

UNIVERSIDADE FEDERAL DE UBERLÂNDIA  
FACULDADE DE CIÊNCIA DA COMPUTAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO



**ADAPTIVE-MOODLE: ADAPTATIVIDADE E  
INTEROPERABILIDADE EM AMBIENTES DE E-LEARNING  
UTILIZANDO TECNOLOGIAS DA WEB SEMÂNTICA**

JOSÉ DOS REIS MOTA

Uberlândia - Minas Gerais

2010

UNIVERSIDADE FEDERAL DE UBERLÂNDIA  
FACULDADE DE CIÊNCIA DA COMPUTAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO



JOSÉ DOS REIS MOTA

**ADAPTIVE-MOODLE: ADAPTATIVIDADE E  
INTEROPERABILIDADE EM AMBIENTES DE E-LEARNING  
UTILIZANDO TECNOLOGIAS DA WEB SEMÂNTICA**

Dissertação de Mestrado apresentada à Faculdade de Ciência da Computação da Universidade Federal de Uberlândia, Minas Gerais, como parte dos requisitos exigidos para obtenção do título de Mestre em Ciência da Computação.

Área de concentração: Inteligência Artificial.

Orientadora:

Prof<sup>ª</sup>. Dr<sup>ª</sup>. Márcia Aparecida Fernandes

Uberlândia, Minas Gerais  
2010

UNIVERSIDADE FEDERAL DE UBERLÂNDIA  
FACULDADE DE CIÊNCIA DA COMPUTAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Os abaixo assinados, por meio deste, certificam que leram e recomendam para a Faculdade de Ciência da Computação a aceitação da dissertação intitulada “**Adaptive-Moodle: Adaptatividade e Interoperabilidade em Ambientes de e-learning utilizando Tecnologias da Web Semântica**” por **José dos Reis Mota** como parte dos requisitos exigidos para a obtenção do título de **Mestre em Ciência da Computação**.

Uberlândia, 9 de Setembro de 2010

Orientadora:

---

Prof<sup>a</sup>. Dr<sup>a</sup>. Márcia Aparecida Fernandes  
Universidade Federal de Uberlândia

Banca Examinadora:

---

Prof. Dr. Alexandre Cardoso  
Universidade Federal de Uberlândia

---

Prof<sup>a</sup>. Dsc. Rosa Maria E. Moreira da Costa  
Universidade Estadual do Rio de Janeiro

UNIVERSIDADE FEDERAL DE UBERLÂNDIA  
FACULDADE DE CIÊNCIA DA COMPUTAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Data: Setembro de 2010

Autor: **José dos Reis Mota**  
Título: **Adaptive-Moodle: Adaptatividade e Interoperabilidade em Ambientes de e-learning utilizando Tecnologias da Web Semântica**  
Faculdade: **Faculdade de Ciência da Computação**  
Grau: **Mestrado**

Fica garantido à Universidade Federal de Uberlândia o direito de circulação e impressão de cópias deste documento para propósitos exclusivamente acadêmicos, desde que o autor seja devidamente informado.

---

Autor

O AUTOR RESERVA PARA SI QUALQUER OUTRO DIREITO DE PUBLICAÇÃO DESTE DOCUMENTO, NÃO PODENDO O MESMO SER IMPRESSO OU REPRODUZIDO, SEJA NA TOTALIDADE OU EM PARTES, SEM A PERMISSÃO ESCRITA DO AUTOR.

*Dedico este trabalho à minha família e a todos que contribuíram para sua realização.*

# Agradecimentos

À Professora Márcia Aparecida Fernandes, pelo apoio, pela sabedoria, pela oportunidade de aprender e pela orientação no decorrer deste trabalho.

Ao Professor Carlos Roberto Lopes, pelas preciosas contribuições durante as discussões em torno dos temas deste trabalho.

Aos estudantes de graduação Lucas e Romualdo, pela colaboração na construção de um curso para o ambiente Moodle e na realização dos testes.

Aos professores e funcionários da FACOM, pelos ensinamentos e pela disponibilidade.

À minha família, pela compreensão e apoio em todos os momentos de minha vida.

Aos amigos e a todos aqueles que, de alguma maneira, me acompanharam nesta trajetória.

*“The logic of education systems should be reversed so that it is the system that conforms to the learner, rather than the learner to the system.*

*This is the essence of personalisation.”*

*(Hannah Green, Keri Facer, Tim Rudd, Patrick Dillon, Peter Humphreys.*

*Personalisation and Digital Technologies)*

# Resumo

Sistemas adaptativos no domínio de *e-learning* são aqueles capazes de adaptar o conteúdo e o modo de apresentação de acordo com o perfil de um determinado usuário ou grupo de usuários. Esses sistemas, geralmente, mantêm-se no contexto de pesquisas acadêmicas e são pouco utilizados em situações reais. Por outro lado, os ambientes de *e-learning* que apresentam uma plataforma completa de recursos e ferramentas disponíveis são largamente utilizados pelas instituições de ensino, mas não são adaptativos. Neste trabalho, propõe-se uma forma de integrar esses dois tipos de sistemas, com o objetivo de aproveitar as vantagens de ambos, ou seja, a adaptatividade de um, somado à abrangência de utilização de outro. Com base nessa proposta, foi projetada a integração entre um sistema adaptativo inteligente ao ambiente Moodle e implementou-se um agente que realiza essa integração, sem impacto na estrutura de cursos existente no ambiente. Além disso, projetou-se e implementou-se um sistema adaptativo baseado em ontologias e em padrões para *e-learning*. Com isso, espera-se contribuir para tornar os ambientes de *e-learning* mais adequados a propiciar uma aprendizagem significativa, ao considerar as necessidades específicas de cada estudante, assim como facilitar a criação de cursos ao possibilitar o compartilhamento de recursos de aprendizagem entre diferentes sistemas, por meio de tecnologias da web semântica.

**Palavras chave:** adaptatividade, e-learning, interoperabilidade, web semântica, inteligência artificial

# Abstract

Adaptive systems in the field of e-learning are those capable of adapting the content and mode of presentation according to the profile of a particular user or user group. These systems generally remain in the context of academic research and are rarely used in real situations. Moreover, the e-learning environments that present a complete platform of tools and resources available are largely used by educational institutions, but are not adaptive. We propose to integrate these two types of systems, aiming to take advantage of both - adaptivity coupled with the scope of use of e-learning environments. Therefore, we describe the implementation of an agent that integrates an intelligent adaptive system with the e-learning environment Moodle. Furthermore, we design and implement an adaptive system based on ontologies and e-learning standards. We hope to contribute to making e-learning environments best suited to provide meaningful learning, as well as to facilitate the creation of courses, to enable the sharing of learning resources between different systems through semantic web technologies.

**Keywords:** adaptivity, e-learning, interoperability, semantic web, artificial intelligence

# Sumário

|                                                                                                |            |
|------------------------------------------------------------------------------------------------|------------|
| <b>Lista de Figuras</b>                                                                        | <b>xi</b>  |
| <b>Lista de Tabelas</b>                                                                        | <b>xii</b> |
| <b>1 Introdução</b>                                                                            | <b>13</b>  |
| 1.1 Considerações Iniciais . . . . .                                                           | 13         |
| 1.2 Objetivos . . . . .                                                                        | 14         |
| 1.3 Organização da Dissertação . . . . .                                                       | 15         |
| <b>2 Sistemas Adaptativos e Ambientes de <i>e-learning</i></b>                                 | <b>16</b>  |
| 2.1 Sistemas Adaptativos . . . . .                                                             | 17         |
| 2.1.1 Sistemas adaptativos: conceitos e componentes . . . . .                                  | 17         |
| 2.1.2 Sistemas adaptativos: exemplos . . . . .                                                 | 18         |
| 2.1.3 Sistemas adaptativos baseados em tecnologias da web semântica . .                        | 22         |
| 2.2 Ambientes de <i>e-learning</i> . . . . .                                                   | 26         |
| 2.2.1 Moodle . . . . .                                                                         | 27         |
| 2.3 Objetos de Aprendizagem e Padrões para <i>e-learning</i> . . . . .                         | 29         |
| 2.3.1 Learning Object Metadata (LOM) . . . . .                                                 | 30         |
| 2.3.2 Shareable Content Object Reference Model (SCORM) . . . . .                               | 32         |
| 2.4 Estilos de Aprendizagem . . . . .                                                          | 33         |
| 2.5 Considerações . . . . .                                                                    | 36         |
| <b>3 Web Semântica</b>                                                                         | <b>37</b>  |
| 3.1 Web Semântica: Conceitos e Objetivos . . . . .                                             | 37         |
| 3.2 Camadas da Web Semântica . . . . .                                                         | 39         |
| 3.2.1 Identificação e representação de recursos: URI/IRI . . . . .                             | 41         |
| 3.2.2 Representação sintática de recursos: Extensible Markup Language<br>(XML) . . . . .       | 42         |
| 3.2.3 Representação semântica de recursos: Resource Description Fra-<br>mework (RDF) . . . . . | 43         |
| 3.2.4 Representação do conhecimento: Ontologias . . . . .                                      | 44         |
| 3.3 Considerações . . . . .                                                                    | 55         |

|          |                                                                                               |            |
|----------|-----------------------------------------------------------------------------------------------|------------|
| <b>4</b> | <b>Adaptive-Moodle: Adaptatividade e Interoperabilidade em Ambientes de <i>e-learning</i></b> | <b>56</b>  |
| 4.1      | Arquitetura do Adaptive-Moodle . . . . .                                                      | 57         |
| 4.2      | Integração entre Ambientes de <i>e-learning</i> e Sistemas Adaptativos . . . . .              | 60         |
| 4.2.1    | Moodle e Sistemas Adaptativos . . . . .                                                       | 60         |
| 4.2.2    | Agente de Integração . . . . .                                                                | 63         |
| 4.3      | Sistema Adaptativo Baseado em Ontologias . . . . .                                            | 68         |
| 4.3.1    | Visão geral do processo de adaptação . . . . .                                                | 68         |
| 4.3.2    | Ontologias . . . . .                                                                          | 71         |
| 4.3.3    | Arquitetura do Sistema Adaptativo . . . . .                                                   | 74         |
| 4.3.4    | Regras de Adaptação . . . . .                                                                 | 77         |
| 4.4      | Testes e Resultados . . . . .                                                                 | 80         |
| 4.5      | Considerações . . . . .                                                                       | 83         |
| <b>5</b> | <b>Conclusões e Trabalhos Futuros</b>                                                         | <b>85</b>  |
| 5.1      | Publicações . . . . .                                                                         | 87         |
|          | <b>Referências Bibliográficas</b>                                                             | <b>89</b>  |
| <b>A</b> | <b>Ontologia LOM</b>                                                                          | <b>95</b>  |
| <b>B</b> | <b>Ontologia do Domínio</b>                                                                   | <b>105</b> |
| <b>C</b> | <b>Ontologia do Estudante</b>                                                                 | <b>111</b> |
| <b>D</b> | <b>Casos de testes baseando-se na seleção de uma organização no Pacote SCORM</b>              | <b>116</b> |
| <b>E</b> | <b>Casos de testes baseando-se na seleção dos objetos de aprendizagem</b>                     | <b>120</b> |

# Lista de Figuras

|      |                                                                                                            |     |
|------|------------------------------------------------------------------------------------------------------------|-----|
| 2.1  | Arquitetura do ActiveMath . . . . .                                                                        | 19  |
| 2.2  | Arquitetura do AHA . . . . .                                                                               | 20  |
| 2.3  | Arquitetura do SIMEDUC . . . . .                                                                           | 21  |
| 2.4  | Arquitetura do AdaptWeb . . . . .                                                                          | 23  |
| 2.5  | Arquitetura do sistema proposto por [Neto 2006] . . . . .                                                  | 24  |
| 2.6  | Hierarquia de elementos do padrão LOM . . . . .                                                            | 32  |
| 2.7  | Exemplo de resultado do Índice de Estilos de Aprendizagem proposto por<br>[Felder e Soloman 2010]. . . . . | 35  |
| 3.1  | Camadas da web semântica . . . . .                                                                         | 39  |
| 3.2  | Representação gráfica de uma tripla RDF . . . . .                                                          | 44  |
| 4.1  | Arquitetura do Adaptive-Moodle . . . . .                                                                   | 58  |
| 4.2  | Classes do Sistema Adaptativo . . . . .                                                                    | 59  |
| 4.3  | Atividades - Agente de Integração . . . . .                                                                | 65  |
| 4.4  | Visão geral do processo de adaptação . . . . .                                                             | 69  |
| 4.5  | Hierarquia de classes da Ontologia LOM . . . . .                                                           | 72  |
| 4.6  | Classes e relacionamentos da Ontologia de Domínio . . . . .                                                | 73  |
| 4.7  | Classes e relacionamentos da Ontologia do Estudante . . . . .                                              | 74  |
| 4.8  | Arquitetura do Sistema Adaptativo . . . . .                                                                | 75  |
| 4.9  | Atividades do Agente de Adaptação . . . . .                                                                | 76  |
| 4.10 | Regras para seleção de objetos de aprendizagem . . . . .                                                   | 78  |
| 4.11 | Estrutura do Pacote SCORM com várias organizações . . . . .                                                | 81  |
| 4.12 | Curso original e conteúdo das organizações . . . . .                                                       | 82  |
| 4.13 | Curso adaptado e conteúdo da organização selecionada . . . . .                                             | 83  |
| E.1  | Anotações em um objeto de aprendizagem utilizando a ferramenta Reload<br>Editor . . . . .                  | 121 |

# Lista de Tabelas

|      |                                                                                             |     |
|------|---------------------------------------------------------------------------------------------|-----|
| 2.1  | Estatísticas de utilização do Moodle em 22/07/2010 . . . . .                                | 28  |
| 4.1  | Dados relacionados ao Modelo de Domínio. . . . .                                            | 61  |
| 4.2  | Dados relacionados às atividades a serem avaliadas. . . . .                                 | 62  |
| 4.3  | Dados relacionados ao Modelo do Estudante. . . . .                                          | 62  |
| 4.4  | Tabelas acrescentadas à base de dados do Moodle referentes aos cursos adaptados. . . . .    | 67  |
| 4.5  | Tabelas acrescentadas à base de dados do Moodle para ampliar o Modelo do Estudante. . . . . | 68  |
| D.1  | Seção e organização selecionada após a matrícula dos estudantes . . . . .                   | 117 |
| D.2  | Variáveis utilizadas para avaliação do comportamento . . . . .                              | 117 |
| D.3  | Valores obtidos para o comportamento após a aplicação de regras <i>fuzzy</i> . . . . .      | 117 |
| D.4  | Valores referentes à avaliação do comportamento . . . . .                                   | 117 |
| D.5  | Variáveis utilizadas para avaliação do conhecimento . . . . .                               | 118 |
| D.6  | Valores obtidos para o conhecimento após a aplicação de regras <i>fuzzy</i> . . . . .       | 118 |
| D.7  | Valores referentes à avaliação do conhecimento . . . . .                                    | 118 |
| D.8  | Valores referentes à avaliação do desempenho . . . . .                                      | 119 |
| D.9  | Resultados finais . . . . .                                                                 | 119 |
| D.10 | Seção e organização selecionada após os resultados de avaliação . . . . .                   | 119 |
| E.1  | Testes: estudantes e estilos de aprendizagem . . . . .                                      | 120 |
| E.2  | Testes: objetos de aprendizagem . . . . .                                                   | 121 |

# Capítulo 1

## Introdução

### 1.1 Considerações Iniciais

Diversas pesquisas, como disposto em [Brusilovsky e Peylo 2003], enfatizam a necessidade do desenvolvimento de sistemas educativos adaptativos, capazes de se adequar ao perfil de cada estudante, e interoperáveis, que compartilham recursos com outros ambientes. Tais características são importantes em ambientes de *e-learning*, pela possibilidade de atenderem a uma grande demanda de estudantes e, assim, haver a necessidade de se adequarem a diferentes perfis de usuários e contextos de aprendizagem.

Atualmente, estão disponíveis diversos ambientes que são utilizados pelas instituições de ensino, inclusive algumas opções *open-source*, como [Moodle 2010d], [Teleduc 2010], [Claroline 2010]. Tais sistemas apresentam vários recursos e ferramentas, como fóruns, *chats*, ferramentas de autoria e de avaliação. No entanto, ainda são restritos em termos de adaptatividade.

Em relação à interoperabilidade, existem alguns padrões já consolidados para a produção e distribuição de conteúdos educacionais, como o *Learning Object Metadata* (LOM) [Committee 2010], o *IMS Learning Design* [Consortium 2010a] e o *Sharable Content Object Reference Model* (SCORM) [Learning 2010b], visando ao reuso de tais conteúdos em diversos contextos e ambientes. Entretanto, ainda são poucos os ambientes que implementam os padrões citados e, mesmo quando implementam, os materiais produzidos para um determinado sistema geralmente precisam de ajustes para serem utilizados em outro ambiente, ou seja, não podem simplesmente ser compartilhados.

Por outro lado, os sistemas adaptativos inteligentes existentes mantêm-se, em sua maioria, no contexto de pesquisas acadêmicas e não são aplicados a situações reais. Uma das causas disso é que, ao propor um sistema que possibilite adaptatividade, normalmente os pesquisadores não se preocupam com a integração com outros sistemas e implementam uma plataforma de *e-learning* completa, com todos os recursos necessários. Assim, as instituições que já disponibilizam cursos, para se beneficiarem dessa característica, teriam

que migrar de uma solução que utilizam para um novo sistema, o que, na maioria das vezes, não é viável em relação a custo e tempo.

Assim, nesse trabalho, foi projetada uma estrutura genérica, baseada nos componentes existentes nos sistemas adaptativos, para explorar as possibilidades de como esses sistemas podem se integrar a ambientes de *e-learning*, acrescentando-lhes adaptatividade e, baseado nesta estrutura, implementou-se um agente que realiza a integração de um sistema adaptativo inteligente com o ambiente de *e-learning* Moodle. Além disso, foram modeladas as ontologias e, a partir delas, definiu-se um ambiente adaptativo que também foi integrado ao ambiente Moodle por meio do agente de integração.

## 1.2 Objetivos

Este trabalho apresentou dois objetivos principais. O primeiro foi projetar uma estrutura genérica, baseada nos componentes de um sistema adaptativo, para explorar as possibilidades de como os sistemas adaptativos podem se integrar a ambientes de *e-learning*, acrescentando-lhes adaptatividade e, baseado nesta estrutura, desenvolver um agente que realiza a integração de um sistema adaptativo inteligente com um ambiente de *e-learning*. Para esta integração, em particular, utilizou-se o ambiente de *e-learning* Moodle, por sua abrangência de utilização e por suportar padrões para *e-learning*, tais como SCORM [Learning 2010b], que por sua vez abrange o padrão LOM [Committee 2010]. Assim, esse ambiente foi integrado ao *Sistema Inteligente Multiagente para Educação à Distância* (SI-MEDUC), descrito em [Dorça 2004], que se baseia na tecnologia de sistemas multiagentes para criar um ambiente adaptativo.

O segundo objetivo foi projetar e implementar um sistema adaptativo baseado em ontologias que pudesse ser integrado a ambientes de *e-learning*. O agente de integração, implementado para atingir o primeiro objetivo, foi utilizado também para integrar este sistema adaptativo ao ambiente de *e-learning* Moodle. Para o desenvolvimento desse sistema foram modeladas e implementadas ontologias, baseando-se em padrões consolidados em *e-learning*, na teoria sobre estilos de aprendizagem, e nas tecnologias indicadas como padrão pela World Wide Web Consortium (W3C) na área de web semântica [Consortium 2010b]. As ontologias contemplam a representação dos objetos de aprendizagem no padrão LOM, assim como a representação do Modelo de Domínio e do Modelo do Estudante, de forma a possibilitar, além da adaptatividade, a interoperabilidade entre ambientes de *e-learning*.

Espera-se ainda, com este trabalho, contribuir na área de *e-learning*, especialmente em modelagem de estudante e adaptatividade, assim como na área de web semântica, ao propor ontologias e regras baseadas em padrões consolidados para *e-learning* e em teorias sobre estilos de aprendizagem. Tanto o processo de integração quanto o de adaptação serão validados por meio do desenvolvimento de um curso no ambiente Moodle.

## 1.3 Organização da Dissertação

Esta dissertação está estruturada da seguinte forma.

No Capítulo 2 serão apresentados os conceitos básicos relacionados a ambientes de *e-learning* e sistemas adaptativos, assim como exemplos desses sistemas. O ambiente de *e-learning* selecionado para a integração com sistemas adaptativos é descrito, buscando extrair as informações já existentes que facilitem o processo de integração. Nesse intuito, serão apresentados ainda os padrões para *e-learning* utilizados no desenvolvimento do sistema de integração e do sistema adaptativo propostos, assim como a teoria de estilos de aprendizagem proposta por [Felder e Silverman 1988], que foi utilizada para definição de regras para adaptação.

O Capítulo 3 descreve a web semântica, um breve histórico de como surgiu e o estado atual de seu desenvolvimento, assim como as principais tecnologias pertencentes às suas camadas e como elas se relacionam ao trabalho proposto. Um foco especial será dado às ontologias, já que serão a base para o desenvolvimento do sistema adaptativo. Assim, a linguagem para definição de ontologias para a web semântica é apresentada, assim como as ferramentas utilizadas na modelagem e manipulação dessas ontologias.

No capítulo Capítulo 4 são descritos os sistemas desenvolvidos. Como são dois processos complementares, mas diferentes, sendo que o primeiro consiste na integração entre sistemas adaptativos e ambientes de *e-learning* e o segundo consiste no desenvolvimento de um sistema adaptativo baseado em tecnologias da web semântica, optou-se, a partir da definição dos requisitos e da arquitetura para integração dos módulos, por descrever os trabalhos em duas etapas. Em primeiro lugar, é descrito o processo de integração; a estrutura e as atividades necessárias para esse processo são detalhadas e o agente responsável pela execução das atividades é apresentado. Na segunda etapa, são descritas as atividades envolvidas no processo de adaptação, em que são apresentadas as ontologias, suas classes, propriedades, relacionamentos e as regras de adaptação. Nesse momento, descreve-se ainda a forma como serão geradas as instâncias das ontologias e como colaboram para criar um ambiente adaptativo. Por fim, são apresentados alguns testes realizados com os sistemas.

No Capítulo 5 são apresentadas as conclusões do presente trabalho e perspectivas para trabalhos futuros.

## Capítulo 2

# Sistemas Adaptativos e Ambientes de *e-learning*

Os ambientes de *e-learning* provêm uma infraestrutura para educação à distância que consiste, basicamente, em um conjunto de ferramentas de autoria, de comunicação, de apresentação de conteúdo e de avaliação. A maioria dos ambientes existentes, como [Modle 2010d], [Teleduc 2010] e [Claroline 2010], disponibilizam um ambiente completo e estável para ministrar cursos à distância e, por isso, são muito utilizados pelas instituições de ensino. Já os sistemas adaptativos são aqueles capazes de adaptar, de alguma forma, o conteúdo a ser apresentado pelo usuário de acordo com seu perfil, a partir de determinadas regras. Essa característica é importante, já que uma das vantagens da educação à distância é justamente a possibilidade de atender a uma grande demanda de estudantes; no entanto, pela dificuldade de implementar tais sistemas, já que muitas vezes incorporam técnicas de inteligência artificial e, ainda, impõem exigências na estruturação dos conteúdos, dificultando essa tarefa, eles são restritos a domínios específicos e a pesquisas acadêmicas, e não são utilizadas em larga escala.

Assim, na seção 2.1 deste capítulo são descritas as características dos sistemas adaptativos, a partir de alguns exemplos. Será descrito, de forma mais detalhada, o sistema adaptativo *e-learning* selecionado para integração, o SIMEDUC [Dorça 2004]. Na seção 2.2, serão apresentados os conceitos básicos relacionados aos ambientes de *e-learning*. A subseção 2.2.1 descreverá o ambiente utilizado para integração com sistemas adaptativos, o Moodle, apontando suas características básicas.

Os sistemas propostos baseiam-se em padrões para *e-learning*. Assim, na seção 2.3 deste capítulo serão apresentados o conceito de objeto de aprendizagem e os padrões utilizados para construção desses objetos e definição das regras de adaptação, mais especificamente, serão apresentados os padrões LOM e SCORM. Na seção 2.4 será apresentada a teoria educacional sobre estilos de aprendizagem, com ênfase naquela proposta por [Felder e Silverman 1988], já que ela também foi utilizada na definição das regras de adaptação.

Por fim, na seção 2.5 são expostas as considerações finais do presente capítulo.

## 2.1 Sistemas Adaptativos

Nesta seção serão apresentados os principais conceitos e componentes relacionados aos sistemas adaptativos. Para mostrar a representação desses componentes, serão apresentados alguns exemplos disponíveis desses sistemas, em especial o sistema selecionado para integração com um ambiente de *e-learning*, o Sistema Inteligente Multiagente para Educação à Distância (SIMEDUC), descrito em [Dorça 2004]. Serão apresentados ainda alguns trabalhos baseados em tecnologias da web semântica para prover adaptatividade, já que o sistema adaptativo proposto baseia-se nessas tecnologias.

### 2.1.1 Sistemas adaptativos: conceitos e componentes

Sistemas adaptativos no domínio de *e-learning* são aqueles capazes de adaptar o conteúdo e o modo de apresentação de acordo com o perfil de um determinado usuário ou grupo de usuários. Assim, usuários com diferentes objetivos, preferências e conhecimentos podem acessar diferentes conteúdos, com diferentes formatos de apresentação. [Brusilovsky e Peylo 2003] classificam os sistemas educacionais adaptativos e inteligentes baseados na web como aqueles que “atentam-se para a adaptatividade pela construção de um modelo de objetivos, preferências e conhecimento de cada estudante e uso deste modelo na interação com o estudante no sentido de adaptar o conteúdo de acordo com suas necessidades. Eles também procuram ser mais inteligentes ao incorporar e executar algumas atividades tradicionalmente executados por um professor humano”.

Ainda de acordo com [Brusilovsky e Peylo 2003], Apresentação Adaptativa e Suporte à Navegação Adaptativa são as duas maiores tecnologias exploradas pelos sistemas de hipertexto e hipermídia adaptativa. O objetivo da tecnologia de Apresentação Adaptativa é adaptar o conteúdo presente em cada página hipermídia de acordo com os objetivos do estudante, seu conhecimento, e outras informações armazenadas no modelo do estudante. Já o Suporte à Navegação Adaptativa tem como objetivo auxiliar o estudante na orientação e navegação por meio da mudança da aparência dos *links* visíveis. Por exemplo, um sistema de hipermídia pode classificar, realçar ou ocultar parcialmente os *links* da página corrente para tornar mais fácil a escolha do próximo passo. A navegação adaptativa auxilia os estudantes a encontrarem um caminho ótimo pelo material de aprendizagem, assim como o sequenciamento de currículo.

[Oppermann et al. 1997] distingue os conceitos de adaptabilidade e adaptatividade. *Adaptabilidade* é a capacidade do sistema em permitir ao usuário alterar certos parâmetros, podendo adaptar as funcionalidades, ao ativar, desativar ou configurar recursos, ou adaptar a interface, ao modificar a forma de acesso aos recursos. Já *adaptatividade* é a capacidade do sistema em se adaptar ao usuário automaticamente, baseando-se em inferências sobre suas necessidades. Nesse trabalho, portanto, utiliza-se o termo adaptatividade, já que a proposta é que o sistema realize, dinamicamente, a adaptação do

conteúdo, por meio de regras de adaptação que se baseiam no estilo de aprendizagem e no desempenho do estudante durante o curso.

[Wu 2002] mostra que os sistemas adaptativos possuem, conceitualmente, três componentes básicos, que não necessariamente estão representados ou implementados separadamente. O primeiro componente é o *Modelo de Domínio*, que representa o conteúdo apresentado pelo sistema, tanto os itens de informação, como os relacionamentos entre eles. O segundo componente é o *Modelo do Usuário*, contendo as características relevantes para a adaptação, tais como preferências, conhecimento, objetivos e histórico de navegação. O terceiro é o *Modelo de Adaptação*, que mantém atualizado o perfil do usuário, de acordo com a observação de seu comportamento, e define como realizar a adaptação, utilizando informações dos demais componentes.

De acordo com essa visão conceitual, um modelo clássico de adaptação no domínio de *e-learning* deve conter o Modelo de Domínio, que propõe a estruturação dos cursos e dos conteúdos, o Modelo do Estudante, que determina o perfil do estudante e é atualizado constantemente, à medida que interage com o sistema, e o Modelo de Adaptação, que normalmente incorpora técnicas de inteligência artificial ao definir regras para adaptação. ActiveMath [ActiveMath 2010], [Melis et al. 2001], AHA (Adaptive Hypermedia for All) [Bra et al. 2006], [AHA 2010] e SIMEDUC [Dorça 2004] são exemplos de sistemas adaptativos que contêm estes componentes e que serão apresentados na seção 2.1.2.

### 2.1.2 Sistemas adaptativos: exemplos

No ActiveMath, sistema dedicado ao ensino de Matemática, a representação do conteúdo baseia-se na linguagem XML (*eXtensible Markup Language*) [W3C 2010b]. Para cada usuário, o conteúdo apropriado é recuperado de uma base de conhecimento e o curso é gerado individualmente, de acordo com regras pedagógicas, objetivos, preferências do usuário, competências e contexto de aprendizagem. A arquitetura do sistema ActiveMath, descrita na Figura 2.1, é constituída por componentes primários, que são utilizados para configuração e gerenciamento do sistema e para troca de informações entre os componentes, e por componentes fortemente acoplados, a parte central da aplicação, que pode ser associada aos componentes básicos de um sistema adaptativo da seguinte forma: na arquitetura percebe-se de forma explícita o componente *Modelo do Estudante*; o *Modelo de Domínio* relaciona-se com os componentes *Conteúdo do Banco de Dados*, que permite acesso a OMDoc (*Open Mathematical Documents*) [OMDoc 2010], e com o *Gerenciador de Conteúdo*, que representa o conteúdo e seus metadados em forma de itens individuais de aprendizagem; o componente *Apresentação* associa o *Modelo de Domínio* ao *Modelo de Adaptação*, já que é responsável por definir e apresentar o conteúdo em um determinado formato, como por exemplo HTML, Latex ou PDF.

Outro exemplo de sistema adaptativo é o AHA (*Adaptive Hypermedia for All*), em que

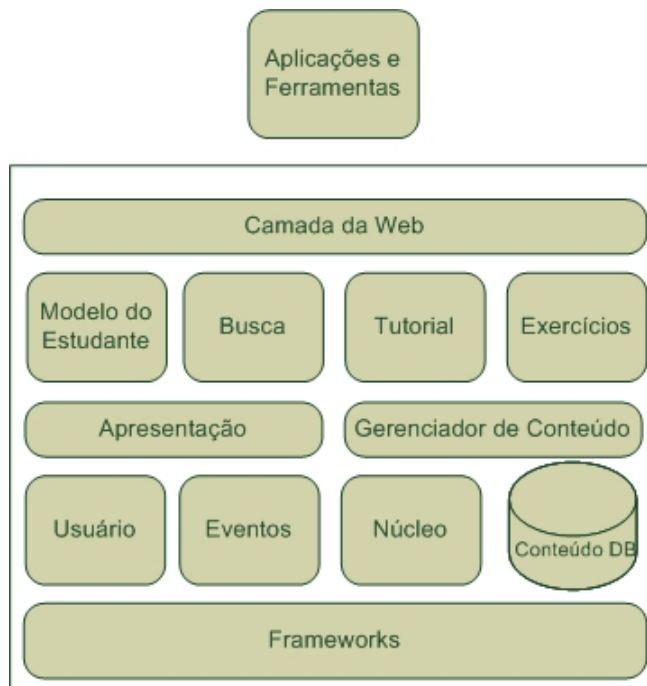


Figura 2.1: Arquitetura do ActiveMath

a adaptatividade é determinada por relacionamentos de pré-requisito e condições definidas por expressões booleanas. Nesse sistema, alguns modelos conceituais, navegacionais e adaptativos são definidos, como:

- Um modelo do estudante baseado em conceitos que contém atributos. A cada vez que processa-se a interação do usuário com uma página, o modelo do estudante é atualizado.
- Mudança adaptativa: a adequabilidade das páginas ou arquivos é determinada por requisitos, que podem ser relacionamentos comuns de pré-requisito ou outras condições definidas por uma expressão booleana. Quando uma página é gerada, os *links* são marcados como condicionais e mostrados de forma diferente dependendo de sua adequabilidade (não-visitados, se a expressão é verdadeira; visitado, se a expressão não tem valor, e sem *link*, se a expressão é falsa).
- Inclusão condicional de fragmentos: usado para incluir explicações que são pré-requisito ou qualquer outra parte de conteúdo, inclusive objetos externos.

A Figura 2.2 apresenta os principais componentes deste sistema. O componente *Processamento* pode utilizar informação local e, potencialmente, também páginas residentes em outros servidores na web. O *Modelo de Domínio* e o *Modelo de Adaptação* representam a estrutura conceitual e as regras de adaptação. Esta informação é recuperada no momento em que o usuário acessa o sistema. Quando o usuário está interagindo com a aplicação, o *Modelo do Usuário* é usado e constantemente atualizado. Os três modelos - domínio, adaptação e usuário - são usados para decidir quais páginas ou arquivos

recuperar, em determinada requisição, e quais objetos incluir de forma condicional na página.

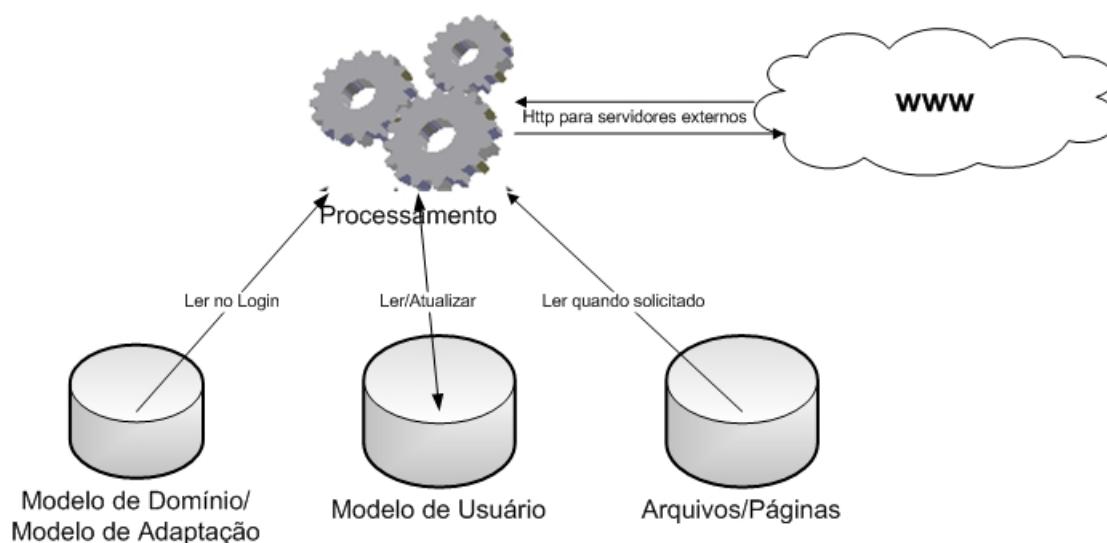


Figura 2.2: Arquitetura do AHA

O sistema adaptativo SIMEDUC [Dorça 2004] apresenta uma arquitetura baseada em sistemas multiagentes. As interações do estudante com o sistema são feitas por meio de um *Learning Management System* (LMS - Sistema Gerenciador de Cursos), que implementa ferramentas e recursos de interface necessários ao ambiente. As informações resultantes da interação do estudante com o LMS são processadas por um Sistema Inteligente Multiagente (SIM), que fornece adaptatividade e inteligência. As informações resultantes deste processamento são utilizadas pelo LMS, que apresenta conteúdo personalizado ao estudante de acordo com as informações geradas pelo SIM, respeitando suas preferências, nível de conhecimento, dificuldades e problemas de aprendizagem.

Conforme representado na figura 2.3, quatro agentes colaboram na arquitetura. O Agente Assistente monitora o estudante, fornecendo ajuda, acompanhamento e reagindo de acordo com o comportamento do estudante. O Agente de Avaliação é responsável pelo processo de avaliação do conhecimento e do comportamento do estudante e por atualizações do Modelo do Estudante para que este reflita o grau de conhecimento atual. O Modelo do Estudante é armazenado na Base de Dados do Estudante e é responsável por armazenar o perfil de cada estudante, ou seja, de suas capacidades perceptivas e do seu grau de conhecimento alcançado durante a realização do curso. Além disso, a Base de Dados do Estudante também contém os cursos que ele está realizando e todas as atividades apresentadas durante cada sessão de aprendizagem.

