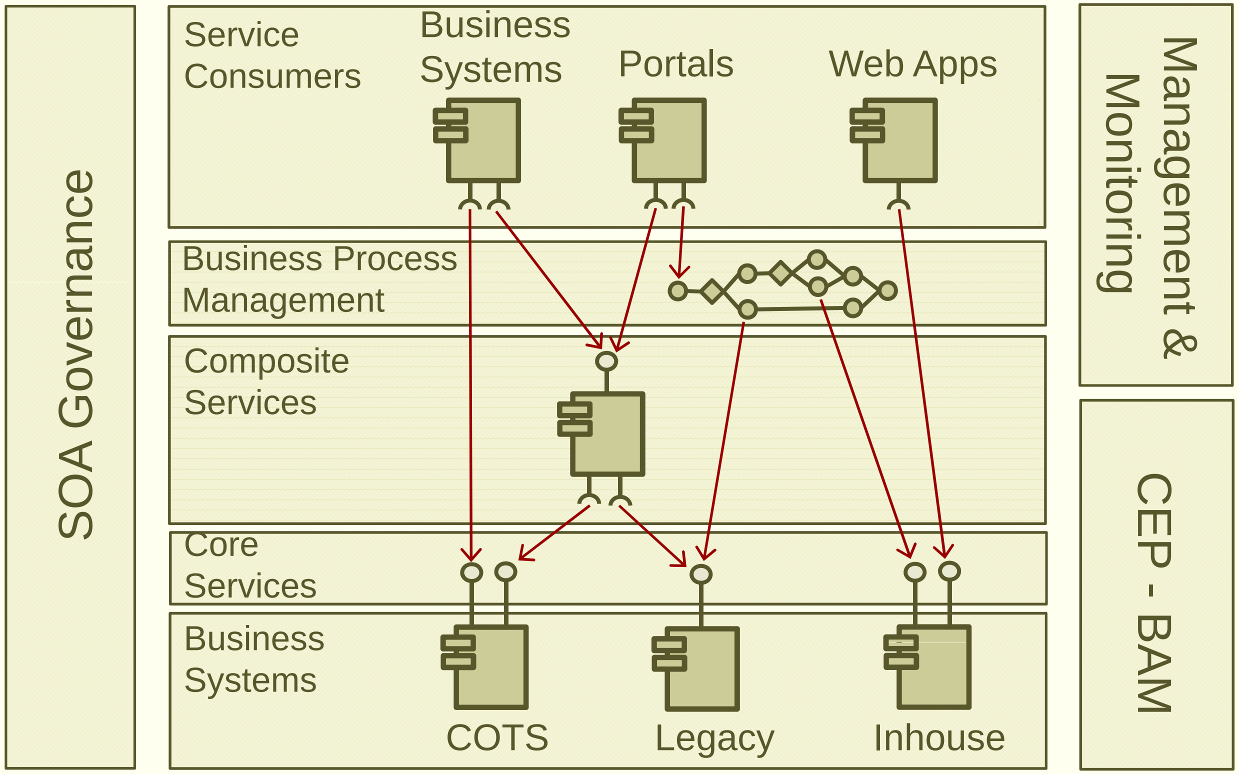
SOA and Open Source



Magnus Larsson

Callista Enterprise AB

CALLISTA

Vendor support of Open Source SOA

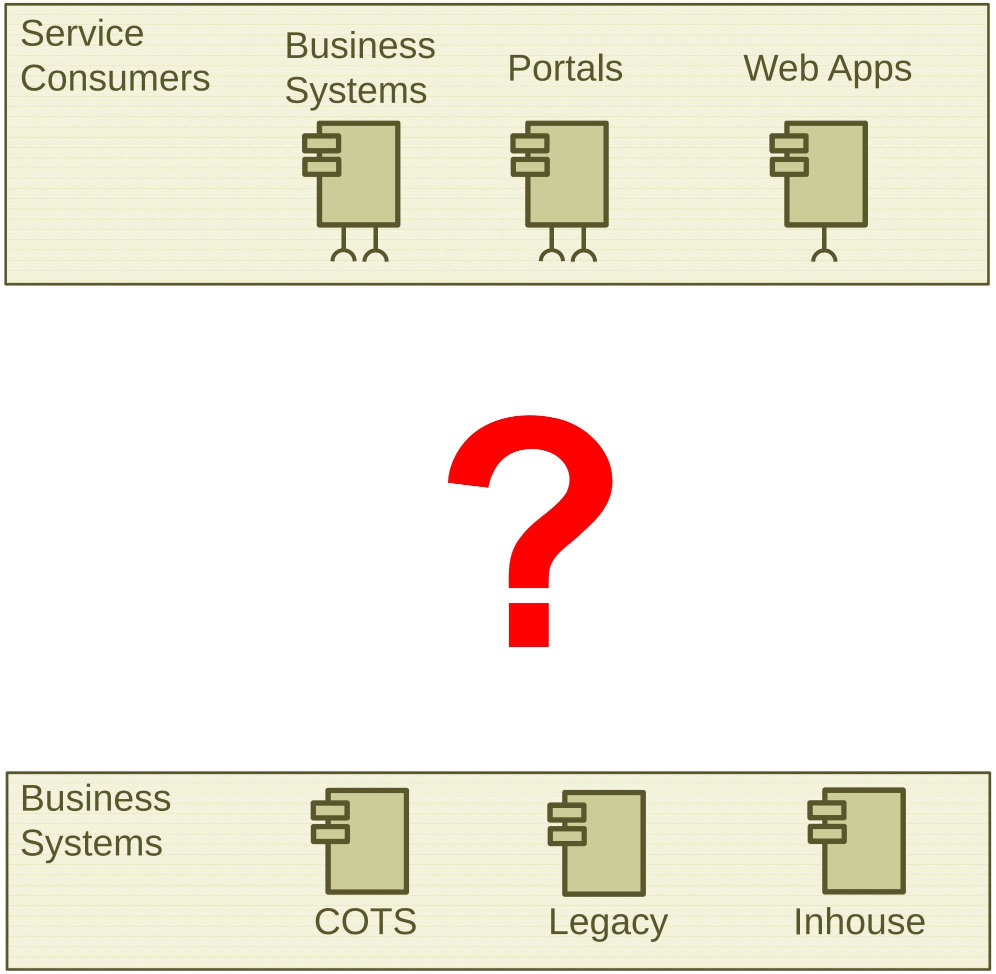
- Vendors provide services for training, consulting and support on selected Open Source SOA products

- MuleSource
