

Aspectos de projeto dos Sistemas Distribuídos

Sistemas Distribuídos

Eduardo Furlan Miranda

Adaptado de: PEREIRA, C. S. Sistemas Distribuídos.
Londrina: EDE SA, 2019. ISBN 978-85-522-1443-4.

Aspectos de Projeto

- Explorar aspectos diferenciais de sistemas distribuídos
 - Segurança, escalabilidade, resiliência e heterogeneidade são os principais
 - Essenciais para maximizar faturamento e satisfação do cliente

Relevância dos Aspectos

- Aspectos importantes no desenvolvimento de sistemas distribuídos
 - Segurança, escalabilidade, resiliência e heterogeneidade merecem destaque especial
- Entendimento sobre escalabilidade ajuda empresas a maximizarem o faturamento
 - Evitando uso desnecessário de recursos e mantendo a usabilidade

Segurança Distribuída

- Um dos aspectos mais importantes no projeto de sistemas distribuídos
- Sistema funciona em plataforma com várias máquinas, chamadas nós
- Comunicação entre nós ocorre por meio de redes de comunicação

Confidencialidade e Integridade

- Níveis de segurança em sistemas distribuídos (Coulouris et al., 2013)
 - Confidencialidade: acesso ao dado por indivíduos ou sistemas não autorizados
 - Exemplo: hacker acessa valor de atributo em banco de dados
 - Integridade: além do acesso, modificação do dado
 - Exemplo: hacker altera senha de usuário
- Segurança remete a um exercício de equilíbrio entre custo e ameaças (Coulouris et al., 2013)

Pontos de Atenção

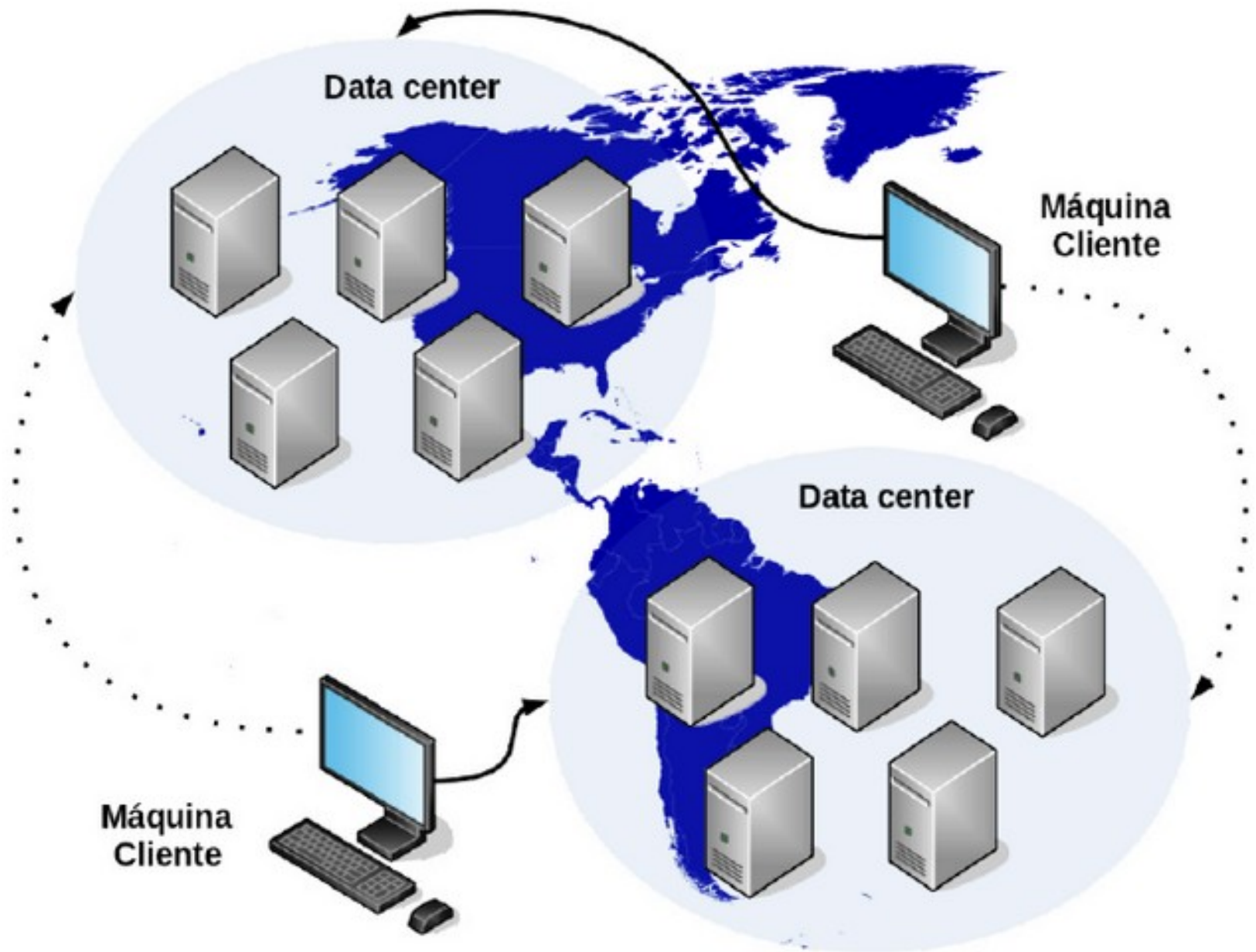
- Portas são expostas: permitindo que hackers enviem mensagens a elas
- Redes de computadores não são seguras
 - Remetentes de mensagens podem ser falsificados
 - Endereços IP podem estar duplicados
- Validade de chaves criptográficas deve ser limitada
- Algoritmos de criptografia podem ter falhas
- Hackers podem ter acesso a recursos poderosos, como ataques de força-bruta

