

GERAÇÃO AUTOMÁTICA DE APLICAÇÕES WEB

Prof^a Dr^a Itana Maria de Souza Gimenes &
Bruno Miguel Nogueira de Souza

Agenda



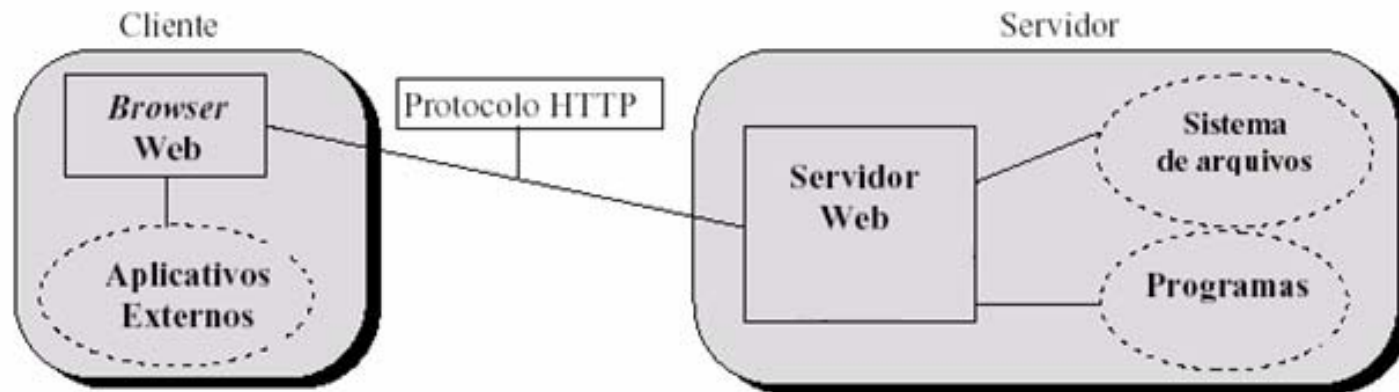
- Primeira Aula 22/11/2008
 - Introdução a aplicações Web
 - Características de Aplicações Web
 - Arquitetura Web
 - Métodos de Desenvolvimento de Aplicações Web
 - Geração de código
 - Introdução a WebML
 - Intervalo – Almoço
 - Desenvolvimento com WebML (laboratório)

Agenda



- Segunda Aula 29/11/2008
 - ▣ Desenvolvimento de uma aplicação com WebML e WebRatio
 - Modelagem
 - Geração de código
 - ▣ Outras tecnologias de geração de código
 - ▣ Arquitetura orientada a serviços

Introdução – Visão geral da Web



Introdução – Padrões Web



- Transferir informações – HTTP (protocolo *stateless*)
- Descrever a formatação das informações - HTML
- Localizar a informação – URL.

Tipos de aplicações de software

- Software básico
- Software para sistema em tempo real
- Software comercial
- Software para engenharia e aplicações científicas
- Software embarcado (ex. microwave)
- Software para computadores pessoais
- Software baseado em inteligência artificial
- Software de entretenimento
 - O que mudou com a Web?

Contexto das aplicações Web

- Ubiquidade da Internet:
 - Está em todos os lugares;
 - Facilidade de conexão entre empresas.
- Interface com o Usuário - ambiente navegável.



Contexto das aplicações Web



- Dessa forma, as aplicações Web são similares a aplicações convencionais com interface Web Browser e comunicação através de protocolos específicos.
- Exemplos: Imposto de renda, Compra de livros, ... Google Docs, Orkut, Blogs.

Aplicações Web - Características



- são sistemas de informação baseados em documentos dinâmicos criados, compartilhados, navegados, atualizados e acessados por vários tipos de usuários;
- utilizam documentos compostos de hierarquias de elementos estáticos (ex.: textos e gráficos) e dinâmicos (ex.: applets *Java* e formulários *HTML*);

Aplicações Web - Características

- Utilizam dados armazenados nos próprios documentos ou em banco de dados, possivelmente distribuídos;
- código executável localmente nos documentos e em servidores.



As aplicações convencionais ganharam uma nova interface gráfica e uma nova dinâmica de execução.

Aplicações Web - Requisitos

- **Qualidade**: utilização de padrões de qualidade, tais como ISO 9000 e o *Capability Maturity Model (CMM)*;
- utilização de técnicas de **gerenciamento de projeto**, como as sugeridas pelo *Project Management Institute 5*;
- **desenvolvimento com componentes, modularizados de forma que possam ser substituídos independentemente do resto do sistema, facilitando a implementação, as atualizações e evoluções**;
- inclusão de um novo profissional na equipe de desenvolvimento, o **Web designer**, que cuida do desenvolvimento de Páginas e da interatividade.

Aplicações Web - Estrutura



- Uma Aplicação Web pode ser dividido em duas partes principais
 - ▣ *Back-end*: máquina computacional responsável pelas funções da aplicação (regras de negócio) e da administração da informação (banco de dados).
 - ▣ *Front-end*: é a interface do usuário - permite aos usuários da aplicação ver os dados e ativar as funcionalidades da aplicação.

Aplicações Web – Front-End



- O *front-end* usualmente reside em uma máquina separada da do *back-end*, essas máquinas são conectadas por alguma infra-estrutura de comunicação, uma rede local por exemplo.
- Os usuários acessam as funcionalidades da aplicação Web usando o *front-end*. Dependendo da segmentação da parte lógica da aplicação, cada operação executada pelo usuário causará uma comunicação entre o *front-end* e o *back-end* requerendo vários serviços.

Aplicações Web – Front-End

- O *Front-end* sempre gerencia a apresentação do usuário e captura as entradas feitas por ele.
 - ▣ *Fat Front-end* (Cliente “Gordo”)- implementa parte da lógica da aplicação. Neste caso, o *back-end* funciona usualmente como um servidor de dados para o *front-end*.
 - ▣ *Lean Front-end* (Cliente “magro”) - O *front-end* recebe a apresentação direto do *back-end*, e envia as entradas do usuário para o *back-end*. O *back-end* então executa algumas funções da aplicação e cria uma nova apresentação para ser usada pelo *front-end*.

Arquiteturas de Aplicações Web



- São arquiteturas do tipo Cliente/Servidor:
 - ▣ Parte do sistema responsável pela lógica da aplicação fica localizada no servidor
 - ▣ Cliente permite o acesso à lógica da aplicação.
- Mais recentemente podemos falar em arquiteturas baseadas em serviços
- Em geral, a aplicação localizada no servidor é dividida em várias camadas (Multi-Camada)

Arquiteturas de Aplicações Web – Cliente / Servidor

- Cliente
 - ▣ Exibir e solicitar documentos, ex. HTML.
- Servidor
 - ▣ Atender pedidos dos clientes Web.
 - ▣ Disparar aplicações externas, ex. execução de programas via CGI.
- Comunicação
 - ▣ HTTP
 - ▣ URLs

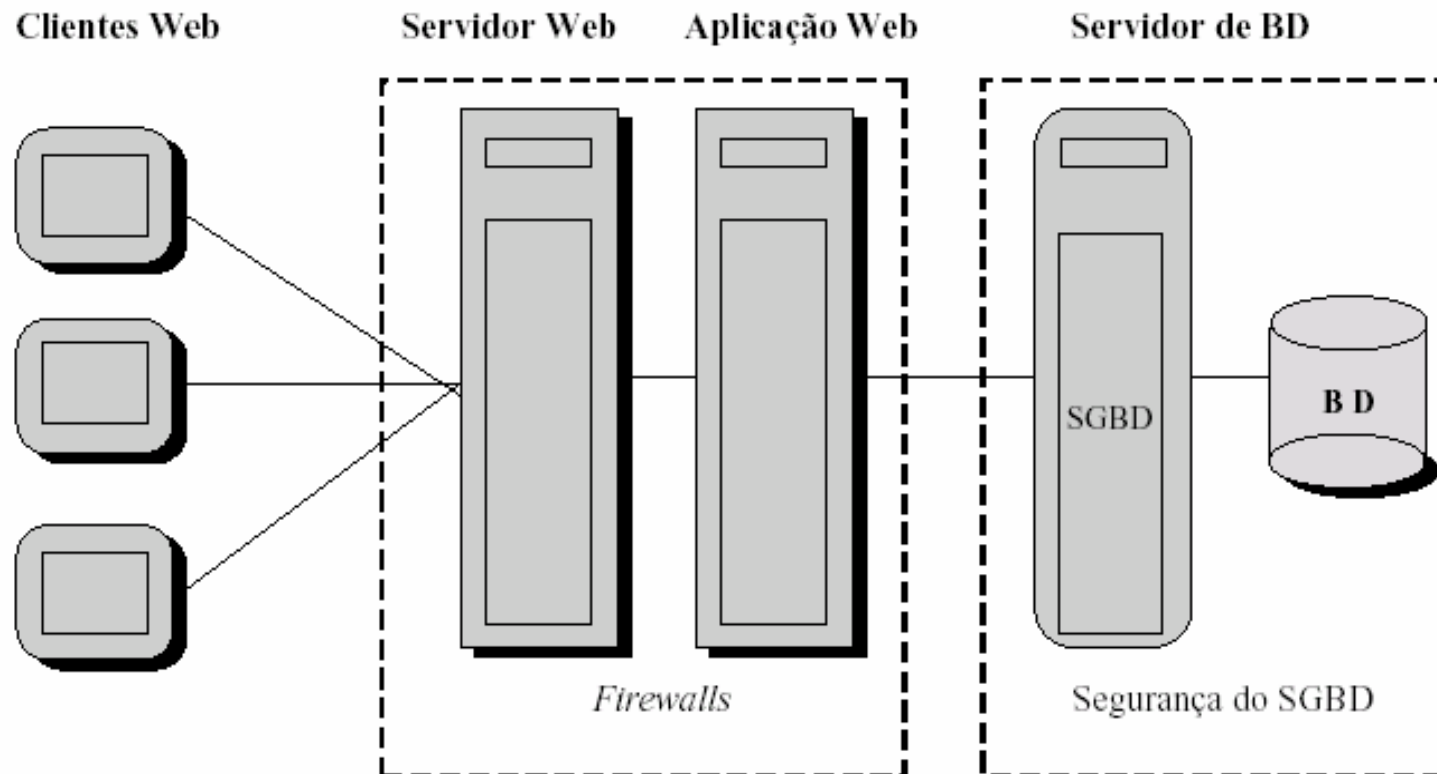


Arquiteturas no lado do Servidor Web

Divisão em camadas (three-tier)

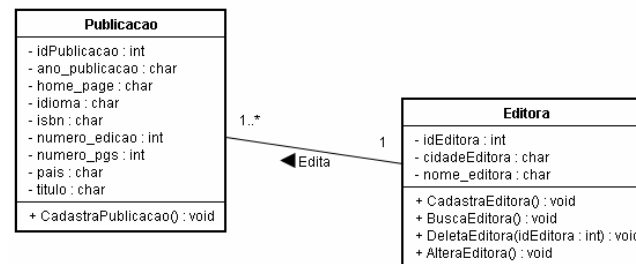
- apresentação: responsável pela lógica da apresentação e controle da interação entre o usuário e o computador.
- lógica: esta camada é composta pelo servidor Web e pelo software que acessa a base de dados. Contém a lógica da aplicação, responsável pelas decisões, cálculos e operações que a aplicação deve realizar, mecanismos de segurança para acesso à aplicação, e controle de acesso multiusuário à aplicação. Nesta camada podem estar presentes os *firewalls* responsáveis pelos mecanismos de segurança.
- dados: responsável pelos dados, metadados e regras de integridade.

Arquiteturas no lado do Servidor Web



Arquiteturas de Aplicações Web – Arquitetura MVC

- A Arquitetura MVC foi proposta nos anos 70 por Smalltalk
- A arquitetura MVC é constituída de três camadas:
 - ▣ (M) **Model**, ou camada do modelo, em termos de Orientação a Objetos é onde fica o conjunto de classes da aplicação, cada aplicação possui um único modelo.



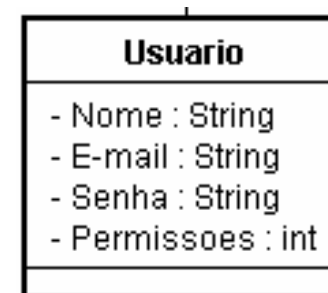
Arquiteturas de Aplicações Web – Arquitetura MVC

- (V) View, ou camada de visão, é a camada que possui uma ou mais interfaces com a camada do Modelo, sendo esta camada a responsável por mostrar as informações ao usuário final através das GUIs.

