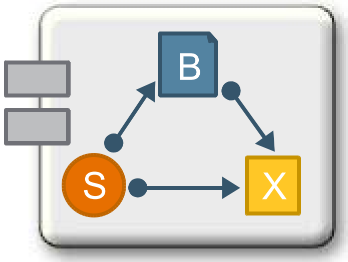


JavaOne

Introduction to CASA: An Open Source Composite Application Editor

Tientien Li, Ph.D. and Jun Qian

Sun Microsystems, Inc.

TS-8683

Introduction to CASA

An Open Source Composite Application Editor

We will discuss...

REST ~~WWW~~	SOA
Hypermedia	Composite Apps
Linked Resources	Connected Services
URL/HTML	WSDL/XML
Data Representation	Business Logic

(... discussed elsewhere)

and demonstrate CASA...

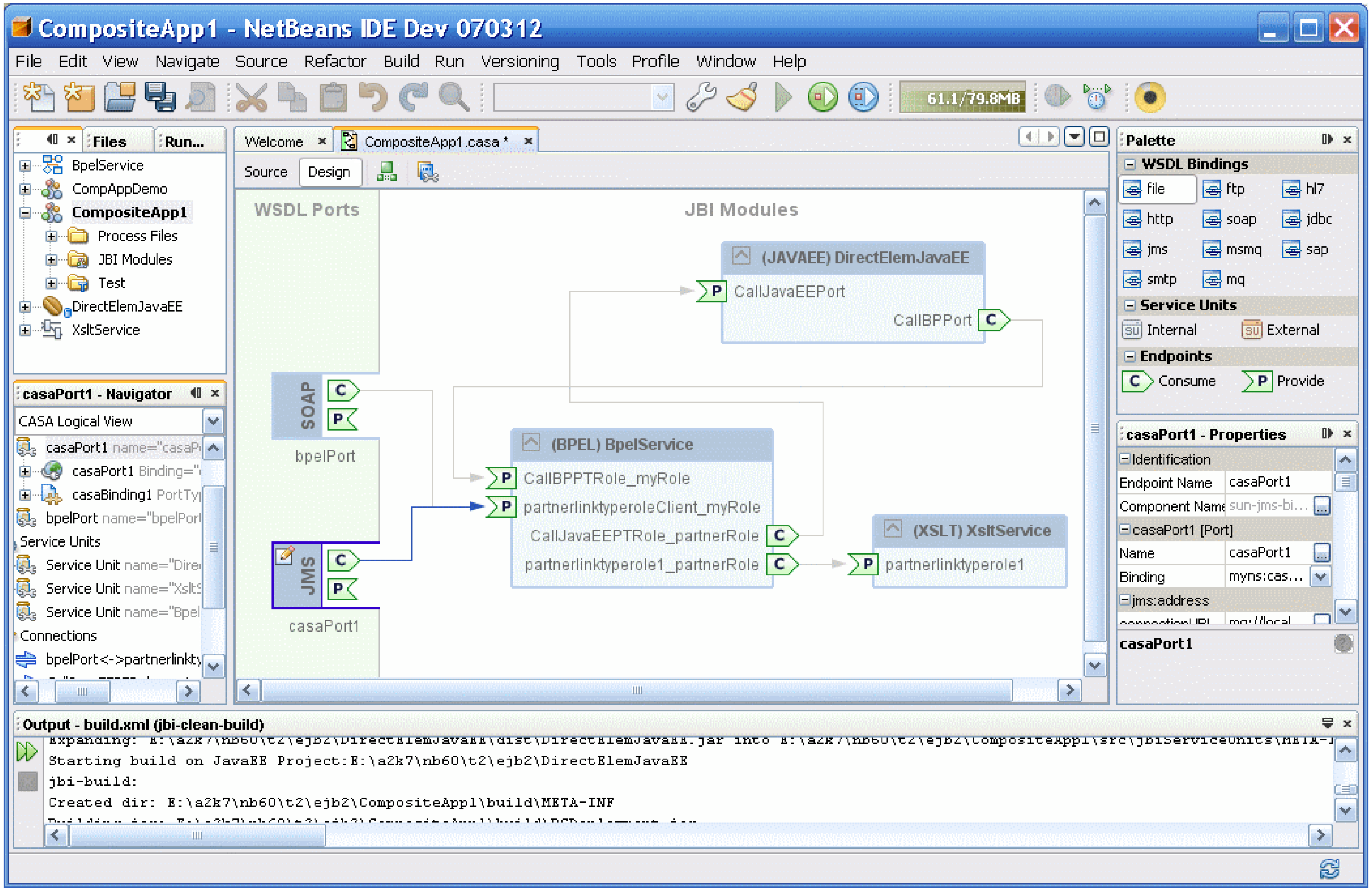
Session Agenda

- Introduction (15 minutes)
  - CASA, SOA, Composite App, Java™ Business Integration (JBI), NetBeans™ Software Enterprise Pack
- Composite Application Concepts (20 minutes)
  - AppModel, Dev Environment, Architecture, Interface, Component Model, AutoGen, Workflow
- CASA Demos (20 minutes)
  - Edit external and internal connectivity
- Summary and Q&A (5 minutes)

Session Agenda

- **Introduction (15 minutes)**
  - **CASA, SOA, Composite App, Java™ Business Integration (JBI), NetBeans™ Software Enterprise Pack**
- Composite Application Concepts (20 minutes)
  - AppModel, Dev Environment, Architecture, Interface, Component Model, AutoGen, Workflow
- CASA Demos (20 minutes)
  - Edit external and internal connectivity
- Summary and Q&A (5 minutes)

1.intro.CASA



Endpoint Name	casaPort1
Component Name	sun-jms-bi...
Name	casaPort1
Binding	myns:cas...
connectionURL	mq://local

```
Expanding: E:\a2k7\nb60\t2\ejb2\DirectElemJavaEE\dist\DirectElemJavaEE.jar into E:\a2k7\nb60\t2\ejb2\CompositeApp1\src\jbiServiceUnits\META-I
Starting build on JavaEE Project:E:\a2k7\nb60\t2\ejb2\DirectElemJavaEE
jbi-build:
Created dir: E:\a2k7\nb60\t2\ejb2\CompositeApp1\build\META-INF
```

