

Microcosm——开放的超媒体模型

Microcosm——An Open Hypermedia Model

谢杰华

Xie Jiehua

(广西计算中心广西软件新技术实验室 南宁市星湖路 32 号 530022)
(Guangxi New Software Technology Lab., Guangxi Computing Center,
32 Xinghu Road, Nanning, Guangxi, 530022)

摘要 给出了开放超媒体系所应具有的特征,且对开放的超媒体模型 Microcosm 系统进行了分析,研究这一模型的优点和缺点。最后还对将来的应用系统所应具有的功能提出了看法。

关键词 开放 超媒体 Microcosm

Abstract Gives a definition for an open hypermedia system. It also analyses the enhanced Microcosm model for open hypermedia, investigates the advantages and the shortcomings of this model. Finally, it gives some views for the functions which the further applications should be have.

Key words open, hypermedia, Microcosm

1 引言

从 1989 年起,南安普敦大学开始设计开放的超媒体模型,并开发了一个叫 Microcosm 的系统。这个系统最初是让研究组在上面实验各种多媒体领域的想法。如今,它被用做为集成多媒体应用研究的结果和教学材料,系统开放的观点是其迷人之处。最终的目标是在开放系统提供一个完整的超媒体环境。

许多人使用 PC 或工作站作为他们日常工作的工具,他们用这些机器作些什么呢,总的来说,他们只是用少量的软件来处理数据,其它的时间,就把这台机器作为一个电子记录本。为他们准备文档,发送和接收电子邮件。保存接收到的文档,以及保留工作中涉及的记录,所有这些活动都产生大量数据并保留在机器中,他们如何查阅这些数据呢?大多数用户通过使用文件目录结构作为查找数据的主要方法。

如果超媒体技术能有效地链接和导航信息,为什么用户不使用超媒体系统来集成他的数据呢? Malcolm et al^[1]认为,因为当前的系统并没有足够优越的性能。为了在一个超媒体系统中利用一个文档,首先要求先把文档加入系统,当链加入系统时,表示链的

数据将以某种形式标注并存入文件中,这样,数据就在一个封闭的系统中,其它系统再也无法处理这些数据了。数据就象被以一种私有形式存入超媒体系统一样。如果系统允许用户再次输入数据,它将会失掉标注信息。而在日常中,任何数据都是动态的,说服用户把其数据锁入超媒体系统好象不太可能,除非用户相当了解超媒体系统提供的扩展功能。

当超媒体系统关闭时,它可能继续限制那些相关的静态存贮数据。下一代超媒体必须作为一个操作系统的功能展现给用户。它应该具有方便增加信息联系和导航的功能;需要很少用户的干涉,且不减少任何以前有用的功能;它们必须能够在其它信息处理工具提供的数据之间链接,且必须有自动产生和维护链的工具。Sun 公司的 Link Service 和 Microcosm 就是新一代超媒体系统的例子。

2 用户想从一个系统那里得到什么

· 一个集成数据、工具和服务的合适的环境封闭系统把数据以本身特有的形式保存,它就必须为访问和处理这些数据提供工具。例如 Intermedia 就提供了诸如 Inter Text 和 Inter Draw 等应用。然而,这种方法是一种封闭式的,用户关于选择文本编辑和绘画软件之间的争论几乎上升为信仰的争论。用户不想被限

制在一个特定的软件包里。在任何情况下, 想要预测所有用户将需求的功能是不太可能的。超媒体系统必须允许用户在他们所选择的软件中建立数据, 并且在数据之间加链, 或在这些应用上加活动锚。

- 一个独立平台或分布交叉平台

用户将希望把一台机器的数据和应用链接到另一台机器的数据和应用。他们还希望这些操作是透明的, 在大多数情况中, 在分布式操作系统控制下是很难实现这些操作的。然而, 超媒体功能的分布必须跨越操作系统, 且信息链必须方便地跨越硬件平台。

- 用户能够容易查找、更新、注释和交换信息和系统

如果用户打算采用超媒体系统, 那么系统必须加上导航机制, 以辅助那些在操作环境提供的功能, 这样, 就易于用户使用。用户可以用普通的工具来改变数据, 增加新的信息、注释。既可以私有, 也可公共, 当用户改变公共信息时, 系统必须给予揭示。系统必须具有公共和私有工作空间的概念, 还可以在空间之间移动信息 (包括信息链)。

