



(12) 发明专利申请

(10) 申请公布号 CN 101997783 A

(43) 申请公布日 2011.03.30

(21) 申请号 200910168505.8

(22) 申请日 2009.08.21

(71) 申请人 阿里巴巴集团控股有限公司
地址 英属开曼群岛大开曼岛资本大厦一座
四层 847 号邮箱

(72) 发明人 鲁一鹏

(74) 专利代理机构 北京集佳知识产权代理有限
公司 11227
代理人 逯长明 王宝筠

(51) Int. Cl.
H04L 12/58 (2006.01)

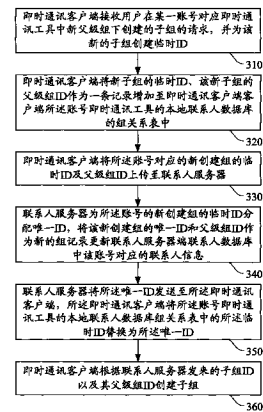
权利要求书 3 页 说明书 12 页 附图 8 页

(54) 发明名称

即时通讯工具联系人结构化分组的方法及服务器

(57) 摘要

即时通讯工具联系人结构化分组的方法及服务器。方法包括：IM 客户端接收用户在某一父级组下创建子组的请求，并创建临时 ID；IM 客户端将新子组的临时 ID、该新子组的父级组 ID 作为一条记录增加至组关系表中；IM 客户端将新创建组的临时 ID 及父级组 ID 上传至联系人服务器；联系人服务器为所述新创建组的临时 ID 分配唯一 ID，将该新创建组的唯一 ID 和父级组 ID 作为新的组记录更新联系人服务器端联系人数据库中该账号对应的联系人信息；联系人服务器将所述唯一 ID 发送至所述 IM 客户端并由 IM 客户端替换；IM 客户端根据联系人服务器发来的子组 ID 以及其父级组 ID 创建子组。利用本发明，可以实现对联系人多级分组的灵活管理。



1. 一种即时通讯工具联系人结构化分组的实现方法,其特征在于,包括:

即时通讯客户端接收用户在某一父级组下创建子组的请求,并为该新的子组创建临时 ID;

即时通讯客户端将新子组的临时 ID、该新子组的父级组 ID 作为一条记录增加至即时通讯客户端的本地联系人数据库的组关系表中;

即时通讯客户端将新创建组的临时 ID 及父级组 ID 上传至联系人服务器;

联系人服务器为所述新创建组的临时 ID 分配唯一 ID,将该新创建组的唯一 ID 和父级组 ID 作为新的组记录更新联系人服务器端联系人数据库中该账号对应的联系人信息;

联系人服务器将所述唯一 ID 发送至所述即时通讯客户端,所述即时通讯客户端将本地联系人数据库组关系表中的所述临时 ID 替换为所述唯一 ID;

即时通讯客户端根据联系人服务器发来的子组 ID 及其父级组 ID 创建子组。

2. 如权利要求 1 所述的方法,其特征在于,还包括:

联系人服务器更新所述联系人数据库中联系人信息的操作完成后,标记该操作的时间戳,并将该时间戳通知即时通讯客户端;

用户登录即时通讯客户端时,即时通讯客户端上传本地联系人最后一次操作的时间戳至联系人服务器;

联系人服务器对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳,并将比较结果返回至即时通讯客户端,即时通讯客户端根据返回的比较结果加载联系人组结构。

3. 如权利要求 2 所述的方法,其特征在于,所述即时通讯客户端根据返回的比较结果加载联系人组结构,包括:

如果即时通讯客户端接收的返回结果为相同,则即时通讯客户端直接加载本地联系人数据库组关系表中组结构的联系人;

如果即时通讯客户端接收的返回结果为即时通讯客户端上传的时间戳迟于服务器自身保存的时间戳,则即时通讯客户端直接加载本地联系人数据库组关系表中组结构的联系人,并发送联系人组结构更新消息至联系人服务器,联系人服务器接收到该更新消息后,将联系人服务器保存的联系人数据库组关系表中的组结构更新为与即时通讯客户端保存的一致,并标记该操作的时间戳,或将自身的时间戳设置为与即时通讯客户端的一致;

如果联系人服务器对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳的结果是服务器自身保存的时间戳迟于即时通讯客户端的时间戳,则联系人服务器将保存的联系人数据库组关系表中的组结构发送至即时通讯客户端,即时通讯客户端加载接收的联系人组结构,并将自身的联系人组结构更新为接收到的联系人组结构,且标记该操作的时间戳,或将自身的时间戳设置为与联系人服务器的一致。

4. 如权利要求 1 所述的方法,其特征在于,还包括:

用户在即时通讯客户端删除组时,即时通讯客户端先检查该组以及该组的所有子组是否包含有联系人,如果包含联系人,则提示用户是否结束本次删除操作,如果不包含联系人,或用户选择不结束本次删除操作,则即时通讯客户端在本地数据库中删除该组及其子组;

即时通讯客户端将该组及其子组的删除信息提交至联系人服务器;

联系人服务器端收到所述删除信息后,删除联系人服务器端联系人数据库中该组及其子组。

5. 一种即时通讯工具联系人结构化分组的实现方法,其特征在于,包括:

用户在即时通讯客户端选择并拖动即时通讯工具中的第一组时,即时通讯客户端记录第一组的 ID;

用户在即时通讯客户端拖动第一组至第二组释放时,即时通讯客户端记录获取第二组的 ID;

即时通讯客户端将第一组移动到第二组下,并将第一组的父级组 ID 设置为第二组的 ID,更新至本地联系人数据库的组关系表;

即时通讯客户端将第一组的父级组 ID 设置为第二组的 ID 的信息提交至联系人服务器;

联系人服务器根据收到的所述信息,修改联系人数据库中第一组的父级组 ID 设置为第二组的 ID,并在修改成功后通知即时通讯客户端。

6. 如权利要求 5 所述的方法,其特征在于,还包括:

联系人服务器修改所述联系人数据库中联系人信息的操作完成后,标记该操作的时间戳,并将该时间戳通知即时通讯客户端。

7. 如权利要求 6 所述的方法,其特征在于,还包括:

用户登录即时通讯客户端时,即时通讯客户端上传本地联系人最后一次操作的时间戳至联系人服务器;

联系人服务器对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳,并将比较结果返回至即时通讯客户端,即时通讯客户端根据返回的比较结果加载联系人组结构。

8. 如权利要求 7 所述的方法,其特征在于,所述即时通讯客户端根据返回的比较结果加载联系人组结构,包括:

如果即时通讯客户端接收的返回结果为相同,则即时通讯客户端直接加载本地联系人数据库组关系表中组结构的联系人;

如果即时通讯客户端接收的返回结果为即时通讯客户端上传的时间戳迟于服务器自身保存的时间戳,则即时通讯客户端直接加载本地联系人数据库组关系表中组结构的联系人,并发送联系人组结构更新消息至联系人服务器,联系人服务器接收到该更新消息后,将联系人服务器保存的联系人数据库组关系表中的组结构更新为与即时通讯客户端保存的一致,并标记该操作的时间戳,或将自身的时间戳设置为与即时通讯客户端的一致;

如果联系人服务器对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳的结果是服务器自身保存的时间戳迟于即时通讯客户端的时间戳,则联系人服务器将保存的联系人数据库组关系表中的组结构发送至即时通讯客户端,即时通讯客户端加载接收的联系人组结构,并将自身的联系人组结构更新为接收到的联系人组结构,且标记该操作的时间戳,或将自身的时间戳设置为与联系人服务器的一致。

9. 如权利要求 5 所述的方法,其特征在于,还包括:

用户在即时通讯客户端删除组时,即时通讯客户端先检查该组以及该组的所有子组是否包含联系人,如果包含联系人,则提示用户是否结束本次删除操作,如果不包含联系

人,或用户选择不结束本次删除操作,则即时通讯客户端在本地数据库中删除该组及其子组;

即时通讯客户端将该组及其子组的删除信息提交至联系人服务器;

