ANA ALICE RODRIGUES PEREIRA BAPTISTA

Modelação de Aplicações Hipermedia - Uma Abordagem

Dissertação para Mestrado em Informática
(Área de Informática de Gestão)

Escola de Engenharia
Departamento de Informática
UNIVERSIDADE DO MINHO

Braga, 1994

ANA ALICE RODRIGUES PEREIRA BAPTISTA
(Bolseira da Junta Nacional de Investigação Científica e Tecnológica)

Modelação de Aplicações Hipermedia - Uma Abordagem

Tese submetida à Universidade do Minho para obtenção do grau de mestre em Informática, área de especialização de Informática de Gestão, elaborada sob orientação do Professor Doutor Altamiro Barbosa Machado.

Escola de Engenharia
Departamento de Informática
UNIVERSIDADE DO MINHO

Braga, 1994

RESUMO

Desde sempre se assistiu à necessidade de planear as tarefas a desenvolver. Mesmo as mais simples exigem que se faça algum raciocínio prévio sobre o processo de as executar da forma mais rápida e menos trabalhosa. Para as tarefas mais complicadas, ou que sem serem complicadas implicam um grande volume de informação, já é necessário um planeamento mais cuidado e preciso, registando todos os passos que se forem dando no sentido da resolução do problema: vai ser necessário representar, de alguma forma, o conhecimento dentro do universo do problema.

Se para várias áreas, dentro da informática, já existem várias técnicas de representação do conhecimento, para a área dos sistemas hipermedia isso ainda não se verifica: das poucas que existem, a maioria ainda está sob investigação. Mais ainda: uma vez que hipermedia e multimedia exigem, por si só, uma grande interdisciplinaridade, as técnicas de representação do conhecimento devem ser o mais universais possível de modo a que possam ser usadas não só pelos informáticos, mas por todos os elementos da equipa de desenvolvimento da aplicação em causa.

Esta tese centra-se no desenvolvimento de uma técnica de representação do conhecimento para aplicações hipermedia tendo em conta as necessidades de uma equipa pluridisciplinar destinada à concepção deste tipo de aplicações. Além disso, são fornecidas as principais linhas de orientação para a sua execução.

Para o desenvolvimento deste trabalho tomou-se como base alguns dos conceitos fundamentais do paradigma orientado ao objecto tendo em conta que este está a ser cada vez mais utilizado como base dos sistemas autor hipermedia.

Palavras chave: Hipermedia, Multimedia, Representação do Conhecimento, Modelação, Metodologia, Paradigma Orientado ao Objecto.

ABSTRACT

The act of planning tasks has always been a critical requirements need through the time. Even the simpler tasks demand some intuitive planning in order to be executed faster and hardless. However, more complex ones, or those ones that deal with large volumes of data, have to be planned much more carefully and precisely way, writing down all the steps carried out to solve the problem: the knowledge within the problem's universe has to be represented some way.

If it is true that for several areas in computer science there are already many knowledge representation techniques, for hypermedia this isn't yet a reality: they are few and most of them are still under research. Moreover, once that hypermedia and multimedia demand, on their own, a great interchange of skills between specialists, the language they use to communicate with each other has to be as universal as it can be, in order to be used not only by computer science specialists, but by all members of the application development team.

This thesis is centred in a knowledge representation technique to hypermedia applications developed on the belief that the needs of such a team are of crucial importance. Additionally, are provided some guidelines regarding its execution.

Once that hypermedia authoring tools are increasingly supporting features based on the object oriented paradigm, this has also been used as a basis to the development of this approach.

Keywords: Hypermedia, Multimedia, Knowledge Representation, Modelling, Methodology, Object Oriented Paradigm.

ENTÃO um Professor disse:

- Fala-nos do Ensino.

E ele respondeu:

- Nenhum homem vos pode revelar nada
que não repouse já meio adormecido
na manhã do vosso conhecimento.

O mestre que caminha à sombra do templo,
entre os discípulos
não reparte a sua sabedoria
mas antes da sua fé e do seu amor.

Se for verdadeiramente sábio,
não vos convidará
a entrar na casa da sabedoria,
mas levar-vos-á
aos umbrais do vosso próprio espírito.

(...)

O ENSINO, Khalil Gibran.

Dedico este trabalho a
todos aqueles que foram e a
todos aqueles que são
meus professores.
Com um profundo agradecimento...

Gostaria de agradecer:

- ao Prof. Dr. Altamiro Machado, meu orientador e meu professor, por me ter "seduzido" para vir trabalhar nesta área. Foi, sem dúvida, uma das melhores opções que tomei em termos profissionais. Saliento a liberdade de acção que sempre me deu, o dinamismo que ajudou a imprimir ao meu trabalho bem como os desafios que constantemente me lançou;

- ao Dr. Pedro Rangel Henriques, meu eterno professor, por me ter convencido a fazer o mestrado e por sempre ter acreditado em mim, no meu trabalho e nas minhas convicções. Sempre que necessário apoiou-me, não só em termos profissionais como em termos humanos. Um obrigado muito grande para si, professor!

- ao Dr. João Álvaro Carvalho e ao Engº Mário Martins por me terem esclarecido algumas dúvidas em relação a partes muito específicas da minha tese e por se terem sempre mostrado disponíveis para a troca de ideias;

- à Olga Fernandes, minha companheira de mestrado e amiga, pelas muitas horas em que ficámos juntas em frente a "esta coisa de multimedia e hipermedia". E também por me teres trazido aqueles livros de Inglaterra!...

- à Maribel Alves por me ter disponibilizado o seu gabinete em Guimarães, local para onde eu ia trabalhar aquando do desenvolvimento da tese;

- a todas as pessoas que, apesar de não serem aqui citadas, contribuíram directa ou indirectamente para que esta caminhada chegasse ao fim com sucesso.

ÍNDICE

I - INTRODUÇÃO **1**

II - SISTEMAS E APLICAÇÕES HIPERMEDIA **6**

2.1 - O QUE É HIPERMEDIA? 6

2.1.1 - Hipertexto 8

2.1.2 - Multimedia 17

2.1.3 - Hipermedia 19

2.2 - UTILIZAÇÃO DE SISTEMAS HIPERMEDIA 22

2.2.1 - Porquê utilizar sistemas hipermedia? 22

2.2.2 - Vantagens / Desvantagens dos sistemas Hipermedia 28

2.2.3 - Aplicações Hipermedia 32

III - O ESTADO DA ARTE **35**

3.1 - HDM 36

3.2 - LIVRO ELECTRÓNICO HIPERMEDIA 40

3.3 - MODELO BASEADO NO HIPERGRAFO 45

3.4 - TRELLIS 50

3.5 - MODELO CONCEPTUAL USADO NO MULTUS 55

3.6 - DISCUSSÃO 58

IV - MODELAÇÃO DE APLICAÇÕES HIPERMEDIA - UMA ABORDAGEM **62**

4.1 - RESUMO 62

4.2 - O PARADIGMA ORIENTADO AO OBJECTO 64

4.3 - O DIAGRAMA DE LIGAÇÕES 70

4.3.1 - Entidade 70

4.3.2 - Ligação 72

4.3.3 - Simplificação do Diagrama 78

4.3.4 - Exemplo 81

4.4 - O DIGRAMA DE CONTEÚDOS 84
4.4.1 - Entidade 84
4.4.2 - Ligação 94
4.4.4 - O Anexo ao Diagrama de Conteúdos 96
4.4.5 - Exemplo 98
4.5 - METODOLOGIA A SEGUIR 103

V - TESTES E RESULTADOS 108

5.1 - TESTES REALIZADOS 108
5.2 - AVALIAÇÃO SEGUNDO OS PARÂMETROS UTILIZADOS PARA AS OUTRAS ABORDAGENS 112

VI - CONCLUSÃO 117

VII - BIBLIOGRAFIA 119

LISTA DE FIGURAS

Fig. 1.1 - Fases básicas da vida de um produto ou serviço. 2

Fig. 2.1 - A origem da palavra "hipermedia". 7

Fig. 2.2 - Documento organizado sob uma forma linear. 8

Fig. 2.3 - Organização da informação sob uma forma não-linear. 9

Fig. 2.4 - Os nodos de um documento hipertexto. 12

Fig. 2.5 - Os nodos e as ligações de um documento hipermedia. 14

Fig. 2.6 - Os nodos, as ligações e as âncoras de um documento hipermedia. 15

Fig. 2.7 - Exemplificação de parte de um roteiro turístico hipermedia. 16

Fig. 2.8 - Origem da palavra "hipermedia". 17

Fig. 2.9 - Hiperbase original. 26

Fig. 2.10 - Uma hipervisão sobre a hiperbase original na qual o utilizador criou mais dois nodos e ligações de e para eles. 27

Fig. 2.11 - Hipervisão sobre a hiperbase original de onde o utilizador retirou ligações. 27

Fig. 2.12 - Um *browser* gráfico e a sua relação com a base documental de hipertexto (hiperbase). 31

Fig. 3.1 - Modelo básico de um livro hipermedia. 40

Fig. 3.2 - Estrutura dos livros hipermedia. 41

Fig. 3.3 - Composição de uma página hipermedia. 42

Fig. 3.4 - Estrutura de um objecto hipermedia. 43

Fig. 3.5 - Exemplo do estado do utilizador usando o modelo do grafo orientado pesado. 46

Fig. 3.6 - Exemplo do estado de um utilizador de acordo com o modelo do hipergrafo. 48

Fig. 3.7 - Estrutura de uma aplicação hipertexto baseada numa rede Petri. 52

Fig. 3.8 - Exemplo de um documento multimedia e sua estrutura conceptual. 56

Fig. 4.1 - As classes "PASSAGEIROS" e "MERCADORIAS" segundo a notação de Coad e Yourdan [CY90]. 66

Fig. 4.2 - Parte de uma rede de classes denotando a estrutura de herança segundo a notação de Coad e Yourdan [CY90]. 66

Fig. 4.3 - Parte de uma rede de classes denotando as estruturas de herança e composição segundo a notação de Coad e Yourdan [CY90]. 67

Fig. 4.4 - O paradigma orientado ao objecto como base de aplicações hipermedia. 69

Fig. 4.5 - Representação gráfica de uma entidade. 71

Fig. 4.6 - Existe uma ligação 11 de cada instância da entidade E1 para uma instância da entidade E2. E1 é a entidade-fonte e E2 é a entidade-destino. 73

Fig. 4.7 - Existe uma ligação de cada instância da entidade E1 para uma ou mais instâncias da entidade E2. E1 é a entidade-fonte e E2 é a entidade-destino. 73

Fig. 4.8 - Existe uma ligação com volta 11 de cada instância da entidade E1 para uma instância da entidade E2. E1 é a entidade-fonte e E2 é a entidade-destino 73

Fig. 4.9 - Convenção sobre o uso de nomes nas ligações. 74

Fig. 4.10 - Exemplo da existência de várias ligações da entidade "ÍNDICE" para a entidade "PROFISSÕES"... 75

Fig. 4.11 - Simplificação de duas ligações com sentido inverso entre as mesmas entidades 78

Fig. 4.12 - Simplificação de duas ligações com volta com sentido inverso entre as mesmas entidades 78

Fig. 4.13 - Simplificação de três ligações 11 com diferentes entidades-fonte. 79

Fig. 4.14 - Simplificação de uma ligação 11 que tem várias entidades-destino 80

Fig. 4.15 - Simplificação de três ligações 11 (com vários destinos cada uma) que têm diferentes entidades-fonte. 80

Fig. 4.16 - Diagrama de Ligações não simplificado relativo à aplicação sobre animais. 81

Fig. 4.17 - Diagrama de Ligações simplificado relativo à aplicação sobre animais. 81

Fig. 4.18 - Notação de representação gráfica de um componente. 87

Fig. 4.19 - Componente com cardinalidade "1 ou mais". 88

Fig. 4.20 - Existe uma relação de convivência entre o componente C1 e o componente C2, i.e., os componentes C1 e C2 são componentes conviventes. 89

Fig. 4.21 - Definição de alguns espaços de um nodo. 91

Fig. 4.22 - Notação de representação gráfica de um espaço. 91

Fig. 4.23 - Utilização dos espaços num Diagrama de Conteúdos 91

Fig. 4.24 - Notação de representação gráfica de uma perspectiva-componente. 93

Fig. 4.25 - Notação de representação gráfica de uma perspectiva-espaço. 93

Fig. 4.26 - Notação de representação gráfica de uma perspectiva por defeito (componente ou espaço). 93

Fig. 4.27 - Notação de representação gráfica de uma ligação.....95

Fig. 4.28 - Anexo ao Diagrama de Conteúdos para a entidade E1. 96

Fig. 4.29 - Diagramas de Conteúdos para a entidade CAPA. 98

Fig. 4.30 - Diagrama de Conteúdos para a entidade ESCOLHA.....99

Fig. 4.31 - Diagrama de Conteúdos para a entidade ESPÉCIE. 101

Fig. 4.32 - Diagrama de Conteúdos para a entidade FAMÍLIA. 104

Fig. 4.33 - Diagrama de Conteúdos para a entidade GÉNERO.....104

LISTA DE QUADROS

Quadro 4.1 - Notação de representação da cardinalidade das prórpias ligações. 76

Quadro 4.2 - Notação de representação da cardinalidade dos destinos de cada uma das ligações. 77

Quadro 4.3 - Notação de representação da cardinalidade dos componentes. 87

I - INTRODUÇÃO

Desde sempre se assistiu a uma necessidade de planear as tarefas a desenvolver. Por vezes não nos apercebemos desta etapa de planeamento, pois ela é realizada de uma forma automática no interior da nossa mente. Por exemplo, sempre que alguém vai fazer um desenho, sabe que antes de tudo tem que pegar no papel e no lápis, colocar o lápis na mão mais forte (se não quiser fazer uns rabiscos) e só depois começa a desenhar.

Como esta, existem milhares de situações quotidianas que são realizadas utilizando uma forma de "planeamento automático" que a nossa brilhante mente se encarrega de realizar e gerir sem que tenhamos de tomar uma consciência directa de todas estas fases.

Há, contudo, situações que sendo mais complexas, exigem uma forma de planeamento mais cuidado. Além disso, existem tarefas que pela interdisciplinaridade que lhes é inerente, exigem vários especialistas de áreas diferentes a trabalhar em conjunto, pelo que necessitam de um vasto "repositório" de informação partilhada. Por outro lado, sempre que há tarefas sequenciais, é necessária a comunicação de informação entre os vários realizadores dessas tarefas.

Parece óbvio que se queremos ser eficientes e eficazes na realização de tarefas complexas que, eventualmente, envolvam várias pessoas a realizá-las, ou ainda das quais outras dependem, é necessário planear de algum modo o que irá ser feito e arranjar uma forma de representação desse planeamento compreensível para todos os envolvidos.

Qualquer produto, seja um bem ou um serviço, deverá passar por quatro fases básicas durante a sua vida, de modo a que possa ter sucesso (aqui só se está a pensar no produto em si, não entrando em considerações sobre técnicas de marketing para colocação do produto no mercado). Estas fases, como se mostra na figura são: 1) Análise de requisitos; 2) Concepção; 3) Execução; e 4) Manutenção.

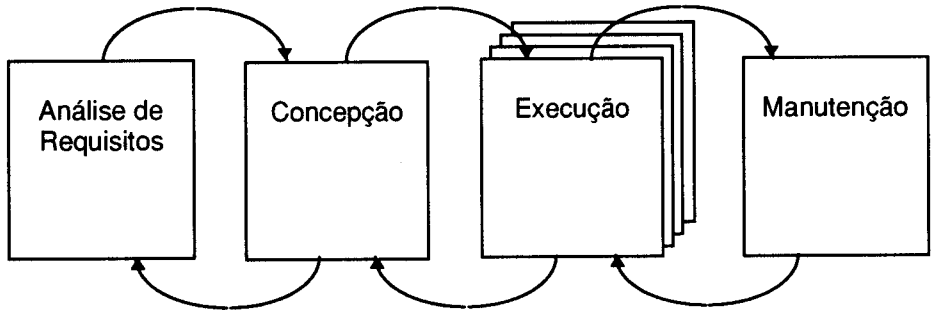


Fig. 1.1 - Fases básicas da vida de um produto ou serviço.

A primeira fase concentra-se na definição dos requisitos que o produto deverá ter para satisfazer as necessidades do consumidor potencial.

A segunda fase diz respeito à concepção do produto tendo como ponto de partida o resultado da primeira fase. Aqui podemos já introduzir as variantes técnicas mais relevantes.

A terceira fase prende-se com a execução ou "fabrico" do produto.

A quarta e última fase, sem dúvida muito importante para o prolongamento da vida do produto, diz respeito à sua readaptação às novas realidades socio-tecnológicas. Esta fase supõe um retorno às fases anteriores de modo a que se possam realizar alterações coerentes com o próprio produto.

Isto é também verdade para as aplicações hipermedia ou hiperdocumentos. É com a segunda fase, i.e., a de concepção e desenho do produto que nos vamos preocupar.

Tradicionalmente, em sistemas de informação, é-se habituado desde cedo a planear sempre a aplicação, por mais pequena e simples que seja, e depois passar à sua implementação. Quando apareceram no mercado os sistemas de hipertexto mais simples, várias pessoas foram levadas a pensar e a afirmar que finalmente tinham chegado sistemas cujas aplicações não seria necessário planear cuidadosamente, sob o pretexto da grande vantagem da organização não-linear da informação. Era muito simples: iam-se acrescentando nodos e ligações *(links)* sempre que necessário e estava o trabalho todo feito. Mas entretanto, com o tempo

apareceram três problemas graves: desorientação do utilizador, sobrecarga cognitiva e grandes dificuldades na manutenção de tais hiperdocumentos devido às extensas ligações entre os nodos (o que Janet Fiderio chama de *"spaguetti code"* [FID88] ou Ben Shneiderman chama de *"hyperchaos"* [SHN89]).

Sob o ponto de vista da organização da informação, é ainda mais difícil criar boas aplicações agora do que com os sistemas tradicionais de organização da informação, pois a permissividade é tal, que se não se tem cuidado acaba-se por chegar a hiperdocumentos com ligações incontroláveis.

Um outro problema que se foi notando era o mau aproveitamento da possibilidade de re-utilização da informação: deveria ser possível utilizar um nodo várias vezes (fazer-lhe diversas ligações a partir de outros nodos). A não utilização desta grande vantagem dos sistemas de hipertexto - uma das suas características básicas - levou à duplicação de informação e, consequentemente, a um volume ocupado bastante superior ao que seria de esperar.

Posteriormente, com o aparecimento de ferramentas mais sofisticadas (com poderosos *interfaces* gráficos, linguagens de programação muito aproximadas da natural e uma abordagem já com algumas similaridades com o paradigma orientado ao objecto) a facilidade de uso e manipulação por não-informáticos, reclamada por muitos - e com razão - como uma das grandes vantagens deste tipo de sistemas, foi-se tornando cada vez mais notória.

Começaram-se a criar aplicações, se não sem planeamento prévio nenhum, pelo menos com um planeamento realizado sobre uma base intuitiva com pouco ou nenhum suporte científico. Dada a facilidade de uso de algumas ferramentas autor, estas começaram também a ser utilizadas para rápida prototipagem e além dos problemas antes mencionados permanecerem, surgiu um outro, devido a esta nova utilização destes sistemas: um protótipo é um protótipo e não passa disso - o problema surge exactamente quando se quer utilizar na aplicação final a mesma estruturação da informação usada no protótipo. Este que é utilizado essencialmente para medir a correspondência entre a aplicação final e os requisitos do utilizador, tem como preocupações fundamentais o *interface* com o utilizador e a realização de funções por ele definidas. Quanto à organização interna da informação, ele não é de modo nenhum uma base de partida para a aplicação final.

Agora, com o aparecimento de várias extensões media aos sistemas anteriormente existentes, torna-se urgente a definição prévia da estruturação da informação de uma dada aplicação. Há a necessidade de tomar as decisões importantes antes de criar a aplicação para que os dados e o comportamento das aplicações sejam adicionados coerente e coesamente e não "colados" depois de a aplicação já estar feita.

Philip Barker, em [BAR92], salienta que a tarefa de construir uma aplicação hipermedia é difícil e leva muito tempo e, por isso, «é importante assegurar que se usem modelos bem fundados e linha mestras de acção como base para os processo de desenho e produção envolvidos». Diz, ainda que uma das facetas importantes de um modelo de desenho «(...) é a produção de *interfaces* para o utilizador final de alta qualidade que facilitarão o controlo e a transferência de conhecimento»[BAR92].

Independentemente da ferramenta autor onde qualquer aplicação vai ser desenvolvida, existem muitas vantagens em modelar o seu desenho. Franca Garzotto, Paolo Paolini e Daniel Schwabe citam, em [GAR91], os seguintes benefícios dos modelos:

- «(...) providenciam uma linguagem na qual um analista pode especificar uma dada aplicação ajudando a **comunicação entre o analista e o utilizador final**»;

- «(...) podem ser usados para **documentar a aplicação**, providenciando apoio aos utilizadores da aplicação e à manutenção do sistema»;

- « a existência de uma linguagem de especificação comum permite facilmente **comparar aplicações**»;

- « providenciam uma base de trabalho na qual os autores das aplicações hipermedia podem **desenvolver, analisar e comparar metodologias de desenho e estilos retóricos de *hyperauthoring***, a um alto nível de abstracção (...) A este nível é possível analisar a organização conceptual do domínio do conhecimento representado da aplicação e examinar a adequação aos utilizadores finais»;

- « a tarefa de ***proofreading*** (de uma aplicação de hipertexto) deve ser ajudada pela disponibilidade de uma linguagem de especificação»;

- « os modelos de desenho podem ser usados por **ferramentas de desenho**»;

- « as aplicações desenvolvidas de acordo com um modelo resultarão numa estrutura de representação muito previsível. Consequentemente, o ambiente de navegação para os leitores deve também ser previsível **reduzindo**, por isso, o tão falado problema de **desorientação/sobrecarga cognitiva**».

A esta extensa lista penso que só falta acrescentar a maior possibilidade de tirar partido da **re-utilização da informação** (usar o mesmo nodo várias vezes) providenciada pelos sistemas de hipertexto/hipermedia, uma vez que a partir do modelo da aplicação é possível analisá-la a um nível mais abstracto e, por isso, ter uma noção global muito mais completa e real.

É propósito deste trabalho apresentar uma abordagem à modelação de sistemas hipermedia com base no paradigma orientado ao objecto. O seu âmbito de actuação situa-se na fase inicial de concepção. Diz-se inicial porque se considera que neste tipo de aplicações existem duas grandes fases de concepção: a primeira fase em que toda a equipa pluridisciplinar (característica deste tipo de aplicações) modela a aplicação e uma segunda fase em que esta modelação é refinada, de acordo com o anteriormente definido, por cada uma das partes disciplinares da equipa (conteúdos, desenho de *interfaces*, análise informática, etc.). Assim, depois da aplicação dos métodos e técnicas aqui definidos, cada uma das partes envolvidas no projecto usá-la-á como base de trabalho para a segunda fase: a parte informática, por exemplo, utilizá-la-á para realizar a análise orientada ao objecto.

A descrição exaustiva da abordagem desenvolvida é feita no capítulo IV. No entanto, antes, no capítulo II, faz-se uma breve digressão sobre as filosofias de organização e apresentação dos sistemas hipermedia em geral, tendo como base tanto os sistemas de hipertexto como os sistemas multimedia. No capítulo III apresentam-se algumas abordagens à modelação de sistemas hipermedia: o estado da arte. A avaliação da abordagem aqui apresentada segundo os parâmetros utilizados para avaliar as aproximações apresentadas no terceiro capítulo e os resultados do seu uso são discutidos no capítulo V. Finalmente, no capítulo VI, descrevem-se as conclusões do trabalho.

II - SISTEMAS E APLICAÇÕES HIPERMEDIA

2.1 - O que é Hipermedia?

> «Imagine, if you will, walking into the New York Public Library and picking up a book on Mozart. You begin to read and learn that Mozart was an Austrian composer in the latest 1700s. You wonder what else was happening in Austria then, so you go to the card catalog, find a book on Austrian history, go to the stacks, locate the volume (if it's not checked out), and read it before you continue.
>
> In this book, you find a reference to old Salzburg, and you wonder what it looked like. Back to the card catalog, and the stacks, to find a book of images from that time. Finally, you get back to Mozart and read of a piano concerto you've never heard. This time you head for the library's record collection and listening room.
>
> This process continues until you have either satisfied your desire of knowledge on the subject or worn yourself out of searching for it, whichever comes first.
>
> Now imagine sitting at your computer and bringing up a hypertext system on music. You begin to read about Mozart. When you wonder about Austrian History, you simply highlight the text and request more information with a mouse click or a few keystrokes. To find images of old Salzburg, you use the same process. And to hear the piano concerto? The same.
>
> Sounds a lot simpler, doesn't it? The only restriction to this seemingly endless fountain of knowledge is that the author of the hypertext system had to establish the connections for you to follow and provide the additional knowledge for you to retrieve.»
>
> in TAZELAAR, Jane M. - *Hypertext*. «BYTE», 1988, Outubro, p.234.

Actualmente, no domínio da informática, assiste-se a uma tendência cada vez maior para proporcionar ferramentas úteis poderosas e de fácil manuseamento ao utilizador final, não-informático. Os ambientes de trabalho são atraentes, geralmente possuindo um *interface* gráfico que, com a ajuda do rato, permitem uma

utilização quase intuitiva, ficando, assim, de parte a necessidade de recorrer a palavras de comando esotéricas, aparentemente sem nenhum significado para o utilizador.

Hoje em dia, cada vez mais pessoas comunicam através dos media electrónicos e informáticos. Talvez isto aconteça porque também cada vez é mais rápido fazê-lo, para o que muito contribui a pesquisa e desenvolvimento de meios de transmissão digital da informação, em particular das fibras ópticas. Os tempos mortos de comunicação esbatem-se no pano de fundo da percepção humana.

Em cooperação com estas facilidades aparece, ainda, a possibilidade de armazenamento de grandes quantidades de informação na forma digital proporcionada pelos discos ópticos, o aumento da capacidade de memória central dos computadores, a cada vez maior velocidade de processamento, a evolução dos conhecimentos algorítmicos e, por isso, o desenvolvimento de *software* bastante poderoso e fortemente interactivo.

Os sistemas **hipermedia** que apareceram recentemente no mercado, são poderosas ferramentas de *software* que tiram vantagem de todas as potencialidades de *hardware* e conhecimento algorítmico disponíveis actualmente. A sua história é longa e o conceito a eles associado já data de há algumas décadas atrás, embora só há muito pouco tempo tivesse podido ser posto em prática. Na verdade, primeiro apareceu o conceito de **hipertexto** e só mais recentemente o de hipermedia.

Hipermedia poderia ser definido através da seguinte figura:

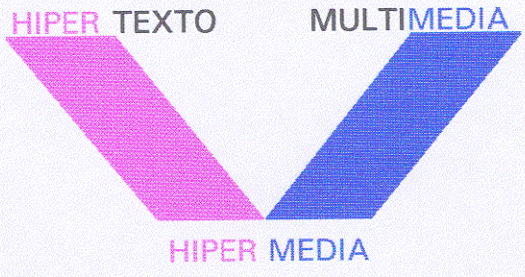


Fig. 2.1 - A origem da palavra "hipermedia".

No entanto, este é um caso em que o todo é maior que a soma das partes, como se constatará depois da explicação de cada uma delas.

2.1.1 - Hipertexto

O hipertexto, antes de tudo, apresenta uma nova filosofia de organização da informação: a informação escrita é apresentada sob uma forma sequencial em livros, tratados e outros documentos; o hipertexto permite organizá-la de uma forma não-linear. Um corpo de informação que seja constituído por vários fragmentos é organizado através de relações não-lineares entre esses fragmentos, sendo possível "saltar" de um fragmento para outro qualquer que a ele esteja ligado. As figuras seguintes mostram a diferença entre estas duas formas de organização da informação:

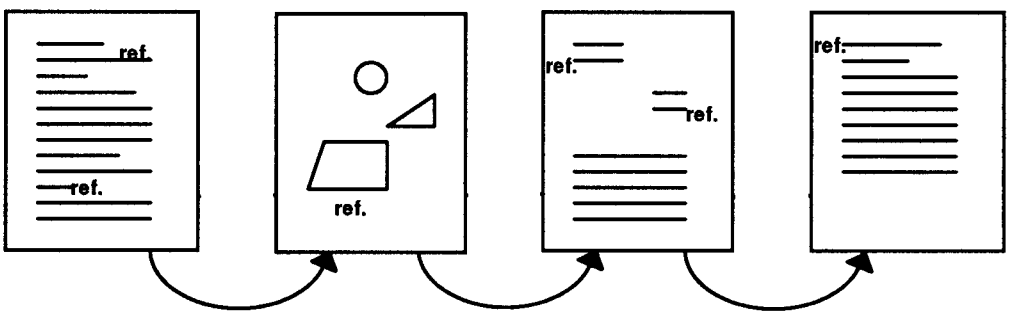


Fig. 2.2 - Documento organizado sob uma forma linear.

A figura anterior mostra um exemplo da organização da informação sob uma forma puramente linear: é o clássico "página anterior" - "página seguinte". As palavras "**ref.**" designam referências às quais o leitor do documento não tem um acesso directo a partir do ponto em que se encontra. Para encontrar os pontos aos quais se fazem referência o leitor tem que os procurar em todo o corpo de informação existente.

Na figura seguinte vê-se um exemplo da organização da informação sob uma forma não-linear: aqui existe um entrecruzar de referências electrónicas entre as várias páginas do documento. O leitor não necessita de reduzir a sua consulta a uma forma sequencial - antes, pode seguir as referências existentes de um modo imediato e automático sem ter que andar a procurar por elas em todo o corpo de informação envolvido.

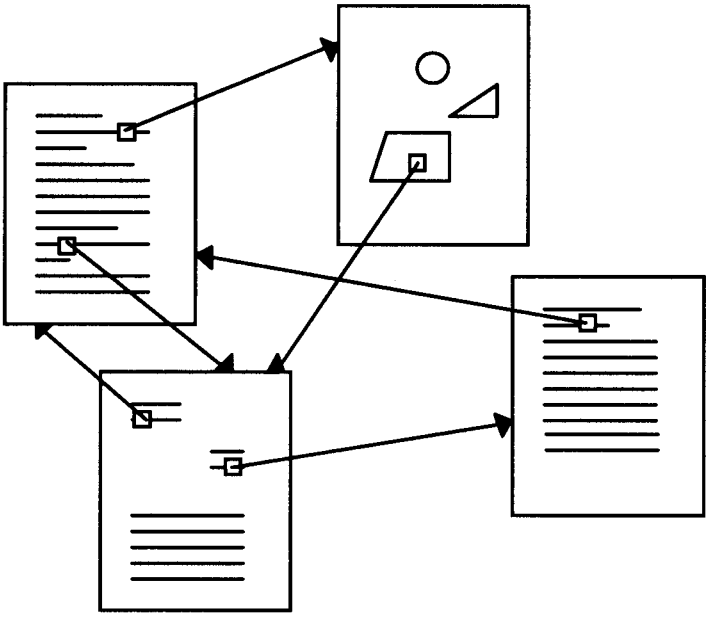


Fig. 2.3 - Organização da informação sob uma forma não-linear.

