

Meta-modelos para Aplicações de Hipermissão Adaptativa e Meta-adaptação

Patricia Seefelder de Assis*
e-mail: patricia@iprj.uerj.br

Daniel Schwabe
e-mail: dschwabe@inf.puc-rio.br

Simone Diniz Junqueira Barbosa
e-mail: simone@inf.puc-rio.br

PUC-RioInf.MCC45/04 Dezembro, 2004

Abstract: The emergence of meta-adaptive systems seems to be a natural evolution process in adaptive systems. Meta-adaptive systems are able to vary the adaptation technique based on various factors. This work analyzes the main aspects related to adaptivity by proposing a reference meta-model for AH Systems. This model is a first step towards the development of a general meta-model intended to be the basis for a meta-adaptive system.

Keywords: Hypermedia, Adaptability, Semantic Web, Models, e-Learning.

Resumo. A meta-adaptação parece ser a evolução natural na área de hipermissão adaptativa. Sistemas meta-adaptativos são capazes de variar a técnica de adaptação de acordo com vários fatores. Este trabalho analisa os principais aspectos relacionados à adaptação, propondo um meta-modelo de referência para Sistemas de Hipermissão Adaptativa (Adaptive Hypermedia Systems). Este modelo pretende ser o primeiro passo na construção de um meta-modelo geral que sirva como base para sistemas meta-adaptativos.

Palavras-chave: Aplicações Hipermissão, Adaptabilidade, Web Semântica, Modelagem, Educação à Distância.

* Departamento de Informática, PUC-Rio e Instituto Politécnico, Campus Regional de Nova Friburgo;
Membro do Grupo de Pesquisa Rede Operativa de Conhecimento e Aprendizagem (ROCA) / UERJ

1 Introdução

O uso da Internet na educação à distância e no comércio eletrônico, entre outras aplicações, está amplamente difundido. A WWW, desde a sua concepção, é um sistema hipermídia, conceituado essencialmente como uma rede de nós conectados por elos. O desenvolvimento de boas aplicações hipermídia neste contexto pressupõe a ampla exploração de suas possibilidades, ou seja, uma rede rica em conexões e informações que permita ao usuário navegar livremente pelo hiperespaço. Não há sentido em se utilizar aplicações hipermídia apenas de forma linear ou como se estivesse lendo um livro, seqüencialmente.

Diversos estudos na área de Hipermídia Adaptativa têm sido realizados com o objetivo de facilitar o uso de aplicações hipermídia, adequando-as às características individuais de utilização. “Sistemas Hipermídia Adaptativos” (*Adaptive Hypermedia Systems*, AHSs) são aplicações hipermídia que podem mudar algumas de suas características baseados em vários fatores, tais como: identidade do usuário, papel do usuário, dispositivo de acesso, localização física, histórico de navegação. De uma maneira geral, AHSs utilizam um Modelo do Usuário (*User Model*, UM) e um Modelo de Adaptação (*Adaptation Model*, AM) para mudar a apresentação, a navegação ou o conteúdo disponibilizado de acordo, basicamente, com as características do usuário e do modo como está sendo feita a navegação. Em geral, um Modelo do Domínio (*Domain Model*, DM) é utilizado para estruturar o conteúdo.

A percepção de que a própria seleção do tipo de técnica de adaptação a ser empregado, e seus correspondentes mecanismos, deve se adequar aos usuários tem voltado o foco das pesquisas para sistemas hipermídia meta-adaptativos. A concepção dos sistemas meta-adaptativos é que o tipo de adaptação também é adaptado em função de diversos fatores como, por exemplo, a classe dos usuários.

Este trabalho propõe um meta-modelo para AHSs que serve como referência para a comparação e análise de tais sistemas. Além disto, o meta-modelo proposto permite identificar os principais aspectos relacionados à adaptação servindo como base para um posterior desenvolvimento de um meta-modelo geral que contemple também a meta-adaptação.

O trabalho está organizado da seguinte forma: na seção 2 apresentamos um breve resumo das técnicas de adaptação mais conhecidas. Em seguida (seção 3), detalhamos um pouco mais o objeto de estudo deste trabalho. Na seção 4 abordamos as principais questões de adaptação. O meta-modelo de referência é apresentado na seção 5 e é instanciado para os principais AHSs na seção 6. Uma comparação entre estes é feita na seção 7. Por fim, apresentamos a conclusão e trabalho futuros (seções 8 e 9).

2 Técnicas de Adaptação

[Brusilovsky 96a] define algumas técnicas de adaptação, reconhecidas pela maioria dos autores. Entre as mais utilizadas, podemos citar:

- inclusão condicional de fragmentos (*conditional fragments*): é feita uma seleção de quais informações devem ser mostradas;
- visita guiada ou orientação direta (*direct guidance*): o sistema indica o melhor caminho a seguir através de um botão (*button*) “próximo”, por exemplo;

- anotação de elos (*link annotation*): as âncoras dos elos são apresentadas com aparências diferentes para indicar o grau de relevância. Uma metáfora bastante utilizada é a do “sinal de trânsito” (*traffic light metaphor*), onde a cor vermelha indica que os elos não devem ser explorados, a cor verde é usada para elos recomendados e a cor amarela é utilizada para indicar elos que podem ser visitados, mas não são recomendados;
- elos escondidos (*link hiding*): é um caso especial de anotação de elos. O elo existe, mas a âncora não pode ser identificada como tal - tem a mesma aparência que o texto;
- ordenação de elos (*link sorting*): os elos são apresentados em ordem decrescente de relevância. A desvantagem desta técnica é que a mesma página pode aparecer diferentemente a cada acesso do usuário.

3 Objeto de Estudo

Em [Brusilovsky 03], o autor aponta para a necessidade de meta-adaptação. Segundo ele, as técnicas de adaptação têm sido o foco das pesquisas em hipermídia adaptativa, mas esta visão tem evoluído para a percepção de que a própria seleção do tipo de técnica a ser empregado, e seus correspondentes mecanismos, deve se adaptar aos usuários, em função de diversos fatores. Como exemplo, podemos citar as evidências fornecidas pelo referido trabalho de que para usuários com pouco ou nenhum conhecimento sobre o domínio, técnicas de adaptação mais restritivas, que limitam o espaço de navegação, tais como, visita guiada ou ocultação de elos, são mais relevantes. Por outro lado, usuários com bom conhecimento prévio sobre o domínio parecem preferir métodos de adaptação não restritivos, tais como anotação e geração de elos. Assim, a tendência é o desenvolvimento de sistemas hipermídia meta-adaptativos que deverão possibilitar o uso de diferentes técnicas de adaptação, dispondo de informações sobre os limites e a aplicabilidade de cada uma delas, de modo a serem capazes de selecionar, de maneira adaptativa, a melhor técnica de acordo com os usuários e a situação de uso.

O enfoque utilizado neste trabalho é a análise das principais questões de adaptação como abordadas por diferentes AHSs. Dada a variedade de propostas e de modelos de AHSs, é interessante que se possa sistematizar um modo de análise e comparação dos sistemas ou modelos propostos. Para tanto, é apresentado um meta-modelo que funciona como referência, provendo uma base comum para esta comparação. Para avaliar a relevância do meta-modelo proposto, alguns AHSs foram estudados sob a ótica do mesmo. Um objetivo ulterior é desenvolver um meta-modelo geral que ajude a estruturar sistemas meta-adaptativos.

Os atuais sistemas de hipermídia são verdadeiros *frameworks* para desenvolvimento de cursos baseados na *Web*, alguns incluindo até ferramentas de autoria. Dentre os *frameworks* mais destacados, são abordados neste trabalho: o **AHA!** ([De Bra & Ruitter 01], [De Bra & Stash 02], [De Bra et al 02a e 02b]), de propósito geral, e, na área educacional, o **InterBook** ([Eklund et al 97], [Brusilovsky et al 98]), o **NetCoach** ([NetCoach Manual 03], [Weber & Specht 97], [Weber & Brusilovsky 01]) e o **KBS** ([Fröhlich et al 98], [Henze & Nejd 99a], [Henze et al 99] [Nejd & Wolpers 99]). O **AHAM** (*Adaptive Hypermedia Application Model*) [De Bra et al 99], um modelo para a arquitetura de aplicações hipermídia adaptativas, também é avaliado.

[Medina-Medina et al 02] descreve uma taxonomia para AHSs baseada em quatro questões consideradas como fundamentais para a adaptação: “O quê?”, “Quem?”,

“Como?” e “Quando?” (*What? Whom? How? When?*). No referido trabalho, os tipos de AHSs são classificados de acordo com critérios associados às respostas para as quatro questões consideradas. Entretanto, os conceitos envolvidos na adaptação não ficam claramente evidenciados. A construção de um meta-modelo de referência objetiva distinguir tais conceitos. Sob nosso ponto de vista, a questão de para “quem” a adaptação é feita é mais bem expressa pela idéia de “em função de quê” ela ocorre. Explicitando, consideramos que AHSs baseiam-se em quatro questões principais:

- O quê é adaptado;
- Em função de quê a adaptação é feita (que aspectos são levados em consideração);
- Como a adaptação é feita (como os aspectos considerados para a adaptação são capturados, quais mecanismos são usados);
- Em que granularidade de tempo (quando) a adaptação ocorre.

As questões de adaptação são abordadas na próxima seção. Cada questão tem várias respostas. Alguns AHSs consideram todas as respostas, enquanto outros abordam apenas algumas. Para a apreciação das características adaptativas dos sistemas, apenas os aspectos diretamente relacionados com a adaptação da apresentação são levados em consideração no meta-modelo apresentado; algumas questões de interface e implementação não são abordadas. Como já visto, o que se deseja é determinar o quê é adaptado, em função de quê e como é feita esta adaptação e em que momentos esta adaptação ocorre.

O meta-modelo de referência para AHSs proposto é baseado no modelo do AHAM e no “Munich Reference Model for Adaptive Hypermedia Systems” [Koch 00] (um modelo de referência para sistemas hipermídia adaptativos) e é especificado através da notação do diagrama de classes da UML [UML Resource Center 02].

4 Questões de adaptação

4.1 O quê

Segundo [Brusilovsky 96a], de uma maneira geral, sistemas hipermídia consistem de páginas conectadas por elos, podendo incluir índices e um mapa global com elos para todas as páginas disponíveis. Cada página contém algumas informações locais e elos para páginas relacionadas. Por conseguinte, a adaptação pode ocorrer no conteúdo das páginas (*content-level adaptation*) e nos elos: das páginas, dos índices e dos mapas (*link-level adaptation*). Via de regra [Brusilovsky 96b], o primeiro tipo de adaptação (nível de conteúdo) é utilizado para atender a diferentes classes de usuários, sendo classificado como adaptação da apresentação. Já a adaptação que ocorre nos elos é considerada como adaptação do suporte à navegação, já que seu objetivo é auxiliar a navegação dos usuários, evitando que eles se percam no hiperespaço. O suporte à navegação pode ocorrer através da limitação do espaço de navegação, da sugestão dos elos mais relevantes e da anotação dos elos [Brusilovsky 03].

[Koch 00] considera, ainda, uma adaptação no nível da apresentação, englobando as alterações no *layout* que não afetam o conteúdo. No referido trabalho, são considerados os seguintes tipos de adaptação:

- adaptação de conteúdo (seleção diferenciada de informação, pela utilização, por exemplo, de conteúdos alternativos e adição ou ocultação de conteúdo);

- adaptação de navegação (mudanças nas aparências dos elos, nos destinos dos elos, no número de elos ou na ordem em que estes elos são apresentados ao usuário).
- adaptação de apresentação (*layouts* diferentes dos elementos de interface tais como: tipo de mídia; ordenação; cores; tipo da fonte; tamanho das imagens).

Neste trabalho, busca-se uma classificação mais precisa do que pode ser adaptado, ou seja do resultado real da adaptação: qual aspecto da aplicação muda. A adaptação do teor do conteúdo é diferenciada da adaptação da estrutura do conteúdo, esta última considerada como um tipo de adaptação de navegação. Outro tipo de adaptação de navegação considerado é a adaptação da topologia do hiperespaço (alterações nos destinos dos elos ou no número de elos). Em relação à apresentação (interface), separa-se as alterações na aparência do conteúdo das alterações na aparência dos elos. Vale notar que a navegação é considerada como sendo composta por nós e elos, onde um nó representa uma “visão” (como na área de banco de dados) sobre a classe conceitual correspondente e, analogamente, os elos sobre as relações conceituais. A idéia geral baseia-se no método OOHDM (veja [Rossi et al 99] para uma discussão detalhada), proporcionando uma separação clara entre conteúdo, navegação e interface. A Tabela 1 apresenta essa classificação.

