**UNIVERSIDADE FEDERAL DE SANTA CATARINA**
**PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**

**Ricardo Flores Zago**

# MODELO DE RECUPERAÇÃO E INDEXAÇÃO DE CONHECIMENTO EM DOCUMENTOS CORPORATIVOS ANOTADOS SEMANTICAMENTE

Dissertação de Mestrado submetida à Universidade Federal de Santa Catarina como parte dos requisitos para obtenção do grau de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Fernando Álvaro Ostuni Gauthier

Florianópolis, agosto de 2005

# MODELO DE RECUPERAÇÃO E INDEXAÇÃO DE CONHECIMENTO EM DOCUMENTOS CORPORATIVOS ANOTADOS SEMANTICAMENTE

**Ricardo Flores Zago**

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação Área de Concentração Sistemas de Conhecimento e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

_____

Prof. Raul Sidnei Wazlawick, Dr.

Banca Examinadora

_____

Prof. Fernando Ostuni Gauthier, Dr. (orientador)

_____

Prof. Nilson Ribeiro Modro, Dr.

_____

Prof. João Bosco Mangueira Sobral, Dr.

_____

Prof. Rogério Cid Bastos, Dr.

*Para os meus pais, Ari Zago e Regina Flores Zago. Vocês são meus eternos mestres.*

**Agradecimentos**

Agradeço a Deus pelas coisas boas que proporciona em minha vida. E ao Menino Jesus de Praga, o qual dedico uma forte devoção.

Aos meus pais, Ari e Regina e ao meu irmão Roberto, pelo carinho e apoio nas horas mais difíceis. Vocês são a razão do meu esforço.

À Jaqueline, pelo seu "amor sem fronteiras" e por sempre me fazer acreditar.

A todos os meus familiares, os quais nos afastamos na distância, mas jamais no coração. Em especial à tia Orilde, minha madrinha.

Ao professor Gauthier, pela orientação, por ter acreditado em mim desde o início desta caminhada e pelo incentivo na realização do trabalho.

Ao grande amigo Rafael Speroni, por nunca ter medido esforços para auxiliar diretamente na concepção deste trabalho e pela parceria no samba.

À Verinha e à Dayane, por cuidarem tão bem da nossa secretaria e pelos cafezinhos sempre deliciosos.

Às amizades cultivadas em Florianópolis, que fizeram com que minha passagem por esta terra fosse inesquecível. Aos "irmãos" do apartamento, Adriano e Renato, minha família em Floripa. Ao Régis, Speroni, Lília, Adamô, Marcelle, Márcio, Deise, Neves, Cássia, Guidão, Beto, Ana, Barreto, Michel, Fabian, Carol, sempre prontos para um churrasco.

Ao pessoal de Brasília, Rodolfo, Irla, Moc, Luciana, Cristiano, Michelle, em especial ao Balzan, com quem dividi algo que pode ser até chamado de apartamento.

Aos amigos de Santa Maria, que me recebem sempre de braços abertos, Richard, Débora, Anderson, Fafi, Marcelo, Paula, Christian, Roberta, em especial ao "pessoal do industrial", Marco, Cris, Bolão, Fabiane, Zuca, Nininha, Igor, Renata, Claiton, Borowski.

A todos aqueles que, direta ou indiretamente, participaram da realização deste trabalho.

**Sumário**

# Lista de Figuras

# Lista de Tabelas

# Lista de Siglas

| | |
|---|---|
| API | Aplication Programmer Interface |
| DAML | DARPA Agent Markup Language |
| OIL | Ontology Inference Layer |
| OWL | Ontology Web Language |
| PHP | Hypertext Preprocessor |
| NLP | Natural Language Processing |
| RDF | Resource Description Framework |
| RDFS | Resource Description Framework Schema |
| URI | Uniform Resource Identifier |
| URL | Uniform Resource Locator |
| WWW | World Wide Web |
| W3C | World Wide Web Consortium |
| XML | Extensible Markup Language |

# Resumo

A tarefa de recuperação de conhecimento tem se tornado cada vez mais penosa devido ao crescimento excessivo de documentos e à desordem com que são mantidos nas organizações. A Web Semântica traz uma nova conceituação no que diz respeito à recuperação de conhecimento, visando inserir conteúdo compreensível por computadores nos documentos. O presente trabalho objetiva a criação de um modelo de recuperação do conhecimento corporativo por meio da extração de conteúdo semanticamente anotado, presente nos documentos da organização. O modelo proposto é aplicado num ambiente de trabalho de uma instituição onde os documentos são mantidos de forma desordenada na sua rede interna. Os documentos da instituição foram submetidos a um processo de anotação semântica para, através do modelo proposto, serem recuperados posteriormente. O modelo proposto trouxe maior objetividade na recuperação do conhecimento do setor e um melhor entendimento do ambiente de trabalho por parte dos funcionários através da ontologia desenvolvida.



Palavras-chave: Web Semântica; Conhecimento; Recuperação de Informação

# Abstract

The knowledge recovery task has becoming hard due to excessive document growing and to disorder in that are maintained in organizations. The Semantic Web bring us a new conceptualization about knowledge recovery aiming to insert computer comprehensive content into documents. The present work objects the creation of an enterprise knowledge recovery model by means of the extraction of semantic annotations presents in enterprise documents. The proposed model is applied in a work environment of an institution where the documents are maintained disordered in the intranet. The documents generated by the istitution had been submitted to a process of semantic annotation for, through the considered model, to be recovered later. The proposed model brought greater objectivity in the recovery of the knowledge of the sector and a better agreement of the environment of work by the employees through the developed ontology.



Keywords: Semantic Web; Knowledge; Information Retrieve

# 1 Introdução

## *1.1 Considerações iniciais*

O rápido crescimento da Web trouxe consigo a geração de uma quantidade excessiva de informações disponíveis para serem compartilhadas. Tal crescimento fez com que os documentos contendo estas informações fossem dispersos de forma desordenada nos servidores de todo o planeta, acarretando numa difícil e confusa localização destes documentos.

Atualmente, as empresas tratam e disponibilizam o seu conhecimento de várias maneiras, tais como manuais, cartas e formulários de resposta a clientes, além do conhecimento adquirido através de processos de trabalho. O gerenciamento do conhecimento nas empresas é realizado com o auxílio de diversas ferramentas de tecnologia de informação, entre elas, gerenciadores de correio eletrônico, bancos de dados, *datawarehouses*, navegadores e mecanismos de busca. Frente a este quadro, pode-se afirmar que o desafio dos gerentes e gestores de empresas para gerenciar conhecimento em suas corporações é tornar este conhecimento acessível e reutilizável para as mesmas (O'LEARY, 1998).

O diferencial competitivo gerado pelo acesso às informações faz com que as corporações busquem ferramentas que auxiliem nos processos gerenciais, ocasionando, assim, a necessidade de investimentos em ferramentas de tecnologia de informação, que normalmente acarretam em custos financeiros expressivos. Nos últimos anos, surgiu uma comunidade mundial que está se opondo aos produtores de software proprietário, como forma de protesto à não liberação do código fonte dos programas por estes e também contra os preços abusivos que cobram pela tecnologia desenvolvida. A comunidade de Software Livre entende que a arte da programação não deveria se tornar a atividade lucrativa a qual se tornou, atividade esta considerada imoral pelo criador da comunidade, Richard Stallman. Para ele, as licenças de software os tornavam presos às características que a empresa que os criou impunham em seu código. Desta forma, um usuário de software proprietário não pode acrescentar ao sistema uma característica necessária somente para seu domínio de aplicação, pois não tem acesso ao código fonte.

O compartilhamento do código não restringiria a liberdade dos usuários (FERRAZ, 2002).

Dentre as ferramentas utilizadas para edição de textos construídas e distribuídas sob a política de software livre, o pacote de ferramentas para escritório OpenOffice (OPENOFFICE.ORG) merece uma atenção mais cuidadosa frente ao destaque que vem tendo sobre as outras por sua semelhança com o Microsoft Office (OFFICE.MICROSOFT.COM) – pacote proprietário de mesma finalidade mais vendido e utilizado (TALLIS, 2003). Esta ferramenta caminha ao lado dos sistemas operacionais quando as corporações decidem abandonar a plataforma proprietária e aderir a sistemas livres, reduzindo seus investimentos na área de Tecnologia da Informação sem perder a produtividade.

A Internet atual visa oferecer uma grande rede de informação que possa ser compartilhada por pessoas do mundo inteiro. Este objetivo foi alcançado com a criação do serviço World Wide Web. O “www”, como ficou conhecido, surgiu em torno de três idéias que tornariam a rede eficiente: distribuição geográfica e física de documentos, localização não-ambígua destes documentos e interface de acesso a estes documentos uniformizada. Com estes princípios, foram desenvolvidas três ferramentas, cada uma delas para realizar uma das idéias: um protocolo de transmissão (http), um sistema de endereçamento próprio (URLs) e uma linguagem de marcação (HTML).

Desta forma, montou-se um cenário em que pessoas pudessem acessar conteúdos geograficamente distantes, separados por cidades ou países. O que se tinha até o momento eram compartilhamentos de documentos em um mesmo prédio, apenas da organização a qual os documentos pertenciam.

Estes documentos compartilhados, através da linguagem HTML, necessitam que pessoas façam o trabalho de formatação e ligação entre eles, o que pode ser considerada a parte difícil do trabalho. Aos computadores, cabe a tarefa de exibi-los de acordo com a formatação definida. Pode-se afirmar que esta seria a parte mais fácil.

Neste ponto, surgiu a iniciativa de fazer com que os computadores façam a parte mais difícil do trabalho, ou seja, os computadores devem entender o conteúdo dos documentos. A esta iniciativa, somou-se a idéia de acrescentar conteúdo semântico aos documentos dispostos na Web, formando, assim, o que se chamou de Web Semântica.

## *1.2 Justificativa*

A tarefa de gerenciar conhecimento em corporações tem se tornado cada vez mais árdua devido ao volume excessivo de informação gerado em períodos de tempo cada vez menores. O acesso a tais informações depende de uma classificação robusta e de mecanismos de busca que localizem rapidamente a informação necessária.

Um processo semelhante ocorre com a informação disposta na Web onde, da mesma forma, a quantidade de dados é acrescida diariamente. A Web Semântica está sendo considerada a evolução da Web atual (LEE, 2001) para fazer com que o conteúdo dos documentos seja compreendido por computadores.

Alguns dos esforços envolvidos na concepção da Web Semântica visam tornar possível a disponibilização e recuperação de informações em documentos levando em conta a sua relevância semântica dentro do contexto em que estão inseridas, de forma automatizada, robusta e compartilhada.

Este trabalho visa utilizar as técnicas que vêm sendo empregadas na aplicação dos conceitos da Web Semântica para auxiliar empresas no gerenciamento do seu conhecimento, através de um modelo de recuperação de conhecimento que seja compreendido pelos computadores dispostos na rede corporativa.

## *1.3 Objetivos*

### 1.3.1 Objetivo geral

Este trabalho tem como objetivo geral propor uma arquitetura eficiente e escalável para indexar e recuperar conhecimento presente em documentos corporativos, envolvendo o a mineração e armazenamento de metadados semânticos e a realização de buscas baseadas em ontologias.

### 1.3.2 Objetivos específicos

- Identificar técnicas de mineração de metadados em documentos anotados semanticamente;

- Apresentar um modelo de indexação de documentos com anotação semântica baseada em ontologias;
- Desenvolver uma ferramenta para recuperação de documentos com conteúdo semanticamente anotado que permita consulta sem conhecimento prévio de Web Semântica;
- Implementar um protótipo utilizando documentos no padrão OpenOffice com anotação semântica na linguagem OWL.

## *1.4 Organização do trabalho*

O presente trabalho está dividido em 6 capítulos, da seguinte forma:

- O **primeiro capítulo** apresenta uma introdução ao trabalho desenvolvido, bem como seus objetivos e a justificativa para o estudo;
- No **segundo capítulo**, será apresentada uma tecnologia da área de Inteligência Artificial conhecida como Web Semântica e como ela é utilizada para recuperar informação através da classificação ontológica, bem como uma abordagem sobre como as empresas adquirem e tratam o seu conhecimento;
- No **terceiro capítulo**, o modelo proposto neste trabalho é apresentado, identificando os diferentes módulos necessários para a sua implementação;
- A implementação do modelo e implantação deste num ambiente corporativo é apresentada no **quarto capítulo**, onde também são descritos os resultados após a sua utilização;
- O **quinto capítulo** relaciona as conclusões relativas ao trabalho, além de sugestões para trabalhos futuros;
- Finalmente, é apresentada a bibliografia consultada para realização do trabalho.

# 2 Web Semântica e gestão do conhecimento

## *2.1 Considerações iniciais*

Atualmente, os documentos dispersos na *web* utilizam-se de linguagens de formatação de texto (HTML, por exemplo), para que o conteúdo seja entendido pelos usuários. Como grande parte deste conteúdo é disponibilizado em linguagem natural, ocorre uma lacuna entre a informação disponibilizada para leitura humana e a utilizada por mecanismos de busca (MOURA, 2001).

A Web Semântica é considerada a evolução da *web* atual, fornecendo estrutura ao conteúdo significativo das páginas da *web*. "A Web Semântica habilitará máquinas a compreender documentos semânticos, não voz e escrita humana" (LEE, 2001).

Para GUHA (2003), a Web Semântica conterá recursos que não somente correspondam a objetos digitais (páginas da web, imagens ou vídeos) como a *web* atual, mas também objetos do mundo real, como pessoas, lugares ou eventos.

A Web Semântica é uma extensão da Web onde os documentos são anotados com meta-informação, que define quais informações ele contém (DAVIES, 2003). Esta meta-informação acompanhada de alguma teoria de domínio (ontologias, por exemplo) será capaz de formar uma *web* que fornecerá um novo nível de serviços.

O par formado entre a meta-informação e uma ontologia faz com que os recursos sejam dispostos na web de forma mais abrangente. Isto faz com que mecanismos de recuperação de conteúdo atuem de forma mais precisa e com maior qualidade em suas tarefas (MOURA, 2001).

Frente a esta abordagem, surgiram vários padrões de metadados direcionados a domínios específicos de conhecimento que, quando aplicados, mostraram-se mais eficientes na elaboração de resultados em consultas que técnicas atuais empregadas em ferramentas de consulta (MOURA, 2001).

## *2.2 A pirâmide de linguagens da Web Semântica*

Berners-Lee, ao idealizar a Web Semântica, visualizou o que ele mesmo chamou de “Pirâmide de Linguagens”, apresentada na figura 1.

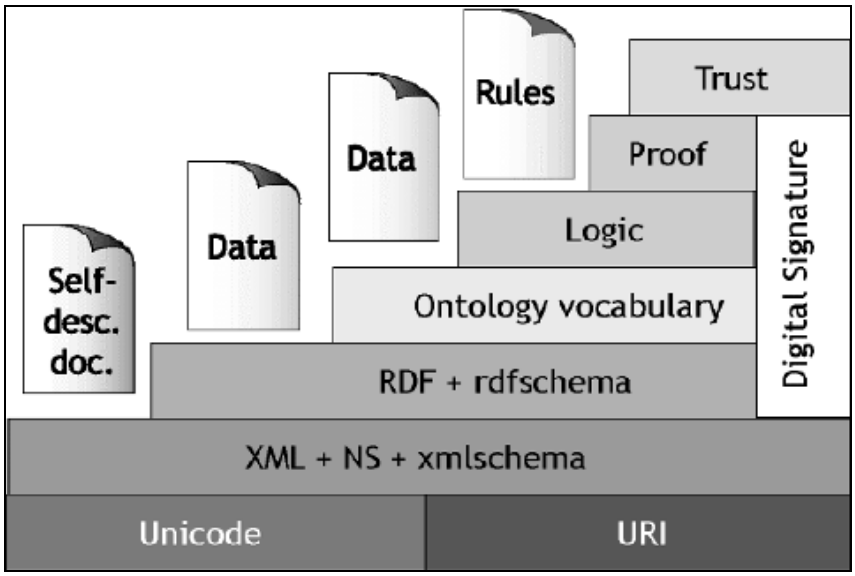


**Figura 1 - Pirâmide de Linguagens da Web Semântica (LEE, 2000)**

A primeira camada da pirâmide (Unicode/URI) fornece o meio para referenciar entidades e identificar recursos. A segunda camada (XML/NS/XMLSchema) torna a informação processável por computadores. Estas duas camadas servem como base para que a Web Semântica possa ser realizada (GOBLE, 2002).

A partir da terceira camada a Web Semântica começa a surgir. A camada RDF + RDFSchema é responsável pela representação dos metadados necessários para a descrição de recursos. A quarta camada já traz uma forma de eliminar ambigüidades entre conceitos, através da definição de termos nas ontologias. Nesta camada, percebe-se a evolução das linguagens das camadas anteriores, em busca de acrescentar à Web Semântica a capacidade de raciocínio. Surgem aí linguagens como DAML+OIL e sua evolução, a OWL (GOBLE, 2002).

As últimas camadas da pirâmide servem para testar a confiabilidade dos resultados. Na camada de lógica são feitas inferências de acordo com a base de conhecimento. A seguir, a camada de prova testa as deduções para que, na última camada, os fatos deduzidos sejam considerados confiáveis.

### 2.2.1 Camada 1: Unicode + URI

O padrão Unicode (UNICODE.ORG) foi adotado por empresas como IBM, Apple, Microsoft e HP com a finalidade de identificar através de um número único cada caractere, independente da plataforma que se utiliza.

O Unicode é utilizado pelos Uniform Resource Identifiers (URI), que são cadeias de caracteres compactas para identificar um recurso físico ou abstrato (LEE, 1998b). Os URIs foram criados para que os conteúdos sejam identificados de forma única na Web, sendo caracterizados (segundo LEE, 1998b) por:

**Identificador**: Um identificador é uma seqüência de caracteres com sintaxe restrita que faz referência a alguma coisa que tenha identidade.

**Uniforme**: A uniformidade fornece vários benefícios, como permitir que diferentes tipos de identificadores de recursos sejam usados no mesmo contexto, permitir a introdução de novos identificadores de recursos sem interferir nos existentes e permitir a interpretação semântica uniforme de convenções sintáticas nos diferentes tipos de identificadores.

**Recurso**: Um recurso pode ser observado como qualquer documento eletrônico, imagem, serviço ou coleção de outros recursos que possua identidade.

Tipos mais específicos de URI podem ser classificados como um localizador, um nome ou ambos. O tipo de URI mais conhecido é o URL (Uniform Resource Locator), que identifica o recurso através da representação de seu mecanismo primário de acesso – em outras palavras, sua localização na Internet. Ainda convém citar o Uniform Resource Name, ou URN, tipo menos conhecido de URI que difere do URL porque mantém a persistência do recurso mesmo quando este deixa de existir ou se torna indisponível.

### 2.2.2 Camada 2: XML + NS + xmlschema

A sigla XML vem do nome “eXtensible Markup Language”, que traduzida para o português significa Linguagem de marcação extensível.

Segundo DEITEL (2003), “XML é uma tecnologia para criar linguagens de marcação para descrever dados de virtualmente qualquer tipo de uma maneira

estruturada". Sua sintaxe é semelhante ao HTML, que também é uma linguagem de marcação, mas seu objetivo é outro. Enquanto HTML trata da formatação dos dados para sua exibição em navegadores, a XML fornece estrutura a dados de diversas naturezas.

Um documento escrito em XML pode conter um conjunto infinito de marcações, enquanto na linguagem HTML este conjunto de marcações é limitado. Isto é possível devido à linguagem XML permitir que os programadores criem suas próprias marcações (LEE, 2001). Por marcações entende-se que são rótulos ocultos que demarcam seções de texto num documento.

MYLLYMAKI (2001) afirma que "no futuro, alguns, senão a maior parte do conteúdo da *web* pode estar disponível em formatos mais adequados para processamento automatizado, em particular a XML".

O exemplo mostrado na Figura 2 tem como objetivo ilustrar um documento XML simples.

```
<?xml version = "1.0"?>
<mensagem>
  <texto>Exemplo XML</texto>
</mensagem>
```

**Figura 2 - Documento XML simples**

A primeira linha contém apenas o prólogo, uma declaração para que o documento identifique qual a versão da linguagem XML que foi utilizada na escrita.

Todo documento XML deve conter exatamente um elemento *raiz*, no exemplo identificado por `<mensagem>`. Além disso, todo elemento possui um marcador final (no exemplo, `</mensagem>`). Ao contrário da HTML, onde pode haver marcadores sem o correspondente final, em XML este é obrigatório. Os demais elementos estarão contidos no elemento raiz.

