



[12] 发明专利申请公开说明书

[21] 申请号 200510115392.7

[43] 公开日 2006年10月25日

[11] 公开号 CN 1852309A

[22] 申请日 2005.11.16

[21] 申请号 200510115392.7

[71] 申请人 华为技术有限公司

地址 518129 广东省深圳市龙岗区坂田华为  
总部办公楼

[72] 发明人 喻宏斌 陈春荣 周英军 高 伟

[74] 专利代理机构 北京同达信恒知识产权代理有限公司  
代理人 黄志华

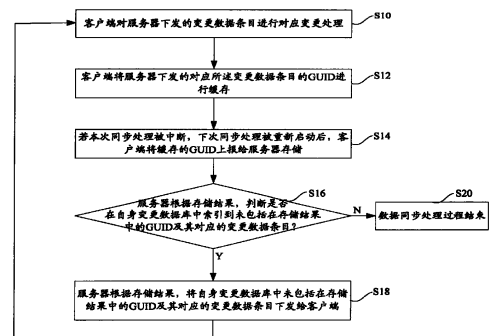
权利要求书 2 页 说明书 12 页 附图 4 页

[54] 发明名称

数据同步处理方法及其客户端

[57] 摘要

本发点分明公开了一种数据同步处理方法，包括客户端对服务器下发的变更数据条目进行对应变更处理；并将服务器下发的对应所述变更数据条目的全球唯一标识进行缓存；若本次同步处理被中断，下次同步处理被重新启动后，客户端将缓存的全球唯一标识上报给服务器存储；服务器根据所述存储结果，将自身变更数据库中未包括在所述存储结果中的全球唯一标识及其对应的变更数据条目下发给客户端，然后返回处理过程初始。相应的本发明还提出了一种客户端设备。本发明可以使数据同步处理过程在被中断的情况下也能正常完成数据的同步，达到了提高用户良好使用感受的目的。



1、一种数据同步处理方法，其特征在于，包括步骤：

A.客户端对服务器下发的变更数据条目进行对应变更处理；并

B.将服务器下发的对应所述变更数据条目的全球唯一标识进行缓存；

C.若本次同步处理被中断，下次同步处理被重启后，客户端将缓存的全球唯一标识上报给服务器存储；

D.服务器根据所述存储结果，将自身变更数据库中未包括在所述存储结果中的全球唯一标识及其对应的变更数据条目下发给客户端，然后执行A。

2、如权利要求1所述的方法，其特征在于，所述步骤C和D之间还包括步骤：

服务器将客户端上报的全球唯一标识成功存储后，向客户端反馈成功存储响应消息；

客户端接收到服务器反馈的成功存储响应消息后，删除自身缓存的全球唯一标识。

3、如权利要求1所述的方法，其特征在于，所述步骤D中若服务器根据所述存储结果，在自身变更数据库中索引不到未包括在所述存储结果中的全球唯一标识及其对应的变更数据条目时，则数据同步处理过程结束。

4、如权利要求1所述的方法，其特征在于，所述步骤B具体包括步骤：

客户端为服务器下发的变更数据条目分配对应的本地唯一标识；并

将所述本地唯一标识和服务器下发的对应所述变更数据条目的全球唯一标识之间建立映射关系；并

将所述建立的映射关系进行缓存，实现将服务器下发的全球唯一标识进行缓存。

5、如权利要求4所述的方法，其特征在于，所述步骤C中客户端通过将缓存的映射关系上报给服务器实现将缓存的全球唯一标识上报给服务器。

6、如1~5任一权利要求所述的方法，其特征在于，所述变更数据条目为

新增加数据条目。

7、一种客户端，其特征在于，包括：

变更处理单元，用于对服务器下发的变更数据条目进行对应变更处理；

标识缓存单元，用于将服务器下发的对应所述变更数据条目的全球唯一标识进行缓存；

标识上报单元，用于在本次同步处理被中断，下次同步处理被重启后，将所述标识缓存单元缓存的全球唯一标识上报给服务器存储。

8、如权利要求 7 所述的客户端，其特征在于，还包括标识删除单元，用于接收到服务器反馈的成功存储响应消息后，删除所述标识缓存单元中缓存的全球唯一标识。

9、如权利要求 7 所述的客户端，其特征在于，所述标识缓存单元具体包括：

标识分配子单元，用于为服务器下发的变更数据条目分配对应的本地唯一标识；

映射关系建立子单元，用于将所述标识分配子单元分配的本地唯一标识和服务器下发的对应所述变更数据条目的全球唯一标识之间建立映射关系；

映射标识缓存子单元，用于将所述映射关系建立子单元建立的映射关系进行缓存，实现将服务器下发的全球唯一标识进行缓存。

10、如权利要求 9 所述的客户端，其特征在于，所述标识上报单元通过将所述映射标识缓存子单元缓存的映射关系上报给服务器实现将缓存的全球唯一标识上报给服务器。

## 数据同步处理方法及其客户端

### 技术领域

本发明涉及同步标记语言（Syncml, Synchronization Markup Language）协议同步处理技术，尤其是涉及一种数据同步处理方法及其客户端。

