储器存储设备的本地和远程计算存储媒体中。

关于图 1，用于实现本发明的用作范例的系统包括以计算机 110 形式存在的通用计算设备。计算机 110 的部件可包括，但非限制，处理单元 120、系统存储器 130 以及系统总线 121，系统总线 121 将包括系统存储器的各种各样的系统部件耦合到处理单元 120 上。系统总线 121 可以是任何一些类型的总线结构，总线结构包括存储器总线或存储器控制器、外围总线、以及使用任何各种总线结构的本地总线。作为例子，而非限制，这样的结构包括工业标准体系结构（ISA）总线、微通道体系结构（MCA）总线、增强型 ISA（EISA）、视频电子标准协会(VESA)本地总线、以及外围部件互连(PCI)总线(也称为 Mezzanine 总线）。

计算机 110 一般包括各种计算机可读媒体。计算机可读媒体可以是可被计算机 110 访问的任何可用媒体，并且包括易失的和非易失的媒体，可移动的和不可移动的媒体。作为例子，而非限制，计算机可读媒体可包括计算机存储媒体和通信媒体。计算机存储媒体包括易失的和非易失的、可移动的和不可移动的媒体，这些媒体以用于诸如计算机可读指令、数据结构、程序模块或其他数据的信息存储的任何方式或技术被实现。计算机存储媒体包括，而非限制，RAM、ROM、EPROM、闪存或其他存储技术、CDROM、数字多用光盘（DVD）或其他光盘任何存储、磁盒、磁带、磁盘存储或其他磁存储设备，或可用来存储所需信息并且可被计算机 110 访问的任何其他媒介。通信媒体一般包含计算机可读指令、

数据结构、程序模块或在诸如载波或其他传输装置的调制数据信号中的其他数据，并且包括任何信息传递媒体。术语“调制数据信号”意思是，具有一个或多个以编码信号中的信息的信号这样的方式设置或变换的特征的信号。作为例子，而非限制，通信媒体包括诸如有线网络或直接连线连接的有线媒体，以及诸如声的、RF、红外或其他无线媒体的无线媒体。以上任何的组合也应被包括在计算机可读媒体的范围内。

系统存储器 130 包括以易失的和/或非易失的存储器形式存在的计算机存储媒体，如只读存储器（ROM）131 和随机存取存储器（RAM）132。基本输入/输出系统 133（BIOS），包括帮助在计算机 110 中的单元之间传送信息的基本例程，如在启动期间，一般被存储在 ROM131 中。RAM132 一般包括可被处理单元 120 立即访问和/或正被处理单元 120 操作的数据和/或程序模块。作为例子，而非限制，图 1 说明了操作系统 134、应用程序 135、其他程序模块 136 和程序数据 137。

计算机 110 也可包括其他可移动的/不可移动的、易失的/非易失的计算机存储媒体。仅仅作为例子，图 1 说明了从不可移动的、非易失的磁媒体中读取或写入的硬盘驱动 140，从可移动的、非易失的磁盘 152 中读取或写入的磁盘驱动 151，以及从可移动的、非易失的光盘 156 中读取或写入的光盘驱动 155，光盘如 CDROM 或其他光媒体。可被用在用作范例的操作环境中的其他可移动的/不可移动的、易失的/非易失的计算机存储媒体包括，而非限制，磁带盒、闪存卡、数字多用光盘、数字录象带、固态 RAM、固态 ROM 等等。硬盘驱动 141 一般通过诸如接口 140 的不可移动的存储器接口被连接至系统总线 121，磁盘驱动 151 和光盘驱动 155 一般通过可移动存储器接口，如接口 150，被连接至系统总线 121。

以上讨论的并且在图 1 中被说明的驱动和与它们有关的计算机存储媒体向计算机 110 提供了计算机可读指令、数据结构、程序模块以及其他数据的存储。在图 1 中，例如，硬盘驱动 141 被作为存储操作系统 144、应用程序 145、其他程序模块 146 和程序数据 147 来说明。注意，这些部件可以与操作系统 134、应用程序 135、其他程序模块 136 和程序数据 137 相同或不同。操作系统 144、应用程序 145、其他程序模块 146 和程序数据 147 在此被给予不同的数字以说明，至少，它们是不同的复制。用户可通过诸如键盘 162 和指点器 161 的输入设备向计算机 20 输入命令和信息，指点器 161 通常指鼠标、轨迹球或触摸板。

其他输入设备（未展示）可包括话筒、游戏杆、游戏板、卫星抛物面、扫描仪等等。这些或其他输入设备常通过与系统总线耦合的用户输入接口 160 被连接至处理单元 120，但也可通过其他接口和总线结构，如并行口、游戏口或通用串行总线（USB）被连接至处理单元 120。监视器 191 或其他类型的显示设备也经由接口，如视频接口 190，被连接至系统总线 121。除了监视器，计算机还可包括诸如扬声器 197 和打印机 196 的外围输出设备，它们可通过输出外围设备接口 190 被连接。

计算机 110 可运行于使用逻辑连接至一个或多个远程计算机，如远程计算机 180，的联网环境中。远程计算机 180 可以是个人计算机、服务器、路由器、网络 PC、对等设备或其他普通的网络节点，并且一般包括以上描述的与计算机 110 相关的计算机许多或所有单元，尽管只有存储器存储设备 181 在图 1 中被说明。图 1 中所描述的逻辑连接包括局域网（LAN）171 和广域网（WAN）173，但也可包括其他网络。这样的联网环境在办公室、企业级计算机网络、内部网和因特网中是平常的。

当被用在 LAN 联网环境中，计算机 110 通过网络接口或适配器 170 被连接至 LAN171。当被用在 WAN 联网环境中，计算机 110 一般包括调制解调器 172 或用于建立在 WAN173，如因特网，上通信的其他装置。调制解调器 172 可以是内置的或外置的，可经由用户输入接口 160，或其他合适的装置被连接至系统总线 121。在联网环境中，所描述的与计算机 110 有关的程序模块，或其部分，可被存储在远程存储器存储设备中。作为例子，而非限制，图 1 将远程应用程序 185 作为驻存在存储器设备 181 上来说明。要理解的是，所示的网络连接是用作范例的，并且在计算机之间建立通信链接的其他装置也可被使用。

用作范例的网络浏览器

图 2 展示了用作范例的网络浏览器的可视接口。如本领域中所知的，网络浏览器是使得用户能够与某类型内容（如 HTML 内容）交互，并且能够从网络中检索这样的内容的一段软件。浏览器 200 可以，例如，组成存储在计算机 110 上（图 1 中所示的）、并且在处理单元 120 上执行（图 1 中所示的）的软件。浏览器 200 可从计算机 110 所连接到的广域网 173（图 1 中所示的），例如因特网，中访问内容，。一般地，浏览器也可以访问在计算机 110 上本地存储的内容。

图 2 的用作范例浏览器向用户显示各种各样的信息。特别是，浏览器 200

向用户显示站点链接栏 202。站点链接栏 202 包括框 204，用户可输入统一资源定位（URL）到框中，依次点击浏览器到特定内容项目。在图 2 的例子中，用户输入 URL http://search.msn.com 到框 204 中，由此表明用户希望访问由 URL 标识的内容。浏览器 200 从它可能位于的因特网的哪个地方中检索这个内容，并且在视图区域 206 中显示内容。在图 2 的例子中，内容用 HTML 或 XHTML 传送。（在 URL 中的词语 http 表示“超文本传输协议”；在 URL 中它的出现表明基本内容是 HTML 或 XHTML 内容）。浏览器 200 包含，或访问，HTML 和/或 XHTML 解释器，解释器呈现由浏览器 200 接收到的 HTML 或 XHTML 内容。在图 2 的例子中，浏览器 200 在标题栏 208 中显示内容的标题(“MSN Search——“More Useful Everyday”）。