O elemento `<texto>` é dito filho do elemento `<mensagem>` e contém o texto (dado) "Exemplo XML".

Em determinados momentos, pode haver o que se chama de colisão de nomes, onde dois elementos XML são descritos com o mesmo nome, mas possuem significados

diferentes. Isto ocorre devido à característica da linguagem XML de permitir que os próprios autores dos documentos criem suas marcações (DEITEL, 2003).

Para que os autores de documentos XML possam tratar a colisão de nomes de forma segura, foi criado o conceito de *NameSpaces* (espaços de nomes). Por exemplo, os elementos `<folha>Tripla</folha>` e `<folha>A4</folha>` identificam diferentes dados em um documento XML, sendo que o primeiro encontra-se num domínio de partes de uma árvore e o segundo identifica o tipo de papel a ser utilizado num documento. Para que não haja colisão, os elementos podem ser reescritos da seguinte forma: `<arv:folha> Tripla </arv:folha>` e `<doc:folha> A4 </doc:folha>`.

Os prefixos “arv” e “doc” acima escritos são os prefixos de espaços de nome do documento. Cada prefixo é vinculado a um URI que identifica de forma única o *NameSpace* previamente definido pelo autor do documento. Assim, para os exemplos acima, os NameSpaces poderiam ser definidos da seguinte forma:

```
<raiz  xmlns = “http://www.aa.com/”
       xmlns:arv = “http://www.bb.com/arvore#”
       xmlns:doc = “http://www.cc.com/documento#”>
   <elemento1></elemento1>
   <elemento2></elemento2>
   <doc:folha>A4</doc:folha>
   <arv:folha>Tripla</arv:folha>
</raiz>
```

### 2.2.3 Camada 3: RDF + rdfschema

“Resource Definition Framework (RDF) é uma linguagem baseada em XML para descrever a informação contida em um recurso” (DEITEL, 2003). Um recurso pode ser uma página, um site inteiro ou qualquer item na Web que contém informação em algum formato.

Para DEITEL (2003), RDF é “informação sobre informação”, ou seja, utiliza o conceito de metadados para que mecanismo de busca ou agentes possam catalogar ou listar conteúdo da Web.

Enquanto XML é o formato de documentos padrão recomendado pelo Consórcio WWW (W3C) para escrita e troca de informação na Web, RDF é o modelo padrão também recomendado pelo W3C para descrever semântica e raciocínio sobre informações na Web (PATEL-SCHNEIDER, 2002).

RDF expressa o significado do conteúdo, codificado em conjuntos de triplas, cada tripla contendo o sujeito, o verbo e o objeto de uma sentença. Com isto, podem-se fazer declarações que coisas particulares (pessoas, páginas da Web) têm propriedades ("é irmão de", "é autor de") com certos valores (outra pessoa, outra página) (LEE, 2001).

KLYNE (2004) enumerou diversos conceitos acerca de RDF. Um destes conceitos é o Graph Data Model (Figura 3), onde as triplas são representadas como ligações nó-arco-nó, com o predicado (também conhecido por propriedade) definindo o relacionamento entre o sujeito e o objeto.

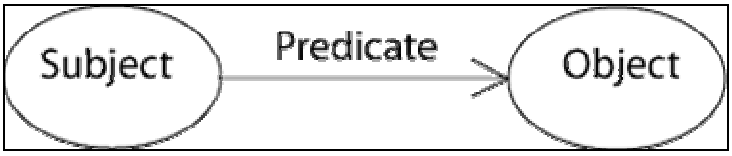


**Figura 3 - Graph Data Model (KLYNE, 2004)**

DAVIES (2003) escreve sentenças RDF como uma tripla objeto-atributo-valor, representada como A(O, V), onde um objeto O possui um atributo A de valor V. O autor também define sentenças RDF como um arco entre dois nós ([O] – A → [V]). Assim definidos, as sentenças abaixo, no formato A(O, V), podem ser representadas de acordo com a Figura 4:

```
temNome("http://www.ufsc.br/alunos/id2345", "João Silva")
alunoDe("http://www.ufsc.br/alunos/id2345", "http://www.ufsc.br/cursos/id123")
```

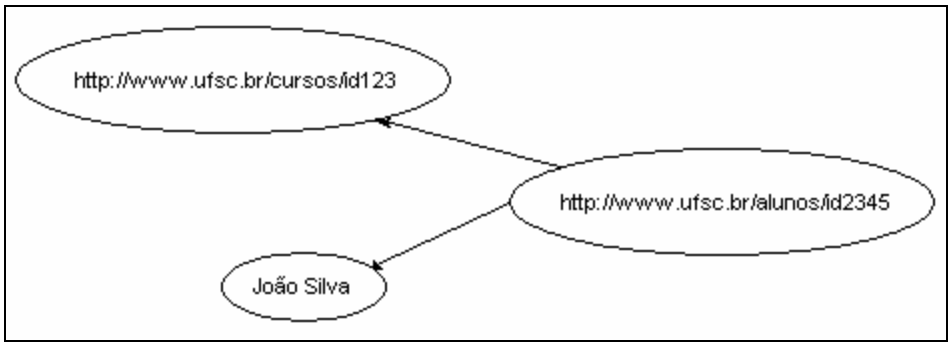


**Figura 4 - Representação gráfica de uma sentença RDF**

Com o conceito de RDF, os pesquisadores tinham a capacidade de descrever relações entre recursos, mas não conseguiam armazenar certas propriedades destes

recursos. Esta habilidade foi realizada com a criação dos RDF Schemas (BRICKLEY, 2000).

RDF Schema é a linguagem de descrição do vocabulário, ou seja, uma extensão semântica do RDF que fornece mecanismos para descrever grupos de recursos relacionados e as relações entre estes recursos (BRICKLEY, 2004).

Segundo DAVIES (2003), "com RDFS pode-se falar a respeito de classes, subclasses, subpropriedades, domínios e restrições de intervalos de propriedades, num contexto baseado na Web".

Na próxima seção será apresentado o conceito de ontologia, um recurso importante para o funcionamento da Web Semântica no que diz respeito à classificação do conhecimento.

## *2.3 Ontologias*

Para que a Web Semântica faça sentido, é necessário fazer com que os computadores entendam o significado dos conteúdos. O instrumento utilizado para esta finalidade chama-se Ontologia.

O conceito ontologia tem base na Filosofia, e está relacionado a teorias a respeito da natureza da existência das coisas, que tipo de coisas existem (LEE, 2001). É um conceito ligado a taxonomias. O termo foi estudado por várias comunidades de pesquisadores de Inteligência Artificial desde o início da década de 90, tornando-se mais comum recentemente nas suas áreas ligadas ao gerenciamento de conhecimento (DAVIES, 2003).

Uma das definições que se tornou bastante popular foi dada por GRUBER (1993), dizendo que "uma ontologia é uma especificação explícita de uma conceitualização".

Para MOURA (2001), ontologia é a "descrição explícita e precisa de conceitos e relações que existem em um domínio particular". LEE (2001) utiliza a mesma definição, mas já visando seu uso final para Web Semântica, considera ontologia "um documento ou arquivo que define formalmente as relações entre os termos".

DAVIES (2003) conceitua ontologia como um “entendimento comum e compartilhado de um domínio que pode ser comunicado entre pessoas e sistemas de aplicação”.

O uso de ontologias na Web Semântica tem o objetivo de descrever termos de determinados domínios para evitar ambigüidades. Isto se deve à possibilidade de dois bancos de dados identificarem o mesmo conceito de maneira diferente (LEE, 2001).

### 2.3.1 Linguagens para representação de ontologias

Segundo MOURA (2001), “ontologias provêm o mecanismo formal capaz de viabilizar o processamento semântico da informação através de uma máquina”. Para representação de ontologias, foram criadas diversas linguagens. Uma característica importante destas linguagens é a representação em RDF/RDFS.

#### 2.3.1.1 DAML+OIL

A linguagem de representação de ontologias DAML+OIL surgiu da fusão da linguagem DAML-ONT (DARPA Agent Markup Language) com componentes da linguagem OIL (CONNOLY, 2001).

A versão inicial da linguagem DAML-ONT trazia consigo a integração de RDF (que possui XML embutido) com RDF Schema, o que a tornava altamente compatível com os padrões da Web e permitia a interoperabilidade com diversas ferramentas que vinham sendo desenvolvidas sob estes padrões (MCGUINNESS, 2002).

A linguagem OIL resultou de diversas pesquisas na área de Lógicas de Descrição (MCGUINNESS, 2002), que são formalismos para representação de conhecimento onde interpretações fornecem significado, e estas interpretações fornecem a semântica formal da lógica (GRAU, 2004). Neste sentido, segundo MCGUINNESS (2002), “pesquisadores desenvolveram OIL para ser uma lógica de descrição expressiva integrada com uma tecnologia da Web moderna”.

A fusão das duas linguagens foi um processo natural dada a semelhança dos objetivos das duas linguagens – que é a geração de conteúdo compreensível por computadores – através do aproveitamento do que cada uma delas possuía de melhor (MCGUINNESS, 2002).

#### 2.3.1.2 OWL

MCGUINNESS (2004) justifica o uso da OWL (Ontology Web Laguage) devido ao fato de ser uma linguagem designada a suprir uma necessidade das linguagens para representação de ontologias, que é ir além das semânticas básicas do RDF Schema. Assim, o autor conclui que OWL possui mais vocabulário para descrever classes e propriedades do que XML, RDF e RDF Schema.

Segundo a recomendação do Consórcio WWW (MCGUINESS, 2004), existem três sublinguagens de OWL designadas para que comunidades específicas de implementadores possam utilizá-la de acordo com suas necessidades:

**OWL Lite**: Designada para usuários que necessitam primeiramente de recursos e restrições simples da linguagem.

**OWL DL**: Foi definida para usuários que utilizem todos os recursos da linguagem OWL, mas que desejam assegurar-se da integridade computacional. Isto quer dizer que a utilização da linguagem deve obedecer a algumas restrições. A denominação DL vem do campo de pesquisa na qual a linguagem é baseada (lógicas de descrição).

**OWL Full**: Esta sublinguagem permite o uso de todas as funcionalidades da linguagem sem nenhum tipo de restrição, permitindo uma maior liberdade ao usuário com a restrição de não apresentar nenhuma garantia computacional.

Cada uma destas sublinguagens é uma extensão da predecessora, o que quer dizer que toda ontologia OWL Lite válida é uma ontologia DL válida, e toda OWL DL válida é uma ontologia OWL Full válida.

#### 2.3.1.3 Evolução das linguagens para representação de ontologias

A Web Semântica começou a fazer sentido com o aparecimento da linguagem XML. Porém, esta ainda não foi considerada uma linguagem completa para representação de ontologias. Isto acarretou no aparecimento da linguagem RDF, com o conceito de descrição de recursos. Uma das primeiras linguagens utilizadas para representar ontologias no contexto da *web* foi a RDFS, que estendeu a RDF no sentido de conceitualizar classes, subclasses e propriedades, tornando as ontologias próximas à orientação a objetos.

Apesar de a linguagem RDFS introduzir primitivas básicas de modelagem de ontologias na Web (DAVIES, 2003), seu uso ficou restrito por algumas condições que ela não atendia:

- Facilidade de entendimento, pois a RDFS é muito confusa;
- Especificação formal, pois a RDFS é uma extensão da RDF;
- Possibilidade de fornecer suporte a raciocínio

Para atender a estas condições, duas linguagens começaram a ser desenvolvidas.

Na Europa, um grupo de pesquisadores que já atuava em projetos relacionados ao conhecimento criou a linguagem OIL, sigla para *Ontology Inference Layer* (ONTOKNOWLEDGE.ORG/OIL), enquanto que nos Estados Unidos, o grupo de pesquisas militares DARPA, que já havia desenvolvido a linguagem DAML (DAML.ORG), criou a DAML-ONT.

Como estas duas linguagens eram semelhantes e possuíam a mesma finalidade, os dois grupos uniram seus esforços criando uma linguagem chamada DAML+OIL que foi submetida ao *World Wide Web Consortium* como uma base para padronização de escrita de ontologias. A partir deste ponto, foi criado um grupo chamado *Web-Ontology (WebOnt) Working Group* (W3.ORG/2001/SW/WEBONT) para ficar responsável pela padronização em linguagens para ontologias. Este grupo, baseado na linguagem DAML+OIL, criou a OWL, que é hoje reconhecida como padrão do W3C para escrita de ontologias.

A Figura 5 ilustra a evolução das linguagens para representação de ontologias.

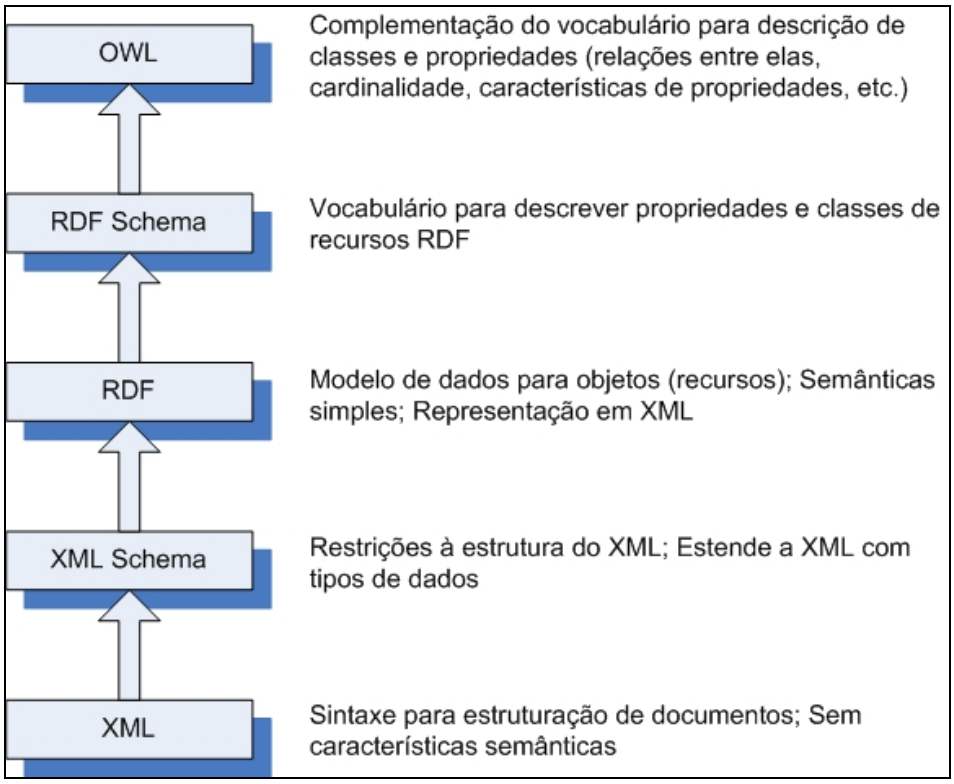


**Figura 5 - Evolução das linguagens para representação de ontologias**

### 2.3.2 Anotação semântica

A Web Semântica acrescenta metadados ao conteúdo de documentos, visando descrevê-lo de acordo com a ontologia especificada. Estes metadados são responsáveis por fornecer semântica aos documentos, tornando-os compreensíveis por computador.

O processo de anotação semântica consiste em duas partes distintas:

- Geração de metadados específicos dos documentos, para que se possa associar seus conceitos e entidades às descrições semânticas definidas através de ontologias;
- Inclusão destes metadados no conteúdo do documento.

KIRYAKOV (2003) define anotação semântica como um “esquema específico para geração de metadados e uso”. O termo esquema, citado na definição do autor, refere-se a *entidades nomeadas*, que podem ser descritas como pessoas, localizações, organizações ou outras entidades que são referenciadas pelo nome em documentos de texto. Para o autor, o processo de anotação semântica consiste em atribuir às entidades que aparecem no documento ligações com suas descrições semânticas, sendo aplicado a

qualquer tipo de texto (documentos HTML, documentos de texto comuns, campos de banco de dados, entre outros).

Na Figura 6, podem ser observadas as anotações semânticas referentes a um texto que pode ser de um documento disponível na Web.

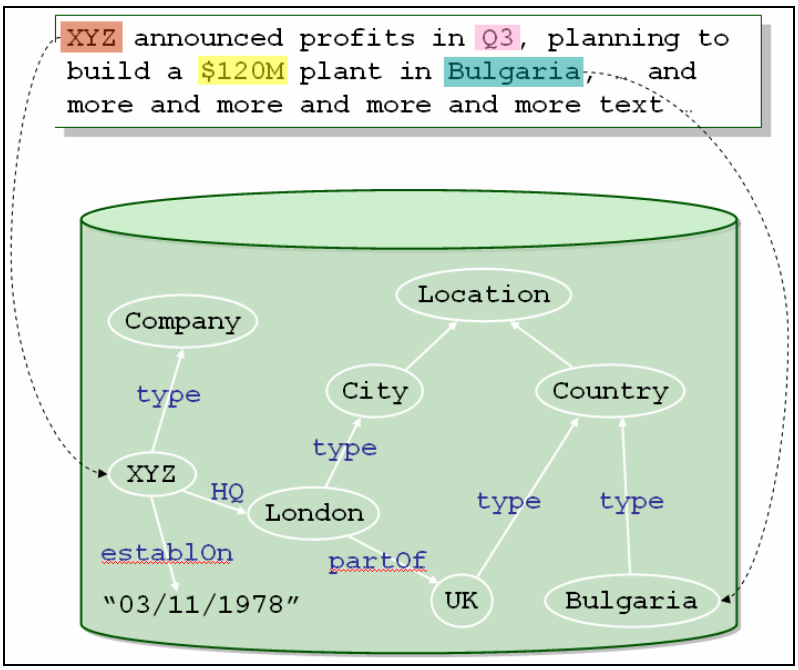


**Figura 6 - Exemplo de marcação semântica, onde as entidades presentes no texto são associadas à sua definição (KIRYAKOV, 2003)**

Segundo KOGUT (2001), existem pelo menos três tipos de ferramentas que podem ser utilizadas para anotação em documentos:

**Semi-automática**: Associa palavras a classes e propriedades utilizando-se do julgamento humano. Esta associação geralmente é efetuada através de interfaces "arraste-e-solte". A ferramenta Ont-O-Mat é um exemplo deste tipo de anotador.

**Automática**: Aplicam técnicas de Processamento de Linguagem Natural (NLP) para associar palavras a classes e propriedades. Estas ferramentas podem utilizar ontologias padrão (por exemplo, IEEE Standard Upper Ontology) ou ontologias de domínios específicos (Unified Medical Language System (UMLS))

**Híbrida**: Utiliza as definições de anotação semântica semi-automática e automática para combiná-las em uma só ferramenta, ou seja, pode utilizar tanto o julgamento humano quanto técnicas de NLP para determinar as associações de palavras com classes e propriedades.

Na próxima seção será explanada a forma como o conhecimento disponibilizado na *web* é acessado através de agentes.

## *2.4 Agentes*

LEE (2001) afirma que “o poder real da Web Semântica será concretizado quando as pessoas criarem programas que coletem o conteúdo da Web de diversas fontes, processarem a informação e compartilharem os resultados com outros programas”. Estes programas capazes de entender o conteúdo dos documentos são conhecidos como Agentes.

Agentes são projetados para perceber seu meio através de sensores e interagir com o mesmo através de seus atuadores (RUSSEL, 2004). Portanto, espera-se que ele possa agir sobre o ambiente de forma autônoma e se adaptar a novas situações para alcançar o melhor resultado.

Os agentes racionais são os que selecionam as suas ações conforme seu conhecimento interno, limitado a seu ambiente e possuindo pouca capacidade de aprendizado (RUSSEL, 2004). Já os agentes inteligentes distinguem-se pela capacidade de tomar iniciativas para alcançar seus objetivos, e também pela capacidade de interagir com outros agentes e até mesmo com pessoas, aumentando, assim sua precisão nas decisões (FERBER, 1998).

Estas decisões tomadas a cada momento pelos agentes são, muitas vezes, baseadas em incertezas. Este fator é o que determina a capacidade dos agentes para atingir seus objetivos (RUSSEL, 2004).

Frente ao exposto, pode-se definir um agente de software como um programa de computador que explora um ambiente dinâmico de acordo com um objetivo específico, de interesse de outra entidade, humana ou computacional.