Conceito de Escalabilidade

- Desempenho de um sistema aumenta com o acréscimo de hardware e software
 - Proporcionalmente à capacidade acrescida (Tanenbaum e Steen, 2008)
- Sistema permite aumentar ou diminuir a quantidade de recursos
 - Exemplo de escalabilidade “para baixo”: diminuição de servidores em baixa demanda para economia

Exemplo de escalabilidade geográfica

8/19



Escalabilidade Geográfica

- Sistema apresenta-se como único para o usuário
- Rodando em várias réplicas em dois ou mais datacenters geograficamente distintos
 - Exemplo: datacenters em São Paulo (Brasil) e Arizona (EUA)
- Benefícios: melhor experiência em conectividade e latência para os usuários
 - Tolerância a desastres, acesso a datacenter alternativo

Escalabilidade Administrativa

- Afetada pela escalabilidade geográfica
- Escopo administrativo é ampliado
 - Exemplo: provedores de internet diferentes para cada localidade
- Responsável pelo sistema distribuído terá mais trabalho para administrá-lo
 - Incluindo a necessidade de abrir um chamado de suporte técnico em outro idioma

Automatização Escalabilidade

- Pode ser aumentada ou diminuída de maneira automatizada
 - Sem intervenção do desenvolvedor
- Ferramentas utilizadas: Chef e Ansible
 - Diminuem recursos em baixa demanda, como um portal de notícias à noite
 - Aumentam recursos em picos de acesso, como notícias esperadas

Resiliência de Processos

- Um dos principais objetivos em aplicações distribuídas
- Relacionada com a comunicação confiável entre Cliente e Servidor
- Ideia básica: processos da aplicação replicados em grupos
 - Proteção contra falhas relacionadas a processos
- Para projetos tolerantes a falhas: detecção e mascaramento de falhas, replicação imperceptível