  - Over 1000 mission-critical production installations worldwide!
  - http://www.mulesource.com/customers/casestudies.php

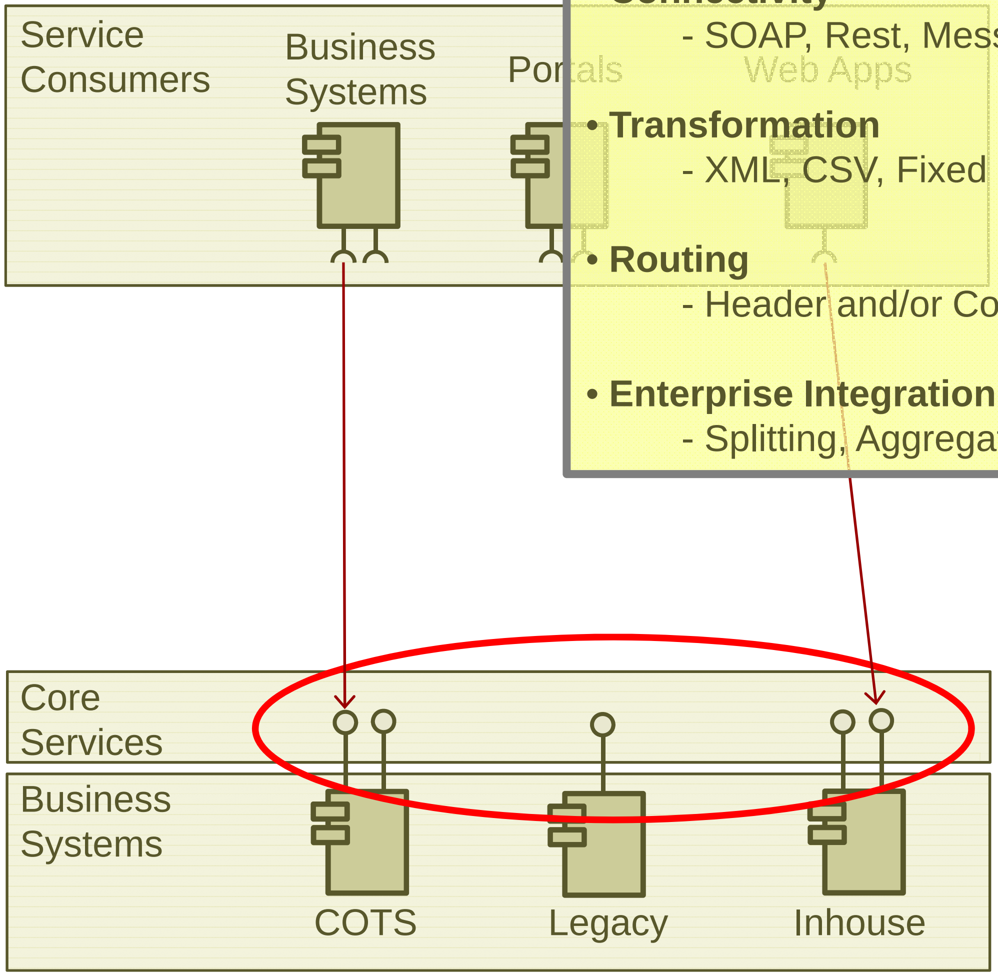
- WSO2
  - http://wso2.com/about/whitepapers/

- Progress FUSE
  - http://fusesource.com/resources/collateral/

Building a SOA Reference Model...