### 2.4.1 Mecanismos de busca na web

ARASU (2001) descreve uma arquitetura genérica de mecanismos de busca para *web*, mostrada na Figura 7:

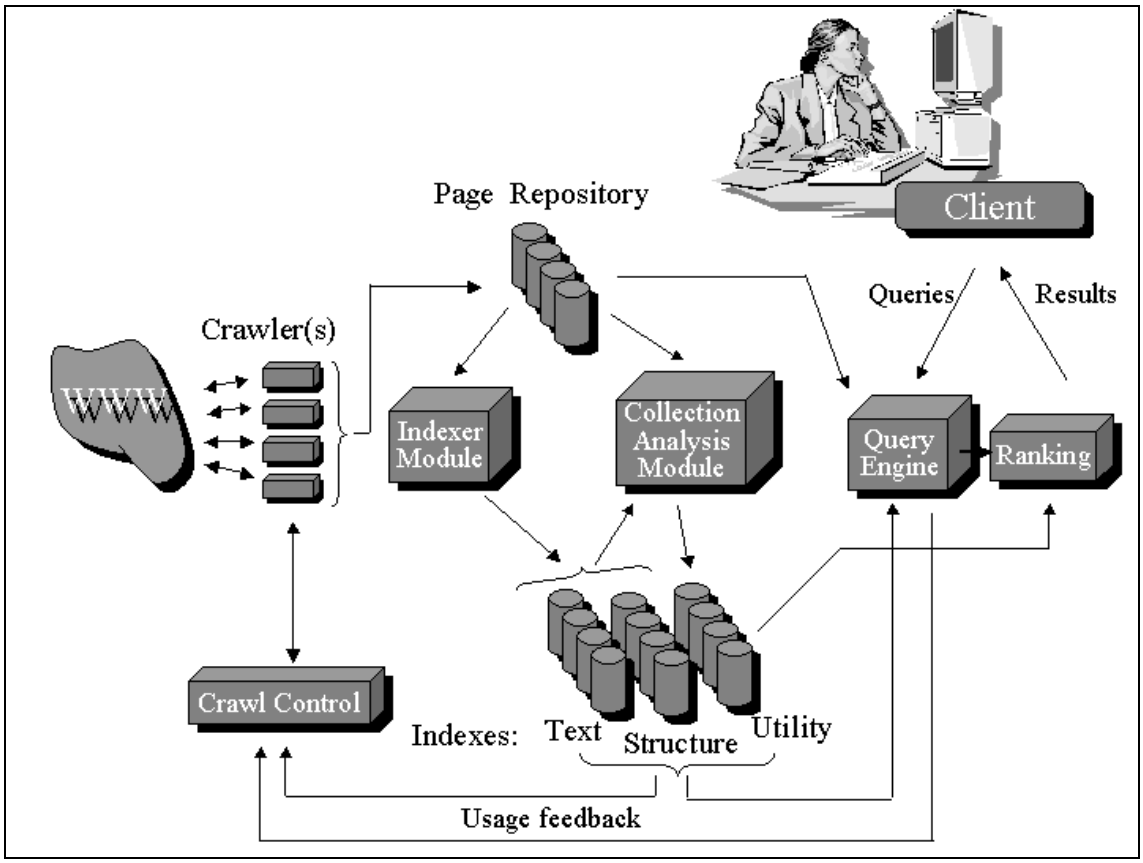


**Figura 7 - Arquitetura geral de mecanismos de busca (ARASU, 2001)**

Para que seja efetuada a busca, um conjunto inicial de URLs de documentos é informado ao mecanismo. Este faz uma varredura nos documentos iniciais e coleta os URLs que aparecem como ligações (*links*) no conteúdo destes documentos. Os URLs coletados são encaminhados ao módulo chamado *Crawl Control*. Este módulo é responsável por definir quais documentos serão os próximos a serem visitados para que sejam coletados novos URLs. Cada *crawler* trata o final deste processo de acordo com seus objetivos, ou seja, cada *crawler* possui um algoritmo diferente que determina quando este processo deve parar, implementado no seu *Crawl Control*.

Os documentos visitados e retornados para o *crawler* são armazenados num módulo chamado *Page Repository*. Este armazenamento ocorre de forma completa, o que significa que cada documento visitado é armazenado na sua íntegra. O resultado deste armazenamento pode ser uma coleção de todas as páginas disponíveis na Web com seus conteúdos, dependendo de como o *Crawl Control* gerencia o final do processo (neste caso, a varredura ocorre de forma exaustiva).

A partir deste ponto, entra em ação o *Indexer Module*. Ele é responsável por extrair todas as palavras de cada página e gravar o URL onde a palavra ocorreu. Isto gera uma grande tabela capaz de retornar onde cada palavra ocorreu. Esta tabela é representada na ilustração através do índice "*Text*" gerado pelo *indexer module*.

O *indexer module* também cria um outro índice chamado "S*tructure*", devido ao tamanho do índice *text* e à sua rápida taxa de mudança. O índice structure armazena todas as ligações entre as páginas, pois estas ligações acresceriam ainda mais o índice *text* dado que existem inúmeras páginas que não contém ligações, somente texto puro.

Outros índices necessários ao processo de busca na Web são definidos pelo *Collection Analysis Module*. Este módulo utiliza os índices *text* e *structure* para gerar novos índices com informações que possam ser relevantes ao processo de busca. Na ilustração, é apresentado um índice chamado *utility* que foi criado pelo *Collection Analisys Module* para responder com mais rapidez a questões como tamanho das páginas, ou classificá-las por importância, ou mesmo pelo número de imagens que nelas aparecem.

O módulo chamado *Query Engine* é responsável por receber as consultas e retornar os resultados para o usuário. Este módulo busca por resultados nos índices criados, e raramente utiliza o *page repository* nas suas consultas. A função do móduo *ranking* é organizar os resultados de acordo com critérios pré-estabelecidos de importância para que sejam exibidos numa determinada seqüência aos usuários.

### 2.4.2 Crawlers

Os agentes de software que buscam páginas em servidores web e as armazenam para uso posterior através de um motor de busca são denominados *crawlers* (CHO, 2002). A Web Semântica utiliza este tipo de agente para vasculhar a *web* em busca de conteúdo semântico presente nos documentos.

ARASU (2001) define *crawlers* como "pequenos programas que vasculham a Web de acordo com as regras do motor de busca, semelhante a como uma pessoa segue *links* para visitar diferentes páginas".

O processo de rastreamento de páginas Web pode ser visto como um grafo de problema de busca, onde a *web* é um grande grafo tendo as páginas como nós e as

ligações como arcos. O *crawler* inicia a busca por um conjunto de poucos nós, seguindo os arcos para chegar a outros nós (PANT, 2004).

QIN (2004) acrescenta o conceito de *crawlers* direcionados, que são programas designados a retornar páginas w*eb* seletivamente, relevantes a um domínio específico e para uso de um mecanismo de busca específico para este domínio. Ao contrário dos *crawlers* dos motores de busca mais comuns, que extraem todas as páginas ao seu alcance, os *crawlers* direcionados tentam inferir quando uma página é relevante ao domínio antes de buscá-la.

A forma como o conhecimento é adquirido e gerenciado pelas corporações será apresentada na próxima seção, bem como a utilização da Web Semântica na tarefa de gerenciamento do conhecimento.

## *2.5 Conhecimento nas corporações*

### 2.5.1 Gerenciamento de conhecimento

O conhecimento, segundo SUNASSEE (2003), pode ser definido como “a opinião humana armazenada na mente de uma pessoa, ganha com a experiência e interação com o seu ambiente”.

Existem dois tipos de conhecimento, o tácito e o explícito. O conhecimento tácito é aquele desenvolvido através da experiência, entendido e aplicado subconscientemente, sem formalismos para ser transmitido e compartilhado através de conversações. Já o conhecimento explícito é o conhecimento formal e sistemático, o que faz com que possa ser facilmente transmitido.

SUNASSEE (2003), depois de analisar os conceitos para gerenciamento de conhecimento elaborados por diversos autores, definiu gerenciamento do conhecimento como “processo de identificação, crescimento e aplicação efetiva do conhecimento existente de uma organização a fim de alcançar seus objetivos”.

O conhecimento pode ser visto como o capital intelectual de uma empresa. Muitos gerentes percebem que a propriedade deste conhecimento é a parcela do patrimônio de

uma organização responsável pela diferenciação frente a seus competidores (LIEBOWITZ[1] apud SUNASSEE, 2003).

Para que o conhecimento adquirido pelas organizações seja aproveitado, é necessária sua reutilização de forma adequada. Esta tarefa é denominada gerenciamento do conhecimento da empresa, que envolve o controle formal de recursos do conhecimento através do uso de tecnologia de informação avançada, com o objetivo de facilitar o acesso e reuso do conhecimento (O'LEARY, 2003).

De acordo com BABILON (1998), grandes empresas sofreram, nos anos 90, uma vasta "explosão" de informação interna. Uma destas empresas, a NCR Corporation, foi objeto de estudos do autor, onde este constatou que o gerenciamento de conhecimento trouxe um diferencial competitivo para a corporação, pois os processos desenvolvidos pela empresa para este fim contribuíram para que o acesso à informação fosse facilitado.

### 2.5.2 Utilização da Web Semântica para gerência do conhecimento

Estudos recentes remetem a Web Semântica como uma importante ferramenta para gerência de conhecimento, pois integra a definição de ontologias que podem ser lidas por computador e ligações de fragmentos de texto com estas ontologias (TALLIS, 2003).

A ligação entre o documento e as ontologias é conhecida como anotação semântica. Esta tarefa é considerada a chave para que as técnicas da Web Semântica sejam corretamente empregadas.

Para que a Web Semântica possa ser utilizada como instrumento de organização do conhecimento, e conseqüentemente como acelerador na busca de resultados, algumas ferramentas vem sendo desenvolvidas em estudos acadêmicos ou patrocinados por grandes empresas interessadas na utilização da Web Semântica como forma de gerenciar seu conhecimento.

---

[1] LIEBOWITZ, J. **Building Organizational Intelligence: A Knowledge Management Primer**. CRC Press, Boca Raton, 2000.

Uma das ferramentas em destaque é a AeroDAML, que efetua marcação de conhecimento para gerar anotações em linguagem DAML+OIL através da aplicação de técnicas de extração de informação de linguagem natural (KOGUT, 2001).

AeroDAML é um exemplo de uma tecnologia emergente na área de Web Semântica conhecida como IES, que são Sistemas de Extração de Informação (Information Extraction Systems). Esta tecnologia visa diminuir o esforço humano na tarefa de anotação de documentos, implementando técnicas de extração automática de conhecimento.

Um recente estudo realizado por TALLIS (2003) resultou numa ferramenta para anotação semântica chamada SemanticWord. Foi desenvolvida para que autores de documentos em ambiente Microsoft Word possam gerar, simultaneamente, conteúdo e anotação semântica.

A ferramenta estende o MS Word de várias maneiras, entre elas:

- Oferece barras de ferramentas para suportar a criação de anotações semânticas, mostrar estas anotações enquanto o texto é digitado e manipular estas anotações com o mouse;
- Estende a pesquisa nos documentos no sentido de incorporar conceitos de Web Semântica e integração dos documentos com esta.
- Acrescenta um novo tipo de serviço para o Microsoft Word (análogo ao corretor gramatical ou ortográfico) quando integra um sistema de extração de informação (AeroDAML) para analisar e gerar a anotação no momento em que o conteúdo é digitado.

A Figura 8 ilustra as extensões oferecidas ao MS Word.

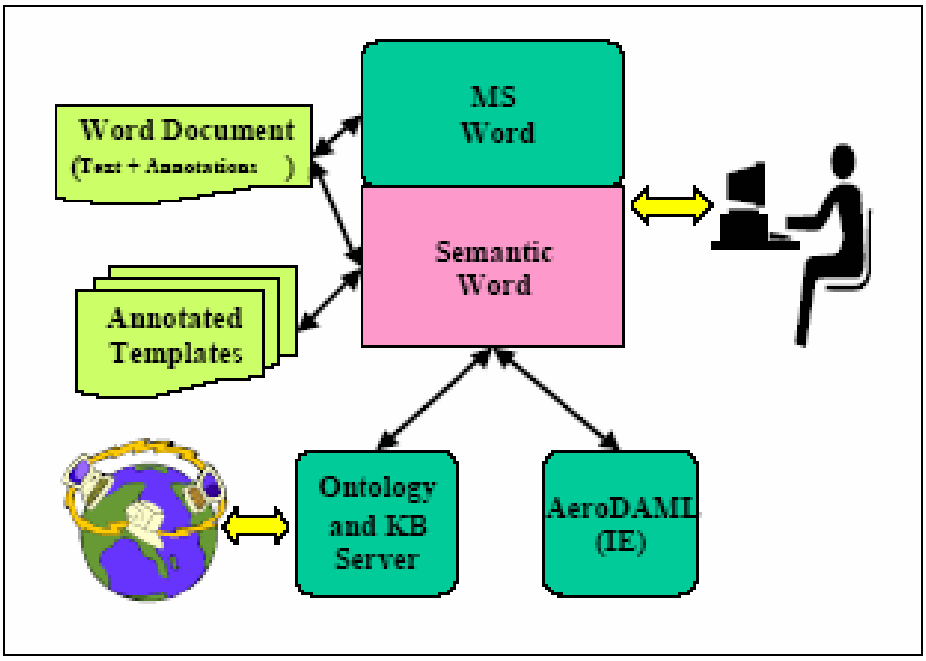


**Figura 8 - Semantic Word, uma ferramenta para anotação de conteúdo no MS Word (TALLIS, 2003)**

As anotações geradas pelo Semantic Word são baseadas na linguagem de definição de ontologias DAML+OIL.

Um dos primeiros sistemas para anotação semântica foi o Ont-O-Mat (hoje evoluído para OntoMat-Annotizer), um *framework* para geração de conteúdo e anotação desenvolvido para facilitar a criação de metadados relacionais. Possui seu próprio editor HTML (Figura 9) para geração e visualização do conteúdo e um navegador de ontologias que faz a leitura da ontologia para que o autor tome conhecimento das definições de classes, propriedades e instâncias do domínio em questão. Seu funcionamento baseia-se em comandos "*arraste-e-solte*" com o mouse.

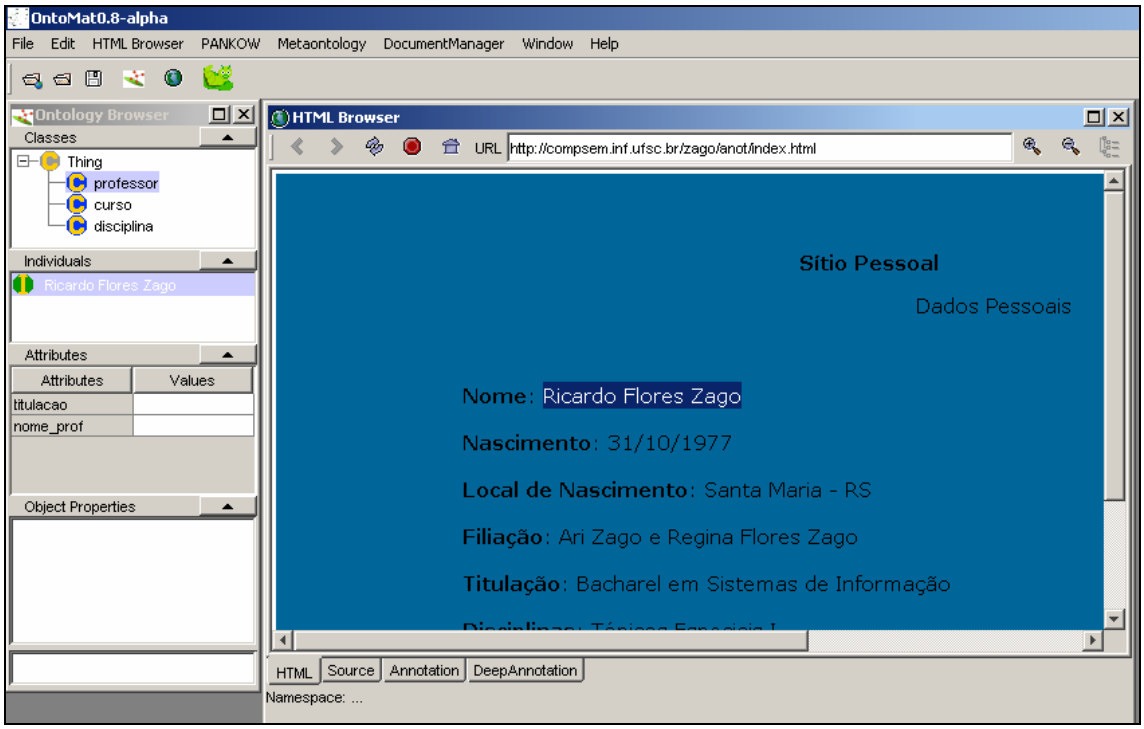


**Figura 9 - Área de trabalho do OntoMat-Annotizer**

Na parte esquerda da figura, observa-se o navegador de ontologias, onde pode-se visualizar as classes da ontologia que contém as classes para gerar a anotação. Na parte direita da figura está o navegador próprio da ferramenta. Para fazer a ligação entre as entidades presentes na página aberta no navegador e as classes da ontologia, basta selecionar a entidade e arrastá-la até a classe correspondente no navegador de ontologias.

### 2.5.3 OpenOffice

O OpenOffice é considerado, atualmente, a referência em termos de pacote de ferramentas para escritório (que envolvem editor de texto, planilha de cálculo e ferramenta para criação de apresentações, entre outras) utilizada sob a política de software livre.

O pacote tem sua origem quando do surgimento da StarDivision, empresa de software fundada na Alemanha em meados dos anos 80. A empresa lançou no mercado o pacote de ferramentas StarOffice. O sucesso da ferramenta fez com que a Sun Microsystems a adquirisse em 1999. Em junho de 2000, foi lançada a versão 5.2 da ferramenta, ainda utilizando a tecnologia da ferramenta desenvolvida pela StarDivision.

As novas versões do StarOffice, a partir da 6.0, foram construídas utilizando as fontes, APIs, referências de implementação e formatos de arquivos da OpenOffice.org. A empresa continua patrocinando o desenvolvimento do OpenOffice, utilizando no seu código-fonte as mesmas tecnologias que serão utilizadas nas versões seguintes do StarOffice (OPENOFFICE.ORG).

A ferramenta tem se tornado cada vez mais popular devido à sua semelhança com o produto da mesma linha – pacotes de ferramentas para escritório – mais utilizado e vendido em todo o mundo, o pacote Office da americana Microsoft.

O OpenOffice traz em sua interface uma quantidade considerável de recursos para uso de forma semelhante ao concorrente proprietário, fazendo com que usuários que sempre utilizaram a ferramenta da Microsoft tenham perdas não muito significativas de produtividade quando migram desta para o OpenOffice.

O OpenOffice utiliza XML como padrão de armazenamento dos arquivos. Como todo documento XML, um documento OpenOffice é representado por elementos e atributos. A estrutura de documentos aplica-se a todas as aplicações, o que faz com que o formato dos documentos de uma planilha eletrônica, editor de textos ou apresentação são os mesmos, variando apenas o conteúdo.

Cada arquivo OpenOffice é composto de vários sub-documentos. Cada sub-documento armazena as informações em formato XML, cada um contendo uma característica do arquivo. A tabela 1 descreve os sub-documentos que compõem um arquivo exemplo do OpenOffice.

**Tabela 1 - Sub-documentos que compõem arquivos do OpenOffice (OPENOFFICE.ORG XML FILE FORMAT 1.0, 2002)**

| **Arquivo OpenOffice: exemplo.sxw** | | |
|---|---|---|
| **Sub-documento** | **Elemento Raiz** | **Conteúdo** |
| Meta.xml | <office:document-meta> | Metadados sobre o documento, pode conter informações tipo autor, data de última alteração, descrição, etc. |
| Styles.xml | <office:document-styles> | Definição dos estilos de formatação utilizados no documento. |
| content.xml | <office:document-content> | Conteúdo do documento, contendo o texto digitado e formatações. |

| | | |
|---|---|---|
| settings.xml | <office:document-settings> | Configurações específicas, como tamanho da janela ou informações de impressão. |

Quando um arquivo OpenOffice é armazenado no computador, todos estes sub-documentos são empacotados e armazenados em uma única entrada do pacote completo. Para efetuar este empacotamento, o OpenOffice utiliza algoritmos de compactação em um formato bastante conhecido para esta tarefa, o formato ZIP.

Em todos os sub-documentos do OpenOffice são utilizados espaços de nome para identificar cada elemento do documento. Assim, para identificar elementos de texto nos documentos OpenOffice (elemento de texto pode ser um parágrafo, por exemplo), é utilizado o espaço de nome *text*, cuja URI é "http://openoffice.org/2000/text". Para identificar elementos de tabela em qualquer documento, é utilizado o espaço de nome *table*, cuja URI é "http://openoffice.org/2000/table".

