

2 Fundamentos

2.1. OOHDM e SHDM

O Object Oriented Hypermedia Design Method (OOHDM) [Schwabe & Rossi, 98] e sua versão semântica Semantic Hypermedia Design Method (SHDM) [Lima & Schwabe, 2003] são métodos de desenvolvimento de aplicações hipermídia que permitem uma concisa especificação e posterior implementação de aplicações Web. O OOHDM faz uso de diversos modelos que se diferenciam de outros métodos de desenvolvimento ao enfatizar o aspecto navegacional da aplicação.

2.1.1.Fases

O método OOHDM é composto de cinco fases: levantamento de requisitos, projeto conceitual, projeto navegacional, projeto da interface abstrata e implementação. Cada fase enfatiza um interesse em particular.

2.1.1.1.Levantamento de Requisitos

Nesta fase é realizado o levantamento dos requisitos. Para isto é necessário identificar os atores, que são os usuários ou sistemas externos que interagem com a aplicação, e as tarefas que estes podem executar. Também são identificados os cenários, que devem ser descritos de forma textual, que são representados utilizando um Diagrama de Iteração do Usuário (User Interaction Diagrams - UID). Estes diagramas proporcionam uma representação gráfica da interação do usuário com o sistema durante a execução de uma tarefa. Um exemplo de UID é apresentado na Figura 1. Neste diagrama está exemplificada uma tarefa de login no sistema.

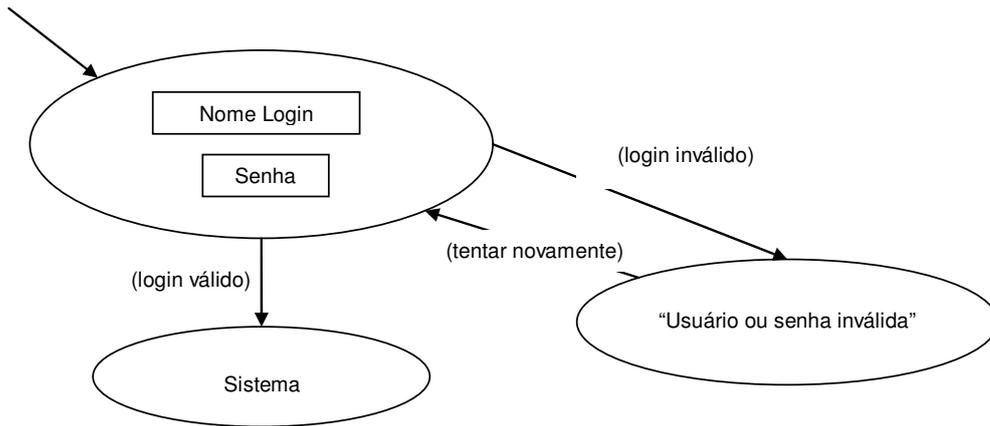


Figura 1 - Exemplo de UID representando uma aplicação de login

Para detalhes sobre a notação deste diagrama, ver [Schwabe, 2000].

2.1.1.2. Projeto Conceitual

Nesta etapa, é desenvolvido um modelo da aplicação utilizando princípios de modelagem orientada a objetos. Para isto o projetista deve analisar o domínio da aplicação, com o objetivo de produzir um modelo conceitual que represente objetos e relacionamentos existentes no domínio do sistema de forma não ambígua .

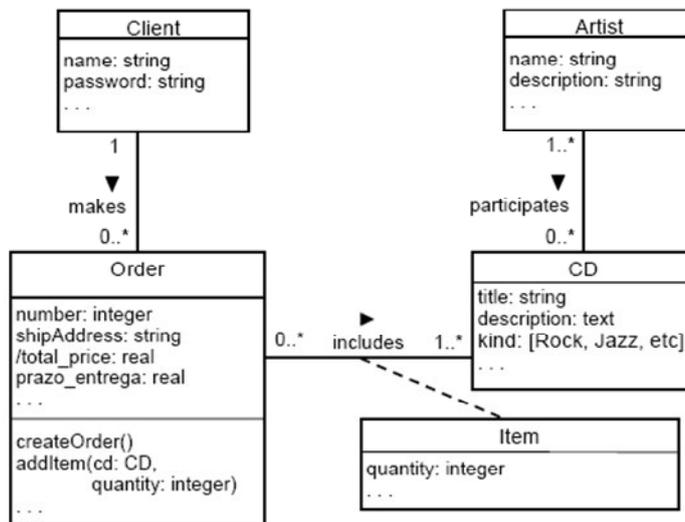


Figura 2 – Exemplo de modelo conceitual de uma Loja de CDs apresentado por [Jacyntho, 2002].

2.1.1.3. Projeto Navegacional

O projeto navegacional da aplicação é uma das principais inovações do OOADM. Nesta fase são desenvolvidos o modelo de classes navegacionais e o modelo de contextos navegacionais.