O Agente Pedagógico gera e atualiza automaticamente o sequenciamento de conteúdo do estudante de acordo com o seu modelo. O objetivo principal deste agente é diagnosticar problemas de aprendizagem do estudante sobre o domínio. O Agente Especialista, cuja implementação não foi feita no trabalho, seria utilizado pelo Agente de Avaliação para

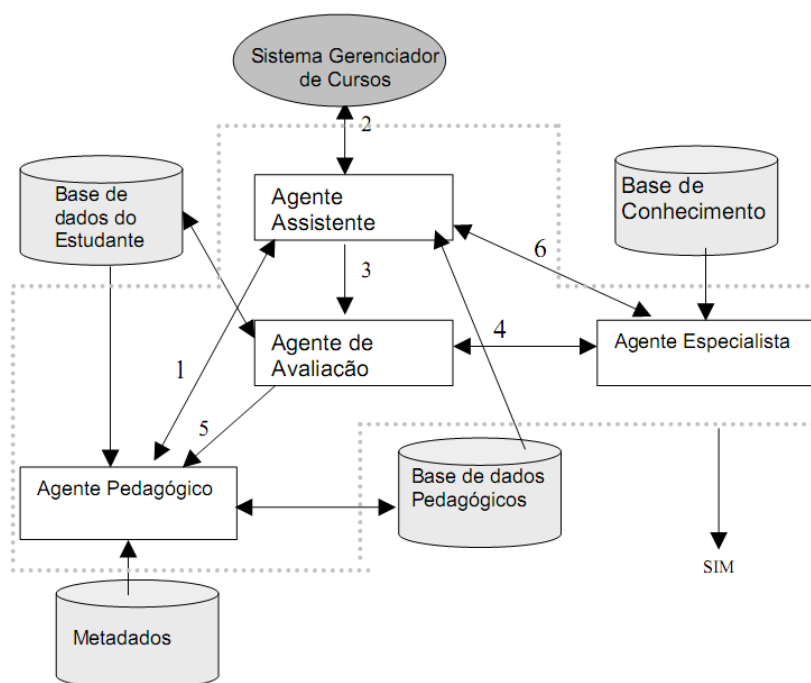


Figura 2.3: Arquitetura do SIMEDUC

resolução de problemas. As respostas do especialista, bem como o procedimento adotado na resolução do problema, poderiam ser comparados com os do estudante, o que permitiria uma atualização do perfil do estudante bem como uma nova estratégia de ensino.

Na figura 2.3 há ainda a representação do fluxo de informações pelo sistema. Antes de iniciar o curso, o Agente Assistente solicita ao Agente Pedagógico a construção de um sequenciamento de conteúdo para o aluno (1). O plano gerado é então armazenado na base de dados pedagógicos. O Agente Pedagógico informa ao Agente Assistente quando o plano estiver pronto. O Agente Assistente recebe informações sobre o comportamento e as respostas do estudante aos testes (2) e as envia para o Agente de Avaliação (3). Para avaliar a resposta do estudante a um exercício, o Agente de Avaliação pode requerer ajuda do Agente Especialista (4). Após realizada a avaliação, o Agente Assistente é informado do resultado e informa o estudante sobre seu desempenho (2). Com o resultado da avaliação do conhecimento e do comportamento do estudante, a Base de Dados do Estudante é atualizada e isto é comunicado ao Agente Pedagógico (5).

Baseado no Modelo do Estudante atualizado, o Agente Pedagógico pode replanejar o respectivo sequenciamento de conteúdo, caso seja necessário, e atualizá-lo na Base de Dados Pedagógicos. Para isto, o Agente Pedagógico acessa a Base de Metadados em busca de conceitos mais apropriados de acordo com o perfil do estudante para compor o plano a ser gerado. O Agente Assistente solicita ao Agente Especialista (6) que busque respostas para as perguntas do estudante sobre o conteúdo sendo estudado. O Agente Especialista, depois de efetuada a busca na Base de Conhecimento, devolve a resposta ao Agente Assistente, para que este, possa apresentar a resposta ao estudante (2). A comunicação

ocorre através da troca de mensagens e é assistida por um programa facilitador. As mensagens obedecem a um formato comum e bem definido, interpretável por todos os agentes do sistema. O agente receptor da mensagem efetua a leitura desta, e de acordo o conteúdo e o seu próprio conhecimento, ele determina as ações a executar, que podem implicar no envio de novas mensagens.

Por propor uma divisão clara entre as tarefas envolvidas na adaptação, realizada pelo SIM, e aquelas executadas pelo ambiente de *e-learning*, representado pelo componente Sistema Gerenciador de Conteúdo, assim como implementar a arquitetura com base no paradigma multiagente, permitindo uma distribuição das atividades em diversos agentes, que podem trabalhar paralelamente, de forma cooperativa, este sistema foi escolhido para integração com o ambiente de *e-learning* Moodle. No entanto, pretende-se mostrar, a partir da estrutura de integração proposta, que outros sistemas adaptativos, ainda que sejam necessárias algumas personalizações, podem se integrar a ambientes de *e-learning*, tanto que um sistema adaptativo baseado em tecnologias da web semântica também foi desenvolvido e integrado ao Moodle utilizando a mesma estrutura.

Em relação aos sistemas adaptativos analisados, observa-se que todos eles possuem os componentes básicos apontados por [Wu 2002]. Entretanto, embora possuam os componentes clássicos de um sistema adaptativo, estes sistemas não permitem a interoperabilidade, pois os cursos não são criados de acordo com padrões adequados para este fim, restringindo-se a um domínio específico, ou seja, são utilizados apenas no próprio sistema para os quais foram criados.

### 2.1.3 Sistemas adaptativos baseados em tecnologias da web semântica

Sistemas baseados em tecnologias da web semântica também têm sido propostos como alternativa para adaptatividade, pois são capazes de capturar o conhecimento de um determinado domínio, permitindo a definição, a formalização e o compartilhamento de conceitos, restrições, instâncias, relacionamentos e axiomas existentes nesse domínio. Os trabalhos desenvolvidos por [Muñoz 2004] e [Neto 2006] são exemplos de sistemas que utilizam ontologias e que se baseiam em metadados associados aos objetos de aprendizagem para promover adaptatividade.

Em [Muñoz 2004] é proposta uma arquitetura de um sistema, denominado *AdaptWeb*, baseado em tecnologias da web semântica para fornecer adaptatividade e obter interoperabilidade, relacionando os metadados existentes no padrão LOM com os recursos da web semântica e modelando as ontologias necessárias à arquitetura proposta. No entanto, seu trabalho não contempla, de modo mais abrangente, a questão da adaptatividade, já que não popula a ontologia de estudantes, nem implementa agentes capazes de utilizá-la.

Como mostra a figura 2.4, são propostas três ontologias que colaboram na arquitetura

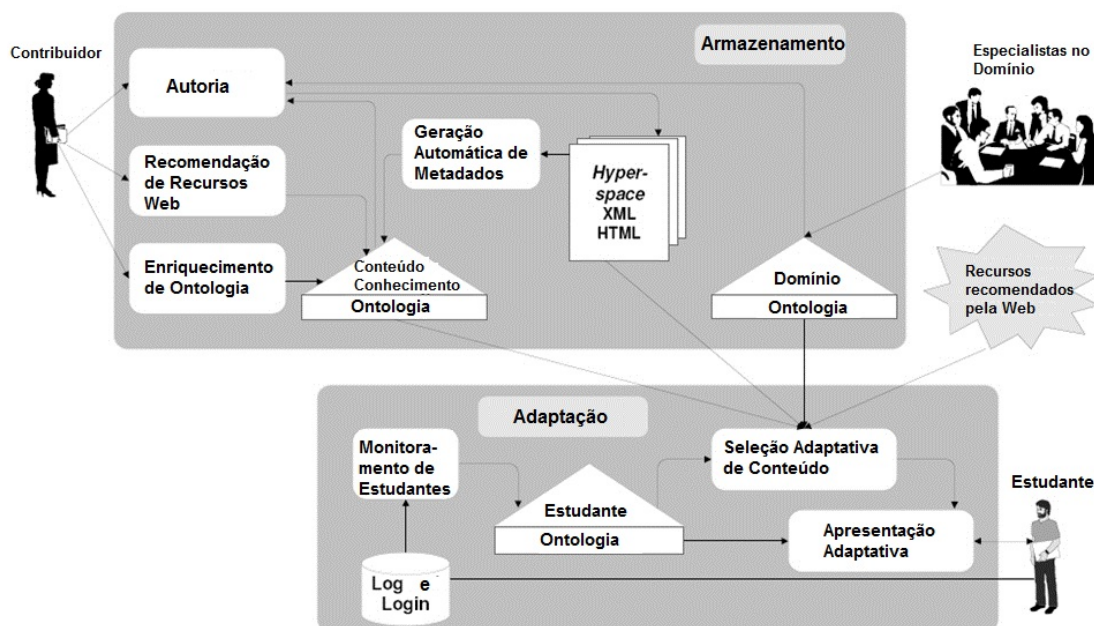


Figura 2.4: Arquitetura do AdaptWeb

do sistema, sendo:

- Ontologia do Domínio, que define a estrutura e relacionamentos do domínio a ser ensinado.
- Ontologia do Estudante, que modela o perfil do estudante e associa o seu conhecimento aos tópicos de conteúdo.
- Ontologia do Conteúdo de Conhecimento, que representa a estrutura de conhecimento sobre os objetos de aprendizagem capaz de prover regras de composição para apresentá-los de acordo com o perfil de um estudante.

Já [Neto 2006] propõe ontologias que contemplam informações sobre o domínio do conhecimento a ser ensinado, teorias de aprendizagem e modelo do estudante, com o objetivo de produzir cursos no padrão SCORM, conforme apresentado na figura 2.5.

As ontologias presentes na figura 2.5 são descritas como:

- Ontologia Pedagógica, construída com o objetivo de representar o conhecimento sobre o processo de aprendizagem. A partir de suas informações, pretende auxiliar as pessoas a construírem e a estruturarem as atividades dos cursos de forma a facilitar a aprendizagem dos alunos e fazer com que eles atinjam os objetivos estabelecidos.
- Ontologia do Estudante, cujo objetivo é definir os termos utilizados para caracterizar os tipos de estudantes, descrevendo-os em duas dimensões: pedagógica e técnica.
- Ontologia de Domínio, cujas informações podem ser divididas em duas dimensões: a primeira, define o mapa conceitual de uma área do conhecimento, com seus conceitos e os relacionamentos entre eles; a segunda, representa os objetivos, atividades e objetos de aprendizagem utilizados na construção dos cursos.

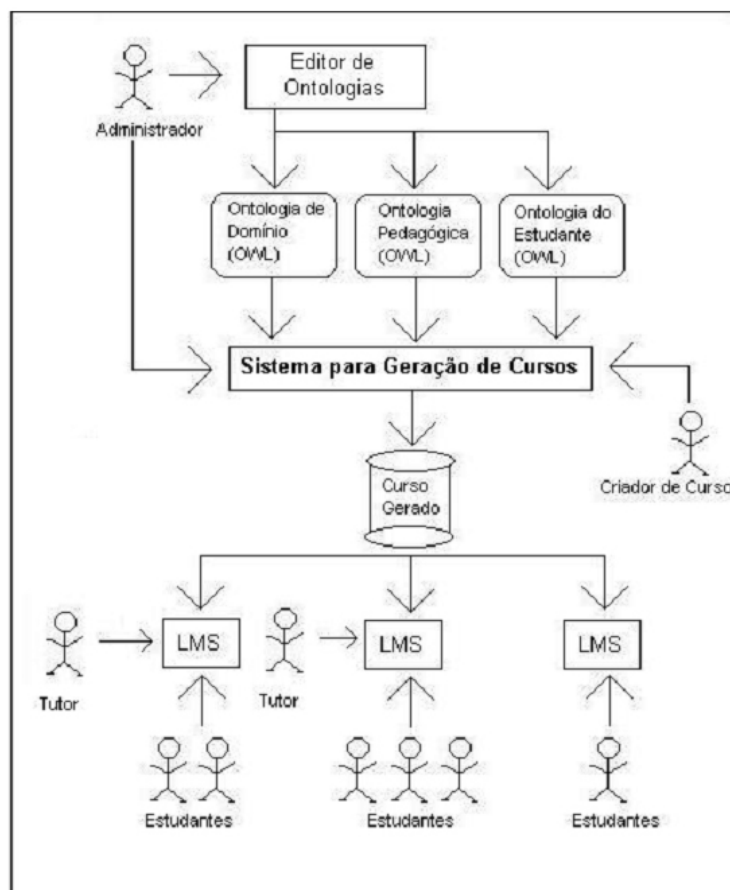


Figura 2.5: Arquitetura do sistema proposto por [Neto 2006]

Percebe-se ainda, na figura 2.5, que nessa proposta a adaptatividade não ocorre no próprio ambiente de *e-learning*, mas como uma etapa prévia, no momento da criação do curso. Posteriormente, os cursos adaptados, como se baseiam no padrão SCORM, podem ser importados para ambientes de *e-learning* que suportem o padrão.

Como pode-se notar, há semelhanças entre as ontologias nos sistemas propostos por [Muñoz 2004] e [Neto 2006], visto que ambos implementam os componentes básicos dos sistemas adaptativos, assim como serão notadas semelhanças com as ontologias desenvolvidas no presente trabalho. No entanto, as propostas diferenciam-se em alguns aspectos. Em [Muñoz 2004] é proposta a arquitetura de um sistema completo para *e-learning*, envolvendo tanto o ambiente para apresentação quanto a adaptação, sendo que as ontologias são componentes dessa arquitetura. Já em [Neto 2006], as ontologias são componentes de um sistema que gera cursos no padrão SCORM, que podem ser importados para um ambiente de *e-learning*. Portanto, depois dessa importação, é necessário que, para cada curso gerado, sejam associados os estudantes que possuem características condizentes com o curso. Já na proposta apresentada neste trabalho, as ontologias também colaboram na arquitetura de um sistema adaptativo. No entanto, esse sistema é integrado ao ambiente Moodle e a adaptação é feita em tempo de execução, baseando-se nos estilos de aprendizagem dos estudantes e em seu desempenho durante o curso, conforme descrito no capítulo

4.

[Durao et al. 2008] apresenta possíveis abordagens para sistemas adaptativos e modelagem do perfil de usuário, sugerindo estratégias de personalização de recursos de aprendizagem. Assim, ao contrário de aplicações web tradicionais que simplesmente utilizam *cookies* para obter informações dos usuários, podem ser utilizadas tecnologias adicionais como: web hipermídia, modelagem de usuário, web semântica, personalização de grupos e análise estatística de ações de usuários. Web Hipermídia fornece o conhecimento de como organizar hipertextos dentro do hiperespaço. Modelagem de usuário é uma atividade essencial para manter informações sobre o conhecimento do usuário, crenças, objetivos, planos, habilidades, atitudes e preferências. Decisões de personalização podem ser formalmente representadas pelas regras da web semântica juntamente com restrições de ontologia para decidir, por exemplo, que *links* mostrar, ocultar ou recomendar. Personalização de grupos pode trazer o conhecimento necessário para formação de grupos e satisfação individual. E finalmente, a aplicação de análises estatísticas sobre os dados aumenta a chance de efetivar a personalização no sistema.

[Durao et al. 2008] acrescenta que as tecnologias da web semântica provêm uma representação formal para o conhecimento na web, utilizando a modelagem de usuário como base para a adaptatividade. Assim, ontologias podem ser instrumentos para gerar *links* adequados aos interesses de um determinado usuário ou simplesmente ocultar *links* não adequados a seus objetivos. De acordo com [Dolog e Nejdl 2007], ontologias do modelo do estudante provêm significados adicionais para decidir quais *links* mostrar, ocultar, gerar e reordenar. A web semântica permite a utilização de sistemas de raciocínio para decisões de personalização. Regras de dedução podem ser empregadas e técnicas de raciocínio podem ser aplicadas com base nos metadados das ontologias, para inferir novos fatos ou informações que não são explicitamente apresentados, e assim, gerar *links* baseados no conteúdo, contexto do usuário e modelo do estudante.

Um exemplo de como usar semântica para personalização na web pode ser encontrada em SEWeP [Eirinaki et al. 2006], um sistema baseado em similaridades semânticas calculadas de acordo com os termos da ontologia. Apresenta um sistema dinâmico de *workflow*, que começa quando o conteúdo web é anotado automaticamente com termos da ontologia, incluindo uma extração de palavras-chave e uma caracterização semântica, baseada no contexto.

Percebe-se, na apresentação desses trabalhos, que a web semântica é uma proposta viável para a implementação de sistemas adaptativos. Além disso, conforme descrito no capítulo 3, a web semântica define padrões a serem utilizados no desenvolvimento das aplicações, o que possibilita a interoperabilidade. Assim, neste trabalho, visando a alcançar essas duas características, optou-se por utilizar essas tecnologias no desenvolvimento do sistema adaptativo proposto, descrito no capítulo 4.

## 2.2 Ambientes de *e-Learning*

[Masie 2001] sugere três dimensões para o significado da letra "e" no termo *e-learning*:

1. Experiência: ao aumentar o envolvimento e a experiência dos alunos na aprendizagem, disponibilizando opções de aprendizagem independentes do local e do instante, e mecanismos de comunicação em rede.
2. Extensão: ao disponibilizar um conjunto de opções de aprendizagem, a fim de alcançar a perspectiva do aluno num *processo* e não apenas num *evento*.
3. Expansão: ao possibilitar expandir a aprendizagem para além das limitações da sala de aula tradicional, permitindo acesso global a um número ilimitado de tópicos.

De acordo com [Kemczinski 2005], um ambiente de aprendizagem deve conter todos os recursos necessários para que se consiga executar um processo educacional informatizado. Assim, apresenta uma definição sobre *e-learning* como sendo: *sistemas computacionais disponíveis na Internet, que auxiliam o processo de ensino-aprendizagem por meio de atividades mediadas pelas Tecnologias de Informação e Comunicação, permitindo construir e organizar conteúdos, desenvolver interações entre pessoas por meio da integração de múltiplas mídias e recursos (ferramentas)*.

Muitos cursos baseados na web têm sido disponibilizados nos últimos anos. Isso se deve, entre outros fatores, à diversidade de plataformas disponíveis para utilização pelas instituições como, por exemplo, [Moodle 2010d], [Teleduc 2010] e [Claroline 2010]. Em [EduTools 2010], pode-se comparar as características presentes em diversos ambientes, considerando-se as seguintes funcionalidades:

- Ferramentas para o estudante:
  - Ferramentas de comunicação: fórum de discussão, gerenciamento de listas de discussão, troca de arquivos, correio eletrônico, anotações, bate-papo, quadro branco.
  - Ferramentas de produtividade: *bookmarks*, calendário/revisão de progresso, busca interna no curso, trabalho *offline*/sincronização, ajuda/orientação
  - Ferramentas para gerenciamento de grupos: colaboração, comunidades, portfólios.
- Ferramentas para suporte:
  - Ferramentas de administração: autenticação, autorização de cursos, possibilidade de integração com outros sistemas para registro/authenticação.
  - Ferramentas para entrega de cursos: gerenciamento e suporte a testes automatizados, gerenciamento de notas/avaliações, gerenciamento de cursos, rastreamento do estudante.

- Ferramentas para desenvolvimento de cursos (autoria): acessibilidade, suporte a padrões para reuso/compartilhamento de conteúdos, modelos (*templates*) de cursos, customização do *layout*, ferramentas para projeto instrucional, conformidade a padrões instrucionais.
- Especificações técnicas:
  - *Hardware/Software*: suporte a diferentes bancos de dados, navegadores, sistemas operacionais.
  - Licenciamento: tipo de licenciamento, custos.
  - Outras opções técnicas.

Em geral, as opções disponíveis nos ambientes são semelhantes, apresentando a maioria das características citadas. Alguns apresentam aspectos mais específicos, como suporte a padrões para *e-learning* e facilidade de integração com outras bases de dados. No entanto, geralmente nada é dito sobre adaptatividade nesses ambientes; ou seja, eles são voltados principalmente para tarefas de gerenciamento de cursos e apresentação de conteúdos. Assim, possuem informações que se relacionam ao componente Modelo do Domínio dos sistemas adaptativos, como a estruturação de conteúdos, mas não implementam um Modelo do Estudante elaborado, de forma a obter características que permitam um acompanhamento dinâmico individualizado, e, menos ainda, um Modelo de Adaptação, com regras definidas para adaptação do conteúdo ou de sua apresentação de acordo com o perfil de cada estudante.

No contexto de utilização desses ambientes, um dos mais adotados pelas instituições é o Moodle, conforme revelam as estatísticas presentes na tabela 2.1. Além disso, trata-se de um sistema *open-source* e extensível, que suporta os principais padrões para *e-learning*. Assim, esse ambiente foi escolhido para ser integrado a um sistema adaptativo e uma breve descrição de suas características é apresentada na seção 2.2.1.

### 2.2.1 Moodle

Nesta seção apresenta-se as principais características do ambiente e discute-se a viabilidade de integrá-lo aos sistemas adaptativos. No capítulo 4 são descritas as informações presentes no Moodle que podem associar-se aos componentes dos sistemas adaptativos, assim como as atividades envolvidas na implementação de um agente que integra o ambiente a sistemas adaptativos.

O Moodle [Moodle 2010d]) é um sistema LMS que se tornou muito popular entre educadores de todo o mundo. Em [Moodle 2010f] são apresentadas as estatísticas de utilização do ambiente; entre elas, destacam-se aquelas descritas na tabela 2.1.

---

<sup>1</sup>Fonte: <http://moodle.org/stats/>.

Tabela 2.1: Estatísticas de utilização do Moodle em 22/07/2010

|   |                                     |            |
|---|-------------------------------------|------------|
| 1 | Sites registrados válidos           | 51.948     |
|   | Sites registrados válidos no Brasil | 3.350      |
|   | Número de países                    | 214        |
|   | Cursos                              | 3.659.212  |
|   | Usuários                            | 36.465.561 |
|   | Usuários registrados                | 776.856    |
|   | Professores                         | 1.230.640  |

Entre as principais características do ambiente, que contribuem para sua popularidade na área acadêmica, pode-se citar:

1. Composição flexível das atividades dos cursos, tais como fóruns, *wikis*, questionários, pesquisas de opinião, tarefas, *chats*.
2. Extensibilidade, por meio de *plugins* com novos temas, novos módulos de atividades e pacotes de idiomas, entre outras opções.
3. Suporte aos padrões SCORM e IMS *Content Package*, permitindo o compartilhamento de recursos com outros ambientes.
4. Extensa documentação, tanto para administradores quanto para desenvolvedores, e uma comunidade ativa, como se pode perceber pelo número de usuários registrados.

Apesar de possuir todas essas características, que lhe concedem uma posição de destaque em relação à maioria dos ambientes disponíveis, o Moodle é restrito em termos de adaptatividade. Os próprios desenvolvedores do sistema já atentam-se para essa questão ao propor ferramentas, como o rastreamento do progresso do estudante (*Progress Tracking*) [Moodle 2010b], que permitirá acompanhar o usuário, mostrando o que foi alcançado e o que ainda deve ser aprendido. Segundo a especificação da ferramenta, isso poderia ser feito de duas maneiras: por curso, mostrando o percentual de conclusão, ou por usuário e competência, de forma a permitir o relacionamento de cursos e atividades a determinados resultados que se pretende alcançar; o relatório de progresso mostraria uma lista de resultados como conhecimentos adquiridos e objetivos futuros para cada usuário.

Essa ferramenta permitiria, portanto, ao professor um acompanhamento personalizado da aprendizagem dos estudantes e a realização de intervenções, como traçar ou ajustar planos de aprendizado individual. Além disso, os estudantes poderiam observar seu próprio planejamento individual de aprendizado e seu progresso em determinado curso. Sua implementação, no entanto, foi abandonada por falhar em entregar código suficiente para a próxima versão do Moodle (2.0), conforme exposto em [Moodle 2010b]. Nada foi dito se o projeto terá continuidade em uma nova versão.

Outro recurso que procura proporcionar adaptatividade ao ambiente, disponível para a versão 2, cuja versão estável ainda não foi lançada, são as Atividades Condicionais. Esse

recurso permite ao professor restringir a disponibilidade de uma atividade de acordo com determinadas condições como datas, resultados obtidos ou outras atividades concluídas. Assim, uma das opções para o professor é agrupar estudantes de acordo com seu desempenho. Após um questionário, por exemplo, pode-se definir uma atividade para estudantes que tenham um obtido uma nota baixa, assim como outras atividades para quem obteve nota média ou alta.

Percebe-se que essas características podem trazer certo nível de adaptatividade ao ambiente. Mas ainda insuficiente, já que não se baseiam nos componentes dos sistemas adaptativos, de forma a possibilitar uma visão geral do processo de adaptação, nem propõem um mecanismo dinâmico para realizar a adaptação, ou seja, dependem da intervenção direta do usuário.

O banco de dados do Moodle tem em torno de duzentas tabelas. As tabelas são distribuídas em grupos de acordo com a atividade ou informação que armazenam; assim, aquelas que trazem informações sobre os cursos iniciam-se com o prefixo *course*; as tabelas do fórum iniciam-se com o prefixo *forum*, e assim por diante. Basta, portanto, analisar as tabelas do bloco que se pretende utilizar ou conhecer para obter uma compreensão incremental do sistema.

Na documentação do Moodle, encontra-se uma breve descrição das principais tabelas [Moodle 2010a], e ainda o Diagrama Entidade-Relacionamento da versão 1.8 do sistema [Moodle 2010e]. Esse diagrama não apresenta o relacionamento entre as tabelas, apenas agrupa regiões de forma de cada uma corresponda a uma área chave do ambiente; assim tem-se *R\_course* para as tabelas de cursos, *R\_user* para as tabelas de usuários, *R\_foruns* para fóruns, e assim por diante. A partir da versão 2.0 do sistema, ainda em desenvolvimento, pode-se ainda acessar a opção *Administração - Desenvolvimento - XMLDB editor - Doc*, para conhecer mais detalhes sobre as informações da base de dados.

O fato de ser um sistema *open-source*, bem documentado, facilita o trabalho de análise para integração com outros sistemas. No capítulo 4 será feita uma descrição das principais tabelas do Moodle, relacionando-as aos componentes dos sistemas adaptativos.

## 2.3 Objetos de Aprendizagem e Padrões para *e-learning*

Existem vários termos utilizados como sinônimos para Objetos de Aprendizagem, tais como objetos instrucionais [Gibbons et al. 2000] e objetos educacionais [Tarouco et al. 2003]. O grupo de trabalho que estuda a padronização de metadados para objetos de aprendizagem, *Learning Object Metadata Working Group* [Committee 2010], define um objeto de aprendizagem como sendo qualquer entidade digital ou não digital que possa ser usada, reutilizada ou referenciada durante o uso de tecnologias que suportem o ensino.

Para [Wiley 2000] a principal idéia de Objetos de Aprendizagem é permitir aos *designers* instrucionais a construção de pequenos componentes instrucionais que podem ser

reutilizados inúmeras vezes em diferentes contextos de aprendizagem. Esta idéia possibilita que os materiais de aprendizagem tornem-se mais estruturados, organizados e que possam ser disponibilizados na web em vários formatos diferentes.

Para que possam ser reutilizados e compartilhados, os objetos de aprendizagem são descritos segundo padrões de metadados. Metadados são um conjunto de informações que descrevem um recurso. Literalmente, metadados significa dados sobre dados [Wiley 2000]. Os metadados são utilizados para descrever as principais informações de um objeto de aprendizagem, através de uma estrutura padronizada, de forma a facilitar a recuperação e o acesso ao recurso desejado. Um dos padrões mais utilizados é o LOM, cujos principais propósitos, segundo [Committee 2010], são:

- Permitir a estudantes e instrutores pesquisar, avaliar, adquirir e utilizar objetos de aprendizagem.
- Desenvolver objetos de aprendizagem em unidades que podem ser combinadas ou decompostas de maneiras significativas.
- Possibilitar que sistemas computacionais, de forma automática e dinâmica, componham lições personalizadas para um estudante, ou ainda compartilhem e troquem objetos de aprendizagem.

A partir das anotações feitas nos objetos com base nos padrões de metadados é possível, portanto, compartilhar objetos entre diferentes ambientes, assim como definir objetos personalizados a serem apresentados para um estudante. Assim, no trabalho proposto, tanto para a integração de sistemas adaptativos a ambientes de *e-learning*, quanto para o desenvolvimento do sistema adaptativo, será utilizado o padrão SCORM para estruturação do conteúdo a ser apresentado ao estudante. O padrão SCORM incorpora o padrão LOM para descrever os objetos de aprendizagem. Assim, será feita uma breve descrição de ambos nas subseções 2.3.1 e 2.3.2.

### 2.3.1 Learning Object Metadata (LOM)

O padrão LOM é um dos mais adotados para descrição dos objetos de aprendizagem, principalmente por ser incorporado por outros padrões, como o próprio SCORM. A representação em XML do padrão LOM está descrita em [Committee 2005]. Este padrão identifica diferentes aspectos pelos quais um objeto de aprendizagem pode ser anotado e é suportado, parcialmente ou não, pela maioria dos repositórios e plataformas de *e-learning*. Os elementos que descrevem um objeto de aprendizagem no padrão LOM são agrupados em nove categorias:

1. Geral: agrupa informações gerais que descrevem o objeto de aprendizagem como um todo. Apresenta os seguintes subelementos: rótulo, título, língua adotada, palavras-chave.

2. Ciclo de vida: agrupa informações relacionadas à história do objeto de aprendizagem e a outros objetos que influenciaram em sua evolução. Algumas informações nessa categoria são: versão, estado de conclusão, entidades que contribuíram para elaboração do material.
3. Meta-Metadados: agrupa informações sobre os metadados utilizados, como nome e versão da especificação, língua, esquema de catalogação.
4. Técnica: agrupa os requisitos e características técnicas do objeto, tais como formato, tamanho em *bytes*, requisitos de *software* e *hardware* para ser exibido.
5. Educacional: agrupa as características educacionais e pedagógicas do objeto de aprendizagem, tais como *tipo de interatividade*, *tipo de recurso de aprendizagem*, *nível de interatividade*, *grau de dificuldade* e *densidade semântica*. Nesse grupo, concentram-se os principais elementos utilizados para descrição das características dos objetos de aprendizagem a serem utilizadas pelos sistemas.
6. Direitos: descreve informações sobre direitos, propriedade intelectual e termos de uso do objeto de aprendizagem.
7. Relacionamento: contém informações sobre o relacionamento entre o objeto de aprendizagem e outros objetos, tais como *pré-requisito*, *parte de*, *requerido por*.
8. Anotações: descreve comentários sobre a utilização educacional do objeto de aprendizagem e informações de quando e quem fez o comentário.
9. Classificação: descreve o objeto em relação a algum sistema de classificação. Algumas propriedades nessa categoria são: propósito da classificação, caminho taxonômico, descrição do objeto em relação ao seu propósito.

A figura 2.6 apresenta uma visão geral dessas categorias e de todos os seus elementos. [Friesen et al. 2003] apresenta um guia de boas práticas que visa simplificar e auxiliar a interpretação do padrão, identificando um subconjunto dos elementos, selecionados em função de sua simplicidade e utilidade para descobrir e compartilhar recursos, e fornecendo uma orientação sobre a interpretação e aplicação desses elementos.

Apesar de ter como finalidade principal a interoperabilidade, algumas críticas são feitas em relação ao uso do padrão LOM. [Brooks e McCalla 2006] defendem que ele não é adequado para permitir a montagem dinâmica e automática de cursos, pois muitos repositórios suportam apenas poucos elementos disponíveis e a maioria não possui um formato que permita buscas externas de forma que agentes de software possam recuperar objetos. Os recursos são descritos utilizando XML, que possui limitações, expressando apenas a estrutura e organização dos elementos. Assim, neste trabalho, pretende-se superar essas limitações ao utilizar uma ontologia, desenvolvida com base nos elementos do padrão, para representar os objetos de aprendizagem.

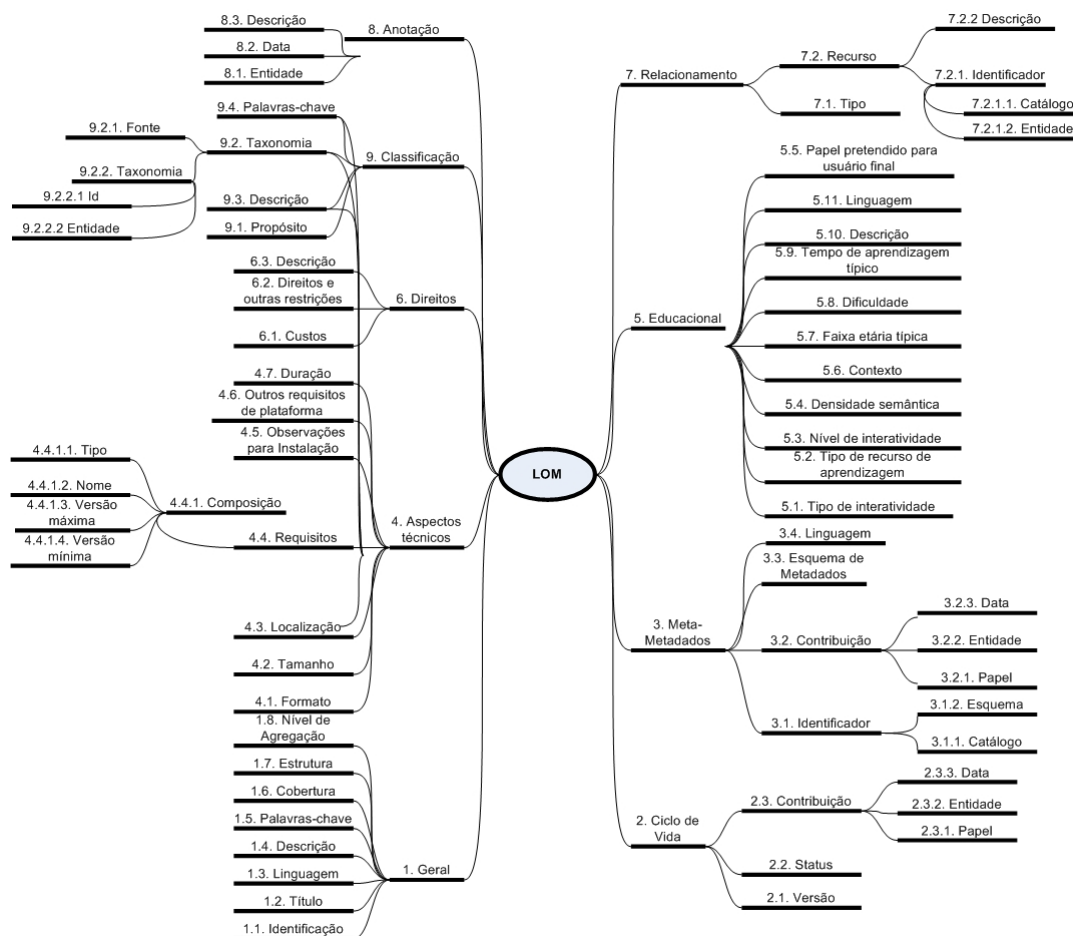


Figura 2.6: Hierarquia de elementos do padrão LOM

### 2.3.2 Shareable Content Object Reference Model (SCORM)

O SCORM integra um conjunto de padrões e especificações para permitir a interoperabilidade entre sistemas. Seu objetivo vai além da descrição dos objetos de aprendizagem isoladamente, pois define uma forma de uniformizar a troca, a gestão e a reutilização dos conteúdos e das tecnologias de aprendizagem, independentemente da sua origem e aplicação. Atualmente, este modelo, disponível na versão 2004 4ª Edição, é composto por três seções: Modelo de Agregação do Conteúdo (CAM - Content Aggregation Model), Ambiente de Tempo de Execução (RTE - Run Time Environment) e Sequenciamento e Navegação (SN - Sequencing and Navigation) [Learning 2010b].

O primeiro grupo, o CAM, descreve os componentes usados em um ambiente, como organizar os objetos de aprendizagem para permitir o compartilhamento entre diferentes sistemas, como descrever esses objetos para permitir a busca e como definir regras de sequenciamento para os componentes.

Algumas informações do CAM são utilizadas pelo segundo grupo, o RTE, para definir como os conteúdos dos objetos de aprendizagem devem ser processados pelos LMS. As especificações do RTE definem os requisitos que ambientes de *e-learning* devem possuir para utilizarem a API SCORM e manipularem cursos construídos de acordo com este

padrão. O RTE também apresenta métodos para conduzir e intermediar as comunicações entre o estudante e o ambiente, incluindo dados sobre a situação do curso, como os materiais que estão sendo apresentados para o estudante, e informações sobre o seu progresso durante o curso, sua pontuação em atividades, limites de tempo para execução de tarefas e outras informações que possam ser utilizadas por sistemas e usuários.

