

CORBA

Cvičení č.10

Úvod

- Filozofie CORBA
 - Nezávislý implementační jazyk
 - Nezávislý komunikační protokol
 - Nezávislý operační systém

Datové typy

- boolean
- char
- octet
- short
- long
- long long
- float
- double
- string
- enum
- struct
- union
- any
- in
- out
- inout

Příklad - IDL

```
module Remote Time {  
  interface My Remote Time {  
    string get Time ();  
  };  
};
```

Pro klientskou aplikaci jsou vytvořeny tyto třídy:

- My RemoteTime.java obsahuje rozhraní definované v IDL.
- MyRemoteTimeStub.java
- MyRemoteTime.java je třída poskytující metody:
 - extract(),insert() sloužící k zapouzdření objektu do datového typu any.
 - narrow() jenž slouží k přetypování na stejný typ jako je daná třída
 - bind()
- MyRemoteTimeHolder.java obsahuje třídy umožňující předat objekt referencí

Příklad - čas

Pro stranu serveru jsou pak vygenerována třída:

- `_MyRemoteTimeImplBase.java` což je abstraktní třída pro vzdálený objekt, z které se musí zdědit nová třída a implementovat její abstraktní metody.

Implementace metod vzdálených objektů.

Vytvoříme novou třídu `MyRemoteTimeImpl` zděděním z abstraktní třídy `_MyRemoteTimeImpl`.

- V metodě `main` pak provedeme:
- vytvoření instance ORBu
- vytvoření instance vzdáleného objektu
- získání reference na jmenný kontext, ve kterém objekt zaregistrujeme
- registraci objektu `MyRemoteTime`

Příklad - server

```
import RemoteTime.*;
import org.omg.CosNaming.*;
import org.omg.CosNaming.NamingContextPackage.*;
import org.omg.CORBA.*;
import java.util.*;
import java.text.*;

class MyRemoteTimeImpl extends _MyRemoteTimeImplBase {
    public String getTime(){
        return new Date().toString();
    }
}
```

Příklad - server

```
public class RemoteTimeServer{
    public static void main (String args[]){
        try{
            ORB orb = Orb.init(args,null);
            MyRemoteTimeImpl server = new MyRemoteTimeImpl();
            orb.connect(server);
            org.omg.CORBA.Object objRef = orb.resolve_initial_reference("Name Service");
            NamingContext ncRef = NamingContextHelper.narrow(objRef);
            NameComponent nc = new NameComponent ("MyRemoteTime","");
            NameComponent path [] = {nc};
            ncRef.rebind (path, server);

            java.lang.Object sync = new java.lang.Object();
            synchronized (sync) {
                sync.wait();
            }
        }catch (Exception e ) {//...}
    }
}
```

Příklad - klient

```
public class RemoteTimeClient {
    public static void main (string args[]) {
        try {
            Orb orb = ORB.init (args,null);
            org.omg.CORBA.Object objRef = orb.resolve_initial_reference("Naming Service ");
            NamingContext ncRef = NamingContextHelper.narrow(objRef);
            NameComponent nc = new NameComponent("MyRemoteTime","");
            NameComponent path[] = {nc};
            MyRemoteTime time = MyRemoteTimeHelper.narrow (ncRef.resolve(path));
            String nTime = time.getTime();
            System.out.println(nTime);
        } catch (Exception e) {//}
    }
}
```