2068-PS

SUPPORT TO KNOWLEDGE MANAGEMENT IN VIRTUAL PRACTICAL COMMUNITIES

Júnio César de Lima (Instituto Federal Goiano – Campus Urutaí, Goiás, Brasil) - junio.ozn@gmail.com
Paulo Henrique Garcia Mansur (Instituto Federal Goiano – Campus Urutaí, Goiás, Brasil) - paulomansur@yahoo.com.br

Abstract

The Web was designed to support sharing information resources globally. Currently, this sharing is limited, particularly as for the recovery of imperfect information and the Web hidden. The Semantic Web appears as a possible solution for such limitations. Virtual communities of practice are groups of people join and virtually share a common set of knowledge. A use of Semantic Web technologies can contribute much in the implementation of tools support to these communities. This article discusses and presents a tool based on the principles of Web Semantics that assists in the management of knowledge circulates in virtual communities of practice on the Web.

Keywords: Virtual Practical Community, Knowledge Management, Semantic Web, Information, OWL.

Apoio à Gestão do Conhecimento em Comunidades Virtuais de Prática

Resumo

A Web foi projetada para suportar o compartilhamento de recursos de informação em nível global. Atualmente, esse compartilhamento é limitado, principalmente, no que se refere à recuperação de informações imperfeitas e à Web oculta. A Web Semântica surge como uma possível solução para tais limitações. As comunidades virtuais de prática são grupos de pessoas que se associam virtualmente e compartilham um conjunto comum de conhecimentos. A utilização das tecnologias da Web Semântica pode contribuir muito na implementação de ferramentas de suporte a essas comunidades. Este texto discute e apresenta uma ferramenta baseada nos princípios da Web Semântica que auxilia na gestão do conhecimento que circula em comunidades virtuais de prática na Web.

Palavras-chave: Comunidade Virtual de Prática, Gestão do Conhecimento, Web Semântica, Informação, OWL.

1 Introdução

A WWW (*World Wide Web*) ou simplesmente Web guarda uma enorme quantidade de informações criadas por uma grande variedade de diferentes organizações, comunidades e indivíduos. Hoje temos, por exemplo, empresários, artistas, políticos, religiosos e pessoas comuns que fazem uso de redes sociais de conhecimento, como o Orkut e o Twitter, para se relacionar com outras pessoas, comentando seus trabalhos ou criticando outros, por exemplo.

Todavia, o estrondoso crescimento do número usuários e de páginas disponíveis tem dificultado, cada vez mais, a localização, o acesso e a manutenção de informações na Web. Esses problemas ocorrem porque usuários, com diferentes níveis de conhecimento, disponibilizam uma vasta quantidade de recursos, sem nenhuma padronização. Surge daí a insatisfação dos próprios usuários, pois quando estes realizam uma pesquisa na Web, geralmente, demoram muito para obter um resultado. O mais grave, entretanto, é que os resultados das pesquisas trazem informações, que em grande parte, não satisfazem as necessidades do usuário. Fica a cargo deste a filtragem destas informações para obter respostas às suas perguntas.

Esses problemas vêm do fato de que o conteúdo das informações é disponibilizado, principalmente, para o entendimento humano, havendo, portanto, uma grande dificuldade na execução dos serviços de recuperação processados por máquina.

Uma das propostas para resolução desses problemas é o uso da Web Semântica. A Web Semântica representa uma revolução do processamento da informação e, por conseqüência, uma revolução na maneira de se obter e organizar os conhecimentos. Pretende-se, com a Web Semântica, que os recursos disponibilizados sejam expressivos o bastante para que máquinas e/ou agentes de software sejam capazes de processar e “entender” o real significado dos dados. Ela é baseada em várias tecnologias que estão sendo desenvolvidas e recomendadas pelo W3C, tais como XML, RDF/RDFS e ontologias (OWL) (HENDLER; LASSILA; BERNERS-LEE, 2001; BRAY et al, 2004; MANOLA; MILLER, 2004; HARMELEN; MCGUINNESS, 2004).

Outra área que tem atraído a atenção no campo de gerenciamento de conhecimento são as comunidades virtuais de práticas (Virtual Communities of Practice - VCoPs) na Web. Uma comunidade virtual de prática possui três aspectos fundamentais que a definem: o domínio de conhecimento, a comunidade e a prática.

Uma comunidade deve centrar-se em um domínio (contexto) específico de conhecimento. Um contexto compartilhado é essencial para o desenvolvimento de uma comunidade, pois dá sentido e orientação às discussões que ocorrem e pode ajudar os membros a decidirem em que direção levar a comunidade.

Um aspecto chave é que cada membro deve possuir um nível mínimo de conhecimento do domínio para poder participar da comunidade. Além disso, esse tipo de comunidade pode ser instalada na Web, adotando os recursos da Web Semântica, principalmente as ontologias, que podem ser utilizadas para delimitação do contexto.

Outra característica de uma comunidade de prática é a forma como os membros se mantém engajados em atividades conjuntas, através de discussões, ajuda mútua e compartilhamento de informações. A interação é requisito chave para membros pertencerem a uma comunidade (WENGER, 1998).

O foco nas VCoPs possibilita uma delimitação de contexto que represente seus respectivos domínios de conhecimento. Ferramentas e ambientes que usem idéias da Web Semântica podem fazer uso de um domínio bem definido, permitindo que os recursos possam se encaixar dentro de um contexto. Por exemplo, em uma comunidade cujo domínio de conhecimento é “programação orientada a objetos”, o conceito de “herança” não está relacionado com transmissão de bens pós-morte. Isto permite implementar ferramentas de compartilhamento ou recuperação de recursos que considerem o contexto, que é previamente conhecido e representado na VCoP.

O compartilhamento de conhecimento entre os seus membros é um dos principais focos de qualquer comunidade. Assim, as bases de conhecimento das comunidades tendem a ter uma grande quantidade de informações. Muitas destas informações não são explicitamente interrelacionadas ou contextualizadas através do uso de metadados. Dessa forma, a localização ou descoberta de conhecimento, implícito ou explícito, nessas bases pode ser uma tarefa difícil. A utilização de mecanismos de inferência pode ajudar na minimização ou mesmo na solução deste problema.

Além disso, na atualidade, as grandes organizações estão criando e sustentando com sucesso as VCoPs, para poderem acessar o valioso conhecimento que existe dentro das instituições. Esses conhecimentos são acumulados com o passar do tempo pelas VCoPs, que se desenvolvem compartilhando suas práticas pela interação entre os membros sobre problemas e soluções e pela construção de uma base de informações compartilhada.