Na próxima seção será apresentado o modelo para extração e recuperação de conhecimento corporativo anotado semanticamente, proposto neste trabalho.

# 3 Modelo para recuperação de conhecimento corporativo

## *3.1 Considerações iniciais*

No capítulo anterior, foram introduzidos conceitos acerca de Web Semântica e técnicas para seu funcionamento, como o uso de ontologias e agentes. Também foi levantada a necessidade das organizações de gerenciar seu conhecimento para recuperá-lo de forma rápida e eficiente.

O modelo proposto no presente trabalho visa desenvolver uma solução para que as empresas consigam gerenciar a parte do seu capital relativo ao conhecimento adquirido, utilizando técnicas de busca e recuperação de informação conforme os padrões de utilização de metadados empregados na área de Web Semântica.

A figura 10 apresenta a arquitetura modular proposta para tornar possível esta tarefa.

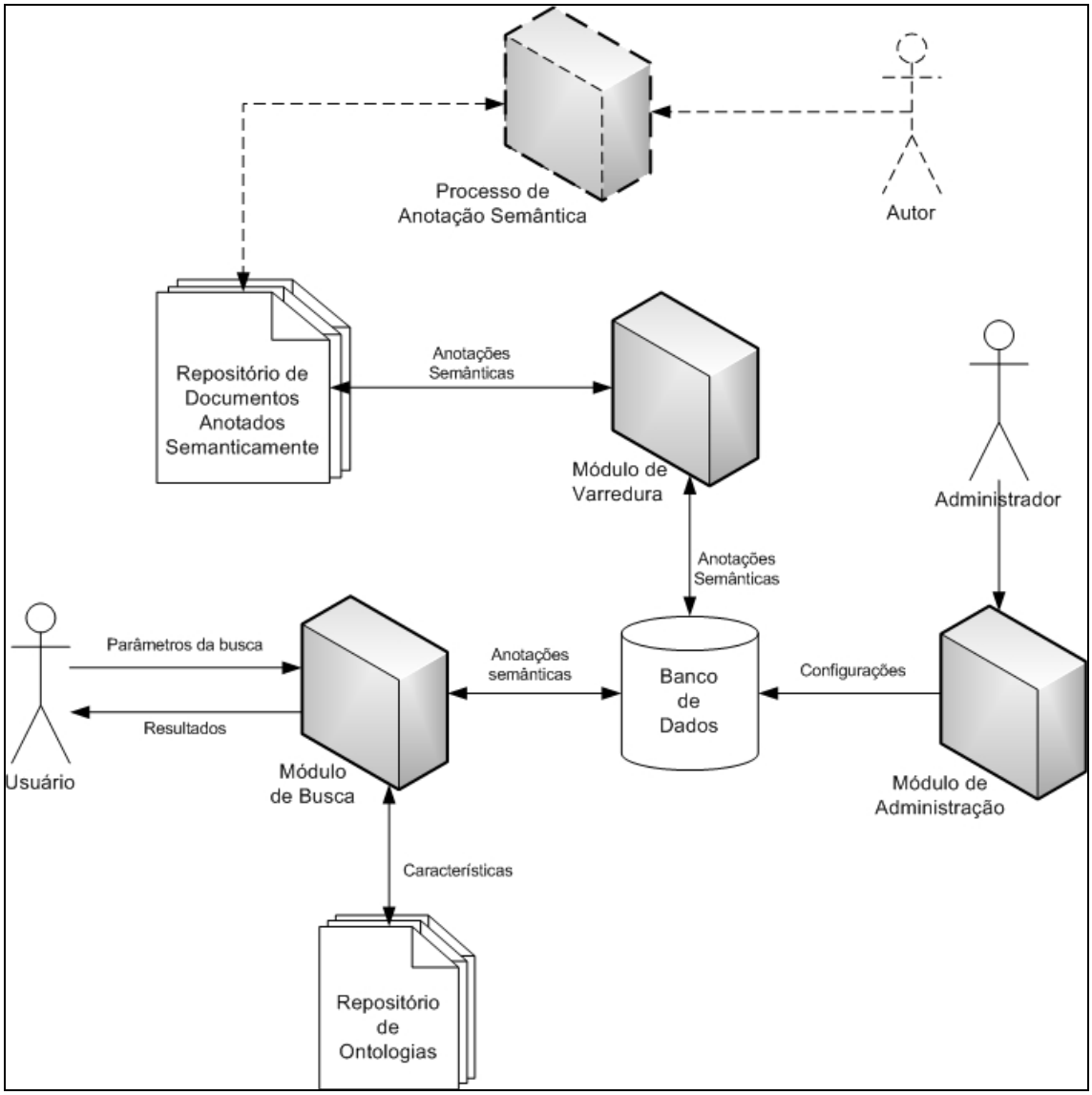


**Figura 10 - Arquitetura proposta para recuperação do conhecimento**

No diagrama, os processos que não fazem parte da proposta estão representados por linhas pontilhadas. Nestes processos, o autor do documento, utilizando a ferramenta para anotação, efetua as devidas ligações já durante a escrita do documento. Como exemplo, pode-se afirmar que o autor, ao digitar o nome de uma disciplina no seu documento, já pode vincular o nome escrito a uma classe de uma ontologia que defina as características de uma universidade, por exemplo.

Os demais módulos participantes do modelo, ilustrados no diagrama, são fundamentais para a recuperação de conhecimento corporativo utilizando o modelo apresentado neste trabalho e serão descritos na seqüência desta seção.

## *3.2 Repositórios de documentos e de ontologias*

Para que o modelo seja implantado, a empresa manterá dois repositórios, um para armazenar as ontologias criadas pelos administradores e outro para concentrar os documentos que contém o conhecimento de diversos setores da empresa.

### 3.2.1 Repositório de documentos

De modo geral, as empresas atualmente armazenam seus documentos de forma desordenada, mantidos em diferentes locais. Ao mesmo tempo em que destinam servidores de arquivos para que todos os funcionários armazenem seus documentos neste servidor, alguns usuários manipulam estes arquivos e os armazenam em suas próprias estações de trabalho, e ainda existem os que replicam os documentos nos dois locais citados acima.

Visando organizar esta gama de documentos mantida pelos funcionários da empresa, para que assim possa ser realizada uma varredura eficiente, o modelo apresentado propõe o conceito de repositório de documentos.

O repositório de documentos utilizado no modelo não pode ser interpretado como um único local onde a empresa tenha que armazenar os seus dados corporativos. Caso uma arquitetura deste tipo fosse utilizada, a forma com que a empresa armazena seu conhecimento deveria ser repensada e reestruturada para que atendesse à ferramenta de busca.

Para evitar esta reestruturação da empresa, o conceito de repositório de documentos foi expandido para que possa adaptar-se ao ambiente empresarial. Neste trabalho, portanto, ao invés de apenas um local onde a empresa concentra seus documentos, o repositório de arquivos será o conjunto formado pelos caminhos de rede da empresa armazenados no cadastro do repositório juntamente com todos os seus subdiretórios.

Isto faz com que sejam contemplados tanto os servidores de arquivos como os próprios locais onde os usuários mantêm seus documentos nas estações de trabalho.

O formato de documentos previsto pelo modelo são os documentos escritos utilizando o pacote de ferramentas para escritório **OpenOffice.** Com o pacote, podem

ser escritos textos, planilhas de cálculo e apresentações, por exemplo, que conterão o conhecimento da organização.

Todos os tipos de documento escritos com OpenOffice armazenam informações a seu respeito (metadados) num mesmo local. Exemplos destas informações são nome do autor, data de criação, entre outras.

Para o funcionamento do modelo, as anotações semânticas nos arquivos OpenOffice serão inseridas num local específico onde estes arquivos armazenam os seus metadados. Este local pode ser observado através do acesso ao menu “Arquivo → Propriedades” do documento OpenOffice e escolha da aba “Descrição”. O conteúdo semântico será inserido no campo “Comentários”.

A Figura 11 exemplifica uma anotação presente num documento OpenOffice que poderá ser utilizada de acordo com o modelo.

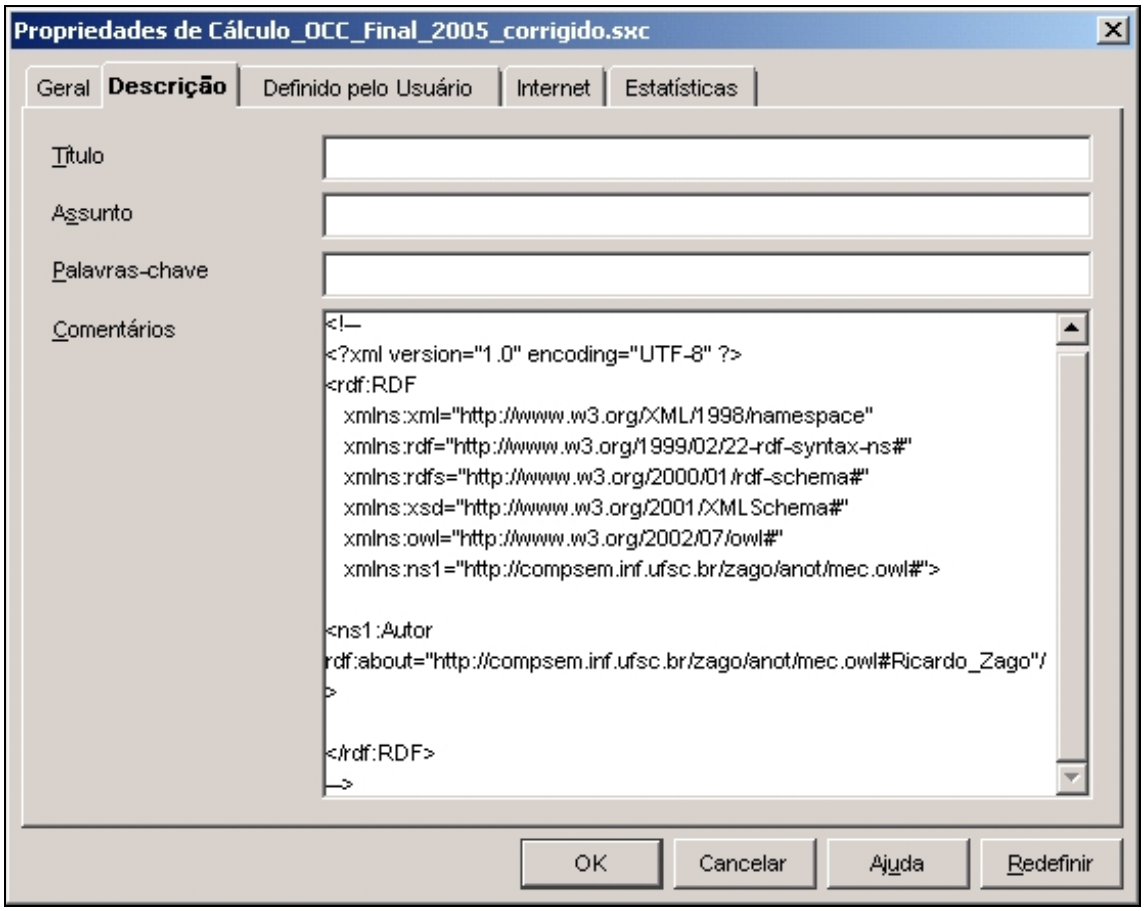


**Figura 11 - Exemplo de uma anotação inserida no documento OpenOffice**

A figura apresenta uma anotação semântica escrita e inserida no documento através da ferramenta de anotação (GLONVEZYNSKI, 2005). A anotação apresenta regiões que poder ser facilmente identificadas:

- **Definição dos espaços de nomes**: Os espaços de nomes contém as URIs utilizadas na anotação, podedo ser observados nas linhas que iniciam com a expressão “xmlns:”. A última destas linhas contém o espaço de nomes da ontologia utilizada para a anotação, identificado por “ns1”;
- **Sentenças**: Após a definição dos espaços de nome são descritas as sentenças anotadas com o uso da ontologia, através da marca “<rdf:About>”.

Na figura, observa-se a presença de apenas uma sentença, descrita em partes através da tabela 2:

**Tabela 2 - Sentença contida na anotação semântica do documento**

| Parte | Conteúdo | URI |
|---|---|---|
| Sujeito | Recurso “Ricardo_Zago” | http://www.inf.ufsc.br/zago/mec.owl#Ricardo_Zago |
| Predicado | Propriedade “type” | http://www.w3.org/1999/02/22-rdf-syntax-ns#type |
| Objeto | Recurso “Autor” | http://www.inf.ufsc.br/zago/mec.owl#Autor (representada pelo espaço de nomes “ns1”) |

### 3.2.2 Repositório de ontologias

Conforme descrito por DOAN (2003), atualmente podemos encontrar diversas ontologias previamente elaboradas por grupos de estudos de uma extensa gama de domínios.

Para que estas ontologias possam ser aproveitadas, o conceito de repositório de ontologias foi alterado para que estas ontologias já construídas sejam utilizadas nas consultas realizadas pelo modelo, de forma a aproveitar não só a conceituação definida pela empresa, mas também o conhecimento estabelecido por estes grupos de estudo. Desta forma, juntamente com o local designado para que a empresa armazene suas ontologias, o repositório de ontologias ainda possui um cadastro onde serão buscadas as ontologias externas, já criadas por outros grupos de estudo e disponibilizadas na *web*.

O repositório de ontologias possui características diferentes do repositório de documentos. Enquanto no segundo foi levado em conta que diferentes usuários da empresa armazenam seus documentos em locais de rede distintos, para a especificação do repositório de ontologias utilizou-se a definição de um único local onde a empresa deve armazenar as ontologias que construir para a conceituação dos diferentes elementos identificados em seu funcionamento.

No modelo proposto, foi determinado um padrão de linguagem para a elaboração das ontologias pelas empresas. A linguagem padrão escolhida foi a **OWL**, por ser a recomendação do World Wide Web Consortium (MCGUINNESS, 2004) e por ser a linguagem para manipulação de ontologias que oferece uma flexibilidade maior que as demais.

Para elaboração das ontologias, qualquer software desenvolvido para esta finalidade pode ser utilizado. Um exemplo deste tipo de software é o **Protégé**,

desenvolvido pelo *Stanford Medical Informatics* no *Stanford University School of Medicine*. Este software vem sendo bastante utilizado por diversos grupos de pesquisa em escala mundial. Por este motivo, grande parte das APIs de diferentes linguagens de programação são escritas e testadas utilizando o software.

A Figura 12 exemplifica a ontologia elaborada para testes através da ferramenta Protégé, utilizada para construir a anotação semântica mostrada na seção anterior. Na figura, estão ressaltadas a classe e a propriedade utilizadas na anotação.

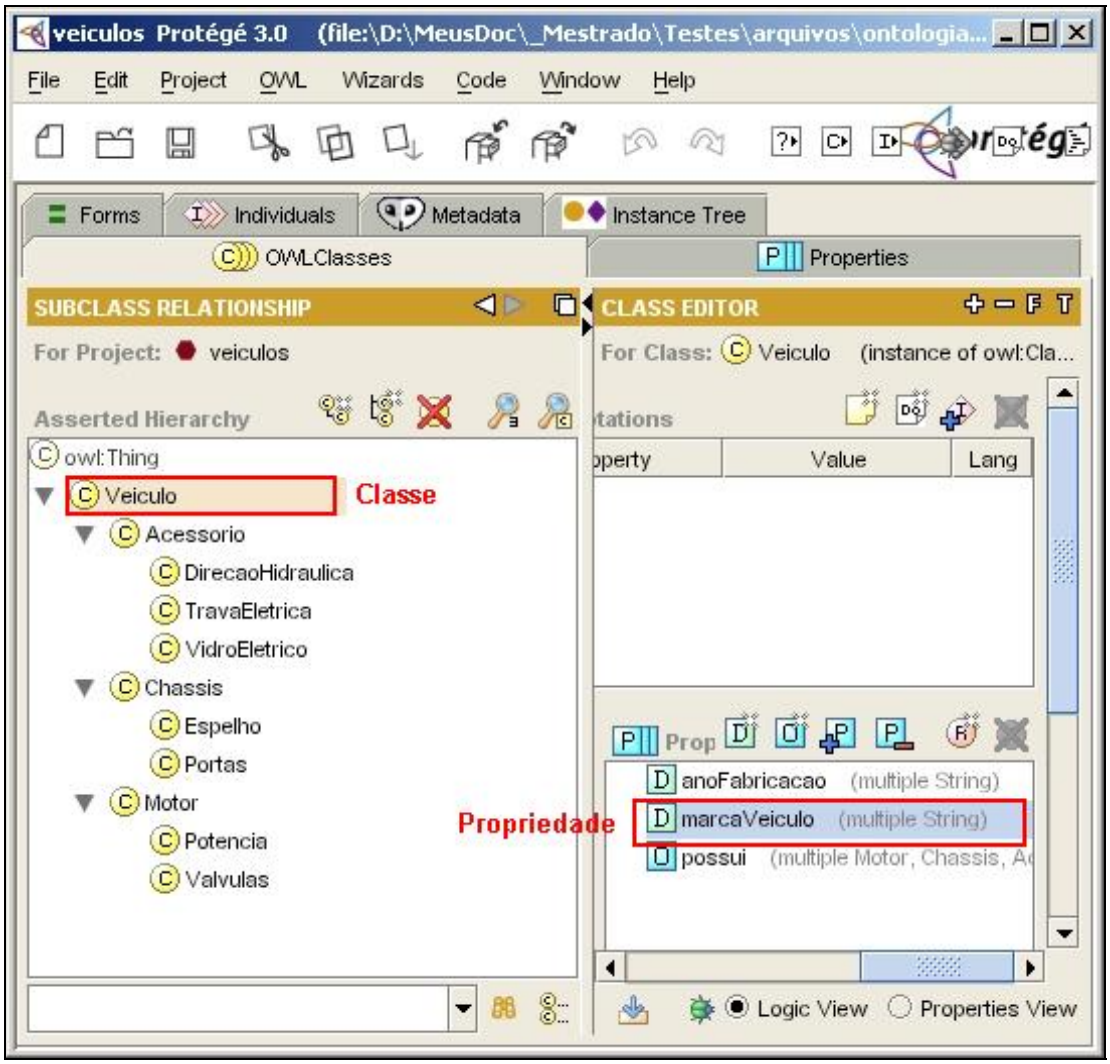


**Figura 12 - Exemplo de ontologia elaborada com a ferramenta Protégé**

## *3.3 Módulo de varredura*

O módulo de varredura é responsável pela busca de conteúdo semântico presente nos arquivos contidos no repositório de documentos e gravação destas anotações no banco de dados do modelo para que possam ser recuperadas pelo módulo de busca.

Este módulo percorrerá toda a árvore de subdiretórios dos caminhos presentes no repositório, procurando por anotações semânticas nos arquivos de tipo especificado.

Esta arquitetura é semelhante à utilizada pela ferramenta de buscas *Google* (GOOGLE.COM), uma ferramenta para recuperação *on-line* de documentos dispostos na Web, onde uma lista inicial com URLs cadastradas é pesquisada inicialmente e todas as ligações contidas nestas URLs iniciais indicam as próximas URLs a serem visitadas. No modelo apresentado, os caminhos iniciais de rede podem ser vistos sob o mesmo aspecto das URLs iniciais para o Google, e a estratégia do modelo para decidir qual o próximo local a ser visitado, que no Google é percorrer todos os links internos das páginas *web*, é percorrer toda a árvore de subdiretórios do repositório (BRIN, 1998).

A varredura ocorrerá em períodos configuráveis (todas as noites, por exemplo). Estes períodos são determinados pelos responsáveis pela tecnologia de informação na empresa. No modelo, não há um módulo específico para este propósito, visto que tarefas deste tipo são geralmente configuradas no agendador de tarefas do servidor.

## *3.4 Banco de dados do modelo*

O modelo proposto necessita da administração de um banco de dados para seu funcionamento. Este banco de dados será utilizado para armazenar dois tipos distintos de informação:

**Configurações do sistema**: Todas as configurações necessárias para o funcionamento do modelo são estabelecidas pelos administradores do modelo na empresa e armazenadas no banco de dados;

**Anotações dos arquivos**: As anotações encontradas pelo módulo de varredura são armazenadas no banco de dados, para utilização do módulo de busca.

Para armazenamento das anotações dos arquivos, são criadas entidades no banco de dados representando um relacionamento “um-para-muitos” (1..*) que armazenarão,

para cada arquivo pesquisado no módulo de varredura, as triplas sujeito-predicado-objeto contidas neste arquivo.