联系人服务器端收到所述删除信息后,删除联系人服务器端联系人数据库中该组及其子组。

10. 一种实现即时通讯工具联系人结构化分组的服务器,其特征在于,包括:

接收单元,用于接收即时通讯客户端上传的新创建组的临时 ID 及父级组 ID;

分配单元,用于为所述新创建组的临时 ID 分配唯一 ID;

更新单元,用于将该新创建组的唯一 ID 和父级组 ID 作为新的组记录更新联系人数据库中的联系人信息;

发送单元,用于将所述唯一 ID 发送至所述即时通讯客户端。

11. 如权利要求 10 所述的服务器,其特征在于,还包括时间戳标记单元,用于在更新所述联系人数据库中联系人信息的操作完成后,标记该操作的时间戳,并将该时间戳通知即时通讯客户端。

12. 如权利要求 11 所述的服务器,其特征在于,还包括时间戳比较单元,比较结果返回单元,其中,

时间戳比较单元,用于接收即时通讯客户端上传的最后一次操作的时间戳;

比较结果返回单元,用于对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳,并将比较结果返回至即时通讯客户端。

13. 如权利要求 12 所述的服务器,其特征在于,还包括删除单元,用于在收到即时通讯客户端发来的删除组及其子组的信息后,删除联系人服务器端联系人数据库中该组及其子组。

14. 一种实现即时通讯工具联系人结构化分组的服务器,其特征在于,包括:

接收单元,用于接收即时通讯客户端发来的将第一组的父级组 ID 设置为第二组的 ID 的信息;

修改单元,用于根据接收单元接收的信息修改联系人数据库中第一组的父级组 ID 设置为第二组的 ID;

通知单元,用于在修改单元修改成功后通知即时通讯客户端。

15. 如权利要求 14 所述的服务器,其特征在于,还包括时间戳标记单元,用于在修改所述联系人数据库中联系人信息的操作完成后,标记该操作的时间戳,并将该时间戳通知即时通讯客户端。

16. 如权利要求 15 所述的服务器,其特征在于,还包括时间戳比较单元,比较结果返回单元,其中,

时间戳比较单元,用于接收即时通讯客户端上传的最后一次操作的时间戳;

比较结果返回单元,用于对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳,并将比较结果返回至即时通讯客户端。

17. 如权利要求 16 所述的服务器,其特征在于,还包括删除单元,用于在收到即时通讯客户端发来的删除组及其子组的信息后,删除联系人服务器端联系人数据库中该组及其子组。

即时通讯工具联系人结构化分组的实现方法及服务器

技术领域

[0001] 本发明涉及互联网技术领域,特别涉及一种即时通讯工具联系人结构化分组的实现方法及服务器。

背景技术

[0002] 即时通讯 (Instant Messaging, IM 或称实时通讯),是一种可以为使用者提供网络上类似于私人聊天室 (chatroom) 中进行的实时通讯的服务。

[0003] 现在有多种被人们广泛接收和使用的 IM 通讯工具。用户在使用 IM 工具过程中,可以添加联系人,以方便与固定好友等联系人的通讯。

[0004] 现有的众多 IM 工具中,添加的联系人只能提供一级分组管理。如图 1 中,示出了某即时通讯工具中联系人的分组方式。其中,所有的联系人可以被分为几个组中,例如被分至家人、朋友、同学、同事这些组中。这样,便于使用者分组管理不同的联系人。

[0005] 在对现有技术的研究和实践过程中,发明人发现现有技术中存在以下问题:

[0006] 现有即时通讯工具的联系人分组管理方式中只提供一级分组,较为单一,而对于具有多级关系、或者具有多级从属关系的联系人分组时,无法灵活的提供这样的分组管理,无法满足用户的需求。

发明内容

[0007] 本申请实施例的目的是提供一种即时通讯工具联系人结构化分组的实现方法及服务器,以实现对联人多级分组的灵活管理。

[0008] 为解决上述技术问题,本申请实施例提供一种即时通讯工具联系人结构化分组的实现方法及服务器是这样实现的:

[0009] 一种即时通讯工具联系人结构化分组的实现方法,包括:

[0010] 即时通讯客户端接收用户在某一父级组下创建子组的请求,并为该新的子组创建临时 ID;

[0011] 即时通讯客户端将新子组的临时 ID、该新子组的父级组 ID 作为一条记录增加至即时通讯客户端的本地联系人数据库的组关系表中;

[0012] 即时通讯客户端将新创建组的临时 ID 及父级组 ID 上传至联系人服务器;

[0013] 联系人服务器为所述新创建组的临时 ID 分配唯一 ID,将该新创建组的唯一 ID 和父级组 ID 作为新的组记录更新联系人服务器端联系人数据库中该账号对应的联系人信息;

[0014] 联系人服务器将所述唯一 ID 发送至所述即时通讯客户端,所述即时通讯客户端将本地联系人数据库组关系表中的所述临时 ID 替换为所述唯一 ID;

[0015] 即时通讯客户端根据联系人服务器发来的子组 ID 及其父级组 ID 创建子组。

[0016] 一种即时通讯工具联系人结构化分组的实现方法,包括:

[0017] 用户在即时通讯客户端选择并拖动即时通讯工具中的第一组时,即时通讯客户端

记录第一组的 ID；

[0018] 用户在即时通讯客户端拖动第一组至第二组释放时，即时通讯客户端记录获取第二组的 ID；

[0019] 即时通讯客户端将第一组移动到第二组下，并将第一组的父级组 ID 设置为第二组的 ID，更新至本地联系人数据库的组关系表；

[0020] 即时通讯客户端将第一组的父级组 ID 设置为第二组的 ID 的信息提交至联系人服务器；

[0021] 联系人服务器根据收到的所述信息，修改联系人数据库中第一组的父级组 ID 设置为第二组的 ID，并在修改成功后通知即时通讯客户端。

[0022] 一种实现即时通讯工具联系人结构化分组的服务器，包括：

[0023] 接收单元，用于接收即时通讯客户端上传的新创建组的临时 ID 及父级组 ID；

[0024] 分配单元，用于为所述新创建组的临时 ID 分配唯一 ID；

[0025] 更新单元，用于将该新创建组的唯一 ID 和父级组 ID 作为新的组记录更新联系人数据库中的联系人信息；

[0026] 发送单元，用于将所述唯一 ID 发送至所述即时通讯客户端。

[0027] 一种实现即时通讯工具联系人结构化分组的服务器，包括：

[0028] 接收单元，用于接收即时通讯客户端发来的将第一组的父级组 ID 设置为第二组的 ID 的信息；

[0029] 修改单元，用于根据接收单元接收的信息修改联系人数据库中第一组的父级组 ID 设置为第二组的 ID；

[0030] 通知单元，用于在修改单元修改成功后通知即时通讯客户端。

[0031] 由以上本申请实施例提供的技术方案可见，即时通讯客户端接收用户在某一账号对应即时通讯工具中新创建的子组，为该新的子组创建临时 ID，即时通讯客户端将新子组的临时 ID、该新子组的父级组 ID 作为一条记录增加至即时通讯客户端所述账号即时通讯工具的本地联系人数据库的组关系表中，即时通讯客户端将所述账号对应的新创建组的临时 ID 及父级组 ID 上传至联系人服务器，联系人服务器为所述账号的新创建组的临时 ID 分配唯一 ID，将该新创建组的唯一 ID 和父级组 ID 作为新的组记录更新联系人服务器端联系人数据库中该账号对应的联系人信息，联系人服务器将所述唯一 ID 发送至所述即时通讯客户端，所述即时通讯客户端将所述账号即时通讯工具的本地联系人数据库组关系表中的所述临时 ID 替换为所述唯一 ID，即时通讯客户端根据联系人服务器发来的子组 ID 及其父级组 ID 创建子组，这样，可以实现对联系人多级分组的灵活管理。

附图说明

[0032] 为了更清楚地说明本申请实施例或现有技术中的技术方案，下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本申请中记载的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动性的前提下，还可以根据这些附图获得其他的附图。