Com isso, as VCoPs tendem a ser um mecanismo adequado para gestão do conhecimento devido à facilidade, à rapidez e à abrangência da comunicação, juntamente com sua capacidade de extrair conhecimento de recursos a partir de uma variedade de fontes e disponibilizá-los de forma transparente. Para alcançar seus objetivos, as VCoPs devem desenvolver práticas adequadas para fomentar a confiança entre seus membros no processo de gestão de conhecimento.

O objetivo desse texto é discutir e apresentar uma ferramenta para auxiliar o gerenciamento de conhecimento em um ambiente de comunidades virtuais de prática na Web, que adota as tecnologias da Web Semântica definidas pelo W3C.

2 Gestão do Conhecimento

Uma das principais tendências que se pode identificar no momento atual é o deslocamento do paradigma de sociedade industrial para sociedade da informação ou sociedade do conhecimento. Essas mudanças ocorrem em razão dos avanços tecnológicos das últimas décadas, em especial a Web, possibilitando uma nova dimensão de comunicação, bem como um novo método de compartilhamento de conhecimento, disponibilizado por diversas fontes do saber humano.

Na sociedade do conhecimento, onde este é o único recurso realmente significativo, é necessário que existam espaços de divulgação de textos, vídeos, imagens e outros que retratem o pensamento das empresas, de profissionais, pesquisadores, professores, estudantes etc.. Em todos possíveis espaços de divulgação é necessária uma ordem e coerência das informações para que o conhecimento seja realmente adquirido.

O termo Gestão do Conhecimento surgiu em meados da década de 1990 e é a confluência entre as áreas cobertas pela Tecnologia da Informação e pela Administração. Uma definição para o termo gestão do conhecimento, segundo Landini e Damiani (2001), “é um processo sistemático de conexão entre pessoas e entre estas e o conhecimento de que necessitam para tornar mais eficaz suas ações e permitir a criação de novo conhecimento”. Identificando, capturando, validando e transferindo informações, as iniciativas de gestão do conhecimento melhoram o desempenho da organização, tendo por objetivo não só o compartilhamento desse conhecimento, apesar desta ser uma de suas funções.

Outra definição para gestão do conhecimento é de um processo necessário para capturar, codificar e transferir conhecimento para uma organização conseguir atingir plenamente os seus objetivos (ARCHER, 2006).

Inicialmente, a gestão do conhecimento foi vista como uma forma inovadora para solucionar diversos problemas organizacionais, criando o que foi referenciado por Peter Drucker (2006) como “sociedade do conhecimento”. A importância global de gestão do conhecimento somente tem sido reconhecida recentemente, sendo tratada como um recurso crítico para o sucesso das organizações.

Entretanto, a gestão do conhecimento ainda está em sua infância. Se, por um lado, a tecnologia e o desenvolvimento das redes de informações possibilitam a difusão do conhecimento, por outro lado impulsionam a publicação direta da fonte ao consumidor. Isto gera uma falta de padrões para a disponibilização de documentos/informação na Internet e, consequentemente, dificulta a busca e a recuperação da informação, embora, em ambientes estáticos os problemas também apareçam. Logo, a tecnologia da informação encontra-se diante do crescente desafio de proporcionar à essa “nova sociedade” a disponibilização de conhecimentos confiáveis, precisos, oportunos e relevantes.

Além da falta de padrão na disponibilização, o processamento da informação para geração de conhecimento, por si só, é uma atividade complexa uma vez que a informação, dependendo do contexto e do domínio do conhecimento, pode ter significados diversos, ou seja, um termo ora representa um conceito ora pode representar outro e constituir relações conceituais diferentes. Esse problema é conhecido como relacionamento semântico de homônimos. Além desse tipo de relacionamento semântico, uma informação pode ter vários outros, como, por exemplo, o relacionamento semântico de sinônimo (um termo X tem quase o mesmo significado que o termo Y) e de subclasse (um termo X tem significado mais restrito do que o termo Y).

Apesar das evoluções dos processos comunicativos, as organizações têm encontrado dificuldades na busca por processos que minimizem ou resolvam os problemas relacionados à gestão do conhecimento de forma a se manterem competitivas frente às inúmeras necessidades de inovação (DAVENPORT; PRUSAK,

2000). Para que esse objetivo organizacional aconteça é necessário criar mecanismos e processos que facilitem o manuseio do conhecimento.

O primeiro desafio é o entendimento acerca do que é conhecimento. O conhecimento, definido por Davenport e Prusak (2000, p. 6), "é uma mistura fluída de experiência condensada, valores e informação contextual, a qual proporciona uma estrutura para a avaliação e incorporação de novas experiências e informações". Segundo os autores, sua origem e aplicação estão na mente de quem conhece e nas organizações costumam serem encontrados em documentos, arquivos, em documentos, arquivos, processos, práticas, etc.

Já, segundo Fischler e Firschein (apud HAYKIN, 1999), conhecimento se refere à informação armazenada ou a modelos utilizados por uma pessoa ou máquina para interpretar, prever e responder apropriadamente ao mundo exterior.

Na literatura, o conhecimento pode existir em duas formas: tácita e explícita. O conhecimento explícito é o conhecimento que existe em documentos, livros, software e outros meios. O conhecimento expresso na forma explícita pode ser facilmente reproduzido e distribuído com custo baixo ou nulo e, por essa mesma razão, é mais difícil de garantir seu uso não autorizado.

O conhecimento tácito é o conhecimento que o indivíduo adquiriu ao longo da vida, que está na cabeça das pessoas, podendo ser a parte mais valiosa do conhecimento de um indivíduo ou organização. Geralmente, é difícil de ser formalizado ou explicado a outra pessoa, pois é altamente localizado, subjetivo e inerente às habilidades pessoais e requer, para sua transferência, um envolvimento direto de fontes e usuários e um processo ativo de ensinamento e aprendizagem.

Segundo Nonaka e Takeuchi (1997), o conhecimento é criado através de um processo cíclico em que o conhecimento tácito é convertido em formalizações, símbolos e torna-se publicamente acessível na forma de conhecimento explícito e vice-versa. Transformar o conhecimento tácito em explícito, tornando-o reutilizável por outras pessoas, não é uma tarefa simples, pois, como descrito anteriormente, o conhecimento tácito é pessoal e difícil de ser articulado em uma linguagem formal, já que envolve diversos fatores (emocionais, psicológicos e outros).

