

[19] 中华人民共和国国家知识产权局



## [12] 发明专利申请公布说明书

[21] 申请号 200580052407.8

[43] 公开日 2009 年 1 月 14 日

[51] Int. Cl.  
G06F 17/30 (2006.01)  
H04L 29/08 (2006.01)

[11] 公开号 CN 101346718A

[22] 申请日 2005.10.28

[21] 申请号 200580052407.8

[86] 国际申请 PCT/EP2005/011580 2005.10.28

[87] 国际公布 WO2007/048432 英 2007.5.3

[85] 进入国家阶段日期 2008.6.24

[71] 申请人 意大利电信股份公司

地址 意大利米兰

[72] 发明人 L·伯里阿诺 G·洛贝洛

[74] 专利代理机构 中国国际贸易促进委员会专利  
商标事务所

代理人 李 玲

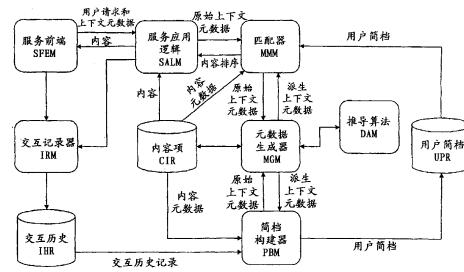
权利要求书 3 页 说明书 34 页 附图 4 页

### [54] 发明名称

用于向用户提供选定内容项的方法

### [57] 摘要

本发明公开了一种用于向用户提供选定内容项的方法。内容项的选择是以预先指定给内容项的元数据以及以后产生和关联的元数据为基础的，其中预先指定给内容项的元数据通常是原创内容元数据，而以后产生和关联的元数据则被称为派生内容元数据；此外，内容项的选择还可以基于上下文元数据，尤其是派生上下文元数据。派生元数据是根据与所要应用的算法相对应的推导规则而自动产生的，其中举例来说，该算法被应用于内容项的内容、原创内容元数据以及上下文元数据。用户简档可以用于改善选择质量；此外，本发明还公开了一种根据机器学习技术来构建和维护用户简档的方法。



1. 一种用于向用户提供选定内容项的方法，包括以下步骤：

A) 根据用户请求来识别第一内容项集合，其中第一内容元数据被预先指定给所述第一集合的内容项，

B) 至少根据第一推导规则来为所述第一集合的内容项自动产生第二内容元数据，所述推导规则与至少应用于所述第一集合的内容项的算法相对应，

C) 将所述第二内容元数据与所述第一集合的内容项相关联，以及

D) 根据所述第一内容元数据和所述第二内容元数据来提供源自所述第一集合的第二选定内容项集合。

2. 根据权利要求 1 所述的方法，其中所述算法还被应用于所述第一内容元数据的至少一些。

3. 根据前述任一权利要求所述的方法，其中步骤 B) 还根据与所述选定内容项的交互上下文相关的第一上下文元数据来执行，以便自动产生第二上下文元数据。

4. 根据前述任一权利要求所述的方法，其中所述第二内容元数据是从应用于多个内容项的算法中推导的。

5. 根据前述任一权利要求所述的方法，其中步骤 D) 还根据所述用户的用户简档来执行。

6. 根据权利要求 5 所述的方法，其中所述用户简档包括预测模型。

7. 根据权利要求 5 或 6 所述的方法，其中在步骤 D) 中，针对所述第一集合的每一个内容项的排序是根据所述第二内容元数据以及所述预测模型提供的，由此根据所述排序定义所述第二内容项集合。

8. 根据权利要求 7 所述的方法，其中针对所述第一集合的每一个内容项的排序是根据所述第二内容元数据、所述第二上下文元数据、以及所述用户简档来提供的，并且所述方法还包括根据所述排序来选

择所述第二内容项集合。

9. 根据前述任一权利要求所述的方法，其中所述第二内容项集合是作为针对所述用户的相应请求的答复提供的。

10. 根据权利要求 6 所述的方法，还包括以下步骤：将来自所述用户的反馈与所述第二集合的至少一个内容项相关联。

11. 根据权利要求 10 所述的方法，还包括以下步骤：记录所述第二集合的至少一个内容项以及来自所述用户的并与所述至少一个被记录内容项相关联的反馈。

12. 根据权利要求 11 所述的方法，还包括以下步骤：记录用于选择所述至少一个内容项的所述第二元数据的至少一部分。

13. 根据权利要求 11 或 12 所述的方法，还包括以下步骤：记录所述用户请求。

14. 根据权利要求 11~13 中任一权利要求所述的方法，还包括以下步骤：至少根据所述被记录内容项以及用户反馈来构建或更新所述用户的预测模型。

15. 根据权利要求 14 所述的方法，还包括以下步骤：记录所述第二元数据的至少一部分，以及至少根据所记录的第二元数据来构建或更新所述用户的预测模型。

16. 根据权利要求 15 所述的方法，其中所述预测模型是通过应用于至少所述第二内容元数据的至少一种机器学习算法来构建或更新的。

17. 根据权利要求 16 所述的方法，其中所述机器学习算法还应用于至少一些上下文元数据。

18. 根据权利要求 6 所述的方法，其中用户简档包括至少两个预测模型。

19. 根据前述任一权利要求所述的方法，其中所述第二内容项集合是由服务供应商提供的。

20. 根据前述任一权利要求所述的方法，其中所述选定内容项是通过电信网络提供的。

21. 一种计算机程序产品，该产品可以加载到至少一个计算机的存储器中，并且包括用于执行权利要求 1~20 中任一权利要求所述的方法的软件代码部分。

22. 一种基于内容的服务，该服务包括以下步骤：收集来自用户的内容请求，以及根据权利要求 1~20 中任一权利要求所述的方法来向用户提供选定的内容项。

## 用于向用户提供选定内容项的方法

### 技术领域

本发明涉及一种用于向用户提供选定内容项的方法。本发明旨在用于那些通过诸如数据网络（例如因特网）或电话网络（例如 UMTS 网络）之类的电信手段而被提供的基于内容的先进服务。

本发明与内容过滤、信息检索、服务个性化以及用户归档（user profiling）领域相联系。

特别地，在被应用于丰富的多媒体内容项、也就是包含不同媒体内容组件（文本、图像、音频、视频、……）的内容项时，本发明将是非常有益的。

### 背景技术

每一天，在世界上都会有大量信息发布，这些信息可以通过印刷品、电视、因特网之类的不同信息媒体而被人们得到。此外，信息量也在快速增长。

不幸的是，对个人来说，当前可用信息源提供的庞大信息量通常是不可抗拒的，而个人则有可能无法或者没有兴趣在这些信息中拣选其感兴趣的项目。因此，目前需要一种服务或能力来为用户仅仅提供其感兴趣的信息。

长期以来，较为普遍的是根据“关键词”来过滤内容项。关键词由用户提供给软件应用；该软件应用既可以处于用户计算机本地，也可以运行在与用户计算机相连、例如通过因特网相连的远端计算机上。该软件应用会向用户返回与用户指定的关键词相关的所有可用内容项。

关键词是一种普通类型的元数据。在过去，“元数据”被定义成是“与信息相关的信息”；举个例子，对“文章”的“标题”和“摘要”来说，

---

由于它们提供了关于文章内容的信息，因此它们是元数据，而文章的内容则是信息本身。

为了过滤内容信息，目前还开发了更复杂的方法。这些方法是以使用元数据标引不同内容项为基础的，特别地，所述元数据可以是“原创元数据（authored metadata）”，也就是通过内容项作者或是他人而与内容项关联的元数据。

一般来说，有效过滤内容项需要关于用户的知识，例如，该用户知识可以是用户习惯和/或用户偏好。

在 Erika Savia 等人发表于 Proc. 8th Finnish Artificial Intelligence Conference, Human and Artificial Information Processing, 第 61-69 页, 1998 的论文“Metadata Based Matching of Documents and User Profiles”中可以发现一份关于通过使用元数据和用户简档来过滤文档的有趣评述。

从国际专利申请 WO02/41579 中可以知道一种用于聚集和传送多媒体数据的方法。多媒体数据依照其内容而被分析，元数据提取模块提取相应元数据，以及预备用户简档。在从中心单元接收多媒体数据之前，用户借助通信设备来设置用户简档中的至少一部分用户数据和/或修改该数据。多媒体数据借助元数据并且根据用户简档而被选择，此外还借助再封装模块而从选定的多媒体数据中产生以用户特定方式优化的、面向内容的多媒体数据。所述以用户特定方式优化且面向内容的多媒体数据将会保存在中心单元的内容模块的数据库中，并被提供给用户。

根据这份国际专利申请（第 8 页第 6~14 行），元数据是根据基于内容标引技术来检索的，例如，该标引技术可以是美国专利 5,210,868 和 5,414,644 描述的技术之一。

## 发明内容

本发明涉及一种用于向用户提供内容项的方法，尤其涉及的是在考虑用户偏好的情况下提供选定内容项的方法。

本发明的基本思想是自动产生元数据，以及使用这些所产生的元数据来选择要提供的内容项。

申请人认识到，手动产生元数据、即手动产生一般原创元数据的处理是一个极其耗时的活动（我们是无法跟上日益增长的已发布信息的量），此外，该处理还很容易出错，并且通常不是面向服务的。因此，在实践中是很难达到适于过滤数量庞大的已发布信息的精度的。

申请人认识到，具有精确的用户简档以及具有构建和自动更新这些用户简档的能力是非常重要的。

此外，申请人还注意到，在提供内容时可以很有利地使用交互上下文来构建或更新用户简档。

根据本发明的方法可以由向用户提供个性化内容递送服务的服务供应商来提供。此外，上述考虑因素在以 PULL( 拖拉 ) 模式和 PUSH ( 推送 ) 模式提供内容项的时候都是适用的。