[0033] 图 1 为现有技术即时通讯工具中一级分组管理的示意图；

[0034] 图 2 为本申请涉及的网络实体及其连接关系示意图；

- [0035] 图 3 为本申请即时通讯工具联系人结构化分组的实现方法第一实施例的流程图；
- [0036] 图 4 为本申请即时通讯工具联系人结构化分组的实现方法第一实施例生成的树结构示意图；
- [0037] 图 5 为本申请即时通讯工具联系人结构化分组的更新方法实施例的流程图；
- [0038] 图 6 为本申请即时通讯工具联系人结构化分组的删除方法实施例的流程图；
- [0039] 图 7 为本申请即时通讯工具联系人结构化分组的实现方法第二实施例的流程图；
- [0040] 图 8 为本发明实现即时通讯工具联系人结构化分组的服务器第一实施例的框图；
- [0041] 图 9 为本发明实现即时通讯工具联系人结构化分组的服务器第一实施例的另一框图；
- [0042] 图 10 为本发明实现即时通讯工具联系人结构化分组的服务器第一实施例的另一框图；
- [0043] 图 11 为本发明实现即时通讯工具联系人结构化分组的服务器第一实施例的另一框图；
- [0044] 图 12 为本发明实现即时通讯工具联系人结构化分组的服务器第二实施例的框图；
- [0045] 图 13 为本发明实现即时通讯工具联系人结构化分组的服务器第二实施例的另一框图；
- [0046] 图 14 为本发明实现即时通讯工具联系人结构化分组的服务器第二实施例的另一框图；
- [0047] 图 15 为本发明实现即时通讯工具联系人结构化分组的服务器第二实施例的另一框图。

具体实施方式

- [0048] 本申请实施例提供一种即时通讯工具联系人结构化分组的实现方法及服务器。
- [0049] 为了使本技术领域的人员更好地理解本申请中的技术方案，下面将结合本申请实施例中的附图，对本申请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申请一部分实施例，而不是全部的实施例。基于本申请中的实施例，本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例，都应当属于本申请保护的范围。
- [0050] 本申请涉及的网络实体及其连接关系可以如图 2 所示，包括，即时通讯客户端，网络，联系人服务器。用户可以用自己的账号登录即时通讯客户端，即时通讯则通过网络与联系人服务器进行通信。而且，联系人服务器上一般会保存每一账号的联系人信息，则登陆某账号后的即时通讯工具，可以通过所述联系人服务器获取、同步该账号的联系人。
- [0051] 以下介绍本申请即时通讯工具联系人结构化分组的实现方法第一实施例，图 3 示出了该方法第一实施例的流程，如图 3 所示，包括：
- [0052] S310：即时通讯客户端接收用户在某一父级组下创建子组的请求，并为该新的子组创建临时 ID。
- [0053] 用户通过注册的账号登陆即时通讯客户端，选择需要创建子组的父级组，然后发起在该父级组下创建子组的请求，即时通讯客户端为该子组创建临时 ID。

[0054] 即时通讯客户端中,不同分组具有不同的 ID,用于在即时通讯内部标识组。这里,用户新建子组后,该新建的子组会被分配一与之对应的 ID。

[0055] S320 :即时通讯客户端将新子组临时 ID、新子组的父级组 ID 作为一条记录增加至即时通讯客户端的本地联系人数据库的组关系表中。

[0056] 父级组先于所述新建子组已存在,因此,父级组的 ID 是已存在的。即时通讯客户端的每个联系人分组都有其父级组。最上级的父级组可以是预先设置的,例如可以设置为一个树形结构表示的父子关系的根节点。

[0057] 即时通讯客户端为新建子组创建临时 ID 后,查询本地联系人数据库的组关系表中的父级组 ID,对应该父级组 ID 记录新建子组的临时 ID。

[0058] 以下给出一个表格,以表示在即时通讯客户端本地联系人数据库的组关系表中增加新子组 ID ;

[0059]

组 ID	组名称	父组 ID	...
01	集团		
02	分公司 A	01	
03	分公司 B	01	
04	A 财务部	02	
05	A 采购部	02	
06	A 销售部	02	
07	B 财务部	03	
08	B 采购部	03	
09	B 销售部	03	
100	A 研发部	02	

[0060] 表 1. 增加新子组临时 ID

[0061] 如上面表 1 的例子中,增加新的子组为 A 研发部,其临时 ID 100,其父级组 ID 02 被作为一条记录增加至该组关系表中。

[0062] S330 :即时通讯客户端将新创建组临时 ID 及父级组 ID 上传至联系人服务器。

[0063] S340 :联系人服务器为所述新创建组的临时 ID 分配唯一 ID,将该新创建组唯一 ID 和父级组 ID 作为新的组记录更新联系人服务器端联系人数据库中该账号对应的联系人信息。

[0064] 所述唯一 ID 在服务器数据库组关系表中具有唯一性。

[0065] 如上面表 1 中的例子,例如临时 ID 100 的 A 研发部由联系人服务器分配唯一 ID

10。

[0066] S350 :联系人服务器将所述唯一 ID 发送至所述即时通讯客户端,所述即时通讯客户端将本地联系人数据库组关系表中的所述临时 ID 替换为所述唯一 ID。

[0067] 如上面的表 1,原来临时 ID100 的 A 研发部在由联系人服务器分配唯一 ID10 后,即时通讯客户端本地联系人数据库的组关系表中可以如下所示 :

[0068]

组 ID	组名称	父组 ID	...
01	集团		
02	分公司 A	01	
03	分公司 B	01	
04	A 财务部	02	
05	A 采购部	02	
06	A 销售部	02	
07	B 财务部	03	
08	B 采购部	03	
09	B 销售部	03	
10	A 研发部	02	

[0069] 表 2. 增加新子组唯一 ID

[0070] S360 :即时通讯客户端根据联系人服务器发来的子组 ID 以及其父级组 ID 创建子组。

[0071] 即时通讯客户端本地联系人数据库的组关系表中增加一条组记录后,即时通讯客户端可以从本地数据库中获取该增加的组记录数据,根据子组 ID、子组名称及其父级组 ID 重新创建联系人列表结构,从而完成子组的创建。例如在当联系人列表重新刷新后,即时通讯客户端从本地数据库中获取该增加的组记录数据,根据子组 ID、子组名称及其父级组 ID 重新加载联系人列表结构,完成子组的创建。

[0072] 以上面的表 1 为例,即时通讯客户端从本地数据库中根据子组 ID、子组名称及其父级组 ID 重新加载联系人列表结构,具体的,可以如下述方式实现 :

[0073] S1 :即时通讯客户端在本地联系人数据库中对组列表做循环检索。

[0074] S2 :当检索到父级组 ID 为空的组时,获取该组的 ID(即 ID 01),并用该组 ID 创建第一层节点。

[0075] S3 :当检索到父级组 ID 为 01 的组时,利用这些父级组 ID 为 01 的组的 ID 创建第二层的节点组,使其从属于 ID 01 的组。该步骤即在 ID 01 组之下创建分别为 ID 02、ID 03

的组。

[0076] S4:当检索到父级组 ID 为 02 的组时,利用这些父级组 ID 为 02 的组的 ID 创建第二层的节点组,使其从属于 ID 02 的组。即创建 ID 04、ID 05、ID 06、ID 10 的组。该步骤即在 ID 02 组之下创建分别为 ID 04、ID 05、ID 06、ID 10 的组。

[0077] S5:当检索到父级组 ID 为 03 的组时,利用这些父级组 ID 为 02 的组的 ID 创建第二层的节点组,使其从属于 ID 02 的组。即创建 ID 04、ID 05、ID 06 的组。该步骤即在 ID 02 组之下创建分别为 ID 04、ID 05、ID 06 的组。