2.1.1.1 - História do Hipertexto

Apesar de esta filosofia de organização da informação só ter sido implementada em sistemas informáticos comercias há cerca de meia dúzia de anos, a verdade é que já há muito tempo que se faz investigação sobre ela, no meio científico.

Com efeito, a ideia original de um tipo de sistema deste género, considerado o antepassado do hipertexto, foi introduzida por Vannevar Bush, director do escritório de pesquisa e desenvolvimento científico dos EUA Em 1945, no artigo «*As we may think*», Bush defendia que «os nossos métodos de transmissão e revisão dos resultados de pesquisa são extremamente antigos e actualmente são totalmente inadequados nos seus propósitos» [RIT89]. A sua intenção era criar uma máquina, a que ele chamou *The Memex* que permitiria a ligação, através de um esquema de referências pré-definido, de todos os livros, registos e outros tipos de informação existente na biblioteca presidencial. Segundo ele próprio afirmaria, a máquina *(The Memex)* consistiria em «um dispositivo no qual um indivíduo guarda os seus livros, registos e comunicações, e que é mecanizada de tal modo que permita a consulta com excessiva velocidade e flexibilidade» [RIT89].

Contudo, foi Ted Nelson que, num artigo intitulado *Dream Machines* (1965), introduziu o termo *hypertext*:

> «Por hipertexto eu refiro-me a escrita não-sequencial.
>
> A escrita vulgar é sequencial por duas razões. Primeiro, ela saiu da fala... que tem de ser sequencial e, segundo, porque não é cómodo ler os livros de um modo não-sequencial.
>
> Mas as estruturas de ideias não são sequências; elas entrelaçam-se em qualquer direcção. E quando nós estamos a escrever, estamos sempre a tentar entrelaçar as coisas em nodos não-sequenciais» (RIT89)

O projecto de hipertexto de Ted Nelson, "Xanadu", pode ser considerado o mais ambicioso de todos até à data: tem a ver com a construção de um *hypertext server* que permitiria interligar toda a literatura mundial tornando-a disponível ao utilizador a partir de um terminal. Já há bastante tempo que estão a ser desenvolvidos outros projectos de grande envergadura a nível laboratorial em várias universidades, institutos e empresas de todo o mundo. Um dos que se encontra actualmente em desenvolvimento, por exemplo, é o *Hypertext Authoring* (HYTEA), um projecto no âmbito do programa ESPRIT, cujo objectivo é implementar um conjunto de ferramentas de apoio ao desenvolvimento de aplicações hipertexto/hipermedia grandes e complexas.

Só cerca de 43 anos depois do artigo de Vannevar Bush é que os sistemas de hipertexto/hipermedia encontram o seu caminho para o mercado, fora dos laboratórios de pesquisa. Alguns exemplos desses sistemas são o *SuperCard*, para o Apple Macintosh, o *ToolBook* para IBM compatíveis (corre sob o *Microsoft Windows*), ou o *Craftman*, para a plataforma NeXT.

2.1.1.2 - Conceitos

Ben Shneiderman [SHN89] apresenta o que ele considera as **três regras de ouro do hipertexto**:

- existe um grande corpo de informação composto por vários fragmentos
- os fragmentos estão inter-relacionados e
- o utilizador precisa apenas de um pedaço de cada vez

Estas regras implicam que os fragmentos de informação são **modulares**, i.e., o conteúdo de cada um deles não é directamente dependente do conteúdo de nenhum outro. Implicam, também, que os fragmentos de informação estão relacionados uns com os outros de uma forma **não-sequencial**, não linear, i.e., a relação entre eles não se resume ao clássico anterior-seguinte, mas a um entrelace de referências de uns para os outros.

Os conceitos utilizados na generalidade dos sistemas de hipertexto são os seguintes:

- pedaços ou fragmentos de informação - **nodos**
- ligações entre esses fragmentos - **ligações**
- "lugares" onde as ligações estão "penduradas" - **âncoras**

De seguida far-se-á uma exposição breve de cada um destes conceitos.

Nodos

Um nodo diz respeito a um pedaço de informação **modular**, i.e., o seu conteúdo não está directamente dependente de nenhum outro nodo - não há relações de precedência, em termos de conteúdo, entre os nodos.

Este é um aspecto muito importante e que se deve ter em conta na construção de uma aplicação. O leitor / utilizador não deve ser obrigado a lembrar-se do conteúdo dos nodos anteriores para compreender o conteúdo do nodo corrente. Se isto acontecer diz-se que há **sobrecarga cognitiva**. Deste modo, na impossibilidade de se anular, deve reduzir-se ao máximo a sobrecarga cognitiva entre os nodos.

Cada um desses fragmentos modulares de informação pode ser constituído por outros pedaços ou fragmentos que por si só não têm significado aparente - todos juntos constituem um nodo para representar uma ideia ou um assunto. Um nodo pode conter texto, gráficos e ainda outro *software* ou dados. Cada nodo de uma aplicação hipermedia diz respeito a um dos conceitos associados ao assunto de que trata a aplicação.

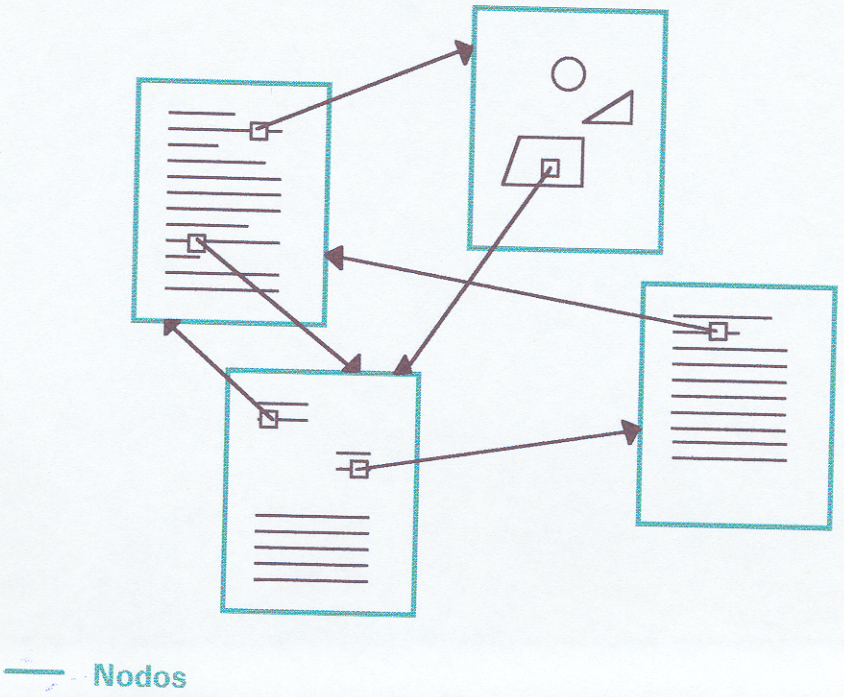


Fig. 2.4 - Os nodos de um documento hipertexto.

Ligações

As ligações são o que permite definir e manter as **relações não-lineares** entre os vários nodos de um hipertexto. É claro que os nodos podem ser ligados de uma forma sequencial, mas assim não se estão a aproveitar as verdadeiras potencialidades das ferramentas de hipertexto - é que «(...) ironicamente, muitas vezes, é o desejo de imitar o papel que resulta em hipertexto pobre»[RIT89].

O poder do hipertexto reside nas ligações que transformam pedaços de informação isolados numa única estrutura documental organizada sob a forma de um grafo. São estas ligações (hiperligações) que permitem a navegação não-sequencial de documentos hipertexto (hiperdocumentos). Do ponto de vista da programação, uma ligação pode ser vista como um simples **apontador** de um nodo para outro. Este apontador está definido no nodo-fonte (nodo de onde parte a ligação) e aponta para um ou mais nodos-destino (nodo para onde a ligação "aponta"). Quando a ligação é atravessada, o nodo-fonte é desactivado sendo activado um dos nodos-destino.

Nos sistemas de hipertexto mais simples, uma ligação relaciona apenas dois nodos - o nodo-fonte e o nodo-destino. Há, contudo, sistemas que permitem que uma ligação tenha mais do que um nodo-destino. A selecção do nodo-destino é feita durante a execução (em *runtime*) de acordo com condições pré-especificadas. Uma destas condições pode ser, por exemplo, a história navigacional do leitor, ou o seu perfil. Existem, ainda, sistemas de hipertexto que permitem fazer ligações não só entre os vários nodos, mas também entre os componentes desses nodos.

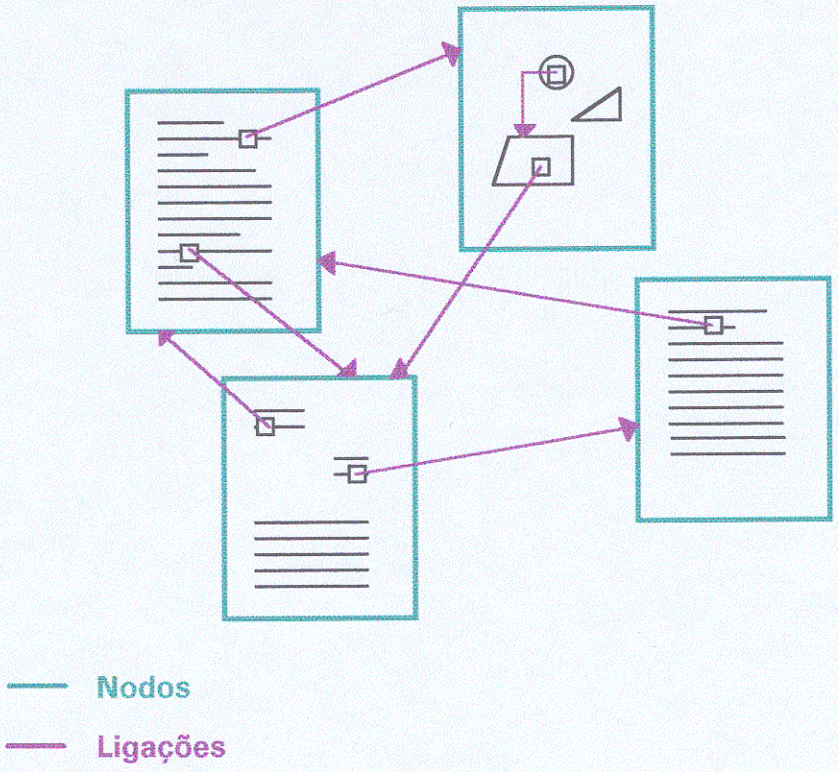


Fig. 2.5 - Os nodos e as ligações de um documento hipermedia.

Âncoras

Uma âncora (ou âncora de ligação) consiste geralmente num **botão**, num desenho, numa **palavra** ou num **segmento de palavras** que tem uma ligação associada. Ao seleccionar-se uma âncora (por exemplo clicando no botão de um rato) "executa-se" a respectiva ligação. Geralmente as âncoras estão bem visíveis de modo a que o leitor do hiperdocumento se aperceba facilmente de que existe ali uma ligação potencial. Mesmo que a âncora corresponda a uma palavra ou a um segmento de palavras, o cursor modificará a sua forma ao passar por cima dela.

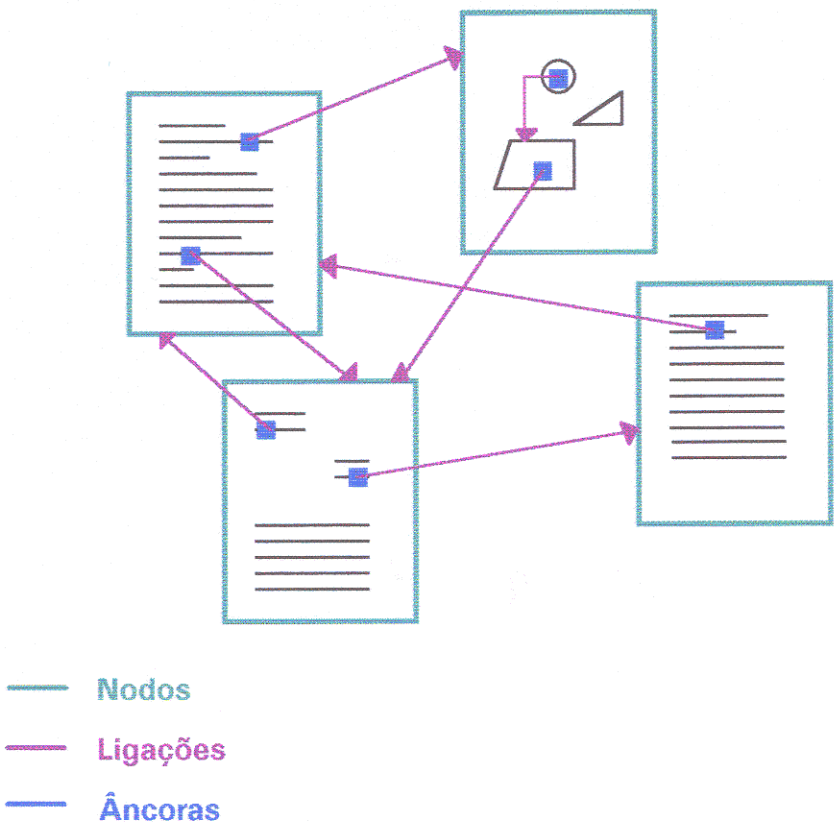


Fig. 2.6 - Os nodos, as ligações e as âncoras de um documento hipermedia.

Geralmente, um hiperdocumento é apresentado ao utilizador como um conjunto de janelas num écran. Cada uma destas janelas corresponde a um nodo do hiperdocumento, ou então a informação organizacional sobre ele (*browsers* gráficos por exemplo). Misturados com o texto ou gráficos, aparecem palavras ou símbolos realçados de alguma forma (a *highlight* ou com a sensação de relevo por exemplo), a que vulgarmente se chamam botões, que permitem passar para outro pedaço de informação, abrindo uma nova janela no écran. Cada um destes botões, ao serem premidos, desencadeia uma acção que executa a ligação (previamente definida).

A partir daqui pode-se continuar a "saltar" para outros nodos do hiperdocumento, ou então voltar para trás, à vontade do utilizador. Através dos botões é também possível invocar outro *software* ou dados. Estas potencialidades permitem, com imaginação, realizar hiperdocumentos com características bastante interessantes, facilitando a aquisição de informação e conhecimento.

Um exemplo de uma aplicação hipermedia é o roteiro turístico de uma cidade. Aqui pode haver nodos que contenham informação sobre estátuas, edifícios históricos, ruas e praças antigas, hotéis, restaurantes, serviços, etc. No início da aplicação pode aparecer ao leitor um mapa com pontos sensíveis aos quais estão associadas ligações que "apontam" para determinado nodo, sendo assim possível ao leitor "passar" directamente do mapa para o nodo que contém a informação que ele quer consultar. Pode, ainda, haver referências cruzadas entre os vários nodos, como por exemplo entre pontos importantes da mesma rua. A figura seguinte mostra um pequeno exemplo desenhado do que seria parte dessa aplicação:

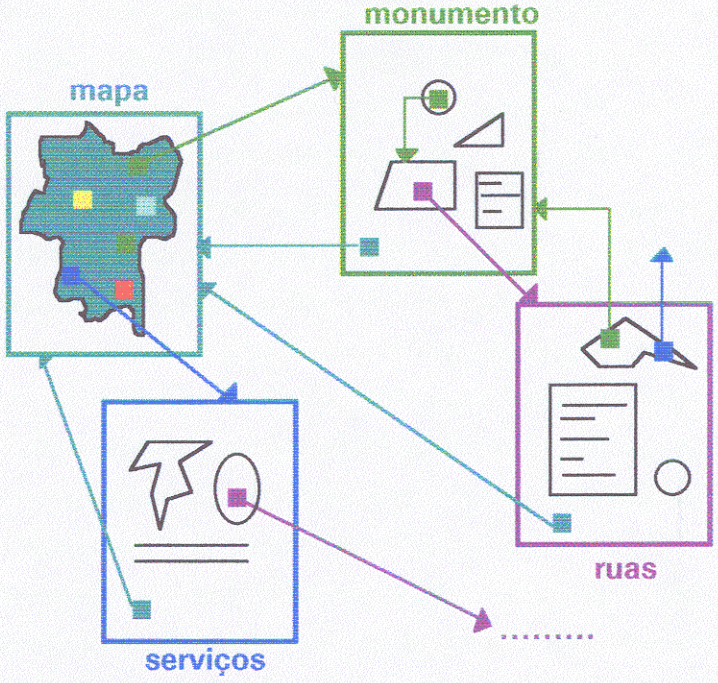


Fig. 2.7 - Exemplificação de parte de um roteiro turístico hipermedia.

Outro exemplo poderia ser uma aplicação sobre cantores e grupos famosos. Neste caso existiria um nodo inicial tipo índice com informação sobre os vários tipos de música abordados (jazz, rock, pop, ópera, etc.). A partir de uma selecção neste nodo, poder-se-ia chegar a outro com opções sobre os vários cantores ou grupos. Os nodos com informação relativa a estes poderiam estar inter-relacionados tendo em conta o tipo de

música (consulta para mais cantores ou grupos de determinado tipo de música), a época em que se tornaram famosos etc.

2.1.2 - Multimedia

Recorde-se a seguinte figura, já apresentada anteriormente:

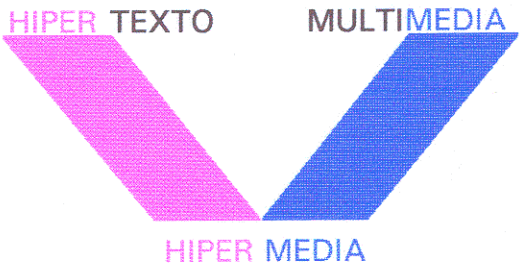


Fig. 2.8 - Origem da palavra "hipermedia".

Isto significa que hipermedia conjuga os conceitos de hipertexto com os de multimedia. Já se falou em relação ao hipertexto, pelo que chegou a vez de se explicar o que significa multimedia. A possibilidade de uma aplicação hipermedia poder manipular e apresentar informação em vários formatos media diferentes advém não das características do hipertexto, mas das características de multimedia.

É importante notar que multimedia não é a mesma coisa que "muitos media". É diferente ter-se, por exemplo, a apresentação de um produto utilizando um retroprojector, um filme em vídeo e um projector se *slides*, ou ter uma apresentação feita utilizando uma ferramenta multimedia, possuindo todos estes media integrados e ajustando-os uns aos outros de modo a que se possa obter determinado impacto, numa apresentação coerente e "viva". Aqui nem se fala em outras vantagens de multimedia em relação a "muitos media", como por exemplo a interactividade fornecida ou espaço físico de armazenamento necessário. A intenção é, apenas, focalizar a diferença óbvia do impacto causado nos consumidores finais.

Suponha-se, por exemplo, que um gestor está a fazer uma apresentação à administração da empresa em que trabalha. Se utilizar "muitos media", ou tem uma pessoa a controlar cada um dos media e todas estas pessoas estão perfeitamente sincronizadas e ajustadas umas com as outras, ou então usa um media de cada vez ou, ainda - de modo a não arriscar um fiasco total - utiliza o tradicional retroprojector, talvez o quadro e mais nada! Suponha-se, agora, que ele possui um sistema multimedia para fazer a apresentação: uma vez que se supõe a integração, pelo menos lógica, de todos os media no computador, a tarefa deste gestor torna-se muito mais facilitada pois a apresentação usando os diferentes media é controlada através do sistema de acordo com as pré-especificações realizadas. Assim concerteza que ele pode contar com uma "ajudinha" da tecnologia para convencer a administração.

Os formatos media existentes dizem respeito a:

- Texto
- Imagem
    * Fixa
        - ♦ Gráfico
        - ♦ Desenho
        - ♦ Fotografia
    * Móvel
        - ♦ Animação
        - ♦ Vídeo

* Som
    - ♦ Ruído
    - ♦ Voz
    - ♦ Música
* Programas de Computador

Ao contrário dos outros, a animação, o vídeo e os vários tipos de som são formatos media dependentes do tempo. Isto quer dizer que o seu conteúdo se altera à medida que o tempo avança, sem intervenção humana.

2.1.3 - Hipermedia

Realmente hipermedia é algo muito maior que a simples junção de hipertexto e multimedia. A razão é simples: hipertexto, como já foi dito, pressupõe a organização não-linear da informação, podendo-se "saltar" entre os vários pedaços de informação (texto ou gráficos); multimedia pode-se definir como a integração lógica ou física de vários formatos media no computador; hipermedia, no entanto, por ser a junção das duas filosofias, i.e., por permitir que os fragmentos de informação possam ser representados por outros formatos media, integrados no computador, tem uma funcionalidade muito mais alargada no seu todo do que se se tiver, por um lado, hipertexto, e por outro, multimedia.

Jeff Conklin, com uma definição mais arrojada, diz que em hipermedia, além dos acima citados, os elementos que estão interligados podem ser «(...) presumivelmente gostos, odores e sensações tácteis» [CON87].

Vai agora descrever-se os formatos de apresentação da informação que os vários media suportam na ordem do mais barato para o mais caro.

2.1.3.1 - Texto

O texto é o formato mais barato e mais fácil de usar pois não são necessários requisitos especiais nem de *hardware* nem de *software*. Até há *software* especializado em converter ficheiros de texto em algum formato hipermedia. Estes criam as ligações automaticamente através, por exemplo, de *parsers* de texto. Pode-se falar em dois tipos de "conversores" de texto em hipertexto. No primeiro, o utilizador selecciona, a partir do texto, as "palavras-chave" às quais são associadas ligações de hipertexto, indicando o destino dessas ligações. O segundo, e mais automático, permite ao utilizador fazer uma lista das "palavras-chave" indicando o destino das ligações a elas associadas. Depois funciona exactamente como um *parser*, i.e., o sistema vai em busca das palavras especificadas associando-lhes as devidas ligações (com opção de confirmação do utilizador).

2.1.3.2 - Gráficos, desenhos e fotografias

A fotografia é a forma mais barata de se passar de hipertexto para hipermedia (é assumido que o hipertexto pode ter texto, gráficos e desenhos). Fica mais cara que os desenhos e que os gráficos, pois é necessário um *scanner* e, além disso ocupa muito mais espaço de disco que os outros dois.

Qualquer um destes formatos media imprime uma nova dinâmica à aplicação em causa, quanto mais não seja, porque "uma imagem vale mais que mil palavras". Aplicações que classicamente só contenham texto, ficam grandemente enriquecidas com a adição de algumas imagens ou desenhos. Pense-se, por exemplo, numa aplicação sobre o código da estrada: embora haja vantagens óbvias em relação à organização da informação, pois há imensas referências entre os artigos, o facto de só existir texto não torna a ferramenta tão atraente como se houver um ou outro desenho relacionado com o assunto tratado em cada artigo. Além de o próprio desenho chamar a atenção, só o facto de haver alguma coisa diferente com cores é importante.

É possível colocar âncoras em qualquer um destes formatos media permitindo ligações para outros nodos.

2.1.3.3 - Animação

A animação adiciona algo muito importante a uma aplicação hipermedia: o **movimento**. Animação simples pode ser um pouco mais cara de adicionar do que os formatos media anteriormente citados, mas animação artística fica, sem dúvida, muito mais caro, rivalizando o preço com filmes de vídeo. De qualquer modo, é indiscutível que uma boa animação pode fazer com que o leitor não se esqueça facilmente do nodo (e respectivo conteúdo semântico) em que ela foi apresentada.

De qualquer forma, é difícil criar ligações num formato como a animação. Aliás, isto é característica de todos os formatos dependentes do tempo. Uma vez que eles mudam ao longo do tempo, os nodos que a eles se querem ligar também podem mudar, o que implica que as ligações também se alteram com o avançar do tempo. Não é impossível criá-las, mas ainda constitui objecto de grande esforço laboratorial definir em que condições e onde se devem criar essas âncoras. Contudo, se um filme animado for considerado todo um único objecto, então pode ser tratado como os outros formatos não dependentes do tempo.

2.1.3.4 - Som

O som é outro formato que pode trazer grandes benefícios a uma aplicação hipermedia. O tipo de efeito sonoro usado pode variar muito dependendo da qualidade que se deseja e, consequentemente do equipamento que se vai usar. O som precisa de *hardware* e *software* específicos. O investimento em equipamento de som vai desde as centenas aos milhares de contos. Outro problema é o espaço ocupado em disco: 15 a 20 segundos de som podem ocupar alguns *megabytes* de espaço de disco. É claro que existe *software* de compressão, mas actua à custa da perda de qualidade e, mesmo assim continua a ocupar imenso espaço. Uma das soluções é usar discos ópticos: um CD-ROM já consegue armazenar mais de 660 MB!

Como na animação, aqui o problema de colocar âncoras é crítico - ainda mais crítico, porque pura e simplesmente é impossível "clicar" em som. De qualquer modo, em aplicações hipermedia, geralmente, o som serve apenas de pano de fundo de uma qualquer apresentação visual.

2.1.3.5 - Programas de Computador

O recurso a programas externos de computador pode ser muito útil numa aplicação hipermedia, melhorando-a: podem ser usados para permitir ao leitor adicionar texto, gráficos, desenhos, ou outros formatos à aplicação. Geralmente os programas disponíveis para introduzir em nodos de aplicações hipermedia são programas gerais de entrada-saída de dados. Existem, contudo, sistemas autor que permitem chamar qualquer programa que corra no mesmo ambiente dele.

2.1.3.6 - Vídeo

O vídeo numa aplicação hipermedia é, de longe, o formato media que fica mais caro. Não se fala só de toda a tecnologia necessária, mas também do imenso espaço ocupado em disco, mesmo usando técnicas de compressão (espacial e temporal) bastante sofisticadas (taxas de compressão superiores a 100 para 1). Também aqui a utilização de CD-ROM's se reveste de grande importância, mas há ainda problemas de qualidade devido à velocidade de acesso a este tipo de discos.

Os problemas da colocação de âncoras em filmes de vídeo, são os mesmos da animação. O vídeo hipermedia interactivo ainda não é utilizado na generalidade dos sistemas hipermedia actualmente existentes no mercado.

2.2 - Utilização de Sistemas Hipermedia

2.2.1 - Porquê utilizar sistemas hipermedia?

A possibilidade de integração de vários formatos media e a forma não-linear de organização da informação, tornam este tipo de ferramentas bastante atraente tanto aos olhos dos especialistas informáticos como dos não-especialistas que, sem formação profunda, conseguem utilizá-las com bastante facilidade.

Até ao aparecimento de algumas linguagens de 4ª geração e, depois, das ferramentas de hipertexto e hipermedia, o domínio da informática e da produção de *software* estava reservado exclusivamente aos técnicos, como se esta fosse uma ciência oculta, envolta até em algum misticismo. Os sistemas hipermedia,

em particular, além de permitirem juntar e relacionar num mesmo documento pedaços de informação com características bastante distintas, conseguiram pôr à disposição dos utilizadores não-especialistas em informática todo este potencial.

Uma vez que permitem que a técnica do audiovisual seja manipulada e utilizada com uma facilidade nunca anteriormente vista, os sistemas hipermedia ganharam adeptos com muita facilidade, sendo utilizados nas mais variadas áreas desde a documentação de projectos a brochuras turísticas, passando pela formação profissional nas organizações e pela educação.

As regras de ouro do hipertexto (ponto 2.1.1.2) aplicam-se também às aplicações hipermedia. De facto, por mais que se queira, não faz sentido pôr sob a forma de aplicações hipertexto ou hipermedia (daqui para a frente só se falará em hipermedia, dado que hipertexto está aí incluído) qualquer corpo de informação que não satisfaça estas regras.

Imagine-se por exemplo, um romance... realmente existe um grande corpo de informação, mas está separado em fragmentos? Se se considerarem os capítulos ou as secções, pode-se dizer que sim... mas esta não é uma separação em termos de conceitos, de significados, mas uma separação lógica se se quiser. Há uma grande sobrecarga cognitiva entre os possíveis pedaços de informação: a relação entre as secções ou os capítulos (possíveis fragmentos) é forte demais para cada um deles poder ser considerado modular e independente de todos os outros no que toca ao seu significado. Não se pode pedir a um leitor que nos conte a história de "Os Maias", por exemplo, se lhe foi proporcionado um amontoado de secções deste romance. Isto quer dizer que ele precisa não só de ler cada uma das secções do romance, como de lê-las sequencialmente. Note-se que hipermedia implica não-linearidade na apresentação da informação - há uma grande diferença entre ter a informação apresentada de forma sequencial, mesmo utilizando vários formatos media (multimedia) e tê-la apresentada de forma não-linear, não-sequencial, onde o leitor pode escolher exactamente o que necessita e, provavelmente até estabelecer os seus próprios caminhos de navegação (hipermedia).

Quer se queira, quer não, para se ter alguns tipos de documentos de estrutura puramente linear sob a forma electrónica é necessário que eles sejam muito fortes, muito apelativos. Isto acontece devido quer a limitações dos meios electrónicos, quer a hábitos acumulados durante gerações que obrigam a que as mudanças se processem de uma forma mais lenta do que deseja, principalmente, a comunidade científica. As principais razões que levam a que ainda se prefira que muita da informação, especialmente texto, seja representada em

papel são as seguintes:

- ainda é **incómodo** estar em frente a um monitor de computador durante muito tempo;

- **não é fácil de levar** e ser utilizado em qualquer lado como um livro (quem levaria, mesmo um portátil, para férias dentro de uma mochila?);

- obriga a ter **suportes especiais**, quer de *hardware* quer de *software* para utilização;

- não é tão **personalizável** como um livro onde se pode sublinhar (conforme se quer), fazer anotações etc., tudo isto com a letra do utilizador, dando-lhe um ar mais pessoal - factor psicológico a ter em conta;

- não tem a **função social** do livro - por exemplo, pelo menos hoje em dia, ninguém (ou quase ninguém...) pensa em oferecer de presente, com dedicatória e tudo, uma versão electrónica de "A queda de um Anjo" de Camilo Castelo Branco.

- a **sensação de controlo** sobre a informação é menor do que num livro - aqui possui-se uma melhor noção do volume de informação envolvida; pode ser folheado rapidamente tendo a noção da posição de uma página em relação ao resto do livro. O livro manuseia-se, sente-se, conferindo assim, uma maior segurança a quem o manipula.

Imagine-se, porém, que se quer representar, de alguma forma, informação sobre animais mamíferos tendo em conta a sua ordem, família, género e espécie. O leitor deve poder consultar dados relativos só a determinada família, só a determinado género ou só a determinada espécie. Adicionalmente, quer-se que estando a consultar uma família, se possa aceder directamente a outra família, a todos os géneros dentro dessa família ou a todas as espécies dentro dessa família. Se se estiver a consultar determinado género deve poder aceder-se a outro género, da mesma família ou não, à família correspondente e a todas as espécies desse género. Ainda, a partir de uma espécie deve poder consultar-se outra espécie pertencente ou não aos mesmos género e família e consultar o género ou a família correspondente a essa espécie.