A Figura 13 demonstra este relacionamento entre as entidades do banco de dados responsáveis pelo armazenamento das anotações.


**Figura 13 - Diagrama de classes que representa as entidades envolvidas no armazenamento das anotações contidas nos arquivos do repositório**

Na figura, observam-se os atributos mínimos necessários para a identificação dos arquivos através do relacionamento entre as tabelas. Desta forma, cada arquivo localizado pelo módulo de varredura terá seu nome e seu caminho completo armazenado na entidade que receberá os arquivos, juntamente com um código identificador.

Este código identificador do arquivo fará o relacionamento 1..* com a entidade que receberá as anotações semânticas presentes nos arquivos. Além deste código, a entidade deve prever o armazenamento da tripla semântica, através dos atributos sujeito, predicado e objeto, que armazenarão o valor do recurso completo, ou seja, a URI presente na anotação semântica.

As informações relativas à configuração do sistema ficarão a cargo das necessidades gerais relativas ao funcionamento da ferramenta como um todo.

## *3.5 Módulo de busca*

O módulo que torna possível o objetivo principal do modelo, que é a recuperação do conhecimento corporativo, é o módulo de busca.

Através dele, o modelo receberá parâmetros de consulta informados pelo usuário, para acessar o banco de dados (que já deve conter as anotações previamente inseridas pelo módulo de varredura) e retornar os resultados de acordo com os parâmetros informados.

O primeiro destes parâmetros informados pelo usuário é a ontologia através da qual ele deseja obter informações de acordo com a classificação nela contida.

Após a escolha da ontologia, são previstas as seguintes alternativas para realização das consultas:

- Encontrar todos os arquivos que contém qualquer anotação para a ontologia escolhida;
- Encontrar todos os arquivos que contém anotação para uma determinada classe desta ontologia;
- Encontrar todos os arquivos que contém anotação para uma determinada instância de uma classe desta ontologia;

Além destas, outros tipos de consulta envolvendo raciocínios sobre a ontologia podem ser implementadas, mas constituem uma outra área de pesquisa e não fazem parte do presente trabalho. O modelo fornece as informações semânticas que viabilizam estes tipos de busca.

## *3.6 Módulo de administração*

Através do módulo de administração, os responsáveis pelo funcionamento do modelo na empresa efetuarão as tarefas de configuração do modelo. Estas configurações realizadas pelos funcionários da empresa serão utilizadas tanto pelo módulo de varredura dos arquivos quanto pelo módulo de consultas.

É importante salientar que o módulo de administração não faz parte do motor de busca, sendo considerado apenas um mecanismo através do qual os responsáveis da empresa por manter o funcionamento do modelo alteram parâmetros e estabelecem regras para o funcionamento da busca.

Os responsáveis pela administração do modelo na empresa informarão, através do módulo de administração, itens como repositórios de arquivos, ontologias externas, extensões de arquivos para busca e configurações gerais do sistema.

A aplicação do modelo, através de sua implementação e implantação num ambiente de trabalho corporativo, será apresentada na próxima seção, bem como os testes efetuados e resultados destes testes.

# 4 Implementação e aplicação do modelo de recuperação de conhecimento

Este capítulo tem por objetivo apresentar a implementação do modelo proposto e relatar os procedimentos necessários para a implantação do modelo para recuperação de conhecimento num ambiente corporativo.

O modelo implementado foi batizado de COBS Engine (Corporate Ontology-Based Search Engine). O nome e interfaces foram implementados em língua inglesa.

## *4.1 Implementação do modelo*

### 4.1.1 Ferramentas e tecnologias empregadas

Para efetuar a implementação do modelo, foi escolhida a linguagem **PHP** como interface de programação, por ser uma linguagem que possui recursos poderosos para programação para *web*, além de possuir uma vasta documentação e sua licença estar de acordo com as políticas de software livre.

Para que se possa fazer uso das técnicas da Web Semântica através das diferentes linguagens de programação, interfaces de programação (APIs) são escritas para que programadores possam implementar soluções utilizando sua linguagem preferida.

No presente trabalho, foram adotadas duas bibliotecas construídas para utilização da Web Semântica através da linguagem de programação PHP, envolvendo o uso de ontologias para classificação e anotação semântica de documentos e serão descritas a seguir.

#### 4.1.1.1 OWLLib

A OWLLib é uma biblioteca de funções que possibilita o acesso a ontologias escritas em linguagem OWL através da linguagem de programação PHP. O desenvolvimento da biblioteca é recente – o projeto foi registrado no *SourceForge* (SOURCEFORGE.NET), o maior repositório com abrangência mundial de projetos em software livre, no dia 14/03/2004 – e mantido pelo seu criador, Krzysztof Langner.

Apesar de o projeto não conter nenhuma documentação, o código-fonte é de fácil entendimento, e foi escolhida para efetuar o acesso às ontologias presentes no repositório.

#### 4.1.1.2 RDF API

O projeto RDF API trata de um conjunto de ferramentas para trabalhar com Web Semântica utilizando PHP como linguagem de programação. O projeto originou-se na Universidade de Berlin em 2002 sob a política de software livre, e ainda hoje recebe contribuições de colaboradores internos e externos ao programa.

A RDF API efetua a leitura de documentos em vários formatos (como OWL, RDF e N3) que podem ser representações de ontologias ou anotações semânticas em arquivos. A biblioteca manipula estes documentos de várias formas distintas, de acordo com o uso desejado pelo programador:

- **Modelo em memória**: Os documentos RDF são lidos e armazenados em memória para posterior utilização;
- **Modelo em banco de dados**: Depois de carregados, as partes dos documentos RDF são armazenadas em um banco de dados para que sejam utilizadas;
- **Modelos de inferência**: Os modelos de inferência são semelhantes aos modelos em memória, com o acréscimo de algoritmos encadeados para geração de regras de inferência para sentenças adicionais.
- **Modelo de recursos**: Este modelo representa cada grafo RDF como um conjunto de recursos que possuem propriedades.
- **Modelo de Ontologias**: Este modelo trata de arquivos RDF que contém ontologias, com métodos para manipulação de subclasses, instâncias, propriedades, entre outros.

A biblioteca RDF API foi escolhida para utilização na implementação do modelo para representação de conhecimento para efetuar a leitura de anotações semânticas nos documentos durante a varredura através do armazenamento em memória das triplas RDF retiradas das anotações. Outro fator que contribuiu para a escolha foi a vasta documentação disponibilizada pelos desenvolvedores.

### 4.1.2 Banco de dados do modelo

Um banco de dados para o modelo foi construído para armazenar informações que podem ser simples configurações da ferramenta ou até mesmo as próprias anotações utilizadas para efetuar a busca. Abaixo serão descritas as informações armazenadas no banco e sua utilização:

- **Cadastro de usuários**: O módulo de administração é a ferramenta pela qual os responsáveis pelo sistema de busca na empresa balizarão as atividades e o funcionamento do modelo. Estes responsáveis devem estar cadastrados e possuir senhas de acesso individuais para evitar o acesso não autorizado;
- **Extensões dos arquivos**: O modelo prevê o armazenamento das extensões dos arquivos dos repositórios que contém informação semântica. Desta forma, o módulo de varredura deve procurar as anotações nos formatos de arquivo informados pelos administradores. Isto faz com que o modelo não fique limitado aos tipos de arquivo do OpenOffice, pacote de aplicativos escolhido como principal objeto de uso pela empresa para manutenção de suas atividades diárias;
- **Caminhos iniciais do repositório de documentos**: O banco de dados armazenará os caminhos iniciais do repositório de documentos na empresa. O módulo de varredura, quando acionado, efetuará a varredura em profundidade de todos os caminhos inseridos no banco de dados. Isto faz com que o conceito do repositório de documentos seja formado pela união de diferentes caminhos dispostos na rede interna da empresa;
- **Ontologias externas**: As ontologias previamente definidas por outros grupos de pesquisa que possam ser utilizadas pela empresa devem ficar armazenadas no banco de dados do modelo;
- **Anotações dos arquivos**: As anotações semânticas serão retiradas dos arquivos e armazenadas no banco de dados através do módulo de varredura. Desta forma, a busca será efetuada diretamente no banco de dados, processo semelhante ao utilizado pela ferramenta *Google*.

O sistema gerenciador de banco de dados escolhido para implementação do modelo foi o **MySQL**, devido a sua licença estar de acordo com a política de software

livre e por ser considerado um gerenciador de banco de dados eficiente no que diz respeito à velocidade de recuperação dos dados, fator este fundamental para a ferramenta, visto que à medida que o número de documentos anotados cresce, um conjunto maior de sentenças é manipulado pelo banco de dados.

A Figura 14 ilustra o banco de dados implementado para o modelo.

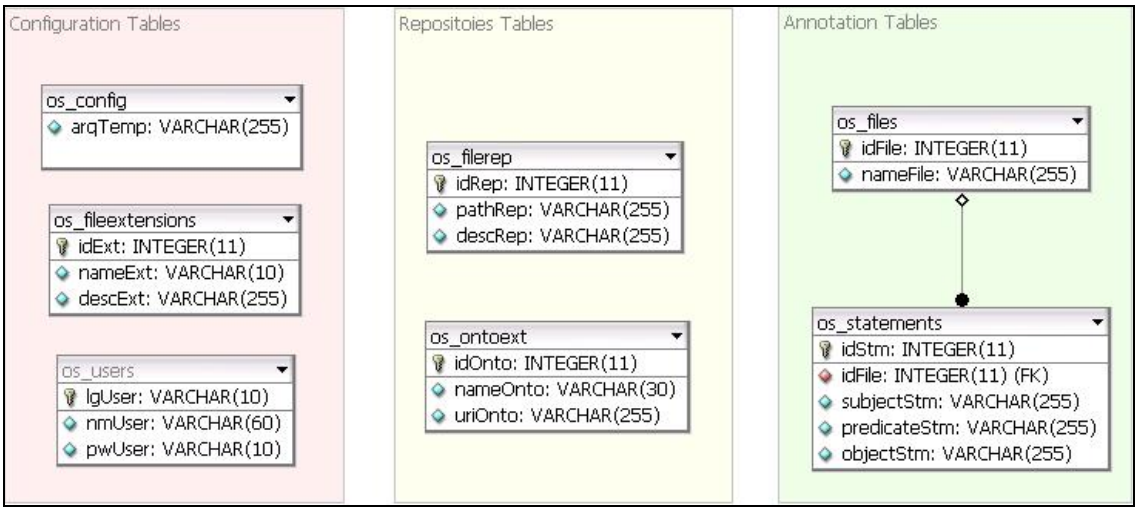


**Figura 14 - Banco de dados do modelo**

A biblioteca RDF API utilizada neste trabalho fornece um modelo para gravação das anotações em banco de dados. Este modelo mostrou-se ineficiente durante os testes com a implementação do modelo, não sendo utilizado por este motivo.

### 4.1.3 Módulo de varredura

Nesta seção, será descrita a implementação do módulo responsável pela varredura no repositório de documentos. Este processo pode ser identificado como um agente de software semântico, devido à sua finalidade de buscar todas as anotações semânticas encontradas nos arquivos do repositório de documentos.

O modelo de recuperação de conhecimento procura por anotações semânticas nos documentos OpenOffice nos locais estipulados para este fim (campo “Comentários” da aba “Descrição” no menu “Arquivo → Propriedades”).

Desta forma, o módulo de varredura efetua uma seqüência de passos desde a leitura do arquivo até a gravação no banco de dados, da seguinte forma:

- Descompactação do documento OpenOffice, mantendo os arquivos que o compõem em memória;
- Leitura do arquivo “meta.xml” que contém a anotação semântica em sua marcação “<dc:description>”;
- Separação do conteúdo da marcação “<dc:description>” do restante do arquivo “meta.xml”. Este conteúdo (anotação semântica) é gravado no arquivo RDF temporário;
- Remoção das triplas RDF do arquivo temporário (através da biblioteca RDF API) para serem gravadas no banco de dados.

Para cada anotação encontrada num arquivo, o módulo de varredura efetua a gravação no banco de dados através do relacionamento entre as tabelas “os_files” e “os_statements”, da seguinte maneira:

- Na tabela “os_files” é armazenado o nome do arquivo (com o caminho completo).
- Na tabela os_statements são armazenadas as anotações semânticas no formato de triplas RDF e a identificação do arquivo em que estavam contidas, para que se saiba qual o documento de origem.

### 4.1.4 Módulo de busca

O objeto principal do trabalho proposto é o dispositivo pelo qual os usuários finais poderão efetuar suas consultas a fim de recuperar o conhecimento necessário para suas atividades.

Semelhante a outras ferramentas de busca disponíveis na *web*, o processo de busca no modelo implementado efetua a pesquisa no banco de dados. As informações semânticas dos documentos já foram previamente coletadas dos documentos no repositório através do módulo de varredura.

No modelo proposto a consulta é realizada através de um número maior de passos do que as ferramentas de consulta atualmente disponíveis na *web*. Esta consulta foi desenvolvida de forma que o usuário não necessite de conhecimento mais aprofundado

sobre formatos como RDFS ou OWL, diferindo, desta forma, de outras interfaces semelhantes para realização de consultas utilizando Web Semântica. Um exemplo de ferramenta com esta arquitetura é o “Swoogle” (SWOOGLE.UMBC.EDU), uma ferramenta semelhante ao *Google* onde o usuário deve informar, nos parâmetros da sua consulta, itens como “type: rdfs” ou “encoding: n3” para balizar sua consulta.

Desta forma, para realizar uma consulta utilizando a implementação do modelo, é necessário que o usuário conheça quais ontologias estão disponíveis para uso (criadas pela empresa ou já existentes) e para qual finalidade são utilizadas pela empresa nos documentos.

#### 4.1.4.1 Fluxograma do sistema de busca

A Figura 15 demonstra a interação entre o usuário e o sistema de busca. Num primeiro momento, ao acessar o sistema, o usuário recebe uma lista com as ontologias disponíveis. Estas ontologias são buscadas pela ferramenta no repositório de ontologias (que contém as externas e as criadas pela empresa).

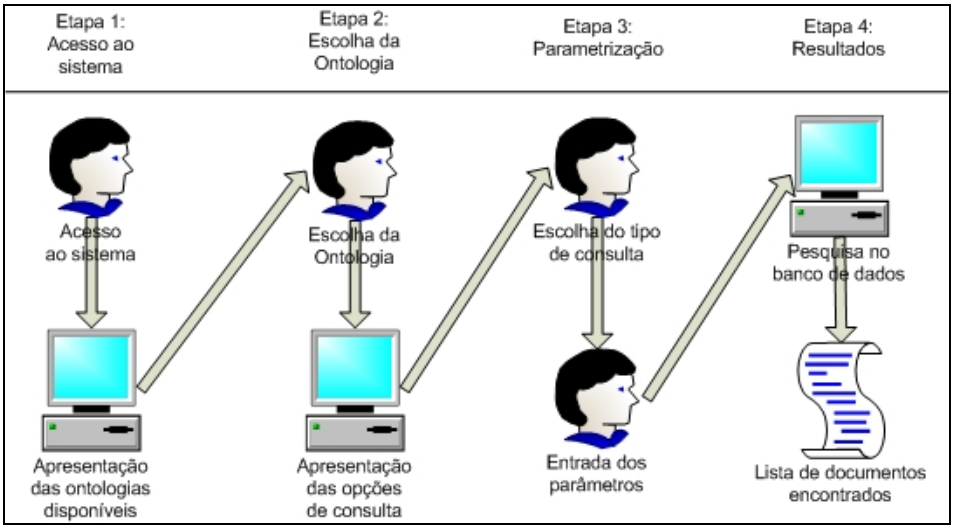


**Figura 15 - Fluxo de trabalho do COBS Engine**

A tela inicial do modelo implementado (Figura 16) traz apenas um dispositivo onde o usuário escolhe a ontologia que será utilizada na consulta atual.

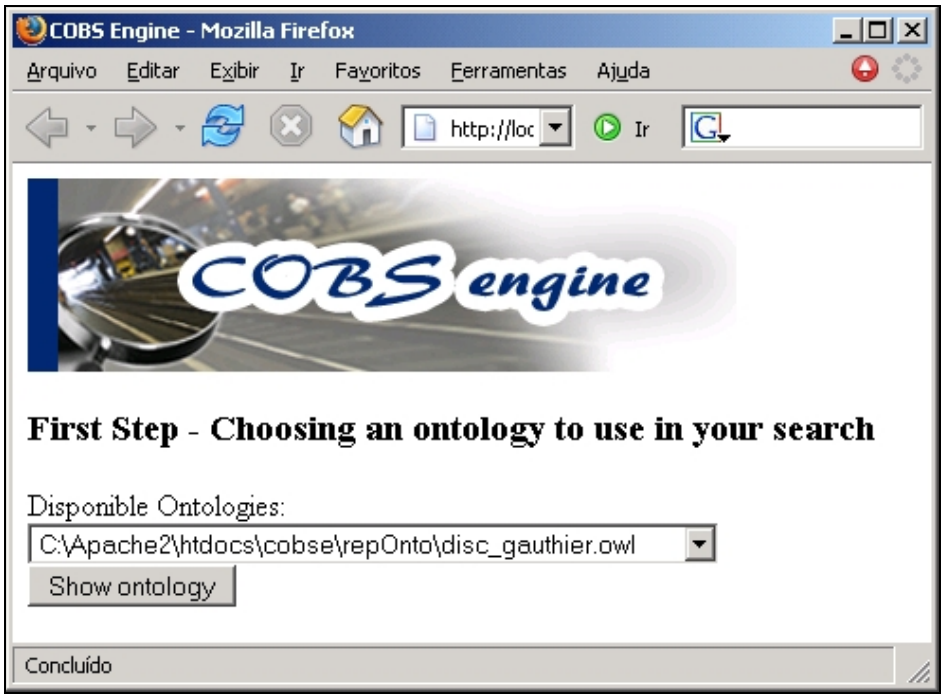


**Figura 16 - Tela de entrada do COBS Engine - Escolha da ontologia**

#### 4.1.4.2 Tipos de consulta

A segunda etapa do processo de busca semântica proposto neste trabalho envolve a escolha do tipo de consulta desejada. Para a realização desta etapa da consulta, o modelo para recuperação de conhecimento prevê quatro tipos possíveis de busca a ser efetuada nos documentos:

- **Todos os documentos com anotações da ontologia:** Quando escolhido este tipo de consulta, a ferramenta retorna todos os arquivos do repositório que contém qualquer anotação (classe, propriedade ou instância) da ontologia escolhida.
- **Documentos que apresentam anotações para uma classe específica:** Este tipo de consulta retorna os documentos anotados com uma determinada classe da ontologia. Para isto, o usuário deve indicar, além da escolha do tipo da consulta, qual a classe que deseja encontrar documentos anotados.
- **Documentos que apresentam anotações para instâncias de uma classe:** Quando escolhida esta consulta, serão retornados os documentos que contém anotações para instâncias de classes específicas.

A Figura 17 ilustra a interface através da qual os usuários escolhem o tipo de busca desejada.

