



(12) 发明专利申请

(10) 申请公布号 CN 102939601 A

(43) 申请公布日 2013. 02. 20

(21) 申请号 201180029422. 6

(51) Int. Cl.

(22) 申请日 2011. 04. 27

G06F 17/21 (2006. 01)

(30) 优先权数据

12/769, 839 2010. 04. 29 US

(85) PCT申请进入国家阶段日

2012. 12. 14

(86) PCT申请的申请数据

PCT/US2011/034050 2011. 04. 27

(87) PCT申请的公布数据

W02011/137146 EN 2011. 11. 03

(71) 申请人 单版画股份有限公司

地址 美国马萨诸塞州

(72) 发明人 S. H. 李

(74) 专利代理机构 北京市柳沈律师事务所

11105

代理人 邵亚丽

权利要求书 3 页 说明书 11 页 附图 8 页

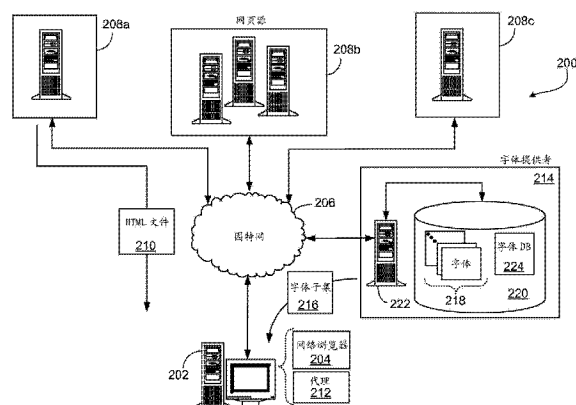
按照条约第19条修改的权利要求书 3 页

(54) 发明名称

启动字体子集

(57) 摘要

一种计算机实施的方法包括以自发的方式分析所接收的电子文件的内容,以识别电子文件的内容中包括的一个或多个字体的每个唯一的字符。该方法还包括发起对电子文件的内容中包括的一个或多个字体的子集的请求,其中,该请求包括各自的一个或多个字体中各个字体的每个识别出的唯一的字符。



1. 一种计算机实施的方法,包括:

以自发的方式分析所接收的电子文件的内容,以识别电子文件的内容中包括的一个或多个字体的每个唯一的字符;并且

发起对电子文件的内容中包括的一个或多个字体的子集的请求,其中,该请求包括一个或多个字体中的各个字体的每个识别出的唯一的字符。

2. 如权利要求 1 所述的计算机实施的方法,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源。

3. 如权利要求 1 所述的计算机实施的方法,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源的不存在。

4. 如权利要求 1 所述的计算机实施的方法,其中,通过运行代理来发起以自发的方式分析所接收的电子文件的内容。

5. 如权利要求 1 所述的计算机实施的方法,其中,分析所接收的电子文件的内容包括解析电子文件的内容以识别该文件中表示的字符。

6. 如权利要求 5 所述的计算机实施的方法,其中,分析所接收的电子文件的内容包括过滤识别出的在电子文件中表示的字符,以识别电子文件的内容中包括的一个或多个字体中的每一个字体的每个唯一的字符。

7. 如权利要求 1 所述的计算机实施的方法,其中,发起对一个或多个字体的子集的请求包括将识别出的字符添加到请求。

8. 如权利要求 1 所述的计算机实施的方法,其中,对所述请求的响应包括所述字体中的一个字体的全部字符。

9. 如权利要求 1 所述的计算机实施的方法,其中,分析电子文件的内容独立于平台来执行。

10. 一种系统,包括:

字体服务器,用于基于请求向计算设备提供用于分析由计算设备接收的电子文件的内容的代理,其中,由该代理执行的分析识别该电子文件的内容中包括的一个或多个字体的每个唯一的字符;

其中,所述字体服务器被配置为基于来自计算设备的请求产生所述一个或多个字体的子集,其中,每个字体子集包括各个字体的每个识别出的唯一的字符;并且

其中,所述字体服务器还被配置为启动发送每个字体子集到计算设备。

11. 如权利要求 10 所述的系统,其中,通过分析所接收的电子文件的内容,所述代理识别所述一个或多个字体中包括的字体的源。

12. 如权利要求 10 所述的系统,其中,通过分析所接收的电子文件的内容,所述代理识别所述一个或多个字体中包括的字体的源的不存在。

13. 如权利要求 10 所述的系统,其中,所述计算设备运行从字体服务器接收的代理,以分析所接收的电子文件的内容。

14. 如权利要求 10 所述的系统,其中,所述代理解析电子文件的内容,以识别该文件中表示的字符。

15. 如权利要求 14 所述的系统,其中,所述代理过滤识别出的在该文件中表示的字符,以识别电子文件的内容中包括的一个或多个字体中的每一个字体的每个唯一的字符。

16. 如权利要求 10 所述的系统,其中,所述代理启动对所述字体服务器的请求的产生。

17. 如权利要求 10 所述的系统,其中,所述代理被配置为独立于计算设备类型来操作。

18. 如权利要求 10 所述的系统,其中,所述字体服务器被配置为确定包括在字体子集中的字体字符的部分。

19. 如权利要求 18 所述的系统,其中,包括在字体子集中的字体字符的部分包括字体的全部字符。

20. 如权利要求 20 所述的系统,其中,所述字体服务器将所产生的字体子集与所接收的电子文件相关联。

21. 一种计算设备,包括:

存储器,被配置为存储指令;和

处理器,被配置为运行所述指令以执行方法,该方法包括:

以自发的方式分析所接收的电子文件的内容,以识别电子文件的内容中包括的一个或多个字体的每个唯一的字符;并且

发起对电子文件的内容中包括的一个或多个字体的子集的请求,其中,该请求包括一个或多个字体中的各个字体的每个识别出的唯一的字符。

22. 如权利要求 21 所述的计算设备,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源。

23. 如权利要求 21 所述的计算设备,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源的不存在。

24. 如权利要求 21 所述的计算设备,其中,通过运行代理来启动以自发的方式分析所接收的电子文件的内容。

25. 如权利要求 21 所述的计算设备,其中,分析所接收的电子文件的内容包括解析电子文件的内容以识别文件中表示的字符。

26. 如权利要求 25 所述的计算设备,其中,分析所接收的电子文件的内容包括过滤识别出的在电子文件中表示的字符,以识别电子文件的内容中包括的一个或多个字体中的每一个字体的每个唯一的字符。

27. 如权利要求 21 所述的计算设备,其中,发起对一个或多个字体的子集的请求包括将识别出的字符添加到请求。

28. 一个或多个计算机可读介质,存储可由处理设备运行的指令,并且当执行这样的指令时,使得处理设备执行操作,所述操作包括:

以自发的方式分析所接收的电子文件的内容,以识别电子文件的内容中包括的一个或多个字体的每个唯一的字符;并且

发起对电子文件的内容中包括的一个或多个字体的子集的请求,其中,该请求包括一个或多个字体中的各个字体的每个识别出的唯一的字符。

29. 如权利要求 28 所述的计算机可读介质,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源。

30. 如权利要求 28 所述的计算机可读介质,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源的不存在。

31. 如权利要求 28 所述的计算机可读介质,其中,通过运行代理来启动以自发的方式

分析所接收的电子文件的内容。

启动字体子集

技术领域

[0001] 本描述涉及启动字体子集的产生和递送 (delivery)。

背景技术

[0002] 在由诸如因特网的计算机网络提供的不断扩展的连接性中,各种类型的内容,如文本、图形、音频、视频等等,可以在一些计算机设备之间交换或广播给大众。由于这样的连接性,遍及全球各个区域的用户可以访问内容提供者,而内容提供者则需要灵活地将内容提供给不同国家和文化的用户。例如,提供给北美洲用户的内容可能不能被亚洲的用户辨别出来 (discernable)。因而,提供者可以为特定地理区域的用户量身定制内容,或者为不同的用户提供可调整的内容。

发明内容

[0003] 这里描述的系统和技术涉及产生和递送字体子集。

[0004] 在一个方面,一种计算机实施的方法包括以自发的方式 (autonomous manner) 分析所接收的电子文件的内容,以识别 (identify) 电子文件的内容中包括的一个或多个字体的每个唯一字符。该方法还包括发起对电子文件的内容中包括的一个或多个字体的子集的请求,其中,该请求包括一个或多个字体中的各个字体的每个识别出的唯一字符。