E-mail :

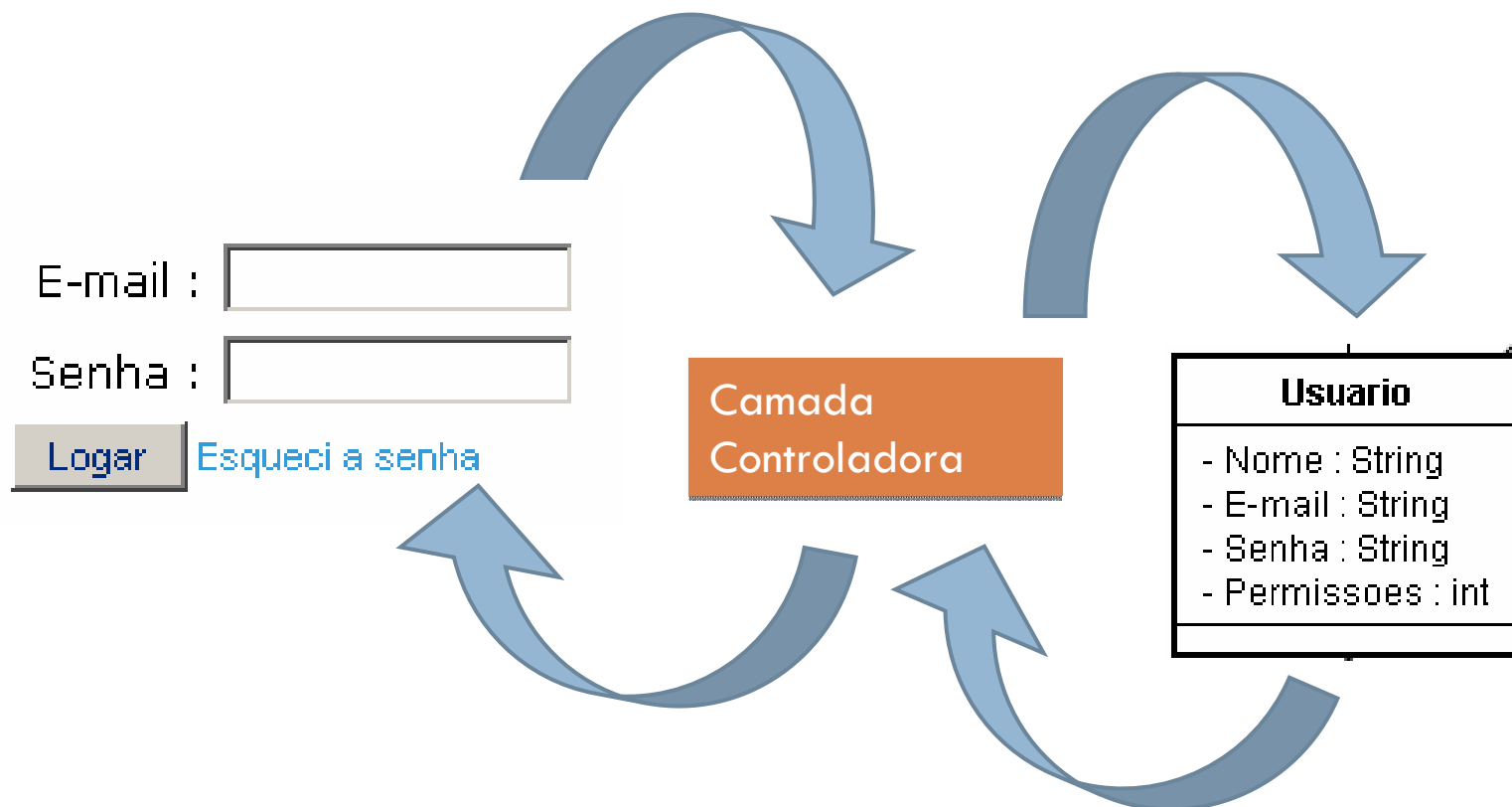
Senha :

[Esqueci a senha](#)



Arquiteturas de Aplicações Web – Arquitetura MVC

- (C) Controller, ou camada controladora, é a camada que efetua o controle entre as camadas de modelo e visão.



Arquiteturas de Aplicações Web - Tecnologias de desenvolvimento

- Banco de Dados:
 - PostgreSQL;
 - MySQL;
 - SQL Server;
 - Oracle;
 - Microsoft Access;
 - Entre outros...
- Linguagens de Programação:
 - HTML;
 - Java Server Pages (JSP);
 - JavaScripts;
 - ASP e ASP.NET;
 - PHP;

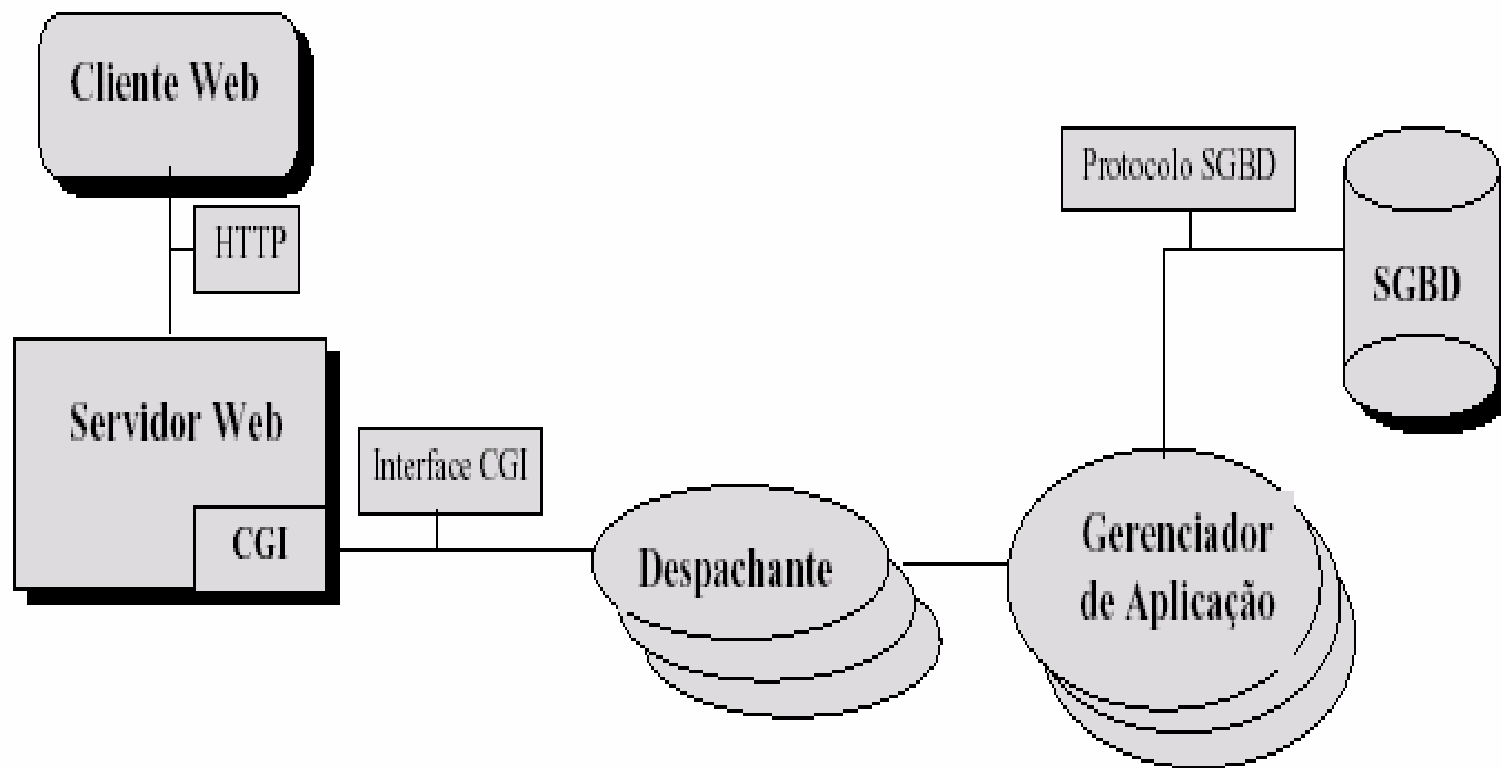
Arquiteturas de Aplicações Web - Tecnologias de desenvolvimento



- Java Web Frameworks – Struts, Java Server Faces (JSF), Spring, Hybernate;
- Ruby on Rails;
- PHP: MIOLO Framework, PRADO, Cake PHP;

Outras arquiteturas de aplicações Web

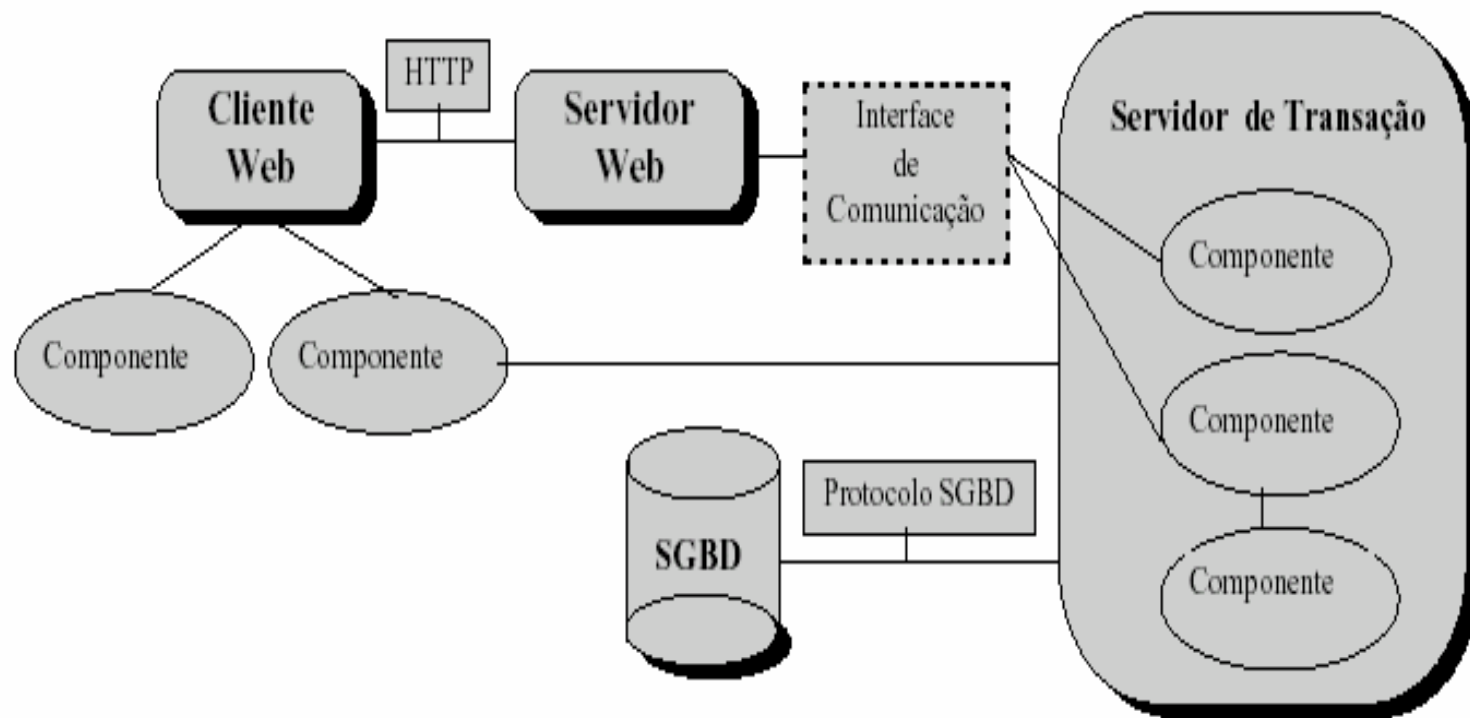
Web



Vão desde arquiteturas que usam a interface padrão CGI – Gerenciador de Aplicação CGI ... até

Outras arquiteturas de aplicações Web

Web



Arquitetura Servidor de Transações

Desenvolvimento de Aplicações Web.



- Engenharia de Software para aplicações Web;
- Web é vista como um novo domínio de sistemas de informação – Criação de modelos de apresentação e navegação .
- Baseado no desenvolvimento convencional (UML)

Desenvolvimento de Aplicações Web.

- Demanda: Sistemas de informação de grande porte, freqüentemente distribuído e contendo componentes altamente dinâmicos e interativos.
- **Process-intensive** x **Data-intensive**

Desenvolvimento de Aplicações Web – Aspectos Importantes

- **Modelagem de Documentos:** hipertextos com componentes estáticos e dinâmicos.
- **Modelagem de Banco de Dados:** além de documentos, os dados podem ser armazenados em bancos de dados por motivos de desempenho e segurança.

Desenvolvimento de Aplicações Web – Aspectos Importantes

- **Modelagem de Transações:** o usuário utiliza as funcionalidades do sistema através da interação com seus documentos, que disparam as transações apropriadas, portanto, para modelar completamente uma transação, é necessário ter em mente os documentos envolvidos, caminhos de navegação, categorias de usuários, dados de entrada e resultados.

Desenvolvimento de Aplicações Web – Aspectos Importantes



- **Modelagem de Navegação:** muitos documentos podem estar envolvidos em uma transação, e a modelagem da navegação deve especificar os caminhos possíveis ao usuário, as funções executadas em cada documentos, e como a base de dados é acessada.

Desenvolvimento de Aplicações Web – Aspectos Importantes

- **Modelagem de Aplicação:** muitas aplicações podem compor um sistema de informação para Web, como *applets Java* e programas *CGI*. A modelagem deve especificar suas funcionalidades e a localização de cada código executável.

Métodos de desenvolvimento de Aplicações WEB.



- Existem métodos de desenvolvimento específicos para desenvolver aplicações WEB.
 - WebML (Data-intensive);
 - OOHDM;
 - OOWS;
 - UWE;
 - WAE.

Métodos de desenvolvimento de Aplicações WEB.



- Em geral, os métodos de desenvolvimento de aplicações WEB possuem aspectos similares:
 - Concepção de um modelo conceitual;
 - Construção de modelos navegacionais ;
 - Construção de modelos de apresentação;
 - Implementação da solução.

Aplicações WEB – Modelo Conceitual



- Consiste em capturar os requisitos e representar os conceitos básicos do sistema;
 - ▣ Diagramas de caso de Uso;
 - ▣ Diagramas de Classe;
 - ▣ Diagramas de Entidade Relacionamento;
 - ▣ Diagramas de Estados;
 - ▣ Diagramas de Sequência.

