



Departamento de Informática  
PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO



## OOHDM

### Object-Oriented Hypermedia Design Method

*Gustavo Rossi e Daniel Schwabe*

Adriana Pereira de Medeiros

e-mail: adri@inf.puc-rio.br



## Resumo



- Introdução
  - Hipermídia
  - Tipos de Aplicações Hipermídia
- Processo de Desenvolvimento de Aplicações Hipermídia
- O Método OOHDM
  - Levantamento de Requisitos
  - Projeto Conceitual
  - Projeto da Navegação
  - Projeto da Interface Abstrata
  - Implementação



## Definição de Hipermedia



- Hipermedia é um estilo de se construir sistemas para a criação, manipulação, apresentação e representação da informação no qual
  - a informação é armazenada em uma coleção de nós multimedia;
  - os nós são organizados implicita ou explicitamente em uma ou mais estruturas;
  - os usuários podem acessar a informação navegando através das estruturas de informação disponíveis.

*(Definição de Hipermedia proposta por Halasz na HT'91)*



## Aplicações Hipermedia

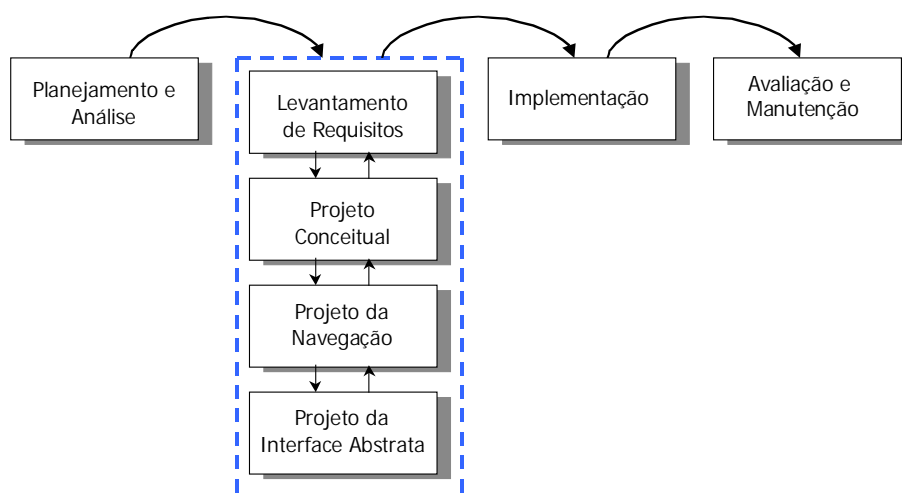


- Aplicações hipermedia são sistemas de informação avançados que permitem a representação do conhecimento
  - informalmente - para processamento feito pelo homem (hipertexto/hipermedia)
  - formalmente - para processamento feito pela máquina (inteligência artificial/banco de dados)

## Tipos de Aplicações Hipermedia

- Comerciais
  - Documentação Técnica - manuais, normas, etc.
  - Treinamento
  - Publicações
  - Catálogos e Propaganda
  - Lojas
  - Comparação de Produtos
  - Leilões
  - Repositórios (imagens, músicas, software)
  - Legislação
- Computação
  - Documentação online
  - Assistência ao usuário (help) - erros
  - Groupware
- Outros
  - Guias
  - Enciclopédias
  - Livros
  - Jogos
  - Museus

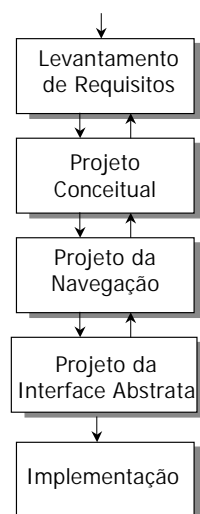
## Processo de Desenvolvimento



## O Método OOHDM

- Método integrado para autoria de aplicações hipermídia
- Pode ser usado como forma de comunicação entre projetistas, implementadores e usuários
- Permite a implementação em diversos ambientes de hardware e software, não necessariamente orientados a objetos
- Permite descrever a aplicação independentemente de sua implementação

## Atividades do método OOHDM





## Levantamento de Requisitos

### ■ Etapas:

- Identificação dos atores e tarefas
- Especificação dos Cenários
- Especificação dos Use Cases
- Especificação dos Diagramas de Interação do Usuário (UIDs)
- Validação dos Use Cases e dos UIDs