- 所有形式的数据和媒体都以一个概念相似的方式处理

如果在一个软件里建立链的过程与另一个软件的界面不同, 那么用户将不易于学习和利用所有的功能和媒体。

3 什么叫开放的系统

一个开放的系统至少应该具有如下特性:

- 不在数据上强加任何标志来防止非本系统的访问。

• 系统可以集成在宿主操作系统上运行的任何工具, 工具所产生的数据不需要另加特定的值。并不是该开放系统的一部分, 可以在整个操作系统中使用, 而且, 系统也可以使用外部数据。

• 系统的数据和进程的分布可以跨越网络和硬件平台。

- 系统在读者与作者之间没有人造的分别。
- 系统易于加上新的功能, 也就是说, 易于插入新程序。

符合这些定义的系统将会长期地满足用户需求。

4 开放超媒体的一个例子——Microcosm

4.1 基本模型

Microcosm 运行在 MS-Windows 3.1 上, 且正在 Apple Macintosh 上开发。在 Unix 系统上的 X-Window 上也有其版本。这一模型最先在 Fountain et al^[2] 里描述, 为了保证系统的开放性, 已做了根本的改变和扩展。Microcosm 最容易被理解为一个由操作系统提供的自由通讯补充功能。Microcosm 允许用户使用各种技术在大量的多媒体信息中导航, 没有标志被强加在信息上。所有的数据都可自由访问、编辑。应用系统还可以建数据。所有相关链接信息被存在 Microcosm 所指定的链库中。

在 Microcosm 中, 用户与浏览器相互对话。浏览器是可以显示数据的应用程序。执行动作的消息从浏览器发送到 Microcosm, 然后, 通过一个系统过滤器链迅速的发送。(如图 1 所示)

各个过滤器通过阻挡、放行或在放行前改变这些消息, 根据消息的内容, 一些过滤器可以在链中加入新消息。最后, 这些消息将从过滤器链中出来, 到达调度器, 这时调度器将检验消息, 看看是否包括任何可用的动作 (如沿着链向下等), 以及决定它是否给用户提供这些动作。

特别重要的过滤器是链库, 它包含所有关于可用的链的信息。当一个消息到达链库, 要查找一条导航链时, 这个进程就会查找数据文件的链源。如果找到, 就通过链调度器返回详细的链目标。其它过滤器的功

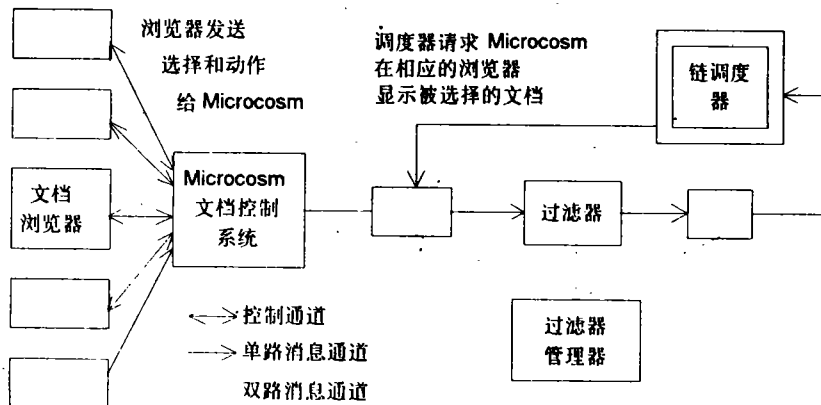


图 1 Microcosm 模型

Fig. 1 The microcosm model

能包括辅助导航（如历史机制）和处理动态链计算。

4.1.1 链 Microcosm 模型允许用户通过大量不同层的链机制导航。

- 特殊链，在源文档中一个特定对象上的特定点引出的链，它连到目标文档的一个特定的对象。

- 局部链，在特定文档中一个特定对象上任何点引出的链，它并连到一个目标文档的一个特定的对象。

- 普通链，在任何文档中一个特定对象上任何点引出的链，它连到一个目标文档的一个特定的对象。

因为模型允许文档到文档的连接，这些链可以从一个特定文档（或过程）指向另一个文档。

以上所述是静态目标链，因为目标是固定的。上述情况的目标可以是一个文档的开始，也可以是一个处理的过程而不仅是简单的数据显示。有动态源锚的局部链和普通链是 Microcosm 的一个特别强有力的特征。在 Fountain et al.^[2]1990 里有详细的描述。它允许一个目标只被固定一次，而后继的链可以从一个适当的源对象链接到新建的数据或文件上。除此外，有其它链：

