UNIVERSIDADE ESTADUAL DA PARAÍBA – UEPB
CENTRO DE CIÊNCIAS EXATAS E SOCIAIS APLICADAS – CCEA

# MODELOS ARQUITETURAIS DE SISTEMAS DISTRIBUÍDOS

*Ingrid Morgane Medeiros de Lucena*

# ROTEIRO

- Conceito de Arquitetura de Software
- Principais elementos arquiteturais
- Modelos Arquiteturais
- Evolução das Arquiteturas
- Estilos Arquiteturais

# INTRODUÇÃO

- Sistema Distribuído
  - Parte complexa de software + componentes de hardware espalhados em múltiplas máquinas
  - Complexidade controlada devido a organização dos componentes;

- Arquitetura de Software
  - Apresenta como os componentes estão organizados e como devem interagir entre si e com outros;
  - A implantação de um SD necessita de instanciação e associação de componentes de SW em máquinas reais **distribuídas**, das mais diversas formas

# CONCEITO DE ARQ DE SOFTWARE

- É a estrutura de um sistema, formada por:
- Componentes de software;
- Propriedades externamente visíveis desses componentes e;
- Os relacionamentos entre eles.

- Estrutura ou organização dos mais significativos componentes do sistema e suas interações.

# MODELOS ARQUITETURAIS

Universidade ESTADUAL DA PARAÍBA

**Modelo arquitetural define como:**

- Os componentes se conectam uns com os outros;
- Como ocorre a troca de dados entre eles;
- Como é configurado ante o sistema num âmbito geral;

**Partes:**

- Componente: Unidade modular de interfaces;
- Conector: Mecanismo responsável pela comunicação, coordenação e cooperação entre os componentes do sistema

# PRINCIPAIS ELEMENTOS ARQUITETURAIS

Componentes:

- São unidades modulares com interfaces fornecidas bem definidas que são substituíveis dentro do seu ambiente.
- Estes são elementos de uma arquitetura que geralmente implementam:
- Processamento (funcionalidade ou comportamento);
- Estado (informação ou dados);
- Interação (interconexão, comunicação, coordenação e mediação).

# PRINCIPAIS ELEMENTOS ARQUITETURAIS

Componentes:

Podem ser:

- Uma simples operação
- Um operação complexa, como um sistema inteiro.
- Pode ser visto pelo usuário (humano ou outro software) somente através da sua interface pública.

# PRINCIPAIS ELEMENTOS ARQUITETURAIS

Universidade ESTADUAL DA PARAÍBA

## Conectores:

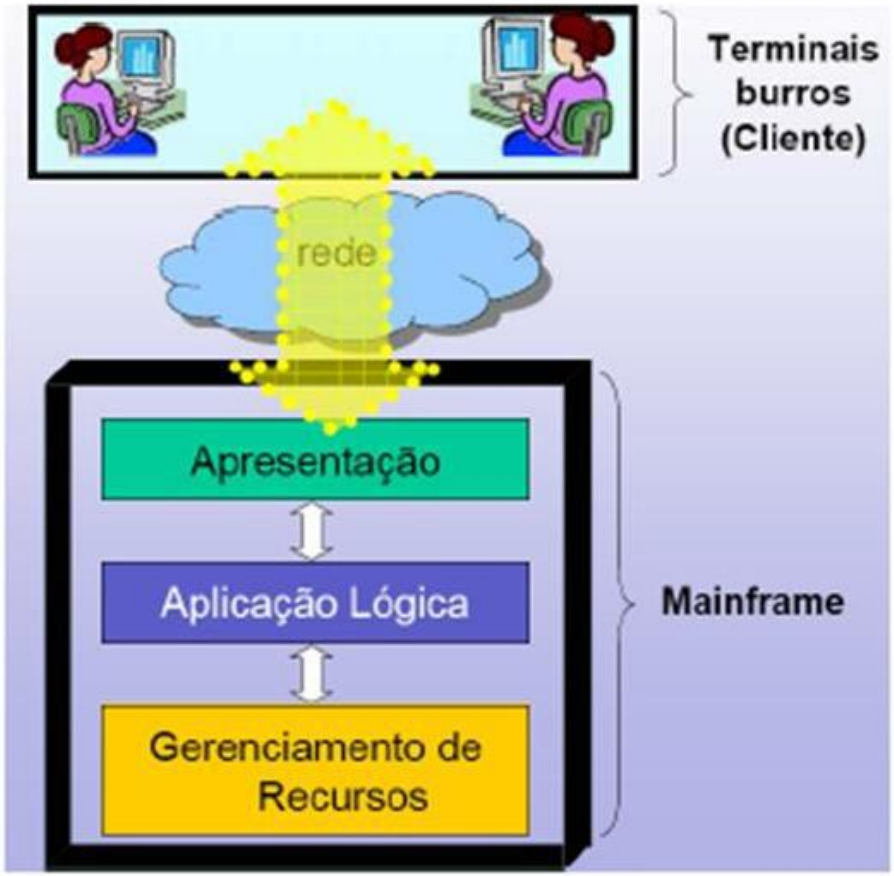
- É um tipo de componente responsável pela interação entre outros componentes.
- Em desktops convencionais os conectores são geralmente representados por simples chamadas de procedimento ou acesso a dados compartilhados em diferentes locais.
- Em sistemas complexos eles passam a ter identidades, papéis e artefatos de implementação.

