

# **Uma Contribuição à Modelagem e Geração Automática de Conteúdos Educacionais**

**Vanessa Araujo Borges, Ellen Francine Barbosa**

<sup>1</sup>Instituto de Ciências Matemáticas e de Computação – ICMC/USP  
Caixa Postal 668 – 13.560-970 – São Carlos – SP – Brasil

{va.borges, francine}@icmc.usp.br

**Abstract.** *The development of educational modules – concise units of study, composed by theoretical and practical content which can be delivered to learners by using technological resources – has been extensively explored in the context of Teaching and Learning. In this paper we present and discuss supporting mechanisms for developing educational modules, emphasizing aspects of modeling and generating content. We present  $\text{ALM-CID}$  – an integrated approach for modeling educational content which considers different didactic perspectives. We also propose a collaborative tool that supports modeling and generating content based on the  $\text{ALM-CID}$ .*

**Resumo.** *Entre as linhas que têm sido objeto de pesquisa no contexto de Ensino e Aprendizado destaca-se o desenvolvimento de módulos educacionais – unidades de estudo, compostas por conteúdos teóricos e práticos, cuja disponibilização aos aprendizes é apoiada por recursos tecnológicos. Neste artigo são apresentados e discutidos mecanismos de apoio ao desenvolvimento de módulos educacionais, com ênfase nos aspectos de modelagem e geração dos conteúdos associados. É apresentada a  $\text{ALM-CID}$  – uma abordagem integrada para modelagem de conteúdos educacionais que considera diferentes perspectivas didáticas. Propõe-se, ainda, uma ferramenta colaborativa de apoio à modelagem e geração de conteúdos educacionais, com base na  $\text{ALM-CID}$ .*

## **1. Introdução**

A crescente disseminação da informação, aliada à facilidade de criação e disponibilização de material didático, com conteúdos interativos e diferentes mídias, tem proporcionado transformações significativas no processo educacional. O desenvolvimento de módulos educacionais – unidades concisas de estudo, compostas por conteúdos teóricos e práticos, cuja disponibilização aos aprendizes é apoiada por recursos tecnológicos e computacionais [Barbosa e Maldonado 2006] – insere-se nesse contexto.

Ferramentas e recursos tecnológicos e computacionais, em especial, ambientes e sistemas de apoio ao ensino e aprendizado, estão sendo adotados como mecanismos facilitadores em interações presenciais, híbridas (*blended*) e a distância. De modo geral, tais ambientes são adequados à estruturação e administração de cursos, à apresentação de conteúdos e à avaliação e acompanhamento do aprendiz. Além disso, possibilitam a condução de trabalhos colaborativos, por meio de um conjunto de ferramentas síncronas e assíncronas associadas. No entanto, apesar da diversidade de características e funcionalidades identificadas, uma limitação comum à maioria dos ambientes refere-se ao fato dos

mesmos concentrarem-se na criação da estrutura, armazenamento do conteúdo e controle de acesso ao material didático, deixando a tarefa de modelagem do conteúdo educacional inteiramente a cargo do autor do curso, sem qualquer tipo de suporte ou mesmo diretrizes e atividades sistemáticas que apóiem sua elaboração.

Tal limitação direciona-nos à uma linha de pesquisa relevante no contexto de ensino e aprendizado – a modelagem de conteúdos educacionais. No processo educacional, a modelagem representa uma atividade essencial para a estruturação do conhecimento, apoiando a identificação e a definição de conceitos e informações associadas, possibilitando, em última análise, que os mesmos sejam disponibilizados de modo coerente e ordenado, com base em teorias e princípios educacionais previamente definidos.

Adicionalmente à modelagem de conteúdos, o desenvolvimento de conteúdos educacionais livres, assim chamados devido a semelhanças com o processo de produção de software livre, tem se mostrado adequado por tratar questões essenciais para a produção de conteúdos confiáveis e de qualidade. Uma característica importante associada ao desenvolvimento de conteúdos livres está na possibilidade de sua construção de forma colaborativa, permitindo que equipes dispersas geograficamente possam contribuir efetivamente para sua elaboração. Além disso, a evolução contínua observada no desenvolvimento de software livre, em resposta às necessidades e opiniões dos usuários, também deve ser tratada na construção de conteúdos educacionais tendo-se em vista o caráter dinâmico e evolutivo associado ao conhecimento.