Para haver a transformação do conhecimento e, consequentemente, a expansão deste, é necessária uma interação social entre os dois formatos, ou seja, fazer uma "mistura" dos conhecimentos tácitos e explícitos. A partir daí, o conhecimento individual acumulado precisará ser socializado novamente de forma a gerar novos conceitos, quando aplicado às novas necessidades. O processo de conversão do conhecimento tácito em explícito e vice-versa pode ser realizado de quatro modos (AHMAD; AL-SAYED, 2006):

- Socialização: conversão de tácito para tácito. É o processo de compartilhamento de experiências entre os indivíduos de um grupo e que se desenvolve, frequentemente, por meio da observação, da imitação e da prática. Dessa forma, é possível transferir o conhecimento tácito entre os indivíduos e a associação de um mesmo tipo de conhecimento a diferentes contextos individuais;
- Externalização: conversão de tácito em explícito. É o processo de organização do conhecimento tácito em conhecimento explícito, por meio

de metáforas, analogias, conceitos, hipóteses e modelos, permitindo a criação de conceitos novos e explícitos baseando-se no conhecimento tácito;

- Combinação: conversão de explícito em explícito. É o processo de sistematização de conceitos em um sistema de conhecimento. Assim, envolve a combinação de conjuntos de conhecimento explícito (tais como, a classificação, a sumarização, a pesquisa e a categorização das informações) com a utilização da tecnologia de banco de dados, podendo levar à criação de novos conhecimentos;
- Internalização: conversão de explícito para tácito. É o processo pelo qual o conhecimento explícito torna-se ferramenta de aprendizagem, por meio de manuais ou documentos, e volta a assumir um contexto abstrato e subjetivo para cada membro na organização.

Estes quatro tipos de conversão de conhecimento: socialização (conhecimento compartilhado), externalização (conhecimento conceitual), combinação (conhecimento sistêmico) e internalização (conhecimento operacional), ao longo do tempo, formam a denominada Espiral do Conhecimento (Figura 1).

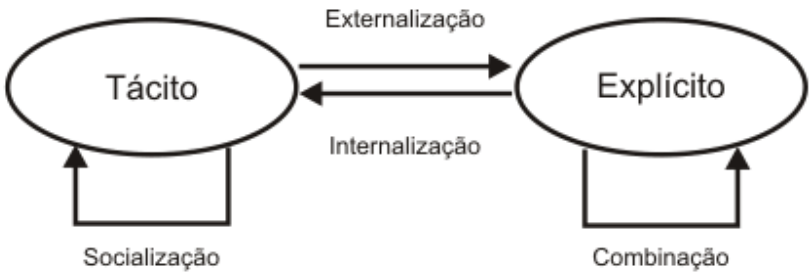

Figura 1 - Espiral do Conhecimento.

Na Figura 1, o conhecimento tácito constitui a base do conhecimento organizacional, pois é ele que está na mente dos membros que a compõem e pode ser transmitidos aos outros membros. Neste caso, pode-se modelar tais agrupamentos através de Comunidades Virtuais de Prática, VCoPs, empregando as tecnologias da Web Semântica.

3 Objetivos e Motivação

O trabalho aqui descrito faz parte de um projeto maior denominado Projeto DWeb, (acrônimo para Dream Web, ou seja, a Web dos sonhos). Ele tem como meta criar um meio universal para a troca de dados, informações e conhecimentos. Para tanto, o usuário deve ser contemplado com uma visão transparente, acessível e intuitiva com relação aos recursos de informação providos e manipulados pelo seu ambiente (CARVALHO et al, 2005).

O objetivo geral do Projeto DWeb é contribuir com a consolidação e o desenvolvimento da Web Semântica oferecendo um ambiente para o estabelecimento de comunidades virtuais de prática na Web. Este ambiente deverá facilitar a comunicação entre os membros das comunidades e entre estas com a utilização de componentes semânticos associados aos recursos disponibilizados. Melhorando-se o processo de comunicação, espera-se tornar mais ágil e eficiente o desenvolvimento de atividades colaborativas dentro e fora de comunidades específicas.

No Projeto DWeb são adotadas as tecnologias da Web Semântica dentro do contexto de comunidades virtuais de prática. Estas fornecem um modelo bastante adequado para se criar um contexto bem definido de forma a permitir a recuperação de informações (recursos de informação) com alto nível de relevância para as necessidades dos usuários.

O Projeto DWeb é composto por um núcleo de funcionalidades que disponibiliza várias ferramentas para uma gestão completa de uma comunidade. Dentre essas ferramentas, uma das principais é a que gerencia o conhecimento, ou seja, uma ferramenta capaz de capturar o conhecimento que circula por uma comunidade na Web e armazená-lo de forma estruturada em uma base de conhecimento. Essa base servirá como repositório de todo o conhecimento da comunidade e poderá ser utilizada para possíveis buscas.

Como discutido anteriormente, para se estruturar uma comunidade virtual de prática na Web três aspectos fundamentais devem ser levados em consideração: o domínio de conhecimento, a comunidade e a prática. Para implementar esses três aspectos, a ferramenta desenvolvida utiliza os principais recursos da Web Semântica, dentre eles o modelo RDF para armazenar as informações (práticas) e a ontologia para definir o domínio, ou seja, a comunidade e mecanismos de inferência que fazem uso da ontologia para melhorar a exatidão das buscas.

Logo, a ferramenta faz parte do núcleo central do Projeto DWeb. Ela permite que os vários membros possam ter acesso a uma comunidade e, assim, ao contexto específico de seu interesse. Esse contexto é modelado através de uma ontologia. Uma vez selecionado o contexto, o usuário pode armazenar e recuperar conhecimento em um repositório comum a todos os membros da comunidade. Uma visão geral desse cenário é mostrado na Figura 2.

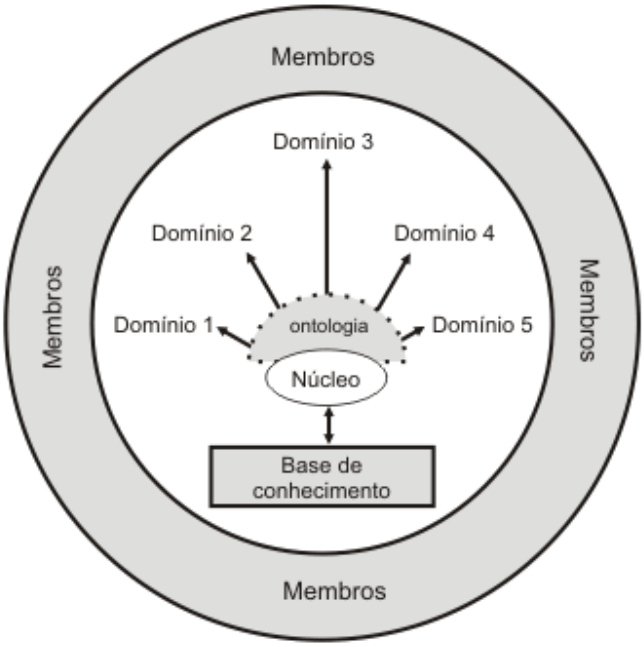

Figura 2 - Visão abstrata de uma comunidade no Projeto DWeb.