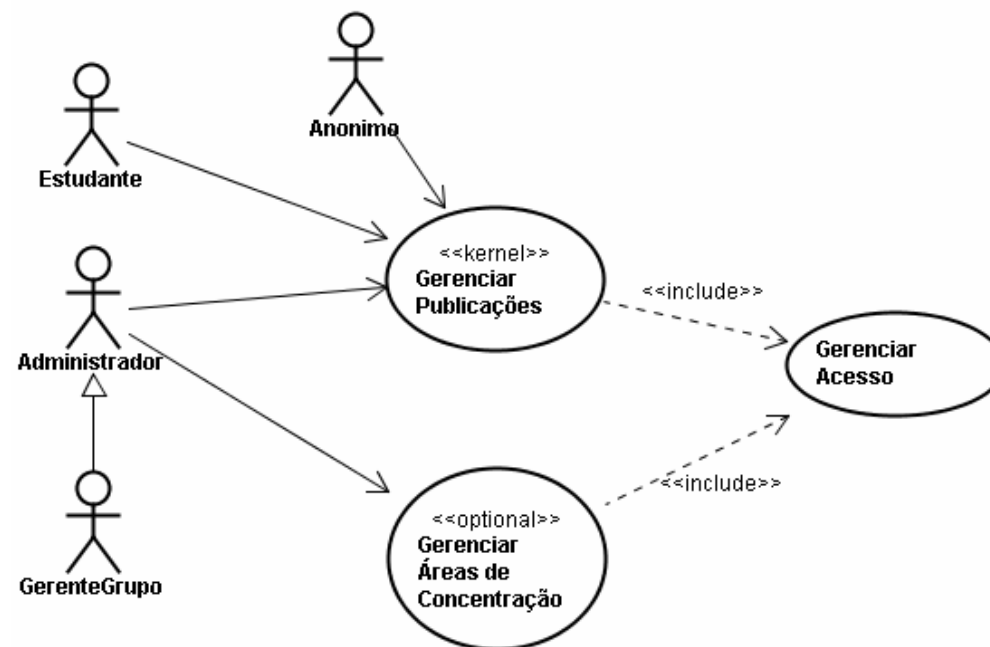
Aplicações WEB – Modelo Conceitual



- ❑ Para ilustrar o processo de desenvolvimento do modelo conceitual de aplicações Web, será utilizado como exemplo um Sistema de Gerenciamento de publicações.
- ❑ Será abordado de uma forma genérica sem a referência a um método de desenvolvimento Web específico.

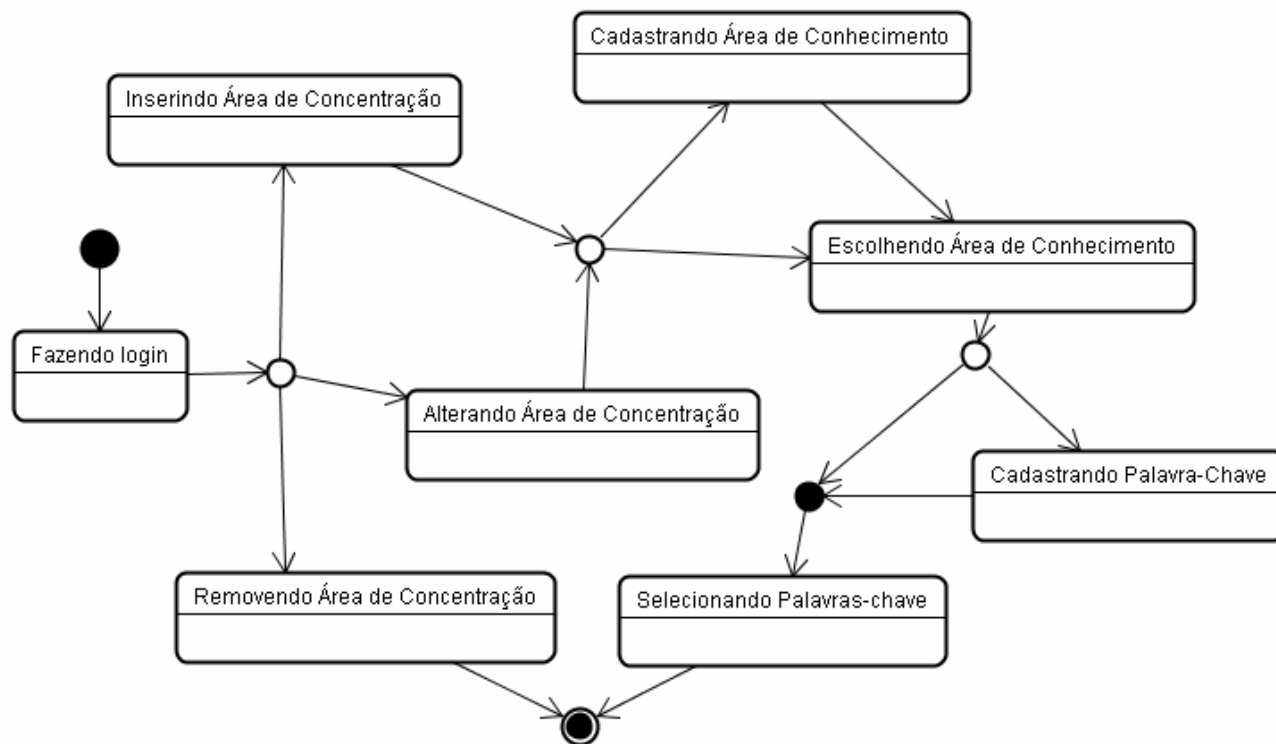
Aplicações WEB – Modelo conceitual

- Levantamento de Requisitos: Diagramas de Casos de Uso.

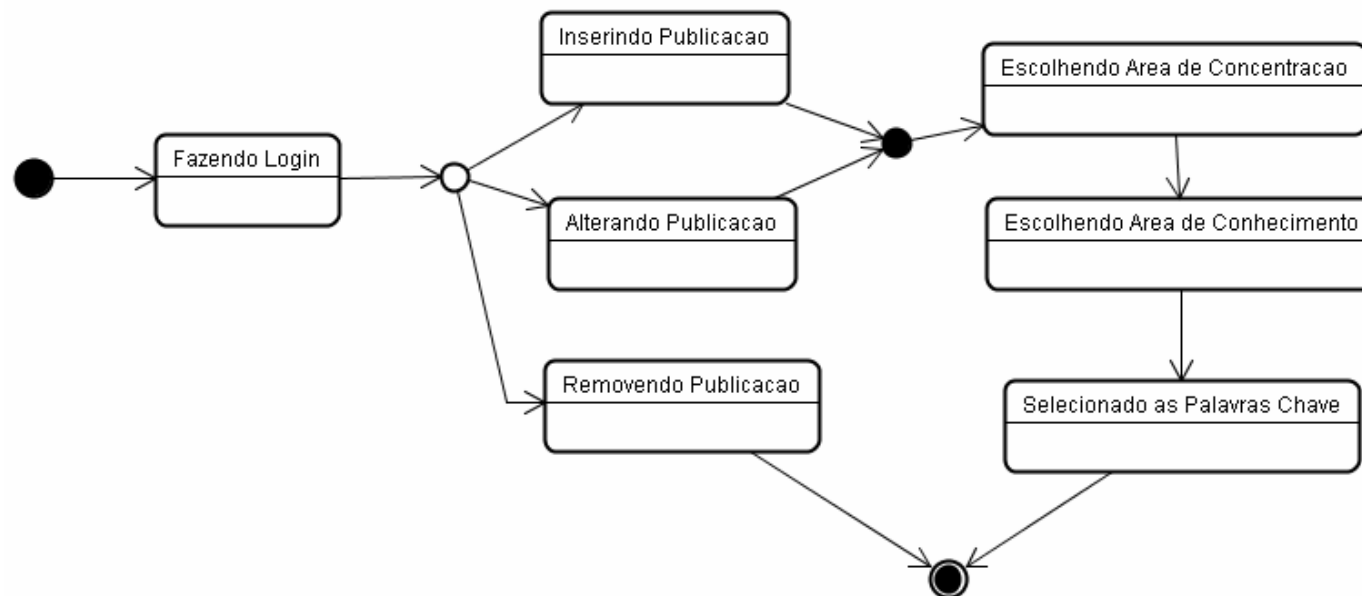


Aplicações WEB – Modelo conceitual

- Modelo de Estados – Representa a Interatividade do sistema – um modelo de estados para cada caso de uso. (gerenciar Áreas de Conhecimento)

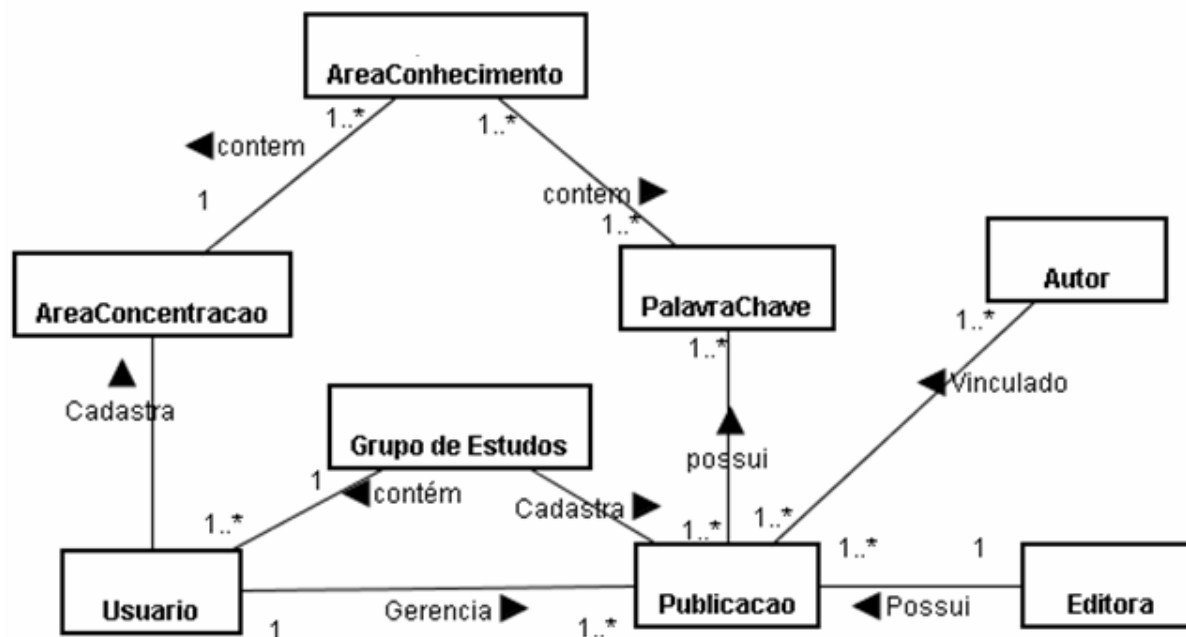


□ Modelo de Estados – Gerenciar Publicações



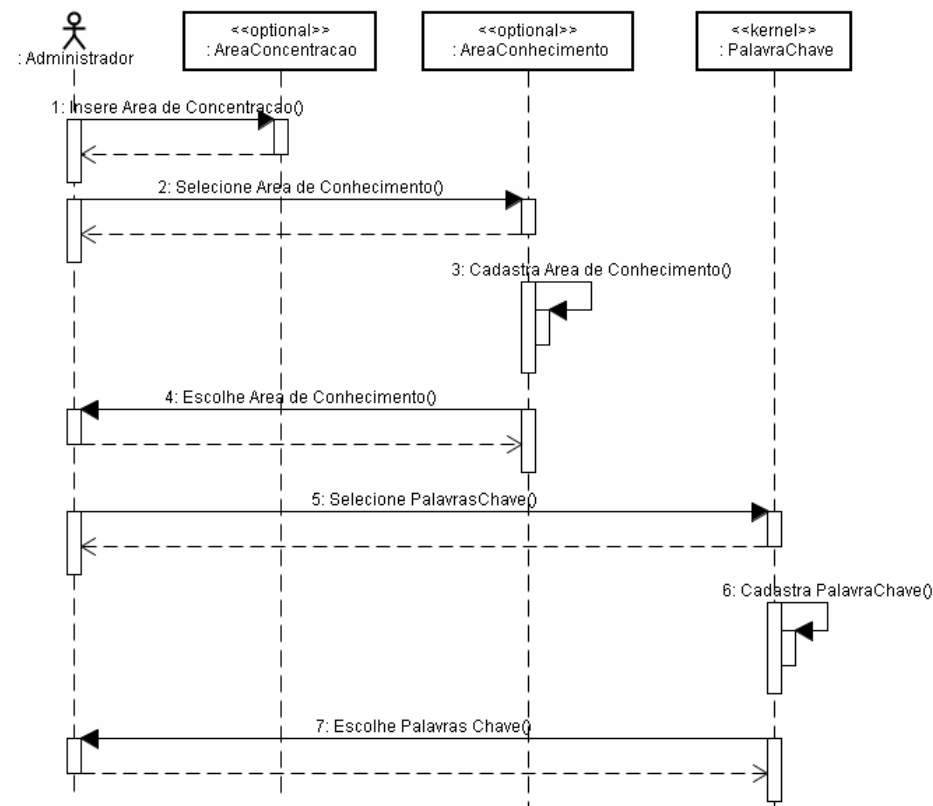
Aplicações WEB – Modelo conceitual

- Modelo Estático – Ele representa de uma forma geral a interação entre as entidades (futuramente - classes) do sistema