Frente a esse cenário, em trabalhos anteriores foram propostos alguns mecanismos de apoio à modelagem e desenvolvimento de módulos educacionais. Em especial, foi proposta a *ALM-CID* (Abordagem Integrada de Modelagem – Conceitual, Instrucional e Didática), na qual se estabelece um conjunto de modelos utilizados na estruturação do conteúdo [Barbosa e Maldonado 2006]. Em continuidade aos trabalhos realizados, pretende-se agora explorar a adequação/evolução dos mecanismos propostos no processo de desenvolvimento aberto, cooperativo e distribuído de módulos educacionais, nos contextos de educação presencial, híbrida e a distância. Em linhas gerais, a idéia é que os conteúdos construídos sejam interativos, customizáveis e reutilizáveis, contribuindo assim para a melhoria do processo de ensino e aprendizado, tornando-o compatível com as novas demandas impostas pelas transformações educacionais em curso.

Este trabalho tem como objetivo discutir os aspectos de adequação/evolução da abordagem *ALM-CID*, sintetizando as principais extensões propostas com respeito ao desenvolvimento aberto, colaborativo e distribuído de módulos educacionais. Considerando que a automatização da atividade de modelagem constitui um fator relevante no processo de desenvolvimento dos módulos, também são discutidos aspectos relacionados à proposição de uma ferramenta de apoio à modelagem e geração de conteúdos educacionais, com base na *ALM-CID*. A ferramenta *ALM Tool* apóia a construção colaborativa dos modelos propostos na abordagem, possibilitando ainda sua utilização na geração automática de conteúdos personalizados. Os conteúdos devem ser gerados segundo o padrão LOM [IEEE LTSC 2002], permitindo assim interoperabilidade, compartilhamento e reuso dos mesmos.

O restante do artigo está estruturado da seguinte forma. Na Seção 2 é apresentada uma visão geral da *ALM-CID*, com ênfase nos modelos que compõem a abordagem e

nas extensões propostas para sua evolução. A Seção 3 apresenta a ferramenta *AIM Tool*, discutindo aspectos de seu desenvolvimento, principais características e funcionalidades. Por fim, a Seção 4 sumariza os resultados obtidos até o momento e discute as perspectivas para a continuidade da pesquisa.

## 2. Abordagem Integrada para Modelagem de Conteúdos Educacionais

A abordagem *AIM-CID* (Abordagem Integrada de Modelagem – Conceitual, Instrucional e Didática) visa a reunir, em uma única proposta, diferentes perspectivas de modelagem, caracterizando assim um conjunto de modelos genéricos para a representação de conteúdos educacionais [Barbosa e Maldonado 2006]. A abordagem foi proposta em 2004, sendo adaptada e evoluída recentemente a fim de incorporar aspectos de desenvolvimento e aplicação distribuída e colaborativa dos módulos educacionais resultantes.

### 2.1. Modelo Conceitual

O modelo conceitual consiste em uma descrição em alto nível do domínio que se deseja ensinar. Sua construção envolve a definição dos conceitos relevantes para a compreensão do domínio, a especificação de como tais conceitos se relacionam e o estabelecimento de uma estrutura para sua representação. Para a construção do modelo conceitual, foi utilizada a técnica de Mapas Conceituais [Novak 1981]. Tal técnica foi selecionada por possibilitar uma representação gráfica adequada do domínio de conhecimento e, além disso, por tratar-se de ferramenta intuitiva mesmo para usuários não familiarizados.

Como uma extensão à caracterização de conceitos, foi introduzida na etapa de modelagem conceitual a possibilidade de importação de termos definidos em uma ontologia, mais especificamente, um arquivo no formato OWL (*Web Ontology Language*)<sup>1</sup> [W3C 2004]. A entrada por ontologias foi escolhida, sobretudo, devido ao fato de uma ontologia descrever um vocabulário comum de um dado domínio de conhecimento, por meio da definição do significado de termos e suas relações [Guarino 1998]. Esse vocabulário, disponível no formato OWL, ou seja, estruturado de maneira hierárquica, é capaz de representar integralmente as informações necessárias à modelagem conceitual.

É importante observar que a adoção de ontologias no nível conceitual proporciona um melhor entendimento e consequente compartilhamento e reúso dos conceitos e relacionamentos representados. Além disso, o desenvolvimento colaborativo e distribuído de conteúdos também pode ser facilitado pela utilização de um vocabulário comum a respeito do domínio de conhecimento em questão.