- 文本检索链，当有检索请求时，动态计算这些链。有两种方法可以到达这种链。第一种是使用 grep 过滤器。它先在一些预先定义的文本集中选择匹配的文本，然后把所有可能的文档名字返回给到链调度器。这种方法相当慢。第二种方法是，建立所有以前系统使用的文档的一个反向索引。使用标准的信息相似检索算法来查找与被选的词相匹配的文档。并提供用户最佳的匹配链，这种方法在很短的时间里产生高质量的匹配。

- 相关链接，如果一个文档集已经经过预索引了，那么，就可以用根据词汇集的相似性将文档。在这种情况下，就可以让用户在相同的束里沿着链走到其它的文档。

以上所描述的机制在标准的文件目录结构导航方法上给新信息的查找提供了一个丰富的方法集。还应该强调的是，开放模型易于描述和实施新的导航方法。

4.1.2 消息和保持系统开放 为了确保系统尽可能的灵活，Microcosm 采用一种节标志的 ASCII 码消息形式，任何浏览器或过滤器都可以在消息中引入它喜欢的任何标志和数据到消息中，任何过滤器都只响应它知道的标志而忽视其它的。Windows 版的 Microcosm 利用 DDE 技术在过程之间传送消息，Macintosh 系统使用苹果事件，而 Unix 版本则用 Sockets 来传递消息。

写一个新的过滤器是一个简单的任务，任何 Windows 程序员都可以做，无需对 Microcosm 内部作特殊的了解。其中要写的一个程序是为所需要标志分析接收到的消息，并产生适当的动作，然后插入到一个拥有 DDE 通道且了解所有通道的外壳。这样一个新的过滤器就完成了。这个灵活的制造方法使程序员可以很容易地改变系统的功能。

4.1.3 浏览器 Microcosm 的浏览器有这样一个任务，允许用户在显示数据中作选择。然后，从一个菜选择单一个动作。如沿链浏览，建链或计算动态链。这就意味着，一个浏览器必然是一个与 Microcosm 相关的程序，这就会与任何以显示数据作为浏览器的程序的要求冲突。

为了解决这个问题。Microcosm 里使用了三类浏览器。

- 完全相关的 Microcosm 浏览器，它有一个如上所描述的动作菜单，它包含相应的消息并且用 DDE 通道直接通信，在 Microcosm 中有十个这样的浏览器，用来处理环境中共同的数据形式。诸如：文本、位图、视频、音频、Windows 元文件以及丰富混合文本，使用这些浏览器的优点是它可能有相当清楚的功能，而其它的程序是没有的，一个 Microcosm 浏览器可用按钮来显示活动区域。这些与对象的选择或动作相结合，它们有时是以高亮度显示的。任何特殊链源可以是一个按钮，而有关“什么该高亮度显示”的信息存在链库中。浏览器装入数据时需要这些信息。Microcosm 浏览器还可以通过聚焦请求在数据的任何特定点上起动作。

- 部分相关浏览器。这是来自外部源的应用。许多软件如 Word for win, Toolbook 以及 Superbase 有程序级的可能访问 DDE。在这些应用中，相当直接的写出必要的代码，产生一个动作菜单，把所做的选择和动作封装成信息包发送给 DDE 通道。修改这些应用的过程与重写这些应用是大不相同的。任何合法用户的软件都可以理解它，并给出相应的指南。

- 无关的浏览器。在最坏的情况里，不能在浏览器里建任何形式的动作菜单。也没有 DDE 访问。Microcosm 引入了剪接板链的概念。用户在通常方式下从应用中作一个选择。然后，拷贝到剪接板。可以从 Microcosm 的图标选择一个动作菜单，而 Microcosm 将给剪接板的内容和动作选择作出响应，然后把它们封装进一个消息。这种方法的一个改进是允许用户命令 Microcosm “管理剪接板”，无论何时剪接板的内容改变了，Microcosm 将自动地将一个预选动作的新内容（如向下链）封装起来。

动作菜单的界面对各种不同应用来说是不一样的,并且,因为部分相关浏览器和无关浏览器通常不能提供被选择数据在文档中精确位置的信息,它不可能从这个文档中提供特定的链,而局部链和普通链可以从任何应用中给用户提提供超媒体功能。

