OpenESB

Keh-Yoe Ong
FAST (Field Assistance Support Team)
Sun Microsystems

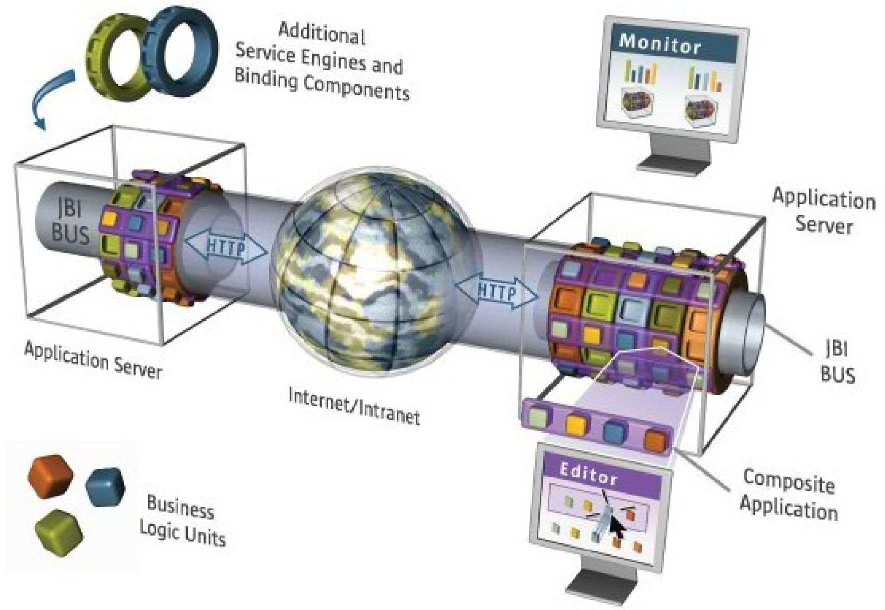
Agenda

- What is OpenESB ?
- What is JBI ?
- JBI and GlassFish
- OpenESB Feature Details
- Deployment Packaging
- Demo
- Summary and Q&A

What is Open ESB?

Open ESB

- Open Source Enterprise Service Bus (ESB) runtime implemented on Java Business Integration (JBI) foundation
- Runs within Glassfish/Sun App Server



What is JBI?

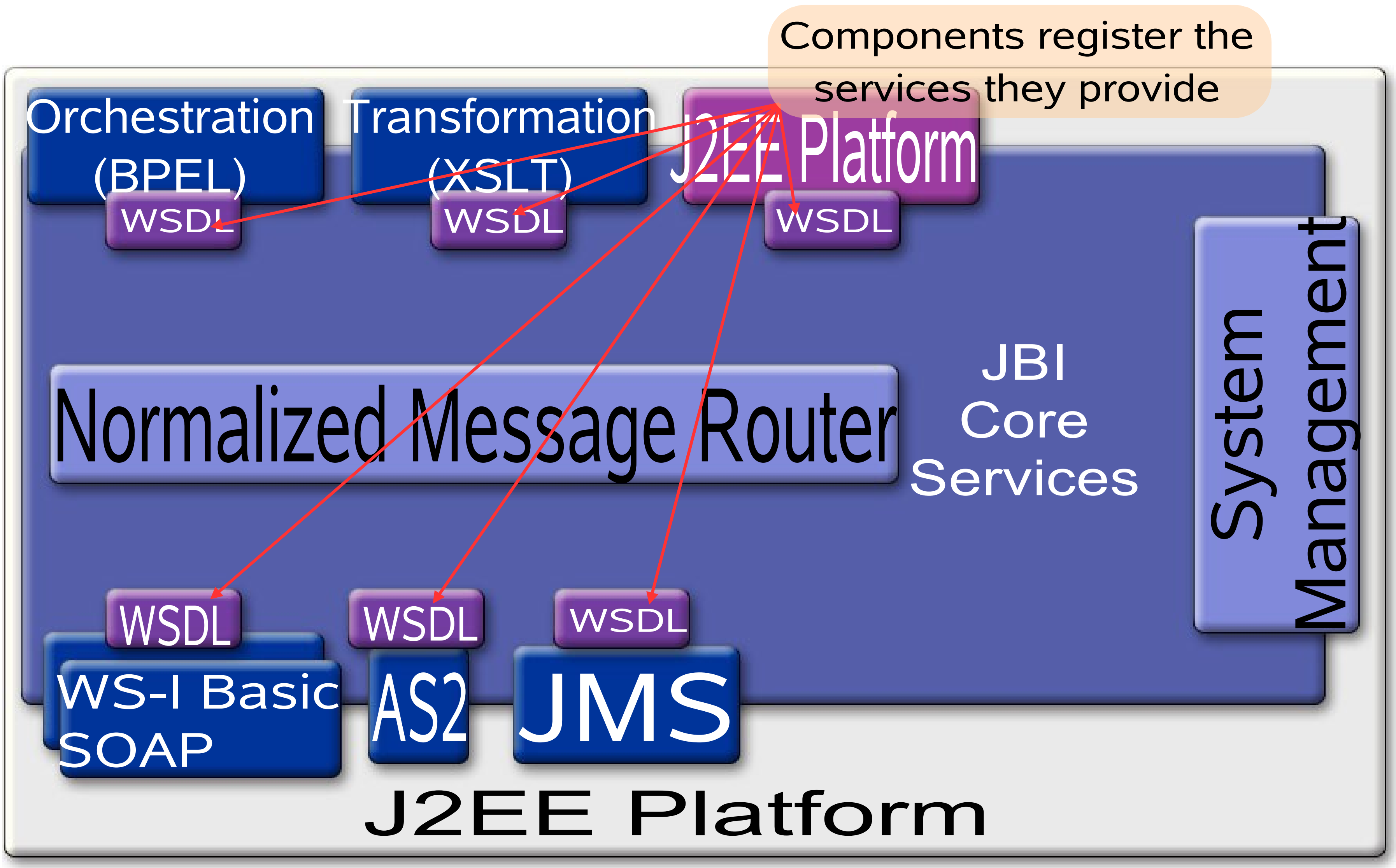
What is JBI ?

- JCP JSR (208) defines an extensible, standards-based integration architecture
- Allows third-party components to be “plugged in” into a standard integration infrastructure
- Components communicate via WSDL-based mediated message exchanges
- Defines a 'meta-container' or 'container of containers'
- Dependency on Java SE/Java EE

Why do we need JBI?

- Point-to-point integration model is not scalable, not easily maintained and lacks flexibility
- The traditional EAI model has its problems –
  - > proprietary Integration Server technology
  - > vendor lock in
  - > high barrier for entry for small, independent, innovative ISVs providing best-of-breed solutions
- Need of an open standard that allows *containers* to inter-operate
- Eliminate 'vendor lock in'

JBI Architecture



JBI Architecture (Components)

JBI Components

- JBI Components
  - Service Engines (SE)
  - Binding Components (BC)
  - Normalized Message Router
  - System Management layer
- SEs and BCs are only logically and functionally different – technically both implement the same interfaces
- SEs and BCs register the services they provide with the JBI framework using WSDL-based service descriptors

The Normalized Message Router

- “Backbone” of JBI
- Facilitates inter-operation between JBI Components using WSDL-based service descriptors
- Service Providers and Consumers are de-coupled
- Components exchange messages based on Message Exchange Patterns defined in WSDL
- Messages exchanged in “Normalized” format
- A normalized message consists of 3 parts
  - payload
  - meta data
  - message attachments

JBI Components

- Service Engines
  - > BPEL SE
  - > XSLT SE
  - > JavaEE SE
  - > IEP SE
  - > ETL SE
  - > SQL SE
  - > Workflow SE
- Binding Comps
  - > MQSeries BC
  - > HL7 BC
  - > SAP BC
  - > SMTP BC
  - > HTTP BC
  - > JMS BC
  - > File BC
  - > CICS BC
  - > DCOM BC
  - > CORBA BC
  - > ...
- Other
  - > Clustering
  - > CASA
  - > JBI Mock
  - > WSIT Tech
- In Progress
  - > CAM
  - > Aspect SE
  - > Encoding SE
  - > Rules SE
  - > Scripting SE