根据本发明，内容是自动生成的，优选地，上下文元数据同样是自动生成的。派生元数据 ( derived metadata ) 是根据推导规则 ( derivation rules ) 自动生成的，并且该推导规则与所应用的算法是对应的，其中举例来说，所述算法是应用于内容项的内容、原创内容元数据以及原始上下文元数据的算法。

上述特征为根据本发明的选择方法提供了灵活性和动力。

除了显性和/或隐性用户反馈之外，在这里还可以很有利地使用派生元数据来构建和维护用户简档。以此方式可以精确构建用户简档，并且该用户简档可以随时间精确维护。优选地，构建和维护（也就是更新）用户简档的处理是通过使用机器学习技术来执行的。

本发明包括与内容项处理相关的第一个方面。内容处理与内容项的恰当选择基本对应。在优选实施例中，本发明还涉及与用户简档处理相关的第二个方面，其中该方面与恰当构建和维护用户简档的处理基本对应。

由于精确的内容选择可以依照用户简档匹配来执行，换言之，内容项是对照用户简档来匹配的，因此，这两个方面是相互联系的。为

了构建和维护用户简档，较为有利的是使用与选定内容有关的用户反馈（显性和/或隐性）。

根据本发明的第一个方面，在接收到来自用户的内容请求时执行下列步骤：

- 从上述请求开始产生一个查询；
- 根据该查询来识别第一集合的内容项；
- 识别那些有可能与该第一集合中的每个内容项关联的预先指定的内容元数据（通常是原创的）；
- 优选地，识别那些代表与上述请求相关联的上下文信息的原始上下文元数据，以及
  - 根据推导规则来为每一个内容项自动生成派生元数据，其中该推导规则与应用于内容项以及优选还应用于与所述内容项相关联的预先指定的内容元数据（如果存在的话）的算法相对应。更为优选的是，这些算法还被应用于原始上下文元数据。

优选地，在生成了派生元数据之后，第一内容项集合被保存在内容项储存库中。更为优选的是，依照用户请求产生并且与内容项相关联的派生元数据同样会保存在内容项储存库中，以便在以后使用，从而避免重复执行元数据推导处理。

当已经产生了与所述第一集合的每一个内容项相关联的派生元数据时，这时会将第二选定内容项集合提供给用户，其中所述第二集合包含在第一内容项集合内部。根据本发明的优选实施例，第二内容项集合是通过执行下列步骤提供的：

- 为每一个内容项识别（派生和预先指定的）内容元数据；
- 优选地，为每一个内容项识别上下文（派生的和原始的）元数据；
  - 识别产生该请求的用户所具有的用户简档，
  - 根据至少某些派生元数据来将所述第一集合的内容项与用户简档相匹配，以及为该集合的内容项产生一个排序；
  - 根据所述排序，将第二内容项集合提供给用户，其中所述第二

集合与根据所述排序排列的第一集合相对应，或者（优选地）与包含最佳排序内容项的内容项子集相对应，以及

- 优选地，收集关于所提供的内容项的用户反馈（显性和/或隐性的）。

对属于与用户反馈相关联的所述第二集合的内容项来说，每一个内容项都对应于一个交互事件，其中该交互事件优选作为一个记录而被保存在交互历史储存库中。由此，根据本发明的优选实施例，为了更新用户简档，执行下列步骤：

- 检索与产生请求的用户相关的多个交互历史记录，其中每一个记录都包括一个内容项以及与所述内容项相关联的至少一个用户反馈（显性和/或隐性），并且该用户反馈通常是以用户投票表示的。优选地，这些记录还包括用户请求、原始上下文元数据以及与内容项相关联的内容元数据，其中该内容元数据是推导得到并且是预先指定的；

- 选择用于构建该用户的预测模型的机器学习算法；

- 将每一个记录（也就是所存储的每一个交互事件）编码成特征矢量，其中该特征矢量是被适配成与选定的机器学习算法结合使用的形式表示。所述特征矢量包含与关联于特定内容项的元数据相对应的多个元素，其中该元数据是推导得到以及（如果存在的话）是预先指定的，此外，该特征矢量还包含用户反馈。如果在记录中并未给出推导得到的元数据，那么可以从内容项储存库中检索这些元数据，其中该内容储存库包含第一集合的所有内容项，并且通常包含依照查询所选择的内容项；

- 将选定的机器学习算法应用于所述特征矢量，其中每一个矢量与一个交互事件相对应，由此构建预测模型（用户模型）；

- （更为优选的是）验证所构建的预测模型，以及

- 通过使用新预测模型取代旧预测模型来更新上述用户的简档。

由此，根据本发明的优选方面，在考虑用户反馈的情况下，通过将机器学习算法应用于派生元数据，并且优选应用于预先指定的元数据，可以在第一内容项集合内部定义一个排序，由此提供第二内容项

集合。更为优选的是，上下文元数据（原始的和派生的）是作为应用了机器学习方法的特征矢量中的独立特征而被考虑的。如果在交互历史储存库的记录中不存在派生上下文元数据，那么，由于原始上下文元数据的元数据推导的计算通常并不繁重，因此，他们可以从原始上下文元数据中同时（on-the-fly）被推导得到。

### 附图说明

通过结合附图来考虑下文中的描述，可以更清楚地了解本发明，其中：

图 1 显示的是实施根据本发明的方法实施例的系统的框图，

图 2 显示的是根据本发明来处理内容项的主要步骤的流程图，

图 3 显示的是根据本发明来处理用户简档的主要步骤的流程图，以及

图 4 示意性显示了一种被适配成存储内容项和相关元数据并且可以用于本发明的数据结构。

### 具体实施方式

在详细描述本发明之前，在下文中将会提供某些术语定义和描述。

#### 基于内容的服务

在本说明书中，基于内容的服务是任何一种通过利用已有内容项集合来构建有可能对服务的订户有价值的信息内容的软件应用。聚集以及向用户呈现选定内容的方式是由服务应用逻辑来定义的。

用户可以采用 PULL 或 PUSH 这两种模式以及通过服务前端来与基于内容的服务进行交互；基于内容的服务可以提供一种或两种模式。在 PULL 模式中，用户通过直接访问服务前端来发起交互，其中该用户有可能提供特定输入，以便即时获取预期内容。在 PUSH 模式中，当用户预订（并且有可能在后续时间）基于内容的服务时，用户可以提供输入，并且该输入可以在以后产生服务内容。根据这些输入，

当产生内容时，用户将被告知访问服务前端，以便获取该内容。

### 内容项 (CONTENT ITEM)

在本技术领域中，内容项是用于基于内容的服务的用户内容交互的基本单元。基于内容的服务提供了通常充当服务订户（即用户）请求应答的内容项。内容项是作为服务递送的单个实体而被用户察觉的事物。但是，内容项也可以包括一个或多个内容成分。举例来说，如果内容项是足球比赛视频，那么该项目可以包括作为内容成分的两个比赛半场。

关于内容项的示例是：

- 电影或电视节目，例如来自按需点播媒体递送环境的电影或电视节目；
- 新闻文章，例如来自在线新闻阅读器的新闻文章；
- 单独的万维网 URL 结果，例如来自万维网搜索引擎的万维网 URL 结果；
- 曲目，例如来自内容共享网络环境的曲目；
- 图片，例如来自在线媒体目录的图片；
- 网页，例如来自因特网导航的网页；
- 产品页面，例如来自电子商务目录的产品页面。

通常，内容项是包含一个或多个内容成分的结构化对象。

每一个内容成分都是多媒体元素，例如文本、图像、音频、视频、三维模型、矢量图形、图形布局。

关于文本成分的示例是：在线报纸文章的文本、新闻文章的文本部分、包含在网页中的文本、电子商务目录中的产品文本描述。关于图像成分的示例是：包含在网页中的图片和绘图、包含在新闻页面中的照片、包含在在线媒体目录中的图片。关于音频成分的示例是：包含按需点播媒体递送环境中的曲目的音频文件、包含内容共享环境中的曲目的音频文件、电影中的音轨、新闻文章中的音轨。关于视频成分的示例是：包含按需点播媒体递送环境的电影或电视节目的文件、包含内容共享环境中的视频的文件、新闻页面的视频部分。关于三维

模型成分的示例是：代表在线电子商务目录中的器具的 3D 模型。关于矢量图形成分的示例是：网页中的 Flash 动画，SVG（可缩放矢量图形）文档。关于图形布局的示例是网页的图形布局。

举例来说，关于多成分内容项的示例是：与某条新闻相关并且包含文本成分（也就是描述该新闻的简短文本）、音频-视频成分（也就是描述和显示该新闻的音频-视频序列）以及音频成分（也就是描述该新闻的音频序列）的新闻项目。

### 内容元数据(CONTENT METADATA)

在过去，“元数据”被简单地定义为“关于信息的信息”。

更具体地说（并且根据 W3 【“www”】联盟），内容元数据通常包含描述指定内容项的数据结构，并且可以由计算机之类的机器自动处理。

内容元数据可以描述属于内容项的每一个内容成分，或者将该内容项作为一个整体来进行描述。

可供内容供应商使用的内容项通常具有元数据，例如标识标引（例如主题、字段、……）。当内容项可供基于内容的服务使用时，与该内容项相关联的元数据将被称为预先指定的元数据。预先指定的元数据通常是“原创元数据”，它是一种由他人或内容项作者而关联于内容项的内容元数据，所述他人或内容项作者通常位于内容供应商组织内部。原创元数据通常是借助注释处理而被手动指定给每一个内容项的。

