



2010: An Acronym Odyssey

Brian O'Neill

(boneill@verilogue.com)

Introductions

Stanley Goodspeed: Well, I'm one of the those fortunate people who like my job, sir. Got my first chemistry set when I was seven, blew my eyebrows off, we never saw the cat again, been into it ever since.", The Rock, 1996

- s/Stanley Goodspeed/Brian O'Neill/g
- s/chemistry set/comput/g
- s/cat/dog/g

Hanging in the balance

➤ Is J2EE dead?



The Acronyms

- SOA: Service-Oriented Architecture
- ESB: Enterprise Service Bus
- JBI: Java Business Integration
- SCA: Service Component Architecture
- OSGi: (formerly known as...)
 - Open Services Gateway Initiatives

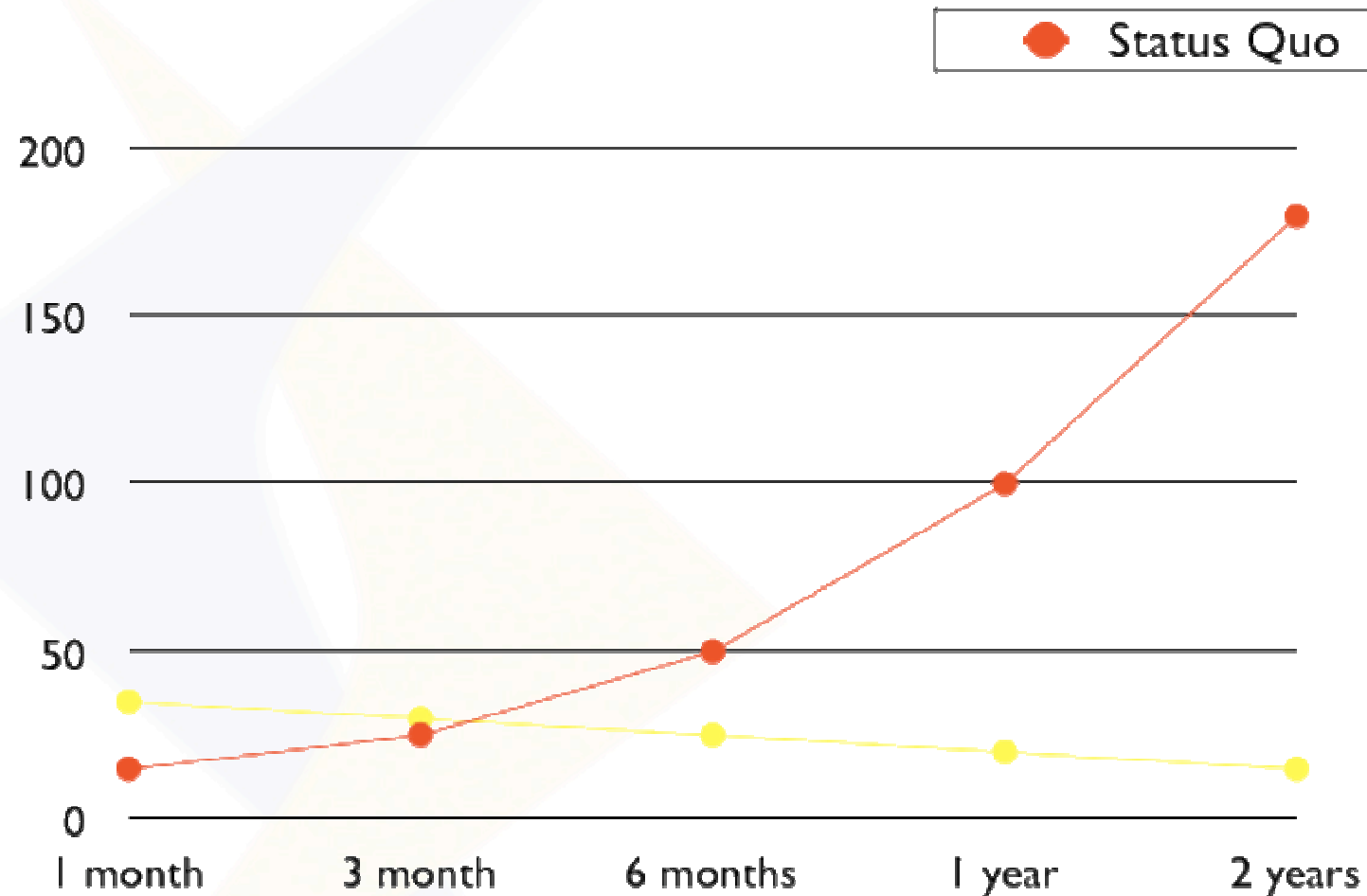
Service Oriented Architecture

- Holy Grail and/or a Snake Oil Cash Cow?
- Service Definitions
 - enabling...
- Rapid Composition
 - with...
- Interchangeable Components



*Loose
Coupling*

The Curves: Cost to Implement



Breaking that down...