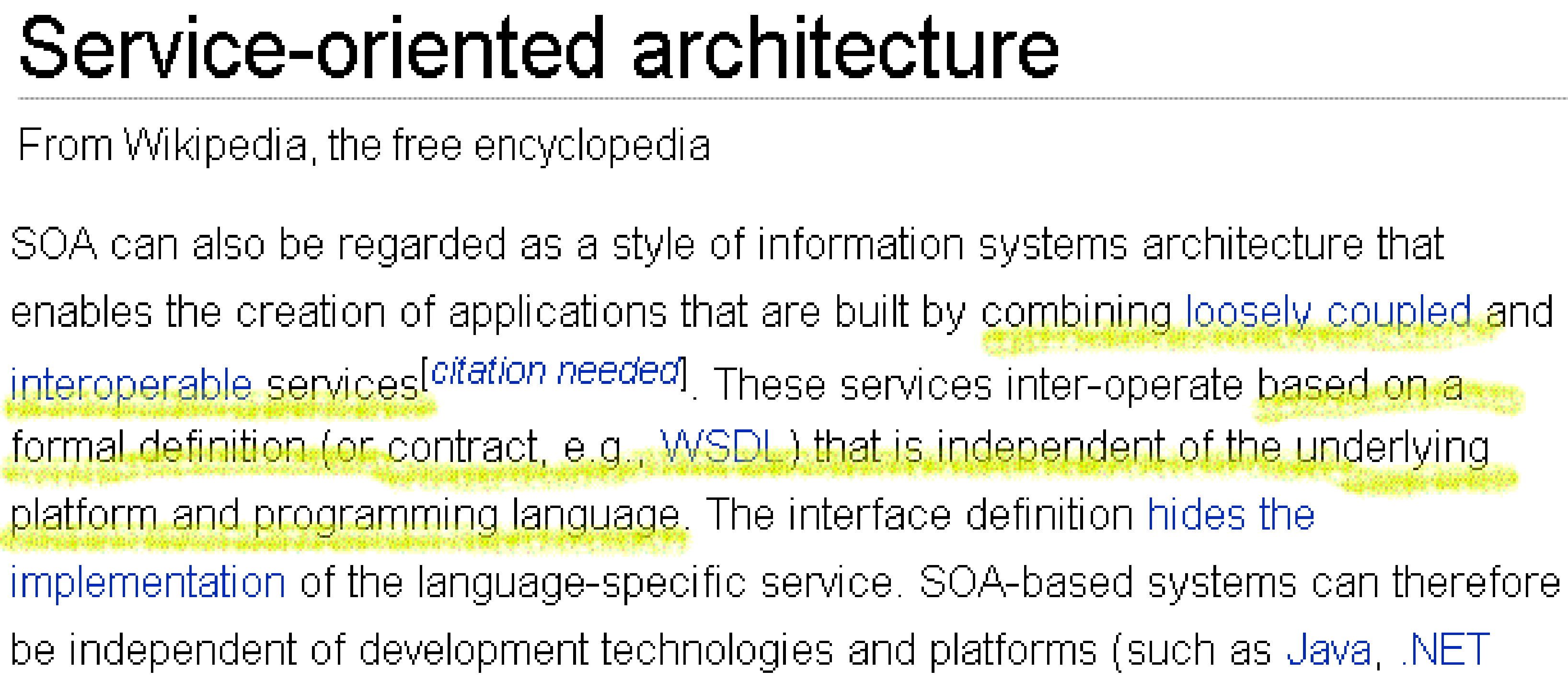
JBI/CompApp… …NetBeans Software EntPack CASA

Fall 2004 06 2005 Fall 2005 9/10 2006 mid 2007

1.intro.CASA

- Composite Application Service Assembly Editor
  - A new editor in NetBeans release 6.0 (Open Source)
  - For editing Composite Application project configurations
  - Targeting the JBI platform, Project Open Enterprise Service Bus (Open ESB)
- Development History/Status
  - JBI/Composite Application (see beyond)—late 2004
  - Merged with Sun—June 2005
  - NetBeans software Enterprise Pack—most of 2006
  - CASA editor—late 2006
- CASA development team
  - Tientien Li, Jun Qian, Josh Sandusky, Ramesh Dara

1.intro.SOA



Service-oriented architecture

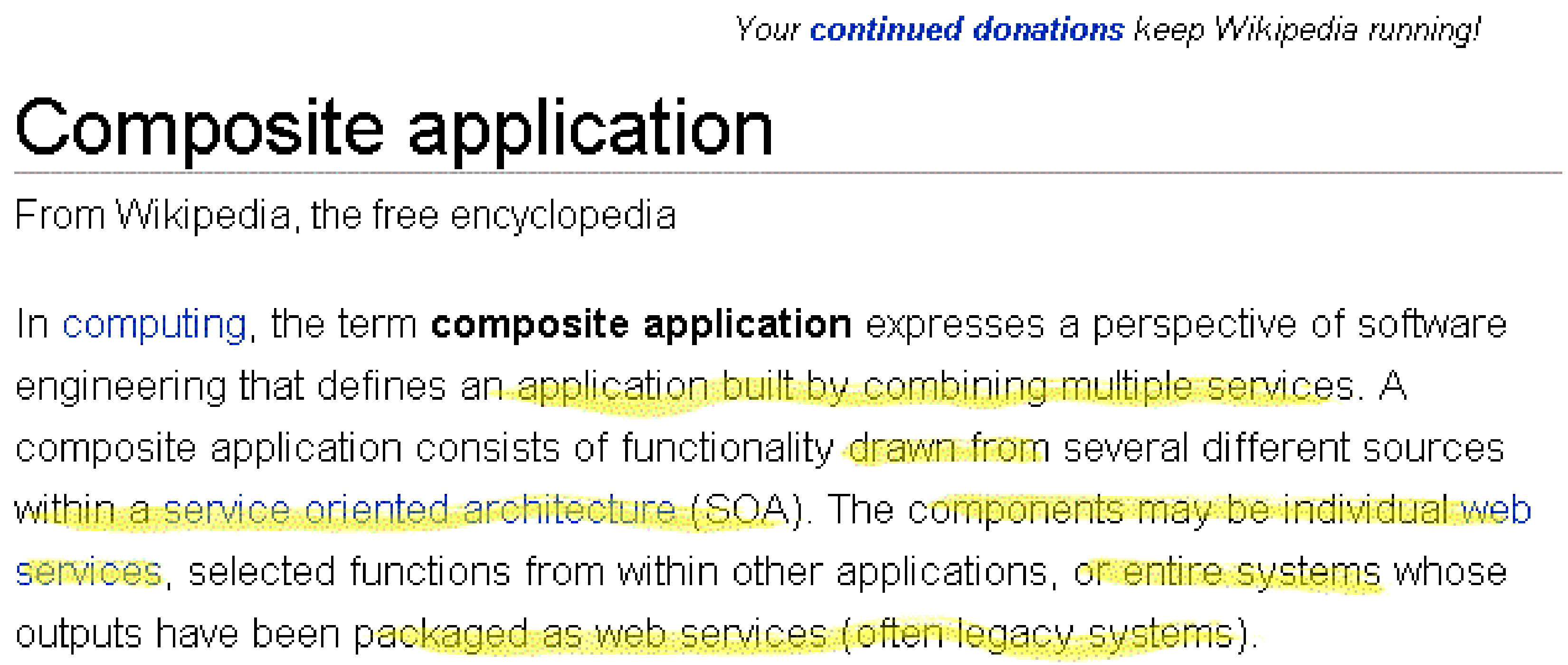
From Wikipedia, the free encyclopedia

SOA can also be regarded as a style of information systems architecture that enables the creation of applications that are built by combining loosely coupled and interoperable services[citation needed]. These services inter-operate based on a formal definition (or contract, e.g., WSDL) that is independent of the underlying platform and programming language. The interface definition hides the implementation of the language-specific service. SOA-based systems can therefore be independent of development technologies and platforms (such as Java, .NET

1.intro.SOA

- SOA enables applications be built by combining:
  - Loosely coupled and interoperable services
- Services interoperate based on:
  - A formal definition (or contract, e.g., WSDL)
- WSDL is independent of the underlying:
  - Implementation platform and programming language
- SOA-based systems can therefore be:
  - Independent of development technologies