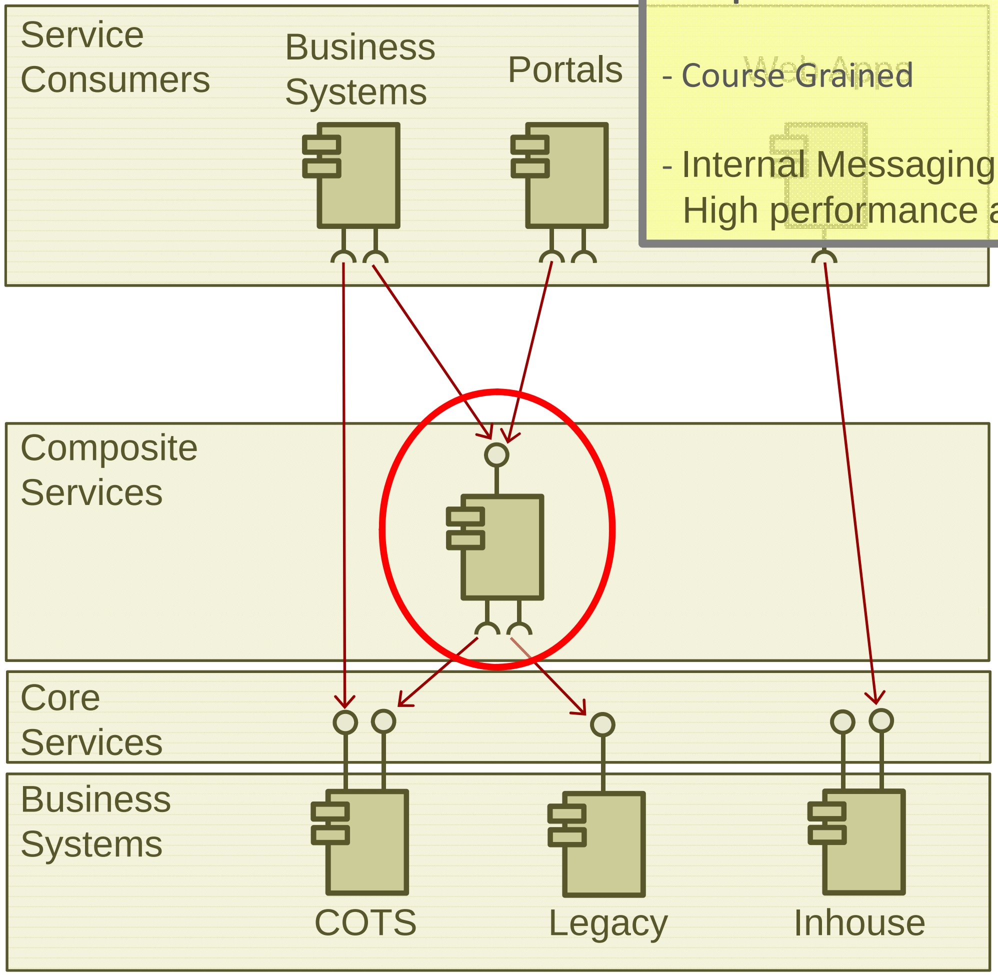


Building a SOA Reference Model...



- **Connectivity**
  - SOAP, Rest, Messaging, Database, FTP…
- **Transformation**
  - XML, CSV, Fixed Position…
- **Routing**
  - Header and/or Content based
- **Enterprise Integration Patterns**
  - Splitting, Aggregation, Resequencing…

Building a SOA Reference Model...

