

Arquiteturas de Sistemas Distribuídos

- Sistema distribuído
 - O processamento de informações é distribuído em vários computadores ao invés de confinado em uma única máquina.
 - Bastante comum em qualquer organização
- Estilos de arquitetura comuns
 - Arquiteturas de múltiplos processadores
 - Arquiteturas cliente-servidor
 - Arquiteturas de objetos distribuídos
 - Computação interorganizacional

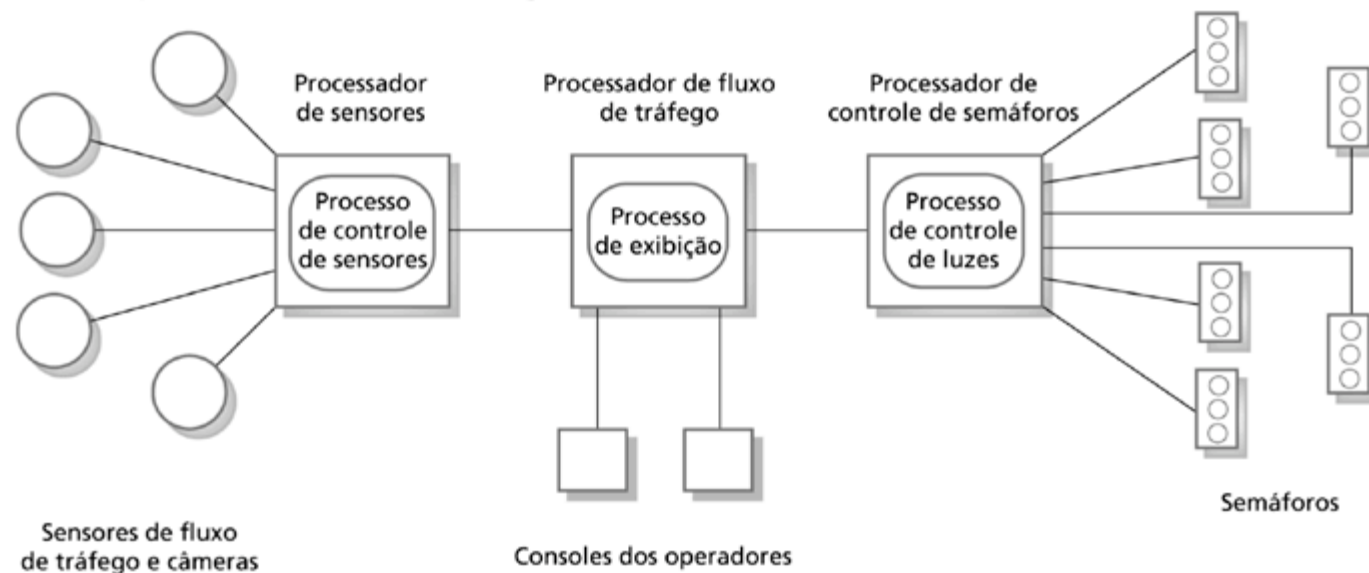
Características do sistema distribuído

- Compartilhamento de recursos
 - Compartilhamento de recursos de hardware e software.
- Interoperabilidade
 - Uso de equipamento e software de fabricantes diferentes.
- Concorrência
 - Processamento concorrente para aumentar o desempenho.
- Escalabilidade
 - Capacidade ampliada pela adição de novos recursos.
- Tolerância a falhas
 - A capacidade de continuar em operação após a ocorrência de uma falha.
- Complexidade
 - Tipicamente, sistemas distribuídos são mais complexos que sistemas centralizados.
- Segurança
 - Mais suscetível a ataques externos.
- Gerenciamento
 - Mais esforço é necessário para o gerenciamento do sistema.
- Imprevisibilidade
 - Respostas imprevisíveis dependendo da organização do sistema e da carga de rede.

Arquiteturas de multiprocessadores

- Sistema composto de múltiplos processos que podem (mas não precisam) executar em processadores diferentes.
- Distribuição de processo para processador pode ser predeterminada ou pode estar sob o controle de um *escalonador*.

