

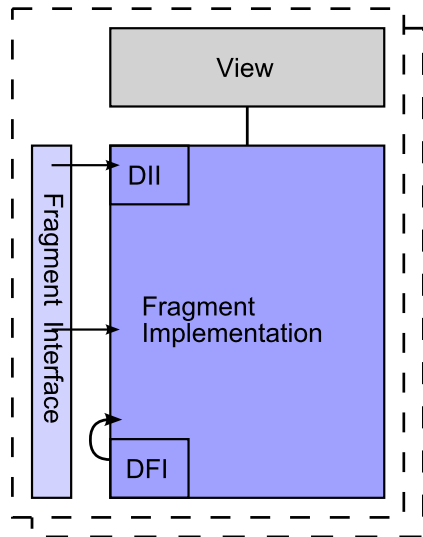
AVO Übung 4

AspectIX

18. Januar 2005 (WS 2004)

Andreas I. Schmied (schmied@inf...)

1 Wiederholung AspectIX-Grundlagen



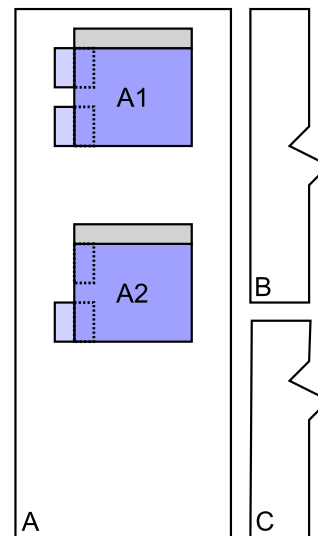
Beispiel:
3 Prozesse

Prozess A:

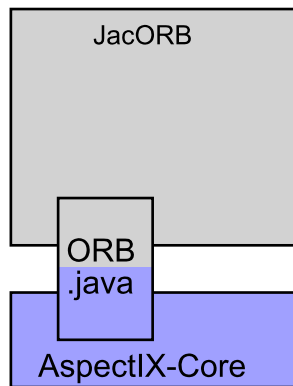
FI Impl A1 mit
zwei FI fcs
(z.B. nach 2. Cast)

FI Impl A2 mit
Default View und
einem FI fcs

Fragmente in
Prozessen B, C
nicht dargestellt



2 Technische Realisierung



2.1 Technische Realisierung – Integration in JacORB

Die ORB-Klasse (... extends org.omg.CORBA.ORB)

```
public final class ORB extends org.jacorb.orb.ORB {  
  
    CreationService cs = new CreationServiceImpl(this);  
  
    String object_to_string (org.omg.CORBA.Object obj) {  
        return cs.IORToString(cs.objectToIOR(obj));  
    }  
  
    org.omg.CORBA.Object string_to_object (String url) {  
        org.omg.IOP.IOR ior=cs.stringToIOR(url);  
        return cs.IORToObject(ior);  
    }  
  
    org.omg.CORBA.Object resolve_initial_references (String identifier);  
}
```

2.2 Technische Realisierung – Projekt-Beispiele

Wir betrachten das Beispiel `simple-apx-server` aus der Distribution.

Entwicklungsprozess

1. Schnittstelle in IDL konzipieren
2. Schnittstelle in Java übersetzen mit `IDLflex`: **APX-Mapping**
3. Implementierungen, Clients erstellen
4. Übersetzen des geschriebenen und generierten Codes

1. Schnittstelle in IDL konzipieren

Datei `apx-test.idl`

```
#include "aspectix_orb.idl"

module apxtest2 {

    exception NO_SUCH_ELEMENT {};

    interface simpledb : org::aspectix::orb::APXObject {
        string getInfo(in string key) raises (NO_SUCH_ELEMENT);
    };
};
```

