**UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE FÍSICA DE SÃO CARLOS
DEPARTAMENTO DE FÍSICA E INFORMÁTICA**

**"Análise e Avaliação de Hiperdocumentos: uma abordagem baseada na
Representação Estrutural".**

RENATA PONTIN DE MATTOS FORTES

OK

Tese apresentada ao Instituto de Física de São Carlos, Universidade de São Paulo, para obtenção do título de Doutor em Ciências "Física Aplicada-opção Física Computacional".

Orientador: Prof.Dr. Alvaro Garcia Neto

SÃO CARLOS

SÃO PAULO

1996

RECIBO DE SERVIÇO DE BIBLIOTECA E

Fortes, Renata Pontin de Mattos

Análise e Avaliação de Hiperdocumentos: uma abordagem baseada na Representação Estrutural. - São Carlos, 1996.

179 p.

Tese (Doutorado) - Instituto de Física de São Carlos, 1996.

Orientador: Prof. Dr. Alvaro Garcia Neto

1. Hiperdocumentos 2. Engenharia de Hiperdocumentos 3. Avaliação de Hiperdocumentos

**UNIVERSIDADE
DE SÃO PAULO**

Instituto de Física de São Carlos

Av. Dr. Carlos Botelho, 1465
CEP 13560-250 - São Carlos - SP
Brasil

Fone (016) 272-6222

Fax (016) 272-2218

MEMBROS DA COMISSÃO JULGADORA DA TESE DE DOUTORADO DE **RENATA
PONTIN DE MATTOS FORTES** APRESENTADA AO INSTITUTO DE FÍSICA DE SÃO
CARLOS, UNIVERSIDADE DE SÃO PAULO, EM 30/08/1996.

COMISSÃO JULGADORA:

Prof. Dr. Alvaro Garcia Neto/IFSC-USP

Prof. Dr. Osvaldo Novais de Oliveira Junior/IFSC-USP

Profa. Dra. Beatriz Mascia Daltrini/UNICAMP

Profa. Dra. Maria da Graça Pimentel/ICMSC-USP

Profa. Dra. Marina Teresa Pires Vieira/UFSCar

*A minha família: Fábio e meus filhos
Gustavo, Laura e Francisco, pelo
amor e toda a compreensão.*

Aos meus pais, pelo apoio constante.

À D. Ercília, pelo exemplo de amor à vida.

Agradecimentos

Ao Prof. Álvaro Garcia Neto, meu orientador, pelo apoio e constante otimismo para que este trabalho fosse realizado.

À Profa. Maria do Carmo Nicoletti pelas suas idéias e ajuda constante, indispensáveis à conclusão deste trabalho.

Às amigas Júnia, Léa e M. Elisa, pela confiança em mim depositada e o companheirismo nas experiências do dia-a-dia.

Aos amigos do ICMSC-USP, que de uma maneira ou outra, contribuíram com o estímulo para o prosseguimento deste trabalho.

Aos amigos do IFSC-USP, pela atenção que me foi dedicada.

Sumário

Lista de Figuras

Lista de Tabelas

Resumo

Abstract

1	Introdução	1
2	Sistemas hipertexto e hiperdocumentos	4
2.1	Contextualização e características.....	5
2.1.1	Definição de hiperdocumento.....	5
2.1.2	Objetivo dos sistemas hipertexto.....	8
2.1.3	Principais funções dos sistemas hipertexto.....	9
2.1.3.1	Autoria.....	12
2.1.3.2	Apresentação.....	13
2.1.3.3	Navegação.....	14
2.1.4	Problemas × impacto da tecnologia de hipertexto.....	15
2.2	Principais modelos de sistemas hipertexto.....	18
2.2.1	Modelo HAM.....	19
2.2.2	Modelo de referência Dexter.....	21
2.2.3	HAM × Dexter: uma análise comparativa.....	22
2.3	Representação de hiperdocumentos.....	25
2.3.1	Conceituação de estrutura dos hiperdocumentos.....	25
2.3.2	Técnicas para representação da estruturação dos nós.....	27
2.3.2.1	SGML.....	29
2.3.2.2	HTML.....	31

2.3.2.3	HyTime.....	33
2.3.2.4	MHEG.....	34
2.3.2.5	ODA.....	36
2.3.3	Técnicas para representação da estruturação dos <i>links</i>	37
2.3.3.1	Grafos direcionados	39
2.3.3.2	Hipergrafos	40
2.3.3.3	Redes de Petri.....	45
2.3.3.4	Conjuntos	49
2.4	Considerações finais.....	51
3	Evolução dos hipertextos	53
3.1	Estado da arte da análise de conteúdo e melhorias de navegação	55
3.2	Evolução do processo de autoria	57
3.3	Evolução dos modelos de hiperdocumento	61
3.3.1	HDM.....	65
3.3.2	MCA.....	67
3.3.3	AHM.....	68
3.3.4	RMM	73
3.4	Estado da arte da avaliação da estruturação de hiperdocumentos	75
3.5	Conclusões e a tendência das evoluções	88
4	Uma proposta de abordagem orientada a <i>links</i> para avaliação da estrutura de hiperdocumentos	92
4.1	Uma sistematização para exame da estruturação de hiperdocumentos	94
4.1.1	O StatSim como ferramenta de auxílio ao exame de hiperdocumentos	100
4.1.2	Proposta de uma arquitetura para exame da estrutura de hiperdocumentos	103
4.2	Uma proposta de classificação de reuso de estrutura.....	105
4.2.1	Classe de <i>links</i> de uso-geral	111
4.2.2	Classe de <i>links</i> de uso-particular.....	112
4.2.3	Classe de <i>links</i> de uso-contextual	113
4.3	Especificação formal da classificação dos <i>links</i>	117
4.3.1	Degenerações e agregações derivadas	118
4.3.2	Formalização dos casos de <i>links</i>	125
4.4	Higrafo: uma proposta para a representação da estruturação de <i>links</i>	127
4.5	Relevância da classificação dos <i>links</i>	132

5	LiOS - um sistema orientado a <i>links</i> para exame de WWW	135
5.1	Arquitetura do sistema LiOS	137
5.2	O módulo Representação do HD	139
5.3	O módulo Conversor	144
5.4	O módulo Apresentação dos <i>links</i>	145
5.5	Usando o LiOS	147
5.6	Análise empírica do uso de LiOS	150
6	Resultados	152
6.1	Definição do experimento	153
6.2	Descrição dos resultados	154
6.3	Análise	157
6.4	Experiência controlada	159
7	Conclusões	163
7.1	Atividades realizadas	164
7.2	Principais contribuições	167
7.3	Trabalhos futuros	167
8	Referências bibliográficas	170
	Apêndice A	178

Lista de Figuras

Figura 2.1 - Um exemplo de hiperdocumento	7
Figura 2.2 - Os processos típicos de um sistema hipertexto: autoria, navegação e apresentação.	11
Figura 2.3 - Processos típicos existentes nos sistemas hipertexto e as respectivas dimensões que podem ser analisadas no projeto de tais sistemas.	17
Figura 2.4 - Arquitetura dos modelos de sistema hipertexto, Nielsen (1990a). .	20
Figura 2.5 - Módulo de gerenciamento de objetos do sistema hipertexto SEPIA utilizado por Schütt; Streitz (1990)	23
Figura 2.6 - Exemplo de componentes de informação de um nó de hiperdocumento.	28
Figura 2.7 - Componentes básicos da estrutura de um <i>link</i> de hiperdocumento	37
Figura 2.8 - Diagrama de Transição de Estados (DTE).....	38
Figura 2.9 - Representação gráfica do hipergrafo H , onde $H = (\{s_1, s_2, s_3, s_4\}, \{E_1, E_2, E_3, E_4\})$	41
Figura 2.10 - Diagramas do hipergrafo da Figura 2.9: (a) seguindo o padrão de arestas e (b) seguindo o padrão de subconjuntos	42
Figura 2.11 - (a) Representação da estrutura da estrutura de um hiperdocumento por um grafo dirigido rotulado e (b) Representação por um hipergrafo dirigido rotulado, seguindo exemplo fornecido por Tompa (1989).	43
Figura 2.12 - Uma rede de Petri representando a estrutura de um hiperdocumento com múltiplos caminhos de navegação independentes.	47
Figura 2.13 - Diagramas de Venn/Euler, representando conjuntos e suas relações estruturais.	51
Figura 3.1 - Exemplo de hiperbase H com três níveis de aninhamento segundo o modelo MCA, conforme proposta de <i>browser</i> de Muchaluat; Soares (1995)	68
Figura 3.2 - Exemplos de diferentes contextos de <i>links</i> , adaptados dos exemplos utilizados por Hardman et al. (1993).	71
Figura 3.3 - (a) Visão hierárquica definida no sistema CMIFed, (b) A respectiva visão de árvore da estrutura definida em (a) , segundo Hardman et al. (1994).	72
Figura 4.1 - Geração de protótipo: abstração e instanciação, segundo Nanard; Nanard (1995).	95

Figura 4.2 - Etapas do Ciclo de Vida de Prototipação Evolutiva para o desenvolvimento de hiperdocumentos.....	97
Figura 4.3 - Arquitetura do ambiente StatSim.....	101
Figura 4.4 - A tela do módulo “Analisador LES”, ativada para simulação textual de uma especificação em LES.	102
Figura 4.5 - As abordagens de análise para exame da estrutura de um hiperdocumento.	103
Figura 4.6 - DTE representando um <i>link</i> bidirecional de hiperdocumento	106
Figura 4.7 - Árvore de geração dos casos de reuso de componentes de <i>links</i> ..	108
Figura 4.8 - Esquemas de casos de reuso de componentes de <i>link</i>	109
Figura 4.9 - Classes de casos de <i>links</i> considerando-se o reuso de seus componentes.....	110
Figura 4.10 - Casos derivados do caso 0.	120
Figura 4.11 - Casos derivados do caso 1.	121
Figura 4.12 - Casos derivados do caso 2.	123
Figura 4.13 - Casos derivados do caso 3.	124
Figura 4.14 - Representação gráfica de um higráfico.....	130
Figura 4.15 - A árvore que representa a estrutura hierárquica das “bolhas” que compõem o higráfico da Figura 4.14.	131
Figura 5.1 - O objeto <i>LINK</i> e seus componentes.....	137
Figura 5.2 - Arquitetura modular do LiOS.....	137
Figura 5.3 - DTE representando um <i>link</i> unidirecional de hiperdocumento. ..	139
Figura 5.4 - A página de entrada do <i>site</i> do servidor WWW do ICMSC-USP codificada em HTML.	142
Figura 5.5 - Declaração em LES dos <i>links</i> contidos na página da Figura 5.4, resultante do módulo “Representação do HD”..	143
Figura 5.6 - As janelas que apresentam os <i>links</i> de um hiperdocumento da WWW, mostrando os nós-fonte e destino que são conectados por eles.	146
Figura 5.7 - Formas alternativas de apresentação da estrutura do hiperdocumento, com base nos seus <i>links</i>	148
Figura 6.1 - Gráfico de barras representando a evolução dos casos em três momentos distintos.	158
Figura 6.2 - Gráfico apresentando as proporções de incidências de cada um dos casos e a evoluções nos três momentos distintos.....	159

Lista de Tabelas

Tabela 3.1 - Abstrações de nós definidas nos modelos de hiperdocumento estudados.....	89
Tabela 3.2 - Resumo dos critérios e métricas que podem ser analisados em hiperdocumentos, considerando-se sua dimensão estrutural.	91
Tabela 4.1 - Componentes estruturais genéricos de um hiperdocumento e seus correspondentes componentes de DTE	106
Tabela 4.2 - Casos primitivos e as respectivas derivações por agregação ou degeneração	118
Tabela 5.1 - Componentes de estruturação de hiperdocumento como componentes de DTE.....	140
Tabela 6.1 - Consultas para obtenção das incidências de <i>links</i> em cada um dos casos de reuso de componentes de <i>links</i>	154
Tabela 6.2 - Contabilizações resultantes de consultas a incidências de <i>links</i> de cada um dos casos de reuso de componentes de <i>links</i>	155

RESUMO

A construção de um hiperdocumento envolve, geralmente, a composição de um volume significativo de informações e, portanto, é considerada uma atividade com grande risco de erros. Uma análise do hiperdocumento que auxilie a atividade de autoria e conduza à melhoria de sua qualidade exige do autor uma visão completa do hiperdocumento a partir de várias perspectivas; muitas vezes, tal visão é impossível de ser obtida sem o auxílio de algum procedimento sistemático ou mesmo automatizado. Nesse contexto, este trabalho investiga formas automáticas de avaliação de hiperdocumentos com base na sua estruturação, com o objetivo de melhorar a qualidade da organização das ligações (*links*) construídas pelo(s) autor(es). Uma sistemática para exame de hiperdocumentos, baseada na análise relacional de reuso de *links*, proposta neste trabalho, subsidiou o desenvolvimento do sistema LiOS (*Link Oriented System*), uma ferramenta para auxiliar no processo de avaliação da estrutura de hiperdocumentos. A partir dessa formalização, foi, então, proposta uma classificação de reuso de *links*, que se constitui numa nova métrica para avaliação de hiperdocumentos. O uso do sistema LiOS para o exame de hiperdocumento da *World-Wide Web* comprovou que, por meio da identificação dos casos de reuso de *links*, a qualidade da estrutura do hiperdocumento pode ser melhorada, revelando a eficiência da sistemática proposta neste trabalho.

ABSTRACT

The construction of an hyperdocument usually involves the handling of a great volume of information. It is, therefore, considered an error prone activity. The hyperdocument's analysis procedure, which helps both, the authoring activity and the improvement of the quality of a document, requires from the author a complete view of the hyperdocument, approached from different perspectives. Sometimes this is not feasible unless supported by some kind of systematic or automated procedure. This work is concerned with automated ways of evaluating an hyperdocument, using its structure, aiming to improve the quality of the link organisation defined by the author. The systematic approach for examining the hyperdocument, based on the relational analysis of link reuses, proposed in this work, subsidised the development of the LiOS (Link Oriented System) system, a tool for helping the authoring process, through the evaluation of the structure of an hyperdocument. By using the LiOS system for examining an hyperdocument from the World-Wide Web it was evidenced that, through the identification of the link reuse cases, the quality of the hyperdocument could be improved; this fact stresses the efficiency of the proposed approach.

Capítulo 1

Introdução

Os sistemas hipertexto são um meio de manipulação de informações que permite alta produção de documentos e de ligações entre eles. Ao autor do hiperdocumento é permitida a criação de fragmentos de informação (**nós**), como páginas de documentos, por exemplo, e de referências que os relacionam através de ligações (**links**), com total liberdade. Isso viabiliza a reutilização de nós e ligações com maior frequência e, conseqüentemente, um maior volume de informações se torna disponível. Portanto, os sistemas hipertexto, além de facilitarem a interação do usuário/leitor dos documentos na obtenção das informações que estão na forma de hiperdocumento, oferecem facilidades para o aumento da produção de nós e *links* durante o processo de autoria do hiperdocumento, em função da liberdade e reuso que estão disponíveis.

De acordo com Brown (1990), o principal fator de sucesso de um hipertexto está na qualidade da autoria. Esse padrão de qualidade tem se mostrado baixo devido principalmente à inexperiência dos autores diante da nova tecnologia, e ao estado primitivo das ferramentas de apoio à autoria de hipertextos existentes.

As atividades de manutenção quase sempre são complexas e custosas, pois são realizadas sobre um grande volume de nós e *links*, e devem preservar a estruturação “correta” do hiperdocumento segundo certas regras. Brown salienta o importante papel de ferramentas de *software* de suporte à autoria, fornecendo, por exemplo, visões da estrutura geral do hiperdocumento e mecanismos para avaliação do hiperdocumento, como forma de auxiliar também a fase de manutenção. Destaca, ainda, que uma das principais razões para que os autores se preocupem com a qualidade de sua tarefa é a vida útil do hiperdocumento, potencialmente longa, com um grande número de manutenções e crescimento do hiperdocumento ao longo do tempo. Uma manifestação atual da alta produção de nós e *links* de hipertexto que está disponível na rede Internet são os servidores da *World-Wide Web* (WWW) espalhados pelo mundo inteiro (Berners-Lee et al., 1994).

Para melhorar a qualidade dos hiperdocumentos gerados pelo processo de autoria, muitos trabalhos têm investigado as propriedades dos hiperdocumentos, por exemplo, as métricas identificadas por Botafogo; Shneiderman (1992), o modelo de qualidade de hiperdocumento de Hatzimanikatis et al. (1995), as dimensões mensuráveis de um projeto hipermídia identificadas por Garzotto et al. (1995) e as propriedades de navegação de um hiperdocumento como um autômato-de-*links* de Stotts et al. (1992). Esses trabalhos se fundamentam sobre diferentes formas de avaliação, mas todos visam à melhoria da qualidade e facilidade de manutenção de hiperdocumentos.

Outro aspecto importante a ser considerado nos hiperdocumentos é a atual tendência de incorporação da tecnologia multimídia aos sistemas hipertexto, que passam a ser denominados sistemas hipermídia. Com isso, surgem novas exigências de qualidade dos hiperdocumentos compostos por nós de informação de diferentes mídias. A dimensão temporal passa a ser considerada nos nós cuja exibição do conteúdo esteja vinculada a um período de tempo. Nota-se que novas possibilidades ainda podem surgir para explorar a nova potencialidade de manipulação de múltiplas mídias.

A tecnologia de hipertexto¹ tem atingido as mais diversas áreas de aplicação. Um aspecto fundamental dessa tecnologia é que o hiperdocumento deve, sempre que possível, exprimir coerência para garantir a credibilidade nas informações que coloca à disposição dos leitores. Diante disso, o hiperdocumento deve estar sempre sendo reavaliado.

O propósito deste trabalho é investigar tecnologias de avaliação de hiperdocumentos, baseadas na análise de sua estrutura. O Capítulo 2 apresenta os fundamentos que norteiam os sistemas hipertexto, as características de sua funcionalidade e os modelos mais difundidos. As técnicas para representação da estrutura dos hiperdocumentos revisadas da literatura são apresentadas e discutidas. O Capítulo 3 apresenta a evolução das tendências de modelagem de hiperdocumentos, com base na revisão bibliográfica elaborada. Esse capítulo também descreve o estado da arte sobre o tema de avaliação de hiperdocumentos, juntamente com os trabalhos relevantes relacionados ao tema e o contexto em que se insere o projeto desenvolvido nesta tese.

O Capítulo 4 apresenta a abordagem proposta para avaliação de estrutura de hiperdocumentos. Essa abordagem se fundamenta no Exame de hiperdocumentos segundo uma Análise Relacional para implementação do critério de reuso de componentes de *links*. O Capítulo 5 apresenta um sistema protótipo construído seguindo a abordagem proposta no Capítulo 4. O sistema foi denominado “*Link-Oriented System*” (LiOS), e foi desenvolvido com o objetivo de auxiliar o exame de hiperdocumentos da WWW com relação aos critérios de reuso e consistência. O Capítulo 6 apresenta os resultados quantitativos e as indicações das avaliações qualitativas que resultaram da análise de experimentos feitos em servidores da WWW. Finalmente, o Capítulo 7 apresenta as limitações da abordagem proposta para avaliação de hiperdocumentos, as perspectivas dos trabalhos futuros e as conclusões.

¹ Será utilizado o termo **hipertexto** sempre que for possível, denotando a tecnologia sobre a qual os estudos realizados nesta tese se orientaram. Quando for necessária a diferenciação para o tratamento de múltiplas mídias, o termo **hipermídia** será empregado.

Capítulo 2

Sistemas hipertexto e hiperdocumentos

A tecnologia de sistemas hipertexto oferece um grau de facilidade significativo para o usuário de sistemas que visam a busca e o fornecimento de informações. As premissas de interface simples e quase intuitiva, bem como a liberdade de escolha na busca de informações são cada vez mais requisitos necessários dos sistemas aplicativos.

Um grande número de aplicativos utilizando a tecnologia de hipertextos têm surgido. As tentativas de adequação dos sistemas hipertexto aos vários tipos de requisitos desses aplicativos motivaram o desenvolvimento dos diversos modelos de tais sistemas.

Este capítulo apresenta os fundamentos que norteiam os sistemas hipertexto, as características de sua funcionalidade e os modelos mais difundidos. As técnicas para representação de hiperdocumentos vistas na literatura são também apresentadas e discutidas, com o objetivo de fornecer o embasamento para a proposta de avaliação da estruturação de hiperdocumentos, objetivo principal deste trabalho.

Neste capítulo, a Seção 2.1 contextualiza a área de pesquisa, caracterizando o tema de sistemas hipertexto e hiperdocumentos. São apresentados os problemas inerentes da nova tecnologia de hipertextos e confrontados ao impacto ocorrido. A Seção 2.2 apresenta os principais modelos de sistemas hipertexto que surgiram para dar o apoio ao seu avanço tecnológico. A Seção 2.3 discute as técnicas para representação de hiperdocumentos e a Seção 2.4 finaliza apresentando as considerações relevantes deste capítulo.

2.1 Contextualização e características

Nielsen (1995) inicia seu livro *"Multimedia and Hypertext"* definindo, na forma mais simples possível, o termo **hipertexto**: "Hipertexto é a tecnologia que habilita intrinsecamente a leitura não-seqüencial, em contraste com a forma de leitura seqüencial disponível nos livros-texto." Então, um hipertexto consiste de pedaços de informação (denominados nós) interconectados (pelos chamados *links*).

Apesar de não existir uma definição formal e exata de hipertexto que possa ser convencionalmente utilizada, um consenso de suas características principais tem sido adotado na maioria dos trabalhos da literatura especializada. Também com o objetivo de uniformizar a terminologia, são apresentados a seguir os conceitos sobre sistemas hipertexto e hiperdocumentos adotados nesta tese.

2.1.1 Definição de hiperdocumento

A base da tecnologia de sistemas hipertexto é uma rede de informações que possui interconexões que devem estar facilmente acessíveis pelos usuários¹. Essa rede de

¹ O usuário de sistemas hipertexto pode ser: leitor e autor. Neste trabalho, "leitor" é a referência ao usuário leitor, "autor" é a referência ao usuário autor e, "usuário" é a referência ao usuário que desempenha ambos os papéis, de autor e leitor.

informações, onde os nós correspondem às unidades de informação e os *links* às interconexões entre eles, compõe um hiperdocumento.

Um hiperdocumento poderia ser pensado numa primeira instância como uma Base de Dados para um sistema hipertexto, por se tratar de um repositório de informações que podem ser recuperadas nos nós. No entanto, apesar do hiperdocumento possuir semelhança com Base de Dados ao se acessar² seus nós, sua base de informações não exige uma definição centralizada de sua estrutura (Begoray, 1990). Além disso, não possui uma estrutura regular, que geralmente caracteriza uma Base de Dados convencional. Pode ser o caso, por exemplo, do hiperdocumento ter milhares de nós, vários deles compostos de muitas informações e outros eventualmente com poucas.

Geralmente, a estrutura de um hiperdocumento é definida localmente como a união das decisões feitas durante a construção de cada nó e cada *link* individualmente. Muito embora existam trabalhos relacionados à definição automática de *links* e pesquisas relacionadas à modelagem conceitual das informações alocadas em um hiperdocumento, observa-se que a abordagem de Base de Dados não pode ser diretamente adotada para conceituar a base de documentos que compõe um hiperdocumento (Nielsen, 1995).

Com relação ao termo hiperdocumento, Thuring et al. (1995) apresentam uma distinção importante para a utilização do termo, quando discutem o relacionamento entre cognição e sistemas hipermídia. Eles distinguem dois tipos de aplicação:

1. As que encorajam os usuários que desejam “vagar”³ por grandes “nuvens” de informações, obtendo conhecimento ao longo do caminho. Essas são aplicações que possuem uma Base de Dados “navegável” que pode ser livremente explorada pelo leitor. Esta Base de Dados flexível é denominada **hiperbase**.

² Embora o termo “acessar” não seja uma palavra encontrada nos dicionários da Língua Portuguesa, é amplamente utilizado pela comunidade da área de Ciência da Computação.

³ Muitos termos, atualmente, têm sido utilizados para expressar a nova abordagem de leitura proporcionada pelos hipertextos, como por exemplo, “navegar”, “surfear”, “vagar”.

2. As que são direcionadas à resolução de problemas específicos e que, portanto, possuem informações mais estruturadas. Segundo Stotts; Furuta (1991), talvez essas aplicações tenham mais limitações. Essas são as aplicações que possuem a Base de Dados na forma de documentos eletrônicos que intencionalmente guiam os usuários através do espaço de informações, controlando sua exploração ao longo de uma estrutura pré-definida. Esta Base de Dados é denominada **hiperdocumento**.

Cada uma dessas aplicações, com suas respectivas Bases de Dados, possui vantagens diversas, encorajando diferentes estratégias de leitura que afetam o processo de navegação. Enquanto a primeira é mais adequada quando se requer pesquisa irrestrita e recuperação de informações, a segunda se destina mais a tarefas que exijam o aprendizado e compreensão mais profundos do conteúdo.

Figura 2.1 - Um exemplo de hiperdocumento

No presente contexto de apresentação das principais características dos sistemas hipertexto, será empregado o termo hiperdocumento como referência à Base de Dados dos sistemas hipertexto. Dessa forma, hiperdocumento é o termo utilizado

para o produto criado e manipulado por um sistema hipertexto. Deve-se notar que o mesmo pode ser encontrado com denominações diversas, simplesmente como hipertexto, hiperbase, documentos hipertexto, *web*, hipermídia e hiperespaço de nós e *links*.

A Figura 2.1 mostra um exemplo da rede de nós e *links* que compõe um hiperdocumento, onde os nós estão contidos em retângulos de cantos arredondados, os *links* são representados por setas de linhas tracejadas. São apresentadas as **âncoras**, também denominadas de **botões de controle**, cuja função é a de especificar a localização de um *link* dentro de um nó. Elas correspondem aos *ícones* de *links* mencionados por Conklin (1987a). As âncoras, na Figura 2.1, estão simbolizadas por trechos de texto sublinhados ou símbolos, contidos nos nós.

As âncoras podem ser definidas informalmente como sendo as regiões contidas nos nós que indicam a presença de *links*. O Capítulo 4 discute com maior detalhe sua função, conforme sua posição, no ponto terminal ou no ponto de origem do *link*.

2.1.2 Objetivo dos sistemas hipertexto

Segundo Conklin (1987b), do ponto de vista de Ciência de Computação a essência dos sistemas hipertexto está precisamente na combinação das disciplinas tradicionais. Assim, os sistemas hipertexto devem ser tratados como:

- **Métodos de acesso a Base de Dados**, pois fornecem uma nova forma de acesso direto aos dados diferente do uso convencional de consultas (*queries*);
- **Esquemas de representação**, pois são um tipo de “rede semântica”, que mistura material informal (como textos, por exemplo) com operações e processos formais e automatizados;

- **Modalidades de interface**, pois são caracterizados pela disponibilidade de interação sobre âncoras, que podem estar arbitrariamente embutidas no conteúdo dos nós.

É consenso, entretanto, que um sistema hipertexto se destina a fornecer processos automatizados para as atividades de manipulação e obtenção das informações contidas nos hiperdocumentos. Para que o hiperespaço de documentos se torne acessível, uma forma de acesso não-sequencial é necessária: a navegação. Hiperespaço é o termo adotado para se referir ao espaço de dimensões não passíveis de representação bidimensional, utilizado para o relacionamento semântico entre mais do que dois nós em um hiperdocumento.

O objetivo principal de um sistema hipertexto é, portanto, fornecer processos automatizados para a manipulação de um hiperdocumento. Os processos típicos que devem ser supridos são: a criação de um hiperdocumento (**autoria**) para, em um momento posterior, colocá-lo à disposição para **navegação e apresentação**. A seguir são detalhados esses processos.

2.1.3 Principais funções dos sistemas hipertexto

Ainda segundo Conklin, os sistemas hipertexto podem ser informalmente descritos pelas seguintes características gerais:

1. A Base de Dados deve ser uma rede de nós de informação;
2. Os conteúdos das janelas na tela devem corresponder aos nós de informação na Base de Dados e somente poucos nós devem ser apresentados (“abertos”) simultaneamente;
3. As operações padrão sobre o sistema de janelas devem ser fornecidas (por exemplo, reposicionar, mudar o tamanho e fechar);
4. As janelas na tela podem conter qualquer quantidade de imagens (ícones) de *links*, os quais representam indicadores para outros nós na Base de Dados. Em geral, o

ícone de *link* contém um campo textual pequeno que sugere o conteúdo do nó para o qual ele aponta. A seleção de um ícone de *link* (“clcando” o *mouse* sobre o ícone) faz o sistema buscar o nó referenciado e imediatamente abrir uma nova janela para exibi-lo na tela;

5. Pode-se criar novos *links* para novos nós (anotação, comentário, etc.) ou para nós já existentes (para estabelecer novas conexões);
6. A Base de Dados pode ser navegada de três formas:
 - i. Seguindo os *links*, abrindo janelas e sucessivamente examinando seu conteúdo.
 - ii. Pesquisando na rede (ou em parte dela) por alguma *string*, ou valor de atributos - *search*.
 - iii. Usando um *browser* gráfico que mostre a rede de nós interligados (por exemplo, os nós como ícones e os *links* como linhas entre eles).

Deve-se ressaltar que a identificação dessas características gerais, sob o ponto de vista histórico, foram fundamentais para o desenvolvimento tecnológico da época. Embora muitas implementações de sistemas hipertexto não as atendam integralmente (não suportem *browser* gráfico, por exemplo), a maioria dos sistemas evoluiu procurando, de uma forma ou de outra, contemplá-las. Observa-se também que as características relacionadas informalmente enfatizam a funcionalidade da interface do sistema hipertexto com o usuário (mencionando, por exemplo, os requisitos de apresentação em janelas e de interação sobre os ícones de *links*) e com o hiperdocumento (por exemplo, referindo-se à Base de Dados como rede de nós e *links*, relacionando os nós aos conteúdos das janelas e descrevendo as formas de navegação, ou seja, de acesso aos nós).

Notadamente, a tecnologia de hipertexto abriu novas fronteiras para o processamento de Sistemas de Informação. Os sistemas hipertexto são uma nova forma de pensar, organizar e comunicar, habilitada tecnologicamente. O processo de pensar não constrói novas idéias, uma de cada vez, mas sim segue várias frentes,

desenvolvendo as idéias em diferentes níveis e sobre diferentes pontos em paralelo, cada idéia dependendo e contribuindo para as demais. O registro e comunicação de tal associação de linhas de pensamento é um desafio, pois na prática a comunicação é um processo sequencial e está sempre limitada à “largura de banda” do processamento lingüístico humano (Conklin, 1987a).

É fato, entretanto, que a tecnologia disponível é limitada pela quantidade de informação que deve estar contida nos nós e pelas decisões sobre quais ligações devem ser definidas e de que tipo devem ser.

A Figura 2.2 esquematiza os principais processos de interação, por meio de um sistema hipertexto, de um usuário com um hiperdocumento.

Figura 2.2 - Os processos típicos de um sistema hipertexto: autoria, navegação e apresentação.

O processo de **autoria** se destina à criação do hiperdocumento. O processo de **navegação** é responsável pela interação com o usuário, seja ele um leitor ou um autor. Ao navegar pelo hiperdocumento, o usuário tem acesso às opções de caminhos a seguir para continuar a **apresentação**. Apresentação é o processo que viabiliza uma forma de mostrar ao usuário, geralmente através de janelas na tela ou outro dispositivo físico quando se tratar de multimídia, o conteúdo das informações contidas

nos nós. A liberdade de escolha, que caracteriza o processo da navegação, é um diferencial importante com relação aos demais tipos de sistemas de acesso à informação.

Em linhas gerais, as principais características funcionais dos sistemas hipertexto para os processos típicos de autoria, apresentação e navegação satisfazem determinados requisitos, apresentados a seguir.

Autoria

2.1.3.1 Autoria

A autoria de um hiperdocumento, vista como um dos processos de sistema hipertexto, compreende a definição de um esquema de representação da rede de nós e *links* (i.e. da Base de Dados característica de tais sistemas) e as funções de apoio à manipulação dessa rede.

Nesse processo devem ser fornecidos procedimentos para a edição dos conteúdos dos nós e para a definição dos *links* (por exemplo, através de marcações - *markup* - no conteúdo dos nós).

Como a autoria de um hiperdocumento é um processo complexo, muitos sistemas hipertexto têm subsistemas específicos para a manipulação dos *links* (criação, alteração e remoção) e composição dos nós. Outros sistemas incorporam softwares de mercado para a edição das páginas e desenvolvem subsistemas dedicados apenas à manipulação dos *links* entre as páginas (Nielsen, 1990a).

A tarefa de criação de um hiperdocumento possui características que exigem o uso de ferramentas automáticas específicas, para a representação dos dados relativos aos nós e *links* que o compõem. Por essa razão, a autoria pode ser vista como um processo de implementação de um esquema de representação que se aplica à rede de informações contida no hiperdocumento.

Garzotto et al. (1991a) identificam dois aspectos importantes na atividade de autoria de hiperdocumentos:

- i. O mais direto, de granularidade fina ou **autoria em ponto pequeno**, que se destina à composição dos conteúdos textuais dos nós e à construção dos respectivos *links* a outros documentos;
- ii. O que lida com as informações imprecisas, de granularidade grossa ou **autoria em ponto grande**, que caracteriza o tipo de hiperdocumento de acordo com diferentes domínios de aplicação e diferentes leitores.

O segundo aspecto é considerado de extrema importância para o sucesso da utilização da técnica de navegação, inerente aos sistemas hipertexto, pois pode levar a uma complexidade indesejada na interação e conseqüente desorientação do usuário.

Fazendo uma analogia com as atividades pertencentes à etapa de definição de requisitos de sistemas em Engenharia de Software, observa-se que a atividade de autoria deve considerar o hiperdocumento como um sistema a ser construído, onde o leitor desempenha o papel de usuário final do sistema. O autor, responsável pela organização e estruturação do hiperdocumento, deve realizar uma tarefa de análise de requisitos, para atender aos objetivos do hiperdocumento; esse é o argumento de Garzotto et al. (1991a) quando identificam o aspecto de autoria em ponto grande. Como acontece com a fase de levantamento de requisitos de sistemas em Engenharia de Software, a experiência do autor se reflete no hiperdocumento que ele cria.

2.1.3.2 Apresentação

A apresentação de um hiperdocumento se refere a uma nova modalidade de interface com o usuário, cuja principal característica é a simplicidade: os conteúdos dos nós e o resultado do acionamento do *link* pelo usuário devem ser claramente exibidos. As interfaces mais difundidas, segundo Zheng; Pong (1992), são baseadas em:

- *Frames* - onde o resultado de um *link*, ou seja, o conteúdo do nó-destino, é sobreposto ao conteúdo do nó-origem,

- *Scrolling* - o conteúdo do nó-destino é inserido no ponto onde o *link* foi acionado no nó-origem,
- Múltiplas janelas - o conteúdo do nó-destino é colocado em uma nova janela na tela,
- Híbridas - o conteúdo do nó-destino pode ser incluído (*scrolling*) ou pode ser colocado em uma nova janela, dependendo do tipo de *link* que o autor define.

Uma característica importante é a forma de exibição dos *links* existentes nos nós. Em qualquer hipertexto a presença de um *link* precisa ser comunicada ao leitor. Alguns sistemas hipertexto possuem ícones típicos que indicam os *links* como, por exemplo, o Intermedia (Yankelovich, 1988) e o KMS (*Knowledge Management System*) (Akscyn et al., 1988). Outros usam palavras destacadas como é o caso do Hyperties (Shneiderman, 1989). Outros, ainda, como é o caso do Guide (Brown, 1987) e HyperCard (Apple Computer, 1989), não têm um padrão único para indicação da presença de *link*.

No Capítulo 3 são vistas as tendências na adoção de padrões para codificação de documentos, como forma de dar suporte à portabilidade dos documentos para diferentes “apresentadores” (*browsers*) dos sistemas hipertexto, independentemente do hardware ou do software utilizado na criação do hiperdocumento.

2.1.3.3 Navegação

A navegação por um hiperdocumento, atividade tipicamente difundida pela tecnologia de hipertextos, possui a funcionalidade de um método de acesso à Base de Dados e caracteriza a nova modalidade de interface. Os tipos de navegação que o usuário encontra nos sistemas hipertexto atuais permitem pelo uma das seguintes interações:

- Seguir um dos *links* indicados no nó correntemente apresentado e sucessivamente prosseguir a partir dos nós obtidos;

- Pesquisar na rede (ou em parte dela) por alguma *string*, ou valores de atributos;
- Utilizar um *browser* gráfico que mostre a rede.

A navegação através dos *links* possibilita ao leitor “saltar” para uma informação considerada relevante pelo autor. Os *links* são as opções de interação que o leitor pode escolher durante a apresentação do conteúdo do hiperdocumento; o mecanismo de navegação responde às interações que o leitor faz sobre os *links*, recuperando os nós que foram estabelecidos na ligação (Nielsen, 1990b). Os recursos adicionais que têm evoluído a navegação compreendem opções de “voltar para a página anterior” (*backtrack*), registro de história e marcadores (*bookmarks*) entre outros.

A particularidade do processo de interação com o usuário do sistema hipertexto enquadra-se como função fundamental na nova modalidade de interface (Rada, 1995). Não existe um posicionamento claro quanto à sua funcionalidade — de navegação ou de apresentação — porém, é de fato uma característica importante, uma vez que deve permitir a interação eficiente com o leitor por meio de acionamento direto sobre os *links*. Através do cursor do *mouse* sobre as âncoras ou botões de controle, ou através de telas sensíveis ao toque, tem-se mecanismos de interação direta que ficam disponíveis ao usuário e que habilitam tecnologicamente os sistemas hipertexto em sua tarefa de “rapidamente recuperar informação relacionada, proposta inicial de Vannevar Bush em sua idéia pioneira aplicada ao Memex” (Nielsen, 1995).

2.1.4 Problemas × impacto da tecnologia de hipertexto

A identificação dos processos típicos dos sistemas hipertexto (i.e., autoria, navegação e apresentação), bem como da base hiperdocumento aparecem sob diferentes enfoques na literatura atual.

De fato, as características citadas representam as linhas gerais sobre as quais muitos estudos têm sido realizados para diminuir dois problemas típicos de hipertextos, identificados por Conklin (1987b) e Halasz (1988) a saber:

1. **Desorientação do usuário** (*lost in space*) — tendência do usuário perder o sentido de localização e direção no documento não-linear;
2. **Sobrecarga cognitiva** (*cognitive overhead*) — uma sobrecarga no esforço e concentração exigidos do leitor devido à necessidade de se manter várias tarefas ou trilhas ao mesmo tempo.

Esses problemas são os principais motivadores que impulsionam grande parte das pesquisas relacionadas com a melhoria de qualidade dos sistemas hipertexto. Observa-se que o impacto tecnológico de hipertexto deu ênfase ao uso da nova abordagem habilitada tecnologicamente e muitas implementações foram desenvolvidas, mesmo com a inexistência de um formalismo básico. Delisle; Schwartz (1986) e Garg (1988), já na década de 80, defendiam o desenvolvimento desse formalismo básico para as diversas implementações.

Os problemas típicos da tecnologia de hipertexto não impediram a rápida disseminação da tecnologia de hipertextos nos mais diversos tipos de aplicação, uma vez que a inovação de flexibilidade e facilidade de uso de tais sistemas atraíam, cada vez mais, um maior número de usuários.

Garzotto et al. (1995), interessados na avaliação de projetos de aplicações hipermídia, relacionaram as seguintes dimensões a serem analisadas no projeto de um hiperdocumento: conteúdo, estrutura, apresentação, dinâmica e interação.

