Fractal

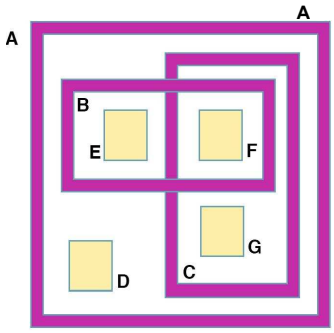
http://fractal.ow2.org

Overview

- General model defined by France Télécom R&D and INRIA (2002)
- Features
  - Few restrictions
  - Self-* capacity
    - observation,
    - control,
    - reconfiguration.
  - Open and adaptable :
    - Extra-functional services can be personalized

Fractal Component

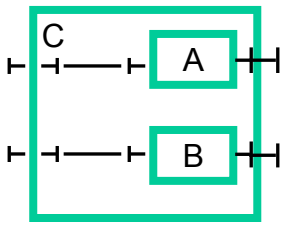
- Hierarchical model
  - Primitive components
  - Composite
- Component can be shared
- Two parts:
  - Contents
    - set of (functional) components
  - Membrane
    - Interfaces and controllers

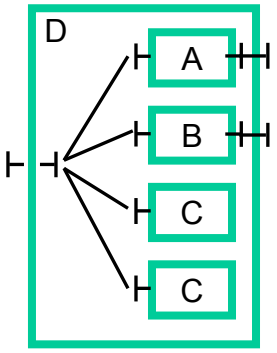


Fractal Components

- Primitive
  - Any language (OO)

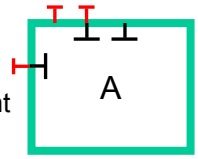
- Composite
  - Internal interfaces
  - Internal bindings
  - Imbrications



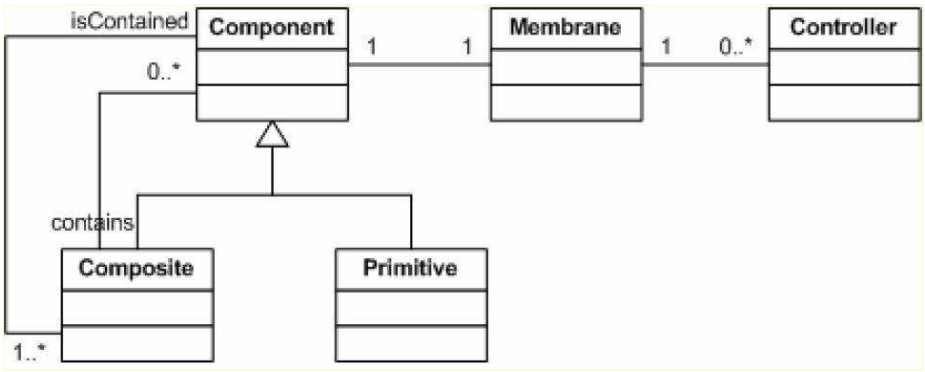


Fractal Component

- Functional interfaces
  - Access points (I/O)
  - Definition
    - Name, Signature (type), client/server, mandatory/optional, simple/multiple
- Controllers
  - Optional
  - Reside in the membrane
    - Used to be object, now can also be component
  - Accessible through a control interface
  - Examples
    - Introspection, binding, etc.



Fractal Component



Examples of Interfaces

```
interface Component {
   any[] getFcInterfaces ();
   any   getFcInterface (string itfName) …;
   Type  getFcType ();
}
interface Type {
   boolean isFcSubTypeOf (Type t);
}

interface Interface {
   string    getFcItfName ();
   Type      getFcItfType ();
   Component getFcItfOwner ();
   boolean   isFcInternalItf ();
}

interface GenericFactory {
   Component newFcInstance (Type t, any controllerDesc,
                            any contentDesc) …;
}
```

Examples of Controllers

```
interface AttributeController { }

interface BindingController {
   string[] listFc ();
   any      lookupFc (string clientItfName) …;
   void     bindFc (string clientItfName, any serverItf) …;
   void     unbindFc (string clientItfName)…;
}

interface ContentController {
   any[]       getFcInternalInterfaces ();
   any         getFcInternalInterface (string itfName)…;
   Component[] getFcSubComponents ();
   void        addFcSubComponent (Component c) …;
   void        removeFcSubComponent (Component c) …;
}
```

Level of conformance

	C	I	CT, IT	AC, BC, CC, LC	F	T
0						
0.1				X		
1	X					
1.1	X			X		
2	X					
2.1	X			X		
3	X	x				
3.1	X	X		X		
3.2	X	X		X	X	
3.3	x	X		X	x	x

C: Component I: Interface CT: ComponentType IT: InterfaceType
AC: Attribute BC: Binding CC: Content LC: LifeCycle
F: Factory T: Template

Fractal ADL

- Primitive component

```
<definition name="ClientImpl">
  <interface name="r" role="server"
             signature="java.lang.Runnable"/>
  <interface name="s" role="client" signature="Service"/>
  <content class="ClientImpl"/>
</definition>
```

Fractal ADL

- Composite component

```
<definition name="HelloWorld">
  <interface name="r" role="server"
             signature="java.lang.Runnable"/>
  <component name="client">
    <interface name="r" role="server"
               signature="java.lang.Runnable"/>
    <interface name="s" role="client" signature="Service"/>
    <content class="ClientImpl"/>
  </component>
  <component name="server">
    <interface name="s" role="server" signature="Service"/>
    <content class="ServerImpl"/>
  </component>
  <binding client="this.r" server="client.r"/>
  <binding client="client.s" server="server.s"/>
</definition>
```