4.1.4 过滤器 在 Microcosm 里所有的链接功能都由过滤器过程提供(见图1)。当用户修改文档编辑时,就产生消息,并把它发送给过滤器管理系统(FMS),这个进程管理消息传播到所有当前的活动过滤器。当前的 FMS 在一个简单的链结构中组织有用的过滤器,如图1所示。当 FMS 接到一个消息,就把它传给链中的第一个过滤器,当消息通过过滤器返回到 FMS 时, FMS 又把它传给链中的下一个过滤器,当一个过滤器接到一个消息时,就检查它,决定是否必须处理这个消息,并可以产生动作或把消息返回 FMS。当消息到达链的末端,它们就返回主 Microcosm 控制过程,并由它作任何必要的动作。

通过小的独立过程的方法来实施系统的超过媒体功能。它们可以动态安装、重移动或重排列以改变系统的行为。还很容易通过建一个新的过滤器过程来增强有用的功能,而不需要改变现有系统的组成。一个典型的过滤器是 Linkbase,它提供一个可用链的数据库,其它例子允许用户建立链或提供导航功能,例如,在 4.2 节描述的访问机制使用过滤器来监视和处理系统活动。

4.2 信息访问层

这个超媒体的开放模型,给用户提提供许多信息导航的不同方式,这里它们按递增的顺序表达。

4.2.1 浏览文档层 最近对 Microcosm 使用的研究,发现尽管使用了有许多按钮的文档链集,用户仍然倾向于使用目录结构作为信息访问的重要方法。由于这个原因,引入了一个高级的文件浏览器来允许用户贮存或浏览文件属性,以辅助导航和文档选择。这些属性根据用户定义的关键字决定。并且,当浏览目录结构或把文档作为一个潜在的链目标来提供时,这些属性被用户作为计算一个特定文档有用性的方法来访问。虚拟笔记本(The Virtual Notebook)^[3]系统使用节点属性作为主要的导航方法。但在这种情况下,属性从一个固定的集里被选择。在将来的工作里,我们希望利用这些属性来计算检索链以及自动生成普通链。

4.2.2 手工建链 有特殊、局部和普通三种链。特殊链与那些在大多数第二代的超媒体系统所提供的相似。局部和普通链无需太多创作努力,并可以从链无关的应用中制作链。

4.2.3 计算链 这些链不需要手工努力,其范围从简单的字符串查找技术到全文检索技术。一旦链被计算,它可能立即沿着链向下走,或永久的合并进 Linkbase。在这种情况下,其结果与手工建链无法区别。

4.2.4 历史 许多操作系统都能通过存贮所有在命令行中已发过的命令的清单,来保留命令历史。与此相同, Microcosm 也能通过维护由链调度器发送的所有消息的清单来保留所有链的历史。这样就可能在其装载的文件或过程的背后浏览这些消息。然后重新发送选择消息。

4.2.5 模仿(Mimics) 当前许多超媒体应用都是示教的形式。在这种情况下,作者希望给用户提提供固定的穿越数据空间的路径。在 Microcosm 中,这一功能以 Mimics 的形式提供,这形成了一个文档集的浏览。用户可以自由的从 Mimic 走开, Mimics 在这里是作为可选的路径,而不是大多数其它的浏览系统那样是一个固定的浏览路径。Mimic 形成一类 Nielsen 所描述的“超链”。Mimics 可用多种方式产生。例如,从用户过去的历史,利用一个基于语言的文本或一个基于构造方法的形式。

5 开放系统的优点

开放的超媒体模型 Microcosm 战胜了许多其它封闭超媒体系统所面临的技术问题。其中主要问题之一,就是处理大系统。一个包含上万篇文档的系统如果仅使用目录结构作为辅助导航是不行的。作为从文档自身分离链服务的结果,开放的超媒体系统允许用户使用链的所有导航辅助。例如,给链加上附加值等。

将链与文档分离的另一个优点是,数据可以记录产生它的原应用。用户可以自由连续的从系统所产生的人工结构操作数据。一个更进一步的优点就是在任何时候,同一地点可能有一个以上的链库。Microcosm 的一个共同的结构是有一个包含文本集上链集的链库。且这个链库是由原作者定义的。每个用户都有自己的链库,用来存贮他们自己的链和注释。允许共享工作空间的概念可能会扩展。允许访问这些共享链库和注释是操作系统的工作。

