

e-Learning e a Web Semântica

Jorge Luis Goñi

jgoni@les.inf.puc-rio.br

Maria Cristina Pfeiffer Fernandes

fernandes@serraon.com.br

Carlos José Pereira de Lucena

lucena@inf.puc-rio.br

PUC-RIOInf.MCC12/02 Junho, 2002

Abstract: The objective of this work is to propose a tool based in agents, denominated *Web Semantic Search (WebSS)*, using the structure of the Semantic Web, where through *ontologies* generated by the Protegé-2000 tool, it will be possible to recover learning contents in AulaNet servers, according to the granularity proposed by the ContentNet Framework. It is presented an Analysis Model of, the MESSAGE Methodology (*Methodology for Engineering Systems of Software Agent*), wich is a Agent-Oriented Software Engineering Methodology, that extends the current techniques of object-oriented software development to simplify and to formalize the process of developing a multi-agent system.

Keywords: e-Learning, Metadata , Web Semantic, RDF/RDFS, Multi-Agent

Resumo: O objetivo deste trabalho é propor uma ferramenta baseada em agentes, denominada *Web Semantic Search (WebSS)*, usando a estrutura de Web Semântica, onde através de ontologias geradas pela ferramenta Protegé-2000, será possível recuperar conteúdos educacionais em servidores AulaNet, procurando respeitar a granularidade proposta pelo Framework ContentNet. Nesta abordagem, estamos interessados em apresentar uma proposta do Modelo de Análise, dentro dos padrões definidos pela Metodologia MESSAGE (*Methodology for Engineering Systems of Software Agents*), que sendo uma metodologia de Engenharia de Software orientada a agentes, procura estender as atuais técnicas de desenvolvimento de software orientadas a objetos para simplificar e formalizar o processo de desenvolvimento do sistema multi-agentes.

Palavras-chave: e-Learning, Metadados, Web Semântica, RDF/RDFS, Multi-Agentes

1 Introdução

Com o avanço das Tecnologias da Informação e da Comunicação (TIC), e com o crescimento da Sociedade da Informação pelo uso da Internet as instituições, corporativas e acadêmicas, estão utilizando técnicas cada vez mais sofisticadas para a geração e disseminação do conhecimento no meio em que atuam.

A *World Wide Web* (WWW), é o principal veículo de comunicação utilizado mundialmente para o acesso, recuperação e utilização de informações. No entanto, com o crescimento desordenado da Web, está se tornando ineficiente a busca por informações mais precisas.

Hoje, uma pessoa leva normalmente muito tempo para achar uma informação na Web, em função de palavras-chave ambíguas fornecidas pelas ferramentas de busca como resultados de pesquisas.

O ramo da Engenharia de Software que trata das aplicações baseadas na Web, está evoluindo para lidar com uma infra-estrutura diferente da Web atual, conhecida como a *Web Semântica*. A principal finalidade desta nova abordagem é melhorar a organização das informações na forma de dados estruturados, possibilitando buscas mais eficientes por agentes de software.

O objetivo deste trabalho é propor uma ferramenta baseada em agentes, denominada *Web Semantic Search* (WebSS), usando a estrutura de *Web Semântica*, onde através de *ontologias* geradas pela ferramenta Protegé-2000, será possível recuperar conteúdos educacionais em servidores AulaNet [1], procurando respeitar a granularidade proposta pelo Framework ContentNet [2].

No artigo também é descrita a literatura pesquisada (item 2), é feita uma descrição do editor de ontologies Protegé-2000 (item 3), e é apresentado o ambiente AulaNet [1] explicitando os serviços oferecidos (item 4).

Também é descrita a arquitetura de agentes proposta (item 5), alguns trabalhos relacionados (item 6), trabalhos futuros (item 7) e no item 8 são listadas algumas considerações finais. Em *Anexos* (item 9) são referenciados alguns padrões em ambientes de gerenciamento e distribuição do conhecimento para a geração de aplicações de novos serviços que estão sendo disponibilizados na Web.

2 Fundamentação Teórica

Atualmente muitas instituições vêm adotando a tecnologia de *e-Learning* (2.1), que exige uma estruturação eficiente das informações disponibilizadas atualmente na Web, para uma implementação eficiente conforme descrito em 2.2. Essa estruturação está originando

uma nova estrutura para a Web, conhecida por Web Semântica, apresentada em 2.3. Por outro lado, para a construção da arquitetura proposta, será necessário utilizar a metodologia MESSAGE orientada a agentes, descrita em 5.2.

2.1 e-Learning

e-Learning é uma estratégia habilitada para a Web, que oferece um amplo conjunto de soluções, que estimulam o crescimento de comunidades de conhecimento/aprendizado, para aumentar o desempenho das instituições. [4]

Usando a tecnologia de *e-Learning* a aprendizagem não ocorre somente a partir de instrução digital, mas principalmente a partir do acesso a informações bem estruturadas, correspondendo a uma forma inovadora para o aprendizado.

Portanto, uma estrutura de *e-Learning* deve conter treinamento on-line e sistemas de gestão do conhecimento (*Knowledge Management* - KM), que devem fornecer o suporte necessário para a criação de uma cultura de aprendizado na instituição.

A estruturação do conhecimento é a base de um sistema de KM para a implementação do *e-Learning*. Quando um conteúdo é bem-estruturado, preciso e de fácil utilização, o aprendizado ocorre. Caso contrário as informações perderão valor em pouco tempo. Em função disso, é fundamental a forma de identificação, categorização, organização e rotulação das informações, bem como seu gerenciamento.

As principais vantagens de um sistema de KM bem estruturado são: permite o aprendizado dos usuários na aplicação das informações armazenadas; deixa gravada a história da instituição, gerando uma memória on-line; disponibiliza recursos adicionais para a manipulação de informações; permite acrescentar idéias e insights dos usuários, aumentando o banco de conhecimento da instituição.

Um dos avanços tecnológicos mais promissores na área do *e-Learning* é a criação de soluções baseadas em objetos educacionais, onde cada um corresponde ao menor bloco de instrução ou informação, elaborado de forma independente capaz de transmitir um significado [4].

Utilizando objetos educacionais, é possível criar bibliotecas de conhecimento, permitindo que cursos diferentes utilizem um mesmo objeto. Outra vantagem está na personalização do aprendizado a partir da seleção e configuração daqueles objetos que atendam auxiliem o aprendiz na construção e apropriação do próprio saber.

Um objeto educacional poderá ser: um programa em *Java*, um arquivo de som, um arquivo de imagem, um filme ou um programa de simulação.

Já existe a catalogação de objetos educacionais usando o padrão IMS [5], que incluem especificações para a interoperabilidade, permitindo reuso. Essa padronização contribuirá principalmente para a redução de custos e agilização na produção de recursos educacionais.