O último grupo, SN, descreve como o conteúdo pode ser sequenciado para os estudantes por meio de um conjunto de eventos de navegação. Esse grupo, que define regras de sequenciamento a serem aplicadas sobre o conteúdo, foi definido a partir da versão 1.4 do SCORM. No entanto, o Moodle suporta apenas a versão 1.2 do padrão, assim não foi possível associar regras de adaptação a partir do modelo desenvolvido pelo SCORM.

Todas estas informações são armazenadas em um arquivo XML, denominado *imsmanifest*. O formato deste arquivo também é descrito e especificado pelo CAM, que por sua vez baseia-se nas especificações do padrão IMS Content Packing [IMS Global Learning Consortium 2010]. Um pacote SCORM é constituído pelos arquivos com seus recursos (objetos de aprendizagem), além do arquivo *imsmanifest.xml*, reunidos em um arquivo compactado do tipo *zip*.

Em um pacote SCORM é possível definir várias organizações, que permitem a estruturação de um curso em vários níveis de complexidade, assim como a associação de diversos recursos de aprendizagem em um mesmo pacote. Essa característica foi a base a partir da qual projetou-se o sistema de integração e o sistema adaptativo descritos no capítulo 4. Após a criação de um pacote SCORM, ele pode ser importado para qualquer ambiente de *e-learning* que suporte o padrão, como o Moodle.

Existem ferramentas disponíveis para a autoria e distribuição de conteúdos em conformidade com o SCORM, como o Reload Editor [Learning 2010a], com o qual é possível criar, importar, editar e exportar pacotes de conteúdo, além de permitir a inserção de anotações para os objetos de aprendizagem, de acordo com o padrão LOM. Essa ferramenta foi utilizada para construção de pacotes SCORM que compõem o curso projetado para os testes do sistema, descrito na seção 4.4.

## 2.4 Estilos de Aprendizagem

Qualquer sistema cuja pretensão seja favorecer a aprendizagem, deve se fundamentar em teorias educacionais consistentes, de forma a oferecer um ambiente propício para que os estudantes desenvolvam suas habilidades. Além disso, ao se apoiar nessas teorias, o sistema ganha credibilidade junto aos profissionais ligados à educação, favorecendo a possibilidade de sua adoção. Uma das teorias educacionais bem conhecidas, com fundamentação psico-pedagógica, é aquela relacionada aos estilos de aprendizagem. Em termos gerais, essa teoria defende que há diversos modos de aprender e cada pessoa tende a aprender melhor de uma determinada forma.

Há diversos estudos sobre estilos de aprendizagem. No trabalho apresentado em [Coffield et al. 2004] há uma comparação entre os principais estudos. [Kolb 1999] sugeriu que, como resultado da bagagem hereditária, das experiências de cada um e das exigências do meio ambiente, as pessoas desenvolvem estilos de aprendizagem que enfatizam algumas habilidades sobre outras, ou seja, cada pessoa desenvolve um estilo pessoal de aprendizagem. [Felder e Silverman 1988] denomina como estilos de aprendizagem uma preferência característica e dominante na forma como as pessoas recebem e processam informações, considerando-os como habilidades passíveis de serem desenvolvidas.

Vários autores, entre eles os citados [Kolb 1999] e [Felder e Silverman 1988], desenvolveram modelos para classificar os estilos de aprendizagem. Esses instrumentos têm o intuito de fornecer subsídios para que a prática docente seja mais adequada, tornando possível realizar um planejamento educacional direcionado à realidade dos estudantes. O modelo proposto por [Kolb 1999] descreve o ciclo de aprendizagem, isto é, a forma como o indivíduo gera, a partir de sua experiência, conceitos que o guiarão em sua prática. Este modelo classifica os estudantes como tendo preferência por:

1. experiência concreta ou conceituação abstrata, que se refere à forma como as pessoas percebem a informação; e
2. observação reflexiva ou experimentação ativa, referindo-se à forma como internalizam ou processam a informação.

Já o modelo proposto por [Felder e Silverman 1988] estabelece medidas para as preferências individuais em quatro dimensões do processo de aprendizagem: percepção da informação (sensorial/intuitivo), recepção da informação (visual/verbal), processamento da informação (ativo/reflexivo) e compreensão ou sequenciamento da informação (sequencial/global). Como esse modelo propõe uma classificação mais ampla e adequada a uma associação com as características dos objetos de aprendizagem, ele foi adotado neste trabalho como base para definição das regras de adaptação. Assim, segue uma descrição das principais características de cada dimensão, baseada em [Felder e Silverman 1988].

- **Ativos x Reflexivos:** os estudantes ativos tendem a compreender e reter melhor a informação trabalhando de modo ativo, agindo sobre algo, discutindo e aplicando a informação ou explicando-a para os outros, tendem a gostar mais do trabalho em equipe. Já os aprendizes reflexivos preferem primeiro refletir sobre a informação, tendem a gostar mais de trabalhar sozinhos.
- **Sensoriais e Intuitivos:** aprendizes sensoriais gostam de aprender fatos, resolver problemas com métodos bem estabelecidos, sem complicações e surpresas. Tendem a ser mais detalhistas e bons para memorizar fatos e fazer trabalho prático, pois são metódicos. Aprendizes intuitivos preferem descobrir possibilidades e relações, gostam de novidades e aborrecem-se com a repetição, preferem mais conceitos e

teorias. Os sensoriais tendem a ser mais práticos e cuidadosos do que os intuitivos; os intuitivos são mais rápidos no trabalho e mais inovadores.

- **Visuais e Verbais:** os aprendizes visuais se recordam mais facilmente de conteúdo com representação gráfica, como figuras, fluxogramas, filmes, demonstrações. Privilegiam as informações que recebem por imagem, diagramas, gráficos, esquemas. Os verbais tem mais facilidade com as palavras, explanações escritas ou faladas. Privilegiam o que é falado ou escrito.
- **Sequenciais e Globais:** aprendizes sequenciais tendem a aprender de forma linear, em etapas sequenciadas. Os aprendizes globais tendem a aprender em grandes saltos, assimilando o material quase aleatoriamente, sem ver as conexões, para, depois, compreender o todo.

Para classificar os estudantes, Felder, juntamente com Bárbara Soloman, desenvolveram o Índice de Estilos de Aprendizagem (ILS - Index of Learning Styles), que determina, com base nas respostas de 44 questões, as preferências de aprendizagem dos estudantes. Após preencher e submeter o formulário com as questões, disponível em [Felder e Soloman 2010], um resultado semelhante ao representado na figura 2.7 é apresentado ao usuário.

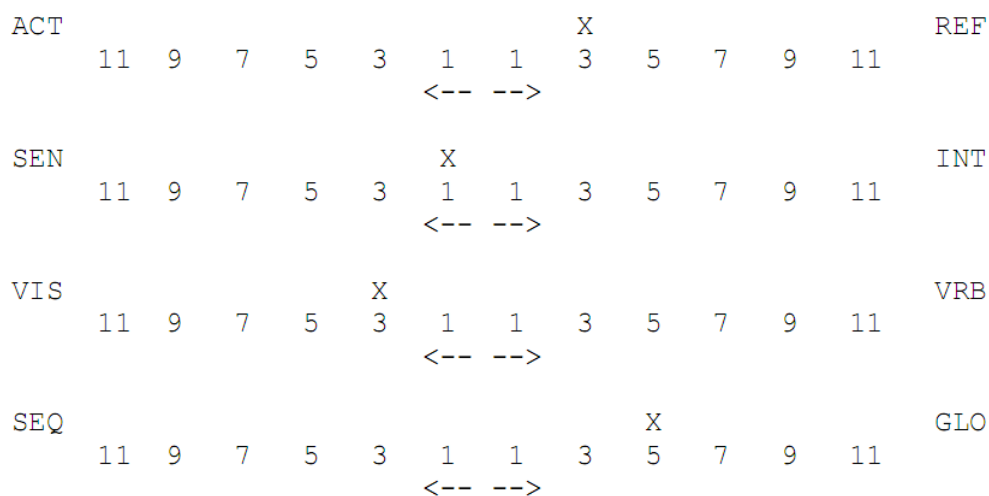


Figura 2.7: Exemplo de resultado do Índice de Estilos de Aprendizagem proposto por [Felder e Soloman 2010].

Como mostra a figura 2.7, cada estudante é classificado em quatro dimensões. Assim, se em determinada dimensão a pontuação estiver em uma escala de 1-3, significa que há um equilíbrio entre os dois estilos. Se a pontuação estiver em uma escala de 5-7, o estudante tem uma preferência moderada para uma dimensão da escala e aprende mais facilmente em um ambiente de ensino que favorece essa dimensão. Se a pontuação estiver em uma escala de 9-11, denota uma preferência forte para uma dimensão da escala; em termos práticos, isso significa que, se o ambiente não suportar essa preferência, o estudante pode ter dificuldades de aprendizagem.

No capítulo 4 há uma associação desses estilos com as características dos objetos de aprendizagem, visando à definição das regras para adaptação.

## 2.5 Considerações

Com o estudo de trabalhos correlatos foi possível conhecer o estado da arte em relação ao desenvolvimento de ambientes de *e-learning* e sistemas adaptativos. Foram apresentados ainda sistema adaptativo SIMEDUC e o ambiente de *e-learning* Moodle, selecionados para o desenvolvimento da proposta de integração.

O conhecimento das características desses sistemas foi fundamental para a definição de estrutura necessária para permitir a integração entre eles, a fim de que os usuários possam se beneficiar tanto do uso abrangente dos ambientes de *e-learning* quanto da adaptatividade presente nos sistemas adaptativos.

Além disso, foram estudados os padrões para *e-learning* SCORM e LOM, assim como a teoria de estilos de aprendizagem, nos quais as propostas de integração e adaptação se baseiam. Dessa forma, os tópicos apresentados neste capítulo permitiram identificar as informações necessárias ao desenvolvimento do módulo de integração. O estudo dos temas abordados neste capítulo contribuíram também para definir os requisitos, as ontologias e arquitetura do sistema adaptativo, que utiliza tecnologias da web semântica, abordadas no próximo capítulo.

# Capítulo 3

## Web Semântica

O termo *web semântica* foi proposto por [Tim Berners-Lee 2001], com a seguinte definição: “a web semântica não é uma web separada, mas uma extensão da atual, na qual a informação é utilizada com significado bem definido, aumentando a possibilidade de computadores e pessoas trabalharem em cooperação”. Com a proposta de prover um *framework* que possibilite o compartilhamento e reuso de dados entre aplicações, empresas e comunidades, ou seja, uma “web de dados” e não apenas uma “web de documentos”, a web semântica tornou-se um projeto liderado pela W3C [Consortium 2010b].

Este capítulo tem por finalidade descrever a web semântica, de forma a apresentar as suas principais tecnologias e como elas estão relacionadas ao trabalho proposto. A maioria das tecnologias citadas são recomendações e padrões adotados pela W3C e possuem uma ampla documentação. Assim, não se pretende fazer uma descrição exaustiva nem apresentar todos os detalhes técnicos sobre as tecnologias envolvidas. As tecnologias serão descritas brevemente, com mais ênfase naquelas que se relacionam diretamente com o trabalho desenvolvido, e o referencial teórico pode ser consultado a fim de se obter maiores detalhes sobre elas. Na seção 3.1 são apresentados os conceitos e objetivos da web semântica. Sua arquitetura e as principais tecnologias pertencentes às suas camadas são descritas na seção 3.2. Na seção 3.3 são feitas as considerações finais deste capítulo.

### 3.1 Web Semântica: Conceitos e Objetivos

Como sua principal proposta é construir uma “web de dados”, a web semântica trata sobre formatos comuns para integração e combinação de dados provenientes de diversas fontes e sobre uma linguagem para armazenar como os dados se referem aos objetos do mundo real, permitindo que pessoas e máquinas encontrem informações relacionadas entre si, construam vocabulários compartilhados e definam regras para a manipulação de dados.

O projeto web semântica, disponível em [Consortium 2010b], está categorizado em cinco grupos principais:

1. Dados interligados: ao propor uma web de dados, isso não significa somente disponibilizar dados na web, mas, principalmente, criar relações entre esses dados de forma que possam ser explorados por pessoas e máquinas. Para tornar isso possível, é necessário disponibilizar os dados na web em um formato padrão, acessível e gerenciável por ferramentas, de forma que permita criar relacionamentos entre esses dados. *Resource Description Framework* (RDF) é o padrão que fornece a base para a publicação e vinculação de dados; assim diversas tecnologias permitem manipular os dados, inserindo dados em documentos, permitindo a busca em documentos RDF e definindo regras para inferir novos relacionamentos entre esses dados.
2. Vocabulários/Ontologias: na web semântica, vocabulários definem os conceitos e relações usados para descrever e representar uma determinada área. São usados para classificar os termos que podem ser usados em uma aplicação particular, caracterizar as possíveis relações e definir restrições no uso desses termos. Os vocabulários podem ser muito complexos, com milhares de termos, ou muito simples, descrevendo apenas um ou dois conceitos.
3. Busca: no contexto da web semântica, busca refere-se às tecnologias e protocolos que podem programaticamente recuperar informações da “web de dados”. Assim como bancos de dados relacionais ou documentos XML necessitam de linguagens específicas de consultas, a web semântica, tipicamente representada usando RDF como formato de dados, necessita de uma linguagem de consulta para RDF, que no caso é o *Simple Protocol and RDF Query Language* (SPARQL).
4. Inferência: refere-se aos procedimentos automáticos que podem gerar novos relacionamentos com base em informações presentes nos vocabulários ou em conjuntos de regras, com a intenção de melhorar a qualidade da integração de dados na web, já que esses mecanismos podem, inclusive, descobrir inconsistências nos dados.
5. Aplicações verticais: aplicações verticais é o termo usado pela W3C para designar aplicações de áreas genéricas, comunidades específicas, que exploram como as tecnologias da W3C, no caso as tecnologias da web semântica, podem ser utilizadas em aplicações para melhorar operações, aumentar a eficiência, proporcionar uma melhor experiência ao usuário, entre outras possibilidades.

[Hardesty 2010] apresenta uma analogia para diferenciar a web atual, que seria como um gigantesco arquivo de texto, que pode ser pesquisado por palavras em particular, e a web semântica, que seria como um banco de dados, em que cada item de informação é categorizado, e novas consultas podem combinar as categorias de qualquer maneira imaginável. E ainda, enquanto um banco de dados tem as categorias previamente selecionadas por um programador, a web semântica é um “banco de dados onde cada pessoa controla seus próprios dados”.

A web semântica tem como proposta, portanto, fornecer uma infra-estrutura que possibilite não apenas *homepages*, mas também bases de dados, serviços, programas, sensores e dispositivos em geral a consumir e a produzir informações para a web. Agentes de software podem usar estas informações para buscar, filtrar e preparar informações de diferentes formas para ajudar as pessoas na execução das mais diversificadas tarefas, conforme descrito em [Hendler 2001]. Assim, optou-se nesse trabalho por utilizar essas tecnologias, tendo em vista a necessidade ambientes interoperáveis em *e-learning*, que permitam o compartilhamento de recursos, de forma a facilitar a tarefa de construção de cursos.

## 3.2 Camadas da Web Semântica

Para que a web semântica atinja seu objetivo, que pode ser interpretado, em termos práticos, como a capacidade de proporcionar ambientes padronizados para o intercâmbio de informações, são necessários tecnologias, linguagens e metadados para estabelecer regras que sejam comuns e compartilhadas em relação aos significados dos conteúdos das informações. Nesse sentido, para facilitar o desenvolvimento da web semântica, o projeto, liderado pela W3C [Consortium 2010b], propõe uma arquitetura em camadas, conforme figura 3.1.

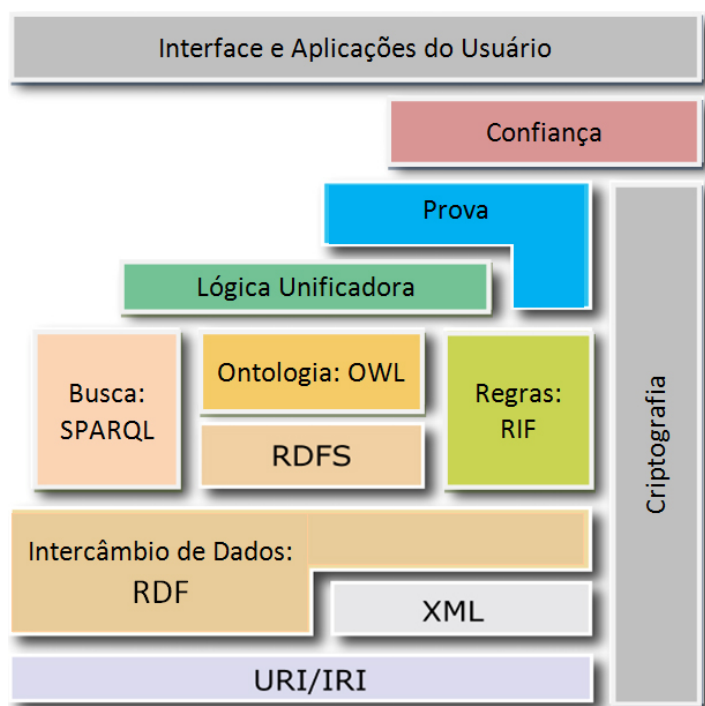


Figura 3.1: Camadas da web semântica

Em relação às tecnologias presentes nas camadas mostradas na figura 3.1, aquelas de nível inferior referem-se à representação e estruturação dos dados, tratando-se de tecnologias já consolidadas. A camada URI/IRI fornece a interoperabilidade em relação à codificação de caracteres e ao endereçamento e nomeação de recursos. A camada XML

abrange a linguagem XML e XML *Schema* e tem como propósito garantir uma sintaxe comum para descrição de recursos, fornecendo elementos para descrever a estrutura e restringir o conteúdo de documentos.

A camada RDF trata-se de um *framework* para representar informação (metadados) sobre recursos, que se baseia em triplas “sujeito-predicado-objeto”. As principais especificações para RDF abrangem um modelo de dados [Lassila e R. Swick 1999], para expressar declarações sobre os recursos, e uma sintaxe baseada em XML [Dave Beckett 2004], para o intercâmbio das declarações. Além do modelo e da sintaxe, RDF também fornece uma linguagem - RDF *Schema* - que estende a especificação básica do RDF para permitir a definição de vocabulários. Assim, o RDF *Schema* é uma linguagem para a representação de ontologias simples, que está definida em uma camada superior a RDF, fornecendo o suporte necessário para descrever classes e propriedades.

A camada Ontologia fornece suporte para a evolução de vocabulários e para processar e integrar informações existentes sem problemas de indefinição ou conflito de terminologia. A linguagem RDF *Schema* permite a construção de ontologias com expressividade e inferência limitadas, pois fornece um conjunto básico de elementos para a modelagem e poucos desses elementos podem ser utilizados para inferência. A *Web Ontology Language* (OWL) estende o vocabulário da RDF *Schema*, incluindo elementos com maior poder de expressividade e inferência. Como são baseadas na lógica descritiva, uma semântica formal é definida tanto para RDF *Schema* [Patrick Hayes 2004] como para OWL [Patel-Schneider et al. 2009].

As ontologias se concentram em métodos de classificação, enfatizando a definição de *classes*, *subclasses* e a forma como cada recurso pode ser associado a essas classes, caracterizando as relações entre as classes e suas instâncias. Regras, por outro lado, concentram-se na definição de um mecanismo geral de descoberta e geração de novas relações com base nas existentes, ou seja, um mecanismo de inferência.

A camada Lógica fornece suporte para a descrição de regras para expressar relações sobre os conceitos de uma ontologia, as quais não podem ser expressas com a linguagem de ontologia utilizada. O grupo de trabalho *Rule Interchange Format* (RIF) foi constituído em 2005 com o objetivo de criar um padrão para troca de regras entre sistemas, em particular entre os mecanismos de regras para a web. Ao invés de tentar desenvolver uma única linguagem para regras, o foco do RIF foi em compartilhamento de regras, pois, em contraste com outros padrões da web semântica, como RDF, OWL e SPARQL, uma única linguagem não é capaz de satisfazer as necessidades de muitos paradigmas populares para uso de regras na representação de conhecimento e na modelagem de negócios. Assim, a abordagem utilizada foi projetar uma família de linguagens, denominada dialetos, com uma sintaxe e semântica bem definidas, de forma que pudesse ser uniforme e extensível, conforme [Boley e Michael Kifer 2010].

Para consulta de dados, o padrão adotado foi o SPARQL, que não é apenas uma

linguagem de consulta, mas também um protocolo para acesso a dados RDF. SPARQL usa triplas e recursos RDF para a definição e retorno de resultados de consultas. Uma vez que tanto RDF *Schema* como OWL são construídos em RDF, SPARQL pode ser usado para consultar ontologias e bases de conhecimento diretamente, conforme exposto em [Prud'hommeaux e Andy Seaborne 2008].

Pela disposição das camadas presentes na figura 3.1 fica claro ainda que a web semântica tem como propósito que a semântica definida nas ontologias e nas regras sejam executados nas camadas abaixo da camada Prova e os resultados sejam utilizados para comprovar as deduções. Uma prova formal, juntamente com as contribuições da camada Confiança para a prova significa que os resultados são confiáveis. Percebe-se ainda, pela camada transversal à figura, que para entradas confiáveis deve-se utilizar Criptografia, como assinaturas digitais para verificação da origem das fontes. Aplicações para o usuário final podem ser construídas apoiando-se nas tecnologias presentes nas camadas descritas.

Uma descrição mais detalhada de algumas dessas tecnologias, especialmente aquelas que foram utilizadas na definição do trabalho proposto, é feita nas próximas seções. Assim, são apresentadas as tecnologias para identificação e representação de recursos (URI/IRI), para representação sintática de recursos (XML), para representação semântica de recursos (RDF) e para representação do conhecimento (ontologias).

### 3.2.1 Identificação e representação de recursos: URI/IRI

A camada de nível inferior da web semântica, presente na figura 3.1, representa duas tecnologias sobre as quais a web se apóia: o *Uniform Resource Identifier* - Identificador Uniforme de Recursos (URI) e *Internationalized Resource Identifier* - Identificador de Recursos Internacionalizado (IRI).

Conforme [URI Planning Interest Group 2001], um URI é formado por um conjunto de caracteres que identifica recursos na web, tais como páginas de texto, imagens, arquivos de áudio, contas de *e-mail* e programas. A forma mais conhecida de um URI é o *Uniform Resource Locator* - Localizador Uniforme de Recursos (URL), que identifica um recurso por meio de sua localização na rede. IRI é uma generalização do URI. Enquanto URIs são limitados a um subconjunto do conjunto de caracteres ASCII, IRIs contêm caracteres Unicode. Um mapeamento de URIs para IRIs é possível, ou seja, IRIs podem ser usados apropriadamente para identificar recursos, conforme [Duerst e Suignard 2005].

O Unicode é um padrão para representar caracteres criado pelo Unicode Consortium Inc., independente de plataforma, programa ou linguagem. Nesse padrão, qualquer caractere é identificado por um número de 16 bits, o que permite representar até 65.536 caracteres [Unicode 2010]. A independência proporcionada pelo Unicode é fundamental para o desenvolvimento da web semântica, pois, desta forma, os níveis superiores não precisam se preocupar em decodificar dados e transformá-los em outro formato de repre-

sentação antes de enviar ou receber mensagens.

Ao assumir esses padrões, percebe-se a preocupação da web semântica com a interoperabilidade, em relação à codificação de caracteres, no caso do UNICODE, e ao endereçamento e à identificação e acesso de recursos, no caso de URI/IRI.

### 3.2.2 Representação sintática de recursos: Extensible Markup Language (XML)

Conforme [W3C 2010b], XML é uma linguagem para representação sintática de recursos de maneira independente de plataforma. Ela descreve documentos que armazenam unidades de dados, com estruturação lógica da informação, definindo como o conteúdo deve ser interpretado. Em XML, não há a preocupação com a forma de apresentação das informações, sua tarefa é definir sua organização interna. Essa possibilidade, aliada à flexibilidade para criar *tags* de acordo com a necessidade do usuário e o domínio em questão, tornaram-na uma linguagem essencial para a web semântica, visto que todas as linguagens apresentadas nas camadas acima são serializadas em XML.

As regras para a criação de um documento válido e bem formado em XML podem ser encontradas em [Bray et al. 2006]. Um documento em XML pode ser validado, por meio de uma comparação com um modelo previamente estabelecido. Desta forma, as aplicações que utilizam os dados do documento XML não precisam verificar a existência de todos os campos necessários e se eles estão estruturados corretamente. É necessário apenas que conheçam o modelo utilizado no desenvolvimento do documento e utilizem um *parser* para validação. Essa validação é feita utilizando-se uma Definição de Tipo de Documento (DTD - Data Type Definition) ou, o que é mais recomendado, um esquema XML (documentos XSD), conforme [Gao et al. 2009] e [Peterson et al. 2009]. Estes recursos são fundamentais para que diferentes organizações possam trocar informações por meio de documentos XML, por permitirem a criação de modelos comuns, utilizados com o intuito de padronizar o conteúdo trocado.

Em XML, há uma liberdade para definir a nomenclatura de *tags*, o que pode acarretar conflitos quando diferentes modelos são utilizados em um mesmo documento, como por exemplo o mesmo nome para elementos ou atributos. Para evitar esse problema, desenvolveu-se o conceito de *namespace* que, segundo [Daconta et al. 2003], caracteriza-se como um mecanismo simples para criar nomes únicos para os elementos e atributos de um documento XML, proposto como solução para unificar definições de recursos a fim de evitar conflitos semânticos. Assim, os *namespaces*, ao lado do XML, constituem-se em tecnologias base para o desenvolvimento da web semântica, já que é necessário que as aplicações obtenham informações de diferentes documentos e sistemas.

Pelas características da linguagem e por permitir a criação e compartilhamento de recursos, os padrões para *e-learning* também utilizam esquemas XML para representá-los.

Como neste trabalho utiliza-se o padrão LOM, cujo esquema XML é definido em [Committee 2005], é necessário processar os arquivos que descrevem os objetos de aprendizagem tendo em vista localizar as informações necessárias ao processamento das regras de adaptação. Para a consulta em arquivos XML, utiliza-se a linguagem XQuery, descrita em [Boag et al. 2007].

Conforme disposto em [Neto 2006], apesar de seus benefícios, a XML possui limitações que impedem o desenvolvimento completo da web semântica. Em XML, é possível apenas anexar palavras aos dados, ou seja, os metadados de um documento XML se constituem de palavras isoladas, e não há como definir sentenças, nem relacionamentos mais complexos sobre estas entidades. Assim, novas linguagens e formas para representação do conhecimento foram desenvolvidas, baseadas na representação da própria linguagem XML, e alocadas em níveis superiores da web semântica para suprir essas limitações.

### 3.2.3 Representação semântica de recursos: Resource Description Framework (RDF)

De acordo com [Tim Berners-Lee 2001], o significado é expresso por RDF, que o codifica em um conjunto de triplas, formadas por um sujeito, predicado e objeto de uma sentença. Cada tripla pode ser escrita usando *tags* XML. Em RDF, um documento faz asserções sobre coisas particulares (pessoas, páginas web), ou seja, o sujeito de uma sentença, que tem propriedades (como “é irmã de”, “é autor de”), ou seja, os predicados, com determinados valores (uma outra pessoa, uma outra página web), que correspondem aos objetos. Essa estrutura torna-se uma forma natural para descrever a grande maioria dos dados processados por máquinas. Cada componente dessa estrutura, sujeito-predicado-objeto, pode ser identificado por um URI/IRI, permitindo que diversos vocabulários compartilhem informações.

Conforme [Tim Berners-Lee 2001], as triplas em RDF formam redes de informação sobre coisas relacionadas. Ao usar URIs para codificar informações em um documento, garante-se que os conceitos não são apenas palavras em um documento, mas estão vinculados a uma definição única que pode ser encontrada na web. Por exemplo, imagine que pode-se ter acesso a uma variedade de bases de dados com informações sobre pessoas, incluindo os respectivos endereços. Para encontrar pessoas que vivem em um CEP específico, é necessário saber quais campos em cada banco de dados representam nomes e quais representam códigos postais. Em RDF pode-se especificar que "(o campo 5 no banco de dados A) (é um campo do tipo) (CEP)", usando URIs em vez de frases para cada termo. Dessa forma ele permite que dados sejam compartilhados por diferentes aplicações. Devido a essa característica, RDF tornou-se o modelo padrão para o intercâmbio de dados na web [W3C 2010a].

A estrutura de triplas em RDF forma um grafo direcionado, nomeado, em que as

arestas representam o predicado (propriedade) que liga dois recursos. Os recursos são representados pelos vértices do grafo, sendo o objeto um valor para o sujeito. A figura 3.2 representa graficamente uma tripla RDF.

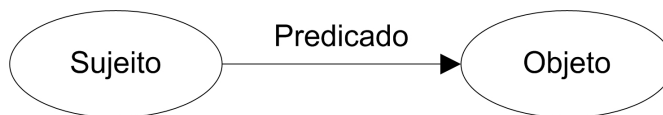


Figura 3.2: Representação gráfica de uma tripla RDF

As tripas RDF são serializadas na forma de XML. Ou seja, RDF é uma linguagem construída sobre o XML e utilizada para fazer afirmações sobre entidades e documentos como um todo. Enquanto XML é utilizada para adicionar metadados a partes de um documento, com o RDF pode-se associar metadados ao documento como uma entidade única [Daconta et al. 2003]. Por ser um *framework* comum, assim como XML, RDF permite que os desenvolvedores utilizem *parsers* e ferramentas disponíveis para gerar e processar documentos RDF.

Assim como no caso de XML, em que são criados esquemas para criar um estrutura padrão para gerar documentos, para RDF definiu-se uma linguagem para descrição de vocabulários, a RDF *Schema*, que é uma extensão semântica de RDF e fornece mecanismos para descrever recursos e os relacionamentos entre eles. Segundo [Daconta et al. 2003] RDF *Schema* é um simples conjunto de recursos e propriedades RDF que possibilita a criação de vocabulários RDF. Em [Brickley e R. V. Guha 2004] é descrito o vocabulário RDF e especificações de sintaxe com classes e propriedades que se originam com RDF *Schema*.

Segundo [Manola e Eric Miller 2004], os recursos fornecidos pelo RDF e pelos esquemas RDF são suficientes para o desenvolvimento de uma série de aplicações úteis, mas ainda não o são para o desenvolvimento completo da web semântica, que requer a existência de linguagens que possibilitem o tratamento da semântica dos recursos pelos computadores. Isso é possível com linguagens para criação de ontologias, em uma relação de extensão com os esquemas RDF, dando um significado comum a termos como *class*, *property*, *subclassof* e demais termos definidos no *namespace* para criação de esquemas RDF. Dessa forma, as ontologias permitem a representação de informações e significados que ainda não podem ser capturados pelos esquemas RDF, aumentando o nível de representatividade da linguagem e utilizando a mesma sintaxe para a criação dos documentos.

### 3.2.4 Representação do conhecimento: Ontologias

#### Conceitos, componentes e metodologia para construção de ontologias

A palavra *ontologia* deriva-se do grego *onto*, ser, e *logia*, discurso escrito ou falado. Na Filosofia, ontologia é uma teoria sobre a natureza da existência, de que tipos de coisas

existem. Pesquisadores em inteligência artificial adaptaram o termo de forma que, na área de ciência da computação, uma ontologia designa um documento ou arquivo que define formalmente as relações entre termos e conceitos.

De acordo com [Gruber 2009], no contexto das ciências da computação e informação, uma ontologia define um conjunto de primitivas representacionais que podem modelar um domínio do conhecimento. Essas primitivas são tipicamente classes (ou conjuntos), atributos (ou propriedades) e relacionamentos (ou relações entre membros de classe) e incluem informações sobre seu significado e restrições em uma lógica consistente.

[Gruber 2009] acrescenta ainda que no contexto de sistemas de banco de dados, ontologias podem ser vistas como um nível de abstração do modelo de dados, semelhante aos modelos relacionais e hierárquicos, mas com a intenção de modelar conhecimento sobre indivíduos, seus atributos e seus relacionamentos com outros indivíduos. Ontologias são tipicamente especificadas em linguagens que não se preocupam em definir estruturas de dados e estratégias de implementação. Na prática, as linguagens para ontologias estão mais próximas do poder expressivo da lógica de primeira ordem do que de linguagens utilizadas para modelar bancos de dados. Por essa razão, as ontologias estão em um nível semântico, enquanto esquemas de banco de dados são modelos de dados de nível lógico ou físico.

Devido a sua independência do nível inferior de modelos de dados, ontologias são usadas para integrar bancos de dados heterogêneos, permitindo a interoperabilidade entre diferentes sistemas e a especificação de aplicações e serviços baseados na representação do conhecimento. Ontologias são parte dos padrões para a web semântica, sendo usadas para especificar vocabulários conceituais e, assim, permitir exportar dados entre sistemas, disponibilizar serviços para resposta a consultas, publicar bases de conhecimento reutilizáveis e oferecer serviços para facilitar interoperabilidade entre múltiplos e heterogêneos sistemas e bancos de dados.

Em [Consortium 2010c] há referência ao termo vocabulário no mesmo sentido que o termo ontologia. Assim, não há uma divisão clara entre os dois termos e a tendência é usar a palavra *ontologia* para uma forma mais complexa e formal para uma coleção de termos, enquanto o termo *vocabulário* é usado quando tal formalismo não é necessariamente usado ou somente em um sentido mais vago. Vocabulários e ontologias são, portanto, os blocos básicos de técnicas de inferência sobre a web semântica.

[Consortium 2010c] acrescenta que depende da aplicação a complexidade usada pelo vocabulário/ontologia. Algumas aplicações podem decidir não usar vocabulários pequenos, e utilizar a lógica imbutida em um programa aplicativo. Alguns aplicativos podem optar por usar um vocabulário simples e deixar que um ambiente geral da web semântica utilize informação extra para fazer a identificação dos termos. Alguns aplicativos precisam de um acordo sobre a terminologia comum, sem qualquer rigor imposto por um sistema lógico. E ainda, algumas aplicações podem necessitar de ontologias mais comple-

xas, com processos de raciocínio complexo. Tudo depende da necessidade e dos objetivos da aplicação.

[Gruber 1993]) destaca os seguintes componentes em uma ontologia: classes, relações, funções, axiomas e instâncias. [Noy e McGuinness 2001] apresentam como componentes das ontologias: classes, propriedades, restrições e instâncias. Pode-se descrever sucintamente os componentes de uma ontologia como:

- **Classes:** são interpretadas como conjuntos que contém indivíduos. As classes representam um conceito ou objeto de um domínio e podem ser organizadas em uma hierarquia de superclasses-subclasses, podendo ser computadas automaticamente por um raciocinador, um sistema de representação de conhecimento que provê formas de manipular bases de dados e extrair conhecimento sobre seu conteúdo. Conforme [Horridge 2009], em OWL classes podem ser nomeadas, ou seja, explicitamente definidas, ou anônimas, isto é, definida por restrições, de forma que uma classe anônima contém todos os indivíduos que possuem o relacionamento requerido para ser membro da classe.
- **Propriedades descritivas:** descrevem as características ou atributos das classes, permitindo associar a um indivíduo um valor de dados.
- **Propriedades relacionais:** descrevem e rotulam os tipos de relações existentes no domínio representado. São também chamadas de propriedades de objetos, pois estabelecem relações binárias entre dois indivíduos (instâncias).
- **Regras e axiomas:** enunciados lógicos que possibilitam impor condições como tipos de valores aceitos, descrevendo formalmente as regras da ontologia e possibilitando a realização de inferências automáticas a partir de informações que não necessariamente foram explicitadas no domínio, mas que podem estar implícitas na estrutura da ontologia.
- **Instâncias ou indivíduos:** indicam os valores das classes e subclasses, constituindo uma representação de objetos ou indivíduos pertencentes ao domínio modelado, de acordo com as características das classes, relacionamentos e restrições definidas.
- **Valores:** atribuem valores concretos às propriedades descritivas, indicando os formatos e tipos de valores aceitos em cada classe.

[Noy e McGuinness 2001] apresentam algumas observações em relação à construção de ontologias:

- não há um modelo correto, existem sempre alternativas viáveis. A melhor solução sempre depende da aplicação que se pretende desenvolver e da evolução que se planeja;
- o desenvolvimento de ontologias é necessariamente um processo iterativo;

- os conceitos presentes na ontologia devem ser próximos aos objetos (físicos ou lógicos) e relacionamentos em seu domínio de interesse, que são, em sua maioria, substantivos (objetos) ou verbos (relacionamentos) em sentenças que descrevem o domínio.

