

**UNIVERSIDADE FEDERAL DA PARAÍBA**

**CENTRO DE CIÊNCIAS E  
TECNOLOGIA**

**COORDENAÇÃO DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

**UM AMBIENTE COLABORATIVO DE  
REPRESENTAÇÃO VISUAL DO  
CONHECIMENTO PARA WEB  
(VEDA)**

**DAVID NADLER PRATA**

**CAMPINA GRANDE  
2000**

---

**David Nadler Prata**

# **Um Ambiente Colaborativo de Representação Visual do Conhecimento para Web – VEDA**

Campina Grande  
2000

## Ficha Catalográfica

P912A

PRATA, David Nadler

Um Ambiente Colaborativo de Representação Visual do Conhecimento para Web (VEDA).

Dissertação de Mestrado, Universidade Federal da Paraíba,  
Centro de Ciências e Tecnologia, Coordenação de Pós-Graduação em Informática,  
Campina Grande - Pb, Setembro de 2000.

139p. Il.

Orientador: Fábio Paraguaçu D. da Costa, Dr.

Palavras Chave:

1. Inteligência Artificial
2. Representação do Conhecimento
3. Redes Semânticas

CDU - 007.52

David Nadler Prata

# **Um Ambiente Colaborativo de Representação Visual do Conhecimento para Web (VEDA)**

Dissertação apresentada ao Curso de Mestrado em Informática da Universidade Federal da Paraíba, como requisito parcial à obtenção do título de Mestre em Informática.

Àrea de concentração: Inteligência Artificial

Projeto Capes/Cofecub

Orientador: Prof. Fábio Paraguaçu  
Universidade Federal de Alagoas

Campina Grande  
Universidade Federal da Paraíba  
2000

A minha esposa Angela  
e aos meus filhos  
Brenda e Gabriel.

---

## AGRADECIMENTOS

A todos que colaboraram para que eu pudesse almejar esse trabalho, meus mais sinceros agradecimentos. Gostaria de fazer um agradecimento especial a algumas pessoas que contribuíram na realização desse trabalho:

- Ao Prof. Fábio Paraguaçu, pela competente orientação, disponibilidade e acima de tudo amizade que sempre prevaleceu em nosso relacionamento.
- A minha esposa Ângela pela paciência e compreensão aos meus dias e noites dedicados a esta dissertação, além de escutar sobre minhas idéias e elucubrações, abdicando seus próprios trabalhos em meu favor. A sua colaboração em discussões sobre o tema e revisão do trabalho.
- Aos professores e funcionários da COPIN pelos valerosos ensinamentos e apoio, especialmente Marcelo Barros, Edilson Ferneda, Bernardo Lula, Jean Sallantin, Aninha, Vera, Arnaldo e Manuela.
- Ao meu irmão Rogério pela revisão dos manuscritos em Inglês.
- Aos meus sogros Alberto e Gislaine, pelo apoio e dedicação aos meus filhos.
- A todos os meus colegas, especialmente Lidiana e Germana, pelo fornecimento de material e acesso a salas e laboratórios.
- Aos amigos Fernando Willian Cruz e Almir Joaquim de Souza, pelo incentivo ao ingresso no mestrado.
- Ao Projeto Capes/Cofecub, pelo financiamento à aquisição de livros.

---

## LISTA DE FIGURAS

FIGURA 2.1 - “Maria lê enquanto João ouve” .....	06
FIGURA 2.2 - Grafos Existenciais de Peirce.....	07
FIGURA 2.4 - Frame para um carro.....	09
FIGURA 2.5 - Frame de um triângulo retângulo.....	10
FIGURA 2.6 - Uma rede de caso em profundidade.....	11
FIGURA 2.7 - Herança nula.....	13
FIGURA 2.8 - Hiper-grafo "HiGraph" de Harel.....	14
FIGURA 2.9 - Rede causal de Pearl.....	16
FIGURA 2.10 - Expansão de um nó em sua rede de definição.....	19
FIGURA 2.11 - Três sub-categorias de um reticulado de casos.....	20
FIGURA 2.12 - O produto de um reticulado de casos.....	21
FIGURA 2.13 - Um exemplo de tipo de estrutura da notação de Aik-Kaci.....	21
FIGURA 2.16 - Grafo Conceitual para: "O gato está sobre o tapete".....	25
FIGURA 3.1 - Conferência simultânea como uma ferramenta de estudo.....	30
FIGURA 3.2 - Internet Relay Chat.....	31
FIGURA 3.3 - MOO da Diversity University.....	31
FIGURA 3.4 - Um ambiente virtual.....	32
FIGURA 3.5 - Videoconferência usando netmeeting.....	33
FIGURA 3.6 - Rubricas para avaliação.....	37
FIGURA 3.7 - Conferência não-simultânea.....	38
FIGURA 3.8 - Tela de entrada do Collaboratory Notebook.....	42
FIGURA 3.9 - Conjectura proposta no Collaboratory Notebook.....	42
FIGURA 3.10 - Um comentário adicionado a base de notas do CaMILE.....	43
FIGURA 3.11 - Especificando o tipo de resposta do CaMILE.....	44
FIGURA 3.12 - Hipermídia.....	47
FIGURA 3.13 - Escola virtual de Andrew Robinson.....	49
FIGURA 3.14 - Corredor virtual.....	49
FIGURA 4.2 - Uma relação conceitual de valência três.....	57

## *Lista de Figuras*

---

FIGURA 4.3 - Exemplificação de um contexto e uma correferência.....	58
FIGURA 4.4 - Exemplo de grafo canônico.....	60
FIGURA 4.5 - Exemplo de grafo canônico.....	60
FIGURA 4.6 - Exemplo de grafo canônico.....	60
FIGURA 4.7 - Regras de cópia e simplificação.....	61
FIGURA 4.8 - Regras de restrição e expansão.....	62
FIGURA 4.9 - Regras de junção e desmembramento.....	63
FIGURA 4.10 - Aplicação da regra de restrição.....	64
FIGURA 4.11 - Aplicação da regra de restrição.....	64
FIGURA 4.12 - Aplicação da regra de simplificação.....	65
FIGURA 4.13 - Aplicação da regra de junção.....	65
FIGURA 4.14 - Categorias de nível superior de uma ontologia.....	69
FIGURA 4.15 - Grafo de representação de quatro participantes.....	70
FIGURA 4.17 - Localização das funções temáticas na ontologia.....	74
FIGURA 4.18 - Exemplo de Searle para uma figura ambígua.....	76
FIGURA 4.19 - Tipos e denotações.....	78
FIGURA 5.1 - Arquitetura de um ambiente visual de representação do conhecimento.....	84
FIGURA 5.2 - Interface de Edição e Comunicação Simultânea.....	85
FIGURA 5.3 - Grafo Conceitual.....	87
FIGURA 5.4. - Notação para representação dos gráficos conceituais em Web.....	88
FIGURA 5.5 - Execução de uma pesquisa no Ambiente de Busca.....	89
FIGURA 5.6 - Navegação na Interface de Navegação.....	90
FIGURA 5.7 - Navegação na Interface de Navegação. Criação de um ponteiro.....	90
FIGURA 5.8 - WeContract Université Montpllier.....	93
FIGURA 5.9 - Visualização de uma Conversa Textual.....	95
FIGURA A.1 - Reticulado das categorias de nível superior.....	113
FIGURA A.2 - Classificação de entidades de acordo com as funções que elas executam.....	119
FIGURA A.3 - Três possíveis formas de representação de: "Uma bola vermelha".....	130

---

## LISTA DE TABELAS

TABELA 2.3 - Cinco níveis de ligações baseados na análise de Brachman.....	08
TABELA 2.14 - As primitivas da Dependência Conceitual.....	23
TABELA 2.15 - Slots de PTRANS.....	23
TABELA 4.1 - Exemplo de referências.....	56
TABELA 4.16 - Funções temáticas como subtipos de quatro espécies de participantes.....	71

---

# SUMÁRIO

AGRADECIMENTOS.....	VI
LISTA DE FIGURAS.....	VII
LISTA DE TABELAS.....	IX
ABSTRACT.....	XV
RESUMO.....	XVI
1 INTRODUÇÃO.....	01
1.1 Apresentação.....	01
1.2 Objetivos.....	01
1.3 Contexto.....	02
1.4 Motivação.....	03
1.7 Hipótese.....	03
1.8 Metodologia.....	03
1.9 Organização do Texto.....	03
2. REDES SEMÂNTICAS.....	05
2.1 Introdução.....	05
2.2 Evolução Histórica do Conceito de Redes Semânticas.....	05
2.3 Utilização de Redes Semânticas em Computadores.....	08
2.4 Histórico da Formalização de Redes Semânticas.....	08
2.5 Sistema de Frames.....	09
2.6 Relações de Casos Semânticos.....	10
2.7 Hierarquia de Abstrações.....	11
2.7.1 Ligações É-UM e INSTÂNCIA-DE.....	12
2.7.2 Herança Simples e Múltipla.....	12
2.7.3 Herança Nula ou Estrita.....	12
2.7.4 Herança Relacional.....	13
2.8 Extensões Lógicas.....	13
2.9 Ontologia.....	14
2.9.1 O Projeto Cyc.....	15
2.9.2 O Projeto "Wordtree".....	15

2.10 Ligações Causais.....	15
2.11 Ligações Semiótica.....	16
2.12 Redes Discretas.....	17
2.13 Relevância do Conhecimento.....	18
2.13.1 Acesso Limitado.....	18
2.14 Dicionários e Primitivas Semânticas.....	19
2.15 Inferência Semântica.....	20
2.16 Ramificações das Redes Semânticas.....	22
2.17 Dependência Conceitual.....	22
2.17.1 Os princípios da Dependência Conceitual.....	23
2.18 Grafos Conceituais.....	24
2.18.1 Abstração.....	25
2.18.2 Conceito de Grafo Conceitual.....	26
2.19 Conclusão.....	27
3 HISTÓRICO DE FERRAMENTAS WEB PARA A EDUCAÇÃO.....	29
3.1 Introdução.....	29
3.2 Conferência Simultânea.....	29
3.2.1 Ferramentas para Conferência Simultânea.....	30
3.2.2 Uso de Conferência Simultânea na Educação.....	34
3.2.3 Avaliação dos Aprendizes.....	37
3.2.4 Vantagens da Conferência Simultânea.....	37
3.2.5 Limitações da Conferência Simultânea.....	37
3.3 Conferência Não-Simultânea.....	38
3.3.1 Ferramentas para Conferência Não-Simultânea.....	38
3.3.2 Uso das Conferências Não-Simultâneas na Educação.....	40
3.3.3 Ferramentas das Conferências Não-Simultâneas na Educação....	41
3.3.4 Vantagens das Conferências Não Simultâneas.....	44
3.3.5 Limitações das Conferências Não Simultâneas.....	45
3.4 Hipermídia.....	46
3.4.1 Uso de Hipermídia na Educação.....	47
3.4.2 Ferramentas para Sistemas Hipermídia.....	48

3.4.3 Vantagens dos Sistemas Hiperímia.....	50
3.4.4 Limitações dos Sistemas Hiperímia.....	50
3.5 Redes Semânticas como Ferramentas da Mente.....	50
3.5.1 Guia de Estudo para Redes Semânticas.....	51
3.5.2 Reflexão do conhecimento.....	51
3.5.3 Instrução com Redes Semânticas.....	52
3.5.4 Avaliação para Redes Semânticas.....	52
3.5.5 Vantagens das Redes Semânticas como Ferramentas da Mente...	53
3.5.6 Limitações das Redes Semânticas como Ferramentas da Mente..	53
3.6 Conclusão.....	53
4. INDEXAÇÃO NO CONTEXTO DE GRAFOS CONCEITUAIS.....	55
4.1 Introdução.....	55
4.2 Definição de Conceito.....	56
4.3 Definição de Relação Conceitual.....	57
4.4 Definição de Contexto e Co-referência.....	58
4.5 Definição de Tipos de Conceitos.....	59
4.6 Definição de Relação de Conformidade.....	59
4.7 Definição de Grafos Canônicos.....	59
4.7.1 Definição: Regras de Formação Canônica.....	61
4.8 Definição de Cálculo de Predicado.....	66
4.9 Definição de Regras de Inferência.....	68
4.10 Categorias Ontológicas.....	69
4.11 Funções Temáticas.....	70
4.12 Localização das Funções Temáticas.....	74
4.13 Ponto de Vista.....	75
4.14 Tipos e Conjuntos.....	77
4.15 Conclusão.....	79
5. ARQUITETURA DO AMBIENTE - VEDA.....	80
5.1 Introdução.....	80
5.2 Arquitetura VEDA.....	81

## *Sumário*

---

5.3 Interfaces.....	83
5.4 Interfaces do Usuário Geral.....	83
5.4.1 Interface de Edição e Comunicação Simultânea.....	84
5.4.2 Interface de Busca.....	88
5.4.3 Interface de Navegação.....	89
5.5 Interfaces de Administração Conhecimento.....	91
5.5.1 Interface Modelos de Ontologia.....	91
5.5.2 Interface Grafos Canônicos.....	92
5.6 Pré-processamento.....	92
5.7 Mapeamento de GML para CGIF.....	93
5.8 Mapeamento de Grafos Conceituais (Forma Linear) para CGIF.....	93
5.9 Operações com Regras de Formação Canônica.....	93
5.10 Gerenciamento do Ambiente.....	93
5.10.1 Gerenciamento de Modelos de Ontologias.....	93
5.10.2 Máquina de Inferência.....	94
5.10.3 Gerenciamento do Conhecimento.....	94
5.10.4 Gerenciamento de Grafos Canônicos.....	94
5.11 Repositórios.....	94
5.11.1 Repositório Modelos de Ontologia.....	94
5.11.2 Repositório de Documentos.....	94
5.11.3 Repositório de Conversa Textual.....	95
5.11.4 Repositório de Grafos Canônicos.....	95
5.12 Conclusão.....	95
6 CONCLUSÃO.....	97
6.1 Considerações Finais.....	97
6.2 Sugestões de Trabalhos Futuros.....	98
ANEXO A – CONCEITOS.....	99
ANEXO B – RELAÇÕES CONCEITUAIS.....	107
ANEXO C – USO DE ARCO COM SETAS.....	112
ANEXO D – CATEGORIAS DE NÍVEL SUPERIOR.....	113
ANEXO E – TIPOS DE RELAÇÃO.....	119

*Sumário*

---

ANEXO F – ADJETIVOS MODIFICANDO SUBSTANTIVOS.....	124
ANEXO G – TESTE TEM.....	127
ANEXO H – ADVERTÊNCIAS PARA DEFINIÇÕES DE BASES DE CONHECIMENTO.....	128
GLOSSÁRIO.....	132
BIBLIOGRAFIA.....	134

---

## ABSTRACT

The work describes a shared environment architecture for collaborative education using conceptual graphs. The approach implies the implementation of software tools for editing, searching, navigation, and management. The system's environment provides the following general properties: visual environment of the knowledge, collaborative learning environment and Internet access; facilitating the conception of a virtual community where the collaborative learning on the Web is natural.

Today the Internet represents the biggest technological advancement since the appearance of print five hundred years ago [DRA.00]. In brief, a large part of the instruction disclosed to the society will be accessed via the Web in the near future. Despite the great appeal for the implementation of long distance education systems, only a few of them have escaped the classic behaviorist paradigm [SKI.58] where the learner is considered a mere receptor of the information and not a constructor of knowledge [JON.95]. To breach with the "page change" educational paradigm and the evaluation by multiple choice, the designer of educational environments must consider changing the educational paradigm from the conception of the learning environment.

---

## RESUMO

Esse trabalho descreve a arquitetura de um ambiente compartilhado para educação colaborativa usando grafos conceituais. A abordagem implica a implementação de ferramentas de software para edição, busca, navegação e gerenciamento na Web. O ambiente do sistema provê as seguintes propriedades gerais: ambiente visual do conhecimento engloba todas as áreas da educação, ambiente de aprendizagem colaborativa e acesso a Internet; facilitando a concepção de uma comunidade virtual, onde o aprendizado colaborativo na Web é natural.

Hoje a Internet representa o maior avanço tecnológico desde o aparecimento da imprensa a 500 anos atrás [DRA.00]. Em breve, uma grande parte da instrução disponibilizada à sociedade será acessada via Internet. Apesar do surgimento de vários sistemas para educação a distância, apenas alguns deles escapam do paradigma behaviorista clássico [SKI.58], onde o aprendiz é considerado um mero receptor da informação e não um construtor do conhecimento [JON.95]. Para romper com o paradigma educacional da “mudança de página” e a avaliação de múltipla escolha, o projetista dos ambientes educacionais devem considerar a mudança do paradigma educacional na concepção do ambiente de aprendizagem.

---

# 1 Introdução

## 1.1 Apresentação

A expansão da comunicação introduzida pelo grande avanço das áreas de telecomunicações e redes de computadores, tem viabilizado a criação de ambientes de aprendizagem colaborativa baseados em redes de computadores, possibilitando interações síncronas e assíncronas entre os participantes, independentemente de tempo e da localização dos mesmos. Em breve, uma grande parte da instrução disponibilizada para a sociedade será acessada via Web.

O crescimento das informações bem como sua rápida proliferação, juntamente com o constante avanço tecnológico e científico, tem resultado em atraso na assimilação do conhecimento e um aumento da complexidade das atividades a serem desenvolvidas. Temos, com isso, uma exigência cada vez maior de que os profissionais realizem uma constante reciclagem de conhecimentos e que sejam capazes de trabalhar em grupo, já que as tarefas passam a ser muito complexas para serem realizadas individualmente. É neste contexto que a aprendizagem colaborativa aparece como uma forma de preparar os alunos para esta realidade, permitindo que eles “aprendam a aprender”, através de seu engajamento em atividades colaborativas de ensino.

Este trabalho visa propor um ambiente colaborativo que represente visualmente a estrutura semântica de uma página Web, utilizando grafos conceituais. Grafos conceituais são equivalentes a lógica de primeira ordem, além de intuitivos e de grande poder de expressão.

## 1.2 Objetivos

Objetivo geral:

- Propor um ambiente colaborativo de representação visual do conhecimento baseado em grafos conceituais para a Web.

## 1.3 Contexto

[SOW.00] A representação do conhecimento desenvolveu-se como um ramo da inteligência artificial—a ciência de projetar sistemas computacionais para executar tarefas que normalmente requerem inteligência humana. Mas hoje, sistemas avançados em toda parte do mundo estão executando tarefas que requerem inteligência humana, como: recuperação da informação, projeto de circuitos, realidade virtual, reconhecimento da fala e máquinas tradutoras.

A representação do conhecimento é um assunto multidisciplinar que se refere a teorias e técnicas em três campos[SOW.00]:

1. A lógica provê a estrutura formal e regras de inferência.
2. A ontologia define os tipos de coisas que existe no domínio de uma aplicação.
3. A computação suporta aplicações que distinguem representação do conhecimento da filosofia pura.

Sem a lógica, a representação do conhecimento é vaga, sem critério para determinar se relatos são redundantes ou contraditórios. Sem a ontologia, os termos e símbolos são mal definidos e confusos. E sem os modelos computacionais, a lógica e a ontologia não poderiam implementar programas computacionais. Representação do conhecimento é a aplicação da lógica e da ontologia para as tarefas de construção de modelos computacionais para algum domínio.

Grafo conceitual é um sistema baseado em lógica dos grafos existenciais de Charles Sanders Peirce e de redes semânticas da inteligência artificial. Eles expressam o sentido de forma logicamente precisa, legível pelos humanos e computacionalmente tratável. Com sua representação gráfica, eles são legíveis, mas formalmente projetados e especificados em linguagem. Grafos conceituais têm sido implementados em uma variedade de projetos de recuperação de informação, banco de dados, sistemas especialistas, e processamento de linguagem natural.

Os grafos conceituais são formalmente definidos em uma sintaxe abstrata e independente de qualquer notação, mas o formalismo pode ser representado em várias notações diferentes. Devido a esta facilidade de representação, várias proposições de notações

diferentes e extensões foram apresentadas. Para evitar distorções do formalismo dos conceitos básicos sobre grafos conceituais, criou-se uma padronização projetada pela ANS (American National Standard).

## 1.4 Motivação

Vários estudos estão sendo direcionados a estruturação, indexação e representação do conhecimento. Chevallet e Chiaramella, Université de Grenoble, em seu artigo "Our experience in logical IR modeling" [CHE.99], relata sobre suas experiências desde 1978, na área de recuperação da informação, e a adoção de grafos conceituais como atual estudo para recuperação da informação.

## 1.5 Hipótese

Apesar do grande aparecimento de sistemas para educação à distância, apenas alguns deles escapam o paradigma behaviorista clássico [SKI.58], onde o aprendiz é considerado um mero receptor da informação e não um construtor do conhecimento [JON.96].

## 1.6 Metodologia

A pesquisa desenvolveu-se em etapas, cada uma das quais assumiu características particulares o que nos permite classificá-la como:

- Teórica ou bibliográfica nas etapas em que estivemos procurando consolidar os conhecimentos nas diversas áreas pertinentes ao tema.
- Exploratórias e descritivas nas fases em que estivemos estabelecendo e identificando os fatos relevantes para o problema, comprovando ou rejeitando as hipóteses e propondo e descrevendo uma arquitetura para o ambiente.

## 1.7 Organização do Texto

O capítulo 2 faz uma recapitulação geral sobre Redes Semânticas. Algumas formalizações criadas ao longo dos anos, desde sua utilização em computadores, são apresentadas. Desta forma, podemos observar as intenções originais de sua invenção e as potencialidades para representação do conhecimento. Esta abordagem ainda visa a seleção de uma forma de

---

indexação que seja conveniente com as expectativas de concepção de um ambiente colaborativo e visual do conhecimento para Web.

O capítulo 3 apresenta um histórico de ferramentas Web combinando recursos de colaboração e representação visual. O texto também oferece uma avaliação de vantagens e limitações das ferramentas e como elas podem ser usadas na educação. Paralelamente é abordada a evolução tecnológica dos softwares utilizados na Internet atingindo as novidades atuais.

O capítulo 4 apresenta a indexação utilizada para a criação do ambiente visual de representação do conhecimento baseada em grafos conceituais. São definições básicas que são abordadas mais profundamente nos anexos. Alguns tópicos oferecem precauções a serem tomadas na concepção e uso dos grafos conceituais.

O capítulo 5 apresenta a arquitetura do ambiente VEDA. A arquitetura integra os recursos de grafos conceituais visando um ambiente visual de representação do conhecimento para Web. Combinando grafos conceituais com hipertexto a arquitetura VEDA disponibiliza um ambiente visual e colaborativo para a educação.

---

## 2 Redes Semânticas

### 2.1 Introdução

Rede semântica é uma representação gráfica da estrutura do significado de sua representação, onde cada *nó* simboliza um conceito, e as ligações entre estes simbolizam relações [LEH.92]. Programas em Inteligência Artificial (IA) podem ser escritos, diretamente baseados nessas estruturas, e inferir conseqüências dos fatos afirmados. Essa teoria apresenta categorias de estruturas semânticas necessárias para o entendimento do mundo real, explorando e combinando abstrações matemáticas, lingüísticas, da ciência da computação e filosofia. No contexto de uma rede semântica, o conhecimento pode ser analisado e dividido em componentes significativos, permitindo a identificação de erros conceituais, alteração de hipóteses "*chaves*", e determinação de suas conseqüências.

Uma rede semântica representa o conhecimento como uma rede gráfica. Uma idéia, um evento, uma situação ou um objeto, tem uma estrutura de composição; assim, podemos representar uma estrutura correspondente de "*nós*" (desenhados por círculos ou caixas), simbolizando unidades conceituais, e ligações (desenhadas por setas entre os *nós*), simbolizando relações entre as unidades.

Uma rede abstrata pode ser diagramada, definida matematicamente, programada em computador, ou composta eletronicamente em equipamento. Um sistema de rede semântica inclui métodos para incorporar o conhecimento implícito. Por exemplo, a afirmação de que Maria lê (Figura 2.1), implica que ela é uma pessoa alfabetizada, e que é uma pessoa do sexo feminino. Quase todos os sistemas, possuem uma hierarquia estruturada de conceitos, ou taxinomia, usadas para este tipo de derivação.

### 2.2 Evolução Histórica do Conceito de Redes Semânticas

A essência da idéia de redes semânticas, é que as estruturas teórico-gráficas, das abstrações e relações, podem ser usadas para inferência e compreensão do sentido. A forma básica foi primeiramente desenvolvida no século passado, nas áreas da matemática avançada, para solucionar problemas de equações simultâneas [LEH.92].

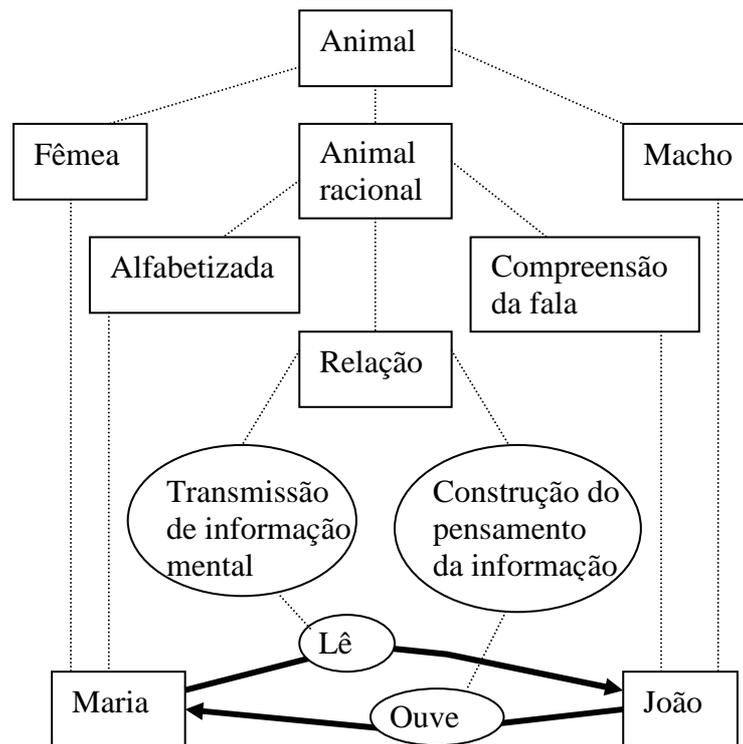


Figura 2.1. "Maria lê enquanto João ouve".

No campo da álgebra, matemáticos da Inglaterra como Cayley e Sylvester [SYL.09] perceberam que certos conjuntos de equações inter-relacionadas, poderiam ser mais bem solucionadas usando-se estruturas abstratas, que tinham características em forma de redes e de árvores.

Alfred Bray Kempe, engenheiro inglês, e Charles S. Peirce, filósofo americano, matemático, cartógrafo, lingüista e lógico, generalizaram o tratamento das estruturas relacionais matemáticas de Sylvester para todas as estruturas conceituais, usando diagramas de relacionamento.

Em "Memoir on the Theory of Mathematical Form" [KEM.86], Kempe descreveu seu sistema de diagrama parecido com os atuais; ele usava *nós* para unidades de conceitos e linhas para diferenciar pares de unidades. Ele escreveu:

*"Meu objetivo neste texto biográfico, é separar do pensamento matemático, a substância essencial do vestuário casual—geométrico, algébrico, lógico—em*

*que é em geral apresentado para consideração; e indicar onde se manifesta a variedade infinita dessa substância essencial".*

Assim, a primeira preocupação de Kemp foi dispensar os significados que tinham intenções particulares da estrutura pura da rede. Peirce [PEI.76] ficou muito entusiasmado com os diagramas de Kemp e trabalhou por décadas para desenvolver seu próprio diagrama de conceitos chamado "Grafos Existenciais" (Figura 2.2).

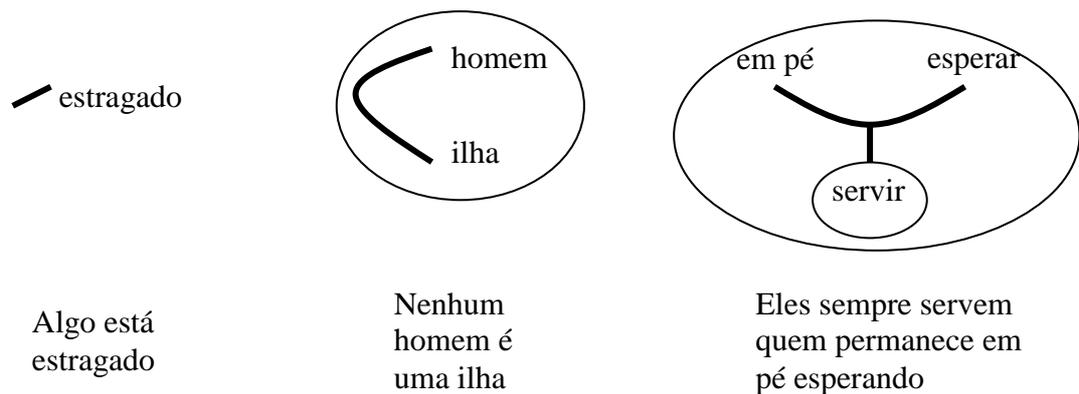


Figura 2.2. Grafos Existenciais de Peirce.

Esse sistema inclui uma versão gráfica bidimensional de lógica de predicados de primeira ordem com algumas extensões; e provê regras rigorosas para manipulação dos diagramas preservando a verdade e constituindo um sistema de prova. Indivíduos são representados por linhas e relações por *nós*, exatamente o oposto da prática moderna. Peirce foi o inventor do Cálculo de Predicados, da Álgebra das Relações e da Teoria do Reticulado. Durante o período dessas invenções, ele desenvolveu e usou os Grafos Existenciais que ele considerava como uma representação superior, quando se tratava de uma análise mais cuidadosa. Em uma topologia anterior, ele tentou derivar verdades metafísicas aplicando resultados topológicos a seus diagramas. Isso o levou a seu controverso Teorema da Redução, de que todos os conceitos podem ser definidos como mônadas (qualidades), díades (relações entre duas coisas) e alguns, necessariamente, tríades (relações entre três coisas), mas nenhuma relação de maior valência. Assim, ele desenvolveu as doutrinas filosóficas do Pragmatismo e a teoria da Semiótica.

## 2.3 Utilização de Redes Semânticas em Computadores

A primeira rede semântica para computadores foi construída no âmbito do projeto Nude, criada por R. H. Richens [RIC.56] da Unidade de Pesquisa Lingüística de Cambridge em 1956, como uma "interlíngua" para tradução de linguagens naturais. A idéia era ao invés do computador traduzir diretamente, por exemplo, do Russo para o Português, traduziria primeiramente para uma linguagem conceitual "neutra", ou interlíngua e depois dessa para a linguagem de destino.

No início, o sistema Nude envolveu a idéia de primitivas semânticas, isto é, um pequeno centro de conceitos básicos onde todos os outros conceitos poderiam ser construídos. Margaret Masterman, chefe da Unidade de Pesquisa Lingüística de Cambridge, criou um léxico para organizar os conceitos da linguagem em uma hierarquia acessível ao computador. Visto que, um ou mais conceitos poderiam ser inclusos abaixo de um ou mais conceitos de nível maior (mais abstrato), ela concluiu que os conceitos eram organizados em um reticulado matemático e não somente uma árvore.

## 2.4 Histórico da Formalização de Redes Semânticas

Redes semânticas foram usadas em IA por anos antes de serem cuidadosamente analisadas. Várias idéias foram implementadas em programas de computador utilizando métodos "ad hoc" e notações mal definidas. Isto foi um convite para uma formalização rigorosa em lógica, teoria dos conjuntos e modelos teóricos semânticos. Brachman [BRA.79] sugeriu cinco diferentes níveis de ligações que foram usados em redes semânticas, do nível mais baixo, localização de dados e ponteiros, ao nível mais alto, de palavras e descrições lingüísticas (Tabela 2.3).

Tabela 2.3. Cinco níveis de ligações baseados na análise de Brachman.

NÍVEL	COMPONENTES	ESTRUTURAS
Lingüístico	Conceitos arbitrários, palavras, expressões.	Sentenças e descrições.
Conceitual	Relações semânticas ou conceituais (casos), objetos primitivos e ações.	Dependências conceituais, casos em profundidade de

		redes semânticas.
'Epistemológico' (estrutural)	Tipos de conceitos, conceitos subdivididos, herança e estruturas relacionais.	Sistema de ligações associativo, relacional, INSTÂNCIA-DE e É-UM.
Lógico	Proposições, predicados e operadores lógicos.	Nós lógicos booleanos, 'partições' e negação de contextos.
Implementação	Átomos e ponteiros.	Estrutura de dados e frames.

## 2.5 Sistema de Frames

A representação padrão de redes semânticas em computadores convencionais usa frames como estrutura de dados [LEH.92]. No exemplo abaixo, observa-se o frame de um carro chamado Batcar (Figura 2.4).

(Batcar)	
Slot	Valores
INSTÂNCIA-DE:	Valor: (Carros)
Nome:	Valor: (Batcar)
Cor:	Valor: (Preto)
Modelo:	Valor: (Futurista)
Fábrica:	Valor: (Nasa Motors)
Preço:	Valor: R\$ 321,00
Freios:	Valor: WXZ, ABS
Pneus:	Valor: Chapa larga

Figura 2.4. Frame para um carro.

Um frame é um nome de objeto de dados com uma coleção flexível de nome "slot" (atributos ou campos) que pode ter valores [HAY.79]. Os valores, muitas vezes, são ponteiros para outros frames, permitindo uma rede de frames apontando um para o outro.

Um sistema de frame é como um grafo direcionado com vértices e arcos rotulados. Um frame é um *nó* ou um vértice; cada slot é um arco direcionado e rotulado, apontando para um vértice (frame), sendo que seu rótulo aparece no slot. Alguns slots podem ter valores

simples que não são outros frames. O slot do valor do carro tem "R\$ 321,00". O valor do slot pode ser um conjunto de valores, um exemplo é o slot "*Freios*" (Figura 2.4).

Minsky [MIN.81] criou frames como uma teoria psicológica da expectativa e do reconhecimento baseada em estereótipos e propôs frames para reconhecimento visual, lingüístico, extensões de narrativas ("*scripts*") e analogias.

Algumas linguagens foram fundamentadas na idéia de frames. Em FRL (Frame Representation Language) e KRL (Knowledge Representation Language), o valor de um slot pode ser calculado no momento da execução, evitando o armazenamento, utilizando um procedimento vinculado ao slot. O procedimento (programa), chamado "daemon", é simplesmente "anexado" ao slot, sendo executado pela requisição de um valor (SE-NECESSÁRIO) ou pela adição ao slot (SE-ADICIONADO). Se o triângulo tem 5 metros de hipotenusa, não é necessário guardar o "5" no slot hipotenusa; em lugar disso, um programa daemon, SE-NECESSÁRIO, no slot hipotenusa, utiliza os valores dos lados (catetos) do triângulo para calcular a hipotenusa e retorna o valor. Dessa forma se o lado do triângulo for alterado, não é necessário alterar o valor do slot hipotenusa (Figura 2.5).

(TRIÂNGULO RETÂNGULO)	
INSTÂNCIA-DE	(TRIÂNGULO)
Nome:	Valor: "triângulo retângulo"
Cateto A:	Valor: 4 m
Cateto B:	Valor: 3 m
Hipotenusa:	Valor: SE-NECESSÁRIO

Figura 2.5. Frame de um triângulo retângulo.

## 2.6 Relações de Casos Semânticos

A maioria das redes semânticas utilizam "ligações de casos" para descrições normais. A idéia de caso vem da gramática. "Caso nominativo" e "caso dativo" descrevem a relação sintática das sentenças entre o verbo e o nome.

O caso ATOR (Figura 2.6) liga a ação smash para algum agente causal animado. O OBJETO (também chamado TEMA ou PACIENTE) aponta para a coisa mais afetada por ele. O INSTRUMENTO aponta para algo que propositadamente foi afetado pelo ATOR trazendo efeitos futuros para o OBJETO. O LOCAL aponta para um objeto físico ou

evento físico. O BENEFICIADO aponta para uma pessoa ou causa servida. O TEMPO aponta para a hora em que houve a ocorrência.

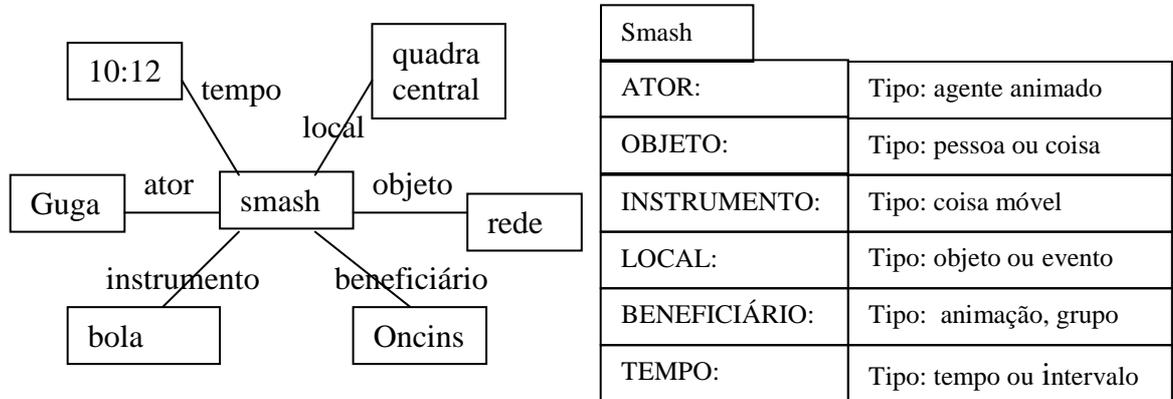


Figura 2.6. Uma rede de caso em profundidade.

## 2.7 Hierarquia de Abstrações

Um importante princípio de organização do pensamento é a abstração. Designando coisas particulares para categorias abstratas conseguimos dispensar detalhes irrelevantes e extrair, instantaneamente, conclusões sobre algo devido a sua participação em várias categorias. Redes semânticas especificam a estrutura de inter-relacionamento de categorias abstratas e usa esta estrutura para descrever conclusões. Temos "tipos de reticulado", "taxinomia", "léxico", "generalização de hierarquias" e "herança de hierarquias" como estruturas de inferência. Essas hierarquias podem ser usadas para uma série de propósitos, como: calcular a "distância semântica" entre dois conceitos, guiar e simplificar a prova de teoremas, classificar objetos descritos, reorganizar analogias válidas, achar fatos na base de dados e gerar programas procedurais. Existem três possíveis princípios de estrutura hierárquica:

1. O projetista decide quais os conceitos são subgrupos de outros e suplanta todas as ligações na mão. É o mais comum em IA e na história geral da taxinomia.
2. A estrutura hierárquica de conceitos é induzida (ou automaticamente gerada) por alguma outra estrutura formal.
3. A hierarquia de conceitos pode emergir diretamente das características estáticas de um conjunto de dados.

### 2.7.1 Ligações É-UM e INSTÂNCIA-DE

Muitas redes semânticas implementam hierarquia de abstrações usando ligações É-UM de sub-conceitos para super-conceitos. A ligação É-UM é diferente das ligações que definem relações particulares entre indivíduos do mundo descrito; ela define relações de abstrações entre dois conceitos. Seria a mesma distinção entre "ser" e "estar", assim como, É-UM e INSTÂNCIA-DE. É a diferença entre um indivíduo sendo o membro de uma classe e a classe sendo a subclasse de uma outra classe. Na interpretação estritamente lógica, INSTÂNCIA-DE significa pertence " $\in$ ", e É-UM significa contido " $\subset$ ". Diferente de É-UM, INSTÂNCIA-DE não é transitivo. Totó é uma INSTÂNCIA-DE Cão, e Cão é uma INSTÂNCIA-DE espécies, mas Totó não é uma INSTÂNCIA-DE espécies.

### 2.7.2 Herança Simples e Múltipla

Muitas estruturas de herança são árvores. Em uma árvore (geralmente desenhada expandida para baixo, com a raiz na cabeça) cada *nó* tem somente um *nó* imediatamente acima dele (seu "*nó* pai"), do qual ele pode herdar; isto é chamado de herança simples. Porém, na maioria das hierarquias É-UM, em IA, um *nó* pode possuir múltiplos *nós* pais e herdar qualidades de múltiplos caminhos. Isto é chamado de herança múltipla. Se o valor em um slot é herdado de múltiplos super-conceitos eles podem ser combinados pela união. No caso da informação herdada de dois super-conceitos serem conflitantes, e a informação conflitante for herdada "*estritamente*" (Item 2.7.3), então existe erro no conjunto de definições.

### 2.7.3 Herança Nula ou Estrita

Muitos sistemas de herança permitem a herança "não *estrita*", ou seja, herança "*nula*" ou, também, herança com exceções. Neste caso, alguma coisa pode impor (ou anular) a herança de um *nó* mais "alto". Exemplo clássico, "*pássaros voam*" e "*Pingo É-UM pássaro*". Você pode concluir que "*Pingo voa*". Suponhamos que "*Pingo É-UM pingüim*", "*pingüim É-UM pássaro*", e "*pingüins não voam*". Pássaros continuam voando, mas pingüins não voam. Então, Pingo voa ou não? Na herança nula isto não é um conflito proibido. Pingüim é uma exceção e a informação mais específica para pingüim impõe a informação de pássaro "*Pingo não voa*" (Figura 2.7).

### 2.7.4 Herança Relacional

Como os conceitos formam uma hierarquia abstrata, as relações (ligações) também possuem sua hierarquia de abstração que pode ser usada para inferência. "Lê" (Figura 2.1) considerada como uma relação, é uma sub-relação da relação "Transmissão de informação mental".