一般的浏览器，诸如浏览器 200，也使得用户能够完成各种各样的功能，如：打印、邮寄、或保存已显示的内容；使用“收藏夹”或“书签”的清单来导航到其他内容；改变文本内容的默认字体等等。这个功能通过菜单栏 210 和/或按钮 212 向用户展示。

如下所述，本发明为生成内容提供了基础结构，如在视图区域 206 中被显示的那些。

用于创建内容呈现的数据结构

本发明提供了归纳和抽象用来创建诸如用于网页的 XHTML 内容的过程的基础结构。（对于这点，浏览器可呈现内容的例子应指的是作为 XHTML 内容的内容。但是，应该理解的是本发明可被用来生成其他类型的内容，如 HTML、无线标记语言（WML）、话音可扩展标记语言（VXML）等等）。

本发明的基础结构利用了四种类型的数据结构：线框、视图、控制和配置文件。

“线框”是定义一页内容的空间区域的数据结构。每个区域被称为“槽”。

“控制”包括动态地生成 XHTML 内容的可执行或可解释编码，被放入由线框定义的特定的槽中。

“视图”是在特定的槽和特定的控制（或特定的一连串控制）之间建立映像的数据结构。视图也可将可扩展样式语言（XSL）标记、或另一个视图映像到槽中（视图也可将一连串 XSL 标记或视图映像到槽中）；到槽中的 XSL 标记和视图的映像在下面更详细讨论。

“配置文件”包括被控制用来影响它们行为的数据。例如，控制可能能

够生成特定内容段的两个或更多的不同变体，并且配置文件可包含表明这些变化中的哪个应被生成的参数。

图 3 展示了用作范例的线框 300。线框 300 将内容的概念页分成区域，区域如上所注，被称为“槽”。在图 3 的例子中，有四个槽，304（1）、304（2）、304（3）和 304（4）。每个槽有个名字；在这个例子中，从 304(1)到 304（4）的名字各自为“slot_header”、“slot_results”、“slot_sidebar”和“slot_footer”。线框 300 本身也有名字（“WireFrame_Site”，在这个例子中）。如下所要讨论的，名字被用在视图数据结构中以识别线框及其每个槽。

一般地，线框被表示为 XHTML 结构。因此，图 3 中所示的用作范例的线框可被定义为如下：

```
<wireframe name=" wireframe_site">
<div><slot name="slot_header"/></div>
<table>
        <tr>
           <td><slot name="slot_results"/></td>
<td><slot name="slot_sidebar"/></td>
</tr>
</table>
<div>< slot name="slot_footer"/></div>
</wireframe>
```

如所见，上述 XHTML 定义设计了四个内容区域：具有槽名“slot_header”的顶端区域；具有槽名“slot_footer”的底端区域；以及在顶端和底端区域之间的两列表格，其中左列被称为“slot_results”，右列被称为“slot_sidebar”。

[0050]图 4 展示了用作范例的控制 400，它生成 XHTML 内容 402。控制 400 有名字（“Control_Footer”,在这个例子中），并且，如下所述，名字被用在视图数据结构以识别控制。控制 400 可被用来产生将被插入到线框 300 中名为“slot_footer”的槽中的内容。

控制 400 包括可执行或可解释编码，当编码执行时被配置成动态地生成 XHTML 内容 402。控制 400 可用各种各样的方式实现。例如，控制 400 可包括机器可执行编码（如.exe 或.dll 文件）、可解释脚本（如在 JAVA 或 C#语言中的脚本），虚拟机器编码（如 Java 字节码）等等。

下面是描述控制 400 功能的伪码的例子。（将易于理解的是下列伪码可在任何上述可执行或可解释部件中被实现。）

```
XTHML Control_Footer()
{
String strFooter;
strFooter=_
     "<TABLE WIDTH=771" HEIGHT="16" CELLPADDING="0"
CELLSPACING="0" BORDER="0">
<TR><TD WIDTH="770" HEIGHT="1" BGCOLOR="#bdbebd">
<SPACER TYPE="block" HEIGHT ="1"/></TD></TR>
<TD><TD HEIGHT="15" class="nsf">&ndsp; 
"
strFooter=strFooter & file_read(copyright_notice.txt);
strFooter=_
"</TD></TR></TABLE>"
reture strFooter;
}
```

上述伪码将控制 400 定义为生成和返回 XHTML 内容的名为“Control_Footer“的函数。在这个例子中，Control_Footer 将 XTHML 编码返回到表格中，表格具有特定尺寸和背景颜色，并且包含诸如” Copyright 2003 Microsoft Corporation- All Rights Reserved” 的文本串(或不管什么文本串被包含在文件“copyright—notice.txt”中。）

如上所注明的，控制可借助于被称为“视图“的结构被映像（或“绑定”）到线框的槽中。图 5 展示了用作范例的控制到线框 200 的槽中的映像。在这个例子中，控制 420（被称为“Control_Header”）映像到槽 304（1），控制 410（”Control_Results”）映像到槽 304（2），控制 430（”Control_Sidebar”）映像到槽 304（3），以及控制 440（”Control_Footer”,如上所述与图 4 有关的）映像到槽 304（4）。这种映像根据线框 200 定义了一页 XTHML 内容，线框中，由槽 304（1）、304（2）、304（3）、和 304（4）定义的每个区域将各自地被由控制 420、410、430、440 生成的 XHTML 内容填充。这个映像表示视图 500。

视图 500，如图 5 所描述的，可由下列数据结构来表示：

```
<view name="view_Classic" wireframe="wireframe_site">
<binding slot="slot_header">
<ctl:Control_Header/>
</binding>
<binding slot="slot_results">
<ctl:Control_Results/>
</binding>
<binding slot="slot_sidebar">
<ctl:Control_Sidebar/>
</binding>
<binding slot="slot_footer">
<ctl:Control_Footer/>
</binding>
</view>
```

以上所示的数据结构将名字“view_Classic”分配给视图。因为一般系统有几个视图，名字允许视图从几个视图中被识别。正如所见，数据结构包含每个槽的“绑定”，并且表明将被“绑定”到槽中的特定控制。

虽然上述例子展示被绑定到每个槽中的简单控制，但是应该注意到，多个控制可被绑定到单个槽中以被按次序地执行。因此，“slot_header”的绑定可如下被读取：

```
<binding slot="slot_header">
<ctl: Control_A/>
<ctl: Control_B/>
<ctl: Control_C/>
</binding>
```

这种绑定将表明名为“Control_A”、“ Control_B ”和“Control_C”的控制按表明的次序执行以便将内容填充 slot_header 中。

[0056]此外，当上述例子仅仅展示被绑定到槽中的控制，如前所注明的，本发明也允许视图或 XSL 标记被绑定到槽中。控制到槽的绑定是绑定的最简单例子。但是，槽绑定的概念可以被通用化，图 6-12 展示了这样通用化的例子。

用视图来明确规定内容的例子

如上所述，控制是生成 XHTML 内容的软件对象。将控制“映像”或“绑定”到槽意味着槽将被由控制生成的 XHTML 内容填充。然而，控制不是生成 XHTML 内容的唯一类型的对象。特别是，本领域所熟知的是 XSL 标记可被计算来产生 XHTML 内容。此外，当视图被计算来将合适的内容放入线框的槽中时，这个计算的结果是 XHTML 内容，所以视图也是产生 XTHML 内容的另一个对象。考虑这些事实，可看到 XSL 标记和视图，以及可被计算来产生 XTHML 内容的任何其他类型的对象，可被映像到槽。

图 6-11 展示如何构建视图，视图可被绑定到线框 300（图 3 中所示的）中名为“slot_header”的槽。结合以前例子，视图包含用于特定线框的槽绑定，所以有必要定义将被用在视图中的线框。图 6 展示了线框 600，被称为”WireFrame_Header”。线框 600 有四个槽，602（1）、602（2）、602（3）和 602（4），各自被称为“slot_big7”、“slot_banner”、“slot_sharkfin”和“slot_querybox”。这个线框可由下列数据结构定义：

```
<wireframe name="WireFrame_Header">
<div><slot name="slot_big7"/></div>
<div><slot name="slot_banner"/></div>
<div><slot name="slot_sharkfin"/></div>
<div><slot name="slot_querybox"/></div>
</wiregrame>
```

