



(12) 发明专利

(10) 授权公告号 CN 102739670 B

(45) 授权公告日 2015. 06. 17

(21) 申请号 201210213797. 4

H04N 21/262(2011. 01)

(22) 申请日 2003. 07. 31

H04N 21/4147(2011. 01)

(30) 优先权数据

H04N 21/4335(2011. 01)

60/403, 995 2002. 08. 17 US

H04N 21/462(2011. 01)

10/632, 003 2003. 07. 30 US

(62) 分案原申请数据

03824299. 0 2003. 07. 31

(56) 对比文件

W0 00/48375 A1, 2000. 08. 17,

W0 00/48375 A1, 2000. 08. 17,

W0 01/98920 A1, 2001. 12. 27,

(73) 专利权人 迪斯尼实业公司

审查员 张俊杰

地址 美国加利福尼亚州

(72) 发明人 斯科特·F·瓦特森

埃里克·C·赫塞尔廷

埃里克·弗里曼

伊丽莎白·M·弗里曼

亚伦·P·拉伯格

亚当·T·弗里特斯

(74) 专利代理机构 中科专利商标代理有限责任

公司 11021

代理人 王玮

(51) Int. Cl.

H04L 29/06(2006. 01)

H04L 29/08(2006. 01)

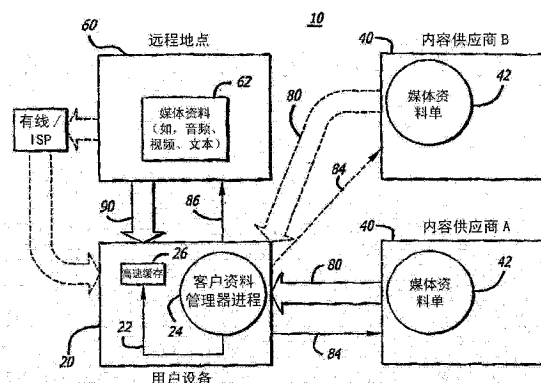
权利要求书1页 说明书18页 附图4页

(54) 发明名称

经带宽受限的网络传输与动态再现大型媒体资料的系统

(57) 摘要

本发明为经带宽受限的网络传输与动态再现大型媒体资料的系统。基于一组预先设定的条件，把媒体内容从内容供应商传输到用户设备的高速缓存，以供以后浏览。一个客户资料管理器进程驻留于用户设备，资料单处于内容供应商的地点，媒体资料位于远程地点。



1. 一种通过网络从内容供应商传输媒体资料以与实时内容流结合的方法,所述方法包括:

由内容供应商通过网络把媒体资料单提供给用户设备;

如果满足预定约束,由内容供应商通过网络把媒体资料单中的选定媒体资料通过网络传输给用户设备以存储,从而无需对选定媒体资料进行实时缓冲以供显示;以及

在向用户设备完全传输选定媒体资料之后,由内容供应商通过网络向用户设备单独地传输实时内容流,所述实时内容流需要实时缓冲以供显示;

其中相比于选定媒体资料的传输,实时内容流的传输需要的网络带宽更低,以便于选定媒体资料与实时内容流结合来形成一个集成内容以供显示。

2. 根据权利要求1所述的方法,其中媒体资料是音频内容、视频内容、文本内容、使用许可的权利、或多媒体文件中的至少一种。

3. 根据权利要求1所述的方法,其中媒体资料单由来自用户的请求而生成。

4. 根据权利要求1所述的方法,进一步包括访问内容供应商的内容网站。

5. 根据权利要求1所述的方法,其中预定约束是用户设备正闲置,网络服务质量(QoS),或者带宽使用情况低于预先设定的运行水平中的至少一种。

6. 根据权利要求1所述的方法,其中预定约束是用户设备CPU使用情况,或用户设备中的存储器使用情况中的至少一个低于预先设定的运行水平。

7. 根据权利要求1所述的方法,其中客户进程发起资料的传输,从内容供应商通过网络传输给用户设备。

8. 根据权利要求1所述的方法,其中媒体资料存储于本地高速缓存。

9. 根据权利要求1所述的方法,其中所述一个集成内容用于在网络浏览器页面上显示。

10. 根据权利要求1所述的方法,其中预定约束包括一天中的某个时间。

11. 根据权利要求8所述的方法,进一步包括从用户设备CPU使用情况,带宽使用情况,本地高速缓存使用情况,和用户设备活动计时器中确定至少一个参数。

12. 根据权利要求8所述的方法,进一步包括当资料没有在用户端就绪时,把替代资料与来自内容供应商的实时内容一起呈现。

经带宽受限的网络传输与动态再现大型媒体资料的系统

[0001] 本申请是申请号为 03824299.0、申请日为 2003 年 7 月 31 日的中国专利申请“经带宽受限的网络传输与动态再现大型媒体资料的系统”的分案申请。

技术领域

[0002] 本发明涉及通过网络把基于媒体的高质量内容传输与再现给用户或进程,其中即使在没有充足的网络带宽可供使用的情况下也能以保证的服务质量(QoS)来传输和再现。

背景技术

[0003] 发明人意识到,即使随着网络技术的发展,传输丰富的高质量的媒体体验仍是一个难题。尤其是传输大型媒体资料,无论它们是音频、视频、FLASH、游戏、数据或其它媒体格式,通常要求比可利用的更多的带宽/网络流量。例如,在音频和视频的情况下,如果用户的有效带宽至少等于资料的比特率,高比特率的资料只能被实时地传输,否则将导致不理想的用户体验,包括发生停顿、停止以及内容缓冲。

[0004] 另一方面,一个大型的可执行游戏可能不与视频一样有实时的限制(或要求的服务质量),然而,下载该资料需要大量的时间和用户开销,即使在最快的网络中也是这样。尽管市场上出现的“下载管理员”可以为用户照顾这些方面,但内容供应商仍然希望以一种得体的、透明的、不需要用户操心的方式,并且智能地、可定制地管理资料下载到用户设备上(比如,计算机、有存储器与/或处理器的机顶盒,或其它设备)。

[0005] 鉴于此,存在这样的需求,即,使用有限的带宽以时间可选的方式来管理和传输大型的、高质量的媒体资料给用户。也就是说,有这样一种需求,要求通过现有的带宽和空闲时隙,顺畅地把内容传输给用户,从而当需要内容的时候,它已经应需要而就绪了,用户体验不打折扣。这自然就引发了一种期望,即用户拥有比实际可利用的更有效的带宽。为此,也需要把这种技术无缝地集成于传输与再现平台(包括但不限于网络浏览器、FLASH 动画和其它平台)以及内容发布系统。本发明达到了这一目的,而且实现了其它功能,并克服了现有技术的局限。

[0006] 为便于理解,下述定义将适用于整个申请文件中。然而,不应该认为这些定义替代了对本领域中已接受的,所列出的术语的理解。

[0007] 术语定义:

[0008] 1. 流量 - 在规定时间内,从一地传输到另一地的数据量,通常,流量用 kbps(千字节每秒)、Mbps(兆字节每秒)和 Gbps(千兆字节每秒)来量度。

[0009] 2. 服务质量, QoS- 该术语代表规定一个受保证的流量水平。

[0010] 3. 客户进程 - 在客户端执行的进程,它接收来自服务器进程的高速缓存/显示管理指令或提示并执行这些指令,从而把高速缓存现在的状态变成期望的状态,还可在做这些的过程中发起一个或更多的通知给用户。

[0011] 4. 高速缓存 - 用“已知”条件或 QoS 对资料的存储。该上下文中所说的高速缓存是一种资料存储机制,其中的 QoS 满足内容需要并且通常高于用来获取资料的媒体。高速

缓存内状态的变化可能导致通知的发生。

[0012] 5. 服务器进程 - 向用户提供为管理高速缓存的状态所需要的信息。它最简单的实现方式是,它类似于一个准动态的、由服务器生成的播放节目单。更详细的实现(都是本发明的)还提供控制指令,以便客户把资料的具体一部分的进展通知给其它进程的客户。

[0013] 6. 到期日 - 资料的到期日,并指出何时资料应当从本地高速缓存中移除。

[0014] 7. 回叫 URL - 资料项目被下载后又重新获得的 URL。

[0015] 8. 用户端令牌 - 当资料项目被下载以后设置的令牌或 COOKIE。这允许客户或服务应用确定一项资料在本地系统中的存在。

[0016] 9. 禁止日 - 表示资料将被使用的最近日期。

[0017] 10. 删除 - 表示资料将被标记成明确删除(覆盖到期日)。这便于资料的收回。

[0018] 11. 更新率 - 确定客户如何频繁地检查要改变的资料列表。

[0019] 12. 资源路径 - 与资源列表有关的任何数量的资源的网络地址。

[0020] 13. 媒体资料 - 文本、音频、视频,或二进制文件/数据中的至少一种。

[0021] 14. 项目 - 单个媒体文件

[0022] 15. 连接 - 用于媒体文件的 URL

[0023] 16. hitCountURL(计时 URL) - 文件被成功下载后要 ping 接的 URL,一个参数,即时长,被附加到 URL 的尾部,以秒为单位表示下载花了多长时间。

