



# [12] 发明专利申请公开说明书

[21] 申请号 02823766.8

[43] 公开日 2005年3月16日

[11] 公开号 CN 1596407A

[22] 申请日 2002.10.31 [21] 申请号 02823766.8

[30] 优先权

[32] 2001.11.29 [33] EP [31] 01204599.3

[86] 国际申请 PCT/IB2002/004605 2002.10.31

[87] 国际公布 WO2003/046760 英 2003.6.5

[85] 进入国家阶段日期 2004.5.28

[71] 申请人 皇家飞利浦电子股份有限公司

地址 荷兰艾恩德霍芬

[72] 发明人 A·A·C·M·凯克

J·A·海特斯马

[74] 专利代理机构 中国专利代理(香港)有限公司

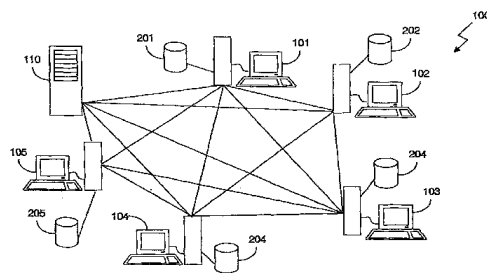
代理人 吴立明 王 勇

权利要求书1页 说明书9页 附图2页

[54] 发明名称 指纹数据库维护方法和系统

[57] 摘要

维护数据库的方法包括对于多个多媒体对象的每一个的指纹和相关元数据集。数据库的各个部分(201、202、203、204、205)被分布到与被安排用作共享所述多媒体对象数量的文件共享网络(100)相连接各个文件共享客户(101-105)。文件共享客户(101-105)能够维护它们自己的各个数据库部分(201-205),或传输指纹和元数据到另一个文件共享客户。在后一种情况中,其它文件共享客户最好是文件共享网络(100)中的超级节点。



1. 一种维护数据库的方法，该数据库包括多个多媒体对象的每一个的指纹和有关的元数据集，该方法包括把数据库的各个部分分布到  
5 与被安排用于共享所述数量多媒体对象的文件共享网络相连接的各个文件共享客户。

2. 一种文件共享客户，包括存储一个或多个多媒体对象的存储器，在存储器中与文件共享网络上的其它文件共享客户来共享多媒体对象的共享装置，用于计算指纹和为被共享设备共享的多媒体对象获取一  
10 组元数据，以及用于添加计算出的指纹和所得到的元数据集到分布到和文件共享网络相连的文件共享客户上的数据库的指纹装置。

3. 权利要求 2 的文件共享客户，还包括用于维护部分分布式数据库的 DBMS 装置。

4. 权利要求 3 的文件共享客户，其中由 DBMS 装置维护的分布式  
15 数据库部分的大小依靠运行它的计算机系统的性能。

5. 权利要求 3 的文件共享客户，其中 DBMS 装置被安排用作添加计算出的指纹和获得的元数据集到各个部分。

6. 权利要求 2 的文件共享客户，指纹装置被安排用作传输计算出的指纹和获得的元数据集到文件共享网络上的另一个文件共享客户。

7. 权利要求 6 的文件共享客户，其中其它的文件共享客户是文件  
20 共享网络中的超级节点。

8. 权利要求 6 的文件共享客户，其中该传输是和传输多媒体对象到其它文件共享客户同时完成的。

9. 权利要求 2 的文件共享客户，当多媒体对象正被存储到存储器  
25 中时，该指纹装置被安排用作计算指纹和获取该多媒体对象的元数据集。

10. 一个计算机程序产品，被安排用作使通用计算机能如权利要求 2 的文件共享客户一样运行。

11. 文件共享网络包括权利要求 2 中所要求的至少一个客户。

### 指纹数据库维护方法和系统

5 本发明是关于维护数据库的方法，该数据库包括多个多媒体对象的每一个的指纹和相关元数据集。本发明还关于文件共享客户、计算机程序产品和文件共享网络。

人类的指纹被用作识别人已经有一百多年。概念上，指纹能被看作是每个人类的唯一简短概括。近来，不断增长的兴趣呈现在处理计算多媒体对象指纹的多媒体领域。为了限定两个多媒体对象相同，不用比较多媒体对象本身，而只用比较它们的指纹。多媒体对象指纹是被讨论对象相对永恒的特征。该指纹有时也被认为是“（强壮的）散列”。