[Noy e McGuinness 2001] ainda sugerem as seguintes etapas para a construção de ontologias:

1. Determinar o domínio e escopo da ontologia.

O desenvolvimento inicia-se pela definição do domínio e escopo, respondendo às questões de competência em relação ao tema estudado. Uma das maneiras de determinar o escopo da ontologia é elaborar uma lista de perguntas que uma base de conhecimento deve ser capaz de responder. Estas questões são denominadas questões de competência. Algumas questões possíveis são:

- (a) Qual é o domínio que a ontologia cobrirá?
- (b) Qual a finalidade de uso da ontologia?
- (c) Quais respostas as informações da ontologia devem trazer?
- (d) Quem usará e manterá a ontologia?

2. Considerar o reuso de ontologias existentes.

Nesta fase é importante a verificação de ontologias já existentes no domínio de conhecimento. A reutilização de ontologias existentes pode ser um requisito se o sistema necessitar interagir com outras aplicações que já tenham sido consideradas por ontologias particulares ou vocabulários controlados. Muitas ontologias já estão disponíveis em formato eletrônico e podem ser importadas para o ambiente de desenvolvimento que está sendo utilizado.

3. Enumerar termos importantes na ontologia.

Nesta fase, listam-se todos os termos que necessitam de definições ou explicações para os usuários. Inicialmente, é importante obter uma lista abrangente de termos, sem se preocupar com a sobreposição entre os conceitos que representam as relações entre os termos, ou qualquer propriedade que os conceitos podem ter, ou se os conceitos são classes ou propriedades.

As duas próximas etapas, o desenvolvimento da hierarquia de classes e definição das propriedades dos conceitos, estão intimamente entrelaçadas. É difícil fazer um deles primeiro e depois fazer o outro. Normalmente, criam-se algumas definições dos conceitos na hierarquia e segue-se descrevendo as propriedades desses conceitos e assim por diante. Estes dois passos são também os mais importantes no processo de concepção da ontologia.

#### 4. Definir as classes e a hierarquia de classes.

A definição de classes e hierarquias pode ser efetivada em uma abordagem *top-down*, de cima para baixo, ou *bottom-up*, de baixo para cima, ou utilizando-se uma combinação das duas abordagens. Um processo de desenvolvimento *top-down* começa com a definição da maioria dos conceitos gerais no domínio e as especializações subsequentes dos conceitos; podem-se criar classes gerais de conceitos e então especializá-las em subclasses, categorizando-as. Um processo de desenvolvimento *bottom-up* começa com a definição da maioria das classes mais específicas, que partem da hierarquia, com subsequente agrupamento destas classes em conceitos mais gerais. Um processo de desenvolvimento por combinação inicia-se primeiro pela definição dos conceitos mais salientes, generalizando-os e especializando-os apropriadamente. Pode-se começar por poucos conceitos de alto nível e poucos conceitos específicos e então relacioná-los com conceitos de nível médio. Nenhum destes três métodos é melhor um do que o outro, o método depende do ponto de vista de quem vai desenvolver a ontologia e do conhecimento que possui do domínio. Nesta etapa, seleciona-se conceitos listados anteriormente. Os conceitos selecionados são as classes da ontologia e orientam a hierarquia.

#### 5. Definir as propriedades de classes.

Esta fase define atributos das classes e visa à estruturação interna dos conceitos, necessária para satisfazer os requisitos de informação do cenário em desenvolvimento. Uma classe pode ter qualquer número de atributos ou nenhum atributo. Para cada atributo da lista, deve-se determinar a que classes pertence.

Conforme exposto em [Horridge 2009], em OWL há dois tipos principais de propriedades, propriedades de objeto e propriedades de tipos de dados. Propriedades de objetos são relacionamentos entre dois indivíduos; já propriedades de dados relacionam indivíduos a valores de dados, que pode ser um tipo de dado pertencente ao XML Schema ou um literal RDF. Existe ainda um terceiro tipo de propriedades, propriedades de anotações, que são usadas para adicionar informações, ou seja, metadados, para classes, indivíduos ou propriedades. Conforme [Horridge 2009], algumas características podem ser associadas às propriedades, aumentando a expressividade e a capacidade de inferência. Assim, uma propriedade de objeto pode ser caracterizada como:

- Inversa: se alguma propriedade liga o indivíduo *a* ao indivíduo *b*, então sua propriedade inversa liga o indivíduo *b* ao indivíduo *a*. Por exemplo, a propriedade *temPai* é inversa à propriedade *temFilho*.
- Funcional: que permite a associação de um único valor à propriedade para um determinado indivíduo. Por exemplo, a propriedade *temMãe* é funcional. Essa

característica pode ser aplicada também às propriedades de tipos de dados.

- Transitiva: se uma propriedade  $P$  é transitiva e relaciona o indivíduo  $a$  ao indivíduo  $b$  e também o indivíduo  $b$  ao indivíduo  $c$ , então infere-se que o indivíduo  $a$  é relacionado ao indivíduo  $c$  pela propriedade  $p$ . Por exemplo, a propriedade *temAncestral* é transitiva para indivíduos das classes *filho* e *neto*.
- Simétrica: se uma propriedade  $P$  é simétrica e relaciona o indivíduo  $a$  ao indivíduo  $b$ , então também relaciona o indivíduo  $b$  ao indivíduo  $a$ . Por exemplo, a propriedade *temIrmão* é simétrica.
- Antissimétrica: o inverso da simétrica, ou seja, se a propriedade  $P$  é antissimétrica e relaciona o indivíduo  $a$  ao indivíduo  $b$ , então o indivíduo  $b$  não pode ser relacionado ao indivíduo  $a$  pela propriedade  $P$ . Por exemplo, a propriedade *éFilhoDe* é antissimétrica.
- Reflexiva: relaciona o indivíduo  $a$  a ele mesmo. Por exemplo, a propriedade *conhece* pode ser reflexiva, informando assim que o indivíduo conhece a ele mesmo. Logicamente, ele pode conhecer outros indivíduos.

#### 6. Definir as restrições (facetas) das propriedades.

Esta fase corresponde à definição das restrições associadas às propriedades. Uma restrição descreve uma classe de indivíduos baseando-se nos relacionamentos dos quais os membros das classes participam. Conforme [Horridge 2009], as restrições em OWL classificam-se em três categorias principais:

- Restrições quantificadoras, categorizadas em restrições *existenciais*, que descrevem classes de indivíduos que se relacionam com *pelo menos um* indivíduo de outra classe, por meio de uma propriedade específica, ou restrições *universais*, que descrevem classes de indivíduos que, para uma determinada propriedade, *somente* tem relacionamentos com indivíduos de uma determinada classe.
- Restrições de cardinalidade: especifica a cardinalidade mínima, máxima ou exata de relacionamentos de uma classe de indivíduos com outros indivíduos ou tipos de dados.
- Restrições de valores: define os tipos de valores que um atributo pode conter, como por exemplo valor numérico ou booleano. Uma propriedade pode ser ainda um tipo de instância de uma classe.

Em OWL, propriedades são usadas para criar restrições e estas são associadas aos quantificadores para restringir os indivíduos que pertencem a uma classe. As restrições definem as instâncias que podem pertencer a uma classe através de propriedades que as relacionam com instâncias de outras classes. Por exemplo, a declaração *hasBase some PizzaBase*, associada à classe *Pizza*, expressa que instâncias da classe

*Pizza* se relacionam com instâncias da classe *PizzaBase* através da propriedade *hasBase*. O termo *some*, usado na restrição, identifica o quantificador existencial *someValuesFrom*, que restringe uma propriedade para uma classe.

As classes sobre as quais são aplicadas restrições podem ser de dois tipos, em razão do tipo de condições associadas a essas restrições [Horridge 2009]:

- Classes Primitivas - uma classe é Primitiva se as restrições sobre seus indivíduos são apenas condições do tipo *Necessária*. Condições necessárias podem ser entendidas como: se alguma coisa é membro desta classe então é necessário que satisfaça estas condições. Diz-se que uma classe primitiva é apenas descrita. Assim, descrever uma classe (Primitiva) é declarar restrições usando condições necessárias. No exemplo citado, se a restrição é descrita como necessária, isso implica que, se algo é instância da classe *Pizza*, então é necessário que tenha, pelo menos, uma base do tipo *PizzaBase*.
- Classes Definidas - uma classe é Definida se as restrições sobre seus indivíduos são condições do tipo *Necessária e Suficiente*. No exemplo citado, se a restrição é descrita como necessária e suficiente, isso implica que se um indivíduo qualquer é relacionado com a classe *PizzaBase* através da propriedade *hasBase*, então ele é uma instância da classe *Pizza*. Ou seja, essa condição é suficiente para inferir o seu tipo.

#### 7. Criar instâncias.

O último passo é criar instâncias de classes. Definir uma instância requer (1) escolher a classe, (2) criar uma instância/indivíduo daquela classe, e (3) completar os valores das propriedades.

### Linguagens e ferramentas para ontologias

Existem diversas linguagens e ferramentas destinadas à área de ontologias, como, modelos de dados, linguagens de especificação e intercâmbio, descritores, linguagens de consultas e regras, editores, mecanismos de inferência, ambientes e *frameworks*. RDF *Schema* e OWL são algumas das linguagens relativas às ontologias. Em paralelo com estas linguagens existem outras para elaboração de consultas e definição regras, que permitem extrair informação das ontologias. A linguagem padrão estabelecida pela W3C para construção de ontologias é a OWL, que se baseia na lógica descritiva e possui um poder de expressividade maior que RDF *Schema*. Em OWL, por exemplo, é possível definir classes disjuntas, restrições de cardinalidade, propriedades transitivas, inversas e simétricas; o que não é possível com RDF *Schema*, conforme [Dean e G. Schreiber 2004].

## OWL (Web Ontology Language)

OWL é a linguagem para definir e instanciar ontologias recomendada pelo W3C para o desenvolvimento da Web Semântica. OWL baseia-se em RDF e RDF *Schema* e complementa o vocabulário para descrever propriedades e classes, como relações entre classes (disjunção, por exemplo), cardinalidade, características de propriedades (simetria, por exemplo), sendo projetada para uso em aplicações que precisam processar o conteúdo da informação em vez de apenas apresentá-lo.

Há duas versões de OWL. Conforme [MacGuinnes e F. Harmelen 2004], a OWL 1 fornece três sublinguagens com níveis de representatividade diferentes. São elas:

- OWL Lite: fornece os mecanismos básicos para classificação dos termos em hierarquias e para a definição de restrições sobre eles.
- OWL DL: possibilita uma representatividade completa da OWL, pois inclui todas as construções, além de garantir completude (todas as conclusões são computáveis) e decidibilidade (todas as computações terminarão em um tempo finito).
- OWL FULL: possui os mesmos construtores da OWL DL, mas oferece uma sintaxe mais flexível, o que não garante que as ontologias desenvolvidas através de seus recursos sejam completas e decidíveis.

Já a OWL 2 também fornece três diferentes sublinguagens - OWL 2 EL, OWL 2 QL, and OWL 2 RL - denominadas perfis (*profiles*), cuja classificação baseia-se principalmente no domínio de aplicação e não em sua computabilidade. Para cada perfil é definida uma restrição sintática diferente, com diferentes aspectos do poder expressivo de OWL, em virtude de benefícios computacionais e de implementação. OWL 2 EL permite a implementação de algoritmos de tempo polinomial para todas as tarefas de raciocínio padrão, que é particularmente adequado para aplicações que necessitam de grandes ontologias e o poder expressivo pode ser trocado por garantias de desempenho. OWL 2 QL permite que consultas sejam respondidas em *LogSpace* utilizando a tecnologia de banco de dados relacional padrão, o que é particularmente adequado para aplicações em que ontologias relativamente leves são usadas para organizar um grande número de indivíduos e onde é útil ou necessário acessar diretamente os dados através de consultas relacionais. OWL 2 RL permite a implementação de algoritmos de raciocínio em tempo polinomial usando tecnologias de banco de dados estendido a regras para operar diretamente sobre triplas RDF, que é particularmente adequado para aplicações onde ontologias relativamente leves são usados para organizar um grande número de indivíduos e onde é útil ou necessário operar diretamente sobre os dados sob a forma de triplas RDF.

Em virtude das ferramentas utilizadas neste trabalho para a modelagem e manipulação das ontologias, além de sua estabilidade e capacidade de suprir as necessidades de representação exigida, neste trabalho foi utilizada a OWL 1, mais especificamente a OWL

DL, cujas especificações estão disponíveis em [MacGuinnes e F. Harmelen 2004], [Smith et al. 2004] e [Dean e G. Schreiber 2004].

Somente uma linguagem robusta e completa não é suficiente para que ontologias úteis possam ser desenvolvidas. [Gruber 1993] resalta os seguintes cuidados durante o projeto de uma ontologia para atingir os resultados esperados:

- Clareza - uma ontologia deve comunicar efetivamente o significado pretendido para seus termos, por meio de definições objetivas e bem documentadas;
- Coerência - os axiomas devem ser logicamente consistentes e as inferências geradas não podem contradizer a documentação em linguagem natural;
- Extensível - o conteúdo de uma ontologia deve ser projetado de forma que ela possa ser atualizada monotonicamente, ou seja, novos termos poderão ser definidos sem que os já existentes precisem ser revistos;
- Limiar de Codificação Mínimo - a conceitualização deve ser especificada no nível do conhecimento, sem depender de uma linguagem específica;
- Compromisso Ontológico Mínimo - uma ontologia deve definir apenas os termos extremamente necessários para que as informações possam ser compartilhadas. As demais devem ser especializadas por cada um dos interessados.

Conforme [Neto 2006], é de fundamental importância que toda ontologia tenha suas definições claramente descritas para que outras pessoas possam reutilizá-las. Além disto, o projeto deve prever a inclusão de novas informações sem que haja necessidade de alterar o que já está implementado. Algumas instruções OWL auxiliam a tarefa de união de ontologias, como:

- *owl:equivalentClass* - define que uma classe definida em uma ontologia possui as mesmas características de outra classe já existente em outra ontologia;
- *owl:equivalentProperty* - funciona de forma semelhante ao comando anterior, sendo que sua abrangência é apenas para uma propriedade;
- *owl:sameAs* - define a equivalência entre instâncias;
- *owl:differentFrom* - possui efeito oposto ao comando *sameAs*.

A criação de ontologias interoperáveis economizará esforços e facilitará a construção de bases de conhecimento mais amplas e detalhadas, que poderão ser utilizada tanto para recuperação de informações pelos agentes de softwares quanto para um alicerce ao desenvolvimento de sistemas inteligentes para diversas áreas do conhecimento, incluindo o *e-learning*.

## Protégé: ferramenta para desenvolvimento de ontologias

Conforme disposto em [for Biomedical Informatics Research 2010a], Protégé é uma plataforma *open-source* que oferece um conjunto de ferramentas para a construção de modelos de domínio e aplicações baseadas no conhecimento com ontologias. Protégé implementa um conjunto de conhecimentos de modelagem de estruturas e ações que suportam a criação, visualização e manipulação de ontologias em vários formatos de representação. O Protégé é baseado em Java, expansível e fornece um ambiente *plug-and-play*, tornando-se uma base flexível para uma rápida prototipagem e desenvolvimento de aplicações. A ferramenta possui as seguintes características, conforme disposto em [Wiki 2010]:

- Visões múltiplas de uma mesma ontologia.
- Suporte a construção de ontologias em OWL.
- Facilidade de modularização, ao utilizar repositórios globais ou locais para manipular dependências de importação.
- Área de trabalho simples.
- Suporte a raciocínio: axiomas inferidos, interface para raciocinadores FaCT++ e Pellet, entre outros que podem ser acrescentados por meio de *plugins*.
- Edição de regras em SWRL (*Semantic Web Rule Language*).
- Arquitetura plugável, com suporte para *plugins* que permitem visualização gráfica de ontologias, integração com mecanismos de regras, consulta e criação de instâncias, entre outros recursos.

Além disso, a ferramenta possui uma comunidade ativa, conforme disposto em [for Biomedical Informatics Research 2010a], com milhares de usuários registrados <sup>1</sup>, e uma ampla documentação, tanto para usuários comuns quanto para desenvolvedores, disponível em [for Biomedical Informatics Research 2010b]. Há duas versões disponíveis da ferramenta: a versão 3.4 e a versão 4. Neste trabalho utilizou-se a versão mais antiga, pois, apesar de não suportar OWL 2, é uma plataforma com mais recursos em termos de conjunto de *plugins* disponíveis. Como disposto em [Wiki 2010], em termos gerais, a ferramenta Protégé pode ser utilizada para:

- Modelagem de classes: por meio de uma interface gráfica e intuitiva é possível modelar as classes, seus atributos e relacionamentos.
- Criação e edição de instâncias: a ferramenta gera, automaticamente, formulários interativos para as classes permitindo entrar com instâncias válidas.
- Processamento do modelo: por meio de uma biblioteca de *plugins* pode-se definir semântica, buscas e comportamento lógico de uma aplicação.

---

<sup>1</sup>153.643 usuários registrados em 04/08/2010. Disponível em <http://protege.stanford.edu>

- Compartilhamento de modelos: as classes e instâncias podem ser carregadas e salvas em vários formatos, como XML, UML e RDF, ou em um banco de dados.

Outra característica importante, na perspectiva dos desenvolvedores, é que o Protégé fornece uma API de código aberto para associar componentes Java e acessar os modelos definidos. Essa API permite ainda a integração com outras ferramentas, como o *framework* Jena, para manipulação do modelo OWL, e raciocionadores, como Pellet, para inferência, conforme exposto em [Knublauch 2010]. Como resultado, pode-se desenvolver sistemas mais rapidamente, começando com o modelo de domínio subjacente e utilizando o Protégé para gerar a interface básica com o usuário, e então, gradualmente, desenvolvendo aplicações que manipulem esse modelo, utilizando a Protégé-OWL API.

Por essas características, que o tornam atualmente o principal editor de ontologias disponível, essa ferramenta foi utilizada para o desenvolvimento e manipulação das ontologias descritas na seção 4.3.2.

### Regras para a web semântica

Os assuntos abordados no presente capítulo constituem os níveis inferiores e intermediários da web semântica e são relacionados à representação do conhecimento, seja ele sobre os conteúdos disponíveis na web ou sobre conceitos de qualquer domínio do conhecimento. Conforme [Neto 2006], as linguagens utilizadas para representar o conhecimento na web semântica (RDF, RDF *Schema* e OWL) podem ser vistas como especializações da lógica de predicados que definem um subconjunto desta lógica, a lógica descritiva, adequada para a construção de linguagens baseadas em *tags*.

Com o objetivo de tornar mais representativas as linguagens utilizadas na web semântica, e de permitir a interoperabilidade na definição de regras, foi desenvolvida a *Semantic Web Rule Language* - Linguagem de Regras para a Web Semântica (SWRL), que é uma combinação das sublinguagens OWL-DL e OWL-Lite com a sublinguagem Unary/Binary Datalog RuleML [Horrocks et al. 2004]. A SWRL estende o conjunto de axiomas existentes na linguagem OWL, permitindo a codificação de cláusulas de Horn e possibilitando a construção de regras sobre o conhecimento descrito na ontologia através das classes, propriedades e instâncias.

Há um editor na ferramenta Protégé para SWRL, que permite aos utilizadores alternar facilmente entre a edição de regras SWRL e a edição normal das ontologias em OWL e, ainda, incorporar as regras nas entidades em OWL. Segundo [O'Connor et al. 2005], um dos principais objetivos do Editor SWRL é permitir a interoperabilidade entre SWRL e motores de regra existentes, por meio de um subsistema denominado Protégé SWRL Factory. Por essas características, a linguagem SWRL foi utilizada para definição das regras de adaptação.

### 3.3 Considerações

Neste capítulo foram apresentadas as tecnologias presentes nas camadas da web semântica, projeto que visa à evolução da web atual, de forma a transformá-la em uma web de dados, de forma que dispositivos e aplicações possam compartilhar e processar informações. Assim percebe-se que algumas das tecnologias utilizadas já estão consolidadas, sendo recomendadas como padrão há muito tempo, como o URI, o Unicode e o XML. Outras foram desenvolvidas recentemente, mas já estão consolidadas, como RDF e OWL. Ao passo que outras ainda são recentes e estão sendo implementadas, como os dialetos de regras propostos pelo RIF.

O estudo das tecnologias da web semântica evidenciou a importância das ontologias para a representação do conhecimento. As ontologias permitem que vocabulários padronizados sejam definidos, podendo ser utilizadas para inferências e manipuladas por diversas aplicações. Com base nesse estudo foi possível definir as tecnologias e ferramentas utilizadas para o desenvolvimento do sistema adaptativo, cuja arquitetura baseia-se em ontologias. No caso desse trabalho, as ontologias foram desenvolvidas utilizando-se a linguagem OWL DL, por meio da ferramenta Protégé. Além de permitir a modelagem das ontologias, essa ferramenta disponibiliza uma API para processar os modelos em OWL, a Protégé-OWL API, que trabalha em conjunto com outras ferramentas, como o *framework* Jena e raciocinadores e mecanismos de regras, e, por isso, foi utilizada neste trabalho.

Pode-se perceber, nesse capítulo, que é necessário a utilização de muitas tecnologias para projetar um sistema com suporte à web semântica. Apesar disso, grande parte dessas tecnologias, especialmente aquelas pertencentes às camadas inferiores, são populares, e, além disso, já existem ferramentas que facilitam a modelagem de ontologias e que dão um bom suporte ao desenvolvimento de aplicações para a web semântica. As tecnologias possuem ainda uma boa documentação para consulta, já que a maioria das tecnologias apresentadas já são padrões recomendados pela W3C, o que facilita o desenvolvimento de aplicações.

## Capítulo 4

# Adaptive-Moodle: Adaptatividade e Interoperabilidade em Ambientes de *e-learning*

Conforme descrito na seção 1.2, os principais objetivos deste trabalho são a integração de ambientes de *e-learning* a sistemas adaptativos existentes e o desenvolvimento de um sistema adaptativo baseado em tecnologias da web semântica, que também se integre a ambientes de *e-learning* e possibilite-lhes adaptatividade e interoperabilidade. Dessa forma, espera-se que os ambientes de *e-learning* sejam capazes de se adequar aos diferentes perfis de usuário, assim como compartilhar informações, recursos, atividades e objetos de aprendizagem.

Neste capítulo, são apresentados os sistemas desenvolvidos, detalhando os requisitos, a arquitetura proposta, as ontologias e as atividades executadas pelo agente de integração e pelo sistema adaptativo. Inicialmente, foi projetado um agente de software que integrou o ambiente de *e-learning* Moodle ao sistema adaptativo SIMEDUC, constituído por uma plataforma de agentes trabalhando de forma cooperativa. Implementou-se a integração em primeiro lugar para verificar a viabilidade de, com base nas informações presentes nos ambientes de *e-learning*, integrá-los a sistemas adaptativos. Após comprovar a viabilidade dessa integração, foi projetado e desenvolvido um sistema adaptativo que se baseia em padrões para *e-learning* e em estilos de aprendizagem e utiliza tecnologias da web semântica, de forma a obter interoperabilidade.

Assim, este capítulo descreve as atividades envolvidas no desenvolvimento dos sistemas propostos, sendo estruturado da seguinte forma. Na seção 4.1 é apresentada a arquitetura proposta, denominada *Adaptive-Moodle*, que engloba todos os módulos envolvidos, ou seja, o ambiente de *e-learning* Moodle, o módulo de integração, o sistema adaptativo baseado em ontologias e o sistema adaptativo SIMEDUC. As seções seguintes descrevem detalhadamente as atividades envolvidas nesses módulos. Na seção 4.2 descreve-se o pro-

cesso de integração entre os sistemas, sendo que na seção 4.2.1, descreve-se o ambiente Moodle com o objetivo de mostrar as informações utilizadas no desenvolvimento do módulo de integração, assim como para processamento do sistema adaptativo. Na seção 4.2.2 é apresentado o Agente de Integração, descrevendo-se detalhadamente as atividades envolvidas no processo. Na seção 4.3 apresenta-se o sistema adaptativo, mostrando, em primeiro lugar, uma visão geral do processo de adaptação e, em seguida, as ontologias, a arquitetura e funcionamento do sistema, e, por fim, as regras de adaptação utilizadas. Na seção 4.4 são apresentados os testes realizados para verificar e validar o funcionamento do sistema.

## 4.1 Arquitetura do Adaptive-Moodle

Entre os requisitos técnicos, procurou-se implementar um sistema independente de plataforma, que permitisse uma fácil integração com ambientes de *e-learning*, utilizando os padrões LOM e SCORM, e que apresentasse um fraco acoplamento com as funcionalidades dos módulos envolvidos, de tal forma que os interessados poderão optar por utilizar os recursos de adaptação em determinados cursos, mantendo intacta a estrutura de cursos já inseridos no ambiente de *e-learning*. Dessa forma, espera-se facilitar a adoção dessa proposta, já que não exigirá esforços no sentido de modificar a estrutura de cursos existentes ou utilizar uma nova plataforma.

Como justificado na seção 2.2, o ambiente selecionado para integração foi o Moodle. Realizou-se inicialmente um trabalho de análise desse ambiente para verificar o seu funcionamento e as informações que poderiam ser utilizadas para integração com os sistemas adaptativos. Nem todas as informações que, em geral, são necessárias para o processamento dos componentes dos sistemas adaptativos estão explicitamente disponíveis nos ambientes de *e-learning*, mas, conforme descrito posteriormente, é possível obtê-las ou adaptá-las e, assim, estender a base de dados dos ambientes para armazenar estas informações.

A partir das informações levantadas na análise desse ambiente, em especial, aquelas presentes nos componentes apresentados na Figura 4.1, para atender aos requisitos estabelecidos e comprovar a viabilidade de se integrar sistemas adaptativos a esse ambiente, definiu-se o módulo de integração, que utiliza a tecnologia de sistemas multiagentes.

O agente de integração, a partir de parâmetros definidos, como os cursos que serão adaptados, monitora e efetua as alterações necessárias ao ambiente, para que ele se comporte de forma adaptativa. Inicialmente, o Moodle foi integrado ao sistema adaptativo SIMEDUC, cujas regras foram modificadas para retornar uma organização de um pacote SCORM adequada ao estudante, após os resultados de avaliação. Posteriormente, o sistema adaptativo baseado em ontologias, cuja adaptação ocorre no nível dos objetos de aprendizagem, também foi integrado ao ambiente Moodle. Para instanciar as ontologias e

processar as regras para adaptação, foi utilizado a Protégé-OWL API, que integra-se com outras ferramentas, como *framework* JENA e raciocinadores. Por meio dessas ferramentas é possível mapear as classes das ontologias em classes Java, além de usar mecanismos de inferência e sistemas de regras. Assim, os componentes da arquitetura proposta estão representados na figura 4.1:

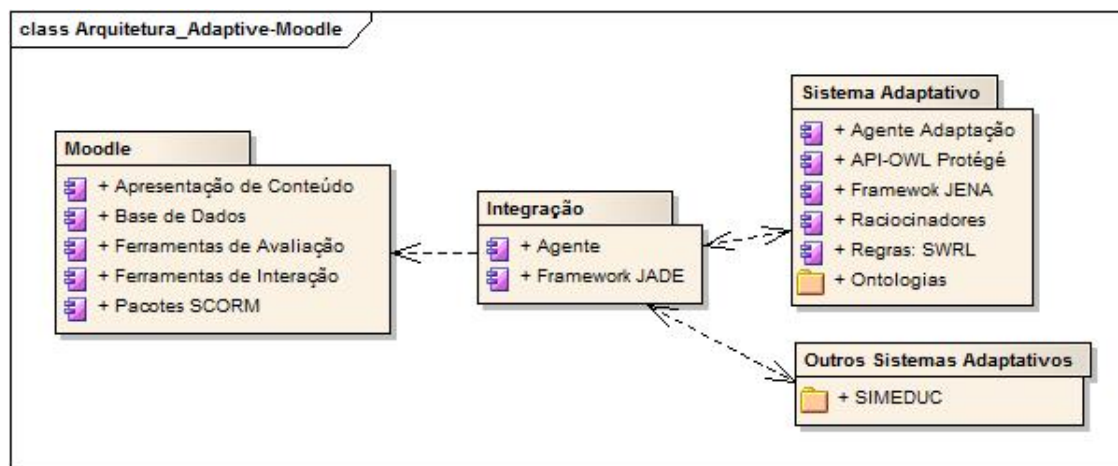


Figura 4.1: Arquitetura do Adaptive-Moodle

Conforme mostra a figura 4.1, a arquitetura proposta procura diminuir o acoplamento entre os sistemas. O ambiente de *e-learning* não apresenta nenhuma dependência em relação aos outros sistemas. Essa característica é importante, já que, para as instituições que disponibilizam cursos no ambiente, seria difícil atualizar todos os cursos para utilizar uma nova plataforma. Os sistemas adaptativos dependem dos parâmetros estabelecidos pelo sistema de integração para processar as regras de adaptação, especialmente aqueles relacionados à avaliação do estudante; além disso, enviam os resultados após o processamento das regras para o agente de integração que realiza as alterações necessárias para que o ambiente de *e-learning* comporte-se de forma adaptativa.

Assim, o sistema de integração, que interage com o ambiente de *e-learning*, buscando as informações dos cursos a serem adaptados, e com os sistemas adaptativos, fornecendo as informações necessárias ao seu processamento, mantém uma relação de dependência com ambos. Conforme mostrado na figura 4.1, o agente de integração utiliza tanto o sistema adaptativo SIMEDUC, quanto o sistema adaptativo baseado em ontologias. Percebe-se, dessa forma, que diferentes sistemas adaptativos podem colaborar na tarefa de adaptação, tornando-a mais adequada, ao possibilitar que mais opções e aspectos do processo de ensino-aprendizagem sejam considerados.

Após análise do ambiente Moodle, descrita na seção 4.2.1, relacionando as informações presentes no ambiente com os componentes de um sistema adaptativo, foram definidas as classes utilizadas para modelar as ontologias do sistema adaptativo. A figura 4.2 apresenta essas entidades e seus relacionamentos.

No diagrama mostrado na figura 4.2 pode-se perceber o Modelo de Domínio, que in-

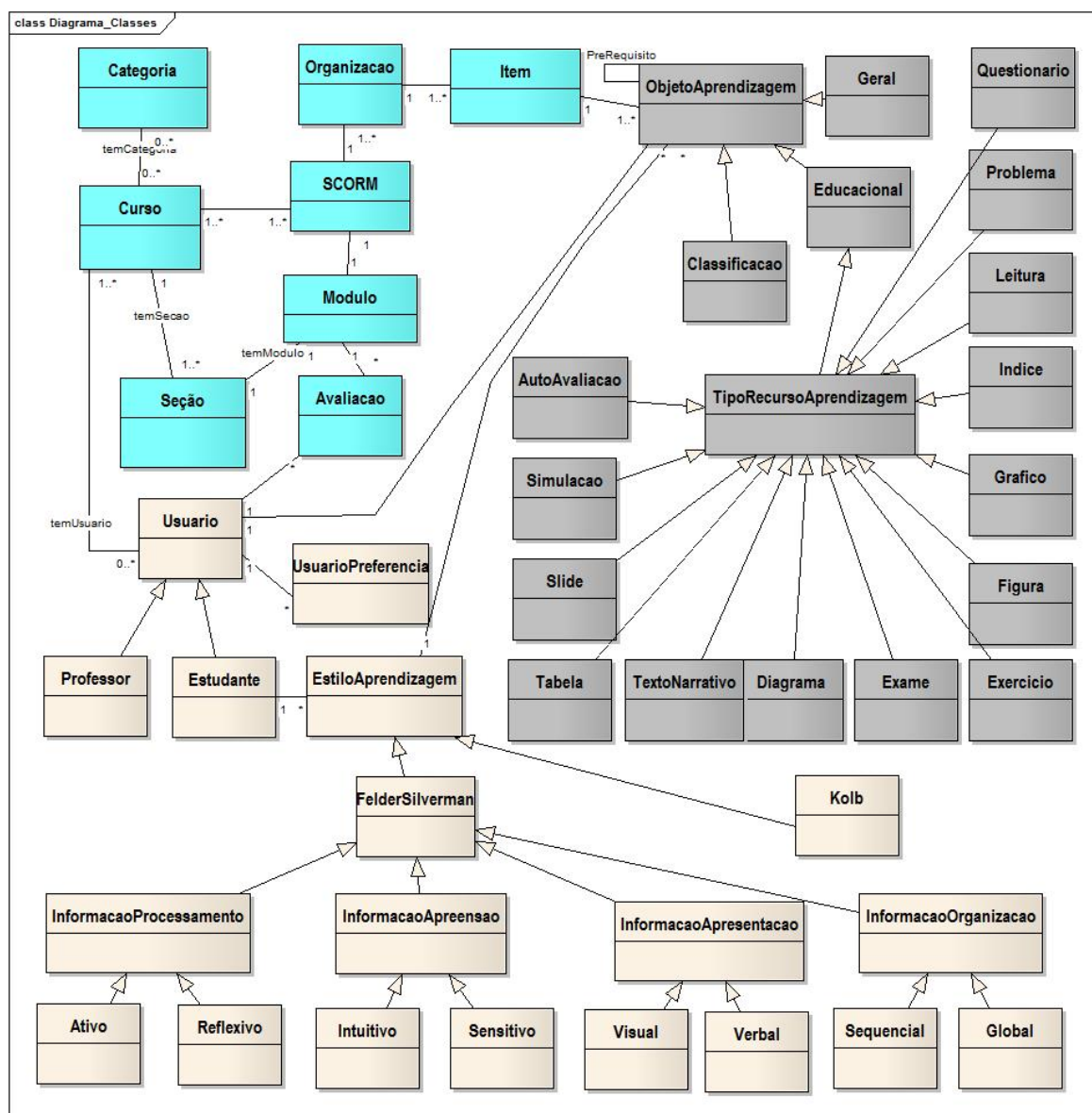


Figura 4.2: Classes do Sistema Adaptativo

clui as entidades básicas: categoria, curso, seção e módulo. Um curso é composto por seções, chamados de tópicos no ambiente Moodle, que por sua vez possuem módulos, que podem ser atividades como fórum e *chat*. Os módulos incluem pacotes SCORM, para estruturação do conteúdo, e questionários, para avaliação do conhecimento. Os pacotes SCORM podem conter diferentes organizações, com diferentes níveis de dificuldade; as organizações, por sua vez, poderão abranger diversos itens. Aos itens poderão ser associados diversos recursos, que correspondem aos objetos de aprendizagem, anotados conforme os metadados do padrão LOM.

O Modelo do Estudante possui as informações dos usuários, assim como a associação de estilos de aprendizagem ao estudante, de forma a possibilitar alternativas pedagógicas. No caso, foi adotada a classificação de estilos proposta por [Felder e Silverman 1988]; no entanto, procurou-se projetar um modelo flexível, que possibilite a inclusão de ou-

tras propostas; no caso, por exemplo, poderia ser acrescentado o modelo de estilos de aprendizagem proposto por [Kolb 1999].

Assim, as regras de adaptação podem se basear tanto no comportamento do estudante, em atividades como fóruns e *chats*, assim como avaliar seu conhecimento, de forma que o sistema apresente diferentes organizações, conforme o grau de dificuldade identificado, e ou associando diferentes objetos de aprendizagem aos itens, de forma que o sistema possa apresentar aqueles mais adequados ao estilo de aprendizagem do estudante.

## 4.2 Integração entre Ambientes de *e-Learning* e Sistemas Adaptativos

Esta seção descreve, em primeiro lugar, a relação entre o ambiente de *e-learning* Moodle e os componentes de um sistema adaptativo. Apesar de focar especificamente em um ambiente de *e-learning*, acredita-se que seja possível utilizar a proposta para outros ambientes que suportem o padrão SCORM, já que as informações levantadas, ou pelo menos grande parte delas, podem ser encontradas em qualquer ambiente. Obviamente, essa possibilidade deve ser demonstrada utilizando-se um outro ambiente de *e-learning*, o que não faz parte do escopo do presente trabalho.

Após a descrição do Moodle, descreve-se o Agente de Integração implementado para integrar o ambiente ao sistema adaptativo SIMEDUC. Esse mesmo agente também foi projetado de forma genérica, de forma a permitir a integração a outros sistemas adaptativos.