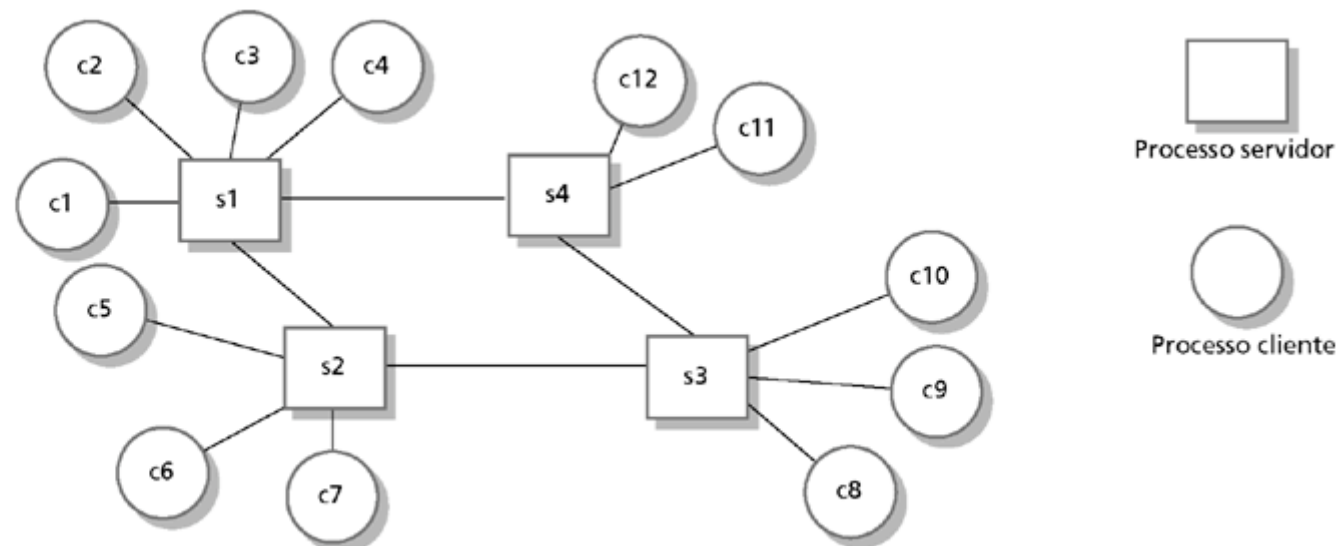
Figura 12.1 Sistema multiprocessador de controle de tráfego.



Arquiteturas cliente-servidor

- A aplicação é modelada como um conjunto de serviços que são fornecidos pelos servidores e um conjunto de clientes que usam estes serviços.
- Os clientes sabem da existência dos servidores mas os servidores não sabem dos clientes.

Figura 12.2
Sistema
cliente-servidor.



Exemplos de arquitetura cliente-servidor

Figura 12.3

Computadores em uma rede cliente-servidor.