### 2.2. Modelo Instrucional

O modelo instrucional é responsável pela definição de informações adicionais relativas aos conceitos previamente identificados. Nessa etapa são definidos e modelados itens de informação e elementos instrucionais. Por itens de informação compreendem-se, além dos conceitos já identificados, todas as demais informações significativas ao domínio de conhecimento as quais também devem ser representadas e incorporadas ao conteúdo educacional [Barbosa e Maldonado 2006]. Várias teorias e técnicas podem ser utilizadas a fim de apoiar o refinamento do modelo conceitual. No caso da abordagem *AIM-CID*

---

<sup>1</sup>OWL especifica uma linguagem para definir e instanciar ontologias na Web.

foi considerada a teoria proposta por Merrill [Merrill 1983], sendo especificados no modelo instrucional os seguintes itens de informação: conceitos (já representados no modelo conceitual), fatos, procedimentos e princípios. Os elementos instrucionais, por sua vez, são informações adicionais que complementam os conceitos e demais itens de informação já definidos. Tais elementos são representados a fim de melhor compreender e assimilar o domínio de conhecimento em questão. Como elementos instrucionais tem-se: exemplos, exercícios, dicas, sugestões de estudo, ferramentas, simulações, entre outros [Barbosa e Maldonado 2006].

O modelo instrucional definido na abordagem *AIM-CID* utiliza como base o modelo HMBS (*Hyperdocument Model Based on Statecharts*) [Turine et al. 1997] – modelo para a especificação da estrutura semântica de hiperdocumentos, baseado na técnica de *Statecharts* [Harel 1987] – como modelo de especificação formal subjacente. Nesse caso, o modelo HMBS original foi desvinculado da representação de transições, eventos e mecanismos de história, e estendido para a representação de diferentes categorias de conhecimento (itens de informação e elementos instrucionais).

Como extensão ao modelo instrucional, foram caracterizadas as mídias que representam os itens de informação e elementos instrucionais. Tal caracterização teve como base a *ALOCoM-Ontology* (*Abstract Learning Object Content Model - Ontology*) [Verbert et al. 2004], uma ontologia que estabelece uma representação padrão para objetos de aprendizado<sup>2</sup> e seus componentes. A *ALOCoM-Ontology* descreve tipos de componentes de objetos de aprendizado, agrupando-os em três conjuntos [Verbert et al. 2004]: (1) fragmentos de conteúdo; (2) objetos de conteúdo; e (3) objetos de aprendizado. Para definir as mídias adequadas aos itens de informação e elementos instrucionais foi utilizado o conjunto fragmentos de conteúdo, que agrupa dois sub-conjuntos: (1) contínuo, caracterizado por animações, vídeos, áudios e simulações; e (2) descontínuo, caracterizado por textos, gráficos, links e imagens.

A especificação de mídias no nível instrucional da abordagem *AIM-CID*, em especial das mídias contínuas, constitui um fator relevante no desenvolvimento de conteúdos educacionais interativos, capazes de despertar no aprendiz o interesse pelos tópicos abordados, transformando-os em agentes no processo de construção do conhecimento.

### 2.3. Modelo Didático

O modelo didático é responsável por associar os objetos anteriormente modelados, estabelecendo uma seqüência de apresentação entre eles. A partir de um mesmo modelo instrucional é possível derivar diferentes modelos didáticos, ou seja, modelos didáticos correspondem a diferentes visões de um modelo instrucional. Desse modo, diferentes maneiras de utilização e disponibilização de conteúdo educacional, estabelecidas segundo objetivos do curso, estratégia pedagógica do professor e perfil de cada aprendiz podem ser representadas por meio de modelos didáticos distintos [Barbosa e Maldonado 2006].

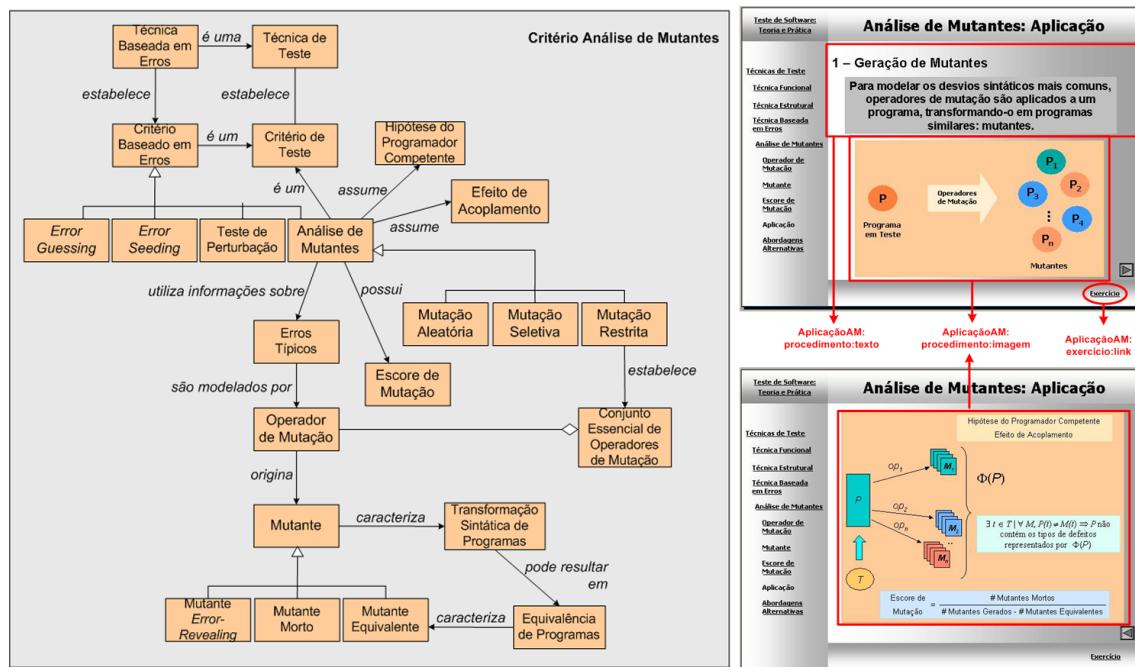
A construção do modelo didático também utiliza como base o HMBS, incorporando a ele a idéia de “especificação aberta” dos aspectos de navegação. Dependendo de fatores tais como duração do curso, objetivos de aprendizado e público-alvo, ma-