System Management, Administration

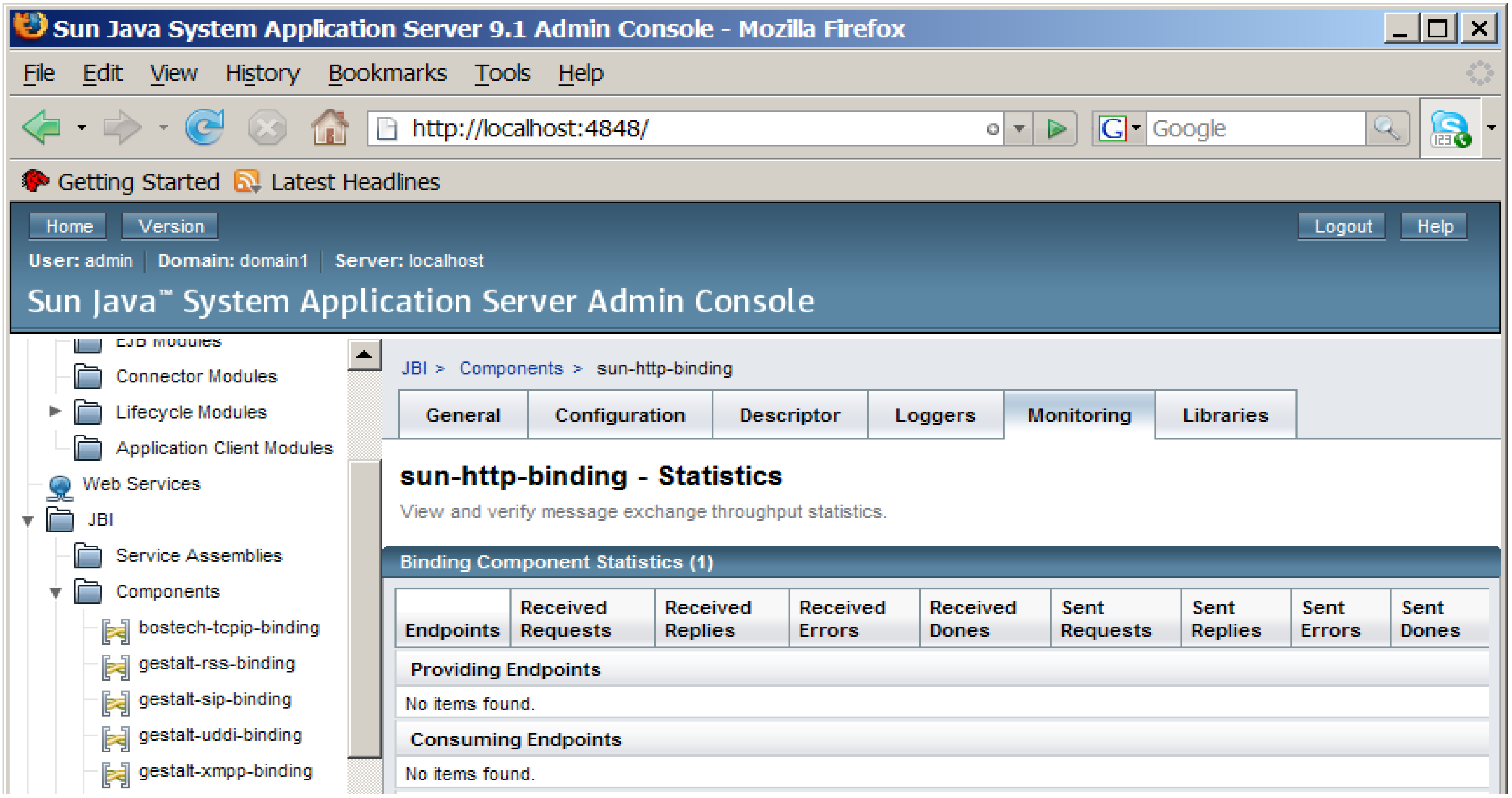
- Java Management eXtensions (JMX) based
- JBI Components and Service Assemblies both have their own life cycles
- Management framework provides services like installation of SEs and BCs
- Provides life cycle services for Composite applications
  - Deploying SAs
  - Starting, Stopping SAs
  - Shutting down SAs
  - Undeploying SAs

JBI and GlassFish

JBI Support in GlassFish

- A JBI runtime has been integrated with GlassFish V2
- JBI extends Glassfish with BI SPI's
- GlassFish admin console now supports JBI
- JBI runtime has been enhanced to adhere to the AppServer clustering architecture
  - Each instance in the appserver cluster will also have a JBI runtime in it
- Java EE Service Engine acts as the bridge between Java EE applications and JBI
- A Java EE application archive (ear/war/jar) can be packaged in a JBI composite application

JBI in Admin Console



Sun Java System Application Server 9.1 Admin Console - Mozilla Firefox

User: admin | Domain: domain1 | Server: localhost

Sun Java™ System Application Server Admin Console

JBI > Components > sun-http-binding

sun-http-binding - Statistics

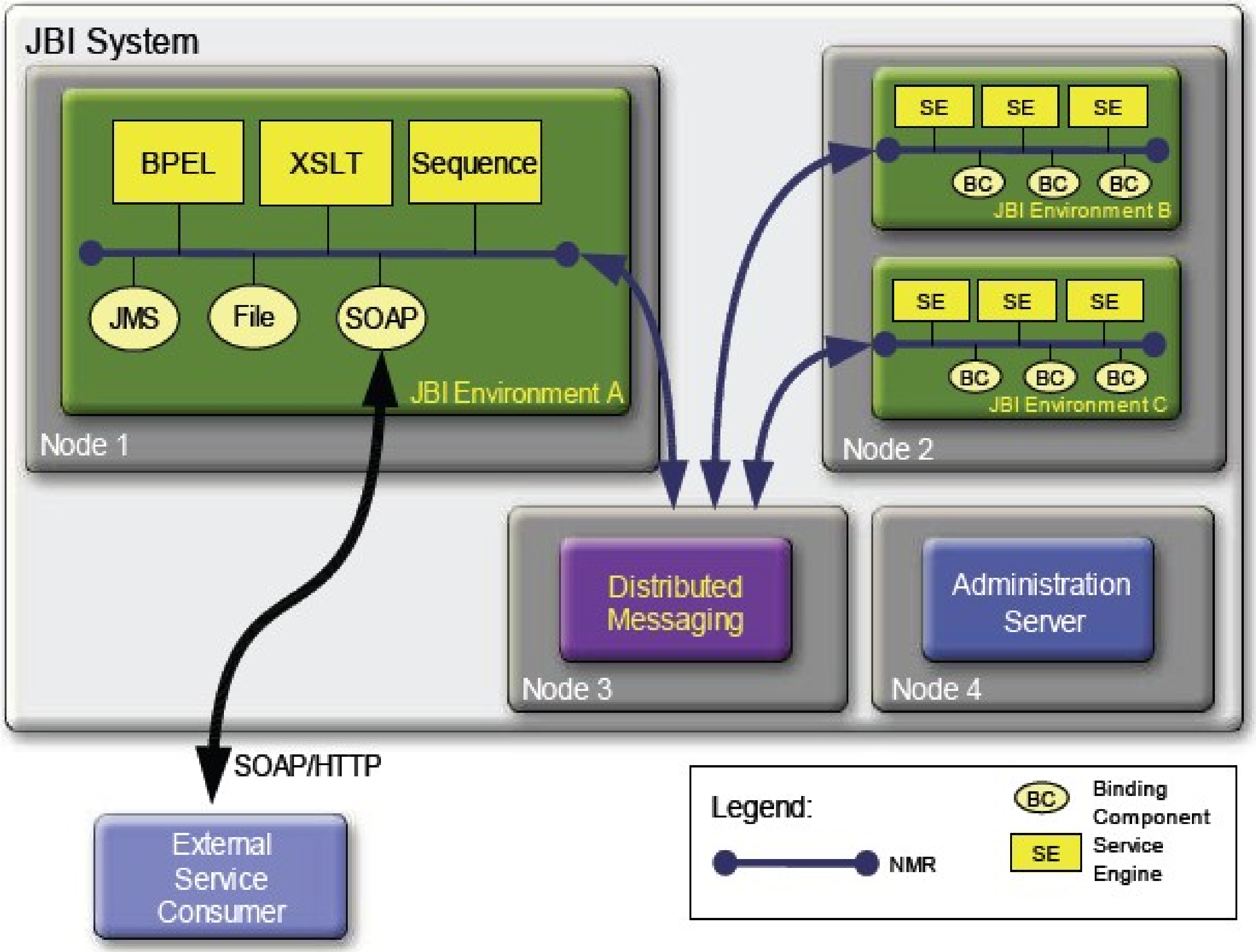
View and verify message exchange throughput statistics.

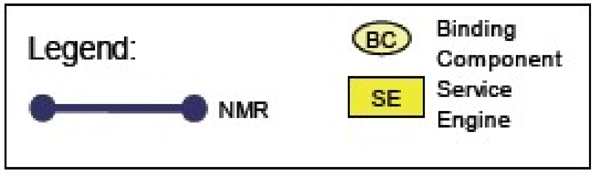
Binding Component Statistics (1)

Endpoints	Received Requests	Received Replies	Received Errors	Received Dones	Sent Requests	Sent Replies	Sent Errors	Sent Dones
Providing Endpoints								
No items found.								
Consuming Endpoints								
No items found.								

