



Exemplos de Sistemas Distribuídos

Sistemas Bancários

(caixas eletrônicos...)

Redes Sociais

(FB, Whatsapp, etc.)

Sistemas RT

(monitoramento...)

Peer-to-Peer

(file sharing...)

WWW

(prot. HTTP)

Redes de Sensores e IoT

(computação ubíqua, sensibilidade a contextos...)

Grid Computing

(CERN: 15PB, SETI,...)

Cloud Computing

(software, dados, infraestrutura,...)



Por que construir sistemas distribuídos?

Sistemas Distribuídos e Concorrentes



Por que Sistemas Distribuídos?

- ◉ porque há sistemas **inerentemente distribuídos**
 - ✦ *(e.g., monitoramento remoto, etc.)*
- ◉ porque há necessidade de se melhorar o **desempenho**
 - ✦ *(e.g., balanceamento de carga)*
- ◉ porque é desejável **compartilhar recursos**
 - ✦ *(“re-uso”)*
- ◉ porque deseja-se manter o sistema “no ar” (**disponibilidade**)
 - ✦ *(tolerância a faltas e “degradação progressiva”)*



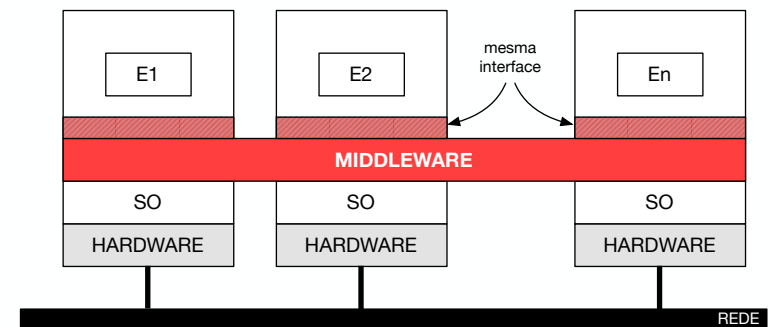
Como construir um sistema distribuído?

Sistemas Distribuídos e Concorrentes



Como implementar um Sistema Distribuído?

- ◉ Interfaces de programação de redes — API de **sockets**
 - baixo nível → complexidade!
 - suscetível a erros
- ◉ **Middlewares** e sistemas distribuídos
 - plataformas de *software* que auxiliam o desenvolvimento
 - organizados em camadas
 - ✦ (esconde diferenças de SO, HW, ...)





Como implementar um Sistema Distribuído?

● *Middleware:*

- recursos para facilitar comunicação entre entidades computacionais

- ✦ **esconde detalhes de comunicação:**

- “transparências”

- **serviços adicionais:**

- ✦ **segurança**
- ✦ **contabilidade**
- ✦ **localização**

- ✦ **etc...**

- fornece funcionalidades necessárias para um SD:

- ✦ **comunicação (*Remote Procedure Call – RPC*)**

- ✦ **transações (propriedades ACID...)**

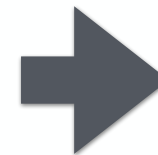
- ✦ **composição de serviços (programar = montar “quebra-cabeça” → *Web Services*)**

- ✦ **confiabilidade (e.g., garantias de entrega, tolerância a faltas)**



Visão Geral

1. O que são sistemas distribuídos?
 - “entidades” ou nós conectados
 - comunicação por trocas de mensagens
2. Por que sistemas distribuídos?
 - desempenho
 - disponibilidade (tolerância a faltas)
3. Como implementar sistemas distribuídos?
 - comunicação de baixo nível (*bytes*)
 - *middleware*





A Disciplina

Sistemas Distribuídos e Concorrentes

1. Fundamentos de Sistemas Distribuídos

- arquiteturas
- modelos de comunicação
- aspectos de projeto de SDs
- fundamentos de comunicação

2. Concorrência e coordenação em SDs

3. *Middleware* para SDs

- conceitos básicos

- classes (OD, serviços, *message-oriented*, *tuple-space*)

✦ *diferentes linguagens de programação (C++, Java, Python, ...)*

4. Tempo em SDs

5. Introdução a Algoritmos

6. Introdução à Computação Distribuída na Presença de Falhas



AULAS PRÁTICAS



Plano de Ensino

Sistemas Distribuídos e Concorrentes