将链与文档分离的最后一个优点是增强处理链的功能。它允许用户通过诸如从其它链库合并链的过程来处理链库,删去被移动的全局参考文件以及改变链的范围。将来的程序将会利用用户定义的链属性。

通过利用诸如生成链和计算链等特征,在开放的超媒体系统中,作者的作用可能会减少。对于开放系

统, 这些特征并不特别。但扩展模式的能力就很特殊所以象计算链等的特征要依靠系统的开放。

如上所述。系统支持任何相同的文件系统: 这就是说, 任何系统上, 所有的文件通过用户的工作站可以访问和处理。在 Microcosm 中, 所有的文件要么驻留在工作站上, 要么在网络设备上。如他们在网络上可用, 程序将在运行前装入工作站。没有什么障碍阻止过滤器在不同的平台上运行。并且, 数据可以在各机器之间以各自的结构传送。没有障碍阻止节点在不同的平台上存在。限制超媒体功能的分布存在于操作系统设计中, 而不是超媒体本身的设计, 当前可以在 Unix 上的基于 X-Window 的文本浏览器使用 Microcosm 的链库, 也可以在运行远程 PC 工作站上的过滤器来导航驻留在 Unix 或 PC 上的文本文件。

最后, 通过定义一个开放的协议。开放的超媒体系统可以增加信息的访问能力。并且可以形成超文本标准。当标准成熟并可以在不同系统间相互作用时, 通过使用相应的交换标准, 链信息就可以在不同系统之间迁移。

6 开放系统的问题

与 Sun 公司的 Link Service 相同, 开放模式的超媒体也有三个主要困难。

6.1 锚的尺寸和位置

在大多数应用中, 超媒体是目录系统的一个扩展。作为导航巨大信息空间的方法, 这个覆盖在正确目标上的颗粒层正是用户希望的。他们可以很容易地从那些由应用本身所提供的任何浏览器所设置的文档或数据继续导航。这样, 作者就产生了一个更好的链, 且通常是静态的超媒体产品。诸如示教。它们通常总是使用提供链到特定点的 Microcosm 相关浏览器。

改善这一状况的可能性有很多。最简单适用的是, 利用系统宏定义。在一个 OEM 软件里构造链, 可以产生一个在数据里搜索相关词条的宏定义。当沿着链向下时, 软件就随着相应的数据集启动。然后, 宏定义就重演了。

将链库与文档分离的一个将来的问题是, 不能立即清楚什么链合适于用户。在封闭系统中, 常有某种形式可见的线索。如黑体字指示一个特定的对象是“活的”可以用鼠标选择。Microcosm 允许作者在按钮中作特殊的链, 且链相关的浏览器将会为任何按钮查询链库。在某种可见的方法下显示对象。然而, 在这种方法里, 为所有普通链作检查是不可行的。以下给出了两种方法解决这个问题。

第一种方法涉及在内存里控制全集链的索引。当被请求查找所有的链时, 浏览器就会匹配当前窗口反向索引的内容。并且在当前窗口里显示找到的所有的链。

第二种方法涉及用户选择文本区域和请求显示链。选择文本被分裂成词和词对。并重组成下一个链消息。链的查找结果将在相应的链窗口里显示。

第一个方法对用户很友好, 但仅能被 Microcosm 相关文本浏览器利用, 而第二种方法却具有让所有浏览器使用的优点。

6.2 编辑问题

因得链与节点分离, 所以文档编辑就会有问题。在 Microcosm 中, 根据文档中绝对位置记录源链和目标锚。如果在一个链前插入或删除对象, 那么链库里的信息将过时。如果链是文档到文档的, 而不是固定在特定文档里的某一处, 就不会有问题。

如果一个链目标文件被移动或删除, 情况就很糟。这样链就会悬空。在一个分布式文件系统里这就是个问题。因为在这种系统里, 可用文件可能依靠网络访问, 或者在用户私有空间里链可以指向文件。

6.3 版本问题

正如其它问题一样。忽视版本问题将会冒风险。这完全是操作系统内部的问题。毕竟, Unix 提供了诸如 SCCS 之类的版本控制系统。在 PC 机上也有实施得很好的 RCS。事实上, 如果实施了保持旧版文档作为更新的参考, 那么这个操作系统提供的功能就很友善。不幸的是, 超媒体提出了新的问题。如果文档被编辑过, 那么链到旧版还是新版呢。还需保持链库的版本吗? 在开放的超媒体系统, 版本问题与编辑问题相互缠绕。

