

Conceitos do Cliente

Funcionamento Básico do Cliente

Iniciando o ORB

- *O ORB provê a comunicação entre o cliente e o servidor.*
- *Quando o cliente faz um pedido, o ORB localiza a implementação do objeto requisitado, faz a ativação do objeto, se necessário, entrega o pedido ao objeto, e retorna a resposta ao cliente.*

Iniciando o ORB

- *O cliente não está ciente se o objeto está sobre a mesma máquina ou através da rede.*
- *O ORB é um usuário intensivo de recursos do sistema.*
- *Uma única instância do ORB é criada.*
- *Embora muito do trabalho feito pelo ORB seja transparente para nós, o programa-cliente deve, explicitamente, iniciar o ORB.*

Iniciando o ORB

- *Opções do ORB podem ser especificadas através de argumentos.*
- *Portanto, devemos passar argumentos ao ORB.init para garantir que essas opções tenham efeito.*
- *Amostra de código.*

Ligando a Objetos

- *Um programa-cliente somente invoca um objeto remoto, por obter uma referência ao objeto. Referências são geralmente obtidas usando o método `bind()` da interface `Helper` (`Visibroker`) ou através de um serviço de nomes.*

Ligando a Objetos

- *O ORB oculta a maior parte dos detalhes envolvidos na obtenção da referência ao objeto, tais como localizar o servidor que implementa o objeto e estabelecer uma conexão ao servidor.*

O processo de Ligação

- *Quando o processo-servidor inicia, ele executa um `ORB.init()` e anuncia ele próprio ao ORB, a presença dele na rede.*
- *O programa-cliente localiza o ORB, em uma máquina qualquer da rede.*
- *O ORB pode estar localizado numa das máquinas da rede, que não é a máquina do programa-cliente, nem a máquina do servidor.*

Funções

- Quando encontra o ORB, o programa-cliente invoca o método `bind()`, e o ORB realiza diversas funções em nome do programa-cliente.
- O ORB localiza a implementação do objeto que oferece a interface requisitada.
- Um nome de objeto pode ser especificado no método `bind()`, e neste caso, quando este método é invocado, esse nome é usado na localização do objeto.

Funções

- *Quando a implementação do objeto é localizada, o ORB tenta realizar uma conexão da implementação do objeto localizado, com o programa-cliente.*
- *Uma vez a conexão seja estabelecida, o ORB cria um objeto-proxy e retorna uma referência para aquele objeto.*

Funções

- *O programa-cliente invocará métodos sobre o objeto-proxy, o qual, por sua vez, interagirá com o objeto servidor, para executar a implementação do objeto localizada.*
- *Ver figura ...*

Funções

- *O programa-cliente nunca invocará o método construtor sobre uma classe do objeto servidor. Ao invés disto, uma referência ao objeto é obtida por invocar o método estático bind().*

Invocando métodos sobre um objeto remoto

- *O programa-cliente usa uma referência a objeto para invocar um método sobre um objeto remoto ou para referenciar dados contidos pelo objeto.*
- *Amostra de código ...*

Manipulando Referências a Objetos

- *Existem algumas maneiras pelas quais uma referência a objeto pode ser manipulada:*
 - (a) Convertendo uma referência para string.*
 - (b) Convertendo um string para uma referência.*

Manipulando Referências a Objetos

- (c) Obtendo nomes de interfaces, de objetos e o identificador do tipo de repositório associado com uma referência a objeto.*
- (d) Determinando o tipo de uma referência a objeto:
Uma referência a objeto pode ser de um particular tipo.*

Manipulando Referências a Objetos

(e) Determinando a localização e a situação de objetos ligados.

Dado uma referência, o programa-cliente pode determinar se o objeto é ligado ou não, e se o programa-cliente e a implementação do objeto residem no mesmo processo ou mesmo espaço de endereços.

Manipulando Referências a Objetos

(f) *Narrowing* referências a objetos.

Convertendo um tipo de referência de um super-tipo para um tipo mais específico.

(g) *Widening* referências a objetos.

Convertendo de um tipo de referência mais específico para um super-tipo.