[0005] 实施方式可以包括任何或全部以下特征。分析电子文件的内容可以包括识别所述一个或多个字体中包括的字体的源 (source) 或字体的源的不存在。以自发的方式分析所接收的电子文件的内容可以通过运行代理 (agent) 来启动。分析所接收的电子文件的内容可以包括解析电子文件的内容以识别文件中表示的字符。分析所接收的电子文件的内容可以包括过滤识别出的在电子文件中表示的字符,以识别电子文件的内容中所包括的一个或多个字体中的每一个字体的每个唯一的字符。启动对一个或多个字体的子集的请求可以包括将识别出的字符添加到请求。对请求的响应可以包括一个或多个字体的全部字符。分析电子文件的内容可以独立于平台而执行。

[0006] 在另一个方面,一种系统包括:字体服务器,用于基于请求向计算设备提供用于分析由计算设备接收的电子文件的内容的代理。由代理执行的分析识别电子文件的内容中包括的一个或多个字体的每个唯一的字符。字体服务器被配置为基于来自计算设备的请求产生一个或多个字体的子集,每个字体子集包括各个字体的每个识别出的唯一的字符。字体服务器还被配置为启动发送每个字体子集到计算设备。

[0007] 实施方式可以包括任何或全部以下特征。通过分析所接收的电子文件的内容,代理可以识别一个或多个字体中包括的字体的源或者字体的源的不存在。计算设备可以运行从字体服务器接收的代理,以分析所接收的电子文件的内容。代理可以解析电子文件的内容,以识别文件中表示的字符。代理可以过滤识别出在该文件中表示的字符,以识别电子文件的内容中包括的一个或多个字体中的每一个字体的每个唯一的字符。代理可以启动产生对字体服务器的请求。代理可以被配置为独立于计算设备类型来操作。字体服务器可以被

配置为确定将要包括在字体子集中的字体字符的部分,并且将要包括在字体子集中的字体字符的部分可以包括字体的全部字符。字体服务器可以将所产生的字体子集与所接收的电子文件相关联。

[0008] 在另一个方面,一种计算设备包括:存储器,其被配置为存储指令;以及处理器,其被配置为运行该指令以执行方法。该方法包括以自发的方式分析所接收的电子文件的内容,以识别电子文件的内容中包括的一个或多个字体的每个唯一的字符。该方法还包括发起对电子文件的内容中包括的一个或多个字体的子集的请求。该请求包括一个或多个字体中的各个字体的每个识别出的唯一的字符。

[0009] 实施方式可以包括任何或全部以下特征。分析电子文件的内容可以包括识别一个或多个字体中包括的字体的源或者字体的源的不存在。以自发的方式分析所接收的电子文件的内容可以通过执行代理来启动。分析所接收的电子文件的内容可以包括解析电子文件的内容以识别文件中表示的字符。分析所接收的电子文件的内容可以包括过滤识别出的在电子文件中表示的字符,以识别电子文件的内容中包括的一个或多个字体中的每一个字体的每个唯一的字符。发起对一个或多个字体的子集的请求包括将识别出的字符添加到请求。

[0010] 在另一个方面,一个或多个计算机可读介质存储可由处理设备执行的指令,并且当执行时使得处理设备执行操作,所述操作包括以自发的方式分析所接收的电子文件的内容,以识别电子文件的内容中包括的一个或多个字体的每个唯一的字符。该操作还包括发起对电子文件的内容中包括的一个或多个字体的子集的请求。该请求包括一个或多个字体中的各个字体的每个识别出的唯一的字符。

[0011] 实施方式可以包括任何或全部以下特征。分析电子文件的内容可以包括识别一个或多个字体中包括的字体的源或者字体的源的不存在。以自发的方式分析所接收的电子文件的内容可以通过执行代理来启动。

[0012] 这些以及其它方面和特征以及它们的各种组合可以表示为方法、装置、系统、用于执行功能的装置、程序产品,以及以其它方式表示。

[0013] 从描述和权利要求中,其它特征和优点将变得清楚。

附图说明

[0014] 图 1 示出可以通过计算机网络传送的示范性内容。

[0015] 图 2 是基于因特网的计算机网络的框图。

[0016] 图 3 示出从字体提供者请求和递送软件代理和字体子集。

[0017] 图 4 是用于产生网页的示范性文件。

[0018] 图 5、图 6 和图 7 示出代码指令的示例。

[0019] 图 8 是软件代理的操作的示例流程图。

[0020] 图 9 是计算设备和系统的框图。

具体实施方式

[0021] 参考图 1, 示范性网页 100 被示出, 其可以由计算设备(例如, 计算机系统、个人数字助理(PDA)、蜂窝式电话等等)的用户通过因特网(或其他类型的计算机网络)来访问。

例如,通过将适当的统一资源定位符(URL)提供给网络浏览器(web browser),网页可以被访问、封包(packaged),并从相应的内容提供者发送到用户的计算设备用于显示。在示出的示例中,网页 100 的内容是生日宴会邀请,其包括文本和图形,并且可以从特定的 URL(即, www.invite.com)访问。为了提供醒目的邀请,使用了提供不同字样(typefaces)的多个字体(font),每个字样可以被认为是提供格式字符(stylistic characters)或字形(glyphs)。“字符”能够被认为是用于组织、控制和表示信息的形状集合的成员,而“字形”能够被认为是字符的具体实例。

[0022] 当访问时,内容供应商可以使用一个或多个技术来将网页 100 的内容提供给用户的计算设备。例如,可以在使用诸如超文本标记语言(HTML)的标记语言的一个或多个文件(例如,文件 102)中表示网页 100,从而网页 100 符合由计算设备运行的浏览器。诸如用于 HTML、CSS、XML 的万维网联盟(World Wide Web Consortium, W3C)标准的标准以及其它标准可以由浏览器实现,从而网页 100 能够适当地呈现在能够显示电子内容的各种类型的平台(例如个人计算机、无线电话、个人数字助理、手持计算机、机顶盒以及因特网设备等等)。

[0023] 除了提供图形(例如,在这种情况下,生日蛋糕的图形)之外,一个或多个 HTML 文件还可以参考用于在网页上绘制文本的一个或多个字体。为了例示,在网页 100 中使用五种不同的字体来提供与所述邀请相关联的信息(例如,时间、日期、位置、事件描述、以及菜单)。同样地,HTML 文件 102 出动(call out)每个字体,从而接收方计算设备知道哪个字体对应于文本信息的每部分(例如,以 Times New Roman 字体表示日期)。当接收和运行 HTML 文件 102 中包括的指令时,接收方计算机设备可以在本地检索表示该文本所需要的字体。然而,相当数量的计算设备不能在本地存储绘制网页可能需要的每个字体的全部字符。有限的资源(例如,存储器)和不断创建的新的字体类型会限制每种类型的计算设备随时以任何以及全部字体显示文本。与不同语言相关联的字体放大许多计算设备不能提供全部字体类型的问题。诸如中文、日文、韩文等的语言使用可能数以万计的字符字母(例如,10000 字符以上),并且需要(call for)1MB 到 20MB 的存储器来存储单一语言的字符。这样的存储器需求是不切实际的,特别是对于具有不太鲁棒的机载(onboard)存储器的计算设备(例如,蜂窝式电话)。而且,如果只需要集合中的几个字符(以呈现特定的网页),则存储完整的字符集合,即使存储少数字体的完整的字符集合,也是低效率的。参考示出的示例,需要相对较少数量的 Times New Roman 字体字符来表示事件日期(例如,“A”、“p”、“r”、“i”、“l”、“2”、“0”、“t”和“h”),因此存储完整的字体字符集合可以被认为是设备存储器的不明智的使用,如果为每种可能的用户语言(例如,英语、中文、日文、等等)都存储完整的字符集合则当然是不明智的。

[0024] 为了节省本地存储器的使用,一些传统的技术为所接收的网页内容(例如,HTML 文件)提供了完整的字体集合,或者在接收到与网页相关联的文件时启动对所需要的字体的完整字符集合的检索。然而,当实际上只有少数一些字符被网页使用时,这样的技术仍然可能使得计算设备存储器被装满收集的字体字符。例如,如附图所示,为了在计算设备上呈现网页 100,完整的字体字符集合 104、106、108、110、112 与 HTML 文件 102 一起提供。同样地,网页 100 使用的五个字体的每个可能的字符被提供给该计算设备,由此消耗了计算设备相当数量的存储器。