Um modelo arquitetural é responsável por apresentar a forma pela qual os componentes dos sistemas interagem bem como a forma pela qual eles são mapeados em uma rede de computadores (Coulouris, 2005)

# EVOLUÇÃO DAS ARQUITETURAS

uepb Universidade ESTADUAL DA PARAÍBA

## Arquitetura 1 camada



# EVOLUÇÃO DAS ARQUITETURAS

Universidade ESTADUAL DA PARAÍBA

## Arquiteturas 1 camada (Mainframes)

- Dominantes até os anos 80 como arquitetura (meramente) corporativa.
- Possuía terminais (totalmente) “burros” somente para apresentar as informações.

# EVOLUÇÃO DAS ARQUITETURAS

Universidade ESTADUAL DA PARAÍBA

**Arquiteturas 1 camada (Mainframes)**

**Desvantagens:**

**– Manutenção difícil e cara, proporcional a sua eficiência.**

**– Problemas de congestionamento na entrada do servidor devido ao excesso de chegada de pedidos.**

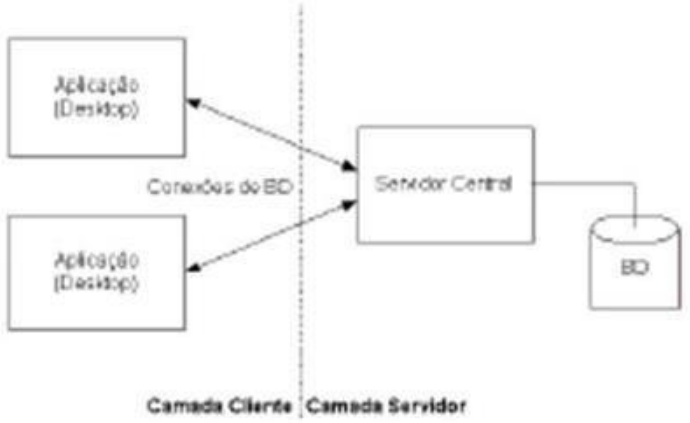
**– Congestionamento no tratamento da informação (sistema central responsável por tudo, até pela interface).**

**– Problema básico: interface nada amigável**

# EVOLUÇÃO DAS ARQUITETURAS

uepb Universidade ESTADUAL DA PARAÍBA

## Arquitetura Cliente/Servidor

# EVOLUÇÃO DAS ARQUITEURAS

Universidade ESTADUAL DA PARAÍBA

## Arquitetura Cliente/Servidor

## Vantagens:

- Camada de apresentação no Cliente utiliza processamento local (aproveita PCs da empresa).
- Possíveis ofertas de sistemas com interfaces gráficas amigáveis.
- Integração do desktop com dados corporativos.

