

Hypertext: 一种信息管理的新方法 *

潘玉平 张荣肖 张 兵(中国科学院计算中心)

摘要:本文讲述了超文本系统的概念及其发展过程,在介绍了国外几个典型超文本系统后,给出了超文本系统结构的一个理论模型及实现原则。

八十年代中后期,由于计算机软硬件技术的发展,特别是大容量光盘技术、图形、图像处理、数据压缩、用户接口设计等技术的推动,一种管理多种媒体信息的技术——Hypertext 得到了异乎寻常的发展。本文讲述了超文本系统的概念及其发展过程,在介绍了国外几个典型超文本系统后,给出了超文本系统结构的一个理论模型及实现原则。

一、超文本系统的概念及发展

1. 什么是超文本和超文本系统

超文本,简单地说是一种以非线性方式建立和表示离散信息间关系的技术,是一种存储和管理信息的方法。系统的基本成份是结点(Node)和链(Link),信息存放在结点中,相关联的结点用链连接起来。因此,超文本就是一个把关联结点用链连接起来构成的相互交叉的信息网络(见图一)。其结点中可存放的信息不仅仅是文本数据,也可是图形、图像、声音、视频等其它形式的数据,不同种类的数据构成了多种类型的结点。超文本结点之间的链是有向的,既可是单向的,也可是双向的。对超文本中信息的浏览可沿着链由一个结点转到另一个结点。这种跳跃式的浏览,正是超文本区别于其它系统的显著特征。超文本系统是一种由软件与硬件构成的配置,它向用户显示并允许用户管理存取其中的信息。

要对超文本的概念有一个更加明确的认识,就应首先了解它与传统的文本到底有什么不同。

传统的文本文件,比如杂志上的一篇文章,一本书,其物理结构与逻辑结构非常相似,物理上,一篇文章是线性的,就是由词顺序相连成行、页,逻辑上,它们也是线性的,由词组成句,句组成段,段组成节等。一般文章的逻辑结构是有层次的,先是摘要,然后是介绍,再就是第一节、第二节等,最后是结尾,这种线性结构将使得读者只能从头至尾顺序地

阅读。

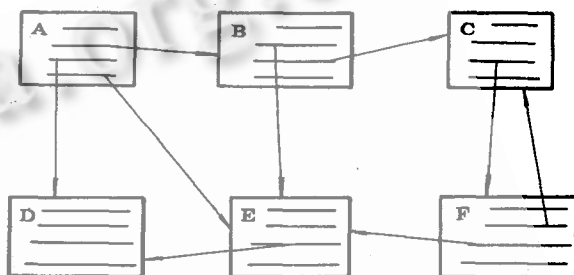


图 1 一个简单的超文本结构图

也有些传统文本文件,比如百科全书、字典、产品目录等,逻辑结构与物理结构有所不同。物理上,这些文件由一些独立单元顺序堆积而成,逻辑上,就要复杂一些,这些独立单元的堆积顺序需要根据某一标准而定,象字典就是按条目的词典序排列。读者在阅读时,比如百科全书,要先确定感兴趣的条目,然后顺序阅读该条目,在阅读过程中,可能会遇到“参阅”这样的情况,为此,读者又须确定合适的卷,合适的条文,虽然编排结构是顺序的,但是读者的阅读路径是相互交叉的网络。

超文本与文本不同,其结构是非线性的。从某一结点出发,根据不同的阅读需要,可以到达不同的结点。结点之间的链结构为交叉引用提供了很大的灵活性。在这里,我们可以将百科全书的一个条目,一幅图等理解成一个结点,条目中出现的“参阅”也就成了到需参阅条目的链。这样在顺序的基础上,用链将关联结点连接起来也就构成了整个百科全书。

超文本提供了表示数据信息的一种新方案和非线性直接存取数据的新方法,各种类型的信息结点与不同类型的链相组合,就可构成复杂而灵活的超文本系统。

2. 超文本系统的发展过程

超文本的概念最早出现在 1945 年, V. Bush 描述的一种称之为 memex 的机器,这种机器能够用于浏览、翻

* 本文得到了国家自然科学基金的支持

阅大量的联机文本和图形,它存储了大量个人的照片、笔记、手稿,以及图书馆的文档资料。这种 memex 机器有若干个窗口,它可以帮助用户在任两个窗口之间建立一种可以命名的链。借助于这种链,人们可以翻阅资料,对不同的资料做比较,也可在机器上作批注,笔记,组织自己的文章。这种沿着链进行的查找方式不同于一般使用文件时的顺序查找,而非常象人的思维,具有跳跃性,能从一个概念联想到另一个概念。

