

Pontifícia Universidade Católica de São Paulo

PUC-SP

Luciano Gonzaga Leme

***Site com Conteúdo Adequado à Web*
Semântica e Mecanismos de Busca**

Mestrado em

Tecnologias da Inteligência e *Design* Digital

São Paulo

2009

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

Pontifícia Universidade Católica de São Paulo

PUC-SP

Luciano Gonzaga Leme

Site com Conteúdo Adequado à Web
Semântica e Mecanismos de Busca

Mestrado em

Tecnologias da Inteligência e *Design Digital*

Dissertação apresentada à Banca Examinadora da Pontifícia Universidade Católica de São Paulo, como exigência parcial para obtenção do título de Mestre em Tecnologias da Inteligência e Design Digital - Inteligência Coletiva e Ambientes Interativos, sob a orientação do Professor Doutor Demi Getschko.

São Paulo

2009

Banca Examinadora

Ao meu orientador: Professor Doutor Demi Getschko
pelo apoio e auxílio para a realização deste trabalho.

Resumo

Leme, Luciano Gonzaga – *Site com Conteúdo Adequado à Web Semântica e Mecanismos de Busca* – São Paulo, 2009.

Esta dissertação apresenta uma abordagem que tem como objetivo principal questionar e apresentar possíveis caminhos para que o conteúdo de um web *site* seja adequadamente oferecido em termos de Web Semântica e Mecanismos de Busca. Para alcançar esse objetivo foram efetuadas pesquisas e utilizada a experiência do autor. Desse modo foram combinados os conceitos e as tecnologias envolvidas em relação a metadados, ontologias, serviços web semânticos e otimização de mecanismos de busca.

Os metadados permitem qualificar informações adequadamente, as ontologias são empregadas para aprimorar a análise do domínio do problema e para obter componentes de software com descrições semânticas, os quais podem ser reutilizados numa grande variedade de aplicações. Os serviços web semânticos são utilizados como componentes de software distribuídos pela Internet e podem realizar tarefas complexas de aplicações.

Partindo do questionamento do que publicar em um web *site*, ou melhor, como deve ser o conteúdo de um web *site* - foram expostos conhecimentos e tecnologias que delimitam a questão: como oferecer conteúdo adequado?

Após a exposição dos conceitos associados e da apresentação de possíveis soluções concluímos com questionamentos e indicativos de possíveis pesquisas que muito contribuirão ao desenvolvimento de *sites* com conteúdo adequado com a Web Semântica e seus possíveis efeitos colaterais positivos, propiciando a melhor utilização do conhecimento e das informações disponibilizadas na web.

Palavras-chave: conteúdo adequado na web, Web Semântica, Metadados, Ontologias, Agentes Semânticos, Serviços Web Semânticos, SEO.

Abstract

Leme, Luciano Gonzaga – **Site with Appropriated Content to Semantic Web and Search Engines** – Sao Paulo, 2009.

This work introduces an approach which main purpose is to ask and present possible ways so that the web site's content will be appropriately offered related to Semantic Web and Search Engines. To achieve this purpose many researches were made and used the author's experience in this area. Therefore many concepts and technologies were combined and used, such as: metadata, ontologies, semantic web services and search engine optimization.

Metadata allow qualifying informations appropriately, ontologies are used to improve domain's analysis problem and to obtain software components with semantic descriptions, which may be reused in a great variety of applications. Semantic web services may be used as software components spread through the Internet to perform applications' complex tasks.

Starting asking what should be published in a web site, or even better, how should be a web site's content – the work exposes knowledges and technologies to answer the question: how to offer appropriated content?

After exposing the associated concepts and showing possible solutions a conclusion is achieved with questions and possible researches which will allow appropriated content sites development with Semantic Web and its possible positive collateral effects, granting improvement in both: knowledge utilization and available informations in the web.

Keywords: appropriated content in the web, Semantic Web, Metadata, Ontologies, Semantic Agents, Semantic Web Services, SEO.

Sumário

I – Introdução	9
II – Conteúdo de um <i>site</i>	11
II.01 - Definindo o conteúdo de um <i>site</i>	11
II.02 – Interação com o usuário.....	23
II.03 – A utilização da Web Semântica	25
II.03.01 – Web Semântica versus Inteligência Artificial	27
II.03.02 – Metadados.....	31
II.03.02.01 – Padrão Dublin Core	34
II.03.02.02 – Resource Description Framework - RDF.....	36
II.03.02.03 – Críticas aos Metadados	40
II.03.03 – Ontologias	44
II.03.03.01 – Linguagens para Representação de Ontologias.....	47
II.03.03.02 – Principais Ontologias	54
II.03.03.03 – Criação de Ontologias.....	59
II.03.03.03.01 – Seleção dos Termos referentes ao domínio da ontologia:.....	59
II.03.03.03.02 – Definir as classes e a hierarquia de classes da ontologia:.....	59
II.03.03.03.03 – Definir as propriedades:.....	59
II.03.03.03.04 – Definir as restrições:.....	60
II.03.03.03.05 – Definir instâncias:	60
II.03.04 – Ferramentas.....	61
II.03.04.01 – Principais Ferramentas.....	62
II.03.04.01.01 - OilEd:	62
II.03.04.01.02 - OilViz:	62

II.03.04.01.03 - Protégé:	62
II.03.04.01.04 - DC.dot:.....	63
II.03.04.01.05 - API JENA:.....	64
II.03.04.01.06 - Chimaera:	65
II.03.04.01.07 - SNOBASE:.....	65
II.03.04.01.08 - FaCT:.....	66
II.03.04.01.09 - Pellet:	67
II.03.05 – Agentes.....	68
II.03.05.01 – Principais Agentes	69
II.03.05.01.01 - CATO (Componente para Alinhamento Taxônomico de Ontologias):.....	69
II.03.05.01.02 - API Jena:	69
II.03.05.01.03 - Serviços Web Semânticos.....	69
II.04 – Como organizar o conteúdo para que os mecanismos de busca sejam mais eficientes	74
III – Conclusão	77
Notas	80
Glossário	81
Referências Bibliográficas.....	82
Impressas:.....	82
<i>Sites</i> da Internet:.....	86
Anexos.....	Erro! Indicador não definido.
Anexo 01 - Exemplo de Arquivo utilizando SHOE:	Erro! Indicador não definido.
Anexo 02 – Tabela Comparativa entre as linguagens OIL e DAML: ...	Erro! Indicador não definido.

I – Introdução

Atualmente existe uma grande quantidade de dados que demandam o nosso tempo. Dentro dessa enxurrada de dados, quais são os que têm relevância aos nossos interesses? As empresas de conteúdo (Abril, BBC, CNN, Globo, iG, Reuters, UOL dentre muitas) demandam o nosso tempo e nem sempre atendem nossas intenções. Ao mesmo tempo em que estar bem informado passou a ser uma condição “*sine qua non*” para nossa integração social e ampliarmos nossas perspectivas de negócios, como escolher o conteúdo que mais nos agrada e que pode nos trazer melhores perspectivas numa sociedade que nos bombardeia a todo o tempo com dados que não são importantes e que nos tomam o pouco tempo que poderíamos despender com atividades mais agradáveis? Outro problema é o lado do comunicador (que será o foco deste trabalho), qual a melhor forma de atingir um público que lhe trará bons retornos, venda de espaço propagandístico, venda de assinaturas etc. – a mídia quer atingir o público em geral, de preferência com eficiência (fazer certo as coisas) e eficácia (fazer as coisas certas). Como a mídia pode oferecer conteúdo relevante com interação inteligente para atender demandas variáveis? Esse é o desafio: *site* ou ambiente Web com conteúdo apropriado.

Segundo Guevara: “As interfaces dos atuais sistemas necessitam não somente responder às solicitações dos usuários, mas também antecipar, adaptar e buscar ativamente maneiras de lhes dar suporte.”

Pretende-se, com o trabalho, descrever ou caracterizar as principais opções que atendam ao objetivo geral da pesquisa que é o de criar mecanismos cognitivos, simbólicos, interativos e inteligentes para atender às demandas de informação relevante. Hoje em dia, um usuário faz uma busca a respeito de um livro de um determinado gênero e pode aparecer no resultado da pesquisa um *link* para um *site* de *e-commerce* que oferece ao usuário sugestões do gênero daquele livro que não condizem com o que o usuário necessita ou deseja (talvez ele não queira pagar frete ou gostaria de obter alguma promoção etc.). Dessa forma, o trabalho visa estudar como oferecer, através de mecanismos inteligentes, conteúdo adequado, com Web

Semântica apropriada, de acordo com o que houver sido identificado ou solicitado pelo usuário.

Um trabalho que apresente opções aos desafios de divulgação de informação e comunicação apresenta-se como fonte inesgotável de soluções que poderão ser adotadas para que as empresas façam dinheiro que é o fator mais atrativo no mundo empresarial.

Iniciaremos nosso trabalho definindo o que seria um *site* de conteúdo e as dificuldades de alcançar diferentes idiomas, assim como diferentes culturas. Apresentaremos quais as possíveis soluções para que o problema possa ser contornado e quais as técnicas que podem ser aplicadas até apresentarmos uma conclusão do que pode ser feito para atingir o público de forma adequada através de *sites* de conteúdo.

II – Conteúdo de um *site*

II.01 - Definindo o conteúdo de um *site*

Num *site* de conteúdo, uma das grandes questões é como disponibilizar e permitir a atualização de conteúdo a ser visto pelo usuário. Haveria um modo de disponibilizar informação relevante e que atinja aos mais variados públicos de forma que o que for enviado ao internauta seja relevante e que atinja a culturas, crenças e condições sociais diversas? Ou seja, teríamos como fornecer a informação relevante e que interessa a cada usuário em específico?

Inicialmente vamos definir o que venha a ser um *site* de conteúdo: é todo e qualquer *site* que não tem como objetivo fim apresentar uma ferramenta ou utilitário para o usuário. Webmails, *internet banking* ou *sites* de busca são exatamente o oposto dos *sites* de conteúdo, já que têm como objetivo fim arrolar e-mails, facilitar o acesso do cliente a sua conta corrente ou fazer uma pesquisa na Internet, respectivamente. Os *sites* de conteúdo podem ser categorizados em um grande número, dada a natureza do que se propõe a publicar. Como exemplos há *sites* de conteúdo infantil, esportivo, econômico, noticiário, opiniões e muito mais. *Sites* puramente institucionais também são considerados como de conteúdo.

Mas para abordar à dificuldade acima: fornecer informação adequada ao perfil específico de cada usuário em um *site* de conteúdo, vários aspectos precisam ser analisados, tais como: qual o público que se deseja atingir? Se a resposta for a todos, já começamos com o problema da língua e para abordarmos exatamente esse ponto: o linguístico - consideraremos inicialmente o trabalho realizado por Raimundo Lúlio retratado por Umberto Eco no livro “A busca da língua perfeita”, que tentou obter uma língua que atendesse às culturas cristã, hebraica e muçulmana, ou seja, escritos que fossem adequados as três culturas. Ramón Lull, em português Raimundo Lúlio, foi um catalão nascido em Maiorca¹ entre aproximadamente 1232 e 1316 – seu local

¹ Maiorca (em catalão e castelhano Mallorca) é a maior ilha do arquipélago das Ilhas Baleares localizado a leste da Espanha e sua maior cidade e capital é Palma de Maiorca.

de nascimento era encruzilhada das culturas cristã, islâmica e hebraica. Após uma juventude mundana, Lúlio teve uma experiência ou crise mística (Umberto Eco não entra em maiores detalhes) e entrou para a Ordem Franciscana.

Raimundo Lúlio foi o primeiro filósofo a escrever obras doutrinárias em língua vernácula² e em sua obra *Ars magna* desenvolveu uma língua filosófica que seria, segundo Lúlio, perfeita – mediante a qual seria possível converter os infiéis. Para Lúlio, os infiéis seriam todos os que não fossem cristãos. Havia em sua *Ars magna* uma forte tendência à crença e doutrina católica, que em plena Idade Média possuía fortíssima influência no mundo ocidental. A língua defendida por Lúlio ambicionava ser universal, porque universal é a combinação matemática que articula o seu plano de expressão. Lúlio elabora no nível de conteúdo o seu sistema de ideias que deve ser universal e comum a todos os povos. Faz uso de letras alfabéticas e de figuras para alcançar seus objetivos de universalização de uma língua que poderia ser considerada perfeita para a finalidade a qual Lúlio se propunha: converter muçulmanos e judeus ao catolicismo.

Aqui cabem parênteses: não estamos ambicionando inventar uma nova língua a ser colocada em *sites* de conteúdo e que possa ser considerada universal, mas a ideia que Lúlio tinha era a de uma língua universal, perfeita e que permitiria atingir culturas diversas, então a referência ao trabalho de Lúlio nesta dissertação é para que possamos indagar se podemos abordar a questão através de um mecanismo de língua que permita a comunicação única, mas que seja adequada a crenças e culturas diversas, de modo a permitir um conteúdo universal adequado a perfis diversos.

Contemporâneo a Lúlio tivemos um franciscano, Roger Vacona, que não queria inventar uma língua nova, nem divulgar o conhecimento das línguas dos outros, queria era a supremacia da cultura cristã ocidental em detrimento às demais culturas ou tradições. Pensamentos franciscanos predominantes na época eram: caráter missionário e utopia universalista – defendidos por São Francisco, Vacona, e, em partes, pelo próprio Lúlio.

² Em contraste com a língua litúrgica.

Também não é intuito deste trabalho, impor pesos ou valores a conteúdos específicos, para que os mesmos sejam considerados os verdadeiros. Partimos do pressuposto de que o conteúdo pode ser bastante diversificado e que não existe conteúdo melhor do que outro, apenas temos conteúdo que pode atender aos interesses de um usuário e que não atenderia aos interesses de outro usuário.

Utilizando-se de elementos de arte combinatória, se queremos dispor n elementos t a t em que a ordem tenha valor diferencial, temos a seguinte fórmula: $n!/(n-t)!$ (fatorial de n dividido por fatorial de $(n-t)$)³ como sendo a possível quantidade de resultados. Exemplo disso é: dadas quatro pessoas, como podemos dispô-las em duplas nos assentos de um teatro. No entanto, se quisermos combinar n elementos t a t em que a ordem não altere o resultado, temos: $n!/t(n-t)!$ (fatorial de n dividido pelo produto de t e fatorial de $(n-t)$). Exemplo deste último é: dados quatro soldados (A, B, C e D) como podemos combiná-los 2 a 2 (dois a dois) para realizarem uma patrulha, enviar A e B ou B e A é a mesma dupla para efetuar a patrulha, ou seja, a ordem não alteraria o resultado.

Numa abordagem semiótica, sistema de expressão (símbolos) e de regras sintáticas (n elementos podem combinar-se t a t , onde t pode coincidir com n) que sejam capazes de revelar automaticamente possíveis sistemas de conteúdo teremos, segundo Umberto Eco, 21 (vinte e uma) letras do alfabeto – quantidade de letras condizente ao alfabeto utilizado à época de Raimundo Lúlio, século XIII – que podem resultar em mais de 51 (cinquenta e um) bilhões de bilhões de sequências de 21 (vinte e uma) letras – número astronomicamente alto que traduz as diversas disposições de 21 (vinte e uma) letras em todas as possíveis quantidades de disposições. O número alcançado não contempla critérios ou restrições que não dizem respeito à combinatória, pois critérios ou restrições vêm do exterior para limitar as possibilidades matemáticas - exemplo é uma palavra formada apenas por

³ Na matemática, o fatorial de um número natural n é o produto de todos os inteiros positivos menores ou iguais a n . Isso é escrito como $n!$ e lido como "fatorial de n ". A notação $n!$ foi introduzida por Christian Kramp em 1808.

vogais ou consoantes que nada possa representar ou foneticamente inviável e que teria que ser descartada.

Lúlio, em sua *Ars magna*, montou um dispositivo com um alfabeto de 9 (nove) letras de B a K e 4 (quatro) figuras e uma *Tabula generalis* - veja Figura 01 - que é uma lista de 6 (seis) conjuntos de 9 (nove) entidades cada um, onde temos: 9 (nove) de cada um dos seguintes itens: Princípios Absolutos, Princípios Relativos, Tipos de Questões, Sujeitos, Virtudes e Vícios, que formariam uma espécie de quadro das dignidades, segundo Lúlio.

Quadro das Dignidades						
	Principia Absoluta	Principia Relativa	Questiones	Subjecta	Virtutes	Vitia
B	Bonitas	Differentia	Utrum?	Deus	Iustitia	Avaritia
C	Magnitudo	Concordantia	Quid?	Angelus	Prudentia	Gula
D	Aeternitas	Contrarietas	De quo?	Coelum	Fortitudo	Luxuria
E	Potestas	Principium	Quare?	Homo	Temperantia	Superbia
F	Sapientia	Medium	Quantum?	Imaginatio	Fides	Acidia
G	Voluntas	Finis	Quale?	Sensitiva	Spes	Invidia
H	Virtus	Majoritas	Quando?	Vegetativa	Charitas	Ira
I	Veritas	Aequalitas	Ubi?	Elementativa	Patientia	Mendacium
K	Gloria	Minoritas	Quomodo? Cum quo?	Instrumentativa	Pietas	Inconstantia

Figura 01 – *Tabula Generalis*

Com relação à Figura 02, observamos que ela permite combinações possíveis que podem unir os princípios da *Tabula Generalis* a predicados, construindo afirmações do tipo “A Bondade é grande”. E teremos 36 (trinta e seis) linhas com 72 (setenta e duas) combinações. As combinações devem permitir silogismos regulares e sendo assim estarão descartadas afirmações do tipo: “A Bondade é boa”.

assim como de algumas modalidades de argumentação que as regras estabelecem, alheias às leis da combinatória.