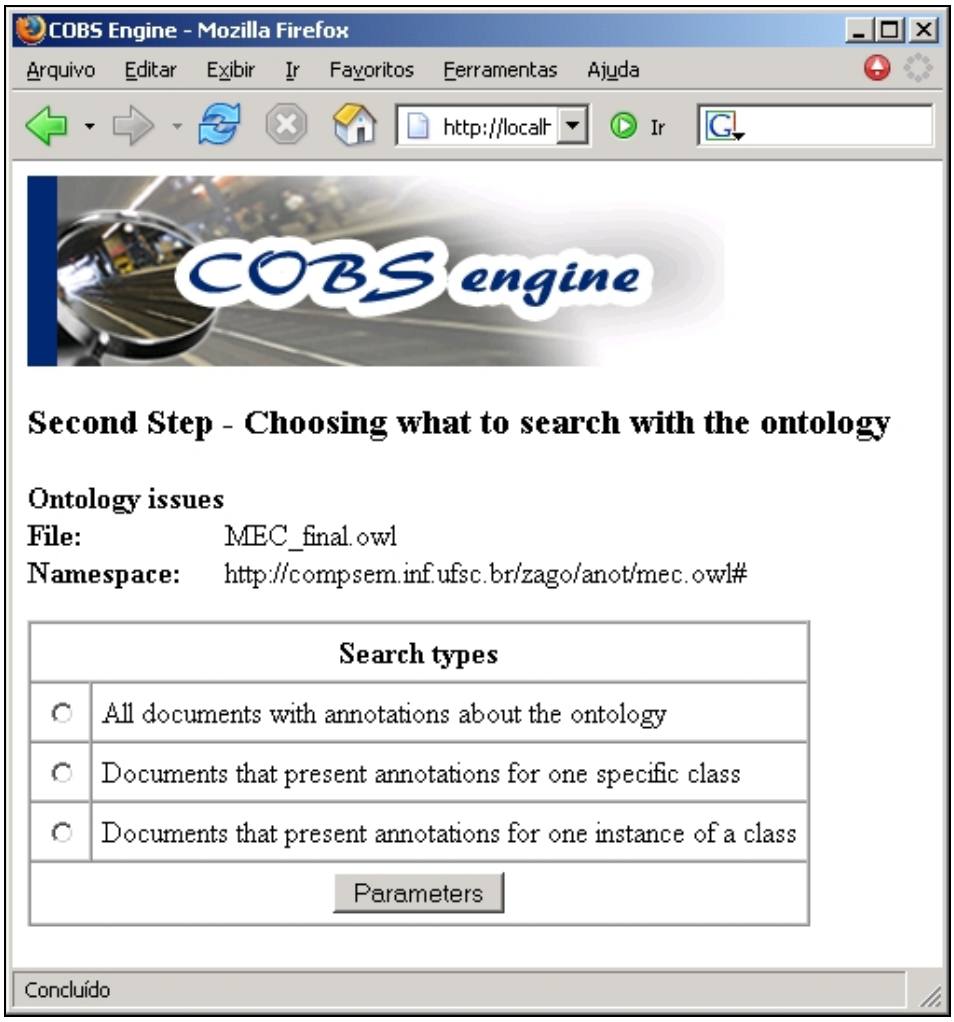


**Figura 17 - Escolha do tipo de consulta desejado pelo usuário**

#### 4.1.4.3 Parâmetros da busca

Na seqüência, depois de escolhidas a ontologia e o tipo da consulta, a ferramenta efetua a leitura de classes, propriedades e instâncias presentes na ontologia para montar, de acordo com o tipo da consulta informado pelo usuário, o terceiro passo da realização da busca. Nesta etapa, o usuário deve informar os parâmetros específicos do tipo de consulta escohido.

O único tipo de consulta que não necessita de parâmetros para consulta é o primeiro, o qual o usuário já escolheu a ontologia que ele deseja pesquisar e receberá como retorno todos os documentos que contém anotações para a ontologia escolhida.

No segundo tipo de consulta, onde são retornados todos os documentos que apresentam anotações para uma determinada classe, o módulo de busca lista para o usuário todas as classes presentes na ontologia, para que este escolha por qual deseja

pesquisar. A Figura 18 ilustra a interface responsável por fornecer estes parâmetros ao usuário.

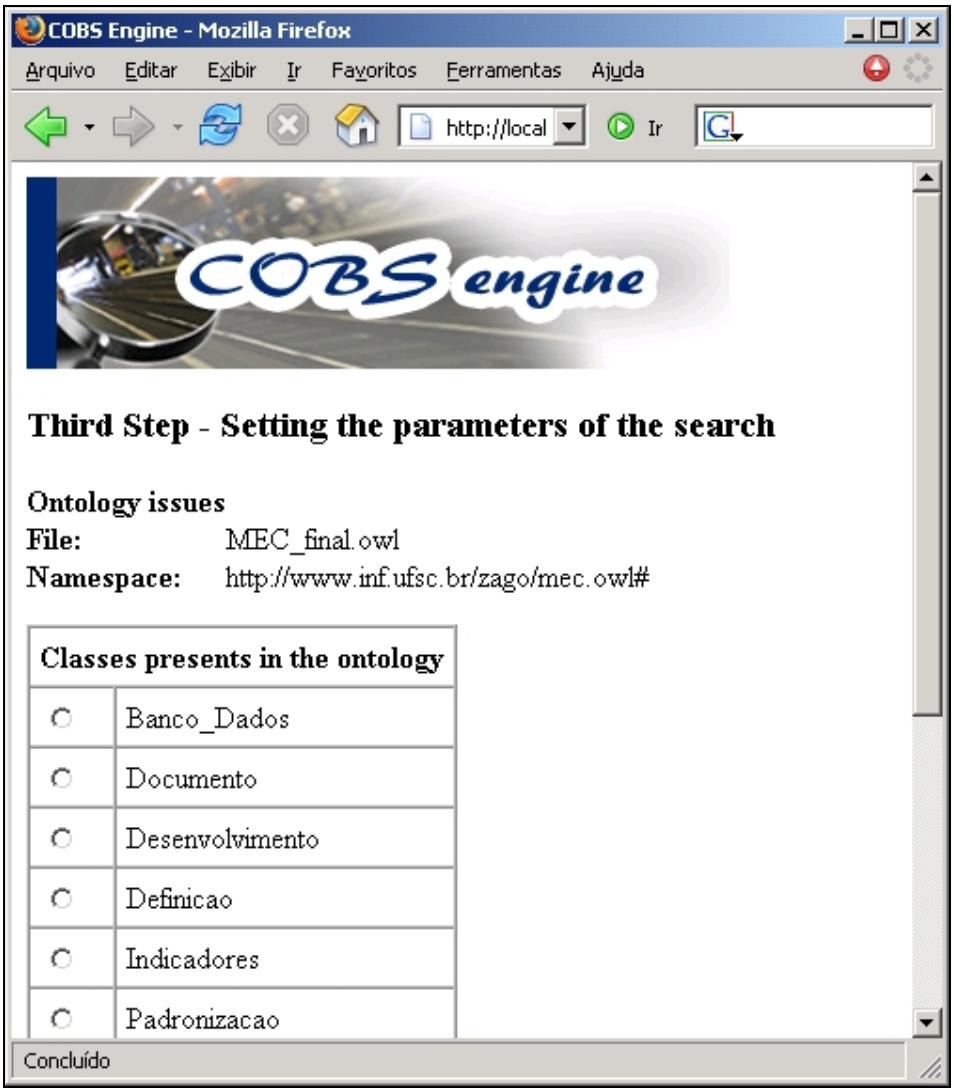


**Figura 18 - Parâmetros de busca por anotações de uma classe específica**

O último tipo de consultas assemelha-se ao anterior, exibindo para o usuário todas as instâncias presentes na ontologia separadas pelas suas classes. Este tipo de consulta aplica-se apenas para os casos em que as classes foram instanciadas diretamente na ontologia. A interface onde o usuário escolhe a instância a ser utilizada na consulta é mostrada na Figura 19.

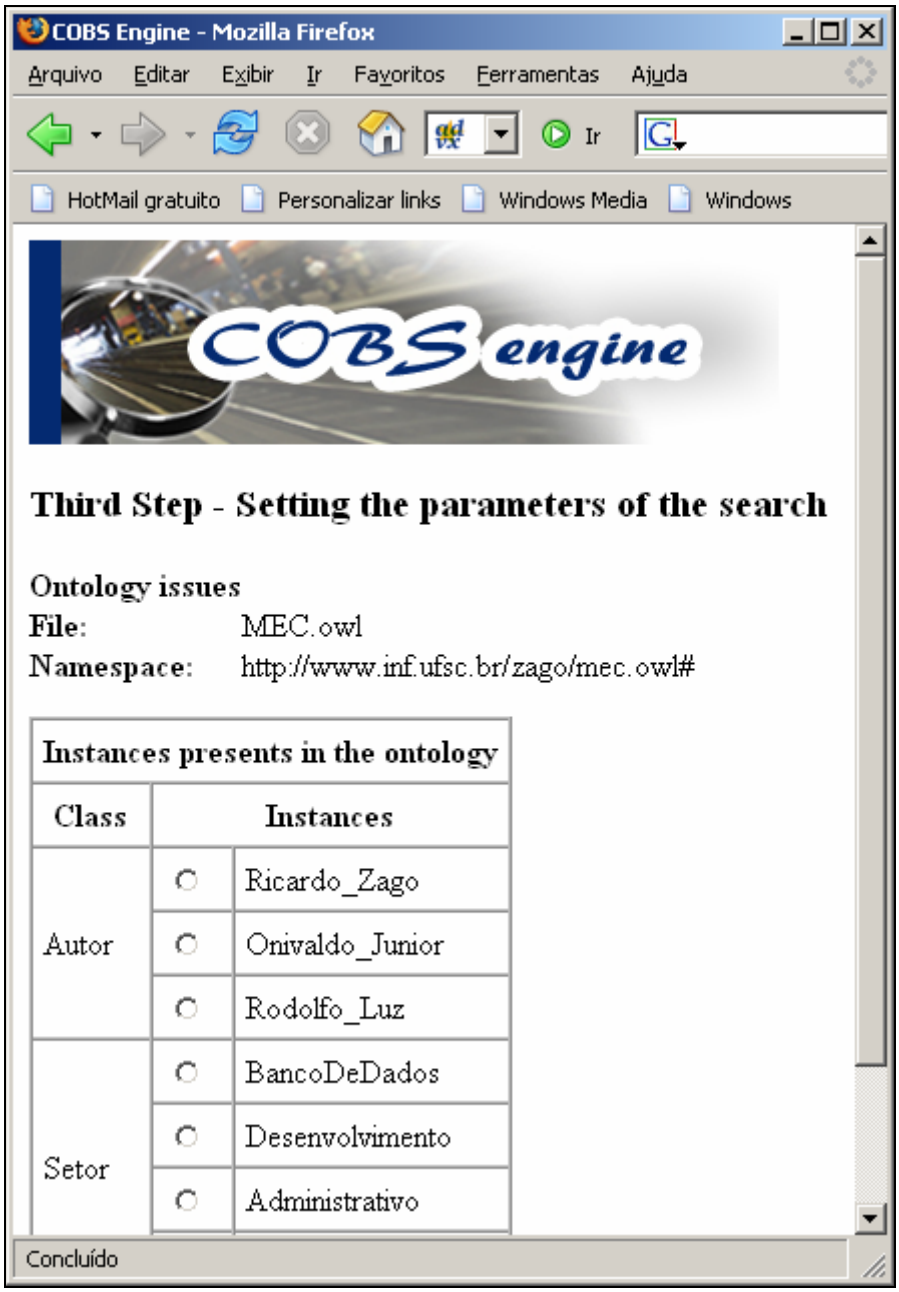


**Figura 19 - Parâmetros para o último tipo de consulta, instâncias da ontologia**

#### 4.1.4.4 Apresentação dos resultados

Com os parâmetros informados, o módulo de busca pode efetuar a última etapa da consulta. Através destes parâmetros o módulo realiza uma pesquisa no banco de dados à procura de documentos que apresentem classes, propriedades ou instâncias de acordo com a solicitação do usuário.

Os dados apresentados para o usuário possuem um único formato, independente do tipo de consulta realizado. Para cada anotação encontrada pela consulta, é retornado

ao usuário o nome do arquivo com o caminho completo, acompanhado da tripla presente na anotação identificada pelas partes que a compõem (classe, propriedade e valor). A Figura 20 ilustra a interface de apresentação dos resultados para o usuário.

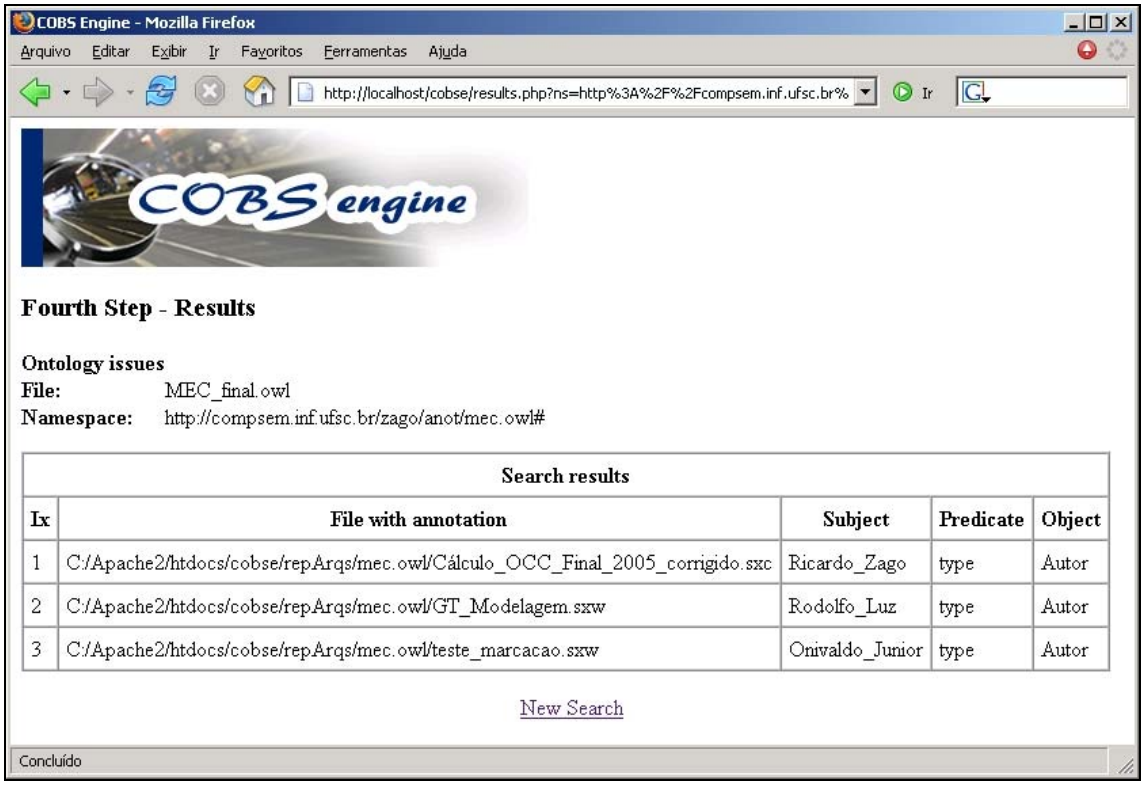


**Figura 20 - Apresentação dos resultados da busca efetuada**

### 4.1.5 Módulo de administração

Este módulo especial do modelo para recuperação de documentos corporativos foi desenvolvido com a finalidade de orientar as tarefas de configuração que irão ser utilizadas tanto pelo módulo de varredura quanto pelo módulo de busca.

É importante salientar que o módulo de administração não atua diretamente na recuperação do conhecimento, sendo considerado apenas um mecanismo através do qual os responsáveis da empresa por manter o funcionamento do modelo alteram parâmetros e estabelecem regras para o funcionamento da busca.

O usuário responsável pela administração do modelo na empresa deve informar, através do módulo de administração, itens como repositórios de arquivos, ontologias externas, extensões de arquivos para busca e configurações gerais do sistema.

O módulo administrativo foi construído sob forma de sistema *web*, ou seja, estará disponível para uso através de um servidor *web* e acessado através de um navegador. Desta forma, questões de segurança (por exemplo, quais computadores deverão acessar o módulo administrativo) deverão ser tratadas através da administração do servidor onde o modelo implementado será instalado.

#### 4.1.5.1 Configurações gerais do sistema

O módulo do sistema de administração denominado "Configurações gerais" tem a finalidade de armazenar algumas informações necessárias para o funcionamento da pesquisa. As duas principais configurações necessárias são:

- **Arquivo RDF temporário**: As anotações retiradas dos arquivos constantes no repositório de arquivos da empresa são armazenadas num único arquivo em forma de triplas RDF antes de serem transportadas ao banco de dados.
- **Localização das ontologias**: O administrador deve configurar o caminho de rede onde as ontologias desenvolvidas pela empresa devem ser armazenadas.

A Figura 21 ilustra a interface através da qual o administrador pode ajustar as opções de configurações gerais.

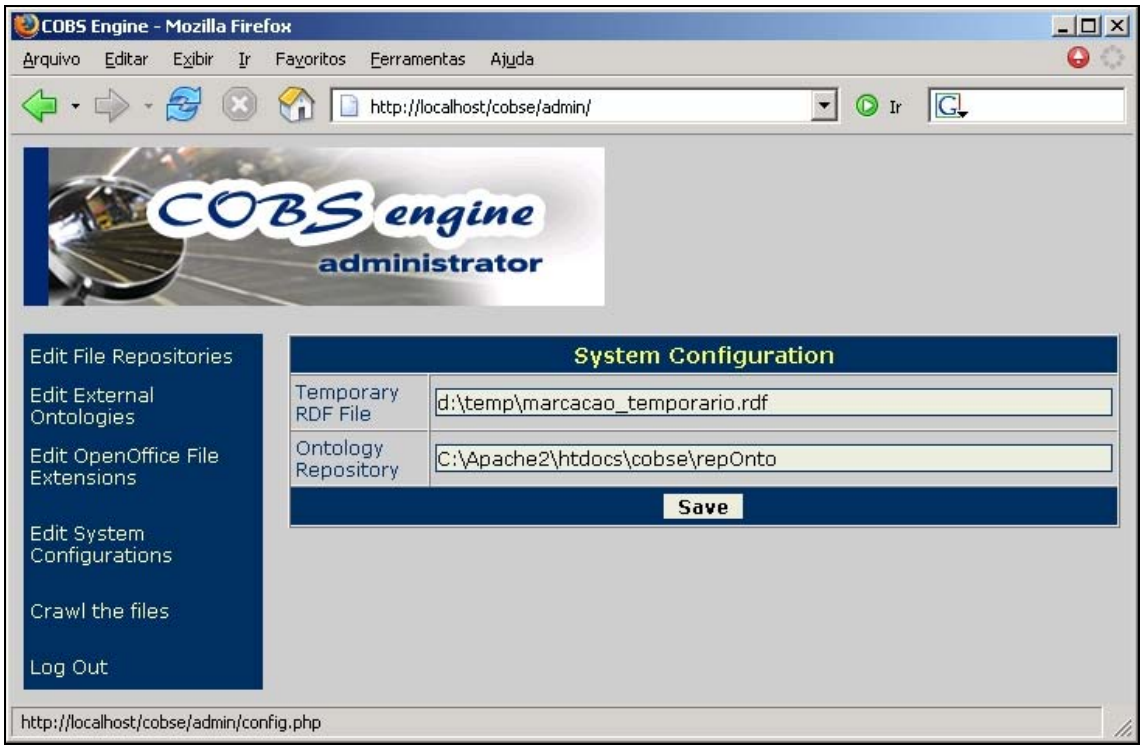


**Figura 21 - Configurações gerais do sistema**

#### 4.1.5.2 Cadastro do repositório de documentos

A Figura 22 ilustra o cadastro do repositório de documentos, local onde os administradores da busca deverão informar todos os locais onde possam estar disponíveis documentos contendo anotações semânticas.

As informações referentes aos locais, armazenadas no cadastro, são o caminho completo de rede (que será utilizado pelo módulo de varredura para efetuar as buscas) e uma descrição deste caminho, para controle do administrador.

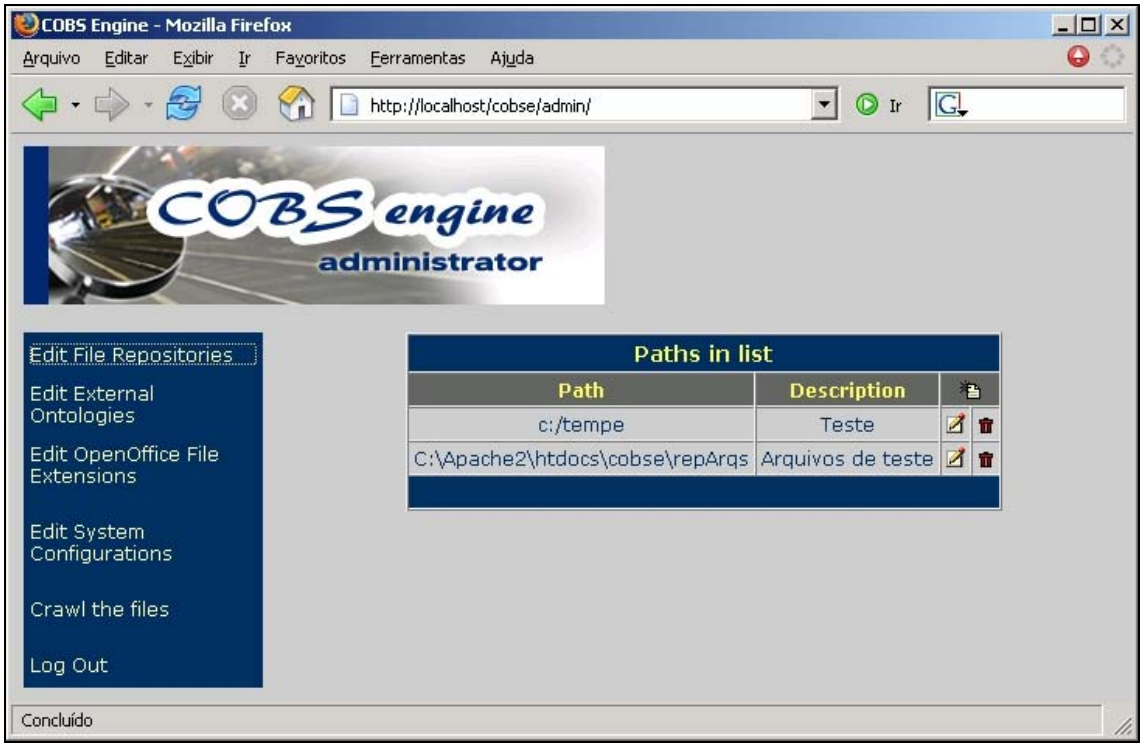


**Figura 22 - Cadastro do repositório de arquivos**

#### 4.1.5.3 Cadastro de ontologias externas

As ontologias externas devem ser armazenadas no módulo administrativo, para que o módulo de busca possa unir estas informações às ontologias presentes no repositório de ontologias da empresa (cadastrado nas configurações gerais) e formar uma lista completa de ontologias utilizadas pela empresa.