### 背景技术

现代社会是信息社会，用户需要随时随地的进行信息的接收和发送，也需要随时随地的执行终端设备中的各种应用程序，而上述信息交互结果和执行应用程序结果都需要存储到终端设备中，也需要与其他通信设备中的相应数据保持一致。例如，在手机中、或在个人数字助理（PDA, Personal Digital Assistant）中都保存有通讯录，同时在办公室和家里的PC机中或者笔记本电脑中也都保存有相同的通讯录，用户总是希望这些通信设备中的信息能够保持一致，即当其中一个通信设备上的数据信息发生变化时，也可以操作其他通信设备中的相应数据信息进行对应改变，这就是不同通信设备之间的数据同步过程。

其中进行数据同步的两端设备可以是任何的网络设备，如掌上电脑、PDA、移动电话或桌面计算机等等；同时在任何一种网络上，同步操作的数据可以是任何一种网络数据。同时一个用户可以通过不同的通信设备访问并操作同样的数据信息集合。其中同步标记语言（SYNCML, Synchronization Markup Language）协议就是为了实现上述数据同步处理过程而提出的一个新的通用标准技术。

请参照图1，该图是现有技术中采用 Syncml 协议，客户端和服务端之间进行数据同步处理的处理过程示意图，其中一次 Syncml 同步处理过程需要客户端和服务端之间交互六个数据包来完成，即从 PKG1 到 PKG6，其主要处理过程如下：

1、无论一次 Syncml 同步处理过程由哪方发起，客户端都会首先向服务器发送同步初始化数据包 (PKG1)，其中在 PKG1 中主要包含有客户端鉴权信息、设备能力信息等；

2、服务器接收到 PKG1 后，会根据 PKG1 中包含的客户端鉴权信息对客户端进行鉴权认证，并验证客户端的账号信息，同时将验证结果信息和指定的同步类型信息通过同步初始化响应数据包 (PKG2) 反馈给客户端；

3、客户端通过分析接收到的 PKG2 中指定的同步类型，把本地数据库中自上次同步处理完成之后发生过变更的所有数据（包括增加、删除、修改等数据）封装到数据包 PKG3 中发送给服务器；

一般情况下，客户端在本地数据库的操作日志中通常通过日志 ID 来标识上次同步处理完成后日志所处的位置，通常采用“anchor”来标识该位置，这样下次进行同步处理时需要同步给服务器的数据就是 anchor 标识之后的日志中所记载的信息，客户端的参考日志信息格式如下表 1 所示（其中 LUID 代表本地唯一标识 local unique identity）：

表 1:

Anchor	LUID	执行的操作
1110	555	Add
1111	168	Replace
1112	123	delete
1113	556	Add

4、服务器接收到 PKG3 后，执行其中对应的各种操作指令将客户端变更的数据依次更新到对应数据库中；

服务器在将上述操作指令执行状态信息通过数据包 PKG4 返回给客户端的同时，也会将自身数据库中自上次同步处理完成之后发生过变更的所有数据（包括增加、删除、修改等数据）封装到数据包 PKG4 中返回给客户端；

5、客户端接收到 PKG4 后，执行其中对应的各种操作指令将服务器中变

更的数据依次更新到本地数据库中；

客户端在将上述操作指令执行状态信息通过数据包 PKG5 返回给服务器的同时,如果接收到的 PKG4 中包含向本地数据库中增加新数据条目的 add 指令,则客户端在成功向本地数据库添加新数据后,还需要生成相应的 ID 映射信息通过数据包 PKG5 通知给服务器;如图 2 所示,该图是客户端根据服务器下发的增加数据 add 指令生成 ID 映射信息表反馈给服务器的状态示意图,其中图中 GUID 代表全球唯一标识 global unique identity, LUID 代表本地唯一标识 local unique identity,对于同一个新增加的数据条目,服务器自身会产生一个在服务器端唯一标识的 GUID 值来对该数据条目进行标识,而客户端也会产生一个在客户端唯一标识的 LUID 值来对该数据条目进行标识;为了能够将服务器端和客户端双方要操作的同一数据条目对应起来,就要在服务器端维护一个 ID 映射信息表,如图 2 中,假设“Car”是服务器中新增加的数据条目,服务器在将该新增加的数据条目和对应的 GUID 值“1010101”下发到客户端时,客户端在将该新增加的数据条目“Car”增加到本地数据库后,还要为该新增加的数据条目分配一个 LUID 值“11”,然后客户端将为该新增加的数据条目分配的 LUID 值“11”和服务器为该新增加的数据条目分配的 GUID 值“1010101”建立映射关系后发送到服务器,同理,后续其他新增加数据条目同样进行上述处理后,服务器端就会得到一张 ID 映射信息表“Server Mapping Table”,同时要对该 ID 映射信息表进行维护。

6、服务器在将接收到的 PKG5 中包含的 ID 映射信息维护到自身映射表之后,会通过数据包 PKG6 向客户端反馈维护指令的执行状态;客户端接收到数据包 PKG6 后,如果其中状态码全部正常,则标志此次数据同步处理成功结束。

根据上述的 Syncml 同步处理过程,当用户使用无线通信终端上安装的 SYNCML 同步软件与有线网络上的服务器进行大数据量的数据同步处理时,经常会出现由于无线网络信号不好或者由于通信终端操作系统负载较大等原因,因而导致数据同步处理速度较慢、耗时较长等情况出现。此种情况下用户