### 4.2.1 Moodle e Sistemas Adaptativos

Serão apresentadas, nesta seção, as informações existentes no ambiente Moodle que podem ser relacionadas aos componentes dos sistemas adaptativos, assim como aquelas que não estão presentes explicitamente na base de dados do ambiente, mas que podem ser obtidas ou acrescentadas de forma a permitir o funcionamento sem restrições de um sistema adaptativo. Para uma descrição mais clara, as informações obtidas foram associadas aos componentes Modelo do Domínio e Modelo do Estudante de um sistema adaptativo.

#### Informações relacionadas ao componente Modelo do Domínio

O Moodle permite o cadastro de cursos em vários formatos. O formato mais comum se baseia em tópicos e, para cada tópico, podem ser associados diversos recursos ou atividades. Um pacote SCORM corresponde a uma atividade de um curso. É possível ainda definir um curso no formato SCORM, sendo que, dessa forma, um único pacote corresponde a um curso completo. Os dados presentes na base de dados do Moodle que

associam-se diretamente ao componente Modelo do Domínio são apresentadas na Tabela 4.1.

Tabela 4.1: Dados relacionados ao Modelo de Domínio.

| Tabela            | Descrição                                                                                                                                                                                                                                                                                                           |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| course            | Informações sobre os cursos.                                                                                                                                                                                                                                                                                        |
| category          | Categorias de cursos.                                                                                                                                                                                                                                                                                               |
| section           | Corresponde a um tópico no ambiente.                                                                                                                                                                                                                                                                                |
| module            | Recursos e atividades que podem ser associados às seções (tópicos).                                                                                                                                                                                                                                                 |
| resource          | Relação de recursos que podem ser associados aos cursos.                                                                                                                                                                                                                                                            |
| course_module     | Identificação dos módulos e recursos/atividades associados a um curso.                                                                                                                                                                                                                                              |
| scorm             | Informações sobre os pacotes no padrão SCORM importados para o ambiente. Além de armazenar os dados nessa tabela, os arquivos do pacote SCORM, inclusive o <i>imsmanifest.xml</i> , arquivo que descreve a estrutura geral do pacote, são armazenados no diretório padrão para armazenamento de arquivos do Moodle. |
| scorm_scoes       | Informações sobre os SCOs ( <i>Shareable Content Object</i> ), que correspondem aos objetos de aprendizagem presentes em cada pacote SCORM. Representa a estrutura de organizações e itens dentro de um pacote.                                                                                                     |
| scorm_scoes_track | Rastreamento de valores do modelo de dados do SCORM para cada tentativa do usuário em cada SCO.                                                                                                                                                                                                                     |

O sistema de avaliação, ou seja, a forma como são avaliadas as atividades no decorrer de um curso, também se relaciona ao Modelo de Domínio de um sistema adaptativo. No caso do Moodle, qualquer atividade disponível no ambiente pode ser avaliada. Neste trabalho, tendo em vista o funcionamento do sistema adaptativo SIMEDUC, ao qual o ambiente foi integrado, restringiu-se a avaliação, inicialmente, às seguintes atividades: *chat* e fórum, com a finalidade de avaliar o comportamento dos estudantes, e questionário, para a avaliação do conhecimento do estudante em cada tópico. Obviamente, pode-se incluir qualquer outra atividade disponível no ambiente, para resultar em determinada forma de avaliação utilizada por um sistema adaptativo específico para o processamento de regras de adaptação. Os dados que representam as atividades a serem avaliadas são descritas na Tabela 4.2.

### Informações relacionadas ao componente Modelo do Estudante

Na tabela 4.3 são mostradas as informações presentes na base de dados do Moodle que se relacionam ao componente Modelo do Estudante. Conforme a relação apresentada na tabela 4.3, além das informações referentes aos usuários, foram obtidas também as informações referentes às permissões concedidas. Em [Moodle 2010a] é apresentado um diagrama que mostra essas entidades e seus relacionamentos.

Tabela 4.2: Dados relacionados às atividades a serem avaliadas.

| Tabela              | Descrição                                                                                                            |
|---------------------|----------------------------------------------------------------------------------------------------------------------|
| chat                | Informações sobre os chats criados, como código do curso, descrição, data de criação e de modificação.               |
| chat_messages       | Registro das mensagens enviadas pelos usuários: contém o código do usuário e do chat, a data e a mensagem enviada.   |
| forum               | Informações sobre os fóruns, como: curso, nome, descrição, data de modificação.                                      |
| forum_discussions   | Tópicos de discussão em determinado fórum. Contém informações como descrição, usuário, data alteração.               |
| forum_posts         | Postagens dos usuários para um tópico de discussão. Armazena dados como assunto, mensagem, anexo, data.              |
| forum_subscriptions | Usuários associados a um determinado fórum.                                                                          |
| quiz                | Informações sobre questionários. O relacionamento entre as tabelas dessa atividade está disponível em [Moodle 2010c] |

Tabela 4.3: Dados relacionados ao Modelo do Estudante.

| Tabela            | Descrição                                                                                                                                                             |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| user              | Cadastro de usuários com informações pessoais, username, senha, entre outros dados.                                                                                   |
| user_lastaccess   | Datas em que o usuário acessou um curso.                                                                                                                              |
| role              | Define um papel do usuário no ambiente, como administrador, professor, estudante.                                                                                     |
| capabilities      | As permissões que podem ser concedidas.                                                                                                                               |
| context           | Um contexto é um escopo de acesso, como, por exemplo, acesso a todo o sistema, ou a um curso, ou a apenas uma atividade. O campo <i>contextlevel</i> indica o escopo. |
| role_assignments  | As permissões concedidas aos usuários, ou seja, a associação entre usuários e seus contextos.                                                                         |
| role_capabilities | Permissões para cada <i>capacidade</i> em um determinado papel.                                                                                                       |
| role_names        | Usada para atribuir nomes diferentes a papéis de usuários em diferentes cursos.                                                                                       |

Os dados presentes nos ambientes de *e-learning* em relação ao Modelo do Estudante geralmente são limitados, ou seja, não possuem informações suficientes para a definição e o processamento de regras em um sistema adaptativo. Isso também acontece no Moodle, como pode-se perceber nas informações presentes na Tabela 4.3, em que são obtidas apenas informações pessoais, sobre as permissões concedidas e sobre o histórico de acesso a atividades, além dos dados sobre o desempenho dos usuários nas avaliações. Em grande parte dos sistemas adaptativos, outras informações são requeridas, como estilo de aprendizagem do estudante, questões de acessibilidade, como velocidade de conexão, deficiências físicas, entre outras. Para atender a essa necessidade, o modelo de dados dos ambientes pode ser estendido de forma a armazenar outras características.

Nesse intuito, foram acrescentadas algumas tabelas à base de dados do Moodle, tanto

para manter informações necessárias para o processo de adaptação, como os estilos de estudantes, quanto para o processo de integração, como informações sobre os cursos adaptados e os resultados dos estudantes. Poderia ter sido criada uma outra base de dados, à parte do ambiente, para armazenar essas informações. No entanto, preferiu-se utilizar a mesma base de dados por já ser uma abordagem recomendada para o desenvolvimento de novas funcionalidades para esse ambiente, além disso, evita-se a necessidade de configurar e acessar uma nova base de dados. Os dados acrescentados estão descritos na seção 4.2.2, que apresenta as atividades envolvidas no processo de integração.

## 4.2.2 Agente de Integração

De acordo com [Wooldridge e Jennings 1994], a autonomia e a capacidade de reação são características que identificam um agente. Um agente, portanto, atua em determinado ambiente, percebe as mudanças e, em resposta a tais mudanças ou simplesmente em decorrência da busca de seus próprios objetivos, modifica esse ambiente, sem necessariamente esperar por nenhum evento e geralmente em favor dos interesses de um usuário, de outro agente ou de outro sistema.

Assim, pela sua capacidade de responder a eventos sem intervenção direta do usuário, possibilitando o desenvolvimento de um sistema que atue de forma independente, baseando-se em mudanças percebidas, optou-se por desenvolver o sistema de integração utilizando-se a tecnologia de agentes.

De acordo com [Bolzan 2002], na área de Inteligência Artificial Distribuída (IAD) tem-se dedicado grande esforço na construção de ambientes para implementação/execução de agentes, de forma que a preocupação se concentre apenas nos aspectos que realmente envolvem questões específicas do projeto a ser desenvolvido, não sendo necessário a implementação das características necessárias aos agentes tais como autonomia, comunicação, cooperação e capacidade de reação a eventos do ambiente.

Desta forma, a escolha de uma plataforma apropriada para implementação/execução de um sistema multiagente oferece muitas vantagens, pois fornece toda a infraestrutura básica para manipulação de mensagens, comunicação, rastreamento, monitoramento e gerenciamento em tempo de execução. Dessa maneira, o processo de desenvolvimento pode ficar concentrado nas funcionalidades principais dos agentes e em como se dará a cooperação, ou seja, a resolução do problema real e nos aspectos cooperativos e sociais do sistema.

Dentre as plataformas para o desenvolvimento de sistemas multiagentes, o *framework* JADE (*Java Agent Development Environment*) destaca-se pela quantidade de recursos oferecidos, conforme descrito em [Jade 2010]. Tais recursos são frutos de uma implementação fiel aos padrões da FIPA para plataformas de agentes, especificando desde protocolos até recursos e serviços que devem ser fornecidos (como serviço de registro, nomes,

roteamento e páginas amarelas). O JADE oferece também uma gama de recursos para monitoração e controle de agentes remotos. Além disso, o sistema adaptativo SIMEDUC, que foi integrado ao ambiente de *e-learning* Moodle, utiliza essa plataforma de agentes. Assim, para facilitar essa integração, e pela quantidade de recursos presentes na plataforma JADE, optou-se por utilizá-la no desenvolvimento do Agente de Integração e do Agente de Adaptação, descrito posteriormente.

O diagrama de atividades da figura 4.3 apresenta as atividades realizadas pelo agente de integração.

Conforme descrito na figura 4.3, inicialmente, deve-se definir as configurações da plataforma a ser integrada em um arquivo de propriedades. Esse arquivo contém as configurações básicas para o funcionamento do agente, tais como parâmetros de conexão à base de dados, diretório para armazenamento de dados do ambiente de *e-learning*, assim como configurações específicas para a adaptação, a saber:

- **Códigos dos cursos a serem adaptados:** esse parâmetro permite diminuir o acoplamento entre o sistema adaptativo e o ambiente de *e-learning* utilizado, de forma que apenas os cursos especificados façam parte do processo de adaptação e os outros não sofram alterações, ou seja, permite tornar o ambiente de *e-learning* independente do sistema adaptativo. Pode-se, dessa forma, optar por utilizar os recursos de adaptação em determinados cursos, assim como manter a estrutura convencional de outros cursos já inseridos.
- **Outros parâmetros exigidos pelas regras de adaptação:** em uma primeira etapa, o ambiente Moodle foi integrado ao sistema adaptativo SIMEDUC. Assim, para o processamento das regras desse sistema, que se baseiam em alguns parâmetros que não são encontrados no Moodle, eles foram também inseridos no arquivo de propriedades: tempo máximo e mínimo de cada seção, quantidade de postagens máxima e mínima no fórum, quantidade de mensagens máxima e mínima no *chat* e peso dos elementos considerados na avaliação - no caso, tempo gasto em virtude do tempo ótimo, o desempenho no comportamento, no conhecimento e o desempenho total do estudante.

Após confirmar os parâmetros, o Agente de Integração passa a monitorar os dados do Moodle para verificar se algum novo estudante se matriculou nos cursos a adaptar e se existem avaliações registradas para esses cursos que ainda não foram processadas. Quando é localizada uma nova matrícula, o agente armazena as informações do novo curso e processa a mensagem *IniciarCurso* a partir da qual se inicia o processo de integração.

No início do processo é gerada uma cópia do curso para o estudante cuja matrícula é registrada. Esse curso compartilha as atividades do curso original, no entanto, os conteúdos apresentados pelos pacotes SCORM são específicos para cada usuário. Como cada pacote disponibiliza várias organizações, em grau crescente de complexidade, no primeiro

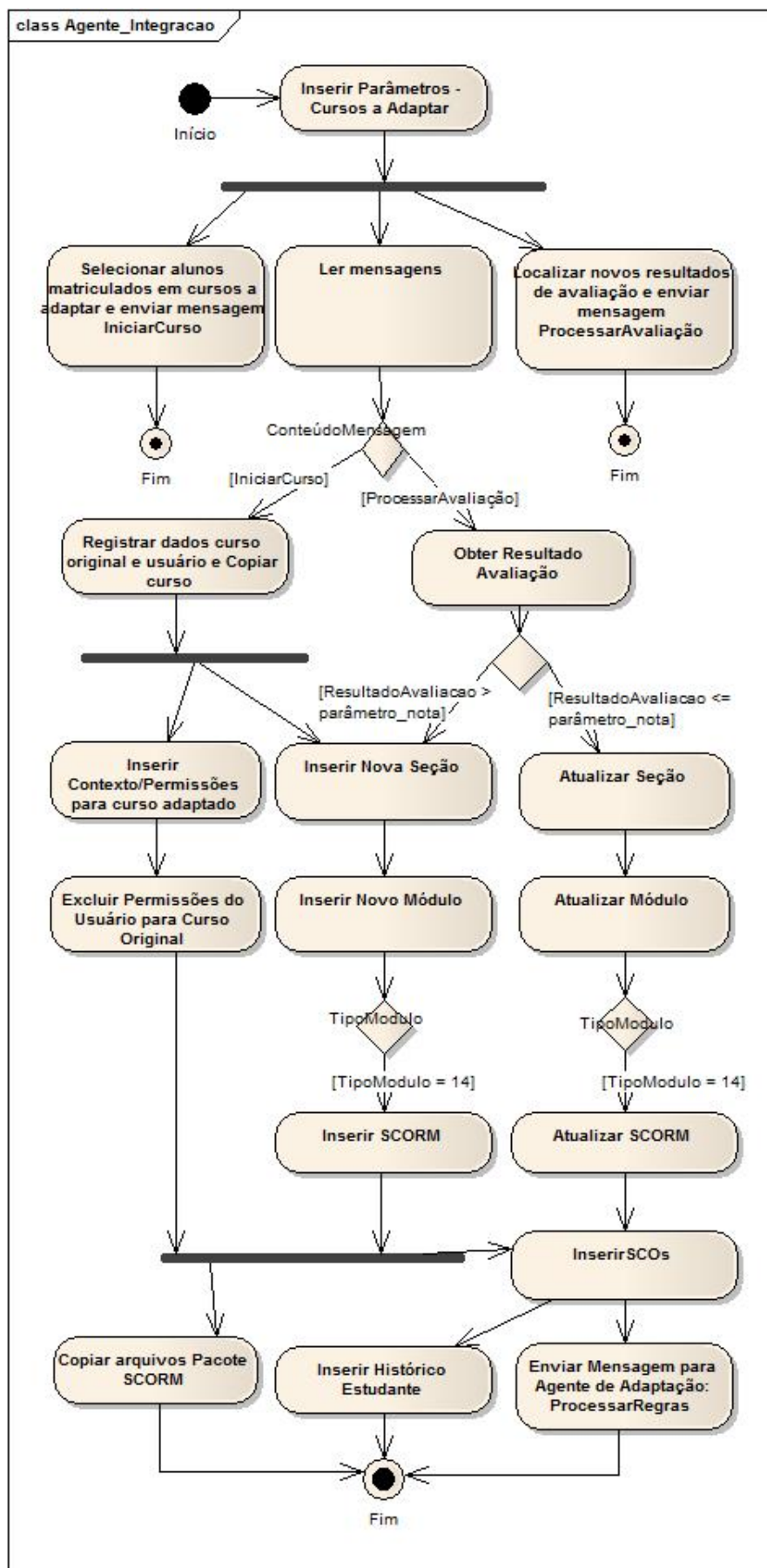


Figura 4.3: Atividades - Agente de Integração

contato, o sistema disponibiliza para o estudante uma organização de complexidade intermediária. A partir daí, novas seções de aprendizagem (que correspondem aos tópicos no Moodle), assim como a organização presente no pacote SCORM, serão disponibilizadas de acordo com o resultado da avaliação do estudante. Se o estudante obtiver um resultado ruim, terá que rever a mesma seção de aprendizagem, com uma organização de complexidade menor. Se obtiver um resultado bom, o sistema disponibiliza uma nova seção com uma organização de complexidade intermediária ou de uma complexidade maior. Após definir uma organização específica para determinado estudante, o Agente de Integração envia uma mensagem para o Agente de Adaptação iniciar o processamento das regras de adaptação, em que será associado a cada item da organização um objeto de aprendizagem adequado ao estilo de aprendizagem do estudante, conforme apresentado na seção 4.3.4.

Procurou-se, a partir do sistema proposto, implementar um sistema menos intrusivo possível, de forma que o ambiente original pudesse manter suas características. Alguns dados exigidos pelos sistemas adaptativos em geral são mais difíceis de serem obtidos a partir do Moodle, como é o caso do tempo ótimo para realização de uma determinada atividade. Nesse caso, flexibilizou-se o sistema para adotar uma das seguintes estratégias: i) o professor informa como parâmetros o tempo mínimo e o máximo para cada seção de aprendizagem do curso; ii) à medida que os estudantes acessam o curso, o sistema calcula o tempo máximo e mínimo em virtude do tempo médio de acesso. Se o professor informar o tempo, o sistema leva em consideração esse valor; caso contrário, o sistema efetua o cálculo.

Da mesma forma que houve a necessidade de definir uma estratégia para buscar o tempo máximo e mínimo de uma seção, outras informações exigidas por um determinado sistema adaptativo podem ser difíceis de serem encontradas, exigindo personalizações específicas para cada sistema. Um aspecto que pode ser melhorado é a avaliação do comportamento do estudante, que se baseia apenas em informações quantitativas, ou seja, no número de acessos ao fórum e ao *chat*. Deve-se, posteriormente, definir estratégias que procurem considerar não apenas a quantidade, mas também o conteúdo das mensagens.

Naturalmente, algumas modificações na base de dados foram necessárias para manter as informações dos cursos adaptados. Assim, foram acrescentadas à base de dados do Moodle as informações descritas na tabela 4.4. Também foram acrescentadas informações ampliar o Modelo do Estudante, permitindo a associação com estilos de aprendizagem e de objetos de aprendizagem apresentados ao estudante, conforme tabela 4.5.

Tabela 4.4: Tabelas acrescentadas à base de dados do Moodle referentes aos cursos adaptados.

|                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sim_curso_adaptado | Registro dos cursos que foram adaptados para os estudantes. Campos: <ul style="list-style-type: none"><li>• idCursoOriginal: código do curso original</li><li>• idCursoAdaptado: código do curso adaptado</li><li>• idUsuario: código do usuário</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| sim_mensagens_cms  | Mensagens processadas pelo agente de integração. Campos: <ul style="list-style-type: none"><li>• evento: Ex.: INICIOCURSO</li><li>• idCurso: código do curso.</li><li>• idUsuario: código do usuário.</li><li>• msgProcessada: <i>true</i> ou <i>false</i></li><li>• dataProcessamento: data em que a mensagem foi processada</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| sim_organizacoes   | Registro das organizações direcionadas para o estudante no início do curso ou após o resultado de uma avaliação. Campos: <ul style="list-style-type: none"><li>• idUsuario: código do usuário</li><li>• idCursoAdaptado: código do curso adaptado</li><li>• idCursoOriginal: código do curso original</li><li>• secao: seção disponibilizada para o estudante</li><li>• organizacao: organização para a qual o estudante foi direcionado.</li><li>• data: data de conclusão da seção</li><li>• scormOriginal: código do pacote SCORM original</li><li>• scormAdaptado: código do pacote SCORM adaptado</li><li>• tempo: quantidade de segundos para concluir a seção.</li><li>• chat: quantidade de mensagens enviadas no chat.</li><li>• forum: quantidade de postagens no fórum.</li></ul> |
| sim_notas          | Resultados de avaliação em um tópico. Campos: <ul style="list-style-type: none"><li>• idGrade: código da grade da avaliação</li><li>• idItem: código da avaliação</li><li>• idUsuario: código do estudante</li><li>• dataHora: data/hora em que foi realizada a avaliação.</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |

Tabela 4.5: Tabelas acrescentadas à base de dados do Moodle para ampliar o Modelo do Estudante.

|                      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sim_estilo_estudante | <p>Associação dos usuários aos respectivos estilos de aprendizagem. Essa tabela foi modelada de forma que pudesse comportar diversos modelos e diversas dimensões para cada modelo. Campos:</p> <ul style="list-style-type: none"> <li>• idUsuario: código do estudante</li> <li>• modelo: código do modelo de estilo de aprendizagem utilizado. No caso, foi utilizado o código 1 para representar o modelo proposto por [Felder e Silverman 1988].</li> <li>• estilo: código do estilo de aprendizagem. No caso, foram definidos os seguintes códigos: 1 - ativo; 2 - reflexivo; 3 - Sensorial; 4 - Intuitivo; 5 - Visual; 6 - Verbal; 7 - Sequencial; 8 - Global.</li> <li>• valorEstilo: valor aferido no resultado do índice de estilos de aprendizagem.</li> <li>• dataHora: data/hora em que foi inserido o valor referente à escala.</li> </ul> |
| sim_lom_estudante    | <p>Associação dos objetos de aprendizagem apresentados ao estudante. Campos:</p> <ul style="list-style-type: none"> <li>• idUsuario: código do estudante</li> <li>• idLO: código do objeto de aprendizagem, ou seja, sua identificação dentro do pacote SCORM.</li> <li>• dataHora: data/hora em que foi inserido para ser apresentado ao usuário.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |

### 4.3 Sistema Adaptativo Baseado em Ontologias

Nesta seção, descreve-se o sistema adaptativo, que se baseia em tecnologias da web semântica e em padrões para *e-learning* no intuito de possibilitar, além da adaptatividade, a interoperabilidade. Assim, na seção 4.3.1 apresenta uma visão geral do processo de adaptação, os atores e suas responsabilidades. Na seção 4.3.2 serão descritas as ontologias que colaboram na arquitetura do sistema adaptativo proposto, descrito na seção 4.3.3. Por fim, para completar a descrição do sistema adaptativo baseado em ontologias, são apresentadas, na seção 4.3.4 as regras de adaptação definidas.

#### 4.3.1 Visão geral do processo de adaptação

O diagrama de atividades presente na figura 4.4 apresenta uma visão geral do processo de adaptação, apresentando graficamente as responsabilidades dos atores envolvidos.

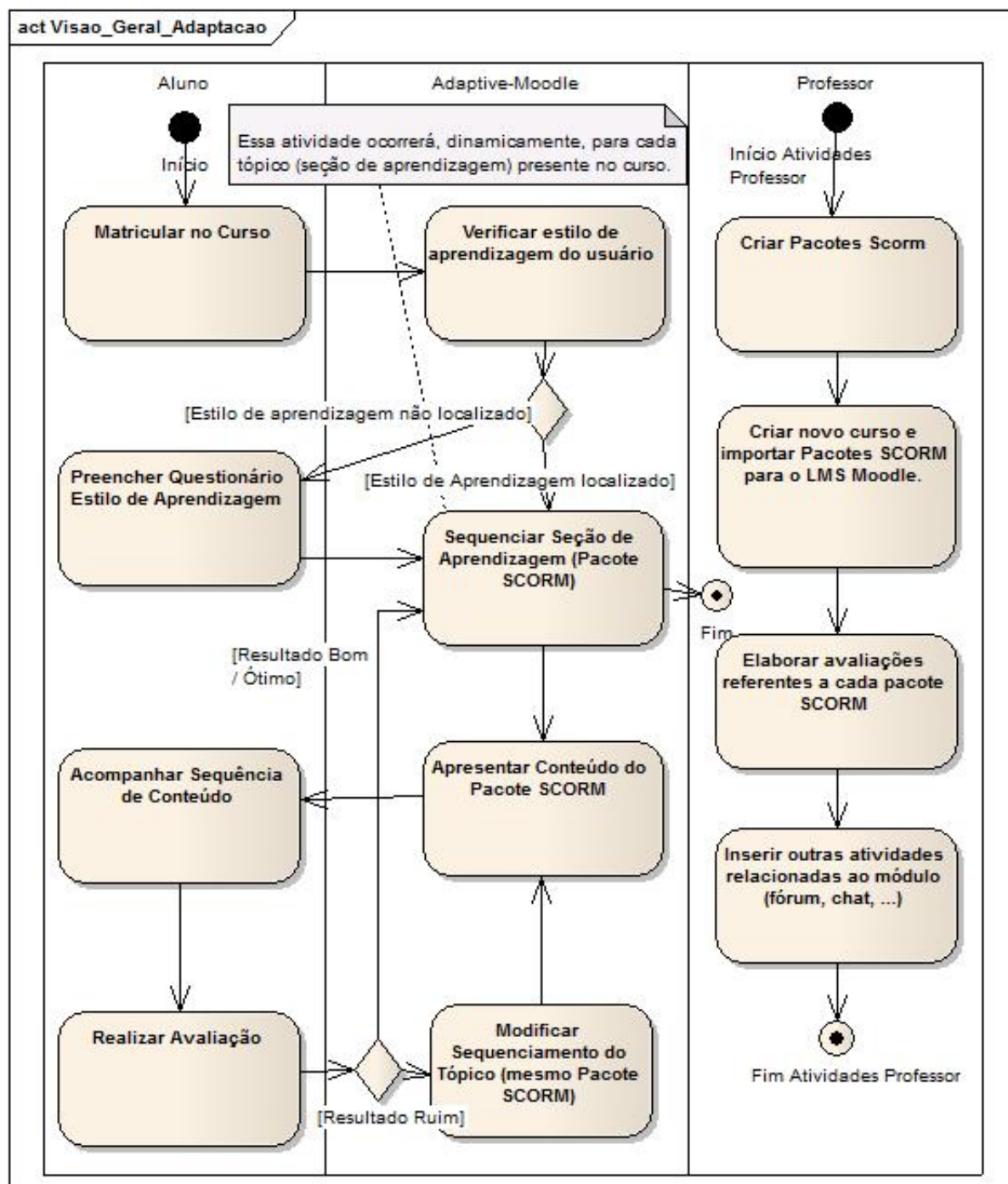


Figura 4.4: Visão geral do processo de adaptação

Conforme apresentado na figura 4.4, o professor ou administrador do sistema deverá estruturar o conteúdo do curso em pacotes SCORM, tendo em vista que, nesse padrão, poderão ser definidas várias organizações permitindo disponibilizar estruturas diferentes para um mesmo curso. Um outro benefício alcançado ao utilizar o padrão SCORM é que os cursos poderão ser migrados para qualquer outro LMS que o suporte, além de ser possível anotar recursos, objetos de aprendizagem que compõem os pacotes, com metadados de acordo com o padrão LOM, frequentemente utilizado por sistemas adaptativos e inteligentes como fonte para regras de adaptação.

Cada tópico no Moodle, neste trabalho também denominado *Seção de Aprendizagem*, conterá um pacote SCORM além de outras atividades disponibilizadas pelo ambiente. Inicialmente serão cobertas por essa proposta as seguintes atividades: questionários, para avaliar o conhecimento do estudante em relação à seção de aprendizagem, fórum e chat para avaliar o comportamento do estudante no decorrer do curso. Essa proposta baseia-se na plataforma de agentes do SIMEDUC, que foi integrado ao Moodle. O processo de adaptação no SIMEDUC não utiliza estilos de aprendizagem, apenas os resultados de avaliação do conhecimento, baseado nas respostas dos questionários de cada seção de aprendizagem, e do comportamento do estudante, baseado no acesso às ferramentas fórum e chat, e no tempo gasto na realização das atividades. Posteriormente, foi integrado outro sistema adaptativo, baseado nas tecnologias da web semântica, que utiliza também os estilos de aprendizagem.

Para a construção dos pacotes SCORM, pode-se utilizar duas abordagens, visando a aumentar as possibilidades de adaptação. Na primeira, os pacotes SCORM serão estruturados em diversas organizações, em grau crescente de complexidade, de forma a serem disponibilizadas pelo sistema aquelas adequadas ao nível de conhecimento dos estudantes. Na segunda abordagem, podem ser inseridos metadados, no padrão LOM, aos recursos dos pacotes SCORM, que correspondem aos objetos de aprendizagem. Em especial o metadado *descrição*, presente na categoria *Classificação* do padrão LOM, deverá conter o título do item correspondente no pacote SCORM, para que, dinamicamente, o sistema possa relacionar ao item os possíveis objetos de aprendizagem a serem apresentados ao estudante. Assim, objetos que se adequam a um determinado estilo de aprendizagem do estudante, poderão ser utilizados para adaptar o conteúdo. As duas abordagens são independentes, ou seja, uma pode ser utilizada sem a outra, mas também se complementam, ou seja, ambas podem ser utilizadas durante o processo de adaptação.

Conforme apresentado na figura 4.4, após se matricular no curso, será verificado para o estudante se já existe um estilo de aprendizagem associado a ele. Neste trabalho, o resultado aferido pelo Índice de Estilos de Aprendizagem [Felder e Soloman 2010], deve ser inserido manualmente para os alunos. É possível também, em uma próxima etapa, implementar um questionário com as questões do índice na própria atividade *Questionário* do Moodle. A verificação das respostas às questões poderão ser feitas pelo Agente de

Integração, que, assim, poderá apresentar ainda, no próprio ambiente, uma página de retorno ao usuário, informando-lhe o resultado. Naturalmente, é necessária uma licença para o uso do questionário, o que é possível, já que, conforme disposto em [Felder 2010], "o Índice de Estilos de Aprendizagem está disponível sem custos para alunos e docentes em instituições de ensino para fins não comerciais".

Na próxima seção serão descritas as ontologias modeladas, que foram a base para desenvolvimento do sistema adaptativo.

### 4.3.2 Ontologias

Baseando-se nos trabalhos já desenvolvidos que utilizam tecnologias da web semântica em ambientes de *e-learning*, assim como nos padrões para e-learning LOM e SCORM, foram modeladas três ontologias: a Ontologia LOM, para representar os objetos de aprendizagem no padrão LOM, a Ontologia do Domínio, que possui os conceitos principais associados a uma possível estruturação de cursos em ambientes de *e-learning* que suportem o padrão SCORM, a Ontologia do Estudante que implementa o Modelo do Estudante, ou seja, apresenta as classes e propriedades relacionadas ao Estudante.

#### Ontologia LOM

A Ontologia LOM baseia-se no padrão LOM, utilizado como referência pela maioria dos sistemas para descrever os objetos de aprendizagem, inclusive pelo padrão SCORM, que, neste trabalho, foi adotado como proposta para estruturar o conteúdo dos tópicos de um curso. O trabalho desenvolvido por [Nilsson et al. 2003] estabeleceu uma representação em RDF para o padrão LOM, que foi utilizado como base para a modelagem da ontologia LOM.

Na figura 4.5 é apresentada a hierarquia de classes presentes nesta ontologia.

Percebe-se, na figura 4.5, que na modelagem desta ontologia não foi considerado todo o padrão LOM, inserindo apenas os itens necessários para definir as regras de adaptação. Assim, foram criadas classes e propriedades que representam as seguintes categorias do padrão LOM: *geral*, para identificação do objeto de aprendizado; *classificação*, utilizado para associação dos objetos de aprendizagem aos itens das organizações de um pacote SCORM; e *educacional*, para definição de regras de adaptação. Nesta última categoria, considerou-se os seguintes elementos: *tipo de interatividade*, *densidade semântica*, *grau de dificuldade* e *tipo de recurso de aprendizagem*. Cada objeto de aprendizagem anotado nos pacotes SCORM de um curso é mapeado como uma instância dessa ontologia.

A seleção de alguns elementos do padrão LOM teve como objetivo definir um conjunto de elementos que foram organizados e aplicados para manter informações relevantes à descrição dos objetos de aprendizagem e que permitam inferir informações necessárias ao processo de adaptação, assim como para o compartilhamento de informações entre

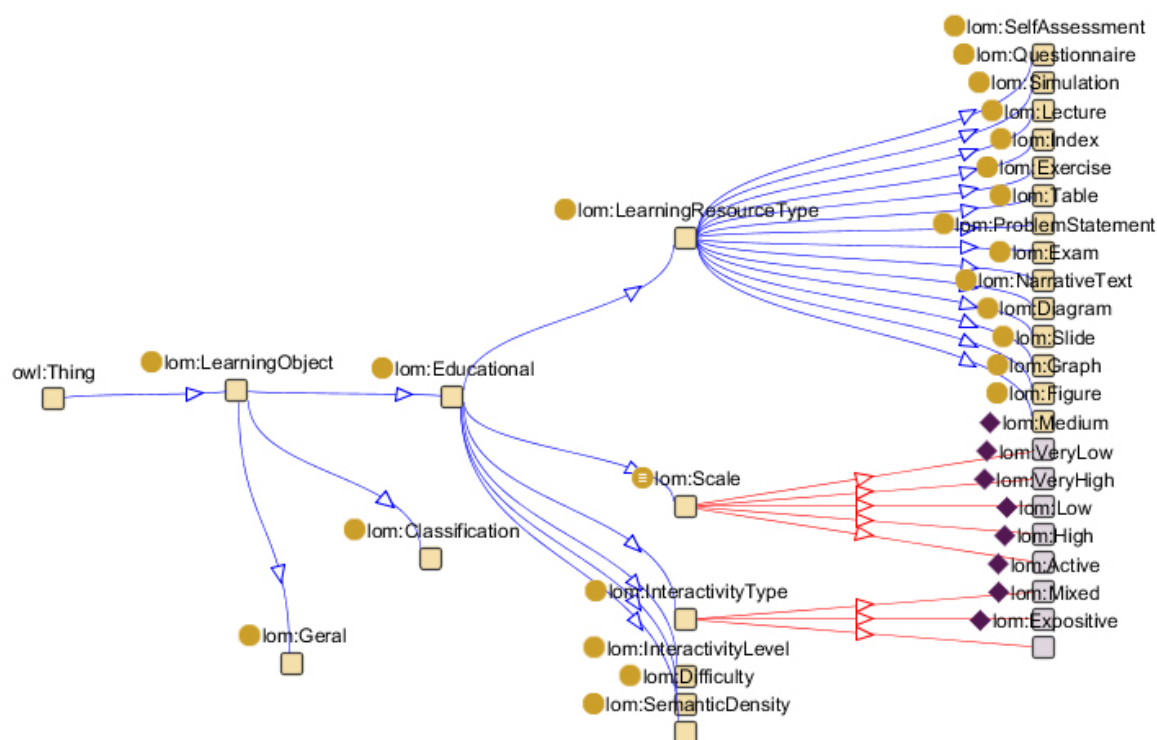


Figura 4.5: Hierarquia de classes da Ontologia LOM

ambientes. Essa ontologia deve ser estendida em um futuro trabalho, adequando-a ao vocabulário proposto por [Nilsson et al. 2003], de forma a considerar todo o padrão LOM, e incorporando outros vocabulários já existentes, como o Dublin Core [Initiav 2010], de forma a prover uma maior interoperabilidade e aumentar as possibilidades de que outros contextos e outras ontologias compartilhem os objetos de aprendizagem. No Apêndice A está incluído o código da ontologia LOM.

## Ontologia do Domínio

Na Ontologia do Domínio foram estabelecidos os conceitos envolvidos na estruturação de conteúdos em um ambiente de *e-learning*. No caso deste trabalho, tendo em vista que, ao adotar um padrão para a construção de cursos, é possível o compartilhamento entre diferentes sistemas, essa ontologia foi modelada com base no padrão SCORM.