在使用指纹技术的大多数系统中，大量多媒体数据以及它们有关的各个元数据的指纹被存储在数据库中。术语“元数据”涉及比如标题、艺人、流派等多媒体对象的信息。多媒体数据的元数据通过计算它的指纹和通过所计算的指纹作为查找键或查询参数执行查找/查询数据库来进行检索。

存储多媒体对象的指纹来代替存储多媒体内容本身有多个优点。提出一些：

- 20 1. 内存/存储器对数据库的要求减少。
2. 比较指纹比较多媒体对象本身更有效，因为指纹基本上要比对象短。
3. 为匹配指纹搜索数据库比为完整的多媒体对象搜索更有效，因为它包含更短的匹配项。
- 25 4. 为匹配指纹的搜索更可能成功，因为多媒体对象的微小变化（比如不同的形式编码或改变比特率）不影响指纹。

为多媒体对象生成指纹的一个例子在国际专利申请 WO 02/064782（代理人文档号 PHNL010110），以及在 Jaap Haitzma, Ton Kalker 和 Job Oostveen, “用于内容识别的强壮音频散列法”，基于内容的多媒体索引国际工厂，Brescia, 2001年9月。

30 在大规模系统中，指纹数据库需要被分布到一定数量的指纹服务器，使得能够处理所有的搜索请求并存储所有的指纹。此外，数据库

需要与时间同步。例如，在音频指纹的情况下，新发行的歌曲的指纹被加入。所需的服务器和数据库时间同步使系统十分昂贵。

本发明的一个目标是根据前言所提供的一种比所知方法便宜的方法。

- 5 根据本发明的方法达到的这个目标，该方法包括把数据库各个部分分布到与用于共享所述多个多媒体对象的文件共享网络相连接的各个文件共享客户。

10 使用这个方法，就不再需要积极地出门去购买比如 CD 上的内容，或自己去寻找内容中的元数据了。通过利用网络中文件共享客户可得的对象和元数据，指纹和元数据能以很便宜和有效的方式收集。这些客户已经使对象可被任何人下载，所以不需要再购买这些对象。另外，这些对象可以典型地和元数据合在一起，所以元数据也可以被使用。

15 把数据库分布到文件共享网络有不需要专用数据库服务器和管理系统的附加优点。该文件共享网络已经包含有潜在的大量相互连接的计算机，这些计算机为维护这样的数据库提供了很合适的基础。

20 并且，根据本发明的方法要比业界以前的方法更能扩展。当更多的用户加入到文件共享网络中，对元数据的请求数量会增加，那么如果期望有满意的响应时间，对数据库服务器的请求也应当增加。但是，当数据库被分布到文件共享网络中的客户时，那么当新用户加入网络时网络中更多的计算机会变为可用。这些新计算机提供的多余计算能力、存储和连接就可以被用作维护一部分数据库。这种方式下，一道扩展了分布式数据库的能力和要求。

25 本发明更进一步的目的是提供一个文件共享客户，包括存储一个或多个多媒体对象的存储器，在存储器中与文件共享网络上的其它文件共享客户共享多媒体对象的共享装置，用于计算指纹和为被共享设备共享的多媒体对象获取一组元数据，以及用于添加计算出的指纹和所得到的元数据集到被分布到和文件共享网络相连的文件共享客户的数据库的指纹装置。

30 这样的文件共享客户能够加入到如上概括的方法中。因为指纹和数据收集装置被集成在文件共享客户中，安装文件共享客户的用户也能自动的安装所需装置，用以帮助维护分布式数据库。这样，当他们随后加入到文件共享网络时，他们的计算能力、连接和存储变得为网

络所用，并且扩展了分布式数据库的功能。

5 在一个实施例中，文件共享客户还包括用于维护部分分布式数据库的 DBMS 装置。通过在文件共享客户中安装该数据库管理系统装置，每个安装该客户（通常在一个计算机系统中）也安装了 DBMS 装置，并且能有助于维护分布式数据库。