---

<sup>2</sup>Módulos educacionais podem ser desenvolvidos como uma combinação de objetos de aprendizado, os quais podem ser utilizados mais de uma vez, por mais de um autor, em seqüências e com objetivos diferentes.

neiras diferenciadas de apresentação e navegação pelo mesmo conteúdo são exigidas. Uma especificação aberta permite que todas as possíveis seqüências de apresentação do conteúdo sejam representadas no mesmo modelo didático. Nesse sentido, a partir de um único modelo, diversas implementações do mesmo conteúdo podem ser geradas em função dos fatores mencionados acima.

A abordagem *ATM-CID* vem sendo aplicada na construção de módulos educacionais nos domínios de teste de software, fundamentos de programação, matemática (ensino básico), software livre, sistemas embarcados, entre outros. A título de ilustração, na Figura 1 é apresentado o módulo desenvolvido no domínio de teste de software. O modelo conceitual associado também é considerado.



**Figura 1. Módulo educacional sobre teste de software.**

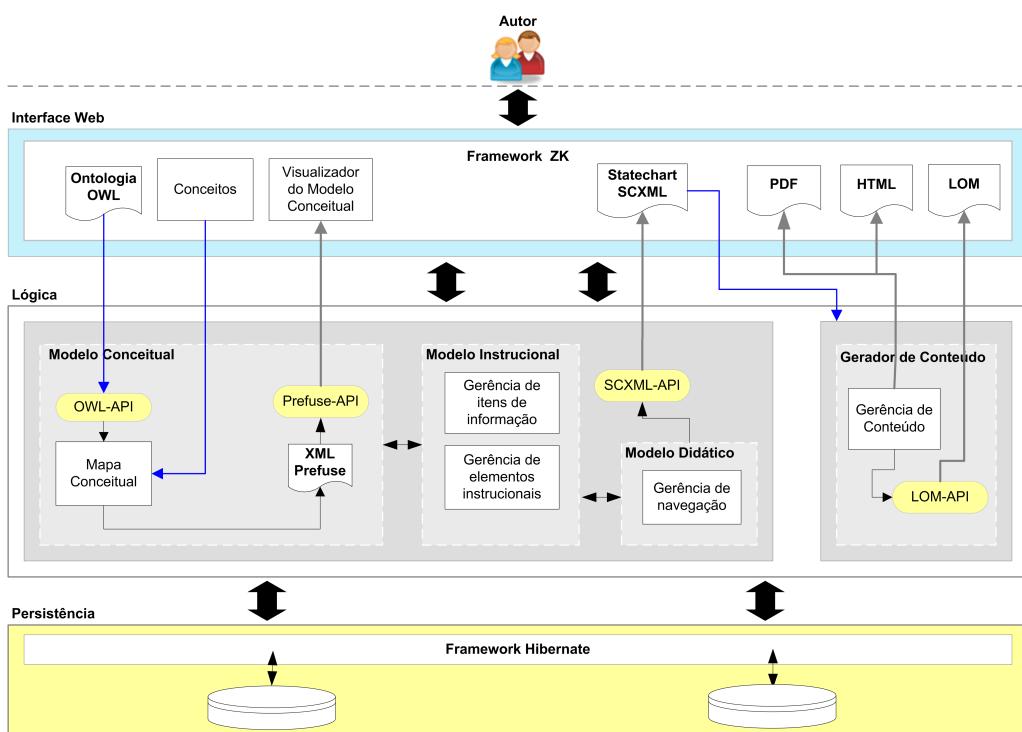
Cabe ressaltar, também, a aplicação dos mecanismos de modelagem propostos no contexto do *QualiPSO* (*Quality Platform for Open Source Software*) – projeto de cooperação internacional, apoiado pela comunidade europeia (IST-FP6-IP-034763), que tem como principal objetivo a definição e a implementação de tecnologias, procedimentos e políticas que apóiem as atuais práticas para o desenvolvimento de software livre. Todo o material de treinamento produzido no contexto do projeto vem sendo elaborado em conformidade com os modelos estabelecidos na *ATM-CID*. Os materiais desenvolvidos serão disponibilizados nos Centros de Competência em Software Livre, a serem estabelecidos na Europa (4), Brasil (1) e China (1). Além disso, os materiais de treinamento a serem desenvolvidos e disponibilizados no INCT-SEC (Instituto Nacional de Ciência e Tecnologia em Sistemas Embarcados Críticos) também serão elaborados em conformidade com a abordagem *ATM-CID*.