BC	CD	DE	EF	FG	GH	HI	IK
BD	CE	DF	EG	FH	GI	HK	
BE	CF	DG	EH	FI	GK		
BF	CG	DH	EI	FK			
BG	CH	DI	FK				
BH	CI	DK					
BI	CK						
BK							

Figura 04 – Combinações entre as letras

Na Figura 05, há tripletas geradas por 9 (nove) elementos. É utilizada a roda, provavelmente devido a tradições cabalísticas, sendo que Lúlio transforma as tripletas em quádruplas, inserindo a letra T. Desse modo, é utilizado um artifício mnemônico em que as letras que precedem a letra T devem ser lidas como princípios ou dignidades, ao passo que aquelas que seguem a T devem ser lidas como princípios relativos. No caso, podemos questionar se todas as milhares de quádruplas levam a uma argumentação válida. Sendo que para Lúlio, devem ser deduzidas somente aquelas fórmulas cujas premissas e conclusões correspondem à disposição real do cosmos. Ele aceita algumas conversões das proposições, mas não outras, que também seriam formalmente corretas.

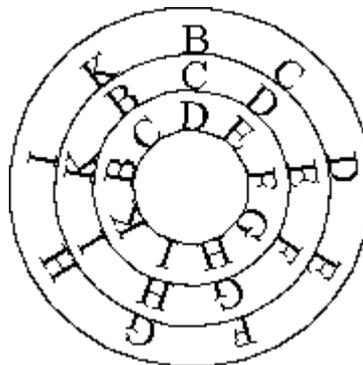


Figura 05 - Tripletas

Não é fácil de entender à primeira vista como era a formação dessa língua universal luliana. Mas já podemos afirmar que a *Ars magna* não é um instrumento

lógico, mas um instrumento dialético, isto é, uma maneira de identificar e relembrar todas as formas válidas para argumentar a favor de uma tese pré-construída. E isso de tal maneira que não há quádrupla, segundo Lúlio, que, devidamente interpretada, não possa resolver a questão à qual é aplicada.

A *Ars*, segundo Lúlio, levaria a conclusões mais seguras do que as da lógica, “e, portanto, o artista desta arte pode aprender mais em um mês do que possa aprender um lógico em um ano” (*Ars magna, Decima pars*, cap. 101).

Leibniz⁴, em 1666, questiona por que Lúlio teria limitado o número de elementos, por que o número de princípios não teria ficado aberto? O motivo é que Lúlio não pensava, de modo algum, em uma combinatória livre de elementos de expressão não vinculados a um conteúdo específico, pois se assim fosse, não consideraria a sua arte como uma língua perfeita, capaz de assumir uma realidade divina que ele assume, desde o início, como absoluta, auto-evidente e revelada por Deus. Ele a concebia como um instrumento para converter os infieis, e para tanto estudara durante longo tempo as doutrinas tanto dos judeus quanto dos muçulmanos.

Lúlio realizou outros trabalhos, além da *Ars magna* – um deles, *A arbor scientiarum* (A árvore das ciências) apresenta noções elementares que seriam comuns também aos infieis. Tais princípios elementares se inserem em um sistema fechado e pré-definido, ou seja, um sistema já rigidamente hierarquizado, o sistema das Árvores das Ciências.

Se no silogismo aristotélico afirmamos: “todas as flores são vegetais, X é uma flor, logo X é um vegetal” - do ponto de vista lógico, o que é X é irrelevante. Mas Lúlio quer saber se X é uma margarida ou um barco. E em sua *arbor scientiarum*, Lúlio pretende construir um instrumento capaz de enfrentar toda a enciclopédia do saber.

Na *arbor scientiarum*, a combinatória luliana é um instrumento retórico mediante o qual se quer demonstrar o que já é conhecido, isto é, aquilo que a

⁴ Gottfried Wilhelm von Leibniz (Leipzig, 1646 – Hanover, 1716) foi um filósofo, cientista, matemático, diplomata e bibliotecário alemão.

estrutura de uma floresta de várias árvores já fixou, de uma vez para sempre, e que nenhuma combinação jamais poderá subverter, segundo Lúlio.

De qualquer forma, a *Ars* poderia aspirar a ser língua perfeita se o já-conhecido que visava comunicar tivesse pertencido realmente a um universo de conteúdo igual a todos os povos. Ela foi traduzida para o árabe e o hebraico, mas ela fala do produto de uma organização do mundo realizado pela tradição cristã ocidental.

Umberto Eco relata:

Lúlio houvera feito da própria reflexão teológica, e da própria investigação de uma língua universal, um meio para lançar uma ponte intelectual e religiosa entre Ocidente europeu e Oriente, e que julgava que a verdadeira autoridade não se devia basear em uma unidade inflexível, mas em uma tensão entre vários centros – de forma que Moisés, Cristo e Maomé pudessem levar a um resultado unitário. O lulismo é acolhido como estímulo místico e filosófico, e como alternativa imaginária e poética para a enciclopédia do aristotelismo escolástico, mas também como inspiração política.

Esse apelo concordatário de Lúlio é retomado dois séculos mais tarde por Nicolau Cusano⁵ que em seu tratado *De pace fidei* causa uma polêmica e tenta um diálogo com os muçulmanos e judeus e se coloca o problema (luliano) de como demonstrar aos representantes de outras duas religiões monoteístas que eles devem concordar com a verdade cristã.

No século XVI, outro que tentou a empreitada de Lúlio foi Giordano Bruno. Ele entendia que a língua perfeita deveria oferecer a chave para exprimir não só este mundo, mas todos os infinitos na sua concordância mútua. Essa ambição parece imperfeita no que concerne à estrutura semiótica: tratava-se de um léxico imenso, com significados vagos, e cuja sintaxe, quando muito, era de uma combinação sem limites.

⁵ Nicolau Cusano (1401-1464), matemático e astrônomo.

A retórica de Giordano Bruno pretendia reforma, renovação e revolução dos costumes, dos conhecimentos e da própria ordem política da Europa. Desenvolvendo os conceitos lulistas, Bruno criou uma estrutura de rodas concêntricas, limitação de letras, com 23 (vinte e três) letras do alfabeto mais caracteres gregos e hebraicos. A sua combinatória era ávida pelo ilimitado, diferenciando-se da combinatória restrita de Raimundo Lúlio.

E questionamos: podemos ter uma máquina para gerar todas as línguas possíveis ou algum conteúdo automático? Dentre outros, deparamo-nos com o problema das combinações válidas. Exemplo: quente, frio, seco e úmido – essa combinação pode formar uma sentença sem o devido significado.

O número de palavras (de comprimentos variáveis de duas letras até 23) é mais de setenta mil bilhões de bilhões⁶. O problema de uma língua universal, na medida em que poderia conter potencialmente todas as línguas possíveis é o de apresentar números astronômicos. E desse modo não é de se surpreender que existam tantas línguas diferentes.

Para Descartes⁷, Lúlio é um autor que tentou sem êxito a criação de um método que pode ser considerado um pseudométodo, pois suas regras não são claras e a determinação das coisas a conhecer é insuficiente, e menciona a promessa errônea de oferecer conhecimentos gerais em pouco tempo.

Com o advento da informática, programas de computador⁸ que mostram o funcionamento da *Ars magna*, assim como inúmeras comunidades na Internet que estudam e comentam a respeito dos trabalhos de Lúlio são uma realidade.

⁶ Não é calculado mais a respeito de conceitos (como fazia Lúlio), e sim sobre sequências alfabéticas, meros elementos da expressão, não controlados por nenhuma ortodoxia que não seja a do número.

⁷ René Descartes (1596, França – 1650, Suécia), filósofo, físico e matemático. Notabilizou-se, sobretudo, pelo seu trabalho revolucionário da Filosofia, tendo também inventado o sistema de coordenadas cartesiano, que influenciou o desenvolvimento do Cálculo contemporâneo.

⁸ <http://lullianarts.net/downloads.htm>; <http://www.ramonlull.net>.

Abordando a questão voltando-se à Inteligência Artificial (IA) temos que, segundo George F. Luger, a maioria das pessoas - quando são questionadas sobre quais habilidades são essencialmente mais humanas e mais difíceis de serem computadorizadas, além da criação artística, da tomada de decisão ética e da responsabilidade social – menciona linguagem e aprendizado. Ao longo dos anos, estas duas áreas têm funcionado como objetivo, desafio e meio de teste para o progresso da IA.

George F. Luger relata:

Uma das razões do porquê linguagem e aprendizado são áreas de pesquisa difíceis, embora importantes, é que elas englobam muitas outras habilidades inteligentes humanas. Comunicar-se através de linguagem natural, quer seja como texto ou como um ato de fala, depende enormemente do nosso conhecimento e expectativas dentro do domínio do discurso. A compreensão de linguagem não é meramente a transmissão de palavras: ela também requer inferências sobre o objetivo, conhecimento e suposições do locutor, bem como sobre o contexto da interação. A implementação de um programa para compreender linguagem natural requer que representemos conhecimento e expectativas do domínio e raciocinemos efetivamente sobre eles. Precisamos considerar questões como não-monotonia, revisão de crença, metáfora, planejamento, aprendizado e as complexidades práticas da interação humana.

E aí voltamos à questão principal sobre como disponibilizar informação relevante e que atinja aos mais variados públicos de forma que o que for enviado ao internauta seja relevante e que atinja a culturas, crenças e condições sociais diversas.

Vemos que se a abordagem for ao âmbito luliano, ou seja, restringirmos o que deve ser dito a uma verdade absoluta e imutável, estaremos divulgando dogmas e crenças e a probabilidade de atingir corretamente pessoas com valores culturais diversificados será extremamente pequena. A retórica revolucionária de Giordano Bruno, no entanto, com suas infinitas possibilidades, pode até atingir o conteúdo “certo” a um público com interesses distintos ao que estiver sendo apresentado, ou seja, poderia não atender a qualquer valor cultural.

Sendo assim, uma possível abordagem é a de que a informação que possa atingir públicos variados deve seguir um caminho em que ao verificar o que tiver sido

pesquisado ou desejado pelo usuário possamos indicar conteúdo adequado, ou seja, personalizado, e, sendo assim, não será o mesmo conteúdo a todos, pessoas diferentes serão tratadas diferentemente.

Se em algum momento reivindicarmos a criação de uma inteligência artificial para dirimir o assunto, devemos abordar questões sobre linguagem natural, raciocínio automático e aprendizado de máquina. Mas, quando falamos de HTML (*HyperText Markup Language*) pensamos em camada de apresentação e não em troca de dados ou indexação de dados de forma adequada. Aí entra a Web Semântica para tentar sanar todo esse problema, ou seja, teremos que através de metadados e ontologias (conceito explicado no parágrafo seguinte) classificar o conteúdo de forma adequada. Com a devida classificação podemos através de mecanismos de buscas adequados fornecer o conteúdo certo ao que o usuário anseia. O mesmo critério valeria para quando estivermos efetuando uma pesquisa num Google, Yahoo, MSN, AltaVista etc. – se buscamos informação sobre redes de computadores e digitamos apenas a palavra rede, obteremos informações diversas sobre redes para pescadores, redes de televisão, redes para descanso e também, claro, sobre redes de computadores. A pesquisa precisaria ser mais específica, mas ao mesmo tempo se nos metadados de todas as páginas que trazem informações sobre redes diversas não houver a devida classificação e os mecanismos de busca não forem suficientemente bons para efetuar a devida triagem, encontraremos o mesmo problema de recebermos informações que terão que ser humanamente analisadas. Com a Web Semântica conseguimos uma classificação adequada em que a máquina poderá filtrar tudo que não nos interessaria nos resultados de busca ou com base no comportamento do usuário. Através da Web Semântica também, pode-se fornecer conteúdo com maior probabilidade de atender aos anseios do mesmo. Uma abordagem luliana, em que o que é informado é fixo, dogmático, não corresponde à realidade e ao dinamismo do mundo Web, ou seja, não podemos determinar possíveis respostas pré-estipuladas a qualquer mecanismo de pesquisa – o caminho é a classificação adequada, a utilização da Web Semântica e claro, podemos utilizar inteligência artificial no sentido de permitir a aprendizagem e o raciocínio automático para aperfeiçoar esses processos, mas no que tange à linguagem natural o fato de utilizarmos Web Semântica e termos documentos compreensíveis por máquinas não implica em uma inteligência artificial que passará a entender o que os humanos falam.

O conceito aqui é que com a Web Semântica os computadores poderão resolver problemas bem definidos através da utilização de dados catalogados de forma adequada.

Precisamos classificar os dados de forma adequada e aí sim, conceituamos ontologia que, na filosofia, é o estudo da existência ou do ser (verbo). É basicamente uma maneira de compreender identidades e grupos de identidades, chamados de classes. Em TI chamamos de ontologias as classificações de identidades, sejam elas informação ou matéria (objetos no mundo real). Podemos criar uma ontologia que descreva livros no Amazon (www.amazon.com) e as relações entre os mesmos, assim como podemos criar uma ontologia sobre os diferentes tipos de barcos de pesca. Importante é saber que ontologias podem descrever qualquer coisa. Importante, também, diferenciar taxonomia de ontologia. A taxonomia também permite classificações, mas elas terão uma estrutura hierárquica e sem relacionamentos. Por exemplo, o homem é um ser vivo, vertebrado, mamífero etc. Essa classificação da taxonomia é rígida e não permite relacionamentos de causa-efeito ou de dependência entre as partes que a ontologia permite.

Definindo o conteúdo de um *site* com mecanismos inteligentes vimos, preliminarmente, que o caminho é a Web Semântica e aqui cabe uma ressalva de Karin Breitman sobre o atual estágio no qual a Web se encontra:

No estágio atual temos, predominantemente, a Web Sintática: onde nos preocupamos apenas em apresentar as informações. Com a Web Semântica teremos condições de oferecer significado adequado às páginas e com isso passaremos a enfatizar os mecanismos de troca e busca de informações.

A grande verdade é que a Internet se desenvolveu mais rapidamente como um meio para a troca de documentos entre pessoas, em vez um meio que fomentasse a troca de dados e informações que pudessem ser processadas automaticamente. Como consequência, o conteúdo semântico das páginas, ou seja, seu significado, é codificado de uma maneira acessível para seres humanos apenas.

Ou seja, com a evolução da Web Semântica vislumbra-se um caminho à oferta de conteúdo adequado, ou até mesmo: inteligente, na interação do usuário.

II.02 – Interação com o usuário

Quando pensamos em interagir com um usuário muitas questões podem surgir: o que deve ser identificado, armazenado, filtrado e atualizado? Devemos rastrear e deixar fixo esse rastreamento? Quantas vezes devemos rastrear um usuário para obter um comportamento definido? O assunto seria inesgotável ou de abrangência para outra dissertação. Esta se limita a abordar a oferta de conteúdo adequado a partir de um interesse manifestado pelo usuário e por isso só mencionamos o fato de que o usuário pode ter seu comportamento rastreado e suas implicações, sem esgotar esse assunto.

Aqui cabe, uma explicação a respeito de perfil de usuário. Os responsáveis por um *site* têm ferramentas de análise de tráfego (WebTrends, Google Analytics, Urchin etc.), navegação, chegadas e saídas, *sites* referenciadores etc. Com essas ferramentas podemos caracterizar alguns perfis de usuários, tais como – classificação esta, informal - utilizada na Web por empresas como Abril⁹ ou iG¹⁰: usuários fiéis (aqueles que visitam o *site* com muita frequência, comentam, dão sugestões, enviam exceções encontradas e acessam o *site* com dois ou mais navegadores principais para verificar possíveis inconsistências – são quase co-autores do *site*); leitores cativos (são assinantes de *feeds*, comentam e participam do *site* mas sem o fanatismo dos usuários fiéis); usuários ocasionais (acessam o *site* de vez em quando, provenientes de algum mecanismo de busca, como o Google por exemplo) e usuários esporádicos ou pára-quadistas (acessam o *site* por uma navegação aleatória, vindo de um mecanismo de busca ou não, muitas vezes nem sabem por quê estão visitando o *site* – aliás, esse tipo de navegação aleatória tem a ver com o termo: “serendipismo”, que é a descoberta por acidente).

⁹ Editora Abril S.A. – a maior editora de revistas da América Latina – www.abril.com.br, publica revistas como Veja e Exame.

¹⁰ iG – Internet Group Brasil – um dos maiores provedores de acesso do Brasil – www.ig.com.br.

