



(12) 发明专利

(10) 授权公告号 CN 1525373 B

(45) 授权公告日 2010.12.15

(21) 申请号 200410007601.1

(22) 申请日 2004.02.26

(30) 优先权数据

10/374,324 2003.02.26 US

(73) 专利权人 微软公司

地址 美国华盛顿州

(72) 发明人 D·拉森 P·D·瓦克斯曼

C·文卡特施

Y·(奥真)罗森菲尔德 A·纳林

(74) 专利代理机构 上海专利商标事务所有限公

司 31100

代理人 顾嘉运

(51) Int. Cl.

G06F 21/00 (2006.01)

(56) 对比文件

US 2002/0138572 A1, 2002.09.26, 全文.

US 2002/0129135 A1, 2002.09.12, 全文.

JP 特开 2002-24182 A, 2002.01.25, 全文.

审查员 宋朝

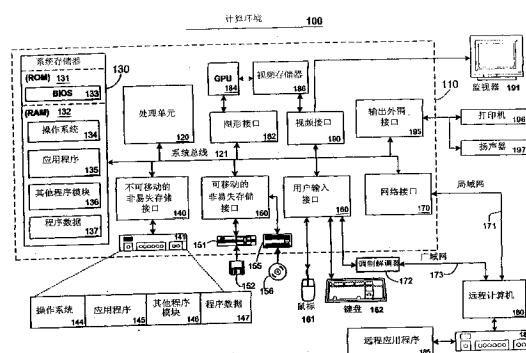
权利要求书 2 页 说明书 25 页 附图 19 页

(54) 发明名称

结合颁发内容的数字版权管理 (DRM) 许可证
审查高速缓存的用户组信息

(57) 摘要

一种计算机网络具有构成 X 等级的存储设备并且存储元素一组的信息。为了确定一个元素是否是任何目标组的一个成员,通过搜索存储设备以发现一个目标组的项目来确认各个目标组,并且,如果能发现,则将该目标组放置于一个确认目标箱 (V) 中。同样,通过搜索存储设备来发现一个元素的项目来确认元素,并且,如果能发现,则从该项目中确定列出元素的所有组都是其成员并且将该成员一组放置于一个搜索箱 (S) 中。如果 (S) 与 (V) 相交叉,则该元素是一个目标组的一个成员。如果不交叉,则将在 (S) 中的所有组都移动至对应于在等级 1 的存储设备的一个队列 (Q1),并且由一个迭代处理过程来搜索从该元素到任何目标组的一个成员关系路径。



1. 一种适用于确定在一个组织中的一个元素是否是在该组织所预先定义的一个或多个目标组中的一个成员的方法,该组织包含了一个计算机网络且该网络包括多个构成 X 个等级的存储设备并存储着元素 - 组的信息,存储设备包括在较高和较昂贵成本等级 X 上的目录和至少一个高速缓存存储器,各个高速缓存存储器是在较低和 X 之下较便宜成本等级上并包括至少一部分在目录中的元素 - 组信息,等级 1 至 X 对应于最低成本至最高成本,该方法包括:

通过搜索存储各等级的设备以发现适用于目标组的一个项目来确认各个目标组,并且,如果发现目标组,就将该目标组放置于一个确认的目标箱 V 中;其中,确认各个目标组包括:将各个目标组放置于一个目标箱 T 中;以及,对从 1 至 X 的各个等级,在该等级的存储设备中搜索对应于所述目标箱 T 中各个目标组的一个项目,并且如果该项目位于从所述目标箱 T 去除的目标组,则将该目标组放置于所述确认的目标箱 V 中;

通过搜索各等级的存储设备以发现适用于元素的一个项目来确认元素,并且,如果发现适用于该元素的项目,就从该项目中确定列出该元素是其成员的所有组并且将这类组的成员放置于搜索箱 S 中;

第一确定在所述搜索箱 S 中的任一组是否与在所述确认的目标箱 V 中的任何组相匹配;

如果相匹配,则肯定该元素就是一个目标组的一个成员;以及,

如果不相匹配,则通过将在所述搜索箱 S 中的所有的组都移动至一个对应于在等级 1 的存储设备的队列 Q1 中,并且执行迭代处理以搜索从该元素到任何目标组的成员关系的路径;

该迭代处理过程包括:对从 1 至 X 的各个等级 x:

第二确定对应于该等级的一个队列 Qx 是否是空的;

如果是空的,则使等级加 1 并且如果 $x < X$ 则返回到第二确定步骤,以及如果 $x = X$ 则可以肯定该元素不是任何目标组的一个成员;

如果不是空的,则从所述队列 Qx 去除该组并且确定该去除的组是否已经在丢弃箱 D 中;

如果已在所述丢弃箱 D 中,则返回第二确定步骤;

如果不在所述丢弃箱 D 中,则在对应于该等级的存储设备中搜索一个对应于去除组的项目;

如果在搜索中没有发现该项目,则如果 $x < X$,就将被去除的组移动至队列 Q(x+1),以及如果 $x = X$ 则将被去除的组移动至所述丢弃箱 D,并返回至第二确定步骤;

如果在搜索中发现了该项目,则将由所发现项目确定的、且所去除的组是其一个成员的每一新发现的组放置于所述搜索箱 S 中,所去除的组移动至所述丢弃箱 D,并返回至第一确定步骤,

借此,该迭代处理过程形成了所有从元素到任何所确认目标组的可能路径,在完成了在所有较低等级存储设备中的所有可能的较低成本的操作之前,不会在一个较高等级的存储设备上发生较昂贵成本的操作。

2. 如权利要求 1 所述方法,其特征在于,还包括:如果发现了对应于所述目标箱 T 中各个目标组的项目,就采用一个相对应的项目来装入任何较低等级的存储设备。

3. 如权利要求 1 所述方法,其特征在于,还包括:在搜索了适用于各个目标组的各等级的存储设备之后,确定所述确认的目标箱 V 是否是空的,并且如果是,则可以肯定没有发现有效的目标组。

4. 如权利要求 1 所述方法,其特征在于,确认该元素包括:对从 1 至 X 的各等级,在该等级的存储设备中搜索对应于该元素的一个项目,直至发现这一项目。

5. 如权利要求 4 所述方法,其特征在于,还包括:如果发现了一个对应于该元素的项目,则采用一个相对应的项目来装入任何较低等级的存储设备。

6. 如权利要求 1 所述方法,其特征在于,还包括:在搜索了适用于该元素的各等级的存储设备之后,确定是否发现一个适用于该元素的项目,并且如果没有,则可肯定没有发现有效的元素。

7. 如权利要求 1 所述方法,其特征在于, $X = 3$,其中,一个本地高速缓存存储器在等级 1,一个远程高速缓存存储器在等级 2,以及目录在等级 3。

8. 如权利要求 1 所述方法,其特征在于,该方法适用于确定一个组织的一个用户是在该组织内预先所限定的一个或多个目标组的一个成员,该组织包含了一个计算机网络且该网络包括多个构成 X 个等级的存储设备并存储着元素 - 组的信息。

结合颁发内容的数字版权管理 (DRM) 许可证审查高速缓存 的用户组信息

[0001] 交叉引用的相关申请

[0002] 下列美国专利申请所披露的主题内容与本发明的主题内容相关,因此本文引用其全部内容包括于此供参考:

[0003] 在 2002 年 6 月 28 日申请,专利代理事务所号码为 MSFT—1330,美国专利申请序号 No. 10/185,527,题为“获得适用于数字内容的签署权限标签 (SRL) 和获得在数字版权管理系统中与基于 SRL 的内容有关的数字许可证”;

[0004] 在 2002 年 6 月 28 日申请,专利代理事务所号码为 MSFT—1333,美国专利申请序号 No. 10/185,278,题为“使用权限模板来获得适用于在数字版权管理系统中数字内容的签署权限标签 (SRL)”;

[0005] 在 2002 年 6 月 28 日申请,专利代理事务所号码为 MSFT—1343,美国专利申请序号 No. 10/185,511,题为“适用于发行数字内容和服务的使用许可证的系统和方法”;

[0006] 在 _____ 申请,专利代理事务所号码为 MSFT—1498,的美国专利申请序号 No. _____,题为“发行在依照数字版权管理 (DRM) 系统的组织中的数字内容”;

[0007] 在 _____ 申请,专利代理事务所号码为 MSFT—1569,美国专利申请序号 No. _____,题为“发行在依照数字版权管理 (DRM) 系统的组织中的数字内容”;

[0008] 在 _____ 与本申请共同申请,专利代理事务所号码为 MSFT—1585,美国专利申请序号 No. _____,题为“发行适用于基于交叉树结构目录信息的内容的数字版权管理 (DRM) 许可证”。

技术领域

[0009] 本发明涉及数字版权管理 (DRM) 系统。尤其是,本发明涉及在应用 DRM 系统来发行一个组织 (例如,一个办公室或公司或类似单位等等) 中的数字内容,从而可以根据相关的使用和许可证项目来限制该内容在组织中的复制和使用。更具体的说,本发明涉及适用于至少是部分根据高速缓存组的信息来颁发相关数字内容的数字许可证。

[0010] 技术背景

[0011] 数字版权的管理和执行与数字内容相结合是非常理想的,数字内容可以包括数字音频、数字视频、数字文本、数字数据、数字多媒体等等,其中这类数字内容将会发行给一个或多个用户。数字内容可以是静态的,例如,文字文件,它也可能是码流形式的,例如,直播的码流音频/视频。典型的发行模式包括有形设备,例如,磁 (软) 盘、磁带、光盘 (小型) (CD),等等,以及无形媒介,例如,电子广告板、电子网络、互联网,等等。一旦用户接受之后,这些用户就可以利用诸如个人电脑等等的媒体播放器这类适用的复制设备来复制或“播放”该数字内容。

[0012] 在这样的背景下,内容所有者或版权所有者,例如,作者、出版商、发行者等等,就希望向支付许可费用或其它一些报酬的许多用户或接受者中的每一位来发行这类数字内容。然而,在这样的背景下,内容可能是一首歌、一本歌曲集、一部电影,等等,发行的目的是

产生许可的费用。这类内容的拥有者,给出的选择是,希望能够约束用户对这类发行的数字内容的使用。例如,内容的拥有者可能希望能限制用户向第二个用户拷贝和再发行该内容,至少是以拒绝向内容的拥有者缴纳第二个用户的许可费用的方式来拷贝和发行。

[0013] 另外,内容的拥有者也希望提供用户采用灵活的方式以不同的许可费用来购买不同类型的使用许可证,与此同时,约束用户在实际购买的许可证的项目上。例如,内容拥有者可能希望允许已发行的数字内容只能播放有限的次数,只能播放限定的总的时间,只能在限定型号的机器上进行播放,只能在限定型号的媒体播放器上进行播放,只能由限定类型的用户来播放,等等。

[0014] 在另一背景下,内容的开发者,例如,在一个组织中的雇员或成员,就希望能将这类数字内容分发给在该组织内的一个或多个雇员或成员或者在该组织外的其它个人,但是又希望能使得其它人不能拷贝该内容。这时,内容的发行就更多地类似于基于同一组织且以一种保密或限制的方式来共享的内容,它不同于基于公众以许可费用或一些其它报酬为交换的发行。

[0015] 那么,在这类背景下,内容可以是文件介绍、电子数据表、数据库、电子邮件、或其它等等,以便于能在办公室架构内进行交换,以及该内容的开发者可能希望确保该内容保留在组织或办公室架构之内并且非授权的个体(例如,竞争者或对手)不能复制该内容。另外,这类内容的开发者还希望限制接受者对这类发行的数字内容所能作的操作。例如,内容拥有者希望限制用户向第二个用户拷贝和再发行该内容,至少是以在允许复制该内容的个体以外暴露该内容的方式来拷贝和再发行。

[0016] 此外,内容开发者可能希望以不同等级的复制权限提供给各种接受者。例如,内容开发者可能希望使得所保护的数字内容,对一类个体是可观看而不可打印,而对另一类个体是可观看且可打印。

[0017] 然而,在任何一类背景下,在发行之后,这类内容拥有者/开发者对数字内容几乎不能进行任何控制。这是特别成问题的,根据目前的事实,每一台个人电脑都包括了所需的软件和硬件,能够进行这类数字内容的原模原样的数字拷贝,以及将这类原模原样的数字拷贝下载到可写的磁盘和光盘中,或者通过网络(例如,互联网)向任何目的文件发送这类原模原样的数字拷贝。

[0018] 当然,作为发行内容的业务的一部分,内容拥有者/开发者可以要求该数字内容的用户/接受者许诺不能以一种不受欢迎的方式再发行该数字内容。然而,这种许诺是容易获得的并且也是容易失信的。内容拥有者/开发者可能试图通过几种众所周知的安全设备中的任何一种,通常涉及加密和解密的方法,来防止这类再发行。然而,很少能防止态度温和的用户来解密已加密的数字内容,以不加密的方式来保存这类数字内容,以及随后再发行该数字内容。

[0019] 因此,就存在着需要提供一种数字版权管理(DRM)和执行的架构和方法,它可允许控制任意格式的数字内容的复制或播放,其中,这种控制是灵活的并且可以由该数字内容的内容拥有者/开发者来限定。更具体的说,就存在着对这类架构的需要,这类架构,特别是在办公室或组织环境中或类似环境中,允许和便于在该环境中由一组所限定的个人或一些所限定的个人团体之间所共享的文档的可控制的复制。再具体的说,就存在着对这类架构的需要,这类架构允许根据该用户是该组的成员将内容许可用户使用,其中,至少是部

分根据高速缓存组的信息来确定这类组的成员。

发明内容

[0020] 本发明至少能部分满足以上的需要,在本发明中,它确定在一个组织中的一员是否是该组织所预先限定的一个或多个目标组的一个成员。该组织维持着一个包括构成 x 等级的多个存储设备的计算机网络并且存储了成员一组的信息。存储设备包括一个目录,它在较高也较昂贵的等级 X 且至少具有一个高速缓存存储器。各个高速缓存都在较低等级,低于 X 的较不昂贵的等级,且至少包括在目录中的部分成员一组信息。等级 1 至 X 对应于最低成本至最高成本。

[0021] 在该方法中,通过搜索各等级的存储设备来发现适用于目标组的项目从而确认各个目标组,并且,如果发现了目标组,就将该目标组放置于一个确认的目标箱 (V) 中。另外,通过搜索各等级的存储设备来发现适用于该元素的项目从而确认该元素,并且,如果发现了适用于该元素的项目,就从该项目中将列出元素的所有组确定为搜索箱中的一个成员并将该组的成员放置于一个搜索箱 (S) 中。

[0022] 之后,首先确定在 (S) 中的任何组是否与在 (V) 中的任何组相匹配。如果能匹配,则可获得结论,该元素就是一个目标组的成员。如果没有匹配的,则该方法就继续移动在 (S) 中的所有组,将其排列成对应于在等级 1 的存储设备的队列,并且执行一个迭代处理过程,以从元素的成员关系路径来搜索任何目标组。

[0023] 迭代的处理过程包括:对从 1 至 X 的各个等级 x ,进行第二确定对应于该等级的队列 (Q_x) 是否是空。如果是空的,则该等级就增 1,并且如果 $x < X$,则处理过程就返回到第二确定的步骤,否则如果 $x = X$ 就可以得出结论,该元素不是任何目标组的成员。如果不是空的,则从 (Q_x) 队列中去除一个组,并且确定所去除的这个组是否已经是在一个放弃箱 (D) 中。

[0024] 如果在 (D) 中,则处理过程就返回到第二确定的步骤。如果不在 (D) 中,则处理过程就在对应于该等级的存储设备中搜索对应于该去除组的项目。如果从搜索中没有发现该项目,则如果 $x < X$ 就将该去除的组移至 ($Q_{(x+1)}$),以及如果 $x = X$ 就将该去除的组移至 (D),并且处理过程就返回到第二确定步骤。如果从搜索中发现了该项目,则将从发现项目所确定的、且该去除组是某一个成员的各个新发现的组,放置至 (S) 中,该去除的组移至 (D) 中,并且处理过程就返回到第一确定步骤。

[0025] 迭代处理过程开发了从该元素到任何所确认的目标组的所有可能的路径。在这样做的过程中,在完成所有较低等级存储设备的所有可能较廉价操作之前,不会发生较高等级存储设备的较昂贵的操作。

[0026] 附图的简要说明

[0027] 当结合附图阅读时,就能较好地理解上述内容以及以下本发明实施例的详细描述。为了便于说明本发明,附图显示了一些较佳的实施例。然而,应该理解的是,本发明并不限制于这里所显示的精确结构和手段。附图包括:

[0028] 图 1 是显示一个能够实现本发明的典型非限制的计算环境的方框图;

[0029] 图 2 是显示一个能够实现本发明的具有多种计算设备的典型网络环境的方框图;

[0030] 图 3 是一种根据本发明适用于发行数字内容的系统和方法的较佳实施例的功能

方框图；

[0031] 图 4 是一种根据本发明适用于发行具有权限管理的数字内容的方法的较佳实施例的流程图；

[0032] 图 4A 显示了一个由图 4 所示方法产生的签署权限标签的结构方框图；

[0033] 图 5 是一种根据本发明适用于许可权限管理的数字内容的系统和方法的较佳实施例的方框图；

[0034] 图 6A 和 6B 是一种根据本发明适用于许可权限管理的数字内容的方法的较佳实施例的流程图；

[0035] 图 7 显示了根据本发明一个实施例由申请者将授权证书和权限标签发送给许可证颁发者和许可证颁发者提交一个组织目录的方框图；

[0036] 图 8 显示了根据本发明一个实施例由多个分支或树构成的组织且其中各个树具有至少一个 DRM 服务器和一个目录的方框图；

[0037] 图 9 显示了与所限定组的成员关系相关的各种实体的方框图；

[0038] 图 10 显示了在图 8 的目录中适用于图 9 所示的实体的目录条目的方框图；

[0039] 图 11 显示了一种典型的诚信系统的执行架构的方框图；

[0040] 图 12 显示了在确定用户是否属于 DRM 服务器树形组中的成员时由图 8 所示 DRM 服务器所执行的关键步骤的流程图；

[0041] 图 13 显示了根据本发明一个实施例在确定用户是否不属于 DRM 服务器树形组中的成员时由图 8 所示 DRM 服务器所执行的关键步骤的流程图；

[0042] 图 14 显示了根据本发明一个实施例在一个高速缓存存储器中的条目的方框图；

[0043] 图 15 和 16 显示了根据本发明一个实施例采用图 14 所示高速缓存存储器的高速缓存存储器架构的一种变型的方框图；以及，

[0044] 图 17A 和 17B 显示了根据本发明一个实施例结合确定用户是否组成员的图 15 和 16 所示高速缓存架构由图 8 所示许可证 DRM 服务器所采用的一种算法所执行的关键步骤的流程图。

[0045] 发明的具体描述

[0046] 计算机环境

[0047] 图 1 和以下的讨论试图对适用于实现本发明的计算机环境提供一个简洁的描述。然而，应该理解的是，所有类型的手持、便携和其它计算设备都可试图结合本发明来使用。而以下是就通用计算机来讨论的，但这只是一个例子，并且本发明只要求一台具有网络服务器的互用性和交互功能的瘦型客户机。于是，本发明可以在网络主机服务器的环境中实现，这意味着在该服务器中只有很少和最少的客户机资源，例如，一个仅仅只是将客户机作为环球网的一个浏览器或界面的网络环境。

[0048] 尽管没有要求，但是本发明也可以通过一个应用程序接口 (API) 来实现，该应用程序接口 (API) 适用于开发者使用，并且 / 或者包含于网络浏览软件中，该软件可以普通的计算机可执行指令的内容来描述，例如，计算机可执行指令可以是一个程序模块并可以由一台或多台计算机来执行，例如，客户工作站、服务器或其它设备。一般来说，程序模块可以包括例程、程序、目标、元件、数据结构、以及其它能够执行特殊的任务或实现特殊的抽象数据类型等等。典型的是，程序模块的功能可以根据各种实施例的要求进行组合或分散。而