[0078] 这样,通过上述步骤生成树结构并完成了子组的创建,所生成的树结构如图 4 所示。

[0079] 本申请第一实施例还可以实现在不同的父级组下创建相同的子组,即该等父级组下的某一子组具有相同名称、组 ID 以及相同的联系人。举例来说,在父级组“分公司 A”下创建一子组“A 和 B 的供应商”,在父级组“分公司 B”下同样创建一子组“A 和 B 的供应商”。这样可以更加方便查找到分公司 A 和分公司 B 共同的供应商。其实现方法可通过在组关系表中增加两条记录,详见表 3。这样在父级组 ID 02 和 ID 03 下均创建了子组 ID 10。

[0080]

组 ID	组名称	父组 ID	...
01	集团		
02	分公司 A	01	
03	分公司 B	01	
04	A 财务部	02	
05	A 采购部	02	
06	A 销售部	02	
07	B 财务部	03	
08	B 采购部	03	
09	B 销售部	03	
10	A 和 B 的供应商	02	
10	A 和 B 的供应商	03	

[0081]

[0082] 表 3 在不同父级组中创建相同分组

[0083] 本申请第一实施例还可实现将同一个联系人分入多个不同的分组中,其实现方法则为在联系人列表中增加相应的记录,具体实例见表 4。

[0084]

编号	联系人 ID	联系人名称	所属组 ID	...
0001	R001	同学兼同事	01	
0002	R002	家人 1	02	
0003	R003	同学 2	01	
0004	R004	同事 1	03	
0005	R005	同事 2	03	
0006	R006	同事 3	03	
0007	R007	同事 4	03	
0008	R008	同事 5	03	
0009	R009	同学 3	01	
0010	R001	同学兼同事	03	

[0085] 表 4 将同一联系人分入不同的分组

[0086] 表 4 所示为将同一联系人 R001 分入组 ID 01 和 ID 03 下, 在创建联系人列表时只需创建编号为 0001 和 0010 的两条记录, 该两条记录中联系人 ID 均为 R001, 所属组 ID 分别为 01 和 03。

[0087] 上述第一实施例提供了新的即时通讯联系人结构化分组管理实现方法, 很好的解决了即时通讯客户端无法管理多级分组的技术问题, 并且针对目前客户端组操作的准确性和便捷性都给出了最优的解决方案, 提高了组操作的客户端相应速度, 降低了用户的使用成本。

[0088] 以下介绍上述第一实施例的一个具体例子, 该例子的场景如下: 用户为一个大卖家, 卖多种不同类型的商品, 主要经营商品为服装类: 有裤子、上衣等; 食品类: 有海鲜、果脯等。该卖家的进货和出货渠道都为网上。该用户日常需要通过 IM 客户端来管理联系人, 以方便联系进货商和即时通知买家新货。因此需要将进货商和买家进行按照不同类目分组管理。按照以上场景, 该例子可以设定为: A 表示进货商, A1 表示服装类进货商, A11 表示裤子进货商, A12 表示上衣进货商, A2 表示为食品类进货商, A21 表示海鲜进货商, A22 表示果脯进货商; B 表示买家分组, B1 表示服装类买家, B11 表示裤子买家, B12 表示上衣买家, B2 表示为食品类买家, B21 表示海鲜买家, B22 表示果脯买家。

[0089] 则该具体例子可以如下:

[0090] T1: 用户在 IM 的联系人列表中建立两个一级分组, 每个分组 ID 分别对应 A、B。其中, 根据用户的需求, 每个组可以有不同的组名, 例如 A 为“进货商”, B 为“买家”。

[0091] T2 :在 A 组（即进货商组）下创建子组 A1。A1 的组名例如可以为“服装”。

[0092] 该创建子组的过程可以如上面 S310 至 S350 所示：

[0093] 用户在一账号的即时通讯客户端中新创建的子组“服装”，客户端为该新的子组创建临时 ID，为 A1；

[0094] 客户端将新子组的临时 ID A1、该新子组的父级组 ID A 作为一条记录增加至客户端所述账号即时通讯客户端的本地联系人数据库的组关系表中；

[0095] 客户端将所述账号对应的新创建组的临时 ID A1 及父级组 ID A 上传至联系人服务器；

[0096] 联系人服务器为所述账号的新创建组的临时 ID 分配唯一 ID（例如为 CONTACT_CLASS_A1），将该新创建组的唯一 ID CONTACT_CLASS_A1 和父级组 ID A 作为新的组记录更新联系人服务器端联系人数据库中该账号对应的联系人信息；

[0097] 联系人服务器将所述唯一 ID CONTACT_CLASS_A1 发送至所述客户端，所述客户端将所述账号即时通讯客户端的本地联系人数据库组关系表中的所述临时 ID 替换为所述唯一 ID CONTACT_CLASS_A1。

[0098] T3 :在 A1 组（即服装组）下创建子组 A11。A11 的组名可以为“裤子”，即将为裤子进货商的联系人都添加到该 A1 组下。

[0099] 该创建子组的过程符合上面 S310 至 S350 所述，也与上述建立 A1 的过程类似，不再赘述，下同。

[0100] T4 :在 A1 组（即服装组）下创建子组 A12。A12 的组名可以为“上衣”，即将上衣进货商的联系人都添加到该 A1 组中。

[0101] T5 :在 A 组（即进货商组）下创建子组 A2。A2 的组名可以为“食品”。

[0102] T6 :在 A2 组（即食品组）下创建子组 A21。A21 的组名可以为“海鲜”，即将海鲜进货商的联系人都添加到该组中。

[0103] T7 :在 A2 组（即食品组）下创建子组 A22。A22 的组名可以为“果脯”，即将果脯进货商的联系人都添加到该组中。

[0104] T8 :在 B 组（即买家组）下创建子组 B1。B1 的组名可以为“服装”。

[0105] T9 :在 B1 组（即服装组）下创建子组 B11。B11 的组名可以为“裤子”，即将常来买裤子的买家联系人都添加到该组中。

[0106] T10 :在 B1 组（服装组）下创建子组 B12。B12 的组名可以为“上衣”，即将常来买上衣的买家联系人都添加到该组中。

[0107] T11 :在 B 组（即买家组）下组创建子组 B2。B2 的组名可以为“食品”。

[0108] T12 :在 B2 组（即食品组）下创建子组 B21。B21 的组名可以为“海鲜”，即将常来买海鲜的买家联系人都添加到该组中。

[0109] T13 :在 B2 组（即食品组）下创建子组 B22。组名可以为“果脯”，即将常来买果脯的买家联系人都添加到该组中。

[0110] 以上的分组操作便于用户对联系人进行管理，卖家如果缺货可以快速定位找到进货商；卖家如果有新货到了，可以快速定位到相关的买家。这样，便于用户管理及精准定位联系人。

[0111] 上述第一实施例中，还可以包括：

[0112] 联系人服务器更新所述联系人数据库中联系人信息的操作完成后,标记该操作的时间戳,并将该时间戳通知即时通讯客户端。

[0113] 这样,联系人服务器端和客户端上都保存有联系人信息修改等操作的时间戳。

[0114] 基于上述客户端和联系人服务器端分别存有的时间戳,后续不论用户在哪个客户端上登录同一账号的所述即时通讯客户端,都可以采用以下过程实现客户端上的联系人信息特别是其结构化分组与联系人服务器上的保持一致,具体包括:

[0115] S410:用户登录即时通讯客户端时,即时通讯客户端上传本地联系人最后一次操作的时间戳至联系人服务器。

[0116] S420:联系人服务器对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳,并将比较结果返回至即时通讯客户端,即时通讯客户端根据返回的比较结果加载联系人组结构。

[0117] 所述即时通讯客户端根据返回的比较结果加载联系人组结构,具体包括以下几种情况:

[0118] 如果即时通讯客户端接收的返回结果为相同,则即时通讯客户端直接加载本地联系人数据库组关系表中组结构的联系人;