O **conteúdo** abrange todas as unidades de informação da aplicação. A **estrutura** é responsável por sua organização. Portanto, essas duas dimensões são processadas durante a autoria e manipulam diretamente o hiperdocumento. A **apresentação** se refere ao modo como o conteúdo e as funções da aplicação são mostradas aos leitores, que inclui a visualização das informações individuais e as características dinâmicas, como a navegação. A **dinâmica** se refere à interação do

usuário com as unidades de informação e à movimentação de uma unidade à outra. A **interação** se refere à utilização das funcionalidades dinâmicas de uma aplicação pela ativação dos elementos da apresentação. Há autores que consideram a interação como uma parte da dinâmica ou da apresentação, pois ela combina esses dois fatores. No Capítulo 3 essas dimensões serão descritas com mais detalhe.

Para ilustrar como os processos típicos de sistemas hipertexto se relacionam com as dimensões de projeto de uma aplicação hipermídia, a Figura 2.3 esquematiza a correspondência identificada.

Figura 2.3 - Processos típicos existentes nos sistemas hipertexto e as respectivas dimensões que podem ser analisadas no projeto de tais sistemas.

As características dos processos típicos de sistemas hipertexto (autoria, apresentação e navegação) não são totalmente independentes umas das outras. As dimensões de dinâmica, interação e apresentação são processadas durante a navegação e correspondem à interface do usuário com o sistema hipertexto com o objetivo de ter acesso ao hiperdocumento.

Com relação à dimensão de apresentação, naturalmente tem-se que considerar principalmente sua execução durante o processo de apresentação. No entanto, com a inclusão de múltiplas mídias nos hiperdocumentos, a dimensão de apresentação tem que ser tratada também pelos processos de autoria e navegação. Por exemplo, para definir as características de apresentação de uma mídia ativa e mesmo para controlar a sua execução, os dois processos envolvidos são de autoria e navegação, respectivamente. Essa correspondência da dimensão de apresentação com os

processos de navegação, apresentação e autoria na Figura 2.3 é representada pelas três setas originando na dimensão de apresentação.

Assim, quando se decide por desenvolver mais recursos funcionais para os sistemas hipertexto, essa decisão pode implicar em tornar disponível um número maior de alternativas para mais de um processo. Por exemplo, no caso do sistema hipertexto ter a funcionalidade de pesquisar na rede (*search*) durante a execução da navegação, o processo de autoria de composição de nós ganha um recurso para criação de nós compostos resultantes da pesquisa.

Diante do exposto, uma gama enorme de alternativas começou a ser explorada por parte dos desenvolvedores de sistemas e aplicações hipermídia. Vários pesquisadores sentiram a necessidade de um formalismo que subsidiasse a área de hipertexto, cuja concentração de trabalhos se polarizava em implementações. Surgiram então os modelos de sistemas hipertexto, para que fosse possível a adequação aos diversos tipos de requisitos de aplicações. Na próxima seção são apresentados os modelos mais difundidos.

2.2 Principais modelos de sistemas hipertexto

Paralelamente ao rápido avanço tecnológico que ocorreu nas implementações de sistemas hipertexto atendendo aos mais diversos tipos de aplicação, se pronunciou a preocupação em se estabelecer um modelo para os mesmos. Delisle; Schwartz (1986) foram os precursores dessa reivindicação; forneceram a primeira especificação da “máquina abstrata de hipertexto”, a qual seria posteriormente identificada como componente essencial da arquitetura de sistemas hipertexto.

As propostas de modelagem dos sistemas hipertexto visam apresentar uma arquitetura genérica que possa ser utilizada para diferentes implementações desses sistemas.

Antes de prosseguir na apresentação dos modelos, é importante ressaltar a distinção entre um modelo de “sistema” hipertexto e um modelo de “aplicação” hipertexto (Isakowitz et al., 1995). Embora ambos tenham o objetivo de fornecer uma arquitetura interna que defina os modelos de dados a serem tratados, o modelo de sistema hipertexto pode ser visto como um ambiente que permite a criação de aplicações hipertexto, que, por sua vez, podem também ter os seus dados modelados. Já a modelagem de aplicações hipertexto deve basicamente fornecer definições lógicas dos objetos que as aplicações utilizarão com o suporte da tecnologia de hipertexto.

Uma aplicação hipertexto pressupõe a existência de um sistema hipertexto que forneça os recursos dos processos para autoria, apresentação e navegação. Como qualquer outro tipo de aplicação, a aplicação hipertexto requer que uma análise sobre o domínio de informações seja realizada, informações essas que, no caso, ficarão alocadas no hiperdocumento.

Diante do exposto, a modelagem da aplicação corresponde em maior parte à modelagem do hiperdocumento e, portanto, no decorrer desta tese é utilizado o termo modelagem de hiperdocumento para se referir à modelagem de aplicação hipertexto, com o intuito de eliminar a ambigüidade que possa surgir com a denominação de modelo de sistema hipertexto.

Os principais modelos para sistemas hipertexto que têm se destacado pelo maior número de sistemas que os utilizam, bem como suas características mais marcantes, serão apresentados a seguir.

2.2.1 Modelo HAM

O modelo HAM (*Hypertext Abstract Machine*) de Campbell; Goodman (1988) é baseado na máquina abstrata que Delisle; Schwartz (1986) desenvolveram para o sistema hipertexto Neptune. Trata-se de um modelo originalmente concebido para ser utilizado em diferentes sistemas hipertexto, pois fornece um servidor baseado em transações como sistema de armazenamento do hiperdocumento. Seus criadores

garantiam que a chave para o sucesso futuro de um sistema hipertexto estaria na utilização de um método de armazenamento de dados eficiente e independente da aplicação. Por isso, o modelo HAM fornece o suporte que possibilita que várias aplicações estabeleçam conexões e manipulem as estruturas e os dados de um hiperdocumento.

A Figura 2.4(a) esquematiza a arquitetura em níveis proposta pelo modelo. No nível mais baixo está um servidor baseado em transações, sobre o qual é alocada a **máquina abstrata de hipertexto (HAM)**; no nível mais alto estão as aplicações e a interface com o usuário. A máquina HAM fornece acesso distribuído sobre uma rede de computadores e é sincronizada para acesso multiusuário com *recovery* e *rollback* de transações interrompidas ou incompletas (onde uma transação é um grupo de uma ou mais operações HAM que são tratadas como uma unidade única).

Figura 2.4 - Arquitetura dos modelos de sistema hipertexto, Nielsen (1990a).

O modelo HAM é considerado um modelo de sistema hipertexto genérico porque, além de estabelecer a arquitetura em níveis dos diferentes serviços identificados nos sistemas hipertexto, define também as operações para criação, modificação e acesso aos componentes do hiperdocumento.

2.2.2 Modelo de referência Dexter

A elaboração do modelo de referência Dexter de Halasz; Schwartz (1990) partiu de uma iniciativa da comunidade de projetistas de sistemas hipertexto, que buscavam uma terminologia comum e uma referência arquitetural que permitisse intercâmbio entre os diversos sistemas que se desenvolviam na área de hipertexto. Segundo Halasz; Schwartz, o modelo se originou a partir de dois *workshops* sobre hipertexto — o primeiro deles realizado em 1988 em Dexter In, New Hampshire; razão para o nome do modelo. Grande parte das discussões foi promovida para a identificação das abstrações comuns dos principais sistemas hipertexto existentes na época.

Por se caracterizar como um modelo de referência para sistemas hipertexto, seus criadores nomearam as abstrações de **componentes**. O termo “nó” não é utilizado devido à grande variação semântica no uso deste termo entre os sistemas hipertexto existentes. Os componentes do modelo Dexter, entretanto, correspondem aos típicos “nós” que se interconectam através dos componentes “*links*”. Por exemplo, os seguintes termos são adotados com o sentido de “nós”: *cards* no *NoteCards* de Halasz (1988) e no *HyperCard* de Apple Computer (1989); *frames* no *KMS (Knowledge Management System)* de Akscyn et al. (1988); *documents* no *Intermedia* de Yankelovich et al. (1988); *articles* no *Hyperties* de Shneiderman (1989).

O modelo Dexter divide um sistema hipertexto em três camadas, como mostra a Figura 2.4(b): camada de tempo-de-execução (*run-time*), de armazenamento (*storage*) e interna-aos-componentes (*within-component*). A camada de armazenamento é o principal foco do modelo e é onde se encontra a modelagem da estrutura básica da rede de componentes nós e *links*. É nessa camada que se encontra a descrição da Base de Dados composta da hierarquia de componentes, contendo dados interconectados por ligações relacionais.

Há duas partes importantes que intermediam as três camadas principais. A primeira interface, entre a rede contida na camada de armazenamento e o conteúdo e

estrutura internos aos componentes, é identificada no modelo Dexter como uma parte crítica, pois um sistema hipertexto exige um mecanismo para referenciar (endereçar) posições ou itens dentro de um conteúdo de um componente individual. Este mecanismo, chamado **Ancoragem**, é necessário, por exemplo, para suportar os *links* de porções a porções (*span-to-span*) de conteúdo dentro dos nós e não somente entre nós inteiros.

A segunda interface, **Especificação da apresentação**, entre as camadas de armazenamento e de tempo-de-execução, fornece informações para a apresentação do conteúdo de um componente ao usuário e informações de codificação na camada de armazenamento. Assim, a forma de apresentação de um componente não depende somente do *browser*, recurso responsável pela apresentação que o sistema hipertexto possui, mas pode ser uma propriedade do próprio componente e/ou do caminho de acesso que levou à obtenção do mesmo.

Dentre os exemplos mais recentes de sistemas hipertexto protótipos baseados em Dexter encontram-se o Devise Hypermedia (DHM) desenvolvido por Gronbaek; Trigg (1994) e o sistema HyperProp de Soares et al. (1995).

2.2.3 HAM × Dexter: uma análise comparativa

A Figura 2.4 destaca que as arquiteturas propostas por esses dois modelos têm em comum a identificação de uma camada central, responsável pela “resolução” das interligações entre os componentes cujo conteúdo se encontra disponível pelos mecanismos convencionais, nos dispositivos de memória dos computadores. Essa camada central é conhecida também por **máquina de hipertexto** (*hypertext engine*) e corresponde ao nível da máquina HAM e à camada de armazenamento em Dexter.

Nas propostas dos dois modelos o enfoque é dado sobre essa máquina, demonstrando que esse é o ponto diferencial para que outras aplicações façam uso da tecnologia de hipertextos sem maiores esforços de implementação. Segundo Schütt;

Streitz (1990), o grande êxito da funcionalidade dos sistemas hipertexto reside no fato dela ser independente da aplicação e, portanto, poder ser implementada como uma camada intermediária entre a aplicação e o sistema de armazenamento persistente. Tal programa é chamado por Schütt e Streitz de *hypermedia engine* e de fato espelha a máquina de hipertexto referida neste trabalho.

Finalmente, deve-se notar que a concepção do modelo HAM data anteriormente ao modelo Dexter, o qual acrescenta duas interfaces entre as camadas: de especificação da apresentação e de ancoragem. A identificação dessas duas partes intermediárias representa um aprimoramento na estruturação da arquitetura, uma vez que o modelo Dexter se trata de um modelo de referência e, além de conduzir a uma abordagem mais independente dos diversos tipos de sistemas de armazenamento de arquivos e gerenciadores de Base de Dados, ressalta a importância de se ter alternativas para o processamento tanto da interface de ancoragem quanto da especificação de apresentação.

Um exemplo prático bem sucedido que se baseou nos dois modelos HAM e Dexter é o sistema hipertexto SEPIA (*Structured Elicitation and Processing of Ideas for Authoring*). Sua arquitetura se baseia na proposta da máquina HAM, incluindo muitos conceitos derivados do modelo Dexter, relacionados às camadas de armazenamento e interna-aos-componentes. O SEPIA foi desenvolvido por Schütt; Streitz (1990) com o objetivo de dar suporte à atividade de autoria. A Figura 2.5 mostra o módulo de gerenciamento de objetos de SEPIA.

Figura 2.5 - Módulo de gerenciamento de objetos do sistema hipertexto SEPIA utilizado por Schütt; Streitz (1990)

No sistema SEPIA, a máquina hipermídia independente da aplicação é denominada *HyperBase*. O modelo de dados da *HyperBase* se fundamenta numa visão orientada a objetos das principais entidades relevantes das aplicações de hipertexto existentes. Essa máquina hipermídia é responsável pela criação, manipulação e armazenamento dos nós, *links*, objetos de composição, atributos definidos pelos usuários e pela manutenção da história dos objetos. Observa-se que o sistema conta com o SGBD *Sybase* para o processamento dos arquivos referentes aos componentes da base de nós e *links*.

Os esforços na definição de novas arquiteturas mostram a tendência de aprimorar com mais funcionalidades tanto o modelo Dexter quanto o modelo HAM, incluindo especificações mais detalhadas em suas camadas. Esse é o caso da arquitetura MMVP (*Model-Map-View-Praxis*) de Arents; Bogaerts (1991). Essa arquitetura é dita ser para sistemas hipermídia de “terceira ordem” por incluir uma separação funcional clara entre os componentes de representação das informações e a manipulação da semântica desses componentes. O objetivo dessa arquitetura é introduzir um sistema baseado em conhecimento que possibilite a representação explícita e manipulação da semântica das informações.

Em suma, pode-se concluir que o formalismo para arquitetura de sistemas hipertexto ficou estabelecido com os modelos HAM e Dexter. É importante notar que diversas soluções na forma de implementações continuam a ser desenvolvidas. No que diz respeito à camada de armazenamento, por exemplo, existem diversas pesquisas que têm sido apresentadas na literatura, como será visto na próxima seção.

Por ser o alvo da pesquisa de avaliação desta tese, a próxima seção descreve as diferentes formas para representação das informações contidas na camada de armazenamento, ou seja, as técnicas para representação da estruturação dos hiperdocumentos.

2.3 Representação de hiperdocumentos

Esta seção resume as principais técnicas que podem ser utilizadas para representação de hiperdocumentos. Com base em suas características, são discutidas as vantagens de cada uma delas.

A partir da utilização de uma determinada técnica para a representação de um hiperdocumento, pode-se avaliar sua estrutura, independentemente do sistema hipertexto que vai ser utilizado.

Entre as diversas formas de se representar um hiperdocumento, distinguem-se duas abordagens principais que refletem prioritariamente a representação dos intra-relacionamentos ou dos inter-relacionamentos, conceitos esses discutidos na próxima seção. São também apresentadas as técnicas utilizadas para essas duas abordagens de representação, as quais são relacionadas à estruturação dos nós e à estruturação dos *links* de um hiperdocumento, respectivamente.

2.3.1 Conceituação de estrutura dos hiperdocumentos

Como já mencionado na Seção 2.1, os sistemas hipertexto devem tratar de um novo “esquema de representação”, pois o hiperdocumento é um tipo de “rede semântica” que contém material informal (como textos e informações de outras mídias) a ser processado automaticamente. Para que seja possível a definição desse esquema de representação, é necessário primeiro se obter informações sobre essa “rede semântica”. Os dados que podem ser diretamente obtidos dessa rede de nós e *links* são os conteúdos dos nós e a sua organização. Além disso, a organização compreende tanto a disposição dos conteúdos (os *links*) quanto a disposição dentro dos conteúdos dos nós. Esses dois tipos de disposição são conhecidos por estrutura.

Wang et al. (1992) apresentam uma abordagem interessante analisando o conceito de relacionamento das entidades de um sistema. O conceito de relacionamento é bastante usado para o projeto e análise de sistemas do mundo real, como é o caso do Modelo Entidade Relacionamento (MER). Existem diferentes tipos de relacionamento em um ambiente de aplicação complexo e não é uma tarefa fácil o uso limitado de tipos de relacionamento para representar cada situação nesses ambientes. Os relacionamentos complexos podem ser classificados em duas classes principais: **intra-relacionamento** e **inter-relacionamento**.

Os intra-relacionamentos descrevem a estrutura interna de um objeto específico, ao passo que os inter-relacionamentos revelam a conexão entre objetos diversos. Tal classificação de tipos de relacionamentos, apresentada por Wang et al., tem o objetivo de argumentar que a maioria dos *links* definidos nos hiperdocumentos existentes são do tipo intra-relacionamentos, que revelam os significados semânticos da sua estrutura interna. Concluem que é importante que um sistema hipertexto tenha capacidade de apresentar convenientemente ambos, os intra-relacionamentos e os inter-relacionamentos.

Fazendo uma analogia de um objeto de um hiperdocumento como um nó, pode-se dizer que a estruturação de *links* do hiperdocumento corresponde ao inter-relacionamento entre os nós e que a estruturação dos nós corresponde ao intra-relacionamento entre seus componentes internos, de conteúdos de informação.

O autor do hiperdocumento geralmente define essa estrutura visando permitir um melhor entendimento pelo leitor do hiperespaço planejado. A Seção 2.3.3 descreve as principais técnicas para representação dos inter-relacionamentos existentes em um hiperdocumento.

O Capítulo 3 aborda outros trabalhos que propõem a modelagem do hiperdocumento e fornecem uma forma de planejar o hiperespaço de nós e *links*, e de adicionar recursos não contemplados na concepção original, tais como a definição de abstrações e o controle de versões.

Com relação às formas de representação dos intra-relacionamentos, várias pesquisas demonstraram que os documentos estruturados são factíveis em aplicações que utilizam a tecnologia de hipertexto. No entanto, durante a autoria em aplicações reais, mesmo se a documentação for bem definida e estruturada, muitos outros objetos estruturados ainda precisam ser definidos e mantidos.

Nesse contexto, a estruturação dos nós contidos no hiperdocumento tem recebido atenção. As pesquisas encontradas na literatura, cujos trabalhos se relacionam à estruturação dos nós, tiveram origem na busca por uma representação abstrata do hiperdocumento que permitisse sua transmissão e armazenamento em diferentes plataformas, sejam elas de hardware ou de software. Essa representação abstrata do hiperdocumento, que inclui a estruturação de todos os seus componentes (nós e *links*), garante com que os nós sejam documentos logicamente estruturados. Na próxima seção são descritos os principais padrões para hiperdocumentos, que fornecem meios de representação abstrata da estrutura lógica do hiperdocumento para que possa ser intercambiada entre diversas plataformas.

2.3.2 Técnicas para representação da estruturação dos nós

Para a representação de intra-relacionamentos, são utilizados padrões que fornecem meios para a descrição da estrutura lógica de documentos. A Figura 2.6 mostra o exemplo de um nó de informação e seus possíveis componentes lógicos. A representação dessa estrutura deve retratar fielmente a composição do nó, independentemente da forma de armazenamento utilizada, ou seja, deve especificar a ordem dos componentes, a localização física do conteúdo dos componentes e outras informações relevantes à montagem para uma posterior apresentação da composição.

Quando se tem a estrutura lógica básica de um documento, muitas vantagens podem ser obtidas por meio de sistemas de processamento de documentos.

Idealmente, de acordo com Brown (1989), a estrutura reflete a maneira com que o autor, ou editor, idealizou o documento, e permite que os sistemas de processamento de documentos forneçam um auxílio importante ao produzirem automaticamente, por exemplo, numeração de seções, verificações de referências-cruzadas e índices analíticos e remissivos. Dessa forma, torna-se possível e viável a integração de textos, gráficos e outros tipos de conteúdos de documentos. Portanto, a preservação da estrutura lógica durante o armazenamento e transmissão de documentos é fundamental para que essas vantagens automatizadas não sejam perdidas.

Figura 2.6 - Exemplo de componentes de informação de um nó de hiperdocumento.

Entre os primeiros sistemas de processamento de documentos estruturados encontra-se o *Grif*, desenvolvido por Quint; Vatton (1989), que tem as vantagens apontadas, mas não fornece um mecanismo genérico para a representação de relacionamentos mais complexos e dinâmicos entre os componentes de um documento.

Atualmente não existe um consenso sobre o modelo "ideal" para a representação da estrutura de um documento. Com a rapidez no desenvolvimento dos sistemas computacionais, têm surgido modelos incompatíveis e particulares, acarretando problemas relacionados à portabilidade dos documentos. Nesse contexto, surgiram os padrões internacionais para representação visando intercâmbio de

documentos. Os padrões que se destacam atualmente são: SGML, HTML, HyTime, MHEG e ODA. Nas próximas subseções, esses padrões são brevemente descritos.

2.3.2.1 SGML

A SGML (*Standard Generalized Markup Language*), especificada em ISO 8879 (1986) é uma representação muito difundida, sendo o mais antigo padrão da ISO (*International Organization for Standardization*) para descrição genérica de documentos. Trata-se de uma linguagem bastante geral, amplamente utilizada para representar estruturas de documentos para processamento de textos, recuperação de informações, intercâmbio de documentos e distribuição *online* em diferentes plataformas, segundo Herwijnen (1994).

A SGML permite que o autor do documento escreva as instruções de processamento de documentos, conhecidas por *markups* (marcações), para a definição de sua estrutura lógica. Um documento em SGML consiste de objetos de várias classes (capítulos, títulos, referências, objetos gráficos, etc.) chamados **elementos** e cada um deles possui um tipo genérico. Os limites das ocorrências desses elementos nos documentos são marcados com *tags* de início e fim. Por exemplo, uma citação poderia ser descrita e armazenada da seguinte forma:

<quotation> Com SGML é possível a descrição da estrutura do documento diretamente. **</quotation>**

onde **<quotation>** e **</quotation>** são as *tags* neste exemplo.

A linguagem SGML fornece meios para descrever o que fazer com os elementos rotulados e evita que eles sejam exclusivos de um programa ou propósito. O processamento do documento para diversos propósitos é especificado separadamente, geralmente através de folhas de estilo (*style sheets*) ou mecanismos similares.

Uma característica importante para o fornecimento das informações da estruturação dos documentos é a estrutura hierárquica dos elementos SGML. Os elementos podem ser subordinados a outros (um capítulo pode conter um título e várias seções, que por sua vez pode conter outros elementos), à semelhança com a estrutura de dados “árvore”. Dessa forma, pode-se criar, manipular e administrar não somente “títulos de capítulos”, como o fazem os processadores de texto, mas também capítulos como unidades inteiras, através do acesso às “sub-árvores” referentes aos capítulos.

O padrão SGML não estabelece os tipos de componentes que podem aparecer em um documento e como eles se relacionam. Tais definições podem ser especificadas como parte do próprio documento em SGML. Um documento em SGML é dividido em três partes:

- i. Uma **Declaração SGML**, que fornece informações gerais sobre as opções de sintaxe SGML que serão utilizadas. Nessa declaração são definidos, por exemplo, o conjunto de caracteres que será utilizado, o comprimento dos nomes dos tipos dos elementos e os caracteres delimitadores (*tags*) do próprio documento (como “<” e “</”). Entretanto, muitos programas *parsers* permitem a omissão dessa declaração, uma vez que o conjunto *default* do padrão, denominado **reference concrete syntax (rcs)**, é geralmente utilizado.
- ii. Uma **Declaração do Tipo do Documento (DTD)**, que indica os tipos de elementos e outros componentes que são permitidos no documento, por meio de declarações detalhadas dos *markups*. Essas declarações também definem a hierarquia entre os tipos de elementos. Com uma DTD, que é declarada antes do início do próprio documento, os elementos específicos contidos naquele tipo de documento são especificados. Após a leitura da DTD, um programa (*validating parser*) pode verificar documentos em SGML e detectar erros ao validar a estrutura previamente definida na DTD.

- iii. Uma **Instância de Documento**, que consiste no conteúdo do documento juntamente com as *tags* e outras marcações que delimitam os elementos de cada tipo.

2.3.2.2 HTML

HTML (*Hypertext Markup Language*) de Berners-Lee (1993) é uma linguagem de marcação de hipertexto, formalmente definida por uma DTD de SGML. Segundo Berners-Lee et al. (1994), tornou-se amplamente utilizada como formato para hiperdocumentos disponíveis na WWW. A maioria dos documentos existentes atualmente na WWW, porém, não utiliza esse formato de forma rígida, pois os *browsers*, em geral, são flexíveis e permitem que nem todas as características de validação oferecidas por um *parser* SGML sejam obrigatórias.

O comportamento interativo que um documento HTML deve descrever é definido pelo elemento âncora, delimitado pelas *tags* *a* e */a*, com o atributo da referência para o nó-destino, denotado pela *tag href*. O conteúdo desses elementos são considerados os *hotspots* (ou “botões de controle”) na apresentação, ou seja, são formatados de maneira distinta na apresentação e sua ativação pelo usuário provoca a apresentação do documento referenciado como nó-destino. Este comportamento é típico dos *links* de hipertexto. Os *links* em HTML são unidirecionais: a seleção de um *hotspot* provoca a apresentação de outro nó, mas este outro nó não possui recursos para atuar como *hotspot* de volta para o nó original.

Outro aspecto distinto de HTML é a utilização da notação **URL** (*Uniform Resource Locator*) para especificar a localização dos documentos na WWW. Por meio de um endereço em URL, a referência ao nó-destino é descrita, numa forma simples e unificada de acesso a arquivos, adotada em toda WWW (Deroose; Durand, 1994). A sintaxe de URL requer, basicamente, a especificação de uma máquina na

Internet, de um protocolo que possa ser usado para recuperar os dados, e o nome de um fragmento de dado a ser recuperado.

A sintaxe genérica definida por Berners-Lee para especificação de localização em URL é dada por:

<esquema>://<local-rede>/<caminho>;<parametros>?<query>#<fragmento>

onde:

- **esquema:** — nome do esquema indicando o método de acesso, ou seja, especifica o tipo do serviço pelo qual o dado é servido (por exemplo http:, gopher:, wais:, ftp:);
- **//<local-rede>** — localização da rede, ou seja, especifica o nome internet do local (*site*) que contém o dado (por exemplo //www.icmsc.sc.usp.br);
- **/<caminho>** — localização do dado na estrutura de diretórios do servidor;
- **;<parametros>** — parâmetros de um objeto;
- **?<query>** — informações para uma consulta;
- **#<fragmento>** — identificador de fragmento.

Por exemplo, um texto em HTML contendo a descrição de um *link* seria codificado da seguinte forma:

<quotation> Com **** HTML **** a definição da estrutura do documento fica embutida. **</quotation>**

e seria apresentado como segue:

Com HTML a definição da estrutura do documento fica embutida.

2.3.2.3 HyTime

HyTime (*Hypermedia/Time-based Document Structuring Language*) é o primeiro padrão oficial para a descrição da estrutura de documentos hipermídia baseados em tempo. Sua especificação está em ISO/IEC 88744 (1992). Esse padrão vai de encontro com duas áreas técnicas de sistemas computacionais que têm crescido e amadurecido bastante nas últimas três décadas: hipertexto e processamento de documentos baseados em declaração de marcações (*markups*) (Newcomb et al., 1991).

O HyTime é fundamentado no padrão SGML, utilizado como a sintaxe para representação dos *links*. A razão pela preferência pelo padrão SGML como o formato de intercâmbio entre documentos, sobre os padrões industriais tais como *PostScript* e RTF (*Rich Text Format*) é que esses últimos descrevem somente formatos e não as estruturas que podem ser apresentadas em um dispositivo de apresentação (*display*) qualquer de diversas formas. SGML é mais poderoso, pois conserva a informação vital que descreve os documentos como uma hierarquia de elementos multimídia e não como uma coleção de composição de páginas. Portanto, HyTime é considerado como uma extensão de SGML no sentido de que as marcações e as DTDs podem ser utilizadas para descrever a estrutura de documentos hipermídia.

Os projetistas do padrão HyTime previram que o SGML não seria suficientemente poderoso para lidar com documentos de hipermídia baseados em tempo, uma vez que não havia suporte para muitos aspectos de hiperdocumentos com mídia baseada em tempo.

No HyTime existe um conjunto de formas arquiteturais (*architectural forms*) para a definição de DTDs hipermídia. Essas formas arquiteturais são agrupadas em seis módulos, a saber:

- *Base* (básico) - requerido por todos os demais módulos, especifica as propriedades que são globais a um documento;

- *Measurement* (medidas) - fornece a um documento a habilidade de representação de conceitos como dimensão, medidas e contabilizações;
- *Location Address* (localização de endereços) - fornece vários meios para especificação de localizações em um documento, que não poderiam ser especificados pelo SGML. Quando o módulo de medidas (*Measurement*) também é usado, as localizações podem ser especificadas utilizando dimensões particulares;
- *Hyperlinks* - fornece o suporte para a definição dos *links* entre as porções de um documento e pode ser chamado pelo módulo de localização de endereços (*Location Address*);
- *Scheduling* (escalador) - coloca os objetos do documento, definidos no módulo *Measurement*, em um espaço de coordenadas finito (*finite coordinate spaces*);
- *Rendition* (interpretador) - expande o módulo escalador (*Scheduling*) para especificar como os eventos em um espaço de coordenadas genérico podem ser mapeados para o espaço de coordenadas de uma apresentação.

A manipulação de múltiplas mídias, no padrão HyTime, conta com recursos detalhados para especificar sistemas e intervalos de coordenadas, sem se preocupar especificamente com a representação, armazenamento e apresentação dos mesmos. Possui também características para planejar e sincronizar as diversas mídias. De acordo com DeRose; Durand (1994), o HyTime será, a princípio, usado para codificar hiperdocumentos com texto, som, gráficos bidimensionais e imagens.

2.3.2.4 MHEG

A ISO tem demonstrado constante preocupação com relação à disponibilidade da tecnologia multimídia e seu emprego nos diversos sistemas computacionais. Para tratar essa problemática fundou o subcomitê JTC1/SC29 ISO/IEC para cuidar da codificação de informações de áudio, imagem, multimídia e hipermídia. Esse

subcomitê consiste em três grupos de trabalho: JBIG/JPEG, MPEG e MHEG, os quais deram seu nome aos padrões por eles desenvolvidos.

O padrão JPEG define algoritmos de compressão de imagens estáticas. Quando aplicados a vídeo, esses algoritmos são freqüentemente chamados de movimento de JPEG. Infelizmente, os algoritmos de compressão não levam em consideração a redundância temporal existente numa seqüência de vídeo, pois o JPEG comprime cada quadro (*frame*) em separado. Para resolver esse problema, foi criado o padrão MPEG, que define um formato de compressão e de intercâmbio para informações em movimento, incluindo áudio.

Ambos, JPEG e MPEG são amplamente utilizados em sistemas multimídia e descrevem somente as informações de conteúdo dos objetos, mas não são capazes de descrever o inter-relacionamento entre os diferentes segmentos de uma apresentação multimídia. Para a definição e padronização estrutural criou-se o padrão conhecido por MHEG (*Multimedia and Hypermedia Information Coding Experts Group*).

O MHEG padroniza as descrições em um nível intermediário entre as representações de baixo-nível utilizadas em JPEG e MPEG e as linguagens e *scripts* estruturados de mais alto-nível, como o HyTime. O padrão MHEG define uma representação das entidades de informação que estejam em unidades de formato final genérico, para que possam ser usadas e intercambiadas em tempo-real pelas aplicações. Os objetos multimídia (MHEG *objects*) definidos através do MHEG atendem às recomendações CCITT, ISO e ainda outros padrões, e podem ser utilizados em arquiteturas e aplicações definidas pelo usuário.

A especificação do padrão MHEG segue a abordagem de orientação a objetos, onde as classes de objetos definidas correspondem a unidades de informações multimídia/hipermídia. O padrão fornece uma descrição de cada classe através da notação de sintaxe abstrata *ASN.1* (*Abstract Syntax Notation* versão 1), cujo objetivo é descrever tipos de dados em uma forma independente de máquina ou aplicação.

2.3.2.5 ODA

ODA (*Office Document Architecture*) é o padrão proposto pela comunidade de computação envolvida com automação de escritórios e comunicação. Segundo Brown (1989), a comunidade de origem do padrão ODA se reflete nas arquiteturas de conteúdo sobre as quais o padrão se fundamenta, particularmente orientadas a atender às recomendações CCITT e denotadas pela sintaxe ASN.1. A especificação de ODA encontra-se em ISO 8613 (1989).

O principal objetivo de ODA é fornecer uma forma padrão simples para transferência de documentos entre diferentes sistemas. Este padrão parte de um modelo hierárquico de documentos orientado a objetos, os quais representam componentes do documento. Atributos fornecem informações sobre esses objetos. Um documento é uma estrutura em forma de árvore, onde o conteúdo é totalmente armazenado em suas folhas. A separação entre a estrutura e o conteúdo é uma característica importante para o fornecimento de um suporte básico para documentos multimídia. Cada objeto folha possui um atributo que define o tipo do seu conteúdo. As regras para manipular os diversos tipos de conteúdo são chamadas de **arquiteturas de conteúdo** (*content architectures*). Já se encontram definidas pelo ODA as arquiteturas para conteúdos de caracteres, gráficos por varredura (*raster graphics*) e gráficos geométricos.

Um documento ODA é descrito por sua estrutura lógica e sua estrutura de *layout*. Ele pode ser intercambiado se estiver no formato ODIF (*Office Document Interchange Format*) ou ODL (*Office Document Language*). A estrutura lógica do documento é dividida em itens que possuem alguma semântica para o autor do hiperdocumento. A estrutura de *layout* é dividida para viabilizar uma representação visível do documento em áreas retangulares. Os objetos lógicos podem representar itens gerais, como títulos e parágrafos, ou itens mais especializados, como números de catálogos e entradas no dicionário. Os objetos de *layout*, por outro lado, representam áreas retangulares dentro das páginas, páginas e conjuntos de páginas. As áreas retangulares também podem estar aninhadas. Por definição, a área que

agrega mais subáreas é chamada “*frame*”, as áreas de mais baixo nível são chamadas “*blocks*”.

Estruturas genéricas de ODA fornecem um meio de definição de classes de documentos ou “estilos”. Elas consistem de dois conjuntos de descrições de objetos (um conjunto para cada uma das estruturas lógica e de *layout*) que definem os tipos e combinações dos objetos permitidos. Assim, a descrição de classe de cada objeto não-folha inclui um atributo chamado “gerador para subordinados”, que define como os objetos podem ser construídos. Esse atributo especifica os subordinados como: opcionais, requeridos, repetidos ou opcionais-repetidos. Além disso, grupos de subordinados podem ser expressos como uma sequência, um agregado ou uma escolha.

2.3.3 Técnicas para representação da estruturação dos *links*

Conforme visto na seção anterior, apesar de existir suporte à organização das informações pertinentes aos nós em um hiperdocumento, essencialmente um hiperdocumento possui os componentes: nós (páginas) e *links* (ligações entre as páginas), como mostra a Figura 2.7.

Figura 2.7 - Componentes básicos da estrutura de um *link* de hiperdocumento

Independentemente do sistema hipertexto utilizado, o hiperdocumento sempre pode ter a estruturação de seus inter-relacionamentos representada e, através dela, pode-se fazer uma avaliação da organização dos *links*.

Diversas técnicas computacionais podem ser usadas para se representar os inter-relacionamentos de um hiperdocumento. Por exemplo, a técnica gráfica dos Diagramas de Transição de Estados (DTE) que foi historicamente introduzida como um método de definição da função de transição de autômatos finitos, em Teoria da Computação. O DTE se trata de um grafo contendo um nó para cada estado do autômato e arestas orientadas rotuladas com os símbolos de entrada para representar as transições. O DTE, por se tratar de um grafo, representa visualmente o comportamento dos estados dos autômatos em função das possíveis entradas, o que o qualifica como uma forma natural de representação.

Posteriormente, as Máquinas de Estados Finitos (MEF) foram amplamente utilizadas para especificar o comportamento de Sistemas Reativos em Engenharia de Software. De fato, grande parte dos métodos disponíveis para especificação e projeto de Sistemas de Tempo Real (STR) fundamenta-se em MEF, segundo Fortes (1991), pois existe o consenso que estados e eventos são a maneira mais natural na qual se basear para representar o comportamento de STR. A descrição desse comportamento é feita, em geral, através de DTE ou alguma forma de extensão de DTE. A semântica para o DTE da Figura 2.8 é:

— *quando o evento x ocorre no estado A, o sistema muda para o estado B.*

Figura 2.8 - Diagrama de Transição de Estados (DTE)

As técnicas propostas encontradas na literatura, geralmente, procuram atender a características mais específicas de hiperdocumentos. A seguir, são apresentados os principais conceitos que subsidiam as técnicas estudadas: grafos direcionados, hipergrafos, redes de Petri e conjuntos.

2.3.3.1 Grafos direcionados

Os **grafos direcionados** (dígrafos) têm sido amplamente adotados como a forma mais natural de representação de hiperdocumentos.

Definição 2.1 - *Dado um conjunto finito $X = \{x_1, x_2, \dots, x_n\}$ não vazio e uma família $U = (u_1, u_2, \dots, u_m)$ de elementos do Produto Cartesiano $X \times X$, diz-se que o par $G = (X, U)$ constitui um grafo.*

Os elementos $x_1, x_2, \dots, x_n$ de X são os **vértices** do grafo G e os elementos $u_1, u_2, \dots, u_m$ de U são as **arestas** do grafo G . Cada aresta $u_i = (x_{ij}, x_{ik})$ determina uma ligação entre x_{ij} e x_{ik} e, geralmente, é representada graficamente por uma linha contínua entre os dois vértices x_{ij} e x_{ik} .

Delisle; Schwartz (1986) utilizam o conceito de grafos para definir que um estado da base de hiperdocumento é uma 3-tupla $\langle N, L, E \rangle$, onde:

- N é um conjunto finito de nós, cada um dos quais contém uma unidade de informação;
- L é um conjunto de rótulos, os quais correspondem aos valores das âncoras;
- E é um conjunto de arcos dirigidos rotulados, descrito pela função $E : N \times L \rightarrow N$

Como E é uma função, esse modelo é equivalente a um autômato de estados finitos determinístico.

A estruturação dos *links* representada por dígrafos focaliza as ligações definidas entre os diversos nós contidos em um hiperdocumento. Entretanto, a semântica de cada *link*, que varia enormemente, e a aleatoriedade da definição de

existência de *links* entre dois nós são aspectos que não podem ser tratados por grafos direcionados simplesmente. É fato que, para a modelagem de muitos problemas, a noção de grafos é insuficiente, uma vez que apenas permite a representação de relações binárias entre elementos do domínio sendo modelado. Já a noção de hipergrafo permite que relações n-árias entre elementos do domínio possam ser definidas formalmente.

2.3.3.2 Hipergrafos

Definição 2.2 - Dado um conjunto $S = \{s_1, s_2, \dots, s_n\}$ e uma família $E = (E_i \mid i \in \{1, \dots, m\})$ de partes de S , diz-se que o par $H = (S, E)$ constitui um **hipergrafo**, se:

- $E_j \neq \emptyset, j = 1, m$
- $\bigcup_{j=1}^m E_j = S$

Os elementos $s_1, s_2, \dots, s_n$ de S são os **nós** do hipergrafo e os elementos $E_1, E_2, \dots, E_m$ de E são as **arestas** do hipergrafo.

Na Figura 2.9 as arestas E_j são representadas por um traço contornando seus elementos se $|E_j| > 2$; e por um traço ligando seus dois elementos se $|E_j| = 2$. Quando $|E_j| = 1$, E_j é representado como um laço. No hipergrafo da Figura 2.9 o conjunto de nós é $S = \{s_1, s_2, s_3, s_4\}$ e o de arestas é $\{E_1, E_2, E_3, E_4\}$, onde $E_1 = (s_1, s_2, s_3)$, $E_2 = (s_2, s_3)$, $E_3 = (s_1, s_4)$, $E_4 = (s_4)$.

Figura 2.9 - Representação gráfica do hipergrafo H , onde $H = (\{s_1, s_2, s_3, s_4\}, \{E_1, E_2, E_3, E_4\})$

Muitas das definições existentes em Teoria dos Grafos podem ser generalizadas para hipergrafos, tais como as de componentes conexos, ciclos, partição, hipergrafo dirigido, hipergrafo acíclico e outras apresentadas por Berge (1973).