[0024] 17. hlpURL(帮助 URL) - 用于用户进程帮助的 URL,在用户选择帮助菜单项目时被显示。

[0025] 18. trackWithCookie(跟踪 COOKIE) - 一个可选单元,当它存在的时候,表示这个资料被添加到被 COOKID 名所指定的 COOKIE 中的资料单中。

[0026] 19. cookieName(COOKIE 名) - 列举了所有被下载的、具有跟踪 COOKIE 单元存在的资料的 COOKIE 的名称。这种 COOKIE 对广告服务是很必要的,以使广告服务器知道哪些广告已经被下载。COOKIE 的这种格式将仅包含文件的名称(无扩展名或路径),由逗号分隔。

[0027] 20. cookieDomain(COOKIE 域) - 将被下载的资料 COOKIE 设置于其中的域。如果用半冒号或逗号分隔,可以指定多个域。

[0028] 21. /regserver(/注册服务器) - 把 ActiveX 控件注册到系统并把客户进程添加到启动文件夹。

[0029] 22. /shutdown(/关机) - 停止另一个在运行客户进程的程序,如果存在的话。

[0030] 23. /unregserver(/解注册服务器) - 解注册 ActiveX 控件,并把快捷键从启动文件夹中卸除。它还停止正在运行的客户进程 .exe 程序,并卸除 COM 对象寄存器条目。

[0031] 24. CDN - 内容分布网络,由第三方拥有和操作的、连结成组的内容服务器。通常,CDN 服务使用高度分散的一群服务器,可以提供额外的业务能力。

发明内容

[0032] 本发明提供一种系统和方法,其中媒体内容基于预先设定的一组条件,在浏览媒体之前,从内容供应商传输到一个用户设备的本地高速缓存。一个资料单含有与要被下载到客户设备的媒体资料有关的信息,并被从内容提供商传输到用户设备。例如,资料单包含有关媒体资料的位置的 URL 或信息。

[0033] 客户资料管理器进程驻留于用户设备,负责从内容提供商下载资料。该资料管理器使用资料单,来请求获得位于远程地点的媒体资料。当具体的条件被满足时,客户进程管理媒体定期地向用户设备传输。例如,当具有理想的网络可用带宽、用户存储器和可保证的服务质量时,资料被传输给用户。

[0034] 本发明通过有效地传输媒体资料给用户还提供了媒体资料的连续、不中断、和持续地无缝的显示(视频和音频)。通过把这些资料集成于一个浏览装置(如网络浏览器),提供给用户无中断的、连续的媒体内容流,无需实时缓冲。

[0035] 本发明还提供了一种改进的方法,用来通过网络或本地资料存储库,智能地和可适配地,传输一个或更多的大型媒体资料,如,音频内容、视频内容、电影、游戏等。这样,本地媒体存储库对于客户进程或最终用户来说是可用的,其中相对高质量的服务将被保证。本发明还包括一种适配方法,基于资料的本地可用性,把这些资料集合组合到实质上无缝的再现。

[0036] 本发明包括一种通过网络传输资料的方法。该方法包括通过网络把资料单提供给一个用户设备。该方法进一步包括在用户设备上运行的客户进程,客户进程在下载和传输资料给用户设备中参考资料单。客户管理程序进一步依据何时满足了至少一个预定的条件而管理资料的下载。

[0037] 内容提供商可以用推定的方式把数字资料放在用户设备中,这样,当用户需要它的时候,它立即可用,无需网络下载。当有用户或进程明示请求时可以发生这种情形,或者可以由内容提供商根据订购的服务而发起。

[0038] 下面结合附图对本发明做进一步说明。

附图说明

[0039] 图1是根据本发明一个实施例的客户进程,与内容提供商站点和远程媒体服务器互动的概况示意图;

[0040] 图2是上述客户进程的另一个示意图,表示用来把流和高速缓存的媒体集成于网络浏览器的脚本;

[0041] 图3示出了文件的程序代码的示例;

[0042] 图4是客户进程应用中的C++类的示意图;

[0043] 图5是客户用于进程下载媒体资料的时间特性的示意图;

[0044] 图6是影响等待媒体资料被下载的时间长度的误差表;

[0045] 图7描绘了有关资料单中的资料信息的属性的分级化性质的一个方面;以及

[0046] 图8描绘了有关资料单中的资料信息的属性的分级化性质的另一个方面。

具体实施方式

[0047] A. 总体说明

[0048] 通过有效地传输媒体资料,媒体内容的连续的、无中断的和持续无缝的显示(视频和音频)被提供给用户。具体地说,当具体的条件被满足时(如,网络带宽可用性,用户设备存储器和保证的QoS等),资料被周期性地传输给一人用户设备。通过把这些资料集成于一个浏览装置(如,网络浏览器),不需要实时缓冲的、无中断的、连续的媒体内容流被提

供给用户。

[0049] 一个或多个大型媒体资料,如音频内容、视频内容、电影、游戏等,被智能地、适配地通过网络传输给一个本地资料存储库。这样,本地资料存储库对于客户进程或最终用户来说是可用的,其中将可以保证相对高质量的服务。本发明还包括一种适配方法,基于资料的本地可用性,把这些资料集合进实质上无缝的再现。

[0050] 可以保证服务质量的本地资料存储库的一个例子,可以是一个由用户设备连接的本地网 (LAN) 提供的一个高速缓存或数据存储设备。本地可用性的含义是,资料在本地对于最终用户或进程存储系统 (或一个本地存储库,在客户与本地存储库之间有高速网络) 是可用的。

[0051] 在一个实施例中,一个进程或应用运行于用户设备中,在此称之为客户进程。客户进程负责管理用户本地资料库中的一个内容高速缓存。应用、网页和多媒体讲稿可以靠所述的高速缓存来组合已经驻留的大型媒体资料。应用、网页和多媒体讲稿可以询问高速缓存的内容并动态调整与可用的内容有关的用户体验。这些应用可以是客户端或服务端面的应用。

[0052] 客户进程可以从互联网下载并装入用户设备的服务。作为替换,客户进程可以作为计算机操作系统的核心部件来提供,也可以与其它软件应用打捆来提供。

[0053] 客户进程为最终用户管理媒体资料的高速缓存。在一个实施例中,包含在高速缓存中的资料专用于一个资料单,该资料单由一个内容提供商提供。在此实施例中,客户进程从一个网络地点按特定和可调的时间间隔重新获得资料单。基于资料单中的信息,当满足预定的条件时,客户进程管理从远程地点的资料下载。

[0054] 一个客户进程可以与不止一个资料单组合,所以许多服务提供商或一个服务提供商的多个业务部门可以独立地维护他们自己的资料单,并使用同一客户进程来管理下载。此外,当客户进程被安装或升级时,客户进程可以被告知资料单的位置。安装可以从一个网站发起。如果客户进程还没被安装在客户设备中,网页中的软件检测到这种状况并提示用户下载并安装客户进程。也有可能通过网页中的脚本 (与 ActiveX 控件相结合) 把资料单添加给客户进程。

[0055] 客户进程根据预定的条件对资料的下载进行管理。如上所述,预定的条件例如可以包括网络带宽可用性、用户设备存储器、时间、和保证的服务质量。例如,在客户进程的一个实现方式中,只在用户设备闲置时才进行下载。在另一个实现方式中,只有当网络活动低于某一个水平时客户进程才进行下载。又例如,客户进程测量用户设备中的 CPU 与存储器的使用状况,当该状况低于某个运行水平时,满足预定的条件。在再一个例子中,客户进程依据一天中的某个时间,依据何时网络使用状况最低对资料的下载进行管理。当然,客户进程此外还使用条件的任何其它组合来管理资料的下载。

[0056] 下载能力可以由用户打开和关闭。用户还可以指定只能当设备处于某一个网络中时 (例如,与拨号连接相对的 LAN) 才能够下载。例如,用户可以向内容供应商订购这样的服务,在深夜时分 (由客户进程) 下载资料,供早晨时分观看。

[0057] 进一步,智能地和可适配地管理资料的下载也是可能的,无需一个外在“下载管理器”处于用户的管理之下。所谓的智能,意思是说,当一个预先设定的条件被满足时 (如网络带宽、一天中的某个时间、QOS 等) 资料才被传输。所谓的可适配,意思是说,资料单,处