Para incentivar a utilização dessa tecnologia foi criado o consórcio *Educational Objects Economy* (EOE), nos Estados Unidos [6], um consórcio mantido pela *National Science Foundation*. Deste fazem parte universidades, empresas e editoras envolvendo educadores, desenvolvedores e comerciantes. No site do EOE existem mais de 2.500 objetos educacionais disponíveis, dos quais aproximadamente 25% tem seus códigos abertos permitindo o compartilhamento e o reuso.

2.2 Estruturação da Informação

Hypertext Markup Language – (HTML), é uma linguagem básica de marcação utilizada nas páginas da Web [7], e é uma derivação da *Standard Generalized Markup Language* (SGML) [8].

HTML foi desenvolvida para a marcação de hipertextos, tanto para os autores dos documentos quanto para os desenvolvedores de *browsers*, os softwares interpretadores dos códigos HTML. A principal limitação apresentada por esta linguagem é a falta de estruturação dos dados de um documento HTML, ou seja, não há distinção entre a codificação de seus componentes básicos (conteúdo, estrutura e apresentação dos dados)

A simplicidade da linguagem HTML não está dando conta da demanda gerada em diversos serviços *on-line* em áreas tais como: e-commerce, saúde, educação e outros. Por isso tornou-se necessária a criação de um padrão que pudesse atender às características de cada uma destas aplicações e serviços que estão sendo disponibilizadas na Web.

O padrão que está sendo definido para atender à essa demanda, baseia-se na utilização de uma linguagem conhecida como XML (*eXtensible Markup Language*).

XML é um aperfeiçoamento da linguagem HTML, pois utiliza *tags* (marcação) para descrever também a estrutura do documento e não somente a aparência deste. XML é uma linguagem que permite solucionar o problema da sintaxe padrão para a troca de informações. Ao todo são sete os principais componentes da XML: declarações, elementos, atributos, entidades, comentários, instruções de processamento e DTD – *Document Type Definition*.

Criar uma DTD é o primeiro passo na estruturação de documentos em XML. Uma DTD define a estrutura de um documento através da especificação de seus elementos básicos, seus relacionamentos, seus atributos, entidades e o conjunto de tags associadas para marcar o documento.

Esse controle em relação ao layout do documento, melhorará a representação dos dados, permitindo uma melhor organização das informações armazenadas, às quais se poderá também agregar um sentido semântico. E isto facilitará a manipulação das informações atualmente disponibilizadas na Web, bem como a qualidade e a velocidade das pesquisas.

Uma das aplicações mais avançadas de XML é o RDF (*Resource Description Framework*), que usa a notação XML como a sintaxe de codificação e descrição para metadados. O principal objetivo do RDF é permitir a automatização do processamento de recursos, facilitando o intercâmbio de informações via Web [9].

E isso é possível a partir da URI (*Universal Resource Identifier*), que deve estar associada a cada elemento do RDF, da mesma forma como se utiliza um link numa página Web. Por isso, as URLs (Uniform Resource Locators) são os tipos mais comuns de URIs.

O RDF possui uma variedade de aplicações entre as quais pode-se destacar:

- Catalogação avançada de recursos e seus relacionamentos dentro de um único sistema ou entre vários sistemas;
- Pesquisa mais acurada de arquivos de dados, pois os dados em si estão melhor identificados;
- Filtragem mais precisa de dados para sistemas de avaliação de conteúdo mais viáveis;
- A capacidade de representar conjuntos inteiros de documentos como um único e grande documento quando apropriado;
- Estabelecimento de relacionamentos seguros entre documentos e computadores para facilitar o intercâmbio de idéias e de recursos bem como de comércio eletrônico.

2.3 Web Semântica

A *Web Semântica*, foi criada pelo mesmo cientista que inventou a WWW - *World Wide Web*, o físico inglês Tim Berners Lee [10]. Segundo Lee, a *Web Semântica* estrutura o conteúdo significativo das páginas Web, criando um ambiente onde agentes de software percorrem página por página para executarem tarefas solicitadas pelos usuários. O desenvolvimento da *Web Semântica* envolve duas importantes tecnologias: XML e RDF.

Um exemplo de como a *Web Semântica* funcionará, suponha que uma pessoa deseja marcar uma consulta médica e que o médico deve estar localizado, no máximo, cinco quilômetros de seu trabalho. Usando a *Web Semântica*, após a solicitação, os agentes saem pesquisando a informação nas páginas da Web.

Através de comparações automáticas entre, a agenda da pessoa que solicitou a consulta e a agenda dos médicos encontrados, serão mostrados os horários disponíveis para que a pessoa possa escolher um deles [11].

A questão gira, então, em torno da seguinte questão: como fazer com que os computadores entendam o conteúdo da Web? O primeiro passo será organizar e estruturar as informações e o segundo será adicionar semântica às informações da Web, de forma tal que agentes de software possam compreendê-las.

Na *Web Semântica*, um conceito importante estará associado à uma URI, (*Uniform Resource Locator*), cuja principal função é assegurar que conceitos não são apenas palavras num documento, mas estão amarrados a um único significado, que qualquer pessoa poderá encontrar na Web.

Mas como deverá proceder um agente, durante o processo de busca, quando encontrar dois bancos de dados que usam URIs diferentes para um mesmo conceito?

Para resolver este problema, existem as ontologias que vão fornecer o vocabulário necessário, para a comunicação entre os agentes de software e as páginas da Web e mostrar as relações entre os conceitos.

Portanto uma ontologia é um documento ou um arquivo onde estão definidos formalmente as relações entre conceitos. Uma ontologia na Web é uma taxonomia formada de classes e subclasses de objetos, relacionadas entre si mais um conjunto de regras de inferência [12].

Ao usar as ontologias, melhorará principalmente a velocidade das pesquisas, uma vez que os agentes pesquisarão somente as páginas que se referem à informação desejada, ao invés de todas as páginas usando palavras-chave ambíguas, conforme realizado atualmente pelas ferramentas de buscas já conhecidas.

Introduzir semântica à Web não é o mais difícil. O problema maior será criar padrões que sejam usados globalmente. A curto prazo, existirão DTDs padrões para certos tipos de aplicações como por exemplo: *e-commerce*, cadastro de alunos, sistemas de bibliotecas, ensino a distância entre outras, que possibilitarão aos projetistas/analistas e desenvolvedores de software, a utilização de um padrão denominado “ontologias” (vide subitem II.3) para a criação e troca de dados. Por exemplo, o grupo IEEE *Standard Upper Ontology* (SUO), vem trabalhando na definição de ontologias para aplicações nas áreas de: educação, *e-commerce* e linguagem natural [13].