且,本领域的熟练技术人员都能意识到:本发明可以采用其它计算机系统结构来实现。其它可以适用于本发明所使用的众所周知的计算机系统,环境,和/或结构包括,但并不限制于,个人电脑(PC)、银行自动取款、服务器计算机、手持或膝上设备、多处理器系统、基于微处理器的系统、可编程消费电子产品、网络PC、小型计算机、大型计算机、以及其它等等。本发明也可以分布式计算环境来实现,在该环境中,任务是由通过通讯网络或其它数据传输媒介连接的远程处理设备来完成的。在分布式计算环境中,程序模块可以设置在包括存储器存储设备的本地和远程计算机存储媒介中。

[0049] 图1说明了一例典型适用于实现本发明的计算系统环境100,虽然,为了是讨论更简洁,该计算系统环境100只是一例适用的计算环境,并且并不试图建议对本发明的使用和功能范围作出任何限制。不应该将计算环境100视为具有与典型操作环境100所说明的任何一个元件和元件组合有关的相关性或必要条件。

[0050] 参考图1,一个适用于实现本发明的典型系统包括一个以计算机形式构成的通用计算设备110。该计算机110的元件可以包括,但并不限制于,一个处理单元120,一个系统存储器130,以及一个系统总线121,该系统总线将包括系统存储器的各个系统元件与处理单元120相耦合。系统总线121可以是几种总线结构中的任何一种,可包括一个存储器总线或存储器控制器,一个外围总线,以及一个使用多种总线架构中的任何一种结构的局部总线。作为一个例子,但并不限制于,这类架构可以包括工业标准结构(ISA)总线、微通道结构(MCA)、增强性ISA(EISA)总线、视频电子标准附件(VESA)局部总线,以及外围元件互联(PCI)总线(也称之为中间件总线)。

[0051] 计算机110一般包括多种计算机可读媒介。计算机可读媒介可以是任何能够被计算机110访问的有效媒介,并且可以包括易失性和非易失性媒介、可移动和非移动的媒介。作为一个例子,但并不限制于,计算机可读媒介可以包括计算机存储媒介和通讯媒介。计算机存储媒介可以包括易失性和非易失性、可移动和非移动的媒介,这类媒介可以采用任何适用于存储信息(例如,计算机可读指令、数据结构、程序模块或者其它等等)的方法和技术来实现。计算机存储媒介包括,但并不限制于,RAM、ROM、EEPROM、闪存存储器,或者其它存储器技术、CDROM、数字视盘(DVD),或者其它光盘存储,磁带盒、磁带、磁盘存储器或者其它磁性存储设备,或者任何其它可以用于存储所需要信息并能被计算机110访问的媒介。通讯媒介一般配备有计算机可读指令、数据结构、程序模块、或者在调制数据信号(例如,载波和其它传递机制)中的其它数据,并且包括任何信息传递媒介。术语“调制数据信号”是指具有以在信号中编码信息的方式来设置和变化它的一个或多个特性的信号。作为一个例子,但并不限制于,通讯媒介包括有线媒介,例如,有线网络或直接有线连接,以及无线媒介,例如,声波、RF、红外、以及其它无线媒介。在计算机可读媒介的范围内也可包括上述任意组合。

[0052] 系统存储器130包括诸如只读存储器(ROM)131和随机存储器(RAM)132这类易失性和/或非易失性存储器类型的计算机存储媒介。基本输入/输出系统133(BIOS)可包含基本的例程,它有助于例如在启动时计算机110内各元件之间传输信息,它一般存储于ROM131。RAM132一般可包含由处理单元120立即访问和/或现场操作的数据和/或程序模块。作为一个例子,但并不限制于,图1说明了操作系统134、应用程序135、其它程序模块136,以及程序数据137。

[0053] 计算机 110 也可以包括其它可移动 / 非移动, 易失性 / 非易失性计算机存储媒介。只是作为一个例子, 图 1 说明了可以对非移动非易失性磁性媒介进行读写的硬盘驱动器 141, 可以对可移动非易失性磁盘 152 进行读写的磁盘驱动器 151, 以及可以对可移动非易失性光盘 156 (例如, CD ROM) 或其它光媒介进行读写的光盘驱动器 155。可以在典型的操作环境中使用的其它可移动 / 非移动、易失性 / 非易失性计算机存储媒介包括, 但并不限制于, 磁带盒、闪存存储卡、数字视盘、数字视带、固态 RAM、固态 ROM, 以及其它等等。硬盘驱动器 141 一般可通过一个非移动存储器接口 (例如, 接口 140) 与系统总线 121 相连接, 以及磁盘驱动器 151 和光盘驱动器 155 一般可通过一个移动存储器接口 (例如, 接口 150) 与系统总线 121 相连接。

[0054] 上述讨论和图 1 所示的驱动器及其相关的计算机存储媒介提供了计算机可读指令、数据结构、程序模块以及用于计算 110 的其它数据的存储。例如, 在图 1 中, 所示的硬盘驱动器 141 可应用于存储操作系统 144、应用程序 145、其它程序模块 146, 以及程序数据 147。值得注意的是, 这些元件可以是相同于也可以是不同于操作系统 134、应用程序 135、其它程序模块 136、以及程序数据 137。这里对操作系统 144、应用程序 145、其它程序模块 146, 以及程序数据 147 以不同的数字给出, 是说明在最低的程度它们是不同的拷贝。用户可以通过输入设备, 例如, 键盘 162 和定位器 162, 通常称之为鼠标、轨迹球或触摸板, 将命令和信息输入计算机 110。其它输入设备 (未显示) 可以是麦克风、操纵杆、游戏垫、卫星天线、扫描仪, 以及其它等等。上述和其它输入设备通常都通过一个与系统总线 121 相耦合的用户输入接口 160 与处理单元 120 相连接, 但是也可以采用其它接口和总线结构, 例如, 并行口, 游戏口和通用串行总线 (USB)。

[0055] 监视器 191 和其它类型的显示设备也可以通过诸如视频接口 190 的接口与系统总线 121 相连接。图形接口 182, 例如, 北桥, 也可以与系统总线 121 相连接。北桥是一个与 CPU 或者与主机处理单元 120 通讯的芯片组, 并负责加速图形接口 (AGP) 的通讯。一个或多个图形处理单元 (GPU) 184 可以与图形接口 182 通讯。在这点上, GPU184 一般可包括片上存储器存储, 例如, 寄存器存储, 并且 GPU184 可以与视频存储器 186 相通讯。然而, GPU184 只是协处理器的一个例子, 并因此在计算机 110 中可以包括多种协处理器件。监视器 191 和其它类型的显示设备也可以通过诸如视频接口 190 与系统总线 121 相连接, 其中视频接口可以依次与视频存储器 186 相通讯。除了监视器 191 之外, 计算机也可以包括其它外围输出设备, 例如, 扬声器 197 和打印机 196, 这些设备都可以通过输出外围接口 195 相连接。

[0056] 计算机 110 可以在网络环境中使用与一个或多个远程计算机 (例如, 远程计算机 180) 相连接的逻辑连接来进行操作。远程计算机 180 可以是个人电脑、服务器、路由器、网络 PC、对等 (点) 设备或其它通用网络节点, 并且一般可包括许多或所有以上所讨论的与计算机 110 有关的元件, 尽管在图 1 中只是已经讨论了存储器存储设备 181。图 1 所标识的逻辑连接可包括局部区域网络 (LAN) 171 和广域网 (WAN) 173, 并且还可以包括其它网络。这类网络环境在办公室、企业内的计算机网络, 企业内部互联网和因特网中是常见的。

[0057] 当用于 LAN 网络环境中时, 计算机 110 可通过网络接口和适配器 170 与 LAN 相连接。当用于 WAN 网络环境中时, 计算机 110 一般可包括一个调制解调器 172 和其它用于建立与 WAN173 通讯的设备 (例如, 互联网)。调制解调器 172 可以是内置的或者是外置的, 可以通过用户输入接口 160 和其它适当的机构与系统总线 121 相连接。在网络环境中, 所描

述与计算机 110 有关的程序模块或者是它的一部分可以存储在远程存储器存储设备中。作为一个例子,但并不限制于,图 1 所说明的远程应用程序 185 可以驻留于存储器设备 181。应该理解的是,所显示的网络连接都是用于举例,并且也可以使用其它在计算机之间建立通讯连接的方法。

[0058] 本领域中的普通技术人员都会意识到,计算机 110 或其它客户设备都可以配置成一个计算机网络的一部分。在这点上,本发明可以属于具有任意数量的存储器或存储单元并具有任意数量可访问任意数量的存储单元或数据块的应用程序和处理能力的任何一类计算机系统。本发明可以应用于在网路环境中配置服务器计算机和客户机计算机并具有远程和本地存储的环境。本发明也可以应用于独立的具有可编程语言功能、编译和执行能力的计算设备。

[0059] 分布式计算方便于共享计算机的资源以及通过在计算设备和系统之间直接交换的服务。这些资源和服务可以包括信息的交换、高速数据缓存存储、以及文件的盘存储。分布计算具有网络连通性的优点,允许客户机充分利用它们集中的能力,以利于整个计划的实现。在这点上,多种设备都可以具有能够于本发明的授权技术交互的应用程序、目标或资源,以便于能监管图形的流水线。

[0060] 图 2 提供了一个典型的网络化或分布式计算环境的示意图。该分布式计算环境包括计算目标 10a, 10b, 等等, 以及计算目标或设备 110a, 110b, 110c 等等。这些目标可以包括程序、方法、数据库、可编程逻辑, 等等。目标可以包括部分相同的或不同的设备, 例如, PDA、电视机、MP3 播放器、个人电脑、等等。各种设备都可以利用网络与另一个设备相通讯。这个网络本身就可以包括其它计算目标和计算设备, 它们可以向图 2 所示的系统提供服务。根据本发明的一个方面, 各个目标 10 和 110 都可以包含一个应用程序, 该程序可以申请本发明的授权技术, 以监管图形的流水线。

[0061] 还应该意识到, 一个目标, 例如, 110c, 也可以是另一个计算设备 10 和 110 的主机。于是, 尽管所说明的物理环境显示了所连接的设备可以作为计算机, 但是这些说明都只是作为举例, 并且另外所说明和讨论的物理环境可以包括各种数字设备, 例如, PDA、电视机、MP3 播放器、等等, 诸如接口的软件目标、COM 目标以及其它等等。

[0062] 可以有多种支持分布式计算环境的系统、元件和网络配置。例如, 计算系统可以通过有线或无线系统、通过局域网络或广域分布网连接在一起。当前, 许多网络都是与互联网相耦合的, 由互联网提供广域分布的计算的基础架构并且包含许多不同的网络。

[0063] 在家用网络环境中, 至少可有四种不同的网络传输媒介且各自可以支持一种唯一的协议, 例如, 电力线、数据 (无线和有线)、语音 (例如, 电话) 和娱乐媒介。大多数家用控制设备, 例如, 电灯开关和家用电器, 都可以用电力线连接。诸如宽带 (例如, DSL 和有线调制解调器) 这样的数据服务可以进入家庭, 并且可以在家庭中使用无线 (例如, 家用 RF 或者 802.11b) 或者有线 (例如, 家用 PNA、Cat5 甚至电力线) 的连通性来访问。语音通信可以采用有线 (例如, Cat3) 或者无线 (例如, 蜂窝式便携无线电话) 进入家庭, 并且可以使用 Cat3 布线方式分布在家庭中。娱乐媒介可以通过卫星和有线方式进入家庭, 并且一般都采用同轴电缆分布在家庭中。IEEE1394 和 DVI 也具有数字互连功能, 作为一组媒介设备。所有的这类网络环境和其它可以使用协议标准的环境都可以相互连接, 以形成一个内部互联网并可以通过因特网与外部世界相连接。简而言之, 所存在的多种分散的资源都可以用于

数据的存储和传输,并继而形成了,计算设备都将需要在数据处理流水线的部分来保护内容。

[0064] “因特网”通常是指采用 TCP/IP 协议的网络和网关的汇集,这些都是在计算机网络领域中众所周知的。TCP/IP 是“传输控制协议 / 接口程序 (TransportControl Protocol/Interface Program) 的各单词的词首字母的缩写。所讨论的互联网可以看成是一个通过计算机来执行允许用户在网络上相互作用和共享信息实现相互连接的地理位置相互分开的远程计算机网络。因为共享了这类广泛分散的信息,所以诸如因特网的远程网络一般都可以进一步发展成一个开放的系统,使得开发者能够,特别是没有限制的条件下,来设计适用于执行特殊操作和服务的软件应用程序。

[0065] 于是,网络的基础架构使得网络的主机可以拓扑成客户机 / 服务器、点对点 (对等)、或者混合架构。“客户机”是一类或一组中的一个成员,它可使用与它无关的另一类或另一组的服务。于是,在计算中,一个客户机是一个处理,例如,粗略地视为一组指令或任务,它可申请另一程序来提供服务。客户机的处理可以应用所申请的服务而不一定需要了解其它程序或服务的本身所工作的细节。在客户机 / 服务器的架构中,尤其是在一种网络系统中,客户机一般都是一台计算机,它能够访问由另一台计算机,例如,一台服务器,所提供的共享的网络资源。在图 2 所示的例子中,计算机 110a、110b 等等都可以视为客户机,而计算机 10a、10b 等等都可视为服务器,这里服务器 10a、10b 等等都保持着随后在客户机计算机 110a、110b 等等中复制的数据。

[0066] 服务器一般是一个可以通过诸如因特网的远程网络访问的远程计算机系统。客户机的处理可以在第一计算机系统中工作,而服务器的处理可以在第二计算机系统中工作,并且可以通过通讯媒介相互通讯,从而提供分布的功能并允许多个客户机可以具有服务器的信息收集的功能。

[0067] 客户机和服务器相互间的通讯可采用由协议层所提供的功能。例如,超文本传输协议 (Hypertext—Transfer Protocol (HTTP)) 是一个与环球网 (WWW) 连接所经常使用的普通协议。典型的是,诸如通用资源定位器 (URL) 或因特网协议 (IP) 地址的计算机网络地址都可以用于对服务器或客户机计算机进行相互识别。网络地址可以是通用资源定位器的地址。例如,可以通过通讯媒介来提供通讯。尤其是,客户机和服务器可以通过 TCP/IP 的连接来实现相互耦合,以满足大容量的通讯。

[0068] 于是,图 2 说明了一个典型的网络化和分布式环境,其中服务器通过一个网络 / 总线与客户机计算机相通讯,在该环境中使用了本发明。更具体的说,根据本发明,多个服务器 10a、10b 等等通过一个网络 / 总线 14 (可以是 LAN、WAN、内部互联网和因特网等等) 与多个客户机和远程计算设备 110a、110b、110c、110d、110e 等等 (例如,便携式计算机、手持式计算机、瘦型客户机、网络化家用电器,或者其它设备,例如, VCR、TV、微波炉、照明、加热器、以及其它等等) 相互连接。因此,可以期待本发明应用于任何与它所希望处理、存储和复制来自监管源的安全内容有关的计算设备。

[0069] 在一个网络环境中,通讯网络 / 总线 14 可以是因特网,例如,服务器 10 可以是环球网服务器,客户机 110a、110b、110c、110d、110e 等都可以通过诸如 HTTP 等多种已知协议与服务器相通讯。服务器 10 也可以像客户机 110 一样服务,也可以具有分布式计算环境的特性。如果合适的话,通讯可以是有线的或者是无线的。客户机设备 110 的通讯可以通过

也可以不通过通讯网络 / 总线 14, 并且可以具有与此相关的独立通讯。例如, 在 TV 和 VCR 的情况, 就可以具有或不具有与它们控制有关的网络。各个客户机计算机 110 和服务计算机 10 都可以装备各种应用程序模块或目标, 并且可以连接或访问各类存储元件或目标, 可以将文件存储于这些存储元件或目标, 或者也可以将文件下载或移植到这些部分。于是, 本发明可以应用于具有能够访问计算机网络 / 总线 14 并能与其交互作用的客户机计算机 110a、110b 等以及能够与客户机计算机 110a、110b 等和其它设备 111 和数据库 20 交互作用的服务器计算机 10a、10b 等的计算机环境。

[0070] 数字版权管理 (DRM) 概述

[0071] 正如所知, 现在参考图 11, 有关数字内容 12, 例如, 数字语音、数字视频、数字文本、数字数据、数字多媒体、等等, 在这类数字内容 12 发行到用户的情况下, 数字版权管理 (DRM) 和执行都是非常需要的。一旦用户接受到, 这样的用户就可以借助于合适的复制设备, 例如, 在个人电脑 14 上的多媒体播放器等等, 来复制和“播放”数字内容。

[0072] 典型的是, 发行这类数字内容 12 的内容所有者或者开发者 (下文称之为“拥有者”) 都希望限制用户再对这类发行的数字内容做些什么。例如, 内容所有者可以希望限制用户为第二个用户拷贝和再复制这类内容 12, 或者可以希望允许所发行的数字内容 12 只能播放有限的次数, 只能播放限定的总的次数, 只能在限定的机器上进行播放, 只能在限定的媒体播放器上进行播放, 只能由限定类型的用户来播放, 等等。

[0073] 然而, 在已经发行之后, 这类内容所有者几乎不能对这类数字内容 12 进行控制。而 DRM 系统 10 允许对任意形式的数字内容 12 的复制或播放进行控制, 其中这种控制是灵活的并且可以由这类数字内容的内容所有者来限定。典型的是, 内容 12 可以通过任何合适的发行通道以包 13 的形式发行给用户。作为发行的数字内容包 13 可以包括采用对称加密 / 解密密钥 (KD) (即, (KD(CONTENT))) 进行加密的数字内容 12, 以及标识该内容、如何获取该内容的许可证、等等的其它信息。

[0074] 基于监管的 DRM 系统 10 允许数字内容等等 12 的拥有者指定可以允许在用户的计算设备 14 上复制这类数字内容 12 之前必须满足的特许规则。这类特许规则可以包括上述的时间要求, 并且将其包含在一个数字许可证或者使用文件 (下文称之为“许可证”) 16 中, 用户 / 用户计算设备 14 (下文中, 该术语是可以相互交换的, 除非环境另有需要) 必须从内容所有者或其代理人那里才能获得该许可证。这类许可证 16 也可以包括解密密钥 (KD), 可用于解密也许是根据由用户计算设备可解密的密钥所加密的数字内容。

[0075] 一件数字内容 12 的内容所有者必须确认用户的计算设备 14 能够遵守在许可证 16 中由该内容所有者所指定的规则和需求, 即, 数字内容 12 如果不能满足许可证 16 中的规则和需求就不能复制。于是, 较佳的是, 向用户计算设备 14 提供一个监管的元件和机制 18, 除非根据在与数字内容 12 有关的许可证 16 中所嵌入的并且用户可以获得的特许规则, 否则就不能复制数字内容 12。

[0076] 监管元件 18 一般具有一个许可证鉴别器 20, 它可以确定该许可证 16 是否有效, 检查在该有效许可证 16 中的规则和需求, 并且在其它事情中以搜索的方式根据所检查的特许规则和需求来确定申请的用户是否具有复制所需要数字内容 12 的权限。应该理解的是, 在 DRM 系统中委托许可证 20 鉴别器 20 根据在许可证 16 中的规则和需求来满足数字内容 12 拥有者的希望, 以及用户应该不能为任何目的、恶意的或者其它轻易的变更这类监管

元件。

[0077] 应该理解的是,在许可证 16 中的规则和需求可以指定用户是否可以根据几种因素中的任一因素具有复制数字内容 12 的权力,这几种因素可以包括用户是谁、用户在哪儿、用户所使用的是哪一种计算设备、什么复制应用程序来调用 DRM 系统,数据、时间等等。另外,许可证 16 的规则和需求可以将许可证 16 限制于,例如,预定数量的播放,或者预定的播放时间。