[0025] 传送字体所需要的时间以及带宽的消耗也带来了顾虑。例如,传送与诸如中文、日

文和韩文的语言相关联的字体的完整字符集合可能需要相当的带宽和传送时间。

[0026] 一个或多个技术可以用来减少用于准备呈现诸如网页 100 的网页的文件传送时间、带宽消耗和需要的存储器空间。例如,可以提供仅包括出现在网页中的字符的字体子集,而不是为每个字体提供完整的字符集合。如此,减少了文件传送时间和带宽需求,并节省了设备存储器,同时提供字体字符的适当集合用于网页呈现。参考示出的示例,字体字符集合 104-112 中的每个都可以用明显更小的字体子集来替换,所述明显更小的字体子集可以与 HTML 文件 102 一起提供或者可以在接收到该文件之后相对更快地提供。类似于减少基于字符的风格表现的字体子集的传送时间和带宽需求,对于提供与不同语言相关联的字体子集也可以实现减少。例如,如果网页仅包含三百个中文字符,则提供局限在仅包括这三百个字符的字体子集可以更有效率,这可以将传送的字体数据的大小从大约 10MB 减少到 50KB。在一些可以为一些字体字符集合(例如,中文语言的字符集合)产生子集的情形下,其它字体字符集合可以作为完整的集合来发送。例如,由于一个完整的字体集合(例如,拉丁文字符集合)中包括相对较小数目的字符,所以创建和传送子集不可能显著地减少传送整个字符集合所需要的带宽或时间。除了减小提供给计算设备以呈现网页或其它类型的电子文件的字体字符集合的大小之外,还可以实施一个或多个技术来有效地将这样的字体子集提供给计算设备。例如,除了呈现网页之外,用户的计算设备还可以用来识别适当的字体子集,并随后请求所识别的子集。

[0027] 参考图 2,计算机网络 200 包括计算机系统 202,用户可以与计算机系统 202 交互(例如,使用键盘或诸如鼠标的定点设备)以识别将要用该计算机系统呈现的目标网页。例如,网络浏览器 204 或类似的软件应用可以被计算机系统 202 运行,供用户将一个或多个网页确定为目标。当被识别时,网络浏览器 204 的操作可以包括对于目标网页(或多个目标网页)经由因特网 206 从一个或多个网页源 208a、208b、208c 请求内容。如图所示,在这个特定的示例中,从网页源 208a 请求网页,并且相应的 HTML 文件 210 通过因特网 206 从该源发送到计算机系统 202。

[0028] 为了识别适当的字体子集以呈现由 HTML 文件 210 定义的网页,计算机系统 202 可以使用一个或多个技术。例如,可以由计算机系统 202 运行扫描 HTML 文件 210 的操作,以识别由该文件定义的网页中包括的各个字体字符。在一种布置中,计算机系统 202 可以运行软件代理 212 来识别各个字体字符,并发送请求以启动呈现该网页所需的字体子集的产生(如果适当的话)。这样的代理可以被认为是能够以基本自发的方式运行的软件模块。例如,在提供给计算机系统 202 时,软件代理可以操作,而无需大量的用户交互。通过以相当灵活的方式进行操作,软件代理可以自适应地识别网页呈现所需的字体字符。在这个特定示例中,为了识别字体字符,软件代理 212 以相当持续的方式(somewhat persistent manner)扫描 HTML 文件 210 的内容。例如,软件代理可以以基本连续不断的方式运行。在一些布置中,在文件或多个文件(例如,HTML 文件 210)的递送之后,软件代理很快被提供给用户计算设备(例如,计算机系统 202)。因而,诸如网页、应用页、用户界面等的电子文件可以被看成(perceived as)在文件被接收时近乎实时地被扫描。

[0029] 计算机网络 200 还包括字体提供者 214,该字体提供者 214 基于由用户计算设备(例如,计算机系统 202)运行的软件代理(例如,软件代理 212)提供的信息来确定产生字体子集(在适当时)。一旦被产生,子集或多个子集(例如,如字体子集 216 所示)由字体提供

者 214 封装,并发送给请求计算设备。除了确定产生一个或多个字体子集之外,字体提供者 214 也能够确定是否应当将完整的字体字符集合提供给请求计算设备。例如,在确定是否应当发送字体子集时,字体提供者 214 可以使用预定义的规则。一个这样的规则可以指示与特定语言(例如,中文)相关联的字体字符集合应当具有由于完整字符集合很大而创建的子集。子集确定也可以以动态的方式来提供。例如,基于能够实现的文件传送速率,可以确定文件大小阈值(例如,2MB),从而为大于该阈值的字符集合产生子集。如果字体字符集合的大小低于该阈值,则可以发送整个字符集合,因为文件传送速率可以被视为处于有效范围之内。这样的阈值可以被动态地调整,例如,如果监视能够实现的传送速率,可以改变阈值。例如,随着能够实现的传送速率水平的减低,可以相应地减少用于创建字符集合的子集的阈值(例如,从 2MB 下降到 1MB)。可以考虑到一个或多个因素来将传送速率确定为处于有效范围之内。例如,用户计算设备和字体提供者的地理位置可以成为影响(factor)是否应当取代完整字符集合而产生和发送子集的因素。如果字体提供者和用户计算设备两者相距较近(例如,都在美国东部),则能够实现相对较高的传送速率,并且可以发送整个字符集合。对于用户计算设备与字体提供者相距较远的情形(例如,一个在美国,另一个在印度),字体提供者可以确定将要发送字体字符集合的子集。类似于基于位置的确定,一天中的时间、一年中的季节、以及其它的时间因素可以被字体提供者用来确定是否需要为识别出的要传送的一个或多个字体字符集合产生子集。

[0030] 在一些布置中,字体提供者 214 也可以向计算设备提供软件代理,以扫描所接收的文件(例如,HTML 文件 210),从而进行字符识别。同样地,字体提供者 214 可以独立于网页源 208a、208b、208c 而操作。一旦从用户计算设备接收到请求,字体提供者 214 就可以向请求设备提供适当的代理软件。一旦电子文件被软件代理扫描,字体提供者 214 就可以基于该运行的代理发起的请求来提供适当的字体和字体子集。

[0031] 为了提供字体和字体子集,字体提供者 214 一般需要访问一个或多个字体库(library),它们可以存储在字体提供者本地或者存储远离字体提供者的位置。如附图中所表示的,字体库 218 被示出为存储在存储设备 220(例如,一个或多个硬驱、CD-ROM 等)当地(on site)。字体库 218 能够被服务器 222 来访问,并且可以与从软件代理提供的信息一起用于产生能够与完整的字体字符集合一起提供的适当的字体子集(如果被字体提供者确定为适当的)。虽然被示出为存储在单一的存储设备 220 中,但是字体提供者 214 可以使用许多存储技术和存储设备,来保留可访问的字体集合的汇集(例如,对于不同的字体风格、语言、等等)。字体提供者 214 也可以为了子集产生而访问处于其它位置的字体。例如,当识别字体子集所需要的字符时,服务器 222 可以用来从字体提供者 214 外部的一个或多个源收集所需要的字符(例如,经由因特网 206)。

[0032] 除了产生字体子集并且将它们(如果适当,则与完整的字体集合一起)提供给请求计算设备之外,字体提供者 214 还可以提供其它的功能。例如,为了将来的请求,可以跟踪与特定网页相关联的字体和字体子集。在一种场景中,一个或多个字体子集(例如,字体子集 216)可以创建,用于呈现特定网页(在计算设备上)。字体子集和网页之间的关联可以(例如,由服务器 222)识别和存储,用于以后的检索。因而,在将来呈现网页所需的子集(例如,在另一个计算设备上)可以被快速地识别,并提供给请求计算设备。在一种布置中,字体数据库 224 存储在字体提供者 214(例如,在存储设备 220 上),并且包括表示网页和字体子集

(以及字体)之间的关联的记录。在一些实例中,从发送到字体提供者 214 的字体子集请求所提供的信息中识别(例如,从软件代理)所述关联。在网页和适当的字体子集之间的关联也可以在用户选择网页之前被存储(例如,基于网页源所提供的指示和信息)。也可以实施其它类型的体系结构和网络技术,用于提供软件代理和字体子集(以及字体)给用户计算设备,用于呈现诸如网页的电子文件。

[0033] 参考图 3, 示图 300 表示用于将软件代理与呈现其它类型的电子文件的网页所需要的适当的字体子集(以及字体)一起提供给用户的计算设备的一些操作。如图 2 中所示, 当用户用计算设备上运行的网络浏览器识别感兴趣的网页时(例如, 提供 URL 给网络浏览器), 一个或多个文件(例如, HTML 文件)可以从相应的网页源提供给用户的计算设备。如图所示, 这样的文件 302 可以包括用于呈现给用户(经由网络浏览器)的内容 304 (例如, 文本、图形、视频、音频、等等)。文件 302 也可以包含用于请求将软件代理提供给用户的计算设备 202 的一个或多个指令 306。当运行这些指令时(在图中标为“取得代理指令”), 可以从用户计算设备(例如, 计算机系统 202)向字体提供者 214 (例如, 服务器 222)发起请求 308 的递送, 如图中箭头 310 所表示的。响应于请求 308, 代理(例如, 代理 212)被从字体提供者 214 的服务器 222 发送到用户计算设备(例如, 计算机系统 202), 如图中箭头 312 所表示的。在一些实例中, 代理的递送可以在文件被接收到之后很快发生, 并且用户不会注意到任何延迟。在一些布置中, 其它信息可以由请求 308 提供。例如, 可以在请求中识别感兴趣的网页(例如, 所提供的网页的 URL), 从而字体提供者 214 可以确定是否已经预先为该网页产生了一个或多个字体子集(以及可能的字体)。