图 7-10 展示将被绑定到线框 600 的槽 602（1）至 602（4）的各种各样的对象。

[0060]图 7 展示了名为“Big7”的 XSL 标记 700。XSL 标记 700 关联于 XSL 数据，它在合适的 XSL 处理软件的帮助下，可被用来生成 XHTML 内容，内容包含出现 search.msn.com 美国版网站的顶端的“big7”链接 702。这些“big7”链接是”MSN Home”、”My MSN”、“Hotmail”、”Search”、“Shopping”、“Money”以及“People & Chat”。

图 8 展示了另一个 XSL 标记 800，被称为“Banner”。 XSL 标记 800 可被 XSL 处理软件以与 XSL 标记 700 相同的方式计算。但是，XSL 标记 800 产生条幅广告 802。在图 8 的例子中，这个条幅广告是用于 Redmond 大学的。

图 9 展示了被称为“Control_Sharkfin”的控制 900。控制 900 生成在曲线

图形（参照数字 902）旁边展示字母 MSN 的 XHTML 内容，内容出现在 search.msn.com 网站上。（图形从鲨鱼的背鳍导出其名字，曲线是它的联想）

图 10 展示了被称为“View_QueryBox”的视图 1000。与视图 500 一起（上述连同图 5 讨论的），视图 1000 表示在线框的槽和将为那些槽生成内容的对象（如控制、XSL 标记、或视图）之间的映像或绑定。在图 10 的例子中，线框 1002 有两个槽，1004（1）和 1004（2），各自被称为“slot_searchbox”和“slot_polead”。视图 1000 将槽 1004（1）（“slot_searchbox”）绑定到为框 1006 和“go”按钮 1008 产生 XHTML 内容的对象中，用户可将查询打入框 1006，用户可点击“go”按钮 1008 向搜索引擎发送查询。槽 1004(2)(“slot_polead”)被绑定到为广告 1010 产生 XHTML 内容的对象。

[0064]将 XSL 标记 700 和 800、控制 900、以及视图 1000 绑定到线框 600 的槽的视图可被创建（图 6 中所示的）。图 11 展示了这样的绑定。XSL 标记 700（”Big7”）被绑定到槽 602(1)（“slot_big7”）。XSL 标记 800 被绑定到槽 602(2)(“slot_banner”)。控制 900(”Control_Sharkfin”)被绑定到槽 602(3)（“slot_sharkfin”）。视图 1000（”View_queryBox”）被绑定到槽 602(4)（“slot_querybox”）。由这种绑定定义的视图 1100 可如下列数据结构表示：

```
<view name-=”view_header” wireframe=”WireFrame header”>
<binding slot=”slot-big7”>
<xsl:Big7/> ‘XSL yields Big7 HTML
</binding>
<binding slot=”slot_banner”>
<xsl:Banner/> “XSL yields Banner Ad creative
</binding>
<binding slot=”slot_sharkfin”>
<ctl:Control_Skarkfin/>
</binding>
<binding slot=”slot_querybox”>
<view:View_QueryBox/>
</binding>
</view>
```

上述数据结构将 “view_header”的名字给于视图 1000，并且创建了图 11

中所描述的绑定。在上面例子中，要注意的是被绑定到槽的每个类型的对象被标志预定为它的类型——例如，“xsl:Banner”，以表明“Banner”是 XSL 标记，并且相似的词语“ctl:Control_Skarkfin”和“view:View_QueryBox”来标志控制和视图。使用这种类型的标志是较佳的，因为控制、XSL 标记、以及视图要求不同类型的处理，并且标志使得视图绘制工具（下面连同图 13-14 所讨论的）能够容易地识别什么类型的对象需要处理。

如上所注明的，图 6-11 展示了例子，其中视图（视图 1100）被绑定到线框 300 中的槽 304（1）（”slot_header”）。（另一个视图不得不被创建以将线框 300 的槽绑定到将填充这些槽的内容生成对象。这个视图将包括在槽 304（1）和视图 1000 之间的绑定。这样的视图的细节能够容易地从后续讨论中被理解。)图 12 展示了根据这个绑定可被创建的结果内容。图 12 展示了线框 300,及其四个槽 304(1)、304(2)、304(3)和 304(4)。因为槽 304(1)(“slot_header”)被绑定至视图 1100 中，线框 600（视图 1100 是以此为基础的）作为覆盖槽 304（1）被展示。线框 600，依次，有四个槽 602(1)、602(2)、602(3)和 602(4)。在这四个槽中，槽 602(4)被绑定至视图 1000（图 10 中所示的），视图 1000，依次，是以线框 1002 为基础的。因此，在图 12 中线框 1002 作为覆盖槽 602(4)被展示。

从图 12 中可理解，其中所示的内容形成了层次，其中线框 1002 是在线框 600 的一个槽中，线框 600，依次，是在线框 300 的一个槽中。从层次底部到向上分析图 12 中所示的内容，可以看到视图 1000 生成下列内容：框 1006 和“go”按钮 1008（在线框 1002 中的槽 1004（1）内），以及广告 1010（在槽 1004（2）内）。这个内容位于线框 600 中的槽 602（4）内。线框 600 也包含内容项目 702、802 和 902，这些可从 XSL 标记和控制中被生成，如以上图 7-9 中所描述的。视图 1100 因此生成被包含在线框 600 的槽中的内容。因为视图 1100 已被绑定到线框 300 的槽 304（1），由视图 1100 生成的内容被放入槽 304（1）中。图 12 也展示了槽 304（2）-304（4）；视图可以被明确规定为以上述方式将这些槽绑定到控制、XSL 标记、以及视图。虽然图 6-12 的例子展示被绑定到单个控制、XSL 标记、或视图的每个槽，如上所注明，槽可以被绑定到多个控制、XSL 标记、或视图，它们都能被计算来填充给定的槽。

应该注明的，虽然图 12 使用虚线来展示线框及其槽的外形，但是这些线实际上不被包括在从视图构建的内容中。这些线仅仅在图 12 中被展示以说明

层次结构，在结构中内容可根据本发明被构建。

从视图生成内容的结构

图 13 展示了依据本发明被用来生成内容的各种各样部件的概述。

绘制工具 1302 是使用视图 1304 来生成 XHTML 内容的软件部件，内容适合于在浏览器（如浏览器 200，图 2 中所示的）上表现。绘制工具可以，作为例子，在搜索引擎的服务器上存在，并且可以被用来动态地生成结果页面以响应用户查询。

如上所讨论的，视图，如视图 1304，包括在线框 1308 的槽的一方和内容生成对象 1308 的另一方之间的映像或绑定。也如上所讨论的，内容生成对象 1316 可采取各种各样的形式；控制 1310、XSL 标记 1312、以及视图 1314 是对象的例子，内容生成以对象为基础。

控制 1310，如上所述，包括动态地生成内容的可执行或可解释编码。因为控制 1310 可以作为程序被实现，根据它们的输入它们能够展现不同行为。配置文件 1316 向控制提供这个输入。例如，根据各种各样情况，“shark fin”控制（参照数字 900，图 9 中所示的）可被用于生成鲨鱼鳍图形的不同版本（如，更宽或更窄的版本）。例如，根据网站是正通过导航栏 202（图 2 中所示的）被直接访问，还是通过将视图区域 206 分割成两个窄的子区域的搜索协助特性被访问，search.msn.com 网站可生成鲨鱼鳍的或宽或窄的版本。配置文件 1316 可包含被鲨鱼鳍控制用来明确规定哪个鲨鱼鳍的版本将被生成的参数。如果鲨鱼鳍控制被设计来响应配置文件中的参数，那么鲨鱼鳍的不同版本可用单个控制和多个配置文件被创建；不同的配置文件可根据哪个鲨鱼鳍的版本将被产生提供给控制。此外，配置信息可作为槽绑定的部分在视图中被明确规定。例如，绑定：