元数据具有不同的种类，例如：文本元数据、关键词元数据、分类元数据（具有处于有限值集合以内的值的分类标签）、数字元数据。元数据可以具有更复杂的结构，其中举例来说，该结构可以是从先前类型（结构化元数据）组合中推导得到的结构，或是与根据 RDF（资源描述框架）的语义网络（语义网络元数据）相对应的结构。

关于文本元数据的示例是：与网页中的图片相关联的文本描述（在这里可以提供与页面中的每一个图片的内容相关联的多个原创元数据）、网页内容的文本概述（与网页整体相关联的原创元数据）、

与曲目项目相关联的曲目歌词。关于关键词元数据的示例是：用于描述新闻项目所覆盖的主题的关键词列表、与电影项目特征（如 IMDB【互联网电影数据库】中的电影项目特征）相关联的关键词列表、用于描述场景主要特征以及图片项目中描述的主题的关键词列表。关于分类元数据的示例是：声明新闻项目的新闻分类（在预定新闻分类集合内部）的“分类”标签、声明曲目音乐流派（在预定的音乐流派集合内部）的“流派”标签、声明电影项目是“黑白”还是“彩色”的“彩色”标签。关于数字元数据的示例是：与电影项目的出品年份相对应的整数、与电影项目的持续时间（例如以分钟为单位）相对应的整数、与在电子商务目录中购买的产品项目的价格相对应的货币数量。

举例来说，结构化元数据可以应用于电影项目；在实践中，电影可以具有一个演职员表，并且该演职员表可以被表示成演员姓名以及演员年龄、演员在电影中的角色、演员性别等等的列表。关于结构化元数据表示的典型实例是由 MPEG-7 描述标准提供的。

### 上下文元数据(CONTEXT METADATA)

交互上下文（简称为“上下文”）是用于基于内容的服务的用户内容交互的一个重要元素。实际上，每一个用户内容交互都是在上下文内部发生的，并且该上下文通常会影响用户在某个指定上下文内部使项目引起关注以及在别的上下文中不使该项目引起关注的用户偏好。

交互上下文信息还可以关联于一个或多个元数据，这些元数据被称为上下文元数据。

该交互上下文是由不同的方面形成的。通常，最重要的方面是：“日期和时间”（何时发生交互）、“用户位置”（何处发生交互）、“交互设备”（供用户用于交互使用的设备）、“内容通道”（通过该通道来发生交互）、“环境状态”（交互期间）、“物理世界状态”（交互期间）、“用户状态”（交互期间）。

“用户位置”可以采用若干种方式以及若干种形式来提供，例如：从 GPS 系统或蜂窝网络获取的空间坐标、由短距离无线信标系统提供的逻辑坐标、传送关于用户所在位置的元数据描述。

举例来说，“交互设备”的特征可以是：设备的移动性（也就是移动或固定设备）、设备的图形显示能力（例如大小、分辨率、显示颜色数量）、设备的声音能力（例如音频通道数量），它的商标和模型。

在诸如按需点播媒体递送环境之类的某些环境中，每一个交互都包括选择“内容通道”，例如电视频道或电影供应商。

举例来说，“环境状态”可以从交互设备中的环境选项的设置中得到。举个例子，移动电话设备可以被设置成“会议”、“工作”或“家庭”模式，或者其状态可以被设置成“振铃”或“静音”模式。

举例来说，与“物理世界状态”相关的信息可以由检测温度、照明状态、湿度、压力、风速的传感器提供。

举例来说，与“用户状态”相关联的信息可以由检测用户身体加速度或是其某些生理参数的传感器来提供，其中举例来说，这些生理参数可以是心率、血压、皮肤导电性（以便确定其紧张/放松状态）。

在下文中，从一个或多个物理设备直接得到的上下文元数据将被称为“原始上下文元数据”。举例来说，该物理设备可以是定时器、传感器、开关（硬件或软件）。通常，这些设备是集成在包含用户接口的终端设备（例如移动电话、个人计算机等等）内部的。

### 元数据表示(MEADATA REPRESENTATION)

为了简化各种类型元数据的存取和处理，较为有利的是使用一种统一和便于扩展的格式。

目前业已发现的是，MPEG-7 描述标准已特别适合本发明。在本说明书稍后描述的实施例中，该标准已被用作所有元数据、即原创和派生元数据的格式。特别地，内容项元数据是作为零个或多个“相关材料”块列表来组织的（参见图 4），其中每一个相关材料都描述了一个可以引用实际内容或是代表元数据的 XML【可扩展标记语言】块的同种信息块。引用始终都会指向相关的附件块，而所述附件块则转而保持实际内容（或元数据）或是提供可以发现内容（或元数据）所在的 URL【统一资源定位符】。这种组织可以适合不同的存储策略，同时向内容成分以及元数据提供集中接入点。除了包含在每个相关材料中

的信息之外，内容元数据还可以保持每一个已发布内容应该拥有的通用信息群组，例如创建日期，或是服务应用逻辑所需要的其他信息，例如用于指示是否可以认为内容可用的有效状态标记。

以下给出的是依照 MPEG-7 标准编码并且报告了成分的某些公共特征的“相关材料”。

```

<RelatedMaterial>
    <MediaType>Image</MediaType>
    <MediaInformation>
        <MediaIdentification>
            <Identifier>image_1</Identifier>
        </MediaIdentification>
        <MediaProfile Master="true" id="image_1_profile">
            <MediaFormat>
                <FileFormat>image/jpeg</FileFormat>
                <AspectRatio Height="322" Width="122"/>
                <FileSize>1002</FileSize>
            </MediaFormat>
            <MediaInstance>
                <Identifier>image_1</Identifier>
                <InstanceLocator>
                    <MediaURL idref="attachment_1"/>
                </InstanceLocator>
            </MediaInstance>
        </MediaProfile>
    </MediaInformation>
</RelatedMaterial>

```

上述“相关材料”指的是下文中用于实际媒体原始数据、尤其是图像的“相关附件”，其中该“相关附件”是根据 MPEG-7 标准编码的。

```

<RelatedAttachment id="attachment_1">
    <AttachmentData Storage="internal" Encoding="Base64Binary">
        <EncodedData>
            BASE 64 ENCODING OF THE IMAGE NOT SHOWN
        </EncodedData>
    </AttachmentData>
</ RelatedAttachment >

```

相同的方法还可以用于引入派生元数据：新的“相关材料”将被添加到列表中，如果需要的话，用于标引新的“相关附件”的引用可以适用于不适合这个“相关附件”模式的元数据 XML 块。

虽然所描述的元数据表示是优选的，但是应该理解，本发明并不

局限于这种元数据表示。

### 派生元数据 (DERIVED METADATA)

本发明提供了除预先指定的元数据以及原始上下文元数据之外的其他元数据，由于这些元数据是从交互事件内部的内容信息和/或上下文信息或是作为一个或多个交互事件的结果而被推导得到的，因此，这些元数据被称为“派生元数据”。

特别地，派生内容元数据可以直接从内容项中得到，也就是说，它可以直接从内容项的内容中得到，此外，派生内容元数据也可以间接地从内容项中得到，例如从关联于内容项的原创内容元数据中得到。当内容项被发布并且随后被软件程序生成时，派生内容元数据并不是直接可用的。

类似地，派生内容元数据可以直接从上下文项中得到，也就是说，它可以从交互事件的上下文中得到，或者从上下文项（即从关联于上下文项的上下文元数据）中间接得到。当检测到交互事件上下文并且随后由软件程序生成该交互事件上下文时，派生内容元数据并不是直接可用的。

派生元数据可以提供关于内容和上下文的更完整和更有用的信息。

派生元数据特别适合由软件程序自动处理。

在下文中将会述及关于元数据推导 (derivation) 的若干示例。

对从文本项内容（也就是文本自身）中推导得到的元数据来说，其示例是在文本中出现的单词连同每一个单词的出现次数的列表，并且它被称为文本的“词袋 (bag of word)”表示；这些元数据给出了关于文本的整体词汇合成的信息。

对从文本项内容（也就是文本自身）中推导得到的元数据来说，它的其他示例包括文本量度，也就是针对文本所计算的数值参数，例如文本的全局长度、句子的平均长度或是属于文本的段落，句法结构的平均嵌套深度，Gunning 的 Fog 索引（例如用于英文文本）以及 Guplease 索引（例如用于意大利语文本）。

举例来说，对从图像项的内容（也就是图像自身）中推导得到的元数据来说，其示例包括：

- 亮度直方图，它是光强度在数字图像像素上的分布——它给出的是关于图像上的量度和对比度的信息；

- 颜色直方图，它是基色分量（红、绿、蓝）在数字图像像素上的分布——它给出的是关于图像颜色组成的信息；

- 图像的空间频率分量，举例来说，该分量是借助二维傅里叶变换计算的——它给出的是关于图像中的图案和纹理的呈现的信息；

- 几何分类元数据，举例来说，该元数据是借助几何散列技术产生的——它给出的是关于图像中诸如线条、弧线、椭圆形、多边形之类的形状的呈现的信息；

- 图案分类元数据，举例来说，该元数据是通过图案识别算法产生的——它给出的是关于图像中的特定信息，例如人脸、动物、植物、风景、建筑、标记、技术绘图、涂色、漫画的呈现的信息；