很有可能会由于无法忍受长时间的等待，而在数据同步处理没有完成之前就中途中断了数据同步处理过程；此外，也有可能会由于其他原因，比如通信终端突然断电、网络信号逐渐消失或者有来电呼入等状况，而导致数据同步处理过程还没有完成就被中途中断掉了。

当由于上述原因而导致通信终端和服务器之间的数据同步处理过程在没有完成之前就产生中断时，后续当通信终端再次发起新的同步请求时，就需要重新对所有的变更数据进行同步，即中断之前已经同步处理完成的数据还需要在新的同步处理过程中重新进行同步处理，然而在重新执行同步处理过程中，有可能还会遇到上述所提及的网络条件问题和终端当前状态变化问题，这样重新开始的数据同步处理过程还有可能会由于上述的同样原因而再次被主动的或者被动的中断，如此循环下去，就会影响到用户的良好使用感受。

## 发明内容

本发明要解决的技术问题在于提出一种数据同步处理方法，以使数据同步处理过程在被中断的情况下也能正常完成数据的同步，达到提高用户良好使用感受的目的。

相应的，本发明还提出了一种客户端。

为解决上述问题，本发明提出的技术方案如下：

一种数据同步处理方法，包括步骤：

A.客户端对服务器下发的变更数据条目进行对应变更处理；并

B.将服务器下发的对应所述变更数据条目的全球唯一标识进行缓存；

C.若本次同步处理被中断，下次同步处理被重启后，客户端将缓存的全球唯一标识上报给服务器存储；

D.服务器根据所述存储结果，将自身变更数据库中未包括在所述存储结果中的全球唯一标识及其对应的变更数据条目下发给客户端，然后执行A。

较佳地，所述步骤C和D之间还包括步骤：

服务器将客户端上报的全球唯一标识成功存储后，向客户端反馈成功存储响应消息；

客户端接收到服务器反馈的成功存储响应消息后，删除自身缓存的全球唯一标识。

较佳地，所述步骤 D 中若服务器根据所述存储结果，在自身变更数据库中索引不到未包括在所述存储结果中的全球唯一标识及其对应的变更数据条目时，则数据同步处理过程结束。

较佳地，所述步骤 B 具体包括步骤：

客户端为服务器下发的变更数据条目分配对应的本地唯一标识；并

将所述本地唯一标识和服务器下发的对应所述变更数据条目的全球唯一标识之间建立映射关系；并

将所述建立的映射关系进行缓存，实现将服务器下发的全球唯一标识进行缓存。

较佳地，所述步骤 C 中客户端通过将缓存的映射关系上报给服务器实现将缓存的全球唯一标识上报给服务器。

较佳地，所述变更数据条目为新增加数据条目。

一种客户端，包括：

变更处理单元，用于对服务器下发的变更数据条目进行对应变更处理；

标识缓存单元，用于将服务器下发的对应所述变更数据条目的全球唯一标识进行缓存；

标识上报单元，用于在本次同步处理被中断，下次同步处理被重启后，将所述标识缓存单元缓存的全球唯一标识上报给服务器存储。

较佳地，所述客户端还包括标识删除单元，用于接收到服务器反馈的成功存储响应消息后，删除所述标识缓存单元中缓存的全球唯一标识。

较佳地，所述标识缓存单元具体包括：

标识分配子单元，用于为服务器下发的变更数据条目分配对应的本地唯一



标识;

映射关系建立子单元,用于将所述标识分配子单元分配的本地唯一标识和服务器下发的对应所述变更数据条目的全球唯一标识之间建立映射关系;

映射标识缓存子单元,用于将所述映射关系建立子单元建立的映射关系进行缓存,实现将服务器下发的全球唯一标识进行缓存。

其中所述标识上报单元通过将所述映射标识缓存子单元缓存的映射关系上报给服务器实现将缓存的全球唯一标识上报给服务器。

本发明能够达到的有益效果如下:

本发明方案通过客户端对服务器下发的变更数据条目进行对应变更处理;并将服务器下发的对应该变更数据条目的 GUID 进行缓存,这样若本次同步处理被中断,下次同步处理被重启后,客户端就可以将缓存的 GUID 上报给服务器存储,服务器再根据存储结果,将自身变更数据库中未包括在存储结果中的 GUID 及其对应的变更数据条目下发给客户端,从而减少了同步处理被中断重启后,服务器向客户端下发变更数据的数据量,节约了同步处理时间,使得数据同步处理过程在被中断的情况下也能正常完成数据的同步,达到了提高用户良好使用感受的目的。

## 附图说明

图 1 为现有技术中采用 Syncml 协议,客户端和服务器之间进行数据同步处理的处理过程示意图;

图 2 为客户端根据服务器下发的增加数据 add 指令生成 ID 映射信息表反馈给服务器的状态示意图;

图 3 为客户端和服务器之间进行变更数据部分交互处理的过程示意图;

图 4 为本发明数据同步处理方法的主要实现原理流程图;

图 5 为本发明客户端的主要组成结构框图;

图 6 为本发明客户端增加标识删除单元后的具体组成结构框图;