É sabido também que existem problemas com relação à representação gráfica de hipergrafo, principalmente no que diz respeito ao conceito de planaridade. Planaridade pode ser informalmente definida em Teoria dos Grafos como a característica que permite representar num único plano todos os vértices e arestas do grafo sem que duas arestas se interceptem. Esta definição não se estende imediatamente para hipergrafos, pois esses tratam explicitamente do caso em que uma aresta pode ter mais do que dois membros e, portanto, passa a corresponder a um caminho entre dois pontos. Johnson; Pollak (1987) propuseram uma generalização do conceito de planaridade para hipergrafos, utilizando uma generalização dos diagramas de Venn para representar a planaridade dos hipergrafos. Entretanto, a generalização proposta atende apenas a determinadas classes de hipergrafos e a determinação da existência desses diagramas de Venn generalizados para qualquer classe de hipergrafos é NP-completo.

Segundo Mäkinen (1990), existem dois métodos amplamente difundidos para descrever um hipergrafo: o que dá ênfase ao hipergrafo como uma coleção de subconjuntos de um dado conjunto (denominado de padrão de subconjuntos) e o que realmente conecta os vértices pertencentes a um mesmo hiperarco (denominado de

padrão de arestas). O problema de desenhar um hipergrafo pode ser reduzido ao problema de desenhar grafos, quando se considera o desenho de hipergrafos no padrão de arestas, ou seja, quando os hiperarcos conectando os vértices são desenhados como curvas. Além disso, o problema de desenhar grafos automaticamente só está totalmente resolvido para alguns casos de grafos. A maioria das propostas de algoritmos para desenhar grafos são de complexidade, no mínimo, equivalente à complexidade dos algoritmos NP-completos. A Figura 2.10 apresenta as duas formas, segundo os diferentes padrões para diagramação de hipergrafos.

Na diagramação de hipergrafos as características de formas, localizações, distâncias e tamanhos não possuem significado; muitos problemas podem ser equacionados através de modelagem via hipergrafos e resolvidos via análise de sua natureza topológica de conectividade.

Figura 2.10 - Diagramas do hipergrafo da Figura 2.9: (a) seguindo o padrão de arestas e (b) seguindo o padrão de subconjuntos.

Os **hipergrafos** foram utilizados por Tompa (1989), para representação da estruturação dos *links* do hiperdocumento, em sua proposta de uma modelagem de dados através de um hipergrafo instanciado. O hipergrafo instanciado corresponde a um esquema dos dados contido no hiperdocumento.

Figura 2.11 - (a) Representação da estrutura de um hipertexto por um grafo dirigido rotulado e (b) Representação por um hipergrafo dirigido rotulado, seguindo exemplo fornecido por Tompa (1989).

O modelo para sistemas de Base de Dados de hipertexto flexíveis (aqui chamadas de hipertextos), como definido por Tompa, possui várias propriedades, a saber:

- uma estrutura relacionando os fragmentos de uma coleção de documentos é separada do conteúdo desses fragmentos, permitindo que outras estruturas possam ser impostas sobre o mesmo material;
- os elementos comuns entre as estruturas podem ser explicitados e, portanto, mais facilmente mantidos;
- a navegação concorrente no hipertexto é possível no modelo, fornecendo as facilidades para construção de poderosas interfaces com o usuário para documentos hipermídia (por exemplo, baseadas em múltiplos dispositivos de saída ou múltiplas janelas);
- visões personalizadas podem ser criadas e mantidas, podendo ser arbitrariamente encadeadas e compartilhadas entre usuários;
- operações práticas de atualização em Bases de Dados são definidas de maneira que o controle de concorrência possa ser implementado;

- um documento hipermídia geral pode ser representado compactamente e navegado eficientemente.

Tompa define um hipertexto com base em hipergrafos rotulados dirigidos, como sendo uma seis-tupla: $D = \langle N, P, R, V, L, E \rangle$, onde:

- N é um conjunto finito de nós;
- P é um conjunto de páginas que são os objetos de dados no sistema (os conteúdos associados aos nós);
- R é um conjunto de leitores para interpretar as páginas no sistema hipermídia,
- $V : N \rightarrow P$ é uma função de valor que mapeia os nós em conteúdos dos dados (na forma de uma página) ,
- L é um conjunto de rótulos e
- E é um conjunto de hiperarcos.

As seguintes propriedades são assumidas:

- Uma página é um objeto com um valor atômico. Um leitor é um interpretador da página, que opera sobre toda ela e a apresenta utilizando algum meio. As operações que criam novos valores a partir dos antigos fazem parte da criação ou edição da página.
- Um rótulo é um valor atômico que pode ser lido, escrito ou comparado por igualdade com um outro rótulo. A habilidade de comparar rótulos fornece o mecanismo de navegar dentro da Base de Dados dos nós.
- O conjunto $E : 2^N \times 2^L \rightarrow 2^N$ é um conjunto de hiperarcos rotulados dirigidos.

No hipergrafo que representa o hiperdocumento, os usuários são posicionados em muitos nós arbitrariamente, em um determinado instante, e dessa forma eles têm acesso concorrente a vários segmentos da Base de Dados. Um estado do usuário compreende o estado da Base de Dados junto com um conjunto $M \subset N$, composto de marcadores que designam os nós atuais. Com esta extensão, a função E captura o

aspecto de compartilhamento das estruturas dos arcos e o conjunto *M* captura o conceito de um conjunto atual de nós.

O conceito de **hipergrafos** foi também utilizado por Watters; Shepherd (1990), mas não para a modelagem do esquema de Base de Dados do hiperdocumento ou para a representação da estrutura do hiperdocumento. Eles usaram hipergrafos para apresentar um modelo de acesso a dados, admitindo até que o esquema pudesse ser representado por hipergrafo. Porém, sua proposta de utilização de hipergrafos é principalmente para modelar instâncias de dados e acesso a essas instâncias. Cada nó representa um item na Base de Dados e um hiperarco representa uma relação entre os conjuntos de nós. Associado a cada nó existe um conjunto de valores de atributos que descrevem o item fundamental. O acesso aos dados é feito através de operadores definidos sobre os hiperarcos e sobre os nós representando os itens da Base de Dados. Esses operadores permitem consulta e navegação entre os nós.

Finalmente, é importante notar que muito embora o conceito de hipergrafos seja relevante para a expressão de relações *n*-árias entre elementos de um domínio, tal conceito não oferece recursos para a expressão da relação de hierarquia entre aqueles elementos, que é fundamental para a representação de composição de nós em hiperdocumentos. O Capítulo 4 retoma o conceito de hipergrafos, no contexto de definição de *higrafos*, que é a técnica de representação proposta neste trabalho.

2.3.3.3 Redes de Petri

O modelo Trellis de hipertexto, criado por Stotts; Furuta (1989), utiliza as **redes de Petri** e sua semântica de execução para descrever a rede de nós e *links* do hiperdocumento. As redes de Petri possuem um formalismo rico que proporciona recursos adicionais para se especificar, além dos *links*, a semântica de navegação. Esta denomina o conjunto das propriedades de navegação que são estabelecidas sobre um hiperdocumento. Por exemplo, controle de acesso, habilitação/desabilitação de botões

(*buttons*) e apresentação simultânea de várias páginas. Assim, quando um autor imagina uma forma de apresentação das páginas, ele pode defini-la previamente como atributo, para que este seja cumprido durante a navegação.

A definição de Stotts; Furuta para um hipertexto H é de uma seis-tupla $H = \langle N, C, W, B, Pl, Pd \rangle$, onde:

- i. $N = \langle S, T, F \rangle$ é uma rede de Petri, sendo que:
 - S é um conjunto de lugares;
 - T é um conjunto de transições;
 - $F = (S \times T) \cup (T \times S)$ é uma relação de fluxo;
- ii. C é um conjunto de conteúdo dos documentos (textos, gráficos, informação de áudio);
- iii. W é um conjunto de janelas;
- iv. B é um conjunto de botões;
- v. Pl é uma projeção lógica para o hiperdocumento;
- vi. Pd é uma projeção de apresentação (*display*) para o hiperdocumento.

É, então, definida a ‘projeção lógica’ Pl de H : dada a estrutura de uma rede de Petri $N = \langle S, T, F \rangle$ em um hipertexto H , a ‘projeção lógica’ de H consiste da três-tupla $Pl = \langle Cl, Wl, Bl \rangle$, na qual:

- i. $Cl : S \rightarrow C \cup \{v\}$; Cl associa um elemento de conteúdo a cada lugar da rede de Petri;
 - ii. $Wl : S \rightarrow W \cup \{v\}$; Wl associa uma janela lógica a cada lugar da rede;
 - iii. $Bl : T \rightarrow B \cup \{v\}$; Bl associa um botão lógico a cada transição da rede;
- e v é o valor nulo.

A “projeção de apresentação” Pd é definida como sendo uma coleção de mapeamentos que associam os botões lógicos e as janelas de um hipertexto a uma representação física e de localização na tela, para que possam ser apresentados.

Pode-se observar que existem dois aspectos importantes tratados indiretamente pela proposta Trellis:

1. A separação do conteúdo da estrutura, permitindo a ocorrência de mais de uma versão de um mesmo hiperdocumento. A estrutura é representada basicamente pela técnica formal de rede de Petri;
2. A projeção de apresentação Pd, permitindo que um conjunto de entidades lógicas sejam representadas de diferentes formas para consumo e que botões de controle não selecionáveis possam ser definidos.

Figura 2.12 - Uma rede de Petri representando a estrutura de um hiperdocumento com múltiplos caminhos de navegação independentes.

Na Figura 2.12 uma rede de Petri representa a estrutura de um hiperdocumento. Nela existem caminhos de navegação ativos, conforme pode ser visto a partir da marcação apresentada, ou seja, os quatro lugares s3, s4, s5 e s6 estão simultaneamente visíveis e a partir deles o usuário pode selecionar três botões distintos, habilitados nas transições t3, t4 e t5. Caso o usuário selecione o botão habilitado na transição t4, a rede poderá então ter duas marcas no lugar s6 e uma

marca no lugar t_7 e, dependendo de condições estabelecidas ao se efetivar a transição t_4 possibilitar, dessa forma, a habilitação de t_6 .

As principais vantagens na utilização do modelo Trellis são:

- Redução da complexidade de especificação da apresentação, pois usando o grafo de alcançabilidade, alguns parâmetros podem ser automaticamente calculados, como, por exemplo, a previsão do número máximo de janelas que pode ser necessário para ler os documentos simultâneos;
- Os caminhos de navegação concorrentes e as sincronizações são bem representadas devido à utilização da teoria de redes de Petri, tipicamente orientada para esse fim;
- Determinação da alcançabilidade ou não dos nós, com o objetivo de se verificar:
 - i. se uma parte do hiperdocumento pode ser atingida;
 - ii. se existe um término;
 - iii. se qualquer estado terminal numa sessão de navegação deixa a tela em branco;
 - iv. se existem relações ou restrições, como, por exemplo, se qualquer sessão de navegação que encontra o elemento Y deve ter passado antes pelo documento Y;
- O controle de acesso a determinadas páginas pode ser especificado a partir da noção de marcação no hiperdocumento;
- Definição de versões personalizadas de um hiperdocumento, incluindo a possibilidade de se ter partes comuns a todas as versões e partes pertencentes a versões específicas.

2.3.3.4 Conjuntos

Segundo Parunak (1991), um hiperdocumento baseado em grafo consiste de uma três-tupla ordenada $H = \langle N, A, L \rangle$, onde:

- $N = \{N_1, \dots, N_m\}$ é um conjunto de **nós** de informações.
- $A = \{A_{11}, \dots, A_{mp}\}$ é um conjunto de **âncoras** ou sub-regiões dentro dos nós, tais que cada nó N_i possui uma ou mais âncoras $\{A_{i1}, \dots, A_{ip}\}$.
- $L = \{\langle A_{ij}, A_{kl} \rangle\}$ é um conjunto de *links* ou pares ordenados de âncoras.

Parunak apresenta também uma definição da hiperbase baseada em conjunto, como sendo um par ordenado $H = \langle N, S \rangle$, onde N é um conjunto dos nós e $S \subseteq 2^N$ é um conjunto de subconjuntos de N . De fato, seu objetivo é focalizar sobre as relações entre os nós de informação, desconsiderando a estrutura definida por meio dos *links*.

Após um estudo minucioso sobre a definição para um hiperdocumento baseado em grafo dada por Parunak, pode-se concluir que ela se encontra incompleta, uma vez que não formaliza algumas situações que podem ser encontradas em hiperdocumentos. A seguir, serão feitas algumas considerações, de maneira a viabilizar a utilização da notação e semântica adotadas nessa definição, quando do levantamento de classes de reuso de componentes de *links* no Capítulo 4.

- O conjunto N pode ser unitário. Isto ocorre quando o hiperdocumento é composto de um único nó. Obviamente, o conjunto $N = \{N_1, \dots, N_m\}$ deve ser um conjunto finito, onde m representa a quantidade dos nós de N ;
- No caso do hiperdocumento ser constituído de apenas um nó, pelo menos duas **âncoras** são necessárias naquele nó (os pontos terminais de um *link*);
- Uma âncora é uma região contida em um nó que identifica e mostra, durante a navegação, a presença de um *link*. Geralmente, existe um tratamento diferente entre as âncoras de partida e de chegada de um *link*. Uma âncora é necessária no ponto de partida (ou origem) de um *link*, pois indica o ponto de ligação com

outro nó e, portanto, pode ser acionada pelo leitor. No ponto de chegada, muitas vezes, uma âncora, tida como o ponto inicial do nó destino do *link*, não precisa ser apresentada ao leitor. Neste caso, será chamada de âncora λ (*lambda*), i.e., a âncora no nó-destino cujo efeito de seguir o *link* que a busca é apresentar o nó-destino inteiro.

- No conjunto de âncoras $A = \{A_{11}, \dots, A_{ip}\}$, $1 \leq i \leq m$ e p deve ser um inteiro positivo ≥ 1 .
- Um *link* pode ser definido também entre duas âncoras de um mesmo nó, ou seja, pelo par $\langle A_{ij}, A_{il} \rangle$, onde A_{ij} e A_{il} são âncoras de um nó N_i .
- O conjunto de âncoras contidas no nó N_i será denotado por A_i . Supondo que o nó N_i possua p âncoras, ou seja, o conjunto de suas âncoras é $\{A_{i1}, \dots, A_{ip}\}$, o número de elementos de A_i é p , ou seja, $|A_i| = p$.
- $L \subseteq A \times A$ e não existe $\langle A_{ij}, A_{ij} \rangle$, para $1 \leq i \leq m$ e $|L| \geq 1$.
- No par $\langle A_{ij}, A_{mp} \rangle \in L$, que representa um *link*, a âncora A_{ij} (j -ésima âncora do nó N_i) é chamada de âncora-fonte e A_{mp} (p -ésima âncora do nó N_m) âncora-destino.

É importante lembrar que a relação de hierarquia entre elementos é fundamental para a representação de composição de nós em hiperdocumentos. A noção de hierarquia pode ser trivialmente modelada através do uso da Teoria dos Conjuntos, cujas principais propriedades são exibidas na Figura 2.13, onde se representa quatro conjuntos, A, B, C e D: D é subconjunto de C, A e D são disjuntos e A possui uma interseção não vazia com B.

Figura 2.13 - Diagramas de Venn/Euler representando conjuntos e suas relações estruturais.

Se, por um lado, grafos e hipergrafos são uma maneira conveniente para representar conjuntos de elementos com alguma(s) relação(ções) especial(is), por outro, os diagramas de Venn/Euler são uma forma mais adequada para a representação de uma *coleção* de conjuntos, com algum relacionamento de hierarquia.

2.4 Considerações finais

A partir dos estudos realizados sobre uma caracterização de hipertextos, pode-se observar que os fundamentos referentes à tecnologia de hipertexto e hiperímídia são recentes; o embasamento teórico ainda está evoluindo. Por exemplo, foi visto que a definição proposta por Parunak comporta melhorias e também que os formalismos têm se tornado mais sofisticados, para que possam ser adotados como convenção. Neste capítulo foi apresentada inicialmente a terminologia relativa às principais características de sistemas hipertexto, juntamente com os processos genéricos que dão suporte à essas características.

Uma primeira iniciativa de se convencionar os fundamentos da tecnologia de hipertextos foi vista, com os modelos HAM e Dexter. Pode ser identificado, em ambos os modelos, que existem diferentes níveis de granularidade de informação a serem tratados em qualquer que seja o sistema hipertexto. Diante disso, uma proposta

de avaliação das informações disponíveis na forma de hiperdocumento deve considerar esses diferentes níveis de granularidade.

Ao final deste capítulo, foram vistas as diferentes técnicas para representação das informações (nós e *links*) contidas em hiperdocumentos em nível estrutural, classificadas em técnicas para representação das estruturas contidas nos nós e técnicas para representação das estruturas definidas pelos *links*.

Deve-se ressaltar que, dentre as técnicas para representação das estruturas definidas pelos *links* encontradas na literatura relacionada até o momento, não existe nenhum trabalho mencionando a utilização da técnica higráfico. Conforme será visto no Capítulo 4, a técnica higráfico é proposta nesta tese para representação da estrutura dos hiperdocumentos, por combinar conceitos das outras técnicas. O argumento principal para utilização dessa técnica é a sua adequação para representar abstrações que podem ser identificadas e construídas a partir de uma análise relacional de *links*. Esta análise relacional, assim como as abstrações consideradas são descritas também no Capítulo 4.

A principal conclusão deste capítulo é que existem várias características dos sistemas hipertexto que podem ser consideradas em uma avaliação, com o objetivo de reduzir os problemas típicos aqui também descritos. O próximo capítulo apresenta as principais tendências de evolução da tecnologia de hipertexto, identificadas na literatura atual.

Capítulo 3

Evolução dos hipertextos

Os problemas inerentes à utilização de hipertextos descritos no capítulo anterior (Seção 2.1.4) e identificados como desorientação do usuário e de sobrecarga cognitiva são os principais motivadores de muitas pesquisas e os responsáveis pela evolução de sistemas hipertexto. Tais problemas interferem na qualidade da interface e da interatividade dos sistemas, na estruturação dos componentes dos hiperdocumentos e de certa forma, refletem na qualidade do hiperdocumento como um todo. Muito embora o termo qualidade seja vago e subjetivo, este trabalho adota qualidade como sinônimo de “adequação para uso” (*fitness for use*), entendendo por isso, a característica do produto que contempla as expectativas e necessidades do usuário (Juran; Gryna, 1970).

Segundo Botafogo; Shneiderman (1992), a maioria das soluções propostas aos problemas inerentes à utilização de hipertextos envolve investimentos em análise de textos e melhorias dos sistemas de navegação disponíveis.

Sob uma perspectiva mais abrangente, além de um maior investimento em análise do conteúdo dos nós e aprimoramento das características funcionais de navegação disponíveis, deve haver uma maior preocupação com a modelagem (o projeto) do hiperdocumento. As soluções propostas em cada um desses campos de

ação (análise de conteúdo, modelagem do hiperdocumento e recursos de navegação), correspondem, de certa forma, a possíveis melhorias que podem ser implementadas em cada uma das camadas identificadas no modelo Dexter (Capítulo 2, Seção 2.2.2) pois lidam com informações que possuem níveis de granularidade equivalentes aos elementos das camadas do modelo Dexter.

Dessa forma, a camada interna-aos-componentes, onde são alocados os conteúdos e estrutura interna dos nós, possui propostas de melhorias nos trabalhos relativos à análise de conteúdo dos documentos. Para a camada de armazenamento, onde se encontra a base da rede de nós e *links*, são investigadas formas para projetos de hiperdocumentos, além da modelagem do domínio de informações da aplicação. Por fim, por meio de mecanismos mais elaborados na interface com o usuário e recursos adicionais para auxílio à navegação, a camada de tempo-de-execução, que trata da dinâmica de apresentação e interação com usuário, tem recebido atenção significativa para ganho de qualidade, principalmente com relação à usabilidade dos sistemas hipertexto.

Neste capítulo a Seção 3.1 apresenta, em linhas gerais, os trabalhos mais relevantes relacionados à análise de conteúdo e melhoria dos recursos de navegação. A seguir, a modelagem do hiperdocumento é abordada do ponto de vista da evolução do processo de autoria, tratada na Seção 3.2 e da evolução dos modelos de hiperdocumento, tratada na Seção 3.3.

Como a busca por soluções dos problemas inerentes a hipertextos deve ser sempre subsidiada por uma avaliação, este capítulo apresenta um estudo das pesquisas relacionadas à avaliação, que têm por objetivo a melhoria da qualidade de hiperdocumentos. Para tanto, uma revisão da literatura atual foi realizada, abrangendo as principais pesquisas relacionadas ao tema: Brown (1990), Botafogo; Shneiderman (1992), Stotts et al. (1992), Rivlin et al. (1994), Hatzimanikatis et al. (1995) e Garzotto et al. (1995). Essa revisão é apresentada na Seção 3.4. Finalmente, as características e tendências das evoluções são discutidas na Seção 3.5.

3.1 Estado da arte da análise de conteúdo e melhorias de navegação

Entre os trabalhos mais representativos na área de análise de textos encontram-se os desenvolvidos por Salton et al. (1994). O foco principal de suas pesquisas objetiva a estruturação e recuperação automática em grandes arquivos-textos, abrangendo: um modelo de processamento vetorial como estratégia para recuperação de informações, estratégias de análise automática de graus de similaridade de textos e geração automática de produtos de hipertexto, ou seja, de hiperdocumentos (ou parte deles).

Quando se trata de textos, o conteúdo dos nós é, na maioria das vezes, uma informação resultante de processos de escrita e formatação. Por esse motivo, os sistemas para verificação de ortografia, corretores gramaticais e sistemas de processamento de linguagem natural são exemplos de ferramentas apropriadas à avaliação automática e auxílio à autoria neste nível de granularidade “fina” de informações contidas no hipertexto. Um bom processador de textos que execute as funções de composição, formatação, verificação de estilo, busca, substituição e justificação é uma ferramenta importante para a tarefa de autoria de conteúdos dos nós. Segundo Perez (1991) e Garzotto et. al (1991b), muitos autores utilizam processadores de textos isolados, externos ao ambiente de autoria para compor os nós, que depois são inseridos ao hiperdocumento em formato ASCII, considerado o Esperanto¹ das ferramentas de processamento de informação digital. Perez ainda enfatiza que boa parte do esforço e tempo dispendidos na autoria é dedicada a essa tarefa.

Bernstein (1990), que propôs um método para descobrir automaticamente os *links* de conteúdo entre unidades relacionadas em um hipertexto de monografias, enfatiza que um hipertexto é inútil caso não forneça aos leitores uma coleção significativa de *links*. Os autores de hipertexto, no entanto, freqüentemente lamentam

¹ Esperanto é uma língua auxiliar de comunicação internacional, elaborada pelo médico judeu-polonês Ludwig Lazar Zamenhof e por ele divulgada em 1887.

a dificuldade que experimentam em descobrir, explicar, e editar os *links* de um hiperdocumento. Halasz (1988) também reforça a queixa com relação à edição dos *links*, como uma atividade maçante.

Com relação ao auxílio e evolução das características funcionais de navegação em sistemas hipertexto, tem-se notado um aumento significativo de pesquisas e de produtos com interfaces mais elaboradas para apresentação e interação nos sistemas hipertexto. Os trabalhos que se enquadram nesse campo de ação são, na sua maioria, dedicados à melhoria da interface, fornecendo recursos de navegação gráfica aos usuários (Hardman, 1995). Exemplos dessa abordagem incluem:

- Diagramas gráficos de visão-geral gerados automaticamente;
Diagramas gráficos de visão-geral de contextos específicos gerados manualmente (Nielsen, 1990b). Utilizados pelo sistema hipertexto HyperCard, onde o espaço de informações do hiperdocumento é mostrado em forma diagramática em diferentes níveis de detalhes, segundo o contexto definido durante a autoria.
- Visões de *web* do Intermedia (*Web Vision*) de Utting; Yankelovich (1989);
- Visões de olho-de-peixe (*fish-eye*) proposto por Furnas (1986);
- Filtros e classificação de *links* de Tomek (1992);
- Agentes de conhecimento de Shibata (1993);
- Visões hierárquicas da estrutura de Feiner (1988).

A avaliação dos sistemas hipertexto com esses recursos é muitas vezes realizada com auxílio de técnicas específicas para avaliação de usabilidade da interface do sistema hipertexto.

Entretanto, por mais distintos que sejam os recursos embutidos nos sistemas, um hiperdocumento que possua uma estruturação bem feita, pode suprir eventuais faltas de um ou outro recurso, quando transportado de um sistema hipertexto para outro (Marshall; Shipman, 1993). Como este trabalho aborda a avaliação de hiperdocumentos com base na sua estruturação, nas próximas subseções será apresentada a evolução das tendências de modelagem de hiperdocumentos, que

corresponde ao nível de granularidade da camada de armazenamento do modelo Dexter. A revisão bibliográfica sugere que as evoluções predominantes direcionam nas atividades de autoria e de modelagem de hiperdocumentos.

3.2 Evolução do processo de autoria

As ferramentas mais específicas de autoria de hipertexto se concentram em auxiliar a atividade de composição dos nós. A criação dos conteúdos dos nós, como já mencionado, é suportada geralmente por ferramentas externas ao sistema hipertexto. As ferramentas específicas de autoria são as que geram listas de *links* e/ou nós e ferramentas para gerenciamento do *design* da tela. Outras ferramentas mais específicas freqüentemente são adicionadas à máquina principal de geração do hiperdocumento: *outliners*, distribuição de conteúdos e importação de arquivos texto. Os sistemas de autoria, a princípio, devem ter disponíveis:

- funções para listar nós e *links*;
- funções para fornecer informações sobre nós;
- ferramentas para anotação de nós (o sistema deve permitir aos autores anexar nós descritivos aos nós existentes).

É fato que, apesar dos autores conseguirem auxílio externo para a edição de textos e gráficos, ferramentas que auxiliem na verificação e atualização de *links* são mais difíceis de serem encontradas.

Young (1990) destaca que a arbitrariedade na definição de *links* entre os nós permite grande flexibilidade mas que, em contrapartida, tem muitas vezes, como resultado, um hiperdocumento onde os usuários facilmente se tornam desorientados. Sempre que possível, é desejável fornecer suporte à “estruturação” do hiperdocumento, de modo a tornar mais fácil seu entendimento e organização e conseqüentemente, sua navegação. Isto pode ser feito através da identificação das estruturas fundamentais, na forma dos conjuntos específicos de dados relacionados. Com o suporte para uso de tais estruturas num sistema hipertexto emergem benefícios

similares àqueles encontrados no uso de construções de programação de alto nível em desenvolvimento de software.

Diante disso, a maior parte dos trabalhos relacionados ao projeto de um hiperdocumento tem procurado enfatizar melhor suporte à atividade de autoria, uma vez que é durante esta atividade que é possível exercer controle sobre a organização das informações.

Segundo Walker (1989), existe uma analogia entre a atividade de autoria de hiperdocumento e a de desenvolvimento de software, devido ao gerenciamento da complexidade que é requerido em ambas as atividades. Ela argumenta que, quando uma entidade cresce em tamanho de várias ordens de magnitude, ocorrem também mudanças qualitativas e não somente quantitativas. A mesma observação vale com relação à escrita e documentação. Walker destaca que a abordagem que a Engenharia de Software aplica para o gerenciamento da complexidade se concentra em dois principais conceitos: **modularidade** e **abstração**.

Modularidade é o resultado do projeto de um software como um pequeno número de partes, os módulos, com os quais o produto software é construído. Abstração é a ação de nomear algo e utilizá-lo pelo que faz, ou seja, enfatizar sua funcionalidade sem considerar seus detalhes de implementação. Uma definição de abstração mais relacionada ao processamento de informações é dada por Garg (1988), que diz ser a abstração o meio pelo qual a informação pode ser armazenada e recuperada a partir de uma estrutura de informação em diferentes níveis de detalhes e a partir de diferentes perspectivas.

Tendo em vista a analogia existente entre a construção de um hiperdocumento e o desenvolvimento de software abordado na perspectiva de Engenharia de Software, o domínio da aplicação em sistemas hipertexto passa a exigir novas considerações. Por exemplo, Garret et al. (1986), baseados na experiência obtida com o desenvolvimento do sistema *Intermedia*, propõem uma categorização de sistemas hipertexto com base na análise dos seus usuários:

1. Um autor, um leitor - o autor usa sua própria Base de Dados de hipertexto e sabe mais ou menos onde estão as informações e o que significam;

2. Um autor, muitos leitores - o autor está produzindo um hiperdocumento para ser distribuído para leitores distintos;
3. Muitos autores, muitos leitores:
 - i. os leitores são também os autores e o objetivo é o crescimento dinâmico do material no hipertexto;
 - ii. os leitores são diferentes dos autores e uma versão de um hipertexto é distribuída para leitores distintos.

Este trabalho se concentra na categoria 3ii, pois 3i. está mais relacionada com a área de trabalho cooperativo assistido por computador, conhecida por CSCW (*Computer-Supported Cooperative Work*).

Hardman (1988) conclui que criar um hiperdocumento se situa como uma atividade cuja dimensão está entre a escrita de um livro texto e a produção de software. O autor precisa tomar decisões iniciais sobre o assunto e a caracterização dos leitores e, então, continuar o projeto do hipertexto propriamente dito. O autor deve estar preparado para investir tempo no projeto, bem como na construção do hipertexto. Por isso, sua atividade se assemelha à atividade de projeto de software e ao seu ciclo de vida, onde as decisões de projeto tomadas prematuramente são mais difíceis de mudar, à medida que a quantidade de informação no hiperdocumento aumenta.

Perez (1991) e Raymond; Tompa (1988) concordam com o uso da abordagem de Engenharia de Hipertextos, embora Perez saliente que os projetistas geralmente trabalham com bases de informações bastante estruturadas, o que nem sempre é verdade com relação ao material disponível aos autores de hiperdocumento. Diante disso, Perez propõe um modelo de planejamento para autoria de hiperdocumentos com a seguinte cronologia de etapas:

1. **Plano** - planejamento, definição de metas, análise da audiência e dos usuários, especificações, recursos, etc.
2. **Montagem das informações** - geração do conteúdo da Base de Dados de informações;

3. **Conversão de informações** - se as informações fazem parte de material já existente, a equipe de autoria deve analisar o conteúdo desse material e proceder à definição de um *parsing* e conversão dos dados para a forma apropriada de conteúdo dos nós.
4. **Geração de material original** - se as informações são criações originais, o plano elaborado na primeira etapa definirá a base que deverá ser preenchida. Um programa *outliner* é recomendado para as definições hierárquicas iniciais.
5. **Organização das informações na Base de Dados** - a utilização de um SGBD (Sistema Gerenciador de Base de Dados) é um auxílio importante nessa etapa. A equipe de autoria deve poder contar com o uso de Base de Dados para definir os campos para entrada e saída dos *links*.
6. **Preparação final** - produção no formato exigido do sistema hipertexto, quando for o caso de formato proprietário.

Deve-se observar que o modelo de planejamento proposto é bastante prático e, embora a etapa de testes não seja considerada (o que numa abordagem de Engenharia de Software é inconcebível), ele destaca a utilização de ferramentas específicas para os ambientes de desenvolvimento de hiperdocumentos.

De acordo com Luther (1994), o processo realizado durante a autoria de um hiperdocumento multimídia se assemelha ao executado durante a produção de um filme cinematográfico, devido à dinâmica exigida para a criação de ambos os produtos resultantes. Além disso, Luther considera que um projeto típico de autoria envolve o desenvolvimento dos seguintes passos:

1. **Concepção** - os objetivos do projeto e o tipo de aplicação são definidos.
2. **Projeto** - é decidido em detalhes o conteúdo usado no projeto e como ele será apresentado. Corresponde à escrita dos *scripts*, projetos de cenas e distribuição dos papéis, na linguagem de cinema.
3. **Obtenção do material** - todos os dados, sons, vídeos e imagens são coletados nos formatos digitais apropriados. Esta fase é conhecida por estágio de produção, em ambientes de cinema.

4. **Montagem** - a estrutura geral é construída, as apresentações são montadas e quaisquer características de interatividade são construídas.
5. **Testes** - a aplicação é executada e é verificado se ela confirma o que o autor realmente pretendia que realizasse. Em cinema, essa etapa é similar à apresentação do filme ou partes dele, que são vistas e aprovadas por pessoas da gerência.
6. **Distribuição** - a aplicação é reproduzida e entregue aos usuários finais.

A execução desses passos durante o processo de autoria exige um novo ambiente de desenvolvimento, com recursos específicos para a obtenção do conteúdo do material (ferramentas de edição e captura de imagens, áudio, e vídeo) e para a montagem da estrutura geral do hiperdocumento.

Uma consideração importante também é que a noção do *link* como alguma coisa fisicamente existente pode limitar a visão para um nível muito baixo de implementação, pois envolve a preocupação com endereços, ponteiros e desvios, de forma semelhante ao que fazem os programadores envolvidos com detalhes de implementação (Salminen et al., 1995). Com os níveis mais altos de abstração, a programação se tornou mais fácil e produtiva, afirmam Yankelovich et al. (1988). Nesse sentido, sob o enfoque de estruturação das ligações, os trabalhos relacionados à modelagem de hiperdocumentos, descritos na próxima seção, têm procurado colaborar fornecendo mecanismos para modelar os *links* de um hiperdocumento com uma visão de mais alto nível.

3.3 Evolução dos modelos de hiperdocumento

Nesta seção são destacadas as características dos principais métodos para modelagem de hiperdocumento encontrados na literatura, que auxiliam um autor na representação de um domínio de informação de um hiperdocumento para posterior disponibilidade em um sistema hipertexto.

De acordo com Hatzimanikatis et al. (1995), na maioria dos casos os autores são também responsáveis pela interface com o usuário, ou seja, eles devem se

preocupar com a forma com que a informação será apresentada aos leitores. Nesse sentido, os modelos para hiperdocumentos não se limitam apenas em definir os relacionamentos conceituais entre os componentes, mas também em estabelecer as composições e os comportamentos dos *links* que serão obedecidos posteriormente, na apresentação.

Uma vez que a atividade de autoria deve procurar projetar o hiperdocumento, torna-se claro que a Base de Dados constituindo a rede de nós e *links* deve ser projetada. Assim, estamos diante de uma abordagem semelhante à modelagem conceitual de Bases de Dados.

A origem do termo **modelo conceitual** para os cientistas de Computação data da década de 70. Segundo Heuser (1991), o termo vem da área de Base de Dados e foi estabelecido pela equipe da Associação Norte-Americana de Normas (ANSI). Os sistemas de processamento de dados construídos antes do surgimento das Bases de Dados tinham o objetivo de atender o domínio da aplicação de uma só classe de usuários. Com o aparecimento dos Sistemas de Base de Dados, uma mesma base começou a servir para atender a diversas classes de usuários com diferentes requisitos funcionais de suas aplicações. Portanto, uma das características fundamentais de Base de Dados é o provimento de dados compartilhados por várias classes de usuários. Desta forma, as Bases de Dados se tornaram um meio de comunicação entre usuários. Para que esta comunicação fosse eficaz e confiável era necessário um mecanismo que fornecesse um entendimento comum do significado dos dados representados na Base de Dados. Esse mecanismo foi fornecido pela descrição centralizada de todos os dados da base e foi denominado **modelo conceitual**.

A partir desse modelo e das evoluções ocorridas, foi denominado **modelo de dados** o mecanismo de descrição geral dos dados e de suas propriedades estáticas, com o objetivo de descrever os estados corretos dos dados na Base de Dados. Para a descrição das propriedades dinâmicas da Base de Dados, que envolvem as alterações dos estados dos dados na Base de Dados, adotou-se a denominação **modelo de comportamento**.

Atualmente, a utilização do termo modelo conceitual foi ampliada visando atender também à área de projeto de sistemas de informação em geral. É definido como um modelo abstrato de uma área de uma organização, excluídos seus detalhes de implementação, e descreve todas as propriedades do sistema modelado.

No contexto de hipertexto, a modelagem de hiperdocumentos pode ser subdividida, segundo o nível de modelagem de seus dados, em:

1. **Modelagem semântica dos dados** — trata da modelagem de domínios de aplicação específicos, definindo meta-estruturas que refletem sua semântica. São exemplos a estruturação de conhecimento do Aquanet, desenvolvida por Marshall et al. (1991) e a política exploratória adotada em gIBIS, ferramenta desenvolvida por Conklin; Begeman (1988). Normalmente os modelos de dados semânticos são construídos sobre modelos existentes de dados lógicos.
2. **Modelagem lógica dos dados** — semelhante à modelagem dos dados para Bases de Dados, trata das definições de abstrações comuns com o objetivo de combinar grandes quantidades de informação em uma base de dados integrada logicamente.

Diversos modelos lógicos de dados têm sido desenvolvidos para hiperdocumentos. Eles se baseiam principalmente na noção de composição e *links* rotulados/ tipados. Os *links* representam um papel de conexão semântica em sistemas hipertexto, segundo Wang et al. (1992) e são o suporte às conexões diretas entre os nós.

Os trabalhos que propõem a modelagem lógica dos dados de um hiperdocumento visam fornecer uma forma de planejar o hiperespaço de nós e *links* e de adicionar recursos não contemplados na concepção original, tais como a definição de abstrações e o controle de versões.

Na realidade, muito embora muitos dos trabalhos não tratem exclusivamente da modelagem lógica do hiperdocumento, pois também fornecem especificações de características funcionais para sistemas hipertexto ou hipermídia, eles se fundamentam prioritariamente em modelagem de dados.

Os trabalhos encontrados na literatura com propostas de modelagem lógica de um hiperdocumento se baseiam em um dos dois enfoques de modelagem lógica de dados, a saber:

1. **Estruturação das ligações** contidas no hiperdocumento - se referem aos modelos de hiperdocumentos que consideram principalmente a definição dos *links*. Uma vez que seja conhecido o domínio da aplicação ou da informação à qual o hiperdocumento servirá de base, os modelos sob este enfoque de estruturação de *links* visam definir os diversos tipos de ligação que darão a estrutura geral de um hiperdocumento, baseados na identificação das abstrações das informações (como composições, generalizações e outras).
2. **Estruturação dos nós** contidos no hiperdocumento - se referem aos modelos para hiperdocumentos que se orientam principalmente por mecanismos que garantam que os nós sejam documentos logicamente estruturados. Documentos ODA e SGML são estruturados e bem definidos. Nessa linha, o surgimento de padrões e de meta-linguagens que os suportem têm se destacado.

A modelagem de um hiperdocumento, sob o enfoque de estruturação das ligações, tem o objetivo de organizar de forma coerente as estruturas do hiperdocumento, ou seja, as ligações entre os nós, de forma a permitir um melhor entendimento pelo leitor do hiperespaço planejado pelo autor e auxiliar o autor no próprio desenvolvimento do hiperdocumento.

Nesse contexto, uma avaliação do hiperdocumento sugere que sejam validadas as estruturas criadas ou examinadas as ligações definidas, com relação à organização pretendida pelo autor. Pode-se notar que sob este enfoque a atividade de autoria se assemelha a uma atividade de definição de esquemas de Base de Dados. Portanto, a validação das estruturas algumas vezes conta com o suporte de SGBDs disponíveis.

Sob o enfoque de estruturação dos nós contidos no hiperdocumento, as pesquisas encontradas na literatura tiveram origem na busca por uma representação abstrata do hiperdocumento que permitisse sua transmissão e armazenamento em diferentes plataformas, sejam elas de hardware ou de software. Essa representação abstrata do hiperdocumento, que inclui a estruturação de todos os seus componentes

(nós e *links*), garante que os nós sejam documentos logicamente estruturados. Na Seção 2.3, que aborda a representação de hiperdocumentos, foram descritos os principais padrões para hiperdocumentos, que fornecem meios de representação abstrata da estrutura lógica de forma que possa ser intercambiadas em diversas plataformas.

