

Aula 5

CORBA - IDL



CORBA - IDL

→ Definição:

- Interface Definition Language é uma linguagem para descrever interfaces de serviços em ambientes heterogêneos. A interface especifica um nome e um conjunto de métodos para que os clientes possam acessar remotamente.
- O arquivo que contém as definições gera através de compilação específica os módulos auxiliares para a implementação dos objetos remotos e do cliente.

CORBA - IDL

- Mapeamento CORBA-IDL para Java
 - Gerais
 - module --> package
 - user exceptions --> org.omg.CORBA.UserException
 - system exceptions --> org.omg.CORBA.SystemException
 - parâmetros **in** --> parâmetros Java
 - parâmetros **in** e **inout** --> classes Java Holder
 - typecasts (narrow) --> classes Java Helper
 - attributes --> métodos de acesso (get e set)

CORBA - IDL

- Mapeamento CORBA-IDL para Java
 - Tipos Primitivos
 - char, wchar --> char
 - octet --> byte
 - string, wstring --> java.lang.String
 - short, unsigned short --> short
 - long, unsigned long --> int
 - long long, unsigned long long --> long
 - float --> float
 - double --> double

CORBA - IDL

- Mapeamento CORBA-IDL para Java
 - Tipos Construídos
 - interface --> interface
 - sequence --> array
 - array --> array
 - struct --> classe Java com a mesmo nome e estrutura
 - enum --> classe Java com o mesmo nome e tipo enum
 - union --> classe Java com o mesmo nome e métodos get/set
 - typedef --> não tem equivalente, mapeado para tipos comuns
 - any --> org.omg.CORBA.Any

CORBA - IDL

→ Mapeamento CORBA-IDL para Java

→ Pseudo-Objetos

- CORBA::ORB --> org.omg.CORBA.ORB
- CORBA::Object --> org.omg.CORBA.Object
- CORBA::NamedValue --> org.omg.CORBA.NamedValue
- CORBA::NVList --> org.omg.CORBA.NVList
- CORBA::Request --> org.omg.CORBA.Request
- CORBA::ServerRequest --> org.omg.CORBA.ServerRequest
- CORBA::TypeCode --> org.omg.CORBA.TypeCode
- CORBA::TCKind --> org.omg.CORBA.TCKind

CORBA - IDL

→ Modules

```
module Loja {  
    ...  
};
```

→ Exceptions

```
exception ErroCasual {  
    string explicacao;  
};
```

→ Parâmetros

```
interface Loja {  
    void quantProdutos (out long quant) raises (ErroCasual);  
    void setEstoque (in short codigo, in short quant) raises (ErroCasual)  
};
```



CORBA - IDL

→ Attribute

```
attribute short quantidade;  
readonly attribute long estoque;
```

→ Constantes

```
interface TiposConstantes {  
    const long min_valor;  
    const long max_valor;  
}
```

→ Sequence

```
typedef sequence <short> ListaProdutos; // não limitada  
typedef sequence <short, 10> ListaProdutosLimitada; // limitada  
void baixaProdutos ( in ListaProdutos lista);
```


CORBA - IDL

→ Array

```
typedef short ListaProdutos[10];  
void baixaProdutos ( in ListaProdutos lista);
```

→ Struct

```
struct Produto {  
    short codigo;  
    string descricao;  
    short quant;  
}
```

→ Enum

```
enum MyEnum {yellow, red, blue};
```

CORBA - IDL

→ Union

```
union Animal switch (short) {  
    case 1: Dog dog;  
    case 2: Cat cat;  
    default: Animal other;  
};
```

Exercícios

- Crie uma aplicação básica CORBA com um pedido a um objeto remoto e sua respectiva resposta.
- Use IDL para implementar uma interface para os métodos da aplicação desenvolvida nas aulas sobre JDBC.
-

Para saber mais

→ Introduction to Corba

<http://java.sun.com/developer/onlineTraining/corba/corba.html>

→ Java IDL Documentation

<http://java.sun.com/j2se/1.4.2/docs/guide/idl/>

→ Corba: Architecture and Specification

<http://www.omg.org/docs/formal/99-10-07.pdf>

→ Tutorial sobre Corba

http://www.gta.ufrj.br/grad/00_2/corba/