1.intro.CompApp



*Your **continued donations** keep Wikipedia running!*

Composite application

From Wikipedia, the free encyclopedia

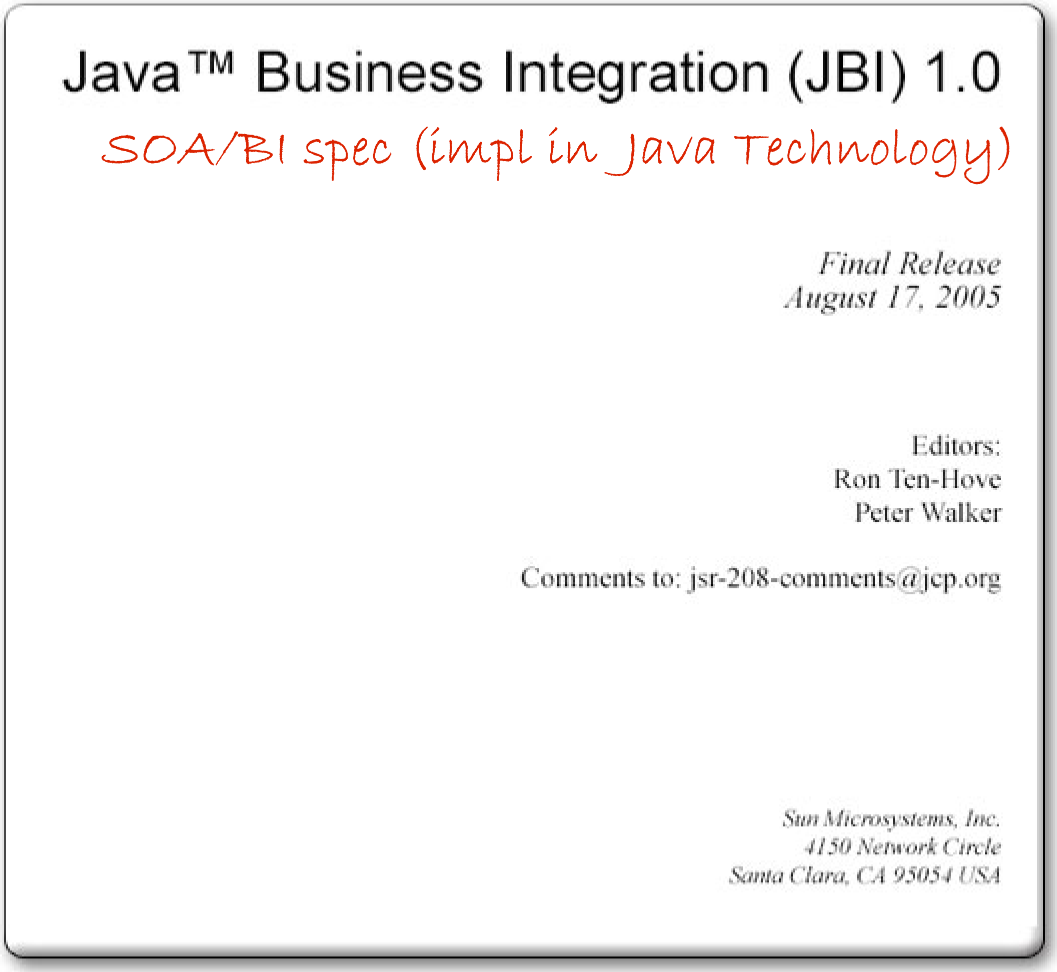
In computing, the term **composite application** expresses a perspective of software engineering that defines an application built by combining multiple services. A composite application consists of functionality drawn from several different sources within a service oriented architecture (SOA). The components may be individual web services, selected functions from within other applications, or entire systems whose outputs have been packaged as web services (often legacy systems).

1.intro.CompApp

- “Composite Application” defined as:
  - An application built by combing multiple services
  - From sources within a Service-Oriented Architecture (SOA)
- The components may be:
  - Individual web services (defined by WSDL/XML)
  - Functions from other applications
  - Legacy systems wrapped as web services

1.intro.jBI

- A standards-based (XML/WSDL) service-oriented architecture (SOA) for integration solutions
- Allow third-party components to be "plugged in" and interoperate in a predictable, reliable fashion
  - Component container spec
  - Message exchange spec
  - Deployment and management spec



Java™ Business Integration (JBI) 1.0

SOA/BI spec (impl in Java Technology)