Os sistemas hipertexto como HyperBase (máquina hipermídia sobre um SGBD relacional) de Schütt e Streitz (1990) e InterSect (sistema hipertexto integrado a uma Base de Dados orientada a objetos, cujo modelo é descrito na linguagem de especificação Z) de Wang et al. (1992), são exemplos importantes de implementações realizadas sobre SGBDs já existentes, que tiveram uma aceitação real da adequação de um SGBD para suportar o acesso à rede de informações existente em um hiperdocumento.

Nas próximas subseções são apresentados os modelos encontrados na literatura que propõem a modelagem lógica de um hiperdocumento, com recursos para definição de esquemas conceituais de dados.

3.3.1 HDM

O HDM (*Hypertext Design Model*) de Garzotto et al. (1991a, 1993) foi o primeiro método formal de projeto de hiperdocumento voltado especialmente para aplicações hipermídia. Ele sugere a utilização uma abordagem estruturada para o desenvolvimento de hiperdocumentos, com a modelagem do domínio de informações de uma aplicação como suporte para a autoria das páginas e de suas ligações. Esta proposta visa diminuir os dois problemas típicos da utilização de sistemas hipertexto, desorientação do usuário e sobrecarga cognitiva, vistos no capítulo anterior.

Dentre as inovações de HDM estão: a noção de perspectiva, a identificação de diferentes categorias de *links* (**estruturais, de aplicação e de perspectiva**) com diferentes papéis de representação, a distinção entre a hiperbase (como é denominada a base de documentos) e as estruturas de acesso e a possibilidade de integrar facilmente a estrutura de uma aplicação hipertexto à sua semântica de navegação.

Um projetista de aplicação hipertexto se confronta com pelo menos duas categorias de tarefas fortemente correlacionadas:

- **globais** - atividades como definir classes gerais de elementos de informação e estruturas de navegação das aplicações, e
- **locais** - atividades relativas ao preenchimento dos conteúdos dos nós.

A distinção de duas categorias de tarefas, globais e locais, se assemelha à abordagem adotada no desenvolvimento de um sistema de software altamente modular, ou seja, projetar a topologia e as interconexões entre os módulos é diferente de escrever o código correspondente aos conteúdos dos módulos. Por analogia com Engenharia de Software, Garzotto et al. usam a terminologia: autoria em ponto grande — para se referir à especificação e projeto de aspectos estruturais e globais da aplicação hipertexto — e autoria em ponto pequeno para se referir ao desenvolvimento dos conteúdos dos nós. A autoria em ponto grande permite a descrição de classes gerais de elementos de informação e estruturas navegacionais de aplicações complexas sem considerar os detalhes de implementação, de uma maneira independente do sistema hipertexto que vai ser utilizado.

O HDM pode ainda ser usado de diferentes maneiras: como uma máquina de modelagem ou como uma máquina de implementação. Como uma máquina de modelagem, ele suporta a produção de especificações de alto nível de aplicações existentes ou a serem desenvolvidas; como uma máquina de implementação, ele é a base para ferramentas de projeto que suportam diretamente o desenvolvimento de aplicações.

Uma das principais vantagens da modelagem através de HDM para projeto e construção de aplicações hipertexto é a derivação automática da definição de um número significativo de *links* a partir de uma descrição em nível de projeto conceitual.

3.3.2 MCA

Para a definição de hiperdocumentos utilizando-se o MCA (Modelo de Contextos Aninhados) de Casanova et al. (1991), que em inglês é conhecido por NCM (*Nested Context Model*), parte-se dos conceitos convencionais de nós e *links*, com a adição de duas classes de nós, que constituem o conceito fundamental do modelo: os nós terminais e os nós de contexto.

Intuitivamente, um **nó terminal** contém dados cuja estrutura interna, se houver, é tratada no nível da aplicação e não é, portanto, modelada através do MCA. Esses nós terminais são objetos multimídia especificados no padrão MHEG que, por sua vez, encarrega-se de fornecer recursos para a definição dos intra-relacionamentos. A classe de nós terminais pode ser especializada em outras classes (texto, voz, imagem, etc.), conforme for necessário à aplicação.

Um **nó de contexto** consiste em um agrupamento de conjuntos de nós terminais e de contexto, recursivamente. Com esta concepção de nó de contexto, o MCA possibilita a organização de conjuntos de nós e fornece um mecanismo para a definição de diferentes visões do mesmo hiperdocumento. Esse conceito de nó de contexto, portanto, generaliza o conceito de mesmo nome introduzido no sistema Neptune de Delisle; Schwartz (1986), as *webs* do Intermedia, descritas por Garret et al. (1986) e os *fileboxes* do Notecards de Halasz (1988).

Um *link*, que basicamente conecta dois nós, liga âncoras sobre nós inteiros ou indiretamente aponta para regiões onde ele atinge os nós. Dessa forma, a modelagem de um nó do hiperdocumento pelo uso do MCA considera que o mesmo possui, além do seu conteúdo, uma máscara que atua como uma interface com as informações externas ao nó. Essa máscara é um mecanismo que possui entradas indiretamente indexadas quando um *link* faz referência “direta” a regiões dentro de um nó.

Figura 3.1 - Exemplo de hiperbase H com três níveis de aninhamento segundo o modelo MCA, conforme proposta de *browser* de Muchaluat; Soares (1995).

Com a possibilidade de modelagem dos nós de contexto e de seus aninhamentos, o MCA define também uma **hiperbase**, que no sistema hipermídia protótipo HyperProp de Soares et al. (1995), pode ser pública ou privada, adequando-se a uma arquitetura aberta. A Figura 3.1 ilustra uma visão de hiperbase, proposta em um *browser* para o HyperProp de Muchaluat; Soares (1995), contendo os aninhamentos de nós terminais e nós de contexto.

3.3.3 AHM

O modelo AHM (*Amsterdam Hypermedia Model*) proposto por Hardman et al. (1993b, 1994) se concentra no modelo de documento para apresentação multimídia interativa. Ele parte do princípio que, para guiar o desenvolvimento de um hiperdocumento por meio de sistemas de autoria, é necessário um modelo de dados bem definido que permita o uso de ferramentas de edição. O modelo possui uma implementação — o sistema de autoria de hipermídia CMIFed (*CWI Multimedia Interchange Format Editor*) de VanRossum et al. (1993) — que serviu de referência para a maioria dos aspectos multimídia do modelo.

O AHM foi desenvolvido como uma extensão do modelo Dexter, com a adição das noções de temporização, atributos de apresentação de alto nível para auxiliar o esforço de composição de nós com conteúdo multimídia e noção de contexto de *links*.

A noção de contextos de *links*, desenvolvida por Hardman et al. (1993a) é um requisito importante que deve ser tratado nas apresentações baseadas-em-tempo que tipicamente consistem em vários elementos de múltiplas mídias simultaneamente ativos. Geralmente não fica óbvio onde o usuário é levado com a ativação de um *link* e não existem regras padronizadas que indiquem o que deve acontecer com as outras partes ativas da apresentação no momento de ativação do *link*. Portanto, foi introduzida a noção de contexto para a fonte e o destino de um *link*, baseando-se na idéia que um contexto torna explícita a parte da apresentação que é afetada quando um *link* é seguido a partir de uma âncora. Dessa forma, com os contextos explícitos de fonte e destino de um *link*, um autor é capaz de estabelecer as características desejadas da apresentação ao seguir um *link*. Pode inclusive optar pela continuidade do contexto corrente.

Para fundamentar a especificação de contexto de *link*, a estrutura *link* do modelo Dexter foi estendida para incluir o contexto associado aos nós terminais do *link* e definir o que acontece ao contexto fonte (contexto corrente) quando o *link* é seguido. O contexto de um *link* é especificado pela associação de cada nó fonte ou destino do *link* com uma âncora e um componente de composição. O modelo AHM também fornece dois tipos de componentes de composição: paralela e escolha.

- A composição **paralela** indica que os filhos do nó deveriam ser ativados em paralelo;
- A composição **escolha** indica que no máximo um dos nós filhos está ativado a qualquer momento.

Essas formas de composição permitem a criação de apresentações que retêm na tela os nós em níveis mais altos da estrutura. O componente de composição define o contexto para o terminal do *link*. A âncora é uma referência para uma ou mais âncoras nos descendentes do componente composto.

Já existem modelos para multimídia que descrevem as relações de temporização e hipertexto que descrevem as relações estruturais, mas nenhum deles é capaz de expressar a complexidade total de uma apresentação hipermídia, ou seja, uma combinação de temporização multimídia e *layout*, com estruturação de

hipertexto. Para fornecer tal modelo, o AHM propõe a descrição dos seguintes tipos fundamentais de relacionamentos, além dos atributos dos itens de cada mídia incluída na apresentação:

1. **relações estruturais** - que definem as conexões lógicas entre os itens, incluindo o agrupamento dos itens a serem apresentados juntos e a especificação dos *links* entre esses agrupamentos;
2. **relações de temporização** - que especificam as dependências da apresentação entre os itens de mídia, que podem estar armazenados em diferentes locais;
3. **especificações de *layout*** - que estabelecem onde a mídia deve ser dimensionada e colocada em tela, em relação a cada uma das outras mídias ou em relação à apresentação como um todo;
4. **interação** - através da navegação, fornece ao usuário final a escolha para “saltar” para a informação relacionada.

Com base nessa proposta, o sistema de autoria CMIFed que foi implementado para apoiar o modelo AHM, constrói uma apresentação hipermídia como uma estrutura hierárquica, cujos nós folhas são componentes atômicos mostrados na apresentação, e cujos nós não-folhas são componentes de composição, contendo uma coleção de outros componentes de composição e/ou componentes atômicos.

Essa estrutura hierárquica também permite que se escolha qual a forma de mostrar os nós filhos (de composição ou atômicos): em paralelo ou sequencial. Dessa forma, tem-se uma primeira definição grosseira da informação de temporização para a apresentação. Cada componente atômico é atribuído a um canal que, na realidade, é um dispositivo de saída lógico mapeado pelo *browser*, em tempo-de-execução, em um dispositivo de saída físico, ou seja, uma área na tela ou um canal de alto-falante.

A partir da descrição do contexto dos *links* se estabelece a especificação e o uso de contextos na forma mais geral possível, a partir da qual são implementadas as idéias centrais do modelo AHM. O modelo exige a definição do contexto fonte e destino para um *link*. Atualmente, o sistema CMIFed deriva automaticamente os contextos fonte e destino a partir da estrutura existente dos terminais do *link*. O

contexto, pela derivação automática, é definido como sendo o filho do nó ancestral mais próximo, do tipo composição escolha, contendo o terminal do *link*.

Figura 3.2 - Exemplos de diferentes contextos de *links*, adaptados dos exemplos utilizados por Hardman et al. (1993a).

Apesar de parecer estranho derivar o contexto a partir da estrutura, no sistema CMIFed as derivações são consideradas a partir dos usos mais prováveis do contexto numa apresentação hipermídia. As ações usualmente desejadas ao seguir um *link* são:

1. uma apresentação está acontecendo e pedaços extras de informação podem ser apresentados juntos (na Figura 3.2, essa situação está esquematizada pelo *link* rotulado por **continue** e cuja âncora é o botão 'CWI') — o contexto do destino é um descendente do contexto do fonte, e o contexto do fonte é retido e continua atuando ao seguir o *link*;
2. o leitor desvia para uma apresentação completamente nova (na Figura 3.2, essa situação está esquematizada pelo *link* rotulado por **substitua''**, cuja âncora fonte

é o texto ‘Roteiros’) — os contextos do fonte e do destino estão em duas apresentações separadas e o contexto do fonte é descartado.

3. a seção em uma apresentação contém várias sub-partes através das quais os leitores podem navegar (na Figura 3.2, essa situação está esquematizada pelo *link* rotulado por *substitua*, cuja âncora fonte é o botão ‘barcos’) — os contextos do fonte e do destino são filhos do mesmo componente de composição escolha. O contexto fonte é substituído pelo contexto destino.
4. voltar para o início de uma apresentação (na Figura 3.2, essa situação pode ser esquematizada pelo *link* rotulado por *substitua* e cuja âncora fonte é o botão ‘índice’) — o contexto do destino é um ancestral do contexto do fonte. A apresentação do contexto do destino cancela o contexto do fonte que estava sendo apresentado. O contexto do fonte poderá ser então re-apresentado como parte do contexto do destino.

Figura 3.3 - (a) Visão hierárquica definida no sistema CMIFed, (b) A respectiva visão de árvore da estrutura definida em (a), segundo Hardman et al. (1994)

Finalmente, a Figura 3.3 ilustra as estruturas definidas pela modelagem através do AHM, onde os componentes atômicos são representados por retângulos cujo preenchimento indica qual canal de mídia o conteúdo do nó utiliza. Nela são esquematizados os diagramas para composição de mídias em paralelo e sequenciais, seguindo uma linha-de-tempo (*timeline*), e a árvore de composição dos nós.

3.3.4 RMM

A motivação para a proposta do método RMM (*Relationship Management Methodology*), que em português pode ser traduzido como Método de Gerenciamento de Relacionamentos, está na necessidade de técnicas para gerenciamento de projeto e de desenvolvimento formal de sistemas, para assegurar que um produto hipermídia alcance seus objetivos e seja terminado dentro das expectativas de tempo e de orçamento previstas. Para tanto, as técnicas convencionais utilizadas na indústria de software devem ser alteradas para atender os seguintes requisitos:

1. O projeto de aplicações hipermídia em geral envolve pessoas com habilidades bastante distintas: autores, bibliotecários, projetistas de conteúdo, artistas, músicos e programadores;
2. O projeto de aplicações hipermídia inclui a captura e organização de uma estrutura de um domínio complexo além de ter que torná-la clara e acessível aos usuários;
3. Os aspectos de multimídia das aplicações hipermídia acrescentam diversas dificuldades.

Isakowitz et al. (1995), criadores do RMM, ainda destacam que o projeto hipermídia é um desafio que atualmente está situado mais como arte do que ciência. Finalmente, a necessidade de prototipação e testes intensivos com o usuário é ainda mais pronunciada porque a tolerância a erros que o usuário de hipermídia possui é bastante baixa.ⁿ

O RMM se destina prioritariamente à classe de aplicações que exibem uma estrutura regular no domínio de interesse, ou seja, em domínios onde existam classes de objetos cujos relacionamentos possam ser definidos entre essas classes e múltiplas instâncias de objetos dentro de cada classe. Além disso, muitas aplicações hipermídia pertencentes a essa classe possuem dados “voláteis” que requerem freqüentes atualizações, o que por sua vez configura a necessidade de meios para sistematizar e automatizar ambos: o desenvolvimento inicial e os subseqüentes processos de atualização. Nesse contexto, o RMM se apresenta como uma abordagem bastante adequada para o projeto de aplicações hipermídia contendo informações estruturadas com alta volatilidade.

O modelo de dados utilizado como base para o RMM é chamado RMDM (*Relationship Management Data Model*). O RMDM fornece uma linguagem para descrever os objetos de informação e os mecanismos de navegação utilizados em aplicações hipermídia. Trata-se de uma linguagem gráfica, semelhante à utilizada nos diagramas do modelo entidade-relacionamento de Base de Dados, que inclui representações para primitivas do domínio e de acesso. São as seguintes as **primitivas de domínio** que modelam as informações sobre o domínio da aplicação:

- **entidades** e seus **atributos**, que representam os objetos físicos ou abstratos definidos na aplicação;
- **relacionamentos associativos**, que podem ser um-para-um e um-para-muitos, e descrevem as associações entre diferentes tipos de entidades;
- **slices** ou “**fatias**”, que são os agrupamentos de atributos. Uma vez que as entidades podem consistir em um grande número de atributos de naturezas diversas, pode ser impraticável ou indesejável apresentar todos os atributos de uma instância de entidade de uma só vez. Daí surge a conceituação de *slices*. Como cada instância da entidade pode ser apresentada por vários *slices*, um usuário pode escolher qual delas deseja ver, no caso da aplicação ter suporte para isso.

O modelo RMDM fornece ainda o suporte ao projeto de navegação através das seguintes **primitivas de acesso**:

- **links unidirecionais e bidirecionais**, que são usados para especificar acesso entre *slices* de uma mesma entidade. Vale ressaltar que esses *links* somente podem ser definidos dentro dos limites de uma entidade.
- **índice**, que atua como *table-of-contents*, ou seja, uma lista de instâncias de entidades que fornece acesso direto a cada item listado;
- **roteiro** (*guided tour*), que implementa um caminho linear através de uma coleção de itens, permitindo ao usuário se movimentar para frente ou para trás no caminho percorrido;
- **agrupamento**, que é uma construção semelhante a um menu, que habilita o acesso a outras partes de um hiperdocumento. Pode-se notar que os índices são tipos especiais de agrupamento e que na realidade os exemplos típicos de agrupamento são encontrados nas telas de entrada de muitas aplicações, que servem para fornecer acesso a índices ou roteiros;
- **condições** (ou predicados lógicos), que qualificam os índices e roteiros e definem as instâncias de uma entidade que vão estar disponíveis nos índices ou roteiros.

Para a navegação entre diferentes entidades, as últimas quatro primitivas de acesso são utilizadas.

3.4 Estado da arte da avaliação da estruturação de hiperdocumentos

Um dos primeiros trabalhos a enfatizar a importância de se avaliar a qualidade de hiperdocumentos foi realizado por Brown (1990). Ele conclui que, numa primeira instância, a avaliação deve sempre contar com o discernimento e bom senso das pessoas envolvidas no processo de desenvolvimento, mas que o auxílio de ferramentas automáticas que analisem as características inerentes ao hiperdocumento é um suporte fundamental para essa atividade humana.

As primeiras métricas automatizadas foram propostas por Botafogo; Shneiderman (1992), e posteriormente utilizadas por Rivlin et al. (1994). Elas se baseiam prioritariamente na aplicação de técnicas baseadas em grafos para a obtenção das medidas relativas às seguintes métricas:

- **compactação (*compactness*)**, que indica a conectividade intrínseca do hiperdocumento. Denotado, nesta tese, por Co. Do ponto de vista do leitor, um alto valor de compactação indica que cada nó possui muitos *links* e, conseqüentemente, existem potencialmente muitos ciclos e referências cruzadas. Por sua vez, um grande número de ciclos e referências cruzadas pode indicar que o hiperdocumento está pouco estruturado e pode desorientar o usuário.
- **estratificação (*stratum*)**, que revela o grau de organização de um hiperdocumento de forma a poder indicar se existe alguma ordem de leitura no mesmo. Denotado, nesta tese, por St. Um valor máximo de estratificação é obtido no caso de um hiperdocumento linear.

Uma proposta de avaliação de hiperdocumento baseada somente no comportamento dos seus *links* foi feita por Stotts et al. (1992). Eles destacam que, em muitos sistemas hipertexto, uma navegação adequada por um hiperdocumento depende das capacidades, características e auxílios navegacionais que fazem parte da implementação do *browser*. Por exemplo, se um leitor navega para um nó que não possui nenhum *link* de saída, voltar ou “desviar” para a página inicial são os únicos recursos que permitem o prosseguimento da sessão de navegação.

Como existe a tendência de que os hiperdocumentos se tornem intercambiáveis entre os diferentes sistemas hipertexto, uma primeira abordagem óbvia, mas complexa, seria tentar padronizar as características de navegação e o comportamento dos sistemas *browser*. Dessa forma, a solução seria construir um sistema *browser* que unisse as capacidades das principais implementações dos sistemas atuais. No entanto, além da complexidade envolvida na definição desse conjunto uniforme com todas as características dos *browsers*, essa abordagem, de certa forma, direciona (ou talvez restrinja) os sistemas futuros, pois as novas

“características” deveriam ser elaboradas atendendo àquele conjunto padronizado de características.

Nesse sentido, uma abordagem alternativa, proposta por Stotts et al. (1992) consiste em reduzir a ênfase das características de *browser* e enfatizar a estrutura inerente do hiperdocumento. Um autor deveria ser capaz de criar a estrutura do documento de maneira que as formas de acesso desejadas sejam inerentemente permitidas pelos *links* ao invés de o serem somente por meio das capacidades do *browser*.

Stotts apresenta um método para analisar as propriedades semânticas de um hiperdocumento verificando unicamente os seus *links*. A partir de uma representação do hiperdocumento descrita por um autômato de *links*, eles propõem a verificação de propriedades de navegação com base no traço desse autômato. Com esse método, o autor pode garantir que o hiperdocumento permitirá determinadas formas de acesso. Qualquer que seja a implementação de *browser*, o método pode ser utilizado, bastando somente que disponibilize a mais simples operação de navegação: seguir a direção do *link*.

O método proposto requer um esforço do autor para criar um modelo mental abstrato do hiperdocumento. Ao invés de visualizar os *links* de um hiperdocumento, juntamente com os nós, como a definição de um grafo direcionado estático, os *links* são vistos como a definição de um programa abstrato, denominado autômato de *links* (*link-automaton*) do hiperdocumento. A partir disso, o trabalho se concentra em propor um mecanismo para responder a consultas sobre as propriedades dinâmicas de um hiperdocumento, isto é, obter as seqüências de *links* permitidas durante a navegação. Para tanto, a técnica básica do método adota:

1. O algoritmo *model checking* descrito por Clarke et al. (1986) (utilizado originalmente no contexto de verificação de sistemas concorrentes) que, adaptado, permite localizar os estados de leitura dentro da estrutura a ser verificada. Essa técnica automática de verificação de programas é utilizada para verificar se as especificações de determinadas propriedades são satisfeitas pelo comportamento do autômato de *links*;

2. Uma notação de lógica temporal de ramificação, chamada HTL*, introduzida para especificar as propriedades que um hiperdocumento deve exibir durante a navegação.

A técnica de autômato de *links* foi aplicada sobre hiperdocumentos originados por dois sistemas hipertexto, Hyperties e Trellis. Por se tratar de uma abordagem inovadora, o enfoque da pesquisa se restringe a viabilizar sua aplicação. Stotts não menciona resultados de medidas ou critérios que tenham sido observados nas aplicações apresentadas. Considerando essa lacuna, foi elaborado como parte deste trabalho um subprojeto para incorporar a técnica proposta por Stotts à abordagem de avaliação de hiperdocumentos. No Capítulo 4, esse subprojeto é situado no contexto da abordagem proposta nesta tese.

De acordo com Hatzimanikatis et al. (1995), não existe uma definição completa de qualidade de hiperdocumento. As metodologias e ferramentas que poderiam ajudar na análise e avaliação da qualidade de hiperdocumentos ainda estão em um estágio bastante primitivo com relação ao auxílio efetivo que deveriam fornecer. A possibilidade de medir os atributos de um hiperdocumento é imprescindível para as atividades de autoria e Engenharia de Hipertexto.

Segundo Hatzimanikatis, os paradigmas úteis e mais adequados para tal tarefa, podem surgir a partir da experiência prática nas áreas de Engenharia de Software e de Métricas de Software. Dessa forma, define um modelo de qualidade de hiperdocumento baseado nas idéias do modelo hierárquico **Fator-Critério-Métrica² (FCM)**, amplamente adotado na área de Qualidade de Software. A abordagem utilizada no modelo FCM é a de identificar atributos-chaves da qualidade de um produto de software, denominados **fatores de qualidade**. Os fatores são abstratos por natureza, assim como a própria qualidade, e devem ser posteriormente decompostos em atributos de nível mais baixo, denominados **critérios de qualidade**. Cada fator pode ter qualquer número de critérios e cada critério pode afetar qualquer número de fatores.

² Modelo de qualidade *Factor-Criteria-Metric* (FCM) de McCall 1977.

Dessa forma, deve se observar que uma análise de qualidade não é tarefa trivial uma vez que o encadeamento de decomposições a partir do conceito abstrato de qualidade é também uma atividade subjetiva.

Uma vez que os critérios de qualidade não são diretamente mensuráveis, um nível a mais de decomposição é necessário para a definição de **métricas** para cada critério. À medida que essa decomposição de fatores em critérios e desses em métricas procede, obtém-se variáveis mensuráveis em um determinado domínio. Observa-se que o FCM pode ser adaptado para ambientes diversos, pois diferentes fatores, critérios e métricas podem ser definidos, conforme o objetivo de qualidade estabelecido.

Os fatores de qualidade importantes, identificados como legibilidade e manutenibilidade de um hiperdocumento, bem como os critérios correspondentes, são definidos por Hatzimanikatis. São também apresentadas as métricas de estrutura que podem ser coletadas para subsidiar a avaliação desses critérios, que podem ser computadas a partir do grafo que representa o hiperdocumento. A maioria dessas métricas é derivada das métricas aplicadas a softwares.

O fator de qualidade **legibilidade** é um conceito que deve indicar a facilidade de se “ler” um hiperdocumento, ou seja, em outras palavras deve mostrar o grau de dificuldade de “leitura” de um hiperdocumento. Esse fator de qualidade representa o esforço requerido do usuário para “ler” o hiperdocumento, ou seja, para navegar sobre ele e entender a estrutura das idéias nele contidas. Dentre as diversas questões relacionadas com esse fator, tem-se:

- ‘O leitor pode facilmente ficar “perdido no hiperespaço”? ’
- ‘Existem mapas ou outras facilidades disponíveis? ’
- ‘O hiperdocumento está consistente? ’, i.e. as idéias contidas nos nós estão organizadas de forma coerente de modo a não possibilitar divergências.
- ‘Os conteúdos dos nós estão corretos e são facilmente compreensíveis? ’

Manutenibilidade é o conceito associado à medida da dificuldade requerida para estender, modificar ou corrigir o hiperdocumento, num momento posterior à sua

criação. Representa o esforço requerido de um autor para manter um hiperdocumento, que pode ter sido criado por outra pessoa. Questões importantes relativas à manutenibilidade são:

— ‘Existe documentação de projeto disponível?’

— ‘Existe, incluído no hiperdocumento, algum comentário do autor sobre a estrutura dos documentos?’

— ‘É fácil entender as intenções e o “estilo de autoria” do autor original?’

Hatzimanikatis destaca também que há uma estreita ligação entre navegação e manutenção. Quanto mais difíceis forem a navegação e a compreensão de um hiperdocumento, mais difícil será sua manutenção. Além disso, quando alguém deseja mudar ou estender um hiperdocumento, deve também ter conhecimento da maneira pela qual o autor original utilizou os efeitos especiais ou estruturas, o chamado “estilo de autoria”, bem como um bom entendimento da estrutura geral do hiperdocumento.

Entretanto, manutenibilidade e legibilidade não podem ser medidas obtidas diretamente. Por isso, esses fatores são decompostos em critérios de nível mais baixo, ou seja, são definidos atributos do hiperdocumento que podem ser mais facilmente medidos, e são definidas a relação de dependência entre tais fatores e seus correspondentes atributos. A maioria dos critérios definidos afeta ambos os fatores, manutenibilidade e legibilidade, a saber:

1. Tamanho - o tamanho real do hiperdocumento. Quanto maior o hiperdocumento, maior a possibilidade de desorientação do leitor. Conseqüentemente, torna-se mais difícil compreendê-lo integralmente quando for preciso realizar alguma manutenção.

2. Complexidade do caminho - o número de caminhos e ciclos do hiperdocumento.

Um hiperdocumento quase linear é facilmente navegado e sua expansão é quase direta. A existência de muitos caminhos de navegação diferentes e um número excessivo de possíveis ciclos aumentam a sobrecarga cognitiva sobre o leitor.

3. **Impureza da árvore** - proporção de desvio do grafo do hiperdocumento quando comparado com uma estrutura do tipo árvore pura. A estrutura de documentos tradicionais impressos é muito semelhante à de uma estrutura de árvore pura e, geralmente, pode ser facilmente compreendida tanto pelos leitores como pelos mantenedores.
4. **Modularidade** - quão fácil é a identificação das partes independentes no grafo que representa o hiperdocumento. Quando o hiperdocumento pode ser decomposto em partes menores, cada uma delas com objetivos claros, torna-se mais fácil adicionar novos nós e *links* e modificá-los para corrigir eventuais falhas. Além disso, a modularização é um critério valioso para a legibilidade do hiperdocumento, uma vez que um hiperdocumento bem projetado e estruturado facilita a navegação.
5. **Complexidade individual do nó** - o número de *links* de entrada e de saída de cada nó. Um grande número de *links* de saída implica em uma alta sobrecarga cognitiva para o leitor e também significa que qualquer mudança feita neste nó pode afetar muitos outros nós.
6. **Coerência** - possibilidade de fácil percepção de um estilo. Quando existe um estilo coerente de apresentação e estruturação, tornam-se mais claras a leitura e a navegação.
7. **Complexidade do conteúdo dos nós** - o tamanho do conteúdo do nó e a quantidade dos diversos tipos de informação, tais como gráficos ou imagens.
8. **Simplicidade** - quantidade de efeitos especiais ou botões de controle e o nível de facilidade para compreensão do conteúdo, da estrutura, e finalmente, do estilo do autor. Muitos efeitos especiais podem atrapalhar a leitura e desviar a atenção do conteúdo real do hiperdocumento. Além disso, é importante para um mantenedor entender o estilo do autor, para garantir a consistência durante a alteração do hiperdocumento.

Interessados em identificar as métricas possíveis de serem automatizadas, Hatzimanikatis et al. utilizam esses critérios, focalizando sobre as métricas estruturais

(ou seja, as métricas que possam ser computadas somente com base na estrutura do hiperdocumento), para **tamanho**, **complexidade do caminho**, **impureza da árvore**, **modularidade** e **complexidade individual do nó**. As métricas para os critérios de coerência, complexidade do conteúdo do nó e simplicidade não são discutidas por eles. As alternativas propostas para obtenção das medidas das métricas escolhidas são:

1. **Tamanho** - pode ser aplicada a métrica de linhas de código, **LOC** (*lines of code*).

Essa medida leva em consideração algumas quantidades básicas de um programa, definidas por Halstead (1977) como: n_1 (número de operadores únicos), n_2 (números de operandos únicos), N_1 (número total de operadores) e N_2 (número total de operandos). Com elas, pode-se calcular:

- Extensão de um programa $Ext = N_1 + N_2$,
- Vocabulário $Voc = n_1 + n_2$,
- Volume $Vol = Ext \times \log_2 Voc$, que é definido como “o número de comparações mentais necessárias para se escrever um programa de extensão N ”.

Ao contrário das outras métricas que serão apresentadas a seguir, a analogia para aplicação dessa métrica na obtenção do tamanho de um hiperdocumento não é direta, mas é sugerido que as **palavras-chaves** que descrevem os nós e os **links** sejam consideradas como operadores e os conteúdos dos nós, as linhas de texto ou figuras sejam considerados como operandos.

Além disso, Brown (1990) sugere a aplicação da métrica **regularidade**, que informa o número de repetições das estruturas. Quando as estruturas e padrões repetidos podem ser identificados e, mais ainda, forem agrupados em classes pela analogia dos operandos e operadores, obtêm-se as medidas relevantes relativas ao tamanho do hiperdocumento.

2. **Complexidade do caminho** - além das propostas por Botafogo; Shneiderman (1992) que foram descritas nesta seção, Hatzimanikatis utiliza as seguintes métricas da área de Engenharia de Software:

- o **número ciclomático** de um programa, denotado nesta tese por N_{cicl} , cujo valor é baseado em seu grafo de controle e é usado para medir a complexidade de caminhos de programas. Este número é dado por:

$$N_{cicl}(G) = e - n + p$$

onde G é o grafo de controle com n vértices (nós), e arestas (*links*) e p componentes conexas.

Analogamente, com base no grafo de um hiperdocumento, pode-se obter o seu número ciclomático. No caso de um hiperdocumento estritamente linear, o seu número ciclomático será igual a 1; este número cresce proporcionalmente ao número de caminhos.

- a **medida da complexidade das estruturas de dados (CED)** definidas nos programas, que se baseia nos caminhos circulares que podem ser encontrados em um grafo e ao mesmo tempo indica o tamanho do grafo. Para seu cálculo, um multigrafo G é produzido a partir das dependências das estruturas de dados de um programa. Para cada componente K fortemente conexa do grafo é produzido um monômio (polinômio com um único termo) da seguinte forma:

$$CED(K) = (n + e) x^L$$

onde K possui n nós, e arestas e L caminhos circulares simples (i.e. aqueles que não incluem caminhos circulares) e x é a variável do polinômio.

Por analogia, a métrica pode ser aplicada ao grafo de um hiperdocumento. Deve-se observar que, nessa métrica, cada nó e aresta de um componente fortemente conexo aumenta sua complexidade, e o número de círculos (L , expoente do monômio) afeta ainda mais a ordem de magnitude da medida obtida.

3. **Impureza da árvore** - algoritmos para identificação de raízes da árvore e hierarquia foram propostos por Botafogo; Shneiderman (1992). As métricas de estratificação e compactação, descritas anteriormente nesta seção, podem também ser utilizadas para obter a medida de impureza da árvore de um hiperdocumento.

Em Engenharia de Software, acredita-se que quanto maior o desvio da estrutura de um projeto em relação à estrutura de árvore pura, pior é o projeto (Kan, 1995). Fenton (1991) apresenta diversas propriedades que uma métrica de impureza de árvore $MI(G)$, para um grafo G , deveria satisfazer. Uma métrica satisfatória é dada pelo número de arestas que excedem a extensão da árvore do grafo G dividido pelo número máximo de arestas que excedem a extensão da árvore.

Sendo e o número de arestas no grafo G e n o número de nós, então a **medida de impureza (MI)** do grafo G em relação a uma árvore é obtida pela seguinte expressão:

$$MI(G) = \frac{2(e - n + 1)}{(n - 1)(n - 2)}$$

De fato, o grafo é comparado a um grafo completo com o mesmo número de nós. Essa métrica caracteriza precisamente a impureza da árvore do projeto de um sistema, bem como do projeto de um hiperdocumento. Deve-se notar que esta métrica independe do tamanho do hiperdocumento.

4. **Modularidade** - duas métricas podem ser aplicadas, a saber:

- a **complexidade do fluxo de informações (CFI)** de um programa (um módulo), dada por:

$$CFI = Ext \times (fanin \times fanout)^2$$

Para o cálculo dessa fórmula, devem ser obtidos os parâmetros *fanin* — definido como o número de fluxos de informação que entram no módulo; os procedimentos que chamam o módulo e as estruturas de dados que o módulo lê — e *fanout* — definido como o número de fluxos de informação que deixam o módulo, os procedimentos que o módulo chama e as estruturas de dados que o módulo atualiza. Considera-se também que, quanto maior o módulo (sua extensão), mais complexo é o módulo.

A analogia para aplicação dessa métrica na obtenção de uma medida da modularidade de um hiperdocumento é feita através da definição de *fanin* como o número de *links* que entram em um nó e de *fanout*, como o número de *links* que partem de um nó.

Com relação à extensão, a métrica de tamanho determina como obter sua medida correspondente.

- O grau de **similaridade entre nós (SN)**. Os trabalhos de Salton et al. (1994) relatam pesquisas para identificação de nós que pertençam a uma mesma “seção”, pelo seu alto grau de similaridade. Observa-se, dessa forma, que o enfoque da linha de pesquisa realizada por Salton; Allan; Buckley não se baseia estritamente na estrutura do hiperdocumento, mas sim no conteúdo dos nós.

Hatzimanikatis et al. salientam que a modularidade é, provavelmente, o critério mais importante para a manutenibilidade de hiperdocumentos.

5. **Complexidade individual do nó (CI)** - a principal medida consiste do número de *links* que partem do nó e o número de *links* que chegam no nó. Se um nó possui muitos *links* (de entrada ou saída) os autores deveriam considerar a subdivisão do seu conteúdo em outros nós.

A mesma fórmula utilizada para a medida de complexidade de modularidade do hiperdocumento, com base em *fanin* e *fanout*, pode então ser usada para a complexidade dos nós individuais, porém sem considerar o fator extensão, pois o tamanho de um nó não contribui para a complexidade individual da sua estrutura. Ao contrário, pode-se argumentar que, no caso de um “grande” nó — consistindo de muitas páginas ou contendo uma combinação de vários conteúdos de texto e gráficos — é natural a existência de muitos *links*. Dessa forma, a fórmula:

$$CI = \frac{(fanin \times fanout)^2}{Ext}$$

seria mais adequada para a complexidade individual do nó.

Finalmente, Hatzimanikatis et al. concluem que experimentos constituem a questão-chave para a aplicação de métricas e, para suportá-los, são necessárias ferramentas flexíveis para a coleta automática dos valores de suas medidas em um hiperdocumento.

Um outro método de avaliação, conhecido como “orientado a projeto”, foi proposto por Garzotto et al. (1995), que relatam sua utilização em um aplicativo comercial bastante popular, o guia hipermídia da coleção de pinturas do *National Gallery* de Londres, “*Art Gallery*” da Microsoft. A ênfase desse método não está na avaliação do projeto de software do sistema hipermídia (para o qual as técnicas de avaliação de software são mais adequadas) nem se refere à avaliação da aplicação sobre um domínio ou necessidade específicos do usuário (para a qual outras técnicas de avaliação específicas à usabilidade são convenientes). O objetivo principal desse método é avaliar o projeto da aplicação hipermídia.

Dessa forma, Garzotto identifica cinco dimensões de um projeto como relevantes: conteúdo, estrutura, apresentação, dinâmica e interação.

1. **Conteúdo** - dimensão que consiste dos segmentos de informação incluídos na aplicação. Podem ser de dois tipos: mídia estática (como dados formatados, cadeias de caracteres, imagens e gráficos) ou mídia dinâmica (como trechos de vídeo, trilhas sonoras e animação). O conteúdo pode ser considerado como a dimensão mais importante na avaliação de eficiência de uma aplicação hipermídia. A análise de conteúdo requer um especialista do domínio da aplicação, bem como um profundo conhecimento do perfil dos usuários e suas atividades.
2. **Estrutura** - é a dimensão que se refere à organização do conteúdo.
3. **Apresentação** - como o conteúdo e as funções da aplicação são mostradas aos leitores. A apresentação inclui a visualização das informações individuais e das características dinâmicas, como a navegação. Para a mídia estática, simplesmente exibe-se as informações; para a mídia dinâmica, uma apresentação mais elaborada deve ser executada.
4. **Dinâmica** - como o usuário pode interagir com os fragmentos individuais de informação (por exemplo, controlando a execução de uma mídia ativa), bem como se movimentar entre eles.
5. **Interação** - constitui-se na utilização das funcionalidades dinâmicas de uma aplicação pela operação sobre os elementos da apresentação. A maioria dos

autores considera que a interação é parte da dinâmica ou da apresentação, uma vez que ela combina esses dois fatores.

O foco principal do método de avaliação “orientada a projeto”, proposto por Garzotto concentra-se nas dimensões de estrutura e dinâmica da aplicação hipermídia. Além disso, o seguinte subconjunto de critérios de avaliação é considerado: **riqueza, facilidade, auto-evidência, previsibilidade, legibilidade, consistência e reuso**.

O critério de **riqueza** expressa a abundância de itens de informação e formas de obtê-los. A **facilidade** mede o grau de facilidade de acesso às informações e o grau de entendimento das operações necessárias para esse acesso. O critério de **auto-evidência** expressa a facilidade dos usuários adivinharem o significado e o propósito do que quer que esteja sendo apresentado (conteúdo ou elemento de navegação). **Previsibilidade** indica a facilidade dos usuários para preverem o resultado de uma operação. A **legibilidade**, considerada como fator de qualidade por Hatzimanikatis, é agora considerada um critério que depende dos outros critérios mencionados e que expressa o “sentimento” geral sobre a validade de uma aplicação.

O critério de **consistência** reflete a regularidade da aplicação, ou seja, avalia-se a forma de tratamento de elementos conceitualmente similares ou distintos.

O critério de **reuso** visa a reutilização de objetos e operações em diferentes contextos ou com propósitos diversos. Quando bem aplicado, o reuso favorece a obtenção de consistência e, portanto, de previsibilidade, diminuindo a sobrecarga cognitiva durante a apresentação e navegação pelo sistema hipertexto. Ao apresentar o mesmo material sob diferentes perspectivas e pontos de vista, a aplicação também é enriquecida.