图7为本发明客户端中标识缓存单元的具体组成结构框图。

### 具体实施方式

本发明方案设计的主要目的是为了解决大数据量同步处理过程由于耗时较长而经常地被主动或被动地中断，从而导致数据同步处理无法完成的问题。

其中本发明方案的主要设计思想是基于下述原理得出的：

通过对上述现有技术中的 Syncml 同步处理过程进行分析可见，客户端和服务器之间交互的数据包 PKG3 和 PKG4，无论是从执行时间还是从传输流量上都占用了整个同步处理过程的很大比重，其具体原因如下：

数据包 PKG3 主要负责将客户端本地变更的数据条目发往服务器，而数据包 PKG4 主要负责将服务器端变更的数据条目下发给客户端，同时双方还会相互反馈执行对方发送的数据变更操作指令的执行结果。这样无论是在客户端中还是在服务器中，一旦发生变更的数据条目达到了上百甚至上千条（其中有可能仅一条变更数据条目就会占用几千字节空间，甚至几兆字节空间），由此可见客户端和服务器之间交互的这两个数据包 PKG3 和 PKG4 都会为整个数据同步过程带来很大的时间消耗及其流量消耗。

在上述大数据量同步处理过程中，无论对于数据包 PKG3 还是对于数据包 PKG4 中的数据而言，都可能分别被客户端和服务器分割成多个消息部分依次进行交互，每一次消息交互的过程都是一次发送请求、等待响应的过程，请具体参照图3所示，该图是客户端和服务器之间进行变更数据部分交互处理的过程示意图，这样在无线链路带宽不足、无线网络信号不好等情况下，无疑比较费时，因此需要尽量减少上述这种变更数据部分的交互次数。

同时对于客户端和服务器交互的每个变更数据消息中，对于新增加的数据条目的封装显然会占用大量的消息空间，尤其是对于容量较大的数据对象而言（如图片、视频等多媒体文件的新增数据条目）有时为了封装一条新增加的数据条目都需要分成多个消息来分别发送。而且考虑到相对于客户端而言，服务

器在一次同步处理过程中需要同步给客户端的数据量往往较大，为此本发明方案主要设计思想的关键点就放在如何减少服务器发送变更数据条目（尤其是新增数据条目）给客户端的数量上。

请参照图 4，该图是本发明数据同步处理方法的主要实现原理流程图，其主要实现过程如下：

步骤 S10，客户端根据服务器下发的变更数据条目进行对应变更处理；

步骤 S12、客户端将服务器下发的对应上述变更数据条目的全球唯一标识 GUID 进行缓存；

步骤 S14、若本次同步处理被中断，下次同步处理被重新启动后，客户端将上述缓存的 GUID 上报给服务器进行存储；

步骤 S16，服务器根据存储结果，判断是否能够在自身的变更数据库中索引到未包括在存储结果中的 GUID 及其对应的变更数据条目，如果是，执行下述步骤 S18，否则执行下述步骤 S20；

步骤 S18，服务器根据存储结果，将自身变更数据库中未包括在所述存储结果中的 GUID 及其对应的变更数据条目下发给客户端，然后返回继续执行步骤 S10；

步骤 S20，本次数据同步处理过程结束。

在上述处理过程中，若服务器将客户端上报的 GUID 成功存储后，会向客户端反馈成功存储响应消息；客户端接收到服务器反馈的成功存储响应消息后，可以选择删除自身缓存的 GUID。

其中上述步骤 S12 中客户端将服务器下发的对应变更数据条目的 GUID 进行缓存的处理过程可以通过如下方式实现：

客户端为服务器下发的变更数据条目分配对应的本地唯一标识 LUID；

并将分配的 LUID 和服务器下发的对述变更数据条目的 GUID 之间建立映射关系；

然后将建立的映射关系进行缓存，从而就实现了将服务器下发 GUID 进行

缓存的目的。

这样在上述步骤 S14 中若本次同步处理被中断，下次同步处理被重新启动后，客户端可以将上述缓存的 GUID 和 LUID 之间的映射关系上报给服务器，从而实现将缓存的 GUID 上报给服务器的目的。

其中上述方法实现过程对于变更数据条目为新增加数据条目的处理过程将显出较佳的技术效果。

综上所述，本发明数据同步处理方法的主要技术实现原理就是在每次进行数据同步处理时，客户端无论在接收数据包 PKG4 的过程中是否会出现异常中断现象，都会在完成每条变更数据条目（下述以新增加数据条目为例进行说明）的本地变更操作后，立即将自身分配给新增加数据条目的 LUID 和服务器下发的 GUID 进行对应映射，并将映射信息保存到文件中，这样在下次同步重启时就可以将此信息发送给服务器，从而避免服务器端将新增加的数据条目在多次同步处理过程中重复发送。

这样，在服务器端存在大量新增加的数据条目需要同步给客户端且在同步过程被多次中断的情况下，本发明方案将会产生显著的效果。客户端可以分多次来完成大数据量的数据同步，类似于网络下载中断点续传功能。

本发明方案的具体实施过程如下：

首先在客户端本地也建立一个和服务器端存储的同样格式的 ID 映射信息表（具体如上述现有技术图 2 中的 Server Mapping Table 所示），并将该建立的 ID 映射信息表进行置空；

每次进行数据同步处理时，客户端无论在接收服务器下发的数据包 PKG4 的过程中是否会出现异常中断现象，都会在完成每条新增加数据条目的本地添加操作后，在内存中生成一条 LUID（客户端完成本地添加操作后会产生该新增加数据条目的 LUID）和 GUID（服务器下发的数据包 PKG4 中有该新增加数据条目的 GUID）的对应信息，同时将此条信息存入到本地 ID 映射信息表。如果本次同步处理成功结束，则客户端会将上述生成的 ID 映射信息承载在数