Aqui está um problema muito fácil de resolver usando sistemas hipermedia e bastante mais complicado usando um meio sequencial de apresentação da informação. É claro que já toda a gente viu livros sobre

animais, mas não obedecem, seguramente, aos requisitos acima especificados: toda esta informação pode ser representada sob uma forma sequencial, mas para poder ter alguma da funcionalidade aqui requerida iam ser necessários vários marcadores de página todos diferentes para não (ou para sim) causar confusão. Aqui a estruturação da informação é notoriamente não-linear. Além disso, nas aplicações hipermedia é possível conseguir um grau de interactividade e uma possibilidade de sucesso na transmissão da mensagem (através da conjugação dos vários formatos media) com os quais os sistemas tradicionais nem sequer ousam competir.

Ao longo deste trabalho, no capítulo IV, vão utilizar-se exemplos relativos a uma solução do problema acima descrito, de modo a facilitar a compreensão dos conceitos utilizados e do formalismo de modelação de aplicações hipermedia proposto.

Ver-se-á, agora, que este corpo de informação respeita as três regras do hipertexto:

1. **Está dividido em fragmentos de informação**. Note-se que esta divisão não é apenas lógica - é também de significado o que não acontecia com o exemplo do romance. Um cão é um cão, um gato é um gato... a informação relativa a cada um destes animais pode, perfeitamente, estar separada, com fronteiras delineadas, sem prejuízo para a aquisição de conhecimento do leitor - correspondem a fragmentos de informação diferentes!

2. **Os fragmentos têm relações entre si**. Realmente, há relação entre família, género e espécie, ou ainda entre as várias espécies (relações de simbiose, por exemplo), ou outro tipo qualquer de relações. Estas não são apenas sequenciais; são antes relações não-lineares entre os vários fragmentos de informação.

3. **O leitor só precisa de um fragmento de cada vez** - estes são modulares. Note-se que nenhuma das relações acima mencionadas é tão forte que seja preciso ter já acedido, ou aceder, a um fragmento de informação para perceber o outro. Para perceber a informação relativa a um cão, o leitor não precisa de saber o que é um gato, um hipopótamo ou um elefante. Além disso, para perceber o que é um cão, não é preciso que o leitor saiba nada relativo à sua família ou ao seu género, embora isso possa constituir informação suplementar. Por outras palavras, não há sobrecarga cognitiva.

Os sistemas hipermedia não são, de modo nenhum, remédio para todos os males. Não resolvem, pelo menos para já, todos os problemas de gestão e apresentação da informação, nem são adequados a todos os casos que aparecem neste domínio. As aplicações hipermedia fazem uma coisa, e fazem-na bem: permitem ao leitor escolher os seus próprios caminhos através de um grande corpo de informação, consoante as suas necessidades, para daí retirarem o conhecimento que realmente lhes faz falta. Alguns sistemas permitem, também, criar aplicações em que o leitor tenha um papel ainda mais activo modificando, personalizando e criando diferentes hipervisões[1] sobre a hiperbase (base documental hipermedia).

Numa aplicação hipermedia os leitores não necessitam de fazer uma procura sequencial sobre o assunto que lhes interessa: o hiperdocumento é desenhado de forma a que os utilizadores possam retirar a informação de que necessitam utilizando a sua própria forma de busca através de ligações pré-estabelecidas pelo autor do hiperdocumento, ou através de ligações estabelecidas pelos próprios leitores. Esta será quase sempre a maneira mais rápida de busca da informação e a mais confortável para os leitores.

As figuras seguintes mostram uma parte de uma hiperbase e duas hipervisões que se podem criar sobre ela:

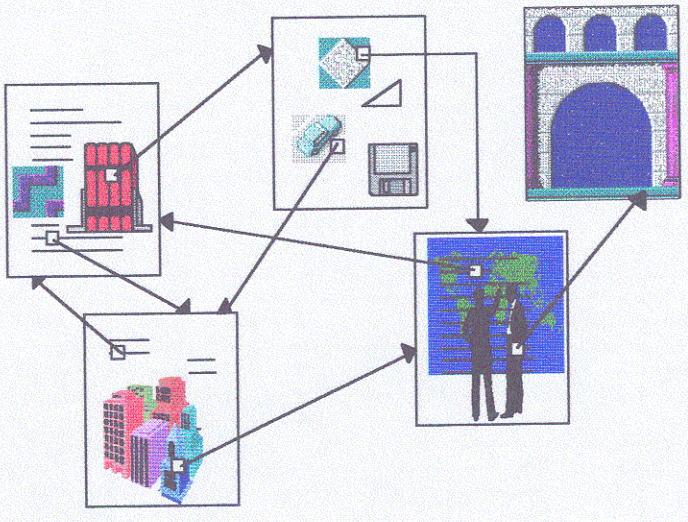


Fig. 2.9 - Hiperbase original.

[1] Hipervisão é uma visão da hiperbase. A própria hiperbase original (com todas as suas inter-relações) é uma hipervisão. Podem criar-se outras hipervisões sobre esta de acordo com alguns parâmetros como o perfil do leitor, a sua história navigacional, etc. Podem, também, criar-se hipervisões de acordo com as necessidades informativas correntes de determinado leitor.

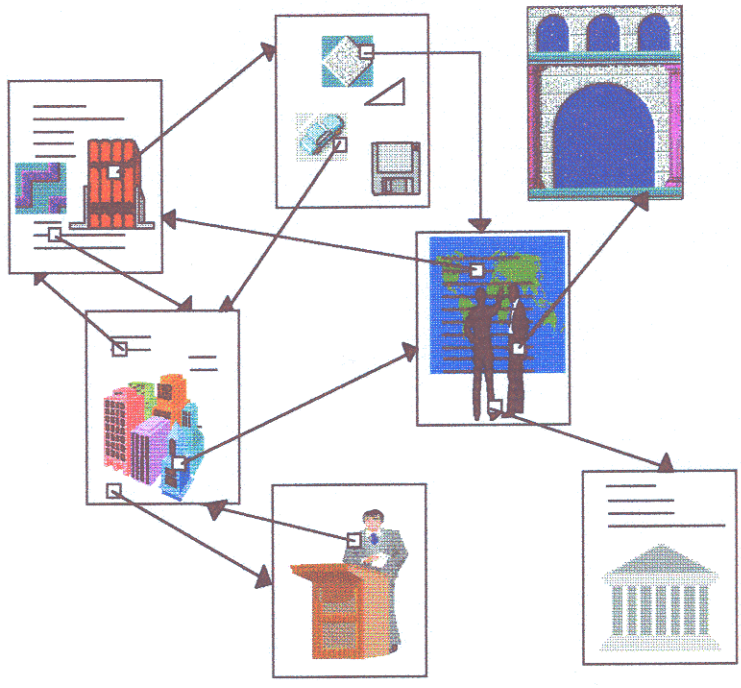


Fig. 2.10 - Uma hipervisão sobre a hiperbase original na qual o utilizador criou mais dois nodos e ligações de e para eles.

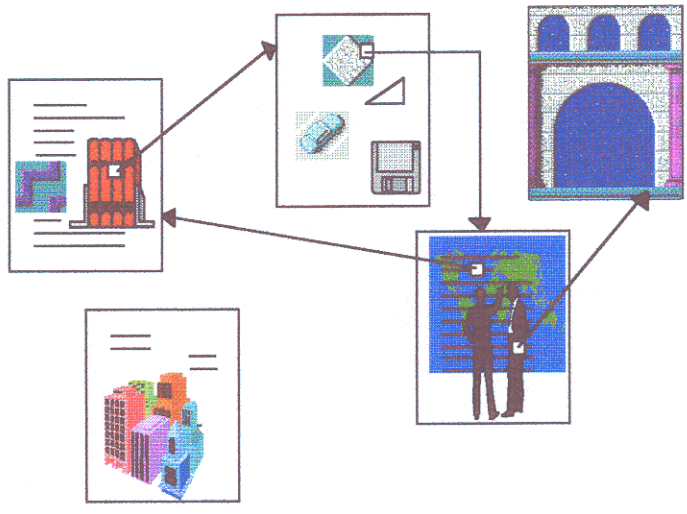


Fig. 2.11 - Hipervisão sobre a hiperbase original de onde o utilizador retirou ligações.

A junção de hipermedia com outras abordagens de organização e gestão da informação como o paradigma orientado ao objecto ou a inteligência artificial traz inúmeras potencialidades.

Boyle e Snell [BS90] defendem o entrosamento entre hipermedia e inteligência artificial, apresentando o sistema SINS *(Semistructured Intelligent Navigation System)*. Dizem que este sistema mostra como as técnicas de bases de conhecimento, podem ser usadas para tornar a navegação mais facilitada através da criação de ligações virtuais.

Se considerarmos hipermedia não só a facilidade de organização não-linear da informação associada ao facto de esta poder estar representado sob vários formatos diferentes, mas também possuindo uma base de dados documental, com atributos em cada um dos fragmentos de informação e com a execução de processos, então existe a possibilidade de fazer **interrogações** (*queries*) sobre essa base de dados e, mais ainda, de estabelecer ligações entre os nodos de uma forma **dinâmica**, i.e., estabelecidos em *runtime*.

Chen et al. [CHE90], dizem que «os utilizadores deveriam poder fazer *queries* tanto sobre o conteúdo como sobre a estrutura, para descreverem a informação de que necessitam». Além disso, deveriam também poder escolher o estilo de apresentação da informação, outra razão pela qual esse autores acham que os sistemas hipermedia necessitam de se "misturar" com o **paradigma orientado ao objecto.**

2.2.2 - Vantagens / Desvantagens dos sistemas Hipermedia

De seguida apresentam-se algumas das vantagens e desvantagens decorrentes da utilização de sistemas hipermedia.

Desvantagens:

- Dificuldade por parte do autor da aplicação em **dividir um pensamento** ou um segmento de informação num nodo;

- Risco de fazer uma aplicação aparentemente simples, mas com ligações a mais produzindo, assim, uma rede de ligações insustentável. É o que Janet Fiderio [FID88] chama de "***spaguetti code***";

- Dificuldades de **manutenção**, muito especialmente se a aplicação não tiver sido bem projectada. Quando, por exemplo, se apaga um nodo e não se apagam as âncoras e ligações respectivas que têm esse nodo como destino criam-se ligações incontroláveis;

- **Desorientação** do leitor - há que haver um compromisso entre o controlo fornecido e a facilidade de orientação do leitor: é que quanto maior é o controlo fornecido (maior número de ligações por nodo), maior é o risco de o leitor se "embrenhar" na rede de informação existente, aumentando a sua dificuldade de "sair" de lá;

- Dificuldade do leitor em ter uma **noção global** do hiperdocumento.

Vantagens:

- Maiores facilidades para **expressão / compreensão** da mensagem, devido à utilização de diferentes formatos media;

- **Maior impacto** causado no(s) leitor(es) do hiperdocumento;

- Possibilidade de **re-utilização da informação**, pois uma vez que é subjacente uma estruturação não-linear da informação, os nodos não necessitam de se repetir: quando se quer referenciar um nodo já existente, basta criar uma ligação cujo destino seja esse nodo. Além disso, pode haver re-aproveitamento de nodos para outros hiperdocumentos.

- Possibilidade de fornecer acesso a **grandes quantidades de informação** armazenada num só suporte físico e de pequenas dimensões;

- **Interactividade**;

- Possibilidade de **adaptação do documento** ao leitor.

Convém, ainda, acrescentar algumas coisas em relação às desvantagens atrás mencionadas, particularmente o que tem a ver com o excesso de ligações num hiperdocumento e com a desorientação do leitor. Os leitores não devem, em princípio, ser obrigados a navegar no hiperdocumento de um modo pré-definido: é por isso que muitos sistemas de hipermedia permitem criar aplicações em que o leitor pode modificar a rede e, assim, conforme se disse atrás, criar as suas próprias hipervisões de modo a personalizá-lo. Outra coisa a evitar é criar um excessivo número de ligações entre os nodos da hiperbase: se tudo estiver ligado a tudo, qual é a vantagem de ter ligações? Na verdade, existe uma grande desvantagem: a desorientação provocada no leitor! Por vezes, este problema pode acontecer quando se dá a possibilidade aos leitores de modificarem a hiperbase, mas neste caso ou se criam previamente restrições ao número de ligações por nodo (ou que um nodo tem como destino, caso seja possível) ou, então, confia-se no bom senso do leitor... é o eterno conflito entre controlo e flexibilidade!

Em relação ao problema da desorientação, a solução passa por fornecer, de algum modo, informação ao leitor sobre o "sítio onde está" na hiperbase. Uma das formas mais simples de o conseguir é através de uma apresentação gráfica da aplicação que possua um sinal do género "Você está aqui!". Em alguns sistemas de hipertexto existe um nodo especial, chamado ***browser* gráfico**, que contém um diagrama estrutural de uma rede de nodos. Esta importante ferramenta de ajuda ao utilizador mostra «parte ou todo o hiperdocumento como um grafo, providenciando uma medida importante de pistas contextuais e espaciais para complementar o modelo do utilizador com informação, especificando que nodos ele está a ver e como estão relacionados uns com os outros no grafo» [CON87]. O *browser* tanto pode ser usado para o utilizador se auto-orientar, como para se mover directamente para uma área ou zona do hiperdocumento, seleccionando determinado ponto do écran com um rato, ou ainda para ajudar o utilizador a decidir sobre a próxima acção a ser tomada. A figura seguinte mostra um exemplo da visualização de um *browser* gráfico e a sua relação com a base de dados subjacente ao hipertexto:

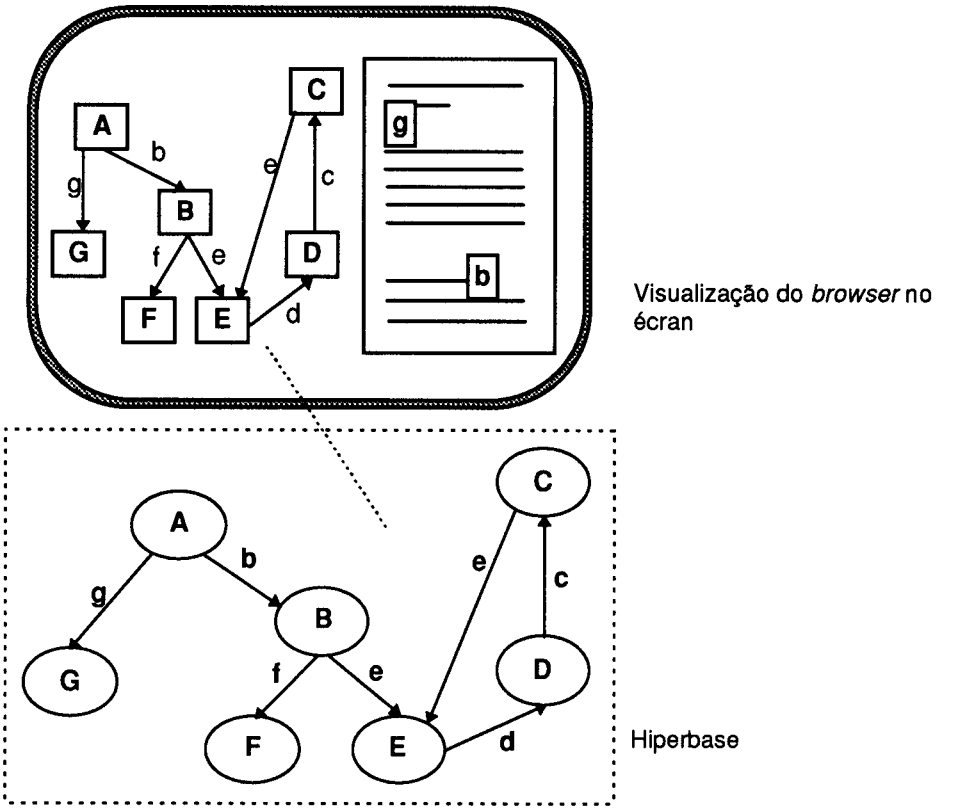


Fig. 2.12 - Um *browser* gráfico e a sua relação com a base documental de hipertexto (hiperbase).

De modo a facilitar o uso do hiperdocumento, existem várias outras facilidades providenciadas por alguns sistemas de hipertexto, entre as quais se podem salientar os **caminhos** (*paths*) e os **filtros de visão** (*viewing filters*). Os caminhos (*paths*) «são caminhos por defeito através de uma base de dados; eles guiam ou direccionam o utilizador através de uma lista ordenada de nodos» [FID88]. Um filtro de visão (*viewing filter*) é um filtro da informação do sistema, i.e., suprime o detalhe apresentando unicamente a informação relevante. «Filtrando o nível mais baixo dos conteúdos de um nodo, pode-se passar rapidamente através de uma rede à procura da informação de que se necessita» [FID88]. No fundo, nem os caminhos nem os filtros de visão passam de hipervisões previamente definidas.

2.2.3 - Aplicações Hipermedia

James Martin [MAR92] distingue seis tipos diferentes de sistemas de hiperdocumentos:

- **Sistemas de Macroliteratura** - Consiste numa arquitectura distribuída de bases documentais com documentos científicos e literários, críticas e outras colaborações inter-relacionados através de ligações hipermedia (hiperligações).

- **Colecções de documentos** - Edições de colecções de documentos com um índice comum, relacionados através de hiperligações quer entre eles, quer entre eles e glossários, índices ou tutoriais.

- **Ferramentas de exploração de problemas** - São ferramentas que permitem ao autor ligar as suas ideias à medida que vão surgindo.

- **Conjuntos inter-relacionados de informações** - Casos em que informação referente a assuntos diversos, se junta por ser mais útil deste modo. Este inter-relacionamento da informação é feito através de hiperligações.

- **Documentos grandes e complexos** - Aqui James Martin dá o exemplo de normas de procedimentos, manuais de qualidade, manuais de manutenção de equipamento e outros tipos de documentos mantidos pelas grandes organizações que, quer pelo espaço ocupado, quer pela velocidade de acesso ao conteúdo, deveriam estar armazenados sob a forma electrónica (talvez em CD-ROM), de fácil e rápido acesso e, inter-relacionados segundo as referências cruzadas entre eles.

- **Documentos inteligentes** - É o nome que James Martin dá ao cruzamento entre hipermedia e **inteligência artificial**. Alguns exemplos são sistemas para treino assistido por computador, manuais de operação ou hiperdocumentos de diagnóstico e reparos.

Já há ferramentas no mercado que possuem, além de algumas das anteriormente mencionadas, estruturas de dados mais ou menos complexas e uma linguagem de programação embutida, muitas vezes próxima da linguagem natural. Isto permite a realização de aplicações mais complexas e mais cuidadas. James Martin também não menciona aplicações hipermedia baseadas, ou relacionadas com, o paradigma orientado-ao-

objecto: este torna-se cada vez mais usado nestas situações, não só pelas facilidades de desenvolvimento (*authoring*) e manutenção que proporciona, como pela filosofia de organização da informação e, também por características como a possibilidade de realização de interrogações (*queries*), de partilha de objectos ou a possibilidade de re-utilização da informação.

O desenvolvimento de qualquer tipo de *software* pressupõe **interdisciplinaridade** na constituição da equipa, pois além de ser necessário saber programar e de saber quais os conteúdos informativos, é preciso saber como deve ser apresentado o produto nas suas mais variadas perspectivas desde *layout* de écrans e janelas a cores a usar. Em hipermedia esta necessidade é ainda mais premente devido ao facto de muito do poder destas ferramentas se centrar na apresentação, no *interface* com o utilizador, e na boa utilização tanto dos formatos media envolvidos, como das fracções de conhecimento a interligar. Além de serem necessários especialistas nos vários formatos media, também será preciso ter peritos em comunicação social, informáticos e, claro, peritos no assunto sobre o qual a aplicação trata. É claro que tudo isto depende da aplicação que se tiver em mãos e dos objectivos que se quiserem atingir com a sua realização. Se, por exemplo, um professor quiser dar uma aula, ou um gestor quiser fazer uma apresentação, é claro que, em princípio, ele consegue realizar a aplicação sozinho. Agora, se for uma aplicação mais complexa, na qual seja necessário profissionalismo em várias áreas, então deve tentar-se o mais possível constituir uma equipa pluridisciplinar.

David H. Jonassen [JON90] defende que, se partirmos do princípio que a actividade de aprender passa pelo mapeamento da **estrutura de conhecimento** do professor no aluno, então as ferramentas de hipermedia são ideais para este processo. Isto, porque depois de se obter a estrutura de conhecimento do professor, é possível representá-la usando estas ferramentas.

Conforme já se disse atrás, hipermedia **não é remédio para todos os males**. Há que ter consciência disto para não se cair no erro de forçar determinado corpo de informação a ser representado sob a forma hipermedia. Hipermedia requer não-linearidade na organização da informação - é necessário ter vários fragmentos de informação com relações entre si. Realmente, se as três regras de ouro do hipertexto ditadas por Ben Shneiderman (ponto 2.1.1.2) não se verificarem, nem sequer vale a pena pensar numa aplicação hipertexto / hipermedia. Não convém esquecer que «(...) ironicamente, muitas vezes, é o desejo de imitar o papel que resulta em hipertexto pobre»[RIT89].

McKnight et al. (McN90] fizeram um estudo comparativo da aquisição de informação por grupos mais ou menos homogéneos de utilizadores, usando formatos lineares (papel e um processador de texto) e formatos

hipertexto (HyperTIES e HyperCard) com o mesmo conteúdo. Testaram, para os vários leitores sujeitos à experiência:

- estimativas do **tamanho do documento**
- **velocidade** no cumprimento de tarefas
- **precisão** nas respostas e
- **navegação** - tempo gasto vendo e alternando entre os conteúdos e o índice

Como resultados gerais, obtiveram que os sujeitos que tiveram acesso aos formatos lineares se portaram melhor que aqueles que acederam a formatos hipertexto. No primeiro *item* testado (estimativas do tamanho do documento) o resultado dos sujeitos que trabalhavam com formatos de hipertexto foi completamente desastroso. Notaram, ainda, que «sujeitos em condições de hipertexto gastavam consideravelmente mais tempo a ver as listas de índices e conteúdos do que os sujeitos em condições lineares, mas tinham menos sucesso a responder correctamente às perguntas».

É claro que há factores externos que influenciam os resultados. Entre eles, podem ser citados: a variação da familiaridade dos sujeitos com o assunto em questão; a variação das velocidades de leitura dos sujeitos; a determinação dos sujeitos em continuar a procurar até obterem uma resposta; e, muito importante, a familiaridade com o meio no qual é apresentada a informação. Este último factor é importante porque os sujeitos conheciam diversos *packages* de *software*, mas nenhum de hipertexto não estando, por isso, habituados àquela filosofia de busca de informação (só lhes foi dada meia hora para treinarem).

De qualquer modo, embora os factores externos tenham influenciado os resultados, esta experiência não é de menosprezar, servindo de aviso a todos aqueles que quiserem utilizar estas ferramentas indiscriminadamente: elas não substituem o papel nem a forma linear de organização da informação, **complementam-os!** São muito úteis e indicadas para determinadas situações, mas não para todas. Não são, de modo nenhum, uma panaceia.

III - O ESTADO DA ARTE

Neste capítulo vai-se dar uma visão geral sobre o estado da arte no que toca à modelação de aplicações hipermedia. Duas das abordagens de que se vai falar (pontos 3.3 e 3.4) têm como base formalismos matemáticos. Todas elas foram, ou estão a ser implementadas em sistemas criados para o efeito (estes não são aqui descritos). São fruto de trabalho de vários anos de equipas de especialistas na matéria estando ainda em laboratório a ser testadas e alteradas.

Talvez por tudo o que tem a ver com hipermedia e multimedia esteja sujeito a grandes alterações em curtos espaços de tempo, principalmente devido às inovações tecnológicas, pouca standardização (quer a nível de *hardware* quer a nível de *software*) e, talvez porque têm a ver com filosofias implementadas (no circuito comercial) ainda muito recentemente, existe ainda muito pouca coisa escrita sobre técnicas de modelação. Afinal, como em todos os outros campos, primeiro vem a novidade das ferramentas e as pessoas experimentam-nas entusiasticamente em vários domínios; depois, já com alguma experiência, sente-se a necessidade de um suporte metodológico e de uma notação que exprima uma linguagem comum de comunicação - é nessa altura que começam a surgir as primeiras abordagens, um pouco a "desbravar caminho".

Em relação à modelação de aplicações hipermedia, é nesta fase que nos encontramos. É, por isso, muito difícil encontrar bibliografia credível nesta área. Aqui apresentam-se quatro abordagens para modelação de aplicações hipermedia e uma para modelação de documentos multimedia. Esta última foi introduzida por se achar com bastante interesse. No final, faz-se uma breve discussão sobre as abordagens apresentadas.

3.1 - HDM

O HDM (**H**ypertext **D**esign **M**odel) é um modelo de desenho para aplicações hipertexto que está a ser desenvolvido no âmbito de um projecto mais alargado chamado HYTEA (Hypertext Authoring). Este projecto está a ser levado a cabo por um consórcio europeu (Itália, Alemanha e Grécia) e o seu objectivo prende-se com a implementação de um conjunto de ferramentas de apoio ao desenvolvimento de aplicações hipertexto/hipermedia grandes e complexas.

Segundo os seus autores, «uma aplicação HYTEA é feita através de um esquema HDM e uma hiperbase HDM» [GAR91]. Afirmam ainda que o modelo conseguido já mostrou ser passível de «translação, quer manualmente quer através de compilação, para um modelo de nodos-e-ligações». Deste modo, a aplicação resultante poderia ser usada em variadas ferramentas existentes no mercado.

Os aspectos mais importantes deste modelo têm a ver, essencialmente, com a noção de entidades, componentes, perspectivas, ligações (estruturais, da aplicação ou da perspectiva), esquema HDM e semântica de *browsing*. De seguida, passa-se a uma definição de cada um destes conceitos.

Uma **entidade** é então definida como «um objecto concreto ou conceptual do mundo real no domínio em que é relevante para uma aplicação»[GAR91]. As entidades pertencem a um tipo.

Um **componente** é «um pedaço de informação descrevendo uma parte de uma entidade»[GAR91]. Estes são agrupados na entidade segundo uma hierarquia arbitrária de acordo com a aplicação. É necessário ter em atenção que todos os componentes de uma entidade devem ser homogéneos nas suas perspectivas, i.e. componentes com perspectivas diferentes não poderão pertencer à mesma entidade. Para tornear este problema, os autores introduziram posteriormente a noção de "tipo entidade agregada" (*agregate entity type*) do qual se falará mais tarde.

A noção de **perspectiva** está associada ao facto de a informação poder ser apresentada de várias formas diferentes (por exemplo, a cidade de Lisboa pode ser apresentada com uma descrição escrita ou uma sequência de imagens da cidade ou ainda algumas canções). Um componente pode ter várias perspectivas sendo cada uma delas denominada de **unidade**. A informação está realmente contida nos corpos das unidades.

Um **tipo de entidade** (*entity type*) agrupa entidades tendo propriedades comuns, nomeadamente usando um conjunto de perspectivas em comum (o conjunto de perspectivas associado a um tipo de entidade refere-se unicamente à intersecção dos conjuntos das perspectivas das várias entidades pertencentes a esse tipo), que são decompostas em componentes de acordo com o mesmo critério e que se relacionam com entidades de outros tipos da mesma forma. É neste contexto que é introduzido um novo conceito: o de perspectiva por defeito ou *default perspective*, que implica que todas as entidades de um dado tipo têm pelo menos esta perspectiva.

São definidos três espécies de *links* ou **ligações**: **estruturais**, **da aplicação** ou **de perspectiva**. As ligações estruturais ligam componentes pertencentes à mesma hierarquia. As ligações da aplicação dizem respeito às relações semânticas no domínio em causa (entre entidades). As ligações de perspectiva dão a possibilidade ao leitor de se mover entre diferentes perspectivas do mesmo componente.

Também existem **tipos de ligações**, mas somente para as ligações da aplicação (*application link types*). Estes são definidos como «um conjunto de instâncias de ligações cujas entidades-fonte e destino são do mesmo tipo de entidade, respectivamente» [GAR91].

Surge, então, a noção de **esquema** como o nível mais elevado de abstracção no HDM pois caracteriza uma classe de aplicações hipertexto num dado domínio. Um esquema é definido como «um conjunto de definições de tipos de entidades e tipos de ligações» [GAR91]. Em determinado domínio, uma dada aplicação é especificada através da instanciação do esquema, i.e., «dando as entidades e as ligações como instâncias dos tipos definidos no esquema» [GAR91]. Deste modo, as várias instâncias de um esquema (aplicações) partilham a estrutura global, mas não a local tanto em termos de ligações como em termos das perspectivas de cada entidade. Uma dada instância de determinado esquema caracteriza os aspectos estáticos de uma aplicação; a definição dos aspectos dinâmicos seria realizada através da semântica de *browsing*.

A **semântica de *browsing*** «fornece o comportamento em *runtime* da aplicação» [GAR91]. Dada uma semântica de *browsing* particular, deverá ser possível passar da especificação HDM para estruturas de implementação de alguns sistemas de hipertexto existentes no mercado. Apesar de o HDM ainda não possuir nenhum conjunto de primitivas de especificação de uma semântica de *browsing*, existe uma por defeito que é compatível com vários sistemas de hipertexto existentes no mercado. Está a ser feita investigação no sentido de se criar uma semântica de *browsing* simples, o que pode ser conseguido através do uso de formalismos como redes Petri.

Desenvolvimentos recentes / futuros

Existem ainda outros conjuntos de características que foram desenvolvidas muito recentemente, estão em fase de desenvolvimento ou estão pensadas como adições a realizar.

Geralmente as aplicações são constituídas pela hiperbase de documentos e pelos seus mecanismos de acesso. Para já a preocupação centrou-se na definição da hiperbase por ela ser o «coração de uma aplicação» [GAR91], estando agora os autores preocupados com as primitivas HDM que modelam os mecanismos de acesso. Os autores identificam 4 tipos diferentes de mecanismos de acesso: *Path mechanisms, Indexes, Queries* e *Associative access* . Os *path mechanisms* dizem respeito a estruturas em rede, das quais o modelo mais simples será uma sequência linear ordenada; os *indexes* correspondem, como o próprio nome indica a índices, i.e., cada um corresponde a um menu com opções - «o resultado da selecção tanto pode ser uma entidade como um caminho linear limitado»[GAR91]; tanto as *queries* como o *Associative access* dizem respeito à ligação dinâmica da informação de acordo com as directivas do leitor.

Em relação à ligação com o paradigma orientado ao objecto, foram adicionadas ao modelo original primitivas para lidar com a noção de agregação de entidades, i.e., agora pode-se falar de **tipos de entidade agregada**. Antes as entidades eram definidas através de um conjunto de componentes hierarquicamente organizados, com uma estrutura homogénea. «Um tipo de entidade agregada é feito de várias sub-entidades componentes. Cada sub-entidade componente tem um nome e um tipo (que pode ser outra vez uma hierarquia agregada ou homogénea» [GAR91]. Estão ainda a ser adicionadas primitivas para lidar com a noção de **herança** a nível de tipos de entidades.

Tendo em conta que a sempre citada possibilidade de partilha de informação é uma das grandes vantagens e um dos grandes factores de atractividade deste tipo de aplicações, está a ser considerada a partilha não só de corpos de unidades (já antes existente), mas «de tudo, i.e., partilha de componentes, corpos, ligações estruturais e ligações da aplicação». Segundo os autores, apesar desta aproximação poder dar origem a situações complicadas, parece ser a melhor.