V.Bush 的这一想法在当时由于技术原因无法实现,但现在人们仍然尊 V.Bush 为超文本之父。

六十年代初,斯坦福研究所的 D.Engelbart 开始了在计算机上实现由 V.Bush 提出的 memex 的实验。1968 年,他建立了一个能反映超文本特征的系统,称为 NLS(ON Line System)。NLS 由一些文件组成,每一个文件划分成不大于 3000 单词的“段”,各段之间可以用层次或非层次的方式进行连接,实现非线性查找。现在 NLS 已商品化,称为 AUGMENT。

六十年代在超文本研究方面另一个有影响的人物是 T. Nelson,他首次提出并正式使用 hypertext 这一术语,并发表了许多这方面的文章。他有关超文本的许多思想已被今天的系统采用。七十年代初,他开始自己研制超文本系统,称为 Xanadu。在 Xanadu 系统里,他试图存储全世界的文学资料以建立一个“全球图书馆”。并使用了大量的链来方便这些资料的使用。1988 年, Xanadu 系统由 Auto-desk 软件公司接手并商品化。

现在,超文本已从实验室走向了市场,也已应用在不同领域,并取得了相应的成果。目前出现的超文本系统都是面向某一应用的,都有自己的应用范围。而在现实世界中,有的信息适合用超文本方法来处理,比如词典、百科全书等;有的信息就不适合用超文本方法,象长篇小说。那么到底什么样的信息适宜于用超文本方法来处理呢?

Ben Shneiderman 给出了应用超文本的“三金律”:

- (1)信息由大量的片断信息组成。
- (2)片断之间存有相互关系。
- (3)用户每次浏览只对其中少量信息感兴趣。

除了这三条定律外,还有一条:信息的存储管理及应用必须在计算机上进行。现实生活中的许多方面都可应用超文本方法。如计算机系统的联机手册,软件工程和逆向软件工程的文档管理,各种设备的电子维修手册,电

子词典,百科全书及工具书,此外还在贸易展示,产品目录介绍,广告业及审计方面均得到广泛的应用。

二、几个典型超文本系统的分析

目前世界上已出现了许多超文本系统的原型或产品,下面分析了几个典型超文本系统的设计及实现方法。

1.NoteCards

NoteCards [1] 是美国智能系统实验室于 1987 年开发的 Hypertext 系统,最初运行在 Xerox Lisp 机上,现在已可在一般工作站上运行。这个系统最早是为信息分析处理设计的,包括阅读、分类、解释等功能。

NoteCards 包含四种基本单元:卡片(notecard),链,卡片叠(FileBoxes)和浏览卡(browser card),见图 2。

.卡片:卡片就是超文本中的结点,在屏幕上可以作为一个窗口打开,用户在需要时可以打开很多卡片。由于卡片中存储数据的不同使得卡片类型也各不相同,NoteCards 有至少 50 种类型的卡片,最简单的卡片仅含有文本和图片。

.链: NoteCards 的链是有向有类型的链,用以连接有关系的卡片。链的类型由用户根据原结点和目标结点的关系来定。比如:在一律师使用的 Notecards 中,可以定义一种链来表示连接到支持自己的法庭判决,另一种链表示连接到反对自己的法庭判决。

.浏览卡:是卡片和链的结构简图。链的不同类型用不同的线来表示。用户可在浏览卡上直接操作建立卡片之间联系。也可通过按键激活一张卡片。

.卡片叠:用来存放有关的多张卡片。Notecards 中的每张卡片都要存在于一个卡片叠中,事实上,卡片叠是一个特殊的卡片,因而卡片叠可以包含卡片叠,也可通过链与其它卡片连接。

2.KMS

KMS [4] (Knowledge Management System)是最早的实用的超文本系统,其前身是美国卡内基-梅隆大学 1972 年开始开发的 ZOG 系统,它可在 SUN 和 Apollo 工作站上运行。最初 KMS 正如其名,是为知识管理设计的,但现在已有了更广泛的应用。

KMS 的数据结构非常简单。结点类型只有一种:框架(frame),因此 KMS 的库文件就是一个框架的集合。框架尺寸有两种:整个屏幕和半个屏幕。框架由多项内容组成:框架名、标题、框架体、链接项、注解项和命令