Tabela 1. “O quê”

O QUÊ É ADAPTADO		DESCRIÇÃO e EXEMPLOS
Conteúdo		O conteúdo propriamente dito. <u>Exemplos:</u> texto coloquial para leigos vs. texto técnico para especialistas; textos simples para novatos vs. textos detalhados para usuários experientes
Navegação	estrutura do conteúdo	Nós - a maneira pela qual os conceitos são colocados juntos para propósitos de navegação <u>Exemplos:</u> inclusão da introdução para novatos vs. pular introdução para usuários avançados; inclusão de advertências de segurança da primeira vez que um conteúdo crítico é visitado vs. a não inclusão em visitas subsequentes
	topologia do hiperespaço	Elos - alterações nas âncoras e índices modificam os caminhos de navegação. <u>Exemplos:</u> inclusão de um elo para a “solução” de um problema utilizado como exemplo vs. a não inclusão se o problema for utilizado como teste.
Apresentação	interface do conteúdo	Fragmentos. <u>Exemplos:</u> utilização de fontes de tamanhos diferentes de acordo com a idade do usuário; uso de realce para enfatizar certos tipos de informação.
	interface de suporte à navegação	Âncoras. <u>Exemplos:</u> utilização de menu <i>drop-down</i> vs. utilização de lista explícita de âncoras; uso de âncoras textuais vs. uso de ícones como âncoras

4.2 Em função de quê

Tabela 2. “Em função de quê”

ASPECTO	EXPLICAÇÃO E EXEMPLO
Modelo do Domínio (<i>Domain Model</i>)	A maneira como o conteúdo é estruturado no DM pode afetar a adaptação. <u>Exemplo:</u> um conceito (hierarquicamente) composto ser considerado como aprendido somente quando todos os conceitos elementares que o compõem tiverem sido aprendidos.
Perfil do Usuário (<i>profile</i>) (armazenado no UM)	Preferências e características do usuário; papel desempenhado por ele. <u>Exemplos:</u> se o usuário é novato ou experiente; se ele prefere textos ou imagens; em que língua ele prefere acessar; se ele é estudante ou professor; se ele se identificou ou está acessando como anônimo; se ele gosta mais de acessar textos inteiros ou resumos; conhecimento dele sobre o domínio; objetivos dele.
Contexto ¹ (<i>User Context</i>) (Situação de uso)	Sob que condições o sistema é utilizado. <u>Exemplos:</u> se o usuário está navegando a partir de um computador pessoal ou a partir de um telefone celular; de qual lugar (tais como: rua, bairro, cidade, casa, universidade, escritório) o acesso está sendo feito; qual a banda de acesso.
Histórico de Navegação (<i>User Navigational Behaviour</i>)	Caminho que o usuário percorre enquanto navega. <u>Exemplo:</u> páginas acessadas previamente.
Histórico de Interação (<i>User Browsing Behaviour</i>)	As ações que o usuário executa (interações) enquanto navega. <u>Exemplos:</u> quais artigos o usuário só lê e quais imprime; quantas vezes ele vê um vídeo; que imagens ele amplia.
Utilização da Funcionalidade da Aplicação (<i>Application Functionality Usage</i>)	Como a funcionalidade da aplicação está sendo usada. <u>Exemplos:</u> resultados de testes; itens comprados previamente.
Granularidade de Tempo (<i>Granularity of Time</i>) (Quando)	A adaptação é sempre feita em função do tempo. A questão é em que granularidade este tempo é considerado, podendo ir desde uma adaptação pontual até episódica até contínua. <u>Exemplos:</u> adaptação feita no <i>login</i> (customização); adaptação após uma configuração do usuário; revisão e adaptação do sistema a cada passo da navegação.

Esta questão diz respeito aos vários parâmetros que podem ser usados para determinar a adaptação. A Tabela 2 detalha os aspectos básicos levados em consideração na adaptação.

¹ O termo “contexto”, neste trabalho, é empregado no sentido de situação / ambiente de uso e não no sentido mais amplo de “gerência de contexto”, onde o contexto seria tudo (perfil, navegação, dispositivo, localização).

4.3 Como

Esta questão diz respeito aos mecanismos utilizados para capturar os aspectos de implementação levados em consideração para a adaptação:

- Modelo do Usuário;
- Modelo de Adaptação;
- Agente de entrada (*input*);
- Agente de atualização (*update*).

Cada sistema tem uma maneira específica de implementar um ou mais destes mecanismos. O Modelo do Usuário pode ser implementado, por exemplo, como pares atributo-valor [Wu et al 00a e 00b]; modelo bayesiano [Henze & Nejd, 99b]; abordagem probabilística e episódica (*probabilistic and episodic approach*) [Specht & Oppermann, 98]; rede de Petri [Medina-Medina et al 02]; redes semânticas; arquivos de transações (*log files*); tabela em bancos de dados relacionais; classes orientadas a objetos. Algumas vezes, o Modelo do Usuário se sobrepõe ao Modelo do Domínio (*overlay model*).

O Modelo de Adaptação pode ser implementado, por exemplo, usando regras, onde as pré-condições (histórico, perfil, localização) definem o “em função de quê” e as pós-condições determinam “o quê” é alterado / adaptado.

A entrada e atualização dos dados podem ser automáticas ou manuais, pelo usuário ou pelo autor. A informação sobre o usuário pode ser obtida por observação ou o usuário pode preencher um formulário.

4.4 Quando

Algumas características do usuário podem ser determinadas antes da utilização do sistema. A adaptação resultante pode ser vista como personalização (*customization*). A adaptação propriamente dita acontece quando as características influenciam dinamicamente as alterações no sistema. Como visto (Tabela 2), é importante saber em que granularidade de tempo – “em função de que” tempo - esta adaptação ocorre.

4.5 Função de adaptação

Descrevemos a seguir, de uma maneira geral, como as questões de adaptação abordadas se integram para executar a função de adaptação dos AHSs. Este funcionamento é a base do meta-modelo de referência.

O papel do usuário e o contexto podem ser informados ou inferidos e são tipicamente estáticos: em geral, não mudam durante a utilização do sistema (embora este possa não ser o caso para o contexto de usuários acessando pelo celular). Estes aspectos podem ser utilizados para construir um Modelo de Usuário inicial (estereótipo). As preferências e características do usuário também podem ser informadas (através do preenchimento de um formulário, por exemplo) ou inferidas a partir do comportamento do usuário. Para simplificar, considera-se que o perfil é sempre armazenado no UM. Usualmente, não há alteração do perfil durante uma sessão. O histórico de navegação e a utilização da aplicação expressam os interesses do usuário. Estes aspectos observados mudam dinamicamente e são usados para atualizar o UM durante uma sessão. O Modelo de Adaptação utiliza as informações obtidas para executar a adaptação.

5 Meta-modelo de referência para AHSs

O meta-modelo (Figura 1) reflete a proposta do AHAM, segundo a qual uma adaptação efetiva em uma aplicação depende de três fatores [Wu et al 01]:

- a aplicação precisa ser baseada em um Modelo de Domínio que descreva, usando conceitos e relações entre eles, como o conteúdo desta aplicação é estruturado;
- é necessário um Modelo de Usuário que represente, essencialmente, o conhecimento, objetivos e preferências do usuário;
- o sistema deve ser capaz de realizar a adaptação baseado nos Modelos do Domínio e do Usuário, utilizando um Modelo de Adaptação.

Além disto, deve haver uma separação clara entre esses três modelos.

De acordo com os princípios do OOHDM que, como já visto, defende a separação entre os aspectos conceituais, navegacionais e de apresentação, foram acrescentados os seguintes modelos:

- Modelo de Navegação para permitir identificar as diferentes visões navegacionais em relação ao domínio, englobando os nós, seus conteúdos e elos e as estruturas de acesso às páginas;
- Modelo de Apresentação para separar da interface as características mais concretas de implementação.

Outros modelos foram identificados, a saber:

- Modelo de Integração que contempla os sistemas abertos, cujo domínio se estende para além do próprio sistema, integrando domínios externos ao Modelo do Domínio;
- Modelo do Contexto do Usuário, referente às condições de utilização do sistema tais como localização, dispositivo e velocidade de acesso.

A notação de diagrama de classe da UML foi utilizada para a descrição do meta-modelo. A Figura 1² apresenta uma visão geral do mesmo. Cada um dos modelos identificados está encapsulado em um pacote UML (caixa cinza na figura) e é descrito nas próximas seções. A funcionalidade da adaptação, detalhada na seção 5.5, aparece com uma tonalidade diferente para melhorar a legibilidade da figura. A instanciação dos sistemas (seção 6) complementa as descrições.

² A notação UML foi estendida com uma relação entre classe e pacote para representar a existência de relacionamento entre a referida classe com qualquer classe integrante do pacote relacionado.

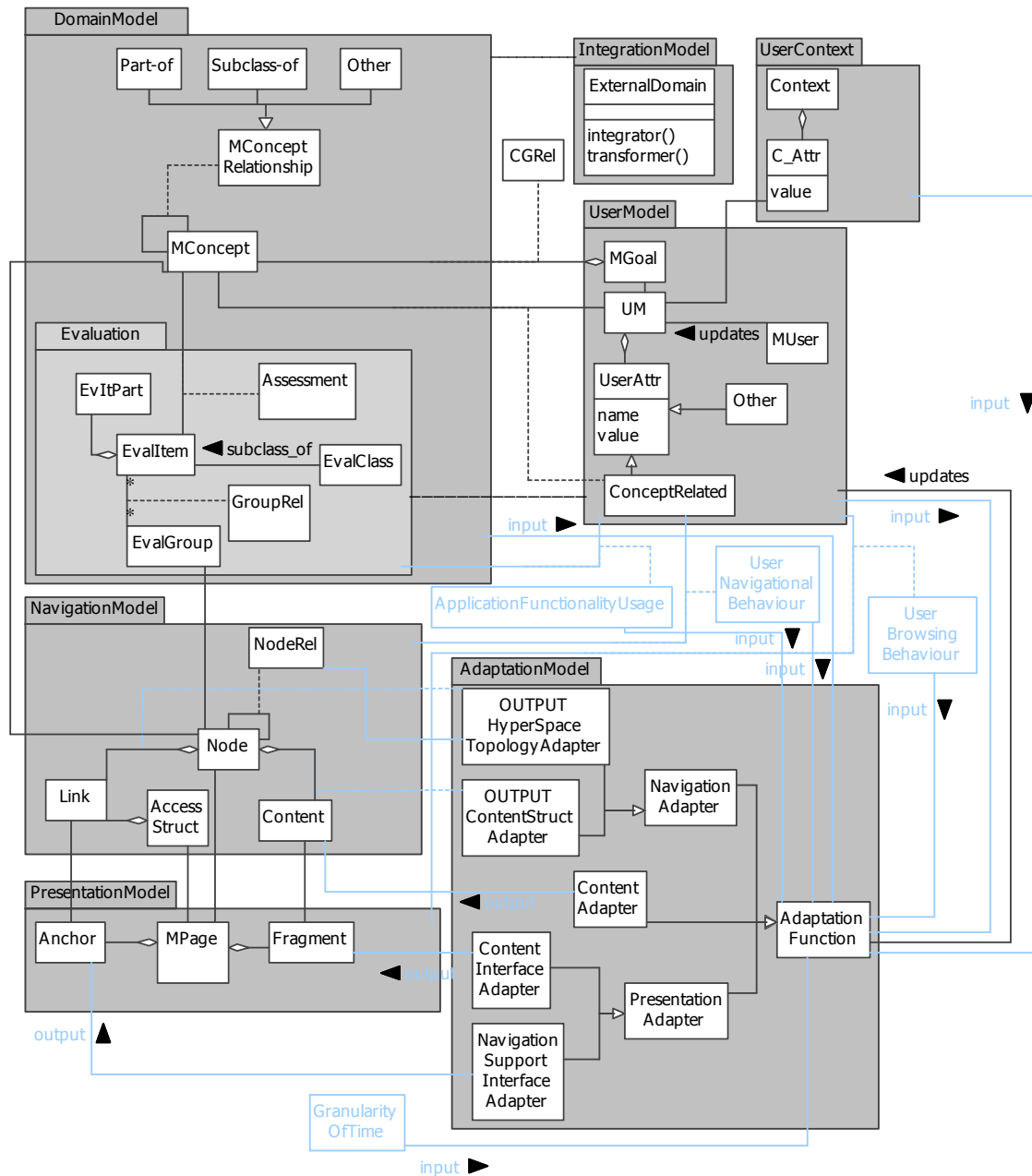


Figura 1. Meta-modelo de referência para AHSs

Vale observar que este meta-modelo representa o núcleo de AHSs - no que diz respeito às funcionalidades básicas referentes à adaptação -, incorporando suas características essenciais. Em outras palavras, pode-se dizer que tais sistemas possuem, ao menos, os componentes principais, embora em muitos casos extensões ao meta-modelo sejam necessárias para acomodar características específicas de alguns sistemas em particular.

5.1 Modelo do Domínio

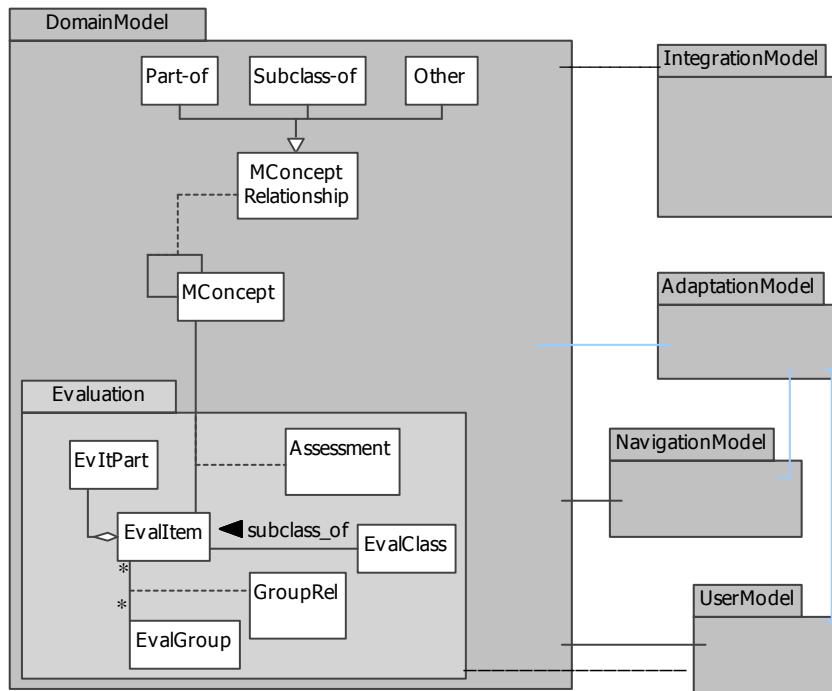


Figura 2. Modelo do Domínio

O Modelo do Domínio consiste, basicamente, de um conjunto de conceitos (*MConcept*³), relacionados entre si (*MConceptRelationship*). As relações são, tipicamente⁴: subparte (*part-of*), subclasse (*subclass-of*) e outras (*other*).