在另一个实施例中，被 DBMS 装置所维护的分布式数据库部分大小依靠运行客户的计算机系统的性能。例如，可被考虑的带宽限制、CPU 速度和/或可用工作内存（RAM）。在这种方式下，数度慢的计算机不会负载一个大型指纹数据库服务器。

10 在另一个实施例中，该 DBMS 装置用于添加所计算的指纹和所得到的元数据集到各个部分。在这种方式下，该分布式数据库用新指纹和来自在文件共享客户上显示的多媒体对象的元数据集来更新。每个客户现在维护包含有至少一个出现在它自己存储器中的对象的分布式数据库部分。

15 在另一个实施例中，该指纹装置用于传输所计算的指纹和所得到的元数据集到文件共享网络上的另一个文件共享客户。在这种方式下，要存储到数据库上的数据能借助文件共享网络被分布，这样它就能被存储到由任意客户所管理的部分，该客户被安排为管理该部分。

20 在上述实例的一个变化中，其它文件共享客户是文件共享网络上的超级节点。超级节点是拥有足够带宽、处理能力和内存的客户。普通客户通过连接到一个超级节点与网络连接并将被共享文件列表发送到该超级节点。超级节点和许多客户连接并且还和许多其它超级节点连接。因为它们在内存、处理能力和带宽方面的强大性能，它们更适合管理一部分分布式数据库。

25 在另一个实施例中，该指纹装置被安排用于计算指纹和当多媒体对象存储在存储器中时，获取多媒体对象的元数据集。通过在此时计算该指纹，任何所获得的新多媒体对象的元数据被自动添加到分布式数据库被完成。

30 本发明的另一个目标是提供一个计算机程序产品，该产品用于使通用计算机能如根据本发明的文件共享客户一样运行。

本发明的这些以及其它方面通过参考附图中所示实施例会清楚明白，其中：

图 1 示意地显示包括多个客户的文件共享网络；并且

图 2 示意地显示更详细的文件共享客户。

在整个图中，相同的引用数字表示类似或相应的特征。在附图中表明的一些特征典型地在软件和这样的代表软件实体，比如软件模块和对象中实现。

图 1 示意地的显示包括多个客户 101、102、103、104 和 105 的文件共享网络 100。虽然这里所示的如同一个物理网络，客户 101—105 之间有直接连接，但文件共享网络 100 最好被当成概念的或虚拟的网络。这就是说，不需要所有的客户 101—105 在所有时间都物理地或网络状地相互直接连接。所需要的是“网络中”的一个客户能从另一个客户获得文件或对象。并且，即使使用客户到客户直接连接时，也不需要所有的客户和所有其它的客户连接。

网络 100 可包括服务器 110，其为客户 101—105 执行目录服务。为了和文件共享网络 100 连接，客户 101 提交它要和服务器 110 共享的文件（或对象）列表。服务器 110 结合它从与网络 100 连接的所有客户接收的列表。其它客户 102—105 就能连接到服务器 110 并浏览结合的列表或搜索在该列表上的特定对象。随后它们能和含有它们所寻找对象的客户联系，并直接从那个客户获得（下载）。在这种方式下，服务器 110 不直接加入客户 101—105 间的文件或对象共享。这种方法在世界范围的 Napster 文件共享网络中很有名。

也可以不用服务器 110 而实现网络 100。在这种情况下，客户 101 通过已经在网络 100 中的客户 102—105 的一个或多个和网络 100 连接。客户通过向与它相连的客户发送一个搜索请求来搜索网络。这些客户检查它们所共享的对象列表，如果所请求对象在列表中则返回结果。并且，该请求被转发给和这些客户连接的其它客户。在这种方式下，该请求被分布到整个网络 100 中，直到它被一个能实现该请求的客户接收，或直到所有客户都已经接收它并且没有一个能实现它。

该实施例可从例如 Gnutella 文件共享网络知道。该实施例的一个缺点是网络 100 不可扩展。例如当前的类 Gnutella 网络不支持一百万客户。此外，如果有许多“慢”计算机，即是对网络 100 的带宽、处理能力和/或内存有限的计算机，网络速度变慢。

或者，客户 101 在连接了一个或多个其它客户 102—105 后能提交

它想与其它客户 102—105 共享的文件或对象列表。该列表就传递到网络 100 上所有的客户。在这种方式下，所有客户知道哪个客户有哪些文件和对象可用并能和那个客户直接联系。