据包 PKG5 中发送给服务器，成功后应清空本地 ID 映射信息表；若在客户端接收服务器下发的数据包 PKG4 过程中，或后续处理过程中被异常中断，则已成功添加到本地数据库中的新增加数据条目的 ID 映射信息就已经保存在客户端本地 ID 映射信息表中。

后续再次被启动同步处理时，在完成同步初始化后，客户端应首先通过数据包 PKG3 将本地 ID 映射信息表中的所有信息（慢同步情况除外，慢同步是一种特殊的同步，主要是在服务器状态出现异常时由服务器发起的。此时客户端会把本地所有的变更数据条目都发给服务器，服务器将这些数据条目和服务端现存的数据条目进行逐一详细的比较，然后将服务器没有包括的数据条目保存下来，同时将服务器端已存在而客户端还没有的数据条目返回给客户端，从而达成双方数据条目的一致）发送给服务器，然后再发送本地的变更数据条目；服务器在接收到客户端发来的数据包 PKG3 后，应该首先根据其中的 ID 映射信息完善自身的 ID 映射信息表，然后在封装数据包 PKG4 时就可以根据自身的 ID 映射信息表，避免把中断之前已经同步给客户端的新增加数据条目再次下发给客户端，这样就在一定程度上减少了数据包 PKG4 中传输的数据量，也减少了数据包 PKG4 被分割成多个消息与客户端依次进行交互的次数，最终减少了数据的同步处理时间。服务器可以在下发给客户端的数据包 PKG4 中承载向客户端反馈的存储 ID 映射信息执行成功响应消息，这样客户端在接收到服务器下发的数据包 PKG4 之后，可以立刻清除本地 ID 映射信息表中的对应信息。如果本次同步处理在此时又被异常中断掉，则此次新增加到客户端的数据条目又会产生新的 ID 映射信息被存入到客户端本地的 ID 映射信息表中，这样在后续启动新的同步处理时又可以减少一定的数据传输量。

下面将列举一个简单的例子来对本发明数据同步处理方法的具体实施情况进行详细的说明：

假设当前服务器端新增加了三条数据条目（如下表 2 所示），而客户端没有变更数据条目；

表 2:

GUID	姓名	电话
10093902	Adam	13600001001
10093903	Alice	13600301101
10093904	Mark	13400021001

此时发起数据同步处理，服务器会通过数据包 PKG4 把自身新增加的三条数据条目发送给客户端，如果客户端在对服务器下发的新增加数据条目进行对应增加处理过程中突然断电，若此时客户端只将第一条新增加的数据条目加入到了本地数据库中，并为其分配了 LUID 为 32，这样客户端本地 ID 映射信息表中就生成了一条 ID 映射信息[10093902, 32]，而另外两条新增加数据条目在客户端断电时还未来得及处理。后续客户端重启后，重新和服务器发起数据同步处理，此次同步处理过程中，客户端会首先通过数据包 PKG3 将自身存储的 ID 映射信息[10093902, 32]上报给服务器，服务器接收到该 ID 映射信息后，就会认为第 10093902 号数据条目已同步给客户端，这样在发送数据包 PKG4 时就会只封装后两条未同步给客户端处理的新增加数据条目，从而减少了本次数据同步处理的同步数据流量。

本发明相应于上述提出的数据同步处理方法，这里还相应提出了一种客户端设备，请参照图 5，该图是本发明客户端的主要组成结构框图，其主要包括变更处理单元 10、标识缓存单元 20 和标识上报单元 30，其中各个组成部分的主要作用如下：

变更处理单元 10，主要用于对服务器下发的变更数据条目进行对应的变更处理；

标识缓存单元 20，主要用于将服务器下发的对应所述变更数据条目的全球唯一标识 GUID 进行缓存；

标识上报单元 30，主要用于本次同步处理被中断，下次同步处理被重新启动后，将上述标识缓存单元 20 缓存的 GUID 上报给服务器端进行存储。

请参照图 6，该图是本发明客户端增加标识删除单元后的具体组成结构框图，其中新增加的标识删除单元 40，主要用于接收到服务器反馈的成功存储响应消息后，删除上述标识缓存单元 20 中缓存的 GUID。

请参照图 7，该图是本发明客户端中标识缓存单元的具体组成结构框图，其主要包括标识分配子单元 210、映射关系建立子单元 220 和映射标识缓存子单元 230，其中各个组成部分的主要作用如下：

标识分配子单元 210，主要用于为服务器下发的变更数据条目分配对应的本地唯一标识 LUID；

映射关系建立子单元 220，主要用于将上述标识分配子单元 210 分配的 LUID 和服务器下发的对应上述变更数据条目的 GUID 之间建立映射关系；

映射标识缓存子单元 230，主要用于将上述映射关系建立子单元 220 建立的映射关系进行缓存，以实现将服务器下发的 GUID 进行缓存。

相应的，上述标识上报单元 30 要通过将映射标识缓存子单元 230 中缓存的 LUID 和 GUID 之间的映射关系上报给服务器，从而实现将缓存的 GUID 上报给服务器的目的。

