



Universidade de Brasília

Érica de Oliveira Carvalho

Uma proposta de Interdisciplinaridade entre Arquitetura
da Informação e Ciência da Computação: Linguagem
“SOWL” para as ontologias da Web utilizando o
formalismo dos grafos conceituais

Brasília

2013

Érica de Oliveira Carvalho

Uma proposta de Interdisciplinaridade entre Arquitetura
da Informação e Ciência da Computação: Linguagem
“SOWL” para as ontologias da Web utilizando o
formalismo dos grafos conceituais

Tese apresentada à Faculdade de Ciência
da Informação da Universidade de
Brasília como requisito parcial para a
obtenção do título de Doutor.

Orientador: Prof. Dr. Cláudio Gottschalg Duque

Brasília

2013

CARVALHO, ÉRICA DE OLIVEIRA.

Uma proposta de Interdisciplinaridade entre Arquitetura da Informação e Ciência da Computação: Linguagem “SOWL” para as ontologias da Web utilizando o formalismo dos grafos conceituais/ Érica de Oliveira Carvalho. – 2013.

248p. : il. (algumas color.) ; 30 cm.

Tese (doutorado) – Universidade de Brasília, Faculdade de Ciência da Informação, 2013.

Inclui bibliografia.

Orientação: Prof. Dr. Cláudio Gottschalg Duque

Tese (doutorado) – Universidade de Brasília, Faculdade de Ciência da Informação, 2013.

I. Arquitetura da Informação. II. Gottschalg Duque, Cláudio. III. Universidade de Brasília. IV. Faculdade de Ciência da Informação. V. Título



FOLHA DE APROVAÇÃO

Título: "Interdisciplinaridade entre Arquitetura da Informação e Ciência da Computação para uma proposta informacional: uma nova linguagem para as ontologias da Web utilizando o formalismo dos grafos conceituais".

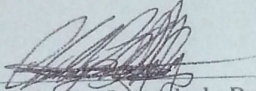
Autor (a): Erica De Oliveira Carvalho

Área de concentração: Transferência da Informação

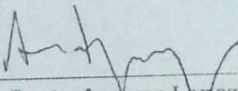
Linha de pesquisa: Arquitetura da Informação

Tese submetida à Comissão Examinadora designada pelo Colegiado do Programa de Pós-graduação em Ciência da Informação da Faculdade em Ciência da Informação da Universidade de Brasília como requisito parcial para obtenção do título de **Doutor** em Ciência da Informação.

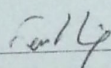
Tese aprovada em: 16 de dezembro de 2013.



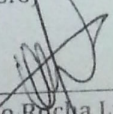
Prof. Dr. Cláudio Goitschalg Duque
Presidente (UnB/PPGCINF)



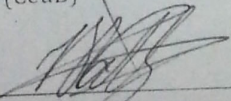
Prof. Dr. Andre Porto Ancona Lopez
Membro Interno (UnB/PPGCINF)



Prof.ª. Dr.ª. Fernanda Lima
Membro Externo (CIC)



Prof.ª. Dr.ª. Mauricio Rocha Lyra
Membro Externo (CeUB)



Prof.ª. Dr.ª. Vinicius Sebba Patto
Membro Externo (UFG)

Prof.ª. Dr.ª. Ivette Kafure Muñoz
Suplente (UnB/PPGCINF)

Aos meus pais, que sempre foram exemplo de valores e meu porto seguro. Sem eles a finalização desta e outras jornadas não teria sido possível.

Agradecimentos

Agradecer é uma forma de recordar e homenagear todos aqueles que foram essenciais para o êxito desta empresa. Mais do que isso, é o reconhecimento de que a finalização de uma tese de doutorado é uma consequência de um trabalho em conjunto. Por isso, agradeço:

À CAPES, pela ajuda financeira durante todo o percurso da tese, inclusive a bolsa para o estágio doutoral em Toulouse, França.

Ao Programa de Pós Graduação em Ciência da Informação da Universidade de Brasília, pela oportunidade de realizar etapa tão importante para minha carreira acadêmica. Também agradeço a este programa por ampliar sobremaneira a forma de compreender a informação, fazendo perceber que seu escopo vai muito além da Ciência da Computação, e também contribuindo para o anelo sincero de contribuir de forma mais acessível para a organização da informação. Além disso, o estudo da Arquitetura da Informação me fez compreender muito melhor a importância de “dar forma a” para a organização e estrutura de qualquer sistema. Tudo que aprendi no tempo da FCI me permitiu uma libertação do pensamento técnico da Informática. Ampliou de forma gigantesca a visão do problema da informação, que é por natureza multidisciplinar e não poderia ficar preso a uma só ciência. Agradeço a oportunidade deste doutorado, pois fez com que eu valorizasse ainda mais minha formação acadêmica, pois esta possibilitou, de alguma forma, o contato com a Ciência da Informação.

Aos professores e colegas da FCI que tanto contribuíram para ampliar a compreensão dos conceitos e tornar as lutas diárias mais aprazíveis.

Às secretárias do programa, Martha e Jucilene, por sua disponibilidade e alegria sempre presentes; favorecendo muito a formação de um ambiente agradável e eficiente na pós-graduação.

Ao CLE - Centro de Lógica, Epistemologia e História da Ciência da Unicamp, por ter aberto suas portas para que fosse possível cursar disciplinas relacionadas à lógica matemática, por dois semestres; o que muito contribuiu para a capacidade de adquirir novos conhecimentos relacionados à organização da informação, e também abriu espaço para o segundo ano fora de Brasília, durante o estágio doutoral no exterior. Agradeço os

professores e colegas desse departamento pela hospitalidade e ajuda na assimilação das disciplinas.

Ao Institut de Recherche en Informatique de Toulouse, da Université Paul Sabatier, em Toulouse-França, pela hospitalidade e oportunidade em contribuir com a pesquisa em lógica matemática e mecanismos de representação do conhecimento, que mais uma vez, ampliou a capacidade de pensar através da ótica da informação, indo muito além da Informática e Ciência da Computação.

Às bancas do exame de qualificação e examinadora da defesa, pela contribuição e participação.

Ao meu orientador, Prof. Dr. Cláudio Gottschalg-Duque, pela confiança, humildade, paciência, dedicação, generosidade e grande oportunidade para conseguir finalizar o trabalho. Também por sua disponibilidade e presença constante, mesmo nos meses em que se encontrava ausente, na Alemanha. Sem sua imprescindível contribuição, a conclusão do trabalho não teria sido possível.

À minha família, por todo o apoio, estímulo e compreensão pelos momentos de ausência. Em especial aos meus pais, sempre um grande exemplo de valores e dignidade, sem eles nada disso teria sido possível.

À vida, por ter me possibilitado ampliar meus conhecimentos em diversas ordens, seja científicos, técnicos, acadêmicos, culturais e, principalmente, conhecer tantas pessoas especiais que fizeram de mim uma pessoa melhor depois dessa trajetória.

E por fim, agradeço ao Criador, Pensamento Universal, Amor Universal, grande força propulsora para nossa superação e sem o qual nada disso faria sentido.

[...] Effective information architects make the complex clear; they make the information understandable to other human beings. If they succeed in doing that, they're good information architects. If they fail, they're not.. (WURMAN et al., 2001, p.23).

Resumo

A Web Semântica, sendo a nova visão proposta pelo World Wide Web Consortium (W3C) para a estrutura atual da Web, é destinada a aumentar as possibilidades que ela oferece e, dessa forma, tornar seus recursos mais acessíveis para as máquinas. Entretanto, as técnicas de raciocínio de inferência utilizadas atualmente sobre os conhecimentos descritos pelas linguagens de representação de semânticas Web se baseiam somente sobre as capacidades inferenciais do formalismo de lógicas de descrição e os da lógica de predicados, já largamente utilizados nos motores de inferências. Este fato introduz a importância das pesquisas que tratam da representação do conhecimento e das técnicas de raciocínio sobre as ontologias na ótica da Web Semântica. O presente trabalho se propõe a estudar a capacidade dos grafos conceituais para representar e operacionalizar as ontologias da Web, assim como as contribuições dessa abordagem, de um ponto de vista simbólico (poder e facilidade de representação e interpretação) e inferencial (tipos de inferência aplicáveis a esse formalismo). Sendo a Arquitetura da Informação uma disciplina indissociavelmente ligada às tecnologias da informação, o presente trabalho é uma proposta para a interdisciplinaridade entre esta e a Ciência da Computação.

Palavras-chave: Arquitetura da Informação. Web Semântica. OWL. Linguagens de Representação. Lógica. Grafos Conceituais.

Abstract

The Semantic Web, being the new vision proposed by World Wide Web Consortium (W3C), to the current structure of the Web, is designed to increase the possibilities that it provides, and thereby, render its resources more accessible to machines. However, the techniques of inference on the knowledge described by representation languages of Semantic Web are based only on the inferential formalism of description logics capabilities and formalisms of predicate logic, already widely used in the inferences engines. This fact introduces the importance of research that deals with the knowledge representation and reasoning techniques on ontologies from the perspective of the Semantic Web. The present work was geared towards the capacity of conceptual graphs to represent and operationalize the Web ontologies, as well as the contributions of this approach, as a symbolic viewpoint (power and representation, and interpretation facility) and inferential (inference types applicable to this formalism). Since the Information Architecture is one discipline inextricably linked to information technology, the present work is a proposal for the interdisciplinarity between this and Computer Science.

Key-words: Information Architecture. Semantic Web. OWL. Representation Languages. Logic. Conceptual Graphs.

Lista de Ilustrações

Figura 1 - Mapa da Ciência da Informação segundo Hawkins.....	37
Figura 2 - Interdisciplinaridade da Ciência da Informação	38
Figura 3 - Mapa conceitual para compreender a pesquisa na Internet	45
Figura 4 - Camadas da Web Semântica.....	48
Figura 5 - Grafo orientado representando RDF.....	58
Figura 6 - Grafo “em branco”.....	58
Figura 7 - Grafo com triplas	59
Figura 8 - Exemplo de propriedades RDF.....	62
Figura 9 - Resumo das classes e relações no RDFS.....	63
Figura 10 - Exemplo de “reasoner” de IW desenvolvido em Standord.....	76
Figura 11 - Tabela verdade Lógica Booleana.....	92
Figura 12 - Rede Semântica.....	100
Figura 13 - Tipos de Relações	100
Figura 14 – Exemplo 1 de grafo conceitual.....	114
Figura 15 – Exemplo 2 de grafo conceitual.....	114
Figura 16 – Exemplo 3 de grafo conceitual.....	115
Figura 17 - Conceito expresso por cálculo lambda	118
Figura 18 - Exemplo 1 de cálculo lambda.....	118
Figura 19 - Exemplo 2 de cálculo lambda	119
Figura 20 - Exemplo 3 de cálculo lambda.....	120
Figura 21 - Exemplo 4 de cálculo lambda.....	120
Figura 22 – Exemplo 5 de cálculo lambda	121
Figura 23 - Hierarquia de tipos de conceitos.....	123
Figura 24 - Hierarquia de relações	124
Figura 25 - Representação dos fatos.....	125
Figura 26 - Restrição de integridade do domínio	125
Figura 27 - Regra de inferência	126
Figura 28 - Grafo conceitual e lógica predicativa	127
Figura 29 - Escrita dos GCs.....	128
Figura 30 - Especialização e generalização	130
Figura 31 – Exemplo de generalização.....	131
Figura 32 – Exemplo 1 de especialização.....	131
Figura 33 – Exemplo 2 de especialização	132
Figura 34 - Exemplo da expressividade dos grafos conceituais.....	133
Figura 35 - Representação de conhecimento modal.....	133
Figura 36 - Dedução natural	141
Figura 37 - Regras de dedução 1	142
Figura 38 – Procedimento de resolução	143
Figura 39 - Regras de dedução 3	144
Figura 40 - Abdução por cobertura.....	148
Figura 41 - Bases de conhecimento OWL e SOWL.....	170

Figura 42 - Exemplo de tipo de conceito.....	172
Figura 43 - Exemplo de tipo de relação.....	173
Figura 44 - Exemplo de relação de equivalência.....	176
Figura 45 - Interseção de tipos	177
Figura 46 - União de tipos	178
Figura 47 - Complemento de tipo.....	179
Figura 48 - Enumeração de tipos.....	180
Figura 49 - Restrição existencial.....	181
Figura 50 - Restrição universal.....	181
Figura 51 - Restrição de propriedade	182
Figura 52 - Restrição de cardinalidade.....	184
Figura 53 - Restrição de cardinalidade mínima.....	185
Figura 54 - Restrição de cardinalidade máxima.....	185
Figura 55 - Exemplo de restrição de cardinalidade.....	186
Figura 56 - Exemplo de incompatibilidade de tipos.....	187
Figura 57 - Relação inversa.....	189
Figura 58 - Etiquetagem em uma hierarquia de relações.....	192
Figura 59 - Exemplo de base de conhecimento.....	194
Figura 60 - Exemplo de esquema de axioma SameAs.....	194
Figura 61 - Exemplo de esquema de axioma Different From.....	195
Figura 62 - Exemplo de base de fatos.....	196
Figura 63 - Exemplo de comentário.....	198
Figura 64 - Exemplo da relação KBImport.....	199
Figura 65 - Exemplo de controle de versão.....	200
Figura 66 - Exemplo de Compatible_with.....	200
Figura 67 - Exemplo de Prior_version.....	201
Figura 68 - Exemplo de tradução em lógica de predicados.....	204
Figura 69 - Exemplo de base de conhecimento.....	207
Figura 70 - Ciclo de resolução.....	208
Figura 71 - Expressão lambda para o conceito mecânico.....	210
Figura 72 - Expressão lambda para o relação “voar”.....	210
Figura 73 - Grafo para a relação “voar”.....	210
Figura 74 - Grafo resultante.....	211
Figura 75 - Grafo de conceitos.....	214
Figura 76 - Operação Join.....	216
Figura 77 - Base de conhecimentos.....	219
Figura 78 - Generalização do conceito doente.....	221
Figura 79 - Notação gráfica.....	226

Lista de Quadros

Quadro 1 – Alguns construtores conceituais das Lógicas de Descrição	103
Quadro 2 - Construtores para definições de papéis complexos.....	103
Quadro 3 - Marcadores para o catálogo de indivíduos.....	104
Quadro 4- Marcadores	124

Lista de Abreviaturas e Siglas

CGIF - *Conceptual Graph Interchange Form*

DAML-S - *DARPA agent markup language for services*

FTP - *File Transfer Protocol*

GCs - *Grafos Conceituais*

HTML - *HyperText Markup Language*

HTTP - *Hypertext Transfer Protocol*

HTTPS - *Hypertext Transfer Protocol Secure*

IA - *Inteligência Artificial*

IWE - *Inference Web Explainer*

J2EE - *Java Platform, Enterprise Edition*

JTP - *Java Theorem Prover*

KIF - *Knowledge Interchange Format*

LDs - *Lógicas das Descrições*

MKBEEM - *Multilingual Knowledge Based European Electronic Marketplace*

OWL - *Web Ontology Language*

PML - *Proof Markup Language*

UDDI - *Universal Description, Discovery and Integration*

URI - *Uniform Resource Identifier*

RDF - *Resource Description Framework*

RDFS - *Resource Description Framework Schema*

SBC - *Sistemas Baseados em Conhecimento*

SNARK - *SRI's New Automated Reasoning Kit*

SRI's - *Stanford Research Institute's*

STS - *StoredTaxonomic Structure*

W3C - *World Wide Web Consortium*

WSDL - *Web Services Description Language*

WWW - *World Wide Web*

XML - *Extensible Markup Language*

Sumário

1 Apresentação	16
1.1 Introdução	16
1.2 Problemática	19
1.3 Objetivos	21
1.3.1 <i>Objetivo Geral</i>	21
1.3.2 <i>Objetivos Específicos</i>	21
1.4 Justificativa	21
2 Revisão da Literatura	24
2.1 Ciência da Informação	24
2.1.1 <i>Origem e evolução</i>	24
2.1.2 <i>A Ciência da Informação e o conceito de informação</i>	29
2.1.3 <i>Natureza interdisciplinar</i>	34
2.1.4 <i>A necessidade da Ciência da Informação</i>	39
2.2 Arquitetura da Informação	41
2.2.1 <i>Arquitetura da Informação e Web</i>	44
2.3 A Web Semântica	47
2.3.1 <i>Apresentação da Estrutura da Web Semântica</i>	48
2.3.2 <i>Domínios de aplicação e implementação existentes na Web Semântica</i>	78
2.3.3 <i>Discussão</i>	87
2.4 Os formalismos de Representação	88
2.4.1 <i>Formalismos de Representação</i>	89
2.4.2 <i>Discussão</i>	134
2.5 Técnicas de Inferência	135
2.5.1 <i>Raciocínio dedutivo</i>	135
2.5.2 <i>Raciocínio por abdução</i>	145
2.5.3 <i>Raciocínio por analogia</i>	153
2.5.4 <i>Discussão</i>	156
2.6 Trabalhos Relacionados	157
3 Metodologia	163
3.1 Pesquisa documental	164
3.2 Modelagem	165
4 Resultado	167
4.1 Resultado 1 - Linguagem SOWL	167
4.1.1 <i>TBOX → Hierarquias de tipos de conceitos e de relações</i>	171
4.1.2 <i>ABOX → Catálogo de indivíduos e a base de fatos</i>	193
4.1.3 <i>Informações Operacionais</i>	196
4.1.4 <i>Comparação e síntese</i>	201
4.2 Resultado 2 - Mecanismos de Inferência	202
4.2.1 <i>O raciocínio dedutivo</i>	203
4.2.2 <i>Raciocínio por abdução</i>	211
4.2.3 <i>Raciocínio por analogia</i>	217
4.3 Resultado 3 - Exemplos com a nova linguagem SOWL	224
4.3.1 <i>Ontologias e OWL</i>	224
4.3.2 <i>Grafos conceituais</i>	225
4.3.3 <i>Demonstração por exemplos</i>	226
5 Conclusão e considerações finais	234

5.1 Trabalhos futuros	235
Referências.....	239

1 Apresentação

1.1 Introdução

“A Web Semântica não é uma Web separada, mas uma extensão da atual. Nela a informação é dada com um significado bem definido, permitindo melhor interação entre os computadores e as pessoas” (BERNERS-LEE, et al., 2001, p.1). Com estas palavras, Berners-Lee et al. (2001) definem os planos de seu grupo de trabalho *no World Wide Web Consortium* (W3C) para operar a transformação que irá modificar a Web como a conhecemos hoje. “Web Semântica” (BERNERS-LEE et al., 2001; BIZER, HEATH, BERNERS-LEE, 2009) é o nome genérico deste projeto, liderado pelo W3C, que pretende criar mecanismos que proporcionem o desenvolvimento de aplicações inteligentes e contexto nos códigos *Extensible Markup Language* (XML) utilizados para confecção de páginas Web, de modo a melhorar a forma com que programas podem interagir com estas páginas e também possibilitar um uso mais intuitivo por parte dos usuários.

Berners-Lee et al. (2001) imaginam um mundo em que programas e dispositivos especializados e personalizados, chamados agentes, possam interagir por meio da infraestrutura de dados da Internet trocando informações entre si, de forma a automatizar tarefas rotineiras dos usuários. O projeto da Web Semântica, em sua essência, é a criação e implantação de padrões (*standards*) tecnológicos para permitir este panorama, que não somente facilite as trocas de informações entre agentes pessoais, mas principalmente estabeleça uma linguagem comum para o compartilhamento mais significativo de dados entre dispositivos e sistemas de informação de uma maneira geral (MONTEIRO, 2013).

Para atingir tal propósito, é necessária uma padronização de tecnologias, de linguagens e de metadados descritivos, de forma que todos os usuários da Web obedeçam a determinadas regras comuns e compartilhadas sobre como armazenar dados e descrever a informação armazenada e que esta possa ser “consumida” por outros usuários humanos ou não, de maneira automática e não ambígua.

No escopo da Ciência da Informação Souza e Alvarenga argumentam:

[...] o estudo das possibilidades que se abrem e a compreensão de que todo o embasamento filosófico, metodológico e conceitual da Web Semântica parte do núcleo duro da Ciência da Informação nos impelem a demarcar e arrebanhar os legítimos territórios do saber e a buscar ativamente uma atuação no desenho destes novos panoramas informacionais. Precisamos hoje trilhar um caminho rumo a uma nova e necessária valorização da área de Ciência da Informação, que oferece teoria, metodologias e competências que compõem a quintessência daquilo que se espera dos trabalhadores e pesquisadores de uma sociedade baseada em informação e conhecimento. E a importância da Web e das demais redes digitais de troca de informações no panorama mundial são amostras de como a atividade de organização da informação é necessária para a evolução dos indivíduos, organizações e da sociedade em geral. (SOUZA; ALVARENGA, 2004, p.9).

A palavra de ordem que se iniciou com o XML é “interoperabilidade”, ou seja, a possibilidade de sistemas diferentes “conversarem” entre si. Tudo indica que os padrões que estão sendo desenhados para esta nova Web também sejam adotados na arquitetura de bibliotecas digitais e de novos sistemas de informação. Assim, pode-se esperar que estas tecnologias também tragam mudanças para a área e a atividade dos profissionais da Ciência da Informação (MARCONDES; CAMPOS, 2008).

E relacionando as contribuições da Web Semântica para a Ciência da Informação, é possível exemplificar com o presente trabalho, uma atividade específica que será possibilitada ou melhorada com esta “nova Web” (mais semântica).

A Web Semântica, sendo a nova visão proposta pelo W3C para a estrutura atual da Web, é destinada aumentar as possibilidades que ela oferece e, fazendo seus recursos mais acessíveis para as máquinas e, assim alcançar uma melhora qualitativa e quantitativa dos tratamentos possíveis por estes últimos.

Isto será possível essencialmente através da associação aos recursos da Web das informações necessárias para sua utilização, implementadas devido as linguagens formalizadas de modo a torna-las utilizáveis de forma automática.

A implementação dessas descrições se fará graças aos metadados expressos com a ajuda de linguagens de anotação como *Resource Description Framework* (RDF) ou *Resource Description Framework Schema* (RDFS), mas sobretudo de ontologias (cuja linguagem de implementação mais conhecida é *Web Ontology Language* (OWL)) sobre as quais diferentes tipos de raciocínio e inferência poderão ser aplicados a fim de desenvolver tratamentos “inteligentes” sobre o conteúdo dos recursos descritos (SILVA et al., 2012).

A revisão da literatura do presente trabalho é dividida nos seis tópicos seguintes:

1 A Ciência da Informação: Apresenta-se nesta seção as origens e evolução da Ciência da Informação, bem como suas definições, sua relação com o conceito de 'informação', a natureza interdisciplinar da Ciência da Informação, com ênfase na interdisciplinaridade com a Ciência da Computação e, por fim, sobre a necessidade da Ciência da Informação.

2 A Arquitetura da Informação: São apresentadas as definições atuais da Arquitetura da Informação, sua importância para a Ciência da Informação e interdisciplinaridade com outras ciências, em particular, com a Informática, e a relação e importância desta para com a Web.

3 A Web Semântica: Serão apresentados as diferentes definições da Web semântica, seus objetivos, assim como as diferentes tecnologias que constituem a plataforma aplicada para sua elaboração. Isto será feito enfatizando a importância da descrição de recursos (devido as linguagens de formulação dos metadados como RDF ou RDFS ou outras linguagens de representação como OWL) e dos métodos de exploração de suas descrições (motores de inferência, técnicas de exploração ligados aos formalismos das linguagens de representação).

4 Os formalismos de representação: Apresenta-se aqui diferentes formalismos de representação de conhecimentos (seus fundamentos teóricos e especificações). O trabalho se interessa particularmente por lógicas de predicados (sobre a qual se baseia a maior parte dos motores de inferência atuais), por lógicas de descrição (amplamente utilizadas na Web Semântica) assim como o formalismo dos grafos conceituais (sobre o qual se baseia o presente trabalho).

5 Técnicas de inferência: Onde serão apresentadas três técnicas de inferência para o presente estudo.

- Inferência dedutiva.
- Inferência abdutiva.
- Resolução de problemas por analogia.

6 Trabalhos Relacionados

E para finalizar, a parte dos resultados será composta em três partes:

Parte 1 - Definir a linguagem SOWL através das primitivas de construção da linguagem OWL (linguagem de representação de ontologias para a Web Semântica, cujas bases teóricas se fundamentam no formalismo das lógicas de descrição) e através de outras primitivas retiradas do formalismo dos grafos conceituais;

Parte 2 - Aplicação das técnicas de inferência citadas na Revisão da Literatura na nova linguagem proposta;

Parte 3 – Exemplos com a nova linguagem SOWL.

1.2 Problemática

A Web Semântica é uma nova visão da estrutura da Web, concebida em um artigo de Berners-Lee em 1998 (proposta pelo W3C em 2002). É destinada a aumentar as possibilidades desta última tornando seus recursos acessíveis às máquinas.

A falta de formalismo das tecnologias da Web atual torna seu conteúdo classificável somente sintaticamente. Isto no sentido de que apenas o usuário humano pode identificar, selecionar e manipular os dados e recursos de maneira eficaz; devido ao fato que estes últimos não possuem as estruturas explícitas ou as informações formalizadas. O aumento da taxa de exploração automática de recursos da Web - que é o objetivo da Web Semântica - apenas pode ser conseguido tornando o conteúdo destes últimos semanticamente compreensíveis pelas máquinas, associando as informações operacionais descritas de maneira formal.

Porém, limitar a Web Semântica a uma simples descrição de recursos não permite perceber sua verdadeira dimensão, da mesma forma, as necessidades de sua implementação. Estas descrições formais deverão ser expandidas pelos mecanismos de exploração inferenciais permitindo uma manipulação “mais inteligente” dos conhecimentos, o que levará a um aumento significativo na velocidade de tratamento devido a limitação - até mesmo a eliminação em certos casos - de intervenção

humana. Outras tecnologias de suporte virão operacionalizar esta infraestrutura, vindo de diversas áreas (Engenharia de *Software*, Cálculo Formal e Verificação de Programas, Integração e Bases de Dados) e demandando somente esforços de readaptação às especificidades do ambiente Web, tais como a distributividade, inconsistência de conhecimento, quantidade muito grande de recursos etc. (CHARLET; LAUBLET; REYNAUD, 2003).

A criação de tal infraestrutura permitirá uma manipulação mais ergonômica dos recursos e conhecimentos da Web por seus usuários, e personalizarão também os diferentes métodos de interação que estes últimos serão levados a manipular da mesma forma que um aumento qualitativo e de volume da operação de informações e dos serviços mais variados.

A Web Semântica é uma plataforma evolutiva na qual circular um fluxo importante de dados, sendo que estes últimos serão de natureza heterogênea e seu tratamento será via métodos clássicos, propensos à complexidade e rentabilidade insatisfatória, em um mundo onde as palavras de ordem são rapidez e eficácia.

Igualmente, a semântica propiciada pela Web favorece o entendimento dos conteúdos pelos sistemas digitais, apresentando resultados mais condizentes com as necessidades informacionais dos usuários. Dessa forma, a Web Semântica pode favorecer o avanço significativo das técnicas de indexação, armazenamento e recuperação da informação em rede, contribuindo notadamente para a evolução da Ciência da Informação (VILLALOBOS; SILVA, 2012).

Como consequência, os trabalhos são conduzidos no seio da comunidade de pesquisa com a finalidade de responder a esses problemas; seguindo dois eixos principais: análise de modos de representação de dados e o processamento das informações e da inferência. Assim, a questão que se pretende responder com esta pesquisa é:

É possível uma reescritura das primitivas de representação OWL, considerando diferentes formas de raciocínio com base nos paradigmas do formalismo dos Gráficos Conceituais?

1.3 Objetivos

1.3.1 Objetivo Geral

Propor uma nova linguagem “SOWL” para as ontologias da Web considerando diferentes formas de raciocínio com base nos paradigmas do formalismo dos gráficos conceituais.

1.3.2 Objetivos Específicos

1. Estudar a capacidade dos grafos conceituais para representar e operacionalizar as ontologias da Web.
2. Definir uma nova linguagem “SOWL”, por meio de ferramentas desenvolvidas a partir do formalismo dos grafos conceituais.
3. Propor técnicas de inferência para a nova linguagem proposta.

1.4 Justificativa

Um dos objetivos originais da Web era a troca de informação entre pessoas, de maneira que os computadores pudessem participar da comunicação, ajudando os usuários. Os computadores na Web, atualmente, têm papel somente no direcionamento e entrega de informações, não tendo acesso ao conteúdo das páginas, porque essa informação está estruturada para utilização pelas pessoas e não por máquinas. Hoje, temos uma Web de documentos e não de informações. Por isso, os computadores oferecem ajuda limitada no acesso e processamento da informação, deixando as funções de extração e interpretação dessa informação a cargo dos usuários.

A Web Semântica visa resolver este problema, por meio da estruturação do conteúdo das páginas Web de forma que a informação possa ser interpretada pelas máquinas. A proposta não é a de uma Web separada da atual, mas uma extensão da mesma, baseada em

estruturas – as ontologias – que descrevem relacionamentos entre objetos e contendo informação semântica dos mesmos para automatizar o processamento pelas máquinas.

Na Web há uma quantidade imensa de informações não pertinentes que é fornecida pelos processos de busca. As ferramentas de busca enfrentam a dificuldade de executar pesquisas entre documentos que não estão diferenciados em termos de assunto, qualidade e relevância. A tecnologia atual não é capaz de diferenciar uma informação comercial de uma educacional, ou informação entre culturas e mídia. É necessário haver informações de qualificação da própria informação, chamada de metadados, para ser possível classificá-las e tornar os processos de busca mais eficazes. Algumas dessas novas estruturas necessárias já foram definidas e outras ainda estão sendo desenvolvidas pelo *Word Wide Web Consortium* (W3C). W3C é um composto de organizações interessadas na definição e desenvolvimento de novos conceitos, protocolos e padrões de estruturas para a Web, visando obter maior eficácia de seus recursos.

Devido a essa carência tecnológica, a Web Semântica pretende solucionar o problema da falta de informação semântica e significativa a respeito do conteúdo disposto na Web através de novas tecnologias, tais como: a utilização de um sistema de marcação de páginas flexível como o XML, que permite incluir informações significativas a respeito de palavras ou termos do documento; uma arquitetura de metadados como RDF, que padroniza a criação de metadados na Web permitindo maior intercâmbio e reutilização de descrições de termos e palavras; e ontologias, que disponibilizem um vocabulário específico de conhecimento e descrevem os termos e seus relacionamentos.

Entretanto, as técnicas de raciocínio de inferência utilizadas atualmente sobre os conhecimentos descritos pelas linguagens de representação de semânticas Web se baseiam somente sobre as capacidades inferenciais do formalismo de lógicas de descrição - que é o formalismo de representações de conhecimento, conhecido como base teórica das linguagens de implementação de ontologias - e os da lógica de predicados, já largamente utilizados nos motores de inferências.

Este fato poderia introduzir a importância das pesquisas que tratam da representação do conhecimento e das técnicas de raciocínio sobre as ontologias na ótica da Web Semântica. Com este viés, o presente trabalho explora o estudo da capacidade de grafos conceituais (formalismo de representação apresentado por SOWA 1984) para representar e operacionalizar as ontologias da Web, assim como suas contribuições; de um ponto

de vista simbólico (poder e facilidade de representação e interpretação) e inferencial (tipos de inferência aplicáveis a esse formalismo).

A contínua evolução da Internet, meio em que se desenvolve a Web Semântica, pode tornar esta pesquisa trabalhosa e ao mesmo tempo ressaltar a importância de estudos sobre organização da informação nas camadas semânticas da OWL, assim como este, inserido na linha de pesquisa de Arquitetura da Informação da Pós-graduação em Ciência da Informação da Faculdade de Ciência da Informação da Universidade de Brasília (PPGCInf – UnB). Portanto, vale ressaltar que a Ciência da Informação possui abordagens e conceitos fundamentais para esta pesquisa e ainda, profissionais adequados para elaboração e execução deste tipo de estudo, conforme completa Tillman (2003), ao apontar os profissionais da informação como os mais preparados para determinar e expandir critérios para a informação disponível na Internet.

Além disso, a Web Semântica possibilita transformações para a Ciência da Informação, havendo um processo de simbiose entre estas áreas, representando a junção real da Tecnologia da Informação com as técnicas utilizadas em Biblioteconomia e Documentação (VILLALOBOS; SILVA, 2012).

2 Revisão da Literatura

Apresenta-se neste capítulo o referencial teórico da pesquisa. Isto se dá a partir de uma pesquisa sobre Ciência da Informação, Arquitetura da Informação, e os benefícios que a sua interdisciplinaridade, principalmente com a Ciência da Computação, podem ser alcançados para uma melhor organização da informação.

Da mesma forma, uma pesquisa sobre a Web Semântica e OWL, mais especificamente sobre as carências da camada semântica. Seguindo a perspectiva, é realizado estudo dos princípios e apresentação da Web, metadados, ontologias e domínios de aplicação. Na sequência, realiza-se o estudo sobre formalismos de representação, incluindo lógica clássica, redes semânticas, lógicas de descrição e critérios de avaliação na escolha da representação. Por fim, estuda-se as técnicas de inferência e tipos de raciocínio. Assim, evidenciam-se as contribuições dessa pesquisa para a melhor eficiência da Web Semântica, por meio de tradução das primitivas de representação OWL e da consideração de diferentes formas de raciocínio com base nos paradigmas do formalismo dos grafos conceituais, do estudo sobre a capacidade de grafos conceituais para representar e operacionalizar as ontologias da Web e da aplicação das técnicas de inferência elencadas em uma nova linguagem proposta.

2.1 Ciência da Informação

2.1.1 *Origem e evolução*

O nascimento da Ciência da Informação pode ser visto como consequência de uma sucessão de técnicas relacionadas com o registro físico do conhecimento, principalmente a escrita. A escrita permitiu registrar, estocar e recuperar o conhecimento; consequentemente, gerou uma espiral cumulativa de textos cujo potencial foi amplificado quando Johann Gutenberg inventou o tipo móvel e apresentou a primeira prensa na Europa em meados do século XIII.

Segundo (FERNEDA, 2003), o sucesso do invento de Gutenberg só não foi mais imediato

pelo fato de que naquela época poucas pessoas sabiam ler. Em uma sociedade basicamente agrária, os camponeses nada tinham a ganhar com a alfabetização, e em geral não aspiravam a ela. Porém, a Revolução Industrial iniciada em meados do século XVIII provocou o êxodo das populações do campo para a cidade e deu impulso à procura por mais informação e à qualificação necessária para sua interpretação e utilização. A construção de estradas e o surgimento das estradas de ferro facilitaram a expansão do comércio e a distribuição de livros e jornais. A velocidade das mensagens passou da velocidade do cavalo para a da locomotiva e desta para a eletricidade.

Novas invenções se seguiram durante a segunda metade do século XIX, a maioria delas ligadas à transmissão de informação. Em 1876, Alexander Graham Bell, que em 1844 havia inventado o telégrafo, estendeu o alcance da voz humana ao inventar o seu “telégrafo falante”, o telefone. No ano seguinte, Thomas Edison criou a primeira máquina de gravar sons e em 1879, projetou a lâmpada elétrica (MCGARRY, 1999).

Segundo Castells (1999), esse período de transformações tecnológicas em aceleração marca uma descontinuidade histórica irreversível na base material da espécie humana. O repentino aumento de aplicações tecnológicas transformou os processos de produção e distribuição de bens e serviços, criou uma grande quantidade de novos produtos e mudou de maneira decisiva a localização das riquezas e do poder no mundo, que ficou ao alcance dos países e elites capazes de comandar esse sistema tecnológico.

De acordo com (MACEDO, 2005), o conceito de Ciência da Informação surge em um momento em que os indivíduos começam a se preocupar com a questão da sistematização e do acesso a uma quantidade crescente de informações. A industrialização da imprensa comercial, associada ao barateamento dos materiais gráficos, promoveu a chamada explosão bibliográfica, fenômeno não menos importante do que o advento da Imprensa de Gutenberg, ocorrido por volta de 1450. Os efeitos do fenômeno ficaram mais evidentes após a 2ª Guerra Mundial. Mas a preocupação com a organização e o acesso ao conhecimento vem de tempos mais remotos. De acordo com Edson Nery da Fonseca (2007):

Os conhecimentos acumulados pelo homem através dos tempos foram pela primeira vez sistematizados na China antiga; os gregos, porém, deram maior ênfase a essa sistematização, principalmente com a obra de Aristóteles, primeiro filósofo a tratar de matérias variadas como a Lógica, a Física, a Psicologia e a Antropologia, a Zoologia, a Metafísica, a Ética, a Política, a Retórica e a Poética (FONSECA, 1991 apud MACEDO, 2005, p.59).

Em 660 a.C., um rei assírio organizava suas tabuletas de argila por assunto. Em 330 a.C. a Biblioteca de Alexandria já armazenava uma coleção de pergaminhos. Em 1873, Melvil Dewey (1851-1931) desenvolveu a Classificação Decimal de Dewey como uma ferramenta para organizar e facilitar o acesso ao crescente número de livros (MACEDO, 2005).

Otlet, em seu “*Traité de Documentation*” (1934), mostra-se interessado em toda novidade tecnológica que permita condensar e organizar a informação de acordo com suas necessidades e objetivos. Otlet e Henri La Fontaine entraram para a história da biblioteconomia como autores da Classificação Decimal Universal (CDU). Em 1895 fundam em Bruxelas, na Bélgica, o International Institute for Bibliography – IIB, marco no desenvolvimento do que veio a se chamar Documentação e posteriormente Ciência da Informação. O primeiro objetivo do IIB era a elaboração do Repertório Bibliográfico Universal (RBU), que tinha a pretensão de sintetizar toda a produção bibliográfica internacional em fichas padronizadas. Para Otlet as fichas rompiam a linearidade do texto Escrito, permitindo a livre associação entre as informações nelas registradas. Devidamente conectadas através dos códigos da CDU, essa rede de fichas pode ser vista como um prenúncio do hipertexto. As solicitações de pesquisa nesse grande banco de dados eram feitas através do correio e sua operacionalização era bastante demorada. Em uma época na qual não existiam fotocopiadoras ou computadores, era necessário remover as fichas do arquivo, copiá-las à mão e recolocá-las de volta no arquivo. Além da execução das “buscas”, era também tarefa dos funcionários sintetizar e copiar nas fichas os materiais enviados por colaboradores de toda a parte do mundo (RAYWARD, 1994).

Para navegar sobre os repositórios de dados, Otlet vislumbrou um tipo de estação de trabalho em formato de roda, uma mesa móvel alimentada por uma rede de dobradiças com várias superfícies móveis, algo semelhante ao engenho proposto por Agostino Ramelli, descrito na obra *Le diverse et artificiose machine del Capitano Agostina Ramelli* (1588) (RAMELLI, 1970), que possibilitaria a consulta a vários livros simultaneamente. A diferença básica é que a máquina de Otlet permitiria aos usuários não só ler, escrever e procurar informações a sua maneira, mas também fazer anotações e relacionamentos entre as informações.

Otlet imaginou ainda a possibilidade de acesso às bases de dados a partir de grandes distâncias, por meio de um ‘telescópio elétrico’, conectado por uma linha telefônica, capaz de recuperar uma imagem em fac-símile a ser projetada remotamente em uma tela plana. A estação de trabalho estaria conectada a imensos centros de informações. O esforço total representaria uma grande rede do conhecimento humano.

Entretanto, alguns autores afirmam que as soluções propostas por Otlet para os problemas de estruturação da informação refletiam uma visão ingênua da natureza do conhecimento e da dinâmica de seu crescimento. Ele teria subestimado a complexidade dos processos físicos e sociais envolvidos, apresentando uma abordagem tecnicista e positivista. Nesse sentido, seu ponto de vista relacionado à sistematização do conhecimento diferia essencialmente dos atuais na medida em que não tinha como foco o usuário. Seu compromisso seria com o ‘conhecimento objetivo’ contido nos documentos, independente de autoria e uso. Os usuários passaram a ser o foco do processo de desenho de tecnologias a partir da publicação *The design of everyday things* (RAYWARD, 1994).

Otlet era um homem com imensa curiosidade em relação às inovações tecnológicas que pudessem ser úteis no processo de condensação e registro da informação. Fez diversas experimentações com a microfimagem e previu um futuro promissor para uma invenção surgida na época: a televisão. Anteviu vários equipamentos tecnológicos como o fax, os microcomputadores, as *work-stations*, a Internet (OTLET, 1934). Paul Otlet morreu em 1944, às vésperas do final da Segunda Guerra Mundial.

Após a Segunda Guerra Mundial, o entusiasmo na busca de soluções para os problemas advindos da explosão informacional pode ser resumido pelo artigo de Vannevar Bush (1945) intitulado ‘*As We May Think*’. Nesse artigo, Bush define o problema do gerenciamento da informação e propõe como solução uma máquina, denominada Memex,

que agregava as mais modernas tecnologias de informação existentes na época. O Memex nunca foi construído, mas as idéias que inspiraram sua idealização ainda fazem parte das aspirações de pesquisadores e cientistas da atualidade. Em uma escala muito maior, enfreta-se hoje os mesmos problemas apontados por Otlet, e, como Bush, busca-se na tecnologia a solução para tais problemas.

Em 1938, Herbert George Wells (1866-1946) publicou uma coleção de ensaios sobre futuras organizações do conhecimento e educacionais, denominada *World Brain*, que incluía o ensaio: *The idea of a permanent world encyclopaedia* (WELLS, 1937). O ‘Cérebro Mundial’ de Wells conectaria homens e conhecimentos através do tempo e do espaço. Com acesso instantâneo em todo o mundo, permitiria um aprendizado mais rápido e possibilitaria a criação de um ‘mundo melhor’. Sua proposta era extremamente aguçada, porém utópica para a época. Hoje, uma realidade: a *World Wide Web*.

Seguem-se a estas as idéias de Vannevar Bush (1945), que idealizou em seu artigo *As we may think*, um mecanismo de armazenamento e recuperação de informações denominado memex, que funcionaria também como uma extensão da memória humana. Com propósitos similares, Douglas Engelbart escreveu em 1963 o artigo *A conceptual framework*, afirmando que o computador poderia ser uma extensão do pensamento humano. Em 1987, Theodore Nelson implementou o conceito de ‘hipertexto’ - termo por ele cunhado - em seu revolucionário projeto Xanadu (MACEDO, 2005).

As visões apresentadas anteciparam a idéia do que viria a ser a *World Wide Web*, proposta por Tim Berners-Lee em 1989. Com o advento da *Web*, o fenômeno da explosão de informações adquire proporções ainda maiores, e cresce a preocupação com a sistematização e o acesso ao conhecimento. O desafio trazido por esse fenômeno certamente promoveu as principais inovações nas áreas de organização e recuperação da informação.

Capurro e Hjørland (2003) indicam duas raízes para a Ciência da Informação, sendo estas a Biblioteconomia clássica, que ele considera como o estudo dos problemas relacionados com a transmissão de mensagens; e a computação digital, que teve impacto nos processos de produção, coleta, organização, interpretação, armazenagem, recuperação, disseminação, transformação e uso da informação.

De acordo com Neveling e Wersig (1975), a Ciência da Informação se estabeleceu como disciplina por volta de 1950, surgindo a partir das exigências de áreas práticas da documentação ou recuperação da informação. A ciência não é algo que se possa justificar em si mesma, mas que é sempre justificável por alguma necessidade social que deve ser atendida por esta ciência (NEVELING; WERSIG, 1975).

2.1.2 *A Ciência da Informação e o conceito de informação*

Segundo Shera e Cleveland (1977), a década de 1960 forneceu um clima favorável para o desenvolvimento da Ciência da Informação. Os problemas relacionados com o tratamento da informação começavam a ser abordados por parte da comunidade científica mundial, ao mesmo tempo em que se vivia um período de acelerado desenvolvimento tecnológico (SHERA E CLEVELAND, 1977 apud FERNEDA, 2003).

A primeira formulação do que seria a Ciência da Informação surgiu como resultado das conferências do *Georgia Institute of Technology* (ou simplesmente “*Georgia Tech*”), realizadas entre 1961 e 1962:

Ciência da Informação é a ciência que investiga as propriedades e comportamento da informação, as forças que regem o fluxo da informação e os meios de processamento da informação para uma acessibilidade e usabilidade ótimas. Os processos incluem a origem, disseminação, coleta, organização, recuperação, interpretação e uso da informação. O campo deriva de ou relaciona-se com a matemática, a lógica, a linguística, a psicologia, a tecnologia da computação, a pesquisa operacional, as artes gráficas, as comunicações, a biblioteconomia, a administração e alguns outros campos tecnológicos (SHERA E CLEVELAND, 1977 apud FERNEDA, 2003).

Em 1968, Harold Borko formulou uma definição complementar, ressaltando suas características tanto de ciência pura como de ciência aplicada.

Ciência da Informação é a disciplina que investiga as propriedades e o comportamento da informação, as forças que regem o fluxo da informação e os meios de processamento da informação para acessibilidade e usabilidade ótimas. Está relacionada com o corpo de conhecimento que abrange a origem, coleta, organização, armazenamento, recuperação, interpretação, transmissão, transformação e utilização nos sistemas naturais e artificiais, o uso de códigos para a transmissão eficiente de mensagem, e o estudo dos dispositivos e técnicas de processamento de informação tais como computadores e seus sistemas. É uma ciência interdisciplinar derivada de e relacionada a vários campos tais como matemática, lógica, linguística, psicologia, tecnologia da computação, pesquisa operacional, artes gráficas, comunicações, biblioteconomia, administração e outros campos similares. Possui um componente de ciência pura, que investiga o assunto sem considerar suas aplicações, e um componente de ciência aplicada, que desenvolve serviços e produtos. (BORKO, 1968, p.1).

Saracevic (1996) aponta que:

[...] a Ciência da Informação é um campo dedicado às questões científicas e à prática profissional voltadas para os problemas da efetiva comunicação do conhecimento e de seus registros entre os seres humanos, no contexto social, institucional ou individual do uso e das necessidades de informação. No tratamento destas questões são consideradas de particular interesse as vantagens das modernas tecnologias informacionais. (SARAVEVIC, 1996, p.7).

O componente tecnológico, principalmente a “tecnologia da computação”, aparece em várias definições de Ciência da Informação. Alguns autores inserem a tecnologia em uma posição central, outros a colocam como resultado da interdisciplinaridade da Ciência da Informação.

Segundo (MCGARRY, 1999), a palavra “informação” se tornou popular logo após a invenção da imprensa no século XV, quando normalmente se utilizava uma palavra em latim para expressar uma nova idéia ou conceito. A raiz do termo vem de *formatio* e *forma*, ambos transmitindo a idéia de “moldar algo” ou dar “forma a” algo indeterminado.

Shannon e Weaver (1949) definem informação como:

O que acrescenta algo a uma representação [...] Recebemos informação quando o que conhecemos se modifica. Informação é aquilo que logicamente justifica alteração ou reforço de uma representação ou estado de coisas. As representações podem ser explicitadas como num mapa ou proposição, ou implícitas como no estado de atividade orientada para um objetivo do receptor. (SHANNON E WEAVER, 1949 apud MCGARRY, 1999, p.3).

Na visão de Shannon (1949), a informação não depende de um suporte material, mas de um emissor, um receptor e um canal, podendo ser facilmente quantificada. Esta definição de informação, base da Teoria da Informação, foi fundamental na construção dos primeiros

computadores eletrônicos, e ainda desempenha um papel importante no estudo da informação em diversos contextos (MACEDO, 2005).

Numa abordagem pragmática, (BUCKLAND, 1991) identifica três principais usos do termo “informação”:

- Como processo – o ato de informar ou a comunicação do conhecimento ou notícias sobre um fato ou ocorrência;
- Como conhecimento – o que é percebido pela informação enquanto processo, o conhecimento comunicado. Sua principal característica é a intangibilidade;
- Como coisa – aquilo que é visto como informativo: objetos, documentos, textos, dados ou eventos. A sua principal característica é a sua tangibilidade, sua materialidade.

Nos dois primeiros usos a informação para ser comunicada precisa estar “expressa, descrita ou representada em algum modo físico”, que seria a informação como coisa. Buckland (1991) define a “informação como coisa” em termos de potencial para o processo de informar, e defende o papel fundamental desta definição em sistema de recuperação de informação por este ser o único sentido o qual tais sistemas podem lidar diretamente.

Ruyer (1972), associando dados e informação, apresenta a seguinte definição:

A palavra ‘informação’, em seu sentido usual, parece comportar, necessariamente, um elemento de consciência e de sentido. [...] A informação, no sentido habitual do termo, é a transmissão a um ser consciente de uma significação, de uma noção, por meio de uma mensagem com base em um suporte espaço-temporal: imprensa, mensagem telefônica, onda sonora, etc. (RUYER, 1972, p.3).

Robredo (2003) apresenta e avalia diversos conceitos de informação. Inerente a quase todas as definições de informação analisadas no contexto da Ciência da Informação está evidenciado o seu caráter semântico.

Segundo Pinheiro e Loureiro, em conferência da *Special Libraries Association*, no ano de 1967, Rees e Saracevic (1992) elaboram a seguinte definição:

A Ciência da Informação não é uma melhor recuperação de dados, como a física não é uma mecânica reforçada... é um ramo de pesquisa que toma sua substância, seus métodos e suas técnicas de diversas disciplinas para chegar à compreensão das propriedades, comportamento e circulação de informação. (REES; SARACEVIC, 1992 apud PINHEIRO; LOUREIRO, 1995, p.2)

Sinteticamente, Ciência da Informação é, para os dois autores, o “estudo dos fenômenos da comunicação e das propriedades dos sistemas de comunicação” (REES; SARACEVIC, 1992 apud PINHEIRO; LOUREIRO, 1995)

No artigo *Ciência da Informação: o que é?*, Borko (1968) reúne e reelabora as idéias de Taylor, em uma definição que descortina ampla visão dos caminhos possíveis da reflexão sobre a natureza e conceitos da Ciência da Informação, cujas idéias se harmonizam, também, com as de Rees e Saracevic:

[...] Ciência da Informação é aquela disciplina que investiga as propriedades e o comportamento da informação, as forças que governam o fluxo da informação e os meios de processamento para acesso e uso otimizados. Ela diz respeito àquele corpo de conhecimento ligado à origem, coleta, organização, armazenagem, recuperação, interpretação, transmissão, transformação e utilização da informação [...] possui um comportamento de ciência pura, que investiga o interior do assunto sem considerar suas aplicações, é um componente de ciência aplicada, que desenvolve serviços e produtos. (BORKO, 1968, p.1).

No mesmo artigo, Borko (1968) afirma, de forma mais sintética, que,

em essência, a pesquisa na Ciência da Informação investiga as propriedades e comportamento da informação, a utilização e a transmissão da informação, bem como o processamento da informação para armazenagem e recuperação ótimas. (BORKO, 1968, p.2).

De acordo com (MACEDO, 2005), pelo levantamento realizado por Pinheiro e Loureiro (1995), o termo ‘Ciência da Informação’ foi possivelmente criado por volta de 1960, de acordo com (HEILPRIN, 1989), e se originou do estudo da produção, processamento e uso da informação. A partir de uma pesquisa terminológica, Wellish (1977) estabeleceu o ano de 1959 como datação provável para o primeiro uso do termo, que se referia ao estudo do conhecimento registrado e de sua transferência. As obras *Cybernetics or control and communication in the animal and machine* (1948), de Norbert Wiener e *The mathematical theory of communication* (1949), de Claude Shannon e Warren Weaver, prenunciaram o surgimento da Ciência da Informação, na opinião dos autores.

A formação do especialista em Ciência da Informação e o escopo da área foram temas das conferências realizadas em 1961 e 1962, no *Georgia Institute of Technology*. Os pesquisadores reunidos estavam interessados na ‘informação em si’ e na ciência que trataria de seu armazenamento e recuperação. Essas conferências são consideradas por diversos autores o ponto de partida para a consolidação da Ciência da Informação. (PINHEIRO; LOUREIRO, 1995) e (ROBREDO, 2003).

À época, Taylor (1963) definiu o escopo da área como o estudo das propriedades, estrutura e transmissão do conhecimento especializado, e [...] o desenvolvimento de métodos para a sua organização e disseminação úteis (ROBREDO, 2003).

Em 1968, quando o *American Documentation Institute* se tornou *American Society for Information Science*, Harold Borko propôs, com base nas idéias de Taylor (1963), a clássica definição para a Ciência da Informação que se segue (BATES, 1999; CAPURRO; HJØRLAND, 2003; PINHEIRO; LOUREIRO, 1995; ROBREDO, 2003).

Griffith (1980), motivado pelo mesmo evento, propôs definição similar, que estabelece a Ciência da Informação como uma disciplina que busca a criação e estruturação de um corpo de conhecimentos científicos, tecnológicos e sistêmicos relacionados à transferência de informações (GRIFFITH, 1980 apud CAPURRO, 1985).

Bates (1999) questiona esse paradigma, que permanece praticamente invariável e estável há mais de trinta anos. Afirma que, apesar de considerado por muitos a melhor base conceitual para definir a Ciência da Informação, há que ser repensado, tendo em vista as tendências que enfatizam os aspectos sociais, econômicos e culturais dos fluxos informacionais e da relação destes com a gênese de novos conhecimentos.

Bates (1999) atenta ainda para o fato de que as atividades da área se concentram na representação e organização da informação e não na compreensão de seu conteúdo. A especialidade dos profissionais da informação é muitas vezes ignorada quando pessoas inexperientes em questões informacionais gastam fortunas em pesquisa para redescobrir o que a Ciência da Informação sabe desde 1960.

Em uma crítica à definição clássica apresentada, Capurro (1985) afirma que aquela não identifica adequadamente o foco específico da Ciência da Informação, na medida em que nenhuma ciência deve ser definida por suas ferramentas, por exemplo, por ‘modernas tecnologias’, pois supõe-se que todos os campos devam se utilizar das mais avançadas tecnologias disponíveis. Ou seja, este atributo não pode se constituir na essência da ciência, que deve ser determinada por seu objeto de estudo, sendo, neste caso, a informação propriamente dita. Assim, faz-se necessário identificar o papel específico da Ciência da Informação no processo de geração, coleta, organização, interpretação, armazenamento, recuperação, disseminação, transformação e uso da informação.

Segundo Saracevic (1999), a Ciência da Informação trata basicamente da natureza da

informação e de seu uso por seres humanos. Como foi visto acima, originou-se do fenômeno da ‘explosão bibliográfica’ ocorrido após a 2ª Guerra, e teve como ideias iniciais a recuperação da informação e o conceito de relevância, associados à necessidade de informação dos usuários, e, duas décadas depois, a interação homem-máquina nos Sistemas de Recuperação da Informação (SRIs).

Saracevic (1999) divide a Ciência da Informação em duas grandes áreas ou sub-disciplinas: uma com foco na recuperação da informação; e outra com enfoque na comunicação e no uso da informação. Esta última, segundo Dias (2002), relaciona-se tanto à comunicação científica (com atividades que vão desde o momento em que o cientista tem uma idéia de pesquisa até a incorporação dos resultados da pesquisa no correspondente corpus de conhecimento) quanto à comunicação como um processo básico (que ocorre em todos os demais segmentos informacionais, ou seja, os sistemas informais).

Neveling e Wersig (1975) apresentam quatro enfoques para a Ciência da Informação, sendo estes: a visão orientada para o fenômeno (responsabiliza-se pelo fenômeno da informação); a visão orientada para os meios (é definida a partir da prática); a visão orientada para a tecnologia (é vista como um subsistema da Ciência da Computação); e a visão orientada para os fins (deve servir a determinadas necessidades sociais, desenvolvendo um trabalho prático para atendê-las).

Robredo (2003) considera a Ciência da Informação como o estudo, com critérios, princípios e métodos científicos, da informação, e seu objeto como a informação, em todos os seus aspectos e de todos os pontos de vista.

2.1.3 Natureza interdisciplinar

A natureza interdisciplinar da Ciência da Informação propicia o surgimento de diferentes correntes e estimula discussões sobre o seu objeto de estudo, a informação. Nesse ambiente, no qual se juntam conceitos de áreas diversas, a construção de conceitos interdisciplinares apresenta-se como um grande desafio. De todo modo, a sistematização da Ciência da Informação deve passar obrigatoriamente pela definição do conceito de “informação” (MARCONDES; CAMPOS, 2008).

É unanimidade na literatura que teorias, conceitos e métodos de outras disciplinas sejam utilizados para a compreensão dos problemas da Ciência da Informação. Devido sua natureza interdisciplinar, a Ciência da Informação foi definida sob diversos pontos de vista, influenciados pelo contexto da disciplina de origem de cada autor. Sofreu, portanto, influências na definição de seu objeto de estudo e de sua abrangência temática. Saracevic (1995) observa que seria inviável tentar solucionar os problemas da área por meio de uma única disciplina.

Bates (1999) ressalta que, em decorrência da complexidade dos problemas da Ciência da Informação, faz-se necessário um misto de metodologias para solucioná-los. E as abordagens qualitativas, com múltiplas perspectivas, trazidas pelas grandes mudanças decorrentes da pós-modernidade, enriqueceram e incrementaram a variedade de metodologias disponíveis para a compreensão do objeto de estudo da área. Nesse sentido, Saracevic (1995) ressalta que problemas complexos demandam abordagens interdisciplinares e soluções multidisciplinares (SARACEVIC, 1995).

Em um estudo acerca do caráter interdisciplinar da Ciência da Informação, Pinheiro e Loureiro (1995) apresentam os pontos de vista de diversos autores da área. Segundo eles, Foskett (1973) se refere à Ciência da Informação como:

[...] disciplina que surge de uma ‘fertilização cruzada’ de idéias que incluem a velha arte da Biblioteconomia, a nova arte da Computação, as artes dos novos meios de comunicação e aquelas ciências como Psicologia e Linguística, que, em suas formas modernas, têm a ver diretamente com todos os problemas da comunicação a transferência do conhecimento organizado (FOSKETT, 1973 apud GOMES, 1980, p.56).

O conjunto de disciplinas apontadas na literatura como estando relacionadas à Ciência da Informação é bastante diversificado, mas há concordância acerca da origem da área e de suas inter-relações com determinadas disciplinas na maioria dos casos.

Em definição clássica (BORKO, 1968) afirma que a Ciência da Informação, ao investigar as técnicas e os métodos que possibilitam a compreensão melhor das propriedades, do comportamento e do fluxo da informação, assume um caráter altamente interdisciplinar. Afirma ainda que a área deriva e se relaciona com campos como Matemática, Lógica, Linguística, Psicologia, Computação, Pesquisa Operacional, Artes Gráficas, Comunicação, Biblioteconomia, Administração e outros campos similares.

Em relação ao campo disciplinar da Ciência da Informação, (MACHLUP; MANSFIELD,

1983) afirmam que este emergiu da Biblioteconomia e da Ciência da Computação, e que envolve conceitos e técnicas como comunicação, classificação, bibliometria, transferência de informação (redes e telecomunicações), controle de acesso e regulação, comportamento de usuários e outros fatores humanos. Com base nos autores, Rayward (1994) conclui que a Ciência da Informação é realmente composta por fragmentos de outras disciplinas.

Saracevic (1995) afirma que a Ciência da Informação é interdisciplinar por natureza, e a relação com outras disciplinas vem mudando constantemente; está inexoravelmente conectada à tecnologia da informação; e além e acima da tecnologia, tem uma forte dimensão humana e social. O autor afirma ainda que a Biblioteconomia, a Ciência da Computação, as Ciências Cognitivas e a Comunicação são as disciplinas que mais influenciam a Ciência da Informação.

Para Fonseca (2007), a Ciência da Informação reúne conceitos e métodos de várias disciplinas, como a Biblioteconomia, a Ciência da Computação, a Engenharia, a Linguística e a Psicologia, para desenvolver técnicas e mecanismos de apoio ao tratamento da informação.

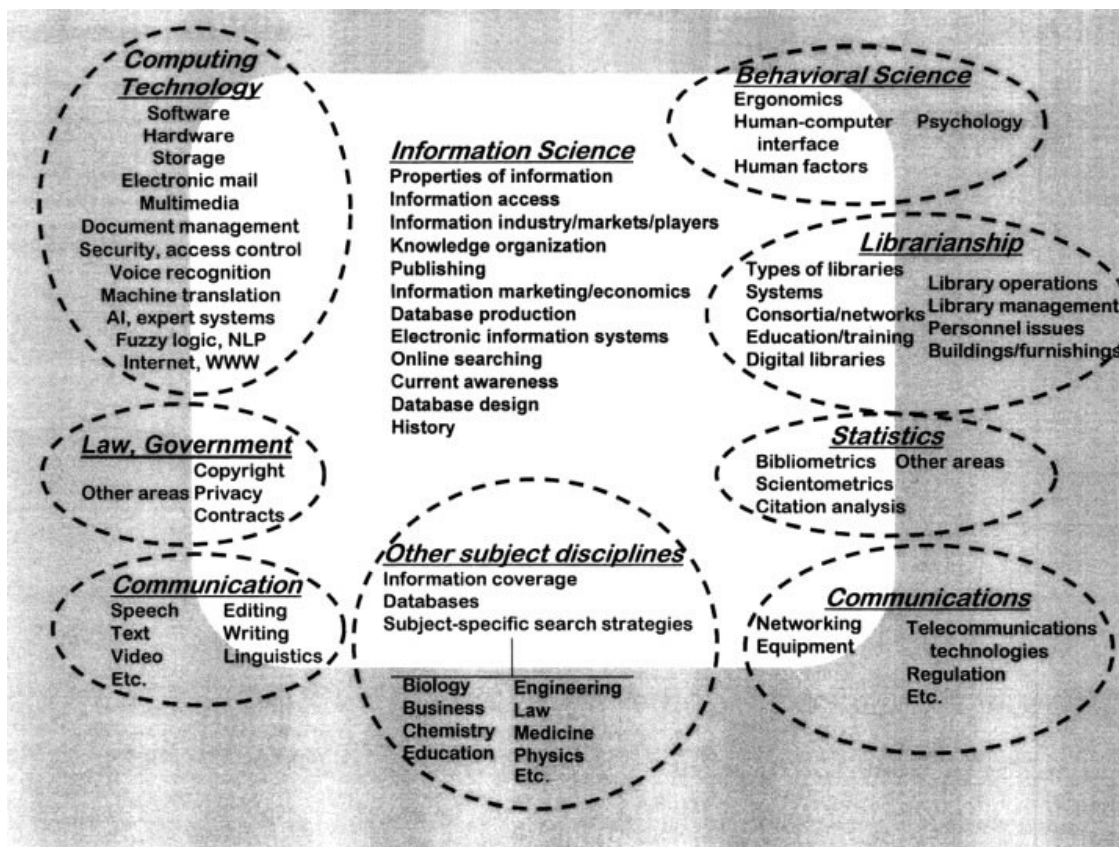
Com base em uma série de autores, (KOBASHI et al., 2001) identificam a Ciência da Informação associada à Biblioteconomia, e relacionada com a Arquivologia, a Museologia, além de manter interfaces com diversas outras ciências, como a Comunicação, a Computação e as Ciências Cognitivas.