Além disso, o reuso é um recurso que possui outros méritos originados na área de Engenharia de Software, por reduzir erros, esforço de desenvolvimento e necessidade de espaço físico. A redução da necessidade de espaço físico é particularmente importante no caso de aplicações com grandes quantidades de informação multimídia. O reuso expande muito o tamanho aparente de uma aplicação ao passo que aumenta minimamente seu espaço de armazenamento físico.

Um exemplo de pesquisa relacionada com reuso foi feita por Creech et al. (1991), em que é relatada a experiência na construção de ferramentas baseadas em hipertexto, denominadas KIOSK, como um auxílio ao processo de seleção de software para reuso a partir de uma biblioteca de software reutilizáveis. Essa experiência foi feita nos laboratórios da Hewlett-Packard e foram utilizados componentes (nós) chamados *workproducts* (módulos contendo a documentação, notas de projeto, código-fonte, testes) interligados para formar um hiperdocumento “biblioteca de software reutilizáveis”. A autoria, nesse caso, é feita a partir dessas bibliotecas, com documentos tipicamente estruturados.

3.5 Conclusões e a tendência das evoluções

Neste capítulo foram apresentadas inicialmente as tendências que os sistemas hipertexto têm incorporado objetivando a melhoria da qualidade do hiperdocumento. Pode-se observar que, em linhas gerais, duas principais tendências evolutivas se evidenciam:

1. Adoção de um procedimento mais disciplinado durante a autoria;
2. Inclusão de esquemas conceituais (abstrações) nos modelos para projeto de hiperdocumentos.

O termo Engenharia de Hipertexto foi utilizado pela primeira vez por Glushko et al. (1988) e reflete adequadamente a primeira tendência evolutiva identificada. Segundo ele, surgia “a necessidade de uma abordagem disciplinada de **Engenharia de Hipertexto**, que trabalhasse com a análise da tarefa, análise do documento e considerasse cuidadosamente as lições obtidas a partir de técnicas de escrita e disciplinas de recuperação de informação”.

Com relação à tendência de modelagem conceitual, pode-se observar as abstrações de composição de nós adotadas em cada um dos modelos estudados. Essas abstrações são apresentadas na Tabela 3.1, onde o termo nó-concreto se refere a um nó existente no hiperdocumento.

Analisando os modelos de hiperdocumento estudados, algumas colocações devem ser consideradas. Os modelos MCA e HDM foram divulgados no ano de 1991, sob enfoques diferentes. O modelo HDM se destina prioritariamente a ser um modelo para se projetar uma aplicação hipermídia, contribuindo, portanto, para a nova área de Engenharia de Hipertextos. O modelo MCA se orienta mais especificamente a subsidiar com a definição de estruturas flexíveis e abstratas dos “contextos aninhados”, dando maior contribuição à segunda tendência evolutiva identificada.

Tabela 3.1- Abstrações de nós definidas nos modelos de hiperdocumento estudados

modelo	nó-concreto	abstração
HDM	nó-componente	nó-raíz
MCA	nó-terminal	nó de contexto
AHM	nó-folha	nó de composição (paralelo/ seqüencial)
RMM	<i>slices</i>	entidade

O modelo AHM, divulgado pela primeira vez em 1993, possui uma abordagem específica inexistente até então: o tratamento de multimídia durante o desenvolvimento das aplicações hipermídia. Hardman; Bulterman (1995) descrevem os estágios no processo de geração de um hiperdocumento multimídia e o chamam “*informativo pipeline*”. Portanto, além de enfatizar a necessidade de mecanismos para a definição de abstrações, ou composições em um hiperdocumento multimídia, o AHM contribui também para a Engenharia de Hipertextos.

O modelo mais recente, o RMM, divulgado em 1995, se fundamenta no modelo HDM e apresenta também uma grande contribuição à Engenharia de Hipertextos. Atualmente, é o método que engloba maior número de fases e transições entre elas no desenvolvimento de aplicações hipermídia.

Um outro modelo bastante recente foi proposto por Schwabe et al. (1996) no último Congresso Internacional sobre Hipertextos (*Hypertext'96*). Trata-se de uma proposta de sistemática de projeto baseada no modelo denominado OOHDM (*Object-*

Oriented Hypermedia Design Model). Além de subsidiar as fases de desenvolvimento de hiperdocumentos para aplicações hipermídia, nele existe um diferencial importante que é a visão orientada a objetos na fase de modelagem das informações contidas no hiperdocumento. Deve-se observar que o modelo proposto por Schwabe já considera a implementação de hiperdocumentos em HTML, ou seja, de um servidor de WWW (*World-Wide Web*).

Enfim, pode-se concluir que tem sido cada vez maior a concentração de esforços no sentido de sistematizar o desenvolvimento tanto de sistemas hipertexto quanto das bases de informação por eles manipuladas.

Um aspecto ainda mais desafiante nessa busca por sistemáticas de desenvolvimento é a globalização da base de informações através da WWW, que nos últimos três anos tem se ampliado muito rapidamente. Diante disso, todos os esforços para melhoria de hiperdocumentos também devem considerar aplicações no contexto da WWW. De fato, os trabalhos científicos ainda estão em plena pesquisa e muitos resultados estão por vir. Neste contexto, o trabalho desenvolvido nesta tese vem contribuir com uma experiência e proposta de avaliação das informações disponíveis em servidores WWW.

Para que se possa ter uma visão resumida dos critérios apresentados e as métricas que podem ser obtidas para auxiliar a avaliação de cada critério, a Tabela 3.2 os sumariza juntamente com suas respectivas siglas. Observa-se que, para os critérios descritos por Hatzimanikatis, de coerência, complexidade do conteúdo do nó e simplicidade, não são identificadas métricas. Esses critérios se concentram no conteúdo dos nós e por isso dependem de outro tipo de granularidade de informações.

A Tabela 3.2 também mostra os critérios definidos por Garzotto e como eles se relacionam com os apontados por Hatzimanikatis. Para a identificação dos relacionamentos entre os critérios, foi considerada somente a dimensão estrutural e, portanto, não foram abordados seus aspectos referentes à dimensão da dinâmica do projeto da aplicação hipermídia, conforme inicialmente sugerido por Garzotto.

Tabela 3.2 - Resumo dos critérios e métricas que podem ser analisados em hiperdocumentos, considerando-se sua dimensão estrutural.

Fatores	Critérios		Métricas
	Hatzimanikatis et al.	Garzotto et al.	
legibilidade e manutenibilidade	tamanho	riqueza, reuso	Ext; Voc; Vol; <i>regularidade</i>
	complexidade do caminho	facilidade	Co; St; Ncicl; CED
	impureza da árvore	facilidade, previsibilidade	Co; St; MI
	modularidade	consistência	CFI; SN
	complexidade individual do nó	riqueza	CI
	coerência	consistência, auto-evidência, previsibilidade	—
	complexidade do conteúdo do nó	auto-evidência	—
	simplicidade	auto-evidência	—

Pode-se notar que, excetuando o critério de auto-evidência, os critérios para avaliação de hiperdocumentos de Garzotto podem ser implementados usando métricas derivadas das métricas de software. É importante ressaltar que as propostas de avaliação de hiperdocumento existentes se baseiam prioritariamente na aplicação de técnicas baseadas em grafos para a coleta de medidas.

O critério “tamanho” mencionado por Hatzimanikatis inclui também o critério de “reuso” identificado por Garzotto e pode ser implementado pela métrica “regularidade” proposta por Brown (1990). Entretanto, essa proposta é vaga e limitada, uma vez que aborda apenas a “quantidade de repetições de estruturas”.

Nesta tese é também proposta uma métrica que auxilia prioritariamente a avaliação do critério de reuso, por meio de uma abordagem orientada a *links* baseada na estrutura do hiperdocumento. Deve-se ressaltar que, além do critério de reuso, uma consequência da abordagem proposta é a definição de indicativos de medidas para avaliação dos critérios de consistência e modularidade. Os critérios de auto-evidência, previsibilidade e legibilidade também são contemplados, mas indiretamente. O critério de reuso é ressaltado porque possui relacionamento direto com a abordagem para exame de hiperdocumentos, apresentada no próximo capítulo.

Capítulo 4

Uma proposta de abordagem orientada a *links* para avaliação da estrutura de hiperdocumentos

Nielsen (1995), ao discorrer sobre autoria de hiperdocumentos, menciona que muito embora seja possível obter facilmente os requisitos e informações sobre o processo de autoria a partir da experiência como leitor de sistemas hipertexto, existe um problema maior a ser tratado nessa atividade; as pessoas não aprenderam como estruturar informações nas redes dos sistemas hipertexto da mesma forma que aprenderam e treinaram a escrever relatórios lineares através de composições escolares. Dessa forma, a partir da experiência como leitor e da compreensão da situação do novo tipo de leitura, ele aconselha três cuidados para os autores de hiperdocumentos:

1. Manter o conteúdo de cada nó focalizado em um único tópico. Desta forma, é mais fácil entendê-lo e também reconhecê-lo em diagramas de visão geral (*overview*), listas de história (*log*), etc. A associação de um determinado tópico a cada nó torna mais fácil para o autor saber os *links* que deve construir. Portanto, essa recomendação direciona o autor a buscar sempre a modularização das informações.

2. Associar aos nós do hiperdocumento volumes de informação menores do que os contidos em artigos em papel; é sabido que a velocidade de leitura em tela de computador é menor do que em papel. A própria natureza de hipertexto ajuda o autor a alcançar este objetivo, pois os tópicos secundários, definições e outros itens não precisam ser elaborados no nó principal. Nós adicionais deveriam ser construídos para conter esse tipo de informação. Portanto, os conteúdos dos nós devem ser criados como **pequenos segmentos de informação**.
3. Estabelecer *links* com parcimônia e critério, evitando adicionar todo *link* possível entre conceitos remotamente relacionados. Trata-se de uma estratégia geral que orienta o autor a tornar a “estrutura clara”, pois assim fica mais fácil para os usuários navegarem sobre ela. Além disso, muitos sistemas hipertexto permitem que os leitores adicionem *links* extras durante a navegação. É importante lembrar que, mesmo em hipertexto, é tarefa do autor estabelecer prioridades para os leitores. Portanto, a autoria de hiperdocumentos deve ser guiada por uma **criteriosa definição dos links**.

Esses são os cuidados que orientam diretamente a atividade de autoria. Entretanto, exigem do autor uma visão completa do hiperdocumento a partir de várias perspectivas que, muitas vezes, é impossível obter sem auxílio de algum procedimento sistemático ou mesmo automatizado.

Este capítulo apresenta a proposta de uma sistemática para exame da estrutura de hiperdocumentos. Com base nessa sistemática, é definida então, uma abordagem de análise relacional de *links*, a partir da qual é proposta uma classificação de reuso de componentes de *links*. Finalmente, é apresentado o formalismo de higrافos, proposto como técnica para representação de estrutura de hiperdocumentos e a seguir, é discutida a relevância da abordagem proposta.

A Seção 4.1 descreve o Ciclo de Vida por Prototipação Evolutiva para o desenvolvimento de hiperdocumentos, que fundamenta a sistemática proposta. Para subsidiar os requisitos da fase de exame é apresentado o ambiente StatSim, que foi utilizado para simular o comportamento dos *links* de um hiperdocumento. A Seção 4.2 apresenta a proposta de uma abordagem orientada a *links* fundamentada na análise

relacional. Buscando coletar as informações relevantes para os critérios de reuso e consistência, são definidos os casos de reuso de componentes de *links*. Tais casos de reuso de *links* são formalmente especificados na Seção 4.3, onde é apresentada uma investigação mais completa das possíveis abstrações derivadas. Na Seção 4.4, os higrافos são propostos como técnica adequada para representação estrutural de hiperdocumentos. Finalmente, a Seção 4.5 conclui este capítulo discutindo a relevância da abordagem proposta.

APU

4.1 Uma sistematização para exame da estruturação de hiperdocumentos

É fato que, além da melhoria da qualidade de projeto de um hiperdocumento, a redução do custo de seu desenvolvimento é um desafio para a indústria de novos sistemas de informação (Isakowitz et al., 1995). Uma maneira de enfrentá-lo é fornecer aos projetistas ambientes de desenvolvimento apropriados.

Segundo Nanard; Nanard (1995), os ambientes de Engenharia de Hipertexto podem ser modelados até certo nível utilizando os ambientes de desenvolvimento de Engenharia de Software. Para o projeto do modelo de dados conceitual do hiperdocumento e do modelo de navegação, por exemplo, é possível a adoção quase direta desses ambientes. Entretanto, sua transposição total é difícil e inadequada, pois existem diferenças fundamentais entre os processos de desenvolvimento de software e de hiperdocumentos. Grande parte do projeto de um hiperdocumento se refere aos aspectos de estética (ou estilo) e de tratamento do conhecimento ao qual os ambientes de Engenharia de Software não dão suporte.

Procurando um ambiente mais adequado para projetar hiperdocumentos, que conduzisse à melhoria da qualidade do projeto e reduzisse custos durante o desenvolvimento, Nanard conclui que esse ambiente deve ser baseado em uma estratégia equilibrada de duas abordagens extremas de desenvolvimento: a *top-down* e

a *bottom-up*. Assim, o projeto de um hiperdocumento é orientado pelas seguintes perspectivas:

1. O uso de técnicas de projeto formais e prescritivas para a produção do projeto;
2. A observação do comportamento das pessoas durante a elaboração do projeto.

Dessa forma, Nanard; Nanard propõem a ligação entre as abordagens *top-down* e *bottom-up* no desenvolvimento de hiperdocumentos de maneira semelhante à abordagem utilizada no Ciclo de Vida de Prototipação de Software, como mostra a Figura 4.1. Nessa figura estão representados os dois sentidos que são considerados para a ligação entre as abordagens *top-down* e *bottom-up*: abstraindo modelos genéricos a partir de exemplos em um sentido e, no outro, instanciando modelos para a implementação e construção de hiperdocumentos.

Segundo Pressman (1994) e Kan (1995), este ciclo de vida compreende a construção de protótipos sob constantes avaliações, com o objetivo de permitir ao projetista o aprimoramento gradual de seus conhecimentos sobre os requisitos do sistema alvo. Um protótipo é uma implementação parcial do produto, onde todas as interfaces externas estão presentes e podem ser utilizadas pelos usuários potenciais. Assim, o protótipo fornece *feedback* para a equipe de desenvolvimento antes que o desenvolvimento prossiga em grande escala. Com sua utilização, os usuários e a equipe de desenvolvimento podem esclarecer e determinar os requisitos e sua interpretação.

Figura 4.1 - Geração de protótipo por meio de abstração e instanciação, segundo Nanard; Nanard (1995)

Nielsen (1995) também reforça a necessidade de constantes avaliações durante o desenvolvimento de um hiperdocumento. Ele menciona que a estrutura geral de um hiperdocumento sofre, muitas vezes, a interferência das definições locais relativas a decisões tomadas durante a construção de cada nó e cada *link* individualmente.

Foi visto no Capítulo 3, que Isakowitz et al. (1995), criadores do método RMM (*Relationship Management Methodology*), salientam a necessidade de prototipação e testes intensivos com o usuário durante o desenvolvimento de aplicações hipermídia, uma vez que a tolerância a erros que o usuário de hipermídia possui é bastante baixa.

Dessa forma, o Ciclo de Vida de Prototipação parece ser uma maneira bastante adequada de se proceder às etapas de desenvolvimento de hiperdocumentos, respeitando essa necessidade de constantes avaliações.

Com base em experiências anteriores no desenvolvimento de sistemas na área de Engenharia de Software, o projeto desenvolvido como parte da pesquisa realizada neste trabalho se baseia em uma abordagem de prototipação para as etapas de desenvolvimento de um hiperdocumento, semelhante ao Ciclo de Vida de Prototipação definido em Engenharia de Software.

Deve-se ressaltar que o processo de desenvolvimento do hiperdocumento tende a ser **uma abordagem evolutiva**, diferentemente do Ciclo de Vida de Prototipação de Software: seus protótipos vão sendo aprimorados e aproveitados nas manutenções subseqüentes. Ao contrário dos protótipos do Ciclo de Vida de Prototipação, que são, geralmente, descartados, sendo que o produto final é posteriormente desenvolvido integralmente. Admitindo-se um Ciclo de Vida Evolutivo para o desenvolvimento de hiperdocumentos, os mecanismos para a avaliação do hiperdocumento durante a atividade de autoria são fundamentais. Os auxílios automatizados para a coleta das medidas para as métricas do capítulo anterior (Seção 3.4) tendem a ser cada vez mais necessários aos autores de hiperdocumentos.

O projeto desenvolvido nesta tese visa fornecer auxílio à atividade de autoria pela identificação de abstrações que possam ser examinadas freqüentemente pelo autor. Além disso, quando necessário, forneça subsídios para uma reorganização do

modelo conceitual abstrato previamente construído. Neste projeto, considera-se que o modelo de Ciclo de Vida para o desenvolvimento de hiperdocumentos por Prototipação Evolutiva possui as seguintes etapas básicas, esquematizadas na Figura 4.2:

1. **Análise de requisitos,**
2. **Autoria,**
3. **Exame e**
4. **Reorganização.**

Figura 4.2 - Etapas do Ciclo de Vida de Prototipação Evolutiva para o desenvolvimento de hiperdocumentos

Esse modelo de Ciclo de Vida por Prototipação de hiperdocumentos, assume que a atividade de avaliação propriamente dita fica a cargo do discernimento humano do autor do hiperdocumento, acompanhada do *feedback* dos leitores.

Cada uma das etapas básicas abrange um conjunto de atividades a serem realizadas durante o desenvolvimento do hiperdocumento. A etapa de **Análise de requisitos** de uma aplicação hipermídia, geralmente o ponto de partida de qualquer desenvolvimento, deve englobar as tarefas relativas a:

- identificação das necessidades de desenvolvimento da aplicação,

- identificação dos objetivos do hiperdocumento,
- análise de mercado e do usuário,
- levantamento das restrições gerais de software e hardware,
- identificação do domínio de informações e de suas fontes,
- obtenção de permissão de uso das informações,
- levantamento e análise dos requisitos de navegação,
- análise de custo e benefício.

Nessa etapa, geralmente é construído um primeiro modelo conceitual das informações que serão alocadas ao hiperdocumento, utilizando algum método para modelagem de informações (como modelo orientado-a-objetos ou modelo entidade-relacionamento) ou alguma organização mental informalmente expressa, que pode ser efetivada pela alocação das informações no sistema de arquivos e diretórios do Sistema ou Ambiente Operacional e geralmente registrada em um documento de requisitos. Com base nessa análise de requisitos, é possível passar para a elaboração de um projeto rápido e a construção de um protótipo, que são as atividades da etapa de **Autoria**. Essa etapa, portanto, compreende a fase de execução das tarefas relativas à:

- organização das informações (elaboração de um projeto) e
- criação do protótipo do hiperdocumento, com base no projeto, por meio da composição dos nós (criação de seus conteúdos e especificação de sua apresentação) e definição de seus *links*.

Para a etapa de Autoria, os métodos apresentados por Isakowitz et al. (1995) e Schwabe et al. (1996) possuem descrições detalhadas das abordagens e respectivas tarefas que devem ser cumpridas durante um procedimento adequado de projeto de hiperdocumento.

“ A etapa de **Reorganização**, por sua vez, também é responsável por disparar as atividades de Autoria. Porém, já conta com a existência de um protótipo já

examinado por leitores e autores. A principal tarefa a ser realizada durante esta etapa é o refinamento do projeto, com base nos resultados obtidos na etapa de exame. Geralmente, esse refinamento é registrado em um documento de requisitos, complementando o documento de requisitos anteriormente elaborado. Além de conter, na forma de especificações, o que deve ser atendido no projeto, também indica as modificações a serem realizadas no protótipo, na forma de refinamentos de implementação.

A etapa denominada **exame**, que fornece o suporte básico para a avaliação do hiperdocumento, é o alvo principal da pesquisa relatada nesta tese. Nessa etapa são incluídas as atividades responsáveis pelas informações que permitem a ligação das duas abordagens *top-down* e *bottom-up*, de Nanard. Na busca das construções abstratas baseadas na estrutura do hiperdocumento (*bottom-up*), as instâncias podem ser validadas com relação ao modelo abstrato pretendido para as mesmas, estabelecido durante o projeto (*top-down*).

Durante a etapa de exame, aborda-se “com detalhes” o produto em construção, com o objetivo de subsidiar a avaliação da qualidade do serviço realizado. O termo **exame** foi escolhido para denominar essa etapa em analogia a seu uso pelos profissionais da área de saúde ao consultarem ou “examinarem” os seus pacientes, entrevistando-os sobre os pontos mais genéricos de queixa. Este tipo de exame não depende de ferramental preciso de laboratório ou equipamentos sofisticados para coleta de dados de exame específico.

Para a definição das medidas relevantes a serem coletadas para essa etapa de exame, há a necessidade de se estabelecer o escopo e as características das informações que podem ser utilizadas como dados de entrada. Seguindo o modelo de referência Dexter para sistemas hipertexto (Seção 2.2), os hiperdocumentos podem ser examinados sob dois ângulos de granularidade distintos: de conteúdo e estrutural¹.

¹ O modelo Dexter também define a camada de apresentação de sistemas hipertexto, que se refere às formas resultantes de exibição do hiperdocumento em tempo-de-execução, mas essa camada está fora do escopo de avaliação proposto neste trabalho, cuja abordagem é baseada em *links*.

O nível de conteúdo, que é chamado no modelo Dexter de camada internos-componentes, engloba o conteúdo dos nós e o seu intra-relacionamento como, por exemplo, parágrafos e títulos. No nível estrutural, correspondente à camada de armazenamento no modelo Dexter, está definida a Base de Dados que define a rede de nós e *links*, ou seja, os inter-relacionamentos entre nós e *links*.

Como o exame dos conteúdos dos nós é uma atividade bastante relacionada com o processo de escrita, as ferramentas de apoio e verificação de textos são as adequadas para subsidiar sua avaliação. Por exemplo, os trabalhos de Salton et al. (1994), vistos no capítulo anterior (Seção 3.1), são também relevantes na área de avaliação do conteúdo de documentos pois concentram-se na medição do grau de similaridade entre nós, visando a definição de *links* automaticamente.

Nesta tese, o enfoque do exame é dado para a avaliação sobre a estruturação de um hiperdocumento. Para isso, a estratégia de simulação é uma das mais convenientes, pois independe da intervenção do usuário para ativar os diferentes *links* do hiperdocumento: esse controle pode ser programado e fica a cargo do simulador. Na próxima seção é apresentado um ambiente originalmente desenvolvido para apoiar a Engenharia de Software na fase de especificação do comportamento de sistemas reativos em *Statechart*, mas que, com modificações realizadas neste trabalho, pode ser adaptado para auxiliar a fase de exame de hiperdocumentos.

4.1.1 O StatSim como ferramenta de auxílio ao exame de hiperdocumentos

A maneira mais simples de representação de hiperdocumento é através de dígrafo. As representações em *Statechart* podem ser tratadas como dígrafos, uma vez que *Statechart* constitui uma linguagem de especificação considerada uma extensão de DTE. O StatSim de Masiero et al. (1991) é um software de apoio à especificação de *Statechart*, que conta com um simulador de *Statechart* desenvolvido por esta autora (Fortes, 1991). Por essa razão, durante a fase de exame, foi utilizado o StatSim, procurando monitorar a dinâmica dos *links* definidos no hiperdocumento.

Para utilizar o StatSim com esse propósito, foi necessário alterar o módulo “Analisador LES” — módulo que interpreta especificações *Statechart* em LES para gerar uma Base de Dados dos estados e transições —, adaptando-o para dar suporte à etapa de exame de hiperdocumento. LES é uma linguagem (Linguagem de Especificação de *Statechart*) que foi concebida por Fortes; Masiero (1991) para dar suporte à descrição de diagramas *Statechart* de forma textual. A gramática da linguagem LES na notação BNF se encontra descrita no apêndice A.

A adaptação feita principalmente no módulo “Analisador LES” constituiu-se de modificações na implementação do StatSim para lidar com maior volume de dados e customização da interface para acomodar o novo tipo de dado.

O StatSim é um ambiente gráfico automatizado cujo principal objetivo é auxiliar a validação de especificações em *Statechart*. A Figura 4.3 mostra sua arquitetura.

Figura 4.3 - Arquitetura do ambiente StatSim.

Na maioria dos casos, a especificação de um *Statechart* e sua simulação visual satisfazem o principal interesse por parte dos projetistas de sistemas reativos. No entanto, o “Analisador LES” é sempre utilizado pelo pacote EGS (Editor Gráfico de *Statechart*) de Batista (1991), para fornecer os algoritmos para compilação das descrições de transições em LES. Com a adaptação do “Analisador LES”, a interface de uma simulação textual no StatSim tem a forma da Figura 4.4.

Figura 4.4- A tela do módulo “Analisador LES”, ativada para simulação textual de uma especificação em LES

A simulação evidenciou uma grande quantidade de valores para as âncoras em contraste com a quantidade existente dos nós-destino para os quais elas rotulavam. Com essa percepção foi iniciada uma investigação sobre os relacionamentos de reuso dos componentes de *links*. Para dar suporte a essa investigação, uma arquitetura para exame de estrutura de hiperdocumentos foi definida, apresentada na próxima seção.

4.1.2 Proposta de uma arquitetura para exame da estrutura de hiperdocumentos

O ponto inicial para o exame da estrutura de um hiperdocumento está em sua codificação, que é feita mais naturalmente por um dígrafo (grafo direcionado) rotulado.

Uma arquitetura de suporte à etapa de exame da estruturação dos *links*, descrita na forma de um dígrafo, foi definida e é esquematizada na Figura 4.5. Foi indentificada a necessidade de duas abordagens, denominadas recursiva e relacional, as quais são representadas pelos processos denominados de Análise Recursiva e de Análise Relacional nesta figura.

Figura 4.5 - As abordagens de análise para exame da estrutura de um hiperdocumento.

A **Análise Recursiva** considera a estrutura do hiperdocumento como um grafo, em que cada nó corresponde a um vértice e cada *link* a um arco do grafo. Com a análise recursiva é possível examinar os critérios que foram apontados por

Hatzimanikatis et al. (1995), apresentados no capítulo anterior (Seção 3.4), e transcritos a seguir:

- **complexidade dos caminhos;**
- **modularidade;**
- **impureza da árvore; e**
- **complexidade individual do nó.**

De fato, essa análise remete para as métricas propostas pela maioria dos trabalhos relacionados com avaliação vistos no Capítulo 3.

É importante notar que as propostas de exame de hiperdocumento existentes fazem uso apenas da Análise Recursiva para a extração de resultados. Dessa forma, é proposta deste trabalho a ampliação da abordagem para exame de hiperdocumentos por Análise Recursiva, utilizando a Análise Relacional, de forma a complementar e expandir/refinar as informações obtidas.

A **Análise Relacional** considera a estrutura do hiperdocumento como uma coleção de *links*, os quais possuem uma semântica própria, mas que podem ser comparados com os demais. Com a Análise Relacional é possível examinar prioritariamente alguns dos critérios de Garzotto et al. (1995) e descritos no capítulo anterior, a saber:

- **consistência;**
- **modularidade e**
- **reuso.**

A Figura 4.5 mostra que tanto a abordagem de Análise Recursiva como a de Análise Relacional fazem uso do hiperdocumento representado por um grafo. A Análise Relacional, entretanto, constrói, a partir dessa representação, uma Base de Dados Relacional de *links*, a partir da qual o hiperdocumento poderá ser tratado como um higrfo (que são definidos na Seção 4.4), sobre o qual as operações de “degeneração” e “agregação” viabilizam esse novo enfoque para representação de

hiperdocumentos. Essas operações, que identificam abstrações, são apresentadas na Seção 4.3.

4.2 Uma proposta de classificação de reuso de estrutura

Com o objetivo de dar suporte à abordagem de Ciclo de Vida de Prototipação Evolutiva de desenvolvimento de hiperdocumentos, este trabalho propõe uma classificação para o reuso de componentes de *links*. Esta proposta visa auxiliar a fase de exame, mais especificamente em relação à análise relacional. Para tanto, casos de reuso de nós e *links* foram então definidos, que conduzem a avaliações genéricas, indicando possíveis inconsistências ou informações a serem investigadas de forma mais geral, do ponto de vista do estilo retórico adotado no hiperdocumento como um todo.

A estrutura que um *link* define como inter-relacionamento entre os nós de um hiperdocumento pode ser representada por uma transição de DTE. Quando o *link* é unidirecional, pode ser descrita textualmente, pelos componentes de DTE da seguinte maneira:

transição:

evento	nó-fonte. âncora-fonte
origem	nó-fonte
destino	nó-destino. âncora-destino

No caso do *link* bidirecional, o **botão de controle** é mapeado no nó corrente da apresentação, podendo estar ora no nó-fonte, ora no nó-destino do *link*. Somente em tempo de navegação é que o DTE que representa o hiperdocumento informa ao sistema qual é o estado-atual, estado em que a página correntemente apresentada recebe a interação do usuário sobre suas âncoras disponíveis e habilitadas.

O mapeamento genérico dos componentes de um hiperdocumento em componentes de DTE pode ser feito como mostra a Tabela 4.1.

Tabela 4.1 - Componentes estruturais genéricos de um hiperdocumento e seus correspondentes componentes de DTE

HIPERDOCUMENTO	DTE	situação	
nó-fonte	estado	corrente	próximo
nó-destino	estado	próximo	corrente
âncora-fonte		botão de controle	
link	transição		
âncora-destino			botão de controle
botão de controle	evento		

Dessa forma, os *links* bidirecionais indicam que as âncoras podem estar no nó-fonte ou nó-destino e que o **botão de controle** pode estar ora no nó-fonte, ora no destino, dependendo do estado corrente do leitor durante a navegação. Para a representação em DTE é necessário, portanto, que a transição contenha os atributos que identificam totalmente o relacionamento entre os nós conectados pelo *link*, como mostra a Figura 4.6. Um importante aspecto de implementação dos *links* bidirecionais é o suporte requerido para a navegação por eles, que requer a função adicional de gerenciar o estado-corrente.

Figura 4.6 - DTE representando um *link* bidirecional de hiperdocumento.

Observa-se que uma linguagem que permita as descrições dos *links* bidirecionais deve incluir a informação que é válida para ambas as direções uma única vez, ou seja, os terminais do *link*. Logo, uma descrição textual de transição bidirecional, em termos dos componentes de DTE, poderia ser:

link:

terminal1	nó-destino. âncora-destino
terminal2	nó-fonte. âncora-fonte

transição1:

evento	âncora-destino
origem	terminal1
destino	terminal2

transição2:

evento	âncora-fonte
origem	terminal2
destino	terminal1

Essa descrição textual do *link* possui os descritores “terminal1” e “terminal2” como especificadores dos terminais do *link*, independentemente da direção. Dessa forma, a definição de um *link* requer que duas transições sejam adicionadas, uma para cada direção do mesmo.

Embora os *links* bidirecionais sejam passíveis de descrição, conforme discutido, a proposta de análise relacional de *links* adota o mecanismo de *links* unidirecionais para a definição da classificação a ser apresentada. Este é o mecanismo mais comumente adotado nos sistemas atuais. Quando existirem suportes melhores para a definição de *links* bidirecionais, os aspectos a sofrerem interferência em relação aos casos de reuso de componentes de *links* apresentados são os seguintes:

- a ancoragem no nó-destino como um elemento de controle que também poderá ser comparado por reuso;
- a apresentação explícita de ambas as âncoras (fonte e destino) para interação, i.e., o caráter sugestivo de navegação pelo *link*, desempenhado pela âncora, passa a ser um elemento mais decisivo durante a autoria, pois tem que rotular a direção inversa e, conseqüentemente, requer também avaliação.

Neste trabalho, supondo os *links* unidirecionais, foram então definidos os casos de reuso. Essa definição se fundamenta na análise do reuso envolvendo as três variáveis correspondentes aos componentes dos *links* unidirecionais: nó-fonte, âncora-fonte e nó-destino. Para cada uma dessas variáveis pode ser verificada a característica de reuso em relação aos respectivos componentes dos outros *links*.

Portanto, o atributo de reuso para cada uma dessas variáveis assume valor binário (existe reuso ou não). Dessa forma, a definição dos casos de reuso de componentes de *links* se estabelece em um universo finito contendo 2^3 elementos, os quais vão do Caso 0 $(000)_2$ ao Caso 7 $(111)_2$.

Figura 4.7 - Árvore de geração dos casos de reuso de componentes de *links*.

Para a geração de um esquema de ocorrência desses casos, é proposto que a cada reuso do componente correspondente do *link* (nó-fonte/âncora-fonte/nó-destino) seja atribuído valor “verdadeiro” ou 1 (significando que existe reuso) para o atributo que identifica o componente correspondente na estrutura do *link*. Quando não há reuso, o valor atribuído é “falso” ou 0 (significando que inexistente reuso). Portanto, o caso dos *links* que possuem mesmos nós-fonte (1, ou seja, existe reuso), diferentes âncoras-fonte (0, ou seja, inexistência de reuso) e diferentes nós-destino (0) foi identificado como *Caso 4*, i.e., 100 em base binária. Dessa forma, o mapeamento desses casos é definido como o conjunto dos elementos folha gerados pela árvore da Figura 4.7.

A Figura 4.8 mostra como os diferentes casos podem então ser semanticamente esquematizados. Nessa figura os retângulos arredondados

representam os nós, as setas são os *links* e os rótulos “x” e “y” correspondem aos valores das âncoras-fonte.

Figura 4.8 - Esquemas de casos de reuso de componentes de *link*

A semântica desses casos permite uma classificação para os relacionamentos entre os *links* contidos em cada caso. A Figura 4.9 mostra três tipos de classe identificadas:

1. Classe de **uso-geral** - Casos 3 e 4, que podem ser interpretados, respectivamente, como os casos que representam páginas bastante referenciadas (com muitos *links* de chegada) e páginas índices (com muitos *links* de saída). Esses casos revelam certa similaridade com a escrita impressa em papel;.
2. Classe de **uso-contextual** - Casos 1, 2, 5 e 6, onde, a partir de uma interpretação somente estrutural pode-se suspeitar da existência de possíveis anomalias. É necessário que, na análise desses casos, o contexto do *link* seja considerado.
3. Classe de **uso-particular** - Casos 0 e 7, que sugerem a necessidade de uma análise mais detalhada, pois correspondem aos casos extremos de reuso.

Figura 4.9 - Classes de casos de *links* considerando-se o reuso de seus componentes

A seguir, são discutidas algumas avaliações para cada uma das três classes de *links* ilustradas.

4.2.1 Classe de *links* de uso-geral

A avaliação dos *links* estabelecidos na classe de uso-geral da Figura 4.9(b) requer uma análise de caráter geral seguindo as regras de avaliação para o estilo retórico de documentos impressos. Ficam a cargo do autor alternativas de avaliação para, por exemplo, pequenos ajustes na padronização dos nomes das âncoras-fonte, tais como uso de maiúsculas ou minúsculas, siglas e verbos. As métricas relativas aos critérios de modularidade e de complexidade individual do nó, vistas no capítulo anterior (Seção 3.4), também podem ser aplicadas aos elementos pertencentes a essa classe de uso-geral por considerarem as medidas de *fanin* e *fanout*. Essa classe inclui os Casos 3 e 4. As seguintes considerações de avaliação podem ser feitas para os casos identificados nessa classe:

Caso 3 (011)₂ : **fontes diferentes, âncoras-fontes iguais, destinos iguais** — muitas referências iguais conduzem à mesma informação. Neste caso, os nós-fonte distintos poderiam ser agrupados em um “super” nó-fonte, a partir do qual a mesma âncora-fonte leva ao nó-destino. Uma questão que surge é quanto ao relacionamento entre os nós-fontes; se houver algum *link*/caminho entre eles, então pode ser que o mesmo *link* seja excessivo. A questão típica a ser respondida é:

— *Quais links que, partindo de nós-fonte distintos, possuem mesma âncora-fonte e chegam no mesmo nó-destino?*

Caso 4 (100)₂ : **fontes iguais, âncoras-fontes diferentes, destinos diferentes** — este caso não pode ser analisado como uma anomalia, a não ser sob a visão do próprio conteúdo dos nós, uma vez que não existe qualquer reutilização de *links*. Do ponto de vista do autor, ele deve tomar ciência da quantidade de *links* desse tipo, pois pode ser indicativo de uma baixa taxa de reutilização de objetos. Esse padrão de *links* permite que sejam respondidas questões do tipo:

— *Quais links que, partindo do mesmo nó-fonte, possuem âncoras-fonte distintas e chegam em diferentes nós-destino?*

4.2.2 Classe de *links* de uso-particular

Nessa classe de *links*, denominada de uso-particular, estão aquelas estruturas que desempenham os papéis ilustrados pelos Casos 0 e 7 da Figura 4.9(a). Sua análise conduz a avaliações bastante particulares ao autor de cada nó e *link* pertencentes a essa classe.

Caso 0 (000)₂ : fontes diferentes, âncoras-fontes diferentes, destinos diferentes —

representa a diversidade de definição de *links* e pode, portanto, servir como uma medida de criatividade expressa em um hiperdocumento. Essa medida de criatividade, no entanto, deve ser analisada sob dois aspectos. Por refletir o volume de diferentes informações e de acessos disponíveis no hiperdocumento, sua incidência pode ser uma medida que auxilia a métrica de tamanho do hiperdocumento. Geralmente, a ocorrência de *links* no caso 0 coincide com o conjunto de *links* da fronteira do hiperdocumento, uma vez que os nós-fonte não são reutilizados como nós-fonte em quaisquer outros *links* e os nós-destino não desempenham esse papel nos demais *links* do hiperdocumento. Pode-se verificar o estilo de redação e de apresentação dos nós-fonte e nós-destino. A questão típica a ser analisada é:

— *Quais links que, partindo de diferentes nós-fonte, possuem âncoras-fonte distintas e chegam em diferentes nós-destino?*

Ou, simplificadaamente:

— *Quais links não possuem qualquer reuso de estrutura em relação aos demais?*

Caso 7 (111)₂ : fontes iguais, âncoras-fontes iguais, destinos iguais — embora este

caso corresponda à relação de identidade entre *links* (por possuir os mesmos componentes), trata-se de *links* diferentes. Esse caso ocorre quando as âncoras-fonte estão posicionadas em lugares distintos ao longo do conteúdo

do nó-fonte e indicam um típico reuso do *link*. Do ponto de vista de uniformidade de uso do termo associado à âncora-fonte, este reuso é cabível. Porém, o seu uso em excesso é um possível indicativo de necessidade de uma análise local dos nós envolvidos.

Geralmente, ocorre consistentemente quando o nó-fonte está com um volume de informações grande e é necessário o uso do mesmo *link* em um outro ponto distante ao uso anterior. Assim, cabe somente ao autor avaliar este caso criteriosamente com relação ao volume de conteúdo do nó-fonte, pois a página em que seu conteúdo será apresentado possivelmente requererá intervenção do usuário, pois não caberá em uma só janela. Quando o reuso não é feito em um ponto distante, esse *link* pode ser excessivo. A questão típica a ser analisada é:

— *Quais links que, partindo do mesmo nó-fonte, possuem mesmas âncoras-fonte e chegam no mesmo nó-destino?*

4.2.3 Classe de *links* de uso-contextual

Nesta classe de uso-contextual estão os casos de *links* que possuem os esquemas representacionais vistos nos Casos 1, 2, 5 e 6 na Figura 4.9(c). Tais casos conduzem a interpretações passíveis de ambigüidades ou falta de uniformidade no estilo e terminologia empregados no hiperdocumento.

A característica principal dessa classe de *links* é que, apenas pela descrição dos *links*, pode-se identificar contradições. As perguntas sobre a existência desses *links* são do seguinte tipo:

-
- *Existem vários links “ancorados” da mesma forma mas levando a informações diferentes?*
 - *Existem links “ancorados” diferentemente e que levam à mesma informação?*
-

As respostas a essas interrogações possibilitam a classificação desses *links* como de uso-contextual.