Por fim, é importante observar que todo o desenvolvimento dos módulos educacionais vem sendo realizado a partir da aplicação manual da abordagem *ATM-CID*, sendo esta uma tarefa trabalhosa e propensa a erros. De fato, faz-se fundamental a existência de ferramentas automatizadas, que dêem apoio à edição, interpretação e execução dos mode-

los propostos. Ainda, a partir de especificações “executáveis” é possível fornecer apoio à geração automática de conteúdos educacionais. Tal cenário serviu como motivação para a proposição da ferramenta *AIM Tool*, apresentada na próxima seção.

### 3. *AIM-Tool – Apoio Automatizado à Modelagem e Geração de Conteúdos*

A ferramenta *AIM Tool* tem como objetivo fornecer apoio automatizado à modelagem de conteúdos educacionais, com ênfase na construção colaborativa dos modelos conceitual, instrucional e didático propostos na abordagem *AIM-CID*. A construção colaborativa dos modelos utiliza a Web como meio de acesso e manipulação dos conteúdos. Desse modo, os autores podem modelar e produzir conteúdos educacionais em conjunto com outros autores, contribuindo para a disseminação do conhecimento e colaborando para a construção e evolução do conteúdo modelado. Adicionalmente, pretende-se que as especificações resultantes sejam utilizadas na geração automática de conteúdos, os quais poderão ser personalizados de acordo com interesses pedagógicos, perfil do aprendiz, duração do curso, público-alvo, entre outros. A *AIM Tool* vem sendo desenvolvida em Java, como uma aplicação Web. Sua arquitetura (Figura 2) é composta por três camadas, estabelecidas segundo o padrão MVC.



**Figura 2. Arquitetura da ferramenta *AIM Tool*.**

A interface da aplicação (primeira camada) utiliza o framework Ajax *ZK* [Open Source Ajax 2007], responsável por receber as requisições do autor (cliente) e enviá-las para a lógica de negócios (segunda camada). A interface da ferramenta fornece dois métodos para a entrada de conceitos: (1) por meio de um arquivo OWL, representando a entrada por ontologias; e (2) por meio da inserção de conceitos manualmente.

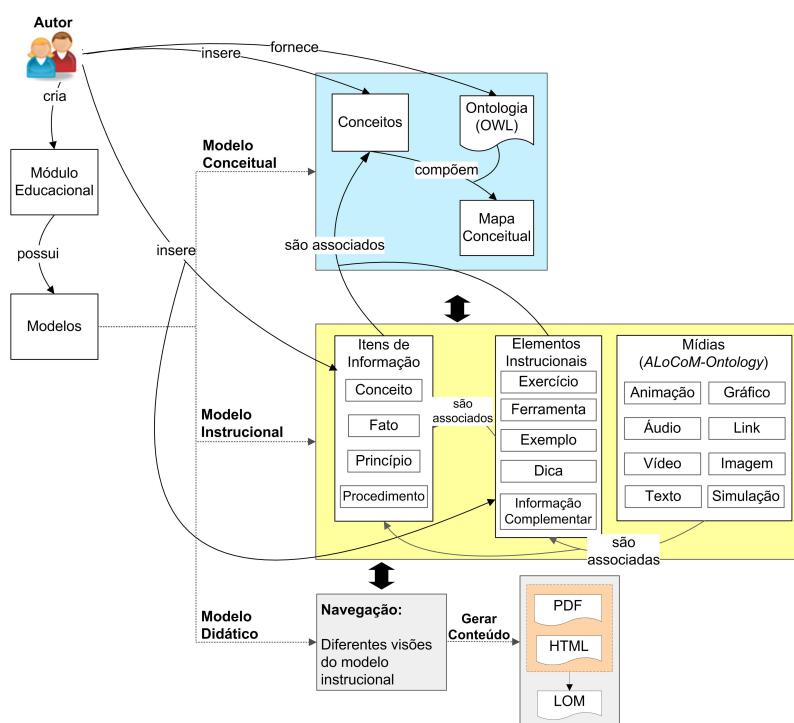
A lógica de negócios processa as requisições do cliente por meio da gerência dos modelos conceitual, instrucional e didático. O modelo conceitual recebe os conceitos do

