



(12) 发明专利申请

(10) 申请公布号 CN 106326223 A

(43) 申请公布日 2017. 01. 11

(21) 申请号 201510334446. 2

(22) 申请日 2015. 06. 16

(71) 申请人 广州市动景计算机科技有限公司
地址 510627 广东省广州市天河区黄埔大道西平云路 163 号广电平云广场 B 塔 14 楼

(72) 发明人 梁捷 刘沛祥

(74) 专利代理机构 北京超凡志成知识产权代理事务所(普通合伙) 11371
代理人 饶钱

(51) Int. Cl.
G06F 17/30(2006. 01)

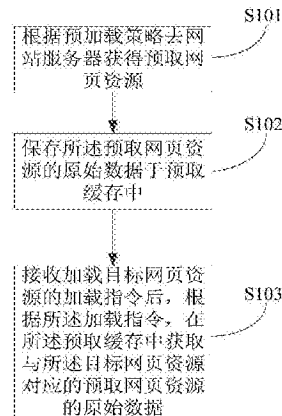
权利要求书2页 说明书9页 附图3页

(54) 发明名称

一种网页资源获取方法和装置

(57) 摘要

本发明提供了一种网页资源获取方法和装置,涉及互联网技术领域。本实施例提供的网页资源获取方法及装置,在预读网页时,将成功预取的预取网页资源的原始数据进行缓存,而不做任何解析处理。当需要加载该网页资源时,只需要从读取本地的预取缓存中的原始数据即可,而无需再发起网络请求。避免了预取后就对预取的预取网页资源的原始数据进行解析、解压缩、解密、预排版及预渲染等操作,系统资源占用相比现有方法大大减低,有效地解除了原有预加载方案中大量占用系统资源导致预加载页面数量较少的限制。



1. 一种网页资源获取方法,其特征在于,所述方法包括:

根据预加载策略去网站服务器获得预取网页资源;

保存所述预取网页资源的原始数据于预取缓存中;

接收加载目标网页资源的加载指令后,根据所述加载指令,在所述预取缓存中获取与
所述目标网页资源对应的预取网页资源的原始数据。

2. 根据权利要求 1 所述的方法,其特征在于,所述在所述预取缓存中获取与所述目标
网页资源对应的预取网页资源的原始数据之前,所述方法还包括:

检查 HTTP 缓存内是否保存有与所述目标网页资源对应的网页数据,如果没有,执行所
述在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据。

3. 根据权利要求 1 所述的方法,其特征在于,所述保存所述预取网页资源的原始数据
于预取缓存中,包括:

将所述预取网页资源的 URL 与预取网页资源的原始数据相对应的保存于预取缓存中。

4. 根据权利要求 3 所述的方法,其特征在于,所述加载指令为加载所述目标网页资源
的 URL,

其中,接收加载目标网页资源的加载指令后,根据所述加载指令,在所述预取缓存中获
取与所述目标网页资源对应的预取网页资源的原始数据,包括:

根据所述目标网页资源的 URL 在所述预取缓存中查找所述目标网页资源的原始数据;

获取所述查找到的目标网页资源的原始数据。

5. 根据权利要求 4 所述的方法,其特征在于,还包括:

当满足预设的清理条件时或者在预设的清理周期内,根据所述预取网页资源保存入所
述预取缓存的时间,以先保存先删除的方式,依次将所述预取缓存中的已经失效的预取网
页资源删除。

6. 一种网页资源获取装置,其特征在于,所述装置包括:

网页资源预取单元,用于根据预加载策略从网站服务器获得预取网页资源;

预取缓存单元,用于存储所述预取网页资源的原始数据;

预取网页资源保存单元,用于将所述预取网页资源的原始数据保存于预取缓存中;

目标网页资源获取单元,用于接收加载目标网页资源的加载指令后,根据所述加载指
令,在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据。

7. 根据权利要求 6 所述的装置,其特征在于,所述装置还包括:

HTTP 缓存检查单元,用于在所述目标网页资源获取单元在所述预取缓存中获取与所述
目标网页资源对应的预取网页资源的原始数据之前,检查所述 HTTP 缓存内是否保存有与
所述目标网页资源对应的网页数据。

8. 根据权利要求 6 所述的装置,其特征在于,所述目标网页资源获取单元包括:

预取缓存查找子单元,用于根据所述目标网页资源的 URL 在所述预取缓存中查找所述
目标网页资源对应的原始数据;

目标网页资源获得子单元,用于获取所述预取缓存查找子单元查找到的原始数据。

9. 根据权利要求 8 所述的装置,其特征在于,所述装置还包括:

预取缓存清理单元,用于当满足预设的清理条件时或者在预设的清理周期内,根据所
述预取网页资源保存入所述预取缓存的时间,以先保存先删除的方式,依次将所述预取缓

存中的已经失效的预取网页资源删除。

一种网页资源获取方法和装置

技术领域

[0001] 本发明涉及互联网技术领域,具体而言,涉及一种网页资源获取方法和装置。

背景技术

[0002] 用户使用浏览器浏览网页过程中,网页的打开速度很大程度受制于网络环境。资源查找的绝大部分时间消耗在网络传输过程。预读技术是其中一个较好解决此问题的方法。现有网页预读的过程是:当用户使用浏览器的过程中,浏览器通过某种方式或手段判断用户接下来将要访问的网页,并提前加载判断需要预加载的网页的相关资源之后对所有的预读页面都进行预排版及预渲染。随后在用户访问其中某个预读网页时,则直接切换至已经完成排版渲染的页面。

[0003] 所以从现有的预读过程来看,现有的预读方案会对预取的网页进行预先的解析、排版及渲染等操作,而这些操作会占用大量系统资源,导致终端系统性能受到很大影响。

发明内容

[0004] 本发明的目的在于提供一种网页资源获取方法和装置,能够降低系统资源的占用,进一步提升使用效果和用户体验。

[0005] 为了实现上述目的,本实施例采用的技术方案如下:

[0006] 第一方面,本实施例提供了一种网页资源获取方法,所述方法包括:

[0007] 根据预加载策略去网站服务器获得预取网页资源;

[0008] 保存所述预取网页资源的原始数据于预取缓存中;

[0009] 接收加载目标网页资源的加载指令后,根据所述加载指令,在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据。

[0010] 结合第一方面,本实施例提供了第一方面的第一种可能的实施方式,其中,所述在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据之前,所述方法还包括:

[0011] 检查 HTTP 缓存内是否保存有与所述目标网页资源对应的网页数据,如果没有,执行所述在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据。

[0012] 结合第一方面,本实施例提供了第一方面的第二种可能的实施方式,其中,所述保存所述预取网页资源的原始数据于预取缓存中,包括:

[0013] 将所述预取网页资源的 URL (Uniform Resource Locator, 统一资源定位符) 与预取网页资源的原始数据相对应的保存于预取缓存中。

[0014] 结合第一方面的第二种可能的实施方式,本实施例提供了第一方面的第三种可能的实施方式,其中,所述加载指令为加载所述目标网页资源的 URL,

[0015] 其中,接收加载目标网页资源的加载指令后,根据所述加载指令,在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据,包括:

[0016] 根据所述目标网页资源的 URL 在所述预取缓存中查找所述目标网页资源的原始

数据；

[0017] 获取所述查找到的目标网页资源的原始数据。

[0018] 结合第一方面的第三种可能的实施方式，本实施例提供了第一方面的第四种可能的实施方式，其中，还包括：

[0019] 当满足预设的清理条件时或者在预设的清理周期内，根据所述预取网页资源保存入所述预取缓存的时间，以先保存先删除的方式，依次将所述预取缓存中的已经失效的预取网页资源删除。

[0020] 第二方面，本实施例还提供了一种网页资源获取装置，所述装置包括：

[0021] 网页资源预取单元，用于根据预加载策略从网站服务器获得预取网页资源；

[0022] 预取缓存单元，用于存储所述预取网页资源的原始数据；

[0023] 预取网页资源保存单元，用于将所述预取网页资源的原始数据保存于预取缓存中；

[0024] 目标网页资源获取单元，用于接收加载目标网页资源的加载指令后，根据所述加载指令，在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据。

[0025] 结合第二方面，本实施例提供了第二方面的第一种可能的实施方式，其中，所述装置还包括：

[0026] HTTP 缓存检查单元，用于在所述目标网页资源获取单元执行所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据之前，检查所述 HTTP 缓存内是否保存有与所述目标网页资源对应的网页数据，如果没有，跳转到所述目标网页资源获取单元。

[0027] 结合第二方面，本实施例提供了第二方面的第二种可能的实施方式，其中，所述目标网页资源获取单元包括：

[0028] 预取缓存查找子单元，用于根据所述目标网页资源的 URL 在所述预取缓存中查找所述目标网页资源对应的原始数据；

[0029] 目标网页资源获得子单元，用于获取所述预取缓存查找子单元查找到的原始数据。

[0030] 结合第二方面的第二种可能的实施方式，本实施例提供了第二方面的第三种可能的实施方式，其中，所述装置还包括：

[0031] 预取缓存清理单元，用于当满足预设的清理条件时或者在预设的清理周期内，根据所述预取网页资源保存入所述预取缓存的时间，以先保存先删除的方式，依次将所述预取缓存中的已经失效的预取网页资源删除。

[0032] 本实施例提供的网页资源获取方法及装置，在预读网页时，将成功预取的预取网页资源的原始数据直接进行缓存，而不做任何解析处理。当需要加载该网页资源时，只需要从读取本地的预取缓存中的原始数据即可，而无需再发起网络请求。避免了预取后就对预取网页资源的原始数据进行解析、解压缩、解密、预排版及预渲染等操作，系统资源占用相比现有方法大大减低，有效地解除了原有预加载方案中大量占用系统资源导致预加载页面数量较少的限制。