Le-Coadic e Gomes (1996), por sua vez, afirmam que a Ciência da Informação tem suas origens na Biblioteconomia e se relaciona, principalmente, com os seguintes campos: Psicologia, Linguística, Sociologia, Informática, Matemática, Lógica, Estatística, Eletrônica, Economia, Direito, Filosofia, Epistemologia, História, Política e Telecomunicações.

Em dois estudos bibliométricos publicados no *Journal of the American Society for Information Science*, Hawkins (2001) caracteriza a interdisciplinaridade da Ciência da Informação e traça um mapa da área. Segundo o autor, o mapa se baseia (ver Figura 1) nas diversas definições encontradas na literatura, e em seu próprio conhecimento empírico da área, decorrente de longa experiência, como ressalta. No centro do mapa são listados os principais assuntos e sub-disciplinas da Ciência da Informação. As disciplinas ao redor do centro envolvem conceitos da área. A fronteira entre as disciplinas centrais e as externas perpassam através dos campos relacionados. Sub-campos relevantes para a área foram

posicionados no interior do esquema. O autor afirma que todos os assuntos centrais e muitos dos relacionados são próprios para o domínio de assuntos das publicações da área.

Figura 1 - Mapa da Ciência da Informação segundo Hawkins

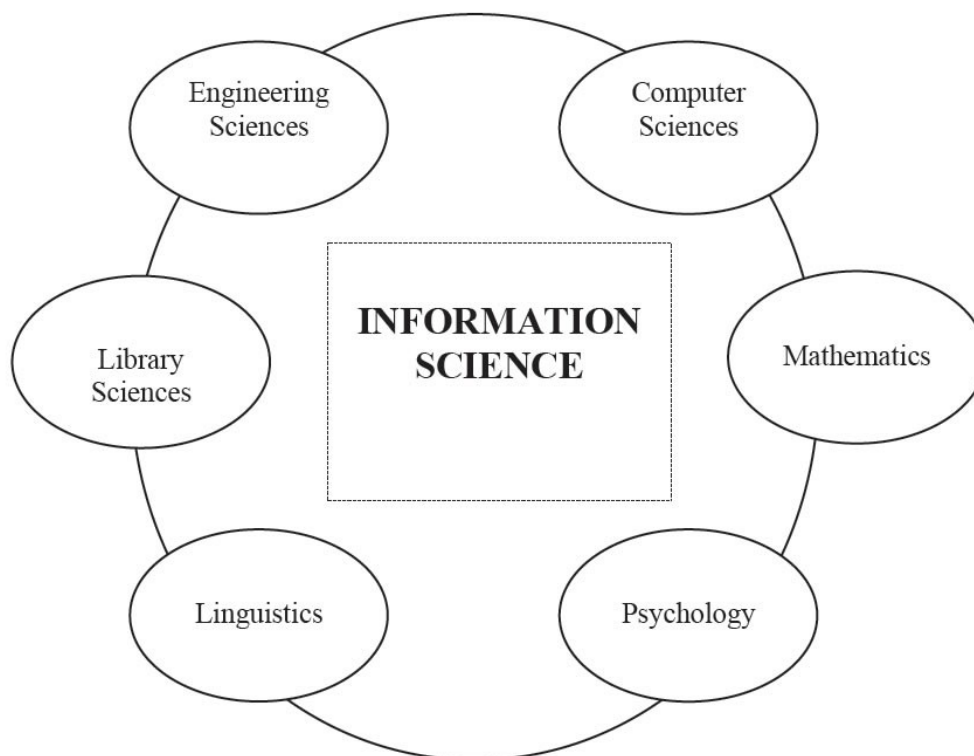


Fonte: Hawkins (2001, p. 8)

De acordo com Dragulanescu (2003) um dos mais importantes fenômenos para a evolução da Ciência da Informação é sua abertura a outras disciplinas, como: Informática, Eletrônica, Telecomunicações, Linguística, Matemática, Psicologia, Sociologia. A transgressão das fronteiras conduz a influências recíprocas e ao desenvolvimento de sinergias, que têm como consequência o enriquecimento mútuo. O autor apresenta em um quadro a estrutura interdisciplinar da área, descrevendo as disciplinas e os temas de pesquisa a estas relacionados, bem como os objetivos e resultados produzidos em cada uma.

Para representar a interdisciplinaridade da área, o autor ilustra, com base em Norton (2000), o diagrama apresentado na Figura 2 e ressalta apenas as principais disciplinas da área.

Figura 2 - Interdisciplinaridade da Ciência da Informação



Fonte: Norton (2000 apud MACEDO, 2005, p. 89)

2.1.3.1 Ciência da Informação e Ciência da Computação

De acordo com Saracevic (1999), a base da relação entre Ciência da Informação e ciência da computação reside na aplicação dos computadores e da computação na recuperação da informação, assim como nos produtos, serviços e redes associados. Como define Denning et al. (1989):

A disciplina da computação é o estudo sistemático dos processos algorítmicos que descrevem e transferem informações; sua teoria, análise, desenho, eficiência, implementação e aplicação. A questão fundamental subjacente a toda computação é: o que pode ser eficientemente automatizado? (DENNING, 1989, apud Saracevic, 1996, pág.50).

Como pode ser percebido, a ciência da computação trata de algoritmos que transformam informações enquanto a CI trata da natureza mesma da informação e sua comunicação para uso pelos humanos. Ambos os objetos são interrelacionados e não competidores, mas complementares. Eles levam a agendas básicas e aplicadas diferentes.

Na opinião do autor, a Ciência da Informação complementa a Ciência da Computação, na

medida em que esta fornece a infraestrutura e aquela o contexto. As relações com as Ciências da Computação dizem respeito aos aspectos práticos da área, haja vista que desenvolvem tecnologias para a recuperação e disseminação da informação, promovendo muitas das inovações em aplicações, tais como: sistemas de informação, sistemas especialistas, hipertexto, bases de conhecimento, interface inteligente e a interação homem-máquina (SARACEVIC, 1995 e PINHEIRO; LOUREIRO, 1995).

Um grande número de cientistas da computação estiveram proximamente envolvidos com pesquisa e desenvolvimento em recuperação da informação, assim como em muitos de seus desdobramentos, a ponto de serem reconhecidos como líderes em Ciência da Informação. Gerald Salton é um bom exemplo. Por outro lado, existe uma vertente de pesquisa e desenvolvimento em Ciência da Computação que pouca ligação tem com os estudos iniciais em Ciência da Informação. Essa inclui os trabalhos com sistemas inteligentes, bases de conhecimento, hipertextos e sistemas relacionados, interfaces inteligentes e interação homem-computador e mesmo reutilização de software. Essas áreas tem um significativo componente informacional, associado com a representação da informação, sua organização intelectual e encadeamentos; busca e recuperação de informação; a qualidade, o valor e o uso da informação - todos tradicionalmente tratados pela Ciência da Informação. De modo inverso, essa corrente de pesquisa e desenvolvimento na ciência da computação fornece diferentes visões, modelos e enfoques e um paradigma diverso para a pesquisa e desenvolvimento em Ciência da Informação. Portanto, as relações interdisciplinares estão evoluindo em direção a um nível diferente de cooperação intelectual. Aqui também, como no caso da biblioteconomia, este não é um argumento fortuito. Se existe um questionamento, esse deve se relacionar com a adequação do paradigma, isto é, a interrelação dos problemas e do enfoque das soluções.

2.1.4 A necessidade da Ciência da Informação

Segundo Saracevic (1999), o problema proposto pela Ciência da Informação é a tarefa massiva de tornar mais acessível um acervo crescente de conhecimento, assim como todos os problemas mais específicos que se seguiram, estão ainda à nossa volta e estarão aí com ou

sem a Ciência da Informação. A questão de se aplicar a tecnologia da informação na solução dos problemas informacionais continua e continuará com ou sem a Ciência da Informação. A evolução da ecologia informacional pode se processar com ou sem a Ciência da Informação. Existindo ou não um campo organizado chamado Ciência da Informação, os problemas não terminarão. Os problemas estão aí, independentemente de sua rotulação. As razões sociais para sua colocação são evidentes e urgentes. A questão é, então, quem vai direcioná-los, como e onde.

Nas últimas quatro décadas a Ciência da Informação apresentou contribuições que influenciaram o modo como a informação é manipulada na sociedade e pela tecnologia e também permitiu melhor compreensão para um rol de problemas, processos e estruturas associados ao conhecimento, à informação e ao comportamento humano frente a informação. A Ciência da Informação desenvolveu um corpo organizado de conhecimentos e competências profissionais ligados às questões informacionais. Os fatos demonstram a veracidade dessas conclusões.

Certamente, a Ciência da Informação não é o único campo que se ocupa com estas questões. Ela não detém o seu monopólio, como também não o faz nenhum outro campo. Entretanto, mudanças significativas estão ocorrendo em muitos campos pelo surgimento de problemas informacionais semelhantes, embora algumas vezes, de forma bastante diferente. Eventualmente, a questão é: que formas parecem ser mais promissoras no desvendamento do rol de questões informacionais? Como poderemos atingir maior aprofundamento no trato dessas questões? Essas são situações legítimas para o debate intelectual e profissional. (SARACEVIC, 1996).

Finalmente, não importa se a atividade que trata dessas questões seja chamada de Ciência da Informação, informática, ciências da informação, estudos de informação, ciências da computação e da informação, inteligência artificial, Ciência da Informação e engenharia, biblioteconomia e Ciência da Informação, ou qualquer outra forma, desde que os problemas sejam enfocados em termos humanos e não tecnológicos. Mas, a Ciência da Informação sob qualquer nome, significando um corpo organizado de conhecimentos e competências, teve e pode continuar tendo grande contribuição nesses estudos. Tem um registro comprovado de interdisciplinaridade. Sob qualquer nome ou patrocínio, as atividades profissionais e científicas desempenhadas pela Ciência da Informação são necessárias. Sobretudo, a necessidade dessa atividade organizada é crítica para a sociedade moderna. Ao se preencher

tal necessidade, a Ciência da Informação poderá ser melhor definida e reestruturada, como a sociedade requer.

2.2 Arquitetura da Informação

De acordo com (SIQUEIRA, 2010), a Arquitetura da Informação, quando considerada como disciplina científica, situa-se no contexto da Ciência da Informação e Documentação (LIS - Librarian and Information Science). A aplicação da Arquitetura da Informação viabiliza a redução do custo de acesso à informação, potencializando o seu valor para o usuário (BAYLE, 2003). No domínio da Ciência da Informação e Documentação o problema central não é o da Arquitetura - canonicamente entendida como ato de organização de espaços para o habitar humano (MONTEIRO, 2006), mas o problema da informação.

As origens da expressão 'arquitetura da informação', mostram a contribuição de Wurman (ele mesmo arquiteto), que comprovadamente cunhou a expressão em 1979 como uma afortunada metáfora que desenvolveu posteriormente com uma abrangente visão sistêmica em suas obras *Information Anxiety* e *Information Anxiety 2*, publicadas respectivamente em 1989 e 2001, e traduzidas no Brasil em 2001 e 2005. Destaca-se, finalmente, a apropriação da expressão 'arquitetura da informação' para aspectos relativos à interface usuário-sistema, nos sistemas de informação computadorizados e na Internet, por outros autores, com destaque para Morville e Rosenfeld, com sua obra de grande sucesso '*Information Architecture for the World Wide Web*' (3ª ed., 2000), os quais reconhecem que muito tem contribuído para o sucesso de suas idéias a própria experiência anterior no campo da Ciência da Informação.

Em sua obra *Information Anxiety 2*, (WURMAN et al., 2001) – cuja tradução foi publicada no Brasil em 2005 – declara, em duas rápidas passagens, ter introduzido a expressão 'arquiteto da informação' (*information architect*) em 1996, embora sem informar uma fonte precisa. Se hoje a opinião dos estudiosos parece corroborar de forma maciça essa declaração, é pelo menos curioso que as expressões *information architect* e/ou *information architecture* não apareçam na obra anterior *Information Anxiety*, (WURMAN, 1989), publicada no Brasil em 1991.

Segundo Robredo (2010), um estudo mais aprofundado da literatura induz a pensar que as expressões 'arquiteto da informação' e 'arquitetura da informação' introduzidas por Wurman

como uma afortunada metáfora – sendo ele mesmo arquiteto de grande sucesso, não só no desenho de casas, prédios e cidades, mas na construção a partir de seus elementos, de uma nova visão da informação como representação do conhecimento para sua difusão e uso – que incorporou rapidamente essas expressões ao vocabulário de um novo tipo de profissionais. A chave do sucesso de Wurman consiste em fazer entender como os ‘arquitetos da informação’ podem se abrir para a compreensão e o aprendizado, tornando-se capazes de explicar a informação aos outros. O sucesso dos trabalhos para entender a complexidade de grandes centros urbanos (Tóquio, Barcelona), o levou ao desenvolvimento de um novo tipo de mapas e guias inteligentes que ‘falam’ com os usuários usando inovadoras abordagens do graphic design. Um dos maiores sucessos de Wurman foi o Guia do Metrô de Tóquio.

Em 1984 Richard Saul Wurman e Harry Marks fundaram as conferências anuais denominadas *Technology Entertainment Design* (TEDs), que reúnem algumas das brilhantes mentes do planeta. O repositório em linha reúne mais de 50 milhões de slides das apresentações das conferências. A missão das TEDs foi definida assim: “idéias que merecem ser disseminadas”.

Mas, voltando à indagação sobre provas documentais, em (WURMAN et al., 2001), lembrando a marcante apresentação de Muriel Cooper – fundadora e diretora da Oficina de Linguagem Visual do Laboratório de Mídia do Massachusetts Institute of Technology (MIT) – por ocasião da TED de 1994, Wurman declara:

Embora sempre fosse fascinado pela informação e tenha encontrado meu caminho na vida através da informação, precisei ver o trabalho de Muriel para vivenciar o sonho de voar através da informação. Dediquei meu livro *Information Architects* a Muriel. (WURMAN, 2001, p.161).

Dessa forma, as expressões ‘arquiteto da informação’ e ‘arquitetura da informação’ se espalham com rapidez entre os novos profissionais embalados pelo dinamismo de empresas cada vez mais competitivas, os quais, frequentemente oriundos das escolas de Ciência da Informação, onde a ênfase numa forte fundamentação informática se generaliza, descobrem o potencial da Internet para criar uma interface que facilite a comunicação com os usuários e/ou clientes.

Surgem os novos designers (visual designers, information designers, experience designers, etc.) que aliam a modelagem inteligente de dados e informações (linguagens de marcação, HTML, XML, etc.) aos horizontes abertos pelas recentes tendências observadas na descrição e representação da informação e do conhecimento (metadados, FRBR, RDF, ontologias, etc),

e tudo isso num quadro onde o visual e a agilidade na comunicação nascem da compreensão dos processos e das pessoas, numa realidade que muda constantemente.

Neste ponto, parece oportuno lembrar o viés sofrido pelas acepções das expressões ‘arquiteto da informação’ e ‘arquitetura da informação’, na direção dos novos perfis dos designers, sob influência da rápida evolução da Internet, com todas as possibilidades abertas para uma ampla gama de profissões emergentes ou em acelerada evolução. (WURMAN et al., 2001) toma o cuidado de lembrar que é ele o criador dessas expressões no sentido de:

[...] arquiteto da informação: o indivíduo que organiza os padrões inerentes aos dados, tornando o complexo claro; a pessoa que cria a estrutura ou mapa da informação, que permite aos outros encontrar seus próprios caminhos na direção do conhecimento; a atividade profissional que surge no século 21 apontando para as necessidades da época, com foco na clareza, na compreensão humana e na ciência da organização da informação. (WURMAN, 1996, apud ROBREDO 2010, p.9).

O que, pensando bem, também inclui as atividades dos diversos tipos de novos designers. Para arquitetura da informação, reteremos a definição de Rosenfeld: “[...] a arte e a ciência de organizar a informação para ajudar as pessoas a satisfazer suas necessidades de informação de forma efetiva [...] o que implica organizar, navegar, marcar e buscar mecanismos nos sistemas de informação”. (MORVILLE; ROSENFELD, 2008, p.11).

Segundo esse autor, a arquitetura da informação faz parte da análise, do desenho e da implementação do espaço informacional “A arquitetura da informação é um campo de aplicação que se inspira em uma série de disciplinas.” (DILLON, 2002, p.2). Numerosas disciplinas contribuem para expansão da arquitetura da informação: psicologia organizacional, Ciência da Computação, educação, ciências cognitivas, design centrado no usuário, design gráfico e industrial, design instrucional, Web design, sociologia, antropologia, engenharia de software, modelagem de dados, gestão de bases de dados, interação usuário-computador, recuperação de informação e Ciência da Informação.

Recentrando o conceito original, Wurman (2005) lembra:

Quando criei o conceito e a expressão ‘arquitetura de informação’, em 1975, pensava que meus colegas de profissão iriam aderir em massa e passariam a se apresentar como ‘arquitetos de informação’. Ninguém fez isso – pelo menos não até recentemente. De repente, a expressão se generalizou. Como acontece com qualquer palavra que entra na moda, é natural que alguns

autodenominados arquitetos de informação correspondam à definição, mas há muitos que estão longe disso. Os verdadeiros arquitetos de informação dão clareza ao que é complexo; tornam a informação compreensível para outros seres humanos. Se conseguirem fazer isso, são bons arquitetos de informação. Se falharem, não o são. (WURMAN et al., 2005, p.37).

2.2.1 *Arquitetura da Informação e Web*

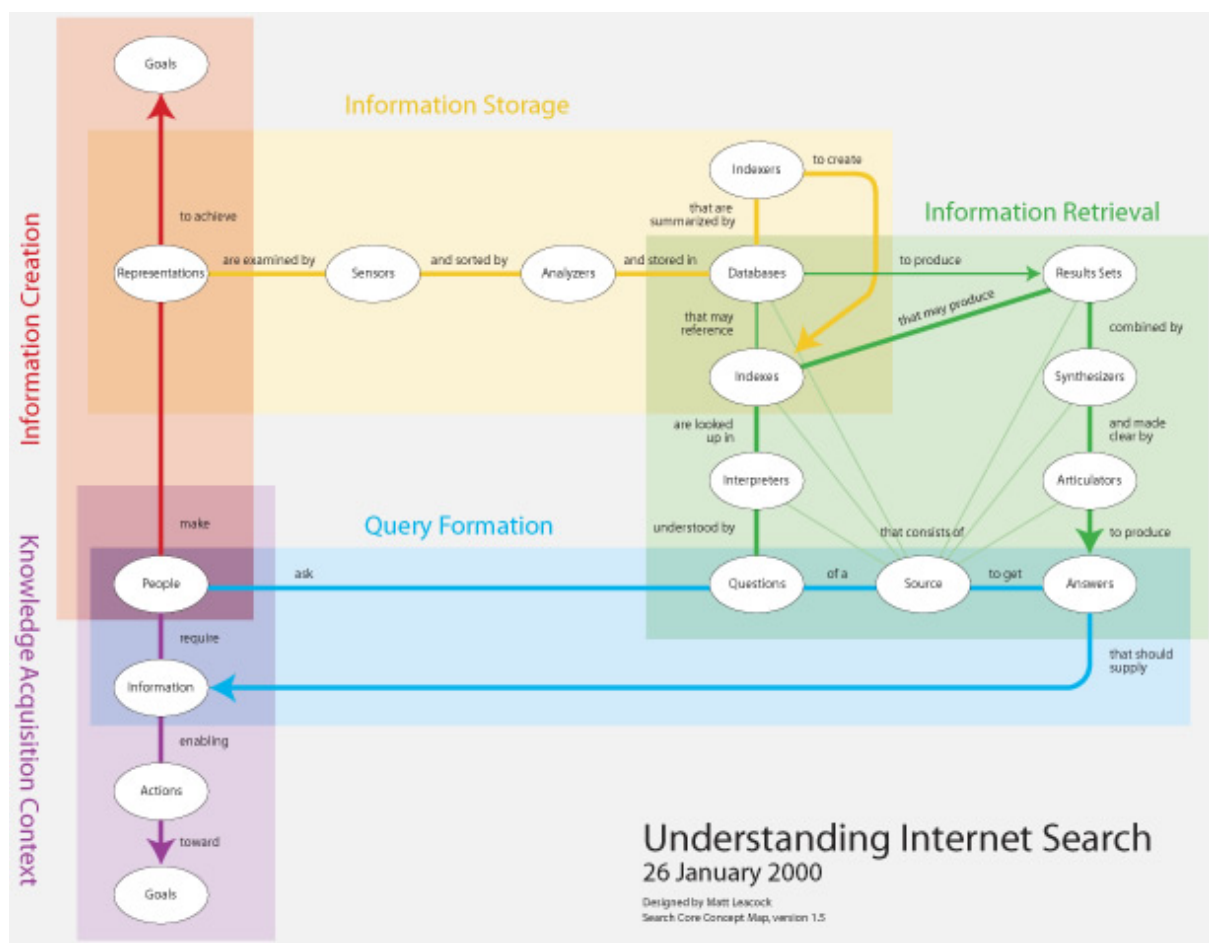
Um exemplo brindado por Wurman (2001) se refere ao mapa conceitual reproduzido sob o título (*Understanding Internet Search*), onde o processo de busca é estruturado em cinco blocos interligados: Criação da Informação (*Information Creation*), Contexto de Aquisição do Conhecimento (*Knowledge Acquisition Context*); Armazenamento da Informação (*Information Storage*); Formação da Pergunta (*Query Formation*) e Recuperação da Informação (*Information Retrieval*), com seus respectivos desdobramentos. Uma reprodução da figura colorida original, encontra-se na Figura 3.

A título de exemplo, em um trabalho de Rehman e Chaudhry (2005), se pode ler que das doze escolas de LIS (Library and Information Science) credenciadas pela ALA (American Library Association), que oferecem programas de pós-graduação em organização do conhecimento, sete dentre elas, incluem formação em arquitetura da informação.

Dentre elas, destacam-se nos Estados Unidos da América a Universidade de Michigan e a Universidade de Texas (Austin), e no Canadá destaca-se a *University of Western Ontario*.

Na Faculdade de Estudos de Informação e Mídia, da University of Western Ontário, onde existe a pós-graduação em nível de mestrado e doutorado (PhD in *Library and Information Science* e *Master of Library and Information Science – MLIS*), chama a atenção a grande quantidade de disciplinas optativas e de tópicos especiais. É entre estas últimas que se encontra a ‘Arquitetura da Informação’, com a seguinte ementa: “(...) trata da organização, ‘encontrabilidade’ e usabilidade dos ambientes informacionais, como os Web sites. Os estudantes deverão aprender a desenhar sistemas de navegação, interfaces de busca, fluxos de tarefas e outros elementos de arquitetura da informação. Não é um curso de tecnologia. Dar-se-á ênfase aos conceitos, técnicas, métodos e ao processo de design. Pré-requisitos: planilhas, software para apresentações, codificação básica em HTML e outras linguagens de marcação.” (*University of Western Ontario*, 2008)

Figura 3 - Mapa conceitual para compreender a pesquisa na Internet



Parece ser tendência associar a biblioteconomia e Ciência da Informação a outros domínios como gestão do conhecimento, informação em ambiente empresarial, inteligência empresarial e competitiva, comunicação e mídia, Web design, etc., sem esquecer determinados aspectos de fundamentação, tais como filosofia, teoria do conhecimento, metodologia da pesquisa, linguística, e também conhecimentos instrumentais como métodos quantitativos, estatística, estudos métricos da informação, linguagens e modelos para Internet, HTML, XHTML, XML, entidade-relacionamento, 'triplas', sujeito-predicado-objeto, etc., abrindo as portas à compreensão da Web semântica e de suas potencialidades para a transição da Internet como veículo de informação para veículo do conhecimento.

Observa-se que os conceitos e técnicas idealizados e desenvolvidos, nos anos 1970 e 1980, pelos sistemas de documentação e informação especializados, com o auxílio da emergente informática, são redescobertos – naturalmente com outras denominações e, reconhecamos, aprimorados – no avassalador bojo inventivo da Web. Exemplos: 'arquivo

invertido’ versus ‘reverse file’; ‘esquemas com setas’, na representação de relações entre termos de um tesouro versus ‘clusters temáticos’, ‘topic maps’, ‘ontologias’; ‘ID fixo’ (código que identifica um documento unívoca e permanentemente, na base de dados) versus URI (conjunto de cadeias de caracteres para identificar recursos – implementa as recomendações de diversas recomendações funcionais); ‘Folhas de entrada’ ou ‘Templates’ baseados em sistemas tipo CDS/ISIS, MARC, etc., com campos de dados identificados por etiquetas (tags), para entrada de dados versus codificação em HTML, XML, etc.; Formato Comum de Comunicação (CCF) e norma ISO 2709 (Formato de intercâmbio) versus interoperabilidade (ROBREDO, 2005).

Pelo exposto constata-se que a inclusão da arquitetura da informação, com todas as suas características de Web design e usabilidade, mas também como propiciadora e facilitadora da passagem da informação para o conhecimento e a compreensão, em harmonia com a pedra angular da Ciência da Informação, qual seja a representação e organização da informação e do conhecimento, abre e alarga os horizontes num futuro imediato.

Fecharemos estas reflexões com a última frase do trabalho apresentado pelos estudantes, no IX ENANCIB (ROBREDO, 2010):

“[...] a fundamentação da arquitetura da informação vem, de fato, da fundamentação da Ciência da Informação” (ROBREDO, 2010, p.17). O que, em outras palavras, significa “[...] a fundamentação teórica da arquitetura da informação tem seus pressupostos vindos dos pressupostos da Ciência da Informação, [...] porque a segunda contém a primeira”. (ROBREDO, 2010, p.17).

De fato, caberá aos futuros pesquisadores não só implementar essas mudanças, mas também explorar a metáfora da ‘Arquitetura da Informação’, que se apresenta como um novo desdobramento em aplicações cada dia mais Web, dos princípios da representação e organização da informação e do conhecimento. Nesse aspecto, a Ciência da Informação tem muito a contribuir para tirar a biblioteconomia e áreas afins – bem como a *Library and Information Science* (LIS) – do ‘buraco negro’ conceitual em que hoje se encontra aprisionada, sem encontrar o caminho que levaria a trazer o imenso conhecimento escondido em catálogos e bibliotecas à luz da Web semântica, via metadados, taxonomias e ontologias (ROBREDO, 2010).

2.3 A Web Semântica

A Web Semântica é uma nova visão da estrutura da Web, concebida em um artigo de Berners-Lee em 1998 (proposta pelo W3C em 2002). É destinada a aumentar as possibilidades desta última ao tornar seus recursos acessíveis às máquinas (SÁNCHEZ, 2011).

A falta de formalismo nas tecnologias da Web atual torna seu conteúdo classificável somente sintaticamente. Isto no sentido de que apenas o usuário humano pode identificar, selecionar e manipular os dados e recursos de maneira eficaz; devido ao fato que estes últimos não possuem as estruturas explícitas ou as informações formalizadas.

Portanto, o aumento da taxa de exploração automática de recursos da Web - que é o objetivo da Web Semântica - pode ser conseguido ao transformar o conteúdo destes últimos semanticamente compreensíveis pelas máquinas, associando as informações operacionais descritas de maneira formal.

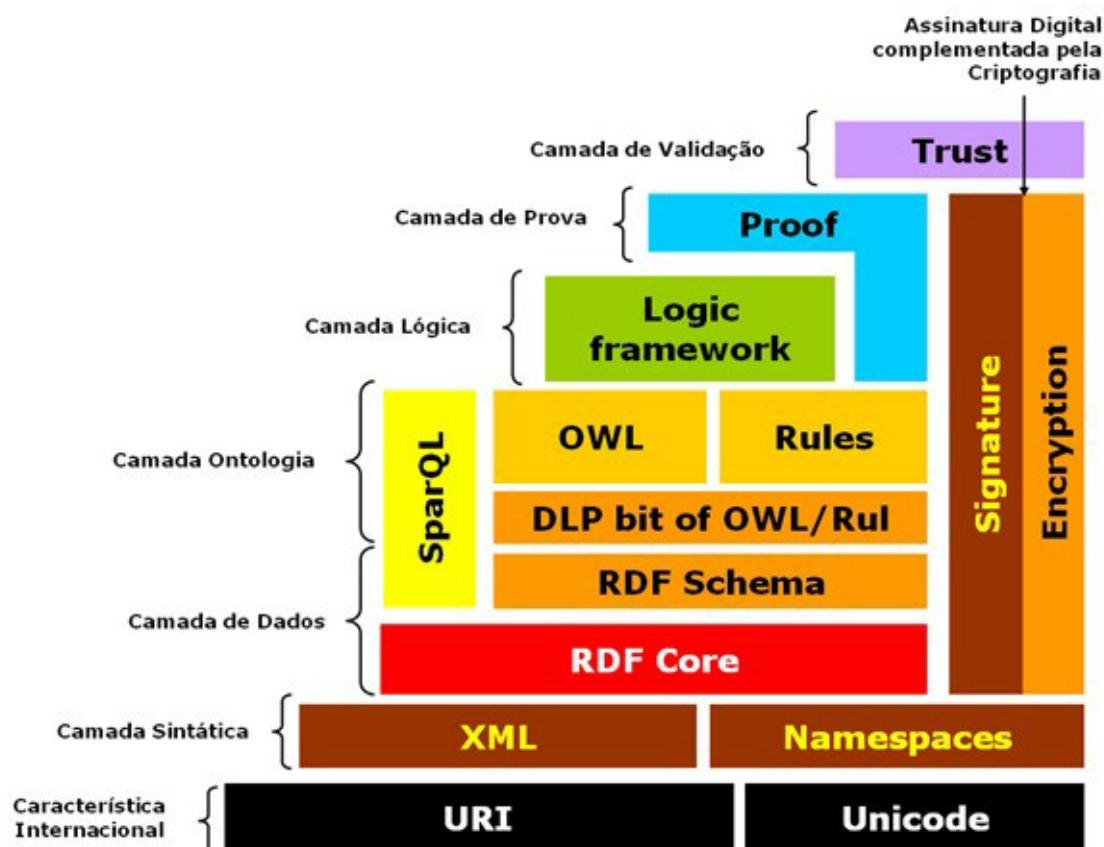
Mas limitar a Web Semântica à uma simples descrição de recursos não permite perceber sua verdadeira dimensão e as necessidades de sua implementação. Estas descrições formais deverão ser expandidas pelos mecanismos de explorações inferenciais para permitir uma manipulação “mais inteligente” dos conhecimentos, o que levará a um aumento significativo na velocidade de tratamento, devido a limitação - até mesmo a eliminação em certos casos - de intervenção humana.

Outras tecnologias de suporte podem operacionalizar esta infraestrutura, vindo de diversas áreas (Engenharia de Software, Cálculo Formal e Verificação de Programas, Integração e Bases de Dados) e demandando somente esforços de readaptação às especificidades do ambiente Web, tais como a distributividade, inconsistência de conhecimento e quantidade muito grande de recursos (CHARLET; LAUBLET; REYNAUD, 2003).

A criação de tal infraestrutura permitirá uma manipulação mais ergonômica dos recursos e conhecimentos da Web por seus usuários, e personalizará os diferentes métodos de interação que estes últimos serão levados a manipular. Da mesma forma que um aumento qualitativo e de volume da operação de informações e dos serviços mais variados (ALLEMANG; HENDLER, 2011).

2.3.1 Apresentação da Estrutura da Web Semântica

Figura 4 - Camadas da Web Semântica



Fonte: Santos e Alves (2011, p.7)

O estabelecimento de uma estrutura oferece as premissas necessárias para a realização de tais objetivos (automatização dos tratamentos da Web). Ela foi elaborada através da colaboração de diferentes organismos de diversas origens: 1. Comercial - empresas tais como IBM, HP, Microsoft etc.; 2. Pesquisa - Universidade de Stanford, MIT, Manchester, INRIA etc.; 3. Normalizadores tais como W3C ou ISO; e uma extensa padronização de diferentes tecnologias utilizados na sua elaboração. (CHARLET; LAUBLET; REYNAUD, 2003).

Sob essa ótica, ao W3C foi dada a tarefa de coordenar os diferentes esforços propostos, como a estrutura mostrada na Figura 4, (CHARTON; GAGNON; OZELL, 2010).

Embora apenas suas camadas inferiores apresentem uma certa estabilidade, esta estrutura apresenta várias vantagens, sendo que as mais importantes vem de sua definição em camadas que permite a utilização gradual das tecnologias (de um ponto de vista das

necessidades e vantagens). Também oferece mecanismos para (CHARLET; LAUBLET; REYNAUD, 2003):

A identificação de recursos

As *Uniform Resource Identifier's* (URI's) permitem a designação e a manipulação dos recursos da Web Semântica de maneira homogênea, oferecendo um sistema de indexação dos mais eficientes e aceitos.

As URI's poderão, pelo fato de que eles associam seus recursos às diferentes informações rudimentares necessárias ao seu funcionamento (protocolos de comunicação, informações operacionais) permitir uma utilização mecânica destes recursos. Além disso, elas permitem a indexação de entidades externas à rede (usuários, sistema financeiro, cartão inteligente, objetos reais etc.) e assim, descrever suas interações com as aplicações da Web (FENSEL, 2005).

Representação do conhecimento

A incorporação da semântica na Web volta a descrição dos seus recursos e de suas interações e propriedades de maneira a acessar as manipulações mais ou menos automatizadas seguindo o grau de formalismo das descrições.

Por isto, várias linguagens - essencialmente implementadas em XML - foram propostas para a expressão de anotações e de relações simples entre recursos (cujas linguagens principais são *Resource Description Framework* (RDF) e *Resource Description Framework Schema* (RDFs)) permitindo a criação de uma “rede” semântica entre os diferentes recursos da Web e das entidades que interagem com elas (os identificadores com seus URIs) e isto com o objetivo de aplicá-los às manipulações simples, tais como a pesquisa de informações ou a personalização de conteúdo.

Mas uma utilização mais intensa utilizará linguagens mais complexas, que possam oferecer um poder expressivo maior a medida que são mais formais. OWL (*Ontology Web Language*) é a representante desta categoria de linguagens: nascida de pesquisas sobre a representação de conhecimento e mais precisamente das Lógicas de Descrição e de Frames. Ela permite a modelagem de objetos que interagem em um dado domínio e sua implementação com base em conhecimento, chamadas nesse contexto “ontologias” (ALLEMANG; HENDLER, 2011).

Tais conhecimentos podem ser utilizados depois pelos motores de inferência, para a elaboração de técnicas de utilização mais eficientes, utilizáveis, por exemplo, na integração de dados heterogêneos ou na pesquisa e na combinação automática de serviços Web (CHARLET; LAUBLET; REYNAUD, 2003).

Exploração do conhecimento

A recuperação da camada de representação do conhecimento (ou descrição de recursos) por uma camada inferencial permitirá, devido a utilização de bases lógicas das técnicas de representação, a aplicação de regras de dedução para a geração de novos conhecimentos; isto através da combinação de outros conhecimentos, geralmente distribuídos, heterogêneos e até mesmo compostos (BELHADEF; KHOLLADI, 2007).

Além disso, a utilização de descrições formais de recursos (metadados, ontologias, etc.) permitem, devido seus fundamentos teóricos, a aplicação de tratamentos podendo em certos casos limitar e até mesmo inibir a intervenção humana, aumentando assim o rendimento das aplicações como as de tempo real, como os sistemas comerciais ou de Bolsa de valores, que devem gerar as flutuações de parâmetros heterogêneos, e isto o mais rápido possível, e aplicar os tratamentos *d'expertise* para tirar conclusões. (CHARLET; LAUBLET; REYNAUD, 2003).

Prova e nível de confiança

Para que tais manipulações possam ser aceitas e validadas pelos usuários, diferentes mecanismos de “explicação” e de provas de raciocínio devem ser implementadas. Uma máquina deve poder justificar suas escolhas, sobretudo em um ambiente versátil como a Web no qual as hipóteses contraditórias podem coexistir. Da mesma forma, a confiabilidade nas premissas do raciocínio (argumentos) devem claramente ser estudadas (premissas obtidas de diferentes fontes, de nível de confiança e de integridade geralmente muito variadas). Por isto, diferentes técnicas podem ser adotadas tais como as redes de confiança ou a autentificação usando assinaturas digitais (MCGUINNESS; SILVA, 2004; CHARLET; LAUBLET; REYNAUD, 2003).

2.3.1.1 A identificação de recursos - as URIs

2.3.1.1.1 Definição

No espaço de informações que é a Web, a capacidade de identificação e de endereçamento dos recursos é o primeiro passo para sua exploração. Seguindo este ponto de vista, as *Uniform Resources Identifier* (URIs) propostas por Engelbart em 1990 (FENSEL, 2005), são o sistema de indexação de recursos da Web de amanhã.

As URIs são as “cadeias de caracteres” que permitem a identificação e o acesso aos recursos de diferentes tipos (CHARLET; LAUBLET; REYNAUD, 2003), tais como:

- Documentos de textos e hipertexto;
- Arquivos para download;
- Serviços Web, agentes e diversas aplicações;
- Caixa de mensagens;
- Outras - as URIs são extensíveis.

E isso sob diferentes esquemas de protocolos (http, File Transfer Protocol (FTP) etc.). As URIs têm a seguinte forma global: “Esquema URI: descritor de recurso”. O esquema URI permite especificar os formatos de recursos, assim como seus protocolos de troca e métodos de exploração. O descritor de recurso permite identificar um recurso de maneira única (FENSEL, 2005). Exemplo de URI:

- mailto: joao@exemplo.org (especifica a caixa de mensagem de João)
- ftp://exemplo.org/aDiretorio/aArquivo (especifica o arquivo aArquivo que pode ser visualizado através de um servidor ftp)
- news: comp.infosystems.www (fonte de tipo de informação)
- fone:+9-816-555-1212 (fonte de tipo de número de telefone)
- my://exemplo.org (fonte de tipo personalizado ‘my’)

As URIs permitem identificar os recursos no sentido mais amplo:

- Recursos informativos: podem ser descritos, armazenados e enviados de acordo com um modelo informacional, como os documentos de texto, as imagens, as mídias, as aplicações etc.;
- Recursos não informativos: as entidades não se localizam sobre a rede, tais como os

seres vivos (um autor de uma página Web), os documentos impressos (uma nota de 100 reais) ou objetos (um carro).

O último ponto fornece as formas de representar a interação do mundo real com as diferentes aplicações da rede (agentes lógicos, serviços Web etc.), como a digitação de um número de cartão de crédito por um usuário.

2.3.1.1.2 Porque as URIs?

A utilização das URIs como padrão de indexação para a Web Semântica foi motivada por numerosas vantagens, por exemplo (BELHADEF; KHOLLADI, 2007):

- A aceitação da tecnologia URIs é a primeira vantagem desta escolha porque é primordial uniformizar, a nível mundial, as tecnologias de identificação para o compartilhamento de recursos (objetivo inicial da Web) e isto entre usuários e máquinas (objetivo final da Web Semântica).
- As URIs, por sua forma, permitem uma interpretação intuitiva da natureza dos recursos procurando flexibilidade de utilização pelo homem.
- A possibilidade de definir novos esquemas de recursos e permitir uma extensão do conjunto de recursos que podem ser indexados, e então explorados pelas máquinas que serão capazes de acessar estes dados e poder associá-los à especificações de aplicativos.
- As URIs oferecem a possibilidade de identificar os agentes lógicos - tais como os serviços Web especialistas, os servidores Proxy ou dos portais - e chamar os serviços de maneira automática.
- É possível definir as propriedades sobre as URIs devido as linguagens e mecanismos de modelagem e raciocínio, permitindo assim a definição de propriedades entre recursos, como a equivalência, por exemplo. Mas, o nível de confiança de tais “deduções” continua a ser assunto de pesquisa.

As qualidades inegáveis das URIs fazem delas uma ferramenta indispensável à infraestrutura da Web Semântica, mas uma melhor especificação de sua semântica permitirá a

resolução do problema da múltipla identificação de recursos. Por exemplo: “representar o site da UnB ou a própria universidade?”. Sem a utilização de controle humano, este tipo de ambiguidade corre o risco de conduzir à situações indecidíveis (HENDLER; BERNERS-LEE, 2010).

2.3.1.2 Descrição de recursos - Metadados e Ontologias

A Web dos dias atuais não permite uma abordagem sintática dos documentos e mídias, a maior parte dos tratamentos que lhe são aplicados não ultrapassam o domínio procedural. E embora os meios de indexação e referência de recursos, tais como as URIs, certamente facilitam sua utilização, ainda são insuficientes para resolver este problema. Uma utilização menos rígida deve ser considerada (ALLEMANG; HENDLER, 2011).

Para isso, é necessário redefinir os princípios da Web sob o ponto de vista da representação de recursos de seus métodos de utilização, sem forçar uma transformação radical da infraestrutura atual (BERNERS-LEE, 2000), pois isto corre o risco de ser impossível sob a ótica operacional e financeira. Trata-se então de cobrir a Web com uma camada semântica que permitirá este gênero de tratamentos. No que diz respeito a representação de recursos, pode-se essencialmente distinguir duas linhas principais de desenvolvimento (HORROCKS et al., 2004):

- A expressão de dados e metadados;
- Passagem para a noção de conhecimento formalizado por meio da definição de bases de conhecimento padronizadas (ontologias).

A seguir, serão expostos estes diferentes conceitos, e discutidos os desenvolvimentos atuais, as ferramentas utilizadas e, por último, exemplos práticos de suas aplicações; utilizando para isto a estrutura da Web Semântica definida no modelo do W3C.

Percebe-se que as linguagens propostas oferecem uma sintaxe em Extensible Markup Language (XML), que é a linguagem escolhida para codificar o conjunto - ou pelo menos a maioria - dos dados armazenados ou trocados por meio das diferentes partes que interagem com a plataforma da Web Semântica.

2.3.1.2.1 Metadados e linguagens de notação

a) Definição e utilização

Os metadados são as estruturas de dados que fornecem uma informação útil sobre outros dados. Eles permitem, entre outras coisas (CHAMPIN, 2001):

- A anotação de documentos (ou de fragmentos de documentos) e o estabelecimento de catálogos segundo os critérios: 1. Descritivos: tipo, domínio da aplicação, critérios de gestão (direitos de acesso, licença, propriedade intelectual etc.); 2. Operacionais: descrevem a forma de utilização.
- Definem os direitos de acesso, as informações de autenticação e as assinaturas eletrônicas.
- Estruturação de dados e avaliação do conteúdo permitindo a melhoria dos mecanismos de busca e a extração de informações.
- Descrição de dados de natureza bastante variada: é possível anotar os recursos, as partes de recursos e inversamente, as coleções de recursos. É possível também estabelecer uma hierarquia de metadados e assim estabelecer os metas sobre os metas.

Porém, embora os metadados sejam uma ajuda preciosa nos processos de estruturação, de catálogos e de pesquisa de documentos através da rede, eles são dados adquiridos, mas longe de serem “compreendidos” por qualquer agente de software (ALLEMANG; HENDLER, 2011).

b) Metadados e Web Semântica

Os metadados, por causa de seu aspecto semi-independente dos recursos que eles anotam, são perfeitamente adaptados à visão evolutiva da Web Semântica. Eles apresentam a vantagem de serem compartilháveis e de oferecer uma descrição estrutural e cognitiva dos recursos.

E é precisamente na Web Semântica que os metadados apresentam um aspecto bastante relevante. A combinação de técnicas de anotação tradicionais e das contribuições da

semântica formal permitem a anotação de documentos com os esquemas e metadados. Desde a representação do Dublin Core aos das ontologias, o tratamento foi irremediavelmente melhorado: a pesquisa e recuperação das informações são simplificadas e fornecem resultados mais pertinentes (MILES et al., 2005).

Estes metadados podem servir aos agentes de software como dados durante um processo de inferência, eles podem até mesmo ser utilizados para criar outros metadados devido aos mecanismos de composição de dados, ou permitir a composição de serviços de acordo com suas descrições a fim de aumentar o grau de automatização.

Vários trabalhos já foram utilizados nesse sentido, além disso as normas foram estabelecidas seguindo os diferentes aspectos relativos aos sistemas de anotação, que são (ALVES, 2005):

- As normas de concepção do conteúdo Web: HyperText Markup Language (HTML) (Marcação, Uniform Resource Locator (URLs) etc.), Extensible Markup Language (XML), XHTML.
- As linguagens de anotações (anotação textual) estruturadas como é XML ou uma linguagem de descrição de recursos como RDF;
- Os esquemas de metadados como o Dublin Core ou as linguagens de descrições de ontologias DARPA agent markup language (DAML)+OIL, OWL etc;
- As diferentes pesquisas sobre as técnicas de armazenamento (armazéns de metadados, bases de dados etc.) e troca de metadados (adoção de protocolo Hypertext Transfer Protocol (HTTP) para a transmissão).

c) RDF - *Ressource Description Framework*

RDF apareceu em 1997, e foi então utilizado como um padrão W3C para a descrição de metadados, e desde então foi generalizado como formalismo de anotação onde a função é afirmar diferentes tipos de relações entre os recursos. Para isso, RDF possui várias vantagens: por exemplo, dispõe de uma sintaxe XML e de espaços de nomes (BERNERS-LEE, 2013).

Além disso, utiliza o esquema das URIs para definir os recursos e não existe restrições sobre a natureza do que pode ser descrito, quer dizer, é possível descrever qualquer coisa utilizando RDF, o que lhe proporciona uma grande flexibilidade, seja sintática (sua

escalabilidade, que provem em grande parte da estrutura XML) ou semântica (independência da estrutura em relação ao contexto). Além da sua relação com XML, RDF pode ser representável por dois formalismos importantes: a Lógica de Primeira Ordem e a Teoria de Grafos.

Antes de ir mais longe nos detalhes e as vantagens dessas reconciliações, é necessário primeiro apresentar a estrutura de RDF, que possui (BERNERS-LEE et al., 2001):

- Princípios fundamentais da linguagem RDF

O elemento raiz de um documento RDF é a declaração. Um documento RDF contém uma ou mais declarações ou afirmações. Uma declaração está na forma de uma tripla <Sujeito, Predicado, Objeto> cujos elementos Sujeito e Predicado são as URIs, o elemento Objeto pode ser uma URI ou um valor. A tripla pode ser lida: “O Sujeito possui o valor Objeto para a propriedade Predicado”. As URIs podem ser externas (“http://www.mySamples.org/John”) ou locais (avaliados: “#nome”). Os valores são os recursos ou as sequências de caracteres (os valores fixos) (FERRÉ; HERMANN et al., 2011).

O seguinte exemplo auxilia a ilustrar o grau da representatividade de RDF. Seja a declaração: *Clyde works at Sun, she is 35 years old and is married to Bonnie*. Esta frase pode ser traduzida em triplas RDF: <Clyde, Works, Sun Micro Systems > <Clyde, Aged, 35 > <Clyde, HasWife, Bonnie>.

Esta representação é facilmente traduzível em um conjunto de fórmulas atômicas do tipo Predicado (Objeto, Sujeito). É possível assim representar um documento RDF por uma fórmula de conjunções positivas existenciais de primeira ordem cujos elementos (Sujeito, Objeto) representam as constantes no caso de URIs ou literais, ou outras variáveis.

Representar assim os documentos RDF tornará possível o uso de mecanismos lógicos para eventuais deduções ou de provas de consistência: É possível afirmar, por exemplo, que as informações de um documento R são incluídas em um documento R' se e somente se a fórmula lógica correspondente ao documento R é uma consequência lógica da fórmula de R'. Isto pode ajudar na pesquisa de informações ou na verificação da coerência (BAGET et al., 2004).

Em termos do nosso exemplo, podemos colocar como segue: “Trabalha(Sun,Clyde) E Aged(35,Clyde) E HasWife(Bonnie,Clyde)”. Isto também é expresso na sintaxe XML:

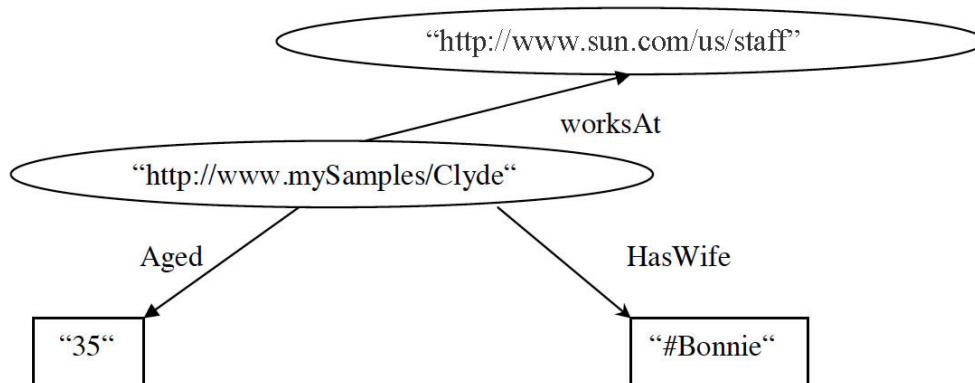
```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:sample=
"http://www.mySamples.org/sample">
  <rdf:Description rdf:ID="Bonnie">
    <rdf:Description about="http://www.mySamples.org/Clyde">
      <sample:Aged>"35"</sample:Aged>
      <sample:WorksAt>"http://www.sun.com/us/staff"
      </sample:WorksAt>
      <sample:HasWife>"#Bonnie" </sample:HasWife>
    </rdf:Description>
  </rdf:RDF>
```

Outras sintaxes XML podem ser utilizadas, dentre elas a seguinte sintaxe abreviada:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:sample=
"http://www.mySamples.org/sample">
  <rdf:Description rdf:ID="Bonnie">
    <rdf:Description about="http://www.mySamples.org/Clyde">
      <sample:Aged resource="35"/>
      <sample:WorksAt resource="http://www.sun.com/us/staff"/>
      <sample:HasWife resource="#Bonnie"/>
    </rdf:Description>
  </rdf:RDF>
```

Um documento RDF pode igualmente ser representado por um grafo orientado cujos nós são os recursos e onde os arcos representam as relações entre eles. Esta representação é bastante relevante, pois torna possível tirar proveito das técnicas de exploração e de extração definidas na teoria dos grafos e apresentam a vantagem de ser mais ergonômica e mais representativa da rede de conhecimentos e relações (ABITEBOUL; BUNEMAN; SUCIU, 2000). A Figura 5 mostra um pequeno exemplo:

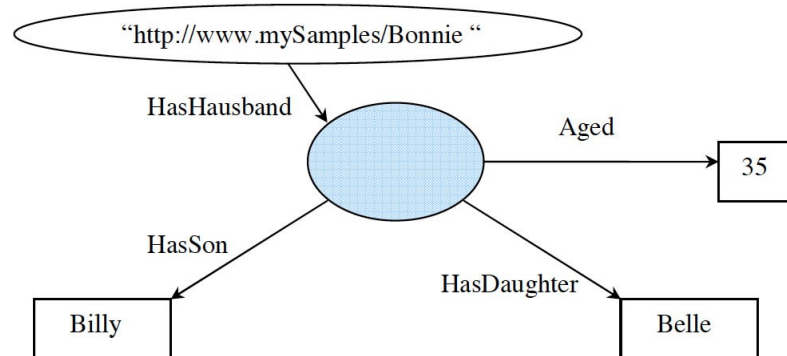
Figura 5 - Grafo orientado representando RDF



Fonte: elaborada pela autora

Além disso, RDF permite ultrapassar certas restrições em favor de uma maior expressividade. É possível definir os nós anônimos para permitir a representação de tal declaração: *“Bonnie's husband is one who is 35 years old and has a son Billy and Belle for daughter”*. A Figura 6 mostra o que é possível representá-la através de um grafo “em branco”:

Figura 6 - Grafo “em branco”



Fonte: elaborada pela autora

Existe também uma ferramenta poderosa do ponto de vista expressivo. Uma declaração podendo ela própria ser identificada por uma URI. É possível então construir declarações envolvendo outras declarações: É o princípio da “reificação - coisificação”. Devido a isto, é possível expressar declarações mais complexas, como: *“Clyde thinks that his son had the bachelor”*. Será decomposto em triplas RDF (UDREA; RECUPERO; SUBRAHMANIAN, 2010):

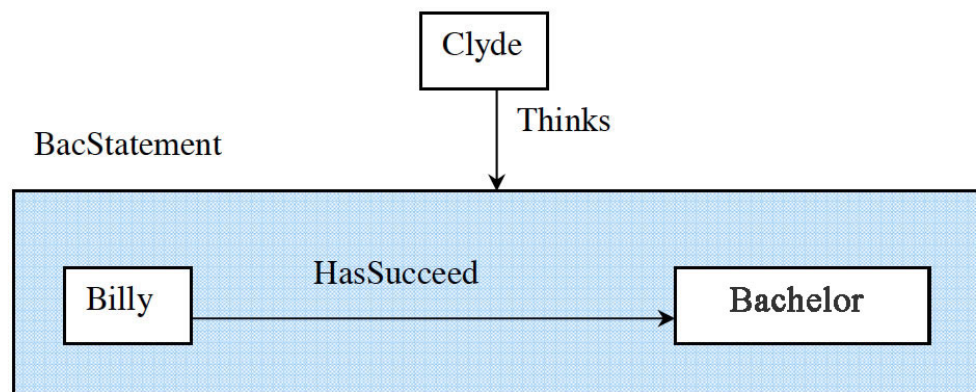
```

<#bacStatement, rdf : subject, #Billy>
<#bacStatement, rdf : predicate, sample: Has Succeed>
<#bacStatement, rdf : object, #Baccalaureate>
<#Clyde, sample : Thinks, # BacStatement >

```

A Figura 7 mostra o grafo com triplas.

Figura 7 - Grafo com triplas



Fonte: elaborada pela autora

- Tipos de base

A liberdade que oferece RDF sobre a utilização de URIs necessita de meios para diferenciá-las, como as palavras-chave:

- `rdf : Recurso` : Identifica a URI como sendo um recurso.
- `rdf : Propriedade` : Afirma que a URI define uma propriedade.
- `rdf : Elas` : A URI em questão é uma declaração.

A declaração é feita usando a palavra-chave “`rdf:tipo`”.

Exemplo:

```

<#BacStatement, rdf : tipo, rdf : Statement>

```

A combinação de tipos é permitida (como no caso de uma reificação em que a afirmação é também um recurso).

- Tipos Complexos

Existem dois tipos compostos (BRUIJN; WELTY, 2010):

- Contêineres: são do tipo “rdfs : Contêiner” e se dividem em três sub-classes:
 - rdf : Bag: Agrupa um conjunto de recursos. Os membros da coleção são designados por: rdf : _1, rdf : _2 etc.
 - rdf : Sequência: Indica uma sequência de recursos.
 - rdf : Alt: Apresenta uma lista de recursos.

Exemplo:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:example="http://www.mySamples.org/sample">
  <rdf : Description about = "http://www.mySamples.org/Clyde">
    <sample:hasProjects>
      <rdf:Bag>
        <rdf:_1 rdf:resource="http://www.Sun.com/Projects/onlinePDA"/>
        <rdf:_2 rdf:resource="http://www.Sun.com/Projects/Java3"/>
        <rdf:_3 rdf:resource="http://www.Sun.com/Projects/JRE2007"/>
      </rdf:Bag>
    </sample:hasProjects>
  </rdf:Description>
</rdf:RDF>
```

- As listas: Define-se um recurso como sendo uma lista utilizando “rdf : Lista, que se beneficia dos construtores de listas usuais descritas por :

« rdf :first, rdf :rest, rdf :nil »

O ganho proporcionado pelos tipos complexos não é expressivo, somente denotacional.

- RDF Schema

RDFS (NOY; MUSEN, 2000) é uma extensão de RDF, que surgiu essencialmente para superar a falta de precisão em relação aos tipos e a hierarquização dos recursos. RDFS

propõe mecanismos de definição de parentesco entre recursos e restrição nos seus domínios de valores. Da mesma forma que XMLS fez com XML, RDFS amplia o vocabulário RDF permitindo a definição de novos tipos de propriedades e classes específicas aos domínios de trabalho. A estruturação através dos schemas permite também verificar a validade dos documentos.

- *As classes*

As classes em RDFS são definidas pela notação (BRUIJN; WELTY, 2010):

```
<nomClasse, rdf:type,rdfs:Class>
```

Para o elemento `nomClasse` (“example:Guitarra”, por exemplo), é possível definir as sub-classes desta maneira:

```
< example:Fender, rdfs:subClassOf, example:Guitarre>
```

Todas as instâncias dessa classe herdam as propriedades da classe mãe, por exemplo:

```
<example, rdf:type, sample:Fender>
```

Stratocaster é uma instância da classe ‘Fender’ e possui as propriedades de suas classes superiores (Fender - guitarra elétrica, caixa cheia etc., Guitarra - ter uma caixa, possuir cordas etc...).

- *As propriedades*

Todas as propriedades são instâncias da classe geral “Propriedade” e são, por consequência, os recursos do tipo “rdf:Property”. As primitivas RDFS de tratamento sobre as propriedades são:

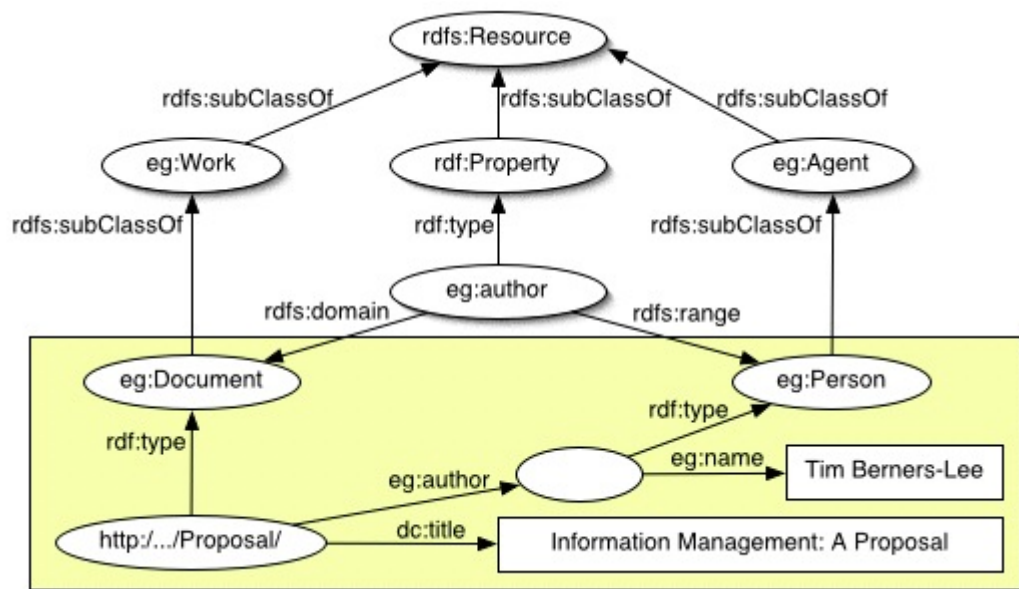
- `rdfs:subPropertyOf`: usado para definir uma sub-propriedade.
- `rdfs:domain`: Restrições para definir o escopo de uma propriedade.
- `rdfs:range`: Define o co-domínio de uma propriedade, se uma propriedade pode ter um ou vários domínios, ela pode ter, entretanto, um só co-domínio ou mais.

A Figura 8 mostra um exemplo de propriedade RDF, e a Figura 9 um pequeno resumo das classes e relações no RDFS.

Para concluir, RDF se apresenta como uma ferramenta poderosa, que oferece mecanismos

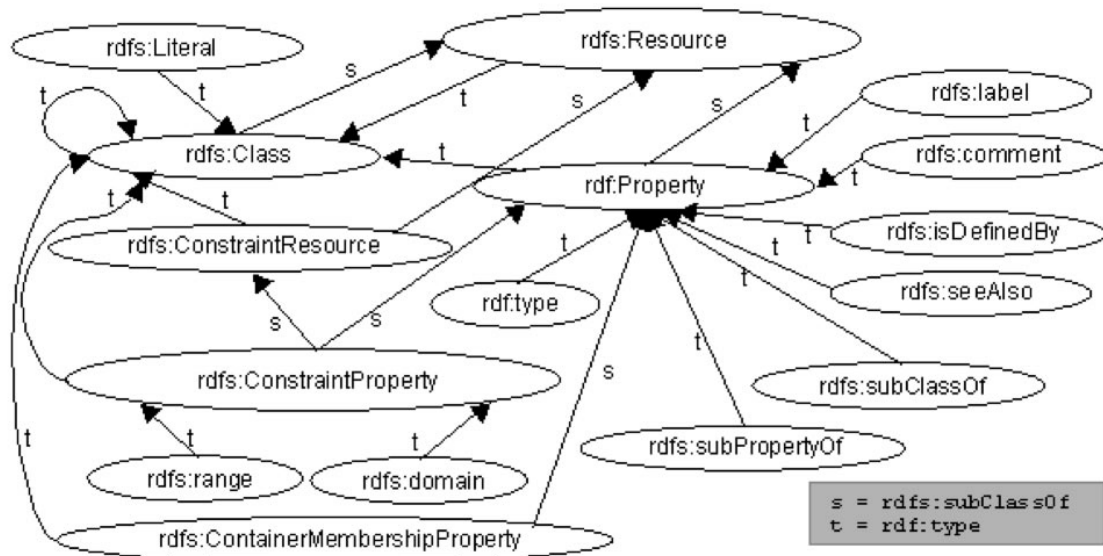
para a representação de recursos heterogêneos e proporcionando um alto nível de interoperabilidade e representatividade. As linguagens como RQL servem para trazer meios de expressão de consultas. RDF é uma linguagem que tem sido comprovada e numerosas aplicações utilizam suas funcionalidades, por exemplo: o Dublin Core, P3P, Labels PICS e também sites Web como XMLfr (CHO; ROY; ADAMS, 2005). Entretanto, RDF é muito limitado em relação à representação de objetos de natureza cognitiva (ontológica). Dois documentos podem ter estruturas diferentes e contudo, representar a mesma informação.

Figura 8 - Exemplo de propriedades RDF



Fonte: <<http://www.w3.org/TR/2002/WD-rdf-schema-20021112/>>

Figura 9 - Resumo das classes e relações no RDFS



Fonte: < <http://www.w3.org/TR/2000/CR-rdf-schema-20000327/>>

d) Topics Maps

Conhecidos como sendo o concorrente de RDFS, os “Topic Maps” (KUNZ; BOTSCH, 2002) são uma norma ISO, utilizados originalmente para anotar o conteúdo multimídia e baseados em uma DTD SGML e possui uma sintaxe XML (LACHER; DECKER, 2001).

Os fundamentos dos Topic Maps se baseiam sobre quatro princípios (PEPPER; MOORE, 2010):

- Os Tópicos: Podem ser qualquer entidade ou indivíduo, ou as associações para uniões. Estas últimas podem elas mesmas ser os Tópicos.
- Os nomes dos Tópicos: Um ponto forte dos Topic Maps consiste em que as entidades ou indivíduos representados pelos Tópicos são independentes dos nomes que lhes são atribuídos, isso é muito importante porque aumenta a expressividade da linguagem e permite as abordagens multilingues.
- As ocorrências: São os recursos que possuem URI e podem ser representadas pelos Tópicos ou pelas entidades literais.

- Os escopos: Outra inovação dos Topic Maps, permitem definir melhor o contexto sobre o qual as relações entre recursos são definidos.