5 所知的 KaZaa 文件共享网络也能不需要服务器 110 而操作，但为克服了上面提到的问题使用两种客户：超级节点和“普通”客户。超级节点是拥有足够带宽、处理能力和内存的客户。普通客户通过连接超级节点连接网络，并发送被共享的文件列表到该超级节点。超级节点有到许多客户的连接并且还和许多其它超级节点连接。

10 一个超级节点同时也是一个普通客户。也就是说，对于用户来说他的计算机是超级节点的事实是透明的。当用户想搜索一个文件时，它的客户发送一个请求到当前和他的客户连接的超级节点。该超级节点返回它的客户所发送的文件列表中的匹配文件。此外，如果需要，该超级节点转发该请求到所有与它以类似上述 Gnutella 实施例中的方式连接的超级节点。但是，因为超级节点之间的连接有巨大的带宽，  
15 所以这种方法比 Gnutella 网络更快。而且，它能扩展到几百万个客户。

被典型地称为点对点或 P2P 文件共享网络的该文件共享网络十分普及。这些网络的著名例子：Napster, Musiccity, Gnutella, Kazaa, Imesh 和 Bearshare。一旦用户已经安装了适当的客户软件到他们的个人电脑上，他们就能共享他们的文件并且他们能够下载被其它用户共  
20 享的文件。客户 101—105 可被连接到如因特网的网络，这能方便文件共享网络 100 的建立。客户可例如使用直接 TCP/IP 连接到另一个客户来获得文件或对象。

在最流行的网络上，通常超过 500,000 的人同时连接。在写的时候，大多数人们共享音乐文件（经常以 MP3 格式），但共享电影也不  
25 断流行。术语“多媒体对象”被用在包括有音乐、歌曲、电影、TV 节目、图片和其它类型二进制文件的指示文件，并且文本文件也可以用这种方式共享。注意到，多媒体对象可以由几种不同文件组成。

网络 100 还能包括分布式数据库。分布式数据库由多个各个部分  
201—205 组成，每个部分分别由客户 101—105 维护。这将在下面通  
30 过参考图 2 解释。

图 2 更详细地显示了文件共享客户 101。文件共享客户 101 最好能被实现为其上运行文件共享软件 301 的个人电脑，这是业界周知的。

文件共享软件 301 典型地利用网络模块 302，比如现代操作系统中可用的 TCP/IP 栈。存储器 303 包括一个或多个被文件共享软件 301 共享的多媒体对象。典型地，该存储器 303 可以是一个硬盘上的目录。在  
5 一些情况下，存储器 303 可包括存储了已下载多媒体对象的不同部分。该部分，典型地也是目录，不需要和存储了要被共享的多媒体对象的目录一样。

文件共享客户 101 还包括指纹模块 304，其能从多媒体对象计算指纹。如上所述，计算指纹的方法在国际专利申请 WO 02/065782（律师诉讼事件表 PHNL010110）中描述，虽然当然可以使用任何计算指纹  
10 的方法。指纹模块 304 还获得多媒体对象的一个元数据集。

通常该元数据集被包括到多媒体对象或和多媒体对象一道被包括，这样在获得多媒体对象时自动获得该元数据集。

指纹模块 304 最好能被实现为一个或多个硬件或者软件模块，例如作为在客户 101 上运行的文件共享软件 301 的插件程序。

15 指纹模块 304 能计算在存储器 303 中的多媒体对象的指纹。多媒体对象的元数据集类似地能通过从存储器 303 中的多媒体对象简单地读取来获得。例如，含有以流行的 MP3 为格式的音乐的多媒体对象通常在对象末包含如 ID3 “tag”的元数据。

20 因为计算多媒体对象指纹可能是 CPU 密集的，现在需要小心避免消耗太多的 CPU 能力。这样做可能会使文件共享软件的用户烦恼，因为他明白这妨碍了正常的系统使用。

指纹可根据用户请求计算或者在后台中计算。在后一种情况下，推荐周期地扫描共享驱动或指纹还未被计算的新多媒体对象的目录。如果发现这样的对象，指纹被自动计算。如果对于该对象没有可用元  
25 数据，用户可被提示输入一个元数据集。