# EVOLUÇÃO DAS ARQUITEURAS

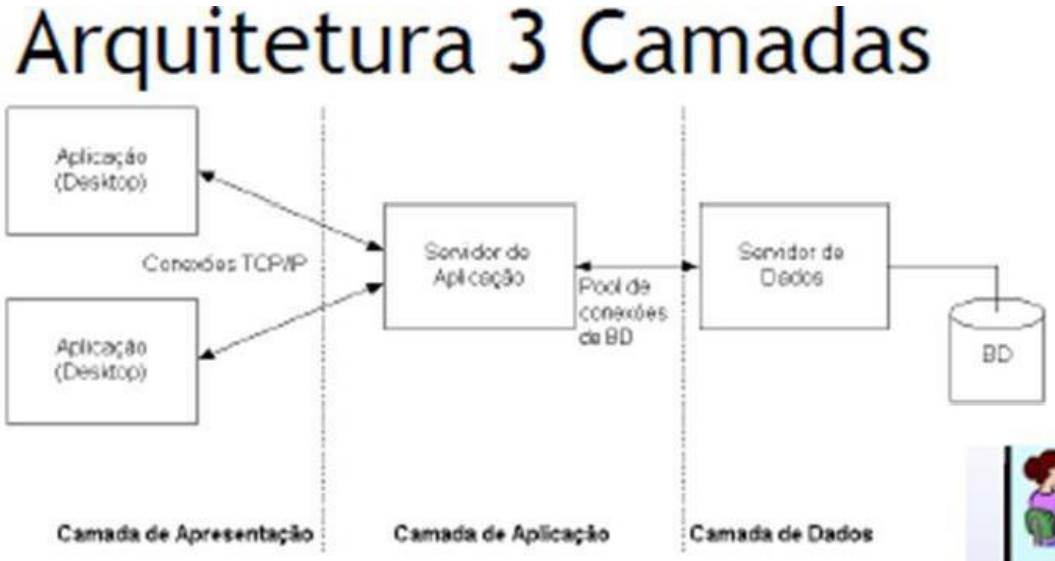
Universidade ESTADUAL DA PARAÍBA

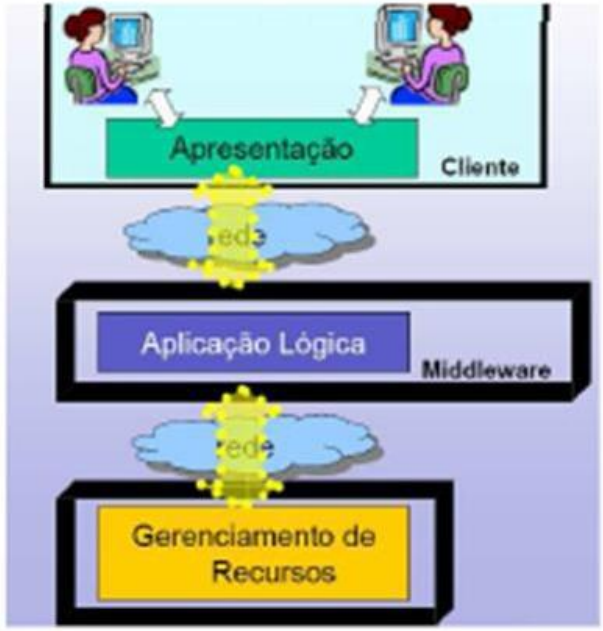
## Arquitetura Cliente/Servidor

## Desvantagens:

- Escalabilidade limitada
- Enormes problemas de manutenção (mudanças na lógica de aplicação forçava instalações)
- Cada cliente -> uma conexão como SGBD

# EVOLUÇÃO DAS ARQUITEURAS



# EVOLUÇÃO DAS ARQUITEURAS

Universidade ESTADUAL DA PARAÍBA

## Arquitetura 3 Camadas

- A arquitetura cliente/servidor em 2 camadas sofria de vários problemas, entre elas destacamos:
  - Falta de escalabilidade (conexões a bancos de dados).
  - Enormes problemas de manutenção (mudanças na lógica de aplicação forçava instalações).
  - Dificuldade de acessar fontes heterogêneas.