- Rapid Composition & Interchangeable Parts
 - What is the cost and risk to integrate Consumer A w/ Service B?

Transport Protocol + *Message Format*

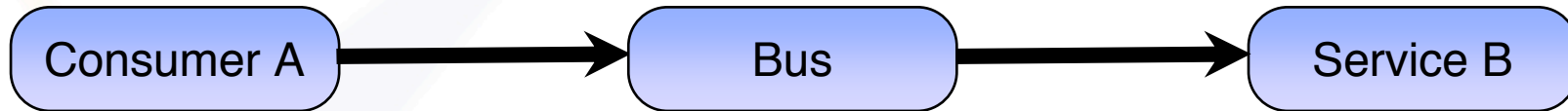
A quick digression...

➤ Service-Oriented \neq WS-*

➤ That said,

- HTTP is quickly becoming the de facto transport (though JMS still makes sense sometimes)
- XML, REST, JSON, & RSS, are good formats. (Don't just go buy into SOAP for SOAP's sake)

Enters the Enterprise Service Bus

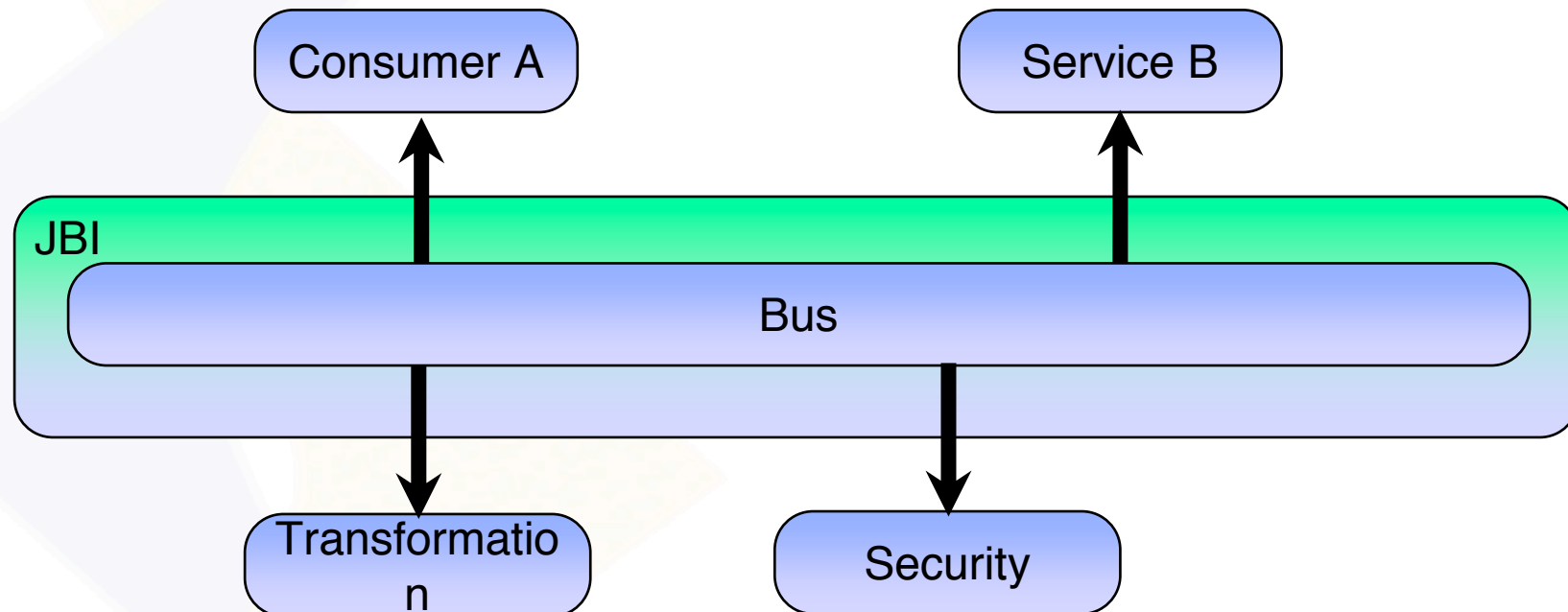


- Eliminate / isolate the transport and format concern
- Provide an intermediary between two endpoints
 - allows for re(routing) of a message
 - for security, transformation, auditing, compression, etc.
- Business process isolation
- Flexible deployment topologies
 - Dynamic Discovery, Single Point of Configuration