在任何情况下，一旦指纹模块 304 已经计算了多媒体对象的指纹并且已经获得该多媒体对象的元数据集，它就包含了在分布式数据库 201—205 中的指纹和元数据集。推荐地，该指纹和元数据集被包括到由 DBMS 模块 305 维护的部分 201。

30 指纹数据库维护（DBMS）模块 305 维护分布式指纹数据库的部分 201。该数据库 201 含有指纹和相关元数据集。如果存储器 303 明显没有包含某个多媒体对象的多个拷贝，该数据库 201 典型地能为每个共



享多媒体对象包含指纹和相关元数据集。

另外，数据库 201 可随文件共享客户 101 从在网络 100 上的其它文件共享客户 102—105 下载的多媒体对象的指纹和元数据而被扩展。在多媒体对象被下载的同时该对象的指纹可以被计算。一些计算指纹的方法每次操作多媒体对象的一小部分。例如，上述欧洲专利申请计算多媒体对象中音频数据每三秒钟的“亚指纹”，并从所有的亚指纹中构建出实际的指纹。三秒钟量的数据被接收后可以立即计算亚指纹。

如果那个对象的元数据也可得，在对象被完全下载之前指纹和元数据能被包含到数据库 201 中。如果在这个处理期间确定出指纹已经在数据库 201 中，那么用户很可能已经在他的所有物中已经拥有这个特定多媒体对象的拷贝。那么用户可以被警告，这样他能退出下载。

当文件共享客户 101 从另一个客户 102 下载多媒体对象时，该客户 101 还能从客户 102 下载一个或多个指纹以及相关的元数据集。这些指纹相对较小（相对典型的多媒体对象几兆比特，一般情况为十千比特）因此不会影响客户 101 的性能。这提供了将数据库有关的指纹和元数据以任意的形式分布到网络 100 的客户 101—105 上。

在 KaZaa 文件共享系统中，超级节点推荐被用作分布指纹和元数据到网络 100。在比如象 Napster 的文件共享系统中，可能是中心服务器分布指纹。

还可在超级节点或中心服务器的协助下获得正确的元数据。客户提交一个对特定指纹的搜索请求到它所连接的超级节点。该超级节点将该请求传递到其它的超级节点。没有一个过滤数据库中元数据集来确定确切集的中心服务器，超级节点就可能收到多个对该询问的回答。于是，超级节点应用多数投票或另一种技术来确定确切的元数据集，该元数据集被提供回提交该请求的客户。

例如，假设对特定指纹的搜索请求的响应所接收的元数据集如下：

1. (artist= “Jewel” , title= “Hands” )
2. (artist= “Jewel” , title= “Hands” )
3. (artist= “Jewel” , title= “Hnds” )
4. (artist= “Jewel” , title= “Hands” )
5. (artist= “Jewel” , title= “Hands” )

可以容易看出在本例中五个集中的四个给出的艺人名字是

“Jewel”，同时只有一个给出的名字是“Jewel”。使用简单的多数  
取胜方法，确切的元数据集所给出的艺人名字应当是“Jewel”。类似  
地，五个集中的四个给出的歌曲名称是“Hands”，因此确切的元数据  
集所给出的歌曲名字也应当是“Hands”。同样的方法当然可以被用在  
5 其它类型的包括在集合中元数据，比如专集名称、发行年份、风格、  
艺人网站的 URL 等等。

当然也可以利用其它自动从多个候选值中确定确切值的更先进技术。  
这些技术在智能代理领域是普通的，其中它们被用作消除从代理  
上接收信息的噪音。它们包括决策树剪枝和交叉验证。正好需要什么  
10 来组成“足够数量”依赖于所使用技术。

可观察到，不是所有的元数据集要求完整。例如，一个元数据集  
可能只包含某个特定歌曲的标题和艺人的名字，同时另一个也可包含  
获得该歌曲的专集标题以及该专集发行的年份。那么，上面的过程应  
当在个别类型的元数据上执行，例如一次是为基于所有可得标题的标  
15 题，一次是为基于所有可得艺人姓名的艺人名字，一次是为发行年份  
等等。在这种方式下，确切的元数据集能被尽可能广泛地得到，即该  
元数据集不仅包括标题和艺人，而且包括专集名称和发行年份。该扩  
展的确切元数据集是最有价值的。