于内容供应商端或用户设备端,可以基于用户浏览的倾向来自动地进行更新。

[0058] 资料单被用作内容制作者在客户的本地商店发布和管理内容的一种手段。所以,该功能允许内容制作者发布那些将在后来的某时被集成于应用的再现、网页、或基于 FLASH 的讲稿之中的媒体资料。在一种情形中,制作者在一个后台发布系统中录入有关资料的信息,包括它的地点、再现的时间、文字、和与资料关联的其它信息。这些信息然后被发布和合并到有关的资料单。所述的发布系统还可以发布(立即或在未来的某个日期或时间)合并了这些资料的网页或讲稿。

[0059] 一种通过网络传输资料的方法,包括:(i) 由内容供应商通过网络把资料单提供给一个用户进程,其中所述的进程运行于一个用户设备中;(ii) 当预先设定的条件被满足时,对应于资料单,把资料通过网络传输给用户设备。作为一个例子,资料可以是二进制数据、音频内容、视频内容、文本内容、或一个多媒体文件中的至少一个。预先设定的条件可以包括,一天中的某个时间、用户设备状态(即,设备是否在使用或闲置)、带宽使用状况(即,带宽的使用是否低于一个预先设定的运行水平)、用户设备 CPU 使用状况、用户设备存储器的使用状况(即存储器使用是否低于预先设定的运行水平)。此外,客户进程可以存储在用户设备的本地高速缓存中。不仅如此,客户进程可以把存储的资料与来自于内容提供商的实时内容结合起来,以无中断的、无缝再现的方式将内容提供给用户。此外,资料单可以经常更新,从而让客户进程(或高速缓存)有能力周期性地、按预先设定的条件接收丰富的内容。

[0060] 为用户提供无中断的内容流的系统包括:(i) 一个资料单,由内容供应商准备就绪,通过互联网供客户进程使用,其中的客户进程运行于用户的设备上;(ii) 一个资料,当预先设定的条件被满足时从一个远程地点准备就绪,通过互联网供用户设备使用,其中远程地点的信息从所述的资料单中获得;和(iii) 一个用于把传输的资料集成于由用户设备正从远程地点通过网络接收的内容流的集成器工具。无中断的和连续的内容流被提供给用户。

[0061] 制作者可以录入可供选择的低带宽媒体资料和相应的文字信息,它们可以在后台与丰富的媒体资料信息结合起来,从而产生两种不同版本的页面(一种用于不管出于何种原因未能下载大型资料的用户,另一种用于丰富的媒体资料)。用户可能因为多种原因无法下载丰富的媒体资料,比如,(1) 不兼容的系统;(2) 系统不在线,无法接收下载;(3) 未为收费服务付费;(4) 盘空间不够;(5) 计算机一直在使用,用户不希望其资源被用于下载。所以以上述两种方式提供可选的用户体验是重要的能力。

[0062] 例如,广告中的内容可以以序列的方式传输,只允许内容在相应的广告也存在于本地高速缓存的情况下才被显示。在一个实施例中,这一功能是通过使用(客户端或服务端)脚本来确定是否两种媒体资料都存在于用户设备中(内容部分与广告)来实现的。在另一个实施例中,这一功能是通过使用资料单的集束来实现的,其中的资料单集束当两种资料都在本地设备中时设置了令牌。

[0063] 用户可以主动地指定媒体被下载和高速缓存在本地设备中。在一种实施方式中,这是通过一个为每个用户管理资料的服务器端数据库实现的。例如,在一周的中间,用户可以指定他想要在周末看一部一般电影长度的 DVD 质量的电影。这将导致资料被加入到资料单中,并导致在接下来的时间里客户进程下载资料。用户可能因为资料加入到单子而被收费,

或者（见下文）是数据权利管理可能被使用来控制资料。在另一个例子中，这种方法与来自内容提供商的一种“推动”结合起来，即，为了在每天早上看深夜时的新闻广播，用户通过一项服务来订购。其结果是，这项服务会每晚在用户的资料单中插入一条新的资料以确定新闻广播被下载。

[0064] 进一步，系统可以利用 ISP 和有线服务提供商的协同定位传输设备来工作。在一种实现方案中，当制作者发布一项资料时，它随即被转移到一个 ISP 和有线服务提供商的协同定位传输设备或系统（它在网络布局中接近用户的位置）。客户进程获得该资料并把它放在本地的高速缓存中，方法是从 ISP 和有线服务提供商的协同定位传输设备或系统中获得，而不是从服务供应商的源媒体服务器中获得。

[0065] 此外，内容提供商能够以基于它手头的资料来优化用户体验。例如，如果没有丰富的媒体资料可供使用，有关新闻的网页可以提供一个新闻事件的图片，然而，如果有一项视频资料在本地存储，网页可以在网页中图象的位置用视频来替代。这可能包括视频的相应的文字“拷贝”，因为视频和图象可以不是针对同一事件的，因而会需要不同的支持文字和字幕。在一个实施例中，这一功能是这样实现的，让服务器端应用通过相应的资料令牌来检测资料的存在（参见上文的资料单属性）。依据令牌是否存在，为用户生成不同的页。在另一个实施例中，它是在客户端通过网络浏览器中脚本来实现的。就是说，使用网页中的脚本，基于媒体资料令牌的存在利用动态 HTML 来生成不同的页。

[0066] 本发明的一个优点是它允许内容供应商优化其网络带宽的使用。具体而言，当使用状况低于峰值时增加内容的传输，带宽使用的低谷被填充，从而优化了带宽的使用。

[0067] 另一方面，数字权利管理 (DRM) 方案可以用于数据流。如，被 DRM 保护的资料可以利用本发明的方案来发布。所以，当用户“播放”资料时，提示他获得合适的许可，作为资料的“使用权”。

[0068] 另一方面，GAME 资料可以以可能用于客户进程的真实或虚拟 ISO 图象的方式来发布。

[0069] B. 机顶盒的例子

[0070] 客户进程可以下载和管理来自不同来源的有关媒体内容的媒体资料，所述的来源例如可以是电视机顶盒，其中资料是通过一个载波信号的边频带或通过数据网络，如有线网或卫星网或互联网或内部网传输的。例如，在一个实施例中，在家中的浏览者可以访问一个电影库，或供随时浏览的任何其它的音频 / 视频内容。所述方法涉及传输媒体资料，如电影，给位于家中的机顶盒并允许电影聚合起来。

[0071] 机顶盒中的一个硬盘用来存储电影。电影的传输利用通过标准广播电视信号传输大量数据的数据广播技术。有关数据广播技术的信息可以从 Hartson 等人名下的专利申请 W09955087 号文件中找到，该申请的申请日为 1999 年 4 月 16 日，公开日为 1999 年 10 月 28 日，在此作为本专利文件的参考。例如，在 24 小时时间内，该广播技术可以传播 20 部一般片长的高质量电影。机顶盒很容易连接到电视机上，就象任何其它外设一样，如 VCR 或 DVD 播放机。由于计费的缘故，机顶盒还连接于电话线上。

[0072] 在一个作为示例的实施例中，机顶盒位于一个人的家中，用标准视频电缆连接于他们的电视机。机顶盒有一个天线，通过广播电视信号接收数据。机顶盒最好有一个至少 80G 字节的硬盘驱动器用来存储多部电影。机顶盒有一个调制解调器，机顶盒用它来周期性

地与服务供应商联系。在服务供应商与机顶盒之间传输的信息包括：用户“浏览 / 租赁历史”，用来计费之用，机顶盒日志，用来监测系统的运行和用来解密电影的“电影密钥”。

[0073] 机顶盒有一个处理器，能够接收来自广播信号的数据流，能够重建数据，和把数据写到硬盘上。处理器必须还能够在播放一部电影的同时，对来自遥控器的红外信号作出反应，并对调制解调器作出反应。出于安全的考虑，硬盘与机顶盒一对一关联搭配，使得它一旦被卸除或用于其它地方都不可用。机顶盒最好没有风扇，因为它需要被设计成很安静的设备。机顶盒还包含一个安全处理器，作为其安全系统的一部分。安全处理器是一个智能卡的一个主动构件，该智能卡用环氧物质物理地连接于 PCB，以使其不易被改变。

[0074] 电影并不是实时地“流传输”到机顶盒，相反，内容文件被打包，这些包被连续地传输到机顶盒，在那里，它们被累加重组起来。用户并不清楚什么数据被传输给他们的机顶盒。供应商将电影下传，并被动地驻留在机顶盒中一个有限的时间。数据传输被内容或服务供应商所控制。为保证电影被完整地接收，同一部电影可能被广播传递给机顶盒几次。任何在第一次传输尝试中没收到的数据包会在以后的广播传输中被收到。在本发明中，一个客户进程驻留于机顶盒中，当预先设定的条件被满足时（如网络带宽，QOS），媒体资料（电影内容）被从远程地点获得。