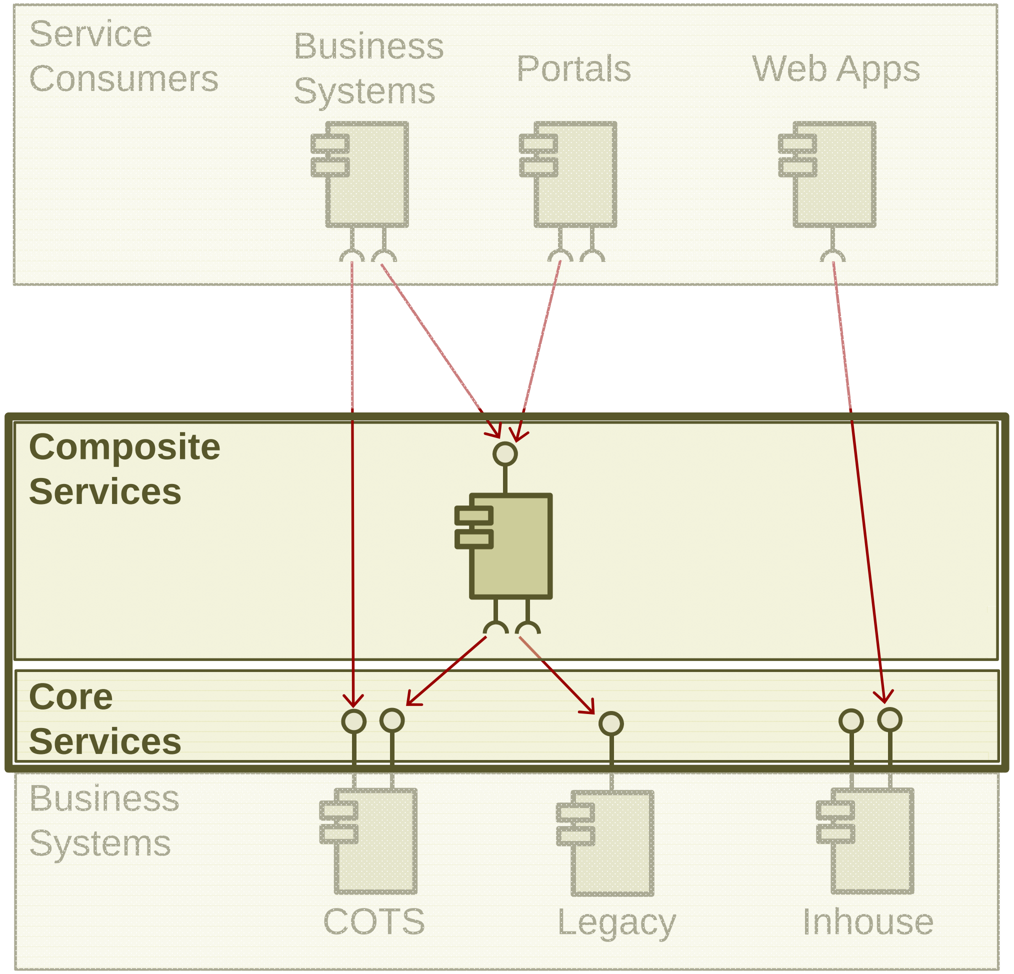


Composite Services

- Course Grained

