



(12) 发明专利申请

(10) 申请公布号 CN 103532920 A

(43) 申请公布日 2014. 01. 22

(21) 申请号 201210233920. 9

(22) 申请日 2012. 07. 06

(71) 申请人 腾讯科技(深圳)有限公司

地址 518044 广东省深圳市福田区振兴路赛格科技园2栋东403室

(72) 发明人 侯红君 王俊山 唐孟松

(74) 专利代理机构 广州华进联合专利商标代理有限公司 44224

代理人 王茹 黄晓庆

(51) Int. Cl.

H04L 29/06(2006. 01)

H04L 29/08(2006. 01)

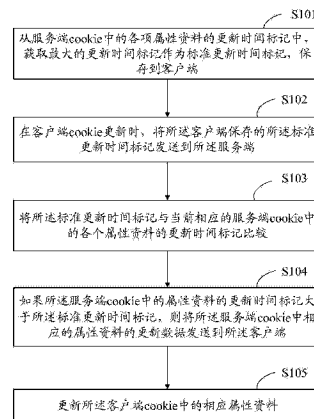
权利要求书2页 说明书7页 附图3页

(54) 发明名称

cookie 更新方法以及 cookie 更新系统

(57) 摘要

本发明提供一种 cookie 更新方法及其系统, 所述方法包括步骤: 从服务端 cookie 中的各个属性资料的更新时间标记中, 获取最大的更新时间标记作为标准更新时间标记, 保存到客户端; 在客户端 cookie 更新时, 将所述客户端保存的所述标准更新时间标记发送到所述服务端; 将所述标准更新时间标记与当前相应的服务端 cookie 中的各个属性资料的更新时间标记比较; 如果所述服务端 cookie 中的属性资料的更新时间标记大于所述标准更新时间标记, 则将相应的所述服务端 cookie 中的属性资料的更新数据发送到所述客户端, 更新所述客户端 cookie 中的相应属性资料。能够减少客户端 cookie 更新对客户端和服务端之间的数据传输资源占用, 提高客户端 cookie 更新效率。



1. 一种 cookie 更新方法,其特征在于,包括步骤:

从服务端 cookie 中的各个属性资料的更新时间标记中,获取最大的更新时间标记作为标准更新时间标记,保存到客户端;

在客户端 cookie 更新时,将所述客户端保存的所述标准更新时间标记发送到所述服务端;

将所述标准更新时间标记与当前相应的服务端 cookie 中的各个属性资料的更新时间标记比较;

如果所述服务端 cookie 中的属性资料的更新时间标记大于所述标准更新时间标记,则将所述服务端 cookie 中相应的属性资料的更新数据发送到所述客户端,更新所述客户端 cookie 中的相应属性资料。

2. 如权利要求 1 所述的 cookie 更新方法,其特征在于,在所述服务端 cookie 中的属性资料的更新时间标记大于所述标准更新时间标记时,进一步从所述服务端 cookie 中的各个属性资料的更新时间标记中,获取最大的更新时间标记作为所述标准更新时间标记,保存到所述客户端。

3. 如权利要求 1 或者 2 所述的 cookie 更新方法,其特征在于,将相应的所述服务端 cookie 中的属性资料的更新数据发送到所述客户端,更新所述客户端 cookie 中的相应属性资料的步骤包括:

按照预定的数据包打包协议将所述服务端 cookie 中的属性资料的更新数据打包成更新数据包,并将所述更新数据包发送至所述客户端;

按照与所述数据包打包协议相对应的数据包解析协议,解析所述更新数据包,获得所述客户端 cookie 中的相应属性资料的更新数据,对所述客户端 cookie 中的相应属性资料进行更新。

4. 如权利要求 1 或者 2 所述的 cookie 更新方法,其特征在于,所述更新时间标记以及所述标准更新时间标记为:

所述服务端 cookie 中的各个属性资料的更新版本号。

5. 如权利要求 1 或者 2 所述的 cookie 更新方法,其特征在于,以预定时间间隔更新所述客户端 cookie。

6. 一种 cookie 更新系统,其特征在于,包括:包括客户端和服务端;

所述客户端用于保存所述服务端发送的标准更新时间标记;并在客户端 cookie 更新时,将所述标准更新时间标记发送到所述服务端;根据服务端 cookie 中的属性资料的更新数据,更新所述客户端 cookie 中的相应属性资料;

所述服务端用于从所述服务端 cookie 中的各个属性资料的更新时间标记中,获取最大的更新时间标记作为所述标准更新时间标记发送到所述客户端;接收所述客户端发送的所述标准更新时间标记,并将所述标准更新时间标记与当前相应的服务端 cookie 中的各个属性资料的更新时间标记比较;在所述服务端 cookie 中的属性资料的更新时间标记大于所述标准更新时间标记,将所述服务端 cookie 中相应的属性资料的更新数据发送到所述客户端。