[0033] 为使本发明的上述目的、特征和优点能更明显易懂，下文特举较佳实施例，并配合所附附图，作详细说明如下。

附图说明

[0034] 为了更清楚地说明本发明实施例的技术方案,下面将对实施例中所需要使用的附图作简单地介绍,应当理解,以下附图仅示出了本发明的某些实施例,因此不应被看作是对范围的限定,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他相关的附图。通过附图所示,本发明的上述及其它目的、特征和优势将更加清晰。在全部附图中相同的附图标记指示相同的部分。并未刻意按实际尺寸等比例缩放绘制附图,重点在于示出本发明的主旨。

[0035] 图 1 示出了本发明实施例提供的一种网页资源获取方法;

[0036] 图 2 示出了本发明实施例提供的另一种网页资源获取方法;

[0037] 图 3 示出了本发明实施例提供的一种网页资源获取装置;

[0038] 图 4 示出了本发明实施例提供的另一种网页资源获取装置;

[0039] 图 5 示出了本发明实施例提供的另一种网页资源获取装置。

具体实施方式

[0040] 下面将结合本发明实施例中附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本发明一部分实施例,而不是全部的实施例。通常在此处附图中描述和示出的本发明实施例的组件可以以各种不同的配置来布置和设计。因此,以下对在附图中提供的本发明的实施例的详细描述并非旨在限制要求保护的本发明的范围,而是仅仅表示本发明的选定实施例。基于本发明的实施例,本领域技术人员在没有做出创造性劳动的前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0041] 本实施例提供的网页资源获取方法,对于预加载的网页资源不进行解析,直接将网页资源的原始数据保存在本地的预取缓存中,待到确认需要时再将原始数据从预取缓存中取出后生成网页,减少系统资源占用的同时,增大了预加载的范围。

[0042] 参阅图 1,本实施例提供的一种网页资源获取方法,所述方法包括:

[0043] 步骤 S101,根据预加载策略去网站服务器获得预取网页资源。

[0044] 当用户使用浏览器的过程中,浏览器可以通过一些预加载策略预测用户接下来将要访问的网页,并提前查找网页的相关网页资源,进而去网站服务器获得预取网页资源。

[0045] 步骤 S102,保存所述预取网页资源的原始数据于预取缓存中。

[0046] 从网站服务器获得预取网页资源后,将取网页资源的原始数据直接保存与预取缓存中,不进行任何处理,而进不用去调用系统资源。

[0047] 步骤 S103,接收加载目标网页资源的加载指令后,根据所述加载指令,在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据。

[0048] 已经预读的预取数据都是针对用户将要执行的动作的预测,当用户发出网页浏览指令时,浏览器会发出加载目标网页资源的加载指令,所述加载指令中包含了用户需要浏览的目标网页相关信息。当接收到加载指令后,从预取缓存中找到目标网页资源的原始数据。

[0049] 在缓存容量有限的情况下,采用本实施例提供的网页资源获取方法,以原始数据的形式保存预取网页资源,占用空间较小,所以能够预取更多链接对应网页资源。而且对预取的网页资源只进行保存,不进行处理,占用的系统资源非常少。

[0050] 本实施例提供的网页资源获取方法,在预读网页时,将成功预取的预取网页资源的原始数据直接进行缓存,而不做任何解析处理。当需要加载该网页资源时,只需要从读取本地的预取缓存中的原始数据即可,而无需再发起网络请求。避免了预取后就对预取网页资源的原始数据进行解析、解压缩、解密、预排版及预渲染等操作,系统资源占用相比现有方法大大减低,有效地解除了原有预加载方案中大量占用系统资源导致预加载页面数量较少的限制。

[0051] 参阅图 2,本实施例提供的另一种网页资源获取方法,所述方法包括:

[0052] 步骤 S201,根据预加载策略去网站服务器获得预取网页资源。

[0053] 根据浏览器内核的预测结果从预测出来的用户将要访问的网页的网站服务器获得预取网页资源。

[0054] 步骤 S202,将所述预取网页资源的 URL 与预取网页资源的原始数据相对应的保存于预取缓存中。

[0055] 为了查找方便,将预取网页资源以键值形式保存于所述预取缓存,其中,所述预取网页资源的 URL 为键,预取网页资源的原始数据为值。在保存时也可以将一些与预取网页资源相关的其他信息一起保存,比如预取网页资源的有效时间等。

[0056] 步骤 S203,检查 HTTP 缓存内是否保存有与所述目标网页资源对应的网页数据,如果没有,执行所述在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据。

[0057] 网页资源获取装置还包括 HTTP 缓存,所述 HTTP 缓存保存有已经访问过的网页的网页数据。当浏览器发出加载目标网页资源的加载指令时,优先检查所述 HTTP 缓存,如果所述 HTTP 缓存中有与目标网页资源对应的网页数据时,直接调用即可;如果没有,再进行后续的步骤。