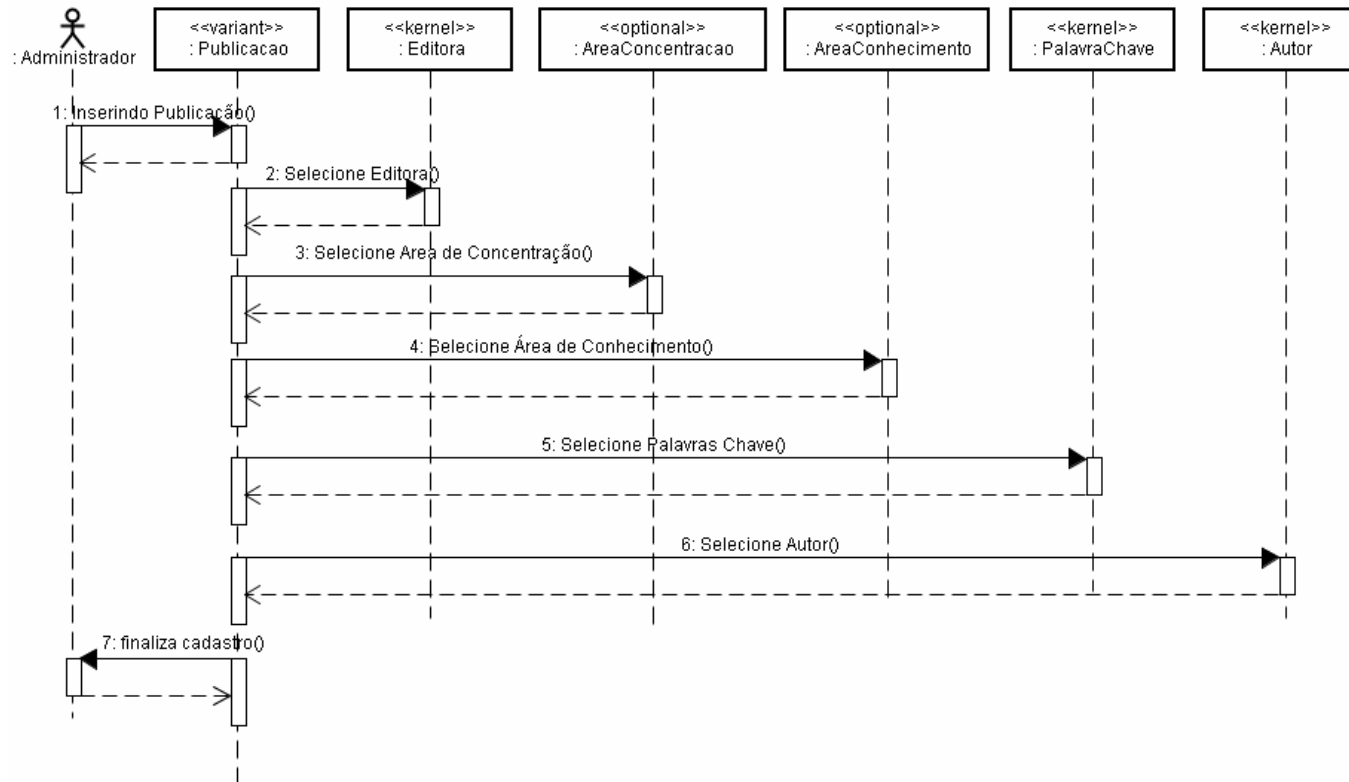
Aplicações WEB – Modelo conceitual

- Diagrama de sequência - para representar a interação entre as classes do sistema. (um para cada caso de uso)



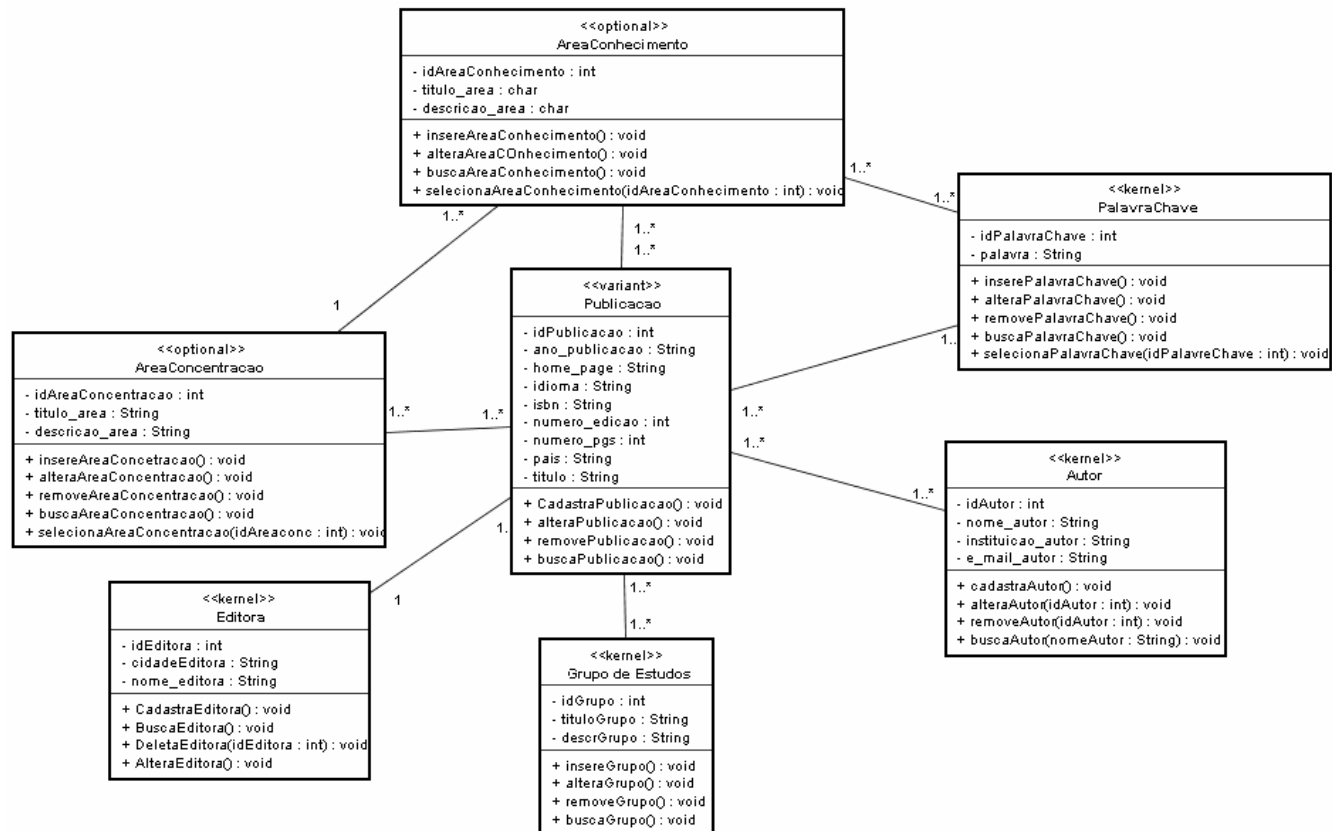
Aplicações WEB – Modelo conceitual

□ Gerenciamento de publicações



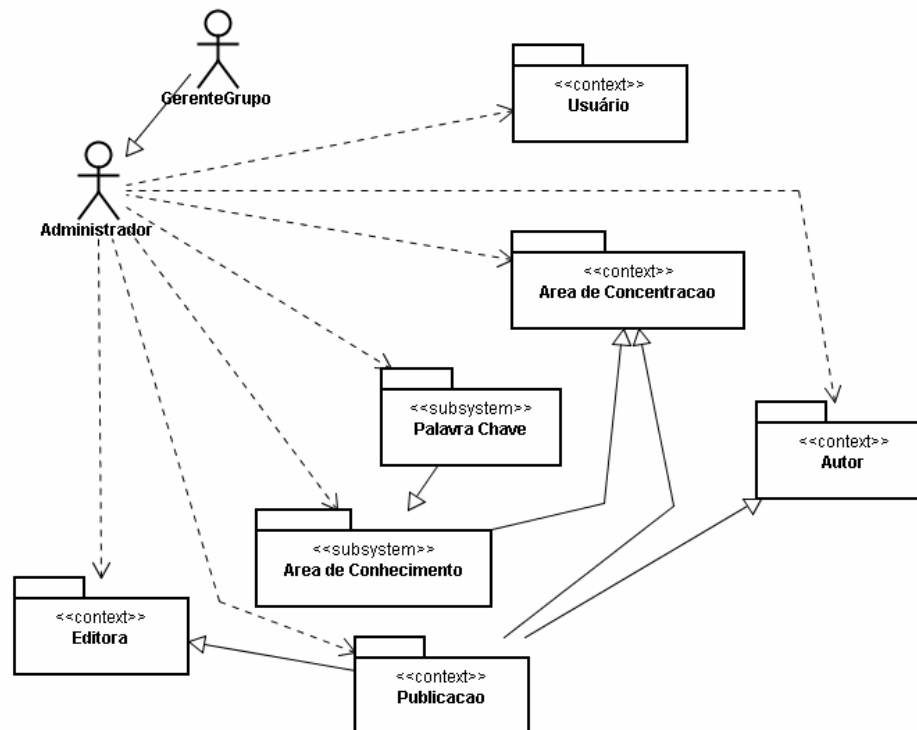
Aplicações WEB – Modelo conceitual

- Realizando um mapeamento das ações descritas nos modelos criados anteriormente, extrai-se os atributos e métodos para cada classe.



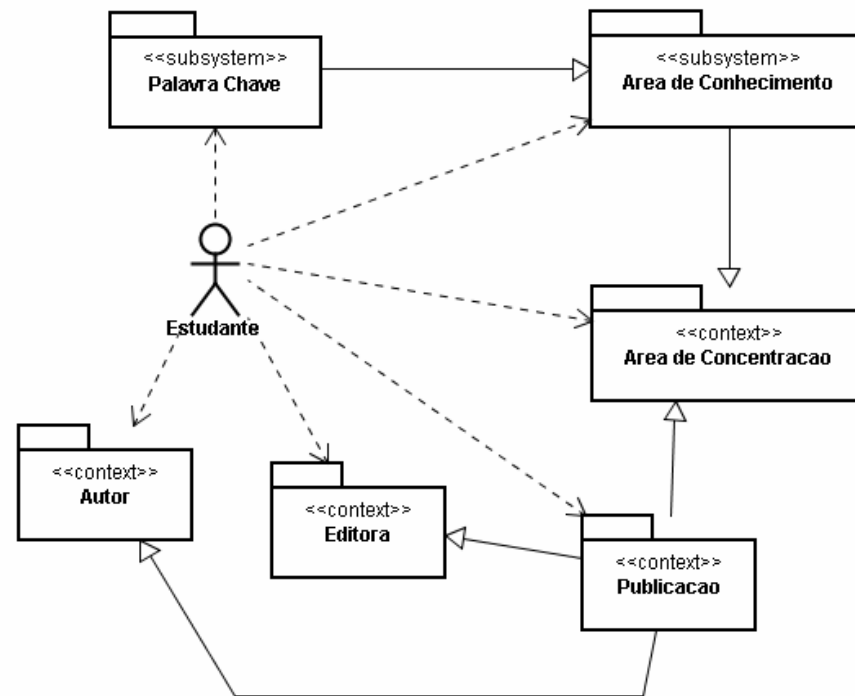
Aplicações WEB – Modelo de Navegação

- O Modelo de Navegação deverá representar os objetos a serem acessados por cada tipo de usuário, assim como o relacionamento entre os objetos. (Usuários Administrador e GerenteGrupo)



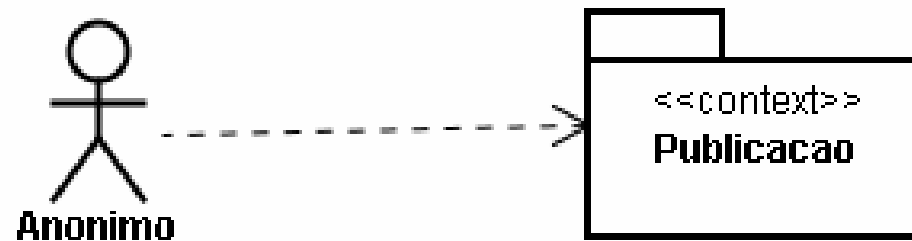
Aplicações WEB – Modelo de Navegação

- Modelo navegacional para o usuário Estudante.



Aplicações WEB – Modelo de Navegação

- Modelo Navegacional para o usuário Anônimo



Aplicações WEB – Modelo de Apresentação

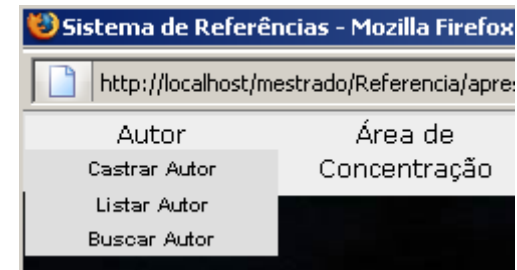
- ❑ O modelo de apresentação é dependente do modelo navegacional, pois para cada contexto será necessário a criação de um modelo de apresentação.
- ❑ O Diagrama de Estados, tem um importante papel na apresentação, pois, ele mostrará a interação entre os elementos, onde é possível identificar quais itens deverão ser incluídos em cada contexto.

Aplicações WEB – Modelo de Apresentação

- Exemplo – Árvore de Ação Hierárquica (contexto - Autor):

Nível 2	Nível 3	Descrição
Serviços UI	Validação de campos	Gerenciador de Formulários.
	Lista de seleção	Busca Autor, Lista de autores
Instância UI	Conjunto de Exibição	nomeAutor
	Ações	Cadastrar Autor, Alterar Autor, Excluir Autor, Listar Autor.
	Navegação	Publicação, Area de Conhecimento, Area de Concentração, Palavra-chave, Autor, Editora.
População UI	Filtro	Autor por publicação, autor por área de concentração, autor por área de conhecimento.
	Conjunto de exibição	QueryAutor
	Ações	Cadastrar Autor, Alterar Autor, Excluir Autor, Listar Autor.
	Navegação	Publicação, Area de Conhecimento, Area de Concentração, Palavra-chave, Autor, Editora.
Mestre-específico UI	Mestre	nomeAutor
	Específico	enderecoAutor, instituicaoAutor

Aplicações WEB – Modelo de Apresentação



Cadastro de Autor

Cadastro de Autor

Nome do Autor:

Modo de Referência ao Autor:

Instituição:

Aplicações WEB - Implementação



- A partir dos modelos desenvolvidos é possível partir para o desenvolvimento da solução.
- A partir das classes retiram-se as entidades que farão parte do banco de dados e os métodos que deverão ser criados para satisfazer a lógica do negócio.
- A partir do modelo de navegação, extrai-se os usuários e suas atribuições, ilustrando quais serão os contextos acessados por cada um.

Aplicações WEB - Implementação



- ❑ O Modelo de apresentação, ilustrará as formas como os contextos exibirão seus conteúdos, assim como, quais serão os dados e a forma como serão apresentados ao usuário.
- ❑ O objetivo das ferramentas que possibilitam a geração automática de código é apoiar o processo de implementação, portanto a modelagem conceitual é de extrema importância para o desenvolvimento de qualquer tipo de aplicação.

Geração de código

- Estruturas de reutilização de software:
 - ▣ Padrões de projeto – descrição de uma solução de software recorrente (análise, projeto, idioma)
 - ▣ Frameworks - uma arquitetura de software semi-definida que consiste de um conjunto de unidades individuais e de interconexões entre elas, de tal forma a criar uma infra-estrutura de apoio pré-fabricada para o desenvolvimento de aplicações de um ou mais domínios específicos.
 - ▣ Componentes - Um componente é definido como uma unidade de software independente, que encapsula dentro de si seu projeto e implementação, e oferece interfaces bem definidas para o meio externo.