Java Business Integration (JBI)

- A standard produced by the Java Community Process (JCP), JSR 208, version 1.0
- Focused on rapid composition of capability by standardizing the interface to the ESB.
 - Consistent Meta-Data, Deployment and Management

The idea: Pluggable Lollipops



The skinny

➤ Supporting it:

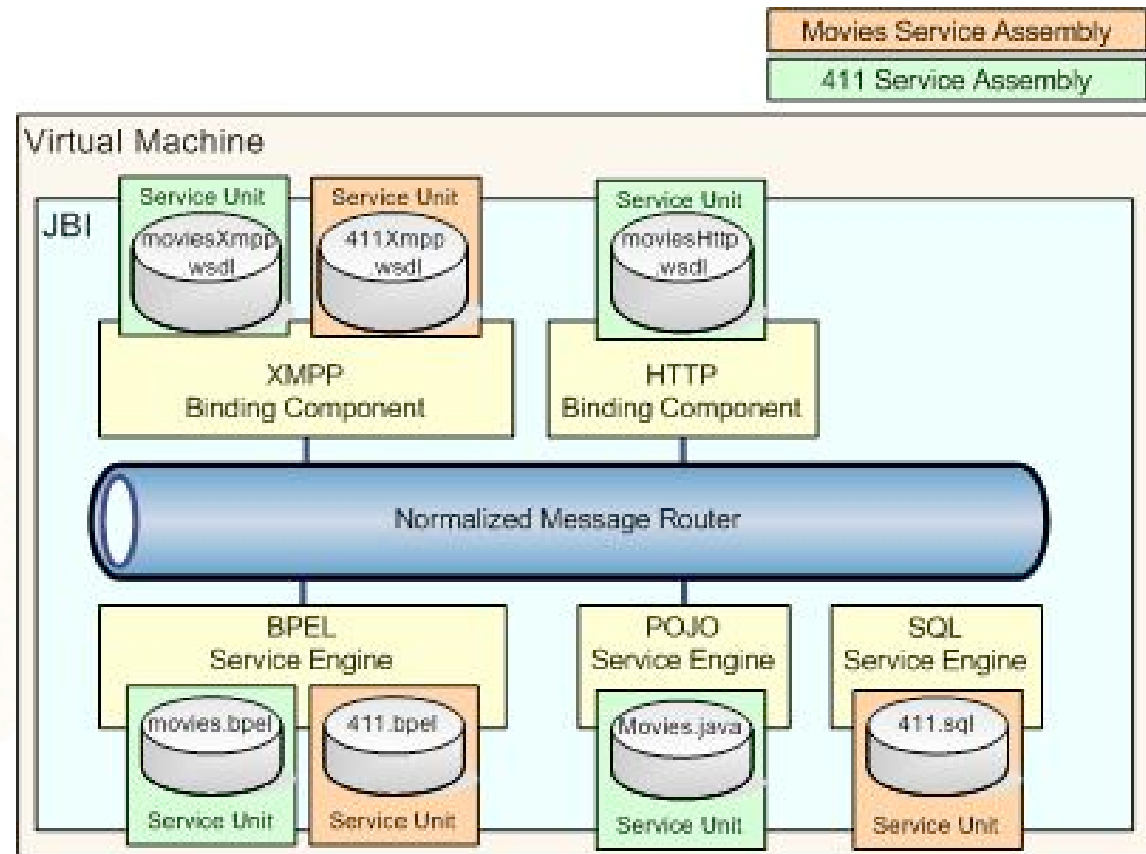
- Tibco, Novell, Sun, SAP, Red Hat, WebMethods, Oracle, RIM, Nokia, Sybase, Borland, and others

➤ Who's got it:

- Apache ServiceMix (LogicBlaze acquired by IONA)
- Java.Net OpenESB (Sun)
- ObjectWeb PEtALS (eBM Websourcing)
- Mule* (MuleSource)