A definição do perfil do usuário, não deixa de ser um tipo de rastreamento do usuário em si. Mas quando pensamos em disponibilizar um conteúdo adequado não necessitamos, necessariamente, saber qual o perfil do usuário, conforme descrito sumariamente acima. E ressaltamos que para esta dissertação o que desejamos é oferecer conteúdo adequado a partir de uma interação de um usuário, um mecanismo de busca, por exemplo, e aí sim oferecer o que ele deseja através de algum mecanismo semântico ou inteligente. Mas se desejarmos registrar o perfil do usuário, para que em visitas futuras já haja um prévio direcionamento ao mesmo, precisamos saber de coisas, tais como: esse usuário em meu *site* – supostamente um *site* com conteúdo diversificado, dá preferências a quais tipos de canais de conteúdo (esportes, economia, saúde, finanças etc.). Dentro de cada canal, por exemplo: esportes, quais as categorias que ele dá preferência: futebol, automobilismo, boxe, dentre outros. Dentro da categoria futebol, por exemplo, esse usuário prefere Campeonato Brasileiro, Italiano ou Espanhol? São inúmeras as possibilidades, o que precisamos é definir o que deve ser registrado e aí armazenar em um banco de dados associado à autenticação do usuário, seu e-mail, por exemplo, para que em suas próximas visitas ele já possa ser direcionado as suas preferências principais com o uso da Web Semântica, mas esse tipo de abordagem, prévio direcionamento, foge aos objetivos desta dissertação.

II.03 – A utilização da Web Semântica

Segundo Tim Berners-Lee¹¹, a Web Semântica é uma extensão da Web tradicional, onde, a partir do uso intensivo de metadados, espera-se obter o acesso automatizado às informações, com base no processamento semântico de dados e [heurísticas](#) feitos por máquinas.

Para tal, desde 1998, a equipe do W3C (World Wide Web Consortium) vem trabalhando arduamente no desenvolvimento de tecnologias avançadas, que visam à representação estrutural e semântica dos recursos na Web. Essas tecnologias, aliadas à teoria de domínios ou ontologias, permitem oferecer um serviço Web com um nível maior de qualidade.

Dentro destas perspectivas, a Web será capaz de tecer uma rede extensa de conhecimento humano, podendo ainda, por meio do processamento via máquina, inferir novos conhecimentos.

As tecnologias acima mencionadas englobam desde linguagens de transporte de dados, como o XML (Extensible Markup Language), linguagens para representação da estrutura desses dados, tais como o RDF (Resource Description Framework) e linguagens para representação da semântica desses dados explicitando restrições sobre a semântica do mundo real. Aliadas a essa gama de linguagens, há também as ferramentas para manipulação das ontologias.

O filósofo francês Pierre Lévy¹² define a Web Semântica como "ferramenta" necessária para o desenvolvimento da inteligência coletiva. Segundo ele, a semântica da Web é a elaboração de um sistema de códigos. Não é necessário que as pessoas utilizem uma nova língua - cada um utilizará sua própria, mas haverá um software que traduzirá o que está sendo escrito ou dito para uma língua universal.

¹¹ Tim Berners-Lee é considerado o criador da Internet. Seu perfil encontra-se em: <http://www.w3.org/People/Berners-Lee/>.

¹² Pierre Lévy é professor da cadeira de Pesquisas sobre Inteligência Coletiva da Universidade de Ottawa (Canadá).

Segundo Breitman, o HTML é uma linguagem que se tornou muito interessante para os usuários, mas os *links* só têm sentido em seu contexto. Para as "máquinas", não é possível traduzir a informação, pois a orientação do HTML é para apresentação. Por isso, a importância dos metadados, relacionando os recursos semanticamente e tendo os conteúdos descritos de forma clara (enquanto o HTML trabalha com palavras-chave, o XML trabalha com metadados).

Para alguns, a Web Semântica é a Web do futuro. As pesquisas atuais estão buscando tecnologias para tornar mais eficazes as localizações das informações na Web. As ontologias, com suas subsequentes classificações, surgem para atender a essa necessidade.

II.03.01 – Web Semântica versus Inteligência Artificial

Segundo Karin Breitman, a Web Semântica não é Inteligência Artificial (IA).
E segundo Antoniou e Harmelen:

Se o objetivo da Inteligência Artificial é construir um agente de software que mostre inteligência no nível humano (e superior), o objetivo da Web Semântica é auxiliar humanos a realizarem suas tarefas diárias na rede.

A intenção de termos documentos compreensíveis por máquinas não implica numa inteligência artificial que fará com que os computadores passem a entender o que os seres humanos falam. Com a Web Semântica os computadores poderão resolver problemas muito bem definidos através do processamento de operações ou instruções que utilizarão dados. Não solicitaremos aos computadores que passem a compreender a linguagem das pessoas, mas serão as pessoas que passarão a ter um trabalho a mais de classificar as informações em representações que possibilitarão processamento automático – tais representações poderiam ser os metadados e as ontologias, por exemplo.

Segundo Karin Breitman, a maioria das técnicas necessárias para a construção da Web Semântica vem da área da Inteligência Artificial. Dada a história de insucessos da IA, uma preocupação seria se a Web Semântica não estaria no mesmo caminho. Segundo Antoniou e Harmelen, essa preocupação é totalmente infundada. A realização da Web Semântica não depende de inteligência superior ou algo no nível da inteligência humana, que era a promessa dos pesquisadores de Inteligência Artificial há vinte anos.

No caso da Web Semântica, soluções parciais serão satisfatórias. Pode ser que um agente de software não chegue nem perto das conclusões a que um ser humano chegaria, mas ainda assim este agente pode auxiliar para uma Internet muito superior à que temos hoje em dia. Aqui cabe uma explicação sobre agentes. Eles são definidos, segundo Breitman, como programas de software autônomos que agem em benefício de seus usuários. Um agente pessoal na Web Semântica, segundo Grigoris Antoniou e Frank Harmelen, vai receber algumas tarefas e preferências de um usuário, procurar informação nos recursos disponibilizados pela Internet, se

comunicar com outros agentes e comparar informações relativas às tarefas que deve desempenhar, de modo a fornecer respostas adequadas ao usuário.

A Web Semântica não exigirá uma Web separada, ela é uma extensão (termo, inclusive, utilizado por Tim Berners-Lee) da Web atual (denominada de Web Sintática). Na Web Semântica a informação vai ter significado bem definido através de linguagens de marcação semântica. Essas linguagens e ontologias serão acrescentadas às páginas atuais, através de marcações específicas ou através de arquivos separados em que as respectivas páginas farão *links* para acrescentar o devido conteúdo semântico.

Segundo Breitman, a Web Semântica não exigirá a utilização de expressões complexas, pois apesar de a linguagem-padrão recomendada para a Web Semântica (aqui Breitman, refere-se à [OWL – *Ontology Web Language*](#), que será explanado mais adiante) permitir a expressão de sentenças muito complexas, isto é, sentenças que utilizam conectivos lógicos, disjunção, inversão e axiomas, entre outros, não será exigido que todas as aplicações utilizem a marcação semântica em todo o seu potencial.

E por fim, a Web Semântica não é uma reprise de um experimento falido. Pois outra questão que poderia ser feita é: não havia sido tentado tudo isso, anteriormente, com sistemas de representação de conhecimento, nos projetos do KIF (Knowledge Interchange Format) e Cyc (redução de EnCYClopaedia, ou enciclopédia)? Segundo Hendler, a resposta é de certa forma, sim, mas havia outros objetivos. A visão da comunidade de Representação do Conhecimento está mais ligada à criação de modelos canônicos que poderiam ser globalmente utilizados, enquanto a Web Semântica tem um foco em ontologias menores, também chamadas de ontologias de domínio¹³ e no processo de integração dessas ontologias.

¹³ Ontologias de domínio descrevem o vocabulário relativo a um domínio específico através da especialização de conceitos presentes na ontologia de alto nível – que descreve conceitos muito genéricos, tais como: espaço, tempo e eventos.

No entanto e segundo Breitman, a experiência adquirida pelo projeto Cyc na área de representação de conhecimento, que forneceria uma ontologia de referência inestimável, não pode ser ignorada na construção da Web Semântica.

O gerenciamento de conhecimento engloba as tarefas de aquisição, disponibilização e manutenção de bases de dados. Na medida em que os sistemas de informação adquirem maturidade surge uma nova atividade, gerência de conhecimento, necessária para dar suporte à utilização da grande massa de dados gerada com a informatização das empresas. Atualmente muitos dos processos e regras de negócios de uma empresa podem estar em código, muitas vezes legado. Grande parte dessa informação está disponível de maneira fraca ou indevidamente estruturada. Do ponto de vista de gerência de conhecimento, as maiores limitações, segundo Breitman, são:

- Busca de informação – a maior parte das empresas utiliza indexação por palavras-chave para conduzir buscas a suas bases de dados.
- Extração de dados – muito tempo é perdido em tarefas ligadas à extração, à filtragem e à conversão de informação para diferentes formatos.
- Manutenção – problemas relativos a inconsistências de modelos conceituais e vocabulário fazem com que seja difícil identificar e eliminar dados obsoletos.
- Mineração de dados – muitas empresas têm recorrido a técnicas ligadas à mineração (*data mining*) de grandes bases de dados como forma de descobrir novas informações. No entanto, essas técnicas são difíceis de serem aplicadas quando as informações estão espalhadas em vários aplicativos distribuídos e pouco ou indevidamente estruturados.

A Web Semântica visa permitir sistemas de gerência de conhecimento muito mais avançados. Com a utilização de tecnologias, tais como: [RDF](#), [OIL](#), [OWL](#) e [lógicas de descrição](#), espera-se poder chegar a um nível de integração e troca de dados muito superior ao que se tem atualmente.

Ou seja:

- Conhecimento poderá ser organizado em espaços conceituais, de acordo com o seu significado. Essa organização será assistida por máquinas que serão capazes de fazer a seleção e a filtragem da informação. Ontologias serão cruciais para essa tarefa.
- Ferramentas automatizadas serão responsáveis pela verificação de consistência e mineração de novas informações.
- Mecanismos de busca baseados em palavras-chave serão substituídos por *queries*¹⁴ sofisticadas. A informação requisitada poderá ser recuperada, extraída e apresentada de maneira amigável.

Atualmente, há uma quantidade gigantesca de informações na Internet e os números não param de crescer. Quando fazemos uma pesquisa sobre um determinado assunto através de um buscador, como o Google por exemplo, muita informação inútil ou que não é de interesse ao usuário poderá ser listada. Para melhorar essa situação, faz-se necessário indexar de forma adequada os recursos na Internet, ou seja, precisamos acrescentar elementos que nos indiquem, por exemplo, que tipo de informação é fornecido em um *site*, necessitamos de metadados - que é o assunto do nosso próximo tópico.

¹⁴ *Query* é o termo em inglês para consulta, seu plural é *queries*. Uma *query* pode usar uma técnica específica ou conjunto de instruções para obter o que se deseja.

II.03.02 – Metadados

A International Federation of Library Associations (IFLA) define metadados da seguinte forma: “Metadados são dados sobre dados. O termo se refere a qualquer informação utilizada para a identificação, descrição e localização de recursos.

O consórcio W3 (W3C – World Web Consortium – www.w3.org) tem uma visão mais voltada para a Web Semântica. “Metadados são definidos como informações para a Web que podem ser compreendidas por máquinas”.

Quando se trata do mundo digital, chama-se de recurso o objeto descrito por metadados, pois este pode ser tanto um simples dado, quanto um documento, uma página da Web, ou até mesmo uma pessoa, uma coleção, um sistema, um equipamento ou uma organização. Na Web, o conceito recurso significa qualquer objeto que pode ser alcançado através de uma URI (Uniform Resource Identifier - www.w3.org/Addressing), como qualquer recurso que é acessado via seu endereço eletrônico. Isso inclui documentos, páginas pessoais, *sites* ou sistemas. A descrição de uma pessoa ou organização é feita através da descrição da página dessa pessoa ou organização.

Metadados descrevem os recursos da Web com a finalidade de facilitar a sua descoberta, localização e utilização. Motores ou mecanismos de busca (em inglês: “Search engines”), ao utilizarem esses metadados, proporcionam consultas bem mais precisas, envolvendo não somente palavras, mas propriedades descritas, como o autor do recurso, o formato do recurso, a data do recurso etc.

No mundo digital, metadados não se restringem àqueles usados para descrever recursos para permitir sua descoberta e localização. Por exemplo, metadados também descrevem os formatos dos recursos, a fim de permitir que computadores identifiquem quais aplicativos podem ser usados para manipular esses recursos. Aplicativos, ao analisarem metadados, verificam se o formato do recurso é compatível, as condições de uso do recurso, restrições de acesso etc.

Em computadores, metadados servem de representantes de um recurso e para caracterizar o trabalho original, para que usuários entendam seu significado, propósito, origem e condições de uso. Metadados são vistos como dados que descrevem propriedades de um recurso para diversos propósitos, como o contexto

em que o recurso se insere, sua qualidade, suas condições de uso, sua identificação, suas estratégias de preservação etc. Essa diversidade de tipos de metadados dá suporte à realização de inúmeras funções, como funções de descoberta do recurso, de localização do recurso, de avaliação do recurso, de uso do recurso etc.

Embora o conceito seja aparentemente novo, bibliotecários estão há séculos produzindo e padronizando metadados, à medida que extraem de documentos (que seriam os dados) informações de indexação e catalogação (que seriam os metadados), para oferecer aos usuários caminhos, para que estes possam buscar os documentos de que necessitarem. Na Biblioteconomia, o padrão MARC (“MACHine-Readable Cataloging – com mais detalhes no *site* da Biblioteca do Congresso Americano em: www.loc.gov/marc) é um exemplo de esquema de metadados. Estes metadados indicam propriedades do documento, como seu autor, sua data de publicação, seu título, seu assunto etc. e têm como finalidade permitir a descoberta e a localização desses documentos.

Os metadados podem estar inseridos nos próprios documentos que descrevem. Em páginas HTML é possível inserir metadados através de marcações (chamadas de meta *tags*), que não são visíveis quando a página é exibida pelo navegador. Esses metadados podem ser lidos por motores de busca ou por aplicativos que os utilizem.

Os metadados também podem ser armazenados em repositórios. Nesses repositórios, os metadados são produzidos através de anotadores, que são sistemas que guiam usuários na descrição dos recursos. Esse tipo de solução tem sido utilizada em organizações para permitir que estas descrevam seus recursos.

Essas organizações consideram como recursos, páginas da Web que identificam seus departamentos, seus especialistas, seus sistemas informatizados disponibilizados, bibliotecas digitais, repositórios de documentos digitais etc.

Através desses metadados, essas organizações possuem um maior entendimento de suas próprias potencialidades.

Um exemplo de metadado popularmente muito utilizado é a *tag* ID3 (também chamada de ID3 Tag) que é um conjunto de informações muito utilizado no formato de áudio MP3. Esse conjunto de informações que compõe a ID3 permite armazenar

informações como o título, artista, álbum, número da faixa ou outra informação sobre o arquivo no próprio arquivo.

A primeira versão da ID3 foi criada em 1996 para atender a necessidade de armazenar mais informações no arquivo e não apenas deixá-las no nome do próprio arquivo com extensão mp3. Desse modo, teve-se a ideia de adicionar alguns dados no fim do arquivo e logo tornou-se o padrão de metadados dos arquivos com extensão mp3.

Para atender a determinados objetivos, a uma comunidade ou a um assunto em específico diversos padrões de metadados foram criados. A seguir, veremos os principais padrões voltados à Web Semântica.

II.03.02.01 – Padrão Dublin Core

Os usuários desse padrão se organizaram em uma iniciativa, o Dublin Core Metadata Initiative (DCMI – www.dublincore.org). Essa organização dedica-se à adoção de padrões de metadados e ao desenvolvimento de vocabulários especializados para a descrição de recursos, de modo a viabilizar sistemas mais inteligentes de recuperação de informações.

Os elementos que compõem o padrão Dublin Core estão listados a seguir (<http://www.dublincore.org/documents/dcmi-terms/>):

-Assunto (<i>subject</i>)	O tópico abordado pelo recurso.
-Título (<i>title</i>)	Nome dado ao recurso.
-Criador (<i>creator</i>)	Pessoa responsável, podendo ser mais do que uma, pelo conteúdo intelectual do recurso.
-Descrição (<i>description</i>)	Descrição do conteúdo do recurso.
-Editor (<i>publisher</i>)	Entidade responsável por tornar o conteúdo disponível.
-Outro agente (<i>contributor</i>)	Entidade responsável por contribuições ao recurso.
-Data (<i>date</i>)	Data da publicação. O padrão sugerido é o W3CDTF (http://www.w3.org/TR/NOTE-datetime).
-Tipo (<i>type</i>)	Natureza do recurso. Padrão sugerido: DCMITYPE (http://dublincore.org/documents/dcmi-type-vocabulary).
-Formato (<i>format</i>)	O formato do arquivo. Exemplo: PDF, TXT etc.
-Identificador (<i>identifier</i>)	Identificação única do recurso. Exemplo: número.
-Relacionamento (<i>relation</i>)	Relacionamento com outros recursos.
-Fonte (<i>source</i>)	O recurso do qual este é derivado, se aplicável.
-Linguagem (<i>language</i>)	Linguagem do recurso.
-Cobertura (<i>coverage</i>)	Abrangência espacial ou temporal do recurso.

-Direitos (*rights*)

Informação sobre os direitos pertinentes ao recurso.

Segundo Breitman, o padrão Dublin Core é um padrão bastante simples, como pode ser observado a partir do grupo básico de elementos que o compõem. Sua simplicidade é um ponto forte, pois permite a disseminação e a utilização em larga escala, e é, ao mesmo tempo, sua maior fraqueza, pois não acomoda uma semântica mais expressiva. É um padrão que fez opção pela facilidade sobre o poder de expressão.

Atualmente o Dublin Core é um padrão ANSI para metadados (ANSI/NISO Z39.85) e norma ISO (ISO Standard I5836-2003).