Geração de código



- Técnicas que visam fornecer meios para o desenvolvedor especificar a aplicação em uma linguagem de alto nível e mecanismos que permitem transformar esta linguagem em código.
- Técnicas envolvidas
 - ▣ DSL (Domain Specific Languages) ex. WebML
 - ▣ MDA – Arquiteturas Dirigidas por Modelos
 - ▣ Geradores de aplicação
 - ▣ Linha de produto de software

Domain Specific Languages (DSL)

- São linguagens concebidas para especificar aplicações em um domínio específico:
- São executáveis ou permitem automação.
- São textuais ou gráficas
- São “similares” ao que antigamente se chamava de linguagens de 4a. Geração
- Exemplos:
 - ▣ WebML
 - ▣ SQL
 - ▣ RPG – geradores de relatório

MDA – Arquiteturas Dirigidas por Modelos

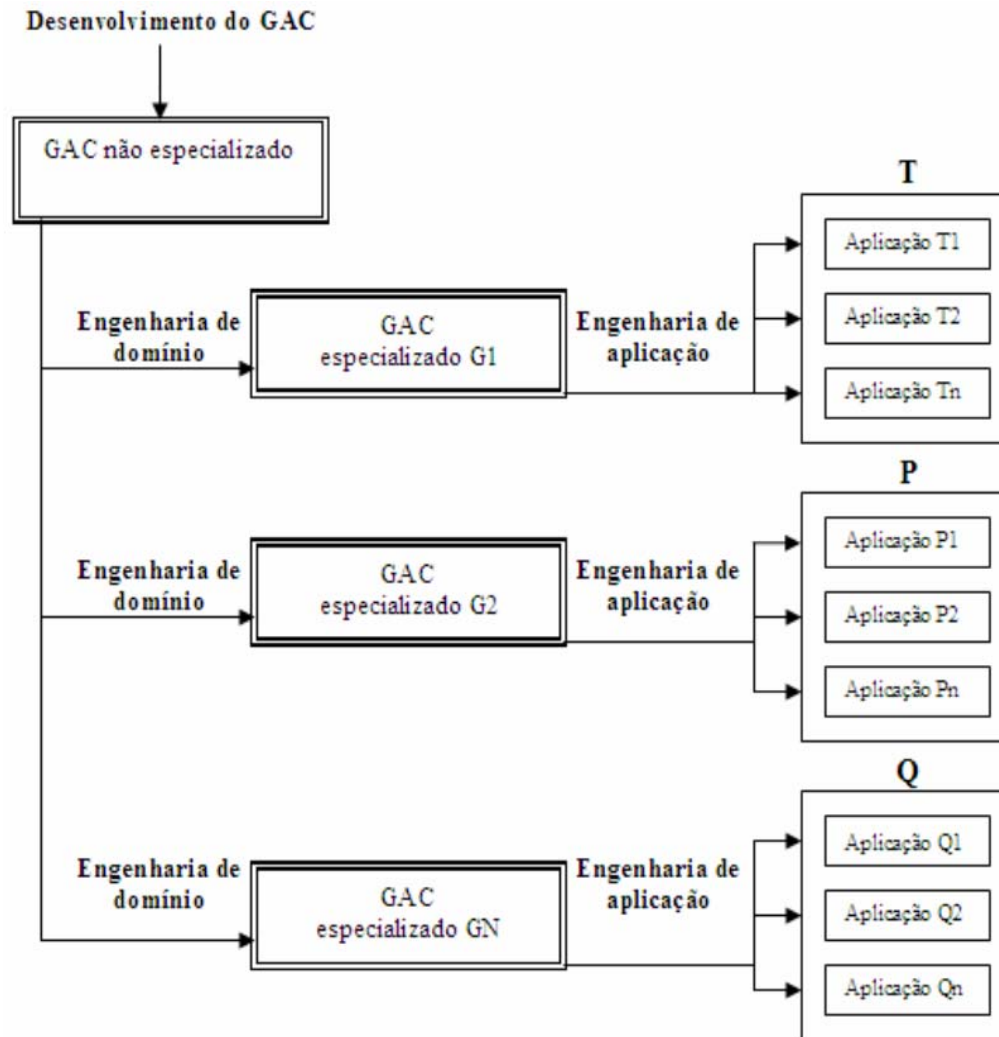
- Abordagem orientada a negócio para a construção de sistemas de software em que modelos são refinados a partir das necessidades do negócio até a sua realização em soluções de software.
- Tem como objetivo especificar o sistema independentemente da plataforma.
- Define uma arquitetura para especificar os modelos em níveis para transformá-los automaticamente.
 - ▣ CIM - *Computation Independent modelo* – captura o contexto e regras de negócio;
 - ▣ PIM – *Platform Independent Model* – serviços do sistema independente de tecnologia;
 - ▣ PSM – *Platform Specific Model* – realização dos serviços em uma plataforma;
 - ▣ Código

Geradores de aplicação



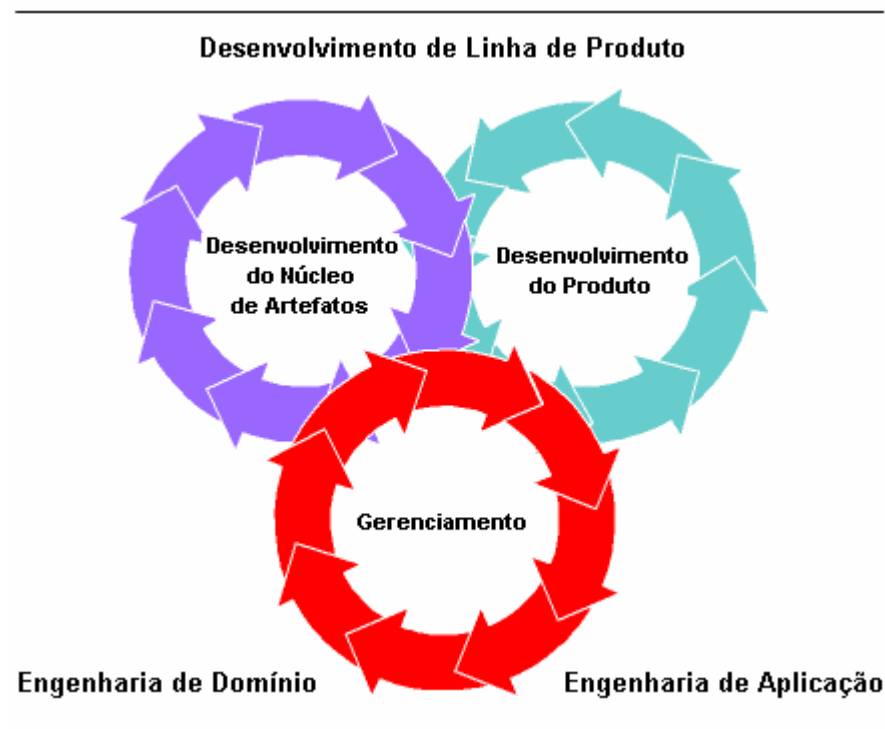
- Pode ter várias conotações: compiladores, pré-processadores, meta-funções, *wizards*, geradores de códigos.
- Algumas ferramentas de desenvolvimento de software geram código (ex. Jude, Poseidon)
- Tipos de geradores
 - ▣ Domínio Único (*lex*, *yacc*)
 - ▣ Múltiplos domínios (*Draco*, *WebRatio*)

Geradores de aplicação configuráveis



Linha de Produto de Software (LPS)

- Uma LPS representa um conjunto de sistemas que compartilham características comuns e gerenciáveis que satisfazem as necessidades de um segmento particular do mercado ou de uma missão. Esse conjunto de sistemas é também chamado de família de produtos.



Conhecendo o WEBML

- O UML para Web – é uma linguagem visual para especificação de conteúdo, composição e características de navegação de aplicações hipertexto.
- Propõe um desenvolvimento em alto nível, sem a necessidade de compreensão de uma arquitetura detalhada.
- Possui uma representação gráfica intuitiva, que pode ter o apoio de ferramentas CASE e ser repassada a membros do grupo que não possuam conhecimento técnico sobre o desenvolvimento (por exemplo: produtores de interface e os produtores de conteúdo).

Conhecendo WebML

- Consiste de quatro perspectivas: o Modelo Estrutural, o Modelo de Hipertexto, Modelo de Apresentação e um Modelo de Personificação.

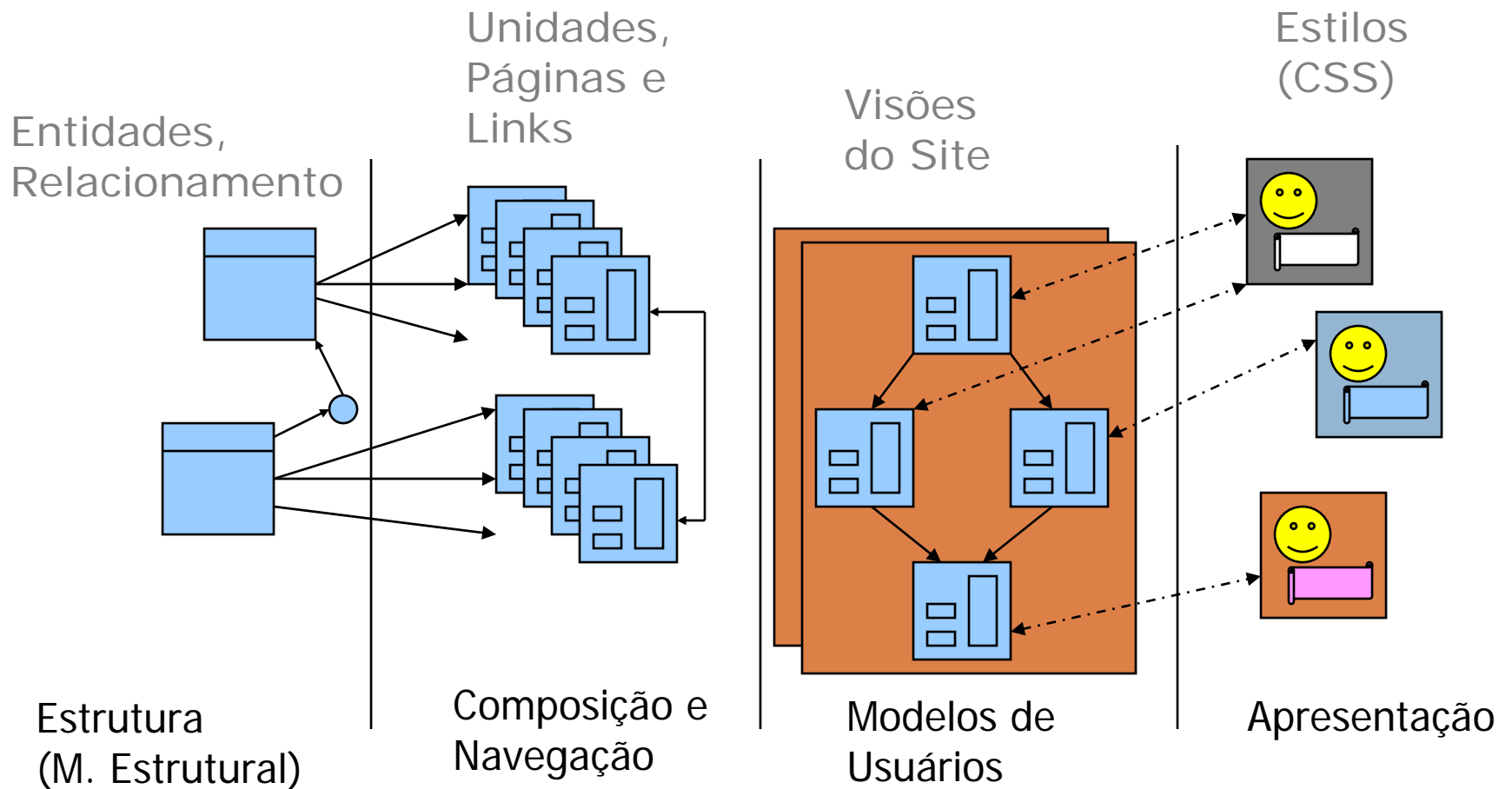
 DSL, MDA, Gerador de aplicação
(WebRatio)

Estrutura do WebML



- Modelo Estrutural;
- O modelo de hipertexto;
 - ▣ Modelo de composição;
 - ▣ Modelo de navegação;
- Modelo de Apresentação;
- Modelo de Adaptação.

WebML – Estrutura Básica



WEBML – Modelo Estrutural



- Modelo E/R, o ODMG, Modelo Orientado a Objetos e diagramas de classe UML.
- Para expressões redundantes e informações calculadas, o modelo estrutural também oferece uma linguagem simplificada, linguagem baseada em “queries” OQL, onde por ela é possível especificar informações derivadas.

WEBML – Modelo Hipertexto

- O Modelo hipertexto descreve um ou mais hipertextos que podem ser publicados no site. Cada hipertexto define uma chamada a um site. A descrição da interface de um site consiste em dois sub-modelos:
 - ▣ Modelo de Composição - especifica quais páginas compõem um hipertexto e quais os conteúdos que constituem uma página;
 - ▣ Modelo de Navegação - expressa o relacionamento entre os conteúdos;

WEBML – Modelo de Apresentação

- Expressa o layout e a aparência gráfica das páginas;
- As especificações da Apresentação podem ser tanto específicas quanto genéricas;
- De uma maneira geral, eles ditam a apresentação de uma página específica incluindo referências explícitas para o conteúdo da página (por exemplo, o layout e a aparência gráfica do título e da capa de um disco).

WEBML – Modelo de Apresentação



- A Personalização, consiste em adaptar cada componente estruturado a partir dos outros modelos.

WEBML – Modelo de Adaptação



- Atribuição de níveis de acesso aos usuários do sistema, podendo assim restringir ou autorizar o acesso a determinados conteúdos.