[0075] 传输给机顶盒的电影还可以包含有关信息，定义电影的一定的特性。如，一个电影可以有一个关联性的开始和结束的日期或时间，用来限定可以看电影的时间范围。电影可以到达并存储于机顶盒中，但它可以有有一个关联的时间，在这个时间之间不可以看电影。这一功能容许电影在从一个到另一地的传输时间上的差异，允许电影被“预装”，使得在正式发行的日期立即就绪。类似地，资料单可以有有一个结束的日期与电影关联，过了这个日期，电影就不可以看了，并自动从机顶盒中删除。

[0076] 存储于机顶盒中的电影是被加密的。当一部电影要被选择观看而且满足商业条件（如，用户还有足够的额度），机顶盒将允许电影被解密和播放。在允许电影被观看之前，机顶盒无需连接服务供应商，因为解密电影的密钥通常是与当前帐户状况一起被先前取得，并驻留在机顶盒中。显然，加密的电影未经解密是无法观看的。所有的解密都被记录，该记录用于确定用户的帐单。与解密和用户帐户状况有关的逻辑处理都由安全处理器来进行。在本发明的一个优选实施例中，在观看电影之前，随着电影被选择用于观看，和接着的“租赁确认”对话，费用就被收取。

[0077] 在另一个实施例中，当电影的主要部分已经被观看，用户就会被收费。一旦电影已经被选取，它可在一段有限的时间内（如，24 小时）重新观看而不收费，或者有限的重看次数而不收费。在另一个实施例中，不是为每部电影收费，而是收取月费。月费的多少取决于各种选择，如可看的电影的数量，一部被选的电影可以看的时间长度，每部电影可看的次数。可以用一根电话线来实现源自用户的信息的传输，所述的信息可以是用户的需要一把密钥，和由此应当收取的费用。因此，即使用户有一个电影库存储于用户的机顶盒中，如果不实际观看电影，则不收费。

[0078] 在另一个实施例中，数据可以通过电缆、卫星、互联网等传输给机顶盒。虽然无线广播是优选实施例，本发明不应局限于无线传输。

[0079] C. 结合附图的说明

[0080] 图 1 是表示用来传输和在用户设备中无中断地再现媒体流的系统 10 的总体示意

图。具体地说,用户设备 20 包括至少一个客户进程 24(如,客户资料管理器进程),该用户进程与客户资料单 42 互动,客户资料 42 处于用 40 表示的内容供应商的地点。在用户设备 20 中还有用来存储信息的一个本地高速缓存 26。高速缓存 26 可以是至少一个随机读取存储器 (RAM),或只读存储器 (ROM),或硬盘。此外,还可能有一个远程地点 60,它包含资料 62,例如程序内容(如一个体育事件、厨艺节目等)。作为另一种方案,资料 62 可以位于内容供应商地点 40,或者位于有线/互联网服务供应商地点(未示出)。在客户开始提取之前,资料已经在有线/ISP 供应商的地点就绪。进一步,客户可以同时从多个物理上不同的地点(如,从有线/ISP 供应商,内容供应商等)提取资料。

[0081] 客户进程 24 为最终用户管理媒体资料高速缓存 26。在一种实现方式中,包含在高速缓存 26 中的资料 62 关联于资料单 42,资料单 42 由内容供应商 40 通过数据通道 80 来提供。在此实现方式中,客户资料管理器进程 24 按指定的(可调整的)时间间隔从网络地点(如内容供应商地点 40)通过数据通道 80 提取资料单 42。具体地讲,客户进程 24 通过控制信号通道 84 提出一个请求给内容供应商 40,要求通过通道 80 传输更新的资料单 42。

[0082] 类似地,在另一种实现方式中,资料单 42 可以被内容供应商的服务器(未示出)以周期性的方式发送给最终用户客户进程 24。资料单 42 是一个数据文件,它至少包含内容资料的一个清单。资料单 42 中的每个单元通常确定获取该资料所需的一个网络地点和协议。例如,在一种实现方式中,该网络地点是一个统一资源定位器 (URL:Universal Resource Locator),而所述的协议是 HTTP。资料单 42 还可以包含与每项资料关联的其它属性(如在发明内容部分所解释的和在术语定义部分所定义的),如:

[0083] (i) 失效日期,(ii) 删除;

[0084] (ii) 回叫 URL- 在一个实施例中,该回叫 URL 用来发起有关资料下载的操作,如跟踪应用(允许跟踪哪些资料被下载,频繁程度如何),和通过电子邮件或即时信息通知客户一项资料已经就绪可以浏览的应用;

[0085] (iii) 客户端令牌- 在一个实施例中,用来基于资料的可用性来调整再现;

[0086] (iv) 流量- 在一个实施例中,该数字是可用的客户流量的一个百分比,小于 100%,客户进程可以尝试仅按这样的流量来取得数据,而不是一下全取得,从而减慢下载。在另一个实施例中,该数字是用字节每秒表示的下载数据率。

[0087] 禁止日期- 任何内容都可以在一个有限的时间窗口内才可用。一旦禁止日期到了,内容会被客户进程移除,不再可用了。

[0088] 一方面,这些资料可以被合并成一束一束的内容。上述的属性可以被指定给一束。例如,这使得一旦一组媒体资料已经被传输给用户设备,一个客户端令牌就被生成。

[0089] 某些些属性可以与整个资料单 42 关联起来,这些属性是:

[0090] (a) 更新率:在一种实现方式中,可以在服务器端对此进行控制(如,在内容供应商地点 40),服务器端把资料单 42 通过数据通道 80 下传给客户管理器 24。

[0091] (b) 资源路径;和 (c) 媒体资料

[0092] 图 7 描绘了有关资料单中的资料信息的属性的分级化性质的一个方面。如图所示,内容供应商/ISP 提供给客户的资料单具有所有资料信息的属性。举例来说,属性 (1,2) 可以指向资料 1 的过期日。该图所描述的属性与每项资料处于一地。

[0093] 作为另一方面,属性对于资料单来说可以是全球性的,如图 8 所示。在这种情形

下,客户进程基于适用于所有资料的全球性属性来管理资料单中的资料的下载。

[0094] 客户进程 24 还能够从远程地点 60 管理它的资料下载行为的许多参数。例如,在一种实现方式中,客户资料管理器 24 通过信号 86 向远程地点 60 发送一个请求,请求通过数据通道 90 传输媒体资料 62 给本地高速缓存 26。客户进程 24 可以基于一组预先设定的条件是否被满足来把这些数据传输(下载)的请求发送给远程地点 60。例如,下载会下列情形下发生,当用户设备 20 闲置时,当网络带宽低某个水平,当用户设备 20 中的 CPU 与/或存储器的使用状况低于某个运行水平时。

[0095] 下载能力也可以由用户打开或关闭。用户可以指定只有当设备处于某个网络中时(如相对于拨号上网的 LAN)下载才可以进行。

[0096] 在没有由客户进程 24 发起请求的情况下,依据网络状况(如,流量大小、可用带宽、时间,等),远程地点 60 可以减缓或推动媒体资料 62。

[0097] 资料单 42 用作一种手段,供内容制作者发布和管理客户的本地存储/高速缓存中的内容。例如,该功能允许内容制作者发布将在某些点被无缝地集成入网页再现或 FLASH 再现(未示出)中的媒体内容。

[0098] 相应地,制作者可以把有关资料的信息录入一个后台发布系统,包括它的地点,再现的日期和文字,以及将会被与资料关联的有关信息。这些信息然后会被发布并包含到相应的资料单中。发布系统还可以(立即或在未来的某个日期或时间)发布包含了该资料的网页或再现。制作者还可以录入低带宽媒体资料及相应的文字信息,这些内容之后会与丰富媒体资料合并生成两种不同版本的页(一种用于不管出于何种原因未能下载大型资料的用户,另一种用于丰富的媒体资料)。

[0099] 客户资料管理器进程 24 是可运行于用户设备 20 上以提供下载和高速缓存的应用。用户将能够自主决定获得客户管理器/进程,很可能是通过内容供应商来获得。作为一个示例的实施例,客户管理器可以被安装成自动启动,且在后台运行。

[0100] 图 2 显示了系统 100 中信息流动的示意图,其中客户资料管理器进程 102 运行在用户设备中。在一个示例性的描述中,客户资料单 110(如,一个 XML 文件)被定期从内容供应商 120 下载,列于那个文件中的资料 130 被从远程地点/内容供应商 140 处下载并被要相应地高速缓存起来。用户通过把脚本 150 包括进来可以导航到显示资料的网页,脚本 150 与客户管理器进程 160 通信。由于客户管理器进程 102 也可以是 ActiveX 服务器,仅一个代理对象可以在网页 180 中运行,实际的控制可以是客户管理器进程 102 的一部分。在另一方面,客户进程 160 可以嵌入在网页 180 中或客户管理器进程 102 的一部分中。

[0101] C(i). 设计方面的说明

[0102] 输入可以是:(i) 配置文件,如 XML 文件,允许远程控制客户管理器进程 102 的行为;(ii) 将被高速缓存于客户端的远程媒体文件。输出可以指向媒体文件的本地路径。