arquivo OWL e processa-os utilizando a *OWL-API* [OWL API 2008]. Os conceitos processados pela *OWL-API*, e também aqueles inseridos manualmente, são convertidos em um modelo XML, que servirá de entrada para a *Prefuse-API* [Prefuse API 2008]. Esse *framework* possibilita a visualização dos conceitos por meio de uma estrutura hierárquica, que caracteriza uma topologia em árvore. O modelo instrucional é responsável por gerenciar os itens de informação e os elementos instrucionais. O modelo didático gerencia de que modo o conteúdo deve ser estruturado, permitindo ao autor personalizar o conteúdo em função de contextos de ensino e aprendizado distintos.

Ainda como parte da lógica de negócios, tem-se a gerência de conteúdos. Essa etapa é necessária ao tratamento das informações recebidas no modelo didático a fim de possibilitar a geração automática do conteúdo nos formatos HTML e/ou PDF. São utilizados: (1) *SCXML* (*State Machine Notation for Control Abstraction*) [W3C 2009], que gerencia o *Statechart* referente ao modelo didático, necessário ao módulo de geração de conteúdo; e (2) *LOM-API* (*Learning Object Metadata - Application Program Interface*), que gerencia as informações do material segundo a especificação do padrão LOM.

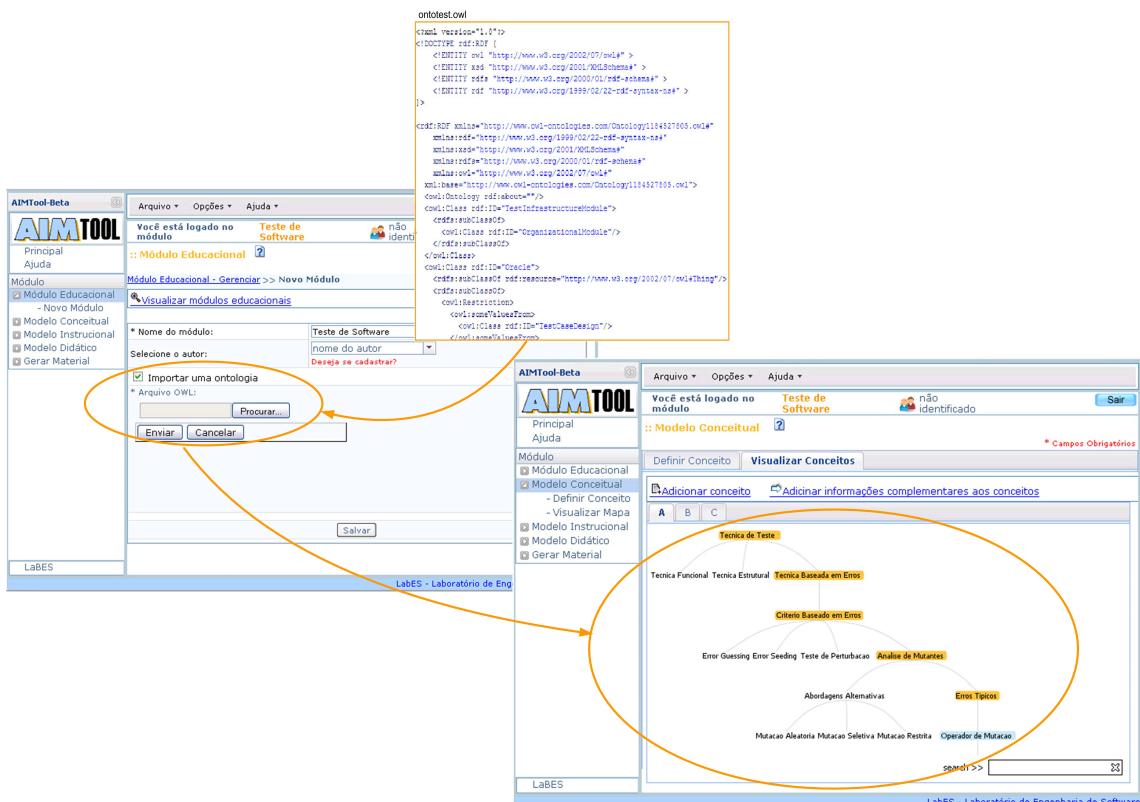
A terceira camada, responsável pela persistência dos dados, utiliza o *framework hibernate* [Hibernate 2006] – um *framework* para o mapeamento objeto-relacional (ORM – *Object-Relational Mapping*) para ambientes Java. Assim, todas as interações com o banco de dados são feitas por meio do *hibernate*.

