

Sistemas Distribuídos

Carlos Ferraz
cagf@cin.ufpe.br

Tópicos

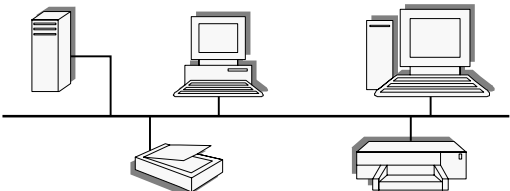
- Referências e Avaliação
- Introdução

Introdução

- Por quê ter sistemas distribuídos?
- Para quê?
- Como funcionam?
- Middleware

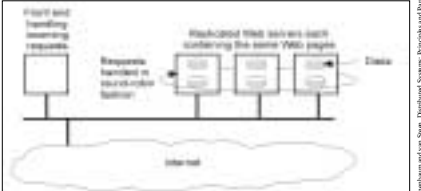
Por quê ?

- Aproveitar recursos de software e hardware distribuídos



Para quê ?

- Escalabilidade: atendimento à demanda
- Desempenho: múltiplos processadores
- Segurança
- Exemplos
 - ARPAnet: Hardware, software e pessoas distribuídos
 - Serviço Web moderno:
- Disponibilidade: replicação; tolerância a falha



Por que construir SDs?

- Maior confiabilidade
- Maior facilidade de crescimento – crescimento incremental
- Muitas aplicações são inerentemente distribuídas
- Robustez

Por que construir SDs é difícil?

- Complexidade
- Pouca experiência no projeto e implementação de aplicações distribuídas
- Dificuldade em tratar questões de redes
- Segurança

Como funcionam?

■ Suporte de comunicação

- Redes rápidas e confiáveis

Infra de comunicação

■ Suporte de software

- Sistemas operacionais de rede
- Sistemas operacionais distribuídos
- *Middleware*

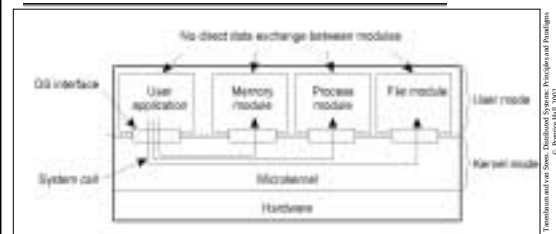
Infra de software



© 2002-2003 Carlos A. G. Ferraz

7

Sistemas Operacionais



- Processos de usuário: "user applications"
- Módulos do SO gerenciam recursos:
 - Físicos – exs.: CPU, disco, memória
 - Lógicos – exs.: arquivos, bancos de dados



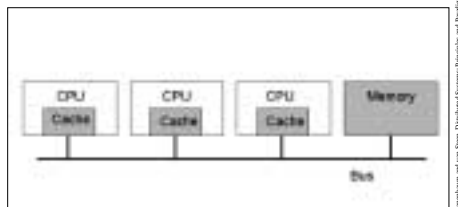
© 2002-2003 Carlos A. G. Ferraz

8

Sistemas Distribuídos

■ Do ponto de vista de hardware

- Multiprocessadores, com memória compartilhada



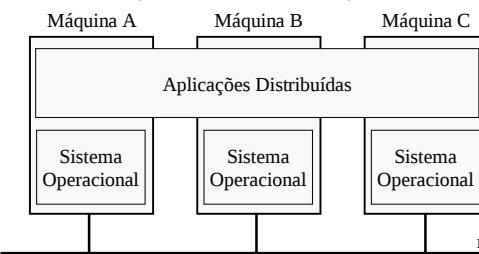
© 2002-2003 Carlos A. G. Ferraz

9

Sistemas Distribuídos

■ Do ponto de vista de hardware

- Multicomputadores, com memória privada



© 2002-2003 Carlos A. G. Ferraz

10

Sistemas Distribuídos (cont)

■ Do ponto de vista de software

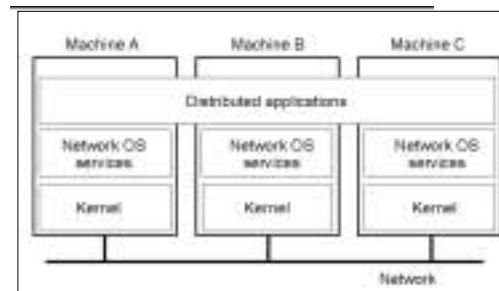
- Sistemas operacionais de rede
 - Exs.: Unix, Windows NT, 2000, XP
- Sistemas operacionais distribuídos
 - Exs.: Chorus, Mach, Amoeba
- *Middleware*
 - Exs.: Java RMI, DCOM/MS, CORBA/OMG