# EVOLUÇÃO DAS ARQUITEURAS

Universidade ESTADUAL DA PARAÍBA

## Arquitetura 3 Camadas

## Vantagens:

- Perda de performance é compensada com aumento de flexibilidade.
- Aumento da escalabilidade e confiabilidade do sistema.
- Problemas de manutenção foram reduzidos, devido a mudanças nas camadas de aplicação e de dados não necessitarem de novas instalações no desktop.

# EVOLUÇÃO DAS ARQUITEURAS

Universidade ESTADUAL DA PARAÍBA

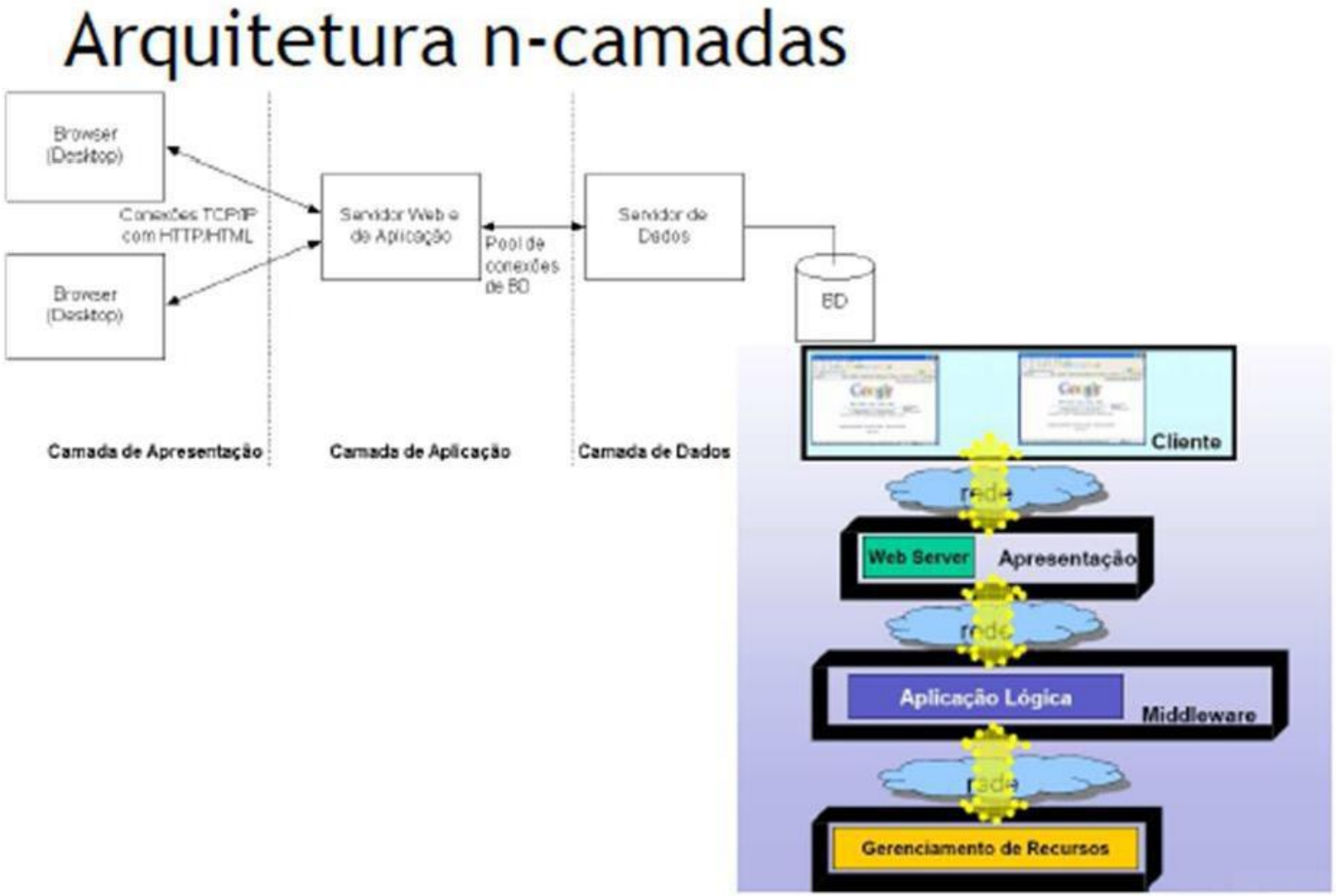
## Arquitetura 3 Camadas

## Desvantagens:

- Integração via Internet
  - Problema que a maioria não foi projetada para internet

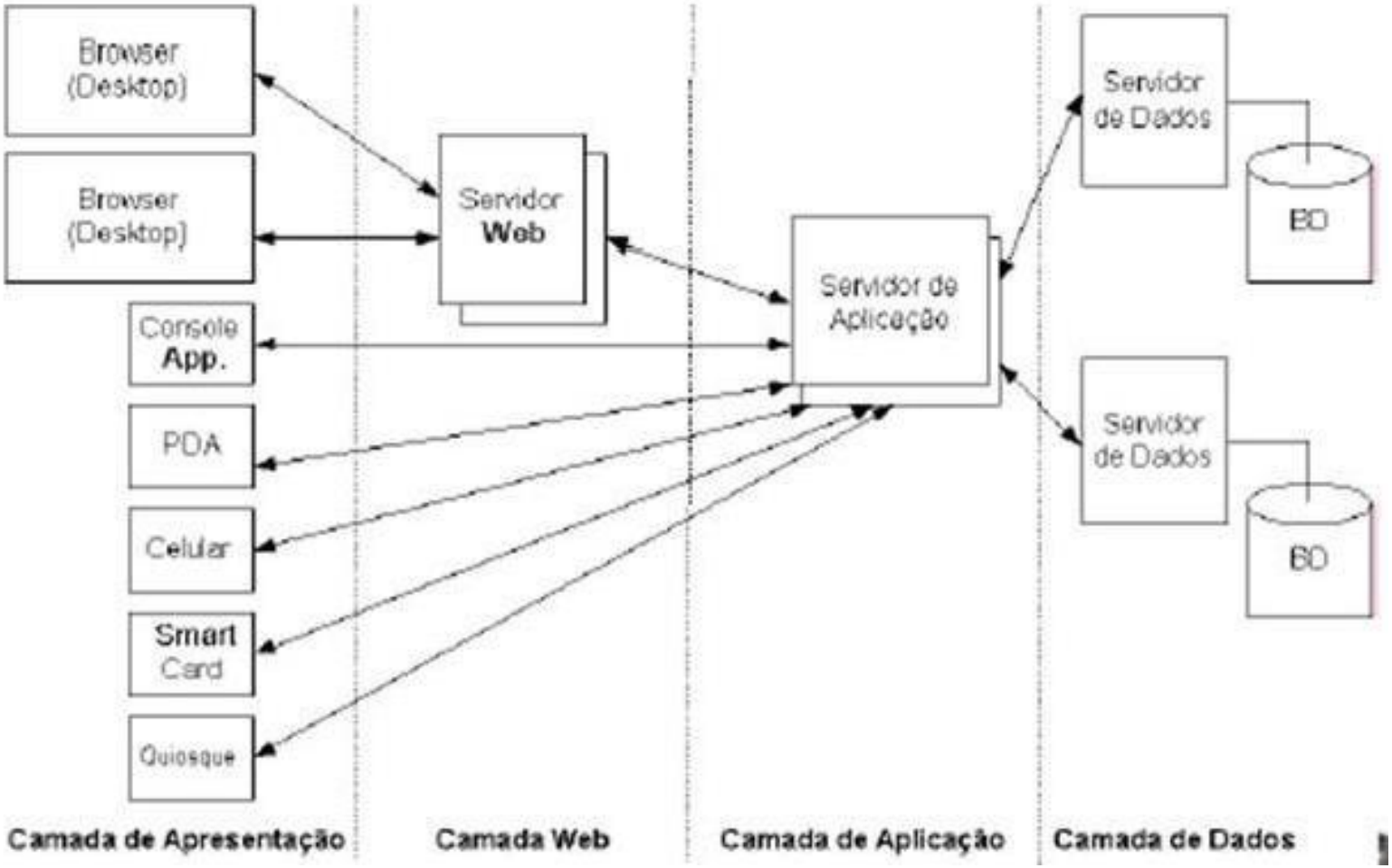
- Integração entre sistemas 3-tiers (3 camadas)
  - Falta de padronização