Usando RDF não é possível criar ontologias, no entanto, ele é a base para diversas linguagens com essa finalidade, tais como RDF *Schema* (RDFS) [7], um recurso que define as primitivas para a criação de ontologias.

A especificação do RDF *Schema* fornece os mecanismos necessários à definição de: elementos, classes de recursos, possíveis restrições de classes e relacionamentos, e detecção de violação de restrições, constituindo uma hierarquia com múltiplas relações de herança.

A especificação do RDF *Schema*, fornece também recursos suficientes para criar modelos RDF, através de sistemas hierárquicos representados por taxonomia de assuntos como mapas de sites, dicionários e esquemas de classificação de bibliotecas.

3 A Ferramenta Protegé-2000

Protegé-2000 é uma ferramenta gráfica para a edição de ontologias e aquisição de conhecimento que vem sendo usado há algum tempo por especialistas principalmente nas áreas de medicina e indústria para a modelagem de domínios. [14]

Com o surgimento da *Web Semântica*, novas linguagens estão sendo definidas, dentre as quais RDF e RDF *Schema*. Em função disso foram implementadas algumas

mudanças na interface do usuário, na representação interna e no framework do Protegé-2000, tornando-o altamente customizável, para permitir a modelagem conceitual dessas linguagens.

Por isso, o Protegé-2000 é altamente customizável, sendo um poderoso editor de ontologias para diversas linguagens novas desenvolvidas para a Web Semântica,

A grande vantagem do Protegé-2000, é que o usuário não precisa se preocupar com a sintaxe da linguagem usada na Web, bastando se concentrar nos conceitos e nas relações do domínio e nos fatos que precisam ser representados.

O modelo de conhecimento do Protegé-2000 tem classes, instâncias destas classes, propriedades (*slots*), descrevendo atributos de classes e instâncias, e restrições (*facets*), expressando informações adicionais sobre as propriedades. E essa forma de representação se aproxima bastante do modelo de conhecimento utilizado pelas novas linguagens.

O Protegé-2000 armazena a definição da ontologia e suas instâncias em arquivos separados, ou seja, enquanto os arquivos de ontologia têm a extensão *.pont*, os arquivos com as instâncias tem extensão *.pins*.

A interface do usuário é simples, consistindo de várias tabelas para edição de diferentes elementos de uma base de conhecimento. Também oferece telas contendo informações detalhadas para a aquisição do conhecimento.

Uma característica importante da interface com o usuário é que permite alterar facilmente a hierarquia de classes, bem como criar novas superclasses para uma determinada classe.

O Protegé-2000 é uma ferramenta aberta que permite adicionar novas funcionalidades, através da criação de *plug-ins* apropriados. A arquitetura atual do Protegé-2000 suporta uma grande quantidade de *plug-ins* [14].

Até o presente momento, os principais problemas encontrados ao se usar esta ferramenta foram: dificuldades em escrever constraints e a representação de operações que envolvem união, interseção, disjunção.

4 Ambiente AulaNet

O AulaNet [1] é um ambiente de software baseado na Web, desenvolvido no Laboratório de Engenharia de Software - LES - do Departamento de Informática da PUC-Rio, para administração, criação, manutenção e participação em cursos a distância.

O ambiente AulaNet auxilia o docente na tarefa de disponibilizar o conteúdo de seu curso na Internet. O AulaNet separa o conteúdo da navegação, fazendo com que os docentes só se preocupem com a produção dos conteúdos didáticos usando suas ferramentas habituais, como o editor de textos, e deixem por conta do ambiente a gerência e

a navegação dos aprendizes. Além disso, o AulaNet oferece ao docente uma gama de serviços, que podem ser usados no curso de forma a complementá-lo [15].

Em cursos do AulaNet, os docentes podem assumir basicamente três papéis: coordenador do curso, docente co-autor e mediador. O coordenador é o responsável por estruturar o curso, selecionando e configurando os serviços que serão disponibilizados, definindo a ementa, a metodologia e outras informações do curso. O docente co-autor é responsável por produzir e inserir os conteúdos didáticos nos serviços selecionados pelo coordenador. O mediador é o facilitador do grupo, responsável por manter a ordem, motivar e avaliar os aprendizes.

Os serviços do AulaNet são divididos baseados no princípio de que para aprender em grupo, um indivíduo tem que compartilhar idéias (se comunicar), estar em sintonia com os outros participantes do grupo (se coordenar), e realizar as tarefas satisfatoriamente (cooperar) [15], o modelo da Figura 1 é baseado na Comunicação, Coordenação e Cooperação [16]. Todos os serviços do AulaNet são organizados baseados nestes conceitos.

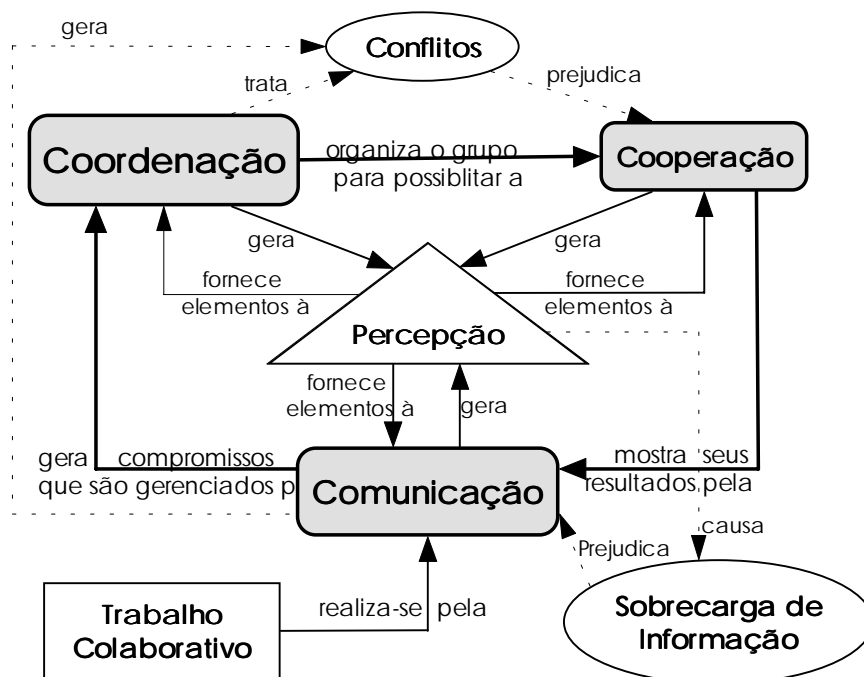


Figura 1 – Diagrama do trabalho colaborativo [17].