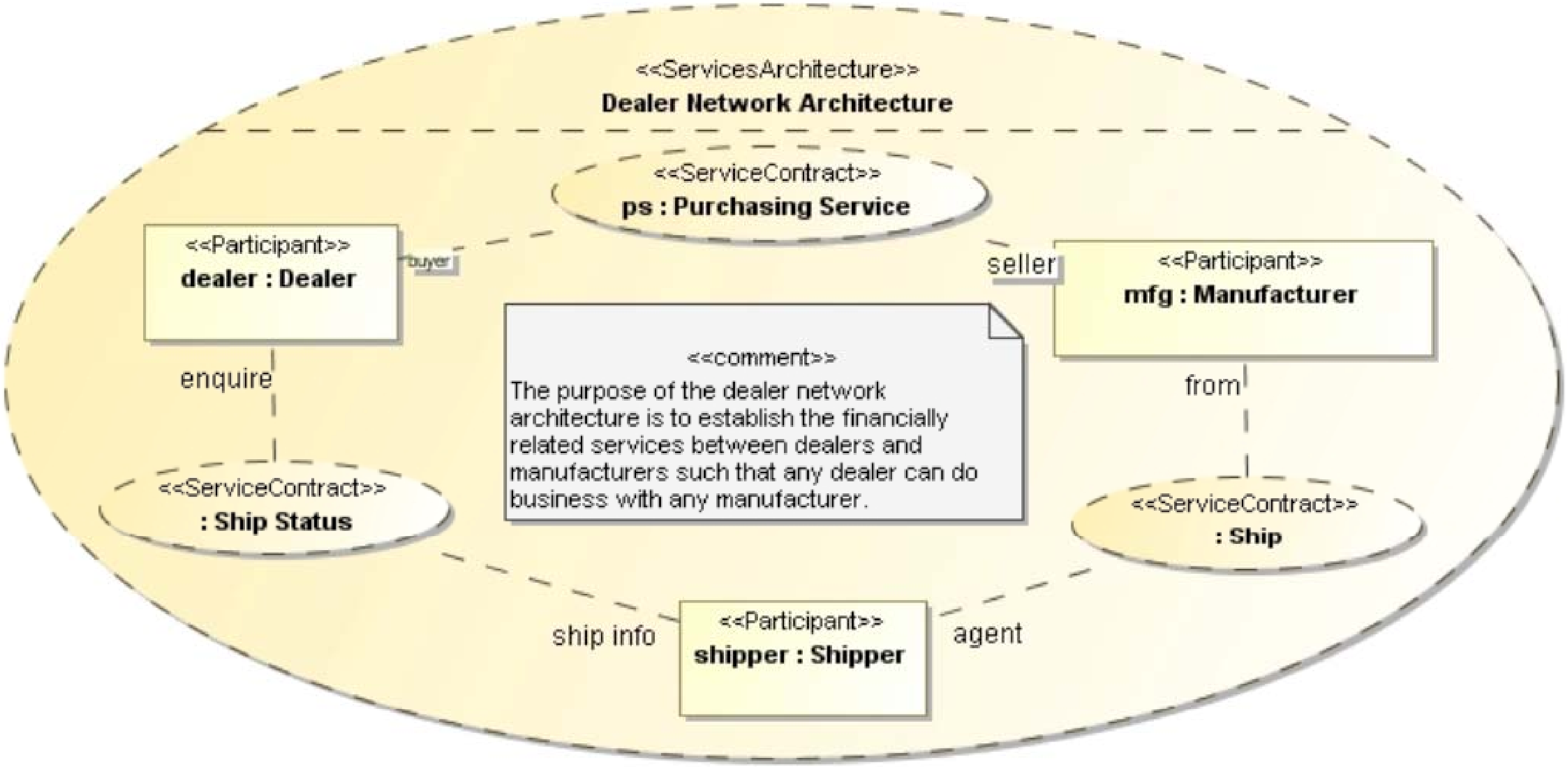
- Internal Messaging
  High performance access to other services