显然，本领域的技术人员可以对本发明进行各种改动和变型而不脱离本发明的精神和范围。这样，倘若本发明的这些修改和变型属于本发明权利要求及其等同技术的范围之内，则本发明也意图包含这些改动和变型在内。

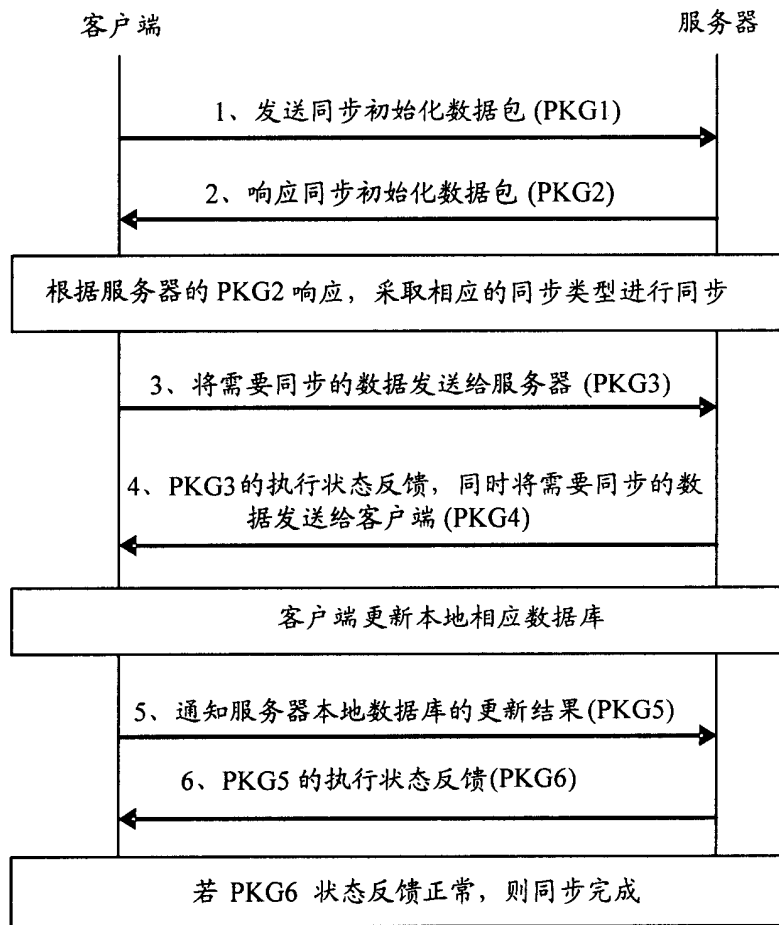


图 1



