

[19] 中华人民共和国国家知识产权局

[51] Int. Cl⁷

G08B 13/24

G06F 17/60 G11B 20/00

G11B 7/00 G11B 7/007

[12] 发明专利申请公开说明书

[21] 申请号 00809032.7

[43] 公开日 2002 年 6 月 26 日

[11] 公开号 CN 1355912A

[22] 申请日 2000.4.18 [21] 申请号 00809032.7

[30] 优先权

[32] 1999.4.21 [33] US [31] 09/295,964

[86] 国际申请 PCT/US00/10414 2000.4.18

[87] 国际公布 WO00/63860 英 2000.10.26

[85] 进入国家阶段日期 2001.12.17

[71] 申请人 研究投资网络公司

地址 美国加利福尼亚州

[72] 发明人 托得·R·科勒特

[74] 专利代理机构 北京康信知识产权代理有限公司

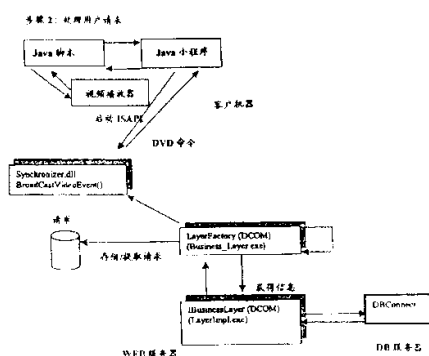
代理人 余刚

权利要求书 4 页 说明书 61 页 附图页数 23 页

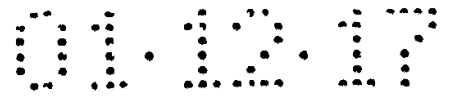
[54] 发明名称 利用激光中心媒体及网络服务器授权使用电子内容的制作的系统、方法及物品

[57] 摘要

本发明提供一种以电子方式跟踪内容配送的制作的系统、方法和物品。首先,一种电子存储媒体跟踪标识符被嵌入到一个电子存储媒体上并存储在一个数据库中。其次,一种包装跟踪标识符被设置在存放所述电子存储媒体的包装中。然后,当所述电子存储媒体在不同实体间传输时,使用包装上的标识符来跟踪所述电子存储媒体。进一步,可以使用电子存储媒体上的标识符来识别所述电子存储媒体,以便提供授权使用包含在电子存储媒体中的信息。



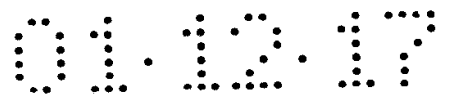
ISSN 1008-4274



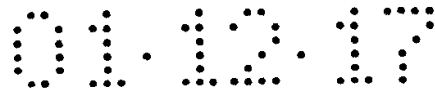
权利要求书

1. 一种以电子方式跟踪内容配送的方法，包含以下步骤：
 - a) 把一个电子存储媒体跟踪标识符嵌入一个电子存储媒体中；
 - b) 把一个包装跟踪标识符嵌入到存放所述电子存储媒体的包装上；
 - c) 将所述跟踪标识符存储到一个数据库；
 - d) 当所述包装连接到一台计算机时检测跟踪信息；
 - e) 将所述跟踪信息传送到一台服务器计算机；以及
 - f) 根据在所述服务器计算机对所述跟踪信息的处理授权从所述包装中重放信息。
2. 根据权利要求1所述的以电子方式跟踪内容配送的方法，其中所述服务器计算机执行表查找来确定一个或多个授权类型。
3. 根据权利要求1所述的以电子方式跟踪内容配送的方法，其中所述服务器计算机执行表查找来确定一个或多个授权标题。
4. 根据权利要求1所述的以电子方式跟踪内容配送的方法，其中所述服务器计算机利用一个来自所述服务器计算机的事项启动对授权信息的授权重放。
5. 根据权利要求1所述的以电子方式跟踪内容配送的方法，其中一个事项被写入一个数据库记录处理。

6. 一种以电子方式跟踪内容配送的装置，包含：
 - a) 一个具有一个脉冲刻写区的光盘电子存储媒体；及
 - b) 一种存储在所述脉冲刻写区的数字代码；
 - c) 所述数字代码代表所述光盘电子存储媒体中内容的标识符；
 - d) 所述装置包含当把电子存储媒体连接到一台计算机时，用以检测跟踪信息的逻辑；
 - e) 所述装置包含将所述跟踪信息传送给一台服务器计算机的逻辑；且
 - f) 所述装置包含如下逻辑：根据在所述服务器计算机对跟踪信息的处理，接收来自所述服务器计算机的授权，从所述电子存储媒体重放信息。
7. 根据权利要求6所述的以电子方式跟踪内容配送的装置，其中所述服务器计算机执行表查找来决定一个或多个授权类型。
8. 根据权利要求6所述的以电子方式跟踪内容配送的装置，其中所述服务器计算机执行表查找来决定一个或多个授权标题。
9. 根据权利要求6所述的以电子方式跟踪内容配送的装置，其中所述服务器计算机利用一个来自所述服务器计算机的事项启动对授权信息的授权重放。
10. 根据权利要求6所述的以电子方式跟踪内容配送的装置，其中一个事项被写入到一个数据库记录处理。



11. 一个包含在一个计算机可读媒体中用来对其上嵌有标识符的电子存储媒体的使用进行识别和提供响应的程序, 所述程序包括:
 - a) 当被用户插入一台计算机时, 读取所述电子存储媒体的标识符的一个程序段;
 - b) 当所述包装与一台计算机连接时检测跟踪信息的一个程序段;
 - c) 将所述跟踪信息传送给一台服务器计算机的一个程序段; 以及
 - d) 根据在服务器计算机对跟踪信息的处理授权从所述包装中重放信息的一个程序段。
12. 根据权利要求 11 所述对其上嵌有标识符的电子存储媒体的使用进行识别和提供响应的程序, 其中所述服务器计算机执行表查找来确定一个或多个授权类型。
13. 根据权利要求 11 所述对其上嵌有标识符的电子存储媒体的使用进行识别和提供响应的程序, 其中所述服务器计算机执行表查找来确定一个或多个授权标题。
14. 根据权利要求 11 所述对其上嵌有标识符的电子存储媒体的使用进行识别和提供响应的程序, 其中所述服务器计算机利用一个来自所述服务器计算机的事项启动对授权信息的授权重放。
15. 根据权利要求 11 所述对其上嵌有标识符的电子存储媒体的使用进行识别和提供响应的程序, 其中一个事项被写入到一个数据库记录处理。



16. 根据权利要求 11 所述对其上嵌有标识符的电子存储媒体的使用进行识别和提供响应的程序, 其中包含一个从所述服务器计算机接收实时更新信息的程序段。
17. 根据权利要求 11 所述对其上嵌有标识符的电子存储媒体的使用进行识别和提供响应的程序, 其中包括一个启动视频重放的程序段, 其中所述重放可以包括对一个安全连接解锁的逻辑。
18. 根据权利要求 11 所述对其上嵌有标识符的电子存储媒体的使用进行识别和提供响应的程序, 其中包含传送视频、用户信息和合适的播放器的标记到一个数据库的一个程序段。

说明书

利用激光中心媒体及网络服务器授权使用电子内容的制作的系统、方法及物品

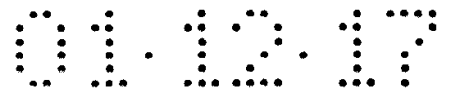
本发明所属技术领域

本发明涉及一种利用电子媒体上的一组位元，利用一个网络服务器以电子方式跟踪和控制内容使用的配送和跟踪系统。

与本发明相关的背景技术

目前人们熟悉的光盘把信息保存为一系列精微的凹点和平滑区域，其排列方式是同心圆或螺旋轨道，另一方面，是环状盘片的光滑平坦的表面。从一个光盘中读取记录信息的方式是让聚焦的激光束沿记录轨道移动，检测激光束强度经过盘片凹点和平滑区域时发生的强度变化。激光的一致性和相对短的波长使得在记录媒体的非常小的空间可以存放大量的信息。

在 1982 年，光盘首先被引入音乐唱片工业；目前占据所有音乐唱片媒体销售额的 43%。根据美国唱片工业协会报道，仅在美国，每年销售的光盘都超过三亿张，零售额超过三十亿美元。最近十年来，唱片工业把直径五英寸的预装光盘包装在六英寸乘十二英寸的纸板箱，即业界所说的“长箱”中。长箱可以很容易的放置在与传统的乙烯基密纹唱片（LP）音乐存储展示箱柜类似的展示箱柜中。更重要的是，长箱的体积使得商场小偷难以把预装光盘藏在衣服或钱包里不付钱就离开音像店。尽管预装光盘的长箱包装技术对于防



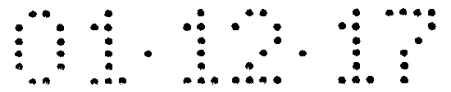
范小偷来说有一定作用，但是这种过分包装每年大约浪费两千五百万磅包装材料。

美国唱片工业协会在 1991 年宣布了放弃长箱的意图。在 1992 年二月，该协会宣布，从 1993 年 4 月开始，所有预装光盘都以 5 英寸乘 5.5 英寸的包装上市销售。

制造光盘（CD）或数字激光视盘（DVD）时，它们通常被装在心轴上传送或存放。这至少部分是因为该存储媒体易脆的本质。因为每个盘片有一个中心孔，而盘片又相对薄且轻，把多个盘片存储在一个心轴上是很方便的。如光盘制造业使用的轴通常有一个大约两英尺长的柱和大约两英寸厚的加重底座。根据制造光盘的自动化程度不同，盘片在印刷或包装之前要在心轴上多次存放或搬运。在最接近全自动化的过程中，仅在检查和印刷步骤之间以及最后包装之前需要把盘片存放在心轴上。在更加手工化的系统中，盘片可能在每个制造步骤之间都要放置在心轴上，包括成型和硬化之间，硬化和薄膜旋涂之间，薄膜旋涂和检查之间，检查和印刷之间，印刷和最后包装之间。但是，不管盘片放置在心轴上的次数有多少，每当盘片被移出以进行加工时，即存在有失窃和标题混淆的可能性。换言之，只要盘片在心轴上，尤其是没有任何印刷识别的情况下，对于该心轴的标题识别就很容易产生问题或发生混淆。有必要使盘片具备内在的能力来跟踪该盘片并提供配送管理、质量控制和消费者取用信息。

类似的，无论盘片在轴上存放多长时间都可能失窃。如果没有任何措施来防范未经授权而从轴上移动盘片，或没有措施来准确跟踪在轴上有多少盘片，那么盗窃总会时有发生。

光盘（以下称“CD”）多媒体商品是一个日渐增长的工业。CD 多媒体用于音频、视频、音像、以及基于计算机的应用。因为一个

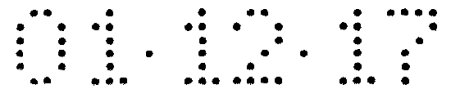


特定 CD 程序的许多外观类似的复制品可以从许多不同的来源获得，经销商很难跟踪、识别和区分他们的存货与其它人的存货。

安全性是这类商品的出租、出借或销售中人们关心的一个重要问题。商业预装光盘节目可以从出租店、商店或图书馆获得。对经销商来说有一个简单的方法来识别和保护其商品是十分重要的。例如，经销商需要确定归还的光盘就是当初出租的光盘，以防止消费者试图用坏盘（例如消费者的被刮坏的盘）来交换好的出租盘。

用来自其它渠道的有缺陷的 CD 盘交换好 CD 盘是经销商面对的一个难题。因为关于光盘的业务量相当大而且难以检测非法掉包，商品被掉包就成了一个很严重的问题。人们需要一个简单可靠的方法来检查 CD 中存放的数字资料是否损坏或失效。尽管可以用简单的肉眼检查明显的缺陷如刮痕或裂缝，但这不能检查数字资料的缺陷。即使这些缺陷通常可以用加速重放整个 CD 的方法来发现，但这些方法在商业上是不可行的，因为这需要花费经销商太多时间来完成检查返回给他们的大量的每一张 CD 的大量的工作。尽管现在有检查数字记录的高速电子扫描设备，但是由于这些设备高昂的价格和对其相关技术的有限的了解，个别经销商难以获得并有效的利用它们。

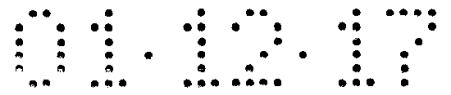
监视敏感物体从被控区域流出的电子商品监视系统已众所周知，而且单独或结合长箱包装技术用于控制未经授权拿走光盘。用一块高渗透性磁材料制成的标记被放置在盘片的包装中。空间隔离的检测平板被放置在商店、图书馆或其它存放被监视的光盘的地方的出入口。检测平板中有磁场线圈和检测线圈，用于产生穿过出入口的磁场，检测通过平板之间的标记。如果有人试图在没有消除盘片包装中的标记之前携带该光盘通过检测平板形成的磁场，就能检测出标记并启动报警。



美国专利第 4,710,754 号披露了一个特地为光盘尺寸设计的多方向 EAS 标记。在该 '754 专利中说明的标记由一个高渗透性、低抗磁力的、通常为平面的磁响应材料组成，它至少包括两个确定转换部分的狭窄区域，以及相邻的较宽的通量收集器部分。狭窄的转换部分和通量收集器部分并列放置，从而使通量高度集中于转换部分。当转换部分高度集中的磁力线在经过变化的磁场时产生高频谐波，从而使磁场中的标记可以被检测到。通过包括与每个转换区域相邻的可磁化的材料，该标记通常有两种状态，即消磁和磁化两种可逆状态。该可磁化材料被磁化时，可影响相邻的转换区域，或者保持磁化状态在改变的检测磁场中不发生逆转，或者至少改变标记在磁场中的响应。在任何一种情形下，根据可磁化材料是否被磁化，可以很容易的区分标记在检测磁场中产生的不同的信号。

美国专利号 4,967,185 披露了一个特地为各式光盘尺寸而设计的多方向双态 EAS 标记。在该 '185 专利中说明的标记包括一个连续不间断残余可磁化材料片，该材料覆盖在类似专利 '754 的响应材料片上。通过在把标记拿到磁场之前，选择性地对该连续残余磁化材料磁化或去磁，可以明确的改变该标记在变化的磁场中的响应。上述现有技术中披露的标记可以附加到光盘的包装中。但是，当试图把现有技术的标记直接附加到光盘的表面时，问题出现了。从盘片读取信息时需要旋转光盘，因此盘片必须自行保持平衡。这样，直接附加到光盘表面的标记应该尽量接近光盘中心以避免影响光盘的平衡。但是，现有技术 EAS 标记，不是自行保持平衡的。而且，传统的光盘包括必须的中心圆孔，它必须保持清洁不能阻塞，而优选现有技术的双状态 EAS 标记则包含磁性材料的连续片，故不能在不阻碍光盘圆孔的情况下被放置在光盘的表面的中心。

美国专利第 4,709,813 号说明了一个光盘防窃设备，可以克服不能直接把 EAS 标记应用到光盘表面的问题。该 '813 号专利说明了一个内表面带 EAS 标记的可分离的锁定片，该片内表面可以选择性



的锁定在光盘的“珠宝箱”上。光盘通过该片物理地锁定在箱腿中。店员或其它授权用户可以在付款时用加锁的释放工具去除该片。应该指出，使用锁定片需要一段准备时间来把该片附加到每个光盘箱上，在付款手续中增加了一道附加步骤，而且当 EAS 标记从光盘上去除后，光盘就失去了 EAS 的保护。当锁定片去除之后没有 EAS 保护使零售商允许客户在没有购买前试播放光盘要冒很大的风险。预装光盘的新包装标准，尽管在环境保护上可行，但大大增加了光盘失窃的问题。因为，小包装更容易被隐藏而带出店外。

尽管使用电子商品监视系统可以部分补偿增加的商店失窃的威胁，但是应该认识到未经授权从包装上去除磁性标记将避开监视系统的检测功能，而且现有的 EAS 标记不能在不影响光盘的可操作性的情况下直接装配到光盘上。结合锁定片使用 EAS 标记可以解决取放的问题，但不能解决当消费者被允许在购买之前先试听光盘时光盘的物理安全问题。一种通过使用电子商品监视系统应用可直接用于光盘表面的 EAS 标记特别设计的防窃压缩激光信息盘具有相应地明显的优点。因此，经销商们需要方便而便宜的保证他们的电子内容媒体的安全性。

发明简述

本发明提供一种以电子方式跟踪内容配送的制作的系统、方法和物品。首先，一种电子存储媒体跟踪标识符被嵌入到一个电子存储媒体上并存储在一个数据库中。其次，一种包装跟踪标识符被设置在存放所述电子存储媒体的包装中。然后，当所述电子存储媒体在不同实体间传输时，使用包装上的标识符来跟踪所述电子存储媒体。进一步，可以使用电子存储媒体上的标识符来识别所述电子存储媒体，以便提供授权使用包含在电子存储媒体中的信息。

附图简要说明

通过以下对本发明的优选实施例的详细说明及参考附图，可以更好地理解前述内容和其他目的、特点及优点，其中：

图 1 是根据本发明跟踪电子媒体的方法的总体方块图；

图 2 是根据一个优选实施例跟踪电子媒体的方法的详细方块图；

图 3 是与本发明的一个实施例相关的硬件实施例的方块图；

图 4 用图的形式比较说明了现有的电子存储媒体和本发明的电子存储媒体的生命周期；

图 5 是根据一优选实施例的用户经验的方块图；

图 6 是根据一个优选实施例的一个电子商务事项的改向操作的流程图；

图 7A 和 7B 是阐明根据一个优选实施例与用户连接和更新 DVD 处理有关的详细逻辑的流程图；

图 8 是根据一优选实施例，说明利用用于智能处理的 BCA 信息显示基于零售商/分销商的特定广告信息的逻辑；

图 9 的流程图是根据一优选实施例，说明了利用用于智能处理的 BCA 信息显示基于 DVD 的类型/种类的特定广告信息；

图 10 的流程图显示根据一优选实施例，利用用于智能处理的 BCA 信息下载和更新特定 DVD 零售商信息的下载操作；

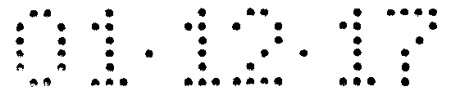


图 11 的流程图显示根据一优选实施例，利用用于智能处理的 BCA 信息下载和更新特定 DVD 标题信息的下载操作；

图 12 的流程图显示根据一优选实施例，利用用于智能处理的 BCA 信息的合适的视频浏览的操作；

图 13 的流程图显示根据一优选实施例，利用用于智能处理的 BCA 信息的合适的视频浏览操作；

图 14 的流程图显示根据一优选实施例，利用用于智能处理的 BCA 信息与合适的多媒体浏览操作相关的逻辑；

图 15 的流程图显示根据一优选实施例，利用用于智能处理的 BCA 信息限制访问特定网络站点的安全性操作；

图 16 的流程图显示根据优选实施例，利用用于智能处理的 BCA 信息的电子商务事项的解锁操作；

图 17 的流程图显示根据优选实施例，利用用于智能处理的 BCA 信息的电子商品事项的解锁操作；

图 18 的流程图显示根据一优选实施例，利用用于智能处理的 BCA 信息的跟踪盗版和滥用 DVD 的登录操作；

图 19 的流程图显示根据一优选实例的智能处理的支持事项的改向操作；

图 20 的流程图说明的是根据一优选实例的智能处理的支持事项的显示操作；

图 21 是根据一优选实例利用用于的智能处理的 BCA 支持跟踪的流程图；

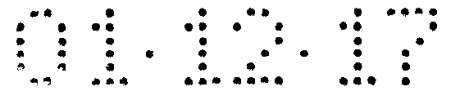


图 22 是根据一优选实施例用于智能处理的支持事项的改向操作的流程图;

图 23 是根据一优选实施例利用用于智能处理的 BCA 信息, 以便进行下载更新信息、支持信息和应用信息的播放操作的流程图。

发明详述

本发明包括以电子方式跟踪内容配送的制作的系统、方法和物品, 并且在此信息基础上提供智能服务。图 1 的方块图概括的介绍介绍了根据本发明的跟踪电子媒体的方法。一开始, 以音乐、视频、资料或任何其它类型可视或可听的娱乐或信息的形式的内容在操作 10 和 12 产生。之后, 一电子存储媒体跟踪标识符, 例如脉冲刻写区 (Burst Cut Area(BCA)), 可在制造时内嵌到电子存储媒体 22。应注意电子存储媒体 22 可以采用任何可以存储内容的电子/光学存储媒体的形式。然而, 在本说明中, 将把焦点集中在电子存储媒体的一个实施例, 即 DVD 上。

如图 1 所示, 内容产生之后, 可以在操作 14 用复制设备复制电子存储媒体。而且, 一个包装跟踪标识符可以嵌入到存放电子存储媒体包装上, 然后这些跟踪标识符被存入一个数据库。

在使用中, 可以在步骤 16, 18 和 20 从分销商到零售商与消费者来跟踪该电子存储媒体。当电子存储媒体在各种实体如复制商、分销商、零售商和消费者之间传输时, 跟踪是利用在包装 22 上的跟踪标识符完成的。而且, 当最终用户获得该电子存储媒体时, 可以使用电子存储媒体 22 中的跟踪标识符来识别该电子存储媒体。正如下面所述, 通过识别电子存储媒体可以提供各种不同的特征。

正如前面提到的, 当电子存储媒体在各种实体如复制商、分销商、零售商和消费者之间传输时, 电子存储媒体可以通过使用包装

上的跟踪标识符而被跟踪。具体的说，复制商是制造或“出版”该 DVD 的公司。复制商从内容开发商（工作室例如 NEW LINE(新线工作室)）接受数字线性磁带（DLT），然后制作基于 DLT 内容的 DVD“玻璃母盘”。然后该玻璃母盘成为 DVD 母盘，其它所有 DVD 都从该母盘复制。作为复制过程的一部分，复制商增加 BCA 号码到每个 DVD 中，然后对将 DVD“包装/装箱”以便配送到分销商或零售商。

另一方面，分销商是把多种标题包装在一起配送给零售商的公
司。分销商的价值在于他们维持与零售商直接的关系和渠道，可以存放大量的产品库存，这是小型零售商难以负起的调节规模经济的作用。零售商要求从分销商处获得多种产品（例如 20 份“迷失在太空”(Lost in Space)，50 份“罗宾”(Ronin)，100 份“你已得到邮件”(You've Got Mail)，它们可能来自不同的工作室)，然后，分销商可以把各种产品包装在一起分发给零售商。

最后，零售商是把产品直接卖给顾客的公司。这种包括砖块和灰泥（brick-and-mortar，店名）店的例子包括炸弹音像（Blockbuster Video），好莱坞音像（Hollywood Video），最优惠（Best Buy），好小子（Good Guys）等等。零售商还包括在线零售商例如 DVDExpress，Amazon.com，及其它电子商务公司。其它公司也在加入零售商的行列，例如 Nimbus，它既是复制商又是分销商。提供直接到消费者的在线销售产品是合理的下一步。应该注意，前面提到的复制商也可以是分销商（Nimbus/Technicolor, WAMO/Deluxe）。同样，复制商也可以直接把商品传输给零售商，尤其是像 Blockbuster 那样的大客户。

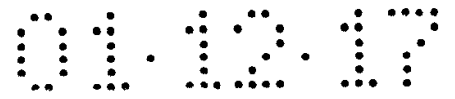
优选实施例

下面我们详细讨论有关跟踪 DVD 的例子。首先，内容拥有者（例如工作室）请求在他们的 DVD 上使用 BCA。基于请求，复制商（例如，WAMO, Panasonic, Nimbus, Technicolor, Pioneer, Crest）为每个 DVD 增加唯一的 BCA 号码。增加 BCA 号码给每个 DVD 需要用到一种特别的（YAG）激光。这可能是制造过程的最后一步。特定 DVD 的 BCA 号码必须输入交互（InterActual）的 BCA 数据库。跟踪的信息包括：DVD 标题，如“Lost in space”；BCA 号/范围，如 12345687890；和装运包装/跟踪集装箱，如到 Hollywood Video 的第 52221 箱。