行。其中既可存放文本,也可存放图形、图像,且文本、图形和图像可随意组合。对于含有大量信息的文档、图书等资料,用户可以先将它们划分为许多框架,然后用这些框架之间的层次式链,按照内容次序有机的组织起来。当在屏幕上得到一帧框架时,用户可以通过鼠标器选择用作链源的字符串、图形或图像,从而得到链另一端的框架。虽然 KMS 不提供翻屏功能,但它对一帧框架的存取速度非常快,可在 0.15 秒内完成。

KMS 的框架结构如图 3。

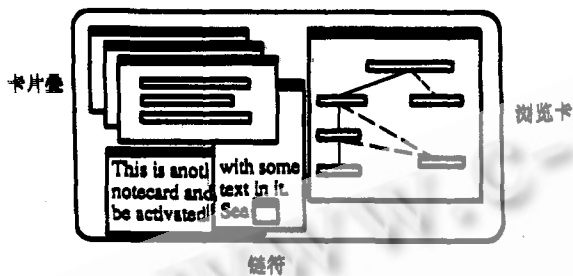


图 2 NoteCards 的四种基本单元

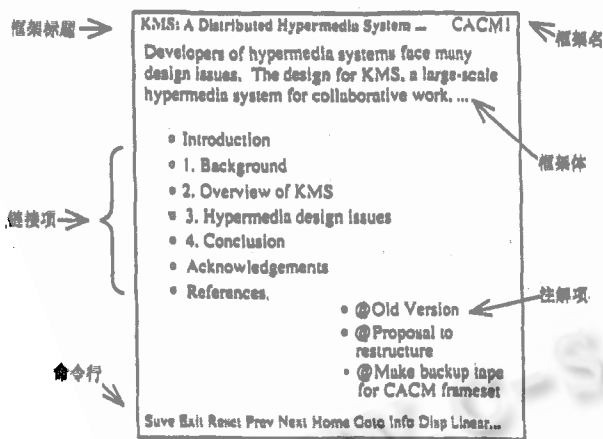


图 3 KMS 的框架结构

KMS 是为多用户网络环境设计的,它的阅读与编辑也是在同一环境下工作的,即用户可以一边浏览,一边批注,写下自己的观点,因而它存在一个如何保持数据一致性的问题。一般的作法是通过加锁来限制其它用户使用,这样为解决一致性问题所花费的代价比较大。KMS 认为在一个大系统中,两用户同时对一结点进行修改的可能性很小,基于这种考虑, KMS 实行了一种“软锁”的解决方法:若一用户正在修改某一结点,此时若其它用户要

编辑此结点将被警告。

3. Hyperties

Hyperties 是马里兰大学的 Ben Shneiderman 于 1983 开始的一个研究项目,当时称为 TIES。TIES (The Electronic Encyclopedia System) 是电子百科全书系统的缩写,后改名为 Hyperties,表示该系统用到了超文本技术,1987 年 Hyperties 已由 Cognetics 公司商品化。商品化的 Hyperties 主要应用于出版电子书。

商品化的 Hyperties 是在字符型显示器上使用的。它的结点是一篇文章或百科全书的一个条目,如果一屏显示不完一个结点时,可用命令行给出的翻页键翻屏;链是文章中的一小部份内容,从这里出发,可以看到有关的别的文章或条目,链指向的只能是一篇完整的文章或条目,而不能是一部份内容。

Hyperties 的用户界面很简单,它将可引出链的那部份内容用亮条显示,用户可用鼠标或光标键加 ENTER 键来跳至该链引出的另一篇文章或另一条目。

4. HyperCard

HyperCard 是 Apple 公司在 Macintosh 机上的产品。HyperCard 的设计者 Bill Atkinson 说: HyperCard 最初并不是为超文本设计的,而是为图像处理设计的。但最终的 HyperCard 产品确实具有超文本的特点:结点和链,因而现在大家都认为 HyperCard 是超文本产品。超文本的结点在 HyperCard 里被定义成“卡片”,具有相同格式的卡片组成“卡片叠”,HyperCard 的一个具体应用可以包含若干个这样的卡片叠。

HyperCard 的卡片由字段、按键、图像组成。字段用以记录字符串、数字等数据;按键表示用 HyperCard 的语言 HyperTalk 所写成的程序。当用户通过鼠标器选择了卡片上的按键时,与之对应的程序将被执行。由于按键所代表的程序由用户自己定义,因此按键经常用于执行由用户根据需要而设计的一系列动作。卡片上的图像可以用系统提供的工具来绘制,也可由其它装置,如扫描仪或摄像机输入。