## Identificação de Atores e Tarefas

- O projetista interage com o domínio para identificar os atores e as tarefas que serão apoiadas pela aplicação
  - Análise dos documentos disponíveis
  - Interações com clientes e usuários
- Exemplo: sistema de revisão para conferências
  - Atores: revisor, autor
  - Tarefas
    - Autor
      - modificar dados sobre um artigo
      - submeter artigo
    - Revisor
      - incluir uma revisão de artigo



## Especificação dos Cenários



- O usuário especifica textualmente ou verbalmente os cenários das tarefas que serão executadas com o apoio da aplicação
- Cenários são descrições narrativas que explicam detalhadamente como a aplicação pode ser utilizada para realizar tarefas
- Não é necessário abordar aspectos de interface nem de navegação



## Especificação dos Use Cases



- O projetista especifica um use case para cada tarefa a partir dos cenários dos usuários
- Os cenários que descrevem uma mesma tarefa são agrupados em um único use case
- No use case são incluídas todas as informações encontradas nos seus cenários
- Use cases podem ser incrementados com informações de outros use cases
- Um use case descreve um uso da aplicação, sem considerar aspectos internos da aplicação
- Um cenário pode ser visto como uma instância específica de um use case



## Use Case - Exemplo



Use Case: modificar dados sobre um artigo

Descrição:

1. A aplicação apresenta uma lista de todos os artigos onde o autor aparece como autor ou co-autor. A lista mostra o id e o título do artigo
2. O autor seleciona um artigo e a aplicação apresenta seus dados: id, título, autor, lista de co-autores, resumo, sessão sugerida e lista de tópicos
3. O autor pode alterar o título, o resumo, co-autores, sessão sugerida, a lista de tópicos e enviar um novo arquivo relacionado ao artigo



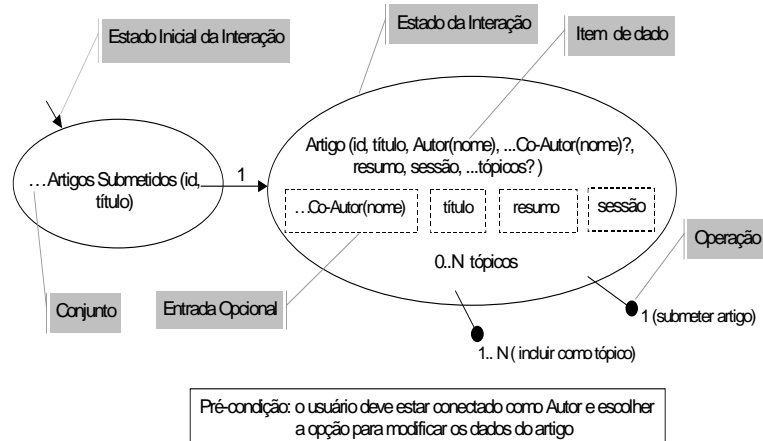
## Especificação dos UIDs



- Os diagramas de interação do usuário (UIDs) representam graficamente as interações entre o usuário e a aplicação, descritas textualmente nos use cases
- Definimos um UID para cada use case

## UID - Exemplo

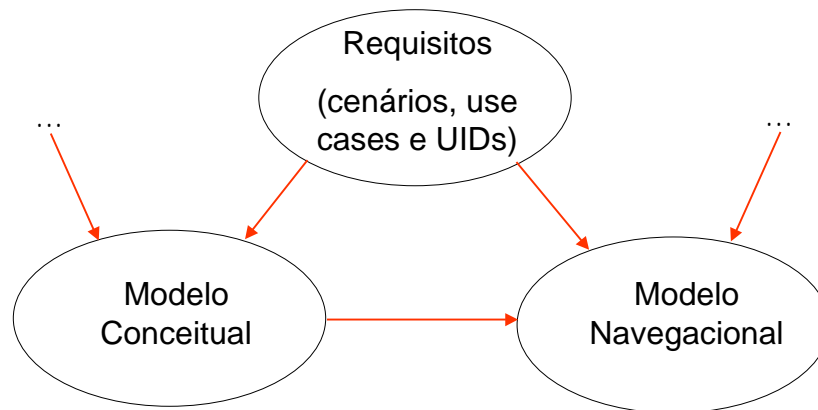
Use Case: modificar dados sobre um artigo



## Validação dos Use Cases e UIDs

- A validação é feita através da interação do projetista com os usuários: o projetista interage com cada usuário
- Cada use case e diagrama de interação do usuário é apresentado ao usuário
- O cenário descrito pelo usuário é referenciado
- Idealmente fazer interações até que os usuários cheguem a um consenso

## Levantamento de Requisitos



© Adriana P. de Medeiros e Daniel Schwabe, 2001

17

## Projeto Conceitual

- Atividade responsável pela análise do domínio da aplicação
- Preocupação com a estrutura conceitual da informação, não com a aparência ou formas de uso
- Poderá ser potencialmente usado por diversas aplicações no mesmo domínio (análogo a BD's)
- Definição de forma independente da plataforma de hardware e software.