© 2002-2003 Carlos A. G. Ferraz

11

Sistemas Operacionais de Rede

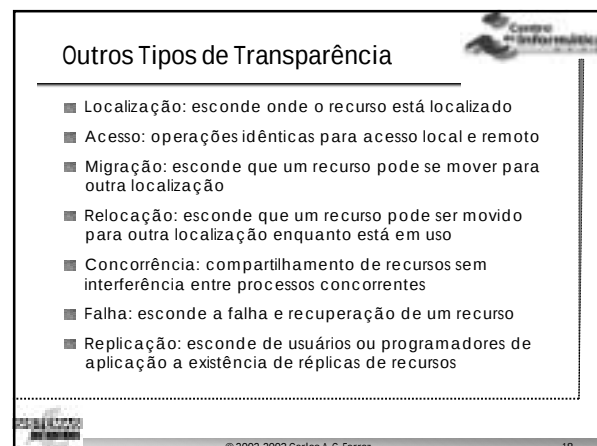
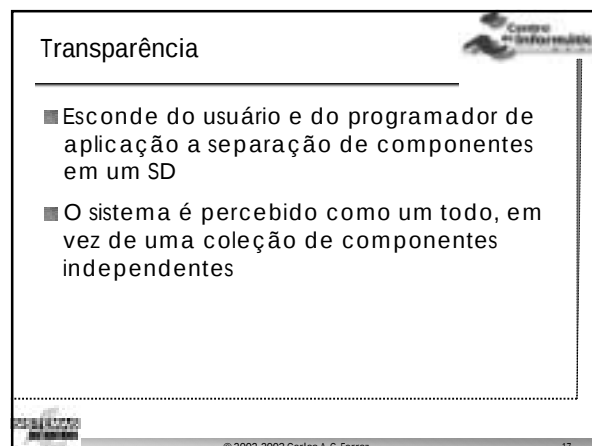
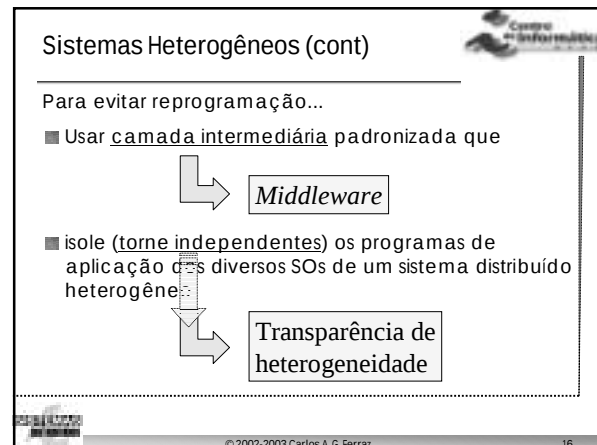
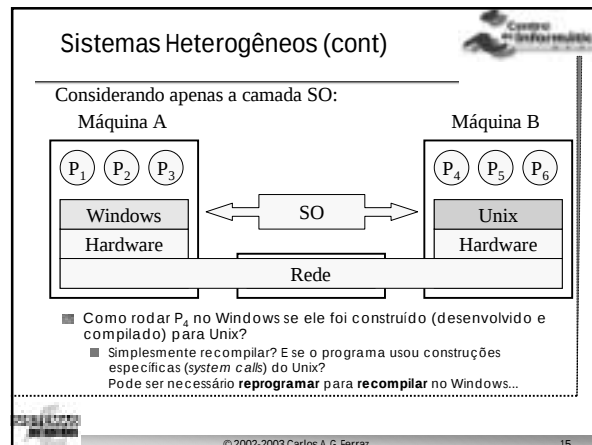
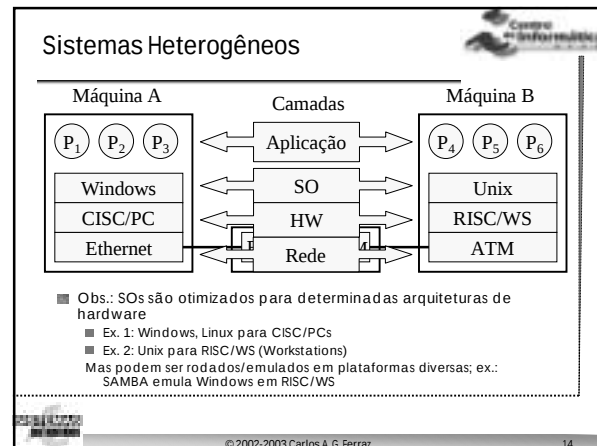
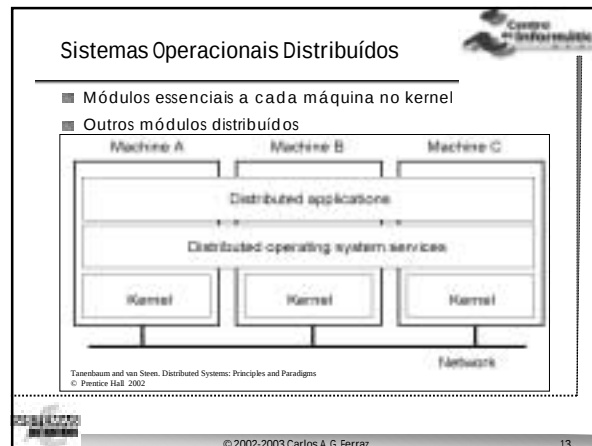