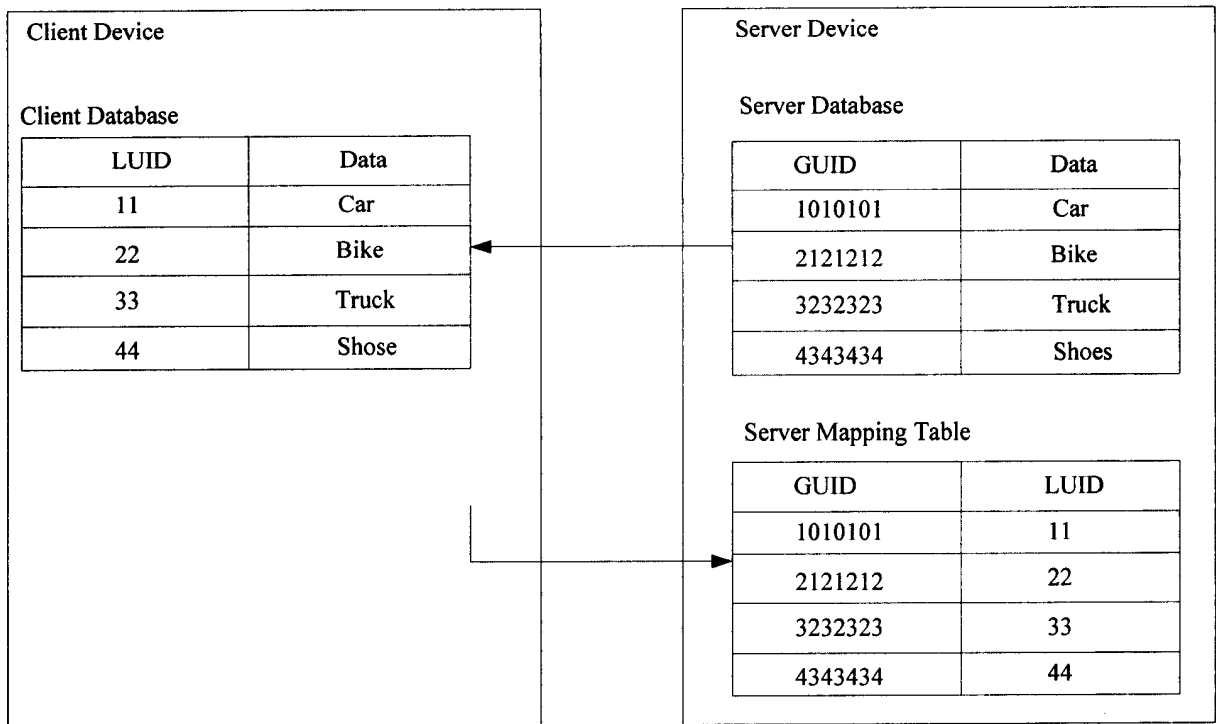


图 2

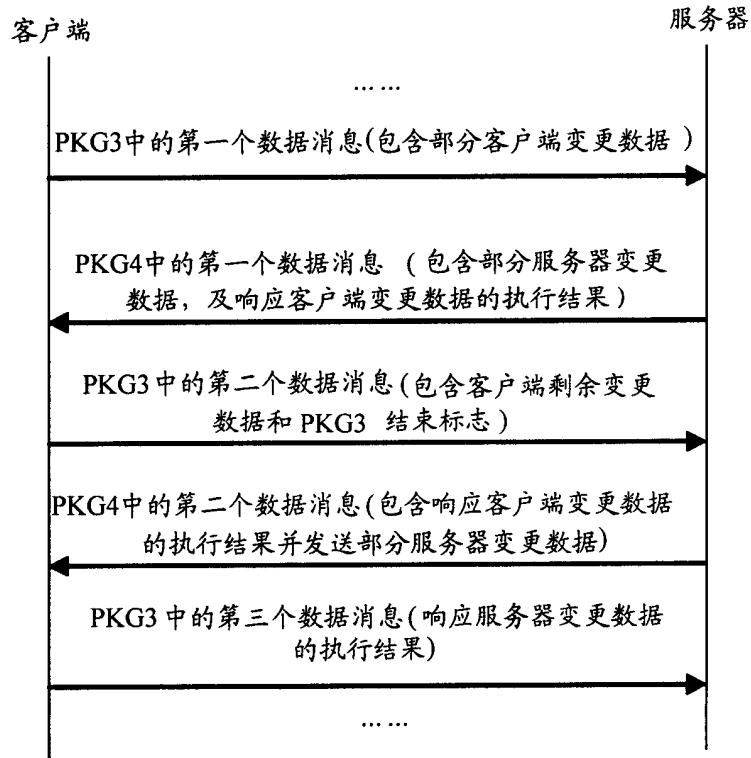


图 3

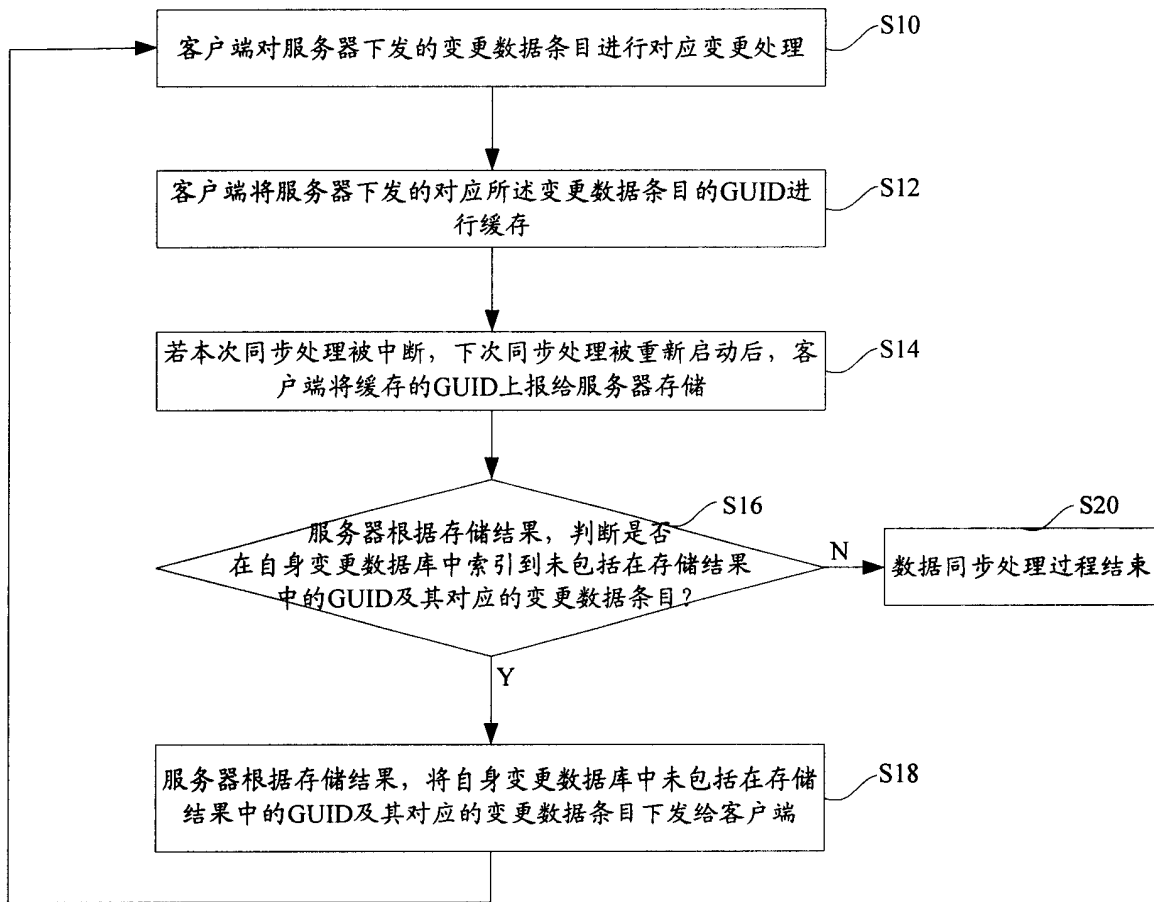