[0103] 任何网页都可以显示被客户进程 102 高速缓存的资料。使用客户进程的 ActiveX 界面,一页可以包含 JavaScript 码 150,来看是否安装了客户进程 102,并询问客户关于媒体文件的本地路径的信息。如果一个路径被返回,脚本然后将该路径信息传达给接受参数的任何基于网络的媒体播放器。

[0104] 设置服务器 120 可以是寄存设置文件 110 的任何网络服务器。媒体服务器 140 可以是寄存媒体文件或资料 130 的任何网络服务器。

[0105] C(ii). 实施流程

[0106] 客户进程 102 可以被打包于一个 .cab 文件并用一个数字证书签证以标记程序的创作者（如，沃尔特迪斯尼互联网集团公司）。可以为不同的内容供应商生成独立的 .cab 文件 xyzvideo.cab。例如，用于 ESPN 的文件名可以是 espnvideo.cab，它可以用 ESPN 数据证书签证。Internet Explorer 使用互联网构件功能下载和跟踪应用。一个 .INF 文件可以被含在 .cab 中来指导构件下载器如果安装客户进程 102。

[0107] 所述的 .cab 文件被放在一个网络服务器上，当浏览器载入一个具有 OBJECT 标签的网页，所述标签包含适合的 ActiveX 控制的类别身份码 (ID)，浏览器会使用由 CODEBASE 属性指定的 URL 来下载 .cab 文件。数字证书被显示，以此通知用户我们的控制已经被下载，并提示用户作出执行的许可。一旦用户接受，该窗口将不被再次显示，除非一个更新的版本被下载。

[0108] 模块 Clientprocess.exe 被加入到名为 Downloaded Program Files(已下载程序文件)的子目录中，客户可以自己在启动(startup)子目录中建立快捷键。快捷键的名字取决于谁建立了这些文件（如，迪斯尼，ESPN 等）。客户进程 102 也可以在 go.com 域中建立一个 cookie，称之为 ClientProcess(客户进程)，在客户进程被安装后被设置为“真”。该 cookie 可以被定期地检查，以防止它被用户无意中删除。

[0109] 该程序首先被操作系统执行，从那时起，无论何时用户启动设备，它都会运行。

[0110] 前述的数字证书可能是必要的，因为大多数用户都有他们的浏览器安全设置，而且安全程序都足够高，不允许安装来自互联网的不可信的应用程序。证书的存在允许浏览器确定软件的来源，并询问用户那个公司或个人是否可信。

[0111] OBJECT 标签的 CODEBASE 属性支持可选性地添加安装希望的软件版本。版本信息附加在 cab 所在地点的 URL 上，用“#”作分隔符。例如，如果客户进程目前的版本是字段 1,0,0,1，新版本 1,0,0,2 已经被放在网络服务器上，把“#1,0,0,2”附加在 CODEBASE URL 的尾部，这将指示构件下载器下载和安装新版本。

[0112] .INF 文件包含一个设置，它使得先用 /shutdown(关机)参数来运行 clientprocess.exe 以便先关闭 clientprocess.exe，使得更新版本不要求重启设备。

[0113] 构件安装器可以用事先定义的键添加信息给注册表。

[0114] 对应于客户进程系统托盘图标菜单中将含有一个卸载选项。点击该选项后，客户进程运行 clientprocess.exe 另一个程序的，传出 /unregserver(解注册服务器)参数，从而卸载它自己。卸载进程将做以下事情：(i) 解注册控件，(ii) 清理由客户进程添加的注册表项目，(iii) 从启动文件夹中移除客户进程项目。

[0115] 当操作系统启动，且 Explorer 运行启动文件夹中的项目时，客户进程 102 将启动。除非用户把它从系统托盘图标中丢了或者是系统关机，否则它不会终止。

[0116] 客户进程 102 对于用户来说是不可见的，除了含有几个菜单项目的系统托盘图标，例如，如果 ESPN 的 BottomLine(底线)应用正在运行，客户进程的 ESPN 版本将不会显示它的图标，因为 BottomLine 将会在它的菜单中为客户进程 102 提供一些选项。这是有用的，因为它减少了可能会放在一个客户进程中的托盘图标的数目。程序 BottomLine 与客户进程 102 可能需要定期彼此通信，保证一方在另一方关闭时可以知道，且在它的菜单项目中的一个被选定时客户进程可以知道。图标可以是 ESPN 图标或取决于创建方式的另一个

图标。

[0117] 下列项目将存在于客户进程托盘图标菜单中：(i) 帮助 - 显示客户进程 102 的帮助，(ii) XML 群组文件可以具有一个可选的 helpURL 项目。当用户选择帮助时，用这个 URL 打开一个浏览器，(iii) About(关于) - 显示程序的版权 / 信息对话框，(iv) Exit(退出) - 停止程序（不将其从启动文件夹中移除），(v) 卸载 - 卸载程序并清除 COM 注册表项目。

[0118] 每次至少客户进程 102 的一个程序可以运行。唯一的例外中，如果 clientproces.exe 在 /unregserver(解注册服务器) 或 /shutdown(关机) 参数运行，所述程序会终止一个已经在运行的程序，如果存在的话。Clientprocess.exe 建立一种互斥关系，后来的程序检查该互斥关系，从而实现单一程序功能。所以，在某些情形下，如果发现一个正在运行的程序，第二个会立即终止。

[0119] 客户进程的线索记录模型可以是公寓模型。就是说，客户进程 ActiveX 控件是这样的设计的，它们希望仅从一个线索记录被访问。然而，客户进程作为一个整体是多线索的，它将启动一个线索来进行单个资料下载，终止那个线索，当下一个下载发生时启动另一个线索。一次只能进行一个下载。

[0120] 在通常的运行模式中，客户进程将不用任何参数而运行。然而，下列命令行参数可能会被支持：(i) /regserver(注册服务器)；(ii) /shutdown(关机) - 停止在运行 clientprocess.ext 的另一个程序，如果它存在的话；和 (iii) /unregserver(解注册服务器)。

[0121] 使用从设置服务器上下载的 XML 群组文件设置客户进程。通过一个网页内为 ClientProcessGroup(客户进程群组) ActiveX 控件而设的 OBJECT(对象) 标签中的 PARAM 标签来获得群组文件的 URL。每当一个 ClientProcessGroup(客户进程群组) 被一个 XML 参数发起时，客户进程进行检查，看看该群组是否已经被加入。如果没有，它会被加到注册表，文件会被下载并拆解。群组文件的注册表项目不会移除，起到该客户进程被卸载。

[0122] 通常，根据可配置的更新率来下载群组文件。如果服务器支持 If-Modified-Since(如果 - 从某时 - 修改) 文件头，客户进程可能不下载该文件，如果它自上次下载以来还没被修改。

[0123] 群组文件中的 XML 使用 RSS 格式，并具有一个客户进程特殊的元素。RSS 是一个公开格式，用于许多迪斯尼作品的合成。这样做的初衷是试图使尽可能充分地使用标准，使得群组文件有可能用于其它系统。

[0124] 图 3 示出了一个用于群级文件的示例 DTD。

[0125] 启动之后，客户进程把自己注册于系统中，作为一个用于后续的 ActiveX 控件的 COM 本地服务器(Local Server)。只有 ClientProcessGroup(客户进程群组) 被标志为安全，可以用于初始化，只有 ClientProcessLocator(客户进程定位器) 被标记为安全，可以用于脚本。这是为了避免控件被用于不良目的。ClientProcess(客户进程) 的初始模型设计成具有用于每个主要构件的一个 ActiveX 界面，允许网页显示状态。这些界面对于开发 / 测试阶段的故障检修也很有用。

[0126] ClientProcessSvc：

[0127] 这是应用的主要类型，放在安装网页上，它有一个 OBJECT 标签，用来强制应用的下载和安装。生成一个单独的 CClientProcessSvc 程序，用于对已经生成的资料和群组对

象进行跟踪的进程。IclientProcessSev 界面提供用于报告程序状态的方法。

[0128] ClientProcessGroup(客户进程群组) :

[0129] ClientProcessGroup(客户进程群组)ActiveX 控件被 JavaScript(Java 脚本) 用来指定一个设置给客户进程。通过 IproperBag 界面, ClientProcessGroup(客户进程群组) 接受 OBJECT 标签参数, 调用被设定给 XML 文件的 URL 的 XML 或资源。单个的 XML 设备文件 (或群组文件) 用一个 ClientProcessGroup(客户进程群组) 对象来表达。IclientProcessGroup 界面提供状态功能。

[0130] ClientProcessLocator(客户进程定位器) :

[0131] ClientProcessLocator(客户进程定位器) 是一个简单的控件, 它对脚本安全, 且只有一个方法, 即 GetLocalPath。如果已被下载, 它被 JavaScript(Java 脚本) 用于显示媒体资料的网页, 以确定本地文件的路径。