O modelo de classes navegacionais é desenvolvido baseado no mapeamento das classes conceituais em classes navegacionais, criando uma visão sobre o modelo conceitual. Os relacionamentos do modelo conceitual, que se transformam em elos no modelo navegacional, podem também ser transformados em outros atributos navegacionais como âncoras ou índices, por exemplo.

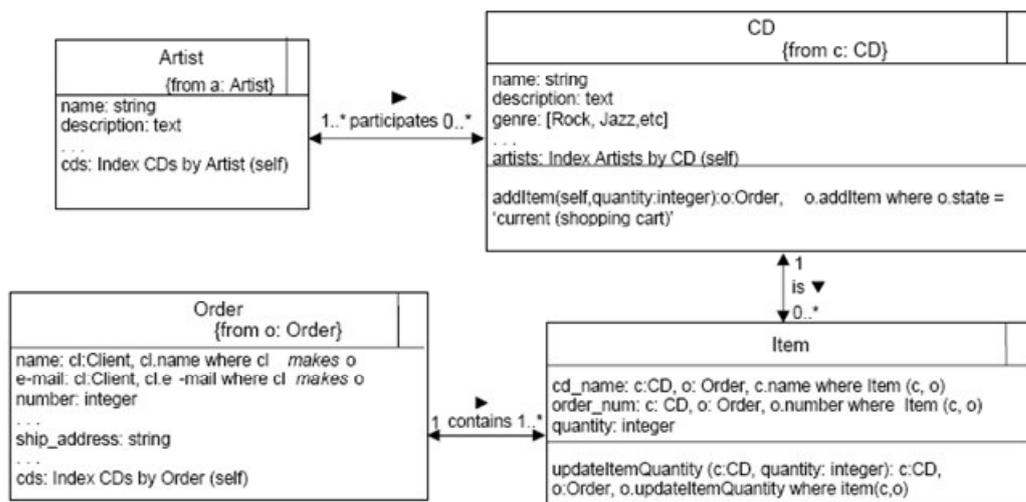


Figura 3 – Exemplo de Diagrama de Classes Navegacionais apresentado por [Jacyntho, 2002].

O modelo de contextos navegacionais representa todas as possibilidades de navegação que estarão disponíveis na aplicação. Esse modelo é composto de contextos, estruturas de acesso e do relacionamento entre eles.

Estruturas de acesso são índices ou dicionários cuja função é ajudar o usuário a encontrar a informação desejada.

Contextos são formas utilizadas para organizar informações envolvendo classes navegacionais.

O relacionamento entre Estruturas de acesso e Contextos é feito através de âncoras que expressam as possibilidades de navegação entre os índices e os contextos.

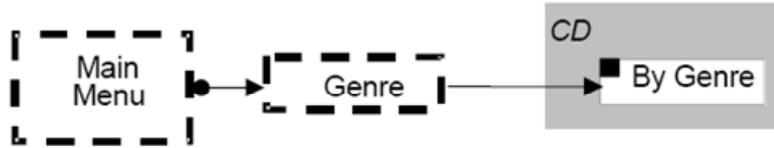


Figura 4 – Exemplo de um Diagrama de Contextos Navegacionais apresentado por [Jacyntho, 2002].

2.1.1.3.1. Classes Navegacionais

Classes navegacionais definem quais dados do modelo conceitual poderão ser acessados pelos usuários da aplicação. As instâncias de classes navegacionais são chamadas de nós.

Uma classe navegacional é composta por atributos, operações e elos, além das relações de especialização/generalização que podemos utilizar para constituir uma herança de classes.

2.1.1.3.1.1. Atributos

Uma classe navegacional é composta de atributos que estão divididos em 4 categorias descritas abaixo:

2.1.1.3.1.1.1. Atributo Simples

São atributos que têm um mapeamento direto para um atributo primitivo. No OOHDN são considerados primitivos os tipos de dados definidos no XML Schema.

2.1.1.3.1.1.2. Atributo Computado

São atributos cujo valor é obtido através de alguma operação executada em tempo de execução.

2.1.1.3.1.1.3. Atributo Âncora

São atributos que definem um caminho navegável entre a instância da classe navegacional e alguma outra instância ou índice. Este caminho é definido em tempo de execução baseado em uma relação navegacional.

2.1.1.3.1.4. Atributo Índice

É um atributo que comporta uma Estrutura de Acesso ou Índice, ou seja, um conjunto de tuplas onde cada tupla contém pelo menos uma âncora para um nó em contexto ou outra estrutura de acesso.

2.1.1.3.1.2. Operações

As operações nas classes navegacionais têm a mesma semântica dos métodos no modelo conceitual ou do modelo orientado a objetos. As operações são definidas por um cabeçalho e por um bloco de código que é executado quando ativado.