7. 如权利要求 6 所述的 cookie 更新系统,其特征在于,所述服务端在所述服务端 cookie 中的属性资料的更新时间标记大于所述标准更新时间标记时,进一步从所述服务端

cookie 中的各个属性资料的更新时间标记中,获取最大的更新时间标记作为所述标准更新时间标记发送到所述客户端。

8. 如权利要求 6 或者 7 所述的 cookie 更新系统,其特征在于,所述服务端按照预定的数据包打包协议将所述服务端 cookie 中的属性资料的更新数据打包成更新数据包,并将所述更新数据包发送至所述客户端;

所述客户端按照与所述数据包打包协议相对应的数据包解析协议,解析所述更新数据包,获得所述客户端 cookie 中的相应属性资料的更新数据,对所述客户端 cookie 中的相应属性资料进行更新。

9. 如权利要求 6 或者 7 所述的 cookie 更新系统,其特征在于,所述更新时间标记以及所述标准更新时间标记为:所述服务端 cookie 中的各个属性资料的更新版本号。

10. 如权利要求 6 或者 7 所述的 cookie 更新系统,其特征在于,所述客户端以预定时间间隔更新所述客户端 cookie。

## cookie 更新方法以及 cookie 更新系统

### 技术领域

[0001] 本发明涉及 cookie 技术的领域,特别是涉及 cookie 更新方法,以及 cookie 更新系统。

### 背景技术

[0002] Cookie(或者 cookies),是指网站或者应用程序的服务器系统为了辨别用户身份、进行 session(会话)跟踪而储存在用户本地终端上的数据,cookie 数据通常经过加密。

[0003] 当用户访问相应的网站或者应用程序时,服务器系统为用户创建 cookie,并将 cookie 保存在客户端(如浏览器,或者应用程序的客户端),并可在所述 cookie 中记载一些关键的通信信息,例如用户的历史记录、登录信息、个人信息以及个性化设置等等。用户在 cookie 的有效期内再次访问相同的网站或者应用程序时,客户端会将相关的操作请求和 cookie 一起发送至网站或者应用程序的服务器,服务器在响应客户端的操作请求时,会根据所述 cookie 中的属性资料来判断,加载用户的历史记录、登录信息、个人信息以及个性化设置等等,非常方便。

[0004] 在实现过程中,常常会出现客户端 cookie 和服务端 cookie 不统一的情况,例如在用户长时间浏览某一网站时,在服务端的用户浏览历史记录不断更新,从而使服务端 cookie 也不断更新;而客户端 cookie 则无法实现自动更新,因此需要根据服务端 cookie 进行客户端 cookie 的更新。

[0005] cookie 的更新机制如下:

[0006] 客户端将整个 cookie 的各项属性资料的版本情况发送到服务端,服务端判定 cookie 中的属性资料的版本,例如个人信息、好友信息等,是否有更新,如果有更新,则服务端通知客户端更新所述属性资料,并且更新完毕后,再同步 cookie 内容。

[0007] 例如,即时通讯录服务系统,有关用户和软件特性有多项 cookie 的属性资料:个人资料、头像、签名、开通的服务和具有的级别属性。上述各项属性资料被写入到一个 cookie 中。在客户端每次登陆服务端时,发送到服务端,服务端根据所述 cookie 调取该用户的个人资料、头像、签名、开通的服务和具有的级别属性。

[0008] 假设每十分钟需要对客户端 cookie 更新一次时,则客户端需要每十分钟将向所述服务端发送更新请求,所述更新请求包括所述客户端 cookie 的各项属性资料,以及其对应的版本号;而服务端接收到所述更新请求后,根据服务端 cookie 的各项属性资料的版本号,判断是否需要所述客户端 cookie 进行更新。如果要更新,则下发所述服务端 cookie 的各项属性资料的更新数据到所述客户端。所述客户端接收所述更新数据之后,完成客户端 cookie 的更新。

[0009] 然而,上述更新过程中,需要客户端先将客户端 cookie 的各项属性资料,以及其对应的版本号发送到服务端,在所述客户端 cookie 包含的属性资料项数较多时,从客户端上行数据流量也会较大。导致在所述客户端和服务端之间的数据传输资源被占用较多,并且延长了更新请求和数据传输的时间,影响客户端 cookie 更新的效率。

## 发明内容

[0010] 针对上述背景技术中存在的问题,本发明的目的在于提供一种能够减少客户端 cookie 更新对客户端和服务端之间的数据传输资源占用,提高客户端 cookie 更新效率的 cookie 更新方法。

[0011] 一种 cookie 更新方法,包括步骤:

[0012] 从服务端 cookie 中的各个属性资料的更新时间标记中,获取最大的更新时间标记作为标准更新时间标记,保存到客户端;

[0013] 在客户端 cookie 更新时,将所述客户端保存的所述标准更新时间标记发送到所述服务端;

