

1

1

ASDPC 06 Middleware

Arquitetura de Sistemas Distribuídos,
Paralelos e Concorrentes
Escola Politécnica – PUCPR
Luiz A. de P. Lima Jr. • luiz.lima@pucpr.br

2

Middleware

- Problemas com a construção de sistemas distribuídos complexos usando diretamente sockets:
 - ✦ Necessidade de definição de protocolos
 - *Tedioso e suscetível a erros...*
 - ✦ Serialização de dados
 - *Sequência de bytes*
 - TCP: "stream" = *necessidade de delimitação*
 - ✦ Conversões de formatos
 - *Little-endian - big-endian*
 - *Entre linguagens de alto nível. Exemplo:*
 - char: C/C++ ⇒ byte: Java

3

3

Middleware

- Solução:
 - ✦ Usar uma "PLATAFORMA" para facilitar o desenvolvimento de aplicações distribuídas

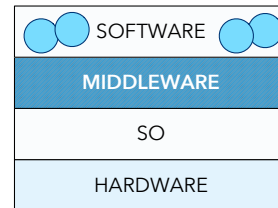
MIDDLEWARE

4

4

Middleware

- Transparência de **localização**
- Transparência de **representação de dados**
- **Operações remotas** (ao invés de *bytes*)
- **Migração** de objetos
- Outros **serviços...**

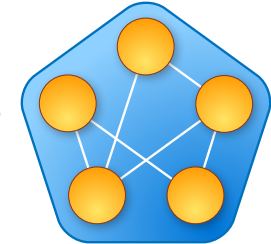


5

5

Middleware

Distribuição
Atualização
Manutenção
Re-uso
Flexibilidade
Disponibilidade
Tolerância a faltas
Etc.

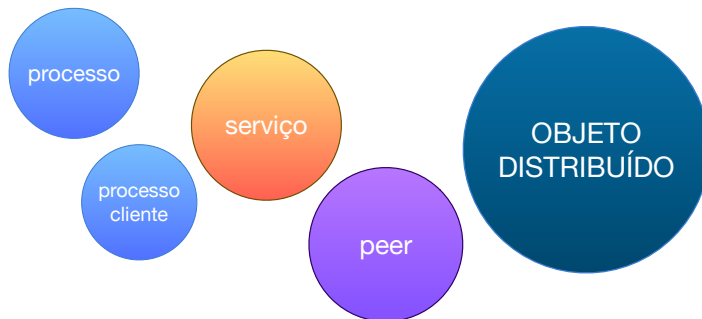


6

6

Middleware

- Quem são as entidades computacionais comunicantes?



7

7



ASDPC

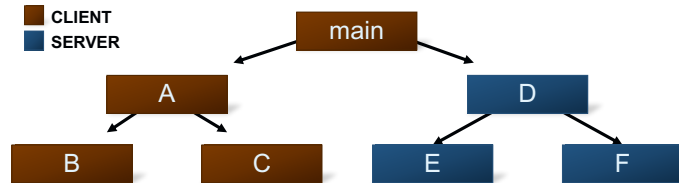
Remote Procedure Call (RPC)

Middleware

8

Remote Procedure Calls (RPC)

- Usar o paradigma de **chamada de procedimento** para construir software cliente/servidor
 - ✦ (programação “orientada a procedimentos”)



9

9

Remote Procedure Calls (RPC)

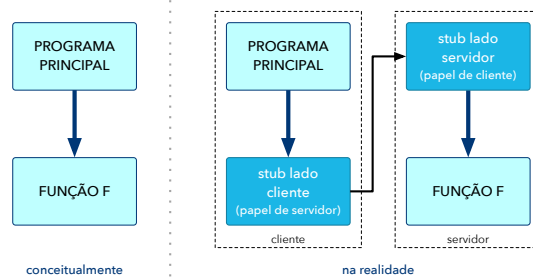
- **Dificuldades** trazidas ao SO pelas RPCs:
 - ✦ **Localização e invocação** de serviços (ambiente *runtime*)
 - ✦ **Falhas** (*timeout, retrial*)
 - ✦ **Segurança** (autenticação, etc.)
 - ✦ **Ligação** cliente/servidor (diretórios)
 - ✦ **Representação de dados** da rede (XDR)
 - ✦ Passagem de **parâmetros**

10

10

Remote Procedure Calls (RPC)

- Código adicional (**stubs/proxies**):
 - ✦ *Marshalling*
 - ✦ Envia mensagem
 - ✦ Espera resposta
 - ✦ *Unmarshalling*



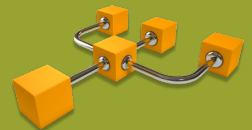
11

11



ASDPC

Objetos e Objetos Distribuídos



Middleware

12

Problemas

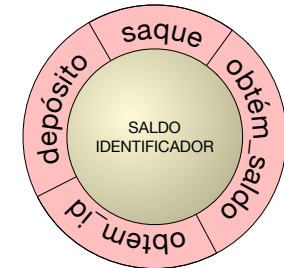
- O que é um **objeto distribuído**?
- Como **compartilhar** estes objetos? Como lidar com a **distribuição**?
- Como lidar com a **heterogeneidade**?
- Como garantir **escalabilidade**?
- Como introduzir **tolerância a falhas** e outras **características não funcionais**?
- **Meta-programação**?