把 BCA 号码加到 DVD 上之后，这些 DVD 就被打包/装箱以便分发到分销商或零售商。应该注意到许多公司有多种身份，因此复制商和分销商可能是同一公司。而且，某些零售商足够大或足够重要，则可直接从复制商获得出货。DVD 包装/出货的方式十分重要，因为人们必须跟踪 BCA 号码到实际的出货集装箱。因此，跟踪信息也必须加入 BCA 数据库。

如果包装好的 DVD 被送往分销商，那么分销商也有相应机制，如扫描仪，输入设备，和监视设备等来跟踪他们的分销过程。例如，Deluxe 收到 100,000 份“Lost in space”的包装。然而，该分销商将 10,000 份发给零售商 A，5,000 份给零售商 B。该分销商应该能够“输入”零售商 A 和 B 的分销信息到该系统。理想情况下，这应该是一个无缝/自动的处理过程。

一旦 DVD 到达零售商（不管来自复制商或分销商），DVD 将进一步被分发到当地的商店或其它配送渠道。此时，零售商应该能够自动“跟踪”这些 DVD 到它们的商店。经过一定时间，所有的三个实体（复制商，分销商和零售商）都能够将跟踪信息加入到 BCA 数据库。由于其复杂性和对现有商业系统的依赖，可将零售跟



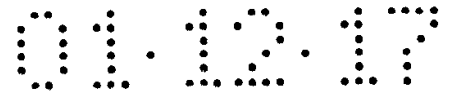
踪概念在各阶段逐步铺开：首先是具有重要零售帐号的复制商，接着是引入分销商，然后零售商开始有能力根据当地的商店或其它配送渠道来跟踪。

BCA 识别在最终用户的使用

如前所述，当最终用户获得电子存储媒体时，电子存储媒体可以用电子存储媒体上的跟踪标识符来识别。通过这种识别，在识别电子存储媒体方面可以形成各种特征。应该注意，在一个实施例中，识别通过计算机和软件控制在电子存储媒体被识别后形成的特征来实现。

例如，本发明可以在个人计算机的环境下如 IBM 兼容个人计算机、苹果麦金塔 (Macintosh) 计算机或 UNIX 的工作站上实践。图 3 说明了代表性的硬件环境，它说明了根据优选实施例的工作站的典型硬件配置，包括中央处理单元 110，如微处理机，以及许多其它通过系统总线 112 互连的单元。图 3 所示的工作站包括随机存取内存 (RAM) 114，只读存储器 (ROM) 116，用于连接外围设备如磁盘存储单元 120 到总线 112 的输入/输出 (I/O) 适配器 118，用于连接键盘 124，鼠标 126，喇叭 128，麦克风 132，和其它用户接口设备如接触式屏幕 (未显示) 到总线 112 的用户适配卡 122，用于连接工作站到通信网络 (如数据处理网络) 的通信适配卡 134，以及用于连接总线 112 到显示设备 138 的显示卡 136。该工作站通常有在其上驻留的操作系统，例如微软的 Windows NT 或 Windows /95 操作系统 (OS)，IBM OS/2 操作系统，MAC OS，或 UNIX 操作系统。本领域技术人员应当理解本发明还可以在上面提到的以外的平台和操作系统上实现。

优选实施例是用 JAVA, C 和 C++ 语言编写，并使用对象导向程序设计方法。越来越多的对象导向程序设计 (OOP) 用于开发复杂的应用程序。当 OOP 逐渐成为软件设计和开发的主流时，各种软



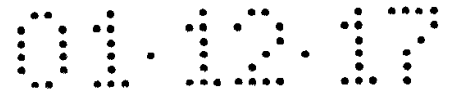
件解决方法都需要适应它并利用它的优点。有必要采用 OOP 的原理来实现电子信息传递系统的信息传递界面，例如信息传递界面可以提供的 OOP 类和对象的集合。

OOP 是使用对象开发计算机软件的程序，包括分析问题、设计系统和构造程序等步骤。对象是包含资料以及其相关的结构和程序的集合的软件包。因为它包含资料以及结构和过程的集合，它可以被看成是不需要其它附加结构、过程或资料来处理其特定任务的自足组件。因此，OOP 把计算机程序看作大量称为对象的自足的组件集合，每个对象代表一个特定的任务。这种把资料、结构和过程等包装到一个组件或模块的概念称为封装。

通常，OOP 组件是可再度利用的软件模块，它提供了遵循某对象模型的界面，并且可以通过组件整合体系结构在运行期间访问。组件整合体系结构是允许不同处理空间中的软件模块利用彼此的能力或功能的一套体系结构机制。这通常是通过假设一个通用组件对象模型来实现，在该模型上建立该体系结构。此处有必要区分一个对象和一类对象。一个对象是该类对象中的单独一个，一类对象通常称作一类。一类对象可以被看作是一个蓝图，从中可以形成许多对象。

OOP 允许程序设计师建立一个对象以作为另一个对象的一部分。例如，代表活塞引擎的对象可说与代表活塞的对象有组成关系。实际上，活塞引擎包含活塞、阀门和许多其它组件；活塞是活塞引擎的一个组件的这一事实可以从逻辑上和语义上用 OOP 中的两个对象来表达。

OOP 还允许建立依赖于另一个对象的对象。如果有两个对象，一个代表活塞引擎，另一个代表包含用陶瓷制成的活塞引擎，那么这两个对象之间的关系就不是组成关系。陶瓷活塞引擎并不构成活塞引擎。它只是一种较活塞引擎多一个限制条件的活塞引擎，即它



的活塞是陶瓷做的。此时，代表陶瓷活塞引擎的对象称为导出对象，它继承了代表活塞引擎的对象的全部特征并进一步为它增加了一个限制或细节。代表陶瓷活塞引擎的对象“依赖于”代表活塞引擎的对象。这样的对象之间的关系称为继承。

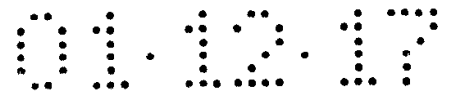
当代表陶瓷活塞引擎的对象或类继承了代表活塞引擎的对象的全部特性时，它继承了定义于活塞引擎类的标准活塞的热学特征。但是，该陶瓷活塞引擎对象置换了有关特定的陶瓷热学特征，它通常与金属活塞的有关特征显著不同。它略过原来的功能，使用有关陶瓷活塞的新功能。不同种类的活塞引擎有不同的特征，但是可能具有相同的基础功能（如，在引擎中有多少活塞，点火序列，润滑油等）。为了访问任何活塞引擎目标中的这些功能的每一项，程序设计师可以用同样的名字呼叫同样的功能，但是每种类型的活塞引擎可以在相同的名字后面采用不同/置换的执行功能。在同一名字后面隐藏不同的执行功能的能力称为多态性，它大大地简化了对象之间的通信。

使用包含组成关系、封装、继承和多态性等概念，一个对象可以表达现实世界中的任何事物。事实上，我们对现实的逻辑理解是在确定可以成为对象导向软件中的对象的事物的种类中的唯一限制。一些典型的分类方法如下：

- * 可以表达物理事物的对象，如在交通流量仿真中的汽车、在电路设计程序中的电子组件、经济模型中的国家、航空交通控制系统中的飞机。

- * 可以代表计算机用户环境的单元的对象、如窗口、菜单、或图形对象。

- * 一个对象可以代表一个目录清单，例如个人文档或城市的经纬度表。



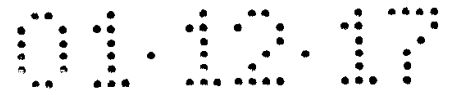
* 一个对象还可以代表用户定义的数据类型，如时间、角度、复数、或平面中的点。

通过以具有众多能力的对象来表达任何逻辑上可分离的事物，OOP 允许软件开发者来设计和实现以现实世界的一部分为模型的计算机程序，无论在现实中它是物理实体、过程、系统或事物的组合。因为对象可以代表任何事物，软件开发者可以建立作为将来更大的软件工程的一部分的对象。

如果新的 OOP 软件程序的 90%包括已证明的、来自以前存在的可再度利用对象的现存的组件，那么仅有剩下的 10%的新软件程序需要编写和测试。由于 90%来自反复测试过的可再度利用对象，这些原始部分产生的错误可能是整个程序的 10%。因此，OOP 可使软件开发者从外部的已有的对象设计对象。

该程序十分类似用零件或子零件组装复杂的机器。因此，OOP 技术使软件工程师更象硬件工程师，软件用现有的组件建立，这些组件可以用对象的形式提供。这既增加了开发速度又改善了软件质量。

程序语言正在开始全面支持 OOP 原理，例如封装、继承、多态性和包含关系。随着 C++语言的出现，许多商用软件开发者开始使用 OOP。C++是一种 OOP 语言，它提供一种快速的机器可执行的代码。而且，C++既适合商业应用又适合系统程序工程。现在，C++似乎是众多的 OOP 程序设计师的最流行的选择，但是还有其它 OOP 语言，如 Smalltalk, Common Lisp Object System(CLOS), 以及 Eiffel。另外，OOP 能力被加入到更传统的流行计算机程序语言如 Pascal 中。



对象类的优点可以归纳如下：

- * 对象及其相应的类把复杂的程序设计问题分解为许多小而简单的问题。

- * 封装通过把资料组织到小而独立的可以互相通信的对象中增强了资料的抽象性。封装避免了对象内的资料被无意破坏，但允许其它对象通过呼叫对象的成员功能和结构来与资料互动。

- * 子类和继承使通过从系统中存在的标准类导出新的对象来扩展和修改对象成为可能。因此，可以不必从头建立新的能力。

- * 多态性和多重继承可使不同的程序设计师混合和匹配多种不同类的特征，并建立特定的对象按预想的方式与相关对象仍旧一起工作。

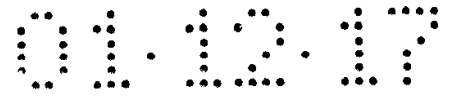
- * 类分级和保存分级提供了灵活的机制来模拟现实世界对象及其之间的关系。

- * 可以重复利用类的库在许多情形下是有用的，但它们也有局限性，例如：

- * 复杂性：在一个复杂系统中，相关类的分级继承可能变得极其容易混淆，有几十个甚至数百个类。

- * 流程控制：在类库的帮助下编写的程序仍要负责流程控制（即，它必须控制来自某一特定库的所有类之间的相互作用）。程序设计师必须确定在何时呼叫何种对象的何种功能。

- * 加倍的努力：尽管类库允许程序设计师使用和再度利用许多小段的程序，但每个程序设计师以不同的方式组合这些程序段。两个不同的程序设计师使用相同的类库来编写完成同样事情的两个程序，其内部结构（例如，设计）可以完全不同，这取决于每个程序设计师在做此事时的几百个小的决定。不可避免的是，做类似事情的小段程序完成的形式稍有不同，而且其不能如预期的做地一样好。

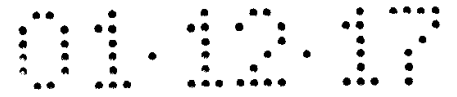


类库十分灵活。当程序变得更复杂时，越来越多的程序设计师被迫一遍又一遍的对基本问题重新发明基本的解决方法。类库概念中的一个较新的扩展是有了类库架构。该架构更复杂，包含相互协作的多个类集合，它们采用小规模的模式和主要的机制来实现在特定的应用范围的通用需求和设计。它们当初是为了将应用程序设计人员从涉及个人计算机的显示菜单、窗口、对话框和其它标准用户接口组件的琐碎的工作中解放出来而开发的。

架构还代表程序设计师关于考虑自己编写的代码和其它人编写的代码之间的相互作用方式的改变。在早期的过程程序设计中，程序设计师呼叫操作系统提供的库来执行特定的任务，但基本上，程序从开始到结束执行代码页，程序设计师单独对流程控制负责。这对打印工资单、计算数学表，或解决其它用一种方式执行程序的问题有效。

图形用户接口的发展开始把程序设计方法完全颠倒过来。界面允许用户而不是程序逻辑来驱动程序，并确定何时执行特定的动作。现在，大部分个人计算机软件通过事件循环完成此动作，事件循环监视鼠标、键盘和其它外部事件来源，并根据用户执行的操作呼叫相应部分的程序代码。程序设计师不再确定事件发生的顺序。相反的，程序被分为在不可预测的时间以不可预测的顺序呼叫的独立的部分。通过此方法把控制权交给用户，开发者可以建立更易于使用的程序。但是，开发者编写的单独的程序片仍然呼叫操作系统提供的库来完成某项任务，而程序设计师仍然必须确定在被事件循环呼叫之后每个程序片之内的流程控制。应用程序代码仍然“位于系统之上”。

即使事件循环程序需要程序设计师编写许多当单独为每个应用程序设计时所不需要的代码，但应用架构的概念还是进一步包含了事件循环概念。使用应用程序构架的程序设计师不再设计基本菜



单、窗口、对话框的所有细节然后再把这些组合到一起，而是从适当的应用程序代码和基本用户接口元素开始工作。然后，他们从那里通过用想要的应用程序的特定功能替换架构的一些一般性功能来建立程序。

应用架构减少了程序设计师必须从开始编写的代码的总量。但是，因为架构实际上是一个显示窗口、支持复制和粘贴等一般性的应用，程序设计师可以比事件循环程序允许的程度更大的放弃控制权。架构代码负责几乎所有的事件处理和流程控制，只是在架构需要时才呼叫程序设计师的代码（例如，建立或操纵个人拥有的数据结构）。

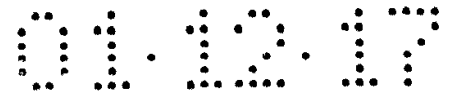
编写架构程序的程序设计师不仅把控制权交给了用户（事件循环程序也已经如此），而且把程序内部的流程控制的细节交给了架构。该方法允许建立以有趣的方式一起工作的更复杂的系统，而不是像单独的程序那样重复建立处理类似问题的常用的代码。

因此，如上述说明的，一个架构基本上是对给定问题域进行可再度利用设计方案的相互调整的类的集合。通常它包括提供默认行为（如对菜单和窗口）的对象，且程序设计师通过继承某些默认行为并置换其它行为来使用它，以便架构在合适的时间呼叫程序代码。

架构和类库有三个主要的不同：

* 行为与协议：类库实质上是行为的集合，当你的程序需要那些个别的行为时你就呼叫它们。另一方面，架构不仅提供行为而且提供协议或一套规则来控制行为组合的方式，包括程序设计师想提供的行为与架构所提供的行为之间的规则。

* 呼叫与置换：对类库而言，程序设计师把其对象实例化并呼叫它们的成员功能。对架构而言，有可能用同样的方法实例化和呼



叫对象（如把架构看作一个类库），但是为了充分利用架构的可再度利用的设计，程序设计师通常写入置换代码并由架构呼叫。架构管理对象内部的流程控制。编写程序的过程是把任务配送给不同的软件供架构呼叫，而不是指定这些不同的部分如何协同工作。

* 实现与设计：对类库而言，程序设计师只再度利用实现，而在架构中，他们还再度利用设计。架构体现了一族有关的程序或软件片。他代表适用于在给定域内的许多特定问题的一般性设计方法。例如，一个架构可以体现用户接口工作的方式，即使用同一架构建立的两个不同的用户接口用来处理大不相同的界面问题。

因此，通过开发各种不同问题和程序任务的架构，可以实现显著减少软件设计和开发的努力。本发明的优选实施例使用超文本链接标记语言（HTML）实现国际互连网络上的文件和通用的安全通信协议的协同工作，传输媒体使用该安全通信协议在客户和 Newco HTTP 或其它可以方便的代替超文本标记语言 HTML 的协议之间通信。关于这些产品的信息参见 T. Berners-Lee, D. Connolly, “RFC 1866: Hypertext Markup language-2.0”（1995 年 11 月）和 R. Fielding, H. Frystyk, T. Berners-Lee, J. Gettys 和 J.C. Mogul 的 “Hypertext Transfer Protocol--HTTP/1.1: HTTP Working Group Internet Draft”（1996 年 5 月 2 日）。HTML 是用于建立便于从一个平台到另一个平台的超文本文件的简单的资料格式。HTML 文件是具有一般的语义学特征的 SGML 文件，适合表达较宽范围内的信息。HTML 被用于从 1990 年开始的全球信息网（WorldWide Web）的全球性信息。HTML 是 ISO 标准 8879 的一个应用：1986 信息处理文本和办公系统标准通用标记语言（SGML）。

目前，网络开发工具在建立动态网页应用时遇到了限制，动态网（Web）应用从客户扩展应用到服务器并与现有的计算机资源互动。直到最近，HTML 已经成为开发基于网的解决方案中应用的主要技术。但是，HTML 已证明在下列方面不合适：

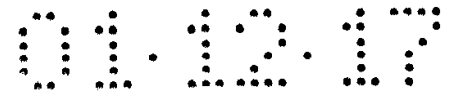
- * 性能差;
- * 限制用户接口的能力;
- * 只能产生静态网页面;
- * 缺乏与现有的应用和资料互动的能力; 及
- * 无法扩充。

Sun Microsystem (太阳微系统) 公司的 Java 语言解决了许多客户端的问题, 通过:

- * 改善客户端的性能;
- * 使能够建立动态、实时网应用; 及
- * 提供建立大范围的各种用户接口组件的能力。

使用 Java, 开发者可以建立功能强大的用户接口 (UI) 组件。订制的“窗口小组件”(如实时取用量计数器, 动画图标等) 可以被建立, 且客户端性能也可以改善。不像 HTML, Java 支持客户端验证的概念, 下载合适的处理程序到客户端以改善性能。可以建立动态的、实时的网页。使用上面提到的订制 UI 组件, 可以建立动态网页。

Sun 的 Java 语言已经成为被业界认可的“国际互连网络程序设计”语言。Sun 定义 Java 为: “一种简单的, 对象导向的, 分配的, 解释的, 强健的, 安全的, 架构中立的, 便携的, 高性能的, 多线程的, 动态的, 术语兼容的, 通用的程序编辑语言”。Java 通过平台独立的 Java 小程序的形式支持国际互连网络程序设计。“Java 小程序是小的特定的应用程序, 遵循 Sun 的 Java 应用程序设计界面 (API), 允许开发者添加“交互式内容”到 Web 文档(如, 简单的动画, 页装饰, 基本游戏等)。Java 小程序在与 Java 兼容的浏览器(例如 Netscape Navigator) 中执行, 通过从服务器复制代码到客户端。从语言的立场, Java 的核心特征基于 C++。Sun 的 Java 文献



指出：Java 是基础的，“从 Objective C 扩展了更多动态方法解决方案的 C++”。

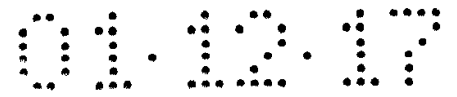
提供与 JAVA 相似功能的另一技术是由 Microsoft（微软）和 ActiveX Technologies 提供的，给开发者和网页设计者必要的工具来设计国际互连网络和微机上的动态内容。ActiveX 包含开发动画，3-D 虚拟现实、视频和其它多媒体内容的工具。这些工具使用国际互连网络标准，在多平台上工作，得到超过 100 家公司的支持。该集团的积木式模块被称为 ActiveX controls，是小而快速的组件使开发者能够把部分软件嵌入超文本标记语言（HTML）页。ActiveX controls 和许多程序设计语言一起工作，包括 Microsoft Visual C++，Borland Delphi，Microsoft Visual Basic 程序设计系统，将来还要和微软的 Java 开发工具，代码命名为“Jakarta”，一起工作。ActiveX Technologies 还包括 ActiveX 服务器架构，允许开发者建立服务器应用。本领域的一般技术人员很容易地认识到可以用 ActiveX 替代 Java 来实现本发明而不会有不适的问题。

根据一优选实施例的系统软件

当消费者在当地商店购买 DVD，或通过在线零售商在线购买时，可以获得给消费者使用的新 DVD。消费者把 DVD 放进计算机，然后 DVD 启动用户和国际互连网络服务器应用程序之间的在线话路与 DVD-ROM 光驱中的 DVD 紧密通信。有三种使用 BCA 的情况：

（1）消费者运行浏览器进入一个需要利用 BCA 信息来查看数据库中的内容的站点。该数据库也用从当前用户和他们的统计资料中收集到的信息来更新。

（2）本地的应用程序（例如 PCFriendly）自动连接到国际互连网络的某个 Web 服务器上，该服务器要查看或操作 BCA 信息。或



(3)本地的应用程序例如 PCFriendly 利用已经包含在 BCA 号中的信息以及基于该信息的局部制作经验。

我们将讨论相关的各种情况的细节。

情况 1: 进入查找 BCA 的站点。使用光驱中的 DVD, 消费者连接到一个特定的站点, 该站点有嵌入可以读取 BCA 信息的网页的代理/组件。该嵌入组件读取 BCA 以及其它潜在的信息(用户识别码等), 传递该信息给网络服务器。然后网络服务器根据预先定义的条件/市场/配置来裁剪返回给消费者的响应。

情况 2: 本地的应用程序(例如 PCFriendly 客户软件)自动连接到网络服务器(不需消费者的手工干预)并传递 BCA 信息给本网络服务器。根据 BCA 号和其它潜在信息, 网络服务器传递信息给消费者的客户软件或根据该信息/配置/零售商/等提供远程国际互连网络信息。

情况 3: 本地的应用程序(例如 PCFriendly)读取 BCA 信息并作用于 BCA 号中预先定义的信息。这种情况并不一定需要连接到国际互连网络。可以利用 ASPI 代码读取 188 字节信息来获得 BCA。

各种情况的实施例:

例 1: 用 C++设计 ActiveX controls 并嵌入到 HTML 页(使用 HTML 中的标准 OBJECT 对象定义)。当安装该网页时, ActiveX controls 也被安装。经用户授权, 该 ActiveX controls 访问该 DVD-ROM 光驱, 获得 BCA 资料以及其它任何潜在信息。该 ActiveX controls 然后用 HTTP 或 FTP POST 方法把这些信息发送至网络服务器。网络服务器自动读取和解析该 POST 信息, 并对这些信息采取相应的行动(例如, 发送给消费者一个独特的 URL, 仅当带有正确的 BCA 的正确的 DVD 在该 DVD-ROM 光驱中时才可访问)。

例 2: 本地 C++ 应用程序 (PCFriendly) 利用 InterActual 开发的远程代理技术。远程代理技术可自动连接到远程网络服务器 (无须消费者干预) 并把 BCA 号和其它任何潜在信息传递给网络服务器。远程代理程序也支持 HTTP 或 FTP POST 方法。网络服务器自动读取并解析该 POST 信息, 并采取相应动作。

例子包括:

消费者购买特定产品的请求被自动转到当初购买 DVD 的零售商。为了支持该例子, 需要在本地下载 (或解锁) 虚拟 POP/MDF 显示文件和信息并提供给消费者。

例 3: 位于本地网页的本地 C++ 应用程序或 ActiveX controls 访问 DVD 中的 BCA 信息。本地应用程序根据这些信息对其采取相应的动作。(在本模式中, BCA 域中必须包含足够本地应用程序对其进行操作的信息)。

本系统包含一个根据 BCA 信息提供实时查找的在线数据库。在数据库中的查找结果可以提取特定信息给应用程序如消费者配置, 零售商和技术支持的位置, 以及盗版信息。

BCA 信息的使用

零售配送

当远程代理程序连接到有 BCA 信息的服务器时,服务器对 BCA 号执行实时查找并对通过的 BCA 号确定复制商、分销商、及/或零售商。然后该信息可以用于各种项目,例如更新或改变 PCFriendly 软件的通道/广告标识/程序。图 2 说明该操作为 RemoteSync 238。把特定信息如 HTML, 视频, 图形和其它功能块 Unlock Server 230 中说明的信息解锁。按照功能块 Unlock Server 230 中所示根据 BCA 信息播放不同的信息或视频。该应用程序还根据 BCA 信息 RemoteSync 238 下载新内容。还可以利用 BCA 信息把电子商务事项或“buy-me(购买我)”按钮发送到合适的零售商,该功能由 RemoteTrak/BCATrak 功能 234 实现。