[0014] 将所述标准更新时间标记与当前相应的服务端 cookie 中的各个属性资料的更新时间标记比较;

[0015] 如果所述服务端 cookie 中的属性资料的更新时间标记大于所述标准更新时间标记,则将相应的所述服务端 cookie 中的属性资料的更新数据发送到所述客户端,更新所述客户端 cookie 中的相应属性资料。

[0016] 本发明的 cookie 更新方法中,通过获取服务端 cookie 中的各个属性资料的最大的更新时间标记作为标准更新时间标记保存到客户端。在客户端 cookie 更新时,只需要将所述标准更新时间标记发送到服务端,而无需发送客户端 cookie 的所有属性资料以及其对应的版本号,节省了大量的上行数据量。并且在服务端接收所述标准更新时间标记后,只将当前服务端 cookie 中更新时间标记比所述标准更新时间标记大的属性资料的更新数据发送到所述客户端,而无需将服务端 cookie 中的所有属性数据的更新数据都发送到所述客户端,同样可以降低对下行数据通道的占用,提高客户端 cookie 更新效率。

[0017] 针对上述背景技术中存在的问题,本发明的目的还在于提供一种能够减少客户端 cookie 更新对客户端和服务端之间的数据传输资源占用,提高客户端 cookie 更新效率的 cookie 更新系统。

[0018] 一种 cookie 更新系统,包括:包括客户端和服务端;

[0019] 所述客户端用于保存所述服务端发送的标准更新时间标记;并在客户端 cookie 更新时,将所述标准更新时间标记发送到所述服务端;根据服务端 cookie 中的属性资料的更新数据,更新所述客户端 cookie 中的相应属性资料;

[0020] 所述服务端用于从所述服务端 cookie 中的各个属性资料的更新时间标记中,获取最大的更新时间标记作为所述标准更新时间标记发送到所述客户端;接收所述客户端发送的所述标准更新时间标记,并将所述标准更新时间标记与当前相应的服务端 cookie 中的各个属性资料的更新时间标记比较;在所述服务端 cookie 中的属性资料的更新时间标记大于所述标准更新时间标记,将相应的所述服务端 cookie 中的属性资料的更新数据发送到所述客户端。

[0021] 本发明的 cookie 更新系统中,所述服务端获取服务端 cookie 中的各个属性资料的最大的更新时间标记作为标准更新时间标记保存到客户端。在客户端 cookie 更新时,所述客户端只需要将所述标准更新时间标记发送到服务端,而无需发送客户端 cookie 的所有属性资料以及其对应的版本号,节省了大量的上行数据量。并且服务端接收所述标准更

新时间标记后,只将当前服务端 cookie 中更新时间标记比所述标准更新时间标记大的属性资料的更新数据发送到所述客户端,而无需将服务端 cookie 中的所有属性数据的更新数据都发送到所述客户端,同样可以降低对下行数据通道的占用,提高客户端 cookie 更新效率。

#### 附图说明

- [0022] 图 1 是本发明 cookie 更新方法的步骤流程图；  
[0023] 图 2 是服务端 cookie 的数据结构示意图；  
[0024] 图 3 是本发明 cookie 更新方法的一个实施例的示意图；  
[0025] 图 4 是本发明 cookie 更新系统的结构示意图。

#### 具体实施方式

- [0026] 请参阅图 1,图 1 是本发明 cookie 更新方法的步骤流程图。
- [0027] 所述 cookie 更新方法包括以下步骤：
- [0028] S101,从服务端 cookie 中的各项属性资料的更新时间标记中,获取最大的更新时间标记作为标准更新时间标记,保存到客户端；
- [0029] 在本步骤中,所述更新时间标记可以是所述服务端 cookie 中的各项属性资料的更新时间,或者版本号,或者其他可以表示各项属性资料的更新时间的标记。所述更新时间标记最大,即对应的所述属性资料的更新时间最迟,更新数据最新。
- [0030] 假设所述服务端 cookie 中包含多项属性资料,比如等级,个人资料,头像,昵称,年龄,是否开通会员,开通新应用等。则所述服务端 cookie 在各项属性资料更新时都记录其各自的更新时间,或者版本号,通常由服务端将更新时间,或者版本号记录在对应的服务端 cookie 中。
- [0031] 在本实施方式中,以各项属性资料的更新版本号作为所述更新时间标记。即在本步骤中,首先获取所述服务端 cookie 中的各项属性资料的更新版本号。找出最大的更新版本号。
- [0032] 如图 2 所示,所述服务端 cookie 包含 Ck1、Ck2、Ck3、Ck4、Ck5、Ck6、Ck7 共七项属性资料,其各自对应的更新版本号是 Ck1 的版本号 T0 最大。因此,在本步骤中获取所述更新版本号 T0 作为所述标准更新时间标记,将所述更新版本号 T0 从服务端发送到客户端,并在客户端保存所述更新版本号 T0。
- [0033] S102,在客户端 cookie 更新时,将所述客户端保存的所述标准更新时间标记发送到所述服务端；
- [0034] 所述客户端 cookie 的更新时间可以根据需要设定,例如设定以预定时间间隔更新所述客户端 cookie,如设定每隔 10 分钟更新一次的更新机制,即使客户端丢失了更新数据,对也不会造成太大的影响,下次进行更新的时候依然能够获得更新数据。又例如在客户端或服务端触发某种操作时,触发更新所述客户端 cookie,比如用户操作,点击。
- [0035] 在本步骤中,将所述客户端保存的所述标准更新时间标记发送到所述服务端,亦即,将上述步骤 1 中保存在客户端的所述更新版本号 T0 发送到服务端。
- [0036] S103,将所述标准更新时间标记与当前相应的服务端 cookie 中的各个属性资料