Em geral, existe um esquema de avaliação (pacote *Evaluation*) onde alguns itens (*EvalItem*) são usados para avaliar certos conceitos, de acordo com um determinado peso (*Assessment*). Estes itens podem ser compostos (por *EvItPart*), podem estar agrupados - em *EvalGroup*, com o critério definido em *GroupRel* - e podem ser de diversos tipos (*EvalClass*). Tal esquema pode ser mais bem entendido pela instanciação do NetCoach (seção 6.4).

Uma vez que a avaliação é feita em relação aos conceitos, ela foi incluída no DM. Entretanto, esta é uma questão em aberto, uma vez que a avaliação pode ser considerada de uma maneira mais ampla, como um mecanismo de observação das relações entre o DM e o UM (seção 5.2).

³ a letra *M* é usada para indicar conceitos do meta-modelo, distinguindo-os dos conceitos dos sistemas, conforme será visto nas instanciações. Analogamente para *MConceptRelationship*, *MUser*, *MGoal* e *MPage*.

⁴ Relações de elo entre conceitos, consideradas por alguns autores, fazem parte do nosso Modelo de Navegação, sendo os conceitos mapeados para nós de navegação.

5.2 Modelo / Contexto do Usuário

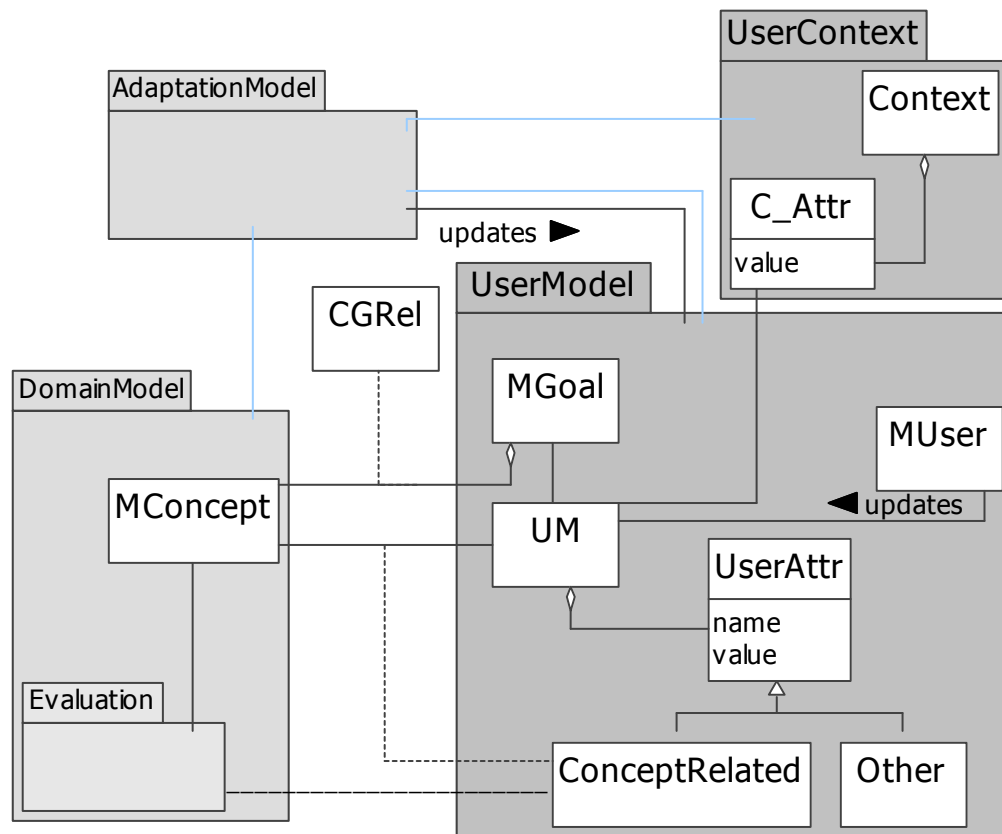


Figura 3. Modelo e Contexto do Usuário

Em especial nos sistemas educacionais, importantes aspectos modelados são: a inclusão, no perfil do usuário, do nível de conhecimento que o usuário tem sobre um conceito e a definição de quais os seus objetivos, ou seja, quais conceitos ele deseja conhecer. O conhecimento pode ser generalizado para uma relação entre o usuário e os conceitos do domínio (*ConceptRelated*) (exemplo: informações que o usuário tem sobre um produto indicando se o perfil dele é de técnico ou de leigo). Da mesma forma, qualquer tipo de aplicação tem um conjunto de objetivos (*MGoal*).

Pode existir uma relação (*CGRel*) entre objetivos e conceitos representando, por exemplo, no caso de objetivos de aprendizagem, em que ordem os conceitos devem ser aprendidos ou se algum conceito é consequência de outro.

Alguns sistemas permitem que o próprio usuário (*MUser*) atualize (*updates*) diretamente o modelo sobre ele armazenado pelo sistema. Um mecanismo de observação das relações entre o DM e o UM também pode gerar uma atualização deste último. Tal mecanismo é representado por avaliações (*Evaluation*), que tanto podem ser o resultado de testes existentes em alguns sistemas, como podem ser generalizadas para outros tipos de uso, como um *feedback* de relevância dado por usuários em outros tipos de aplicações (não-educacionais). O UM também é atualizado como resultado da Função de Adaptação (seção 5.5).

Além dos relacionados aos conceitos, outros atributos do usuário (*Other*) – independentes do domínio – podem ser levados em consideração para a adaptação, tais como preferências do usuário. A situação de utilização do sistema (*Context*) também é

modelada por atributos (*C_Attr*) relacionados ao usuário e independentes do domínio. A diferença é que, neste caso, não se considera a individualidade do usuário. Exemplo: se o acesso é discado ou é banda larga, não importa “quem” está usando, o foco é a velocidade de acesso do usuário, mas não do indivíduo.

5.3 Modelos de Navegação e de Apresentação

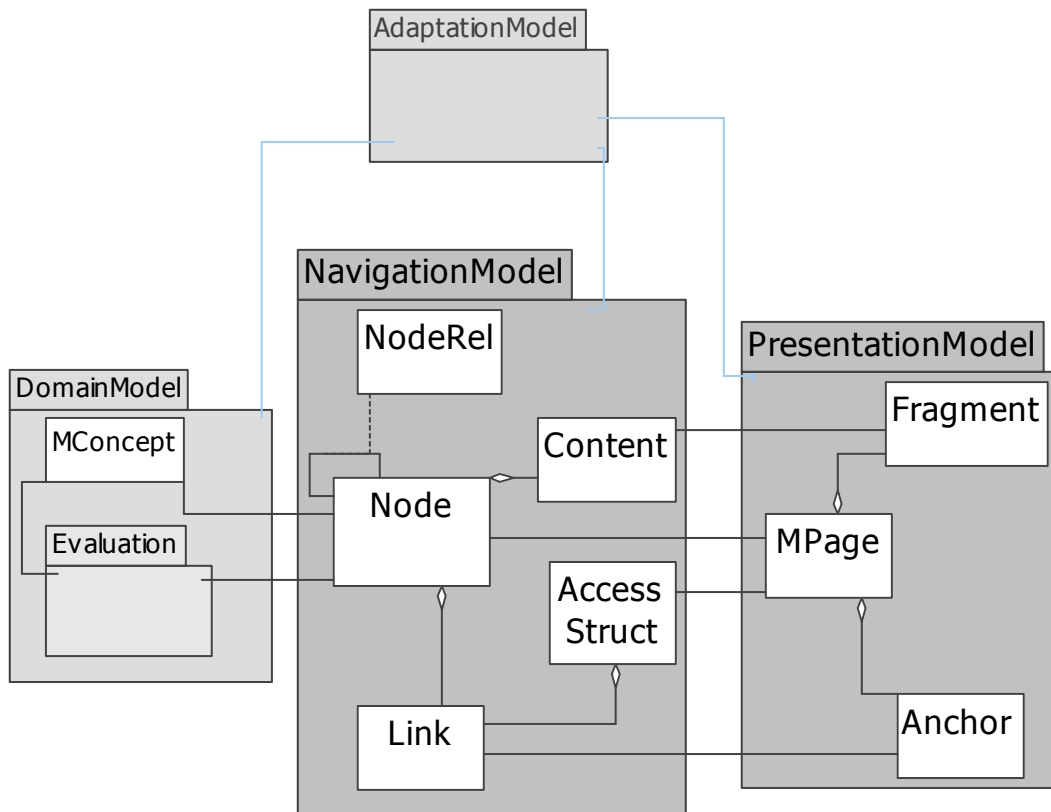


Figura 4. Modelos de Navegação e de Apresentação

Os nós (*Node*) representam visões navegacionais dos conceitos (*MConcept*) e podem estar relacionados entre si (*NodeRel*) através de relações arbitrárias ou padronizadas como no caso de “próximo”, “anterior”, “primeiro”, “último”. Nós correspondem a páginas (*MPage*) do Modelo de Apresentação. Da mesma forma, o conteúdo (*Content*) é mapeado para os fragmentos (*Fragment*), assim como os elos (*Link*) são mapeados para as âncoras (*Anchor*). Estruturas de Acesso (*AccessStruct*) contêm elos (*Link*) e funcionam como índices.

5.4 Modelo de Integração

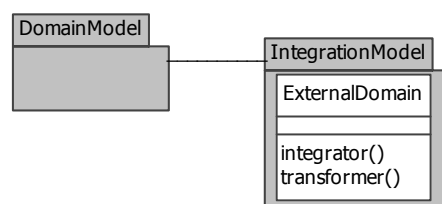


Figura 5. Modelo de Integração

Este modelo, ainda incipiente, utiliza funções de integração responsáveis por buscar dados na *Web* (*integrator*) e convertê-los (*transformer*) para a representação utilizada pelo DM [Vdovjak et al 03].

5.5 Modelo de Adaptação

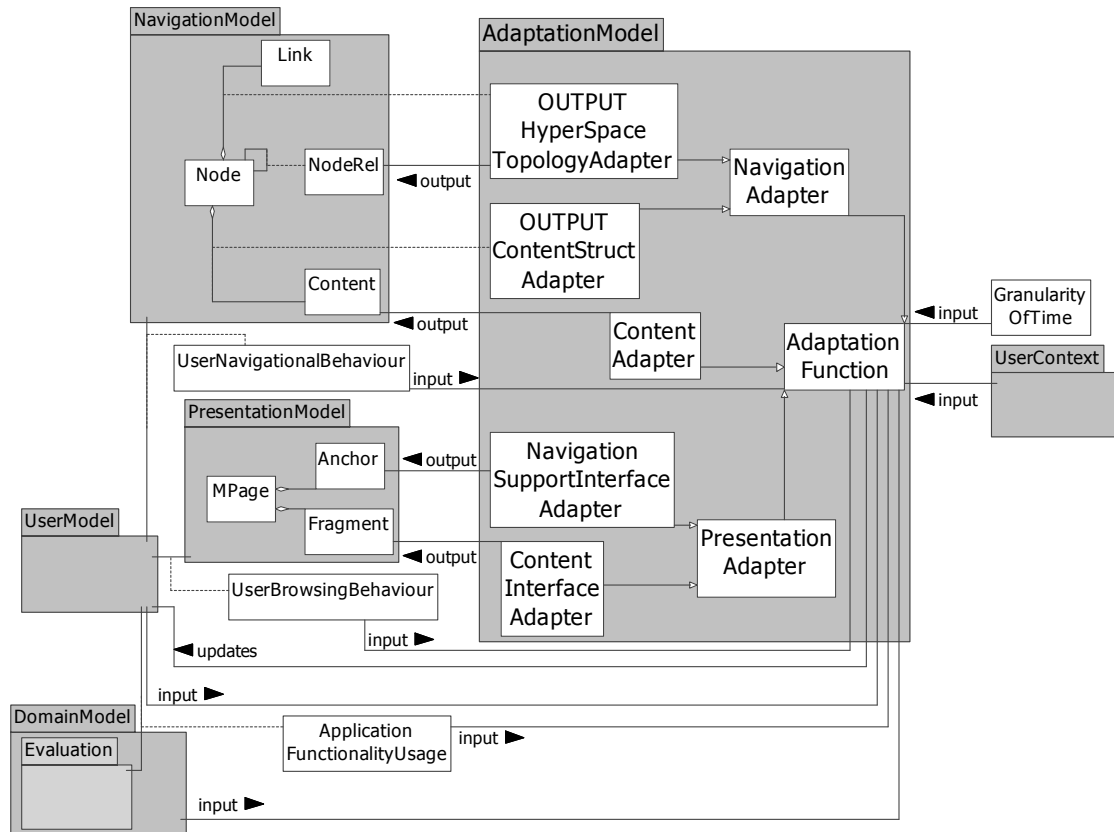


Figura 6. Funcionalidade da aplicação

Este modelo responde às principais questões de adaptação, representando a funcionalidade da aplicação e propiciando uma maneira de expressar os diversos tipos de adaptação. A classe “Função de Adaptação” (*Adaptation Function*) é usada para encapsular o mecanismo de adaptação usado no sistema (por exemplo, regras no caso do AHA!), representando “como” a adaptação é feita (seção 0). Esta função produz como saída (*output*) “o quê” o sistema adapta (seção 4.1). Observa-se, desta forma, que a adaptação da topologia do hiperespaço (*Hyperspace Topology Adapter*) é um tipo de adaptação de navegação (*Navigation Adapter*) que ocorre na relação entre nós (*Node*) e elos (*Link*) - como no caso da inclusão condicional de elos - ou na relação entre nós (*NodeRel*), por exemplo, na orientação direta. A inclusão condicional de fragmentos pode ser representada pela subfunção de adaptação da estrutura do conteúdo (*ContentStructAdapter*) que ocorre na relação entre nós (*Node*) e conteúdo (*Content*). A subfunção de adaptação do conteúdo (*Content Adapter*) atua diretamente no conteúdo dos nós. Já a adaptação da apresentação (*Presentation Adapter*) tem reflexos no PM, mais especificamente nas âncoras (*Anchor*) – adaptação da interface do suporte à navegação (*Navigation Support Interface Adapter*) – e nos fragmentos (*Fragment*) – adaptação da interface do conteúdo (*Content Interface Adapter*).