WebML - Site

□ www.webml.org;



The screenshot shows the homepage of the WebML.org website. At the top left is the WebML.org logo, which consists of three stylized human figures in orange and green. To the right of the logo is a navigation bar with icons for home, information, email, and a book. The text "The Web Modelling Language" is partially visible on the right. Below the navigation bar is a horizontal menu with tabs for Overview, Book, News, People, Teaching, Research, Industry, and Community. The main content area is titled "WELCOME TO THE WEBML SITE ::". It is divided into four sections: "THE UML OF THE WEB ::", "THE WEBML BOOK ::", "TOOL SUPPORT ::", and "TEACHING RESOURCES ::". Each section contains an icon and a brief description. The "Data unit" icon shows a green cylinder on a grid. The "WebML Book" icon shows a blue book with "webml" written on it. The "Tool Support" icon shows two overlapping screens. The "Teaching Resources" icon shows a person with a key. The text in each section describes the respective topic.

WebML.org

The Web Modelling Language

Overview Book News People Teaching Research Industry Community

Overview **WELCOME TO THE WEBML SITE ::**

THE UML OF THE WEB :: **THE WEBML BOOK ::**

Data unit The Web Modelling Language is a visual notation for specifying the content, composition, and navigation features of hypertext applications, building on ER and UML.

entity [conditions]

THE WEBML BOOK :: "Designing Data-Intensive Web Applications" is the primary source of information about the WebML, overviewing the most relevant Web technologies, architectures, and methodologies.

TOOL SUPPORT :: **TEACHING RESOURCES ::**

WebML is accompanied by a CASE tool called WebRatio, which automatically generates fully functional applications from WebML diagrams. [WebRatio](#) is free for all non-commercial uses.

Tutorials, slides, solved exercises, and many other is available for professionals, teachers and students for building a complete training program on all the aspects of Web development.



WEBML e WebRatio



- ❑ A ferramenta WebRatio, oferece suporte para o desenvolvimento de aplicativos Web utilizando os conceitos do WebML.
- ❑ É possível fazer o download de uma versão Trial do software através do site www.webratio.com.
- ❑ Este Software pode ser utilizado tanto no Windows quanto no LINUX.

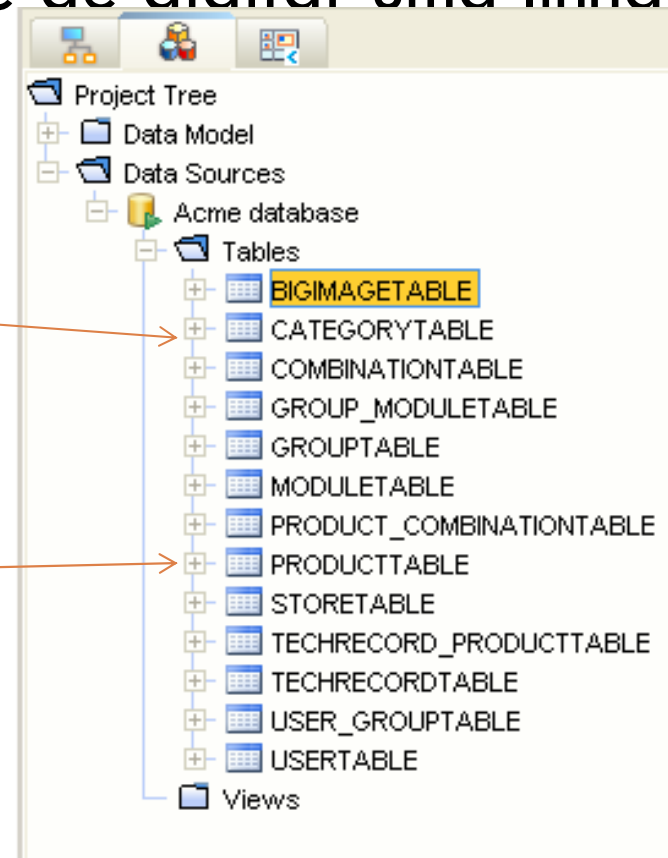
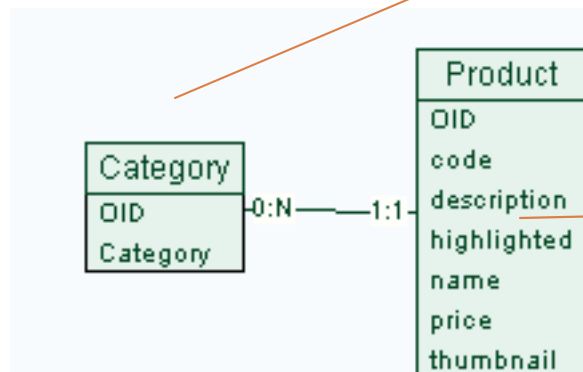
WebRatio – Casos de Sucesso

- Algumas organizações possuem suas aplicações web feitas através da ferramenta WebRatio.

Case	Highlights
	The Acer EMEA and PanAmerica Marketing and Communication divisions have been using WebRatio since 2000, to develop all their B2C and B2B Web multilingual solutions. More than 56 complex Web applications are deployed in over 30 countries and are currently being maintained and evolved by only 5 developers working part-time on these projects. You can find the entry point to the ACER EMEA applications at: www.acer-euro.com .
	SIA (Società Interbancaria per l'Automazione) , the leading Italian software integrator and solution provider in the banking sector, is using WebRatio to develop complex Web front-ends for the customers.
	Gemeaz (Accor Group) has developed a large Web B2B application used by the Italian schools in order to manage the outsourced catering service.
	SIRMI , a leading Italian Marketing Research Company, has developed an online Web application for integrating, managing and Web-deploying the databases of all the Italian IT companies; the B2C front-end supports powerful customizable Web queries and profile-based billing procedures.
	The Como Chamber of Commerce has commissioned the development of an innovative platform, based on web Services and Web front-ends for SMEs in homogeneous industrial districts; the platform supports corporate Web site publication, inter-company supply chain integration, product configuration, secure concurrent engineering, and technical documentation sharing.
	Politecnico di Milano , one of the largest European Technical Universities, has developed with WebRatio the Web sites and CMS applications of the department of Electronics and Information and of the Department of Physics.

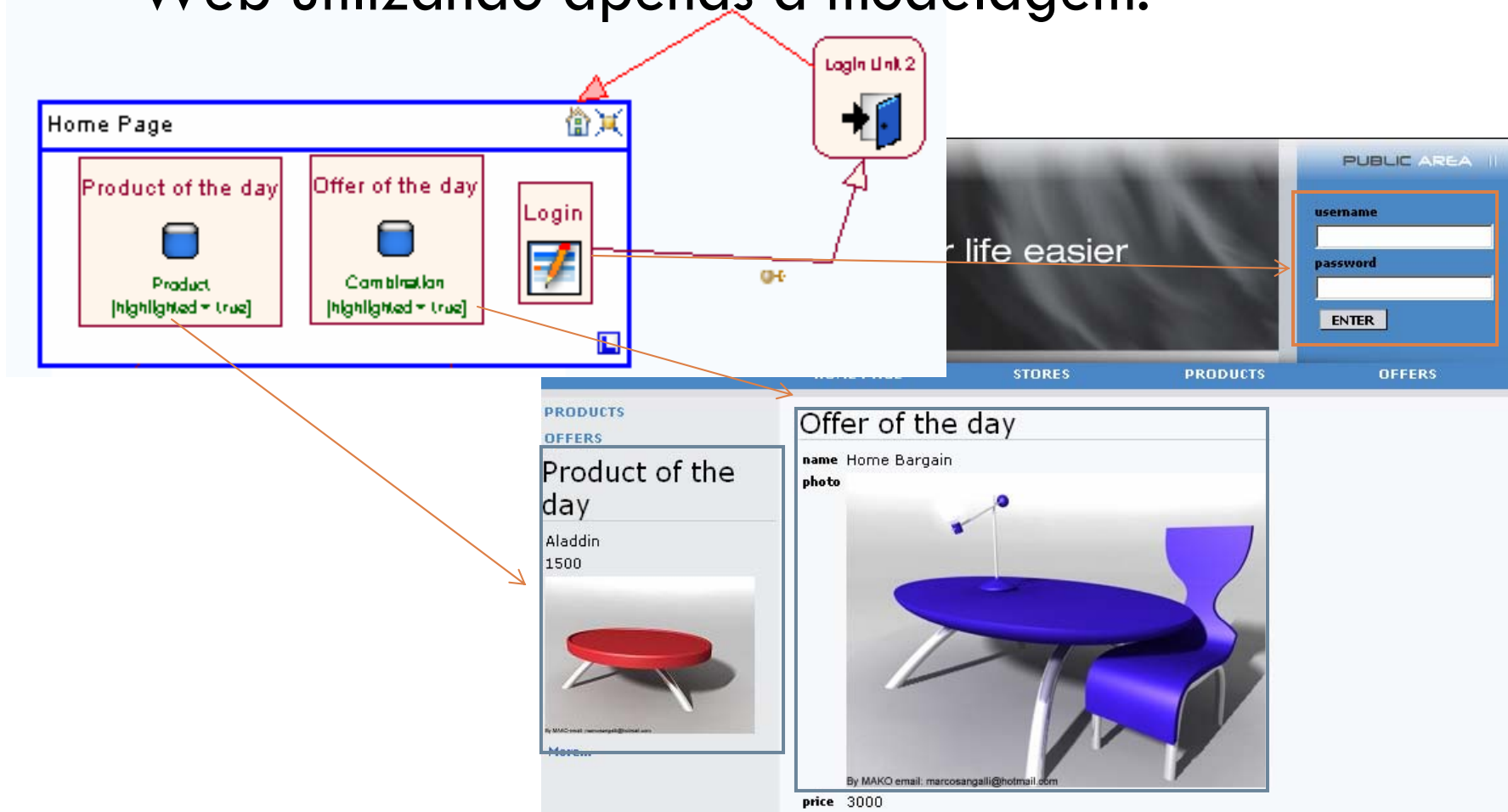
Ferramenta WebRatio

- Com o WebRatio, é possível criar um banco de dados sem a necessidade de digitar uma linha de código SQL.



WebRatio

- Com o WebRatio, é possível gerar uma aplicação Web utilizando apenas a modelagem.



WebRatio



- Porém, para criar a ferramenta, primeiro é necessário configurar o Ambiente de desenvolvimento para que a aplicação seja gerada de forma correta.
- O Objetivo é o desenvolvimento de uma mini-aplicação web para o conhecimento geral de uma ferramenta que gera o código automaticamente.

Pré-requisitos de Instalação

- JDK 6.0 ou superior :
 - <http://java.sun.com/javase/downloads/index.jsp>;
- WebRatio
 - www.webratio.com;
- BASE DE DADOS MySQL (opcional):
 - <http://dev.mysql.com/downloads/connector/j/5.1.html>
;
- TOMCAT (opcional):
 - <http://tomcat.apache.org/download-55.cgi>



Modelando no WebRatio

Visão Geral do WebRatio

- Esta é a página inicial do WebRatio após a criação do projeto.

Visão para modelo Web.

Aqui, ficam as configurações de conexão com o banco de dados.

The screenshot shows the WebRatio IDE interface. On the left, a project tree displays the following structure:

- Project
 - Data Model
 - Web Model
 - Visao Anonimo
 - CountryISOCtxParam
 - GroupCtxParam
 - LanguageISOCtxParam
 - UserCtxParam
 - Service/Data Providers
 - referenciaPos

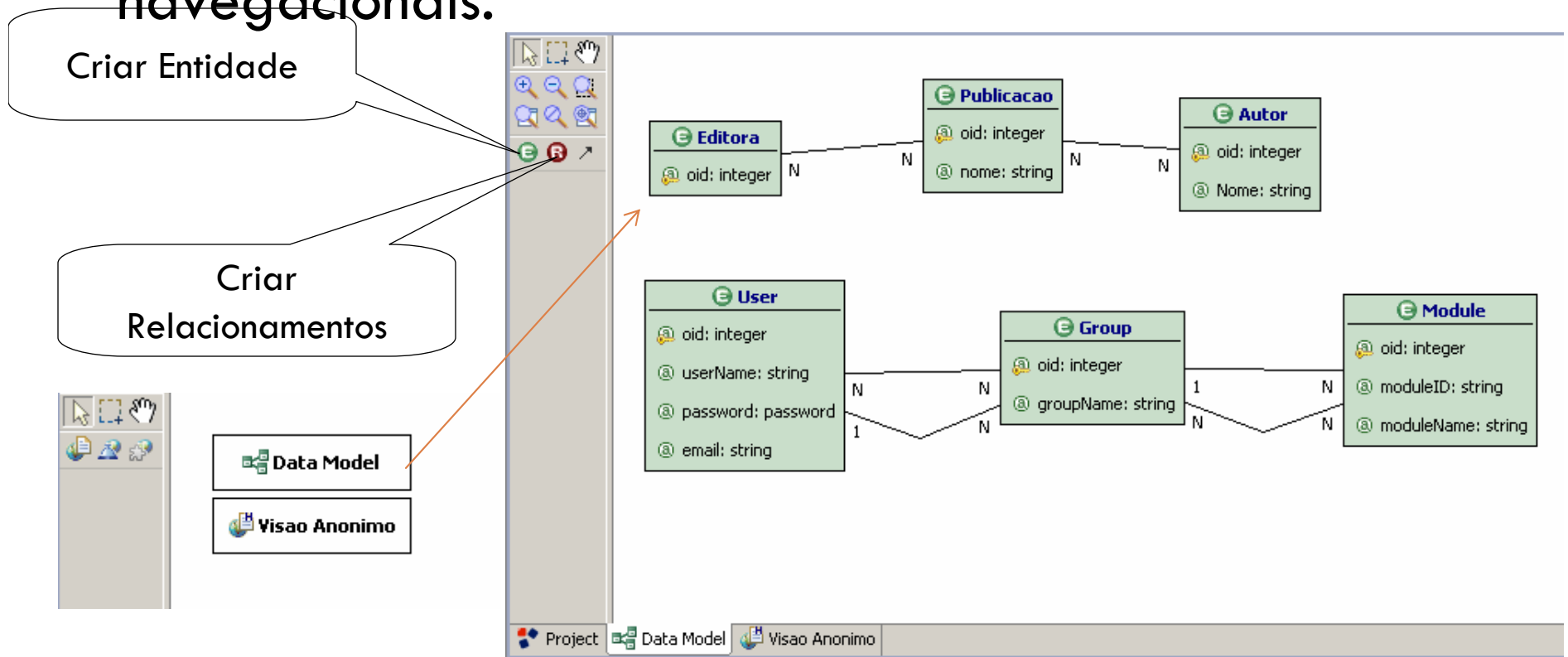
The main workspace on the right contains two buttons: "Data Model" and "Visao Anonimo". A callout bubble points to this area with the text: "Nesta área aparecerá a visão geral do projeto".