Em relação ao que os autores chamam de «formatos media activos», i.e., os formatos media cujos conteúdos se modificam sem intervenção de leitor (vulgarmente chamados de formatos media **dependentes do tempo**), não são introduzidas alterações de grande monta. Assim, adicionam-se novos corpos às unidades de formatos

media que são descritos de uma forma abstracta como objectos. A edição destes objectos é considerada como pertencendo a uma fase posterior de refinamento e não será considerada no âmbito do HDM.

Os autores propõem-se, ainda, a introduzir a possibilidade de o leitor ter acesso a diferentes hipervisões da mesma hiperbase de modo a que a mesma aplicação possa ser usada em situações distintas (de acordo com tarefas, perfis, etc.).

Como outra inovação a realizar, é ainda proposta a permissão de se especificar, ao nível do esquema, qualquer estrutura padrão geral que as várias instâncias de um tipo de entidade têm de seguir. Assim, será possível a «geração automática de ligações de navegação tais como ligar primeiro a introdução e depois cada secção»[GAR91].

3.2 - Livro Electrónico Hipermedia

Philip Barker, notando algumas das grandes limitações dos livros convencionais para suportar a aprendizagem interactiva, apoiou-se na metáfora do livro electrónico para discutir a utilização de um modelo orientado-ao-objecto para o desenho de estruturas de páginas hipermedia reactivas, i.e., que podem possuir ligações não-lineares a outras páginas, ligações essas que são estabelecidas após a existência de um evento externo como, por exemplo clicar num botão com o rato. Estas outras páginas podem conter objectos de texto, gráficos, imagens, som vídeo, animação etc., sendo também elas reactivas. Segundo ele «a utilidade e eficácia da metáfora pode ser melhorada significativamente quando é aumentada com o paradigma hipermedia»[BAR92].

O primeiro modelo apresentado por Barker é o modelo mais básico de um livro electrónico que pode servir de ponto de partida para a criação de novos modelos mais completos. Isto depende muito tanto dos conteúdos a serem apresentados como do objectivo do próprio livro a construir. Este modelo é o seguinte:

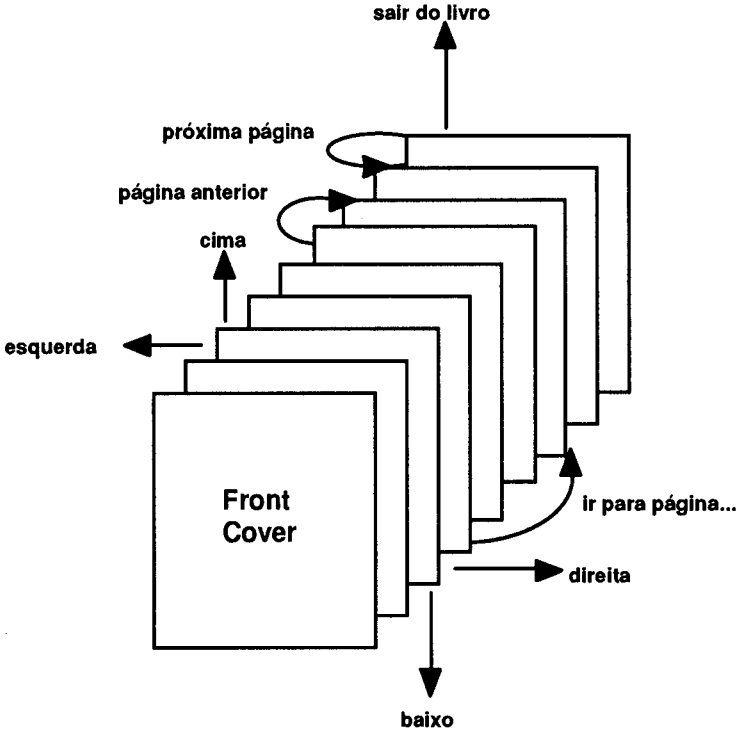


Fig. 3.1 - Modelo básico de um livro hipermedia.

Conforme se pode ver nesta figura, cada página de um livro poderá ter um conjunto de operações de controlo que lhe estão associadas: *próxima página*, *página anterior*, *ir para página N* e *sair do livro*. Por ser tão fácil introduzir *hot spots* (lugares no écran que apresentam determinado comportamento após algum evento - geralmente comportam-se como botões. Por exemplo: palavras de texto que quando clicadas permitem mudar para outra página) ou botões que executem estas operações, é que qualquer página de um livro hipermedia pode ser reactiva, i.e., «pode ter ligações de hipertexto, hiperimagem e hipersom[2] »[BAR92].

Segundo Barker, a estrutura da maior parte dos livros electrónicos consiste em quatro tipos básicos de páginas: Capa, Índice de Conteúdos, Raiz e Destino. As páginas Raiz são simplesmente páginas que podem ser alcançadas directamente a partir da página do Índice de um dado livro. Do mesmo modo, acede-se às páginas-destino a partir de um nodo Raiz embora, em certos casos particulares, se possa aceder a elas directamente .

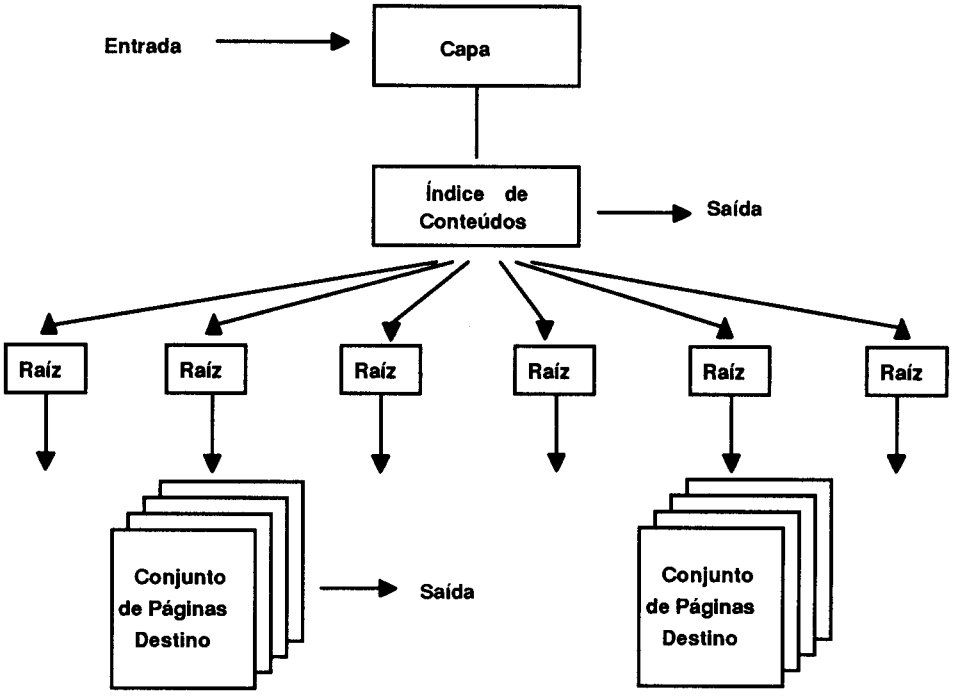


Fig. 3.2 - Estrutura dos livros hipermedia.

[2] «hyperimage» e «hypersound» no original.

Uma página (conforme se mostra na figura que se segue) «é composta por três tipos elementares de objectos: os que facilitam o controlo de página, objectos do contexto, e um *array* espacial de referências reactivas a outros objectos hipermedia»[BAR92]. Os objectos que facilitam o controlo de página são os que encerram operações do tipo "página seguinte", "página anterior", "ir para a página ...", etc. Os objectos de contexto são aqueles que são específicos de determinada página de acordo com o significado da informação que ela apresenta. Por último, o *array* espacial de objectos contém referências a objectos cujas classes são comuns a várias páginas hipermedia.

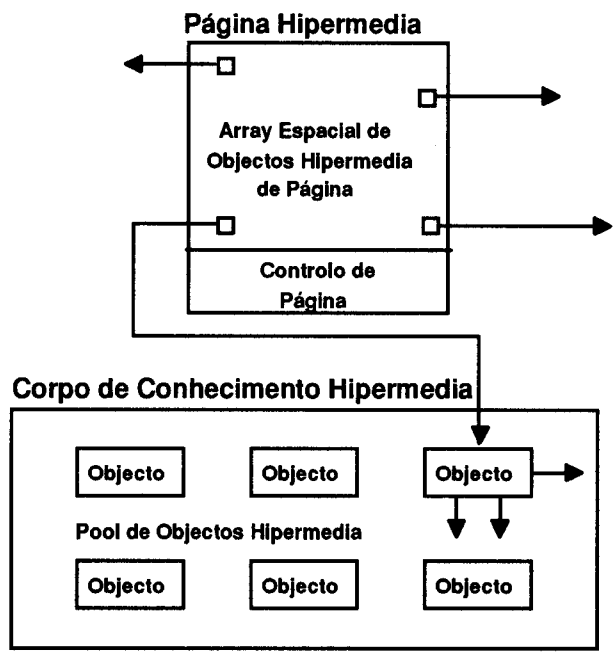


Fig. 3.3 - Composição de uma página hipermedia.

Todos os objectos de uma página hipermedia para um dado livro existem dentro de um espaço comum (o Corpo de Conhecimento que se mostra na figura anterior), possuindo cada um um nome único. Deste modo não há a necessidade de duplicação de um objecto hipermedia como resultado de diferentes páginas fazerem referência a ele.

A estrutura dos objectos, dentro do Corpo de Conhecimento, é representada na figura seguinte:



Fig. 3.4 - Estrutura de um objecto hipermedia.

Todos os objectos têm a mesma estrutura que especifica o que vai ser mostrado (atributos) e também os métodos a que eles respondem tanto em interacção com o utilizador como em interacção com outros objectos. Na figura, os quadrados com as setas dizem respeito a referências para objectos de uma página hipermedia. As setas «apontam para os objectos (no espaço comum) que são activados quando um utilizador segue as referências hipermedia em dada página» [BAR92].

Á medida que o leitor vai fazendo as suas explorações através do Corpo de Conhecimento, pode utilizar alguns dos comandos fornecidos pelo *software* de gestão do sistema que implementa o livro hipermedia electrónico. Algumas das funções realizadas por este gestor são feitas automaticamente de modo a poder ajudar o leitor em situações futuras.

Existem vários mecanismos de ajuda à navegação do leitor, como por exemplo o *back-track* - é mantida uma lista de todos os passos escolhidos pelo leitor de modo a que ele possa voltar para trás se e quando quiser. Outro dos mecanismos providenciados é o *mark-* permite que o leitor marque «objectos de interesse

particular de modo a que possam ser "chamados" e examinados mais tarde»[BAR92]. É criada uma lista LIFO de todos os nodos que o leitor quer revisitar. Existem, ainda, mais comandos implementados e outros que estão em vias de o ser.

Para já a maior parte das experiências foram feitas com textos. No entanto, o autor diz que vão começar a trabalhar com outros formatos media. Adicionalmente, estão a começar a utilizar discos ópticos como meio de armazenamento da informação existente no Corpo de Conhecimento.

Resta somente salientar que o autor, em jeito de conclusão, diz que «uma das aproximações mais úteis à implementação da metáfora do livro electrónico encontradas (até à data) é baseada no paradigma de desenho orientado ao objecto» [BAR92].

3.3 - Modelo Baseado no Hipergrafo

Frank Tompa, assumindo que as bases de dados documentais (que ele chama orientadas-à-página) não podem ser esquematizadas do mesmo modo que as bases de dados relacionais (orientadas-ao-*record*) e que o modelo do grafo orientado é insuficiente, para as modelar apresenta um formalismo baseado no modelo do **hipergrafo**.

No modelo baseado em grafos orientados pesados, o estado de uma base de dados é um triplo:

<N,L,E>

com,

- N = um conjunto finito de **nodos**,
- L = um conjunto de **etiquetas** ou *labels* e
- E : N × L →N = um conjunto de **arcos orientados etiquetados**.

No entanto, nem só os aspectos estáticos devem ser modelados - aspectos ligados à edição (inserção, remoção etc.) e à navegação, assim como outros aspectos dinâmicos, devem constar do modelo do hiperdocumento. Para a navegação, por exemplo, é necessária a noção de posição corrente, ou estado do utilizador e um conjunto de destinos possíveis. Tompa define o **estado do utilizador** como sendo «o estado de uma base de dados <N, L, E> juntamente com a posição M ∈ N, que serve como um marcador a designar um nodo corrente»[TOM89].

Aqui um nodo é entendido como correspondendo apenas a um objecto de informação, i.e., pode corresponder a um texto, um gráfico, uma imagem, um trecho musical, um segmento de vídeo, etc., mas nunca a mais do que uma destas coisas ao mesmo tempo. Um nodo não corresponde a um pedaço modular completo de informação; frequentemente é necessária a apresentação de mais do que um nodo no écran, para se apresentar a informação pretendida de uma forma inteligível. Um écran de informação (que corresponde a um pedaço modular completo de informação) precisa quase sempre de ter activos mais do que um nodo, pelo

que é necessário manter informação sobre os nodos que estão correntemente a ser visualizados, de forma a que se possa navegar na rede. Por isso é que é preciso "marcar" o hipertexto obtendo, assim, o estado do utilizador.

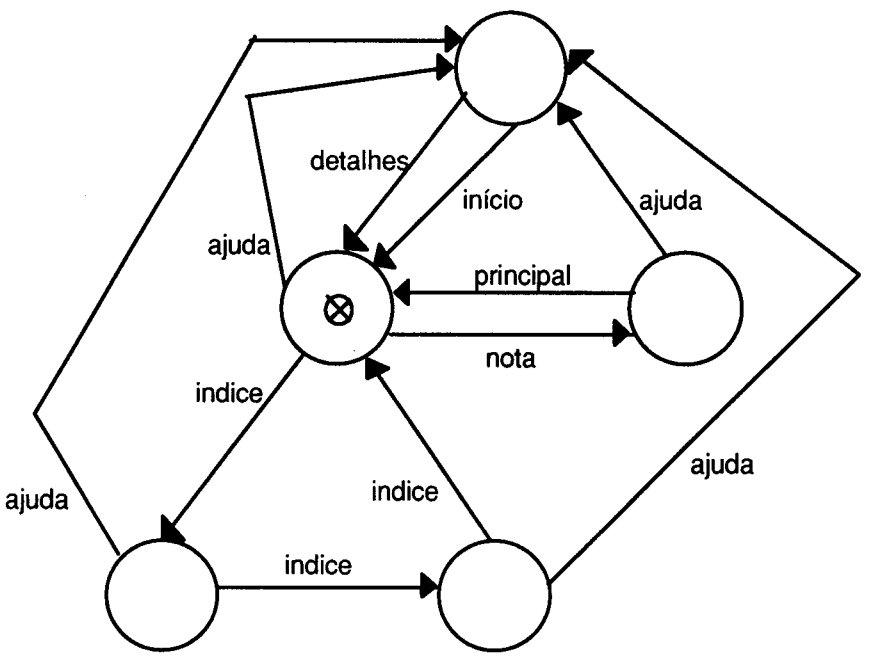


Fig. 3.5 - Exemplo do estado do utilizador usando o modelo do grafo orientado pesado.

O modelo do grafo orientado pesado é muito simples não sendo previstos casos de, por exemplo, padrões de partilha de sub-componentes que na prática ocorrem nas hiperbases. A separação entre a estrutura e o conteúdo também é inadequada: «uma separação forte entre a estrutura e o conteúdo permite várias estruturas serem sobrepostas sob o mesmo conjunto de páginas ou uma página ser partilhada entre vários nodos dentro de uma única estrutura» [TOM89]. Estas características além de facilitarem a manutenção, ajudam a compreender mais facilmente a rede permitindo, por isso, uma navegação mais fiável.

O modelo baseado no hipergrafo facilita a separação entre a estrutura e o conteúdo e incorpora vários níveis de estruturas personalizadas e do sistema.

O estado de uma base de dados é modelado pelo sêxtuplo:

<N,P,R,V,L,E>

com,

- N = um conjunto finito de **nodos**
- P = um conjunto de **páginas** que são os objectos de dados no sistema
- R = um conjunto de **leitores** para interpretar as páginas hipermedia
- V: N $\rightarrow$ P = é uma função que **mapeia nodos em conteúdos**
- L = um conjunto de **etiquetas** ou *labels*
- E : $2^N \times 2^L \rightarrow 2^N$ = um conjunto de **hiperarcos**. Cada hiperarco é equivalente a um conjunto de arcos direccionados etiquetados relacionados uns com os outros.

Uma página é um objecto que tem um valor atómico. Uma página pode ser um pouco de texto, um ficheiro sequencial de códigos gráficos, ou códigos fotográficos, um programa executável, etc. , i.e., pode corresponder informação representada em qualquer formato media, ou *software*. Note-se, no entanto, que a uma página corresponde um e um só destes objectos.

No modelo do hipergrafo um utilizador pode estar posicionado em vários nodos ao mesmo tempo (uma vez que a cada um deles corresponde uma página que, por sua vez, é um objecto com um valor atómico) e tem acesso concorrente a vários segmentos da base de dados. O estado de um utilizador compreende o estado da base de dados e um conjunto $M \subseteq N$, que serve como um conjunto de marcadores designando os nodos correntes. «Com esta extensão, a função E captura as semelhanças da estrutura dos arcos, e o conjunto M captura o conceito de conjunto corrente de nodos» [TOM89].

A figura seguinte mostra o estado de um utilizador de acordo com este modelo:

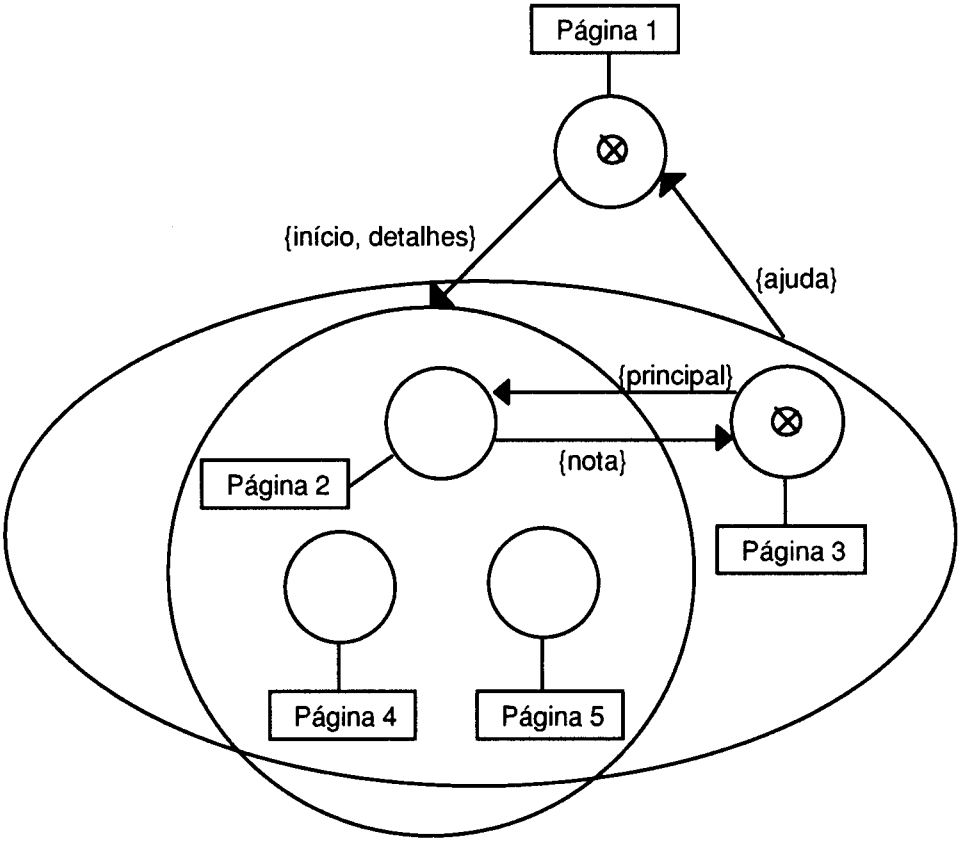


Fig. 3.6 - Exemplo do estado de um utilizador de acordo com o modelo do hipergrafo.

Tompa distingue, ainda, visões do utilizador de estados do utilizador. Por **visões do utilizador**, ele entende personalizar a rede, i.e., o utilizador poder modificar a rede. Diferentes visões trazem ainda as vantagens de permitir ao utilizador ver mais facilmente os dados usados mais frequentemente, escondendo dados desnecessários (formando *viewing filters*), e de permitir diferentes níveis de acesso de acordo com o perfil do utilizador. Uma visão é sempre definida de acordo com algum estado de uma base de dados - assim os arcos podem aparecer na visão mesmo se nenhum dos nodos estiver definido pela visão. Assim, o estado de um utilizador representa a sua posição corrente de acordo com o estado da base de dados e a visão.

Qualquer operação sobre a base de dados pode ser aplicada na visão sobre a base de dados. Em particular, outra visão pode ser sobreposta, i.e., é possível organizar as visões em cascata. São praticáveis níveis de personalização tão fortes como as de qualquer outro ambiente. A definição de visão permite também ao sistema suportar a partilha de estruturas entre os utilizadores, pois com a mesma estrutura é possível ter várias visões diferentes de acordo com as necessidades, o que vem facilitar tarefas comuns entre os utilizadores, em particular o trabalho cooperativo.

Resumindo, o modelo descrito aqui possui as seguintes características:

- a estrutura relacionando fragmentos de uma colecção de documentos é separada do conteúdo dos fragmentos permitindo, assim, que outras estruturas se sobreponham nos mesmos conteúdos e, ao contrário, que outros conteúdos usem a mesma estrutura;

- elementos comuns da estrutura são explícitos sendo, por isso, mantidos mais facilmente;

- é suportado *browsing* concorrente quer a nível de nodos apresentados (pois cada nodo só corresponde a um objecto), quer a nível de utilizadores a navegarem em concorrência na rede de nodos;

- podem ser criadas e mantidas visões personalizadas que podem ser encasquetadas arbitrariamente e partilhadas entre os utilizadores.

3.4 - TRELLIS

David Stotts e Richard Furuta desenvolveram uma definição formal do modelo Trellis do hipertexto. Este modelo não só representa as inter-relações que ligam os vários fragmentos de informação, como especifica uma semântica de *browsing* a ser associada ao hipertexto. É baseado em redes Petri e assume-se como uma generalização do tão bem conhecido grafo orientado.

Tira-se vantagem do facto de «as redes Petri não só capturarem o poder descritivo dos grafos orientados, (...) mas providenciarem também um mecanismo matemático abstracto e preciso para controlo e análise da "execução" e "*browsing*" do hipertexto»[SF89] .

No Trellis um hipertexto H é um sêxtuplo $H = <N, C, W, B, P_l, P_d>$ onde

- $N = <S, T, F>$ é uma **estrutura** de rede Petri;

  com:

  * $S = \{s_1, \ldots, s_n\}$ é um conjunto finito de **lugares** com $n \geq 0$;

  * $T = \{t_1, \ldots, t_m\}$ é um conjunto finito de **transições** com $m \geq 0$ e $S \cap T = \varnothing$;

  * $F \subseteq (S \times T) \cup (T \times S)$ é a relação de fluxo, um mapeamento representado **arcos** entre lugares e transições.

    (Para mais pormenores sobre a teoria básica das redes Petri, consultar [SF89]).

- C = um conjunto de **conteúdos** de documento;

- W = um conjunto de **janelas**;

- B = um conjunto de **botões**;

- P_l = uma **projecção lógica** para o documento;

- P_d = uma **projecção de apresentação** para o documento.

Isto significa que «um hipertexto consiste numa rede Petri representando a estrutura de ligações do documento, vários conjuntos de componentes *human-consumable* (conteúdos, janelas e botões), e duas colecções de mapeamentos, chamados projecções, entre a rede Petri, os *human-consumables*, e os mecanismos de apresentação»[SF89]. A figura 3.7 representa uma estrutura de hipertexto baseada numa rede Petri.

Dada a estrutura de rede Petri N = <S, T, F> num hipertexto H, a **projecção lógica** de H é o triplo P_l = $<C_l, W_l, B_l>$ no qual

- $C_l = S \rightarrow C \cup \{v\}$
- $W_l = S \rightarrow W \cup \{v\}$
- $B_l = T \rightarrow B \cup \{v\}$

com,

* v = null
* C_l = uma função que associa um elemento de conteúdo com cada lugar na rede Petri N
* W_l = uma função que associa uma janela lógica a cada lugar em N
* B_l = uma função que associa botões lógicos com cada transição em N

A um lugar da rede Petri pode estar associado um pedaço de texto, um gráfico, uma imagem digitalizada, um trecho de música, um segmento de vídeo ou, ainda, outro hiperdocumento criando,

assim, uma hierarquia natural no modelo. Uma das vantagens da criação desta hierarquia é que cada rede Petri ser analisável independentemente. Outra, que exige mais elaboração, é que o hiperdocumento pode ser organizado mais eficazmente do que com uma estrutura plana, permitindo visões diferentes de acordo com o nível de abstracção.

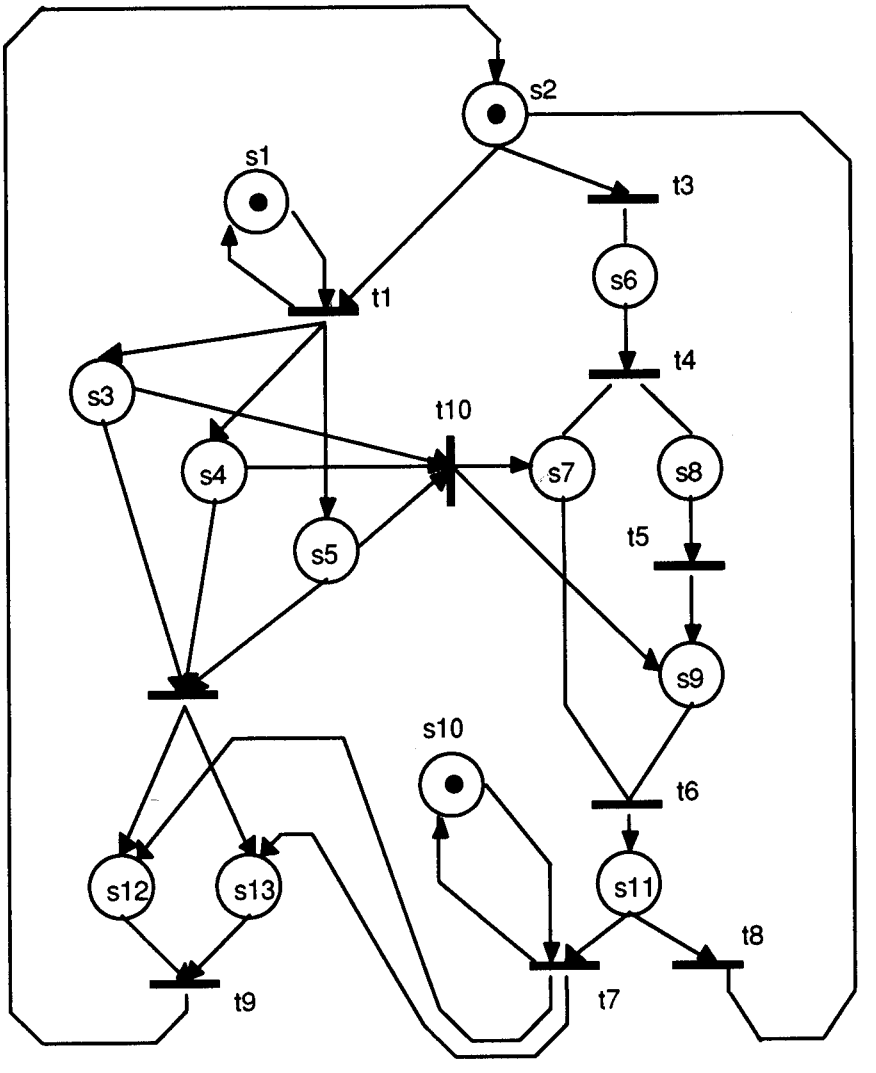


Fig. 3.7 - Estrutura de uma aplicação hipertexto baseada numa rede Petri.

Um **hipertexto marcado** é um par $H_M = <H, M>$, no qual $H = <N, C, W, B, P_l, P_d>$ é um hipertexto, e M é uma marca para a rede Petri N em H. Ocorre o caso especial do estado inicial definido como $H_{M0} = <H, M_0>$.

Pode-se pensar num hipertexto marcado como representando o estado do hipertexto durante a navegação. É uma caracterização do conjunto de caminhos possíveis através de um hipertexto a partir de determinado ponto.

A semântica de execução de uma rede Petri providencia um modelo de navegação num hipertexto marcado. Um testemunho num lugar s indica que os conteúdos do lugar $C_l(s)$ estão a ser apresentados. Quando um testemunho se move para um lugar vazio, o elemento de conteúdo associado é mapeado no dispositivo de apresentação; do mesmo modo, quando todos os testemunhos são removidos de um lugar, deixando-o vazio, o elemento de conteúdo que lhe está associado é removido do dispositivo de apresentação. Os testemunhos movem-se através da rede à medida que as transições são accionadas - isto consegue-se através da selecção dos botões lógicos existentes no dispositivo de apresentação.

Resumindo, um hipertexto marcado serve-se da estrutura gráfica e da semântica de execução da rede Petri para determinar que caminhos são possíveis seguir a partir do estado corrente. Durante a navegação, a marca M corrente da rede Petri determina que elementos do hipertexto são visíveis. As transições possíveis sob M determinam que botões são visíveis, e por isso seleccionáveis, em que janelas. A selecção de um botão pelo leitor acciona uma das transições possíveis gerando, assim, um novo estado M' a partir de M e causando que a apresentação se altere de forma correspondente. A navegação termina formalmente quando não há mais botões seleccionáveis.

Os métodos aplicáveis aos grafos orientados são estendidos às redes Petri. A representação de rede Petri é essencialmente gráfica e, consequentemente, é só uma alteração incremental ao comum modelo do grafo orientado do hipertexto. Qualquer sistema de hipertexto baseado em grafos orientados pesados pode ver a sua representação substituída por redes Petri, ganhando especificação da concorrência e controlo sem sacrificar nenhuma funcionalidade..

O modelo Trellis dá ao autor do hipertexto mais controlo sobre as sequências de navegação. Permite um reforço flexível de tais sequências, tornando as restrições de *browsing* uma parte integral da estrutura de um

hipertexto em vez de aplicá-las como mecanismo externo de *browsing*. É possível o controlo de concorrência e de sequenciação podendo ser especificados até qualquer grau que o autor exigir.

Algumas das vantagens deste modelo são:

- providencia um formalismo unificador para descrever e discutir sobre muitas das características dos sistemas hipertexto existentes;

- é uma alteração incremental ao formalismo do grafo orientado e por isso os modelos de aplicações baseados no grafo orientado são facilmente passados para o Trellis;

- permite a especificação da semântica de *browsing* do hipertexto;

- por haver uma separação entre o conteúdo e a estrutura, uma estrutura pode servir para representar várias versões de um documento;

Adicionalmente, a utilização de redes Petri como base do hipertexto no modelo Trellis permite solucionar alguns problemas como:

- **Parâmetros de apresentação**. Podem ser determinadas algumas características de apresentação, como por exemplo o número máximo de janelas, de modo a que se possa planear os *layouts* de écrans;

- **Caminhos de *browsing* concorrentes e sincronização**. Um hipertexto baseado em redes Petri indica naturalmente os elementos de conteúdos a serem vistos ao mesmo tempo. A concorrência dos conteúdos e o resultado da execução da semântica de *browsing* tem a ver, em grande parte, com a distribuição inicial dos testemunhos na rede;

- **Nodos alcançáveis e não-alcançáveis**. É possível verificar que nodos são alcançáveis e, mais importante, que nodos não são alcançáveis. Também isto depende da posição inicial dos testemunhos na rede. Assim é possível restringir o acesso a determinadas áreas do hipertexto e definir versões personalizadas.