A Figura 3 sintetiza o funcionamento da *ATM Tool*. Os conceitos são inseridos na ferramenta (modelagem conceitual) e associados a itens de informações e elementos instrucionais (modelagem instrucional), sendo estabelecida a seqüência de navegação entre eles (modelagem didática). O módulo de geração recebe a especificação resultante, gerando automaticamente o conteúdo, o qual é empacotado segundo o padrão LOM.



**Figura 3. Funcionamento da *ATM Tool*.**

A Figura 4 ilustra a importação de ontologias, via um arquivo no formato OWL, como apoio à caracterização de conceitos durante a geração do modelo conceitual. A visualização do modelo (ontologia) também é apresentada. No caso, é ilustrada a importação da *OntoTest* [Barbosa et al. 2008] – uma ontologia de teste de software, utilizada como apoio à definição e estruturação dos conceitos referentes a este domínio.

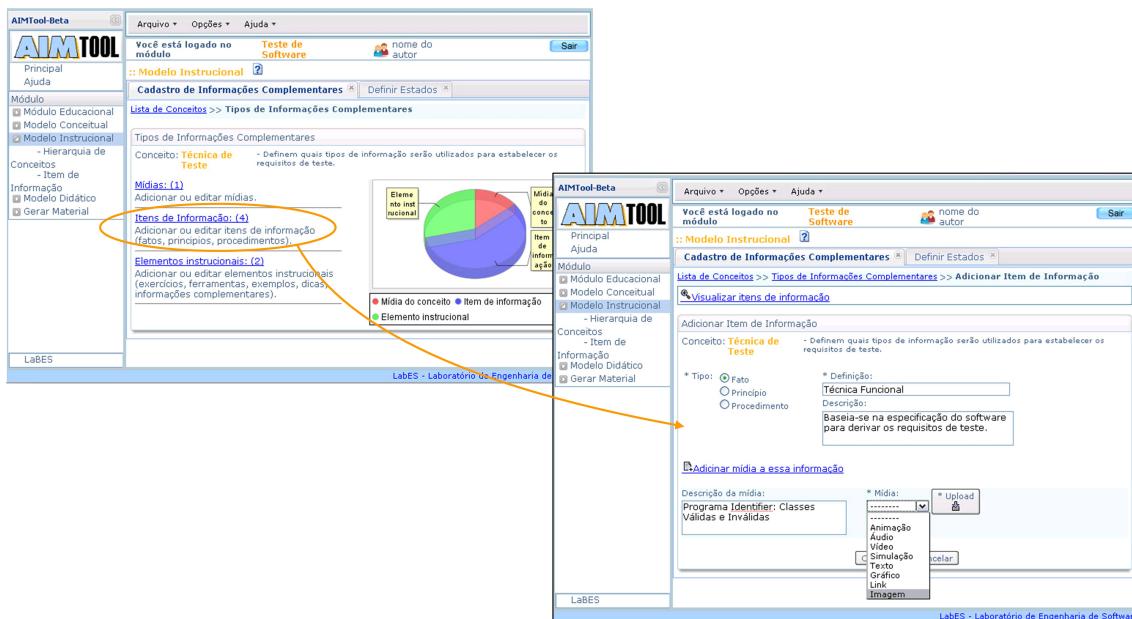


**Figura 4.** *AIM Tool*: Importação de ontologias na construção do modelo conceitual.

A Figura 5 ilustra a adição de itens de informação ao modelo instrucional. Observa-se que as mídias associadas aos elementos instrucionais foram categorizadas em conformidade com a *ALOCOM-Ontology* [Verbert et al. 2004], conforme discutido anteriormente. O modelo didático é construído de maneira similar aos modelos conceitual e instrucional. Por limitações de espaço, sua construção não é ilustrada.

A *AIM Tool* encontra-se em fase final de desenvolvimento. A representação e visualização dos modelos conceitual, instrucional e didático já foi concluída e, atualmente, vem sendo finalizada a implementação do módulo de geração automática de conteúdos.