O principal objetivo do módulo Gerenciador de Conhecimento no Projeto DWeb é auxiliar o trabalho cooperativo na Web, monitorando todo o conhecimento que é transmitido pelos membros, através da fala e/ou escrita, capturando o conhecimento que é realmente importante e armazenando-o em uma base de conhecimento. Na primeira versão implementada da ferramenta, a captura do conhecimento do usuário é feita através de interfaces gráficas simples guiadas por conceitos definidos em ontologias. Suas principais funcionalidades são:

- Permitir que membros da comunidade possam inserir e compartilhar seu conhecimento de forma simples e intuitiva;
- Permitir que membros da comunidade possam recuperar o conhecimento de outros membros de forma simples e intuitiva;
- Armazenar o conhecimento informal dos membros da comunidade em uma base de conhecimento estruturada;
- Usar ontologias para delimitar o domínio;
- Usar ontologias para auxiliar os membros tanto na inserção como na busca de conhecimento.

A seguir essa ferramenta será mais bem detalhada, bem como o desenvolvimento das ontologias utilizadas para testá-la.

4. Implementação da Ferramenta

4.1 Delimitação do Contexto

Uma comunidade virtual de prática na Web deve possuir um domínio bem definido e específico. Uma forma de se delimitar esse domínio é através da especificação de uma ontologia. Para a realização de testes de validação das implementações feitas optou-se por adotar o domínio Agropecuário. A escolha desse domínio se deveu ao fato de se tratar de uma área bem compreendida pelos autores e também por se tratar de uma importante área que ainda não dispõe de muitas ferramentas informatizadas.

Um domínio pode conter vários contextos específicos de interesses para os membros. Por exemplo, em uma comunidade de Agropecuária, pode-se incluir cultivo de soja, criação de peixes, pecuária bovina etc. Assim, quando se cria uma comunidade virtual de prática na Web, uma questão de suma importância é a definição do seu domínio e dos contextos desse domínio. Logo, um usuário deve poder escolher em qual contexto da comunidade ele tem interesse.

Uma ontologia pode ser vista como sendo uma hierarquia de contextos (conceitos), modelada na forma de árvore, sendo que os nodos intermediários e o nodo raiz apontam para os seus contextos mais específicos (nodos filhos) e os nodos folhas guardam o URI do contexto que realmente o usuário tem interesse, também modelado por uma ontologia.

A primeira ontologia construída, chamada de *animal.owl*, modela essas definições de contexto. A linguagem escolhida para a codificação da ontologia foi OWL (HARMELEN; MCGUINESS, 2004), uma vez que ela é recomendada pelo W3C. Ela delimita todos os contextos relacionados à criação de animais em uma fazenda. A Figura 3 mostra como essa ontologia pode ser representada em forma de árvore.

Os conceitos (classes) definidos no nodo raiz e nos nodos intermediários na ontologia *animal.owl* possuem um nome, um rótulo (*label*), um comentário e suas superclasses, caso existam. Já um conceito definido como folha possui, além das definições de um nodo intermediário (nome do conceito, rótulo, comentário e superclasses), a propriedade *ontoURI*, que foi definida para essa ontologia. Essa propriedade guarda o URI que define uma ontologia do contexto que o conceito representa.

Como exemplo, foi construída a ontologia chamada de *doenças.owl*. A cada conceito criado na ontologia *doenças.owl*, foram atribuídas algumas propriedades de restrição que se fizeram presentes em todos os conceitos relacionados às doenças representadas. Por exemplo, toda doença tem causas, sintomas, tratamento e uma descrição para um animal em particular. Além disso, observou-se ser interessante armazenar um comentário sobre a doença, ou seja, uma definição textual mais técnica sobre ela.

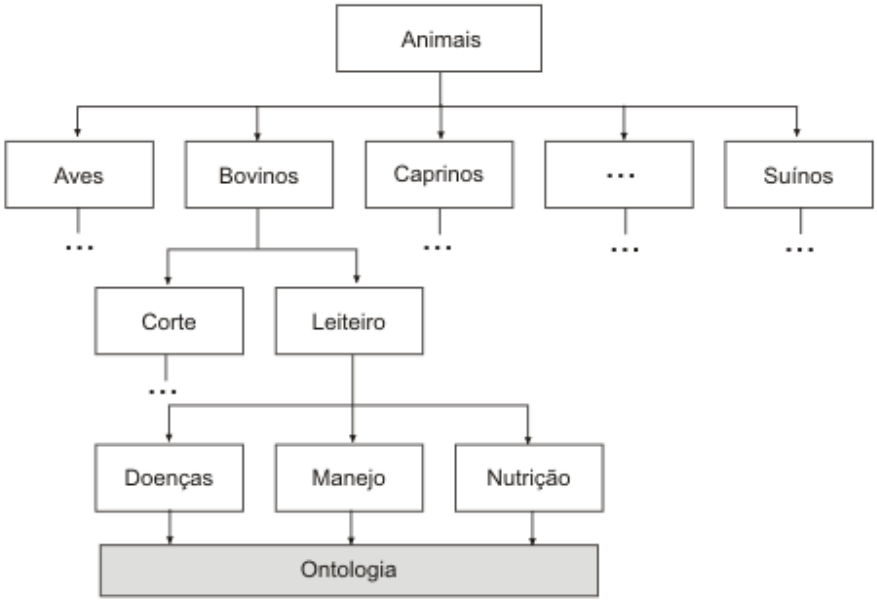

Figura 3 - Uma representação, em forma de árvore, da ontologia *animal.owl*.

Além dos conceitos relacionados especificamente às doenças, outros também se fizeram necessários. Por exemplo, o conceito "sintoma", que é uma propriedade presente em todas as doenças, pode possuir valores. Esses valores foram restritos às instâncias da classe (conceito) chamada de *Sintoma*.