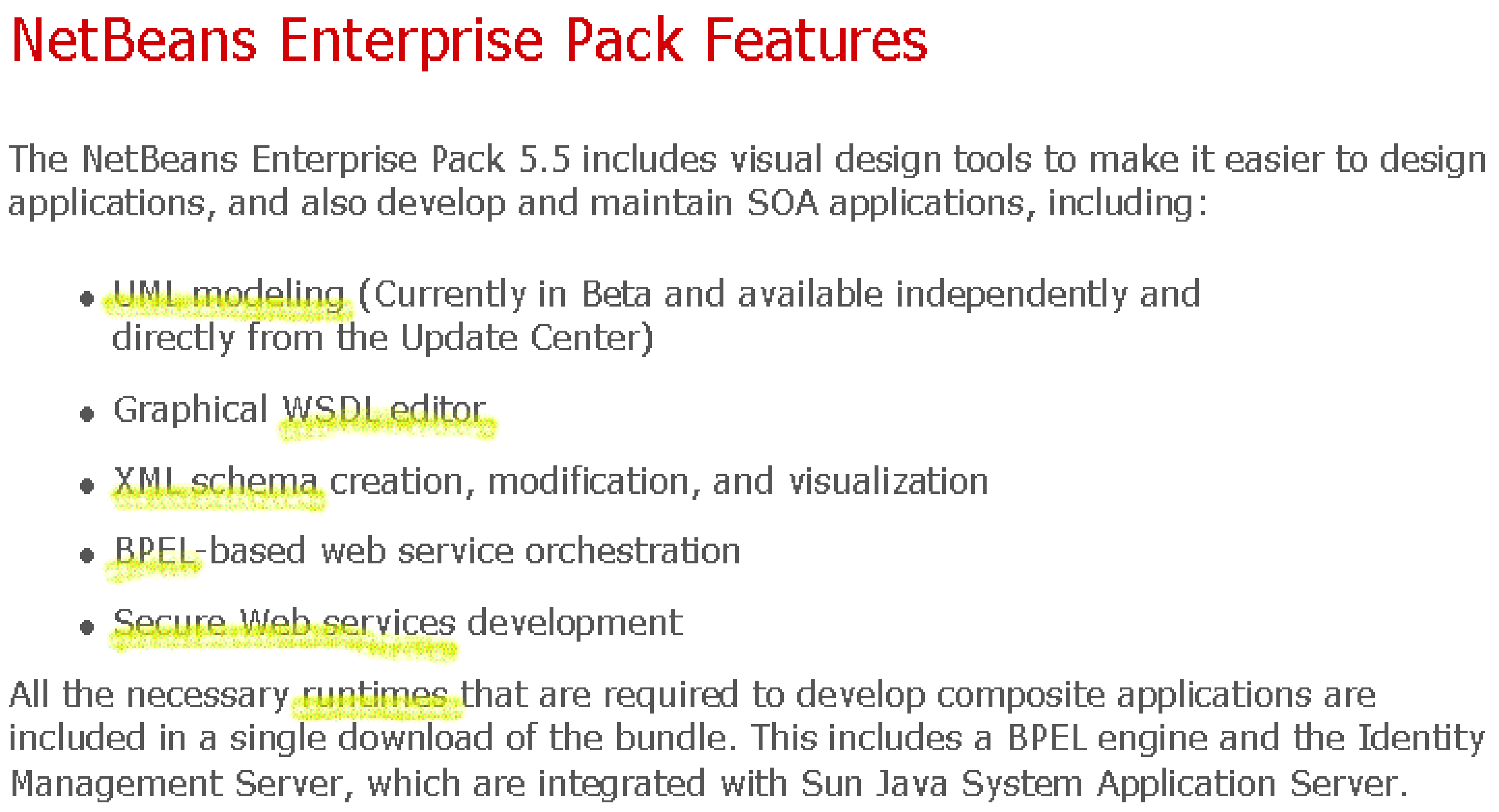
Final Release
August 17, 2005

Editors:
Ron Ten-Hove
Peter Walker

Comments to: jsr-208-comments@jcp.org

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054 USA

1.intro.NbEntPack



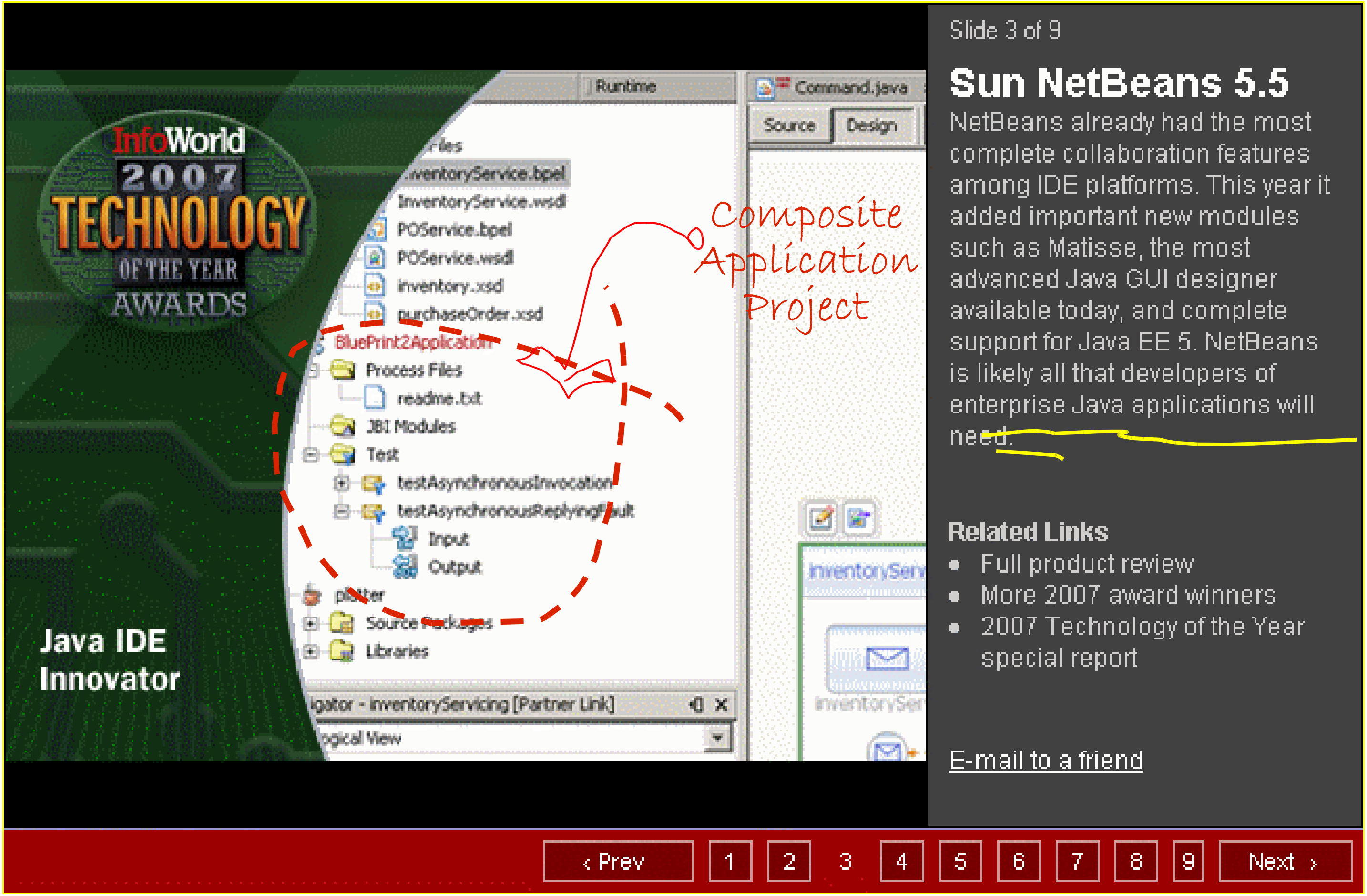
NetBeans Enterprise Pack Features

The NetBeans Enterprise Pack 5.5 includes visual design tools to make it easier to design applications, and also develop and maintain SOA applications, including:

- UML modeling (Currently in Beta and available independently and directly from the Update Center)
- Graphical WSDL editor
- XML schema creation, modification, and visualization
- BPEL-based web service orchestration
- Secure Web services development

All the necessary runtimes that are required to develop composite applications are included in a single download of the bundle. This includes a BPEL engine and the Identity Management Server, which are integrated with Sun Java System Application Server.

1.intro.NbEntPack



Slide 3 of 9

Sun NetBeans 5.5

NetBeans already had the most complete collaboration features among IDE platforms. This year it added important new modules such as Matisse, the most advanced Java GUI designer available today, and complete support for Java EE 5. NetBeans is likely all that developers of enterprise Java applications will need.

Related Links

- Full product review
- More 2007 award winners
- 2007 Technology of the Year special report

E-mail to a friend

Source: http://www.infoworld.com/slideshow/2007/01/25-2007_technology-3.html

Session Agenda

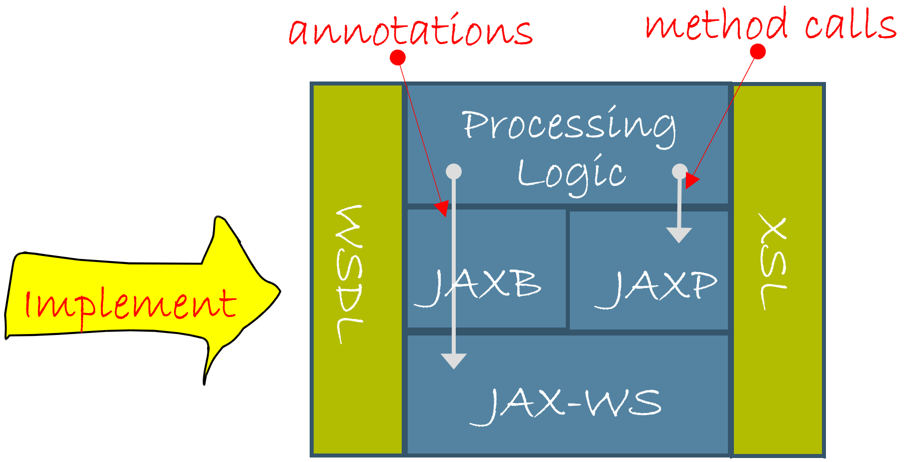
- Introduction (15 minutes)
  - CASA, SOA, Composite App, Java Business Integration (JBI), NetBeans Software Enterprise Pack
- **Composite Application Concepts (20 minutes)**
  - **AppModel, Dev Environment, Architecture, Interface, Component Model, AutoGen, Workflow**
- CASA Demos (20 minutes)
  - Edit external and internal connectivity
- Summary and Q&A (5 minutes)

2.compApp.appModel

- Traditional enterprise service application

an Enterprise Application Spec

- Receive XML input from either:
  - http/soap
  - batch files
- Process input by applying rules
  - Some rules impl'ed in XSL
- Return results