At the bottom, a table displays project settings:

Project	
Style	WRDefault
Page Layout	WRDefault/Default
Grid Layout	WRDefault/Default
Frame Layout	WRDefault/Default
Cell Layout	<Undefined>
Attribute Layout	WRDefault/Default

Visão Geral do WebRatio

- Esta visão é a visão de Edição, é por meio dela que criamos os diagramas de ER e os modelos navegacionais.



Modelagem conceitual



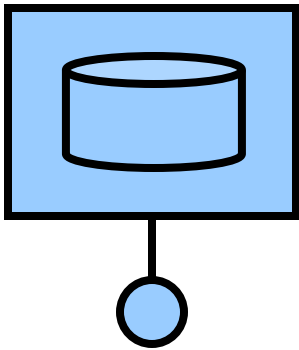
- Para o desenvolvimento da aplicação, primeiro é necessário realizar uma análise de requisitos, aconselho a criação de modelos de caso de uso e um modelo de objeto de negócios para se ter uma visão geral da aplicação.
- Mapear as entidades relacionamentos e então estruturar o modelo navegacional.

Modelo de Navegação e Apresentação

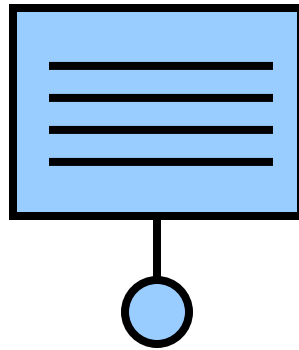
- Para a construção do modelo Navegacional é preciso conhecer os elementos base do WebRatio.
- O WebRatio faz uma mescla do modelo navegacional e o modelo de apresentação.

Modelo de Navegação e Apresentação

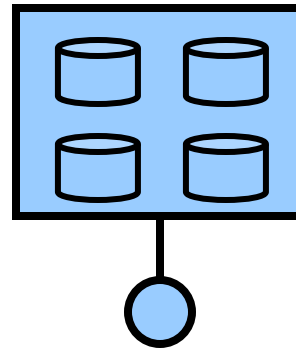
DATAUNIT



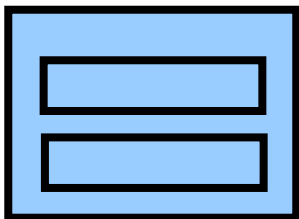
INDEXUNIT



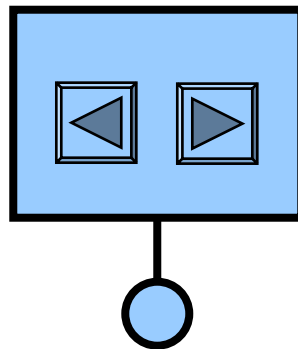
MULTIDATAUNIT



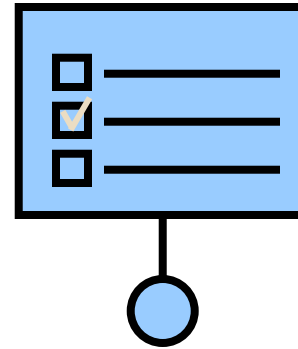
ENTRYUNIT



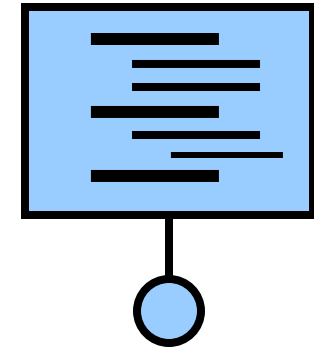
SCROLLERUNIT



MULTICHOICE




HIERARCHICAL



Modelo de Navegação e Apresentação





DATAUNIT

Author
first name:XXX
last name:YYY
photo: 

INDEXUNIT

Index of Authors
<ul style="list-style-type: none">•S. Ceri•P. Fraternali•O.Versand

MULTIDATAUNIT

All Authors	
	
	

ENTRYUNIT

Insert Your Data
<ul style="list-style-type: none">•Fname <input type="text"/>•Lname <input type="text"/>

SCROLLERUNIT

Browse Authors
5/12: go to <input type="text" value="1/12"/>
⏪ ⏩ ⏴ ⏵

MULTICHOICE

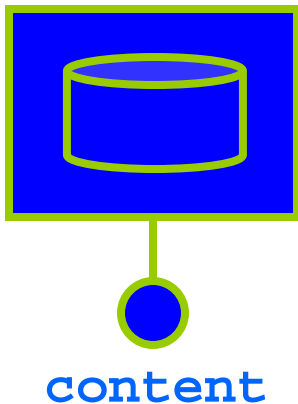
Choose Authors
<ul style="list-style-type: none"><input type="checkbox"/> Ceri<input checked="" type="checkbox"/> Fraternali<input type="checkbox"/> Versand

HIERARCHICAL

Books&Authors
<ol style="list-style-type: none">1. Web Applicat.<ul style="list-style-type: none">CeriFraternali2. Systems<ul style="list-style-type: none">Tannenbaum

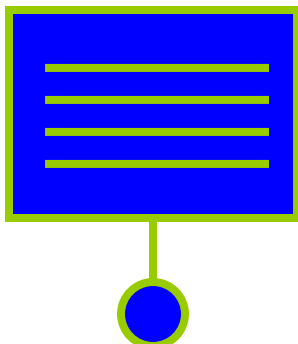
Modelo de Navegação e Apresentação

Data Unit



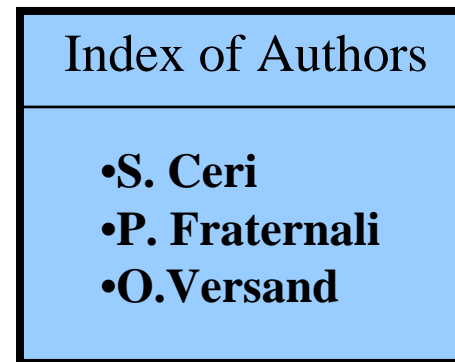
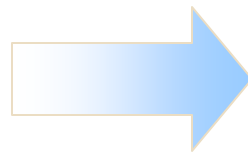
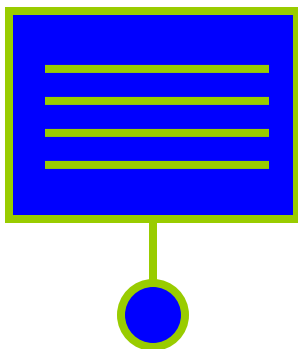
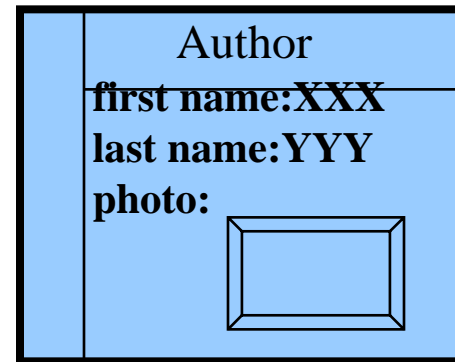
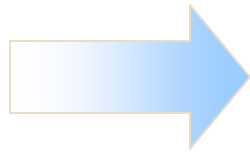
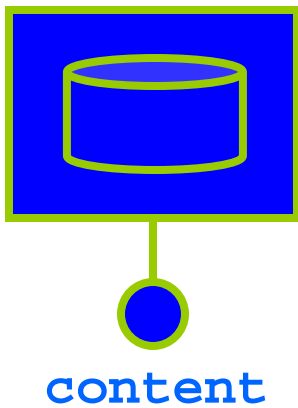
Para publicar informações sobre um único objeto (por exemplo, sobre um autor)

Index Unit

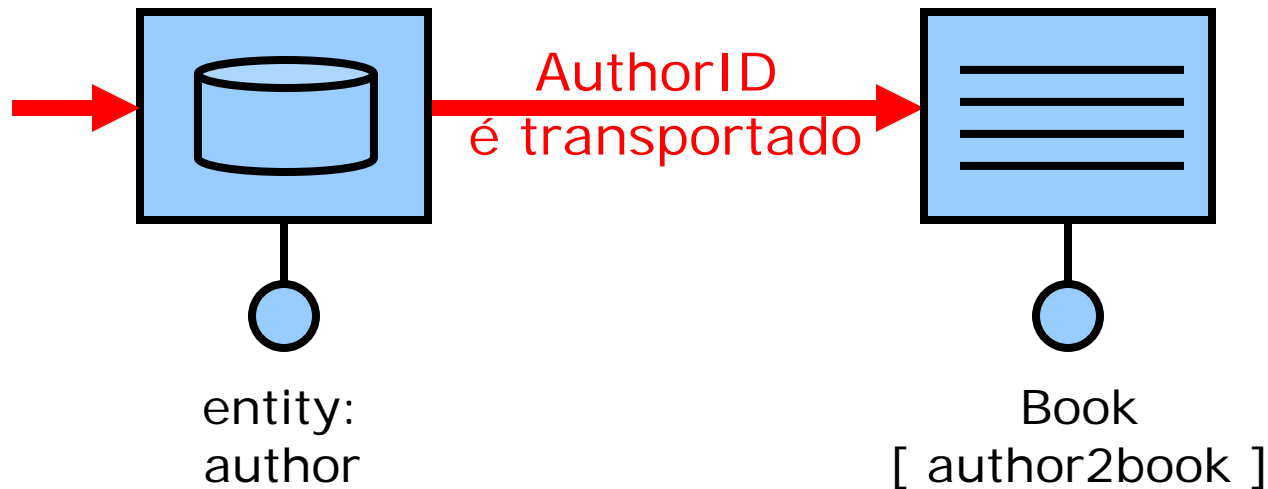


Para publicar informações sobre uma lista de objetos (por exemplo, uma lista de autores)

Modelo de Navegação e Apresentação



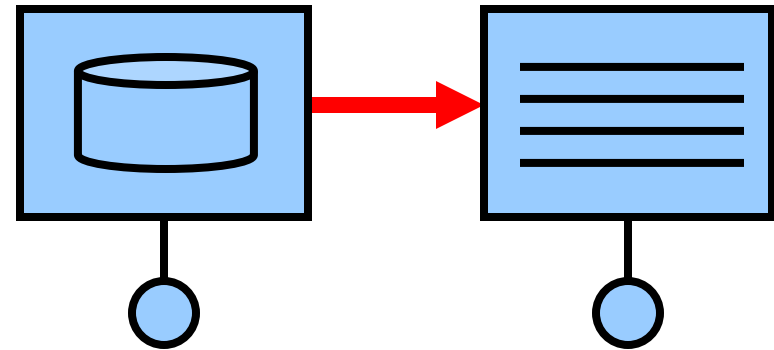
Modelo de Navegação e Apresentação - links



Modelo de Navegação e Apresentação - Tipos de Links

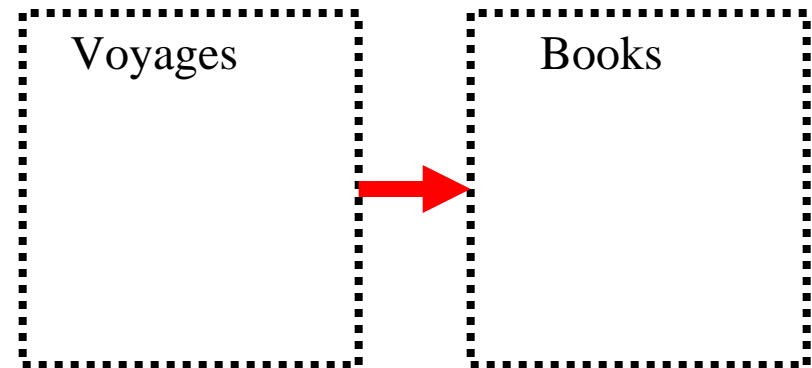
□ Links Contextuais

- Entre Units
- Transportam Contexto



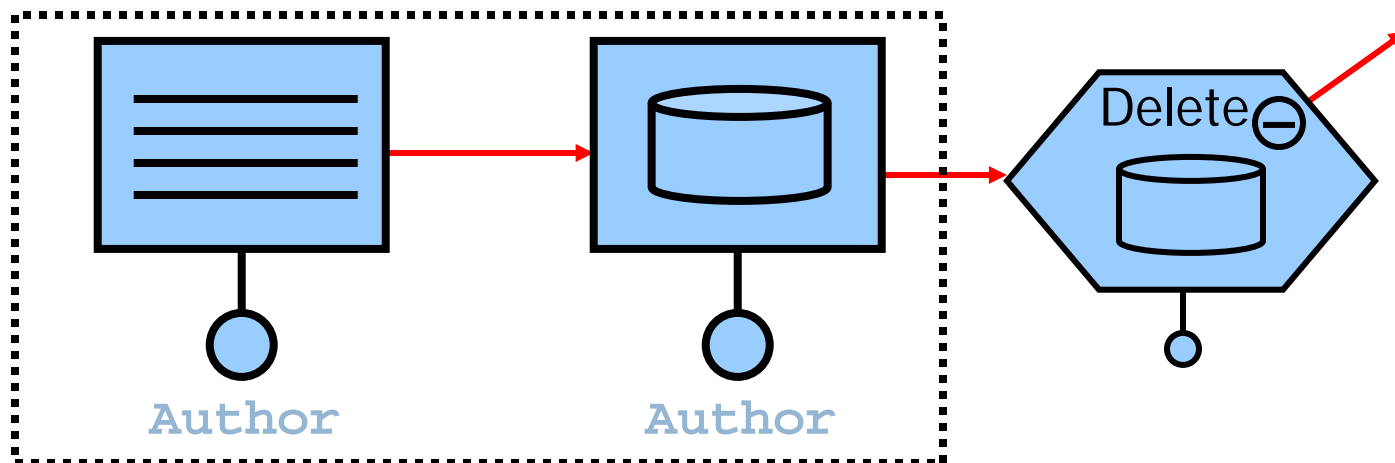
□ Links não-Contextuais

- Entre páginas
- Não transportam contexto



Modelo de Navegação e Apresentação - Operações

- Algumas operações default são disponibilizadas
- Operações customizadas podem ser definidas