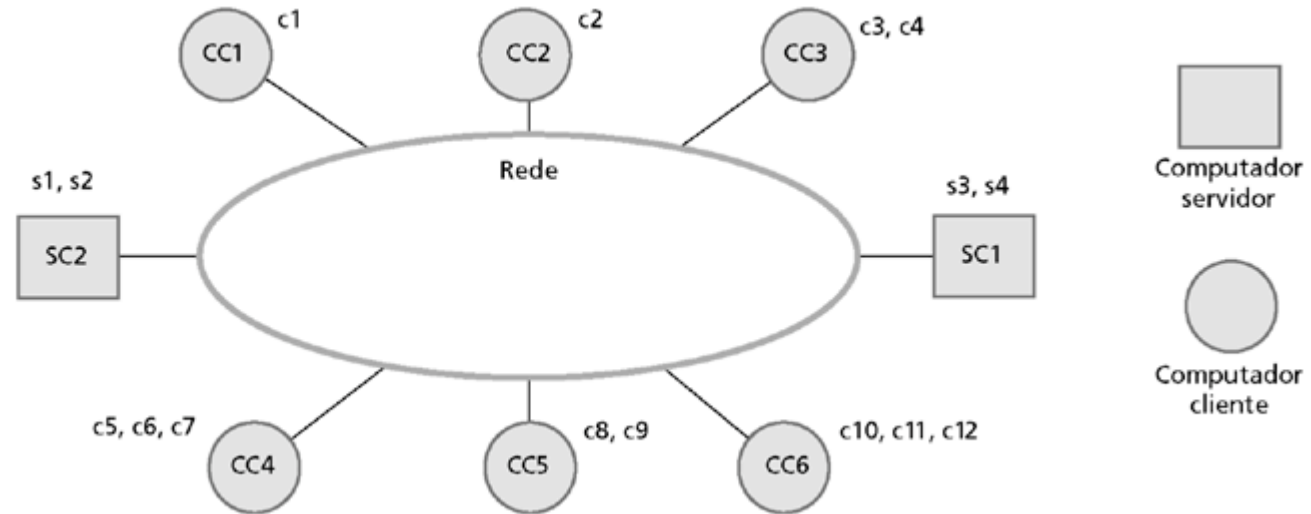
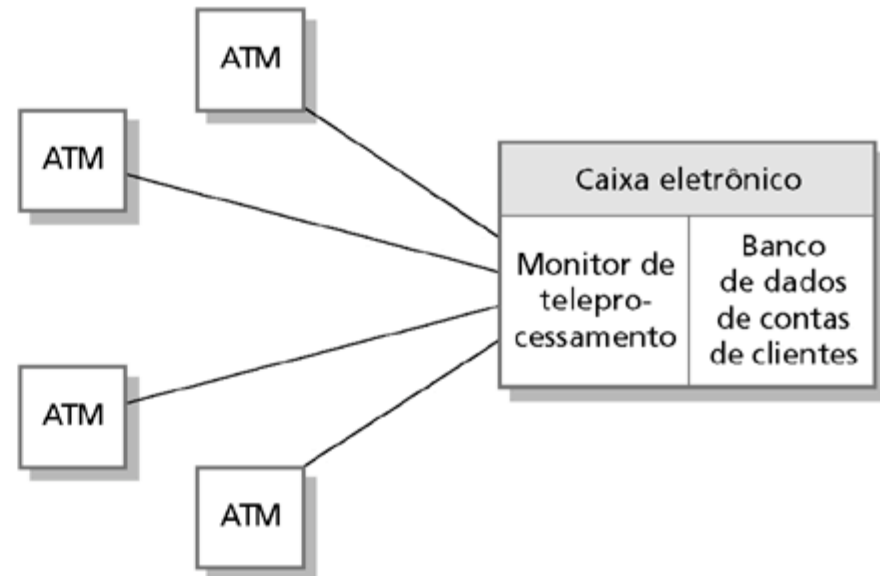


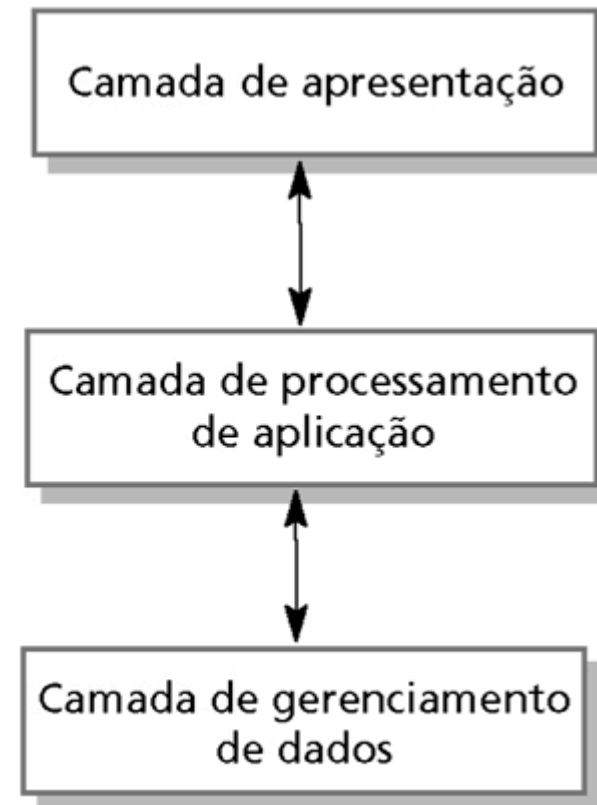
Figura 12.6

Sistema de caixa eletrônico cliente-servidor.



Arquitetura C-S em camadas

- Camada de apresentação
 - Está relacionada à apresentação dos resultados de um processamento para os usuário do sistema, e à coleta de entradas do usuário.
- Camada de processamento de aplicação
 - Está relacionada ao fornecimento de funcionalidade específica da aplicação, por exemplo, em um sistema de banco, funções bancárias, tais como abrir conta, fechar conta, etc.
- Camada de gerenciamento de dados
 - Está relacionada ao gerenciamento do banco de dados do sistema.