依照一优选实例的应用程序还可以广播新信息/更新,如图中 Broadcast Server 功能块 236 所示。并可根椐 BCA 信息提供逻辑解锁及/或控制对特定站点的访问的逻辑,如 RemoteTrak Server 功能块 230 所示。该逻辑使消费者改向至特定的零售商“店面”。

跟踪个体零售店操作

使用根据 BCA 号的信息可以跟踪特定零售店操作以及与特定零售店有关的消费者在线使用。这为本地零售商提供了确定吸引用户上线最有可能成功的机会。例如虚拟购买点数的信息,诸如虚拟购买地点 (POP) 和市场开发资金 (MDF) 这类信息则利用 BCA 信息和 RemoteTrak Server 功能块 230 来跟踪和吸引消费者。

优惠券

优惠券等(如“凑整”优惠券、折扣优惠券、优惠券等此处统称“优惠券”)已经成为许多商品销售策略的一个有机部分,尤其

是零售物品、杂物、食品、五金、衣服等等，通常在本地杂货店、药店和打折店销售。产品制造商开始依赖于优惠券、折扣、礼品券等来推销新的或已有的产品，提高销售额，获得有关消费者购买模式的统计信息。消费者开始依赖于优惠券或礼品券将其作为降低成本的技术。

现有的优惠券技术有一些缺点，不止是响应率低和欺骗。在现有技术中，优惠券以使用直接的邮寄技术、打印到报纸杂志上，或类似方式分发，与其它商品一起分发（如将洗衣服的肥皂和洗衣机一起打包），或与同样或相似的商品一起分发（如通过原始设备制造商或 OEM）或与计算机或其它商品分发（如当你买下一个时给你凑个整数，不要你付零头）。这些方法需要大量印刷和分派工作，而且历史证明有很低的响应率（如，通常低于 2% 的分发的优惠券被兑付）。因此，这种大规模分发技术不是成本高效的，而且由于大量浪费纸张，对环境也是不利的。

这种低响应率一部分是由于消费者在去商店之前难以保存、分类和找到所需的优惠券。特定的消费者只能拥有确实已经真正送给他而且他还恰巧保留着的优惠券。而且，因为许多优惠券有截止日期，消费者必须仔细对优惠券分类来确保在截止日期之前兑现它，这既费时又费力。通常，仅有那些把优惠券纳入预算或把优惠券作为一种爱好的消费者才有足够的时间来最大程度的利用这些优惠券。较繁忙的或较富裕的消费者不相信这样的优惠券管理是成本高效的。后一类消费者代表了产品制造商更期望吸引或跟踪的人群。

随着某些零售店（如杂货连锁店等）开始提供双倍甚至三倍的兑现优惠券以及慷慨的现金折扣（例如礼品证书等），在优惠券销售中欺骗开始成为逐渐增长的问题。彩色摄影技术可以制造和原件难以区分的优惠券。不道德的消费者使用这种复制的优惠券用低价来购买大量的商品，或欺骗性的获得他并未购买的产品的折扣。而

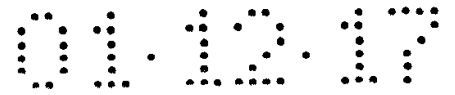
且，一些不道德的零售商伙同优惠券经纪人兑现大量非法所得来欺骗制造商。

因为优惠券折扣和回扣被用于推销产品，给消费者的净价格可能低于产品制造商的批发价。产品制造商提供这种不合理的折扣期望将来以完全的零售价格销售。如果消费者使用假优惠券购买多种零售商品，产品制造商就不能获得预期的以完全零售价格销售产品，于是整个优惠券策略就是失败的。

另外，现有的优惠券技术产生很少向产品制造商提供有关谁兑现了这些优惠券的有用的信息。消费者统计信息对于产品制造商确定哪那些产品针对哪那些消费群体（如通过特定的广告地点）是无价之宝。而且，这些统计资料可以用于将来更有效的分发优惠券。另外，有关购买习惯的信息（如流行程度，频率，以及货币价值或RFM）以及特定消费者或消费群体的统计信息具有市场价值，并且这样的信息可以出售或事项以获得利润。

人们尝试了各种技术来消除或减少欺骗，提供更方便的技术来分发优惠券，更好的跟踪消费者统计资料。De Lapa et al.，美国专利第 5,353,218 号披露了一个集中的优惠券系统。图 6 对 De Lapa et al.作了最清楚说明。该专利 De Lapa et al.揭示的分发优惠券系统有一个包括消费者和优惠券识别码的机器可读代码（条形码）。消费者代码可以用优惠券校验查找中的通用代码代替。整个机器可读代码可以捕获和下载到中心数据库来确定优惠券和消费者识别码。上载的信息可以用于销售目的（以确定下次送给该消费者哪个优惠券）和折扣的目的。

尽管该系统试图提供更集中的分发技术，它仍然依赖于分发给消费者的纸质优惠券。消费者可能不会打开而扔掉这些信件（即垃圾邮件）。而且，该系统依赖于消费者通过问卷等形式提供统计信



息来获得优惠券。此外，因为这些优惠券是预先印刷好的，优惠券事项或复制就可能更流行。

另外，在上述 De Lapa et al. 专利中，没有提供获得后续统计信息的机制。由于消费者资料在商店这一级被获得，需要附加的机制来上载这些消费者信息给中心数据库，以获得消费者统计信息。因此，零售商可能一开始就不愿意向这样的计划投资。

零售时，消费者付款的时间有必要尽可能快。因此，如果在消费者付款的时候需要附加处理时间来处理上述 De Lapa et al 优惠券，零售商可能不会接受该技术。

而且，在 De Lapa et. al 体系下，没有提供机制来确保接受优惠券的个人就是使用优惠券的个人的机制。如果消费者搬家到一个新地址，原有地址的新住户可能收到并兑现寄给他的优惠券。因此，目标跟踪资料就可能不精确或不完整。

Murphy，美国专利 5,305,195，发证于 1994 年 4 月 19 日，披露了一种对在线终端的交互式广告系统。一系列远程终端接受可能存储在内部硬盘中的压缩和编码的视频广告信号。通过广告视频的播放，消费者可能使用该终端来选择产品。在图 4（第 7 栏，第 45-50 行）Murphy 说明了一个可以用于打印所选优惠券的打印机。

Murphy 设备可以解决困扰纸质优惠券的一些有关问题。但是，Murphy 系统似乎更专注于直接广告信息而不是收集统计信息或分发优惠券。因此，Murphy 设备似乎不能处理统计信息或减少优惠券的伪造。更进一步，Murphy 说明该设备用于大学校园，这就限制和缩小了消费人群。

Von Kohorn，美国专利 5,128,752，1992 年 7 月 7 日发证，说明了一个产生和兑付从电视资料上选择代币的系统和方法。产品信息

和鉴定资料可以被传送到电视机和家用打印机上显示。观众可以选择打印一个优惠券并在零售店兑现它。

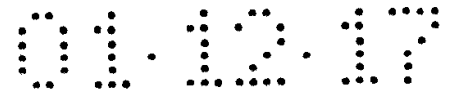
Von Kohorn 确实说明了一种减少欺骗的技术(第 7 栏,第 16-38 行)。但是,该技术需要在零售店这一级来验证优惠券是否合法,而且在一实施例中要求消费者提供识别证书。该技术在零售店使用可能既麻烦又碍事,在零售店任何时候都可能有许多优惠券要求兑换。

此外, Von Kohorn 系统依赖于广播技术,不能对特定的消费者提供特定的优惠券。相反,它把优惠券分发给装备合适设备的所有电视观众。注意在图 6(第 9 栏,第 40-48 行)中, Von Kohorn 说明了记录来自编码到优惠券的消费者信息的销售资料的技术。

Axler et al., 美国专利 5,305,197 号,发证于 1994 年 4 月 19 日,说明了带反馈的优惠券分发机器。消费者亭子被放置在零售店之类的地方来展示广告(LED 滚屏)并允许消费者打印所选的优惠券。附近的转换器检测在设备周围的消费者的存在。

Axler 设备可以解决一些与纸质优惠券分发有关的问题。但是,除了所打印的优惠券数目和类型之外,似乎 Axler 设备不能收集除了所打印的优惠券数目和类型以外的大量的消费者统计资料。而且,在商店内部的环境中,难以输入这些消费资料,尤其是用 Axler 说明的键盘。因此, Axler 设备不适合收集消费者统计资料。

Axler 设备的一个根本缺陷是它没有摆正目标或者把激励消费者带着特定的优惠券来零售店买东西放在更优先的位置。置于商店内的 Axler 设备方便消费者把目标集中到优惠券上。换言之,消费者可能在商店选择许多产品然后到 Axler 优惠券亭子处来确定这些商品是否有折扣或回扣。因此,优惠券的根本目的 - 激励消费者购买商品 - 就大打折扣了。



另外，Axler 亭子还占用有价值的商用零售空间。在零售店（如超市之类）即使是几英寸的柜台都极有价值，可以展示和放置零售的商品。产品制造商甚至以回扣或广告费的形式付“租金”给零售商以便获得主导货架的位置。因此，零售商将不愿意放弃这些有价值的空间给优惠券分发机。对消费者来说，在该发送机排队既费时又烦人。提供附加的发送机可能有成本障碍。

根据一优选实施例的支持服务

为了在商业环境中增强对 DVD 的支持，如图 2 功能块 234 的 RemoteTrak/BCATrak Server 功能块所示，根据利用 BCA 号进行表查找，BCA 号可以用来改向到特定的支持站点。另如功能块 234 RemoteTrak/BCATrak Server 所示，提供逻辑用以跟踪来自制造过程的光盘变形或缺陷。同时，如功能块 234 RemoteTrak/BCATrak Server 所示，提供其它逻辑用以跟踪特定零售商的支持信息，如功能块 234 RemoteTrak/BCATrak Server 所示跟踪地理支持信息，如功能块 234 RemoteTrak/BCATrak Server 所示限制对基于 BCA 信息的支持站点的访问。最后，提供增强支持用以广播利用基于 BCA 信息的支持和驱动器的更新，如功能块 236 Broadcast Server 所示。

根据一优选实施例的安全性

将 BCA 信息与游戏解锁逻辑相结合可为授权用户提供基于 BCA 信息的解锁视频，如功能块 238 DVDUnlock Server 所示。BCA 信息具有独特的识别码，结合其它资料，可以跟踪电影或游戏何时送给了朋友，后者又触发了要付款的事项，如功能块 234 RemoteTrak/BCATrak Server 所示。该信息还可用于跟踪盗版 DVD，如功能块 230 RemoteTrak/BCATrak Server 所示，把信息报告给零售商、报告给制造商、或报告给分销商。如功能块 230 RemoteTrak/BCATrak Server 所示，这种功能可以把盗版限制在特定

的区域或零售商，并跟踪非法使用的地区码和潜在地跟踪到零售商/分销商。

根据一优选实施例的一般/广告逻辑

如功能块 238 RemoteSync 所示，根据与 BCA 有关的消费者配置还提供逻辑来制作基于 BCA 的部分信息的视频（如功能块 238 DVDUnlock Server, RemoteSync 所示为一类统计人群播放视频 1，为另一类播放视频 2，如功能块 238 RemoteTrak/BCATrak Server 所示，剪裁基于 BCA 信息的国际互连网络/浏览器的经验。根据 BCA 信息还可提供目标性广告，并在 PCFriendly 软件内制作用于通道/标识/程序编制的内容）。

图 5 是根据一优选实施例的用户经验的方块图。BCA 号 503 被烧录/添加到 DVD505。当 DVD 被放置到消费者的计算机 510 时，InterActual 的软件自动读取 BCA 号并传递该信息给网络服务器，运行 ISAPI 扩展 520，使用 HTTP 或 FTP 协议 515。该信息可从本地“客户端”应用程序传送，或从将该信息传送给网络服务器的一个网络站点下载小程序或 ActiveX-type control。目前该信息使用如下语法使用 HTTP POST 命令传送：

```
http://www.pcfriendly.com/scripts/RemoteAgentUpgrade.DLL&bc
a=1234568790?userid =1234568790?...
```

目前 Web 服务器的执行是使用 Visual C++ 编写的 ISAPI 扩充，其用于 Microsoft Windows NT 并被命名为 RemoteAgentUpgrade.DLL。一旦接受到该 POST 命令，ISAPI 扩展解析 POST 命令中的信息以便确定 BCA 号和其它有关信息（例如用户 ID 等）。然后该信息被记录到 Web 服务器的记录表 530，并被用来根据 POST 查询 Web 服务器数据库 550 的特定信息。该可变数据库的结构使 BCA 号可以有多种用途。

此处提供的是根据一优选实施例的零售商例子，本领域技术人员可以毫不费力的制作和使用本发明而不需预作试验。消费者在其 DVD-ROM 光驱插入 DVD，可见到一具有“购买我”按钮的 HTML 网页。点击“购买我”按钮，消费者连入国际互连网络并进入包含 ActiveX control 的特定网页。该 ActiveX control 使用目前插入的 DVD 将 BCA 信息自动连接到 ISAPI 扩充。该 ActiveX control 并可告诉 ISAPI 扩充该消费者试图进行电子商务事项。ISAPI 扩充解析来自 POST 命令的信息，并连接到 Web 服务器数据库。因为 ActiveX control 告知 ISAPI 扩展该消费者正试图进行电子商务事项，ISAPI 即连接到 Web 服务器数据库来确定当初该 DVD 是从哪家零售店购买的。可以这样做是因为 Web 服务器数据库包含有三栏的 BCA 查找表 560:

BCA 号	#123458790
DVD 标题名	迷失在太空
零售商/店	Hollywood Video, Store #23

利用零售商/店信息，可以从包含该零售商的特定信息的零售商列表 570 确定合适的电子商务 URL:

零售商/店	Hollywood Video, Store #23
电子商务 URL	http://www.retailer23.com/...

图 6 的流程图说明根据一优选实施例利用用于智能处理的 BCA 信息的电子商务事项的改向操作。处理过程从 600 开始，如功能块 610 所示当一个用户插入 DVD 到播放器并启动电子商务操作。当用户在 610 选择购买选项时，逻辑被启动以便读取 BCA 信息，如在功能块 620 所示，该信息与服务器数据库中的其它用户信息结合起来。然后服务器执行表查找以确定销售该原始 DVD 的零售商，如功能块 630 所示。原始零售商成为用户在功能块 610 启动的购买目标，该电子商务事项被改向到销售该光盘的零售商，如功能块 640

所示。最后，该事项被传送到记录有关该改向操作的事件的服务器数据库。

图 7A 和 7B 的流程图阐明根据一个优选实施例与用户连接和更新 DVD 处理有关的详细逻辑。开始处理时，用户使用 DVD 应用程序连接到国际互连网络，即如功能块 700 所示。远程代理检测到在线国际互连网络连接并把该应用程序连接到服务器以便进一步处理，如功能块 710 所示。然后，服务器用合适版本的标识符连接该应用程序，如果有更新的版本，那么不需要用户进一步输入就更新该远程应用程序，如功能块 720 所示。如果用户是第一次使用该系统的用户，那么服务器从用户处，利用例如来自 DVD 上的资料或一个请求操作，获得用户信息，如功能块 730 所示。然后，应用程序收集当前 DVD 使用信息并记录该信息到数据库，如功能块 740 所示。最后，当前 DVD 信息被传送给用户，如功能块 750 所示。随后，处理过程转移到图 7B 的功能块 752，此处应用程序确定是否可以获得广播事件。然后，在功能块 754，如果用户请求广播事件，那么服务器用 HTTP 格式传送该信息给用户，如功能块 756 所示。远程代理从服务器接受信息并为特定 DVD 播放器转换信息，如功能块 758 所示，最后记录用户信息到服务器的数据库，如功能块 760 所示。

一般广告流程

图 8 的流程图说明根据一优选实施例的一般广告服务的详细逻辑。该流程图说明了为特定的分销商/零售商等提供订制的广告（如识别）的详细逻辑。图 8 显示了根据一优选实施例，利用用于智能处理的 BCA 信息，显示基于零售商/分销商的特定广告信息的逻辑。在 800 开始处理过程，此时，用户插入一张带 BCA 信息的 DVD 到播放器。广告操作被功能块 810 所示的用户操作启动。当用户在 810 连接到国际互连网络的网页时，就在功能块 820 启动逻辑来读取

BCA 信息。该信息与来自服务器数据库的其它用户信息结合起来。然后，服务器执行表查询，找到销售原始 DVD 的零售商，该过程如功能块 830 所示。一旦原始零售商被确认，服务器就再次执行表查找，确定广告标识，如功能块 840 所示。然后，与原始零售商对应的广告标识就显示在 810 功能块中提到的网页中，该过程如功能块 850 所示。最后，该事项被传送到记录有关广告操作的事件的服务器数据库 860。

分销商、零售商、计算机或其它硬件制造商，直销人员，内容开发者或任何配送、销售、或给出 DVD 的人都将从中收益，如下面的优选实施例中详细说明。其中一些包括：Blockbuster, DVDExpress, Amazon.com, Best Buy, Deluxe, Technicolor/Ninbusl, IBM, Gateway, Dell, Creative Labs, New Line, Warner, Activision, Electronic Arts, General Motors 和 Ford Motor Company。

图 9 的流程图显示根据一优选实施例，利用用于智能处理的 BCA 信息，显示基于 DVD 的类型/种类的特定的广告信息。处理在 900 开始，此时用户插入带 BCA 信息的 DVD 到播放器。广告操作被功能块 910 所示的用户操作启动。当用户在 910 连接到互连网网页时，启动逻辑读取 BCA 信息，并且该信息与来自服务器数据库的其它用户信息结合，如功能块 920 所示。然后，服务器执行表查找，确定 DVD 的类型和标题，如功能块 930 所示。一旦标题和类型确定了，服务器就执行另一个查找来确定广告标识，如功能块 940 所示。与 DVD 的类型和标题相对应的广告标识就显示在 910 所提到的网页，如功能块 950 所示。最后，一个事项被发送至记录有关广告操作的事件的服务器数据库 960。

图 10 的流程图是一个根据一优选实施例，利用用于智能处理的 BCA 信息，下载和更新特定零售商信息的下载操作。处理过程开始于 1000，此时用户通过一 DVD 应用程序连接到国际互连网络。逻

辑检测到国际互连网络的连接，读取 BCA 信息，并启动到服务器的连接，如功能块 1010 所示。当逻辑在 1010 将连接进行到服务器后，DVD 应用程序即请求服务器下载与当前插入 DVD 所对应的零售商的信息，即如功能块 1020 所示。服务器执行表查找，确定最初销售该 DVD 的零售商，如功能块 1030 所示。然后服务器执行另一个查找来确定下载的信息，如功能块 1040 所示。一旦功能块 1020 的应用程序所启动的下载请求确定之后，服务器就把下载的信息用 HTTP 协议传到应用程序，如功能块 1050 所示。最后，事项被发送到记录有关下载操作的事件的服务器数据库 1060。

图 11 的流程图是一个根据一优选实施例，利用用于智能处理的 BCA 信息，下载和更新 DVD 标题特定信息的下载操作。处理开始于 1100，此时用户通过一个 DVD 应用程序连接到国际互连网络。逻辑检测到活动的国际互连网络连接，读取 BCA 信息，确定 DVD 版本信息，启动到服务器的连接，如功能块 1110 所示。在逻辑启动到服务器的连接 1110 后，DVD 应用程序请求服务器下载与当前插入的 DVD 对应的标题信息，如功能块 1120 所示。服务器执行表查找，确定当前插入的 DVD 标题，如功能块 1130 所示。然后服务器执行另一个查找来确定下载的信息，如功能块 1140 所示。一旦功能块 1120 的应用程序所启动的下载请求确定之后，服务器就把下载的信息用 HTTP 协议传到应用程序，如功能块 1150 所示。最后，事项被发送至记录有关下载操作的事件的服务器数据库 1160。

图 12 的流程图是根据一优选实施例，利用用于智能处理的 BCA 信息的一个合适的视频观看操作。处理开始于 1200，此时用户插入一个 DVD 到播放器，视频重放被用户操作启动，如功能块 1210 所示。当用户选择播放视频的操作于 1210，就启动逻辑读取 BCA 信息，该信息与服务器数据库的其它用户信息结合，如功能块 1220 所示。该服务器执行表查找确定销售原始 DVD 的零售商，如功能块 1230 所示。然后服务器执行另一个查找来确定当前要播放的零

售商视频，如功能块 1240 所示。一旦功能块 1210 的应用程序所请求的零售商视频信息被确定，服务器就启动销售该盘的零售商的正确的视频的重放，如功能块 1250 所示。最后，一个事项被发送至记录有关视频浏览操作的事件的服务器数据库 1260。

图 13 的流程图是根据一优选实施例利用用于智能处理的 BCA 信息的合适的视频浏览操作。处理开始于 1300，此时用户插入一个 DVD 到播放器，视频重放被用户操作启动，如功能块 1310 所示。当用户选择播放视频的操作于 1310，就启动逻辑读取 BCA 信息，该信息与服务器数据库的其它用户信息结合并传送到服务器，如功能块 1320 所示。该服务器执行表查找确定 DVD 的类型及/或标题，如功能块 1330 所示。然后服务器执行另一个查找来确定当前要播放的正确的类型和标题的视频，如功能块 1340 所示。一旦功能块 1310 的应用程序所请求的类型或标题的视频信息被确定，服务器就启动该类型或标题的正确的视频重放，如功能块 1350 所示。最后，事项被发送至服务器数据库，此数据库可记录有关视频浏览操作的事件 1360。

图 14 的流程图是一个合适的多媒体浏览操作的逻辑，其根据一优选实施例利用用于智能处理的 BCA 信息。处理开始于 1400，此时用户插入一个 DVD 到播放器，浏览被用户操作启动，如功能块 1410 所示。当用户选择浏览选项的操作于 1410，就启动逻辑读取 BCA 信息，如功能块 1420 所示。该 DVD 应用程序执行本地表查找确定类型/标题/零售商，如功能块 1430 所示。然后 DVD 应用程序执行另一个查找来确定当前要播放的正确多媒体元素，如功能块 1440 所示。一旦功能块 1410 的应用程序所请求的多媒体元素被确定，DVD 应用程序就启动该类型/标题/零售商的多媒体元素的正确的浏览重放，如功能块 1450 所示。最后，事项被发送至记录有关多媒体浏览操作的事件的服务器数据库 1460。

根据一优选实施例的安全处理流程图

图 15 是根据优选实施例，利用用于智能处理的 BCA 信息限制访问特定网络站点的安全性操作流程图的流程图。处理开始于处理始于 1500，此时用户插入一个 DVD 到播放器，安全操作被用户操作启动，如功能块 1510 所示。当用户启动一个到安全站点的连接于 1510，就启动逻辑读取 BCA 信息，该信息与来自服务器的其它用户信息结合，如功能块 1520 所示。然后，该服务器根据 BCA 号执行表查找，以确定该用户是否可以被允许访问该安全站点，如功能块 1530 所示。基于 BCA 号，服务器要么允许要么限制进入网站，如功能块 1540 所示。最后，事项被发送至记录有关安全操作的事件的服务器数据库 1550。

图 16 的流程图是根据优选实施例，利用用于智能处理的 BCA 信息，对电子商务事项的解锁操作的流程图。处理开始于 1600，此时用户插入一个 DVD 到播放器，解锁操作被用户操作启动，如功能块 1610 所示。当用户于 1610 选择播放/安装 DVD 操作时，即启动逻辑读取 BCA 信息，如功能块 1620 所示，该信息与来自服务器的其它用户信息相结合。然后，如功能块 1630 所示，该服务器执行表查找以确定该 DVD 是否可以被解锁以便播放或安装。如果服务器确定用户必须首先执行购买事项，服务器会提示用户任何所需的事项信息，如功能块 1640 所示。当用户完成了功能块 1640 的事项，或服务器确定在较早时候发生过事项，或者如果服务器确定事项并不需要发生，服务器就执行解锁操作，如功能块 1650 所示。最后，事项被发送至记录有关解锁操作的事件的服务器数据库 1660。