ESB - Enterprise Service Bus – The hart of SOA



Enterprise Service Bus Example

- **Example:** Purchase Order Process



<<ServicesArchitecture>>
Dealer Network Architecture

<<ServiceContract>>
ps : Purchasing Service

<<Participant>>
dealer : Dealer

<<Participant>>
mfg : Manufacturer

<<comment>>
The purpose of the dealer network architecture is to establish the financially related services between dealers and manufacturers such that any dealer can do business with any manufacturer.

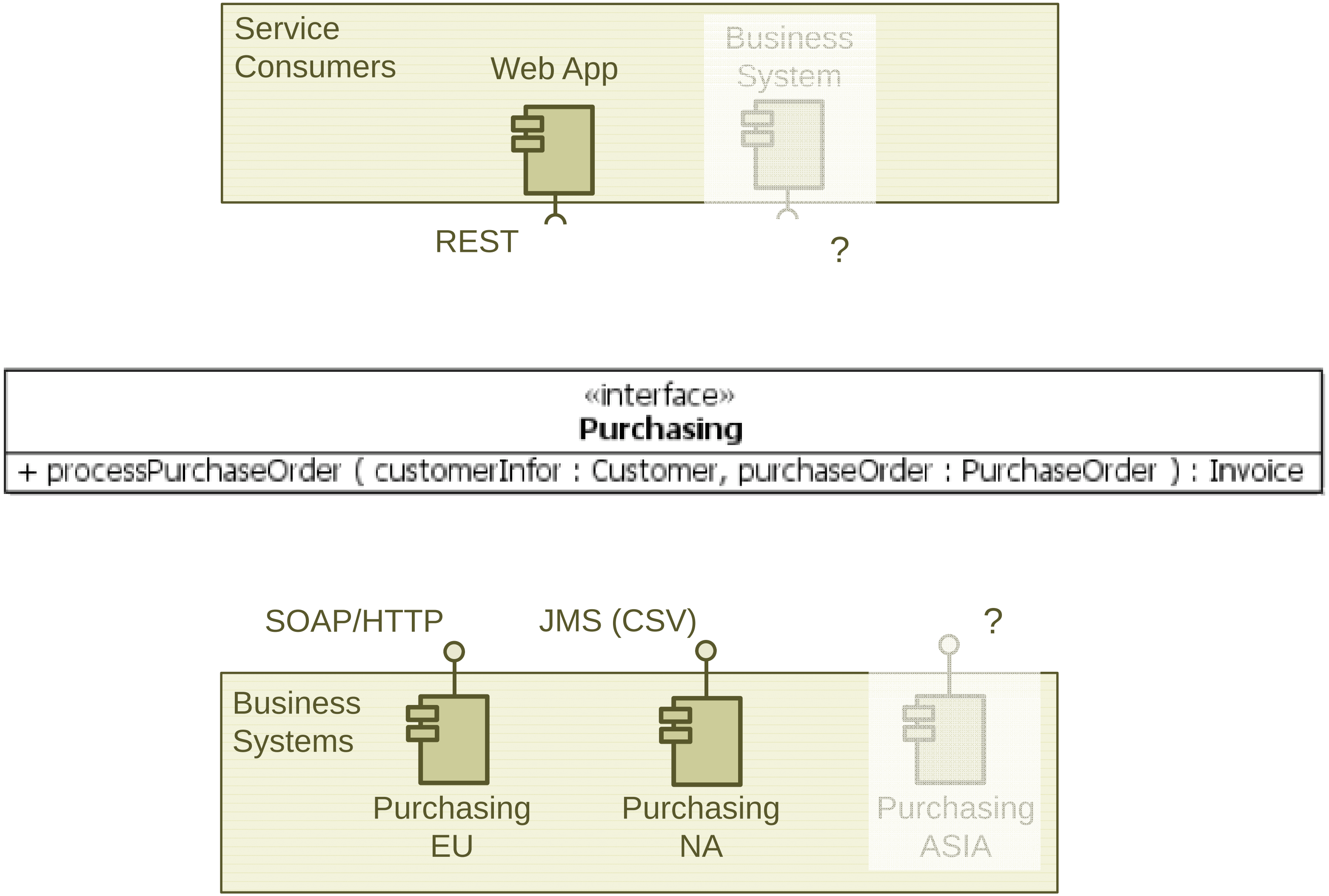
<<ServiceContract>>
: Ship Status

<<ServiceContract>>
: Ship

<<Participant>>
shipper : Shipper

  - From SOA Modeling Language specification (UML Profile)
    www.omg.org/docs/ad/08-08-04.pdf

Enterprise Service Bus Example



Service Consumers

Web App

Business System

REST

?

«interface»
Purchasing

+ processPurchaseOrder (customerInfor : Customer, purchaseOrder : PurchaseOrder) : Invoice

SOAP/HTTP

JMS (CSV)

?