[0078] 可以根据任何合适的语言和语法来指定在许可证 16 中的规则和需求。例如,可以简单地限制语言所以必须满足的特性和数值(例如,DATE 必须大于 X),或要求语言具有根据所限制的字符(例如,如果 DATE 大于 X,则做...)的功能性能。

[0079] 一旦许可证鉴别器 20 确定了该许可证 16 是有效的并且用户满足了其中的规则和需求,随之就能够复制该数字内容 12。尤其是,要复制内容 12,就必须从许可证 16 获得解密密钥(KD)并且应用于来自内容包 13 的(KD(CONTENT)),从而产生实质的内容 12,因此,事实上复制了实质的内容 12。

[0080] 发行数字内容

[0081] 图 3 是一种用于发行数字内容的系统和方法的功能方框图。“发行”,这里所使用的术语,是指申请和服务遵循的过程,以为所监管的实体建立一组权限和条件,以便于该实体能够发布该内容,以及能够发布给他们的权限和条件。根据本发明,发行过程可包括加密数字内容并且将一系列内容作者试图实现的稳定可执行的权限与所有可能的内容用户相关联。这一过程可以一种安全的方法来进行,以便于能够禁止对任何权限或对内容的访问,除非内容的作者自己试图访问。

[0082] 特别是可以采用三种组织机构来发行安全数字内容:一是在客户机 300 上执行的内容准备程序 302 和准备用于发行的内容,另一是也驻留在客户机设备 300 上的数字版权管理(DRM)应用程序接口(API)306,以及还有一个是通过通讯网络 330,例如,诸如因特网、局域或广域网络或者它们的组合的,与客户机 300 进行通讯耦合的 DRM 服务器 320。

[0083] 内容准备应用程序 302 可以是产生数字内容的任何应用程序。例如,应用 302 可以是字符处理器或者其它产生数字文本文件、数字音乐、视频、或其它这类内容的发行商。内容也可以包括码流式内容,例如,在线的或盒带式的码流式音频/视频等。应用程序 302 可以提供一个用密码写的密钥,用于加密数字内容,从而形成加密的数字内容文件 304,并且用户可以具有与在数字内容文件 304 中的加密内容密切相关的权限数据。该权限数据包括了在数字内容中具有权限的各个实体,以及一组适用于各自标识实体的权限和条件。

[0084] 这类实体可以是,例如,一个人,一类个人或者一个设备。这类权限可以包括可用于数字内容的读取、编辑、拷贝、打印等等权限。条件可以包括最小系统需求、数据和时间的限制、播放的次数,以及其它等等。

[0085] 客户机 API306 通过加密的数字内容并且将具有权限的数据输入到 DRM 服务器 320。使用以下所详细描述的处理过程,DRM 服务器 320 就能确定它是否能够执行具有权限的数据,并且如果是的话,DRM 服务器 320 签署权限数据,从而形成了一个签名的权限标签(SRL)308。然而,一般来说,任何所监管的实体都能够签署权限数据,较佳的是采用由 DRM 服务器 320 所确认的密钥来签署。例如,客户机可以使用一个由 DRM 服务器 320 提供给它的密钥来签署权限数据。

[0086] 权限标签 308 可以包括表示权限描述、加密内容的密钥、以及在权限描述和加密内容密钥上的数字签名的数据。如果 DRM 服务器 320 正在签署权限标签,则它使所签署的权限标签 308 通过客户机 API306 返回到客户机了,将该标签存储于在客户机设备 300 中。内容准备程序 302 使所签署的权限标签 308 与加密的数字内容文件 304 相关联,例如,使之串起来,以形成一个权限管理的内容文件 310。然而,值得注意的是, SRL308 能够存储于与内容文件 304 隔开的已知位置上,将参考 SRL308 使内容文件 304 串起来,从而产生内容文件 310。

[0087] 现在,参考图 4,图 4 显示了一种用于发行具有权限管理的数字内容的方法。在步骤 402,一个应用程序 302 产生一个可以用于加密数字内容的内容密钥 (CK)。尽管任何密钥都可以用于加密数字内容,但是,该内容密钥 (CK) 典型的是一个对称密钥。正如众所周知的,采用一个对称的密钥,可以通过一个对称密钥算法来进行加密和解密。因此,在发送方和接受方之间共享时,(CK) 应该能较好地隐藏起来。

[0088] 在步骤 404,应用程序 302 采用 (CK) 来加密数字内容,以形成加密的数字内容 304(例如,(CK(content)))。因此,就由内容的发行商或者由其它实体产生了对应于 CK(content) 的权限数据。值得注意的是,这类权限数据可以是用户的权限数据和从预先定义的模板中所获得权限数据。正如以上所讨论的,该权限数据可以包括一系列构成授权消费该内容的实体、指定各个实体进行与内容有关的处理的权限,以及可能会影响这些权限的任何条件的列表。

[0089] 在步骤 406,API306 产生了一个第二加密密钥 (DES1),它是用于加密内容的密钥 (CK)。较佳的是,(DES1) 也是一个对称的密钥。在步骤 408,API306 用 (DES1) 加密 (CK) 而产生 (DES1(CK))。在步骤 410,API306 放弃了 (CK),因为现在只能通过解密 (DES1(CK)) 来获得 (CK)。为了确保 (CK(content)) 能保护中央 DRM 服务器 320 并且根据权限数据集中对内容进行“许可证申请”,所以,在步骤 412,API306 只与提供服务的 DRM 服务器 320 相接触并且重新找到公共密钥 (PU—DRM)。在步骤 414,API306 采用 (PU—DRM) 来加密 (DES1)。以形成 (PU—DRM(DES1))。于是,就将 (CK) 保护在 (PU—DRM(DES1)) 中,从而确保 DRM 服务器 320 是一个能够访问 (CK) 并能根据要求来解密 (CK(content)) 的唯一实体。在步骤 416,API306 采用 (DES1) 加密权限数据(例如,被授权实体的列表以及与在列表中的各个授权机构有关的各个权限和条件),以产生 (DES1(权限数据))。

[0090] 在另一实施例中,(CK) 可以用于直接加密权限数据,以产生 (CK(权限数据)),并且 (PU—DRM) 可以用于直接加密 (CK),以产生 (PU—DRM(CK)),从而完成 (DES1) 的上述使用。然而,使用 (DES1) 来加密权限数据和 (CK) 允许这类 (DES1) 能够兼容任何也许能适合于 DRM 服务器的特殊算法,但是,(CK) 可以由一个独立于 DRM 服务器的实体来指定并且可能不适合于它。

[0091] 在步骤 418,内容保护应用程序 302 将 (PU—DRM(DES1)) 和 (DES1(权限数据)) 提交给 DRM 服务器 320,作为用于签名的权限标签。另一种可替换的方法是,客户机自身可以采用以下所阐述的方法来签署权限数据。如果正在提交给服务器用于签名的权限数据,那么在步骤 420,DRM 服务器 320 就访问权限数据并且验证它是否能够执行在所提交的权限数据中的权限和条件。为了验证了它能够执行该权限数据,则 DRM 服务器 320 就将对应于 (PU—DRM) 的私钥 (PR—DRM) 应用于 (PU—DRM(DES1)),以产生 (DES1),并随后将 (DES1)

应用于 (DES1(权限数据)), 以用明文来产生权限数据。随后, 服务器 320 能够进行任何策略性校验, 以验证在权限数据中所指定的用户、权限、以及条件是在由服务器 320 所执行的任何策略内。服务器 320 签署原始提交的包括 (PU—DRM(DES1)) 和 (DES1(权限数据)) 的权限标签, 以产生一个签名的权限标签 (SRL) 308, 其中, 该签名是基于 DRM 服务器 320 的私钥 (PR—DRM), 并且将 SRL308 返回到 API306, 由 API306 将所返回的 SRL308 传输至客户机的应用程序 302。

[0092] SRL308 是一个数字签名的文档, 这使得文档能够防篡改。因此, SRL308 是与加密内容所实际使用的密钥类型和算法无关的, 但是它对所要保护的内容保持着稳定的 1—1 的成员关系。现在参考图 4A, 在本发明的一个实施例中, SRL308 可以包括作为 SRL308 的基础的内容的信息, 也许可包括该内容的 ID; 关于签署 SRL308 的 DRM 服务器的信息, 可包括 (PU—DRM(DES1)) 信息以及其它参考信息, 例如, 用于确定 DRM 服务器在网络中的位置的 URL 信息和如果 URL 失败的返回信息; 描述 SRL308 自身的信息; (DES1(权限数据)): (DES1(CK)); 以及一个数字签名 (S(PR—DRM)) 等等。

[0093] 通过确保所监管的实体签署的权限数据能够产生一个签名的权限标签 308, DRM 服务器 320 就能够断定它能够根据由发行商所规定的项目发布用于该内容的许可证, 正如在权限标签 308 的权限数据中所述的。应该理解的是, 要求用户获得复制内容的许可证, 特别是, 因为许可证包含了内容的密钥 (CK)。当用户希望获得适用于加密内容的许可证时, 用户就可以提交许可证申请, 该申请可以包括适用于内容的 SRL308 和授权 DRM 服务器 320 和其它许可证颁发机构验证用户的信用证。许可证颁发机构随后可以加密 (PU—DRM(DES1)) 和 (DES1(权限数据)), 以产生权限数据, 列出所有由作者 (如果有的话) 对许可证申请实体所授予的权限, 以及构成只具有所指定权限的许可证。

[0094] 正如以上所阐述的, 一旦应用程序 302 接受到 SRL308 之后, 这类应用程序 302 就将签名的权限标签 308 与所对应的 (Ck(content)) 304 相串起来, 以形成权限管理的数字内容。可替换的是, 权限数据可参考加密数字数据所提供的位置存储于一个已知的位置上。于是, 由 DRM 允许的复制应用程序就可以通过复制应用程序所试图复制的一段内容来发现该签名的权限标签 308。这一发现可以触发复制应用程序对 DRM 许可证服务器 320 进行许可证申请的初始化。发行应用程序 302 能够将 URL 存储于诸如 DRM 许可证服务器 320, 或者 DRM 服务器 320 可以在将它自己的 URL 作为一段元数据在对权限标签数字签名之前嵌入到权限标签中, 使得复制应用程序所调用的 DRM 客户机 API306 能够识别正确的 DRM 许可证服务器 320。

[0095] 获得适用于发行内容的许可证

[0096] 现在参考图 5, 图 5 显示了一个适用于特许权限管理的数字内容的系统和方法。“特许”, 本文所使用的术语, 是指应用程序和服务器从事申请和接受许可证且该许可证能够使在许可证中登记注册的实体可以根据在许可证中所指定的条目来消费内容的处理过程。输入该特许处理的过程可以包括与适用于申请许可证的内容相关联的签了名的权限标签 (SRL) 308, 以及适用于申请许可证的实体的公共密钥的授权证书。值得注意的是, 并不一定需要该申请许可证的实体是正在申请许可证的实体。典型的是, 一个许可证可以包括由 SRL308 所描述的权限, 一个能够解密所加密内容的加密密钥, 以及一个在权限描述上的数字签名和申明其合法性和防止篡改的加密密钥。

[0097] 预备步骤是,客户机 API306 先将权限管理内容 310 中所签署的权限标签 308 通过通讯网络 330 转发至 DRM 服务器 320。正如以上所讨论的,权限标签 308 可包含根据 DRM 服务器 320 的公共密钥 (PU—DRM) 来加密的内容密钥 (CK) (即, (PU—DRM(CK)))。然后,在发布许可证的处理过程中,DRM 服务器 320 将 (PR—DRM) 应用于 (PU—DRM(CK)),以获得 (CK)。于是,就可使用在公共密钥授权证书中的公共密钥 (PU—ENTITY),该密钥在申请重新加密 (CK) (即, (PU—ENTITY(CK))) 的许可证中是不被理会的。该新加密的 (PU—ENTITY(CK)) 随后放置于许可证中。于是,该许可证就能在没有过多暴露 (CK) 的情况下返回至调用器,因为只有保持着对应于 (PU—ENTITY) 的私钥 (PR—ENTITY) 的拥有者才能够从 (PU—ENTITY(CK)) 中恢复 (CK)。随后,客户机 API306 使用 (CK) 来解密所加密的内容,以形成解密的数字内容 312。客户机应用程序 302 可随后根据在许可证中所提供的权限来使用解密后的数字内容 312。

[0098] 另外,正如以下所详细阐述的,诸如进行发行的客户机可以,例如,向它自己发布消费该内容的使用许可证。

[0099] 现在参考图 6A 和图 6B,显示了特许权限管理的数字内容的方法。在步骤 602,一个许可证发布机构,例如,一个 DRM 服务器 320,可以接受一个许可证申请,该许可证的申请可以包括用于一个和多个所申请许可证的一个公共密钥的授权和一个标识。粗略的说,如果指定了一个标识的话,则 DRM 服务器 320 就可以从目录、数据库、或者其它等等中产生一个对应的公共密钥授权。如果一个所申请的许可证仅仅只适用于一个许可证,则就只命名一个授权证或标识。如果所申请的许可证是可适用于多个许可证,就可以对各个潜在的许可证命名一个授权证或一个标识。在步骤 604,如果需要的话,可以授权于申请的实体 (例如,提出许可证申请的实体)。在步骤 606,如果也是需要的话,就确定该实体是否允许申请许可证。

[0100] 在步骤 608,如果发布机构确定在许可证申请中没有包括公共密钥的授权证,则发布机构就可以使用所指定的标识在目录服务或数据库中进行查找合适公共密钥的授权证。在步骤 610,发布机构确定在目录中有授权证,则在步骤 612,就重新恢复该授权证。如果在申请或在目录中都没有发现适用于一个给定的潜在领许可证者的授权证,则许可证服务器就不能产生适用于该潜在领许可证者的许可证,因此,在步骤 614,就向申请机构发出一个出错的信息。

[0101] 假定 DRM 服务器 320 具有适用于至少一个潜在领许可证者的公共密钥的授权证的话,则在步骤 616,该 DRM 服务器 320 就使得各个领许可证者的授权证的信用有效。如果是无效的话,DRM 服务器 320 就确定领许可证者的授权证的发布者没有在委托发布者的列表中,那么该领许可证者的申请也就失败,并且在步骤 614 产生出错。于是,由委托发布者未发布授权证的任何潜在的领许可证者都不能接受到许可证。

[0102] 另外,DRM 服务器 320 对来自委托发布者的授权链中的所有实体较佳地执行数字签名确认,以授权各个许可证获得者各个密钥授权证。确认在链中的数字签名的处理过程是采用众所周知的算法。如果适用于一个给定潜在的许可证获得者的公共密钥授权证是无效的,或者在链中的授权证不能有效,则该潜在的许可证获得者不能被信任,因此,就不能向该潜在的许可证获得者发布许可证。否则,在步骤 618,就可发布一个许可证。在步骤 620 重复该处理过程,直至都已经处理完了所有已经申请许可证的实体。

[0103] 正如图 6 所示, DRM 服务器 320 开始确认在许可证申请中所接受到的签名权限标签 308。在一个实施例中, DRM 服务器 320 有一个对每一个权限标签签名的主拷贝。在特许的时间(在步骤 622), DRM 服务器 320 可以恢复该主权限标签的拷贝。主权限标签可以具有比在许可证申请中所发送的权限标签拷贝更多的数据, 因此, 可以作为产生所申请许可证使用的权限标签。如果在 DRM 服务器中没有发现主权限标签, 则在步骤 624, 根据预先所规定的策略来确定是否基于在申请中的权限标签发布许可证。如果该策略是不允许的, 则在步骤 626, 中止该许可证申请, 并且在步骤 628 向 API 发送一个出错信息。

[0104] 在步骤 630, DRM 服务器 320 确认 SRL 并且指定它的数字签名。如果 SRL308 没有被确认, 则在步骤 626, 中止该许可证申请, 并且在步骤 628 向 API306 发送一个出错信息。

[0105] 在所有的确认都完成之后, DRM 服务器就根据 SRL308 来构成一个许可证, 它可适用于各个所批准的许可证。在步骤 632, DRM 服务器 320 对各个许可证产生适用于需要发布许可证的各个权限描述。对各个许可证来说, DRM 服务器 320 针对在权限标签中的权限描述所命名的标识来验证在许可证的公共密钥授权证中所命名的标识。在步骤 636, DRM 服务器 320 从 SRL308 中获得 (PU—DRM(DES1)) 和 (DES1(CK)), 并且将 (PR—ENTITY) 应用于所获得的 (CK)。随后, 发布机构使用许可证获得者的公共密钥授权证中得到的 (PU—ENTITY) 重新加密 (CK), 以形成 (PU—ENTITY(CK))。在步骤 638, DRM 服务器 320 将 (PU—ENTITY(CK)) 所产生的权限描述与数字签名串起来, 从而形成了使用 (PR—DRM) 的数据结构 (即, S(PR—DRM))。该签名数据结构随后作为适用于该特殊许可证获得者的许可证。

[0106] 在步骤 640, DRM 服务器 320 确定不需要为特殊申请产生更多的许可证。随后, 在步骤 642, 将所产生的许可证与一个被批准的授权证链一起返回给申请的实体, 该授权证链约束了返回给受委托部分的许可证。

[0107] 利用目录的特许

[0108] 在发行用于保护内容的许可证时, 许可证发行机构 (下文称之为“许可证颁发者”) 考虑由内容所发送的 SRL308, 以确定适用于哪一个用户 / 组 / 群 / 部门 / 平台 / 等等 (下文称之为“机构”) 是需要提供权限的, 并且发送授权证以标识许可证的申请者。据此, 许可证颁发者确定在 SRL308 中所列举的那些权限是可以颁发给申请者的。接着, 许可证颁发者检查在 SRL308 中所列的实体, 并且将这些实体与申请者相比较。如果 SRL308 指定了有一个特殊的组来接受一个许可证并且该申请者是该组中的一个成员, 则申请者就能够获得一个具有如同 SRL308 中对该组所阐述权限的许可证。同样, 如果 SRL308 指定了一个特殊的用户来接受一个许可证并且该申请者就是这个用户, 则申请者就能够获得一个具有如同 SRL308 中对该用户所阐述权限的许可证。应该理解的是, 一个特殊的 SRL308 可以列出几个实体以及其权限, 并且一个特殊的申请者可以获得一个基于作为一个和多个实体成员的许可证。

[0109] 在本发明的一个实施例中, 正如图 7 所示, 利用一个标识符 1204 可以在所发送的授权证 1202 中标识申请者, 其中, 标识符 1204 例如可以只是一个别名, 通过该别名就可以在一个组织的目录 1206 中标识该申请者。进而, SRL308 就可根据这样一个标识符 1204 列出各个权限授予实体。于是, 作为对许可证 1208 申请的一部分, 许可证颁发者 1210, 典型的是一个 DRM 服务器 320, 就可以从授权证 1202 中获得该申请者的标识符 1204, 并且将该标识符 1204 与在所发送的 SRL308 中所列的所有标识符 1204 相比较。如果发现相匹配, 则许

可证颁发者 1210 就向该申请者颁发一个适用于该申请者的标识符 1204 并具有在

[0110] 典型的是,目录 1206 是一个有用信息的资料库,它包含了对该组织或其部分感兴趣的信息。特别是,目录 1206 保持着一些有关各个用户在该组织中以及包括用户所保持成员关系的每一个组的重要信息。事实上,在目录 1206 根据用户所列出的部分至少是可以定义这类用户的成员关系。

[0111] 值得注意的是,在组织特别大以及需要的情况下,对该组织的计算可以一些逻辑方式分成多个部门或“分支”。例如,各个分支都可以是一个基于地理位置的基层组织,例如,该组织在 Pennsylvania 的 Wikes—Barre 有一个办公室,还在 Florida 的 Boynton 有另外一个办公室,在这种情况下,各个这样的办公室都可以作为一个独立的分支。另外,例如,各个分支都可以是一个基于功能的基层组织,例如,该组织具有一个会计部门或一个产品部门,在这种情况下,各个这类部门都可以作为一个独立的分支。当然,分支可以根据任何其它条件来定义,这些都不脱离本发明的精神和范围。