的更新时间标记比较；

[0037] 因为从所述服务端将所述标准更新时间标记发送到客户端保存，到所述客户端 cookie 更新时，将所述标准更新时间标记发送到所述服务端之间，有一定时间间隔。在这一时间间隔之内，所述服务端 cookie 的各项属性资料可能发生更新，例如用户在这一时间间隔之内产生了新的浏览历史记录，或者设定了新的个人资料，又或者开通了新的服务，都将使所述服务端 cookie 中对应的属性资料发生更新，并且所述服务端 cookie 中对应的属性资料的更新时间标记也会刷新，变得比所述标准更新时间标记大。

[0038] 因此，将所述标准更新时间标记与当前相应的服务端 cookie 中的各个属性资料的更新时间标记比较，即可判断更新时间迟于所述标准更新时间标记的属性资料是哪一项，亦即哪一项属性资料在服务端发出所述标准更新时间标记之后发生了更新，与客户端 cookie 中对应的属性资料不同步。亦即判断出所述客户端 cookie 中需要更新的属性资料是哪一项。

[0039] S104，如果所述服务端 cookie 中的属性资料的更新时间标记大于所述标准更新时间标记，则将所述服务端 cookie 中相应的属性资料的更新数据发送到所述客户端；

[0040] 根据上述步骤中的对比，判断出所述客户端 cookie 中需要更新的属性资料之后，将所述服务端 cookie 中相应的属性资料的更新数据发送到所述客户端。

[0041] 例如假设判断出有两项属性资料 Ck3、Ck5 的版本号 T1 和 T2， $T2 > T1 > T0$ ，则两项属性资料 Ck3、Ck5 所对应的属性资料：头像和个人资料需要更新，则将所述头像和个人资料的更新数据发送至所述客户端。

[0042] 在本实施方式中，在服务端，按照预定的数据包打包协议将所述服务端 cookie 中的属性资料的更新数据打包成更新数据包，并将所述更新数据包发送至所述客户端；

[0043] 客户端接收之后，按照与所述数据包打包协议相对应的数据包解析协议，解析所述更新数据包，获得所述客户端 cookie 中的相应属性资料的更新数据。

[0044] 因为只发送部分属性资料的更新数据，所以客户端和服务端之间需要定义相应的传输协议以便客户端可以准确识别接收的数据。

[0045] 因此，服务端根据预先设定的数据包打包协议，将这些需要更新的属性资料的更新数据封装成客户端可读的数据包，发送给所述客户端，客户端根据数据包解析协议进行解析，准确获取所述更新数据包中的相应属性资料的更新数据后，就可以根据对所述客户端 cookie 中的相应属性资料进行更新。

[0046] 这里的所述数据包打包协议主要包括数据类型和数据包的数据内容，所述客户端根据这个数据包的数据类型来执行对应的处理逻辑。

[0047] S105，更新所述客户端 cookie 中的相应属性资料。

[0048] 按照上述的例子，在本步骤中，根据所述属性资料 Ck3、Ck5 所对应的头像和个人资料的更新数据，更新客户端 cookie 中的头像和个人资料两项属性资料。

[0049] 本发明的 cookie 更新方法中，通过获取服务端 cookie 中的各个属性资料的最大的更新时间标记作为标准更新时间标记保存到客户端。在客户端 cookie 更新时，只需要将所述标准更新时间标记发送到服务端，而无需发送客户端 cookie 的所有属性资料以及其对应的版本号，节省了大量的上行数据量。并且在服务端接收所述标准更新时间标记后，只将当前服务端 cookie 中更新时间标记比所述标准更新时间标记大的属性资料的更新数据

发送到所述客户端,而无需将服务端 cookie 中的所有属性数据的更新数据都发送到所述客户端,同样可以降低对下行数据通道的占用,提高客户端 cookie 更新效率。

[0050] 进一步地,在执行步骤 S105,更新所述客户端 cookie 中的相应属性资料之后,进一步执行以下步骤:

[0051] 从所述服务端 cookie 中的各个属性资料的更新时间标记中,获取最大的更新时间标记作为所述标准更新时间标记,更新所述客户端保存的标准更新时间标记。