Resultado: **Esquema Conceitual**

© Adriana P. de Medeiros e Daniel Schwabe, 2001

18



## Primitivas OOHDM



- Objetos
  - Atributos – Tipos e Perspectivas
- Classes
  - Agregações
  - Generalização/Especialização
- Relações
- Sub-Sistemas
- Esquemas de Classes e de Instâncias

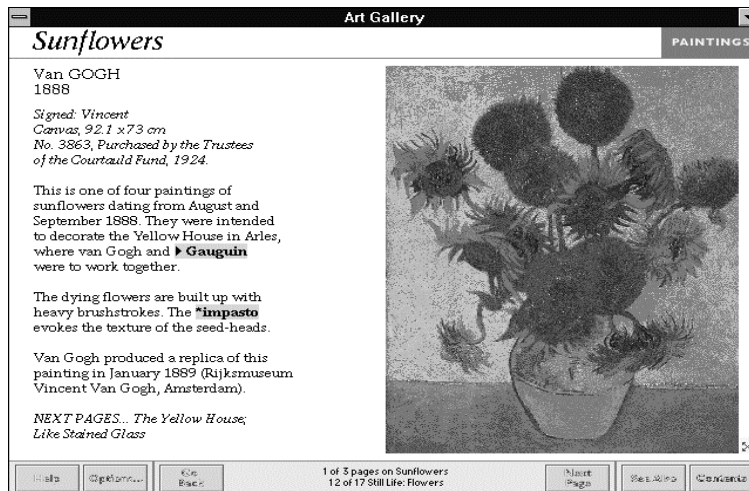


## Objetos e Atributos



- Um objeto é um conceito do domínio da aplicação
  - “A pintura Sunflowers”, “O revisor Bruno”,...
- Objetos são descritos através de atributos
  - nome, descrição, localização,...
- Os atributos podem ser de qualquer tipo
  - texto, vídeo, som, imagem, número, ...
- Um atributo pode ter múltiplas perspectivas
  - descrição: [texto+, foto:imagem]
- As perspectivas de um atributo podem ser associadas à mídia, idioma, etc.
  - texto em português/inglês, descrição formal/informal, tabela...

## Objeto - Exemplo



© Adriana P. de Medeiros e Daniel Schwabe, 2001

21

## Classes

- Uma classe é a descrição de um conjunto de objetos que compartilham os mesmos atributos, operações, relacionamentos e semânticas
  - uma pintura, um artigo, um produto, uma pessoa...

◆ Notação:

Artigo
id: string título: string resumo: text status: string
submeter_artigo () incluir_como_tópico ()

Pessoa
nome: string descrição:[currículo: text+, foto:image]

© Adriana P. de Medeiros e Daniel Schwabe, 2001

22

## Classe - Exemplo

**Pintura**

nome: String

Autor: String

Data: Date

dadosTécnicos: Text

descrição: [Text+,  
pintura: Image]

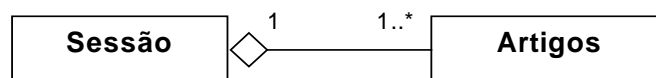
© Adriana P. de Medeiros e Daniel Schwabe, 2001

23

## Mecanismos de Abstração

### ■ Agregação

- Indica que um objeto é composto por outros objetos
- Classes mais complexas podem ser descritas como uma agregação (ou composição) de classes mais simples



© Adriana P. de Medeiros e Daniel Schwabe, 2001

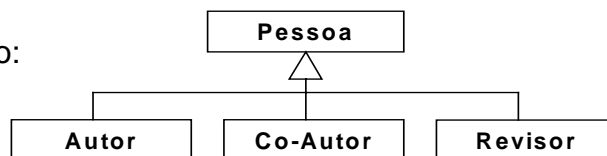
24

## Mecanismos de Abstração

### ■ Generalização/Especialização

- É utilizada para denotar que classes especializadas (sub-classes) herdam todas as características de uma classe generalizada (superclasse)
- Os atributos de uma classe são herdados pelas sub-classes. As perspectivas se acrescentam e se considera *default* o atributo de nível mais baixo na hierarquia