As entradas (*inputs*) da função representam “em função de quê” a adaptação ocorre (seção 4.2: usuário, estruturação do domínio, granularidade do tempo, etc.). A

relação da granularidade de tempo com outros modelos, tais como domínio e contexto do usuário (exemplo: adaptar quando muda o contexto) é uma questão a ser avaliada, uma vez que não foi observada em nenhum dos sistemas estudados.

6 Instanciação

Para demonstrar a viabilidade da utilização do meta-modelo proposto como referencial para a representação de AHSs, apresentamos a instanciação de alguns dos sistemas mais destacados na área. Na instanciação, os nomes das classes são precedidos pela meta-classe correspondente. Por ser a modelagem de tais sistemas feita sob a ótica do meta-modelo de referência, ela pode ser diferente da descrição original feita pelos respectivos autores sem, entretanto, alterar a essência dos sistemas. Exemplos disso são os Modelos de Navegação e Apresentação que não são definidos explicitamente na descrição de nenhum dos sistemas instanciados, mas são “criados” nas instanciações de acordo com o modelo de referência. O Modelo do Contexto do Usuário não aparece porque nenhum dos sistemas abordados considera a situação de uso para a adaptação. A questão do “quando” não é abordada explicitamente pelos sistemas, não sendo, por isto, modelada.

6.1 AHAM

Como já dito, o AHAM é o modelo por trás de diversos sistemas hipermídia adaptativos e defende a separação entre os Modelos do Domínio (DM), do Usuário (UM) e de Adaptação (AM). Segundo [Wu 02], o DM é formado pelos conceitos atômicos, páginas, conceitos compostos e as relações entre os conceitos. No nível mais baixo, estão os fragmentos (conceitos atômicos). Em um nível intermediário, estão as páginas, que são unidades de informação apresentadas ao usuário a cada interação. Estas compõem os conceitos abstratos que, por sua vez, podem compor conceitos abstratos de mais alto nível. Páginas são compostas por fragmentos, possibilitando a representação abstrata de diversas técnicas de adaptação, tais como, inclusão / remoção de fragmentos; alteração de fragmentos; *stretch text* e ordenação de fragmentos. Fragmentos podem ser condicionalmente incluídos, mas não podem ser alterados pelos AHSs. Nossa proposta é que a instanciação, apresentada na Figura 7, fique de acordo com o meta-modelo de referência que propõe a separação das visões conceitual, navegacional e de apresentação. Assim, os conceitos (compostos) - que representam itens de informação abstratos [Wu et al 01] - e as relações entre eles são instanciados no DM (seção 6.1.1). Já relações de elo, conceitos atômicos e páginas correspondem a classes dos Modelos de Navegação e de Apresentação (seção 6.1.3).

Interações com o sistema são descritas através de eventos, modelados como atualizações nos atributos que disparam regras [Wu & De Kort 02] (seção 6.1.5).

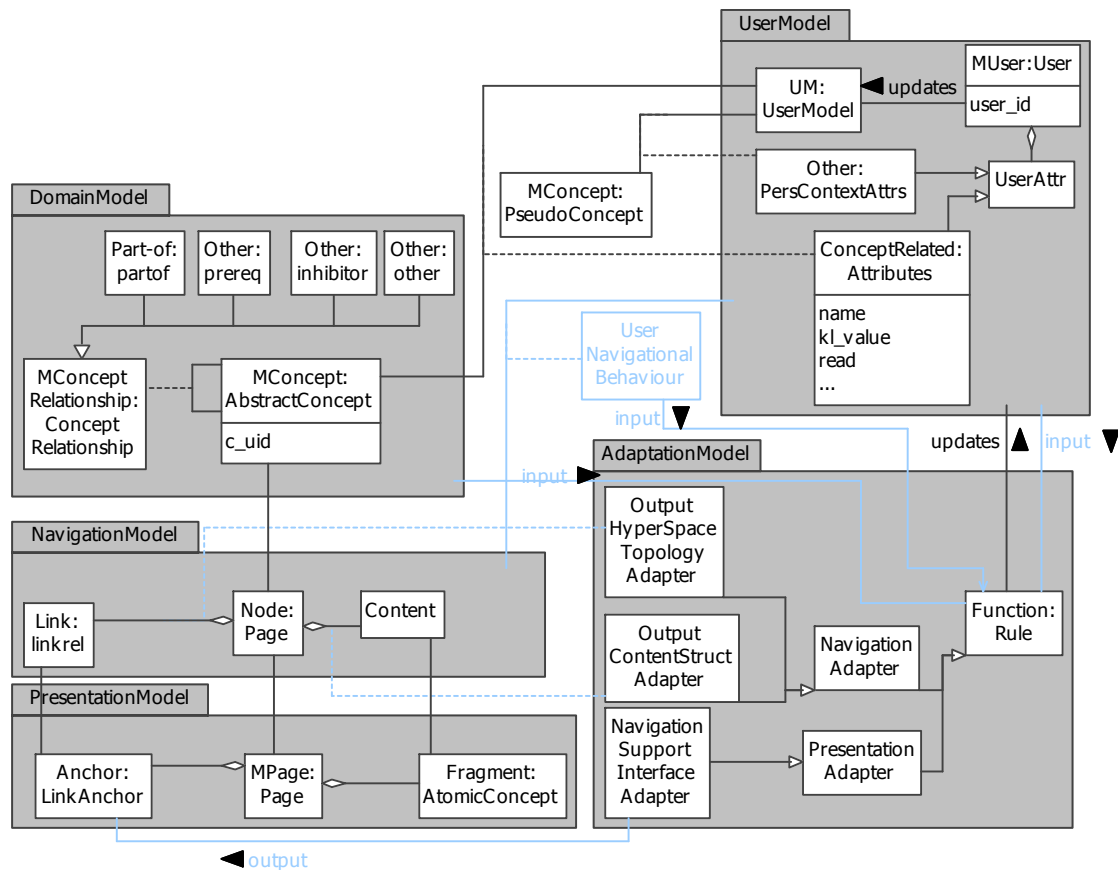


Figura 7. Instanciação do meta-modelo do AHAM

6.1.1 Modelo do Domínio

O domínio é composto por conceitos abstratos [Wu et al 01] e as relações entre eles. A relação “subparte” (*part-of*) pode ser usada para representar a hierarquia entre os conceitos. Outros tipos de relações (abstratas) relevantes para a adaptação - como, por exemplo, pré-requisito (*prereq*) e inibidor (*inhibitor*) - são considerados. Teoricamente, relações podem ser definidas arbitrariamente (*Other: other*).

6.1.2 Modelo do Usuário

O Modelo do Usuário é uma superposição (*overlay*) do DM: para cada conceito são armazenados valores de atributo sobre o conceito. Estes atributos (*Attributes*) representam as relações entre o usuário e os conceitos do DM como, por exemplo, conhecimento do usuário sobre o conceito (*kl_value*) e interesse do usuário pelo conceito. Outro atributo usual é “lido” (*read*), indicando se o usuário leu alguma coisa sobre o conceito. Através destes atributos, o UM rastreia o quanto o usuário sabe sobre os conceitos do domínio. Além destes, o UM pode armazenar conceitos independentes do domínio como, por exemplo, características pessoais e de contexto (*PersContextAttrs*), tais como: “experiência” (*background*); propriedades do dispositivo de acesso; idade. Embora semanticamente bem diferente, este tipo de informação pode ser armazenado e implementado da mesma forma que os conceitos do domínio, pela utilização de “pseudoconceitos” (*PseudoConcept*).

O AHAM assume que o UM é atualizado em função de eventos. Assim, testes, importação de informações sobre o usuário (de sistemas externos) e indicação explícita

das preferências do usuário (através de formulários) podem ser eventos utilizados para atualizar o UM [Wu 02], sendo modelados pelo esquema geral de adaptação (seção 6.1.5). A possibilidade de o usuário alterar seu próprio modelo é mostrada pela relação “updates” entre *User* e *User Model*.

6.1.3 Modelos de Navegação e de Apresentação

O AHAM não separa a navegação e a apresentação. Seguindo o meta-modelo de referência, conceitos atômicos, páginas e âncoras dos elos (*link anchor*) são instâncias, respectivamente, das classes *Fragment*, *MPage* e *Anchor* do Modelo de Apresentação. Por outro lado, de acordo com [Wu 02], conteúdo é um atributo dos fragmentos e a informação é acessada como páginas. Assim, no Modelo de Navegação, página é também instância de *Node* e a relação de elo entre conceitos é mapeada para a classe *Link*.

6.1.4 Modelo de Integração

Uma vez que eventos são utilizados para descrever interações com o sistema, pode-se considerar que a integração com domínios externos poderia ser implementada como um evento, ou seja, como atualizações em atributos que disparam regras.

6.1.5 Modelo de Adaptação

Para descrever o comportamento de um AHS, o AHAM exemplifica a Função de Adaptação como sendo implementada por regras, executadas por um mecanismo de adaptação (*Adaptation Engine*, AE), responsável pelos aspectos dependentes da implementação. A descrição do AE foge ao escopo deste trabalho. De uma maneira geral, atualizações nos atributos disparam as (condições das) regras. As (ações das) regras podem atualizar atributos. Assim, regras podem disparar outras regras. A partir de informações do DM, do UM e da navegação do usuário, as regras geram a especificação da apresentação e atualizam o UM. Tipos de adaptação exemplificados são: a escolha de quais fragmentos e quais elos serão incluídos na página e em que ordem (estrutura do conteúdo e topologia do hiperespaço) e especificações das âncoras (interface de suporte à navegação). O autor pode definir se e quais conceitos do UM podem ser atualizados pelo próprio usuário.

6.2 AHA!

AHA! (*Adaptive Hypermedia Architecture*) é uma ferramenta de propósito geral, desenvolvida para acrescentar adaptação a um *website*. A Figura 8 apresenta a instanciação do seu meta-modelo, baseado na versão 2.0. As maiores diferenças em relação à versão 1.0 estão na utilização de um UM mais bem estruturado e de regras de requisito e geração mais avançadas. Além disto, aspectos independentes do domínio – informados quando o usuário se registra – podem ser utilizados para a adaptação, uma vez que são representados por um tipo especial de conceito (*Personal Concept*) e seus atributos.

De acordo com [De Bra et al 02b], o AHA! se desvia do AHAM por não separar o DM do AM. Os conceitos são definidos junto com (regras de) requisitos (*requirement rules*) - que determinam sob quais condições o usuário está “pronto” para acessar o conceito – e com regras de geração (*generate rules*) que especificam como o comportamento de navegação do usuário é utilizado para atualizar o UM. As regras são do tipo condição-ação e são disparadas por atualizações nos atributos dos conceitos. O

acesso às páginas também dispara regras, por ser considerado como uma atualização de um (pseudo) atributo - “acesso” - associado às páginas. Conforme será visto a seguir, a separação entre o DM e o AM foi mantida na instanciação e as relações entre conceitos e regras e entre páginas e atributos ficam implícitas, podendo ser deduzidas a partir dos relacionamentos existentes no modelo.

Além do acesso às páginas, outros eventos podem disparar as regras. Considera-se que estes eventos são implementados através de módulos adicionados ao AHA!, tornando o mesmo extensível. A versão atual do AHA! já inclui um módulo que implementa testes de múltipla escolha (seção 6.2.1) e outro para permitir a atualização do UM baseada em formulários (seção 6.2.2).

O AHA! usa cores diferentes para as âncoras, de acordo com o tipo de elo. Por *default*, o AHA! utiliza também a técnica de ocultação de elos (usando a cor preta para elos “ruins”). Entretanto, como o esquema de cores fica armazenado no UM e pode ser alterado pelo usuário⁵, o AHA! pode ser configurado para usar apenas a técnica mais geral de anotação de elos (com a utilização de cores diferentes e visíveis para todos os tipos de elo).