[0112] 意味深长的是,假定各个分支都具有它自己的目录 1206 和目录组 1206(下文中称之为“目录 1206”),并且各个分支都具有至少一个 DRM 服务器 320。现在,参考图 8,显示了一个组织有两个分支,分支 A 和分支 B,其中,分支 A 具有一个目录 A 和一个 DRM 服务器 A,而分支 B 具有一个目录 B 和一个 DRM 服务器 B。值得注意的是,各个分支都作为一个保护的领地来工作,在该领地中配置了相对应的目录和 DRM 服务器。于是,至少从安全的观点出发,在分支 A 中的 DRM 服务器 A 应该能够比较方便地访问在目录 A 中的信息,而还是从安全的观点出发,在分支 B 中 DRM 服务器 B 近乎也能够比较方便地访问在分支 A 中的目录 A 中的信息。然而,值得注意的是,多个具有相互操作需要的分支一般都可以复制在它们之间的信息并且可以有选择性地在它们之间建立相互信任的成员关系。

[0113] 分支中组成员关系的确定

[0114] 假定在某一瞬间,并不存在着由图 8 的分支 A 和图 8 的分支 B 所表示的组织总数,应该理解的是,分支 A 的目录 A 包含了有关该组织的所有目录信息,它包括了所有用户和用户所归属的所有组。值得注意的是,这里,这类组实际上可以包括多种层次的组。于是,正如图 9 所示,在这种情况下,组 1 具有一个作为直接成员的组 2 和组 3,并且组 3 可以具有作为直接成员的用户 1。在这种情况下,用户 1 是组 3 的一个直接成员、是组 1 的一个间接成员,但不是组 2 的一个成员。正如可以理解的那样,如果一个 SRL308 列出了适用于组 1SRL308 中所指定权限的许可证 1208。

[0115] 此外,利用目录 1206 的有效性,许可证颁发者 1210 也可以确定申请者是否是在 SRL308 中所列出的任何其它实体的成员,假定目录 1206 包含了能够适当反映在各种这类其它实体中的申请者成员状态的交叉参考信息。典型的是,目录 1206 不仅列出了各个申请者的标识符 1204,还列出了申请者是其成员的各个组/群/部分/平台/其它机构/等等的标识符 1208。值得注意的是,目录 1206 可以包括多个标识符 1208,例如,一个邮件地址,另一个邮件地址,一个 ID,另一个 ID,一个组的成员关系,一个历史上所用的标识符,以及/或者其它等等。

[0116] 利用从具有标识符 1204 的申请者接受到的授权证 1202 和从申请者接受到的 SRL308 中所得到的权限,一般来说,许可证颁发者 1210 就能够向申请者颁发许可证 1208,如果除其它事情外,能从目录中来确定该申请者是一个在 SRL308 中所列出的任何实体的

成员的话。如果是,并假定满足适用于许可证的所有其它条件,则许可证颁发者 1210 就向申请者颁发一个许可证,该许可证可具有适用于指定为从 SRL308 所获得的列出实体的权限和条件。

[0117] 在本发明的一个实施例中,在 SRL308 中列出的一个实体可以表示为某些个体的一组或一群或其它集合(下文中称之为“组”),其中各个组都可以在目录 1206 中进行适当的标识。这类组可以包括一个开放邮件组,例如,一个分布的列表或一个邮件的入口,或者一个诸如在网络操作系统的连接中可定义的安全组,以及其它等等。

[0118] 值得注意的是,在 SRL308 中根据组所指定的权限和根据组的向申请者所颁发的许可证 1208 都能够实现在一个企业和组织结构内部的数字版权的管理。例如,一个文档和电子邮件可以采用 DRM 保护,使得一个给定部门内的所有成员都可以具有读取该文档和电子邮件的权限。假定,在一个组织的目录 1206 中存在着适用于该部门的一个组(例如,电子邮件的别名),这是一种最普遍的情况,文档或电子邮件的作者可以根据组而不是个体来授予权限。应该理解的是,这类面向组授予权限的优点包括便于作者在指定具有权限个体的分类中使用。另外,通过根据组来指定权限,当新的成员加入该组或老的成员离开该组时,所指定的权限就不会变成“过时的”。因此,只有该组的成员关系继续保持在组织目录 1206 中,该组所有目前的成员都能行使这些权限。

[0119] 确定申请者是否是组的成员

[0120] 或 3 成员的权限,假定满足可所有的其它条件,用户 1 就可以根据该权限获得一个许可证 1208,而如果 SRL308 只列出了适用于组 2 成员的权限,则用户 1 就不可能获得这类许可证 1208。更具体的说,是 SRL308 中所命名组的一个直接或间接成员的用户能够获得一个具有准予该组使用的权限和条件的许可证。

[0121] 正如图 10 所示,显示了在目录 A 中对各个组 1 至 3 和用户 1 所记录的目标/记录。值得注意的是,各个记录都以列表的形式记录各个实体并包括该实体的所有直接成员(假定,该实体是组)和该实体所具有的所有直接成员关系,无论该机构是用户还是另一个组中的一个子组。于是,组 1 具有作为其直接成员的组 2 和组 3 并且不是任何其它组的一个直接成员,组 2 没有成员并且是组 1 的一个直接成员,组 3 具有直接成员的用户 1 并且还是组 1 的一个直接成员,用户 1 作为一个用户没有任何成员但是组 3 的一个直接成员。当然,图 9 和图 10 所示的目录 A 过于粗略和简单,但在实质上,这非常类似于具有许多用户和组的目录。不论怎么说,所示的这类目录 A 可适用于本发明的讨论目的。

[0122] 在用户 1 向 DRM 服务器 A 提交了一个 SRL308 的过程中,所提交的 SRL308 列出了一个和多个组以及它们的权限,因此。DRM 服务器 A 至少是能够部分根据目录 A 来确定用户 1 是否是在 SRL308 所列出的任何组的一个直接或间接成员,从而决定是否向用户 1 发行许可证 1208。正如所能理解的那样,这种确定可以采用两种常用方法中的任一方法来执行,一是从所列的组来确定用户,另一是从用户来确定所列的组。

[0123] 执行从组到用户的确定是一种典型的更为直截了当的处理方法,但是由于组可以具有许多成员,因此它就必须花相当多的时间来检查所发现的所有用户以确定是否存在着一个相匹配的。执行从用户到组的确定是一种典型的非直截了当的方法,但是由于一个用户通常不会是一个太多组的成员,这就可以花相对较少的时间来检查所发现的所有组以确定是否存在着一个相匹配的。在许多情况下,至少是从经验上已经发现了执行从用户到组

的确定在整体上显示更加经济和合意。

[0124] 继续参考图 9 和图 10,现在也可以参考图 12,来产生一个确定与用户是组 1 的一个成员有关的组成员确定,随后,DRM 服务器 A 可进行下列处理。在预先准备的步骤,DRM 服务器 A 讯问目录 A 是否存在组 1 (步骤 1201)。正如可以理解的那样,如果对此类讯问响应显示组 1 并不存在,则处理过程就结束,因为用户 1 不能够成为一个不存在组的成员 (步骤 1203)。于是,不论从用户到组所执行的确定的事实如何,都需要执行讯问是否存在这组的预先准备步骤,如果不需要的话,可避免执行任何其它步骤。

[0125] 当然,正如所显示的,组 1 确实存在,则事实上就需要对它响应。于是,DRM 服务器 A 就讯问适用于所有组的目录 A 用户 1 是否是其一个成员并且目录 A 返回用户 1 是组 3 的一个成员的信息 (步骤 1205,1207)。当然,组 3 并不是组 1,所以 DRM 服务器 A 还不能确定用户 1 是否是组 1 的一个成员 (步骤 1209)。不过,DRM 服务器 A 必须继续,只要所处的情况是,通过用户 1 是组 3 的直接成员并且组 3 是组 1 的直接和间接成员,之后又可能发现用户 1 还是组 1 的一个间接成员。

[0126] DRM 服务器 A 继续讯问适用于所有组的目录 A 组 3 是其成员吗,并且目录 A 返回组 3 是组 1 的一个成员的信息 (步骤 1211,1213)。因此,DRM 服务器 A 已经通过用户 1 是组 3 的一个成员来确定用户 1 事实上是组 1 的一个间接成员 (步骤 1215)。

[0127] 正如现在所应该理解的那样,DRM 服务器 A 可能不得不通过这么多的组和这么多的循环的重复讯问,来最后确定用户 1 事实上是组 1 的一个成员。另外,如果所处的情况是用户 1 事实上不是组 1 的成员,则 DRM 服务器 A 不得不在最后确定用户 1 不是组 1 的一个成员之前,通过对用户 1 的所有可能组成员的整体来讯问是否是直接和间接的成员。正如以下更详细的讨论那样,因为这类讯问可以是非常昂贵的并且可以在连续的基础上执行,这类探索的结果可以存储于一个或多个高速缓存存储器中。

[0128] 组成员关系确定跨分支

[0129] 当一个组织具有多个配置的分支,并且需要用户和组的成员关系信息可越过分支边界供使用时,没有直截了当的方法来确定一个用户与一个特殊组相关的组成员关系,特别是当所讯问的 DRM 服务器 320 处于不能自然保持在目录中所需要的信息的时候。但是,DRM 服务器 320 却必须考虑确实保持了这类信息的另一个分支。

[0130] 特别是,一种感兴趣的情况是,要求诸如图 8 所示的 DRM 服务器 A 的 DRM 服务器 320,能够根据列出一个组的 SRL308 向用户发行一个许可证,而该组并不属于驻留在 DRM 服务器 A 中的分支 A。相反,这类组是属于分支 B 并因此在分支 B 的目录 B 中有该组的记录。其结果是,就必须对有关分支 B 的目录 B 进行与正在讨论用户是否是正在讨论组的成员有关的组成员关系确定,因为这类目标具有针对正在讨论组以及针对该组的组信息的记录。

[0131] 然而,明显的是,分支 A 的 DRM 服务器 A 一般不能直接讯问分支 B 的目录 B 以获得大量直接信息,因为这类跨分支的讯问通常受到限制或阻止脱离所关注的安全问题。相反,在本发明的一个实施例中,DRM 服务器 A 要求分支 B 的 DRM 服务器 B 来讯问分支 B 的目录 B 并且向 DRM 服务器 A 报告其讯问的结果。值得注意的是,DRM 服务器 B 对目录 B 的讯问并不是一种跨分支的讯问,因此就不受对它的限制。

[0132] 必须理解的是,DRM 服务器 A 在要求 DRM 服务器 B 进行讯问之前,必须真实的了解 DRM 服务器 B 是事实上最适合于进行讯问的 DRM 服务器 320,因为正在讨论组是属于分支 B

的。为了更加简化, DRM 服务器 A 必须在要求 DRM 服务器 B 直接讯问目录 B 之前知道正在讨论组是属于分支 B 的。然而, DRM 服务器 A 仅仅只能讯问目录 A, 因此就不能了解到正在讨论组所属于的分支, 除非该目录 A 具有这样的信息。

[0133] 因此, 并且在本发明的一个实施例中, 目录 A 确实不具有这类信息, 正如图 8 所示。特别是, 目录 A 具有一个适用于正在讨论组的指针目标, 它将与正在讨论组有关的任何讯问实体都指向分支 B。这类指针目标可以包括例如分支 B 的地址, 在 DRM 服务器 A 需要有关正在讨论组的信息时, 上述地址可以返回到发出讯问的 DRM 服务器 A。预先假定, 分支 B 的目录 B 具有用于正在讨论组的记录目标。于是, DRM 服务器 A 可以采用该地址来发现分支 B 并且从那里继续下去。

[0134] 在本发明的一个实施例中, 尤其是, 分支 B 的地址是在分支 B 中的目录 B 的地址。具有这类地址的这类 DRM 服务器 A 都可以据此与目录 B 相接触。然而, 由于 DRM 服务器 A 是在分支 A 中, 而目录 B 是在分支 B 中, 正如以上所间接提到的, 目录 B 通常是不能向 DRM 服务器 A 提供任何大量跨分支的信息。不过, 目录 B 可以向 DRM 服务器 A 提供 DRM 服务器 B 的地址。

[0135] 因此, 在本发明的一个实施例中, DRM 服务器 A 从目录 B 接受到 DRM 服务器 B 的地址, 并且根据该地址要求 DRM 服务器 B 能代表 DRM 服务器 A 来讯问目录 B。特别是, DRM 服务器 A 要求 DRM 服务器 B 来讯问正在讨论的用户是否是正在讨论组的一个成员, 并且向 DRM 服务器 A 返回该答案。预先假定, 这类要求可以包括一些类型的 DRM 服务器 A 的标识作为 DRM 领域中的 DRM 服务器 320, 例如, 将授权链中所标识的授权证引导到授权的监管层中作为又 DRM 服务器 B 所识别的。因此, 只要所标识的授权证得到了确认, DRM 服务器 B 就可代表 DRM 服务器 A 来执行目录 B 的讯问。

[0136] 假定 DRM 服务器 B 确实代表了 DRM 服务器 A 来执行对目录 B 的讯问, 例如, 可以上述有关图 12 所阐述的方式, DRM 服务器 B 最后确定正在讨论的用户是否是正在讨论组的一个成员, 并且将该信息返回 DRM 服务器 A, 作为对其要求的响应。

[0137] 值得注意的是, 当 DRM 服务器 B 询问目录 B 是否正在讨论的用户是正在讨论组的一个成员时, 则在目录 B 中必须能够表示该正在讨论的用户, 如果没有采用一个记录目标的话, 至少可采用一个指针目标来表示。此外, 这类指针目标必须能包括与正在讨论的用户正是各组成员的有关信息, 特别是因为, 当组成员关系信息是从用户到组来确定时, 可以参考这类有关成员的信息, 正如以上所讨论的那样。

[0138] 现在参考图 13, 概括的说, DRM 服务器 A 和代表 DRM 服务器 A 的 DRM 服务器 B 可执行以下步骤来确定跨分支组成员关系。

[0139] 预先, DRM 服务器 A 接受到用户对使用对应于一段内容的许可证 1208 的申请 (步骤 1301)。该申请可包括标识用户的授权证以及与内容相关联的 SRL308。作为响应, 假定 SRL308 标识了其中的组, 并且也假定不存在适当的高速缓存的信息, DRM 服务器 A 讯问目录 A 以返回适用于该标识组的任何目标 (步骤 1303)。该背景假定了该组属于分支 B, 使得该组事实上以目录 A 来表示指向分支 B 的指针目标, 并且 DRM 服务器 A 从目录 A 接受对分支 B 的指针目标, 于是很明显该组属于分支 B (步骤 1304), 并且从接受到的指针目标中恢复分支 B 的地址 (步骤 1305)。

[0140] 之后, DRM 服务器 A 采用所恢复的地址来讯问在分支 B 中的目录 B 来寻找 DRM 服

务器 B 的地址,并且在响应之后目录 B 就将该地址返回 DRM 服务器 A(步骤 1307,1309)。这种获得 DRM 服务器 B 地址的特殊机制都是众所周知的或者应该在相关出版物上很明了的,因此本文就不再需要进行详细的讨论。于是,可以在不脱离本发明的精神和范围的下采用任何所合适的机制。

[0141] 采用 DRM 服务器 B 的地址,DRM 服务器 A 随后就要求 DRM 服务器 B 来讯问目录 B 该用户是否是该组的一个成员(步骤 1311)。接着,由 DRM 服务器 A 将要求传递给 DRM 服务器 B 采用特殊的机制是众所周知的,并且应该在相关出版物上是很明了的,因此本文就不再需要进行详细的讨论,于是,可以在不脱离本发明的精神和范围的下采用任何所合适的机制。

[0142] 对采用 DRM 服务器 A 来说,再次假定并不存在合适的高速缓存存储器,DRM 服务器 B 要求目录 B 能返回适用于标识组的任何目标(步骤 1313)。本背景假定该组属于分支 B,使得该组事实上在目录 B 中作为一个记录的目标,并且 DRM 服务器 B 从目录 B 接受所记录的目标,从而意味着该组属于分支 B(步骤 1315)。值得注意的是,DRM 服务器 B 在这点上可以审查该组的成员,正如在接受确定该用户是否是该组的直接成员的目标中阐述的那样(步骤 1317)如果是,则确定是肯定的,并且 DRM 服务器 B 就向 DRM 服务器 A 报告该结果(步骤 1319)。如果不是,则 DRM 服务器 B 就要求目录 B 返回适用于用户的任何目标,并且随之采用结合图 2 所示方法通过在目录 B 中对从用户到组的全面考察来继续确定该用户是否是该组的成员(步骤 1323)。一旦结束了该全面考察之后,DRM 服务器 B 将确定在用户和组之间是否存在成员关系,并且向 DRM 服务器 A 报告该结果(步骤 1325)。

[0143] 当然,基于这样的确定,DRM 服务器 A 就能决定是否将来自用户的要求兑现为在步骤 1301 中被接受的(步骤 1327)。应该理解的是,如果用户并不是该组的成员,则要求就被否认,除非发生了已经发现用户是在所接受到的 SRL308 中所标识的另一组的成员。同样,如果用户是该组的成员,则该要求就可兑现,且假设满足了适用于该兑现的所有其它条件。

[0144] 通过高速缓存信息来确定用户组成员关系

[0145] 正如以上所讨论的,在根据 SRL308 向用户发行许可证 1208 的过程中,DRM 服务器 320 可以通过参考在 DRM 服务器 320 分支中的目录 1206 和在其它分支中的目录来确定用户是否是由 SRL308 所标识组的成员。应该理解的是,每次参考目录 1206 产生确定都变得相当昂贵和麻烦,特别是在目录 1206 执行所需要的附加网络通话量和工作方面,以及特别是如果所确定的数量变得相当高时。

[0146] 因此,在本发明的一个实施例中,从一个和多个目录 1206 所产生的用户一组信息可高速缓存在一个和多个可访问 DRM 服务器 320 的位置上。另外,在本发明的一个实施例中,DRM 服务器 320 在产生用户一组确定时可采用最低成本的第一算法,以便于在参考目录 1206 之前按从最低成本至最高成本的顺序来审查多个高速缓存的位置。于是,只要有可能,较高成本的操作可以推迟,确定的速度也可以提高,并且网络的通话量也可降低。

[0147] 高速缓存用户一组信息

[0148] 在从目录 1206 中来确定用户是否是一个组的一个成员的过程中,正如结合图 2 所讨论的那样,DRM 服务器 320 可潜在地放置于至少一些用户是其直接或间接成员的其它组中。值得注意的是,因为一旦事实上发现了这一成员关系路径便停止往返的移动时,就可能不能发现所有的用户组。例如,如果在检查正在讨论组的记录目标过程中,发现了用户是该组的一个直接成员,则就不会发现用户是其它组的直接成员或间接成员。同样,如果试图发

现它们之间的成员关系从用户到组的往返移动是必须的,则至少可发现用户是直接成员的组,但是不能发现一些间接的组,因为事先发现了成员关系路径。不过,在本发明的一个实施例中,DMR 服务器 320 已经在一个或多个高速缓存中存储了有关用户和所有直接或间接组的成员关系,用于以后的参考,如果这是需要的话。

[0149] 现在参考图 14,可以看到高速缓存 1404 的多个条目 1402。特别是,从高速缓存 1404 的一个条目 1402 可以看到,对于用户 3 来说,已经发现该用户 3 是组 4、6 和 7 的直接或间接的一个成员,并且组 4 是组 6 和 7 的一个成员。同样,从高速缓存 1404 的另一个条目 1402 可以看到,对于用户 4 来说,已经发现该用户 4 仅仅只是组 8 的一个成员。该用户 3 和用户 4 和组 4 是否是该组的直接或间接成员可认为是不重要的并因此而不需要进行描述,尽管如果发现有必要和有用的话也可以描述这类信息。适用于用户 / 组的所有组成员关系的内容材料都缓存,并且作为任何特殊的用户一组的相互关系对以后建立相同或不同用户 / 组的用户一组之间的相互关系是非常有用的。