#### ◆ Notação:



© Adriana P. de Medeiros e Daniel Schwabe, 2001

25

## Relações

- A semântica de relações pode ser determinada
  - pela relação entre os objetos do domínio sendo representada
  - pela relação funcional entre os itens de informação sobre os objetos
- Exemplos:
  - Causa/influência
  - Posse
  - Parte de/Contido em
  - Composto de
  - Generaliza/Resume
  - Explica/Demonstra/Ilustra
  - Modifica/Corrige
  - Comenta

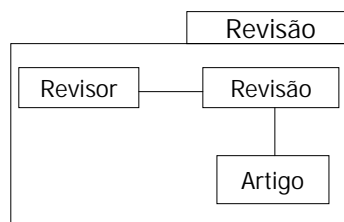
© Adriana P. de Medeiros e Daniel Schwabe, 2001

26

## Sub-Sistemas

- Representam o agrupamento de classes e relacionamentos que tratam de um mesmo assunto e, juntos, são independentes do resto da aplicação, ou seja, são abstrações de um sistema conceitual completo

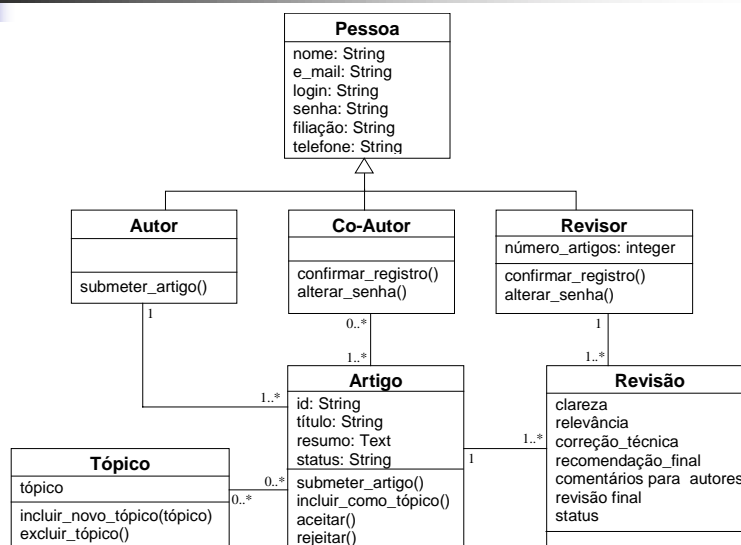
◆ Notação:



© Adriana P. de Medeiros e Daniel Schwabe, 2001

27

## Esquema Conceitual



© Adriana P. de Medeiros e Daniel Schwabe, 2001

28



## Projeto da Navegação



- Define as informações que serão apresentadas e a possível navegação entre elas
- A navegação é determinada primordialmente pelo tipo de usuário e pela tarefa a ser desempenhada

Resultado: **Esquema Navegacional**  
**Esquema de Contextos**  
**Cartões de Especificação**



## Estrutura de Navegação



- Que estruturas o usuário terá acesso?
  - Nós
- Por onde o usuário poderá navegar?
  - Elos
- Em que conjuntos de objetos o usuário irá navegar?
  - Contextos
- Os objetos navegados poderão ter aparências diferentes de acordo com o contexto?
  - Classes em Contexto
- Quais as formas de navegação para o acesso?
  - Estruturas de acesso, visitas guiadas, etc...



## Nós (classes navegacionais)

- Um nó é um objeto navegável definido a partir de objetos conceituais (a base do nó)
- Nós são definidos oportunisticamente
  - analogia: coador e açúcar na prateleira de café do supermercado
- Em alguns casos, um nó pode ser isomorfo (igual) a um objeto conceitual
- Um nó não corresponde necessariamente a uma unidade de percepção

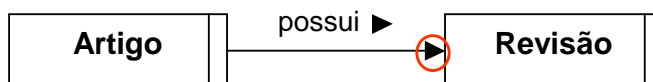


## Atributos

- Cada nó pode apresentar:
  - todos os atributos da sua classe conceitual base
  - somente alguns atributos da classe conceitual base
  - atributos de outras classes conceituais
- Os atributos de um nó só podem apresentar uma perspectiva. O mapeamento de atributos com várias perspectivas, sendo uma default, resulta em vários atributos diferentes
- As âncora, listas e índices apresentados em um nó são especificados como atributos, e são, respectivamente, do tipo *âncora*, *lista* e *índice*

## Elos e Âncoras

- Elos implementam as relações, e permitem aos usuários navegar entre os nós
- Classes derivadas de relações (relações com atributos) corresponderão a classes de nós
- Âncoras especificam como os elos devem ser percebidos
- Âncoras sempre dependem do contexto em que o nó correspondente é visitado