Na figura 4.6 estão representadas as classes e as propriedades de objeto da Ontologia do Domínio. Um curso é constituído por seções, que correspondem a tópicos no ambiente de *e-learning* Moodle, que por sua vez podem conter diversos módulos, que podem se referir a atividades como fórum e *chat*. Neste trabalho, definiu-se que o conteúdo de um módulo deve ser estruturado em um pacote SCORM; além disso, em cada módulo, deve-se incluir avaliações para aferir o conhecimento do estudante. Em um pacote SCORM podem existir várias organizações, que correspondem à estruturação de um curso em determinado nível de complexidade. Cada organização possui diversos itens, que se referem aos tópicos do conteúdo. A cada item podem ser associados diversos objetos de aprendizagem, que

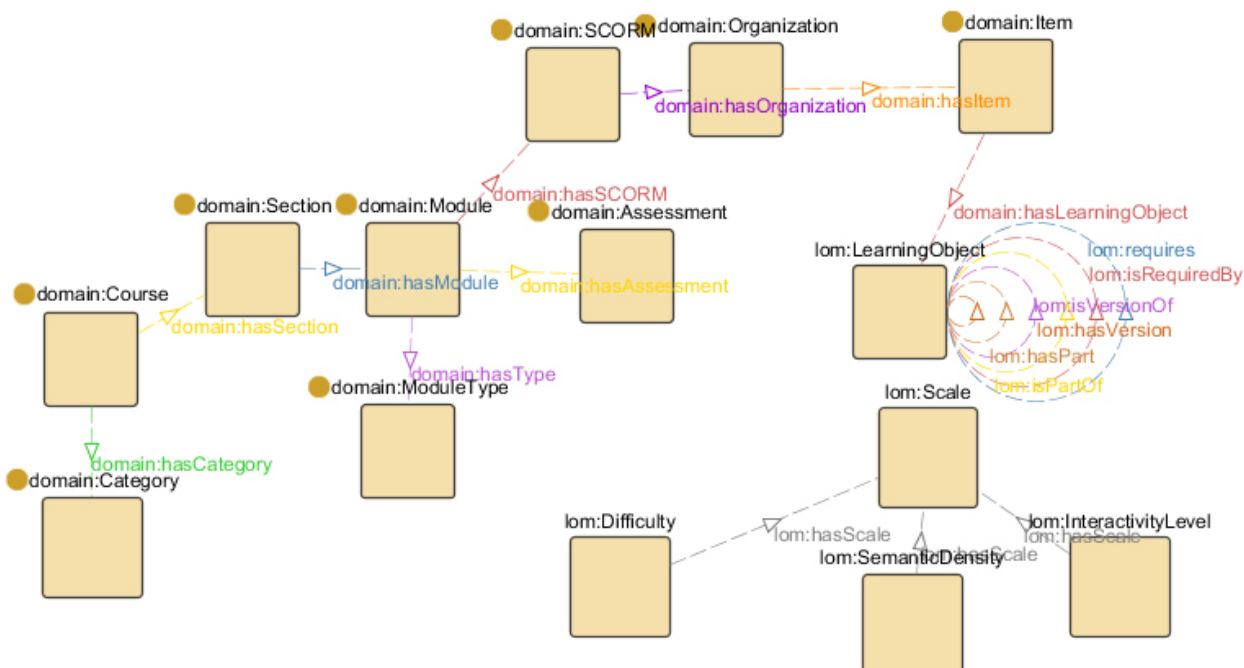


Figura 4.6: Classes e relacionamentos da Ontologia de Domínio

são instâncias da ontologia LOM. Isso permite, conforme exposto na seção 4.2.2, que a adaptação ocorra tanto no nível das organizações, ao se escolher uma de maior ou menor complexidade, assim como no nível dos objetos de aprendizagem, ao possibilitar a seleção daqueles mais adequados ao perfil de cada estudante.

Como a Ontologia de Domínio utiliza, ou seja, importa a Ontologia LOM, na figura 4.6 estão representados ainda os relacionamentos presentes nesta última, que são, basicamente, a representação da categoria *Relacionamento* do padrão LOM, que define alguns tipos de relacionamento entre os objetos, sendo modelados como propriedades inversas em OWL, como *isVersionOf* e *hasVersion*. Está representado ainda um padrão de projeto utilizado para o desenvolvimento de ontologias, o *Value Partition* (Partição de Valores), que, conforme [Horridge 2009], pode ser criado para refinar a descrição das classes, pois restringem o escopo de possíveis valores para uma lista exhaustiva. No caso a Partição de Valores *Scale* restringe os possíveis valores para as classes *Difficulty*, *SemanticDensity* e *InteractivityLevel*, que, conforme o padrão LOM, podem conter os valores *very low*, *low*, *medium*, *high* ou *very high*. No Apêndice B apresenta-se o código da ontologia do domínio.

## Ontologia do Estudante

Esta ontologia representa o Modelo do Estudante, possuindo informações específicas sobre os estilos de aprendizagem dos estudantes, que estabelecem medidas para as preferências individuais em quatro dimensões do processo de aprendizagem: percepção (sensorial/intuitivo), aquisição (visual/verbal), processamento (ativo/reflexivo) e compreensão (sequencial/global). Para um dos estilos em cada uma das dimensões é atribuído um valor

para cada estudante, indicando seu estilo predominante.

Esta ontologia estabelece, também, uma relação com as demais ontologias do ambiente, conforme representa a figura 4.7, que mostra as classes e propriedades de objeto definidas.

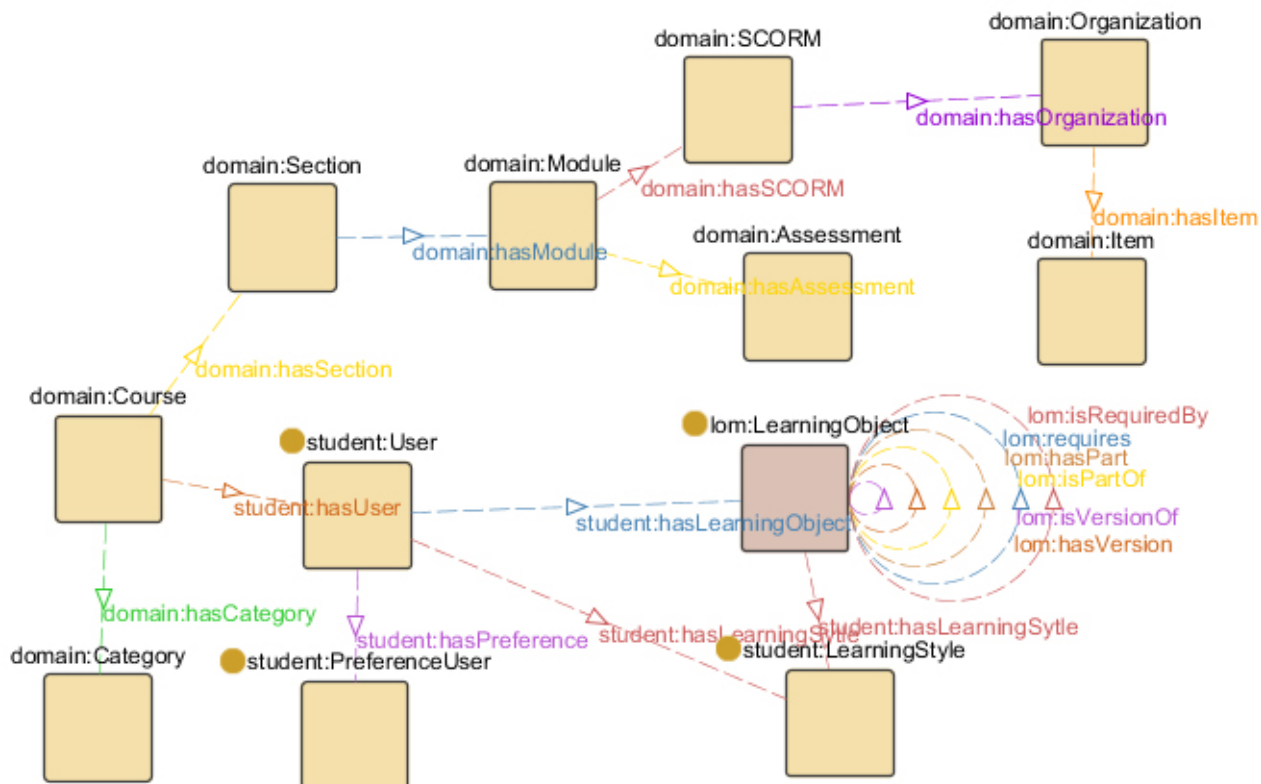


Figura 4.7: Classes e relacionamentos da Ontologia do Estudante

Conforme representa a figura 4.7, além do relacionamento entre o estudante e seus estilos de aprendizagem, a Ontologia do Estudante determina a relação entre o estudante e os objetos de aprendizagem. Também podem ser associadas *preferências* ao estudante, de forma que, posteriormente, o modelo seja enriquecido e novas regras de adaptação possam ser acrescentadas. No Apêndice C está incluído o código da Ontologia do Estudante.

### 4.3.3 Arquitetura do Sistema Adaptativo

Para modelar as ontologias e seus elementos foi utilizada a ferramenta Protégé, que dá suporte à criação de ontologias em OWL, permitindo a representação gráfica e a utilização de raciocinadores para verificar a consistência das ontologias e inferir classes não explicitamente apresentadas a partir da representação utilizada. A geração de instâncias para as ontologias foi feita utilizando a Protégé-OWL API, que juntamente com o *framework* Jena, permitem a criação de aplicações para a web semântica, manipulando as ontologias em OWL. Para popular a Ontologia LOM foram utilizados os objetos de aprendizagem descritos nos pacotes SCORM presentes no curso. Na base de dados do Moodle não há

referência aos metadados dos objetos. No entanto, todos os arquivos presentes no pacote SCORM são armazenadas na pasta de dados do sistema. Assim, processando o arquivo *imsmanifest.xml* é possível gerar as respectivas instâncias para cada objeto descrito.

Os dados necessários para popular a Ontologia do Domínio são encontrados diretamente na base de dados do Moodle, assim, basta mapear os valores para as respectivas propriedades presentes na ontologia. Já para a Ontologia do Estudante, parte dos dados também são encontrados na base de dados e os demais, referentes aos estilos de aprendizagem e a associação do estudante com os objetos de aprendizagem, também foram acrescentados à base de dados para armazenar as informações necessárias ao processamento do sistema adaptativo.

Após a modelagem das ontologias, foi definida a arquitetura do sistema adaptativo com base na proposta de [Knublauch 2004]. Assim, essa arquitetura apresenta as camadas e componentes descritas na figura 4.8.

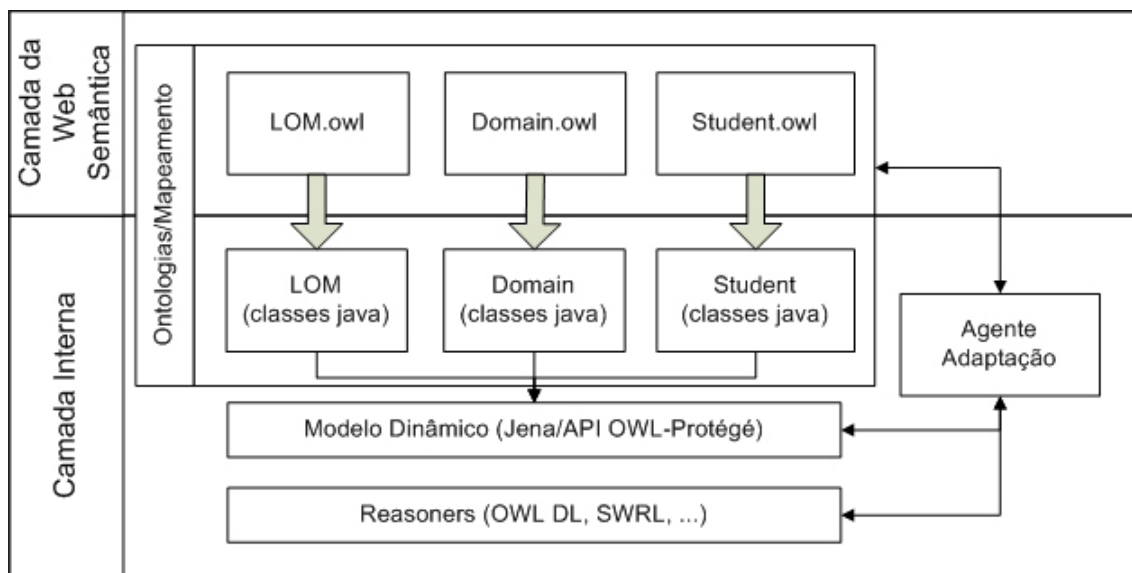


Figura 4.8: Arquitetura do Sistema Adaptativo

Na figura 4.8, estão representadas duas camadas distintas, mas interligadas. A camada da web semântica disponibiliza as ontologias para o acesso público, permitindo que sejam compartilhadas por outras aplicações, enquanto a camada interna apresenta as ontologias mapeadas em classes da linguagem Java, as ferramentas para manipulação dos modelos criados e os mecanismos de inferência e de regras. No caso deste trabalho, implementou-se um agente, denominado Agente de Adaptação, que utiliza as informações das ontologias para selecionar os objetos de aprendizagem mais adequados ao perfil de um estudante. No entanto, as ontologias podem ser exploradas por diferentes tipos aplicações.

As atividades executadas pelo Agente de Adaptação, conforme representado na figura 4.9, são basicamente as seguintes. Primeiro, recebe a mensagem *ProcessarRegras*, enviada pelo Agente de Integração, que contém o código do usuário, do pacote SCORM original, do novo pacote no curso adaptado e da organização específica selecionada neste pacote.

Depois disso, verifica se as instâncias das classes das ontologias foram geradas e estão atualizadas. Se não estiverem, cria ou atualiza essas instâncias. Se já estiverem atualizadas, processa as regras de adaptação, retornando uma relação de objetos de aprendizagem adequada àquela organização do pacote SCORM e armazenando as informações de quais objetos foram associados ao estudante. Como as ontologias são representadas em OWL, agentes externos podem utilizá-las; além disso, elas podem importar ontologias externas, de forma a complementar suas informações.

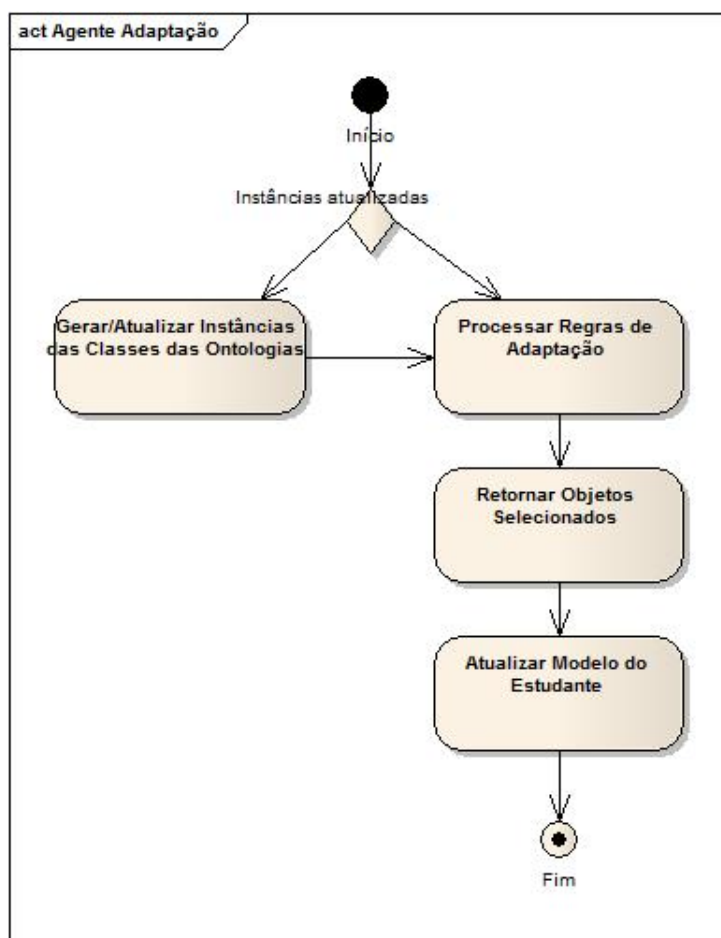


Figura 4.9: Atividades do Agente de Adaptação

O mapeamento das ontologias em classes Java deve-se ao fato de que, como as aplicações gerenciam funcionalidades, como acesso a banco de dados, controle de seção, interface do usuário, necessitam representar os objetos que são manipulados. Além disso, manipular diretamente o modelo OWL não é conveniente, conforme exposto em [Knublauch 2004], visto que as referências são feitas somente por meio de nomes (*strings*), tornando o código difícil de testar e manter. Assim, a melhor alternativa é mapeá-las, utilizando classes convencionais das linguagens orientadas a objeto, que permitem, além de eliminar a inconveniência citada, incluir métodos para essas classes. A geração das classes Java correspondentes a uma ontologia pode ser feita diretamente pelo Protégé, recurso que foi utilizado neste trabalho.

A aplicação pode ainda explorar mecanismos de inferência e motores de regras, para apresentar um comportamento inteligente. No caso desse sistema, o responsável por controlar o comportamento da aplicação é um agente e as regras foram criadas em SWRL. A arquitetura permite que qualquer aplicação possa ser criada, de forma que é possível, por exemplo, interagir com usuário final por meio de tecnologias como interfaces *web*, aplicações *desktop* ou *web services*.

#### 4.3.4 Regras de Adaptação

Para a avaliação do desempenho do estudante, baseou-se na proposta apresentada por [Faria et al. 2008], em que o resultado final é composto por dois fatores: o comportamento e o conhecimento do estudante. Para avaliação do comportamento, leva-se em consideração o tempo que o estudante levou para concluir todo o conteúdo, assim como o tempo gasto para realizar a avaliação, a quantidade de mensagens postadas e a sua participação do fórum. Para a avaliação do conhecimento leva-se em consideração o percentual de questões corretas. Após essa verificação, os valores apurados são submetidos a um agente de avaliação que utiliza regras *fuzzy*. Esse agente já estava implementado no sistema adaptativo SIMEDUC, em outro contexto, em que não seleciona uma organização de um pacote SCORM, mas retorna um número de 1 a 5 que associa o usuário a um determinado grupo. Assim, foi adaptado neste trabalho para permitir a parametrização das variáveis avaliadas com os respectivos pesos. Dessa forma, pode-se estabelecer pesos para o tempo, o comportamento, o conhecimento e o desempenho final do estudante. Ao final, verifica-se a nota do estudante e, com base no número de organizações disponíveis, associa-se a organização específica do pacote SCORM a que ele terá acesso.

Como cada pacote disponibiliza várias organizações, em grau crescente de complexidade, no primeiro contato, o sistema disponibiliza para o estudante uma organização de complexidade intermediária. A partir daí, a seção de aprendizagem, assim como a organização, serão disponibilizadas de acordo com o resultado da avaliação do estudante. Se o estudante obtiver um resultado inferior à nota estabelecida como parâmetro, terá que rever a mesma seção de aprendizagem, com uma organização de complexidade menor. Se obtiver um resultado equivalente ou superior à nota estabelecida, o sistema disponibiliza uma nova seção com uma organização de complexidade intermediária ou de uma complexidade maior. Após o processamento do Agente de Avaliação, presente no sistema SIMEDUC, cujo resultado é usado para definir a organização a que o aluno terá acesso, o Agente de Integração continua a execução, enviando a mensagem *ProcessarRegras* para o Agente de Adaptação. No conteúdo dessa mensagem, o agente identifica a organização específica para a qual, em cada item, associará objetos de aprendizagem adequados ao estilo de aprendizagem e ao desempenho do estudante no decorrer do curso.

Assim, para cada interação do estudante com uma nova organização do pacote SCORM,

e para cada item presente na organização selecionada, o sistema seguirá o esquema representado na figura 4.10.

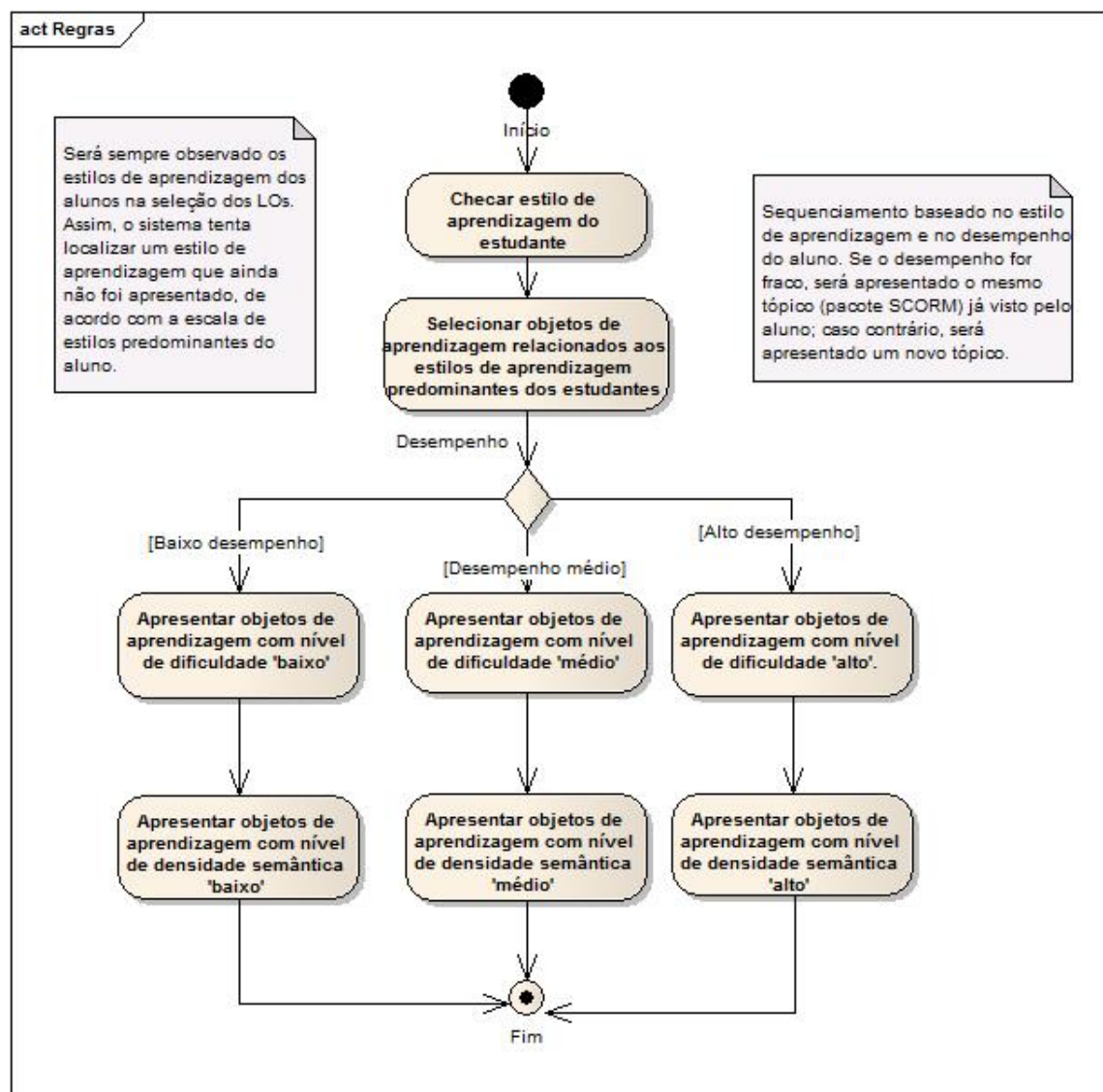


Figura 4.10: Regras para seleção de objetos de aprendizagem

Conforme representado na figura 4.10, o sistema procura as instâncias da ontologia LOM relacionadas ao estilo de aprendizagem do estudante. Essa busca retorna os valores em ordem descendente na escala, de forma a selecionar os estilos de aprendizagem predominantes para o estudante. É necessário armazenar no histórico do estudante os objetos de aprendizagem que já foram apresentados, para que o sistema procure sempre apresentar aqueles com os quais o estudante ainda não teve contato. Assim, o sistema tentará localizar aqueles que sejam mais adequados ao estilo de aprendizagem predominante para o estudante, de acordo com a organização que foi selecionada para ele em determinada seção de aprendizagem e que ainda não foram apresentados.

São avaliadas condições para definir qual o objeto de aprendizagem a ser apresentado. Inicialmente, essas condições estão relacionadas aos seguintes elementos do padrão LOM,

presentes na categoria *Educacional*:

- Tipo de Interatividade: ao qual pode ser associado um dos seguintes valores: *aprendizado ativo*, *aprendizado expositivo* ou *aprendizado combinado*.
- Tipo de Recurso de Aprendizado: que podem ser, de acordo com o padrão LOM, um dos seguintes conteúdos: exercício, simulação, questionário, diagrama, figura, gráfico, índice, slide, tabela, texto narrativo, exame, experiência, problema, auto-avaliação, leitura.
- Densidade Semântica: grau de concisão (tamanho, abrangência, duração).
- Dificuldade: grau de complexidade em relação à faixa etária.

Outros itens poderão ser incluídos posteriormente, visando ao enriquecimento do processo de avaliação, como o tempo para aprendizado do objetivo, o idioma, o contexto - nível para o qual o objeto foi elaborado, a faixa etária pretendida, entre outros. Outro tópico que poderá ser considerado para seleção de objetos de aprendizagem são as relações entre os objetos padrão LOM, como *é parte de*, *é versão de*, *é base de*, *é pré-requisito de*.

Considera-se ainda, durante o processo de seleção dos objetos de aprendizagem, os estilos de aprendizagem do estudante de acordo com a classificação proposta por [Felder e Silverman 1988], em que é definida uma escala para cada estilo de aprendizagem associado ao estudante, de 1 a 11, sendo que:

- O valor 1 equivale a um estudante equilibrado na dimensão, com leve propensão para uma das características.
- O valor de 3 a 5 significa um estudante moderado entre as características das duas dimensões.
- O valor de 7 a 9 significa um estudante muito voltado a uma determinada característica na dimensão.
- O valor 11 significa um estudante extremamente voltado a uma determinada característica na dimensão.

Seleciona-se, portanto, as características dominantes nos estilos de aprendizagem dos estudantes. Após isso, são estabelecidas as seguintes relações entre as características propostas com os respectivos elementos do padrão LOM:

- Dimensão *Processamento da Informação*: relaciona-se com o elemento *Tipo de Interatividade*. Assim, se a característica predominante do estudante é:
  - Ativo; então o sistema procura, primeiramente, por objetos de aprendizagem cujo tipo de interatividade seja *Aprendizado Ativo*. Como segundo critério, aplicado quando não são localizados objetos classificados como *Aprendizado*

*Ativo*, relaciona-se com o elemento *Tipo de Recurso de Aprendizagem*, selecionando objetos de aprendizagem do tipo *exercício*, *simulação*, *questionário*, *problema*.

- Reflexivo; então deve-se selecionar objetos cujo *tipo de interatividade* seja *Aprendizado Expositivo*. Também como segundo critério, leva-se em consideração os seguintes tipos: *experiência*, *auto-avaliação*, *leitura*.
  - No caso de um equilíbrio (valor 1 na escala) entre as duas características; então a preferência é por *Aprendizado Combinado*. Normalmente, outras características são consideradas antes de entrar nessa situação, já que seleciona-se primeiro o estilo de aprendizagem cujo valor na escala é maior.
- Dimensão *Percepção da Informação*: relaciona-se com o elemento *Tipo de Recurso de Aprendizagem*.
    - Sensorial: então o sistema seleciona objetos de aprendizagem do tipo *problema*, *exercício*, *tabela*, *gráfico*.
    - Intuitivo: então são selecionados objetos do tipo *slide*, *texto narrativo*, *diagrama*, *figura*, *leitura*
  - Dimensão *Recepção da Informação*: relaciona-se com o elemento *Tipo de Recurso de Aprendizagem*
    - Visual: seleciona-se objetos do tipo *simulação*, *diagrama*, *figura*, *gráfico*, *índice*.
    - Verbal: seleciona-se objetos do tipo *texto narrativo*, *experiência*, *leitura*.

Como se vê, as possibilidades de associar os estilos de aprendizagem com os metadados dos objetos de aprendizagem, com as opções existentes no padrão LOM, são restritas. Para a dimensão *Compreensão da Informação*, por exemplo, em que se inserem os estilos *sequencial* e *global*, não foi encontrada uma forma coerente de associar objetos de aprendizagem de acordo com os elementos considerados. Uma outra possibilidade, seria associar um metadado específico, no item Classificação, para definir a quais estilos cada objeto se relaciona, e qual a escala de cada um. No entanto, essa proposta também é limitada em termos de interoperabilidade, já que um outro sistema não considerará essa anotação, além de exigir um trabalho adicional do professor durante a organização dos conteúdos de um curso. Testes em ambiente de produção devem ser feitas para verificar a viabilidade de se implementar essa característica. Optou-se, portanto, nesta versão do sistema, pela associação descrita acima. Um trabalho adicional necessário é estudar mais profundamente os estilos de aprendizagem, assim como definir outras teorias educacionais, para refinar as regras descritas e enriquecer as possibilidades de adaptação.

## 4.4 Testes e Resultados

Para a construção de um curso modelo, empregado na realização dos testes, foi utilizada a ferramenta Reload Editor. Assim, construiu-se um curso introdutório sobre a tecnologia Flash, distribuído em três tópicos de conteúdo, que correspondem a três pacotes SCORM. O primeiro tópico está representado na figura 4.11.

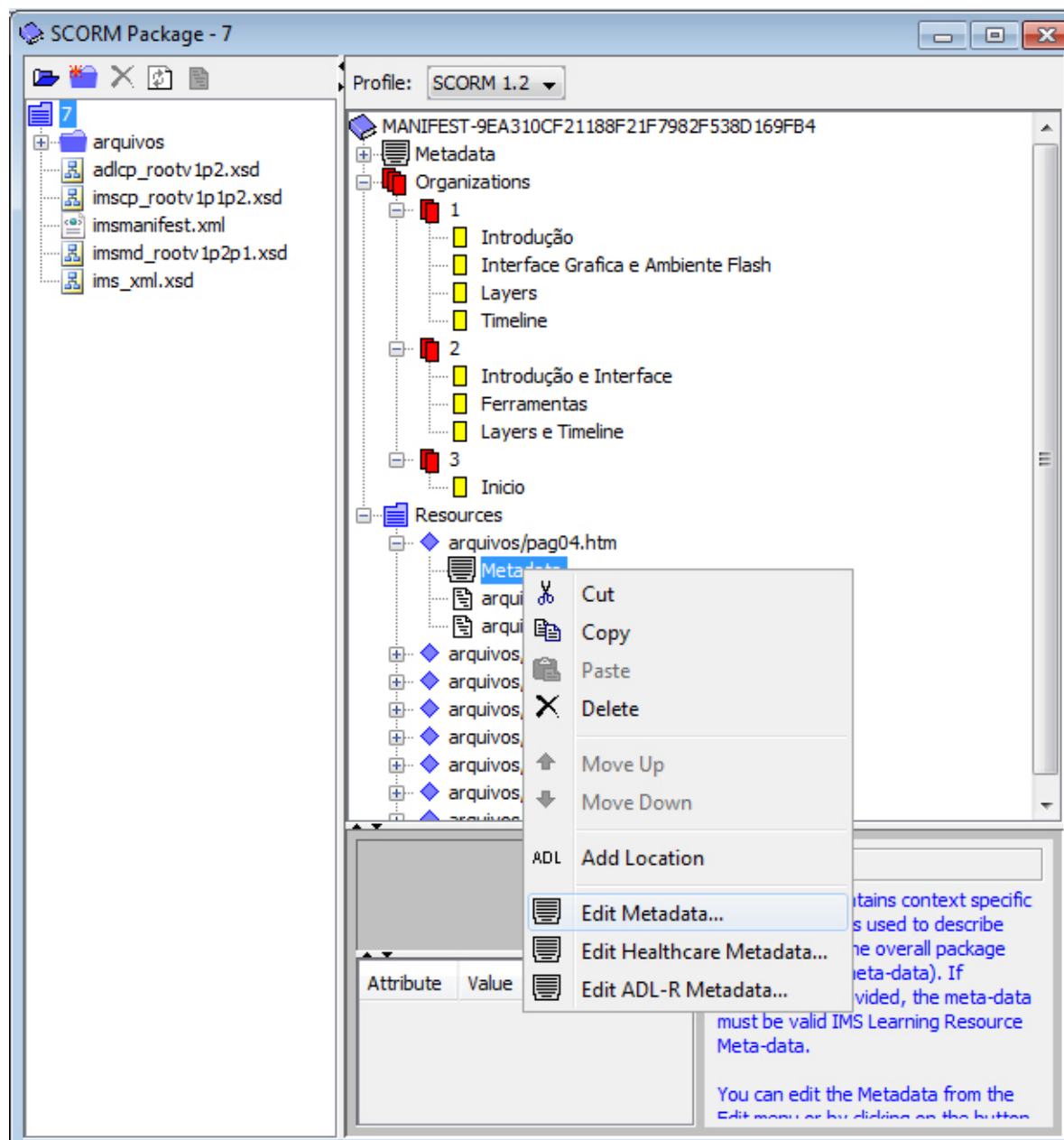


Figura 4.11: Estrutura do pacote SCORM com várias organizações

Como mostra a figura 4.11, para cada pacote foram definidas três organizações representando a organização do conteúdo em diferentes níveis de complexidade. Na figura 4.11 percebe-se ainda que é possível, utilizando o Reload Editor, associar metadados aos recursos, que correspondem aos objetos de aprendizagem. Os recursos devem ser anotados levando-se em consideração, principalmente, os elementos presentes nas regras de

adaptação, conforme descrito na seção 4.3.4. Além disso, para permitir a associação de diferentes recursos a um mesmo item, neste trabalho, o subelemento *descrição*, presente na categoria *Classificação* do padrão LOM, deve ser anotado com o mesmo nome do item presente na organização.



Figura 4.12: Curso original e conteúdo das organizações

Inicialmente foram feitos os testes relacionados à integração do ambiente de e-learning Moodle ao sistema adaptativo SIMEDUC. Como mostra a figura 4.12, um curso normal no ambiente Moodle apresenta todos os tópicos e todas as organizações associadas a um pacote SCORM, sendo que o usuário pode selecionar qualquer uma das organizações presentes por meio de uma caixa de seleção. Já um curso adaptado, conforme a proposta deste trabalho, mostra apenas a organização definida pelas regras de adaptação. Como, no caso exemplificado na figura 4.13, que corresponde ao primeiro acesso do estudante depois que foi matriculado no curso: disponibiliza-se o primeiro tópico do curso e a organização de complexidade intermediária. No Apêndice D são apresentados alguns testes e resultados, baseando-se na seleção de uma organização do pacote SCORM.

Nesse curso, inicialmente, apenas um objeto de aprendizagem foi associado a cada item de conteúdo de uma organização, de forma que as regras de adaptação ainda não estão sendo processadas. Posteriormente, outros objetos de aprendizagem, com características diferentes, foram elaborados e associados aos itens para testar essas regras. Os testes e resultados, que se baseiam nos estilos de aprendizagem para seleção de objetos, estão



Figura 4.13: Curso adaptado e conteúdo da organização selecionada

descritos no Apêndice E.

Como o processo de adaptação ocorre após o término de um tópico, ao processar os resultados de avaliação, não há problemas de desempenho. O estudante deve estar ciente de que um novo tópico será liberado apenas após o processamento dos resultados e das regras de adaptação, quando é definida a organização, no pacote SCORM, a que ele terá acesso e quais os objetos serão associados a cada item dessa organização, de acordo com o seu desempenho e o seu estilo de aprendizagem.

Todos os dados presentes nos testes, como por exemplo as características associadas aos objetos de aprendizagem e aos estudantes foram simulados em um ambiente de teste. Outros testes devem ser efetuados visando a garantir a corretude do sistema em relação ao processo de adaptação. Deve ser criado ainda um curso a ser aplicado em um ambiente real para validar a proposta.

## 4.5 Considerações

Este capítulo apresentou o trabalho desenvolvido visando à consecução dos objetivos propostos, ou seja, a integração de ambientes de *e-learning* a sistemas adaptativos e o desenvolvimento de um sistema adaptativo baseado em tecnologias da web semântica, de forma a implementar adaptatividade e possibilitar interoperabilidade em *e-learning*. Assim, foram descritos os requisitos e a arquitetura do Adaptive-Moodle, sistema que engloba todos os componentes envolvidos, ou seja, o ambiente Moodle, o agente de integração, o sistema adaptativo SIMEDUC e o sistema adaptativo baseado em ontologias. Descreveu-se, ainda, o agente que implementa a integração proposta, a arquitetura do sistema adaptativo, as ontologias e regras que o compõem.

Alguns problemas foram encontrados no desenvolvimento do sistema adaptativo baseado em ontologias, como a dificuldade em definir as regras de adaptação a partir dos elementos presentes no padrão LOM, assim como a integração entre todas as tecnologias envolvidas no processo de desenvolvimento do sistema. Para suprir essas dificuldades, procurou-se, principalmente, definir um ambiente que pudesse ser extensível, de forma que a integração com outros sistemas adaptativos e, assim, complementar e possibilitar outras formas de adaptação.

## Capítulo 5

# Conclusões e Trabalhos Futuros

No intuito de contribuir para o desenvolvimento dos ambientes de *e-learning*, de modo que possuam adaptatividade, ou seja, a capacidade de adequar o conteúdo ao perfil dos estudantes, e sejam interoperáveis, isto é, possam compartilhar recursos de aprendizagem, esse trabalho apresentou como resultado a integração entre o ambiente Moodle e o sistema adaptativo SIMEDUC, por meio de um agente de software, assim como o desenvolvimento de um sistema adaptativo, que se baseia em tecnologias da web semântica e que também foi integrado ao ambiente Moodle. Apesar de utilizar um ambiente específico, acredita-se que essa integração seja possível com outros ambientes de *e-learning*, principalmente aqueles que, conforme a proposta de estruturação de conteúdos apresentada, suportem padrões para *e-learning*, no caso específico o SCORM, para estruturação do conteúdo de um curso, e o LOM, para descrição dos objetos de aprendizagem.