OpenESB Feature Details

OpenESB Architecture



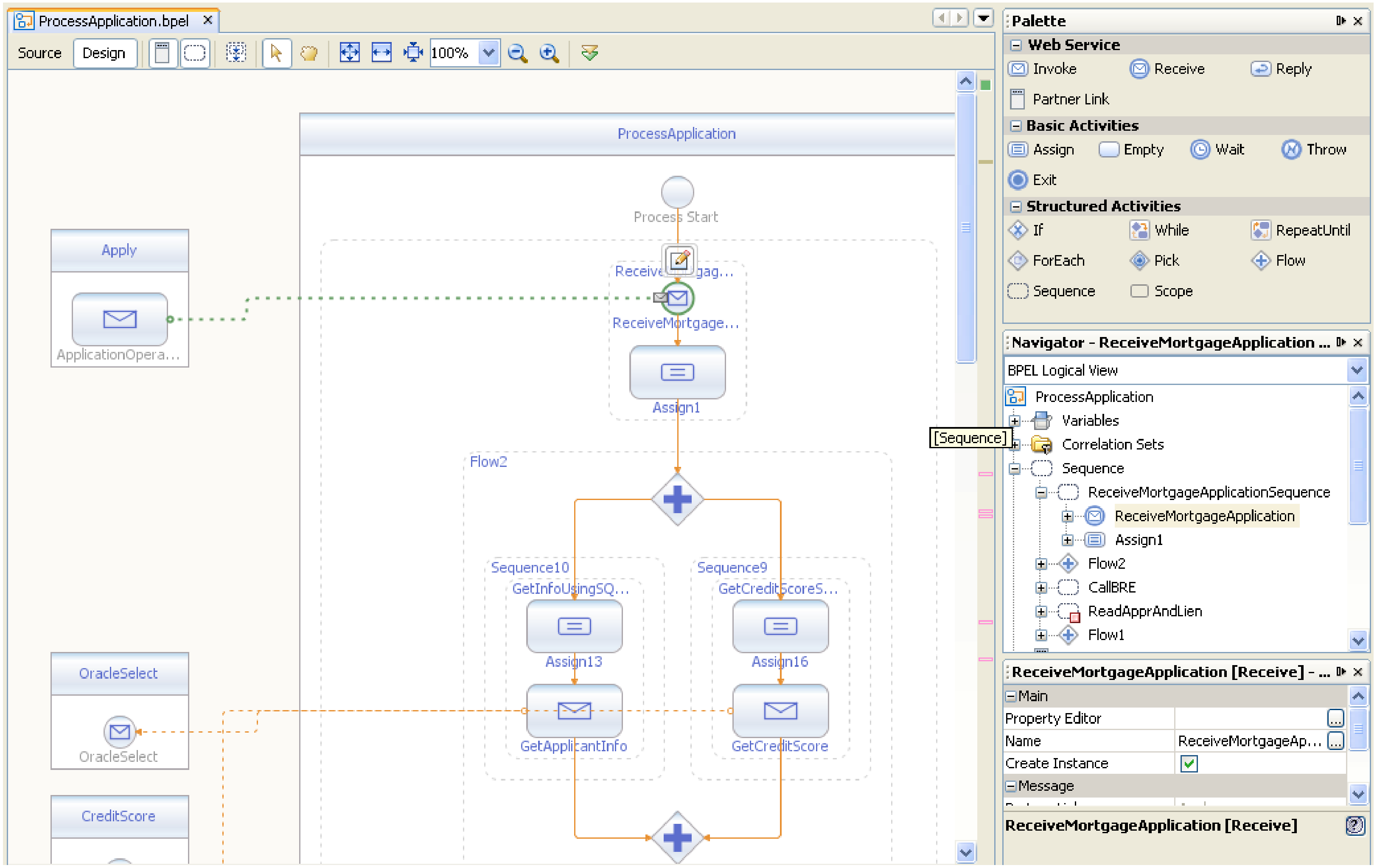


OpenESB Features

- “Killer” Application Server – Glassfish
- Excellent tooling – NetBeans Enterprise Pack
- Fully JBI compliant
- Based on the JBI Reference Implementation
- Latest standards
- Open Source
- Support for Clustering

OpenESB Features

Tooling – BPEL Editor



Palette

Web Service
- Invoke
- Receive
- Reply
- Partner Link

Basic Activities
- Assign
- Empty
- Wait
- Throw
- Exit

Structured Activities
- If
- While
- RepeatUntil
- ForEach
- Pick
- Flow
- Sequence
- Scope

Navigator - ReceiveMortgageApplication ...

BPEL Logical View

```
ProcessApplication
  Variables
  Correlation Sets
  Sequence
    ReceiveMortgageApplicationSequence
      ReceiveMortgageApplication
      Assign1
    Flow2
    CallBRE
    ReadApprAndLien
    Flow1
```

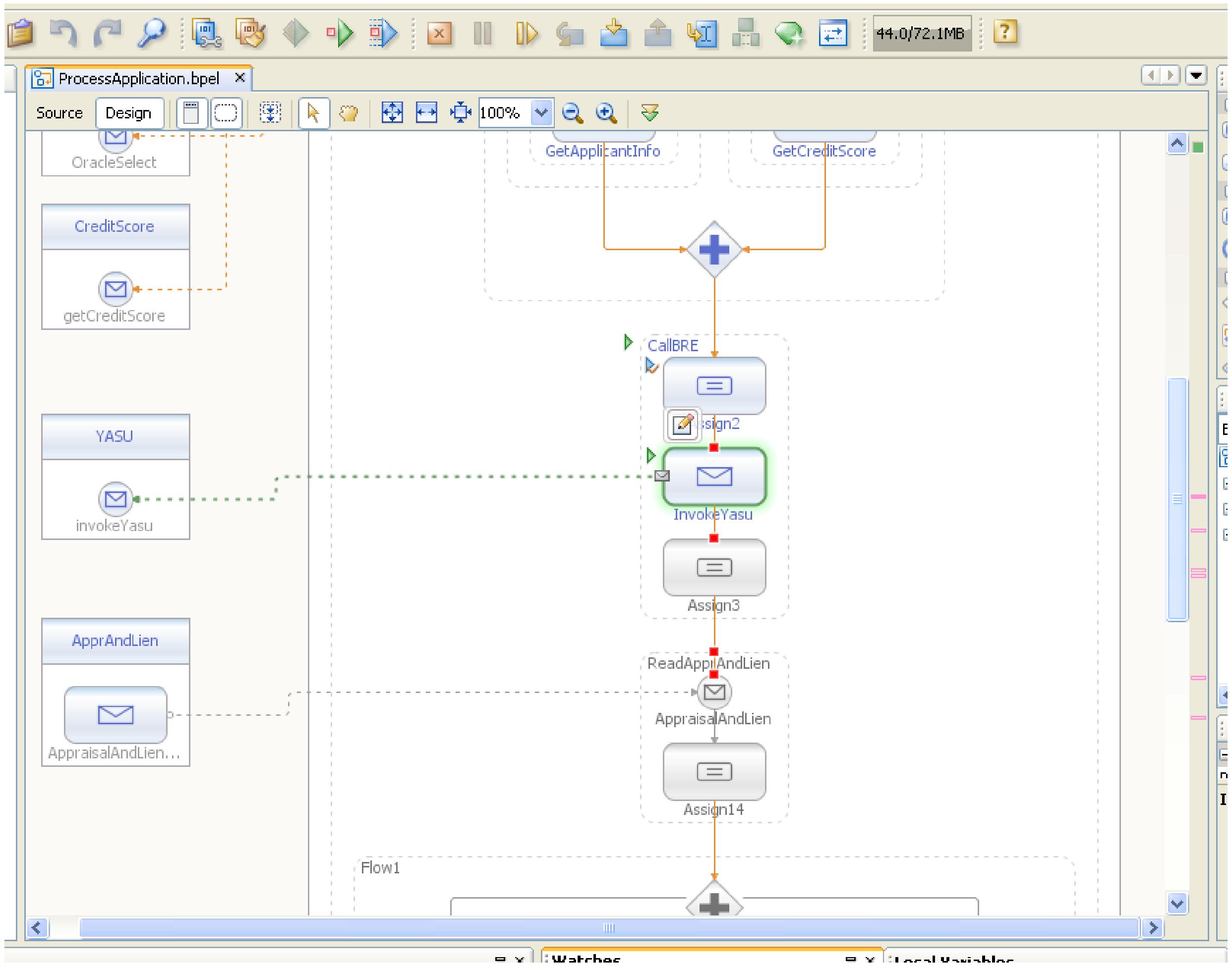
ReceiveMortgageApplication [Receive] - ...