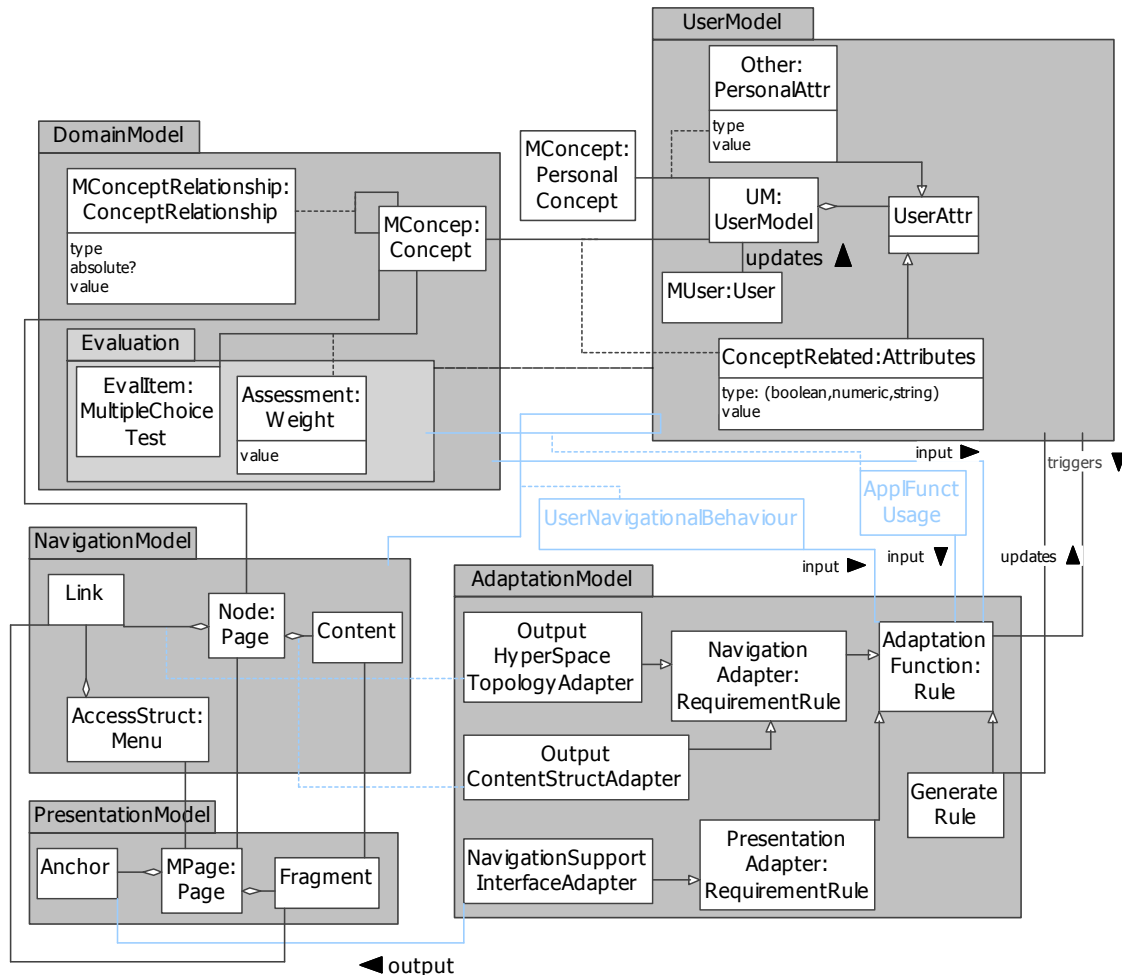


Figura 8. Instanciação do meta-modelo do AHA!

⁵ A representação do esquema de cores precisa ser modelada como uma extensão ao meta-modelo de referência.

6.2.1 Modelo do Domínio

A relação (*ConceptRelationship*) entre os conceitos (*Concept*) pode ser de diversos tipos (*type*) e pode ter um valor (*value*) associado. Este valor pode ser um percentual ou um valor absoluto e é utilizado para atualizar o valor do conceito relacionado no UM (seção 6.2.2). Na maioria das aplicações, o DM é uma hierarquia de conceitos: alguns conceitos representam páginas enquanto outros correspondem a conceitos abstratos de mais alto nível. As relações entre os conceitos dão uma idéia da ordem em que os mesmos devem ser apresentados.

O autor pode elaborar testes de múltipla escolha (*Multiple Choice Test*), cujos resultados são utilizados para atualizar os valores dos conceitos associados, de acordo com o peso (*Weight*) atribuído.

O AHA! representa os atributos independentes do domínio da mesma forma que os dependentes. Assim, a representação foi feita de maneira análoga – tratando-os como conceitos, mas fora do DM (instanciação *MConcept: PersonalConcept*)

6.2.2 Modelo do Usuário

Os conceitos formam a base do UM. Para cada conceito, o autor define um conjunto de atributos que relacionam o usuário ao conceito. Estes podem ou não ser dependentes do domínio (*ConceptRelated:Attributes* ou *Other:PersonalAttr*).

Um formulário de registro simples é utilizado para identificar cada usuário individualmente, embora o *login* anônimo também seja permitido. O UM é inicializado a partir das informações obtidas e de valores *default* definidos pelo autor e persiste a cada sessão. No caso de usuários anônimos, o UM refletirá o uso feito pelo último acesso no mesmo navegador e computador. O autor também pode utilizar formulários para permitir que o usuário atualize diretamente o valor de (alguns) conceitos do UM (relação “*User updates User Model*”). A implementação dos formulários não é modelada.

6.2.3 Modelos de Navegação e de Apresentação

Uma vez que o AHA! não define um Modelo de Apresentação nem um de Navegação, estes foram representados de acordo com o meta-modelo de referência, sendo que a página (*Page*) do AHA! foi instanciada para ambos, de modo a possibilitar a representação dos tipos de adaptação que ocorrem. No AHA!, o próprio autor pode implementar estruturas de acesso (*Menu*). Uma vez que o AHA! considera elos (*Link*) como fragmentos (*Fragment*) (ver seção 6.2.5), uma relação entre eles precisou ser criada.

6.2.4 Modelo de Integração

O AHA! não faz uma referência explícita a um Domínio Externo, talvez porque o acesso a URLs externas seja permitido, mas sem adaptação. Por outro lado, um módulo para implementar a integração de domínios poderia ser adicionado.

6.2.5 Modelo de Adaptação

A Função de Adaptação (*Adaptation Function*) é implementada por regras, que podem ser de geração (*Generate Rule*, atualizam o UM) ou de requisito (*Requirement Rule*, adaptação da navegação e da apresentação).

Para o AHA!, as regras de requisitos provêm dois tipos de adaptação: adaptação da apresentação, feita pela inclusão condicional de fragmentos, e adaptação da navegação, pela anotação dos elos. Entretanto, o meta-modelo de referência classifica a inclusão condicional de fragmentos como uma adaptação da estrutura do conteúdo (*ContentStruct*) e a anotação (das âncoras) dos elos como adaptação da apresentação ou, mais especificamente, adaptação da interface de suporte à navegação (*Navigation Support Interface*). Além disso, como elos (*links*) são considerados fragmentos pelo AHA!, o mecanismo de inclusão condicional de fragmentos possibilita, também, a inclusão ou não de elos, implementando a adaptação da topologia do hiperespaço (*Hyperspace Topology*).

As regras implementam a adaptação e atualizam o UM a partir de informações sobre o histórico de navegação do usuário (acessos às páginas), o perfil do usuário (deduzido do histórico de navegação), resultado dos testes (*Application Functionality Usage*) e DM.

6.3 InterBook

O InterBook, cujo meta-modelo aparece instanciado na Figura 9, é um sistema de autoria para o desenvolvimento de livros-texto eletrônicos (*Electronic Textbooks, ETs*).

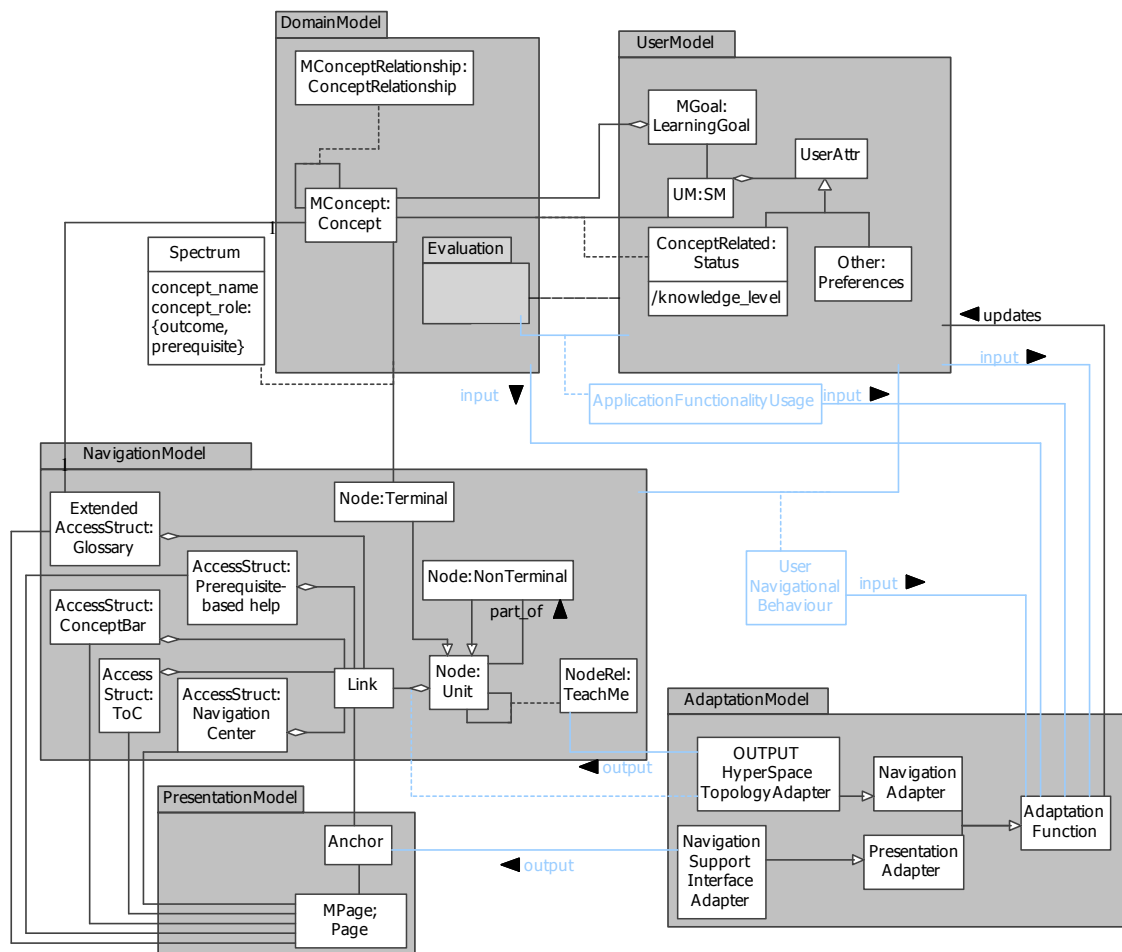


Figura 9. Instanciamento do meta-modelo do Interbook

6.3.1 Modelo do Domínio

O Modelo do Domínio é formado pelos conceitos e relações entre eles. Problemas e *quizzes* podem ser usados para atualizar o conhecimento do usuário [Brusilovsky et al 98]. Tal possibilidade é representada pelo pacote *Evaluation*.

6.3.2 Modelo do Usuário

O Modelo do Usuário é representado pelo Modelo do Estudante (*Student Model*, SM), usado para armazenar a crença (calculada) do sistema sobre o nível de conhecimento do estudante (*knowledge-level*) sobre cada conceito (*Status*). O SM é inicializado através de estereótipos e informações obtidas pela página de registro. Todas as ações do estudante (acesso às páginas, solução de problemas ou *quizzes*) são rastreadas e utilizadas para aumentar ou diminuir os níveis de conhecimento sobre os conceitos envolvidos [Brusilovsky et al 98]. Os objetivos de aprendizagem (*Learning Goals*), ou seja, o conjunto de conceitos a serem aprendidos também fazem parte do SM. As visitas às páginas são armazenadas no UM, de uma forma independente dos conceitos. Consideramos que estas visitas estão representadas pela classe *User Navigational Behaviour*. Atributos independentes do domínio (*Other*) podem ser usados para armazenar preferências (*Preferences*) relativas a características da interface (se deseja marcar páginas já visitadas, se deseja que o texto contenha elos para outras páginas, etc.).

6.3.3 Modelos de Navegação e de Apresentação

No Interbook, a estruturação do conteúdo ocorre no DM, que representa o conhecimento sobre o domínio sendo ensinado através de uma rede, cujos nós correspondem aos conceitos do domínio e cujos elos refletem os vários tipos de relacionamento entre estes conceitos. De acordo com o meta-modelo de referência, nós e elos são representados no Modelo de Navegação.

Um ET pode ser definido como qualquer material hipermídia, hierarquicamente estruturado em unidades (*Units*) de diferentes níveis (capítulos, seções e subseções). Cada unidade no nível final da hierarquia (*Terminal*) pode ser uma unidade de apresentação (*atomic presentation*), um exemplo, um problema ou um teste. Esta classificação não é modelada. Uma barra de navegação (*Navigation Center*) permite o acesso às unidades do mesmo nível ou de níveis superiores. Um sumário (*Table of Contents, ToC*), onde todas as entradas são elos, também pode ser gerado.

Conceitos são usados como índices para o ET. Para implementar isto, uma lista de conceitos (*Spectrum*) (extensão ao meta-modelo de referência) é associada a cada unidade terminal contendo, para cada conceito envolvido, o nome do conceito (*concept_name*) e o papel (*concept_role*) que ele representa para a unidade: pré-requisito (*prerequisite*) ou consequência (*outcome*, ou seja, a unidade introduz o conceito). Isto possibilita ao sistema saber quais conceitos estão presentes em cada página e, conseqüentemente, quais conceitos precisam ser aprendidos antes que a página possa ser acessada. Uma Barra de Conceitos (*Concept Bar*) permite o acesso aos conceitos organizados de acordo com seus papéis.

O glossário é considerado como uma estrutura de acesso por constituir uma visualização da rede do domínio e exercer a função de indexador: o glossário não só descreve o conceito (quando tal descrição é fornecida pelo autor), como também indexa todas as unidades que o introduzem ou que dele necessitam. Para tanto, o glossário está

diretamente associado aos conceitos⁶ (cada entrada do glossário corresponde a um dos conceitos do domínio e cada conceito é representado por uma entrada do glossário).

Outra estrutura de acesso presente no InterBook é a ajuda baseada em pré-requisitos que consiste na geração de uma lista ordenada de elos para todas as seções que apresentam informações sobre conceitos relevantes para o entendimento da seção atual (*background concepts*).

Uma vez que a página no Interbook não é composta por fragmentos, estes, bem como o conteúdo dos nós, não foram modelados.

Apesar do Interbook trabalhar com duas janelas principais: uma para o conteúdo (*Textbook*) e outra para o glossário, consideramos isto como uma questão de implementação. Portanto, em nossa modelagem apenas página (*Page*) e as âncoras (*Anchor*) nela incluídas estão representadas.