[0119] 如果即时通讯客户端接收的返回结果为即时通讯客户端上传的时间戳迟于服务器自身保存的时间戳,则即时通讯客户端直接加载本地联系人数据库组关系表中组结构的联系人,并发送联系人组结构更新消息至联系人服务器,联系人服务器接收到该更新消息后,将联系人服务器保存的联系人数据库组关系表中的组结构更新为与即时通讯客户端保存的一致,并标记该操作的时间戳,或将自身的时间戳设置为与即时通讯客户端的一致;

[0120] 如果联系人服务器对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳的结果是服务器自身保存的时间戳迟于即时通讯客户端的时间戳,则联系人服务器将保存的联系人数据库组关系表中的组结构发送至即时通讯客户端,即时通讯客户端加载接收的联系人组结构,并将自身的联系人组结构更新为接收到的联系人组结构,且标记该操作的时间戳,或将自身的时间戳设置为与联系人服务器的一致。

[0121] 上述即时通讯联系人结构化分组的更新方法,灵活的实现了即时通讯客户端与联系人服务器端联系人结构化分组相一致的目的。

[0122] 以下介绍基于上述第一实施例实现删除分组的方法,如图6所示,包括:

[0123] S510:用户在即时通讯客户端删除组时,即时通讯客户端先检查该组以及该组的所有子组是否包含有联系人,如果包含联系人,则提示用户是否结束本次删除操作;如果不包含联系人,或用户选择不结束本次删除操作,则即时通讯客户端在本地数据库中删除该组及其子组。

[0124] S520:即时通讯客户端将该组及其子组的删除信息提交至联系人服务器。

[0125] S530:联系人服务器收到所述删除信息后,删除联系人服务器端联系人数据库中该组及其子组。

[0126] 通过上述过程,可以实现删除联系人列表中的分组,并保持即时通讯客户端中删除分组后的联系人列表与联系人服务器端的一致。

[0127] 以下介绍本申请即时通讯客户端联系人结构化分组的实现方法第二实施例,图7示出了该方法第二实施例的流程,如图7所示,包括:

[0128] S610 :用户在即时通讯客户端选择并拖动即时通讯客户端中的第一组时,即时通讯客户端记录第一组的 ID。

[0129] S620 :用户在即时通讯客户端拖动第一组至第二组释放时,即时通讯客户端记录获取第二组的 ID。

[0130] 用户鼠标拖动到第二组的位置时,利用鼠标释放的响应事件,调用系统函数获取到第二组的 ID。

[0131] S630 :即时通讯客户端将第一组移动到第二组下,并将第一组的父级组 ID 设置为第二组的 ID,更新至本地联系人数据库的组关系表。

[0132] S640 :即时通讯客户端将第一组的父级组 ID 设置为第二组的 ID 的信息提交至联系人服务器。

[0133] S650 :联系人服务器根据收到的所述信息,修改联系人数据库中第一组的父级组 ID 设置为第二组的 ID,并在修改成功后通知即时通讯客户端。

[0134] 类似的,上述第二实施例中,还可以包括:

[0135] 联系人服务器修改所述联系人数据库中联系人信息的操作完成后,标记该操作的时间戳,并将该时间戳通知即时通讯客户端。

[0136] 这样,联系人服务器端和即时通讯客户端上都保存有联系人信息修改等操作的时间戳。

[0137] 类似的,基于上述即时通讯客户端和联系人服务器端分别存有的时间戳,后续不论用户在哪个即时通讯客户端上登录同一账号的所述即时通讯客户端,都可以采用以下过程实现即时通讯客户端上的联系人信息特别是其结构化分组与联系人服务器上的保持一致,如图 5 所示,具体包括:

[0138] S410 :用户登录即时通讯客户端时,即时通讯客户端上传本地联系人最后一次操作的时间戳至联系人服务器。

[0139] S420 :联系人服务器对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳,并将比较结果返回至即时通讯客户端,即时通讯客户端根据返回的比较结果加载联系人组结构。

[0140] 所述即时通讯客户端根据返回的比较结果加载联系人组结构,具体包括以下几种情况:

[0141] 如果即时通讯客户端接收的返回结果为相同,则即时通讯客户端直接加载本地联系人数据库组关系表中组结构的联系人;

[0142] 如果即时通讯客户端接收的返回结果为即时通讯客户端上传的时间戳迟于服务器自身保存的时间戳,则即时通讯客户端直接加载本地联系人数据库组关系表中组结构的联系人,并发送联系人组结构更新消息至联系人服务器,联系人服务器接收到该更新消息后,将联系人服务器保存的联系人数据库组关系表中的组结构更新为与即时通讯客户端保存的一致,并标记该操作的时间戳,或将自身的时间戳设置为与即时通讯客户端的一致;

[0143] 如果联系人服务器对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳的结果是服务器自身保存的时间戳迟于即时通讯客户端的时间戳,则联系人服务器将保存的联系人数据库组关系表中的组结构发送至即时通讯客户端,即时通讯客户端加载接收的联系人组结构,并将自身的联系人组结构更新为接收到的联系人组结

构,且标记该操作的时间戳。或将自身的时间戳设置为与联系人服务器的一致。

[0144] 上述即时通讯联系人结构化分组的更新方法,灵活的实现了即时通讯客户端与联系人服务器端联系人结构化分组相一致的目的。

[0145] 以下介绍基于上述第一实施例实现删除分组的方法,如图 6 所示,包括:

[0146] S510:用户在即时通讯客户端删除一个组时,即时通讯客户端先检查该组以及该组的所有子组是否包含有联系人,

[0147] 如果包含联系人,则提示用户是否结束本次删除操作;

[0148] 如果不包含联系人,或用户选择不结束本次删除操作,则即时通讯客户端在本地数据库中删除该组及其子组。

[0149] S520:即时通讯客户端将该组及其子组的删除信息提交至联系人服务器。

[0150] S530:联系人服务器端收到所述删除信息后,删除联系人服务器端联系人数据库中该组及其子组。

[0151] 通过上述过程,可以实现删除联系人列表中的分组,并保持即时通讯客户端中删除分组后的联系人列表与联系人服务器端的一致。

[0152] 以下介绍本发明实现即时通讯客户端联系人结构化分组的服务器第一实施例,图 8 示出了该服务器实施例的框图,如图 8 所示,包括:

[0153] 接收单元 71,用于接收即时通讯客户端上传的新创建组的临时 ID 及父级组 ID;

[0154] 分配单元 72,用于为所述新创建组的临时 ID 分配唯一 ID;

[0155] 更新单元 73,用于将该新创建组的唯一 ID 和父级组 ID 作为新的组记录更新联系人数据库中的联系人信息;

[0156] 发送单元 74,用于将所述唯一 ID 发送至所述即时通讯客户端。

[0157] 优选地,所述服务器还可以如图 9 所示,还包括时间戳标记单元 75,用于在更新所述联系人数据库中联系人信息的操作完成后,标记该操作的时间戳,并将该时间戳通知即时通讯客户端。

[0158] 优选地,所述服务器还可以如图 10 所示,还包括时间戳比较单元 76,比较结果返回单元 77,其中,

[0159] 时间戳比较单元 76,用于接收即时通讯客户端上传的最后一次操作的时间戳;

[0160] 比较结果返回单元 77,用于对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳,并将比较结果返回至即时通讯客户端。

[0161] 优选地,所述服务器还可以如图 11 所示,还包括删除单元 78,用于在收到即时通讯客户端发来的删除组及其子组的信息后,删除联系人服务器端联系人数据库中该组及其子组。

[0162] 利用上述实施例实现即时通讯客户端联系人结构化分组的方法、更新方法、删除方法与前述图 3、4、5 所示的方法类似,在此不再赘述。

[0163] 以下介绍实现即时通讯客户端联系人结构化分组的服务器第二实施例,图 12 示出了该服务器第二实施例的框图,如图 12 所示,包括:

[0164] 接收单元 111,用于接收即时通讯客户端发来的将第一组的父级组 ID 设置为第二组的 ID 的信息;