[0034] 在被用户计算设备接收到时, 所请求的代理 212 被运行以扫描 HTML 文件 302 的内容 304 (如图中箭头 314 所表示的), 以识别该内容中表示的每个字体的字符。代理 212 也可以提供识别出现的每个字体的每个唯一的字符的功能。因而, 同一字体字符的多个实例仅可以被代理标注(note)一次, 由此来合并(consolidate)需要从字体提供者 214 请求的字符(对于每个接收的电子文件)。在一些布置中, 代理 212 将为网页中出现的每个字体识别出的每个字符告知字体提供者 214。在被提供这个信息时, 字体提供者识别可能包含在字体子集中的每个字体的每个唯一出现的字符。为了提供这样的扫描操作, 可以实施一个或多个技术, 例如, 代理可以解析内容 304 以识别每个字体出现的每个字符。然后可以(由代理 212 或字体提供者 214)使用一个或多个过滤器, 来识别每个字体的每个唯一的字符。例如, 如果在内容 304 中检测到字体 A 的字符“a”、“B”和“c”, 并且检测到字体 B 的字符“x”、“Y”和“Z”, 则代理可以将字体 A 的子集识别为包含“a”、“B”和“c”, 而字体 B 的子集可以包含“x”、“Y”和“Z”。在扫描时, 识别出的字体字符 316 被代理 212 用来产生字体子集请求 318。一般来说, 请求 318 包括代理 212 识别出的每个字符, 然而, 页内容 304 的内容中包括的一些字符可能未包括在请求 318 中。例如, 识别为可能存储在用户计算设备本地的字符可能未包括在请求 318 中。因而, 代理 212 可以从子集请求 318 中排除页内容 304 中包括的一些字符。

[0035] 可以实施一个或多个技术来提供字体子集请求 318 给字体提供者 214 的服务器 222, 如图中箭头 320 所表示的。例如, 对于用 JavaScript 表示的代理, 与诸如超文本传输协议(HTTP)的协议相关联的技术可以用来传送请求。通过将所识别的唯一的字符附加(append)到感兴趣的 URL 的查询串(querystring), 可以使用命令(例如, GET 命令)将该信

息提供给服务器。类似地,作为应用提供的代理可以利用诸如 HTTP 的协议将字符信息提供给字体提供者 214 的服务器 222。在被提供对于(多个)字体子集的请求 318 时,服务器 222 产生一个或多个需要的字体子集和完整的字体字符集合(如果适当),并答复(reply)用户计算设备。例如,如图中箭头 322 所表示的,(可以表示一个或多个字体子集以及可能的一个或多个完整的字体字符集合的)字体子集 216 被提供给用户计算设备。

[0036] 参考图 4,示范性 HTML 文件 400 的指令被示出,其包括请求代理(诸如图 2 中所示的代理 212)和分配字体给特定字符。在这个特定示例中,在指令 402 被运行时(例如,被计算机系统 202 运行),向字体提供者(例如,字体提供者 214)请求代理。当被计算机系统 202 接收到时,代理被运行以分析 HTML 文件 400 的内容。例如,代理可以逐步通过(step through)HTML 文件 400 剩余的每一行,并且识别用来呈现与该文件的内容相关联的网页的每个字符和字体。例如,通过分析指令 404,该代理可以识别出需要 frutiger 字体的字符“A”、“B”、“C”和“D”用于网页产生。在这种特定的布置中,各个字符(例如,“A B C D”)与用于访问该字体的 URL 一起由指令 404 提供。类似地,运行的代理在指令 406 中也识别出字符“Z”和“W”需要以 frutiger 字体来呈现,以产生该网页。如此,当产生字体子集请求时(例如,在图 3 中示出的请求 318),该代理识别出产生该网页所需要的每个唯一的字符(即,“A”、“B”、“C”、“D”、“Z”和“W”)和相应的字体(例如,frutiger)。在一些布置中,在扫描文件的内容时,该代理可能遇到(come across)未包括在字体子集请求中的字符。例如,HTML 文件可以包括这样的实例,在该实例中,存储在(运行该文件的)计算机系统本地的字体将被用于表示特定字符。因而,不需要从计算机系统外部的源获得字体。示范性 HTML 文件 400 的指令 408 示出了这样事件的发生(occurrence)。在这个实例中,字符“M”、“P”和“Q”通过指令 408 调出,而不需要对特定字体的 URL。因而,在运行文件 400 的计算机系统本地的字体可以用来呈现字符“M”、“P”和“Q”。由于这些特定字符不需要字体或字体子集,所以该代理不将这些字符包括在请求 318 中。然而,虽然本地存储了针对这个特定字体的这些字符,但是这些字符中的一个或多个可以基于文件 400 中的另一个指令(未示出)而被包括在对另一个字体的请求中,所述另一指令调出从计算机系统外部的源获得的字体中需要的“M”、“P”和 / 或“Q”。

[0037] 参考图 5,可以实施一个或多个技术来分析诸如(图 4 中示出的)HTML 文件 400 的文件的内容,以识别用于字体子集的字符。对于基于 JavaScript 的代理,独立于浏览器的库(称为 jQuery,并且强调 JavaScript 和 HTML 之间的交互)可以用来分析文本内容。为了提供这个功能,jQuery 构架可以用来提供代理,用于从串中提取唯一的字符。jQuery 构架也可以包括关联阵列(associative array)(被称为 JSON),用于形成所识别的唯一的字符和相应的字体之间的关联。以迭代的方式逐步通过文件,唯一的字符被识别和存储(例如,高速缓存)用于进一步处理。附图中给出的部分代码 500 能够提供这个功能。

[0038] 参考图 6,在识别唯一的字符时,一个或多个技术可以被实施以便基于字体相应地将所识别的字符编组(group)。例如,对于特定字体(例如,frutiger)识别的每个唯一的字符(例如,“A”、“B”、“C”、“D”、“Z”和“W”)都是针对该字体的组中的成员。另外,对于具有相对较少成员的字体(例如,与拉丁文相关联的字体),可能不形成唯一的字符的组。对于这样的包括相对较少成员的字体,整个字体集合可以被发送而不会消耗相当多的计算资源,如传送时间和带宽。因而,完整的字体集合被提供(例如,从字体提供者)用于产生该字体的

字符。附图中给出的部分代码 600 能够提供这个功能。

[0039] 参考图 7, 当对于每个字体的识别出的唯一字符已经被编组时(除了具有相对较小的字符集合的任何识别出的字体之外), 代理将这个信息提供给字体提供者 214 (例如, 提供给字体提供者的服务器 222)。一个或多个技术可以用来提供这个信息。例如, 诸如 HTTP GET 命令的命令可以用来将识别出的字符和相应的字体附加到 URL 查询串。在接收到(通过 HTTP GET 命令提供的)请求时, 字体提供者 214 (例如, 字体提供者的服务器 222) 创建和发送一个或多个适当的字体子集给请求计算机系统。基于与一个或多个因素相关联的预定义的规则(例如, 用户和 / 或字体提供者的地理位置、时间信息、诸如能够实现的传送速率的数据传送参数、等等), 对于包括相对较少字符的识别出的字体也可以发送完整的字体集合。为了提供识别出的唯一字符和相应的字体, 该图中给出能够提供这个功能的部分代码 700。

[0040] 参考图 8, 流程图 800 表示软件代理的操作(例如, 图 2 中示出的软件代理 212)。代理 212 的操作一般由单一计算设备(例如, 计算机系统 202) 运行, 然而, 代理的操作可以由多个计算设备运行。除了在单一地点(例如, 在一个计算机系统)运行之外, 操作运行还可以分布在两个或更多的位置上。

[0041] 一般来说, 代理由用户计算设备请求(例如, 从诸如 HTML 文件的文件中包括的代码)并从诸如字体提供者 214 的外部源提供。在由用户计算设备接收和运行时, 操作可以包括分析 802 包括请求软件代理的指令的诸如 HTML 文件的电子文件的内容。通过使用软件代理, 以相对自发和持续的方式分析电子文件的内容。可以实施一个或多个技术来分析电子文件, 例如, 代理可以逐步通过文件, 并且将每个所包括的字符与相应的字体相关联。操作还可以包括识别 804 与字体相关联的每个唯一的字符。例如, 虽然文件可以包括用于特定字体(例如, Times New Roman) 的字符“a”的许多实例, 但是代理可以将所述许多实例过滤到剩下一实例, 从而对于 Times New Roman 字体字符“a”仅被请求一次。字符识别还可以包括识别具有相对较小的字符集合的字体。因而, 如果在电子文件中使用了(这些字体中包括的)这些字符中的一个或多个, 则可以请求整个字符集合。操作还可以包括请求 806 在电子文件中表示的每个字体的适当的子集。所请求的子集一般包括对于每个相应的字体的识别出的唯一的字符。然而, 在字体包括相对较少字符的实例中, 所请求的子集可以包括这样的字体的整个字符集合。