Veremos no tópico a seguir um padrão aberto, o RDF, com grande capacidade de expressão e que é o padrão mais compatível com o dinamismo da Web atual.

II.03.02.02 – Resource Description Framework - RDF

Em 1995, um ano após a realização da conferência que deu origem ao padrão Dublin Core, realizou-se um outro evento onde foi proposto um novo padrão para metadados, o Framework de Warwick - que utiliza os descritores básicos do Dublin Core, sendo adicionados novos elementos. Ele é composto por vários pacotes, onde cada um deles pode utilizar uma sintaxe diferente. Apesar de aumentar a flexibilidade do modelo Dublin Core, não garante que dois pacotes poderão trocar dados entre si e nada no Framework de Warwick garante que dois conjuntos de metadados possam estar utilizando um conceito com significados diferentes ou dois conceitos com um mesmo significado. De modo a tratar essas dificuldades, um novo padrão surgiu, o RDF – Resource Description Framework.

O RDF é uma linguagem declarativa que fornece uma maneira padronizada de utilizar o XML para representar metadados no formato de sentenças sobre propriedades e relacionamentos entre itens na Web. Esses itens, denominados de recursos, podem ser virtualmente quaisquer objetos (texto, imagem, vídeo e outros), basta que possuam um endereço na Web.

O RDF recebeu grande influência da comunidade de bibliotecas digitais. O RDF pode ser entendido como uma implementação do Framework de Warwick sem um dos problemas identificados no Framework de Warwick: através da utilização da marcação de *namespace* propiciada pelo XML, a questão da superposição semântica é resolvida.

Um dos objetivos do RDF é tornar a semântica de recursos da Web acessível a máquinas. Apesar de a informação na Web poder ser lida automaticamente, sua semântica não é definida. O RDF vai acrescentar metainformação a esses recursos, de modo, a possibilitar às máquinas lidarem com eles de modo inteligente. Descrições RDF foram projetadas para fornecer informações aos computadores e não aparecerem na tela.

O modelo RDF básico destaca-se pela simplicidade com que busca estruturar o conteúdo de uma página Web. Eles são construídos com a utilização de recursos e propriedades, descritos a seguir:

• **Recursos:** representam o universo de objetos que podem ser descritos pelo modelo RDF. Uma página inteira da Web ou uma parte dela; uma coleção de páginas; ou um objeto que não é diretamente acessível via Web, por exemplo, um livro impresso. Para cada recurso é associado um identificador único (URI) de forma a poder identificá-lo posteriormente.

• **Propriedades:** representam os aspectos do recurso a serem descritos.

Propriedades podem ser visualizadas como atributos (características) de recursos. Também são utilizadas para descrever relacionamentos entre recursos.

Neste sentido, o modelo de dados RDF assemelha-se ao modelo Entidade-Relacionamento. Cada propriedade tem um significado específico, definem seus valores permitidos, os tipos de recursos que podem descrever, e seus relacionamentos com outras propriedades.

Vejam como ficaria, a partir de um exemplo de um *site* de conteúdo, o RDF correspondente. A tabela a seguir ilustra uma parte de um conteúdo HTML de um *site* de conteúdo, por exemplo: www.ig.com.br, dentro do canal de Esportes e da categoria Futebol, os campeões brasileiros dos anos de 2005 e 2006:

Categoria	Campeonato	Ano	Campeão
Futebol	Brasileiro	2005	Corinthians
Futebol	Brasileiro	2006	São Paulo

Um possível arquivo RDF correspondente é o seguinte:

```
<?xml version="1.0"? >
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:esp="http://www.pucsp.br/~demi/vocabulario/esporte/#">
<rdf:Description
rdf:about="http://www.ig.com.br/esportes/futebol/br_camp.aspx">
    <esp:ano>
```

```

        <rdf:Description
            rdf:about="http://www.ig.com.br/#2005">
        </rdf:Description>
    </esp:ano>
</rdf:Description>
<rdf:Description rdf:about="http://www.ig.com.br/#2005">
    <esp:categoria>Futebol</esp:categoria>
    <esp:campeonato>Brasileiro</esp:campeonato>
    <esp:campeao>Corinthians</esp:campeao>
</rdf:Description>
<rdf:Description
    rdf:about="http://www.ig.com.br/esportes/futebol/br_camp.aspx">
    <esp:ano>
        <rdf:Description
            rdf:about="http://www.ig.com.br/#2006">
        </rdf:Description>
    </esp:ano>
</rdf:Description>
<rdf:Description rdf:about="http://www.ig.com.br/#2006">
    <esp:categoria>Futebol</esp:categoria>
    <esp:campeonato>Brasileiro</esp:campeonato>
    <esp:campeao>São Paulo</esp:campeao>
</rdf:Description>
</rdf:RDF>

```

Explicando o arquivo RDF acima, temos:

A primeira linha do arquivo é declarativa, ela diz qual a versão do XML está sendo utilizada, no caso trata-se da versão 1.0.

O elemento `rdf:RDF` é o elemento raiz (e no caso tem o terminador `/rdf:RDF`). Neste caso, ele indica que o conteúdo é RDF.

O elemento `xmlns:rdf` indica o *namespace* do próprio RDF, ou seja, ele indica que todos os elementos ou atributos que comecem com o prefixo `rdf:` estarão localizados dentro de um único arquivo, que contém a especificação do vocabulário de RDF.

Também podemos ter vocabulários criados pelos autores do arquivo RDF em questão. No exemplo acima, foi apontado para <http://www.pucsp.br/~demi/vocabulario/esporte/#>. Neste local estarão descritos os termos utilizados: ano, campeonato, campeão e categoria. O *namespace* deste local é o `xmlns:esp` e que identificará as etiquetas do tipo `esp:`.

O elemento `rdf:Description` (e seu terminador `/rdf:Description`) contém a descrição do recurso identificado através do atributo `rdf:about`.

O elemento `esp:ano` descreve uma propriedade do recurso. O mesmo acontece para os elementos `esp:categoria`, `esp:campeonato` e `esp:campeao`.

É recomendável que o nome do arquivo RDF seja o mesmo do recurso que está sendo descrito com o sufixo `.rdf`. Por exemplo: se a página for: `br_camp.aspx`, o arquivo RDF correspondente será: `br_camp.aspx.rdf`.

Para ligar ou efetuar o link entre o recurso HTML (que expõe o conteúdo ao usuário, sendo que no caso trata-se de um arquivo `asp.net` com extensão `aspx`) e sua respectiva descrição RDF, devemos adicionar a *tag* `<link>` dentro da seção da *tag* `<head>` da página. Por exemplo: `<link rel="meta" href="br_camp.aspx.rdf">`.

Como vemos, o RDF é simples, intuitivo, prático e extensível.

II.03.02.03 – Críticas aos Metadados

Vimos a criação de uma estrutura projetada para que sejam fornecidas informações às máquinas e que não necessariamente aparecerão na tela do computador de um usuário ou internauta. Mas essa é a solução mais adequada? Segundo Cory Doctorow¹⁵ no *site*: <http://www.well.com/~doctorow/metacrap.htm> temos muitos problemas a serem enfrentados:

-As pessoas mentem. Estamos num mundo competitivo e as pessoas querem aparecer ou vender, por isso, não há garantias de que as classificações serão honestas, politicamente corretas ou algo que valha.

-As pessoas são preguiçosas. O modelo baseado em metadados supõe que os usuários ou responsáveis pelas páginas irão classificar, indexar ou fornecer as informações necessárias. Como podemos supor isso se muitos e-mails não têm o assunto a que se referem discriminado e muitas páginas Web não possuem sequer um título descritivo adequado.

-As pessoas são disciplicentes. A falta de cuidado e interesse das pessoas em fazer de forma correta a criação e manutenção de metadados. Doctorow cita como exemplo o *site* de leilão e-Bay (www.ebay.com), um *site* em que seus utilizadores teriam interesse em classificar corretamente os produtos que colocam à venda mas, que apresenta um grande número de produtos cadastrados erroneamente. Ele sugere

¹⁵ Cory Doctorow é um jornalista canadense, atua como professor visitante na Universidade de Southern na Califórnia. Mais informações em: http://en.wikipedia.org/wiki/Cory_Doctorow#Biography.

uma busca por “plam pilot” em vez de “palm pilot” onde se verificará que podem aparecer resultados cadastrados de forma incorreta.

-Missão impossível: conheça a si mesmo. Como diz Doctorow, seria uma metautopia supor que as pessoas se conhecem adequadamente e que saberão descrever seus metadados e domínios de forma adequada. A experiência prática diz que as pessoas não conseguem observar seus próprios comportamentos de forma adequada. Isso pode ser comprovado com experiências, tais como: pergunte a um programador quanto tempo ele levará para codificar uma determinada rotina ou pergunte a um empreiteiro quanto tempo será necessário para construir alguma coisa.

-Falta de neutralidade nos modelos. A forma de organizar as categorias dos metadados já demonstra uma tendência dos usuários. Nenhum modelo é realmente isento de influências. O processo de descrição de metadados não é exceção. Espera-se que a decomposição do domínio e sua categorização sejam influenciadas pelos interesses de quem está fazendo a modelagem. Doctorow cita o exemplo de um fabricante de máquinas de lavar. Se ele for consciente de questões relativas ao meio ambiente ou se a vantagem competitiva oferecida por seu produto for o baixo consumo de energia é de se esperar uma decomposição do seguinte tipo:

Consumo de energia:

Consumo de água limpa:

Tamanho:

Capacidade:

Confiabilidade:

Enquanto que se o fabricante tiver o *design* como diferencial, poderíamos esperar uma decomposição assim:

Cor:

Tamanho:

Programabilidade:

Confiabilidade:

Ou seja, cada um procuraria enfatizar o que tem de destaque ou o item que apresentaria melhores resultados.

-Métricas influenciam resultados. Evidentemente que a escolha das métricas a serem utilizadas na caracterização da informação fornecida vai sofrer a influência de quem está escolhendo o conjunto de métricas que será utilizado. Da mesma forma que no item anterior, pode-se dar preferência a aspectos que sabidamente vão ter uma pontuação maior do que a de seus concorrentes, em detrimento de itens que podem não ser tão interessantes sob o ponto de vista da concorrência. Exemplo disso, são softwares que apresentam boas características em termos de segurança, mas são ruins em termos de conveniência

-Existe mais de uma maneira de descrever alguma coisa. “Não, eu não estou assistindo desenhos animados! É antropologia cultural.” Pessoas razoáveis podem discordar para sempre em como descrever uma determinada coisa. Solicitar para que todos usem o mesmo vocabulário para descrever seus recursos, forçaria a uma homogeneidade de ideias.

E qual a solução? Jogar tudo fora? Os americanos têm uma expressão idiomática que adverte os usuários, quando não estão satisfeitos com algo, para “não jogar fora o bebê junto com a água do banho”. Na essência, o que esse ditado quer dizer é que mesmo com uma série de restrições que podemos fazer ao assunto, no caso os metadados, ainda existem muitas coisas válidas (o bebê), que não devem ser jogadas fora.

Enfim, os metadados podem não ser a solução de todos os problemas, mas são muito úteis e auxiliam na identificação, classificação ou indexação de páginas da

Web, melhorando o resultado de buscas e permitindo que uma parte do processamento possa ser efetuado por computadores, em vez de seres humanos.

Devemos ressaltar que para obter resultados satisfatórios em uma pesquisa precisamos, também, classificar as informações de uma forma adequada e para isso, apresentaremos no próximo capítulo o conceito e a utilização de ontologias para esse fim.

II.03.03 – Ontologias

Neste capítulo serão expostas as características das ontologias.

Segundo Karin Breitman, temos a seguinte definição para ontologias:

Ontologias são especificações formais e explícitas de conceitualizações compartilhadas. Ontologias são modelos conceituais que capturam e explicitam o vocabulário utilizado nas aplicações semânticas. Servem como base para garantir uma comunicação livre de ambiguidades. Ontologias serão a língua franca da Web Semântica.

O termo conceitualização utilizado por Karin Breitman, acima exposto aponta para um modelo abstrato de algum acontecimento que identifique conceitos de importância desse acontecimento. Já a expressão “explícita” quer dizer que os tipos de conceitos usados e as limitações do uso desses conceitos devem ser de forma explícita. Através dessa descrição é possível visualizar algumas características importantes. Uma delas é o fato de que uma ontologia deva ser explícita, as outras se atribuem ao fato de que uma ontologia deva ser formal e descrever um conhecimento comum a um grupo.

Uma segunda definição para o termo é dada por Gómez e Corcho, eles definem uma ontologia da seguinte forma: “Uma ontologia é um conjunto de termos ordenados hierarquicamente para descrever um domínio que pode ser usado como um esqueleto para uma base de conhecimentos.” Diferentemente da primeira, essa definição apresenta algumas informações sobre a estruturação de uma ontologia. Pois, segundo ela, uma ontologia deve possuir termos organizados com uma hierarquia associada, o que é chamado de taxonomia. Ainda nessa descrição é apresentada outra característica importante, a qual consiste numa das principais utilidades de uma ontologia que é servir como um *schema*, ou seja, um marcador, para um banco de conhecimentos, visto essa muito comum dentro da área da gestão de conhecimentos. Ainda nessa última definição de ontologia vale à pena ressaltar que essa definição mostra uma diferença importante sobre o que é uma ontologia e o que é uma base de conhecimento. Como Guimarães descreve, uma ontologia disponibiliza um esqueleto básico no qual se pode edificar uma base de conhecimentos. Ela também provê um conjunto de conceitos e termos para descrever certo domínio, enquanto a base de conhecimento usa essas expressões para detalhar uma determinada realidade. Caso aconteça de que essa realidade seja alterada

a base de conhecimento será modificada, porém a ontologia ficará a mesma, desde que o domínio fique inalterado.

Segundo o trabalho de Guimarães, as principais vantagens para o uso de ontologias são: primeiramente, o fato de que ontologias fornecem um vocabulário para a representação do conhecimento. Esse vocabulário tem uma conceitualização que o sustenta, de forma a prevenir interpretações diferentes dele. As ontologias ainda permitem o compartilhamento de conhecimento, dessa forma caso haja uma ontologia que molde adequadamente certo domínio de conhecimento, essa pode ser compartilhada e usada por pessoas que criem aplicações dentro desse domínio. Um exemplo dessa característica pode ser dado na seguinte situação: imaginemos a existência de uma ontologia para o domínio de barcos. Uma vez que essa esteja disponível, qualquer outro barco pode construir seus catálogos usando o vocabulário fornecido por essa ontologia sem que precisem refazer uma avaliação do domínio de barcos. Diferentemente da linguagem natural, onde as palavras podem ter significados diferentes conforme o seu contexto, as ontologias por serem escritas em linguagem formal fornecem uma descrição exata do conhecimento. Um exemplo disso é quando alguém fala a palavra “globo”, dependendo do estado mental da pessoa que ouviu o termo, poderá associá-lo com o significado da rede de televisão Globo, como também é possível que para essa mesma pessoa a palavra tenha a semântica de um corpo esférico. Isso não aconteceria, ou pelo menos teria pequenas chances de acontecer, se existisse uma conceitualização comum entre as duas pessoas que estão se comunicando. Por exemplo, imagine que as duas pessoas concordem em usar uma ontologia sobre o domínio de formas geométricas, provavelmente não existiria mal entendido entre elas.

Vimos o que é uma ontologia e quais são as suas principais vantagens. Com esses tópicos já é possível ter-se uma ideia básica das partes que compõem uma ontologia, porém vale a pena formalizar essa questão. Segundo Guimarães, entre as várias definições sobre o que compõe uma ontologia, a que mais se destaca é dada por Maedche devido ao seu alto grau de formalismo. Porém, a descrição feita por Gómez também é bastante completa. Isso fez com que Guimarães mesclasse as definições dos dois autores, e é essa mistura que será apresentada aqui.

Um conjunto de conceitos e uma hierarquia entre esses conceitos, ou seja, uma taxonomia é definida como um dos componentes principais que compõem uma

ontologia. Aqui, os conceitos possuem a capacidade de ser abstratos, um exemplo disso é a força; concretos, como por exemplo, um navio; elementares que é exemplificado por um elétron; ou compostos, como por exemplo, um átomo. Para melhor se entender taxonomia pode-se pensar no conceito próton ser um subconceito do conceito átomo.

Outra peça que também compõe uma ontologia é um conjunto de relacionamento entre esses conceitos. Como exemplifica o fato de que entre o conceito pessoa e o conceito carro a relação que há é a de ser dono. Ainda há um conjunto de funções, onde uma função é um caso especial de relacionamento em que um conjunto de elementos tem uma relação única com outro elemento. Um exemplo de função é o fato de certo conceito ser pai biológico de outro.

Por fim, contrário ao conjunto de funções é um conjunto de axiomas, onde há regras que sempre são verdade. Exemplificar axioma é fazer-se pensar que toda pessoa tem uma mãe.

Ressaltamos a principal diferença entre taxonomia e ontologia que é o fato de que a taxonomia permite classificações, mas elas terão uma estrutura hierárquica e sua classificação é rígida e não permite relacionamentos de causa-efeito ou de dependência entre as partes que a ontologia permite.

E para obtermos as especificações formais, relacionadas à definição de Karin Breitman acima, ou, simplesmente, ontologias que desejamos, precisamos de linguagens apropriadas para isso – o que será apresentado no próximo capítulo – e também precisamos conhecer quais são as principais ontologias e qual o processo de criação de uma ontologia, caso desejemos obter uma especificação inexistente – tudo isso será apresentado nos capítulos subsequentes.