<binding slot="slot_a">

<control_name="header" mode="default">

</binding>

可被用来明确规定“header”控制将被用于默认模式，并且相似的绑定：

<binding slot="slot_a">

<control_name="header" mode="searchpane">

</binding>

也可被用来明确规定“header”控制将被用于“searchpane”模式。这种明

确规定不同模式的方式与使用不同配置文件来明确规定用于控制的不同参数的方式同样直作用。

如配置文件使用的另一个例子，在这里描述的基础结构可被搜索引擎用来生成查询和结果页面。不同用户可能希望以不同格式浏览搜索结果，——例如一个用户可能希望仅仅接收搜索结果的文本描述，而另一个用户可能希望浏览伴随文本描述的缩略图。单个“结果”控制可以被写（如，用于绑定到“slot_results”槽 304（2）中，图 3 中所示），这能够生成各种格式的结果。由“results”控制生成的特定格式依赖于包含在配置文件 1316 中的参数。结果被表示的实际格式可通过改变配置文件被简单地改变。在一个例子中，一些配置文件（每个具有不同参数）可被提供，并且用户可以被给以改变结果呈现的机会。例如，当用户接收包含搜索结果的页面时，用户可使用页面上的菜单来选择用户想要接收结果的不同形式。用户的选择然后被发送至搜索引擎服务器，带有服务器用不同配置文件重新生成页面的请求。

应该注意的是配置文件 1316 可被作为传统的“文件”被实现（例如，被存储在文件系统中的指定对象），但不限制于这个实施例。一般地，“配置文件”表示存储在某个地方并且可被控制访问的数据，而不管这个数据是如何被存储的。

根据视图生成内容的过程

根据视图生成内容的过程可由下列伪码描述：

```
RendererEval()
{
For each Slot in the View's WireFrame
{
Find Slot binding in View Bindings
If binding exists
If binding is a Control
Slot=Evaluate the control
If binding is a XSL Tag
Slot=Process XSL
If binding is a View
Slot=RendererEval (sub view)
```

```
If binding does not exist
Slot=empty
}
}
```

上述 RendererEval()方法被实现，例如，通过绘制工具 1302。方法采用视图作为参数，步进通过视图的线框中的槽，并且根据什么对象被绑定到槽为每个槽生成合适的 XHTML 内容。如果槽被绑定到控制，控制被执行并且槽被由控制生成的 XHTML 内容填充。如果槽被绑定到 XSL 标记，那么 XSL 处理软件被应用到标记上以生成合适的 XHTML 内容。如果槽被绑定到视图，那么视图被计算来生成 XHTML 内容。在上面的伪码中，“子视图”是被绑定到“RendererEval()”函数正在计算的槽的视图。正如以上伪码中所见的，RendererEval()可以递归地调用自身以计算这个子视图。

图 14 展示了由绘制工具 1302 实现的过程的流程图。绘制工具得到视图(方块 1402）。在以上伪码例子中，当视图被作为参数被传递到绘制工具的最高层方法时，这个步骤被完成。绘制工具然后检查第一个槽绑定（方块 1404）。如果绑定是空的（方块 1405），那么没有内容放入这个槽中，所以绘制工具继续到方块 1414 以决定是否有任何更多的槽。

然而，如果槽没有绑定，那么绘制工具进行计算绑定到槽的对象，并且根据那个对象用合适的内容填充槽。如上所注明的，绑定较佳地表明槽是否被绑定到控制、XSL 标记或视图。根据什么类型的对象被绑定到槽，绘制工具出现分支（方块 1408）。如果槽被绑定到控制，那么控制被执行以产生 XHTML 内容，并且 XHTML 内容被放入槽中（方块 1408）。如果槽被绑定到 XSL 标记，那么 XSL 标记被处理以生成 XHTML 内容，并且内容被放入槽中（方块 1410）。如果槽被绑定到另一个视图，那么这个其他的视图被计算来生成 XHTML 内容，并且那个内容被放入槽中（方块 1412）。如上所注明的，当视图在槽绑定中被遇到时，计算那个视图的过程是和计算最高层视图的过程本质相同的；方块 1412，事实上，可作为对完成图 14 过程的方法的递归调用被实现。

当前的槽被处理后，绘制工具决定在当前视图中是否有另外的槽（方块 1414）。如果没有这样的槽，那么过程终止。如果有另外的槽，那么绘制工具继续到下一个槽（方块 1416），并且返回到方块 1405 来处理下一个槽。

[0078]要注意的是，上述例子仅仅为了解释被提供，并且决不被理解为本

发明的限制。当本发明参考各种实施例描述时，要理解这里所使用的词是描述和说明性的词，而不是限制的词。此外，尽管本发明参考特定装置、材料和实施例在此描述，但是本发明不试图被限制于这里揭示的特例；而是，本发明扩展至所有功能上等效的结构、方法和使用，如在后附权利要求书的范围中。那些本领域熟练的技术人员，得益于这个规范的指导，在不脱离本发明在其方面的范围和精神下，使得许多本发明的修改和变化可以被做出。