[0150] 正如所理解的那样,在高速缓存 1404 中的信息从目录 1206 获得之后的一段时间之后就变成了“过时的”。也就是说,在一段时间之后,用户就可以从目录中删除,就可以定义其它组和组的成员关系,以及其它等等,高速缓存 1404 并不能自动更新来反映所变化的信息。因此,正如图 14 所示,高速缓存 1404 的各个条目 1402 都包括了一个建立的时间。基于这一建立的时间,任何讯问高速缓存 1404 的讯问条目都能够确定该条目 1404 是否太旧而不可信。另外和在另一些实施例中,高速缓存 1404 也可以由它自己根据其所建立的时间来删除其中被认为太旧的各个条目 1404。

[0151] 现在,在参考图 15 和图 16,可以发现图 14 所示的高速缓存可以采用多种结构来实施。在一种结构中,在分支中的各个许可证发行的 DRM 服务器 320 都具有一个高速缓存 1404,这是相对较轻的存储器方式的高速缓存,并且该分支也具有一个包含高速缓存 1404 的专用的高速缓存服务器 1502(图 15),这是一个相对扩展重量较重存储器方式的高速缓存。此外,专用缓存服务器 1502 可以采用数据库 1404,和 / 或可以代表许可证发行 DRM 服务器 320 执行对目录 1206 的讯问。

[0152] 在另一种结构中,在分支中的各个许可证发行 DRM 服务器 320 是相对扩展的重量较重的存储器方式的高速缓存(图 16),并且可执行对目录 1206 的讯问。尽管数据库高速缓存 1404 可以在分支中的所有许可证发行 DRM 服务器 320 中共享数据库高速缓存 1404,但是并没有提供专用的高速缓存服务器 1502。在任何结构中,各个基本的高速缓存 1404 都仅仅只具有用户信息,而各个扩展的高速缓存才具有用户和组信息。

[0153] 最低成本第一算法

[0154] 与高速缓存 1404 的结构无关,DRM 服务器 320 寻求确定用户是否是组的一个成员,采用了本发明的最低成本第一算法,以从最低成本到最高成本的顺序来搜索高速缓存 1404 和目录 1206。可推测,与 DRM 服务器 320 有关联的高速缓存 1404 具有与此相关的最低成本,该高速缓存 1404 与专用高速缓存 1502 有关并且数据库高速缓存 1404 具有与此相关的较高成本,以及目录 1206 具有与此相关的最高成本。当然,成本可以是主观的,并且在任何比率上,一个特殊的条目都可以具有任何特殊的成本,这些都没有脱离本发明的精神和范围。

[0155] 现在参考图 17A 和 17B,可以看到该算法的主要特性是在相信完成了所有具有最低成本的操作之前并不试图进行任何具有较高成本的操作。一般来说,以增加成本的次序

的操作可以在发许可证的 DRM 服务器 320 的存储器方式的高速缓存中找到,可以在数据库高速缓存 1404 中或者在专用高速缓存服务器 1502 中找到,或者在 AD 中找到。在执行算法的过程中,提交一个用户和一个或多个目标组,并且确定该用户是否是任何目标组的一个直接或间接的成员。基本上,该算法利用从高速缓存 1404 和目录 1206 中得到的所有有效成员关系信息从用户往返移动到任何目标组,在执行较高成本的操作之前始终执行着较低成本的操作。如果用户是一个目标组的一个成员,则该算法就返回“是”作为结果,否则如果用户不是任何目标组的成员,则就返回“否”作为结果,如果用户不在目录 1206 中则就返回“用户没有发现”,如果在目录 1206 中没有发行任何目标组则就返回“没有发现任何目标组”。

[0156] 正如图 17A 所示,通过定义正在讨论的用户和目标组和多个目标组来开始处理过程(步骤 1701)。假定在某一瞬间,DRM 服务器 320 具有一个本地的高速缓存 1404 以及能够访问远程高速缓存 1404(这可以是专用高速缓存 1502 或者是数据库缓存 1404)和目录 1206,则由许可证发行 DRM 服务器 320 所采用的算法使用了多个箱和队列,包括:

[0157] 一个目标箱 (T),在该箱中可驻留目标组或者多个目标组;

[0158] 一个确认的目标箱 (V),在该箱中将驻留已经得到确认的目标组或者多个目标组;

[0159] 一个搜索箱 (S),在该箱中将驻留多个等待着搜索的组;

[0160] 一个本地队列 (Q1),在该队列中将驻留在本地高速缓存中等待搜索的组;

[0161] 一个远程队列 (Q2),在该队列中将驻留在远程高速缓存中等待搜索的组;

[0162] 一个目录队列 (Q3),在该队列中将驻留在目录 1206 中等待搜索的组;以及,

[0163] 一个丢弃箱 (D),在该箱中将驻留已经处理过的组。

[0164] 于是,通过对在目标箱中的各个目标组,搜索发现适用于目标组的记录或条目 1402,确认目标组或多个目标组来继续该算法。因为,从基本上来说,对个有关的存储所设备 (S1,相对应于本地高速缓存 1404 ;S2,相对应于远程高速缓存 1404 ;S3,相对应于目录 1602) 都执行相同的处理过程,所以仅仅只需要参考一个通用存储器 Sx 来进行这类处理的讨论。

[0165] 尤其,该算法首先在 S1,本级高速缓存 1404 中搜索在目标箱 (T) 中各个目标组,如果发现这类目标组是本级高速缓存 1404 中的一个条目 1402,则从 (T) 中去除该目标组,并且将该目标组放置于确认目标箱 (V) 中,它的条目 1402 也放置于任何较低等级的高速缓存 1404 中,这在这种情况下是不存在的(步骤 1703)。之后,该算法对 S2,远程高速缓存 1404 以及对在 (T) 中所保留的各个目标组重复步骤 1703,如果发现了这类目标组作为在远程高速缓存 1404 中的一个条目 1402,则从 (T) 中去除该目标组,并且将该目标组放置于 (V) 中以及将条目 1402 放置于任何较低等级的高速缓存 1404 中,这时,该缓存是本地高速缓存 1404。最后,该算法再对 S3,目录 1602 以及对仍保留在 (T) 中的各个目标组重复步骤 1703,如果发现了这类目标组作为目录 1602 中的一个记录,则从 (T) 中去除该目标组,并且将该目标组放置于 (V) 以及将该记录 1602 放置于任何较低等级的高速缓存 1404,这时,该缓存是远程高速缓存 1404。

[0166] 一旦对各个存储设备执行了步骤 1703,则确认目标箱 (V) 就应该包含已经发现在高速缓存 1404 或在目录 1602 中存在的所有目标组。随后,所有保留在目标箱中的目标组可

以假定为将是坏的或者不再存在的,并因此而被忽略。如果在这种情况下,(V) 是空的,则结束该算法,认为不存在确认的目标组,并且该算法可返回“没有发现目标组”(步骤 1705)。应该注意的是,通过从目录 1602 或远程高速缓存 1404 将一个记录或条目拷贝到所有较低等级的高速缓存 1404 中,这类较低等级的高速缓存 1404 就变成充满了缓存的信息,这些信息可以用于以后确定用户是否是组的一个成员。

[0167] 之后,假定在 (V) 中存在着至少一个被确认的目标组。特别是,该算法搜索适用于正在讨论用户的记录或条目 1402(步骤 1709),首先在 S1 中寻找,S1 是本地高速缓存 1404,随后在 S2,它是远程高速缓存 1404,接着在 S3,它是目录。一旦以步骤 1709 的任何迭代方式发现了适用于正在讨论用户的记录或条目 1402(这里称之为“项目”),该算法就将用户是一个成员的各个组放置于搜索箱 (S)(步骤 1711),并且采用适用于用户的一个条目 1402 来装入所有较低等级的高速缓存 1404(步骤 1713)。值得注意的是,如果从步骤 1709 的任何迭代中并没有发现适用于用户的项目,就结束该算法,表示不存在确认的用户,并且算法返回“没有发现用户”(步骤 1715)。

[0168] 正如所能理解的那样,步骤 1709 是以相对应于从最低成本操作到最高成本的顺序的操作进行迭代的,这在这种情况下迭代的次序是 S1。随后 S2,随后 S3。因此,一旦确认了用户,就可避免最高成本的操作。值得注意的是,可以在目标组或多个目标组之前来确认用户,但是这并不脱离本发明的精神和范围。

[0169] 假定发现了所确认的用户,是同在 (S) 中这类组所表示的至少一个组的成员,以及发现至少一个所确认的目标组如同在 (V) 中相对应的项目所表示的,现在参考图 17B,可以由 (S) 和 (V) 来确定在 (S) 中任何组是否与在 (V) 中的任何组相匹配来继续该算法(即,所确认的目标组和各个目标组)(步骤 1723)。如果确实发现了在 (S) 中的一个组与在 (V) 中一个项目相匹配,则该算法就结束,表示一个匹配的结果,并该算法返回“是”(步骤 1725)。否则该算法就通过将所有在 (S) 中的组都移至本地队列 (Q1) 中来继续(步骤 1727)。

[0170] 之后,该算法进行迭代处理过程,以搜索从用户到任何目标组的成员关系路径。该迭代处理要求在本地和远程高速缓存 1404 和目录 1206 中进行多种搜索,以确定组的成员关系。因为基本上是对各个队列 (Q1, 相对应于本地高速缓存 1404 ;Q2, 相对应于远程高速缓存 1404 ;以及 Q3, 相对应于目录 1206) 进行相同的处理过程,所以这类处理仅仅只需要参考通用的 (Qx) 和通用存储器 Sx,正如图 17B 所示,其中, S1 是本地高速缓存 1404 ;S2 是远程高速缓存 1404 ;以及 S3 是目录 1206。

[0171] 特别是,假定在某一瞬间,图 17B 中的 (Qx) 和 Sx 是 (Q1) 和 S1,本地高速缓存 1404,首先是确定 (Q1) 是否是空的(步骤 1729)。当然,(Q1) 不应该初始化成为空的,但是因为该处理过程是迭代的,所以就由可能出现所指向的那一 (Q1) 确实是空的。假定 (Q1) 初始不是空的,并且其中具有一个组。这样一个组可以从 (Q1) 中选择出和去除,并且可以确定该组是否已经在丢弃箱 (D) 中(步骤 1731)。最初,当然,从 (Q1) 中选择出和去除的组并不是在丢弃箱 (D) 中,因为该组还没有处理过,但是随着过程进行迭代处理,其它组可以放置于 (Q1),也许可以包括原先已经处理过的组,以及在处理各个组的过程,各组要确实移至 (D)。

[0172] 如果从 (Q1) 中选择出和去除的组已经被处理过并因而在 (D) 中,则算法就返回步

骤 (1729), 在该步骤中, 再次进行确定 (Q1) 是否是空的。然而, 如果从 (Q1) 中选择出和去除的组还没有处理过并因此还没有在 (D) 中, 则算法继续搜索对应于 (Q1) 的高速缓存 1404 或目录 1206 中所选择的组的条目 1402, 这时, 该缓存是本地高速缓存 1404 (步骤 1733)。

[0173] 如果在本地高速缓存 1404 中没有发现用于所选择组的这类条目 1402, 则处理就继续将该组从 (Q1) 移至远程高速缓存 (Q2) (步骤 1735), 并且随后再返回至步骤 (1729), 在该步骤中再次确定 (Q1) 是否是空的。然而, 如果在本地高速缓存 1404 中发现了用于所选择组的这样一个条目 1402, 则算法就将该各个新发现的且所选择组是其成员的组放置于搜索箱 (S) 中, 并且将所选择的组移动至丢弃箱 (D) (步骤 1739), 另外, 用对应于所选择组的一个条目 1402 来装入所有较低等级的高速缓存 1404 (步骤 1737)。当然, 值得注意的是, 对于 (Q1) 和本地高速缓存 1404, 并不存在这类较低等级的高速缓存 1404。之后, 算法通过返回步骤 1723 继续, 在该步骤, 确定在 (S) 中的任何组 (即, 新发现的组 (多个组)) 是否与在 (V) 中的任何组 (即, 所确认的目标组或多个组) 相匹配。当然, 如果确认发现在 (S) 中有一个组与在 (V) 中的一个项目相匹配, 则算法就以匹配而结束, 并且算法返回“是”, 正如步骤 1725 所示。否则, 算法就继续, 在步骤 1727 将在 S 中的所有组都移至本地队列 (Q1), 并且继续算法。

[0174] 在这样做的过程中, 算法利用 (Q1) 使得所有从用户到任何所确认的目标组的可能路径都扩展到本地高速缓存 1404 的可能范围。正如现在所应理解的那样, 假定没有发现匹配, 则 (Q1) 随着处理过程的迭代采用最新发现的组来填补, 并且在 (Q1) 中的所有组都利用本地高速缓存 1404 进行处理之后清空至丢弃箱 (D) 或者利用远程高速缓存 1404 移至待处理的 (Q2)。于是, 被认为是最昂贵的远程高速缓存 1404 的操作不会发生, 直至在本地高速缓存 1404 中的所有可能的操作都已完成为止, 如同在步骤 1729 发现 (Q1) 是空的时候所发生的那样。

[0175] 当在步骤 1729 发现 (Q1) 事实上是空的时候, 处理过程就返回到远程队列 (Q2), 其方式类似于对 (Q1) 所进行操作的方式。特别是, 现在假定图 17B 所示的 (Qx) 和 Sx 是 (Q2) 和 S2, 远程高速缓存 1404, 首先确定 (Q2) 是否是空的 (步骤 1729)。这里, (Q2) 可能最初不是空的, 虽然在本地队列 1404 能够自己处理来自 (Q1) 的所有组的情况下有可能发生这类事件。假定 (Q2) 最初不是空的, 那么就具有一个组, 这样一个组从 (Q2) 中选择和去除, 并且确定该组是否因为已经处理过了而处在丢弃箱 (D) 中 (步骤 1731)。

[0176] 如果从 (Q2) 中选择和去除的组已经处理了并且已经在 (D) 中, 则算法就返回至步骤 (1729), 在该步骤中, 再次确定 (Q2) 是否是空的。然而, 如果从 (Q2) 中所选择和去除的组还没有处理并且还不在于 (D) 中, 则算法就继续搜索适用于对应 (Q2) 的高速缓存 1404 和目录 1206 中所选择组的一个条目 1402, 这时, 该缓存是远程高速缓存 1404 (步骤 1733)。

[0177] 如果在远程高速缓存 1404 中并没有发现适用于所选择组的这类条目 1402, 则算法就继续将 (Q2) 中的组移至目录队列 (Q3) (步骤 1735), 并且再返回至步骤 (1729), 在该步骤中, 再次确定 (Q2) 是否是空的。然而, 如果在远程高速缓存 1404 中发现了适用于所选择组的这类条目 1402, 则算法就将各个新发现的且所选择的组是其成员的组放置于搜索箱 (S) 中, 并且将所选择的组移动至丢弃箱 (D) (步骤 1739), 另外, 采用对应于所选择组的一个条目 1402 来装入所有较低等级的高速缓存 1404 (步骤 1737)。当然, 值得注意的是, 对于 (Q2) 和远程高速缓存 1404 来说, 仅仅只需要装入本地高速缓存 1404。之后, 算法就

返回至步骤 1723 继续,在该步骤,确定在 (S) 中的任何组 (即,新发现的组和多个组) 是否与在 (V) 中的任何组 (即,所确认的目标组和多个目标组) 相匹配。当然,如果确实发现在 (S) 中的一个组与在 (V) 中的一个项目相匹配,则算法就以一个匹配来结束,并且算法在步骤 1725 返回“是”。否则,算法就继续在步骤 1727 将 (S) 中的所有组移至本地队列 (Q1),并且算法继续回过来处理 (Q1)。

[0178] 在这样做的过程中,算法利用 (Q2) 使得所有从用户到任何所确认的目标组的可能路径都扩展到本地高速缓存 1404 的可能范围。正如现在所应理解的那样,假定没有发现匹配,则 (Q2) 随着处理过程的迭代采用最新发现的组来填补,并且在 (Q2) 中的所有组都利用本地高速缓存 1404 进行处理之后清空至丢弃箱 (D) 或者利用远程高速缓存 1404 移动至待处理的 (Q3)。于是,被认为是较昂贵的在目录 1206 中的操作,在远程高速缓存 1404 中的所有可能的操作都已完成之前并没有发生,如同发现 (Q2) 是空的时候所发生的那样。

[0179] 很显然,利用 (Q2) 和远程高速缓存 1404 新发现的任何组都可以先利用 (Q1) 和本地高速缓存 1404 进行处理。于是,被认为是较昂贵的在远程高速缓存 1404 中的相对新发现组的操作,在完成了对本地高速缓存 1404 的所有可能操作之前,不会发生,同当 (Q1) 是空时所发生的那样。

[0180] 当在步骤 1729 发现 (Q2) 事实上是空的时候,处理过程就返回到目录队列 (Q3),其方式类似于对 (Q1) 和 (Q2) 所进行操作的方式。特别是,现在假定图 17B 所示的 (Qx) 和 Sx 是 (Q3) 和 S3,目录 1206,首先确定 (Q3) 是否是空的 (步骤 1729)。这里,同样,(Q3) 可能最初不是空的,虽然在本地和远程队列 1404 能够自己处理来自 (Q1) 和 (Q2) 的所有组的情况下有可能发生这类事件。假定 (Q3) 最初不是空的,那么就具有一个组,这就从 (Q3) 中选择和去除一个组,并且确定该组是否因为已经处理过了而处在丢弃箱 (D) 中 (步骤 1731)。

[0181] 如果从 (Q3) 中选择和去除的组已经处理了并且已经在 (D) 中,则算法就返回至步骤 (1729),在该步骤中,再次确定 (Q3) 是否是空的。然而,如果从 (Q3) 中所选择和去除的组还没有处理并且还不在于 (D) 中,则算法就继续搜索适用于对应 (Q3) 的高速缓存 1404 和目录 1206 中所选择组的一个条目 1402,这时,该缓存是目录 1206 (步骤 1733)。

[0182] 如果在目录 1206 中并没有发现适用于所选择组的这类条目 1402,则算法就继续将所选择的组从 (Q3) 移至 (D) (步骤 1735),并且再返回至步骤 (1729),在该步骤中,再次确定 (Q3) 是否是空的。值得注意的是,由于在 (Q3) 之后没有下一个队列,所以被选择的组就不能再作进一步的处理。然而,如果在远程高速缓存 1404 中发现了适用于所选择组的这类条目 1402,则算法就将各个新发现的且所选择的组是其成员的组放置于搜索箱 (S) 中,并且将所选择的组移动至丢弃箱 (D) (步骤 1739),另外,采用对应于所选择组的一个条目 1402 来装入所有较低等级的高速缓存 1404 (步骤 1737)。当然,值得注意的是,对于 (Q3) 和目录 1206 来说,仅仅只需要装入本地和远程高速缓存 1404。之后,算法就返回至步骤 1723 继续,在该步骤,确定在 (S) 中的任何组 (即,新发现的组和多个组) 是否与在 (V) 中的任何组 (即,所确认的目标组和多个目标组) 相匹配。当然,如果确实发现在 (S) 中的一个组与在 (V) 中的一个项目相匹配,则算法就以一个匹配来结束,并且算法在步骤 1725 返回“是”。否则,算法就继续在步骤 1727 将 (S) 中的所有组移至本地队列 (Q1),并且算法继续回过来处理 (Q1)。