[0132] 此外, 如果用户已经停止客户进程, 然后浏览到下一个媒体网页, 含有 ClientProcessLocator(客户进程定位器) 类别号 (id) 的 OBJECT 标签将会引发 clientprocess.exe 再运行。如果用户卸载该产品, 在 CODEBASE 参数被确定的情况下, 该标签将强制重新安装或升级。

[0133] ClientProcessAsset(客户进程资料) :

[0134] ClientProcessAsset(客户进程资料) 代表将被本地高速缓存的任何对象 (群组或媒体文件)。该 COM 对象在 ClientProcessDownloader(客户进程下载器) 中的 ClientProcessSve 或 get_queueAsset 用 get_asset 方法返回。

[0135] ClientProcessDownloader(客户进程下载器) :

[0136] 单个的 ClientProcessDownloader(客户进程下载器) 在应用启动时被生成。它处理资料下载队列。当一项资料来到下载队列的顶端时, 资料用 QueueDownload(下载队列) 将它们自己添加到队列中, 它被交给 CasyncCacheDownloader 程序去处理。

[0137] ClientProcessReference(客户进程参照)

[0138] ClientProcessReference(客户进程参照) 对象代表媒体文件。每个群组可以有一个或多个参照。该 COM 对象在 ClientProcessGroup(客户进程群组) 被 get_reference 方法返回。

[0139] 其它 C++ 类

[0140] 该部分定义了客户进程应用中的 C++ 类的示例, 并总结它们的行为以及相互间的作用。这些类中的一些具有 COM 封面对象, 用于与其它应用通信。图 4 给出了各种类的表格。

[0141] CasyncCacheDownloader-asynccachedownloader.h/.cpp :

[0142] 该类别通过 HTTP 把资料的下载封装进本地设备中。CasyncCacheDownloader 再细分类 Cwindowimpl, 允许它接收信息, 因为它的大部分操作发生在一个单独的线索记录中。客户进程使用 window 信息用于线索内部的通信。

[0143] 其方法中的大部分运行于主线索中, 但 StartDownload(启动下载) 生成一个新的线索, 该线索又调用 DoDownload() 方法。DoDownload() 用 Winnet APIs(Winnet 应用程序接口) 来下载单个资料并为其生成一个 Internet Explorer 高速缓存记录项。

[0144] DoDownload 方法在一个时间用小块的方式下载文件。如果设置的流量小于

100%，它还会在每完成一块的下载后休眠一会儿。例如，如果流量为50%，下载线索休眠的时间将与它用来下载这块数据所需要的时间一样长，以实现50%的流量。

[0145] CCacheAsset-cacheasset.h/.cpp：

[0146] 一个 CCacheAsset 程序代表需要被高速缓存在本地系统中的单个文件。这可以是一个群组文件或一个媒体文件（参照）。CCacheGroup 和 CCacheReference 两个程序都例示一个 CCacheAsset 程序，以处理文件的下载和高速缓存。GetInternet 方法返回其各自的 COM 封面对象的 ClientProcessAsset（客户进程资料）界面指示器。

[0147] 如果下载不成功，CCacheAsset 将设定一个计时器，从而根据图 5 中的表格再尝试一次。如果下载成功，CCacheAsset 观察文件系统，以确认高速缓存项目没被从高速缓存中删除。如果删除了，再次下载该项目。

[0148] CCacheGroup-cachegroup.h/.cpp

[0149] 该类别封装 XML 群组文件。每个 Group XML 文件被传递给 CCacheGroup 的一个程序，程序解析此文件，并生成用于每个媒体文件项目的 CCacheReference 程序。GetInternet 方法返回其各自的 COM 封面对象的 ClientProcessGroup（客户进程群组）界面指示器。一个 CCacheAsset 程序被生成，以代表该组文件，一个计时器也被设定以一定的时间间隔，该间隔被设成 XML 中指定的 refreshRate（更新率）。当计时器到时，用于该组文件的 CCacheAsset 被加入到下载队列。

[0150] CCacheReference-cachereference.h/.cpp

[0151] 当 XML 群组文件被 CCacheGroup 解析，为每个媒体文件项目生成一个 CCacheReference 程序。CCacheReference 生成一个 CCacheAsset 程序来处理下载 / 高速缓存功能。该类别还负责检查项目过期时间，将过期的项目从盘中删除。

[0152] CCacheTime（高速缓存时间）-cachetime.h/.cpp：

[0153] CCacheTime 再细分类 ATL CTime，以提供时间转换功能。客户进程中所有的日期 / 时间变量都利用 CCacheTime 存储。

[0154] CClientProcessModule-clientprocess.cpp：

[0155] 该类别代表可执行进程自己。CClientProcessModule 对标准的 ATL CAtlExeModuleT 类再细分类。其原因是为了添加功能给进程的启动，以及注册和解注册。在启动时，以 ClientProcessSve GUID 为名字生成一个共用的互斥关系。如果该互斥关系已经存在，这意味着应用已经在运行，所以当前的程序存在。所有额外的安装和卸载功能也在该类中完成。

[0156] CException-exception.h/.pp

[0157] 发生错误时，CException 程序被生成并被丢弃。CException 处理错误的信息。

[0158] CRegistryVirtualDeviceX-registryvirtualdevicex.h：

[0159] 这是 ATL 中 CRegistryVirtualDevice 子类。该过度载入 AddStandardReplacements（增加标准替代）来阻止 ATL，不让它检查模块的扩展以确定我们是否应当把 InProcServer32 或 LocalServer32 放入注册表中。客户进程始终中一个 .exe，所以也是一个 LocalServer（本地服务器），因而，在应用上添加外部要求的调用可能是不必要的。

[0160] CTime 和 CTimeable-timer.h/.cpp：

[0161] CTime 包括时间功能。希望使用计时器的对象仅需要从 CTimeable 衍生, 和实现 OnTimer, 并生成一个 CTimer 程序, 调用它的 Start 方法。Stop() 停止计时器。

[0162] CUrlMap-urlmap.h/.cpp :

[0163] CUrlMap 是一个专门的 CatlMap 模板, 它允许用户生成把对象映射到 URL 的杂凑映射类。

[0164] 具体地说, 系统通过一系列方式得到增强 : (i) 为下载码增加下载率 (节流) 支持, (ii) 到期时删除资料, (iii) 实现 ClientProcessLocator (客户进程定位器) 控件并删除 COOKIE 逻辑, (iv) 使用系统托盘图标和菜单, (vi) 提供 HitCountURL 支持, (vii) 提供程序信息 (一个 “关于” 对话框)。

[0165] 本发明还给予一定的注意, 以保证下列事项 : (i) 所有适当的 ActiveX 安全设置都被应用到每个控件, (ii) COM 方法和特性被隐藏 (这就是说, 它们不会显示在控件设计用户界面, 如 Visual Basic 中), (iii) 静态功能被用来决定文件系统, 所以只使用一个计时器, (iv) 添加 BottomLine 逻辑以检查 ClientProcess (客户进程) 并提供系统图标子菜单, (v) 使用启动菜单快捷键, 而不是 windows 运行注册表项目, (vi) 定期地检查 COOKIE, 所述 COOKIE 指示是否 ClientProcess (客户进程) 已安装或在必要时重新生成。

[0166] C(iii).BottomLine 增强

[0167] BottomLine (ESPN 的商标) 可能需要被修改, 以检测客户进程, 如果进程存在, 在它的系统托盘菜单中显示客户进程菜单项目。这排除了如果用户有两个应用安装和运行的情况下在托盘中出现多个图标的情形。

[0168] 该客户进程将在 CLIENTPROCESS_MENU_START (客户进程菜单开始) 和 CLIENTPROCESS_MENU_END (客户进程菜单结尾) 范围内定义其菜单项目标识。当一个 BottomLine 可能访问的头文件被建立起来以后, 这些常数将被定义在该头文件中。

[0169] BottomLine 和客户进程还需知道是否所有的进程在运行, 所以它们需要执行下列步骤 : (a) 当 BottomLine 启动时, 它设法生成客户进程互斥关系, 来看该客户进程是否在运行, (b) 当客户进程启动或关闭时, 它广播一个注册 windows 信息, BottomLine 会看到这个信息, 并用一个回执作出回应, (c) 当 BottomLine 启动或关闭时, 它广播一个注册 windows 信息, BottomLine 会看到这个信息, 并用一个回执作出回应。

[0170] 注册的 windows 信息可以允许应用来定义它们自己唯一的信息, 用来与其它已经注册了同样信息的 windows 通信。当客户进程或 BottomLine 向其它应用或回执宣布它的存在, 它将沿着 window 句柄发送, 以利于未来的通信。然后, 客户进程将使用 WM_COPYDATA 把将显示在菜单上的信息发送给 BottomLine。当一个客户进程菜单项目被选定时, BottomLine 将转发它。