II.03.03.01 – Linguagens para Representação de Ontologias

Neste capítulo serão expostas as características das principais linguagens para representação de ontologias.

Listando as principais linguagens para representação de ontologias, temos:

- RDF (Resource Description Framework): esta linguagem também pode ser utilizada para metadados – e foi explanada detalhadamente em item anterior ([II.03.02.02](#)).
- SHOE (Single HTML Ontology Extension): é uma linguagem que se apresenta como uma extensão do HTML. Ela fornece *tags* particulares que podem ser utilizadas para representar ontologias e que não fazem parte das *tags* que são utilizadas no HTML e por isso não são mostradas a partir dos browsers. Sua intenção é armazenar informações a respeito do conteúdo das páginas, permitindo maior eficiência aos mecanismos de busca. Podemos ter agentes que utilizariam o SHOE para realizar buscas semânticas na rede.

Segundo Heflin, Hendler e Luke, a linguagem SHOE faz uma distinção entre o conteúdo das páginas – asserções ou instâncias – e a terminologia, informação acerca dos metadados. SHOE permite a definição de conceitos, relacionamentos e atributos. Veja no “[Anexo 01 – Exemplo de Arquivo utilizando SHOE](#)” um arquivo da Universidade de Princeton que o utiliza. Nesse exemplo, observe a utilização de novos *tags*: INSTANCE KEY (instância), CATEGORY NAME (conceito) e RELATION NAME (função que relaciona dois conceitos – propriedade). Essas *tags* são adicionadas ao conteúdo HTML como se fosse um novo cabeçalho. Esse conjunto de *tags* com o conteúdo HTML forma a página SHOE.

SHOE é menos expressivo que o RDF e, principalmente, dificulta a manutenção das páginas que a utilizem. Ela foi descontinuada, mas é mantida pela Universidade de Maryland, que a havia criado. Porém, devido a sua praticidade e simplicidade é uma boa opção para criar-se uma ontologia e por isso foi exemplificada no [Anexo 01](#).

Mais informações e exemplos da utilização de SHOE estão em:
<http://www.cs.umd.edu/projects/plus/SHOE/>.

- OIL (Ontology Inference Layer) – segundo Gómez-Pérez OIL pode ser definida assim:

Uma linguagem baseada em frames (quadros) que utilizam lógica de descrição para fornecer uma semântica clara, ao mesmo tempo em que permitem implementações eficientes de mecanismos de inferência que garantam a consistência da linguagem.

Muito do trabalho relacionado à linguagem OIL foi incorporado à OWL, que veremos adiante.

Mais informações no quadro comparativo entre as linguagens DAML e OIL, no “[Anexo 02 – Tabela Comparativa entre as linguagens OIL e DAML](#)”. Informações detalhadas e exemplos relacionados à OIL poderão ser encontrados no seguinte endereço:
<http://www.ontoknowledge.org/oil/>.

- DAML + OIL. (DAML é o acrônimo de: DARPA (Defense Advanced Research Projects Agency) Agent Markup Language):

Segundo Karin Breitman, temos:

DAML + OIL é dividida em duas partes, domínio dos objetos, que consiste nos objetos que são membros de classes definidas na ontologia DAML, e domínio dos tipos de dados, que consiste nos valores importados do modelo XML. A ideia por trás da separação é permitir a implementação de mecanismos de inferência, já que realizar inferências sobre tipos concretos de dados não seria possível. DAML é composta por elementos de classe, expressões de classe e propriedades.

Uma comparação entre as linguagens OIL e DAML encontra-se em: “[Anexo 02 – Tabela Comparativa entre as linguagens OIL e DAML](#)”.

Informações detalhadas e exemplos podem ser encontrados em:
<http://www.daml.org/>.

- OWL (Ontology Web Language)

Lançada como uma revisão da DAML + OIL pelo W3C, foi projetada para atender às necessidades das aplicações para a Web Semântica, tais como:

- Construção de ontologias
- Fornecer claramente fatos sobre um determinado domínio
- Racionalizar sobre ontologias e fatos

Os elementos básicos da OWL são: [namespaces](#), cabeçalhos, classes, indivíduos, propriedades e restrições.

- Namespaces: são declarações que se localizam entre etiquetas do tipo `rdf:RDF` e permitem que os identificadores que estarão presentes na ontologia sejam interpretados sem ambiguidades.

- Cabeçalhos: sob a etiqueta `owl:Ontology` é comum incluir uma coleção de etiquetas sobre a própria OWL. Elas são responsáveis por registrar comentários, pelo controle de versão e pela inclusão de conceitos e propriedades de outras ontologias.

- Classes: são utilizadas para descrever os conceitos mais básicos de um domínio, que servirão como bases ou raízes de várias taxonomias. O construtor fundamental de uma taxonomia é `rdfs:subClassOf`, que define a hierarquia de classes, através de relacionamentos de generalização (tipo-de). Por exemplo, temos a sentença: “Um exame de sangue é um exame médico”. Em OWL classificamos os conceitos `Exame_Medico` e `Exame_De_Sangue` como classes e estabelecemos um relacionamento de generalização entre eles utilizando a *tag* `rdfs:subClassOf`, conforme exemplo a seguir:

```
<owl:Class rdf:ID="Exame_De_Sangue">
    <rdfs:subClassOf rdf:resource="Exame_Medico" />
    ...
</owl:Class>
```

Ressalte-se que esse comportamento é transitivo. Isto é, se existe uma classe `Hemograma` que é subclasse da classe `Exame_De_Sangue` e

esta, por sua vez, é subclasse de Exame_Medico, temos que a classe Hemograma é subclasse de Exame_Medico.

- Indivíduos: são objetos do mundo. Eles pertencem a classes e são relacionados a outros indivíduos e classes através de propriedades. Eles são os membros das classes. Exemplo:

```
<Exame_De_Sangue rdf:ID="Colesterol" />
```

Propriedades: descrevem características em geral. Exemplo:

```
<owl:Class rdf:ID="Exame_De_Sangue" />
```

```
<Exame_De_Sangue rdf:ID="Colesterol">
```

```
  <quantidadeMinimaColeta
```

```
    rdf:datatype="&xsd:positiveInteger">1000</quantidadeMinimaColeta>
```

```
</Exame_De_Sangue>
```

Restrições: são utilizadas para definir limites para indivíduos que pertencem a uma classe.

A linguagem OWL oferece três sublinguagens projetadas para uso de comunidades específicas de implementadores e usuários:

A **OWL Lite** dá suporte aos usuários que necessitam, principalmente, de uma hierarquia de classificação e simples características de restrição. Por exemplo, enquanto a OWL Lite suporta restrições de cardinalidade, ela somente permite 0 (zero) ou 1 (um) como valores de cardinalidade. Dessa forma, é mais simples criar ferramentas que suportam a OWL Lite do que seus custosos parentes, além de

fornecer um rápido caminho de migração para o tesauro¹⁶ e outras taxonomias.

A **OWL DL** dá suporte aqueles usuários que desejam o máximo de expressividade sem perder a completude computacional (todas as conclusões são garantidas de serem computadas) e a capacidade de decisão (todas as computações serão finalizadas em um tempo finito) dos sistemas de raciocínio. A OWL DL inclui todos os construtores da linguagem OWL com restrições como separação entre tipos (uma classe não pode ser ao mesmo tempo um indivíduo ou tipo e uma propriedade não pode ser ao mesmo tempo um indivíduo ou uma classe). OWL DL tem esse nome devido a sua correspondência a Lógica de Descrição, ou *Description Logic*, um campo de pesquisa que tem estudado um fragmento de decisão particular de primeira ordem lógica. OWL DL foi desenhada para suportar o que já existe no segmento de descrição de lógica de negócios e tem propriedades computacionais desejáveis para sistemas de raciocínio.

A sublinguagem **OWL Full** é feita aos usuários que desejam o máximo de expressividade e a liberdade sintática do RDF, sem nenhuma garantia computacional. Por exemplo, na OWL Full uma classe pode ser tratada simultaneamente como uma coleção de indivíduos ou, simplesmente, como um indivíduo. Uma outra diferença significativa em relação à OWL DL é que um elemento `owl:DatatypeProperty` pode ser marcado como um elemento `owl:InverseFunctionalProperty`. A OWL Full permite uma ontologia aumentar o significado de um vocabulário pré-definido (RDF ou

¹⁶ Tesauro, também conhecido como dicionário de ideias afins, é uma lista de palavras com significados semelhantes, dentro de um domínio específico de conhecimento.

OWL). É improvável que todo software de raciocínio venha a ser capaz de suportar todas as características da OWL Full.

Cada uma dessas sublinguagens é uma extensão de sua predecessora mais simples, tanto no que pode ser legalmente expresso, quanto no que pode ser concluído validamente.

A OWL pode ser entendida como um componente da atividade Web Semântica. Esse esforço tenta fazer com que os recursos Web sejam mais acessíveis a processos automatizados, adicionando informações sobre os recursos que descrevem ou disponibilizam conteúdo Web.

Como a Web Semântica é naturalmente distribuída, a OWL deve permitir que as informações sejam reunidas de diferentes fontes. Isto é feito, em parte, permitindo que ontologias sejam relacionadas, incluindo informações explícitas de importações de outras ontologias.

Em adição, a OWL assume uma abordagem *open world*, ou seja, descrições de recursos não são confinadas a um único escopo de arquivo. Enquanto uma classe C1 pode ser definida dentro de uma ontologia O1, ela pode se estender em outras ontologias. As consequências dessas proposições adicionais sobre C1 são monotônicas. Novas informações não podem retratar informações anteriores. Novas informações podem ser contraditórias, mas fatos e conclusões somente podem ser adicionados, nunca excluídos. Trata-se de um comportamento típico de heranças de classes.

A possibilidade desse tipo de contradição é uma coisa que o projetista de uma ontologia deve considerar. É esperado que o suporte dado por ferramentas venha a ajudar a detectar esses casos.

Para se escrever uma ontologia que possa ser interpretada sem ambiguidade e que possa ser devidamente utilizada por agentes de software, precisamos de um sintaxe e uma semântica formal para a

OWL, sendo que a OWL é uma extensão das Semânticas do RDF -- <http://www.w3.org/TR/rdf-mt/> e a semântica da OWL é definida em *OWL Web Ontology Language Semantics and Abstract Syntax*:

<http://www.w3.org/TR/2004/REC-owl-guide-20040210/#FormalModel>.

De qualquer forma, antes de usarmos um conjunto de termos, necessitamos de uma indicação precisa sobre quais vocabulários específicos serão utilizados. Mais detalhes no tópico [II.03.03 03 – Criação de Ontologias](#).

Informações detalhadas e exemplos sobre a linguagem OWL podem ser encontrados em: <http://www.w3.org/TR/owl-ref/>.

Após vermos as principais linguagens para representação de ontologias, veremos a seguir as características das principais ontologias.

II.03.03.02 – Principais Ontologias

Neste capítulo serão expostas as características de algumas das principais ontologias.

Segundo Karin Breitman temos diversas ontologias, dentre as principais temos a SUMO (Suggested Upper Merged Ontology): o que traduzido, seria a ontologia de topo incorporada sugerida, ou seja, trata-se de uma ontologia de topo (base para outras ontologias) e é uma iniciativa da comunidade de software livre tendo como objetivo a construção de padrão público, a ser disponibilizado livremente com o uso da Internet. Como ontologia de topo, sua ideia é conter um número total de termos genéricos, algo estimado entre 1.000 (um mil) e 3.000 (três mil). A partir desses termos genéricos, podemos especializar para a criação de ontologias de domínios específicos.

Citando outras ontologias de domínio, temos: GUM (Generalized Upper Model), EDR (Eletronic Dictionary Research), WordNet (banco de dados léxico que fornece os possíveis significados de mais de 120.000 (cento e vinte mil) palavras em língua inglesa, organizados em conjuntos de sinônimos), ODP (Open Directory Project), NAICS (North American Industry Classification System), GALEN (desenvolvida pela organização OpenGALEN, ela representa uma terminologia clínica e foi desenvolvida para especificar restrições utilizadas em domínios médicos), FOAF (Friend Of A Friend), Gartner (índice de tópicos das pesquisas realizadas pelo grupo Gartner). E, temos também, as de domínio mais específicas ao conhecimento, tais como: KR (Knowledge Representation) e CYC (enCYClopædia, em inglês).

Também para a Gestão de Conhecimento, temos: RCPTIA (Relating Personal, Technological and Institutional Contexts by Associations, ou Relação de Contextos Pessoais, Tecnológicos e Institucionais). Segundo Alexandre Silva, temos:

O método RCPTIA propõe uma taxionomia dinâmica que associa contextos pessoais, contextos institucionais e contextos tecnológicos a todo arquivo digital ou endereço na Internet.

Existem inúmeros projetos voltados a ontologias e se focarmos apenas naqueles voltados ao processamento de linguagens naturais, temos o seguinte quadro (Tabela 02):

Tabela 02 - Projetos relacionados ao processamento de linguagens naturais

Projeto	Descrição
Gazelle	Traduz textos japoneses, árabes e espanhóis para o inglês; inclui processamento e análise semântica das línguas, geração de sentenças em inglês, construção de ontologias interlíngua e criação de léxicos para japonês, árabe, espanhol e inglês [Germann, 1998].
KPML	KMPL (Komet-Penman MultiLingual) desenvolve gramáticas e gera linguagens naturais; oferece um ambiente de desenvolvimento gráfico para a construção, manutenção e uso de gramáticas para diversas línguas (inglês, alemão, holandês, chinês, espanhol, russo, búlgaro e tcheco) [Bateman et al., 1996].
Mikrokosmos	Mecanismo em que textos em linguagem natural são traduzidos para textos no formato interlíngua (linguagem neutra, denominada TMR – <i>Text Meaning Representation</i>); a ligação da ontologia com o TMR é feita por meio de um léxico, no qual os significados dos itens são definidos por conceitos ontológicos [Beale, Nirenburg e Mahesh, 1995].
Oncoterm	Facilita a tradução de textos médicos sobre oncologia mediante uma ontologia baseada em textos especializados e dicionários médicos; os conceitos são organizados em categorias e representados por esquemas [Moreno e Hernández, 2000].
Ontogeneration	Gera textos em espanhol no domínio da química e utiliza linguagens naturais para responder a consultas sobre grupos, elementos e propriedades químicas; utiliza uma ontologia da química (<i>Chemicals</i> – ontologia do domínio da química que contém conhecimento sobre elementos químicos e estruturas cristalinas [Fernández-Lopez et al., 1999]), uma ontologia linguística e uma gramática em espanhol [Aguado et al., 1998].

PANGLOSS	Possui um sistema baseado em exemplos e um sistema de transferência léxica que trabalham em paralelo propondo traduções das entradas; o resultado ou tradução final é selecionado por um modelo estatístico [Frederking e Nirenburg, 1994].
Penman	Gera sentenças em linguagem natural a partir de uma entrada não linguística, aceita várias notações na entrada e foi projetado para o uso por pessoas com vários graus de sofisticação linguística e computacional [Teich e Bateman, 1993].
TechDoc	Gera documentos técnicos multilinguísticos (inglês, alemão e francês) a partir de uma representação independente construída pela análise comparativa de partes dos manuais técnicos e linguísticos disponíveis em diferentes línguas [Rösner e Stede, 1994].

No quadro abaixo (Tabela 03) vemos um agrupamento das ontologias linguísticas, algumas, inclusive, já mencionadas, anteriormente, na introdução deste capítulo.

Tabela 03 – Ontologias linguísticas

Ontologia	Descrição
CoreLex	É uma ontologia léxico-semântica que contém 126 (cento e vinte e seis) tipos semânticos, cobre cerca de 40 (quarenta) mil substantivos e define um grande número de classes polissêmicas, derivadas de análise do WordNet [Buitelaar, 2001].
EDR	Eletronic Dictionary Research – foi desenvolvido para processamento de linguagem natural, integra relações entre entradas léxicas e seus conceitos na forma de uma hierarquia e de relações semânticas, agrupados em um banco de dados de onde os conceitos são extraídos. É constituído por cinco grandes tipos de dicionários [Ogino et al., 1997].
EuroWordNet	Trata-se de um conjunto de dicionários para línguas europeias (alemão, espanhol, estoniano, francês, holandês, italiano, tcheco) organizados a partir de grupos de sinônimos e das relações entre eles. Cada dicionário é conectado a um índice interlíngua que acessa uma ontologia semântica de alto nível, possibilitando navegar de uma palavra em uma língua para a mesma palavra em outra língua. Pode ser utilizado para recuperação da informação multilíngue [Vossen, 1998].
GT (Goi-Taikéi's) Ontology	Trata-se de um léxico de 40 (quarenta) mil palavras japonesas que consiste de uma ontologia, um dicionário semântico de palavras e um dicionário semântico da estrutura. A ontologia classifica conceitos expressando as relações entre as palavras. O significado das palavras é dado por uma hierarquia semântica, em que os nós representam uma classe semântica e as arestas representam relações: é-um ou tem-um [Bond et al., 2001].
GUM (Generalized Upper Model)	Ontologia para processamento de linguagens naturais que simplifica a interface entre o conhecimento de um domínio e recursos linguísticos genéricos. Consiste de duas hierarquias: uma contém todos os conceitos e a outra contém todos os papéis

	[Bateman, Magnini e Fabris, 1995].
WordNet	Trata-se de um grande banco de dados léxico para a língua inglesa, em que são organizados 70 (setenta) mil grupos de sinônimos, cada um representando um conceito léxico. Os sinônimos são conectados por relações e o léxico é dividido em substantivos, verbos, adjetivos e advérbios [Miller, 1995].