图 17 是根据优选实施例，利用用于智能处理的 BCA 信息，电子商务事项的解锁操作的流程图。处理开始于 1700，此时用户插入一个 DVD 到播放器中，解锁操作被用户启动，如功能块 1710 所示。

当用户选择播放/安装 DVD 操作于 1710，就启动逻辑读取 BCA 信息，该信息与来自服务器的其它用户信息结合，如功能块 1720 所示。该 DVD 应用程序使用 BCA 信息执行表查找，确定该 DVD 的用户信息，如功能块 1730 所示。然后，服务器执行表查找以确定该 DVD 是否可以被解锁以便播放或安装，如功能块 1740 所示。如果服务器确定用户必须首先执行购买事项，服务器会提示用户任何所需的事项信息，如功能块 1750 所示。当用户完成了功能块 1750 的事项，或服务器确定在较早时候发生过事项，或者如果服务器确定事项并不需要发生，服务器就执行解锁操作，如功能块 1760 所示。最后，一个事项被发送至服务器数据库，该数据库记录有关解锁操作的事件 1770。

图 18 是根据优选实施例中，利用用于智能处理的 BCA 信息，来跟踪盗版和滥用 DVD 的登录操作的流程图。处理开始于 1800，此时用户插入一个 DVD 到播放器，登录操作被用户操作启动，如功能块 1810 所示。当用户选择播放/安装 DVD 操作于 1810，就启动逻辑读取 BCA 信息，该信息与来自服务器的其它用户信息结合，如功能块 1820 所示。该服务器使用 BCA 信息执行表查找确定该用户是否被允许应用或安装该 DVD，如功能块 1830 所示。然后，服务器会允许或禁止数字激光视盘的播放/安装，如功能块 1840 所示。最后，一个事项被发送至服务器数据库，该数据库记录有关登录操作的事件 1850。登录信息可以用于把盗版盘片定位到特定的地区，跟踪非法地区代码，跟踪滥用/盗版 DVD 至零售商、分销商、制造商或内容开发者。

支持服务

图 19 是根据优选实施例，用于智能处理的支持事项的改向操作的流程图。处理开始于 1900，此时用户插入一个 DVD 到播放器中，改向操作被用户的操作启动，如功能块 1910 所示。当用户选择支



持操作于 1910，就启动逻辑读取 BCA 信息，该信息与来自服务器的其它用户信息结合，如功能块 1920 所示。该服务器使用 BCA 信息执行表查找确定该原始 DVD 的支持组织，如功能块 1930 所示。支持组织变成用户在功能块 1910 启动的支持请求的目标，支持事项就被改向到与该 DVD 相关的支持组织，如功能块 1940 所示。最后，一个事项被发送至服务器数据库，该数据库记录有关改向操作的事件 1950。

图 20 是根据一优选实施例，利用用于智能处理的支持跟踪的显示操作的流程图。处理开始于 2000，此时用户插入一个带 BCA 信息的 DVD 到播放器中，用户操作启动显示操作，如功能块 2010 所示。当用户选择支持操作于 2010，就启动逻辑读取 BCA 信息，该信息与来自服务器的其它用户信息结合，如功能块 2020 所示。然后，服务器执行表查找确定在用户的播放器中的 DVD 的特定 DVD 支持信息，如功能块 2030 所示。一旦服务器确认用户在功能块 2010 所启动的支持请求对应的特定 DVD 信息，DVD 特定信息就显示给用户，如功能块 2040 所示。最后，一个事项被发送至服务器数据库，该数据库记录有关显示操作的事件 2050。

图 21 是根据一优选实施例，利用用于智能处理的 BCA 支持跟踪的流程图。处理开始于 2100，此时用户插入一个带 BCA 信息的 DVD 到播放器中，显示操作被用户操作启动，如功能块 2110 所示。当用户选择支持操作于 2110，就启动逻辑读取 BCA 信息，该信息与来自服务器的其它用户信息结合，如功能块 2120 所示。然后服务器执行表查找确定在用户的播放器中的 DVD 的特定 DVD 支持信息，如功能块 2130 所示。一旦服务器确定用户在功能块 2110 所启动的支持请求对应的特定 DVD 信息，就使用该 DVD 特定信息，例如，来跟踪零售商特定支持问题或地理支持问题，如功能块 2140 所示。最后，事项被发送至服务器数据库，该数据库记录与显示操

作有关的事件 2150, 并且该记录的信息被用于产生跟踪特定零售商支持问题或地理支持问题的报告。

图 22 是根据一优选实施例, 智能处理的支持事项的改向操作的流程图。处理开始于 2200, 此时用户插入一个带 BCA 信息的 DVD 到播放器中, 改向操作被用户操作启动, 如功能块 2210 所示。当用户选择支持操作于 2210, 就启动逻辑读取 BCA 信息, 该信息与来自服务器的其它用户信息结合, 如功能块 2220 所示。然后服务器执行表查找确定该原始 DVD 的支持组织, 如功能块 2230 所示。支持组织变成在功能块 2210 启动的用户支持请求的目标。如果被允许, 支持事项被改向到与 DVD 相关的支持组织, 如功能块 2240。否则, 用户被改向到一个位置提示该用户支持位置不可达。最后, 事项被发送至记录有关改向操作的事件的服务器数据库 2250。

图 23 是根据一优选实施例, 使用用于智能处理的 BCA 信息下载更新、支持和应用程序信息的播放操作的流程图。处理开始于 2300, 此时用户用一个 DVD 应用程序动态连接到国际互连网络。逻辑检测在线的国际互连网络连接, 读取 BCA 信息, 确定 DVD 应用程序版本信息, 及启动到服务器的连接, 如功能块 2310 所示。在 2310 中, 逻辑启动到服务器的连接之后, DVD 应用程序从服务器请求所有有关该 DVD 的广播信息, 如功能块 2320 所示。服务器执行表查找确定该 DVD 的广播信息, 如功能块 2330 所示。一旦在功能块 2320 被应用程序启动的广播信息请求被确定, 服务器就使用 HTTP 协议传递广播信息到应用程序, 如功能块 2340 所示。然后, DVD 应用程序根据广播信息采取动作, 或者提供信息给用户, 或者根据信息自动动作, 如功能块 2350 所示。最后, 事项被发送至记录有关下载操作事件的服务器数据库 2360。电子商务 URL 被返回到 ActiveX control 以便消费者的购买请求可以被改向到合适的 URL。

下面提供了根据一优选实施例编写的 Visual C++ 程序，以便进一步说明本发明。

*这些功能用于获得 BCA 信息

*

*日期 名称 (姓名) 原因

*---- ---- ----

*3/22/99 ITI Created

*

*说明

*

*© 版权 1999 交互技术公司，保留所有权利

```
#include "stdafx.h"
```

```
#include "scsidefs.h"
```

```
#include "wnaspi32.h"
```

```
DWORD xReportBCA(LPBYTE pbData, WORD cbData);
```

```
DWORD AtapiSendCommand(LPBYTE pPacket, LPBYTE pBuffer, DWORD cbBuffer);
```

```
DWORD AtapiInit(int index);
```

```
void AtapiUninit();
```

```
DWORD xReportBCA(LPBYTE pbData, WORD cbData)
```

```
{
```

```
DWORD nReturn;
```

```

UCHAR    Cdb[16];
DWORD    bWindowsNT = FALSE;

    OSVERSIONINFO vi;
    vi.dwOSVersionInfoSize = sizeof(vi);
    if (GetVersionEx(&vi))
        bWindowsNT = (vi.dwPlatformId == VER_PLATFORM_WIN32_NT);

    if (bWindowsNT)
        return FALSE; // for now not implemented

    ZeroMemory(&Cdb,sizeof(Cdb));
    Cdb[0] = 0xAD;                // CMD_READ_DVD_STRUC;
    Cdb[7] = 0x03;                // Format
    Cdb[8] = HIBYTE(cbData); // sizeof AllocationLength
    Cdb[9] = LOBYTE(cbData); // sizeof AllocationLength
    Cdb[10] = 0;                 // Agid

    nReturn = AtapiSendCommand(Cdb, pbData, cbData);
    return nReturn;
}

typedef DWORD (_cdecl *LPFNSENDASPI32COMMAND)(LPSRB);
typedef DWORD (_cdecl *LPFNGETASPI32SUPPORTINFO)(VOID);
BOOL AspiInquiryCmd(BYTE *pbInq, WORD cbData);

// statics yuk
static BYTE AdapterCount = 0;
static BYTE AdapterID = 0;
static BYTE TargetID = 0;
LPFNSENDASPI32COMMAND g_fnSendASPI32Command = NULL;
LPFNGETASPI32SUPPORTINFO g_fnGetASPI32SupportInfo = NULL;
HINSTANCE g_hWNASPI = NULL;

```

```

DWORD AtapiInit(int index)
{
    if (g_fnSendASPI32Command && g_fnGetASPI32SupportInfo)
        return TRUE;

    if (!(g_hWNASPI = LoadLibrary("WNASPI32.DLL")))
        return FALSE;

    if (NULL == (g_fnSendASPI32Command = (LPFNSENDASPI32COMMAND)
GetProcAddress(g_hWNASPI, "SendASPI32Command")))
        return FALSE;

    if (NULL == (g_fnGetASPI32SupportInfo = (LPFNGETASPI32SUPPORTINFO)
GetProcAddress(g_hWNASPI, "GetASPI32SupportInfo")))
        return FALSE;

    DWORD ASPI32Status = (*g_fnGetASPI32SupportInfo)();
    AdapterCount = (LOBYTE(LOWORD(ASPI32Status)));

    if ((AdapterCount == 0) || (HIBYTE(LOWORD(ASPI32Status)) != SS_COMP))
        return FALSE;

    BYTE pbInq[LEN_INQUIRY_DATA+1];
    for (BYTE aid = 0; aid < AdapterCount; aid++)
        for (BYTE tid = 0; tid < MAX_TARGET; tid++){
            AdapterID = aid;
            TargetID = tid;
            if (AspiInquiryCmd(pbInq, LEN_INQUIRY_DATA)){
                if (DTYPE_CROM == pbInq[0]){
                    if(index-- == 0){
                        return TRUE;
                    }
                }
            }
        }
}

```

```

    }
}
return FALSE;
}

```

```

void AtapiUninit()
{
    if (g_hWNASPI){
        FreeLibrary(g_hWNASPI);
        g_fnSendASPI32Command = NULL;
        g_fnGetASPI32SupportInfo = NULL;
        g_hWNASPI = NULL;
    }
}

```

```

DWORD AtapiSendCommand(BYTE *pCdb, BYTE *pbData, DWORD cbData )
{
    PSRB_ExecSCSICmd pSrb =
(PSRB_ExecSCSICmd)malloc(sizeof(SRB_ExecSCSICmd));
    if (pSrb == NULL)
return FALSE;
    memset(pSrb, 0, sizeof(SRB_ExecSCSICmd));

    // SendCommand
    pSrb->SRB_Cmd    = SC_EXEC_SCSI_CMD;
    pSrb->SRB_Status = 0xff;
    pSrb->SRB_HaId   = AdapterID;
    if ((pCdb[0] == 0xA3) && (cbData != 0))
        pSrb->SRB_Flags = SRB_DIR_OUT;
    else if(pCdb[0] == 0x43)

```

```

    pSrb->SRB_Flags    = SRB_DIR_IN;
else
    pSrb->SRB_Flags    = SRB_DIR_SCSI;
pSrb->SRB_Target      = TargetID;
pSrb->SRB_BufLen      = (DWORD)cbData;
pSrb->SRB_BufPointer  = pbData;
pSrb->SRB_SenseLen    = SENSE_LEN;
pSrb->SRB_CDBLen      = LEN_ATAPI_PACKET;
pSrb->SRB_HaStat      = 0xff;
pSrb->SRB_TargStat    = 0xff;
memcpy(pSrb->CDBByte, pCdb, LEN_ATAPI_PACKET);

    DWORD ASPI32Status = (*g_fnSendASPI32Command)(pSrb);
    DWORD timeout = 600;
while ((pSrb->SRB_Status == SS_PENDING) && (timeout > 0)){
    Sleep(10);
    timeout--;
}

if (pSrb->SRB_Status == SS_COMP){
    free(pSrb);
    return TRUE;
}

if ((pSrb->SRB_Status == SS_ERR) && (pSrb->SRB_TargStat == STATUS_CHKCOND)) {
}
free(pSrb);
return FALSE;
}

BOOL AspiInquiryCmd(BYTE *pbInq, WORD cbData)
{
    BYTE    Cdb[LEN_ATAPI_PACKET];

```



```
memset(Cdb, 0, LEN_ATAPI_PACKET);
Cdb[0] = SCSI_INQUIRY;
Cdb[4] = LEN_INQUIRY_DATA;
```

```
    PSRB_ExecSCSICmd pSrb =
(PSRB_ExecSCSICmd)malloc(sizeof(SRB_ExecSCSICmd));
    if (pSrb == NULL)
        return FALSE;
    memset(pSrb, 0, sizeof(SRB_ExecSCSICmd));
    pSrb->SRB_Cmd      = SC_EXEC_SCSI_CMD;
    pSrb->SRB_Status   = 0xff;
    pSrb->SRB_HaId     = AdapterID;
    pSrb->SRB_Flags    = SRB_DIR_SCSI;
    pSrb->SRB_Target   = TargetID;
    pSrb->SRB_BufLen   = (DWORD)cbData;
    pSrb->SRB_BufPointer = pbInq;
    pSrb->SRB_SenseLen = SENSE_LEN;
    pSrb->SRB_CDBLen   = 6;
    pSrb->SRB_HaStat   = 0xff;
    pSrb->SRB_TargStat = 0xff;
    memcpy(pSrb->CDBByte, Cdb, LEN_ATAPI_PACKET);
```

```
// Send Command
```

```
DWORD ASPI32Status = (*g_fnSendASPI32Command)(pSrb);
    DWORD timeout = 600;
```

```
/* Wait for pending status */
```

```
while ((pSrb->SRB_Status == SS_PENDING) && (timeout > 0)){
    Sleep(10);
    timeout--;
}
```

```
/* Check Error Code */
```

```

if (pSrb->SRB_Status == SS_COMP) {
    free(pSrb);
    return TRUE;
}
/* Set last device error */
if ((pSrb->SRB_Status==SS_ERR) && (pSrb->SRB_TargStat==STATUS_CHKCOND)) {
}

free(pSrb);
return FALSE;
}

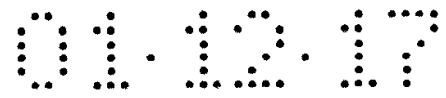
```

替代实施例

应当注意在不偏离本发明的精神的前提下，各种编序置换都可以采用包括，但不限于水印、全息图、或任何其它类型来替代或结合 BCA 信息。

水印

数字视频资料可以在品质无损的情况下重复拷贝。因此，视频资料的版权保护，在数字视频传递网络中比对模拟 TV 广播更重要。一种版权保护方法是在视频信号上附加“水印”，它携带关于所传送视频的发送者和接受者的信息。因此，水印可以用于识别和跟踪视频资料的不同拷贝。这些应用包括在万维网（WWW）上的视频发布、付费浏览视频广播、或者在视频盘或视频磁带上加标签。在所提到的应用中，视频资料通常以压缩格式存储。因此，水印必须嵌入到压缩域。根据另一优选实施例则提供一种强健的 MPEG-2 编码的视频水印方法。该方法的复杂程度比解码之后再于像素域加水印再重新编码的整个过程要减低许多。尽管已知的 MPEG-2 比特流已被部分改变，但该方法通过增加漂移补偿信号而避免了漂移现



象。该方法已得到实现，且结果确认了一强健的水印可以被嵌入到 MPEG-编码的视频信号中，其可用于安全传输速率为每秒几十个位的任意二进制信息。

该方法可方便的应用到其它视频编码方法，例如 MPEG-1、H.261、和 H.263。数字化水印存在于收敛点，此处，数字媒体内容的建立者和发布者要求局部化和安全的识别并认证该内容。因为盗版显然是一种对数字发布版权工作的侵害，所以设计复制和延伸复制的责任制是十分有价值的。考虑到各种形式的多媒体内容，不管“原版 (master)”，立体声，NTSC 制式视频，音频磁带或光盘，对质量降低的容忍程度随着个人情况不同而改变，并影响内容的根本商业和美学价值。

人们希望把版权、所有权、购买者信息或这些信息的组合与内容连以下列方式结在一起：不管是出于商业目的或其它的目的，其内容仅有在经历破坏而降低价值之后，才能在未经授权的情况下发布。人们认识到数字水印作为必要的商业内容发布（音频、视频、游戏等）的组件的重要性，这种法律认可和态度转变，通过参与数字内容商业发布的各种团体，将进一步推动可接受的参数的发展。

这些团体包括艺术家、工程师、工作室、国际互连网络访问供货商、出版商、代理、在线服务供货商、各种形式的内容传递增殖商、在线零售商、参与资金转移来裁定给目标群体实际传输的内容的个人或团体等。因为数字记录的特征变化很广，一个有价值的目标是提供工具来说明在一给定的数字化样本（视频、音频、虚拟现实等）中插入、保护和检测数字水印的最优化参数。以下所说明的最优化技术将使未经授权即移去包含这些参数的数字水印耗费大量的成本，从而保护给定的经济收益免遭未经授权商业发布的影响。如果移除数字水印不需要使用特别昂贵的工具，最优化技术至少需要对内容信号的显著的损坏，使得未经授权商业复制变得没有价值。

假定一些产品的商业价值显示一定程度的盗版实际上无法察觉，并且版权持有者认为被回报以全部的经济利益是“合理的”。例如，总是有 100 美圆的假钞，假 LEVI 牛仔裤，假 GUCCI 包等在市场上存在，并获得潜在的经济回报 - 同样也有未经授权的音乐、操作系统（Windows 98 等）、视频和其它多媒体产品的复制。但是，“数字市场”和物理市场的区别在于前者缺乏任何体系来设计责任制和产品鉴别的信任。对实体产品而言，公司和政府可对产品做标记并监视制造和销售能力来估计盗版的损失。前者仍缺乏增强机制，包括法律的、电子的和信息的活动来更好的教育消费者。

随着数字视频和数字视频广播的出现，版权保护的问题变得越来越重要，因为复制数字视频本质上不像仿真视频那样降低质量。一种版权保护的方法是附加水印到视频信号。水印是嵌入数字视频的比特流的数字编码，通常用来识别版权所有者。水印，如果应用到视频的个体复制，那么也可以用于识别每个复制品的接受者。该过程识别非法复制便于从接受者跟踪到制造者。对数字视频加水印，希望水印具有一些不同的特征。第一，水印应该以视频浏览者不能觉察或几乎不能觉察的方式嵌入。第二，水印应不能被有意或无意的对数字视频流或译码的视频的操作所消除，除非对该视频品质造成显著的降低而减少其商业价值（该特征称为“强健性”）。第三，因为视频可以存放在压缩格式（例如在“视频点播”服务器）中，人们希望能够在比特流中插入水印而无须首先对信号译码，然后在插入水印之后再编码。这可以用数字静态图像的水印来完成，但该方法并不适用于数字视频，这是因为视频信号提供的附加限制。许多数字视频应用程序是一种“恒定位速率”的应用程序，即不能容忍比特流传输速率的增加。即使在对恒定位速率没有限制的应用程序中，也应避免不必要的增加位速率，以便在有给定带宽的通道上传输时，能保持视频信号的实时译码能力。因此，最好在附加水印时不要增加视频信号的位速率。过去数字视频水印技术限制在对未压缩视频资料增加水印。但因为视频序列常以压缩格式（为

了节约存储空间)存储,故独立识别信号的每一个接受者所增加的水印就需要经过信号译码、水印、重新编码、与传送信号等过程。这显然要增加传递视频序列任务的处理时间和处理负担。

全息图

在共享的传送通道上交换信息和传送信息,给敏感信息的安全性带来了挑战。国际互连网络和企业内部网络是这类共享信息传送通道的两个例子,许多计算机通过局域通信网或广域通信网互相连接在一起。因此任何用户或入侵者都有可能截取通过该共享通道传输的敏感资料数据包。特别是国际互连网络正在迅速成为商业论坛,通过它的通道传输安全信息成为信息所有者主要关心的问题。资料加密技术可以用于增加在共享通道上交换和传输的数据的安全性。其最简单的形式是资料加密,使用基于特定运算法的“钥匙”来改变包含秘密信息的数据包的次序(“普通文字”),从而该资料被加密或“混杂”成形式上与嵌入的秘密信息无关的格式(“加密文字”)。未授权用户因不具备加密方法(如加密运算法)的知识或基于加密方法所形成的钥匙的知识,就不能方便地对信息译码。授权用户使用基于加密方法的“钥匙”从混杂资料中把嵌入的信息复原。因此,即使未授权用户获得了混杂资料,还需要加密方法和特定钥匙的信息来破译嵌入其中的秘密信息。

一个著名的加密系统是美国国家标准局于1977年采用的数据加密标准(DES)。这是一个利用混合与扩散技术的安全钥匙加密系统,可以使用64位这样的短钥匙来获得可接受的安全性。根据目前的计算能力,基于DES的加密系统的钥匙的数字可以有512个钥匙。但是,增加钥匙长度的“代价”是导致在传送和接受编码信息过程中时间严重延迟。两种主要的加密系统是一个对称系统,即私人钥匙系统和不对称系统,亦即公用-私人钥匙系统。DES对称加密系统通常用长度为56位的钥匙加密64位块的普通文字。基本的

结构模块 DES (称做一轮) 是根据钥匙进行替代组合然后进行文本置换。

普通文字通过 16 轮的功能编码, 其通常是对文本的子集执行替代、置换、XOR 与偏移操作, 并且钥匙的方式为加密文字的每个位都依赖于普通文字的每个位和钥匙的每个位。这意味如果单一位的加密文字在传输过程中被破坏, 那么整段信息就会损坏。这是 DES 类型块加密文字的另一个弱点。在每轮中, 钥匙的不同子集, K_i , 被用于执行加密 (因此, K_1 被用于第一轮, K_i 被用于第 i 轮等)。一个类似的运算法被用于对加密文字解密, 但钥匙按相反的顺序起作用, 左边的偏移操作改变为右边。由于 DES 运算法的复杂性, 处理 DES 加密的速度就成了硬件或软件实现的处理器特征。例如, 数字设备公司 (Digital Equipment Corporation) 制造的硬件 DES 芯片可以用每秒 1G 位的速率加密和解密, 相当于每秒 1.56 亿个 DES 块。软件实现则要慢一些, 例如, IBM3090 主机每秒可以加密 32,000 个 DES 块。

一般用于微处理机的软件实现的性能如表 1 所示。表 1 加密速率使用一些微处理机总线宽度 DES 块处理器速度 (MHz) (位) (每/秒)

8088	4.7	8	37068000	7.6	16	90080286	6.0	16	1,10068020	16.0	32
3,50068030	16.0	32	3,90080280	25.0	16	5,00068030	50.0	32	9,60068040	25.0	32
16,00068040	40.0	32	23,20080486	33.0	32	40,600					

另一个现有技术的加密系统是加利福尼亚州的 RSA 数据安全性公司 (RAS Data Security) 的 RSA 公用钥匙加密系统 (RAS Public Key Crypto system)。RSA 是不对称加密系统, 使用两种不同的钥匙: 公用钥匙加密普通文字, 私人钥匙解密加密文字。RSA 的硬件实现通常比 DES 的硬件实现慢 1000 到 10,000 倍。在软件实现时, RSA 通常比 DES 慢 100 倍。随着技术进步, 可以改善加密速度, 但是 RSA 的处理速度将难以达到对称加密系统的速度。因此, RSA 通常不被看作 DES 或任何其它快速加密运算法的替代物。相