[0165] 修改单元 112,用于根据接收单元接收的信息修改联系人数据库中第一组的父级

组 ID 设置为第二组的 ID；

[0166] 通知单元 113,用于在修改单元修改成功后通知即时通讯客户端。

[0167] 优选地,所述服务器还可以如图 13 所示,还包括时间戳标记单元 114,用于在修改所述联系人数据库中联系人信息的操作完成后,标记该操作的时间戳,并将该时间戳通知即时通讯客户端。

[0168] 优选地,所述服务器还可以如图 14 所示,还包括时间戳比较单元 115,比较结果返回单元 116,其中,

[0169] 时间戳比较单元 115,用于接收即时通讯客户端上传的最后一次操作的时间戳；

[0170] 比较结果返回单元 116,用于对比自身保存的联系人最后一次操作的时间戳与即时通讯客户端上传的时间戳,并将比较结果返回至即时通讯客户端。

[0171] 优选地,所述服务器还可以如图 15 所示,还包括删除单元 117,用于在收到即时通讯客户端发来的删除组及其子组的信息后,删除联系人服务器端联系人数据库中该组及其子组。

[0172] 利用上述实施例实现即时通讯客户端联系人结构化分组的方法、更新方法、删除方法与前述图 7、5、6 所示的方法类似,在此不再赘述。

[0173] 为了描述的方便,描述以上装置时以功能分为各种单元分别描述。当然,在实施本申请时可以把各单元的功能在同一个或多个软件和 / 或硬件中实现。

[0174] 通过以上的实施方式的描述可知,本领域的技术人员可以清楚地了解到本申请可借助软件加必需的通用硬件平台的方式来实现。基于这样的理解,本申请的技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来,该计算机软件产品可以存储在存储介质中,如 ROM/RAM、磁碟、光盘等,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)执行本申请各个实施例或者实施例的某些部分所述的方法。

[0175] 本说明书中的各个实施例均采用递进的方式描述,各个实施例之间相同相似的部分互相参见即可,每个实施例重点说明的都是与其他实施例的不同之处。尤其,对于系统实施例而言,由于其基本相似于方法实施例,所以描述的比较简单,相关之处参见方法实施例的部分说明即可。

[0176] 本申请可用于众多通用或专用的计算系统环境或配置中。例如:个人计算机、服务器计算机、手持设备或便携式设备、平板型设备、多处理器系统、基于微处理器的系统、置顶盒、可编程的消费电子设备、网络 PC、小型计算机、大型计算机、包括以上任何系统或设备的分布式计算环境等等。

[0177] 本申请可以在由计算机执行的计算机可执行指令的一般上下文中描述,例如程序模块。一般地,程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等等。也可以在分布式计算环境中实践本申请,在这些分布式计算环境中,通过通信网络而被连接的远程处理设备来执行任务。在分布式计算环境中,程序模块可以位于包括存储设备在内的本地和远程计算机存储介质中。

[0178] 虽然通过实施例描绘了本申请,本领域普通技术人员知道,本申请有许多变形和变化而不脱离本申请的精神,希望所附的权利要求包括这些变形和变化而不脱离本申请的精神。