Business Systems

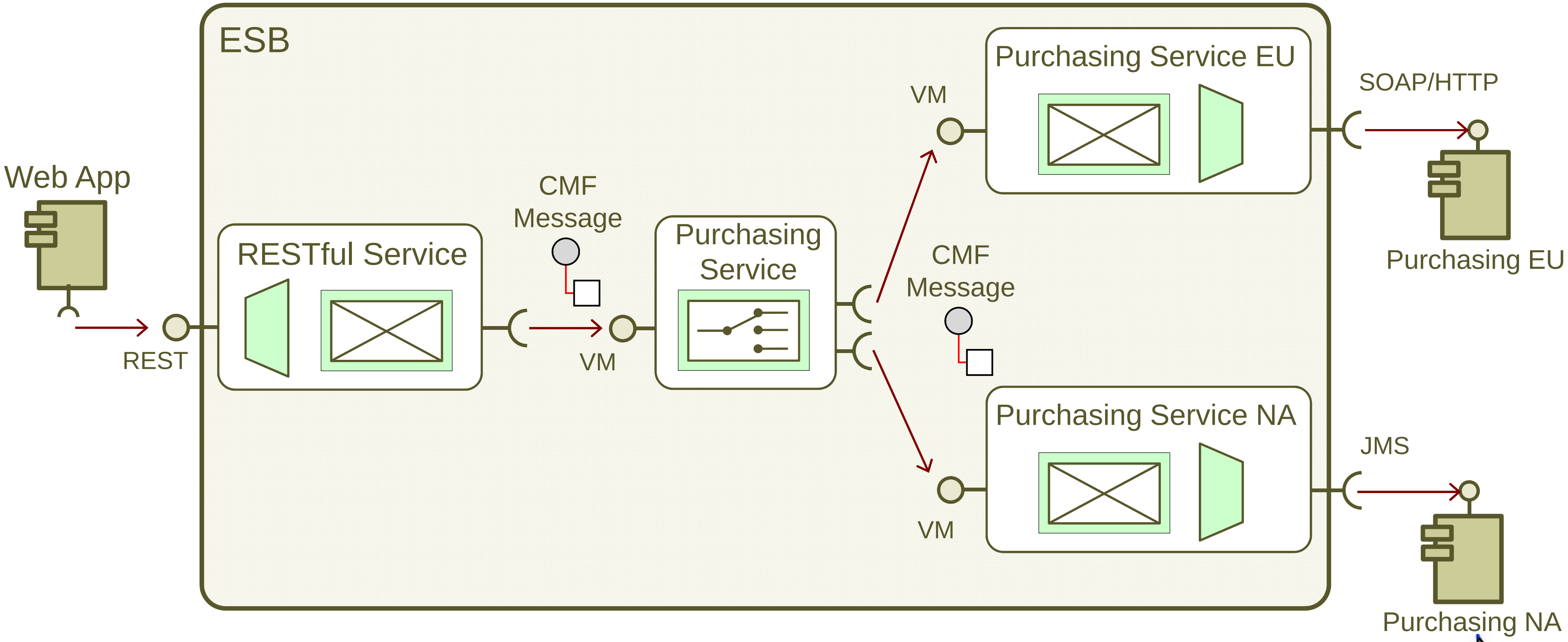
Purchasing EU

Purchasing NA

Purchasing ASIA

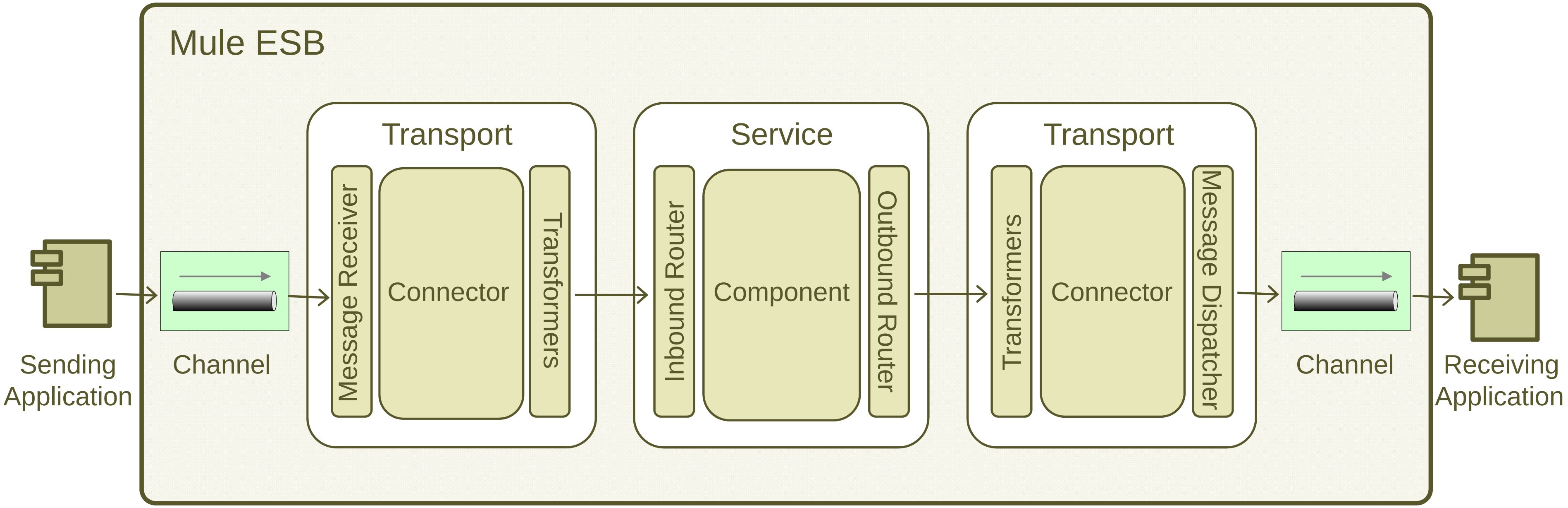
Architecture – ESB Usage Example

- Inside ESB
  - Only use internal messaging, VM - protocol
  - Only use Canonical Message Formats, CMF