反，RSA 通常用于保证在交换秘密信息之前交换安全钥匙，而长信息用 DES 加密。

信息发送时，它的 DES 钥匙使用 RSA 公用钥匙加密。许多其它现有加密系统是 DES 类型的加密系统的变种。通常，人们担心随着计算机微处理器的进步，DES 对付强力入侵将不够安全，因此从 20 世纪 80 年代开始人们就寻求其它方法。为适应该需要，开发了许多其它方法，被认为在所提供的安全性方面可以与 DES 竞争。这些系统的例子包括：

(1) Triple DES (三倍 DES)，这是 DES 的变种，普通文字用 DES 运算法采用三种不同的钥匙连续加密。这通常等价于把 DES 钥匙的长度增加到 112 位。三倍加密普通文字是目前处理对 DES 安全性有疑虑的方法，但是这显然增加了加密和解密信息的流量。

(2) REDOC，有 20 字节 (160 位钥匙) 的块运算法，它对 80 位的块进行操作。所有的操作 (如替代、置换、钥匙 XOR's) 都按字节执行，这使得用软件实现比 DES 更有效，后者的初始和结束置换难以用软件高效实现。另外，160 位的钥匙可以使该运算法非常安全。

(3) Khufu 是最近提出的 64 位块加密，它要求 512 位钥匙，轮数可变 (16, 或 24, 或 32)。因为有大钥匙，和潜在的轮数扩展，该运算法的安全性应该比较高。但是，增加轮数就降低了加密的速率。

(4) IDEA 是 64 位块加密，利用 128 位的钥匙。通常使用三种基本操作：XOR、加法模块 2 超 16 (addition modulo 2^{16})、以及乘法模块 2 超 16。该运算法通常对 16 位子块操作，这使它很高效，即使在 16 位的处理器上。它目前的软件实现速度几乎与 DES 一样快。在考察了本技术领域各种加密运算法的限制和不足之后，

本发明的发明人开发了一个基于光相位调制的新加密系统以及对应的在用户计算机和网络之间的实现界面。根据本发明的实施例可以交换任意方法，用于在数字化和通过共享网络（例如国际互连网）传输前将嵌入到数字比特流中的信息加密。

根据一优选实施例，通过给经过授权的用户对信息解密，全息脱密机可以用于接收端。随着具有高数据传输速率（例如，大于 2.4G 位每秒）的光纤网络越来越普及，本发明的许多优点中的一个是其具有潜力可达到极高速率的加密/解密速率（例如，大于 1G 位每秒）。在本发明的许多优选实施例中，数字资料包首先被转换到载波光束。这可以由二维空间光调制器完成。承载资料的光波的相位被相位混杂媒体连续的改变。然后，被改变了相位的承载资料的光波被用于形成带参考光束的光全息图。然后该全息图被转换成电信号，该电信号以数字形式经过共享传输通道到达目的地。在目的地混杂资料被接受，全息图显示在空间光调制器，并执行共轭重构来产生共轭相位扭曲的资料承载信号波形。全息媒体具有表达相位加密（置乱）媒体的信息，可以用于解除相位加密，并从共轭重构光波中提取嵌入的资料，其方法是使用光检测数组例如 CCD 数组。本发明的一个方面是使光加密钥匙达到大于 10^6 （ $10 \text{ sup } 6$ ）的钥匙，增强了安全性。

这是许多现有系统所难以实现的。本发明可使用如此巨大的加密钥匙是因为采用了独特的光仿真技术。本发明的另一方面是确保大加密钥匙的高速加密和解密，这在以前的系统中是很少达到的。本优选实例实现该方法的方法是使用高速光重构承载资料的全息图，和光数据处理设备的并行处理能力。本发明的另一方面是增强了加密体系的保密性，其方法是使用非传统的基于仿真技术的数字资料加密和解密。考虑到目前缺少基于仿真的加密解密基础理论，本发明就格外具有优势。基于运算法技术的强力加密系统侵入本发明的加密系统是不可能的。本发明的另一个方面是以不可见的方式

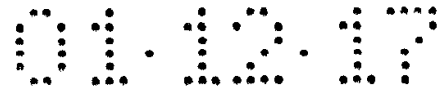
使用光相位信息来加密和解密数字资料。本发明的另一方面是将光全息技术应用用于加密和解密两个过程，进一步增强了加密系统的秘密性。本发明的另一个方面是承载资料的全息图的相位共轭重构在优选实施例中执行，以确保仿真解密过程的高度秘密性。本发明的另一方面是，整合了光处理技术、硬件加密、光电界面和高保真度、高速度数字信号传输来形成高安全性、快速、通用的加密系统，该系统独立于所使用的传输媒体工作。本发明的另一方面是，用一个步骤完成加密或解密过程，而不是在典型的对称加密体系中的 16 轮复杂计算。在根据本发明的光加密系统中，加密速度通常不受加密钥匙的大小所限制，而是受电-光和光-电信息形式转换系统速度的限制。

其它编序

过去，商人们采用了各种方法来试图跟踪或识别他们的存货，却没有取得成功。雕刻、盖章、绘画以及做标记是他们采用过的几种方法。由于实际的应用问题，这些方法不能有效的应用到 CD 多媒体租赁业。

在光盘多媒体业界人们都知道，识别记录程序标题和作者的图形信息通常放置在 CD 的上表面。数字资料被存储在上表面之上或之下。尤其是数字资料被存储 CD 的上表面和下表面之间的这些图形信息的下面。CD 的下表面由光亮的材料组成，在读取资料时，来自光盘播放器的激光束通过该表面向上发射。

数字资料是脆弱的，容易在处理过程中被损坏，典型的处理过程是用于识别商家的处理，如雕刻、盖章或做标记。如上所述，数字资料接近 CD 的上表面而不是接近其下表面。尽管 CD 的上表面包含的图形信息通常采用丝印，部分的保护了数字资料不受损坏，但丝印层比 CD 的下表面（包含光亮的材料）薄而且更脆弱。因此，



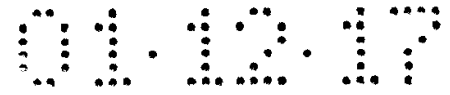
人们很希望保护 CD 的上表面和靠近它的数字资料免遭物理损伤，例如刮痕。

雕刻可以用于识别商家。用带有识别标记的雕刻 CD 是有问题的，因为雕刻通常是在 CD 的上表面，这容易干扰接近它的数字资料。进一步，即使试图在数字资料似乎不太可能被损坏的光盘的下表面雕刻，资料仍可能在雕刻过程中被损坏，因为需要在 CD 的上表面施加压力来使光盘保持在一定的位置并且这些雕刻可能会产生热量。另外，雕刻是不令人喜欢的，因为它的劳动强度相对较大，成本较高，尤其是在高容量的情形下。

因此，商家考虑其它破坏力不强的识别方法，例如绘画。但绘画也不能提供有效的识别和安全，因为它所需的劳动、成本、以及该过程内在的不可靠性（人们可以轻易的复制这些图画）。此外，因为必须避免对数字资料的损坏，绘画可能产生其它问题。

识别产品和保证安全的其它可供选择的方法是使用普通的粘贴标签。这些标签不能提供有效的识别，因为这些标签可以被轻易地去掉，并重新粘贴上类似的标签，没有一种方法能明确地指出标签被篡改过。另外，这些标签难以在没有粘贴工作站（如这里披露的一种）的情况下手工粘贴到 CD 上（因为任何标签都应准确地贴到 CD 的中心）。另外，这些标签可以被容易的复制。

磁类型的 EAS 系统广泛用于防止偷盗商品，例如衣服、书、磁带和光盘等。电子产品监视（EAS）系统通常用于防止未经授权的从被保护区域拿走产品，例如图书馆或零售店。EAS 系统通常包括一个检查区域或走廊，位于被保护区域的出口，检测附加到产品上的标记或识别。EAS 系统基于磁场、射频（RF）、微波和限磁技术。不管关于哪种具体技术，EAS 系统被设计成当被暴露在检查区域或走廊时，这些标记将产生一定的特征响应。检测这些特征响应就说明在走廊中有敏感的标记。EAS 系统然后启动一些合适的安全措



施，例如发出声音警报，锁定出口等。为了允许授权用户从保护区拿走产品，通常使用永久或可变的可消磁标记（如双状态标记）。

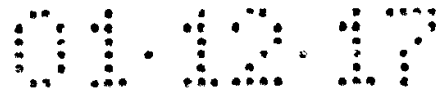
尽管 EAS 标记广泛用于防止窃贼偷盗光记录媒体，例如光盘和 CD-ROM，标记通常附加到包含新光盘的包装上，而不能直接附加到光盘本身，因为图书馆和其它机构要反复登记和检验光盘，以满足消费者和客户的需要。有效的存货控制希望把 EAS 标记附加到光盘上。

一些可以直接附加到光盘的标记已开发出来。现有的 Minnesota 采矿和制造公司，St. Paul, Minn. 的“DCD-1”就是单一的标记条带和安全覆盖可以附加到光盘上。但是，该标记影响了光盘的机械平衡，而进一步影响高速旋转的 CD-ROM 光驱、CD 播放器或其它光记录媒体播放设备的操作，这些设备在正常操作时需要机械平衡。另一个产品，Knogo 北美公司（Hauppauge, 长岛, 纽约）的“CD-Guard（警卫）”也有同样的机械平衡缺陷。在美国专利第 5,347,508 号披露了一种包含一嵌入式、通常为环型的、双状态 EAS 标记的光信息存储盘片。

其它媒体

应当注意本发明的原则除了上面讨论的电子存储媒体外还可以应用于其它类型的媒体。盘状记录媒体（此处是光盘）记录和复制信息信号都使用激光束，现在商业上可以获得所谓的记录音频数据的压缩盘、记录计算机资料的 CD-ROM、信息信号可以被记录一次的一次写入光盘、可以复制、记录和删除信息的可记录光盘。

只读光盘，例如 CD 或 CD-ROM 上有不规则模式的轨道，即根据所记录的信息在其表面形成的同心圆或散射状的相位凹点。特别的，只读光盘包括由透明的合成树脂，例如聚碳酸酯或 PMMA（聚甲基丙烯酸甲酯）制成的盘片基础板、由金属如铝或金形成的反射



薄膜以便覆盖在盘片基础板表面的相位凹点、以及一个保护层以便覆盖并保护反射薄膜。

当信息信号从只读光盘复制时，来自激光源的激光束从盘片基础板一侧被物镜会聚到只读光盘。在光盘上相位凹点产生的反射光被光检测器检测，并转换成具有与反射光的通量强度对应的信号级别的检测信号，这样允许信息信号的复制信号记录到所要得到的只读光盘上。

尽管只读光盘可以在市场上提供大批量不昂贵的产品（光盘），但它不适合小批量需求的产品。为了此目的，一次写入光盘可适合小批量光盘产品，大量的资料可以容易地提供给用户。作为一次写入光盘，现有利用颜料的物理化学变化的一次写入光盘记录系统，单层孔一次写入光盘记录系统，多层孔一次写入光盘记录系统，相位改变一次写入光盘记录系统和冒泡式（bubble-forming）一次写入光盘记录系统。在复制时，以与只读光盘类似的方式，来自激光源的激光束（有较弱的复制激光能量）在被一物镜会聚的情况下从盘片基础板一侧照射到盘片上。然后，由预先记录的凹点调制的反射光被光检测器检测到，检测信号被转换成具有与反射光束强度对应的信号级别的检测信号，在那里使信息信号的复制信号存储在一次写入光盘上。

当信息信号被记录在上述一次写入光盘时，来自激光源的激光束（强记录激光能量）经过物镜会聚后从盘片基础板一侧照射到盘片。然后，该激光束的能量根据信息信号的不同或开或关来调制激光束，沿光盘的记录轨道形成对应信息信号的凹点（这些凹点与记录在只读光盘的凹点类似）。特别的，在单层孔记录系统的情形下，在记录轨道的被强激光束照射的区域形成孔，这些孔就是所记录的凹点。在多层孔记录系统的情形下，在记录轨道的被强激光束照射

的区域形成孔，例如在第一层的薄膜，在第一层的这些孔就是所记录的凹点。

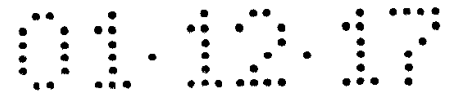
在相位改变记录系统中，在记录轨道的被强激光束照射的区域从无定型状态变为晶体状态，被改变为晶体状态的区域就是所记录的凹点。在冒泡记录系统的记录轨道中，被强激光照射的记录层区域隆起，该隆起区域就是所记录的凹点。

特别地，在一次写入光盘中，要形成引导槽（预开槽部分）以便允许对激光束的跟踪控制。在预开槽的对面形成一个端面为正弦波形（通常称为摆动波形），该波形沿轨道具有预先确定的振幅和预先确定的周期。当摆动波形被激光束检测到时，可以用获得的摆动信号作为绝对的时间信息。摆动波形用于控制记录和复制设备系统，特别是在光盘上记录凹点的定时信息。进一步，摆动信号被用于光盘旋转和驱动方式的伺服控制，例如轴电动机。根据伺服控制操作，对轴电动机的旋转速度进行控制，这样使摆动信号的周期成为常数。

上述一次写入光盘通常具有槽记录 10 系统，其中凹点被记录在预开槽区域。当记录要记录在一次写入光盘上的信息资料时，根据通过光学检测在预开槽区域形成的摆动波形获得的摆动信号的周期，同步搜寻目标位置。当目标位置被检测时，上述要记录到一次写入光盘上的信息资料就根据预定的格式被记录到目标位置。

另一方面，在复制时，目标位置按上述方法搜寻。当目标位置被检测到时，根究插入到要被记录在一次写入光盘上的资料中的帧同步信号，继续读出例如 2 千字节的数据，从而复制记录的数据。

因为只读光盘和一次写入光盘有同样的复制原则，如上所述，即使当一次写入光盘被装载到复制来自只读光盘的信息信号的复



制设备上，记录在一次写入光盘的资料也可以被复制，与只读光盘没有区别。

另外，一次写入光盘允许大量光盘可以方便的使用相对简单的设备复制。因此，一次写入光盘可能被非法复制。特别地，有一个计算机系统，其中用于从只读光盘复制信息信号的复制设备被连接到最终用户的微机的外部输入和输出终端。例如，一次写入光盘记录和复制信息信号的外部存储设备被连接到另一个外部输入输出终端。然后，被复制设备从只读光盘读出的记录资料全被外部存储设备写入到一次写入光盘，这样就产生了只读光盘的盗版盘。

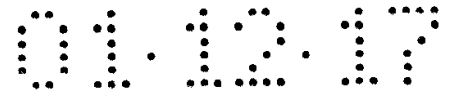
在这种情况下，如果只读光盘是记录计算机资料（包括计算机程序）的 CD-ROM，那么盗版的游戏软件可以很容易地生产。如果只读光盘是记录音乐信息的 CD，那么生产盗版 CD 盘就有可能了。

因为计算机程序是被知识产权保护的有版权的产品，除了那些正规的用户，即接受了软件许可证协议的注册用户之外的拷贝或复制到硬盘都是非法的。

而且，把拥有版权的 CD-ROM 中记录的资料整个的拷贝到一次写入光盘用于发布等目的也是非法的，这些获取不正当利润的非法行为也应该被禁止。

进一步，正规用户免费分发光盘给一个企业或 CAI（计算机辅助教学）中的非正式用户的行为也被认为是严重的侵权。

目前，有许多防止复制的提议的方法已经用于实践。另一方面，称为“复制工具”的用于解除复制保护的软件（程序之类）也可以买得到。除了用户自己的良知之外，目前没有其它方法防止非法复制记录的资料。



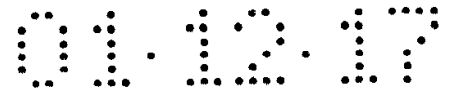
如前所述，本发明的一个目的是提供一种资料记录方法，其中盘类记录媒体之间的非法拷贝即使有复制工具可以有效的防止，而且可以保护记录在盘类记录媒体上的有版权的内容（记录资料）。

交互式产品允许计算机系统的用户与电影、视频、或其它可显示的以较快速率更新的图像互动。这些产品的目的在于给用户提供有用的信息、教育或娱乐。交互式技术的最终目的是使用户感到好象他们在与电影屏幕上的图像互动，从而使得在剧中的角色或事物与人的行为相互起作用。用户的行为可以影响角色、事物或在显示屏幕上的其它图像，从而改变故事的叙述过程。

提供高度互动性的一个方法是使产品完全由计算机产生。这意味着计算机设计三维世界模型并计算和显示屏幕上的图形和对象的方向。但是，该方法受限于目前的技术，因为完全计算和提供活生生的图像，尤其是人类图像所需要的计算能力超出主流市场系统的目前技术水准，这些图像的分辨率要求在实时视频或电影刷新速率的情况下接近电视质量。

另一个不同的方法是预先记录视频、电影或计算机产生的图像序列，然后以高速重放这些预先记录的图像或画面。这可以达到或超过电视的分辨率，而且足够生动可以建立可与电视媲美的可信度。但是，该方法大大限制了用户与产品的互动性，因为用户影响故事情节的能力被限制为预先记录好的小量的不同图像序列“路径”，这些路径是视频或动画序列中预先确定的决策点的分支。重放任何预先记录的图像序列来达到允许用户与这些图像互动的动画技术广义的称为“交互式视频”。

典型的交互式视频产品使用只读存储光盘（CD-ROM）来存放图像，在重放时用 CD-ROM 光驱提取图像。CD-ROM 盘在光媒体上以同心螺旋方式存储信息，用具有激光束“读取头”的 CD-ROM 光驱来“读取”或重放信息。基于 CD-ROM 的交互式产品的大问



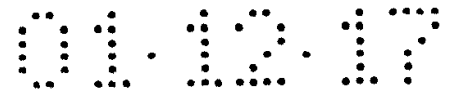
题往往发生在需要查找预期的分支路径时，而该分支路径又与当前路径不在一个光驱的读取头跟踪轨道上，即需要半秒或更多时间的延迟，这就破坏了播放的连续性。另一个问题是基于 CD-ROM 的交互式视频产品严重的限制了用户与视频互动的类型和数量。

取用不同的视频路径的时间长度（称为“读取时间”或“寻找时间”）依赖于与当前 CD-ROM 光驱读取头所放置的位置相对应的各个视频路径的位置。为了取用给定的视频序列，计算机控制器在索引中查找序列并指导 CD-ROM 光驱移动读取头到盘片上新序列的开始来读取新序列。因为读取头通过一机械机制移动，故相对而言需要较长的时间来将读取头复位到轨道上的新点，从而读取不同的视频路径。

现有技术是用高速缓存来改善在 CD-ROM 光驱上取用资料的性能。高速缓存可以在 CD-ROM 光驱中、在处理器和光驱之间的界面卡中、在软件控制的计算机系统的内存中、甚至在硬盘或其它存储媒体中。但是，这些高速缓存只能对视频所关心的取用时间提供有限的改善，因为与来自 CD-ROM 的信息的资料速率相比，缓存的体积相对要小。而且，当信息对应不同的路径时，缓存几乎是无用的，因为缓存中没有包含新资料。缓存必须被“清除”并装载新信息。

尽管目前的 CD-ROM 光驱不足以在交互式视频产品中提供足够的互动性，但它们提供了巨大的安装基础因为数十万的 CD-ROM 已经被销售给消费者。因此，人们希望有一个系统能够消除基于 CD-ROM 的交互式视频的读取时间而不需要改变现有的 CD-ROM 光驱。

传统上所谓的 LD（激光影碟）和 CD（光盘）通常都被称作光盘，可以在其上面记录诸如视频信息、音频信息之类的信息。在 LD



或同类物中，视频信息和音频信息与时间信息记录在一起，时间信息指示出信息相对于复制起始位置的被复制的时间，复制起始位置是 LD 或同类媒体的标准位置。因此，除了通常的正常复制记录信息的记录顺序之外，各种特别的复制方式也是可能的，例如在 CD 中，可以有各种复制品，如从各种记录的音乐中提取和聆听所喜欢的音乐，以任意顺序聆听记录的音乐等。

但是，上面提到的 LD 等同类媒体存在一个问题，即不可能有所谓的交互式的和多种多样的复制品，在后者中听众或观众对播放的视频或音频信息有各种各样的选择分支，听众可以选择观看或聆听。

也就是说，例如在观看 LD 上的外国电影时，不可能选择一种语言用于放映的图片画面的字幕（例如，选择日语字幕和原文字幕中的一种）以便以所选择的语言来显示字幕。或者，当聆听 CD 上记录的音乐时，不可能任意选择一种音乐的声音（如选择英文歌词和日语歌词中的一种）。

另一方面，人们提出了各种建议和发展，DVD 作为一种光盘，与传统的 CD 比较，其存储容量被改善了大约十倍而没有改变光盘的大小。对 DVD 而言，如果多种语言的字幕或多种语言的声音被记录进去，上面提到的交互式和多样性复制品就可能提供给用户选择。

但是，如果在上述 DVD 上存储了各种语言的音频或声音信息或者各种类型的音乐，音频信息或音乐信息的信息量就会变得十分庞大。此时，如果信息不以某种合适的记录形式记录，查找音频信息来复制的过程就会很复杂，音频或音乐在复制中间可能由于需要在复制时查找音频信息而发生中断，这就是一个问题。

虽然上面已经说明了各种实施例，但是应当理解的是它们只是通过实施例的方式来介绍，而非限制。因此，优选实施例的范围不应该限制在上述所举出的例子，而只应按以下所提出的权利要求及其等效设计加以定义。