Main	
Property Editor	
Name	ReceiveMortgageAp...
Create Instance	✓
Message	

ReceiveMortgageApplication [Receive]

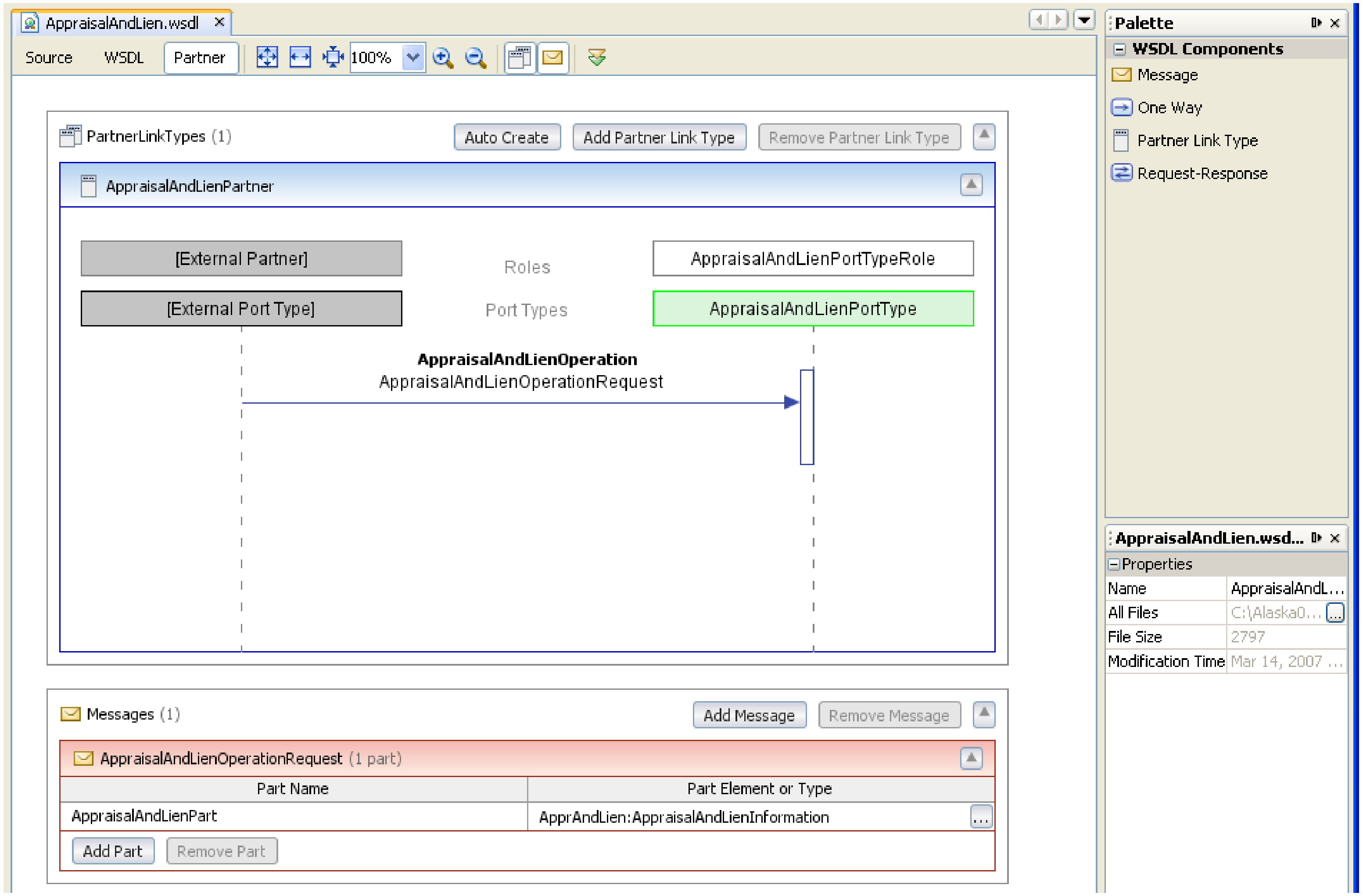
OpenESB Features

Tooling – BPEL Debugger



OpenESB Features

Tooling – WSDL Editor – Graphical View

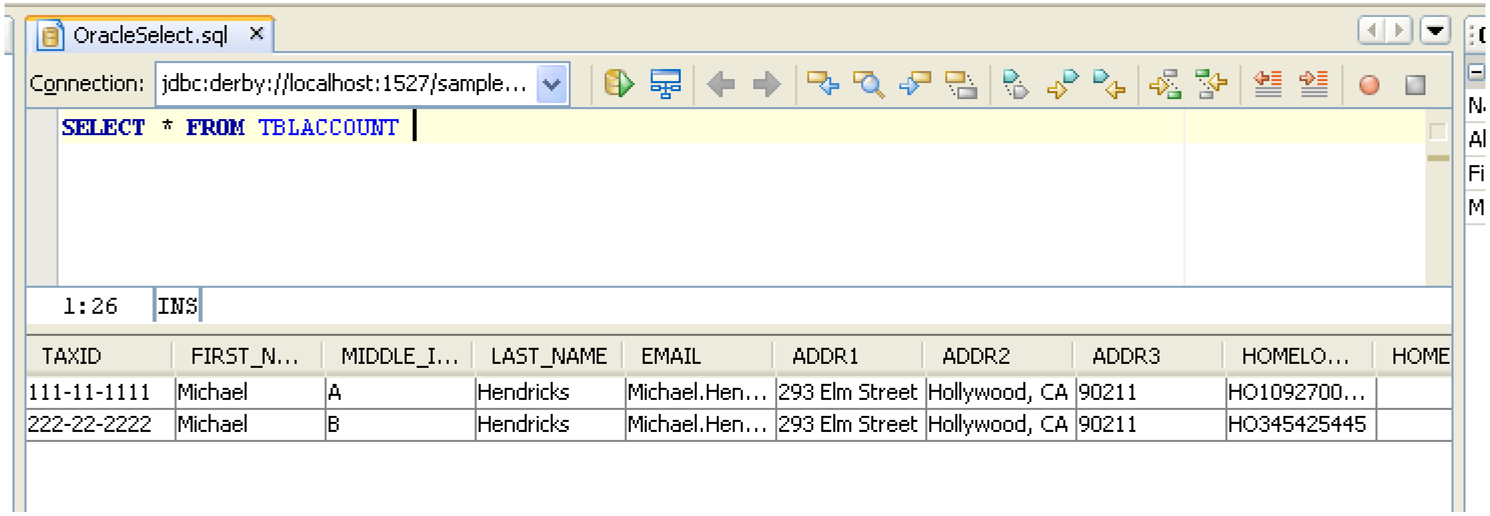


Part Name	Part Element or Type
AppraisalAndLienPart	ApprAndLien:AppraisalAndLienInformation

Name	AppraisalAndL...
All Files	C:\Alaska0...
File Size	2797
Modification Time	Mar 14, 2007 ...