6.3.4 Modelo de Integração

O Interbook não prevê integração com domínios externos, embora possua o conceito de conjunto (*bookshelf*) constituído por vários ETs sobre um mesmo assunto.

6.3.5 Modelo de Adaptação

A Função de Adaptação: (i) provê orientação direta (*direct guidance*), através do botão *TeachMe*; (ii) calcula a lista ordenada de elos apresentada pela ajuda baseada em pré-requisitos, sendo esta ajuda requisitada através de um botão (*button*) especial e (iii) determina a anotação dos elos, usando ícones, fontes e cores diferentes de acordo com o nível de aprendizado de cada unidade. O meta-modelo de referência considera que os dois primeiros tipos são adaptações da topologia do hiperespaço (*Hyperspace Topology*) e estão modelados, respectivamente, como alterações na relação entre os nós (*NodeRel: TeachMe*) e na relação entre nós (*Node*) e elos (*Link*). Como já visto, a anotação (das âncoras) dos elos é considerada adaptação da apresentação, mais especificamente, da interface de suporte à navegação (*Navigation Support Interface*).

Para realizar a adaptação, o InterBook leva em consideração o perfil do estudante - conhecimento sobre o domínio e objetivos – e as ações do estudante, ou seja, o histórico da navegação (páginas acessadas) e a utilização da funcionalidade da aplicação (solução de problemas e *quizzes*). O Modelo de Adaptação caracteriza-se pela indexação do Modelo de Domínio.

6.4 NetCoach

O NetCoach (anteriormente denominado ART-Web) é um sistema de autoria para cursos de ensino à distância baseados na *Web*. Ele foi desenvolvido após os resultados positivos obtidos com o sistema ELM-ART para ensino da linguagem de programação LISP. A instanciação do meta-modelo do NetCoach pode ser vista na Figura 10.

⁶ Esta associação entre glossário e conceito é uma extensão ao meta-modelo de referência.

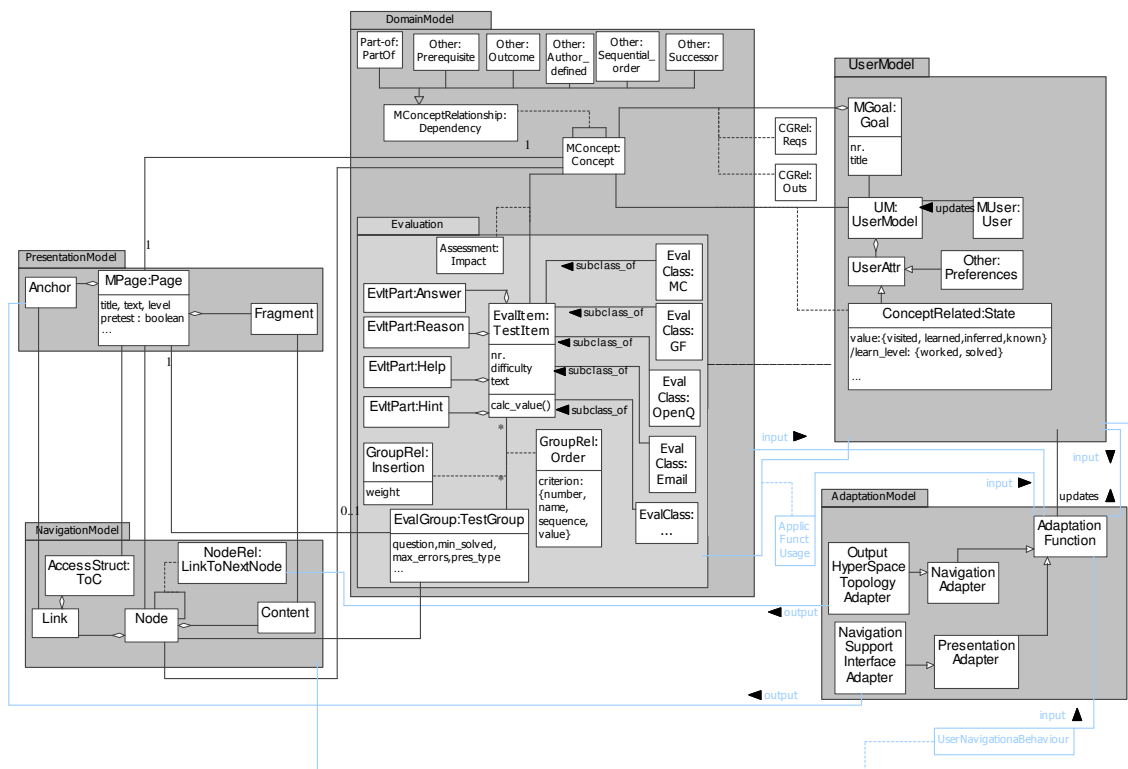


Figura 10. Instânciação do meta-modelo do NetCoach

6.4.1 Modelo do Domínio

O domínio, considerado como sendo uma Base de Conhecimento (*Knowledge Base*), é formado pelos conceitos e pelas relações entre eles. Relações podem ser de pré-requisito (*Prerequisite*) ou inferência (*Outcome*) e devem ser cuidadosamente definidas de acordo com a estrutura do domínio, de modo a propiciar uma adaptação eficiente. Relações de ordenação (*Sequential order*), sucessão (*Successor*) e parte (*Part-of*) são deduzidas a partir da estrutura hierárquica das páginas. Teoricamente, há a possibilidade da definição de relações por parte dos autores (*Author defined*).

No NetCoach, os testes desempenham um papel fundamental na avaliação do conhecimento do usuário e conseqüente adaptação do sistema. O esquema de avaliação (*Evaluation*) no NetCoach consiste de Grupos de Testes (*Test Group*, TG) associados a páginas (do PM)⁷. Um TG pertence a exatamente uma página e é composto por Itens de Teste (*Test Items*, TIs). Estes podem pertencer a mais de um grupo de teste, sendo ordenados (*Order*) por número, nome ou grau de dificuldade. Um mesmo TI pode ter peso (*weight*) diferente, de acordo com o grupo a que pertence (*GroupRel: Insertion*). Existem diferentes tipos de TIs: múltipla escolha (*multiple choice*, MC), lacuna (*gap filling*, GF), discursivos (*open question*, OP), e-mail e outros. Os TIs podem estar associados a respostas (*Answer*), explicações (*Reason*), textos de ajuda (*Help*) e dicas (*Hint*).

Os resultados obtidos pelo usuário nos testes são utilizados para atualizar o UM e influenciam não só os conceitos aos quais os testes estão diretamente associados, como também, embora em menor grau, os conceitos relacionados a estes. A influência

⁷ Esta relação é implícita no meta-modelo de referência.

dos testes nos conceitos é representada pela classe de relacionamento *Assessment:Impact*.

6.4.2 Modelo do Usuário

Além do nível de conhecimento (*learn level*) calculado (se os testes associados foram realizados ou resolvidos), o UM também armazena, para cada usuário, informações sobre o estado (*State*) das páginas, isto é, quais páginas foram visitadas, estudadas, inferidas ou quais páginas são conhecidas (respectivamente, *visited*, *learned*, *inferred*, *known*). Estas últimas podem ser definidas pelo próprio usuário. A forma de implementação do UM por camadas (*layer*) possibilita que informações de diferentes fontes não se sobrescrevam (exemplo: informações obtidas com os testes permanecem, mesmo que o usuário altere seu próprio modelo) [Weber et al 01].

Os estudantes também podem selecionar objetivos (*Goals*) no NetCoach, escolhendo quais conceitos querem aprender. Neste tipo de aprendizado, alguns conceitos serão indicados como necessários (*Reqs*) para que o estudante possa atingir o objetivo, enquanto outros serão marcados como conhecidos (*Outs*), à medida que o estudante for avançando no seu aprendizado.

Alguns sistemas desenvolvidos com o NetCoach permitem que os usuários indiquem suas preferências (*Preferences*). Exemplos: uso ou não de *frames* no *browser*, opção por *feedback* do sistema para respostas corretas).

6.4.3 Modelos de Navegação e de Apresentação

De acordo com [Weber et al 01], os conceitos são representações internas das páginas que serão apresentadas ao estudante e estão relacionados entre si. Assim, uma vez que o NetCoach não faz distinção entre aspectos conceituais (*MConcept*) e de implementação (*MPage*), é necessária a criação (inexistente no meta-modelo de referência) de uma relação 1:1 entre *MConcept* e *MPage*. Esse isomorfismo possibilita que a estrutura hierárquica das páginas seja modelada pelas relações entre os conceitos no DM.

O sumário (*Table of Contents*, ToC), por permitir o acesso a qualquer página, a qualquer momento, é considerado como Estrutura de Acesso (*AccessStruct*).

6.4.4 Modelo de Integração

O NetCoach não provê integração com domínios externos. Autores podem importar o conteúdo de um arquivo HTML existente.

6.4.5 Modelo de Adaptação

Nos sistemas desenvolvidos com o NetCoach, os elos são anotados segundo a metáfora do “sinal de trânsito” (adaptação da apresentação). A adaptação da topologia do hiperespaço, refletida na relação entre nós (*LinkToNextNode*), ocorre na visita guiada: o usuário pode solicitar ao sistema uma orientação sobre o melhor caminho de navegação. O sistema, então, usa informações sobre o estado do UM e sobre o objetivo geral de aprendizagem para calcular dinamicamente a melhor página a ser visitada em seguida.

No NetCoach, o conhecimento, preferências e objetivos do usuário; os resultados dos testes (*Application Functionality Usage*) e o caminho de navegação escolhido (histórico de navegação) são alguns aspectos levados em consideração para a adaptação. A partir dos parâmetros definidos no TG e das informações do UM, o Modelo de Adaptação executa os cálculos: da avaliação do conhecimento do usuário; da escolha

dos testes a serem apresentados e da indicação do próximo passo a ser seguido. A estrutura do domínio, refletida na Base de Conhecimento, também determina o modo como a adaptação ocorre.

6.4.6 Extensões

Além de prover um bloco de notas, onde os usuários podem fazer anotações sobre o que estão aprendendo, o NetCoach armazena estatísticas sobre a utilização do sistema que permitem aos tutores observar como os usuários estão navegando e resolvendo os testes. Página com referências, glossário - acessíveis através de elos no texto ou de *buttons* – e ferramenta de busca também podem ser implementados. O NetCoach disponibiliza, ainda, ferramentas de comunicação – síncronas e assíncronas – que permitem a interação entre os estudantes, sem atrapalhar o ritmo de aprendizagem de cada um.

6.5 KBS

O KBS *Hyperbook System* (KBS) é uma ferramenta voltada para sistemas hipermídia adaptativos abertos. Sua instanciação é mostrada na Figura 11.

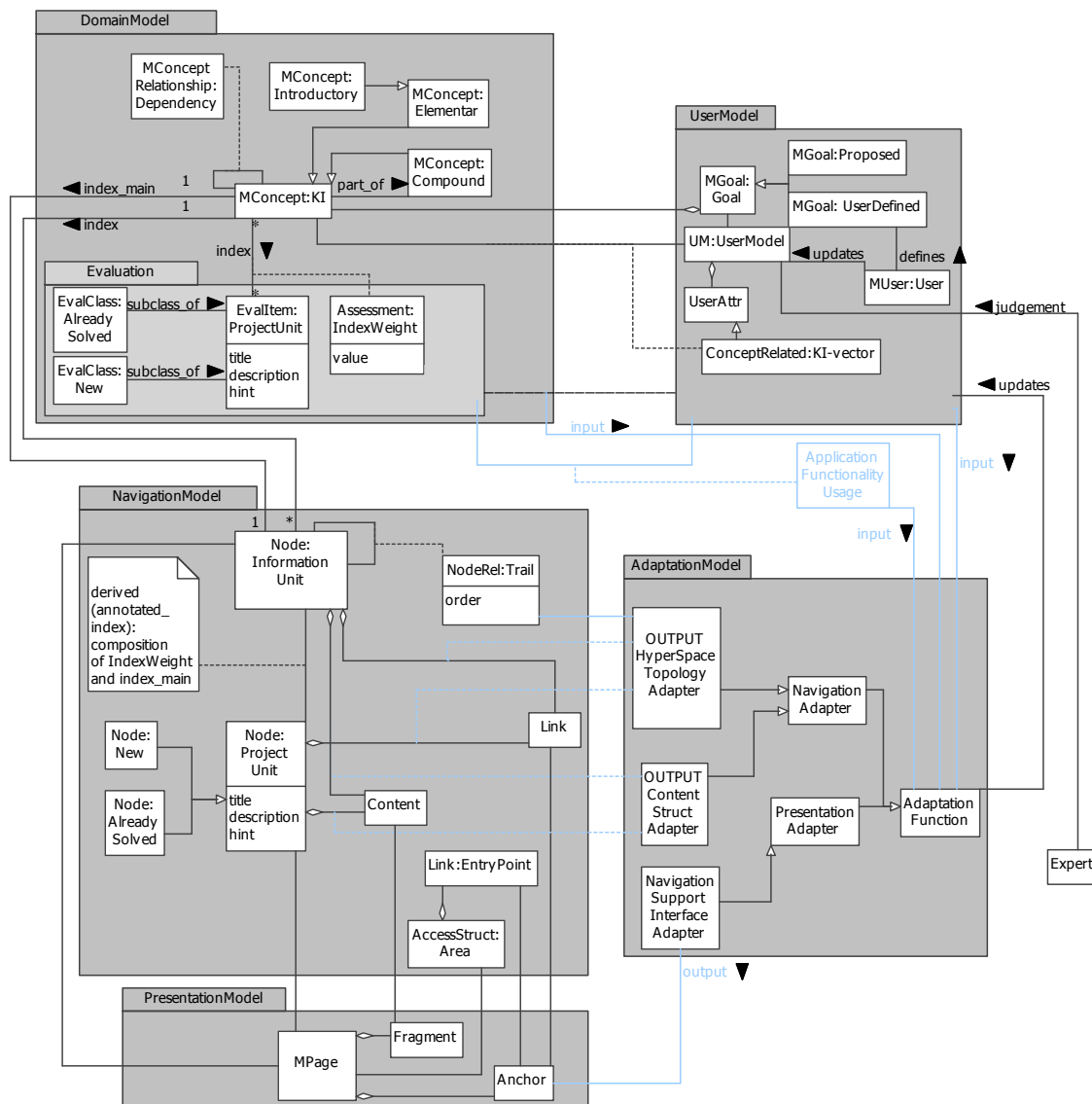


Figura 11. Instanciação do meta-modelo do KBS

6.5.1 Modelo do Domínio

Os conceitos do domínio são representados por Itens de Conhecimentos (*Knowledge Items*, KIs), utilizados como índices. Esta indexação pode ser melhor entendida na descrição do Modelo de Adaptação (seção 6.5.5). KIs podem ser elementares (*Elementar*) ou compostos (*Compound*). Conceitos introdutórios (*Introductory*, subtipo de conceitos elementares) são utilizados na implementação. O KBS considera o DM como um Modelo de Conhecimento (*Knowledge Model*, KM) que agrupa os KIs e as relações de dependência (*Dependency*) entre eles.