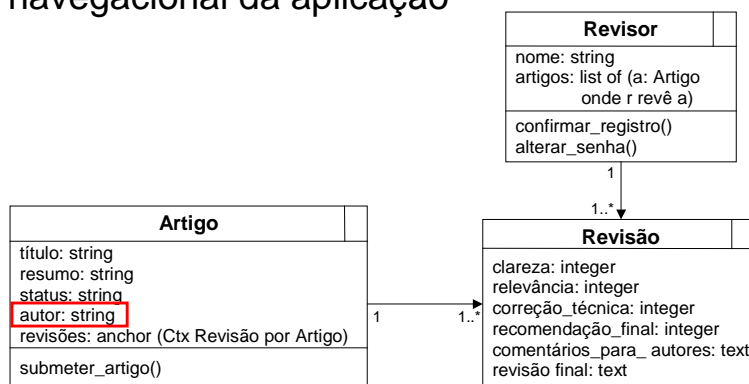


© Adriana P. de Medeiros e Daniel Schwabe, 2001

33

## Esquema Navegacional

- Um esquema navegacional define o conjunto de nós e elos que fazem parte de uma visão navegacional da aplicação



© Adriana P. de Medeiros e Daniel Schwabe, 2001

34



## Contextos de Navegação

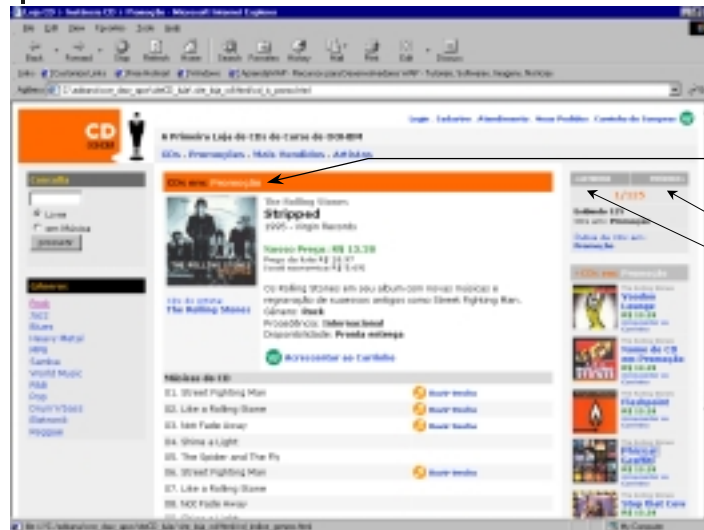
- São conjuntos de objetos que serão explorados pelo usuário durante a realização de uma tarefa
- Determinam o conjunto de objetos de navegação acessíveis a cada momento
- Os contextos também ordenam os objetos, dando significado a “próximo” e “anterior”
- Diferentes tipos de contexto são definidos a partir de nós e elos
- Tem formas pré-definidas de acesso e de navegação entre seus elementos



## Contexto de Navegação - Exemplos

- Sistema de revisão para conferência
  - Todos os artigos em ordem alfabética
  - Todos os artigos de um autor
  - Todas as revisões de um artigo
  - Todos os artigos com o mesmo status
- Loja de CDs online
  - CDs em promoção
  - Todos os CDs da Marisa Montes
  - Todos os CDs de MPB
  - Os CDs mais vendidos

## Contexto de Navegação - Exemplo



Contexto  
"CDs em  
promoção"

"Próximo CD"  
"CD Anterior"

© Adriana P. de Medeiros e Daniel Schwabe, 2001

37

## Classificação dos Contextos

- Caracterização dos elementos componentes
  - Arbitrário, enumerado ou extensional - o autor do contexto seleciona os elementos
  - Não arbitrário ou derivado – define-se uma regra de seleção dos elementos
- Permanência dos elementos no contexto
  - Estático
  - Dinâmico
- Duração do contexto
  - Limitado a seção de criação, pode tornar-se persistente

© Adriana P. de Medeiros e Daniel Schwabe, 2001

38

## Contexto Derivado de Classe

- Os objetos deste tipo de contexto pertencem todos a uma mesma classe e são selecionados por satisfazerem alguma condição
- Exemplos
  - Loja de CDs
    - “Todos os CDs de MPB”
  - Sistema de Revisão para Conferência
    - “Todos os artigos aceitos”

◆ Notação:

*Artigo*

status = 'Aceito'