[0052] 亦即,在完成所述客户端 cookie 的更新之后,重新从所述服务端 cookie 中的各个属性资料的更新时间标记中,获取最大的更新时间标记作为所述标准更新时间标记,发送至所述客户端更新所述客户端保存的标准更新时间标记。例如上述例子中,将最大的版本号 T2 发送到所述客户端,作为所述客户端新的标准更新时间标记。

[0053] 这样使客户端在下次需要更新所述客户端 cookie 时,可以再次根据更新后的所述标准更新时间标记向服务端请求更新数据,使每次客户端 cookie 的更新都只需要在客户端和服务端之间传输所述标准更新时间标记和需要更新的部分属性资料的更新数据,降低对下行数据通道的占用,提高客户端 cookie 更新效率。

[0054] 请参阅图 3,图 3 是本发明 cookie 更新方法的一个实施例的示意图。

[0055] 在本实施例中,客户端是智能手机。

[0056] 所述服务端 cookie 包含 Ck1、Ck2、Ck3、Ck4、Ck5、Ck6、Ck7 共七项属性资料,其各自对应的更新版本号是 Ck1 的版本号 T0 最大。因此将所述版本号 T0 发送到手机;

[0057] 手机保存所述版本号 T0 后,在需要进行客户端 cookie 更新时将所述版本号 T0 发送到服务端;

[0058] 服务端在此时间间隔内进行了服务端 cookie 的数据更新,其中两项属性资料的版本号 T1 和 T2,  $T2 > T1 > T0$ , 则对应的两项属性资料有更新。

[0059] 将所述两项属性资料的更新数据发送到所述客户端后,将当前最大的属性资料的版本号 T2 也发送到所述客户端;

[0060] 客户端保存所述版本号 T2 后,在下次需要进行客户端 cookie 更新时将所述版本号 T2 发送到服务端;

[0061] 如果当前的服务端 cookie 的属性资料的版本号都小于 T2,则无需更新,客户端每 N 分钟发送一次包含 T2 的 Hello 包,检查客户端 cookie 的更新情况。

[0062] 请参阅图 4,图 4 是本发明 cookie 更新系统的结构示意图。

[0063] 所述 cookie 更新系统包括:包括客户端 11 和服务端 12;

[0064] 所述客户端 11 用于保存所述服务端发送的标准更新时间标记;并在客户端 cookie 更新时,将所述标准更新时间标记发送到所述服务端 12;根据服务端 cookie 中的属性资料的更新数据,更新所述客户端 cookie 中的相应属性资料;

[0065] 所述服务端 12 用于从所述服务端 cookie 中的各个属性资料的更新时间标记中,获取最大的更新时间标记作为所述标准更新时间标记发送到所述客户端 11;接收所述客户端 11 发送的所述标准更新时间标记,并将所述标准更新时间标记与当前相应的服务端 cookie 中的各个属性资料的更新时间标记比较;在所述服务端 cookie 中的属性资料的更新时间标记大于所述标准更新时间标记,将所述服务端 cookie 中相应的属性资料的更新数据发送到所述客户端 12。



[0066] 本发明的 cookie 更新系统中,所述服务端获取服务端 cookie 中的各个属性资料的最大的更新时间标记作为标准更新时间标记保存到客户端。在客户端 cookie 更新时,所述客户端只需要将所述标准更新时间标记发送到服务端,而无需发送客户端 cookie 的所有属性资料以及其对应的版本号,节省了大量的上行数据量。并且服务端接收所述标准更新时间标记后,只将当前服务端 cookie 中更新时间标记比所述标准更新时间标记大的属性资料的更新数据发送到所述客户端,而无需将服务端 cookie 中的所有属性数据的更新数据都发送到所述客户端,同样可以降低对下行数据通道的占用,提高客户端 cookie 更新效率。

[0067] 其中,所述更新时间标记可以是所述服务端 cookie 中的各项属性资料的更新时间,或者版本号,或者其他可以表示各项属性资料的更新时间的标记。所述更新时间标记最大,即对应的所述属性资料的更新时间最迟,更新数据最新。

[0068] 假设所述服务端 cookie 中包含多项属性资料,比如等级,个人资料,头像,昵称,年龄,是否开通会员,开通新应用等。则所述服务端 cookie 在各项属性资料更新时都记录其各自的更新时间,或者版本号,通常由所述服务端 12 将更新时间,或者版本号记录在对应的服务端 cookie 中。

[0069] 在本实施方式中,以各项属性资料的更新版本号作为所述更新时间标记。即所述服务端 12 首先获取所述服务端 cookie 中的各项属性资料的更新版本号。找出最大的更新版本号。

[0070] 所述客户端 11 在客户端 cookie 更新时,将所述客户端保存的所述标准更新时间标记发送到所述服务端 12;