3.5 - Modelo Conceptual Usado no MULTUS

MULTUS (Multimedia Office Server) é um sistema experimental desenhado com o objectivo de melhorar a aquisição de documentos multimedia. Um sistema deste género deve exigir que os documentos estejam modelados de uma forma homogénea (cumprindo certas regras) e do modo mais conveniente para permitir que os utilizadores especifiquem correctamente e com facilidade os documentos a que querem aceder. É, por isso, baseado em técnicas de modelação de documentos que serão aqui brevemente apresentadas.

Carlo Meghini, Fausto Rabitti e Costantino Thanos apresentam em [MEG91] um formalismo de modelação conceptual de documentos multimedia. Embora aqui se esteja a tratar de modelação de aplicações hipermedia, esta abordagem é suficientemente interessante para ser referida.

Este formalismo é baseado na modelação orientada ao objecto. Assim, os documentos são representados como objectos complexos compostos por componentes de texto, gráficos imagem e som.

Distinguem-se três estruturas diferentes: a **estrutura lógica**, a **estrutura de *layout*** e a **estrutura conceptual**. A estrutura lógica diz respeito à divisão lógica do documento (por exemplo em capítulos e secções) e determina como esses componentes lógicos se relacionam com as porções de conteúdo (valores básicos de dados como textos, imagens, etc.). A estrutura de *layout* tem a ver com os formatos de apresentação do documento e determina como os componentes de layout (páginas por exemplo) se relacionam com as porções de conteúdo. Por sua vez, a estrutura conceptual prende-se com a representação dos conteúdos do documento.

Segundo esta abordagem, um modelo ideal deveria poder representar numa única base de trabalho as diferentes estruturas de representação da informação: a estrutura lógica, a estrutura conceptual e a estrutura de *layout*. Se se considerar a estrutura conceptual como uma árvore, então as folhas dessa árvore são mapeadas nos componentes da estrutura lógica em diferentes níveis de detalhe.

Adicionalmente, é defendido que o modelo deve adoptar a visão do paradigma orientado ao objecto, tratando os vários documentos e os seus componentes como objectos e usando os mecanismos de abstracção associados a este paradigma para organizar a base documental. Assim, a base de dados semântica associada segue este paradigma.

A definição de tipos de documentos é feita assumindo que a estrutura especificada no tipo pode ser usada para especificar métodos de acesso e formulação de *queries*. No entanto, Meghini et al. consideram que as estruturas multimedia tendem a diferir muito de instância para instância e, por isso, se um tipo de documento fosse definido como a estrutura de todos os documentos pertencentes àquele tipo, haveria um ou dois documentos por tipo. Assim, introduzem a definição de **tipo fraco** que «apenas especifica parcialmente a estrutura das suas instâncias, permitindo componentes de tipo não especificado (chamado tipo de componente complexo)»[MEG91]. Um tipo de componente complexo pode ser especializado noutro tipo possibilitando, assim, a construção de uma hierarquia **is_a** entre tipos fracos.

A figura 3.8 mostra uma exemplo de um documento multimedia contendo textos e imagens juntamente com a sua estrutura conceptual.

Alguns dos componentes conceptuais básicos estão ligados a valores simples, básicos, como é o caso de "remetente", "destinatário", "produto" e "preço"; outros a porções de texto, como é o caso de "termos"; e outros a componentes complexos do documento como é o caso de "descrição".

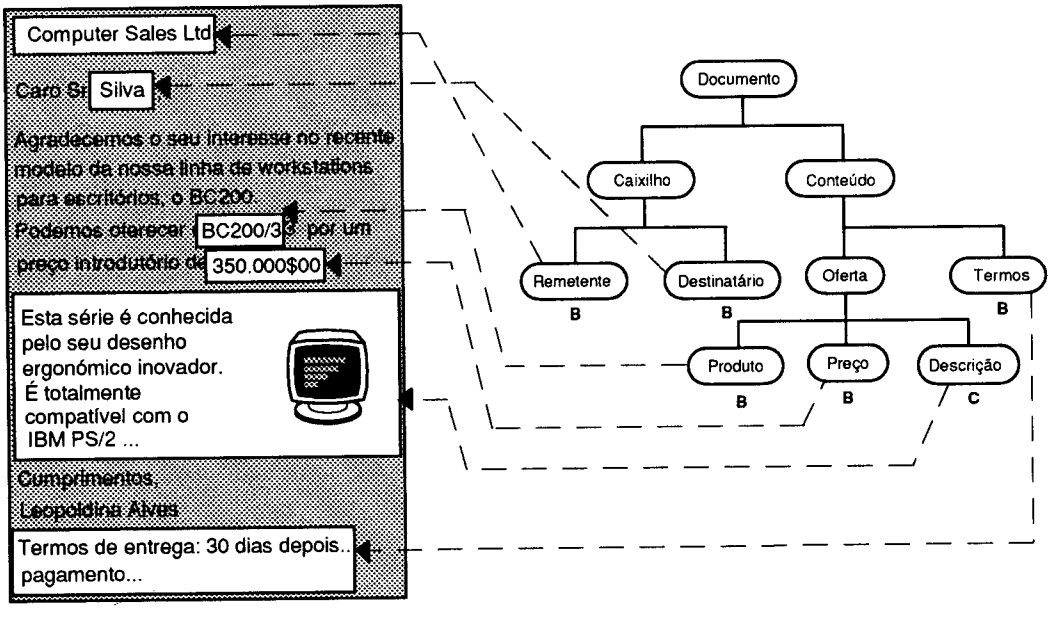


Fig. 3.8 - Exemplo de um documento multimedia e sua estrutura conceptual.

«A multimedialidade[3] é introduzida na estrutura lógica dos documentos. É providenciada uma classe primitiva para cada tipo diferente de formato media; depois o utilizador instancia estas classes, ou então instancia especializações (criadas por ele) delas. Ele próprio pode definir as estruturas de classificação (generalização-especialização) e de agregação (a partir das quais vão surgir os objectos compostos) das classes que vai usar»[MEG91].

[3] «Multimediality» no original.

3.6 - Discussão

Todas estas abordagens são interessantes e é pena não se encontrar muita coisa escrita sobre elas. Se tivessem sido mais divulgadas talvez não houvesse tantas aplicações hipermedia construídas sem nenhum suporte metodológico ou sem nenhuma técnica de representação do conhecimento que as especificasse. De qualquer modo, como se disse atrás, só agora é que os autores de aplicações hipermedia estão a começar a sentir a necessidade de modelar as aplicações que querem construir.

Assim vai-se classificar, segundo alguns parâmetros, as abordagens aqui apresentadas. Note-se que a última abordagem, o modelo usado no MULTUS, não é direccionado para aplicações hipermedia pelo que, em parâmetros que não são do seu âmbito, nem sequer é mencionada.

- **Possibilidade de se estender a partir de um modelo baseado no grafo orientado**

  A noção de nodo numa aplicação de hipertexto/hipermedia nem sempre é pacífica: para uns corresponde a um único pedaço, atómico, de informação usando um único medium; para outros pode ser um pedaço de informação, desde que modular, constituído, se necessário for, por outros pedaços de informação usando vários media.

  Aparentemente, em todas as abordagens aqui descritas, excepto no modelo utilizado pelo MULTUS (porque não é hipermedia), existe esta facilidade.

- **Facilidade de uso**

  Este item de comparação diz respeito unicamente ao uso em papel uma vez que não se conhecem os sistemas de implementação destas técnicas. Nenhuma destas abordagens (excepto a última) parece ser muito fácil de usar em papel.

  O HDM, uma vez que usa só um diagrama para representar todas as entidades, seus conteúdos e as várias ligações que suporta, parece tornar a representação em papel bastante confusa.

Á segunda abordagem falta-lhe um modo de especificar ligações entre páginas destino, particularmente originárias de raízes diferentes, ou ligações noutros sentidos ou, ainda, maneira de especificar a relação entre os vários conteúdos. Uma vez que é baseada no paradigma orientado ao objecto, e criando a estrutura de classificação dos objectos este problema é minorado. No entanto, esta abordagem ficaria bastante mais forte se se pudesse representar os conteúdos dos nodos-tipo.

A terceira e quarta abordagens, embora bastante interessantes, em particular a que é baseada em redes Petri, não parecem poder ser usadas com facilidade sem ser fora do ambiente em que foram implementadas, principalmente por uma questão de coerência e integridade do modelo. Além disso, se a rede for grande, é difícil determinar em que lugar se devem por os testemunhos, de modo a proporcionar determinado tipo de navegação. Contudo, a abordagem baseada no hipergrafo, embora menos completa, parece ser mais simples de usar fora do seu ambiente de implementação.

A última abordagem, direccionada para documentos multimedia, é fácil de utilizar, embora se se estiver fora do ambiente de implementação, não pareça ser fácil ter uma noção global da aplicação. Uma vez que ela tem como objectivo modelar conceptualmente, numa única aplicação, inúmeros tipos diferentes de documentos multimedia, a introdução do conceito de "tipo fraco" veio melhorar as coisas neste campo.

- **Semântica de *browsing* associada**

Segundo [GAR91], a **semântica de *browsing*** «fornece o comportamento em *runtime* da aplicação».

O único modelo que onde se fala abertamente de uma semântica de *browsing* associada é o TRELLIS.

- **Suporte de diferentes hipervisões**

  No HDM, embora se esteja a trabalhar nisso, para já não suporta diferentes hipervisões. O livro electrónico hipermedia também não. Os outros dois (terceira e quarta abordagens) possuem esta característica.

- **Suporte de várias versões**

  Já que hipermedia significa também re-utilização da informação a nível de nodos, um versão nova aproveitaria os nodos comuns com a anterior de uma forma transparente ao leitor. Assim, este reduz-se ao caso de diferentes hipervisões para o mesmo utilizador pois a hiperbase será a mesma.

- **Possibilidade de modelar ligações dinâmicas (ligações estabelecidas em *runtime*)**

  Podem ser distinguidos dois tipos de ligações dinâmicas: aquelas que são pré-especificadas mas que só se estabelecem mediante determinadas condições ou as que se estabelecem a partir de informações dadas pelo utilizador (resultantes de *queries*, por exemplo). Este parâmetro diz respeito aos primeiros, dado que os outros dependem da vontade do utilizador e não podem ser modeladas.

  Aparentemente só o modelo TRELLIS permite a modelação deste tipo de ligações.

- **Possibilidade de usar a mesma estrutura com conteúdos diferentes**

  Todas as abordagens o permitem.

- **Especificação dos diferentes formatos media usados**

  O modelo usado no MULTUS permite fazer a distinção dos formatos media usados na estrutura lógica. Mais nenhuma das abordagens o permite.

- **Implementação em sistema autor próprio**

  Todas as abordagens.

- **Facilidade de entender a estrutura global da aplicação modelada**

  O esquema do HDM permite que se perceba com facilidade a estrutura global da aplicação. O modelo do livro electrónico hipermedia também apresenta esta característica. Os outros dois (o modelo baseado no hipergrafo e o TRELLIS) não possuem esta facilidade, embora o primeiro seja melhor que o segundo. Isto deve-se principalmente ao facto do conceito deles de nodo (um pedaço de informação representado num único formato media que, muitas vezes, não tem significado relevante por si só).

IV - MODELAÇÃO DE APLICAÇÕES HIPERMEDIA - UMA ABORDAGEM

4.1 - Resumo

Como já se disse anteriormente, um modelo básico de aplicações de hipertexto é um grafo orientado. Os fragmentos de informação correspondem aos nodos e as ligações entre esses fragmentos correspondem aos arcos.

Este modelo já se torna insuficiente para representar a informação relativa a uma aplicação hipermedia em desenvolvimento. Vai ser necessário fazer-lhe uma expansão.

Não é lógico representarem-se no modelo todos os nodos existentes na hiperbase - é muito mais prático conseguir arranjar uma abstracção que represente vários nodos com características semelhantes, i.e., que correspondam a um mesmo conceito em relação à aplicação em causa. A esta abstracção vai chamar-se **entidade**.

As ligações entre os nodos também podem representar mais informação do que o que acontece no modelo clássico do grafo orientado. Aqui como se representa várias ligações com a mesma fonte e o mesmo destino? E uma ligação que pode, ou não existir? E, ainda, uma ligação com vários destinos? Neste caso, como é que a ligação "sabe" qual o seu destino, em dada altura?

Adicionalmente, torna-se necessário saber como é que cada um dos nodos vai ser constituído, quer em termos de informação apresentada, quer em termos dos diferentes formatos media usados nessa apresentação.

Para cada entidade existente na aplicação deve ser possível representar:

- a própria **entidade** e distingui-la das outras entidades da mesma aplicação;
- as **ligações** *(links)* que a têm quer como destino, quer como fonte;
- a **separação da informação,** nessa entidade, a um nível lógico - que vai corresponder a uma separação física no écran.
- todos os seus **componentes** e a forma como estão organizados e relacionados;
- a que **formatos media** (texto, imagem, som, etc.) correspondem os seus componentes;
- as **diferentes visões** (perspectivas) sobre essa entidade;

É também necessária a definição de algumas regras responsáveis por manter a coerência global do modelo pelo que se desenvolveu uma filosofia de abordagem à fase de análise tendo sido definidos conceitos, notações e regras de representação do conhecimento, tendo como base os pontos acima descritos.

De modo a poder representar de uma forma clara e concisa não só cada uma das entidades, como a visão global de todas elas e de todas as ligações entre elas (proporcionando uma visão panorâmica sobre toda a aplicação), vão considerar-se dois diagramas diferentes:

- O Diagrama de **Ligações** e
- O Diagrama de **Conteúdos**.

Cada um destes diagramas tem funções e características bem específicas e diferentes das do outro. O **Diagrama de Ligações** preocupa-se apenas com as entidades e com as ligações que as relacionam. O **Diagrama de Conteúdos** serve, essencialmente, para representar a constituição de uma entidade tendo em

conta os seus componentes e a sua estrutura lógica. É feita uma representação gráfica de cada entidade da aplicação que, posteriormente, é complementada com uma caracterização escrita mais pormenorizada - o **Anexo ao Diagrama de Conteúdos.**

Convém, neste ponto, recordar o exemplo da aplicação hipermedia sobre os animais mamíferos descrito anteriormente:

> Quer-se representar, de alguma forma, informação sobre animais mamíferos tendo em conta a sua ordem, família, género e espécie. O leitor deve poder consultar dados relativos só a determinada família, só a determinado género ou só a determinada espécie. Adicionalmente, quer-se que estando a consultar uma família, se possa aceder directamente a outra família, a todos os géneros ou a todas as espécies dentro dessa família. Se se estiver a consultar determinado género deve poder aceder-se a outro género, da mesma família ou não, à família correspondente e a todas as espécies desse género. Ainda, a partir de uma espécie deve poder consultar-se outra espécie pertencente ou não aos mesmos género e família e consultar o género ou a família correspondente a essa espécie.

4.2 - O paradigma orientado ao objecto

A maior parte das ferramentas-autor hipermedia utilizadas hoje em dia são baseadas, senão totalmente pelo menos parcialmente, no paradigma orientado ao objecto, percebendo-se uma tendência clara da cada vez maior utilização desta filosofia como base deste tipo de ferramentas. Esta tendência prende-se com razões tão diversas como:

- a maior facilidade e flexibilidade de utilização por parte dos autores das aplicações - é base de acordo mais ou menos consensual que o paradigma orientado ao objecto modela a nossa forma de organizar e classificar tudo o que nos rodeia;

- a possibilidade de fornecer a cada objecto os seus atributos e o seu comportamento;

- a possibilidade de reutilização da informação;

- a maior flexibilidade providenciada para a manutenção das aplicações.

Assim, também as fases de desenvolvimento da aplicação têm de reflectir esta tendência acentuada, pelo que se escolheu este paradigma como base da abordagem aqui descrita.

O paradigma orientado ao objecto difere da programação convencional logo a partir da forma de concepção do *software*, da maneira como se pensa e se "olha" para a realidade de modo a conceber um modelo dela. É reclamado por muitos como a forma mais fiel de representar e simular a realidade. Isto deve-se, fundamentalmente, ao facto de um programa não ser constituído por um conjunto de variáveis e procedimentos aninhados que actuam sobre elas, mas por um conjunto de entidades, chamadas **objectos**, que têm determinados **atributos** e **comportamento** e que interagem entre si de modo muito semelhante ao que se passa no mundo real. Os objectos reagem a estímulos exteriores a eles próprios utilizando para tal o seu conjunto de métodos internos, isto é, o seu comportamento. Os objectos comunicam entre si através do envio de **mensagens**. De qualquer modo tanto os atributos como o comportamento de determinado objecto só são conhecidos por ele próprio; nenhum dos outros objectos tem acesso a este tipo de informação. Por exemplo, um automóvel pode ter entre os seus atributos a marca, o modelo ou a cor. O comportamento pode incluir ligar o carro, andar ou apitar. Considere-se um sistema homem-carro; considere-se, agora, que o homem **toca** na buzina do **carro** - **ouve-se** uma buzinadela! Note-se, no entanto, que o homem **não conhece** os mecanismos interiores do carro responsáveis pelo ruído que se ouviu: ele sabe apenas que tocando em determinado sítio o carro apita... Fazendo o paralelo para o paradigma orientado ao objecto e tendo em conta, como se disse em cima, que "apitar" pode ser um método do objecto-carro, então o objecto-homem envia uma mensagem "toca" ao objecto-carro, ao que este reage executando o método "apitar". Este método é conhecido apenas pelo objecto-carro, i.e., mais nenhum objecto o conhece.

Também como no mundo real, os objectos com características similares são agrupados em **classes**: todas as cadeiras de uma mobília de sala de jantar são parecidas mas nenhuma exactamente igual - são, no entanto, classificadas como "cadeiras da sala de jantar". Voltando ao exemplo dos automóveis, pode-se classificar **um** PEUGEOT 106 XR, **um** BMW 316i ou **um** FIAT UNO 60 SX como automóvel ligeiro de passageiros. Por outro lado, **um** PEUGEOT 205 XAD ou **um** VOLKSWAGEN PARATI, pode ser classificado como automóvel ligeiro de mercadoria. Cada PEUGEOT 106 XR existente é **um** automóvel ligeiro de passageiros. Cada VOLKSWAGEN PARATI existente é **um** automóvel ligeiro de mercadorias.

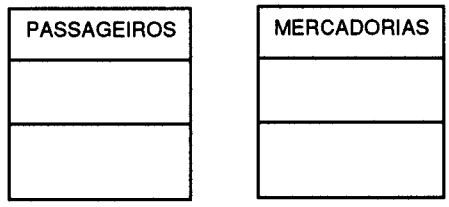


Fig. 4.1 - As classes "PASSAGEIROS" e "MERCADORIAS" segundo a notação de Coad e Yourdan [CY90].

Note-se que se fez a seguinte classificação: "automóveis ligeiros de passageiros" e "automóveis ligeiros de mercadorias" - tanto uns como outros são automóveis ligeiros. Então há características comuns aos dois tipos de automóveis: em primeiro lugar por serem automóveis e, depois, por serem ligeiros. Estas características podem ser todas definidas no seu nível correspondente, i.e., de automóveis e, depois, de ligeiros. Assim, pode-se dizer que qualquer automóvel ligeiro de passageiros ou qualquer automóvel ligeiro de mercadorias possui determinadas características pelo facto de ser um automóvel e pelo facto de ser ligeiro, i.e., **herda** essas características de níveis superiores.

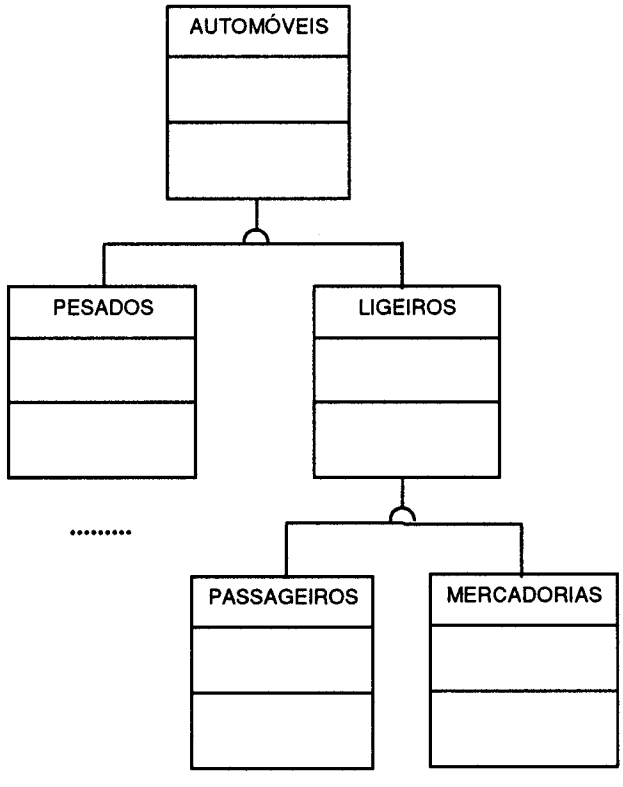


Fig. 4.2 - Parte de uma rede de classes denotando a estrutura de herança segundo a notação de Coad e Yourdan [CY90].

Os objectos podem também ser **compostos** por outros objectos, assim como se passa no mundo real. Qualquer automóvel tem assentos, um motor, rodas e pelo menos um volante, entre outras coisas. Assim, pode-se dizer que um automóvel é **composto** por assentos, rodas, motor, etc.; por outro lado um assento é **parte de** um automóvel, uma roda **é parte de** um automóvel e um motor **é parte de** um automóvel. Qualquer um destes objectos tem atributos e um comportamento que lhe permite responder a estímulos exteriores. O motor, por exemplo, responde a estímulos do acelerador aumentando ou diminuindo o número de rotações por minuto. Este envia estímulos a um eixo que, por sua vez, faz girar as rodas que lhe estão acopladas. No paradigma orientado ao objecto toda esta comunicação se realiza através do envio de mensagens (estímulos) entre os vários objectos envolvidos, que despoletam determinado comportamento.

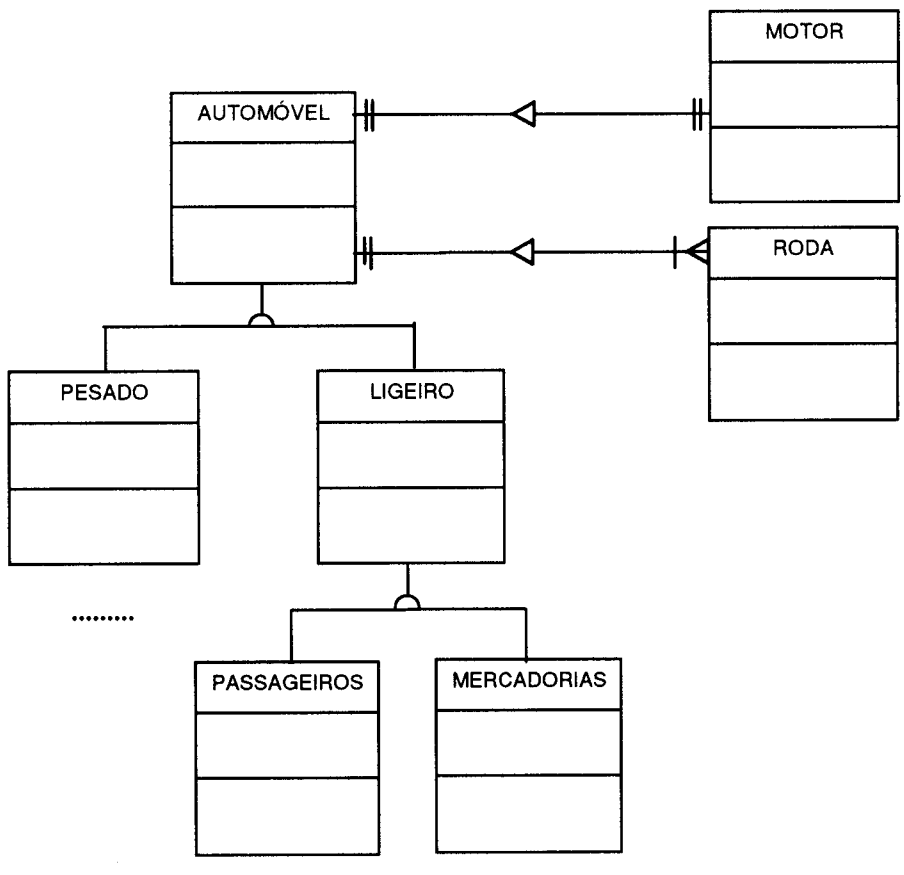


Fig. 4.3 - Parte de uma rede de classes denotando as estruturas de herança e composição segundo a notação de Coad e Yourdan [CY90].

De modo a que tudo fique suficientemente claro vai agora dar-se uma breve explicação de alguns dos conceitos básicos do paradigma orientado ao objecto segundo [SB90]:

Classe - Uma classe é uma descrição de um ou mais objectos similares.

Composição de objectos - Um conjunto de objectos interligados que são instanciados ao mesmo tempo.

Encapsulamento - O facto de os atributos e comportamento de determinado objecto só ser conhecido por ele próprio.

Especialização - O processo de modificar algo genérico para algo específico. "Automóveis ligeiros" é uma especialização de "automóveis".

Herança - Quando uma classe é colocada na rede de classes, ela herda os atributos e os métodos (comportamento) das suas superclasses. A classe "automóveis ligeiros" herda os atributos e métodos da classe "automóveis". Por outro lado, a classe "automóveis ligeiros de passageiros" herda os atributos e métodos da classe "automóveis ligeiros".

Instância - Descreve a relação entre um objecto e a sua classe. Um determinado PEUGEOT 106 XR (objecto) é instância da classe "automóveis ligeiros de passageiros".

Mensagem - A especificação de uma operação a ser realizada por um objecto. É algo semelhante à chamada de um procedimento.

Método - A função que implementa a resposta quando uma mensagem é enviada a um objecto. O conjunto de métodos de um objecto define o seu comportamento.

Objecto - O elemento primitivo do paradigma orientado ao objecto. Um objecto possui atributos e um comportamento (especificado pelos seus métodos). Os objectos guardam valores para os seus atributos e respondem a mensagens executando os

seus métodos.

Subclasse - Uma classe que está a um nível inferior na rede de classes do que a classe especificada. A classe "automóveis ligeiros" é uma subclasse de "automóveis".

Superclasse - Uma classe que está a um nível superior na rede de classes do que a classe especificada. A classe "automóveis" é uma superclasse de "automóveis ligeiros".

Na abordagem aqui apresentada tudo é tratado como objectos desde as ligações até aos próprios nodos e todos os seus componentes. Assim, e seguindo o raciocínio de Philip Barker em [BAR92], pode-se considerar que existe uma base comum de objectos. No entanto, aqui vai-se ainda mais longe: até o próprio nodo é um objecto que é composto por outros objectos. Deste modo garante-se a partilha de objectos em toda a aplicação hipermedia.

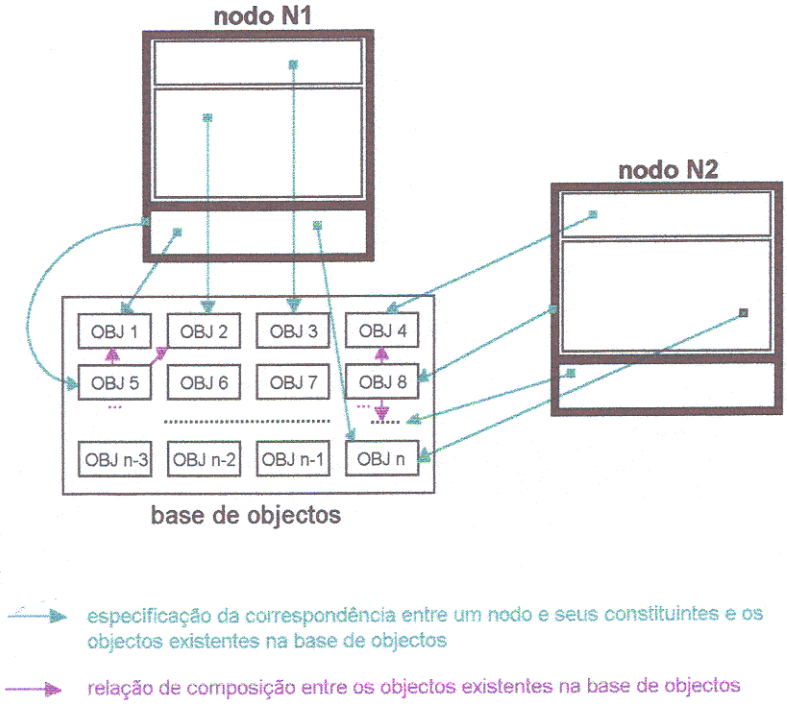


Fig. 4.4 - O paradigma orientado ao objecto como base de aplicações hipermedia.

Os objectos dentro da base de objectos estão relacionados uns com os outros através de relações de composição - por exemplo, o objecto OBJ5 (que corresponde ao nodo N1) é composto pelos objectos OBJ1, OBJ2, OBJ3 e OBJn.

4.3 - O Diagrama de Ligações

Este diagrama é um descendente do grafo orientado, modelo comumente usado para representar aplicações hipertexto. Uma vez que este se revelou insuficiente para as necessidades actuais em termos de representação do conhecimento de aplicações hipermedia, achou-se necessário expandi-lo.

Na abordagem que aqui vai ser apresentada existem dois conceitos básicos: o de **entidade** e o de **ligação**. De seguida são definidos cada um desses conceitos, a sua funcionalidade e a notação que os representa. Posteriormente apresenta-se um exemplo prático.

4.3.1 - Entidade

Uma **entidade** representa um conceito.

Um **nodo** é uma instância de um conceito; mais especificamente, um nodo é uma instância de uma entidade. Corresponde, em hipermedia, a um pedaço modular de informação que pode ser constituído por vários outros pedaços dependentes uns dos outros utilizando, ou não, o mesmo formato media. Conforme já se disse anteriormente, os vários nodos de uma aplicação hipermedia estão relacionados entre si (o que não implica que eles não sejam modulares) através da existência de referências cruzadas, ou ligações.

Uma vez que esta aproximação tem por base os conceitos do paradigma orientado ao objecto, um nodo define-se como um objecto composto, exactamente porque é formado por outros objectos elementares ou compostos - o único objecto que não pode fazer parte de um nodo é outro nodo (um nodo pode conter referências para outro nodo mas nunca contê-lo). Dadas estas ligações com o paradigma orientado ao

objecto, é garantida a partilha de objectos entre os nodos, i.e., os objectos pertencentes a dado nodo (objecto composto) podem também pertencer a outro.