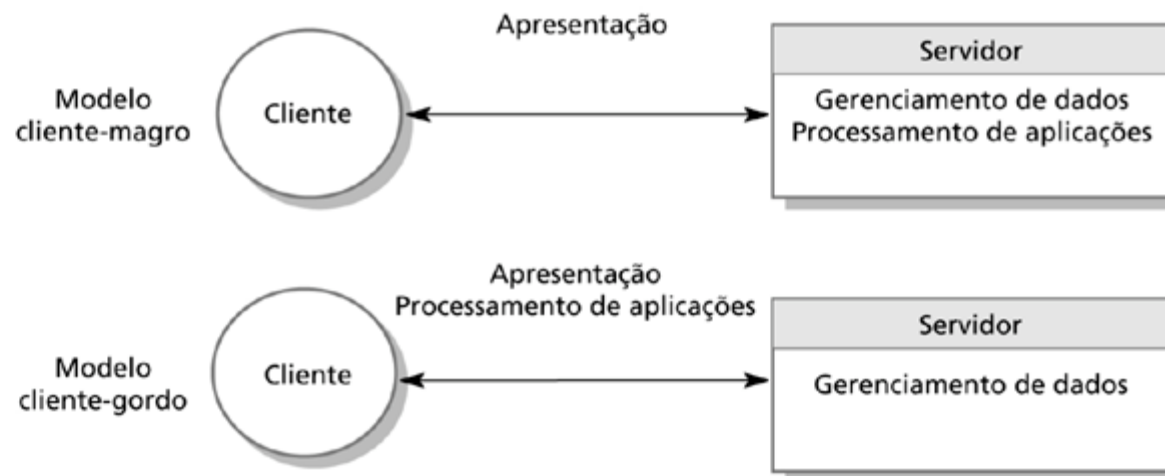


Clientes magros e gordos

- Modelo de cliente-magro
 - Em um modelo cliente-magro, todo o processamento de aplicação e o gerenciamento de dados é realizado no servidor. O cliente é responsável, simplesmente por executar o software de apresentação.
- Modelo de cliente-gordo
 - Nesse modelo, o servidor é responsável somente pelo gerenciamento de dados. O software do cliente implementa a lógica da aplicação e as interações com o usuário do sistema.

Figura 12.5

Clientes magros e gordos.



Arquiteturas de três camadas

- Podem executar em máquinas separadas
- Escalabilidade e Interoperabilidade

Figura 12.7

Arquitetura cliente-servidor de três camadas.

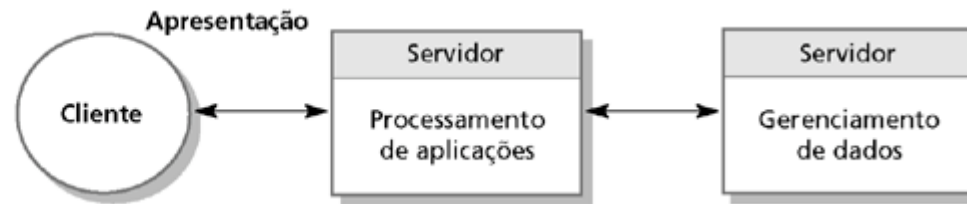
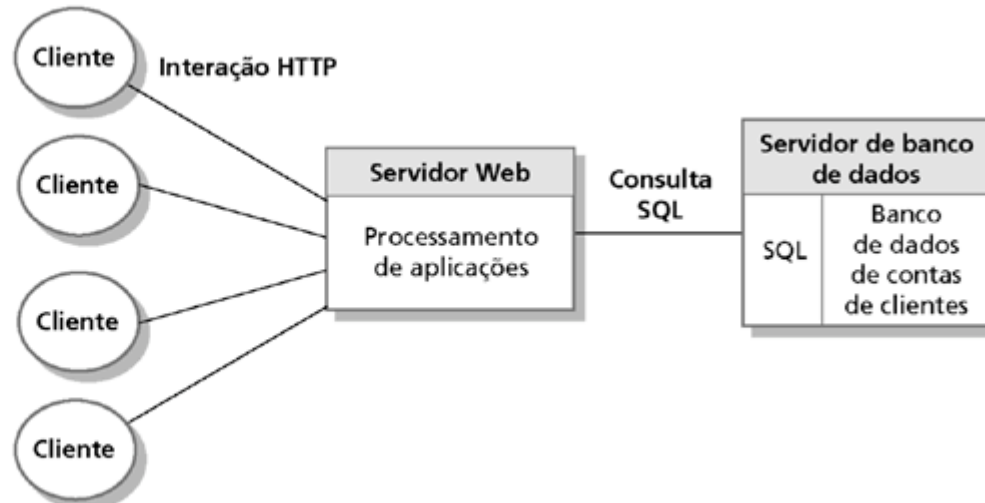