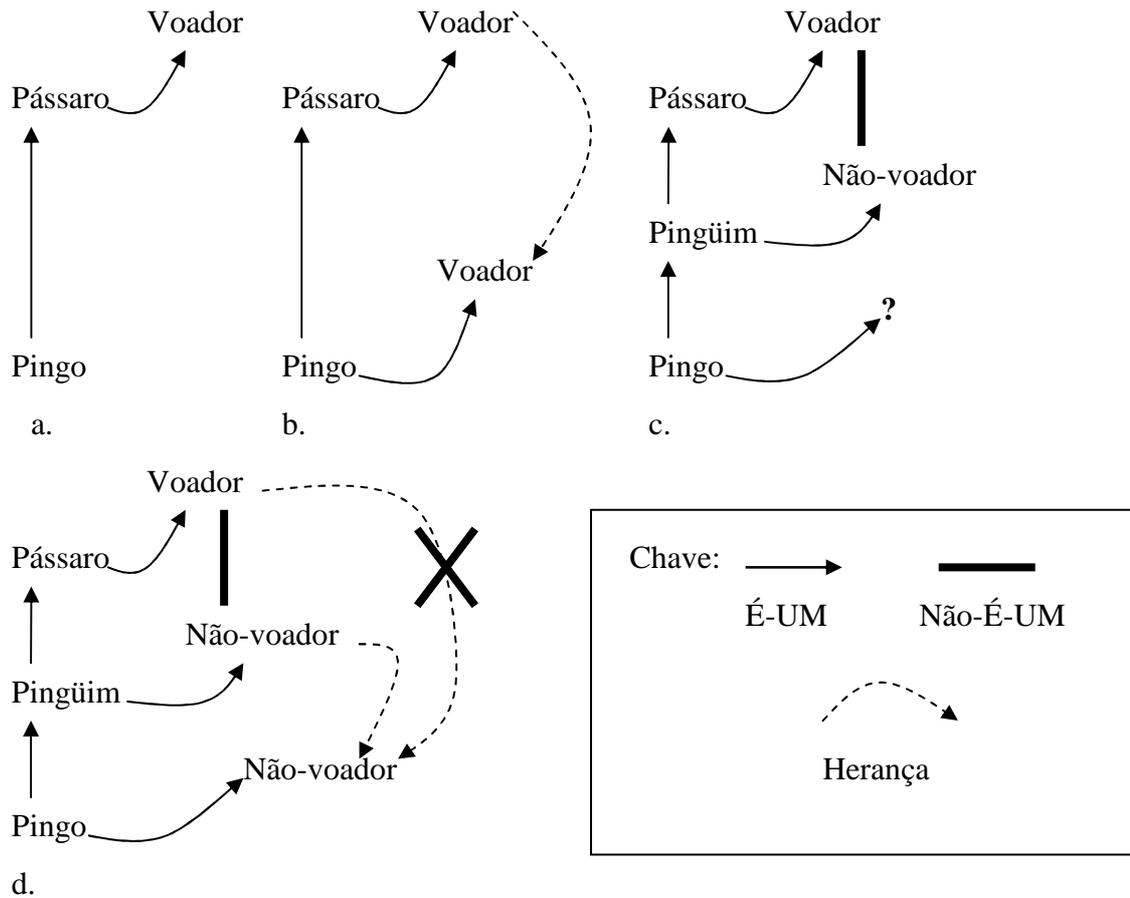


Figura 2.7. Herança nula.

### 2.8 Extensões Lógicas

Redes semânticas não possuem o mesmo poder de expressão da lógica de predicados [SOW.84]. Não há uma maneira simples de expressar a negação ("Jogador *NÃO* titular"), disjunção ("Vocês podem jogar tênis *OU* futebol"), ou quantificador universal ("TODOS juntos venceremos"). Grafos com afirmações quantificam apenas conjunções de relações. Redes semânticas com esta deficiência são expressivamente inferiores, não somente para a lógica, mas para todas as linguagens naturais. Existem várias extensões de redes semânticas para corrigir isto.

O sistema hiper-grafo "*HiGraph*" de Harel [HAR.88] encapsula uma parte de um hiper-grafo semântico e o trata como uma unidade. Ele também usa um conjunto de cápsulas (bolhas), para indicar a inclusão de um *nó* em múltiplos conjuntos. A finalidade é combinar a integração de classes com n-ária relações sem sobrecarregar o diagrama (Figura 2.8).

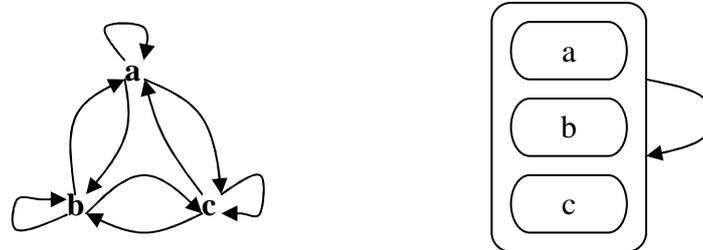


Figura 2.8. Hiper-grafo "*HiGraph*" de Harel

Grafos conceituais usam um sistema diferente de aninhamento de contextos, que foi derivado originalmente do aninhamento de negações dos Grafos Existenciais de Peirce [SOW.00]. Todas as conexões e quantificadores da lógica podem ser negociados, adicionando-se aninhamento de negação para as conjunções implícitas e para o quantificador existencial. Assim, não há necessidade de *nós* especiais para disjunção, implicação, equivalência ou qualquer outra conexão, nem tampouco para o quantificador universal. Todos os conectivos lógicos são definidos com NÃO e E, assim,  $\forall xP(x)$  é definido como  $\neg(\exists x\neg P(x))$ .

## 2.9 Ontologia

Para solucionar um problema no nível conceitual, você terá que escolher uma ontologia para sua área de aplicação [SOW.00]. Numa aplicação particular de IA nem sempre são óbvias as categorias, objetos, atributos, entidades e estruturas conceituais usadas. Muitas aplicações requerem uma ontologia especializada e uma ontologia geral do mundo. A profundidade semântica do modelo de domínio depende da riqueza da ontologia [LEH.92].

[Sowa, 94] Ontologia é o estudo das categorias de coisas que existe ou pode existir em algum domínio. O produto de um estudo assim, chamado uma ontologia, é um catálogo do tipos de coisas que são assumidas para existir em um domínio de interesse D, de uma perspectiva de uma pessoa que usa a linguagem L, com o propósito de falar sobre D. Os tipos na ontologia representam os conceitos (predicados) e tipos de relações de uma linguagem L quando usada para discutir tópicos no domínio D. Por ela mesma, lógica não

diz nada sobre nada, mas a combinação de lógica e ontologia provê uma linguagem que pode ser expressa em relações sobre as entidades no domínio de interesse. Uma ontologia formal é especificada por uma coleção de nomes para conceitos e tipos de relações organizada numa ordem parcial de relação por tipo-subtipo.

A maior parte das ontologias de IA tratam situações ou estados do mundo como objetos que podem participar em relações. Uma situação é alterada por um evento e gera uma nova situação. No mundo virtual toda ontologia tem alguma noção de tempo e espaço.

### 2.9.1 O Projeto Cyc

O mais ambicioso projeto de rede semântica é o projeto Cyc, sob a direção de Douglas Lenat e R. V. Guha [GUH.90]. Seu principal objetivo é determinar uma ontologia para o mundo real. O projeto prevê uma base de conhecimento com quase todo o conhecimento que uma pessoa pode possuir antes de ler uma enciclopédia. Isso inclui o conhecimento de "senso comum", por exemplo, que a pessoa tem uma existência física, família, duração de vida e desejos; e que dois objetos sólidos não podem ocupar o mesmo espaço ao mesmo tempo. Cyc usa uma rede semântica baseada em frame junto com uma linguagem lógica modificável que é capaz de limitar os valores dos slots (como exemplo, a idade de sua filha deve ser menor do que a sua).

### 2.9.2 O Projeto "Wordtree"

O projeto "Wordtree", de Henry G. Burger [BUR.84], define significados de palavras e conceitos combinando pares de conceitos mais primitivos. Isto inclui um imenso vocabulário que ele chama de transitivos (como verbos transitivos), sendo cada qual uma combinação de outros dois. Existe um nível base de 44 primitivas de verbos indefinidos com seus opostos, e.g. CONVEXO/CONCAVO, ESPACIAL/VAZIO; a partir do qual os outros são construídos.

## 2.10 Ligações Causais

Muitas ontologias usam ligações para representar um modelo causal profundo de sustentação do comportamento. Ligações causais são ligações relacionais especializadas que indicam a propagação à mudança. Causalidade envolve dependência e tempo. Em uma

fórmula matemática como  $P = VA$ , cada variável depende das outras duas, sendo que uma mudança em qualquer uma delas afetará as outras, não existindo uma ordem causal. Em contraste, jogando uma bola na janela, causa a quebra do vidro—quebrar o vidro não causa jogar a bola. Modelos causais de sistemas dinâmicos usam diagramas de influência para mostrar quais componentes influenciam os outros. Isto também pode mostrar quais partes do sistema são irrelevantes.

Toda ligação causal entre dois estados envolve duas ligações evidentes (em oposição contrária) entre eles. Apesar do fogo causar fumaça e não vice-versa, fogo é uma evidência de fumaça, e fumaça é uma evidência de fogo. Pearl [PEA.88] usa uma rede baseada em probabilidade de proposições de *nós* e ligações causais (Figura 2.9) com pesos numéricos para indicar o quanto um *nó* de proposição apoia (é evidência de crença) o outro.

A probabilidade de qualquer *nó* ser verdadeiro depende de todos os outros *nós* da rede, exceto alguns *nós* que se tornam irrelevantes se bloqueados por certos conjuntos de *nós* apoio. Por exemplo (Figura 2.9), "o regador está ligado" é uma boa razão para acreditar

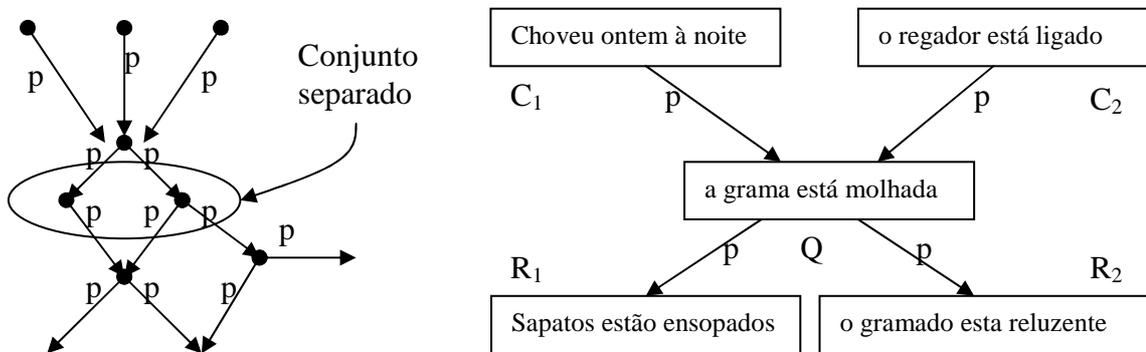


Figura 2.9. Rede causal de Pearl.

que "a grama está molhada", e separadamente, "a grama está molhada" é uma boa razão para acreditar que "choveu ontem à noite", mas "o regador está ligado" certamente é uma boa razão para não acreditar que "choveu ontem à noite". O caminho de crença de  $C_1$  para  $C_2$  é dito "bloqueado" por  $Q$ . Porém, devido à convergência das direções causais das ligações,  $C_1$  e  $C_2$  são conhecidas somente como causas possíveis para a grama molhada, elas ficaram mutuamente dependentes—sabendo-se uma delas elimina o apoio à crença da outra. Não existe este tipo de dependência entre  $R_1$  e  $R_2$ .

## 2.11 Ligações Semiótica

Outro tipo especializado de ligação é a ligação semiótica entre uma representação e o que é representado. Existe uma ligação entre o padrão físico atual das rodovias no Brasil e o mapa esquemático rodoviário. Existe uma outra ligação entre o mapa e a minha recordação mental do mapa, que é o modelo entre a ligação semiótica da coisa representada para minha mente. A maior parte da informação do modelo está perdida no processo, mas alguma informação é preservada. Isto também é verdade nos padrões mais abstratos como redes semânticas. Uma rede semântica simbolizando entidades que podem ser representadas necessita de ligação semiótica.

## 2.12 Redes Discretas

Uma rede semântica é uma estrutura discreta como qualquer linguagem de descrição. A representação do contínuo com esta estrutura é necessariamente incompleta e requer decisões de quais informações são aproveitadas e quais são perdidas.

Em operações de tempo real, um controlador fornece sinais de direção baseado nos valores do sistema de sensores. As leis de controle são fixadas por antecipação. Em ambientes complexos com eventos inesperados e mudança de prioridades (como combate aéreo), a teoria de controle clássica é inadequada porque novos modelos e novas leis de controle devem ser selecionadas de tempos em tempos baseados nas descrições semânticas dos modelos de estados. Isto é chamado de "*controle semântico*".

A teoria de *controle semântico* adiciona regras de mudança para os conceitos. Estas regras formam redes de transições relacionadas ao tempo entre diferentes redes semânticas. As regras iniciam o processo e limitam os estados de transições. Uma das redes semânticas representa o "*estado semântico*" corrente e as outras representam os estados de destino. O sistema precisa explorar o espaço de mudança de estados de uma rede à outra. Isto ocorre obedecendo às regras de mudança e as últimas informações. Quando um caminho para um estado objetivo é encontrado, uma série de "*geradores de programas*" geram as funções executáveis necessárias para o controlador. As regras de mudança podem ditar alterações discretas na rede, como: adição ou eliminação de objetos ou relações—tornando o sistema "*auto modificável*".

Controles semânticos envolvem o uso de técnicas de IA para entendimento de níveis mais altos de abstração além do nível básico de domínio dos sistemas de sensores. As hierarquias das redes semânticas permitem isso. Um sistema de aeronave pode negociar com estratégias de alto nível, como: "*sacrifique armas o suficiente para estender o alcance do vôo, evite radar terrestre, distraia o inimigo MiG por dez minutos e dirija-se para o ponto de pouso mais próximo*". Hierarquia de vários níveis são usadas em controle semântico para ascender ao mais alto nível de aplicação de abstração e executar todas as possíveis considerações desse nível. Usando o resultado, o sistema seleciona ou talvez, automaticamente projeta, o modelo de controle, e finalmente cria o domínio de entrada em concordância com a teoria de controle clássica.

## 2.13 Relevância do Conhecimento

É importante considerar no raciocínio os objetos relevantes, as características e relações em uma situação. O óbvio e o irrelevante devem ser descartados. Um problema sério das bases de conhecimento que utilizam a lógica padrão é que todo o fato representado do mundo tem uma chance de interferir em qualquer questão. Para aceitar uma inferência padrão, um sistema deve inspecionar a base de conhecimento inteira, certificando-se que nenhum outro fato implicará no cancelamento da inferência—uma perda de tempo gigantesca (dependendo da base de conhecimento).

### 2.13.1 Acesso Limitado

Redes semânticas oferecem uma disposição do conhecimento devido a sua implementação hierarquizada, de tal forma que algum conhecimento é mais acessível do que o resto. Uma presunção comum é de que fatos relevantes a um conceito serão alcançáveis seguindo suas ligações, e essa relevância vai sendo decrescida de acordo com a distância na rede. Alguns sistemas, como KRL e componentes especializados do Cyc, simplesmente cortam o acesso durante o processamento de busca, declarando "*até 25 ligações*" do conceito em questão.

Existem outras formas de limitar o acesso em redes semânticas. *Redes associativas* usam pesos numéricos para ligações similares. *Limite lógico de acesso* é uma formalização baseada em regras para frame, onde qualquer informação não acessível via encadeamento das relações, não pode ser usada em uma regra. Além disso, todas as regras estão em uma ou mais "*partições*". Em uma busca, a seqüência de regras não pode atravessar os limites

de uma partição. *Hierarquia de tópicos* pode ser usada para limitar buscas a áreas relevantes. Os assuntos são divididos em áreas taxinomia e sua relevância depende dos tópicos.

## 2.14 Dicionários e Primitivas Semânticas

Para que *nós* e ligações possam ser mais do que símbolos insignificantes, precisamos de alguma forma provar os seus sentidos. Uma abordagem é integrar uma forma léxica, ou dicionário, próximo à base de conhecimento. Cada definição no léxico é ele próprio uma rede semântica que é acessível pelo computador.

Quillian [QUI.85] considerou o "*sentido total*" de uma palavra como sendo a soma total de todas as estruturas construídas recursivamente, substituindo cada palavra por sua definição. Numa rede semântica, substituir a rede de definição de um conceito pelo *nó* representante do próprio conceito, é de fato, uma substituição gráfica de um grafo complexo por apenas um (Figura 2.10). Uma substituição dessas é um incremento granular de detalhes, ou seja, um "zoom" semântico.

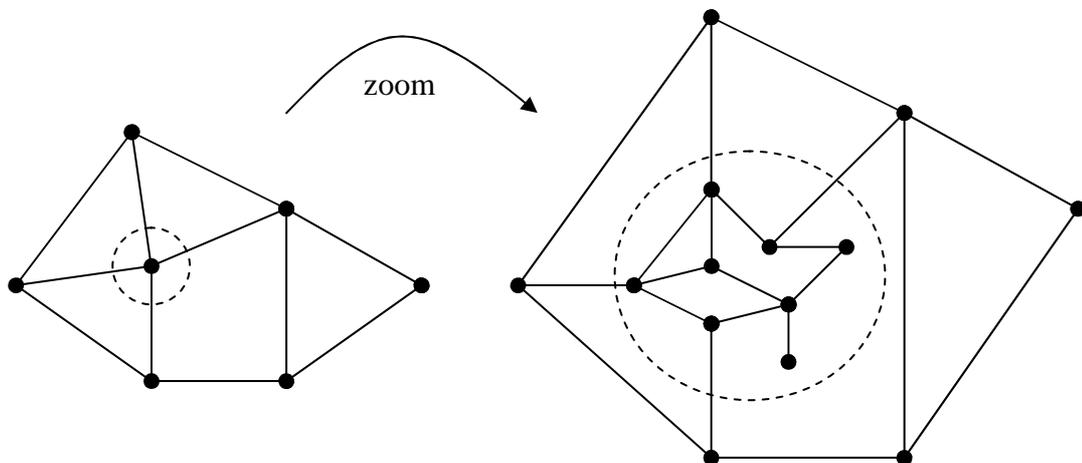


Figura 2.10. Expansão de um *nó* em sua rede de definição.

Essa substituição possibilita um retrocesso infinito (loop), pois cada termo definido pode encontrar ele mesmo. Para evitar isto, existem alguns níveis elementares de *primitivas* conceituais. A existência de primitivas semânticas é controversa. Schank em sua teoria original de "*Dependência Conceitual*" propôs primitivas para IA. A noção de primitivas semânticas não é nova, Leibniz [LEI.89] baseou-se nelas para "*Combinatória*" e Bishop John Wilkins [WIL.68] para "*Essay towards a Real Character and a Philosophical*

Language". Sistemas axiomáticos como Whitehead & Russel [WHI.27] "*Principia Mathematica*" ou Hayes "*Naïve Physics*" [HAY.79] são tentativas continuadas e são, na essência, ontologias de rede semântica como Cyc.

## 2.15 Inferência Semântica

A hierarquia de conceitos do mundo não é arbitrária. Não precisamos de um conceito mental para todas as possíveis combinações de qualidades, relações ou objetos. Temos uma, relativamente pequena e eficiente, codificação da estrutura do mundo, parcialmente adquirida no nascimento ou aprendida, baseada na economia do esforço da mente. O sistema do mundo tem uma estrutura dividida. Existem componentes maiores que são mutuamente independentes. A estrutura conceitual da lei da verdade, por exemplo, tem poucas conexões. Varias descrições de formas podem ser inter-relacionadas, mas todas são independentes de cor. Desta forma, podemos fatorar a hierarquia conceitual em componentes sub-hierárquicos, facilitando a computação.

Consideremos um reticulado de casos. Seus casos não são apenas símbolos, mas são relacionados em três sub-reticulados ontológicos diferentes, chamadas: Fase, Modo, e Grau (Figura 2.11).

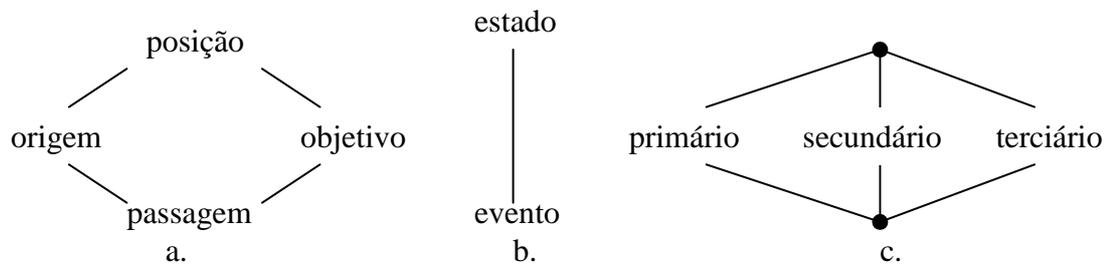


Figura 2.11. Três sub-categorias de um reticulado de casos.

Fase refere-se a orientação direcional: "*origem*", "*objetivo*", "*posição*", e "*passagem*". Modo define se a relação é passiva ou ativa. Grau envolve status causal primário (causa independente ou espontânea), secundário (resultado de causa), e terciário (fundo estacionário envolvido).

Os casos profundos aparecem no "*produto*" desses três reticulados, no reticulado maior (Figura 2.12). *Doador* difere de *Receptor* no senso óbvio da direção oposta (diferença de Fase), e difere de *Patrono* que é um conceito estático (diferença de Modo), e difere de

Origem porque o Doador incita uma transferência, ao contrário de meramente Origem (diferença de Grau).

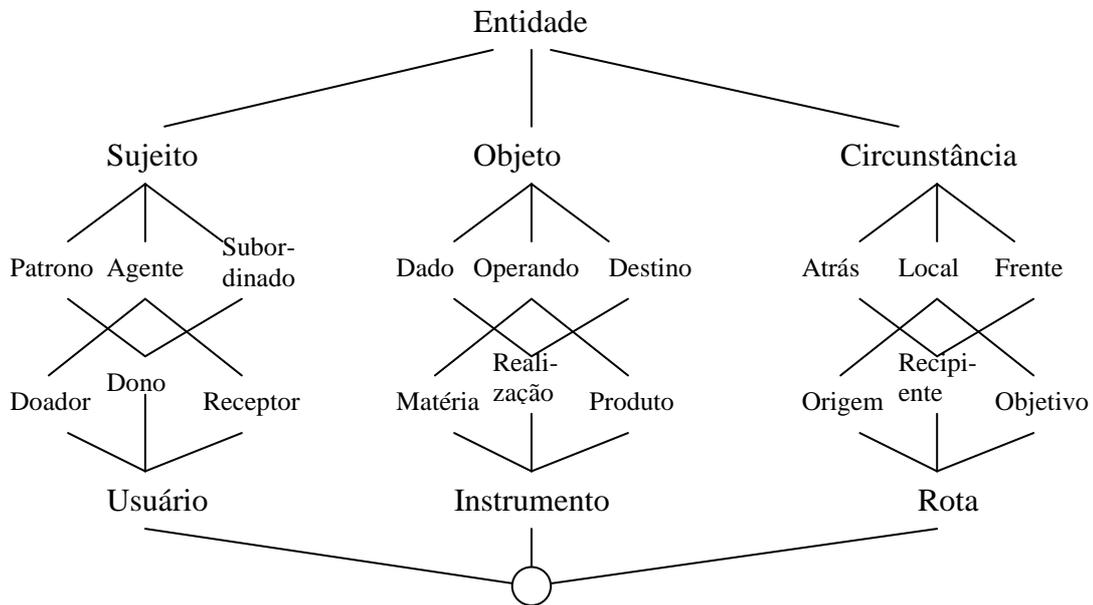


Figura 2.12. O produto de um reticulado de casos.

Em sua tese, "A Lattice Theoretic Approach to Computation Based on a Calculus of Partially Ordered Type Structures", Ait-Kaci [AIT.89] apresentou um sumário da "Teoria do Reticulado" utilizando uma abordagem para classificação de *tipos de dados abstratos*. Tipos de dados, em ciência da computação, são esquemas que especificam a disposição e o conteúdo das estruturas de dados. Exemplo, o tipo do número complexo é especificado em duas partes, uma real para a parte de número real, e outra real para a parte imaginária. Tipos de dados complexos são definidos usando estruturas de aninhamento (Figura 2.13).

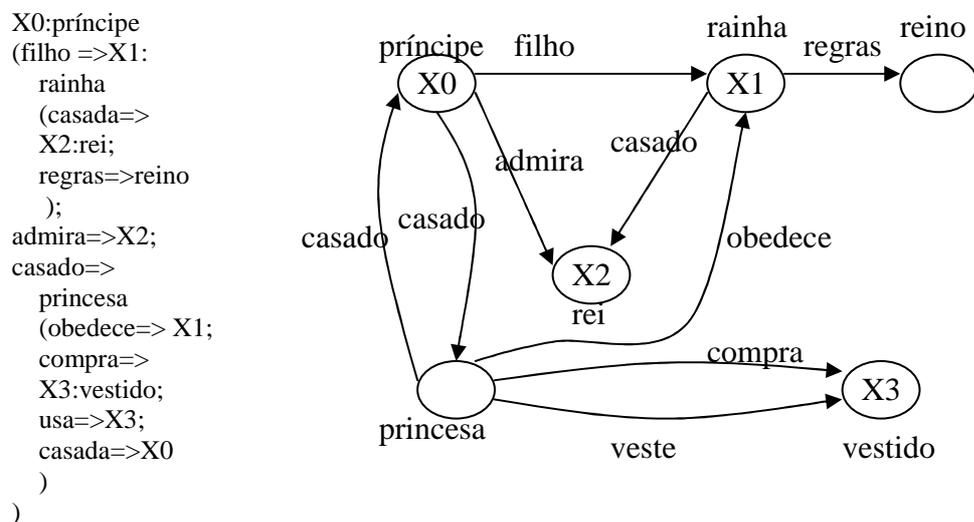


Figura 2.13. Um exemplo de tipo de estrutura da notação de Aik-Kaci.

## 2.16 Ramificações das Redes Semânticas

Duas das maiores famílias de pesquisa de sistemas de rede semântica que existem, independentemente em projetos ao redor do mundo, são apresentadas para fins de estudo, com intuito de desenvolvimento dos trabalhos em uma das categorias:

- Dependência Conceitual;
- Grafos Conceituais.

## 2.17 Dependência Conceitual

O objetivo original do trabalho de Dependência Conceitual (DC), foi desenvolver uma "*representação da base conceitual que fundamenta todas as linguagens naturais*" [SCH.75]. Esse objetivo resultou na proposta de um conjunto pequeno de ações primitivas e um conjunto de dependências que conectam as ações primitivas nelas mesmas e aos seus atores, objetos e instrumentos. As teorias de representação que descenderam de DC, como scripts, tem um número ilimitado de itens de vocabulário. A noção de "*primitiva*" parece ter desaparecido [LYT.92].

### 3.17.1 Os princípios da Dependência Conceitual

A teoria de dependência conceitual foi baseada em duas suposições [SCH.75]:

1. Se duas sentenças possuem o mesmo sentido, elas deveriam ser representadas como a mesma, sem considerar as palavras particulares usadas.
2. A informação implicitamente declarada deverá ser representada explicitamente. Isto é, qualquer informação que pode ser inferida a partir dela, deverá ser incluída na representação.

Essas suposições têm muitas implicações com a aparência que deve ter uma linguagem de representação. A primeira suposição implica que as representações devem ser gerais; isto é, elas devem capturar as similaridades do significado dos sinônimos. Esta suposição tem como hipótese que inferências sejam feitas para produzir representações completas. Isto quer dizer que representações devem suportar inferência. Para que isso seja feito eficientemente, elas devem ser canônicas. Sempre que uma inferência particular é feita, é desejável que a mesma regra de inferência possa sempre ser usada para este fim. Sem uma

representação canônica, várias regras poderiam ser solicitadas, cada uma para uma possibilidade diferente de representação.

O vocabulário de dependência conceitual consiste do seguinte:

- Um conjunto de primitivas, usadas para representar ações no mundo.
- Um conjunto de estados, usados para representar precondições e resultados de ações.
- Um conjunto de dependências, ou possíveis relacionamentos conceituais que podem existir entre primitivas, estados, e objetos envolvidos.

Por exemplo, (Tabela 2.14), juntando os conjuntos abordados acima, podemos formar um *grafo de dependência conceitual*.

Tabela 2.14. As primitivas da dependência conceitual [LYT.92].

PTRANS:	A transferência de local de um objeto.
ATRANS:	A transferência de posse, ou controle de um objeto.
MTRANS:	A transferência mental de informação entre agentes.
MBUILD:	A construção de um pensamento ou de nova informação do agente.
ATTEND:	A ação de focar a tenção de um órgão sensor para um objeto.
GRASP:	A captura de um objeto pelo ator, de forma que possa ser manipulado.
PROPEL:	A aplicação de uma força física em um objeto.
MOVE:	O movimento de uma parte do corpo do agente pelo agente.
INGEST:	A absorção de um objeto (comida, ar, água, etc.) por um animal.
EXPEL:	A expulsão de um objeto por um animal.
SPEAK:	A ação de produzir som, incluindo sons não comunicativos.

Cada primitiva tem um conjunto de slots associado a ela, oriundo do conjunto de dependências conceituais. Associado a cada slot existe restrições de quais objetos podem aparecer naquele slot. Por exemplo, os slots do PTRANS são os seguintes:

Tabela 2.15. Slots de PTRANS [LYT.92].

ATOR:	Um HUMANO (ou objeto animado), que inicia o PTRANS.
OBJETO:	Um OBJETO FÍSICO, que é movido.
DE:	Um LOCAL, onde o PTRANS inicia.
PARA:	Um LOCAL, onde o PTRANS termina.

Para explicitar a informação implícita, regras de inferência são escritas baseadas nas primitivas. Por exemplo, da primitiva PTRANS, pode-se inferir que o OBJETO que foi movido estava inicialmente no local DE, e depois do PTRANS foi para o local PARA.

Com intuito de executar inferências sob certa perspectiva (ponto de vista), estruturas de conhecimento chamadas *scripts* foram criadas. Scripts representam seqüências estereotipadas de eventos, como por exemplo, a ida a um restaurante. Usando script, teoricamente, o computador tem acesso rápido aos fatos que sempre acontecem numa seqüência de eventos estereotipada, sem necessitar fazer outras inferências que em sua grande maioria são irrelevantes.

Um script consiste de um conjunto de *funções* e *cenas*, cada qual descrevendo um evento típico. Por exemplo, em um script \$Restaurante, algumas funções são o cliente, o garçom, o restaurante e a comida. As cenas incluem ENTRAR, PEDIR, COMER, PAGAR e PARTIR.

### 3.2. Grafos Conceituais

[SOW.84] Grafos conceituais é um sistema baseado na lógica dos grafos existenciais de Charles Sanders Peirce. Eles expressam o sentido de forma logicamente precisa, legível pelos humanos e computacionalmente tratável. Com um mapeamento direto para linguagem, grafos conceituais servem como uma linguagem intermediária para traduzir formalismo orientado a computação, de e para linguagens naturais. Com sua representação gráfica eles são legíveis, mas formalmente projetados e especificados em linguagem. Grafos conceituais foram implementados em uma variedade de projetos de recuperação de informação, projeto de banco de dados, sistemas especialistas, e processamento de linguagem natural.

Grafos conceituais são formalmente definidos em uma sintaxe abstrata e independente de qualquer notação, mas o formalismo pode ser representado em várias notações diferentes. Devido a esta facilidade de representação, várias proposições de notações diferentes e extensões foram apresentadas. Para evitar distorções do formalismo dos conceitos básicos sobre grafos conceituais, criou-se uma padronização desenhada pela ANS (American

National Standard). Abaixo (Figura 2.16), temos um grafo conceitual que representa o conteúdo de uma sentença em Português. “O gato está sobre o tapete”.



Figura 2.16: Grafo conceitual para: "O gato está sobre o tapete".

Em um grafo conceitual, as caixas são chamadas conceitos e os círculos são chamados relações conceituais. Essa forma de grafo, que é chamado de *forma de exibição (display)*, é altamente legível para humanos e mapeada diretamente para a forma abstrata de grafo conceitual. Existe também uma segunda forma chamada *forma linear*, que é exatamente equivalente à notação gráfica no poder de expressão:

$$[\text{Gato}] \leftarrow (\text{Sobre}) \leftarrow [\text{Tapete}].$$

Desta forma, os *nós* conceitos são representados por um colchete ao invés de caixas, e a relação conceitual representada por parênteses ao invés de círculos. Ambas as notações, gráfica e linear, são designadas para serem usadas na comunicação entre computadores e humanos. Para comunicações entre computadores, outra forma chamada de *Conceptual Graph Interchange Format (CGIF)*, tem uma sintaxe simples e um conjunto de caracteres mais restritos:

$$[\text{Gato: *x}] [\text{Tapete: *y}] (\text{Sobre ?x ?y}).$$

CGIF pretende intensificar a velocidade de transferência entre os sistemas de computador que usam grafos conceituais. Para comunicação de sistemas baseado em outras linguagens e notações, CGIF pode ser mapeada para KIF (*Knowledge Interchange Format*):

$$(\text{exists } ((?x \text{ gato}) (?y \text{ tapete})) (\text{Sobre ?x ?y})).$$

### 2.18.1 Abstração

Informalmente, um grafo conceitual é uma estrutura de conceitos e relações conceituais onde todo arco conecta alguma relação conceitual  $r$  para algum conceito  $c$ . Formalmente, a estrutura dos grafos é definida num estilo abstrato que mostra os detalhes em grande precisão; mas sem compromisso com qualquer detalhe da notação ou implementação.

### 2.18.2 Conceito de Grafo Conceitual

Um grafo conceitual  $g$  é um grafo bipartido, que tem dois tipos de *nós* chamados conceitos e relações conceituais.

- Cada arco  $a$  de  $g$ , deve ligar uma relação conceitual  $r$  em  $g$ , para um conceito  $c$  em  $g$ . O arco  $a$  é dito pertencer à relação  $r$ . (O arco  $a$  está ligado ao conceito  $c$ , mas não pertence a  $c$ ).
- O grafo conceitual  $g$  pode ter conceitos que não são ligados a nenhuma relação conceitual; mas todo arco que pertence a  $a$ , de qualquer relação conceitual em  $g$ , deve ser ligado a exatamente "um" conceito em  $g$ .
- Três tipos de grafos conceituais têm nomes distintos:
  1. O *branco* é um grafo conceitual vazio sem nenhum conceito, relação conceitual ou arco.
  2. Um *singular* é um grafo conceitual que consiste em um simples conceito; mas nenhuma relação conceitual ou arco.
  3. Uma *estrela* é um grafo conceitual que consiste de uma simples relação conceitual  $r$ , e um ou mais conceitos que são ligados ao(s) arco(s) de  $r$ .

Para ilustrar essa definição, considere o seguinte grafo conceitual que representa a sentença: "*João vai para Natal*":

$$[\text{Pessoa: João}] \rightarrow (\text{Agente}) \rightarrow [\text{Ir}] \leftarrow (\text{Dest}) \leftarrow [\text{Cidade: Natal}].$$

Esse grafo contém três conceitos: [Pessoa:João], [Ir], e [Cidade:Natal]. Contém também duas relações conceituais: (Agente) relaciona [Ir] com o agente João, e (Dest) relaciona [Ir] com o destino Natal. O termo bipartido significa que todo arco de um grafo conceitual conecta um conceito e uma relação conceitual. Não existe grafos que conectam conceitos a conceitos, ou relações a relações. Dois dos quatro arcos no grafo pertencem a (Agente), e os outros dois pertencem a (Dest).

Um grafo conceitual  $g$  com  $n$  relações conceituais, pode ser construído de  $n$  grafos *estrela*, um para cada relação conceitual em  $g$ . Como o grafo conceitual exemplificado acima, tem duas relações conceituais, ele pode ser construído dos dois grafos estrelas seguintes:

$$[\text{Pessoa: João}] \rightarrow (\text{Agente}) \rightarrow [\text{Ir}] \quad [\text{Ir}] \leftarrow (\text{Dest}) \leftarrow [\text{Cidade:Natal}].$$

Se deixado assim, esses dois grafos *estrela* constituem um grafo conceitual desconectado. Para formar um grafo conceitual conectado, eles podem ser juntados cobrindo os dois conceitos idênticos do tipo [Ir] para formar o grafo conectado acima.

## 2.19 Conclusão

As estruturas de representação que descenderam de dependência conceitual são muitas e variadas. Por causa disso, parece que elas não compartilham muito em comum. Porém, a motivação de todas elas parece ser a mesma "*facilitar inferências*". Como pessoas, podemos fazer inferências em todos os níveis de generalidade, a linha de atuação dessas estruturas reflete isso [LYT.92].

Como podemos ver, grafos conceituais foram derivados de uma síntese da lógica e da lingüística. A lógica é baseada nos grafos de Peirce, unificação e  $\lambda$  cálculo. A lingüística é baseada na dependência de grafos e gramática de casos. A primeira versão de grafos conceituais foi desenvolvida em 1968 como um "paper" do curso de Marvin Minsky [MIN.75] em IA (MIT). Com  $\lambda$  cálculo, grafos conceituais permitiram uma forma de reorganização do conhecimento, definindo conceitos e relações de uma base de conhecimento em termos das primitivas de outra.

A proposta da teoria de grafos conceituais não é excluir ou alterar as outras teorias de redes semânticas, mas sim, acomodá-las e mostrar como elas se inter-relacionam [SOW.84].

Grafos conceituais podem gerar grafos em vários níveis de abstração. Quando os detalhes são irrelevantes, eles podem ser escondidos pela contração  $\lambda$ . Quando detalhes são necessários, eles podem ser recuperados pela expansão  $\lambda$ .

Frames podem ser diretamente mapeado em um grafo conceitual. Toda a complexidade, incluindo estruturas de aninhamento, pode ser recuperada quando as definições são expandidas. Apesar de Frames serem convenientes em muitas finalidades, eles não podem expressar todas as possíveis combinações de negação, disjunção, e quantificadores universais [SOW.84]. Grafos conceituais, porém, podem representar totalmente a lógica de primeira ordem [SOW.00].

---

## 3 Histórico de Ferramentas Web para a Educação

*"Quem mais aprende com os materiais de instrução são os próprios projetistas, e não o aprendiz para quem foi planejado. Então, deixemos os aprendizes se tornarem projetistas, melhor que aprendizes e construtores do conhecimento, melhor ainda que usuários do conhecimento. Eles aprenderão mais no processo".(David H. Jonassen).*

### 3.1 Introdução

Segundo os construtivistas o conhecimento não é um objeto adquirido e possuído por indivíduos. Ele está embutido nas relações sociais e na identidade dos aprendizes, nas conversações e discursos sociais. Nesse sentido, a aprendizagem é o resultado de uma negociação social. Quando os aprendizes compartilham idéias, questionam entre si outras crenças, argumentam sobre o sentido de alguma coisa, eles estão construindo o conhecimento da comunidade, e estabelecendo sua identidade. Quando pessoas constroem socialmente o pensamento, o meio primário do discurso é um relato.

O ponto é simples: aprendizes nem sempre aprendem somente através do professor. Eles podem aprender discutindo problemas, crenças, e expectativas. Muitas pessoas entram em discussões sociais todos os dias, como uma forma de solucionar problemas, articulando sua identidade, e "co-construindo" o conhecimento. Milhões de pessoas gastam muitas horas por semana fazendo a mesma coisa com estranhos, em salas de bate-papo e grupos de usuários, da Internet. Apoiar conversações significativas entre seus aprendizes e outros em volta do mundo podem abrir novas perspectivas, e ajudar na construção colaborativa do conhecimento em novas comunidades de aprendizagem.

### 3.2 Conferência Simultânea

Comunicação simultânea (também chamada de "tempo real"), ocorre normalmente face a face, entre duas ou mais pessoas que se comunicam ao mesmo tempo. Conferências simultâneas ocorrem quando dois ou mais computadores são conectados a um outro na

rede, para compartilhar dados (e.g., texto, gráficos, áudio, vídeo e arquivos), possibilitando pessoas se comunicarem com outras ao mesmo tempo.

Conferências simultâneas podem suportar comunidades em rede, podendo ser constituído de aprendizes e professores, comunicando-se com outros aprendizes, professores, mentores ou especialistas, aumentando, com isso, a capacidade de ensinar ou aprender (Figura 3.1).

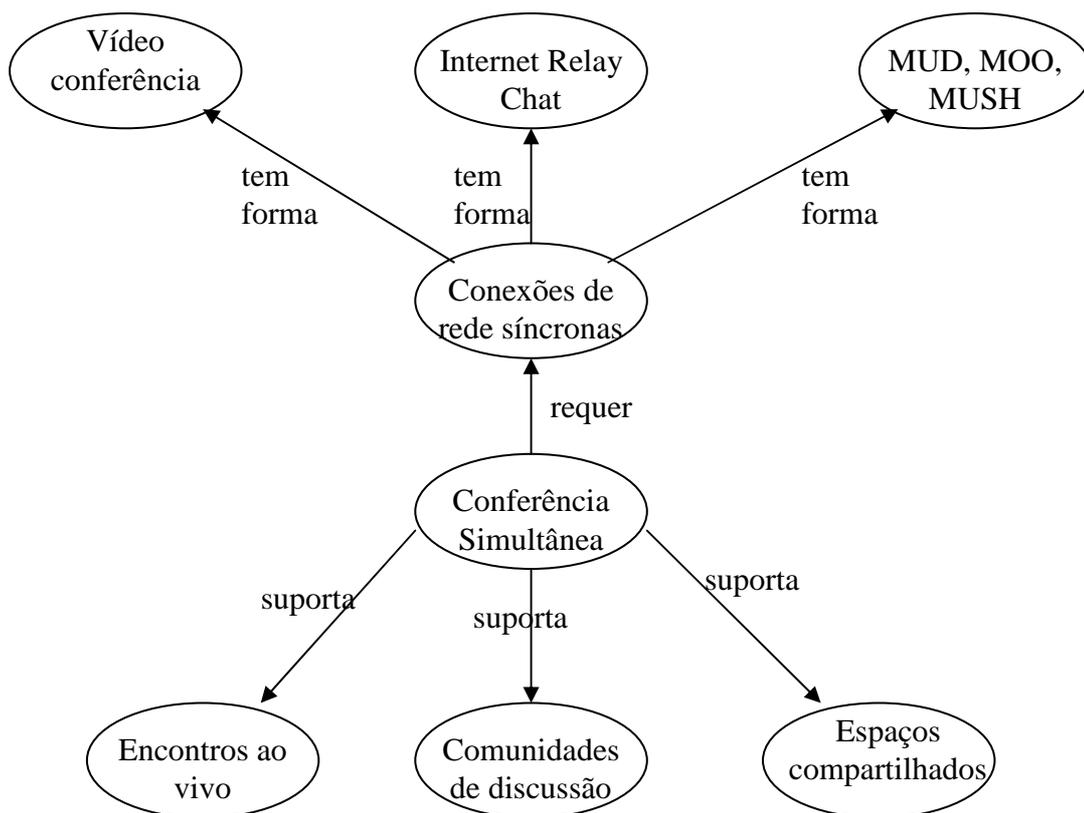


Figura 3.1. Conferência simultânea como uma ferramenta de estudo.[JON.96]

### 3.2.1 Ferramentas para Conferência Simultânea

Internet Relay Chat (IRC), é a forma mais antiga de conferência simultânea. Ele provê interações básicas de mensagens digitadas entre um grupo de usuários (Figura 3.2).