Os serviços de comunicação são: Contato com os Docentes, para correio eletrônico com os docentes; Lista de Discussão, para correio eletrônico com a turma; Conferências, para discussão textual assíncrona no estilo de fórum; Debate, para conferência síncrona textual no estilo de chat; e Mensagem para Participantes, para troca instantânea de mensagens entre os participantes simultaneamente conectados.

Os serviços de coordenação são: Avisos, para notificações; Plano de Aulas, para coordenação básica do fluxo do curso; Tarefas e Exames, para avaliação e auto-avaliação dos aprendizes; e Acompanhamento da Participação, para acompanhamento.

Os serviços de cooperação são: Co-autoria de Docente e Co-autoria de Aprendiz, para co-autoria do curso; Bibliografia e Webliografia para fornecer referências; Download, para transferência de conteúdos; e Documentação para conteúdos que não se enquadrem em nenhuma aula.

A utilização do AulaNet possibilita que sejam criados cursos a distância, através da Internet sem que o autor precise ter um conhecimento profundo do ambiente Web.

No AulaNet os conteúdos produzidos pelo professor referem-se a:

- um conteúdo: um arquivo mais um texto explicativo;
- uma aula: um conjunto de materiais mais um texto explicativo;
- uma documentação: um conjunto de materiais mais um texto explicativo;
- uma bibliografia: um texto explicativo e um arquivo;
- uma webliografia: um texto explicativo e um link;
- uma avaliação: inclui uma questão e
- uma tarefa: que envolve texto.

Um curso: o que compreende o grupamento de todos os conteúdos citados anteriormente, mais ferramentas de comunicação e coordenação.

Em função do grande número de servidores de informação AulaNet disponíveis atualmente, dentro e fora do país [18], surgiu a necessidade de se compartilhar conteúdos entre os diversos servidores [2]. Esta necessidade gerou outras demandas devido à dificuldade de localização do conteúdo desejado e, mais difícil ainda, à maneira pela qual este conteúdo será utilizado.

5 Arquitetura da Proposta

Dado que a filosofia do ambiente AulaNet está centrada em cursos, o nível de granularidade existente é o próprio curso. Ao buscar conteúdos educacionais usando a *Web Semântica*, a fim de tornar possível a reutilização de todos os conteúdos existentes em ambientes AulaNet, independentemente da hierarquia existente entre eles, será necessário transformar a granularidade do ambiente para o menor nível que possuíse semântica. Em nossa proposta, o docente co-autor utiliza ontologias para localizar o conteúdo, que corresponda ao menor nível de granularidade em diversos servidores AulaNet [1]. Através de ontologias geradas pela ferramenta Protegé-2000 e usando uma arquitetura de agentes como suporte, será possível recuperar conteúdos educacionais nos servidores de acordo com os padrões de granularidade definidos pelo Framework ContentNet [2].

6 Arquitetura Genérica

O comportamento colaborativo de uma organização de agentes pode ser especificado de acordo com a arquitetura proposta por Sycara [19] e compreende três diferentes tipos de agentes: (i) *agentes de interface* adquirem, modelam e utilizam requisições dos usuários, direcionando a coordenação do sistema para executar as tarefas; (ii) *agentes de tarefa* suportam decisões e resolvem tarefas oriundas da formulação do problema, exteriorizando-o, através de consultas e trocas de informação com outros agentes e (iii) *agentes de informação* processam inteligentemente e eficientemente a recuperação de informação.

A Figura 2 mostra como esta arquitetura de agentes é implementada pela ferramenta WebSS e como pode ser aplicada no contexto AulaNet para suportar a realização da busca. A solução proposta promove o acesso, descrição, localização e reuso de conteúdo educacional, respeitando a granularidade do ambiente para o menor nível semântico possível, de acordo com os padrões definidos no Framework ContentNet [2].

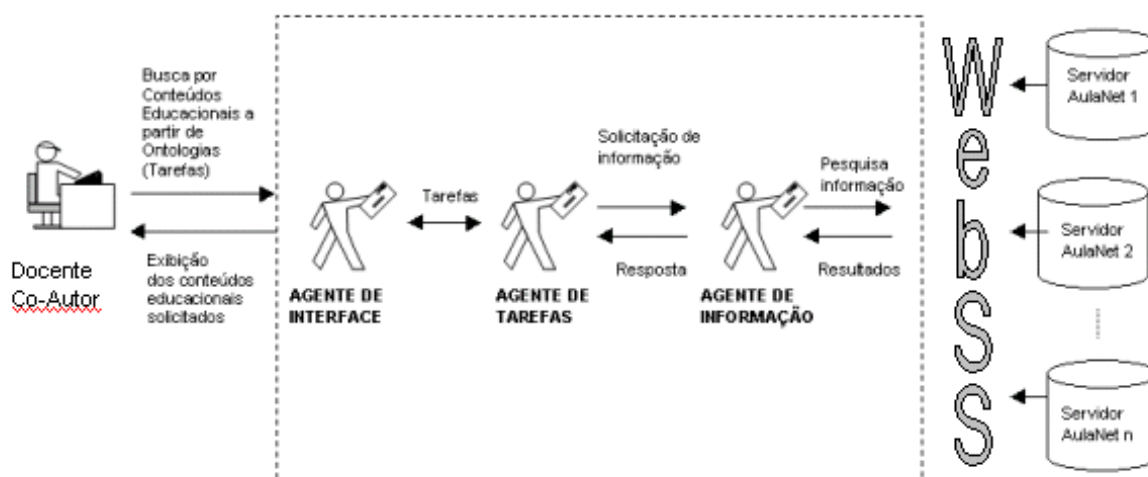


Figura 2. Passos de execução dos agentes para a realização da busca

Os passos para a execução da busca podem ser resumidamente descritos da seguinte forma:

Um docente co-autor informa o conteúdo que deseja localizar nos servidores AulaNet, iniciando a busca a partir de ontologias através de agentes de interface.

Os *agentes de tarefas* recebem as tarefas dos *agentes de interface* e requisitam dados aos *agentes de informação*.

Os *agentes de informação* efetuam pesquisas nas bases de conhecimento dos diversos servidores AulaNet, até localizarem e recuperarem as informações solicitadas pelo usuário.

Os *agentes de informação* disponibilizam os resultados das pesquisas aos *agentes de tarefa*, que disponibilizam em seguida os resultados ao usuário através dos *agentes de interface*.

Em todas os passos, a troca de informações entre os agentes é feita através de ontologias.