The gritty

- Service Engines
- Binding Components
- Service Unit
- Service Assembly



Available Components: Prego & Presto

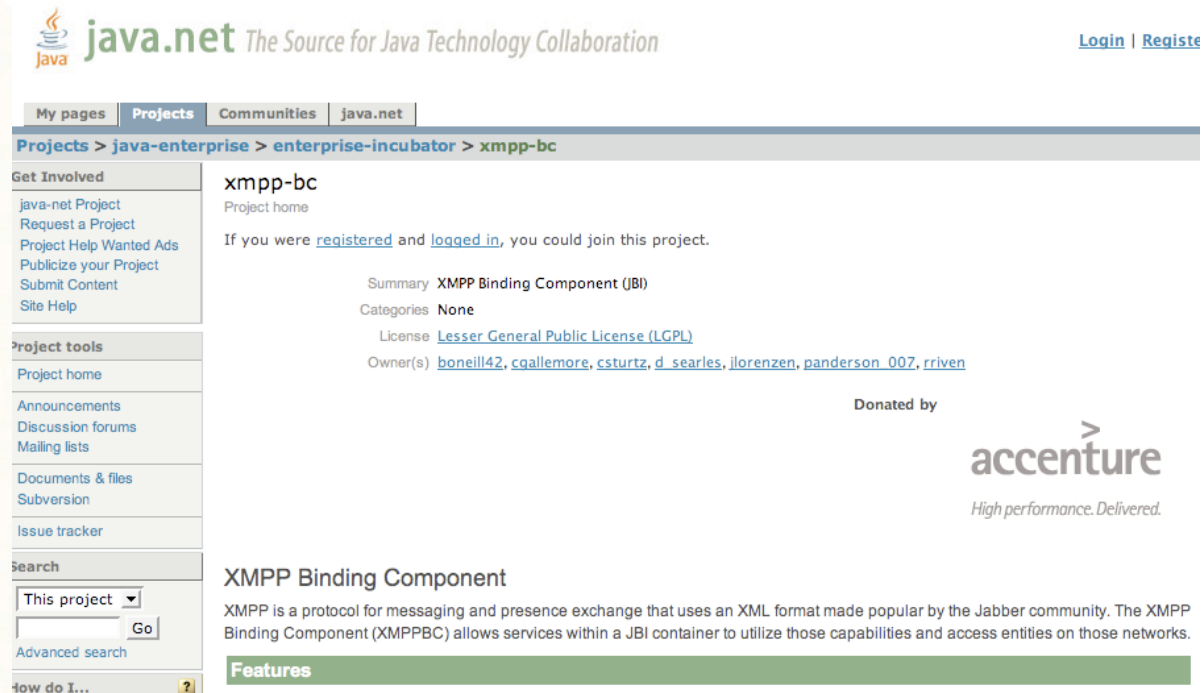
- BPEL, Drools, File, FTP, HTTP, JMS, JSR181, Quartz, Corba, SAP, SMTP, and...

➤ XMPP

➤ SIP

➤ RSS

➤ UDDI



The screenshot shows the Java.net website interface. At the top, the Java logo and "java.net The Source for Java Technology Collaboration" are visible, along with "Login" and "Register" links. Below the navigation bar, the breadcrumb trail reads "Projects > java-enterprise > enterprise-incubator > xmpp-bc". The left sidebar contains sections for "Get Involved" (with links like "java-net Project", "Request a Project", etc.), "Project tools" (with links like "Project home", "Announcements", etc.), and a "Search" box. The main content area is titled "xmpp-bc" and includes a "Project home" link. It states: "If you were [registered](#) and [logged in](#), you could join this project." Below this, it lists: "Summary: XMPP Binding Component (JBI)", "Categories: None", "License: [Lesser General Public License \(LGPL\)](#)", and "Owner(s): [boneill42](#), [cgallemore](#), [csturtz](#), [d_searles](#), [jlorenzen](#), [panderson_007](#), [rriven](#)". To the right, it says "Donated by" followed by the "accenture" logo and the tagline "High performance. Delivered.". At the bottom, there is a section titled "XMPP Binding Component" with a description: "XMPP is a protocol for messaging and presence exchange that uses an XML format made popular by the Jabber community. The XMPP Binding Component (XMPPBC) allows services within a JBI container to utilize those capabilities and access entities on those networks." Below the description is a green bar labeled "Features".