2.1.1.3.1.3. Relacionamentos (Elos)

Os relacionamentos entre nós ou elos delimitam os possíveis caminhos de navegação da aplicação.

Elos são as ligações entre nós, e entre nós e estruturas de acesso, e definem os possíveis caminhos de navegação da aplicação. Os elos de uma aplicação SHDM podem ser definidos através do mapeamento dos relacionamentos do modelo conceitual ou através da declaração de âncoras em estruturas de acesso ou classes navegacionais.

2.1.1.3.2. Contextos Navegacionais

Um contexto navegacional é composto por um conjunto de nós ou instâncias de classes navegacionais. Este conjunto de nós que compõem um contexto é definido através de uma consulta. Esta consulta pode ter como critério de seleção atributos ou meta-atributos do modelo navegacional.

2.1.1.3.3. Classe em contexto

Uma classe em contexto é um decorador [Gamma et al., 1995] de uma classe navegacional. Um decorador acrescenta atributos a uma instância da classe

navegacional permitindo que este nó possa apresentar informações diferentes de acordo com o contexto navegacional em que está sendo exibido.

No SHDM .Net a visibilidade ou não de um atributo navegacional é definida apenas no projeto da interface abstrata.

2.1.1.3.4.Estrutura de Acesso

Estruturas de Acesso ou Índices são formados por um conjunto de valores dentre os quais pelo menos um valor tem que ser uma âncora para uma outra estrutura de acesso ou um nó em um determinado contexto. Estruturas de acesso são organizadores de informações que servem de ponte entre uma identificação de uma informação e sua descrição mais detalhada.

2.1.1.3.5.Âncoras

O objetivo é definir um caminho navegacional que pode ser percorrido por um usuário. Este caminho pode ter como origem uma estrutura de acesso conectando a outra estrutura de acesso ou a um determinado nó em um contexto, bem como pode ter origem em um nó de um contexto conectando-o a um outro nó.

2.1.1.3.6.Landmarks

Um Landmark define quais índices ou contextos podem ser acessados a qualquer momento na aplicação. Os landmarks servem também como menus ou pontos de entrada na aplicação.

2.1.1.3.7.Nós

Os nós são instâncias de classes navegacionais definidas no modelo de classes navegacionais da aplicação. Estes só podem ser acessados através de um contexto navegacional.

2.1.1.3.8.Navegação Facetada

Navegação facetada é um sistema de navegação que provê diversos caminhos navegacionais para uma mesma informação. Este estilo de navegação

foi introduzido no SHDM por [Lima & Schwabe, 2003]. Este tipo de navegação é implementada através da aplicação de filtros sucessivos em uma coleção de objetos. A versão atual do SHDM .Net não dá suporte à navegação facetada.

2.1.1.4. Projeto da Interface Abstrata

O projeto da Interface Abstrata tem por objetivo especificar os objetos perceptíveis que estarão disponíveis ao usuário. Cada objeto da interface final do usuário é representado por um objeto abstrato, independente de tecnologia. Este objeto abstrato tem um mapeamento com um objeto navegacional. Desta forma é possível saber quais informações estarão disponíveis para o usuário da aplicação mesmo desconhecendo como esta informação será apresentada.