Como ponto negativo, pode-se dizer que eles são demasiado vagos e não muito restritivos e apresentam uma semântica ambígua. Em resumo, eles pagam o preço de uma expressividade muito forte (CAUSSANEL et al., 2002) & (GARSHOL, 2002).

e) Discussão

O desenvolvimento de aplicações baseadas nos Metadados poderá conduzir à bons resultados nos domínios nos quais existe a necessidade de especificar os dados limitados destinados a pequenas comunidades (o compartilhamento do significado é simples) ou cujos metadados são de uma complexidade reduzida (tais como os serviços de indexação ou de pesquisa). Porém, sem normalização, e sem consenso sobre a semântica de diferentes entidades tratadas, e sem a formalização de sua representação e as operações permitidas sobre eles, alcançar a taxa de automatização prometido pela Web Semântica é ilusória. Isso somente poderia ser alcançado pela implementação de estruturas permitindo sua automatização, portanto, a introdução do conceito de ontologia.

2.3.1.2.2 Ontologias

a) Definição e propriedades

A palavra ontologia deriva do grego *onto* (ser) + *logia* (estudo). Ciência que estuda o ser, como tal. As primeiras definições de ontologia surgiram na Filosofia, com forma generalista no século XVII, e estavam ligadas à metafísica, designando ‘aquilo que existe’. Do ponto de vista filosófico, “Ontologia” com “O” maiúsculo é o ramo da Filosofia que lida com a natureza e a organização da realidade. Na organização e representação do conhecimento, conceito da Ciência da Informação, o sentido é diferente. A partir de uma determinada área de domínio, a ontologia (com “o” minúsculo) se propõe a classificar as coisas em categorias, na perspectiva do sujeito e da linguagem do domínio. Partindo da definição de Gruber et al. (1993) na qual “Uma ontologia é uma especificação explícita de uma conceitualização” (GRUBER, 1993 apud SCHREIBER, 2008, p.296, tradução nossa).

No presente trabalho ontologia é restrita à ótica da Ciência da Informação. Ontologias podem

ser muito úteis para um Sistema de Recuperação de Informação porque são estruturadas de tal modo (classes-instâncias-relações), que permitem ir consideravelmente além das possibilidades oferecidas por outros sistemas de classificação como Tesaurus, por exemplo, que é um vocabulário de um ramo do saber que descreve, sem ambiguidade, os conceitos a ele atinentes (DUQUE, 2005).

As ontologias são um meio comumente utilizado para padronização de conceitos dentro de um determinado domínio. Esta padronização é benéfica em relação à comunicação porque pode normalizar os conceitos utilizados na comunicação e assim diminuir as variações de interpretação ou representação (MORI, 2009).

As ontologias vêm sendo bastante pesquisadas e utilizadas como modelos conceituais em sistemas de informação (DIAS et al., 2007). Tal advento nos diz que esta tecnologia é bastante útil em vários aspectos, dentro os quais se podem destacar a padronização e a representação conceitual. O conceito de ontologia, ainda que discutível, possui certo direcionamento sobre representações conceituais consensuais e formais:

“Uma ontologia é uma conta explícita, parcial de uma conceituação.” (GUARINO, 1995, p.15).

“[...] Uma ontologia é um documento ou arquivo que define formalmente as relações entre termos.” (BERNERS-LEE et al., 2001, p.5).

“Uma ontologia é uma teoria de que entidades podem existir na mente de um agente inteligente.” (WIELINGA et al., 1994, p.6).

Pesquisa com origem na Engenharia do Conhecimento, as ontologias são um dos pilares da plataforma da Web Semântica, o que a torna uma das áreas mais ativas e bem sucedidas. Poderiam ser definidas como sendo os modelos que descrevem a maneira de ser das diferentes entidades presentes em um dado domínio. Em outras palavras, permitem a representação de todo o conhecimento útil de um domínio, com o objetivo de normalizar seus significados (BOUAUD et al., 1995) & (CHARLET; BACHIMONT; TRONCY, 2003), permitindo assim:

- O compartilhamento do mesmo vocabulário conceitual através da mesma atribuição da mesma semântica pelos diferentes atores do domínio modelado.
- Definir uma semântica formal para as informações modeladas para a ontologia, permitindo à aplicação utilizá-los para manipular de maneira automática.

Isto se faz explicando de maneira formal as propriedades destes conhecimentos e as relações que existem entre eles, todos eles estando baseados em um consenso entre as comunidades que trabalham neste domínio.

A contribuição das ontologias para a Web é inegável, pode ser resumido em termos do desenvolvimento de aplicações de acordo com Patel-Schneider:

Os usuários de um servidor Web de ontologias terão acesso à numerosos serviços poderosos para armazenar e analisar o conhecimento disponível sobre a Web e fazer as inferências à partir deste conhecimento, fornecendo aos programadores muito mais do que os dados contidos explicitamente nos documentos XML. (SCHNEIDER, 2004, p.23).

O que conduz a um aumento da interoperabilidade das aplicações, devido a normalização das entidades que eles manipulam e a descrição das operações permitidas sobre estas, assim como uma certa facilidade de implementação devido a reutilização fornecida pelas ontologias. Mas para isso, as ontologias devem estar sujeitas aos critérios de qualidade, segundo Gruber et al., 1993:

- Clareza: Axiomatizar tudo o que for possível e fornecer as descrições as mais claras possíveis em linguagem natural (se possível);
- Coerência: Consistência dos axiomas e coerência das descrições em linguagem natural (trabalho de documentalistas);
- Extensibilidade: Poder adicionar sem comprometer o que já foi feito;
- Compartilhamento: Conceitualizar a ontologia independentemente das linguagens de implementação, permitindo assim sua reescritura por diversas aplicações sob diferentes formas;
- Compromisso ontológico mínimo: A ontologia deve modelar uma visão do domínio para uma dada aplicação e não fazer suposições sobre outras visões do mesmo domínio, permitindo assim uma melhor especialização e uma combinação com o mínimo de contradições entre as representações.

Da mesma forma, para realizar tais funcionalidades e permitir uma melhor concepção de ontologias, as linguagens utilizadas para sua modelagem devem possuir formalismos permitindo representar os três tipos de conhecimento seguintes (MOURA, 2009):

i. Conceitos

Os conceitos são os objetos e as noções que representam um papel no domínio modelado,

são caracterizados por:

- Seus termos (denominação). Ex.: trigo, pão.
- Suas instâncias (chamadas extensões). Ex.: conceito pão, instância ‘croissant’.
- Intenção (o conjunto de propriedades, as definições). Ex.: cor, forma etc.

ii. Relações entre conceitos

Uma relação é uma ligação existente entre diferentes conceitos, é caracterizada por:

- Seu termo ou denominação (o pão é feito de farinha).
- Sua extensão: o conjunto de relações nas quais ela aparece (o pão é feito de farinha, o bolo é feito de farinha - isto faz duas extensões da relação X é feita de Y).
- Sua intenção: o que define os conceitos podem ser ligados pela relação e a significação desta relação e o que resulta (é feito de : quer dizer que faz parte de sua composição).

iii. Axiomas

Os conceitos e as relações são as primitivas terminológicas do domínio modelado, os axiomas servem para definir as intenções que lhes são atribuídas, eles permitem também representar os conhecimentos que não podem ser descritos pelas duas estruturas antecedentes tais como os processos (“como fazer o pão”), os raciocínios possíveis relacionados ao domínio ou ainda definir sobre os conceitos e as relações as restrições semânticas (“as galinhas são aves mas elas não voam”), definindo as condições de coerência do domínio modelado. São os axiomas particulares chamados axiomas schemas, compartilhados por numerosas terminologias primitivas:

- As propriedades algébricas de uma relação (transitividade, simetria, reflexividade etc.);
- A subclassificação entre conceitos ou relações (por exemplo, a relação tipo);
- A abstração de um conceito;
- A cardinalidade das relações;
- A incompatibilidade e a exclusividade entre primitivas.

Outros axiomas específicos aos domínios tratados devem poder ser definidos, tais como a cardinalidade de um conceito (“deve haver pelo menos um médico no barco”). A falta de expressividade de RDF ou RDFS e sua inaptidão à representação de conhecimentos assim complexos como as ontologias ficam claras, devido a necessidade de introduzir novas

linguagens que ofereçam maior poder de expressão, tendo em conta as particularidades do contexto de implementação (um ambiente distribuído e inconsistente), sendo um dos mais promissores: OWL.

b) A Ontologia da Linguagem Web (OWL)

- Definição e histórico

OWL é uma linguagem que nasceu da necessidade de padronizar os meios de representação de ontologias sobre a Web e de compensar a falta de expressividade das linguagens dos metadados como RDF que permite somente a expressão de declarações sobre as relações entre objetos, e cuja extensão, RDFS, fornece somente mecanismos primitivos para especificar as classes, inaptas para representar as classes e as propriedades complexas necessárias à implementar os conhecimentos de tipo ontológico (SILVA et al., 2012).

Um grande número de esforços de pesquisa durante o final da década de 1990 exploraram como a idéia de representação de conhecimento da inteligência artificial poderia ser utilizada na Web. Estes esforços incluíram linguagens baseadas em HTML (chamada SHOE), XML (chamada XOL e, mais tarde, OIL), e várias linguagens baseadas em frames e abordagens de aquisição de conhecimento.

A W3C criou o "Web Ontology Working Group", que começou a trabalhar em 1 de novembro de 2001, presidido por James Hendler e Guus Shreiber. O primeiro rascunho da sintaxe abstrata, da referência e da sinopse foram publicado em julho de 2002. Os documentos da OWL se tornaram uma recomendação formal da W3C em 10 de fevereiro de 2004, e o grupo de trabalho foi dispensado em 31 de maio de 2004 (MCBRIDE, 2002).

Dentro do grupo de trabalho, o esforço para identificar os objetivos de projeto e os requisitos foi liderado por Jeff Heflin. Alguns requisitos foram contribuição de Deborah McGuinness, baseada em mais de uma década de trabalho na construção de sistemas baseados em ontologias. Outros requisitos foram identificados como parte do trabalho do Ph.D. de Heflin na construção de um protótipo de um sistema da Web semântica. Os outros membros do grupo de trabalho contribuíram com mais de 25 casos de uso, que foram depois resumidos na definição de um conjunto de casos de uso (HEFLIN, 2004).

A OWL é uma linguagem para definição e instanciação de ontologias Web. Uma ontologia

OWL pode formalizar um domínio, definindo classes e propriedades destas classes, definir indivíduos e afirmações sobre eles e, usando-se a semântica formal OWL, especificar como derivar consequências lógicas, isto é, fatos que não estão presentes na ontologia, mas são vinculados pela semântica (SMITH; WELTY; MCGUINNESS, 2009).

- Descrições de OWL e de seus construtores

Sucessor de DAML+OIL, a linguagem OWL surgiu para compensar seus defeitos de complexidade conservando suas qualidades inegáveis em matéria de representação. Dotada de uma sintaxe XML, oferece numerosos mecanismos e construtores, permitindo Allemang e Hendler (2011):

- A declaração de classe de diferentes maneiras

- Por referência: É feito dando o URL da definição de classe.
- Por enumeração de suas instâncias: (owl:oneOf)

Exemplo: a classe 'estação': verão, inverno, primavera, outono

```
<owl:Class rdf:ID="estação">
  <owl:oneOf rdf:parseType="Collection">
    <owl:Thing rdf:about="#verão"/>
    <owl:Thing rdf:about="#inverno"/>
    <owl:Thing rdf:about="#primavera"/>
    <owl:Thing rdf:about="#outono"/>
  </owl:oneOf>
</owl:Class>
```

- Por suas propriedades:

Um estudante é um matrícula, sobrenome, nome

```
<owl:Class rdf:ID="estudante">
  <owl:Property rdf:resource="matricula"/>
  <owl:Property rdf:resource="sobrenome"/>
  <owl:Property rdf:resource="nome"/>
</owl:Class>
```

- Por restrição de suas propriedades:

```
owl:allValuesFrom,owl:someValuesFrom,owl:hasValue
owl:maxCardinality,owl:minCardinality, owl:Cardinality
```

Exemplo: “um solteiro é um homem onde a propriedade número de esposas=0”.

```
<owl:Class rdf:ID="solteiro">
  <rdfs:subClassOf rdf:resource="HOMEM"/>
  <owl:Restriction>
    <owl:onProperty rdf:resource="número de esposas" />
    <owl:Cardinality rdf:resource="0" />
  </owl:Restriction>
</owl:Class>
```

- Por composição de outras classes: (união, intersecção, complemento).

owl:intersectionOf, owl:unionOf, owl:complementOf

Uma aranha intersecção homem = homem aranha

```
<owl:Class rdf:ID="HOMEM ARANHA">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="HOMEM">
    <owl:Class rdf:about="ARANHA">
  </owl:intersectionOf>
</owl:Class>
```

- A definição de relação entre classes

- As sub classes: rdfs:subClassOf

A classe galinha é uma subclasse de pássaro.

```
<owl:Class rdf:ID="GALINHA">
  <rdfs:subClassOf rdf:resource="PASSARO"/>
</owl:Class>
```

- A equivalência de classes owl:equivalentClass

```
<owl:Class rdf:ID="estudante">
  <owl:equivalentClass rdf:resource="student"/>
</owl:Class>
```

- Disjunção de classes: owl:disjointWith

```
<owl:Class rdf:ID="AGUA">
  <owl:disjointWith rdf:resource="FOGO"/>
</owl:Class>
```

- A definição das propriedades e das relações entre propriedades

rdfs:subPropertyOf, rdfs:domain, rdfs:range owl:equivalentProperty, owl:inverseOf
owl:SymmetricProperty ,owl:TransitiveProperty

Descendente é a propriedade inversa de ancestral.

```
<owl:ObjectProperty rdf:ID="descendente">
  <owl:inverseOf rdf:resource="ancestral"/>
</owl:ObjectProperty>
```

- A definição de instâncias de classes ou de propriedades e das relações entre elas:

```
“Rafael é um dentista” : <dentista rdf:ID="Rafael" />
```

- A definição de informações de suporte

rdfs:label, rdfs:comment, rdfs:seeAlso,rdfs:isDefinedBy owl:versionInfo, owl:priorVersion,
owl:backwardCompatibleWith, owl:incompatibleWith, owl:DeprecatedClass,
owl:DeprecatedProperty

```
<rdfs:label> Isto é um comentário. </rdfs:label>
```

- Problema da computabilidade

OWL fornece um grande número de construtores permitindo expressar de forma bastante precisa as propriedades das classes definidas. Porém, o reverso da medalha é a complexidade dos mecanismos de inferência e a indecibilidade da linguagem obtida considerando o conjunto destes construtores. Por isso a necessidade de dividir em três linguagens distintas (MOTIK et al., 2009):

- OWL LITE - Inclui todos os construtores de RDF e RDFS. Contém somente um subconjunto reduzido de construtores de OWL. Coloca restrições de utilização sobre alguns fabricantes (como as restrições de cardinalidade que se limitam a 1 ou 0) o que dá à esta “personalização” uma flexibilidade na comparação de tipos (que se torna um problema NP).
- OWL DL - Contém todos os construtores de OWL, mas com restrições sobre sua utilização (Exemplo: ele não permite definir uma classe de classes), concedendo-lhe

assim a propriedade de decidibilidade e de computabilidade, mas fica complexo de utilizar devido a introdução das abordagens de manipulações heurísticas.

- OWL FULL - Contém todas as construções de OWL, e não coloca nenhuma restrição sobre sua utilização, devido a impossibilidade dos tipos. Tomando como exemplo a utilização de uma classe que pode ser considerada como classe ou como indivíduo o que criará numerosas situações de indecidibilidade.

- OWL 2

A OWL 2 *Language Ontology Web*, informalmente OWL 2, é uma linguagem ontológica para a Web Semântica com significado formalmente definido. Ontologias OWL 2 fornecem classes, propriedades, indivíduos, e valores de dados e são armazenadas em documentos da Web Semântica. Ontologias OWL 2 podem ser usadas junto com informações escritas em RDF, e as próprias ontologias OWL 2 são trocadas essencialmente por documentos RDF.

A estrutura conceitual de ontologias OWL 2 está definida no documento OWL 2 *Structural Specification*. Este documento utiliza UML para definir os elementos estruturais disponíveis em OWL 2, explicando suas funções e funcionalidades em termos abstratos e sem referência a qualquer sintaxe especial. Ele também define a sintaxe de estilo funcional, que segue a especificação estrutural e permite a escrita de ontologias OWL 2 de forma compacta.

Uma ontologia OWL 2 pode ser vista como um grafo RDF. A relação entre estes dois pontos de vista é especificada através do *Mapping* para documentos RDF *Graphs*, que define um mapeamento da forma estrutural para a forma de grafo RDF, e vice-versa. O OWL 2 Quick Reference Guide fornece uma visão simples dos dois pontos de vista. (Em: <http://www.w3.org/TR/owl2-overview/>. Acessado em 27/09/2013).

c) Discussão

A contribuição de representação de ontologias é importante, mas não parece muito factível acreditar que a Web Semântica possa ser uma utilidade qualquer somente pela descrição de recursos, sem o estabelecimento de mecanismos de exploração que permitem a utilização de conhecimentos - implantados por exemplo aos agentes de software - e de promover os

serviços cada vez mais complexos e variados.

2.3.1.3 Inferências e Lógica

O objetivo estabelecido para a Web Semântica é uma melhor utilização dos recursos e do conhecimento da Web e isto por intervenção mais intensa das máquinas e a minimização da implicação humana. Este objetivo poderá ser realmente alcançado quando as aplicações que utilizam as particularidades da infraestrutura da Web Semântica - formalização e identificação das descrições dos recursos - possam ser desenvolvidas (CHARTON; GAGNON; OZELL, 2010).

Mas para elaborar tais aplicações, é necessário primeiramente adaptar ou inovar as técnicas de exploração levando em conta as estruturas de dados impostas por esta nova visão da Web, tais como a descrição de recursos através de metadados formalizados e/ou de conhecimentos implementados por ontologias, e que, nos diferentes processos de extração, de combinação e de transformações implica: *“Uma melhor especificação para uma melhor exploração”*.

Os métodos de exploração variam de acordo com sua origem, sua complexidade e sua capacidade de combinar os recursos. Os mais simples são aqueles de serviços que utilizam os sistemas de gestão de informação (tipo motor de pesquisa para palavra chave ou SGDB) que, através da normalização semântica e estrutural e devido a utilização de metadados e tesouros, poderão refinar a pesquisa de dados por atributo (autor, ano, domínio de pesquisa etc.) e acessar um número crescente de dados distribuídos - devido a normalização da representação e aos sistemas mediadores - e isto através de algoritmos de pesquisa clássicos. Mas esses sistemas são um pouco rígidos e forçam o usuário a especificar suas consultas sob um formato muito restrito e em uma terminologia demasiado precisa sob pressão de resultados medíocres. Esta abordagem poderá ser então denominada como sendo “uma melhoria de exploração da Web atual”, não utilizando todo o poder dos formalismos que envolve a infraestrutura da Web Semântica permitindo uma exploração mais “inteligente” dos conhecimentos (VIEIRA et al., 2005).

2.3.1.3.1 A Web inferencial

A Web inferencial é o que será obtido através da recuperação da camada de descrições de recursos (metadados e ontologias) por uma outra definição de inferências e de raciocínios sobre este último, permitindo assim uma exploração e uma combinação mais pujantes das diferentes representações (MCGUINNESS; SILVA, 2004).

2.3.1.3.2 O que é inferência?

A inferência indica o conjunto de mecanismos pelos quais é possível deduzir os novos conhecimentos (resultados da inferência) a partir de outros conhecimentos (as premissas). Quando uma inferência possa ser justificada (em relação a certas regras pré-estabelecidas), então se trata de raciocínio (ZOU; FININ; CHEN, 2005).

2.3.1.3.3 Motores de inferência

Os motores de inferência são aplicações baseadas sobre os sistemas a base de conhecimentos (e de seus antecedentes, os sistemas especialistas) podendo fazer inferências sobre a base de premissas implantadas devido as ontologias ou às partes de ontologias, e isto possibilita realizar (ZOU; FININ; CHEN, 2005):

- A formalização da interpretação de consultas em linguagem natural (os motores de questão/resposta).
- A pesquisa de informação inteligente para a incorporação da semântica do conceito em sua pesquisa.
- A combinação e a incorporação de tipos de dados cada vez mais diversos para o aumento da capacidade de reconhecimento de formatos devido a utilização de ontologias.
- A verificação da consistência e a pesquisa de incoerências nas bases de conhecimento implementadas.
- Elaborar reações mais rápidas e mais refinadas sobre os processos em tempo real através da assistência da intervenção do fator humano (tais como as aplicações de controle de segurança ou bolsa de valores).

2.3.1.3.4 Ontologias e motores de inferência

As ontologias são as especificações conceituais dos conhecimentos e da semântica de um dado domínio que são - por uma questão de utilização - independentes da representação operacional do mesmo. Para poderem ser utilizadas por um motor de inferências, os conhecimentos devem ser transcritos em uma visão operacional para adaptar a um dado objetivo. Este processo é chamado de compilação do conhecimento, e consiste em transcrever um conjunto de enunciados lógicos em uma lógica formal permitindo o cálculo de consultas, e assim (FÜRST; LECLÈRE; TRICHET, 2003):

- Determinar os axiomas da ontologia e sua finalidade:

- Objetivo descritivo (para os axiomas de descrições de conceitos).
- Objetivo dedutivo (os axiomas que formulam as regras de dedução do domínio).
- Objetivo reativo (os axiomas que definem uma reação no caso em que estejam verificadas as condições).
- Conservação da integridade (os axiomas que precisam ser verificados para conservar a coerência e a representação).

- Definir a estratégia de aplicação para cada axioma:

- Aplicar automaticamente quando as pré-condições forem respeitadas (utilização implícita).
- Perguntar ao usuário quando aplicar uma dada regra (utilização explícita).

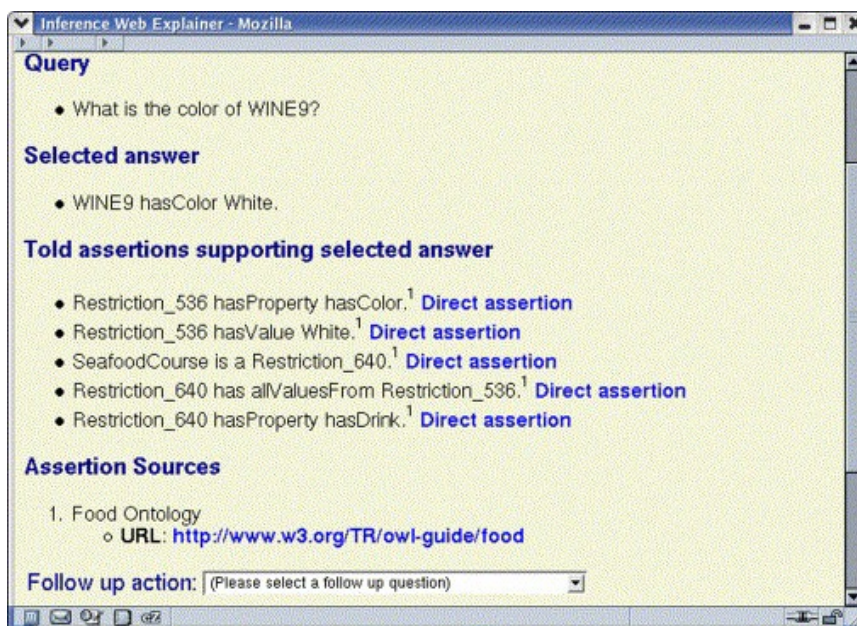
Depois disso, é determinado o schema de aplicação dos axiomas, intervindo ou não o usuário. A escolha da linguagem de operacionalização depende do objetivo do motor de inferência porque cada um tem suas especificidades, o tornando mais apto à uma categoria de problemas e um tipo de inferências bem precisas (por exemplo o cálculo de subclassificação para a linguagem OWL baseada sob os DLs). Mas a linguagem deve compartilhar o máximo de paradigmas com a linguagem de especificação da ontologia.

2.3.1.3.5 Alguns “reasoners” (WU et al., 2008)

- OTTER e New Automated Reasoning Kit (SNARK): motores de inferência baseados sobre os dados RDF e o formalismo da lógica de primeira ordem.
- Java Theorem Prover (JTP): motor de inferência dos laboratórios de Stanford, Web orientados, desenvolvidos pelo “KSL Inference Web Project”.
- RACER e FaCT.: (baseados sobre a lógica de descrição) os motores de referência principais do W3C para as inferências sobre ontologias, já implantados em várias aplicações tal como PROTEGE (para a verificação de inconsistência).

A Figura 10 mostra um exemplo de interface gráfica de um “reasoner” de inferência Web (IW), desenvolvido na Universidade de Stanford com base do “reasoner” JTP.

Figura 10 - Exemplo de “reasoner” de IW desenvolvido em Standord



Fonte: McGuinness e Silva (2004, p.21)

2.3.1.3.6 Discussão

Importante enfatizar que os tipos de inferências atualmente estudadas no escopo da Web Semântica se limitam àqueles relacionados à lógica de predicados e definidos pela lógica de descrição - tal como o cálculo de subclassificação e os testes de satisfabilidade.

Portanto, estudar a possível contribuição pela implementação e operacionalização de conhecimentos por meio de outros formalismos de representação, e seus métodos de inferência, poderia ser uma linha de pesquisa promissora para a Web Semântica. Isto permitiria,

entre outras coisas, a ampliação do número dos domínios de aplicação possíveis e que, pelo enriquecimento dos métodos de exploração aplicáveis aos dados, resume os resultados que se espera conseguir com o presente trabalho.

2.3.1.4 Prova e nível de confiança

O objetivo esperado pela estrutura da Web Semântica é alcançar um espaço no qual a intervenção do usuário é reduzida ao mínimo pelo aumento das capacidades de extração de dados e do raciocínio sobre estes últimos pelas aplicações permitindo à Web clássica se tornar uma Web inferencial. Da mesma forma, oferecer mecanismos de comunicação e de troca de conhecimentos diversos. Mas, para que os usuários possam aceitar, qualquer agente que ofereça conhecimento adquirido ou construído por meio dos métodos de combinações e de raciocínios devem possuir a justificativa, oferecendo para isto as informações sobre (ANTONIOU et al., 2008):

- Os dados e metadados utilizados (autenticação das premissas):

- Sua origem.
- A versão da informação e a data de sua atualização.
- As agências avalistas e certificadoras da veracidade da informação assim como seu nível de confiança.
- Informar a existência de outros conhecimentos contraditórios com aqueles utilizados e os parâmetros que lhes dizem respeito.

- Provar seu raciocínio (eventualmente explicar suas escolhas), especificando:

- Os motores de inferência utilizados e suas versões.
- As regras de dedução aplicadas.
- A forma que essas regras foram aplicadas.

É necessário também implementar os mecanismos de apresentação destes dados ao usuário que devem ser definidos devido as linguagens formais permitindo sua manipulação pelos agentes que poderão se basear sobre estes resultados para escolher os dados mais seguros para reutilizar.

Para isto, várias tecnologias foram ou estão sendo desenvolvidas, em que seguem alguns resultados (ZAIHRAYEU; SILVA; MCGUINNESS, 2005):

- PML (Proof Markup Language): baseada na OWL, esta linguagem permite a especificação e a troca de provas.
- IWE (Inference Web Explainer): motor de elaboração de explicações interativas dos raciocínios baseados sobre documentos PML, permitindo uma melhor apresentação dos mesmos.
- IWBase: bases de metadados distribuídos contendo os conhecimentos relativos aos motores de inferência, linguagens, fontes de dados e regras, necessários ao estabelecimento de provas.
- IWTrust Net (rede de confiança): redes que descrevem quantitativamente a confiança de diferentes partes entre elas (usuário e aplicação) permitindo o cálculo da confiança atribuída aos recursos através das informações conhecidas por todos os elementos da rede.
- OWL-QL-based tools: ambiente que permite ao usuário explorar os conceitos e as relações entre conceitos (descritos em OWL) da mesma forma criar novos axiomas, colocar questões e receber respostas, assim como explicações sobre as respostas.

2.3.2 Domínios de aplicação e implementação existentes na Web Semântica

O verdadeiro poder da plataforma Web Semântica poderá somente ser alcançado quando as aplicações em diversos domínios levarem em conta a gama de tecnologias que ela oferece mostrando, desta forma, o verdadeiro potencial de uma tal infraestrutura e demonstrando, ao mesmo tempo, às partes financeiras, seu verdadeiro poder comercial.

2.3.2.1 Domínios de aplicação

Várias áreas de aplicação poderiam usar a Web Semântica, assim como outras novas poderiam emergir devido as novas possibilidades que ela oferece. Apresentamos as mais

promissoras dentre elas:

2.3.2.1.1 Personalização e adaptação da Web

Os usuários da Web se deparam com dificuldades de utilização devido a não-adequação dos recursos às suas exigências ou à suas faculdades de utilização limitadas, provocando inconvenientes, tais como:

- A necessidade de consulta de fontes de informação muito grandes e dos recursos cada vez mais complexos (mesmo estando interessado apenas em uma parte deles) para poder acessar a informação desejada.
- A dificuldade de especificar essas demandas utilizando a pesquisa por meio de palavras chaves (mais frequentemente) e a obrigação de fazer uma triagem manual das informações não pertinentes.
- A desorientação durante a navegação em um exemplo de mídias devido a incoerência dos schemas de representação de dados e a carga cognitiva conduzida pelo número de informação para gerenciar.

Estas dificuldades poderiam ser bastante reduzidas pelo aumento da capacidade de processamento automático dos dados da Web Semântica, e isso permitindo às máquinas:

- Ajudar o usuário a formular suas consultas (propondo, por exemplo, termos adequados escolhidos em virtude do domínio de pesquisa combinado as informações de seu perfil) a fim de refinar os resultados obtidos.
- Filtrar os resultados obtidos e isto seguindo os critérios personalizados para o usuário ou outros deduzidos de seus “hábitos” de utilização.
- Permitir uma representação personalizada dos recursos da Web seguindo os schemas de apresentação indo desde a triagem alfabética aos modelos mais complexos tais como os modelos de tarefas ou estruturas narrativas etc.

Para isto, as pesquisas existentes são baseadas em três eixos principais:

- A modelagem do usuário: Será necessário poder possuir conhecimentos (conjunto de dados relevantes e persistentes) descrevendo um usuário ou um grupo de usuários a fim de poder personalizar um recurso em função de suas preferências, capacidades e necessidades.
- As hiper-mídias adaptativas: São as hiper-mídias que têm diferentes maneiras de

serem apresentadas, permitindo personalizar as informações fornecidas e o método de apresentá-las, tornando assim possível a limitação da carga cognitiva do usuário.

- Os documentos virtuais personalizáveis: A evolução das hiper-mídias adaptativas deu nascimento aos documentos visuais construídos por demanda, permitindo personalizar seguido o perfil dos usuários e suas necessidades atuais (especificados por demanda) dos dados para representar e sua organização, permitindo assim uma reutilização de fontes de informações.

2.3.2.1.2 Serviços da Web Semântica

Os serviços Web são as aplicações publicadas pelos fornecedores, utilizando a plataforma da Web para prover os serviços, indo do boletim meteorológico até a compra de uma pizza. A maior ocupação da comunidade de pesquisa sobre os serviços Web é a interoperabilidade entre aplicações através da Web para torna-la mais dinâmica. Isto somente pode ser feito através da automatização de tarefas asseguradas pelos serviços Web, tais como (STUDER; GRIMM; ABECKER, 2007):

- A descrição e a publicação.
- A descoberta e a seleção.
- A composição.

Permitindo assim, o estabelecimento de serviços nos quais as propriedades, as capacidades, as interfaces e os efeitos são descritos de maneira não ambígua e legíveis pelas máquinas. Para esta finalidade, as tecnologias da Web Semântica poderiam fornecer a normalização e o formalismo que faltam ao processo de elaboração de serviços, removendo, ao mesmo tempo, outras limitações da intervenção humana (como os limites frente à quantidade de informações a ser processadas ou a tomada de decisões nos ambientes de reatividade elevada) ou financeira (para a diminuição dos custos de desenvolvimento e da manutenção devido a reutilização e a combinação de serviços e o compartilhamento do vocabulário de desenvolvimento através da abordagem ontológica).

Para alcançar esses objetivos, é necessário primeiramente normalizar os conceitos e as abordagens de elaboração dos serviços Web propostos por diversas comunidades de pesquisa tais como a engenharia de software, os workflows, as bases de dados, a modelagem de empresas, a representação dos conhecimentos ou os multiagentes dando nascimento às visões conceituais algumas vezes inconsistentes ou incompatíveis.

É possível, por exemplo, descrever as interfaces de um serviço Web (o formato das mensagens, as assinaturas das operações, o protocolo de transporte e a localização do serviço) permitindo assim chama-lo sem conhecimento de sua linguagem de desenvolvimento (Java, C++ etc.) ou de sua plataforma de implementação (*.net, Java Platform, Enterprise Edition (J2EE), etc.*). Os três atores principais sobre os quais se baseiam a utilização de um serviço Web:

- O fornecedor do serviço (podendo ser materializado pela plataforma de implementação do serviço).
- O cliente: o usuário do serviço podendo ser humano ou uma aplicação (ou mesmo um outro serviço Web).
- A correspondência: o diretório no qual o provedor publicou o seu serviço onde o cliente procurou depois de chamado.

2.3.2.1.3 Linguagem de descrição e de composição dos serviços Web

Várias linguagens de descrição de serviços foram propostas em pesquisas acadêmicas e industriais, porém a maioria foi desenvolvida para tarefas precisas. Foram listadas algumas linguagens principais de descrição, descoberta, composição e invocação de serviços (HEPP et al., 2007):

a) *Universal Description Discovery and Integration (UDDI)*

Esta plataforma é destinada à descrição e à descoberta de serviços Web e isto devido a um sistema de pesquisa baseado sobre as palavras-chave fornecidas pelos organismos que oferecem seus serviços (como as páginas amarelas).

Ela permite também reagrupar nas estruturas chamadas páginas brancas as informações permitindo contactar as empresas inscritas (endereço, número de fax, de telefone, identificação etc.).

Estas descrições são submetidas à uma certa restrição taxonômica a fim de permitir uma certa categorização dos serviços Web de empresas onde a descrição é armazenada e isto para melhorar o serviço de pesquisa.

Permite também associar à todos os serviços, desde “páginas verdes”, que são um conjunto

de informações técnicas sobre o serviço e isto apontando uma descrição do serviço técnico descrito em outras linguagens especializadas (como ROSETANET ou os serviços de interface do WSDL (Web Services Description Language)). As informações UDDI são armazenadas em um servidor centralizado onde o conteúdo é sincronizado com outros.

Devido sua simplicidade, UDDI permite uma categorização de todos os serviços Web, mas algumas questões são colocadas sobre a possibilidade ou não de uma solução centralizada sem descrição, limitando-se às palavras-chaves (desprovidas de semântica e não permitindo nenhuma aproximação de pesquisas) podendo gerenciar o “cluster” provenientes de serviços futuros e descrevelos de maneira a permitir uma interoperabilidade entre máquinas.

b) E-SPEAK

Desenvolvido nos laboratórios da HP desde 1995, E_SPEAK pretende ser uma abordagem mais global que a UDDI na descrição e execução de serviços (porque ele permite, por exemplo, gerenciar praticamente todas as tarefas implantadas nas redes, até mesmo, serviços de impressão).

Permite descrever os serviços sobre a base de atributos pré-fixados devido a possibilidade de incorporar uma certa semântica (custo a pagar: sua falta de flexibilidade).

O vocabulário de descrição e entre outros, os atributos, de um grupo de serviços são fixados para o grupo de trabalho que lhe é associado, onde existe uma certa falta de homogeneidade entre as descrições dos diferentes grupos de serviços (falta de uma semântica formal global). Exemplo: o tamanho pode ser definido por um grupo como sendo um número de página, quando outro grupo pode lhe definir como byte.

Uma conciliação entre E_SPEAK e UDDI foi anunciada pela HP devido a possibilidade de encontrar as especificidades do primeiro incluídas nas próximas versões do segundo.

c) WSDL

Linguagem baseada sobre XML, permite a descrição de interfaces de serviços Web e isto de maneira abstrata.

Sua tarefa é bem precisa, descrever os fluxos de informação que entram e saem de um serviço e deixam a tarefa de apresentação de invocação ou de operacionalização das tarefas

para outras linguagens especializadas (tais como UDDI, DARPA agent markup language for services (DAML_S) ou SOAP).

Pode também ser utilizada como porta de entrada entre linguagens de níveis diferentes, devido ao fato que suas descrições abstratas e sua sintaxe XML permitem uma tradução automática. WSDL permite descrever as operações feitas pelos serviços devido a quatro operações básicas: solicitação e notificação (sentido único); resposta à consulta e resposta à solicitação (duplo sentido).

Estas operações são associadas às portas de comunicação chamadas “*End Points*” para as quais são definidos os protocolos e os formatos de dados permitindo assim uma descrição universal, independente das implementações.

A abordagem modular das linguagens de descrição é bem representada, porque mesmo se o fluxo de dados são representados, a descrição de seu conteúdo é deixada para outras linguagens permitindo assim um desenvolvimento mais fácil e uma melhor especificação dos serviços.

O principal inconveniente de WSDL é a falta de restrições lógicas sobre as tarefas, permitindo, portanto, somente a representação de cenários simples de comunicação.

d) DAML_S

Baseado em XML e utilizando um modelo de lógica de descrição, DAML_S é uma linguagem de alto nível para a descrição e a invocação de serviços incluindo sua semântica. Para isto utiliza três tipos de descrição:

- Serviço de perfil - Permite a descrição e a promoção do serviço, incluindo as informações sobre os serviços fornecidos, assim como as pré-condições para a obtenção desses últimos. Ele fornece a possibilidade de pesquisa sobre qualquer critério dessa seção.
- Serviço modelo - É apresentado aqui o funcionamento do serviço nos detalhes (oposto ao WSDL) mas de uma maneira abstrata.

Os tipos de interações ou os procedimentos de comunicação descritos podem ser utilizados como critério de pesquisa como o serviço de perfil ou permitir o controle de tarefas em

execução, pois a descrição abstrata permite sua composição (para a automatização, porém, são necessárias descrições mais formais).

- Service grounding - Permite anexar a um comportamento abstrato de um serviço Web o procedimento real de sua execução e isto indicando de maneira não ambígua:
 1. Um formato de dados especifica cada tipo abstrato descrito pelo serviço Web, 2. Um protocolo de comunicação garante os procedimentos de comunicação e a troca de dados e 3. Os métodos de autenticação ou de segurança das transferências.

DAML_S poderia estar dedicado para descrição semântica dos serviços, podendo assim, por exemplo, incluir as restrições lógicas do tipo (SE..ENTÃO..SENÃO) e utilizar as interfaces WSDL (proporcionando uma estruturação das trocas) permitindo a utilização SOAP para a especificação de trocas de mensagens em XML (abordagem de descrição modular).

Embora ela resuma as diferentes tarefas à executar, as linguagens necessárias para isto não refletem o problema da coordenação do cenário de utilização onde figura ou a gestão de combinações de serviços (onde o cliente é ele próprio um serviço). Para isto, os métodos de preservação da coerência do diálogo entre os diferentes atores foram propostos pela comunidade de pesquisa com base nos fluxogramas (gestão de processos). Eles são baseados sobre a modelização dos seguintes conhecimentos: 1.“Quais são as atividades a serem realizadas?” 2.“Que competências são necessárias para realizar essas atividades?” 3.“Quando deverão ser realizadas esas atividades?” e 4.“Quais são as ferramentas e as informações necessárias para a realização dessas atividades?”

Para esta finalidade diferentes linguagens foram propostas para a descrição de processos públicos (a maneira que ele opera) tal como WSCL ou a especificação dos procedimentos de composição de serviços tais como o BPML ou BPEL4WS. Mas a modelagem de serviços não poderia assegurar a automatização, para isto foram desenvolvidos os algoritmos e mecanismos de raciocínio explorando a semântica dos serviços para automatizar as diferentes funcionalidades (descoberta de serviço, composição, etc.)

A avaliação de desempenho é também um outro fator importante que demanda a elaboração de critérios (quantitativos ou qualitativos) descrevendo “a qualidade de um serviço” em relação à satisfação do cliente e a carga do fornecedor (custo em recursos do sistema, por exemplo).

2.3.2.1.4 Portal e memória corporativa

Todo organismo (comercial ou não), tendo uma certa magnitude, pode perceber a experiência e os conhecimentos que foram adquiridos em seu campo de trabalho (processos de fabricação, técnicas de desenvolvimento, mecanismos de gestão e logística etc.) como o capital em si mesmo, podendo ser armazenado, compartilhado e reutilizado. Daí o surgimento de uma nova disciplina que se interessa à representação dos conhecimentos: A “Gestão do Conhecimento” (BENATALLAH et al., 2005).

Assim, a memória da empresa pode ser definida como sendo uma representação explícita, persistente, e desvinculada, dos conhecimentos e das informações em uma organização. Conhecimentos tais como:

- Os conhecimentos sobre os produtos (composição, processos de produção, preço, compatibilidade, nota de uso, etc.);
- As informações sobre os clientes (identidade, endereço, contrato, etc.);
- As estratégias de venda (metodologias de aplicação, parâmetros, etc.);
- Os resultados financeiros, as estatísticas de produtividade, o balanço, etc.).

A contribuição da Web Semântica (representação dos conhecimentos e métodos de raciocínio sobre os dados), proporcionando um aspecto semântico à representação, poderia proporcionar um novo impulso às memórias das empresas limitadas atualmente pela representação documentalista textual armazenada nas sedes das empresas.

Isso poderia permitir um melhor compartilhamento para a elaboração de portais do conhecimento corporativo automatizados (compartilhamento essencialmente manual nos dias de hoje) o que permite uma melhor exploração e fornece soluções evolutivas e dinâmicas permitindo assim:

- O acesso aos conhecimentos da empresa pelo pessoal de campo (mobilidade);
- Permite que diferentes empregadores incorporem novos conhecimentos e isto de maneira simples e natural (devido as ferramentas de edição de ontologias);
- Permite a implementação de memórias de empresas compartilhadas e descentralizadas;
- Oferecer os serviços de e-learning aos empregados no portal corporativo permitindo-lhes acessar percursos de formações personalizados.

2.3.2.2 Implementações existentes

Seguem dois exemplos de aplicação utilizando a plataforma da Web Semântica (LI et al., 2004):

2.3.2.2.1 Ontoseek (1996 - 1998) (CHARLET; LAUBLET; REYNAUD, 2003)

É uma plataforma de pesquisa multilíngue baseado sobre:

- A utilização de uma ontologia linguística generalista (SENSUS);
- Mecanismos de raciocínio baseados sobre os grafos conceituais utilizando a intersecção semântica entre pesquisa do usuário e descrição dos produtos (cálculo de subclassificação);
- Assistência para escrever consultas devido às ferramentas de generalização e de especialização.

Tal abordagem semântica permite ao usuário especificar em linguagem natural as descrições do produto que ele pesquisa sem saber o vocabulário de codificação dos produtos.

2.3.2.2.2 Multilingual Knowledge Based European Electronic Marketplace (MK- BEEM)

É uma plataforma de comércio eletrônico multilíngue e multicultural centralizado para aplicações B2C. Demonstra que o acoplamento entre tecnologias de representação de conhecimentos com aqueles do tratamento automático da informação podem - nos domínios comerciais simples - oferecer serviços de tradução e de interpretação de grande qualidade e operacionais a curto prazo, tais como:

- Representação das consultas em linguagem natural em uma representação semântica (ontologia);
- Manutenção fácil dos catálogos de produtos e serviços multilíngues;
- Fácil criação de ofertas de produtos e serviços compostos;
- Pesquisa em linguagem natural de produtos ou de serviços através de conteúdo semântico;
- Categorização e indexação automáticas dos produtos ou dos serviços descritos em

linguagem natural;

- Integração fácil e rápida de novas ofertas de produtos ou de serviços em um contexto multilíngue e multicultural.

2.3.3 *Discussão*

A visão operacional da plataforma Web Semântica somente será aplicável quando as técnicas necessárias à sua implementação sejam combinadas e adaptadas para o contexto atual que se encontra seja similar àquela que viu nascer a Web (restrições tecnológicas, econômicas, políticas e humanas etc.).

Esta adaptação se fará em duas vias: primeiro sobre um plano técnico, devido a dificuldade de implementação de uma estrutura assim imponente, e segundo porque a abertura relativa à Web não ajuda em nada o trabalho dos normalizadores. A Web Semântica continua uma perspectiva concreta e realizável, e que muitos aspiram a trabalhar. Trabalhos estes principalmente orientados a:

- Padronização: Seja ao nível das linguagens utilizadas, da estrutura dessas linguagens, da semântica das ontologias utilizadas, sejam genéricas ou específicas aos domínios particulares, mas também os mecanismos adotados pelos serviços Web ou os métodos de armazenamento e de manipulação de dados. A Web Semântica é destinada em parte para ser manipulada pelos agentes “inteligentes”, mas não menos automáticos, portanto, um certo nível de formalismo deve ser alcançado.
- Adaptação e mediação: A padronização não é sempre possível. Como foi exposto, excesso de rigidez na Web compromete a expressividade e a abertura da Web. Tecnicamente falando, é ilusório pensar em uma formalização total que assegure a eficiência e a plenitude na teia. É por isso que é importante desenvolver sistemas de mediação de diferentes níveis, entre linguagens (comunicação e transformações entre linguagens) ou entre conceitos, o mesmo para estruturas mais complexas (como foi visto com os sistemas mediadores).
- Evolução e desenvolvimento: Isto será feito em paralelo e deverá assegurar as soluções eficazes aos problemas que surgirão através da implantação desta estrutura.

O segundo eixo é certamente o mais delicado, se trata de se fazer aceitar esta plataforma pelos diferentes usuários e organizações. Este parâmetro deve ser regido pela regra da necessidade, daí a necessidade de destacar as aplicações da Web Semântica e de lhes orientar de uma perspectiva:

- Qualitativa: Os usuários estão interessados em eficácia e originalidade das aplicações e dos serviços que lhes são propostos, e que lhes permitam realizar as tarefas relativamente complexas (na Web atual) com um mínimo de esforço e de investimento, porque deve-se notar que, mesmo os trabalhos simples como a anotação de documentos, dificilmente são parte do cotidiano da maioria dos desenvolvedores da Web, é necessário então pensar em aumentar a ergonomia das aplicações da Web Semântica e sobretudo, o aspecto semi-automático das ferramentas e processos de desenvolvimento.
- Substancial: As organizações deverão ter um interesse real (financeiro) nesta plataforma, que lhes permitirá uma melhor gestão de seus recursos, uma comunicação maior, um desenvolvimento de serviços mais pertinentes e menos dispendioso para serem mais produtivos e competitivos.

Para finalizar e no contexto do presente trabalho, ressalta-se as escolhas no que concerne as tecnologias da Web Semântica devem levar a necessidade de explorar outros caminhos e abordagens, sobretudo no que concerne as linguagens de representação e as técnicas de exploração - essencialmente inferenciais - ligados aos seus formalismos.

2.4 Os formalismos de Representação

Uma das maiores ambições da informática moderna é de tornar as máquinas capazes de imitar o homem em seu processo de raciocínio. Desde a máquina de cálculo universal de Babbage (1792-1871) à aparição do primeiro computador (ENIAC 1950) e passando pela máquina abstrata de Turing (1912-1954), as tentativas foram realizadas nessa sequência, sem portanto alcançar o objetivo desejado porque todas essas abordagens estavam interessadas

somente em problemas de cálculo e suas derivadas no contexto dos algoritmos clássicos, sem estudar realmente a natureza do raciocínio humano (FREITAS, 2003).

Significa a concepção de máquinas capazes de perceber o ambiente no qual elas evoluem e que oferecem a possibilidade de resolver problemas complexos de maneira “inteligente”, oferecendo as plataformas intuitivas com a finalidade de melhor responder ao trato com seus interlocutores humanos.

As etapas que se seguiram nesse sentido foram primeiramente globalizantes, como foi o General Problem Solver de Newell e Simon, cuja abordagem resultou em um fracasso retumbante devido a orientação em direção aos sistemas ditos especialistas. Este fato deu nascimento às implementações tais como Dendral (1969) da Universidade de Stanford ou ainda MYCIN, que beneficiou em 1979 da primeira execução do motor de inferência (parte destinada à resolução de problemas) da base de conhecimentos por Van Melle. Através de sua eficácia, demonstrou as vantagens desta nova abordagem que consiste na separação entre os programas e os conhecimentos, dando assim nascimento aos sistemas baseados em conhecimento ou SBC.

A partir disto, a representação de conhecimentos que será abordada nesta seção, e os mecanismos de inferência - que será visto na Seção seguinte - se tornaram os eixos de pesquisa principais em inteligência artificial.

O que nos interessa é a representação de conhecimentos e os diferentes formalismos que a compõem. Naturalmente não será abordada a totalidade dos formalismos de representação existentes porque eles estão - na maioria - fora do contexto deste trabalho. Será abordado mais particularmente o aspecto formal das representações clássicas, bem como do crescente interesse oferecido pelas representações estruturadas.

2.4.1 Formalismos de Representação

A representação é a aproximação abstrata que descreve os objetos do mundo que nos cerca, podendo ser vista como um “resumo” das experiências e dos dados utilizados para (DOMINÉ, 1988; FEIGENBAUM; BARR; COHEN, 1989):

- Permitir o compartilhamento destas experiências;

- Permitir uma manipulação teórica (menos cara, segura, possível, acessível) do nosso saber e dos objetos do universo para um avanço tecnológico mais rápido e a elaboração de novos tratamentos;
- Desobrigar o usuário de dados desnecessários oferecendo assim uma melhor representação mental das relações entre objetos manipulados, para identificar os mais relevantes e deduzir outros novos.

O discurso (linguagem, fala), os mapas, os livros, os globos terrestres, ou mesmo as fotografias são as representações mais utilizadas em nossos dias, ao ponto de nos fazer esquecer que são meras representações. As matemáticas também estão entre as formas de representação mais universais e mais formais, cuja contribuição para a civilização humana é inegável (KAYSER, 1997).

Desta forma, é possível destacar que certas representações podem ser qualificadas como informais devido suas interpretações múltiplas como as piadas que representam situações engraçadas para alguns mas não para outros, ou como um quadro que deveria expressar os sentimentos de seu autor mas que pode ser interpretada de mil formas diferentes. E outras formais onde o significado foi estudado e conciliado entre diferentes partes que as utilizam tais como uma fórmula matemática descrevendo um processo seguindo parâmetros, ou um mapa que devido a seu índice permite uma interpretação não ambígua de seu conteúdo.

Entretanto, não deve ser confundido representação ruim e interpretação ruim, pois estes dois pontos dependem do usuário (experiência, ponto de vista, concentração) e da utilização da representação.

Serão apresentados a seguir diferentes métodos de representação de conhecimentos em IA que poderão ser descritos como sendo mapeamentos (correspondência) de conhecimentos e de estruturas de dados de informática representando-as, permitindo assim, devido a normalização de seu significado, definir os mecanismos de manipulação, de combinação e de interpretação, para atingir uma determinada finalidade (SOWA et al., 2000).

A descrição destes formalismos se baseia sobre a especificação de três componentes:

- A linguagem de representação: são definidas as estruturas elementares abstratas utilizadas na representação (alfabeto), seus utilizadores e as regras de construção sintática (gramática) de fórmulas fazendo referência aos objetos do mundo modelado,

permitindo assim sua descrição e sua manipulação;

- Os sistemas de dedução associados: necessário não confundir com as regras de composição sintática, pois eles permitem deduzir novos conhecimentos a partir de antigos e isto através de mecanismos chamados dedutivos através da aplicação de regras chamadas regras de inferência;
- Regras de avaliação: permitem associar à cada fórmula descrita pela linguagem de representação um valor de um conjunto, e isto de acordo com o uso que deve ser feito - e onde a utilização não se limita aos tradicionais valores booleanos.

Se faz necessário especificar os critérios que vão ajudar a avaliar estes formalismos, critérios gerais como a velocidade do cálculo ou a otimização do espaço que permanece em vigor. Também são necessários critérios mais específicos, especialmente sobre as linguagens com uma lógica (COHEN, 1990):

- A decidibilidade: Se para toda fórmula da linguagem existe um algoritmo que pode afirmar em um tempo de cálculo finito se esta fórmula é ou não um teorema desta lógica (demonstrável sem suposições);
- A correção: Se todos os teoremas desta lógica são tautologias (tudo o que é demonstrável é verdadeiro);
- A completude: propriedade recíproca da correção. Existe completude se todas as fórmulas tautológicas são teoremas da lógica adotada pela linguagem (tudo o que é verdadeiro é demonstrável).

Para o presente trabalho, são mais importantes as representações baseadas sobre o formalismo das lógicas clássicas (proposicionais e predicativas) mas também sobre os formalismos de representação estruturada tais como as redes semânticas, os frames, ou ainda os grafos conceituais sobre os quais será baseado.

2.4.1.1 As lógicas clássicas

A origem das lógicas clássicas se encontra na lógica proposicional. Vários aperfeiçoamentos foram feitos, introduzindo assim a lógica de predicados a fim de aumentar a taxa de expressividade e de representatividade dos conhecimentos neste modelo. Na sequência serão apresentados estes tipos de representação expondo seus fundamentos básicos, suas sintaxes, assim como os sistemas dedutivos que lhes são atribuídos. Será finalizado com uma breve

crítica sobre cada tipo de representação.

2.4.1.1.1 A Lógica Proposicional

Uma proposição é o enunciado (declaração) de um fato sobre o qual é possível atribuir um valor de verdade. A sintaxe da linguagem proposicional pode ser dividida aproximadamente como segue (ARJONA; SÁNCHEZ, 2004):

1. Um conjunto de símbolos proposicionais: conjunto infinito enumerável composto de letras alfabéticas;
2. Um conjunto de símbolos lógicos: \neg , \wedge , \vee , \Rightarrow etc.

O princípio de representação é decompor os fatos complexos do mundo e das entidades proposicionais simples ligadas pelos conteúdos lógicos, por exemplo, a afirmação seguinte: “Se ele está bem treinado e está confiante, então ele vencerá a corrida.” Pode ser traduzida simplesmente por uma fórmula lógica do tipo:

$P \wedge Q \Rightarrow S$, onde:

- P é a proposição: “Ele está bem treinado.”
- Q é a proposição: “Ele está confiante.”
- A consequência S é a proposição: “Ele vencerá a corrida.”

Estas proposições estão submetidas às regras de validação permitindo lhes atribuir uma valoração e isto devido a uma função V que vai do conjunto das proposições aos valores de verdade *Verdadeiro*, *Falso*. Se a fórmula a é atômica, sua valoração será $V(a)$, se ela é composta, sua valoração é consequência de diferentes composições de suas fórmulas atômicas seguindo as tabelas-verdade usuais da lógica booleana (COHEN, 1990), como mostra a Figura 11:

Figura 11 - Tabela verdade Lógica Booleana

P	Q	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$\neg P$
V	V	V	V	V	F
V	F	F	V	F	F
F	V	F	V	V	V
F	F	F	F	V	V

Fonte: elaborada pela autora

Este sistema de validação define vários conceitos e propriedades sobre as fórmulas proposicionais, por exemplo:

- As tautologias: Uma fórmula ‘f’ é uma tautologia se e somente se: \forall valoração V temos: $V(f) = \text{verdadeiro}$.
- A equivalência: Duas fórmulas ‘f’ e ‘g’ são equivalentes se e somente se \forall valoração V temos: $V(f) = V(g)$.
- A causalidade: Seja Σ um conjunto de fórmulas, diz-se que uma fórmula ‘f’ é uma consequência de Σ se e somente se toda valoração que satisfaz todas as fórmulas de Σ satisfaz também ‘f’, o que é equivalente a dizer que o conjunto $\Sigma \cup \neg f$ não é satisfatório.

Para a dedução de novos conhecimentos, o sistema dedutivo para a lógica das proposições - e nas lógicas clássicas em geral - utiliza:

- Um conjunto de proposições básicas chamadas axiomas: essas proposições são admitidas e definidas como verdadeiras. Nota-se também a presença de mecanismos - como a substituição - permitido a dedução do conjunto de fórmulas que lhes são equivalentes e, como consequência, não necessitam ser demonstradas, que são as “premissas do raciocínio”.
- Um conjunto de regras de dedução: são os regentes do processo de prova lógica, baseiam-se no princípio da consequência e permitem encontrar por dedução novas fórmulas à partir dos axiomas. Por exemplo:
 - i.** A substituição: consiste em substituir em uma fórmula ‘f’ (composta) todas as ocorrências de uma fórmula ‘p’ por outra ‘g’ que é equivalente e assim obter uma nova fórmula ‘g’.
 - ii.** Regras de comutação: a partir de duas cláusulas C1 e C2 - uma cláusula é uma disjunção de fórmulas atômicas e da negação de fórmulas atômicas - contendo a proposição ‘p’ tal que ela aparece positivamente na primeira e negativamente na outra ($\neg p$), pode-se inferir uma nova cláusula C que será consequência lógica de C1,C2 e definida por: $C \equiv C1-p \vee C2-p$.

Na prática, a manipulação das fórmulas se faz depois de tê-las transformado em cláusulas de

Horn. As cláusulas de Horn são as cláusulas que compreendem no máximo somente uma fórmula proposicional positiva. Esta transformação não é obrigatória mas reduz consideravelmente o tempo do cálculo (HAARSLEV; MÖLLER, 2003b).

Discussão

A lógica das proposições é relevante porque, além do fato de expressar de maneira simples os fatos, ela também é econômica. Existem modelos que permitem a representação na integralidade do formalismo com somente dois símbolos lógicos (a negação e a intersecção, por exemplo) e três regras de inferência (HAARSLEV; MÖLLER, 2003b). Entretanto, elas sofrem de uma falta de expressividade porque não dispõem de nenhum meio de distinguir os atores que atuam. Em outras palavras, não é possível descrever os indivíduos que compõem esta representação nem seus atributos ou relações entre eles. Por isto, os modelos de representação baseados na lógica predicativa foram introduzidos.

2.4.1.1.2 A Lógica de Predicados

A sintaxe de sua linguagem é similar àquela da lógica das proposições, ela se distingue pela introdução de novos símbolos relativos à manipulação de variáveis. A sintaxe se define então por (LEVADA, 2012):

- Um conjunto de símbolos de predicados e funções às quais se associam os números naturais que correspondem à sua aridade.
- Um conjunto de símbolos de variáveis e constantes.
- Um conjunto de símbolos lógicos: \neg , \Rightarrow , \forall , . . . etc.

Assim é possível expressar fatos tais como

“todo número inteiro é real” ou

“alguns cursos são interessantes” ou ainda

“Pedro é o pai de Paulo”

distinguindo os indivíduos inteiro, real, curso, Pedro, Paulo e as propriedades que lhes descrevem. A representação em lógica de primeira ordem ficaria assim (MINEAU, 2000):

$\forall x \text{ inteiro} \Rightarrow \text{real}(x)$

$\exists y \text{ curso}(y) \wedge \text{interessante}(y)$

$\text{Pai_de}(\text{Pedro}, \text{Paulo})$

Onde \forall é o quantificador universal e \exists o quantificador existencial. ‘Inteiro’, ‘real’, ‘curso’ e ‘interessante’ são predicados de aridade 1 e Pai_de de aridade 2. x e y são variáveis. ‘Pedro’ e ‘Paulo’ são símbolos de constantes. Deve-se notar que essas representações são essencialmente intuitivas. Seria possível utilizar um predicado unário ‘Pedro’ tal que $\text{Pedro}(\text{Paulo})$ seria verdadeiro se Paulo for filho de Pedro. Da mesma forma, é possível demonstrar que todo predicado n -ário pode ser decomposto em uma conjunção de predicados binários. Exemplo:

$\text{Passaro}(\text{Lili}) \rightarrow \text{eh_um}(\text{Passaro}, \text{Lili})$

$\text{Ir}(\text{Pedro}, \text{Brasilia}, \text{Rio}) \rightarrow \text{Agente}(\text{Pedro}, \text{Ir}) \wedge \text{Partida}(\text{Brasilia}, \text{Ir}) \wedge \text{Destino}(\text{Rio}, \text{Ir})$

O sistema dedutivo da lógica de predicados incorpora os princípios da lógica de proposições, se baseia sobre uma infinidade de axiomas reagrupados sob formas de esquemas gerais para construir, por dedução, o conjunto de fórmulas da linguagem (MINEAU, 2002).

É necessário, entretanto, atuar sobre uma sintaxe de fórmulas diferentes. Se fazem necessários os termos (variáveis, constantes e composição de funções) que são representações dos objetos do domínio, e das fórmulas predicativas que são as relações entre estes termos.

A valoração das fórmulas difere um pouco porque é necessário atribuir uma interpretação aos múltiplos predicados e funções nas fórmulas, a valoração será, então, em relação à uma interpretação dada (uma aplicação que conecta cada termo ao objeto que representa).

O processo de prova mantém as regras e as técnicas de inferência precedentes como o *modus ponens* e a *transitividade* da implicação se adaptando às fórmulas predicativas. Existem também novas regras como a *generalização* que é relativa ao *quantificador universal*.

Criticando as lógicas clássicas seguindo os critérios descritos anteriormente, pode-se dizer que a lógica de proposições é ao mesmo tempo correta, completa e decidível. A lógica de primeira ordem apresenta somente uma semi-decidibilidade. Foi demonstrado que é possível deduzir que uma fórmula é um teorema em um tempo finito, mas não é possível dizer o inverso, quer dizer que não existe um algoritmo que permita afirmar que, em um número de etapas finito, uma fórmula não é um teorema desta lógica.

Discussão

A lógica de primeira ordem apresenta vantagens inegáveis, ela oferece a vantagem de uma representação simples e natural e oferece uma grande liberdade intuitiva, o que é louvável em matéria de expressividade ou de otimização. Ela dispõe, além disso, de um formalismo rigoroso assim que um suporte de validação e de inferências sólidas, ademais o número de linguagens e sistemas inferenciais que são baseados sobre seu formalismo. Foi dado anteriormente, um exemplo com RDF utilizando literalmente uma forma predicativa para a apresentação de suas triplas (FENSEL et al., 2002).

Se observa, entretanto, que seus pontos fortes induzem também a algumas lacunas, pois a abordagem muito intuitiva reduz o rigor do formalismo. Além disso, sua expressividade é limitada quando se trata de representar certos aspectos cognitivos relativos aos conhecimentos não afirmativos (dúvida, crença, desejo, incerteza etc.), da mesma forma os conhecimentos de ordem espacial ou temporal. Por isso, as extensões foram feitas através das lógicas modais ou temporais para remediar este problema (lógicas não tratadas no presente trabalho).

A lógica de primeira ordem sofre também por não poder representar um reagrupamento hierarquizado das diferentes entidades que a compõem (sistemas de hierarquização ou de indexação). Assim, para definir um indivíduo em particular, é necessário pesquisar e coletar conjuntos de afirmações onde ele aparece, o que corre o risco de ser pernicioso em termos de memória e tempo de cálculo.