HyperCard 允许在两张卡片之间建立链。链两端的卡片可以是来自相同或不同的卡片叠。链的建立由用户与系统采取交互方式完成,建立好的链可以选择用户设计好的图符表示在卡片上,当用鼠标器选择到这些图符时,用户将由一张卡片转移到另一张卡片。

除了使用链的非线性查找外,HyperCard 还允许用

用户对卡片叠的线性顺序查找,也允许利用索引进行查找,对卡片进行分类等复杂操作。

三、超文本系统结构的理论模型 及实现原则

1. 超文本系统结构的一个理论模型

从国外几个典型超文本系统的分析来看:除了结点之间存在链这一点相同,其它方面,象结点结构、链的连接形式、用户界面、数据的存取方法,都不相同,这主要由于超文本系统尚缺乏完整的理论指导。国外目前已有有人在超文本理论方面进行研究,提出了超文本系统结构的一些理论模型。

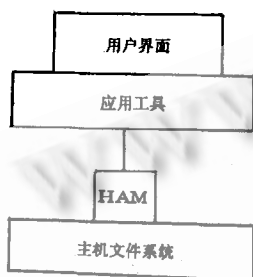


图4 使用HAM的超文本系统结构模型

Campbell 和 Goodman [2] 提出了一个使用 HAM (Hypertext Abstract Machine) 的超文本系统结构的理论模型(见图4)。HAM 是从 Tektronix 公司开发的 Neptune 抽象出来的,它是一个通用的超文本存储系统,是为基于事务处理的多用户环境设计的。由于 HAM 是一个低层次的通用灵活的存储系统,所以它可用于多种类型的超文本系统。

在这个模型中,HAM 将它管理的所有信息存放在主机的文件系统中,它通过流(byte stream)方式与应用通讯,应用工具通过用户界面与外部世界通讯,用户界面应是一基于窗口的环境,HAM 是多用户环境,应用可在不同的工作站上运行。

美国 NIST(National Institute of Standards and Technology)的超文本标准化工作组为解决超文本交换格式的标准提出了一个 Dexter 参考模型(见图5)。

为了解决实现上的困难,讨论问题方便,综合 HAM 模型和 Dexter 参考模型,可将一个超文本系统分成三层:即外部层、逻辑层、物理层(见图6)。外部层是超文本系

统的用户界面层,逻辑层是超文本系统结点和链的组织层,物理层是数据的存储、检索、网络共享的支持层。



图5 Dexter参考模型



图6 超文本系统结构的理论模型

(1)物理层。物理层是超文本系统的底层,其主要功能是信息的存储与检索,大量的超文本信息既可存储在硬盘、光盘上,也可存储在网络服务器里。不管信息如何存储、存储在何处,都须做到在短时间内能查到用户需要的那部分信息。

在物理层的实现上,它面临的问题:信息如何组织存放;多用户操作时,如何保证数据的安全性和一致性,如何对敏感数据进行保护;修改信息时,如何进行版本管理。当然有些问题的解决,比如存取控制,还可在物理层以外进行。对这些问题的解决,可以利用数据库管理系统已有的研究成果。

(2)逻辑层。逻辑层居于用户界面和物理存储的中间,是超文本系统的结点和链一层,这一层决定了结点结构和链的属性。在超文本系统的三个层次中,外部层随应用而定,物理层随计算机硬件而变,因此对超文本系统的理论研究和标准化工作可缩小在逻辑层中进行。

逻辑层的物理实现主要是定义一个合适的结点结构和链的属性。在现有的超文本系统中,结点结构各不相同,有的是框架,有的是链表;有的系统只有一种结点类型,有的类型有几十种。链有单向的,有双向的;有的链具有结构,有的链没有结构。其中结点的结构和链的属性主要还是由应用决定的。

(3)外部层。外部层是超文本系统的用户界面层,它决定了用户以何种方式使用超文本系统:如何显示结点和链;向用户提供什么命令;是否显示全局视图等。外部层的设计与三方面的因素有关:超文本系统的应用领域、逻辑层对结点和链的设计、超文本系统运行的硬件环境。

外部层的设计,比如显示如何布局;如何解决“迷路”问题等,对超文本系统的成功与否非常关键,外部层的实现可以利用用户界面的最新成果,比如可用触摸屏来方便非专业用户的使用。