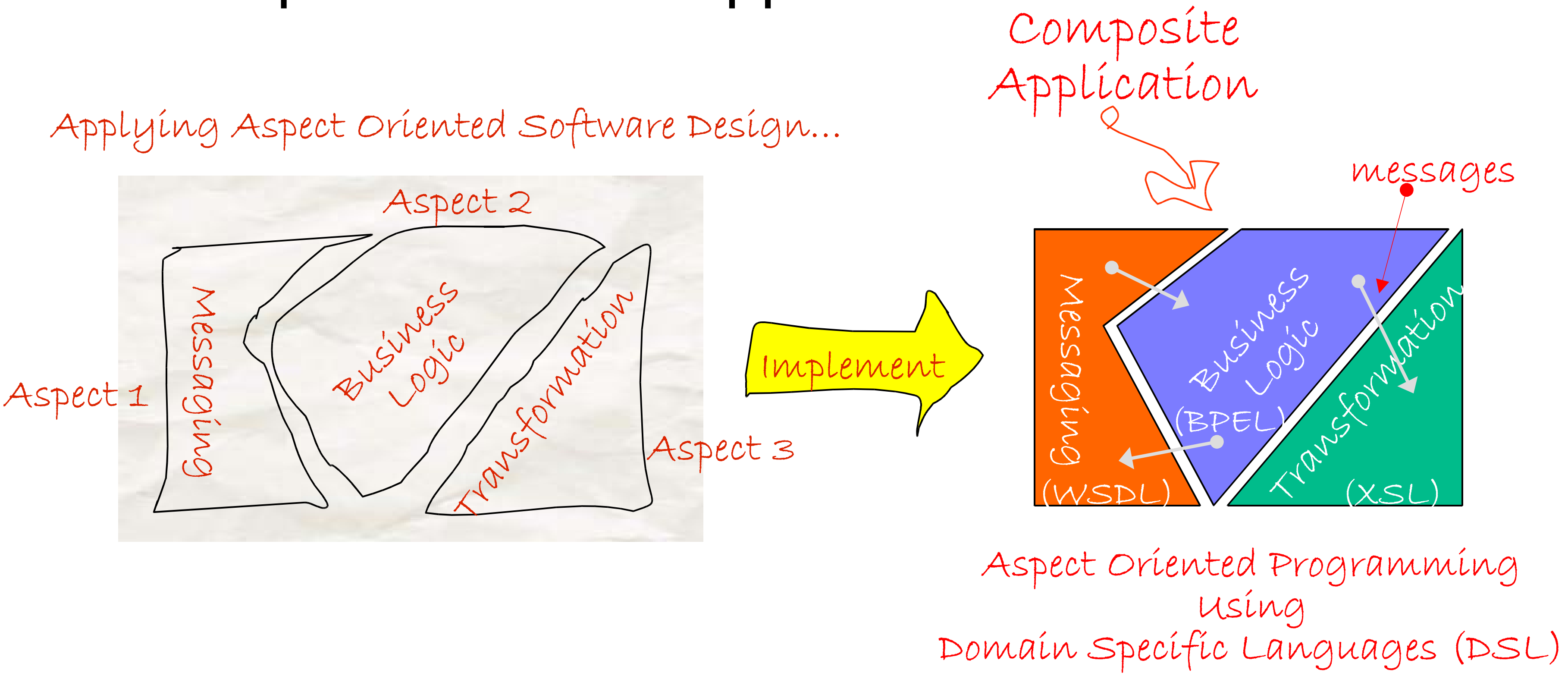


A Java Service Application
for
An Enterprise App Platform

JAXB = Java API for XML Binding | Java API for XML Processing | Java API for XML Web Services Addressing

2.compApp.appModel

- Composite service application



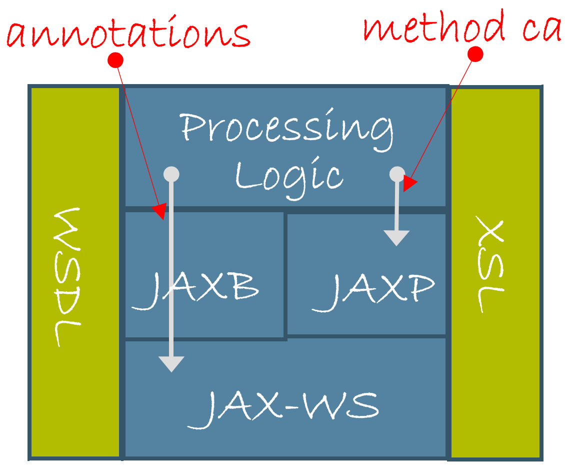
Aspect Oriented Programming
Using
Domain Specific Languages (DSL)

2.compApp.appModel

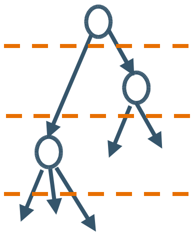
- Comparison of service application models

Traditional application model using annotations and domain specific libraries, e.g., Java Platform, Enterprise Edition (Java EE platform) and .NET web services

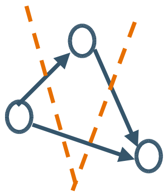
Composite application model using messages and domain specific languages, e.g., NB CompApps on Open ESB



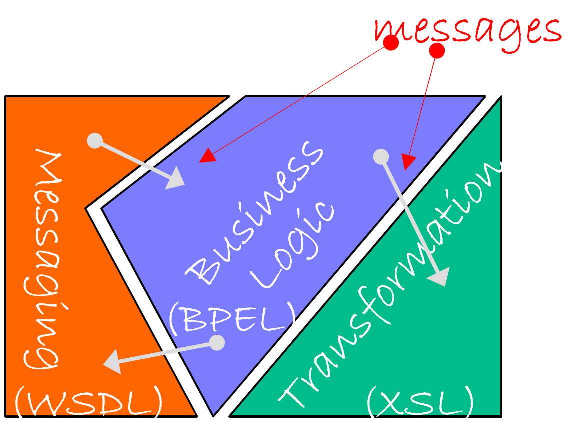
A Java Service Application
for
An Enterprise App Platform



Hierarchical, Layered, Stacked App Model



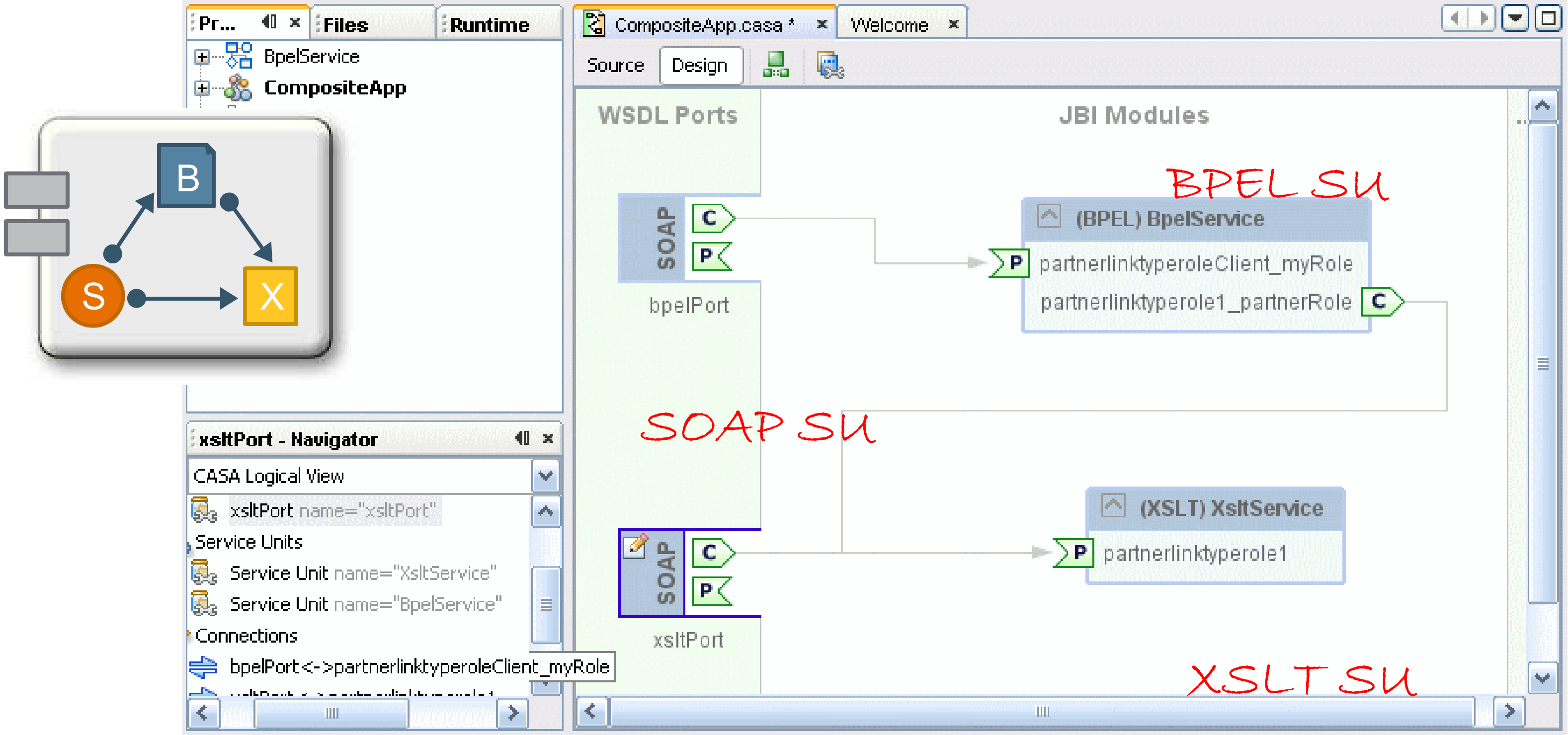
Network Peer-to-peer, Compositional App Model