[0171] 一些功能也包括在实现客户进程的系统中, 它们是 : (i) 安装与升级的停止 (自动地), (ii) 客户端用户界面很少或没有, (iii) 窗户程序的服务器端设置, (iv) 跟踪有多少个媒体文件被下载的能力, (v) 通过 JavaScript (Java 脚本) 来确定一个媒体文件下载的状态的能力, (vi) 按其命名发送内容供应商的特殊控件 (如 ESPN 的特殊控件) 的版本的的能力。

[0172] C(iv). 系统功能

[0173] 客户进程程序 :

[0174] 客户进程程序负责在媒体内容显示于网页前的几小时下载该媒体内容。当用户浏览到一个含有显示那个文件的 HTML 的页面时,该文件的本地拷贝会被用到,这将显著地提高播放质量。

[0175] 安装:

[0176] 客户程序被允许自动安装。任何网页都可以含有一个指向客户进程的标签,必要时,浏览器将自动安装它或对它进行升级。

[0177] 用户界面:

[0178] 通过一个提供了关闭或卸载产品的能力的菜单,客户程序的用户界面可以被集成于一个系统托盘图标中。如果 BottomLine 应用存在,那么就使用它现有的系统托盘菜单。

[0179] 执行:

[0180] 客户进程可以被安装于用户设备的启动文件夹中,所以它一直在运行。程序会定期地从一个 WDIG 服务器下载一个 XML 文件,该 WDIG 服务器包含关于哪个资料要被下载、何进过期、以何种速率下载的信息。还有一个界面,供内容提供商网站与客户进程进行通信,以确定本地媒体文件的地点。

[0181] 寄存 (hosting):

[0182] .cab 安装文件可以被寄存于由使用该产品的 Vertical 提供的一个或多个首端文件。由于它是一个静态文件,它可被任何网络服务器寄存。随着新版本的客户进程的发布,该文件将在服务器中被升级。

[0183] 内容供应商 Bottom Line(底线)修改:

[0184] 可以对 BottomLine 应用作合适的修改,从而为客户进程提供系统托盘界面。

[0185] HTML 和脚本范例:

[0186] HTML 和 JavaScript(Java 脚本)范例可以被提供给开发商,以演示如何寄存客户进程和与之进行通信。

[0187] 外部界面要求

[0188] 这一部分描述外部系统,其中,界面需要为之生成和修改。

[0189] ESPN BottomLine:

[0190] 需要对 BottomLine 做小的改动,以检测客户进程的存在,增加菜单项目以停止和卸载该客户进程。

[0191] 客户进程检测:

[0192] 每当 BottomLine 被显示时,BottomLine 必须检测客户进程的存在,如果存在并且正在运行,显示一个包含客户进程菜单项目的子菜单。通过检查互斥关系的存在,客户进程的检测很容易做,其中的互斥关系是客户进程在启动或寻找具有客户进程 window 类的 window 时生成的。

[0193] 客户进程通信:

[0194] 如果客户进程在运行,BottomLine 可能会使用 window 消息以询问哪个菜单项目被显示。当一个客户进程项目被选定时,BottomLine 将会使用该信息来告诉客户进程哪个项目被选定了。

[0195] 媒体内容服务器:

[0196] 寄存媒体文件的服务器可以是任何 HTTP 1.1 网络服务器,只要它支持 HTTP 字节

范围的下载 (206success 码), 以实现下载节流。如果服务器不支持字节范围, 整个文件可能会被一次下载, 不管设置如何。

[0197] 配置服务器界面:

[0198] 可以有一个寄存 XML 配置文件的服务器。XML 可以跟从一个按照设计参数发布的格式。

[0199] 功能要求:

[0200] 1. 在用户设备上设置一个 COOKIE, 表示程序被安装了。该 COOKIE 可以定期被检查, 以确保用户不删除它。

[0201] 2. 通过 Internet Explorer 构件下载功能来更新支持软件, 这将保证当一个新版本被放在首端和当 HTML 被修改以要求一个更新的版本时, 客户代码被适当地更新。而且, 还可以保证在更新时不需要重新启动系统。

[0202] 3. 程序可以被安装在启动文件夹中, 所以将一直运行, 除非用户专门停止它。

[0203] 4. 在一个时间只有一个程序运行。

[0204] 5. 通过一个系统托盘图标菜单提供卸载功能。卸载可以从程序外部发起。

[0205] 6. 由系统托盘图标提供的菜单项目。

[0206] 7. 帮助 - 显示迪斯尼高速缓存的帮助。

[0207] 8. 配置文件可以有一个可选的 helpURL (帮助 URL) 项目。当用户选择了帮助时, 用该 URL 打开一个浏览器。

[0208] 关于 - 显示程序版权 / 信息对话框。

[0209] 退出 - 停止程序 (不将它从启动文件夹中删除)。

[0210] 卸载 - 卸载程序并将它从注册表项目中删除

[0211] 配置文件: (i) 可以是 XML 格式, (ii) 可以提供更新率, 用来确定以何种频度下载配置文件, (iii) 可以提供 URL 的按项目设置、过期时间、点击跟踪的 URL、和下载率。进一步, 一个项目的优先性可以按设置文件的次序来确定。而且, 两个不同的设置文件不可以包含同样 URL 的同样的项目。此外, 有关 XML 设置文件的地点的信息可以被 HKEY_CURRENT_USER 之下的注册表来定位。

[0212] 文件下载

[0213] 浏览器 (如 Internet Explorer) 高速缓存:

[0214] 用户进程可以用 Internet Explorer 高速缓存来提供媒体文件的客户端高速缓存。存于高速缓存中的文件可以被 Internet Explorer 清除, 如果必要的话, 以便为其它项目留出空间。XML 设置文件, 可以被标记成粘性, 以让它在硬盘上存在一定长的时间 (如 30 天)。由于 Internet Explorer 高速缓存是按每个 Windows 联机注册来工作的, 每个用户需要有单独的高速缓存。

[0215] 客户进程需要定期 (或使用文件系统观察工具) 检查以检测何时一个项目被从高速缓存中删除。如果它还没到期, 它将会再次下载。

[0216] 项目有效期:

[0217] 每个项目都有一个到期日和时间。定期地, 客户进程将检查这些时间, 从高速缓存中删除已经过期的项目。客户进程需要确保跟踪资料, 如果它们在客户进程有机会作废它们之前就把它们从 XML 配置文件中删除, 它们将被从用户硬盘中删除 (如果用户让他的设

备关机时间过长,这有可能发生)。

[0218] 一旦一个项目被下载,它将不会再被下载,除非 Internet Explorer 已经将它从高速缓存中删除。

[0219] 配置的更新:

[0220] 客户进程通过把 XML 配置文件从服务器按照由 refreshRate(更新率)XML 元素所定义的速率下载,定期地更新 XML 设置文件。

[0221] 下载错误:

[0222] 如果在下载一个项目时发生错误,在一段暂停后,客户进程将再试。等待的时间取决于依据图 6 所提供的图表的错误类型。

[0223] 进一步,如果网络连接就绪,客户进程可以进行下载,如果需要,它可以不通过拨号上网来进行下载。

[0224] 节流:

[0225] 配置文件中的每个项目都可以被可选择地配置成具有一个具体的下载流量比例。这意味着,客户进程可以分块地下载文件,而且可以在块与块之间停顿,以实现流量比例。寄存该文件的网络服务器应该能够支持 HTTP 字节范围特性。

[0226] 网络界面

[0227] 用于配置文件的 URL 可以通过一个 ActiveX 控件特性被传递给客户进程。ActiveX 控件对脚本是安全的,对初始化也是安全的。客户进程可以检查 XML 设置 URL 的服务器部分,以证明它是来自于 go. com 域。这可以防止恶意网页用可能有害的数据(以低更新率下载的大型文件)对客户进程进行初始化。OBJECT 标签不需要有一个 CODEBASE 参数,以致客户进程不会被意外地安装给某个无权限的人。JavaScript(Java 脚本)可以有代码来检查是否对象在进行任何 API 调用之前就存在,而且检查已安装的 COOKIE。用户进程的至少一个版本可以被安装在客户设备中。

[0228] D. 总结

[0229] 以上关于本发明的例子和希望的实施例的说明是为了解释和说明之目的。其意图不在于穷举和将本发明限定于具体描述的方式。

[0230] 根据本发明的精神还可以做多种修改和变化,例如:

[0231] (a) 有可能创建一个新的构件,它提供通用的系统托盘图标,其中已安装的应用可以自我注册,以添加菜单项目。

[0232] (b) 服务器端用户设置确定客户进程的行为。客户进程需要一个唯一的标志,固定关联于安装了它的用户。服务器可以更复杂,为不同的用户发送 XML 设备文件。