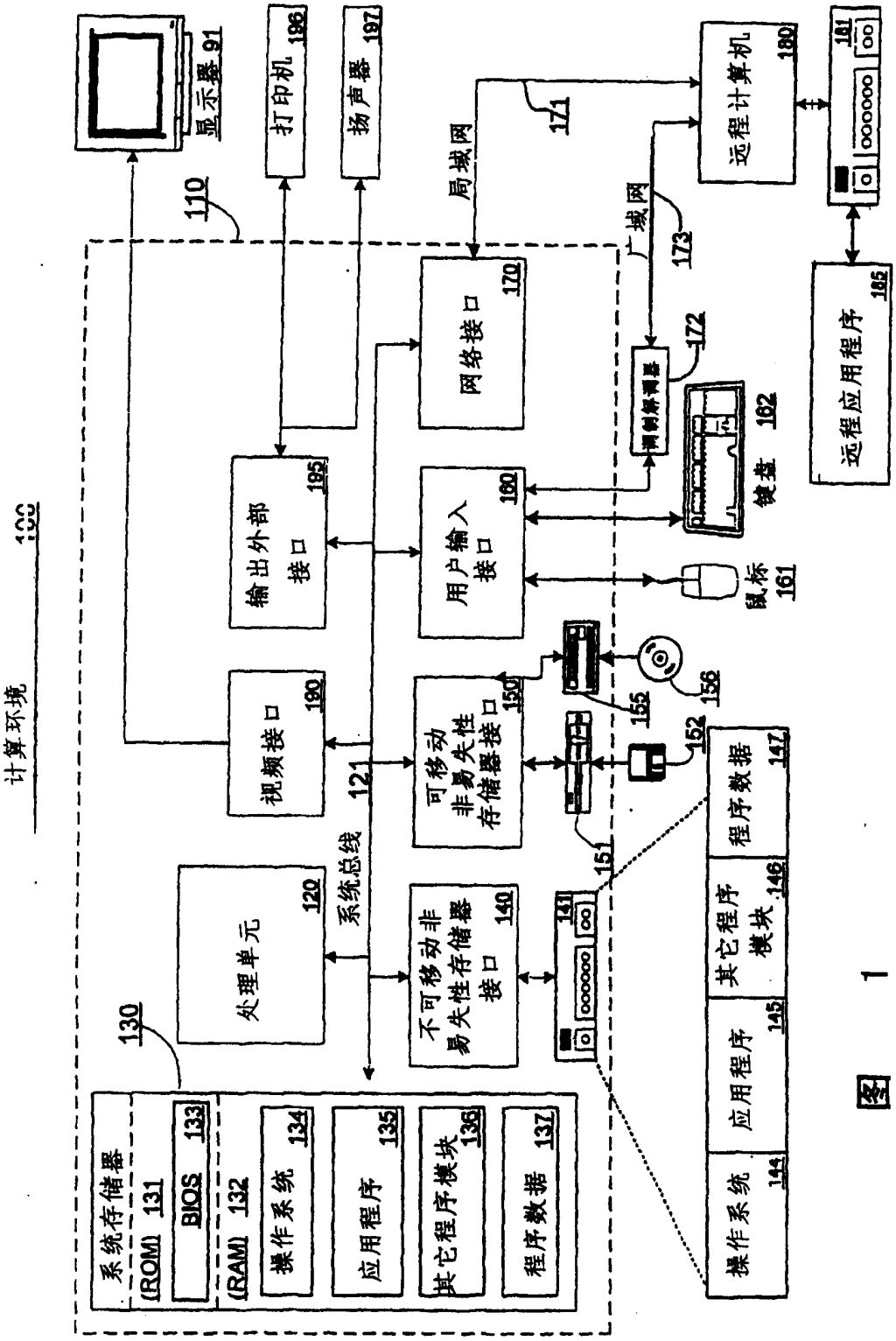

计算环境
100
110
显示器 91
打印机 196
扬声器 197
局域网
171
远程计算机 180
181
广域网
173
远程应用程序 185
网络接口 170
172
调制解调器
键盘 162
鼠标 161
输出外部接口 195
用户输入接口 160
视频接口 190
可移动非易失性存储器接口 150
155
156
152
151
121
系统总线
处理单元 120
不可移动非易失性存储器接口 140
141
130
系统存储器
(ROM) 131
BIOS 133
(RAM) 132
操作系统 134
应用程序 135
其它程序模块 136
程序数据 137
操作系统 144
应用程序 145
其它程序模块 146
程序数据 147

图 1

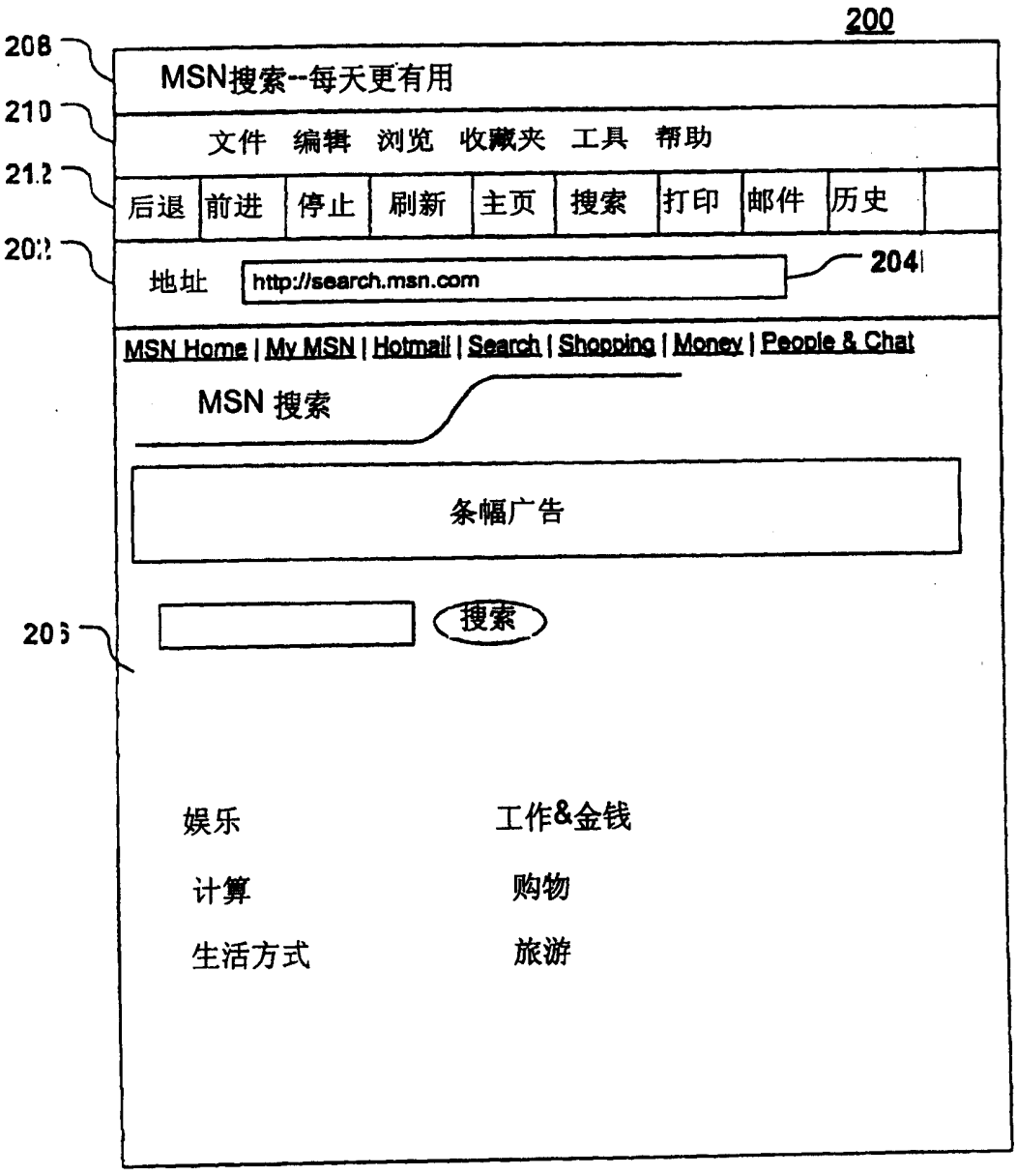

200
208
MSN搜索--每天更有用
210
文件 编辑 浏览 收藏夹 工具 帮助
212
后退 前进 停止 刷新 主页 搜索 打印 邮件 历史
202
地址 http://search.msn.com
204
MSN Home | My MSN | Hotmail | Search | Shopping | Money | People & Chat
MSN 搜索
条幅广告
206
搜索
娱乐
工作&金钱
计算
购物
生活方式
旅游