Figura 12.8

Arquitetura de distribuição de um sistema de operações bancárias pela Internet.

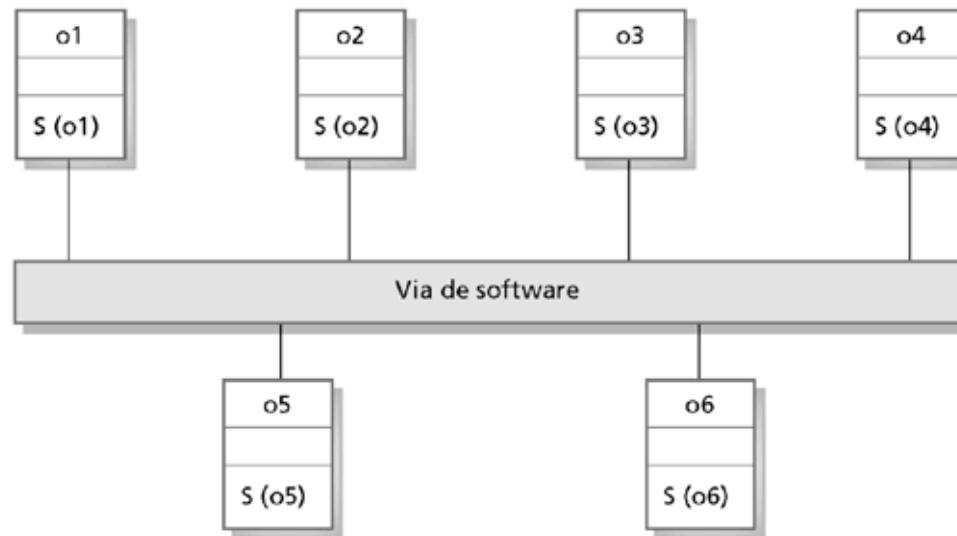


Arquiteturas de objetos distribuídos

- Não existe distinção entre clientes e servidores em uma arquitetura de objetos distribuídos.
- Cada entidade distribuível é um objeto que fornece serviços para outros objetos e recebe serviços de outros objetos.
- Os objetos se comunicam através de um sistema de middleware chamado requisitor de objetos.

Figura 12.9

Arquitetura de objetos distribuídos.

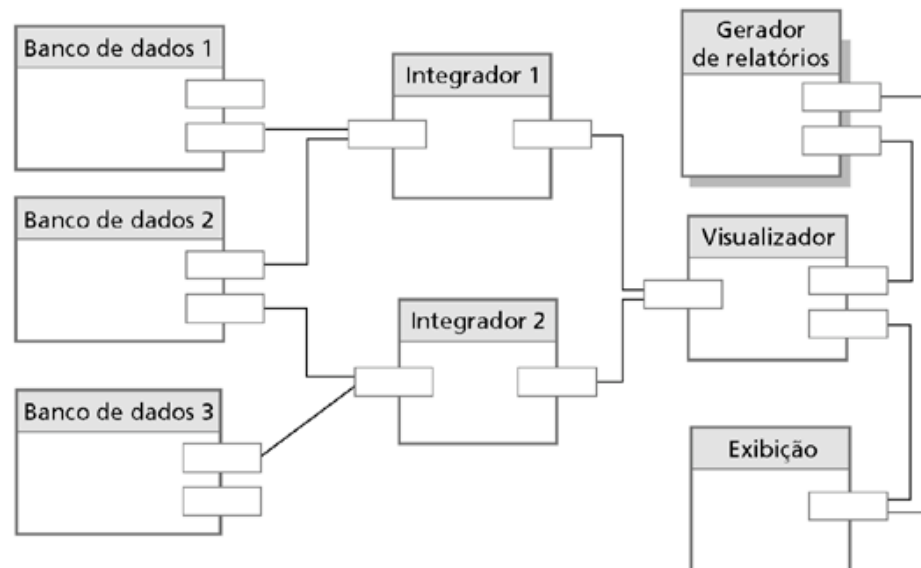


Exemplo: mineração de dados

- O modelo lógico do sistema não é um dos fornecimentos de serviço onde existem serviços distintos de gerenciamento de dados.
- Permite que uma série de banco de dados seja acessada para serem aumentados sem interrupção do sistema.