## Contexto Derivado de Elo

- É formado por objetos de uma mesma classe, que são selecionados por pertencer a uma relação 1-a-n com outra classe
- Exemplos
  - Loja de CDs
    - “Todos os CDs da Marisa Montes”
  - Sistema de Revisão para Conferência
    - “Todos os artigos revisados por Daniel Schwabe”

◆ Notação:

*Artigo*

Revisor = 'Daniel Schwabe'

## Grupos de Contexto

- É um conjunto de contextos
- A regra de seleção é expressa de forma parametrizada
- Exemplos:
  - Grupo derivado de classe
    - “CDs por gênero musical”
    - “Artigos por status”
  - Grupo derivado de elo
    - “CDs por cantor”
    - “Artigos por revisor”

◆ Notação:

*Artigo*

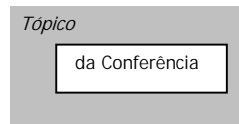
por Revisor

## Contextos: Enumerado e Dinâmico

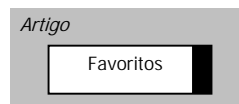
- Enumerado – é um contexto no qual seus elementos são explicitamente enumerados, e podem pertencer a classes diversas
  - visitas guiadas
- Dinâmico – é um contexto cujos elementos são determinados dinamicamente, como resultado do processo de navegação
  - “histórias” mantidas pelos *browsers* para WWW
  - “bookmarks” – usuário acrescenta referências

## Notação

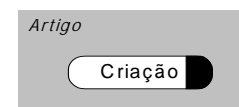
- Contexto Enumerado:



- Contexto Dinâmico:



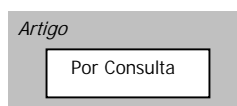
- Contexto Dinâmico de Criação:



## Contexto por Consulta

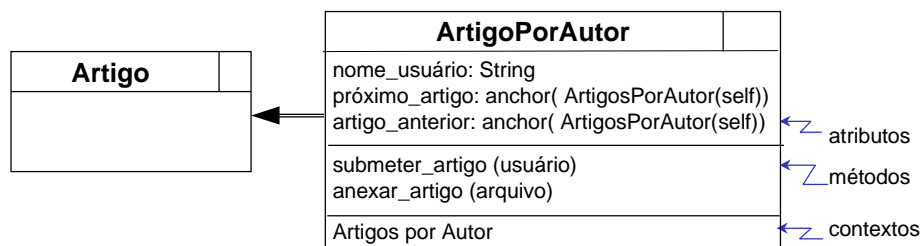
- Os elementos são selecionados a partir de uma cláusula de consulta que utiliza parâmetros cujo valor é determinado em tempo de execução
- Os valores dos parâmetros podem ser:
  - calculados pelo sistema, de forma automática
    - "Todos os artigos submetidos nos últimos 30 dias"
  - fornecidos pelo próprio usuário, de forma interativa
    - "Todos os artigos que possuem a palavra-chave "Hipermissão" em uma das opções fornecidas pelo usuário: título ou resumo"

- ◆ Notação:



## Classes em Contexto

- Classes especiais que decoram os nós, permitindo que um mesmo nó apresente dados e navegações diferentes em cada contexto onde seus objetos são acessados



## Especificação de um Contexto

- Elementos que compõem o contexto
- Informações sobre o contexto em si
- Tipo de navegação entre os elementos
  - Seqüencial
  - Circular
  - Limitada ao índice
  - Combinação da navegação por índice e seqüencial
  - Livre
- Operações aplicáveis aos elementos do contexto
- Usuários e permissões
- Classes em contexto

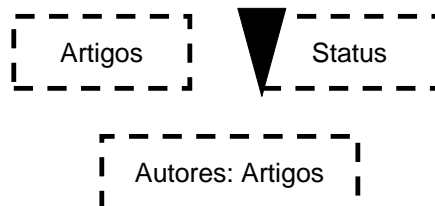
## Cartão de Especificação de Contextos

<b>Contexto:</b> Artigos por Autor	
<b>Tipo:</b> Estático	
<b>Parâmetros:</b> Id_Usuário	
<b>Elementos:</b> A: Artigo ONDE autor = Id_Usuário	
<b>Classes em Contexto:</b> ArtigoPorAutor	
<b>Ordenação:</b> A.autor, ascendente	
<b>Navegação Interna:</b> por índice	
<b>Operações:</b>	
<b>Usuário:</b> autor	<b>Permissão:</b> leitura
<b>Comentários:</b> Todos os artigos submetidos pelo autor	

## Estruturas de Acesso

- Ajudam o usuário final a encontrar a informação desejada
- Podem ser definidos apontando para pontos arbitrários do hiperdocumento, inclusive outros índices
- Podem possuir múltiplos critérios de ordenação
- Podem ser simples ou hierárquicas

◆ Notação:





## Estrutura de Acesso

- É um conjunto de objetos ordenados
- Cada objeto tem pelo menos um elemento reativo (seletor) que aciona um elo para um outro objeto
- Um mesmo objeto pode ter vários seletores que podem ativar a navegação até diferentes destinos:
  - outra estrutura de acesso,
  - um objeto em algum contexto (o próprio objeto ou outro vinculado), etc.