- 文本元数据，举例来说，该元数据是通过光学字符识别技术产生的——它给出的是关于在图像中出现的字母、数字和单词的信息；

举例来说，从声音项的内容（也就是从声音自身）中得到的元数据包括：

- 音频频谱成分，举例来说，该成分是借助快速傅里叶变换计算的——它给出的是关于声音的特性和组成的信息；

- 音频波形——它给出的是关于声音动态特性的信息；

- 图案分类元数据，举例来说，该元数据是通过图案识别算法产生的——它给出的是关于特定特征，例如特定音乐、语音、拍击声、爆炸声在音轨中的呈现的信息；

- 文本元数据，举例来说，该元数据是通过语音识别技术产生的——它从音轨中提取发出声音的单词或句子。

通过使用特定分析和算法，可以从视频项内容（也就是从视频自身）中推导得出元数据。

场景分段分析技术可以给出关于视频时间结构的信息。例如，这

种分析可以告知电影包括指定数量的场景，其中该场景的百分比是用强烈的运动行为表征的，而另一个百分比则是用声音很大的音乐的呈现来表征的。

运动对象识别算法可以给出关于特定对象在视频中的呈现的信息，其中该呈现是用特定的运动行为表征的，例如正在行走的人、交谈或唱歌的人、行驶中的汽车、正在掉落的物体、正在打开的门。

如果将视频分解成一系列静止图像，那么，倘若存在将图像序列上的最终得到的元数据平均的方式，则可以将某些用于静止图像的元数据提取技术应用于视频。

举例来说，从 3D 模型中推导得到的元数据包括：

- 整体面积、整体体积、凸性、分形维度；

- 图案分类元数据，其中举例来说，该元数据是通过图案识别算法产生的——它给出的是关于特定 3D 形状，例如方框、管道、轮形、线路、人形、物体形状的呈现的信息。

如先前所述，通常，元数据是可以从其他任何元数据中得到的。

举例来说，从数值元数据开始，符号范围可以通过使用能够聚集数值的离散化技术来产生；这是一种提供关于数值的更紧凑和更语义性的表示的方式。

元数据还可以通过使用本体（ontology）来得到。本体是一种使用了机器可读表示的概念化形式。本体可以用于组织元数据中的分类和关系；这样则允许将用户偏好模型构建到高阶语义分类和概念上。

举例来说，关于本体的信息可以在 W3 联盟网站（当前位于地址“<http://www.w3.org>”）以及 Christiane Fellbaum 编著并由 MIT Press 于 1998 年 5 月出版的“WordNet: An Electronic Lexical Database”一书中找到。

在下文中提供了与通过本体得到的元数据相关的两个示例。

用于时间的简单本体是依照“日间时间”和“夜间时间”的时间值分类；用于日期的简单本体是依照“工作日”和“周末日”的日期值分类。在用户偏好模型的构造中，时间值元数据和日期值元数据是没有意义

的；而“日间时间/夜间时间”和“工作日/周末日”元数据则有可能会更为有效。举个例子，用户偏好模型可以声明用户喜爱在周末日夜间与关联于某个分类的内容项进行交互，而在同时包含日间和夜间的内的工作日中则不喜欢这些内容项。

本体特别适合产生分类元数据。让我们以三个文本项 A、B、C 为例，其中每一个文本项都包含了一篇新闻文章。文章 A 讲述的是人类肺部的计算机模型，并且包含（除了别的因素之外）元数据单词“计算机”和“哮喘”。文章 B 讲述的是机器人辅助手术，并且包含了元数据单词“软件”和“外科医生”。文章 C 讲述的是因特网，并且包含了元数据单词“网站”。通过将元数据单词与词汇本体相联系，可以扩充文章 A 和 B，其中举例来说，所述扩充是通过抽象分类“医疗”来实施的，并且该抽象范畴可以添加到这两篇文章的元数据中。与之类似的是，所有这三篇文章 A、B、C 的元数据可以通过抽象分类“计算机科学”和“技术”来扩充。由此，用户对这些文章的兴趣有可能涉及该抽象分类，而不是单个字词。

#### 元数据推导 (METADATA DERIVATION)

本发明的一个重要方面是产生（也就是推导）元数据。这一点是根据推导规则（derivation rules）来实现的。

推导规则与应用于内容项和/或上下文信息和/或元数据的算法相对应，对应用了推导算法的数据来说，该数据通常被称为源。

推导规则规定的是所要应用的算法，其中该规定是通过引用实施该算法的插件模块来实现的，此外，该推导规则还规定了所要处理的源。

特别地，在这里可以提供下列类型的源：

- 内容成分；
- 原创内容元数据；
- 原始上下文元数据；
- 派生元数据（也就是从其他推导规则获取的元数据）；
- 扩展分析。

举例来说，推导规则可以借助规定上述元素（也就是用于实施所要使用的算法的模块、所需要的参数，以及所要使用的输入源）的 XML【可扩展标记语言】文档来描述。

在下文中给出了关于推导规则的某些实例。

第一组实例涉及新闻浏览器应用。在这个应用中，内容项是新闻文章。每一个内容项都包含了两个内容成分：新闻文章标题和主体，并且这二者全都采用了文本形式。每一个内容项的内容都与作为原创元数据的文章的日期、分类、来源以及作者名称相关联；特别地，这个元数据包含在内容项中。

第一元数据推导规则是如下定义的：

```
<DerivationRule ID="Body_BagOfWords">
  <Module name="BagOfWords">
    <Parameter name="language">Italian</Parameter>
  </Module>
  <Source type="ContentComponent">news.body</Source>
  <Destination type="DerivedMetadata">bodyBow</Destination>
</DerivationRule>
```

通过执行插件模块提供的名为“BagOfWords（词袋）”的算法，该规则规定：获取构成新闻文章主体的文本数据，以此作为输入（源），以及产生其“词袋”表示，以此作为输出（目的地），也就是在主体中出现的单词连同每一个单词的出现数量的列表。

第二元数据推导规则是如下定义的：

```
<DerivationRule ID="Body_WordOntology">
  <Module name="WordOntology">
    <Parameter name="ontology">LexicalOntology1</Parameter>
    <Parameter name="language">Italian</Parameter>
    <Parameter name="hyponyms">2</Parameter>
    <Parameter name="topsemanticlevel">true</Parameter>
    <Parameter name="minoccur">3</Parameter>
  </Module>
  <Source type="ContentComponent">news.body</Source>
  <Destination type="DerivedMetadata">bodyWordOnto</Destination>
</DerivationRule>
```

通过执行名为“WordOntology”的插件模块所提供的算法，该规则规定：获取构成新闻文章主体的文本数据，以此作为输入，以及产

生相关的概念词集合，以此作为输出。

该规则执行以下步骤：

- 从源文本中产生“词袋”
- 对照“LexicalOntology1”词汇本体来匹配源自所获取的“词袋”的每一个单词。对每一个匹配单词来说，它会提取：

- 与单词相关联的上位体（hyperonyms），两个等级以上（hypernym=2）。类似 LexicalOntology1 的词汇本体将被组织成树。如果从树的叶子朝着根部运动，那么这意味着从特定含义的单词朝着代表一个或多个抽象概念的单词移动。如果给出了某个单词，那么树中位于该单词上方的单词将被称为“上位体”（例如“医生”-“个人”-“活物”；“医生”-“专业”-“工作者”）。实际上，在本体中可以具有一个以上的树：

- 与单词相关联的顶级语义分类（topsemanticlevel=“真”）（例如，单词“医生”属于顶级语义分类“医学”）。
- 对所提取的概念（上位体和顶级语义分类）的出现进行计数，由此仅仅保持至少出现了三次的概念（minoccur=3）。通过执行这个处理，可以限制最终的概念数量，由此仅仅保持最接近的概念。

总的结果是一个新闻文章文本主体的语义表示，该表示采用了来自词汇本体的概念单词集的形式，并且每一个表示都具有其出现次数。

以下的第三和第四推导规则分别与上文中的第一和第二推导规则类似，但是其来源是新闻项标题而不是主体。然而应该指出的是，如在“Destination tag（目的地标签）”中声明的那样，从标题产生的元数据将会形成一个与从主体产生的元数据相分离的元数据集合。

```

<DerivationRule ID="Title_BagOfWords">
  <Module name="BagOfWords">
    <Parameter name="language">Italian</Parameter>
  </Module>
  <Source type="ContentComponent">news.title</Source>
  <Destination type="DerivedMetadata">titleBow</Destination>
</DerivationRule>

<DerivationRule ID="Title_WordOntology">
  <Module name="WordOntology">
    <Parameter name="ontology">LexicalOntology1</Parameter>
    <Parameter name="language">Italian</Parameter>
    <Parameter name="hyponyms">2</Parameter>
    <Parameter name="topsemanticlevel">true</Parameter>
    <Parameter name="minoccur">3</Parameter>
  </Module>
  <Source type="ContentComponent">news.title</Source>
  <Destination type="DerivedMetadata">titleWordOnto</Destination>
</DerivationRule>

```

以下规则将会通过执行插件模块“TextMetrics(文本量度)”提供的算法来产生与新闻主体长度相对应的数值元数据，并且这个元数据是作为其内单词的计数来提供的。

```

<DerivationRule ID="Body_Length">
  <Module name="TextMetrics">
    <Parameter name="metric">textLength</Parameter>
    <Parameter name="unit">word</Parameter>
  </Module>
  <Source type="ContentComponent">news.body</Source>
  <Destination type="DerivedMetadata">bodyTextLength</Destination>
</DerivationRule>

```

以下规则将会产生与新闻文章主体中的平均句子长度相对应的数值元数据。

```

<DerivationRule ID="Body_AvgSenLen">
  <Module name="TextMetrics">
    <Parameter name="metric">AvgSenLen</Parameter>
  </Module>
  <Source type="ContentComponent">news.body</Source>
  <Destination type="DerivedMetadata">avgSenLen</Destination>
</DerivationRule>