[0042] 图 9 是可以使用和实施以执行与代理 212 相关联的操作的计算系统 900 的框图。计算系统还可以由字体提供者 214 使用以执行操作。计算设备 900 意图用来表示各种形式的数字计算机, 如膝上型计算机、桌上型计算机、工作站、个人数字助理、服务器、刀片服务器、大型机、以及其它适当的计算机。

[0043] 计算设备 900 包括处理器 902、存储器 904、存储设备 906、连接到存储器 904 和高速扩展端口 910 的高速接口 908、以及连接到低速总线 914 和存储设备 906 的低速接口 912。组件 902、904、906、908、910 和 912 中的每一个可以使用各种总线互连, 并且可以安装在公共主板上或者以其它适当的方式安装。处理器 902 可以处理用于在计算设备 900 之内运行的指令, 包括存储在存储器 904 中或存储设备 906 上的指令, 以便在诸如耦接到高速接口 908 的显示器 916 的外部输入 - 输出设备上显示用于 GUI 的图形信息。在其它实施方式中, 可以视情况使用多个处理器和 / 或多个总线, 以及多个存储器和多个类型的存储器。而且,

可以连接多个计算设备 900, 每个设备提供部分必要的操作(例如, 作为服务器群、刀片服务器组、或多处理器系统)。

[0044] 存储器 904 将信息存储在计算设备 900 内。在一个实施方式中, 存储器 904 是计算机可读介质。在一个实施方式中, 存储器 904 是一个或多个易失性存储器单元。在另一个实施方式中, 存储器 904 是一个或多个非易失性存储器单元。

[0045] 存储设备 906 能够为计算设备 900 提供大容量存储。在一个实施方式中, 存储设备 906 是计算机可读介质。在各种不同的实施方式中, 存储设备 906 可以是软盘设备、硬盘设备、光盘设备、或者磁带设备、闪存或其它类似的固态存储设备, 或者设备的阵列, 包括存储区域网络或其它配置中的设备。在一个实施方式中, 计算机程序产品有形地体现在信息载体中。计算机程序产品包含指令, 当所述指令被运行时, 执行一个或多个方法, 诸如上述那些方法。信息载体是计算机可读介质或机器可读介质, 诸如存储器 904、存储设备 906、处理器 902 上的存储器、等等。

[0046] 高速控制器 908 管理用于计算设备 900 的带宽密集(bandwidth-intensive)的操作, 而低速控制器 912 管理带宽较不密集(lower bandwidth-intensive)的操作。这样的任务分配仅仅是示范性的。在一个实施方式中, 高速控制器 908 耦接到存储器 907、显示器 916(例如, 通过图形处理器或加速器), 并且耦接到高速扩展端口 910, 高速扩展端口 910 能够接受各种扩展卡(未示出)。在所述实施方式中, 低速控制器 912 耦接到存储设备 906 和低速扩展端口 914。可以包括各种通信端口(例如, USB、蓝牙、以太网、无线以太网)的低速扩展端口可以耦接到一个或多个输入-输出设备, 诸如键盘、定点设备、扫描仪、或诸如开关或路由器的网络设备, 例如, 通过网络适配器。

[0047] 计算设备 900 可以以许多不同的形式来实施, 如图中所示。例如, 其可以实施为标准服务器 920, 或者这样的服务器的组的多倍。其还可以实施为机架式服务器系统 924 的一部分。此外, 其可以实施在诸如膝上型计算机 922 的个人计算机中。或者, 来自计算设备 900 的组件可以与移动设备(未示出)中的其它组件组合。

[0048] 这个说明书中描述的本主题的实施例和功能操作可以实施在数字电子电路中、或者实施在计算机软件、固件、或硬件中、所述硬件包括在这个说明书中公开的结构以及它们的结构等效物, 或者实施在它们的一个或多个的组合中。在这个说明书中描述的本主题的实施例可以实施为一个或多个计算机程序产品, 即, 被编码在计算机可读介质上的用于由数据处理设备运行或者控制数据处理设备的操作的计算机程序指令的一个或多个模块。计算机可读介质可以是机器可读存储设备、机器可读存储基底(substrate)、存储器件、影响机器可读传播信号的物质的合成物、或者它们的一个或多个的组合。术语“数据处理装置”包括用于处理数据的所有装置、设备、以及机器, 包括, 举例来说, 可编程处理器、计算机、或多处理器或计算机。除了硬件以外, 所述装置还可以包括创建用于所讨论的计算机程序的运行环境的代码, 例如, 构成处理器固件、协议组、数据库管理系统、操作系统、或者它们的一个或多个的组合的代码。

[0049] 计算机程序(也称为程序、软件、软件应用、脚本、或代码)可以以任何形式的编程语言来编写, 包括编译或解释语言, 并且其可以以任何形式来部署, 包括作为单独的程序或者作为模块、组件、子例程、或者适于在计算环境中使用的其它单元。计算机程序不必对应于文件系统中的文件。程序可以存储在保存其它程序或数据的文件的一部分中(例如, 存储

在标记语言文件中的一个或多个脚本),存储在专用于所讨论的程序的单一文件中,或者存储在多个协调的文件中(例如,存储一个或多个模块、子程序、或部分代码的文件)。计算机程序可以被部署为在一个计算机或位于一个地点或分布在多个地点并通过通信网络互连的多个计算机上运行。

[0050] 在本说明书中描述的过程和逻辑流可以由运行一个或多个计算机程序的一个或多个可编程处理器来执行,以便通过对输入数据进行操作并生成输出来执行功能。所述过程和逻辑流还可以由特殊用途的逻辑电路来执行,并且装置也可以实施为特殊用途的逻辑电路,所述特殊用途的逻辑电路,例如,FPGA(现场可编程门阵列)或ASIC(专用集成电路)。

[0051] 适于运行计算机程序的处理器包括,举例来说,全部通用和专用微处理器,以及任意种类的数字计算机的任意一个或多个处理器。一般,处理器将从只读存储器或随机存取存储器或者从两者接收指令和数据。计算机的主要元件是用于执行指令的处理器和用于存储指令和数据的一个或多个存储设备。一般,计算机还将包括用于存储数据的一个或多个大容量存储设备,或者被操作地耦接以便从所述大容量存储设备接收数据或向所述大容量存储设备传送数据,或者两者都有,所述大容量存储设备例如磁盘、磁光盘、或者光盘。然而,计算机并不必具有这样的设备。而且,计算机可以嵌入另一个设备,例如,移动电话、个人数字助理(PDA)、移动音频播放器、全球定位系统(GPS)接收器,仅举数例。适合存储计算机程序指令和数据的计算机可读介质包括所有形式的非易失性存储器、介质和储存设备,举例来说,包括半导体存储器件,例如,EPROM、EEPROM、闪存器件;磁盘,例如,内部硬盘或可移动盘;磁光盘;以及CD-ROM和DVD-ROM盘。处理器和存储器可以通过专用逻辑电路来补充,或者合并到专用逻辑电路中。

[0052] 在本说明书中描述的主题的实施例可以实施在计算系统中,该计算系统包括后端组件,例如,作为数据服务器,或者包括中间件组件,例如,作为应用服务器,或者包括前端组件,例如,作为具有图形用户界面或网络浏览器的客户端计算机,用户通过所述图形用户界面或网络浏览器可以与在本说明书中描述的本主题的实施方式交互,或者包括一个或多个这样的后端组件、中间件、或前端组件的任意组合。系统的组件可以通过任何形式或介质的数字数据通信,例如,通信网络,互连。通信网络的示例包括局域网(“LAN”)和广域网(“WAN”),例如,因特网。

[0053] 计算系统可以包括客户端和服务端。客户端和服务端一般彼此远离,并且典型地通过通信网络进行交互。客户端和服务端的关系根据运行在各自计算机上并且彼此具有客户端-服务端关系的计算机程序而产生。