随后超级节点能用确切集更新它自己的数据库，以避免必须在每  
20 次他的客户的其中一个再次提交询问时再次传递询问到所有其它超级  
节点。但是，这要冒着他的信息可能在一些时候过期的危险。

可以使由 DBMS 模块 305 维护的分布式数据库 201 部分的大小依靠  
于运行它的个人电脑的性能。例如，可以考虑的带宽限制、CPU 数度  
和/或可用的工作内存（RAM）。在这种方式下，数度慢的计算机不会  
25 负载一个大型指纹数据库服务器。

文件共享客户 101—105 能使数据库 201—205 的至少一部分为其  
它部分所用。这可以通过，例如提供一个搜索接口来完成，通过该接  
口客户提交指纹并接收返回的元数据集。从数据库提取和所提交的指  
纹有关的元数据集的各种方法可以从如上所述的在国际专利申请 WO  
30 02/065782（律师诉讼事件表 PHNL010110），以及从国际专利申请 WO  
02/058246（律师诉讼事件表 PHNL010532）中得知。当然也可以使用  
其它的方法。

如果特定客户 101 不能在它的分布式数据库部分 201 找到和所提交指纹有关的元数据集，它可以转发该已提交指纹到连接文件共享网络 100 的另一个客户 102。如果该网络 100 包括超级节点，该另外的客户 102 最好是文件共享网络 100 中的一个超级节点。该客户 102 如果不能在它的部分 202 中找到这个集，类似地也能转发已提交指纹，并且一直下去直到客户 101—105 的其中一个在它的部分 201—205 找到这个元数据集，或直到在文件共享网络 100 中的所有客户 101—105 都没有能找到该集。

分布式数据库 201—205 的内容能成为免费或只支付订费。或者，可向每个在数据库执行的询问收取费用。响应提交指纹所返回给客户的元数据量也可以变化：例如，免费服务只返回艺人和名称，而基于订费的服务返回数据库中所有可得元数据。

需要注意上述实施例是列举说明而不是限制本发明，业界技术人员能设计许多可选实施例而不需要偏离附加的权利要求的范围。

在权利要求中，任何在括号之间的参考标志不应被解释为对权利要求的限制。词“包括”不排除那些列在权利要求之外的其它元素或步骤的存在。在一个元素前的冠词“一”不排除多个该元素的存在。

本发明可以通过包括有多个不同元件的硬件方式和通过被适当编程的计算机方式实现。在列举了几个装置的设备权利要求中，这些设备的几个可以由硬件的一个和相同项来体现。在不同的互相依赖的权利要求中书面陈述某个措施的事实不表明这些措施的组合能被用作取得优势。