Por exemplo, em relação ao problema sobre os animais posto anteriormente (ponto 4.1), um nodo representando determinada espécie de raposas e outro representando determinada espécie de baleias representam o mesmo conceito - descrição de uma espécie de animais mamíferos - as instâncias desse conceito é que são diferentes - uma é sobre uma espécie de raposas, outra é sobre uma espécie de baleias. O conceito é representado na entidade ESPÉCIE; cada instância desse conceito corresponde a uma instância desta entidade que é um nodo.

A cada entidade é associado um nome que a identifique claramente de acordo com o conceito que ela representa. Esta é representada graficamente através do seu nome com um caixilho rectangular à volta, conforme se mostra na figura seguinte:

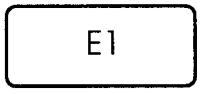


Fig. 4.5 - Representação gráfica de uma entidade.

Diz-se que uma entidade E1 é uma **entidade-fonte** de uma ligação quando essa ligação "parte" de qualquer instância dessa entidade.

Do mesmo modo, diz-se que uma determinada entidade E2 é uma **entidade-destino** de uma ligação quando essa ligação "aponta" para uma ou mais instâncias dessa entidade[4]. Ver figuras 4.6 e 4.7.

Como se verá no ponto seguinte, uma ligação entre nodos pode ter vários nodos-destino (instâncias ou não da mesma entidade) e é perfeitamente definida através do seu conjunto de nodos-destino e das condições de selecção de cada um deles. Existem, no entanto, duas regras fundamentais a cumprir:

[4] Note-se que, aqui, o termo "ligação" é usado tanto para a instância como para a abstracção em relação a essa instância. Poder-se-ia ter utilizado termos diferentes, mas optou-se por esta via por uma questão de simplicidade de notação.Assi, quando se fala em ligações entre nodos, está-se

- o conjunto das entidades-fonte de uma ligação tem que ter pelo menos um elemento;

- o conjunto das entidades-destino de uma ligação tem que ter pelo menos um elemento;

De outro modo a ligação não existe, i.e., fica "suspensa" pois não tem uma das "extremidades".

4.3.2 - Ligação

Uma **ligação** é também um objecto e é perfeitamente definida, tendo em conta o seu tipo, através do seu conjunto de nodos-destino e das condições de selecção de cada um deles. Não se definindo o nodo-fonte na ligação criam-se condições para também poder haver partilha de ligações entre os vários nodos existentes. Assim, quando se diz que determinado nodo é um nodo-fonte de uma ligação, significa-se apenas que **a ligação está também definida nesse nodo**, e não que lhe pertence única e exclusivamente. Esse nodo pertence, então ao conjunto dos nodos-fonte dessa ligação.

A ligação, no fundo, é o que permite que a informação esteja organizada sob uma forma não-linear, pois dá a possibilidade do leitor "saltar" de um nodo para outro nodo qualquer, definido como destino dessa ligação. A não-linearidade advém do facto de um nodo poder ter várias ligações para outros nodos permitindo, assim, o estabelecimento de relações de significado entre eles. É o que Ted Nelson chama o «entrelaçar de ideias»[CON87].

Existem dois **tipos** básicos de ligações: as ligações simples e as ligações com volta. Uma **ligação simples**, o tipo mais vulgar, liga um nodo-fonte a um ou mais nodos-destino. Uma **ligação com volta** do nodo A para o nodo B é uma ligação de A para B que permite ao leitor ir do nodo A para o nodo B e, depois, voltar ao nodo A se assim o desejar. Note-se que isto é diferente de se ter uma ligação simples do nodo A para o nodo B e, depois, outra do nodo B para o nodo A: no primeiro caso a ligação de B para A só existe se o leitor tiver alcançado o nodo B a partir do nodo A; no segundo caso, a ligação de B para A existe sempre, seja qual for o caminho escolhido pelo utilizador para chegar ao nodo B.

De modo a facilitar a comunicação pode-se, por vezes, falar numa ligação entre dois ou mais nodos. Isto realmente significa que essa ligação tem como fonte determinado nodo e "aponta" para os outros nodos. Cada ligação tem um nome que geralmente está associado aos seus destinos. Acontece, por vezes, num

diagrama de ligações existirem várias representações da mesma ligação. Uma vez que se trata de uma só ligação, pois são do mesmo tipo, têm os mesmos destinos e as mesmas condições de selecção de cada um deles (não importa que não tenha a mesma fonte), representa-se sempre usando o mesmo nome.

Uma ligação simples representa-se graficamente através de uma linha com uma seta numa das extremidades que indica o destino dessa ligação, e o seu nome escrito sempre em letras minúsculas, conforme se pode ver nas figuras seguintes (ver notação nos pontos i) e ii)):

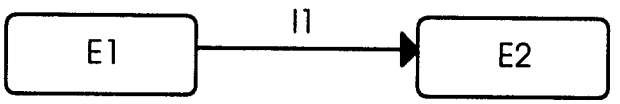


Fig. 4.6 - Existe uma ligação l1 de cada instância da entidade E1 para uma instância da entidade E2. E1 é a entidade-fonte e E2 é a entidade-destino.

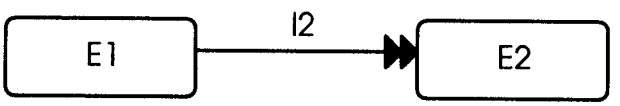


Fig. 4.7 - Existe uma ligação de cada instância da entidade E1 para uma ou mais instâncias da entidade E2. E1 é a entidade-fonte e E2 é a entidade-destino.

Uma ligação com volta representa-se graficamente através de duas linhas paralelas com uma seta numa das extremidades que indica o destino dessa ligação, e o seu nome escrito sempre em letras minúsculas, conforme se pode ver na figura seguinte:

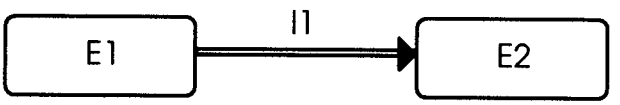


Fig. 4.8 - Existe uma ligação com volta l1 de cada instância da entidade E1 para uma instância da entidade E2. E1 é a entidade-fonte e E2 é a entidade-destino.

De modo a que não se gerem confusões nos nomes das ligações, convenciona-se que se escrevem sempre do lado esquerdo quando se está a olhar no sentido da seta, conforme se pode ver no exemplo seguinte:

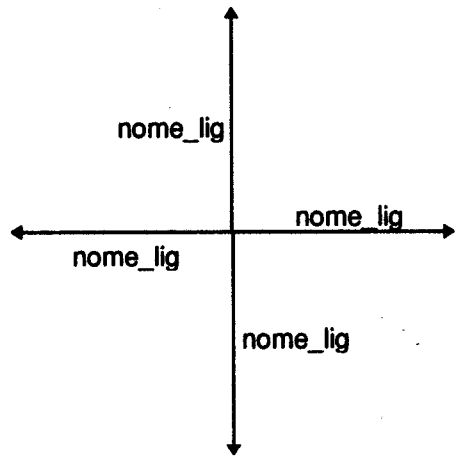


Fig. 4.9 - Convenção sobre o uso de nomes nas ligações.

Em relação às ligações entre as entidades, há que ter em conta dois tipos diferentes de **cardinalidade**:

i) **das próprias ligações** - quaisquer duas entidades podem ter zero, uma ou mais ligações entre elas (0, 1, ou mais ligações entre duas entidades). Um exemplo de várias ligações entre as entidades será um índice com ligações para várias instâncias de alguma entidade:

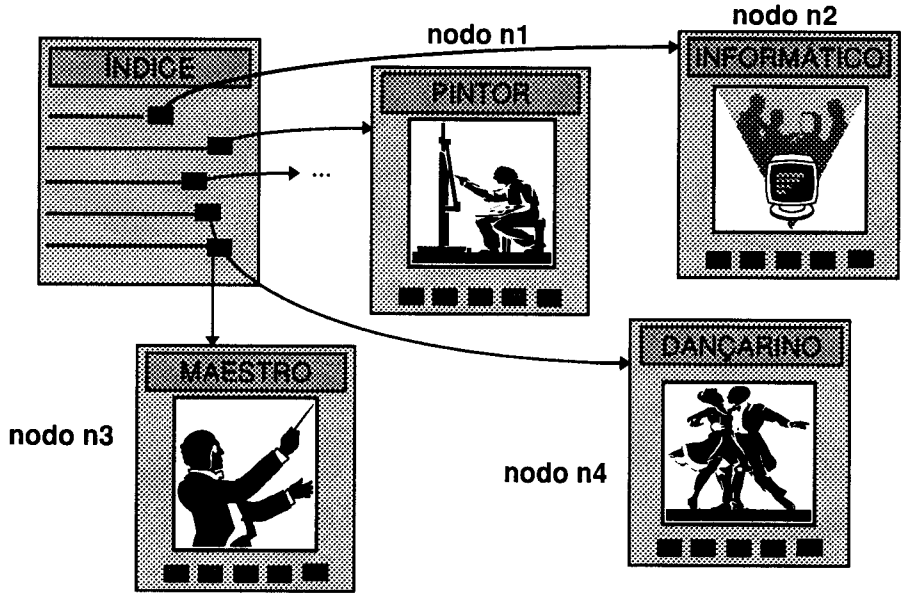


Fig. 4.10 - Exemplo da existência de várias ligações da entidade " ÍNDICE " para a entidade " PROFISSÕES ".

Os nodos n1, n2, n3 e n4 serão, provavelmente instâncias de uma entidade "PROFISSÕES". O nodo do índice é, provavelmente, instância de uma entidade "ÍNDICE". Assim, existem uma ou mais ligações da entidade "ÍNDICE" para a entidade "PROFISSÕES".

ii) **dos destinos de cada uma das ligações** - qualquer ligação entre duas ou mais entidades pode ter como destino uma ou mais instâncias dessas entidades (1 ou mais destinos por cada ligação).

Imagine-se o caso de numa aplicação haver a hipótese de três tipos diferentes de pesquisa: sumária, normal e completa. Vão existir nodos que só serão visitados na pesquisa completa e outros na pesquisa completa e na normal. Existirão, ainda, nodos que serão visitados nos três tipos de pesquisa. Assim, a ligação que leva ao nodo seguinte pode ter três destinos diferentes de acordo com o tipo de pesquisa previamente escolhido pelo utilizador. Outro caso será o do uso de versões múltiplas de um mesmo hiperdocumento.

i) Notação de representação da cardinalidade das próprias ligações.

De notar que:

- a cardinalidade das ligações é expressa junto à entidade-fonte.
- no quadro seguinte consideram-se os sentidos das ligações da esquerda para a direita.
- no quadro seguinte a notação apresentada refere-se a ligações simples e a ligações com volta.

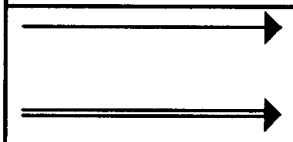
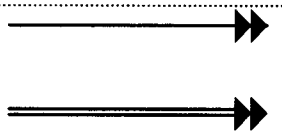
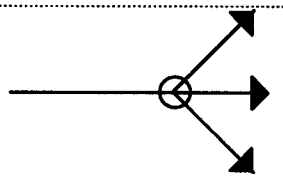
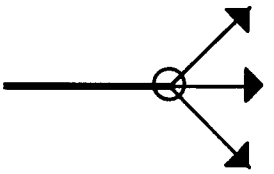
Notação	Cardinalidade	Explicação
.....	0 ou 1	Existe 0 ou 1 ligação
:::::::::::::::::::::::::::::::::		
———————	1	Existe exactamente 1 ligação
═══════════		
*.....	0 ou mais	Existem 0 ou mais ligações
*:::::::::::::::::::::::::::::::::		
*———————	1 ou mais	Existe pelo menos uma ligação (1 ou mais ligações)
*═══════════		

Quadro 4.1 - Notação de representação da cardinalidade das próprias ligações.

ii) Notação de representação da cardinalidade dos destinos de cada uma das ligações.

De notar que:

- a cardinalidade dos destinos é expressa junto à entidade-destino.
- no quadro seguinte considera-se que a cardinalidade das ligações é de um.
- no quadro seguinte a notação apresentada refere-se a ligações simples e a ligações com volta.

Notação	Cardinalidade	Explicação
	1	Cada ligação tem como destino uma e uma só instância da entidade-destino
	1 ou mais	Cada ligação tem como destino uma ou mais instâncias da entidade-destino. Só há uma entidade-destino, mas várias instâncias-destino.
 	1 ou mais	Cada ligação tem como destino uma e uma só instância de cada entidade-destino. Existem várias entidades-destino. Neste exemplo a ligação refere-se a uma instância de cada entidade-destino. Este caso também se pode utilizar em conjunção com o imediatamente anterior. Assim, cada ligação teria como destino uma ou mais instâncias de cada entidade-destino.

Quadro 4.2 - Notação de representação da cardinalidade dos destinos de cada uma das ligações.

A uma ligação, como objecto que é, vão estar associados determinados atributos e um comportamento. Alguns dos atributos podem ser, além do seu nome, o conjunto de nodos-destino. No comportamento podem estar incluídas, além das condições de acesso a cada um dos nodos-destino, o processo de voltar ao nodo anteriormente visitado, se for uma ligação com volta.

4.3.3 - Simplificação do Diagrama

Por vezes, a leitura do diagrama de ligações pode tornar-se um pouco confusa, especialmente se houver vários tipos diferentes de ligações entre as entidades. Assim, achou-se por bem encontrar formas de simplificação.

Uma das coisas mais comuns num diagrama de ligações é a existência, entre duas entidades, de ligações com sentido inverso. É possível representar duas ligações em sentido inverso usando uma só linha, conforme se mostra nas figuras seguintes:

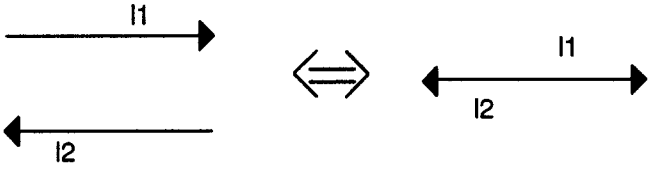


Fig. 4.11 - Simplificação de duas ligações com sentido inverso entre as mesmas entidades

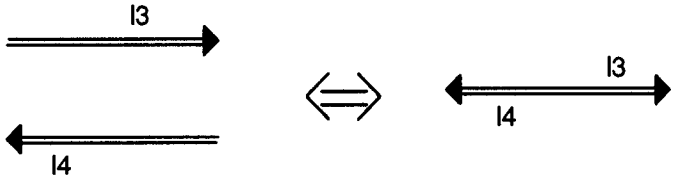


Fig. 4.12 - Simplificação de duas ligações com volta com sentido inverso entre as mesmas entidades

Num diagrama, por vezes, pode também existir representada várias vezes a mesma ligação. Uma vez que as ligações são caracterizadas pelo seu tipo, conjunto de nodos-destino e pelas condições de selecção de cada um deles, torna-se mais ou menos simples identificar quais as ligações que são representadas repetidamente num mesmo diagrama, eliminando-se as redundâncias de representação o mais possível. Assim, fazendo algo parecido com o que Frank Tompa faz no modelo do Hipergrafo (ver ponto 3.3), consegue-se que o diagrama fique bastante mais legível e mais simples de usar e modificar. As simplificações que seguidamente se explicam utilizam-se apenas nos casos em que as ligações têm a mesma cardinalidade e o mesmo tipo.

Para simplificar representações de uma ligação que tem diferentes entidades-fonte e uma entidade-destino, as regras são as seguintes: agrupam-se as entidades-fonte (utilizando uma circunferência ou uma elipse) e, usando só uma linha a representar as várias ligações, fazê-la sair da fronteira do agrupamento até à entidade-destino, mantendo o nome da ligação.

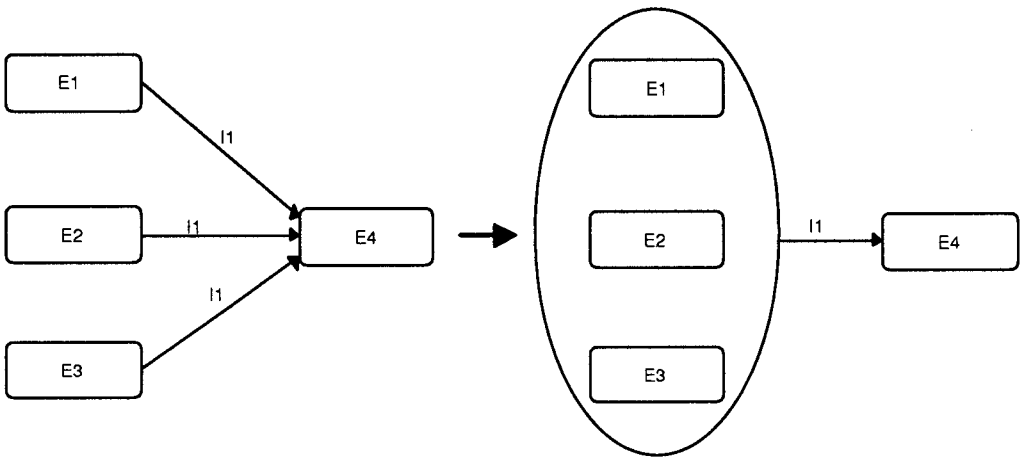


Fig. 4.13 - Simplificação de três ligações l1 com diferentes entidades-fonte.

É, ainda, possível simplificar a representação de uma ligação que tenha várias entidades-destino do seguinte modo: agrupam-se as entidades-destino (usando uma circunferência ou uma elipse) e leva-se a ligação até à fronteira desse agrupamento, mantendo o símbolo "O " de vários destinos e o nome da ligação.

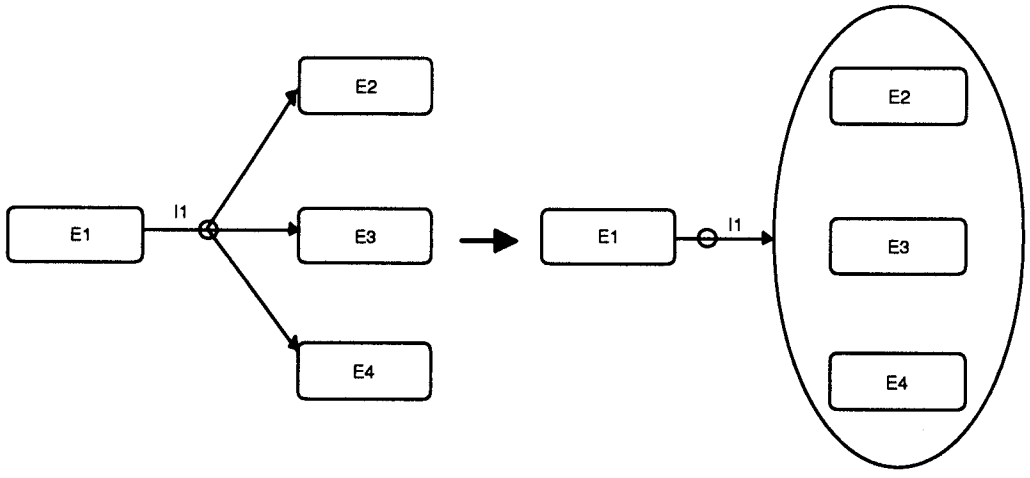


Fig. 4.14 - Simplificação de uma ligação l1 que tem várias entidades-destino

Assim, a simplificação de várias representações de uma ligação que tem diferentes entidades-fonte e várias entidades-destino, processa-se de acordo com as regras conjuntas dos dois casos anteriores. Agrupam-se as entidades-fonte (usando uma circunferência ou uma elipse) e as entidades-destino (usando uma circunferência ou uma elipse) e desenha-se uma ligação da fronteira do primeiro agrupamento à fronteira do segundo, mantendo o símbolo "○ " de vários destinos e o nome da ligação.

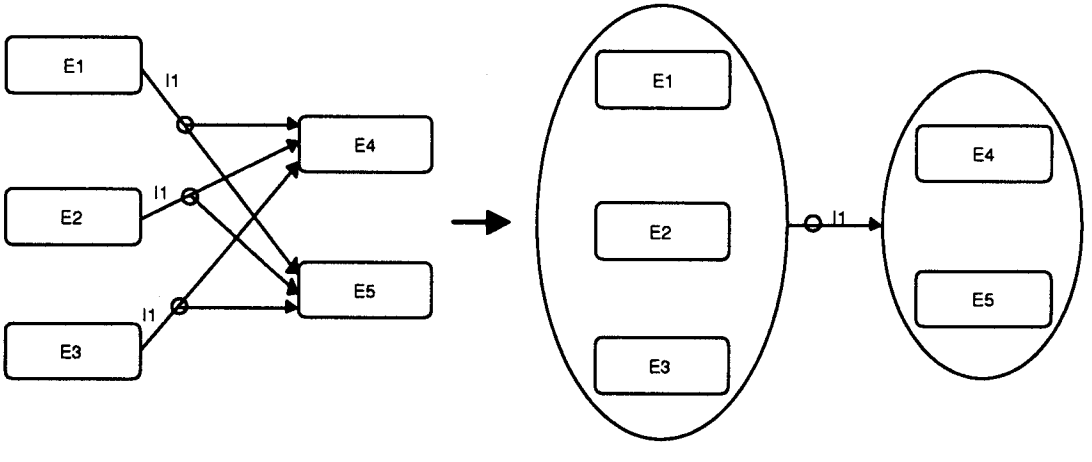


Fig. 4.15 - Simplificação de três ligações l1 (com vários destinos cada uma) que têm diferentes entidades-fonte.

4.3.4 - Exemplo

De seguida mostra-se um exemplo e respectiva explicação de como poderia ser o Diagrama de Ligações relativo ao problema sobre os animais (ponto 4.1).

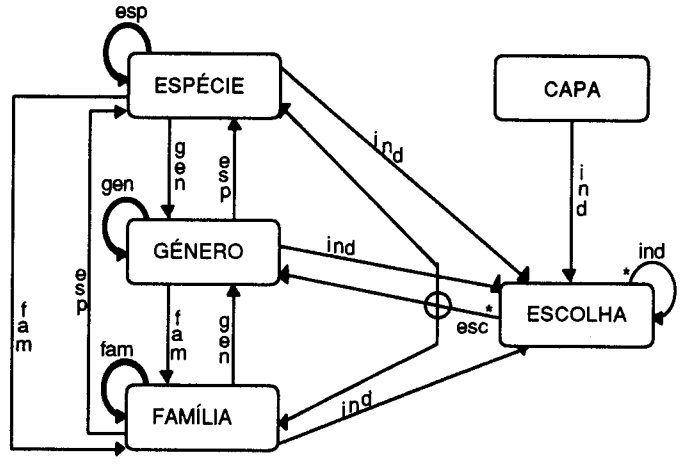


Fig. 4.16 - Diagrama de Ligações não simplificado relativo à aplicação sobre animais.

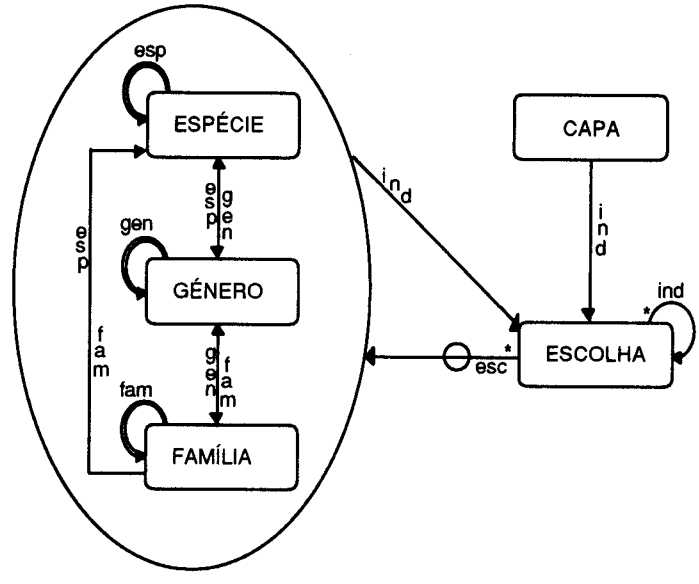


Fig. 4.17 - Diagrama de Ligações simplificado relativo à aplicação sobre animais.

De notar que nem todas as técnicas de simplificação foram usadas - é que elas devem realmente ser usadas se na verdade simplificarem alguma coisa. Como por vezes o diagrama já está suficientemente simplificado sem as utilizar, corre-se o risco de, ao tentar simplificá-lo, acontecer exactamente o contrário. A finalidade é que o diagrama fique o mais "legível" possível fazendo, nesse sentido, as únicas simplificações necessárias.

Proceda-se, então, à interpretação da figura, analisando-a entidade a entidade.

Entidade CAPA

Esta entidade só vai ter uma instância que é o nodo que faz de capa para a aplicação em causa (é o nodo de entrada). Neste nodo vai existir apenas uma ligação ("ind") para a entidade ESCOLHA.

Uma vez que não há nenhuma ligação que tenha a entidade CAPA como destino, quer dizer que não se volta lá naquela sessão. Se a ligação "ind" fosse com volta, então já seria possível voltar lá.

Entidade ESCOLHA

As instâncias desta entidade vão ter opções que permitem escolher o tipo de navegação (por família, género ou espécie) e o sítio por onde se começa essa navegação, i.e., o leitor vai poder escolher qual a família, qual o género, ou qual a espécie que quer visitar primeiro. Assim, cada instância desta entidade tem uma ou mais ligações (situação representada pelo * na ligação "esc"), cada uma das quais tendo como destino uma instância da entidade ESPÉCIE, ou uma instância da entidade GÉNERO, ou uma instância da entidade FAMÍLIA. Adicionalmente, é possível passar de uma instância da ESCOLHA para outras da mesma entidade - as ligações circulares "ind" representam este facto.

Note-se que todas as outras entidades têm definidas ligações para aqui. A verdade é que ESCOLHA é a primeira entidade da aplicação com alguma "função executiva", pois é aqui que começa a navegação determinada pelo utilizador: na entidade CAPA só existia uma ligação pelo que o utilizador não tinha liberdade nenhuma de escolha. Além disso, na realidade, os primeiros nodos "úteis" da aplicação são

instâncias de ESCOLHA: CAPA está ali por uma questão de estética e não por ser mesmo necessária ao bom funcionamento da aplicação.

Entidade ESPÉCIE

A partir de cada instância da entidade ESPÉCIE deve ser possível aceder a outra instância dessa entidade. É por isso que existe lá representado uma ligação "esp" circular. Note-se que esta ligação é "com volta", o que significa que se vai poder "andar para trás", revisitando os nodos já anteriormente visitados. Adicionalmente, a partir de cada instância desta entidade, deve poder-se aceder ao género ou à família correspondente a essa espécie e, por isso, existem lá ligações para as entidades GÉNERO ("gen") e FAMÍLIA ("fam"). Ainda, a partir daqui acede-se à entidade ESCOLHA, através da ligação "ind", pelas razões já acima citadas.

Entidade GÉNERO

A partir de cada instância da entidade GÉNERO deve ser possível aceder a outra instância dessa entidade. É por isso que existe lá representado uma ligação "gen" circular. Note-se que esta ligação é "com volta", o que significa que se vai poder "andar para trás", revisitando os nodos já anteriormente visitados. Adicionalmente, a partir de cada instância desta entidade, deve poder-se aceder a uma espécie desse género e, ainda, à família correspondente a esse género, por isso, existe lá uma ligação para a entidade ESPÉCIE ("esp") e uma ligação para a entidade FAMÍLIA ("fam"). Ainda, a partir daqui acede-se à entidade ESCOLHA, através da ligação "ind", pelas razões já acima citadas.

Entidade FAMÍLIA

A partir de cada instância da entidade FAMÍLIA deve ser possível aceder a outra instância dessa entidade. É por isso que existe lá representado uma ligação circular "fam". Note-se que esta ligação é "com volta", o que significa que se vai poder "andar para trás", revisitando os nodos já anteriormente visitados. Adicionalmente, a partir de cada instância desta entidade, deve poder-se aceder a um género e a uma espécie dessa família e, por isso, existem lá ligações para as entidades GÉNERO ("gen") e ESPÉCIE ("esp"). Ainda, a partir daqui acede-se à entidade ESCOLHA, através da ligação "ind", pelas razões já acima citadas.

4.4 - O Diagrama de Conteúdos

Cada Diagrama de Conteúdos vai especificar a estrutura de composição de uma entidade da aplicação. Assim, vão existir tantos Diagramas de Conteúdos quantas as entidades da aplicação.

Porque se refere a uma estrutura de composição, o Diagrama de Conteúdos é esquematizado em árvore. Adicionalmente, representam-se também aqui as ligações existentes quer dentro de determinada entidade quer para fora dela e também os formatos media utilizados para representação da informação. Existem, ainda, outros conceitos como o de perspectiva e o de espaço que estão intimamente relacionados com a apresentação da informação.

4.4.1 - Entidade

No que diz respeito ao Diagrama de Conteúdos, existem mais alguns conceitos a adicionar ao que já foi definido para entidade exactamente porque aqui se trata da caracterização interna de cada uma delas e não unicamente das inter-relações entre elas, como acontecia no Diagrama de Ligações.

Em cada entidade definem-se os seus:

- componentes (incluindo ligações)
- espaços
- perspectivas (que podem ser ou espaços ou componentes)

Em termos físicos, uma entidade é constituída, apenas e só, por componentes. Em termos lógicos, existem os espaços, que separam os componentes de uma entidade de acordo com o seu carácter. As perspectivas podem ser definidas quer como espaços, quer como componentes.

Esta definição é completada com o Anexo ao Diagrama de Conteúdos que especifica mais detalhadamente alguns dos constituintes de cada entidade.

4.4.1.1 - Componente

Um componente de uma entidade é, como o próprio nome indica, uma parte dessa entidade. Assim como os objectos, os componentes podem ser elementares ou compostos. Um componente elementar, ou atómico, é aquele cuja definição não tem incluídos outros componentes. Um componente composto, pelo contrário, é um componente em cuja definição se encontram referências a outros componentes, i.e., é um componente com sub-componentes. Os **componentes compostos** são divisíveis em novos componentes até se chegar a componentes indivisíveis que são os **componentes elementares**. Em fases posteriores de modelação, um componente vai corresponder a uma classe de objectos.

Nesta abordagem, vai-se considerar ainda uma outra divisão entre os componentes, mais concretamente entre os componentes elementares. Assim, um componente elementar está enquadrado num dos seguintes tipos:

- componente mediático;
- ligação.

Os componentes mediáticos são aqueles que apresentam informação directamente ao utilizador final, em qualquer um dos formatos media considerados. Por exemplo, um texto, uma fotografia ou um filme de vídeo são componentes mediáticos. A cada componente mediático é associado um tipo de acordo com o formato media usado na sua representação.

Definiram-se os seguintes cinco grandes tipos diferentes de classificação deste género de componentes:

- **Texto** ▶ **(T)**[5]
- **Imagem**
  - * **Fixa**
    - ♦ Gráfico
    - ♦ Desenho
    - ♦ Fotografia

    ▶ **(IF)**
  - * **Móvel**
    - ♦ Animação
    - ♦ Vídeo

    ▶ **(IM)**
- **Som**
  - * Ruído
  - * Voz
  - * Música

  ▶ **(S)**
- ***Software*** (ou dados) ▶ **(S/W)**

[5] **T** - Texto
IF - Imagem Fixa
IM - Imagem Móvel
S - Som
S/W - *Software*

Ao contrário dos outros, a animação, o vídeo e os vários tipos de som são **formatos media dependentes do tempo**, uma vez que o seu conteúdo se altera à medida que o tempo avança, sem intervenção de qualquer espécie. Um dos atributos destes objectos será, assim, o tempo de duração.