Aspect Oriented Programming
using
Domain Specific Languages (DSL)

2.compApp.devEnv

- Composite application design-time (NetBeans software)



xsltPort name="xsltPort"
Service Units
Service Unit name="XsltService"
Service Unit name="BpelService"
Connections
bpelPort<->partnerlinktyperoleClient_myRole

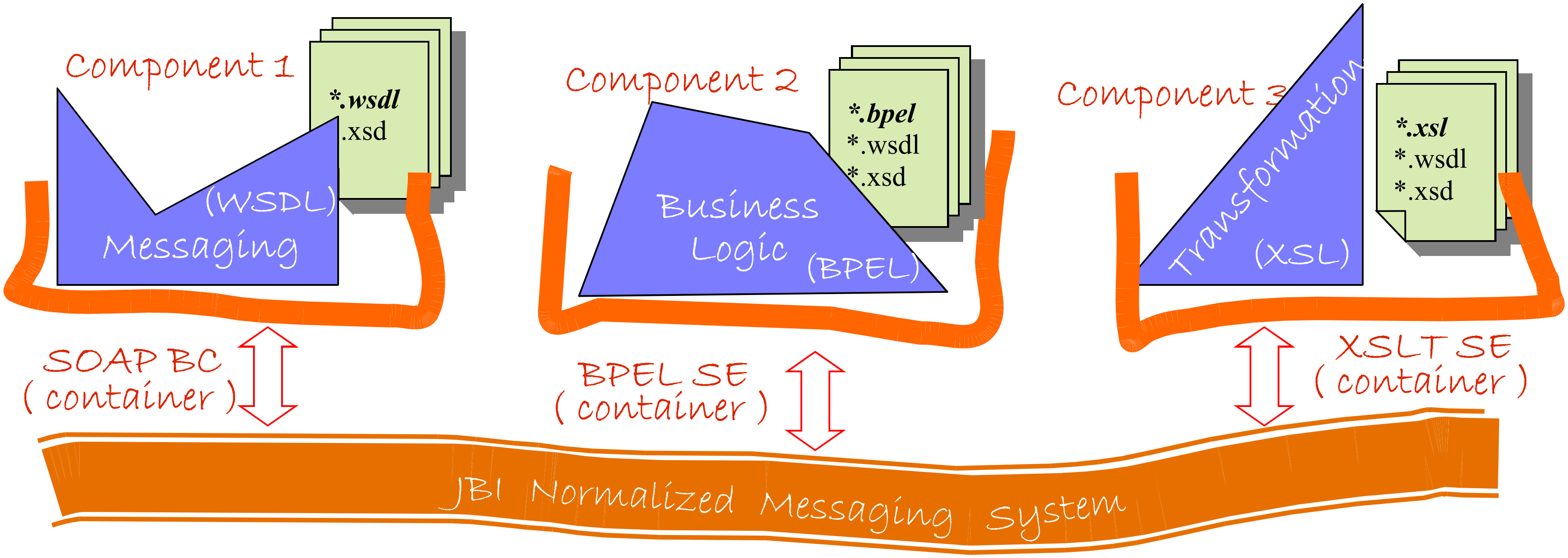
BPEL SU

SOAP SU

XSLT SU

2.compApp.devEnv

- Composite application run-time (Open ESB)



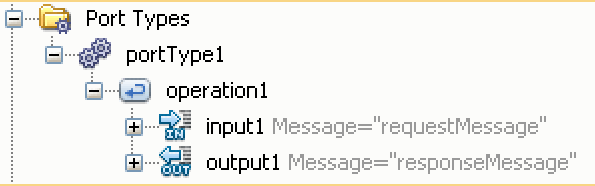
2.compApp.Interface

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions ...
```

Internal Interface

```
  <message name="requestMessage">
    <part name="inputType" type="xsd:string"/>
  </message>
  <message name="responseMessage">
    <part name="resultType" type="xsd:string"/>
  </message>
  <portType name="portType1">
    <operation name="operation1">
      <input name="input1" message="tns:requestMessage"/>
      <output name="output1" message="tns:responseMessage"/>
    </operation>
  </portType>
```

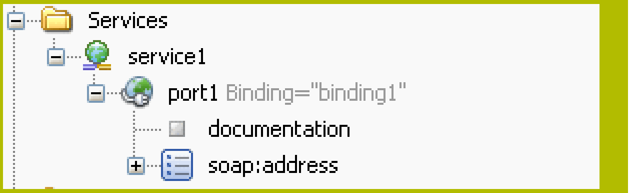
Abstract WSDL Elements



```
Port Types
  portType1
    operation1
      input1 Message="requestMessage"
      output1 Message="responseMessage"
```

External Interface

```
  <binding name="binding1" type="tns:portType1">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
    <operation name="operation1">
    <input name="input1"><soap:body use="literal"/></input>
     <output name="output1"><soap:body use="literal"/></output>
    </operation>
  </binding>
  <service name="service1">
    <port name="port1" binding="tns:binding1">
      <documentation/>
      <soap:address location="http://localhost:18181/SynchronousSample"/>
    </port>
  </service>
```