OpenESB Features

Tooling – SQL Editor

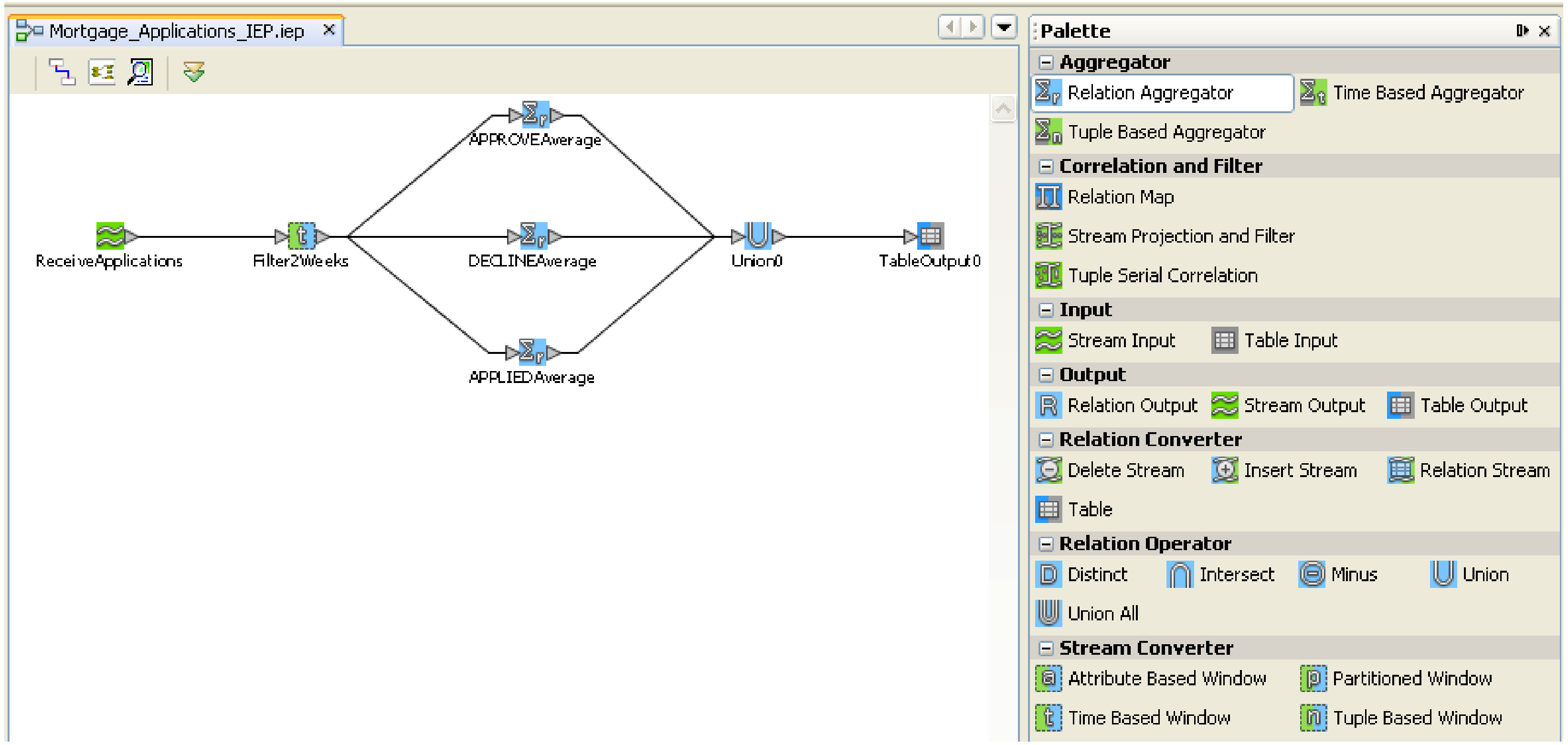


```
SELECT * FROM TBLACCOUNT
```

TAXID	FIRST_N...	MIDDLE_I...	LAST_NAME	EMAIL	ADDR1	ADDR2	ADDR3	HOMELO...	HOME
111-11-1111	Michael	A	Hendricks	Michael.Hen...	293 Elm Street	Hollywood, CA	90211	HO1092700...	
222-22-2222	Michael	B	Hendricks	Michael.Hen...	293 Elm Street	Hollywood, CA	90211	HO345425445	

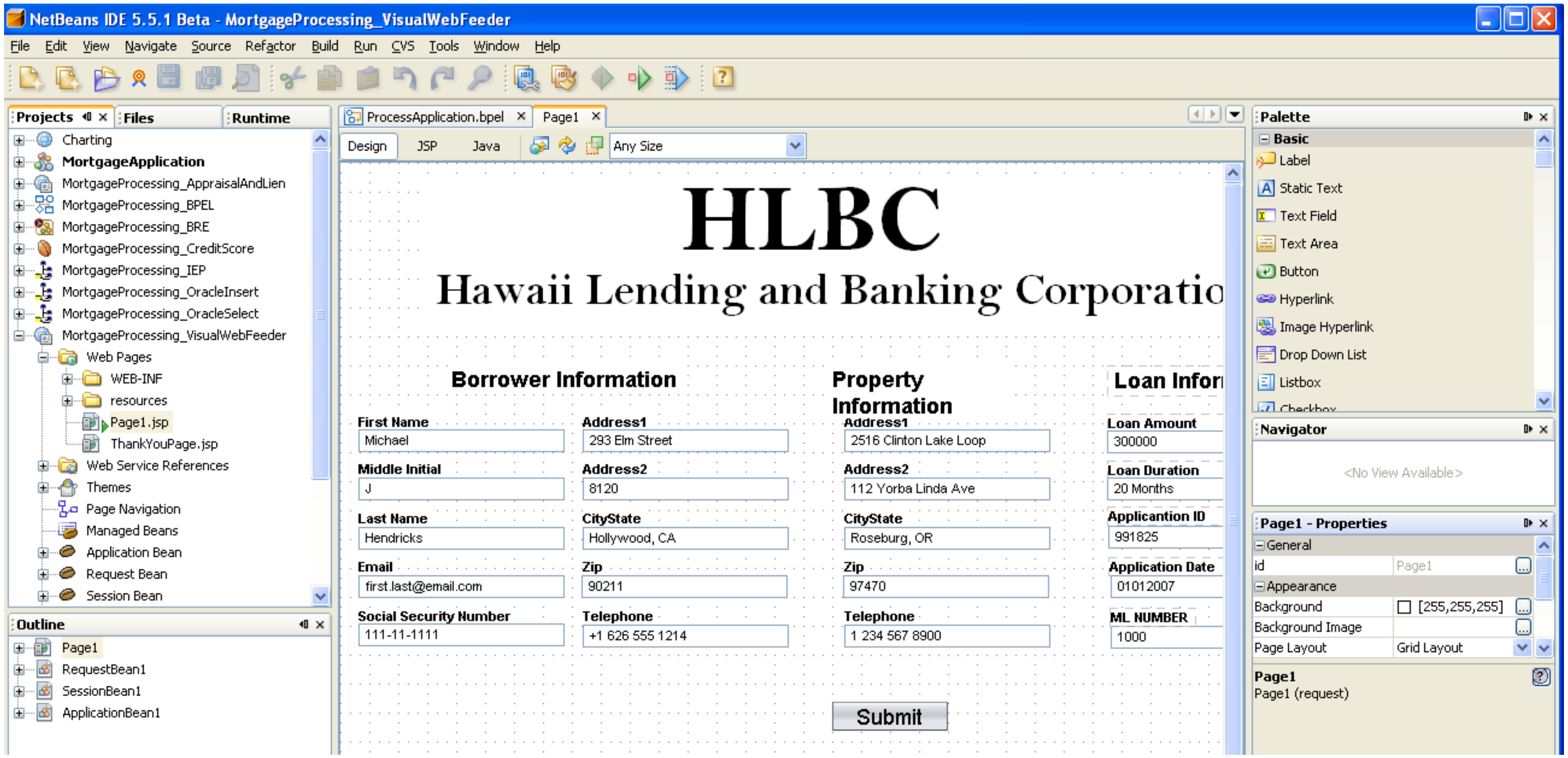
OpenESB Features

Tooling – IEP Editor



OpenESB Features

Tooling – Visual Web Editor



Projects:
- Charting
- **MortgageApplication**
- MortgageProcessing_AppraisalAndLien
- MortgageProcessing_BPEL
- MortgageProcessing_BRE
- MortgageProcessing_CreditScore
- MortgageProcessing_IEP
- MortgageProcessing_OracleInsert
- MortgageProcessing_OracleSelect
- MortgageProcessing_VisualWebFeeder
  - Web Pages
    - WEB-INF
    - resources
    - Page1.jsp
    - ThankYouPage.jsp
  - Web Service References
  - Themes
  - Page Navigation
  - Managed Beans
  - Application Bean
  - Request Bean
  - Session Bean

Outline:
- Page1
- RequestBean1
- SessionBean1
- ApplicationBean1

HLBC

Hawaii Lending and Banking Corporatio

Borrower Information

First Name	Michael
Middle Initial	J
Last Name	Hendricks
Email	first.last@email.com
Social Security Number	111-11-1111
Address1	293 Elm Street
Address2	8120
CityState	Hollywood, CA
Zip	90211
Telephone	+1 626 555 1214

Property Information

Address1	2516 Clinton Lake Loop
Address2	112 Yorba Linda Ave
CityState	Roseburg, OR
Zip	97470
Telephone	1 234 567 8900

Loan Infor

Loan Amount	300000
Loan Duration	20 Months
Applicantion ID	991825
Application Date	01012007
ML NUMBER	1000