5.2 Modelagem

Nesta abordagem, estamos interessados em apresentar uma proposta para o Modelo de Análise, dentro dos padrões definidos pela Metodologia MESSAGE. MESSAGE (*Methodology for Engineering Systems of Software Agents*) [3] é uma metodologia de Engenharia de Software orientada a agentes, que procura estender as atuais técnicas de desenvolvimento de software orientadas a objetos para simplificar e formalizar o processo de desenvolvimento do sistema multi-agentes (MAS) [3].

Os principais fundamentos para a implementação da modelagem são baseados no modelo *Analysis Model* proposto pela metodologia MESSAGE. Este modelo é composto por 5 sub-modelos básicos: (i) *Organisation Model*; (ii) *Goal/Task Model*; (iii) *Agent Model*; (iv) *Domain Model* e (v) *Interaction Model*. A ordem da implementação dos cinco modelos é relativa, e na maioria das vezes depende do feeling do projetista.

(i) *Organisation Model* é o modelo estrutural básico que provê uma abstração muito importante e útil para o entendimento do MAS, focalizando as várias estruturas da organização e os relacionamentos entre os agentes que ela contém. Através do *Organisation Model*, pode-se ter uma visão geral da estrutura do sistema e o comportamento de um grupo de agentes, trabalhando para executar tarefas e atingir objetivos comuns [3]. *Organisation Model* é composto por dois modelos básicos: **o modelo estrutural** e **o modelo comportamental**.

O modelo estrutural é formado por uma arquitetura composta por três entidades básicas: *PurposeDescription*, *OrganisationStructure* e *WorkflowsStructure*, que descrevem respectivamente o propósito da organização, sua estrutura organizacional e o fluxo de trabalho para executar as tarefas. A Figura 3 mostra o *Organisation Model* da arquitetura proposta, cuja estrutura é derivada de um meta-modelo da metodologia. As classes estruturais raiz aparecem na cor cinza escuro no diagrama de classes.

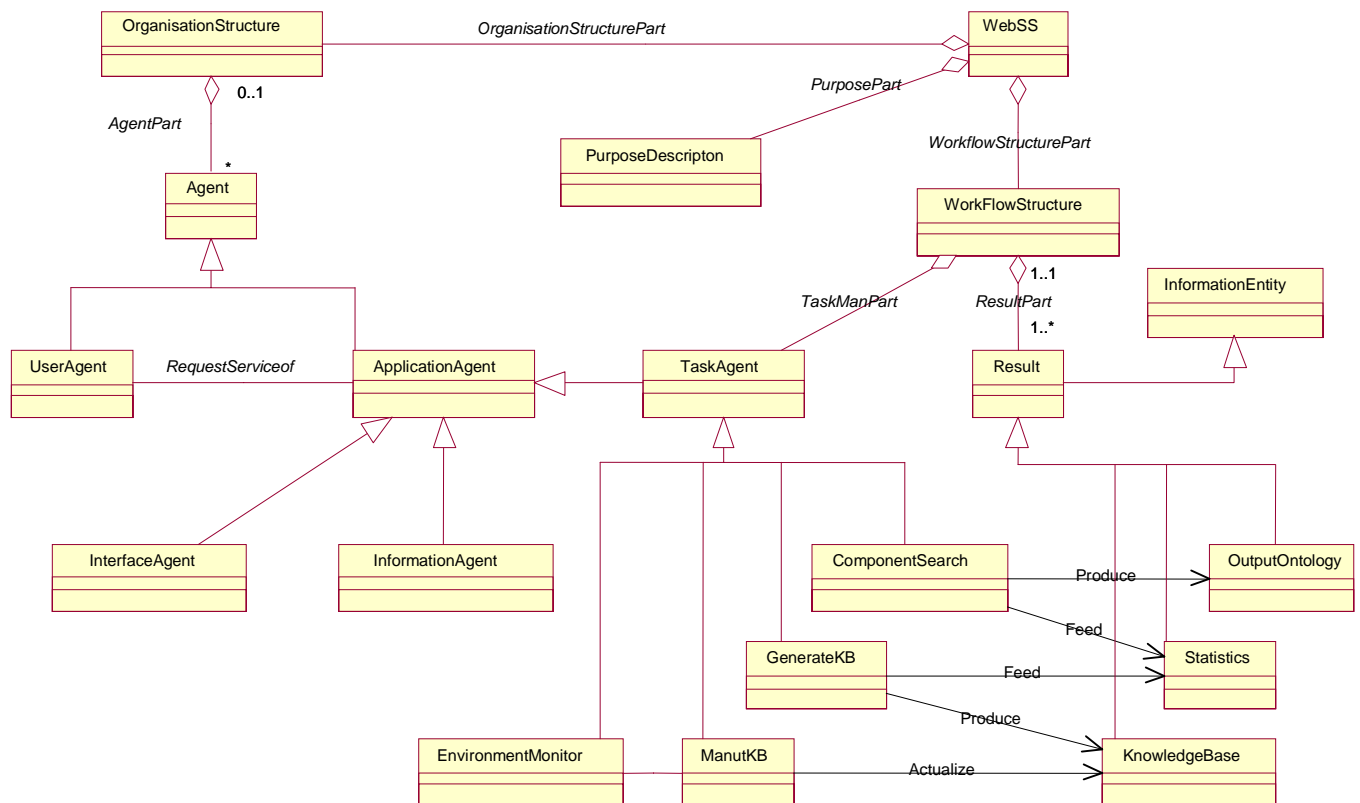


Figura 3. Estrutura do modelo Organisation Model de WebSS, derivado do meta-modelo Organisation Model.

A classe *PurposeDescription* descreve os propósitos da organização. Pode ser modelada em um documento em forma de template e deve conter uma descrição identificando a organização de agentes, a missão desta organização e os seus goals e responsabilidades.

A classe *OrganisationStructure* define a organização de agentes e os seus relacionamentos. Na WebSS, os agentes são compostos por dois grupos: *UserAgent* e *ApplicationAgent*. Existe entre eles um relacionamento *RequestServiceof* que indica que *UserAgent* controla *ApplicationAgent*.

A classe *ApplicationAgent* é decomposta em três tipos de agentes: *InterfaceAgent*, *InformationAgent* e *TaskAgent*.

A classe *WorkFlowStructure* descreve o *workflow* de tarefas e os resultados da organização. A classe *WorkFlowStructure* é composta por duas partes: (1) *TAT-Task* que representa a coleção de tarefas executadas na organização (*ComponentSearch*, *GenerateKB*, *Environment-Monitoring* e *MaintenanceKB*) para produzir os resultados (*OutputOntology*) esperados por *TAResult*; (2) *TAResult* produz resultados, representados

por uma estrutura de herança. As classes que compõem os resultados aparecem indicadas pelos relacionamentos *produce*, *feed* e *actualize*, que associam as tarefas ao resultado.