Os membros que compartilham do grupo de discussão aparecem na coluna da direita. A discussão é revelada linha por linha na coluna da esquerda. Essa forma de discussão é bastante popular, mas pode ser confusa para aprendizes mais novos ou novatos em IRC, porque as linhas, tendem a ser adicionadas muito rapidamente quando o número de participantes é incrementado [SCD.97].

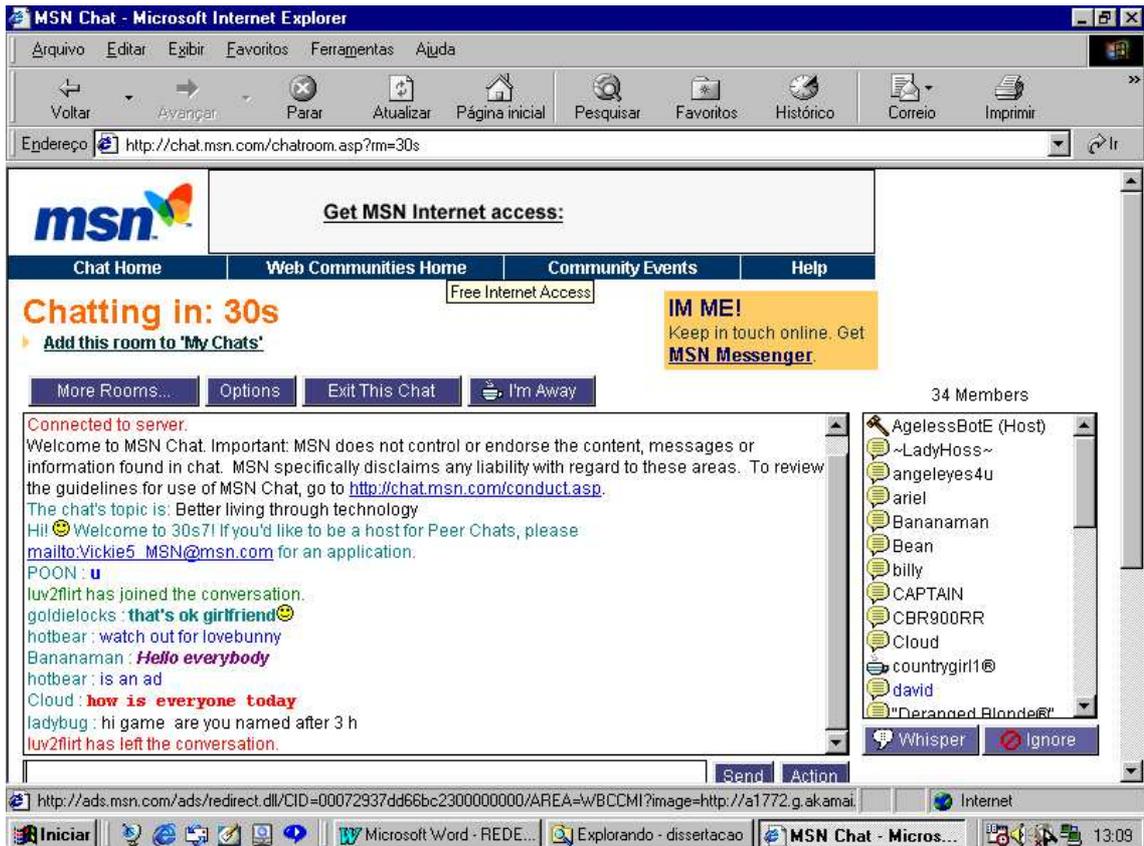


Figura 3.2. Internet Relay Chat.

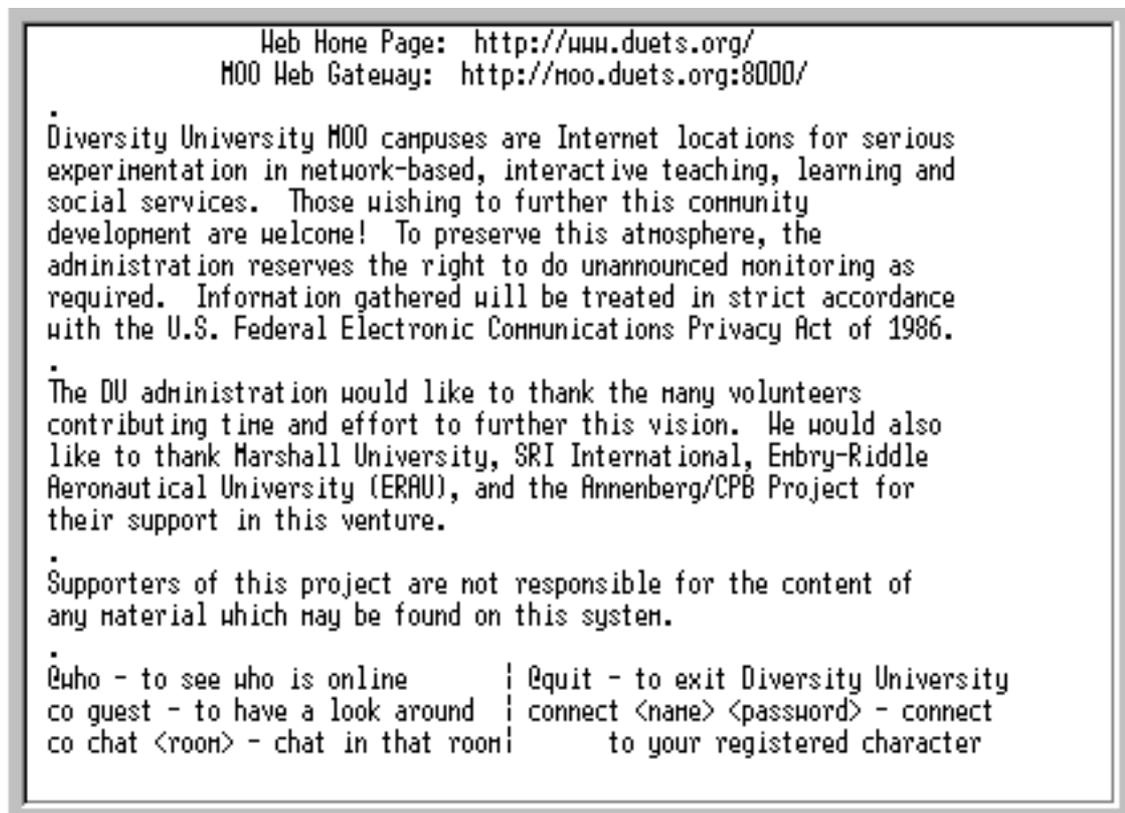


Figura 3.3. MOO da Diversity University.

Extensões de IRC incorporaram ambientes "online" mais elaborados. Esses ambientes compartilhados online ficaram conhecidos como domínios multiusuário (MUDs), MUDs orientado a objeto (MOOs), multiusuário, baseado em Web, orientado a objeto (WOOs), multiusuário, compartilhando alucinações (MUSH). Um benefício dessas aplicações de conferência simultânea é a habilidade de salvar ou imprimir uma conversação inteira. Os formatos mais simples deles são: o formato MUD e MOO (Figura 3.3). Na parte de baixo da figura, os participantes podem usar comandos para saber mais sobre o ambiente, incluindo quais objetos estão presentes (e.g., livros, pinturas, monumentos), quem está presente (e.g., quem, quanto tempo, de onde, quantos anos), e onde mais se pode visitar no ambiente virtual (e.g., o ginásio, o salão de visitas).

Existem outros ambientes virtuais mais sofisticados, que incluem objetos e ambientes visuais, caracteres em bolhas de texto, ou áudio (Figura 3.4). Os participantes podem se vestir (e.g., cantor, policial, rosto sorrindo), viajar a locais diferentes dentro do ambiente, e participar de jogos online (e.g., pinball, backgammon), enquanto conversa.



Figura 3.4. Um ambiente virtual.

Um outro modo mais sofisticado de comunicação simultânea através de rede de computadores, é a videoconferência. Este tipo de conferência, pode incluir a transmissão

ao vivo de dados entre computadores na rede, como: áudio, vídeo, texto, arquivos, telas, figuras, e aplicações compartilhadas (Figura 3.5).



Figura 3.5. Videoconferência usando netmeeting.

O software netmeeting suporta os seguintes tipos de interações:

- ❑ Bate-papo de texto ao vivo (pode ser salvo ou impresso).
- ❑ Mural de desenho e pintura compartilhado (pode ser salvo ou impresso).
- ❑ Imagens de vídeo ao vivo ou pré-gravadas.
- ❑ Áudio.
- ❑ Troca de arquivos (instantâneo).
- ❑ Compartilhamento de aplicações (pode compartilhar o controle de uma aplicação com usuários distantes).

Essas capacidades de conferência eram reservadas a grandes corporações. Agora, um computador (desktop) com modem conectado a Internet, permite esse tipo de comunicação. O netmeeting é um software gratuito que funciona apenas em computadores

Windows. CU-SeeMe (<http://www.cuseeme.com>) é um software gratuito, originalmente desenvolvido na Cornell University, que trabalha em Macintosh e Windows, mas, por enquanto, não suporta transferência de arquivos ou compartilhamento de aplicações.

### 3.2.2 Uso de Conferência Simultânea na Educação

O uso do computador como ferramenta de aprendizagem, tem sido tradicionalmente centrado em métodos individualizados de instrução. Porém, a interação social esta sendo reconhecida como crítica para o processo de aprendizagem [BRU.93][LAV.88]. Aprendizagem significativa é menos focada na transmissão e mais comprometida com a negociação e o discurso [SCE.95].

A tecnologia global de rede pode ajudar os aprendizes a desenvolver habilidades sociais, de leitura, de escrita, de comunicação, e de colaboração através de participação nas discussões online. Participando em um discurso global, os aprendizes são expostos a uma grande diversidade de perspectivas, assim, seus pontos de vista podem se tornar menos paroquial. A aprendizagem se transformou em um processo cultural das comunidades de aprendizes ou profissionais. Usualmente, profissionais existem fora da comunidade de sala de aula. A conferência simultânea permite aos alunos participarem como membros discursivos dessas comunidades.

Ambientes de conversação simultâneos, como: Internet Relay Chat, conferências em PCs, MUDs, MOOs, WOOs, e MUSHs funcionam como ambientes virtuais baseados em gráficos e texto, para imersão dos alunos em uma conversação. Medos como vício, e perda de identidade, tem sido trocada pela importância da sociedade, e a preferência por uma forte interação pessoal, como em grandes reuniões sociais de MUDs. A comunicação simultânea permite aos aprendizes, testarem e refinarem, sobre o que aprenderam na comunidade. A comunidade oferece um "retorno" imediato ao processo de pensamento e escrita.

Acreditando que somente a socialização de aprendizes em salas de bate-papo tem benefício educacional limitado, podemos pensar em melhorar a produtividade das ferramentas

síncronas. Uma proposta seria de que os aprendizes possam planejar um projeto, debater um assunto, solucionar um problema — concentrando-se intelectualmente na discussão. Através da oferta de um contexto para a aprendizagem, os ambientes virtuais são capazes de formar comunidades de alunos, e facilitar uma aprendizagem mais significativa.

Bruffe [BRU.93], descreveu um contexto de instrução usando o Internet Relay Chat. Essa ferramenta de comunicação em tempo real permitiu aos alunos uma sessão com especialistas. É importante deixar claro que um parceiro de comunicação remota estará presente durante a conferência, e delimitar o tempo dessa conversa.

Vacca e Andres [VAC.94] usou tecnologia de conferência em desktops, nos projetos "Global Schoolhouse" e "Scientist on Tap". Um grupo de especialistas voluntários, gradualmente emergiu para interagir com os aprendizes, via e-mail, e via conferência CU-SeeMe. De acordo com um professor participante das atividades, "[a atividade] ... abriu uma janela cognitiva aos nossos aprendizes, criando uma nova "escola em casa virtual", somente possível através da Internet" [VAC.94]. Andres declarou que, "Como o orçamento está contraído, e os recursos de construção da escola reduzidos, a Internet pode conectar aprendizes para a expansão dos recursos de uma "escola em casa global" limitada somente pela banda larga, e popularizada com aprendizes, professores, cientistas, astronautas, e outros especialistas em volta do mundo".

Conferências simultâneas podem suportar alguns exercícios de "jogos de personalidade" interessantes. Jonassem, Peck, e Wilson [JON.99], descreveram uma atividade de "jogo de personalidade" em que, figuras famosas da história, literatura, política são apresentadas. Os aprendizes podem retratar (talvez em forma de teatro) essas personalidades. O ponto importante é que cada um deles (aprendizes) representa crenças, perspectivas, ou atitudes diferentes, sobre o mesmo tópico. Esses debates online podem certamente testar o entendimento de cada aluno sobre os assuntos, e deixar transparecer suas personalidades.

Utilizando-se áreas de trabalho compartilhadas, os usuários podem trabalhar em colaboração, compartilhando o mesmo documento. Esses sistemas costumam ser muito caros, e requerem altas velocidades das redes de computadores, para suportá-los.

Gay e Lentini [GAY.99], identificaram um processo para engajar alunos em trabalhos colaborativos online:

1. Orientação: seja familiar com os recursos de comunicação e informação, e crie materiais. Estabeleça contatos com outros grupos, e seja familiar ao ambiente e ao problema a ser resolvido.
2. Defina o problema: articule o problema: discuta as características emergentes que definem o problema.
3. Subdivida o problema: defina tarefas, objetivos, estabeleça exigências e faça delimitações.
4. Estabeleça regras: identifique os grupos e indivíduos responsáveis em solucionar cada aspecto do problema.
5. Peça informação: pergunte questões técnicas específicas, e procure informação em banco de dados, ou com outros membros do grupo.
6. Compartilhe informação: responda a questões, compartilhe esboços, organize material de vídeo, de forma que outros grupos possam ver os desenhos, e, faça referência a informações relevantes.
7. Monitore: assista aos canais de comunicação para monitorar o progresso de outros grupos, e entender o que eles estão fazendo.
8. Avalie o entendimento: tenha certeza que o conteúdo de todas as partes do tópico foram entendidas.



- ❑ Dificuldade de evitar que a conferência se desenvolva em uma discussão social, porque conferências simultâneas são sociais por natureza.

### 3.3 Conferência Não-Simultânea

Comunicação não-simultânea ocorre quando uma única pessoa pode comunicar a cada tempo. Faxes são assíncronos. A maioria das formas de comunicação por computador é não-simultâneas. O usuário deixa notas, papéis, figuras, ou outro tipo de comunicação para outro. Isto é transmitido em forma digital, e posteriormente decodificado. Existem várias formas de comunicação não-simultâneas (Figura 3.7).

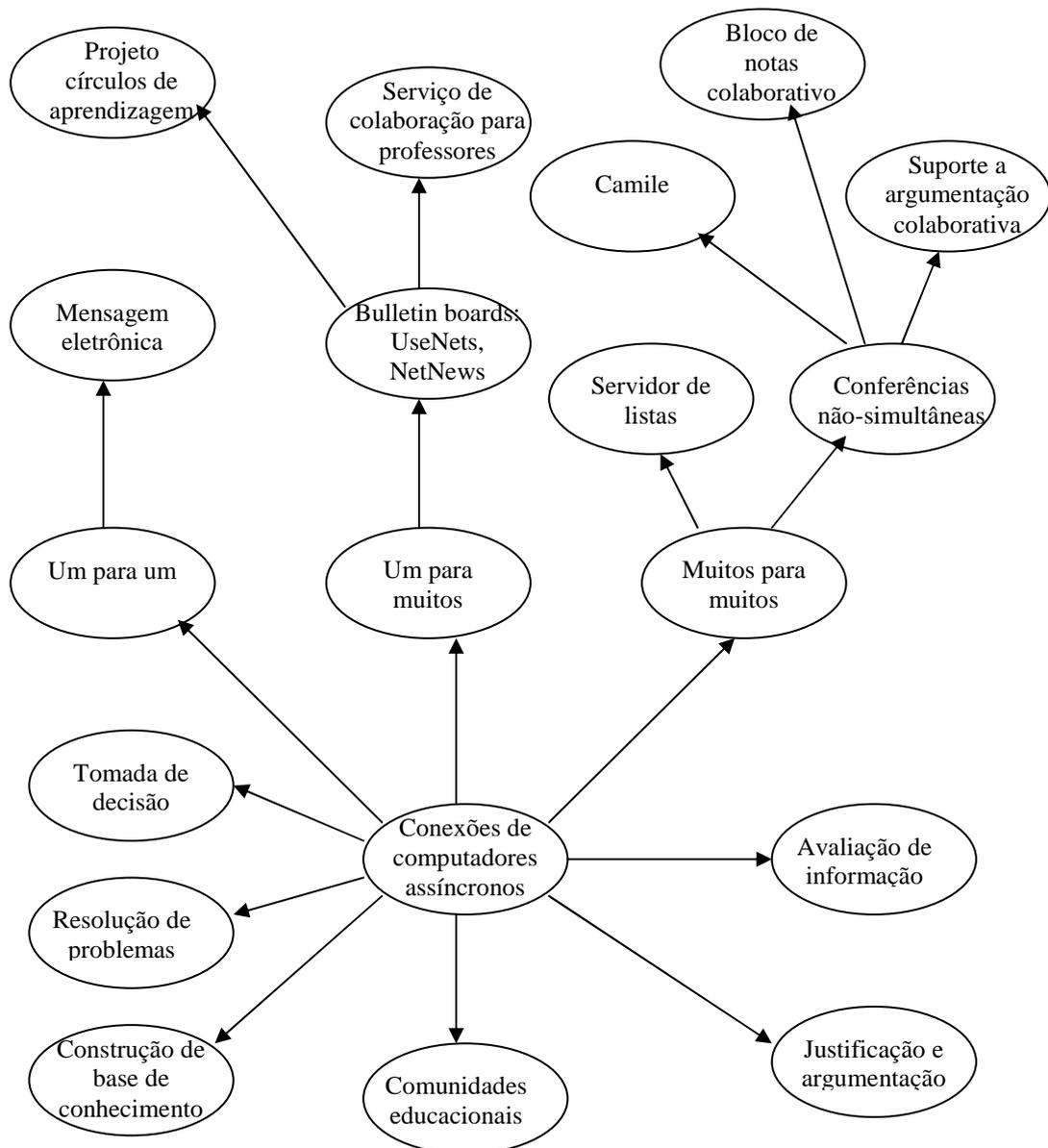


Figura 3.7. Conferência não-simultânea.[JON.96]

### 3.3.1 Ferramentas para Conferência Não-simultânea

A mais conhecida forma de comunicação por computador é o e-mail. Mensagens podem incluir qualquer forma de caracteres de texto. A maioria dos programas de e-mail permite anexar as mensagens, arquivos e documentos formatados, que são enviados ao destinatário. Pode-se também enviar gráficos e programas aplicativos (convertidos em arquivos binários). Dessa forma, podemos considerar que as correspondências para estudo, e educação à distância, são possíveis por e-mail.

Serviços de quadro de avisos (BBS) são programas de computadores, que permitem ao indivíduo postar mensagens em um quadro de avisos metafórico, ou ler mensagens de outros contribuintes, e copiar as mensagens para seus computadores. Usualmente BBSs suportam discursos comunitários, indivíduos que compartilham de um mesmo interesse e querem se comunicar. BBSs provêm acesso a jornais, revistas, classificados ou avisos. Muitas BBSs contém programas aplicativos, como jogos, que podem ser copiados para outros computadores. Em todas essas aplicações, uma pessoa está se comunicando com um grupo de pessoas.

O custo e o tempo perdido em publicação e distribuição de jornais impressos, especialmente acadêmicos, com um número limitado de assinaturas, levam a considerar a publicação eletrônica. Postando artigos em BBS, e restringindo o acesso às pessoas que o assinam, permitem publicações mais rápidas. BBSs também são usadas para palestras. As transcrições das palestras podem ser postadas em quadros de aviso, sendo assim, disponibilizadas para reflexão adicional e análise.

Com os custos altos de viagens, e locais físicos ficando cada vez mais difíceis, profissionais (de diversas áreas) estão começando a conectar computadores, para realizar suas conferências. Essas reuniões virtuais conectam pessoas que podem estar continentes afóra, como se estivessem juntos para discutir.

Conferências não-simultâneas são usadas para criar salas de aula virtuais. Salas de aula virtuais são espaços de comunicação e aprendizagem, localizados dentro de um sistema de computador [HIL.95]. A Pennsylvania State University, oferece cursos de tecnologia da

integração, via conferência WWW. O Campus delibera vários outros graus universitários, e certifica programas, assincronamente.

Conferência não-simultânea também suporta colaboração a longa distância entre alunos. Seja em diferentes continentes, ou em uma escola da cidade; os alunos podem se corresponder em colaboração construindo jornais, noticiários ou outros documentos, solucionando problemas, conduzindo experimentos, debatendo, ou simplesmente trocando idéias e perspectivas.

Conferência não-simultânea é geralmente conduzida via e-mail, com ListServs. Indivíduos podem subscrever para um ListServ, da mesma forma que se subscreve para um jornal. Qualquer mensagem enviada para um ListServ, é automaticamente transferida para todos os subscritores do serviço. As conferências de ListServ focam, usualmente, um único tópico. Elas são efetivas, porque não necessitam de softwares sofisticados, e não usam muito espaço para armazenamento no computador do usuário. Depois dos indivíduos terem localizado um ao outro na conferência geral, eles podem se corresponder individualmente, ou em grupos menores, enviando mensagens um para o outro.

### **3.3.2 Uso das Conferências Não-Simultâneas na Educação**

Conferência não-simultânea é diferente de conferência simultânea, primariamente no nível da reflexão e na construção do pensamento que se é permitido. Conferência simultânea não suporta reflexão antes da fala. Os participantes tendem a registrar seus primeiros pensamentos nas salas de bate papo, e esses pensamentos, não são usualmente coerentes ou convincentes com as idéias, que foram desenvolvidas mais lentamente.

Em conferências não-simultâneas, o usuário lê a resposta, e decide se vai ou não responder, como responder, e quais as conseqüências dessa resposta. Considerando e construindo respostas mais cuidadosamente, envolve uma análise mais profunda do pensamento. A "necessidade de verbalizar todos os aspectos da interação, dentro do ambiente baseado em texto, aumenta as habilidades cognitivas como: a reflexão, e a revisão da aprendizagem" [HAR.93]. Kwon [KWO.98], descobriu que grupos de conferência que faziam tomada de

decisões produziam menos comentários de fora do trabalho, quando comparados com grupos que as faziam face a face.

### 3.3.3 Ferramentas das Conferências Não Simultâneas para a Educação

Os criadores do projeto "Global Learning Cicles", onde as classes de aula americanas foram conectadas a salas de aula em todo o mundo, via "AT&T Learning Network", acreditam que quando os aprendizes escrevem para um grande número de computadores em rede, eles ficam mais motivados em executar as tarefas do que quando fazem isso somente para o professor.

Jornais com autoria em colaboração, coletando arquivos de aprendizes em parceria por todo o mundo, resultam numa melhor gramática e sintaxe [RIE.92]. A equipe do projeto trabalhou com aprendizes do primeiro e segundo grau, distribuindo os trabalhos por grupos em linhas de tempo, cada escola controlava um projeto em cima do currículo da equipe. Ao final, produzia-se uma publicação geral. Isso é um bom exemplo de uma aplicação global de ensino recíproco [BRO.00].

Os aprendizes dos grupos se tornam autores, repórteres, poetas, e pesquisadores, respondendo a requisições de outros círculos de aprendizagem via e-mail. Círculos de aprendizagem são projetados, para expor os aprendizes a pontos de vista diferentes, aumentando a sabedoria multicultural em uma escala global, e desenvolvendo habilidades cooperativas, para lidar com pessoas de diferentes culturas [RIE.92].

Na Catalúnia, Espanha, o Departamento de Educação, viabilizou uma rede com recursos de banco de dados educacionais para professores, e um sistema de conferência em computadores, que permitia aos aprendizes conhecerem, e consultar os aprendizes e professores de outras escolas [SIM.98]. Foram, então, oferecidos cursos inovadores, como planilha eletrônica, e busca online, através de conferências com professores, submetendo suas tarefas ao tutor via transferência de arquivos.

Um número de problemas ocorre, normalmente, em conferências não-simultâneas. Os aprendizes criam respostas para algumas idéias, e deixam que outras se vão. Às vezes, mudam tópicos ou não interpretam corretamente o sentido de outro aprendiz. Assim, novos tipos de conferências não-simultâneas emergiram, com intuito de estruturar a comunicação. Essas conferências estruturadas analisam o tipo de pensamento que o aprendiz pode estar criando, e produzem exigências para que ele comunique dessa forma.

O projeto CoVis tem sido usado para conectar aprendizes nos Estados Unidos, em diálogos sobre ciência. Eles desenvolveram uma conferência conhecida como "Collaboratory Notebook" [PEA.87]. Este é um sistema colaborativo hipermídia, projetado para suportar projetos de ciência dentro das escolas, e destas com outras escolas. Durante o projeto, o professor ou qualquer aprendiz poderá colocar uma questão (conjectura), que pode ser endereçada a outros participantes do país (Figura 3.9). As conversações podem ser públicas ou privadas (Figura 3.8).

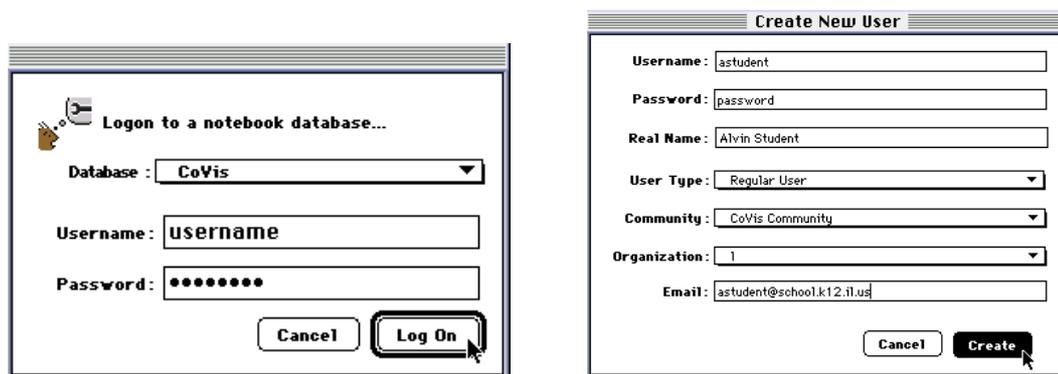


Figura 3.8. Tela de entrada do Collaboratory Notebook.

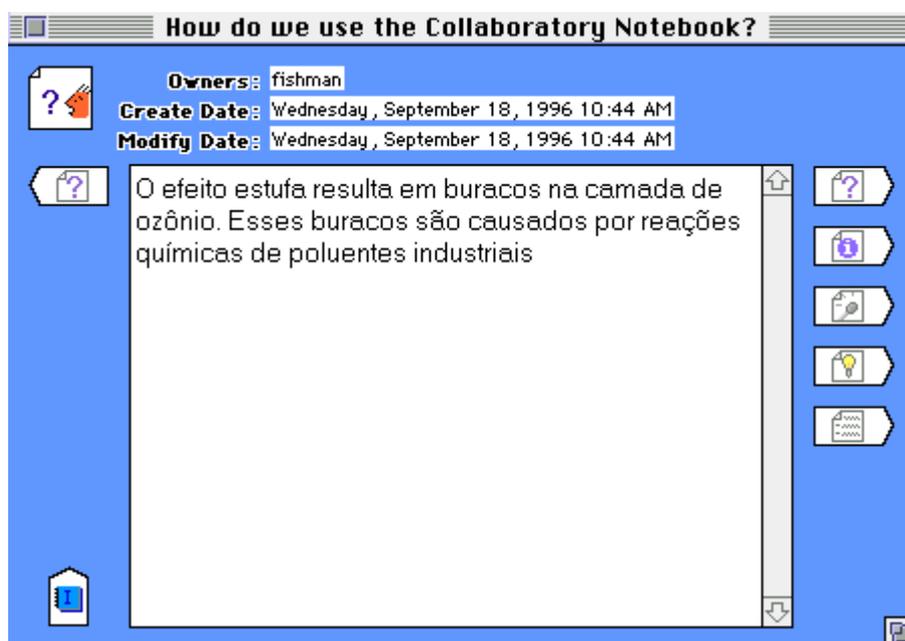


Figura 3.9. Conjectura proposta no Collaboratory Notebook.

A estrutura do Collaboratory Notebook provê conversações, requerendo tipos específicos de resposta para as mensagens. Para responder a conjuntura feita na Figura 3.8, os aprendizes podem apresentar uma evidência, ou desenvolver um plano, conforme os ícones apresentados do lado direito. Essa forma de estruturação da conversação resulta em mais coerência e convicção nas respostas. Esse sistema produz o armazenamento das conversações, para revisão e reconsideração dos aprendizes.

O projeto "CaMILE Notebase", desenvolvido no EduTech Institute da Georgia Tech, também é uma base de notas colaborativa, onde aprendizes postam notas associadas a grupos de discussão. Cada nota é adicionada em resposta a nota de outra pessoa que contribuiu para a discussão. Os aprendizes entram com uma nota de comentário (Figura 3.10), sobre uma discussão em andamento (Figura 3.10). Além do texto, os aprendizes podem incluir um vídeo (Figura 3.10), como evidência, e ligações para figuras, sons, gráficos, ou qualquer outro tipo de arquivo.

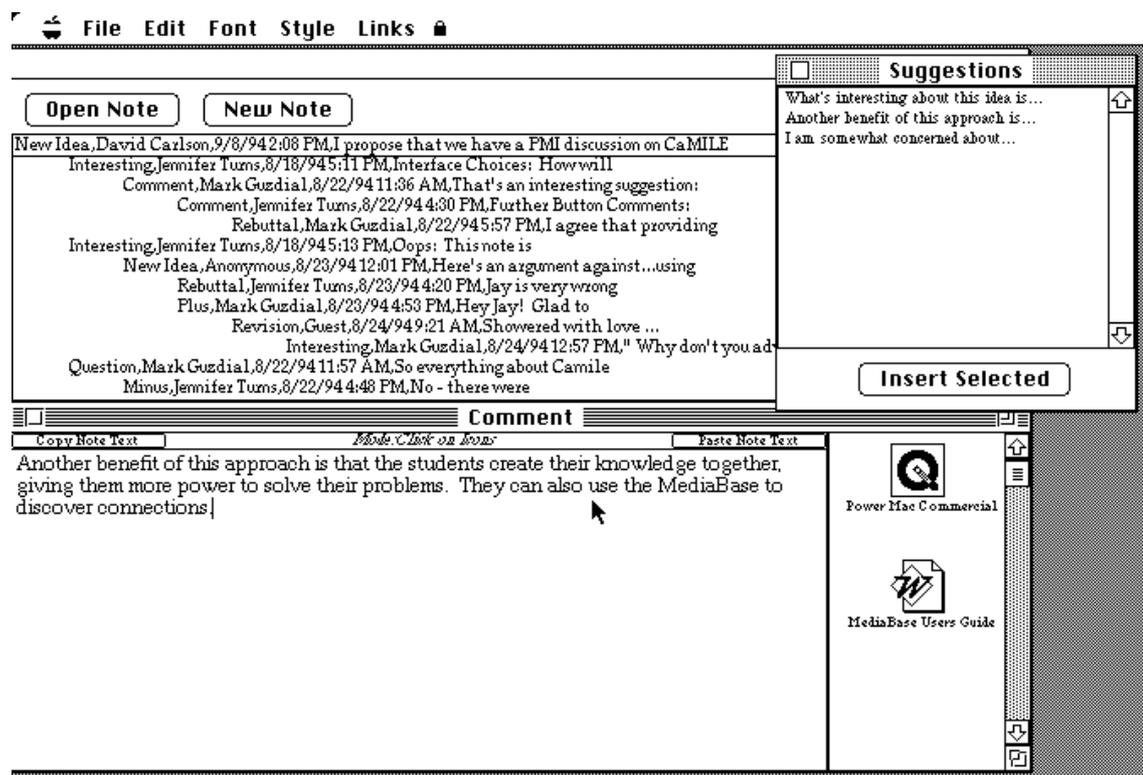


Figura 3.10. Um comentário adicionado à base de notas do CaMILE.

O CaMILE também provê espaço para fazer sugestões (Figura 3.10), baseado no comentário da nota. Quando um aprendiz lê alguma discussão, e queira fazer algum comentário, ele deve especificar que tipo de resposta ele vai dar (Figura 3.11).

Uma versão do CaMILE para a Internet foi desenvolvida com o nome de WebCaMILE, com as mesmas funcionalidades, mas com a interface diferente.

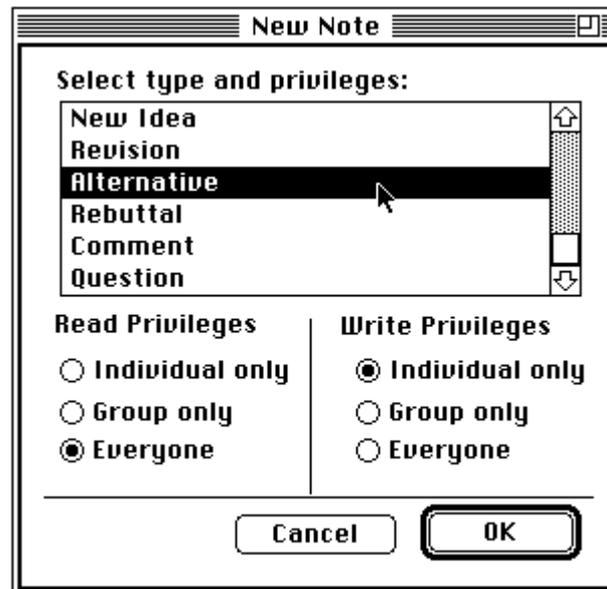


Figura 3.11. Especificando o tipo de resposta do CaMILE.

### 3.3.4 Vantagens das Conferências Não-Simultâneas

- Um objetivo primário da educação, segundo algumas teorias, é socializar o aprendiz. Rede de computadores é um grande agente para propagação e disseminação de habilidades sociais [COL.00].
- Para Hiltz [HIL.95], as conferências não-simultâneas produzem mais interação, e envolvem mais troca de informações entre os aprendizes, se comparado à face-a-face. Isto acontece, provavelmente, porque os indivíduos podem se manter anônimos, reduzindo o medo e aumentando a eficácia acadêmica. Os alunos acham que têm melhor acesso aos instrutores através de conferência e que os cursos são mais interessantes do que os tradicionais.
- A conferência parece aumentar a efetividade da colaboração entre os alunos, porque ela utiliza acesso a outros grupos participantes, eliminando distorções sociais e barreiras, contribuindo para um senso de informalidade, encorajando uma forte identidade do grupo [PFA.98].

- ❑ Berger e Collins [BER.98], discutiram sobre a independência de tempo da conferência não-simultânea. Conferências são abertas 24 horas por dia, sete dias por semana. Esse tempo pode ser usado para refletir sobre a mensagem, antes de respondê-la. Os aprendizes podem fazer seus trabalhos, quando acharem mais conveniente.
- ❑ Os aprendizes podem perguntar e discordar sem se envolverem excessivamente em conflitos [PHI.98]. Isto ajuda muito os aprendizes mais introvertidos, tímidos, e pessoas pensativas.
- ❑ A conferência suporta aprendizagem colaborativa. Quando trabalham em grupos, os aprendizes participam mais uniformemente e executam mais tarefas objetivas [SCO.99].
- ❑ Quando comparado com salas de aula tradicionais, onde os professores contribuem com até 80% da permuta verbal, as conferências de computadores mostram os professores contribuindo com apenas 10 a 15 % do volume de mensagens [HAR.88].

### 3.3.4 Limitações das Conferências Não-Simultâneas

- ❑ As complexas técnicas para conferência não-simultânea, aprender a usar um software novo e as dificuldades de conexão ao sistema são frustrantes e induzem a ansiedade. Sistemas que combinem disponibilidade de recursos com uma interface amigável são o ideal.
- ❑ Os usuários devem ser habilidosos com a comunicação. Nem todos os aprendizes têm essa facilidade.
- ❑ O modo primário de entrada é texto. Isto significa que os usuários devem possuir uma certa habilidade em digitação. Isto é problemático para muitos aprendizes.
- ❑ A interface com o usuário, na maioria dos sistemas, não é amigável, dificultando o uso.
- ❑ Conferências com indivíduos de outros continentes podem atrasar em horas ou dias. O atraso pode afetar o impacto de certas mensagens.
- ❑ A participação de grupos de usuários varia. É desejável que haja uma total participação dos aprendizes, como são as discussões em sala de aula. Algumas pessoas podem ter fobia a tecnologias, ou ansiedade nas comunicações.

- ❑ Em situações de tomada de decisão em grupo, a mediação do computador, pode produzir decisões mais polarizadas do que em situações face-a-face [LEA.00]. As tomadas de decisões ficam mais lentas [KIE.86].
- ❑ As conferências podem ampliar inseguranças sociais [MCG.98].
- ❑ As linhas de comunicação e os equipamentos não são 100% confiáveis, podendo haver perda de dados ou atraso nas conversações. Esses problemas tendem a frustrar os usuários e reduzir a participação.

### 3.4 Hipermídia

Hipermídia é o casamento da multimídia com o hipertexto. O hipertexto é baseado no termo hiper, que quer dizer acima, além, super—mais que normal. O texto normal é linear, e é construído para ser lido do início ao fim. O hipertexto refere-se a um método não seqüencial, não linear, para organização e exibição do texto [JON.93]. Hipertexto foi projetado para permitir aos leitores acessar informações de um texto, da forma que mais lhe faça sentido [NEL.98].

Hipertexto é um "supertexto" porque o leitor tem maior controle sobre o que lê, e sobre a seqüência de leitura. Assim, a organização da leitura do texto é personalizada por quem está lendo, e não da forma como foi imposta pelo autor. Hipermídia é um hipertexto com múltiplas formas de representação (texto, gráfico, som e vídeo)(Figura 3.12).

A característica mais intrigante da hipermídia é o nó, que consiste de ligações para texto, figura, som, etc. Em alguns sistemas, o usuário pode adicionar ou mudar a informação em um nó, ou criar ele mesmo seu próprio nó de informação. Com isso, a hipermídia pode ser uma base de conhecimento dinâmica, representando novos e diferentes pontos de vista. A arquitetura hipermídia é aberta. O mesmo conjunto de nós pode ser organizado de várias maneiras diferentes, refletindo diferentes orientações conceituais.