4.2 Funcionamento da Ferramenta

No desenvolvimento da ferramenta foram utilizadas aplicações que auxiliam no trato das tecnologias da Web Semântica. Dentre as várias ferramentas recomendadas pelo W3C, a que foi julgada ser a mais indicada foi o Arcabouço Jena (HEWLETT-PACKARD, 2006). Ele foi escolhido por se tratar de uma das ferramentas mais completas atualmente para desenvolvimento de aplicações na Web Semântica, já estando disponível há mais de 6 anos e sendo uma das mais citadas da literatura. Além disso, é mantido por uma empresa de renome mundial e alguns de seus desenvolvedores são também integrantes do próprio W3C.

A linguagem de trabalho utilizada na implementação foi Java. Ela foi utilizada porque é uma linguagem orientada a objetos independente de plataforma, bastante utilizada na Internet; é robusta, segura e *multithreaded* (SUN, 2009). Além disso, o arcabouço Jena é também implementado em Java.

As principais classes implementadas foram: *Principal*, *OntoRaiz*, *ListaConceitos*, *SelecionaDados* e *Armazena*. Além dessas classes, são utilizadas várias classes do arcabouço Jena, dentre elas as principais são: *Model*, *OntModelSpec*, *ModelFactory*, *OntModel* e *OntClass*. Detalhes sobre as funcionalidades das classes do arcabouço Jena utilizadas na implementação da

ferramenta podem ser encontrados na página dos desenvolvedores (http://jena.sourceforge.net/).

A primeira classe a ser criada foi a classe *Principal*. O objetivo dessa classe é permitir que o usuário possa escolher a comunidade de seu interesse. Essa classe possuiu a funcionalidade de permitir que o usuário possa escolher entre armazenar o seu conhecimento ou fazer pesquisas na base de conhecimento. Essa funcionalidade foi codificada em forma de botões de opção, onde o usuário pode selecionar apenas uma das duas opções.

4.2.1 Captura do Conhecimento

Quando o usuário deseja inserir seu conhecimento na base, o primeiro passo é carregar a ontologia que vai guiar a ferramenta na captura desse conhecimento. Para carregar uma ontologia o uso das classes *OntModelSpec*, *ModelFactory* e *OntModel* do arcabouço Jena se faz necessário. A ontologia utilizada para guiar o usuário na escolha do contexto é a ontologia *animal.owl*. Essa ontologia é carregada na classe *OntoRaiz*. Essa classe, além de carregar a ontologia, possui todos os métodos necessários para poder percorrê-la.

O primeiro método da classe *OntoRaiz* a ser chamado é *buscaConceitos()*. Esse método tem por objetivo percorrer a ontologia (neste caso, *animal.owl*) escolhida e selecionar todos os conceitos que são filhos diretos do conceito raiz.

Quando o método *buscaConceitos()* é chamado, ele lista todas as subclasses diretas do elemento raiz e passa a listagem para uma classe interface que disponibiliza esses conceitos para o usuário, através de uma interface gráfica (o termo interface quando citado se refere a uma interface gráfica com o usuário).

Essa interface gerada representa cada conceito como um botão. Cada botão representa uma subclasse, sendo que cada subclasse identifica um contexto. Logo, o usuário poderá selecionar qual contexto ele tem interesse clicando em um botão.

Quando se seleciona um conceito, através dos botões, o método *buscaFilhos()* da classe *OntoRaiz* é chamado. Esse método é parecido com o método *buscaConceitos()*, só que ele busca os elementos filhos de qualquer conceito e não somente do conceito raiz, como ocorre em *buscaConceitos()*. Além disso, ele verifica se o conceito selecionado é folha ou não. A Figura 4 mostra um diagrama com as principais classes e métodos utilizados nesse processo de captura do conhecimento do usuário.

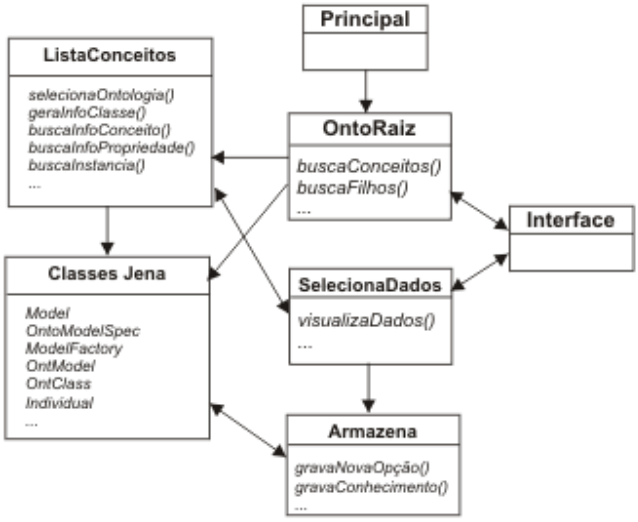

Figura 4 - Diagrama dos principais métodos utilizados na captura do conhecimento

Caso o conceito seja folha, então o método *selecionaOntologia()* é chamado, passando como argumento o conceito que o usuário selecionou. Neste caso, esse conceito representa um contexto que é modelado através de uma ontologia. O método *selecionaOntologia()* faz parte da classe *ListaConceitos*. A função dessa classe é permitir que se possa percorrer uma ontologia de um contexto específico dado o seu URI (seu endereço). As principais funcionalidades dessa classe são: recuperação de todos os conceitos ascendentes ou descendentes de um dado conceito, recuperação de todas as instâncias de um conceito, recuperação de todos os conceitos equivalentes e recuperação de todas as propriedades de um conceito, que podem ser as definidas pelo criador da ontologia, (como "sintoma" e "raça", por exemplo), bem como as definidas pela linguagem OWL, como *label* e *comment*.

Quando o método *selecionaOntologia()* é chamado, ele lista todos os conceitos, que devem ser visualizados pelo usuário da ontologia passada como parâmetro. Alguns conceitos não devem ser visualizados pelos usuários, pois são usados apenas para dar suporte à definição da ontologia. Isso é feito verificando-se se um conceito possui a propriedade *visualizacao*, que se aplica a todos os conceitos que devem ser visualizados.

Depois de listar todos os conceitos visualizáveis de uma ontologia, uma nova interface gráfica com o usuário é gerada. Esses conceitos são implementados na interface em forma de botão. Quando um botão é clicado, uma nova interface é criada para que o usuário possa disponibilizar o seu conhecimento em relação ao conceito selecionado. Essa interface, é implementada na classe *SelecionaDados*.