A classe *InformationEntity*, que aparece ao lado direito do diagrama, representa as entidades que são instâncias do domínio ou as classes que compõem *Domain Model*.

As classes estruturais de *Organisation Model* podem ser interpretadas como informações que documentam todos os aspetos relevantes da organização, e definem links para outros modelos, tais como *Goal/Task Model* e *Agent Model*.

O modelo comportamental mostra como a organização se comporta em relação às entidades externas, tais como outras organizações e usuários. Ela também define o comportamento interno da organização para atingir *goals* específicos ou executar tarefas específicas.

(ii) *Domain Model* tem como propósito apontar os conceitos específicos do ambiente com os quais o agente terá que lidar. Por exemplo, um software em desenvolvimento está relacionado a um determinado domínio. Na maioria das vezes, os agentes ativos no sistema terão que lidar com entidades que são instâncias dos conceitos específicos do domínio. Estas entidades correspondem a classe *InformationEntity*, representada em *Organisation Model*.

(iii) *Goal/Task Model* descreve como *goals* de alto nível são decompostos em *goals* de mais baixo nível, da mesma forma que tarefas são decompostas em conjuntos de sub-tarefas.

- Um *goal* descreve uma motivação. Um *goal* é uma situação, isto é, um conjunto de estados do mundo que um agente tem o compromisso de atingir ou manter. Um *goal* descreve um objetivo para um agente, ou o estado desejado do sistema o ambiente.
- Uma tarefa (*Task*) descreve uma atividade que pode ser decomposta em sub-tarefas e ações. Ações (*Action*) são tarefas atômicas que podem ser executadas pelo agente para satisfazer seus compromissos (*commitment*) de *goal*. O relacionamento ternário *Commitment* é definido entre dois agentes e um *goal*.

Um agente é um executor de uma tarefa. Ele pode ser humano, ou um software de computador ou qualquer outra entidade capaz de executar uma tarefa [20]. As principais características de um agente são sua identidade, seu comportamento e seu estado mental. O Estado Mental de um agente é uma representação do próprio agente em relação ao seu ambiente em termos de crenças e conhecimento, *goals*, intenções, etc. Seu Comportamento pode ser pensado como um conjunto de regras que determinam como ele responderá a eventos, tais como a chegada de novas informações [3].

(iv) *Agent Model* consiste da descrição da arquitetura de agentes da organização, e também de propósitos, relacionamentos, comportamentos e atributos individuais de agentes.

O comportamento individual de um agente pode ser descrito como um ciclo repetitivo de três estágios: (1) *percepção*: um agente percebe seu ambiente através de entidades de informação que descrevem eventos ou informações sobre o estado do mundo; (2) *decisão*: um agente decide quais tarefas e ações vai executar de acordo com seus objetivos, conhecimentos e crenças. Um agente decide executar ações que mudarão o estado do mundo de modo a satisfazer seus *goals* e (3) *ação*: um agente executa ações de acordo com suas decisões. As ações podem ser disparadas diretamente como um resultado de percepção (resposta reativa) ou como o resultado de um processo de tomada de decisão (resposta deliberativa).

A identificação de agentes, a partir dos modelos *Organisation Model* e *Goal/Task Model*, torna possível estabelecer uma correspondência entre *sub-goals* e propósitos de agentes e entre sub-tarefas e serviços que um agente pode prover.

(v) *Interaction Model* captura a forma pela qual agentes (ou papéis) tocam informações uns com os outros (assim como com o seu ambiente). O *Interaction Model* consiste de um conjunto de *Interactions*, e de uma representação mais detalhada em termos de *Interaction Protocols*.

Um protocolo de interação é um padrão de comunicação que permite uma seqüência de mensagens entre agentes. O conceito de *Interaction* em MESSAGE [3] é baseado em Protocol da metodologia Gaia [21], onde um protocolo pode ser visto como um institucionalizado padrão de interação. Isto é, um padrão de interação que foi formalmente definido e abstraído de acordo com uma seqüência específica de execução.

O objetivo desta abordagem é apresentar inicialmente uma visão geral da arquitetura, sem especificar detalhes de projeto e implementação. O foco principal deste artigo é mostrar a estrutura da Web semântica e sua aplicação no contexto AulaNet. Questões relacionadas com design e implementação estão ainda em fase de desenvolvimento, e serão abordadas em trabalhos subsequentes.

6 Trabalhos Relacionados

A questão da interoperabilidade de conteúdos educacionais foi anteriormente abordada em ContentNet [2], que se constitui em um ambiente que permite a busca de conteúdos educacionais AulaNet [1]. ContentNet é um framework para Interoperabilidade de Conteúdos Educacionais que utiliza o Padrão IMS [5], baseando-se na descrição do conteúdo que o usuário deseja localizar e buscar nos diversos servidores AulaNet. No padrão IMS, a todo conteúdo, é relacionada uma série de características como, por exemplo, o título do conteúdo, sua descrição, seu formato e o usuário pretendido para o mesmo. O papel do catalogador é atribuir valores a essas características. Sendo assim, todos os conteúdos AulaNet precisam ser descritos e catalogados em metadados. A partir do conhecimento de um novo conteúdo gerado, e contato com o professor que o criou, o catalogador consegue obter todas as informações necessárias para a catalogação. A granularidade do ambiente AulaNet, é proposta em ContentNet [2].

Construir repositório para *e-Learning* é um processo iterativo, onde a estrutura e conteúdo de um curso estão sempre se alterando. *Open Learning Repository* (OLR) [22] é um repositório para *e-Learning*, que foi desenvolvido devido a necessidade de separar o conteúdo de um curso de sua estrutura. OLR usa RDF/RDFS como linguagem de modelagem, para facilitar a troca de metadados entre os repositórios. Foi usado um banco de dados relacional para armazenar os metadados, que descrevem a informação sobre a estrutura e os comandos de acesso dentro de um curso em particular.

A modelagem de bases de conhecimento utilizando sistemas multi-agentes tem sido adotada em diversos trabalhos, como por exemplo *Indutive Specification Recovery* [23], XSearch [24] [25] e outros. XSearch apresenta uma abordagem baseada em agentes para recuperação de componentes em ambientes de objetos distribuídos, baseada na arquitetura proposta em MESSAGE [3]. O objetivo é manter uma base de conhecimento que permita identificar, localizar e recuperar componentes em bases de conhecimento, caracterizando um ponto de aproximação com a nossa abordagem.

7 Trabalhos Futuros

7.1 Catalogação dos elementos de metadados armazenados nos servidores AulaNet

Para a implementação da arquitetura WebSS, será necessário a catalogação dos elementos de metadados relacionados aos conteúdos educacionais armazenados nos servidores AulaNet [1], permitindo assim a interoperabilidade e reuso dos mesmos em um ambiente de *Web Semântica*.