2.2 Technische Realisierung – Projekt-Beispiele

2. Schnittstelle übersetzen mittels APX-Mapping

```
<property name="idlflex.jvmarg" value="
  -Dorg.omg.CORBA.ORBClass=JavaORB.CORBA.ORB
  -Dorg.omg.CORBA.ORBSingletonClass=JavaORB.CORBA.ORBSingleton
  -DAspectixBase=../../aspectix-core/ "/>

<property name="idlflex.arg" value="-DPOA -DWRITEPATH=${basedir}/${srcGenDir}"/>

<java fork="yes" classname="org.aspectix.IDLflex.IDLflexMain">
  <jvmarg line="${idlflex.jvmarg}"/>
  <arg line="${idlflex.arg}"/>
  <arg value="-mAPX/APXJavaMapping.xml"/>
  <arg value="-i../../aspectix-core/idl"/>
  <arg value="-I::apxtest2"/>
  <arg value="idl/apx-test2.idl"/>
</java>
```

- Ausschnitt aus ANT-Buildfile
- java-Task ruft IDLflex auf
 - APXJavaMapping ausgewählt für Interface ::apxtest2

2.2 Technische Realisierung – Projekt-Beispiele

3.a Client erstellen

Datei SimpledbClient.java

```
BufferedReader br = new BufferedReader(new FileReader("apxtest2.ior"));  
String s = br.readLine();  
  
ORB orb = ORB.init(args, null);  
org.omg.CORBA.Object o = orb.string_to_object(s);  
  
Simpledb h = SimpledbHelper.narrow( o );  
  
System.out.println( h.getInfo( "Test1" ) );
```

2.2 Technische Realisierung – Projekt-Beispiele

3.b Implementierungen erstellen (1)

Datei SimpledbFragment.java - Basisimplementierung

```
public class SimpledbFragment extends _SimpledbFragment {  
  
    private Hashtable ht = new Hashtable();  
  
    public SimpledbFragment(org.aspectix.orb.View view) {  
        super(view);  
        ht.put("Test1", "data1");  
    }  
  
    public String getInfo(String key) throws NO_SUCH_ELEMENT  
    {  
        String s = (String) ht.get(key);  
        if (s==null) throw new NO_SUCH_ELEMENT(); else return s;  
    }  
}
```

2.2 Technische Realisierung – Projekt-Beispiele

3.b Implementierungen erstellen (2)

Datei SimpledbServerFragment.java

```
public class SimpledbServerFragment extends _SimpledbFragment {

    public org.omg.CORBA.Object me;

    public SimpledbServerFragment(CreationService ics, APXProfileManager mgr)
    {
        org.omg.IOP.IOR ior = ics.createNewIOR("IDL:apxtest/simpledb:1.0");
        org.omg.IOP.IORHolder iorh = new org.omg.IOP.IORHolder(ior);
        mgr.insertProfile(iorh, "apxtest2.SimpledbFragment", new String[0]);
        this.prepareBlessedIOR(iorh);
        ior = iorh.value;
        me = mgr.blessFragment(ior, this);
        view = APXObjectHelper.narrow(me).get_view();
    }

    public void prepareBlessedIOR(org.omg.IOP.IORHolder ior) { ... }
    public void confirmBlessedIOR(org.omg.IOP.IOR ior) { ... }

    public org.omg.CORBA.Object getObject() { return me; }

    // Rest wie SimpledbFragment
}
```

2.2 Technische Realisierung – Projekt-Beispiele

Datei SimpledbCreateFragment.java

```
...  
  
org.omg.CORBA.ORB orb = org.omg.CORBA.ORB.init(args, null);  
  
CreationService ics = CreationServiceHelper.narrow(  
    orb.resolve_initial_references("CreationService") );  
  
APXProfileManager mgr = (APXProfileManager)  
    ics.getProfileManager(APX_PROFILE_TAG.value);  
  
SimpledbServerFragment frag = new SimpledbServerFragment(ics, mgr);  
Simpledb obj = SimpledbHelper.narrow(frag.getObject() );  
  
org.omg.IOP.IOR ior = obj.get_view().getIOR();  
org.omg.IOP.IORHolder iorh = new org.omg.IOP.IORHolder(ior);  
  
PrintWriter pw = new PrintWriter(new FileWriter("apxtest2.ior"));  
pw.println( ics.IORToString(iorh.value) );  
pw.close();
```