图 2

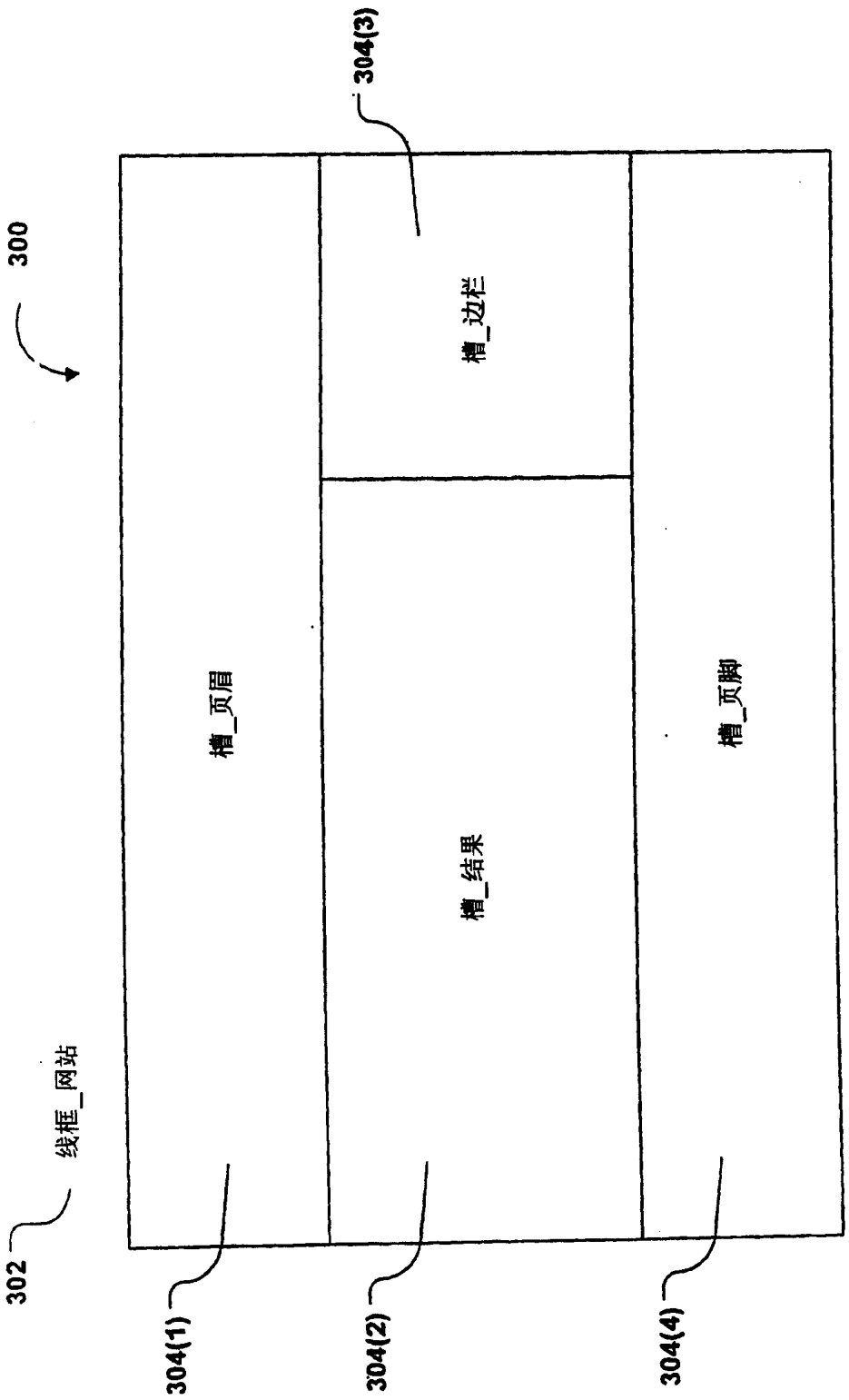


图 3

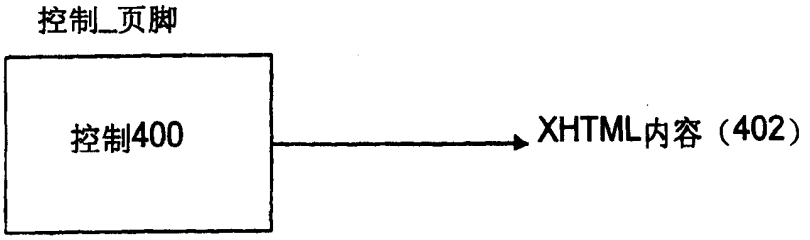

控制_页脚
控制400
XHTML内容（402）

图 4

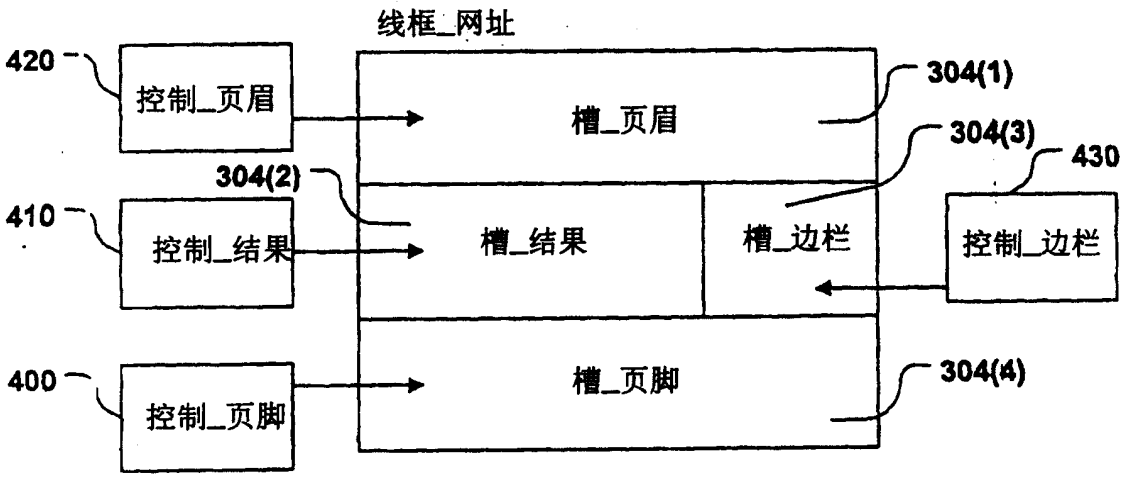

线框_网址
420
控制_页眉
槽_页眉
304(1)
304(3)
430
304(2)
410
控制_结果
槽_结果
槽_边栏
控制_边栏
400
控制_页脚
槽_页脚
304(4)