Figura 12.10

Arquitetura de distribuição de um sistema de mineração de dados.

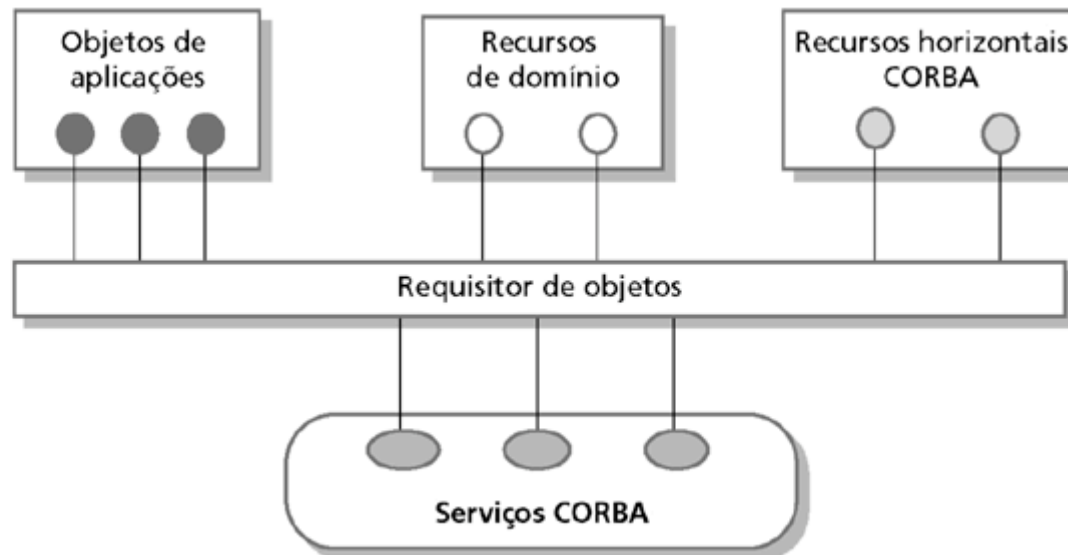


Middleware CORBA

- Middleware
 - Software que gerencia e apóia os componentes diferentes de um sistema distribuído. Em essência, ele se situa no *meio* do sistema.
- CORBA
 - Padrão internacional para um localizador de objetos (*Object Request Broker*) para gerenciar a comunicação entre objetos distribuídos.

Figura 12.11

Estrutura de uma aplicação distribuída baseada em CORBA.