```

以下规则将会产生用于表述估计得到的阅读新闻文章主体的易读性的数值元数据。

```
<DerivationRule ID="Body_ ReadabilityIndex">
<Module name="TextMetrics">
<Parameter name="metric">ReadabilityIndex</Parameter>
</Module>
<Source type="ContentComponent">news.body</Source>
<Destination type="DerivedMetadata">bodyTextLength</Destination>
</DerivationRule>
```

以下的三个规则将会产生与新闻主体文本中的日期、图表(数字、百分比、价格)以及人名的出现次数相对应的三个数值元数据。

```
<DerivationRule ID="Body_Dates">
<Module name="TextMetrics">
<Parameter name="metric">Dates</Parameter>
</Module>
<Source type="ContentComponent">news.body</Source>
<Destination type="DerivedMetadata">bodyDates</Destination>
</DerivationRule>

<DerivationRule ID="Body_Numbers">
<Module name="TextMetrics">
<Parameter name="metric">Figures</Parameter>
</Module>
<Source type="ContentComponent">news.body</Source>
<Destination type="DerivedMetadata">bodyFigures</Destination>
</DerivationRule>

<DerivationRule ID="Body_PeopleNames">
<Module name="TextMetrics">
<Parameter name="metric">PeopleNames</Parameter>
</Module>
<Source type="ContentComponent">news.body</Source>
<Destination type="DerivedMetadata">bodyPNames</Destination>
</DerivationRule>
```

第二组实例涉及音乐目录浏览器应用。在这个应用中，项目是音乐片段。每一个项目都包含了单个内容成分：编码音乐片段的音频文件(例如以 MP3 格式)。每一个项目还包含了音乐的标题、日期、流派、表演者姓名以及作者姓名，以此作为原创元数据。

在下文中定义了第二集合的四个元数据推导规则：

```

<DerivationRule ID="MusicBeatSpeed">
  <Module name="AdvancedSpectralAnalysis">
    <Parameter name="metric">BeatSpeed</Parameter>
  </Module>
  <Source type="ContentComponent">music.audiofile</Source>
  <Destination type="DerivedMetadata">BeatSpeed</Destination>
</DerivationRule>

<DerivationRule ID="MusicVocalImpact">
  <Module name="AdvancedSpectralAnalysis">
    <Parameter name="metric">VocalImpact</Parameter>
  </Module>
  <Source type="ContentComponent">music.audiofile</Source>
  <Destination type="DerivedMetadata">VocalImpact</Destination>
</DerivationRule>

<DerivationRule ID="MusicSoundBrigthness">
  <Module name="AdvancedSpectralAnalysis">
    <Parameter name="metric">SoundBrigthness</Parameter>
  </Module>
  <Source type="ContentComponent">music.audiofile</Source>
  <Destination type="DerivedMetadata">SoundBrigthness</Destination>
</DerivationRule>

<DerivationRule ID="MusicLoudnessDynamic">
  <Module name="AdvancedSpectralAnalysis">
    <Parameter name="metric">LoudnessDynamic</Parameter>
  </Module>
  <Source type="ContentComponent">music.audiofile</Source>
  <Destination type="DerivedMetadata">LoudnessDynamic</Destination>
</DerivationRule>

```

上述规则产生了分别表示“节拍速度”（也就是音乐片段的节奏特征的量化量度）、“人声效果”（也就是音乐片段中人声成分相对于器乐成分的加权）、“声音亮度”（也就是声音亮度的量化量度）以及“音量动态特性”（也就是声音音量随时间而发生的改变的量化量度）。该规则可以通过执行名为“AdvancedSpectralAnalysis”的插件模块提供的算法，以及通过应用于编码在音频文件中的音频信号的频谱分析技术来实现。

以下规则产生的是用于表述音乐片段年代的文本标记元数据，其中该元数据是从该片段的年份开始的。该规则可以通过由名为

“NumericalDiscretizer”的插件模块提供的离散化技术、也就是语义描述（年代）中的数值元数据的伸缩范围来实现。

```
<DerivationRule ID="MusicDecade">
  <Module name="NumericalDiscretizer">
    <Parameter name="intervals">
      -1960=old,
      1961-1970=sixties,
      1971-1980=seventies,
      1981-1990=eighties,
      1991-2000=nineties,
      2001=recent
    </Parameter>
  </Module>
  <Source type="AuthoredMetadata">AuthMetadata.year</Source>
  <Destination type="DerivedMetadata">Decade</Destination>
</DerivationRule>
```

以下规则产生的是用于表述关于音乐主要表演者的流行度的粗略估计的数值元数据。该估计是通过将主要表演者的姓名提交到名为“SearchEngineQuery”的插件模块所提供的万维网搜索引擎以及获取估计的点击（包含该姓名的网站）次数作为结果来实现的。

```
<DerivationRule ID="MusicPerformerPopularity">
  <Module name="SearchEngineQuery">
    <Parameter name="searchengine">yyy</Parameter>
    <Parameter name="metric">EstimatedHitsNumber</Parameter>
  </Module>
  <Source type="AuthoredMetadata">AuthMetadata.MainPerformerName</Source>
  <Destination type="DerivedMetadata">PPopularity</Destination>
</DerivationRule>
```

### 使用扩展分析源的推导规则

扩展分析提供了一种可以通过推导规则来影响的特殊类型的来源。这种来源规定了分析过程在储存库中包含的项目的整个子集（乃至所有项目）上的应用，以便获取关于域结构的整体分析。换句话说，使用扩展分析的推导规则并不局限于仅仅使用单个项目中包含的信息来提取元数据，而是可以为每一个项目产生新的元数据，并且这个新的元数据考虑了项目自身的整体结构。所述域是推导技术的应用领域，也就是向用户提供个性化选定内容的领域。

这些算法实际执行的是由扩展分析所规定的分析，并且这些算法

可以由专用软件模块或是通用元数据生成器模块来执行；如果提供了专用模块，那么该专用模块可以是一个在需要时由通用元数据生成器模块调用的“插件”模块。

以下是基于扩展分析的推导规则；该规则作用于如上所述的音乐目录应用。

```

<DerivationRule name="NearestAudioClusters">
  <Module name="NearestClusters">
    <Parameter name="numclusters">1</Parameter>
    <Parameter name="distancemetric">euclidean</Parameter>
  </Module>
  <Source type="DerivedMetadata">VocalImpact</Source>
  <Source type="DerivedMetadata">BeatSpeed</Source>
  <Source type="DerivedMetadata">SoundBrightness</Source>
  <Source type="DerivedMetadata">LoudnessDynamic</Source>
  <Source type="ExtendedAnalysis">
    <ExtendedAnalysis ID="SpectralFeatureClusterAnalysis">
      <ContentRange type="query">All</ContentRange>
      <Module name="NumericalClusterAnalysis">
        <Parameter name="method">Ward</Parameter>
        <Parameter name="maxclusters">10</Parameter>
      </Module>
      <Source type="derivedMetadata">VocalImpact</Source>
      <Source type="derivedMetadata">BeatSpeed</Source>
      <Source type="derivedMetadata">SoundBrightness</Source>
      <Source type="derivedMetadata">LoudnessDynamic</Source>
    </ExtendedAnalysis>
  </Source>
  <Destination type="DerivedMetadata">NearestAudioCluster</Destination>
</DerivationRule>

```

在上述推导规则中作为扩展分析而被规定的分析会采用四个派生数值元数据作为输入，这些数值元数据是通过应用先前示例中阐述的推导规则来获取的。这四个数值描述了每一个音乐片段的四个相关音频特征。

扩展分析规定的是对包含在内容储存库中的所有项目（也就是音乐片段）执行“群集分析”。群集分析是一种已知的统计技术，其中如果用相似值表征的项目群组（也就是群集）代表的是项目域中令人感兴趣的规则，那么该统计技术将被用于识别这些群组。对音乐域来说，

最终得到的群集可以将共享相似音频外部特征、例如紧密性的音乐片段聚集到相同的音乐流派中。

实际执行该分析的算法是由名为“**NumericalClusterAnalysis**”的插件模块提供的。

特别地，本示例中的扩展分析规定了所要应用的群集分析方法（Ward 方法），必须应用分析的项目的范围（储存库中的所有项目），以及所要提取的群集的最大数量（10 个群集）。

该分析允许推导规则为指定项目（音乐片段）产生用于指示群集与其紧密度的元数据。这个新的元数据标识的是音乐片段在群集所表示的“音乐风景”中的位置。

### 机器学习方法

机器学习方法允许计算机系统从属于特定应用领域（也就是域）的实际数据集合执行自动学习（也就是通过软件程序）。在给出了这种数据集合的情况下，机器学习方法能从数据自身当中提取图案和关系。

已被学习的图案和关系会由机器学习方法以一种形式量化模型来进行编码，其中该模型根据所使用的机器学习技术而采用不同的形式。关于模型形式的示例包括逻辑规则、数学等式以及数学图表。

机器学习方法的目标是更好地理解和量化数据内部的图案以及数据之间的关系，以便获取作为数据表示的模型。

大多数机器学习方法都使用特征矢量表示。如果将这些方法应用于构建与所提供的内容相关的用户偏好的预测模型，那么每一个特征矢量都会与一个内容项相关联，并且包括：

- 独立特征，其中每一个特征都与一个关联于该内容项以及优选关联于原始或派生上下文元数据的派生或预先指定的元数据相对应，以及

- 一个或多个目标特征，这些特征是由用户作为反馈（显性或隐性）提供并与所提供的内容项相关的分数表示的。举例来说，该反馈是由从 1~10 的数值表示的，其中较高的值对应正反馈。