图 5

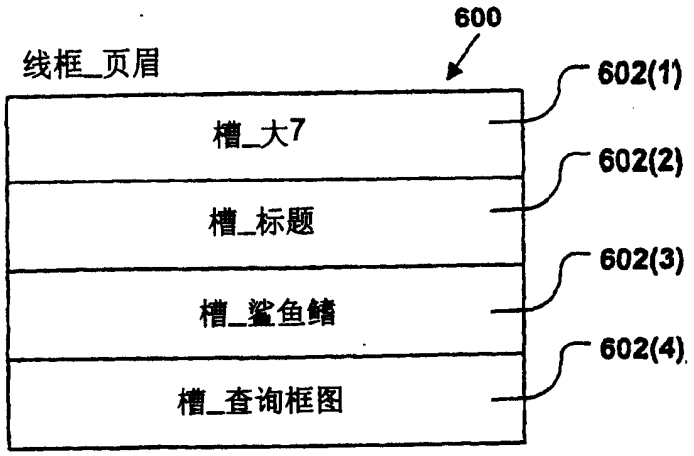

600
线框_页眉
602(1)
槽_大7
602(2)
槽_标题
602(3)
槽_鲨鱼鳍
602(4)
槽_查询框图

图 6


700
XSL标记
("Big7")
用于
"MSN Home | My MSN | Hotmail | Search |
Shopping | Money | People & Chat"
"的XHTML内容
702

图 7

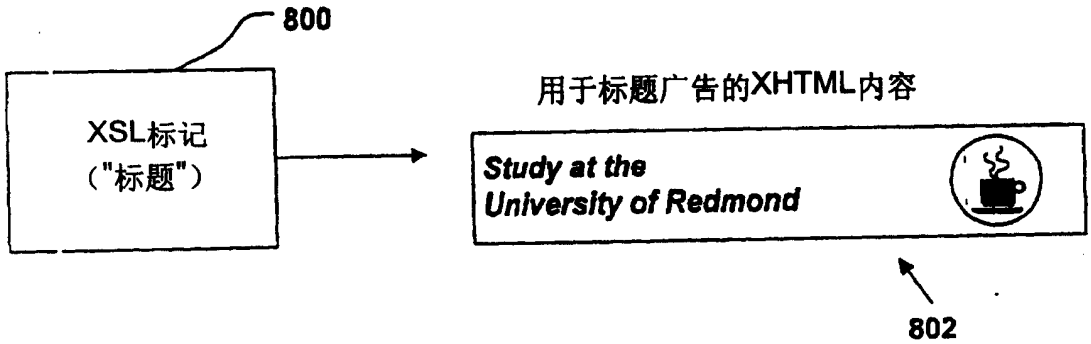

800
XSL标记
("标题")
用于标题广告的XHTML内容
Study at the
University of Redmond
802

图 8


900
控制
("控制_鲨鱼鳍")
用于称作"鲨鱼鳍"的图形符号的XHTML内容
MSN搜索
902

图 9

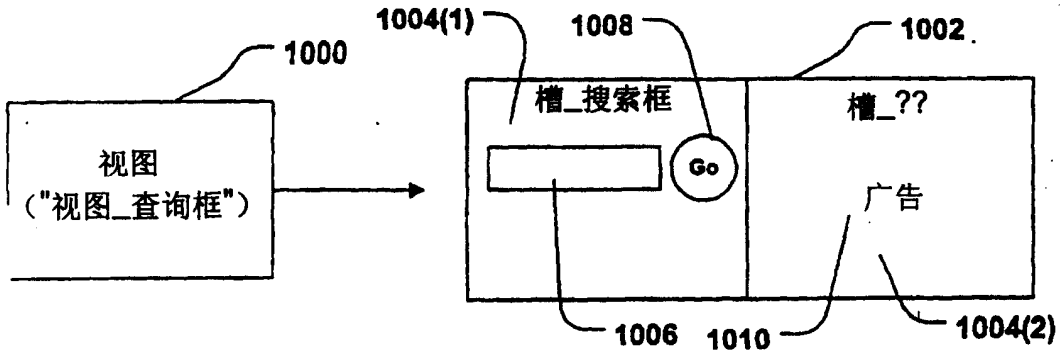

1000
视图
("视图_查询框")
1004(1)
1008
1002
槽_搜索框
槽_??
Go
广告
1006
1010
1004(2)

图 10

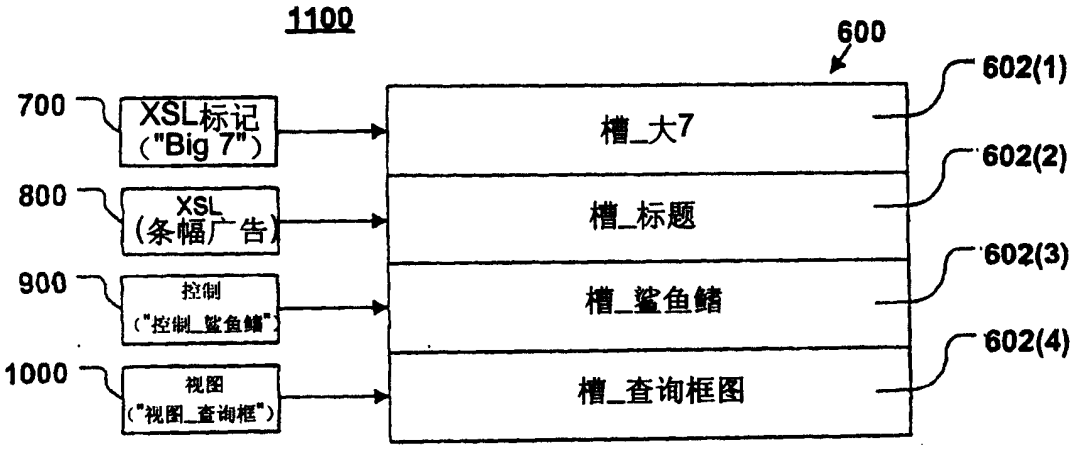

1100
600
700
XSL标记
("Big 7")
800
XSL
(条幅广告)
900
控制
("控制_鲨鱼鳍")
1000
视图
("视图_查询框")
槽_大7
槽_标题
槽_鲨鱼鳍
槽_查询框图
602(1)
602(2)
602(3)
602(4)

图 11

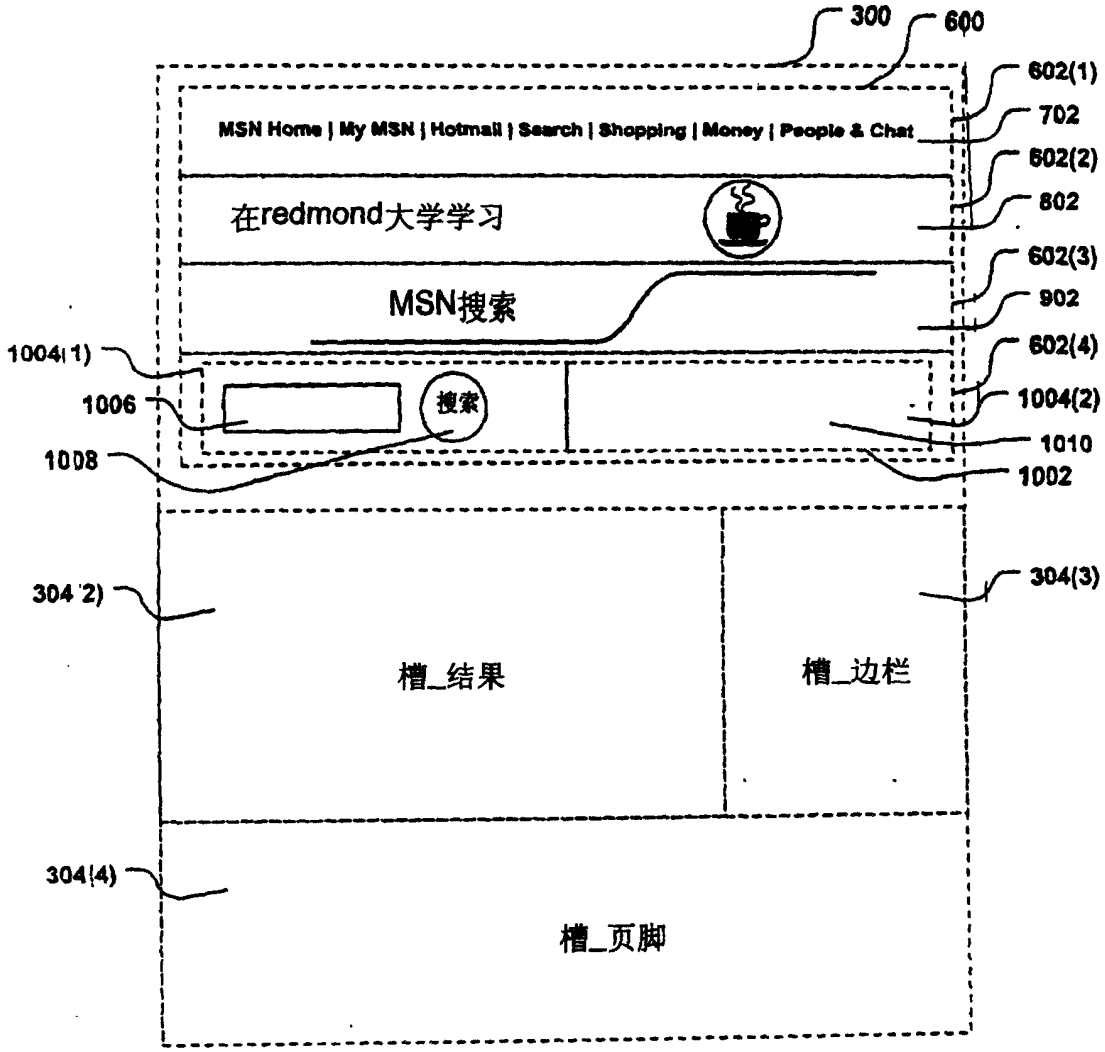

300
600
602(1)
MSN Home | My MSN | Hotmail | Search | Shopping | Money | People & Chat
702
602(2)
在redmond大学学习
802
602(3)
MSN搜索
902
602(4)
1004(1)
1006
搜索
1004(2)
1010
1008
1002
304(2)
304(3)
槽_结果
槽_边栏
304(4)
槽_页脚