A classe *SelecionaDados* busca todas as informações relacionadas ao conceito selecionado. Ou seja, ela lista todas as superclasses do conceito e todas as suas propriedades, inclusive as propriedades das suas superclasses. Devem-se listar as superclasses de uma classe pelo seguinte motivo: uma subclasse é uma classe específica de uma superclasse, tendo todas as suas características, além das suas

específicas. Logo, toda instância de uma subclasse também é instância das suas superclasses, embora o contrário não seja verdadeiro.

Quando a classe *SelecionaDados* é instanciada, o método *buscaInfoConceitos()*, da classe *ListaConceitos*, é chamado. Esse método recebe como argumento o conceito selecionado pelo usuário. Dado esse conceito, o método busca na ontologia todos os outros conceitos (superclasses) relacionados com o conceito selecionado. Para cada um desses conceitos, o método *buscaInfoPropriedades()*, da classe *ListaConceitos*, é chamado. Esse método retorna todas as propriedades (características) de um conceito passado como parâmetro.

Depois que todas as propriedades são recuperadas, o método *vizualizaDados()*, da classe *SelecionaDados*, é chamado, passando como argumento todas as propriedades recuperadas. Esse método implementa uma nova interface que tem por objetivo mostrar ao usuário todas as opções que ele tem para poder descrever o seu conhecimento. Cada opção corresponde a uma das propriedades selecionadas. Para cada opção também é mostrada uma lista com todos os valores que existem na base de conhecimento para a propriedade. A recuperação dessa lista de valores é feita através do método *buscaInstancias()*, da classe *ListaConceitos*, que recebendo como parâmetro um conceito, retorna todas as instâncias desse conceito no arquivo instanciado da ontologia.

A busca pelas instâncias de uma classe é feita seguindo-se a definição das informações manipuladas por Jena, ou seja, através de triplas:

- O sujeito da tripla é a instância;
- O predicado da tripla é a propriedade *type* (tipo do sujeito);
- O objeto da tripla é o conceito (classe).

Todos os valores da lista são buscados em um arquivo que mantém a base de conhecimento, que é uma instância da ontologia selecionada. Essas informações bem como todo o conhecimento da comunidade é armazenado no arquivo chamado de *Fatos.owl*.

No caso específico da ontologia sobre doenças em vacas leiteiras (*doenças.owl*), é retornado ao usuário uma interface com as seguintes propriedades definidas na ontologia: o tipo de informação que ele deseja inserir, o identificador do animal, a causa da doença, os sintomas da doença, o tratamento realizado, qual foi o resultado do tratamento, quanto tempo durou o tratamento, uma porcentagem de melhora ou piora para cada sintoma e um comentário de todo o processo, com as próprias palavras, caso ache necessário.

Caso o usuário queira escolher uma opção que não foi recuperada pelo método *BuscaInstancias*, ele pode adicionar essa nova opção na base de conhecimento. Isso é feito através do método *gravaNovaOpcao()* da classe *Armazena*.

Por fim, o método *gravaConhecimento()*, da classe *Armazena*, é chamado para armazenar o conhecimento do usuário. Essa classe tem por objetivo gravar no arquivo de conhecimento, *Fatos.owl*, todos o conhecimento transmitido pelo usuário.

Depois de se armazenar o conhecimento do usuário na base, outros usuários podem acessar e recuperar esse conhecimento. Logo, a ferramenta deve ser capaz,

da mesma forma que armazenou o conhecimento em uma base, também recuperá-lo.

4.2.2 Busca do Conhecimento

Quando o usuário escolhe uma comunidade e a opção para recuperar conhecimento, várias interfaces são criadas, como as discutidas anteriormente. Quando ele escolhe um contexto que ele tem interesse (um nodo folha), a ontologia que define esse contexto é carregada.

Essa ontologia é carregada na classe *InterTipoBusca*. Essa classe, além de carregar a ontologia, monta uma interface com todas os possíveis tipos de busca que o usuário pode realizar na base de conhecimento. Essas opções são guiadas pela ontologia que define o contexto, neste caso, a ontologia *doencas.owl*.

O primeiro passo realizado pela classe *InterTipoBusca* é buscar todas as propriedades das classes (conceitos) existentes na ontologia. Depois, para cada propriedade, são buscados todos os seus valores. Por fim, uma interface é montada com todas as opções para a realização da busca.

Depois que o usuário seleciona a busca desejada, a classe *Inferencia* é instanciada. O objetivo dessa classe é realizar inferências na base de conhecimento, guiada pela ontologia. Quando essa classe é instanciada o método *tipoBusca()* é chamado. Esse método recebe como parâmetro todas as opções de busca selecionadas pelo usuário.

A função do método *tipoBusca()* é definir qual tipo de busca vai ser realizado, pois esta pode ser feita com base em uma ou mais restrições. Além disso, uma busca por uma única restrição pode ser diferente de outra busca com uma única restrição também. Por exemplo, buscar todas as vacas com certos sintomas e buscar todas as vacas com certas doenças. A Figura 5 mostra um diagrama com as principais classes e métodos utilizados nesse processo de busca do conhecimento.

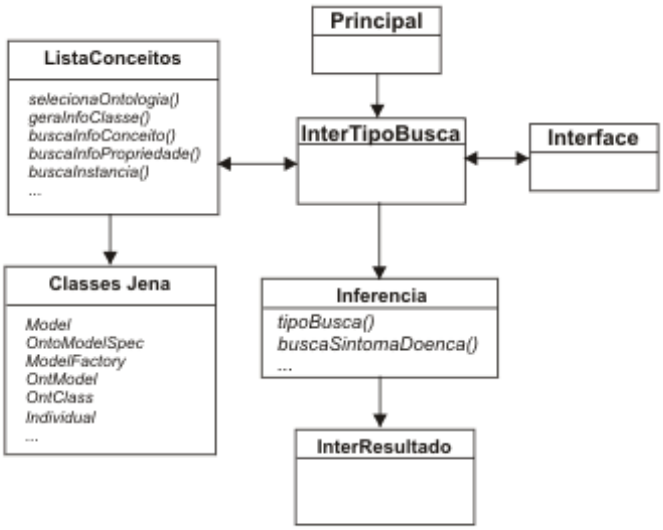

Figura 5 - Diagrama dos principais métodos utilizados na busca do conhecimento

Além disso, esse método também permite ao usuário especificar que tipo de informações a busca deve retornar. Por exemplo, caso o usuário selecione certa doença, o retorno da busca pode trazer somente os possíveis sintomas ou os sintomas e os tratamentos realizados. Existem também as possíveis causas e em quais vacas essa doença foi registrada. Com isso, ele pode saber o peso, a idade e a raça da vaca. Essa consulta é realizada através de uma interface onde o usuário pode selecionar através de botões de opções, todas as informações que ele tem interesse.