然后，数据集合的每一个实例被表示为特征矢量。对单个目标特征来说，代表实例的矢量是“n+1”维的，并且采用如下形式：

<特征 1, 特征 2, ..., 特征 n, 目标特征>

该特征矢量模型是域数据的形式表示，并且适合大多数机器学习方法。

在 Tom Mitchell 的“Machine Learning”，McGraw-Hill, 1997 中可以找到关于机器学习方法及其应用的大量论述。

优选地，数据集（将要由机器学习方法处理，以便构建用户简档的预测模型）包含了内容元数据（原创和派生）以及上下文元数据（原始和派生），并且以此作为独立特征。而用户反馈则是目标特征。机器学习方法的目标是发现一个用于预测用户偏好的模型（被称为用户模型或预测模型），也就是用于表述元数据与用户反馈之间的关系的机器学习模型。然后，当新内容项可用时，由此获取的预测模型可以用于估计用户对这些新内容项的评价。

对用特征矢量表示的数据集来说，该数据集的实例对应于单个交互事件，其中用户将会表述其对一个内容项的偏好，并且将会采用如下形式：

<内容元数据 1, ... , 内容元数据 m, 上下文元数据 1, ... , 上下文元数据 p, 用户投票>

其中  $m+p=n$ 。

如果用户表达了对于多个内容项的偏好，那么将会创建多个特征矢量，并且这些特征矢量在形式上可以用矩阵 $(n+l) \times q$  来表示，其中  $q$  是交互事件数量。举例来说，如果用户表达了其对 10 个内容项的偏好，那么将会创建矩阵 $(n+l) \times 10$ 。然后，选定的机器学习算法将被应用于该矩阵。

目前有若干种公知的机器学习方法可以用于这个目的，这其中包括决策树，关联规则、神经网络以及贝叶斯方法，以及那些专门设计用于构建用户偏好模型的任务的方法。

在下文中参考先前所述的音乐目录应用而给出了在构建用户简

档的过程中使用机器学习方法的示例。

在本示例中，例示内容项（音乐片段）是用两段元数据表示的：

- **MusicGenre**, 音乐流派（作为原创数据提供），
- **MusicBeatSpeed**, 每分钟的音乐片段的节拍（作为通过应用“**MusicBeatSpeed**”推导规则的派生元数据提供）。

交互上下文是用下列（派生）上下文元数据表示的：

- “**Time**（时间）”，它可以具有“日间”或“夜间”这两个值中的任何一个，并且该元数据论述的是用户与内容项的交互是在日间还是夜间发生（作为派生元数据并且通过应用基于简单时间本体的推导规则来提供）。

用户偏好是由以下特征给出的：

- “**UserVote**（用户投票）”，它可以具有“喜欢”或“不喜欢”这两个值中的任何一个，并且这个元数据论述的是用户为音乐片段提供的肯定还是否定分数。

由此，参考先前了解的关于所述域的常规机器学习表示，用于用户表达其对音乐片段偏好的单个事件可以采用如下的矢量形式：

**<MusicGenre, MusicBeatSpeed, Time, UserVote>**

以下的用户/项目交互数据集合是作为示例给出的：

ID	Genre	MusicBeatSpeed	Time	UserVote
1	摇滚	128	日间	喜欢
2	舞曲	130	日间	喜欢
3	舞曲	125	夜间	不喜欢
4	舞曲	130	夜间	不喜欢
5	摇滚	130	夜间	不喜欢
6	古典	55	日间	不喜欢
7	古典	60	日间	不喜欢
8	舞曲	70	夜间	喜欢
9	爵士	65	夜间	喜欢
10	古典	75	夜间	喜欢
11	爵士	60	夜间	喜欢
12	摇滚	125	日间	不喜欢
13	舞曲	135	夜间	喜欢

通过将决策树机器学习方法应用于上述数据集合，可以产生包含下列规则的用户偏好模型。

IF Time = "日间" AND MusicBeatSpeed >= 125 THEN UserVote = "喜欢"

IF Time = "夜间" AND MusicBeatSpeed <= 75 THEN UserVote = "喜欢"

应该指出的是，用于产生预测分数的上述用户偏好规则并不是用于产生派生元数据的推导规则。

上文中依照用户偏好规则表述的简单预测模型论述的是，这个特定用户在日间喜欢快节奏音乐（MusicBeatSpeed $\geq 125$ ），而在夜间则更喜欢镇静的音乐（MusicBeatSpeed $\leq 75$ ）。

应该指出的是，这个模型对上述数据集合的大多数情形都是成立的（12/13），但并不是对所有情形全都成立。

### 实施例详述

在下文中将会特别参考图 1 框图（服务应用）来提供关于本发明有利实施例的详细描述；在该图中使用了两个符号，即代表软件模块的矩形形状和代表储存库的圆柱形形状。

这个实施例参考的是为用户提供基于内容的服务的服务供应商。该服务可以是 PULL 类型的，也可以是 PUSH 类型的，还可以同时是这两种类型的。

基于内容的服务将选定内容项提供给用户。内容项可以由服务供应商直接或间接地递送给用户，其中举例来说，该内容项可以通过提供内容项所在或是可以访问内容的地址（例如因特网地址）来递送。内容项通常是由内容供应商通过分组数据网（例如因特网）或移动电话网（例如 UMTS 网络）之类的通信网络直接提供的。

基于内容的服务提供了构建和维护用户简档，以便提供更好的内容项选择。

这种基于内容的服务可以分成两种活动：

- 处理内容项，

- 处理用户简档。

#### 处理内容项

处理内容项的活动包括：

- 接收来自用户的请求，
- 根据用户请求来选择内容项，
- 优选地，根据用户的呈现简档来格式化选定内容项和将所述内容项呈现给用户（也就是依照服务应用逻辑个性化内容呈现），以及
- 将选定内容项提供给用户。

在 PULL 模式中，一旦接收到用户请求，那么服务供应商将会根据该请求来识别一组内容项，然后，该服务供应商将会对这组内容项执行上述步骤；这意味着某些选定内容项通常是在用户请求之后不久作为针对用户的答复而提供的。

在 PUSH 模式中，服务供应商接收来自用户的请求，并且在不立即回复以及通常不立即处理的情况下将其存储，然后存在两种可能性。根据第一种可能性，服务供应商周期性识别所有新发布的内容项，然后它会对所有新发布的内容项执行上述步骤。根据第二种可能性，服务供应商在发布内容项的时候就立即识别该内容项，然后则会对新发布的内容项执行上述步骤。在 PUSH 模式中，内容项的提供可以分两个步骤来执行：首先，服务供应商简单地向用户告知其感兴趣的某些内容项可用，然后，一旦用户表达了其接收这些内容项的愿望，那么服务供应商将会发送（直接或间接）这些内容项；此外，用户还可以表达其只接收这些内容项中的一部分的意愿。

用户的每一个输入都会由服务前端模块（SFEM）、例如 PC 或移动终端接收和处理。用户的请求将会发送到服务应用逻辑模块（SALM），该模块则会嵌入特定于所要提供的基于内容的服务的逻辑。此外，模块 SFEM 向模块 SALM 发送与用户请求相关联的原始上下文元数据（例如日期和时间，用户位置等等）。

当模块 SALM 接收到来自模块 SFEM 的用户请求时，它会产生相应的内容查询（图 2 中的步骤 201）。根据这个内容查询（从用户

请求中推导)及其服务逻辑,模块 SALM 将会依照该内容查询而在内容项储存库 CIR 中识别第一内容项集合(图 2 中的步骤 202)。举例来说,依照该内容查询,服务应用逻辑会在储存库 CIR 中识别与电影和电视连续剧相关的第一内容项集合。

储存库 CIR 存储了内容项以及预先指定给所存储的内容项的内容元数据(通常是原创元数据)。如以下部分中更详细说明的那样,储存库 CIR 还可以存储与内容项相关联的派生内容元数据,其中该内容元数据是从先前的元数据生成处理产生、也就是从先前的内容查询中触发的。

如果将用户反馈施加到所述第一内容项集合,那么,由于服务应用可以向用户询问的输入数量、例如显性偏好(“我喜欢”或“我不喜欢”)通常是受到限制的,因此将会出现可用性问题。此外,在很多情况下,关于这种输入的精确的形式化是不能实行的。

为了避免不良过滤,以及为用户保持精确内容的最终目标,模块 SALM 将会要求匹配器模块(MMM)根据用户简档来产生所述第一集合的内容项排序(图 2 中的步骤 203)。然后,模块 SALM 从模块 MMM 接收的排序可以由模块 SALM 使用,以便滤除低分数内容项,选择最佳分数内容项,以及重排序所保留的内容项。这个处理可以根据已知的方法来完成。

由此,模块 SALM 将会借助 MMM 来过滤这个第一集合的内容项(图 2 中的步骤 204),以便选择所识别的第一内容项集合内部的第二内容项集合。模块 MMM 是过滤活动的关键部件,并且它负责考虑用户的简档(或用户简档),这一点将会在下文中进行说明。优选地,通过 MMM 的过滤活动所获取的第二内容项集合是第一内容项集合的一个子集,尽管如此,虽然第二内容项集合是根据排序偏好来进行排序的,但是,在这里并不排除第二内容项集合包含了第一集合的所有内容项,由此用户可以依照与项目相关的排序来查看这些项目。