[0058] 步骤 S204,根据所述目标网页资源的 URL 在所述预取缓存中查找所述目标网页资源的原始数据。

[0059] 如果在所述 HTTP 中没有查找到与目标网页资源对应的网页数据,则在所述预取缓存查找与所述目标网页资源的原始数据。收到的加载指令为加载所述目标网页资源的 URL,而预取网页资源是以 URL 为键,原始数据为值的方式保存在预取缓存中的,因此只需要查找所述目标网页资源的 URL 对应的原始数据即可。

[0060] 步骤 S205,获取所述查找到的目标网页资源的原始数据。

[0061] 将查找到的目标网页资源的原始数据从预取缓存中取出。

[0062] 步骤 S206,清理预取缓存,当满足预设的清理条件时或者在预设的清理周期内,根据所述预取网页资源保存入所述预取缓存的时间,以先保存先删除的方式,依次将所述预取缓存中的已经失效的预取网页资源删除。

[0063] 具体实践中,预取缓存的空间是有限的,需要及时对其中保存的预取网页资源进行清理,以保证后续的预取网页资源能够保存。而且网页资源通常都是有实效性的,在一定时间过后,该网页资源就会失效,保存在预取缓存中也没有价值,例如预取网页资源的有效时间为 20 分钟,当该预取网页资源在预取缓存中已经保存了 20 分钟,就认为该预取网页资源已经失效了。根据预取网页资源存入预取缓存的时间以及该网页资源的有效时间,可以判断保存在预取缓存中的预取网页资源是否已经失效,再根据系统的配置对已经失效的预

取网页资源删除。可以预先设定清理条件,例如预取缓存的占用率达到一定的数值或者保存的预取网页资源大小超过一定的数值等;也可以按照清理周期进行清理,例如一天一次或者一周一次等。进行清理时,按照先进先出的方式删除已经失效的预取网页资源,即先保存先删除。

[0064] 例如,用户正在通过浏览器浏览网页,通过分析用户的行为,设定的预加载策略,分析出用户下一步可能需要获取的网页资源,网页资源获取装置依次获得这些网页资源的原始数据并保存到预取缓存中。在优选的实施方式中,获得预取网页资源时,是通过 URL 来指向网页资源位置的,每个链接对应的网页资源都有一个唯一的 URL。在用户点击了某一个链接,浏览器发出加载该链接对应的网页资源的加载指令,所述加载指令为加载所述目标网页资源的 URL。如果用户在之前已经浏览过该链接对应的网页,那么就on 直接可以在 HTTP 缓存中找到这个已经浏览过的网页的网页数据,直接调用即可。如果没有,就需要再从预取缓存中进行查找,判断该链接对应的网页资源是否已经预取,查找的方式就是对比已经保存在预取缓存中的预取网页资源的 URL 与加载指令的目标网页资源的 URL,如果有相同的,说明该资源已经预取,直接从本地的预取缓存中取出预取网页资源的原始数据即可,如果没有,浏览器就需要向互联网发出网络请求。将预取网页资源取出后,对原始数据进行解压、解密、页面解析、页面排版、页面渲染等操作,得到目标网页。在使用一段时间后,需要对预取缓存进行清理,以先进先出的方式,将预取缓存中已经失效的预取网页资源删除。清理操作可以通过预设周期来执行,也可以通过预设条件来限定。

[0065] 本实施例提供的网页资源获取方法,在预读网页时,将成功预取的预取网页资源的原始数据直接进行缓存,而不做任何解析处理。当需要加载该网页资源时,只需要从读取本地的预取缓存中的原始数据即可,而无需再发起网络请求。避免了预取后就对预取网页资源的原始数据进行解析、解压缩、解密、预排版及预渲染等操作,系统资源占用相比现有方法大大减低,有效地解除了原有预加载方案中大量占用系统资源导致预加载页面数量较少的限制。

[0066] 参阅图 3,本实施例提供的一种网页资源获取装置,所述装置包括:

[0067] 网页资源预取单元 301,用于根据预加载策略从网站服务器获得预取网页资源。

[0068] 预取缓存单元 302,用于存储所述预取网页资源的原始数据。

[0069] 预取网页资源保存单元 303,用于保存所述预取网页资源的原始数据于预取缓存中。

[0070] 目标网页资源获取单元 304,用于接收加载目标网页资源的加载指令后,根据所述加载指令,在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据。

[0071] 本实施例所提供的网页资源获取装置,在具体实现中,可以由预取引擎和浏览器内核组成,两者相互独立但配合工作,也可以将具有预取功能的预取引擎与浏览器的内核集成在一起构成网页资源获取装置。用户使用浏览器的过程中,网页资源预取单元 301 可以通过预加载策略预测用户接下来将要访问的网页,从网站服务器获得预取网页资源。然后将获得的预取网页资源保存于预取缓存。网页资源获取装置接收到的是预取网页资源的原始数据,需要浏览器内核对原始数据进行处理后,才能得到用户可以浏览的网页。预取网页资源保存单元 303 对接收到的预取网页资源的原始数据不进行处理,而是将其保存于所述预取缓存。因此不用去调用系统资源,同时原始网络流数据的大小是远小于将其处理后