Depois que o usuário escolhe o retorno, o método *tipoBusca()* escolhe e chama o método que deve ser utilizado para realizar a inferência. Por exemplo, caso o usuário queira saber quais são os possíveis sintomas causados por certa doença, o método *tipoBusca()* deve selecionar o método *procuraSintomaDoenca()*, que lista todos os sintomas da doença selecionada. O resultado da busca desse método é retornado e uma interface é montada mostrando ao usuário o resultado da busca.

4.3 Uso da Ferramenta

Quando o usuário tem acesso à ferramenta, a primeira interface que ele percebe é a da classe *Principal*. Essa interface permite o usuário escolher a comunidade (domínio) que ele tem interesse. Depois de o usuário selecionar a comunidade, ele pode escolher o que deseja fazer dentro da comunidade selecionada. Neste caso, ele tem duas alternativas:

- Disponibilizar o seu conhecimento na base de conhecimento da comunidade selecionada;
- Realizar buscas na base de conhecimento.

Caso o usuário escolha a opção de disponibilizar o seu conhecimento, será então instanciada a classe *OntoRaiz*. Neste momento, o método *BuscaConceitos()* é chamado e todos os conceitos filhos do conceito raiz são retornados através de uma interface gerada com base nesse retorno. Junto com o nome do contexto também é exibida uma figura que indica o principal aspecto do contexto. Por exemplo, em relação ao conceito "bovino", além do próprio nome do conceito, também é exibida a figura de um bovino. Essa figura tem por objetivo facilitar ao usuário a identificação do contexto que ele tem interesse.

O processo se repete, ou seja, todos os conceitos filhos diretos de um item selecionado pelo usuário são retornados e uma nova interface é mostrada com esses novos conceitos. Neste caso, esse processo é visto como uma alternativa de navegação na ontologia, sendo que em cada interface aparece somente os conceitos relevantes naquele domínio.

Esse processo se repete até que seja selecionado um conceito que seja folha, quando então o método *SelecionaOntologia()* é chamado, passando como argumento o conceito selecionado (o contexto escolhido).

Dado que o usuário selecionou um conceito folha, por exemplo, "doenças", esse conceito possui como valor para a propriedade *ontoURI* o URI que define o contexto para o qual ele deseja disponibilizar o seu conhecimento. Neste momento, uma nova interface é criada exibindo todos os conceitos do contexto, modelado pela ontologia selecionada anteriormente (*doencas.owl*), que devem ser visualizados pelo usuário. Essa ontologia tem por objetivo delimitar o domínio do contexto especificado. Isto é, dentro desse contexto o objetivo dos membros é discutir sobre as doenças das vacas leiteiras, somente com os conceitos ali definidos.

Os conceitos não definidos nessa ontologia não poderão ser usados. Caso esses conceitos sejam importantes, eles devem ser adicionados à ontologia seguindo um procedimento formal, como por exemplo, todos os membros devem concordar com a adição de um novo conceito e uma pessoa responsável pela comunidade realiza esse procedimento.

Nesse momento, o usuário pode informar seu conhecimento. Por exemplo: “A minha vaca mimosa estava doente; ela estava com dor de barriga (diarréia). A causa foi tristeza. Apliquei um medicamento a base de penicilina durante 14 dias e ela melhorou”.

O procedimento usado pela ferramenta é o seguinte: o usuário ao selecionar um conceito, ou seja, uma doença, uma nova interface é criada para que o usuário possa disponibilizar o seu conhecimento em relação ao conceito selecionado. Essa interface, implementada na classe *SelecionaDados*, é criada via informações encontradas na ontologia. Nessa interface, são disponibilizados ao usuário todas as propriedades (características), do conceito que ele selecionou e que ele poderá utilizar para descrever o seu conhecimento. Para cada propriedade há várias opções de seleção.

Por exemplo, na opção "sintoma" o usuário pode escolher quantos sintomas ele achar necessários ou, se não houver sintomas, ele pode escolher a opção padrão, ou seja, que não percebeu nenhum sintoma. Caso o usuário queira escolher uma opção que não aparece para a sua escolha, ele pode clicar no botão *Nova Entrada*, do lado de cada opção de propriedade, para poder adicionar na base de conhecimento um novo conhecimento.

Caso o usuário queira armazenar a informação que um dos sintomas de sua vaca seja febre e não exista essa opção na lista, ele pode adicionar esse novo conceito na base de conhecimento de forma isolada. Depois de adicionado, essa nova instância aparece como opção para o usuário selecionar.

Após a estruturação de seu conhecimento guiado pela ontologia, o usuário pode validar a sua intenção de armazenar o seu conhecimento, lendo o último rótulo nesta interface. Este rótulo capta todas as escolhas do usuário e cria uma frase na língua portuguesa sobre o conhecimento do usuário.

Essa frase sintetiza o que o usuário selecionou. Isto é, toda vez que o usuário seleciona uma instância para as propriedades, é feito um processo parecido com o que ocorre no processamento natural de linguagem, ou seja, o conhecimento estruturado é convertido em uma frase na língua portuguesa.

Caso o usuário concorde com o que ele selecionou, ele pode clicar no botão *Gravar*, que por sua vez chama o método *gravar()*, que é definido na classe *Armazena*.

Depois de se armazenar o conhecimento do usuário na base de conhecimento, outros usuários podem acessar e recuperar esse conhecimento. Quando o usuário tem acesso à interface principal da ferramenta, implementada pela classe *Principal*, o usuário, além de selecionar a comunidade, pode selecionar a opção de realizar buscas na base de conhecimento. Quando essa opção é selecionada uma nova interface é criada. Como se está trabalhando com a ontologia *doenças.owl*, as propriedades recuperadas serão as seguintes: o nome da doença, sintoma, causa e tratamento. O usuário pode selecionar somente uma dessas propriedades, as duas ou todas.

Depois de selecionar a busca, uma nova interface é criada para perguntar ao usuário que tipo de informações devem ser recuperadas. A partir dessa escolha, o usuário recebe uma nova interface com as informações recuperadas. Por exemplo, caso o usuário selecione "diarréia negra" como sintoma, podem ser retornadas todas as informações sobre as vacas que tiveram esses sintomas. São mostrados ao usuário as possíveis causas, qual o tratamento realizado e os efeitos desse tratamento. Ou ainda o usuário pode selecionar quais informações ele que saber. Por exemplo, dado um sintoma ele que saber somente as possíveis causas ou as causas e o tratamento.