```
Services
  service1
    port1 Binding="binding1"
      documentation
      soap:address
```

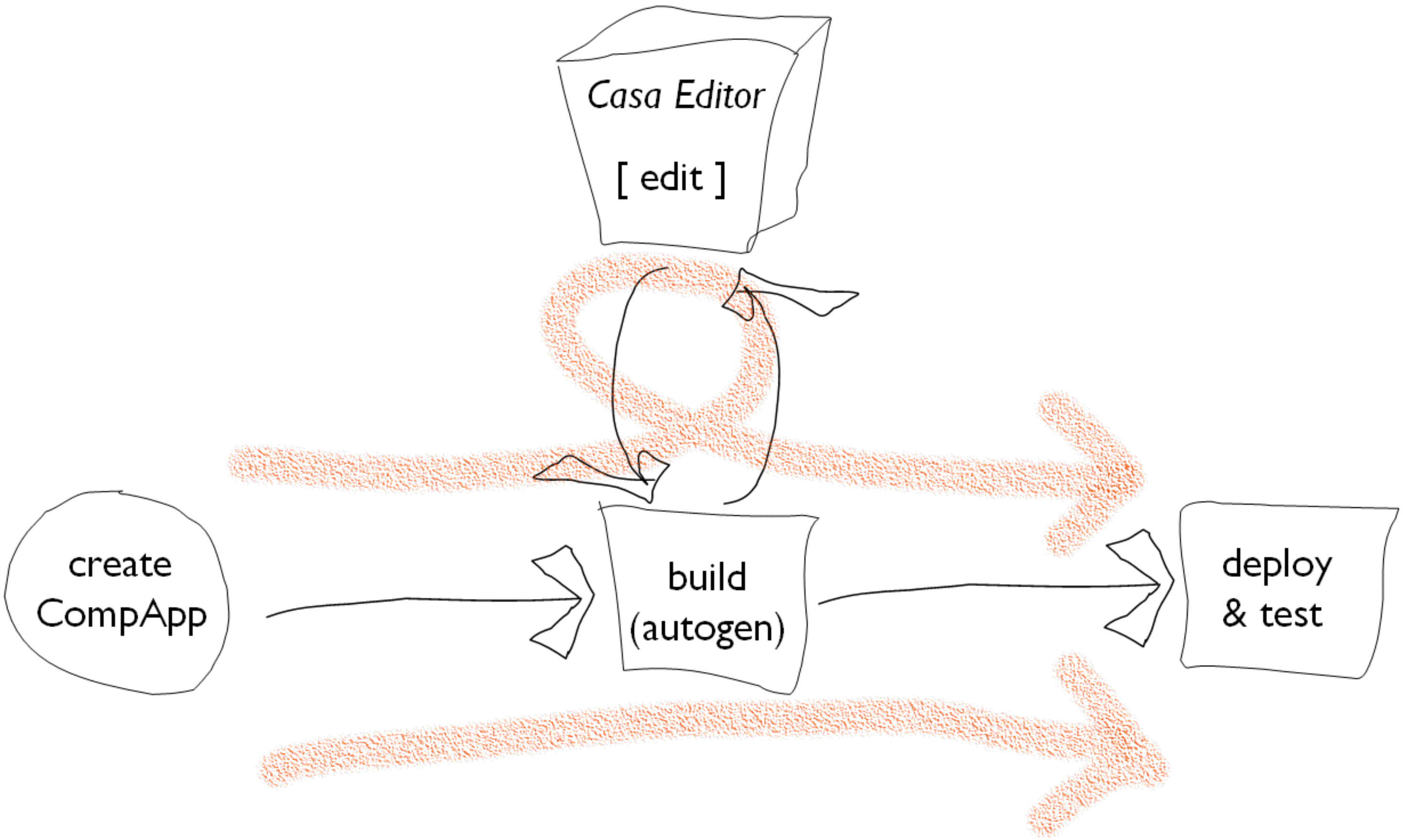
Concrete WSDL Elements

```
  <plink:partnerLinkType name="partnerlinktype1">
    <plink:role name="partnerlinktyperole1" portType="tns:portType1"/>
  </plink:partnerLinkType>
```

BPEL Extension Elements

```
</definitions>
```

2.compApp.Workflow



Session Agenda

- Introduction (15 minutes)
  - CASA, SOA, Composite App, Java Business Integration (JBI), NetBeans Software Enterprise Pack
- Composite Application Concepts (20 minutes)
  - AppModel, Dev Environment, Architecture, Interface, Component Model, AutoGen, Workflow
- **CASA Demos (20 minutes)**
  - **Edit external and internal connectivity**
- Summary and Q&A (5 minutes)

DEMO 1

Edit External Connectivity—To Create Virtual Services

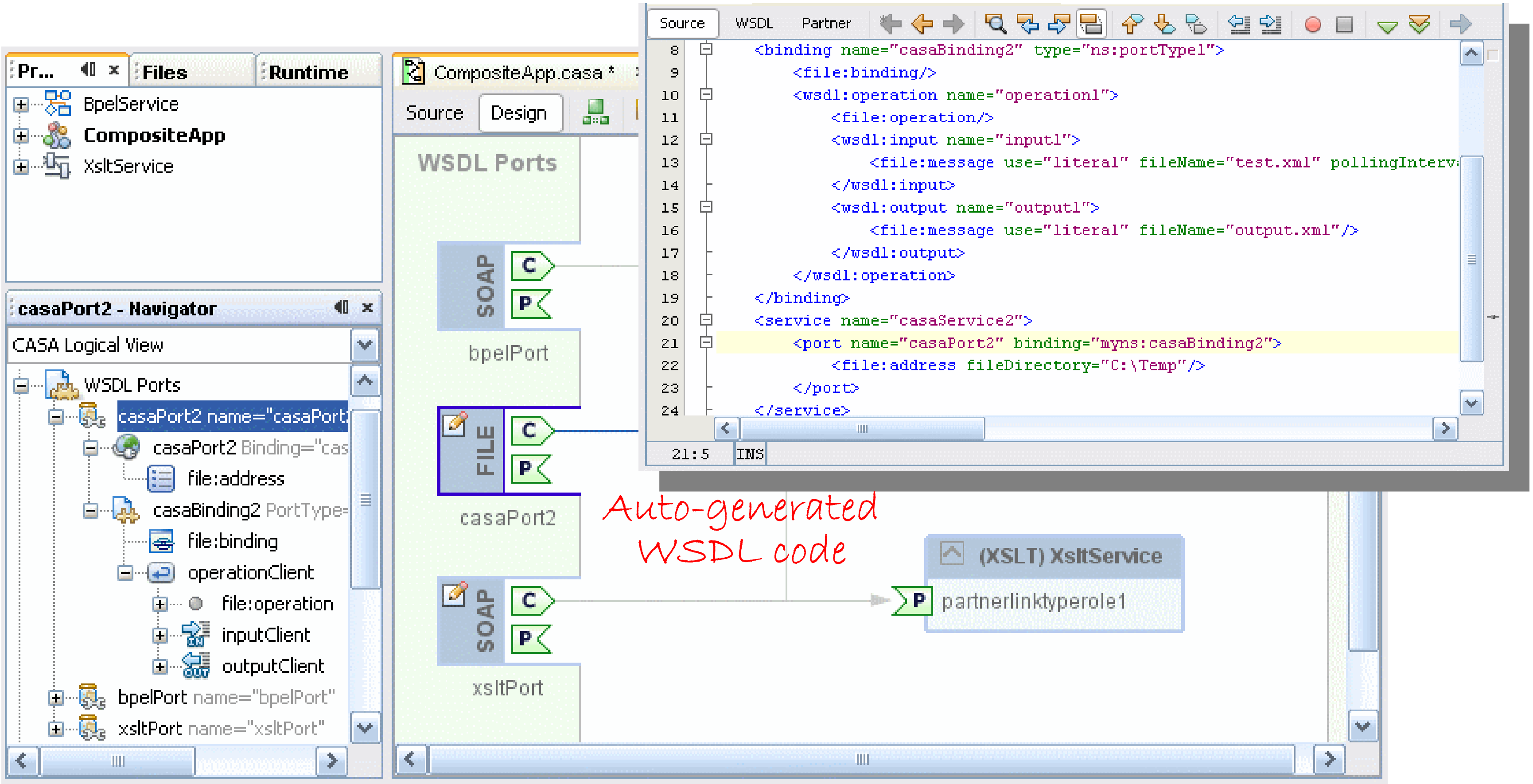
3.demo1.description

- Add a new file binding port and a connection to the sample composite application using CASA; this is to show how to edit the external connectivity to create a new virtual service
- Steps
  - Create a composite app project and a bpel project
  - Build, deploy, and test the composite app project
  - Open the composite app project in CASA
  - Add a file binding port
  - Create a new connection
  - Edit port attributes
  - Save the composite application project
  - Deploy and test the project

DEMO 1

3.demo1.casaScreen

- Create an external connection