得到的网页的。

[0072] 如果用户执行了打开目标网页的操作,浏览器会发出加载目标网页资源的加载指令到网页资源获取装置。目标网页资源获取单元 304 接收到加载指令后,对预取缓存进行查找,如果在预取缓存中发现与加载指令指示的需要加载的目标网页资源,说明对应预取网页资源的原始数据已经保存在预取缓存了,将该预取网页资源的原始数据从预取缓存中取出。由于预取缓存中的原始数据都是本地资源,浏览器就无需再发起网络请求了。获得目标网页资源的原始数据后,按照现有的对原始数据的处理方式对目标网页资源的原始数据进行处理后,获得用户实现需要打开的网页。

[0073] 缓存的容量是有限的,本实施例提供的网页资源获取装置,以原始数据的形式保存预取网页资源,占用空间较小,所以能够预取更多链接对应网页资源。而且对预取的网页资源只进行保存,不进行处理,占用的系统资源非常少。

[0074] 本实施例所提供的网页资源获取装置,在预读网页时,将成功预取的预取网页资源的原始数据直接进行缓存,而不做任何解析处理。当需要加载该网页资源时,只需要从读取本地的预取缓存中的原始数据即可,而无需再发起网络请求。避免了预取后就对预取网页资源的原始数据进行解析、解压缩、解密、预排版及预渲染等操作,系统资源占用相比现有方法大大减低,有效地解除了原有预加载方案中大量占用系统资源导致预加载页面数量较少的限制。

[0075] 参阅图 4,本实施例提供的另一种网页资源获取装置,所述网页资源获取装置包括用于保存数据的预取缓存,所述装置包括:

[0076] 网页资源预取单元 401,用于根据预加载策略从网站服务器获得预取网页资源。

[0077] 预取缓存单元 402,用于存储所述预取网页资源的原始数据。

[0078] 预取网页资源保存单元 403,用于将所述预取网页资源的 URL 与预取网页资源的原始数据相对应的保存于预取缓存中。

[0079] HTTP 缓存检查单元 404,用于在所述目标网页资源获取单元执行所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据之前,检查所述 HTTP 缓存内是否保存有与所述目标网页资源对应的网页数据。

[0080] 目标网页资源获取单元 405,用于接收加载目标网页资源的加载指令后,根据所述加载指令,在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据。

[0081] 目标网页资源获取单元 405 包括预取缓存查找子单元 406 以及目标网页资源获得子单元 407。

[0082] 预取缓存查找子单元 406,用于根据所述目标网页资源的 URL 在所述预取缓存中查找所述目标网页资源对应的原始数据。

[0083] 目标网页资源获得子单元 407,用于获取所述预取缓存查找子单元 406 查找到的原始数据。

[0084] 预取缓存清理单元 408,用于当满足预设的清理条件时或者在预设的清理周期内,根据所述预取网页资源保存入所述预取缓存的时间,以先保存先删除的方式,依次将所述预取缓存中的已经失效的预取网页资源删除。

[0085] 用户使用浏览器浏览网页时,由网页资源预取单元 401 根据预加载策略从网站服务器获得预取网页资源,对用户的行为进行一个预估。然后预取网页资源保存单元 403

由将所述预取网页资源的 URL 与预取网页资源的原始数据相对应的保存于预取缓存单元 402。如果用户执行了打开目标网页的操作,浏览器会发出加载目标网页资源的加载指令到网页资源获取装置,所述加载指令为加载所述目标网页资源的 URL。首先由 HTTP 缓存检查单元 404 检查所述 HTTP 缓存内是否保存有与所述目标网页资源对应的网页数据。如果用户之前已经浏览过目标网页,那么该网页的网页数据是保存在 HTTP 缓存中的,如果在所述 HTTP 缓存中没有与所述目标网页资源对应的网页数据,则由目标网页资源获取单元 405,根据所述加载指令,在所述预取缓存中获取与所述目标网页资源对应的预取网页资源的原始数据,所述加载指令为加载目标网页资源的 URL。获取目标网页资源的原始数据分为两个子步骤,有目标网页资源获取单元 405 包括的两个子单元完成。预取缓存查找子单元 406 根据所述目标网页资源的 URL 在所述预取缓存中查找所述目标网页资源对应的原始数据;目标网页资源获得子单元 407 获取所述预取缓存查找子单元 406 查找到的原始数据。预取缓存单元 402 的容量是有限的,在使用一段时间后,需要通过预取缓存清理单元 408 对预取缓存进行清理,以先进先出的方式,将预取缓存中已经失效的预取网页资源删除。清理操作可以通过预设周期来执行,也可以通过预设条件来限定。

[0086] 本实施例所提供的网页资源获取装置,其实现原理及产生的技术效果和前述方法实施例相同,为简要描述,装置实施例部分未提及之处,可参考前述方法实施例中相应内容。