## Projeto de uma Estrutura de Acesso

- Atributos
  - devem permitir distinguir os objetos do conjunto entre si
  - devem ajudar o usuário a achar o objeto que procura;
  - devem ser significativos para a tarefa
- Ordenação
  - deve facilitar a pesquisa;
  - não tem que ser igual à ordenação do contexto alvo (que deve facilitar o processamento que o usuário deseja fazer com o conjunto de objetos);
  - os cenários podem descrever algum critério ou informações sobre a tarefa que ajudem a decidir
- Operações
  - devem ser disponibilizadas na estrutura de acesso quando é útil
    - perceber o conjunto como um todo ou
    - executar a operação sem ter que acessar o objeto

## Estrutura de Acesso - Exemplo



© Adriana P. de Medeiros e Daniel Schwabe, 2001

51

## Especificação da Estrutura de Acesso

- Elementos apresentados na estrutura
- Informações sobre a estrutura em si
- Atributos apresentados
- O destino de cada atributo seletor (índice, elementos acessados e em qual contexto)
- Ordenação dos elementos
- Usuários e Permissões

© Adriana P. de Medeiros e Daniel Schwabe, 2001

52



## Cartão de Estrutura de Acesso



Estrutura de Acesso: Artigos	
<b>Tipo:</b> Simples	
<b>Parâmetros:</b>	
<b>Elementos:</b> A: Artigo $\in$ Artigos por Autor e co-Autor	
<b>Atributos</b>	<b>Destino</b>
A.título A.autor	Ctx Artigos por Autor + co-Autor
<b>Ordenação:</b>	
<b>Usuário:</b> autor, co-autor	<b>Permissão:</b> leitura
<b>Comentários:</b>	

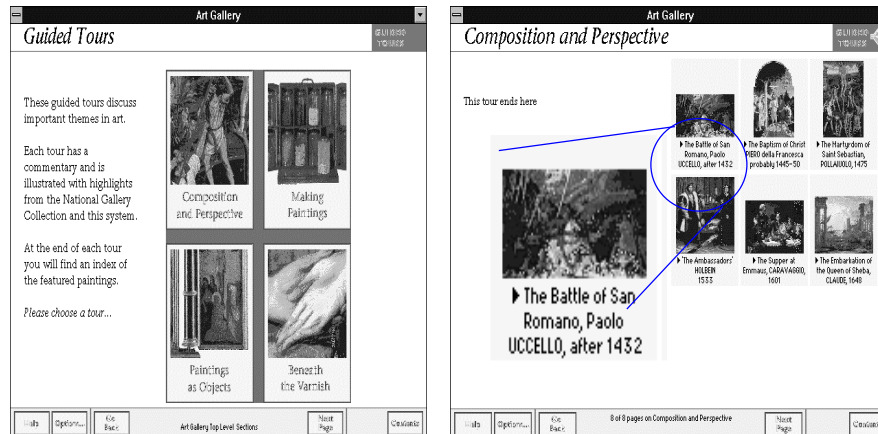


## Visitas Guiadas



- São utilizadas para orientar o leitor em sequências pré-determinadas
- Podem opcionalmente permitir “desvios”
- “Contam uma história”
- Uma visita guiada é caracterizada por
  - ponto de entrada
  - ponto de saída
  - possibilidades de navegação a cada passo
- Os elementos de uma visita guiada podem ser definidos:
  - enumeração
  - “queries” ou “recuperação de informações”