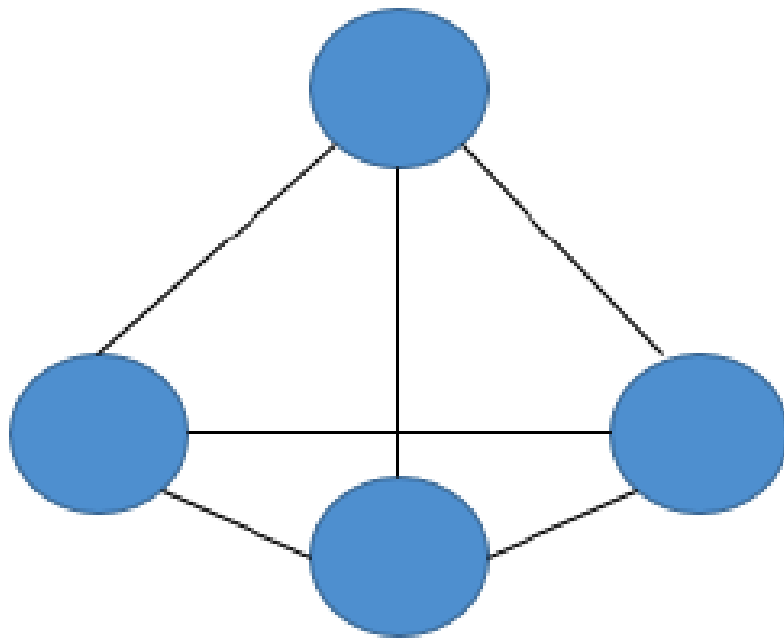
Grupos de Processos

- Processos idênticos organizados em grupos
- Mensagem enviada a um grupo: todos os processos membros recebem
- Ocorrência de falha em um processo: outro processo do grupo deve tratar a mensagem
- Sistemas distribuídos possuem grupos dinâmicos, processos podem se mover entre grupos

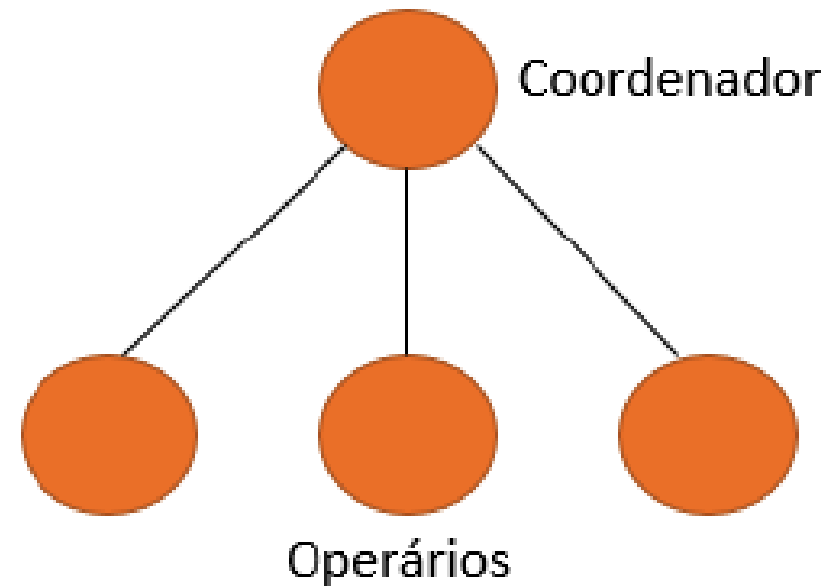
Estrutura dos grupos e comunicação dos processos