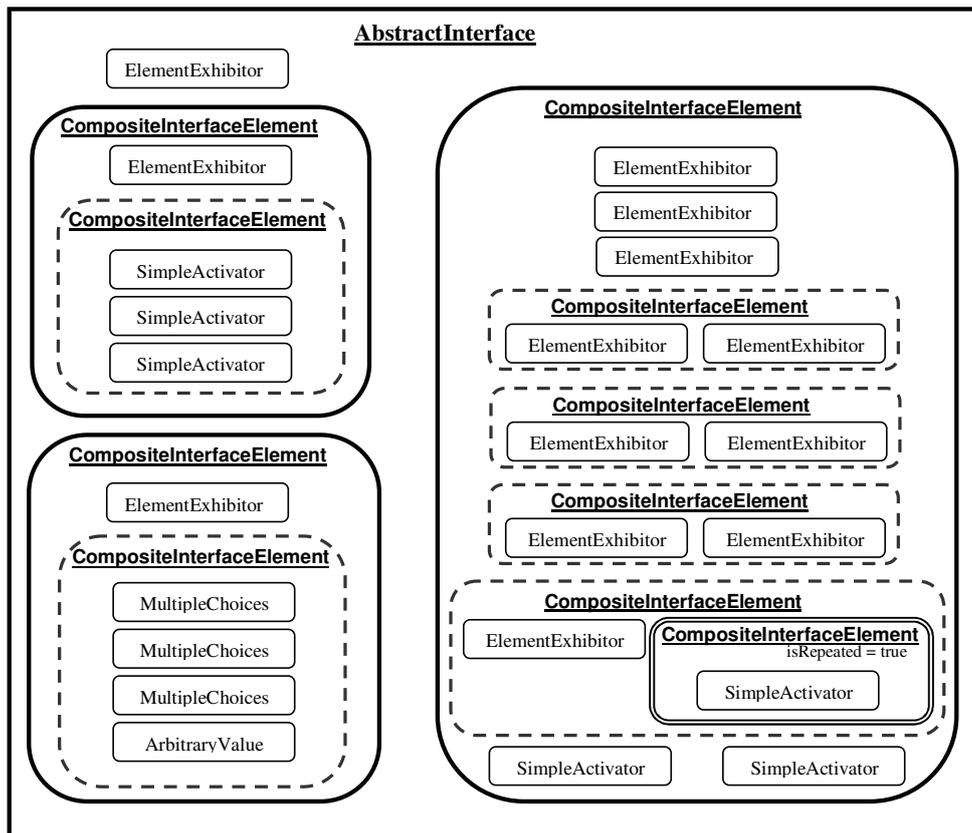


Figura 5 - Exemplo de Interface Abstrata apresentado por [Moura, 2004].

2.1.1.5. Projeto da Interface Concreta

O projeto da interface concreta da aplicação não faz parte de um ciclo de desenvolvimento OOHDM / SHDM. Esta é uma das contribuições propostas por

esta dissertação. Colocamos esta etapa do desenvolvimento na parte de fundamentos com o objetivo de deixar a base da dissertação mais concisa.

O projeto da interface concreta compreende o mapeamento dos elementos da interface abstrata em elementos concretos. Os elementos concretos da interface são definidos em uma ontologia de elementos concretos, permitindo que os elementos abstratos sejam mapeados em um elemento concreto de uma determinada tecnologia. A ontologia de elementos concretos é apresentada no item 4.6. Após este mapeamento a interface concreta pode ser gerada de forma automática, baseada no diagrama de interface abstrata, na ontologia de elementos concretos e no mapeamento entre eles.

Um exemplo do diagrama de interface concreta é apresentado na Figura 6. No item 4.5 esta nova fase é descrita com maiores detalhes. O diagrama de interface concreta pode ser compreendido como um diagrama de interface abstrata com dependência tecnológica, pois são adicionados algumas propriedades tecnológicas em um diagrama de interface abstrata, tornando-o dependente da tecnologia de implementação.

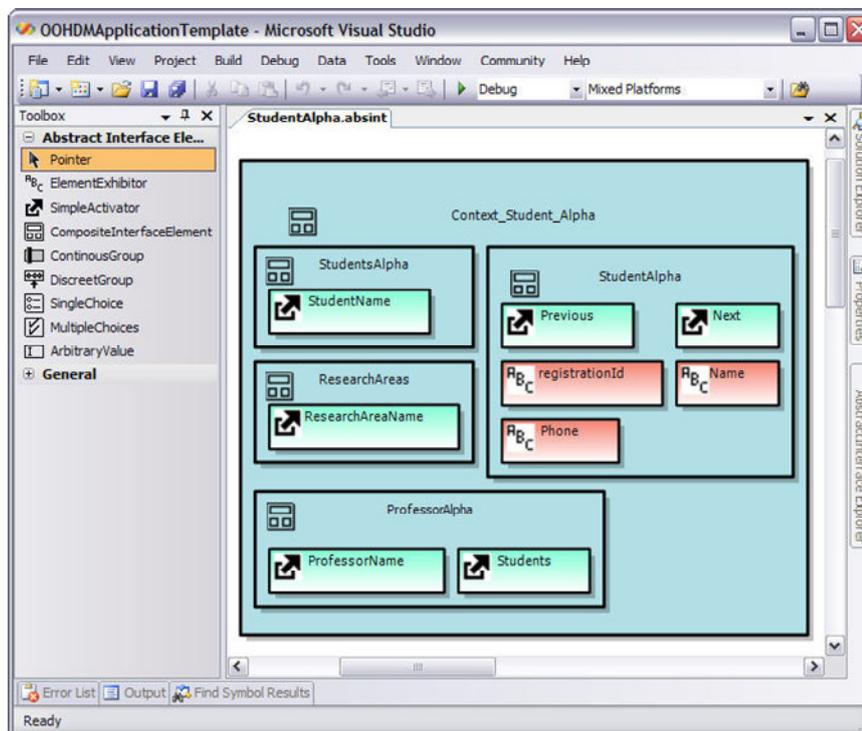


Figura 6 - Exemplo do Diagrama de Interface Concreta

2.1.1.6.Implementação

A etapa de implementação é a fase onde é realizado o mapeamento dos modelos definidos nas fases anteriores em uma aplicação. Somente nesta fase o desenvolvedor estará sujeito às particularidades de implementação ocasionadas pela tecnologia a ser utilizada.