[0054] 虽然这个说明书包含了许多具体细节,但是这些不应被解释为对本发明的或可以要求的范围的限制,而是对本发明的特定实施例的特有特征的描述。在本说明书中在单独的实施例的上下文中描述的某些特征也可以组合地实施在单一实施例中。反之,在单一实施例的上下文中描述的各种特征也可以被单独地或者以任何适当的子组合(subcombination)的方式实施在多个实施例中。而且,虽然特征在以上描述为以一定的组合动作,甚至最初这样要求(claim),但是在一些情况下,所要求的组合中的一个或多个特征也可以从所述组合中消除,并且所要求的组合也可以指示子组合或者子组合的变体。

[0055] 类似地,虽然在附图中以特定的次序描绘了操作,但是这不应被理解为要求这样的操作以示出的特定次序或顺序的次序来执行,或者全部示出的操作都要执行,以达到期

望的结果。在某些情形下,多任务处理或并行处理也可以是有利的。而且,在如上所述的实施例中的各种系统组件的分离不应被理解为在所有实施例中都要求这样的分离,并且应当理解,所描述的程序组件和系统一般可以整合到单一的软件产品中,或者封装到多个软件产品中。

[0056] 因此,已经描述了本发明的特定实施例。其它实施例也在权利要求的范围内。例如,在权利要求中记载的动作也可以以不同的次序来执行,并且仍然能够达到期望的结果。