A seguir, veremos como podemos criar uma ontologia.

II.03.03.03 – Criação de Ontologias

Neste capítulo serão expostas as características de como criar sua própria ontologia. Apresentaremos uma metodologia que pode ser aplicada para a criação de ontologias.

Segundo Tim Berners-Lee a descentralização típica da Web fará com que inúmeras ontologias sejam criadas e utilizadas.

Segundo Natalya F. Noy e Deborah L. McGuinness, podemos através dos passos seguintes, criar uma ontologia que seguirá uma metodologia de criação de fácil entendimento:

II.03.03.03.01 – Seleção dos Termos referentes ao domínio da ontologia:

Esta fase constitui em criar uma lista com todos os termos da ontologia, através de um vocabulário controlado, que tem a finalidade de nomear os conceitos, identificar suas propriedades e as relações desejadas.

II.03.03.03.02 – Definir as classes e a hierarquia de classes da ontologia:

Nesta etapa os termos que resultaram da fase anterior podem ser colocados em hierarquia de forma que os mais genéricos sejam refinados em termos mais específicos, conhecida como estratégia *top-down*. Ou os termos podem ser generalizados, conhecida como estratégia *bottom-up*. Ou até mesmo, pode ser utilizada uma combinação de ambas. De qualquer forma, os principais conceitos podem ser definidos primeiro e depois refinados e/ou generalizados e dessa forma podemos definir as diversas classes que irão compor a ontologia.

II.03.03.03.03 – Definir as propriedades:

Devemos considerar que somente as classes não fornecerão informação suficiente para responder às perguntas da competência do escopo. Ou seja, uma vez definidas as classes, deve-se descrever a estrutura interna dos conceitos, que são as propriedades.

II.03.03.03.04 – Definir as restrições:

Devemos, nesta etapa, descrever ou limitar o conjunto de valores possíveis para uma propriedade.

II.03.03.03.05 – Definir instâncias:

As instâncias são entidades do domínio que atendem às especificações de uma classe. A atividade de instanciação consiste em criar os registros de uma base de dados a partir do seu esquema (descrição das classes).

Após a execução dos passos acima mencionados, temos uma ontologia. Atualmente, existem inúmeras ontologias que são criadas, no entanto, não existe um padrão oficial para a sua criação o que pode dificultar a reutilização de conteúdos, no entanto, como já mencionado anteriormente, faz parte do dinamismo característico da Web.

E temos ferramentas que podem nos auxiliar. Tanto na parte de metadados como de ontologias, conforme veremos a seguir.

II.03.04 – Ferramentas

Neste capítulo serão expostas as características das ferramentas.

Temos diversas ferramentas voltadas à Web Semântica. Podemos classificá-las em três grandes categorias: Editores de Metadados, Editores de Ontologias e ferramentas de Visualização em Mecanismos de Inferência.

Os Editores de Metadados facilitam a criação e manutenção de metadados.

E analogamente, temos os Editores de Ontologias que nos permitem trabalhar tanto na criação quanto na edição ou manutenção de ontologias diversas.

As ferramentas de Visualização em Mecanismos de Inferência são softwares capazes de derivar novos fatos ou associações a partir de informações existentes. Existe uma crença de que esse tipo de ferramenta é capaz de emular o modo como os seres humanos pensam e chegam a conclusões. A realidade é que essas ferramentas não implementam capacidades de inteligência artificial, elas utilizam estratégias conhecidas de processamento de dados. Essas inferências serão tão boas quanto a informação que estiver disponível e, segundo Breitman, no caso da Web Semântica, esses mecanismos de inferência terão disponíveis apenas as classes e associações que possam ser possíveis de se descrever em modelos OWL ou RDF, ou seja, um conjunto muito restrito. Exemplos desse tipo de ferramenta são: [FaCT](#) e [Pellet](#), descritos no tópico seguinte [II.03.04.01](#).

II.03.04.01 – Principais Ferramentas

Neste capítulo serão expostas as características das principais ferramentas. Não é um estudo completo, pois seria muito extenso listar todas as ferramentas disponíveis atualmente. Foram selecionadas as que são mais utilizadas ou mais comentadas pela literatura referenciada.

Listando algumas ferramentas voltadas à Web Semântica, temos:

II.03.04.01.01 - OilEd:

Segundo Horrocks, Sattler e Tobies, o OilEd é um editor simples, o “NotePad” dos editores de ontologias. Ele utiliza as linguagens DAML+OIL. Sua intenção inicial é propiciar um simples editor que facilite o uso e estimule o interesse na linguagem OIL. Sua versão atual não provê um ambiente completo de desenvolvimento de ontologias – não suporta o desenvolvimento destas em larga escala, migração, integração, versionamento, argumentação e muitas outras tarefas envolvidas no processo de construção de uma ontologia.

II.03.04.01.02 - OilViz:

Segundo Karin Breitman, trata-se de uma ferramenta que pode ser adicionada à ferramenta OilEd. OilViz é o visualizador de hierarquias de classes. Auxilia na visualização da estrutura de classes da ontologia e oferece uma opção ao visualizador básico do OilEd. Mostra a hierarquia no formato de uma árvore de diretórios.

II.03.04.01.03 - Protégé:

Segundo Noy e Musen, trata-se de um ambiente interativo para projeto de ontologias de código aberto, que oferece uma interface gráfica para edição de ontologias e uma arquitetura para criação de ferramentas baseadas em conhecimento. A arquitetura é modulada e permite a inserção de novos recursos.

Com o Protégé temos um ambiente para criação e edição de ontologias e bases de conhecimento, que permite ao usuário iniciar o projeto de uma ontologia de forma rápida e intuitiva. Além do quê, ele é gratuito e de código aberto. Suas ontologias podem ser exportadas nos formatos RDF e OWL. Ele é feito em Java,

extensível e propicia rápido desenvolvimento de ontologias com sua interface amigável.

O Protégé também utiliza um mecanismo de inferência para a verificação de ontologias e subsequente classificação automática.

Na Figura 06 vemos o início da construção de um novo projeto, utilizando-se o Protégé e começando, a priori, pela construção de classes.

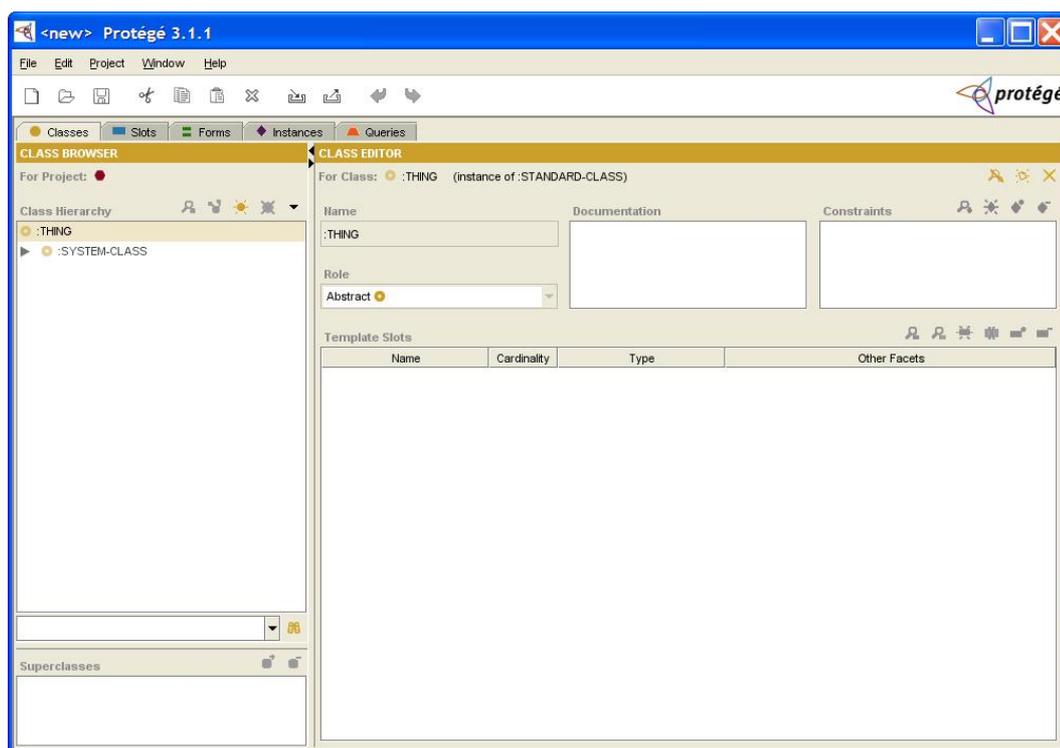


Figura 06 – Protégé

Mais informações e detalhes sobre o Protégé podem ser obtidas no seguinte endereço: <http://protege.stanford.edu/>.

II.03.04.01.04 - DC.dot:

Trata-se de um editor de metadados, cuja ferramenta se encontra na própria Web, no seguinte endereço: <http://www.ukoln.ac.uk/metadata/dcdot/>.

Karin Breitman relata:

O processo de criação de metadados acerca de um documento pode ser realizado no momento de sua criação ou posteriormente. Como o grande argumento a favor da Web Semântica é a falta de marcação de

metadados nas páginas da Internet, estamos presenciando o desenvolvimento de um grande número de ferramentas dedicadas à identificação e à edição de conteúdos de metadados a posteriori. Algumas ferramentas, utilizando páginas como marcação HTML como entrada, são capazes de gerar uma sugestão de marcação de metadados. Tipicamente são identificadas informações acerca do autor, data de criação, assunto e linguagem do conteúdo dessa página. O DC.dot é uma ferramenta desse tipo. A partir da submissão de qualquer página da Internet, essa ferramenta faz uma análise e sugere um conjunto de metadados, descritos no padrão Dublin Core (ver item [II.03.02.01](#)).

II.03.04.01.05 - API JENA:

Trata-se de uma API (Application Programming Interface) Java desenvolvida para dar suporte à implementação de aplicativos para a Web Semântica. Jena é software livre e foi originalmente desenvolvida pela HP. A API Jena suporta as linguagens RDF, RDF Schema e OWL e inclui um mecanismo para a realização de inferência. Ela transforma uma dada ontologia em um modelo abstrato de dados orientado a objetos e possibilita que suas primitivas (conceitos e relacionamentos) possam ser tratadas como objetos. A grande vantagem é possibilitar a manipulação das ontologias através de linguagens orientadas a objetos, como, por exemplo, o Java (bastante difundido e multi-plataforma). O endereço que apresenta a API Jena, tutorial e mais informações é: <http://jena.sourceforge.net>.

Na Figura 07 vemos a utilização da API Jena na IDE (Integrated Development Environment) Java NetBeans 6.0¹⁷. Como podemos ver, com a API Jena a manipulação de classes, propriedades etc. em um ambiente integrado de desenvolvimento torna-se uma realidade - o que propicia o desenvolvimento de programas que utilizam adequadamente os preceitos da Web Semântica.

¹⁷ NetBeans 6.0 é um ambiente de desenvolvimento integrado em Java, na sua versão 6.0 – Mais informações em: <http://www.netbeans.org>.

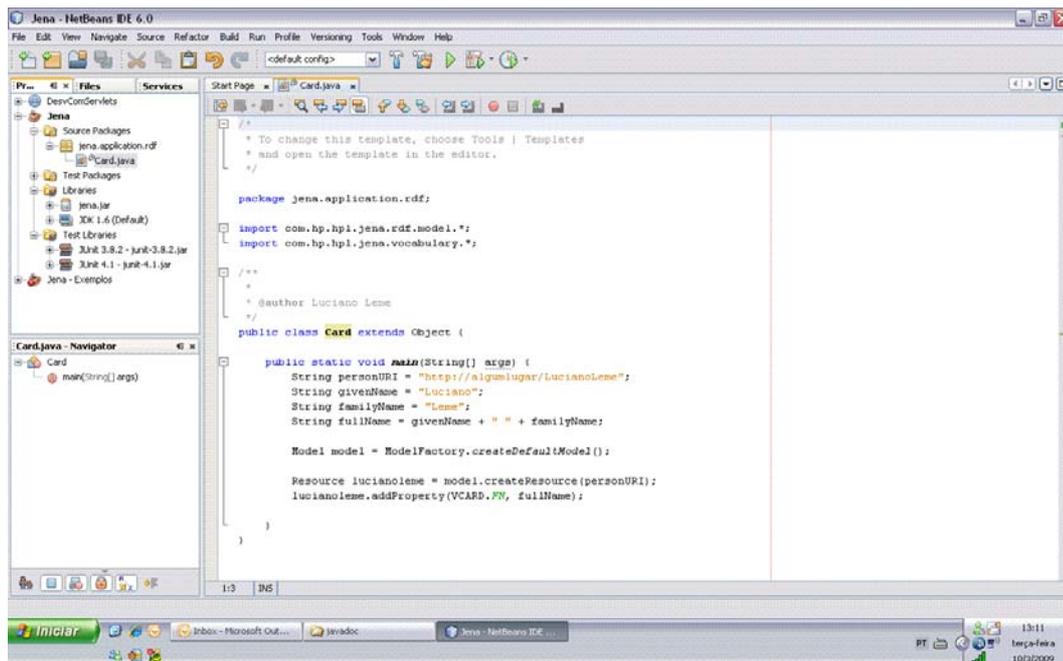


Figura 07 – Utilização da API Jena no NetBeans IDE 6.0

II.03.04.01.06 - Chimaera:

É um software que apoia o usuário na criação e manutenção de ontologias distribuídas na Web. Suas duas principais funções são: combinação de múltiplas ontologias e diagnóstico de ontologias individuais ou múltiplas. Apoia o usuário nas tarefas de carregamento de bases de conhecimento em diferentes formatos, reorganização taxionômica, resolução de conflitos com nomes, busca de ontologias, edição de termos etc. Sua intenção original era combinar fragmentos de bases de conhecimento.

Mais informações e detalhes sobre o sistema Chimaera podem ser obtidas no endereço: <http://www.ksl.stanford.edu/software/chimaera/>.

II.03.04.01.07 - SNOBASE:

O SNOBASE (Semantic Network Ontology Base) da IBM é um sistema de gerenciamento de ontologias que permite que uma aplicação de software realize buscas em ontologias sem que seja preciso se preocupar como e onde essa ontologia está armazenada. Um sistema de gerenciamento de ontologias é para uma ontologia o

que um SGBD (Sistema de Gerenciamento de Banco de Dados) é para dados. O SNOBASE permite que ontologias sejam carregadas através de arquivos na Internet. Ontologias também podem ser criadas localmente. Internamente o SNOBASE utiliza um mecanismo de inferência, uma base de armazenamento de ontologias persistente, um diretório de ontologias e um conjunto de conectores que permite o acesso a fontes de ontologias distribuídas na rede.

Mais informações e detalhes sobre o SNOBASE podem ser obtidas no endereço: <http://www.alphaworks.ibm.com/tech/snobase>. Sendo que desde 2004, o SNOBASE migrou para um conjunto de ferramentas: IBM Integrated Ontology Development Toolkit.

II.03.04.01.08 - FaCT:

Podemos utilizar o FaCT (Fast Classification of Terminologies) para efetuar a verificação da consistência de ontologias. Ele possibilita a verificação automática ao mapear as ontologias para uma linguagem de lógica de descrição. Os serviços oferecidos pela ferramenta FaCT incluem detecção de inconsistências e identificação automática de relacionamentos taxonômicos. Um exemplo simples de classificação automática efetuada pelo FaCT é o seguinte: imaginemos que a classe Vaca, inerente à taxonomia dos seres vivos, tenha a seguinte restrição: “Come Vegetal”. Suponhamos também que a classe Herbívoro defina todos os animais que se alimentem apenas de vegetais. A partir disso, estabelecemos que essa classe seja necessária e suficiente para definir herbívoros, ou seja, todos os animais que se alimentam apenas de vegetais são herbívoros. Ferramentas que utilizem mecanismos de inferência também são conhecidas como classificadores e o FaCT, como já dissemos, pode efetuar essa classificação de forma automática. O FaCT pesquisará todas as classes e automaticamente classificará todas as classes de animais que se alimentem somente de plantas como pertencentes à classe Herbívoro.

Mais informações e detalhes sobre o FaCT podem ser obtidas no seguinte endereço: <http://www.cs.man.ac.uk/~horrocks/FaCT/>.

II.03.04.01.09 - Pellet:

Trata-se de um mecanismo de inferência para a linguagem OWL, construído em Java. Pode ser utilizado com a API Jena, descrita anteriormente. Essa ferramenta implementa algoritmos que manipulam expressões em lógica de descrição. Ela pode ser utilizada online, através da submissão da URI (Uniform Resource Identifier) de sua ontologia ou do texto. O Pellet faz parte do conjunto de ferramentas disponibilizado pelo grupo de Web Semântica da Universidade de Maryland, conhecido como MindSwap.

Mais informações e detalhes sobre o Pellet podem ser obtidas no seguinte endereço: <http://www.mindswap.org/2003/pellet/>.

Em seguida veremos as características dos agentes voltados à Web Semântica.