[0233] (c) 未完成的下载继续进行。

[0234] (d) 添加更复杂的下载节流(例如,对客户的基准带宽进行测量,把它用于计算流量控制)。

[0235] (e) 服务器端的智能设置,用于基于 MRTG 数据来确定服务器负载,并用它来决定返回 503 回应给客户进程以防止资料下载的过重负荷。

[0236] 很显然,本领域技术人员可以在不背离本发明的概念的情况下对本发明进行多种使用和修改,而且不同于这里所做的具体实施例的说明。所以,本发明应被理解为包括了所述的所有新的特征和新的组合,或所公开的装置方法所具备的所有新的特征与新的组合。

其保护范围以该发明的精神和权利要求的范围为限。

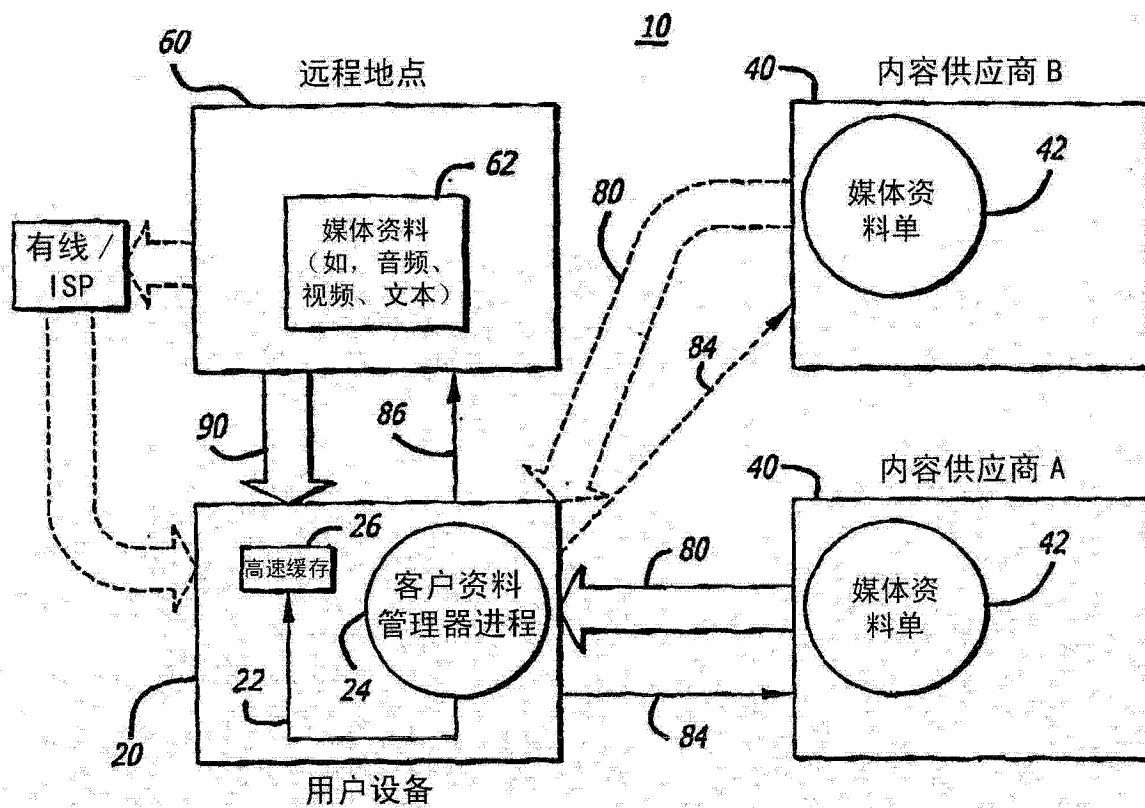


图 1

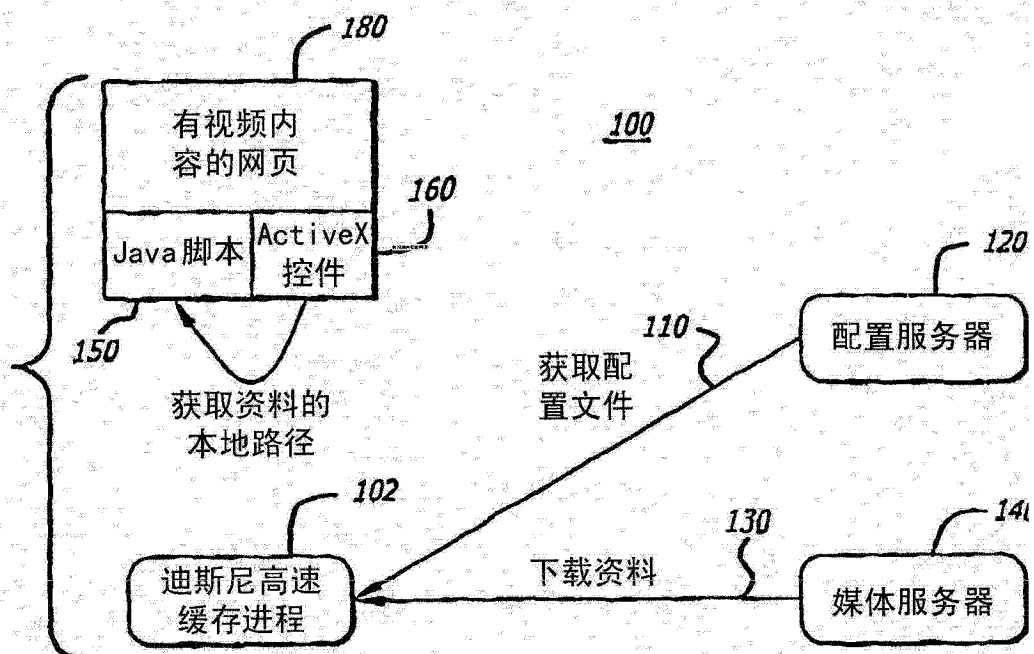


图 2

```

<|ELEMENT channel (item+ | refreshRate | helpUrl? | cookieDomain? |
cookieName?)>
<|ELEMENT item (link | expires | hitCountUrl? | throughput? | trackWithCookie?
)*>

```

The following is an example XML group file:

```

<?xml version="1.0" ?>
  <channel>
    <item>
      <link>http://myserver/myfile.wmv</link>
      <expires>2002-22-03T00:17:15z</expires>
      <hitCountUrl>http://hitserver/hiturl</hitCountUrl>
      <throughput>75</throughput>
      <trackWithCookie/>
    </item>
    <refreshRate>60</refreshRate>
    <helpUrl>http://myserver/help.html</helpURL>
    <cookieName>adAssets</cookieName>
    <cookieDomain>.go.com</cookieDomain>
  </channel>

```

图 3

内部类名	COM/ActiveX 类名
CCacheAsset	CDisCacheAsset
CCacheGroup	CDisCacheGroup
CCacheReference	CCisCacheReference

图 4

错误类型	再次尝试	再试前暂停
Invalid item URL	no	N/A
Server Connection	yes	5 minutes
Server not found	yes	5 minutes
File not on server	yes	15 minutes
Unknown server error	yes	15 minutes
Internal Error	yes	1 hour

图 5

错误类型	再次尝试	再试前暂停
Invalid item URL	no	N/A
Server Connection	yes	5 minutes
Server not found	yes	1 hour
File not on server	yes	1 hour
Unknown server error	yes	15 minutes
Internal Error	yes	1 hour

图 6

资料单

资料 1: { 属性 (1,1); 属性 (1,2);...; 属性 (1,M)}
资料 2: { 属性 (2,1); 属性 (2,2);...; 属性 (2,M)}
⋮
资料 N: { 属性 (N,1); 属性 (N,2);...; 属性 (N,M)}

图 7

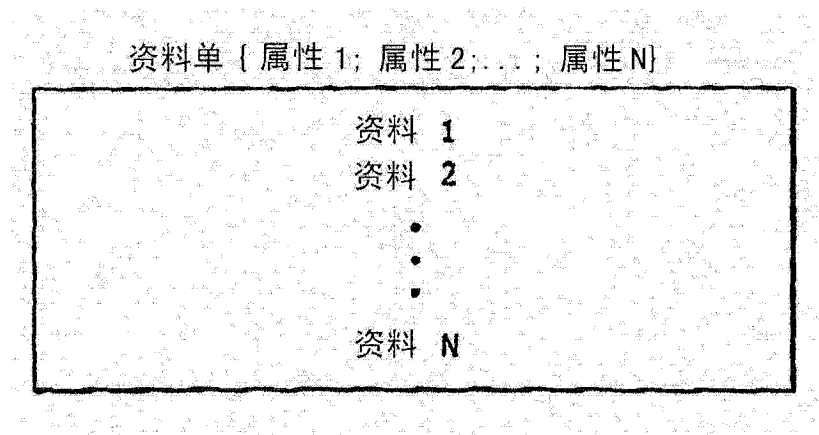


图 8