Ressalta-se que a *AIM Tool* vem sendo desenvolvida sob a concepção de conteúdos educacionais livres. Desse modo, é possível modelar e gerar automaticamente conteúdos de qualidade, evoluídos continuamente, que foram produzidos de forma colaborativa e distribuída por meio da contribuição de diversos autores (geograficamente dispersos ou não).



**Figura 5.** *AIM Tool*: Adição de itens de informação ao modelo instrucional.

#### 4. Conclusões e Trabalhos Futuros

Neste trabalho foram explorados mecanismos de apoio ao desenvolvimento aberto, colaborativo e distribuído de módulos educacionais, dando-se ênfase à atividade de modelagem dos conteúdos associados aos módulos. Foram abordados aspectos de adequação/evolução da abordagem *AIM-CID*, sobretudo quanto ao uso de ontologias na etapa de modelagem conceitual e à definição das mídias que representam os objetos instrucionais na etapa de modelagem instrucional. Também foram discutidos aspectos referentes à automatização da abordagem por meio do desenvolvimento da ferramenta *AIM Tool* – uma ferramenta de apoio à modelagem e geração automática de conteúdos educacionais, baseada na abordagem *AIM-CID*. Observa-se que os aspectos investigados contribuem para a melhoria tanto do processo de desenvolvimento como dos próprios módulos resultantes, de modo a torná-los personalizados, flexíveis e reusáveis, capazes de despertar o interesse do aprendiz pelos tópicos em estudo. Educação presencial, híbrida e a distância podem ser beneficiadas nessa perspectiva.

Como trabalhos futuros, pretende-se realizar a validação da ferramenta na construção automática de um módulo educacional no domínio de teste de software. Deverão ser utilizados como apoio a *Ontotest* [Barbosa et al. 2008] e os módulos já desenvolvidos manualmente. Os mecanismos discutidos neste artigo também poderão ser explorados na construção de módulos em outros domínios de conhecimento, os quais poderão ser aplicados tanto em disciplinas de ensino básico como superior.

#### 5. Agradecimentos

Os autores agradecem o apoio financeiro da FAPESP, CNPq, CAPES e Projeto *QualiPSO*.

#### Referências

Barbosa, E. F. e Maldonado, J. C. (2006). An integrated content modeling approach for educational modules. In *IFIP 19th World Computer Congress – International Conference on Computer Information Systems*.

*rence on Education for the 21st Century*, p. 17–26, Santiago, Chile.

- Barbosa, E. F., Nakagawa, E. Y., Riekstin, A. C., e Maldonado, J. C. (2008). Ontology-based development of testing related tools. In *20th Int. Conference on Software Engineering and Knowledge Engineering (SEKE 2008)*, p. 697–702, San Francisco, CA.
- Guarino, N. (1998). *Formal ontology in information systems*. IOS Press.
- Harel, D. (1987). Statecharts: A visual formalism for complex systems. *Science of Computer Programming*, 8:231–274.
- Hibernate (2006). Hibernate – Relational Persistence for Java and .NET. <http://www.hibernate.org>, Acessado em: Junho/2009.
- IEEE LTSC (2002). Learning Object Metadata (LOM).
- Merrill, M. D. (1983). Component display theory. In *Instructional Design Theories and Models: An Overview of their Current States*, Hillsdale, NJ. Lawrence Erlbaum.
- Novak, J. D. (1981). *Uma Teoria de Educação*. Editora Pioneira, São Paulo.
- Open Source Ajax (2007). ZK – Open Source Ajax. <http://www.zkoss.org>, Acessado em: Junho/2009.
- OWL API (2008). OWL API – University of Manchester. <http://owlapi.sourceforge.net>, Acessado em: Novembro/2008.
- Prefuse API (2008). Prefuse API – information visualization toolkit. <http://prefuse.org>, Acessado em: Janeiro/2009.
- Turine, M. A. S., Oliveira, M. C. F., e Masiero, P. C. (1997). Designing structured hypertext with HMBS. In *VIII International ACM Hypertext Conference (Hypertext 97)*, p. 241–256, Southampton, UK.
- Verbert, K., Jovanovic, J., Gasevic, D., Duval, E., e Meire, M. (2004). Towards a Global Component Architecture for Learning Objects: A Slide Presentation Framework. In *Proc. Of the 17th ED-MEDIA*.
- W3C (2004). OWL web ontology language. <http://www.w3.org/TR/owl-features>, Acessado em: Fevereiro/2009.
- W3C (2009). Commons SCXML. <http://commons.apache.org/scxml>, Acessado em: Junho/2009.