[0071] 所述客户端 cookie 的更新时间可以根据需要设定,例如设定以预定时间间隔更新所述客户端 cookie,如设定每隔 10 分钟更新一次的更新机制,即使所述客户端 11 丢失了更新数据,对也不会造成太大的影响,下次进行更新的时候依然能够获得更新数据。又例如在所述客户端 11 或所述服务端 12 触发某种操作时,触发更新所述客户端 cookie,比如用户点击、操作等。

[0072] 所述服务端 12 将所述标准更新时间标记与当前相应的服务端 cookie 中的各个属性资料的更新时间标记比较;

[0073] 因为从所述服务端 12 将所述标准更新时间标记发送到客户端 11 保存,到所述客户端 cookie 更新时,将所述标准更新时间标记发送到所述服务端 12 之间,有一定时间间隔。在这一时间间隔之内,所述服务端 cookie 的各项属性资料可能发生更新,例如用户在这一时间间隔之内产生了新的浏览历史记录,或者设定了新的个人资料,又或者开通了新的服务,都将使所述服务端 cookie 中对应的属性资料发生更新,并且所述服务端 cookie 中对应的属性资料的更新时间标记也会刷新,变得比所述标准更新时间标记大。

[0074] 因此,所述服务端 12 将所述标准更新时间标记与当前相应的服务端 cookie 中的各个属性资料的更新时间标记比较,即可判断更新时间迟于所述标准更新时间标记的属性资料是哪一项,亦即哪一项属性资料在服务端发出所述标准更新时间标记之后发生了更新,与客户端 cookie 中对应的属性资料不同步。亦即判断出所述客户端 cookie 中需要更新的属性资料是哪一项。

[0075] 所述服务端 12 判断出所述客户端 cookie 中需要更新的属性资料之后,将所述服

务端 cookie 中相应的属性资料的更新数据发送到所述客户端 11。

[0076] 在本实施方式中,所述服务端 12 按照预定的数据包打包协议将所述服务端 cookie 中的属性资料的更新数据打包成更新数据包,并将所述更新数据包发送至所述客户端 11;

[0077] 所述客户端 11 接收之后,按照与所述数据包打包协议相对应的数据包解析协议,解析所述更新数据包,获得所述客户端 cookie 中的相应属性资料的更新数据。

[0078] 因为只发送部分属性资料的更新数据,所以客户端 11 和服务端 12 之间需要定义相应的传输协议以便客户端 11 可以准确识别接收的数据。

[0079] 因此,所述服务端 12 根据预先设定的数据包打包协议,将这些需要更新的属性资料的更新数据封装成客户端 11 可读的数据包,发送给所述客户端 11,所述客户端 11 根据数据包解析协议进行解析,准确获取所述更新数据包中的相应属性资料的更新数据后,就可以根据对所述客户端 cookie 中的相应属性资料进行更新。这里的所述数据包打包协议主要包括数据类型和数据包的数据内容,所述客户端 11 根据这个数据包的数据类型来执行对应的处理逻辑。

[0080] 作为一种优选实施方式,所述服务端 12 在判断所述服务端 cookie 中的属性资料的更新时间标记大于所述标准更新时间标记时,进一步从所述服务端 cookie 中的各个属性资料的更新时间标记中,获取最大的更新时间标记作为所述标准更新时间标记,更新所述客户端 11 保存的标准更新时间标记。

[0081] 亦即,在完成所述客户端 cookie 的更新之后,重新根据所述服务端 cookie 中的各个属性资料的更新时间标记中,最大的更新时间标记产生所述标准更新时间标记,发送至所述客户端 11,更新所述客户端 11 保存的标准更新时间标记。

[0082] 这样所述客户端 11 在下一次更新所述客户端 cookie 时,可以再次根据更新后的所述标准更新时间标记向服务端 12 请求更新数据,使每次客户端 cookie 的更新都只需要在客户端 11 和服务端 12 之间传输所述标准更新时间标记和需要更新的部分属性资料的更新数据,降低对下行数据通道的占用,提高客户端 cookie 更新效率。

[0083] 本领域普通技术人员可以理解实现上述实施方式中的全部或部分流程,以及对应的控制系统,是可以通过计算机程序来指令相关的硬件来完成,所述的程序可存储于一计算机可读取存储介质中,该程序在执行时,可包括如上述各实施方式的流程。其中,所述的存储介质可为磁碟、光盘、只读存储记忆体(Read-Only Memory, ROM)或随机存储记忆体(Random Access Memory, RAM)等。

[0084] 以上所述实施例仅表达了本发明的几种实施方式,其描述较为具体和详细,但不能因此而理解为对本发明专利范围的限制。应当指出的是,对于本领域的普通技术人员来说,在不脱离本发明构思的前提下,还可以做出若干变形和改进,这些都属于本发明的保护范围。因此,本发明的保护范围应以所附权利要求为准。