14/19

Grupo Simples



Grupo Hierárquico



Tipos de Grupos

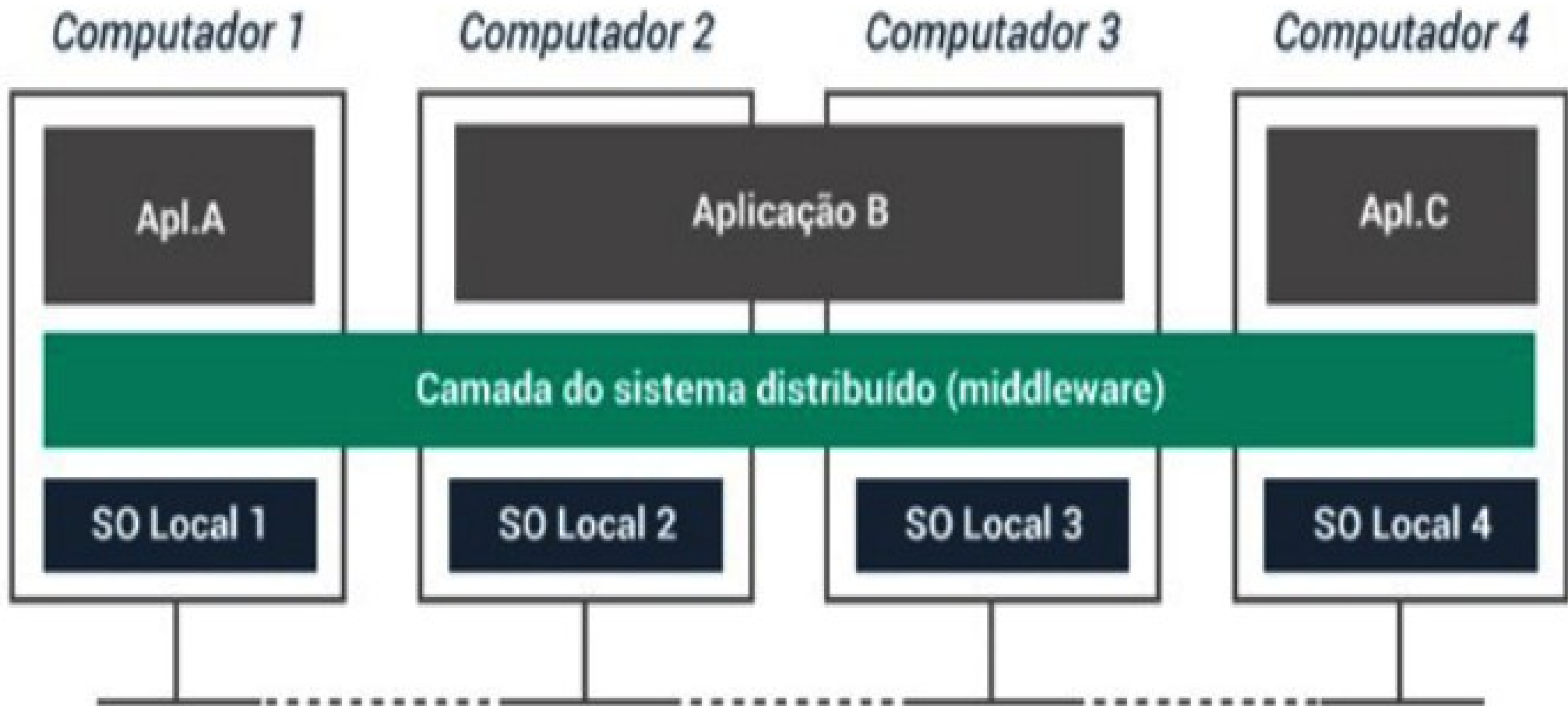
- Grupo simples: todos os processos são iguais
 - Decisões tomadas coletivamente, sem ponto único de falha
 - Desvantagem: tomada de decisão mais demorada
- Grupo hierárquico: baseado em hierarquia
 - Processos “coordenador” e “operários”
 - Coordenador decide o melhor operário para executar a requisição
 - Vantagem: agilidade na tomada de decisão
 - Desvantagem: falha no coordenador pára todo o serviço

Conceito de Heterogeneidade

- Significa algo desigual, com estrutura, função ou distribuição diferentes
- Sistema contém máquinas (nós) com sistemas operacionais, hardware e fabricantes diferentes
- Um dos aspectos mais frequentes em arquitetura distribuída

Middleware

17/19



Middleware Heterogeneidade

- Protocolos de rede são fundamentais, mas middleware é necessário
- Atua como camada central entre plataforma, sistema operacional e aplicações
 - Permite que diferentes aplicações rodem em diferentes plataformas e se comuniquem adequadamente
- Atua como camada de tradução para comunicação correta entre máquinas

Frameworks de Middleware

- CORBA: baseado em C++, arquitetura aberta, SOA, OO, eficiente no transporte de dados
- .NET Remoting: baseado no ambiente .NET, utiliza Visual Basic, C#
- Akka.NET: baseado no ambiente .NET, utiliza Visual Basic, C#
- Java RMI: baseado na linguagem de programação JAVA
- JAX-WS: baseado na linguagem de programação JAVA