II.03.05 – Agentes

Neste capítulo serão expostas as características dos agentes de software.

Agentes são programas de software autônomos que agem em benefício de seus usuários. Um agente pessoal na Web Semântica, segundo Grigoris Antoniou e Frank Harmelen, vai receber algumas tarefas e preferências de um usuário, procurar informações nos recursos disponibilizados pela Internet (metadados e ontologias, por exemplo), se comunicar com outros agentes e comparar informações relativas às tarefas que deve desempenhar, de modo a fornecer respostas adequadas ao usuário.

Veremos, a seguir, os principais agentes.

II.03.05.01 – Principais Agentes

Neste capítulo serão expostas as características dos principais agentes:

II.03.05.01.01 - CATO (Componente para Alinhamento Taxônomico de Ontologias):

Segundo Karin Breitman, trata-se de uma iniciativa da PUC do Rio de Janeiro. O CATO (Componente para Alinhamento Taxonômico de Ontologias) é uma ferramenta para o alinhamento de ontologias em OWL disponível na Internet. O CATO foi construído baseado na visão de que a Web Semântica será composta por diversas aplicações com ontologias parciais e respectivas contextualizações.

II.03.05.01.02 - API Jena:

Não seria bem um agente, embora alguns autores classifiquem dessa forma. Ela, mais propriamente dita, pode ser utilizada para a construção de agentes. Mais informações sobre a API Jena foram detalhadas no item [II.03.04.01.05](#).

II.03.05.01.03 - Serviços Web Semânticos

Os Serviços Web Semânticos são utilizados como componentes de software distribuídos pela Internet e podem realizar tarefas complexas de aplicações. Eles podem ser modificados sem que as aplicações que os utilizam tenham que ser adaptadas. Os Serviços Web são popularmente conhecidos no Brasil pelo termo em inglês: *web services*. A natureza dos *web services* é a utilização do protocolo HTTP¹⁸ e com isso conseguimos que eles sejam utilizados por aplicações que sejam executadas em qualquer plataforma.

¹⁸ HTTP: é o acrônimo para *HiperText Transfer Protocol* que significa Protocolo de Transferência de Hipertexto e é um protocolo de comunicação muito utilizado na Web.

Entretanto, o uso desses serviços ainda oferece desafios: encontrar o serviço necessário; compor um conjunto desses serviços para atender a tarefas complexas e monitorar um desses serviços ou uma composição deles, com o intuito de garantir o sucesso na execução da tarefa.

Um exemplo de utilização é o seguinte caso: uma aplicação no domínio da adaptação de conteúdo. A adaptação é realizada por um serviço de tradução de linguagens de marcação. Baseia-se em informações sobre o ambiente de uso de dispositivos móveis, por exemplo: preferências do usuário, rede de acesso, contrato com o provedor de serviços, características do dispositivo, conteúdo a ser adaptado. Essas informações são descritas em ontologias e denominadas perfis, a fim de que páginas da Web possam ser acessadas por esses dispositivos [Santana et al., 2007].

Para facilitar o entendimento dos requisitos dessa aplicação, as Figuras 08 e 09 fornecem uma visão geral da sua arquitetura. Na Figura 08 um usuário acessa o serviço de adaptação de conteúdo através de um dispositivo móvel. Um [Proxy](#) de Adaptação tem o papel de interceptar requisições desse dispositivo para um Servidor de Conteúdo, que armazena conteúdo em seu formato original. Nesses Proxies estão localizados os Agentes Móveis, que utilizam Servidores de Adaptação disponíveis em Serviços Web Semânticos. Esses servidores realizam tradução de páginas HyperText Markup Language (HTML) para Wireless Markup Language (WML) e adaptações das imagens presentes nessas páginas.



Figura 08 – Adaptação Local de Conteúdo

Numa adaptação local, o fluxo de execução tem início após uma requisição de um usuário (1) ser interceptada por um Proxy de Adaptação. Em seguida, este Proxy de Adaptação requisita (2) e recebe (3), do Servidor de Conteúdo, o conteúdo a ser adaptado. De posse desse conteúdo, o Proxy de Adaptação delega ao seu agente a tarefa de realizar a adaptação necessária. Baseando-se na requisição e nos perfis, esse agente encontra, compõe e monitora a execução (4) dos Serviços Web Semânticos necessários.

Finalmente, o Proxy de Adaptação recebe (5), armazena em sua memória (*cache*), e envia os conteúdos adaptados ao Usuário (6).



Figura 09 – Adaptação Remota de Conteúdo

A Figura 09 considera o caso de uma adaptação remota, quando é necessário um Serviço Web Semântico que não está disponível no Proxy de Adaptação de Origem, nesse exemplo a adaptação de HTML para *Compact HTML* (cHTML¹⁹). A requisição do usuário é interceptada por um Proxy de Adaptação (1), o agente do Proxy da Adaptação de Origem migra para um Proxy de Adaptação Remoto (2), requisita (3) e recebe (4) o conteúdo a ser adaptado. No Proxy de Adaptação Remoto o Agente utiliza o Servidor de Adaptação necessário (5) e (6), realiza a adaptação e retorna ao Proxy de Adaptação de Origem (7) para enviar o conteúdo adaptado ao Usuário (8).

Existem diversos trabalhos que empregam ontologias, serviços Web e agentes de software para apoiar o processo de desenvolvimento de software. Alguns têm foco apenas nas ontologias como o de [Linhais e Moreira 2006], que utiliza linguagem natural para descrever os requisitos dos componentes de software, que posteriormente são transformadas em ontologias. A abordagem proposta utiliza os modelos de ontologias complementados por modelos em UML (Unified Modeling Language).

Em [Elias et al. 2006] é proposto um repositório de componentes que utiliza descrições sintáticas baseadas em XML para representação de metadados. A abordagem proposta utiliza a linguagem OWL-S - que tem semântica para descrever os componentes de Serviços Web Semânticos.

O trabalho de [Ha e Lee, 2006] utiliza Serviços Web Semânticos num ambiente específico de *e-business*. A abordagem proposta é genérica e também pode ser empregada em outros domínios de aplicações.

¹⁹ cHTML: acrônimo para *Compact HTML* – trata-se de um subgrupo do HTML para alguns telefones móveis.

Dentre os trabalhos que utilizam Agentes de Software para busca, composição e monitoramento de Serviços Web Semânticos destaca-se o de [Charif-Djebbar e Sabouret, 2006]. Este trabalho propõe uma abordagem baseada em sistemas multiagentes, na qual os agentes possuem um protocolo de interação que permite selecionar e compor os serviços. Na abordagem proposta os agentes podem migrar através dos diferentes nós da rede, para utilizar os Serviços Web Semânticos que estão distribuídos.

Poucos agentes foram listados, e isso porque, segundo a pesquisa realizada, a Web Semântica encontra-se ainda em sua fase embrionária ou muita classificação, através de ontologias, e qualificação, através de metadados, ainda precisam ser realizadas para que mais agentes sejam desenvolvidos e/ou efetivamente utilizados, como se deseja, em termos de autonomia e relevância. Inclusive o agente CATO, acima mencionado - o *site* relacionado pela literatura não apresentou disponibilidade no início de 2009 e por isso seu endereço não foi mencionado nem listado, no final deste trabalho, na bibliografia através de *sites* da Internet.

Mas não bastam classificações ou qualificações adequadas para utilizar agentes. E também não basta para que o conteúdo seja apropriado, considerarmos apenas os metadados e a classificação de conteúdo através de ontologias, precisamos também considerar a otimização dos mecanismos de busca para que tenhamos conteúdo com relevância. E isso porque também não podemos descartar os mecanismos de busca dos buscadores como sendo agentes voltados à Web Semântica. É o que veremos no capítulo a seguir.

II.04 – Como organizar o conteúdo para que os mecanismos de busca sejam mais eficientes

Neste capítulo serão expostas algumas das principais características que devem ser consideradas para a otimização dos mecanismos de busca. Essa otimização é mais conhecida pelo termo em inglês: SEO (*Search Engine Optimization*).

Não basta, para que o conteúdo seja apropriado, considerarmos apenas os metadados e a classificação de conteúdo através de ontologias, precisamos também considerar a otimização dos mecanismos de busca.

Segundo Jerri L. Ledford, um *site* bem otimizado e com ótimos *rankings* nos maiores buscadores, leia-se principalmente o Google, mas também o Yahoo, garante muitas visitas ao *site* pois hoje em dia o mais comum é que os internautas façam suas visitas a partir dos buscadores. No entanto, os buscadores não abrem explicitamente todos os seus critérios para *ranking* dos *sites*, pois muitos podem utilizar essas informações de critérios para exatamente burlar os buscadores e fazer com que o *site* obtenha melhores *rankings* artificialmente. Exemplo seria o Google deixar explícito que a quantidade de palavras na página que coincidam com a palavra-chave da busca seja um critério muito eficaz, o que até pode ser. Muitos poderiam colocar a palavra com fonte reduzidíssima no final da página simplesmente para melhorar seu *ranking*. Sendo assim, sem uma sequência de passos muito específica pois os buscadores não explicitam isso, podemos resumir os critérios de SEO, com ênfase na Web Semântica, com o que deve ser evitado, conforme segue:

- Priorizar busca antes do usuário:

A otimização é um processo que objetiva colocar um *site* nas listas dos resultados de buscas mais relevantes. A razão para perseguir com afinco o topo desta lista é atrair tráfego de relevância. Assim, é surpreendente que, na ânsia de obter esse resultado, muitos *sites* acabem esquecendo o seu público-alvo ou seus usuários.

Exemplo disso, seria aumentar o maior número possível de palavras-chave em cada página do *site* – isso pode garantir bons resultados no início, mas também poderá afugentar visitantes. Aos olhos do usuário, textos bem escritos ainda prevalecem sobre o *ranking* dos buscadores.

Melhor é a classificação e qualificação através de metadados que não prejudicam a navegação do usuário e melhoram o ranking do *site* em relação aos buscadores.

- *Link spamming*:

Trata-se do termo em inglês para fazer com que o *link* do *site* se espalhe e seja colocado em outras páginas (o que incluiria, até mesmo, a pouco recomendável prática de responder tópicos em fóruns e blogs só para ser referenciado). O fato aqui é que os *links* conseguidos são de pouca relevância ou de valor comercial reduzido. Os mecanismos de busca avaliam os *links* baseando-se na relevância do *site* que os inclui, de maneira que se utilizar do *link spamming* na verdade poderá rebaixar sua classificação ou, na melhor das hipóteses, não ajudará na qualificação.

O ideal é utilizar uma política mais eficiente utilizando-se de campanhas estrategicamente direcionadas, focando a construção de *links* em *sites* com afinidade e bem qualificados. A estratégia de focar os esforços em *sites* relevantes resulta numa maior probabilidade de também atingir a audiência desejada.

- Negligenciar *links* de texto:

O texto-âncora de cada *link* de uma página deve ser escrito o mais cuidadosamente possível, uma vez que os mecanismos de busca como o do Google são capazes de avaliar a qualidade e a quantidade dos *links* internos de um *site*. Textos sem imaginação do tipo ‘clique aqui’ não ajudam nem na experiência do usuário nem na busca.

- Não rastrear o progresso:

É praticamente impossível determinar a eficiência de uma otimização se não há marcação do ponto de partida. O rastreamento é frequentemente negligenciado, pois muitos pensam que basta acompanhar as listagens dos *sites* de buscas. Não é só isso: o número de vendas e as novas inclusões de busca são importantes. Mas não devem ser adotados como parâmetros exclusivos para determinar o sucesso da

estratégia de SEO. Um relatório periódico e confiável desses fatores pode determinar mudanças de rumo e verificar o impacto de alguma eventual alteração.

- Escolher palavras-chave ruins:

Alguns profissionais de SEO propõem garantias de resultados imediatos, mas eles se baseiam em palavras-chave ruins, com pouca afinidade ao *site* em específico. Isso pode propiciar resultados sem relevância. Exemplo disso seriam palavras-chave que envolvam nomes de produtos e modelos - elas são de uso restrito numa estratégia de otimização, pois tendem a ser muito pouco eficientes. Exemplificando: “lancha com cabine” é melhor que “*Brazilian Boat 230*”. Obviamente que os metadados devem refletir esse aspecto pois eles afetarão os resultados de busca também.

- Desconsiderar a acessibilidade:

Acessibilidade pode ser entendida em termos de disponibilizar o *site* para pessoas com algum tipo de deficiência, inclusive visual e essa preocupação pode ser um ponto crítico. Tornar seus *links* de texto bastante acessíveis e deixar o *site* de acordo com as normas do W3C só otimizarão o *site*, ou seja, permitirão melhorar os resultados do *site*, em todos os aspectos.

- Não criar uma estratégia de SEO:

Planejar uma campanha é análogo ao planejamento de uma viagem. Um dos erros mais comuns cometidos pelos incautos é se precipitar no lançamento do *site* e deixar de mapear os pontos de interesse pelo caminho. Em vez de focar no objetivo final, é muito melhor ou mais fácil, determinar objetivos de curto prazo que, juntos, formem o caminho para o tão desejado primeiro lugar no *ranking* de um Google, por exemplo.

III – Conclusão

Integrando os conceitos para oferecer conteúdo apropriado, vemos que um *site* que apresente metadados no padrão RDF, que nada mais é que um XML com [namespaces](#) adequados e que permite a devida identificação e qualificação do metadado tornará o conteúdo mais acessível e devidamente desenvolvido para que a Web Semântica e todos os seus subsequentes recursos sejam mais bem utilizados.

A utilização de ontologias ou a criação de uma ontologia utilizando o [Protégé](#), que é bastante amigável e permite o rápido desenvolvimento de uma ontologia são boas práticas ao conteúdo semântico, conforme foi exposto nesta dissertação. Mas reinventar a roda é uma opção que não será necessária na maioria dos casos, pois a utilização de ontologias já prontas, inclusive através de alinhamento e integração de ontologias, permite a devida classificação, hierarquização e relacionamentos de informações classificadas.

Também podemos utilizar agentes para disponibilizar conteúdo apropriado à miríade de usuários que um *site* de conteúdo pode apresentar. Ou então, para ajudar a relacionar esses dados de forma mais apropriada. E conforme a Web Semântica, com seus conceitos e recursos, estiver sendo cada vez mais utilizada e difundida podemos analisar a criação e oferta de agentes para que o conteúdo que for disponibilizado seja alcançado como se espera pelos agentes que oferecermos.

As boas práticas de otimização de mecanismos de busca, conforme foram expostas, também serão cruciais para que o conteúdo seja bem classificado em *sites* de busca como, por exemplo, o Google – fator primordial à Internet atual.

Os mecanismos de busca não possuem a habilidade de pensar, apenas seguem uma sequência de instruções ou regras dispostas em um algoritmo. Não é porque um ser humano consegue entender o assunto de um *site* que ele será listado em um buscador. Criar uma página que acerte exatamente as instruções do algoritmo do buscador pode levar a boas posições, mas ela poderá ser péssima para a experiência do usuário em termos de navegação na mesma.

Assim, existe um dilema na otimização de nosso conteúdo, mas uma boa regra, segundo Jerri L. Ledford, é sempre colocar os usuários em primeiro lugar. Seria ruim ter um excelente tráfego a partir de palavras-chave úteis para um *site* péssimo, pois ele perderá relevância aos usuários e poderá, com o tempo, deixar de ter um tráfego minimamente desejável.

Não existe uma sequência de passos pré-definida que atenda a todas as situações possíveis e não é o escopo dessa dissertação apresentar uma espécie de caminho único ou mágico que propiciará excelentes resultados. Ela permite que sejam conhecidas, discutidas ou refletidas as principais questões que a disponibilização de conteúdo apropriado oferece e nesse sentido apresenta diversas possibilidades de soluções que devidamente combinadas poderão ser bem-sucedidas - conforme o que foi exposto nos capítulos anteriores. Tudo forma um quadro que levará à realidade de uma Web Semântica, Web 3.0 ou Web do Futuro.

Um dos sonhos dos criadores da Web é a Web Semântica, proposta por Tim Berners-Lee, que, resumindo o que já foi explanando ao longo da dissertação, consiste em documentos organizados de tal forma que facilitem a coleta, pesquisa e cruzamento de informações de maneira automática e baseada no significado real do conteúdo desses documentos ao contrário da simples pesquisa por palavras-chave que é realizada pelas atuais ferramentas de busca. Embora muitas pessoas tenham dúvida em relação a expansão da Web Semântica, ninguém tem dúvidas de que ela dependerá muito da aplicação de padrões compreensíveis tanto para pessoas como para máquinas. E se pensarmos em uma Internet em que agentes forneçam opções inteligentes de conteúdo, muito pode e deve ser desenvolvido nessa parte, pois essa é a maior lacuna que vemos atualmente – talvez porque muito ainda precise ser efetivamente classificado para termos pesquisas semânticas com inteligência que nos levariam a inúmeras opções de escolha com pré-seleções realizadas pelos agentes. Enfim, um campo imenso para pesquisa e desenvolvimento está aberto.

A dissertação procurou mostrar os benefícios que poderiam ser obtidos se os computadores fossem capazes de “entender” as informações disponíveis na Web e, dessa forma, realizar parte do trabalho que grande parte dos internautas executa manualmente. Procuramos mostrar que o objetivo da Web Semântica é criar uma Web mais adequada às necessidades de seus usuários e que muita tecnologia tem sido

desenvolvida e disponibilizada de modo a permitir que se adicione semântica a páginas e aplicativos para realizar tudo isso.