Em relação ao sistema de dedução, ele apresenta um rigor que é louvável mas não menos restritivo porque não permite atuar sobre os dados completamente definidos em um ambiente totalmente controlado, enquanto na prática somos frequentemente confrontados às situações onde existe somente fragmentos de informações, até mesmo informações errôneas, para os quais os mecanismos de inferência clássica não são adaptados.

A lógica de primeira ordem é, então, uma ferramenta eficaz no momento em que é utilizada para sistemas completamente definidos e onde todas as entidades que lhe compõem são conhecidas, estáveis e invariáveis.

2.4.1.2 As Representações Estruturadas

2.4.1.2.1 Os Frames

Em 1975 Minsky propõe a idéia de representar as categorias de objetos através da descrição de suas propriedades e dos diferentes fatos relacionados à sua utilização nas estruturas chamadas *frames*; permitindo assim, através de sua instanciação, construir diferentes atores do mundo modelado. Os *frames* se baseiam sobre vários suportes de descrição, tais como (GROSOF et al., 2003):

- Mecanismos de ligação entre *frames*, permitindo a fixação de frames à outros mais genéricos e isto através da utilização de duas relações seguintes:
 - **Is_a** : permite fixar um modelo de frame (descrição de categoria) à um outro frame mais geral.
Exemplo: *cadeira is_movel*
 - **Ako (A kind of)**: permite fixar um frame individual ao seu frame instanciador.
Exemplo: *A cadeira_de_Paulo A kind of cadeira*
- A descrição das propriedades através das estruturas declaradas no próprio frame, suas estruturas são chamadas *slots*. Exemplo:

O frame cadeira tem slots como: número de pés, estilo, número de braços etc.

- As diferentes informações sobre os slots (descritivas e procedurais) são descritas sob forma de *facetas*, permitindo definir:
 - O domínio de valores permitidos para o slot;
 - O valor do slot para uma frame individual;
 - O valor a atribuir como padrão ao slot;
 - Os procedimentos a serem invocados para a adição, a supressão e a modificação do valor de um slot (tais como a utilização do valor como padrão ou o questionamento do usuário).

Exemplo:

Frame cadeira

is_: Movel

Numero de pés: **deve ser** um inteiro positivo.

padrão 4

se removido chamar a função reparar pés

Estilo: **dever ser** acolchado

se adicionar avisar o usuário

Frame cadeira_recamier

Kind of: cadeira

Numero de pés: 4

Estilo: acolchado

Numero de braço: 0

A representação em *frames* oferece várias vantagens tais como:

- A clareza e a coerência das estruturas propostas que, pela representação intuitiva dos objetos sob essa forma, facilitam a declaração e a compreensão dos conhecimentos modelados pelos usuários;
- A não redundância das declarações permitidas pelos mecanismos de herança definidas devido as relações inter-frames *is_a* e *ako*;
- A fixação dos procedimentos aos slots de frames permite uma automatização dos tratamentos sobre eles atribuindo, por exemplo, capacidades de interatividade com o usuário;
- Permite alguns tratamentos da lógica dos padrões pelo fato de especificar os valores por padrão.

Entretanto, a rigidez das estruturas e sua falta de adaptação não permite uma grande liberdade de expressão e a dificuldade de modela-los ou de faze-los evoluir desde que as representações atinjam um certo nível de complexidade (árvore de herança muito grande, declarações volumosas etc).

É possível também notar as semelhanças entre o formalismo dos frames e aqueles da Orientação à Objetos que utilizam a noção estrutural do primeiro, refina os diferentes mecanismos de heranças (a herança múltipla) e introduz a noção de encapsulamento de dados para poder oferecer um nível de abstração mais elevado, e onde o sucesso pode ser visto no número de linguagens de programação usando sua semântica (Smalltalk, C++, Java etc.) (NARDI; BRACHMAN, 2003).

2.4.1.2.2 As redes semânticas

Para remediar o problema da computabilidade das abordagens lógicas essencialmente devido a sua característica indecidível ou semi-indecidível (mais especialmente para as lógicas não-clássicas) assim como a lentidão dos processos de cálculo ligados à seus sistemas dedutivos completos (como no caso da lógica de primeira ordem) duas soluções foram propostas:

- Interesse nos sub-conjuntos destas linguagens, para encontrar um meio termo entre expressividade e formalismo, atribuindo a possibilidade de lhes associar os algoritmos dedutivos completos e rápidos;
- Interesse nos sistemas de dedução incompletos, permitindo as manipulações simples e eficazes.

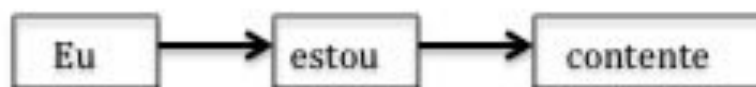
As redes semânticas nasceram da convergência destas duas abordagens, no início para a modelagem do funcionamento da memória e da compreensão da linguagem natural (Quillian 1968), (COLLINS; QUILLIAN, 1969). Sua filosofia se baseia sobre a representação dos conhecimentos devido aos grafos, e isto utilizando as diversas ferramentas que lhes são inerentes, tais como:

- A rotulação de nós e arcos;
- A orientação dos arcos e os conceitos de grafos múltiplos;
- Os diferentes algoritmos de análise, de pesquisa e transformação estabelecidos pela teoria dos grafos.

As primeiras utilizações desta modelagem gráfica faltava formalismo semântico, baseando a interpretação do significado dos nós e dos arcos das redes sob a intuição do usuário com o surgimento de diversas questões a respeito do mérito desta abordagem devido a confusão a que levou.

Para remediar, a solução mais radical foi associar a todo objeto da rede (nós, arcos, subgrafos etc.) um correspondente fixo nas representações lógicas, formalizando-os mas os transformando ao mesmo tempo em simples variantes denotacionais de representações lógicas, privando-os assim de toda sua originalidade como suporte de representações à interpretações múltiplas e evolutivas. Exemplo: a frase “Eu estou contente.” pode ser modelada pela rede semântica, como ilustra a Figura 12:

Figura 12 - Rede Semântica



Fonte: elaborada pela autora

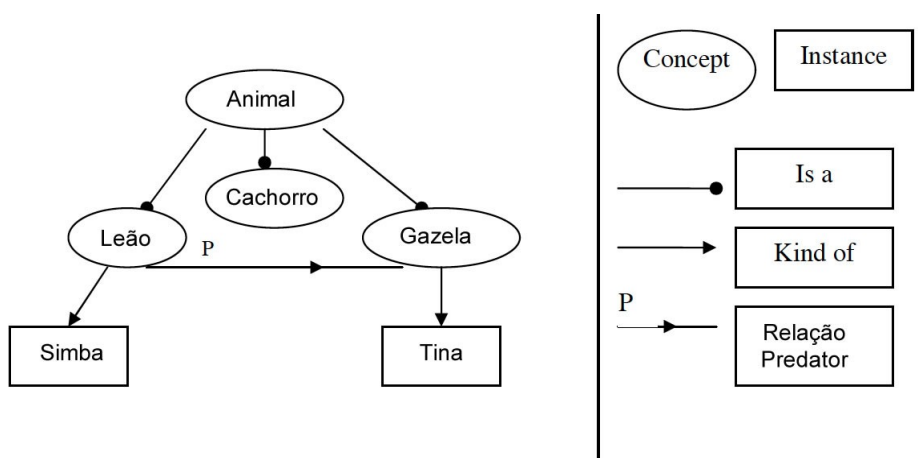
Onde a semântica dos nós é o que define as palavras da frase, e dos arcos, a sequência de palavras.

Este exemplo ilustra o fato de que nenhuma restrição é feita sobre a forma de construir uma rede semântica ou sobre o grau de formalismo que ele deve possuir, nem sobre a semântica. Mas isto não impede a definição das primitivas de interpretação (por uma questão de modularidade) permitindo uma utilização independente dos elementos da representação. Por exemplo, o fato de que dois nós A e B estejam ligados por um arco R significa de maneira geral o fato que as entidades representadas pelos dois nós (conceitos, indivíduos, propriedades) verificam a relação R. É possível, da mesma forma, definir alguns tipos de relações usuais tais como (CORCHO; FERNANDEZ-LOPEZ; GOMEZ-PEREZ, 2007):

- A relação *is_a* (hierarquização entre classes e conceitos);
- A relação *kind_of* (relação de instanciação de indivíduos pelos conceitos);
- As relações de propriedades (ligando dois conceitos, e definindo um como propriedade do outro).

A Figura 13 ilustra um exemplo:

Figura 13 - Tipos de Relações



Fonte: elaborada pela autora

Discussão

A modelagem física - sob forma de nós e arcos - dos objetos e das relações entre objetos contribuem para os processos cognitivos de representação mental e da vinculação dos diferentes conhecimentos representados pelos usuários permitindo assim definir de maneira mais fácil os mecanismos de tratamento e de extração de dados frequentemente mais eficazes pois mais direcionados, mas também menos formais e portanto, pouco demonstráveis.

A validação das relações - como as apresentadas acima - requerem a tipagem dos nós e arcos - onde a distinção das representações dos indivíduos e conceitos parecem ser de utilização elementar - assim como os mecanismos de construção de grafos (ou a gramática dos grafos aceitos) de graus de rigor e de expressividade variados, e onde serão apresentados em seguida, os mais conhecidos entre eles: as lógicas de descrição e os grafos conceituais.

a) As Lógicas de Descrições

As lógicas de descrições (LDs), conhecidas também com o nome de lógicas terminológicas, são o fruto de estudos conduzidos sobre as redes semânticas e sob a herança estruturada (Brachman 77), mais precisamente do sistema KL-One (Brachman & al 78),(SCHMOLZE; LIPKIS, 1983) da linguagem dos Frames. Elas se beneficiam das redes KL-One (o reconhecimento dos elementos da rede especificando os nós dos conceitos e os papéis) e vieram superar sua falta de rigor semântico utilizando uma semântica formal baseada na lógica predicativa. Isto permite também se beneficiar da representação linear dos conhecimentos sob forma de fórmulas além de oferecer as vantagens de uma representação gráfica.

- Fundamentos e estrutura

As LDs são utilizadas para representar os conhecimentos sobre um modelo do domínio de aplicação (mundo), seu formalismo se baseia a princípio, sobre a representação conceitual das entidades que lhe compõe e sua organização em hierarquia de generalização/especialização ou taxonomia (MAEDCHE et al., 2003). Estas operações compõem a parte terminológica das LDs e permitem, entre outras coisas, ter uma visão

global sobre a composição da base de conhecimentos, e a classificação contribui, por sua vez, a melhorar o desempenho das inferências.

A segunda parte é aquela que se interessa nos indivíduos, à suas propriedades e as relações entre as entidades terminológicas, esta parte permite definir o estado da atividade do mundo com base nos indivíduos, conceitos e tarefas definidas na terminologia.

- O componente terminológico

Serão definidos os conceitos e as relações entre eles. Os conceitos são as estruturas abstratas que representam as classes de objetos manipuladas no domínio modelado, os papéis apresentam as propriedades e as relações entre os conceitos. Os conceitos e papéis podem ser atômicos ou complexos.

Os conceitos atômicos são aqueles em que não é possível descrever de forma completa e concisa, eles são o elemento básico para a definição do restante da terminologia. Exemplo: Pessoa, Animal, Planta etc. Brachman propõe uma abordagem de concepção baseada sob três mecanismos (HAARSLEV; MÖLLER, 2003a):

- Os elementos básicos são os conceitos atômicos e os papéis atômicos considerados respectivamente como os predicados unários e dos predicados binários.
- Limita-se a expressividade das linguagens à um número restrito mas suficiente de construtores que servirão para definir o restante.
- Os conhecimentos implícitos relativos aos conceitos e indivíduos poderão ser induzidos por inferência.

Os conceitos atômicos não são suficientemente expressivos, por isto, são frequentemente utilizados para construir conceitos mais complexos por meio de construtores conceituais.

As LDs se caracterizam pela riqueza dos construtores que elas dispõem. Diferencia-se geralmente uma LD de uma outra seguindo os construtores que elas usam. Segue uma tabela apresentando alguns construtores conceituais muito utilizados nas lógicas de descrição. Seja:

- Ψ O domínio da interpretação.
- \mathcal{I} É uma interpretação sobre Ψ , tudo o que segue é em relação à ele.
- A Conceito atômico.
- CD Conceitos complexos.

- PS Papéis atômicos.

Quadro 1 – Alguns construtores conceituais das Lógicas de Descrição

$T \uparrow$	$\Psi \uparrow$	Conceito universal
$\perp \uparrow$	\emptyset	Conceito absurdo
$(\neg A \uparrow)$	$\Psi \uparrow / A \uparrow$	Negação de um conceito atômico
$(C \sqcap D) \uparrow$	$C \uparrow \cap D \uparrow$	Intersecção entre conceitos
$(C \sqcup D) \uparrow$	$C \uparrow \cup D \uparrow$	(U) União de conceitos
$(\forall R.C) \uparrow$	$a \in \Psi \uparrow / \forall b.(a,b) \in R \uparrow \rightarrow b \in C \uparrow$	Restrição de valores
$(\exists R.T) \uparrow$	$a \in \Psi \uparrow / b.(a,b) \in R \uparrow$	Quantificação existencial restrita
$(\exists R.C) \uparrow$	$a \in \Psi \uparrow / b.(a,b) \in R \uparrow \wedge b \in C \uparrow$	(E) Quantificação existencial completa
$(\geq nR) \uparrow$	$a \in \Psi \uparrow / b / (a,b) \in R \uparrow \geq n$	(N) Restrição de cardinalidade superior
$(\leq nR) \uparrow$	$a \in \Psi \uparrow / b / (a,b) \in R \uparrow \leq n$	(N) Restrição de cardinalidade inferior
$(\neg C \uparrow)$	$\Psi \uparrow / c \uparrow$	(C) Complemento do conceito

Fonte: elaborada pela autora

É possível também definir os conceitos como “fazendeiro” que é uma pessoa que cria gado, em outras palavras “pessoa \sqcap criaGado.T”, ou para pessoas que não são “fazendeiras” pela definição “pessoa $\sqcap \forall$ criaGado. \perp ”. “criaGado” é um papel atômico.

Os construtores assinalados (*) são conhecidos como construtores da linguagem AL (Linguagem Atributiva) proposta por Schmidt-SchauB & Smolka em 1991 como sendo a linguagem mínima apresentando um interesse prático e à título de exemplo a linguagem e a remoção da quantificação existencial permite definir a linguagem FL^o .

Os símbolos (E), (N), (C), (U) servem para descrever uma linguagem baseada sobre AL a qual foi enriquecida a expressividade por uma ou mais das regras correspondentes. Nota-se, por exemplo, ALC uma linguagem baseada em AL admitindo o complemento de conceitos (HAARSLEV; MÖLLER, 2003a). Existem outros construtores para as definições de papéis complexos como:

Quadro 2 - Construtores para definições de papéis complexos

A intersecção de papéis: $(R \sqcap S) \uparrow = R \uparrow \cap S \uparrow$.
A união de papéis: $(R \sqcup S) \uparrow = R \uparrow \cup S \uparrow$.

A negação de um papel: $(\neg R) \uparrow = \Psi \uparrow X \Psi \uparrow R \uparrow$.
O inverso de um papel: $R^- = (b,a) \in \Psi \uparrow X \Psi \uparrow / (a,b) \in R \uparrow$.

Fonte: elaborada pela autora

Assim como outros, como a composição de papéis (R^oS), o fechamento transitivo R^+ etc. Nota-se que as equivalências induzidas pelo formalismo lógico continua válido para o caso dos construtores de conceitos, por exemplo:

$C \sqcup D$ é equivalente à $\neg(\neg C \neg \sqcap D)$

$\exists R.C$ é equivalente à $\neg(\forall R.\neg C)$

Entre outras coisas, isso permite passar de uma linguagem à outra por substituição de um ou vários construtores através de suas lógicas equivalentes (como através dos construtores de união e de intersecção de conceitos). A tabela seguinte define algumas das DLs mais conhecidas, definidas pela lista de seu construtor terminológico (HAARSLEV; MÖLLER, 2004):

Quadro 3 - Marcadores para o catálogo de indivíduos

A ling. T-Box	BACK	KRIPTON	KANDO	CLASSIC	LOO	NIKL
C primitivo	X	X	X	X	X	X
C e D	X	X	X	X	X	X
C ou D					X	
Não C	X				X	
Todo R C	X	X	X	X	X	X
Algum C					X	
R no mínimo n	X	X	X	X	X	X
R no máximo n	X	X	X	X	X	X
Max u				X	X	
Mín u				X	X	
Rvm R1,R2				X	X	X
Rpreenche $(i_1,..,i_n)$				X	X	
Um dos $(i_1,..,i_n)$	X			X	X	
Inverso de R		X			X	X
Cadeia $(R_1,..,R_n)$	X	X			X	
Cadeia $(R_1,..,R_n)$	X	X			X	
Resto R C	X				X	X
Domínio C	X		X		X	X
Range C					X	X
R1 e R2						
Não R						
R1 ou R2						

Fonte: elaborada pela autora

Isto define as ferramentas necessárias para a composição dos conceitos e dos papéis com a finalidade de definir novos, mais complexos ou mais expressivos, faltando somente apresentar os métodos de geração destas definições ou os axiomas terminológicos.

- Axiomas terminológicos

- O axioma da definição

Uma igualdade onde o membro esquerdo é um conceito (resp.papel) atômico é chamado “Definição”. Ela serve para atribuir um nome simbólico à uma descrição conceitual complexa e é denotada: $C \equiv D$ resp. $R \equiv S$. Exemplo:

$Bacharel \equiv Pessoa \sqcup \exists \text{ PossuiDiploma. } Bacharelado$

$\text{Pais “nome simbólico”} \equiv \text{Pai} \sqcup \text{Mãe “Descrição complexa”}$

$\text{FilhoUnico} \equiv \text{Pessoa} \sqcap \exists \text{ um_irmao. } \perp$

São possíveis as seguintes definições (SANTOS et al., 2005):

- Chama-se um conjunto finito de definições onde cada nome simbólico é definido somente uma vez numa terminologia, comumente definida: T_Box .
- Os elementos da terminologia aparecem somente do lado direito das definições representando os conceitos primitivos. Os que aparecem à esquerda são os conceitos definidos.

Os conceitos primitivos são os conceitos sobre os quais dispõem as condições necessárias para a adesão; para os conceitos definidos, ao contrário, dispõem-se de uma definição completa e de condições necessárias e suficientes. Quanto a definição de um conceito, encontra-se um problema devido a recursividade na definição. Exemplo: $\text{Fator} \equiv \text{Pessoa} \sqcap \text{a_trabalho.Fator}$

Trabalhar sobre as terminologias que comportam os ciclos não é desejável, existem os métodos que permitem resolver este problema (tais como a semântica do ponto fixo). Nos interessa somente no caso das terminologias acíclicas.

- O axioma da inclusão

Às vezes não é possível definir totalmente um conceito. Por exemplo, pode-se pretender definir o conceito “orientador”: pode-se dizer que um orientador é, a princípio, uma

“pessoa”, na sequência pode-se dar um traço característico de orientador, por exemplo “o orientador não está jamais satisfeito”. Entretanto a definição seguinte:

$$\text{Orientador} \equiv \text{Pessoa} \sqcap \forall \text{EhSatisfeito}.\perp$$

Neste gênero de caso define-se o conceito definindo as condições necessárias de pertinência à este último e que utiliza a inclusão (ARTALE; FRANCONI, 1998). Assim, define-se a inclusão como sendo um axioma onde o membro esquerdo é uma especialização do membro direito, e observa-se: $C \subseteq D$. Exemplo: orientador \subseteq pessoa.

Mas a utilização de tal axioma tem como consequência a redução da precisão das definições da terminologia. Existe a tendência de efetuar uma operação de normalização das definições em uma T_Box que é qualificada de generalizada. Se trata simplesmente de gerar uma terminologia τ na qual substitui-se as definições do tipo “ $A \subseteq C$ ” por outras do gênero “ $A \equiv \tilde{A} \sqcap C$ ” onde \tilde{A} é um conceito que combina todas as características específicas de A em relação à C . Chama-se \tilde{A} a normalização de A da mesma forma que τ será chamado normalização de T (HAARSLEV; MÖLLER, 2003b).

- O componente de declaração

Como foi citado anteriormente, a componente de declaração se interessa no estado do domínio da atividade em termos de conceito e de papéis. Ele introduz a noção de indivíduo permitindo assim de lhe atribuir - usando afirmações - suas propriedades pertencentes aos conceitos ou classes de indivíduos e as relações com outros elementos do domínio. Existem dois tipos de declarações (HAARSLEV; MÖLLER, 2003b):

- Declaração de conceitos

São da forma: $C(a)$ ou $a:C$ ou ‘ C ’ é um conceito, ‘ a ’ é um indivíduo. Eles expressam o fato que o indivíduo ‘ a ’ pertence ao conceito ‘ C ’. Uma interpretação \mathcal{I} satisfaz $a:C$ se e somente se: $a\mathcal{I} \in C\mathcal{I}$. Exemplos:

Malfeitor (Antônio José)

Rezende (Ernesto) ; Rezende (Flávio).

- Declaração de papéis

São da forma: $R(b, c)$ ou $\langle b, c \rangle : R$ onde R é um papel e b, c são indivíduos. A semântica correspondente é que ‘ c ’ é um interpretador do papel R para o indivíduo ‘ b ’. Uma interpretação \mathcal{I} satisfaz $\langle b, c \rangle : R$ se e somente se: $\langle b^{\mathcal{I}}, c^{\mathcal{I}} \rangle \in R^{\mathcal{I}}$. Exemplo: $\text{um_filho}(\text{Raquel}, \text{Bruna})$ define que Bruna é filha de Raquel.

Um conjunto finito de declarações (de conceitos e/ou de papéis) corresponde à um componente de declaração e será chamado A_Box (HAARSLEV; MÖLLER, 2004). Nota-se que, às vezes, faz sentido empregar definições baseadas sobre os indivíduos. Alguns construtores terminológicos empregam este método, por exemplo:

- Os conjuntos: $a_1, a_2, a_3, \dots, a_n$ sendo os indivíduos
- Os interpretadores: $R : a = \{d \in \Psi \mid (d, a) \in R\}$

Geralmente quando fala-se de A_Box, subentende-se que o faz em relação à uma terminologia bem precisa, caso contrário, pode ser que admita-se as incoerências na base de conhecimentos, tomando como exemplo as declarações: $\text{FilhoUnico}(\text{Bruna})$ e $\text{IrmãoDe}(\text{Bruna}, \text{Cecilia})$, estas declarações são bem formadas e podem ser muito bem admitidas em uma mesma A_Box. Entretanto, se são definidas em relação à terminologia precedente (onde foram definidos os papéis utilizados) estaremos diante de uma inconsistência que resultará na rejeição destas proposições.

Será importante encontrar as interpretações que satisfaçam também todas as definições de T_Box assim como a totalidade das declarações de A_Box. Em outras palavras as interpretações que são por sua vez modelos para T_Box e A_Box. Este gênero de modelo permite definir uma boa abstração do domínio, respeitando as definições terminológicas e de declarações, não provocando nem ambiguidade e nem conflitos.

- As lógicas terminológicas e a lógica de predicados

A semântica ligada às LDs é formalizada segundo a lógica predicativa, com efeito se houver uma interpretação \mathcal{I} para a qual os conceitos atômicos serão assimilados aos predicados unários, e os papéis atômicos aos predicados binários e sabendo que se dispõe das ferramentas de composições precedentes e que são perfeitamente substituíveis através de seus equivalentes lógicos pode-se então verdadeiramente retranscrever não importa qual

conhecimento definido em LDs em fórmulas da lógica predicativa. Os diferentes elementos da LD serão então retranscritos em lógica de primeira ordem (LEVADA, 2012):

- Os conceitos atômicos: serão representados pelos predicados unários;
- Os papéis atômicos: serão representados pelos predicados binários;
- Os conceitos e papéis definidos - devido às fórmulas deduzidas da tradução dos construtores terminológicos tem-se alguns exemplos:

$$- \xi \exists R.C(y) = \exists x R(y, x) \wedge \xi c(x)$$

$$- \xi \forall R.C(y) = \forall x R(y, x) \rightarrow \xi c(x)$$

$$- \xi \geq nR(y) = \exists y_1, \dots, y_n. R(x, y_1) \wedge \dots \wedge R(x, y_n) \wedge \bigwedge_{i < j} y_i \neq y_j$$

$$- \xi \leq nR(y) = \exists y_1, \dots, y_n. R(x, y_1) \wedge \dots \wedge R(x, y_n) \rightarrow \bigvee_{i < j} y_i = y_j$$

- Os construtores: como \neg , \cap , \cup e seus representantes lógicos \neg , \wedge , \vee etc. .

- Classificação

A classificação é um método que consiste em estabelecer os vínculos entre os diferentes conceitos básicos de conhecimentos baseando-se em um processo incremental e iterativo reagrupando os conceitos seguindo as semelhanças que eles apresentam e isto a fim de modelar os conhecimentos segundo uma estrutura hierárquica organizada. Pode-se citar três abordagens de classificação:

- Categorização clássica

Ela se faz sobre as propriedades necessárias e suficientes de pertinência à uma categoria para reagrupar os conceitos, por exemplo na declaração seguinte: “há aqueles que têm uma arma carregada, e há aqueles que escavam” pode-se indicar duas categorias, para a primeira pode fazer corresponder os conceitos como militares, policiais etc. Entretanto esse tipo de classificação é impotente frente aos conhecimentos mal definidos como as propriedades não mensuráveis ou qualitativas (grande, pequena, bonita, ...)

Da mesma forma, frente aos conceitos que não apresentam propriedades necessárias e suficientes, se é definido um peixe para a propriedade vive no oceano e um mamífero como sendo uma criatura terrestre, não será possível saber em qual categoria classificar o conceito “Baleia”.

- A classificação por conceitos

Consiste em definir as classes pelos dados de descrições estruturais ou SD. Uma descrição estrutural é uma entidade conceitual descrevendo as propriedades características do conceito que se quer definir (HAARSLEV; MÖLLER, 2003b). A classificação se fará em seguida com base nessas descrições estruturais.

- A prototipagem

Esta abordagem define as classes de adesão seguindo as propriedades que elas apresentam geralmente e que constituem o protótipo da classe. Os objetos são representados por seus protótipos, e são considerados como sendo membros de uma classe dada se eles apresentam muitas similaridades com o protótipo desta classe. Tais objetos são ditos típicos desta classe, no caso inverso eles são atípicos. O que esta abordagem tem de particular é que ela considera as propriedades padrão como as propriedades na íntegra e não como resultado de exceções aos conceitos gerais. Exemplo: “Os criminosos são indivíduos violentos, mesquinhos e irremediavelmente perversos”. Esta descrição concordaria perfeitamente com aquela do indivíduo Ernesto Rezende que é considerado como um criminoso típico. Entretanto, o protótipo de indivíduo Flávio Rezende seria “dócil, ingênuo, até mesmo bobo e irremediavelmente gentil” o que faz dele um objeto atípico da classe dos criminosos.

Nota: Na prática, os algoritmos de classificação se baseiam sobre duas operações essenciais quando de inserção de um novo conceito na taxonomia. A primeira consiste em encontrar “os englobados” do conceito mais específico, a segunda se encarrega de encontrar todos os conceitos mais gerais que o último englobado. Não resta mais que inserir entre estes englobadores e englobados e atualizar os links.

- Inferências sobre as lógicas descritivas

Como foi exposto acima, as linguagens descritivas servem essencialmente para definir uma representação de uma dado domínio, o que se faz definindo os elementos que o compõe assim como o estado das atividades e dos atributos dos indivíduos. Mas acontece que certos conhecimentos são implícitos e não são apresentados em nenhuma das definições ou das declarações da base de conhecimentos, entretanto eles estão lá e se mostram geralmente importantes e úteis (SEBBEN et al., 2012).

A adição desses conhecimentos pode ser uma tarefa pouco desejável porque não será mais que encobrir de vantagem uma base de conhecimento geralmente muito pesada, por isso é preferível uma abordagem mais rentável que consiste em inferi-los de outros conhecimentos já representados.

Com seu aspecto estrutural, as LDs oferecem os mecanismos de inferências terminológicas com foco na criação de uma taxonomia de conceitos através de um processo de classificação - veja acima - e de outros de declarações que permitem definir entre outras coisas as classes de pertinência dos indivíduos da taxonomia.

No que segue, serão apresentados estes conceitos da mesma forma métodos que eles empregam. Citaremos também de um ponto importante que consiste na verificação da integridade dos conhecimentos representados pelos testes de satisfabilidade dos conceitos e da consistência das declarações que permitem determinar se a base de conhecimentos é ou não significativa (não contraditória em si mesma).

- Inferências terminológicas

Será apresentado agora as principais inferências terminológicas das lógicas de descrição, considerando uma terminologia acíclica τ e uma interpretação \mathcal{I} que é definida como sendo um modelo de τ .

- Satisfabilidade

Diz-se que um conceito C é satisfável em relação à uma terminologia τ se existe um modelo \mathcal{I} de τ pelo qual o conjunto dado por C \mathcal{I} é não vazio. Em outras palavras diz-se que C é satisfável se existe um modelo \mathcal{I} da terminologia que seja também um modelo de C .

O estudo da satisfabilidade diz respeito aos conceitos, ela permite no decorrer do processo da definição de um novo conceito determinar se este está em harmonia com os elementos pré-estabelecidos na terminologia, em outras palavras se este conceito não provoca contradições. Estes testes são importantes porque permitem evitar as incoerências na base de conhecimentos. Serão abordadas também outras inferências terminológicas que podem ser reduzidas aos problemas de satisfabilidade.

- Subclassificação (“*Subsumption*”)

Certamente uma das inferências capitais e características das lógicas de descrição e das descendentes de *KL-One*, se faz pela criação da teia de relação *is_a* entre todos os conceitos da terminologia. “Diz-se que *C* é subclassificado por *D* se e somente se os conjuntos das instâncias de *C* está incluído no conjunto de instâncias de *D* (ARTALE; FRANCONI, 1998)”.

Mais formalmente dizemos que *C* é englobada por *D* em relação à terminologia τ :

$$\tau \models C \subseteq D \text{ ou } C \subseteq_{\tau} D \text{ se e somente se } \forall \mathcal{I} \text{ modelo de } \tau \text{ tem-se: } C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$$

Esta definição corresponde à um tipo de subclassificação qualificada de *subclassificação extensional*, o raciocínio extencional permite definir os conceitos dando o conjunto de suas instâncias. Exemplo: as instâncias do conceito Pão (pão de manteiga, pão de cevada, pão doce etc.) são incluídas no conjunto das instâncias do conceito Alimento.

Woods 1991, descreve outros tipos de subclassificação, cita-se primeiramente a subclassificação intensional ou estrutural que é considerada como o inverso da subclassificação extensional. Afirma: “*C* subclassifica *D* se e somente se o conjunto das propriedades de *C* está no conjunto das propriedades de *D*”. O conceito subclassificado dispõe naturalmente das propriedades herdadas do conceito mais geral, seu conjunto de propriedades se completa em seguida pelas propriedades que lhes são específicas.

Um outro tipo é a subclassificação registrada, ela se baseia sobre a pesquisa de links diretos ou indiretos (via fechamento transitivo) sobre uma *Stored Taxonomic Structure* (STS) estrutura contendo os links de subclassificação previamente estabelecidos, a nova relação de subclassificação assim definida será incrementada na STS.

- Equivalência

Dois conceitos *C* e *D* são ditos equivalentes em relação à uma terminologia τ e escreve-se:

$$\tau \models C \equiv D \text{ ou seja } C \equiv_{\tau} D \text{ se e somente se } \forall \mathcal{I} \text{ modelo de } \tau \text{ temos: } C^{\mathcal{I}} = D^{\mathcal{I}} .$$

- Disjunção

Dois conceitos C e D são ditos disjuntos em relação à uma terminologia τ se:

$$\forall \mathcal{J} \text{ modelo de } \tau \text{ temos: } C \mathcal{J} \cap D \mathcal{J} = \emptyset.$$

Nota: Como foi citado anteriormente, é possível passar de um tipo de inferência à outra e assim reduzir os problemas ligados à igualdade de conceitos, por exemplo um problema de subclassificação ou de satisfabilidade; propõe-se, por exemplo - e devido sua importância - a formulação em termos de subclassificação:

- C é insatisfável $\Leftrightarrow C$ é subclassificado por \perp
- $C \& D$ são equivalentes $\Leftrightarrow C$ subclassifica D e D subclassifica C
- $C \& D$ são disjuntos $\Leftrightarrow C \sqcap D \subseteq \perp$

Entretanto uma nova abordagem baseada nos testes de satisfabilidade se encontra cada vez mais utilizada e em numerosos. Será apresentado a transcrição em termos de satisfabilidade (HAARSLEV; MÖLLER, 2003b):

- C é subclassificado por $D \Leftrightarrow C \sqcap D$ é insatisfável
- $C \& D$ são equivalentes $\Leftrightarrow \neg C \sqcap D$ e $C \sqcap \neg D$ são insatisfáveis
- $C \& D$ são disjuntos $\Leftrightarrow C \sqcap D$ é insatisfável

- Inferências de declarações

As inferências de declarações são aquelas destinadas à validação e à criação de novos fatos. Serão apresentados dois métodos de inferência, o primeiro baseado sobre os testes de consistência e o segundo sobre a inclusão de pessoas nas classes de conceitos.

▪ Consistência

Os testes de consistências servem para garantir a validade (a consistência) das declarações definidas na base de conhecimentos em relação à terminologia utilizada. Mais formalmente “Diz-se que uma A_Box é consistente em relação à $T_Box \tau$ se e somente se $\exists \mathcal{J}$ modelo de τ tal que \mathcal{J} é também modelo de A ”.

É possível definir a satisfabilidade em termos de consistência, diz-se que C é satisfável se e somente se $C(a)$ é consistente.

Isso permite evitar situações contraditórias e dispor de uma base de conhecimentos

confiáveis e consistentes (PATEL-SCHNEIDER, 1998). As declarações como “William é um criminoso” e “William não tem antecedentes criminais” serão talvez aceitas se considera-se uma terminologia imediatamente rejeitada em relação à uma terminologia onde será definido um criminoso como sendo um indivíduo tendo antecedente criminal.

- Reconhecimento de instâncias

Este método consiste em efetuar testes a fim de determinar a inclusão de um indivíduo à um conceito definido em uma terminologia; para isto, existem duas abordagens:

- A primeira abordagem se trata de encontrar os indivíduos que são as instâncias pertencentes à um conceito particular. Uma taxonomia bem definida é necessária à este gênero de tratamentos.
- A segunda abordagem, ao contrário, determina, sendo dado um indivíduo, o conjunto de conceitos mínimos ao qual este indivíduo pertence. Em outras palavras pesquisa-se os conceitos mais específicos os quais este indivíduo pertence. Isto se faz em primeiro lugar por um método de abstração sobre o qual define-se a estrutura abstrata do conceito que resultou o indivíduo, em seguida uma classificação clássica do conceito na rede conceitual hierarquizada.

- Os grafos conceituais

Desenvolvidos por JF Sowa (1984) com base nas redes existenciais de Charles Sanders Peirce e nas redes particionadas, eles utilizam um dos princípios da psicologia cognitiva que diz que um conceito é considerado como sendo bem definido se é conhecida sua posição em relação à outros conceitos presentes no domínio modelado (as relações existentes entre ele e todos os conceitos do domínio). Tal abordagem foi muito útil para a modelagem da linguagem natural e é hoje em dia amplamente utilizado em IA para, por exemplo (DIENG-KUNTZ; GANDON, 2005):

- A extração de informações;
- O projeto de bases de dados;
- A operacionalização das ontologias e a concepção de sistemas especialistas e de SBC;
- O processamento das linguagens.

O interesse por esta representação vem do fato que ela combina (ZHONG et al., 2002):

- Flexibilidade de modelagem, permitindo uma certa facilidade de especificação e de interpretação dos conhecimentos pelos usuários humanos.
- Um bom grau de formalismo, atribuindo a possibilidade de definir e de justificar os diferentes métodos de inferências a eles ligados.

- Definições

Utilizando as noções de conceitos e de relações conceituais, os grafos conceituais são grafos bipartidos (dotados de dois tipos de nós: conceito e relação), eles utilizam arcos numerados para conectar os relacionamento nós à seus argumentos (os conceitos nós). Exemplo: “Os cães detestam os gatos.”, ilustrado pela Figura 14.

Figura 14 – Exemplo 1 de grafo conceitual



Fonte: elaborada pela autora

Nota: O grafo acima indica que existe cães os quais existem gatos que eles detestam, o grafo equivalente ao fato que “todos o cães detestam todos os gatos” é ilustrado pela Figura 15:

Figura 15 – Exemplo 2 de grafo conceitual



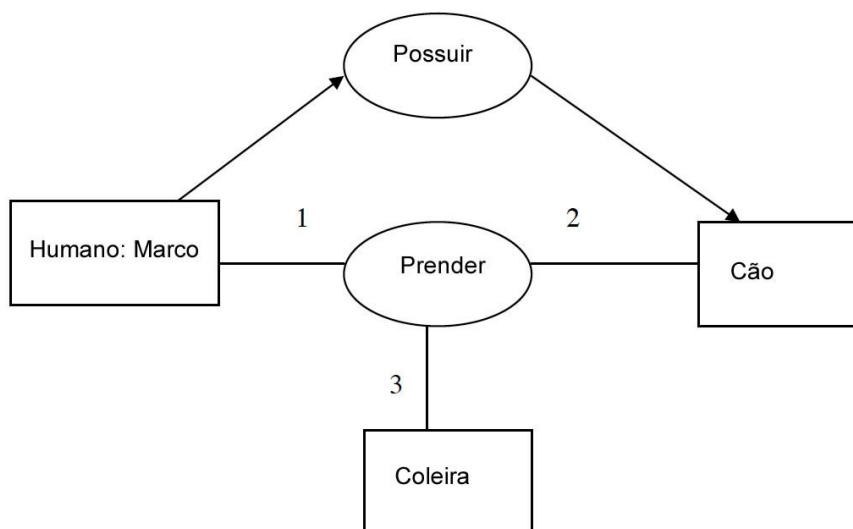
Fonte: elaborada pela autora

Outro exemplo: “Marco prende seu cão na coleira.” (Para as relações binárias o arco de entrada representa o argumento número 1, o de saída representa o número 2), na Figura 16.

A tradução precisa deste grafo é que existe um cão que pertence à Marco e que existe uma

coleira que Marco utiliza para prender este cão.

Figura 16 – Exemplo 3 de grafo conceitual



Fonte: elaborada pela autora

- Os nós conceituais

Como mostra os exemplos anteriores, os conceitos são os nós representativos das entidades descritas pelos grafos conceituais onde eles aparecem, são caracterizados por:

Um tipo de conceito

Um tipo de conceito permite determinar a natureza dos elementos descritos pelo conceito rotulado. Exemplo: “Cão, Gato, Homem”. Os tipos de conceitos podem ser primitivos ou definidos pelos cálculos lambda de valência l implementadas pelos grafos conceituais (onde a utilização será descrita mais tarde).

Uma referência

Permite definir a entidade representada pelo nó conceito, e isto devido aos diferentes campos “opcionais”:

- Um quantificador: Além do quantificador existencial, diferentes quantificadores podem ser definidos para expressar as noções de quantidade (tais como a existência de “ n ” indivíduos) do conjunto, assim como a expressão de semânticas mais

complexas (noção de segmento, de tempo, de modalidade etc.);

- Um designador: serve para identificar um recurso, podendo ser:
 - Um literal: número, cadeia de caracteres, literal codificado (precedido do símbolo %);
 - Uma localização: index, nome, marcador do indivíduo;
 - Indeterminado (nesse caso, pode ser omitido).
- Um descritor: o descritor é um grafo conceitual que serve para descrever o conceito, permite precisar a semântica de um nó conceitual e de descrever sua referência (definem-se os contextos como sendo os nós conceituais tendo um grafo não vazio como descritor).
 - Os nós relação

Os nós relação permitem definir as propriedades que ligam os conceitos do domínio modelado, eles estão em conformidade com um tipo de relação - com o qual os rotulamos. Um tipo de relação permite definir as diferentes propriedades de uma relação conceitual, tais como sua valência (sua aridade), sua assinatura (os tipos dos argumentos que usa e sua ordem) e para os tipos definidos sua semântica devido aos cálculos lambda.

- O cálculo lambda

Notação matemática de funções desenvolvida por Alonzo Church, para a distinção entre igualdade intensional das funções (equivalência de suas fórmulas de definição) das extensionais (igualdade dos conjuntos de pares $(x, f(x))$) (MANZANO, 2004). Exemplo: Sejam 3 grupos de estudantes denotados respectivamente I, II, III. Seja a função f que associa à cada grupo, a média da idade destes estudantes. f será definida pela fórmula $f(x) = (\text{soma das idades do grupo } x) / \text{o número de estudantes do grupo } x$. É possível assim definir esta função para o conjunto de pares $(x, f(x))$ seguindo $(I, 26)$, $(II, 25)$, $(III, 30)$. Seja uma outra função g que associa à cada grupo de estudantes, o número de alunos repetentes: $g(x) = \text{número de aluno de } x - \text{número de alunos não repetentes de } x$. Tal que o conjunto de pares $(x, g(x))$ seja $(I, 26)$, $(II, 25)$, $(III, 30)$.

Observa-se que a fórmula de f , não tem nenhuma relação com a fórmula de g (não existe

nenhuma tradução ou equivalência algébrica ou lógica permitindo passar de uma à outra) embora o seu conjunto de valores seja idêntico. Afirma-se então que f é igual à g de maneira extensional mas diferente intensionalmente.

A definição das fórmulas em cálculo lambda se faz pela separação do nome da função e de seus parâmetros e isto da seguinte forma: $\text{Nome_funcao} \equiv \text{lista_argumentos } \lambda \text{ fórmula}$.

A separação do nome da função destes argumentos permite a manipulação da função sem especificar seus argumentos (que serão definidos durante a substituição do nome, pela sua fórmula equivalente e sua avaliação).

Várias utilizações foram implementadas, tais como:

- A adoção por John McCarthy (1960) do cálculo lambda como notação de base da especificação e avaliação de funções na linguagem LISP.
- A especificação da semântica de frases da linguagem natural pelo cálculo lambda foram estudadas (William Wood - 1968 - definiu a semântica dos quantificadores graças ao cálculo lambda).

Assim, para o presente trabalho o cálculo lambda foi escolhido para a definição da semântica de tipos definidos utilizando tipos primitivos no formalismo dos grafos conceituais, esta escolha permite a especificação de fórmulas permitindo definir se uma entidade pode estar de acordo ou não à um tipo dado, e isto substituindo-a no cálculo lambda que o descreve. Exemplo: “Um domador de leão”

Domador

→(Def)→[Lambda(Humano*x)[?x]→[leao]]

- Expansão dos grafos conceituais

A definição de tipos de conceitos e de relações pelo cálculo lambda permite também modifica-los - os tipos e suas definições - devido os métodos de expansão permitindo a construção de novos grafos (até a obtenção de grafos contendo apenas os tipos de conceitos e relações primitivas). Esta operação é definida como segue (SOWA et al., 2000):

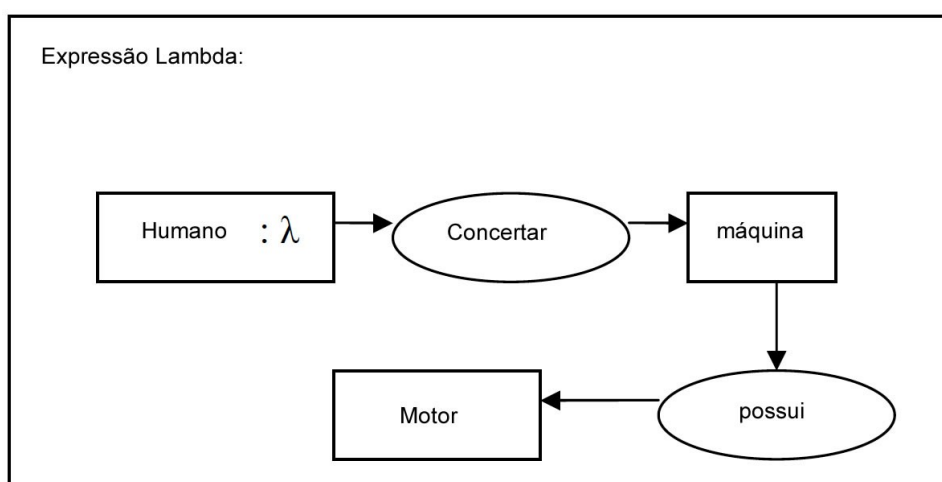
Expansão de nós conceituais

A substituição de um tipo de conceito T (em um nó conceito c) por sua expressão lambda (de valência l , assinatura (s), de parâmetro formal p tendo por tipo s e de fórmula g - define como grafo conceitual contendo p como conceito - se faz em várias etapas):

1. Apagar o tipo de conceito T do nó c e lhe substituir pelo tipo s .
2. Remover p do grafo g e lhe substituir pelo designador de c .
3. Será obtida a expansão pela união do grafo contendo c e do grafo g pelos conceitos c e p (que serão idênticos nesta etapa).

O exemplo da Figura 17 mostra o tipo de conceito “mecânico” definido pela seguinte expressão lambda.

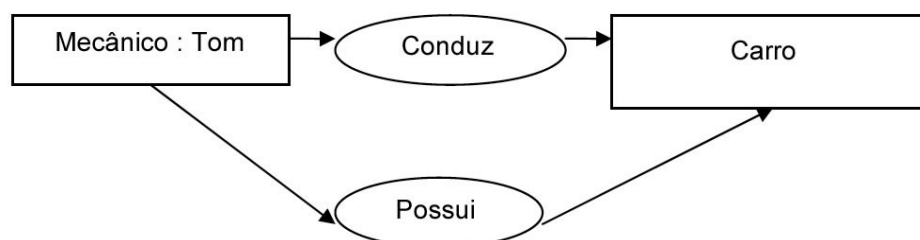
Figura 17 - Conceito expresso por cálculo lambda



Fonte: elaborada pela autora

Seja a substituição do tipo de conceito “mecânico” na declaração “Tom o mecânico conduz seu carro”, ilustrado pelo grafo da Figura 18.

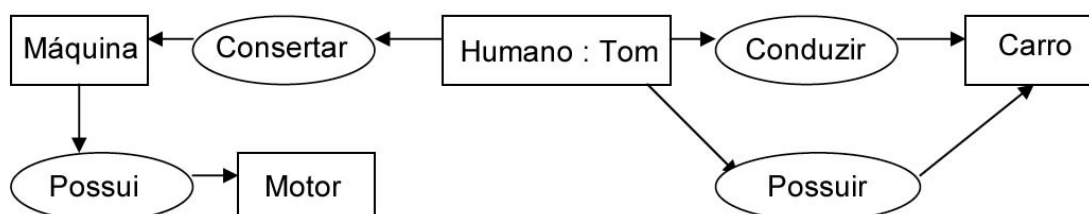
Figura 18 - Exemplo 1 de cálculo lambda



Fonte: elaborada pela autora

O que resulta no grafo ilustrado na Figura 19.

Figura 19 - Exemplo 2 de cálculo lambda



Fonte: elaborada pela autora

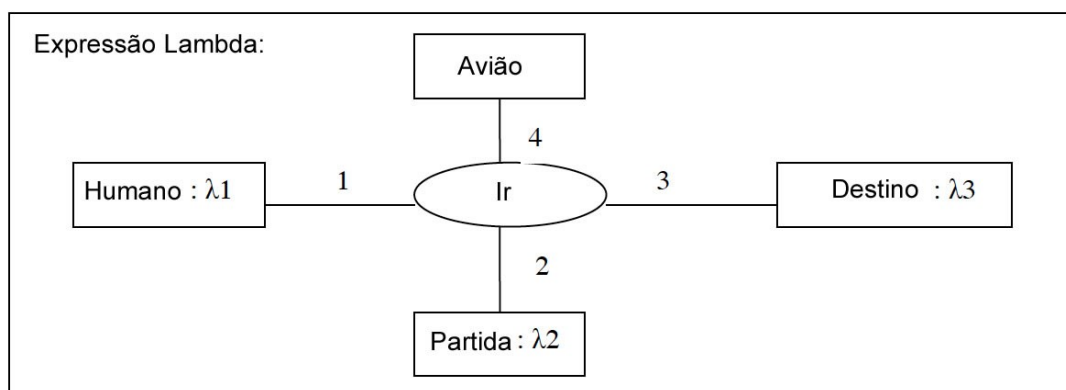
Expansão dos nós relações

A substituição do grafo conceitual de uma relação do tipo R (de valência n) por sua definição em lambda cálculo (de assinatura (t_1, t_2, \dots, t_n) , para os parâmetros formais (p_1, p_2, \dots, p_n)) utilizados como conceitos em seu grafo conceitual g) se faz como segue:

1. Suprimir a relação conceitual do grafo onde ela se encontra, desconectando os arcos e ligando-a aos conceitos que ela tem como argumentos.
2. Para cada conceito " p_i " do grafo g , substituir o parâmetro formal que ele contém pelo designador do argumento ' i ' (o conceito " c_i ") da relação à substituir.
3. Restringir o tipo de conceito de s , p_i e c_i , até seus sub-tipos comuns maximais (para obter os conceitos idênticos) depois fazer a união do grafo da relação com g através destes sub-tipos.

A Figura 20 mostra o exemplo: seja a relação "voar (agente, partida, destino)" - pegar o avião para ir de "partida" em direção "destino" - definido por um cálculo lambda.

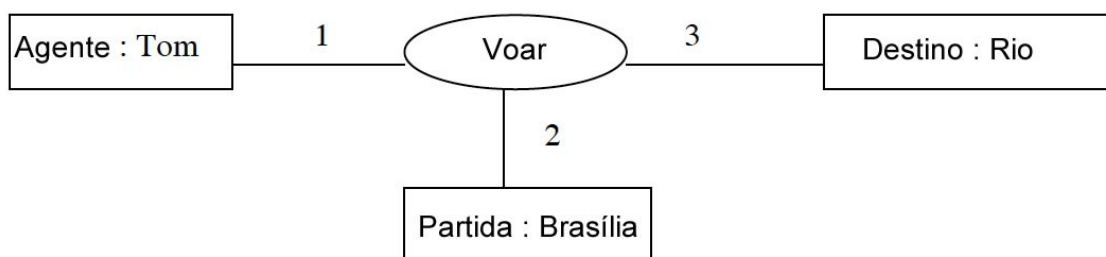
Figura 20 - Exemplo 3 de cálculo lambda



Fonte: elaborada pela autora

Sua substituição no grafo: “Tom voa de Brasília ao Rio.” é mostrado na Figura 21.

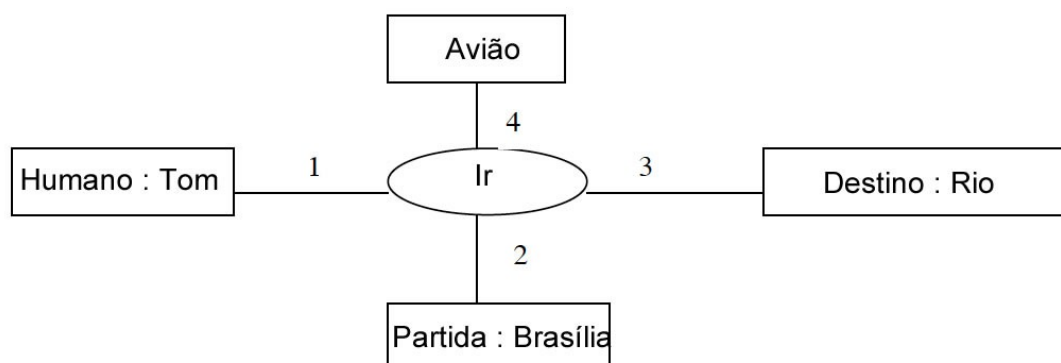
Figura 21 - Exemplo 4 de cálculo lambda



Fonte: elaborada pela autora

O que resulta no grafo da Figura 22.

Figura 22 – Exemplo 5 de cálculo lambda



Fonte: elaborada pela autora

- As bases de conhecimentos em GC

A utilização dos grafos conceituais não se limitam à especificação de conhecimentos únicos, destinados à troca ou à manipulação individual. A especificação da base de conhecimentos é possível utilizando formalismos dos grafos conceituais e isto pela especificação dos seguintes elementos (SMITH et al., 2002):

- Suporte

Uma das originalidades dos grafos conceituais devido a Chein e Mugnier (1992) é a representação separada dos conhecimentos gerais do domínio (sobre os tipos de conceitos e de relações) que serão reagrupados através de um conjunto de grafos chamados suporte e dos conhecimentos personalizados dos usuários (definições de fatos, de restrições de integridade, de regras de inferência, etc.). Conhecimentos estes que serão modelados posteriormente. Tal abordagem aumenta a capacidade de leitura das informações representadas e busca uma certa modularidade na concepção das bases de conhecimentos.

O suporte especifica antes de tudo a semântica das primitivas conceituais (tipos de conceitos e de relações conceituais), e isto utilizando schemas de axiomas e definições em cálculo lambda, isto feito ao nível de três estruturas descritivas (ZHONG et al., 2002):

Hierarquia de tipos de conceitos (Treillis de tipos de conceitos T_c)

É um grafo de tipos de conceitos conectados por arcos de relações de subclassificação (is_a).

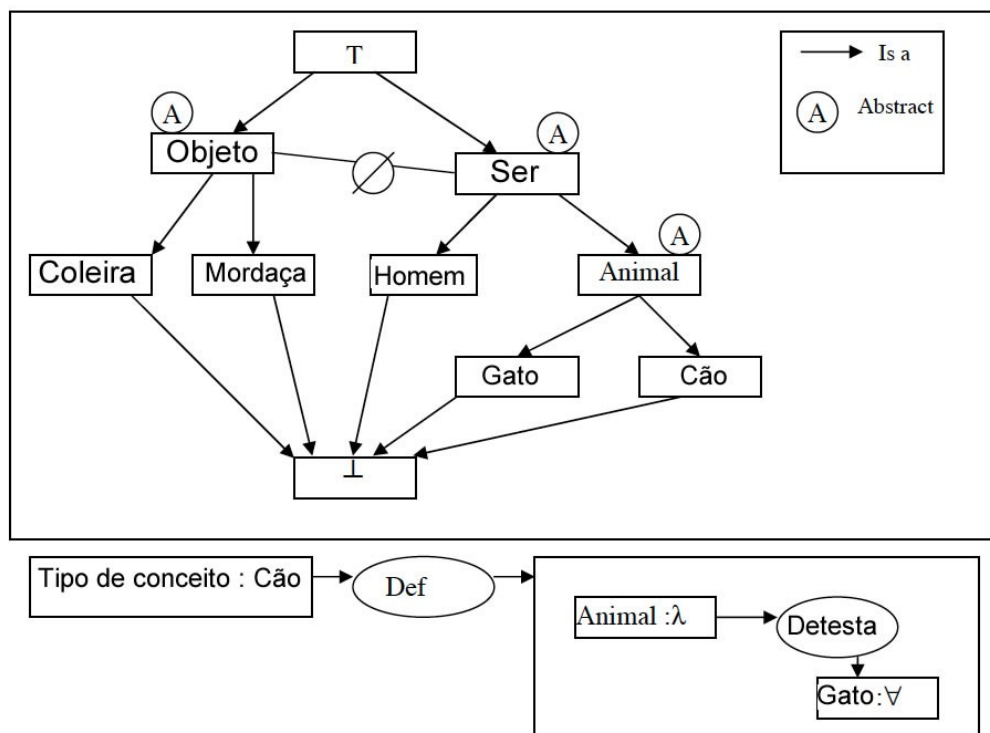
Permite uma organização hierárquica dos diferentes tipos seguindo seus níveis de especificidade, da mesma forma a declaração de suas propriedades devido a sua rotulação com outros schemas de axiomas, assim como definir a semântica dos tipos definidos e associando-os aos cálculos lambda que serão equivalentes aos tipos (CAFÉ; MEDEIROS, 2011). A Figura 23 mostra um exemplo. Define-se o tipo de conceito “Cão como sendo todo animal que detesta todos os gatos”. Notas:

- Todos os tipos de conceitos resultam de um tipo original comumente chamado tipo de conceito T (Entidade) e todos por sua vez o tipo absurdo - *Absurdity* - denotado \perp (devido a forma de treliça).
- Todo schema de axiomas utilizados (à parte a relação Is_a) deve ser descrito sob forma de grafos de declaração definindo sua semântica e sua aplicação (FÜRST; LECLÈRE; TRICHET, 2003).

Hierarquia de relações (*Treillis* de tipos de relações *Tr*)

É um grafo de subclassificação de tipos de relações, define suas propriedades tais como suas aridades ou suas assinaturas (tipagem e ordem de seus argumentos), da mesma forma que o grafo T_c , ele especifica também, graças ao lambda cálculo e diferentes schemas de axiomas (tais como a comutatividade, a simetria, a equivalência de relações etc.) a semântica dos tipos (CAFÉ; MEDEIROS, 2011). A Figura 23 ilustra um exemplo:

Figura 23 - Hierarquia de tipos de conceitos



Fonte: elaborada pela autora

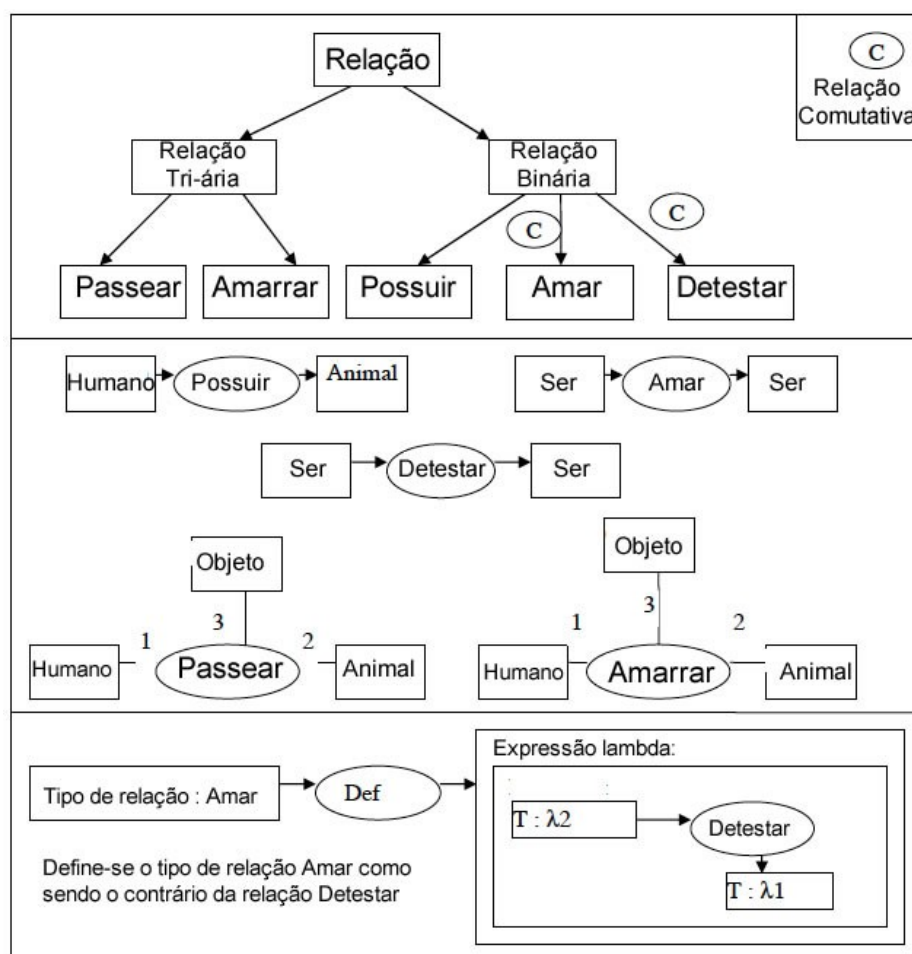
O catálogo de indivíduos - Conjunto de marcadores

Os marcadores são os ponteiros de manipulação das instâncias de conceitos (à parte do marcador * que designa um objeto qualquer). Cada marcador é definido pelas seguintes propriedades:

- Seu nome;
- Os conceitos com os quais está de acordo de maneira explícita, ou a utilização das seguintes regras (no que se segue, m é um marcador e c , c_1 , c_2 são os tipos de conceito):
 - Se m está de acordo à c então ele está de acordo à todo tipo especializado c .
 - Se m está de acordo à c_1 e c_2 , então está de acordo à $c_1 \cap c_2$ (o tipo maior de conceitos especializados c_1 e c_2).
 - O marcador * está de acordo à todos os conceitos, exceto o conceito absurdo (\perp).
- As informações de definição do indivíduo (seus atributos elementares).

Exemplo:

Figura 24 - Hierarquia de relações



Fonte: elaborada pela autora

Quadro 4- Marcadores

Marcadores	Tipo compatível
Marco, a	Humano
X	Ser
X	Ser
X	Ser
X	Ser
b, c, Felix	Gato
D, Rex	Cão
L	Coleira

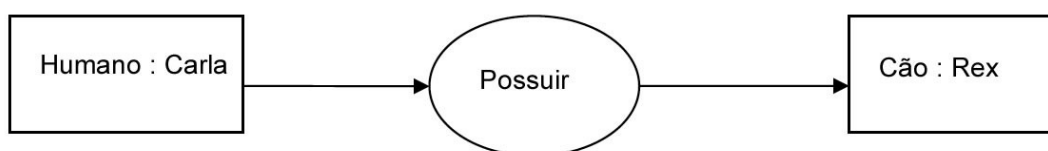
Fonte: elaborada pela autora

- A representação dos fatos

Os conhecimentos assim especificados oferecem mecanismos de validação permitindo especificar os diferentes fatos do domínio modelado por meio dos grafos formados de conceitos (rotulado por um tipo de conceito e um marcador) conectados por instâncias de relações conceituais (rotulados por tipos de relações), estes fatos podem representar:

- Situações declarativas (julgadas corretas ou possíveis) “Carla possui um cão que se chama Rex.”, ilustrada na Figura 25.

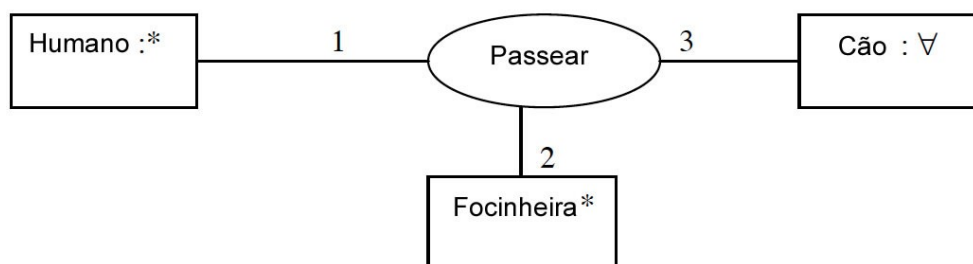
Figura 25 - Representação dos fatos



Fonte: elaborada pela autora

- Restrições de integridade do domínio: “Todo cão pode passear somente com uma pessoa que utilize uma focinheira.”, ilustrada na Figura 26.