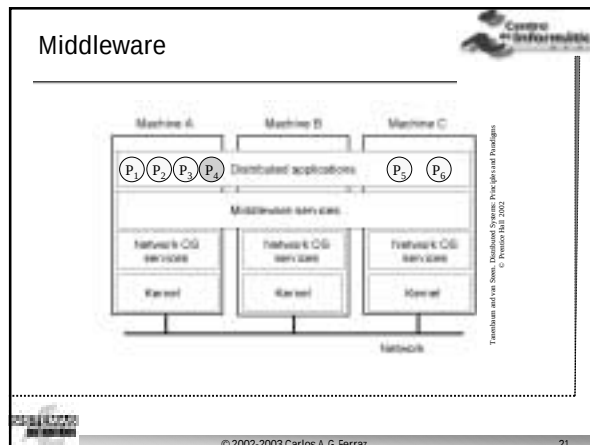
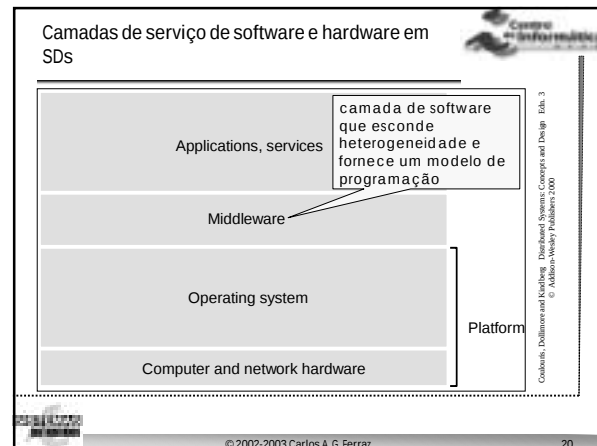
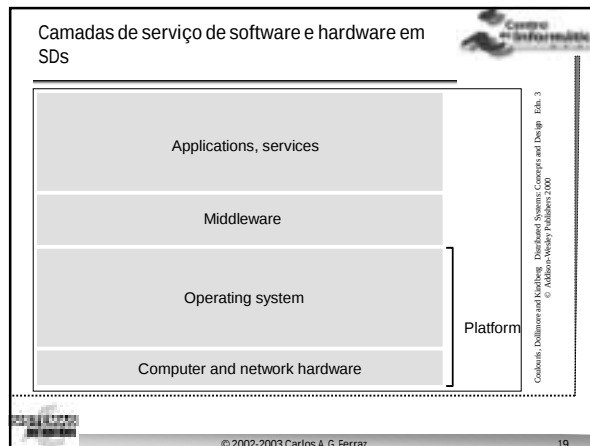
- Um dos módulos do SO é o **subsistema de comunicação**



© 2002-2003 Carlos A. G. Ferraz

12





Middleware: Transparências

- **Java RMI**
 - Transparência de SO, graças à JVM (Java Virtual Machine)
 - Restrito à programação em Java
- **DCOM**
 - Transparência de linguagem de programação (LP), mas dependente de SO (Windows)
- **CORBA**
 - Transparência de SO
 - Transparência de LP

© 2002-2003 Carlos A. G. Ferraz

Referências

- Home-page do curso
 - <http://www.cin.ufpe.br/~cagf/sdgrad>
- Distributed Systems: Principles and Paradigms
 - Andrew Tanenbaum, Maarten van Steen
 - Ed: Prentice Hall, 2002
- Distributed Systems: Concepts and Design
 - 2nd / 3rd Edition
 - George Coulouris, Jean Dollimore, Tim Kindberg
 - Ed: Addison-Wesley, 1995 / 2001

© 2002-2003 Carlos A. G. Ferraz

Avaliação

- Prova (50%)
- Desafio de programação (15%)
- Projeto (35%)

Segue...

© 2002-2003 Carlos A. G. Ferraz

Avaliação: Projeto

- Tarefas
 - Modelagem (ex. UML)
 - Especificação de interfaces de comunicação entre objetos usando uma IDL
 - Desenvolvimento usando Java RMI/CORBA/Webservices
- Produtos
 - Programas
 - Apresentação
- Critérios
 - "Executabilidade"
 - Usabilidade
 - Qualidade da definição das interfaces
 - Qualidade da apresentação
- Acompanhamento
 - Home-page
 - Cronograma atualizado
 - Documentação
 - etc



© 2002-2003 Carlos A. G. Ferraz

25