As ligações são também componentes de uma entidade, mas são diferentes dos componentes mediáticos, porque não vão apresentar informação nenhuma; vão apenas permitir ao utilizador "saltar" de uma instância dessa entidade para outro lado qualquer.

A cada componente (excepto ligações e perspectivas) é associado um nome (em letras maiúsculas) que o identifique claramente de acordo com o seu significado. Um componente é representado graficamente através do seu nome com um caixilho rectangular à sua volta conforme se mostra na figura seguinte:

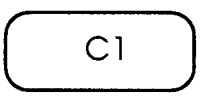


Fig. 4.18 - Notação de representação gráfica de um componente.

Há uma cardinalidade associada aos componentes em termos das suas instâncias, conforme se mostra a seguir:

Notação	**Cardinalidade**	**Explicação**
C1	0 ou 1	Existe 0 ou 1 instância do componente C1.
C1	1	Existe exactamente 1 instância do componente C1.

Quadro 4.3 - Notação de representação da cardinalidade dos componentes.

Notação	Cardinalidade	Explicação
C1 *	0 ou mais	Existem 0 ou mais instâncias do componente C1.
C1 *	1 ou mais	Existe pelo menos uma instância do componente C1 (existem 1 ou mais instâncias do componente C1).

Quadro 4.3 (cont.) - Notação de representação da cardinalidade dos componentes.

Se se pensar, por exemplo, numa aplicação sobre a cidade de Lisboa em que os nodos sobre monumentos tenham várias fotografias do monumento em causa, isso pode perfeitamente ser representado associando ao componente respectivo uma cardinalidade adequada, como se mostra na figura seguinte:

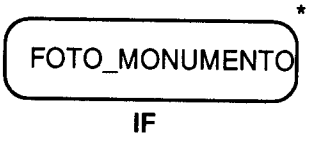


Fig. 4.19 - Componente com cardinalidade "1 ou mais".

Até agora só se considerou uma única relação existente entre os componentes: a relação de composição, i.e., um componente pode ser composto por mais componentes e daí por diante até se chegar aos componentes elementares ou não-decomponíveis. No entanto, por vezes, pode haver a necessidade de se criar um outro tipo de relação entre eles - a **relação de convivência**. Diz-se que dois componentes são **conviventes** se existe entre eles outro tipo de relação significativa que não seja a de composição. A relação de convivência

representa-se graficamente por uma linha ponteada unindo os componentes conviventes conforme se mostra na figura 4.20. Adicionalmente, no Anexo ao Diagrama de Conteúdos dá-se uma explicação mais detalhada dessa relação (ver ponto 4.4.4).

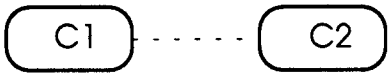


Fig. 4.20 - Existe uma relação de convivência entre o componente C1 e o componente C2, i.e., os componentes C1 e C2 são componentes conviventes.

De notar que aqui não se têm em conta as relações de **herança** entre componentes como se de classes de objectos se tratassem. A razão para isto acontecer é que, mais uma vez, nesta fase de concepção da aplicação, ainda não se faz a estruturação das entidades e dos componentes seguindo todas as regras do paradigma orientado ao objecto.

No paradigma orientado ao objecto, quando se define uma classe de objectos associam-se-lhe determinados atributos e determinado comportamento que vão ser característica de todas as suas instâncias. Do mesmo modo, e porque cada componente vai corresponder a uma classe de objectos, também estes vão ter atributos e um comportamento associados que são definidos no Anexo ao Diagrama de Conteúdos (ver ponto 4.4.4).

4.4.1.2 - Espaço

Qualquer aplicação hipermedia deve, em princípio, manter uma grande coerência à medida que o leitor vai navegando através da rede de nodos que a suporta. O desenho de determinada entidade e a própria noção de nodo como instância dessa entidade vão ajudar a criar esta coerência, pelo menos entre as várias instâncias de cada entidade. Contudo, não basta especificar que objectos existem num écran: é preciso também separá-los de acordo com o seu carácter em relação à respectiva entidade e à globalidade da aplicação. Assim, em relação à maior parte dos nodos, há-de existir um local (no écran) onde estão situadas a maior parte das ligações a outros nodos, outro local privilegiado para apresentar a informação, etc. A cada um destes locais

lógicos do écran que correspondem a um agrupamento de determinado conjunto de objectos (de acordo com o seu carácter) vai chamar-se **espaço**. Não é importante para esta abordagem determinar onde se situam fisicamente, no écran, estes objectos: o que realmente interessa é saber que determinados objectos estão "vocacionados" para realizarem determinadas funções em relação ao resto da entidade e à globalidade da aplicação.

Podem ser criados tantos espaços quantos forem necessários para a definição da entidade em causa. No entanto definem-se já alguns espaços que vão ser comuns à generalidade das entidades:

- Um **espaço de controlo** (os componentes aí definidos mantêm-se sempre em apresentação mesmo que a perspectiva mude - ver ponto 4.4.1.3). É o lugar lógico que vai conter as ligações de qualquer instância dessa entidade para instâncias dessa ou de outras entidades, ou então de ligações de controlo dentro da própria instância, por exemplo para as diferentes perspectivas aí existentes.

- Um **espaço de apresentação**. O espaço de apresentação diz respeito ao lugar lógico onde vai ser apresentada a informação em causa. Este e qualquer outro espaço também pode conter ligações desde que sejam específicas desse espaço.

    * **Espaço comum** (entre as várias perspectivas). Pode haver coisas que são comuns entre as várias perspectivas, no espaço de apresentação, que não vai valer a pena repetir à medida que se muda de perspectiva para perspectiva. Um exemplo disso, pode ser o título e o subtítulo de cada uma das instâncias de determinada entidade.

    * **Espaço de perspectiva**. É onde se apresentam as diferentes visões, ou pontos de vista, relativos ao conceito que o nodo representa.

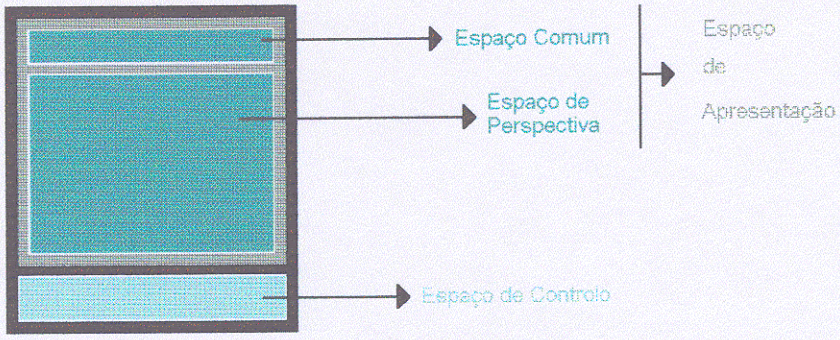


Fig. 4.21 - Definição de alguns espaços de um nodo.

A cada espaço é associado um nome (escrito em letras maiúsculas) de acordo com a sua razão de existir em relação à entidade onde está definido. Um espaço é representado graficamente através do seu nome com uma caixilho rectangular sombreado à sua volta, conforme se mostra na figura seguinte:

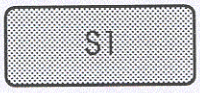


Fig. 4.22 - Notação de representação gráfica de um espaço.

Como exemplo da utilização de espaços, veja-se a seguinte figura:

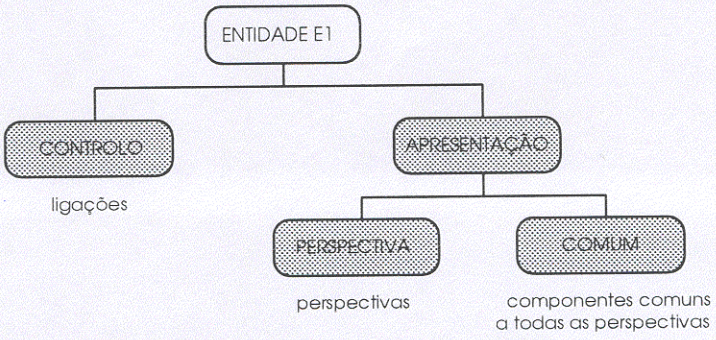


Fig. 4.23 - Utilização dos espaços num Diagrama de Conteúdos

Cada um destes espaços pode, ou não, existir. Podem existir mais espaços que estes. Em princípio o espaço de apresentação existe sempre. Mesmo que o único formato media utilizado seja o som, no modelo, ele deve ser representado no espaço de apresentação. Ás vezes, o espaço de controlo pode deixar de existir e as ligações que porventura existam podem estar definidas no espaço de apresentação. Se se pensar num mapa de uma cidade, por exemplo, que contenha pontos sensíveis que estabeleçam ligações para outros nodos, então o espaço de controlo deixa de existir, pois as ligações encontram-se definidas no espaço de apresentação.

Os espaços não correspondem a objectos: são apenas "lugares" lógicos onde os objectos vão estar "depositados". Deste modo não existe necessariamente nenhuma relação (por exemplo de agregação) entre os componentes pertencentes a um mesmo espaço, a não ser exactamente este facto. Pode-se dizer que um espaço corresponde a um mapeamento lógico dos objectos no écran.

4.4.1.3 - Perspectiva

Como já se disse, uma entidade representa um conceito. A esse conceito podem estar associados pontos de vista diferentes - uma **perspectiva** representa um desses pontos de vista. Assim, a uma entidade podem estar associadas várias perspectivas de acordo com o conceito que ele representa. Por exemplo, quando se descreve um animal, uma perspectiva poderá consistir na visão científica do animal e outra uma mera explicação não-científica e fácil de entender do seu comportamento.

Uma perspectiva tanto pode ser um componente como um espaço. Será um componente se se quiser que numa fase de implementação corresponda a uma classe de objectos. Se, por outro lado, se preferir que exista apenas para separar a informação, então será um espaço.

A cada perspectiva de uma entidade é associado um nome (escrito em letras maiúsculas) que a identifique claramente de acordo com o seu significado. Uma perspectiva é representada graficamente através do seu nome com um caixilho rectangular à sua volta e com um meio-círculo na parte superior do caixilho, conforme se mostra na figura seguinte:

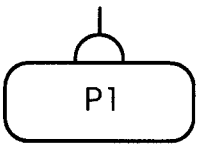


Fig. 4.24 - Notação de representação gráfica de uma perspectiva-componente.

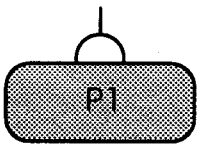


Fig. 4.25 - Notação de representação gráfica de uma perspectiva-espaço.

Quando, através de uma ligação, se passa de uma instância de determinada entidade para outra instância dessa ou de outra entidade e não há informação em contrário, a perspectiva que se encontra activa chama-se a **perspectiva por defeito** dessa entidade. Assim, quando existem duas ou mais perspectivas definidas na mesma entidade, uma delas é a perspectiva por defeito dessa entidade.

Uma perspectiva por defeito é representada graficamente exactamente da mesma forma que uma perspectiva normal, mas com mais um caixilho, conforme ilustra a figura seguinte:

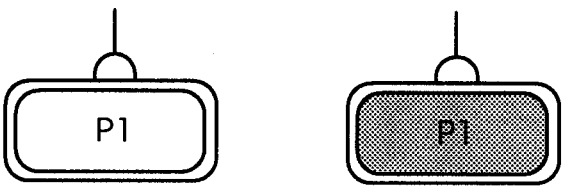


Fig. 4.26 - Notação de representação gráfica de uma perspectiva por defeito (componente ou espaço).

Conforme se verá no ponto seguinte, uma perspectiva pode ser o destino de uma ligação. Estando a visualizar determinada perspectiva de um nodo deve poder-se ir para outra também através de ligações.

4.4.2 - Ligação

Ao contrário do que se passa no Diagrama de Ligações, nos Diagramas de Conteúdos as ligações não são especificadas através de linhas entre as entidades. Estes diagramas têm por objectivo caracterizar cada uma das entidades existentes na aplicação , i.e., cada Diagrama de Conteúdos diz respeito a uma entidade e só a ela. Deste modo, ao invés do Diagrama de Ligações, cada um dos Diagramas de Conteúdos não fornece, por si só, uma visão global de toda a aplicação. No entanto, as ligações são também representadas, mas de outra forma - como componentes!

Os Diagramas de Conteúdos e o Diagrama de Ligações pretendem ser complementares e não ser redundantes na representação da informação. Cada Diagrama de Conteúdos está vocacionado para representar toda a informação relativa a uma entidade, mas não é necessário que represente extensivamente as relações entre as várias entidades da aplicação - isto é função do Diagrama de Ligações! Assim, no Diagrama de Conteúdos, uma ligação é tratada como qualquer outro componente sendo representada a sua cardinalidade, mas não a cardinalidade dos seus destinos. As regras a aplicar às ligações são as mesmas que se aplicam a qualquer outro componente.

Contrariando as mais diversas definições e conceitos, aqui **uma ligação é perfeitamente definida através dos seus destinos e das condições de selecção de cada um deles** (se tiver mais que um), tendo em conta o seu tipo. Assim, sempre que estas características se verificarem, utiliza-se a mesma ligação, resultando daqui vantagens óbvias quer em termos de partilha, quer em termos de reutilização da informação.

Entre os atributos de uma ligação estará, pelo menos, o conjunto de nodos-destino. O comportamento será definido, entre outras coisas, pelas condições de selecção de cada um deles. Se a ligação tiver volta, i.e., se se quiser que seja possível retornar ao nodo de onde se partiu, a informação relativa ao nodo-fonte é-lhe dada na altura em que ela é seleccionada pelo leitor dado que não se controla o conjunto de nodos-fonte de uma ligação.

Os destinos de uma ligação podem ser:

- uma perspectiva da mesma entidade (a mesma instância ou não) , ou
- a mesma entidade (accionando a perspectiva por defeito), i.e., a um nível de abstracção mais baixo, diferentes instâncias da mesma entidade, ou
- entidades diferentes, i.e., a um nível de abstracção mais baixo, instâncias de entidades diferentes.

Um tipo de ligações que não é representado em nenhum dos diagramas é aquele em que as ligações dependem unicamente do contexto dos objectos ou da própria instância em que estão inseridas sendo, por isso, específicas de cada situação. São ligações definidas ao nível da instância e não da classe de objectos. É o caso da explicação do significado de palavras mais complexas ou também de ligações de circunstância entre dois nodos.

O nome de uma ligação que tem como destino uma entidade ou uma perspectiva de uma instância diferente daquela em que está definida) é, aqui, igual ao que essa ligação tem no Diagrama de Ligações. O nome de uma ligação que tem como destino uma perspectiva da mesma instância da entidade em que está definida é o nome dessa perspectiva. De qualquer modo, o nome é sempre escrito em letras minúsculas. Para se representar graficamente, utiliza-se um caixilho rectangular à volta do nome com um triângulo na parte de cima, conforme se mostra na figura seguinte:

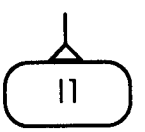


Fig. 4.27 - Notação de representação gráfica de uma ligação.

4.4.4 - O Anexo ao Diagrama de Conteúdos

O Anexo ao Diagrama de Conteúdos é uma parte muito importante da descrição de uma entidade, pois é aqui que se especifica mais detalhadamente coisas que porventura possam não ter ficado suficientemente claras na leitura gráfica do Diagrama de Conteúdos. Aqui descrevem-se com minúcia, usando linguagem natural, as características dos espaços e das perspectivas existentes, as propriedades e os comportamentos dos componentes, assim como as relações de convivência entre eles.

Adicionalmente, junto ao nome de cada componente mediático, e entre parêntesis, deve-se especificar o formato media em que é representado segundo a notação definida no ponto 4.4.1.1.

Para cada Diagrama de Conteúdos sugere-se uma organização em cascata, conforme se mostra na figura seguinte, embora não seja obrigatório que assim seja.

Anexo ao Diagrama de Conteúdos

Entidade: E1

Espaço: S1

Espaço: S1.1

Perspectiva: P1

.....

Componente: C1 (IF)

.....

Componente: C2

Componente: C2.1(T)

.....

Componente C2.2 (F)

.....

Ligação: l1

.....

Convivência (l1, C1)

.....

Espaço S1.2

Componente: .....

Espaço: S2

.....

Fig. 4.28 - Anexo ao Diagrama de Conteúdos para a entidade E1.

Pretende-se que estas especificações sejam detalhadas e informais, mas concisas e claras. Foi tomada a decisão neste sentido porque esta aproximação é dirigida a uma equipe pluridisciplinar onde há muita gente que não tem formação específica em informática. As pessoas vão ter que comunicar entre si usando uma linguagem comum, que não deve ser específica de nenhuma área de estudo, mas antes uma linguagem que todos entendam e, mais que isso, que todas aprendam sem grandes dificuldades. Deste modo, não parece correcto "obrigar" toda a equipe a falar "informático" ou um qualquer sucedâneo dele, tentando obrigá-la a comunicar numa linguagem que a maior parte das pessoas não entende e, muito provavelmente, não tem formação de base, tempo e disponibilidade para aprender. Um dos objectivos desta aproximação é realmente que as pessoas possam manuseá-la com facilidade, e não fazer uma coisa que sendo compreendida só por especialistas em informática, acabe por não ser usada por ninguém dada a pluridisciplinaridade exigida na equipa de concepção de aplicações hipermedia.

Assim, prefere-se que cada um dos espaços, perspectivas, componentes e relações de convivência sejam especificados informalmente, mas de forma clara e concisa de modo a que todos, sem dúvidas, entendam o que está lá escrito e, também muito importante, de modo a que quem vai trabalhar com estes diagramas numa fase posterior do ciclo de vida da aplicação, também os possa entender sem dificuldades.

4.4.5 - Exemplo

De seguida mostram-se alguns exemplos de Diagramas de Conteúdos. Estes exemplos ilustram uma das soluções que poderia ter o problema dos mamíferos anteriormente exposto.

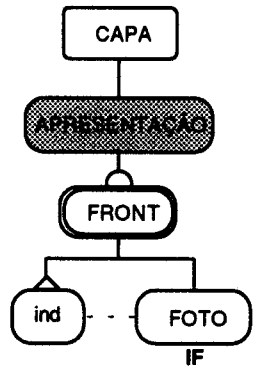


Fig. 4.29 - Diagramas de Conteúdos para a entidade CAPA.

Anexo ao Diagrama de Conteúdos

Entidade: CAPA

Espaço: APRESENTAÇÃO

Perspectiva: FRONT

Componente: FOTO (IF)
Fotografia de capa que faça subentender o assunto tratado. Deve ser apelativa.

Ligação: ind
Destino: entidade ESCOLHA.

Relação de convivência: ind - FOTO

O componente mediático FOTO funciona como âncora da ligação "ind". Clicando em qualquer ponto de uma instância do componente mediático FOTO deve poder passar-se para o destino da ligação que é uma instância da entidade ESCOLHA.

A entidade CAPA é usada para modelar o primeiro nodo da aplicação - a *front cover*. Assim, esta entidade só vai ter uma instância. Conforme se pode ver, aqui não há espaço de controlo, só existe espaço de apresentação no qual está incluído uma perspectiva. Como esta é a única perspectiva da entidade e não há mais nenhum espaço dentro do espaço de apresentação (espaço comum por exemplo), é escusado incluir o espaço de perspectiva no diagrama, uma vez que se estaria a complica-lo sem razão nenhuma.

Na perspectiva por defeito, a única, existe uma ligação e um componente mediático do tipo IF que estão ligados por uma linha. Conforme já se explicou atrás, esta linha significa que há uma relação de convivência entre os dois. Deste modo, o componente mediático em causa passa a funcionar como âncora da dita ligação.

Veja-se, no entanto, outro Diagrama de Conteúdos diferente:

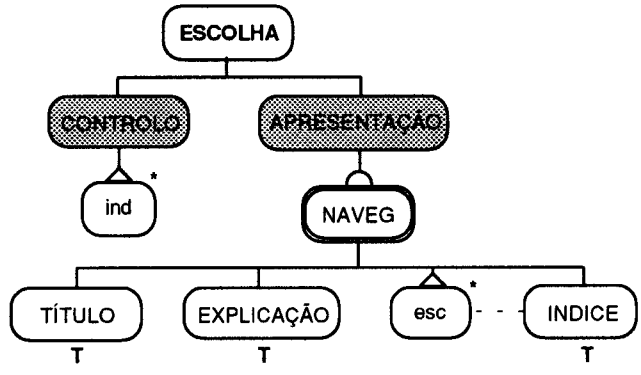


Fig. 4.30 - Diagrama de Conteúdos para a entidade ESCOLHA.

Anexo ao Diagrama de Conteúdos

.....

Espaço: APRESENTAÇÃO

.....

Ligação: esc

Destinos: entidade FAMÍLIA, entidade GÉNERO ou entidade ESPÉCIE

caso (titulo do nodo) = "Índice de Géneros"

então destino = instância da entidade GÉNERO

(título do nodo) = "Índice de Espécies"

então destino = instância da entidade ESPÉCIE

senão destino = instância da entidade FAMÍLIA

.....

Relação de convivência: esc - ÍNDICE

A cada opção do índice está associada uma ligação "esc" respectiva a essa opção.

.....

Aqui, o que chama mais a atenção, por ser diferente, é a ligação "esc". Comece-se, no entanto, pelo princípio. Nesta entidade já há um espaço de controlo, no qual existem uma ou mais ligações "ind" para ESCOLHA. Cada uma destas ligações tem um único destino: uma instância da entidade ESCOLHA. Também aqui não se especifica o espaço de perspectiva pelas mesmas razões do que acontecia na entidade CAPA acima descrita.

Na perspectiva NAVEG encontram-se um texto explicativo do que consiste cada instância ou nodo, o seu título, um índice e várias ligações "esc". Estas ligações são conviventes com outro componente (o componente ÍNDICE): neste caso a relação de convivência quer significar que a cada opção do índice está associada a ligação respectiva. Note-se que aqui, cada uma das ligações tem definidas três entidades-destino - ou a entidade FAMÍLIA, ou a entidade GÉNERO ou, ainda, a entidade ESPÉCIE. A selecção da instância destino vai ser feita de acordo com as condições especificadas na definição da ligação no anexo ao Diagrama de Conteúdos.

Como último exemplo, apresenta-se o Diagrama de Conteúdos da entidade ESPÉCIE.

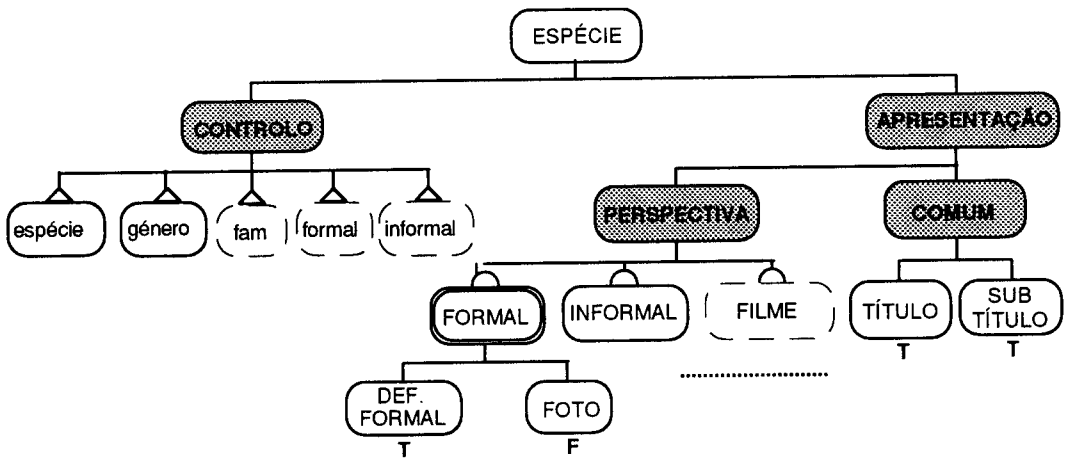


Fig. 4.31 - Diagrama de Conteúdos para a entidade ESPÉCIE.

Anexo ao Diagrama de Conteúdos

Entidade: ESPÉCIE

Espaço: CONTROLO

.....

Ligação: formal
destino = perspectiva FORMAL
Só existe se a perspectiva corrente não for a FORMAL.

.....

Espaço: APRESENTAÇÃO

Espaço: COMUM

Componente: TITULO
O titulo deve ser comum a todas as instâncias (nodos) da entidade ESPÉCIE e deve ser o seguinte: "ESPÉCIE".
Componente: SUBTÍTULO
Deve haver um subtítulo em todas as instâncias (nodos) da entidade ESPÉCIE. Este subtítulo varia de acordo com a espécie que está a ser descrita.

O Diagrama de Conteúdos desta entidade é o mais alargado, mas nem por isso o mais complexo. Veja-se, então: o espaço de controlo tem definidas cinco ligações, três para entidades e duas para perspectivas. As três primeiras são ligações vulgares - uma ligação, um destino. As ligações "formal" e "informal" também não são de muito difícil interpretação: podem, ou não, existir - só não existem quando o leitor está na perspectiva FORMAL ou INFORMAL, respectivamente.

Há três perspectivas diferentes - duas delas não estão representadas de modo a que a leitura do diagrama se torne mais simples. A perspectiva por defeito é a perspectiva FORMAL. Aqui, além de haver uma fotografia de um animal de dada espécie existe uma definição científica, ou formal, do animal.

Note-se que neste diagrama já se representa o espaço de perspectiva e isto acontece porque já há mais que uma perspectiva e há um espaço comum que, como se disse antes, é onde está definido tudo o que é comum a todas as perspectivas. Neste espaço comum existem dois componentes mediáticos do tipo T: um título e um subtítulo. O título é sempre o mesmo para todas as instâncias (nodos) desta entidade e o subtítulo varia de instância para instância.

4.5 - Metodologia a seguir

Vai agora apresentar-se genericamente as fases que poderão ser realizadas para aplicação da técnica de representação do conhecimento aqui especificada.

Num primeiro estádio ter-se-ão as seguintes grandes fases:

1. **Definição das entidades.**

   Nesta fase devem separar-se os conceitos a representar na aplicação. Note-se que nesta separação a semântica tem um grande peso, pois apesar de se poder ter duas entidades sintacticamente iguais, elas podem representar diferentes conceitos, com significados distintos, pelo que devem continuar a existir as duas.

   É preciso garantir que:

   - não há duas ou mais entidades definidas sob a capa de uma única e, ao contrário,
   - não há uma única entidade definida duas ou mais vezes.

Veja-se o seguinte exemplo em entidades correspondentes ao problema que se tem vindo a tratar:

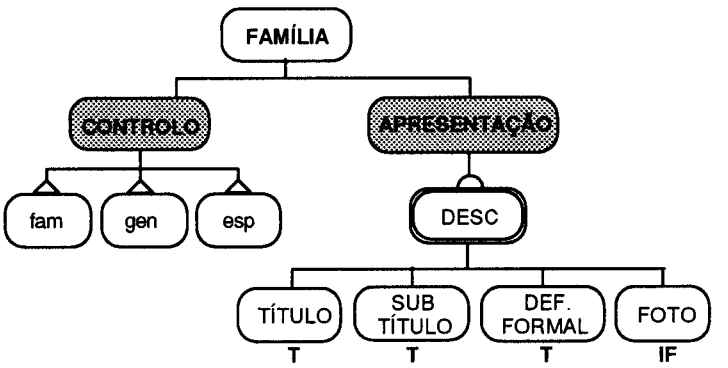


Fig. 4.32 - Diagrama de Conteúdos para a entidade FAMÍLIA.

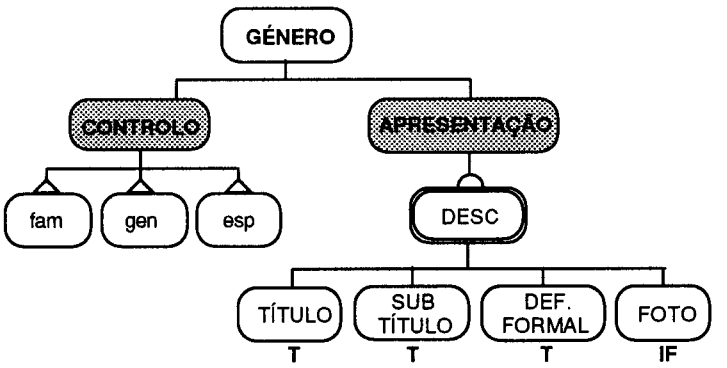


Fig. 4.33 - Diagrama de Conteúdos para a entidade GÉNERO.

Conforme se pode ver, estas entidades são exactamente iguais em termo dos seus constituintes. Até as ligações são as mesmas, têm as mesmas entidades-destino e a mesma cardinalidade quer a nível das próprias ligações quer a nível dos seus destinos. Então porque é que não se utiliza só uma entidade em vez de duas? Porque, pura e simplesmente elas estão associadas a conceitos diferentes, sendo cada um desses conceitos suficientemente relevante dentro do âmbito de abordagem da aplicação.

Note-se que nos requisitos da aplicação se fala em consulta por família, género e espécie - distinguem-se bem os três tipos de consulta! Adicionalmente, em termos de significado, consultar uma instância da entidade

GÉNERO não é, seguramente, a mesma coisa que consultar uma instância da entidade FAMÍLIA. Por exemplo, deve-se poder mostrar com clareza que a partir de instâncias da entidade ESCOLHA é possível fazer cada um dos três tipos de consulta. Adicionalmente, quer-se que a partir de qualquer instância da entidade FAMÍLIA se possa saltar para uma instância da entidade GÉNERO correspondente à família em causa e vice-versa. Se em vez daquelas duas entidades existir só uma que as represente, estas ligações não ficam claras tornando-se bastante fácil cair em erros de concepção e, posteriormente de realização (implementação). Assim, porque representam coisas diferentes e, porque possuem significados diferentes, apesar de serem iguais em termos sintácticos e de implementação e de se poderem, mais tarde, associar a uma única classe de objectos, nesta fase estas entidades devem ser representadas separadamente.

2. Definição das ligações existentes entre essas entidades.

Depois de se ter definido as entidades é necessário definir as ligações existentes entre elas, tendo em conta a cardinalidade das próprias ligações e a cardinalidade dos seus destinos.

Assim, deve-se olhar para cada entidade existente e ver que ligações a deverão ter como fonte e que ligações a deverão ter como destino. Se isto se fizer em todas as entidades, funciona como garantia de que as ligações que se representam são exactamente as que se querem representar.

Neste ponto já se tem o Diagrama de Ligações completo. Se se quiser, e se se achar necessário deve agora proceder-se às simplificações. A partir deste momento começa-se a trabalhar nos Diagramas de Conteúdos que devem estar coerentes com o Diagrama de Ligações. De qualquer modo é importante frisar que de qualquer fase posterior é possível voltar à anterior e, sempre que em determinado momento de realização de um Diagrama de Conteúdos se der conta que é necessário alterar o Diagrama de Ligações então deve-se realmente alterá-lo, procedendo em seguida à alterações daí decorrentes (se as houver) nos outros Diagramas de Conteúdos.