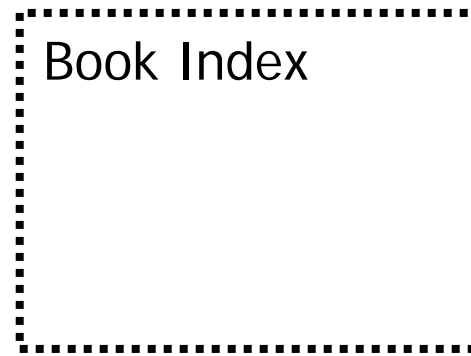
Modelo de Navegação e Apresentação - Páginas

- Uma página é um container de uma ou mais informações mostradas ao usuário simultaneamente
- Uma página pode ser composta de sub-páginas
- Um usuário navega pela páginas de um web site



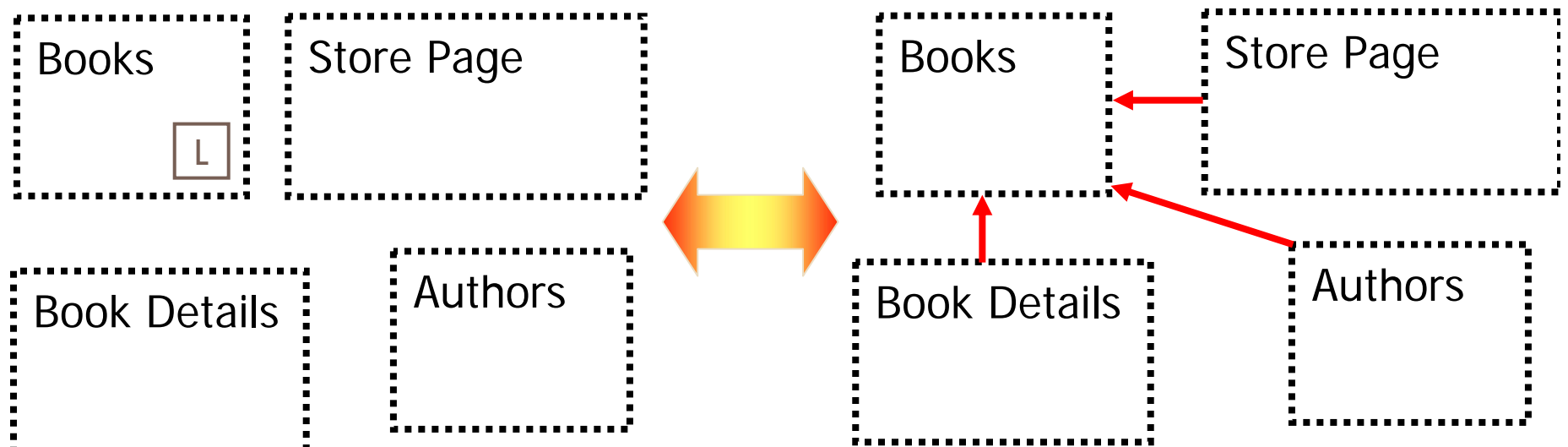
Modelo de Navegação e Apresentação - Páginas

- Home page: é a página principal de um site
- Primeira página que o usuário vê
- Todo site deve ter uma página marcada como home



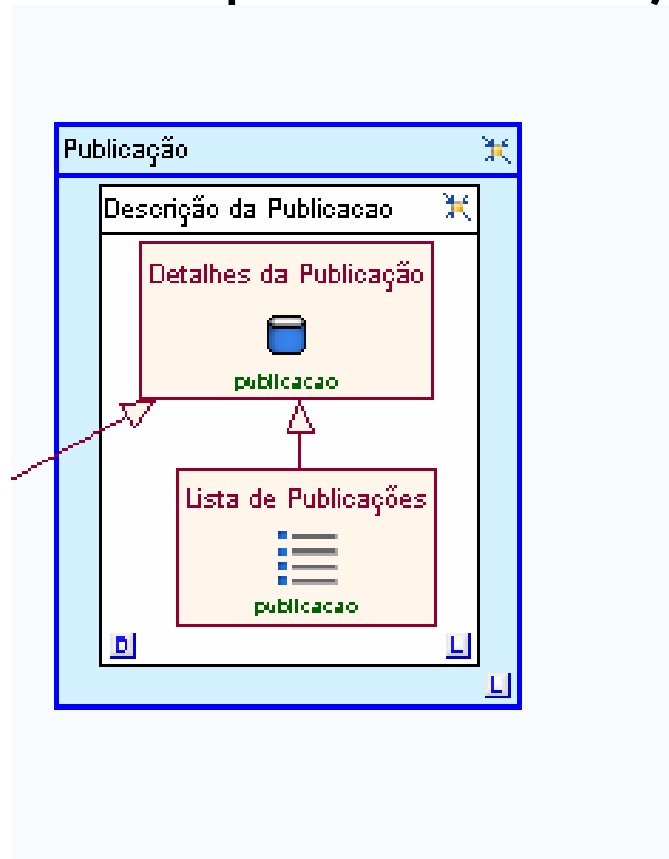
Modelo de Navegação e Apresentação - Páginas

- Landmark pages: páginas visíveis globalmente. O usuário pode ir para elas de qualquer lugar do site.
- É equivalente a uma página que possui links não-contextuais vindos de todas as outras



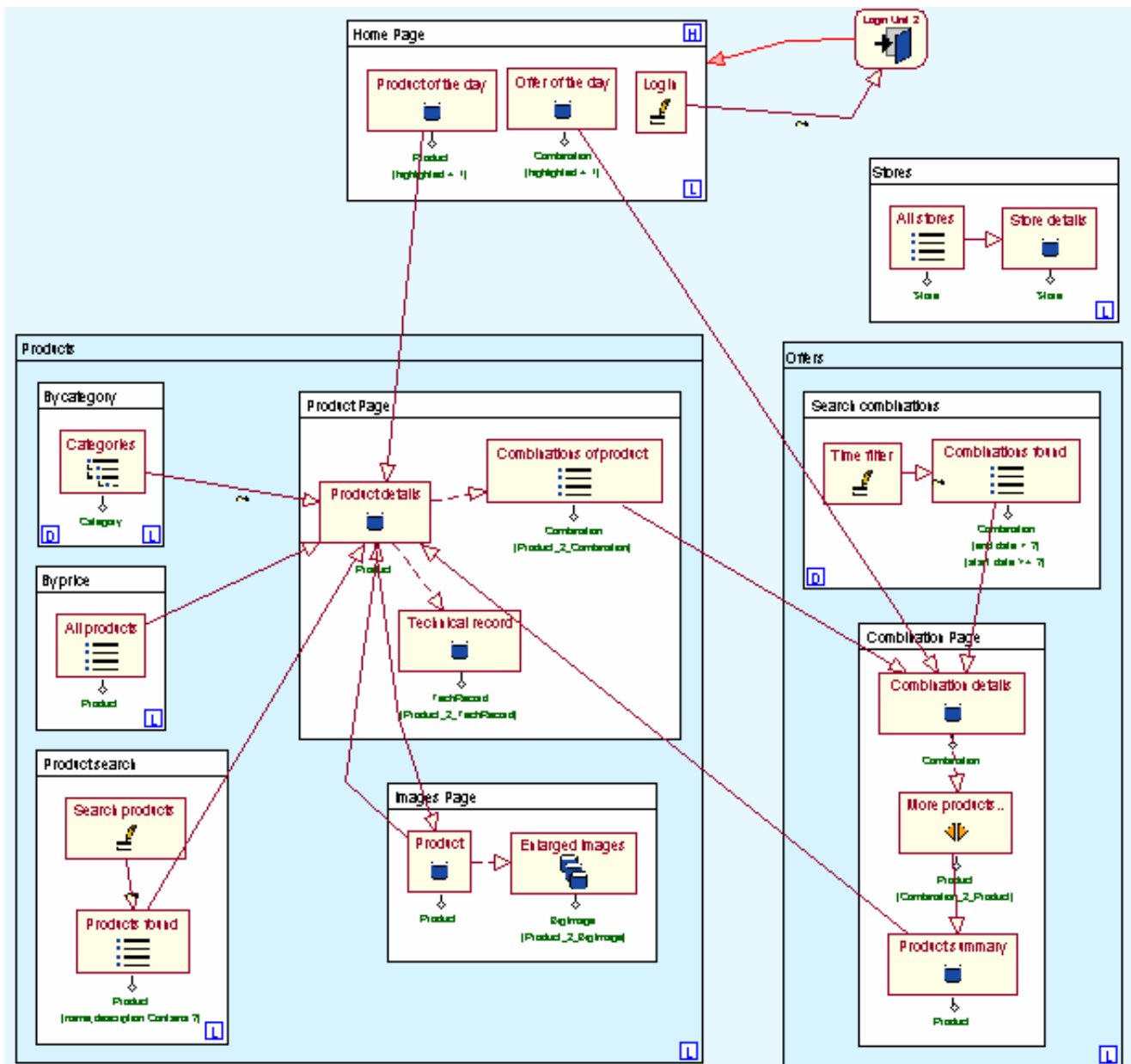
Modelo de Navegação e Apresentação - Áreas

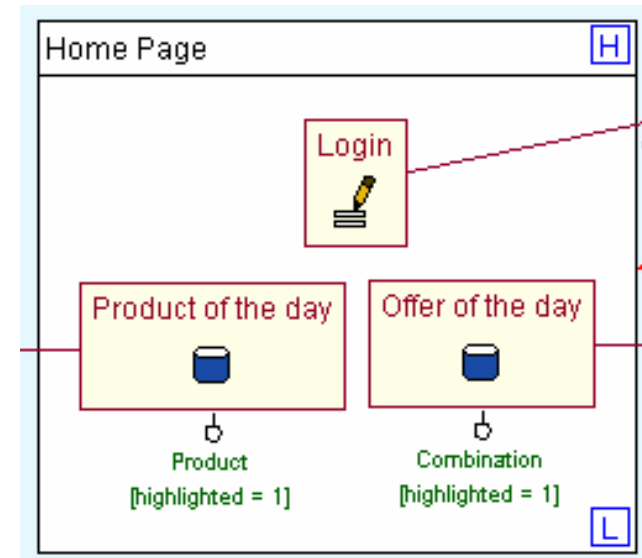
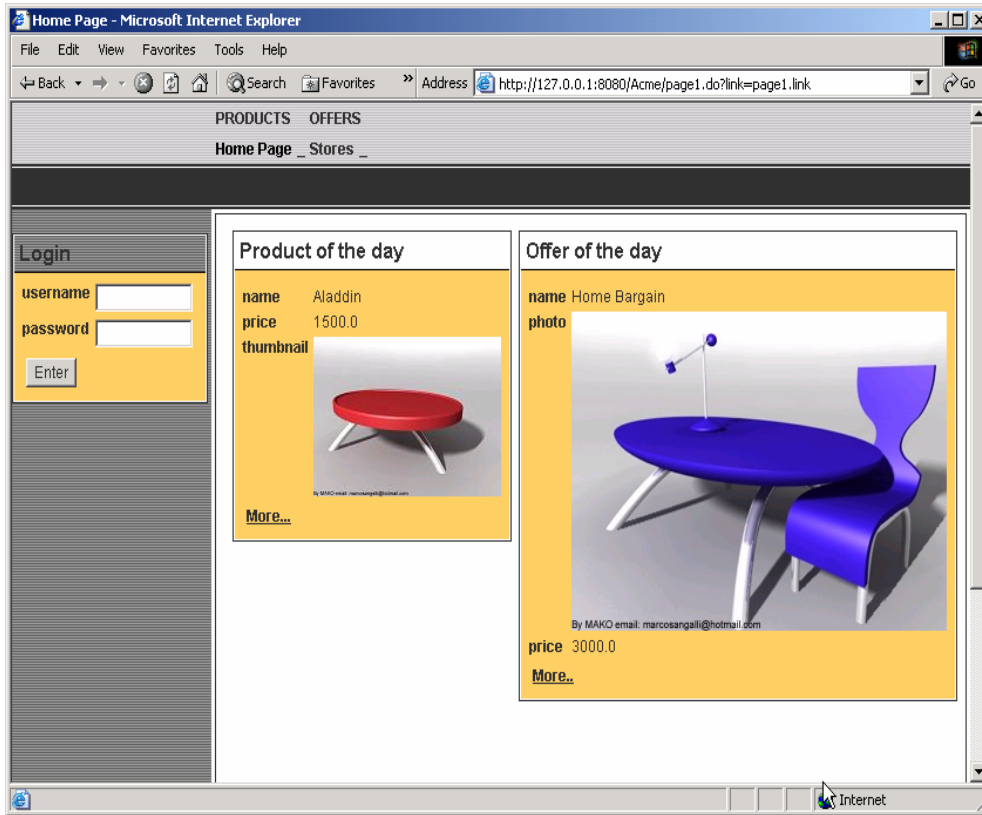
- São agrupamentos lógicos de páginas (por exemplo, seções de um portal: música, esportes, etc)



Modelo de Navegação e Apresentação - Siteviews

- Conjunto de páginas que um usuário visualiza
- Diferentes siteviews podem ser disponibilizados para diferentes grupos de usuários e/ou dispositivos
- Proporciona controle de acesso e conteúdo independente de plataforma





Pausa...

- Pausa para almoço.