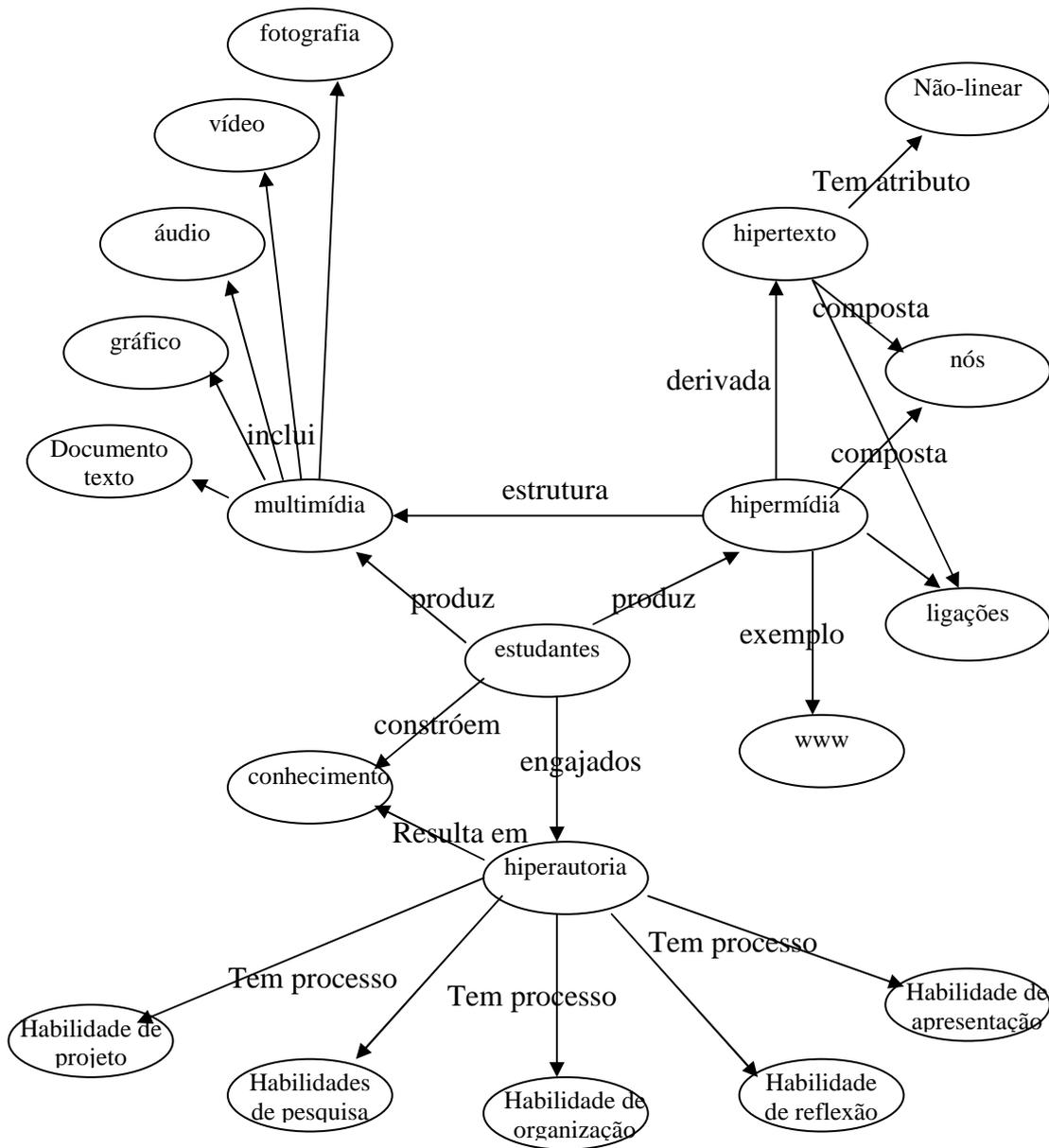


Figura 3.12 Hipermedia.[JON.96]

### 3.4.1 Uso da Hipermedia na Educação

A construção hipermedia é um exemplo de "esboço do conhecimento", reformulando o processo educacional onde: o conhecimento se torna informação, e o professor o transmissor dessa informação [PER.95]; passando para, professores e aprendizes como colaboradores no processo de construção do conhecimento. A composição de hipermedia coloca os aprendizes no lugar dos autores, assim, eles podem construir seu próprio entendimento das coisas, ao invés de interpretar o entendimento do mundo do professor.

Alunos participam e interagem com o ambiente hipermídia, inventando e negociando seus próprios pontos de vista sobre o assunto [JON.96]. Vejamos alguns exemplos:

- À medida que aumenta as experiências dos aprendizes com a construção hipermídia, seus documentos se tornam mais integrados, ao invés de meras anotações de texto [KIE.86] Os aprendizes ficam entusiasmados com as produções hipermídia, acreditando que aprendem mais porque entendem melhor as idéias.
- Simon [SIM.98], mostrou que aprendizes que construíram hipermídia, aparentemente, desenvolveram uma proficiência para organizar o conhecimento sobre um assunto, de uma forma mais inteligente.
- A aprendizagem do espanhol é facilitada pela construção de apresentações hipermídia em Espanhol, porque elas representam o conhecimento pessoal de cada aprendiz [SCE.95].

### 3.4.2 Ferramentas para Sistemas Hipermídia

Lea [LEA.00] desenvolveu uma ferramenta, HyperAuthor, e empregou oito graduados para projetar lições de história. O conhecimento, ele acredita, desenvolve-se em um processo de planejamento. Não é algo para ser transmitido do professor para o aluno, assim, os alunos deveriam compor seu próprio sistema hipermídia, num processo que ele chama de HyperComposition.

Lea [LEA.00] conduziram outro estudo com nove aprendizes graduados, desenvolvendo um sistema hipermídia sobre o estilo de vida na época da primeira guerra mundial (1870 a 1920), imigração e imperialismo. Eles acharam resultados similares no comportamento do desenvolvimento das tarefas com relação ao tempo, e a percepção dos aprendizes com relação aos estágios de transformação do projeto.

Projetar e desenvolver sites Web têm sido o modo preferido de construção hipermídia atualmente. Aprendizes, em todo o mundo, estão desenvolvendo sites Web sofisticados e fornecendo informação importante para o usuário. Aprendizes da Andrew Robinson Elementary School em Jacksonville, Flórida, criaram uma "escola em casa virtual"

(<http://www.rockets.org/>) [JON.99]. O site provê um passeio pela escola, fornecendo informações sobre o que os aprendizes estão aprendendo.

Os desenhos dentro do esboço do prédio são usados como mapas, os usuários clicam na figura, e passeiam virtualmente na escola. Por exemplo, (Figura 3.13), mostra o corredor. Clicando na porta, entra-se dentro do ambiente.

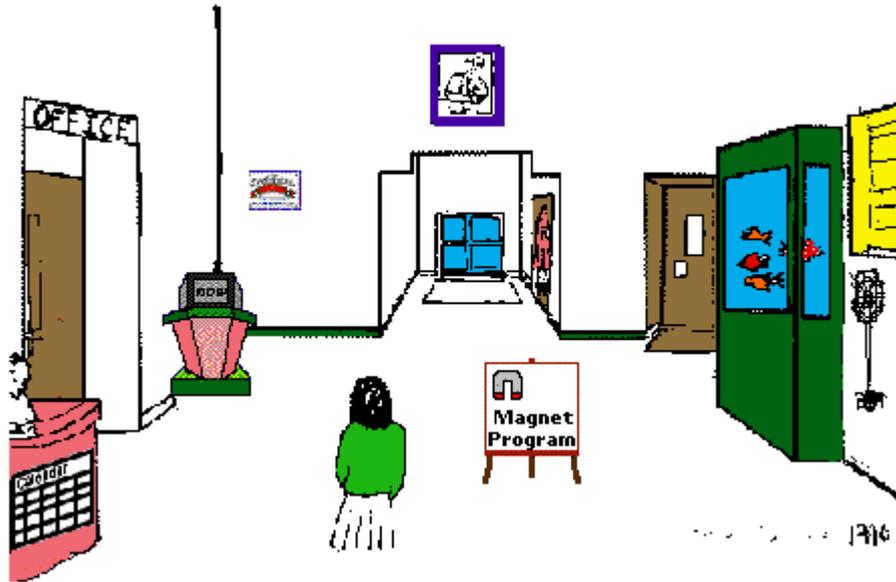


Figura 3.13. Escola virtual de Andrew Robinson.

O próximo passo é um passeio pela escola com realidade virtual (Figura 3.14). O software para produzir isto se chama RealVR. A foto pode se mover em qualquer direção.



Figura 3.14. Corredor virtual.

Pressionando e movendo o botão do mouse, pode-se ver em volta (até 360 graus), e para cima e para baixo. Pode-se também usar o zoom, para aumentar ou diminuir a imagem.

### 3.4.3 Vantagens dos Sistemas Hipermedia

- ❑ Os aprendizes são mais engajados mentalmente desenvolvendo seus próprios materiais do que os estudando.
- ❑ A hipermedia permite representar palpavelmente, múltiplas abstrações de idéias [HAR.93].
- ❑ Os aprendizes que constroem hipermedia, criam representações de seu próprio entendimento, usando o seu próprio modo de expressão.
- ❑ Os aprendizes ficam mais motivados porque são donos do produto, deixando-os orgulhosos disso.
- ❑ A construção de hipermedia, muda à orientação dos professores e aprendizes sobre a noção de que conhecimento é informação, e da função do professor como transmissor dessa informação [LEA.00].
- ❑ Projetar o conhecimento na forma de apresentações hipermedia, promove o desenvolvimento de teoria crítica do conhecimento (nem todo projeto é bem sucedido), e do pensamento crítico, como definir a natureza do problema e executar um programa para resolvê-lo.

### 3.4.4 Limitações dos Sistemas Hipermedia

- ❑ A construção de hipermedia é um processo que consome tempo.
- ❑ Existem mais requisitos de equipamentos e softwares para a construção de hipermedia, do que para outras ferramentas. Para integrar áudio, vídeo, e gráficos em uma apresentação, é necessário equipamentos como: scanner, microfones, câmeras de vídeo, câmeras fotográficas digitais.

## 3.5 Redes Semânticas como Ferramentas da Mente

Redes semânticas são usadas para representar o conhecimento do aprendiz, organizando as idéias básicas do contexto sendo estudado. A aprendizagem significativa requer que os aprendizes conectem novas idéias no conhecimento que eles já possuem [GUH.90]. Mapas

conceituais ajudam a organizar o conhecimento do aprendiz, integrando informação progressivamente mais complexa. O processo de criar redes semânticas engaja o aluno na análise das suas próprias estruturas de conhecimento, ajudando a integrar o novo saber ao que ele já conhece.

Ferramentas de redes semânticas são ferramentas da mente para representação estruturada do conhecimento [JON.93]. A psicologia cognitiva classifica as formas de conhecimento em declarativo e procedural. O conhecimento declarativo expressa as idéias (fatos) lembradas. O conhecimento procedural representa a maneira de operacionalizar o conhecimento declarativo.

### **3.5.1 Guia de Estudo para Redes Semânticas**

Redes semânticas são ferramentas de aprendizagem. Ao invés de memorizar o conteúdo, os aprendizes devem identificar os conceitos mais importantes de um capítulo e gerar uma rede semântica como estratégia de pesquisa. Depois eles podem comparar suas redes semânticas com as de outros aprendizes e raciocinar sobre suas conexões e escolhas; ou redes semânticas de diferentes capítulos podem ser combinadas ou fundidas em estudo para exames volumosos.

### **3.5.2 Reflexão do conhecimento**

O mapeamento de conceitos provê um método valioso para o aprendiz auto avaliar sua aprendizagem, refletindo mudanças no seu próprio conhecimento. Pensando sobre as conexões entre conceitos, avaliando suas ligações através de aprimoramento, revendo as relações lógicas entre conceitos, e planejando como organizar conceitos significativos; determinam valiosa evidência de auto-reflexão e raciocínio cognitivo. Reorganizando o que se sabe, ajuda-os a entender e tentar novas soluções para problemas.

Redes semânticas fornecem uma forma fácil e rápida de organização e seqüenciamento de idéias. Produzir redes semânticas reflete uma reorganização da memória semântica, visto que as redes semânticas descrevem uma teoria sobre o que o aprendiz sabe. Assim, as redes semânticas do aprendiz geradas depois da instrução refletem o crescimento da sua estrutura de conhecimento.

### 3.5.3 Instrução com Redes Semânticas

Vejam algumas observações para a construção de redes semânticas em sala de aula:

1. Faça um plano e fixe um ponto de vista para analisar um domínio.
2. Identifique conceitos importantes.
3. Crie, defina, e elabore nós.
4. Construa elos e ligue conceitos.
5. Continue expandindo a rede.
6. Alunos refletem sobre o processo.

Alunos utilizando redes semânticas têm mais habilidades no pensamento crítico, do que no criativo ou complexo. É uma ferramenta que focaliza o aprendiz em uma análise crítica de idéias de um certo domínio do conhecimento. A integração do conhecimento emprega mais habilidades ao pensamento criativo. A arte da imaginação está mais restrita, por causa das limitações visuais impostas pelos programas.

Todas as habilidades do pensamento complexo são usadas no planejamento de outras construções do conhecimento. O planejamento das apresentações e construções do conhecimento faz com que o aprendiz realize uma grande quantidade de tomada de decisões.

### 3.5.4 Avaliação para Redes Semânticas

Alguns passos para avaliar as redes semânticas dos aprendizes, são:

1. Compare a rede semântica do aprendiz com a de um experiente (professor).
2. Determine o crescimento do conhecimento do aprendiz.
3. Aceite diferentes pontos de vista do aprendiz.
4. Compare a rede do aprendiz com os objetivos do curso.

Os programas devem ser capazes de representar conceitos como nós, serem capazes de posicionar estes nós na tela, fornecer uma disposição hierárquica de sub-mapas sob mapas, e serem capazes de mostrar tipos de ligações na tela.

### 3.5.5 Vantagens das Redes Semânticas como Ferramentas da Mente

- São fáceis de usar.
- Apresentam representações espaciais do conteúdo, o que ajuda na memorização.
- Aumentam a compreensão e memorização das idéias, ajudando o aprendiz a construir o conhecimento estruturado.
- Melhoram a performance da solução de problemas do aprendiz.

### 3.5.6 Limitações das Redes Semânticas como Ferramentas da Mente

- Não são mapas reais da mente, mas representações do que pensamos.
- O conhecimento representado é dinâmico, ele muda dependendo do contexto e das experiências de quem produziu a rede.
- As redes propostas na mente são muito mais complexas do que qualquer coisa que possa ser representada em mapas conceituais (são multidimensionais).

## 3.6. Conclusão

Os programas de redes semânticas fornecem um conjunto de ferramentas de conceitualização gráfica, para criação de mapas conceituais. Esses mapas conceituais representam a estrutura das idéias na memória ou no conteúdo do domínio. Redes semânticas engajam o aprendiz em uma análise do conteúdo do domínio, ajudando-o a organizar seu conhecimento para melhor compreensão e memorização.

As Conferências simultâneas em computadores estão se tornando uma realidade. Muitas novas aplicações serão evidentes nesses próximos anos. As implicações desse tipo de aprendizagem ainda não foram estudadas em profundidade.

Recentemente, o fenomenal crescimento da World Wide Web (WWW), mudou a face da computação. A maioria dos computadores pessoais são estações multimídia. Na década de 1990, para os círculos educacionais que utilizaram computação, a resposta foi multimídia. Os produtores comerciais de software focaram seus produtos para publicação multimídia na Web. É pouco provável que essas ferramentas afetem substancialmente a aprendizagem

nas escolas, a não ser que os aprendizes sejam os projetistas. A hipermídia provê um sistema conceitual útil para o planejamento de materiais multimídia.

## 4. Indexação no Contexto de Grafos Conceituais

### 4.1 Introdução

Para a indexação dos grafos conceituais na arquitetura do ambiente VEDA, foi projetado o uso de cinco níveis de tratamento, são eles:

- *Ontologia* é o primeiro passo para a construção de uma base de conhecimento. Na teoria de banco de dados, as categorias ontológicas são usualmente chamadas de domínios, em lógica e Inteligência Artificial elas são chamadas de tipos e em sistemas orientado a objetos elas são chamadas de classes. A seleção das categorias ontológicas define tudo que poderá ser representado na aplicação. Qualquer ontologia incompleta, restrita, ou distorcida na estrutura limitará a generalidade de todo o programa e base de conhecimento.
- *Axiomas* podem ser definidos para as categorias ontológicas de uma base de conhecimento através de regras de inferência.
- *Relações de Conformidade* limitam as possíveis combinações entre tipos e marcadores, atuando no nível local dos conceitos.
- *Grafos Canônicos*, como as relações de conformidade, previnem a geração de grafos conceituais sem sentido, mas atuando previamente nos níveis dos sub-grafos. Os grafos canônicos impõem restrições aos conceitos e relações.
- *Restrição/Expansão* (Regras de Formação Canônica) generalizam e especificam os grafos canônicos.

A ontologia é a base de fundamentação de toda a indexação. Ela proporciona um reticulado hierarquizado de categorias ontológicas (tipos de conceitos e relações). A forma de reticulado define que os *nós* podem possuir mais de um *nó* pai (*nó* supertipo de um tipo). Desta forma, a hierarquização das categorias ontológicas da ontologia proporciona herança múltipla. Os tipos de conceitos e relações podem possuir axiomas que determinam suas propriedades, propagando aos seus subtipos em forma de herança. As relações de conformidade restringem as referências dos tipos de conceitos, propagando aos seus subtipos em forma de herança. Os grafos canônicos determinam padrões para a estruturação de tipos de conceitos e relações, delimitando a construção dos grafos conceituais, e propagando os padrões em forma de herança aos seus subtipos. As regras de

formação canônica de restrição e expansão utilizam as propriedades de herança do reticulado ontológico para generalizar ou especificar os tipos de conceitos e relações. A restrição limita o campo de domínio dos grafos conceituais, passando de *nós* pai para *nós* filho. A expansão generaliza o campo de domínio dos grafos conceituais, passando de *nós* filho para *nós* pai.

## 4.2 Definição de Conceito

Todo conceito tem um tipo  $t$  e uma referência  $r$ . Nesta definição abstrata, a representação do tipo e referência não está especificado. Na armazenagem do computador, eles devem ser representados como um par de ponteiros, um apontando para a especificação do tipo e o outro apontando para a especificação da referência. O tipo é usualmente escrito do lado esquerdo do conceito e a referência do lado direito. Os dois campos são separados por dois pontos. No conceito [Ônibus], "*Ônibus*" é o tipo, sendo branca a referência, que representa um quantificador existencial. No conceito [Pessoa: João], "*Pessoa*" é um tipo, e a referência "*João*" é o nome de alguma pessoa. Há diversas espécies de referência, algumas das quais são indicadas no quadro abaixo:

Tabela 4.1. Exemplo de referências.

Referência	Notação	Leitura
Genérico	[Gato: *] ou [Gato]	Algum gato
Individual	[Gato: #]	O gato
Nome	[Gato: Tom]	Um gato chamado Tom
Conjunto	[Gato: {Tom, Felix}]	Os gatos Tom e Felix
Universal	[Gato: $\forall$ ]	Todos os gatos
Quantidade	[Gato: @2]	Dois gatos
Medida	[Peso: @2kg]	Um peso de 2 quilos

Os referentes básicos são os marcadores individuais #1, #2,..., e o marcador genérico \*.

Uma variável é simplesmente um marcador genérico ao qual se deu um nome. A notação é '\*nome'. As variáveis aparecem no campo referente e podem ser usadas na notação linear. A frase "*João gosta de si próprio*" poderia ser representada como:

[Pessoa: João = \*x]  $\rightarrow$  (Agente)  $\rightarrow$  [Gostar]  $\leftarrow$  (Obj)  $\leftarrow$  [Pessoa: \*x].

### 4.3 Definição de Relação Conceitual

Toda relação conceitual  $r$  tem uma relação tipo  $t$  e um inteiro positivo  $n$ , chamado valência.

- O número de arcos que pertencem a  $r$  é igual a sua valência  $n$ . Uma relação conceitual de valência  $n$  é dita ser  $n$ -área, e seus arcos são numerados  $1, 2, \dots, n$ .
- Para toda relação conceitual  $n$ -área  $r$ , existe uma seqüência de  $n$  tipos de conceitos  $t_1 \dots t_n$ , chamados de *assinatura* de  $r$ .
- Toda relação conceitual de um mesmo tipo de relação tem a mesma valência  $n$  e a mesma *assinatura*  $s$ .

Nos exemplos, *Agente*, *Dest*, e *Sobre*, são relações do tipo díades. Exemplos de tipos de relações monadas incluem *Psbl* para possibilidade e *Pass* para a sentença passada. O exemplo abaixo (Figura 4.2) mostra a relação entre (Entre) como um exemplo de relação triádica (valência três), os dois arcos são ligados a dois conceitos que estão do outro lado de um terceiro. Esse grafo pode ser lido: "*Uma pessoa esta entre uma pedra e um lugar rudimentar*".

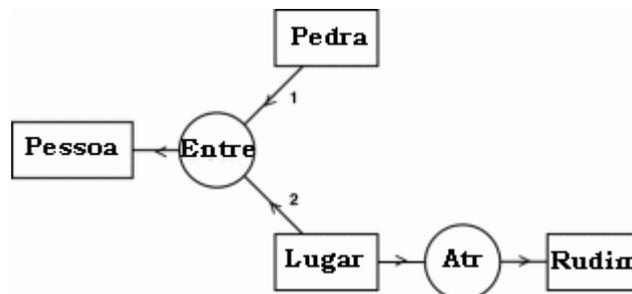


Figura 4.2: Uma relação conceitual de valência três.

A *assinatura* da relação representa uma limitação sobre os tipos de conceitos que podem ser ligados aos seus arcos. Para a entidade *Agente*, a *assinatura* indica que o tipo de conceito ligado ao primeiro arco deve ser *Ação* ou algum subtipo como *Ir*, e o tipo de conceito ligado ao segundo arco deve ser *Animado* ou algum subtipo como *Pessoa*. Para a entidade *Entre*, a *assinatura* mostra que todos os três conceitos devem ser do tipo geral entidade, o que não impõe nenhuma restrição de tipo.

A notação linear (Figura 4.2), pode ser representada da seguinte forma:

[Pessoa] -> (Entre) -

->1- [Pedra]

->2- [Lugar] <- (Atr) <- [Rudim].

O hífen depois da relação indica que o seu outro arco continua nas linhas subseqüentes. Os dois arcos que apontam adiante da relação são numerados 1 e 2.

#### 4.4 Definição de Contexto e Co-referência

Um contexto é um conceito cujo tipo é *Proposição*, o qual pode ser omitido, como acontece nos exemplos seguintes; e cujo referente ou é o marcador genérico (\*) ou um conjunto de grafos. Os contextos são usados para exprimir afirmações e crenças. Nesse sentido, algumas relações como *Neg* (negação), *Nec* (necessidade) e *Psbl* (possibilidade) só podem ser ligadas a contextos.

Os elos de co-referência são usados para ligar conceitos que se encontram em grafos e/ou em contextos diferentes, mas que denotam o mesmo indivíduo. Visualmente são representados por linhas pontilhadas. Por exemplo, "Rose não é Rosa", é representada pelo grafo abaixo (Figura 4.3):

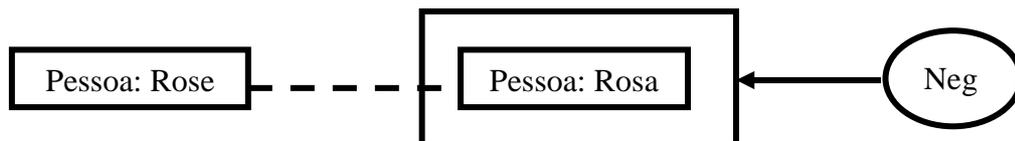


Figura 4.3. Exemplificação de um contexto e uma co-referência.

Cuja notação linear é:

[Pessoa: Rose] ; (Neg) -> [Proposição: [Pessoa: Rosa = Rose] ].

(os ponto-e-vírgulas separam os sub-grafos).

## 4.5 Definição de Tipos de Conceitos

Os tipos de conceitos formam uma hierarquia, mais precisamente um reticulado, cuja relação de ordem é representada por  $\leq$ . O topo é o tipo universal  $T$  e a base é o tipo absurdo  $\perp$ . Por exemplo,  $Comunicar < Dar < Ato$  significa que *Comunicar* é um subtipo de *Dar* que, por sua vez, é um subtipo de *Ato*, pelo que,  $Comunicar < Ato$ .

Para impedir a formação de grafos sem sentido, existem dois mecanismos para especificar restrições sobre quais grafos são permitidos: a relação de conformidade, e os grafos canônicos.

## 4.6 Definição de Relação de Conformidade

A relação de conformidade limita as combinações possíveis entre tipos e marcadores. Dito de outro modo, um conceito  $t:m$  só será permitido se o marcador  $m$  estiver em concordância ao tipo  $t$ . Ao especificar a relação de conformidade, o engenheiro do conhecimento estará definindo quais os conceitos que fazem sentido para a aplicação em causa. No entanto, é preciso observar algumas restrições: o marcador genérico "\*" é conforme a qualquer tipo; e, se um marcador individual for conforme a um tipo  $t$ , também o é em relação a todos os supertipos de  $t$ . Isto significa que, para qualquer tipo  $t$ , o conceito  $t:*$  é permitido. Assim, se por exemplo, *Pessoa: #123* for considerado conforme, então *Animal: #123* também é, visto ser  $Pessoa < Animal$ . A razão destas restrições é que, basicamente, a relação de conformidade serve para especificar quais as instâncias possíveis de cada tipo.

## 4.7 Definição de Grafos Canônicos

Grafos canônicos não atuam no nível local dos conceitos, mas anteriormente, no nível dos sub-grafos. Enquanto a relação de conformidade enumera as combinações possíveis entre tipos e marcadores, os grafos canônicos impõem restrições de seleção sobre conceitos e relações, evitando assim a geração de grafos que não fazem sentido. Por isso, tem que

existir um grafo canônico para cada tipo de conceito ou relação indicando como é que outros conceitos e relações se relacionam com ele. Por exemplo, os grafos canônicos para os três tipos acima mencionados, poderiam ser:

$$[\text{Ato}] \rightarrow (\text{Agente}) \rightarrow [\text{Animado}].$$

Um ato tem como agente um ser animado (Figura 4.4).



Figura 4.4. Exemplo de grafo canônico.

$$[\text{Entidade}] \leftarrow (\text{Obj}) \leftarrow [\text{Dar}] \rightarrow (\text{Rcpt}) \rightarrow [\text{Animado}].$$

Dar envolve um objeto (uma entidade) e um recipiente (um ser animado) (Figura 4.5).



Figura 4.5. Exemplo de grafo canônico.

$$[\text{Informação}] \leftarrow (\text{Obj}) \leftarrow [\text{Comunicar}] \rightarrow (\text{Inst}) \rightarrow [\text{Entidade}].$$

O objeto comunicado é a informação e o instrumento utilizado é uma entidade (Figura 4.6).



Figura 4.6. Exemplo de grafo canônico.

No que se segue, assume-se que as relações hierárquicas entre os tipos ocorrentes, são: *Informação* < *Entidade* e *Animado* < *Entidade*; além de *Comunicar* < *Dar* e *Dar* < *Ato*. Não se considera ser *Ato* < *Entidade* nem *Informação* < *Animado* ou vice-versa.

As chamadas regras de formação canônica derivam novos grafos canônicos a partir dos já existentes. Elas também definem uma hierarquia sobre os grafos conceituais: se o grafo  $g_1$  for canonicamente derivável do grafo  $g_2$ , então  $g_1$  é uma especialização de  $g_2$  (escrevendo  $g_1 \leq g_2$ ) e  $g_2$  é uma generalização de  $g_1$ .

### 4.7.1 Definição: Regras de Formação Canônica

Todas as operações sobre grafos conceituais são baseadas em combinações de seis regras de formação canônica; cada uma delas executa uma operação básica de grafo. Logicamente, cada regra tem uma das três possibilidades de efeito: ela faz um grafo conceitual mais especializado, ela faz um grafo conceitual mais generalizado ou ela muda a forma do grafo conceitual, deixando-o logicamente equivalente ao original. Todas as regras vêm em pares. Para cada regra de especialização, existe uma regra de generalização correspondente; e, para cada regra de equivalência, existe uma outra regra de equivalência equivalente que transforma um grafo conceitual para sua forma original. Essas regras são fundamentalmente gráficas, elas são mais fáceis de mostrar do que de descrever.

As duas primeiras regras (Figura 4.7) são: copiar e simplificar. Abaixo, está um GC para a

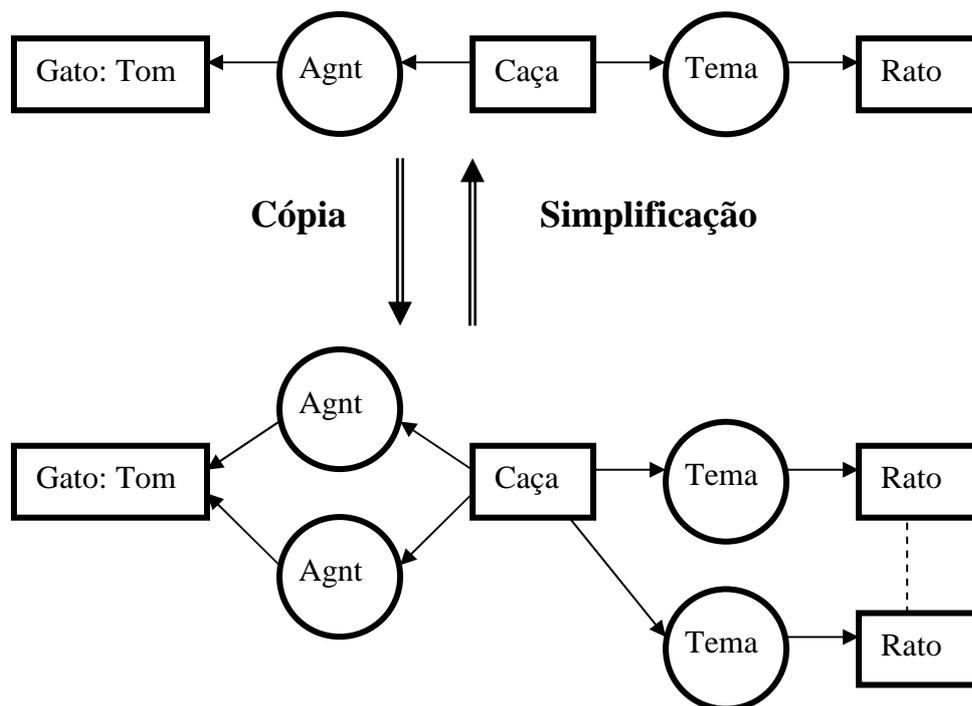


Figura 4.7. Regras de cópia e simplificação.

sentença: "O gato Tom está caçando um rato". A seta para baixo representa a regra de cópia. Uma primeira aplicação da regra copia a relação *Agnt*, e a segunda aplicação, copia o sub-grafo  $\rightarrow$  (*Tema*)  $\rightarrow$  (*Rato*). Ambas as cópias são redundantes porque elas não adicionam nenhuma informação. A seta para cima representa duas aplicações da regra de simplificar que executa a operação inversa de apagar cópias redundantes. As regras de

copiar e simplificar são chamadas de regras de equivalência, porque quaisquer dos dois grafos conceituais podem ser transformados de um para outro por qualquer combinação de regras de cópia e simplificação que serão logicamente equivalentes. As duas fórmulas em cálculo de predicado que são derivadas dos grafos conceituais (Figura 4.7) são logicamente equivalentes. O grafo conceitual (Figura 4.7, cima) mapeia para a seguinte fórmula:

$$\bullet (\exists x: \text{Gato}) (\exists y: \text{Caçador}) (\exists z: \text{Rato}) (\text{nome}(x, \text{'Tom'}) \wedge \text{agnt}(y, x) \wedge (\text{tema}(y, z))),$$

O qual é verdadeiro ou falso sob exatamente às mesmas circunstâncias, como a fórmula que corresponde ao grafo conceitual (Figura 4.7, baixo):

$$\bullet (\exists x: \text{Gato}) (\exists y: \text{Caçador}) (\exists z: \text{Rato}) (\exists w: \text{Rato}) (\text{nome}(x, \text{'Tom'}) \wedge \text{agnt}(y, x) \wedge \text{agnt}(y, x) \wedge (\text{tema}(y, z)) \wedge (\text{tema}(y, w)) \wedge z=w,$$

Pela regra de inferência do cálculo de predicado, todas as duas fórmulas podem ser derivadas uma da outra.

No exemplo das regras de restrição e expansão (Figura 4.8), acima tem um grafo conceitual para a sentença: "Um gato está caçando um animal". Aplicando duas vezes as regras de restrição, transforma-se o grafo conceitual para: "O gato Tom está caçando um rato". O primeiro passo é uma restrição da referência do conceito [Gato] que representa

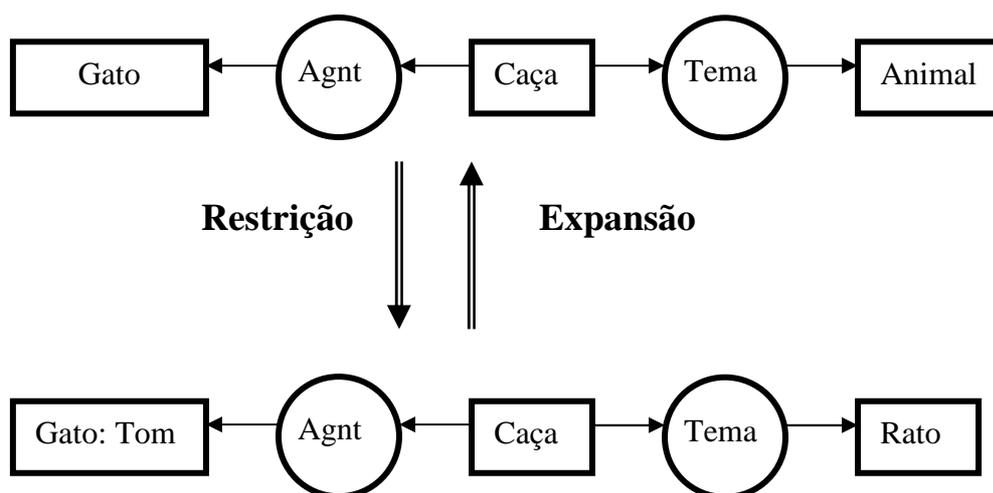


Figura 4.8. Regras de restrição e expansão.

algum gato não identificado, para um conceito mais específico [Gato: Tom] que representa um gato individual, chamado Tom. O segundo passo é a restrição por tipo do conceito [Animal] para o subtipo de conceito [Rato]. Duas aplicações de regras de expansão executam a transformação inversa do grafo de baixo para o grafo de cima. A regra de

restrição é chamada uma regra de especialização; e a regra de expansão, é uma regra de generalização. O grafo mais especializado implica no mais generalizado: se *o gato Tom está caçando um rato*, segue que *um gato está caçando um animal*. A mesma implicação serve para as fórmulas correspondentes em cálculo de predicado. A fórmula mais geral:

- $(\exists x: \text{Gato}) (\exists y: \text{Caçador}) (\exists z: \text{Animal}) \text{agnt}(y,x) \wedge (\text{tema}(y,z))$ ,

É implicada pela fórmula mais especializada:

- $(\exists x: \text{Gato}) (\exists y: \text{Caçador}) (\exists z: \text{Rato}) (\text{nome}(x, \text{'Tom'}) \wedge \text{agnt}(y,x) \wedge (\text{tema}(y,z)))$ ,

No exemplo das regras de junção e desmembramento (Figura 4.9, cima) existem dois

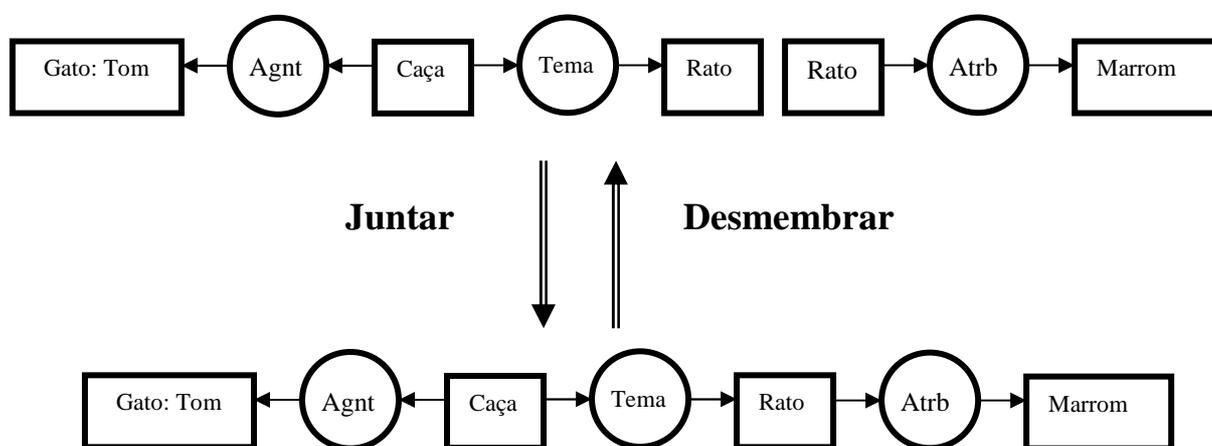


Figura 4.9. Regras de junção e desmembramento.

grafos conceituais para a sentença: "*Tom está caçando um rato*" e "*Um rato é marrom*". A regra de junção sobrepõe as duas cópias idênticas do conceito [Rato] para formar um grafo conceitual simples para a sentença: "*Tom está caçando um rato marrom*". A regra de desmembramento executa a operação inversa. A mesma implicação acontece para as fórmulas correspondentes em cálculo de predicado. A conjunção das fórmulas para os dois grafos conceituais (Figura 4.9, cima):

- $(\exists x: \text{Gato}) (\exists y: \text{Caçador}) (\exists z: \text{Rato}) (\text{nome}(x, \text{'Tom'}) \wedge \text{agnt}(y,x) \wedge (\text{tema}(y,z)) \wedge ((\exists w: \text{Rato})(\exists v: \text{Marrom}) \text{atr}(w,v)))$ ,

É implicado pela fórmula do grafo conceitual (Figura 4.9, baixo):

- $(\exists x: \text{Gato}) (\exists y: \text{Caçador}) (\exists z: \text{Rato}) (\exists v: \text{Marrom}) (\text{nome}(x, \text{'Tom'}) \wedge \text{agnt}(y,x) \wedge (\text{tema}(y,z)) \wedge \text{atr}(w,v))$ ,

Para exemplificar o uso das regras de formação canônica, seguem alguns exemplos baseados nos grafos conceituais que foram exibidos no texto.

Nos exemplos anteriores (Figura 4.4) pode-se restringir *Dar* a *Comunicar* e, em seguida, juntar os grafos obtendo-se (Figura 4.10):

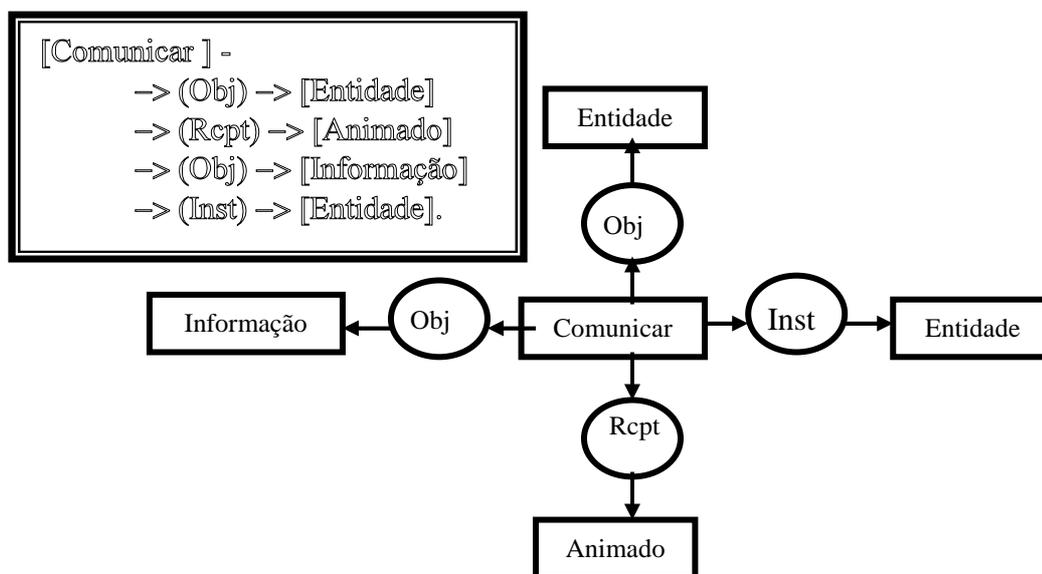


Figura 4.10. Aplicação da regra de restrição.

Visto Entidade ser um supertipo de Informação, faz-se de novo uma restrição seguida de uma junção (Figura 4.11):

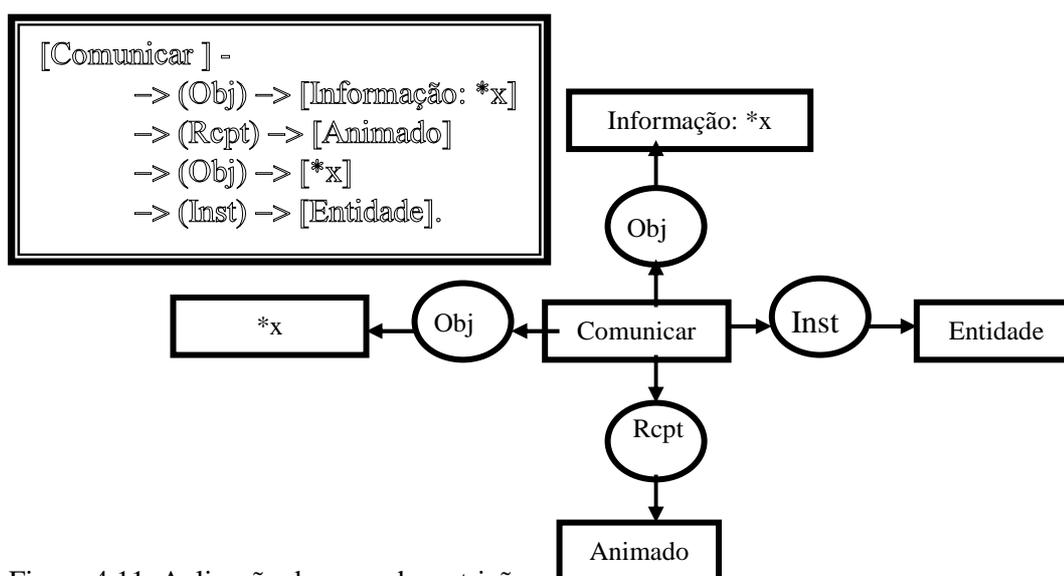


Figura 4.11. Aplicação da regra de restrição.

Como se vê, há duas relações de tipo *Obj* (obtidas da função dos grafos para *Dar* e *Comunicar*) ligadas aos mesmos conceitos. Aplica-se a regra da simplificação (Figura 4.12).