Para alcançar esses resultados, conforme explicitado no decorrer deste trabalho, inicialmente foi feito um estudo em relação aos sistemas adaptativos e aos ambientes de *e-learning*, considerando também a utilização de padrões para *e-learning* e a teoria de estilos de aprendizagem. Esse estudo permitiu fundamentar a proposta e obter os dados necessários para a definição dos requisitos e da arquitetura dos sistemas propostos. O conhecimento das características desses dois tipos de sistemas foi fundamental para a definição de estrutura necessária para permitir a integração entre eles, a fim de que os usuários possam se beneficiar tanto do uso abrangente dos ambientes de *e-learning*, quanto da adaptatividade presente nos sistemas adaptativos.

Foi feito ainda um estudo sobre as tecnologias envolvidas no desenvolvimento da web semântica, para compreender sua estrutura e definir como essas tecnologias poderiam contribuir para o desenvolvimento do sistema adaptativo proposto, de forma a, principalmente, possibilitar uma maior interoperabilidade. Dessa forma foi possível definir as ontologias que colaboram para o sistema adaptativo, bem como outras tecnologias utilizadas no desenvolvimento desse sistema.

Tanto para a integração, quanto para o gerenciamento das informações do sistema adaptativo, optou-se por implementar as aplicações utilizando a tecnologia de sistemas

multiagentes, pela sua autonomia, ao possibilitar a resposta a determinados eventos em função de perceber mudanças no ambiente em que atuam, e não unicamente em virtude de solicitações de usuários, assim como por permitirem a extensibilidade do sistema, ou seja, novos agentes, com novas tarefas podem facilmente ser acoplados.

Preocupou-se, na modelagem das ontologias, que foram estruturadas de forma genérica, para a definição de cursos em quaisquer áreas do conhecimento, também com a questão da extensibilidade. Assim, por exemplo, novas teorias educacionais podem ser inseridas para permitir outras possibilidades a serem exploradas pelas regras de adaptação. As ontologias são representadas em OWL, a linguagem padrão para criação de ontologias definida pela W3C, facilitando a sua manipulação, já que existem ferramentas disponíveis, como a Protégé-OWL API, utilizada nesse trabalho, que permitem a criação de quaisquer aplicações, inclusive agentes externos, para processar essas ontologias. Além disso, as ontologias podem importar vocabulários ou ontologias externas, de forma a complementar suas informações.

É necessário ainda, em relação às ontologias, refatorá-las de forma que possam incorporar vocabulários/ontologias já disponíveis na web, possibilitando uma maior interoperabilidade. Instituições poderão instanciar as classes da ontologia e publicar os indivíduos como arquivos OWL em seus *sites*. Assim, um agente poderá coletar informações sobre esses indivíduos e utilizá-los em aplicações para *e-learning*, de forma que, se em determinado repositório não existir um objeto de aprendizagem adequado ao estilo de aprendizagem do estudante, esse agente poderá investigar outros repositórios e retornar o resultado desejado. Podem ser definidas também, em novas aplicações, especializações das classes utilizadas, criando restrições semânticas específicas para desenvolver um ambiente customizado.

A proposta de adaptação apresentada baseia-se em duas possibilidades complementares e independentes. A primeira é a definição de várias organizações em um pacote SCORM com diferentes níveis de complexidade e a segunda é a associação de diferentes objetos de aprendizagem a cada item de forma que os mais adequados ao estilo de aprendizagem e ao nível de conhecimento do estudante sejam selecionados. Assim, estruturar o conteúdo dos cursos de acordo com essa proposta apresenta alguns pontos positivos, como propiciar uma maior interoperabilidade, pois o curso estará disponível para qualquer ambiente que suporte o padrão, além de orientar a organização de cursos focados nas necessidades dos alunos, no seu contexto e nos seus objetivos de aprendizagem. Porém, dificulta o trabalho no sentido de será uma tarefa árdua definir um curso nesse formato, em que muitas organizações e muitos objetos de aprendizagem podem ser necessários para que o processo de adaptação ocorra de forma adequada.

Assim, serão necessárias propostas para facilitar a criação de cursos, como ferramentas para retornar informações para o professor sobre a organização do curso e acompanhamento dos estudantes, informando, por exemplo, que não encontrou objetos de aprendi-

zagem adequados ao estilo de aprendizagem de determinado estudante. A geração das organizações em um pacote SCORM em diversos níveis de complexidade também pode ser automatizada, utilizando-se, por exemplo, um mapa conceitual que relacione todos os itens de um curso e estabeleça relações entre eles, como *parte de*, *pré requisito de*. Novas ferramentas para interação com o aluno também são possíveis, como, por exemplo, para sugerir atividades baseadas em seu estilo que podem contribuir para uma melhor apreensão do conteúdo.

Como os estilos de aprendizagem de cada aluno estão sendo inseridos manualmente, é necessário ainda um trabalho para definição dos estilos de aprendizagem como um questionário dentro do próprio ambiente do Moodle que seja processado pelo Agente de Integração. Assim, ao verificar uma nova matrícula, o sistema deve verificar se já existem estilos de aprendizagem associados ao estudante. Caso não exista, a primeira atividade do curso será esse questionário.

Como a definição dos estilos de aprendizagem é de natureza subjetiva, baseada apenas em respostas a um questionário, pode-se desenvolver também, em um trabalho futuro, uma rede neural para avaliar e atualizar os estilos dos estudantes, visando a uma definição mais precisa desses estilos e, conseqüentemente, aprimorar o processo de adaptação. Assim, como as características de um estudante podem se modificar no decorrer do tempo, essa rede neural será responsável por ajustar os valores referente a cada estilo de aprendizagem observando-se os dados presentes no modelo do estudante. Uma rede neural pode ser desenvolvida também para determinar a organização mais adequada ao perfil de um estudante, trabalhando em conjunto com as regras *fuzzy* do sistema adaptativo SIMEDUC.

Em suma, nesse trabalho apresentou-se uma proposta cuja pretensão é contribuir para o avanço dos ambientes de *e-learning* e dos sistemas adaptativos, em relação à adaptividade e à interoperabilidade, e para que outras propostas sejam desenvolvidas, de modo a consolidar essas tecnologias como uma ferramenta eficaz, tanto para estimular a aprendizagem dos alunos quanto para auxiliar o trabalho do professor. Espera-se que, a partir das propostas apresentadas, surjam novas idéias para integração com outros ambientes e utilização de outros sistemas adaptativos, inclusive possibilitando novas formas de adequar o conteúdo às necessidades dos estudantes, assim como novas possibilidades de compartilhar objetos de aprendizagem com outros sistemas, facilitando a construção de cursos.

## 5.1 Publicações

Também como contribuições desse trabalho, foram aceitos para publicação os seguintes artigos:

1. *Adaptive-Moodle: Adaptatividade no Ambiente Moodle*, para o evento MoodleMoot

2010, uma conferência dedicada aos usuários, desenvolvedores e administradores do ambiente Moodle, a ser realizado no período de 29 de setembro a 1º de outubro de 2010, juntamente com o Congresso Latino Americano de Objetos de Aprendizagem “LACLO 2010”.

2. *Adaptivity and Interoperability in e-Learning using Ontologies*, para o evento Iberamia 2010, a 12ª edição da Conferência Íbero-Americana de Inteligência Artificial, um simpósio que reúne as comunidades íbero-americanas para compartilhar resultados de pesquisas e experiências com pesquisadores em Inteligência Artificial, a ser realizado no período de 1 a 5 de novembro de 2010.

# Referências Bibliográficas

- [ActiveMath 2010] ActiveMath (2010). ActiveMath. <http://www.activemath.org>.
- [AHA 2010] AHA (2010). AHA: Adaptive Hypermedia for All. <http://aha.win.tue.nl/>.
- [Boag et al. 2007] Boag, S., Chamberlin, D., Fernández, M. F., Florescu, D., Robie, J., e Jérôme Siméon, E. (2007). XQuery 1.0: An XML Query Language. *W3C Working Draft*. <http://www.w3.org/TR/xquery/>.
- [Boley e Michael Kifer 2010] Boley, H. e Michael Kifer, E. (2010). RIF Framework for Logic Dialects. *W3C Recommendation*. <http://www.w3.org/TR/rif-ld/>.
- [Bolzan 2002] Bolzan, W. (2002). Estudo sobre metodologias para o desenvolvimento de Sistemas Multiagentes. Trabalho individual, PUC-RS.
- [Bra et al. 2006] Bra, P. D., Smits, D., e Stash, N. (2006). Creating and Delivering Adaptive Courses with AHA! In *Proceedings of the first European Conference on Technology Enhanced Learning*, pp. 21–33. Springer LNCS. <http://wwwis.win.tue.nl/debra/ec-tel2006.pdf>.
- [Bray et al. 2006] Bray, T., Paoli, J., Sperberg-McQueen, C. M., Maler, E., Yergeau, F., e John Cowan, E. (2006). Extensible Markup Language (XML) 1.1 (Second Edition). *W3C Recommendation*. <http://www.w3.org/TR/xml11>.
- [Brickley e R. V. Guha 2004] Brickley, D. e R. V. Guha, E. (2004). RDF Vocabulary Description Language 1.0: RDF Schema. *W3C Recommendation*. <http://www.w3.org/TR/rdf-schema/>.
- [Brooks e McCalla 2006] Brooks, C. e McCalla, G. (2006). Towards Flexible Learning Object Metadata. In *International Journal of Continuing Engineering and Lifelong Learning*, pp. 50–63. [http://www.cs.usask.ca/cab938/ijceell2006\\_brooks\\_mccalla.pdf](http://www.cs.usask.ca/cab938/ijceell2006_brooks_mccalla.pdf).
- [Brusilovsky e Peylo 2003] Brusilovsky, P. e Peylo, C. (2003). Adaptive and intelligent web-based educational systems. *international journal of artificial intelligence in education*, 13:156–169.
- [Claroline 2010] Claroline (2010). Claroline. <http://www.claroline.net/>.
- [Coffield et al. 2004] Coffield, F., Moseley, D., Hall, E., e Ecclestone, K. (2004). Learning styles and pedagogy in post-16 learning: a systematic and critical review. *Learning and Skills Research*.

- [Committee 2005] Committee, I. L. T. S. (2005). Draft Standard for Learning Technology - Extensible Markup Language (XML) Schema Definition Language Binding for Learning Object Metadata. [http://ltsc.ieee.org/wg12/files/IEEE\\_1484\\_12\\_03\\_d8\\_submitted.pdf](http://ltsc.ieee.org/wg12/files/IEEE_1484_12_03_d8_submitted.pdf).
- [Committee 2010] Committee, I. L. T. S. (2010). WG12: Learning Object Metadata. <http://ltsc.ieee.org/wg12/index.html/>.
- [Consortium 2010a] Consortium, I. G. L. (2010a). Specifications. <http://www.imsglobal.org/specifications.html/>.
- [Consortium 2010b] Consortium, W. (2010b). Semantic Web. <http://www.w3.org/standards/semanticweb/>.
- [Consortium 2010c] Consortium, W. (2010c). Vocabularies. <http://www.w3.org/standards/semanticweb/ontology>.
- [Daconta et al. 2003] Daconta, M. C., Obrst, L. J., e Smith, K. T. (2003). *The Semantic Web. A Guide to the Future of XML, Web Services, and Knowledge Management*. Wiley Publishing Inc, Oxford, New York, USA.
- [Dave Beckett 2004] Dave Beckett, E. (2004). RDF/XML Syntax Specification (Revised). *W3C Recommendation*. <http://www.w3.org/TR/rdf-syntax-grammar/>.
- [Dean e G. Schreiber 2004] Dean, M. e G. Schreiber, E. (2004). OWL Web Ontology Language Reference. *W3C Recommendation*. <http://www.w3.org/TR/2004/REC-owl-ref-20040210/>.
- [Dolog e Nejdl 2007] Dolog, P. e Nejdl, W. (2007). Semantic web technologies for the adaptive web. *Lecture Notes in Computer Science*, 4321:697–719.
- [Dorça 2004] Dorça, F. A. (2004). Um sistema inteligente multiagente para educação à distância. Master's thesis, Universidade Federal de Uberlândia.
- [Duerst e Suignard 2005] Duerst, M. e Suignard, M. (2005). Internationalized Resource Identifiers (IRIs). *The Internet Society*. <http://tools.ietf.org/html/rfc3987>.
- [Durao et al. 2008] Durao, F., Dolog, P., e Jahn, K. (2008). State of the Art: Personalization. Technical report, KIWI project FP7. Project Number: ICT-2007.4.2-211932 Document number: ICT211932/SFRG/D2.7/D/PU/b1. Disponível em [http://wiki.kiwi-project.eu/index.jsp?title=kiwi-pub:KiWi\\_D2.7\\_final.pdf](http://wiki.kiwi-project.eu/index.jsp?title=kiwi-pub:KiWi_D2.7_final.pdf).
- [EduTools 2010] EduTools (2010). CMS: Feature List. [http://www.edutools.info/feature\\_list.jsp?pj=4](http://www.edutools.info/feature_list.jsp?pj=4).
- [Eirinaki et al. 2006] Eirinaki, M., Mavroeidis, D., Tsatsaronis, G., e Vazirgiannis, M. (2006). Introducing Semantics in Web Personalization: The Role of Ontologies. *Lecture Notes in Computer Science*, 4289:147–162.
- [Faria et al. 2008] Faria, M. N., Malva, G. R. O., Dorça, F. A., Lopes, R. S., e Fernandes, M. A. (2008). Um sistema de avaliação baseado em lógica fuzzy. *Horizonte Científico*, 1(8). <http://www.seer.ufu.br/index.php/horizontecientifico/article/view/3971>.

- [Felder 2010] Felder, R. M. (2010). Frequently Asked Questions about the ILS. <http://www4.ncsu.edu/unity/lockers/users/f/felder/public/ILSdir/ILS-faq.htm>.
- [Felder e Silverman 1988] Felder, R. M. e Silverman, L. K. (1988). Learning and Teaching Styles In Engineering Education. *Engr. Education*, 78(7):674–681. <http://www4.ncsu.edu/unity/lockers/users/f/felder/public/Papers/LS-1988.pdf>.
- [Felder e Soloman 2010] Felder, R. M. e Soloman, B. A. (2010). Index of Learning Styles Questionnaire. <http://www.engr.ncsu.edu/learningstyles/ilsweb.html>.
- [for Biomedical Informatics Research 2010a] for Biomedical Informatics Research, S. C. (2010a). Protégé. <http://protege.stanford.edu>.
- [for Biomedical Informatics Research 2010b] for Biomedical Informatics Research, S. C. (2010b). Protégé: user documentation. <http://protege.stanford.edu/doc/users.html>.
- [Friesen et al. 2003] Friesen, N., Fisher, S., Roberts, A., Hesemeier, S., e Habkirk, S. (2003). CanCore Guidelines for the Implementation of Learning Object Metadata. Version 2.0. *CanCore*. <http://cancore.athabascau.ca/en/guidelines.html>.
- [Gao et al. 2009] Gao, S. S., Sperberg-McQueen, C. M., e Henry S. Thompson, E. (2009). W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures. *W3C Working Draft*. <http://www.w3.org/TR/xmlschema11-1/>.
- [Gibbons et al. 2000] Gibbons, A. S., Nelson, J., e Richards, R. (2000). The nature and origin of instructional objects. *The Instructional Use of Learning Objects: Online Version*. <http://reusability.org/read/chapters/gibbons.doc>.
- [Gruber 2009] Gruber, T. (2009). Ontology. In Liu, L. e Özsu, M. T. (editores), *Encyclopedia of Database Systems*. Springer-Verlag. Disponível em <http://tomgruber.org/writing/ontology-definition-2007.htm>.
- [Gruber 1993] Gruber, T. R. T. (1993). Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In Guarino, N. e Poli, R. (editores), *Formal Ontology in Conceptual Analysis and Knowledge Representation*. Kluwer Academic Publishers.
- [Hardesty 2010] Hardesty, L. (2010). Toward the Semantic Web. *MIT News Office*. <http://web.mit.edu/newsoffice/2010/semantic-web-0622.html>.
- [Hendler 2001] Hendler, J. (2001). Agents and the Semantic Web. *IEEE Intelligent Systems*, 16(2):30–37.
- [Horridge 2009] Horridge, M. (2009). *A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools*. The University Of Manchester. Oxford, New York, USA.
- [Horrocks et al. 2004] Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosz, B., e Dean, M. (2004). SWRL: A Semantic Web Rule Language Combining OWL and RuleML. *W3C Member Submission*. <http://www.w3.org/Submission/SWRL/>.
- [IMS Global Learning Consortium 2010] IMS Global Learning Consortium, I. (2010). Content Packaging Specification. <http://www.imsglobal.org/copyright.html>.

- [Initiativ 2010] Initiativ, D. C. M. (2010). DCMI Specifications. <http://dublincore.org/specifications/>.
- [Jade 2010] Jade (2010). JADE - Java Agent DEvelopment Framework. <http://jade.tilab.com/>.
- [Kemczinski 2005] Kemczinski, A. (2005). Método de avaliação para ambientes e-learning. Master's thesis, Universidade Federal de Santa Catarina.
- [Knublauch 2004] Knublauch, H. (2004). Ontology-Driven Software Development in the Context of the Semantic Web: An Example Scenario with Protégé/OWL. In *in Annex XVII (7) , and*.
- [Knublauch 2010] Knublauch, H. (2010). Protege-OWL API Programmer's Guide. [http://protegewiki.stanford.edu/wiki/ProtegeOWL\\_API\\_Programmers\\_Guide](http://protegewiki.stanford.edu/wiki/ProtegeOWL_API_Programmers_Guide).
- [Kolb 1999] Kolb, D. A. (1999). The Kolb Learning Style Inventory, Version 3. *Boston: Hay Group*.
- [Lassila e R. Swick 1999] Lassila, O. e R. Swick, E. (1999). Resource Description Framework (RDF) Model and Syntax Specifications. *World Wide Web Consortium*. <http://www.w3.org/TR/REC-rdf-syntax/>.
- [Learning 2010a] Learning, A. A. D. (2010a). ADL SCORM 2004 Reload Editor. <http://www.adlnet.gov/Technologies/scorm/SCORMSDocuments/2004%204th%20Edition/Reload>.
- [Learning 2010b] Learning, A. A. D. (2010b). SCORM Documents. [http://www.adlnet.gov/Technologies/scorm/SCORMSDocuments/Forms/All Documents.aspx](http://www.adlnet.gov/Technologies/scorm/SCORMSDocuments/Forms/All_Documents.aspx).
- [MacGuinnes e F. Harmelen 2004] MacGuinnes, D. L. e F. Harmelen, E. (2004). OWL Web Ontology Language. *W3C Recommendation*. <http://www.w3.org/TR/owl-features/>.
- [Manola e Eric Miller 2004] Manola, F. e Eric Miller, E. (2004). RDF Primer. *W3C Recommendation*. <http://www.w3.org/TR/rdf-primer/>.
- [Masie 2001] Masie, E. (2001). E-learning com e de experiência. *Revista TIMASTER*. [http://www.timaster.com.br/revista/artigos/main\\_artigo.asp?codigo=38](http://www.timaster.com.br/revista/artigos/main_artigo.asp?codigo=38).
- [Melis et al. 2001] Melis, E., Andrès, E., Büdenbender, J., Frishauf, A., Goguadse, G., Libbrecht, P., Pollet, M., e Ullrich, C. (2001). ActiveMath: A web-based learning environment. *International Journal of Artificial Intelligence in Education*, 12(4):85–407.
- [Moodle 2010a] Moodle (2010a). Development: database schema introduction. [http://docs.moodle.org/en/Development:Database\\_schema\\_introduction/](http://docs.moodle.org/en/Development:Database_schema_introduction/).
- [Moodle 2010b] Moodle (2010b). Development: progress tracking. [http://docs.moodle.org/en/Development:Progress\\_tracking/](http://docs.moodle.org/en/Development:Progress_tracking/).
- [Moodle 2010c] Moodle (2010c). Development: quiz database structure. [http://docs.moodle.org/en/Development:Quiz\\_database\\_structure](http://docs.moodle.org/en/Development:Quiz_database_structure).

- [Moodle 2010d] Moodle (2010d). Moodle. <http://www.moodle.org/>.
- [Moodle 2010e] Moodle (2010e). Moodle 1.8 Data Structure. [http://moodle.org/file.php/5/moddata/forum/28/376698/moodle\\_1.8\\_data\\_structure.rar](http://moodle.org/file.php/5/moddata/forum/28/376698/moodle_1.8_data_structure.rar).
- [Moodle 2010f] Moodle (2010f). Moodle Statistics. <http://moodle.org/stats/>.
- [Muñoz 2004] Muñoz, L. S. (2004). Ontology-based Metadata for e-learning Content. Master's thesis, Universidade Federal do Rio Grande do Sul. <http://alexandria.tue.nl/extra2/200213488.pdf>.
- [Neto 2006] Neto, W. C. B. (2006). Web semântica na construção de sistemas de aprendizagem adaptativos. Master's thesis, Universidade Federal de Santa Catarina.
- [Nilsson et al. 2003] Nilsson, M., Palmr, M., e Brase, J. (2003). The LOM RDF binding - principles and implementation. In *Proceedings of the 3rd annual ARIADNE conference*. Katholieke Universiteit Leuven. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.12.5758>.
- [Noy e McGuinness 2001] Noy, N. e McGuinness, D. L. (2001). Ontology development 101: a guide to creating your first ontology. *Stanford University*. [http://protege.stanford.edu/publications/ontology\\_development/ontology101-noy-mcguinness.html](http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html).
- [O'Connor et al. 2005] O'Connor, M., Knublauch, H., Tu, S., Grosz, B., Dean, M., Grosso, W., e Musen, M. (2005). Supporting Rule System Interoperability on the Semantic Web with SWRL. *The Semantic Web - ISWC 2005*, pp. 974–986.
- [OMDoc 2010] OMDoc (2010). OMDoc - Open Mathematical Documents. <https://trac.omdoc.org/OMDoc/>.
- [Oppermann et al. 1997] Oppermann, R., Rashev, R., e Kinshuk (1997). Adaptability and adaptivity in learning systems. *Knowledge Transfer (Ed. A. Behrooz)*, 2:173–179.
- [Patel-Schneider et al. 2009] Patel-Schneider, P. F., Grau, B. C., e Boris Motik, E. (2009). OWL Web Ontology Language. *W3C Recommendation*. <http://www.w3.org/TR/owl2-direct-semantics/>.
- [Patrick Hayes 2004] Patrick Hayes, E. (2004). RDF Semantics. *W3C Recommendation*. <http://www.w3.org/TR/rdf-mt/>.
- [Peterson et al. 2009] Peterson, D., Gao, S. S., Malhotra, A., Sperberg-McQueen, C. M., e Henry S. Thompson, E. (2009). W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes. *W3C Working Draft*. <http://www.w3.org/TR/xmlschema11-2/>.
- [Prud'hommeaux e Andy Seaborne 2008] Prud'hommeaux, E. e Andy Seaborne, E. (2008). SPARQL Query Language for RDF. *W3C Recommendation*. <http://www.w3.org/TR/rdf-sparql-query/>.
- [Smith et al. 2004] Smith, M. K., Welty, C., e D. L. McGuinness, E. (2004). OWL Web Ontology Language Guide. *W3C Recommendation*. <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>.

- [Tarouco et al. 2003] Tarouco, L. M. R., Fabre, M.-C. J. M., e Tamusiunas, F. R. (2003). Reusabilidade de objetos educacionais. *RENOTE: revista novas tecnologias na educação. UFRGS, Centro Interdisciplinar de Novas Tecnologias na Educação*.
- [Teleduc 2010] Teleduc (2010). Teleduc. <http://www.teleduc.org.br/>.
- [Tim Berners-Lee 2001] Tim Berners-Lee, James A. Hendler, O. L. (2001). The Semantic Web. *Scientific American*, 284(5):34–43. <http://www4.ncsu.edu/unity/lockers/users/f/felder/public/Papers/LS-1988.pdf>.
- [Unicode 2010] Unicode, I. (2010). What is Unicode? <http://www.unicode.org/standard/WhatIsUnicode.html>.
- [URI Planning Interest Group 2001] URI Planning Interest Group, W. (2001). URIs, URLs, and URNs: Clarifications and Recommendations 1.0. *W3C Note*. <http://www.w3.org/TR/uri-clarification>.
- [W3C 2010a] W3C (2010a). RDF current status. [http://www.w3.org/standards/techs/rdf#w3c\\_all](http://www.w3.org/standards/techs/rdf#w3c_all).
- [W3C 2010b] W3C (2010b). XML - eXtensible Markup Language. <http://www.w3.org/XML/>.
- [Wiki 2010] Wiki, P. (2010). Protege 4.0 Features. <http://protegewiki.stanford.edu/wiki/Protege4Features>.
- [Wiley 2000] Wiley, D. A. (2000). Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. *The Instructional Use of Learning Objects: Online Version*. <http://reusability.org/read/chapters/wiley.doc>.
- [Wooldridge e Jennings 1994] Wooldridge, M. e Jennings, N. R. (1994). *Intelligent Agents: Theory and Practice*. Knowledge Engineering Review.
- [Wu 2002] Wu, H. (2002). A reference architecture for adaptive hypermedia applicationsa. Master's thesis, Universiteit Eindhoven. <http://alexandria.tue.nl/extra2/200213488.pdf>.

# Apêndice A

## Ontologia LOM

```
<?xml version="1.0"?>
```

```
<!DOCTYPE rdf:RDF [  
  <!ENTITY dcterms "http://purl.org/dc/terms/" >  
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >  
  <!ENTITY swrl "http://www.w3.org/2003/11/swrl#" >  
  <!ENTITY swrlb "http://www.w3.org/2003/11/swrlb#" >  
  <!ENTITY dc "http://purl.org/dc/elements/1.1/" >  
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >  
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >  
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >  
  <!ENTITY protege "http://protege.stanford.edu/plugins/owl/protege#" >  
  <!ENTITY xsp "http://www.owl-ontologies.com/2005/08/07/xsp.owl#" >  
>
```

```
<rdf:RDF xmlns="http://moodle.unipam.edu.br/lom.owl#"  
  xml:base="http://moodle.unipam.edu.br/lom.owl"  
  xmlns:dc="http://purl.org/dc/elements/1.1/"  
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"  
  xmlns:swrl="http://www.w3.org/2003/11/swrl#"  
  xmlns:protege="http://protege.stanford.edu/plugins/owl/protege#"  
  xmlns:xsp="http://www.owl-ontologies.com/2005/08/07/xsp.owl#"  
  xmlns:owl="http://www.w3.org/2002/07/owl#"  
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"  
  xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"  
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
```

```

    xmlns:dcterms="http://purl.org/dc/terms/#">
<owl:Ontology rdf:about=""/>
<owl:AllDifferent>
    <owl:distinctMembers rdf:parseType="Collection">
        <rdf:Description rdf:about="#VeryLow"/>
        <rdf:Description rdf:about="#Low"/>
        <rdf:Description rdf:about="#Medium"/>
        <rdf:Description rdf:about="#High"/>
        <rdf:Description rdf:about="#VeryHigh"/>
    </owl:distinctMembers>
</owl:AllDifferent>
<InteractivityType rdf:ID="Active"/>
<owl:Class rdf:ID="Classification">
    <rdfs:subClassOf rdf:resource="#LearningObject"/>
</owl:Class>
<owl:DatatypeProperty rdf:ID="description">
    <rdfs:domain rdf:resource="#Classification"/>
    <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>
<owl:Class rdf:ID="Diagram">
    <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
    <owl:disjointWith rdf:resource="#Exam"/>
    <owl:disjointWith rdf:resource="#Exercise"/>
    <owl:disjointWith rdf:resource="#Figure"/>
    <owl:disjointWith rdf:resource="#Graph"/>
    <owl:disjointWith rdf:resource="#Index"/>
    <owl:disjointWith rdf:resource="#Lecture"/>
    <owl:disjointWith rdf:resource="#NarrativeText"/>
    <owl:disjointWith rdf:resource="#ProblemStatement"/>
    <owl:disjointWith rdf:resource="#Questionnaire"/>
    <owl:disjointWith rdf:resource="#SelfAssessment"/>
    <owl:disjointWith rdf:resource="#Simulation"/>
    <owl:disjointWith rdf:resource="#Slide"/>
    <owl:disjointWith rdf:resource="#Table"/>
</owl:Class>
<owl:Class rdf:ID="Difficulty">
    <rdfs:subClassOf rdf:resource="#Educational"/>
</owl:Class>
<owl:Class rdf:ID="Educational">

```

```

    <rdfs:subClassOf rdf:resource="#LearningObject"/>
</owl:Class>
<owl:Class rdf:ID="Exam">
    <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
    <owl:disjointWith rdf:resource="#Diagram"/>
    <owl:disjointWith rdf:resource="#Exercise"/>
    <owl:disjointWith rdf:resource="#Figure"/>
    <owl:disjointWith rdf:resource="#Graph"/>
    <owl:disjointWith rdf:resource="#Index"/>
    <owl:disjointWith rdf:resource="#Lecture"/>
    <owl:disjointWith rdf:resource="#NarrativeText"/>
    <owl:disjointWith rdf:resource="#ProblemStatement"/>
    <owl:disjointWith rdf:resource="#Questionnaire"/>
    <owl:disjointWith rdf:resource="#SelfAssessment"/>
    <owl:disjointWith rdf:resource="#Simulation"/>
    <owl:disjointWith rdf:resource="#Slide"/>
    <owl:disjointWith rdf:resource="#Table"/>
</owl:Class>
<owl:Class rdf:ID="Exercise">
    <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
    <owl:disjointWith rdf:resource="#Diagram"/>
    <owl:disjointWith rdf:resource="#Exam"/>
    <owl:disjointWith rdf:resource="#Figure"/>
    <owl:disjointWith rdf:resource="#Graph"/>
    <owl:disjointWith rdf:resource="#Index"/>
    <owl:disjointWith rdf:resource="#Lecture"/>
    <owl:disjointWith rdf:resource="#NarrativeText"/>
    <owl:disjointWith rdf:resource="#ProblemStatement"/>
    <owl:disjointWith rdf:resource="#Questionnaire"/>
    <owl:disjointWith rdf:resource="#SelfAssessment"/>
    <owl:disjointWith rdf:resource="#Simulation"/>
    <owl:disjointWith rdf:resource="#Slide"/>
    <owl:disjointWith rdf:resource="#Table"/>
</owl:Class>
<InteractivityType rdf:ID="Expositive"/>
<owl:Class rdf:ID="Figure">
    <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
    <owl:disjointWith rdf:resource="#Diagram"/>
    <owl:disjointWith rdf:resource="#Exam"/>

```

```

    <owl:disjointWith rdf:resource="#Exercise"/>
    <owl:disjointWith rdf:resource="#Graph"/>
    <owl:disjointWith rdf:resource="#Index"/>
    <owl:disjointWith rdf:resource="#Lecture"/>
    <owl:disjointWith rdf:resource="#NarrativeText"/>
    <owl:disjointWith rdf:resource="#ProblemStatement"/>
    <owl:disjointWith rdf:resource="#Questionnaire"/>
    <owl:disjointWith rdf:resource="#SelfAssessment"/>
    <owl:disjointWith rdf:resource="#Simulation"/>
    <owl:disjointWith rdf:resource="#Slide"/>
    <owl:disjointWith rdf:resource="#Table"/>
  </owl:Class>
  <owl:Class rdf:ID="Geral">
    <rdfs:subClassOf rdf:resource="#LearningObject"/>
  </owl:Class>
  <owl:Class rdf:ID="Graph">
    <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
    <owl:disjointWith rdf:resource="#Diagram"/>
    <owl:disjointWith rdf:resource="#Exam"/>
    <owl:disjointWith rdf:resource="#Exercise"/>
    <owl:disjointWith rdf:resource="#Figure"/>
    <owl:disjointWith rdf:resource="#Index"/>
    <owl:disjointWith rdf:resource="#Lecture"/>
    <owl:disjointWith rdf:resource="#NarrativeText"/>
    <owl:disjointWith rdf:resource="#ProblemStatement"/>
    <owl:disjointWith rdf:resource="#Questionnaire"/>
    <owl:disjointWith rdf:resource="#SelfAssessment"/>
    <owl:disjointWith rdf:resource="#Simulation"/>
    <owl:disjointWith rdf:resource="#Slide"/>
    <owl:disjointWith rdf:resource="#Table"/>
  </owl:Class>
  <owl:ObjectProperty rdf:ID="hasPart">
    <rdfs:domain rdf:resource="#LearningObject"/>
    <owl:inverseOf rdf:resource="#isPartOf"/>
    <rdfs:range rdf:resource="#LearningObject"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="hasScale">
    <rdfs:type rdf:resource="#owl:FunctionalProperty"/>
    <rdfs:domain>

```

```

    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Difficulty"/>
        <owl:Class rdf:about="#InteractivityLevel"/>
        <owl:Class rdf:about="#SemanticDensity"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="#Scale"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasVersion">
  <rdfs:domain rdf:resource="#LearningObject"/>
  <owl:inverseOf rdf:resource="#isVersionOf"/>
  <rdfs:range rdf:resource="#LearningObject"/>
</owl:ObjectProperty>
<Scale rdf:ID="High"/>
<owl:Class rdf:ID="Index">
  <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
  <owl:disjointWith rdf:resource="#Diagram"/>
  <owl:disjointWith rdf:resource="#Exam"/>
  <owl:disjointWith rdf:resource="#Exercise"/>
  <owl:disjointWith rdf:resource="#Figure"/>
  <owl:disjointWith rdf:resource="#Graph"/>
  <owl:disjointWith rdf:resource="#Lecture"/>
  <owl:disjointWith rdf:resource="#NarrativeText"/>
  <owl:disjointWith rdf:resource="#ProblemStatement"/>
  <owl:disjointWith rdf:resource="#Questionnaire"/>
  <owl:disjointWith rdf:resource="#SelfAssessment"/>
  <owl:disjointWith rdf:resource="#Simulation"/>
  <owl:disjointWith rdf:resource="#Slide"/>
  <owl:disjointWith rdf:resource="#Table"/>
</owl:Class>
<owl:Class rdf:ID="InteractivityLevel">
  <rdfs:subClassOf rdf:resource="#Educational"/>
</owl:Class>
<owl:Class rdf:ID="InteractivityType">
  <rdfs:subClassOf rdf:resource="#Educational"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="isPartOf">

```

```

    <rdfs:domain rdf:resource="#LearningObject"/>
    <owl:inverseOf rdf:resource="#hasPart"/>
    <rdfs:range rdf:resource="#LearningObject"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="isRequiredBy">
    <rdfs:domain rdf:resource="#LearningObject"/>
    <owl:inverseOf rdf:resource="#requires"/>
    <rdfs:range rdf:resource="#LearningObject"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="isVersionOf">
    <rdfs:domain rdf:resource="#LearningObject"/>
    <owl:inverseOf rdf:resource="#hasVersion"/>
    <rdfs:range rdf:resource="#LearningObject"/>
</owl:ObjectProperty>
<owl:Class rdf:ID="LearningObject"/>
<owl:Class rdf:ID="LearningResourceType">
    <rdfs:subClassOf rdf:resource="#Educational"/>
</owl:Class>
<owl:Class rdf:ID="Lecture">
    <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
    <owl:disjointWith rdf:resource="#Diagram"/>
    <owl:disjointWith rdf:resource="#Exam"/>
    <owl:disjointWith rdf:resource="#Exercise"/>
    <owl:disjointWith rdf:resource="#Figure"/>
    <owl:disjointWith rdf:resource="#Graph"/>
    <owl:disjointWith rdf:resource="#Index"/>
    <owl:disjointWith rdf:resource="#NarrativeText"/>
    <owl:disjointWith rdf:resource="#ProblemStatement"/>
    <owl:disjointWith rdf:resource="#Questionnaire"/>
    <owl:disjointWith rdf:resource="#SelfAssessment"/>
    <owl:disjointWith rdf:resource="#Simulation"/>
    <owl:disjointWith rdf:resource="#Slide"/>
    <owl:disjointWith rdf:resource="#Table"/>
</owl:Class>
<Scale rdf:ID="Low"/>
<Scale rdf:ID="Medium"/>
<InteractivityType rdf:ID="Mixed"/>
<owl:Class rdf:ID="NarrativeText">
    <rdfs:subClassOf rdf:resource="#LearningResourceType"/>