[0183] 在这样做的过程中,算法利用 (Q3) 使得所有从用户到任何所确认的目标组的可

能路径都扩展到目录 1206 的可能范围。正如现在所应理解的那样,假定没有发现匹配,则 (Q3) 随着处理过程的迭代采用最新发现的组来填补,并且在 (Q3) 中的所有组都利用目录 1206 进行处理之后清空至丢弃箱 (D)。

[0184] 很显然,利用 (Q3) 和目录 1206 所发现的任何组都可以先利用 (Q1) 和本地高速缓存 1404 进行处理。于是,被认为是较昂贵的在目录 1206 中的相对新发现组的操作在完成了对本地和远程高速缓存 1404 的所有可能操作之前,不会发生,如同当 (Q1) 和 (Q2) 是空的时所发生的那样。

[0185] 当在步骤 1729 发现 (Q3) 事实上是空的时候,处理过程就不再进一步继续,因为在这种背景下没有 (Q4),并因此完成了处理。特别是,该算法就在没有匹配的情况下结束,并且算法返回“否”(步骤 1741)。

[0186] 现在应该理解的是,可以将本发明的算法设计成在执行任何较高成本操作之前先执行较低成本的操作。正如所应该理解的那样,虽然是根据三个等级的存储(即,本地和远程高速缓存 1404 和目录 1206)公开的,该算法可以通过定义类似数量的队列来替代应用于任何其它数量等级的存储,包括 2、4、5、6、7 等等。因此,本发明的算法也可以应用于高速缓存用户一组信息的任何多个等级的存储布置,这些都没有脱离本发明的精神和范围。

[0187] 结论

[0188] 能够实现结合本发明所执行的处理过程所需的编程是相对直截了当的,并且在相关的程序设计出版物上应该是很显而易见。因此,这里就不再涉及这类编程。然而,任何特殊的编程都可以应用于实现本发明且并没有脱离本发明的精神和范围。

[0189] 在本发明中,数字版权管理 (DRM) 和实现的结构以及方法都允许对任意形式的数字内容的复制或播放进行控制,其中这种控制是灵活的并且可以由该数字内容的内容所有者/开发商来限定。该结构允许且方便于这种可控的复制,特别是在办公室或组织环境中或类似于文档需要在一个限定的个体组或一类个体之间共享的场合。这类结构允许根据用户是一个组的成员来向用户提供具有许可证的内容,其中,这类用户的成员关系至少是部分根据高速缓存的组信息来确定的。

[0190] 应该理解的是,对上述所讨论的实施例进行的变化都没有脱离本发明的概念。例如,虽然本披露已经讨论了确定从用户到组的组成员关系,但是也可以在不脱离本发明精神和范围的条件下替换成从组到用户的确定。因此,应该理解的是,本发明并不限制于所披露的特殊实施例,但是可试图覆盖在由后附的权利要求所定义的本发明精神和范围内的变更。

计算环境 100

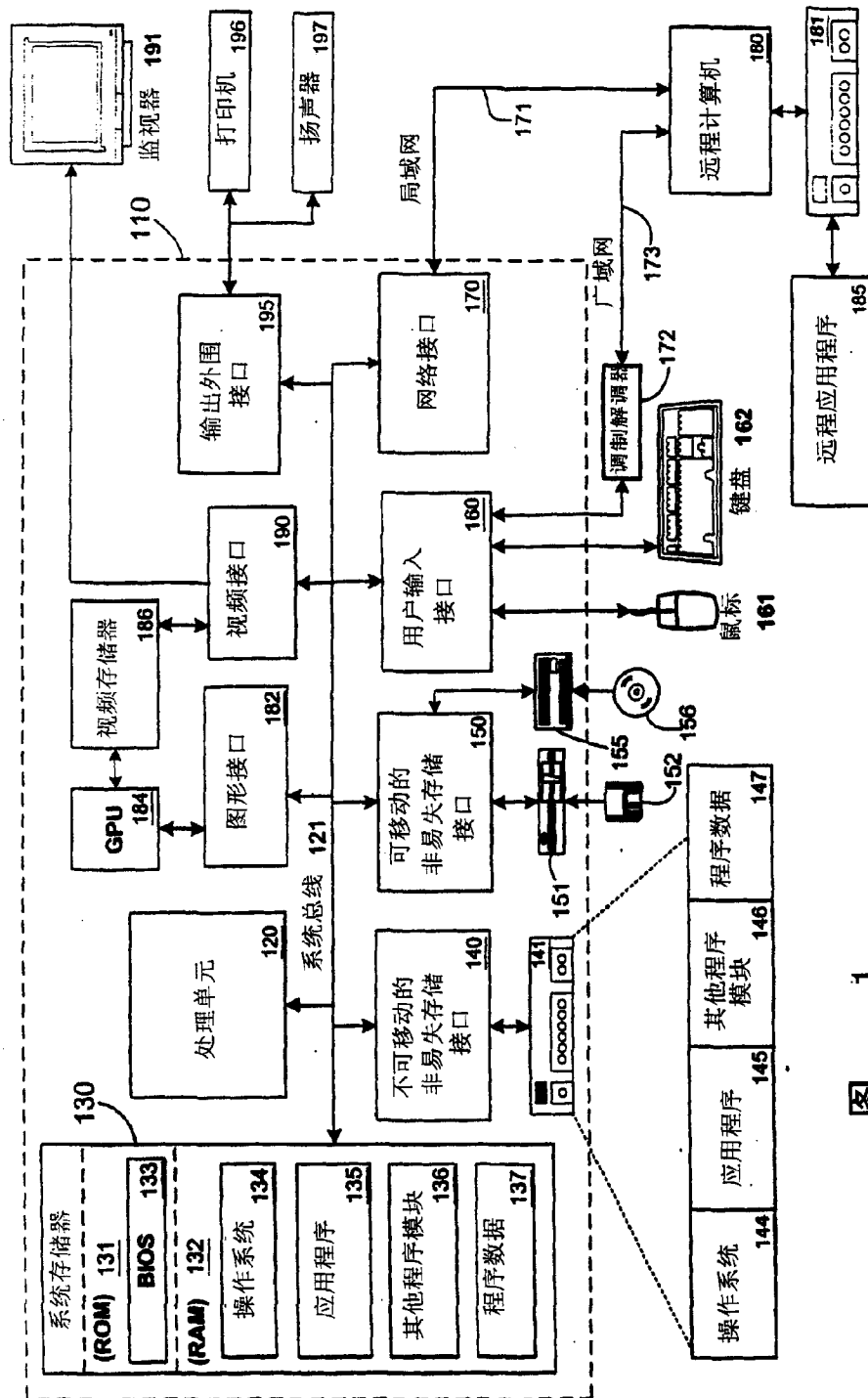


图 1

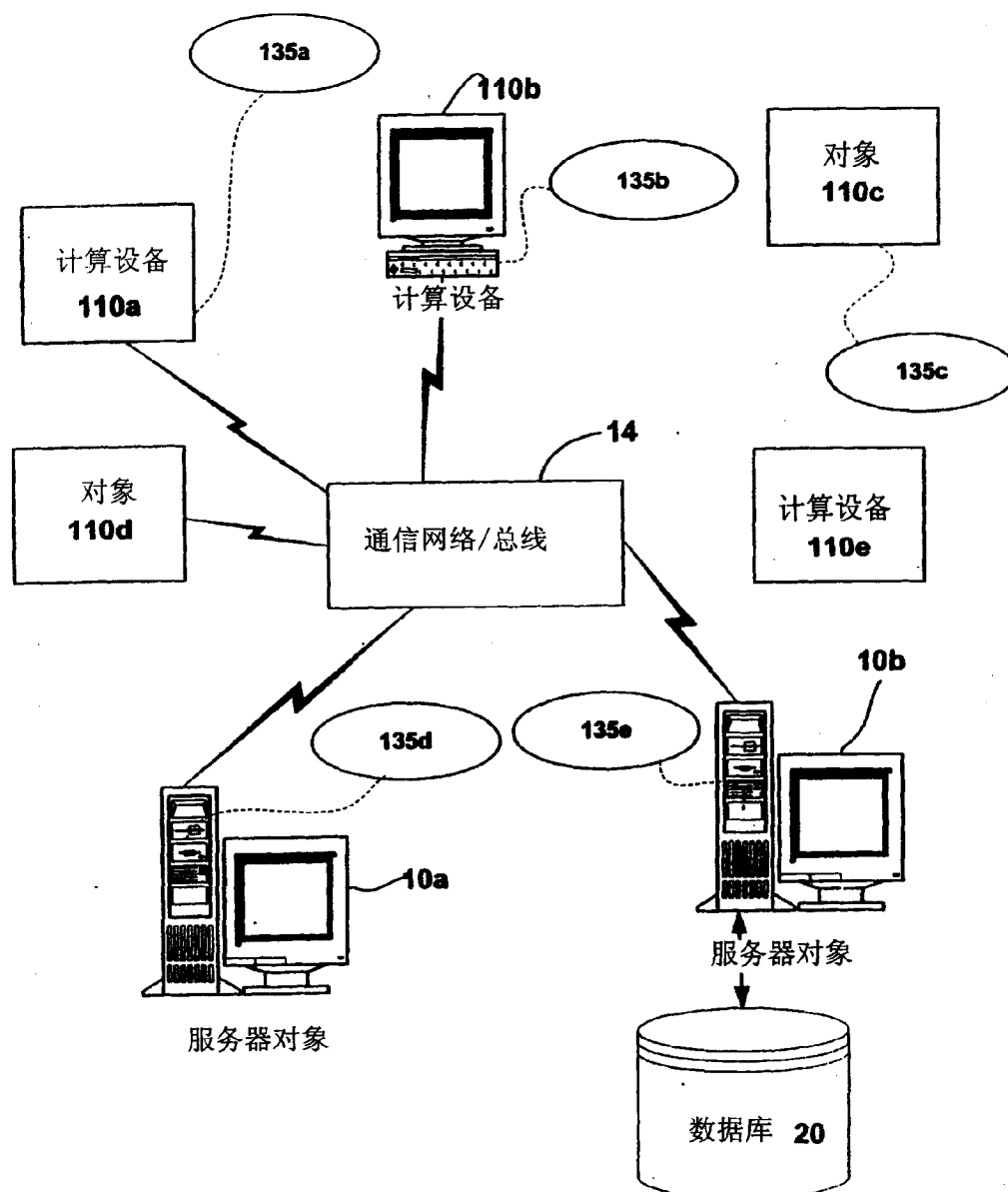


图 2

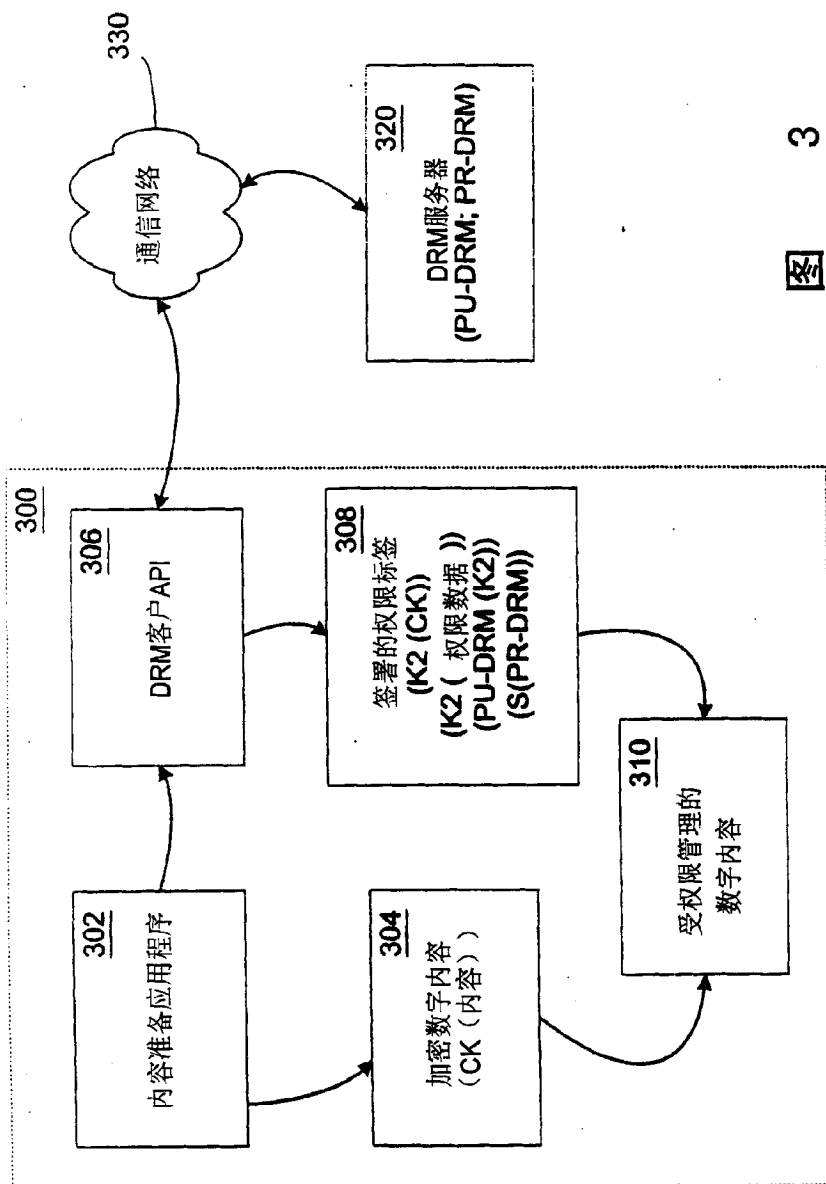


图 3

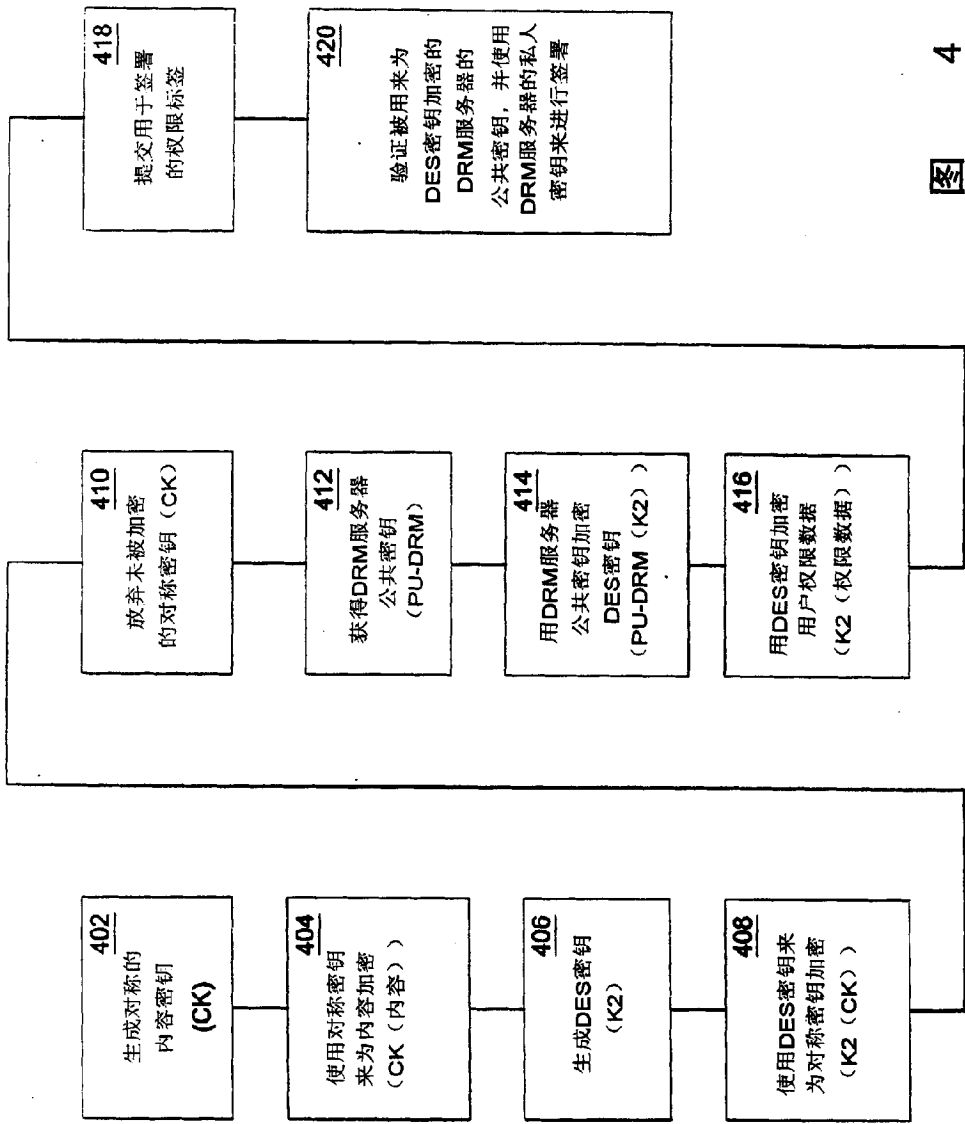


图 4

SRL 308
内容信息
DRM服务器信息
- (PU-DRM(K1))
- 介绍信息
-- URL
-- 后退
权限标签信息
(DES1(权限数据))
(DES1(CK))
S (PR-DRM)