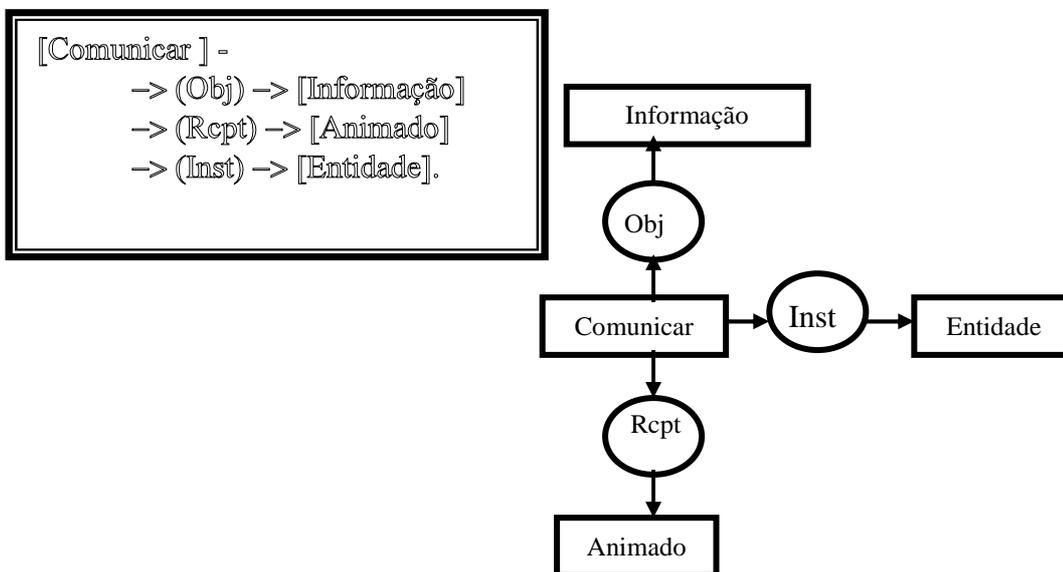


Figura 4.12. Aplicação da regra de simplificação.

Finalmente, restringe-se (Figura 4.4) o tipo *Ato* e junta-se ao grafo anterior. O resultado é o seguinte grafo canônico completo de *Comunicar*; uma especialização comum dos três grafos acima (Figura 4.13).

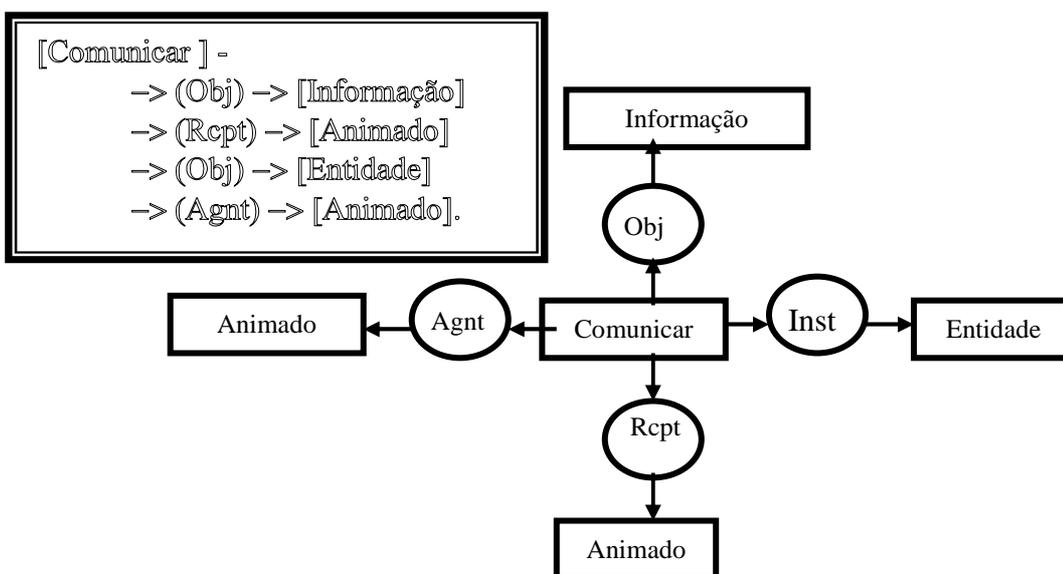


Figura 4.13. Aplicação da regra de junção.

Este exemplo também mostra que juntando os grafos canônicos respectivos, os tipos podem herdar novos "atributos" dos seus supertipos (neste caso *Comunicar* herda o

recipiente de *Dar* e o agente de *Ato*) e restringir os existentes (o objeto passa de *Entidade* à *Informação*). As regras canônicas de formação, além de gerarem novos grafos que não violam as restrições de seleção, implementam um mecanismo de herança entre tipos.

Utilizando apenas as regras canônicas, os grafos gerados sempre fazem sentido, mas podem não ser "verdadeiros" (que expressam somente afirmações verdadeiras) mesmo que as premissas sejam. Por exemplo, assumindo que *Moça* < *Pessoa*, e que:

[Pessoa: Suzana] ← (Agente) ← [Comer] → (Obj) → [Tarde].

[Moça] ← (Agente) ← [Comer] → (Manr) → [Depressa].

São grafos verdadeiros, aplicando as regras canônicas de formação, obtém-se a especialização:

[Pessoa: Suzana] ← (Agente) ← [Comer] -

-> (Manr) -> [Depressa]

-> (Obj) -> [Tarde].

Que pode ser falsa, pois, o fato de existir uma moça que come depressa, não implica que ela seja *Suzana*. Para ter somente grafos verdadeiros, pode-se recorrer as regras de inferência.

## 4.8 Definição de Cálculo de Predicado

Quando um grafo conceitual é representado em CGIF, a regra gramatical permite várias opções diferentes, todas elas são logicamente equivalentes. Como exemplo, considere o seguinte CGIF representado:

[Ir: \*x] (Agente ?x [Pessoa: João]) (Dest ?x [Cidade: Natal]) (Inst ?x [Ônibus])

Como outra opção, a gramática permite que todos os *nós* conceitos fossem movidos para frente com todos os arcos representados por variáveis delimitadas:

[Ir \*x] [Pessoa: João \*y] [Cidade: Natal \*z] [Ônibus \*w]

$$(\text{Agente } ?x \text{ ?y}) (\text{Dest } ?x \text{ ?z}) (\text{Inst } ?x \text{ ?z})$$

Essa representação requer mais espaço, e rótulos de co-referência.

Porém, ela tem um mapeamento mais direto para cálculo de predicado:

$$(\exists x: \text{Ir}) (\exists y: \text{Pessoa}) (\exists z: \text{Cidade}) (\exists w: \hat{\text{Ônibus}})$$

$$(\text{Nome}(y, \text{'João'}) \wedge \text{Nome}(z, \text{'Natal'})) \wedge$$

$$(\text{Agente}(x,y) \wedge \text{Dest}(x,z) \wedge \text{Inst}(x,w))$$

Em CGIF, os nomes *João* e *Natal*, são representados nos campos referentes dos conceitos.

Em cálculo de predicado, o predicado *Nome* é adicionado para indicar que eles são nomes.

Em KIF, porém, a falta de um marcador de questionamento indica que eles são constantes,

e o predicado *Nome*, pode ser omitido. Segue o enunciado correspondente em KIF:

$$(\text{existe } ((?x \text{ Ir}) (?w \hat{\text{Ônibus}})))$$

$$(\text{e } (\text{Pessoa João}) (\text{Cidade Natal}))$$

$$(\text{Agente } ?x \text{ João}) (\text{Dest } ?x \text{ Natal}) (\text{Inst } ?x \text{ ?w}))$$

Para um dado grafo conceitual abstrato, todas as variantes de CGIF permitidas pela gramática são logicamente equivalentes, e elas mapeiam para enunciados em cálculo de predicado ou KIF que são logicamente equivalentes.

A tradução de CGIF para cálculo de predicado é definida pela função  $\phi$ . Se  $u$  é qualquer grafo conceitual, então  $\phi(u)$  é a fórmula correspondente em cálculo de predicado. Como exemplos, segue uma sentença em Português e sua representação em CGIF, cálculo de predicado e KIF:

$$(\text{Sobre } [\text{Gato: @todos}] [\text{Tapete}])$$

$$(\forall: \text{Gato}) (\exists y: \text{Tapete}) \text{ Sobre } (x,y)$$

$$(\text{para\_todo } (?x \text{ Gato}) (\text{existe } (?y \text{ tapete}) (\text{sobre } ?x \text{ ?y})))$$

## 4.9 Definição de Regras de Inferência

As regras de formação canônicas são os fundamentos para todas as operações sobre grafos conceituais. Para cada regra, existe uma regra de inferência correspondente para o cálculo do predicado. Se algumas regras transformam um grafo conceitual  $u$  para um grafo conceitual  $v$ , onde  $u$  implica  $v$ , então a fórmula correspondente  $\phi(u)$  em cálculo de predicado implica a fórmula  $\phi(v)$ . As regras de grafos, porém, são usualmente mais simples do que as regras em cálculo de predicado.

- Regras equivalentes. As regras equivalentes podem mudar a aparência do grafo, mas elas não mudam seu status lógico. Se um grafo conceitual  $u$  é convertido para um grafo conceitual  $v$  por estas regras, então  $u$  implica em  $v$ , e  $v$  implica em  $u$ .
- Regras de especialização: As regras de especialização transformam um grafo conceitual  $u$  em outro grafo conceitual  $v$  que é logicamente mais especializado:  $v$  implica  $u$ .
- Regras de generalização: As regras de generalização transformam um grafo conceitual  $u$  em outro grafo conceitual  $v$  que é logicamente mais generalizado:  $u$  implica  $v$ .

De acordo com os detalhes da sintática dos grafos conceituais, as regras de generalização e especialização permitem que as regras de inferência sejam declaradas de uma forma geral, independentemente da notação do grafo.

- Apagar. Em um contexto positivo, qualquer grafo  $u$  pode ser substituído por uma generalização de  $u$ ; em particular,  $u$  pode ser apagado (substituído por *branco* que é uma generalização de todo GC).
- Inserção. Em um contexto negativo, qualquer grafo  $u$  pode ser substituído por uma especialização de  $u$ ; em particular, qualquer grafo pode ser inserido (ele pode substituir o *branco*).
- Iteração. Se um grafo  $u$  ocorre em um contexto  $C$ , outra cópia de  $u$  pode ser desenhada no mesmo contexto  $C$  ou em outro contexto aninhado em  $C$ .

- Reverter Iteração. Qualquer grafo  $u$  que pode ter sido derivado pela iteração, pode ser apagado.
- Equivalência. Qualquer regra (cópia, simplificação ou dupla negação) pode ser executada em qualquer grafo ou sub-grafo em qualquer contexto.

### 4.10 Categorias Ontológicas

Na lógica, o quantificador existencial  $\exists$  é uma notação para afirmar que alguma coisa existe. A ontologia preenche essa lacuna: é o estudo da existência de todas os tipos de entidade — concreta e abstrata — que constitui o mundo. Os componentes metodológicos das categorias ontológicas são: observação e raciocínio. A observação fornece conhecimento do mundo físico e o raciocínio dá sentido a observação, gerando uma moldura das "abstrações metafísicas".

A árvore (Figura 4.14) é uma síntese da percepção filosófica abrangendo Heráclito até

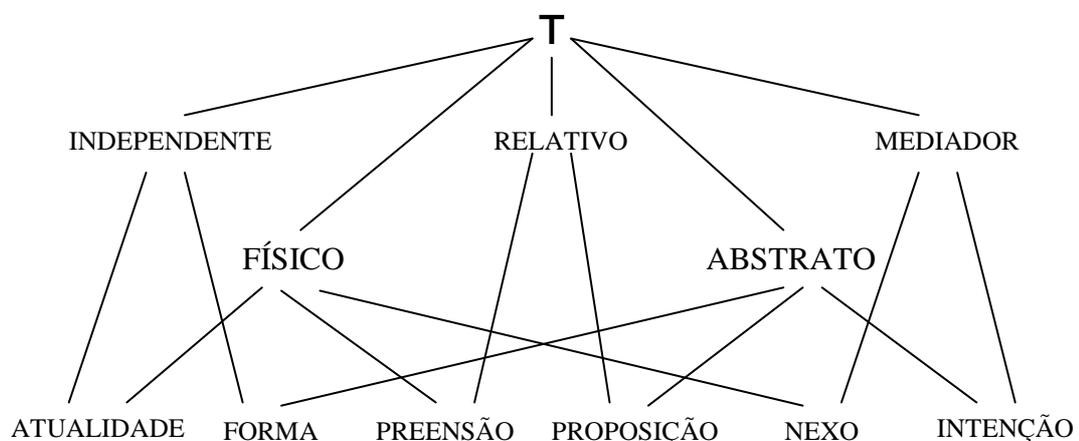


Figura 4.14. Categorias de nível superior de uma ontologia.[SOW.00]

Peirce e Whitehead. O símbolo do topo é  $T$ , que é uma representação neutra do tipo universal. Partindo do princípio que tudo que existe deve ser uma instância de  $T$ , um sinônimo pronunciável para  $T$  é *Entidade*, que vem do Latim *ens* (ser). Abaixo de  $T$ , existe uma separação para a categoria *Física* consistindo da matéria e energia; e a categoria *Abstrata* para estruturas puramente de informação. O terceiro nível divide *Físico* e *Abstrato* em tríades. A "atualidade" corresponde a uma entidade real que pode existir por ela mesma (*Independente*). Uma "preensão" é uma relação direta ou reação entre duas entidades reais (*Relativo*). Um "nexo" é a junção de duas ou mais *preensões*, devendo

incluir no mínimo três entidades (*Mediador*). Uma "*forma*" é uma abstração, como um círculo (*Independente*). Uma "*proposição*" é a representação de um conceito de forma abstrata envolvendo duas entidades, o próprio conceito e sua representação (*Relativo*). Uma "*intenção*" envolve formas subjetivas, como: emoções, propósitos e valores; envolvendo assim, mais de duas entidades (*Mediador*).

### 4.11 Funções Temáticas

As funções temáticas são representadas pelas relações conceituais que ligam o conceito de um verbo ao conceito dos participantes da ocorrência expressada pelo verbo.

"*Participante*" (Figura 4.15) é subdividido classificando-se em dois pares de distinções: "*determinante*" ou "*iminente*" e "*origem*" ou "*produto*".

- Um *participante determinante* controla a direção da atividade, seja do início como precursor, ou do fim como objetivo.
- Um *participante iminente* está presente em um evento, mas não controla ativamente o que acontece.
- Um *origem* deve preservar o início do evento, mas não necessita participar do acontecimento do evento.
- Um *produto* deve ser apresentado no fim do evento, mas não precisa participar do acontecimento do evento.

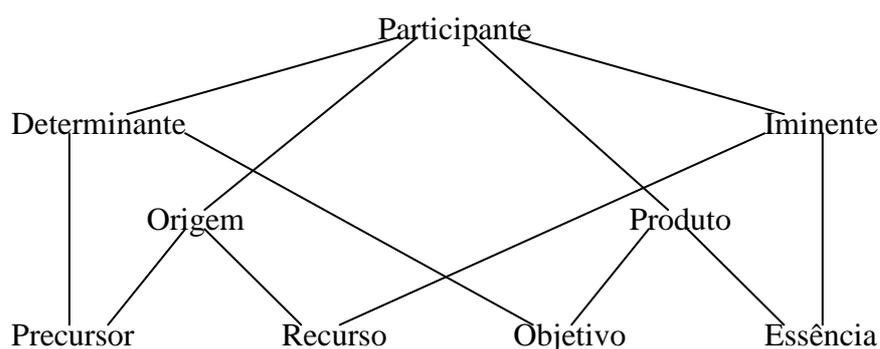


Figura 4.15. Grafo de representação de quatro participantes.[SOW.00]

Como exemplo, considere a sentença: "*Sueli enviou um presente a João pelo correio*". O presente e o correio são *participantes iminentes* porque o presente (*essência*) e o correio (*recurso*) estão presentes do início ao fim. *Sueli* e *João*, porém, são *participantes determinantes* porque eles controlam o curso do processo, do *precursor* (*Sueli*) até o

objetivo (João). Diferente de *participantes iminentes*, os *participantes determinantes* são envolvidos primariamente nos pontos extremos. Se Sueli escrever o endereço errado, o intencionado *receptor*, João, não estará totalmente envolvido.

No caso de ambiguidade, a hierarquia permite que os tipos mais especializados sejam generalizados para qualquer supertipo. Na sentença "*Tony assou a torta*", a torta pode ser um *resultado* que está sendo criado, ou um *paciente* que está sendo esquentado.

[Pessoa: Tom]←(Agnt)←[Assar]→(Rslt)→[Torta: #].

[Pessoa: Tom]←(Agnt)←[Assar]→(Pcnt)→[Torta: #].

Mas de acordo com a matriz (Tabela 4.16), *Resultado*<*Objetivo*<*Produto* e *Paciente*<*Essência*<*Produto*. Como *produto* é um supertipo comum, a interpretação inicial pode ter o seguinte rótulo:

[Pessoa: Tony]←(Agnt)←[Assar]→(Prod)→[Torta: #].

Na sentença "*O cão quebrou a janela*", o cão pode ser o *agente* ou o *instrumento*:

[Cão: #]←(Agnt)←[Quebrar]→(Pcnt)→[Janela: #].

[Cão: #]←(Inst)←[Quebrar]→(Pcnt)→[Janela: #].

Mas *Agente* < *Precursor* < *Origem* e *Instrumento* < *Recurso* < *Origem*. Assim, a interpretação inicial poderia ter o rótulo *Origem* (Orgm). Quando a informação ficar disponível, o tipo geral pode ser restrito a um subtipo mais especializado (Tabela 4.16).

Tabela 4.16. Funções temáticas como subtipos de quatro espécies de participantes.[SOW.00]

	Origem		Produto	
	Precursor	Recurso	Objetivo	Essência
Ação	Agente, Efeito	Instrumento	Resultado, Receptor	Paciente, Tema
Processo	Agente, Origem	Matéria	Resultado, Receptor	Paciente, Tema
Transferência	Agente, Origem	Instrumento, Mídia	Experiente, Receptor	Tema

Espacial	Origem	Caminho	Destino	Local
Temporal	Início	Duração	Término	Ponto no tempo
Ambiente	Origem	Instrumento, Matéria	Resultado	Tema

*Agente* < *Precursor*; *Agnt* (*Ato*, *Animado*). Uma *entidade* animada ativa que voluntariamente inicia uma ação.

Exemplo: *Eva mordeu uma maçã.*

[Pessoa: Eva]←(Agnt)←[Morder]→(Pcnt)→[Maçã].

*Beneficiário*<*Receptor*; *Benf*(*Ato*, *Animado*). Um *receptor* que produz um benefício do sucesso da finalização do evento.

Exemplo: *Os diamantes foram dados à Vanessa.*

[Diamantes: { \* } ]←(Tema)←[Dar]→(Benf)→[Pessoa: Vanessa].

*Caminho*<*Recurso*; *Cmnh*(*Processo*, *Lugar*). Um *recurso* de umnexo espacial.

Exemplo: *A pizza foi enviada via Goiânia e Palmas.*

[Pizza: #]←(Tema)←[Enviar]→(Cmnh)→[Cidade: { Goiânia, Palmas }].

*Começo* < *Precursor*; *Cmço*(*Entidade*, *Tempo*). Um *determinante origem* de umnexo temporal.

*Destino*<*Objetivo*; *Dest*(*ProcessoEspacial*, *Físico*). Um *objetivo* de um processo espacial.

Exemplo: *João foi a Maceió.*

[Pessoa: João]←(Agnt)←[Ir]→(Dest)→[Cidade: Maceió].

*Duração*<*Recurso*; *Dur*(*Estado*, *Intervalo*). Um *recurso* de um processo temporal.

Exemplo: *O caminhão prestou serviço por 5 horas.*

[Caminhão: #]←(Tema)←[Serviço]→(Dur)→[Intervalo: @5hs].

*Efeito*<*Precursor*; *Eft*(*Entidade*, *Entidade*). Um *determinante origem* ativo, animado ou inanimado, que inicia uma ação mas sem intenção voluntária.

*Experiente* < *Objetivo*; *Expr* (*Estado*, *Animado*). Um *objetivo* animado ativo de um experiente.

Exemplo: *Tom vê o peixe.*

[Gato: Tom]←(Expr)←[Ver]→(Tema)→{Peixe: #}.

*Início*<Precursor; *Orgm*(Processo, Físico). Um *determinante origem* pacífico de um espacial ou ambiente nexos.

Exemplo: *O capítulo começa na página 20.*

[Capítulo: #]←(Tema)←[Começa]→(Orgm)→[Página: 20].

*Instrumento*<Recurso; *Inst*(Ato, Entidade). Um *recurso* que não é mudado por um evento.

Exemplo: *A chave abriu a porta.*

[Chave: #]←(Inst)←[Abrir]→(Tema)→[Porta: #].

*Local*<Essência; *Loc*(Físico, Físico). Um *participante* essencial de um nexos espacial.

Exemplo: *Os veículos chegaram a estação.*

[Veículo: { \* }]←(Tema)←[Chegar]→(Loc)→[Estação].

*Matéria*<Recurso; *Matr*(Ato, Substância). Um *recurso* que é trocado pelo evento.

Exemplo: *A arma foi entalhada no sabão.*

[Arma]←(Rslt)←[Entalhar]→(Matr)→[Sabão].

*Mídia*<Recurso. *Med*(Processo, Físico). Um *recurso* para transmissão de informação, como o som da fala, ou sinais eletromagnéticos que transmitem dados.

*Paciente*<Essência; *Pcnt*(Processo, Físico). Um *participante* essencial que passa por uma mudança estrutural como resultado do evento.

Exemplo: *O gato engoliu o canário.*

[Gato: #]←(Agnt)←[Engolir]→(Pcnt)→[Canário: #].

*PontoNoTempo*<Essência; *PTmp*(Físico, Tempo). Um *participante* essencial do nexos temporal.

Exemplo: *Às 5:25, Érica embarcou.*

[Tempo: 5:25]←(Ptmp)←[Proposição: [Pessoa: Érica]←(Agnt)← [Embarque]].

*Receptor*<Objetivo; *Rcpt*(Ato, Animado). Um *objetivo* animado de um ato.

*Resultado*<Objetivo; *Rslt*(Processo, Entidade). Um *objetivo* inanimado de um ato.

Exemplo: *Gabriel construiu uma casa.*

[Pessoa: Gabriel]←(Agnt)←[Construir]→(Rslt)→[Casa].

*Tema*<Essência; Tema(Situação, Entidade). Um participante essencial que pode ser movido, dito, ou experimentado.

Exemplo: *João gosta de cerveja.*

[Pessoa: João]←(Expr)←[Gostar]→(Tema)→[Cerveja: #].

*Término*<Objetivo; Trmn(ProcessoTemporal, Físico). Um objetivo de um processo temporal.

### 4.12 Localização das Funções Temáticas

Como subtipos de *participante*, as funções temáticas ocupam um nível intermediário na ontologia (Figura 4.17). Cada uma das funções temáticas poderiam ser organizadas abaixo

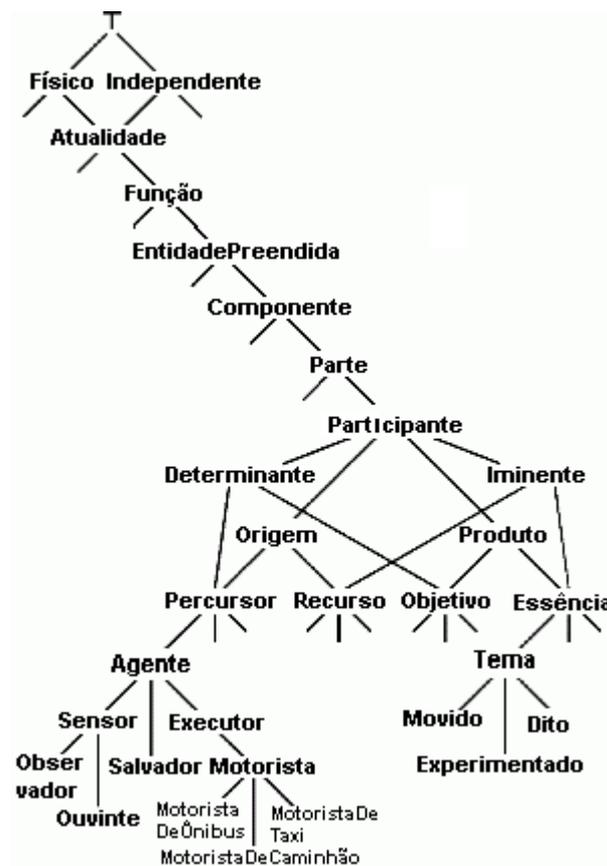


Figura 4.17. Localização das funções temáticas na ontologia.[SOW.00]

das quatro categorias de *participantes*: *Percursor*, *Recurso*, *Objetivo*, e *Essência*. Na parte inferior (Figura 4.17) temos ramos abaixo de *Agente* e *Tema*. *Executor*, por exemplo, tem

um subtipo *Motorista* que tem mais subtipos específicos, como: *MotoristaDeOnibus*, *MotoristaDeCaminhão*, e *MotoristaDeTaxi*. Em princípio, qualquer função temática poderia ser subdividida, para mostrar distinções que podem ser significativas em alguma cultura ou domínio de interesse.

Apesar das funções temáticas representarem uma importante classe lingüística das categorias lógicas da ontologia, seu supertipo comum *Participante* está vários níveis abaixo da categoria mais geral *Função*. Assim, *Função* poderia incluir muitos tipos que não são diretamente associados com os verbos. Como exemplo, a função *Motorista* representa somente a pessoa que esta dirigindo um veículo. Essa função poderia ser incompatível com a função *Pedestre*. A categoria *MotoristaLicenciado*, porém, inclui pessoas que são legalmente autorizadas a dirigir, não importando se elas estão dirigindo ou não no momento. Na cidade de São Paulo, motoristas licenciados, provavelmente, gastam mais tempo como pedestres do que realmente dirigindo. Como outro exemplo, uma pessoa pode ter um período contínuo de emprego como chofer, mas poderia não estar ativamente dirigindo, continuamente. Assim, o tipo *Chofer* poderia ser um subtipo de *Empregado* e *MotoristaLicenciado*, mas não um subtipo de *Motorista*.

### 4.13 Ponto de Vista

As distinções entre *Físico* e *Abstrato* são independentes do ponto de vista do observador, mas a distinção entre *Contínuo* e *Ocorrência* depende da escolha da escala de tempo. Numa escala de minutos, uma geleira é um contínuo e uma avalanche é uma ocorrência. Mas em uma escala de séculos, a geleira é um processo cuja característica pode ser transformada além do reconhecimento. Um ser humano que tem uma identidade estável em um nível macro, pode ser visto como um processo de mudança contínua no nível molecular, já que em aproximadamente sete anos a maioria das moléculas do corpo humana é trocada.

Peirce e Whitehead notaram que a maneira que uma entidade física é classificada depende da intenção ou forma subjetivas da percepção de algum agente. Uma proposição, por ela mesma, caracteriza alguma entidade pela forma abstrata. O estado mental de uma pessoa que acredita ou expressa uma proposição, não está envolvida na proposição. Uma intenção, porém, é a mediação mental que dirige a atenção de um agente para alguma forma que caracteriza alguma entidade.

John Searle, em *Intentionality* [SEA.83], (Figura 4.18) apresentou agentes com diferentes intenções interpretando intencionalidade de maneiras diferentes. Ela pode ser vista como a palavra "TOOT", como uma mesa com dois balões debaixo, como o numeral 1001 com uma linha em cima, como uma ponte com dois tubos cruzando debaixo ou como os olhos de um homem usando um chapéu com um fio caindo de cada lado. Em cada caso, temos uma experiência diferente, embora as linhas no papel fiquem em frente ao agente, e a luz refletida dela seja constante. Mas essas experiências e as diferenças entre elas dependem de nossas habilidades culturais e lingüísticas. Um cão não veria essa figura como a palavra "TOOT".

Cada interpretação poderia ser declarada como uma proposição diferente: "*Existe uma palavra TOOT*", ou "*Existe uma mesa com balões debaixo*". Nessas sentenças, a palavra existir indica a entidade física, e a frase seguida da palavra existe especifica alguma forma que é adotada para a entidade. Mas a proposição não faz nenhuma referência a qualquer agente, explicitamente ou implicitamente, que acredita nela. Uma crença é intencional porque ela envolve o agente que relaciona a forma à entidade. *Maria* acredita (Figura 4.18) que seja a palavra TOOT; mas *João* acredita que ela mostra uma mesa com dois balões debaixo. As intenções de *Maria* e *João* são essenciais para suas crenças, mas não para a proposição que constitui o conteúdo dessas crenças.

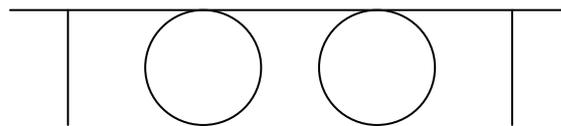


Figura 4.18. Exemplo de Searle para uma figura ambígua.

Diferentes intenções conduzem a diferentes tipos de conceitos para classificar a mesma entidade. Um tipo estrutural descreve uma entidade pela sua estrutura, ou forma natural, independentemente de suas relações com entidades externas. A frase "*um cubo de madeira*", por exemplo, descreve uma coisa de duas maneiras. O substantivo cubo nomeia uma forma geométrica que descreve sua forma. O adjetivo "*de madeira*" descreve uma personificação desta forma como um objeto feito de madeira. Sem este adjetivo, a palavra cubo, poderia referir: a uma forma abstrata ou a um objeto físico. Formalmente, um tipo estrutural classifica uma entidade  $x$  pelo predicado monada, que depende somente das

propriedades que podem ser observadas em  $x$ . Para esse exemplo, a frase "cubo de madeira" poderia ser traduzida para uma conjunção de dois predicados fenomenais,  $madeira(x) \wedge cubo(x)$ .

Um tipo função caracteriza uma entidade por alguma função que ela representa em relação à outra entidade. O tipo *SerHumano*, por exemplo, é um tipo fenomenal que depende da forma interna de uma entidade. Mas a mesma entidade poderia ser caracterizada pelos tipos função: *Mãe*, *Empregado*, ou *Pedestre*. Os tipos de função podem ser aplicados para coisas de aparências radicalmente diferentes. Entidades de tipos fenomenais: *Batata* ou *Carne*; poderiam servir na função de *Comida* para algum ser humano. O tipo fenomenal *Cavalo*, *Bicicleta* e *Avião* descrevem entidades que poderiam representar a função de *ModeloDeTransporte*. Um *SerHumano* e *OrganizaçãoNegócios* descrevem entidades que poderiam representar a função de *PessoaLegal*. Formalmente, tipos de função dependem de uma relação díade: se  $x$  é classificado por um tipo de função, então  $x$  tem uma relação díade com outra entidade  $y$ . Em lógica, ambas entidades numa relação díade podem ser expressas pelos argumentos de um predicado, como:  $comidaPara(x, y)$ .

Sendo tipos fenomenais monadas e tipos de função díades, Peirce sugeriu que haveria uma terceira espécie de categoria, baseada numa relação tríade. Peirce observou que qualquer entidade física pode também atender como um símbolo que dependeria de uma relação tríade de representação: um símbolo  $x$  representa alguma coisa  $y$  para algum agente  $z$ .

### 4.14 Tipos e Conjuntos

Tipos de conceitos são especificações abstratas, não são conjuntos de coisas. Para cada tipo  $t$  existe um conjunto  $\delta t$ , chamado denotação de  $t$  (Figura 4.19). Tartaruga é um tipo no

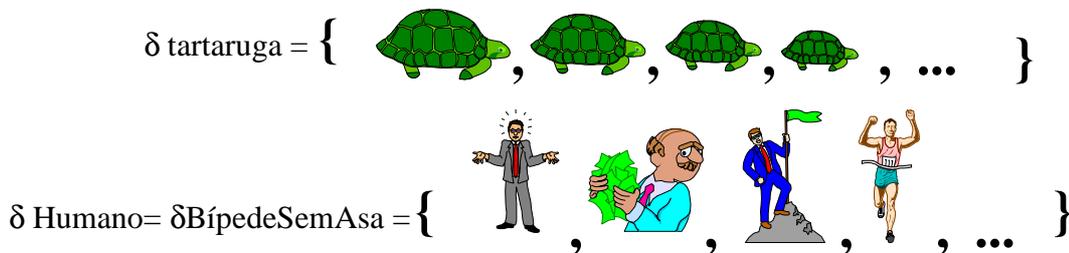


Figura 4.19. Tipos e denotações.

sistema conceitual e  $\delta \text{Tartaruga}$  é o conjunto de todas as tartarugas no mundo. Mas  $\delta$  não é mapeado um para um do reticulado de tipos para o reticulado de conjuntos. Existem muito

mais conjuntos de coisas no mundo do que as pessoas estão cientes, e muitos tipos de coisas que as pessoas acham que não existem. Alguns tipos, como o *Saci-Pererê*, mapea para um conjunto vazio. Outros tipos, como *Humano* e *BípedeSemAsa* mapea para o mesmo conjunto; e um conjunto selecionado de coisas ao acaso poderia não mapear tipo de conceito algum.

Falando sobre o sentido, filósofos descreveram uma distinção entre a intenção<sup>1</sup> de um termo (seu sentido intrínseco ou conceito associado) e suas: extensão ou denotação. Os tipos *Humano* e *BípedeSemAsa* tem diferentes significados ou intenções. Frege [FRE.97] usou o exemplo da estrela da noite e da estrela da manhã. Esses dois termos têm diferentes intenções: uma significa uma estrela que é vista pela manhã e a outra significa uma estrela que é vista a noite. Apesar disso, ambos tem a mesma denotação ou extensão, chamado planeta Vênus.

Uma base semântica poderia ser por extensão ou intencional. Uma definição por extensão do tipo *Vaca*, por exemplo, poderia ser um catálogo de todas as vacas do mundo. Uma definição intencional poderia especificar as propriedades ou critérios para reconhecimento de vacas, sem levar em consideração a sua existência. Como o número de vacas no planeta é grande, e muda constantemente, uma definição por extensão poderia ser impraticável. Assim, o tipo *Vaca* deve ser definido por intenção. Antes de Galileo, os tipos *Sol* e *Lua* só tinham uma instância, poderia ser definida pela extensão. Mas novas descobertas levaram os astrônomos a generalizar esses conceitos e sua extensão por todo o universo é desconhecida.

Em um tipo hierárquico, a posição do tipo de um conceito é mais bem determinada pela intenção do que pela extensão. O Tipo *SaciPererê*, por exemplo, não tem instâncias porque não existem saci-pererês. Assim, sua extensão é vazia. Como o conjunto vazio é um subconjunto de todos os outros conjuntos, a extensão do *SaciPererê* é um subconjunto de todos os outros conjuntos, incluindo as extensões de *Vaca*, *Árvore*, e *Asteróide*. A descrição de um saci-pererê, porém, o define com um ser humano com uma perna só. Pela intenção, o tipo *SaciPererê* poderia ser colocado abaixo de *SerHumano*, mas não abaixo de

---

<sup>1</sup> A palavra intenção, neste caso, não se refere ao tipo "intenção", mas de forma mais generalizada, como: pensamento, sentido.

*Vaca*, *Árvore*, ou *Asteróide*. Para o tipo mais geral, a denotação  $\delta\top$  é tudo o que existe. Para o tipo absurdo, a denotação  $\delta\perp$  é o conjunto vazio  $\{\}$ .

## 4.15 Conclusão

Toda a indexação do conhecimento está fundamentada na ontologia. A definição de uma ontologia é o primeiro passo para a construção de uma base de conhecimento. Uma hierarquia de categorias ontológicas mal formulada limita a funcionalidade dos programas e bases de dados que a utilizam. Qualquer planejamento ontológico, por menor que seja o domínio de representação, exige um conhecimento profundo do projetista para a representação da estrutura hierárquica de heranças. Os axiomas devem definir apenas as propriedades essenciais que caracterizam as categorias ontológicas. Qualquer definição axiomática ambígua pode aprovar a geração de grafos conceituais sem sentido. A definição de grafos canônicos incompreensíveis pode admitir a confecção de grafos conceituais enigmáticos. A estrutura de indexação evita a concepção de grafos conceituais inexplicáveis e proporciona a implementação de máquinas de inferência. Para o bom funcionamento da arquitetura é imprescindível que as definições de ontologias, axiomas, grafos canônicos e relações de conformidade sejam elaboradas por usuários experientes.

---

## 5. Arquitetura do Ambiente VEDA

### 5.1 Introdução

A arquitetura VEDA foi concebida utilizando indexação fundamentada em grafos conceituais. A abordagem sugere a implementação de ferramentas de software para edição, busca, navegação e gerenciamento de documentos.

O conceito Web viabiliza a concepção de comunidades virtuais para aprendizagem colaborativa. Vários sistemas para educação a distância têm sido disponibilizados, mas apenas alguns deles escapam do paradigma behaviorista clássico, onde o aprendiz é considerado um mero receptor da informação e não um construtor do conhecimento [JON.95]. A arquitetura surge com a finalidade de mudança do paradigma educacional da "mudança de página" e avaliação por múltipla escolha, nascendo de dois princípios da escola social-construtivista [GIL.97]: o conhecimento é disponibilizado para o aprendiz de forma visual; e o aprendiz pode co-elaborar seu conhecimento através da reflexão da comunidade Web virtual.

A arquitetura é aberta, podendo ser utilizada em várias áreas de ensino/aprendizagem. Isto ocorre porque o ambiente permite ao administrador do conhecimento, a criação de seus próprios temas (ontologias). O conhecimento é representado em páginas WWW, confeccionadas em formato Hyper-Text Mark-up Language (HTML). Essa propriedade permite o uso de navegadores Web que estão disponíveis para uma infinidade de plataformas de computadores.

O ambiente VEDA proporciona as seguintes propriedades gerais:

- ❑ Ambiente visual do conhecimento;
- ❑ Ambiente colaborativo de ensino/aprendizagem;
- ❑ Engloba várias áreas de ensino/aprendizagem;
- ❑ Disponível para Internet.

## 5.2 Arquitetura VEDA

Os grafos conceituais são representados em páginas Web de forma gráfica (display form). Isto é feito através do uso de applets Java. Assim, os grafos conceituais podem ser embutidos dentro de qualquer página HTML. As coordenadas dos diagramas são representadas dentro das páginas e passadas como parâmetros para as applet Java.

A construção dos grafos conceituais da arquitetura obstrui a criação de grafos conceituais sem sentido. Isto é propiciado devido às inferências realizadas pela esfera de domínio da Inteligência Artificial do ambiente VEDA. Podemos citar a hierarquia de tipos de conceitos e relações da ontologia juntamente com as regras de formação canônica; os grafos canônicos; as relações de conformidade dos tipos de conceitos; e, a inserção de axiomas nos tipos de conceitos e relações.

A edição dos documentos é feita de forma colaborativa. Os usuários que estiverem participando da criação do mesmo documento podem visualizar, ao mesmo tempo, o que os outros estão fazendo, identificando-os. O ambiente é totalmente on-line. A edição também permite conversas simultâneas de forma textual. Assim, os participantes podem se interagir em discussões sobre o trabalho.

Existe a possibilidade de divisão dos repositórios em bases de conhecimento. As comunidades podem armazenar seus documentos de forma segura através da requisição de senhas de acesso.

O ambiente VEDA é dividido em áreas de trabalho para Usuário Geral e Administração do Conhecimento. Dessa forma, alguns módulos do ambiente que requerem maior habilidade na utilização, só podem ser acessados pelos Administradores do Conhecimento. Modificações no modelo de ontologia, grafos canônicos e axiomas podem afetar o ambiente como um todo. Os administradores do conhecimento que fazem manutenção nos modelos de ontologia devem conhecer profundamente sobre o assunto.

O ambiente de navegação permite ao usuário a criação de ponteiros (links) de acordo com seus próprios critérios, mesmo que a autoria já tenha criado um ponteiro para aquele conceito ou relação. Por exemplo, o usuário pode criar ponteiros para páginas Web de outros autores. A navegação também proporciona opção de pontos de vista ontológicos e

bases de conhecimento diferentes. Além disso, o usuário pode generalizar ou especificar os conceitos e relações de acordo com seu interesse. Os conceitos e relações que ainda não tiverem ponteiros podem ser criados pelo usuário da navegação, de forma persistente, desde que tenha permissão para isso. Essa é outra característica de construção do conhecimento em colaboração.

A criação de ponteiros virtuais (definidas pelo usuário de navegação) pode gerar ponteiros para várias páginas Web. Neste caso, a navegação se torna uma pesquisa em que o usuário faz a escolha de um dos ponteiro disponibilizadas para visualizar a página Web desejada.

A arquitetura disponibiliza a criação de várias ontologias. Isto possibilita a contextualização de ontologias para domínios de interesse de um grupo específico de usuários. Podem existir vários ambientes particulares dentro do mesmo ambiente VEDA. Os administradores do conhecimento podem adequar as ontologias de acordo com cada tema de ensino.

O ambiente VEDA permite a inclusão das referências criadas pelos usuários gerais para os tipos de conceitos na ontologia em foco. Isto permite uma representação das instâncias dentro da ontologia. Relacionando com linguagens orientadas a objeto, os tipos de conceitos são como classes, e os tipos de conceitos com referência como os objetos do ambiente. Dessa forma, a ontologia representa não só a estrutura hierárquica de classes do domínio, mas também os objetos que fazem parte do contexto.

A edição de grafos conceituais restringe a autoria do usuário geral aos tipos de conceitos e relações existentes na ontologia em foco. Isto evita a idiosincrasia ou mesmo a perda do sentido no desenvolvimento dos grafos conceituais. A criação do conhecimento se torna mais rápida, além da possibilidade de fazer comparações com diagramas de outros autores.

As ontologias permitem a inserção de axiomas que evitam a criação de grafos sem sentido. Os axiomas são designados aos tipos de conceito e relações da ontologia, e são herdados pelos seus subtipos.

A recuperação dos documentos é realizada pela pesquisa textual dentro dos repositórios que estão em formato HTML. Através do uso de inferências, o ambiente é capaz de buscar o conhecimento de forma: precisa, generalizada ou especificada. A recuperação do

conhecimento, satisfazendo a Inteligência Artificial do ambiente, deve especificar uma ontologia como referência.

A possibilidade de divisão em bases de conhecimento permite a recuperação dos grafos conceituais em todo o repositório de documentos disponível<sup>1</sup> para o usuário, ou em bases de conhecimento particulares.