Figura 26 - Restrição de integridade do domínio



Fonte: elaborada pela autora

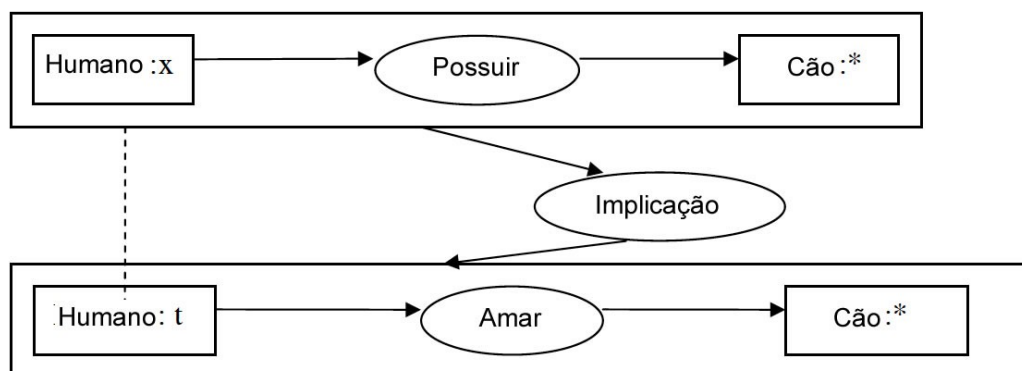
O quantificador universal \forall deve estar explicitamente declarado, inversamente ao quantificador existencial que é utilizado como padrão.

- As regras de inferência: “Se x possui um cão então x ama os cães.”, como mostra a Figura 27.

A linha tracejada indica que estes dois nós fazem referência à mesma entidade (instância equivalente), este link se chama link de correferência. Notas:

- A diferença entre esses três tipos de conhecimentos de declaração somente é palpável no decorrer da operacionalização dos conhecimentos para uma dada utilização (um mesmo grafo pode, por exemplo, representar uma restrição em um sistema e uma regra de inferência em outra) (FÜRST; LECLÈRE; TRICHET, 2003).
- Os links para as co-referências definem os elementos podendo ser substituídos por um deles, que será denominado “elemento dominante”.

Figura 27 - Regra de inferência



Fonte: elaborada pela autora

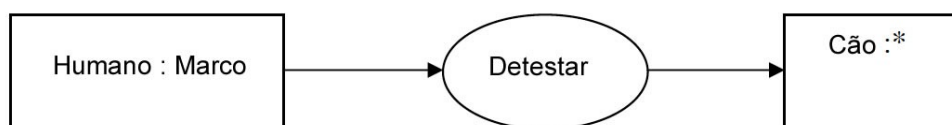
- Os grafos conceituais e a lógica predicativa

Várias funções de tradução dos Grafos Conceituais (GCs) em direção às fórmulas predicativas foram desenvolvidas permitindo assim aproveitar as diferentes ferramentas de raciocínio e de validação desenvolvidas para estas últimas, tais funções permitem, ao mesmo tempo, uma operacionalização mais fácil das ontologias baseadas sobre os GCs pelos motores de inferência predicativos (CARVALHO, 2003). Uma das transformações possíveis pode ser obtida pelo seguinte algoritmo:

1. Associar à cada conceito genérico (Conceito : *) uma variável;
2. Associar à cada elemento de Tc um predicado unário;
3. Associar à cada elemento de Tr à N argumento um predicado n-ário;
4. Associar à cada marcador do suporte uma constante da linguagem de predicados;
5. Para cada grafo G , associa-se a fórmula F , sendo F o fechamento existencial da conjunção da tradução de todos os nós de G .

A Figura 28 ilustra o exemplo: “Marco detesta gatos.” : $\exists (x) (\text{humanos}(\text{Marco}) \wedge \text{gato}(x) \wedge \text{detestar}(\text{Marco}, x))$.

Figura 28 - Grafo conceitual e lógica predicativa

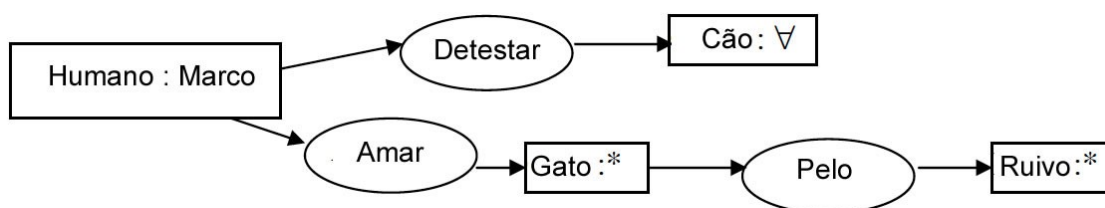


Fonte: elaborada pela autora

▪ Escrita dos GCs

Além da representação gráfica (*Display Form*) e predicativa, os grafos conceituais possuem várias escritas e translações, o que é um meio muito bom de troca de conhecimentos inter-sistemas e uma plataforma de correspondência de formalismos de representações. Exemplo mostrado na Figura 29. Algumas das reescrituras deste exemplo:

Figura 29 - Escrita dos GCs



Fonte: elaborada pela autora

- Forma Linear (LF): variante da representação gráfica, ela é destinada essencialmente à apresentação dos conhecimentos aos usuários humanos.

[Humano : Marco] (Detestar) → [cão]

(Amar) → [gato]-(pelo) → [ruivo]

- *Conceptual Graph Interchange Form* (CGIF): formato especializado na troca de grafos conceituais entre máquinas (norma ISO).

[Cão *x][Gato *y][Ruivo *z](Detestar [humano : Marco] ?x) (Amar [humano : Marco] ?y)
(Pelo ?y ?z)

- *Knowledge Interchange Format* (KIF): linguagem de representação de lógica, permite também a representação de grafos conceituais.

(exists ((?x cao) (?y gato)(?z ruivo)) (and (Humano Marco) (Detestar Marco ?x) (Amar Marco ?y) (Pelo ?y ?z)))

- Raciocínio sobre os GCs

Além da exploração clássica dos conhecimentos induzidos pela tradução dos grafos em fórmulas predicativas (amplamente estudadas e possuindo uma base teórica muito desenvolvida) ou a expansão de expressões lambda permitindo a obtenção de novos conhecimentos, os grafos conceituais permitem a possibilidade de utilizar outros métodos de inferência (pode-se falar de exploração e extração de dados no sentido amplo) vindo dos algoritmos desenvolvidos pela teoria dos grafos ou no contexto das redes semânticas onde uma das aplicações mais elementares é a dedução das relações de especificação entre os diferentes conhecimentos modelados (FÜRST; LECLÈRE; TRICHET, 2003).

A especificação dos conhecimentos dedutivos (por exemplo a especificação dos schemas de axiomas utilizados no suporte ou aquele das propriedades das primitivas conceituais) se faz devido a três tipos de primitivas de raciocínio:

1. as restrições positivas: com um grafo hipótese e um grafo conclusão onde a semântica é “se a hipótese está presente em um grafo, a conclusão deve estar presente no grafo” (senão a restrição não é respeitada e o grafo não é válido no domínio modelado);
2. as restrições negativas: com um grafo hipótese e um grafo conclusão onde a semântica é “se a hipótese está presente em um grafo, a conclusão deve ser ausente no gráfico (senão a restrição não é respeitada e o grafo não é válido no domínio);
3. as regras: com um grafo hipótese e um grafo conclusão onde a semântica é “se a hipótese está presente em um grafo, a conclusão pode ser adicionada ao gráfico”.

A utilização destes três tipos de conhecimentos pelos motores de inferência pode alcançar certo grau de automatização desobrigando o usuário dos processos de operacionalização e de raciocínio.

- A especialização e a generalização

A especialização é uma operação de projeção de grafo utilizada para saber se um conhecimento representado por um grafo é uma variante mais específica que outra representada por um outro grafo (que representa o papel de modelo ou de validador da primeira) (FÜRST; LECLÈRE; TRICHET, 2003). A generalização é a operação inversa. Como mostra o exemplo da Figura 30.

Figura 30 - Especialização e generalização

Os cães detestam os gatos.



Rex detesta Tina..



Fonte: elaborada pela autora

Mais formalmente, diz-se que um grafo H especializa G se e somente se existe duas aplicações f e g , f tendo o conjunto de nós de G em direção aos de H ; e g de conjuntos dos arcos de G em direção H que:

- Associa à cada nó conceito de G (rotulado pelo conceito c e o marcador m), um nó de H $f(c, m)$ denominado (c', m') tal como c' , m' especializado c , m ;
- Associa à cada nó relação de G um nó de relação de H de mesmo tipo (marcado pelo mesmo rótulo);
- Associa à cada arco de $G(a)$ conectando um nó de relação (R) e um nó conceito (c, m) um arco de H $g(a)$ que religa $f(R)$ à $f(c, m)$.

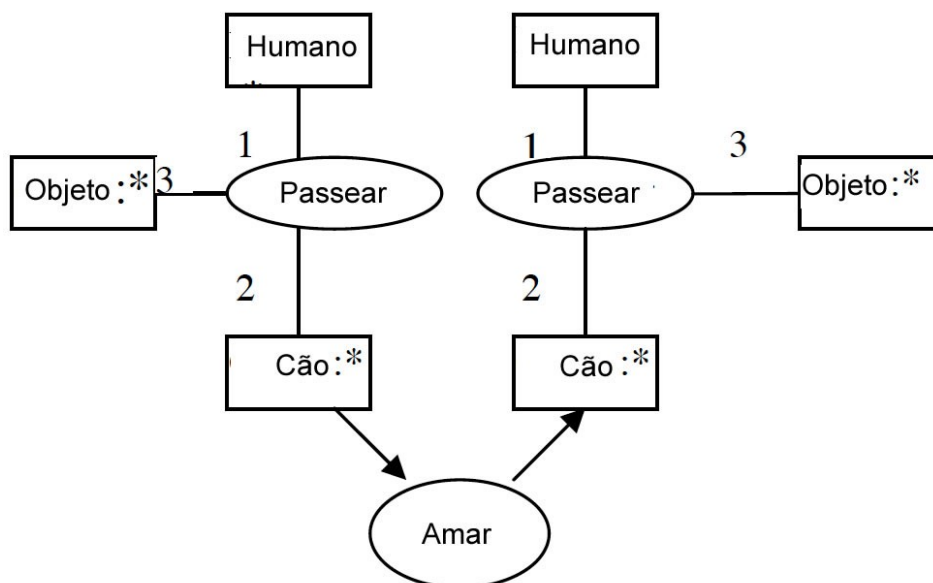
Sendo a generalização a relação dual da especificação, é suficiente encontrar as aplicações inversas de f e g . Exemplo: “Dois cães que se amam passeiam com duas pessoas.”, ilustrado na Figura 31.

Especificação 1: “Marco e Sofia passeiam com seus cães.”, como mostra a Figura 32.

Especificação 2: “Sofia passeia com Rex e Lesse que se amam.”, como mostra a Figura 33.

Percebe-se que pela adição da regra de dedução “ $x (c(x) c'(x))$ ” para cada par c , c' do conceito do domínio tal como c especializa c' aos axiomas de dedução da lógica predicativa é possível demonstrar que:

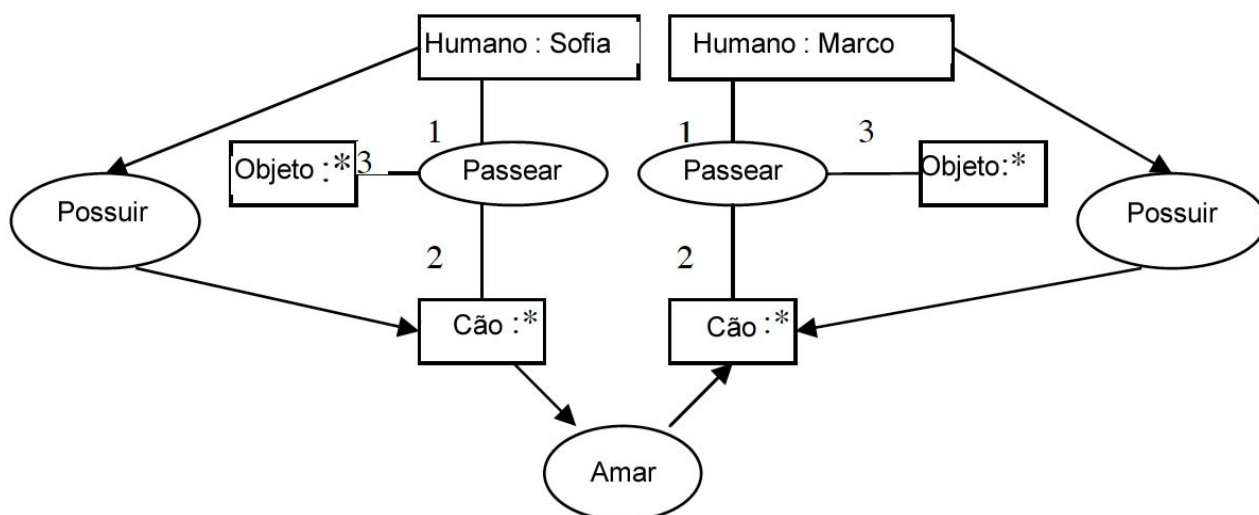
Figura 31 – Exemplo de generalização



Fonte: elaborada pela autora

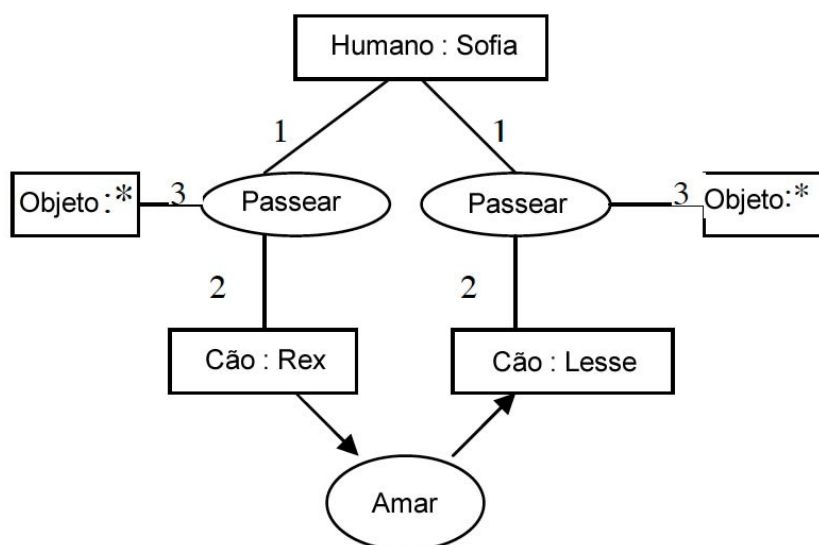
(H especializa G) \longleftrightarrow (A fórmula lógica de H \rightarrow a fórmula lógica de G)

Figura 32 – Exemplo 1 de especialização



Fonte: elaborada pela autora

Figura 33 – Exemplo 2 de especialização



Fonte: elaborada pela autora

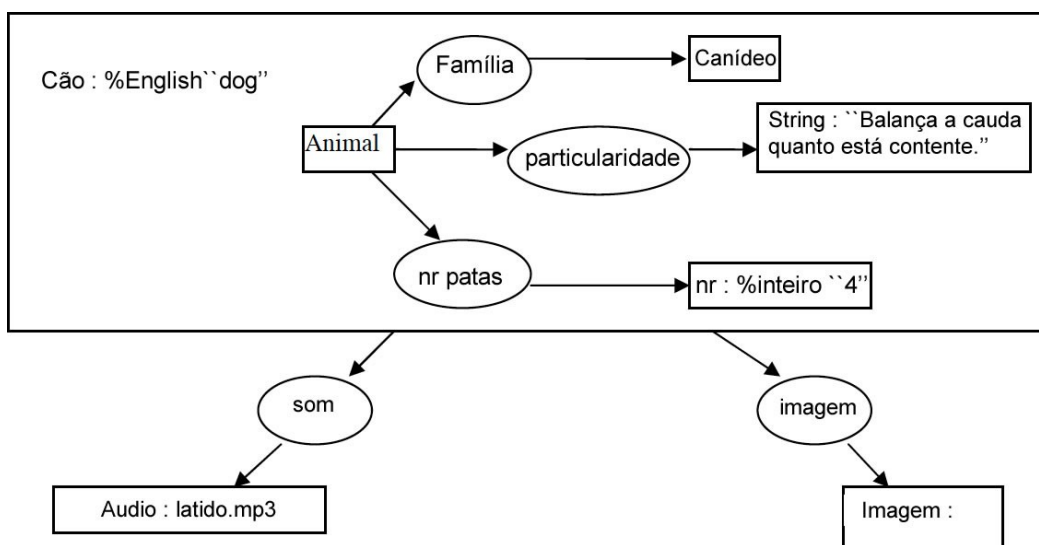
- Expressividade dos grafos conceituais

Os grafos conceituais permitem pela adição de notações especiais para os conjuntos e os nomes, métodos de referência às entidades do mundo (físicas ou lógicas) assim como mecanismos de aninhamento de grafos para fazer crescer sua expressividade, permitindo assim (CARVALHO, 2003):

- Definir uma semântica mais forte de conceitos do domínio e criar conceitos personalizados

A Figura 34 ilustra o seguinte exemplo: Os cães, chamados em inglês *dog* definidos como sendo os animais da família dos canídeos, possuindo 4 patas e como particularidade balançam a cauda quando estão contentes (também é especificada sua aparência e o som que eles fazem pelo fato de apontar dois arquivos, um de imagem e outro de som).

Figura 34 - Exemplo da expressividade dos grafos conceituais

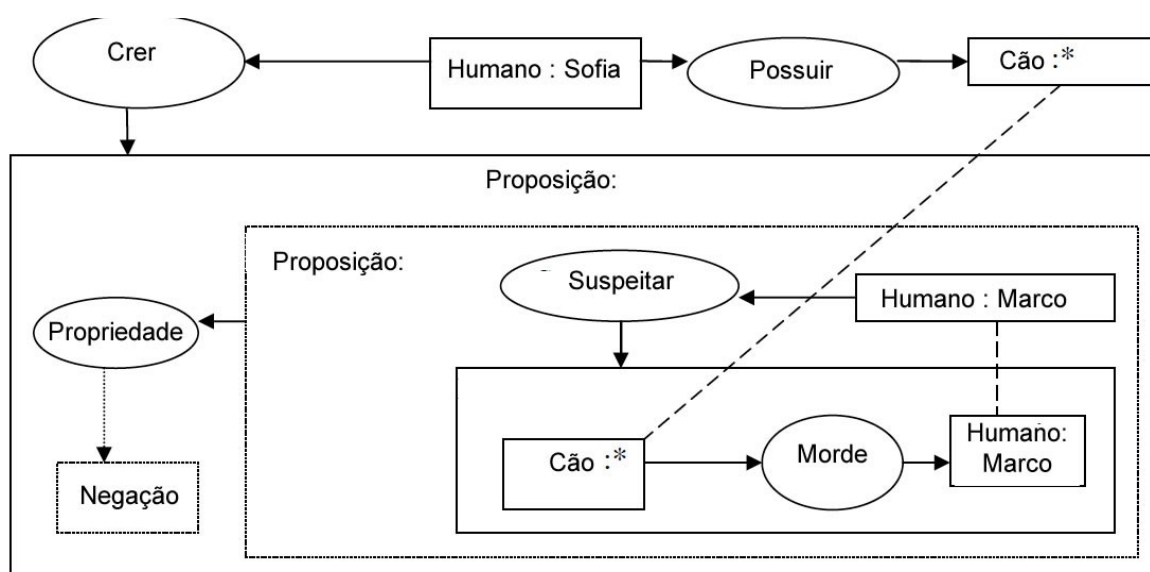


Fonte: elaborada pela autora

- Representar os conhecimentos modais

A Figura 35 ilustra o exemplo: “Sofia não acredita que Marco suspeita que seu Cão (o cão de Sofia) o mordeu.”

Figura 35 - Representação de conhecimento modal



Fonte: elaborada pela autora

2.4.1.3 Critérios de avaliação da escolha da representação

Não existem meios concretos para afirmar que um domínio é melhor representado utilizando um formalismo ou outro, porque algumas características são mais levadas em consideração por algumas representações que outras, tais como (RAMOS; AUGUSTO; SHAPIRO, 2008):

- A eficácia dos métodos de tratamento e sua aptidão para produzir inferências mais extensivas e robustas e oferecer mecanismos de prova e explicação de raciocínios;
- A expressividade da linguagem de representação permite a modelagem de mais conhecimentos mas isto ao detrimento do primeiro critério;
- A facilidade da interpretação e da formulação dos conhecimentos pelos usuários que devido a uma melhor visualização dos conceitos e relações poderão ser melhor utilizados.

2.4.2 *Discussão*

É evidente que a escolha do modelo de representação adequado é primordial a fim de oferecer uma melhor oferta frente aos objetivos que são fixados pela plataforma da Web Semântica, ou seja, uma melhor representação dos dados dos usuários que apresentará as vantagens do ponto de vista da modelagem mas também em relação ao poder e a precisão das inferências que lhes são aplicadas e sua viabilidade.

Como foi visto, a evolução em direção a Web Semântica repousa em grande parte sobre o aspecto dos dados que se propõe manipular. A passagem desses dados à escala das ontologias é um dos eixos mais promissores. Assim, a criação e a modernização das linguagens de representação das ontologias estão no centro de muitas pesquisas atuais.

Se trata principalmente de propor linguagens que se associam ao mesmo tempo a modularidade e a intuitividade prometidas pelas ontologias, mas também as ontologias resultantes oferecem esta facilidade de inferência e de dedução necessária para o trabalho dos “reasoners”.

Propor uma linguagem de concepção de ontologias, é primeiramente propor o formalismo sobre o qual ela será baseada, o que se mostra um fator decisivo. Fazer uma boa escolha neste nível pode melhorar muito a qualidade das ontologias, e pode também atribuir oportunidades de diversificação dos métodos de raciocínio que lhes são aplicáveis.

2.5 Técnicas de Inferência

Na primeira Seção, foi introduzido o fato de que a incorporação de uma camada semântica na Web implicaria um trabalho sobre dois níveis. O primeiro trata da evolução dos métodos e dos formalismos de representação dos conhecimentos, o que foi abordado na seção precedente.

O segundo ponto consiste no estudo e desenvolvimento de métodos de inferência sobre estes conhecimentos, sendo que a normalização da semântica das estruturas de representação dos conhecimentos depende em grande parte dos tratamentos que se deseja atribuir a fim de oferecer a possibilidade de efetuar manipulações inteligentes dos dados que vão além de simples mecanismos de armazenamento e de extração diretos adotados até então na computação clássica em geral.

Assim, é necessário durante o processo de desenvolvimento de um sistema à base de conhecimentos, efetuar um estudo paralelo sobre estes dois planos (HAARSLEV; MÖLLER, 2003b). De fato, a escolha de um formalismo de representação adequado para uma dada aplicação deve levar em conta os tipos de inferência que este formalismo atribuirá sobre os conhecimentos que ele representa e isto em razão do impacto que pode ter sobre a relevância mas também o desempenho do sistema implementado.

Na sequência, será proposto o estudo de três técnicas de inferência. Será visto primeiramente a abordagem clássica baseada na dedução, depois a abordagem da abdução e para finalizar, a inferência por analogia e o tratamento baseado em casos que se revela cada vez mais aplicáveis nos domínios da IA.

Serão abordadas, então, apresentando suas respectivas filosofias, assim como as premissas necessárias para sua operacionalização no formalismo de representação de conhecimentos.

2.5.1 *Raciocínio dedutivo*

A dedução é, sem dúvida, a técnica de raciocínio mais comum atualmente, o estudo de seus princípios remonta a Aristóteles (384-322 a.c.) e a implementação das noções básicas sobre a sintetização do raciocínio lógico.

O processo de dedução pode ser definido como sendo um encadeamento de proposições de acordo com regras definidas, constituídas pelos axiomas e regras de inferência e isto com o objetivo de atingir um dado resultado (HEINZLE, 2012).

O exemplo mais conhecido do raciocínio dedutivo introduz o modus ponens de Aristóteles: “Todo humano é mortal, Sócrates é humano então Sócrates é mortal.”

O estudo que o presente trabalho propõe se baseará sobre o estudo das premissas necessárias ao desenvolvimento deste raciocínio para um dado formalismo de representação.

Estas premissas definem os conhecimentos manipulados durante o processo de inferência (hipóteses de partida, proposições inferidas, axiomas do domínio e conclusões da inferência) e as regras de manipulação (regras de dedução) que constituem o processo de raciocínio.

Em seguida, a apresentação dos diferentes métodos de elaboração do raciocínio dedutivo assim como uma visão operacional de implementação de um motor de inferência dedutivo prático.

2.5.1.1 Premissas do raciocínio dedutivo

2.5.1.1.1 Premissas sobre os conhecimentos manipulados

Os conhecimentos manipulados pelo raciocínio dedutivo devem partir de domínios de aplicações estáveis - no tempo e espaço - nos quais as hipóteses - premissas do raciocínio - sejam definidas como certas. Todo conhecimento deduzido a partir de conhecimentos verdadeiros será também considerado como verdadeiro.

A conclusão do raciocínio, definido como sendo o objetivo a alcançar, representa um conjunto de proposições deduzidas das hipóteses, este conjunto pode até mesmo ser o conjunto vazio - será falado sobre este caso de inconsistência nas hipóteses (STENNING; COX; OBERLANDER, 1995).

2.5.1.1.2 Axiomas do domínio

Além dos axiomas relacionados à lógica clássica, a utilização de axiomas relativos aos domínios da aplicação do raciocínio poderia facilitar o processo de inferência, cita-se por exemplo aqueles da teoria dos números de Peano (tal como $0+x=x$), aqueles da geometria euclidiana, os da teoria dos conjuntos etc.

A dedução módulo é um exemplo desta incorporação, combinando a regra de dedução e a regra de cálculo, ela permite diminuir a lentidão da inferência devido a formalização intensiva dos conhecimentos e isto substituindo diretamente as fórmulas, podendo ser calculadas por seus “verdadeiros” valores, tal como por “ $4+3=6$ ” que será substituído pelo valor “ $7=6$ ” ou pelo valor \perp (contradição ou o falso absoluto), por exemplo.

2.5.1.1.3 As regras de dedução

A combinação de conhecimentos e a inferência de novos dados se faz com a utilização de regras de manipulação definidas de acordo com as bases lógicas.

Seguem alguns exemplos (onde P e Q são proposições) (BUVAE; MASON, 1993):

Modus ponens

Se P então Q; P : então Q

“Se o crime aconteceu no quarto, ele foi cometido de noite; o crime foi cometido no quarto: então foi cometido de noite”

Modus tonens:

Se P então Q; não-Q : então não-P

“Se o jardineiro fosse culpado, ele teria fugido; o jardineiro não fugiu : então o jardineiro não é culpado”

Modus tollendo ponens (ou silogismo disjuntivo):

P ou Q ; não P : então Q

“Ou o secretário é culpado, ou o guarda é culpado; o guarda não é culpado : então o secretário é culpado”

Eliminação:

P e Q : então P

“O cavaliariço é inocente e a guarda é suspeita : então o cavaliariço é inocente”

Introdução:

P : então P ou Q

“A governanta estava dormindo : então ou a governanta estava dormindo, ou ela estava ausente”

Adição:

P ; Q ; $P \& Q$ (P ; Q : então P e Q)

“O cavaliariço é inocente, a guardiã é inocente : então o cavaliariço e a guardiã são inocentes”

Dupla negação:

(não-(não- P)) : então P

“O cavaliariço não estava inconsciente : então o cavaliariço estava consciente”

Contraposição:

Se P então Q : então se não- Q então não- P

“Se o crime aconteceu no quarto, ele foi cometido de noite : então se o crime não foi cometido de noite, ele não aconteceu no quarto.”

Outras regras relativas aos quantificadores (tais como o universal “ \forall ” e o existencial “ \exists ” definidos na lógica dos predicados) podem ser utilizadas se o formalismo de representação

das hipóteses incorpora a noção de variável. Por exemplo a regra de especialização “ $\forall x P(x)$ ” então $P(a)$ onde x é uma variável e a é uma constante do domínio.

2.5.1.2 Sistemas dedutivos

Os sistemas dedutivos são as visões teóricas dos motores de inferência. Eles descrevem as diferentes regras envolvidas durante o processo de dedução, e também a forma de aplica-las (MARTINS, 2004).

Vários sistemas dedutivos foram elaborados, de graus de formalismo e de possibilidade de automatização variados, evoluindo e oferecendo uma base teórica cada vez mais complexa - e eficaz - aos motores de inferência atuais. No que segue, serão apresentados os mais pertinentes:

2.5.1.2.1 O sistema dedutivo de Hilbert

Conhecido como sendo o primeiro sistema dedutivo estudado, ele possui somente uma regra de dedução que é o modus ponens (RAGGIO et al., 2008). Contudo, ele define um grande número de axiomas lógicos, como os seguintes exemplos:

1. $A \Rightarrow B \Rightarrow A$
2. $(A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C))$
3. $\perp \Rightarrow A$
4. $A \Rightarrow (\neg A \Rightarrow \perp)$
5. $(A \Rightarrow \perp) \Rightarrow \neg A$
6. $(A \wedge B) \Rightarrow A$
7. $(A \wedge B) \Rightarrow B$
8. $A \Rightarrow (B \Rightarrow (A \wedge B))$
9. $A \Rightarrow (A \vee B)$
10. $B \Rightarrow (A \vee B)$
11. $(A \vee B) \Rightarrow ((A \Rightarrow C) \Rightarrow ((B \Rightarrow C) \Rightarrow C))$

Outros axiomas relativos ao domínio podem ser definidos (como $x+0=x$ na teoria dos números de Peano etc.)

A dedução se resume então em uma sucessão de aplicações de modus ponens, sobre o conjunto de proposições, inicializada com as hipóteses iniciais do raciocínio e incrementada pelas proposições deduzidas até a obtenção do objetivo (ou chegar à uma contradição). Exemplo: Tentar deduzir a proposição $A \Rightarrow A$. Primeiramente, percebe-se que esta proposição não se encontra entre os axiomas, então deve ser demonstrada para ser considerada verdadeira. Uma das demonstrações possíveis pode ser:

1. $A \Rightarrow A \Rightarrow A$ é um axioma lógico (1.1 em que b é substituído por A)
2. $A \Rightarrow (A \Rightarrow A) \Rightarrow A$ é também um axioma lógico (1.1 em que B é substituído por $A \Rightarrow A$)
3. $(A \Rightarrow ((A \Rightarrow A) \Rightarrow A) \Rightarrow ((A \Rightarrow (A \Rightarrow A)) \Rightarrow (A \Rightarrow A)))$ é também um axioma lógico (1.2 em que B e C são substituídos por A)
4. É possível aplicar o modus ponens às proposições 2 e 3 obtendo a seguinte proposição: $((A \Rightarrow (A \Rightarrow A)) \Rightarrow (A \Rightarrow A))$
5. É possível ainda aplicar o modus ponens às proposições 4 e 1 para obter $A \Rightarrow A$.

Isto anuncia bem os limites da automatização da dedução de Hilbert, sendo a escolha o encadeamento dos axiomas a utilizar parece ser de grande complexidade para uma manipulação automatizada e recorre muitas vezes à “intuição” humana para a escolha dos axiomas propícios para a elaboração da dedução.

2.5.1.2.2 A dedução natural - versão antiga

A dedução natural veio para corrigir as deficiências da dedução de Hilbert, isto se faz essencialmente através da substituição dos axiomas lógicos - apresentados anteriormente - através das regras de dedução equivalentes, permitindo assim eliminar a complexidade introduzida pela combinação dos axiomas - frequentemente feita de maneira *ad hoc* (MORTARI, 2001). A dedução natural define dois tipos de regras, os que introduzem os operadores e os que os retiram, como ilustra a Figura 36.

Figura 36 - Dedução natural

$\frac{[A] \quad \vdots \quad B}{A \Rightarrow B} \Rightarrow\text{-intro}$ $\frac{A \quad B}{A \wedge B} \wedge\text{-intro}$	$\frac{A \quad A \Rightarrow B}{B} \Rightarrow\text{-elim}$ $\frac{A \wedge B}{A} \wedge\text{-elim}$ $\frac{A \wedge B}{B} \wedge\text{-elim}$
--	---

Fonte: elaborada pela autora

A regra de introdução da implicação “ \Rightarrow ” implica a noção de “rejeição/descarte” da hipótese, isto se faz pela utilização de colchetes “[]” para marcar as hipóteses que não são mais necessárias no processo da demonstração. Assim, a demonstração da proposição $A \Rightarrow A$ fica bastante simplificada, se faz como segue.

$$\frac{[A]}{A \Rightarrow A}$$

2.5.1.2.3 A dedução natural - versão moderna

A dedução natural traz uma melhora significativa e uma certa “elegância” no processo de demonstração. Essa clareza foi ainda melhorada pela idéia de reagrupar as proposições demonstráveis e suas hipóteses - idéia introduzida nos anos 30 - de modo a tornar as demonstrações mais explícitas e organizadas (MORTARI, 2001).

A notação adotada será então a seguinte: $\Gamma \vdash P$. Onde Γ é um conjunto de hipóteses e P é a proposição a deduzir. A apresentação das regras de dedução será feita agora como ilustra a Figura 37.

Figura 37 - Regras de dedução 1

$\frac{}{\Gamma \vdash A}$ axioma, se $A \in \Gamma$	
$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B} \Rightarrow$ -intro	$\frac{\Gamma \vdash A \quad \Gamma \vdash A \Rightarrow B}{\Gamma \vdash B} \Rightarrow$ -elim
$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \wedge$ -intro	$\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \wedge$ -elim $\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B} \wedge$ -elim
$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \vee$ -intro $\frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \vee$ -intro	$\frac{\Gamma \vdash A \vee B \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C} \vee$ -elim
$\frac{\Gamma, A \vdash \perp}{\Gamma \vdash \neg A} \neg$ -intro	$\frac{\Gamma \vdash A \quad \Gamma \vdash \neg A}{\Gamma \vdash C} \neg$ -elim, C qualquer
$\frac{\Gamma \vdash \perp}{\Gamma \vdash C} \perp$ -elim	
$\frac{\Gamma \vdash \langle c/x \rangle P}{\Gamma \vdash \forall x P} \forall$ -intro*	$\frac{\Gamma \vdash \forall x P}{\Gamma \vdash \langle t/x \rangle P} \forall$ -elim
$\frac{\Gamma \vdash \langle t/x \rangle P}{\Gamma \vdash \exists x P} \exists$ -intro	$\frac{\Gamma \vdash \exists x P \quad \Gamma, \langle c/x \rangle P \vdash A}{\Gamma \vdash A} \exists$ -elim*

(*). Na condição de que a constante c não apareça nem em Γ nem em A . Essa condição é necessária senão haveria uma prova para: $\vdash \exists x P(x) \Rightarrow \forall x P(x)$.

Fonte: elaborada pela autora

Isso permite tornar as demonstrações mais explícitas. Uma demonstração de P sob as hipóteses G será então uma árvore onde a raiz é $\Gamma \vdash P$ e as folhas são as regras do axioma ou terceiro excluído ($A \vee (\neg A)$), porque são as únicas que não requerem hipótese, e construídas de acordo com as regras de dedução natural.

Nesse sistema, a demonstração da proposição $A \Rightarrow A$ se faz da seguinte maneira:

$$\frac{A \vdash A \text{ axioma}}{\vdash A \Rightarrow A}$$

Nota: Foi demonstrado que os sistemas de dedução natural e os de Hilbert são equivalentes, isto quer dizer que: uma proposição P é demonstrável à partir das hipóteses G em dedução à la Hilbert se e somente se ela também o é em dedução natural (HAARSLEV; MÖLLER, 2003a).

2.5.1.3 Motores de inferência dedutivos

Os sistemas dedutivos se confrontam com diversos problemas de ordem técnica durante sua operacionalização (tais como os tempos de cálculo proibitivos, a indisponibilidade dos recursos de sistemas, etc). Isto deixa sua implementação um pouco diferente dos modelos teóricos sobre os quais se baseiam, entretanto, são construídos de acordo com o seguinte algoritmo (ZOU; FININ; CHEN, 2005):

1. $DB \leftarrow P$.
2. Selecionar uma regra de inferência $p_1, p_2, \dots, p_k \Rightarrow q$.
3. Encontrar as proposições $p_1, \dots, p_k \in DB$. Se elas são encontradas: se $q \notin \{BD\}$ adicionar q à $\{BD\}$; se q corresponde à uma solução, imprimir q (e parar).
4. Repetir a partir de 2.

Onde $\{BD\}$ é o conjunto de proposições já inferidas - inicializadas no início devido as hipóteses de partida $\{P\}$.

A diferença entre os motores de inferências se limita geralmente às escolhas das regras aplicadas e da estratégia de suas aplicações. Na prática limita-se geralmente os conhecimentos manipulados à um conjunto de fatos - geralmente descritos devido as proposições simples - sobre as quais atribui-se as manipulações através de um conjunto de regras da seguinte forma: $Cond_1 \wedge Cond_2 \wedge \dots \Rightarrow Consequencia$.

Um procedimento de resolução pode ser descrito como sendo uma aplicação em cadeia de conjuntos de proposições, ligadas por regras de inferência, tendo em um de seus objetivos as hipóteses de partida e em outra a solução à encontrar. Como mostra a Figura 38.

Existe então duas maneiras de estabelecer esta “cadeia dedutiva”:

Figura 38 – Procedimento de resolução

$$F_1 \xrightarrow{R_1} F_2 \xrightarrow{R_2} \dots \xrightarrow{R_n} \text{Solução}$$

Fonte: elaborada pela autora

- **Por encadeamento “para frente”**: à partir de todos os fatos conhecidos, produzir todas as consequências até que uma solução seja encontrada entre as inferências, como ilustra a Figura 39.

Figura 39 - Regras de dedução 3



Fonte: elaborada pela autora

- **Por encadeamento “para trás”**: mais sutil mas também mais difícil de implementar, consiste em produzir as etapas intermediárias hipotéticas que permitirão deduzir a solução de uma descrição da solução pesquisada (PY, 2009).

2.5.1.4 Discussão

É necessário salientar que o poder desta abordagem vem sobretudo do fato que ela tem uma plataforma teórica muito desenvolvida, oferecendo a base necessária ao desenvolvimento de ferramentas tais como mecanismos de validação e de explicação de raciocínios, de testes de decidibilidade assim que outros ligados às melhoras definidas sobre alguns tipos de conhecimentos manipulados (modais, temporais etc.).

Mas a dificuldade de elaboração de sistemas dedutivos frente aos conhecimentos mal definidos (dificuldade de um ponto de vista técnico e financeiro) e a limitação dos conhecimentos que esses sistemas podem causar mostra bem os limites da dedução e encoraja o estudo de novas formas de inferência, respondendo às necessidades cada vez mais presentes na prática.

2.5.2 *Raciocínio por abdução*

A abdução é um processo de raciocínio que foi introduzido na IA a partir dos anos 70 (FLACH; KAKAS, 2000) e que parece vir oposto ao raciocínio dedutivo. Se por um lado a dedução se baseia sobre as premissas bem definidas ou axiomas para dar algumas conclusões, por outro lado a abdução é um método que parte de um conjunto de axiomas definidos para dar um conjunto de conclusões de natureza provável e até mesmo incerta.

Mais formalmente, considera-se a título de exemplo a regra do Modus Ponens, pode-se dizer que a partir dos axiomas A e $A \rightarrow B$, é possível deduzir B .

No raciocínio abduativo, se dispõe de axiomas B e $A \rightarrow B$ então pode-se dizer que A é potencialmente verdadeiro, ou que é possível que o fato que A é verdadeiro seja a causa de que $A \rightarrow B$.

Portanto, a abdução é de algum modo uma dedução posterior, sob certas condições, ela pode ser definida como sendo “a inferência para a melhor explicação” (HARTLEY; COOMBS, 1990a).

2.5.2.1 *Abordagens e tipos de abdução*

Como cita Josephson (CLERGERIE; RAJMAN, 2003), as perspectivas da abdução podem ser divididas de acordo com a maneira onde se considera as explicações inferidas:

A primeira abordagem - que é a mais frequente - consiste em considerar a abdução como uma ferramenta de demonstração servindo para fornecer provas para as observações presentes, a explicação consistirá no conjunto das suposições/premissas utilizadas na

construção da prova. A escolha das regras se baseia sobre uma semântica extensional, quer dizer, que as hipóteses devem implicar extensionalmente as observações anotadas, a implementação mais evidente desta abordagem é naturalmente a utilização dos mecanismos dedutivos comuns à demonstração de teoremas.

A segunda abordagem tenta por sua vez pesquisar e estabelecer uma relação intencional entre a explicação e as observações que lhe são associadas, e diz-se que uma observação é explicada se os conhecimentos disponíveis cobrem os dados observados.

2.5.2.1.1 A abdução como ferramenta de demonstração

Como foi citado anteriormente, esta abordagem trata das explicações como sendo os conjuntos de hipóteses que implicam de forma extensional as observações, ela utiliza para isto os mecanismos e algoritmos clássicos utilizados na demonstração de teoremas.

Serão expostas três abordagens, por cobertura, os sistemas de manutenção da verdade e a abordagem probabilística, o presente trabalho se interessa mais particularmente à primeira que será repetido depois, no contexto da aplicação deste raciocínio aos grafos conceituais.

a) Abordagem da abdução por cobertura

O fundamento desta abordagem consiste em estabelecer um conjunto de fatos considerados como causas hipotéticas, de tal maneira que o conjunto de consequências destes fatos cobre perfeitamente o conjunto de observações que se deseja explicar (FLACH; KAKAS, 2000).

De um ponto de vista prático, fala-se mais frequentemente de uma cobertura parcimoniosa de tais conjuntos; a parcimônia é um critério comumente associado ao termo “minimização”. Uma abordagem abdutiva baseando-se sobre a cobertura parcimoniosa deve principalmente ter em conta duas funções:

1. A cobertura : onde o papel é de reagrupar todas as explicações possíveis aos dados observados.

2. A parcimônia : onde o papel é de minimizar a complexidade da explicação obtida (de acordo com os critérios tais como a cardinalidade do conjunto de causas, ou a não redundância nas hipóteses obtidas).

Isto está perfeitamente alinhado com a abdução onde a principal função é de construir explicações potenciais depois de selecionar entre elas a melhor explicação possível.

Para melhor explicar a noção de cobertura, propõe-se estudar a abordagem proposta por Peng e Regia (PENG; REGGIA, 1986) para a resolução de problemas de diagnóstico. Um problema de diagnóstico consiste em encontrar uma explicação para a existência de um conjunto de manifestações (observações, sintomas etc.) utilizando o conhecimento disponível; compreende dois elementos principais: “as manifestações e os transtornos”.

As manifestações são os dados observados em um dado domínio, e onde busca encontrar explicações potenciais, já os transtornos, são assimilados aos diferentes elementos podendo induzir estas manifestações. O conjunto das manifestações e dos transtornos podem ser representados sob a forma de uma rede de causalidades, mas formalmente tem-se as seguintes definições (PENG; REGGIA, 1986):

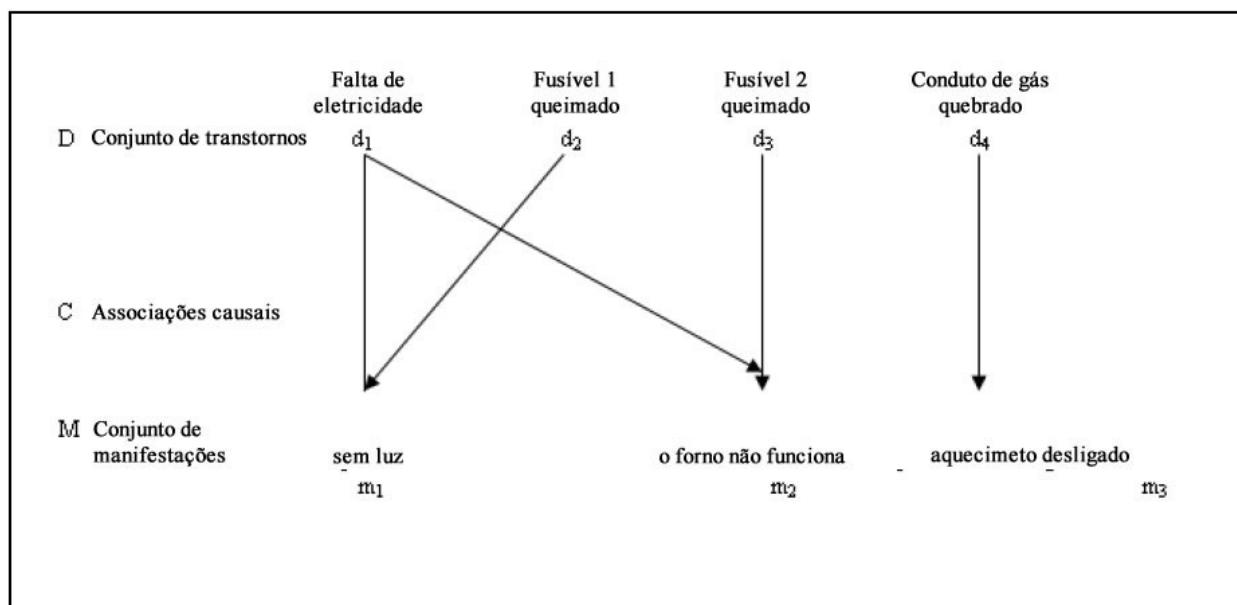
Definição 1:

Um problema de diagnóstico P é uma 4-tupla $\langle D, M, C, M^+ \rangle$ onde:

- $D = d_1, \dots, d_n$ é um conjunto finito de objetos, não vazio, chamado desordens.
- $M = m_1, \dots, m_k$ é um conjunto finito não vazio de objetos, chamado manifestações.
- $C \subseteq D \times M$ é uma relação tal que $\text{domínio}(C) = D$ e $\text{rang}(C) = M$, chamada causalidade.
- $M^+ \subseteq M$ é um conjunto distinto de M que é dito estar presente.

No contexto da resolução de problemas, os conjuntos M, D, C podem ser assimilados à nossa base de conhecimentos, o conjunto M^+ , entretanto, simboliza as entradas dos sistemas e é construído gradualmente de forma incremental. A Figura 40 ilustra um exemplo.

Figura 40 - Abdução por cobertura



Fonte: elaborada pela autora

Definição 2:

Para um $d_i \in D$ e $m_j \in M$ quaisquer em um problema de diagnóstico $P = \langle D, M, C, M^+ \rangle$
 $efeitos(d_i) = \{m_j : \langle d_i, m_j \rangle \in C\}$, o conjunto dos objetos diretamente causados por d_i
 $causas(m_j) = \{d_i : \langle d_i, m_j \rangle \in C\}$, o conjunto dos objetos que podem diretamente causar m_j

A generalização desta definição para o caso em que o conjunto dos transtornos não será único:

Para todo $D_I \in D$ e $M_J \in M$ em um problema de diagnóstico $P = \langle D, M, C, M^+ \rangle$ $Efeitos(D_I) = \bigcup_{d_i \in D_I} efeitos(d_i)$.

$Causas(M_J) = \bigcup_{m_j \in M_J} causas(m_j)$.

Uma vez estes conjuntos definidos, é possível dar a definição formal da cobertura assim como os relativos ao critério de parcimônia.

Definição 3:

O conjunto $D_I \subseteq D$ é chamado cobertura de $M_J \subseteq M$ se $M_J \subseteq efeitos(D_I)$.

Ou mais simplesmente, um conjunto de transtornos D_I é uma cobertura de um conjunto de manifestações se e somente se o conjunto de efeitos (ou implicações diretas) de D_I inclui (de

maneira extensional) este conjunto de manifestações. Isso leva à definição da explicação no sentido da cobertura (PAGNUCCO, 1996):

Definição 4:

Um conjunto $E \subseteq D$ é considerado ser uma explicação de M^+ para um problema $P = \langle D, M, C, M^+ \rangle$ se e somente se E cobre M^+ e E satisfaz um dado critério de parcimônia.

Existem vários critérios de parcimônia, a pertinência de um critério em relação a outro depende do contexto no qual se trabalha; a seguinte definição expõe alguns critérios entre os mais utilizados.

Definição 5:

Uma cobertura D_I de M_J é dita minimal se sua cardinalidade é a menor entre todas as coberturas de M_J .

Uma cobertura D_I de M_J é dita não redundante se nenhum de seus sub-conjuntos não é também uma cobertura de M_J ; ela é redundante, caso contrário.

Uma cobertura D_I de M^+ é dita pertinente se ela é um sub-conjunto de $causas(M^+)$; ela é não pertinente, caso contrário.

Essas definições e propriedades aplicadas ao exemplo precedente permite definir os seguintes conjuntos:

$$Efeitos(d_2) = \{m_1\}$$

$$Causas(m_2) = \{d_1, d_3\}$$

$$Efeitos(\{d_2, d_3\}) = \{m_1, m_2\}$$

$$Causas(\{m_1, m_2\}) = \{d_1, d_2, d_3\} \dots etc$$

E supondo que as luzes e o forno não funcionem (i.e. $M^+ = \{m_1, m_2\}$), é possível encontrar os conjuntos de seguintes coberturas:

$\{d_1\}$ pertinente, minimal e não redundante

$\{d_2, d_3\}$ pertinente e não redundante

$\{d_1, d_2, d_3\}$ pertinente e redundante

$\{d_1, d_4\}$ não pertinente

Foi falado relativamente em detalhes sobre a abdução por cobertura sobre os conjuntos, pois

isto será visto mais tarde (Capítulo dos Resultados), ou seja, a abordagem da abdução no contexto da resolução dos problemas e sobretudo em um formalismo baseado sobre as primitivas dos grafos conceituais. Segue uma breve descrição de outras abordagens propostas pela inferência abdutiva.

b) Sistemas de manutenção da verdade

Os sistemas de manutenção da verdade ou ATMS (*Assumption-Based Truth Maintenance Systems*) foram inicialmente desenvolvidos por Reiter e deKleer (KLEER; REITER, 1987); tais sistemas operam em cooperação com um solucionador de problemas em um processo cíclico que pode ser descrito como segue:

- O resolvidor de problemas dispõe das informações contidas na base de conhecimento e é utilizado para realizar as inferências necessárias à geração de novas hipóteses. Uma vez feito isso, ele comunica os novos dados ao sistema de manutenção de verdade TMS.
- O sistema TMS mantém o controle das inferências que são consideradas como justificadas, ele recupera as novas inferências que têm origem no resolvidor de problemas e assegura a manutenção da verdade (consistência da base) validando as crenças já presentes ou modificando-as a fim de evitar qualquer contradição.
- No contexto da abdução, o resolvidor envia as observações ao TMS, este último retorna os contextos podendo ter gerado essas observações; a explicação será dada pelo conjunto minimal dos contextos onde a conjunção induz as manifestações observadas.

Esta abordagem se adequa bem ao raciocínio abdutivo; no entanto, verifica-se que ela apresenta problemas de computabilidade (BYLANDER et al., 1989), por consequência, não é adequada para uma implementação na área da computação.

c) Abordagem probabilística

Baseado sobre as estatísticas Bayesianas e operando sobre as probabilidades condicionais, seu princípio consiste em escolher a explicação onde a probabilidade que conheça os

elementos já presentes na base será maximal.

Uma abordagem proposta por Charniak e Simony (CHARNIAK, 1997) conecta cada caso a um custo, e em seguida, trata de fornecer a explicação cujo custo é o menor. Isto se reduz em um problema de percurso de grafo, a fim de encontrar o caminho à custo mínimo, este problema é chamado problema da melhor escolha (*Best selection problem*) (FLACH; KAKAS, 2000).

2.5.2.1.2 A abordagem intensional por abdução

Encontra-se a seguir várias técnicas de raciocínio onde alguns já foram vistos anteriormente, estes métodos podem ser reagrupados sob dois temas:

a) Abordagem baseada em casos

Esta abordagem consiste em aproveitar a experiência do passado, apresentando-a como sendo um contexto envolvendo uma série de conclusões. Nesse caso de abdução pela resolução de problemas, se trata de estudar a aplicabilidade de um contexto precedentemente estudado, e adaptar as explicações definidas no conteúdo atual; isto necessita uma base de conhecimentos incluindo contextos de experiências passadas assim que seus resultados, e terá como efeito melhorar a pesquisa e a validação das explicações (ESHGHI; KOWALSKI, 1989).

b) As abordagens lógicas:

O formalismo que trazem as representações lógicas permite definir o processo de abdução sobre uma tripla (Teoria do domínio, Observações, Hipóteses), a teoria do domínio ou *Background Theory* é o conjunto das fórmulas que formam o contexto no qual se fará nosso raciocínio, as observações quanto à elas representam os fatos deste modelo, a abdução permitirá então gerar, aprimorar e selecionar as hipóteses a fim de lhes apresentar como explicações destes fatos tendo o cuidado de respeitar a coerência destes últimos com a teoria do domínio.

Existem várias abordagens, citaremos a abordagem proposta por Poole (1988) para o

raciocínio por padrão e que é baseado sobre o sistema “teórico” que utiliza o raciocínio por abdução.

É possível também citar os trabalhos de Eshghi e Kowalski (ESHGHI; KOWALSKI, 1989) onde a abordagem muito parecida à do “teorista” é orientada para o domínio da programação lógica. Esta abordagem apresenta a noção de contexto da abdução (tripla <teoria, restrições, predicados abduíveis>) e adota a técnica da negação por falha para não selecionar as hipóteses potenciais mas sobretudo rejeitar os cenários incoerentes com o contexto da abdução.

2.5.2.2 Discussão

A abdução não tem certamente o aspecto perfeitamente formal que pode-se encontrar na dedução lógica, mas ela oferece a vantagem de permitir inferências mais “livres”; contrariamente à inferência dedutiva que não pode ser considerada num domínio de aplicação completamente definido (mundo fechado), a abdução pode ser aplicada em um meio “aberto” onde se pode encontrar os componentes do mundo que são somente parcialmente definidos, e outros que não são de todo definidos.

Certamente, teoricamente falando, a abdução não oferece o poder e a segurança dos resultados perfeitamente estabelecidos, mas ela permite entretanto - e em muitos casos - não se encontrar em uma situação onde o sistema inferencial permanece congelado por causa da falta de dados, dados justamente que poderiam ser deduzidos em seguida através de uma hipótese obtida por abdução, sem falar do ganho de tempo no caso de um sistema em tempo real onde os dados não são totalmente definidos mas fluem progressivamente da evolução do sistema no tempo, estes parâmetros fazem que a abdução possa se revelar crucial no número de aplicações práticas.

Por isto, a abdução se vê ganhando cada vez mais terreno na computação, e pode-se encontrar em domínios tão variados como a programação em lógica (KAKAS; KOWALSKI; TONI, 1992), a assimilação do conhecimento e o diagnóstico (KAKAS; MANCARELLA, 1991) e mais ainda em alguns tratamentos atribuídos sobre a base de dados e os SBC (PAGNUCCO, 1996).

2.5.3 *Raciocínio por analogia*

Na prática, a concepção dos sistemas baseados em conhecimentos se encontra geralmente confrontada aos problemas ligados à elucidação do conhecimento à tratar; acontece que em muitos domínios - particularmente no aspecto técnico - ela é descrita devido a experiência de usuários (supostos como especialistas do domínio) que expressam uma base de casos singulares e de experiências práticas e não possuem o modelo teórico englobando todos os aspectos do problema à tratar e oferecendo assim a possibilidade para a dedução de obter uma aproximação do problema universal.

Nesse gênero de casos, recorre-se habitualmente à diferentes abordagens heurísticas assim como outras destinadas à exploração direta das experiências adquiridas durante a prática e a simulação - sem tentar abranger todas as facetas do problema. O raciocínio por analogia é o representande mais comum desta abordagem.

2.5.3.1 Definições

O raciocínio por analogia é um tipo de raciocínio baseado sobre a exploração de resoluções anteriores dos problemas já encontrados (RIPOLL; COULON, 2001).

A primeira aplicação efetiva do raciocínio por analogia foi do programa de Evans em 1968 (BURSTEIN, 1986), programa para resolver testes de inteligência baseados sobre analogias entre figuras geométricas.

Sumariamente, o processo de analogia serve à elaboração da solução de um novo problema (denominado problema alvo) à partir de outros armazenados. O processo pode ser definido como segue:

- Encontrar um problema referência na base de conhecimentos, acompanhado de sua solução (denotado problema básico) capaz de ajudar à resolver o problema alvo;
- Medir a similaridade entre o problema básico e aquele a resolver e a adequação da solução do primeiro à resolução deste último;
- Adotar e ajustar - se possível - a solução do problema ao problema alvo, e adaptar à uma utilização ou uma apresentação ao usuário.

2.5.3.2 Pré-requisitos do raciocínio por analogia

A utilização do raciocínio por analogia requer certas hipóteses definidas sobre as diferentes partes colocados em jogo por este último, os mais importantes são (RIPOLL; COULON, 2001):

- O domínio da aplicação;
- As propriedades representacionais do formalismo de modelagem;
- Os mecanismos de exploração definidos sobre o formalismo de representação.

2.5.3.2.1 O domínio da aplicação

O domínio no qual se quer aplicar o raciocínio por analogia deve oferecer algumas garantias no que concerne a estabilidade das soluções encontradas ou à deduzir, ele deve então (DIDIERJEAN, 2001):

- Garantir a validade das soluções no espaço e no tempo - uma solução conhecida como válida por um problema encontrado permanecerá para futuras ocorrências do mesmo problema;
- A recorrência de casos já encontrados, ou pelo menos, situações similares tendo a mesma solução (porque senão, a analogia não será aplicável).

2.5.3.2.2 O formalismo de representação

Ele deve oferecer os mecanismos de representação dos problemas já resolvidos e isso, sob forma de base de casos. Um caso será então uma estrutura descrevendo um cenário (acontecido ou simulado), ele deve imperativamente descrever as três seguintes partes (DIDIERJEAN, 2001):

- **As premissas:** Esta parte descreve o problema à resolver, ela pode ser a descrição dos diferentes parâmetros de um sistema, os sintomas de uma anomalia, ou uma representação formal do objetivo a ser alcançado. As premissas devem ser implementadas de maneira à oferecer os métodos de indexação, de triagem e de

seleção de casos destinados à elucidação do problema alvo.

- **O contexto:** Os contextos permitem a consideração do estado inicial dos problemas, a descrição de seus ambientes de validade e podem mesmo em certos casos estar associados às premissas nos processos de estudo de similaridade entre casos. Eles devem representar os diferentes valores dos parâmetros exteriores aos sistemas onde a consideração é necessária para a validade do raciocínio.
- **A conclusão:** A conclusão representa as soluções propostas para o problema, ele pode também ser um texto para o usuário, uma equação tomando as premissas e o contexto como parâmetro ou uma ação a ser realizada automaticamente. É possível também incluir uma apresentação dos riscos eventuais que esta solução, uma previsão dos obstáculos que podem surgir antes, depois e/ou durante sua aplicação. Dois pontos importantes devem ser citados:
 - A correspondência perfeita entre o problema alvo e aqueles de base escolhidos geralmente nunca são alcançados e então é geralmente necessário realizar modificações nas soluções propostas, isto pode se fazer de maneira automática - tal como no raciocínio aproximativo - ou semi automático fazendo agir o utilizador como juiz frente às diferentes escolhas possíveis.
 - No caso onde o problema está conforme - ou não análogo - somente aos problemas de base não sendo jamais resolvidos de maneira satisfatória, é necessário então indica-lo especificando as ações que foram desenvolvidas e os resultados obtidos previamente.

Nota: Nos sistemas destinados à interação humana, adiciona-se geralmente aos casos uma explicação destinada aos usuários, ela pode comportar uma explicação precisa e detalhada da escolha da solução ou se limitar à um simples referencial teórico ou prático dos conceitos colocados em jogo.

2.5.3.2.3 Os mecanismos de exploração

Os diferentes mecanismos de raciocínio por analogia devem poder ser definidos sobre a base de casos implementados, e isto para poder executar as diferentes etapas do raciocínio.

Propostos por Campbell e Wolstencroft 1989-90, (WOLSTENCROFT, 1989) & (CAMP-

BELL; WOLSTENCROFT, 1990), estas etapas (número de 7) permitem uma boa visualização dos mecanismos colocados em jogo e seu encadeamento nesse tipo de raciocínio, se resumem em:

1. Identificação que o RA pode ajudar à resolução do problema alvo atual.
2. Pesquisa de um problema base análogo ao problema alvo.
3. Elaboração ou eliminação dos aspectos não pertinentes do problema base.
4. A correspondência das partes do problema da base com os aspectos similares do problema alvo.
5. Inferência ou dedução de características (soluções) atribuídas ao problema alvo à partir dos presentes no problema de base.
6. Justificação ou verificação da validade das conclusões inferidas.
7. Aprendizagem, quer dizer, memorização do problema alvo enriquecido de sua solução.

O raciocínio por analogia é um modelo bastante amplo na medida em que não impõe nenhuma restrição sobre o número de domínios à gerenciar ou a ter em conta durante o processo de inferência. Uma forma mais restrita do raciocínio por analogia é o raciocínio com base em casos que, manipula somente as entidades de um mesmo domínio a fim de poder propor uma solução. O exemplo mais frequentemente citado para ilustrar esta diferença é o da pesquisa da solução de um problema médico à partir da solução de um problema militar análogo.

O raciocínio com base em casos é uma técnica de raciocínio tendo feito suas provas nos domínios de controle de sistemas (diagnóstico, monitoramento, manutenção etc.) ou da concepção e da elaboração de planos. Os sistemas desenvolvidos à partir desta técnica são destinados à trazer uma ajuda ao usuário nestes diferentes domínios. Isto permite também reagrupar a perícia de uma empresa e coloca-la à disposição de qualquer pessoa habilitada a consulta-la, perícia que será agora incluída no capital da empresa.

2.5.4 Discussão

Foram apresentados nesta seção diferentes abordagens utilizadas atualmente para efetuar as inferências sobre um sistema baseado em conhecimento. A primeira a ser apresentada foi a dedução que continua sem dúvida a mais segura e a mais difundida, em seguida foi

abordada a abdução e o raciocínio por analogia, e foram expostas para essas três abordagens inferenciais as diferentes premissas necessárias à sua implantação em um sistema que opera sobre as bases de conhecimentos assim como as diferentes técnicas de exploração que lhes são inerentes.

A questão que se coloca é: Estas premissas sobre as quais se fundamentam estas abordagens são disponíveis no formalismo dos grafos conceituais que este estudo propõe como substituto aos métodos tradicionais utilizados até o presente? As técnicas utilizadas nos processos de inferência são aplicáveis em uma plataforma baseada sobre os grafos conceituais?

No Capítulo 3, será proposta uma nova abordagem de edição e de manipulação das ontologias tentando retranscrever a linguagem principal utilizada para esta finalidade, ou seja, OWL. Serão definidas, para isto, os diferentes conceitos sobre os quais ele será baseado e as técnicas que empregará. A segunda parte, se encarregará à responder a questão precedente estudando a aplicabilidade dos três métodos de inferência estudados durante este capítulo na nova linguagem OWL resultante onde os fundamentos serão construídos no formalismo dos grafos conceituais.