Hello World

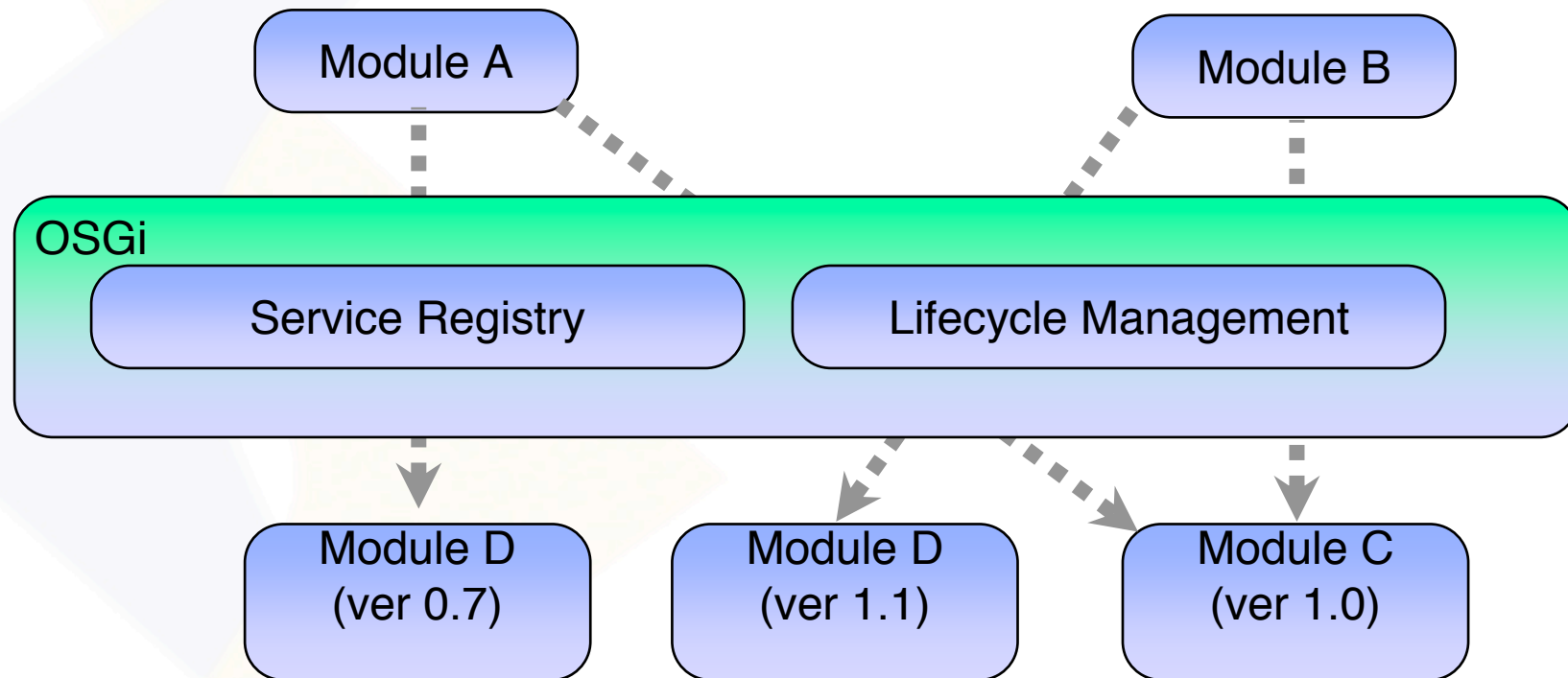
- Do ***NOT*** build a component.
- Well, I would show you some code, but there isn't any.
- Instead, just wire inputs and outputs of together via xml files.

➤ JBI Anecdotes

OSGi

- Made famous by Eclipse, it is a recent entrant to the enterprise-computing environment
- Turns J2EE Application Server attitudes on their head
- Seeks to provide lifecycle management, modularity and a service registry

The idea: Component Re-Use



The skinny

➤ Supporting it:

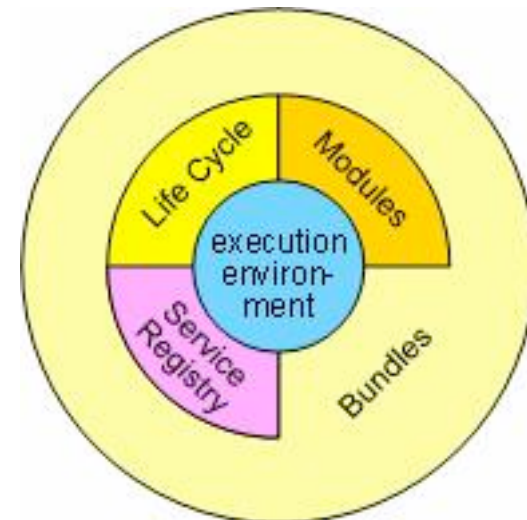
- Ericsson, IBM, Motorola, Nokia, NTT, Oracle, Red Hat, Samsung Electronics, Siemens, and Telefonica

➤ Who's got it:

- Eclipse Equinox
- Apache Felix
- Knoplerfish

The gritty

- Its all about the bundle.
- Bundles are just fancy jars with extra meta-information that specifies dependencies, and exposed interfaces.