A recuperação das páginas Web pode ser feita através da informação dos conceitos, relações ou grafos conceituais que fazem parte dela, de forma generalizada ou especificada. Além disso, os grafos conceituais possuem ponteiros para as conversas textuais que foram feitas, quando da criação do documento.

A arquitetura é dividida em módulos. Cada módulo desempenha uma função específica no ambiente. Os módulos são agrupados em funções maiores para melhor organização de sua representação. As linhas que ligam os módulos, representam a passagem de um módulo para outro, independentemente do fluxo de dados (Figura 5.1).

### 5.3 Interfaces

As Interfaces fazem a conexão dos usuários gerais e administradores do conhecimento com o gerenciamento do ambiente. Esse conjunto de telas proporciona uma interação do ambiente com os usuários, permitindo a entrada e saída de dados do ambiente. As interfaces são divididas em usuários gerais e administração do conhecimento. Todas as interfaces do ambiente são disponibilizadas como páginas Web.

### 5.4 Interfaces do Usuário Geral

As interfaces do usuário geral são compostas de módulos que interagem com os usuários do ambiente em geral. Incluem funções como edição, recuperação do conhecimento e navegação em páginas Web. Os usuários gerais precisam de um entendimento superficial sobre grafos conceituais para utilização das interfaces. Entre os requisitos, incluímos:

- ❑ Definições de conceitos, relações, e grafos conceituais;
- ❑ Noções gerais sobre ontologias;
- ❑ Noções gerais sobre grafos canônicos, e regras de formação canônica.

---

<sup>1</sup> A disponibilidade referenciada é em relação aos documentos que não possuem senhas de acesso, ou que o usuário tenha permissão de acesso.

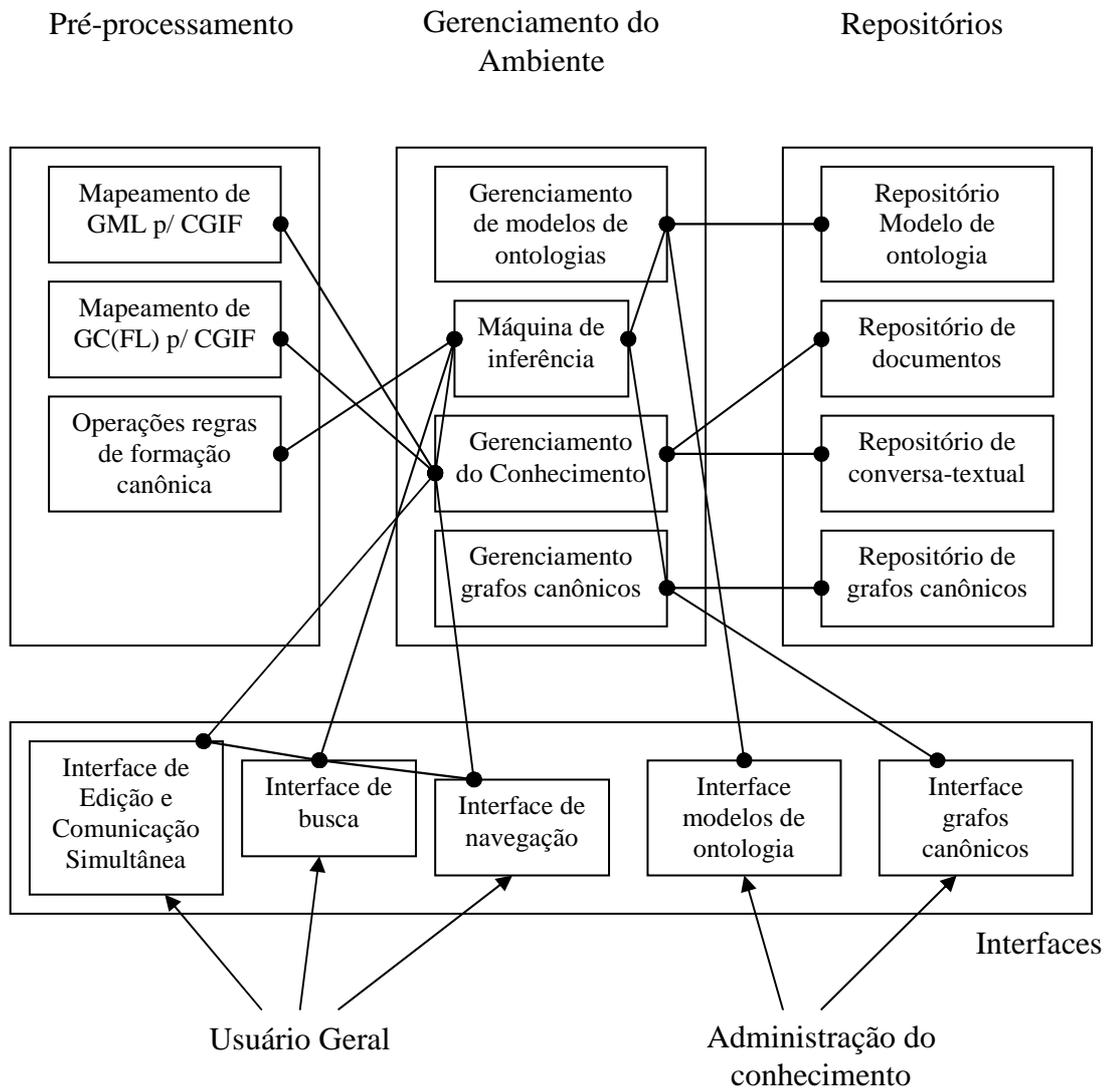


Figura 5.1. Arquitetura de um ambiente visual de representação do conhecimento.

### 5.4.1 Interface de Edição e Comunicação Simultânea

Esse módulo fornece uma interface de comunicação simultânea entre os usuários "on-line" que estiverem editando o mesmo grafo conceitual (Figura 5.2). A interface permite abrir, salvar e criar novos grafos conceituais. A conversa textual permite a identificação do autor das sentenças, e no desenho dos diagramas identifica-se, momentaneamente, o autor da figura.

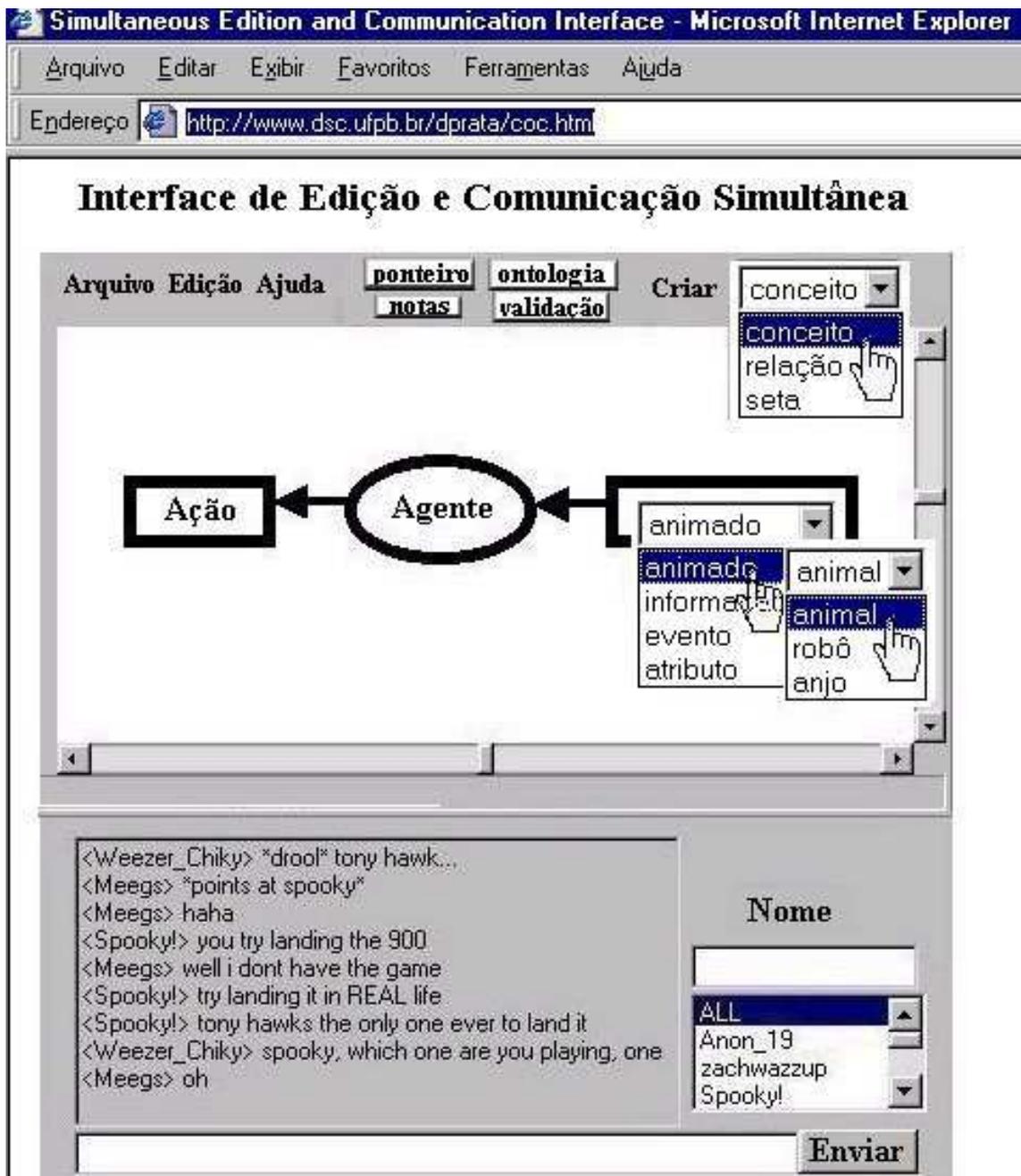


Figura 5.2. Interface de Edição e Comunicação Simultânea.

Os usuários constroem os grafos conceituais criando:

- ❑ Conceitos (representados por caixas);
- ❑ Relações (representadas por elipses);
- ❑ Setas (representadas por arcos direcionados);
- ❑ Ponteiros (links).

Para a criação de conceitos e relações:

- ❑ O usuário escolhe uma das ontologias criadas previamente pelo Administrador do Conhecimento disponíveis na opção ontologia (Figura 5.2);
- ❑ Existe uma ontologia padrão que o ambiente assume caso o usuário não faça a escolha da ontologia;
- ❑ A Interface de Edição e Comunicação Simultânea não proporciona ao usuário geral condições para a criação de novos tipos de conceitos e relações<sup>2</sup>;
- ❑ Os usuários podem criar referências para os tipos de conceitos existentes, de acordo com as relações de conformidade;
- ❑ As referências dos tipos de conceitos que forem criadas pelos usuários gerais serão automaticamente criadas na ontologia utilizada.

Para a criação de ponteiros:

- ❑ O protocolo utilizado é o http;
- ❑ Os ponteiros são referenciados nos conceitos e relações;
- ❑ O ponteiro endereça outra página Web, como exemplo: <http://www.ufpb.br/>.

Para evitar grafos conceituais sem sentido, é necessário a:

- ❑ Verificação dos grafos canônicos referentes à ontologia utilizada;
- ❑ Verificação das relações de conformidade das referências dos tipos de conceitos;
- ❑ Verificação dos axiomas dos tipos de conceitos e relações da ontologia utilizada.

Os grafos conceituais são gravados em páginas HTML, seguindo um formato predefinido. Esse formato atende as necessidades de representação gráfica dos diagramas e facilita a busca que pode ser feita textualmente nas páginas Web. A notação básica dos grafos conceituais é representada em CGIF. A representação de um grafo conceitual (Figura 5.3) segue uma notação pré-definida (Figura 5.4).

As informações dos documentos são armazenadas como parâmetros para a applet Java. A notação CGIF estrutura as informações em uma seqüência de conceitos, relações e setas. As coordenadas de cada elemento do diagrama são definidas logo após a definição de cada elemento do grafo conceitual.

---

<sup>2</sup> Os tipos de conceitos e relações devem ser criados previamente pelo Administrador do Conhecimento.

O primeiro número da coordenada define uma numeração seqüencial dos elementos que estão no grafo. A separação entre a seqüência é feita utilizando-se o símbolo "|". O segundo e terceiro números definem a localização do vértice no quadro. Para que isso

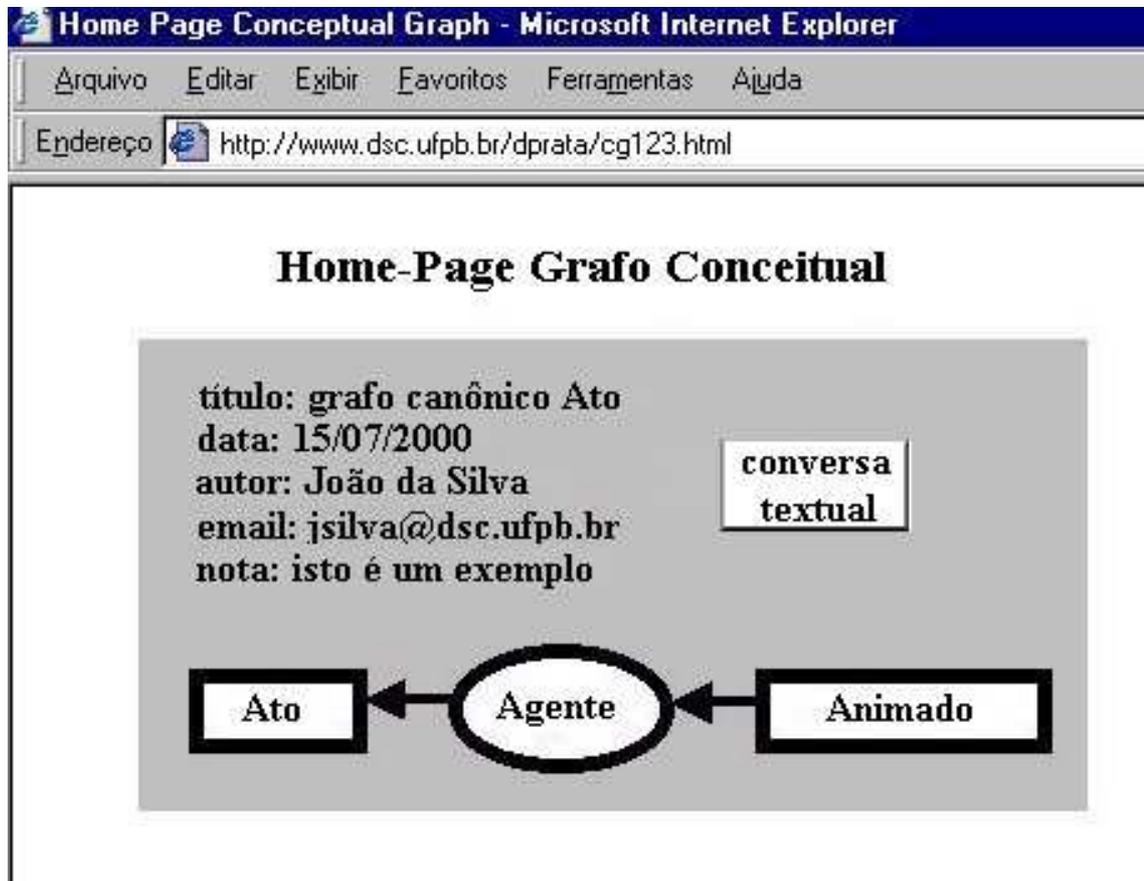


Figura 5.3. Grafo conceitual.

aconteça, o quadro deve ser mapeado em termos de linhas e colunas. O quarto e quinto números definem a altura e o comprimento do elemento do diagrama. No caso das setas existem o sexto e o sétimo números, que definem os elementos do diagrama que são ligados pela seta, e a sua direção.

Para conceitos e relações podem existir ponteiros que carregam outras páginas Web. Esses ponteiros são referenciados logo após as coordenadas do elemento do diagrama. A ontologia também é referência através de um ponteiro Web.

A Interface de Edição e Comunicação Simultânea oferece uma área de conversa textual simultânea, para discussão dos usuários sobre o grafo conceitual. Todas as conversas textuais são armazenadas no Repositório de Conversa Textual de forma persistente. As

páginas Web dos grafos conceituais possuem um ponteiro para as páginas das conversas textuais (Figura 5.3).

```
<html>
<applet codebase="http://www.mgc.ufal.br/applet7932/"
code="mgc.class" width=800 height=800>
<param name="Título" value="Grafo Canônico ATO">
<param name="Autor" value="João da Silva">
<param name="Email" value="jsilva@mail.com.br">
<param name="Data" value="07/06/2000">
<param name="Obs" value="Este é um exemplo">
<param name="ConvTextRef"
value="http://www.mgc.ufal.br/repositorio/ctref3849.html">
<param name="Ontologia" value="http://www.mgc.ufal/ontologia/ontologia.html">
<param name="conceito" value="[Ato:*x1] 01|010|010|020|020">
<param name="conceito" value="[Animado:*x2] 02|100|010|020|020
mref=http://www.mgc.ufal.br/documentos/gc20125.html">
<param name="relação" value="(agente: ?x1 ?x2) 03|050|010|020|020">
<param name="seta" value="04|100|020|030|000|02|03">
<param name="seta" value="05|050|020|030|000|03|01">
</applet>
</html>
```

Figura 5.4. Notação para representação dos gráficos conceituais em Web.

## 5.4.2 Interface de Busca

Esse módulo recebe a entrada de dados para uma pesquisa e retorna as páginas Web correspondentes (Figura 5.5). As entradas de dados podem ser conceitos, relações, grafos conceituais (forma linear) ou data.

Para a execução de uma busca, o usuário faz a escolha de:

- ❑ Ontologia;
- ❑ Base de conhecimento;
- ❑ Igualdade, generalização ou especificação.

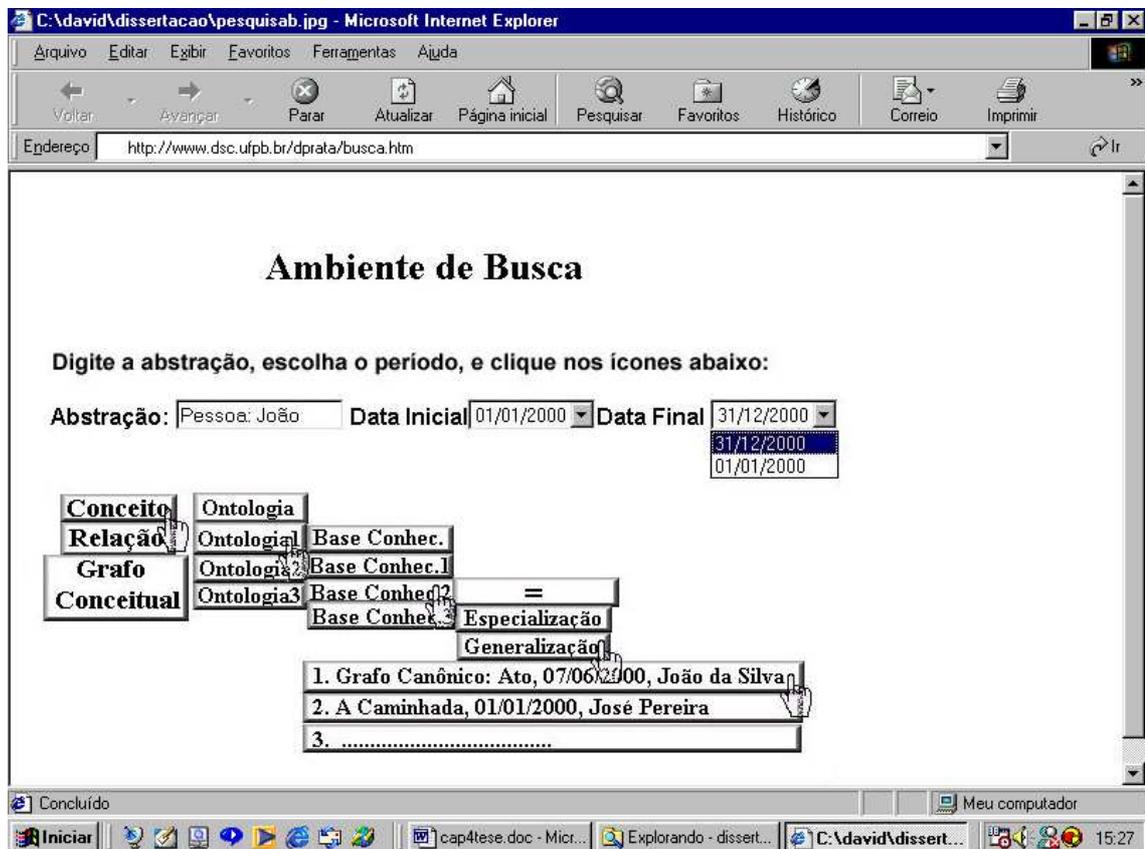


Figura 5.5. Execução de uma pesquisa no Ambiente de Busca.

O módulo Máquina de Inferência provê a funcionalidade para resposta a busca através da unificação da entrada de dados relativa à grade ontológica escolhida com o módulo Repositório de Documentos. A unificação, neste caso, pode envolver operações com regras de formação canônica.

### 5.4.3 Interface de Navegação

Permite navegar nas bases de conhecimento que estejam disponíveis para o ambiente (Figura 5.6). A navegação pode ser feita através dos ponteiros criados pelo autor. Mas o usuário também pode criar seu próprio ponteiro, independentemente do ponteiro do autor. Isto pode ser feito em qualquer conceito ou relação que possua ou não um ponteiro. Se o conceito ou relação não possuir um ponteiro, o usuário pode criar um ponteiro persistente na página como se fosse o próprio autor, em conformidade com as permissões de gravação da página (Figura 5.7). É uma característica do ambiente colaborativo da arquitetura.

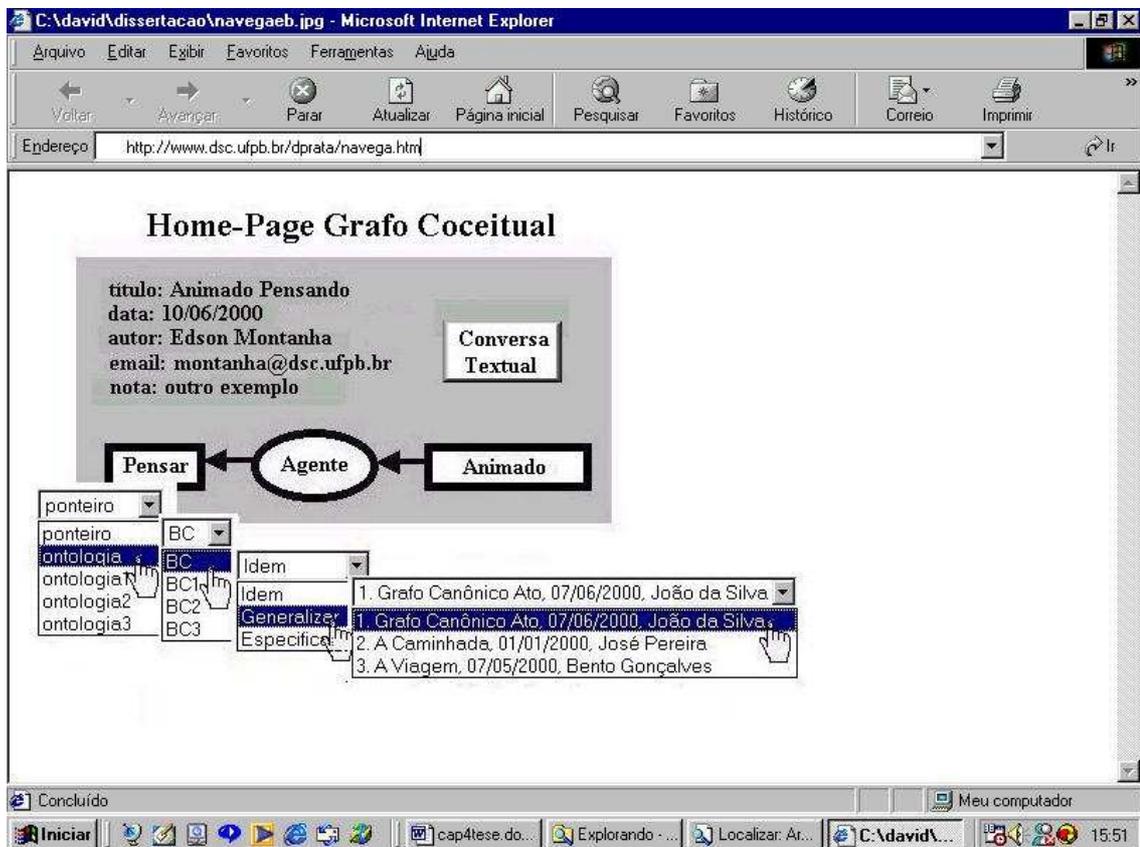


Figura 5.6. Navegação na Interface de Navegação.

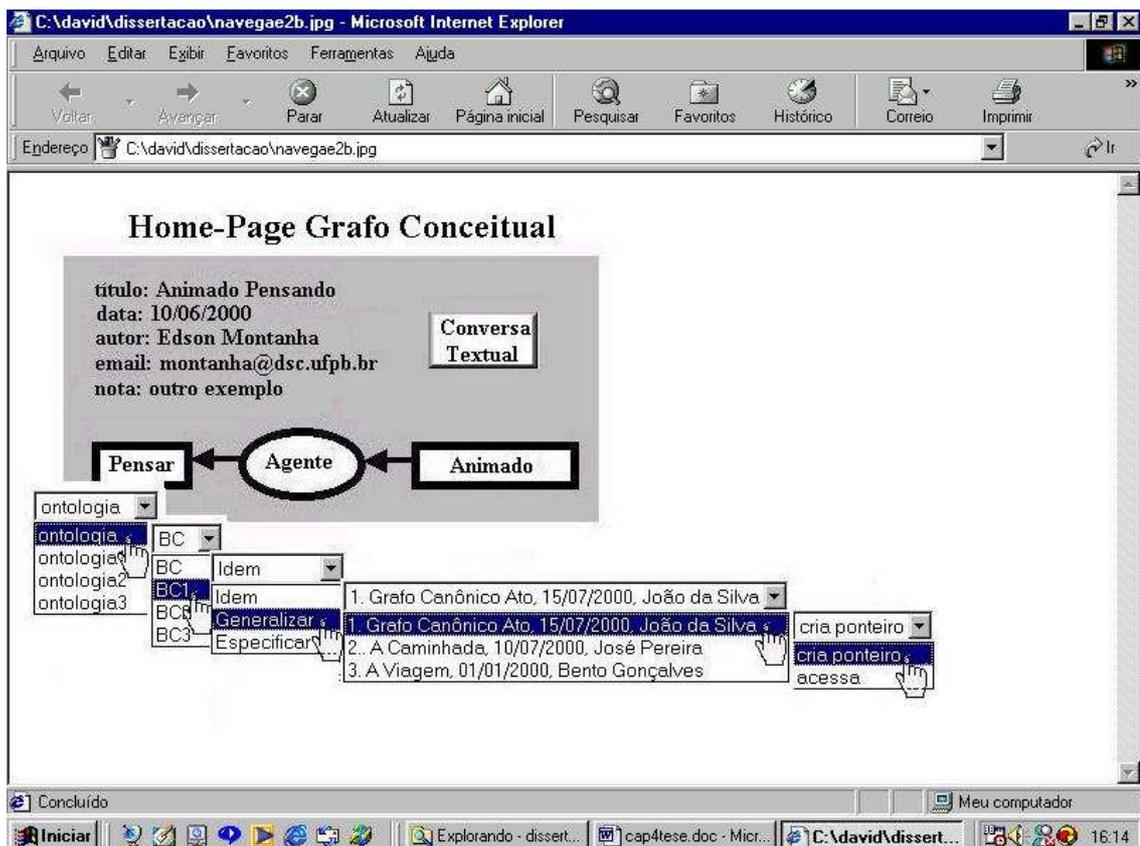


Figura 5.7. Navegação na Interface de Navegação. Criação de um ponteiro.

Para criar ponteiros (dinâmicos ou persistentes) durante a navegação, o usuário faz a escolha de:

- ❑ Ontologia;
- ❑ Base de conhecimento;
- ❑ Igualdade, generalização ou especificação.

Diferente de uma "*ligação*" (link) hipertexto normal, a *ligação* (ponteiro ou link) do ambiente VEDA pode levar a várias páginas Web e não somente a uma, comportando-se como uma pesquisa (busca). Os ponteiros para as conversas textuais dos documentos são expostos na interface juntamente com as informações de título, autor, data, e-mail, ontologia e observações.

## 5.5 Interfaces de Administração do Conhecimento

As interfaces de Administração são de uso exclusivo dos administradores do conhecimento. Esses módulos não podem ser acessados pelos usuários gerais do ambiente. As funções executadas são de caráter administrativo. Os administradores do conhecimento fazem manutenção nas bases de ontologia e grafos canônicos. Por isso, eles devem ter experiência com grafos conceituais. As modificações nessas bases de dados podem afetar o comportamento do ambiente.

### 5.5.1 Interface Modelos de Ontologia

Permite ao administrador do conhecimento tratar todas as interações que tenham como propósito à criação, busca e atualização dos modelos de ontologia. O ambiente propicia a navegação pelas hierarquias de conceitos e relações das ontologias armazenadas no Repositório de Ontologia.

A interface possui as seguintes propriedades:

- ❑ A hierarquia de tipos de conceitos e relações permite múltipla herança;
- ❑ Existe uma ontologia padrão;
- ❑ Permite a criação de axiomas para tipos de conceitos e relações que serão herdados aos subtipos;
- ❑ Utilização do WebContract da Université Montpellier como interface (Figura 5.8).

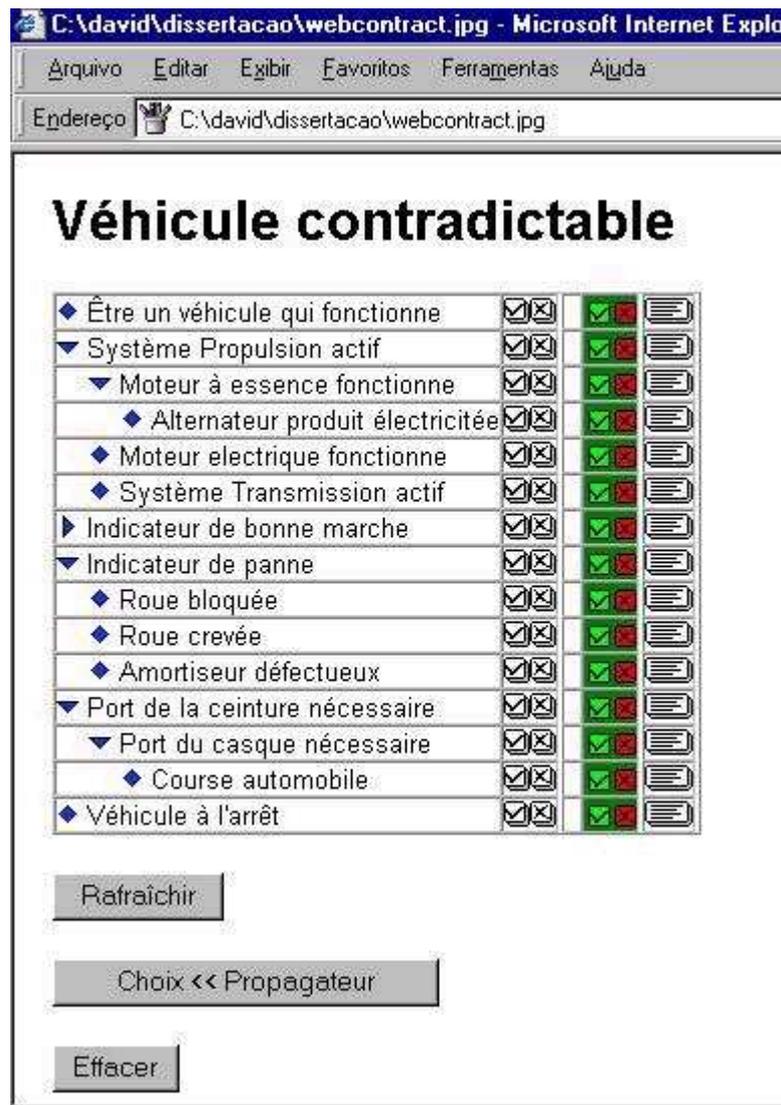


Figura 5.8. WebContract Université Montpellier.

### 5.5.2 Interface Grafos Canônicos

Esse módulo permite ao administrador do conhecimento tratar todas as interações que tenham como propósito à criação, busca e atualização dos grafos canônicos de uma determinada ontologia. Essa interface administra os grafos canônicos de forma gráfica. A interface visualiza todos os grafos canônicos referentes a uma ontologia.

## 5.6 Pré-processamento

Os módulos de pré-processamento são funções que auxiliam os módulos de gerenciamento do ambiente na execução de suas tarefas.

## 5.7 Mapeamento de GML (Graph Modeling Language) para CGIF

Esse módulo recebe os dados de posicionamento das caixas, elipses e setas juntamente com os dados de conceitos, relações e ponteiros; e os transforma para um formato CGIF com adição das coordenadas de posicionamento dos diagramas e referências dos ponteiros existentes (Figura 5.3).

## 5.8 Mapeamento de Grafos Conceituais (Forma Linear) para CGIF

Esse módulo é a API (Application Program Interface) "Notio" da University of Waterloo [NOTIO], facilitando a entrada de dados do usuário no módulo Interface de Busca para pesquisa de grafos conceituais. Dessa forma, o usuário entra com o grafo conceitual em forma linear e o Notio transforma para CGIF, deixando no formato adequado para a execução de busca no Repositório de documentos.

## 5.9 Operações com Regras de Formação Canônica

Esse módulo sempre é acionado quando for necessário generalizar ou especificar conceitos, relações e grafos conceituais. Ele interage fortemente com o módulo Máquina de Inferência. É sempre utilizado quando for necessária a verificação (validação) dos grafos conceituais gerados com os grafos canônicos da ontologia de domínio. As operações com regras de formação canônica se restringem a generalização/especialização.

## 5.10 Gerenciamento do Ambiente

Os módulos de gerenciamento do ambiente são responsáveis por gerenciar o fluxo de dados entre as interfaces e os repositórios. Eles solicitam a execução das funções de pré-processamento sempre que for necessário.

### 5.10.1 Gerenciamento de Modelos de Ontologias

Esse módulo é responsável por acessar e atualizar o Repositório de Modelos de Ontologia. O gerenciamento implementa as funções de criação, busca e atualização dos modelos de ontologia. O modelo ontológico é uma hierarquia de conceitos e relações que admite múltipla herança.

### **5.10.2 Máquina de Inferência**

Esse módulo possibilita a compatibilidade dos dados que trafegam no ambiente. Trabalha em conjunto com todas as operações que exigem unificação, como busca e operações com regras de formação canônica.

### **5.10.3 Gerenciamento do Conhecimento**

Permite a criação, acesso e atualização dos repositórios de documentos e conversa textual. Os módulos externos requisitam o serviço do Gerenciamento do Conhecimento através de documentos HTML usando uma sintaxe específica.

### **5.10.4 Gerenciamento de Grafos Canônicos**

Esse módulo é responsável por acessar e atualizar o Repositório de Grafos Canônicos. O gerenciamento implementa as funções de criação, busca e atualização dos grafos canônicos para um modelo de ontologia.

## **5.11 Repositórios**

Os módulos dos repositórios têm a função de armazenar, de forma persistente, os dados do ambiente.

### **5.11.1 Repositório Modelo de Ontologia**

Esse módulo armazena as ontologias disponibilizadas para o ambiente. O conteúdo do repositório é tratado pelo módulo Gerenciamento de Ontologia. Os modelos são armazenados em formato do WebContract.

### **5.11.2 Repositório de Documentos**

Esse módulo é responsável pela armazenagem das páginas Web criadas pelo módulo Gerenciamento do Conhecimento, em formato HTML.

### 5.11.3 Repositório de Conversa Textual

Esse módulo armazena as conversas textuais que foram feitas durante o processo de criação de uma página Web no formato HTML (Figura 5.9).

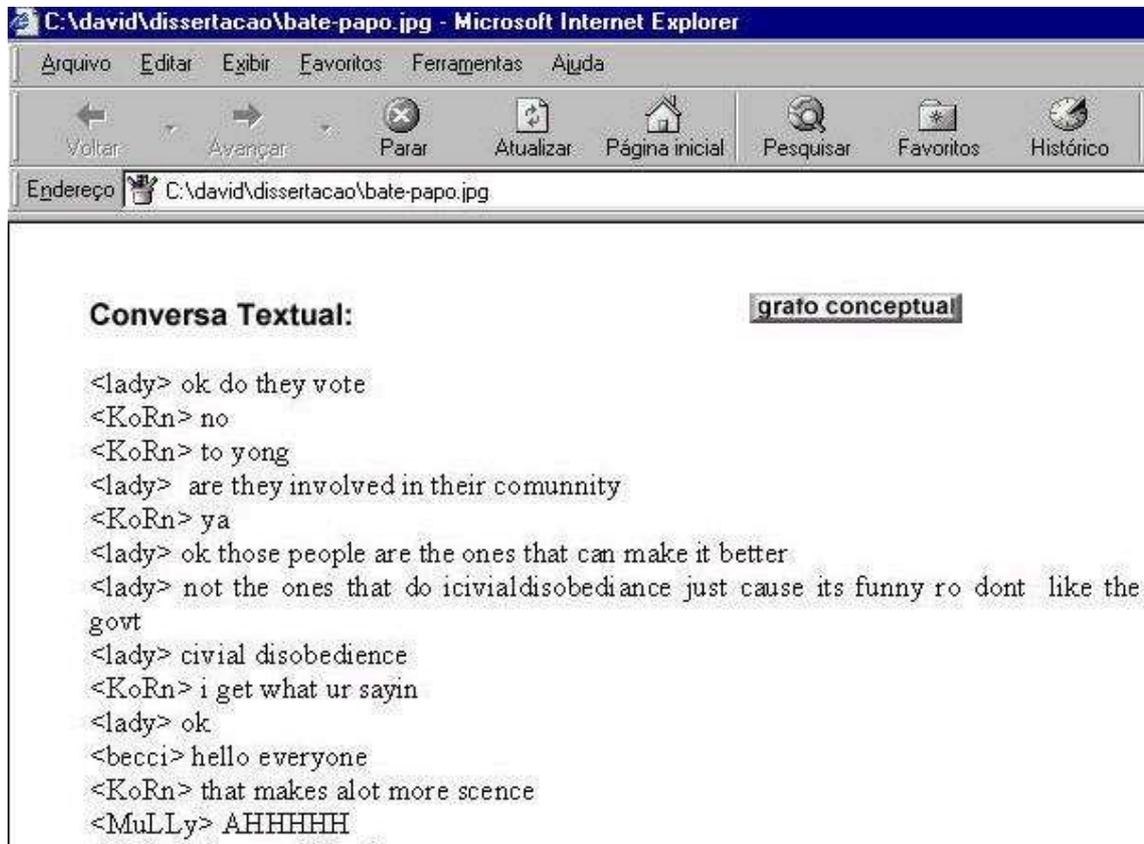


Figura 5.9. Visualização de uma Conversa Textual.

### 5.11.4 Repositório de Grafos Canônicos

Esse módulo armazena os Grafos Canônicos criados para um modelo de ontologia em formato HTML.

## 5.12 Conclusão

Criando grafos conceituais dentro desta arquitetura, os aprendizes podem refletir sobre o entendimento de suas idéias enquanto estão colaborando com outros colegas. Podemos pensar nos alunos pesquisando opiniões entre eles e desenvolvendo suas próprias interpretações. É importante ressaltar que os iniciantes no uso desse ambiente não serão capazes de começar imediatamente produzindo páginas sofisticadas. Porém, o importante é o processo de seduzi-los na representação do conhecimento.

A construção de grafos conceituais conduz professores e alunos a um conceito diferente de tratamento da aprendizagem como informação transmitida pelo professor. O conhecimento é construído em um processo de planejamento, ao planejar o conhecimento na forma de representação em grafos conceituais é possível desenvolver o pensamento crítico. Desta forma, permitindo aos aprendizes se tornarem projetistas do conhecimento. A autoria de grafos conceituais requer criatividade e complexidade, motivando intrinsecamente o aprendiz.

A arquitetura pode ajudar os aprendizes a desenvolver habilidades sociais de leitura, escrita, comunicação e colaboração, através de participação em discussões on-line. Participando em um discurso global, os aprendizes são expostos a uma grande diversidade de perspectivas, ampliando seus pontos de vista. O suporte a aprendizagem em rede (Internet) aumenta a capacidade de ensino/aprendizagem, promovendo a possibilidade de participação de comunidades de outros continentes com baixo custo.

O processo de construção dos grafos conceituais na arquitetura inclui:

- ❑ Identificação de conceitos importantes;
- ❑ Organização dos conceitos de forma espacial;
- ❑ Identificação das relações entre os conceitos;
- ❑ Designação da natureza semântica dos conceitos e relações;
- ❑ Criação de ponteiros de conceitos e relações com outros grafos conceituais;
- ❑ Reorganização do conhecimento na mente;

Os grafos conceituais podem ser utilizados para representar o que os aprendizes aprenderam sobre um assunto. Os alunos podem refletir sobre os grafos, analisando e estruturando as idéias na memória a respeito de um domínio.