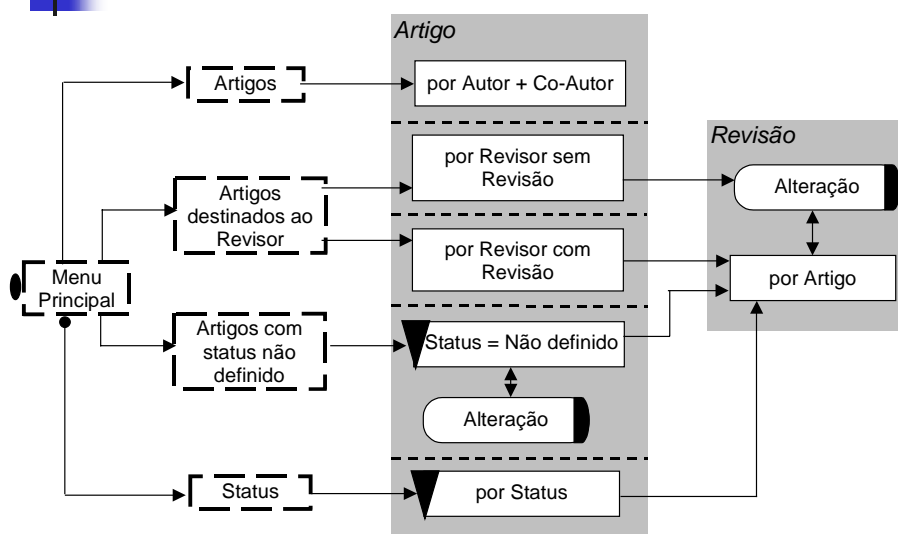
## Visita Guiada - Exemplo



© Adriana P. de Medeiros e Daniel Schwabe, 2001

55

## Esquema de Contextos de Navegação



© Adriana P. de Medeiros e Daniel Schwabe, 2001

56



## Projeto da Interface Abstrata

- Define os objetos que serão perceptíveis para o usuário, suas propriedades e transformações
- Os objetos de navegação são mapeados em objetos perceptíveis
- A navegação se passa num contexto de percepção, que contém um conjunto de objetos perceptíveis, e um “foco de atenção”
- Objetos perceptíveis podem conter objetos ativáveis, que correspondem às âncoras
- A ativação de um objeto ativável causa uma transformação no contexto de percepção



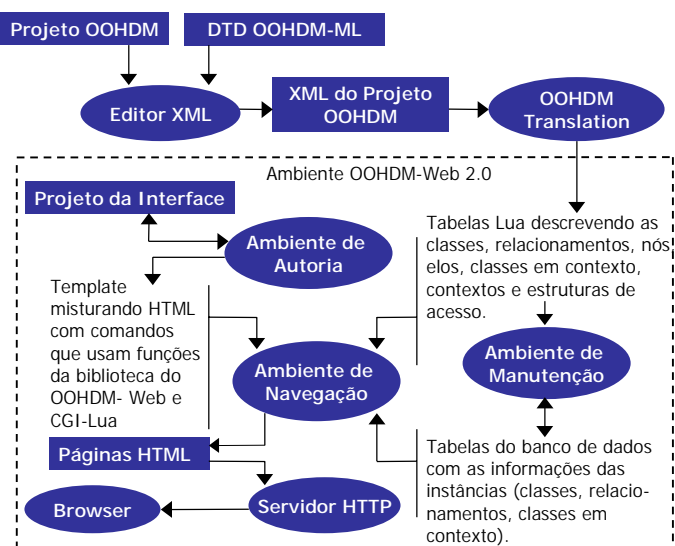
## Especificação do Projeto da Interface

- Abstract Data Views (ADVs)
  - são objetos de interface usados para especificar a aparência e interface dos objetos da aplicação
- Abstract Data Objects (ADOs)
  - são objetos que modelam a interface mas não suportam eventos externos, somente interagem com as estruturas de dados e de controle da aplicação
- Diagramas de Configuração
  - são usados para representar os relacionamentos entre os objetos de interface e os objetos navegacionais
- ADVCharts
  - apresentam as transformações ocorridas nos ADVs, responsáveis por mudanças na interface com o usuário e nos objetos navegacionais

## Implementação

- Atividade responsável pela tradução do projeto da aplicação para um ambiente de implementação
- Ambientes
  - OOHDM-Web
    - ambiente baseado na linguagem Lua
  - OOHDM-XWeb
    - composto pela folha de estilo OOHDM-Translation e pelo sub-ambiente OOHDM-Web 2.0
  - OOHDM-Java
    - framework para implementação de aplicações hipermídia modeladas com o OOHDM

## Arquitetura do Ambiente OOHDM-XWeb





## Referências

---



<http://www.telemidia.puc-rio.br/oohdm>

Daniel Schwabe

[schwabe@inf.puc-rio.br](mailto:schwabe@inf.puc-rio.br)

Adriana Pereira de Medeiros

[adri@inf.puc-rio.br](mailto:adri@inf.puc-rio.br)