<http://www.osgi.org/About/Technology>

The How-To: Step 1 of 3

➤ Implement BundleActivator

➤ // Register my service

➤ public void start(BundleContext context) throws Exception {

- MyService myService = new MyServiceImpl();
- context.registerService(MyService.class.getName(), myService, null);

➤ }

➤ // Get another service

➤ public ServiceReference[] getOtherServices(BundleContext context) {

- return context.getServiceReferences(OtherService.class.getName(), null);

➤ }

The How-To: Step 2 of 3

Create your bundle

- Manifest-Version: 1.0
- Export-Package:
`org.rvooz.salient.cpe,org.rvooz.salient.cpe.impl;uses:="org.rvooz.salient.interfaces,org.rvooz.salient.cpe,org.apache.log4j,org.rvooz.salient.matcher,javax.xml.bind",org.rvooz.salient.cpe.osgi;uses:="org.rvooz.salient.cpe,org.apache.log4j,org.rvooz.salient.matcher,org.rvooz.salient.cpe.impl,org.osgi.framework"`
- Tool: Bnd-0.0.160
- Bundle-Name: rVooz :: Salient :: Context Processor :: OSGi Bundle
- Created-By: 1.6.0_04-dp (Apple Inc.)
- Bundle-Vendor: rVooz.org
- Bundle-Version: 0.0.4
- Bnd-LastModified: 1203912467809
- Bundle-ManifestVersion: 2
- Bundle-Activator: `org.rvooz.salient.cpe.osgi.ContextProcessorActivator`
- Bundle-Description: Salient Server :: Context Processor :: OSGi Bundle
- Import-Package:
`javax.xml.bind,javax.xml.transform,javax.xml.transform.stream,org.apache.log4j,org.osgi.framework;version=1.3.0,org.rvooz.salient.cpe.impl,org.rvooz.salient.interfaces,org.rvooz.salient.matcher`
- Include-Resource: `src/main/resources`
- Bundle-SymbolicName: `org.rvooz.salient.cpe`

The How-To: Step 3 of 3

Install & Start

- `osgi> install file:///Users/bone/dev/rvooz.org/rvooz/archive/distros/equinox_salient/trunk/plugins/javax.servlet_2.4.0.v200706111738.jar`
 - Bundle id is 15
 - `osgi> 37336 [Framework Event Dispatcher] INFO [undefined] - BundleEvent INSTALLED`
 - `osgi> ss`
 - Framework is launched.
 - | id | State | Bundle |
|----|-----------|--|
| 0 | ACTIVE | org.eclipse.osgi_3.3.1.R33x_v20070828 |
| 7 | INSTALLED | org.eclipse.equinox.http.jetty_1.0.1.R33x_v20070816 |
| 10 | ACTIVE | org.springframework.osgi.log4j.osgi_1.2.13 |
| 11 | ACTIVE | org.apache.commons.logging_1.0.4.v200706111724 |
| 12 | ACTIVE | org.ops4j.pax.logging.pax-logging-api_0.9.7.SNAPSHOT |
| 13 | ACTIVE | org.ops4j.pax.logging.pax-logging-service_0.9.7.SNAPSHOT |
| 15 | INSTALLED | javax.servlet_2.4.0.v200706111738 |
 - `osgi> start 15`
 - `osgi> 42297 [Framework Event Dispatcher] INFO [undefined] - BundleEvent RESOLVED`
 - `42297 [Framework Event Dispatcher] INFO [undefined] - BundleEvent STARTED`
-
- `osgi>`

OSGi Available Services

- Log, Configuration, Preferences, Security, Device Management, UPnP, and....

- HTTP Service



The screenshot shows the rVooz website with a blue header containing the logo and the tagline "...making contextions". A navigation bar includes links for Home, Mood Demo, Geo Demo (soon), Development, Wiki, and Blogs. A central yellow box announces the presentation at eclipseCON™ 2008, March 17th - 20th in Santa Clara, California, and also mentions the Emerging Technologies for the Enterprise Conference on March 26-27th, 2008. Below this, an "Overview" section describes rVooz as an open project for dynamic connections between people with shared context. To the right of the text is an illustration of three stick figures with speech bubbles and a central cloud-like shape. At the bottom, there are links for Get Started, Get Involved, Latest Activity, and License.

rVooz ...making contextions

[Home](#) [Mood Demo](#) [Geo Demo \(soon\)](#) [Development](#) [Wiki](#) [Blogs](#)

We're presenting here

eclipseCON™ 2008 March 17th - 20th
Santa Clara, California

And at the [Emerging Technologies for the Enterprise](#) Conference, March 26-27th, 2008.