Em geral, quando bem utilizados e definidos, esses *links* são de grande importância e evidenciam uma visão refinada durante o processo de autoria do hiperdocumento. Portanto, sua análise não pode se basear simplesmente nos esquemas que os representam, mas deve considerar também o “contexto” em que foram estabelecidos. O termo “contexto” é empregado nesta tese para referenciar a porção de texto de um documento que circunda a âncora, ou seja, que fica próxima à âncora, durante a sua apresentação para o leitor.

Distinção deve ser feita com relação à classe de **nós de contexto** definida no modelo MCA, por Casanova et al. (1991) e Soares et al. (1993b), cuja utilização visa o agrupamento recursivo de nós de contexto ou de nós terminais, para a organização hierárquica ou não do conjunto de nós. O conceito de nós de contexto do MCA corresponde a um mecanismo genérico para prover os requisitos da modelagem de uma abstração que o autor pode construir no seu hiperdocumento e situa-se em um nível mais alto do que o **contexto de nó** tratado em nível da âncora aqui discutido. A seguir, são apresentadas algumas considerações de avaliação para os casos identificados nessa classe de uso-contextual.

Caso 1 (001)₂ : **fontes diferentes, âncoras-fontes diferentes, destinos iguais** — caso típico de liberdade na nomeação dos *links* durante a autoria e, que pode se refletir em uma possível desorientação do usuário, pois, a partir de nomes de *links* diferentes, ele vai obter a mesma informação. Uma justificativa possível para essa ocorrência é de que possa ter havido uma falha de uniformização dos valores de suas âncoras-fonte. Assim, casos como esse parecem ser indesejáveis.

Entretanto, considerando-se o contexto da âncora-fonte, pode ser que o mesmo nó-destino tenha sido planejado para ser referenciado por âncoras-fonte distintas. Por exemplo, quando a ida ao nó-destino serve para observação de detalhes diferentes sobre o mesmo conteúdo do nó-destino.

A questão típica a ser analisada é:

— *Quais links que, partindo de nós-fonte distintos, possuem âncoras-fonte diferentes e chegam no mesmo nó-destino?*

Caso 2 (010)₂ : fontes diferentes, âncoras-fontes iguais, destinos diferentes — à primeira vista, este é um caso típico de inconsistência, pois o mesmo valor de âncora indicando o *link* leva a obtenção de informações diferentes, dependendo do nó-fonte em que estiver inserida.

Entretanto, considerando-se o contexto da âncora-fonte, a reutilização da âncora-fonte pode indicar uma classe de nós-destino. Uma degeneração ainda possível é o caso em que o nó-destino com uma âncora-fonte ‘x’ é o nó-fonte que possui também a âncora-fonte ‘x’ levando para outro nó-destino. Este caso pode provocar desorientação, exceto no caso de âncoras-fonte do tipo “continue na próxima página”.

A questão típica a ser analisada é:

— *Quais links que, partindo de nós-fonte distintos, possuem mesmas âncoras-fonte e chegam em nós-destino diferentes?*

Caso 5 (101)₂ : fontes iguais, âncoras-fontes diferentes, destinos iguais — essa prática pode aborrecer o leitor. Não é desejável para o leitor que o acionamento de âncoras-fonte distintas levem à mesma informação, pois pode deixá-lo confuso. Essa situação seria avaliada, estruturalmente, como um erro na nomeação dos *links*.

Entretanto, considerando-se o contexto da âncora-fonte, pode ser que o mesmo nó-destino tenha sido planejado dessa forma. Por exemplo, quando se quer utilizar o mesmo nó-destino com um conteúdo de “ilustração”, com o objetivo de reutilizar seu conteúdo, seja para “reforçar” o conhecimento sobre seu conteúdo sob enfoques distintos, ou mesmo observar diferentes detalhes a respeito daquele mesmo conteúdo. A prática desses *links* pode demonstrar

uma preocupação do autor em não aumentar a carga cognitiva com novos conteúdos, mas sim realçar diferentes enfoques.

A questão típica a ser analisada é:

— *Quais links que, partindo do mesmo nó-fonte, possuem âncoras-fonte distintas e chegam no mesmo nó-destino?*

Caso 6 (110)₂: fontes iguais, âncoras-fontes iguais, destinos diferentes — quando a partir de um nó-fonte com duas âncoras de mesmo conteúdo disparam *links* para diferentes informações, numa primeira análise, seria indicativo de uma possível inconsistência. A expectativa de leitura sugere que haja coerência das informações, mesmo durante os diferentes momentos de navegação pelos *links*.

Entretanto, considerando-se o contexto da âncora-fonte, esse caso pode ser bem empregado, demonstrando uma uniformidade na nomenclatura. Por exemplo, uma âncora “Exemplo” que, aplicada a diferentes regiões no texto do nó-fonte, direciona a nós-destino distintos com exemplos relacionados ao contexto de onde a âncora no nó-fonte possui seu ponto de partida. Isto pode indicar que o tipo de *link* com âncoras iguais é um desvio para uma classe de nós.

A questão típica a ser analisada é:

— *Quais links que, partindo do mesmo nó-fonte, possuem mesmas âncoras-fonte e chegam em nós-destino distintos?*

4.3 Especificação formal da classificação dos *links*

Apesar de não serem implementadas nos hiperdocumentos existentes atualmente, as estruturas virtuais são apontadas por Halasz (1988) como um mecanismo poderoso a ser tratado pelas próximas gerações de sistemas hipertexto. Segundo Halasz, a noção de estruturas virtuais para hipermídia consiste em uma adaptação direta do conceito de visões (tabelas virtuais) da área de Bases de Dados Relacionais. De fato, tais estruturas requerem um suporte substancial de busca e consulta (*search & query*) sobre a rede de informações mantida no hiperdocumento. Apesar de resultarem de consultas, os componentes das estruturas virtuais, por definição, impõem que tais estruturas sejam construídas dinamicamente como entidades do hiperdocumento e não sejam simplesmente uma apresentação de resultados das consultas em tela.

Para se ter uma visão mais genérica dos casos de reuso de componentes de *links* apresentados na seção anterior, foi feito um estudo das abstrações possíveis, com dois objetivos:

1. Generalizar o procedimento de classificação dos *links* e,
2. Fornecer uma visão das alternativas para construção de estruturas virtuais que tenham sido definidas durante a autoria de forma implícita.

Duas classes de abstrações são então definidas, a saber:

- A primeira classe de casos derivados inclui os que foram criados a partir dos possíveis agrupamentos de nós-destino. Foram denominados de **agregações**.
- A segunda classe de casos inclui os que foram obtidos a partir da possibilidade dos *links* serem locais a um determinado nó, ou seja, quando os nós-fonte e nós-destino coincidirem. Foram denominados de **degenerações**. Por exemplo, se o conjunto de nós N definido segundo a representação de Parunak (1991) (apresentada na Seção 2.3.3) fosse em última instância um conjunto unitário, seria uma degeneração, representando um hiperdocumento provavelmente incomum.

São também denominadas degenerações os casos em que um dos *links* com componentes de reuso é totalmente local.

A partir da visão completa das possibilidades de todos esses casos de reuso de componentes de *links*, uma descrição formal das regras de definição de cada um dos casos é apresentada a seguir.

4.3.1 Degenerações e agregações derivadas

Uma análise do reuso dos nós-fonte permite identificar os casos 0, 1, 2 e 3 como os casos **primitivos**. Esses casos primitivos possuem suas âncoras-fonte sempre partindo de nós-fonte distintos. A partir desses casos primitivos são derivados os demais casos, usando degeneração (identificados com a letra **d**), agregação (identificados com a letra **g**), ou ambas, como mostra a Tabela 4.2.

Para manter a coerência na notação, foi decidido que, depois que uma abstração é construída, o último dígito da notação adotada para rotular os casos passa a representar ‘âncoras-destino’, ao invés de ‘nós-destino’. É o que acontece, por exemplo, quando da abstração do caso 0, obtendo o caso 0g da Figura 4.10.

Tabela 4.2 - Casos primitivos e as respectivas derivações por agregação ou degeneração

<i>casos primitivos</i>	<i>casos com agregação</i>	<i>casos com degeneração</i>	<i>casos com agregação/ degeneração</i>
Caso 0	0g, 4, 4g	0d, 4d, 0dd	0gd, 4gd
Caso 1	5	5d	-
Caso 2	2g, 6, 6g	2d, 6d, 2dd	2gd, 6gd
Caso 3	7	7d	-

Seja tomado inicialmente o caso 0, que é o que define os *links* que possuem fontes diferentes, âncoras-fonte diferentes, destinos diferentes. Esses *links*, como já foi observado, não possuem qualquer reuso de componentes de *links*. Ao se estudar

as suas possíveis derivações por agregação e ou degeneração, o caso 4 foi identificado.

Os *links* dos casos 0 e 4 são caracterizados por possuírem nomes de âncoras-fonte diferentes e âncoras-destino também diferentes. Em outras palavras, esses *links* são independentes e possuem seus pontos terminais distintos, ou seja, o conteúdo de indicação de *link* (âncora-fonte) e o ponto de chegada são distintos. A Figura 4.10 mostra os agrupamentos que resultam nos casos derivados do caso 0. O que diferencia o caso 0 do 4 é o fato das âncoras-fonte estarem em nós distintos ou não. O caso 4 representa uma agregação dos nós-fonte em apenas um nó; no Caso 0, os nós-fonte são distintos.

Existem ainda duas possíveis agregações, a primeira agrupa os nós-destino como no caso 0g. A segunda agrupa ambos, nós-fonte e nós-destino, cujo esquema é apresentado pelo caso 4g.

Com relação às degenerações dos casos 0 e 4, pode-se ter as seguintes possibilidades:

- i. os nós-fonte são distintos, como no caso 0, porém os *links* são destinados a pontos distintos alocados em âncoras-destino pertencentes aos respectivos nós-fonte. Esse caso está esquematizado na Figura 4.10 como caso 0dd.
- ii. as âncoras-fonte estão no mesmo nó-fonte, mas os *links* direcionam para uma âncora em nó-destino distinto e para uma âncora-destino alocada no mesmo nó-fonte. Esse caso é esquematizado na Figura 4.10 pelo caso 4d. Deve-se notar que esta é uma degeneração do caso 4, uma vez que é assumido que um dos *links* possui seus pontos terminais localmente distribuídos em um único nó.
- iii. as âncoras-fonte estão no mesmo nó-fonte, como ocorre no caso 4, mas os *links* são destinados a pontos distintos alocados em âncoras-destino pertencentes ao próprio nó-fonte, como esquematizado no caso 4gd, na Figura 4.10. Deve-se notar que esta é uma degeneração do caso 4g, que assume que o hiperdocumento é composto por um único nó.

- iv. os nós-fonte são distintos, como no caso 0, porém os *links* são destinados a pontos distintos alocados em âncoras-destino pertencentes ao mesmo nó que é fonte de um dos *links*, caso este esquematizado por 0gd na Figura 4.10. Deve-se notar que esta é uma degeneração do caso 0g, uma vez que é assumido que um dos *links* possui seus pontos terminais localmente distribuídos em um único nó.

Figura 4.10 - Casos derivados do caso 0.

- v. os nós-fonte são distintos, como o caso 0, mas os *links* direcionam para uma âncora em nó-destino distinto e para uma âncora-destino alocada no mesmo nó-

fonte, como no caso 0d, na Figura 4.10. Deve-se notar que esta é uma degeneração do caso 0, uma vez que é assumido que um dos *links* possui seus pontos terminais localmente distribuídos em um único nó.

O caso 1 primitivo define os *links* com fontes diferentes, âncoras-fonte diferentes, destinos iguais. Esses *links* são caracterizados por possuírem valores de âncoras-fonte distintos e âncoras-destino iguais e são, portanto, relacionados ao caso 5. Ou seja, a característica desses *links* é que eles possuem o ponto de indicação de *link* (âncora-fonte), i.e., o conteúdo apresentado como a informação de interação para o leitor navegar, com conteúdo diverso. Porém, eles possuem o mesmo ponto de chegada do *link*. A Figura 4.11 mostra os agrupamentos que resultam nos casos derivados deste caso.

O que diferencia o caso 1 do caso 5 é o fato das âncoras-fonte estarem em nós distintos ou não. A degeneração possível é o caso extremo em que o hiperdocumento é composto por um único nó (caso 1d, Figura 4.11). Pode-se observar que o caso 5 representa uma agregação dos nós-fonte em um único nó; no caso 1, os nós-fonte são distintos.

Figura 4.11 - Casos derivados do caso 1.

O caso 2 primitivo define os *links* com fontes diferentes, âncoras-fonte iguais, destinos diferentes. A Figura 4.12 mostra os agrupamentos que resultam nos casos derivados do caso 2. A característica básica desses *links* é que a partir do mesmo valor de âncora-fonte, os *links* chegam em âncoras-destino distintas. O caso 6 define os *links* que possuem fontes iguais, âncoras-fonte iguais, destinos diferentes.

O que diferencia o caso 2 do caso 6 é o fato das âncoras-fonte estarem em nós distintos ou não. O caso 6 representa uma agregação dos nós-fonte em apenas um nó; no caso 2, os nós-fonte são distintos.

Existem ainda duas possíveis agregações, a primeira agrupa os nós-destino como no caso 2g. A segunda agrupa ambos, nós-fonte e nós-destino, como no caso 6g.

Há as seguintes possibilidades de degenerações dos casos 2 e caso 6:

- i. os nós-fonte são distintos, como no caso 2, porém os *links* são destinados a pontos distintos alocados em âncoras-destino pertencentes aos respectivos nós-fonte, como no caso 2dd da Figura 4.12.
- ii. as âncoras-fonte estão no mesmo nó-fonte, mas os *links* direcionam para uma âncora em nó-destino distinto e para uma âncora-destino alocada no mesmo nó-fonte, como esquematizado no caso 6d na Figura 4.12. Deve-se notar que esta é uma degeneração do caso 6, uma vez que é assumido que um dos *links* possui seus pontos terminais localmente distribuídos em um único nó.
- iii. as âncoras-fonte estão no mesmo nó-fonte, como ocorrem no caso 6, mas os *links* são destinados a pontos distintos alocados em âncoras-destino pertencentes ao mesmo nó-fonte. Dessa forma, tem-se o caso 6gd. Deve-se notar que esta é uma degeneração do caso 6g, uma vez que é assumido que o hiperdocumento é composto por um único nó.
- iv. os nós-fonte são distintos, como no caso 2, porém os *links* são destinados a pontos distintos alocados em âncoras-destino pertencentes ao mesmo nó que é fonte de um deles (caso 2gd). Deve-se notar que esta é uma degeneração do caso

2g, uma vez que é assumido que um dos *links* possui seus pontos terminais localmente distribuídos em um único nó.

- v. os nós-fonte são distintos, como o caso 2, mas os *links* direcionam para uma âncora em nó-destino distinto e para uma âncora-destino alocada no mesmo nó-fonte, tem-se o caso esquematizado em caso 2d. Deve-se notar que esta é uma degeneração do caso 2, uma vez que é assumido que um dos *links* possui seus pontos terminais localmente distribuídos em um único nó.

Figura 4.12 - Casos derivados do caso 2.

O caso 3 primitivo define os *links* que possuem fontes diferentes, âncoras-fonte iguais e destinos iguais. A Figura 4.13 mostra os agrupamentos que resultam nos casos derivados do caso 3. Esses *links* são caracterizados por possuírem nomes de âncoras-fonte iguais e âncoras-destino iguais. O caso 7 define os *links* com fontes iguais, âncoras-fonte iguais, destinos iguais.

Em outras palavras, eles possuem o ponto de indicação de *link* (âncora-fonte), apresentado como uma informação de interação para o leitor navegar, com mesmo conteúdo, mas possuem o mesmo ponto de chegada do *link*.

Figura 4.13 - Casos derivados do caso 3.

O que diferencia o caso 3 do caso 7 é o fato das âncoras-fonte estarem em nós distintos ou não. A degeneração possível é quando o hiperdocumento é composto por um único nó (caso 7d). Pode-se observar que o caso 7 representa uma agregação dos nós-fonte em apenas um nó; no caso 3, os nós-fonte são distintos.

4.3.2 Formalização dos casos de *links*

Um hiperdocumento consiste de uma três-tupla ordenada $H = \langle N, A, L \rangle$ de conjuntos de nós, âncoras e *links* (Parunak,1991). Com o objetivo de formalizar os esquemas de casos de reuso de componentes de *links* utilizando a definição de Parunak, a notação adotada nessa definição foi estendida com a introdução de uma *função extratora* e de uma *função de atributo*:

1. *nó-fonte*: $L \rightarrow A \mid \text{nó-fonte}(\langle A_{ij}, _ \rangle) = A_{ij}$: função que extrai do *link* o nó-fonte.
2. *nome*: $A \rightarrow \langle \text{string} \rangle \mid \langle \text{ícone} \rangle$: função atributo denominada *nome*, que mapeia a âncora ao valor apresentado durante a navegação.

A partir dessas extensões, pode-se descrever formalmente os oito casos de reuso de componentes de *links* como:

Caso 0

- a) $\{ \langle \{N_i, N_j\}, \{A_i, A_j\}, \{N_r, N_t\} \rangle \mid \text{nome}(\text{nó-fonte}(\langle A_{ik}, A_{rs} \rangle)) \neq \text{nome}(\text{nó-fonte}(\langle A_{jl}, A_{tx} \rangle))$
para $1 \leq i, j, r, t \leq m, 1 \leq k \leq |A_i|, 1 \leq s \leq |A_r|, 1 \leq x \leq |A_t|, r \neq t$ e $1 \leq l \leq |A_j| \}$
 - b) $\{ \langle \{N_i, N_j\}, \{A_i, A_j\}, \{N_r, N_t\} \rangle \mid \text{nome}(\text{nó-fonte}(\langle A_{ik}, A_{rs} \rangle)) \neq \text{nome}(\text{nó-fonte}(\langle A_{jl}, A_{tx} \rangle))$
para $1 \leq i, j, r, t \leq m, 1 \leq k \leq |A_i|, 1 \leq s \leq |A_r|, 1 \leq x \leq |A_t|, r = t, s \neq x$ e $1 \leq l \leq |A_j| \}$
-

Caso 1

- a) $\{ \langle \{N_i, N_j\}, \{A_i, A_j\}, \{N_r\} \rangle \mid \text{nome}(\text{nó-fonte}(\langle A_{ik}, A_{rs} \rangle)) \neq \text{nome}(\text{nó-fonte}(\langle A_{jl}, A_{rs} \rangle))$
para $1 \leq i, j, r \leq m, 1 \leq k \leq |A_i|, 1 \leq s \leq |A_r|$ e $1 \leq l \leq |A_j| \}$
-

Caso 2

- a) $\{ \langle \{N_i, N_j\}, \{A_i, A_j\}, \{N_r, N_t\} \rangle \mid \text{nome}(\text{nó-fonte}(\langle A_{ik}, A_{rs} \rangle)) = \text{nome}(\text{nó-fonte}(\langle A_{jl}, A_{tx} \rangle))$
para $1 \leq i, j, r, t \leq m, 1 \leq k \leq |A_i|, 1 \leq s \leq |A_r|, 1 \leq x \leq |A_t|, r \neq t$ e $1 \leq l \leq |A_j| \}$
 - b) $\{ \langle \{N_i, N_j\}, \{A_i, A_j\}, \{N_r, N_t\} \rangle \mid \text{nome}(\text{nó-fonte}(\langle A_{ik}, A_{rs} \rangle)) = \text{nome}(\text{nó-fonte}(\langle A_{jl}, A_{tx} \rangle))$
para $1 \leq i, j, r, t \leq m, 1 \leq k \leq |A_i|, 1 \leq s \leq |A_r|, 1 \leq x \leq |A_t|, r = t, s \neq x$ e $1 \leq l \leq |A_j| \}$
-

Caso 3

- a) $\{ \langle \{N_i, N_j\}, \{A_i, A_j\}, \{N_r\} \rangle \mid \text{nome}(\text{nó-fonte}(\langle A_{ik}, A_{rs} \rangle)) = \text{nome}(\text{nó-fonte}(\langle A_{jl}, A_{rs} \rangle))$
para $1 \leq i, j, r \leq m, 1 \leq k \leq |A_i|, 1 \leq s \leq |A_r|$ e $1 \leq l \leq |A_j| \}$
-

Caso 4

$$a) \{ \langle \{N_i\}, \{A_{ij}, A_{ik}\}, \{N_r, N_t\} \rangle \mid \text{nome}(\text{nó-fonte}(\langle A_{ij}, A_{rs} \rangle)) \neq \text{nome}(\text{nó-fonte}(\langle A_{ik}, A_{tx} \rangle)) \\ \text{para } 1 \leq i, r, t \leq m, 1 \leq s \leq |A_r|, 1 \leq x \leq |A_t|, s \neq x \text{ e } 1 \leq j, k \leq |A_i| \}$$

Caso 5

$$a) \{ \langle \{N_i\}, \{A_{ij}, A_{ik}\}, \{N_r\} \rangle \mid \text{nome}(\text{nó-fonte}(\langle A_{ij}, A_{rs} \rangle)) \neq \text{nome}(\text{nó-fonte}(\langle A_{ik}, A_{rs} \rangle)) \\ \text{para } 1 \leq i, r \leq m, 1 \leq j, k \leq |A_i| \text{ e } 1 \leq s \leq |A_r| \}$$

Caso 6

$$a) \{ \langle \{N_i\}, \{A_{ij}, A_{ik}\}, \{N_r, N_t\} \rangle \mid \text{nome}(\text{nó-fonte}(\langle A_{ij}, A_{rs} \rangle)) = \text{nome}(\text{nó-fonte}(\langle A_{ik}, A_{tx} \rangle)) \\ \text{para } 1 \leq i, r, t \leq m, 1 \leq s \leq |A_r|, 1 \leq x \leq |A_t|, r \neq t, 1 \leq j, k \leq |A_i| \}$$

$$b) \{ \langle \{N_i\}, \{A_{ij}, A_{ik}\}, \{N_r, N_t\} \rangle \mid \text{nome}(\text{nó-fonte}(\langle A_{ij}, A_{rs} \rangle)) = \text{nome}(\text{nó-fonte}(\langle A_{ik}, A_{tx} \rangle)) \\ \text{para } 1 \leq i, r, t \leq m, 1 \leq s \leq |A_r|, 1 \leq x \leq |A_t|, r = t, s \neq x, 1 \leq j, k \leq |A_i| \}$$

Caso 7

$$a) \{ \langle \{N_i\}, \{A_{ij}, A_{ik}\}, \{N_r\} \rangle \mid \text{nome}(\text{nó-fonte}(\langle A_{ij}, A_{rs} \rangle)) = \text{nome}(\text{nó-fonte}(\langle A_{ik}, A_{rs} \rangle)) \\ \text{para } 1 \leq i, r \leq m, 1 \leq j, k \leq |A_i| \text{ e } 1 \leq s \leq |A_r| \}$$

As abstrações de agregação e degeneração dos esquemas diagramados nas Figuras 4.10 a 4.13 ficam assim formalmente especificadas. Por exemplo, na especificação formal do caso 7, o conjunto de nós N_i e N_r que atendem ao requisito de possuir o mesmo *nome* de âncora-fonte e chegar na mesma âncora-destino, pode ser unitário. Dessa forma, constata-se que a especificação da abstração de degeneração do caso 7, referente ao caso 7d, está contemplada.

Para fornecer os mecanismos de especificação das abstrações, na próxima seção é proposto do formalismo de higráficos e mostrada sua adequação como técnica para representação de estrutura de hiperdocumentos.

4.4 Higrafo: uma proposta para a representação da estruturação de *links*

As seções anteriores descreveram os casos de reuso de componentes de *links* como uma forma de fornecer subsídios para a fase de exame, durante a construção de um hiperdocumento. Também apresentaram algumas abstrações que podem ser construídas, com base no reuso de componentes de *links*. Essas abstrações auxiliam o autor a identificar alternativas para uma possível reorganização do hiperdocumento e para possível melhoria da qualidade de sua estrutura geral em relação aos critérios de reuso, consistência e modularidade.

Esta seção apresenta os conceitos básicos de higraphos, que constituem a base sobre a qual são discutidos os recursos necessários desta proposta. Principalmente, são vistos os formalismos que higrapho possui para a representação de *links* entre conjuntos de nós e de estruturação desses conjuntos.

O conceito de **higrafo** (que, em inglês, é *higraph*) foi proposto por Harel (1987a) e detalhado, posteriormente, pelo mesmo autor (Harel, 1988). Harel inspirou-se nos termos *high graphs* e *hierarchical graphs* para a escolha do termo higrapho para nomear o novo conceito que propunha, muito embora os diagramas de higrapho não sejam limitados à estratificação induzida pela palavra *hierarchical*. Harel sugere a aplicabilidade dos higraphos como linguagem gráfica para Diagramas Entidade Relacionamento (do MER), Redes Associativas e Semânticas, e ainda Diagramas de Fluxo de Dados (DFD).

Higrafo é um conceito topológico-visual que combina noções de conjuntos (representados pelos diagramas de Venn/Euler) e hipergrafos. Se por um lado, grafos e hipergrafos são uma maneira conveniente para representar conjuntos de elementos com alguma(s) relação(ões) especial(is) entre eles, por outro, os diagramas de Venn/Euler são uma forma mais adequada para a representação de uma *coleção* de conjuntos, com algum relacionamento de hierarquia entre eles.

Harel afirma que a principal motivação para a definição dos higrافos foi a de ter observado que, na maioria das aplicações computacionais, deve-se considerar a complexidade dos sistemas, dos seus objetos, estados e seus relacionamentos. Para tanto, é necessária uma representação que permita a expressão dos relacionamentos de hierarquia e de relações n-árias dos elementos que compõem as aplicações.

O higrافo é o componente gráfico básico utilizado pelas técnicas *Statechart* (Harel, 1987b) e *Modechart* (Jahanian; Mok, 1994), destinadas à especificação do comportamento de sistemas reativos. Até o momento, não se tem conhecimento na literatura de outras técnicas baseadas em higrافos, a não ser, é claro, aquelas que fazem uso de extensões de *Statechart*.

A pesquisa relatada por Oliveira et. al (1995) propõe a modelagem de hiperdocumentos com base em *Statechart*, com a utilização do modelo denominado HMBS (*Hypertext Model Based on Statecharts*) definido por eles. Nesse modelo, porém, é feita uma adequação da representação estrutural do hiperdocumento, por meio dos tipos de estados possíveis em *Statechart*, dirigida a especificar também uma semântica de navegação baseada na execução do *Statechart* associado ao hiperdocumento.

Um dos objetivos desta tese é propor a utilização de higrافos, especificamente, por ser um formalismo bastante adequado para representação da estrutura de hiperdocumentos. Os higrافos fornecem meios para especificação de relacionamentos entre os nós e entre os conjuntos de nós, tanto estruturais ou hierárquicos (intra-relacionamentos), bem como das relações n-árias dos elementos componentes daqueles conjuntos (inter-relacionamentos). A seguir, sua definição dada por Harel.

Definição 4.1 - Um *higrافo* é uma quádrupla (B, σ, π, A) , onde:

- B é um conjunto finito de elementos chamados “bolhas”;
- σ é a função de hierarquia que associa a uma bolha $\in B$, um conjunto de “sub-bolhas”;
- π é a função de partição que define a decomposição de uma bolha $\in B$ em seus “componentes sub-bolhas”, através de alguma relação entre eles;

- A é um conjunto finito de arcos que correspondem a “hiperarcos direcionados”.

A Figura 4.14 ilustra uma representação gráfica de higrfo, onde se pode observar:

a) O conjunto das “bolhas” que compõem B.

$B = \{A, C, F, B, B1, B2, D, D1, D2, E, G, R, I, I1, I2, P, Q, L, J, J1, J2, K, M, N\};$

b) A função σ de hierarquia. A partir das bolhas é possível obter suas “sub-bolhas”, por exemplo:

- $\sigma(A) = \{C, F\};$
- $\sigma(C) = \emptyset;$
- $\sigma(B) = \{B1, B2\};$
- $\sigma(B1) = \{D\};$
- $\sigma(D) = \{D1, D2\};$

De forma análoga, podem ser obtidas as demais “sub-bolhas” filhas imediatas das “bolhas”. Duas outras funções podem também ser definidas. A função σ^+ fornece o conjunto de todas as sub-bolhas descendentes de uma bolha na hierarquia. A função σ^* fornece as sub-bolhas descendentes de uma bolha e inclui a própria bolha ao conjunto fornecido. Dessa forma, tem-se:

- $\sigma^+(D) = \{D1, D2, F, E\};$
- $\sigma^*(D) = \{D, D1, D2, F, E\};$

A Figura 4.15 esquematiza a árvore que representa toda a hierarquia das “bolhas” definida no higrfo da Figura 4.14. Foi acrescentado um nó “raiz” à descrição dessa estrutura, para mostrar a abrangência da estrutura definida.

Figura 4.14 - Representação gráfica de um higráfico.

- c) A **função de partição** π . A função π define a decomposição de uma bolha $\in B$ em seus “componentes sub-bolhas”, através de alguma relação entre eles. Na definição de Harel, a linha tracejada é acrescentada como notação para representar o Produto Cartesiano. Dessa forma, a Figura 4.14 mostra que o estado J não é simplesmente constituído pela união de K, L, M e N, mas sim pelo produto cartesiano de K com a união dos três últimos. Ou seja,

$$J = J1 \times J2 = K \times (L \cup M \cup N)$$

- d) O **conjunto A de “hiperarcos direcionados”**. São representados na Figura 4.14 pelas setas que ligam “bolhas”. Os hiperarcos direcionados possuem aridade 2; o que liga C a F possui significado semelhante aos arcos de grafos. Já o hiperarco direcionado ligando I a J conecta o conjunto de “bolhas” de I ao conjunto de “bolhas” de J. Comumente interpreta-se os arcos do higráfico como uma coleção de arcos regulares, conectando todo elemento de um conjunto a cada elemento do

outro conjunto. Porém, essa semântica de todos-para-todos não é obrigatória, pois o significado básico do arco do higráfico é de que o relacionamento que ele representa seja válido entre os conjuntos que ele conecta. Portanto, existe a liberdade de se atribuir um significado aos relacionamentos que são definidos entre os conjuntos de elementos através dos “hiperarcos direcionados”. As técnicas *Statechart* e *Modechart*, por exemplo, definem esse relacionamento e para tanto, incluem a noção de “estados *default*” em certos casos.

Uma distinção que se deve considerar entre higráficos e diagramas de Venn/Euler (conjuntos) é a de que, nos higráficos, todo conjunto de interesse deve ser representado graficamente através de uma “bolha” com seus próprios delimitadores de área. Dessa forma, é possível estabelecer os níveis hierárquicos. Portanto, o fato das bolhas R e I se interceptarem no higráfico da Figura 4.14 não significa, necessariamente, que os conjuntos representados por R e I devam ter uma intersecção ou que o conjunto diferença $I - R$ não deva ser vazio. Ou seja, conforme a semântica do formalismo definido em higráficos, a intersecção de duas bolhas não tem significado intrínseco, a não ser que apareçam bolhas totalmente contidas na região de intersecção. Por isso, a bolha R poderia ter sido desenhada não interceptando I.

Figura 4.15 - A árvore que representa a estrutura hierárquica das “bolhas” que compõem o higráfico da Figura 4.14.

Dessa forma, apesar de higraros constituírem um conceito visual, uma linguagem textual contendo sua semântica é suficiente para representar seus elementos e respectivos relacionamentos. Neste sentido, a LES abrange esse requisito.

Com base nessa noção de estruturação hierárquica de bolhas, pode-se concluir que o conjunto de bolhas que realmente pode ser identificado é o conjunto de **bolhas atômicas**, ou seja, o conjunto de bolhas que estão alocadas no nível mais interno do diagrama e que não contêm nenhuma outra bolha. As bolhas atômicas podem ser associadas aos nós existentes em um hiperdocumento. Será visto no Capítulo 5, um sistema protótipo que representa os nós de um hiperdocumento da WWW em estados atômicos de *Statechart*.

Um outro aspecto importante está também relacionado à propriedade de intersecção representada no higraro. O fato da intersecção de duas bolhas ser não vazio sugere a idéia de uso comum das bolhas compartilhadas, isto é, uma possível representação de reuso. Note que, na árvore da Figura 4.15, as bolhas ‘F’ e ‘L’ foram sombreadas, para indicar o fato de serem compartilhadas por diferentes bolhas ancestrais. Dessa forma, como esta tese propõe, o formalismo de higraros contempla a possibilidade de representação do reuso de nós de hiperdocumento, ou seja, a representação daqueles nós que são componentes de nós de composição (como abstrações) distintos.

4.5 Relevância da classificação dos *links*

A metodologia de classificação de *links* proposta e utilizada neste trabalho é um fundamento poderoso para auxiliar o processo de autoria. A classificação dos *links* proposta fornece informações sobre um hiperdocumento em função do reuso dos componentes dos *links* nele definidos.

Ao identificar o reuso dos componentes dos *links*, torna-se possível a visualização de abstrações que não são implementadas em qualquer sistema

hipertexto. Com essas abstrações, o autor pode validar o modelo conceitual a partir dos *links* definidos “à granel” e até mesmo descobrir abstrações implicitamente construídas ao longo da utilização e manutenção do hiperdocumento. Além disso, na obtenção dos *links* que pertencem aos diversos casos, o autor pode examinar suas definições com relação à organização, estilo e uniformidade dos termos criados para sua ancoragem.

A classificação dos casos de links proposta é uma contribuição relevante à etapa de exame durante a confecção de um hiperdocumento. Além de sistematizar hierarquicamente as possíveis abstrações que podem ser construídas, fornece subsídios para a criação de heurísticas para avaliação de hiperdocumentos a serem implementadas. Um exemplo de heurística é considerar que, na ocorrência de muitas abstrações como as do caso 7, dependendo do tamanho dos nós-fonte envolvidos, seja um indicativo de que deve-se segmentar esses nós-fonte.

Dentre as pesquisas relacionadas à avaliação da estruturação de hiperdocumentos é evidente a preocupação maior com relação à análise recursiva do grafo que representa a estruturação dos *links* de um hiperdocumento. Neste sentido, como pode ser visto no capítulo anterior, diversos trabalhos de pesquisa analisaram o grafo que descreve o hiperdocumento e propuseram algoritmos para a coleta de métricas.

O trabalho relevante realizado por Stotts et al. (1992), aborda a avaliação somente a partir do comportamento dos *links* de um hiperdocumento, propondo que o hiperdocumento seja representado por um autômato de *links*, sobre o qual é possível a verificação de propriedades de navegação. Esse autômato de *links*, já mencionado no capítulo anterior (Seção 3.4), é um programa abstrato similar a uma árvore geradora das possíveis seqüências de eventos (seqüências de “cliques” de botões ou seqüências de apresentação de conteúdos, dependendo das propriedades a serem estudadas). Além disso, com base na descrição do hiperdocumento obtida através desse autômato, que representa a estruturação do hiperdocumento, pode-se utilizar uma linguagem de especificação baseada em lógica temporal para expressar várias propriedades de navegação que se quer avaliar. Dessa forma, a ênfase não se

restringe somente à análise recursiva, pois também viabiliza a recuperação de *links* ou nós que atendam a determinada propriedade.

A abordagem original proposta nesta tese, portanto, complementa o trabalho de Stotts et al (1992). Essa combinação permite a recuperação de outras propriedades mais orientadas a caminhos em grafos. Porém, deve-se ressaltar que a técnica de Stotts baseia-se em máquinas de estados para representação da estrutura do hiperdocumento, que não contemplam a visão de estruturas abstratas, como as agregações e degenerações que foram identificadas neste trabalho. Por isso é que a adoção de uma técnica com mais recursos semânticos é proposta nesta tese.

Os dois pontos de maior interesse deste trabalho na técnica de higráficos se referem ao poder de representação das relações n -árias dos conjuntos e o poder de representação da estrutura de hierarquia.

É importante notar que a proposta de representação de hiperdocumentos como higráficos não compromete nem interfere no uso de quaisquer dos métodos para modelagens (Seção 3.2) para definição conceitual do hiperdocumento. É antes disso, um formalismo matemático proposto nesta tese para auxiliar na validação das construções abstratas adotadas por aqueles modelos e permite o exame do hiperdocumento segundo a classificação baseada em *links* proposta anteriormente. Para dar suporte automático à abordagem de análise relacional descrita neste capítulo e demonstrar a sua aplicabilidade, foi desenvolvido um sistema orientado a *links* (LiOS) para experimentos em hiperdocumentos da WWW, descrito no próximo capítulo.

Capítulo 5

LiOS - um sistema orientado a *links* para exame de WWW

A teoria básica da abordagem definida como análise relacional de *links*, a partir da qual foi proposta uma classificação de reuso de componentes de *links* no Capítulo 4, fornece subsídios para a construção de um sistema protótipo que visa obter um levantamento dos requisitos de qualidade, quanto aos critérios de reuso e consistência. O sistema foi denominado Sistema Orientado-a-*Links* que, em inglês, é traduzido para *Link-Oriented System* (LiOS).

A implementação atual do LiOS viabiliza que um hiperdocumento da WWW possa ter sua estrutura examinada pelos seus autores, os *webmasters*. A escolha por hiperdocumentos da WWW foi feita considerando-se:

1. Como observado no Capítulo 3, sobre as tendências de evolução, a recente globalização dos hiperdocumentos como base de informações da WWW é um novo desafio para a avaliação de aplicações hipermídia;
2. A disponibilidade de material que dia após dia é modificado consiste em um considerável corpo de prova para análises consecutivas.

A WWW é um exemplo de sistema hipertexto que possui as seguintes características:

- As informações são organizadas como um ‘mundo sem fronteiras’, na forma de uma rede de nós (documentos) que possuem uma referência pela qual podem ser recuperados;
- O ‘mundo’ de nós referenciáveis obedece a um sistema de endereçamento denominado URI (*Universal Resource Identifier*) (Berners-Lee et al., 1994) que permite o acesso a objetos que estejam disponíveis por qualquer protocolo, de forma legível para o homem. Atualmente, o método de endereçamento URL (*Uniform Resource Locator*) é que, de fato, efetiva um ‘padrão’ para localização dos recursos distribuídos pela Internet. O URL segue a sintaxe URI, mas ainda existem discussões a respeito da viabilidade na adoção de URI. URL é de fundamental importância na arquitetura atual da WWW;
- A comunicação na WWW se dá segundo uma arquitetura cujo modelo é cliente-servidor. Essencialmente, um servidor é constituído de programas servidores de arquivos (que correspondem aos nós) que atendem a requisições de outros programas, denominados clientes. Os programas clientes, em geral, correspondem aos apresentadores (*browsers*);
- Existe um protocolo de rede (HTTP) usado pelos servidores da WWW que possibilita a transferência das informações com a eficiência necessária para o leitor do hipertexto “saltar” para outros nós, por meio dos *links*;
- Os nós são formatados segundo HTML, linguagem ‘entendida’ por todo cliente da WWW.

O LiOS identifica os casos de reuso de componentes de *links*, com base na estruturação de um hiperdocumento da WWW. Seu principal objetivo é auxiliar os autores a avaliar a organização, estilo e uniformidade dos termos criados para ancoragem dos *links*. Dessa forma, a avaliação proporcionada com o LiOS se enquadra entre os dois níveis de granularidade, de conteúdo e de estrutura de um

hiperdocumento, uma vez que a ancoragem dos *links* se situa como uma interface entre esses dois níveis, segundo o modelo Dexter.

O sistema LiOS foi, então, projetado e construído de acordo com essas características. A seguir, a arquitetura do sistema e os módulos funcionais que foram desenvolvidos são apresentados.