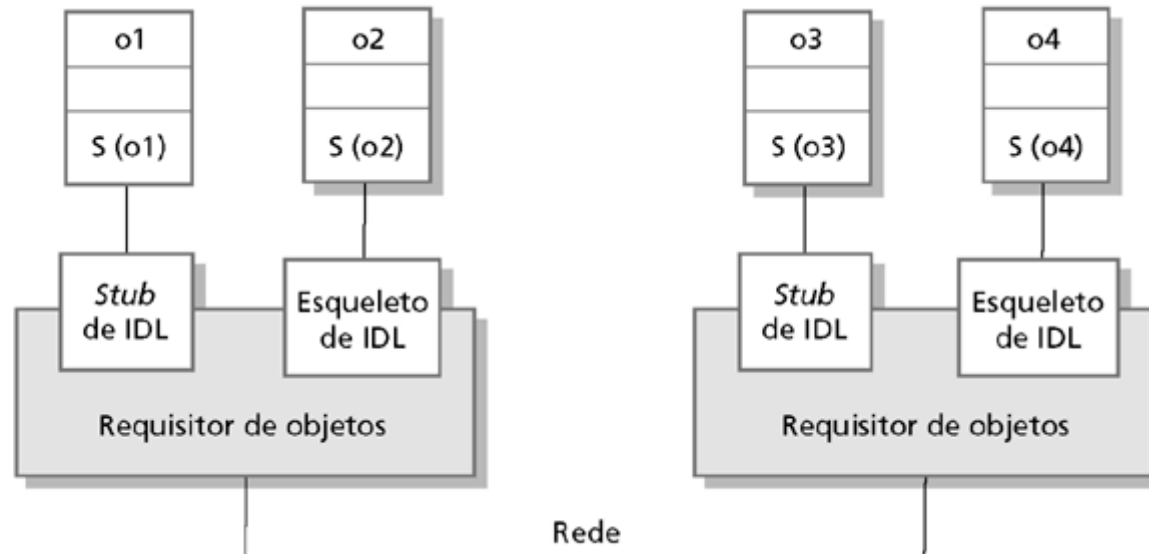


Requisitor e Corretor de Objetos

- O ORB manipula comunicações entre objetos. Ele sabe de todos os objetos no sistema e suas interfaces.
- ORBs são um conjunto de objetos em uma biblioteca que são ligadas a uma aplicação quando ela é desenvolvida.
- Usando um ORB, o objeto chamador liga um *stub* IDL que define a interface do objeto chamado.
- O chamado desse stub resulta em chamadas para o ORB, que então chama um objeto requisitado por meio de um esqueleto IDL publicado que liga a interface à implementação de serviços.

Figura 12.13

Comunicações inter-ORBs.

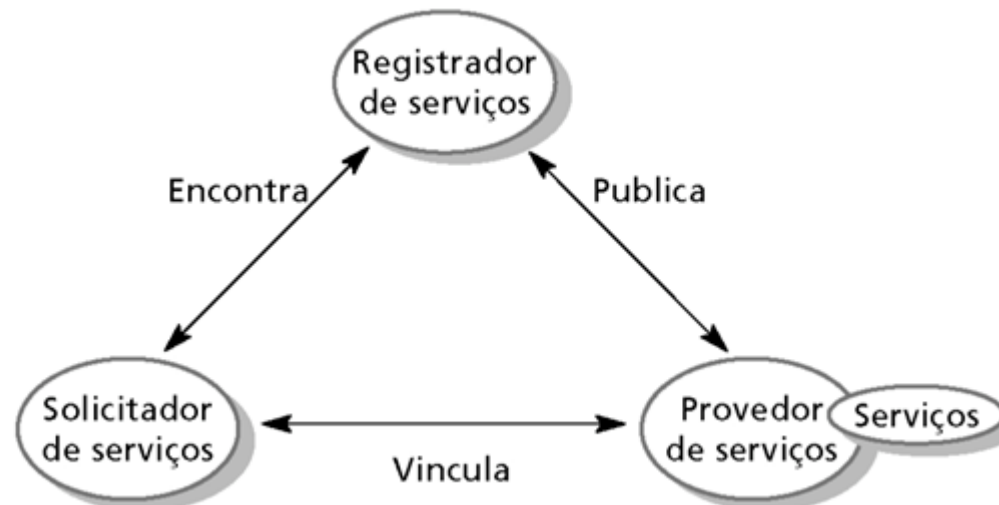


Arquiteturas orientadas a serviços

- Serviços
 - “Uma ação ou desempenho oferecido de um grupo para um outro. Embora o processo possa estar ligado a um produto físico, o desempenho é essencialmente intangível e não resulta normalmente em propriedade de algum dos fatores de produção”.
 - O fornecimento dos serviços é, portanto, independente da aplicação que usa o serviço.
 - Um *web service* é uma abordagem padronizada para tornar um componente reusável disponível e acessível através da rede

Figura 12.16

A arquitetura conceitual de um sistema orientado a serviços.



Exemplo

Figura 12.17

Sistema de informações de automóvel baseado em serviços.

Um sistema de informações de um automóvel fornece aos motoristas informações sobre o clima, condições de tráfego, informações locais, etc. O sistema é ligado ao aparelho de rádio do automóvel que apresenta as informações como um sinal de emissora de rádio específica.

O automóvel é equipado com um receptor GPS para informar sua posição e, baseado nessa posição, o sistema acessa uma gama de serviços de informações. A informação pode ser fornecidas no idioma especificado pelo motorista.