Segundo Berners-Lee, a coordenação das diversas comunidades ou subculturas é “dolorosamente lenta e requer muita comunicação”. Claro que uma linguagem comum é essencial para esse processo. Talvez a Web Semântica seja capaz de fornecer essa linguagem. Segundo Berners-Lee, Hendler e Lassila, se conseguirmos projetar essa futura Web Semântica poderemos ajudar na evolução do conhecimento humano como um todo.

Notas

- Este trabalho foi escrito atendendo às normas do Novo Acordo Ortográfico vigente a partir de janeiro de 2009.

- Todas as páginas ou endereços Web discriminados neste trabalho estavam acessíveis durante o mês de fevereiro de 2009.

Glossário

Feed: O termo feed vem do verbo em inglês "alimentar". Na Internet, este sistema também é conhecido como "RSS Feeds" (RDF Site Summary ou Really Simple Syndication). Na prática, feeds são usados para que um usuário de internet possa acompanhar os novos artigos e demais conteúdos de um *site* sem que precise visitar o *site* em si. Sempre que um novo conteúdo for publicado em determinado *site*, o "assinante" do *feed* poderá ler imediatamente ao receber um arquivo no formato XML com o conteúdo adequado.

Heurística: Denomina-se heurística à capacidade de um sistema fazer inovações e desenvolver técnicas de forma imediata e positivas para um determinado fim.

Namespace: Espaço nominal ou mais popularmente conhecido no Brasil pelo termo em inglês: namespace é utilizado para eliminar a confusão de nomes ao combinar formatos. RDF é um bom exemplo de formato XML que utiliza espaço nominal. Nesse espaço nominal você pode evitar confusões com elementos ou atributos homônimos: “<p>” significa parágrafo ou pessoa? Você pode definir no namespace o que significa o elemento ou atributo utilizado ou definir um novo formato de documento, inclusive combinando ou reutilizando outros formatos.

Proxy: Servidor que atende a requisições e repassa os dados a outros servidores ou para usuários em geral. Eles podem filtrar conteúdo, verificar autenticação etc.

Tag: Etiqueta utilizada em páginas Web que pode conter alguma informação a respeito do documento ou que pode ser utilizada para uma linguagem para sua devida interpretação, como as etiquetas do padrão HTML, por exemplo: <head>, <body>, <a> etc.

Referências Bibliográficas

Impressas:

- Aguado, G. et al. – Ontogeneration reusing domain and linguistic ontologies for Spanish text generation – European Conference on Artificial Intelligence – Brighton, England, 1998.
- Antoniou, G.; Harmelen, F. – A Semantic Web Primer – Cambridge, Massachussets: MIT Press, 2004.
- Barceinas Guevara, A. - Un Marco de Comunicación Inter-Agentes en una Biblioteca Digital. Tesis Licenciatura. Ingeniería en Sistemas Computacionales. Departamento de Ingeniería en Sistemas Computacionales, Escuela de Ingeniería, Universidad de las Américas, Puebla. Mayo.
Derechos Reservados © - Universidad de las Américas, Puebla, 1998.
- Bateman, J.; Magnini, B.; Fabris, G. – The generalized upper model knowledge base organization and use – Conference on Knowledge Representation and Sharing – The Netherlands, 1995.
- Berners-Lee, T.; Lassila, O.; Hendler, J. – The Semantic Web – Scientific American, 2001.
- Bond, F. et al. – Design and construction of a machine – tractable Japanese-Malay lexicón – Annual Meeting of the Association for Natural Language Processing, 2001.
- Breitman, Karin Koogan – Web Semântica: a Internet do Futuro – Rio de Janeiro: LTC, 2005.
- Buitelaar, Paul – Semantic lexicons: between ontology and terminology – Proceedings of Ontolex: Ontologies and Lexical Knowledges Bases 2000. OntoText Lab.: Sofia, Bulgaria, 2001.

- Charif-Djebbar, Y; Sabouret, N. - “Dynamic Service Composition and Selection through an Agent Interaction Protocol”. Anais da International Conference on Web Intelligence and Intelligent Agent Technology, p. 105-108, 2006.
- Eco, Umberto - A Busca da Língua Perfeita – 2ª edição – Bauru/SP: EDUSC, 2001.
- Eco, Umberto – Como se faz uma Tese – 20ª edição – São Paulo: Editora Perspectiva, 2006.
- Elias, G.; Schuenck, M.; Negócio, Y.; Dias, J.; Filho, M. - “X-ARM: an asset representation model for component repository systems” Anais do Symposium on Applied Computing, p. 1690 – 1694, 2006.
- Fernández-López, M. et al. – Building a chemical ontology using methotology and the ontology design environment. – IEEE Intelligent Systems & their Applications, p. 37-46, 1999.
- Frederking, R.; Nirenburg, S. – Three heads are better than one, Proceedngs of the Fourth Conference on Applied Natural Language Processing, Stuttgart, Germany, 1994.
- Gómez-Pérez, A.; Corcho, O. – Ontology Languages for the Semantic Web. The IEEE Intelligent Systems, January/February, 2002.
- Guimarães, F. Z. - Utilização de ontologias no domínio B2C – Dissertação de Mestrado – PUC-RJ, 2002.
- Ha, Y.; Lee, R. – “Integration of Semantic Web Service and Component-Based Development for e-business environment” Anais da International Conference on Software Engineering Research, Management and Applications, p. 315 – 323, 2006.
- Heflin, J.; Hendler, J.; Luke, S. – SHOE: A Knowledge Representation Language for Internet Applications. Technical Report CS-TR-4078 (UMIACS TR-99-71), 1999.

- Hendler, J. – Agents and the Semantic Web – IEEE Intelligent Systems, 2001.
- Horrocks, I.; Sattler, U.; Tobies, S. – Practical reasoning for expressive description logics. *Logic Journal of the IGPL*, v. 8, n. 3, p. 239-264, May, 2000.
- Ledford, Jerri L. – Search Engine Optimization Bible – 1st edition - Indianapolis: Willey Publishing Inc., 2008.
- Linhalis, F.; Moreira, D.A. - Ontology-Based Application Server to the Execution of Imperative Natural Language Requests. *Anais do International Conference on Flexible Query Answering Systems*, p. 589-600, 2006
- Luger, George F. – Inteligência Artificial: estruturas e estratégias para a solução de problemas complexos – 4^a edição – Porto Alegre: Bookmann, 2004.
- Maedche, A. – Ontology Learning for the Semantic Web – 1^a edição – Kluwer Academic Publisher – 272p., 2002.
- Matos, L. et al. – Segredos do Google – São Paulo: Digerati Books, 2004.
- Miller, George A. – WordNet: a lexical database for English *Communications of the ACM*, v. 38, n. 11, p. 39-41, 1995.
- Moreno, A. O.; Hernández, C. P. - Reusing the Mikrokosmos Ontology for concept-based multilingual terminology databases. – *International Conference on Language Resources & Evaluation* - Athens, 2000.
- Noy, N.; Musen, M. The Prompt Suite: Interactive Tools For Ontology Merging And Mapping. *International Journal of Human-Computer Studies*, 2003.
- Ogino, T. et al. – An experiment on matching EDR concept classification dictionary with WordNet. *IJCAI-97 Workshop WP24 Ontologies and Multilingual*, 1997.

- Rösner, D.; Stede, M. – Techdoc: multilingual generation of online and offline instructional text. Disponível em: <http://acl.ldc.upenn.edu/A/A94/A94-1044.pdf>, 1994.
- Salam, A. F./Stevens, Jason R. – Semantic Web Technologies and E-Business – University of North Carolina – e-book – 2007.
- Santana, L.; Prado, A.; Souza, W.; Biajiz, M. – Usando Ontologias, Serviços Web Semânticos e Agentes Móveis no Desenvolvimento Baseado em Componentes – UFSCar – São Paulo/SP, 2007.
- Silva, A. C. – Gestão de Conhecimento: linguagem, forma e impacto na comunicação em redes de informação – Tese de Doutorado em Comunicação e Semiótica – São Paulo: PUC/SP, 2005.
- Teich, E.; Bateman, A.; Maier, A.; Wanner, L. - Towards an architecture for situated text generation - International Conference on Current Issues in Computational Linguistics, Penang, Malaysia. Também disponível em: Technical Report of GMD/Institut für Integrierte Publikations- und Informationssysteme, Darmstadt, Germany, 1991.
- [Towards the Semantic Web: Knowledge Representation in a Dynamic Distributed Environment](#), by Jeff Heflin. Ph.D. Thesis, University of Maryland, College Park, 2001.
- Vossen, P. – EuroWordNet: A Multilingual Database with Lexical Semantic Networks, Kluwer Academic Publishers, Dordrecht, 1998.
- Wiley, John - Towards the Semantic Web – Ontology-Driven Knowledge Management – England, 2003.

Sites da Internet:

- Almeida, M.; Bax, M. – Uma visão geral sobre ontologias: pesquisa sobre definições, tipos, aplicações, métodos de avaliação e de construção – Disponível em:
<http://revista.ibict.br/index.php/ciinf/article/viewArticle/17/11>.
- Beale, S.; Nirenburg, S.; Mahesh, K. – Semantic analysis in the mikrokosmos machine translation project – Proceeding of the Second Symposium on Natural Language Processing – Disponível em:
<http://crl.nmsu.edu/users/sb/papers/thai/thai.html>.
- Chimaera – Disponível em:
<http://www.ksl.stanford.edu/software/chimaera/>.
- DAML – Disponível em: <http://www.daml.org/>.
- Doctorow, C. – Metacrap: Putting the Torch to Seven Straw Men of the Meta Utopia - Disponível em:
<http://www.well.com/~doctorow/metacrap.htm>.
- FaCT – Disponível em:
<http://www.cs.man.ac.uk/~horrocks/FaCT/>.
- Germann, U. - Making semantic interpretation parser-independent – Proceedings of the AMTA Conference, 1998. Disponível em:
<http://www.isi.edu/natural-language/people/germann/>.
- Informações e *downloads* sobre Raimundo Lúlio e a Ars Magna:
<http://lullianarts.net/downloads.htm>; <http://www.ramonlull.net>.
- Informações sobre o padrão MARC:
<http://www.loc.gov/marc>.
- Jena (API Java) – Disponível em: <http://jena.sourceforge.net>.
- Milstead, J.; Feldman, S. – Metadata cataloging by any other name. On Line Magazine, New Haven, v. 23, n. 1, 1999.

Disponível em:

<http://www.infoday.com/nom99/nomprogram.htm>.

- Noy, F. N.; McGuinness, D. L. - Ontology development 101: a guide to create your first ontology. Stanford University, USA, 2002.

Disponível em:

<http://ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness.doc>.

- Pellet – Disponível em: <http://www.mindswap.org/2003/pellet/>.
- OIL – Disponível em: <http://www.ontoknowledge.org/oil/>.
- Ontology.org – <http://www.ontology.org>.
- OWL – Disponível em: <http://www.w3.org/TR/owl-ref/>.
- Protégé – Disponível em: <http://protege.stanford.edu/>.
- RDF – W3C RDF Primer – Disponível em: <http://www.w3.org>.
- SHOE – Disponível em:
<http://www.cs.umd.edu/projects/plus/SHOE/>.
- Simplicity Media Survey – Disponível em:
<http://simplicity.media.mit.edu>.
- SNOBASE – Disponível em:
<http://www.alphaworks.ibm.com/tech/snobase>.

Anexos

Anexo 01 - Exemplo de Arquivo utilizando SHOE:

```
<HTML>

<HEAD>
<TITLE>Princeton SHOE Data</TITLE>
<META HTTP-EQUIV="SHOE" CONTENT="VERSION=1.0">
</HEAD>

<BODY>

<A HREF="http://www.cs.umd.edu/projects/plus/SHOE">
<IMG SRC="http://www.cs.umd.edu/projects/plus/SHOE/images/SHOEpage.gif" ALT=""
VALIGN=TOP ALIGN=RIGHT BORDER=0>
</A>

<H1>Princeton Summary Data</H1>
<P>This page contains SHOE information for the
<A HREF="http://www.cs.princeton.edu/">Princeton Department of Computer
Science</A>. The SHOE data includes:

<UL>
<LI>Basic university information
<LI>Basic department information
<LI>Faculty
</UL>

<INSTANCE KEY="http://www.cs.princeton.edu/">

<USE-ONTOLOGY ID="cs-dept-ontology" VERSION="1.0" PREFIX="cs" URL=
"http://www.cs.umd.edu/projects/plus/SHOE/onts/cs1.0.html">
<CATEGORY NAME="cs.University" FOR="http://www.princeton.edu/">
<RELATION NAME="cs.name">
<ARG POS="1" VALUE="http://www.princeton.edu/">
<ARG POS="2" VALUE="Princeton University">
</RELATION>
<CATEGORY NAME="cs.Department" FOR="http://www.cs.princeton.edu/">
<RELATION NAME="cs.name">
<ARG POS="1" VALUE="http://www.cs.princeton.edu/">
<ARG POS="2" VALUE="Princeton Department of Computer Science">
</RELATION>
<RELATION NAME="cs.subOrganization">
<ARG POS="1" VALUE="http://www.cs.princeton.edu/">
<ARG POS="2" VALUE="http://www.princeton.edu/">
</RELATION>

</INSTANCE>

<INSTANCE KEY="http://www.cs.princeton.edu/fac/">

<USE-ONTOLOGY ID="cs-dept-ontology" VERSION="1.0" PREFIX="cs" URL=
"http://www.cs.umd.edu/projects/plus/SHOE/onts/cs1.0.html">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~appel">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~arora">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~august">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~chazelle">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~doug">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~prc">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~dpd">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~felten">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~af">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~funk">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~bwk">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~li">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~sahai">
```

```

<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~rs">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~jps">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~mona">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~ken">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~ret">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~rywang">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~wayne">
<CATEGORY NAME="cs.Faculty" FOR="http://www.cs.princeton.edu/~yao">

<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~appel">
  <ARG POS="2" VALUE="Andrew Appel">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~arora">
  <ARG POS="2" VALUE="Sanjeev Arora">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~august">
  <ARG POS="2" VALUE="David August">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~chazelle">
  <ARG POS="2" VALUE="Bernard Chazelle">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~doug">
  <ARG POS="2" VALUE="Douglas Clark">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~prc">
  <ARG POS="2" VALUE="Perry Cook">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~dpd">
  <ARG POS="2" VALUE="David Dobkin">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~felten">
  <ARG POS="2" VALUE="Edward Felten">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~af">
  <ARG POS="2" VALUE="Adam Finkelstein">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~funk">
  <ARG POS="2" VALUE="Thomas Funkhouser">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~bwk">
  <ARG POS="2" VALUE="Brian Kernighan">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~li">
  <ARG POS="2" VALUE="Kai Li">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~sahai">
  <ARG POS="2" VALUE="Amit Sahai">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~rs">
  <ARG POS="2" VALUE="Robert Sedgewick">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~jps">
  <ARG POS="2" VALUE="Jaswinder Singh">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~mona">
  <ARG POS="2" VALUE="Mona Singh">

```

```
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~ken">
  <ARG POS="2" VALUE="Kenneth Steiglitz">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~ret">
  <ARG POS="2" VALUE="Robert Tarjan">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~rywang">
  <ARG POS="2" VALUE="Randy Wang">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~wayne">
  <ARG POS="2" VALUE="Kevin Wayne">
</RELATION>
<RELATION NAME="cs.name">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/~yao">
  <ARG POS="2" VALUE="Andrew Yao">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~appel">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~arora">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~august">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~chazelle">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~doug">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~prc">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~djd">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~felten">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~af">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~funk">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~bwk">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~li">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~sahai">
```

```
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~rs">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~jps">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~mona">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~ken">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~ret">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~rywang">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~wayne">
</RELATION>
<RELATION NAME="cs.member">
  <ARG POS="1" VALUE="http://www.cs.princeton.edu/">
  <ARG POS="2" VALUE="http://www.cs.princeton.edu/~yao">
</RELATION>

</INSTANCE>

</BODY>

</HTML>
```

Anexo 02 – Tabela Comparativa entre as linguagens OIL e DAML:

Similaridades	Diferenças
Ambas oferecem suporte a hierarquias de conceitos e propriedades, baseadas nos relacionamentos de subclasse e subpropriedade.	A OIL oferece melhor compatibilidade com RDF-Schema do que DAML.
Ambas permitem que conceitos sejam construídos a partir de outros conceitos utilizando combinações dos conectivos OR, AND e NOT.	A OIL foi projetada de modo a permitir a inferência de modo completo e eficiente. Algumas construções em DAML fazem com que a inferência em DAML se torne praticamente impossível.
Ambas permitem que sejam definidos o domínio, o escopo e a cardinalidade das propriedades que vão compor as restrições dos conceitos.	Em OIL podemos explicitar condições que sejam suficientes ou necessárias e suficientes. A última opção permite a classificação automática de conceitos.
Ambas oferecem propriedades transitivas e inversas.	DAML permite a especificação de valores <i>default</i> , ou seja, valores que serão inicializados se nenhum outro for definido pelo modelador da ontologia. A OIL, no entanto, não permite a utilização de valores <i>default</i> , pois não existe uma semântica formal clara para tratá-los.
Ambas oferecem apoio a tipos de dados concretos (inteiros, cadeias etc.).	

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)