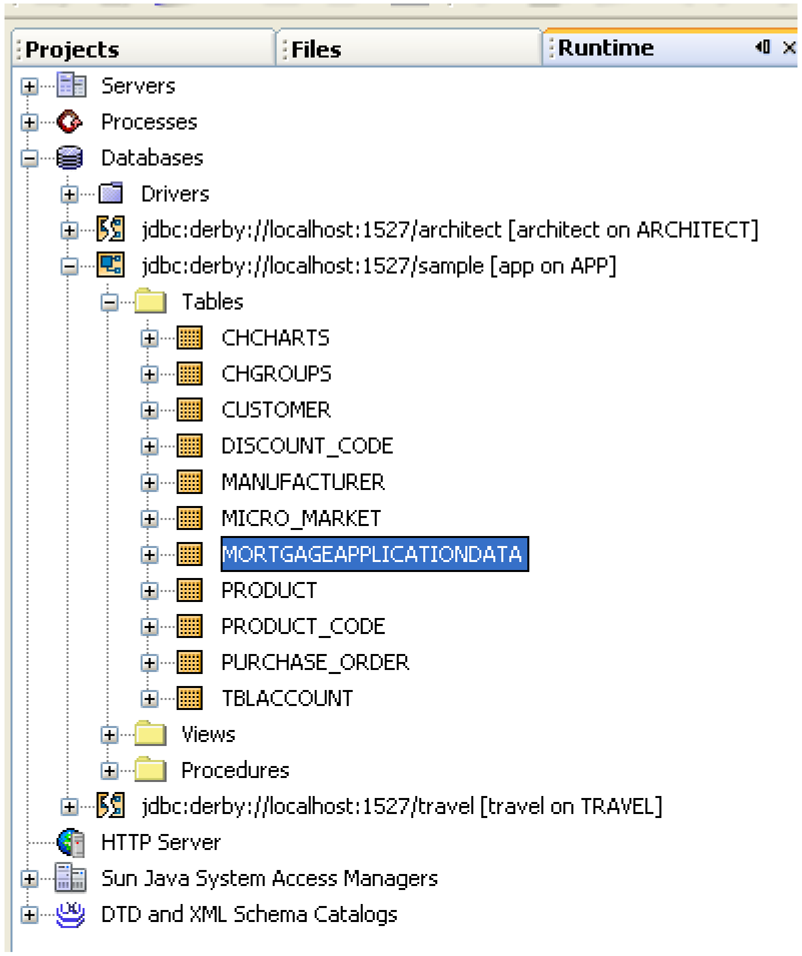
Page1 - Properties

General	
id	Page1
Appearance	
Background	[255,255,255]
Background Image	
Page Layout	Grid Layout

Page1
Page1 (request)

OpenESB Features

Tooling – Derby (Java DB) Front-End

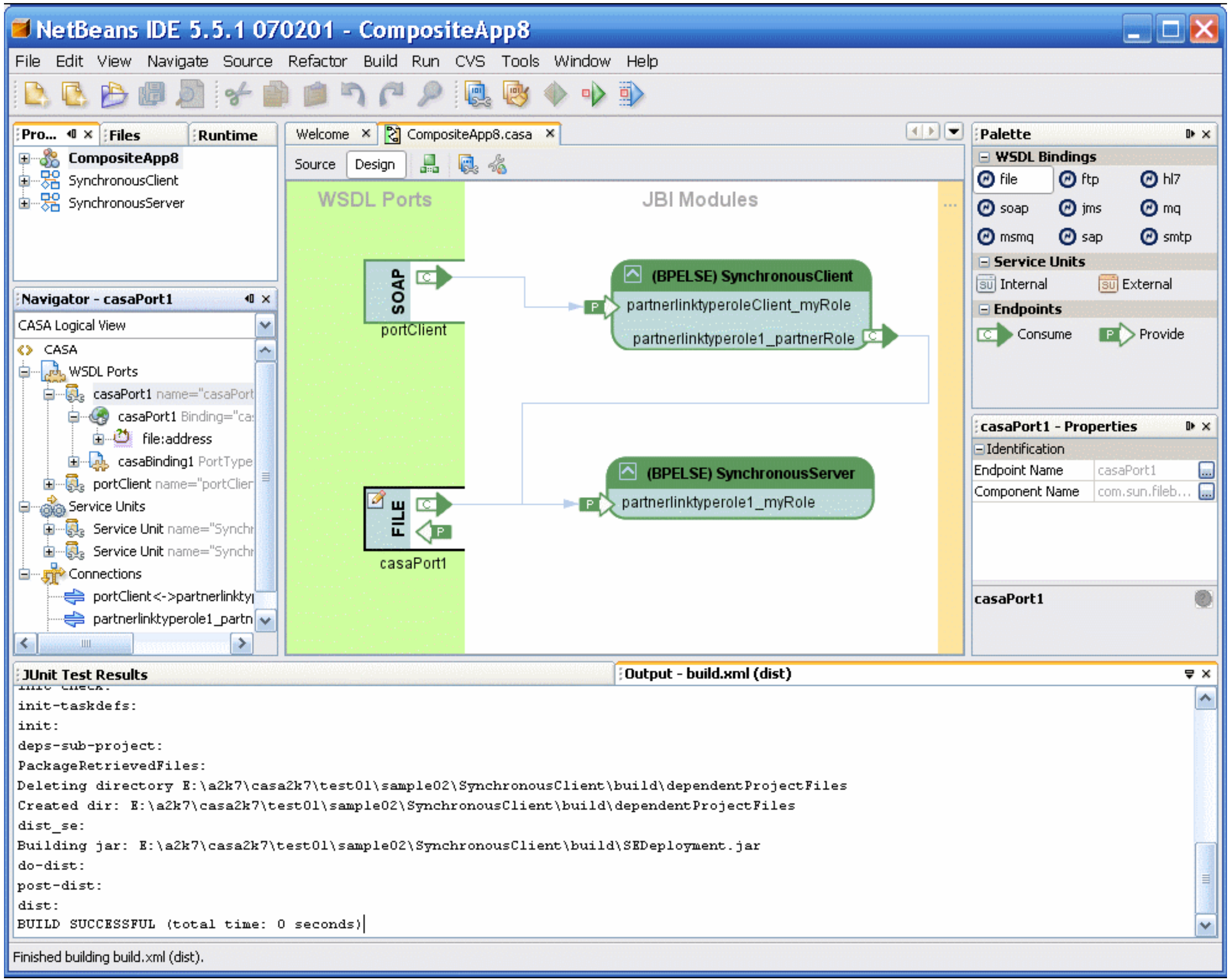


Projects | Files | Runtime

```
Servers
Processes
Databases
    Drivers
    jdbc:derby://localhost:1527/architect [architect on ARCHITECT]
    jdbc:derby://localhost:1527/sample [app on APP]
        Tables
            CHCHARTS
            CHGROUPS
            CUSTOMER
            DISCOUNT_CODE
            MANUFACTURER
            MICRO_MARKET
            MORTGAGEAPPLICATIONDATA
            PRODUCT
            PRODUCT_CODE
            PURCHASE_ORDER
            TBLACCOUNT
        Views
        Procedures
    jdbc:derby://localhost:1527/travel [travel on TRAVEL]
HTTP Server
Sun Java System Access Managers
DTD and XML Schema Catalogs
```

OpenESB Features

Tooling – Visual Service Assembly editor (CASA)