2.6 Trabalhos Relacionados

Nesta seção são apresentados, resumidamente e em ordem cronológica, os trabalhos relacionados referentes a traduções entre ontologias, OWL e RDF e Grafos Conceituais. Também são apontadas as diferenças conceituais desses trabalhos com a abordagem proposta nesta tese.

O critério para inclusão de trabalhos nesse capítulo foi selecionar aqueles que, de alguma forma, sugerem a incorporação de boas práticas para a tradução das primitivas OWL para o formalismo dos grafos conceituais, os quais podem envolver questões de melhoria na linguagem proposta e modelos de desenvolvimento.

Este estudo, sobre os trabalhos relacionados, foi necessário para que a tese, aqui apresentada, fosse desenvolvida baseada em contribuições originais e pertinentes, advindas de pesquisas já validadas.

2.6.1 *Conceptual Graphs for Semantic Web Applications*

Dieng-Kuntz *et al.* (2005) pretendem mostrar as vantagens do formalismo dos Grafos Conceituais para a Web Semântica através de várias aplicações do mundo real em um framework da *Corporate Semantic Webs*. Descrevem um motor de busca semântico RDF(S)-dedicado, *Conceptual Resource Search Engine* (CORESE), baseado em uma correspondência entre RDF(S) e Grafos Conceituais, e ilustram o interesse de Grafos Conceituais através da análise de várias aplicações no mundo real.

Apontam que quando Tim Berners-Lee apresentou sua visão da Web Semântica, várias comunidades de pesquisa estudaram a fundo como os resultados de seus campos de pesquisa poderiam contribuir para alcançar este objetivo ambicioso. Particularmente, pesquisadores em Representação do Conhecimento reconheceram o importante potencial do papel que os formalismos de Knowledge Representation (KR) poderia representar para representar as ontologias necessárias na Web Semântica. Formalismos de representação Orientados à objetos (OO), Lógicas da Descrição (DL) e Grafos Conceituais (CG) foram os principais candidatos para alcançar este propósito. A comunidade DL estava fortemente envolvida na definição da *Ontology Web Language* (OWL).

Entretanto, alguns pesquisadores da comunidade de CG também trouxeram sua contribuição muito cedo, com várias estratégias. Alguns pesquisadores adotaram CG diretamente como formalismo para representação de ontologias no contexto da Web Semântica, por exemplo, WebKB. Outros preferiram contar com uma correspondência entre CG e RDF(S): a equipe ACACIA propôs e implementou uma tradução de RDF(S) em CG e construiu um motor de busca semântico, CORESE, baseado nesta correspondência, enquanto outros autores sugeriram usar CG como Ontolingua para permitir a tradução automática de estruturas de conhecimento entre diferentes formalismos KR, e descreveram um metamodelo RDF(S) em CG.

Dieng-Kuntz *et al.* (2005) resumem a abordagem da equipe ACACIA para a Web Semântica e enfatizam o papel dos CGs nesta abordagem. Os autores mostram que no *framework* da Web Semântica, Grafos Conceituais têm expressividade suficiente para Representação do Conhecimento e capacidade de raciocínio suficiente para aplicações do mundo real.

2.6.2 *Raisonnement sur des ontologies lourdes à l'aide de Graphes Conceptuels*

Fürst *et al.* (2006) enfatizam o desafio atual para desenvolver um trabalho dedicado à engenharia de ontologias pesadas (*heavyweight ontologies*), tanto de um ponto de vista de representação de conhecimentos (em particular os axiomas) quanto de um ponto de vista de mecanismos de implementação de raciocínio subjacente às atividades de avaliação, alinhamento e fusão de ontologias pesadas. Em seguida, se trata de mostrar que um ambiente fundamentado sobre o modelo dos grafos conceituais revela-se uma solução eficaz para tal desafio, sendo que os axiomas, elementos intrínsecos das ontologias pesadas, são naturalmente, representáveis em termos de grafos e facilmente e facilmente e exploráveis usando as operações de gráficos.

Para raciocinar sobre uma ontologia de domínio, os autores sugerem representá-la em um nível de abstração suplementar (chamado nível meta) de forma a considerá-la como uma base de conhecimentos sobre a qual é possível de implementar diferentes tipos de atividades ligadas ao raciocínio tais como o alinhamento, a fusão ou a avaliação. A abordagem do trabalho é baseada no uso de uma representação de ontologias (nomeada MetaOCGL) para representar uma ontologia de domínio expressa em Grafos Conceituais. MetaOCGL é uma ontologia de representação que descreve todas as primitivas da linguagem e sua semântica formal.

Além disso, o trabalho mostra as vantagens que aporta uma solução baseada sobre os Grafos conceituais para a engenharia de ontologias pesadas, no quadro de uma das problemáticas centrais de interoperabilidade semântica: o alinhamento de ontologias de domínio.

Importante destacar que os trabalhos dedicados ao raciocínio sobre as ontologias podem igualmente contribuir para outras atividades relevantes da engenharia de ontologias, principalmente a avaliação no contexto onde esta abordagem permite à engenharia de conhecimentos definir explicitamente, através da definição dos axiomas ao nível meta, os

critérios à utilizar para avaliar o conteúdo das ontologias em termos de consistência, completude e concisão. Esta abordagem declarative dos critérios à nível conceitual aumenta a portabilidade e a modularidade de mecanismos de avaliação de ontologias que, na maioria dos casos, são diretamente implementados no código de ferramentas de engenharia ontológica tal como Protégé.

2.6.3 Ontology Mapping Using Fuzzy Conceptual Graphs and Rules

BUCHE *et al.* (2008) apresentam um novo método de mapeamento de ontologias entre uma ontologia fonte e outra alvo considerada como uma referência. Ambas ontologias são compostas de triplas da forma (objeto, característica, valor). Valores que descrevem os objetos da ontologia de referência são hierarquicamente organizados usando um tipo de relação.

O método proposto considera o problema de mapeamento de ontologias como um problema de aplicação de regras no modelo dos Grafos Conceituais. Primeiro, um vocabulário comum para ambas ontologias é definido usando mapeamento entre valores e características. Cada valor da ontologia de origem é associado com um conjunto de valores da ontologia de referência. Então, a ontologia de origem é traduzida em uma base de grafo conceitual difuso e a ontologia de referência em uma regra de base do grafo conceitual. Finalmente, regras são aplicadas na base de fatos para encontrar correspondências entre objetos de ambas as ontologias. Este método é implementado e aplicado para o mapeamento de ontologias para várias aplicações do mundo real.

Os autores mostram que, devido a extensão difusa do modelo CG, é possível representar e manipular resultados de mapeamento léxico com propriedades semânticas. Afirmam além disso, que a experimentação mostrou que o método tem uma boa resposta, porem um precisão pobre.

A primeria perspectiva para melhorar o método citado é estudar outras técnicas comparativas entre característias e valores tais como técnicas semânticas ou técnicas de combinação contextual. Outra perspectiva é aplicar, em tratamento posterior, restrições semânticas nos mapeamentos gerados entre objetos. Finalmente, importante comparar os resultados do trabalho com os obtidos com o uso de outros métodos de alinhamento de ontologias, graças a

sistemas de comparação de alinhamento de ontologias.

2.6.4 *Translations between RDF(S) and Conceptual Graphs*

Baget *et al.* (2010) apresentam transformações entre Grafos Conceituais (CGs) e a linguagem RDF(S), o padrão para anotações da Web Semântica. O objetivo do artigo é relacionar as duas linguagens, haja vista as diferenças entre esses formalismos tem sido uma fonte de dificuldades na tentativa de traduzir objetos de uma linguagem para a outra. Após elencar as dificuldades, propõem duas transformações, que foram posteriormente implementadas na plataforma CoGUI.

Segundo os autores, a propriedade fundamental para tais transformações é a preservação da noção de vinculação semântica (a base para o raciocínio, portanto, para a consulta). Outras propriedades desejáveis são o aspecto natural da transformação, por exemplo, a concisão e intuitividade dos objetos gerados, da mesma forma a preservação de algumas propriedades de algoritmos da linguagem a ser traduzida. Desenvolvendo estas transformações, algumas tarefas da Web Semântica serão beneficiadas, onde otimizações estruturais e baseadas em grafos (amplamente abordadas pelos Grafos Conceituais) são necessários.

O trabalho apresenta contribuições importantes, por exemplo, a segunda transformação. Pois esta transformação está alinhada com a visão de representação de conhecimento da Web Semântica, no sentido que ela distingue claramente entre diferentes tipos de conhecimento. Além disso, a implementação em CoGUI permite visualizar o conhecimento base obtido de acordo com esta separação. Isso define um fragmento de RDF(S) que pode ser fornecido com a semântica através de lógicas de primeira-ordem, que é compatível com a maioria das lógicas de descrição, em particular o fragmento OWL-DL.

Portanto, é potencialmente extensível para uma ontologia mais rica, que poderia ser representada através das lógicas de descrição. Na verdade, muitos arquivos RDFS usam recursos OWL simples. A combinação de RDFS e OWL-DL em documentos leva a uma explosão combinatória de problema de consulta. Os autores afirmam que existem inúmeros estudos sobre o preciso relacionamento entre essas restrições e fragmentos de CGs no contexto de resposta às consultas. Concluímos que o trabalho destes autores pode ser um grande aliado ao desenvolvimento de futuros trabalhos relacionados com a assunto da tese.

2.6.5 Matching and merging of ontologies using conceptual graphs

Ganapathy *et al.* (2011) definem *matching* de m grafos conceituais e apresentam aspectos matemáticos de correspondência e ligação de m grafos conceituais. Os algoritmos desenvolvidos para combinação e fusão são mostrados no artigo.

Aplicações de gestão do conhecimento precisam determinar se duas ou mais representações codificam o mesmo conhecimento. A solução para o problema de “combinação” é difícil porque representações podem codificar o mesmo conteúdo mas diferem substancialmente na forma. Abordagens anteriores para este problema usaram medidas sintáticas ou conhecimento semântico para determinar a distância entre duas representações.

A representação captura o mesmo conhecimento se seus gráficos correspondentes correspondem. A codificação múltipla do mesmo conhecimento raramente combina exatamente, dessa forma um *matcher* deve ser flexível para evitar uma alta taxa de falso-negativos. Entretanto, um *matcher* que é muito flexível pode sofrer de uma alta taxa de falso-positivos. Este problema tem várias causas, incluindo (i) A ontologia é suficientemente expressiva para permitir que a mesma informação seja codificada de diferentes formas, (ii) As representações são construídas por diferentes engenheiros do conhecimento (ou programas de computador), aumentando a probabilidade de diferirem e (iii) As representações são grandes, aumentando a oportunidade de diferenças.

No trabalho o algoritmo generalizado para combinar m grafos conceituais é desenvolvido. A exatidão do grafo combinado é investigado depois. Além disso o algoritmo é estendido para combinar m grafos conceituais.

3 Metodologia

A partir do final da década de 1990 um grande número de esforços de pesquisa exploraram como a idéia de representação de conhecimento em inteligência artificial poderia ser utilizada na Web. Estes esforços incluíram linguagens baseadas em HTML (chamada SHOE), XML (chamada XOL e, mais tarde, OIL), e várias linguagens baseadas em *frames* e abordagens de aquisição de conhecimento. Até que mais tarde surgiu a OWL baseada nas lógicas de descrição (GÓMEZ-PÉREZ; CORCHO, 2002).

Entretanto, a camada semântica da Web apresenta diversos desafios, devido a característica questionável da confiabilidade das informações. Existem muitas pesquisas sobre como as ontologias, pelo fato de serem consideradas “uma especificação explícita e legível por máquina de uma conceitualização compartilhada” (HENDLER; BERNERS-LEE, 2010), pode resolver a troca de dados heterogêneos.

O presente trabalho teve seu ponto de partida nas pesquisas sobre novas aplicações baseadas em ontologias e arquiteturas de conhecimento para a nova Web. Pois o clamor para as diversas abordagens existentes tem origem na necessidade de linguagens para representar a informação semântica. Portanto, após a pesquisa inicial das linguagens existentes, o formalismo dos grafos conceituais foi escolhido como o *insight* inicial para uma nova linguagem de representação.

Após estudo exploratório realizado por meio de pesquisas bibliográficas em diversas fontes citadas, foi definida uma metodologia para desenvolver a nova representação e destacar sua contribuição para a camada semântica da Web. A metodologia é o conjunto de processos ou operações mentais empregadas na pesquisa científica (Gil 2002; Feyerabend e Motterlini 1999). Para Gil (2002), p.42, a pesquisa tem caráter pragmático, é um “processo formal e sistemático de desenvolvimento do método científico. O objetivo fundamental da pesquisa é descobrir respostas para problemas mediante o emprego de procedimentos científicos”. Assim, a metodologia proposta tem a preocupação de minimizar o caráter subjetivo da pesquisa, pois sistematiza esta ação, primeiramente pautada pela interpretação, e evita impasses na extensão dos resultados a outros contextos.

Com o objetivo de propor uma tradução das primitivas de representação OWL, considerando diferentes formas de raciocínio, foram estudados a fundo aspectos da Web, Web Semântica, ontologias, OWL, formalismos de representação e técnicas de inferência. Em busca de

algumas lacunas da OWL, foram encontradas: falta de suporte para tipos de dados complexos, falta de noção de variável, inadaptação para a representação de lógicas de ordem superior à 1, indecibilidade para alguns fragmentos da linguagem (OWL full) e o tempo de cálculo NP-completo para outros (OWL DL), inadaptação (do A-Box do OWL) para as manipulações de um número significativo de indivíduos e a inexistência de classes pré-definidas que poderiam ser utilizadas na especificação das ontologias.

Para tentar superar essas limitações, foi proposta a reescritura de suas primitivas de representação e de seus mecanismos práticos de ativação de conhecimentos baseados nos paradigmas do formalismo dos grafos conceituais, como será visto no capítulo dos resultados.

Foi proposta uma representação de duas bases de conhecimento, uma escrita em OWL e outra com a ajuda da linguagem proposta (LD's e CG's); e a criação do T_Box e A_Box para a nova linguagem. E em seguida, como será visto na segunda parte dos resultados, mostramos como o raciocínio dedutivo poderá ser aplicado aos grafos conceituais, da mesma forma o estudo da aplicabilidade das formas de raciocínio abduutivo e por analogia.

3.1 Pesquisa documental

Esta etapa é fundamental para caracterizar os problemas e carências da camada semântica da Web, a estrutura da Web Semântica, a identificação de recursos, a importância dos metadados, definição e propriedade das ontologias e formalismos de representação. A pesquisa documental se constitui da pesquisa, leitura e interpretação de documentos científicos (artigos, teses, dissertações, livros, etc.) nos idiomas português, inglês, espanhol e francês, com preferência às publicações mais atuais. Os resultados dessa etapa antecedem as demais, pois subsidiam a compreensão do contexto da pesquisa e ressaltam como os assuntos abordados contribuem para o desenvolvimento do modelo a ser proposto. Os resultados desta etapa são apresentados na Revisão bibliográfica (Capítulo 2), com o seguinte destaque:

- A Ciência da Informação: apresenta-se nesta seção as origens e evolução da Ciência da Informação, bem como suas definições, sua relação com o conceito de ‘informação’, a natureza interdisciplinar da Ciência da Informação, com ênfase na interdisciplinaridade com a Ciência da Computação e, por fim, sobre a necessidade da Ciência da Informação.
- A Arquitetura da Informação: são apresentadas as definições atuais da Arquitetura da Informação, sua importância para a Ciência da Informação e interdisciplinaridade com outras ciências, em particular, com a Informática, e a relação e importância desta para com a Web.
- A Web Semântica: são apresentados as diferentes definições da Web semântica, seus objetivos, assim como as diferentes tecnologias que constituem a plataforma aplicada para sua elaboração. Isto foi feito enfatizando a importância da descrição de recursos (devido as linguagens de formulação dos metadados como RDF ou RDFS ou outras ontologias de representação como OWL) e dos métodos de exploração de suas descrições (motores de inferência, técnicas de exploração ligados aos formalismos das linguagens de representação).
- Os formalismos de representação: são elencados diferentes formalismos de representação de conhecimentos (seus fundamentos teóricos e especificações). O trabalho se interessa particularmente às lógicas de predicados (sobre a qual se baseia a maior parte dos motores de inferência atuais), as lógicas de descrição (amplamente utilizadas na Web semântica) assim como o formalismo dos grafos conceituais (sobre o qual se baseia o presente trabalho).
- Técnicas de inferência: são abordados três técnicas de inferência potencialmente - dedutiva, abdutiva e resolução do problema por analogia.

3.2 Modelagem

Adquirida a fundamentação teórica, consolidou-se a modelagem proposta. Foi definida uma tradução das primitivas de construção da ontologia da linguagem OWL através de outras

primitivas retiradas do formalismo dos grafos conceituais. E mostramos a aplicação das técnicas de inferência citadas na Revisão da Literatura na nova linguagem proposta. Os passos foram definidos como segue:

1. Parte 1: Definição da linguagem SOWL.

- (a) Definir as hierarquias de tipos de conceitos e suas relações;
- (b) Relacionar e definir as primitivas terminológicas;
- (c) Especificar os construtores (schemas de axiomas);
- (d) Declarar os tipos e as propriedades sobre os mesmos;
- (e) Definir as propriedades das relações;
- (f) Catalogar os indivíduos;
- (g) Criar a base de fatos.

2. Parte 2: Definição dos mecanismos de inferência.

- (a) Traduzir para lógica de predicados;
- (b) Implementar as regras de dedução e de comportamento dos axiomas;
- (c) Utilizar o cálculo lambda para a avaliação e expansão das preposições;
- (d) Criar o domínio de aplicação.

3. Parte 3: Exemplos com a nova linguagem SOWL

4 Resultado

4.1 Resultado 1 - Linguagem SOWL

A adequação das técnicas de representação de conhecimentos ao ambiente da Web Semântica parece ser uma das condições “*sine qua non*” de sua propriedade e sua aceitação como plataforma de desenvolvimento e evolução legítima da Web atual.

Por isto, a linguagem OWL é classificada como a linguagem principal para a especificação de ontologias, pedras angulares dos métodos de raciocínio adotados pelos motores de inferência mais difundidos atualmente e base de toda manipulação automática da informação de grande escala.

Este privilégio é devido ao fato que esta linguagem explora as diferentes tecnologias desenvolvidas em torno dos formalismos das lógicas de descrição e aquelas dos Frames, sua integração de mecanismos de correspondência a ontologias distribuídas (permitindo assim a reutilização de representações, a definição incremental de conhecimentos e o raciocínio sobre os conhecimentos distribuídos) assim como a facilidade de utilização de seus construtores através das plataformas de manipulação efetivamente em ação (Protégé, OntEd...).

Entretanto, OWL permanece ainda uma linguagem em curso de especificação e apesar do fato de ter sido provada em vários domínios de aplicação, apresenta ainda algumas lacunas tais como:

- A falta de suporte para tipos de dados (*datatype*) complexos;
- A falta de noção de variável;
- A inadaptação para a representação de lógicas de ordem superior à 1;
- A indecibilidade para alguns fragmentos da linguagem (OWL *full*) e o tempo de cálculo NP-Completo para outros (OWL *DL*);
- A inadaptação (do A-Box do OWL) para as manipulações de um número significativo de indivíduos (aspecto inevitável para as aplicações avançadas da Web);
- A inexistência de classes pré-definidas que poderiam ser utilizadas na especificação das ontologias.

Para tentar superar essas limitações, é proposto neste capítulo uma re-escritura de suas primitivas de representação (vindo dos LDs e dos Frames) e de seus mecanismos práticos de ativação de conhecimentos (necessários à toda linguagem operacional) e isto baseando-se sobre os paradigmas do formalismo dos grafos conceituais e os diferentes mecanismos neles ligados.

Esta abordagem foi motivada pelo fato de que os grafos conceituais apresentam várias vantagens de natureza (SOWA, 1984):

– Teórica:

Os GCs constituem um formalismo poderoso baseado sobre as lógicas clássicas; a utilização do cálculo lambda para a formalização das primitivas terminológicas e o fato que eles introduziram a noção de variáveis lambda torna possível a separação entre a estrutura dos dados e suas representações reais (separação da intenção e da extensão);

– Cognitiva:

A modelagem gráfica, a modularidade da representação e a esquematização relacional deste formalismo permitem uma melhor modelagem mental e então uma melhor apreensão dos conhecimentos; além disso, oferece a possibilidade de representar os conhecimentos informais ou de natureza intuitiva (tal como a linguagem natural);

– Operacional:

Este formalismo já é adotado por vários sistemas e por várias utilizações (projeto de bases de dados, extração de informações, o tratamento das linguagens etc.), ele é também utilizado como plataforma de troca de informações inter formalismos (parte da facilidade de sua retranscrição para vários formalismos); ele permite entre outras coisas de otimizar os mecanismos de pesquisa e de extração de dados devido as manipulações a partir da teoria dos grafos e atribui um tratamento sequencial das consultas à medida de seu avanço (como as linguagens de consulta SQL, por exemplo).

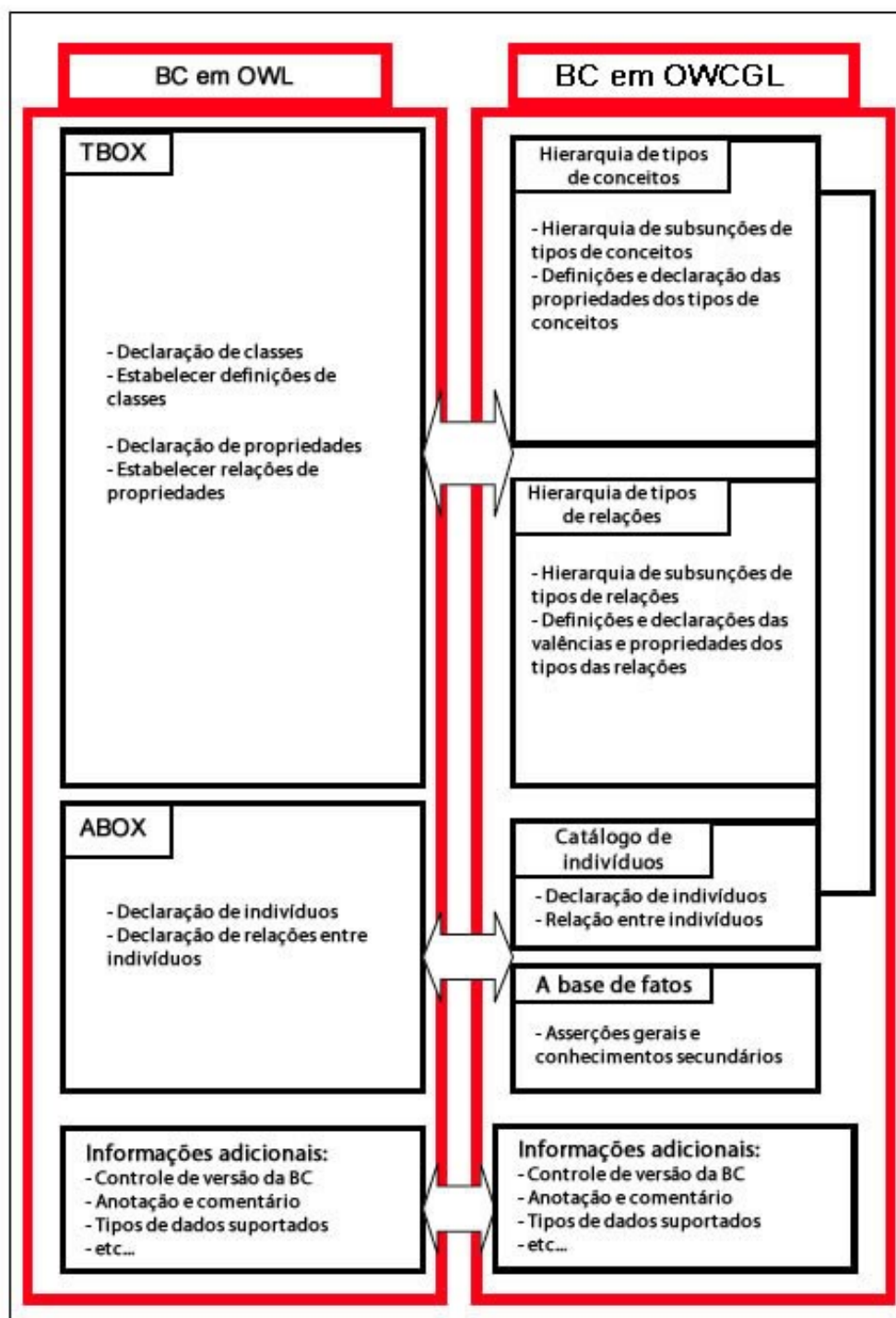
A linguagem proposta é fruto de um estudo sobre as similaridades entre o formalismo dos grafos conceituais e aqueles das lógicas de descrição. Optou-se pela utilização dos contextos pertencentes aos GCs para representar a maioria dos construtores utilizados pela linguagem OWL (mais particularmente OWL-DL), e foram propostos os construtores personalizados para abrir o caminho para outras opções de representação (gestão das relações n-árias, por

exemplo); depois disso, foi proposto como perspectiva um estudo da linguagem obtida e sob o ponto de vista:

- Expressividade: Com o objetivo de facilitar a especificação de conhecimentos representáveis devido a DL, até mesmo a representação de outros tipos de conhecimentos (tais como os conhecimentos modais ou difusos);
- Exploração: Com o objetivo de permitir a aplicação de novos mecanismos de raciocínio desenvolvidos para os GCs, possivelmente mais eficientes (tempo de cálculo, consultas mais livres, melhores combinações dos conhecimentos etc)

Primeiramente, a representação e da correspondência de duas bases de conhecimentos, uma escrita em OWL e outra com a ajuda da linguagem proposta e isto reagrupando os conhecimentos representados seguindo seus formalismos respectivos (LD's e CG's). Como mostra a Figura 41.

Figura 41 - Bases de conhecimento OWL e SOWL



Fonte: elaborada pela autora

Toda a base de conhecimentos da nova linguagem SOWL será então representada por um contexto do tipo KnowledgeBase representado como segue:

```
[“Knowledgebase”:
  { [“TypeHierarchy”: CG ] [“RelationHierarchy”: CG ] [“CatalogOfIndividuals”: CG]
  [“Assertion”: CG] }
```

Onde os CGs são os grafos conceituais que descrevem o conteúdo dos contextos que eles incluem. Os conhecimentos representados pela base serão repartidos em quatro contextos (três para o suporte e um para os fatos).

4.1.1 *TBOX* → Hierarquias de tipos de conceitos e de relações

Antes de começar a reescritura dos construtores terminológicos do OWL (baseados sobre as LDs) na nossa nova linguagem (baseada sobre os grafos conceituais), será útil apresentar primeiro como serão traduzidos:

- As primitivas terminológicas da linguagem: como é que os conceitos e os papéis serão definidos;
- A relação de subordinação: dado sua importância, é imperativo definir bem sua semântica.

4.1.1.1 Relação e definição das primitivas terminológicas

As primitivas terminológicas das LDs serão representadas nos grafos conceituais devido as noções de tipos de conceitos e de tipos de relações.

4.1.1.1.1 Os tipos de conceitos

A noção de conceito (no âmbito das LD's) é definido como sendo um exemplo de indivíduos tendo propriedades comuns, reagrupadas em classes, em GC's, isto correspondia mais à definição dos tipos de conceitos que aqueles dos conceitos eles mesmos.

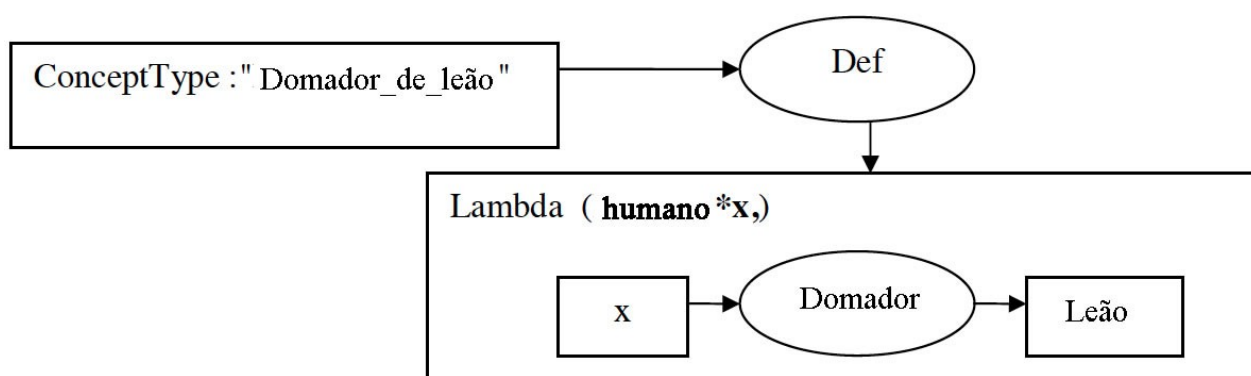
De fato, os tipos de conceitos são os conjuntos de indivíduos definidos com a ajuda de

fórmulas especificadas em Cálculo Lambda (o nome do tipo - o *ConceptType* - é na verdade um atalho para manipular a expressão lambda). Exemplo na Figura 42:

Um domador de leões:

```
(Def [ConceptType "Domador_de_leao"] [Lambda (humain *x) (Dresser ?x
[Leao]))
```

Figura 42 - Exemplo de tipo de conceito



Fonte: elaborada pela autora

Quanto aos conceitos dos GC's, por sua definição, permitindo “designar” qualquer entidade do domínio modelado (indivíduos - pontuados com a ajuda de marcadores ou quaisquer outros - conjuntos, coleções, etc.) e de lhe atribuir as noções existenciais, universais, de cardinalidades ou outras, devido a especificação de quantificadores; desse modo, a noção de conceito engloba aquele indivíduo das DL's e oferece então uma maior expressividade, o que será mais detalhado durante a especificação do catálogo de indivíduos).

4.1.1.1.2 Os tipos de relações

A correspondência entre papéis nas LD's e as relações conceituais se fará muito facilmente devido a noção de relações matemáticas, e isto definindo os papéis como sendo relações conceituais de valência 2 tendo como assinatura (Dominio, Imagem); sua definição se fará devido as expressões lambda tendo como assinaturas aquelas das relações representadas e definindo o conjunto dos argumentos verificando aquela relação. Estas expressões lambda

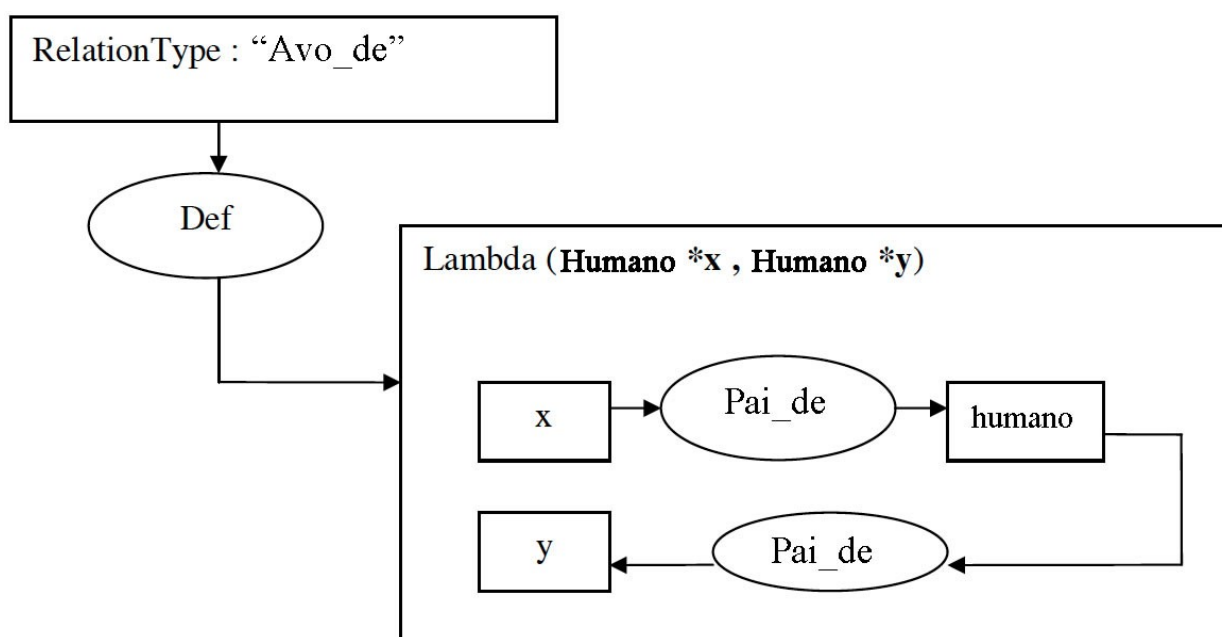
serão equivalentes aos identificadores de relações do tipo RelationType.

Definição da relação avô:

```
(def [RelationType "avo_de"] [lambda (Humano *x, Humano *y) (pai_de ?x
[Humano *z] )] (Pai_de ?z ?x ) ] )
```

Exemplo na Figura 43:

Figura 43 - Exemplo de tipo de relação



Fonte: elaborada pela autora

Notas:

- Os tipos de conceitos e de relações associados às expressões lambda como definições serão consideradas como os tipos definidos, os outros serão considerados como primitivos;
- Os ConceptType e as RelationType são as cadeias de caracteres declaradas como identificadores de tipos (de conceitos ou de relações).

4.1.1.2 As subordinações como relação de ordem

As relações de ordem utilizadas para a hierarquização dos tipos de conceitos e dos tipos de relações nos GC's (ordem parcial) não possuem as semânticas "livres", dando assim a

possibilidade aos “*designers*” de ajustar seu significado às necessidades da modelagem desejada.

Se por exemplo a hierarquia de tipos de conceitos designada é uma hierarquia de parentes (filho, pai, mãe, tio, avô, etc.). A relação de subordinação entre conceitos poderá se ver atribuída a seguinte semântica de idade:

X subordina Y significa que X é mais velho que Y (exemplo: pai subordina filho)

Ou então se é necessário organizar uma hierarquia de funções como sendo as relações conceituais, a subordinação poderia ser atribuída o significado da derivada (a função ‘*f*’ subordina a função ‘*f*’).

No caso do presente trabalho, será atribuído à subordinação (que denotamos ‘>’) de tipos de conceitos a semântica expressa pela seguinte fórmula lógica de primeira ordem :

$(C > C') \Leftrightarrow (\forall x \rightarrow C'(x))$ se um tipo de conceito *C* subordina um outro tipo de conceito *C'* então toda instância de *C'* pertence à *C*

Da mesma forma será atribuído à subordinação de tipos de relações de valência *n* e de assinatura (T_1, T_2, \dots, T_n) a semântica definida pela seguinte fórmula lógica:

$(R > R') (\forall x_1, x_2, \dots, x_n (T_1(x_1) \wedge T_2(x_2) \wedge \dots \wedge T_n(x_n) \wedge R'(x_1, x_2, \dots, x_n) \rightarrow R(x_1, x_2, \dots, x_n)))$

Isto equivale à dizer que se um vetor de indivíduos verifica uma relação *R'*, então ele verifica também a relação *R* que o subordina.

Notas:

- As equivalências das classes *thing* e *nothing* do OWL existem em GC’s e são nomeadas *T* (Entidade) e *⊥* (Absurdo), o primeiro subordina todos os tipos de conceitos e o segundo se faz subordinar por todos os tipos de conceitos;
- Todas as relações conceituais de valência *n*, são subordinadas por uma relação abstrata de valência *n* e de assinatura (Entidade, . . . , Entidade, Entidade);
- Nota-se que duas relações *R* e *R'* de valências respectivas *n* e *m* tal que $n \neq m$ não admitem relações de ordem entre elas ($R > R'$ e $R' > R$ são axiomáticamente falsas);
- Toda primitiva conceitual deve ser mencionada na hierarquia de subordinação do tipo correspondente à sua natureza.

Da mesma forma, podemos definir a relação $<$ (especialização) como sendo o inverso da subordinação ou a relação $=$ que é a relação de equivalência para uma dupla subordinação.

4.1.1.3 Especificação dos construtores (esquemas de axiomas)

Será apresentado os diferentes construtores da linguagem do presente trabalho descrevendo suas intervenções na especificação de uma base de conhecimentos.

4.1.1.3.1 Hierarquia de tipos de conceitos

Será definido aqui todas as informações a respeito dos tipos de conceitos encontrados em nossa base, suas informações serão apresentadas em um contexto de tipo *TypeHierarchy* definido como segue:

```
“[ “TypeHierarchy (TypeLabelOrdering | TypeDefinition | Typeproperty)* “]”
```

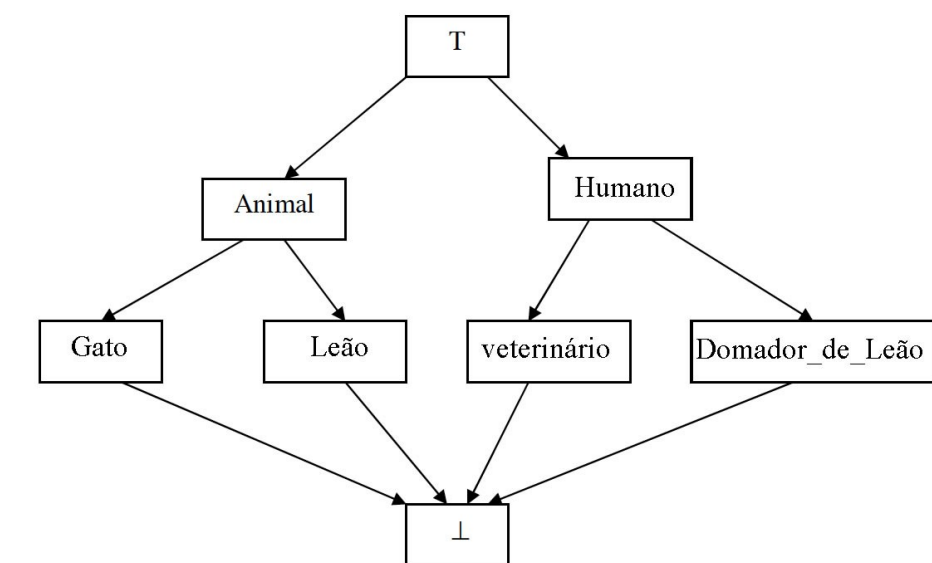
a) Construção da hierarquia de tipos (*TypeLabelOrdering*)

Se faz devido as relações de ordem apresentadas acima. Cada subordinação será expressa em um contexto de tipo *TypeLabelOrdering*:

```
“( “EQ” | “GT” | “LT” )
“[ “ConceptType” “” “ConceptType” “” “]”
“[ “ConceptType” “” “ConceptType” “” “]” “)”
```

A relação EQ é a relação de equivalência entre tipos de conceitos, GT é a subordinação quanto à LT, ela representa a especialização de conceito. Exemplo mostrado na Figura 44 (os arcos representam a relação de subordinação):

Figura 44 - Exemplo de relação de equivalência



Fonte: elaborada pela autora

Em SOWL:

```

(GT [ConceptType "T"][ ConceptType "Animal"])
(GT [ConceptType "T"][ ConceptType "Humano"])
(GT [ConceptType "Animal"][ ConceptType "Gato"])
(GT [ConceptType "Animal"][ ConceptType "Leao"])
(GT [ConceptType "Humano"][ ConceptType "Domador_de_leao"])
(GT [ConceptType "Humano"][ ConceptType "Veterinario"])
(GT [ConceptType "Gato"][ ConceptType "⊥" ])
(GT [ConceptType "Leao"][ ConceptType "⊥" ])
(GT [ConceptType "Veterinario"][ ConceptType "⊥"])
(GT [ConceptType "Domador_de_leao"][ ConceptType "⊥" ])
  
```

b) A declaração de tipos (*TypeDefinition*)

A declaração de tipos se fará devido à relação *Def* que ligará os tipos e suas definições em cálculo lambda (implementados devido aos contextos lambda de valência 1) uma definição de tipos se faz da seguinte maneira:

```

“(” “Def” t
“[” “TypeLabel” “”” ConceptType “””“]”
“[” “Lambda” “”” LambdaExpression” “” “[” “)”

```

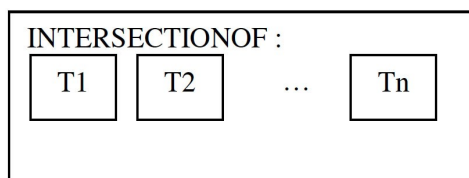
* Ver o exemplo do domador de leão.

A expressão das definições dos tipos poderia assim se fazer pela combinação dos seguintes construtores (inteiramente representáveis em expressões lambda).

- A interseção de tipos

A interseção de tipos será descrito por um contexto do tipo UNIONOF tendo por descritor um conjunto de conceitos únicos tendo por tipo os tipos de unir e um quantificador vazio. Isso é exemplificado na Figura 45:

Figura 45 - Interseção de tipos



Fonte: elaborada pela autora

Fórmula lógica equivalente:

$(x \text{ elemento da interseção}) \Leftrightarrow (T_1(x) \wedge T_2(x) \wedge \dots \wedge T_n(x))$ tal que T_1, T_2, \dots, T_n são os predicados associados respectivamente aos tipos de conceitos T_1, T_2, \dots, T_n ;

- A escrita em SOWL: [INTERSECTIONOF ([ConceptType])*];
- Tradução em cálculo lambda: [Lambda(Entity*x) (sameAs[T₁?x] (sameAs[T₂?x]. . . (sameAs[T_n?x])

A relação *sameAs* que define que dois indivíduos são idênticos será introduzida mais tarde.

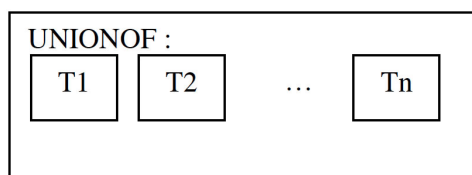
Exemplo: O tipo de conceito Leao:

```
(Def [ConceptType "Leao"] [INTERSECTIONOF : [Leao] [Fêmea]])
```

- A união de tipos

A união será descrita por um contexto de tipo UNIONOF tendo por descritor um conjunto de conceitos únicos tendo por tipo os tipos à unir e um quantificador vazio, como mostrado na Figura 46.

Figura 46 - União de tipos



Fonte: elaborada pela autora

- Fórmula lógica equivalente: $(x \text{ elemento da união}) \Leftrightarrow (T_1(x) \vee T_2(x) \vee \dots \vee T_n(x))$ tal que T_1, T_2, \dots, T_n são os predicados associados respectivamente aos conceitos T_1, T_2, \dots, T_n .
- A escrita em SOWL: `[UNIONOF ([ConceptType])*]`
- A tradução em cálculo lambda: `[lambda (Entity *x) [Either [Or (sameAs [T1] ?x)] [Or (sameAs [T2] ?x)] [Or (sameAs [Tn] ?x)]]]`

O contexto *either* definido entre diferentes grafos conceituais, sua tradução em grafo conceitual simples pode ser obtido pela utilização do contexto da negação.

$$(x \vee y) \Leftrightarrow \neg(\neg x \wedge \neg y)$$

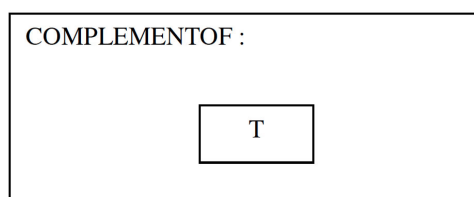
Exemplo: O tipo de conceito Humano é a união dos homens, das mulheres e das crianças.

`UNIONOF[[Homem]][Mulher][Criança]]`

- A complementariedade em relação à um tipo

O complemento será descrito por um contexto de tipo COMPLEMENTOF tendo como descritor uma variável única tendo como tipo o tipo o qual nós queremos definir o complemento, como exemplifica a Figura 47.

Figura 47 - Complemento de tipo



Fonte: elaborada pela autora

- Fórmula lógica equivalente: $(x \text{ elemento do complemento do tipo } T) \Leftrightarrow (\neg T(x))$

Tal que o predicado T corresponde ao tipo de conceito T.

- A escrita em SOWL [COMPLEMENTOF [ConceptType]]
- Tradução em cálculo lambda: [Lambda (Entity *x) [(sameAs ?x [T])]]

A expressão lambda escolherá todos os indivíduos do tipo Entidade, que não são do tipo T.

Exemplo: O tipo Inerte é o complemento do tipo Vivo.

(DEF [Concept Type "Inerte"] [COMPLEMENTOF [Vivo]])

- Declaração de tipos por enumeração

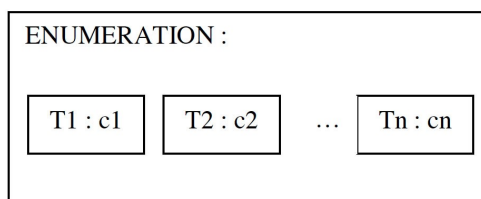
A declaração por enumeração se fará pela especificação dos marcadores de indivíduos do conjunto à construir no preenchedor de um contexto do tipo ENUMERAÇÃO utilizando variáveis únicas contendo os tipos associados aos marcadores e o nome do marcador - omitir um tipo significa que se trata do tipo Entidade (tipo universal), como mostra na Figura 48.

- Fórmula lógica equivalente:

$$(x \text{ elemento da enumeração}) \Leftrightarrow (= (x, c_1) \vee = (x, c_2) \vee \dots \vee = (x, c_n))$$

Onde T_1, T_2, \dots, T_n são as constantes predicativas associadas aos indivíduos marcados pelos marcadores c_1, c_2, \dots, c_n , e o predicado = é o predicado de igualdade.

Figura 48 - Enumeração de tipos



Fonte: elaborada pela autora

- A escrita em SOWL [ENUMERATION:([ConceptType:individualmarker])*]
- Tradução em cálculo lambda [Lambda (Entity *x) [Either [Or (sameAs[T₁:c₁] ?x)] [Or (sameAs[T₂:c₂] ?x)] [Or (sameAs[T_n:c_n] ?x)]]]

Exemplo:

Seja o tipo Estação tendo por indivíduos inverno, primavera, verão e outono.

```
(Def[ConceptType“Estação”][ENUMERATION([Entity:verão][Entity:outono][Entity:primavera][Entity:inverno])])
```

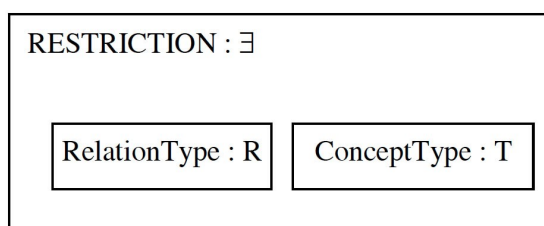
- Restrição de tipos:

A restrição de tipo é definida pelas relações de valência 2, ela se apresenta sob as seguintes formas:

- Existencial

Fórmula lógica equivalente: Se x pertence à restrição existencial $REST$ sobre a relação R em relação ao tipo de conceito C (tipo que deve ser subordinado pelo tipo do segundo argumento da relação) então x satisfaz a seguinte fórmula lógica: $\exists y T(y) \wedge R(x,y)$ (existe e do tipo T tal que x está em relação com ele através de R).

Figura 49 - Restrição existencial



Fonte: elaborada pela autora

Com R sendo o predicado atribuído à relação conceitual R , T o predicado unário atribuído ao tipo de conceito T .

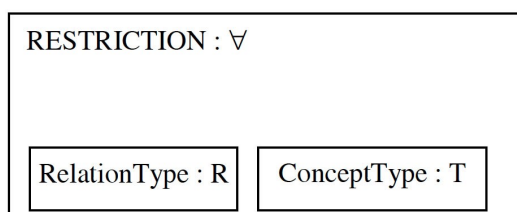
A escrita em SOWL [RESTRICTION:∃ [“RelationType” : RelationType] [“ConceptType” : ConceptType]]

Tradução em cálculo lambda

Será associado a este construtor a seguinte fórmula lambda: [Lambda (Entity *x) (R ?x [T])]

- Universal

Figura 50 - Restrição universal



Fonte: elaborada pela autora

Fórmula lógica equivalente: x pertence à restrição universal $REST$: \forall sobre a relação R e o tipo de conceito T (tipo que deve ser subsumido pelo tipo do segundo argumento da relação) então x satisfaz a seguinte fórmula lógica: $\forall y R(x,y) \Rightarrow T(y)$ (se x é em relação com uma entidade em relação R , ela é do tipo T).

Com R o predicado atribuído à relação R , T o predicado unário atribuído ao tipo de conceito T .

A escrita em SOWL: [RESTRICTION: ∀ [“RelationType”: RelationType] [“ConceptType” : ConceptType]]

Tradução em cálculo lambda

Será associado a este construtor a seguinte fórmula lambda:

[Lambda (Entity *x) [If (R ?x [thing *y]) [Then:(sameAs ?y [T])]]]

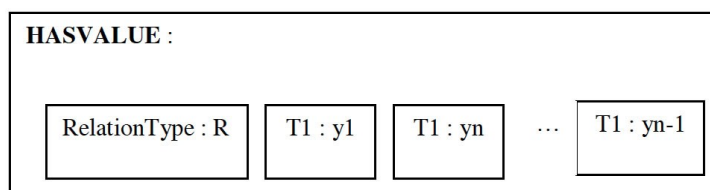
Os contextos aninhados if then permitem expressar a implicação, sua tradução em grafos conceituais simples podem ser obtidos devido ao contexto da negação.

$$X \Rightarrow Y \cong \neg(X \wedge \neg Y)$$

$$[if X[then Y]] \cong \sim[X \sim [Y]]$$

- Restrição de propriedade

Figura 51 - Restrição de propriedade



Fonte: elaborada pela autora

A restrição *HASVALUE* permite construir um conjunto de indivíduos que estão relacionados com os indivíduos y_1, y_2, \dots, y_{n-1} por uma relação R .

Fórmula lógica equivalente: Se x é incluso na restrição *HASVALUE* definida pela propriedade R de valência $n+1$ e de assinatura $(T, T_1, T_2, \dots, T_n)$ e os indivíduos y_1, y_2, \dots, y_n de tipos respectivos T_1, T_2, \dots, T_n então x verifica a seguinte fórmula lógica: $T(x) \wedge T_1(y_1) \wedge T_2(y_2) \wedge T_n(y_n) \wedge R(x, y_1, y_2, \dots, y_n)$

A escrita em SOWL:

```
[HASVALUE [RelationType:"RelationType"]
  [ConceptType:individualmarker]
  [ConceptType:individualmarker]
  (n vezes)
  [ConceptType :individualmarker]]
```

Tradução em cálculo lambda: Este construtor é associado à seguinte fórmula lambda:

$$[\text{Lambda } (T *x) (R ?x [T_1:y_1] [T_2 :y_2]. \dots [T_n :y_n])]$$

Exemplo:

Os homens casados são os homens que estão em relação com uma mulher pela relação casado_com de valência 2 e de assinatura (Homem,Mulher):

(Def [ConceptType Homem_Casado]

[HASVALUE [RelationType casado_com] [mulher : \exists]])

- Restrição de cardinalidade

Serão apresentadas as restrições de cardinalidade para as relações conceituais de valência 2, para isto começamos pela apresentação das seguintes relações:

As relações sup, inf e equal: de valência 2, elas têm como assinatura (%inteiro,%inteiro) elas permitem declarar que um número é superior (para a relação sup), inferior (para a relação inf) ou igual (para a relação igual) em relação à outra.

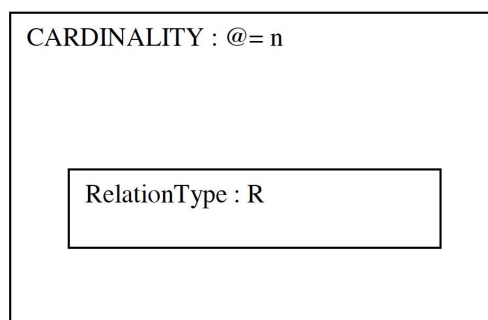
A relação CARD: de valência 2 e de assinatura (ConceptType, %inteiro) ela permite associar à cada tipo de conceito o número de indivíduos distintos podendo ser compatíveis com ele.

- Cardinalidade

Este construtor de cardinalidade permite reagrupar os indivíduos que estão em relação com n entidades pela relação R .

Fórmula lógica equivalente: Dizemos que x pertence à restrição de cardinalidade definida sobre a relação R de valência 2 e de assinatura (T_1, T_2) e o número n significa que ele satisfaz a seguinte fórmula: $T_1(x) \wedge y : T_2(y) \wedge R(x,y) = n$

Figura 52 - Restrição de cardinalidade



Fonte: elaborada pela autora

Escrita em SOWL [CARDINALITY:@=n ["RelationType ": RelationType]

Expressão lambda equivalente [Lambda (T₁ *x) (CARD ([HASVALUE [RelationType R'] ?x] [%entier *z]) (equal ?z [%entier n]))]

Tal que a relação R' é a relação inversa de R (devido o contexto HASVALUE retorna somente o primeiro argumento de uma relação enquanto que neste caso precisamos do segundo).

- Cardinalidade mínima

Este construtor de cardinalidade permite reagrupar os indivíduos que são relacionados com mais de n entidades pelas relações R.

Fórmula lógica equivalente :

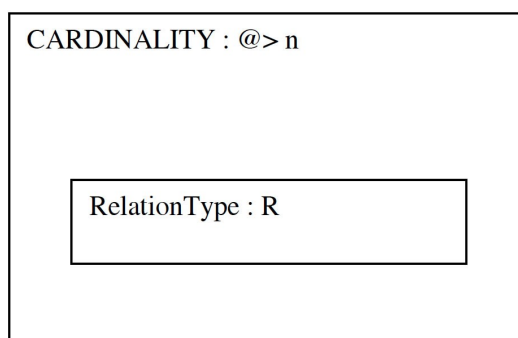
Seja x pertencente à restrição de cardinalidade definida sobre a relação R de valência 2 e de assinatura (T_1, T_2) e o número n significa que ele satisfaz a seguinte fórmula:

$$T_1(x) \wedge y : T_2(y) \wedge R(x,y) > n$$

Escrita em SOWL: [CARDINALITY:@>n ["RelationType": RelationType]]

Expressão lambda equivalente [Lambda (T₁ *x) (CARD ([HASVALUE [RelationType R'] ?x] [%entier *z]) (sup ?z [%entier n]))]

Figura 53 - Restrição de cardinalidade mínima



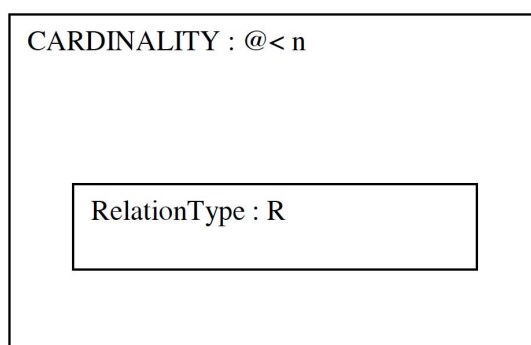
Fonte: elaborada pela autora

Tal que a relação R' é a relação inversa de R (devido o contexto *HASVALUE* retornar somente o primeiro argumento de uma relação enquanto nesse caso precisamos do segundo).

- Cardinalidade máxima

Este construtor de cardinalidade permite reagrupar os indivíduos que são relacionados com menos de n entidades pelas relações R .

Figura 54 - Restrição de cardinalidade máxima



Fonte: elaborada pela autora

Fórmula lógica equivalente: Dizemos que x pertence à restrição de cardinalidade definida sobre a relação R de valência 2 e de assinatura (T_1, T_2) e o número n significa que ele satisfaz a seguinte fórmula: $T_1(x) \wedge y : T_2(y) \wedge R(x,y) < n$

Escrita em SOWL: [CARDINALITY:@>n ["RelationType":RelantionType]]

Expressão lambda equivalente: [Lambda (T₁*x) (CARD ([HASVALUE [RelationType R] ?x] [%inteiro *z]) (inf ?z [%inteiro n]))]

Tal que a relação R' é a relação inversa de R (devido o contexto *HASVALUE* retornar somente o primeiro argumento de uma relação enquanto nesse caso precisamos do segundo).

Exemplo:

Um homem poligâmico é um homem que tem mais de uma mulher:

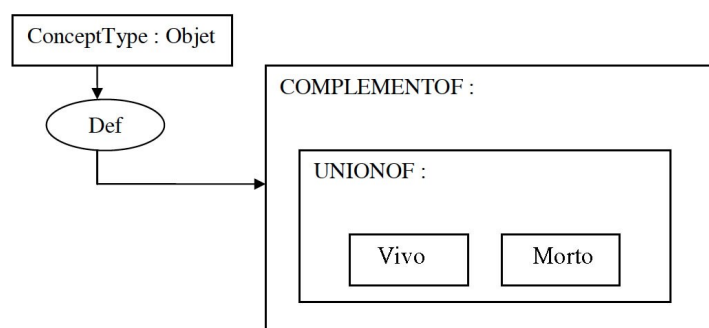
(DEF[ConceptType“HPoli”][CARDINALITY:@>1[RelationType:casado_com]]

Nota:

A combinação destes construtores é assegurada pelo fato que ele possui uma forma em expressão lambda - de valência 1 - que pode ser comparada a algum tipo de conceito.

Exemplo na Figura 55: ‘Toda entidade que não é viva ou morta é um objeto’.

Figura 55 - Exemplo de restrição de cardinalidade



Fonte: elaborada pela autora

(Def Objet [ConceptType] [COMPLEMENTOF [UNIONOF [VIVO] [MORTO]]])

c) As propriedades sobre os tipos (*Typeproperty*)

Além da relação de subordinação, definiremos diferentes esquemas de axiomas de relação e de declaração de propriedades sobre os tipos de conceitos, úteis para aguçar suas definições, ou utilizáveis durante diferentes processos de inferência:

- A incompatibilidade de tipos

Para expressar o fato de dois tipos de conceitos C_1 e C_2 não poderem ser compatíveis introduziremos uma relação *Disjoint* de valência 2, que suporta como parâmetro dois tipos de conceitos.

Sua forma em SOWL é:

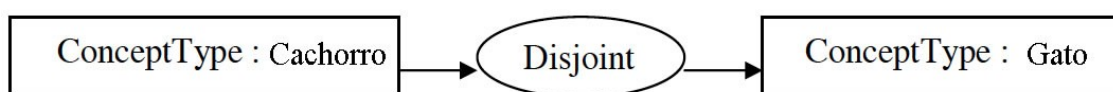
```
“(” “DISJOINT”
 “[” “TypeLabel” “” “ConceptType” “” “]”
 “[” “TypeLabel” “” “ConceptType” “” “]” “)”
```

Sua semântica pode ser expressa pela seguinte fórmula predicativa:

$$(C_1 \text{ é separada de } C_2) \Leftrightarrow (\forall x(C_1(x) \Rightarrow \neg C_2(x)) \wedge C_2(x) \Rightarrow \neg C_1(x))$$

Exemplo na Figura 56:

Figura 56 - Exemplo de incompatibilidade de tipos



Fonte: elaborada pela autora

- Abstração de tipos

Um tipo rotulado como abstrato não pode ter instância direta, tal tipo pode ser utilizado para a organização da hierarquia dos tipos de conceitos:

```
(ABSTRACT [ “ConceptType” “” “ConceptType” ])
```

Para declarar que o conceito *Animal* é abstrato:

```
(ABSTRACT [ConceptType “Animal”])
```

4.1.1.3.2 Hierarquia de tipos de relações (*RelationHierarchy*)

A definição de um tipo de relação se fará através de sua incorporação na hierarquia de relações (hierarquia de subordinações), a especificação de sua valência n (seu número de argumentos), e sua relação com uma definição de cálculo lambda (definindo ao mesmo tempo sua assinatura), tudo isso se faz no contexto *RelationHierarchy*:

```
“[” “RelationHierarchy”
(RelationLabelOrdering| ValenceSpec |RelationDefinition|RelationProprety)* “]”
```

A hierarquia de relações é construída devido a um conjunto de *RelationLabelOrdering* que assume as seguintes formas:

```
“(” (“EQ” | “GT” | “LT”)
“[” “RelationType” “” “RelTypeLabel” “” “]”
“[” “RelationType” “” “RelTypeLabel” “” “]” “)”
```

A relação *EQ* é a relação de equivalência entre tipos de relações, *GT* é a subordinação quanto à *LT*, ela representa a especialização de tipos de relações.

A definição da valência se faz devido a relação *Has* da seguinte maneira: “(” “Has”

```
“(“ “[” “Relationtype” “” “RelationType” “” “]” “[” “Valência” Inteiro “]” “)”
```

Quanto à definição da relação através de sua expressão lambda (*RelationDefinition*), ela se faz devido a relação *Def*:

```
“(” “Def”
“[” “Relationtype” “” “RelationType” “” “]”
“[” “Lambda” “” “LambdaExpression” “” “]” “)”
```

Serão dados exemplos de utilização de hierarquias de relações após ser apresentado o rótulo das relações.

Da mesma forma para os tipos de conceitos será definido um construtor para as relações inversas e sua tradução em lambda cálculo.

a) Relação inversa

Para introduzir o construtor de relações inversas, será introduzido primeiramente a semântica que desejamos atribuir.

Sejam duas relações R, R' de valência m , e de assinaturas respectivas $(T_1, T_2, \dots, T_m, T_{m+1}, \dots, T_n)$ e $(T_{m+1}, T_{m+2}, \dots, T_n, T_1, \dots, T_m)$.

Diremos que uma relação R' é a inversa de uma outra R em relação ao Vetor de dimensão m sobre seus m primeiros argumentos se ela satisfaz a seguinte fórmula predicativa:

$$\forall x_1, x_2, \dots, x_m, x_{m+1}, \dots, x_n (T_1(x_1) \wedge T_2(x_2) \wedge \dots \wedge T_m(x_m) \wedge T_{m+1}(x_{m+1}) \wedge \dots, T_n(x_n) \wedge R(x_1, x_2, \dots, x_m, x_{m+1}, \dots, x_n) \Rightarrow R'(x_{m+1}, x_{m+2}, \dots, x_n, x_1, \dots, x_m))$$

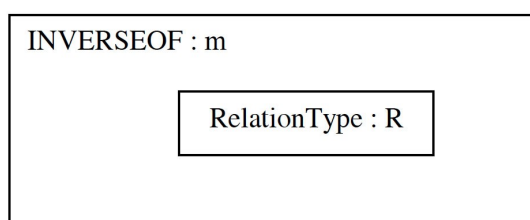
Tal que T_1, T_2, \dots, T_n são os predicados correspondentes respectivamente aos tipos de conceitos T_1, T_2, \dots, T_n . R e R' são os predicados n -ários correspondentes à relação conceitual R e sua relação inversa R' .