```
<binding name="casaBinding2" type="ns:portType1">
    <file:binding/>
    <wsdl:operation name="operation1">
        <file:operation/>
        <wsdl:input name="input1">
            <file:message use="literal" fileName="test.xml" pollingInterv
        </wsdl:input>
        <wsdl:output name="output1">
            <file:message use="literal" fileName="output.xml"/>
        </wsdl:output>
    </wsdl:operation>
</binding>
<service name="casaService2">
    <port name="casaPort2" binding="myns:casaBinding2">
        <file:address fileDirectory="C:\Temp"/>
    </port>
</service>
```

Auto-generated WSDL code

DEMO 2

Edit Inter-SA Connectivity—
To Use Third-Party Services

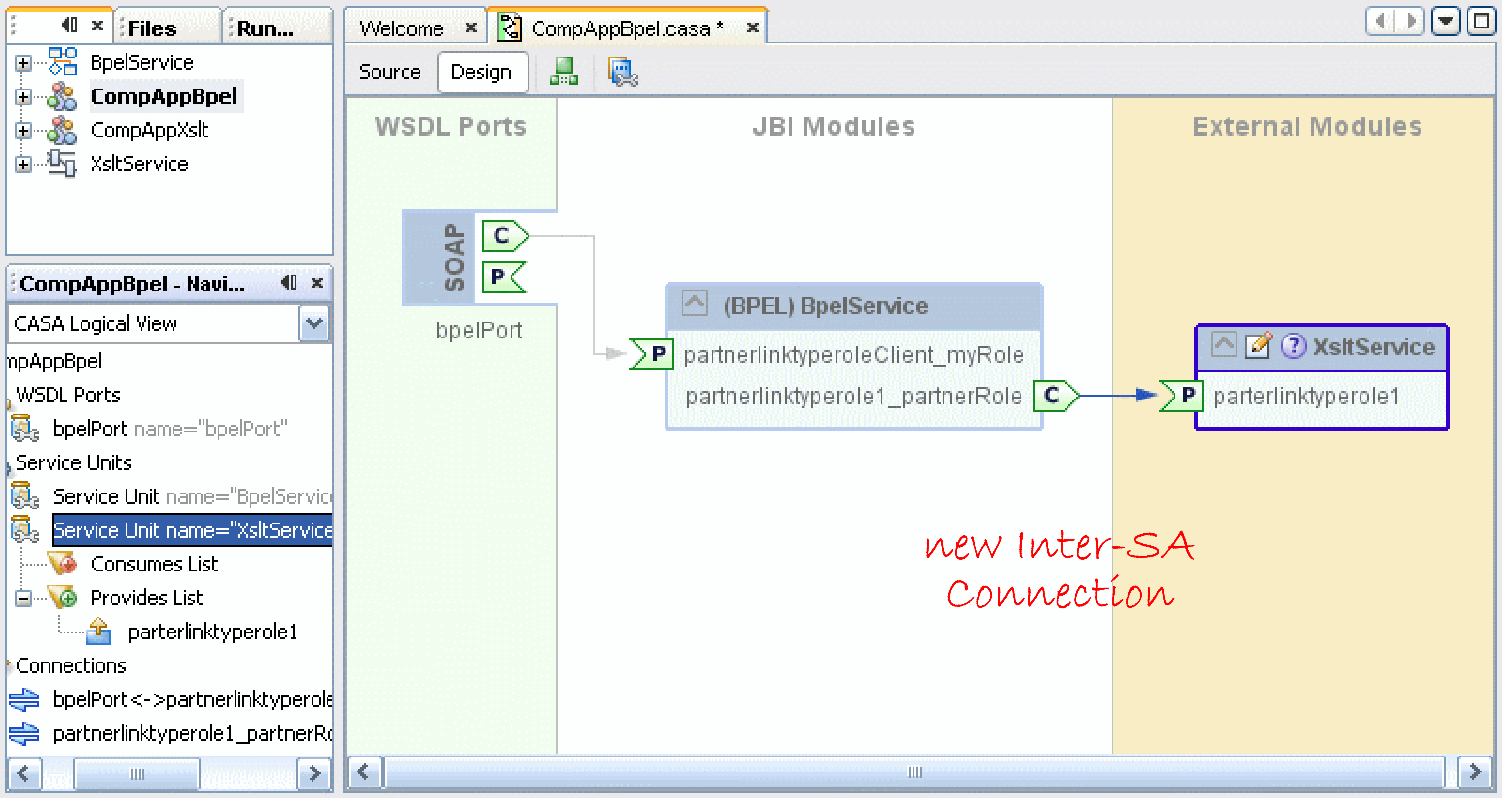
3.demo2.description

- Assuming a third-party composite application, CompAppXslt, already deployed to the run-time, we need to make a connection to an endpoint of a service unit within CompAppXslt from our application
- Steps
  - Open our composite application in CASA
  - Add an external service unit
  - Add an provide endpoint to the external service unit
  - Create a new connection
  - Edit endpoint attributes
  - Save the composite application project
  - Deploy and test the project

DEMO 2

3.demo2.casaScreen

- Create an inter-SA connection



BpelService
CompAppBpel
CompAppXslt
XsltService

mpAppBpel
WSDL Ports
bpelPort name="bpelPort"
Service Units
Service Unit name="BpelServic
Service Unit name="XsltService
Consumes List
Provides List
parterlinktyperole1
Connections
bpelPort<->partnerlinktyperole
partnerlinktyperole1_partnerR

new Inter-SA Connection

3.demo2.connectionComparison

- Comparison of service connections

	Connection Type		
	Inter-SU	Inter-SA	External
Connection Endpoints	Services of a Service Assembly	Services of Different Service Assemblies	Services of Different Service Platforms
Service Platform	Same JBI Platform	Same JBI Platform	Maybe Different
Target Service	Early Binding (@design-time)	Late Binding (@run-time)	Late Binding (@run-time)
Message Format	JBI Normalized XML Message	JBI Normalized XML Message	WSDL Specified Message Format

Session Agenda

- Introduction (15 minutes)
    - CASA, SOA, Composite App, Java Business Integration (JBI), NetBeans Software Enterprise Pack
- Composite Application Concepts (20 minutes)
    - AppModel, Dev Environment, Architecture, Interface, Component Model, AutoGen, Workflow
- CASA Demos (20 minutes)
    - Edit external and internal connectivity
- **Summary and Q&A (5 minutes)**

Summary

SOA

- JBI = SOA/BI spec (implemented in Java technology)
- SOA = { loosely coupled, interoperable services }
- Service = interface (WSDL) + implementation (hidden)

Comp.App

- CompApp = { connected services (each implements an Aspect) }
- CompApp.dev = NB EntPack(design-time) + Open ESB(run-time)
- CompApp.componentModel SCA|restricted to SOA Services
- CompApp.connectivity = { inter-SU, inter-SA, external }
- CompApp.connection = { consume, provide endpoints }
- CompApp.endpoint = implements a WSDL.portType

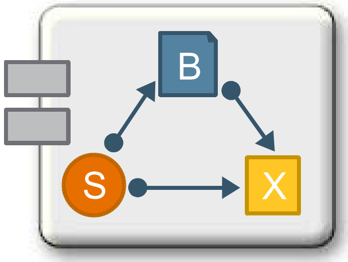
CASA

- CASA = CompApp.connectivity Editor
  - Edit inter-SU, inter-SA, external connections
  - Edit consume and provide endpoints
  - Edit WSDL port concrete attributes
  - Edit WS-Policy definitions attached to WSDL ports

URLs

- NetBeans Software/Enterprise Pack
  - http://www.netbeans.org
  - http://enterprise.netbeans.org (Enterprise Pack)
- Open ESB
  - http://open-esb.org
  - http://www.glassfishwiki.org/jbiwiki/ (Developer Wiki)
  - http://www.jcp.org/en/jsr/detail?id=208 (JBI, JSR-208)
- Blogs
  - http://blogs.sun.com/tientien
  - http://blogs.sun.com/jqian
  - http://blogs.sun.com/joshweb

Q&A



Introduction to CASA: An Open Source Composite Application Editor

Tientien Li, Ph.D. and Jun Qian

Sun Microsystems, Inc.

TS-8683