[0087] 参阅图 5,本实施例提供的另一种网页资源获取装置 600,包括:处理器 504,存储器 501,总线 502 和通信接口 503,所述处理器 504、通信接口 503 和存储器 501 通过总线 502 连接;处理器 504 用于执行存储器 501 中存储的可执行模块,例如计算机程序。

[0088] 其中,存储器 501 可能包含高速随机存取存储器 (RAM:Random Access Memory),也可能还包括非不稳定的存储器 (non-volatile memory),例如至少一个磁盘存储器。通过至少一个通信接口 503(可以是有线或者无线)实现该系统网元与至少一个其他网元之间的通信连接,可以使用互联网,广域网,本地网,城域网等。

[0089] 总线 502 可以是 ISA 总线、PCI 总线或 EISA 总线等。所述总线可以分为地址总线、数据总线、控制总线等。为便于表示,图 5 中仅用一个双向箭头表示,但并不表示仅有一根总线或一种类型的总线。

[0090] 其中,存储器 501 用于存储程序,所述处理器 504 在接收到执行指令后,执行所述程序,前述本实施例任一实施例揭示的流程定义的装置所执行的方法可以应用于处理器 504 中,或者由处理器 504 实现。

[0091] 处理器 504 可能是一种集成电路芯片,具有信号的处理能力。在实现过程中,上述方法的各步骤可以通过处理器 504 中的硬件的集成逻辑电路或者软件形式的指令完成。上述的处理器 504 可以是通用处理器,包括中央处理器 (Central Processing Unit,简称数据请求端)、网络处理器 (Network Processor,简称 NP) 等;还可以是数字信号处理器 (DSP)、专用集成电路 (ASIC)、现成可编程门阵列 (FPGA) 或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件。可以实现或者执行本实施例中的公开的各方法、步骤及逻辑框图。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。结合本实施例所公开的方法的步骤可以直接体现为硬件译码处理器执行完成,或者用译码处理器中的硬件及软件模块组合执行完成。软件模块可以位于随机存储器,闪存、只读存储器,可编

程只读存储器或者电可擦写可编程存储器、寄存器等本领域成熟的存储介质中。该存储介质位于存储器 501, 处理器 504 读取存储器 501 中的信息, 结合其硬件完成上述方法的步骤。

[0092] 另外, 附图中的流程图和框图显示了根据本发明的多个实施例的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上, 流程图或框图中的每个方框可以代表一个模块、程序段或代码的一部分, 所述模块、程序段或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意, 在有些作为替换的实现中, 方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如, 两个连续的方框实际上可以基本并行地执行, 它们有时也可以按相反的顺序执行, 这依所涉及的功能而定。也要注意的, 框图和 / 或流程图中的每个方框、以及框图和 / 或流程图中的方框的组合, 可以用执行规定的功能或动作的专用的基于硬件的系统来实现, 或者可以用专用硬件与计算机指令的组合来实现。

[0093] 本实施例所提供的进行一种网页资源获取装置的计算机程序产品, 包括存储了程序代码的计算机可读存储介质, 所述程序代码包括的指令可用于执行前面方法实施例中所述的方法, 具体实现可参见方法实施例, 在此不再赘述。

[0094] 所属领域的技术人员可以清楚地了解到, 为描述的方便和简洁, 上述描述的系统、装置和单元的具体工作过程, 可以参考前述方法实施例中的对应过程, 在此不再赘述。

[0095] 在本申请所提供的几个实施例中, 应该理解到, 所揭露的系统、装置和方法, 可以通过其它的方式实现。以上所描述的装置实施例仅仅是示意性的, 例如, 所述单元的划分, 仅仅为一种逻辑功能划分, 实际实现时可以有另外的划分方式, 又例如, 多个单元或组件可以结合或者可以集成到另一个系统, 或一些特征可以忽略, 或不执行。另一点, 所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些通信接口, 装置或单元的间接耦合或通信连接, 可以是电性, 机械或其它的形式。

[0096] 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的, 作为单元显示的部件可以是或者也可以不是物理单元, 即可以位于一个地方, 或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0097] 另外, 在本发明各个实施例中的各功能单元可以集成在一个处理单元中, 也可以是各个单元单独物理存在, 也可以两个或两个以上单元集成在一个单元中。

[0098] 所述功能如果以软件功能单元的形式实现并作为独立的产品销售或使用, 可以存储在一个计算机可读存储介质中。基于这样的理解, 本发明的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的部分可以以软件产品的形式体现出来, 该计算机软件产品存储在一个存储介质中, 包括若干指令用以使得一台计算机设备 (可以是个人计算机, 服务器, 或者网络设备等) 执行本发明各个实施例所述方法的全部或部分步骤。而前述的存储介质包括 :U 盘、移动硬盘、只读存储器 (ROM, Read-Only Memory)、随机存取存储器 (RAM, Random Access Memory)、磁碟或者光盘等各种可以存储程序代码的介质。