图 4

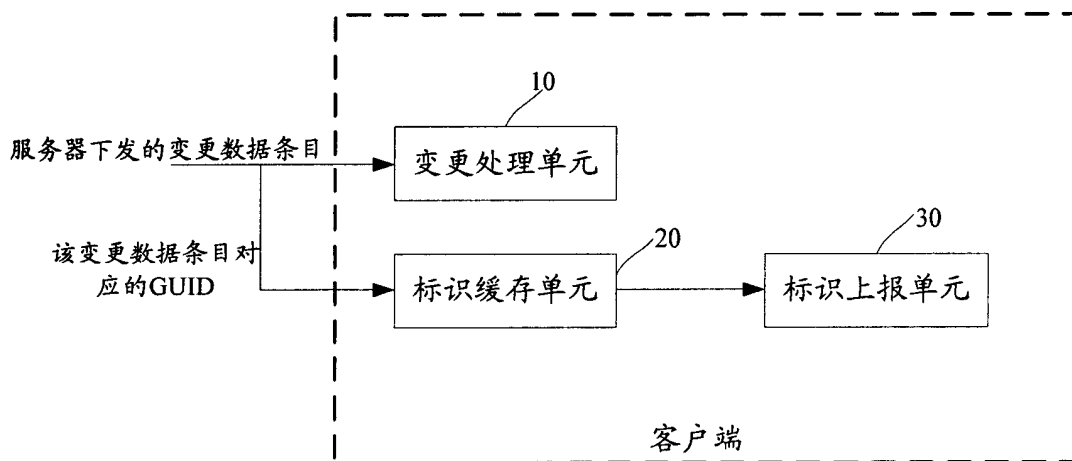


图 5

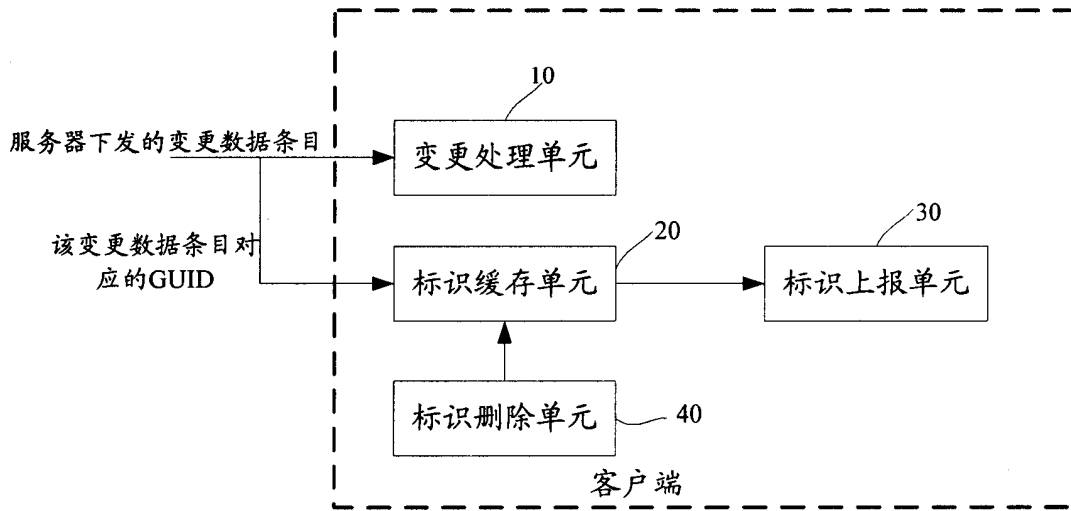


图 6

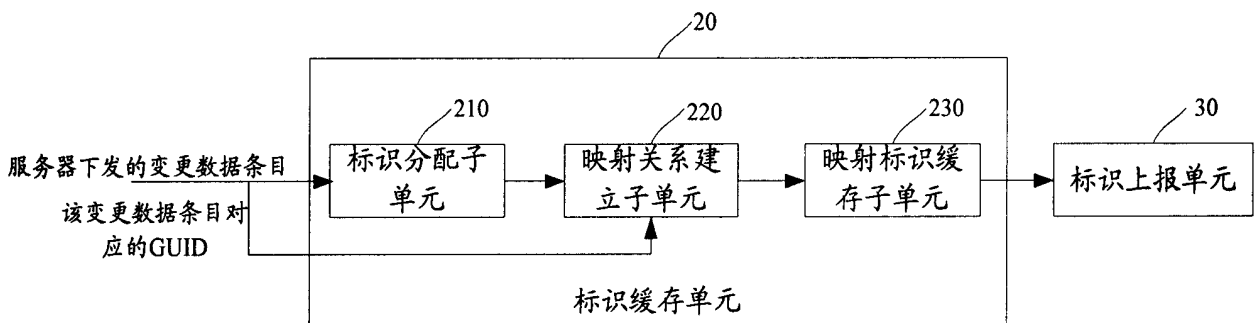


图 7