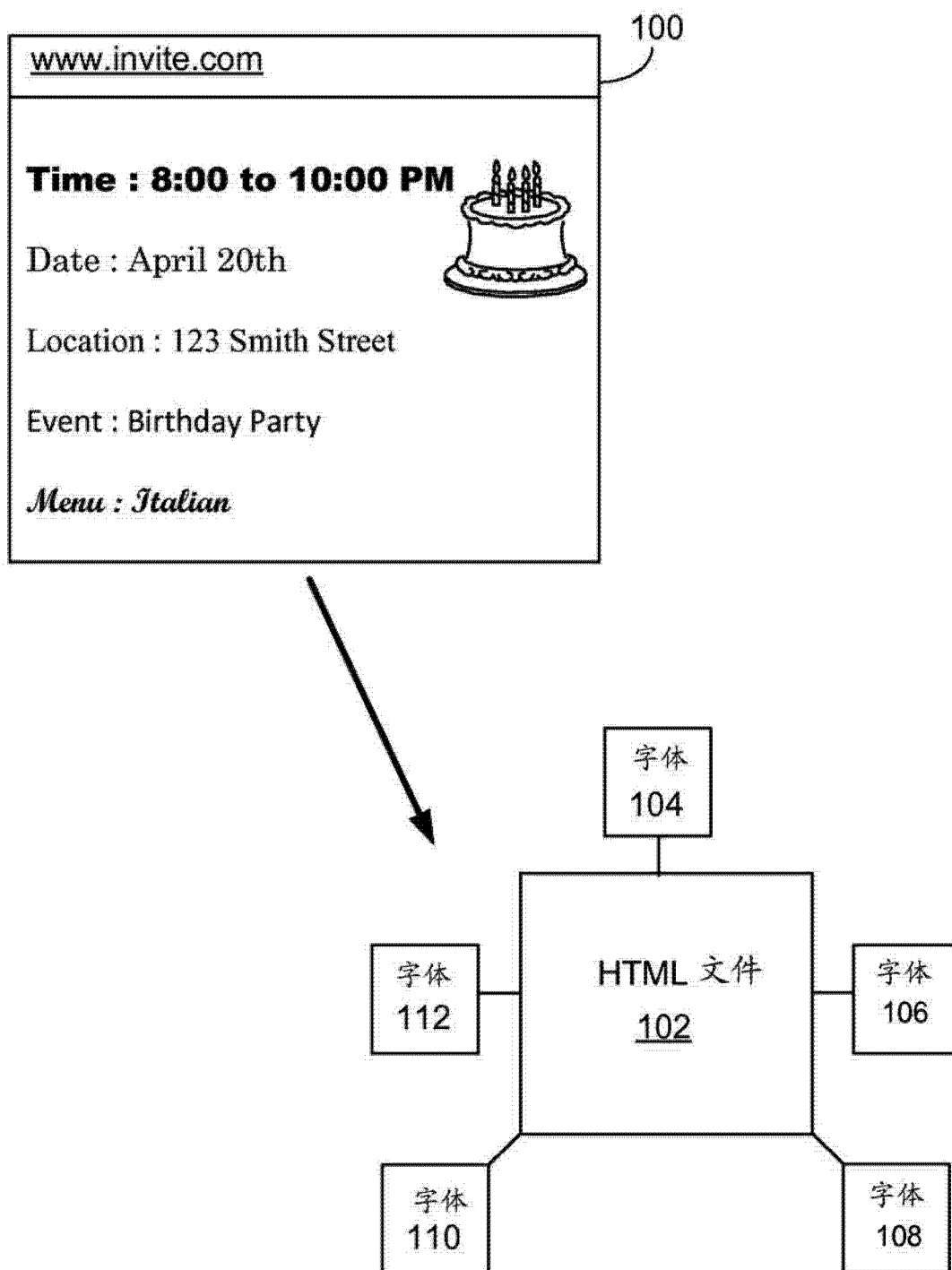


图 1

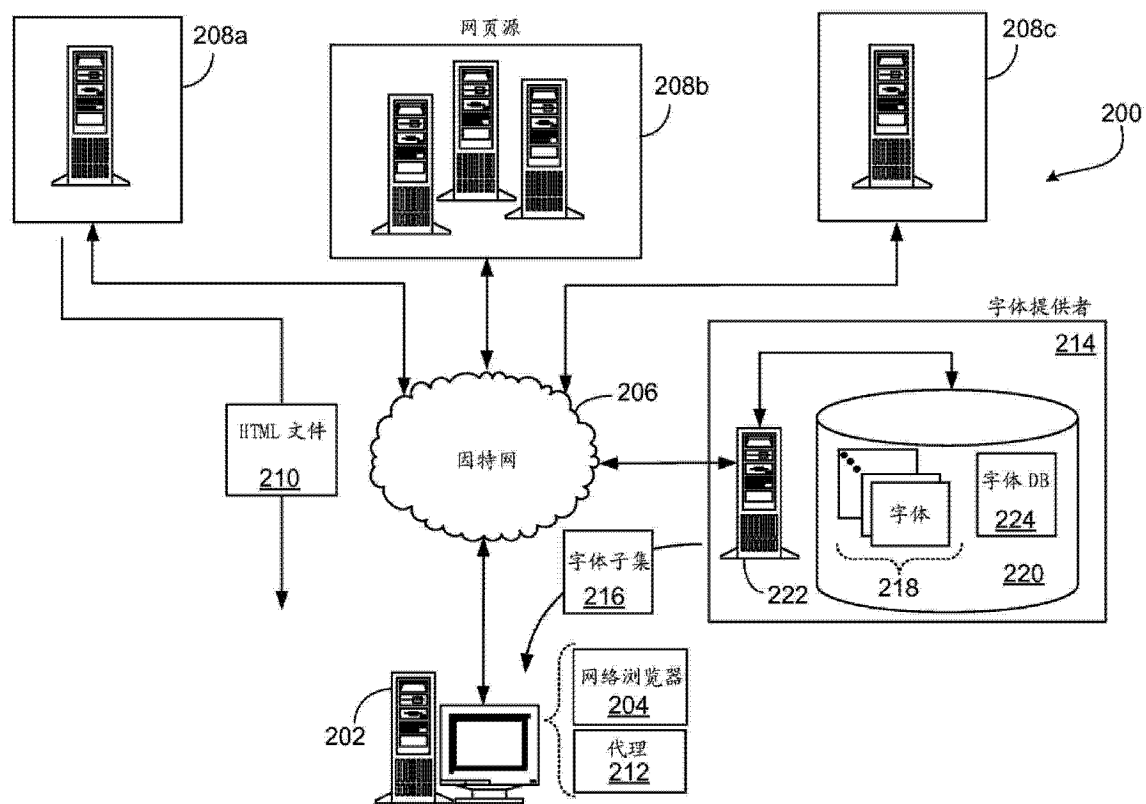


图 2

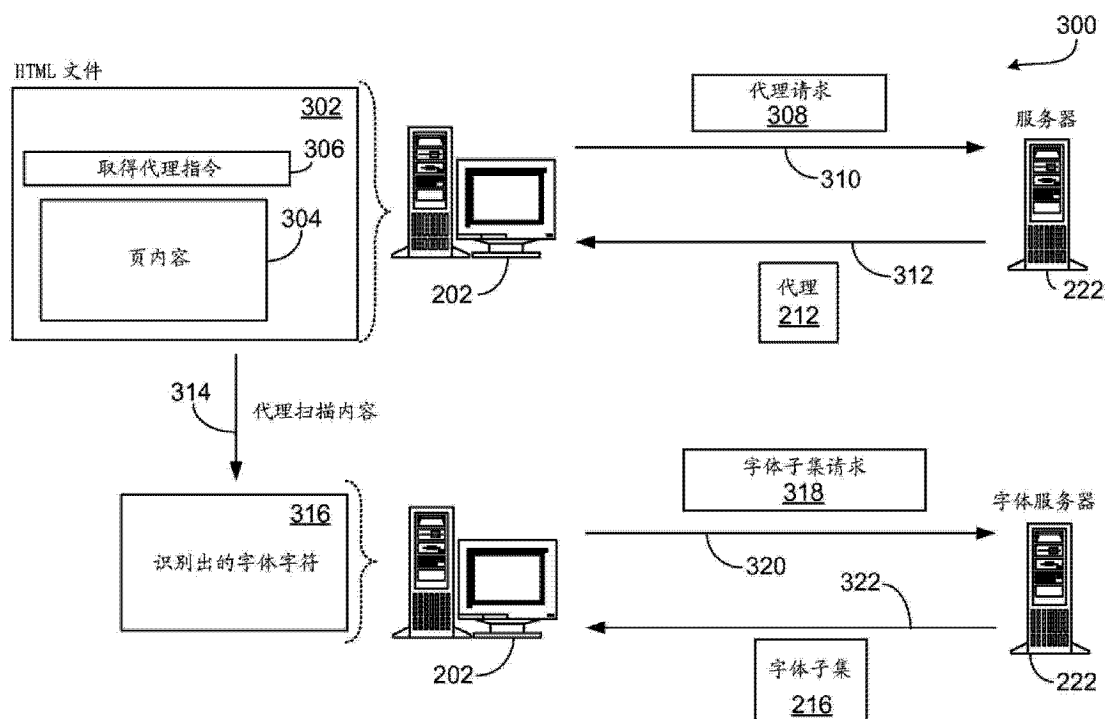


图 3

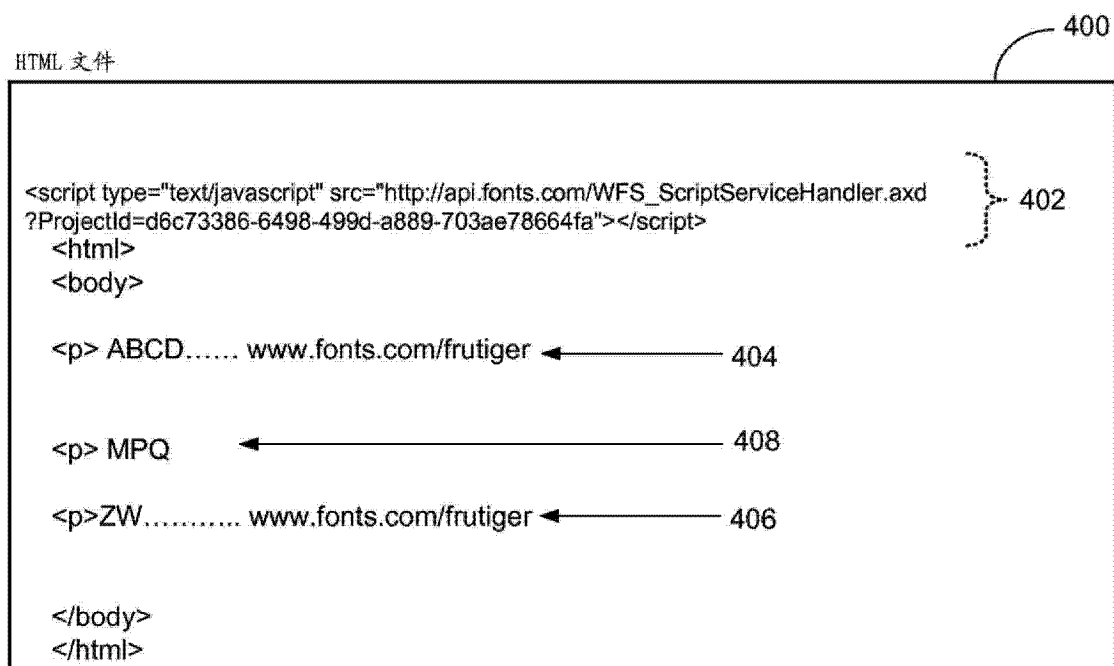


图 4

```
500 { $.each(projectSelectorList, function(selector, font){  
    var $val = $(selector);  
  
    elementCache.push($val.css({'visibility':'hidden'}));  
    //if dynamicsubsetting is required extract text from the  
    selected elements  
    if(requiredDynamicSubsetting)  
    {  
        //accumulate the text for specific font  
        if(queryFontList[font])  
            queryFontList[font] += $val.text();  
        else  
            queryFontList[font] = $val.text();  
    }  
});
```

图 5

```

600 {
    $('*', $body).filter(function(){
        var $this = $(this);
        var hasProjectFont = false;
        var fontFamily = $this.css('font-family');

        $.each(projectFontList, function(i, fontName){
            if(fontFamily.replace(/^\s|\s$/g, '') == fontName.replace(/^\s|\s$/g, ''))
            {
                // if the current element is not in element cache
                if($.inarray($this, elementCache) < 0)
                {
                    elementCache.push($this.css({'visibility':'hidden'}));
                    if(requiredDynamicSubsetting)
                    {
                        if(queryFontList[fontFamily])
                        {
                            queryFontList[fontFamily] += $this.text();
                        }
                        else
                        {
                            queryFontList[fontFamily] = $this.text();
                        }
                    }
                }
                hasProjectFont = true;
            }
        });
    });
    return hasProjectFont;
});

function extractUniqueChars(node, ignoreWhiteSpace) {
    var text = extractText(node, ignoreWhiteSpace);
    var uniq = "";
    var textLength = text.length;
    var curChar = null;
    var charAt = text.charAt;
    for (var i = 0; i < textLength; i++) {
        curChar = text.charAt(i);
        if (uniq.indexOf(curChar) == -1) {
            uniq += curChar;
        }
    }
    return uniq;
}

```

图 6

```

700 {
    if(requiredDynamicSubsetting)
    {
        $.each(queryFontList, function(i, fontName){
            //Mention which font uses what characters, to server
            queryString += '&' + fontName + '=' +
            escape($.extractUniqueChars(val));
        });
        MTI_WFSurl += queryString + 'g';
    }
    head.append('<link rel=""stylesheet"" type=""text/css""
                href=""'+MTI_WFSurl+'"" />');
}