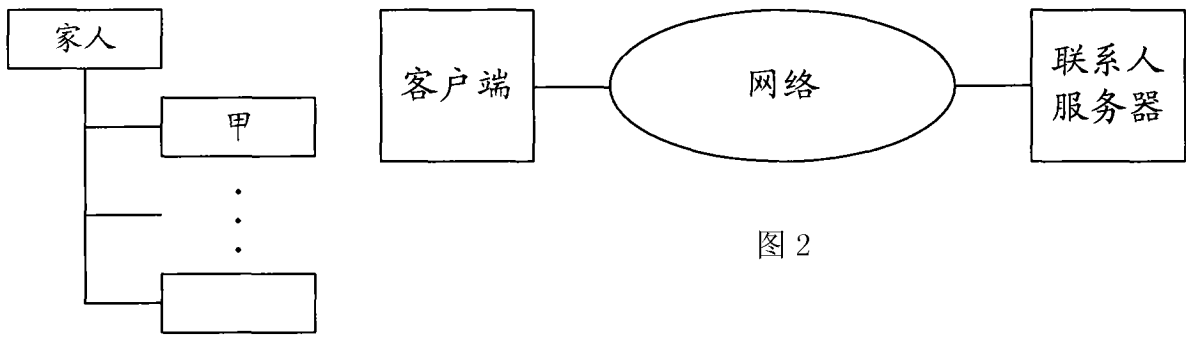


图 1



图 2

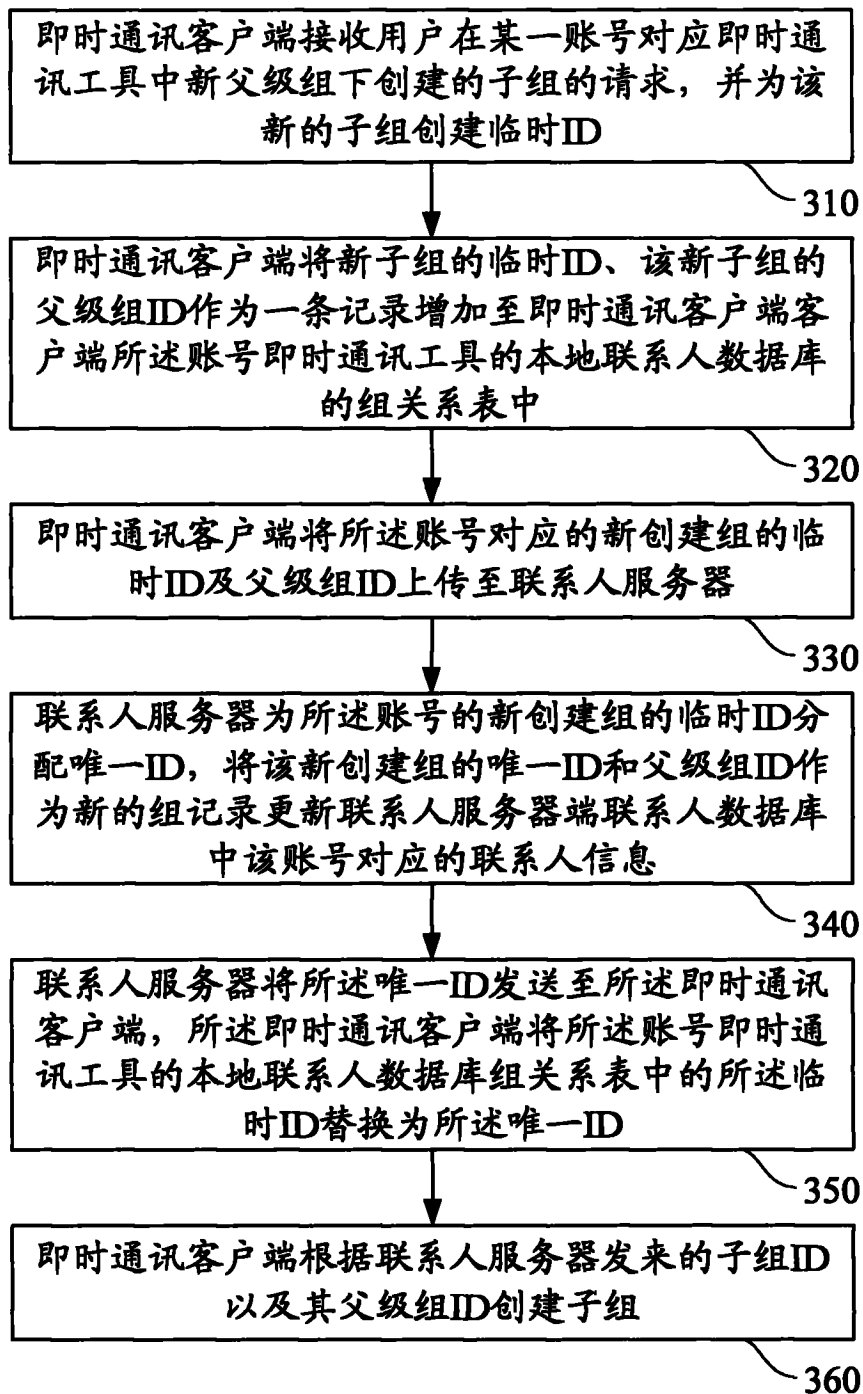


图 3

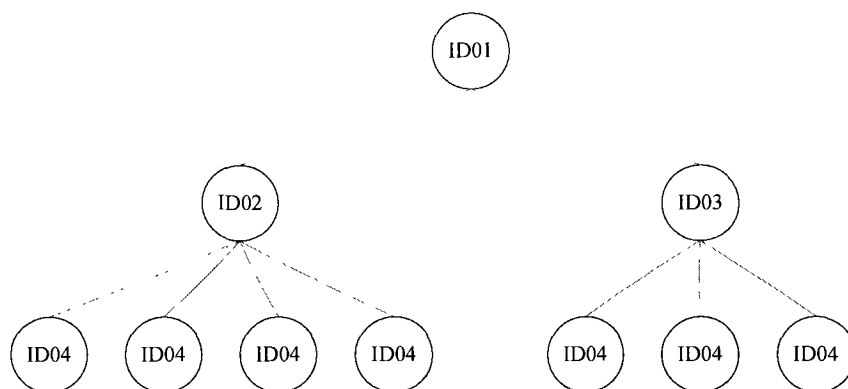


图 4

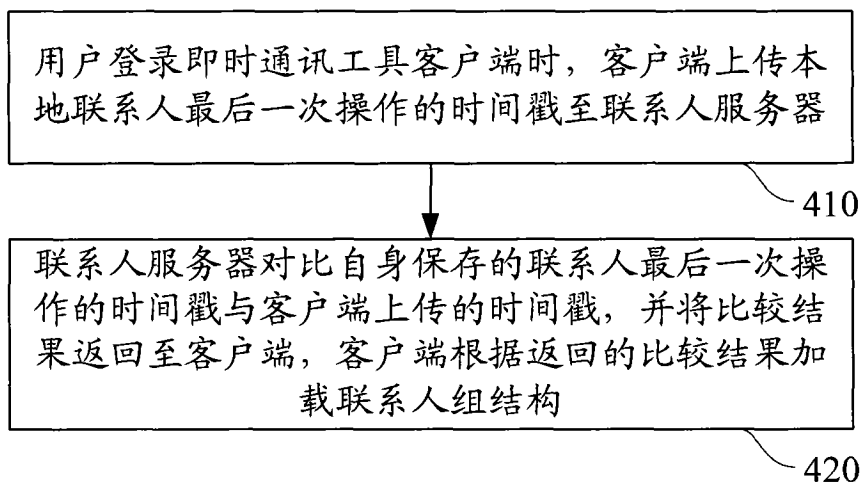


图 5

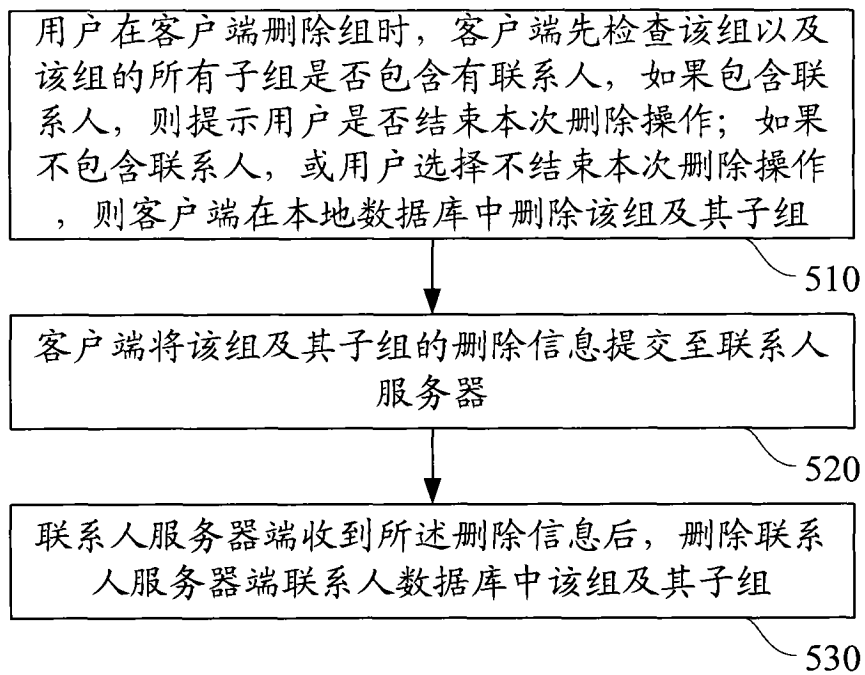


图 6

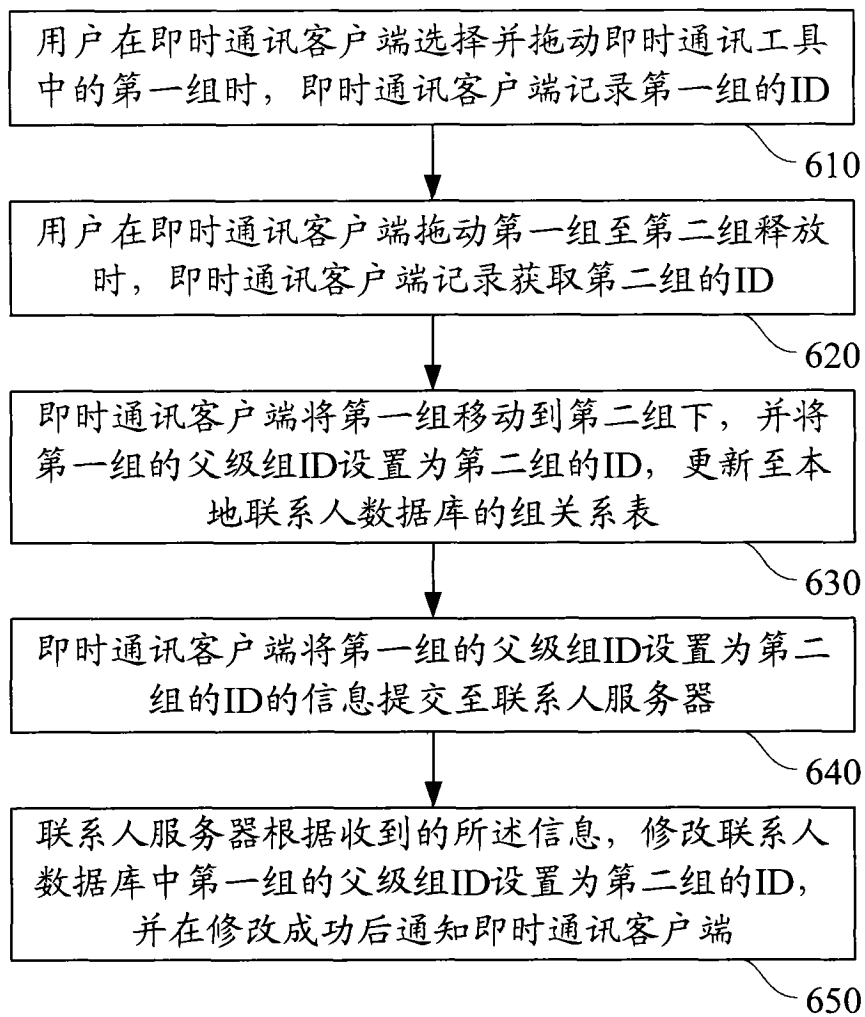


图7

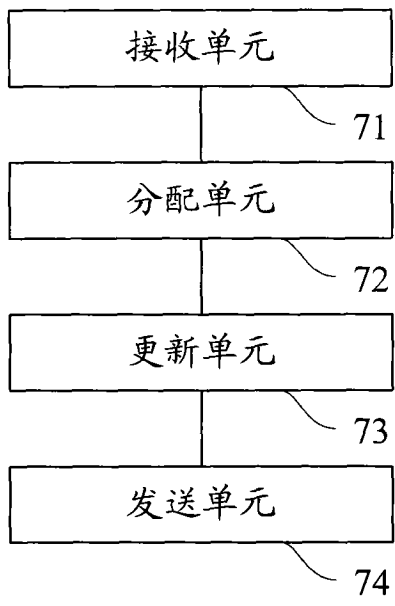