Unidades de Projeto (*Project Units*, PUs) formam a base do KBS, sendo representadas no DM por *EvalItem*. PUs podem ser “novas” (*New*) ou “já resolvidas” (*Already Solved*, funcionando como exemplos). Os KIs também são usados como índices ponderados (*IndexWeight*) para as PUs. Tais valores indicam a relevância dos KIs para a execução dos projetos.

6.5.2 Modelo do Usuário

O UM armazena a estimativa sobre o conhecimento do usuário em relação a cada conceito do KM sob a forma de um vetor de KIs (*KI-vector*), implementado por redes Bayesianas [Henze & Nejd, 99b]. A atualização do UM, baseada no desempenho do estudante em algum projeto, é feita a partir de um *feedback* dado pelo próprio usuário (representado pela relação *updates*) ou de uma avaliação feita por especialistas (*expert*, classe que estende o meta-modelo de referência).

A aprendizagem por objetivos também é contemplada pelo KBS, sendo que o objetivo (*Goal*) – formado pelos KIs a serem aprendidos - pode ser escolhido pelo usuário (*User Defined*) ou proposto pelo sistema (*Proposed*).

6.5.3 Modelos de Navegação e de Apresentação

Unidades de Informação (*Information Units*, IUs) são consideradas nós de navegação. Elas podem estar relacionadas entre si formando uma trilha (*Trail*, seção 6.5.5). O *Hyperbook* pode ser dividido em áreas (*AccessStruct:Area*) que agrupem as IUs, estabelecendo pontos de entrada (*EntryPoints*) que podem ser seguidos pelos estudantes.

Como há uma mistura entre o conceitual e o navegacional, PUs – além de itens de avaliação (seção 6.5.1) - são consideradas também como nós.

As IUs relevantes para os projetos podem ser deduzidas (*derived*), uma vez que são indexadas pelos KIs (vide seção 6.5.5).

O KBS não especifica a apresentação que é, portanto, modelada segundo o meta-modelo de referência.

6.5.4 Modelo de Integração

A abordagem de indexação (seção 6.5.5) utilizada pelo KBS permite que as IUs sejam tratadas de maneira equivalente, independentemente de suas origens, possibilitando a integração de páginas HTML existentes na *Web* como se elas fossem documentos da própria biblioteca do *Hyperbook*.

6.5.5 Modelo de Adaptação

O KBS é baseado em problemas / projetos e suas relações com as IUs. A conexão entre o *Hyperbook* e o UM é feita pela indexação de qualquer tipo de recurso de informação

(páginas HTML do *Hyperbook*, projetos, exemplos, páginas da *Web*). Assim, KIs indexam IUs - sendo que um único KI é o índice principal (*index_main*) de uma determinada IU - e KIs associados a valores indexam PUs. Desta forma, as IUs relevantes para os projetos podem ser deduzidas (*derived*),

As estratégias de adaptação implementadas pelo KBS são baseadas nos KIs. De acordo com o objetivo e o conhecimento do usuário, o sistema pode:

- (i) sugerir um projeto para o estudante;
- (ii) apresentar as IUs relevantes para a tarefa sendo executada (objetivo ou projeto). Estas IUs são apresentadas como um índice ordenado, onde cada elo é anotado (adaptado), usando a metáfora do sinal de trânsito, de acordo com o conhecimento do usuário;
- (iii) propor uma trilha (*trail*: seqüência ordenada de IUs) a ser seguida. As trilhas podem ser propostas para ajudar o estudante a atingir seu objetivo, resolver um projeto ou completar o curso baseado no KBS.

Baseado no conhecimento do estudante, o sistema pode propor objetivos e, a partir destes, sugerir projetos, IUs ou trilhas.

A geração de trilhas, por ser um tipo de orientação direta, é considerada adaptação da topologia do hiperespaço. A escolha dos projetos a serem executados pelo estudante e a determinação de objetivos de aprendizagem podem ser vistas como adaptação da navegação; tanto da estrutura do conteúdo, como da topologia do hiperespaço, uma vez que a navegação muda dependendo dos projetos e objetivos escolhidos. Vale observar que a atribuição de objetivos está representada apenas de forma generalizada pela atualização do UM (relação *updates*), uma vez que os objetivos são somente “disparadores” para as outras formas de adaptação.

A adaptação da apresentação, mais especificamente, da interface de suporte à navegação ocorre pela mudança na aparência dos elos para indicar a relevância dos mesmos nos pontos de entrada (*entry points*), na indicação do que explorar em seguida (estrutura de navegação) e, como já visto, na sugestão das IUs adequadas a um objetivo / projeto.

Para a adaptação, o KBS leva em consideração o conhecimento do usuário e seus objetivos de aprendizagem (perfil). Por considerar a dificuldade de se avaliar o conhecimento obtido em visitas às páginas, o KBS não considera o histórico de navegação para a atualização do sistema. Para motivar o estudante a explorar o material de aprendizagem realizando projetos, a atualização só ocorre pela avaliação do desempenho do estudante em algum projeto (funcionalidade da aplicação). Toda a estratégia de adaptação do KBS é baseada nos KIs.

6.5.6 Extensões

No KBS, os estudantes podem “marcar” (*bookmark*) as Unidades de Informação que julgarem interessantes, assim como podem fazer anotações sobre elas. A integração dos projetos dos estudantes ao *hyperbook* é baseada em *portfolios*, usados para demonstrar os conceitos utilizados nos projetos. As *homepages* destes estudantes também são integradas ao *hyperbook*. O KBS gera um glossário a partir do KM.

7 Comparação

Tabela 3. Comparação de alguns AHSs

		AHA!	InterBook	NetCoach	KBS
O Q U Ê	Apresentação: conteúdo				
	Apresentação: navegação	anotação de elos			
	Navegação: Estrutura do Conteúdo	fragmentos condicionais			atribuição projetos e objetivos; indicação Unidades de Informação (IUs)
	Navegação: Topologia do Hiperespaço	elos condicionais	orientação direta (botão "TeachMe"); ajuda baseada em pré-requisitos (<i>special button</i>)	orientação direta (botão "Next")	atribuição projetos e objetivos; indicação IUs; geração trilhas (<i>trails</i>)
	Conteúdo				
E M F U N Ç Ã O	Modelo do Domínio (DM)	conceitos e relações; misturado com AM	rede: nós (conceitos) e elos (relações)	Base de Conhecimento: conceitos + relações	Modelo de Conhecimento (<i>Knowledge Model</i>): Itens de Conhecimento (KIs) + relações
	Perfil do usuário (<i>profile</i>)	conhecimento	conhecimento, objetivos, preferências	conhecimento, objetivos, preferências	conhecimento; objetivo escolhido ou proposto
	Contexto (Situação de uso)	teoricamente, pode ser implementado da mesma forma que o perfil do usuário			
	Histórico de Navegação	páginas visitadas			NÃO CONSIDERADO
D E Q U Ê	Utilização funcionalidade aplicação	solução de testes	solução problemas e <i>quizzes</i>	solução de testes / problemas	desempenho em projetos
	Histórico de Interação				
	Modelo do Usuário (UM)	pares atributo-valor implementados por tabela	pares conceito-valor	relação entre conceito e usuário armazenada em camadas (independência da atualização)	vetor de KI; rede Bayesiana
	Modelo de Adaptação (AM)	regras de geração / requisitos	indexação baseada no Modelo do Domínio	computações baseadas nos resultados dos testes e na estruturação do domínio	baseada nos Itens de Conhecimento (KIs)
C O M O	Agente de entrada (<i>input</i>)	formulário registro + valores <i>default</i>	estereótipo (página de registro)	itens de teste / problemas	desempenho nos projetos julgado pelos estudantes ou por especialistas
	Agente de atualização (<i>update</i>)	sistema / usuário	sistema	sistema / usuário	

Para ilustrar mais um benefício da utilização de meta-modelos, os sistemas são comparados a partir de cada meta-modelo em relação aos aspectos fundamentais da adaptação: "o quê", "em função de quê" e "como". A Tabela 3 apresenta o quadro comparativo dos sistemas estudados. O AHAM não é considerado pois é um modelo e não um sistema implementado. A granularidade de tempo, por ser ainda um objeto de estudo, não aparece. Observa-se, entretanto, que em todos os sistemas o UM persiste entre as sessões e os sistemas são adaptados continuamente, de acordo com os acessos feitos (visita às páginas), e após a resolução de testes / problemas, caso haja. O Interbook pode customizar a adaptação previamente, de acordo com o estereótipo do usuário e alguns sistemas desenvolvidos com o NetCoach aplicam um teste inicial para medir o conhecimento prévio.

Nenhum dos sistemas estudados adapta a apresentação do conteúdo, nem o conteúdo propriamente dito. Por outro lado, todos se valem da anotação de elos para orientar a navegação, embora o tipo de anotação varie entre os sistemas. Para a adaptação da navegação, diferentes técnicas são utilizadas.

A definição do Modelo do Domínio varia entre os sistemas, principalmente porque não há uma distinção clara entre os aspectos conceituais, navegacionais e de implementação. Em essência, o domínio é composto por conceitos estruturados hierarquicamente (relação de subparte) e as relações entre eles.

O AHA! considera que a adaptação ocorre através de eventos que podem ser implementados por módulos. Além disto, o mecanismo de adaptação se baseia em atributos que podem não ser dependentes do domínio. Assim, haveria a possibilidade de se realizar a adaptação em função da situação de uso do sistema. A interação entre os usuários e os sistemas é, basicamente, a navegação. Outras ações não são consideradas para a adaptação. O KBS se distingue por não levar em conta a navegação, se baseando no desempenho em projetos.

Para todos os sistemas, o Modelo do Usuário (ou Modelo do Estudante, quando uma terminologia educacional é utilizada) representa o relacionamento entre o usuário e o domínio, sendo usado, essencialmente, para armazenar a crença que o sistema tem sobre o conhecimento do usuário a respeito dos conceitos do domínio. O KBS se diferencia por usar uma rede Bayesiana para a modelagem. O NetCoach implementa o UM em camadas (*layer*), com atualização independente de cada uma delas [Weber et al 01].

O mecanismo de adaptação é diferente para cada um dos sistemas estudados. Uma característica marcante do KBS é a separação entre o conhecimento e a informação, que permite aos professores determinarem a estrutura navegacional ou conceitual desejada para o material didático, personalizando-o, e torna o sistema robusto a mudanças. O KBS possui, ainda, um esquema peculiar de indexação que permite sua abertura (*openness*). Outra particularidade é a utilização apenas dos projetos para a atualização do sistema.

O InterBook usa estereótipos para a configuração inicial da aplicação. O NetCoach oferece a opção de se fazer uma avaliação preliminar do conhecimento do usuário. Sistemas desenvolvidos com o AHA! e com o NetCoach podem permitir que usuários atualizem o próprio modelo. O KBS permite indiretamente, através da auto-avaliação nos projetos. O NetCoach e o InterBook podem possibilitar ao usuário a alteração de funcionalidades de interface que influenciam no nível de adaptação realizado. O usuário do AHA! pode mexer na adaptação, em menor escala, pela escolha do sistema de cores para a anotação de elos.

8 Conclusão

Sistemas Hipermídia Adaptativos têm, como fim último, orientar aqueles que os utilizam a fazê-lo de forma a aproveitar eficientemente todo o potencial oferecido e Sistemas Hipermídia Meta-Adaptativos objetivam nortear a adaptação de modo a empregá-la de modo eficaz. Sistemas meta-adaptativos deverão, também, ser capazes de estender constantemente o próprio conhecimento sobre a aplicabilidade de diferentes tecnologias, através da observação do desempenho das mesmas em contextos diversos. Nota-se, portanto, a relevância do meta-modelo para AHSs apresentado, que permite

discernir, com detalhes, as possíveis dimensões de adaptação que devem ser contempladas em um modelo que explore todas as oportunidades de adaptação almejadas.

O meta-modelo de referência proposto, inicialmente, procurou exprimir a estrutura dos atuais AHSs. Ele reflete o fato de que aspectos conceituais, navegacionais e de implementação estão misturados nestes sistemas. Não é possível elaborar um meta-modelo comum, do qual os outros sejam casos particulares, porque eles são baseados em premissas muito diferentes. O que se tem, na verdade, são versões ligeiramente alteradas do meta-modelo proposto, que são meta-modelos de cada um dos sistemas. A grande vantagem do meta-modelo de referência proposto é definir, em relação aos aspectos essenciais de adaptação, o núcleo comum aos principais sistemas hipermídia adaptativos propostos na literatura. Ele mostra que o Modelo do Usuário desempenha um papel fundamental nos AHSs, já que para atingir seu objetivo, os AHSs precisam se adaptar aos seus usuários, que são o ponto central da adaptação. Para tanto, é levado em consideração o quanto o usuário sabe sobre cada conceito. Conseqüentemente, existe uma relação entre o Modelo do Usuário e o Modelo do Domínio que expressa esta crença do sistema. As informações do UM e do DM são usadas como entrada para o Modelo de Adaptação. O meta-modelo de referência proposto usa ainda os Modelos de Integração, de Navegação, de Apresentação e o Contexto do Usuário para isolar aspectos de navegação e de implementação, de modo a facilitar o projeto de adaptação.