说明书附图

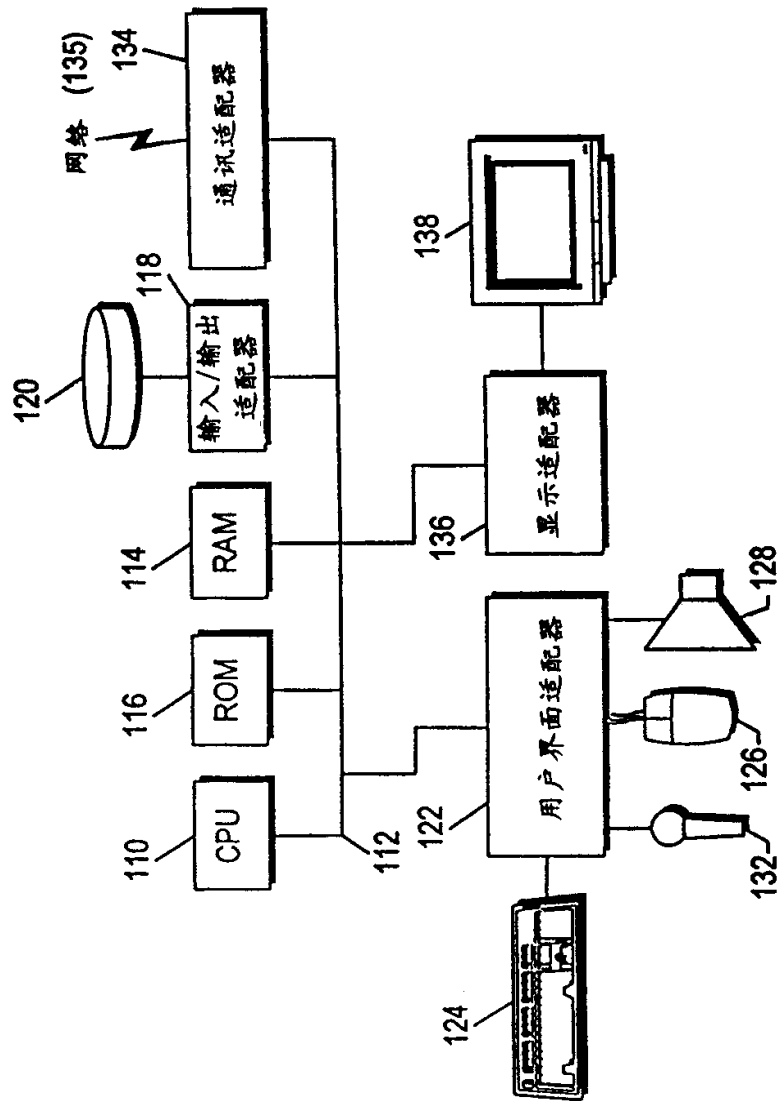


图 1

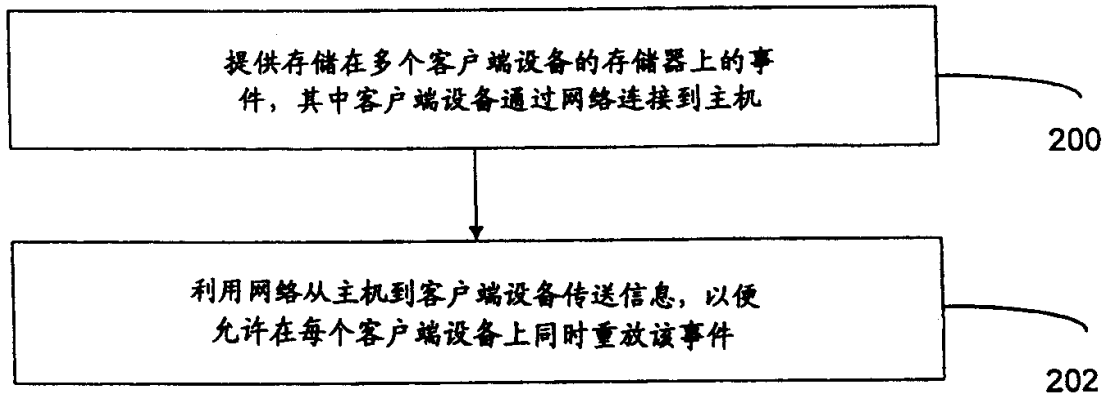


图 2

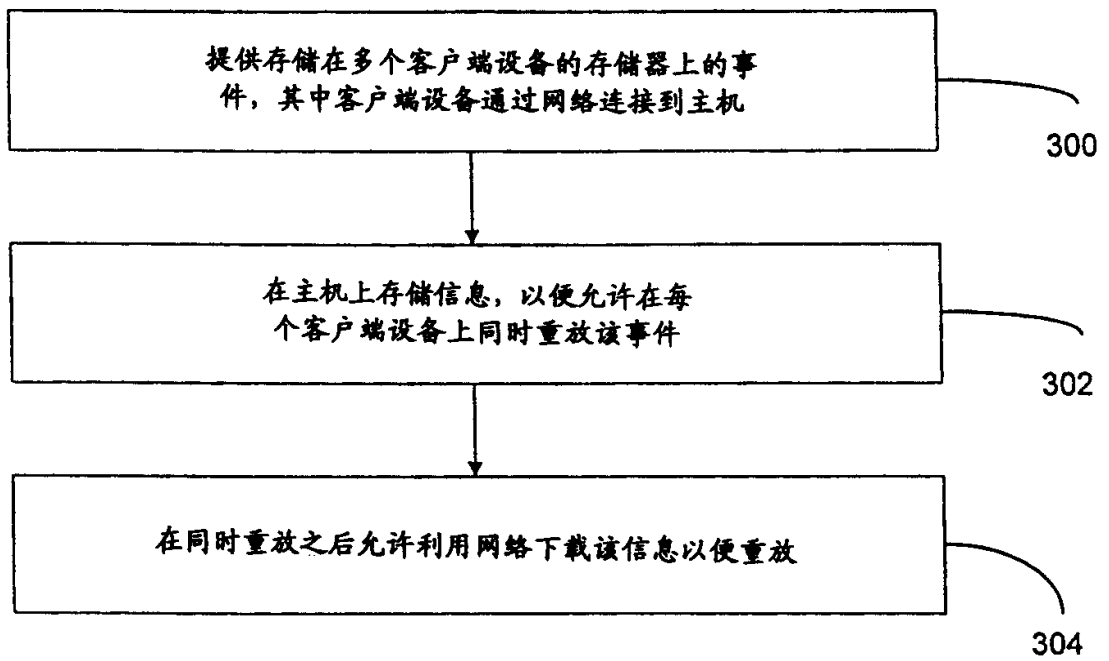


图 3

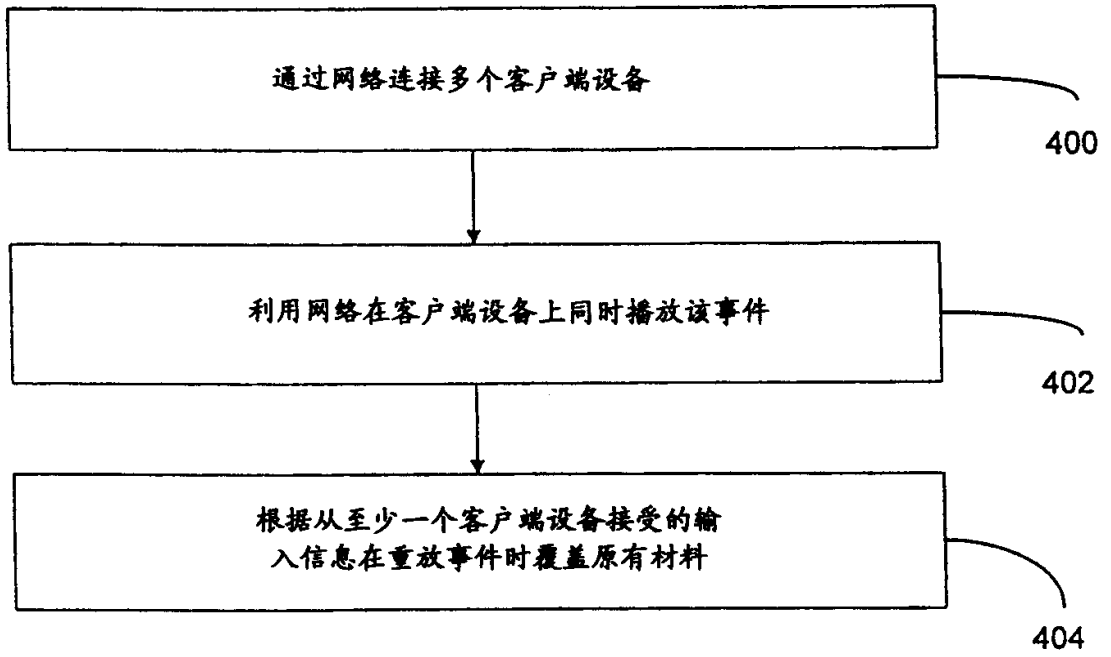


图 4

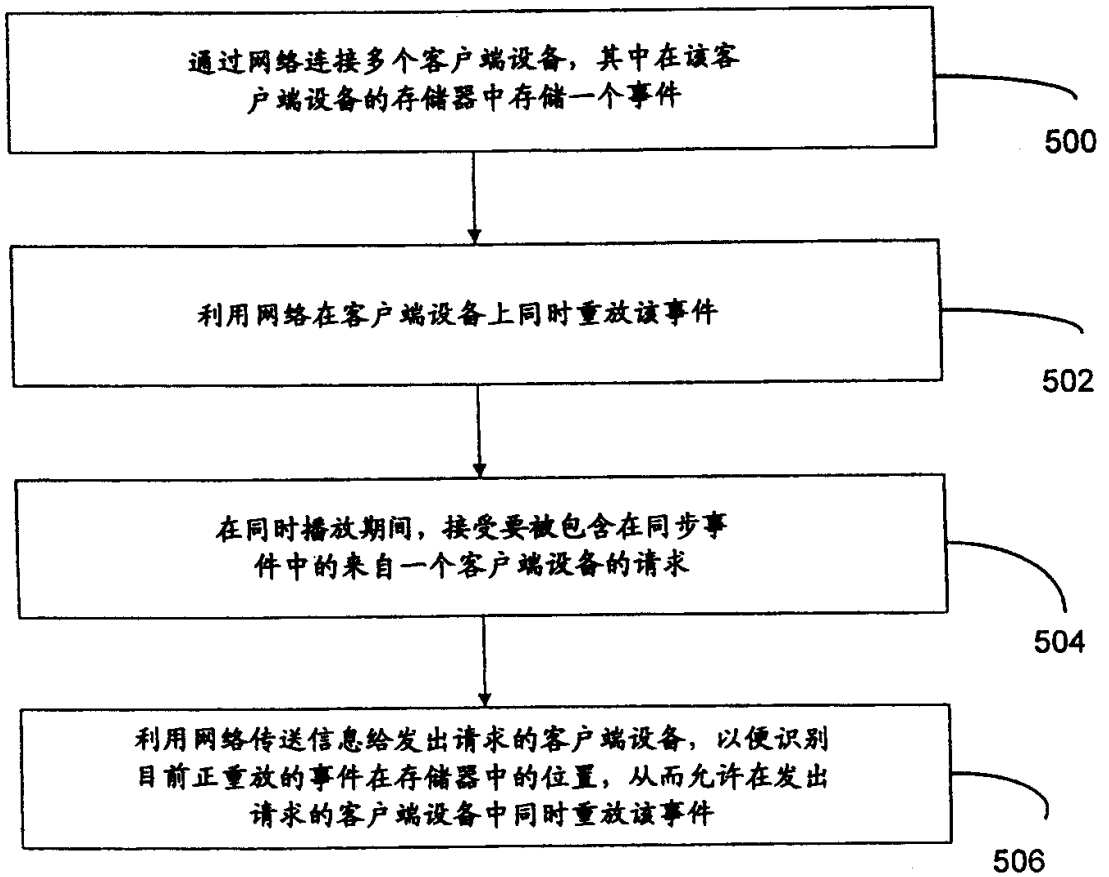


图 5

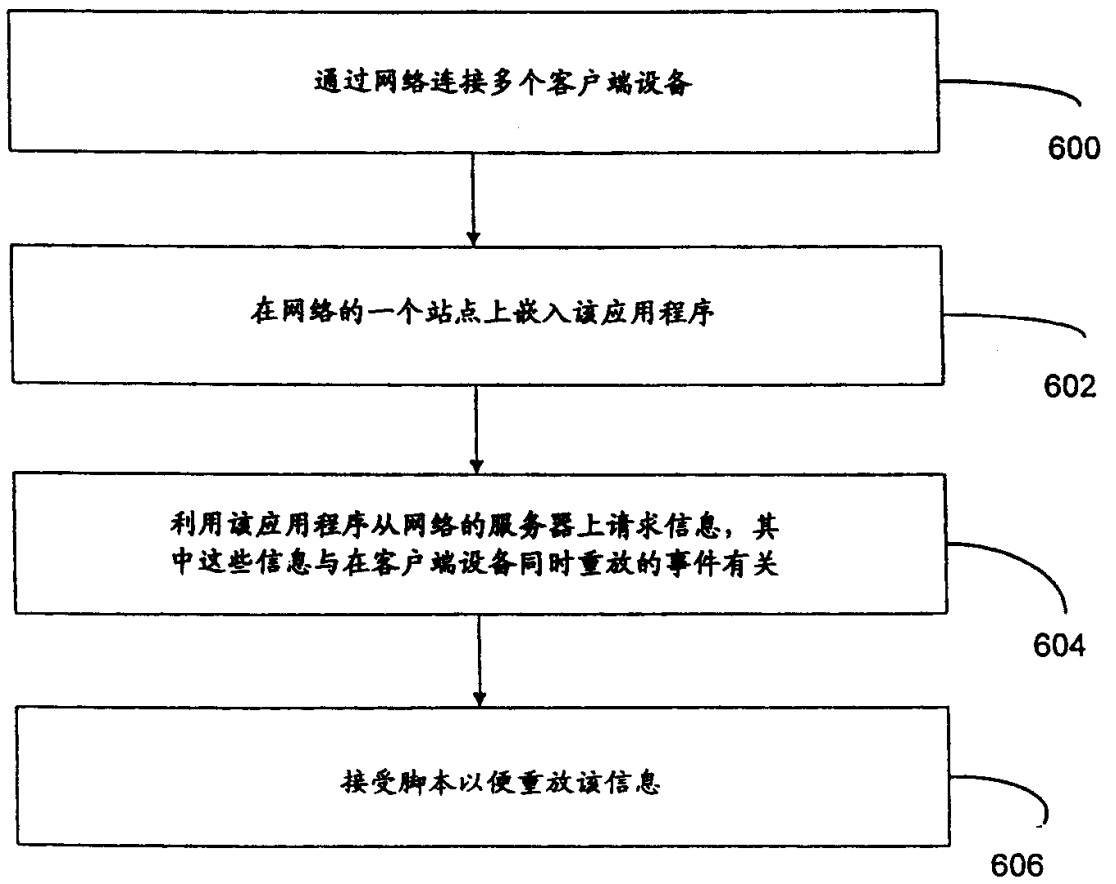
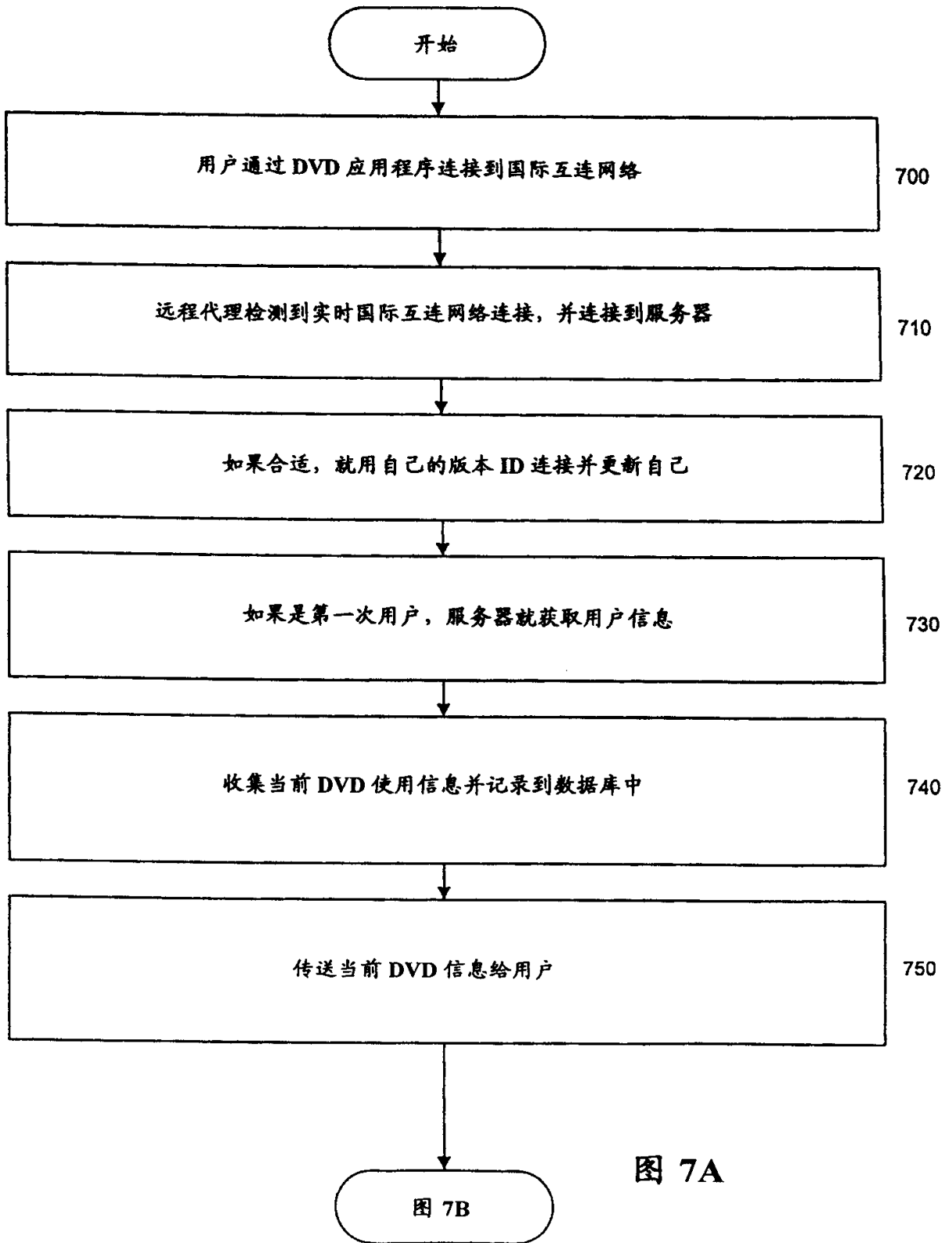