[0099] 需要说明的是, 在本文中, 诸如第一和第二等之类的关系术语仅仅用来将一个实体或者操作与另一个实体或操作区分开来, 而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。而且, 术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含, 从而使得包括一系列要素的过程、方法、物品或者设备不仅包括那些要

素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、物品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、物品或者设备中还存在另外的相同要素。

[0100] 在本发明的描述中,还需要说明的是,除非另有明确的规定和限定,术语“设置”、“安装”、“相连”、“连接”应做广义理解,例如,可以是固定连接,也可以是可拆卸连接,或一体地连接;可以是机械连接,也可以是电连接;可以是直接相连,也可以通过中间媒介间接相连,可以是两个元件内部的连通。对于本领域的普通技术人员而言,可以根据具体情况理解上述术语在本发明中的具体含义。

[0101] 以上所述仅为本发明的优选实施例而已,并不用于限制本发明,对于本领域的技术人员来说,本发明可以有各种更改和变化。凡在本发明的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。应注意到:相似的标号和字母在下面的附图中表示类似项,因此,一旦某一项在一个附图中被定义,则在随后的附图中不需要对其进行进一步定义和解释。

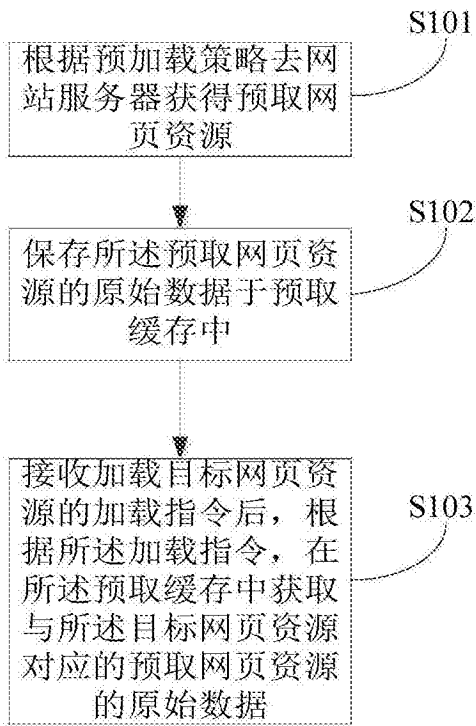


图 1

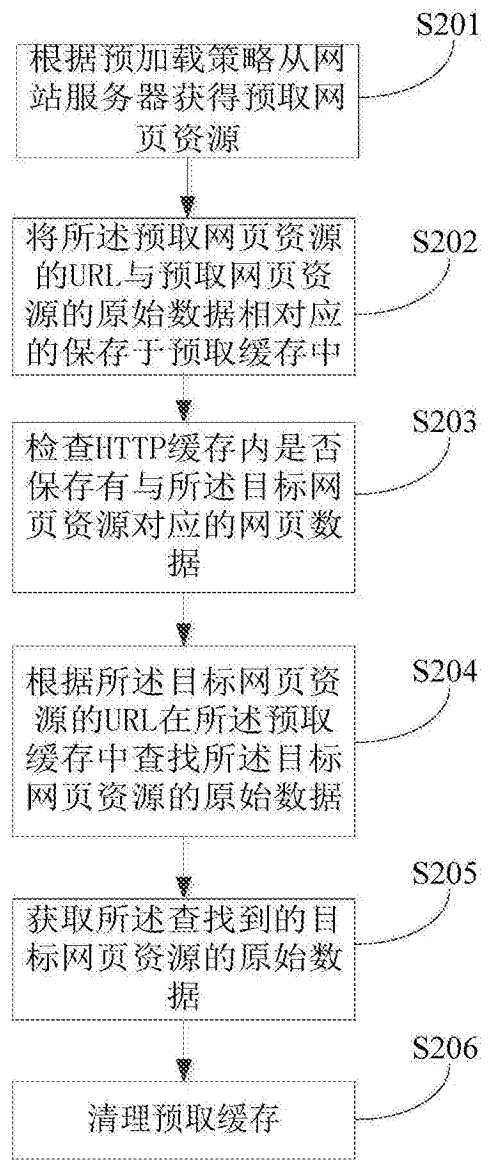


图 2

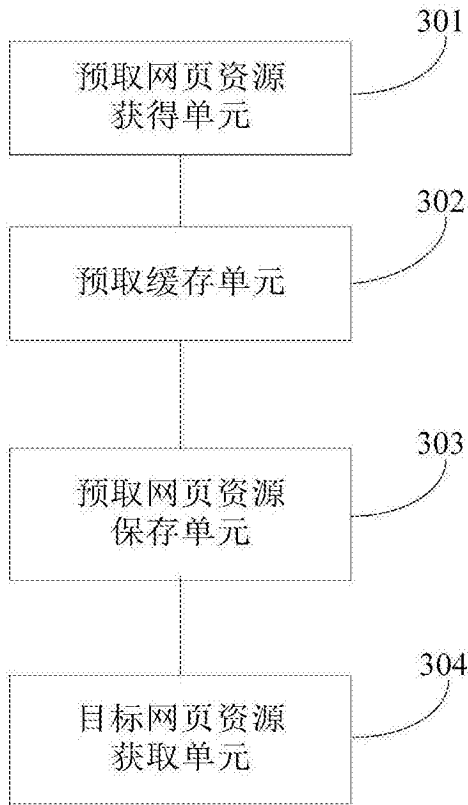


图 3

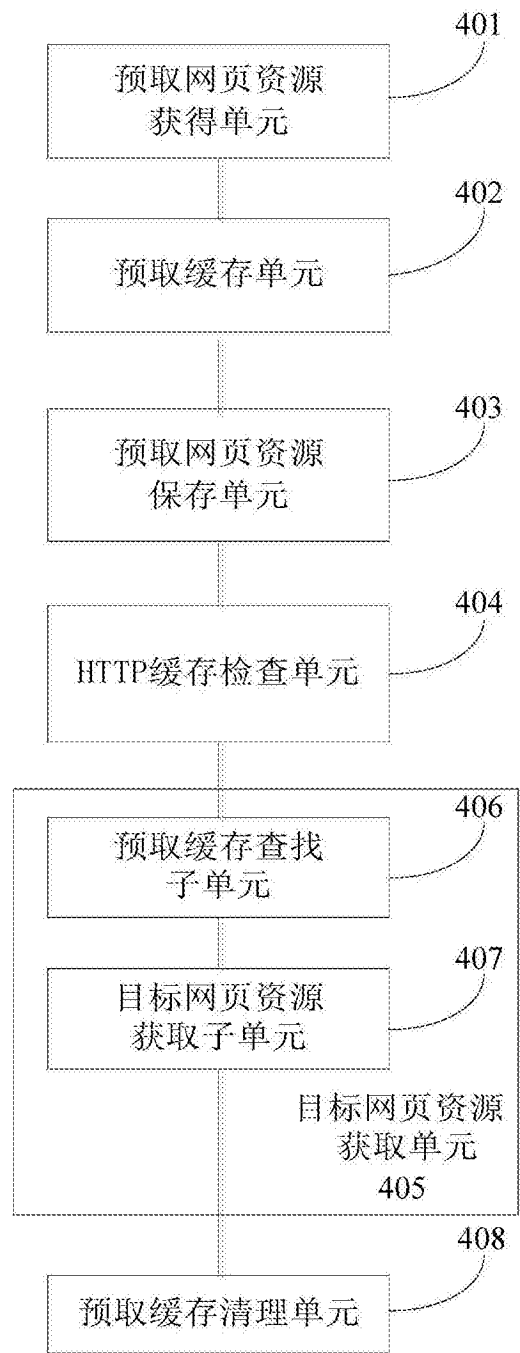


图 4

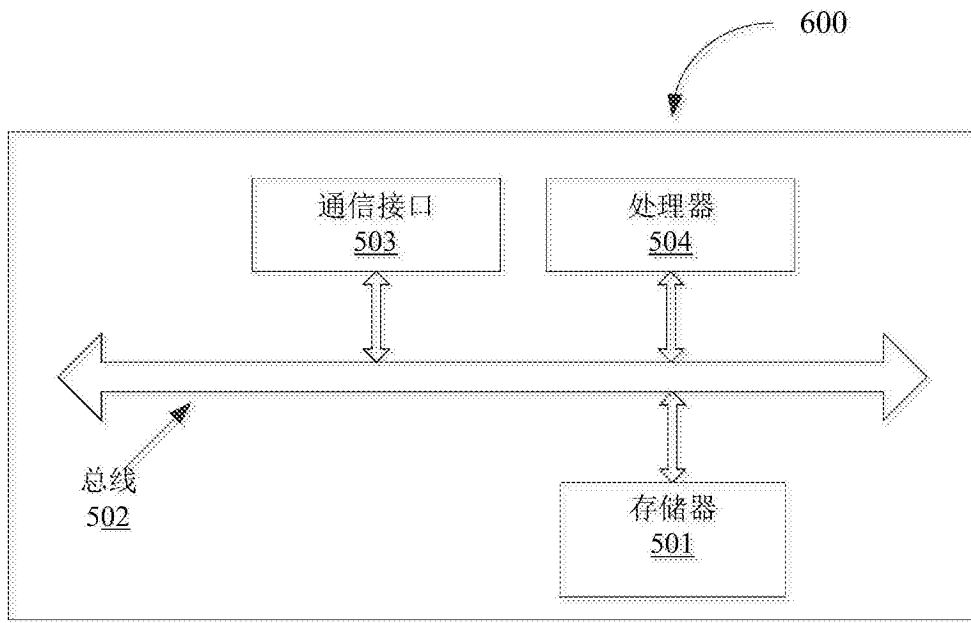


图 5