Com o objetivo de avaliar a relevância do meta-modelo proposto, alguns AHSs foram estudados sob a ótica do mesmo. Observou-se que, apesar de terem sido desenvolvidos para diferentes propósitos e de possuírem características próprias, foi possível identificar o núcleo comum nos sistemas estudados e comparar os mesmos. Para tanto, foi necessário isolar os aspectos conceituais, navegacionais e de implementação presentes nos sistemas.

Inicialmente, procurou-se modelar o que já existe. Em um segundo momento, foram identificados aspectos não considerados pelos sistemas analisados como, por exemplo, as ações executadas durante a navegação. O papel do usuário não é um aspecto do perfil ressaltado. Também não se observou a adaptação da apresentação do conteúdo nem do conteúdo propriamente dito.

O Modelo do Usuário é central à adaptação. Embora não tenha sido classificado assim, observa-se que o UM não responde apenas à questão “como”. O perfil do usuário, armazenado no UM, já é entrada e saída da Função de Adaptação. Entretanto, o UM como um todo poderia ser classificado como “em função de quê”, ou seja, a forma como ele está estruturado poderia influenciar a adaptação. Ou, por outro lado, “o quê” pode ser adaptado influencia a definição do UM. Além disto, o UM é considerado “o quê” apenas na medida em que o conhecimento e, possivelmente, os objetivos do usuário são atualizados. Porém, poder-se-ia modelar o caso em que o sistema percebesse que as preferências indicadas não correspondem ao uso e adaptasse as mesmas (exemplo: apresentação prioriza texto, como indicado, mas usuário sempre amplia imagem). Ou seja, além da apresentação, navegação e conteúdo, poder-se-ia considerar que o sistema também adapta o UM.

A identificação dos tipos de adaptação proporcionados, dos fatores levados em conta para a adaptação e do modo como a adaptação é executada em relação à caracterização do sistema adaptativo, ou seja, seus objetivos, seu público-alvo, sua estrutura, etc. é importante para a definição da meta-adaptação. O meta-modelo de

referência é relevante por facilitar a identificação dos principais aspectos relacionados à adaptação. Com o meta-modelo, a aplicação pode ser decomposta em diversas visões pela separação clara entre os modelos do domínio, do usuário, de navegação, de apresentação, de adaptação, etc. A adaptação pode ocorrer em cada um dos modelos e nas relações entre eles. Esta percepção é essencial para o desenvolvimento de um meta-modelo geral para AHSs que contemple também a meta-adaptação. Por causa das diferenças existentes entre os sistemas, sugere-se que este meta-modelo geral deverá ser, na verdade, um modelo-base e versões dele serão geradas para descrever cada um dos sistemas.

Como já dito, a tendência na área de Hipermissão Adaptativa é o desenvolvimento de sistemas meta-adaptativos. Tais sistemas deverão possibilitar o uso de diferentes técnicas de adaptação e deverão ser capazes de selecionar, de maneira adaptativa, a melhor técnica de acordo com os usuários e a situação de uso [Brusilovsky 03]. Este trabalho é um primeiro passo na direção do desenvolvimento de um meta-modelo geral, capaz de expressar todas as técnicas de adaptação. Suas instâncias seriam a entrada (*input*) para regras de meta-adaptação, utilizadas para a seleção da melhor técnica de adaptação. Este é o ponto de partida para o desenvolvimento dos sistemas meta-adaptativos, onde as regras de adaptação podem atuar, não só sobre os dados, como também sobre os próprios modelos.

9 Trabalhos Futuros

O próximo passo é propor um modelo de autoria para sistemas hipermissão adaptativos que seja meta-adaptável. A independência entre aspectos conceituais, navegacionais e de implementação deverá ser mantida, mas a notação UML não deverá mais ser utilizada, pela sua expressividade limitada. O modelo deverá abordar as questões de adaptação apresentadas, permitindo a especificação da adaptação em função de parâmetros. Como um dos objetivos é que o modelo seja extensível, a definição desses parâmetros, aos quais o modelo vai ser sensível, será um dos pontos de investigação, uma vez que as dimensões não contempladas inicialmente poderão ser facilmente incorporadas através de simples instanciações do meta-modelo.

Essa proposta de arquitetura de Sistemas Hipermissão Adaptativos e Meta-Adaptativos será baseada no SHDM [Lima 2003], que utiliza ontologias para expressar os princípios propostos pelo OOHDM [Schwabe & Rossi 98], além de estendê-lo, possibilitando o mapeamento direto dos modelos em código executável.

Referências Bibliográficas

- Brusilovsky, P. "Adaptive navigation support in educational hypermedia: The role of student knowledge level and the case for meta-adaptation". *British Journal of Educational Technology*, 34 (4), 487-497, 2003.
- Brusilovsky, P. "Methods and Techniques of adaptive hypermedia". In: *User Modeling and User Adapted Interaction*, v. 6, n. 2-3, p. 87-129, 1996 (a).
- Brusilovsky, P. "Adaptive hypermedia, an attempt to analyze and generalize". In P. Brusilovsky, P. Kommers, & N. Streitz (Eds.), *Multimedia, Hypermedia, and Virtual Reality (Lecture Notes in Computer Science, Vol. 1077)*. Berlin: Springer-Verlag, 288-304, 1996 (b).

- Brusilovsky, P., Eklund, J., and Schwarz, E. "Web-Based Education For All: A Tool For Developing Adaptive Courseware", In: Computer Networks and ISDN Systems. Proceedings of Seventh International World Wide Web Conference, 30 (1-7), p.291-300, April 1998.
- De Bra, P., Aerts, A., Smits, D. and Stash, N. "AHA! Meets AHAM". In: Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, Springer LNCS 2347, p. 381-384, May 2002 (a).
- De Bra, P., Aerts, A., Smits, D., Stash, N. "AHA! Version 2.0, More Adaptation Flexibility for Authors". Proceedings of the AACE ELearn'2002 conference, pp. 240-246, October 2002 (b).
- De Bra, P. and Stash, N. "AHA! A General-Purpose Tool for Adaptive Websites". In: World Wide Web Conference, Poster Session, May 2002.
- De Bra, P. and J.P. Ruiter. "AHA! Adaptive Hypermedia for All". In: Proceedings of the AACE Webnet Conference, Orlando, Florida. p. 262-268, 2001
- De Bra, P., Houben, G.J. Houben and Wu, H. "AHAM: A Dexter-based Reference Model for Adaptive Hypermedia". In: Proceedings of the ACM Conference on Hypertext and Hypermedia, Darmstadt, Germany. p. 147-156, 1999.
- Eklund, J., Brusilovsky, P. and Schwarz, E. "Adaptive Textbooks on the WWW". In: Proceedings of Ausweb97, the Third Australian Conference on the World Wide Web, July 5-9, 1997, Queensland, Australia. In: H. Ashman, P. Thistewaite, R. Debrecey and A. Ellis (eds.): Southern Cross University Press, p. 186-192, 1997.
- Frasincar, F. and Houben, G.J. "Hypermedia Presentation Adaptation on the Semantic Web". In: Second International Conference, AH2002, Malaga, Spain. In: Adaptive Hypermedia and Adaptive Web-Based Systems, P. De Bra, P. Brusilovsky, R. Conejo (Eds.), Springer LNCS 2347, p. 133-142, May 2002.
- Frasincar, F., Houben, G.J. and Vdovjak, R. "Specification Framework for Engineering Adaptive Web Applications". In: WWW2002, The Eleventh International World Wide Web Conference, Web Engineering Track, Honolulu, Hawaii, USA, May 2002.
- Fröhlich, P., Nejd, W. and Wolpers, M. "KBS-HYPERBOOK - an Open Hyperbook System for Education". In: Proceedings of the Ed-Media World Conference on Educational Multimedia and Hypermedia, Freiburg, Germany, June 1998.
- Henze, N. "Open Adaptive Hypermedia: An approach to adaptive information presentation on the Web". In: First International Conference on Universal Access in Human-Computer Interaction (UAHCI 2001); HCI international 2001, held jointly with HCI International 2001, New Orleans, Louisiana, USA, August 2001.
- Henze, N. and Nejd, W. "Adaptivity in the KBS Hyperbook System". In: Proceedings of the 2nd Workshop on Adaptive Systems and User Modeling on the WWW, May 11th Toronto, and June 24th, Banff. Held in conjunction with the World Wide Web (WWW8) and the International Conference on User Modeling, 1999 (a).
- Henze, N. and Nejd, W. "Bayesian Modeling for Adaptive Hypermedia Systems". ABIS99, 7. GI-Workshop Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen, Otto-von-Guericke-Universität Magdeburg, 29./30.9.1999 (b).

- Henze, N., Nejdil, W. and Wolpers, M. "Modeling Constructivist Teaching Functionality and Structure in the KBS Hyperbook System". In: Computer Supported Collaborative Learning Conference (CSCL'99), Stanford, CA, USA, December 1999.
- Isakowitz, T., Stohr, E and Balasubramanian, P. "RMM: A Methodology for Structured Hypermedia Design". *Communications of the ACM*, 38(8), 34-44, 1995.
- Koch, N.P. "Software Engineering for Adaptive Hypermedia Systems: Reference Model, Modeling Techniques and Development Process" Tese de Doutorado - Institut für Informatik, Ludwig-Maximilians-Universität München, 2000.
- Lima, F. "Modelagem Semântica de Aplicações na WWW", Tese de Doutorado, Departamento de Informática, PUC-Rio, Março 2003.
- Medina-Medina, N., García-Cabrera, L., Rodríguez-Fortiz, M.J. and Parets-Llorca, J. "Adaptation in an Evolutionary Hypermedia System: Using Semantic and Petri Nets". *Adaptative Hypermedia and Adaptative Web-Based Systems* (second international conference, AH 2002). En *Lectures Notes in Computer Science*. Vol. 2347. Pp: 284-295. Ed LNCS Editorial, Germany. ISSN: 0302-9743, 2002.
- Nejdil, W. and Wolpers, M. "KBS Hyperbook - a Data-Driven Information System on the Web". In: WWW8 CONFERENCE, Toronto, May 1999.
- NetCoach Manual. <http://art.ph-freiburg.de/NetCoach-Manual>, último acesso em março/2003.
- Rossi, G., Schwabe, D. and Lyardet, F; "Web application models are more than conceptual models", *Lecture Notes in Computer Science* 1727, pp. 239-252, ISBN 3-540-66653-2, *Proceedings of the World Wild Web and Conceptual Modeling'99 Workshop*, ER'99 Conference, Springer, Paris, 1999.
- Schwabe, D. and Rossi, G. "An Object Oriented Approach to Web-Based Application Design", *Theory and Practice of Object Systems* 4(4). Wiley and Sons, New York, ISSN 1074-3224), 1998.
- Specht, M and Oppermann, R. "ATS - Adaptive Teaching System a WWW-based ITS". In U. Timm (Eds.), *Proceedings of Workshop Adaptivität und Benutzermodellierung in Interaktiven Softwaresystemen: ABIS*, 1998.
- UML Resource Center. <http://www.rational.com/uml/>, último acesso em dezembro/2002.
- UML Resource Page. <http://www.omg.org/uml/>, último acesso em dezembro/2002.
- Vdovjak, R., Frasincar, F., Houben, G.J. and Barna, P. "Engineering Semantic Web Information Systems in Hera". In: *Journal of Web Engineering*, Vol. 2, No. 1&2, p. 3-26, Rinton Press, 2003.
- Weber, G. and Brusilovsky, P. "ELM-ART: an adaptive versatile system for web-based instruction". In: *International Journal of Artificial Intelligence in Education*, 2001.
- Weber, G., Kuhl H.-C. and Weibelzahl, S. "Developing adaptive internet based courses with the authoring system NetCoach". In: Reich, S., Tzagarakis, M.M. & De Bra, P.M.E., (Eds.) *Hypermedia: Openness, Structural Awareness, and Adaptivity* (pp. 226-238). Berlin: Springer-Verlag, 2001.

- Weber, G. and Specht, M. "User modeling and adaptive navigation support in WWW-based tutoring systems". In: Proceedings of User Modeling '97, Wien. Springer-Verlag, p. 289-300, 1997.
- Wu, H. "A Reference Architecture for Adaptive Hypermedia Applications". Tese de Doutorado. Technische Universiteit Eindhoven, 2002.
- Wu, H., De Bra, P., Aerts, A. and Houben, G.J. "Adaptation Control in Adaptive Hypermedia Systems". Proceedings of the AH2000 Conference, pp. 250-259. Lecture Notes in Computing Science, Vol. 1892, Springer, 2000 (a).
- Wu, H., Houben, G.J. and De Bra, P. "Supporting User Adaptation in Adaptive Hypermedia Applications". On-line Conference and Informatiewetenschap (De Doelen, Rotterdam), 2000 (b).
- Wu, H., De Kort, E. "Cross-References in Web-Based Adaptive Hypermedia". Proceedings of the AH'2002 Workshop on Personalization in Electronic Publishing, Malaga, Spain, May 2002.
- Wu, H., De Kort, E. and De Bra, P. "Design Issues for General-Purpose Adaptive Hypermedia Systems". In: Proceedings of the ACM Conference on Hypertext and Hypermedia, Aarhus, Denmark. p. 141-150, August 2001.