模块 SALM 累积、变换并且格式化子集中的内容项,以便将其呈现给用户(图 2 中的步骤 205);与呈现选定的内容项不同,在这

里可以仅仅将其告知用户。并且呈现和/或通知处理是由模块 SFEM 执行的。

根据本发明的优选实施例，图 1 的架构包括：

- 用户简档储存库（UPR），
- 元数据生成器模块（MGM）。

模块 MGM 提供了一组推导规则，以便产生派生元数据（内容和/或上下文）。这些推导规则是以推导算法为基础的。在图 1 的实施例中，这些算法处于模块 MGM 外部，并且是由推导算法模块（DAM）提供的，而该模块则是通过“插件”技术实现的；这样做允许具有本地和远端存储的算法，并且该算法将会由模块 MGM 进行调用。

**模块 MMM：**

- 从储存库 UPR 中检索当前用户的用户简档；以及
- 从储存库 CIR 中为所述第一内容项集合中的每一个内容项检索与之关联的内容元数据，其中该内容元数据是预先指定的（通常是原创的）以及可能是派生的（作为先前交互事件的结果）。

此外，模块 MMM 从其他模块接收与当前上下文相关联的上下文元数据。特别地，原始上下文元数据是通过模块 SALM 而从模块 SFEM 接收的，以及派生上下文元数据是从模块 MGM 接收的。

模块 MMM 将原始上下文元数据发送到模块 MGM，由此请求产生派生元数据（至少从原始上下文元数据开始），此外它还接收所产生的派生上下文元数据。这样一来，至少某些上下文元数据可以是同时（on the fly）推导得到的，也就是在与用户交互的过程中推导得到的。

然后，模块 MMM 会将用户简档应用于与所识别的第一内容项集合内部的每一个内容项相关联的内容元数据（预先指定的和派生的），并且优选将其应用于与当前上下文相关联的上下文元数据（原始和派生的）。这样一来，模块 MMM 将会对照用户简档来匹配第一内容项集合。在本实施例中，用户简档至少包含了一个预测模型（优选是通过机器学习方法产生的）。该预测模型被应用于所述第一集合

的每一个内容项，并且为每一个内容项产生一个预测投票。与第一内容项集合相关联的预测投票集合将会由模块 MMM 使用，以便产生第一集合的内容项排序。所述排序将被提供给模块 SALM，该模块则会定义一个第二内容项集合，该第二内容项集合是依照内容项的有序集合或是依照作为所述排序的结果而被选择的第一内容项集合的子集（例如只包括第一集合中最佳排序的内容项）来形成。

优选地，本发明的实施例还提供：

- 用户交互记录器模块 IRM，以及
- 交互历史储存库 IHR。

交互历史可以采取记录序列的形式，每个记录包含一些例如与用户请求（或相应的查询）、系统回复、上下文、元数据、用户反馈有关的信息。优选地，使用合成格式（例如链接或索引而不是物理项）。通常，交互历史的每个记录对应于不同的交互事件。

模块 IRM 的任务是更新交互历史（图 2 中的步骤 206）。为此，模块 IRM 直接将用户请求（从模块 SFEM 接收的）记录到储存库 IHR 中。此外，模块 SALM 还通过模块 IRM 将其对用户请求的答复（以内容项形式）记录到储存库 IHR 中。通过模块 IRM，模块 SALM 还可以将预测投票和/或用于答复用户的所有或部分（内容和/或上下文）元数据记录到储存库 IHR 中。

非常有利的是，为了节省储存库 IHR 中的存储空间，在储存库 IHR 中仅存储一种类型的元数据，即原始上下文元数据（这是因为在任何时间，其他元数据可以从储存库 CIR 中检索得到，或者由模块 MGM 产生）；该处理可以由直接从模块 SFEM 接收此类元数据的模块 IRM 来执行。

服务应用可以要求用户提供其反馈，其中该反馈与在对请求的答复中提供的内容项相关；模块 SALM 也可以将模块 SFEM 用于这个目的。来自用户的典型反馈是用投票表示的（它可以直接与预测投票相比较）。在这种情况下，模块 SALM 可以通过模块 IRM 而将这类显性反馈存入储存库 IHR 中。非常有利的是，服务应用逻辑被设计成

让用户自主选择是否提供显性反馈。

作为替换，当服务应用逻辑没有提供来自用户的显性反馈时，这时可以对用户行为进行监视，以便从中得到隐性反馈（举例来说，该处理可以由模块 SFEM 执行）；举个例子，投票可以与用户在阅读新闻服务所提供的新闻项目中花费的时间相关联。在这种情况下，模块 IRM 可以将隐性反馈记录到储存库 IHR 中。

用户反馈的处理和记录（图 2 的步骤 207）既可以是显性的，也可以是隐性的，该处理可以在每次答复之后执行，或者在服务交互会话结束时执行。

应该指出的是，如果内容项包含了多个内容成分，那么反馈还可以涉及整个内容项；非常有利的是，作为替换或补充，该反馈可以与内容项中的每一个成分有关。举个例子，用户可以总体表达与电影相关的投票，或者为其视频成分和音频成分表达单独的投票。在这种情况下，单独的投票是作为交互历史而被记录的。

#### 处理用户简档

处理用户简档的活动包括创建（构建）和维护（例如更新）用户简档。在图 1 的架构中，用户简档被保存在储存库 UPR 中，并且用户简档构建器模块 PBM 被提供用于执行处理用户简档的活动。

非常有利的是，这个活动可以“脱机”执行，例如在用户交互数量较少的夜间执行。

根据本实施例，模块 PBM 执行下列步骤：

- 从储存库 IHR 中检索用户的交互历史（图 3 中的步骤 301）（其中该交互历史是完整的交互历史，或是与从最后一次用户简档更新到当前时间的时间范围相对应的部分交互历史）。该交互历史至少包括事件，并且通常包括一组事件。时间通常包括内容查询（对应于用户请求）、原始上下文元数据，以及依照该查询以及优先依照用户反馈提供的选定内容项集合；

- 根据交互历史中包含的信息来选择适合构建用户偏好预测模型的恰当的机器学习算法（图 3 中的步骤 302）；

- PBM 为交互历史中的每一个交互事件  $E_i$  产生一个“ $n+1$ ”维的特征矢量，该特征矢量通常是单个矢量  $V_i$ （图 3 中的步骤 303，步骤 304 和步骤 305），其中  $n$  是与内容元数据（预先指定和派生的）相关的特征数量以及与上下文元数据（原始和派生的）相关的特征数量的总和。

- 将选定的机器学习算法应用于先前步骤中产生的特征矢量（关联于事件  $E_i$ 、 $E_j$ 、 $E_k$ 、……的  $V_i$ 、 $V_j$ 、 $V_k$ ……），以便构建引入到用户简档中的新的预测模型（步骤 306）。机器学习算法一次仅仅能够处理一个单独的特征矢量，或者它们也可以设法一次处理一组特征矢量（与前述实例中相同，其中该模型是通过处理与十三个交互事件相对应的十三个矢量来产生的）；

-（优选地）对照预订接受判据（通常是“优于先前”类型的判据）来验证新构建的预测模型的性能，以此作为用户简档更新条件（步骤 307）。举个例子，一种有效的已知验证技术是“十等分交互验证（ten-fold cross-validation）”，该技术基于事件的十种不同划分（举例来说，90% 的事件用于构建模型，10% 的事件用于验证模型）。根据特定实施方式，验证可以集成在机器学习方法内，以及

- 通过在储存库 UPR 中用新模型替换先前模型来更新用户简档（步骤 308）。

与交互事件  $E_i$  相关的特征矢量  $V_i$  的生成处理可以依照下列步骤来执行：

- 检索原始内容元数据；

- 将原始上下文元数据（记录在交互历史中）发送到模块 MGM，请求从原始上下文元数据中产生派生上下文元数据；将原始上下文元数据和派生上下文元数据（从模块 MGM 中获取）编码到维度为  $p$  的上下文特征矢量  $V_{ix}$  中（在图 3 的流程图中，以上的两个步骤是用单个步骤 303 表示的）；

- 从储存库 CIR 中检索内容元数据（原创和派生的）；

- 将内容元数据编码到维度为  $m$  的内容特征矢量  $V_{ic}$  中，其中

$m+p=n$  (在图 3 中, 上述两个步骤被表示为单个步骤 304) ;

- 将内容特征矢量  $V_{ic}$  加入先前步骤中产生的上下文特征矢量  $V_{ix}$  中, 并且将作为目标特征的用户投票  $t$  添加到  $n+1$  维的单个特征矢量中,  $V_i = \langle V_{ix}, V_{ic}, t \rangle$  (图 3 中的步骤 305) ;

- 识别机器学习方法算法, 以及

- 将所述机器学习方法算法应用于所述特征矢量  $V_i$ , 以便获取用户偏好的预测模型。

应该指出的是, 即使先前不存在预测模型, 上文列举的步骤也是可以使用的, 换言之, 这些步骤不但可以用于更新用户简档, 而且还可以用于构建新的用户简档。在这种情况下, 举例来说, 如果为任何内容都假设一个正反馈, 那么将会使用一个虚构的用户模型。

如果用户表达了关于多成分内容项中的每一个成分的反馈, 那么模块 PBM 应该考虑这类更详细的反馈。

在以上描述中假设用户只具有一个用户简档。但是, 本发明还可以扩展到用户具有更多用户简档并且可以在其间切换的情形。举例来说, 当上下文元数据不足以精确描述交互上下文时, 例如当终端很难自动确定用户在家还是在办公室 (除非用户在交互设备中设置了环境选项) 时, 这种处理将会是非常有利的。