5.1 Arquitetura do sistema LiOS

Um dos objetivos principais do sistema LiOS é enfatizar a apresentação dos *links* criados no hiperdocumento da WWW para possibilitar um **exame** da estrutura existente no mesmo, com base nos casos de reuso dos componentes estruturais, ou seja, no reuso de seus *links*.

A Figura 5.1 apresenta a descrição elaborada para o componente estrutural *link* na forma de objetos, conforme a notação proposta no método FUSION (Coleman et al., 1994) para a modelagem orientada a objetos. Nessa descrição existe o objeto LINK, que corresponde a uma classe do tipo agregada, composta das classes nó-fonte, âncora e nó-destino, e dos relacionamentos (indicado por losangos) entre elas, estabelecidos como: nó-fonte ‘possui’ várias âncoras e âncora ‘leva a’ um nó-destino.

Figura 5.1 - O objeto *LINK* e seus componentes.

Como o objetivo do sistema LiOS se fundamenta na abordagem relacional apresentada no capítulo anterior, a Base de Dados central do protótipo, contendo as informações referentes aos componentes dos *links*, foi modelada com uma entidade

principal *LINK*, com os atributos constituídos de cada um dos seus componentes: nó-fonte, âncora e nó-destino. A partir desse modelo de dados, o LiOS foi projetado com três módulos funcionais, como mostra a Figura 5.2. Nessa arquitetura foram especificados os seguintes processos:

- módulo **Representação do HD** — para representação da estrutura de *links* de um hiperdocumento referente a um *site* da WWW;
- módulo **Conversor** — para criação da base contendo as estruturas obtidas no processo anterior; e
- módulo **Apresentação dos links** — para apresentação da base de *links*, com vistas a possibilitar o exame da estrutura geral do hiperdocumento.

Figura 5.2 - Arquitetura modular do LiOS.

Nas próximas seções são descritas com mais detalhes as funções dos processos que compõem o LiOS.

5.2 O módulo Representação do HD

Inicialmente, o sistema LiOS deve obter a representação dos *links* de um hiperdocumento em HTML. Como o objetivo principal é a base de informações de um hiperdocumento da WWW, o enfoque é dado sobre um servidor da WWW.

A representação natural seria a de um dígrafo. Porém, visando a investigação das abstrações na estrutura do hiperdocumento, foi utilizada a técnica *Statechart*, que se fundamenta em higrafos, definidos no Capítulo 4.

A representação da estrutura de um hiperdocumento através de dígrafos possibilita a utilização da técnica DTE de forma trivial, para descrever o comportamento mais simples dos *links* do hiperdocumento, conforme ilustra a Figura 5.3. Os *links* são os elementos de interação do usuário com o sistema hipertexto, que não são definidos na interface do sistema mas sim no conteúdo dos nós de um hiperdocumento pelas âncoras. Neste contexto, portanto, o enfoque não está na representação da interatividade do sistema mas sim na representação dos elementos que estruturam as informações contidas em um hiperdocumento.

Figura 5.3 - DTE representando um *link* unidirecional de hiperdocumento.

A solução mais direta para se descrever a estruturação do hiperdocumento na forma de um DTE, é pelo mapeamento dos componentes do hiperdocumento em componentes do DTE, como mostra a Tabela 5.1.

Tabela 5.1 - Componentes de estruturação de hiperdocumento como componentes de DTE

HIPERDOCUMENTO	DTE
nó-fonte	estado-origem (A)
nó-destino	estado-destino (B)
âncora-fonte/ ponto-de-partida botão de controle	evento (x)
<i>link</i>	transição
âncora-destino/ ponto-de-chegada	estado-destino

O componente **transição** no DTE, apesar de não ser um elemento gráfico, sintetiza a semântica de mudança na configuração dos estados, ou seja, ele corresponde ao elemento *link*, que define a ligação entre os nós de informação e é o responsável pela habilitação do mecanismo de navegação inerente aos sistemas hipertexto.

Esse mapeamento leva em conta que a transição de estado (a navegação de um nó-fonte para um nó-destino) é efetivada quando a âncora-fonte estiver habilitada como botão de controle (o evento) e for selecionada (ou ativada) por pressionamento do *mouse* sobre a mesma (ou, ainda, por outro mecanismo de acionamento), pelo usuário. Dessa forma, o evento de mudança de estado de um nó no hiperdocumento é disparado, obtendo-se por exemplo, como resultado, as informações contidas em um nó ligado pelo *link*.

Deve-se notar que a definição de *links* bidirecionais na WWW é, atualmente, praticamente um trabalho de redundância. Existem, evidentemente, muitas queixas relacionadas ao problema de *dangling links* (pendentes), que poderiam ser eliminadas se a HTML fornecesse algum suporte para a definição em um determinado nó, dos nós-fonte dos *links* que chegam nele.

No entanto, como os documentos WWW são escritos em HTML, que é uma linguagem que possui elementos de descrição somente para *links* unidirecionais, o sistema LiOS foi projetado considerando somente *links* unidirecionais. Essa consideração pode ser reavaliada e reformulada, uma vez que está pontualmente estabelecida. As discussões, apresentadas no Capítulo 4 (Seção 4.2), para uma descrição textual de *links* bidirecionais apresentam uma solução viável.

Dessa forma, pode-se concluir que um hiperdocumento pode ter sua estrutura descrita em LES, mesmo utilizando parcialmente os recursos que a linguagem possui. De fato, no contexto desta tese, a técnica *Statechart* está sendo utilizada na forma da LES com o objetivo de explorar seu fundamento no formalismo de higrافos.

Por *Statechart* ser uma extensão de DTEs, a LES contém as características de DTE e adiciona outras para atender à semântica específica de *Statechart*.

A descrição obtida dos *links* poderia ter sido diagramada, como mostra o higrافo pontilhado na Figura 5.2, entretanto a descrição textual por LES atende todos os propósitos estabelecidos. Como o hiperdocumento WWW não possui abstrações de composição ou de contexto, a representação do *Statechart* consiste na especificação do DTE, acrescida de um superestado abstrato ‘hiperdoc’ que contém todos os estados (nós) realmente existentes.

O módulo “Representação do HD” desenvolvido varre o hiperdocumento, seguindo os seus *links* locais para gerar um arquivo ASCII com a estrutura do hiperdocumento, ou seja, uma especificação dos seus *links* descrita em LES. A varredura não avança na obtenção de *links* para arquivos que tenham que ser obtidos por algum protocolo de transferência. Neste caso; obtém-se uma **representação da estrutura do hiperdocumento local**, ou seja, uma representação confinada ao sistema de arquivos de um servidor da WWW, sobre o qual os autores locais possuem autoria direta.

Para a descrição dos *links* obtidos dos documentos HTML, em LES, foi utilizada a noção de que a âncora apresentada ao leitor no nó corrente (nó-fonte) corresponde ao evento que o leitor aciona para que seja seguido o *link*. A cadeia de caracteres que é delimitada pela marcação (*tag*) `</a>` no documento HTML é adotada como sendo a âncora-fonte, quer seja ela a própria a ser apresentada durante a navegação, quer seja uma composição de um arquivo que possui um *icone* (imagem) que represente a âncora. Como a HTML somente permite a definição de *links* unidirecionais, foi feita a correspondência dos nós-fonte com as origens das transições e dos nós-destino com os estados-destino das transições.

Uma outra característica importante de funcionalidade desse módulo é a “resolução dos endereços em URL”. Para que seja possível a identificação do real reuso dos componentes, o endereçamento embutido nas referências dos *links* contidos nos documentos em HTML devem ser decodificados e uniformemente codificados para o endereçamento local do servidor de arquivos do hiperdocumento WWW, ou seja, daquele *site*.

O ambiente de programação das rotinas desse módulo foi o de estações de trabalho Sun (sparc) com Sistema Operacional Unix. As rotinas foram implementadas em linguagem de programação C e totalizam em torno de 1.500 linhas de código.

```
<html><head><title>ICMSC-USP Homepage</title></head>

<p>    <a href="/ICMSC/"><IMG SRC="/fotos/button.gif" ></a>
    <a href="/ICMSC/">About the Institute</a><br>
    Location, organization, departments, courses, research
<p>    <a href="/staff/"><IMG SRC="/fotos/button.gif" ></a>
    <a href="/staff/">Faculty </a><br>
    List by Departments; links to homepages
<p>    <a href="/ICMSC/icmsc-jr.html"><IMG SRC="/fotos/button.gif" </a>
    <a href="/ICMSC/icmsc-jr.html">ICMSCjr</a><br>
    Junior Enterprise of the ICMSC students
<p>    <a href="/eventos/"><IMG SRC="/fotos/button.gif" ></a>
    <a href="/eventos/">Events at the Institute</a><br>
    Seminars, speeches, symposiums, Summer Courses
<p>    <a href="/servicos.html"><IMG SRC="/fotos/button.gif" ></a>
    <a href="/servicos.html">Services</a><br>
    Phone book; WWW, FTP and Gopher servers; news
<p>    
    <h5 align=center>
    <i>Last edited November 06, 1995 <br>
    <a href = "mailto:webmaster@icmsc.sc.usp.br">
    webmaster@icmsc.sc.usp.br</a></i><p>
    <a href="/Portugues/">texto em Portugues </A>
    <a href="http://www.usp.br/">USP Homepage</a>

</h5>
</body>
</html>
```

Figura 5.4 - A página de entrada do *site* do servidor WWW do ICMSC-USP codificada em HTML.

A Figura 5.4 mostra um exemplo de uma página codificada em HTML, um dado de entrada para o módulo.

```

Estado "hiperd" ou subestados " /httpd/docs/index.html" def,
"/httpd/docs/ICMSC/index.html", "httpd/docs/staff/index.html",
"/httpd/docs/ICMSC/icmsc-jr.html", "/httpd/docs/eventos/index.html",
"/httpd/docs/servicos.html", "mailto:webmaster@icmsc.sc.usp.br",
"/httpd/docs/Portugues/index.html", "http://www.usp.br/"
Estado "/httpd/docs/index.html" atomo
Estado "/httpd/docs/ICMSC/index.html" atomo
Estado "/httpd/docs/staff/index.html" atomo
Estado "/httpd/docs/ICMSC/icmsc-jr.html" atomo
Estado "/httpd/docs/eventos/index.html" atomo
Estado "/httpd/docs/servicos.html" atomo
Estado "mailto:webmaster@icmsc.sc.usp.br" atomo
Estado "/httpd/docs/Portugues/index.html" atomo
Estado "http://www.usp.br/" atomo ;

Evento "<IMG SRC=~/fotos/button.gif'"
origem "/httpd/docs/index.html"
destino "/httpd/docs/ICMSC/index.html"
Evento "About the Institute"
origem "/httpd/docs/index.html"
destino "/httpd/docs/ICMSC/index.html"
Evento "<IMG SRC=~/fotos/button.gif'"
origem "/httpd/docs/index.html"
destino "/httpd/docs/staff/index.html"
Evento "Faculty"
origem "/httpd/docs/index.html"
destino "/httpd/docs/./staff/index.html"
Evento "<IMG SRC=~/fotos/button.gif'"
origem "/httpd/docs//index.html"
destino "/httpd/docs/ICMSC/icmsc-jr.html"
Evento "ICMSCjr"
origem "/httpd/docs/index.html"
destino "/httpd/docs/ICMSC/icmsc-jr.html"
Evento "<IMG SRC=~/fotos/button.gif'"
origem "/httpd/docs/index.html"
destino "/httpd/docs/eventos/index.html"
Evento "Events at the Institute"
origem "/httpd/docs/index.html"
destino "/httpd/docs/eventos/index.html"
Evento "<IMG SRC=~/fotos/button.gif'"
origem "/httpd/docs/index.html"
destino "/httpd/docs/servicos.html"
Evento "Services"
origem "/httpd/docs/index.html"
destino "/httpd/docs/servicos.html"
Evento "webmaster@icmsc.sc.usp.br"
origem "/httpd/docs/index.html"
destino "mailto:webmaster@icmsc.sc.usp.br"
Evento "texto em Portugues"
origem "/httpd/docs/index.html"
destino "/httpd/docs/Portugues/index.html"
Evento "USP Homepage"
origem "/httpd/docs/index.html"
destino "http://www.usp.br/" ;

```

Figura 5.5 - Declaração em LES dos *links* contidos na página da Figura 5.4, resultante do módulo "Representação do HD".

A Figura 5.5 mostra a listagem resultante do processamento do módulo “Representação do HD”, que corresponde à representação dos *links* codificada em LES, *links* esses contidos na página em HTML da Figura 5.4.

5.3 O módulo Conversor

O passo seguinte é o da criação de uma Base de Dados Relacional contendo os elementos da estrutura do hiperdocumento, elementos esses resultantes do processamento do módulo anterior “Representação do HD”. Os *links* são mapeados na forma de componentes evento/origem/destino, correspondendo às transições do *Statechart*.

No modelo de referência de Dexter essa Base de Dados corresponde à camada de Armazenamento e subsidia a máquina de hipertexto na resolução de *links*. Essa Base de Dados contém as informações sobre os *links* do hiperdocumento. O módulo “Conversor” é responsável pela criação de identificadores únicos para cada um dos componentes de *links*, em adequação ao modelo de objetos apresentado na Figura 5.1 e para suprir as consultas disponibilizadas no módulo de “Apresentação dos *links*”. No caso, foi utilizado o software Microsoft Visual FoxPro versão 3.0 (um ambiente de desenvolvimento de programas com suporte de um Sistema Gerenciador de Base de Dados Relacional) que já possui um conversor automático interno de Bases de Dados em ASCII e gera a tabela correspondente no formato de Base de Dados Relacional (.dbf).

Por se tratar da estruturação de páginas escritas em HTML, as tuplas referentes aos *links* podem ser assim definidas:

(id-link, âncora-fonte, nó-fonte, nó-destino.âncora-destino)

onde :

id-link = identificador-de-link (chave- primária);
 âncora-fonte = evento;
 nó-fonte = origem;
 nó-destino.âncora-destino = destino.

A âncora-destino é considerada como pertencente ao atributo destino porque ela simplesmente desempenha um papel de desvio dentro do ponto de apresentação do nó-destino quando um usuário-leitor decide seguir o *link* que a contenha na terminação do destino. Dessa forma, são considerados *links* distintos. Por HTML suportar a definição de *links* unidirecionais, a âncora-destino não possui o papel de controle, como acontece com a âncora-fonte, que é o ponto de entrada para a interação durante a navegação de um leitor.

5.4 O módulo Apresentação dos *links*

Ao invés do autor ver somente os *links* a partir dos nós que ele cria, a abordagem do LiOS focaliza os *links*, e assim os nós que eles conectam. Dessa forma, procura-se fornecer ao autor uma visão complementar de sua autoria.

Essa visão orientada a *links* apresenta uma relação geral de todos os *links* do hiperdocumento. O autor obtém a visão de todos os componentes estruturais do hiperdocumento na forma de uma lista, uma vez que a montagem da estrutura toda em mapas gráficos ou diagramáticos é uma tarefa complexa e geralmente leva a um enorme emaranhado de linhas de conexão, conforme se discute na literatura em geral.

A Figura 5.6 mostra um *dump* de tela com as janelas resultantes desse módulo "Apresentação dos *links*". Este módulo de apresentação foi prototipado utilizando-se a linguagem TCL (versão 7.4) com o *toolkit* gráfico TK (versão 4.0). Para o armazenamento dos dados foi utilizado um servidor SQL "Mini-SQL" (versão 1.0.6), instalado em uma máquina remota.

Figura 5.6 - As janelas que apresentam os *links* de um hiperdocumento da WWW, mostrando os nós-fonte e destino que são conectados por eles.

A janela principal, situada à esquerda na tela da Figura 5.6, possui uma lista de todos os nomes de âncoras que o autor usou no hiperdocumento. No modelo Dexter, as âncoras possuem um tratamento específico e estão associadas a uma camada de interface, entre as camadas de armazenamento e a camada interna aos componentes, denominada Ancoragem. Portanto, a Ancoragem é um mecanismo diferencial dos sistemas hipertexto, que também deve ser avaliado para contribuir para melhoria da qualidade de um hiperdocumento.

A relação dos nomes de âncoras fica disponível para seleção e, uma vez selecionado um *link* através de um determinado nome (de âncora), são obtidas duas novas listas (janelas ao lado da janela principal): a primeira, com todos os nós-fonte que possuem a âncora selecionada como ponto de partida para o *link* e a segunda, com todos os nós-destino que possam ser alcançados como ponto de chegada dos *links* que partiram da âncora selecionada.

5.5 Usando o LiOS

Como a Base de Dados dos *links* é relacional, para complementar esse tipo de exame da estrutura do hiperdocumento a ser feito pelos autores, as formas alternativas de apresentação dos *links* requerem apenas o esforço de visualizar cada um dos componentes de *link* como atributo-chave.

Uma expressão para formalizar o relacionamento existente entre os componentes nó-fonte (NF), âncora (A) e nó-destino (ND), para a obtenção das possíveis visões é dada por:

$$NF(A) = ND$$

Essa expressão pode ser lida como a instancição de uma âncora A em um determinado nó-fonte NF, estabelecendo um certo nó-destino ND. Assim, para se obter as visões possíveis desse relacionamento, basta substituir um ou dois dos elementos por variável livre. Por exemplo, a expressão $NF(?) = ?$ fornece a visão das âncoras que levam aos nós-destino a partir de um nó-fonte instanciado. As visões que possuem maior relevância para o propósito de exame dos *links* são esquematizadas na Figura 5.7.

A visão apresentada pelo módulo que tem como atributo-chave a lista dos nomes de âncoras está esquematizada na Figura 5.7(a). Essa visualização, além de habilitar o autor a poder examinar todos os valores de âncoras utilizados no hiperdocumento, permite que sejam formuladas questões como:

— *Quais são todos os nós-fonte que possuem uma determinada âncora A em seus links, e para quais nós-destino esses links remetem?*

Em suma, as respostas a esse tipo de questão fornecem a visão dos *links* dos casos cuja característica é possuir mesma âncora-fonte. Esses são o caso 2, caso 3, caso 6 e caso 7.

Figura 5.7 - Formas alternativas de apresentação da estrutura do hipertexto, com base nos seus *links*.

As âncoras são os elementos de controle através dos quais os *links* ficarão disponíveis durante a navegação. Como o *link* é um tipo de relacionamento *n* para *m*, (ou seja, do ponto de vista do identificador de *link*, a mesma âncora-fonte pode rotular *n* nós-fonte a *m* nós-destino) o autor pode desejar ver as diferentes conexões que as âncoras-fonte estão rotulando no hipertexto. Como resultado dessa apresentação, destaca-se o fato do autor poder ver as dependências relativas ao *link*, ou seja, todos os nós que possuem *links* rotulados com aquelas âncoras-fonte. Nesse sentido, o autor pode analisar as inconsistências geradas por nomeação indevida do *link*.

As outras duas formas de apresentação dos *links* ilustradas em 5.7(b) e 5.7(c), permitem as seguintes visões:

1. Se o atributo-chave for o nó-fonte - tem-se a visão resumida da forma convencionalmente utilizada durante a autoria de documentos HTML. Com essa visualização, além do autor poder examinar todos os arquivos correspondentes a nós-fonte, a questão que ele pode formular para obter informações diretas é:

— *Quais são todas as âncoras que existem em um determinado nó-fonte NF, e os links estabelecidos por essas âncoras remetem para quais nós-destino?*

As respostas a esse tipo de questão fornecem a visão dos *links* dos casos cuja característica é possuir mesmo nó-fonte. Esses são o caso 4, caso 5, caso 6 e caso 7.

2. Se o atributo-chave for o nó-destino - tem-se a visão resumida da forma inversa utilizada pelos autores de documentos HTML, i.e., a visão dos *links* que chegam em determinado nó-destino. Uma consequência imediata dessa forma de apresentação é a visualização dos *links* pendentes (*dangling*). Com essa visualização, além do autor poder examinar todos os arquivos correspondentes a nós-destino utilizados no hiperdocumento, a questão que ele pode formular para obter informações diretas é:

— *Quais são todas as âncoras que rotulam os links que chegam em um determinado nó-destino ND, e de quais nós-fonte esses links partem?*

As respostas a esse tipo de questão fornecem a visão dos *links* dos casos cuja característica é possuir mesmo nó-destino. Esses são o caso 1, o caso 3, o caso 5 e o caso 7.

Essas visões alternativas podem ser mais refinadas, com a implementação de “filtros”, para se obter visões específicas de cada caso. Esses filtros são módulos que, por exemplo, suprimem as ocorrências em duplicata das instâncias de um componente.

5.6 Análise empírica do uso de LiOS

A partir das aplicações iniciais do sistema LiOS em servidores da WWW para exame de sua estruturação, análises mais específicas para avaliação de hiperdocumentos podem ser realizadas.

Um dos pontos que pode ser analisado é o da **liberdade de nomeação de links**. A máquina de hipertexto requer endereços precisos para localizar os nós-destinos durante a navegação, obtidos por:

- mecanismo antecipado (o texto do nó-fonte está previamente compilado, ficando disponíveis para o sistema de navegação os endereços dos nós-destino já resolvidos). Neste caso, a Base de Dados dos *links* já possui o endereçamento especificado, bastando somente buscar o nó-destino durante a execução da navegação.
- mecanismo de interpretação (através de um protocolo de comunicação, como é o caso do HTTP, que interpreta os arquivos-texto dos nós-fonte em HTML na WWW, deixando os endereços completamente especificados em URL. Neste caso, inexistente uma Base de Dados dos *links* e o servidor da WWW é o programa que juntamente com outros programas da Internet, como o DNS (*Domain Name Server*), são responsáveis por interpretar o endereço especificado em URL e buscar o nó-destino durante a execução da navegação.

Portanto, enquanto a nomeação dos nós está consolidada, o mesmo não acontece com a nomeação dos *links*, que fica inteiramente livre e a cargo do autor. É nesse ponto que o autor usa sua criatividade para imprimir o estilo e organização das informações que pretende imprimir ao hiperdocumento.

Outro ponto que pode ser considerado é o de **reuso de nós e links**. A seção anterior apresentou uma proposta de classificação de casos de reuso de componentes de *links*, com o objetivo de identificar estruturas abstratas. Com a utilização do sistema LiOS, foi possível visualizar as diversas ocorrências de tais componentes na estrutura do hiperdocumento, o que contribui para a abordagem de exame proposta no capítulo anterior. Apesar de se ter a visualização no sentido *bottom-up* de

construção das estruturas do hiperdocumento, uma vez que são apresentados os componentes sem qualquer abstração, o autor pode verificar os *links* que deveriam ser definidos para certas abstrações como classes de informação.

Capítulo 6

Resultados

O sistema protótipo LiOS, apresentado no capítulo anterior, foi projetado para viabilizar o exame da estrutura de hiperdocumentos da WWW pelos *webmasters*, com base na análise relacional de reuso de componentes de *links* proposta no Capítulo 4. Com o propósito de mostrar a relevância da análise relacional de reuso de *links* definida, este capítulo apresenta os resultados obtidos, em termos quantitativos, de experimento realizado sobre hiperdocumento da WWW.

O experimento, em linhas gerais, consistiu da extração da estrutura de um hiperdocumento da WWW e do posterior levantamento, utilizando essa estrutura, das incidências dos *links* de cada um dos casos de reuso definidos.

Inicialmente, neste capítulo é feita a descrição do experimento e do corpo de prova utilizado. A seguir, os resultados quantitativos obtidos são apresentados e analisados. Finalmente, são relatadas as impressões causadas, junto aos *webmasters*, dos resultados do experimento.

6.1 Descrição do experimento

O objeto em estudo para o levantamento dos dados experimentais foi a estrutura de um hiperdocumento da WWW. O objetivo almejado do experimento foi a coleta das porcentagens de ocorrências de cada um dos casos de reuso de *links* identificado. Para obtenção dessas porcentagens foi utilizado um sistema de Base de Dados Relacional, o Visual FoxPro, que fornece respostas às consultas construídas em sua SQL embutida.

A idéia principal do experimento foi a de possibilitar um acompanhamento da evolução dos casos de reuso dos *links* desse hiperdocumento, paralelamente às manutenções nele ocorridas. Para a condução do experimento de obtenção dos dados em questão, o objeto de estudo escolhido foi o hiperdocumento do ICMSC-USP (*site* www.icmsc.sc.usp.br) da WWW. Esse hiperdocumento, que se encontra disponível no servidor local do ICMSC-USP da WWW, possui características que lhe conferem a propriedade de ser um corpo de prova representativo. Entre as características desse corpo de prova, ressalta-se:

1. A autoria de suas páginas é efetuada por diversos *webmasters*. Além de incluir páginas criadas por funcionários, alunos e professores, cuja efetiva composição pode ter origens diversas, existe uma rotatividade com relação à administração do hiperdocumento inteiro, pois os *webmasters* são, em geral, alunos;
2. O volume de nós e *links* que tem sido criado é significativo, alcançando a ordem de milhares;
3. O início da construção desse hiperdocumento data de meados de 1994;
4. O domínio de informações disponibilizado é bastante aleatório e, apesar de ter as informações básicas preservadas, constantemente ocorrem inclusões/remoções de nós e *links* para propósitos temporários de aulas, exercícios e manuais.

Tabela 6.1 - Consultas para obtenção das incidências de *links* em cada um dos casos de reuso de componentes de *links*.

caso	recuperação dos <i>links</i>	consulta (query)
0	que não possuem qualquer reuso de estrutura em relação aos demais <i>links</i> ;	<i>Complementaridade</i>
1	que chegam ao mesmo nó-destino;	Select count(*) as c1, * from base group by destino having c1 > 1
2	que possuem mesma âncora-fonte;	Select count(*) as c2, * from base group by evento having c2 > 1
3	que possuem mesma âncora-fonte e mesmo nó-destino;	Select count(*) as c3, * from base group by evento, destino having c3 > 1
4	que partem de mesmo nó-fonte;	Select count(*) as c4, * from base group by fonte having c4 > 1
5	que possuem mesmo nó-fonte e mesmo nó-destino;	Select count(*) as c5, * from base group by fonte, destino having c5 > 1
6	que possuem mesmo nó-fonte e mesma âncora-fonte;	Select count(*) as c6, * from base group by fonte, evento having c6 > 1
7	que possuem mesma âncora-fonte, mesmo nó-fonte e mesmo nó-destino;	Select count(*) as c7, * from base group by evento, fonte, destino having c7 > 1

Esse hiperdocumento foi fornecido como *input* para os dois primeiros módulos do LiOS, o módulo “Representação do HD” e o “Conversor”. Os casos de reuso de *links* identificados por rótulos variando de 0 a 7, apresentados no Capítulo 4, puderam ser transcritos por consultas na SQL do Visual FoxPro, conforme pode ser visto na Tabela 6.1. A partir dessas consultas, obteve-se as incidências dos *links*, relativas a cada um dos casos.

6.2 Descrição dos resultados

A coleta de dados foi realizada várias vezes, tendo sido iniciada em setembro de 1995. São apresentados os dados referentes a três diferentes coletas: as realizadas em setembro e novembro de 1995 e a realizada em julho de 1996. A escolha dessas datas foi aleatória, de forma a não induzir possíveis falsas inferências. Na Tabela 6.2 são apresentados os resultados de contabilizações dos casos de reuso de componentes de *links* a partir dos dados obtidos nos três levantamentos.

Tabela 6.2 - Contabilizações resultantes de consultas a incidências de *links* de cada um dos casos de reuso de componentes de *links*.

caso	setembro/1995	novembro/1995	julho/1996
	2.334 <i>links</i> 1.406 nós	2.433 <i>links</i> 1.419 nós	8.066 <i>links</i> 1.201 nós
0	R=19 L=19 0,81%	R=18 L=18 0,74%	R=26 L=26 0,32%
1	R=127 L=413 17,69%	R=118 L=361 14,77%	R=426 L=1.583 19,62%
2	R=89 L=281 12,03%	R=83 L=196 8,02%	R=356 L=1.198 14,85%
3	R=227 L=649 27,80%	R=261 L=712 29,14%	R=1.076 L=3.541 43,9%
4	R=155 L=1.674 71,72%	R=169 L=1.735 71,01%	R=801 L=7.117 88,23%
5	R=86 L=216 9,25%	R=82 L=209 8,55%	R=165 L=358 4,43%
6	R=29 L=182 7,79%	R=16 L=150 6,13%	R=55 L=257 3,18%
7	R=34 L=140 5,99%	R=36 L=145 5,93%	R=83 L=244 3,02%

Cada relação intermediária, resultante de consulta em SQL, foi contabilizada nas porcentagens de incidência de cada um dos casos (1 a 7) no hiperdocumento da WWW, por meio da aplicação das funções COUNT() e SUM(). A relação com os *links* do caso 0 foi obtida apenas após a construção das relações dos demais casos, por razões apresentadas mais adiante.

Os valores dos resultados obtidos revelam informações relevantes sobre o comportamento do reuso de *links* no hiperdocumento. Antes de analisar esse comportamento, entretanto, algumas considerações relacionadas ao procedimento para obtenção dos dados (constantes da Tabela 6.2) devem ser mencionados:

1. Os valores associados a R representam os totais de ocorrências distintas dos *links* em cada relação resultante da consulta em SQL. Corresponde ao valor retornado pela função COUNT(), que conta o número de registros da tabela. Mais especificamente, esses valores de R correspondem às quantidades de tipos de *links* que possuem determinado reuso, tipo de reuso esse descrito pelo operador de agrupamento (*group by*) na consulta;

2. As quantidades de *links* efetivos, contabilizadas em L, correspondem aos valores retornados pela função SUM(), que faz o somatório das quantidades de incidências para cada tipo de *link* agrupado;
3. Para a contabilização das incidências de *links* nos casos, foram consideradas as relações de conjunto de casos que são subconjuntos de outro. Assim, os casos que eram subconjuntos de outros, foram contabilizados utilizando-se os devidos predicados de seleção na cláusula *where*.

Deve-se notar que as porcentagens de todos os casos somadas não totalizam exatamente 100%, pois os *links* não são considerados exclusivamente como pertencendo a um só caso. Um determinado nó pode desempenhar o papel de nó-fonte em vários tipos de *links*.

Somente o conjunto de incidências de *links* classificados como do caso 7 (*links* com mesmo nó-fonte, nó-destino e âncora) não contém nenhum dos outros conjuntos de *links* que pertence a outros casos. Os demais casos (0 a 6) podem estar contidos ou possuir interseção não vazia entre si.

Como não foi possível especificar diretamente numa consulta em SQL os requisitos de recuperação dos *links* pertencentes ao caso 0, foi definido o procedimento de *Complementaridade*. Esse procedimento contabiliza as incidências do caso 0 ao final de obtidas as relações dos demais casos, selecionando os *links* que não se enquadraram em nenhum dos outros casos.

A relação dos *links* classificados como de caso 0, que é caracterizado por possuir todos os componentes de *links* distintos em relação à Base de Dados de *links* do hiperdocumento, foi obtida após uma junção (*join*) de todas as relações dos demais casos (1 a 7) e, sobre a relação resultante, aplicada a seguinte consulta em SQL:

```
select * from base;
      where base.c1 = 0 and base.c2 = 0 and base.c3 = 0;
            and base.c4 = 0 and base.c5 = 0 and base.c6 = 0;
            and base.c7 = 0 ;
into table caso0 ;
```

Assim, as ocorrências contabilizadas no caso 0 se referem às quantidades de *links* totalmente distintos entre si, que não foram identificados como se adequando em nenhum dos outros casos. Portanto, representam os *links* que não possuem qualquer reuso identificado pelos demais casos.

6.3 Análise

A análise dos resultados obtidos com o experimento foi efetuada sob as seguintes perspectivas:

- as diferentes proporções de incidências dos diversos casos;
- o volume de *links* existente em cada um dos casos, nos três momentos distintos de coleta de dados.

Nota-se, observando-se os resultados obtidos nas três coletas, que o volume de *links* cresceu cronologicamente. Embora não se possa afirmar que essa tendência seja sempre verdadeira, informalmente observa-se que, à medida que os autores dominam o conhecimento sobre a estrutura geral dos nós e de suas informações, eles se sentem mais estimulados a criar novas ligações entre as informações visando disponibilizá-las mais facilmente.

A Figura 6.1 apresenta um gráfico mostrando os relacionamentos analisados, a partir dos quais se pode constatar que:

- Os *links* classificados como de uso-geral (caso 3 e caso 4) são os mais comumente encontrados na estrutura do hiperdocumento;
- Na classe de uso-particular de *links* (caso 0 e caso 7) estão aqueles que deveriam ser raramente encontrados na estrutura do hiperdocumento. Entretanto, a contagem de *links* pertencentes ao caso 7 foi maior do que o esperado, devido à existência de muitos nós cuja autoria se assemelha a de referências cruzadas utilizada em extensos documentos impressos. Isso, de certa forma, aponta para a existência de documentos inadequados para leitura em tela.

- Os *links* de uso-contextual (casos 1, 2, 5 e 6) são menos comuns na estrutura do hiperdocumento com relação aos *links* dos casos da classe de uso-geral, porém duas tendências podem ser evidenciadas com relação ao seu emprego:
 - dada a reutilização de objetos (nós ou páginas, no caso) ser uma premissa importante quando da modularidade de informações (menos duplicidade de informações), o aumento de *links* contextuais, quando bem empregados, sugere uma indicação de boa qualificação no hiperdocumento;
 - por outro lado, por serem causadores de ambigüidades ou inconsistências, existe a tendência de serem evitados e, conseqüentemente, terem sua freqüência diminuída. Isso, de certa forma, pode ser um indicativo que a equipe de autoria tem se preocupado com a melhoria da qualificação do hiperdocumento.

Figura 6.1 - Gráfico de barras representando a evolução dos casos em três momentos distintos.

Portanto, a contabilização da frequência dos casos de uso-contextual é, na maioria das vezes, inconclusiva, exigindo que outros fatores sejam considerados na sua análise.

Figura 6.2 - Gráfico apresentando as proporções de incidências de cada um dos casos e a evoluções nos três momentos distintos.

A Figura 6.2 apresenta o gráfico com as proporções dos resultados das incidências de cada um dos casos, obtidas em cada uma das coletas. Muito embora algumas curvas pareçam ascendentes/descendentes no decorrer do tempo, foge da capacidade de esta autora fazer conjecturas sobre o comportamento previsível de tais frequências.

6.4 Experiência controlada

A partir dos resultados contabilizados em cada caso de ocorrência de reuso, obtidos do experimento usando o hiperdocumento do ICMSC, foram apresentadas as porcentagens de suas ocorrências aos *webmasters* que gerenciam o hiperdocumento

do ICMSC, disponível no servidor local WWW. Esses autores relataram que tinham expectativas com relação à maior (ou menor) incidência de alguns dos casos. Essa expectativa refletiu, para a maioria dos casos, a situação real de reuso de *links*, evidenciada pelo experimento.

As avaliações informais acrescentaram algumas justificativas ao uso de certas classes e reforçaram uma preocupação de que algumas porcentagens poderiam ser melhoradas (diminuídas ou aumentadas, conforme seu emprego) com o objetivo de aumentar a qualidade do hiperdocumento. Nesse sentido, os *webmasters* demonstraram interesse em ter em mãos as tabelas construídas por meio das *queries* de cada um dos casos, com o objetivo de estudar melhor cada ocorrência de reuso das estruturas de *links*.

Numa breve reprodução, a seguir estão alguns comentários feitos pelos *webmasters* ao tomar conhecimento dos resultados obtidos no experimento com relação às consultas em SQL, referente a cada caso de reuso de componentes de *links*:

caso 0 “... — existem ocorrências do caso 0 pelo fato de o hipertexto em HTML possuir somente *links* unidirecionais. Como as páginas são apresentadas por sistemas que possuem botões de volta (*backtrack*), como Mosaic e Netscape, muitas delas não possuem nenhum *link* de retorno, e dessa forma não possuem reuso. Mas, geralmente são evitadas tais ocorrências”.

caso 1 “... — muitos desses links podem ser redefinidos para serem mais uniformes e conseqüentemente se enquadrarão no caso 3. Entretanto, alguns casos se referem a reuso proposital de conteúdos de página-destino”.

caso 2 “... — o reuso de nome de *links* (âncoras-fonte) como visto no caso 2 é bastante útil, quando indica uma orientação seqüencial, ou então, quando o autor quer que o leitor siga um trajeto e não fuja muito de um roteiro, como no caso de notas de aulas. Porém, quando se está visitando o hiperdocumento em busca de uma informação específica,

que aparece identificando um *link*, e se depara com esse nome diversas vezes durante o caminho de busca, esse reuso favorece a desorientação”.

caso 3 “... — quanto maior a porcentagem de incidência do caso 3, melhor, pois demonstra uniformidade de termos”.

caso 4 “... — a ocorrência bastante alta do caso 4 é devido às páginas-índice. Seria interessante um estudo de distribuição da quantidade de *links* pela quantidade de registros distintos em que ocorrem”.

caso 5 “... — a incidência de *links* distintos entre mesmas páginas deve ser evitada. A maioria das ocorrências que apareceram foram construídas à revelia da gerência dos *webmasters* e se configuraram como testes de autores novatos utilizando as páginas de WWW como intercâmbio de informações, como acontece com o correio eletrônico”.

caso 6 “... — as ocorrências do caso 6 são provenientes de páginas transcritas de textos lineares, algumas vezes herdados de impressos que possuem reuso de palavras para evitar uma sobrecarga de termos. São evitados, no caso 6, ocorrências do tipo “clique aqui” que levam a diferentes informações, porém não se pode omitir casos de tabelas de “Preços” que são endereçadas de diferentes pontos ou ainda diferentes versões para o nó-destino, como é o caso de páginas-destino em diferentes idiomas”.

caso 7 “... — é geralmente evitado o caso 7 quando os *links* idênticos estão próximos, para não ficar uma leitura “poluída”. Os exemplos típicos em que ocorrem com mais frequência são os hipertextos provenientes de documentos impressos como os manuais, onde a frequência de páginas extensas se deve à adequação ao documento estruturado em papel”.

Concluindo, pode-se observar das averiguações feitas junto aos *webmasters*, que as incidências de *links* nos casos de reuso resultantes do experimento realmente possuíam um caráter de avaliação que lhes interessava e sobre o qual queriam

investigar as devidas justificativas. Além disso, a porcentagem significativa dos casos indicando possíveis anomalias, em sua maioria, exigiam manutenções corretivas no hiperdocumento.

Capítulo 7

Conclusões

As tentativas em se estabelecer parâmetros que indiquem o nível de qualidade de hiperdocumentos têm atualmente esbarrado em várias dificuldades. A complexidade na identificação de métricas significativas para a avaliação de hiperdocumentos tem sido agravada pela defasagem que existe entre a tecnologia disponível para sistemas hipertexto e sua fundamentação teórica. Nos últimos anos isto vem sendo intensificado em virtude da globalização dos hiperdocumentos da *World-Wide Web*.

A partir da proposta de uma sistemática para análise relacional de reuso de *links*, da construção de um sistema protótipo LiOS que a implementa e da utilização do LiOS em hiperdocumentos da WWW, o trabalho de pesquisa descrito nesta tese atingiu o seu objetivo primeiro que é o de formalizar um mecanismo para auxiliar na avaliação de hiperdocumentos.

Este capítulo resume as atividades desenvolvidas durante a pesquisa e apresenta suas principais contribuições. Além disso, são descritas algumas limitações detectadas e possíveis linhas para pesquisa futura.

7.1 Atividades realizadas

Este trabalho contribui para a redução de problemas inerentes aos sistemas hipertexto — relacionados à desorientação do usuário e sobrecarga cognitiva —, uma vez que investe em avaliação de hiperdocumentos visando fornecer elementos para identificação de possíveis reorganizações de suas informações.

A construção de sistemas hipertextos e de hiperdocumentos nas mais diversas áreas de aplicação tem se difundido amplamente nos últimos anos. A tecnologia de hipertextos, que serve como um meio mais flexível para tornar disponível as informações aos usuários de aplicativos, está evoluindo rapidamente. Nota-se, entretanto, que tal evolução acontece principalmente ao nível do desenvolvimento de aplicações e suas implementações.