---

## 6 Conclusão

### 6.1 Considerações Finais

Ao finalizarmos esta dissertação, podemos concluir que o ambiente VEDA apresenta uma nova forma de construção do conhecimento (colaborativa) e uma maneira diferente de navegar na Web. Durante todo o processo de pesquisa, observamos que a proposta da arquitetura concebida é inovadora segundo certos aspectos:

- ❑ Existem projetos que desenvolveram softwares para a educação utilizando redes semânticas como indexação, mas não encontramos nenhum que utilizasse grafos conceituais;
- ❑ Nenhum dos softwares encontrados apresentaram a utilização de uma vasta abordagem de grafos conceituais, integrando ontologia, axiomas, grafos canônicos, regras de formação canônica e relação de conformidade.
- ❑ Nenhum dos softwares encontrados utilizam a abordagem de criação de várias ontologias para um mesmo ambiente;
- ❑ O ambiente permite a criação de ponteiros (links) de uma página Veda<sup>1</sup> para outra página Veda que integra o ambiente; de forma dinâmica, onde o ponteiro não é armazenado na página Veda; ou de forma persistente, armazenando o ponteiro na página Veda;
- ❑ O ambiente permite a generalização e especificação do conhecimento dentro do ambiente Veda;
- ❑ O ambiente permite a colaboração na criação de páginas Veda;
- ❑ O ambiente permite a colaboração na criação de ponteiros (links) de uma página Veda para outra página Veda no momento da navegação;
- ❑ O ambiente permite a navegação em forma de pesquisa através de páginas Veda;
- ❑ O ambiente permite embutir uma página Veda dentro de uma página Web;
- ❑ O ambiente permite que o usuário opte por um ponteiro (link) diferente do qual foi definido pelo autor da página Veda.

Algumas problemáticas surgiram no transcorrer da pesquisa, como:

- ❑ A implementação do software não foi realizada devido a sua complexidade;

- Para representar o conhecimento em páginas Web, foram apreciados formatos HTML e XML. A grande maioria das informações disponíveis na Internet é formulada usando Hyper-Text Mark-up Language (HTML). XML (Extensible Mark-up Language) é uma linguagem que ainda está em processo de aceitação pela comunidade de desenvolvimento de páginas Web. Caso XML tivesse sido a escolha, a arquitetura sofreria apenas algumas adaptações da linguagem HTML para XML, não alterando sua funcionalidade ou paradigma de concepção. Isto poderá ocorrer com a utilização do XGMML (Extensible Graph Mark-up and Modeling Language). No momento que XGMML estiver disponibilizado nas próximas versões dos navegadores Web mais conhecidos, como Internet Explorer e Netscape Navigator, a arquitetura VEDA poderá ser reformulada ou adaptada, visando uma melhor performance. Isto devido à arquitetura atual utilizar applet Java para representar visualmente grafos conceituais. XGMML representará gráficos através do código fonte de uma página Web.

## 6.2 Sugestões de trabalhos futuros

A principal sugestão para trabalhos futuros é a implementação do ambiente VEDA. Algumas idéias surgiram para uma ampliação dos objetivos gerais apresentados para este trabalho, e desta forma, fogem do escopo da proposta inicial. Isto ocorreu devido ao crescimento tecnológico das áreas de Internet e aprendizagem colaborativa. Além disso, o conhecimento é uma área de demanda atual que necessita mais investigação, e grafos conceituais, ainda foram pouco explorados, podendo ser uma área de pesquisa que poderá ainda trazer resultados pragmáticos.

Visando o aprimoramento da arquitetura, algumas idéias podem ser citadas, como:

- A inclusão das regras de formação canônica de junção e desmembramento;
- A inclusão de som e imagem no ambiente colaborativo da arquitetura VEDA.

---

<sup>1</sup> Página Veda, significa que é uma página Html criada dentro do ambiente Veda.

## Anexo A - Conceitos

Para cada tipo, um ou mais rótulos de tipos são apresentados como supertipos. Muitos dos tipos de conceitos, especialmente os que representam ações, atributos, e tipos de função são anexados a certos tipos de relações. Para esses tipos, um grafo canônico é listado mostrando o contexto esperado. Os grafos canônicos definem limitações de seleção em combinações possíveis de conceitos. As sentenças que seguem cada tipo são comentários informais que poderão ajudar o leitor a relacionar o tipo a rede semântica de sua própria representação mental.

*AMOR* < *ESTADO*. O amor é o estado de uma experiência de um ser animado em relação a uma entidade.

[AMOR] -

(EXPR) → [ANIMADO]

(OBJ) → [ENTIDADE].

*ANIMADO* < *ENTIDADE*. Os seres animados são os agentes de ações; eles incluem não-animais, como os anjos e robôs.

*ANIMAL* < *ANIMADO*, *ENTIDADE-MÓVEL*, *OBJFISIC*. Os animais são objetos físicos, diferentes de anjos; e não são máquinas, diferentes de robôs.

*ANIMAL-DOMÉSTICO* < *ANIMAL*. Um animal domesticado é um animal que representa a função ser domesticado.

[ANIMAL-DOMÉSTICO] ← (POSS) ← [PESSOA].

*ANJO* < *ANIMADO*, *ENTIDADE-MÓVEL*, *¬OBJFISIC*. Um anjo é um ser animado, mas não é um animal.

*ATO* < *EVENTO*. Um ato é um evento com um agente animado.

[ATO] → (AGNT) → [ANIMADO].

*ATRIBUTO* < *T*. Um atributo é uma qualidade de uma entidade.

[ATRIBUTO] ← (ATRB) ← [ENTIDADE].

*BOLO*<*COMIDA, OBJFISIC*. Bolo é um tipo de comida, e é um objeto físico.

*CARACTERÍSTICA*<*ATRIBUTO*. Uma característica é um atributo essencial como idade.

*CHEGADA*<*ATO*. Uma entidade móvel chega a um lugar.

[CHEGADA] -

(AGNT)→[ENTIDADE-MÓVEL]

(LOC)→[LUGAR].

*CIDADE*<*LUGAR, SOCIEDADE*. Uma cidade pode ser vista como um lugar ou uma sociedade.

*COMANDO*<*MENSANGEM*. Um comando é uma mensagem dada a uma pessoa que está sendo ordenada para fazer alguma coisa.

[COMANDO]←(OBJ)←[ORDEM]→(RCPT)→[PESSOA].

*COMUNICAÇÃO*<*DAR*. Comunicar é dar uma informação; o instrumento é um meio de comunicação como uma correspondência, telefone, fala ou gesto.

[COMUNICAÇÃO] -

(AGNT)→[ANIMADO]

(RCPT)→[ANIMADO]

(INST)→[ENTIDADE]

(OBJ)→[INFORMAÇÃO].

*CONJUNTO*<*ATO*. Um agente animado agrupa uma entidade.

[CONJUNTO] -

(AGNT)→[ANIMADO]

(OBJ)→[OBJFISIC]

(LOC)→[LUGAR].

*CONTÉM*<*ESTADO*. Uma entidade x contém uma entidade y, se y estiver localizada no interior z de x.

[CONTÉM] -

(LOC)→[ENTIDADE: \*x]→(PART)→[INTERIOR: \*z]

(OBJ)→[ENTIDADE: \*y]→(LOC)→[INTERIOR: \*z].

*COR*<*ATRIBUTO*. Uma cor é um atributo de um objeto físico.

[COR]←(ATRB)←[OBJFISIC].

*CORTE*<*ATO*. Um ser animado corta um objeto físico com outro objeto físico que tem atributo afiado.

[CORTE] -

(AGNT)→[ANIMADO]

(INST)→[OBJFISIC]→(ATRB)→[AFIADO]

(OBJ)→[OBJFISIC].

*COZINHA*<*AMBIENTE*. Uma cozinha é um tipo de ambiente de uma casa.

*CRENÇA*<*ESTADO*. Acreditar é experimentar um estado particular em respeito a uma proposição.

[CRENÇA] -

(EXPR)→[ANIMADO]

(OBJ)→[PROPOSICAO].

*CRIANÇA*<*PESSOA*. Uma criança é uma pessoa que é ligada a alguma outra pessoa pela relação (CRNC).

[CRIANÇA]←(CRNC)←[PESSOA].

*DEITAR*<*ATO*. Um agente animado deita um objeto físico em um lugar, depois disso, o objeto está em um estado deitado.

[DEITAR] -

(AGNT)→[ANIMADO]

(OBJ)→[OBJFISIC]

(LOC)→[LUGAR].

*DESGASTE*<*PROCESSO*. Algo desgasta um objeto físico.

[FERRAMENTA]←(INST)←[DESGASTE]→(OBJ)→[OBJFISIC].

*DIFICULDADE*<*MANEIRA*. Um ato tem uma maneira difícil se existir um ser animado que possa experimentar a dificuldade.

[DIFICD] -

(EXPR)→[ANIMADO]

(MANR)→[ATO].

*ENSINAR*<ATO. Um ser animado ensina algum assunto para um outro ser animado.

[ENSINAR] -

(AGNT)→[ANIMADO]

(RCPT)→[ANIMADO]

(OBJ)→[ASSUNTO].

*ENTIDADE*< $\tau$ . As entidades incluem objetos físicos tão bem quanto às abstrações.

*ENTIDADE-MÓVEL*<*ENTIDADE*. Entidades móveis são entidades que podem se mover.

*ESTADO*< $\tau$ . Estado tem duração. Eventos não têm duração, eles estão em fluxo.

[ESTADO] -

(DUR)→[PERÍODO-TEMPO]

(LOC)→[LUGAR].

*EVENTO*<  $\tau$ . Os eventos incluem atos de agentes animados e acontecimentos como explosões, onde um agente pode não estar presente.

*DAR*<ATO. Um ato de dar pressupõe um doador, um receptor, e um presente.

[DAR] -

(AGNT)→[ANIMADO]

(RCPT)→[ANIMADO]

(OBJ)→[ENTIDADE].

*FAZER*<ATO. O fazer costuma usar algum material para chegar a um objetivo.

[FAZER] -

(AGNT)→[ANIMADO]

(MATR)→[SUBSTÂNCIA]

(RSLT)→[OBJFISIC].

*FERRAGENS*<*MATERIAL*. Ferragens é um material, diferente de pregos e parafusos que são contados como entidades separadas.

*FERRAMENTAS*<*ENTIDADE*. Uma ferramenta é uma entidade que representa a função de um instrumento para algum ato.

[*FERRAMENTA*]←(*INST*)←[*ATO*].

*GATO*<*ANIMAL*. Um gato é um animal, do tipo natural, e não tem necessariamente um grafo canônico.

*GRANDE*<*MAGNITUDE*. Comparações como *GRANDE* requerem uma escala que pode ser diferente para cada tipo que está sendo comparado.

[*GRANDE*] -

(*TMNH*)←[*OBJFISIC*]

(*COMP*)→[*OBJFISIC*].

*IDADE*<*CARACTERÍSTICA*. Idade é uma característica de uma entidade em um ponto do tempo.

[*IDADE*] -

(*CRCT*)→[*ENTIDADE*]

(*PTMP*)→[*TEMPO*].

*INFORMAÇÃO*<*T*. A informação inclui qualquer coisa que pode ser comunicada.

*INTERIOR*<*ENTIDADE*. O interior é uma função representada como parte de um objeto físico.

[*INTERIOR*]←(*PART*)←[*ENTIDADE*].

*LUGAR*<*ENTIDADE-ESTACIONÁRIA*. Um lugar é uma função representada por um lugar estacionário.

[*LUGAR*]←(*LOC*)←[*τ*].

*MACACO*<*FERRAMENTA*. Um macaco é uma ferramenta, neste caso, e não um tipo de animal.

*MEDIDA* < *T*. A medida não tem supertipos, a não ser o *T*.

*MENSAGEM* < *INFORMAÇÃO*. Uma mensagem é informação na função de ser um comunicado.

[*MENSAGEM*] ← (*OBJ*) ← [*COMUNICAÇÃO*].

*NAVIO* < *ENTIDADE-MÓVEL*. Um navio navega no mar.

*OBJFISIC* < *ENTIDADE*. Um objeto físico é um tipo de entidade.

*ORDEM* < *COMUNICADO*. Uma ordem é um tipo de comunicação.

*PARENTE* < *PESSOA*. Um parente (relativo a pais) é ligado a uma outra pessoa pela relação (*CRNC*), mas na direção oposta do tipo *CRIANÇA*.

[*PARENTE*] → (*CRNC*) → [*PESSOA*].

*PENSAMENTO* < *PROPOSIÇÃO*. O pensamento é o objeto de pensar.

[*PENSAMENTO*] ← (*OBJ*) ← [*PENSAR*].

*PENSAR* < *ATO*. O pensar é um ato que seres animados executam sobre proposições.

[*PENSAR*] -

(*AGNT*) → [*ANIMADO*]

(*OBJ*) → [*PROPOS*].

*PESSOA* < *ANIMAL*. Uma pessoa é um tipo de animal.

*PROPOSIÇÃO* < *INFORMAÇÃO*. Uma proposição é um tipo de informação simbólica.

*RECEBER* < *ATO*. Para o verbo receber, o sujeito é um receptor.

[*RECEBER*] -

(*RCPT*) → [*ANIMADO*]

(*INST*) → [*ENTIDADE*]

(*OBJ*) → [*ENTIDADE*]

(*ORGM*) → [*LUGAR*].

*REMESSA* < *TRANSPORTE*. Uma remessa é um ato de expedir.

[REMESSA] -

(AGNT)→[ANIMADO]

(INST)→[VEÍCULO]

(ORGM)→[LUGAR]

(DEST)→[LUGAR]

(CAMH)→[LUGAR:{\*}]

(OBJ)→[OBJFISIC].

*ROBÔ*<*ANIMADO, MÁQUINA, ENTIDADE-MÓVEL*. Um robô é uma máquina animada que pode se mover.

*SABER*<*ESTADO*. O saber é experimentar um estado particular com respeito a uma proposição.

[SABER] -

(EXPR)→[ANIMADO]

(OBJ)→[PROPOSIÇÃO].

*TELEFONE*<*OBJFISIC*. Um telefone é um meio de comunicação.

[TELEFONE]←(INST)←[COMUNICAÇÃO].

*TRABALHO*<*ATO*. O trabalho é um tipo de ato.

*TRABALHO-DE-CASA*<*TRABALHO*. O trabalho de casa é um tipo de trabalho.

*UNICÓRNIO*<*MAMÍFERO, CRIATURA-MÍSTICA*. Um unicórnio é uma criatura mística que poderia ser um mamífero, caso existisse.

*USO*<*ATO*. O objeto do verbo usar é o instrumento de algum ato não especificado.

[USO] -

(AGNT)→[ANIMADO]

(INST)→[¬ANIMADO].

*VESTIR*<*ATO*. Pessoas vestem roupas.

[HUMANO]←(AGNT)←[VESTIR]→(OBJ)→[ROUPA].

$\top$  > todos os outros tipos.

$\perp$  > todos os outros tipos.

## Anexo B - Relações Conceptuais

Para cada relação conceptual apresenta-se o nome da relação, por exemplo "agente", seguido do rótulo de tipo (quatro letras ou menos), por exemplo (AGNT). Depois seguem os tipos generalizados dos conceitos que serão ligados a cada arco da relação, como "ligue [ATO] a [ANIMADO]". Depois temos um comentário informal sobre o uso da relação. Finalmente, temos um exemplo que mostra como a relação pode ser usada.

*Acompanhamento.* (ACMP) liga [ENTIDADE: \*x] a [ENTIDADE: \*y], onde \*y esta acompanhando \*x. Exemplo: Ronaldo partiu com Nana.

[PARTIR]→(AGNT)→[PESSOA: Ronaldo]→(ACMP)→[PESSOA: Nana].

*Agente.* (AGNT) liga [ATO] a [ANIMADO] onde o conceito ANIMADO representa o ator da ação. Exemplo: Eva mordeu uma maçã.

[PESSOA: Eva]←(AGNT)←[MORDER]→(OBJ)→[MAÇÃ].

*Argumento.* (ARG) liga [FUNÇÃO] a [DATA] sendo uma entrada da função. Se a função tem mais de uma entrada, os argumentos devem ser distinguidos como ARG1, ARG2, ARG3, .... Essas relações são usadas primariamente para representação de expressões matemáticas e não de linguagens naturais. Exemplo: SQRT(16)=4.

[NUMERO: 16]←(ARG)←[SQRT]→(RSLT)→[NUMERO:4].

*Até.* (ATÉ) liga um ESTADO a um TEMPO no qual o estado deixa de existir. Exemplo: Os bilhetes são válidos até as 10:00h.

[ESTADO: [BILHETES]→(ATRB)→[VÁLIDO]→(ATÉ)→[TEMPO: 10:00h].

*Atributo.* (ATRB) liga [ENTIDADE: \*x] a [ENTIDADE: \*y], onde \*x tem um atributo \*y. Exemplo: A rosa é vermelha.

[ROSA: #]→[ATRB]→[VERMELHA].

*Caminho.* (CMNH) liga um [ATO] a um conjunto de [LUGAR](s) aonde as ações ocorrem. Exemplo: A pizza foi enviada via Goiânia e Palmas.

[PIZZA: #]←(OBJ)←[ENVIAR]→(CMNH)→[CIDADE: {Goiânia, Palmas}].

*Característica.* (CRCT) liga [ENTIDADE: \*x] a [ENTIDADE: \*y], onde \*x tem uma característica \*y. Exemplo: Euler tinha 100 anos de idade.

(PASS)→[PROPOSIÇÃO: [PESSOA: Euler]→(CRCT)→(IDADE: @100anos)].

*Causa.* (CAUS) liga [ESTADO: \*x] a [ESTADO: \*y], onde \*x tem causa \*y. Exemplo: Se você está molhado, então está chovendo.

[ESTADO: [PESSOA: Você]←(EXPC)←[MOLHADO]]→(CAUS)→[ESTADO: [CHOVER]].

*Criança.* (CRNC) liga uma [PESSOA] a outra [PESSOA], que é uma criança da seguinte forma. Exemplo: Lilia é mãe de Kátia.

[MÃE: Lilia]→(CRNC)→[PESSOA: Kátia].

*Conteúdo.* (CONT) liga [ENTIDADE: \*x] a [ENTIDADE: \*y], onde \*x tem o conteúdo \*y. Isto pode ser definido em termos das relações LOC e PART. Exemplo: Um bebê esta dentro do cercado.

[CERCADO]→(CONT)→[BEBÊ].

*Destino.* (DEST) liga um [ATO] a uma [ENTIDADE] para onde a ação for direcionada. Exemplo: João foi a Maceió.

[PESSOA: João]←(AGNT)←[IR]→(DEST)→[CIDADE: Maceió].

*Duração.* (DUR) liga um [ESTADO] a um [PERÍODO-TEMPO] durante o qual o estado persiste. Exemplo: O caminhão prestou serviço por 5 horas.

[CAMINHÃO:#]←(OBJ)←[SERVIÇO]→(DUR)→[PERÍODO-TEMPO: @5hs].

*Experiente.* (EXPR) liga um [ESTADO] a um [ANIMADO] que esta experimentando este estado. Exemplo: Clara está com frio.

[PESSOA: Clara]←(EXPR)←[FRIO].

*Freqüência.* (FREQ) liga um [EVENTO] a um espaço de [TEMPO] no qual ele ocorre. Exemplo: As embalagens são enviadas as segundas-feiras.

[EMBALAGENS: {\*}]←(OBJ)←[ENVIAR]→(FREQ)→[SEGUNDAS-FEIRAS: {\*}].

*Início.* (INIC) liga um [ATO] a um [ANIMADO] que é responsável pelo início, mas que não o executa diretamente. Exemplo: Tonho ferveu as batatas.

[PESSOA: Tonho]←(INIC)←[FERVER]→(OBJ)→[BATATAS: {\*}].

*Instrumento.* (INST) liga uma [ENTIDADE] a um [ATO] no qual a entidade esta casualmente envolvida. Exemplo: A chave abriu a porta.

[CHAVE: #]←(INST)←[ABRIR]→(OBJ)→[PORTA: #].

*Ligação.* (LIG) liga [T] a [T]. É usada primariamente como uma primitiva de forma que todas as outras relações podem ser definidas. A relação tipo AGNT pode ser definida em termos de um tipo de conceito AGENTE.

Relação AGNT(x,y) é

[ATO: \*x]→(LIG)→[AGENTE]→(LIG)→[ANIMADO: \*y].

*Local.* (LOC) liga um [T] a um [LUGAR]. Exemplo: Os veículos chegam a estação.

[VEICULO: {\*}]←(AGNT)←[CHEGAR]→(LOC)→[ESTAÇÃO].

*Maneira.* (MANR) liga um [ATO] a um [ATRIBUTO]. Exemplo: A ambulância chegou rapidamente.

[AMBULÂNCIA: #]←(AGNT)←[CHEGAR]→(MANR)→[RÁPIDA].

*Material.* (MATR) liga um [ATO] a uma [SUBSTÂNCIA] usada no processo. Exemplo: A arma foi entalhada no sabão.

[ARMA]←(RSLT)←[ENTALHAR]→(MATR)→[SABÃO].

*Medida.* (MED) liga uma [DIMENSÃO] a uma [MEDIDA] dessa dimensão. Exemplo: O esqui tem 167cm de comprimento.

[ESQUI]→(CRCT)→[TAMANHO]→(MED)→[MEDIDA: 167cm].

Pela contração da medida, a relação MED pode ser contraída para formar o conceito [TAMANHO: @167cm].

*Método.* (MET) liga um [ATO: \*x] a uma [SITUAÇÃO: \*y], que mostra como o ato \*x é executado. Exemplo: Leo pegou o bumerangue com um salto potente.

[ATO: [PESSOA: Leo=\*x]←(AGNT)←[PEGAR]→(OBJ)→[BUMERANGUE]] -  
(MET)→[ATO: [PESSOA: \*x]←(AGNT)←[SALTO]→(MANR)→[POTENTE]].

*Negação.* (NEG) é uma relação monada que é ligada a uma [PROPOSIÇÃO] que é declarada falsa. Exemplo: Kiko não comeu a maçã.

(NEG)→[PROPOSIÇÃO: [PESSOA: Kiko]←(AGNT)←[COMER]→(OBJ)→ [MACA]].

*Necessidade.* (NECS) é uma relação monada que é ligada a uma [PROPOSIÇÃO] que é necessariamente verdadeira. Exemplo: É necessariamente verdadeiro que uma mulher seja fêmea.

(NECS)→[PROPOSIÇÃO: [MULHER]→(ATRB)→[FÊMEA]].

*Nome.* (NOME) liga uma [ENTIDADE] a uma [PALAVRA] que é o nome da entidade. Exemplo: Cícero é chamado Túlio.

[PESSOA: Cícero]→(NOME)→["Túlio"].

Exemplo: "4" e "IV" são nomes para o mesmo número.

["4"]←(NOME)←[NÚMERO]→(NOME)→[NÚMERO].

*Objeto.* (OBJ) liga um [ATO] a uma [ENTIDADE] a qual foi atuada. Exemplo: O gato engoliu o canário.

[GATO:#]←(AGNT)←[ENGOLIR]→(OBJ)→[CANÁRIO:#].

*Origem.* (ORGM) liga um [ATO] a uma [ENTIDADE] da qual ela originou. Exemplo: O balde foi carregado do galpão.

[BALDE:#]←(OBJ)←[CARREGAR]→(ORGM)→[GALPÃO].

*Parte.* (part) liga uma [ENTIDADE: \*x] a uma [ENTIDADE: \*y] onde \*y é parte de \*x. Exemplo: Um dedo é parte da mão.

[MÃO]→(PART)→[DEDO].

*Passado.* (PASS) é uma relação monada que é ligada a uma [PROPOSIÇÃO] que é verdadeira em um momento que antecede o presente. Exemplo: Júlio abandonou.

(PASS)→[PROPOSIÇÃO: [PESSOA: Júlio]←(AGNT)←[ABANDONAR]].

*Ponto no tempo.* (PTMP) liga [T] para um [TEMPO] no qual ocorre. Exemplo: Às 5:25, Érica embarcou.

[TEMPO: 5:25] ← (PTMP) ← [PROPOSIÇÃO: [PESSOA: Érica] ← (AGNT) ← [EMBARQUE]].

*Possessão.* (POSS) liga um [ANIMADO] a uma [ENTIDADE] que é possuída pelo agente animado. Exemplo: O relógio de Guga parou.

[PESSOA: Guga]→(POSS)→[RELÓGIO-PULSO]←(OBJ)←[PARAR].

*Possibilidade.* (POSB) é uma relação monada que é ligada a uma [PROPOSIÇÃO] que é possivelmente verdadeira. Exemplo: O bebê pode falar.

(POSB)→[PROPOSIÇÃO: [BEBÊ:#]←(AGNT)←[FALA]].

*Quantidade.* (QTD) liga um conjunto de [ENTIDADE: {\*}] a um [NÚMERO] que indica o número de entidades do conjunto. Exemplo: Existem 50 passageiros no ônibus.

[ÔNIBUS:#]←(LOC)←[PASSAGEIRO: {\*}]→(QTD)→[NÚMERO: 50].

Pela contração da quantidade, a relação QTD pode ser contraída para formar o conceito [PASSAGEIRO: [{\\*}@50](#)].

*Recipiente.* (RCPT) liga um [ATO] a um [ANIMADO] que recebe o objeto ou resultado da ação. Exemplo: Os diamantes foram dados para Ângela.

[DIAMANTES: {\*}]←(OBJ)←[DAR]→(RCPT)→[PESSOA: Ângela].

*Resultado.* (RSLT) liga um [ATO] a uma [ENTIDADE] que é gerada pela ação. Exemplo: Gabriel construiu uma casa.

[PESSOA: Gabriel]←(AGNT)←[CONTRUIR]→(RSLT)→[CASA].

*Sucessor.* (SUCS) liga [T] a outro [T], que segue o primeiro. Exemplo: Depois que Brenda foi à escola, ela foi passear no parque.

[EVENTO: [PESSOA: Brenda= \*x]←(AGNT)←[IR]→(LOC)→[ESCOLA: #]] - (SUCS)→[EVENTO: [PESSOA: \*x]←(AGNT)←[PASSEAR]→(LOC)→[PARQUE]].

*Suporte.* (SUPT) liga uma [ENTIDADE: \*x] a outra [ENTIDADE: \*y] onde \*x tem suporte \*y. Exemplo: O gelo esta sobre a abóbora.

[GELO]→(SUPT)→[ABÓBORA].

---

## Anexo C - Uso do Arco com Setas

Para lembrar a direção das setas dos arcos, leia o nome da relação e os conceitos anexados para a seguinte frase: "a relação de um conceito1 é a um conceito2". Dessa convenção, temos: [Morder] → (Agnt) → [Pessoa: Eva].

A frase pode ser lida como: "o agente da Mordida é a Pessoa, Eva". Uma convenção equivalente é ler "uma Mordida tem um agente que é uma pessoa, Eva". Examinando o grafo na direção oposta às setas, outra convenção é ler as setas apontando para fora do círculo como "é um", e apontar em direção ao círculo como "de". Com essa convenção, o grafo acima poderia ser lido: Pessoa, Eva é um agente de Mordida.

## Anexo D - Categorias de Nível Superior

As categorias ontológicas (Figura A.1) são derivadas de uma pesquisa avançada em filosofia, lingüística e inteligência artificial. Elas fornecem uma moldura para organização dos tipos de conceitos mais especializados usados em bases de conhecimento ou processamento de linguagem natural.

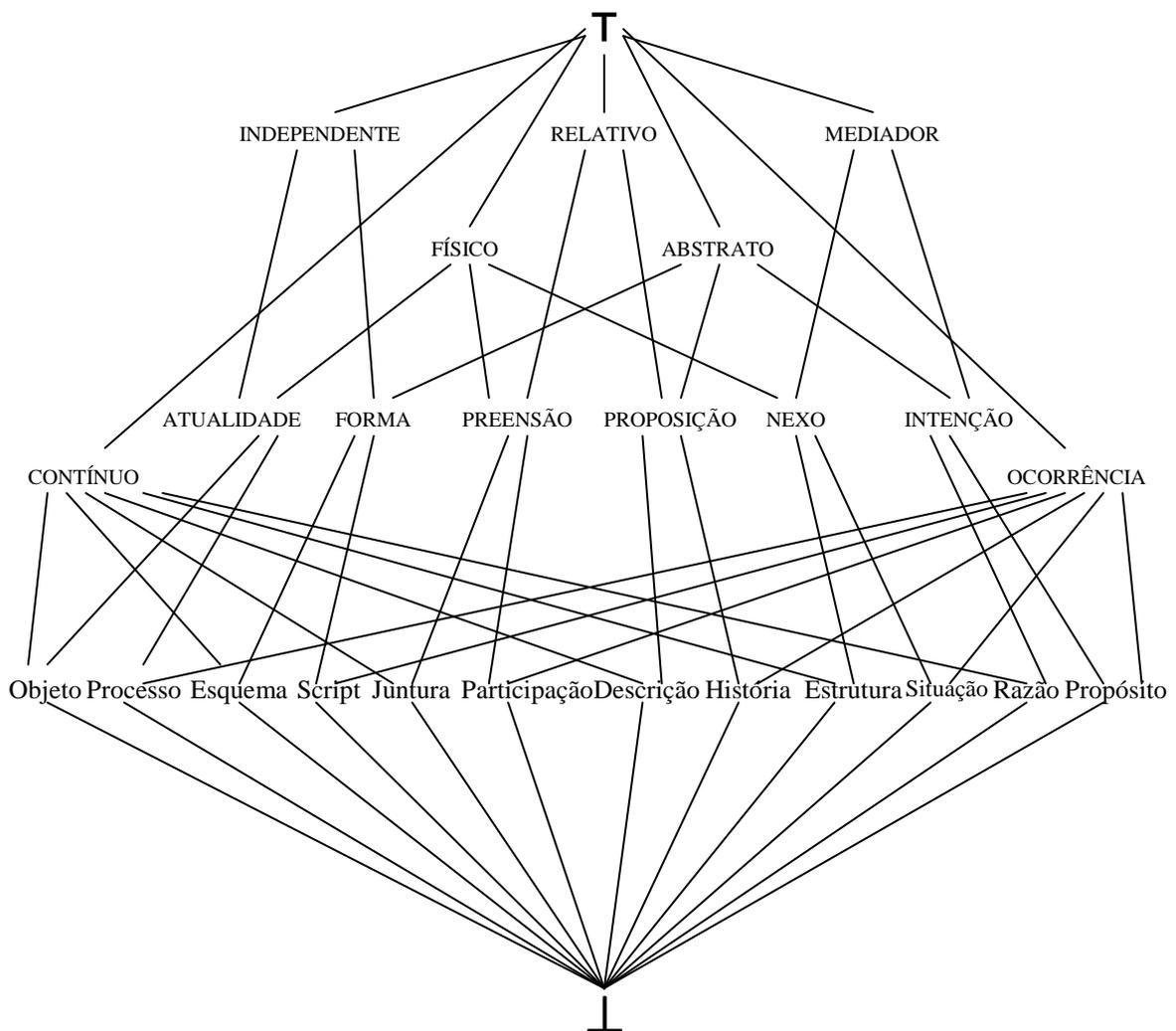


Figura A.1. Reticulado das categorias de nível superior.

Os tipos  $\tau$  e  $\perp$  podem ser representado pelos tipos Entidade e Absurdo. Existem nove categorias nessa lista que possuem axiomas associados, são:  $\tau$ ,  $\perp$ , Independente, Relativo, Mediador, Físico, Abstrato, Contínuo e Ocorrência. Os outros tipos são definidos como maior subtipo comum de dois supertipos, herdando seus axiomas. Por exemplo, o Tipo Forma é

definido  $\text{Independente} \cap \text{Abstrato}$ , assim ele herda os axiomas de Independente e Abstrato, e sua abreviatura é IA para indicar seus dois supertipos.

$\top$  (). O tipo universal que não tem diferenciador. Formalmente,  $\top$  é uma primitiva que satisfaz os seguintes axiomas:

- Existe alguma coisa:  $(\exists x)\top(x)$ .
- Tudo é uma instância de  $\top$ :  $(\forall x)\top(x)$ .
- Todo tipo é um subtipo de  $\top$ :  $(\forall t: \text{Tipo})t \leq \top$ .

$\perp$  (IRMFACO). O tipo absurdo que herda todas as características. Formalmente,  $\perp$  é uma primitiva que satisfaz os seguintes axiomas:

- Nada é uma instância de  $\perp$ :  $\sim(\exists)\perp(x)$ .
- Todo tipo é um supertipo de  $\perp$ :  $(\forall t: \text{Tipo})\perp \leq t$ .

Abstrato (A). Uma estrutura pura de informação. Formalmente, Abstrato é uma primitiva que satisfaz os seguintes axiomas.

- Nenhuma abstração tem um local no espaço:  $\sim(\exists x: \text{Abstrato})(\exists y: \text{Lugar}) \text{loc}(x,y)$ .
- Nenhuma abstração ocorre em um ponto do tempo:  $\sim(\exists x: \text{Abstrato})(\exists t: \text{Tempo}) p\text{Tmp}(x,t)$ .

Absurdo (IRMFACO) =  $\perp$ . Um sinônimo pronunciável para  $\perp$ . Ele não pode ser o tipo de nada que exista.

Atualidade (IF) = Independente  $\cap$  Físico. Uma entidade física (F) cuja existência é independente (I) de qualquer outra entidade.

Contínuo (C). Uma entidade cuja identidade é preservada em algum intervalo de tempo. Formalmente, Contínuo é uma primitiva que satisfaz os seguintes axiomas:

- Um contínuo  $x$  tem partes espaciais e não tem parte temporal. Em qualquer tempo  $t$  que  $x$  existe, todos os  $x$  existem ao mesmo tempo.
- As condições de identidade para um contínuo são independente de tempo. Se  $c$  é um subtipo de Contínuo, então o predicado  $\text{Id}(x,y)$  para identificar duas instâncias  $x$  e  $y$  de tipo  $c$ , não dependem do tempo.

Descrição (RAC) = Proposição  $\cap$  Contínuo. Uma proposição (RA) sobre um contínuo (C) relacionando um esquema (IAC) como uma caracterização do contínuo. Um esquema por ele mesmo é um padrão não instanciado. Uma descrição é a aplicação de um esquema para descrever um contínuo, seja físico ou abstrato. Em uma descrição verbal, a descrição é uma sentença que caracteriza algum aspecto de contínuo. Uma descrição expressa um aspecto ou configuração de contínuo, mostrando como ele se corresponde com a estrutura de um esquema (IAC). Uma pintura ou fotografia pode ser usada como uma descrição quando acompanhada de uma referência para um objeto (IFC) que a imagem descreve.

Entidade (). T. Um sinônimo pronunciável para T. Ela pode ser usada como um rótulo padrão para qualquer coisa de qualquer tipo.

Esquema (IAC) = Forma  $\cap$  Contínuo. Uma forma (IA) que tem a estrutura de um contínuo (C). Um esquema é uma forma abstrata (IA) cuja estrutura não especifica tempo. Exemplos incluem formas geométricas, as estruturas sintáticas das sentenças de uma linguagem ou a codificação de fotos em sistemas multimídia.

Estrutura (MFC) = Nexo  $\cap$  Contínuo. Um nexo (M) considerado como um contínuo (C) por alguma razão (MAC) que explique, como as juntas de seus componentes são organizadas para uma função. A razão não precisa ser conscientemente entendida por um agente humano. Exemplo: estrutura pode ser um ninho de passarinho ou uma colmeia.

Físico (F). Uma entidade que tem uma local no tempo e no espaço. Formalmente, físico é uma primitiva que satisfaz o seguinte axioma:

- Qualquer coisa física é localizada em algum lugar:  $(\forall x: \text{Físico})(\exists y: \text{Lugar}) \text{loc}(x,y)$ .
- Qualquer coisa física ocorre em um ponto no tempo:  $(\forall x: \text{Físico})(\exists t: \text{Tempo}) p\text{Tmp}(x,t)$ .

Axiomas mais detalhados que relacionam entidades físicas no espaço, tempo, matéria e energia, podem envolver uma grande negociação da teoria física sendo que a maioria está incompleta.

Forma (IA) = Abstrato  $\cap$  Independente. Informação Abstrata (A) e Independente (I) de qualquer materialização.

História (RAO) = Proposição  $\cap$  Ocorrência. Uma proposição (RA) sobre uma ocorrência (O). Uma história é uma proposição (RA) que relaciona algum script (IAO) para os estágios de

uma ocorrência (O). Um programa de computador, por exemplo, é um script (IAO). Um computador executando o programa é um processo (IFO) e a informação Abstrata (A) codificada pelo caminho trilhado das instruções executadas é uma história (RAO). Como qualquer proposição uma história não necessita ser verdadeira e não necessita ser predicado do passado. Exemplo: uma história lendária é uma história de uma imaginação no passado, uma profecia é uma história de um futuro esperado e uma sinopse é uma história de alguma ocorrência hipotética.

Independente (I). Uma entidade que existe independentemente de outras entidades:

$$\square (\forall x: \text{Independente}) \sim \square (\exists y)(\text{ter}(x,y) \vee \text{ter}(y,x)).$$

Se  $x$  é uma entidade independente, não é necessário que exista uma entidade  $y$  em que  $x$  tenha  $y$  ou que  $y$  tenha  $x$ .

Intenção (MA) = Abstrato  $\cap$  Mediador. Uma Abstração (A) Mediadora (M) de outras entidades.

Juntura (RFC) = Preensão  $\cap$  Contínuo. Uma preensão (RF) considerada como um Contínuo (C) durante algum intervalo de tempo. A entidade preensora é um objeto (IFC) que tem uma relação estável com uma entidade contínua. Exemplos de juntas incluem: junção dos ossos, conexões entre as partes do carro.

Mediador (M). Uma entidade que causa relacionamento entre outras entidades. Formalmente, Mediador é uma primitiva que satisfaz o seguinte axioma:

$$\square (\forall m: \text{Mediador})(\forall x,y: \text{Entidade})((\text{ter}(m,x) \wedge \text{ter}(m,y)) \supset \square (\text{ter}(x,y) \vee \text{ter}(y,x))).$$

Para qualquer entidade  $m$  e quaisquer outras entidades  $x$  e  $y$ , se  $m$  tem  $x$  e  $m$  tem  $y$ , então não é necessário que  $x$  tenha  $y$  ou que  $y$  tenha  $x$ . Uma entidade independente não necessita ter nenhuma relação; uma entidade relativa deve ter alguma relação com alguma coisa a mais; e uma entidade mediadora causa relações com outras entidades.

Nexo (MF) = Físico  $\cap$  Mediador. Uma entidade física (F) mediadora (M) que faz a mediação de duas ou mais outras entidades. Exemplos incluem: uma ação que relaciona um agente a um paciente ou uma planta que relata as partes de uma construção. As entidades mediadas pelo nexos podem ser, elas mesmas, componentes de um nexos. Exemplo: a ação consiste do que o agente faz ao paciente.

Objeto (IFC) = Atualidade  $\cap$  Contínuo. Atualidade (IF) é considerado como contínua (C) mantendo sua identidade sobre um intervalo de tempo. Apesar de nenhuma entidade física ser para sempre contínua, um objeto poder ser reconhecido pelas condições de identidade que mantém estável durante sua vida. O tipo Objeto inclui objetos físicos comuns e as instâncias de classes das linguagens de programação orientadas a objeto.

Ocorrência (O). Uma entidade que não tem uma identidade estável durante um intervalo de tempo. Formalmente, Ocorrência é uma primitiva que satisfaz os seguintes axiomas:

- As partes de uma ocorrência que são chamadas estágios, podem existir em tempos diferentes.
- Não existe critério para identificar duas ocorrências observadas em regiões não sobrepostas pelo tempo e espaço.

Participação (RFO) = Preensão  $\cap$  Ocorrência. Uma preensão (RF) considerada como uma ocorrência (O) durante um intervalo. A entidade preensora deve ser um processo (IFO) de uma relação estável com alguma entidade contínua. Exemplos de participação incluem: o latido de um cão, uma maçã sendo comida ou uma sentença sendo falada.

Processo (IFO) = Atualidade  $\cap$  Ocorrência. Atualidade (IF) é considerada como uma ocorrência (O) durante o intervalo de interesse. Dependendo da escala de tempo e nível de detalhamento, a mesma entidade atual pode ser vista como um objeto estável ou um processo dinâmico. Mesmo um diamante pode ser considerado um processo quando visto por um longo período de tempo ou pelo nível atômico das partículas vibrantes.

Preensão (RF). Uma entidade física (F) relativa (R) a alguma entidade ou entidades.

Proposição (RA). Uma abstração (A) relativa (R) a alguma entidade ou entidades. Em lógica, a afirmação de uma proposição é uma declaração que a abstração corresponde a algum aspecto da(s) entidade(s) envolvidas.

Propósito (MAO) = Intenção  $\cap$  Ocorrência. Intenção (MA) de algum agente que determina a interação das entidades na situação (MFO). Como exemplo, as palavras e notas de uma canção "Parabéns Pra Você", constituem um script (IAO). A descrição de como uma pessoa em uma festa canta a canção é história (RAO), e a intenção (MA) que explica a situação (MFO) é um propósito (MAO). Os axiomas básicos para Propósito são herdados de seus

supertipos Mediador, Abstrato e Ocorrência. Axiomas de mais baixo nível relaciona propósito a suas ações e agentes:

- Seqüência de tempo: se um agente  $x$  executa um ato  $y$ , cujo propósito é uma situação  $z$ , o início de  $y$  ocorre antes do início de  $z$ .
- Contingência: se um agente  $x$  executa um ato  $y$ , cujo propósito é uma situação  $z$ , descrita pela proposição  $p$ , então é possível que  $z$  deva não ocorrer, ou que  $p$  não seja verdadeiro de  $z$ .
- Sucesso ou fracasso: se um agente  $x$  executa um ato  $y$ , cujo propósito é uma situação  $z$ , descrita por uma proposição  $p$ , então  $x$  é dito ser bem sucedido, se  $z$  ocorre em  $p$  é verdadeiro de  $z$ ; senão  $x$  é dito ter fracassado.

Razão (MAC) = Intenção  $\cap$  Contínuo. Intenção (MA) que tem a forma de um contínuo (C). Diferente de uma simples descrição (RAC), a razão (MAC) explica uma entidade em termos de uma intenção (MA). Para a festa de aniversário, uma descrição poderia listar os presentes, mas a razão poderia explicar porque os presentes são relevantes para a festa.

Relativo (R). Uma entidade em uma relação com outra entidade.

- $(\forall x: \text{Relativo}) \square (\exists y)(\text{ter}(x,y) \vee \text{ter}(y,x))$ .
- Para qualquer relativo  $x$ , deve existir algum  $y$ , que  $x$  tem  $y$ , ou  $y$  tem  $x$ .