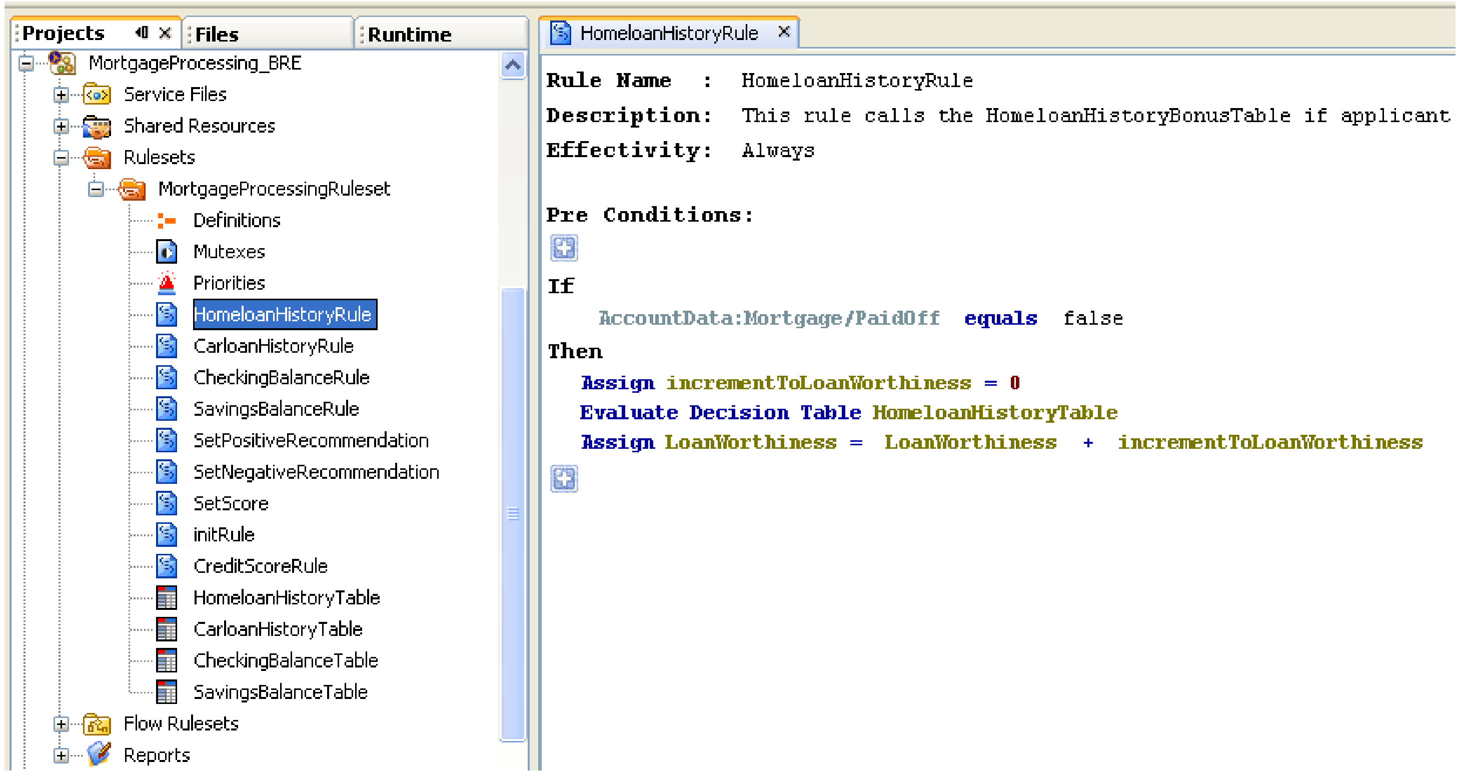
```
init-taskdefs:
init:
deps-sub-project:
PackageRetrievedFiles:
Deleting directory E:\a2k7\casa2k7\test01\sample02\SynchronousClient\build\dependentProjectFiles
Created dir: E:\a2k7\casa2k7\test01\sample02\SynchronousClient\build\dependentProjectFiles
dist_se:
Building jar: E:\a2k7\casa2k7\test01\sample02\SynchronousClient\build\SEDeployment.jar
do-dist:
post-dist:
dist:
BUILD SUCCESSFUL (total time: 0 seconds)
```

Finished building build.xml (dist).

OpenESB Features

Tooling – Other

- JUnit testing tooling, Logging
- Partner tooling (Yasu Editors)



```
Rule Name   :   HomeloanHistoryRule
Description:   This rule calls the HomeloanHistoryBonusTable if applicant
Effectivity:   Always

Pre Conditions:

If
    AccountData:Mortgage/PaidOff  equals  false
Then
   Assign incrementToLoanWorthiness = 0
   Evaluate Decision Table HomeloanHistoryTable
   Assign LoanWorthiness =  LoanWorthiness  +  incrementToLoanWorthiness
```