5 Conclusão

Novas tecnologias e paradigmas relacionados ao uso de semântica no contexto de recuperação de informações têm contribuído para o desenvolvimento do que se denomina Web Semântica. Fazendo uso de diversas ferramentas e facilidades de comunicação da Web, principalmente, levando-se em conta as possibilidades trazidas pela Web Semântica, as VCoPs ganham espaço e tornam-se, hoje, uma importante área de pesquisa e aplicação, possibilitando o desenvolvimento de ambientes cada vez mais completos e úteis para o gerenciamento de conhecimentos, ou seja, o compartilhamento das melhores práticas pelos membros das comunidades.

Para que haja uma efetiva transferência de conhecimento entre membros de uma comunidade é necessário um elevado nível de confiança entre eles e uma forte cultura de cooperação e colaboração. Esta confiança é desenvolvida através de práticas de trabalho que incentivam e permitam aos membros trabalhar em conjunto em projetos e problemas. Com isso, os membros tendem a cooperar e colaborar dentro da comunidade realizando uma efetiva gestão do conhecimento que circula pela VCoPs.

A adoção das tecnologias da Web Semântica no contexto de VCoPs mostra-se um caminho bastante interessante para a gestão do conhecimento nessas comunidades. Além disso, com o uso das tecnologias da Web Semântica, a gestão de VCoPs tende a ser aprimorada, pois a facilidade de inserção e busca de conhecimento faz com que os membros se sintam mais à vontade para disponibilizar seus conhecimentos, enriquecendo sinergicamente as comunidades.

A concepção da ferramenta descrita foi fundamentada nas tecnologias da Web Semântica. O uso dessas tecnologias empresta características muito

interessantes como, por exemplo, a interoperabilidade de dados proporcionada por XML, a padronização da estrutura dos recursos, através do emprego do modelo RDF, a definição e a delimitação de contexto e fornecimento de definições compartilhadas, possibilitando o reuso de conceitos, proporcionado pelo emprego de ontologias.

A ferramenta permite a inserção e o compartilhamento de conhecimento de forma transparente aos membros da comunidade, ou seja, possibilita aos membros inserirem seu conhecimento, sua experiência do dia-a-dia, sem que precisem conhecer as tecnologias que estão sendo empregadas, como XML, RDF e ontologias (OWL), para dar suporte a essa inserção. Isso é oferecido através de interfaces gráficas que permitem aos membros navegar em uma comunidade e escolher o contexto de seu interesse, inclusive com o uso de figuras que simbolizam os contextos. Somente são exibidos aos usuários os conceitos que possam interessar em um dado instante (de acordo com a ontologia), os demais são omitidos.

Uma vez que o usuário teve acesso a um contexto, a ferramenta disponibiliza uma interface que permite a entrada de conhecimento. Ela é montada com base nas escolhas anteriores do usuário e na ontologia que define o contexto. Com isso, os termos empregados pelo usuário são todos previamente selecionados, ficando a cargo deste apenas a seleção dos que melhor condizem com o conhecimento que ele tem interesse em compartilhar. O mesmo processo é usado para a busca de conhecimento.

Desta forma, as VCoPs caminham no sentido de proporcionar ferramentas intuitivas, que permitam a troca de conhecimento e o desenvolvimento de práticas de forma simples, através do uso de ontologias de domínio, ferramentas de inferência, buscas semânticas, armazenamento de conhecimento e o casamento destes com as informações de interesse, conferências, fóruns, sítios baseados em ambientes colaborativos, ferramentas para vídeo conferências e ensino à distância, dentre outros recursos.

6 Bibliografia

AHMAD, K.; AL-SAYED, R**.** Community of Practice and the Special Language “Ground”, In: CLAKE, S.; COAKES, E. **Encyclopedia of communities of practice in information and knowledge management**, Hershey (EUA): Idea Group, 2006, p. 77-88.

ARCHER, N. A Classification Communities of Practice, In: CLAKE, S.; COAKES, E. **Encyclopedia of communities of practice in information and knowledge management**, Hershey (EUA): Idea Group, 2006, p. 21-29.

BRAY, T. et al. **Extensible markup language (xml) 1.0**. 2004. Disponível em <http://www.w3.org/TR/2004/REC-xml-20040204/>. Acesso em 29 set. 2009.

CARVALHO, C. L. et al. **Uma Arquitetura para Desenvolvimento da Web Semântica Baseada em Comunidades Virtuais de Prática**. Instituto de Informática. Universidade Federal de Goiás. 2005, acessado em Setembro de 2009.

DAVENPORT, T.; PRUSAK, L. **Working Knowledge**. Boston (EUA): Harvard Business School, 2000.

DRUCKER, P. **Classic Drucker**. Boston (EUA): Harvard Business School Press, 2006.

HARMELEN, F.; MCGUINNESS, D. L. **OWL Web Ontology Language Overview**. W3C Recommendation, Fevereiro 2004. Disponível em <http://www.w3.org/TR/2004/REC-owl-features-20040210/>. Acesso em 29 set. 2009.

HAYKIN, S. **Neural Networks: A Comprehensive Foundations**. Upper Saddle River (EUA): Prentice Hall, 1999.

HENDLER, J.; LASSILA, L.; BERNERS-LEE, T. The Semantic Web. **Scientific American**, v. 5, n. 284, p. 34-43, maio 2001.

HP LABS SEMANTIC WEB RESEARCH. **Jena - A Semantic Web Framework for Java**. 2006. Disponível em <http://jena.sourceforge.net>. Acesso em 02 out. 2009.

LANDINI, M.; DAMIANI, J. Knowledge Management: An Exploratory Study in the Brazilian Aerospace Program. In: Portland International Conference on Management of Engineering and Technology, 2001, Portland - OR. **Proceedings...** Portland - OR : Portland State University, 2001. v. 2.

MANOLA, F.; MILLER, E.. **Rdf primer**. 2004. Disponível em <http://www.w3.org/TR/2004/REC-rdf-primer-20040210/>. Acesso em 29 set. 2009.

NONAKA, I.; TAKEUCHI, H. **The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation**. New York (EUA): Oxford University Press, 1997.

SUN MICROSYSTEMS. **The Source for Java Developers**. 2006. Disponível em <http://java.sun.com/>. Acesso em 29 set 2009.

WENGER, E. Communities of Practice: Learning, Meaning and Identity. Cambridge: Cambridge University, 1998.

WORLD WIDE WEB CONSORTIUM (W3C). Disponível em <http://www.w3.org/>. Acesso em 29 set. 2009.