6.4 寻找解决的方法

当前, Microcosm 使用的解决办法如下。所有对于源和目标文档的链都包含一个数据标记。它象被操作系统控制一样, 与文档自己的日期匹配。为了提供工作空间的某些概念, 文档管理系统 (DMS) 保留了属性, 用来陈述用户当前在哪个应用或应用的某个特定文档中。另一种文档属性是文档的别名。使用别名可以更好地描述文档, 还可以移动文件。在 DMS 中, 只需更新别名, 而不是更改所有在链库里的参考文档, 在这些文档中, 有些甚至可能没有联机。这种解决办法帮助化小了由于早期系统的组织结构的局限性, 而在大的舞台上可能变得不适应的问题。

无论何时, 当装入一个文档时, 都检查文档里所有特殊链。如果所有这些链中有一个源文档的日期比文档本身日期早, 那么就警告用户。

无论何时,当一条链指向一个文档时,就把目标文档的日期与链日期对照检查。同样,如果它们不匹配,被警告用户。或者,如果目标文档不存在了,也警告用户。

对于文档,存在一个链相关编辑器。当装入文档时,就会查找所有当前可用链库里的文档参考。并且在编辑完后,更新所有这些链锚。在编辑期间,文档对其它用户来说被加上了锁。

这一解决办法使用户留心任何不相容情况的发生。但它只是给用户发警告,并没有真正解决问题。确保这些不相容情况不发生,显然是最好的。因此,作者应该尽量避免使用特殊链。使用文档到文档链或普通/局部文档链很少发生不相容情况。所有使用特定应用的链库应保持联机。这样链相关编辑器可以更新它们。

如上所说,保持一个开放超媒体的一致性并非微不足道的任务。保持这样一个系统的一致性有大量的先决条件。

这些先决条件是:

- 不能编辑链服务器不关心的文档。
- 文档管理系统关心所有受编辑影响的链库,并且为了适应变化,可以访问这些链库。
- 有一个用于从旧版文档移动特殊链锚到新版的算法。
- 文档和链库的版本应自动地维护。通过一条链漫游文档,用户可以走向先前版本的文档。但大多数当前版本的文档必须适用于外部的系统版本,可以让外部应用访问和处理。

7 结论

本文表述了超媒体作为一个开放系统,为信息在一个分布式的异种环境里的集成和处理提供的服务。

并给用户丰富多彩的方法去导航这些信息。所描述的系统能够实施所有有一个简单的第二代超媒体系统可以料想的任务。但保持系统开放要付出一定的速度代价。并且,为了保持系统一致,开放系统引入了关于把链锚与数据捆绑在一起的新问题。这里讨论了两种解决办法。但是,很显然,将来工作所需要的,尤其象诸如信息网络规模的领域正在增长。更进一步,Microsoft模型所研究的一些仅关于文档的技术也包括文本。还不清楚这些技术如何能够扩展来处理一般的多媒体,也不知道这样的企图是否适当。

真正开放的超媒体系统是一个相关的新概念。从长远来看,操作系统的卖主将提供越来越多地有用链和对象管理系统,把它们作为操作系统的集成部分,并且还将这写一些,应用软件以理解这些服务。

参考文献

- 1 Li K, Wendy Hall and Hugh Davis, Hypermedia Links and Information Retrieval. The Proceedings of the 14th British Computer Society Research Colloquium on Information Retrieval, Lancaster University, 1992.
- 2 Fountain A M., Wendy Hull, Ian Heath and Hubh C. Davis, MICROCODM: An Open Model for Hypermedia With Dynamic Linking, in: A. Rizk, N. Streitz and J. Andre, eds., Hypertext: Concepts, Systems and Applications. The Proceedings of The European Conference on Hypertext, Cambridge University Press, 1990.
- 3 FRANK m. Shipman, R. Jesse Chaney and G. Anthony Gorry, Distributed Hypertext for Collaborative Research: The Virtual Notebook system, Hypertext, 1989 Proceedings, ACM Press 1989.
- 4 Davis H., Hall W, Heath I.. Towards An Integrated Information Environment With Open Hypermedia Systems. Hypertext '92 Proceedings. ACM press 1992.