图 12

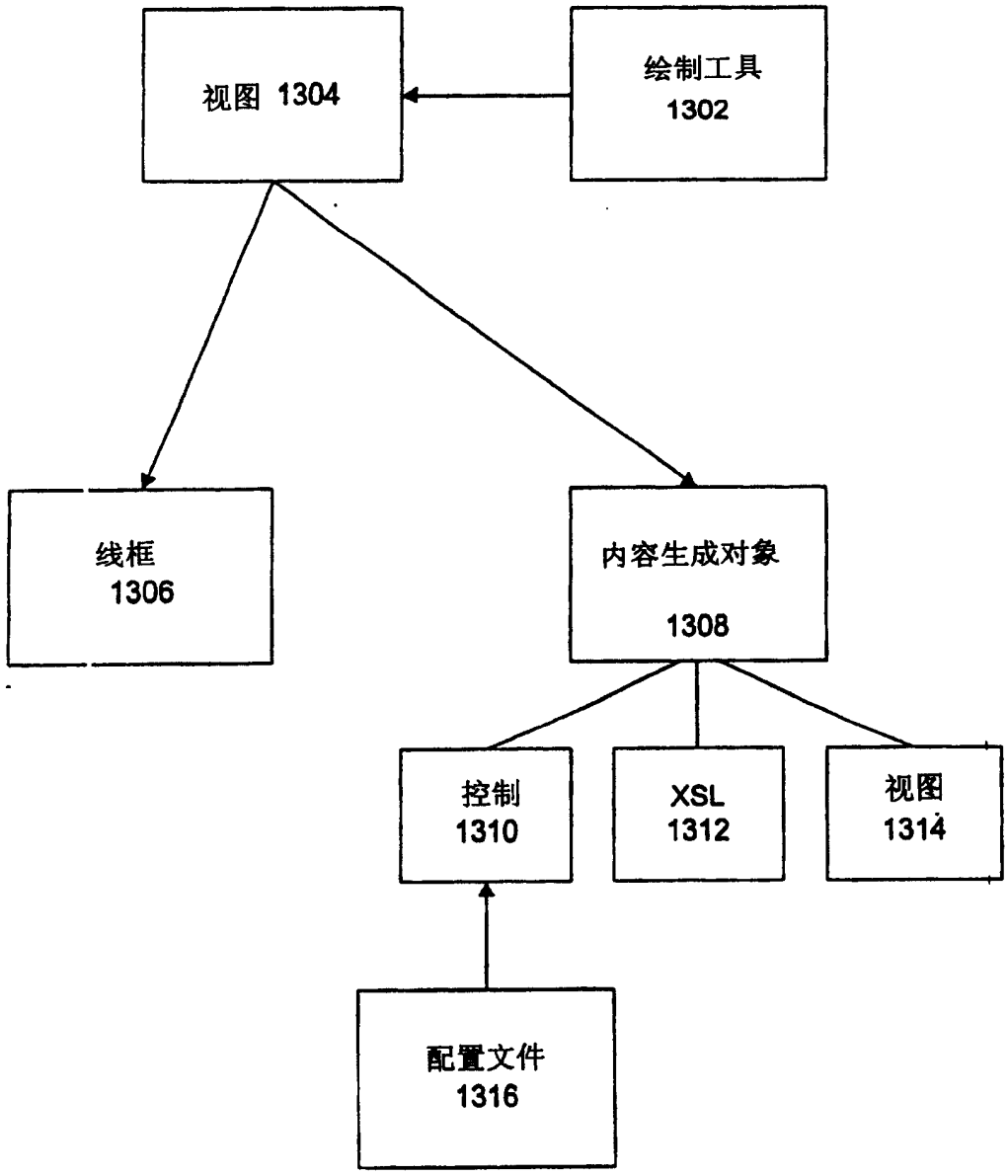

视图 1304
绘制工具
1302
线框
1306
内容生成对象
1308
控制
1310
XSL
1312
视图
1314
配置文件
1316

图 13

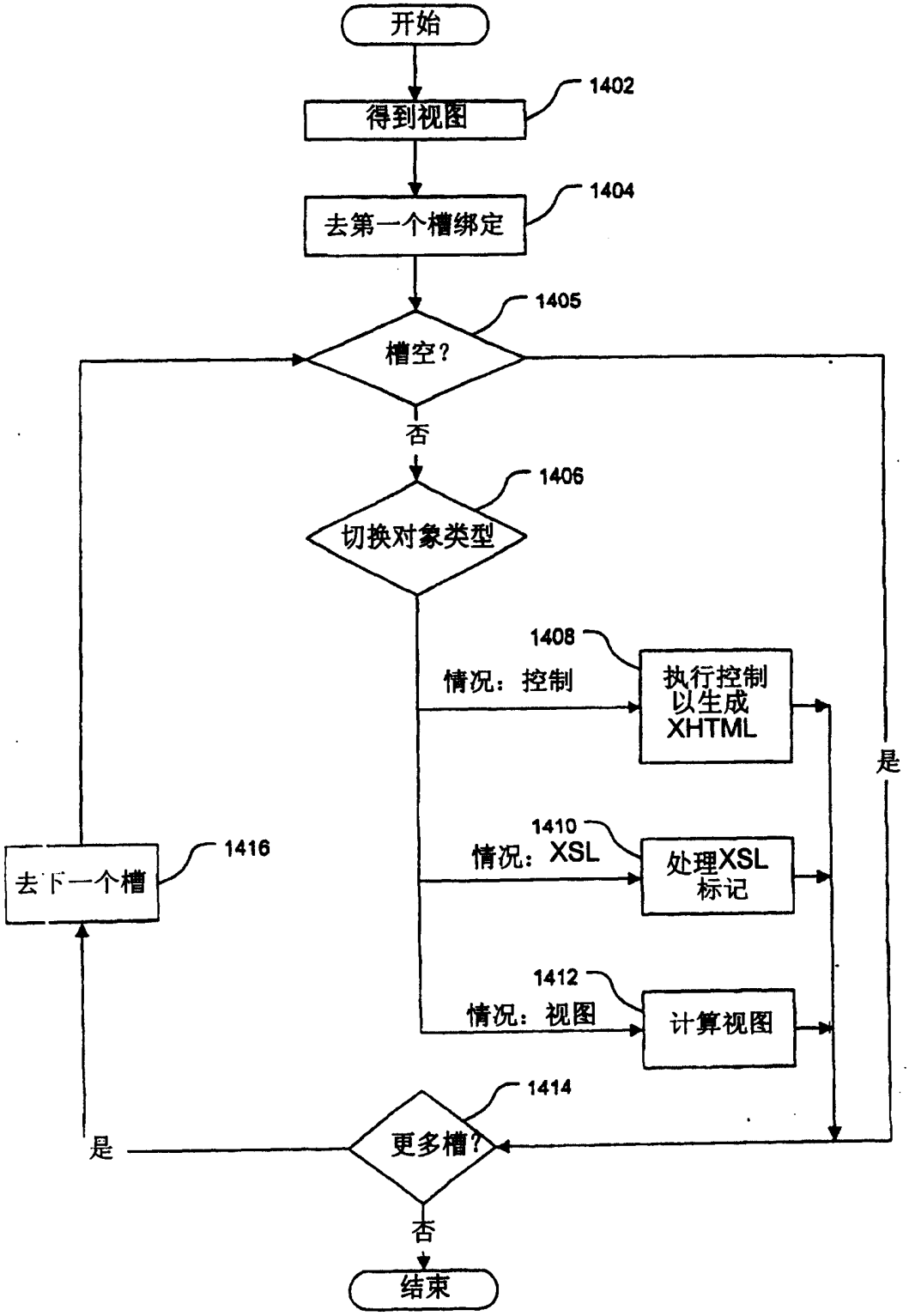

开始
1402
得到视图
1404
去第一个槽绑定
1405
槽空?
否
1406
切换对象类型
1408
情况：控制
执行控制以生成XHTML
是
1410
情况：XSL
处理XSL标记
1412
情况：视图
计算视图
1416
去下一个槽
1414
更多槽?
是
否
结束

图 14