As informações referentes às ontologias externas são a URI (utilizada pela ferramenta de busca para localizar a ontologia) e o nome desta ontologia. Estas informações podem ser observadas na Figura 23.

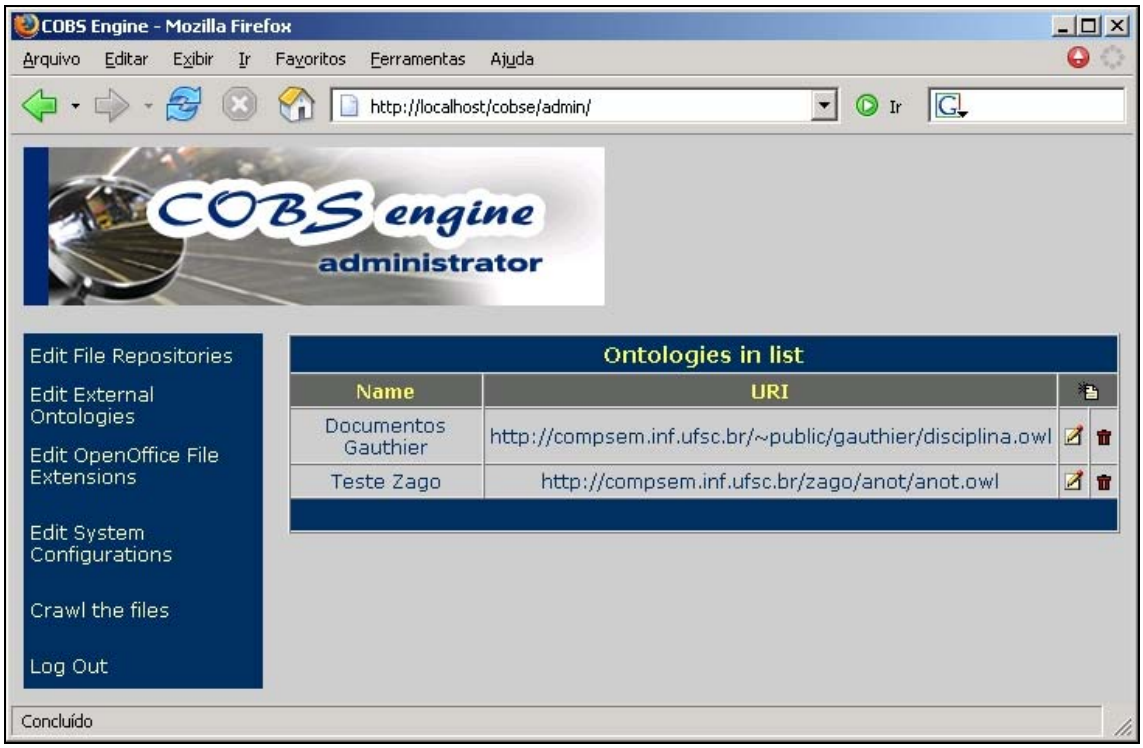


**Figura 23 - Cadastro de ontologias externas**

#### 4.1.5.4 Extensões de arquivos para busca

Apesar de ter sido designado o pacote de ferramentas para escritório OpenOffice como principal objeto de arquivos que contém anotação semântica, o módulo de administração prevê a configuração de quais arquivos a empresa utilizará para realizar a procura de anotações.

Esta configuração é realizada através do cadastro de extensões de arquivos (Figura 24) que serão analisados pelo módulo de varredura. As informações necessárias são a extensão do arquivo (no formato “.xxx”) e a descrição do tipo de arquivo.

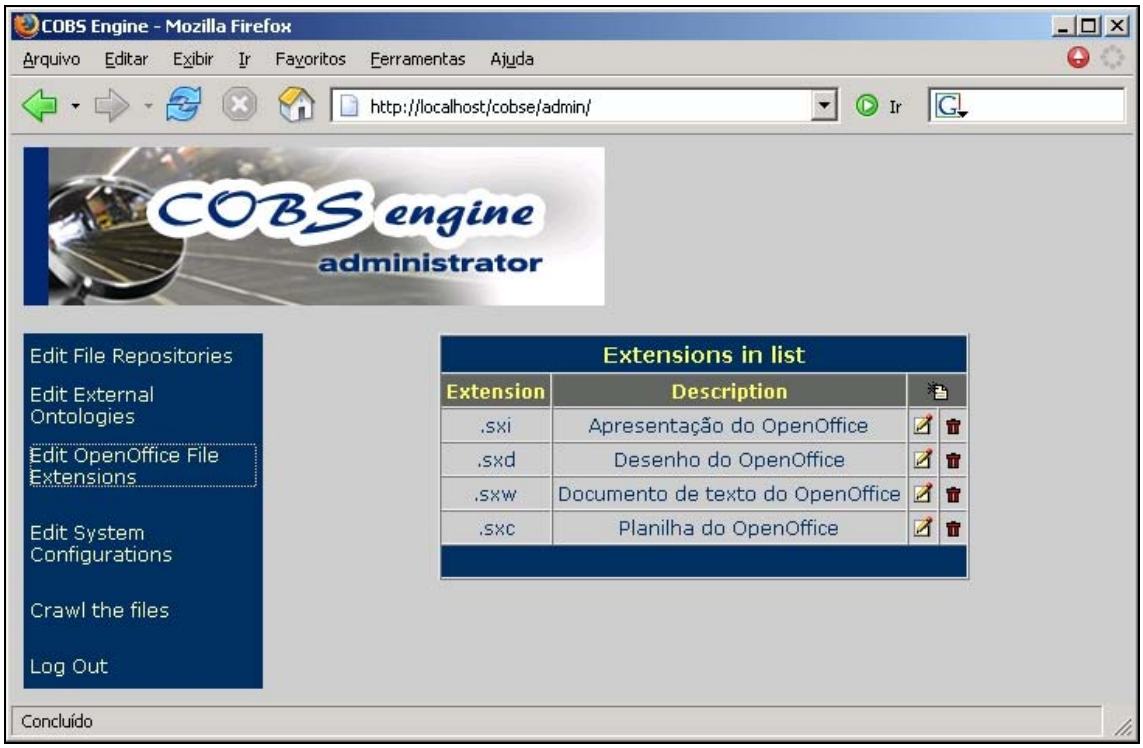


**Figura 24 - Cadastro de extensões de arquivos para busca**

## *4.2 Aplicação do modelo implementado num ambiente de trabalho*

O local escolhido para aplicação foi o Departamento de Tecnologia da Informação da Secretaria de Educação Superior (TI/SESu), no Ministério da Educação.

A escolha da TI/SESu como local de aplicação do modelo deve-se à necessidade dos integrantes da equipe de trabalho de compartilhar os documentos produzidos, sendo que um componente de um setor específico utiliza documentos gerados por outro setor. Assim, pode ser feita uma classificação ontológica no ambiente de trabalho e, quando definidos os repositórios de arquivos, o modelo pode ser implementado.

Outro fator que influenciou na escolha pela TI/SESu como ambiente para aplicação do modelo foi o fato deste setor já utilizar, em grande parte de seus documentos, a ferramenta OpenOffice, em atendimento ao Decreto de 29 de outubro de 2003 (MANTEGA, 2003), que institui a criação de comitês técnicos para implementação de *software* livre no governo.

A equipe TI/SESu é composta por uma equipe de dez pessoas (sendo um coordenador), que atuam em três setores diferentes:

- **Desenvolvimento**: Equipe composta por cinco pessoas, sendo um consultor que atua como supervisor da equipe e dois servidores públicos que, juntamente com dois estagiários, são responsáveis pela programação;
- **Banco de Dados**: Equipe composta por um consultor que atua como supervisor e um servidor público como administrador de banco de dados;
- **Indicadores**: Equipe composta por um servidor público e um consultor responsáveis por gerar novos indicadores para aferição do Ensino Superior no Brasil.

Todos os componentes da TI/SESu recebem documentos de outros órgãos, criam novos documentos e os compartilham de forma que todos os outros setores tenham como acessá-los para compartilhar o conhecimento. Este compartilhamento é necessário para que haja total integração entre os setores. Como exemplo, quando um indicador é criado, a equipe de banco de dados deve ler a documentação criada pelo setor de indicadores para que realize as devidas alterações no banco de dados e então gerar outra documentação para que os desenvolvedores criem rotinas para efetuar o cálculo do indicador.

### 4.2.1 Elaboração de uma ontologia para a aplicação

Para que qualquer ambiente corporativo implemente a solução para recuperação de conhecimento proposta neste trabalho, um passo essencial esperado é a construção de uma ontologia, que deve ser específica para o conhecimento armazenado na empresa.

Este passo não impede que sejam reutilizadas ontologias já existentes (a própria ferramenta prevê o uso destas), mas uma ontologia mais próxima ao ambiente de trabalho é importante no sentido que os conceitos por ela classificados são de conhecimento rotineiro dos usuários.

A ontologia específica para esta aplicação do modelo levou em conta a estrutura de arquivos já utilizada no ambiente de trabalho da TI/SESu, bem como as informações contidas nos arquivos e os tipos de documentos que podem ser criados por cada setor.

Desta forma, foram observadas algumas características relativas aos documentos que auxiliaram na construção da ontologia:

- Cada **documento** é elaborado sob responsabilidade de um **autor** de um determinado **setor** da equipe TI/SESu;
- Os documentos utilizados pelos setores podem ser classificados em três classes distintas de acordo com a sua criação: os que foram criados pelo setor de **desenvolvimento**, os relacionados ao setor de **banco de dados** e os relacionados ao setor de **indicadores**.
- Os documentos relacionados ao setor de desenvolvimento especializam-se em: a) documentos de **modelagem UML**, que são os documentos contendo as especificações dos sistemas construídos pela equipe; b) documentos de **padronização de código**, que são os documentos que definem os padrões a serem utilizados pela equipe para estabelecer padrões de codificação;
- Os documentos do setor de banco de dados são especializados em 2 classes: a) documentos de **modelagem de dados**, que são os documentos onde são descritas as tabelas geradas para cada banco de dado criado; b) documentos de **dados extraídos** das bases mantidas pelo setor de banco de dados;
- Os documentos relativos ao setor de indicadores podem ser especializados em: a) documentos de **definição de indicadores**, que contém a documentação necessária para o entendimento dos indicadores trabalhados pela TI/SESu, como exemplos, fórmulas e conceitos das variáveis utilizadas nas fórmulas; b) documentos de **aplicação de indicadores**, que utilizam os dados brutos extraídos pela equipe de banco de dados e aplicam as fórmulas dos indicadores para chegar ao cálculo final; c) documentos de **resultados**, onde são feitos cruzamentos de dados de diferentes indicadores para se chegar a algum resultado em nível estratégico.

De acordo com estas definições, foi criada uma estrutura de classes para a construção da ontologia, ilustrada na Figura 25:

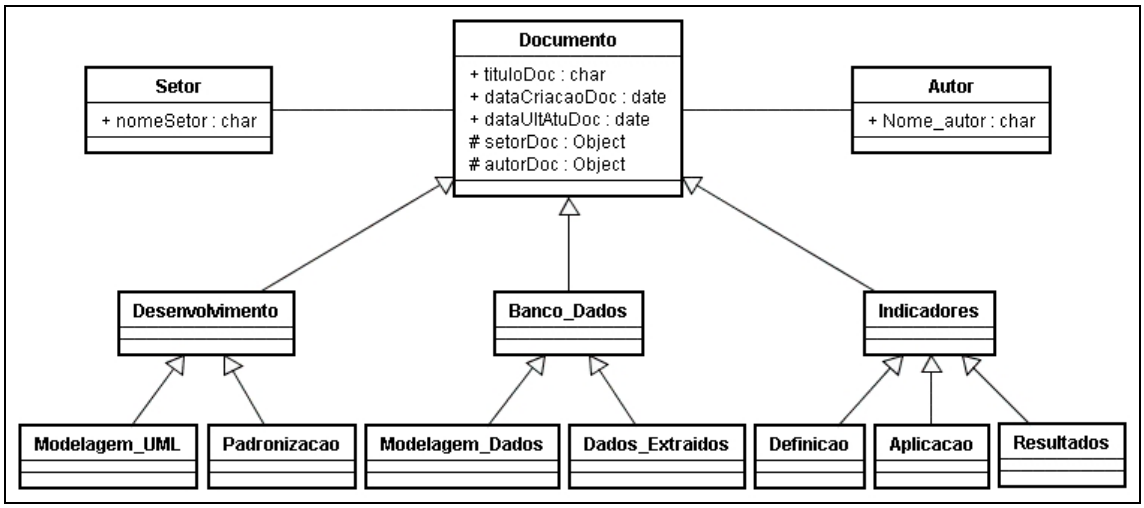


**Figura 25 - Especificação de classes para criação da ontologia**

Após a especificação das classes, a ontologia foi criada utilizando-se a ferramenta Protégé (PROTÉGÉ ONTOLOGY EDITOR). A Figura 26 mostra as classes na ontologia.

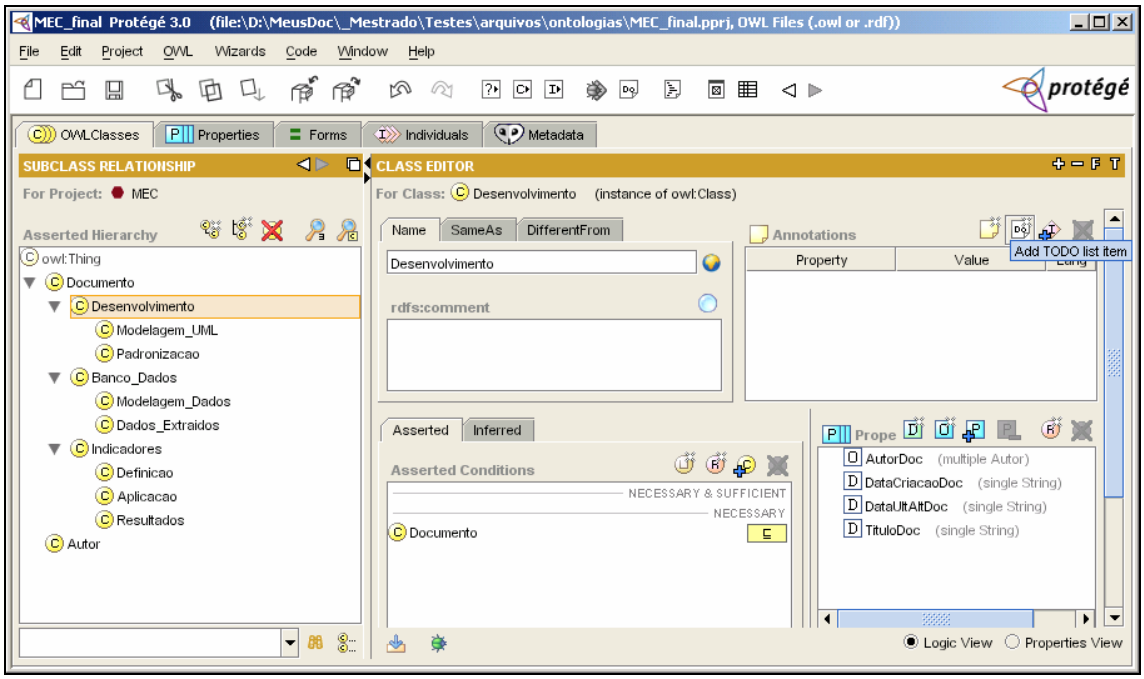


**Figura 26 - Ontologia utilizada nesta aplicação do modelo – apresentada na ferramenta Protégé**

A definição da ontologia permite a compreensão do conhecimento envolvido na organização para posterior utilização nos processos de anotação e recuperação nos documentos.

### 4.2.2 Testes efetuados

Durante um período de duas semanas, o modelo implementado foi testado no ambiente de trabalho da equipe TI/SESu, sendo que a primeira semana foi reservada para explicação do processo de anotação semântica com o uso da ferramenta de anotação (GLONVEZYNSKI, 2005) para a equipe e realização de anotações em arquivos. Somente depois de ter alguns arquivos anotados semanticamente, na segunda semana, foram efetuadas consultas através do modelo.

Neste período, foram efetuados testes no que diz respeito a:

- Viabilidade do processo de anotação semântica;
- Eficiência do módulo de varredura;
- Documentos recuperados pelo módulo de busca;

#### 4.2.2.1 Anotação de documentos

Para realização desta tarefa fundamental no processo de recuperação de conhecimento, foram utilizados os documentos existentes mantidos pela SESu/MEC bem como os novos documentos gerados pela equipe durante o período destinado a este fim.

A Figura 27 exemplifica um conteúdo semântico gerado pela ferramenta de anotação através da ontologia desenvolvida para esta aplicação do modelo. O conteúdo pode ser observado já inserido no documento OpenOffice mantido pela equipe TI/SESu.

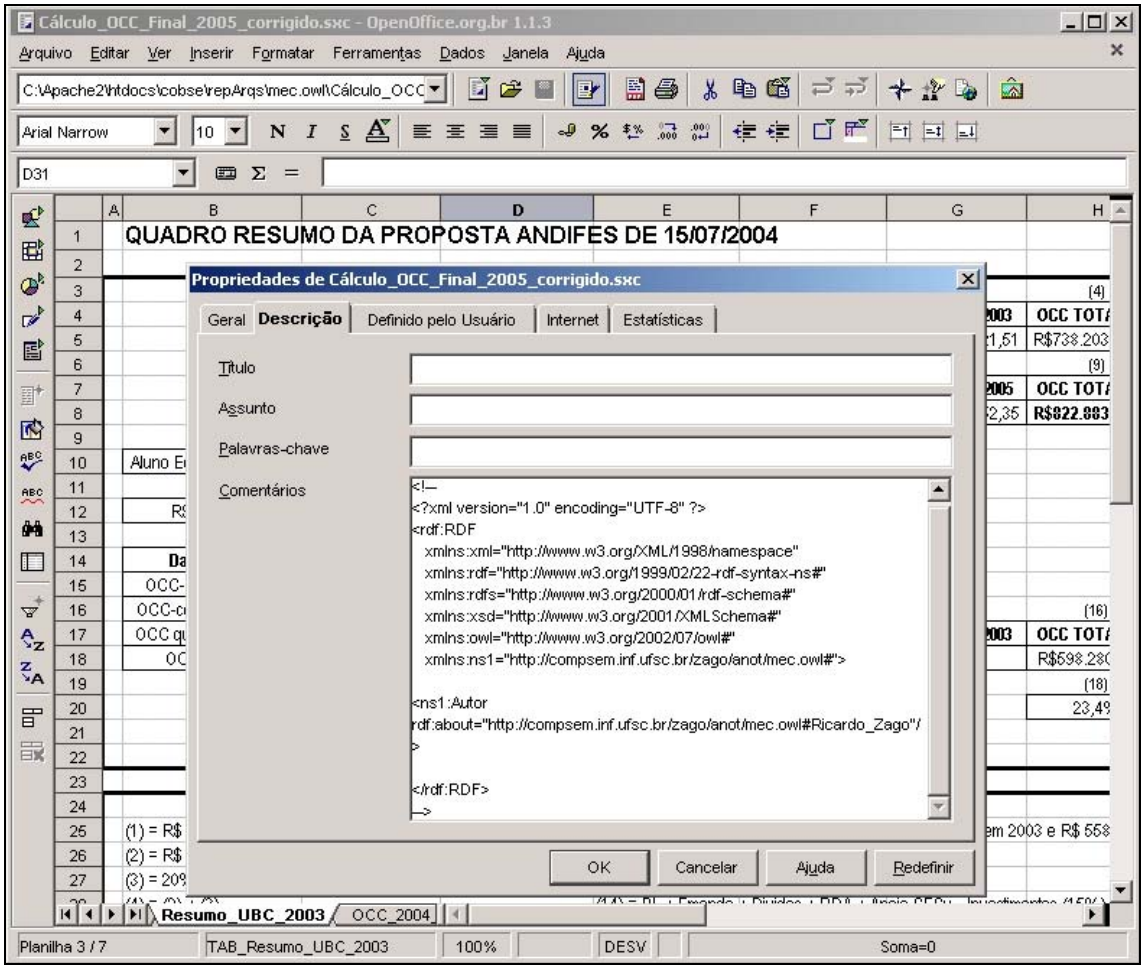


**Figura 27 - Documento do OpenOffice com anotação semântica**

#### 4.2.2.2 Módulo de varredura

Após as anotações terem sido inseridas nos documentos mantidos pela equipe, o módulo de varredura entrou em funcionamento.