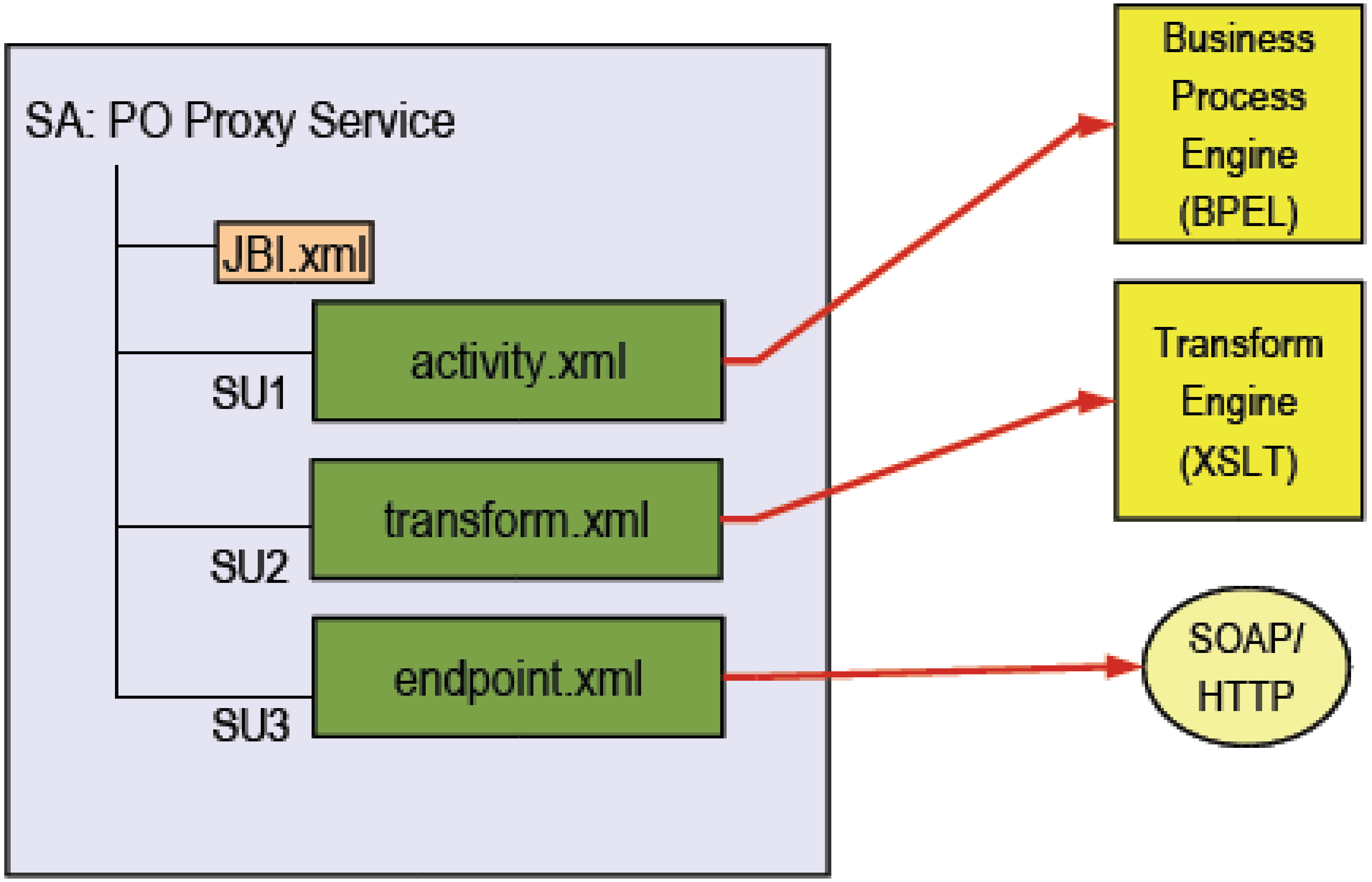
Deployment Packaging

Deployment Packaging

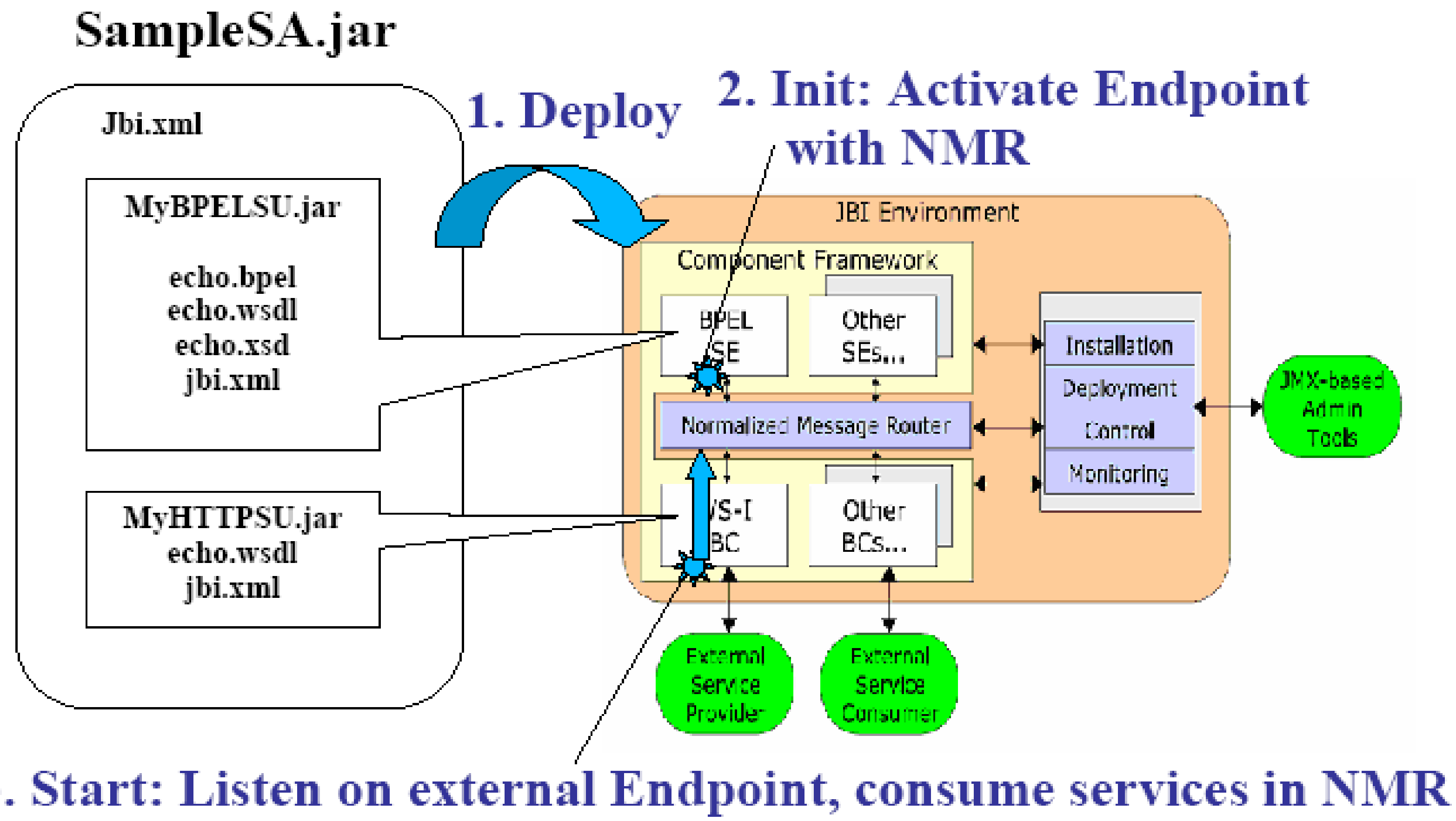
- Service Units group together artifacts meant to be deployed to a particular container/component
- The contents are opaque to JBI except for a descriptor
  - > The descriptor declares what services are consumed/provisioned with the deployment of this SU
- Service Assemblies group together Service Units
  - > Descriptor defines which SU should be deployed to which component
  - > Can declare service connections to affect routing
- The SU and SA packaging is only a design time concept

Composite App: Service Assembly

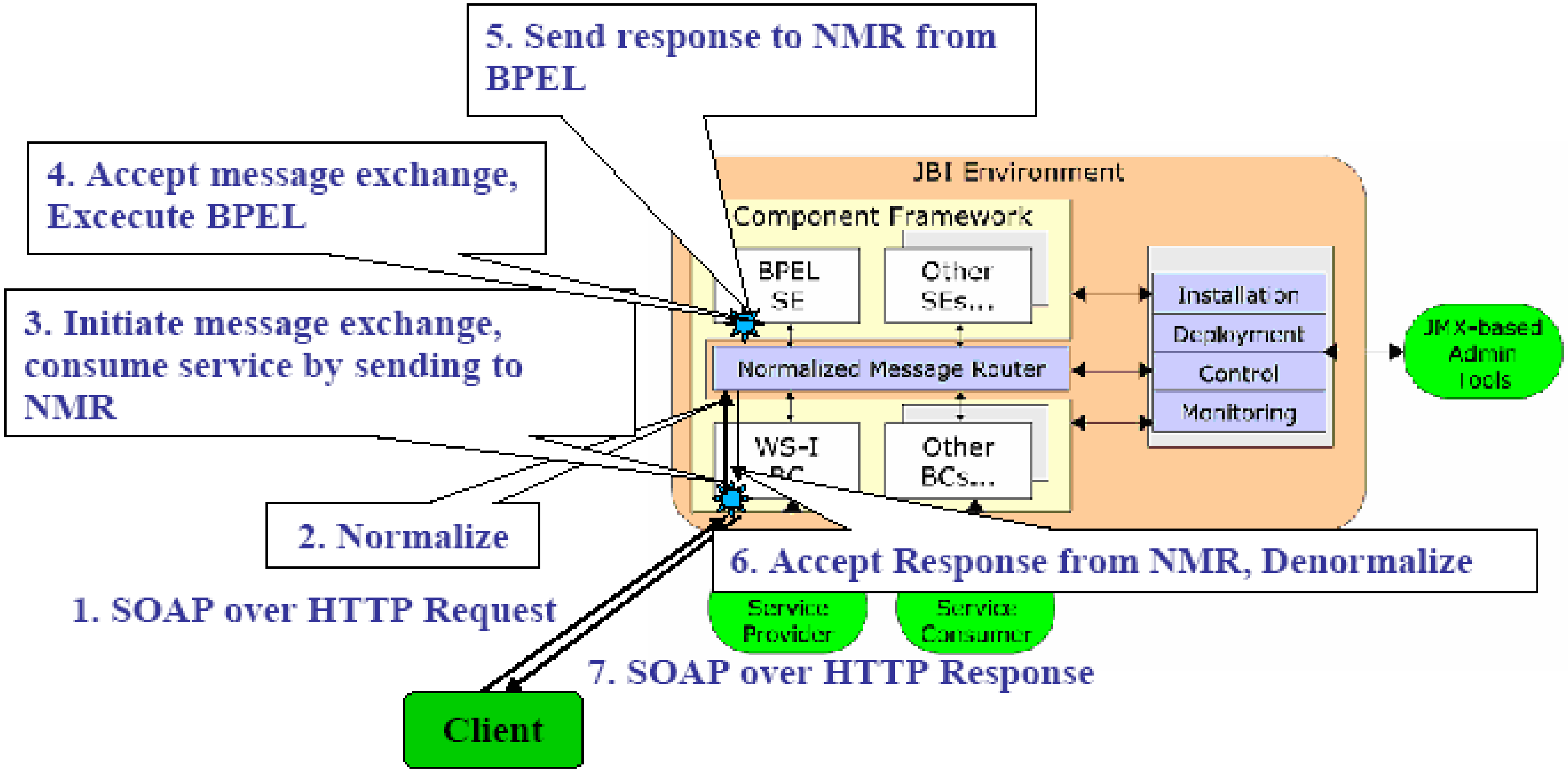
A Collection of Service Units



Example Deployment



Example Message Exchange


5. Send response to NMR from BPEL
4. Accept message exchange, Excecute BPEL
3. Initiate message exchange, consume service by sending to NMR
2. Normalize
1. SOAP over HTTP Request
6. Accept Response from NMR, Denormalize
7. SOAP over HTTP Response
Client
JBI Environment
Component Framework
BPEL SE
Other SEs...
Normalized Message Router
WS-I BC
Other BCs...
Installation
Deployment
Control
Monitoring
JMX-based Admin Tools
Service Provider
Service Consumer

Demo

Summary and Q&A

Product Information

- **Java CAPS 5.1.3**
  - Sun's commercial Composite Application Platform **product** offering
- **Open ESB**
  - Open source **community** project bringing together JBI, JBI components, GlassFish, and NetBeans
- **Java CAPS 5.2**
  - Sun's next generation Composite Application Platform **product** offering based on Open ESB

Summary

- JBI (JCP JSR 208) defines an integration architecture – components plug-in
- JBI Container is a 'container of containers'
- Business and communication protocol components decoupled and hosted in independent containers
- Components expose interface through WSDL
- Complete service or application can be created by composing business and communication logic
- OpenESB facilitates lifecycle management of such “composite applications”

Q&A

Any Questions ?

Thanks

Keh-Yoe Ong
FAST (Field Assistance Support Team)
Sun Microsystems

References

- JBI (JSR 208) specification area on JCP website - http://jcp.org/en/jsr/detail?id=208
- The OpenESB website on sun.com - http://java.sun.com/integration/
- Download location for the latest OpenESB bits - http://java.sun.com/javaee/downloads/ea/ (choose 'Download with tools')
- The project Open ESB website - http://open-esb.org
- The JBI Wiki area on Glassfish wiki - http://www.glassfishwiki.org/jbiwiki/
- Blogs – LOTS of them - http://blogs.sun.com
- Open ESB open source project mailing lists - https://open-esb.dev.java.net/servlets/ProjectMailingListList
- The “Java Business Integration” discussion forum on forums.sun.com - http://forum.java.sun.com/forum.jspa?forumID=512