Overview

rVooz is an open and collaborative project designed to make dynamic connections between people who have a shared context. rVooz is a software suite being built to make contextual connections, or "contextions," between people who may or may not have prior knowledge of each other. The idea is to bring people together even if they don't have each other in their buddy lists or know each other's phone numbers.

[Get Started](#) | [Get Involved](#) | [Latest Activity](#) | [License](#)

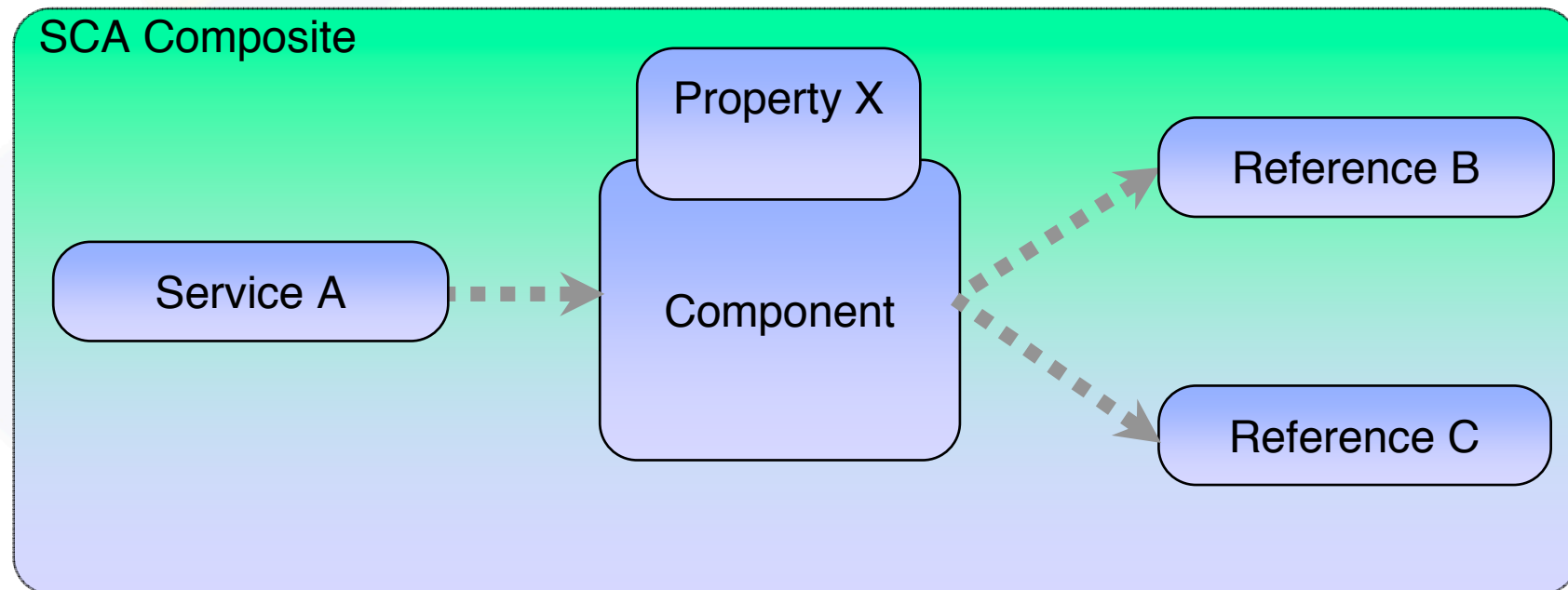


➤ OSGi Anecdotes

Service Component Architecture

- “Service Component Architecture (SCA) is a set of specifications which describe a **model** for building applications and systems using a Service-Oriented Architecture. SCA extends and complements prior approaches to implementing services, and SCA builds on open standards such as Web services.”

The idea: High-Level Model-Driven SOA



The skinny

- Produced by the Open Service-Oriented Architecture (OSOA)
 - BEA, IBM, Oracle, SAP, IONA, Siebel and Sybase
 - "A Deployment Descriptor On Steroids"
 - › <http://webservices.sys-con.com/read/158318.htm>
- Who's got it?
 - Apache Tuscany
 - BEA?
 - IBM?