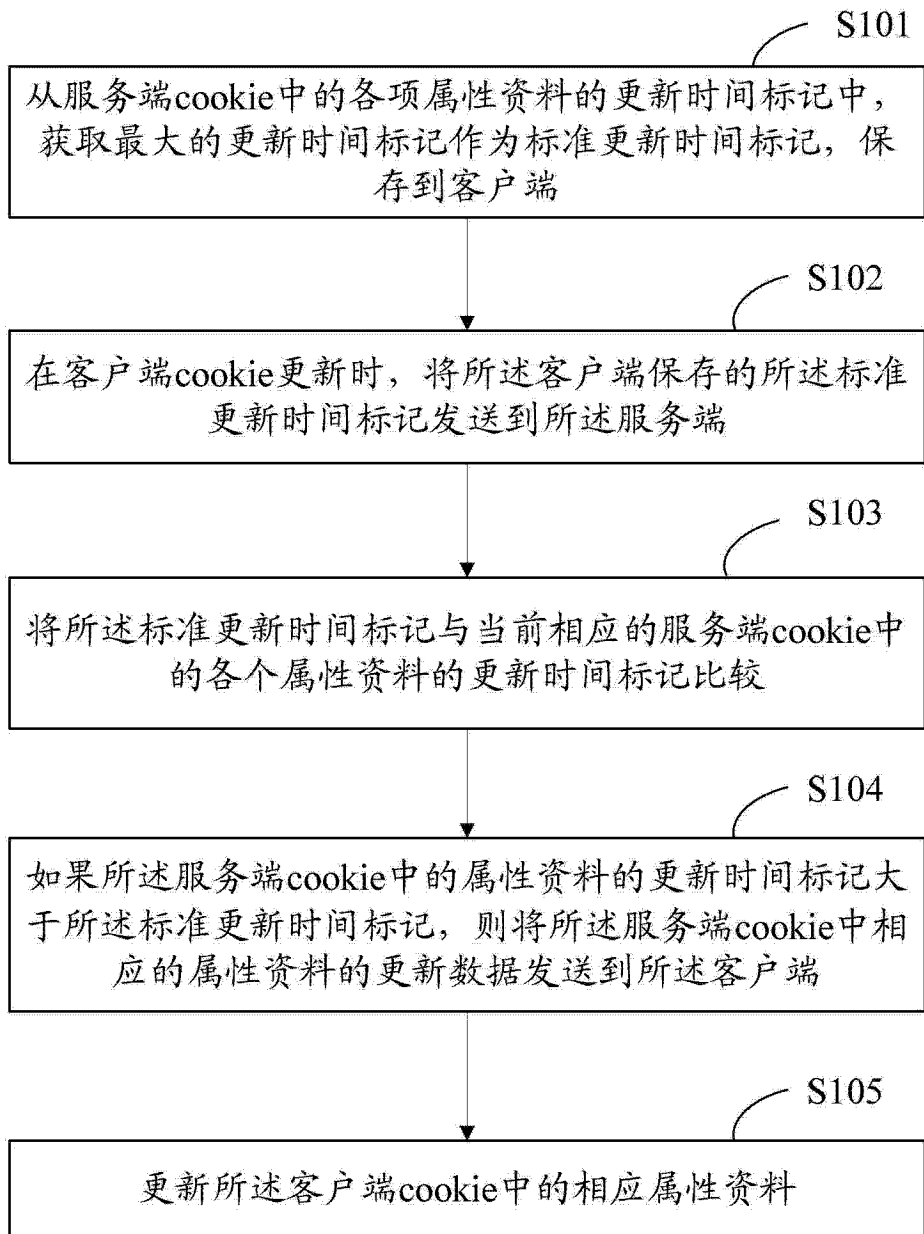


图 1

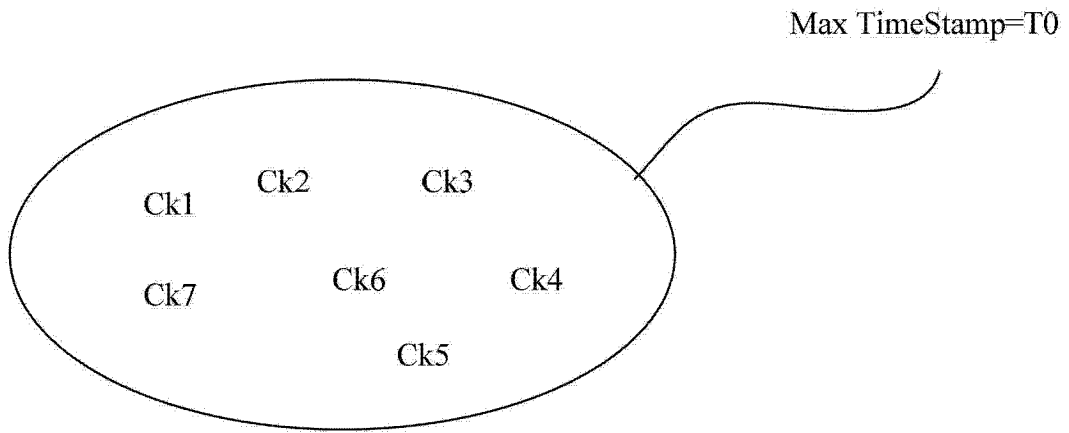


图 2