Script (IAO) = Forma  $\cap$  Ocorrência. A forma (IA) que tem a estrutura de uma ocorrência (C). Um script é uma forma abstrata (IA) que representa seqüências de tempo. Exemplos incluem programas de computadores, uma receita para fazer um bolo, uma partitura para ser tocada em um piano ou uma equação diferencial que governa a evolução de um processo físico.

Situação (MFO) = Nexo  $\cap$  Ocorrência. Um nexo (MF) considerado como uma ocorrência (O) para algum propósito (MAO). O aspecto mediador de uma situação (MPO) é o propósito (MAO) de algum agente que determina por que a interação das entidades na situação é significativa. A maioria dos verbos expressa situações que incluem um evento ou estado, e seus participantes. Situações complexas podem ter outras situações como componentes. Uma corte de julgamento, por exemplo, pode consistir de muitas ações de um juiz, júri, advogados e vítimas.

## Anexo E - Tipos de Relação

Dividimos o tipo Atualidade em três subtipos: Fenômeno, Função, e Símbolo. Uma entidade fenomenal é uma entidade atual considerada por ela mesma; uma função é considerada em relação a alguma coisa mais; e um símbolo é considerado como representando alguma coisa para um agente. A distinção entre Independente, Relativo, e Mediador, refere-se a natureza das entidades por elas mesmas, mas a distinção de Fenômeno, Função, e Símbolo, refere-se a maneira que as entidades são vistas por algum observador.

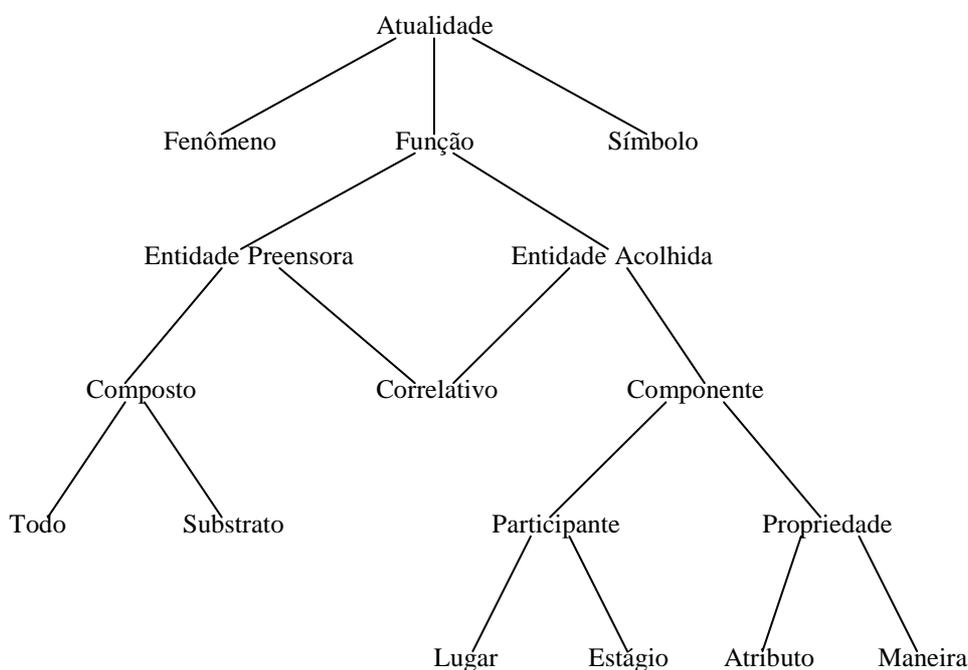


Figura A.2. Classificação de entidades de acordo com as funções que elas executam.

Toda função é envolvida numa preensão (Figura A.2). No cálculo de predicado, funções são usualmente representadas pelos predicados díades, que relacionam uma entidade preensora a uma entidade acolhida. Em frames, elas são usualmente representadas pelos slots, que são preenchidos com um identificador da entidade acolhida. Em grafos conceituais, uma expressão lambda pode ser usada, para definir um tipo de relação díade  $r$ , em termos de um tipo de conceito  $R$ , que corresponde ao nome do slot do frame:

$$R = [t: \lambda_1] \rightarrow (\text{ter}) \rightarrow [R: \lambda_2].$$

Essa definição usa o tipo de relação primitiva ter para relacionar o primeiro parâmetro que representa uma entidade preensora de tipo  $t$ , para o segundo parâmetro, que representa a entidade acolhida do tipo função  $R$ .

Na seguinte lista, cada tipo função é especificado como um subtipo de algum tipo que é independente de um observador, e de um tipo função (Figura A.2). Acompanhamento, por exemplo, é um subtipo de Objeto na função de Participante. O tipo de relação díade correspondente é  $Acmp$ , cuja assinatura (Objeto, Objeto) indica que ela relaciona um objeto a outro objeto. A maioria das sentenças exemplos pode ser parafraseada com o verbo ter e uma construção sintática que usa o nome da função.

Acompanhamento  $\langle$  Objeto  $\cap$  Participação:  $Acmp$  (Objeto, Objeto). Um objeto que participa com outro objeto em um processo.

Exemplo: Ronaldo partiu com Nana.

(Pass)  $\rightarrow$  [Situação: [Partir]  $\rightarrow$  (Agnt)  $\rightarrow$  [Pessoa: Ronaldo]  $\rightarrow$  (Acmp)  $\rightarrow$  [Pessoa: Nana]].

Paráfrase: Quando partiu, Ronaldo teve Nana como acompanhante.

Argumento  $\langle$  Dado  $\cap$  Função;  $Arg$ (Função, Dado). Dado executa o papel de entrada para uma função. Se a função tiver mais de uma entrada, os argumentos podem ser distinguidos como  $Arg1$ ,  $Arg2$ , .... Essa relação é usada primariamente para representar expressões matemáticas.

Exemplo:  $Sqrt(16)=4$ .

[Número: 16]  $\leftarrow$  (Arg)  $\leftarrow$  [Sqrt]  $\rightarrow$  (Rslt)  $\rightarrow$  [Número: 4].

Paráfrase: A raiz quadrada da função tem 16 como argumento, e 4 como resultado.

Atributo  $\langle$  Entidade  $\cap$  Propriedade;  $Atrb$ (Objeto, Entidade). Uma entidade que é uma propriedade de algum objeto. Atributos são usualmente expressados por adjetivos, como circular, vermelho, longo, e pesado.

Exemplo: A rosa é vermelha.

[Rosa: #]  $\rightarrow$  (Atrb)  $\rightarrow$  [Vermelha].

Paráfrase: A rosa tem vermelho como atributo.

Base  $\langle$  Tipo;  $Base$ (Atributo, Tipo). Um tipo de função que determina como um atributo é relacionado a seu objeto. Sérgio é um bom músico.

[Pessoa: Sérgio]  $\rightarrow$  (Atrb)  $\rightarrow$  [Bom]  $\rightarrow$  (Base)  $\rightarrow$  [Type: Músico].

Paráfrase: Sérgio tem o atributo bom, como músico.

Característica < Tipo  $\cap$  Propriedade; Crct(Entidade, Entidade). Um tipo cuja instâncias são propriedades de entidades. Características são usualmente expressadas por substantivos, como: formato, cor, tamanho, e peso.

Exemplo: A cor da rosa é vermelha.

[Rosa: #]  $\rightarrow$  (Crct)  $\rightarrow$  [Cor: Vermelha].

Paráfrase: A rosa tem uma cor vermelha como característica.

Criança < SerHumano  $\cap$  Correlativo; Crnc(SerHumano, SerHumano). Um ser humano que é uma criança de algum ser humano. Uma criança é correlativa a parente.

Exemplo: Lilia é mãe de Kátia.

[Mãe: Lilia]  $\rightarrow$  (Crnc)  $\rightarrow$  [Pessoa: Kátia].

Paráfrase: Lilia tem Kátia como criança.

Comparação < Objeto  $\cap$  Correlativo; Comp(Atributo, Objeto). Um objeto que serve como um padrão de comparação para algum atributo. Um objeto sendo comparado é correlativo a outro objeto.

Exemplo: João é maior que Maria.

[Pessoa: João]  $\rightarrow$  (Atrb)  $\rightarrow$  [Alto]  $\rightarrow$  (Comp)  $\rightarrow$  [Comparado: Maria].

Paráfrase: João é maior em comparação a Maria.

Função; Ter(Entidade, Entidade). Ter é um tipo de relação geral usada para definir todas as funções.

Maneira < Entidade  $\cap$  Propriedade; Manr(Processo, Entidade). Uma entidade que é uma propriedade de algum processo. Maneiras são usualmente expressadas por advérbios, como: rapidamente.

Exemplo: A ambulância chegou rapidamente.

[Ambulância: #]  $\leftarrow$  (Agnt)  $\leftarrow$  [Chegar]  $\rightarrow$  (Manr)  $\rightarrow$  [Rápida].

Paráfrase: A chegada da ambulância teve uma maneira rápida.

Medida < Quantidade  $\cap$  Medida; Med(Atributo, Quantidade). Uma quantidade usada como medida de algum atributo. O primeiro argumento da relação Med, é um atributo, que é usualmente expressado por um adjetivo, como longo, alto, pesado, velho, rápido, ou quente. A

relação Med liga um atributo a uma medida, e a relação Qt, liga a característica correspondente para a mesma medida.

Exemplo: O esqui é 167cm longo.

[Esqui: #]→(Atrb)→[Longo]→(Med)→[Medida: <167, Cm>].

Paráfrase: O esqui tem um atributo longo, cuja medida é 167 cm.

Parte<Objeto∩Componente; Part(Objeto, Objeto). Um objeto que é um componente de algum objeto. Diferente de um atributo, uma parte é capaz de existir independentemente.

Exemplo: Um dedo é parte da mão.

[Mão]→(Part)→[Dedo].

Paráfrase: Uma mão tem um dedo como parte.

Porque<Situação∩Correlativo; Pque(Situação, Situação). Uma situação na função de causar outra situação. Uma causa é correlativa a um efeito.

Exemplo: Você está molhado porque está chovendo.

[Situação: [Pessoa: #você] ← (Atrb) ← [Molhado]] → (Pque) → [Situação: [chove]].

Paráfrase: Você estando molhado, tem chuva como causa.

Possessão<Entidade∩Correlativo; Poss(Animado, Entidade). Uma entidade de propriedade de algum ser animado. Uma possessão é correlativa ao possessor.

Exemplo: O relógio de Guga parou.

[Pessoa: Guga]→(Poss)→[Relógio-Pulso]←(Tema)←[Parar].

Paráfrase: Guga tem como possessão, um relógio que parou.

Quantia<Quantidade∩Medida; Qt(Característica, Quantidade). Uma quantidade usada como uma medida de alguma característica. O primeiro argumento da relação Quantia é uma característica, que é usualmente expressada por um nome, como tamanho, peso, altura, idade, velocidade, ou temperatura.

Exemplo: O esqui tem 167cm de comprimento.

[Esqui: #]→(Crct)→[Tamanho]→(Qt)→[Medida: <167, cm>].

Pela contração da medida, a relação MED pode ser contraída para formar o conceito: [Esqui: #]→(Crct)→ [Tamanho: @167cm].

Paráfrase: O esqui tem uma característica de tamanho cuja quantia é 167 cm.

Sucessor<Ocorrência∩Correlativo; Sucs(Ocorrência, Ocorrência). Uma ocorrência que ocorre depois de alguma outra ocorrência. Um sucessor é correlativo ao predecessor.

Exemplo: Depois que Brenda foi à escola, ela foi passear no parque.

[Situação: [Pessoa: Brenda= \*X] ← (Agnt) ← [Ir] → (Loc) → [Escola: #]] - (Sucs) →

[Situação: [Pessoa: \*X] ← (Agnt) ← [Passear] → (Loc) → [Parque]].

Paráfrase: A situação de Brenda ir para a escola, teve como sucessor, a situação dela passear no parque.

## Anexo F - Adjetivos Modificando Substantivos

Nos livros de lógica elementar, adjetivos e substantivos são usualmente traduzidos para predicados mônadas. Para muitas frases comuns, a tradução produz uma fórmula aceitável em lógica.

Um garoto feliz  $\Rightarrow (\exists x)(\text{feliz}(x) \wedge \text{garoto}(x))$ .

Um cão despenteado  $\Rightarrow (\exists y)(\text{despenteado}(y) \wedge \text{dog}(y))$ .

Uma árvore verde  $\Rightarrow (\exists z)(\text{verde}(z) \wedge \text{árvore}(z))$ .

Essas relações implicam corretamente que: existe algum  $x$  que é feliz e é um garoto, algum  $y$  que é despenteado e é um cão, e algum  $z$  que é verde e é uma árvore. Mas quando o método é aplicado para todos os adjetivos e substantivos, isto causa sérias dificuldades. Um programa de questão/resposta usado para traduzir as duas sentenças abaixo, para lógica:

Samuel é um bom músico  $\Rightarrow \text{bom}(\text{Samuel}) \wedge \text{músico}(\text{Samuel})$ .

Samuel é mau cozinheiro  $\Rightarrow \text{mau}(\text{Samuel}) \wedge \text{cozinheiro}(\text{Samuel})$ .

Usando essas traduções, o programa pode responder "sim" as seguintes questões:

É Samuel um bom músico?

É Samuel um bom cozinheiro?

É Samuel um bom mau músico cozinheiro?

O problema é causado pela maneira que os adjetivos, bom e mau, modificam os substantivos. Diferente do adjetivo feliz, que aplica diretamente a pessoa, os adjetivos bom e mal aplicam a uma função particular que a pessoa representa. Samuel não está sendo considerado bom ou mau como ser humano, mas somente nas funções de músico ou cozinheiro.

Quando o adjetivo é aplicado a um tipo fenomenal como Garoto, Cão, ou Árvore, existe apenas uma entidade  $x$  que pode ser caracterizada. Mas quando ele é aplicado a um tipo função como Músico, Cozinheiro ou Moradia, ele poderia modificar a descrição de  $x$ , ou seja, a função que  $x$  representa em relação a alguma entidade  $y$ . Adjetivos como feliz, despenteado, e verde, descrevem alguma entidade  $x$  independentemente de qualquer função que  $x$  possa representar. Um cão despenteado e um animal doméstico despenteado são ambos

despenteados da mesma maneira e a função de animal doméstico é independente da despenteado. Esses adjetivos podem ser aplicados para tipos de função como nas frases: "músico feliz", "moradia verde". Nessas combinações, o adjetivo descreve a entidade base. Assim, o adjetivo pode ser representado pelo predicado monada:

Um músico feliz  $\Rightarrow (\exists x)(\text{feliz}(x) \wedge \text{músico}(x))$ .

Uma moradia verde  $\Rightarrow (\exists y)(\text{verde}(y) \wedge \text{moradia}(y))$ .

Os adjetivos bom e mau, porém, modificam a função. Um bom músico e um bom cozinheiro são considerados bons somente em relação à música e cozinha. Outros exemplos incluem "físico nuclear", "primeiro ministro" e "suposto ladrão". Um físico feliz é uma pessoa feliz, mas um físico nuclear não é uma pessoa nuclear. Um suposto ladrão não é uma suposta pessoa.

Uma simples maneira de representar um adjetivo modificando um substantivo é inventar predicados especiais como  $\text{bomMúsico}(x)$  ou  $\text{nuclearFísico}(y)$ . Esse método requer um novo predicado para cada combinação de adjetivo e substantivo. Uma abordagem mais geral poderia ser baseada na técnica de Richard Montague que trata modificadores como funções que convertem um predicado em outro. O adjetivo bom, por exemplo, poderia corresponder a função que mapeia o predicado  $\text{músico}(x)$  para o predicado que é equivalente a  $\text{bomMúsico}(x)$ :

Um bom músico  $\Rightarrow (\exists x)\text{bom}(\text{músico})(x)$ .

Essa fórmula diz que a função bom, quando aplicada ao predicado músico, gera um novo predicado  $\text{bom}(\text{músico})$  que é, então, aplicado à entidade  $x$ . Mas essa abordagem fica mais complicada com a sentença "Ivan é uma escolha fraca para atacante, mas ele é uma boa escolha para goleiro". Escolha é uma função que implica que Ivan poderia ser escolhido por uma outra função, Atacante ou Goleiro, que seria bom ou mal. Generalizando, a representação acima, poderia produzir alguma coisa, como:

Uma boa escolha para goleiro  $\Rightarrow (\exists x)\text{bom}(\text{escolha})(\text{goleiro})(x)$ .

Essa fórmula diz que bom é uma função, que quando é aplicada para "escolha" produz outra função, que quando aplicada a goleiro, gera um predicado que é aplicado a  $x$ .

A representação dos adjetivos como funções é apenas o primeiro passo. Isto mostra que um adjetivo modifica um substantivo, mas não mostra como isto afeta o significado do substantivo. Um tipo função como músico ou cozinheiro, descreve a função de algum ser humano relativamente à música ou a cozinha. Nos termos de Whitehead, ele expressa a preensão que consiste de três fatores: (1) a entidade "preensora", nomeia a pessoa ou coisa que o substantivo refere; (2) a entidade "acolhida", músico ou cozinheiro; e (3) a forma subjetiva ou intenção que determina como a primeira entidade representa a música ou o feito da comida. Um adjetivo que modifica um substantivo poderia ser aplicado a qualquer dos três fatores:

1. Entidade preensora. Nas frases: "músico feliz" e "cozinheiro elegante", o adjetivo é aplicado diretamente à entidade referenciada pelo substantivo. A relação desse indivíduo à música ou cozinha não sofre efeito do adjetivo.
2. Entidade acolhida. Na frase "físico nuclear", o adjetivo descreve a entidade acolhida: o ramo da física.
3. Intenção. Na frase "bom músico", o adjetivo modifica a forma subjetiva ou intenção que relaciona a entidade acolhida à música.

Nesses exemplos, o substantivo implica a função, mas algumas vezes o modificador determina a função. O monstro de Loch Ness, por exemplo, mora em Loch Ness, mas o cozinheiro monstro come bolinhos. O desafio de achar a relação correta pode fazer da linguagem natural um entendimento difícil para pessoas e computadores.

---

## Anexo G – Teste Tem

Como um teste para distinguir a entidade preensora ou acolhida, aplica-se o padrão "X tem Y" para o par de palavras que a descrevem. Se o padrão soa normal ou natural, então X é a entidade preensora e Y é a entidade acolhida. Por exemplo, alguém pode dizer "O carro tem um motor" ou "O carro tem uma cor", mas não "O motor tem um carro" ou "A cor tem um carro". Se a entidade acolhida é abstrata, o teste "tem" pode soar mais natural com a frase "tem conhecimento de". Um físico tem conhecimento de física ou um músico tem conhecimento de música. Para alguns pares, o tem-teste é unidirecional: o todo tem partes, mas as partes não têm o correspondente todo. Para correlativos, porém, o membro do par pode ser a entidade preensora ou acolhida: uma mãe tem um filho e um filho tem uma mãe.

Se as mesmas entidades são descritas com palavras diferentes, o teste "tem" pode achar relações implícitas diferentes. Se Samuel contrata a sua filha Sueli para trabalhar em seu negócio, a relação implícita depende de onde elas são descritas como pai/filha, empregador/empregado ou colega de trabalho.

Existem relações substantivo/substantivo que o teste "tem" não é aplicado. Terra e céu, por exemplo, são fortemente associados, mas ambos os padrões: "A terra tem o céu", e "O céu tem a terra", não soa natural.

## Anexo H - Advertências para Definições de Bases de Conhecimento

O seguinte silogismo ilustra um erro entre nomes e tipos:

Dumbo é um elefante.

Elefante é uma espécie.

Assim, Dumbo é uma espécie.

Na primeira premissa, o termo um elefante refere-se a um indivíduo particular chamado Dumbo. Na segunda premissa, a palavra elefante refere-se a toda a espécie ou tipo.

Outro silogismo que apresenta falhas poderia ser:

*Todos os salários são distribuídos em um cheque de pagamento.*

*Tony e Suely recebem o mesmo salário.*

*Assim, Tony e Suely recebem o mesmo cheque de pagamento.*

Uma sentença afirmando que duas pessoas ganham o mesmo salário, pode significar que os valores são os mesmos para ambos, mas cada um deles recebe um cheque de pagamento separado. Esse tipo de observação, em que existe falha na distinção das definições, é comum em erros de programação.

Para representar sentenças como "Tony gosta de Suely", podemos utilizar a fórmula  $\text{gosta}(\text{Tony}, \text{Suely})$ . Geralmente, os estudantes generalizam esta técnica para representar a sentença: Gatos gostam de peixe, pela fórmula  $\text{gostar}(\text{gatos}, \text{peixe})$ . O erro dessa generalização é causado pela diferença entre nomes próprios que denotam indivíduos particulares e substantivos comuns que denotam tipos. Os substantivos comuns gato e peixe devem ser representados pelo tipo rotulado em lógica tipificada ou predicado mônada em lógica não tipificada. Em lógica tipificada, a sentença "Gatos gostam de peixe" poderia ser:

$$(\forall x:\text{Gato})(\forall y:\text{Peixe})\text{gosta}(x,y).$$

Essa fórmula pode ser lida: Para todo gato  $x$  e peixe  $y$ ,  $x$  gosta de  $y$ .

A regra que nomes próprios mapeiam para constantes enquanto substantivos comuns mapeiam para tipos ou predicados, funciona corretamente para a maioria das aplicações.

Mas isto impõe algumas restrições: na lógica de primeira ordem, as variáveis podem referir somente a entidades individuais; elas não podem representar tipos ou predicados. Uma maneira de resolver essa restrição seria introduzir Tipo, como um tipo de segunda ordem. Assim, qualquer variável do tipo Tipo, pode se referir a tipos como sendo seus valores. Para relacionar uma entidade  $x$  ao seu tipo  $t$ , um predicado  $\text{tipo}(x,t)$  poderia ser usado. Assim, a sentença sobre gatos e peixe poderia ser

$$(\forall x)(\forall y)((\text{tipo}(x,\text{Gato}) \wedge \text{tipo}(y,\text{Peixe})) \supset \text{gosta}(x,y)).$$

Essa fórmula pode ser lida como: para todo  $x$  e  $y$ , se  $x$  é um tipo de gato, e  $y$  é um tipo de peixe, então  $x$  gosta de  $y$ .

A distinção entre nomes e tipos provê uma maneira de resolver a charada sobre Dumbo o elefante. Na sentença "Dumbo é um elefante", a ausência do artigo que precede a palavra Dumbo sugere um nome próprio que é colocado no campo referente de um conceito. Como o tipo Dumbo não é especificado, o tipo padrão  $\tau$  que é o topo do tipo hierárquico pode ser usado. O resultado é o conceito  $[\tau: \text{Dumbo}]$ . O artigo indefinido na frase um elefante, indica algum indivíduo do tipo Elefante que deveria ser representado pelo conceito  $[\text{Elefante}]$ . A palavra corresponde a uma ligação de co-referência entre os dois conceitos.

$$[\tau: \text{Dumbo}] - \text{-----} - [\text{Elefante}].$$

A ligação de co-referência que conecta os dois indivíduos mostra que os dois se referem ao mesmo indivíduo. O primeiro conceito diz que existe alguma coisa chamada Dumbo, o segundo diz que existe um elefante, e a linha pontilhada indica que os dois indivíduos são o mesmo. Em cálculo de predicado, co-referência é representado pelo sinal de igual:

$$(\exists x: \text{Elefante})x = \text{Dumbo}.$$

Sem um artigo, o substantivo comum elefante não se refere a um indivíduo singular. Na sentença: "Elefante é uma espécie", ele foi usado como um nome de tipo:

$$[\text{Tipo: Elefante}] \text{-----} [\text{Espécies}].$$

Esse grafo, pode ser lido: "O tipo elefante é um espécie", que é um relato de ordem alta sobre tipos. Os tipos: Tipo e Espécie são tipos de segunda ordem, suas instâncias são tipos de primeira ordem como Elefante. As instâncias do tipo Elefante são indivíduos normais como Dumbo. O grafo conceitual mapeia para a seguinte fórmula:

$$(\exists x: \text{Espécies})(\text{tipo}(\text{Elefante}) \wedge x = \text{Elefante}).$$

Esses grafos conceituais e fórmulas derivadas dele capturam o sentido das sentenças sem permitir a inferência incorreta de que Dumbo é uma espécie.

Essa distinção entre nomes e tipos aparece em muitas áreas de projeto de bases de dados e bases de conhecimento. Um exemplo importante é a representação de cores, formas e tamanhos. Existem três grafos conceituais (Figura A.3) para representar a frase "Uma bola vermelha". Na primeira, a relação conceptual mônada (Vermelha) é anexada diretamente ao conceito [Bola]. Isto corresponde a seguinte fórmula:

$$(\exists x: \text{Bola})\text{vermelha}(x).$$

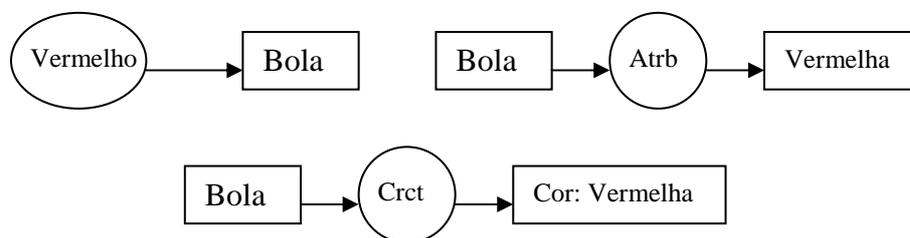


Figura A.3. Três possíveis formas de representação de "Uma bola vermelha".

Essa fórmula diz que: existe uma bola  $x$  a qual é vermelha. Apesar dessa representação ser simples, ela é mal formada para projetistas de banco de dados e linguagem semântica natural.

O segundo grafo mostra uma maneira mais sistemática de representação das sentenças em linguagem natural. Todas as palavras contidas — substantivos, adjetivos e advérbios — são representados por conceitos separados. O nome bola mapea para um conceito [Bola], o adjetivo vermelho mapea para o conceito [Vermelho] e a relação entre eles mapea para a relação atributo (Atrb). Na tradução para lógica classificada, ambos os nós conceitos mapeam para um quantificador universal e a relação Atrb passa a ser um predicado díade:

$$(\exists x: \text{Bola})(\exists y: \text{Vermelho}) \text{atr}(x,y).$$

Essa fórmula diz que existe uma bola  $x$  e uma instância de vermelho  $y$  onde o predicado atrb relaciona  $x$  a  $y$ .

Para projetistas de base de dados, o terceiro grafo é mais conveniente. A relação característica (Crct), liga o conceito de uma bola ao conceito [Cor: Vermelha] cujo rótulo

de tipo é um tipo de segunda ordem Cor, e cujo referente é um tipo de primeira ordem Vermelho. Este grafo mapea para a seguinte fórmula:

$$(\exists x: Bola)(cor(Vermelha) \wedge crct(x, Vermelha)).$$

---

## GLOSSÁRIO

ANS - American National Standard.

API - Application Program Interface. Método prescrito de requisição do sistema operacional ou outro programa qualquer.

Applet - Programa de computador que é executado dentro de páginas Web.

BBS - Bulletin Board Systems. Sistema de publicação de informações, onde os participantes compartilham uma área informativa na Internet/Intranet.

CGIF - Conceptual Graph Interchange Format.

DC - Dependência Conceitual.

Documento Web - Documento da Internet formatado para utilização com o protocolo http.

FRL - Frame Representation Language.

GC - Grafos Conceituais.

GML - Graph Modeling Language. Formato de armazenagem gráfica.

HTML - Hypertext Mark-up Language.

HTTP - Hypertext Transport Protocol.

IA - Inteligência Artificial.

IR - Information Retrieval.

IRC - Internet Relay Chat.

Java - Linguagem de programação orientada a objeto.

KIF - Knowledge Interchange Format.

KRL - Knowledge Representation Language.

Ligação - o mesmo que ponteiro.

Link - Endereço Web carregado no caso de ser acionado.

ListServ - Lista de discussões, onde os participantes trocam e-mails sobre um determinado tema.

Metadata - Linguagens formuladas com tags "<" e ">". Exemplo: HTML e XML.

MIT - Massachusetts Institute Technology.

Módulo - Parte de um sistema que executa uma função.

MOO - Multi User Object Oriented.

MUD - Multi User Domain.

MUSH - Multi User Shared.

Navegação - Carregar páginas Web utilizando ferramentas Web.

Nó - é uma subdivisão de uma estrutura hierárquica.

OML - Ontology Markup Language.

Página Web - Documento da Internet formatado para utilização com o protocolo http.

Ponteiro - o mesmo que link.

RDF - Resource Description Framework.

Robô Web - Computadores que navegam automaticamente na Internet armazenando informações de seu interesse e disponibilizando a seus usuários.

VEDA - em tibetano significa Conhecimento.

Web - Ambiente da Internet que utiliza o protocolo http como meio de transmissão.

WOO - Web Object Oriented.

WWW - World Wide Web. Endereço de acesso as páginas Web.

XGMML - Extensible Graph Mark-up and Modeling Language.

XML - Extensible Mark-up Language.

## Bibliografia

- [AIT.89] Aït-Kaci, H. *Efficient implementation of lattice operations*. ACM Trans. On Programming Languages and Systems 11, 115-146, 1989.
- [AKO.98] Akoulchina, I. *SAGE Agent for the SATELIT Web-based system*. Université Paris. França, 1998.
- [ATI.98] Atieh, S. **How to Get a College Degree Via the Internet: The Complete Guide to Getting Your Undergraduate or Graduate Degree from the Comfort of Your Home**. Prima Publishing. 1998.
- [BAC.89] Bacchus, F. *A modest, but semantically well founded, inheritance reasoner*. In Proc. IJCAI, 1989.
- [BER.98] Berger, P. **Internet for Active Learners**. Amer Library Assn. 1998.
- [BLA.00] Black, B. Concerto: *Conceptual indexing, querying and retrieval of digital documents*. <http://concerto.ccl.umist.ac.uk/>. 2000.
- [BRA.79] Brachman, R. *On the epistemological status of semantic networks*. In Association Networks. Academic Press. Orlando, Florida, 1979.
- [BRO.00] Brown, S. **500 Tips on Group Learning**. Kogan Page Ltd. 2000.
- [BRO.93] Brockenbrough, A. e Hoffman, R. *Varied Levels of Support for Constructive Activity in Hypermedia-Based Learning Environments*. Department of Educational Technology, San Diego State University, San Diego, California, USA, 1993.
- [BRU.93] Bruffe, K. *A Short Course in Writing: Composition, Collaborative Writing, and Constructive Reading*. Addison-Wesley Pub Co, 1993.
- [BUR.84] Burger, H. *The Wordtree*. Wordtree Publ., 10876 Bradshaw, Overland Park, Kansas, 1984.
- [BUR.94] Burrow, A. *Meta Tool Support for a GUI for Conceptual Structures*. Thesis. University of Adelayde., 1994.
- [CG] *Conceptual Graphs*, <http://concept.cs.uah.edu/CG/cg-standard.html>
- [CHE.99] Chevallet, e Chiramella. *Our experience in logical IR modeling*. Glasgow workshop on logic, 1999.

- [CHI.96] Chiaramella, Y. *Browsing and Quering: two complementary approaches for Multimedia Information Retrieval*. CLIPS Laboratory, Grenoble, France, 1996.
- [COL.00] Collins, M. *Changing Student Behaviors: A Positive Approach*. 2000.
- [DEL.00] Delugach, H. *CharGer - A Conceptual Graph Editor*.  
<http://www.cs.uah.edu/~delugach/CharGer/>, 2000.
- [DOB.98] Dobrev, P. e Toutanova, K. *CGWorld - A Web Based Workbench for Conceptual Graphs Management and Applications*. ProSysyt. Bulgaria Ltd., Sofia Bulgaria, 1998.
- [dpANS] *Conceptual Graph Standard Information Technology (IT)*. Conceptual Graph draft proposed American National Standard. NCITS.T2/98-003, 1998.
- [DRA.00] Draves, W. *Teaching Online*, Lern Books, Winsconsin, 2000.
- [EKL.98] Eklund, P., et ali. *WebKB-GE - A Visual Editor for Canonical Conceptual Graphs*. ICSS98, 1998.
- [EKL.97] Eklund, P. e Martin P. *The WebKB set of tools*, Proceedings of 5<sup>th</sup> International Conference on Conceptual Structures (ICCS'97), 1997.
- [ELL.95] Ellis, G et all. *Object-Oriented conceptual graph*, 1995.
- [FEN.98a] Fensel, D. Decker, S. Erdmann, M., e Studer, R. *Ontobroker: How to make the WWW Intelligent*, research report, Institute AIFB.
- [FEN.98b] Fensel, D. Decker, S. Erdmann, M., e Studer, R. *Ontobroker: The Very High Idea*. In Proceedings of the 11th International Flairs Conference (FLAIRS-98) , Sanibal Island, Florida, May 1998.
- [FRE.97] Freg, G. *The Frege Reader* (BlackWell Readers). BlackWell Pub. 1997.
- [GAY.99] Gay, D. *Ampl : A Modeling Language for Mathematical Programming*. Scientific Press. 1999.
- [GIL.95] Gilly, M. *Approaches Socio-Construtives du Développement Cognitif de l'enfant d'âge Scolaire*, In Gaonach, D. & Golder (Eds.) Manuel de Psychologie pour l'Enseignement, Paris: Hachette, Éducation 1995.
- [Gramática] *Categorias Gramaticais*. Gramática Universal. <http://www.priberam.pt/>
- [GUH.90] Guha, M. e Lenat, D. *Cyc: a midterm report*. AI Magazine, 11 (3), 32-59, 1990.
- [HAE.95] Haemmerlé, O. *CoGITO: une plate-forme de développement de logiciels sur les graphes conceptuels*. PhD Thesis, Montpellier II University, France, 1995.
- [HAR.88] Harel, D. *On visual formalisms*. Communications of the ACM, 31, 514-530, 1988.
- [HAR.93] Harasim, L. *Global Networks : Computers and International Communication*. MIT Press. 1993.

- [HAY.79] Hayes, P. *On semantic nets, frames and associations*, In Proc. IJCAI, 1979.
- [HEN.79] Hendix, G. *Encoding knowledge in partitioned networks*. In Associative Networks. Academic Press, Orlando, Flórida, 1979.
- [HIL.95] Hiltz, S. **Learning Networks**. MIT Press. 1995.
- [ISO/IEC] 14977: 1996E. *Extended BNF*, 1996.
- [JON.93] Jonassen, D. A & Mayes, T. & Macaleese, R. *Manifesto for a Constructivist Approach to uses of Technology in Higher Education*, In (eds) Duffy, T& Jonassen, D. Design Environments for Constructive Learning, pp. 231-249, Springer Verlag, Berlin, 1993.
- [JON.96] Jonassen, D. *Computers as mindtools for schools: engaging critical thinking*. Second Edition. Prentice Hall. 1996.
- [JON.99] Jonassen, D. **Task Analysis Methods for Instructional Design**. Lawrence Erlbaum Assoc. 1999.
- [JSDT] *Java Shared Data Toolkit*. <http://www.sun.com/software/jsdt/>
- [KEM.86] Kemp, A. *A memoir on the theory of mathematical form*. Philosophical Transactions of the Royal Society of London, 177, pp. 1-70, 1886.
- [KIE.86] Kiesler, S. **Computing and Change on Campus**. 1986.
- [KWO.98] Kwon, W. **Distributed Computer Control Systems**. Pergamon Pr. 1998.
- [LAV.88] Lave, J. *Cognition in Practice : Mind, Mathematics and Culture in Everyday Life*. Cambridge Univ, 1988.
- [LEA.00] Lea, M. **Student Writing in Higher Education : New Contexts**. Open Univ Pr. 2000.
- [LEH.92] Lehmann, F. *Semantic Networks in Artificial Intelligence*. Pergamon Press Ltd, Oxford, New York, Soul, Tokyo, 1992.
- [LEI.89] Leibniz, G. *A calculus of consequences*. In philosophical Essays. Hackett Publ. Co., Indianapolis, Indiana, 1989.
- [LEN.95] Lenat, D. *CYC: A large-scale investment in knowledge infrastructure*. Communications of the ACM.
- [LIN.73] Lindsay, R. *In defense of ad hoc systems, In Computer Models of Thought and Language*, (R. Shank e K.M. Colby, Eds.), W. H. Freeman, San Francisco, 1973.
- [LYT.92] Lytinen, S. *Conceptual Dependency and Its Descendants*. The University of Michigan, USA, 1992.

- [MAR.96] Martin, P. *Exploitation de graphes conceptuels et de documents structurés et hypertextes pour l'acquisition de connaissances et la recherche d'informations*. Ph.D thesis, University of Nice Sophia Antipolis, France, 1996.
- [MCG.98] McGuire, B. **Books on Wheels : Cooperative Learning Through Thematic Units**. 1998.
- [MIL.93] Miller, G. et ali. *Introduction to WordNet: An-Online Lexical Database*. Princeton University. Princeton, NJ, 1993.
- [MIL.98] Miller, G. et ali. *WordNet - a Lexical Database for English*. Princeton University. Princeton, NJ, 1998.
- [MIN.75] Minsky, M. *The Psychology of Computer Vision*. P.H. Winston, Ed. McGraw Hill, New York, 1975.
- [MIN.81] Minsky, M. *A framework for representing knowledge, In Mind Design*. J. Haugeland, Ed. Mit Press, Cambridge, Mass., 1981.
- [NEL.98] Nelson, M. **Children and Social Studies : Creative Teaching in the Elementary Classroom** Wadsworth Pub Co. 1998.
- [NOTIO] Notio Project, <http://backtrack.math.uwaterloo.ca/CG/projects/notio/>
- [OML] *Ontology Markup Language* <http://www.oasis-open.org/cover/om19808.html>
- [PEA.87] Pea, R. **Mirrors of Minds : Patterns of Experience in Educational Computing** (Cognition and Computing, Vol 1)
- [PEA.88] Pearl, J. *Probabilistic Reasoning in Intelligent Systems*. San Mateo, California. 1988.
- [PEI.76] Peirce, C. *The new elements of mathematics*. Humanities Press, N.J., 1976.
- [PER.95] Perkins, D. **Smart Schools : Better Thinking and Learning for Every Child**. Free Press. 1995.
- [PFA.98] Pfaffenberger, B. **Internet Training CD Rom Tutorial**. Charles River Media. 1998.
- [PHI.98] Philips, V. **The Best Distance Learning Graduate Schools: Earning Your Degree Without Leaving Home (Serial)**. Princeton Review
- [QUI.85] Quillian, M. *Word Concepts. Behavioral Science*. Los Altos, California, 1985.
- [RDF] *Resource Description Framework* <http://www.w3.org/TR/1999/PR-rdf-syntax-19990105/>
- [RIC.56] Richens, R. *Mechanical Translation*. Report on research of the Cambridge Language Research Unit. Washington University, St. Louis, 1956.

- [RIE.92] Riel, M. **Environmental Structures : Semantic and Experimental Components** (Publications in Environmental Perception, Report 8). American Geographical Society. 2000.
- [RUM.73] Rumelhart, D. Norman, D. *Active semantic networks as a model of human memory*. In Proc. IJCAI, 1973.
- [SCD.97] Schiano, D. *The First Noble Truth of CyberSpace: People are People Even in the MUD*. Stanford University. 1997.
- [SCE.95] Schegloff, E.. *Research on Language and Social Interaction: Special Issue on Co-Construction*, Discourse as an interactional achievement III: The omnirelevance of action 28, 185-213. 1995.
- [SCH.75] Schank, R. *Conceptual Information Processing*. North Holland, Amsterdam, 1975.
- [SCH.89] Schank, R. *A Thematic Hierarchy for Indexing Stories in Social Domains*. Technical Report. Northeastern University. 1989.
- [SCH.90] Schank, R. *A Content Theory of Memory Indexing*. Technical Report. Northeastern University. 1990.
- [SCO.99] Scott, P. **The Globalization of Higher Education**. Open Univ Pr. 1999.
- [SEA.83] Searle, John. *Intentionality*. Cambridge University Pr. 1983.
- [SIM.98] Simon, C. **Bill Gates : Helping People Use Computers (Community Builders)**. Children's Press. 1998.
- [SKI.58] Skinner, B. *Teaching Machines*, Science 129, pp 969-977, 1958.
- [SKI.76] Skinner, B. *About Behaviorism*. Random House (Paper). 1976.
- [SOW.00] Sowa F. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*, Brooks Cole Publishing Co., Pacific Grove, CA, 1994.
- [SOW.84] Sowa F. *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley publishing company, 1984.
- [SYL.09] Sylvester, J. *Chemistry and Algebra, Nature*. pp. 103-104, Cambridge University Press, 1909.
- [VAC.94] Vacca, R. **Whole Language in Middle and Secondary Classrooms**. Addison-Wesley Pub Co. 1994.
- [VGJ] Visualizing Graphs with Java,  
[http://www.eng.auburn.edu/department/cse/research/graph\\_drawing/graph\\_drawing.html](http://www.eng.auburn.edu/department/cse/research/graph_drawing/graph_drawing.html)
- [WebContract] Université Montpellier. <http://www.lirmm.fr/>

[WHI.27] Whitehead, A. e Russel, B. *Principia Mathematica*, Cambridge Univ. Pres, Cambridge, 1927.

[WhiteBoard] <http://www.groupboard.com/> ,  
<http://osprey7.npac.syr.edu:3768/cps606spring98-docs/co98par/jdraw.html>

[WIL.68] Wilkins, J. *Essay Towards a Real Character and a Philosophical Language*. The Royal Society, Gellibrand by Martin, London, 1968.

[WOO.97] Woods, W. *Conceptual Indexing: A Better Way to Organize Knowledge*. Sun Microsystems, 1997.

[XGMML] (*eXtensible Graph Markup and Modeling Language*)  
<http://www.cs.rpi.edu/~puninj/XGMML/draft-xgmml.html>

[XML] *Extensible Markup Language* <http://www.w3.org/XML/>