O RDF usa notação XML como uma sintaxe de codificação para descrição de metadados. Seu principal objetivo é facilitar o intercâmbio de informações, que podem ser interpretadas por máquinas, entre aplicativos via Web, permitindo a automatização do processamento de recursos Web.

O Suporte RDF *Schema* apresenta as primitivas para a criação de ontologias, utilizando as classes de conceitos em uma estrutura hierárquica com múltiplas relações de herança.

O modelo de conhecimento do Protegé-2000 tem classes, instâncias destas classes, propriedades (*slots*), que descreve atributos destas classes e instâncias, e restrições (*facets*), expressando informações adicionais sobre as propriedades. E essa forma de representação se aproxima bastante ao modelo de conhecimento utilizado pelas novas linguagens, como RDF e RDF *Schema*.

O modelo de conhecimento do Protegé-2000 se aproxima bastante ao modelo de conhecimento utilizado pelas novas linguagens, como RDF e RDF *Schema*. Ou seja, tem classes, instâncias destas classes, propriedades (*slots*), que descrevem atributos destas

classes e instâncias, e restrições (*facets*), que expressam informações adicionais sobre as propriedades.

7.2 Geração de ontologias usando Protegé-2000

O Protegé é um editor de ontologias para diversas linguagens desenvolvidas para a Web Semântica, dentre as quais RDF e RDF *Schema* e possibilita estruturar a informação fornecendo assim, o vocabulário necessário para a comunicação entre os agentes de software.

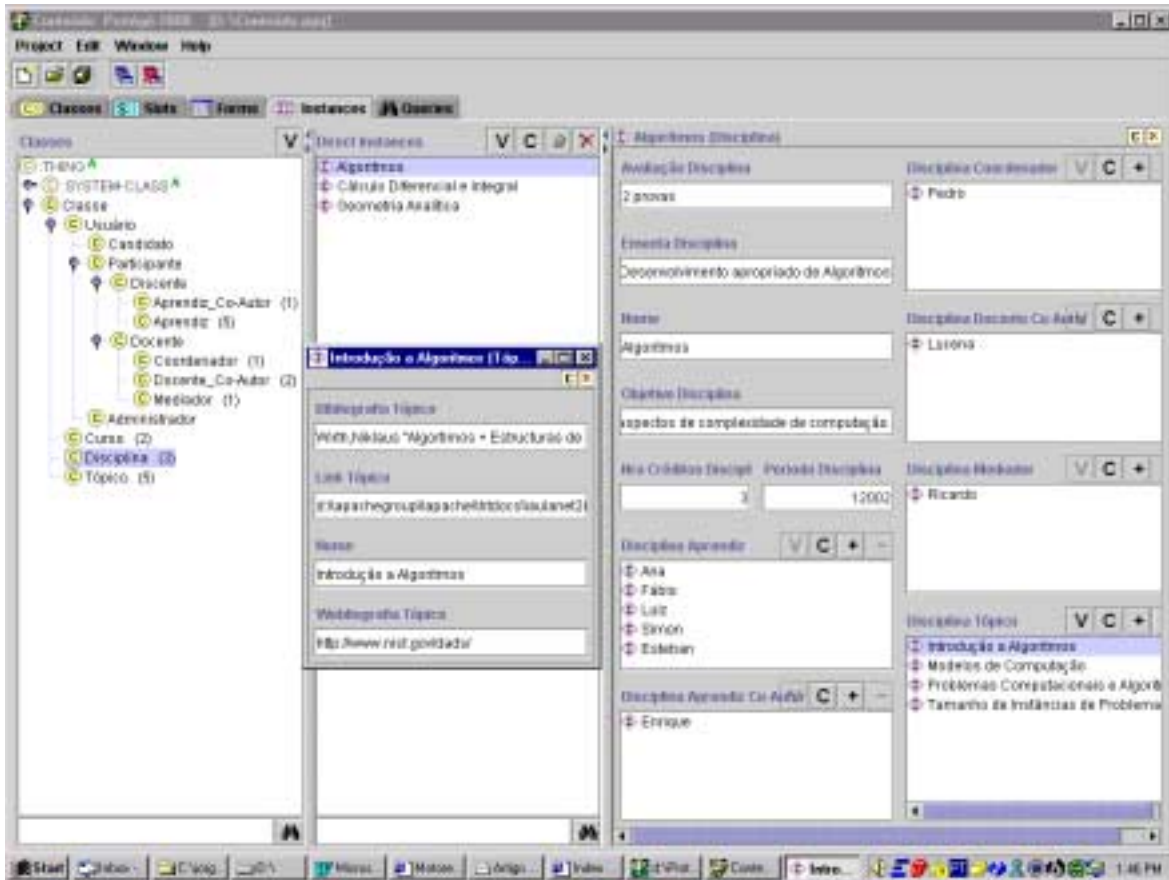


Figura 4. Implementação do Protegé na geração de ontologias de conteúdo educacionais

No lado esquerdo da Figura 4, está representada uma ontologia com a taxonomia de uma determinada classe chamada **curso**. Esta classe está relacionada a outra chamada **disciplina**, onde cada **disciplina** está relacionada a uma classe chamada **tópicos**. Uma **disciplina**, está relacionada também a uma classe chamada usuário, que tem as subclasses *discente* e *docente* que estão associadas aos aprendizes e docentes do ambiente AulaNet [1]. Em nosso caso, *docente* é *Docente_Co-Autor*.

De acordo com a Figura 4, é feita uma instanciação da **disciplina** *Algoritmo*, onde são mostradas todas as suas propriedades e relacionamentos. Por exemplo, a **disciplina** *Algoritmo* tem os **tópicos** *Introdução a Algoritmo*, *Modelos de Computação*, *Problemas*

Computacionais e Algoritmo, Tamanho de Instâncias de Problemas. Cada **tópico**, por sua vez, tem uma *bibliografia* e *Web bibliografia* e um *link* que aponta para o servidor AulaNet, onde se encontra armazenado fisicamente o conteúdo do **tópico**.

8 Considerações Finais

O mercado de *e-Learning* das universidades, no futuro, explodirá e poderá facilmente igualar-se ou superar o mercado de *e-Learning* corporativo. E, na medida em que muda o processo de ensino-aprendizagem dos estudantes, também mudarão suas expectativas em relação à maneira como aprenderão no trabalho.

Os provedores de objetos educacionais proliferarão na Web, as empresas de *e-Learning* fornecerão serviços completos, que misturam conteúdo com gerenciamento do aprendizado, e venderão esses serviços de várias maneiras, incluindo inscrições individuais e corporativas.