图 4A

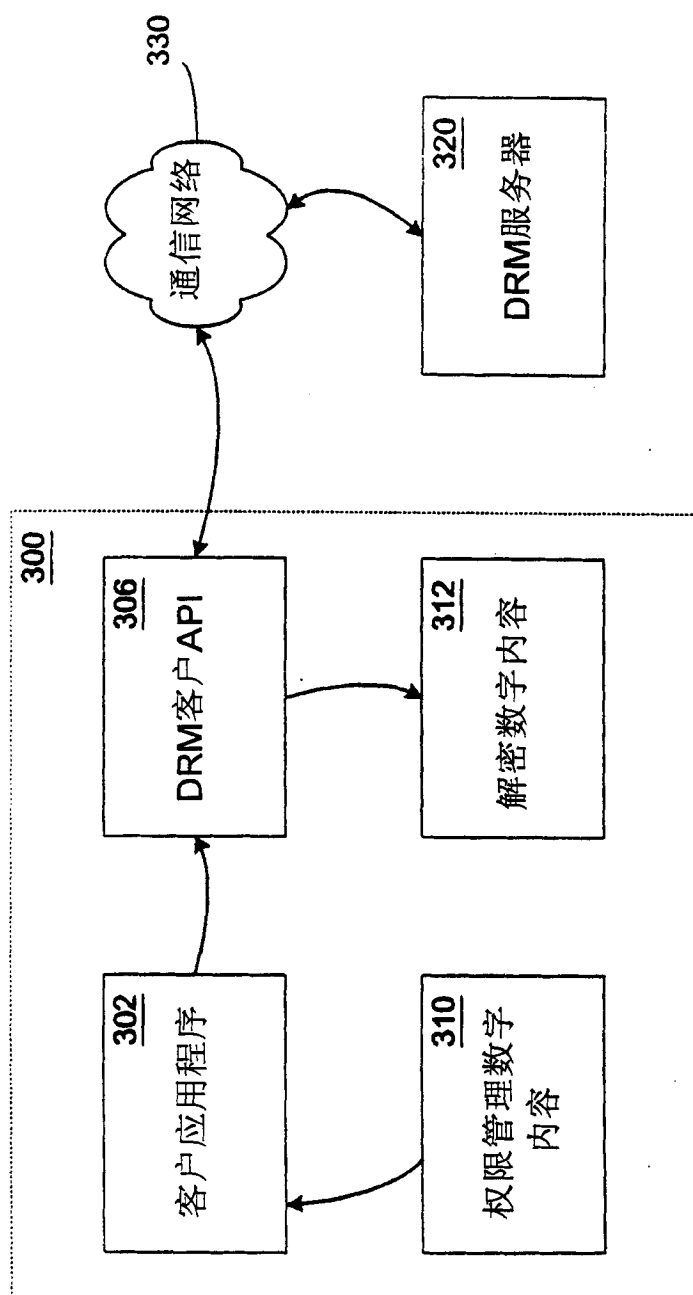
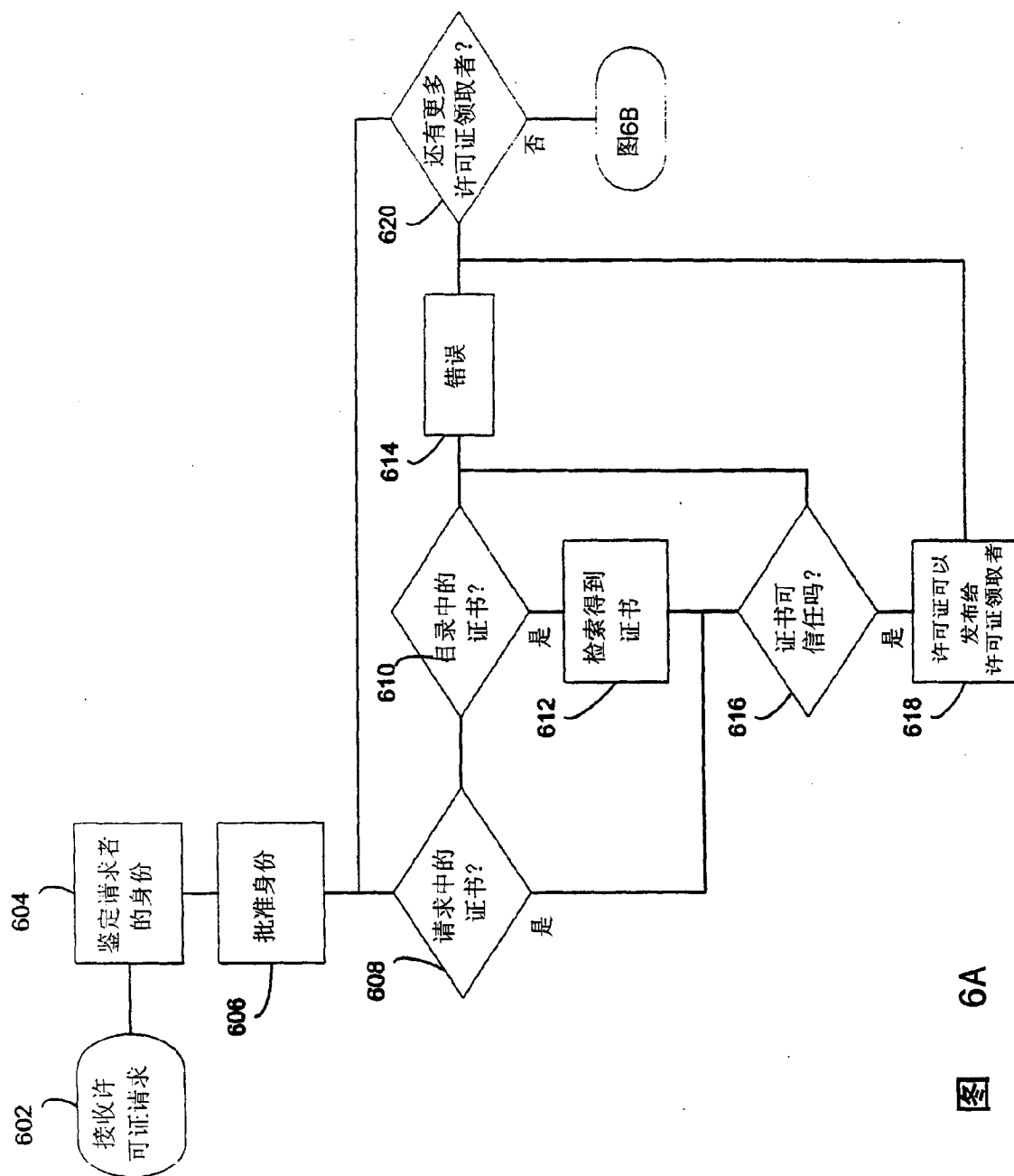