图 6



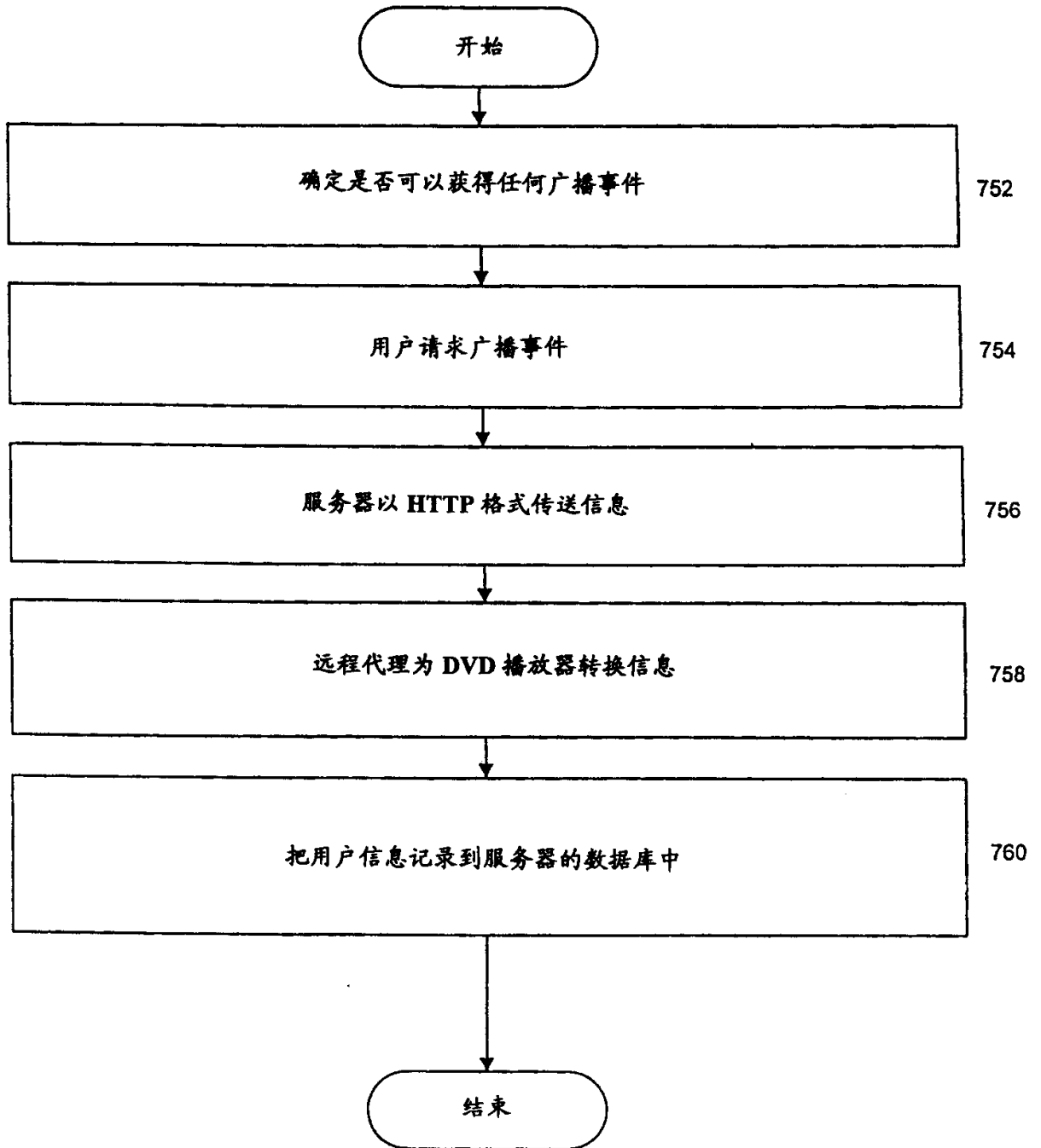


图 7B

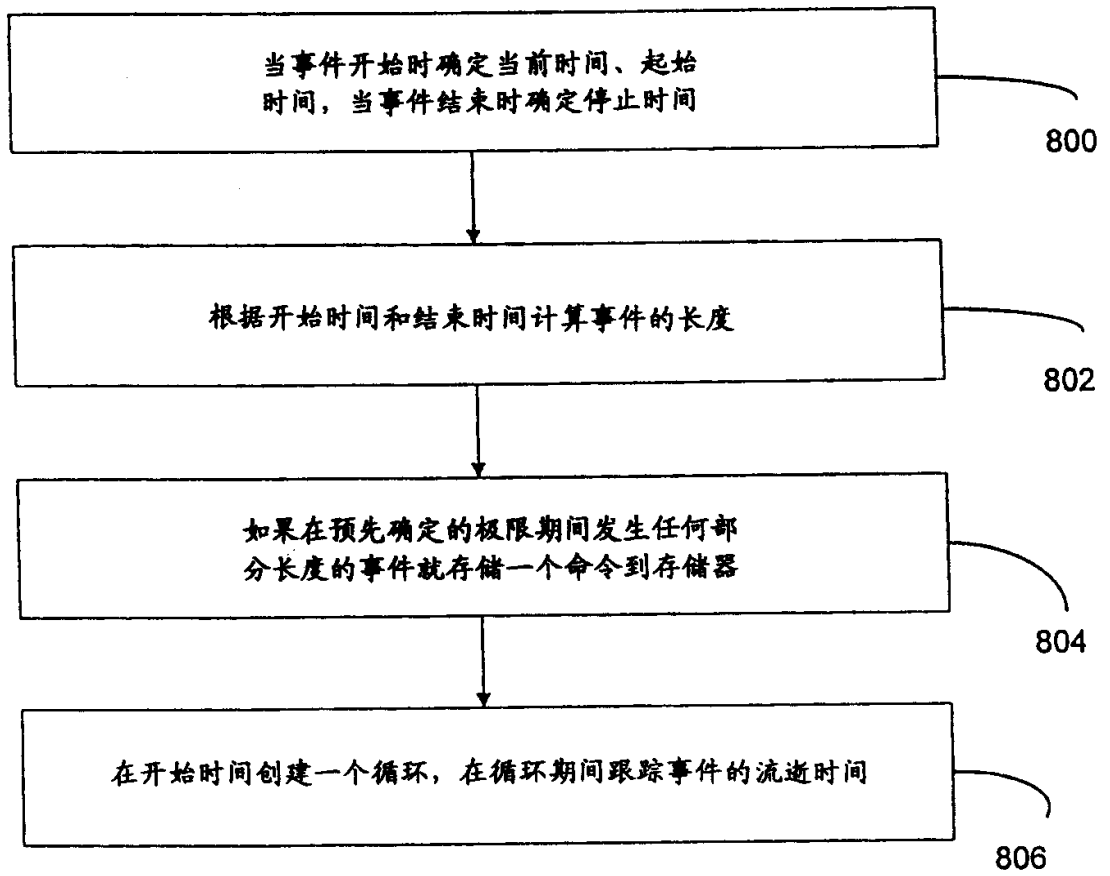


图 8

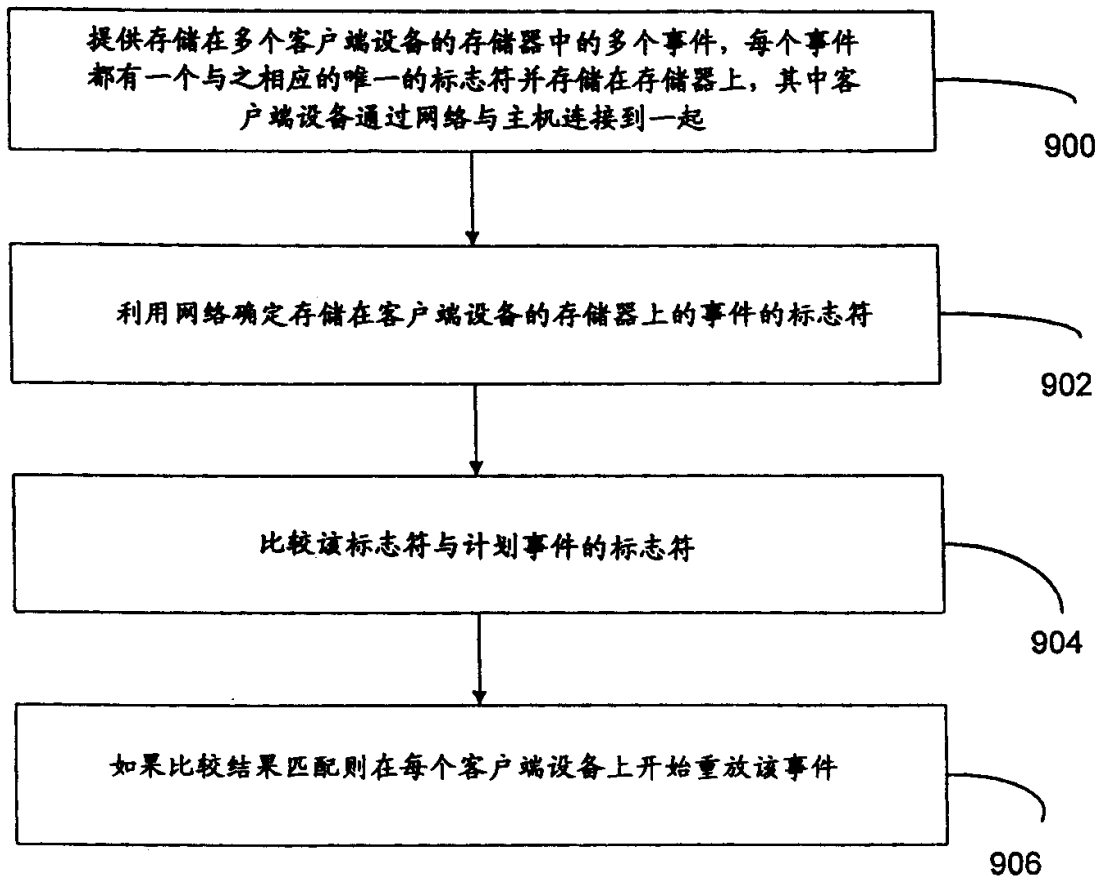


图 9

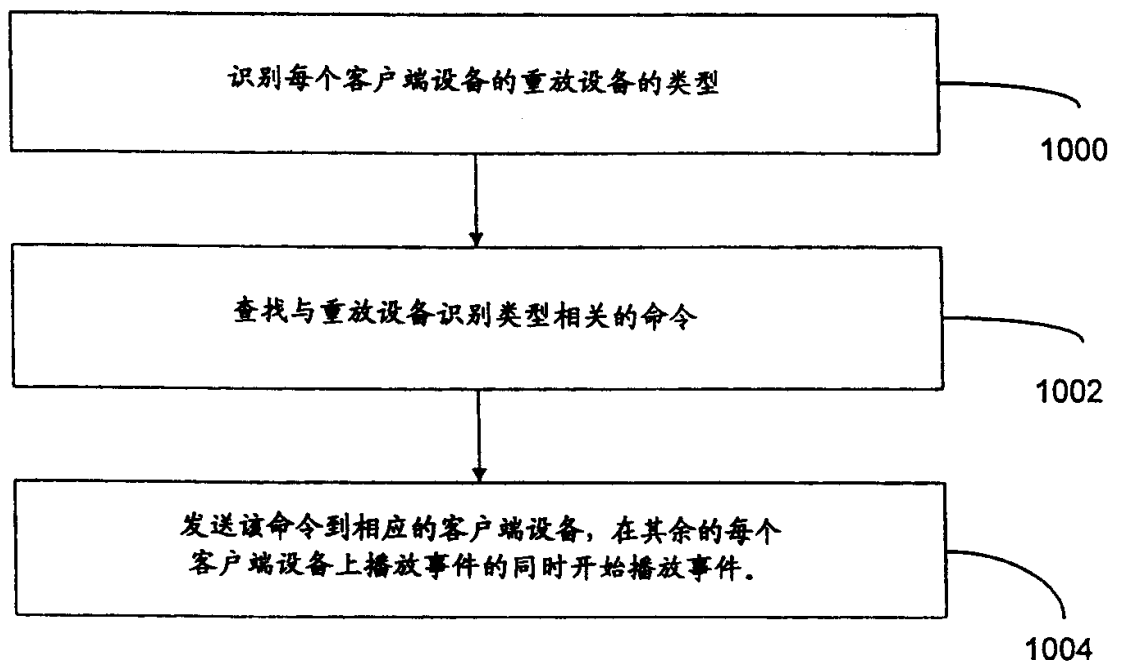


图 10

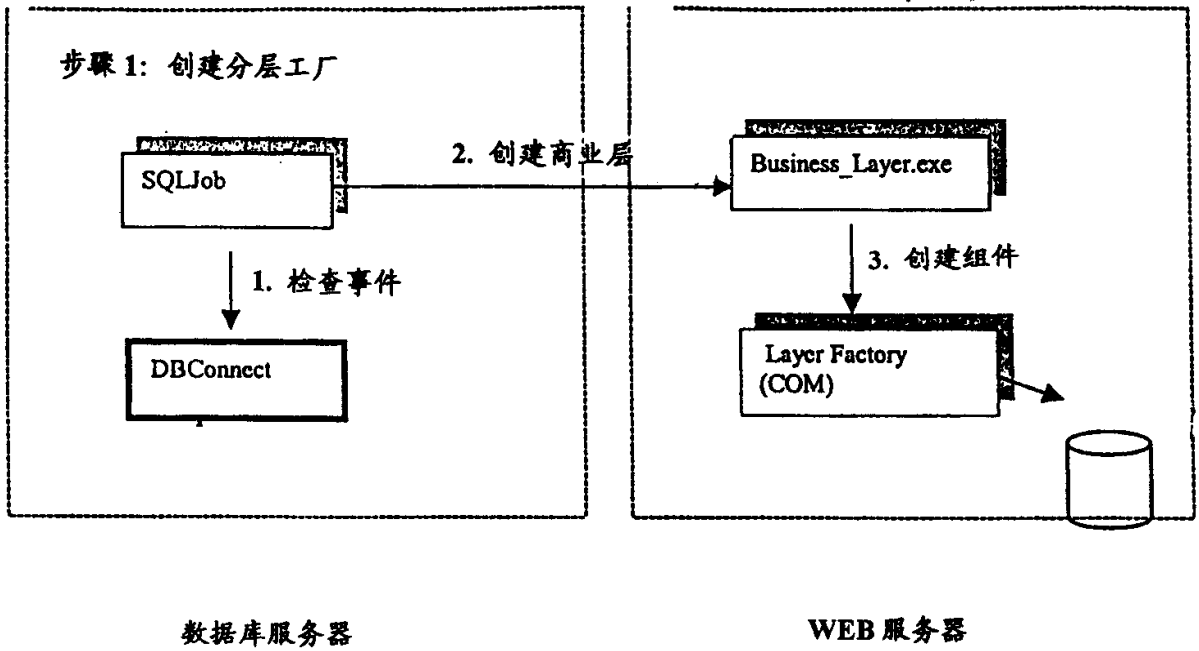


图 11

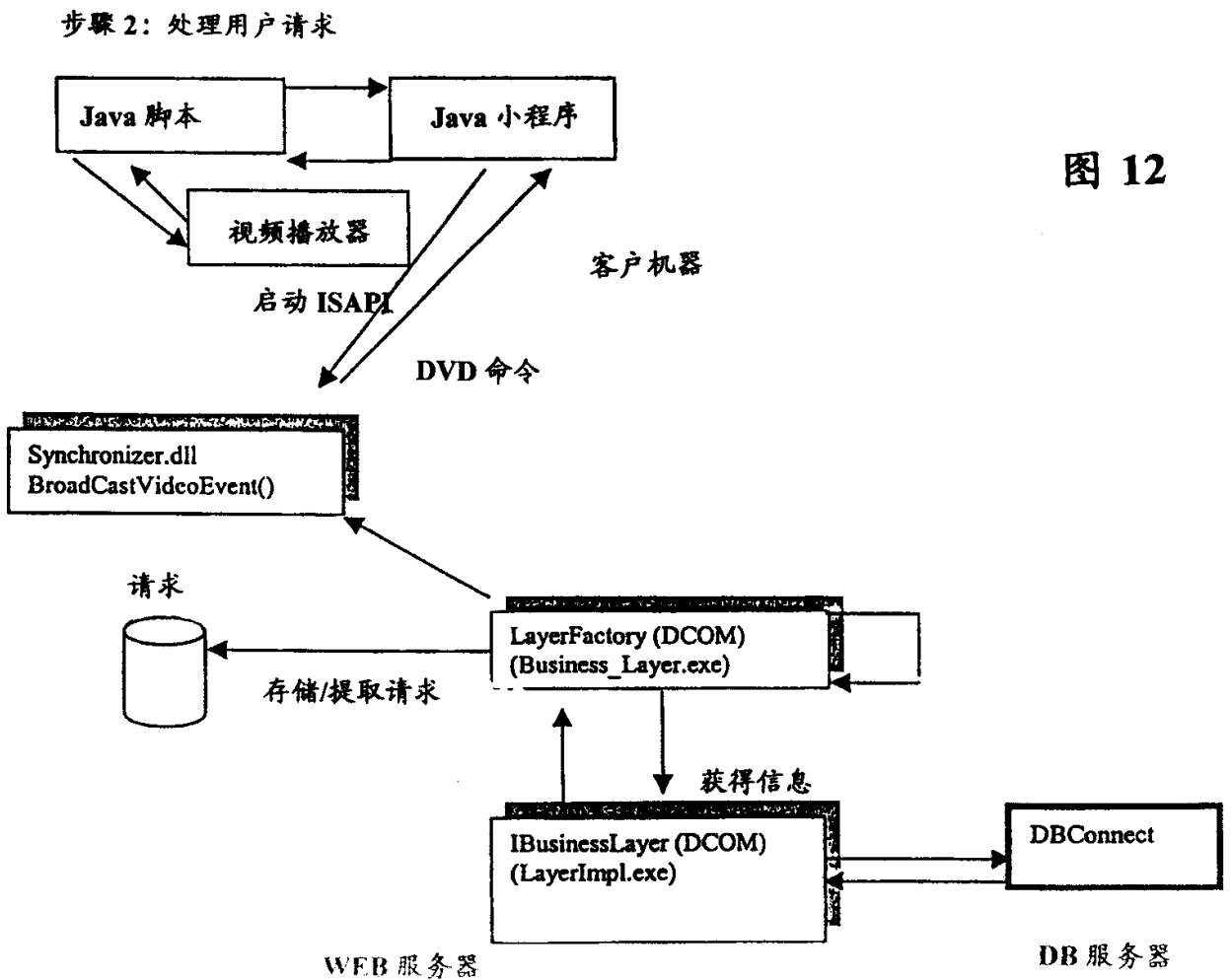


图 12

1300

同步组件

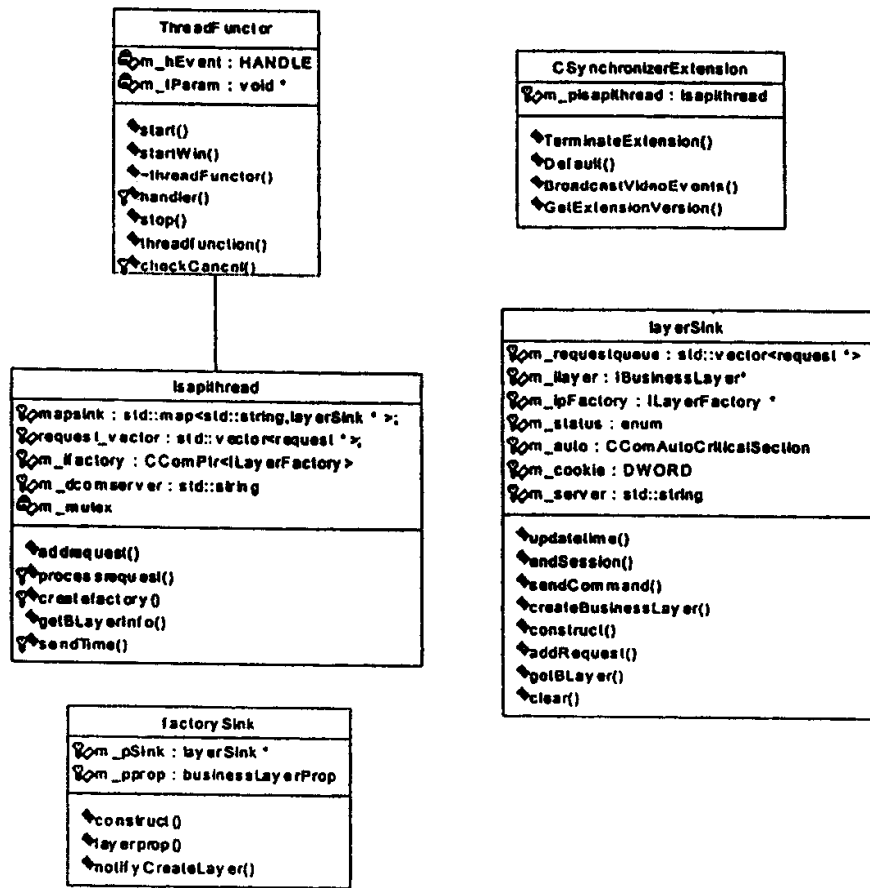


图 13

Layerimpl 组件

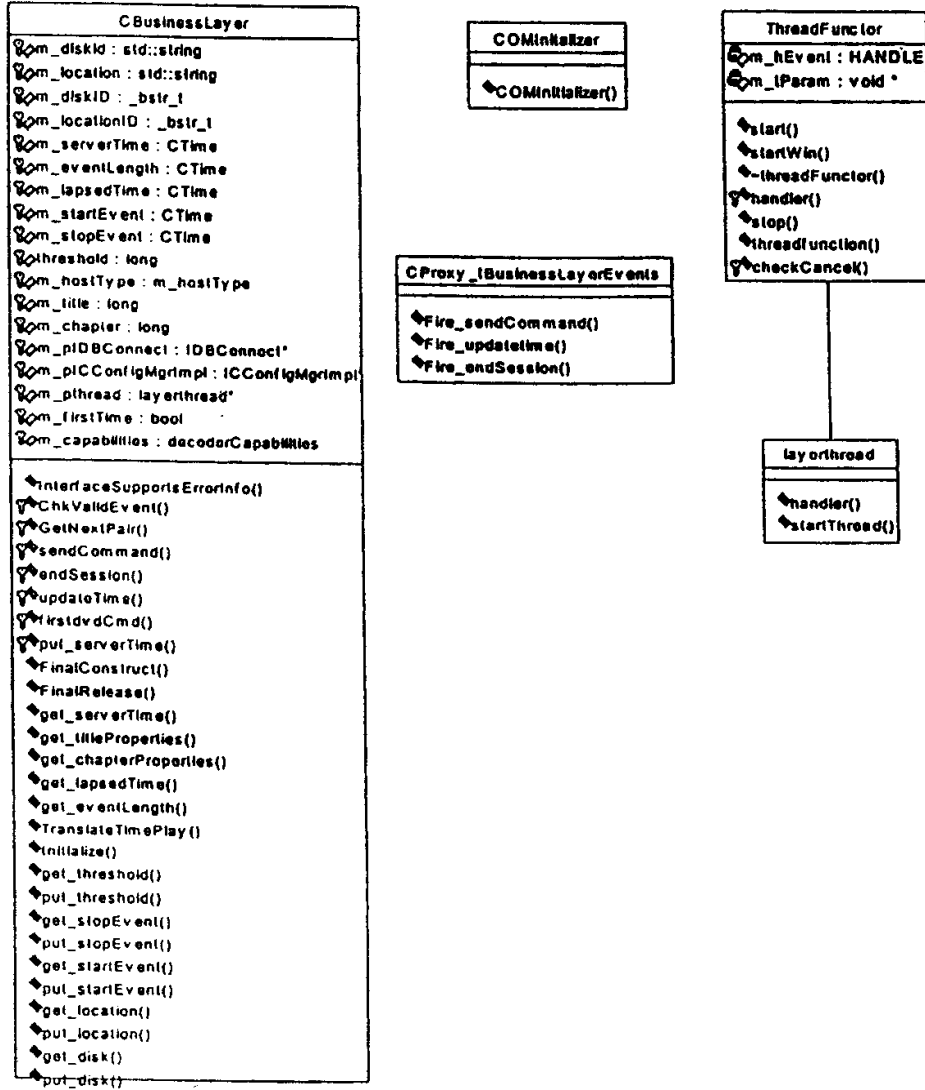


图 14