The gritty

- Composite Applications are defined in XML files.
- Annotations are used in the code to connect the meta-data files to reality
- “Language independence”
 - Too Meta-Meta?

The How-To: Step 1 of 3

Create your composite app:

```
><composite xmlns="http://www.ooa.org/xmlns/sca/1.0"
>   targetNamespace="http://sample"
>   xmlns:sample="http://sample"
>   name="Calculator">
>     <component name="CalculatorServiceComponent">
>       <implementation.java class="calculator.CalculatorServiceImpl"/>
>       <reference name="addService" target="AddServiceComponent" />
>     </component>
>
>     <component name="AddServiceComponent">
>       <implementation.java class="calculator.AddServiceImpl"/>
>     </component>
> </composite>
```

The How-To: Step 2 of 3

Annotate your Java:

```
➤ package calculator;
➤ import org.osoa.sca.annotations.Reference;
➤ public class CalculatorServiceImpl implements CalculatorService {
➤     private AddService addService;

➤     @Reference
➤     public void setAddService(AddService addService) {
➤         this.addService = addService;
➤     }
➤     public double add(double n1, double n2) {
➤         return addService.add(n1, n2);
➤     }
➤ }
```

The How-To: Step 3 of 3

Call your Service:

```
➤ import org.apache.tuscany.sca.host.embedded.SCADomain;

➤
➤ public class CalculatorClient {
➤
➤     public static void main(String[] args) throws Exception {
➤         SCADomain scaDomain = SCADomain.newInstance("Calculator.composite");
➤
➤         CalculatorService calculatorService =
➤             scaDomain.getService(CalculatorService.class, "CalculatorServiceComponent");
➤         System.out.println("3 + 2=" + calculatorService.add(3, 2));
➤         scaDomain.close();
➤
➤     }}
```

The Table of Truth (Sort-Of)

	Mediator	Dependency References	Service Registry	Loose Coupling
JB1	Yes (NMR)	No	Yes	?
OSGi	No	Yes (Java)	Yes (Dynamic)	?
SCA	No	Yes (Independent Model)	Not Really	?

Swordfish to the rescue...

➤ <http://www.eclipse.org/swordfish/>



➤ “In Deutscher's view of Swordfish, SCA provides a common programming model and a common description format. JBI provides a common messaging model. OSGi provides a common deployment model and a common runtime component model.”

Completing the circle...

<imho>

➤ YES

</imho>

Thanks...



I'm Speaking @

Emerging Technologies
for the Enterprise
Conference



March 26 -27, 2008

BACKUP

The dissenters

➤ BEA:

- BEA believes that the JBI specification is an incomplete attempt to standardize the interfaces between multi-vendor infrastructure and contributes little to the usefulness of the Java platform for business application integration, one of the real pain point for our customers. It's unfortunate that it's name alone will result in significant confusion in the marketplace.

➤ IBM:

- IBM abstains because the JBI specification doesn't represent a sufficient step forward in terms of what we believe our customers need, and above what they can already do. Many technologies and open specifications are available to the Java programmer today with more compelling interoperability and better mechanisms for component composition. IBM's priority is to enable integration with the broadest range of platforms, applications, and existing business assets. This demands a language-neutral approach using today's Web Services standards, and simpler programming and application models.

What others say... 1 of 2

- n “Since JBI is based around a NormalizedMessage and talks about WSDLs being used for each endpoint, you might think that a JBI container cannot be used to wire together things using POJOs. Well, bending the JBI specification some, but keeping to the APIs - you can pass around any named POJO in a NormalizedMessage and completely ignore the XML payload if you wish. So within the JBI bus you can work purely with POJOs and ignore the XML, which tends to only be used (if at all) to marshal things onto a wire (which is when you want to use XML anyways).”
- n <http://incubator.apache.org/servicemix/does-jbi-depend-on-xml-and-wsdl.html>

What others say... 2 of 2

- n “JBI uses the WSDL 2.0 services model extensively. This gives it very good alignment with WSDL described services, including WSDL 1.1. (JBI itself does not provide APIs for manipulating WSDL documents; that is for the next version of JSR 110 to provide).

Components that are service providers **MUST** return appropriate WSDL descriptions of those services when they are activated. If a component returns null from its `getServiceDescription()` method implementation, it means that the given endpoint is not active, or that the component is behaving incorrectly.”

- n <http://forum.java.sun.com/thread.jspa?threadID=700153&tstart=210>