```

图 7

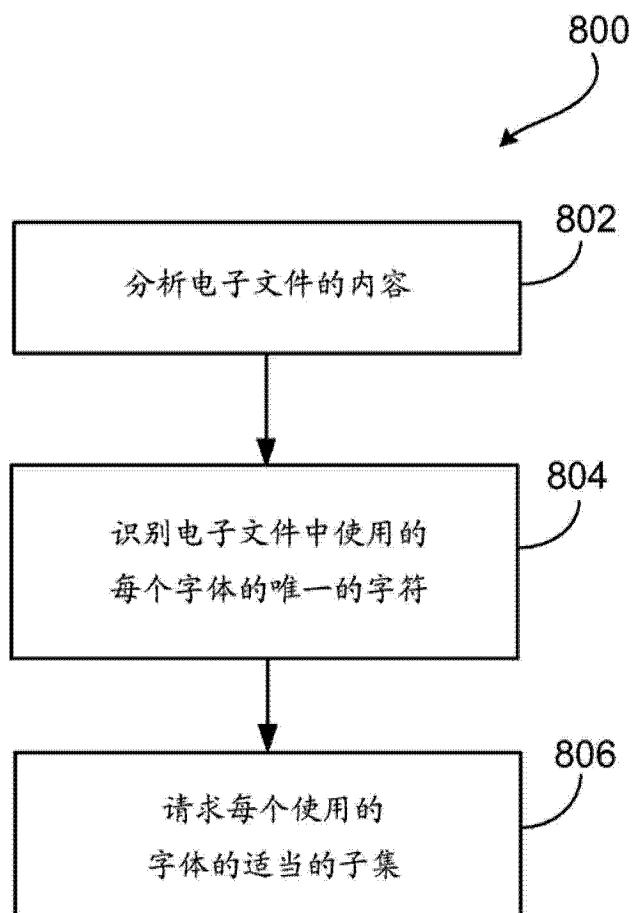


图 8

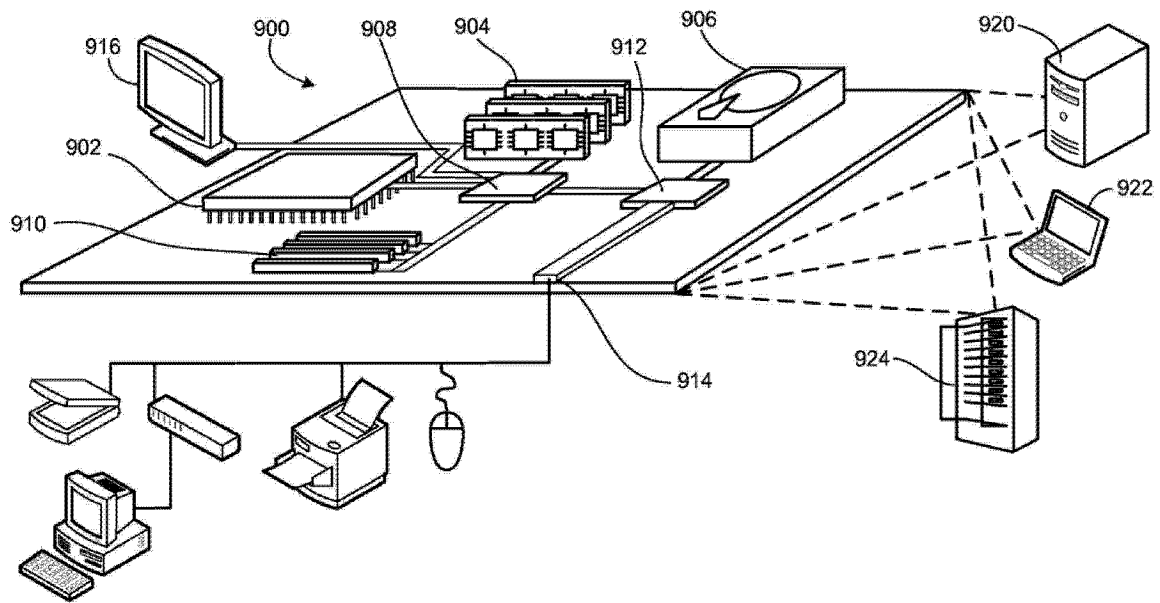


图 9

1. 一种计算机实施的方法,包括:

以自发的方式分析所接收的电子文件的内容,以识别电子文件的内容中包括的一个或多个字体的每个唯一的字符;并且

发起对电子文件的内容中包括的一个或多个字体的子集的请求,其中,该请求包括一个或多个字体中的各个字体的每个识别出的唯一的字符。

2. 如权利要求1所述的计算机实施的方法,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源。

3. 如权利要求1所述的计算机实施的方法,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源的不存在。

4. 如权利要求1所述的计算机实施的方法,其中,通过运行代理来发起以自发的方式分析所接收的电子文件的内容。

5. 如权利要求1所述的计算机实施的方法,其中,分析所接收的电子文件的内容包括解析电子文件的内容以识别该文件中表示的字符。

6. 如权利要求5所述的计算机实施的方法,其中,分析所接收的电子文件的内容包括过滤识别出的在电子文件中表示的字符,以识别电子文件的内容中包括的一个或多个字体中的每一个字体的每个唯一的字符。

7. 如权利要求1所述的计算机实施的方法,其中,发起对一个或多个字体的子集的请求包括将识别出的字符添加到请求。

8. 如权利要求1所述的计算机实施的方法,其中,对所述请求的响应包括所述字体中的一个字体的全部字符。

9. 如权利要求1所述的计算机实施的方法,其中,分析电子文件的内容独立于平台来执行。

10. 一种系统,包括:

字体服务器,用于基于请求向计算设备提供用于分析由计算设备接收的电子文件的内容的代理,其中,由该代理执行的分析识别该电子文件的内容中包括的一个或多个字体的每个唯一的字符;

其中,所述字体服务器被配置为基于来自计算设备的请求产生所述一个或多个字体的子集,其中,每个字体子集包括各个字体的每个识别出的唯一的字符;并且

其中,所述字体服务器还被配置为启动发送每个字体子集到计算设备。

11. 如权利要求10所述的系统,其中,通过分析所接收的电子文件的内容,所述代理识别所述一个或多个字体中包括的字体的源。

12. 如权利要求10所述的系统,其中,通过分析所接收的电子文件的内容,所述代理识别所述一个或多个字体中包括的字体的源的不存在。

13. 如权利要求10所述的系统,其中,所述计算设备运行从字体服务器接收的代理,以分析所接收的电子文件的内容。

14. 如权利要求10所述的系统,其中,所述代理解析电子文件的内容,以识别该文件中表示的字符。

15. 如权利要求14所述的系统,其中,所述代理过滤识别出的在该文件中表示的字符,以识别电子文件的内容中包括的一个或多个字体中的每一个字体的每个唯一的字符。

16. 如权利要求 10 所述的系统,其中,所述代理启动对所述字体服务器的请求的产生。

17. 如权利要求 10 所述的系统,其中,所述代理被配置为独立于计算设备类型来操作。

18. 如权利要求 10 所述的系统,其中,所述字体服务器被配置为确定包括在字体子集中的字体字符的部分。

19. 如权利要求 18 所述的系统,其中,包括在字体子集中的字体字符的部分包括字体的全部字符。

20. 如权利要求 10 所述的系统,其中,所述字体服务器将所产生的字体子集与所接收的电子文件相关联。

21. 一种计算设备,包括:

存储器,被配置为存储指令;和

处理器,被配置为运行所述指令以执行方法,该方法包括:

以自发的方式分析所接收的电子文件的内容,以识别电子文件的内容中包括的一个或多个字体的每个唯一的字符;并且

发起对电子文件的内容中包括的一个或多个字体的子集的请求,其中,该请求包括一个或多个字体中的各个字体的每个识别出的唯一的字符。

22. 如权利要求 21 所述的计算设备,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源。

23. 如权利要求 21 所述的计算设备,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源的不存在。

24. 如权利要求 21 所述的计算设备,其中,通过运行代理来启动以自发的方式分析所接收的电子文件的内容。

25. 如权利要求 21 所述的计算设备,其中,分析所接收的电子文件的内容包括解析电子文件的内容以识别文件中表示的字符。

26. 如权利要求 25 所述的计算设备,其中,分析所接收的电子文件的内容包括过滤识别出的在电子文件中表示的字符,以识别电子文件的内容中包括的一个或多个字体中的每一个字体的每个唯一的字符。

27. 如权利要求 21 所述的计算设备,其中,发起对一个或多个字体的子集的请求包括将识别出的字符添加到请求。

28. 一个或多个计算机可读介质,存储可由处理设备运行的指令,并且当执行这样的指令时,使得处理设备执行操作,所述操作包括:

以自发的方式分析所接收的电子文件的内容,以识别电子文件的内容中包括的一个或多个字体的每个唯一的字符;并且

发起对电子文件的内容中包括的一个或多个字体的子集的请求,其中,该请求包括一个或多个字体中的各个字体的每个识别出的唯一的字符。

29. 如权利要求 28 所述的计算机可读介质,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源。

30. 如权利要求 28 所述的计算机可读介质,其中,分析电子文件的内容包括识别一个或多个字体中包括的字体的源的不存在。

31. 如权利要求 28 所述的计算机可读介质,其中,通过运行代理来启动以自发的方式

分析所接收的电子文件的内容。

32. 一种系统,包括:

字体服务器,用于基于来自计算设备的请求,产生一个或多个字体的子集,其中,每个字体子集包括由所述计算设备中包括的代理从电子文件的内容中识别出的各个字体的每个唯一的字符;以及

其中,所述字体服务器被配置来启动发送每个字体子集给所述计算设备。

33. 如权利要求 32 所述的系统,其中,所述字体服务器被配置来将所述代理提供给所述计算设备。

34. 如权利要求 32 所述的系统,其中,所述代理分析所述电子文件的内容,以识别一个或多个字体中包括的字体的源。

35. 如权利要求 32 所述的系统,其中,所述代理分析所接收的电子文件的内容,以识别一个或多个字体中包括的字体的源的不存在。

36. 如权利要求 32 所述的系统,其中,所述代理解析所述电子文件的内容,以识别文件中表示的字符。

37. 如权利要求 36 所述的系统,其中,所述代理过滤识别出的在电子文件中表示的字符,以识别所述电子文件的内容中包括的一个或多个字体中的每一个字体的每个唯一的字符。

38. 如权利要求 32 所述的系统,其中,所述代理启动产生对所述字体服务器的请求。

39. 如权利要求 32 所述的系统,其中,所述代理被配置来独立于计算设备类型而操作。

40. 如权利要求 32 所述的系统,其中,所述字体服务器被配置来确定要包括在字体子集中的字体字符的部分。

41. 如权利要求 40 所述的系统,其中,所述要包括在字体子集中的字体字符的部分包括所述字体的全部字符。

42. 如权利要求 32 所述的系统,其中,所述字体服务器将所产生的字体子集关联到所述电子文件。