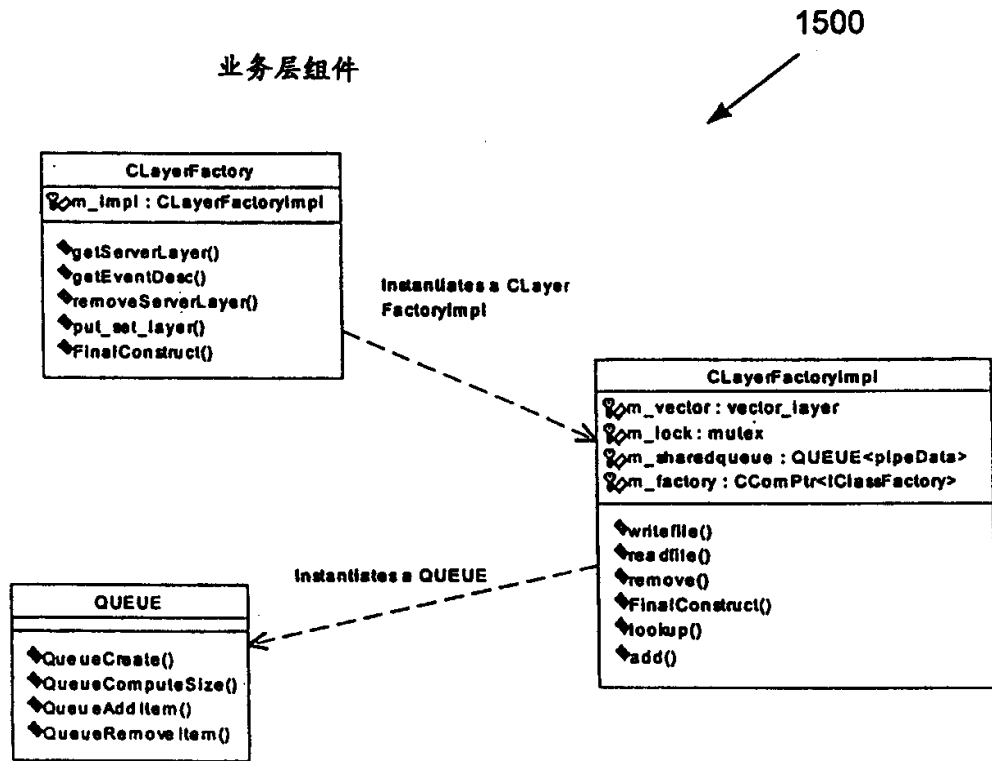


图 15

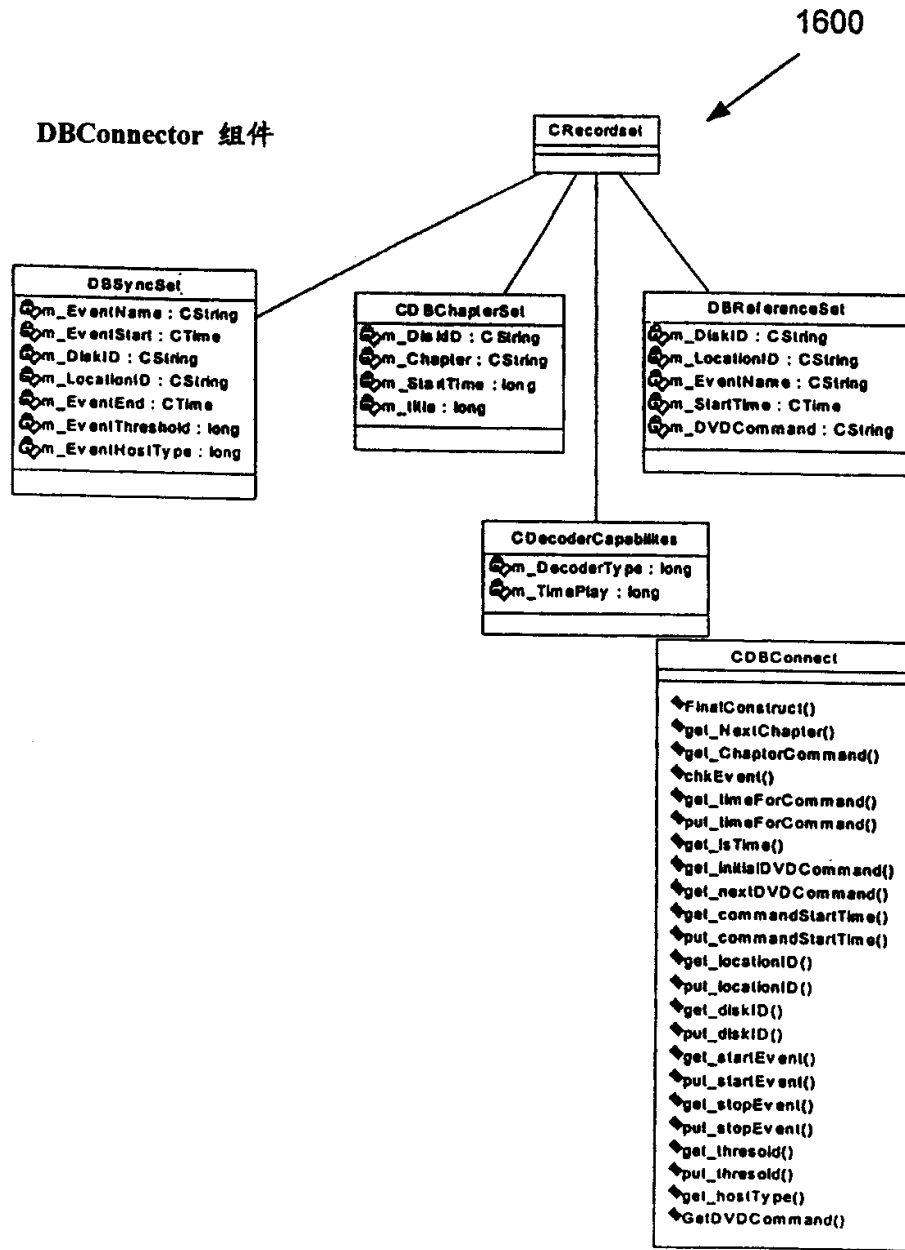


图 16

1700

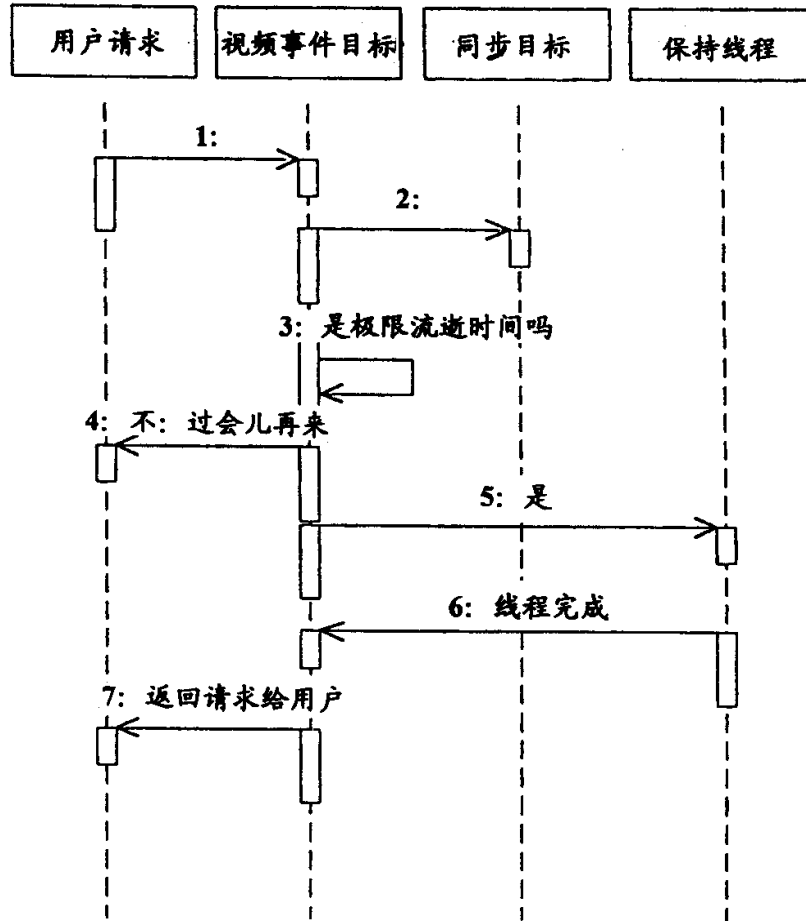


图 17

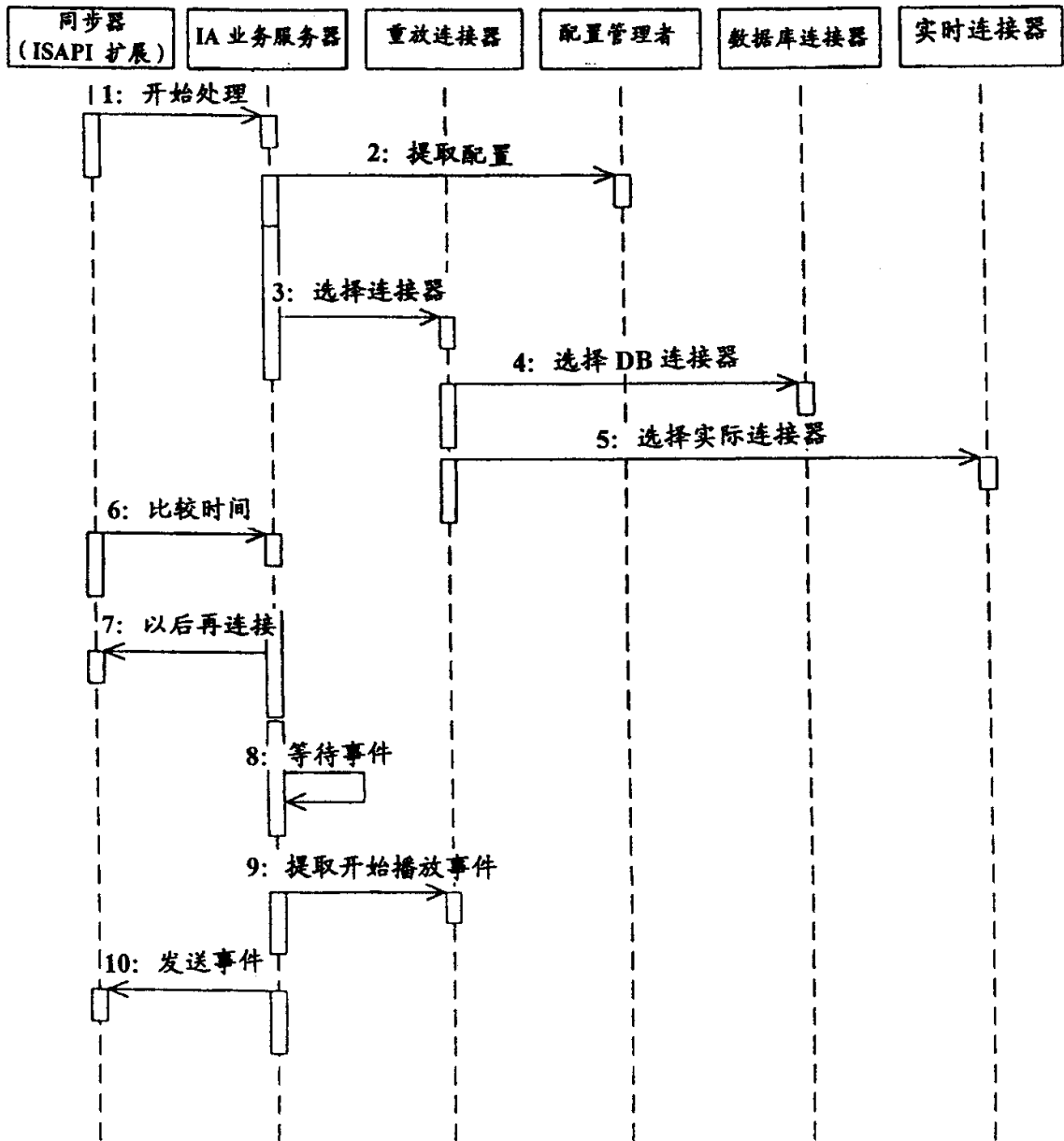


图 18

1900

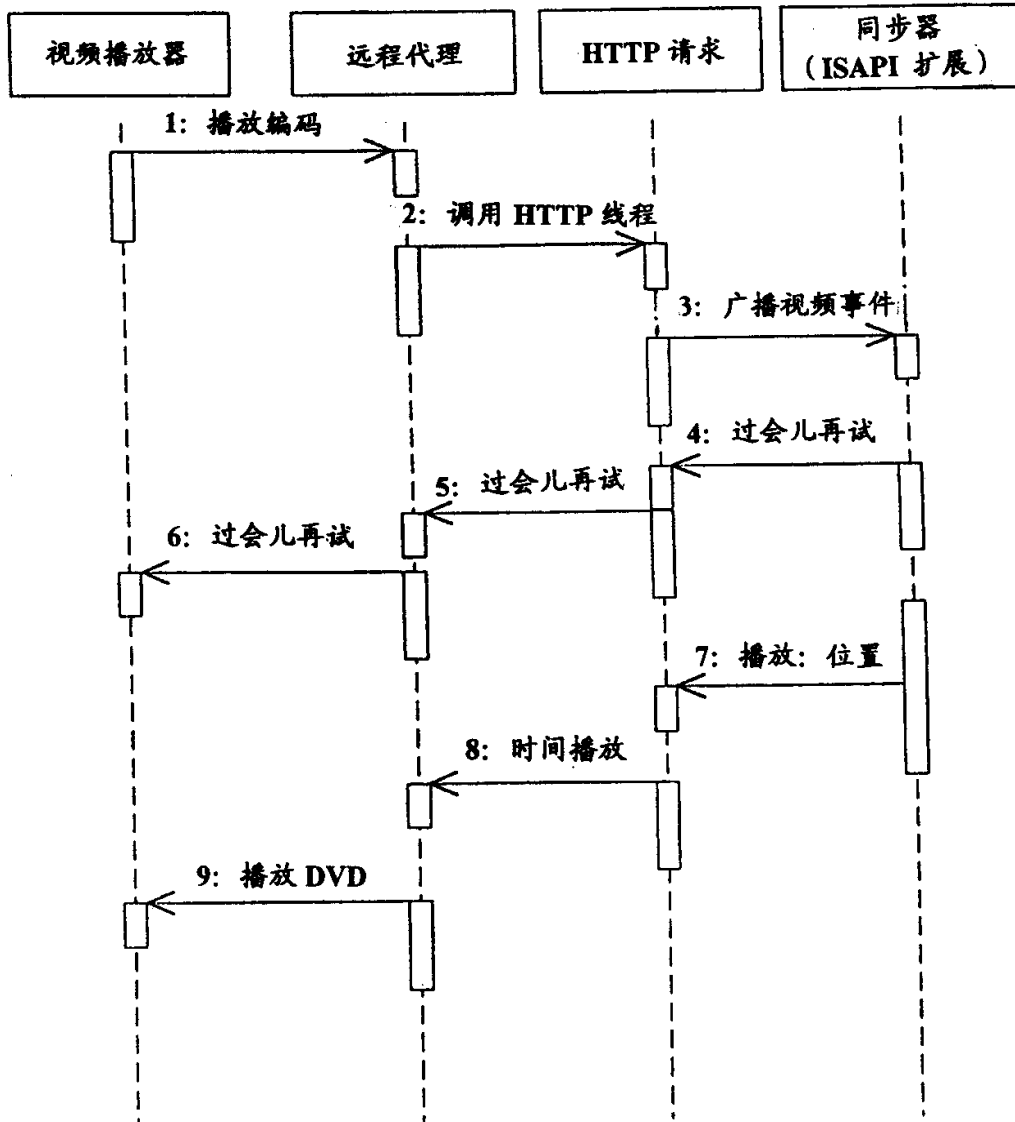


图 19

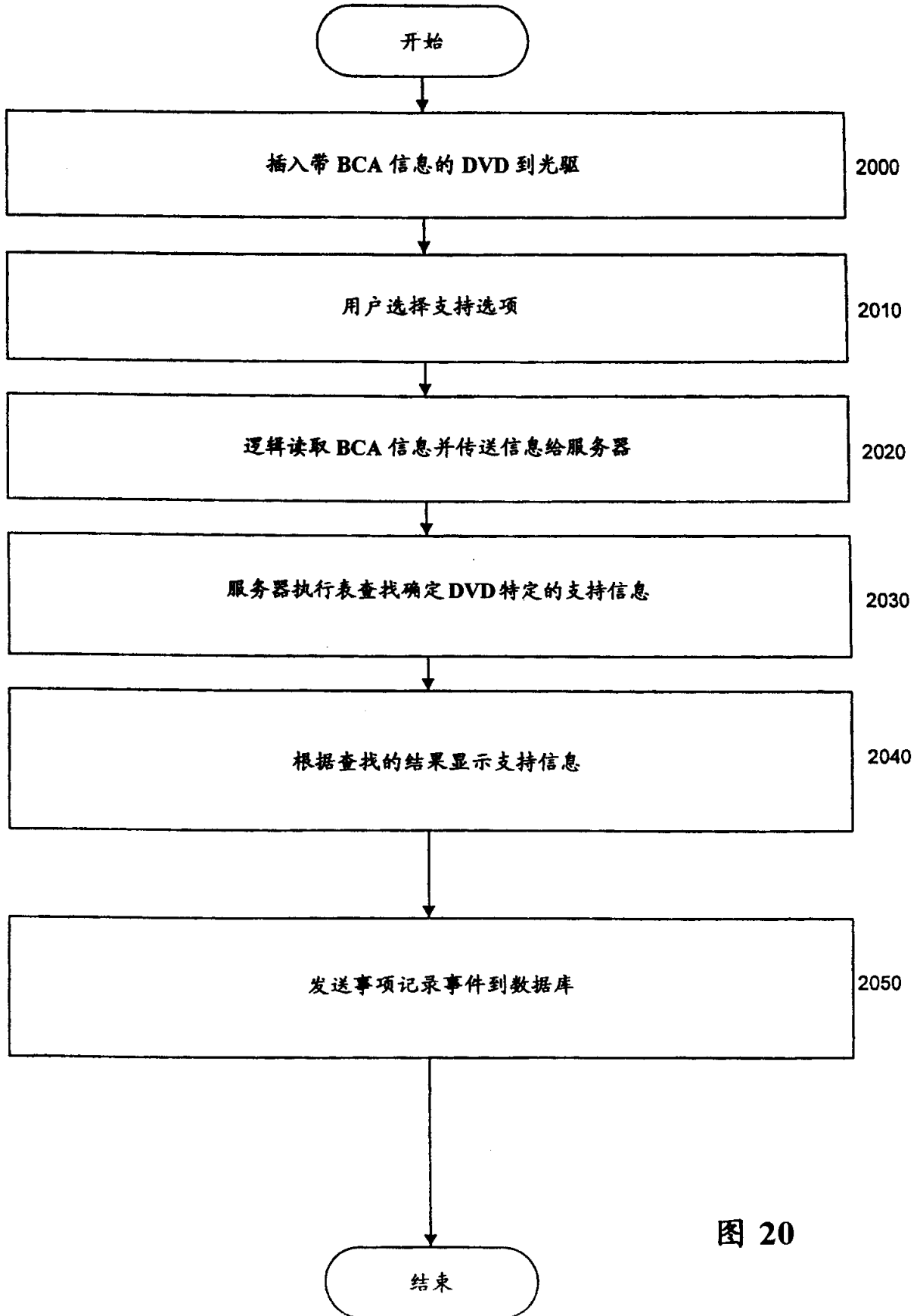


图 20

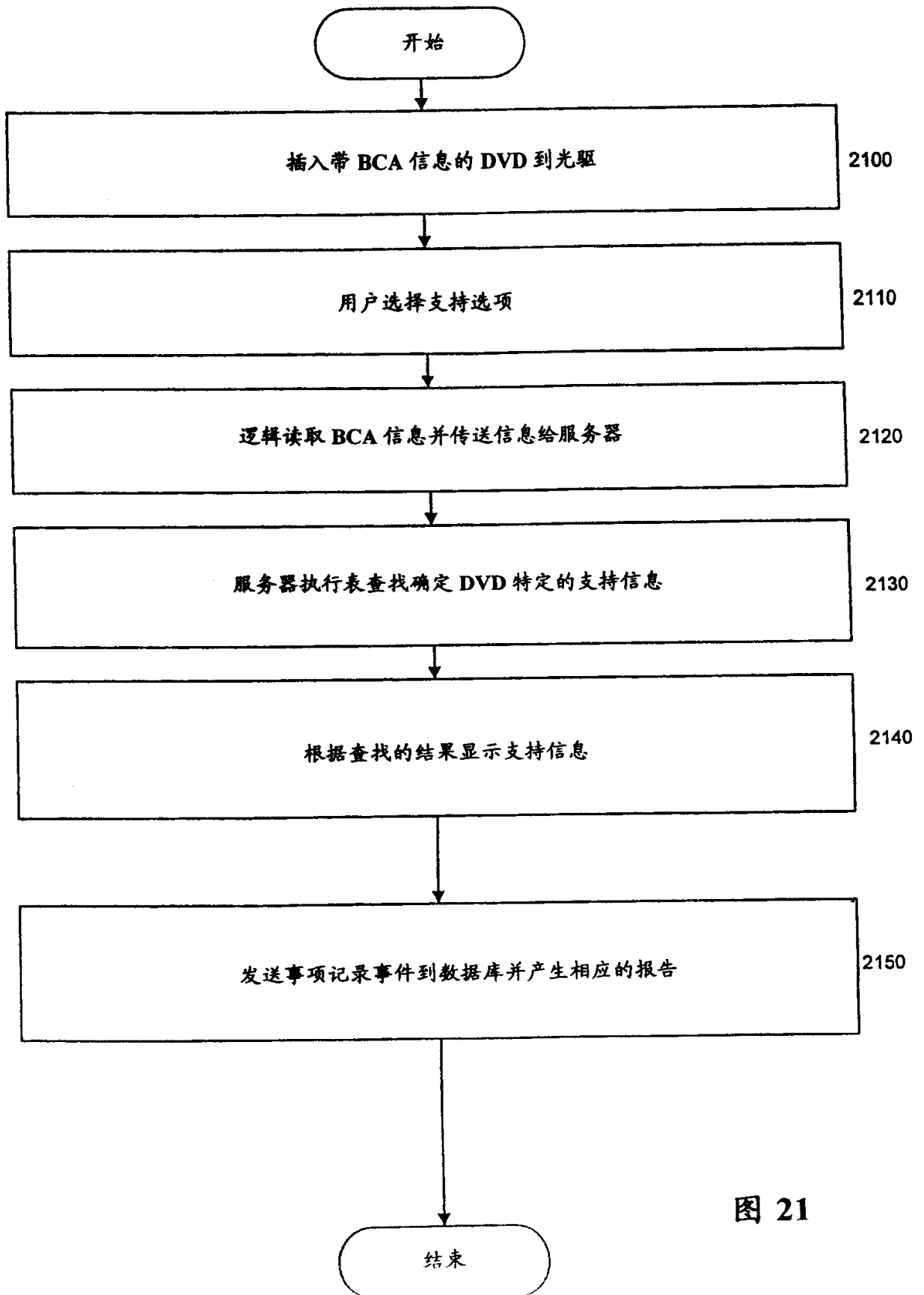


图 21

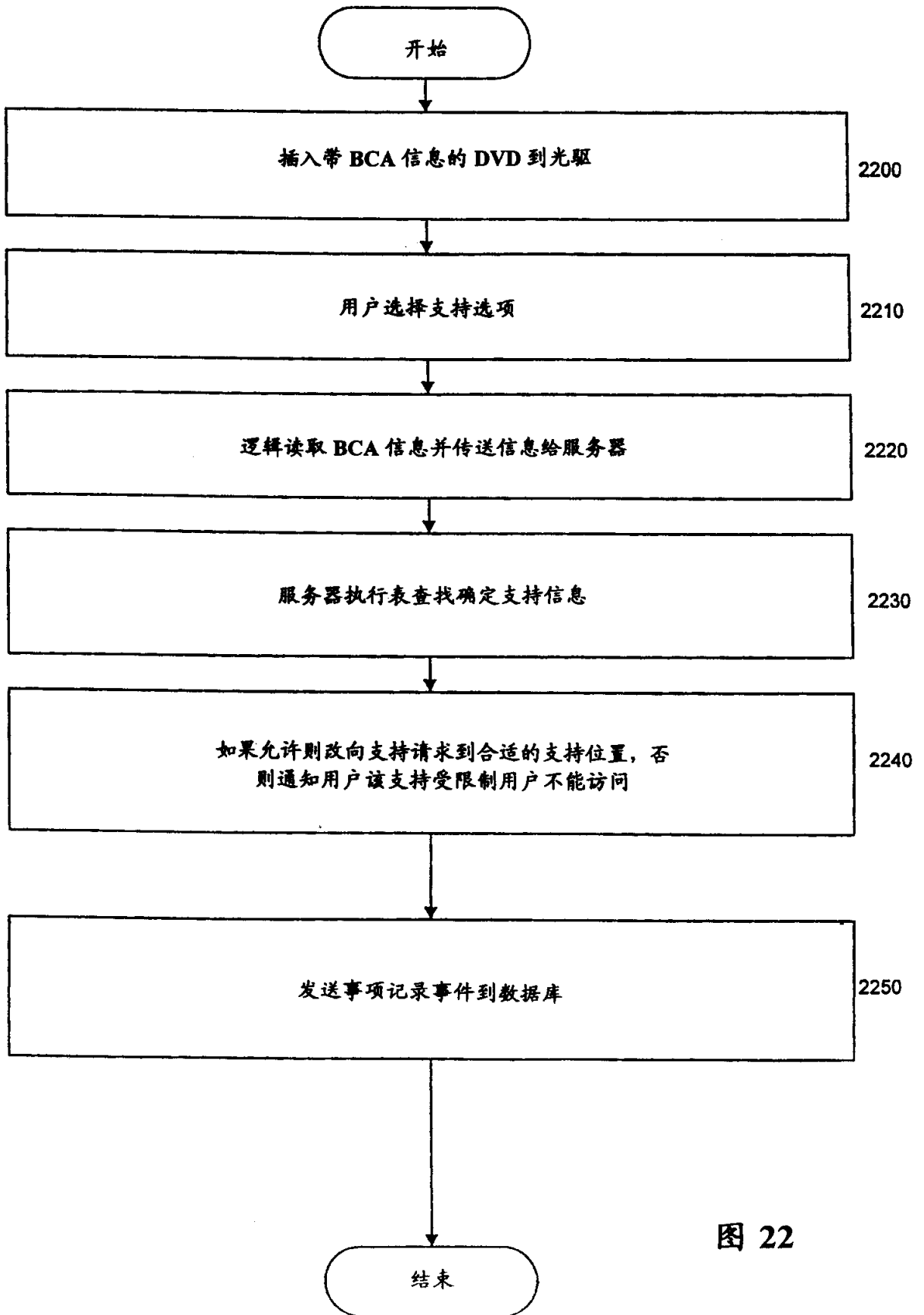


图 22

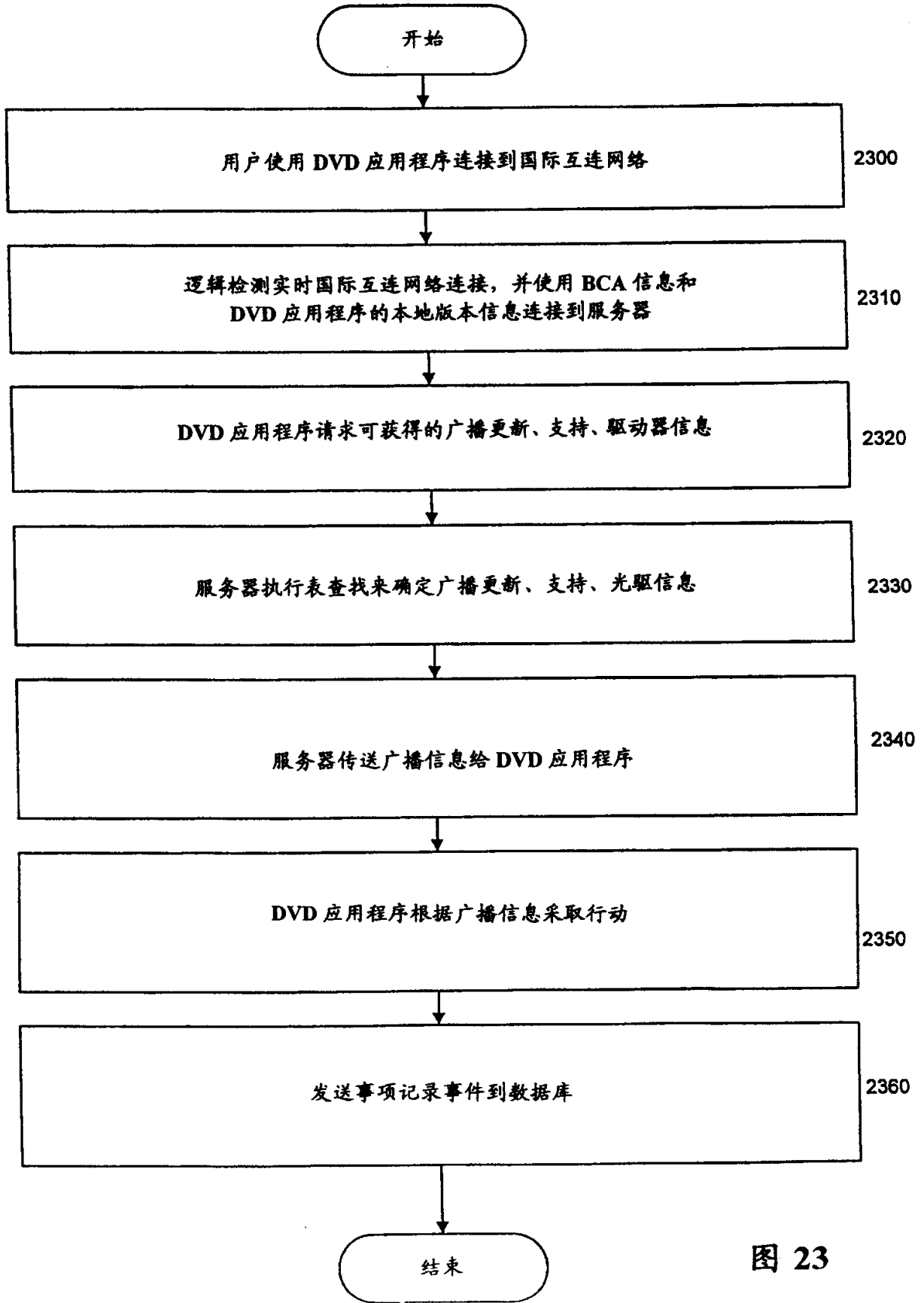


图 23