Será expresso através do contexto *INVERSEOF* que conterà em seu referente um inteiro m especificando o tamanho do vetor de inversão, ele terá em seu descritor uma variável única de tipo *RelationType* - nome da relação - que especificará o nome da relação à inverter (onde a valência deve ser estritamente superior à m), como mostra a Figura 57:

A escrita em SOWL correspondente é:

INVERSEOF : inteiro ["RelationType" "RelationType"]

Figura 57 - Relação inversa



Fonte: elaborada pela autora

Fórmula lambda correspondente:

A expressão lambda correspondente à inversa de uma relação R tendo por assinatura (T_1, T_2, \dots, T_n) para o parâmetro m descrito por este contexto é:

[Lambda ([T_{m+1} *x_{m+1}] [T_{m+2} *x_{m+2}] . . . [T_n *x_n] [T₁ *x₁] [T₂ *x₂] . . . [T_m *x_m]) (R ?x₁ ?x₂ . . . ?x_n)]

b) Propriedades de relações

Também forneceremos a possibilidade de associar aos tipos de relações das propriedades, definindo melhor sua semântica e especificidades e isto devido aos esquemas de axiomas que serão especificados por um sistema de rotulação na hierarquia de subordinação na forma gráfica, e serão implementados em SOWL como as *RelationProperty* assim definidas:

“((“SYM” | “FONC@m” | “TRANS”) “[Reliointype” “” “RelationType” “” “]” “)”

Rótulo (*SYM*) para as relações simétricas: Definido pela seguinte semântica: Se uma relação R de valência 2_n (n inteiro positivo) e de assinatura $(T_1, T_2, T_m, T_1, T_2, \dots, T_n)$ é dita simétrica então:

$$\forall x_1, x_2, \dots, x_n, x_{n+1}, \dots, x_{n+2}, \dots, x_{2n} T(x_1) \wedge T(x_2) \wedge \dots \wedge T(x_n) \wedge T(x_{n+1}) \wedge T(x_{n+2}) \wedge \dots \wedge T(x_{2n}) \\ (R(x_1, x_2, \dots, x_n, x_{n+1}, x_{n+2}, \dots, x_{2n}) R(x_{n+1}, x_{n+2}, \dots, x_{2n}, x_1, x_2, \dots, x_n))$$

R é o predicado correspondente à relação conceitual R , e T_1, T_2, \dots, T_n corresponde respectivamente aos tipos de conceitos T_1, T_2, \dots, T_n .

Rótulo (*FONC@m*) para as relações funcionais:

Definido através da seguinte semântica: Se uma relação R de valência n (n inteiro positivo) e de assinatura (T_1, T_2, \dots, T_n) é rotulada funcional de parâmetro m (m inteiro estritamente positivo, estritamente inferior à n) então:

$$\forall x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_{n-m} T_1(x_1) \wedge T_2(x_2) \wedge \dots \wedge T_n(x_n) \wedge T_{m+1}(y_1) \wedge T_{m+2}(y_2) \wedge \dots \wedge T_n(y_{n-m}) \\ \wedge R(x_1, x_2, \dots, x_n) \wedge R(x_1, x_2, \dots, x_m, y_1, y_2, \dots, y_{n-m}) \Rightarrow (= (x_{m+1}, y_1) \wedge = (x_{m+2}, y_2) \wedge \dots \wedge = (x_n, y_{n-m}))$$

R é o predicado correspondente à relação conceitual R , e T_1, T_2, \dots, T_n correspondem respectivamente aos tipos de conceitos T_1, T_2, \dots, T_n .

Notas:

- * Se m é omitido seu valor por definição será igual à $n-1$;
- * Para as relações de valência 2, encontramos a mesma noção de relação funcional definida em OWL.

Rótulo (*TRANS*) para as relações transitivas:

Definida pela seguinte semântica: Se uma relação R de valência $2n$ (n inteiro positivo) e de assinatura $(T_1, T_2, \dots, T_n, T_1, T_2, \dots, T_n)$ é rotulada transitiva então

$$x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n, z_1, z_2, \dots, z_n \quad T_1(x_1) \wedge T_2(x_2) \wedge \dots \wedge T_n(x_n) \wedge T_1(y_1) \wedge T_2(y_2) \\ \wedge \dots \wedge T_n(y_n) \wedge T_1(z_1) \wedge T_2(z_2) \wedge \dots \wedge T_n(z_n) \wedge (R(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n) \wedge R(y_1, y_2, \dots, y_n, z_1, \\ z_2, \dots, z_n)) \Rightarrow R(x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_n)$$

R é o predicado correspondente à relação conceitual R , e T_1, T_2, \dots, T_n correspondem respectivamente aos tipos de conceitos T_1, T_2, \dots, T_n . Exemplo sobre a etiquetagem em uma hierarquia de relações, mostrado na Figura 58:

Em SOWL (A hierarquia de tipos, de valência, a definição depois da rotulagem de relações):

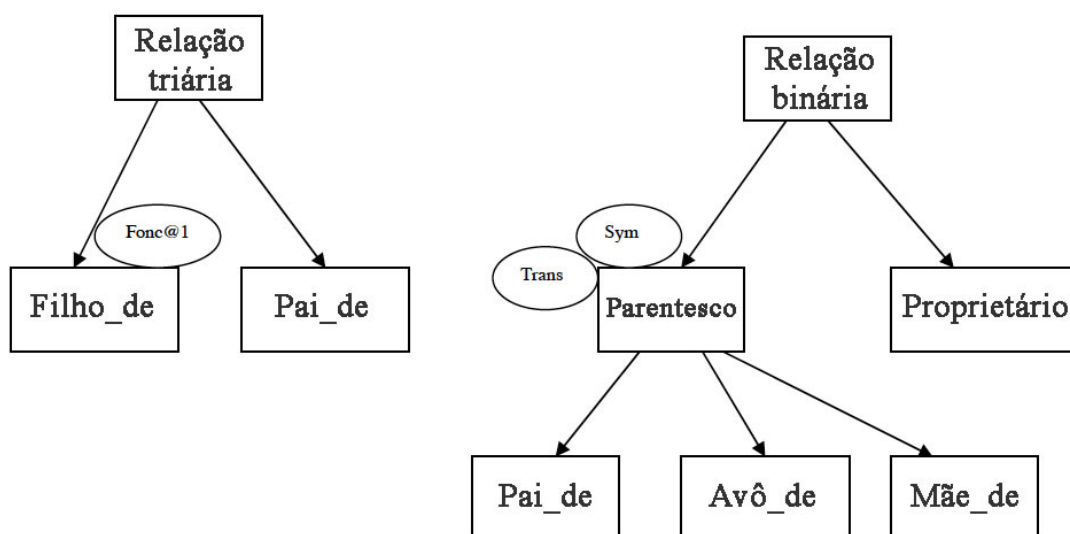
(GT [RelationType “Relacao_binaria”][RelationType “Parentesco”])

(GT [RelationType “Relacao_binaria”][RelationType “proprietario”])

(GT [RelationType “Relacao_triaria”][RelationType “Filho_de”])

(GT [RelationType “Relacao_triaria”][RelationType “Pais_de”])

Figura 58 – Etiquetagem em uma hierarquia de relações



Fonte: elaborada pela autora

(GT [RelationType "Parentesco"] [RelationType "Pai"])

(GT [RelationType "Parentesco"] [RelationType "Avo"])

(GT [RelationType "Parentesco"] [RelationType "Mae"])

(Has [Relationtype "Relacaobinaria"] [Valencia 2])

(Has [Relationtype "Relacaotriaria"] [Valencia 3])

(Has [Relationtype "Filho_de"] [Valencia 3])

(Has [Relationtype "Pais_de"] [Valencia 3])

(Has [Relationtype "Parentesco"] [Valencia 2])

(Has [Relationtype "Proprietario"] [Valencia 2])

(Has [Relationtype "Pai_de"] [Valencia 2])

(Has [Relationtype "Avo_de"] [Valencia 2])

(Has [Relationtype "Mae_de"] [Valencia 2])

(Def [RelationType "avo"] [lambda(Humano*x, Humano*y)

(Pai_de ?x, [Humano *z])]

(Pai_de ?z, ?y)])

(Def [Relationtype "Filho_de"] [lambda(Filho *x₁, Homem *x₂, Mulher *x₃)

(Pai_de ?x₂ ?x₁)

(mae_de ?x₃ ?x₁))

(DEF [Relationtype "Pai_de"] [INVERSEOF:1 [Relationtype "Filho_de"]])

(SYM [Relationtype "Parentesco"])

(TRANS[Relationtype "Parentesco"])

(Fonc@1[Relationtype "Filho_de"]) "Para um filho, existe somente um casal de pais"

4.1.2 ABOX → Catálogo de indivíduos e a base de fatos

Uma das particularidades da abordagem dos GC's em relação à das DL's é a separação da declaração de indivíduos (pertencendo à uma classe de conceitos ou relações lhes definindo em relação às outras entidades manipuladas) de afirmações gerais, aumentando assim a modularidade dos conhecimentos permitindo uma melhor definição e uma reusabilidade aumentada.

4.1.2.1 Catálogo de indivíduos

Será declarado aqui todos os indivíduos manipulados na base de conhecimentos e isto especificando para cada indivíduo as informações tais como:

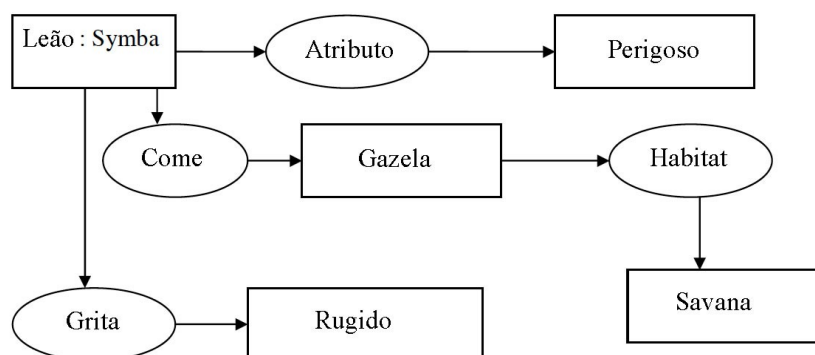
Seu nome.

Os tipos com os quais ele é compatível.

Os grafos conceituais que definem seus atributos.

Exemplo na Figura 59:

Figura 59 - Exemplo de base de conhecimento



Fonte: elaborada pela autora

Em SOWL:

```

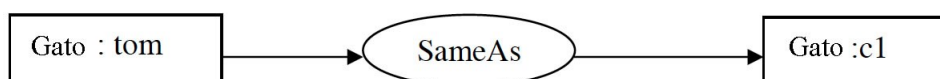
[Leao:Symba *x]
(Come ?x[Gazela*y])
(Grita ?x[Rugido])
(Atributo ?x [Perigoso])
(Habitat ?y [Savana])
  
```

Vamos agora declarar alguns esquemas de axiomas úteis para especificar alguns atributos recorrentes na declaração de indivíduos:

1. *SameAs* Para não confundir com a noção de correferência, que permite substituir os nós de um conjunto de co-referência por seu nó dominante (uma relação de uma só direção), a relação *SameAs* permite declarar que dois indivíduos são idênticos se são permutáveis entre si.

SameAs será declarado como sendo uma relação de valência 2 e de assinatura (indivíduo, indivíduo), ela poderá ser associada ao predicado de igualdade da lógica predicativa “=”.

Exemplo na Figura 60:

Figura 60 - Exemplo de esquema de axioma *SameAs*

Fonte: elaborada pela autora

2. *DifferentFrom* A relação complementar de *SameAs*, poderá ser utilizada por exemplo, para definir que dois marcadores não podem apontar o mesmo indivíduo.

Exemplo na figura Figura 61:

Figura 61 - Exemplo de esquema de axioma *DifferentFrom*



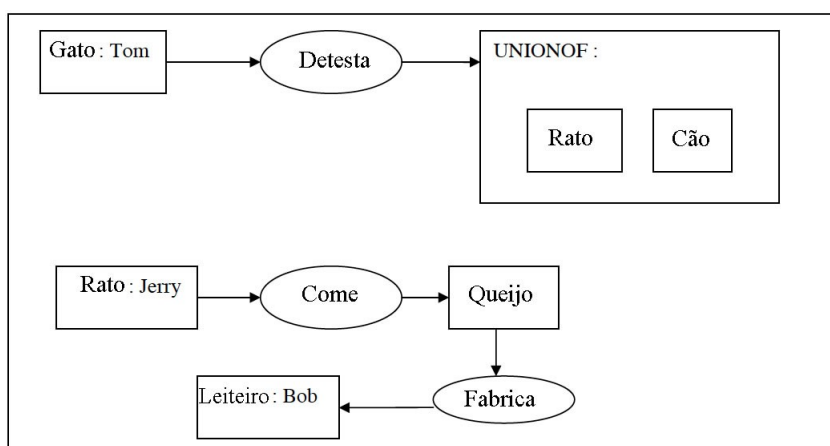
Fonte: elaborada pela autora

4.1.2.2 Base de fatos

A base de fatos é composta de contextos do tipo asserção, contendo os GC's definidores dos fatos gerais, as situações e outros conhecimentos, e isto utilizando as primitivas conceituais descritas pelo suporte:

Exemplo na figura Figura 62:

Figura 62 - Exemplo de base de fatos



Fonte: elaborada pela autora

Em SOWL:

(Detesta [Gato : tom] [UNIONOF : [Rato][Cao]])

(Come [Rato : Jerry] [Queijo *x])

(Fabrica ?x [Leiteiro : Bob]).

4.1.3 Informações Operacionais

As informações operacionais são os dados de suporte, elas servem para operacionalizar a linguagem e é frequentemente usada para adornar os conhecimentos necessários à sua utilização nos sistemas práticos, no que vai se seguir, tentaremos lhes apresentar.

4.1.3.1 A utilização de tipos de dados predefinidos

A utilização dos tipos de dados predefinidos é simples nos GC's, ela tem a seguinte forma:

[%tipo_dos_dados valor_dos_dados]

Os tipos de dados que os grafos conceituais poderiam suportar (de acordo com Sowa) variam dos tipos de dados clássicos, encontrados nas linguagens de representação e de programação (inteiro, cadeia de caracteres, real, booleano. . . etc) às URI's que definem os arquivos (mp3,

avi, gif. . . etc.) os quais poderemos associar as aplicações apropriadas.

Assim, a declaração de relações à assinatura incluindo um tipo de dados predefinido é possível, é necessário somente proceder o tipo por um “%”.

Exemplo:

Relação IDADE

```
(DEF[RelationType:"idade"] [Lambda (humano *x, %inteiro_positivo *y)... ] )
```

Além disso, é possível definir as estruturas de dados mais complexos utilizando a seguinte notação:

```
("%" Identificador_da_estrutura)? "{" (Conceito|correferente)* "}"
```

Exemplo: estrutura equivalente à uma anotação sobre os livros:

```
(%anotacao_livro) ? (Autor) (Edicao) (%preco) (%ISBN) (%String_Titre_Original)
```

```
[Livro : Os_Miseraveis : % anotacao_livre ? [Autor:Victor_hugo]
                                [Edicao:Eclipse]
                                [%prix 100]
                                [%ISBN "01248752145"]
                                [%String_Titulo_Original "Os Miseraveis" } ]
```

4.1.3.2 Anotações e importação de conhecimentos

Para tornar essa linguagem operacional é necessário também definir vários mecanismos de anotação, de nível de formalismos e de normalizações diversas, e isto para expressar:

4.1.3.2.1 Os comentários

Será adotada a seguinte notação para a definição de comentários: uma cadeia de caracteres entre dois “;” (o ‘;’ será expresso por “;” no comentário).

Será definido também uma relação “Anot” de valência 2, tomando como assinatura (Thing, %string) que permitirá associar aos nós conceitos (ou contexto) uma mensagem ou uma definição em linguagem natural sob forma de uma cadeia de caracteres.

Exemplo na Figura 63:

[Gato ; Isto é um comentário ; *x]

(Anot [?x] [%string “Isto é um comentário ligado ao conceito gato.”])

Figura 63 - Exemplo de comentário



Fonte: elaborada pela autora

4.1.3.2.2 A definição e a importação de conceitos e de relações externas

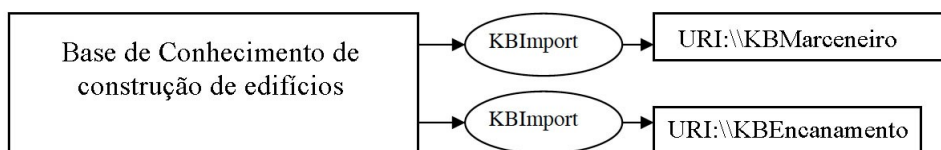
Para levar em conta a natureza distribuída da Web, os mecanismos de importação de componentes ou de bases de conhecimentos inteiros deverão ser previstos, para isto, será especificado as relações conceituais subordinadas da relação “Anot” que dedicamos para importar as informações:

a) A relação KBImport

De valência 2 e de assinatura (*KBcontext*, URI), ela permitirá a importação de bases de conhecimentos ligando-os aos definidos: (Import [KB] [URI]).

Exemplo na Figura 64:

Figura 64 - Exemplo da relação KBImport



Fonte: elaborada pela autora

Será introduzido assim um contexto especial chamado *foreignDef*, permitindo apontar uma URI, uma definição em cálculo lambda permitindo assim uma definição estrangeira das primitivas terminológicas.

(Def ["ConceptType" ConceptType][foreignDef : URI]

Exemplo:

(Def ["ConceptType" Tigre] [foreignDef : HTTP: Ontologia_do_mundo_animal.edu]

b) O controle de versões da base de conhecimentos

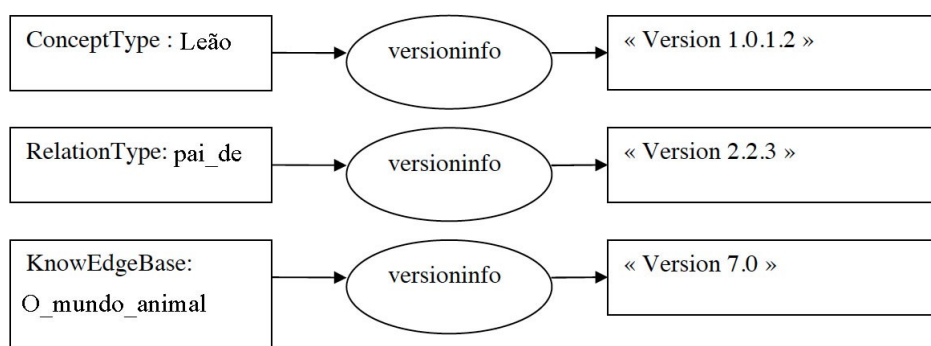
Serão definidas várias relações subordinadas da relação "*Anot*" para especificar as informações relacionadas à atualização e a compatibilidade dos conhecimentos reorganizados e isto através da especificação das versões da *KB*:

- Versão_info:

Toma como parâmetros (conceito, String_version) e permite especificar a versão de qualquer componente da base (ou a base ela mesma).

Exemplo na Figura 65:

Figura 65 - Exemplo de controle de versão

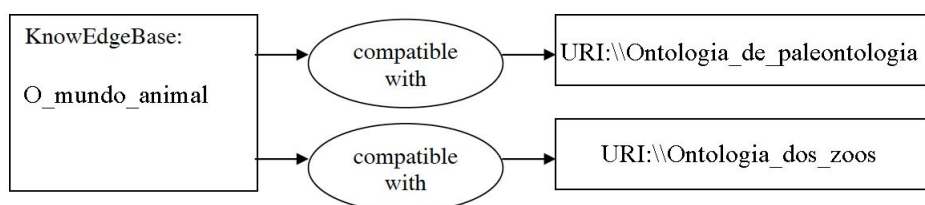


Fonte: elaborada pela autora

- *Compatible_with*:

Toma como parâmetros (base de conhecimentos, URI_de_base_de_conhecimento) ela permite definir o conjunto de bases compatíveis com aquela especificada permitindo uma melhor combinação do conjunto por seus conceituadores de aplicação (ou porque não uma composição automática dos conhecimentos).

Exemplo na Figura 66:

Figura 66 - Exemplo de *Compatible_with*

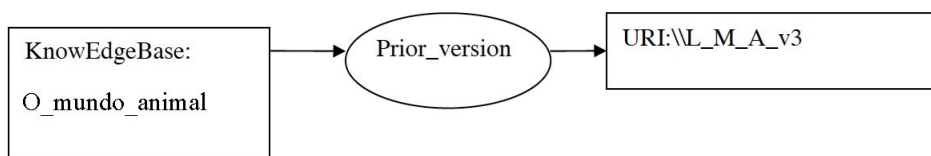
Fonte: elaborada pela autora

- *Prior_version*

Esta relação que tem como assinatura (base de conhecimento, URI_de_base_de_conhecimento) permite ligar a base de conhecimentos à suas versões precedentes, e isto devido ao fato de acompanhar a evolução dos conhecimentos.

Exemplo na Figura 67:

Figura 67 - Exemplo de Prior_version



Fonte: elaborada pela autora

4.1.4 Comparação e síntese

Como foi visto, no começo desse capítulo, que a linguagem OWL foi certamente a linguagem principal para a especificação das ontologias mas que ainda está, entretanto, sujeita à muitas críticas. Foi proposto no presente trabalho um modelo conceitual que incorporasse as funcionalidades do OWL se baseando para isto sobre o formalismo dos grafos conceituais como substituto das lógicas descritivas; a finalidade de tudo isso é naturalmente estudar a participação e a contribuição que tal trabalho poderia trazer para a comunidade de desenvolvimento no âmbito da Web Semântica.

Neste contexto, é importante notar que o modelo proposto garante perfeitamente o tratamento (criação e manipulação de ontologias) fornecido por OWL e, portanto, qualquer ontologia OWL poderá ser transcrita e manipulada na linguagem apresentada no presente trabalho.

Além disso, esta modificação beneficia através das vantagens atribuídas pelos grafos conceituais, inclusive aqueles ligados às possibilidades de inferências e de resolução de problemas sob uma plataforma baseada sobre os GC's, mas também sobre o plano estrutural da linguagem onde já podemos citar algumas diferenças primordiais entre os dois formalismos:

- O formalismo dos GC's introduziu à nova linguagem a noção de variáveis e isto através das variáveis lambda que caracterizam os grafos conceituais.

- Os GC's atribuem a possibilidade de representar os conhecimentos até mesmo os muito difíceis de expressar em OWL, de fato, é possível no presente criar as ontologias capazes de incluir os conhecimentos modais, os conhecimentos que pertencem às lógicas de ordem superior à *I* ou ainda os conhecimentos de lógica difusa e isto devido - entre outras - à noção dos contextos aninhados oferecidos pelos grafos conceituais.
- Foi mencionado o problema da sobrecarga do A-Box quando o número de indivíduos na base se torna grande, isto devido ao fato que nas lógicas descritivas, um indivíduo é conhecido depois de todas as relações que ele mantém na ontologia, isto significa que para definir um determinado indivíduo, é necessário procurar todas as afirmações onde ele aparece; naturalmente este problema não se coloca dentro do problema da linguagem baseada sobre os grafos conceituais porque os indivíduos são definidos de outro modo nesse formalismo e são reconhecidos com base em suas posições no catálogo de indivíduos e isso de maneira agrupada e concisa (devido a separação dos fatos gerais dos que são usados na definição de indivíduos).

Além disso, foi observado que até o presente, e sempre devido ao formalismo proposto pelos GC's, é possível considerar algumas formas de raciocínio que não eram aplicáveis no contexto da OWL onde os paradigmas de inferência eram fortemente focados na lógica de primeira ordem, estes diferentes pontos serão tratados na segunda parte da concepção.

4.2 Resultado 2 - Mecanismos de Inferência

Foi mencionado anteriormente que ao projetar uma nova linguagem de representação ou mais geralmente de um sistema à base de conhecimentos qualquer, que é imperativo não se limitar ao aspecto estrutural do formalismo de representação dos conhecimentos e estudar as afinidades e as incompatibilidades de tal sistema com as diferentes técnicas de inferências existentes - ou para inovar - porque são em grande parte esses últimos que destacam o poder do aspecto semântico e operacional que possuem esses sistemas. A consequência é que a criação - ou pelo menos a adaptação - de motores de inferência operando em tais plataformas é tão importante quanto o próprio desenvolvimento.

Hoje em dia, a maioria dos motores de inferência são baseados na dedução, mas apesar de suas qualidades inegáveis (domínio no ponto de vista teórico, elaboração de técnicas de otimização, mecanismos de prova. . . etc) esta técnica de raciocínio não se revela nem mais nem menos que uma simples combinação de dados e se revela muitas vezes ineficaz em frente a diversas aplicações (ou os conhecimentos não são claramente ou suficientemente definidos). Desse modo, o estudo de soluções de exportação baseados sobre outras abordagens podem ser menos formais - que a abordagem dedutiva - poderia se revelar crucial no processo de evolução dos motores de inferência clássicos para o reais “resolvedores de problemas” imperativos na abordagem de automatização que defende a Web Semântica.

No que segue, será mostrado como o raciocínio dedutivo poderá ser aplicado aos grafos conceituais, encadaremos em seguida para o estudo da aplicabilidade duas formas de raciocínio que já havia sido apresentado no capítulo precedente, que são:

- O raciocínio por analogia: Aplicado nos domínios onde a modernização dos conhecimentos é difícil e/ou cara, ele permite a resolução dos problemas através de sua comparação com outros já resolvidos ou pelo menos anteriormente encontradas.
- O raciocínio por abdução: Abordagem original de raciocínio, ela consiste na geração de hipóteses supostamente verdadeiras e que poderiam então ser verificadas através de procedimentos de validação, e permite um ganho de tempo apreciável nos sistemas evolutivos (onde os conhecimentos não são fixados ou completamente definidos).

Isto será feito então pela apresentação de algumas soluções que podem ser colocadas em jogo na elaboração de plataformas inferenciais baseadas sobre esses três raciocínios (e utilizando os grafos conceituais como linguagem de operacionalização).

4.2.1 O raciocínio dedutivo

A implementação de um suporte dedutivo - plataforma de manipulação, de tratamento e de combinação através de procedimentos dedutivos - para os grafos conceituais é uma condição crucial para sua aceitação como formalismo de operacionalização dos conhecimentos da Web Semântica, isto é devido a atual arquitetura inferencial – motores de inferências existentes - principalmente dedutivo.

A implementação de tais manipulações poderia ser através de duas abordagens: a primeira será a tradução pura e simples dos conhecimentos expressos em grafos conceituais para a lógica de predicados, solução de curto prazo para o estabelecimento rápido de soluções de manipulação assim como a reutilização daqueles já existentes; a segunda abordagem será a aplicação de uma “arquitetura inferencial” com base na recuperação de manipulações dedutiva através das primitivas de manipulação dos grafos conceituais e às tecnologias anexadas (a teoria dos grafos, o cálculo lambda, etc.)

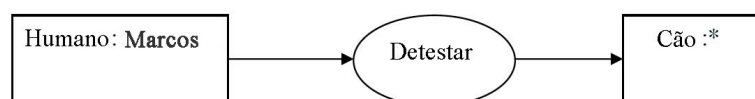
4.2.1.1 A tradução em lógica de predicados

Como foi apresentado no capítulo 2, os grafos conceituais (devido a sua base teórica) oferecem numerosas possibilidades de tradução para outros formalismos, pelo fato de que numerosos algoritmos de tradução para a lógica de predicados foram desenvolvidos, citamos como exemplo o seguinte algoritmo:

1. Associar à cada conceito genérico (Conceito: *) uma variável.
2. Associar à cada elemento de T_c um predicado unário.
3. Associar à cada elemento de T_r de N argumentos um predicado N-ário.
4. Associar à cada marcador do suporte uma constante da linguagem dos predicados.
5. Para cada grafo G , é associada a fórmula F tal que F é o fechamento existencial da conjunção da tradução de todos os nós de G .

Exemplo na Figura 68 – “Marcos detesta os cães”:

Figura 68 - Exemplo de tradução em lógica de predicados



Fonte: elaborada pela autora

$\exists x (\text{humanos}(\text{Marcos}) \vee \text{Cao}(x) \vee \text{detestar}(\text{Marcos}, x))$

Tal transformação permitirá a implantação de tratamentos, fazendo intervir os motores de inferência definidos sobre a base da lógica de predicados, permitindo assim uma operacionalização rápida - e menos cara - dos conhecimentos em um primeiro tempo e isto pela reutilização de plataformas dedutivas já existentes.

O inconveniente é que esta abordagem não permite compreender o interesse de uma operacionalização baseada sobre o formalismo dos grafos conceituais, porque ela não utiliza as diferentes propriedades - com exceção de sua traduzibilidade - e tecnologias que ele implica.

4.2.1.2 A manipulação dedutiva direta

Os grafos conceituais oferecem várias técnicas de representação e primitivas de manipulação podendo ser utilizadas durante a elaboração de tratamentos baseados sobre a inferência dedutiva onde aqui os principais ramos:

4.2.1.2.1 A representação dos conhecimentos dedutivos através dos grafos bipartidos

A representação dos conhecimentos postos em jogo no processo de inferência dedutiva (hipóteses, conhecimentos verificados, objetivos) sob forma de grafos conceituais alocará a possibilidade de lhes manipular através de diversas sintaxes nomeadamente a sintaxe gráfica, útil na interação homem/máquina, facilitando dessa forma a especificação dos conhecimentos e das preferências do utilizador e lhe oferecendo uma melhor ergonomia nos processos de validação e de monitoramento do raciocínio (os processos de inferência e de validação permanecem explícitos).

Por outro lado, os grafos conceituais oferecem várias outras sintaxes destinadas à manipulação e à troca de conhecimentos através dos sistemas automatizados. Permitindo assim o estabelecimento de mecanismos de importação de conhecimentos simples, da mesma forma os tratamentos sequenciais sobre os conhecimentos transmitidos na forma de linguagens de consulta. Além de oferecer a possibilidade de começar os tratamentos sobre os conhecimentos antes mesmo que estes tenham sido inteiramente transmitidos.

Exemplo:

Seja o caso da recepção do conhecimento: “Tom voa de Paris à Alger que é uma capital.”

Este conhecimento é descrito pelo seguinte grafo conceitual:

(Voar [Humano : Tom] [Cidade : Paris] [Capital : Alger])

Após a recepção da porção “Voar”, a plataforma de recepção poderá determinar que se trata de uma relação; utilizando as informações de suporte, será possível determinar sua valência - 3 parâmetros - e através de sua definição em cálculo lambda, será possível definir sua assinatura (humano, Locação, Locação), outras informações poderão ser verificadas à medida que elas atinjam o sistema, como por exemplo:

- A compatibilidade dos tipos de conceitos resulta no grafo (Humano, cidade e capital) com os indivíduos (respectivamente Tom, Paris e Alger) e isto através das informações do catálogo de indivíduos e dos algoritmos de cálculo de compatibilidade de tipos.
- Devido a hierarquia de conceitos, é possível também deduzir que os tipos Cidade e Capital são especializações do tipo de conceito Locação e então conforme a assinatura da relação Voar validando dessa forma o grafo.

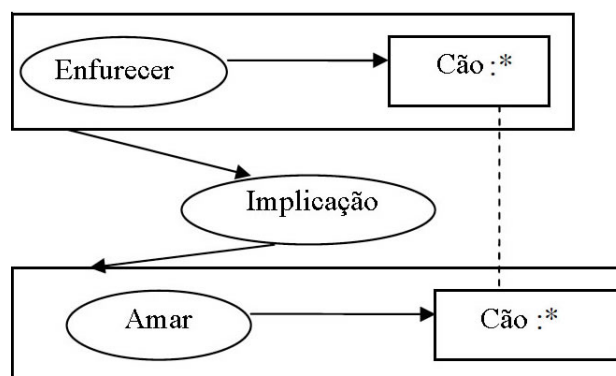
4.2.1.2.2 A utilização do suporte para o armazenamento e acesso aos axiomas

O suporte é uma ferramenta muito poderosa para a organização e a gestão da base de conhecimentos. O isolamento dos conhecimentos gerais do domínio de tais especialidades permitem a elaboração das técnicas de validação e de testes de consistência mais simples. Esta abordagem oferece também uma modularidade no contexto da inferência dedutiva porque ela permitiria a consideração dos axiomas do domínio estudado, através do isolamento da concepção do motor de inferência (tornando-o adequado para o tratamento de domínios muito diferentes).

Exemplo na Figura 69:

Se definimos em nossa base, que um cão raivoso é necessariamente perigoso:

Figura 69 - Exemplo de base de conhecimento



Fonte: elaborada pela autora

Esta afirmação poderá facilmente ser definida em um motor de inferência como sendo uma regra de dedução.

Nota-se que, além dos axiomas definidos nas bases de conhecimentos, os grafos conceituais definem vários axiomas incorporados no formalismo em si mesmo, tal como a subordinação de tipos que permitem afirmar a pertença de um indivíduo a um tipo de conceito sob hipótese que este último seja subordinado por um outro que é compatível com este indivíduo.

Exemplo:

A presença no suporte dos dois conhecimentos:

(GT [ConceptType "Animal"] [ConceptType "Gato"])

e

(GT [RelationType "Parentabilidade"] [RelationType "Pai_de"])

Permite definir as duas seguintes regras de dedução:

Se um indivíduo é um Gato, então ele é também Animal. Se x é o pai de y então x é um parente de y .

4.2.1.2.3 A implementação das regras de dedução e a operacionalização dos axiomas

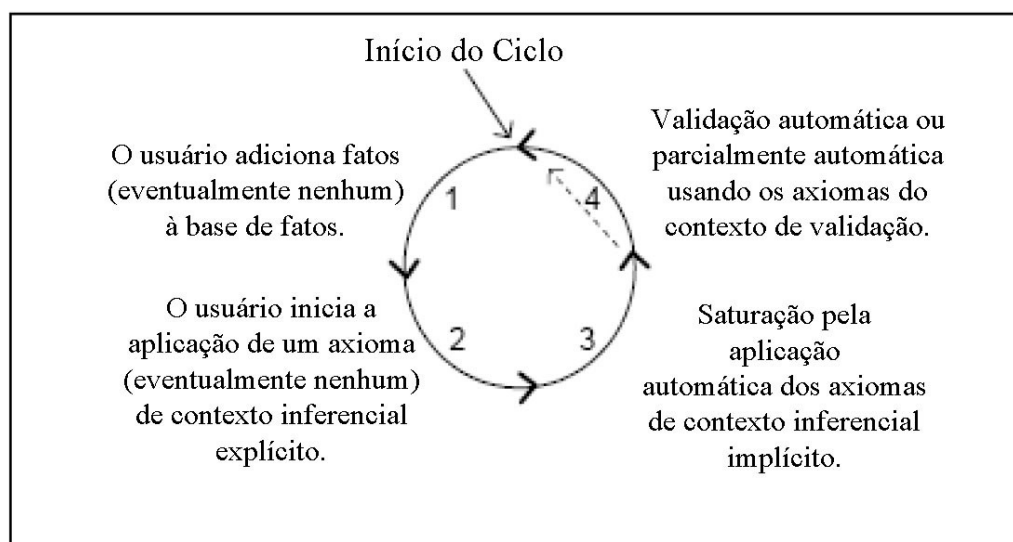
A implementação das regras de inferência e operacionalização dos axiomas, se fará como já foi mostrado no capítulo 2 pela especificação da semântica e do papel dos diferentes

componentes no processo de dedução e isto sendo:

1. **Restrições positivas:** com um grafo hipótese e um grafo conclusão onde a semântica é “Se a hipótese está presente em um grafo, a conclusão deve estar presente no grafo (senão a restrição não é respeitada e o grafo não é válido no domínio modelado)”.
2. **Restrições negativas:** com um grafo hipótese e um grafo conclusão onde a semântica é “Se a hipótese está presente em um grafo, a conclusão deve estar ausente do grafo (senão a restrição não é respeitada e o grafo não é válido no domínio).”
3. **As regras de inferências:** com um grafo hipótese e um grafo conclusão onde a semântica é “Se a hipótese está presente em um grafo, a conclusão pode ser adicionada ao grafo”.

A utilização destes três tipos de conhecimentos pelos motores de inferência pode atingir um certo grau de automatização desobrigando o usuário durante o processo de operacionalização e de raciocínio; sua aplicação seguirá o ciclo de resolução mostrado na Figura 70:

Figura 70 - Ciclo de resolução



Fonte: elaborada pela autora

4.2.1.2.4 O cálculo lambda para a validação e expansão das proposições

A utilização do cálculo lambda para a especificação de tipos (conceitos ou relações) oferece um bom número de vantagens, cita-se como exemplo:

a) A avaliação dos grafos:

A avaliação das proposições descritas em grafos conceituais pode ser feita facilmente através da substituição das constantes lambda das descrições dos tipos de conceitos e das relações do grafo para os argumentos que estes últimos levam, permitindo assim:

- Definir o valor do grafo, ou ao menos de uma parte do grafo (simplificando desta forma sua validação).
- Definir as equações de tipos para as variáveis e as constantes utilizadas no grafo conceitual e os confrontar aos conhecimentos descritos no suporte (para eventualmente detectar uma anomalia, validar um fato dado ou mesmo através das descrições do catálogo definir as equivalências etc.).

Exemplo:

Se obtem-se a conclusão durante um procedimento de dedução que um paciente tem tanto hipertermia quanto febre, e se o suporte contém as seguintes informações:

- A hipotermia é definida como ter uma temperatura inferior à 20 graus Celsius;
- A febre é definida como ter uma temperatura superior à 39.2 graus Celsius;
- A relação ter uma temperatura funcional (um indivíduo não pode estar ligado a um único valor através deste relacionamento).

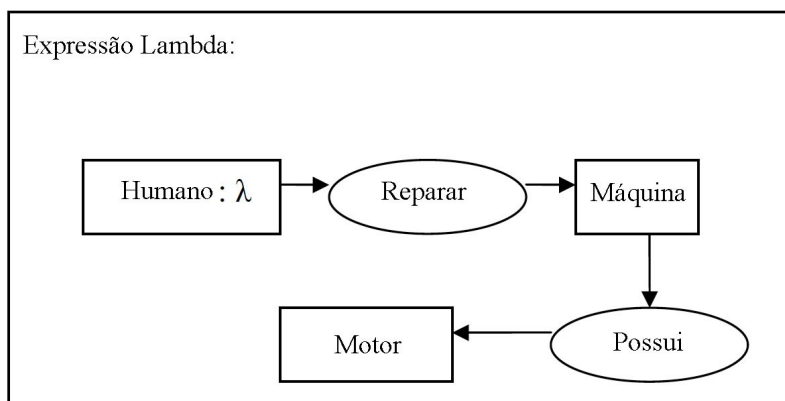
É possível graças à um sistema de equações ($\text{temperatura} > 39.2$ e < 20) identificar o problema e tomar o procedimento adequado para a correção.

b) A expansão dos grafos:

A expansão dos grafos - como foi descrito no capítulo 2 - é uma operação que permite a obtenção de novos conhecimentos e isto através de procedimentos puramente sintáticos e cujo curso segue o exemplo a seguir.

Seja o tipo de conceito “mecânico” definido pela expressão lambda na Figura 71:

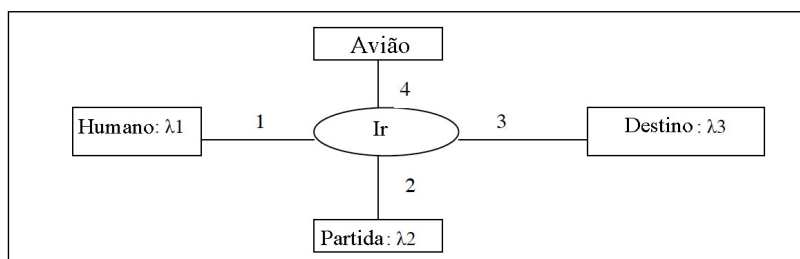
Figura 71 - Expressão lambda para o conceito mecânico



Fonte: elaborada pela autora

E seja a relação: voar(agente, partida, destino) - pegar o avião para ir de “partida” para “destino” - definido pela fórmula lambda na Figura 72:

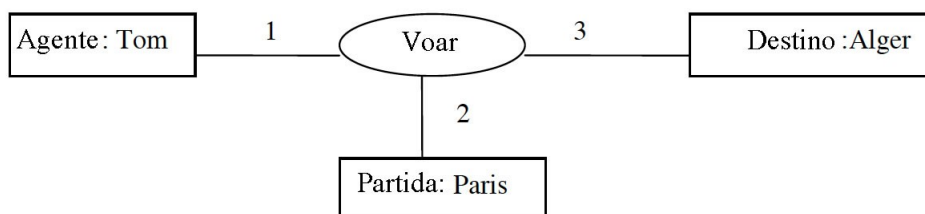
Figura 72 - Expressão lambda para o relação “voar”



Fonte: elaborada pela autora

Sua substituição no grafo na Figura 73: (Tom voa de Paris à Alger)

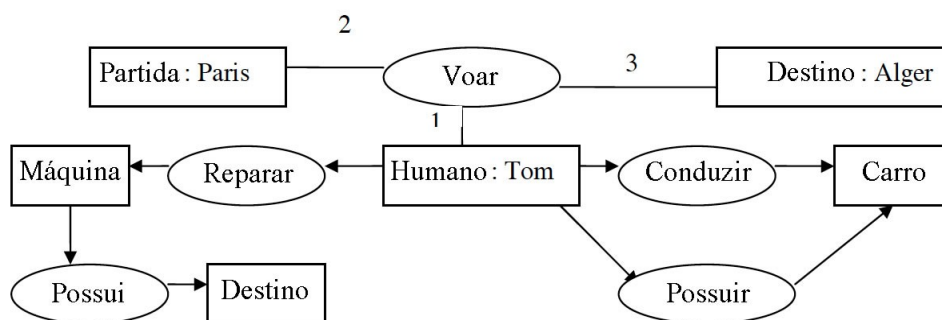
Figura 73 - Grafo para a relação “voar”



Fonte: elaborada pela autora

Nos dá o novo gráfico, na Figura 74:

Figura 74 - Grafo resultante



Fonte: elaborada pela autora

Tendo em conta as definições e métodos precedentes, é possível constatar que é efetivamente possível efetuar as manipulações tais como a inferência dedutiva possa ser naturalmente integrada dentro de uma plataforma cognitiva baseada sobre o modelo dos grafos conceituais. Na prática, foi mostrado que a dedução apesar de suas inegáveis vantagens continua sendo um freio para a expressividade das linguagens e à flexibilidade dos raciocinadores. O presente trabalho propõe então continuar o estudo da aplicabilidade a fim de iniciar vantagens as perspectivas e de aproveitar ao máximo as variedades de raciocínio que poderiam fornecer os grafos conceituais.

4.2.2 Raciocínio por abdução

A abdução é um processo de raciocínio que parece ser oposto ao raciocínio dedutivo. Na verdade, se a dedução se baseia sobre as premissas bem definidas ou axiomas para dar algumas conclusões; a abdução é um procedimento que parte de um conjunto de axiomas definidos para dar um conjunto de conclusões de natureza provável e até mesmo incerto. Mais formalmente, se é considerada, à título de exemplo, a regra do *Modus Ponens*, é possível dizer que à partir dos axiomas A e $A \rightarrow B$ é possível deduzir B .

No raciocínio abduutivo, se temos os axiomas B e $A \rightarrow B$ então podemos dizer que A é potencialmente verdadeiro, ou que é possível que o fato que A é verdadeiro seja a causa de que $A \rightarrow B$ é verdadeiro.

Para o presente trabalho, a abdução é pertinente como parte da resolução de problemas. Isto se faz seguindo um processo cíclico usado para gerar e refinar um conjunto de hipóteses que tem como objetivo explicar (porque é disso que se trata no raciocínio abduutivo) os dados observados.

O raciocínio gerador do modelo implementa este processo cíclico por meio de um conjunto de operações baseadas sobre o formalismo de grafos conceituais o que permite, neste caso, de aplicá-lo em nosso assunto de estudo e de aproveitar assim a possibilidade de utilizar este tipo de raciocínio sobre uma plataforma baseada sobre OWL.

As funcionalidades para garantir para a resolução de problemas no contexto de um raciocínio gerador de modelo se definem essencialmente por dois operadores:

- *Specialize*: Permite, por abdução, gerar hipóteses alternativas.
- *Fragment*: Remove eventuais incoerências das hipóteses mantendo a coerência com as observações.

Isto pode ser visto como uma forma de dedução onde o objetivo é de atribuir de vantagens de hipóteses podendo ser geradas no próximo ciclo.

4.2.2.1 Princípios da inferência abduativa:

Para os resolvidores de problemas em IA, trata-se principalmente de fornecer um conjunto de hipóteses sobre as quais serão baseadas as soluções potenciais respeitando as restrições de calculabilidade, caso contrário, não é possível obter os resultados.

Para continuar com o exemplo anterior, a abdução é um método que permite dizer, dados dois predicados A e B :

“Se B é verdadeiro, e se $A \rightarrow B$ é verdadeiro então A explica B .”

É o princípio dos métodos descritos por (LEVESQUE, 1989). Como cita (SHANAHAN,

1989): “a tarefa associada à inferência abdutiva é a explicação dos dados, como a tarefa associada à dedução é a predição”.

“As explicações são as estruturas hipotéticas geradas para ajustar e coordenar um conjunto de observações, e possuem unicamente o status de “possível” no sistema. Entretanto, se elas são verificadas, então a consequência será naturalmente verificada por dedução. Assim, encontrar uma explicação em um sistema lógico se torna encontrar uma expressão que, se ela é verdadeira, implicará os axiomas de partida” (HARTLEY; COOMBS, 1990b).

Apesar de que a abdução é necessária no método de resolução de problemas - como descrito por (PEIRCE, 1957), (KLEER; REITER, 1987), ela continua, entretanto, considerada como uma forma de raciocínio imperfeito no contexto da lógica dedutiva; além disso, foi demonstrado em (BYLANDER et al., 1989) que todo mecanismo baseado em abdução será, pelo menos, NP-difícil. Para contornar este problema, é necessário pedir alguns princípios para formalizar os aspectos da abdução, a saber:

1. A geração de hipóteses: As hipóteses são as regras que permitem religar os diferentes elementos susceptíveis de serem utilizados durante a abdução tais como as propriedades, os tipos, e os diferentes eventos do sistema; estas regras se dividem em quatro tipos:

– **Restrição seletiva do tipo:**

Exemplo: $\forall x A(x) \rightarrow B(x)$ (todos A 's são B 's).

– **Definição aristotélica:**

Exemplo: $\forall x A(x) \wedge B(x) \wedge C(x) \dots \rightarrow Z(x)$

(Se um objeto tem as propriedades A , B , C então é um objeto do tipo Z .)

– **Contingente ou definição esquemática:**

Exemplo: $\forall x Z(x) \rightarrow A(x) \wedge B(x) \wedge C(x) \dots$

(Inversa da precedente. Se um objeto é um Z então ele tem as propriedades A , B , C)

– **Expressão de causalidade:**

Exemplo: $A \wedge B \wedge C \dots \rightarrow X \wedge Y \wedge Z \dots$

(Se A, B, C ocorrerem então resultará que X, Y, Z se produzirá em consequência).

2. Parcimônia ou minimização: Princípio que evita a geração de hipóteses redundantes ou indesejáveis.
3. A avaliação: Princípio que serve para ordenar as hipóteses de acordo com sua importância. Os critérios de seleção podem ser assim a complexidade, a exatidão que o poder de predição ou de cobertura dos dados.

4.2.2.2 O operador de abdução (*Specialize*):

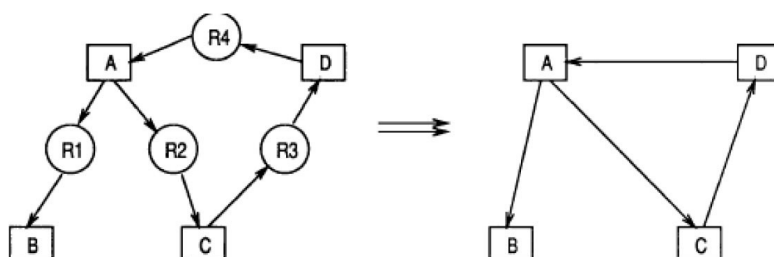
Operando sobre os grafos conceituais, este operador se define como sendo a composição de duas funções, ou seja “a cobertura” (operador *Cover*) e a “junção” (operador *Join*).

$$\bullet \textit{Specialize} = \textit{Join} \circ \textit{Cover}$$

Estes operadores não agem diretamente sobre um grafo conceitual mas sobretudo sobre uma simplificação deste último chamado Grafo de conceitos.

Um grafo de conceitos é um grafo conceitual do qual elimina-se todos os nós relação assim que a orientação dos arcos. A transformação é bastante simples e se aplica perfeitamente à todos os grafos conceituais de grau um (neste caso, foi trabalhado mais esse tipo de grafos). A simplificação em grafo de conceitos vem do fato que os operadores de especialização (*Cover* e *Join*) não afetam os nós relação em um grafo conceitual e como consequência eles podem ser omitidos. Exemplo na Figura 75:

Figura 75 - Grafo de conceitos



Fonte: elaborada pela autora

– Antes de detalhar os operadores que agem sobre esses grafos, serão definidos os seguintes conjuntos:

* F : Conjunto de grafos de entrada (os fatos).

* D : Conjunto de definições (De acordo com os tipos de regras previamente definidas).

* H : Conjunto das hipóteses resultantes depois da aplicação da cobertura.

– Será definido igualmente $C(g)$ como sendo o conteúdo conceitual do grafo ‘ g ’ (o conjunto de seus conceitos) e $M(C_1, C_2)$ como sendo o sub tipo comum maximal entre os dois conceitos C_1 e C_2 .

4.2.2.2.1 O operador Cover

Seu papel consiste em escolher um sub-conjunto apropriado de um conjunto de grafos D que cobre a totalidade dos conceitos de um certo sub conjunto de grafos de F . Mais formalmente é uma aplicação tal que:

Para $f \in F$, $\forall c \in C(f)$ então $\exists c', d$ tal que $c' \in C(d)$ para um $d \in D$ e $M(c, c')$ existe.

Como pode haver um número bastante importante de sub conjuntos que verificam esta condição, é essencial encontrar os sub conjuntos mais apropriados e mais pertinentes. (SOWA, 1984) mostrou que uma cobertura parcimoniosa (cobertura minimal) será apropriada para resolver este caso.

A cobertura minimal pode ser produzida minimizando a seguinte expressão booleana:

$\bigwedge_c \bigvee_i d_i$, onde $c \in C(f)$, $C(d_i)$

Se trata, então, de minimizar de forma que seja o menor número de alternativas possíveis e que cada alternativa seja assim pequena quanto possível.

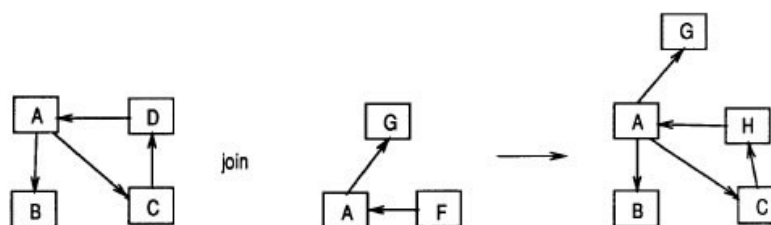
4.2.2.2.2 O operador *Join*

Operador binário que, sendo dado um sub conjunto apropriado de D obtido pela cobertura (*Cover*), permite produzir uma hipótese explicativa.

Assim como foi falado sobre a cobertura parcimoniosa (ou minimal) será falado aqui de junção maximal correspondendo ao operador (*Join*); a diferença reside no fato de que a junção normal permite de mesclar dois grafos em um ponto único onde os dois grafos compartilharão o mesmo nome de conceito, enquanto a junção maximal permite religa-los para todos os conceitos comuns mas também os sub tipos destes conceitos (isto dependerá então do sub tipo comum minimal dos dois conceitos sobre os quais se efetua a junção), e esta junção se faz sobre um número maximal de nós dois grafos.

Exemplo na Figura 76:

Figura 76 - Operação Join



Fonte: elaborada pela autora

Neste exemplo, a junção se faz em relação aos conceitos D e F tais que $M(D, F)=H$. Nota-se que embora seja uma operação binária, a junção pode não ser comutativa se leva em consideração o fator semântica.

Agora é possível compreender melhor o papel da cobertura. Na verdade, a única condição necessária para a junção sobre dois conceitos é que o sub tipo minimal comum seja diferente do tipo vazio \perp ; ou, a cobertura retorna somente os grafos maximamente contactáveis; isto é: um grafo de fatos F é maximamente contactável com todos os grafos que lhe cobrem.

Nota:

A abdução é uma técnica de inferência que permite entre outras coisas explorar os domínios e as proposições dificilmente alcançáveis através de um raciocínio baseado sobre a dedução. Foi visto também que esta técnica se adapta perfeitamente aos grafos de conceitos que são um sub-conjunto dos grafos conceituais, e portanto é um candidato para os futuros *reasoners* operando sobre este formalismo.

É necessário observar, contudo, que a abdução não se constitui sozinha como um método de raciocínio correto e completo. Como foi visto precedentemente, o raciocínio por abdução se baseia sobre um processo iterativo, onde a primeira fase utiliza o operador *Specialize* é baseado na abdução. A segunda fase representada pelo operador *Fragment* é dividido em dois sub-operadores *Project* e *Uncover* e se baseia sobre a dedução a fim de efetuar um trabalho de refinamento, em seguida de validação ou de rejeição das hipóteses produzidas durante o ciclo anterior (HARTLEY; COOMBS, 1990a).

4.2.3 Raciocínio por analogia

Foi estudada a viabilidade da aplicação do raciocínio por analogia sobre as bases de conhecimentos implementadas em grafos conceituais, e isto para as aplicações destinadas à Web Semântica.

Este estudo será feito pela apresentação de algumas sugestões onde o objetivo é demonstrar que os pré-requisitos do raciocínio por analogia podem ser satisfeitos pelo domínio e do formalismo escolhido.

4.2.3.1 O domínio da aplicação (A Web Semântica)

Várias aplicações que utilizam a Web Semântica podem conter os pré-requisitos necessários para a aplicação do raciocínio por analogia e onde este último trará uma grande ajuda na especificação e a consulta de conhecimentos, tais como os seguintes exemplos:

- A memória corporativa: apresentada no primeiro capítulo, serve de “agrupamento” dos diferentes conhecimentos e experiências adquiridas pela empresa

nos diferentes domínios, estas experiências são geralmente expressas sob forma de casos separados, encontrados pelos empregados e portanto o compartilhamento e a combinação permite uma exploração mais poderosa do “know-how” e das práticas desenvolvidas dentro da organização.

- Os fóruns técnicos: neste gênero de fóruns, o participante expõe mais frequentemente um problema encontrado durante uma atividade dada (falha encontrada, dificuldade de utilização, . . .) e isto especificando os parâmetros que foram escolhidos e a configuração de seu material (premissa) ao especificar outras informações que ele julga útil de adicionar (contexto) e espera dos participantes uma ajuda definida de suas experiências pessoais.

Armazenando as diferentes soluções propostas, explorando-as em uma base de casos, e ajudando os usuários a especificar suas consultas sob forma adequada (com a ajuda de interfaces e de máscaras para ataques adaptados) o raciocínio por analogia poderá aumentar a qualidade de tais fóruns oferecendo um serviço de ajuda automática, que os usuários poderão eles mesmos melhorar através de suas sugestões, apresentações e testes.

4.2.3.2 O raciocínio por analogia sob os grafos conceituais

4.2.3.2.1 A implementação de uma base de casos

A modularidade e a separação das generalidades dos fatos particulares devido a utilização do suporte nos grafos conceituais será de uma grande utilidade na implementação de bases de casos para um estudo análogo, é possível por exemplo implementar cada caso em um contexto CAS, contendo ele mesmo três outros contextos:

- Contexto “Premissa”: no qual serão descritas as premissas do caso através de um grafo conceitual.
- Contexto “Contexto”: descrevendo o contexto de validade do caso tal como as restrições espaço-temporais (data de validade, localização).
- Contexto “Solução”: onde será descrita a solução do caso através de um grafo.

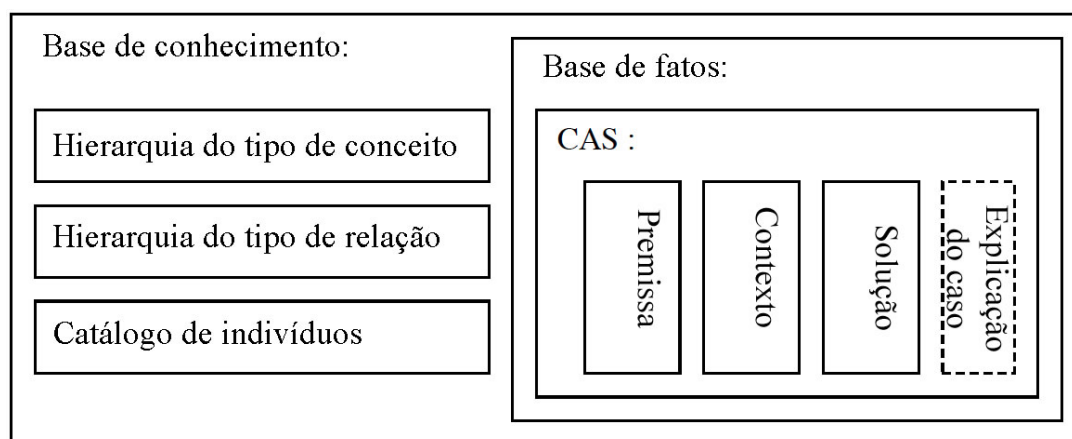
Nota:

A utilização dos *links* de correferência ajudará no mapeamento dos diferentes conceitos nestes três contextos, permitindo melhor descrever por exemplo o papel das premissas na elaboração da solução. Uma tal base de conhecimentos poderá ter a forma mostrada na Figura 77:

A representação do caso assim definido oferece várias vantagens, por exemplo:

- A facilidade de indexar o caso devido a seus contextos premissas que poderão ser facilmente recuperados e atribuídas aos casos que eles indexam.
- Testar a coerência e a validade da base devido aos conhecimentos do suporte, que servirão à validar os novos casos ou à validar os processos de raciocínio.
- A portabilidade dos casos, que poderão ser importados através de outras bases e associados à outros suportes permitindo uma personalização do raciocínio e uma reutilização das especificações de domínios (e isto devido a seu agrupamento na base de fatos).

Figura 77 - Base de conhecimentos



Fonte: elaborada pela autora

4.2.3.2.2 As etapas do raciocínio por analogia nos grafos conceituais

O formalismo dos grafos conceituais oferecem vários mecanismos podendo ser utilizados para elaborar um movimento do raciocínio por analogia, serão apresentados aqui algumas proposições definidas em seguida as sete etapas introduzidas:

1. Identificação que o RA (*Reasoning by Analogy*) pode ajudar na resolução do

problema-alvo atual:

A identificação dos méritos da utilização do raciocínio por analogia poderá ser feito devido a numerosas maneiras indo da demanda explícita do usuário - que poderá julgar que esta abordagem é a mais adaptada a suas necessidades - à observação do domínio tratado e a natureza dos conhecimentos representados. Outros meios mais automáticos poderão ser definidos tais como a forma das consultas do usuário (Ela oferece o aspecto de um caso alvo? Ela é muito confusa para uma utilização clássica?) ou a natureza das ontologias implementadas (elas contêm as estruturas de casos? . . . etc.).

2. Pesquisa de um problema base análogo ao problema-alvo:

Esta etapa pode ser definida como sendo a mais importante do raciocínio por analogia, ela permite a identificação dos casos já encontrados que podem ajudar a resolver o problema-alvo.

Esta escolha é geralmente implementada devido a uma função que toma como parâmetros dois casos, e que retoma uma probabilidade de concordância calculada seguindo a similaridade de suas premissas (tendo em conta seus contextos respectivos se necessário).

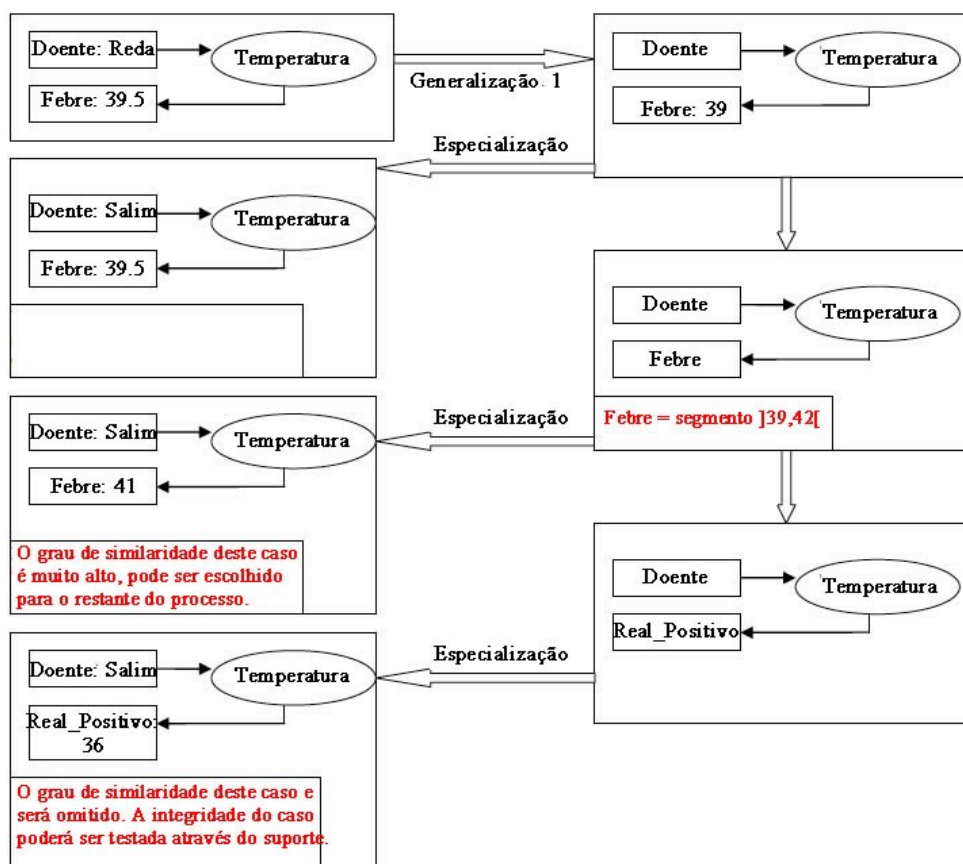
Esta função poderá ser implementada devido aos processos de especialização/generalização e a um sistema de pesos de conceitos - que descreverá o impacto da generalização de um conceito sobre o grau provável de similaridade do caso. É possível, por exemplo, definir que um problema-alvo é análogo à um problema de base, se existe um grafo conceitual definido como generalização dos dois.

As restrições sobre a generalização comum dos dois casos permitirá a melhora do cálculo da similaridade, o presente trabalho se limitará à dois entre eles:

- O grau de profundidade da generalização: supõem-se que quanto menos profundo, mais o grau de similaridade é forte.
- A generalização dos nós conceitos e relações importantes: por exemplo, a generalização de um conceito ‘doente’ definido sob um indivíduo através de um conceito ‘doente’ genérico não corre o risco de afetar o grau de similaridade.

Exemplo na Figura 78.

Figura 78 - Generalização do conceito doente



Fonte: elaborada pela autora

3. Elaboração ou eliminação dos aspectos não pertinentes do problema base:

A escolha dos nós para omitir poderá ser feita para transpor os grafos-alvo e os grafos de base escolhidos assim que para a eliminação dos conhecimentos supérfluos (tal como as informações pessoais dos pacientes, a cor dos equipamentos em pane, etc).