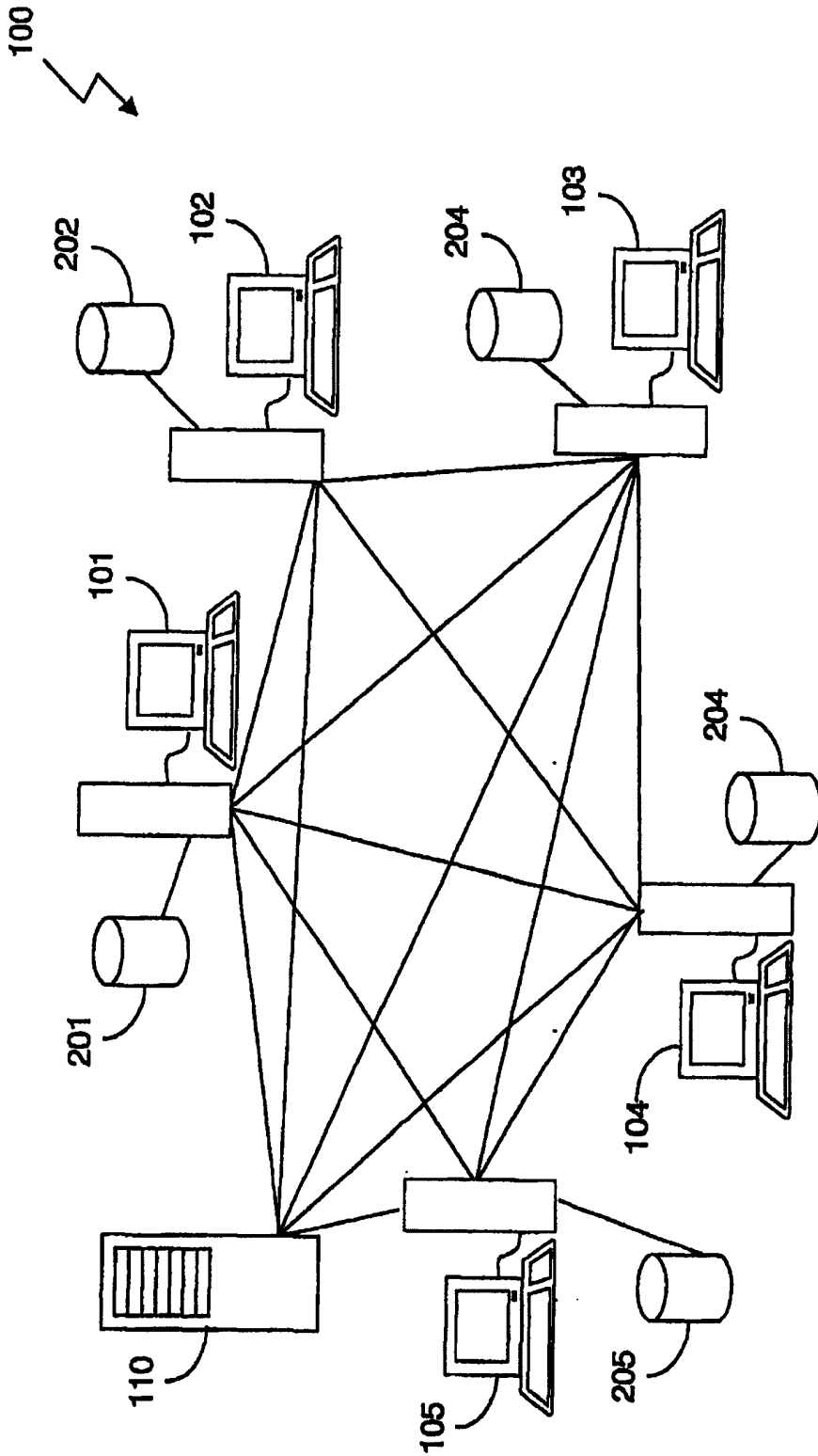


图 1

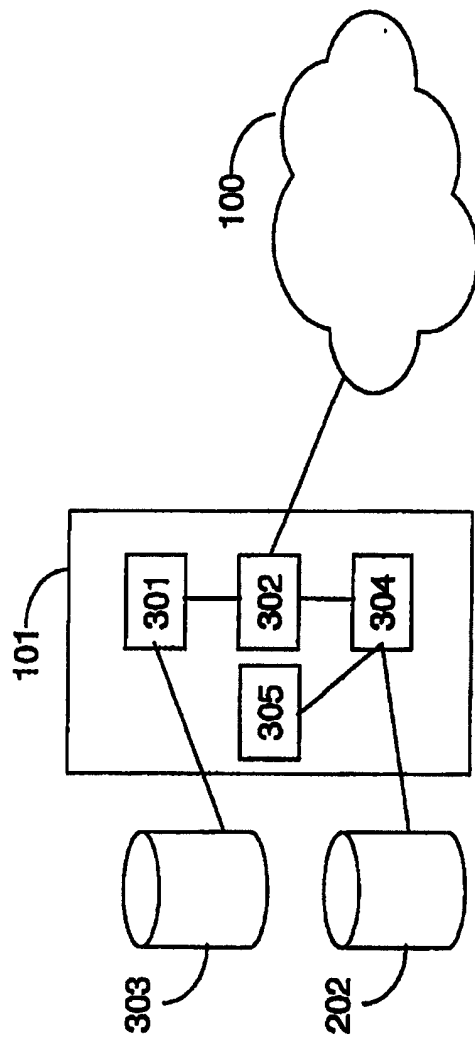


图 2