```

```

    <owl:disjointWith rdf:resource="#Diagram"/>
    <owl:disjointWith rdf:resource="#Exam"/>
    <owl:disjointWith rdf:resource="#Exercise"/>
    <owl:disjointWith rdf:resource="#Figure"/>
    <owl:disjointWith rdf:resource="#Graph"/>
    <owl:disjointWith rdf:resource="#Index"/>
    <owl:disjointWith rdf:resource="#Lecture"/>
    <owl:disjointWith rdf:resource="#ProblemStatement"/>
    <owl:disjointWith rdf:resource="#Questionnaire"/>
    <owl:disjointWith rdf:resource="#SelfAssessment"/>
    <owl:disjointWith rdf:resource="#Simulation"/>
    <owl:disjointWith rdf:resource="#Slide"/>
    <owl:disjointWith rdf:resource="#Table"/>
</owl:Class>
<owl:Class rdf:ID="ProblemStatement">
    <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
    <owl:disjointWith rdf:resource="#Diagram"/>
    <owl:disjointWith rdf:resource="#Exam"/>
    <owl:disjointWith rdf:resource="#Exercise"/>
    <owl:disjointWith rdf:resource="#Figure"/>
    <owl:disjointWith rdf:resource="#Graph"/>
    <owl:disjointWith rdf:resource="#Index"/>
    <owl:disjointWith rdf:resource="#Lecture"/>
    <owl:disjointWith rdf:resource="#NarrativeText"/>
    <owl:disjointWith rdf:resource="#Questionnaire"/>
    <owl:disjointWith rdf:resource="#SelfAssessment"/>
    <owl:disjointWith rdf:resource="#Simulation"/>
    <owl:disjointWith rdf:resource="#Slide"/>
    <owl:disjointWith rdf:resource="#Table"/>
</owl:Class>
<owl:Class rdf:ID="Questionnaire">
    <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
    <owl:disjointWith rdf:resource="#Diagram"/>
    <owl:disjointWith rdf:resource="#Exam"/>
    <owl:disjointWith rdf:resource="#Exercise"/>
    <owl:disjointWith rdf:resource="#Figure"/>
    <owl:disjointWith rdf:resource="#Graph"/>
    <owl:disjointWith rdf:resource="#Index"/>
    <owl:disjointWith rdf:resource="#Lecture"/>

```

```

    <owl:disjointWith rdf:resource="#NarrativeText"/>
    <owl:disjointWith rdf:resource="#ProblemStatement"/>
    <owl:disjointWith rdf:resource="#SelfAssessment"/>
    <owl:disjointWith rdf:resource="#Simulation"/>
    <owl:disjointWith rdf:resource="#Slide"/>
    <owl:disjointWith rdf:resource="#Table"/>
  </owl:Class>
  <owl:ObjectProperty rdf:ID="requires">
    <rdfs:domain rdf:resource="#LearningObject"/>
    <owl:inverseOf rdf:resource="#isRequiredBy"/>
    <rdfs:range rdf:resource="#LearningObject"/>
  </owl:ObjectProperty>
  <owl:Class rdf:ID="Scale">
    <owl:equivalentClass>
      <owl:Class>
        <owl:oneOf rdf:parseType="Collection">
          <rdf:Description rdf:about="#High"/>
          <rdf:Description rdf:about="#Low"/>
          <rdf:Description rdf:about="#Medium"/>
          <rdf:Description rdf:about="#VeryHigh"/>
          <rdf:Description rdf:about="#VeryLow"/>
        </owl:oneOf>
      </owl:Class>
    </owl:equivalentClass>
    <rdfs:subClassOf rdf:resource="#Educational"/>
  </owl:Class>
  <owl:Class rdf:ID="SelfAssessment">
    <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
    <owl:disjointWith rdf:resource="#Diagram"/>
    <owl:disjointWith rdf:resource="#Exam"/>
    <owl:disjointWith rdf:resource="#Exercise"/>
    <owl:disjointWith rdf:resource="#Figure"/>
    <owl:disjointWith rdf:resource="#Graph"/>
    <owl:disjointWith rdf:resource="#Index"/>
    <owl:disjointWith rdf:resource="#Lecture"/>
    <owl:disjointWith rdf:resource="#NarrativeText"/>
    <owl:disjointWith rdf:resource="#ProblemStatement"/>
    <owl:disjointWith rdf:resource="#Questionnaire"/>
    <owl:disjointWith rdf:resource="#Simulation"/>
  </owl:Class>

```

```

        <owl:disjointWith rdf:resource="#Slide"/>
        <owl:disjointWith rdf:resource="#Table"/>
    </owl:Class>
    <owl:Class rdf:ID="SemanticDensity">
        <rdfs:subClassOf rdf:resource="#Educational"/>
    </owl:Class>
    <owl:Class rdf:ID="Simulation">
        <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
        <owl:disjointWith rdf:resource="#Diagram"/>
        <owl:disjointWith rdf:resource="#Exam"/>
        <owl:disjointWith rdf:resource="#Exercise"/>
        <owl:disjointWith rdf:resource="#Figure"/>
        <owl:disjointWith rdf:resource="#Graph"/>
        <owl:disjointWith rdf:resource="#Index"/>
        <owl:disjointWith rdf:resource="#Lecture"/>
        <owl:disjointWith rdf:resource="#NarrativeText"/>
        <owl:disjointWith rdf:resource="#ProblemStatement"/>
        <owl:disjointWith rdf:resource="#Questionnaire"/>
        <owl:disjointWith rdf:resource="#SelfAssessment"/>
        <owl:disjointWith rdf:resource="#Slide"/>
        <owl:disjointWith rdf:resource="#Table"/>
    </owl:Class>
    <owl:Class rdf:ID="Slide">
        <rdfs:subClassOf rdf:resource="#LearningResourceType"/>
        <owl:disjointWith rdf:resource="#Diagram"/>
        <owl:disjointWith rdf:resource="#Exam"/>
        <owl:disjointWith rdf:resource="#Exercise"/>
        <owl:disjointWith rdf:resource="#Figure"/>
        <owl:disjointWith rdf:resource="#Graph"/>
        <owl:disjointWith rdf:resource="#Index"/>
        <owl:disjointWith rdf:resource="#Lecture"/>
        <owl:disjointWith rdf:resource="#NarrativeText"/>
        <owl:disjointWith rdf:resource="#ProblemStatement"/>
        <owl:disjointWith rdf:resource="#Questionnaire"/>
        <owl:disjointWith rdf:resource="#SelfAssessment"/>
        <owl:disjointWith rdf:resource="#Simulation"/>
        <owl:disjointWith rdf:resource="#Table"/>
    </owl:Class>
    <owl:Class rdf:ID="Table">

```

```
<rdfs:subClassOf rdf:resource="#LearningResourceType"/>
<owl:disjointWith rdf:resource="#Diagram"/>
<owl:disjointWith rdf:resource="#Exam"/>
<owl:disjointWith rdf:resource="#Exercise"/>
<owl:disjointWith rdf:resource="#Figure"/>
<owl:disjointWith rdf:resource="#Graph"/>
<owl:disjointWith rdf:resource="#Index"/>
<owl:disjointWith rdf:resource="#Lecture"/>
<owl:disjointWith rdf:resource="#NarrativeText"/>
<owl:disjointWith rdf:resource="#ProblemStatement"/>
<owl:disjointWith rdf:resource="#Questionnaire"/>
<owl:disjointWith rdf:resource="#SelfAssessment"/>
<owl:disjointWith rdf:resource="#Simulation"/>
<owl:disjointWith rdf:resource="#Slide"/>
</owl:Class>
<Scale rdf:ID="VeryHigh"/>
<Scale rdf:ID="VeryLow"/>
</rdf:RDF>
```

# Apêndice B

## Ontologia do Domínio

```
<?xml version="1.0"?>
```

```
<!DOCTYPE rdf:RDF [  
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >  
  <!ENTITY swrl "http://www.w3.org/2003/11/swrl#" >  
  <!ENTITY swrlb "http://www.w3.org/2003/11/swrlb#" >  
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >  
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >  
  <!ENTITY lom "http://moodle.unipam.edu.br/lomv3.owl#" >  
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >  
  <!ENTITY protege "http://protege.stanford.edu/plugins/owl/protege#" >  
  <!ENTITY xsp "http://www.owl-ontologies.com/2005/08/07/xsp.owl#" >  
>
```

```
<rdf:RDF xmlns="http://moodle.unipam.edu.br/domain.owl#"  
  xml:base="http://moodle.unipam.edu.br/domain.owl"  
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"  
  xmlns:swrl="http://www.w3.org/2003/11/swrl#"  
  xmlns:protege="http://protege.stanford.edu/plugins/owl/protege#"  
  xmlns:lom="http://moodle.unipam.edu.br/lom.owl#"  
  xmlns:xsp="http://www.owl-ontologies.com/2005/08/07/xsp.owl#"  
  xmlns:owl="http://www.w3.org/2002/07/owl#"  
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"  
  xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"  
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">  
  <owl:Ontology rdf:about="">  
    <owl:imports rdf:resource="http://moodle.unipam.edu.br/lom.owl"/>
```

```

</owl:Ontology>
<owl:Class rdf:ID="Assessment"/>
<owl:Class rdf:ID="Category"/>
<owl:Class rdf:ID="Course">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasCategory"/>
      <owl:someValuesFrom rdf:resource="#Category"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasSection"/>
      <owl:someValuesFrom rdf:resource="#Section"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="#owl;Thing"/>
</owl:Class>
<owl:DatatypeProperty rdf:ID="domainDescription">
  <rdf:type rdf:resource="#owl;FunctionalProperty"/>
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Category"/>
        <owl:Class rdf:about="#Item"/>
        <owl:Class rdf:about="#ModuleType"/>
        <owl:Class rdf:about="#Organization"/>
        <owl:Class rdf:about="#SCORM"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range rdf:resource="#xsd:string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="fullname">
  <rdf:type rdf:resource="#owl;FunctionalProperty"/>
  <rdfs:domain rdf:resource="#Course"/>
  <rdfs:comment xml:lang="pt">
    >Nome completo do curso</rdfs:comment>
  <rdfs:range rdf:resource="#xsd:string"/>

```

```

</owl:DatatypeProperty>
<owl:ObjectProperty rdf:ID="hasAssessment">
  <rdfs:domain rdf:resource="#Module"/>
  <rdfs:range rdf:resource="#Assessment"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasCategory">
  <rdfs:domain rdf:resource="#Course"/>
  <rdfs:range rdf:resource="#Category"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasItem">
  <rdfs:domain rdf:resource="#Organization"/>
  <rdfs:range rdf:resource="#Item"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasLearningObject">
  <rdfs:range rdf:resource="&lom;LearningObject"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasModule">
  <rdfs:domain rdf:resource="#Section"/>
  <rdfs:range rdf:resource="#Module"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasOrganization">
  <rdfs:domain rdf:resource="#SCORM"/>
  <rdfs:range rdf:resource="#Organization"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasSCORM">
  <rdfs:domain rdf:resource="#Module"/>
  <rdfs:range rdf:resource="#SCORM"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasSection">
  <rdfs:domain rdf:resource="#Course"/>
  <rdfs:range rdf:resource="#Section"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasType">
  <rdfs:domain rdf:resource="#Module"/>
  <rdfs:range rdf:resource="#ModuleType"/>
</owl:ObjectProperty>
<owl:DatatypeProperty rdf:ID="id">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:range rdf:resource="&xsd;int"/>

```

```

</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="instance">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:domain rdf:resource="#Module"/>
  <rdfs:range rdf:resource="&xsd;int"/>
</owl:DatatypeProperty>
<owl:Class rdf:ID="Item">
  <rdfs:subClassOf rdf:resource="&owl;Thing"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasLearningObject"/>
      <owl:someValuesFrom rdf:resource="&lom;LearningObject"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Module">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasAssessment"/>
      <owl:someValuesFrom rdf:resource="#Assessment"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="&owl;Thing"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasType"/>
      <owl:someValuesFrom rdf:resource="#ModuleType"/>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasSCORM"/>
      <owl:someValuesFrom rdf:resource="#SCORM"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="ModuleType"/>
<owl:DatatypeProperty rdf:ID="order">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>

```

```

    <rdfs:domain>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Organization"/>
          <owl:Class rdf:about="#Section"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
    <rdfs:comment xml:lang="pt"
      >Ordem na sequ&#234;ncia de t&#243;picos de um curso</rdfs:comment>
    <rdfs:range rdf:resource="&xsd:int"/>
  </owl:DatatypeProperty>
  <owl:Class rdf:ID="Organization">
    <rdfs:subClassOf rdf:resource="&owl;Thing"/>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#hasItem"/>
        <owl:someValuesFrom rdf:resource="#Item"/>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:DatatypeProperty rdf:ID="performance">
    <rdf:type rdf:resource="&owl;FunctionalProperty"/>
    <rdfs:domain rdf:resource="#Assessment"/>
    <rdfs:range rdf:resource="&xsd;float"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="reference">
    <rdf:type rdf:resource="&owl;FunctionalProperty"/>
    <rdfs:domain rdf:resource="#SCORM"/>
    <rdfs:comment xml:lang="pt"
      >Nome do arquivo zip (pacote SCORM)</rdfs:comment>
    <rdfs:range rdf:resource="&xsd;string"/>
  </owl:DatatypeProperty>
  <owl:Class rdf:ID="SCORM">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#hasOrganization"/>
        <owl:someValuesFrom rdf:resource="#Organization"/>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>

```

```

    </rdfs:subClassOf>
    <rdfs:subClassOf rdf:resource="#owl;Thing"/>
  </owl:Class>
  <owl:Class rdf:ID="Section">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#hasModule"/>
        <owl:someValuesFrom rdf:resource="#Module"/>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf rdf:resource="#owl;Thing"/>
  </owl:Class>
  <owl:DatatypeProperty rdf:ID="shortname">
    <rdf:type rdf:resource="#owl;FunctionalProperty"/>
    <rdfs:domain rdf:resource="#Course"/>
    <rdfs:range rdf:resource="#xsd:string"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="timeModified">
    <rdf:type rdf:resource="#owl;FunctionalProperty"/>
    <rdfs:domain rdf:resource="#SCORM"/>
    <rdfs:range rdf:resource="#xsd:int"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="version">
    <rdf:type rdf:resource="#owl;FunctionalProperty"/>
    <rdfs:domain rdf:resource="#SCORM"/>
    <rdfs:comment xml:lang="pt"
      >vers&#227;o do pacote SCORM</rdfs:comment>
    <rdfs:range rdf:resource="#xsd:string"/>
  </owl:DatatypeProperty>
</rdf:RDF>

```

# Apêndice C

## Ontologia do Estudante

```
<?xml version="1.0"?>
```

```
<!DOCTYPE rdf:RDF [  
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >  
  <!ENTITY swrl "http://www.w3.org/2003/11/swrl#" >  
  <!ENTITY swrlb "http://www.w3.org/2003/11/swrlb#" >  
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >  
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >  
  <!ENTITY domain "http://moodle.unipam.edu.br/dominio3.owl#" >  
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >  
  <!ENTITY protege "http://protege.stanford.edu/plugins/owl/protege#" >  
  <!ENTITY xsp "http://www.owl-ontologies.com/2005/08/07/xsp.owl#" >  
>
```

```
<rdf:RDF xmlns="http://moodle.unipam.edu.br/student.owl#"   
  xml:base="http://moodle.unipam.edu.br/student.owl"   
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"   
  xmlns:swrl="http://www.w3.org/2003/11/swrl#"   
  xmlns:protege="http://protege.stanford.edu/plugins/owl/protege#"   
  xmlns:xsp="http://www.owl-ontologies.com/2005/08/07/xsp.owl#"   
  xmlns:owl="http://www.w3.org/2002/07/owl#"   
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"   
  xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"   
  xmlns:domain="http://moodle.unipam.edu.br/domain.owl#"   
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">  
  <owl:Ontology rdf:about="">
```

```

    <owl:imports rdf:resource="http://moodle.unipam.edu.br/domain.owl"/>
</owl:Ontology>
<rdf:Description rdf:about="&domain;hasLearningObject">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="&domain;Item"/>
        <owl:Class rdf:about="#User"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
</rdf:Description>
<rdf:Description rdf:about="&domain;id">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="&domain;Assessment"/>
        <owl:Class rdf:about="&domain;Category"/>
        <owl:Class rdf:about="&domain;Course"/>
        <owl:Class rdf:about="&domain;Item"/>
        <owl:Class rdf:about="&domain;Module"/>
        <owl:Class rdf:about="&domain;Organization"/>
        <owl:Class rdf:about="&domain;SCORM"/>
        <owl:Class rdf:about="&domain;Section"/>
        <owl:Class rdf:about="#User"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
</rdf:Description>
<owl:Class rdf:ID="Active">
  <rdfs:subClassOf rdf:resource="#InformationProcessing"/>
  <owl:disjointWith rdf:resource="#Reflective"/>
</owl:Class>
<owl:FunctionalProperty rdf:ID="city">
  <rdf:type rdf:resource="&owl;DatatypeProperty"/>
  <rdfs:domain rdf:resource="#User"/>
  <rdfs:range rdf:resource="&xsd:string"/>
</owl:FunctionalProperty>
<owl:FunctionalProperty rdf:ID="country">

```

```

    <rdf:type rdf:resource="&owl;DatatypeProperty"/>
    <rdfs:domain rdf:resource="#User"/>
    <rdfs:range rdf:resource="&xsd;string"/>
</owl:FunctionalProperty>
<owl:DatatypeProperty rdf:ID="email">
    <rdfs:domain rdf:resource="#User"/>
    <rdfs:range rdf:resource="&xsd;string"/>
</owl:DatatypeProperty>
<owl:Class rdf:ID="FelderSilverman">
    <rdfs:subClassOf rdf:resource="#LearningStyle"/>
    <owl:disjointWith rdf:resource="#Kolb"/>
</owl:Class>
<owl:Class rdf:ID="Global">
    <rdfs:subClassOf rdf:resource="#InformationOrganization"/>
    <owl:disjointWith rdf:resource="#Sequential"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="hasLearningStyle">
    <rdfs:domain rdf:resource="#User"/>
    <rdfs:range rdf:resource="#LearningStyle"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasPreference">
    <rdfs:domain rdf:resource="#User"/>
    <rdfs:range rdf:resource="#PreferenceUser"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasUser">
    <rdfs:domain rdf:resource="&domain;Course"/>
    <rdfs:range rdf:resource="#User"/>
</owl:ObjectProperty>
<owl:Class rdf:ID="InformationOrganization">
    <rdfs:subClassOf rdf:resource="#FelderSilverman"/>
</owl:Class>
<owl:Class rdf:ID="InformationPerception">
    <rdfs:subClassOf rdf:resource="#FelderSilverman"/>
</owl:Class>
<owl:Class rdf:ID="InformationPresentation">
    <rdfs:subClassOf rdf:resource="#FelderSilverman"/>
</owl:Class>
<owl:Class rdf:ID="InformationProcessing">
    <rdfs:subClassOf rdf:resource="#FelderSilverman"/>

```

```

</owl:Class>
<owl:Class rdf:ID="Intuitive">
  <rdfs:subClassOf rdf:resource="#InformationPerception"/>
  <owl:disjointWith rdf:resource="#Sensory"/>
</owl:Class>
<owl:Class rdf:ID="Kolb">
  <rdfs:subClassOf rdf:resource="#LearningStyle"/>
  <owl:disjointWith rdf:resource="#FelderSilverman"/>
</owl:Class>
<owl:Class rdf:ID="LearningStyle"/>
<owl:FunctionalProperty rdf:ID="name">
  <rdf:type rdf:resource="&owl;DatatypeProperty"/>
  <rdfs:domain rdf:resource="#User"/>
  <rdfs:range rdf:resource="&xsd:string"/>
</owl:FunctionalProperty>
<owl:Class rdf:ID="PreferenceUser"/>
<owl:Class rdf:ID="Reflective">
  <rdfs:subClassOf rdf:resource="#InformationProcessing"/>
  <owl:disjointWith rdf:resource="#Active"/>
</owl:Class>
<owl:Class rdf:ID="Sensory">
  <rdfs:subClassOf rdf:resource="#InformationPerception"/>
  <owl:disjointWith rdf:resource="#Intuitive"/>
</owl:Class>
<owl:Class rdf:ID="Sequential">
  <rdfs:subClassOf rdf:resource="#InformationOrganization"/>
  <owl:disjointWith rdf:resource="#Global"/>
</owl:Class>
<owl:Class rdf:ID="Student">
  <rdfs:subClassOf rdf:resource="#User"/>
</owl:Class>
<owl:Class rdf:ID="Teacher">
  <rdfs:subClassOf rdf:resource="#User"/>
</owl:Class>
<owl:Class rdf:ID="User"/>
<owl:FunctionalProperty rdf:ID="username">
  <rdf:type rdf:resource="&owl;DatatypeProperty"/>
  <rdfs:domain rdf:resource="#User"/>
  <rdfs:range rdf:resource="&xsd:string"/>

```

```
</owl:FunctionalProperty>
<owl:Class rdf:ID="Verbal">
  <rdfs:subClassOf rdf:resource="#InformationPresentation"/>
  <owl:disjointWith rdf:resource="#Visual"/>
</owl:Class>
<owl:Class rdf:ID="Visual">
  <rdfs:subClassOf rdf:resource="#InformationPresentation"/>
  <owl:disjointWith rdf:resource="#Verbal"/>
</owl:Class>
</rdf:RDF>
```

## Apêndice D

# Casos de testes baseando-se na seleção de uma organização no Pacote SCORM

Os testes foram realizados utilizando-se três usuários, nomeados aqui como: Estudante 1, Estudante 2 e Estudante 3.

Os parâmetros para processamento das regras de adaptação foram definidos com os seguintes valores:

```
pesoTempo = 1.0  
pesoComportamento = 4.0  
pesoConhecimento = 4.0  
pesoDesempenho = 1.0
```

Com base nestes pesos, após o processamento das regras, chega-se a um valor entre 0 e 5. Esse valor é utilizado para seleção de uma organização adequada para o estudante. No curso construído, conforme descrito anteriormente, há três organizações disponíveis. Assim, se o resultado final for menor que 3, o estudante deve estudar novamente a seção de aprendizagem, com a organização de menor complexidade (1). Se o resultado for entre 3 e 4, será apresentada a organização de complexidade intermediária (2). Por fim, se o resultado for maior que 4, será apresentada a organização de complexidade maior (no caso, 3).

Os parâmetros abaixo também podem ser utilizados, retirando-se a letra X que inicia cada um deles. No caso dos testes descritos, os valores abaixo não foram considerados; o sistema calcula um tempo médio, a partir da verificação do acesso dos alunos.

```
XtempoMinimoCurso32Secao1 = 300  
XtempoMaximoCurso32Secao1 = 600  
XforumMinimoCurso32Secao1 = 2  
XforumMaximoCurso32Secao1 = 10  
XchatMinimoCurso32Secao1 = 5  
XchatMaximoCurso32Secao1 = 20
```

Assim, inicialmente, após localizar uma nova matrícula, o sistema efetua uma cópia do curso para o estudante, selecionando-se a primeira seção de aprendizagem, com a organização de complexidade intermediária, como mostra a tabela D.1.

Tabela D.1: Seção e organização selecionada após a matrícula dos estudantes

| Estudante | Seção | Organização |
|-----------|-------|-------------|
| 1         | 1     | 2           |
| 2         | 1     | 2           |
| 3         | 1     | 2           |

Em seguida, após a conclusão da seção de aprendizagem, definida pela realização da avaliação pelo estudante, considera-se a avaliação do comportamento do estudante. No caso dos testes descritos aqui, foram apresentados os valores associados às variáveis para avaliação do comportamento presentes na tabela D.2.

Tabela D.2: Variáveis utilizadas para avaliação do comportamento

| Estudante | Variáveis          |              |             |
|-----------|--------------------|--------------|-------------|
|           | Tempo gasto (seg.) | Índice fórum | Índice chat |
| 1         | 874                | 5            | 22          |
| 2         | 620                | 3            | 10          |
| 3         | 255                | 1            | 0           |

São aplicadas sobre os valores, as regras de adaptação baseadas em lógica *fuzzy* presentes no sistema adaptativo Simeduc. Assim, com base nos valores das variáveis de entrada para avaliação do comportamento, obteve-se os resultados presentes na tabela D.3.

Tabela D.3: Valores obtidos para o comportamento após a aplicação de regras *fuzzy*

| Estudante | Tempo |        |        | Chat |         |       | Fórum |         |       |
|-----------|-------|--------|--------|------|---------|-------|-------|---------|-------|
|           | Lento | Normal | Rápido | Ruim | Regular | Ótimo | Ruim  | Regular | Ótimo |
| 1         | 0.25  | 0.43   | 0.00   | 0.70 | 0.03    | 0.00  | 0.72  | 1.00    | 0.83  |
| 2         | 0.00  | 1.00   | 0.00   | 0.04 | 1.00    | 0.90  | 0.07  | 1.00    | 0.83  |
| 3         | 0.00  | 0.00   | 0.56   | 1.00 | 0.00    | 0.00  | 0.67  | 0.11    | 0.83  |

Dessa forma, foram obtidos os valores referentes à avaliação do comportamento presentes na tabela D.4.

Tabela D.4: Valores referentes à avaliação do comportamento

| Estudante | Valores comportamento |         |       |      |           |
|-----------|-----------------------|---------|-------|------|-----------|
|           | baixo                 | regular | médio | bom  | excelente |
| 1         | 0.25                  | 0.07    | 0.43  | 0.43 | 0.00      |
| 2         | 0.00                  | 0.04    | 0.07  | 0.04 | 0.83      |
| 3         | 0.56                  | 0.11    | 0.00  | 0.00 | 0.00      |

Após a avaliação do comportamento, é feita a avaliação do conhecimento do estudante, baseando-se nos resultados da avaliação e no tempo gasto para realizá-la. As variáveis consideradas, com os respectivos valores obtidos nos testes, estão descritos na tabela D.5.

Tabela D.5: Variáveis utilizadas para avaliação do conhecimento

| Estudante | Questões |         |                        | Tempo (seg.) |        |
|-----------|----------|---------|------------------------|--------------|--------|
|           | Corretas | Erradas | Mais ou menos corretas | Gasto        | Máximo |
| 1         | 6        | 1       | 0                      | 227          | 420    |
| 2         | 3        | 1       | 3                      | 258          | 420    |
| 3         | 2        | 1       | 4                      | 109          | 420    |

Da mesma forma que na avaliação do conhecimento, são aplicadas sobre os valores as regras de adaptação baseadas em lógica *fuzzy* presentes no sistema adaptativo Simeduc. Assim, com base nos valores das variáveis de entrada para avaliação do conhecimento, obteve-se os resultados presentes na tabela D.6.

Tabela D.6: Valores obtidos para o conhecimento após a aplicação de regras *fuzzy*

| Estudante | Respostas |               |         | Tempo |          |      |
|-----------|-----------|---------------|---------|-------|----------|------|
|           | Corretas  | Mais ou Menos | Erradas | Baixo | Razoável | Alto |
| 1         | 0.86      | 0.14          | 0.00    | 0.00  | 1.00     | 0.00 |
| 2         | 0.43      | 0.14          | 0.43    | 0.00  | 0.90     | 0.04 |
| 3         | 0.29      | 0.14          | 0.57    | 0.40  | 0.06     | 0.00 |

Após o processamento, foram obtidos os valores referentes à avaliação do conhecimento, presentes na tabela D.7.

Tabela D.7: Valores referentes à avaliação do conhecimento

| Estudante | Valores conhecimento |         |       |      |           |
|-----------|----------------------|---------|-------|------|-----------|
|           | baixo                | regular | médio | bom  | excelente |
| 1         | 0.00                 | 0.00    | 0.14  | 0.00 | 0.86      |
| 2         | 0.04                 | 0.43    | 0.14  | 0.00 | 0.43      |
| 3         | 0.40                 | 0.06    | 0.06  | 0.14 | 0.29      |

Os valores obtidos para o conhecimento e o comportamento são utilizados para obtenção dos valores referentes ao desempenho do estudante, descritos na tabela D.8.

Por fim, na tabela D.9, são descritos os resultados finais, obtidos da seguinte forma:

- Tempo:
  1. se o tempo gasto estiver entre 20% a mais ou ao menos em relação ao tempo ótimo, considera-se o valor total (multiplica-se o peso da variável *tempo* por 10).

Tabela D.8: Valores referentes à avaliação do desempenho

| Estudante | Valores desempenho |         |       |      |           |
|-----------|--------------------|---------|-------|------|-----------|
|           | baixo              | regular | médio | bom  | excelente |
| 1         | 0.00               | 0.00    | 0.14  | 0.43 | 0.00      |
| 2         | 0.04               | 0.00    | 0.00  | 0.04 | 0.43      |
| 3         | 0.40               | 0.06    | 0.06  | 0.14 | 0.14      |

2. se estiver entre 50% a mais ou a menos, considera-se a metade do valor (divide-se o peso da variável *tempo* por 2 e multiplica-se por 10).
  3. Em outro caso, considera-se 0.
- Conhecimento, comportamento e desempenho: multiplica-se o resultado dessas variáveis pelos respectivos pesos.
  - Resultado final: corresponde à soma dos resultados anteriores, multiplicado por 5. O resultado da multiplicação é dividido por 100, chegando-se a uma nota entre 0 e 5, que definirá a organização a que o estudante terá acesso.

Tabela D.9: Resultados finais

| Estudante | Resultados |               |              |            |            |
|-----------|------------|---------------|--------------|------------|------------|
|           | Tempo      | Comportamento | Conhecimento | Desempenho | Nota Final |
| 1         | 10.0       | 16.0          | 29.6         | 5.5        | 3.055      |
| 2         | 5.0        | 29.6          | 18.8         | 7.3        | 3.035      |
| 3         | 5.0        | 4.8           | 16.4         | 3.4        | 1.48       |

Assim, conforme o resultado encontrado, os estudantes serão direcionados para as seções e organizações descritas na tabela D.10.

Tabela D.10: Seção e organização selecionada após os resultados de avaliação

| Estudante | Seção | Organização |
|-----------|-------|-------------|
| 1         | 2     | 2           |
| 2         | 2     | 2           |
| 3         | 1     | 1           |

## Apêndice E

### Casos de testes baseando-se na seleção dos objetos de aprendizagem

Inicialmente, foram cadastrados no sistema os estilos de aprendizagem de cinco estudantes, conforme Tabela E.1, sendo que o modelo 1, refere-se ao modelo proposto por [Felder e Silverman 1988], com a seguinte codificação para os estilos: 1-Ativo; 2-Reflexivo; 3-Sensorial; 4-Intuitivo; 5-Visual; 6-Verbal; 7-Sequencial; 8-Global.

Tabela E.1: Testes: estudantes e estilos de aprendizagem

| idUsuario | Modelo | Estilo | ValorEscala |
|-----------|--------|--------|-------------|
| 1         | 1      | 1      | 9           |
| 1         | 1      | 3      | 7           |
| 1         | 1      | 5      | 5           |
| 1         | 1      | 7      | 3           |
| 2         | 1      | 2      | 9           |
| 2         | 1      | 4      | 7           |
| 2         | 1      | 6      | 5           |
| 2         | 1      | 8      | 3           |
| 3         | 1      | 1      | 3           |
| 3         | 1      | 3      | 5           |
| 3         | 1      | 5      | 7           |
| 3         | 1      | 7      | 9           |
| 4         | 1      | 2      | 3           |
| 4         | 1      | 4      | 5           |
| 4         | 1      | 6      | 7           |
| 4         | 1      | 8      | 9           |

Após a definição dos dados presentes na tabela E.1, em que se procurou combinar várias possibilidades de estilos, de forma a testar as regras, foram criados objetos de aprendizagem que estabelecem relação com esses estilos. Para isso, selecionou-se o pacote referente ao primeiro tópico (seção de aprendizagem) do curso mostrado na seção 4.4. A relação de objetos de aprendizagem criados para testar a adaptação no nível da seleção de objetos está presente na Tabela E.2.

Tabela E.2: Testes: objetos de aprendizagem

| Descrição                                 | Tipo Interatividade | Tipo Recurso de Aprendizagem | Densidade Semântica | Grau Dificuldade |
|-------------------------------------------|---------------------|------------------------------|---------------------|------------------|
| Introdução; Introdução e Interface        | Ativo               | Exercício                    | Médio               | Médio            |
| Introdução; Introdução e Interface Layers | Expositivo          | Texto Narrativo              | Médio               | Médio            |
| Layers e Timeline                         | Misto               | Experiência                  | Baixo               | Baixo            |
| Layers e Timeline Ferramentas             | Misto               | Diagrama                     | Alto                | Alto             |
| Ferramentas                               | Misto               | Leitura                      | Alto                | Alto             |
| Ferramentas                               | Misto               | Figura                       | Baixo               | Baixo            |
| Ferramentas                               | Misto               | Problema                     | Baixo               | Baixo            |

Por meio do metadado Descrição, presente na categoria Classificação do padrão LOM, é possível associar os objetos a um ou mais itens presentes nas organizações. Como mostrado na tabela E.2, se estiverem associados a mais de um item, devem vir separados por ponto-e-vírgula. Assim, todos os objetos de aprendizagem elaborados devem ser anotados conforme mostrado na figura E.1, tendo em vista o processamento das regras de adaptação.

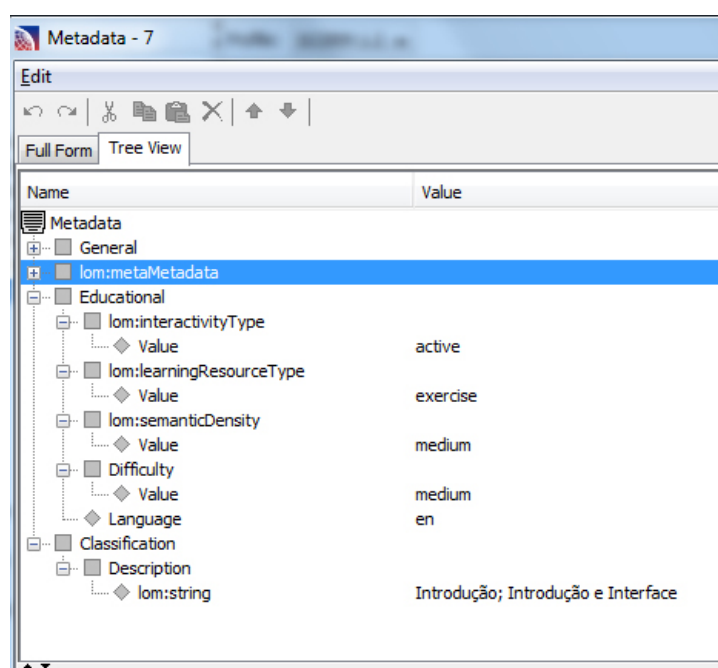


Figura E.1: Anotações em um objeto de aprendizagem utilizando a ferramenta Reload Editor

Como mostra a figura E.1, é preciso anotar apenas alguns elementos relacionados ao padrão LOM. No entanto, seria interessante anotar os demais elementos, já que o sistema pode ser estendido para que novas regras para adaptação de conteúdo possam ser definidas, utilizando outras anotações presentes no padrão, e assim, os objetos já

estariam disponíveis para serem utilizados. O arquivo *imsmanifest.xml* é processado pelo sistema adaptativo, criando as instâncias dos objetos de aprendizagem necessárias para o processamento das regras.

Assim, por exemplo, na primeira interação do estudante 1 com o sistema, como mostrado anteriormente, é selecionada a organização de complexidade intermediária (organização de número 2, conforme figura 4.11), que contém os seguintes itens:

1. Introdução e interface
2. Ferramentas
3. Layers e Timeline

Em primeiro lugar, seleciona-se os estilos de aprendizagem predominantes do estudante, de acordo com o mais alto valor. Assim, por exemplo, para o estudante 1, conforme apresentado na tabela E.1, o sistema retorna a seguinte relação de estilos:

- 1. Ativo
- 3. Sensorial
- 5. Visual
- 7. Sequencial

A partir dos dados, o sistema adaptativo é capaz de localizar os objetos de aprendizagem conforme as regras definidas. Assim, como o estilo de aprendizagem predominante do estudante 1 é *ativo*, seleciona-se o objeto anotado com essa característica, conforme figura E.1 e insere-se a informação de que esse objeto foi associado ao estudante. Após o processamento dos resultados, mesmo que o estudante tenha que rever a seção de aprendizagem em que consta o mesmo tópico, o sistema adaptativo verifica se existem objetos que ainda não foram apresentados e, caso exista, um novo objeto, que seja mais adequado aos seus estilos de aprendizagem é selecionado. Caso não exista, o objeto associado como padrão ao item da organização do pacote SCORM é apresentado.