13

13

Objetos

- **Objetos:**
 - ✦ Dados → **ATRIBUTOS**
 - ✦ Operações → **MÉTODOS**
- **Interfaces:**
 - ✦ grupos de métodos
 - ✦ para comunicação:
 - o que importa é sabe **O QUE** o objetos faz e não **COMO**

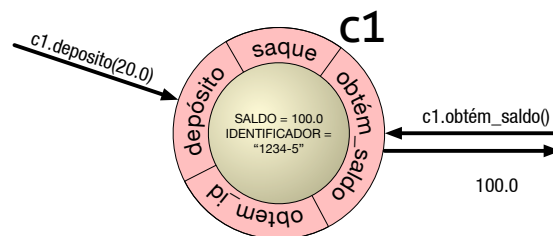


14

14

Comunicação entre Objetos

- **Comunicação** entre objetos:
 - ✦ exclusivamente através de trocas de mensagens
- **Mensagens:**
 - ✦ Objeto-alvo
 - ✦ Método
 - ✦ Parâmetros



15

15

Objetos x Objetos Distribuídos

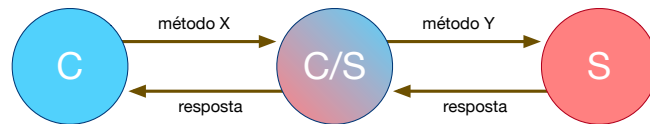
- **Objetos:**
 - ✦ Internos a um programa (reuso de código)
- **Objetos Distribuídos:**
 - ✦ Disponíveis na rede
 - ✦ Componentes “autônomos”
 - ✦ Relacionamentos cliente-servidor (ou P2P)

16

16

Comunicação entre Objetos Distribuídos

- Paradigma **cliente-servidor**
- Relações C/S são dinâmicas
- Invocações estilo **RPC** ("Remote Procedure Call")

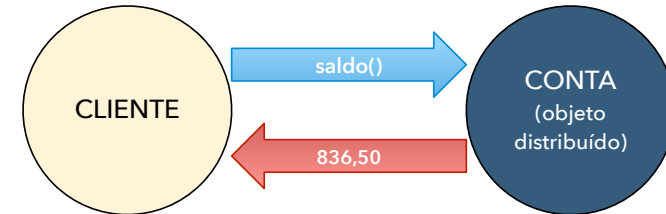


17

17

Middleware OO

- Middlewares orientados a objetos, utilizam a mesma abstração de RPC aplicando-a a **métodos dos objetos**.



18

18

Principais Tecnologias de Middleware

- **Objetos Distribuídos:**
 - ✦ **CORBA** (OMG)*
 - ✦ RMI, J2EE (Oracle)
- **Web services**
 - ✦ **JWS/Glassfish** (Oracle)*
 - ✦ Axis (Apache)
 - ✦ .Net (Microsoft)
 - ✦ **Flask** (REST)*
- **Orientados a Mensagens**
 - ✦ DSS (OMG)
 - ✦ **AMQP (RabbitMQ)** (Pivotal)*
- **Memória Compartilhada Distribuída**
 - ✦ Java TupleSpaces (Oracle)
 - ✦ **CSpaces***

19

19

Entidades Comunicantes e Paradigmas de Comunicação

Entidades Comunicantes		Paradigmas de Comunicação		
Orientadas a Sistemas	Orientadas a Problemas	IPC	Inovação Remota	Comunicação Indireta
Processos	Objetos Distribuídos	Trocas de Mensagens (por sockets)	Requisição-Resposta	<i>Publish-Subscribe</i>
	Componentes Distribuídos		RPC	Comunicação em Grupo
	Serviços		RMI	Espaço de Tuplas

20

20



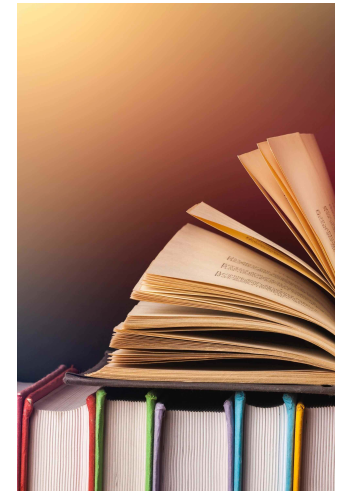
Exercício AVA

Middleware

21

Bibliografia

Steel, Tanenbaum, "**Distributed Systems**"
G. Coulouris et al. "**Distributed System:
Concepts and Design**", 5th edition,
Addison Wesley, 2012.



22

22