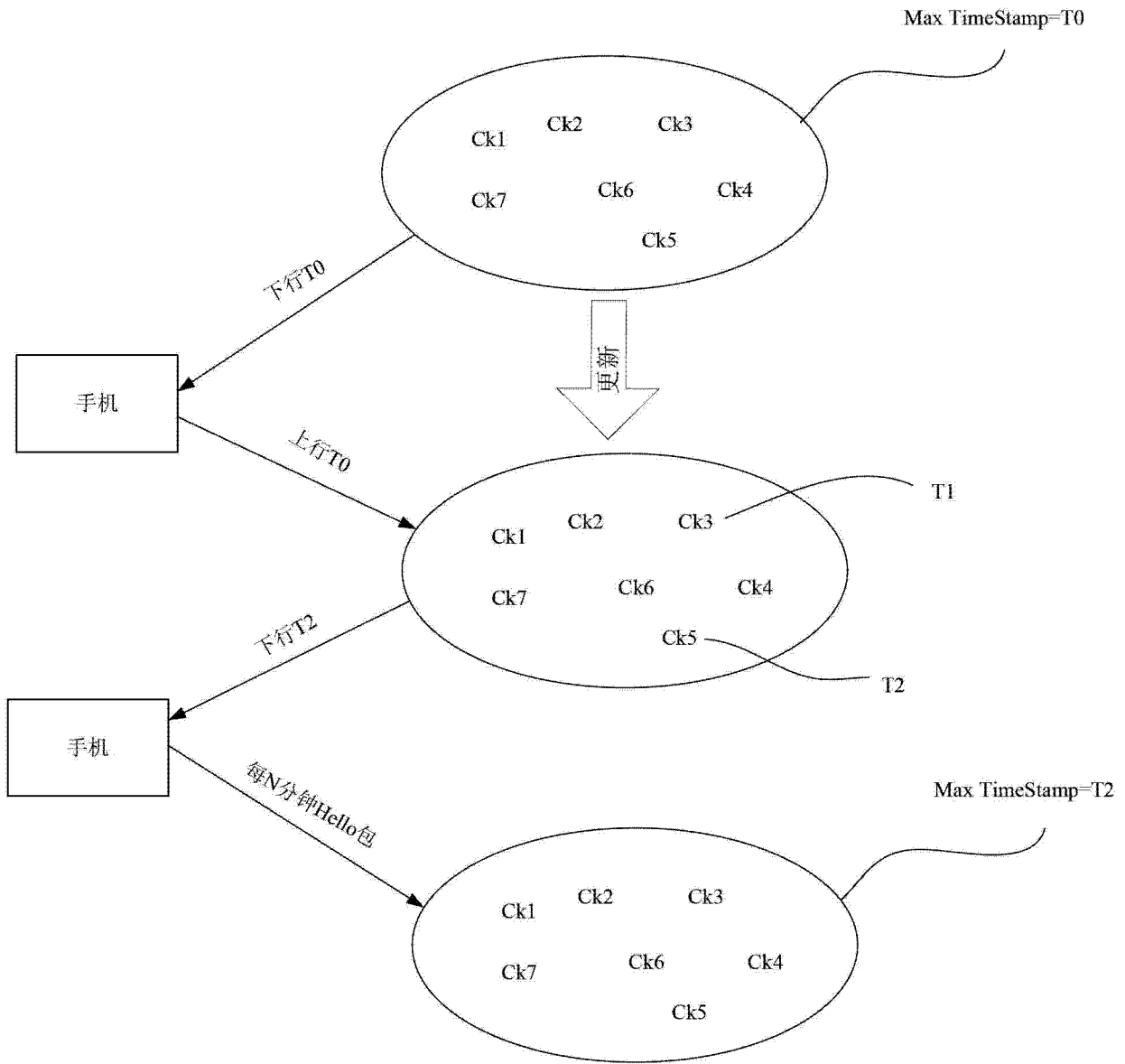


图 3

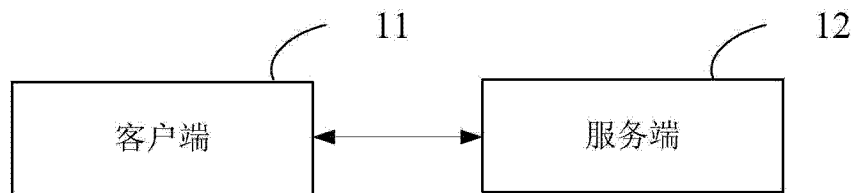


图 4