Num segundo estádio, para cada entidade deve-se proceder a:

3. **Definição dos espaços.**

   Devem definir-se os espaços, ou a separação lógica dos componentes dentro da entidade. Podem considerar-se os espaços previamente definidos, subtrair alguns ou adicionar outros, de acordo com as necessidades de caracterização da entidade em causa.

4. **Definição das perspectivas.**

   Pensando na entidade como abstracção em relação a um conjunto de nodos, deve-se definir as perspectivas de apresentação que essa entidade possuirá.

5. **Definição dos componentes e das relações de convivência.**

   Nesta fase devem definir-se os componentes de acordo com o espaço a que pertencem e a perspectiva em que estão inseridos (se for o caso).

   A cada um dos componentes mediáticos deve associar-se um conjunto de caracteres que define a que tipo de formato media diz respeito (ver ponto 4.4.1.1).

   As ligações, como componentes que são, devem também ser introduzidas no respectivo Diagrama de Conteúdos nesta altura, tendo em conta o Diagrama de Ligações e as restrições de coerência impostas por ele. Sempre que se quiser alterar alguma ligação, esta alteração deve-se reflectir não só no Diagrama de Ligações, mas também nos outros Diagramas de Conteúdos afectados.

   É também nesta fase que se definem as relações de convivência entre os componentes.

6. Realização do Anexo ao Diagrama de Conteúdos.

Por fim, faz-se o Anexo ao Diagrama de Conteúdos tendo o cuidado de especificar as propriedades e comportamentos de todos os componentes elementares e dos componentes compostos que de tal necessitem. Adicionalmente devem caracterizar-se todas as relações de convivência existentes no Diagrama de Conteúdos correspondente.

Com este ponto termina a realização do Diagrama de Conteúdos. Note-se que este e o Diagrama de Ligações devem estar em conformidade.

Tendo completado os dois diagramas termina esta fase do desenvolvimento de uma aplicação hipermedia, podendo passar-se às fases seguintes do ciclo de vida da aplicação.

V - TESTES E RESULTADOS

Neste capítulo vai-se descrever alguns dos testes realizados à abordagem apresentada no capítulo anterior, bem como analisar os resultados obtidos. Posteriormente vai-se comentá-la e avaliá-la usando, para isso, os mesmos parâmetros que se usaram para classificar as outras abordagens apresentadas no terceiro capítulo.

5.1 - Testes realizados

A abordagem aqui apresentada foi, durante o seu desenvolvimento, testada em alunos do Ensino Superior (Escola Superior de Educação de Fafe - E.S.E.F.) não especialistas em informática. Foi-lhes pedido que modelassem determinadas aplicações hipermedia e depois as implementassem em *ToolBook* .

Constituíram-se nove grupos cada um dos quais com dois ou três alunos. As aplicações desenvolvidas foram as seguintes:

a1. Roteiro turístico de Guimarães

a2. Roteiro turístico de Felgueiras

a3. Catálogo de uma discoteca

a4. Leis autárquicas

a5. Apresentação da E.S.E.F.

a6. Ensino Assistido por Computador - MS-DOS

a7. Ensino Assistido por Computador - MICROSOFT WINDOWS

a8. Ensino Assistido por Computador - BASIC

a9. Ensino Assistido por Computador - PASCAL

Foram testados cinco parâmetros, a saber:

p1. **Facilidade de aprendizagem da metodologia.** Testar até que ponto fazia sentido a utilização conjunta dos dois tipos de diagramas e do anexo ao diagrama de conteúdos e as relações entre eles.

p2. **Facilidade de apreensão da notação.** Testar até que ponto a notação estava explícita.

p3. **Coerência da notação.** Testar a coerência da notação através do seu uso correcto ou não. Verificar se a relação entre os vários conceitos em hipermedia correspondia a uma relação na notação e se esta era utilizada correctamente.

p4. **Conformidade entre a notação desenvolvida e as necessidades reais.** Testar até que ponto a notação desenvolvida cobria as necessidades reais sentidas pelos autores de aplicações hipermedia.

p5. **Aplicabilidade da metodologia.** Testar se valia a pena utilizar a metodologia para o desenvolvimento de aplicações hipermedia.

Optou-se por não se lhes explicar nada verbalmente sobre a metodologia: foi-lhes entregue uma versão anterior do capítulo IV deste trabalho, pedindo-lhes que a lessem e que tomassem em atenção os exemplos dados. Marcaram-se reuniões semanais com cada um dos grupos a fim de acompanhar o trabalho deles. Obtiveram-se os seguintes resultados:

Parâmetro p1 - Facilidade de aprendizagem da metodologia

Todos os grupos compreenderam a finalidade dos diagramas existentes, assim como as relações entre eles. As fases da metodologia foram correctamente aplicadas. Como em qualquer outra metodologia de análise, é possível voltar a uma fase anterior de acordo com modificações realizadas em fases posteriores, coisa que todos os grupos faziam à medida que iam progredindo na definição do modelo da aplicação.

Parâmetro p2 - Facilidade de apreensão da notação

A maior parte dos grupos teve dificuldades em apreender a notação. O significado de cada parte da notação era explícito, mas à custa de se querer fazer uma coisa muito completa acabou por se perder em simplicidade, razão pela qual a maior parte dos alunos sentiu dificuldade em compreender as variantes utilizadas. Posteriormente, tendo por base estes resultados e as críticas dos alunos, introduziram-se as alterações que se acharam necessárias e pertinentes.

Parâmetro p3 - Coerência da notação

No início vários grupos sentiram dificuldades na utilização da notação. Ajudou-se-lhes a fazer parte do Diagrama de Ligações para a aplicação em causa e explicou-se a alguns um exemplo de um Diagrama de Conteúdos. Posteriormente todos os grupos completaram o Diagrama de Ligações e fizeram os Diagramas de Conteúdos com os respectivos anexos sem ajuda.

Mostraram-se satisfeitos por conseguirem através da notação exprimir a aplicação que queriam desenvolver. Além disso acharam muito bom poderem discutir uns com os outros tendo a percepção exacta do que estavam a fazer e do que cada um queria dizer.

A notação provou ser coerente dada a interligação correcta entre os vários diagramas quase encaixando como peças de um puzzle.

Parâmetro p4 - Conformidade entre a notação desenvolvida e as necessidades reais.

A cada grupo foi exigido que apresentasse no final um relatório no qual incluiria o modelo da aplicação desenvolvida. A aplicação foi apresentada e comentada tendo presente todos os elementos do grupo. Comparou-se o modelo apresentado no relatório com a aplicação final. Em quase todos os grupos o modelo estava coerente e em conformidade com aplicação final. Por vezes havia pequenas falhas na utilização da notação. Houve apenas um grupo que apresentou algumas discrepâncias entre o modelo apresentado e a aplicação. Anotaram-se todas as críticas relevantes para a continuação do trabalho: em geral diziam respeito

a pequenas falhas na notação. No geral o resultado foi bastante satisfatório dando novo ânimo à continuação da pesquisa.

Parâmetro p5 - Aplicabilidade da metodologia

Depois de terem sido avaliados, pediu-se a todos os alunos que fizessem críticas à aplicabilidade da metodologia. Basicamente, pediu-se que dissessem se realmente a metodologia os ajudou ao posterior desenvolvimento da aplicação. Todos foram unânimes ao dizerem que apesar de haver ainda algumas falhas (especialmente na notação) a utilização da metodologia constituiu para eles uma grande ajuda principalmente porque lhes deu «algo por onde começar» e lhes forneceu uma base sólida e uma linguagem comum de discussão.

Adicionalmente, trocaram-se os relatórios e todos os elementos de todos os grupos conseguiram compreender como estava estruturada a aplicação de cada um dos grupos afirmando que não lhes seria muito difícil fazer a manutenção de qualquer uma das aplicações tendo como base o respectivo modelo.

Tendo como base os resultados obtidos e as críticas feitas, alterou-se a notação especialmente no sentido de a simplificar. Apesar de alguns dos resultados não serem totalmente fiáveis (por a amostra ser pequena e por naquela altura - fim de ano lectivo - os alunos se encontrarem sob grande pressão escolar - frequências, exames e trabalhos - e ainda pelo facto de alguns apenas estarem interessados em "fazer a cadeira") fizeram-se grandes alterações de acordo com as críticas mais pertinentes, pois como diz o velho ditado, "onde há fumo há fogo".

Apesar de depois destas alterações não ter sido possível, por várias razões, fazer mais testes do tipo dos atrás apresentados, várias pessoas acreditadas na matéria tiveram acesso às sucessivas versões do documento criticando e dando opiniões que se crê pertinentes. Acredita-se que as alterações mais significativas e importantes foram concretizadas. Sabe-se que ainda há coisas a melhorar; que muito do trabalho realizado foi a "desbravar caminho". Mas, mais que tudo, há uma convicção profunda de que, pelo menos, agora as pessoas partem mais seguras quer para o desenvolvimento, quer para a documentação e a manutenção de uma aplicação hipermedia.

5.2 - Avaliação segundo os parâmetros utilizados para as outras abordagens

- **Possibilidade de se estender a partir de um modelo baseado no grafo orientado**

  O Diagrama de Ligações é uma clara extensão ao modelo do grafo orientado. Baseia-se nele e introduz novas regras e características de acordo com as necessidades sentidas no geral das aplicações hipermedia.

- **Facilidade de uso**

  Provou ser de fácil uso tanto para informáticos (que já estão habituados a ter que modelar, pelo menos, aplicações construídas em sistemas tradicionais) como para não-informáticos. Este último ponto é muito importante pois um dos objectivos desta abordagem é que seja algo simples a ser usado por uma vasta equipa composta tanto por informáticos como por não-informáticos. Corresponde a uma fase inicial de análise da aplicação realizável por toda a equipa interdisciplinar, razão pela qual deve ser simples e fácil de usar.

- **Semântica de *browsing* associada**

  Segundo [GAR91], a **semântica de *browsing*** «fornece o comportamento em *runtime* da aplicação».

  Apesar de não existirem definidas primitivas para um semântica de *browsing* associada à abordagem aqui descrita, ao nível de abstracção da entidade é perfeitamente possível prever o comportamento da aplicação em *runtime*.

- **Suporte de diferentes hipervisões**

  Uma vez que prevê e modela a utilização de ligações com vários destinos, suporta diferentes hipervisões da mesma hiperbase.

- **Suporte de várias versões**

  Já que hipermedia significa também reutilização da informação a nível de nodos, um versão nova aproveitaria os nodos comuns com a anterior. Assim, o suporte de várias versões reduz-se ao caso de diferentes hipervisões sobre a mesma hiperbase.

- **Possibilidade de modelar ligações dinâmicas (ligações estabelecidas em *runtime)***

  Podem ser distinguidos dois tipos de ligações dinâmicas: aquelas que são pré-especificados mas que só se estabelecem mediante determinadas condições ou as que se estabelecem a partir de informações dadas pelo utilizador (resultantes de *queries*, por exemplo). Este parâmetro diz respeito aos primeiros, dado que os outros dependem da vontade do utilizador.

  A possibilidade de utilização de ligações com vários destinos implica que se podem modelar ligações dinâmicas. A selecção de cada um dos destinos é feita de acordo com um conjunto de condições pré-especificado pelo utilizador.

  O facto de se possibilitar que uma ligação possa, ou não, existir também implica que este ponto seja verdadeiro. Assim, determinada ligação tem uma existência de acordo com determinadas condições, como por exemplo, o perfil do utilizador.

- **Possibilidade de usar a mesma estrutura com conteúdos diferentes**

  O conteúdo é perfeitamente separado da estrutura, razão pela qual se pode usar a mesma estrutura (associada ao Diagrama de Ligações) com diferentes conteúdos (associados a cada um dos Diagramas de Conteúdos).

- **Especificação dos diferentes formatos media usados**

  O Diagrama de Conteúdos não distingue todos os formatos media usados, mas sim os vários tipos de formatos media. Optou-se por esta solução para simplificar o modelo. Embora seja menos preciso, não é necessário os analistas saberem de cor a especificação de cada um dos formatos media implicados, mas apenas a do seu tipo e, além disso, permite uma maior flexibilidade para as fases posteriores do ciclo de vida da aplicação.

- **Implementação em sistema autor próprio**

  Não. Poderá ser algo a realizar no futuro.

- **Facilidade de entender a estrutura global da aplicação modelada**

  Através dos testes realizados com os alunos (com uma versão anterior menos simples), comprovou-se que se percebia com facilidade a estrutura global da aplicação modelada. De facto, basta saber "ler" o Diagrama de Ligações (o que não é muito difícil) para se perceber de forma clara e simples a estrutura global da aplicação hipermedia.

Existe, ainda, um conjunto de características desta abordagem, que não foram atrás mencionadas e que são consideradas importantes:

- As ligações são consideradas objectos e, mais que isso, é considerado que uma ligação não tem informação sobre o nodo onde está colocada (nodo-fonte), mas apenas sobre o seu conjunto de nodos-destino. O que realmente importa é o conjunto de nodos-destino e as condições de selecção de cada um deles. Desta forma é possível utilizar a mesma ligação em vários nodos-fonte (pois ela é independente destes), i.e., é possível reutilizar as ligações.

- Tanto os nodos como os seus componentes são objectos, o que torna possível determinado nodo usar um ou mais componentes previamente definidos noutros nodos. Há a possibilidade de reutilizar componentes sejam eles componentes compostos ou elementares (mediáticos ou ligações).

- Prevê-se a modelação de vários tipos de ligações: 1º o facto de se poder ter 0, 1 ou mais ligações entre quaisquer duas entidades; 2º o facto de qualquer ligação poder ter, ou não, vários destinos possíveis, tanto para instâncias da mesma entidade, como para instâncias de quaisquer outras entidades; 3º o facto de se poder ter, ou não, ligações com volta.

- Aos componentes mediáticos também pode ser associada uma cardinalidade. Deste modo, pode-se modelar o facto de um componente poder ou não existir em determinada instância de uma entidade. Mais ainda, pode-se modelar o facto de existirem, em dado nodo, uma ou mais instâncias de determinado componente (por exemplo, várias fotografias de um monumento).

- As entidades e, por consequência as suas instâncias, podem ter várias perspectivas de apresentação. Assim, um restaurante, por exemplo, pode ser descrito tendo como perspectiva o tipo de comida que possui, o tipo de decoração, ou ainda, o tipo de atendimento que proporciona aos clientes. É ainda de salientar a possibilidade de modelar a perspectiva por defeito de determinada entidade e o facto de dada perspectiva poder, ou não, existir.

- A definição de espaço de uma entidade é também um ponto de realce. Deste modo, pode-se separar a informação logicamente, i.e., os componentes (incluindo ligações) são separados de acordo com a sua função no nodo.

- A relação de convivência entre dois componentes permite estabelecer outra relação entre eles que não seja a relação de composição. Por exemplo, num índice, cada linha pode estar relacionada com uma ligação de modo a que seja a âncora dessa ligação - aqui utiliza-se uma relação de convivência entre esses dois componentes.

- Pode-se modelar anotações usando componentes mediáticos do tipo T (texto).

- Para melhor compreensão e detalhe do modelo, utiliza-se o Anexo ao Diagrama de Conteúdos onde se faz uma descrição detalhada de cada entidade, seus espaços, perspectivas, componentes mediáticos, ligações e relações de convivência.

VI - CONCLUSÃO

A utilização de uma metodologia torna-se cada vez mais crucial para o desenvolvimento com sucesso de qualquer tarefa. Mesmo para as tarefas mais simples do quotidiano utilizam-se métodos de abordagem os quais, por estarem tão bem interiorizados na nossa mente, nos passam despercebidos. Há, contudo, tarefas, que pela sua complexidade, ou até pela falta de prática de quem as executa, necessitam de ser realizadas utilizando uma metodologia científica específica.

Se até há bem pouco tempo se ouvia dizer em qualquer lado que as aplicações hipermedia podiam ser construídas sem nenhum tipo de planeamento prévio ou que os nodos e ligações deveriam ser acrescentados intuitivamente segundo a nossa forma de pensar, agora já existem muitas vozes discordantes. É verdade que, em princípio, se poderia associar um nodo hipermedia a cada fragmento cognitivo de uma determinada pessoa e relacionar os vários nodos da mesma forma que esses fragmentos estão relacionados na maneira de pensar dessa pessoa. No fundo, a aplicação hipermedia assemelhar-se-ia a um mapeamento da estrutura cognitiva de um dado indivíduo numa hiperbase - e o problema reside exactamente aí: "**um dado indivíduo**" que tem uma estrutura cognitiva diferente da de todos os outros! Este aumentar intuitivo da hiperbase resulta, teoricamente, para um indivíduo: o que a aumentou! Assim sendo, para o desenvolvimento de uma aplicação hipermedia a ser utilizada por um vasto grupo de pessoas, é necessário procurar pontos em comum na estrutura cognitiva da maioria dos potenciais utilizadores, defini-los, relacioná-los e modelar a aplicação de acordo com eles.

O problema da não-utilização de uma metodologia no desenvolvimento de uma dada tarefa agudiza-se ainda mais quando esta deverá ser executada em equipa: às tantas está cada pessoa a desenvolver parte da tarefa ao seu modo havendo uma interacção nula, ou quase nula, entre os vários elementos da equipa. Ninguém sabe ao certo o que cada um dos outros está a fazer nem como o está a fazer. As coisas ficam de tal forma desorganizadas que, ao fim de algum tempo, se corre o risco de ter entre mãos um volume de informação disforme e incontrolável.

Associada a qualquer metodologia existe uma linguagem de expressão dos conceitos (e suas inter-relações) que ela utiliza - esta passará a ser a linguagem que a equipa utiliza para comunicar entre si. A abordagem aqui descrita centra-se essencialmente na definição de uma técnica de representação do conhecimento e na respectiva linguagem de suporte para a modelação de aplicações hipermedia.

Sabia-se desde o princípio que a tarefa não seria fácil: em primeiro lugar pela pouca experiência que as pessoas ainda tinham nesta área comparando com a que existia em outro tipo de sistemas e, depois, porque um trabalho desta natureza geralmente é realizado em equipa e com várias interacções de outros profissionais. De qualquer modo o desafio era tentador não só por se sentir a necessidade de uma abordagem deste género, mas também pela vontade de fazer algo realmente válido e que pudesse ser útil no futuro e, essencialmente, pela necessidade de aumentar o próprio conhecimento.

A primeira versão da notação era coerente mas bastante mais complicada: existiam várias coisas que se deu conta não serem necessárias e existia muita redundância entre os dois tipos de Diagramas. Á medida que se ia caminhando de versão em versão, ia-se simplificando a notação e, por vezes, introduzindo coisas ainda não previstas. Os "designers" costumam dizer que o logotipo mais difícil de fazer é o logotipo simples - aqui passa-se exactamente o mesmo: chegar a algo que se crê simples e claro foi difícil e implicou muito envolvimento e muita dedicação.

Como já se disse anteriormente, sabe-se que ainda há muita coisa a fazer nesta área. Um dos pontos essenciais é a modelação do "tempo", não só em relação aos formatos media dependentes do tempo, mas também em relação à dinâmica geral de uma aplicação hipermedia. Outro ponto de estudo futuro poderá ser a definição de uma semântica de *browsing* própria. Seria também interessante começar a pensar em esquematizar um sistema autor hipermedia baseado senão total, pelo menos parcialmente nesta abordagem.

Tinha que se começar por algum lado e foi por aqui que se escolheu começar. Foi um começo difícil dada a escassez de bibliografia apropriada. Desbravou-se muito caminho, descobriu-se muita coisa ainda para fazer e, provavelmente daqui a uns tempos vai-se querer alterar várias coisas nesta abordagem, não se sabe. De qualquer modo, uma coisa é certa: vai-se continuar a trabalhar nesta área e, mais que isso, vai-se continuar a trabalhar com a firme intenção de explorar novos caminhos, de dar um contributo, por pequeno que seja, a este imenso e magnífico mundo dos sistemas hipermedia.

VII - BIBLIOGRAFIA

[AH91] ANDERSON, D.P.; HOMSY, G - *A Continuous Media I/O Server and Its Synchronization Mechanism.* «COMPUTER», Outubro 1991, p.51-57.

[ALP92] ALPERT, Mark - *CD-ROM: The Next PC Revolution.* «FORTUNE», Junho 1992, p.54-59.

[BAR92] BARKER, Philip - *An Object-Oriented Approach to Hypermedia Authoring*, in «Interactive Multimedia Learning Environments», Berlim, 1992, p. 132-12.

[BC88] BEGEMAN, Michael L.;CONKLIN, Jeff - *The Right Tool for the Job.* «BYTE», Outubro 1988, p.255-266.

[BD92] BR0NDMO, Hans Peter; DAVENPORT, Glorianna - *Creating and viewing the Elastic Charles - a hypermedia journal*, in «Interactive Multimedia Learning Environments», Berlim, 1992, p. 43-51.

[BEY92] BEYNON-DAVIES, P. - *Entity models to object models: object-oriented analysis and database design.* «Information and Software Technology», Reino Unido, Vol.34, No.4, Abril 1992, p. 255-262.

[BP89] BJORK, Bo-Christer; PENTILLA, Hannu - *A scenario for the development and implementation of a building product model standard.* «Adv. Eng. Software», Vol.11, No.4, Outubro 1989, p.176-187.

[BS90] BOYLE, Craig D. B.; SNELL, James R. - *Intelligent navigation for semistructured hypertext documents*, in «Hypertext: State of the Art», Londres, ABLEX - Publish Corporation, 1990, p. 28-41.

[CAS90] CASSAGNE, Jean - *ToolBook: L'HyperCard de Windows.* «SCIENCE & VIE MICRO», No.76, Outubro 1990, p.132-136.

[CHE90] CHEN, John C., et al. - *Querying an object-oriented hypermedia system*, in «Hypertext: State of the Art», Londres, ABLEX - Publish Corporation, 1990, p. 231-238.

[CHR91] CHRISMENT, Claude. - *Hypertext: Overview - Formalization.* Outubro 1991.

[CON87] CONKLIN, Jeff - *Hypertext: An Introduction and Survey.* «COMPUTER», Setembro 1987, p.17-41.

[CY90] COAD, Peter; YOURDAN, Edward - *Object-Oriented Analysis.* New Jersey. YOURDAN Press, 1990.

[DUC92] DUCHASTEL, Philippe C. - *Integrating Hypermedia into Intelligent Tutoring*, in «Interactive Multimedia Learning Environments», Berlim, 1992, p. 66-74.

[ELL91] ELLIS, David - *Hypertext: Origins and Use.* «International Journal of Information Management», 1991, p.5-13.

[FID88] FIDERIO, Janet - *A Grand Vision.* «BYTE», Outubro 1988, p.237-244.

[FLE90] FLETTON, Nigel T. - *A hypertext approach to browsing and documenting software*, in «Hypertext: State of the Art», Londres, ABLEX - Publish Corporation, 1990, p. 193-203.

[FM91] FEINER, S.K.; McKEOWN K.R. - *Automating the Generation of Coordinated Multimedia Explanations.* «COMPUTER», Outubro 1991, p.33-41.

[FOX91] FOX, Edward A. - *Advances in Interactive Digital Multimedia Systems.* «COMPUTER», Outubro 1991, p.9-19.

[FRI88] FRISSE, Mark - *From Text to Hypertext.* «BYTE», Outubro 1988, p.247-253.

[FS90] FURUTA, Richard; STOTTS, P. David - *Separating hypertext content from structure in Trellis*, in «Hypertext: State of the Art», Londres, ABLEX - Publish Corporation, 1990, p. 204-213.

[GAR91] GARZOTTO, Franca, et al. - *HDM - A Model for the Design of Hypertext Applications*. «Hypertext'91 Proceedings», Dezembro 1991, p. 313-328.

[GIA92] GIARDINA, Max - *Interactivity and Intelligent Advisory Strategies in a Multimedia Learning Environment: Human Factors, Design Issues and Technical Considerations*, in «Interactive Multimedia Learning Environments», Berlim, 1992, p. 48-66.

[GRI89] GRICE, Roger A. - *Online Information: What Do People Want? What Do People Need?*, in «The Society of Text», Massachusetts, The MIT Press, 1989, p.22-44.

[HM89] HERRSTROM, D., MASSEY D.G. - *Hypertext in Context*, in «The Society of Text», Massachusetts, The MIT Press, 1989, p.45-58.

[HOD89] HODGES, Mathew E., et al. - *Investigations in Multimedia Design Documentation*, in «The Society of Text», Massachusetts, The MIT Press, 1989, p.79-89.

[HS90] HARDMAN, Lynda; SHARRATT, Brian S. - *User-Centered Hypertext Design: the application of HCI design principles and guidelines*, in «Hypertext: State of the Art», Londres, ABLEX - Publish Corporation, 1990, p. 252-259.

[JON88] JONASSEN, David H. - *Designing Structured Hypertext and Structuring Access to Hypertext*. «EDUCATIONAL TECHNOLOGY», Novembro 1988, p.13-16.

[JON90] JONASSEN, David H. - *Semantic Network Elicitation: Tools for Structuring Hypertext*, in «Hypertext: State of the Art», Londres, ABLEX - Publish Corporation, 1990, p. 142-152.

[KOH90] KOH, Toh-Tzu, et al. - *On the design of a frame-based hypermedia system*, in «Hypertext: State of the Art», Londres, ABLEX - Publish Corporation, 1990, p. 154-165.

[KR90] KNOPIK, Thomas; RYSER, Sigrid - *AI methods for structuring hypertext information*, in «Hypertext: State of the Art», Londres, ABLEX - Publish Corporation, 1990, p. 224-230.

[LG91] LITTLE, T.D.C.; GHAFOOR, A. - *Spatio-Temporal Composition of Distributed Multimedia Objects for Value-Added Networks.* «COMPUTER», Outubro 1991, p.42-50.

[MAR92] MARTIN, James - *Hiper Documentos e Como Criá-los.* Editora Campus.1992.

[MAR88] MARCHIONINI, Gary - *Hypermedia and Learning: Freedom and Chaos.* «EDUCATIONAL TECHNOLOGY», Novembro 1988, p.8-12.

[MAY90] MAYES, Terry, et al. - *Signposts for conceptual orientation: some requirements for learning from hypertext*, in «Hypertext: State of the Art», Londres, ABLEX - Publish Corporation, 1990, p. 121-129.

[McN90] McNIGHT, Cliff, et al. - *A comparison of linear and hypertext formats in information retrieval*, in «Hypertext: State of the Art», Londres, ABLEX - Publish Corporation, 1990, p. 9-19.

[MEG91] MEGHINI, Carlo, et al. - *Conceptual Modeling of Multimedia Documents.* «COMPUTER», Outubro 1991, p.23-29.

[MIL91] MILLER, Michael J. - *Multimedia: Seeing is Believing.* «PC MAGAZINE», Dezembro 1991, p.79-80.

[MOR91] MORARIU, Janis - *Hypermedia in Instruction and Training: The Power and the Promisse.* «EDUCATIONAL TECHNOLOGY», Agosto 1991, p.24-29.

[NAR91] NARASIMHALU, A.D.; CHRISTODOULAKIS, S. - *Multimedia Information Systems: The Unfolding of a Reality.* «COMPUTER», Outubro 1991, p.6-8.

[NEW91] NEWCOMB, Steven R., et al. - *"HyTime" - The Hypermedia/Time-based Document Structuring Language.* «Communications of the ACM», Vol.34, No.11, Novembro 1991, p. 67-83.

[NOL92] NOLTHUIS, Joseph - *Multimedia and the Art od Linking Reality*, in «Multimedia Interface Design», New York, ACM Press, 1992, p. 39-52.

[PAR91] PARK, Ok-choon - *Hypermedia: Functional Features and Research Issues.* «EDUCATIONAL TECHNOLOGY», Agosto 1991, p.24-29.

[PF91] PALANIAPPAN, M; FITZMAURICE, G. - *InternetExpress: An Inter-Desktop Multimedia Data- Transfer Service.* «COMPUTER», Outubro 1991, p.58-67.

[RAD91] RADA, R. - *Hypertext and Paper: A Special Synergy.* «International Journal of Information Management», 1991, p. 14-22.

[RAS90] RASKIN, Robin - *Multimedia: The Next Frontier for Business?.* «PC MAGAZINE», Julho 1990, p.151-192.

[RIT89] RITCHIE, I - *HYPERTEXT - Moving Towards Large Volumes.* «THE COMPUTER JOURNAL», Vol.32, No.6, 1989, p.516-523.

[SB90] STEFIK, Mark; BOBROW, Daniel G. - *Object-Oriented Programming: Themes and Variations.* «THE AI MAGAZINE»,1990, p. 40-62.

[SF89] STOTTS, David P.; FURUTA, Richard - *Petri-Net-Based Hypertext: Document Structure with Browsing Semantics.* «ACM Transactions on Information Systems», Vol.7, No.1, Janeiro 1989, p. 3-29.

[SHN89] SHNEIDERMAN, Ben - *Reflections on Authoring, Editing, and Managing Hypertext*, in «The Society of Text», Massachusetts, The MIT Press, 1989, p.115-131.

[SME91] SMEATON, A.F. - *Retrieving information from hypertext: issues and problems.* «European Journal of Information Systems», Vol.1, No.4, 1991, p.239-247.

[SOU92] SOUSA, João de - *As Novas Coordenadas.* «EXAME», Outubro 1992, p.166-170.

[STR88] STROHMER, Joanne C. - *A Model for the Design of a Videodisc.* «EDUCATIONAL TECHNOLOGY», Abril 1988, p.38-40.

[TOM89] TOMPA, Frank WM. - *A Data Model for Flexible Hypertext Database Systems.* «ACM Transactions on Information Systems», Vol.7, No.1, Janeiro 1989, p. 85-100.

[TOM92] TOMPSETT, C. P. - *Contextual Browsing Within a Hypermedia Environment*, in «Interactive Multimedia Learning Environments», Berlim, 1992, p. 95-107.

[VIG92] VIGNANCOUR, Olivier - *Comment Pour Construire une Arbre Généalogique avec ToolBook.* «SCIENCE & VIE MICRO», Maio 1992, p.146-149.

[VIN91] VIN, Harrick M., et al. - *Multimedia Conferencing in the Etherphone Environment.* «COMPUTER», Outubro 1991, p.69-78.

[WAL89] WALKER, Janet H. - *Authoring Tools for Complex Document Sets*, in «The Society of Text», Massachusetts, The MIT Press, 1989, p.132-146.

[WAT92] WATERWORTH, John A. - *Multimedia Interaction with Computers - human factor issues.* West Sussex, Ellis Horwood Limited, 1992.

[WEG87] WEGNER, Peter - *Dimensions of Object-Based Language Design.* «COPSLA'87 Proceedings», Outubro 1987, p. 168-182.

[WK87] WOELK, D.; KIM, W - *Multimedia Information Management in an Object-Oriented Database System.* «Proceedings of the 13thVLDB Conference», Brighton 1987, p.319-329.

[YAG91] YAGER, Tom - *Information's Human Dimension.* «BYTE», Dezembro 1991, p.153-160.

[YOU91] YOUNG, Andrew - *The CD-ROM Standards Frontier: Rock Ridge.* «CD-ROM PROFESSIONAL», Novembro 1991, p.53-56.

[ZEL92] ZELLWEGER, Polle T. - *Toward a Model for Active Multimedia Documents*, in «Multimedia Interface Design», New York, ACM Press, 1992, p. 39-52.