图 8

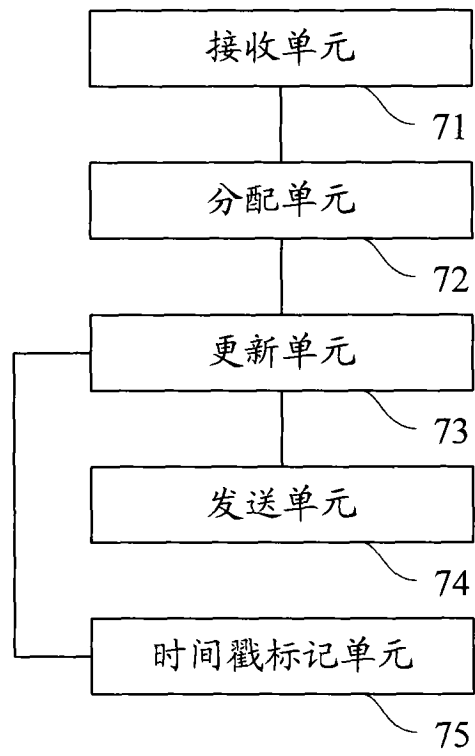


图 9

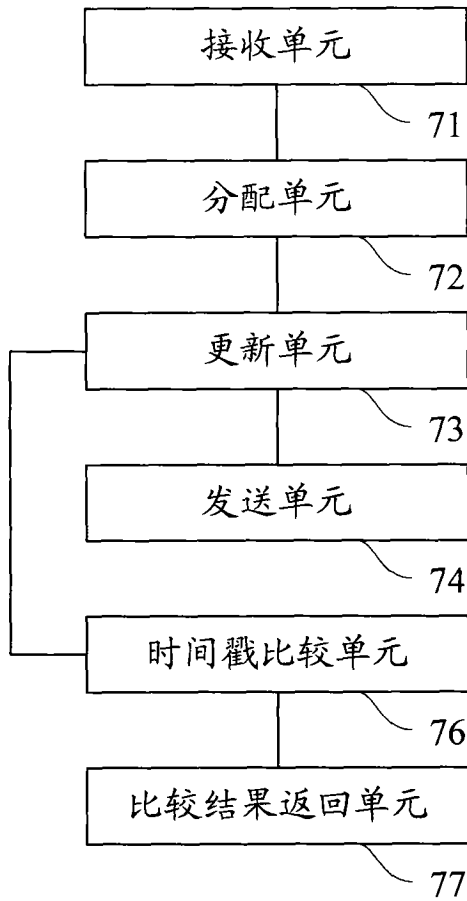


图10

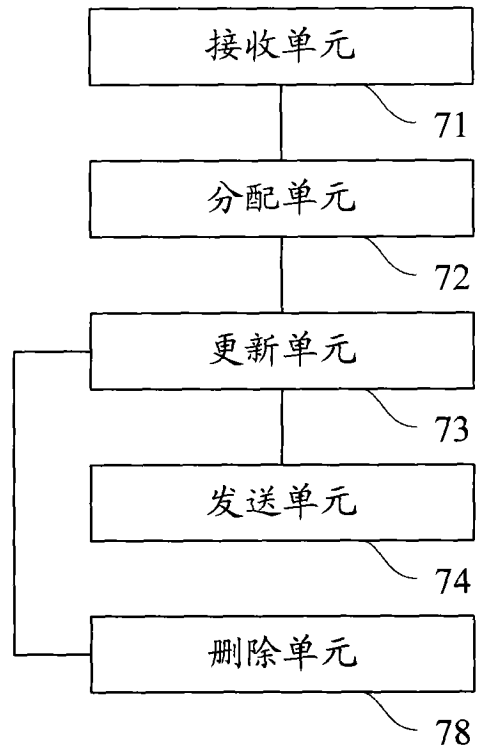


图11

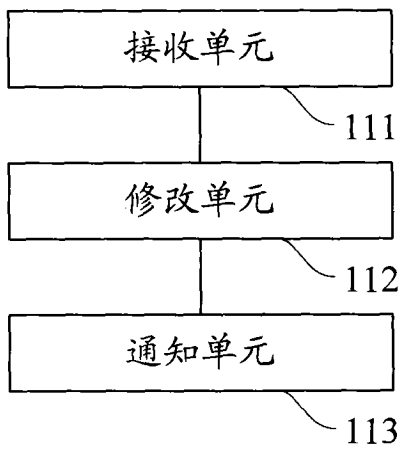


图12

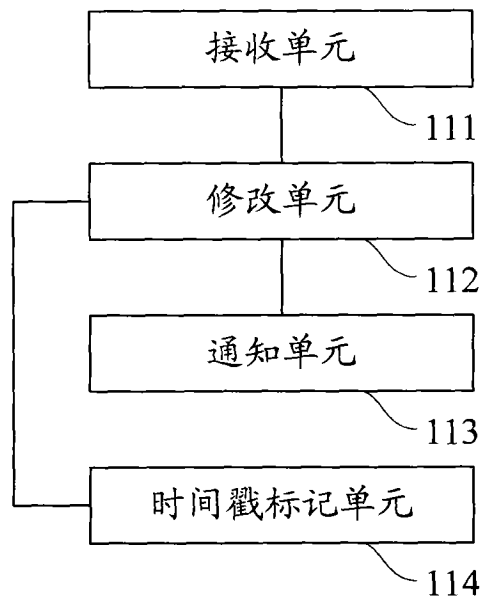


图13

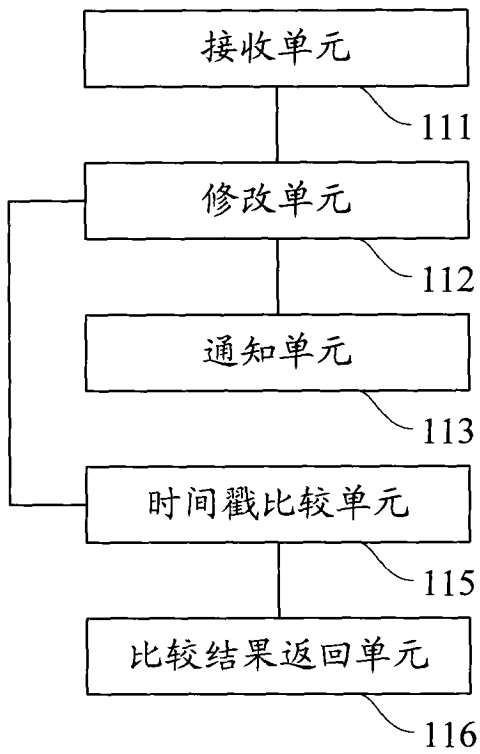


图 14

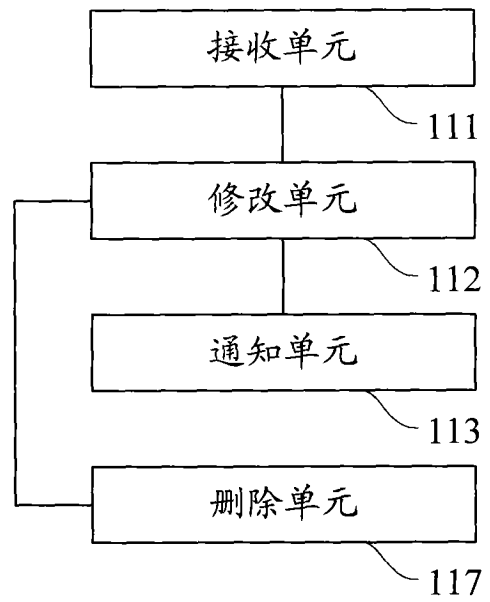


图 15