Os testes realizados com o módulo de varredura envolveram:

- **Número de arquivos pesquisados**: O módulo de varredura deve percorrer, obrigatoriamente, todos os arquivos presentes no repositório de documentos à procura de anotações.
- **Número de anotações retornadas**: Durante a varredura, o módulo deve extrair de cada um dos arquivos que contém anotações semânticas todas as triplas RDF presentes na anotação;
- **Relação entre a anotação e o arquivo de origem**: Cada anotação inserida no banco de dados deve ter seu arquivo de origem relacionado. Esta regra de

negócio foi definida na geração do banco de dados, através de um relacionamento entre as tabelas envolvidas.

#### 4.2.2.3 Módulo de busca

O módulo responsável por receber os parâmetros informados pelo usuário e, através deles, procurar por anotações semânticas no banco de dados já alimentado pelo módulo de varredura foi submetido à validação através dos seguintes itens:

- **Apresentação das características da ontologia**: Todas as classes e instâncias presentes em cada ontologia selecionada devem ser apresentadas para que o usuário possa utilizá-las na consulta;
- **Documentos retornados**: O módulo de busca deverá encontrar documentos que contém anotações utilizando-se cada um dos três tipos de busca possíveis (todos os documentos que contém anotações, documentos que contém anotações para classes e para instâncias).

### 4.2.3 Resultados dos testes

Algumas considerações devem ser levadas em conta no que diz respeito à resposta obtida do modelo enquanto estava sendo testado no ambiente corporativo.

#### 4.2.3.1 Anotação semântica

Ao final da primeira semana de aplicação do modelo, a qual ficou reservada somente para os usuários efetuarem anotações semânticas nos documentos do ambiente de trabalho escolhido, foi constatado apenas um número de 74 documentos anotados pela equipe.

Este número foi considerado baixo, visto que no ambiente de trabalho existem aproximadamente 600 arquivos com conhecimento armazenado que deve ser recuperado pela ferramenta de busca. A razão identificada para este pequeno número de anotações foi o processo de anotação dos documentos, que por ser manual, teve um impacto negativo na equipe.

#### 4.2.3.2 Módulo de varredura

Durante os testes, verificou-se que sempre que os diretórios de rede estavam acessíveis, os arquivos foram percorridos e tiveram suas anotações inseridas no banco de dados. Outro aspecto observado foi que as anotações semânticas foram corretamente relacionadas aos arquivos de origem, podendo, desta forma, serem recuperados de forma eficiente.

Ao final da implantação do modelo, foi confirmada a eficiência do módulo de varredura, visto que todas as anotações presentes nos documentos foram corretamente inseridas no banco de dados, estando, assim, prontas para serem recuperadas pelo módulo de busca.

#### 4.2.3.3 Módulo de busca

Depois de os documentos terem sido anotados semanticamente e as anotações neles inseridas terem sido recuperadas de forma eficiente pelo módulo de varredura, o módulo de busca foi testado na segunda semana em que a equipe TI/SESu utilizou o modelo.

Durante os testes relativos à consulta por conhecimento mantido no ambiente de trabalho, foram testadas diferentes ontologias para observar a recuperação das características. Foram testadas, além de ontologias criadas durante a implementação, a ontologia desenvolvida para o ambiente de trabalho e ontologias desenvolvidas por outros grupos de pesquisa e armazenadas em servidores externos.

Após os testes, verificou-se que as características das ontologias foram recuperadas de forma satisfatória, sendo apresentadas todas as classes, propriedades e instâncias nelas presentes.

A recuperação de documentos através das anotações semânticas presentes no banco de dados do modelo foi verificada em todas as consultas realizadas pela equipe TI/SESu, que considerou os resultados obtidos satisfatórios devido a serem retornados os documentos de acordo com o conhecimento solicitado através das características da ontologia.

#### 4.2.3.4 Ferramentas utilizadas

Outro aspecto importante considerados no período de validação do modelo foram as ferramentas desenvolvidas para utilização com a Web Semântica utilizadas na implementação.

Para acessar as ontologias e recuperar suas características (classes, propriedades e instâncias), a OWL Lib teve resultados satisfatórios, porém algumas considerações devem ser levadas em conta:

- A biblioteca foi submetida apenas a ontologias criadas pela ferramenta Protégé (PROTÉGÉ ONTOLOGY EDITOR), onde teve atuação satisfatória. Outras ferramentas para geração de ontologias não foram utilizadas neste trabalho.
- A biblioteca apresentou algumas falhas no que diz respeito à recuperação de características da ontologia. A principal foi uma divergência entre a recuperação destas características em ambientes diferentes (Linux / Windows). Este transtorno foi contornado através da alteração do código-fonte da biblioteca.

A RDF API foi utilizada neste trabalho para leitura e extração de anotações semânticas dos documentos mantidos pela empresa. Na seqüência são apresentadas algumas considerações quanto a seu uso:

- A biblioteca apresenta uma inconsistência na criação do *alias* para os *NameSpaces* de diferentes ontologias, considerando um mesmo *alias* para todos os espaços de nome das ontologias presentes na anotação;
- Esta inconsistência, citada no item anterior, foi o motivo de o modelo de banco de dados da biblioteca não ser utilizado no modelo proposto no presente trabalho;

# 5 Conclusões e trabalhos futuros

## *5.1 Conclusões*

O valor do conhecimento adquirido pelas corporações cresce em importância na disputa empresarial cada vez mais acirrada, determinada por um mercado cada vez mais competitivo. Neste sentido, o vencedor da concorrência entre corporações pode ser aquele que acessar de forma mais rápida seu conhecimento e utilizá-lo de forma mais eficiente.

A inclusão de metadados semânticos nos documentos mantidos pela organização, de forma a fazer com que computadores possam compreender o conhecimento nele presente, tende a acelerar cada vez mais o processo de recuperação deste conhecimento.

A manutenção de uma ontologia relativa aos conceitos do ambiente de trabalho da empresa é fundamental para evitar ambigüidades entre os diferentes termos que fazem parte do seu cotidiano. O entendimento desta ontologia faz com que a anotação e recuperação do conhecimento ocorram de forma mais precisa.

A classificação ontológica dos termos envolvidos no ambiente de trabalho faz com que a ontologia criada se torne uma interface de comunicação entre o entendimento do usuário e o conhecimento armazenado nas anotações semânticas.

Através do estudo das técnicas de mineração de metadados semânticos, observou-se a necessidade da utilização de padrões pré-definidos para determinadas tarefas. No trabalho, levou-se em conta apenas ontologias escritas em OWL, pois entre as diversas linguagens existentes, é a recomendada pelo W3C.

O modelo apresentado, através de varreduras nos locais definidos pela empresa, alimenta um banco de dados com todos os metadados semânticos encontrados nos arquivos que contém o conhecimento da corporação. Esta alimentação é necessária para que o módulo de busca procure por anotações semânticas diretamente no banco de dados através do conhecimento definido pelas ontologias.

A utilização de um banco de dados para armazenamento das anotações semânticas acelera o processo de busca, uma vez que todos os arquivos já foram previamente

percorridos e suas anotações armazenadas. Desta forma, o tempo de resposta torna-se consideravelmente mais acelerado, pois verifica-se uma consulta simples de banco de dados utilizando a junção entre apenas duas tabelas.

A ferramenta que efetua a busca nos documentos anotados semanticamente foi construída na tentativa de abstrair do usuário a necessidade de conhecer técnicas de utilização da Web Semântica. Este objetivo foi alcançado, porém observou-se que o conhecimento da ontologia é fundamental para fazer com que a ferramenta compreenda a solicitação do usuário e retorne somente resultados relevantes.

O protótipo construído para avaliar a aplicação do modelo foi implementado e utilizado num ambiente em que os dados eram armazenados de forma desordenada. O conceito de repositório de documentos, além de fornecer ao módulo de varredura os locais iniciais para localização de documentos, trouxe uma maior organização no ambiente de trabalho no qual o protótipo foi implementado.

Desta forma, verifica-se a validade do modelo para recuperação de conteúdo anotado semanticamente, apresentado neste trabalho, uma vez que trouxe mais organização ao ambiente de trabalho a que foi submetido e fez com que o conhecimento fosse obtido de forma mais clara e objetiva. Outra contribuição que trouxe ao ambiente de trabalho a que foi submetido foi a melhor compreensão por parte dos usuários das atividades e conceitos do seu ambiente, através da ontologia.

## *5.2 Trabalhos futuros*

Algumas sugestões podem ser consideradas para realização de trabalhos futuros após o estudo realizado. Como a anotação de documentos não foi objeto direto deste estudo, mas é fundamental para que ocorra a tarefa de recuperação de conhecimento, pode-se sugerir um estudo para geração automática de anotações, pois a realização manual deste processo teve impacto negativo entre as pessoas que testaram o modelo.

Outro aspecto que ainda pode ser explorado é a expansão do modelo para documentos gerados por ferramentas diferentes do OpenOffice. Para que isto ocorra, deve-se estudar os tipos de documentos gerados por cada ferramenta a fim de descobrir onde inserir as anotações nestes arquivos e alterar o módulo de varredura para que este localize o conteúdo semântico em cada tipo de documento diferente.

Quanto à ferramenta de busca, pode-se sugerir alterações no sentido de aproximar ainda mais o usuário do conhecimento armazenado nas anotações semânticas, melhorando a forma como esta é apresentada na ferramenta e refinando as opções de busca possíveis com as características da ontologia.

# 6 Referências bibliográficas


ALAVI, Maryam; LEIDNER, Dorothy E. **Knowledge Management Systems: Issues, Chalenges and Beneficts**. Communications of the Association for Information Systems. 1999.

ARASU, Arvind; CHO, Junghoo; PAEPCKE, Andréas et al. **Searching The Web**. ACM Transactions on Internet Technology, Volume 1, Número 1, Páginas 2 a 43, Agosto, 2001.

BABILON, Maria. **The Evolution of Knowledge Management within NCR Corporation**. Proceedings of the 16th annual international conference on Computer documentation. Quebec, Canadá. 1998.

BRICKLEY, Dan; GUHA, R. V. **Resource Description Framework (RDF) Schema Specification 1.0**. W3C Candidate Recommendation 27 March 2000. Disponível em http://www.w3.org/TR/2000/CR-rdf-schema-20000327/. Acessado em 13/02/2005.

BRICKLEY, Dan; MCBRIDE, Brian. **RDF Vocabulary Description Language 1.0: RDF Schema**. W3C Recommendation 10 February 2004. Disponível em http://www.w3.org/TR/rdf-schema/. Acessado em 13/02/2005.

BRIN, Sergey; PAGE, Lawrence. **The anatomy of a large-scale hypertextual web search engine.** Computer Networks Journal, volume 30, páginas 107 a 117. Abril, 1998.

CHO, Junghoo; GARCIA-MOLINA, Hector. **Parallel Crawlers**. Proceedings of the eleventh international conference on World Wide Web. Hawaii, USA. Páginas 124-135. 2002

CONNOLLY, Dan; VAN HARMELEN, Frank; HORROCKS, Ian et al. **DAML+OIL (March 2001) Reference Description**. W3C Note 18 December 2001.

Disponível em http://www.w3.org/TR/daml+oil-reference/. Acessado em 13/02/2005.

DAML.ORG. **Sítio oficial da linguagem DAML.** Disponível em http://www.daml.org/. Acessado em 25/07/2005.

DAVIES, John; FENSEL, Dieter; HARMELEN, Frank van. **Towards the Semantic Web: Ontology-driven Knowledge Management**. England: John Wiley & Sons Ltd, 2003.

DEITEL, H. M; DEITEL, P. J; NIETO, T. R. et al. **XML: How to program**. Porto Alegre: Bookman, 2003.

DOAN, AnHai; MADHAVAN, Jayant; DHAMANKAR, Robin et al. **Learning to match ontologies on the SemanticWeb**. The VLDB Journal — The International Journal on Very Large Data Bases. Volume 12. Páginas 303 a 319. Novembro, 2003.

FERBER, Jacques. **Multi-agent systems: An introduction to distributed artificial inteligence**. Addison-wesley, United Kingdom, London, 1998.

FERRAZ, Nelson. **Vantagens Estratégicas do Software Livre em Ambientes Corporativos**. Monografia apresentada para conclusão de curso Master Business Information Systems. PUC, SP, 2002.

GLONVEZYNSKI, Régis A. **Modelo de anotação de documentos para a codificação do conteúdo semântico no processo de autoria.** Dissertação de Mestrado apresentada para conclusão do Programa de Pós-Graduação em Ciência da Computação. UFSC, SC, 2005.

GOBLE, Carole; DE ROURE, David. **Semantic Web and Grid Computing**. Disponível em http://www.semanticgrid.org/documents/swgc/swgc-final.pdf. Acessado em 25/07/2005. Setembro, 2002.

GOOGLE.COM. **Sítio oficial.** Disponível em http://www.google.com.br/. Acessado em 25/07/2005.

GRAU, Bernardo C. **A Possible Simplification of the Semantic Web Architecture**. Proceedings of the 13th international conference on World Wide Web. Páginas 704-713. 2004.

GRUBER, Thomas R. **A translation approach to portable ontology specifications**. Knowledge Acquisition, 5:199–220, 1993.

GUHA, R.; MCCOOL, Rob; MILLER, Eric. **Semantic Search**. Proceedings of the twelfth international conference on World Wide Web. Budapest, Hungary. Páginas 20-24. 2003.

KLYNE, Graham; CARROLL, Jeremy; MCBRIDE, Brian. **Resource Description Framework (RDF): Concepts and Abstract Syntax**. W3C Recommendation 10 February 2004. Disponível em http://www.w3.org/TR/rdf-concepts/. Acessado em 13/02/2005.

KOGUT, Paul; HOLMES, William. **AeroDAML: Applying Information Extraction to Generate DAML Annotations from Web Pages**. First International Conference on Knowledge Capture (K-CAP 2001), Workshop on Knowledge Markup and Semantic Annotation, Victoria, B.C. 2001.

KIRYAKOV, Atanas. POPOV, Borislav. OGNYANOFF, Damyan. **Semantic Annotation, Indexing, and Retrieval**. 2nd International Semantic Web Conference (ISWC2003), Florida, USA. Páginas 484 a 499, Outubro, 2003.

LEE, Tim-Berners. **Universal Resource Identifiers in WWW**. Internet Engineering Task Force Request for Comments RFC1630, Disponível em http://www.ietf.org/rfc/rfc1680.txt?number=1680, acessado em 13/02/2005. 1998a.

LEE, Tim-Berners. **Uniform Resource Indentifiers (URI): Generic Syntax.** Internet Engineering Task Force Request for Comments RFC2396, Disponível em http://www.ietf.org/rfc/rfc1680.txt?number=2396, acessado em 13/02/2005. 1998b.

LEE, Tim-Berners; **Semantic Web.** *Slides* apresentados no congresso XML 2000, Washington, DC. Disponível em http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html. Acessado em 25/07/2005. Junho, 2000.

LEE, Tim-Berners; HENDLER, J.; LASSILA, O. **The Semantic Web.** Revista Scientific American, maio, 2001.

MANTEGA, Guido. **Decreto de 29 de Outubro de 2003: Institui Comitês Técnicos do Comitê Executivo do Governo Eletrônico e dá outras providências.** Disponível em https://www.planalto.gov.br/ccivil_03/DNN/2003/Dnn10007.htm Acessado em 25/07/2005. Outubro, 2003.

MCGUINNESS, Debora L.; FIKES, Richard; HENDLER, James et al. **DAML+OIL: an ontology language for the Semantic Web**. IEEE – Inteligent Systems. Volume 17. Páginas 72-80. 2002.

MCGUINNESS, Deborah L.; VAN HARMELEN, Frank. **OWL Web Ontology Language Overview**. W3C Recommendation 10 February 2004. Disponível em http://www.w3.org/TR/owl-features/. Acessado em 13/02/2005.

MENCZER, Filippo; PANT, Gautam; SRINIVASAN, Padmini et al. **Evaluating topic-driven web crawlers**. Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval. Páginas 241-249. 2001

MOURA, Ana Maria de C. **A Web Semântica: Fundamentos e Tecnologias**. Congreso Internacional de Ciencias de la Computación – CICC 2001. Universidad de Aquino. Aquino, Bolivia. 2001.

MYLLYMAKI, Jussi. **Effective Web Data Extraction with Standard XML Technologies**. Proceedings of the tenth international conference on World Wide Web. Hong Kong. Páginas 689-686. 2001.

O'LEARY, Daniel E. **Enterprise Knowledge Management**. IEEE Computer Society. Volume 31, número 3, páginas 54-61. 1998.

OFFICE.MICROSOFT.COM. **Sítio oficial.** Microsoft Corporation. Disponível em http://office.microsoft.com/. Acessado em 25/07/2005.

ONTOKNOWLEDGE.ORG/OIL. **Sítio oficial da linguagem OIL.** Disponível em http://www.ontoknowledge.org/oil/. Acessado em 25/07/2005.

OPENOFFICE.ORG. **Sítio oficial**. Sun Microsystems. Disponível em http://www.openoffice.org/. Acessado em 25/07/2005.

OPENOFFICE.ORG XML FILE FORMAT 1.0. **Technical Reference Manual**. Version 2. Sun Microsystems, dezembro 2002.

PANT, Gautam; Tsioutsiouliklis, Kostas; Johnson, Judy et al. Panorama: **Extending Digital Libraries with Topical Crawlers**. Proceedings ACM/IEEE Joint Conference on Digital Libraries, 2004.

PATEL-SCNHEIDER, Peter; SIMÉON, Jérôme. **The Yin/Yang web: XML syntax and RDF semantics**. Proceedings of the eleventh international conference on World Wide Web. Páginas 443-453. Hawaii, USA. 2002.

PROTÉGÉ ONTOLOGY EDITOR. **Sítio oficial**. Stanford University School of Medicine. Disponível em http://protege.stanford.edu/. Acessado em 20/07/2005.

QIN, Jialun; ZHOU, Yilu; CHAU, Michael. **Building domain-specific web collections for scientific digital libraries: a meta-search enhanced focused crawling method**. Proceedings of the 2004 joint ACM/IEEE conference on Digital libraries. Páginas 135-141. 2004

RUSSEL, Stuart. J., NORVIG, Peter, **Inteligência artificial: tradução da segunda edição**. Tradução de PublicCare Consultoria. Rio de Janeiro: Elsevier, 2004.

SOURCEFORGE.NET. Sítio oficial do repositório de projetos em Software Livre. Disponível em http://www.sourceforge.org/, acessado em 25/07/2005.

SUNASSEE, Nakiran N.; SEWRY, David A. **An Investigation of Knowledge Management Implementation Strategies**. Proceedings of the 2003 annual

research conference of the South African institute of Computer Scientists and Information Technologists. Páginas 24-36. 2003.

SWOOGLE.UMBC.EDU. **Sítio oficial.** Disponível em http://www.swoogle.umbc.edu/. Acessado em 25/07/2005.

TALLIS, Marcelo. **Semantic Word Processing for Content Authors**. Proceedings of the Knowledge Markup & Semantic Annotation Workshop. Florida, USA. 2003.

UNICODE.ORG. **Sítio Oficial**. Disponível em http://www.unicode.org/. Acessado em 25/07/2005.

W3.ORG/2001/SW/WEBONT. **Sítio oficial.** World Wide Web Consortium. Disponível em http://www.w3.org/2001/sw/WebOnt/. Acessado em 25/07/2005.