# EVOLUÇÃO DAS ARQUITEURAS



# EVOLUÇÃO DAS ARQUITEURAS

## Arquitetura n-camadas



# EVOLUÇÃO DAS ARQUITEURAS

Universidade ESTADUAL DA PARAÍBA

## Arquitetura n-camadas

- A arquitetura em 3 camadas original sofre de problemas:
  - ✓ A instalação inicial dos programas no desktop é cara.
  - ✓ O problema de manutenção ainda persiste quando há mudanças na camada de apresentação.
  - ✓ Não se pode instalar software facilmente num desktop que não está sob seu controle administrativo

# EVOLUÇÃO DAS ARQUITEURAS

uepb Universidade ESTADUAL DA PARAÍBA

## Arquitetura n-camadas

- Uso do Browser como Cliente Universal/Thin Client
- A camada de aplicação se quebra em duas: Web e Aplicação
- Evita-se instalar qualquer software no desktop e portanto, problemas de manutenção.
- Pode-se chamar de 3 camadas porque às vezes as camadas Web e Aplicação frequentemente rodam na mesma máquina (pequenos volumes).

# EVOLUÇÃO DAS ARQUITEURAS

Universidade ESTADUAL DA PARAÍBA

## Arquitetura n-camadas

**Desvantagens:**

- Complexidade.
- Fazer aplicações distribuídas multicamadas é difícil devido a necessidade de:
  - Implementar tolerância a falhas.
  - Implementar gerência de transações distribuídas.
  - Implementar balanceamento de carga.
  - Implementar resource pooling.
  - Muito middleware envolvido.

# EVOLUÇÃO DAS ARQUITEURAS

Universidade ESTADUAL DA PARAÍBA

## Arquitetura n-camadas

**Desvantagens:**

- Complexidade.
- Fazer aplicações distribuídas multicamadas é difícil devido a necessidade de:
  - Implementar tolerância a falhas.
  - Implementar gerência de transações distribuídas.
  - Implementar balanceamento de carga.
  - Implementar resource pooling.
  - Muito middleware envolvido.
- Informação redundante.
- Dificuldade e o custo de desenvolvimento e manutenção.
- Cresce exponencialmente com o número de camadas.

Pode se introduzir middleware num servidor de aplicação que ofereça esses serviços automaticamente. Além do mais, as soluções oferecidas (J2EE, .net) são baseadas em componentes. (Coulouris, 2005).

# Referências

Andrew S. Tanenbaum; Maarten van Steen - Distributed Systems: Principles and Paradigms, Prentice-Hall, 2007, ISBN-10: 0132392275, ISBN-13: 9780132392273

Lectures dos autores Andrew S. Tanenbaum e Maarteen van Steen ("www.cs.vu.nl" e "www.distributed-systems.net")

George Coulouris; Jean Dollimore; Tim Kindberg – Sistemas Distribuídos: Conceitos e Projeto, Bookman, 4th Edition, 2007, ISBN 9788560031498

Notas de Aula do Prof. Ricardo Anido do Instituto de Computação (IC) da UNICAMP - "www.ic.unicamp.br/~ranido"