Segundo os analistas Trace Urdan e Cornelia Weggen [4], os segmentos-chave do mercado de *e-Learning* são conteúdo, tecnologia e serviços. Com o tempo, universidades, empresas de treinamento *on-line*, portais de aprendizado e outras formas de empresas de *e-Learning*, baseadas na Internet, provavelmente formarão parcerias e outros relacionamentos para abranger todos esses segmentos.

9 Anexos

Muitas empresas do mercado de *e-Learning* vêm investindo no desenvolvimento de um conjunto de padrões para atender ao ambiente de gerenciamento e distribuição do conhecimento. Esses padrões procuram incorporar às ferramentas geradoras, o mesmo sistema de rotulagem de cada componente de um produto de *e-Learning*, de modo que eles sejam padronizados à medida que são criados. Por exemplo, o ambiente de *e-Learning* AulaNet [1] se baseia no padrão IMS [5], de modo que todos os ambientes desse tipo são reconhecidos mundialmente.

Há muitas empresas trabalhando para desenvolver padrões em ambientes de aprendizagem distribuídos. Veja a seguir, alguns dos grupos-chave [4].

- Comitê de CBT da Indústria da Aviação (AICC) <http://www.aicc.org>
- Projeto de Sistemas de Gerenciamento Instrucional da EDUCAUSE (IMS)
<http://www.imsproject.org>
- Aprendizado Distribuído Avançado (ADL) <http://www.adlnet.org>
- Aliança de Redes Remotas de Distribuição e Autoria da Europa (ARIADNE)
<http://ariadne.unil.ch>
- Comitê dos Padrões de Tecnologia do Aprendizado da IEEE (IEEE LTSC)
<http://ltsc.ieee.org>
- Sharable Content Object Reference Model (SCORM)
<http://XML.coverpages.org/scorm.HTML>

Referências Bibliográficas:

- [1] Lucena, C. J. P., Fuks, H., Milidiú, R., Laufer, C., Blois, M., Choren, R., Torres, V., Daflon, L. AulaNet: Helping Teachers to Do Their Homework. In: Multimedia Computer Techniques in Engineering Education Workshop. Graz, Austria: Technische Universitat Graz, 1999.
- [2] Silva, V., Lucena, C. J. P. ContentNet: ContentNet: a framework for the interoperability of educational content using standard IMS. Computers & Education 37 (2001) 273-295 disponível em <http://www.elsevier.com/locate/compedu> acessado em 29/04/2002.
- [3] EURESCOM: MESSAGE: Methodology for Engineering Systems of Software Agents – Deliverable 1, Initial Methodology, Menlo Park CA, 1995.
- [4] Rosenberg, Marc J. e-Learning. S. Paulo: Editora Makron. 2002.
- [5] IMS – Instructional Management Systems. - disponível em <http://www.imsproject.org/specifications.HTML>. acessado em 20/05/2002.
- [6] EOE - Educational Objects Economy - disponível em <http://www.eoe.org/> acessado em 20/05/2002.
- [7] Hypertext Markup Language (HTML). W3C's home page for HTML– disponível em <http://www.w3.org/MarkUp/> acessado em 14/05/2002.
- [8] Standard Generalized Markup Language (SGML) – disponível em <http://www.oasis-open.org/cover/sgml-XML.HTML> acessado em 07/05/2002.
- [9] Resource Description Framework (RDF) – disponível em <http://www.w3.org/RDF/> acessado em 10/06/2002.
- [10] Berners-Lee, Tim & Hendler, James & Lassila, Ora. The Semantic Web. <http://www.sciam.com/2001/0501issue/0501berners-lee.HTML>. Site acessado em 04/04/2002.
- [11] Fortes, Débora. A nova Internet. Artigo publicado na Revista Info-Exame – pg. 64. Abril/2002.
- [12] Hendler, J. ; 2001; Agents and the Semantic Web. IEEE Intelligent Systems, March/April 2001, Vol. 16, No. 2, pp 30-37; Disponível em <http://www.computer.org/intelligent/ex2001/x2toc.htm>.
- [13] IEEE Standard Upper Ontology (SUO) - disponível em <http://ltsc.ieee.org/suo/index.HTML>. acessado em 15/04/2002.

[14] Protegé-2000 - disponível em <http://protege.stanford.edu/download.HTML>. acessado em 10/04/2002.

[15] Fuks, H., Laufer, C., Choren, R., & Blois, M. (1999). Communication, coordination and cooperation in distance education, in: Proceedings of AMCIS'99 - 1999 Americas Conference on Information Systems, Association for Information Systems (AIS) Milwaukee, USA, pp. 130-32.

[16] Ellis, C. A., Gibbs, S. J., Rein, G. L. Groupware: Some Issues and Experiences. Communications of the ACM, v.34, n.1, p.38-58, January 1991.

[17] Fuks, H., Gerosa, M.A. & Lucena, C.J.P. (2002) The Development and Application of Distance Learning on the Internet, The Journal of Open and Distance Learning, Vol. 17, N. 1, ISSN 0268-0513, Fevereiro 2002, pp. 23-38.

[18] AulaNet.- disponível em <http://guiaaulanet.eduweb.com.br> acessado em 05/05/2002.

[19] Sycara, K.: In-Context Information Management through Adaptive Collaboration of Intelligent Agents, in: Klusch, M.: Intelligent Information Agents, Springer-Verlag, 1999.

[20] Iglesias, C. et al.: A Methodological Proposal for Multiagent Systems Development Extending CommonKADS. In M. P. Singh, A. Rao and M. J. Wooldridge, editors, Intelligent Agent (IV) (LNAI 1365), pages 313-326, Springer-Verlag:Berlin Germany, 1998.

[21] Wooldridge, M. et al.: The Gaia Methodology for Agent-Oriented Analysis and Design. Kluwer Academic Publishers, Boston, 1999.

[22] Open Learning Repositories and Metadata Modeling -Dhraief, H. Nejd W. , Wolf B. Wolpers, M. – disponível em <http://www.kbs.uni-hannover.de/Arbeiten/Publikationen/> acessado em 30/05/2002.

[23] Cohen, W.W. et al.: Inductive Specification recovery: Understanding Software by Learning from example behaviours. Automated Software Engineering, 2(2): 107-129, 1995.

[24] Haendchen Filho, A. "XSearch: A Neural Network Based Tool for Components Search in a Distributed Object Environment". Proceedings of 12th. International Conference DEXA'2001, Munich, Germany, 2001. Springer Verlag, Berlin. pp.42 a 51.

[25] Lucena, C.J.P.de, Milidiú, R.L. – Sistemas Multi-Agentes – Papel Virtual Editora – 2001.