A revisão da literatura relativa aos conceitos relevantes da área de hipertexto permitiu a identificação dos principais processos relacionados a um sistema hipertexto, bem como a identificação de sua base de informações. Essa revisão possibilitou ainda, saber que a motivação para a definição dos principais modelos de sistema hipertexto foi o desenvolvimento de um elevado número de aplicativos, utilizando a tecnologia de hipertextos. A partir das arquiteturas propostas pelos modelos de sistemas hipertexto, foi identificada a granularidade do domínio de informações referente à estruturação dos *links* de um hiperdocumento, determinando, assim, a abrangência dos dados de interesse à pesquisa desenvolvida neste trabalho. Além disso, as diferentes formas de representação das informações estruturais de um hiperdocumento (de seus nós e *links*) são discutidas, visando a adoção de uma que forneça os recursos descritivos às características estruturais de um hiperdocumento, ressaltadas nesta tese.

Devido principalmente ao seu relevante papel e ao fato de ser relativamente nova, a tecnologia de hipertextos se mostra em franca evolução. Este trabalho aponta essa evolução apresentando as principais tendências das pesquisas relacionadas com a melhoria na qualidade de hipertextos. São abordados trabalhos que propõem recursos melhores para navegação, pesquisas relacionadas à análise de conteúdo dos nós e são

discutidas as principais pesquisas sobre a avaliação de hiperdocumentos. Finalmente, duas tendências evolutivas foram identificadas: aquela realizada por meio de uma abordagem de Engenharia de Hipertexto, durante o processo de autoria e a realizada pela modelagem dos hiperdocumentos.

No trabalho desenvolvido foram atendidas as duas linhas de evolução identificadas. A proposta elaborada para a avaliação da estruturação dos *links* de hiperdocumento apóia tanto o processo de autoria, pois fornece uma sistemática de exame dos *links* criados, quanto auxilia nos trabalhos de modelagem, pois fornece mecanismos para a identificação de abstrações existentes no hiperdocumento.

Adotando-se a hipótese que as características estruturais do hiperdocumento, bem como seus processos de autoria e de manutenção podem fornecer as informações relevantes para uma avaliação de sua qualidade, foi proposta uma sistemática para exame de hiperdocumentos. Essa sistemática se insere no Ciclo de Vida por Prototipação Evolutiva para o desenvolvimento de hiperdocumentos, descrito neste trabalho, cujas fases são: análise de requisitos, autoria, exame e reorganização.

Durante a fase de exame do hiperdocumento, quando se faz avaliações da organização do hiperdocumento criado, foram identificadas as abordagens relacional e recursiva para análise da estrutura do hiperdocumento. Para coletar os dados para exame, com o objetivo de monitorar a dinâmica dos *links* definidos no hiperdocumento, foi utilizado um ambiente de simulação, StatSim, com uma adequação da interface e devidamente customizado para suportar um maior volume de dados.

Uma análise do hiperdocumento que auxilie a atividade de exame e conduza à melhoria de sua qualidade exige do autor uma visão completa do hiperdocumento a partir de várias perspectivas. Muitas vezes, tal visão é impossível de ser obtida sem o auxílio de algum procedimento sistemático. Nesta tese é proposta uma abordagem orientada a *links* como um auxílio sistemático para avaliação quanto aos critérios de reuso e de consistência, fundamentada na análise relacional.

Buscando coletar as informações relevantes para os critérios de reuso e consistência, foram definidos os casos de reuso de componentes de *links*, passíveis de

avaliação à luz da análise relacional, durante a fase de exame de um hiperdocumento. Foi proposta, então, uma classificação para os casos de reuso de *links*: de uso geral, particular ou contextual. Tais casos de reuso de *links* foram formalmente especificados e investigados de maneira sistemática com relação às possíveis abstrações derivadas. Nesse contexto, os higráficos são propostos como um formalismo adequado para representação estrutural de hiperdocumentos, por ser uma combinação dos conceitos de hipergráficos e de conjuntos, e por viabilizar a descrição das estruturas derivadas obtidas da identificação das abstrações por reuso de componentes de *links*.

A abordagem definida na análise relacional de *links*, a partir da qual foi proposta a classificação de reuso de componentes de *links*, possibilitou a construção de um sistema protótipo. Esse sistema, denominado Sistema Orientado-a-Links (*Link-Oriented System* ou LiOS), consiste de um suporte automático à abordagem de análise relacional definida e é uma comprovação da aplicabilidade de tal análise, uma vez que foi utilizado em experimento de avaliação de hiperdocumento da WWW.

As informações relativas à avaliação coletadas pelo LiOS se enquadram entre dois níveis de granularidade, o de conteúdo e o de estrutura de um hiperdocumento, pois se referem à ancoragem dos *links* do hiperdocumento. Finalmente, os resultados, obtidos da aplicação do LiOS no hiperdocumento da WWW, revelam sua importância e seu potencial, para coleta de informações para a nova métrica relativa ao reuso de componentes estruturais em hiperdocumento.

Apesar do objetivo principal deste trabalho não ter sido o de verificar as falhas cometidas durante o processo de autoria de um hiperdocumento, constatou-se informalmente que diversas ocorrências de casos de reuso estrutural de páginas da WWW eram devidas a falhas de uniformidade no emprego dos termos para ancoragem. De fato, a integração da abordagem de LiOS a um sistema de autoria de hiperdocumentos da WWW é prevista como um auxílio para avaliações alternativas.

As atividades desenvolvidas durante a realização deste trabalho serviram de base para a publicação de sete artigos: avaliação automática de hiperdocumentos (Fortes, 1995); modelagem de hipertextos (Fortes; Garcia Neto, 1995a); utilização de *Statechart* na modelagem de hiperdocumentos para facilitar seu exame (Fortes; Garcia

Neto, 1995b); formalização da abordagem de avaliação por reuso de *links* (Fortes et al., 1996); descrição do sistema de suporte LiOS (Fortes, 1996 e Fortes; Garcia Neto, 1996) e uso de LiOS na avaliação de um hiperdocumento (Fortes et al., 1996).

7.2 Principais contribuições

Esse trabalho formaliza alguns dos principais conceitos relacionados à avaliação de um hiperdocumento. Esse esforço de formalização contribuiu para a proposta de uma sistemática de avaliação de hiperdocumentos, baseada em sua estruturação, e na implementação de um sistema protótipo que implementa tal sistemática.

Esse resultado contribui principalmente para duas áreas de pesquisa, devido ao caráter interdisciplinar da proposta da abordagem sistemática para avaliação de hiperdocumentos:

- A identificação do Ciclo de Vida por Prototipação Evolutiva, como um modelo para o desenvolvimento de hiperdocumentos, fornece uma sistemática prescritiva importante à área de pesquisa em Engenharia de Hipertextos.
- A extração da estrutura de um hiperdocumento para ser examinada, por meio de uma análise relacional dos reusos de componentes de *links*, é uma contribuição relevante à área de pesquisa em técnicas de avaliação da modelagem e estruturação das informações de um hiperdocumento. Por tratar dos dados relacionados à ancoragem dos *links*, que até o momento não possui referências de pesquisas na literatura científica, essa análise representa uma métrica original para auxiliar na indicação da qualificação estrutural de um hiperdocumento.

7.3 Trabalhos futuros

A partir das atividades realizadas neste trabalho, constatou-se que ainda existem limitações a serem contornadas o que, de certa forma, impulsiona a continuidade

desse trabalho em outras direções. É fundamental que sejam investigados os outros possíveis papéis que os componentes de um hiperdocumento possam desempenhar, que sejam passíveis de análise quanto ao reuso.

Além disso, entre as áreas evidenciadas para futuros investimentos em pesquisa, destaca-se:

- o desenvolvimento de uma ferramenta integrando os módulos especificados em LiOS, bem como a implementação do módulo "Representação do HD", responsável pela extração da estrutura do hiperdocumento, em uma linguagem de programação portátil, como Java, por exemplo, visando a avaliação de hiperdocumentos em *sites* remotos;
- o projeto de uma interface mais adequada ao usuário autor fornecendo uma visão mais objetiva dos casos de reuso de *links* que podem ser investigados;
- a exploração do ambiente StatSim para suporte de exames mais específicos incorporando a proposta de autômatos de *links* para a verificação de propriedades estruturais do hiperdocumento.

Espera-se também ampliar os resultados desta pesquisa com a condução de experimentos relativos à criação/utilização de hiperdocumentos junto aos usuários, antes e após uma reorganização das informações, subsidiada pela análise de *links*. Com base nos resultados obtidos, espera-se que algumas heurísticas possam ser vislumbradas e adotadas, com o objetivo de melhoria da qualidade do hiperdocumento. Pretende-se que o investimento em tal tipo de experimento possa ser aproveitado também no desenvolvimento de um Sistema Baseado em Conhecimento (com heurísticas próprias de cada domínio de conhecimento), articulado ao LiOS que, além de diagnosticar problemas relacionados à estruturação do hiperdocumento, deverá sugerir possíveis modificações naquela estruturação, visando uma melhor qualidade do hiperdocumento.

Uma outra linha de pesquisa promissora está relacionada à adoção da abordagem orientada a *links* na área de Engenharia de Software. Como um auxílio no desenvolvimento de programas, os *links* equivaleriam às transferências de fluxo de

controle que ocorrem entre os diversos módulos de um software. Também seria um recurso poderoso como auxílio para a aplicação da técnica de Engenharia Reversa de Software, a qual requer um exame do software, a partir sua implementação, visando a obtenção de sua representação em um nível mais alto de abstração. Nesse contexto, as abstrações derivadas da análise de reuso de componentes de *links* auxiliariam o processo de identificação dos inter-relacionamentos entre os módulos de software, com o objetivo de auxiliar a criação de uma representação mais abstrata do software.

Finalmente, não podem ser esquecidas duas tendências atuais associadas ao desenvolvimento de hiperdocumentos: a da disponibilidade de mecanismos de navegação e a utilização da linguagem de programação Java. Essas tendências têm respaldo nas áreas de Sistemas Operacionais Distribuídos e Programação Concorrente. A abordagem orientada a *links* proposta nesta tese pode fornecer uma base teórica importante para o tratamento dos problemas já conhecidos naquelas áreas.

Referências bibliográficas

- AKSCYN, R.M.; McCracken, D.L.; Yoder, E.A. "KMS: A distributed Hypermedia System for managing knowledge in organizations" *Communications of the Association of Computer Machinery*, v.31, n.7, p.820-35, July 1988.
- APPLE COMPUTER *HyperCard Stack Design Guidelines*. Apple Computer Inc., Cupertino, CA, 1989.
- ARENTS, H.C.; BOGAERTS, W.F.L. "Towards an architecture for third-order Hypermedia Systems." *Hypermedia*, v.3, n.2, p.133-52, 1991.
- BATISTA, J.E.S. *Um Editor Gráfico para Statecharts*. Dissertação de Mestrado, ICMSC-USP, São Carlos, 1991.
- BEGORAY, J.A. "An Introduction to hypermedia issues, systems and application areas" *International Journal Man-Machine Studies*, v.33, p.121-47, 1990.
- BERGE, C. *Graphs and Hypergraphs*. New York, North-Holland Publishing Company, 1973.
- BERNERS-LEE, T. "Hypertext Markup Language (HTML)" January 1993. URL = <ftp://info.cern.ch/pub/www/doc/html-spec.ps> or <http://info.cern.ch/hypertext/WWW/MarkUp/MarkUp.html>
- BERNERS-LEE, T. et al. "The World-Wide Web" *Communications of the Association of Computer Machinery*, v.37, n.8, p.76-82, August 1994.
- BERNSTEIN, M. "An Apprentice That Discovers Hypertext Links" In: RIZS, A.; STRITZ, N. & ANDRÉ, J., eds. *Hypertext: Concepts, Systems and Applications* Cambridge, Great Britain, Cambridge University Press, 1990. /*Proceedings of ECHT'90 - European Conference on Hypertext, Versailles, France, November 1990.*/ p.212-23.
- BOAVENTURA, I.A.G. *Propriedades Dinâmicas de Statecharts*. Dissertação de Mestrado, ICMSC-USP, São Carlos, 1992.
- BOTAFOGO, R.A.; SHNEIDERMAN, B. "Structural Analysis of Hypertexts: Identifying Hierarchies and Useful Metrics" *ACM Transactions on Information Systems*, v.10, n.2, p.142-80, April 1992.

- BROWN, P.J. "Turning ideas into products: The Guide System" In: ACM Hypertext'87, Chapel Hill, NC, November 1987. *Proceedings*. New York, ACM Press, 1987. p.33-40.
- BROWN, H. "Standards for Structured Documents." *The Computer Journal*, v.32, n.6, p.505-14, 1989.
- BROWN, P.J. "Assessing the quality of Hypertext Documents" In: RIZS, A.; STRITZ, N. & ANDRÉ, J., eds. *Hypertext: Concepts, Systems and Applications* Cambridge, Great Britain, Cambridge University Press, 1990. /*Proceedings of ECHT'90 - European Conference on Hypertext*, Versailles, France, November 1990./ p.1-12.
- CAMPBELL, B.; GOODMAN, J.M. "HAM: A General Purpose Hypertext abstract machine" *Communications of the Association of Computer Machinery*, v.31, n.7, p.856-61, July 1988.
- CANGUSSU, J.W.L. *Execução Programada de Statecharts*. Dissertação de Mestrado, ICMSC-USP, São Carlos, 1993.
- CASANOVA, M.A. et al. "The Nested Context Model for Hyperdocuments" In: Hypertext'91, San Antonio, TX, December 1991. *Proceedings*. New York, ACM Press, 1991. p.193-201.
- CLARKE, E. M.; EMERSON, E. A.; SISTLA, A. P. "Automatic Verification of Finite-State Concurrent Systems Using Temporal Logic Specifications" *ACM Transaction on Programming Languages and Systems*, v.8, n.2, p.244-63, April 1986.
- COLEMAN, D. et al. *Object-Oriented development: The FUSION Method*. New Jersey, USA, Prentice Hall, Englewood Cliffs, 1994.
- CONKLIN, J. *A Survey of Hypertext*. Austin, TX, December 1987a. (MCC Technical Report STP-356-86, Revision 2, Microelectronics and Computer Technology Corporation)
- CONKLIN, J. "Hypertext: An Introduction and Survey" *Computer*, v.20, n.9, p.17-41, September 1987b.
- CONKLIN, J.; BEGEMAN, M.L. "gIBIS: A hypertext tool for argumentation" *ACM Transactions on Information Systems*, v.6, n.4, p.303-331, July 1988.
- CREECH, M.L.; FREEZE, D.F.; GRISS, M.L. "Using Hypertext in Selecting Reusable Software Components" In: Hypertext'91, San Antonio, TX, December 1991. *Proceedings*. New York, ACM Press, 1991. p.25-38.
- DELISLE, N. ; SCHWARTZ, M. "Neptune: A Hypertext system for CAD applications" In: ACM SIGMOD'86, Washington, D.C. , May 1986. *Proceedings*. New York, ACM Press, 1986. p.132-142.
- DEROSE, S.J.; DURAND, D.G. *Making Hypermedia Work - A User's Guide to HyTime*. Kluwer Academic Press, Inc., Massachusetts, USA, 384p., 1994.
- FEINER, S. "Seeing the Forest for the Trees: Hierarchical Display of Hypertext Structure" In: Conference on Office Information Systems, Palo Alto, CA, March 1988. *Proceedings*. New York, NY, ACM Press, 1988. p.205-12.

- FORTES, R.P.M. *Uma Ferramenta de Apoio à Utilização de Statecharts para Especificação do Comportamento de Sistemas de Tempo Real Complexos*. Tese de Mestrado, ICMSC-USP, São Carlos, 1991.
- FORTES, R.P.M.; MASIERO, P.C. *Uma Linguagem para Especificação de Statecharts e seu Analisador*. Documento de Trabalho, ME-91/01, ICMSC-USP, 1991.
- FORTES, R.P.M. "Avaliação Automática de Hiperdocumentos" In: I Workshop de Sistemas Hipermídia Distribuídos, São Carlos-SP, Brasil, 7-8 julho de 1995. *Anais*. 1995. p.111-22.
- FORTES, R.P.M.; GARCIA NETO, A. "Modelagem de Hipertextos". In: 2º Simpósio Interno da Pós-Graduação do IFSC, Instituto de Física de São Carlos-USP, São Carlos-SP, abril de 1995. *Resumos*. 1995a. p.48.
- FORTES, R.P.M.; GARCIA NETO, A. "Modelling Hyperdocuments with Statecharts for Easier Perusal" In: The 8th Annual Conference on Writing and Computers, London, U.K., September 7-9, 1995. *Abstracts*. 1995b. p.53.
- FORTES, R.P.M. "Uma Ferramenta orientada a links para Avaliação de Hiperdocumentos" São Carlos, ICMSC-USP, Brasil, março de 1996. (Notas do ICMSC Série Computação, 24).
- FORTES, R.P.M.; GARCIA NETO, A.; NICOLETTI, M.C. "A Formal Approach to Consistency and Reuse of Links in Hypermedia Applications" In: ACM CHI'96 Workshop on Formal Methods in Computer Human Interaction, Vancouver, Canada, April 14-15, 1996. *Informal Proceedings*. 1996.
- FORTES, R.P.M.; GARCIA NETO, A. "A Link-oriented Tool for Evaluation of Hyperdocuments" In: Symposium - The Missing Link: Hypermedia Usability Research & The Web, Knowledge Media Institute (KMI) - The Open University, London, U.K., May 1, 1996.
- FORTES, R.P.M.; GARCIA NETO, A.; NASCIMENTO JR., C.S. "Avaliação de Hiperdocumento através de uma ferramenta orientada a links" In: II Workshop de Sistemas Hipermídia Distribuídos, Fortaleza-CE, Brasil, 18-19 maio de 1996. *Anais*. 1996. p.31-40.
- FURNAS, G.W. "Generalized fisheye views" In: ACM CHI'86 *Human Factors in Computing Systems*, Boston, April 1986. *Proceedings*. 1986. p.16-23.
- GARG, P.K. "Abstraction Mechanisms in Hypertext" *Communications of the Association of Computer Machinery*, v.31, n.7, p.862-70, July 1988.
- GARRET, L.N.; SMITH, K.E.; MEYROWITZ, N. "Intermedia: Issues, Strategies and Tactics in the Design of a Hypermedia Document System" In: Conference on Computer-Supported Cooperative Work, 1., Austin, TX, December 1986. *Proceedings*. 1986. p.163-74.
- GARZOTTO, F.; PAOLINI, P.; SCHWABE, D. "HDM - A Model for the Design of Hypertext Applications" In: Hypertext'91, San Antonio, TX, December 1991. *Proceedings*. New York, ACM Press, 1991a. p.313-28.
- GARZOTTO, F. et al. "Tools for Designing Hyperdocuments" In: BERK, E. & DEVLIN, J. eds. *Hypertext/Hypermedia Handbook* New York, Intertext Publications, McGraw-Hill, 1991b. p.179-207.

- GARZOTTO, F.; PAOLINI, P.; SCHWABE, D. "HDM - A Model-Based Approach to Hypertext Application Design" *ACM Transactions on Information Systems*, v.11, n.1, p.1-26, January 1993.
- GARZOTTO, F.; MAINETTI, L.; PAOLINI, P. "Hypermedia Design, Analysis, and Evaluation Issues" *Communications of the Association of Computer Machinery*, v.38, n.8, p.74-86, August 1995.
- GLUSHKO, R.J. et al. "Hypertext engineering: practical methods for creating a compact disc encyclopedia" In: Conference on Document Processing Systems, December 1988. *Proceedings*. ACM Press, 1988. p.11-9.
- GRONBAEK, K.; TRIGG, R.H. "Design issues for a Dexter-based Hypermedia System" *Communications of the Association of Computer Machinery*, v.37, n.2, p.41-9, February 1994.
- HALASZ, F.G. "Reflections on Notecards: Seven Issues for the next generation of Hypermedia Systems" *Communications of the ACM*, v.31, n.7, p.836-52, July 1988.
- HALASZ, F.; SCHWARTZ, M. "The Dexter hypertext reference model" In: MOLINE, J.; BENIGNI, D.; BARONAS, J., eds. *Proceedings of the NIST Hypertext Standardization Workshop* Gaithersburg, January 1990. National Institute of Standards and Technology Special Publication 500-178. Washington: U.S. Government Printing Office, 1990. p.95-133.
- HALSTEAD, M.H. *Elements of Software Science*. New York, Elsevier North Holland, 1977.
- HAREL, D. "Statecharts: a Visual Formalism for Complex Systems" *Science of Comp. Programming*, 8, p.231-274, 1987a.
- HAREL, D. "Statecharts: On The Formal Semantics of Statecharts" In: *Proceedings of the 2nd IEEE Symposium on Logic in Computer Science*, Ithaca, N.Y., p.54-64, 1987b.
- HAREL, D. "On Visual Formalisms" *Communications of the Association of Computer Machinery*, v.3, n.5, p.514-30, May 1988.
- HARDMAN, L. "Hypertext Tips: Experiences in Developing a Hypertext Tutorial" In: JONES, D.M.; WINDER, R., eds. *People and Computers IV*. Cambridge UK, Cambridge University Press, 1988. p.437-51.
- HARDMAN, L.; BULTERMAN, D.C.A.; VAN ROSSUM, G. "Links in Hypermedia: the requirement for Context" In: Hypertext'93, Seattle, Washington USA, November 1993. *Proceedings*. New York, ACM Press, 1993a. p.183-191.
- HARDMAN, L.; BULTERMAN, D.C.A.; VAN ROSSUM, G. "The Amsterdam Hypermedia Model: extending hypertext to support *real* multimedia" *Hypermedia*, v.5, n.1, p.47-69, July 1993b.
- HARDMAN, L.; BULTERMAN, D.C.A.; VAN ROSSUM, G. "The Amsterdam Hypermedia Model: Adding Time and Context to the Dexter Model" *Communications of the Association of Computer Machinery*, v.37, n.2, p.50-62, February 1994.

- HARDMAN, L.; BULTERMAN, D.C.A. "Towards the Generation of Hypermedia Structure" In: First International Workshop on Intelligence and Multimodality in Multimedia Interfaces, Edinburgh, UK, July 1995. *Proceedings*.
- HARDMAN, L. "Experiences in Authoring Hypermedia: Creating Better Presentations" In: SCHULER, W.; HANNEMANN, J.; STREITZ, N., eds. *Designing User Interfaces for Hypermedia*, Springer 1995. p.16-26.
- HATZIMANIKATIS, A.E.; TSALIDIS, C.T.; CHRISTODOULAKIS, D. "Measuring the Readability and Maintainability of Hyperdocuments" *Software Maintenance: Research and Practice*, v.7, p.77-90, 1995.
- HERWIJNEN, E. VAN *Practical SGML*. 2.ed. Massachusetts, USA, Kluwer Academic Press, 1994.
- HEUSER, C.A. *Modelagem Conceitual de Sistemas - Redes de Petri*. V EBAI, Nova Friburgo, RJ, Brasil, 1991.
- INTERNATIONAL ORGANIZATION FOR STANDARDIZATION ISO 8879 *Information Processing - Text and Office Systems - Standard Generalized Markup Language*. October 1986.
- INTERNATIONAL ORGANIZATION FOR STANDARDIZATION ISO 8613 *Information Processing - Text and Office Systems - Office Document Architecture (ODA) and Interchange Format*. 1989.
- INTERNATIONAL ORGANIZATION FOR STANDARDIZATION ISO/IEC 88744 *Hypermedia/Time-Based Structuring Language (HyTime)*. April, 1992.
- ISAKOWITZ, T.; STOHR, E.A.; BALASUBRAMANIAN, P. "RMM: A Methodology for Structured Hypermedia Design" *Communications of the ACM*, v.38, n.8, p.34-44, August 1995.
- JAHANIAN, F.; MOK, A.K. "Modechart: A Specification Language for Real-Time Systems" *IEEE Transactions on Software Engineering*, v.20, n.12, p.933-947, December 1994.
- JOHNSON, D.S.; POLLAK, H.O. "Hypergraph Planarity and the Complexity of Drawing Venn Diagrams" *Journal of Graph Theory*, v.11, n.3, p.309-25, 1987.
- KAN, S.H. *Metrics and Models in Software Quality Engineering*. USA, Addison-Wesley, 1995.
- JURAN, J.M.; GRZYNA Jr, F.M. *Quality Planning and Analysis: From Product Development Through Use*. New York USA, McGraw-Hill, 1970.
- LUTHER, A.C. *Authoring Interactive Multimedia*. Academic Press, London, UK, 1994.
- MÄKINEN, E. "How to Draw a Hypergraph" *International Journal of Computer Mathematics*, v.34, p.177-85, 1990.
- MARSHALL, C.C. et al. "Aquanet: a hypertext tool to hold your knowledge in place" In: Hypertext'91, San Antonio, TX, December 1991. *Proceedings*. New York, ACM Press, 1991. p.261-276.
- MARSHALL, C.C.; SHIPMAN, F.M. "Searching for the Missing Link: Discovering Implicit Structure in Spatial Hypertext" In: Hypertext'93, Seattle, Washington USA, November 1993. *Proceedings*. New York, ACM Press, 1993. p.217-30.

- MASIERO, P.C.; FORTES, R.P.M.; BATISTA NETO, J.E.S. "Edição e Simulação do Aspecto Comportamental de Sistemas de Tempo Real" In: XVIII SEMISH, Santos-SP, setembro de 1991. *Anais.* p.45-61.
- MUCHALUAT, D.C.; SOARES, L.F.G. "Browsers e Trilhas no Sistema HyperProp" In: I Workshop em Sistemas Hipermídia Distribuídos. São Carlos, SP, julho de 1995. *Anais.* p.58-66.
- NANARD, J.; NANARD, M. "Hypertext Design Environments and the Hypertext Design Process" *Communications of the Association of Computer Machinery*, v.38, n.8, p.49-56, August 1995.
- NEWCOMB, S.R.; KIPP, N.A.; NEWCOMB, V.T. "The "HyTime" Hypermedia / Time-based Document Structuring Language" *Communications of the Association of Computer Machinery*, v.34, n.11, p.67-83, November 1991.
- NIELSEN, J. *Hypertext and Hypermedia*. Academic Press, San Diego, CA-USA, 1990a.
- NIELSEN, J. "The Art of Navigating through Hypertext" *Communications of the Association of Computer Machinery*, v.33, n.3, p.296-310, March 1990b.
- NIELSEN, J. *Multimedia and Hypertext - The Internet and Beyond*. Academic Press, London, United Kingdom, 1995.
- OLIVEIRA, M.C.F.; TURINE, M.A.S.; MASIERO, P.C.M. "A Statechart - based Model for Hypertext" São Carlos, ICMSC-USP, Brasil, 1995. (*Notas do ICMSC Série Computação*, 19).
- PARUNAK, H.Van D. "Don't Link Me In: Set Based Hypermedia for Taxonomic Reasoning" In: Hypertext'91, San Antonio, TX, December 1991. *Proceedings*. New York, ACM Press, 1991. p.233-42.
- PEREZ, E. "Tools for Authoring Hypertexts" In: BERK, E. & DEVLIN, J. eds. *Hypertext/Hypermedia Handbook* New York, Intertext Publications, McGraw-Hill, 1991. Chapter 14, p.209-224.
- PRESSMAN, R.S. *Software Engineering - A Practitioner's Approach*. McGraw Hill Book Company Europe, London, United Kingdom, 1994.
- QUINT, V.; VATTON, I. "Modularity in structured documents" In: ANDRÉ, J. & J. BÉZIVIN, J., eds. *WOODMAN' 89*, Bigre n.63-64, IRISA, Rennes, mai 1989. p.170-7.
- RADA, R. *Interactive Media*. Springer-Verlag, New York, USA, 1995.
- RAYMOND, D.R.; TOMPA, F.W.M. "Hypertext and the Oxford English Dictionary" *Communications of the Association of Computer Machinery*, p.871-9, July 1988.
- RIVLIN, E.; BOTAFOGO, R.; SHNEIDERMAN, B. "Navigating in Hyperspace: Designing a Structure-based toolbox" *Communications of the Association of Computer Machinery*, v.37, n.2, p.87-96, February 1994.
- SALMINEN, A.; TAGUE-SUTCLIFFE, J.; MCCLELLAN, C. "From Text to Hypertext by Indexing" *ACM Transactions on Database Systems*, v.13, n.1, p.69-99, January 1995.

- SALTON, G.; ALLAN, J.; BUCKLEY, C. "Automatic Structuring and Retrieval of Large Text Files" *Communications of the Association of Computer Machinery*, v. 37, .2, p.97-108, February 1994.
- SCHÜTT, H.; STREITZ, N. "HyperBase: a hypermedia engine based on a relational database management system" In: RIZS, A.; STRITZ, N. & ANDRÉ, J., eds. *Hypertext: Concepts, Systems and Applications* Cambridge, Great Britain, Cambridge University Press, 1990. /*Proceedings of ECHT'90 - European Conference on Hypertext, Versailles, France, November 1990.*/ p.95-104.
- SCHWABE, D.; ROSSI, G.; BARBOSA, S.D.J. "Systematic Hypermedia Application Design with OOHDM" In: Hypertext'96, Washington DC, USA, March 1996. *Proceedings*. New York, ACM Press, 1996. p.116-28.
- SHIBATA, Y.; KATSUMOTO, M. "Dynamic Hypertext and Knowledge Agent Systems for Multimedia Information Networks" In: Hypertext'93, Seattle, Washington USA, November 1993. *Proceedings*. New York, ACM Press, 1993. p.82-93.
- SHNEIDERMAN, B. "Reflections on Authoring, Editing and Managing Hypertext" In: BARRET, E., ed. *The Society of text: Hypertext, Hypermedia, and the Social Construction of Information* Cambridge, Massachusetts, MIT Press, 1989. p.115-31.
- SOARES, L.F.G.; CASANOVA, M.A.; RODRIGUEZ, N.L.R. "Um Modelo Conceitual Hipermídia com Nós de Composição e Controle de Versões" In: VII Simpósio Brasileiro de Engenharia de Software, Rio de Janeiro, Brasil, outubro de 1993. *Anais*. Rio de Janeiro, Gráfica da PUC-Rio, 1993b. p.365-80.
- SOARES, L.F.G. et al. "HyperProp: uma Visão Geral" In: I Workshop em Sistemas Hipermídia Distribuídos. São Carlos, SP, julho de 1995. *Anais*. p.1-12.
- STOTTS, P.D.; FURUTA, R. "Petri-Net-Based Hypertext: Document Structure with Browsing Semantics" *ACM Transactions on Information Systems*, v.7, n.1, p.3-29, January 1989.
- STOTTS, P.D.; FURUTA, R. "Dynamic Adaptation of Hypertext Structure" In: Hypertext'91, San Antonio, TX, December 1991. *Proceedings*. New York, ACM Press, 1991. p.219-30.
- STOTTS, P.D.; FURUTA, R.; RUIZ, J.C. "Hyperdocuments as Automata: Trace-based Browsing Property Verification" In: ECHT'92 - European Conference on Hypertext, Milano, December 1992. *Proceedings*. New York, ACM Press, 1992. p.272-81.
- THURING, M.; HANNEMANN, J.; HAAKE, J.M. "Hypermedia and Cognition: Designing for Comprehension" *Communications of the Association of Computer Machinery*, v.38, n.8, p.57-66, August 1995.
- TOMPA, F.W.M. "A Data Model for Flexible Hypertext Database Systems" *ACM Transactions on Information Systems*, v.7, n.1, p.85-100, January 1989.
- TUTUMI, R.; MASIERO, P.C. "Simulação Visual de Statecharts" In: X Congresso de Iniciação Científica e Tecnológica em Engenharia, São Carlos, 1991. *Anais*. São Carlos, CICTE, 1991. p.669.

- UTTING, K.; YANKLOVICH, N. "Context and Orientation in Hypermedia Networks" *ACM Transactions on Information Systems*, v.7, n.1, p.58-84, January 1989.
- VAN ROSSUM, G. et al. "CMIFed: A Presentation Environment for Portable Hypermedia Documents" In: First International Conference on Multimedia, Anaheim, CA, August 1993. *Proceedings*. New York, ACM Press, 1993. p.183-8.
- WALKER, J. "Authoring Tools for Complex Document Sets" In: BARRET, E., ed. *The Society of Text - Hypertext, Hypermedia, and the Social Construction of Information* Cambridge, Massachusetts, MIT Press, 1989. p.132-47.
- WANG, B.; HITCHCOCK, P.; HOLDEN, T. "An Object Oriented Database Approach for Supporting Hypertext" *LNCS*, p.601-28, 1992.
- WATTERS, C.; SHEPHERD, M.A. "A Transient Hypergraph-Based Model for Data Access" *ACM Transactions on Information Systems*, v.8, n.2, p.77-102, April 1990.
- YANKLOVICH, N. et al. "Intermedia: The Concept and the Construction of a Seamless Information Environment" *IEEE Computer*, n.21, v.1, p.81-96, January 1988.
- YOUNG, L. "Linking Considered Harmful" In: RIZS, A.; STRITZ, N. & ANDRÉ, J., eds. *Hypertext: Concepts, Systems and Applications* Cambridge, Great Britain, Cambridge University Press, 1990. /*Proceedings of ECHT'90 - European Conference on Hypertext, Versailles, France, November 1990.*/ p.238-49.
- ZHENG, Y.; PONG, M. "Using Statecharts to model Hypertext" In: ECHT'92 - European Conference on Hypertext, Milano, December 1992. *Proceedings*. New York, ACM Press, 1992. p.242-50.

Apêndice A

Gramática da Linguagem de Especificação de Statecharts (LES)

$\langle \text{estadograma} \rangle ::= [\langle \text{declaracoes} \rangle] \text{ ',' } \langle \text{estados} \rangle \text{ ',' } \langle \text{transicoes} \rangle \text{ ',' }$

$\langle \text{declaracoes} \rangle ::= \langle \text{declaracao} \rangle \{ \text{ ',' } \langle \text{declaracao} \rangle \}$

$\langle \text{declaracao} \rangle ::= \langle \text{tipo-id} \rangle \langle \text{lista-identificadores} \rangle$

$\langle \text{tipo-id} \rangle ::= (\text{ int } | \text{ bool } | \text{ real } | \text{ string } | \text{ cond })$

$\langle \text{lista-identificadores} \rangle ::= \langle \text{identificador} \rangle \{ \text{ ',' } \langle \text{identificador} \rangle \}$

$\langle \text{estados} \rangle ::= \langle \text{descricao estado} \rangle \{ \text{ ',' } \langle \text{descricao estado} \rangle \}$

$\langle \text{descricao estado} \rangle ::= \text{ estado } \langle \text{estado} \rangle \langle \text{tipo} \rangle$

$\langle \text{tipo} \rangle ::= \text{ ou } \langle \text{subestados ou} \rangle | \text{ e } \langle \text{subestados e} \rangle | \text{ atomo}$

$\langle \text{subestados ou} \rangle ::= \text{ subestados } \langle \text{estado} \rangle [(\text{ def } | \text{ def/H } | \text{ def/H* })] \{ \text{ ',' } \langle \text{estado} \rangle \}$

$\langle \text{subestados e} \rangle ::= \text{ subestados } \langle \text{estado} \rangle \{ \text{ ',' } \langle \text{estado} \rangle \}$

$\langle \text{transicoes} \rangle ::= \langle \text{transicao} \rangle \{ \text{ ',' } \langle \text{transicao} \rangle \}$

$\langle \text{transicao} \rangle ::= \text{ evento } \langle \text{evento} \rangle \{ \text{ ',' } \langle \text{evento} \rangle \}$
 $\text{origem } \langle \text{estado} \rangle \{ \text{ ',' } \langle \text{estado} \rangle \}$
 $\text{destino } \langle \text{estado} \rangle [(/ \text{H} | / \text{H*})] \{ \text{ ',' } \langle \text{estado} \rangle [(/ \text{H} | / \text{H*})] \}$
 $[\text{acao } \{ \text{ ',' } \langle \text{acoes} \rangle \text{ ',' } \}]$

$\langle \text{evento} \rangle ::= \langle \text{identificador} \rangle \mid \langle \text{true} \rangle ' (\langle \text{exp} \rangle ')' \mid \langle \text{false} \rangle ' (\langle \text{exp} \rangle ')'$
 $\mid \langle \text{changed} \rangle ' (\langle \text{exp} \rangle ')' \mid \langle \text{exit} \rangle ' (\langle \text{estado} \rangle ')'$
 $\mid \langle \text{entered} \rangle ' (\langle \text{estado} \rangle ')' \mid \langle \text{evento} \rangle \text{ or } \langle \text{evento} \rangle$
 $\mid \langle \text{evento} \rangle \text{ and } \langle \text{evento} \rangle \mid \langle \text{evento} \rangle ' [\langle \text{exp} \rangle ']'$
 $\mid \text{t_out} \{ ' (\langle \text{numero} \rangle \langle \text{unid-tempo} \rangle ')'$

$\langle \text{exp} \rangle ::= \langle \text{expS} \rangle [(= | > | < | \diamond | \leq | \geq) \langle \text{expS} \rangle]$

$\langle \text{expS} \rangle ::= \langle \text{termo} \rangle \{ (+ | - | \text{or}) \langle \text{termo} \rangle \}$

$\langle \text{termo} \rangle ::= \langle \text{fator} \rangle \{ (* | / | \text{and}) \langle \text{fator} \rangle \}$

$\langle \text{fator} \rangle ::= \langle \text{true} \rangle \mid \langle \text{false} \rangle \mid \langle \text{condicao primitiva} \rangle \mid \langle \text{numero} \rangle \mid \langle \text{variavel} \rangle$
 $\mid \text{in} ' (\langle \text{estado} \rangle ')' \mid \langle \text{not yet} \rangle ' (\langle \text{evento} \rangle ')'$
 $\mid \langle \text{current} \rangle ' (\langle \text{exp} \rangle ')' \mid \text{not } \langle \text{exp} \rangle \mid (\langle \text{exp} \rangle)$

$\langle \text{acoes} \rangle ::= \langle \text{acao} \rangle \{ ' ; ' \langle \text{acao} \rangle \}$

$\langle \text{acao} \rangle ::= \langle \text{identificador} \rangle \mid \langle \text{variavel} \rangle ' := ' \langle \text{exp} \rangle$
 $\mid \text{clear} ' (\langle \text{estado} \rangle [' * '] ')'$

$\langle \text{estado} \rangle ::= \langle \text{identificador} \rangle \mid \langle \text{identificador} \rangle ' . ' \langle \text{identificador} \rangle$

$\langle \text{variavel} \rangle ::= \langle \text{identificador} \rangle$

Símbolos terminais :

$\langle \text{true} \rangle ::= ([\text{tT}] [\text{rR}] [\text{uU}] [\text{eE}] \mid \text{tr})$

$\langle \text{false} \rangle ::= ([\text{fF}] [\text{aA}] [\text{lL}] [\text{sS}] [\text{eE}] \mid \text{fs})$

$\langle \text{changed} \rangle ::= (\text{changed} \mid \text{ch})$

$\langle \text{exit} \rangle ::= (\text{exit} \mid \text{ex})$

$\langle \text{entered} \rangle ::= (\text{entered} \mid \text{en})$

$\langle \text{not yet} \rangle ::= (\text{not_yet} \mid \text{ny})$

$\langle \text{current} \rangle ::= (\text{current} \mid \text{cr})$

$\langle \text{identificador} \rangle ::= [\text{a} - \text{zA} - \text{Z}] [\text{a} - \text{zA} - \text{Z} 0 - 9] ^ *$

$\langle \text{numero} \rangle ::= [0 - 9] ^ +$

$\langle \text{unid-tempo} \rangle ::= (\text{seg} \mid \text{min} \mid \text{hr})$