把一个超文本系统分成三层,这样不仅有利于使用计算机其它方面的发展成果,而且有利于超文本系统的理论发展及超文本产品的标准化。比如用户界面的研究成果可用于外部层的设计,数据库管理系统的研究成果可用于物理层的实现,而超文本系统的理论研究及超文本产品的标准化只需在逻辑层进行。

2.超文本系统的实现原则

从超文本系统结构的理论模型及几个典型超文本系统的分析来看,设计并实现任一个超文本系统都将面临一些共同的问题。

(1)用户界面:应非常友好

用户界面是现代计算机系统一个最重要的方面,在软件系统设计中已经越来越重要了,它设计的好坏可能会直接导致一个软件的成败。因为软件系统对一般用户来讲,比较难学,在使用过程中,还会出现各种各样的错误,超文本系统与其它软件系统一样,也存在一个用户学习适应的问题,且超文本系统的用户基本上都是非计算机专业用户,因此用户界面的设计,特别是浏览界面的设计更显重要,不仅要易学易记,而且要少出错,甚至不出错;浏览系统的设计还应考虑迷路问题。

(2)输入输出:所见即所得、所听即所得

超文本系统最大的优点是图文声并茂,如何处理图像和声音,如何将图文声混排,是任何超文本系统都需解决的问题。显示非常重要,因为用户为了阅读超文本文件会在显示器前花费几个小时,目前已有研究表明:阅读同样长度的文章,从计算机上阅读要比读书本多花费 30% 的时间,因此,如何使显示赏心悦目非常重要。

3.反应速度越快越好

超文本系统的反应速度,是超文本系统是否实用的另一个衡量尺度。若反应速度太慢,读者会烦躁不安,显然没法使用。随着存储信息的越来越多,反应速度会显得越来越重要,并行和分布式处理也许是解决这一问题的办法。

4.打印:超文本到文本

超文本是一非线性文本,有些以层次结构为主结构的超文本系统,可以用深度遍历将其线性化。印刷中用黑体

字表示链,若链是用于注解的,可形成一脚注等。

超文本系统也应能接受传统的文本文件,并用适当的方法将其转化成超文本文件。

文本到超文本的转换,还有一个工作就是图像处理,图像格式与文本格式差别很大,处理起来比较困难。现在数字化技术可能会使转化容易一些,但是将图、文混合,一起处理仍是正在研究的课题。

5.协同工作

一个超文本系统应该允许多个人对同一文件进行结点的建立、修改和浏览,即共同工作。它在技术实现上有许多困难,这些困难与多用户数据库管理系统遇到的一样,最主要的问题是存取控制和版本管理。

一个大众都可使用的超文本系统,象数据库管理系统一样,显然应考虑安全问题:如何保持超文本文件的完整性和一致性,如何对敏感数据进行保护。这里也可借鉴数据库管理系统的经验,结点的创建者可以对其它用户定义一定权限:只读、只添加不修改、修改、拷贝、删除等。也可通过设置口令来限制非法用户存取。

6.兼容性

超文本系统对硬件、操作系统也有依赖性,将超文本系统从一计算机系统转到另一计算机系统,是非常困难的。

对兼容问题有两个解决方案:一是对不同机器或终端开发不同系统;另一种方案是超文本系统底层与用户界面分开,这样起码在移植时可以减小工作量。

超文本较之提出这一概念的 1945 年有了商业上、技术上实现它的可能性:内外存容量越来越大,处理速度越来越快,但价格越来越低;广泛的应用领域和巨大的市场潜力必将会使超文本系统在九十年代得到飞速的发展。

参考文献:

- [1] Frank G. Halasz, *Reflections on Notecards: Seven issues for the next generation of hypermedia system*, *Communications of the ACM*, July 1988, Vol.31 No.7, pp. 836-851
- [2] Brad Campbell, Joseph M. Goodman, *HAM: A general purpose hyper-text abstract machine*, *Communications of the ACM*, July 1988, Vol.31 No.7, pp. 856-861
- [3] Conklin E. J., *Hypertext: An introduction and survey*, *IEEE Computer*, Sept. 1987, pp. 17-41
- [4] Robert M. Akscyn, Donald L., Elise A. Yoder, *KMS: A distributed hypermedia system for managing knowledge in organizations*, *Communications of the ACM*, July 1988, Vol. 31 No.7, pp. 820-835