4. O mapeamento das partes do problema de base com os aspectos similares do problema-alvo:

A edição dos links de correspondência entre os conceitos e relações do problema-alvo e os problemas de base se fará devido aos conjuntos de correferência e às diferentes relações de equivalência e de igualdade de conceitos - que poderão ser declarados no suporte.

5. Inferência ou dedução das características (soluções) atribuídas ao problema-alvo a partir daquelas presentes no problema base:

A solução apresentada será obtida então pela combinação das diferentes soluções adotadas durante as etapas precedentes. Esta combinação poderá ser vista como uma composição dos

seguintes mecanismos:

- Junção dos grafos: permitirá mesclar várias soluções a fim de obter uma nova mais abrangente.
- Exclusão de redundâncias e de conhecimentos fúteis: diminuir o tamanho da solução apresentada e refinar sua semântica e sua apresentação.
- Personalização de acordo com o problema-alvo: a edição dos links de correferência permitirá a personalização da solução ou problema-alvo, tal como a substituição dos nós nos grafos da solução ou a resolução de equações de tipos - podendo ser definido em cálculo lambda por exemplo - tendo como variáveis os aspectos próprios do problema à resolver.

Tudo isso somente é possível devido a relação definida na etapa 4.

6. Justificação ou a verificação da validade das conclusões inferidas:

A validação da solução gerada poderá ser feita para várias etapas:

- *Validação referente aos domínios do raciocínio:*

O que se resume em um teste de consistência com a base de conhecimentos descrita pelo suporte e os casos já encontrados, ele permitirá a detecção de toda incoerência ocorrida durante as etapas precedentes do raciocínio.

- *Validação referente ao usuário:*

A intervenção ao usuário no processo de validação permitirá em vários casos detectar as anomalias pelo sistema de raciocínio, mas isto demanda uma apresentação clara ao usuário que deverá facilmente relacionar os diferentes conhecimentos em jogo nos processos de raciocínio.

- *A validação prática:*

O teste final da solução inferida continua sua aplicação no domínio, esta abordagem permitirá - nos domínios onde ela é possível - definir de uma vez por todas sua validade e sua possível exploração na nova inferência; uma solução alternativa ao teste prático será a simulação de maneira artificial.

7. Aprendizagem, quer dizer, memorização do problema-alvo enriquecido de sua solução:

Após a validação da solução, será possível utilizar - através da combinação com as premissas e ao contexto do problema-alvo - como referência e isto na sua implementação usando um novo contexto do tipo CAS na base de conhecimentos.

Discussão

No capítulo anterior foi questionado se os grafos conceituais podem apresentar uma plataforma compatível com os diferentes modos de raciocínio que foram estudados assim como as técnicas a eles atribuídos.

Nessa perspectiva, realizamos este estudo onde o objetivo era identificar as premissas e os métodos necessários à aplicação destes modos de raciocínio, em seguida procurar as equivalências - no melhor dos casos - ou alternativas que permitam a sua implementação nos sistemas inferenciais operando sobre os conhecimentos baseados sobre os grafos conceituais.

Neste capítulo, foram apresentados o substituto estrutural - sob forma de uma nova notação para OWL - e o substituto inferencial expondo os métodos de concepção e as ferramentas de exploração oferecendo a possibilidade de implantar estes raciocínios seguindo a notação proposta equivalente à um OWL-CG.

Em seguida, verifica-se que do lado da inferência, o avanço atual dos trabalhos está longe de ser bem sucedido, e foi possível notar que mesmo se cada um dos três modos de raciocínio que foram abordados apresentasse um certo número de vantagens operacionais o tornando apto à satisfazer certos tipos de aplicações práticas; nenhum dos três saberia conquistar o título de raciocínio por excelência para todo tipo de sistema à base de conhecimentos; uma tal abordagem inferencial deve combinar ao mesmo tempo o rigor que prodiga o raciocínio lógico, a flexibilidade da manipulação e a abertura que apresenta a abdução, e os aspectos operacionais e aplicativos do raciocínio à base de casos.

Se conclui que cada um desses modos de inferência pode ser adotado - que já é o caso - no âmbito dos sistemas especialistas onde será utilizado para responder às necessidades bem específicas. Porém, como no caso da resolução de problemas - que utiliza da abdução para obter novas hipóteses e da dedução para lhes refinar - parece que estabelecer de alguma

forma as “alianças” entre estas abordagens deixa entrever um resultado para novas formas de raciocínio híbridos combinando suas técnicas e reagrupando suas vantagens, e será mais que recomendável seguir um filão assim promissor.

4.3 Resultado 3 - Exemplos com a nova linguagem SOWL

Um dos objetivos do presente trabalho consiste em uma “tradução” que permite recuperar as ontologias OWL baseadas sobre o formalismo das lógicas descritivas e manipular-las a fim de gerar as ontologias equivalentes baseadas sobre o formalismo dos grafos conceituais.

Para deixar mais claros os exemplos, uma breve recapitulação com o objetivo de apresentar praticamente o que é uma ontologia.

4.3.1 Ontologias e OWL

As ontologias são coleções de conhecimentos formalizados e agrupados por domínios. Elas se apresentam como um modelo que descreve a maneira de ser das diferentes entidades que compõem um dado domínio de utilização. Podemos dizer (KLEER; REITER, 1987): “Uma ontologia é a especificação de uma conceitualização de um domínio de conhecimento.”

As ontologias permitem a unificação da estrutura e da semântica das informações modeladas, oferecendo assim a vantagem de uma representação formalizada e independente do tipo do dado; a normalização da semântica dos conhecimentos permite efetuar inferências a fim de expandir seus conhecimentos ou efetuar os tratamentos inteligentes de forma automatizável.

É possível representar e manipular os principais componentes ontológicos tais como as classes, os indivíduos que são as instâncias destas classes, os papéis e os axiomas que regem a base de conhecimento. Estes componentes permitem essencialmente quatro tipos de funcionalidades:

- A definição de classes e as relações entre classes.
- A definição de propriedades e as relações entre propriedades.
- A definição de instâncias de classes ou de propriedades e as relações entre elas.
- A definição de informações de suporte.

Na sequência, estes elementos através de exemplos práticos de sua utilização sobre uma ontologia de domínio, e será abordada a problemática do presente trabalho que consiste em passar da linguagem OWL do formalismo das lógicas de descrição para aquele dos grafos conceituais, preservando suas funcionalidades originais. Mas primeiramente, recordamos algumas noções sobre os grafos conceituais.

4.3.2 Grafos conceituais

Desenvolvidos por J.F. Sowa (1984) sobre a base de redes existenciais de Charles Sanders Peirce e das redes particionadas; utilizam um dos princípios da psicologia cognitiva que afirma que um conceito é considerado como sendo bem definido se é possível conhecer sua posição em relação à outros conceitos presentes no domínio modelado.

Os grafos conceituais são os grafos bipartidos, como as lógicas de descrição, eles utilizam as noções de conceitos e de relações conceituais, mas oferecem entretando uma maior flexibilidade na manipulação destas primitivas, o que acarreta em vantagens interessantes, tais como:

- A flexibilidade de modelagem, que facilita a especificação e a interpretação dos conhecimentos.
- Um bom grau de formalismo, atribuindo a possibilidade de definir e de justificar os diferentes métodos de inferências, sem apresentar o inconveniente de uma grande rigidez.

O conjunto de conhecimentos descritos devido aos grafos conceituais formam uma base de conhecimento dividida em quatro partes maiores - além das informações operacionais contidas no suporte - sob forma de *treillis* regida pelas hierarquias de subclassificação (subsunção). Estas partes são:

- **A hierarquia de conceitos:** grafo de conceitos ligados pelos arcos que re-

presentam as relações de subclassificação.

- **A hierarquia de relações:** grafo de subclassificação de relações, define suas propriedades tais como suas aridades ou suas assinaturas.
- **O catálogo dos indivíduos:** conjunto de marcadores identificando os indivíduos e os tipos com os quais eles são compatíveis, assim que as declarações que os definem.
- **A base de fatos:** grafo de declarações representando os fatos da base de conhecimentos.

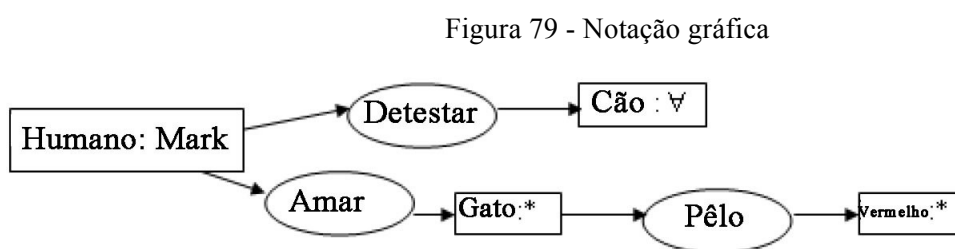
Os grafos conceituais possuem também várias notações permitindo lhes definir e representar, a mais evidente para este formalismo é provavelmente a notação gráfica com as vantagens cognitivas e os inconvenientes práticos conhecidos.

O exemplo seguinte corresponde ao grafo conceitual definindo que o indivíduo “Mark” detesta os cães e ama os gatos de pelo vermelho.

Notação gráfica:

[Cao *x][Gato *y][Vermelho *z](Detestar [humano : Mark] ?x) (Amar [humano : Mark] ?y) (Pelo ?y ?z)

Notação gráfica correspondente na Figura 79:



Fonte: elaborada pela autora

4.3.3 Demonstração por exemplos

Seguem as funcionalidades propostas pelo OWL e mostrar, por exemplo, seus equivalentes após a passagem ao formalismo dos grafos conceituais. Permitindo, para toda ontologia OWL baseada sob as lógicas de descrição, gerar seu equivalente em grafos conceituais sob forma de uma base de conhecimentos utilizando a notação da SOWL.

4.3.3.1 A declaração de classes

Se faz em OWL de várias formas, vamos apresenta-las uma por vez, assim como sua tradução em SOWL.

4.3.3.1.1 Por enumeração de suas instâncias

Exemplo: a classe *Pais* é definida pelo conjunto

{América, Inglaterra, França, Alemanha, Itália} Sua notação em OWL se faz como segue:

```
<owl: Class rdf:ID="Pais">
  <owl:oneOf rdf:parseType="Colecao">
    <owl:Thing rdf:about=" America"/>
    <owl:Thing rdf:about=" Inglaterra"/>
    <owl:Thing rdf:about=" Franca"/>
    <owl:Thing rdf:about=" Alemanha"/>
    <owl:Thing rdf:about=" Italia"/>
  </owl:oneOf>
</owl:Class>
```

Em SOWL:

```
(DEF[TypeLabel'Pais'][[INTERSECTIONOF:[ENUMERATION:[Pais:America]
[Pais:Inglaterra ] [Pais:Franca ] [Pais : Alemanha] [Pais : Italia ] ] [DomainConcept] ] )
```

4.3.3.1.2 Por suas propriedades:

Exemplo: a *Pizza_de_calabresa* se define como sendo uma *Pizza* que possui a propriedade "hasTopping Ingrediente_Calabresa". Sua notação se faz como segue:

```

<owl:Class rdf:about=" Pizza_de_calabreza">
  <rdfs:subClassOf>
    <owl:Class rdf:about="Pizza"/>
  </rdfs:subClassOf>
  <rdfs:Restriction>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:about="hasTopping"/>
      </owl:onProperty>
      <owl:someValuesFrom>
        <owl:Class rdf:ID="Ingrediente_Calabresa"/>
      </owl:someValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

```

Em SOWL:

```

(DEF[TyeLabel'Pizza_de_calabreza']INTERSECTIONOF:[RESTRICTION:@some[Relatin
Type:hasTopping [ConceptType:Ingrediente_Calabresa]] [Pizza]])

```

4.3.3.1.3 Por restrição sobre suas propriedades:

“owl:allValuesFrom, owl: someValuesFrom, owl:hasValue owl:maxCardinality,
 owl: minCardinality, owl:Cardinality”

Exemplo: Uma Pizza interessante é uma Pizza que tem pelo menos 3 ingredientes. Sua notação em OWL fica assim:

```

<owl:Class rdf:ID="Pizza_interessante" >
  < rdfs : subClassOf rdf : resource = "Pizza" / >
  < owl : Restriction >
    < owl : onPropertyrdf : resource = "hasTopping" / >
    < owl : minCardinalityrdf : resource = "3" / >

```

```

        </owl : Restriction >
</owl : Class >

```

Em SOWL:

```

(DEF[TypeLabel'Pizza_interessante'][[INTERSECTIONOF:[CARDINALITY:@>3
[RelationType : hasTopping]][Pizza]])

```

4.3.3.1.4 Por composição de classes:

“owl:intersectionOf, owl:unionOf, owl:complementOf”

Exemplo: Um tempero pode ser picante, moderadamente picante, ou suave.

Sua notação em OWL fica como segue:

```

<owl:Class rdf : ID= “Tempero”>
  <owl:intersectionOf rdf : parseType= “Colecao”>
    <owl:Class rdf : about= “Tempero_Picante”>
    <owl:Class rdf:about= “Tempero_Moderado”>
    <owl:Class rdf:about= “Tempero_Suave”>
  </owl : intersectionOf>
</owl : Class>

```

Em SOWL:

```

(DEF[TypeLabel'Tempero'][[UNIONOF:[Tempero_Picante][Tempero_Moderado]
[Tempero_Suave]])

```

4.3.3.2 A declaração de relações interclasses

4.3.3.2.1 As subclasses

Exemplo: Uma massa_espessa_e_suave também é uma massa. Sua notação em OWL fica

como segue:

```
<owl:Class rdf:ID= "massa_espessa_e_suave" >
  <rdfs : subclassOf rdf:resource= "Massa"/>
</owl:Class>
```

Em SOWL (onde GT representa a subclassificação)

```
(GT [ConceptType Massa] [ConceptType Massa_espessa_e_suave])
```

4.3.3.2.2 A equivalência entre classes

Exemplo: “*Drink*” e “*bebida*” representam o mesmo conceito. Sua notação em OWL fica como segue:

```
<owl:Class rdf:ID= "Drink">
  <owl:equivalentClass rdf :resource = "Bebida"/>
</owl:Class>
```

Em SOWL (*EQ* a dupla subclassificação \Leftrightarrow igualdade interclasses):

```
(EQ [ConceptType Drink] [ConceptType Bebida])
```

4.3.3.2.3 A disjunção entre classes

Exemplo: *Pizza_nao_vegetariana* e *Pizza_vegetariana* são duas classes disjuntas.

Sua notação em OWL fica como segue:

```
<owl:Class rdf:ID= "Pizza_vegetariana">
  <owl:disjointWith rdf:resource= "Pizza_nao_vegetariana"/>
</owl:Class>
```

Em SOWL:

```
(DISJOINT [ConceptType Pizza_nao_vegetariana] [ConceptType Pizza_vegetariana])
```

4.3.3.3 A definição das propriedades e das relações entre propriedades

rdfs:subPropertyOf, rdfs:domain, rdfs:range, owl:equivalentProperty
owl:inverseOf, owl:SymmetricProperty, owl:TransitiveProperty

É feito praticamente da mesma forma, será dado um exemplo que engloba a maioria de suas propriedades, na ocorrência da propriedade “*hasBase*”:

```
<owl:InverseFunctionalProperty rdf:about= “hasBase”>
  <rdfs:domain rdf:resource= “Pizza”/>
  <rdfs:range rdf:resource= “Massa”/>
  <rdf:type rdf:resource= “http://www.w3.org/2002/07/owlObjectProperty”/>
  <rdf:type rdf:resource= “http://www.w3.org/2002/07/owlFunctionalProperty”/>
  <rdfs:subPropertyOf rdf:resource= “hasIngredient”/>
  <owl:inverseOf rdf:resource= “isBaseOf”/>
</owl:InverseFunctionalProperty>
```

Em SOWL (a transitividade sendo obtida por herança de “*hasIngredient*”):

```
(Def[Relationtype‘hasBase’][Lambda LambdaExpression (Pizza ?x, Massa ?y)])
INVERSEOF [relationType isBaseOf] [relationType hasBase] ( FONC@2 [ Relationtype
hasBase ] ( TRANS [ Relationtype hasBase ] )
```

4.3.3.4 A definição dos indivíduos

É feita de uma maneira simples seja em OWL clássico ou em notação SOWL, não obstante tem uma particularidade específica aos grafos conceituais.

A definição de indivíduos “sorvete de chocolate” e “Luigi_especial” escrito em OWL assim:

```
< Sorvete rdf:ID=“IceCream_Chocolate” />
< Pizza_italiana_original rdf : ID = “Luigi_special” / >
```


Em SOWL:

[Sorvete : IceCream_Chocolate] [Pizza_italiana_original : Luigi_special]

Caso da definição da Quatro_Queijos instância de Pizza_de_queijo:

[Pizza_de_Queijo : Quatro_Queijos]

(isBaseOf[Massa_espessa_e_suave:Massa_espessa_e_suave_1]

[Pizza_de_queijo:Quatro_Queijos])

hasBase[Pizza_de_queijo:Quatro_Queijos][Massa_espessa_e_suave:Massa_espessa_e_suave_1])

(hasTopping [Pizza_de_queijo:Quatro_Queijo][ParmesanTopping:ParmesanTopping_1])

(hasTopping[Pizza_de_Queijo:Quatro_Queijos][MozzarellaTopping:MozzarellaTopping_1])

(isToppingOf [ParmesanTopping:ParmesanTopping_1][Pizza_de_queijo:Quatro_Queijos])

(isToppingOf[ParmesanTopping:MozzarellaTopping_1][Pizza_de_queijo:Quatro_Queijos])

Como foi mencionado anteriormente, existe uma particularidade dos grafos conceituais que faz com que um indivíduo seja definido não somente por sua definição formal, mas sobretudo por sua posição no grafo em frente à todos os outros elementos com os quais ele interage, em outras palavras devido as declarações contidas no catálogo de indivíduos, como para o exemplo acima.

Entretanto, o formalismo dos grafos conceituais não oferece nenhuma indicação sobre a metodologia a ser adotada para definir as declarações que são destinadas à identificação dos indivíduos daqueles que servem para “decorar” a base de fatos.

Nossa abordagem para o problema se baseia no seguinte: seja uma agência imobiliária que gera entre outras coisas os clientes com imóveis para a venda. Para a agência, as informações primordiais não serão, certamente, aquelas que nos interessaria no caso de uma base de conhecimentos sobre os clientes humanos, isto é, suas características, sua família, seus

filhos, etc. A agência definirá que as relações que servem para identificar os clientes serão aquelas para que servem para definir sua identidade, depois aquelas que tratam dos imóveis que possui, suas características, suas localizações, etc.

Assim, os domínios de aplicações de ontologias são vastos como a própria vida, por isso impossível limitar a criatividade humana impondo uma abordagem para a classificação dos dados de identificação e os dados de operacionalização, e fica aberta para nossa abordagem a escolha de definir as declarações e classificá-las segundo o julgamento do usuário da ontologia.

5 Conclusão e considerações finais

A Web Semântica representa a evolução necessária da plataforma Web atual. Durante este estudo, houve a tentativa de observar bem as questões ideológicas subjacentes que as diferentes tecnologias sob as quais se fundamenta esta evolução. Constata-se que a Web Semântica é uma plataforma evolutiva na qual circulará (em um futuro próximo) um fluxo importante de dados, estes últimos serão de natureza heterogênea e seu tratamento através dos métodos clássicos correm o risco de se tornarem complexos e de rentabilidade insatisfatória em um mundo onde as prioridades são rapidez e eficácia. Como consequência, os trabalhos são conduzidos no âmbito da comunidade de pesquisa a fim de responder estes problemas; as pesquisas seguem principalmente dois eixos, o primeiro de análise dos modos de representação de dados; o segundo, dos tratamentos das informações e da inferência, e foi visto que estes dois eixos não são disjuntos, mas ao contrário, tendem a se completar.

No contexto da inovação e da melhora das técnicas de representação de conhecimentos, este domínio de pesquisa já tinha uma base teórica sólida resultante de diferentes pesquisas em ciências cognitivas e em inteligência artificial. Deve-se notar também que através das representações baseadas sobre modelos lógicos, apesar de sua falta de expressividade, é possível construir bases de conhecimento coerentes e consistentes se baseando sobre um modelo teórico tão formal quanto as lógicas clássicas, isto apresenta também um ponto essencial quando é orientado sob o ponto de vista da inferência e do raciocínio. Por isto, os modelos estruturados tais como as lógicas de descrições ou ainda os grafos conceituais permitem uma base teórica muito próxima ao formalismo lógico. Estes modelos apresentam maior expressividade e são mais adaptados ao conteúdo rico e diverso da Web; estão no centro das pesquisas atuais visando a formalização, manipulação e o trânsito de dados na rede.

Nesse sentido, são as ontologias o novo aspecto que terão os dados na rede. Elas transcendem a simples representação sintática e atribuem tratamentos inteligentes baseados sob a semântica do domínio e as entidades que lhes são representadas assim como suas relações. Atualmente, a linguagem adotada pelo W3C como linguagem principal de criação e edição de ontologias é OWL, e é justamente sobre ela o foco da escolha deste estudo.

Enquanto que linguagem de representação de ontologias, OWL é capaz de descrever o universo sob formas de conceitos e de relações entre os conceitos e utiliza para isto os formalismos das lógicas de descrição; no qual o formalismo dos grafos conceituais parecem mais adequados para este gênero de representações e oferecem vantagens em termos de expressividade e de operacionalização.

Por isso a importância de encontrar uma abordagem para OWL baseada nos grafos conceituais, e foi proposto neste trabalho uma adaptação da linguagem em direção a este formalismo e isto redefinindo as primitivas de OWL por analogia com seus equivalentes em lógicas de descrição ou propondo novas.

Este trabalho permitiu constatar as vantagens dos grafos conceituais no tratamento das ontologias que se resumem essencialmente nas vantagens estruturais e representativas, ou seja, a possibilidade de representar os conhecimentos de tipos mais variados que não seria possível de representar utilizando o OWL clássico e isto de uma forma mais simples. Como foi visto, a representação de conhecimentos modais se torna mais fácil, e é possível igualmente representar facilmente os conhecimentos que se incluem no domínio das lógicas de ordem superior ou da lógica difusa.

Do ponto de vista da inferência, vimos igualmente que alguns modos de raciocínio eram aplicáveis sobre os SBC baseados sobre os grafos conceituais, foi falado do raciocínio por analogia e da abdução a título de exemplo.

De ponto de vista operacional, a passagem aos grafos conceituais permitiu constatar novas possibilidades quanto aos métodos de otimização de pesquisa, de extração e de manipulação dos dados devido aos estudos pré-estabelecidos em ciências cognitivas e em pesquisa operacional.

5.1 Trabalhos futuros

Este trabalho é uma contribuição sob o ponto de vista da adoção dos grafos conceituais em grande escala no âmbito da Web Semântica. Na verdade, o objetivo real não se resume à uma simples adaptação de uma linguagem ou à uma tradução de primitivas de um

formalismo para outro; mas estudar as perspectivas que esta iniciativa abre, ou seja:

- A implementação de uma interface compatível com um programa de edição de ontologias “Protégé” onde as ontologias OWL seriam baseadas sobre as lógicas de descrição. Esta interface permitiria retranscrever praticamente qualquer ontologia OWL no formalismo dos grafos conceituais devido a uma redefinição das primitivas e dos construtores de OWL;
- A elaboração de linguagens de descrição de ontologias que sejam inteiramente baseadas sobre os grafos conceituais; aproveitando para propor um conjunto inovador de construtores em grafos conceituais e não se restringindo àqueles das lógicas de descrição, a escolha destes construtores será baseada sob um estudo das contribuições que eles poderão trazer e/ou os problemas que eles poderão resolver; é possível por exemplo fazer um estudo completo a fim de identificar, apontar, e eventualmente superar os problemas de decidibilidade e de calculabilidade (verificar se é possível - devido a passagem através dos grafos conceituais - resolver os problemas de ordem NP-difícies).
- Explorar mais profundamente as possibilidades que oferece este novo método do ponto de vista da representação de conhecimentos e das inferências que lhe serão atribuídas, à título de exemplo foi falado da facilidade de representação de conhecimentos modais e difusos, assim como a possibilidade de criar as relações terciárias ou n-árias; e do lado do raciocínio foi citado o raciocínio baseado em casos e o raciocínio por abdução;
- Assim como para a contribuição estrutural e representativa, será sensato estudar - com o objetivo de criar, posteriormente - as possibilidades que poderá oferecer os motores de inferência e dos raciocinadores especialmente projetados para operar nas plataformas baseadas sob os grafos conceituais;
- Projetar precisamente plataformas de desenvolvimento utilizando linguagens de edição de ontologias baseadas em grafos conceituais e tentar aproveitar ao máximo deste formalismo, até mesmo para eventualmente torná-las mais acessíveis e mais ergonômicas devido as possibilidades de expressão e de intuitividade que oferecem os GCs assim que a possibilidade de representar os conhecimentos ambíguos no domínio da linguagem natural, mas também

mais rápidos devido as técnicas de percurso de dados de pesquisa e de extração herdadas da teoria do grafos.

Além disso, enfatizando a importância dos grafos conceituais para a aplicabilidade das ontologias, principalmente no âmbito da Web Semântica, é necessário citar a existência de trabalhos que utilizam seu formalismo para uma combinação (matching) de ontologias baseada em axiomas, como em (FÜRST; TRICHET, 2005). Neste artigo os autores propõem um novo paradigma no contexto dos grafos conceituais, onde a projeção (a principal operação para raciocínio com grafos conceituais que corresponde ao homomorfismo de grafos) é usada como uma forma de combinar semanticamente os conceitos e as relações de duas ontologias através de uma explícita representação de axiomas em termos de grafos conceituais.

Com o objetivo de gerenciar múltiplas ontologias, que é a questão central de muitas aplicações que requerem interoperabilidade, a palavra de ordem é definir correspondências semânticas entre várias ontologias distintas, abrangendo domínios sobrepostos, isto é, resolver o problema de múltiplas ontologias para o mesmo domínio. Nesse contexto, a Web Semântica é a principal aplicação nesse sentido.

Existe um nível de heterogeneidade semântica que leva a um grande número de ontologias cobrindo domínios sobrepostos. Ou seja, existem muitas ontologias (definidas por diferentes comunidades) para o mesmo domínio. Por isso é muito importante descobrir as correlações entre a modelagem de primitivas (por exemplo, conceitos e relações) em ontologias separadas. Esta técnica se chama “*ontology matching*” é a questão central do processo de gerenciamento de ontologias múltiplas.

No artigo, os autores definem uma abordagem de combinação de ontologias baseada no uso explícito de todos os componentes de uma ontologia. Esta abordagem requer a explícita representação dos axiomas de duas ontologias (que são consideradas no processo de combinação) no nível conceitual, e não no nível operacional como é feito na maioria dos casos relacionados à engenharia das ontologias. O trabalho visa preencher esta lacuna permitindo a representação dos axiomas no nível conceitual e o uso a utilização desse aumento na semântica para conceitos e relações de mapeamento.

Os resultados do trabalho citado mostram que através dos grafos conceituais é possível representar conhecimento terminológico através da especificação de conceitos e relações, e

representar propriedades clássicas e qualquer tipo de axiomas no nível conceitual. Portanto, o método de combinação proposto se baseia no morfismo ontológico fundamentado na representação de conhecimento baseada em grafos e mecanismos de raciocínio baseados em grafos.

Para concluir, enfatizamos que os grafos conceituais se apresentam como sendo um formalismo promissor, e como consequência seria conveniente estudar a possibilidade de adotar linguagens como a SOWL como padrão de representação dos conhecimentos no âmbito da Web Semântica. Além disso, o presente trabalho se mostra particularmente relevante para a pesquisa que trata do gerenciamento e *matching* de ontologias, uma vez que os principais componentes destas ontologias são axiomas que podem ser facilmente representados e comparados com soluções baseadas em grafos conceituais.

Referências

- ABITEBOUL, S.; BUNEMAN, P.; SUCIU, D. **Data on the Web**: from relations to semistructured data and XML. [S.l.]: Morgan Kaufmann, 2000.
- ALLEMANG, D.; HENDLER, J. **Semantic web for the working ontologist**: effective modeling in RDFS and OWL. [S.l.]: Access Online via Elsevier, 2011.
- ALVES, R. C. V. **Web Semântica**: uma análise focada no uso de metadados. 2005. 182f. Dissertação (Mestrado) — Universidade Estadual Paulista. São Paulo, 2005.
- ANTONIOU, G. et al. Proof explanation for a nonmonotonic semantic web rules language. **Data & Knowledge Engineering**, North-Holland, v. 64, n. 3, p. 662–687, 2008.
- ARJONA, M. G. M.; SÁNCHEZ, A. H. **Lógica para principiantes**:CD. [S.l.]: Alianza editorial, 2004.
- ARTALE, A.; FRANCONI, E. A temporal description logic for reasoning about actions and plans. **Journal Of Artificial Intelligence Research**, Volume 9, pages 463-506, 1998. Disponível em: <<http://arxiv.org/pdf/1105.5446v1.pdf>>. Acessado em: 30 maio 2013.
- BAGET, J. F., Croitoru, M., Gutierrez, A., Leclere, M., & Mugnier, M. L. (2010). Translations between rdf (s) and conceptual graphs. In *Conceptual Structures: From Information to Intelligence* (pp. 28-41). Springer Berlin Heidelberg.
- BAGET, J.F. et al. Les langages du web sémantique. **Revue I3**, Lyon, 2004. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.80.2174_%26rep=rep1&type=pdf>. Acessado em: 20 jun. 2013.
- BATES, M. J. The invisible substrate of information science. **Journal of the American Society for Information Science**, v. 50, n. 12, p. 1043–1050, 1999.
- BELHADEF, H.; KHOLLADI, M.-K. Conception d’une nouvelle ontologie pour l’interopérabilité des systèmes d’informations géographique. In: THE 4TH INTERNATIONAL CONFERENCE: SCIENCES OF ELECTRONIC, TECHNOLOGIES OF INFORMATION AND TELECOMMUNICATIONS, 4., 2007, Tunísia. **Anais...** Tunísia, SETIT, 2007.
- BENATALLAH, B. et al. On automating web services discovery. **The VLDB Journal**, [S.l.], v. 14, n. 1, p. 84–96, 2005.
- BERNERS-LEE, T. Semantic web on xml. In: SLIDES FROM XML 2000 CONFERENCE, 2000, Washington. **Slides...** Washington, XML2000, 2000.
- BERNERS-LEE, T. Relational databases on the semantic web. 1998. Disponível em: <<http://www.w3.org/DesignIssues/RDB-RDF.html>>. Acesso em: 15 ago. 2012.
- BERNERS-LEE, T. et al. The semantic web. **Scientific american**, New York, v. 284, n. 5, p. 28–37, 2001.

BIZER, C.; HEATH, T.; BERNERS-LEE, T. Linked data-the story so far. **International Journal on Semantic Web and Information Systems**, USA, v. 5, n. 3, p. 1–22, 2009.

BORKO, H. Information science: what is it? **American documentation**, [California], v. 19, n. 1, p. 3–5, 1968.

BOUAUD, J. et al. Methodological principles for structuring an “ontology”. In: PROCEEDINGS OF THE IJCAI’95. **Workshop on “basic ontological issues in knowledge sharing**. [S.l.: s.n.], 1995. p. 19–25.

BRUIJN, J. de; WELTY, C. **Rif rdf and owl compatibility**. 2010. (W3C Working Draft - July 2009). Disponível em: <<http://www.w3.org/TR/rif-rdf-owl>>. Acessado em: 20/10/2010.

BUCHE, P., Dibie-Barthélemy, J., & Ibanescu, L. (2008). Ontology Mapping Using Fuzzy Conceptual Graphs and Rules. *ICCS Supplement*, 1724.

BUCKLAND, M. K. Information as thing. **JASIS**, v. 42, n. 5, p. 351–360, 1991.

BURSTEIN, M. H. Concept formation by incremental analogical reasoning and debugging. **Machine learning: An artificial intelligence approach**, California, v. 2, p. 351–370, 1986.

BUVAE, S.; MASON, I. A. Propositional logic of context. In: **Proceedings of the eleventh national conference on artificial intelligence**. [S.l.: s.n.], 1993.

BYLANDER, T. et al. Some results concerning the computational complexity of abduction. In: PROCEEDINGS OF THE FIRST INTERNATIONAL CONFERENCE ON PRINCIPLES OF KNOWLEDGE REPRESENTATION AND REASONING, 1, 1989, San Francisco. **Anais...** San Francisco, [ICPKRR?], 1989, p.44-54.

CAFÉ, L.; BRÄSCHER, M. Organização do Conhecimento: teorias semânticas como base para estudo e representação de conceitos; Organización del Conocimiento: teorias semânticas como base para estudios y representación de conceptos. **Informação & Informação**, [S.l.], v. 16, n. 2, p. 25-51, dez. 2011. <<http://www.uel.br/revistas/uel/index.php/informacao/article/view/10388>>. Acesso em: 15 ago. 2012.

CAMPBELL, J.; WOLSTENCROFT, J. Structure and significance of analogical reasoning. **Artificial Intelligence in Medicine**, [Elsevier], v. 2, n. 2, p. 103–118, 1990.

CAPURRO, R. Moral issues in information science. **Journal of information science**, London, v. 11, n. 3, p. 113–123, 1985.

CAPURRO, R.; HJØRLAND, B. The concept of information. **Annual review of information science and technology**, [Medford], v. 37, n. 1, p. 343–411, 2003.

CARVALHO, D. M. W. F. D. **Competências e atividades**: uma contribuição a um estudo da representação. 2003. 123f. Dissertação (Mestrado) — Pontifícia Universidade do Paraná. Paraná, 2003.

CASTELLS, M. **A sociedade em rede**. v.1. São Paulo: Paz e Terra, 1999.

CAUSSANEL, J. et al. Les topic maps sont-ils un bon candidat pour l'ingénierie du web sémantique. **Tech-CICO Laboratoire Technologies de la Coopération pour l'Innovation et le Changement Organisationnel**, Paris 2002. Disponível em: < http://www.topicmapslab.de/publications/les_topic_maps_sont-ils_un_bon_candidat_pour_la_ingenierie_du_web_semantique>. Acesso em: 18 ago. 2012.

CHAMPIN, P.A. **Rdf tutorial**. [S.l]: Citeseer, 2001.

HARLET, J.; BACHIMONT, B.; TRONCY, R. Ontologies pour le web sémantique. **Action spécifique**, v. 32, p. 43–63, 2003.

CHARLET, J.; LAUBLET, P.; REYNAUD, C. **Web sémantique**: rapport final de l'action spécifique 32, cnrs. *STIC* (versão 3, 2003, publié chez Cépadues - Hors-série de la collection Information interaction intelligence), 2003.

CHARNIAK, E.; **Statistical techniques for natural language parsing**. AI magazine, v.18, p.33, 1997.

CHARTON, E.; GAGNON, M.; OZELL, B. Extension d'un système d'étiquetage d'entités nommées en étiqueteur sémantique. In: PROCEEDINGS OF TALN, 2010. Montreal. **Anais...** Montreal, TALN, 2010.

CHO, J.; ROY, S.; ADAMS, R. E. Page quality: In search of an unbiased web ranking. In: PROCEEDINGS OF THE 2005 ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, 2005. **Anais...** [S.l], ACM, 2005, p. 551–562. Disponível em: < <http://ftp.cs.ucla.edu/tech-report/2004-reports/040021.pdf> >. Acessado em: 14 fev. 2013

CLERGERIE, É. V.; RAJMAN, M. Petit panorama des approches en analyse syntaxique. **TAL**, v.44, n.3, p.7-14, 2003. Disponível em: <<ftp://tfalati.inria.fr/INRIA/Projects/Atoll/Eric.Clergerie/TALSyntaxIntro.pdf>>. Acesso em: 23 abr. 2013.

LE-COADCIC, Y.F.; GOMES, M. Y. F. **A Ciência da Informação**. Brasília: Briquet de lemos Livros, 1996.

COHEN, J. Constraint logic programming languages. **Communications of the ACM, ACM**, v. 33, n. 7, p. 52–68, 1990.

COLLINS, A. M.; QUILLIAN, M. R. Retrieval time from semantic memory. **Journal of verbal learning and verbal behavior**, [S.l], v. 8, n. 2, p. 240–247, 1969.

CORCHO, O.; FERNANDEZ-LOPEZ, M.; GOMEZ-PEREZ, A. Ontological engineering: what are ontologies and how can we build them? In: CARDOSO, J. **Semantic web services**: theory, tools and applications. [S.l]: Premier Reference Source, 2007. p.44-70.

DENNING, P. J. Teaching computing science. **Communications of the ACM**, v. 32, n. 12, 1989.

DIAS, E. W. Ensino e pesquisa em Ciência da Informação. **Revista de Ciência da Informação**, Brasília, v. 3, n. 5, 2002.

DIAS, G. A. et al. Representando o conhecimento através de ontologias: o caso do chatterbot lunmi. In: ENCONTRO NACIONAL DE PESQUISA EM CIÊNCIA DA INFORMAÇÃO, 8, 2007. Salvador. **Anais...** Salvador, ENANCIB, 2007, p. 28–31.

DIDIERJEAN, A. Apprendre à partir d'exemples: abstraction de règles et/ou mémoire d'exemplaires? **L'année psychologique**, Paris, v. 101, n. 2, p. 325–348, 2001.

DIENG-KUNTZ, R.; Corby, O. (2005). Conceptual graphs for semantic web applications. In *Conceptual Structures: Common Semantics for Sharing Knowledge* (pp. 19-50). Springer Berlin Heidelberg.

DIENG-KUNTZ, R.; GANDON, F. Ontologies pour le web sémantique et le e-learning. In: _____. **Web sémantique pour le e-Learning**. 2005. Disponível em: < http://www-sop.inria.fr/acacia/afia2005/actes/ActesWSeL_Final.pdf >. Acesso em: 18 ago. 2012

DILLON, A. Information architecture in jasist: just where did we come from? **Journal of the american society for information science and technology**, v. 10, n.53, p.821–823, 2002.

DOMINÉ, C.-H. **Techniques de l'intelligence artificielle**: un guide structuré. [S.l.]: Dunod, 1988.

DRAGULANESCU, N. G. **De nouveaux modèles pour les sciences de l'information?** Bucarest: CIFSIC, 2003.

DUQUE, C. G. **SIRiLiCO**: uma proposta para um sistema de recuperação de informação baseado em teorias da lingüística computacional e ontologia. 2005. Xf. Tese (Doutorado em Ciência da Informação) - Universidade Federal de Minas Gerais. Belo Horizonte, 2005.

ESHGHI, K.; KOWALSKI, R. A. Abduction compared with negation by failure. **ICLP**. [S.l.], 1989. v. 89, p. 234–255.

FEIGENBAUM, E. A.; BARR, A.; COHEN, P. R. **The handbook of artificial intelligence**. New York: Addison-Wesley, 1989.

FENSEL, D. **Spinning the Semantic Web**: bringing the World Wide Web to its full potential. USA: MIT Press, 2005.

FENSEL, D. et al. Semantic web application areas. **NLDB Workshop**. [S.l.: s.n.], 2002.

FERNEDA, E. **Recuperação de informação**: análise sobre a contribuição da ciência de computação para a Ciência da Informação. 147f. Tese (Doutorado) - Escola de Comunicação e Artes da Universidade de São Paulo. São Paulo, 2003.

FERRÉ, S.; HERMANN, A. et al. **Camelis2**: explorer et éditer une base rdf(s) de façon expressive et interactive. 2011. Disponível em: < <http://hal.inria.fr/docs/00/65/83/25/PDF/main.pdf> >. Acesso em: 22 set. 2011.

FLACH, P. A.; KAKAS, A. C. Abductive and inductive reasoning: background and issues. **Abduction and Induction**. [S.l.]: Springer, 2000. p. 1–27.

FONSECA, E. N. **Introdução à biblioteconomia**. Brasília: Briquet de Lemos Livros, 2007.

FOSKETT, A. C. **The Universal Decimal Classification: the history, present status and future prospects of a large general classification scheme**. London: Bingley, 1973.

FOSKETT, D. J. Ciência da Informação como disciplina emergente: implicações educacionais. In: GOMES, Hagar Espanha (Org.). **Ciência da Informação ou Informática?** Rio de Janeiro: Calunga, 1980. Originalmente publicado em 1973.

FREITAS, F. L. G. de. Ontologias ea web semântica. In: JORNADA DE MINI-CURSOS DE INTELIGÊNCIA ARTIFICIAL, 2, 2003. **Anais...** [S.l.]: Sociedade Brasileira de Computação, 2003. p. 1–52.

FÜRST, F.; LECLÈRE, M.; TRICHET, F. Toocom: a tool to operationalize an ontology with the conceptual graph model. In: **EON**. [S.l.: s.n.], 2003.

FÜRST, F.; TRICHET, F. Axiom-based ontology matching. In: THE 3RD INTERNATIONAL CONFERENCE ON KNOWLEDGE CAPTURE, 3, 2005. **Anais...** [S.l.]: ACM, 2005. p. 195–196.

FÜRST, F., TRICHET, F. (2006). Raisonner sur des ontologies lourdes à l'aide de Graphes Conceptuels. In *INFORSID* (pp. 879-894).

GANAPATHY, G., LOURDUSAMY, R. (2011). Matching and merging of ontologies using conceptual graphs. In *Proc. of the World Congress on Engineering WCE*.

GARSHOL, L. M. What are topic maps. Site: *XML.com*. [S.l.: s.n.], 2002.

GIL, G. Two complementary modes of foreign language classroom interaction. **Elt Journal**. Oxford: Univ Press, v. 56, n. 3, p. 273–279, 2002.

GÓMEZ-PÉREZ, A.; CORCHO, O. Ontology languages for the semantic web. **Intelligent Systems**, IEEE, v. 17, n. 1, p. 54–60, 2002.

GRAHAM, T. Conceptions of library and information science: historical, empirical and theoretical perspectives. [S.l: s.n], 1992.

GROSOFF, B. N. et al. Description logic programs: Combining logic programs with description logic. **MIT Sloan School of Management**, n.4437-03, 2003. Disponível em: <<http://ssrn.com/abstract=460986>>. Acesso em: 20 ago. 2012.

GRUBER, T. R. et al. A translation approach to portable ontology specifications. **Knowledge acquisition**, [S.l.], v. 5, n. 2, p. 199–220, 1993.

GUARINO, N. Formal ontology, conceptual analysis and knowledge representation. **International Journal of Human-Computer Studies**, [S.l.], v. 43, n. 5, p. 625–640, 1995.

HAARSLEV, V.; MÖLLER, R. Description logic systems with concrete domains:

Applications for the semantic web. [2003?] Disponível em: <<http://www.sts.tu-harburg.de/people/r.f.moeller/papers/2003/HaMo03b.pdf>>. Acesso em 18 set. 2013.

HAARSLEV, V.; MÖLLER, R. Racer: A core inference engine for the semantic web. **EON**. [S.l.], v. 87, 2003. Disponível em: <http://ftp.franz.com/agraph/racer/Racer_whitepaper.pdf>. Acesso em: 3 fev. 2012.

HAARSLEV, V.; MÖLLER, R. Optimization techniques for retrieving resources described in owl/rdf documents: First results. **KR**, [S.l.], p.163-174, 2004. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.59.4227&rep=rep1&type=pdf>>. Acesso em: 1 abr. 2012.

HARTLEY, R.; COOMBS, M. Abduction in model generative reasoning. In: **AAAI Symposium on Automated Abduction**. [S.l.], p.130-134, 1990. Disponível em:<<http://www.cs.nmsu.edu/~rth/publications/abductionMGR.pdf>>. Acesso em: 3 ago. 2013.

HARTLEY, R. T.; COOMBS, M. J. Conceptual programming: Foundations of problem solving. **CITSEER**, [S.l.], 1990. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.131.1577&rep=rep1&type=pdf>>. Acesso em: 18 ago. 2012

HAWKINS, D. T. Information science abstracts: tracking the literature of information science. part 1: definition and map. **Journal of the American Society for Information Science and Technology**, New York, v. 52, n. 1, p. 44–53, 2001.

HAYES, R. M. Information science education. **ALA World Encyclopedia of Library and Information Science**. Chicago: American Library Association, 1986.

HEFLIN, J. Owl web ontology language-use cases and requirements. **W3C Recommendation**, v. 10, p. 12, 2004.

HEILPRIN, L. B. Foundations of information science reexamined. **Annual review of information science and technology**, v. 24, p. 343–372, 1989.

HEINZLE, R. Uso de ontologias e tratamento da não-monotonicidade nos sistemas de apoio a decisão. **Revista da UNIFEFE**, v. 1, n. 10 jul/dez, 2012.

HENDLER, J.; BERNERS-LEE, T. From the semantic web to social machines: A research challenge for ai on the world wide web. **Artificial Intelligence**, v. 174, n. 2, p. 156–161, 2010.

EPP, M. et al. **Ontology management**: semantic web, semantic web services, and business applications. [S.l.]: Springer, 2007.

HORROCKS, I. et al. Swrl: A semantic web rule language combining owl and ruleml. **W3C Member submission**, v. 21, p. 79, 2004.

KAKAS, A. C.; KOWALSKI, R. A.; TONI, F. Abductive logic programming. **Journal of logic and computation**, Oxford, v. 2, n. 6, p. 719–770, 1992.

KAKAS, A. C.; MANCARELLA, P. Knowledge assimilation and abduction. **Truth Maintenance Systems**, [S.l.], 1991. p. 54–70.

KAYSER, D. La représentation des connaissances. 1997. Disponível em: <<http://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=web&cd=5&ved=0CEkQFjAE&url=http%3A%2F%2Farchives.limsi.fr%2FIndividu%2Fhabert%2FPublications%2FFichiers%2Fhabert99c.doc.gz&ei=9vmPUuafMcfbkQeMvYH4Cw&usg=AFQjCNGC3mGYKwP4tE4sV5oJTZXroGOMNw&sig2=dkF8t9CILAMpfIKPoTSwdg>>. Acesso em: 26 nov. 2011.

KLEER, J. de; REITER, R. Foundations for assumption-based truth maintenance systems: Preliminary report. In: **Proc. American Assoc. for Artificial Intelligence Nat. Conf.** [S.l.], p. 183–188, 1987. Disponível em: <<http://www.dekleer.org/Publications/Foundations%20of%20Assumption-Based%20Truth%20Maintenance%20Systems.pdf>>. Acesso em: 3 abr. 2011.

KOBASHI, N. Y. et al. A função da terminologia na construção do objeto da Ciência da Informação. **IASI**, v. 2, 2001.

KUNZ, C.; BOTSCH, V. Visual representation and contextualization of search results—list and matrix browser. **International Conference on Dublin Core and Metadata Applications**. [S.l.], 2002. Disponível em: <<http://dcpapers.dublincore.org/index.php/pubs/article/viewFile/720/716>>. Acesso em: 7 set. 2011.

LACHER, M. S.; DECKER, S. On the integration of topic maps and rdf data. **SWWS**, Montreal, p.331-344, 2001. p. 331–344. Disponível em: <<http://kennison.name/files/zopestore/uploads/topic-maps/topic-maps-rdf.pdf>>. Acesso em: 4 set. 2011.

LEVADA, A. L. M. Lógica dos predicados. 2012. Disponível em: <http://repositorio.sead.ufscar.br:8080/jspui/bitstream/123456789/1047/1/Fundamentos%20Lógica%20Matemática_Lógica%20dos%20predicados.pdf>. Acesso em: 29 ago. 2012.

LEVESQUE, H. J. A knowledge-level account of abduction. **IJCAI**. [S.l.], p.1061-1067, 1989. Disponível em: <<http://ijcai.org/Past%20Proceedings/IJCAI-89-VOL-2/PDF/034.pdf>>. Acesso em: 7 out. 2011.

LI, W; FENG, Z; LI, Y; XU, Z. Ontology based intelligent information retrieval system. In: **Canadian Conference on Electrical and Computer Engineering**, 1, 2004, Canada. **IEEE**. Canada, v.1, p.373-376, 2004.

MACEDO, F. L. **Arquitetura da Informação**: aspectos epistemológicos, científicos e práticos. 190f. 2005. Dissertação (Dissertação de Mestrado) — Universidade de Brasília, 2005.

MACHLUP, F.; MANSFIELD, U. **The study of information**: interdisciplinary messages. [S.l.]: John Wiley & Sons, Inc., 1983.

MAEDCHE, A. et al. Ontologies for enterprise knowledge management. **Intelligent Systems**, v. 18, n. 2, p. 26–33, 2003.

MANZANO, M. *Lógica, Lógicas y Logicidad*. Lima: Universidad de Salamanca, 2004.

MARCONDES, C. H.; CAMPOS, M. L. d. A. *Ontologia e web semântica: o espaço da pesquisa em Ciência da Informação*. **Ponto de Acesso**, Bahia, v.2, n.1, 2008.

MARTINS, M. **Behavioral reasoning in generalized hidden logics**. 240f. 2005. Tese (Doutorado) - Faculdade de Ciências, Universidade de Lisboa, Lisboa, 2004.

MCBRIDE, B. J. A semantic web toolkit. **Internet Computing**, v. 6, n. 6, p. 55–59, 2002.

MCGARRY, K. **O contexto dinâmico da informação**. Brasília: Brinquet de Lemos, 1999.

MCGUINNESS, D. L.; SILVA, P. Pinheiro da. Explaining answers from the semantic web: the inference web approach. **Web Semantics: Science, Services and Agents on the World Wide Web**, v. 1, n. 4, p. 397–413, 2004.

MILES, A. et al. Skos core: simple knowledge organisation for the web. In: INTERNATIONAL CONFERENCE ON DUBLIN CORE AND METADATA APPLICATIONS. [S.l], p.3, 2005. Disponível em: <http://www.ukoln.ac.uk/nkos/nkos2005/presentations/miles_matthews_wilson.pdf>. Acesso em: 15 ago. 2013.

MINEAU, G. W. The extensional semantics of the conceptual graph formalism. In: **Conceptual Structures: logical, linguistic, and computational issues**. [S.l.]: Springer, 2000. p. 221–234. Disponível em: <http://sd3.uchebalegko.ru/tw_files2/urls_1142/2/d-1716/7z-docs/1.pdf>. Acesso em: 26 fev. 2013.

MINEAU, G. W. A first step toward the knowledge web: Interoperability issues among conceptual graph based software agents part i. **Conceptual Structures: Integration and Interfaces**. [S.l.]: Springer, 2002. p. 250–260. Disponível em <<http://www.glotta.ntua.gr/StateoftheArt/CGs/23930250.pdf>>. Acesso em: 7 out. 2013.

MONTEIRO, F. S. **Web semântica e repositórios digitais educacionais na área de saúde: uma modelagem com foco no objetivo de aprendizagem para refinar resultados de busca**. 199f. 2013. Tese (Doutorado em Ciência da Informação) – Universidade de Brasília, Brasília, 2013.

MORI, A. **Modelagem de conhecimento baseada em ontologias aplicada às políticas públicas de habitação**. 98f. 2009. Dissertação (Mestrado em Ciência da Informação)- Universidade de Brasília, Brasília, 2009.

MORTARI, C. A. **Introdução à lógica**. São Paulo: Unesp, 2001.

MOTIK, B. et al. Owl 2 web ontology language: Structural specification and functional-style syntax. **W3C recommendation**, v. 27, p. 17, 2009.

MOURA, M. A. Informação, ferramentas ontológicas e redes sociais ad hoc: a interoperabilidade na construção de tesouros e ontologias. **Informação & Sociedade: Estudos**, v. 19, n. 1, 2009.

MORVILLE, P.; ROSENFELD, L. **Information architecture for the World Wide Web: Designing large-scale web sites.** [S.l.]: O'Reilly Media, Inc, 2008.

NARDI, D.; BRACHMAN, R. J. An introduction to description logics. **Description Logic Handbook.** [S.l.], p.1-40, 2003. Disponível em: <<http://www.inf.unibz.it/~franconi/dl/course/dlhb/dlhb-01.pdf>>. Acesso em: 15 out. 2010.

NEVELING, U.; WERSIG, G. **Terminologie der information und dokumentation.** [S.l.]: Saur, 1975.

NORMAN, A. D. **The design of everyday things.** USA: MIT Press, 1998.

NORTON, B. **Identity and language learning: Gender, ethnicity and educational change.** [S.l.]: Allyn & Bacon, 2000.

NOY, N. F.; MUSEN, M. A. Algorithm and tool for automated ontology merging and alignment. In: Proceedings of the 17th National Conference on Artificial Intelligence, 17, 2000. **Anais...** [S.l.]: SMI technical report, 2000. Disponível em: <<http://www.aaai.org/Papers/AAAI/2000/AAAI00-069.pdf>>. Acesso em: 9 set. 2011.

OTLET, P. **Traité de documentation: le livre sur le livre, théorie et pratique.** [Paris?]: Editions Mundaneum, 1934.

OWL 2 Web Ontology Language: Conformance (Second Edition) Michael Smith, Ian Horrocks, Markus Krötzsch, Birte Glimm, eds. W3C Recommendation, 11 December 2012, <http://www.w3.org/TR/2012/REC-owl2-conformance-20121211/>. Latest version available at <http://www.w3.org/TR/owl2-conformance/>.

PAGNUCCO, M. **The role of abductive reasoning within the process of belief revision.** 295f. 1996. Tese (Doutorado) — Universidade de Sydney, Austrália, 1996.

PATEL-SCHNEIDER, P. F. Dlp system description. **Description Logics.** [S.l.], 1998. Disponível em: <<http://www.inf.unibz.it/krdp/events/dl-2007/dl-workshops/dl1998/pfps-dlp.pdf>>. Acesso em: 15 set. 2010.

PEIRCE, C. S. **Essays in the Philosophy of Science.** [S.l.]: Liberal Arts Press, 1957.

PENG, Y.; REGGIA, J. A. Plausibility of diagnostic hypotheses: The nature of simplicity. In: **AAAI.** 1986. v.86, p. 140–145. Disponível em: <<http://aaai.org/Papers/AAAI/1986/AAAI86-022.pdf>>. Acesso em: 25 ago. 2011.

PEPPER, S.; MOORE, G. Topic maps. **Encyclopedia of Library and Information Sciences,** Citeseer, p. 1–19, 2010.

PINHEIRO, L. V. R.; LOUREIRO, J. M. M. Traçados e limites da ciência da informação. **Ciência da informação,** Brasília, v. 24, n. 1, 1995.

POPPER, K. R. Le mythe du cadre de référence. In: BOUVERESSE R. **Karl Popper et la science d'aujourd'hui:** Colloque de Cerisy. Paris: Aubier. 1989. p. 11–44.

PY, M. X. Sistemas especialistas: uma introdução. **Instituto de Informática.** Rio Grande do

Sul: UFRGS, 2009.

RAGGIO, A. et al. A evolução da noção de sistema axiomático [l'évolution de la notion de système axiomatique]. **Philosophos-Revista de Filosofia**, v. 8, n. 1, 2008.

RAMELLI, A. **Le diverse et artificiose machine del capitano Agostino Ramelli**. [S.l.]: Gregg International, 1970.

RAMOS, C.; AUGUSTO, J. C.; SHAPIRO, D. Ambient intelligence: the next step for artificial intelligence. **Intelligent Systems**, v. 23, n. 2, p. 15–18, 2008.

RAYWARD, W. B. Visions of xanadu: Paul otlet (1868-1944) and hypertext. **JASIS**, v. 45, n. 4, p. 235–250, 1994.

REHMAN, S. U.; CHAUDHRY, A. S. Km education in lis programs. **Education for Information**, v. 23, n. 4, p. 245–258, 2005.

RIFFITH, B. C. **Key papers in information science**. [S.l.]: American Society for Information Science, 1980.

RIPOLL, T.; COULON, D. Le raisonnement par analogie: une analyse descriptive et critique des modèles du mapping. **L'année psychologique**, Paris, v. 101, n. 2, p. 289–323, 2001.

ROBREDO, J. **Da Ciencia Da Informacao Revisitada...** Brasília: Thesaurus Editora, 2003.

ROBREDO, J. **Documentação de hoje e de amanhã: uma abordagem revisitada e contemporânea da Ciência da Informação e de suas aplicações biblioteconômicas, documentárias, arquivísticas e museológicas**. Brasília: Edição de autor, 2005.

ROBREDO, J. Sobre arquitetura da informação. **Revista Ibero-Americana de Ciência da Informação**, v. 1, n. 2, 2010.

SÁNCHEZ, J. A. P. **Tecnologías de la web semántica**. [S.l.]: Editorial UOC, 2011.

SANTOS, P.L.V.A.C; ALVES, R.C.V. Metadados e Web Semântica para estruturação da Web 2.0 e Web 3.0. **PBCIB**, [S.l.], v.4, n.2, 2011.

SARACEVIC, T. Evaluation of evaluation in information retrieval. In: PROCEEDINGS OF THE 18TH ANNUAL INTERNATIONAL ACM SIGIR CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL. 18. **Anais...** [S.l.], ACM, p. 138–146, 1995. Disponível em: <<http://comminfo.rutgers.edu/~muresan/IR/Docs/Articles/sigirSaracevic1995.pdf>>. Acesso em: 5 abr. 2013.

SARACEVIC, T. Ciência da Informação: origem, evolução e relações. **Perspectivas em Ciência da Informação**, v.1, n. 1, p.41-62, 1996.

SARACEVIC, T. Relevance reconsidered. In: PROCEEDINGS OF THE SECOND CONFERENCE ON CONCEPTIONS OF LIBRARY AND INFORMATION SCIENCE, 2,

1996, Copenhagen. **Anais...** Copenhagen: CoLIS, 1996. p. 201–218. Disponível em: <http://scholar.google.fr/scholar?q=Relevance+reconsidered.+In%3A+Proceedings+of+the+second+conference+on+conceptions+of+library+and+information+science&btnG=&hl=fr&as_sdt=0%2C5>. Acesso em: 2 jun. 2012.

SARACEVIC, T. Information science. **Journal of the American Society for Information Science**, v. 50, n. 12, p. 1051–1063, 1999. Disponível em: <<http://www3.interscience.wiley.com/cgi-bin/abstract/69500813/ABSTRACT>>. Acessado em: 22 mar. 2013.

SARACEVIC, T.; REES, A. M. Towards the identification and control of variables in information retrieval experimentation. **Journal of Documentation**, v. 23, n. 1, p. 7–19, 1967.

SCHMOLZE, J. G.; LIPKIS, T. A. Classification in the kl-one knowledge representation system. In: **IJCAI**, v. 1, p. 330–331, 1983.

SCHREIBER, G. Knowledge engineering. **Foundations of Artificial Intelligence**, v.3, p. 929–946, 2008.

SEBEN, A. et al. **Uma lógica para a referência ambígua**. 182f. 2012. Dissertação (mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2012.

SHANAHAN, M. Prediction is deduction but explanation is abduction. In: **IJCAI**, v.89, p. 1055–1060, 1989.

SHERA, J. H.; CLEVELAND, D. B. History and foundations of information science. **Annual review of information science and technology**, v. 12, n. 12, p. 249–275, 1977.

SILVA, T. D. M. S. et al. Extração de informação para busca semântica na web baseada em ontologias. 79f. 2012. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2012.

SIQUEIRA, A. H. A lógica e a linguagem como fundamentos da arquitetura da informação. 141f. 2010. Dissertação (Mestrado em Ciência da Informação e Documentação) – Universidade de Brasília, 2010.

SMITH, M. et al. **Web ontology language (owl) guide version 1.0**. World Wide Web Consortium (W3C) Working Draft. Disponível em: <<http://www.w3.org/TR/owl-guide>>, Acesso em: 20 jun. 2012.

SMITH, M.; WELTY, C.; MCGUINNESS, D. L. **Owl web ontology language guide. w3c recommendation** (2004). Disponível em: <<http://www.w3.org/TR/owl-guide>>. Acesso em: 20 jun. 2012.

SOUZA, R. R.; ALVARENGA, L. A web semântica e suas contribuições para a Ciência da Informação. **Ciência da Informação**, Brasília, SciELO Brasil, v. 33, n. 1, p. 132–141, 2004.

SOWA, J. F. **Conceptual Structures: Information Processing in Mind and Machine**.

Boston: Addison-Wesley, 1984.

SOWA, J. F. et al. **Knowledge representation: logical, philosophical, and computational foundations**. [S.l.]: MIT Press, 2000.

STENNING, K.; COX, R.; OBERLANDER, J. Contrasting the cognitive effects of graphical and sentential logic teaching: reasoning, representation and individual differences. **Language and Cognitive Processes**, v. 10, n. 3-4, p. 333–354, 1995.

STUDER, R.; GRIMM, S.; ABECKER, A. **Semantic web services**. [S.l.]: Springer, 2007.

TAYLOR, R. S. The information sciences. **Library Journal**, v. 88, n. 19, p. 4161–4163, 1963.

UDREA, O.; RECUPERO, D. R.; SUBRAHMANIAN, V. Annotated rdf. **ACM Transactions on Computational Logic**, v. 11, n. 2, p. 10, 2010.

VIEIRA, R., SANTOS, D. A.; SILVA, D. M.; SANTANA, M. R. Web semântica: Ontologias, lógica de descrições e inferências. **Web e Multimídia: Desafios e Soluções**, v. 1, p. 127–165, 2005. Disponível em: <<https://wiki.dcc.ufba.br/pub/Residencia/MaterialModuloTI5/webmedia-webs.pdf>>. Acesso em: 20 nov. 2010.

VILLALOBOS, A. P. de O.; SILVA, D. C. As potencialidades da web semântica para a Ciência da Informação. **PBCIB**, v. 6, n. 1, 2012.

WELLS, H. World brain: the idea of a permanent world encyclopedia. **Contribution to the new Encyclopédie Française**, 1937.

WIELINGA, B. et al. Framework and formalism for expressing ontologies. **ESPRIT project**, v. 8145, 1994.

WOLSTENCROFT, J. Restructuring, reminding and repair: What's missing from models of analogy. **AI Communications**, v. 2, n. 2, p. 58–71, 1989.

WU, Z. et al. **Database-based inference engine for RDFS/OWL constructs**. [S.l.]: Google Patents, ago. 2008. US Patent App. 12/188,267.

WURMAN, R. S. **Information anxiety: What to do when information doesn't tell you what you want to know**. New York: Bantam Books, 1989.

WURMAN, R. S. et al. **Information anxiety 2**. Indianapolis: IN, 2001.

WURMAN, R. S. **Ansiedade de informação 2: um guia para quem comunica e dá instruções**. São Paulo: Editora de Cultura, 2005.

ZAIHRAYEU, I.; SILVA, P. P. D.; MCGUINNESS, D. L. Iwtrust: Improving user trust in answers from the web. **Trust Management**. [S.l.]: Springer, 2005. p. 384–392.

ZHONG, J. et al. Conceptual graph matching for semantic search. **Conceptual Structures:**

Integration and Interfaces. [S.l.]: Springer, 2002. p. 92–106.

ZOU, Y.; FININ, T.; CHEN, H. F-owl: An inference engine for semantic web.

Formal Approaches to Agent-Based Systems. [S.l.]: Springer, 2005. p. 238–248.

ZWIES, R. Observations on the american society for information science summit 2000 meeting: defining information architecture. **Bulletin of the American Society for Information Science and Technology**, [S.l.]: Wiley Online Library, v. 26, n. 5, p. 10–12, 2000.