2.2 Technische Realisierung – Projekt-Beispiele

Programme starten

```
<property name="tests.jvmarg" value =      "  
  -Dorg.omg.CORBA.ORBClass=org.aspectix.orb.ORB  
  -Dorg.omg.CORBA.ORBSingletonClass=org.aspectix.orb.ORB  
  -Dcustom.props=../aspectix.properties  "/>  
  
<java classname="apxtest2.SimpledbCreateFragment "  
  classpathref="classpath" fork="true" failonerror="true">  
  <jvmarg line="${tests.jvmarg}"/>  
</java>  
  
<java classname="apxtest2.SimpledbClient "  
  classpathref="classpath" fork="true" failonerror="true">  
  <jvmarg line="${tests.jvmarg}"/>  
</java>
```

- Ausschnitt aus ANT-Buildfile

2.3 Technische Realisierung – Das APX-Mapping

- Standard-CORBA: IOP-Mapping vgl. `idlj`
- APX-Mapping erzeugt AspectIX-spezifischen Interface-, Proxy-, Helper/Holder-Code

Schnittstellen (Core)

```
interface APXObject { // IDL
    View      get_view();
    APXObject duplicate_with_new_view();
};
```

```
class org.aspectix.orb.Fragment extends LocalObject {

    APXObject duplicate_with_new_view()      {...}
    Fragment  getFragOfType(String id)      {...}
    void      exchangeFragImpl(Fragment frag){...}
    View      get_view()                    {...}
    void      prepareBlessedIOR(IORHolder ior){...}
    void      confirmBlessedIOR(IOR ior)     {...}
}
```

```
class org.aspectix.orb.FragmentIfc implements CORBA.Object {
    View get_view()
    void _setFragImpl(Fragment frag)
}
```

2.3 Technische Realisierung – Das APX-Mapping

Schnittstellen (Simpledb)

```

interface SimpledbOperations extends org.aspectix.orb.APXObjectOperations {
    public String getInfo(String key) throws NO_SUCH_ELEMENT;
}

interface Simpledb extends SimpledbOperations,
                        org.aspectix.orb.APXObject, org.omg.CORBA.Object {}

interface SimpledbFragOpe extends org.aspectix.orb.APXObjectFragOpe {
    public String getInfo(String key)
        throws org.aspectix.orb.exc.ROLL_BACK_EXCEPTION, NO_SUCH_ELEMENT;
}

```

Fragment-Rumpf

```

class _SimpledbFragment extends Fragment implements SimpledbFragOpe {
    public _SimpledbFragment(org.aspectix.orb.View view) { super(view); }
    public static final String[] _ids_list = {
        "IDL:apxtest2/Simpledb:1.0" , "IDL:org/aspectix/orb/APXObject:1.0"
    };
}

```

2.3 Technische Realisierung – Das APX-Mapping

Stub

```
class _SimpledbFragmentIfc extends FragmentIfc implements Simpledb, CORBA.Object
{
    public String getInfo(String arg) throws NO_SUCH_ELEMENT {
        return ((SimpledbFragOpe)_FragImpl).getInfo(arg);
    }

    public APXObject duplicate_with_new_view() {

        // new View and FragIfc
        View view = new org.aspectix.orb.View(_View);
        _SimpledbFragmentIfc fragIfc = new _SimpledbFragmentIfc(view);

        // initial implementation for APXObject
        Fragment fragImpl = new org.aspectix.orb.Fragment(view);
        view.setFragImpl(fragImpl);

        return (APXObject) fragIfc;
    }
}
```

3 Aufgabenstellung

Aufgabenstellung

Entwickeln Sie einen **Chat-Dienst auf Basis von AspectIX-Fragmenten!**

- Chat-Software mittels Multicast-Sockets
 - Initiator-Programm erzeugt initiale IOR, legt MCast-Address fest
- UI-Client (text-basiert, swing, ...)
 - User-Thread: Eingabe und Versenden
 - Peer-Thread: Empfang und Ausgabe
- Wie kann man Standard-CORBA-Clients integrieren?

Vorbereitungen zum Projekt

- Download der AspectIX-Distribution (siehe AVO-Seite) inkl. IDLflex, Beispiele
- ANT- bzw. Eclipse-Konfiguration (siehe README)