用户简档选择可以在交互会话开始时进行, 并且该选择通常包括多个请求以及具有隐性或显性反馈的相应答复。

作为替换, 用户简档选择也可以在反馈操作的每一个时刻同时进行。

举例来说, 假设用户在电影浏览应用中发现了他很喜欢的一部恐怖电影。该用户给出了关于该项目的很高的第一投票, 由此规定该第一投票参考的是“个人”简档。由于恐怖电影对他的孩子并不是很好, 因此, 它还为这部电影给出了很低的第二投票, 由此这一次规定了第二投票参考的是“家庭”简档。作为替换, 用户可以设置其简档之一作为当前简档; 用户给出的投票将参考所设定的简档。当用户请求关于电影的排序或推荐时, 他需要指定给出所述推荐所要依照的简档。

在提供多个用户简档的实施例中，图 1 的模块需要考虑这种多重性。模块 IRM 还需要在储存库 IHR 中记录关于用户简档的信息。模块 PBM 需要选择所要更新的正确用户简档。模块 MMM 需要选择和使用正确的用户简档来产生内容项排序。

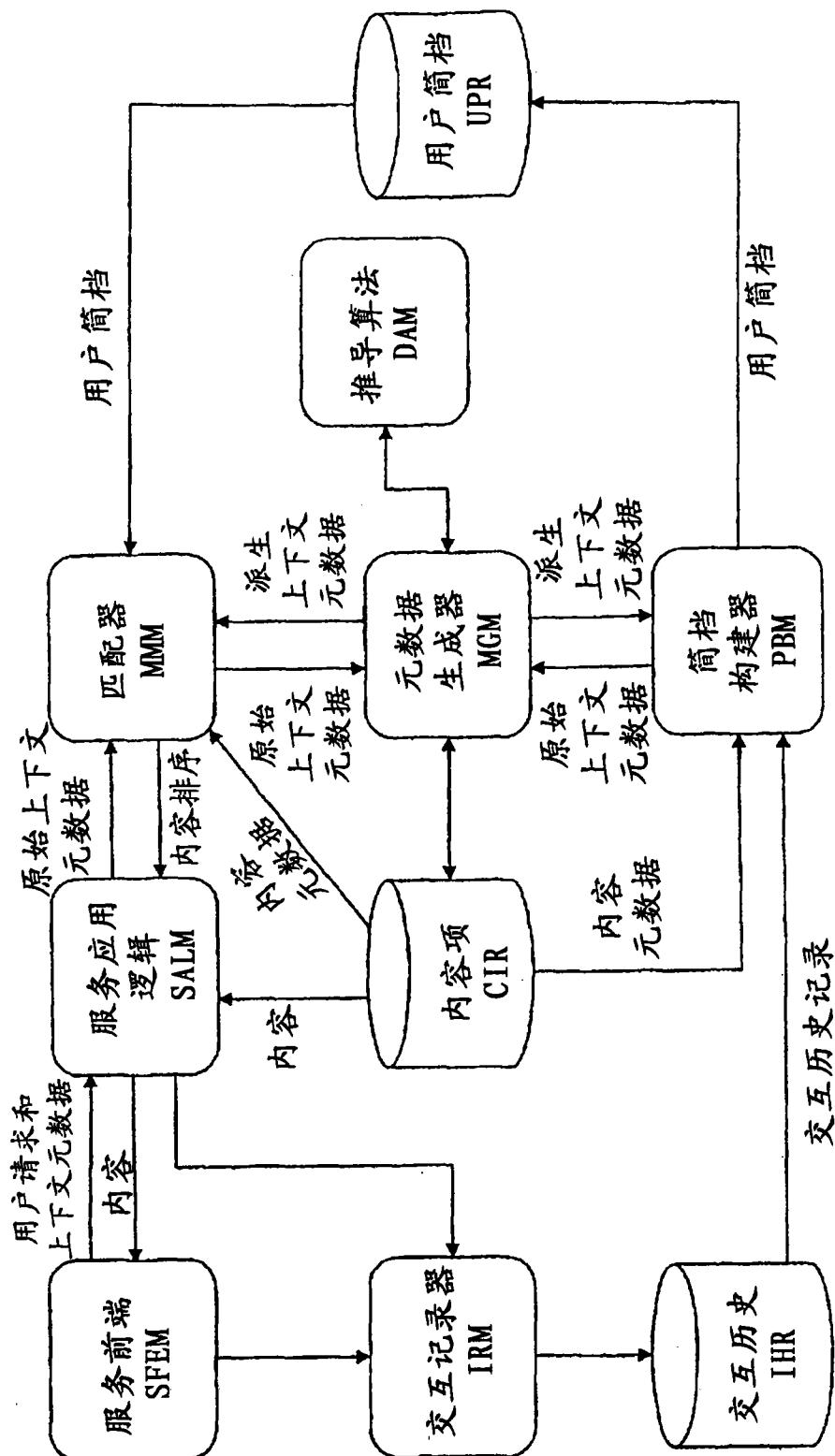


图 1

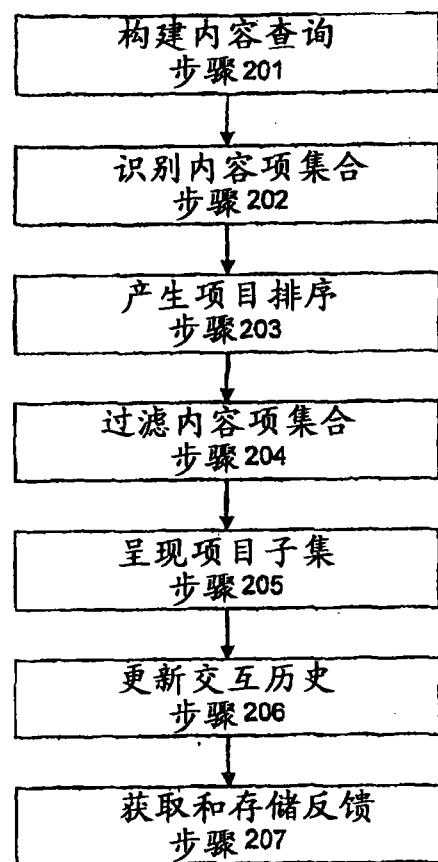


图 2

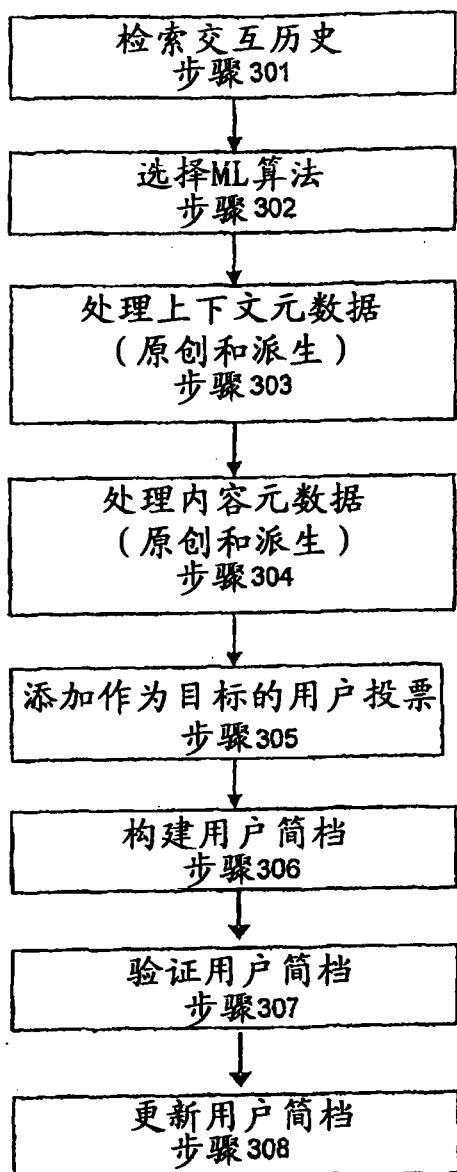


图 3

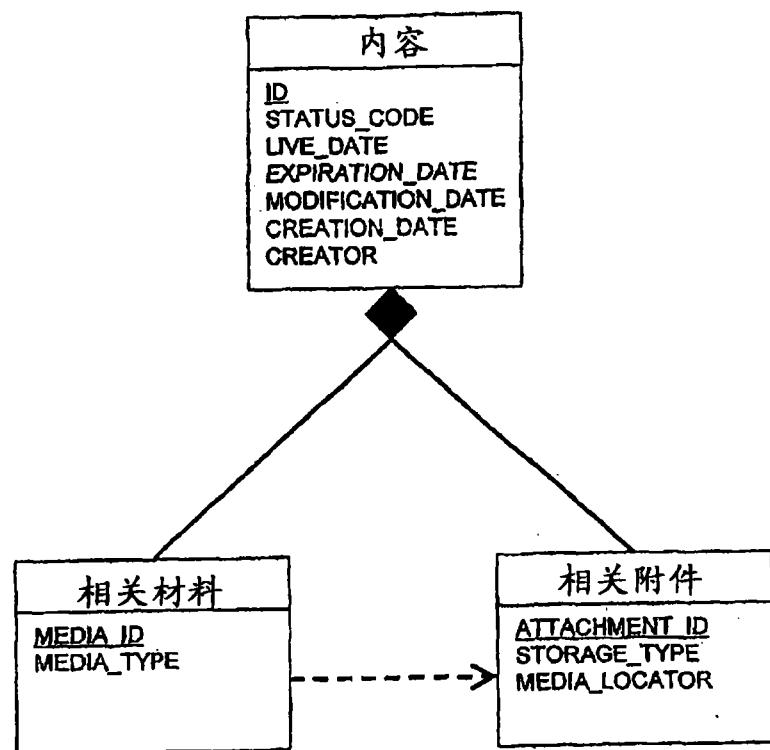


图 4