图 5



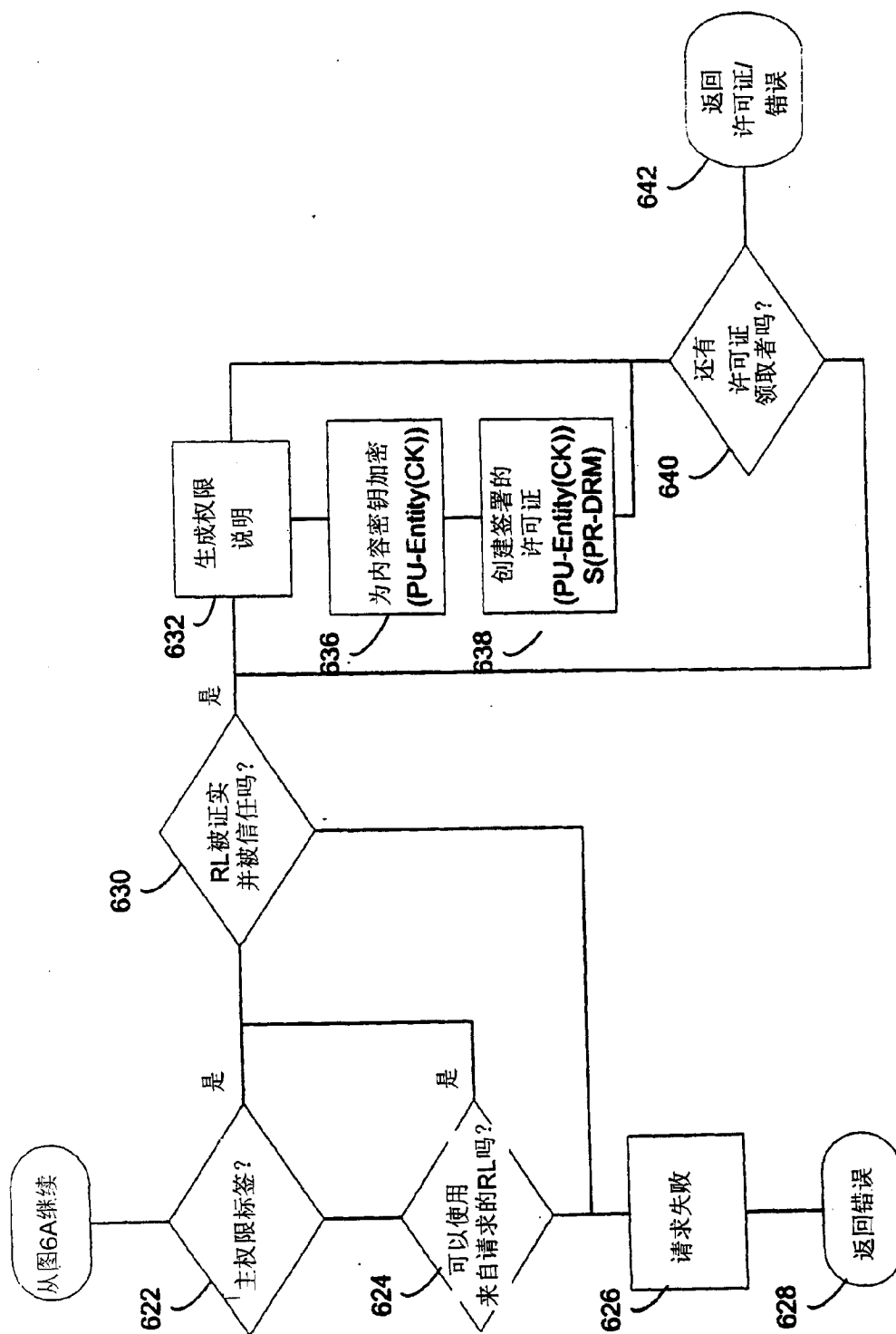


图 6B

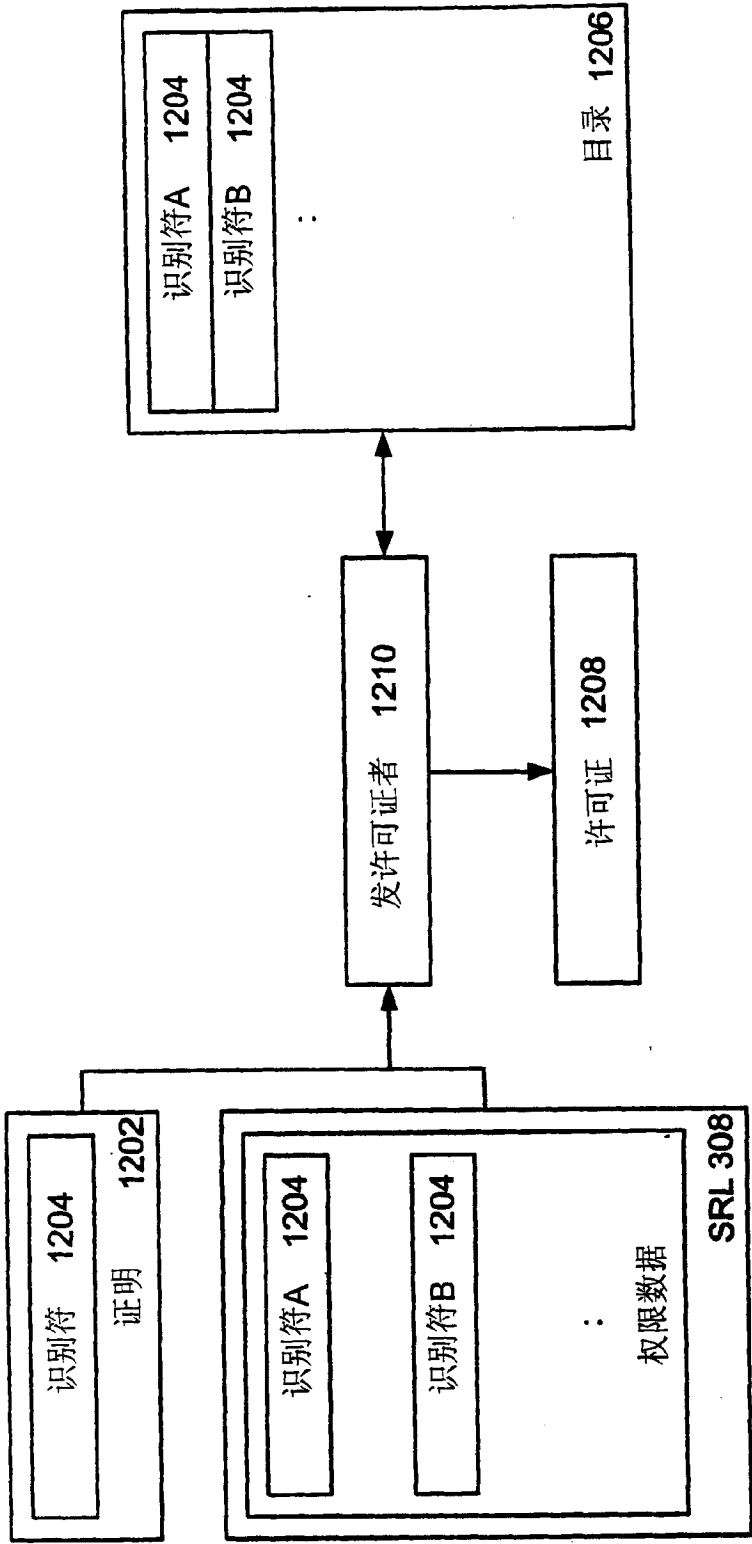


图 7

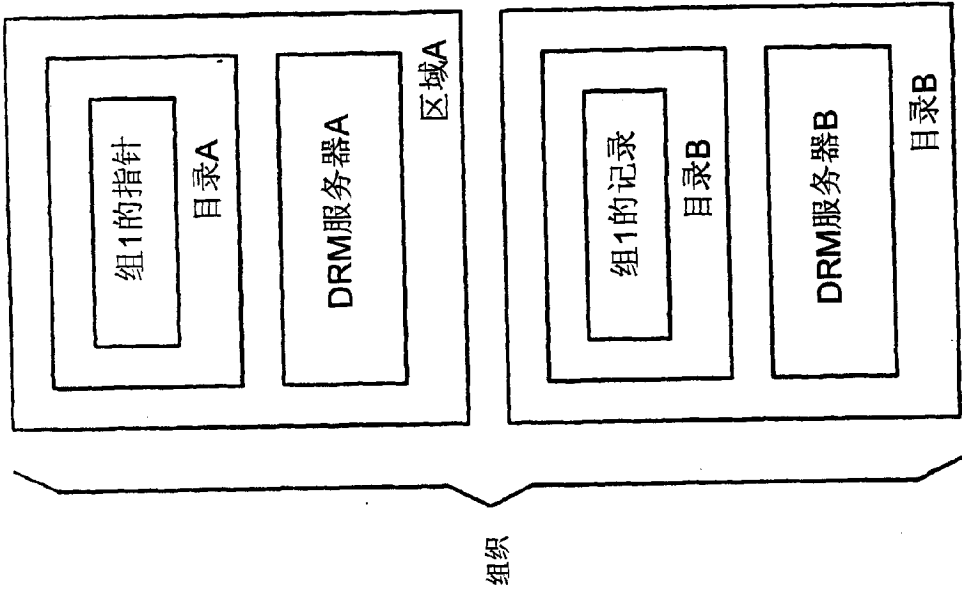


图8

8

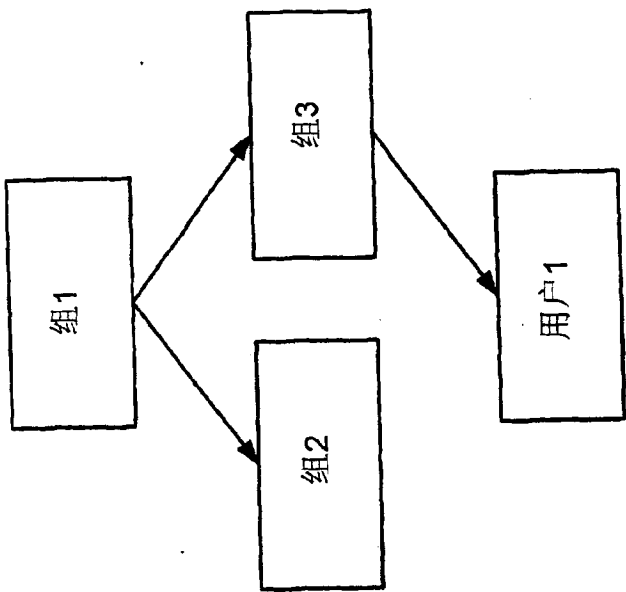


图9

9

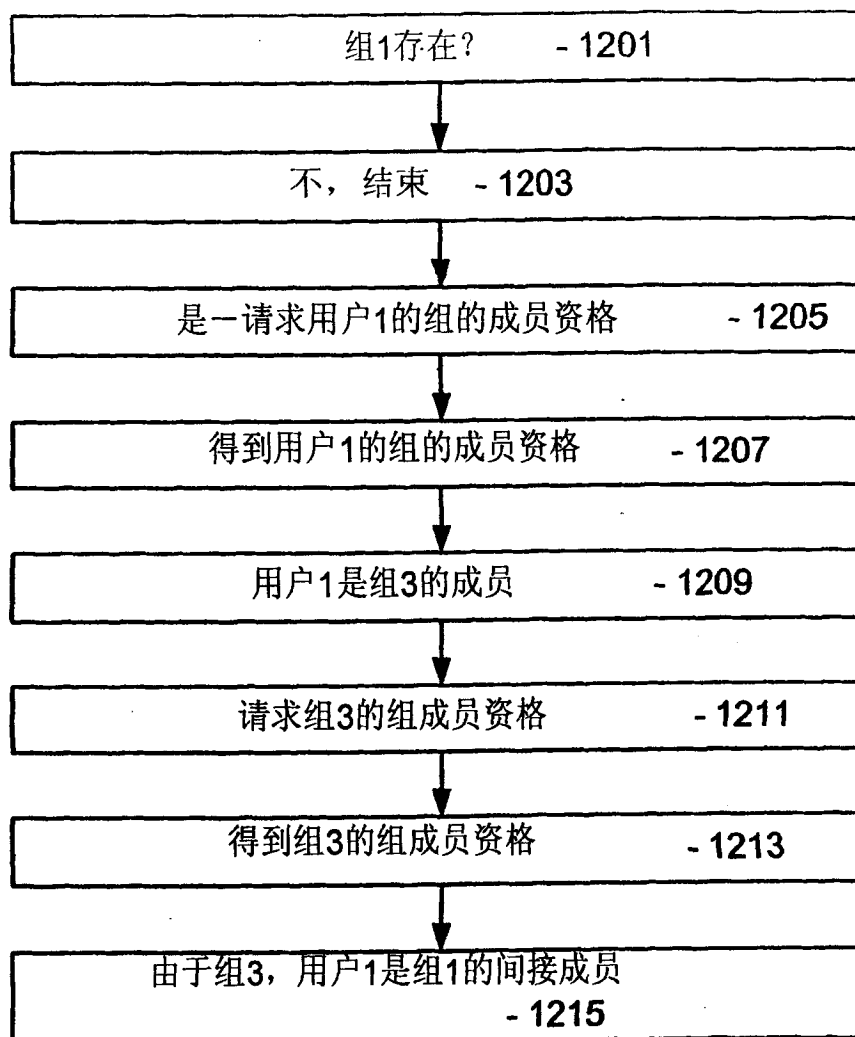


图 12

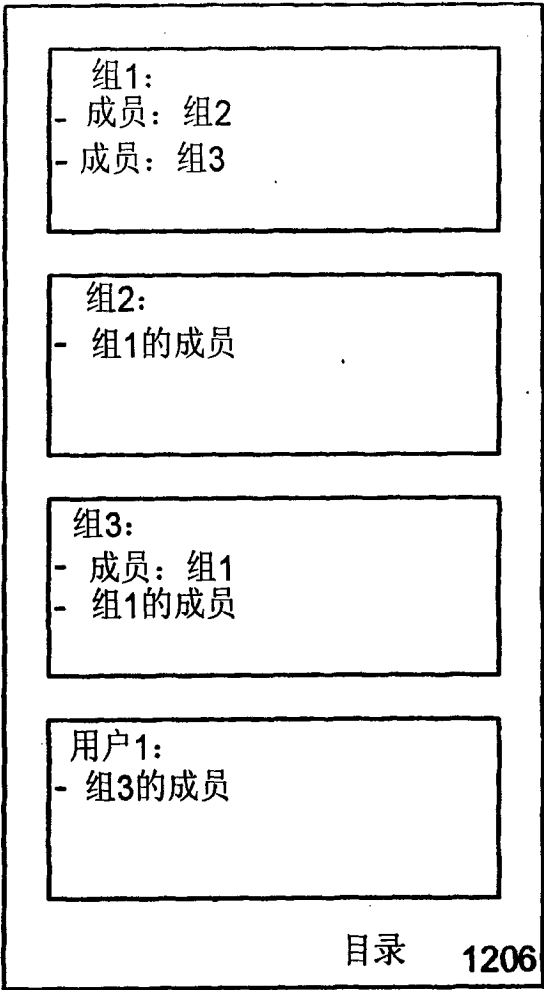


图 10

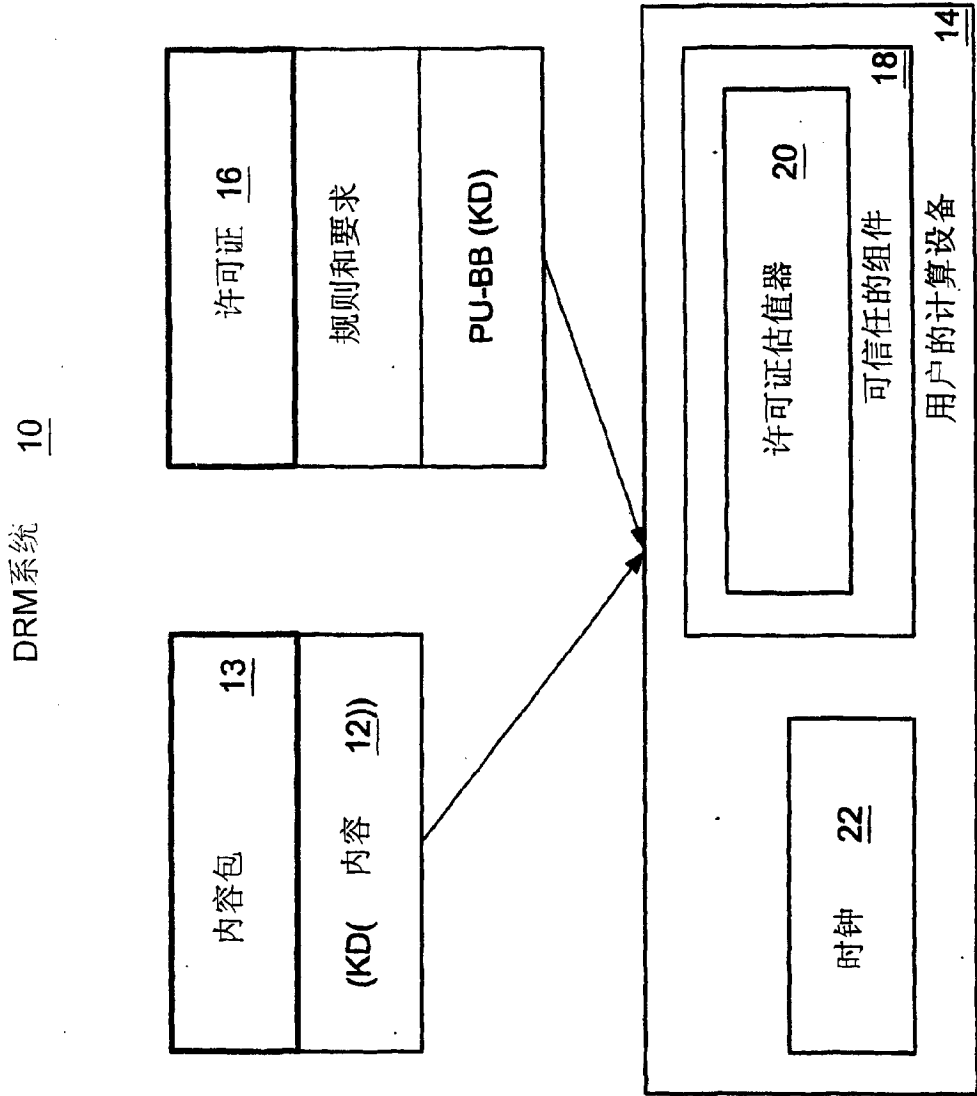
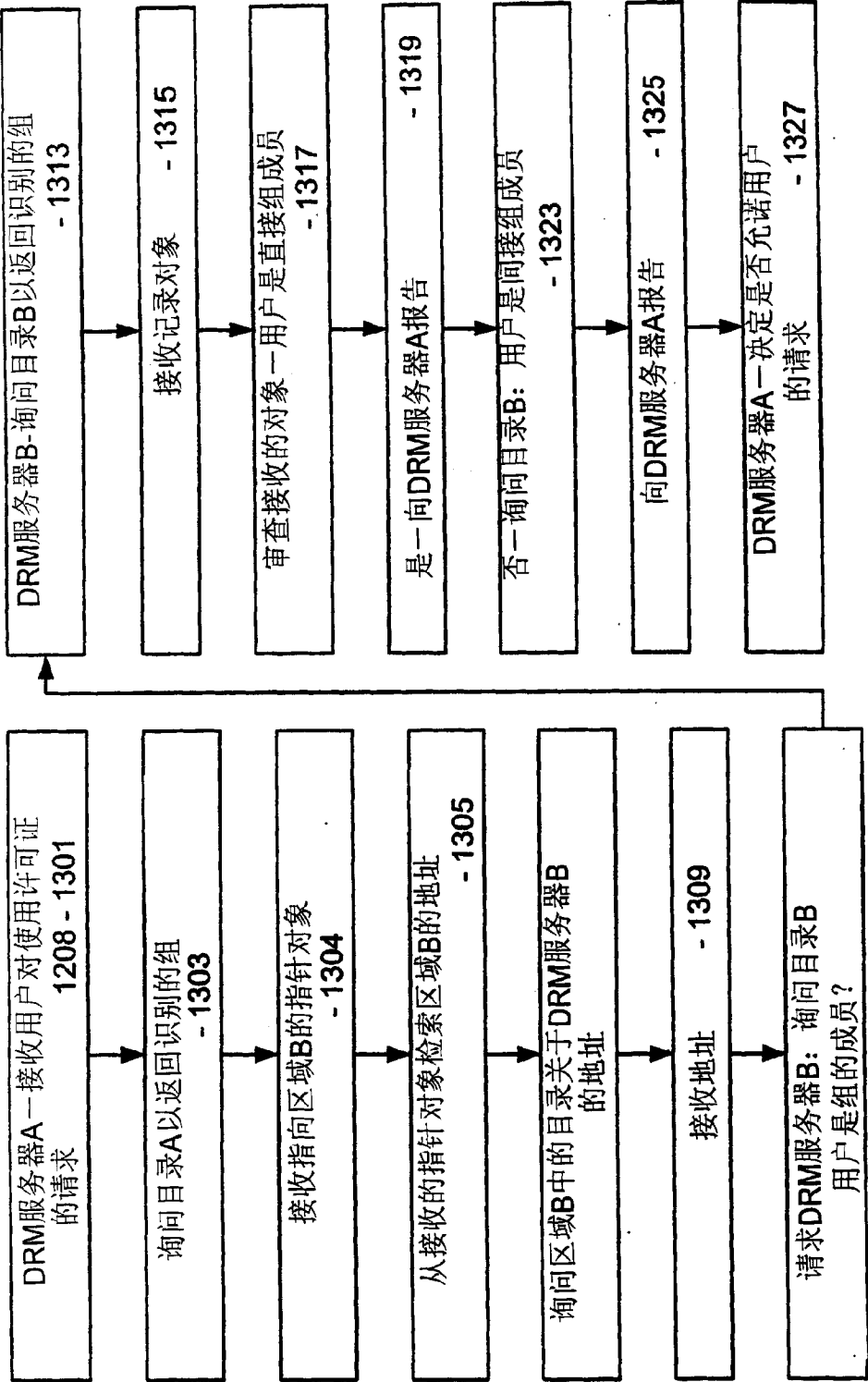


图 11



图

13

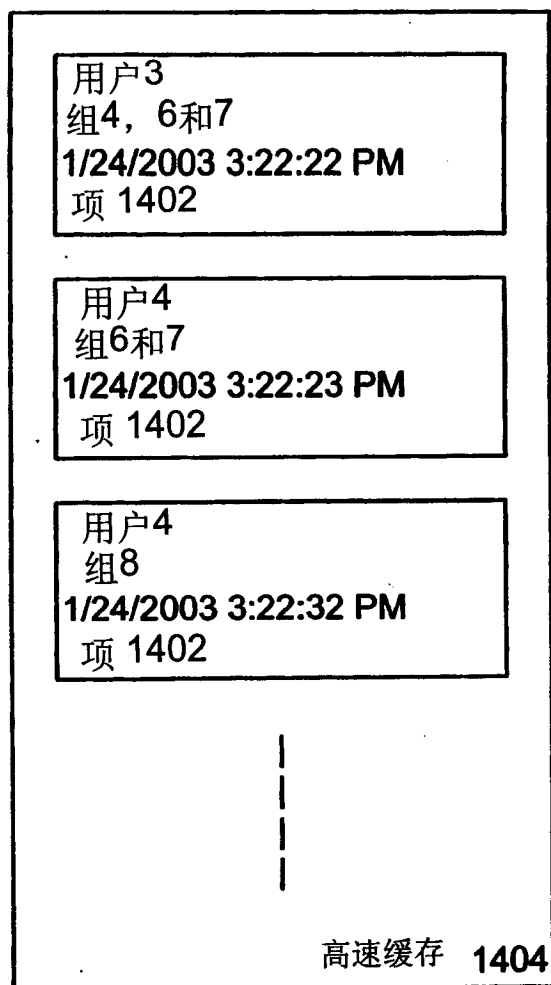


图 14

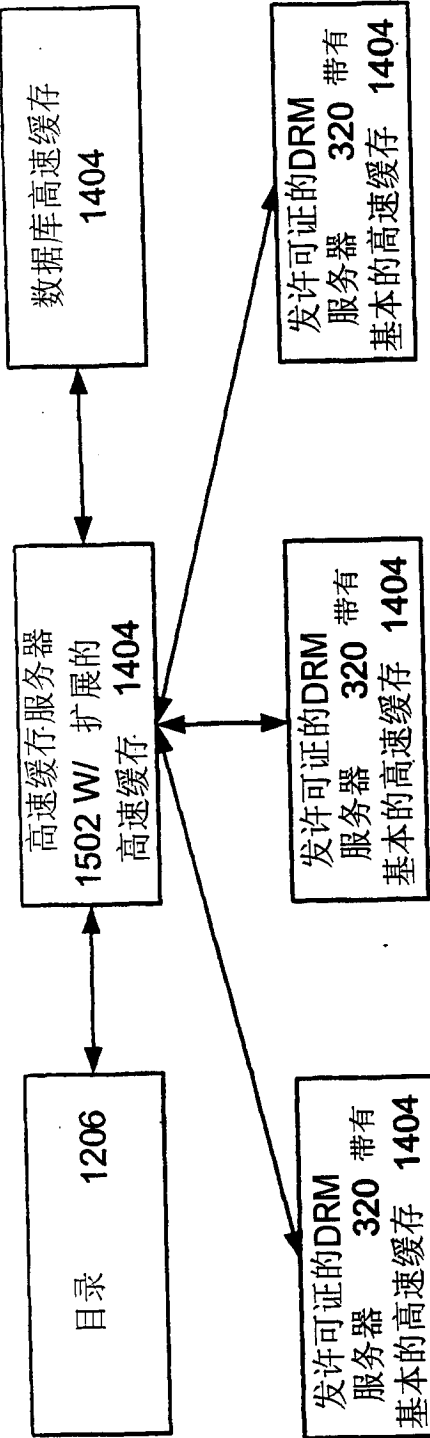


图 15

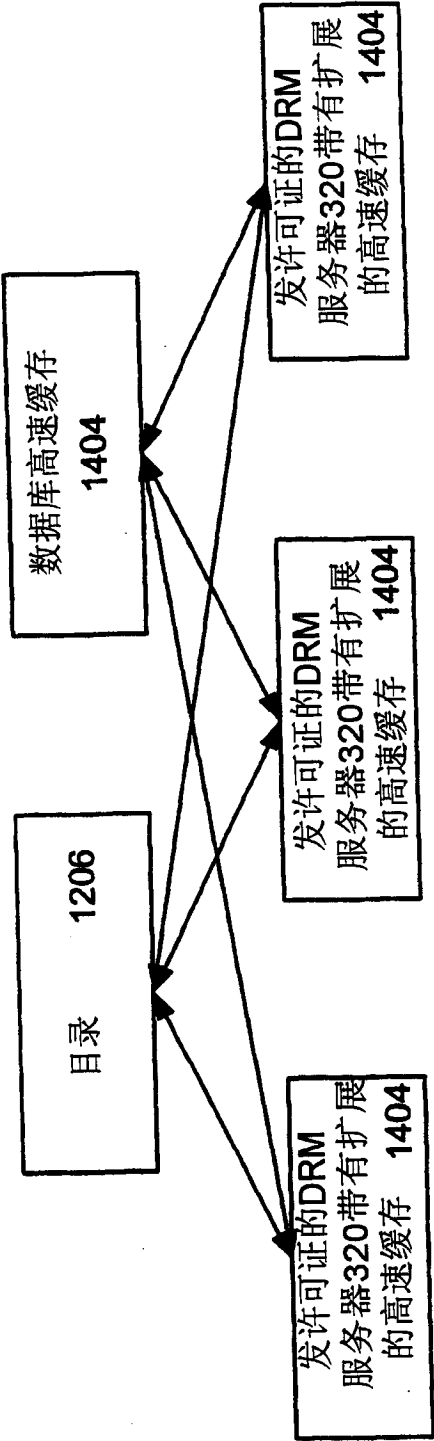


图 16

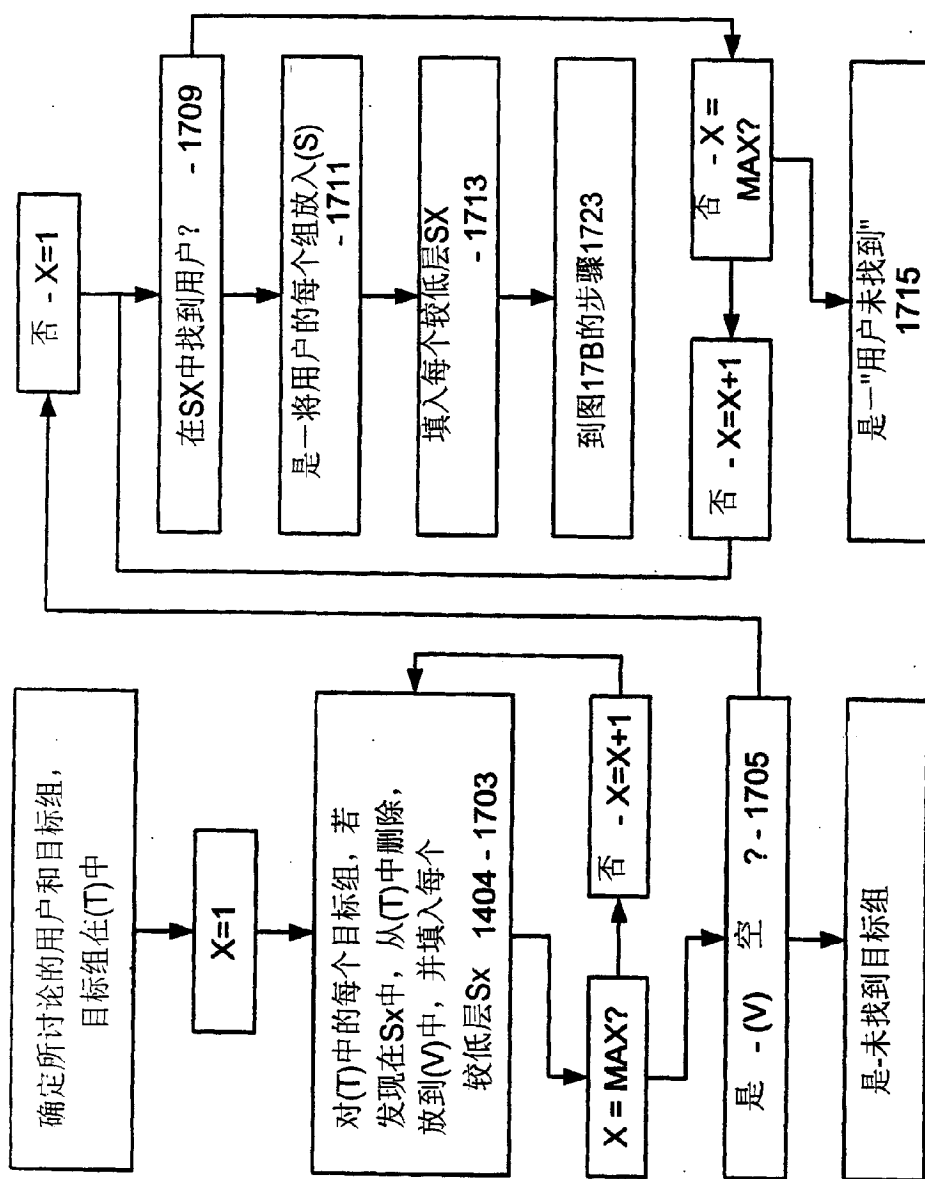


图 17A

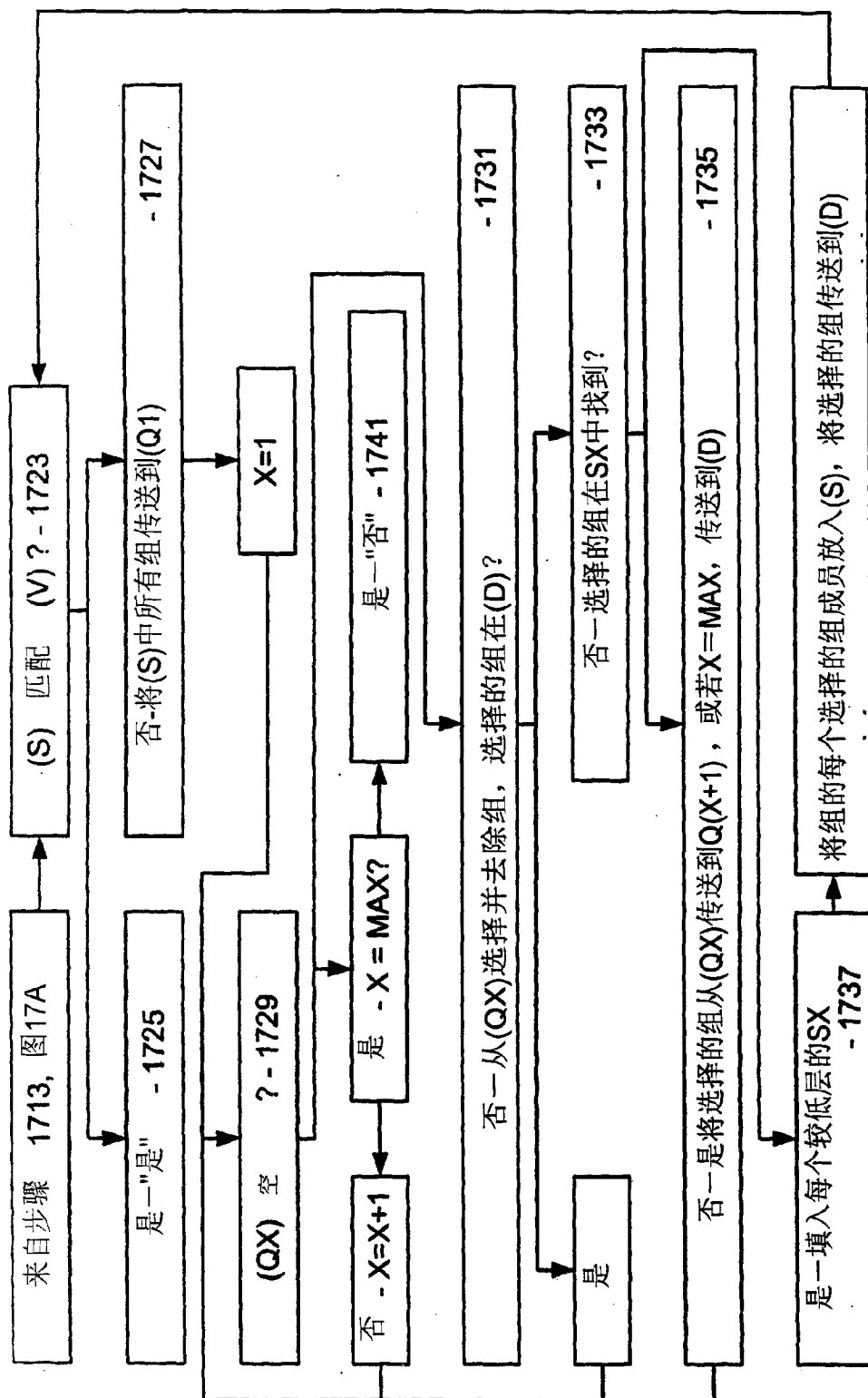


图 17B