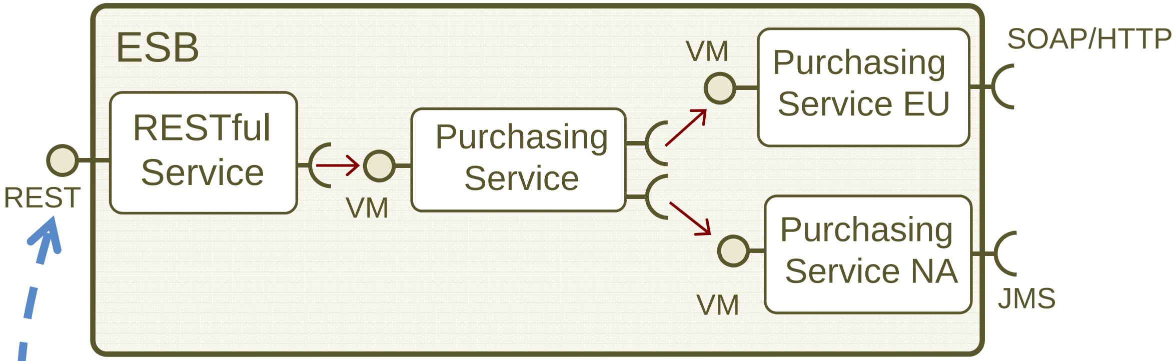
SOA and Open Source

Introducing Mule ESB



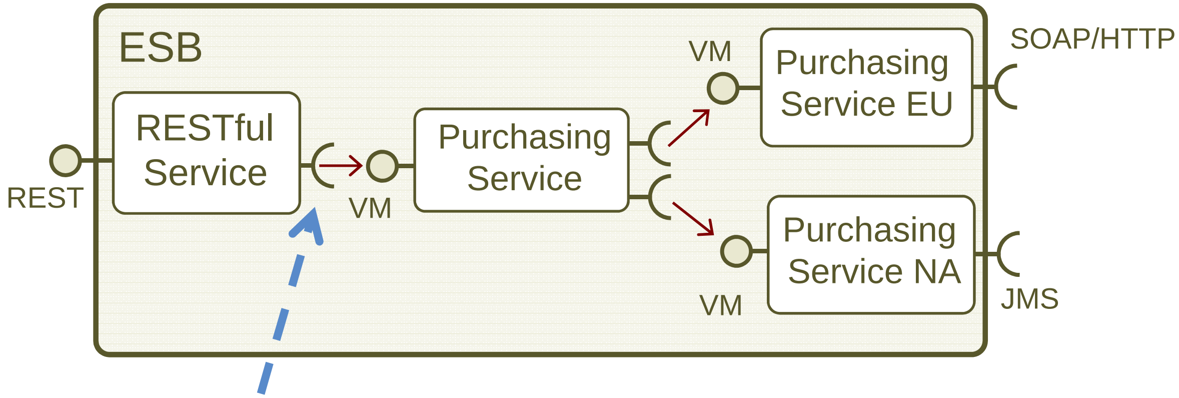
- Configuration based on Spring Framework
- Transports: http://www.mulesource.org/display/MULE2USER/Available+Transports
  - Additional available on http://www.muleforge.org/activeprojects.php
- Routers: http://www.mulesource.org/display/MULE2USER/Using+Message+Routers
- Transformation (using Smooks): http://www.mulesource.org/display/SMOOKS/Home

Enterprise Service Bus Example – Mule ESB



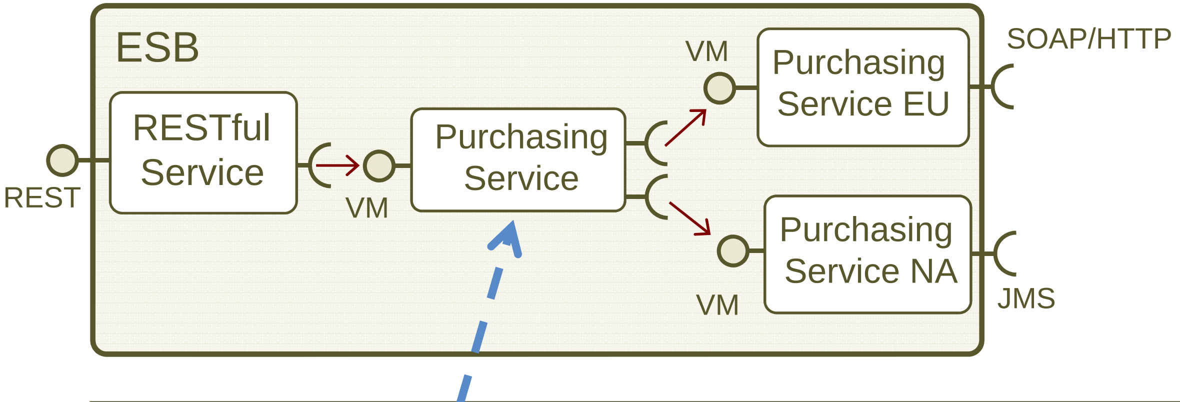
```
<service name="PurchasingService_Client">

 <inbound>

   <inbound-endpoint address="http://localhost:20000" synchronous="true”>
     <acegi:http-security-filter realm="mule-realm"/>
   </inbound-endpoint>

 </inbound>

 ...
```

Enterprise Service Bus Example – Mule ESB



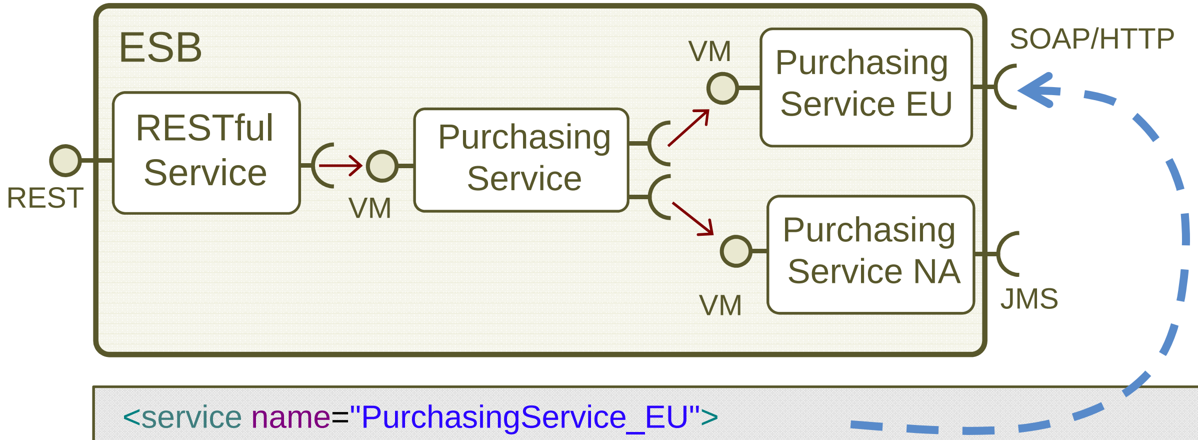
```
  <outbound>

    <filtering-router>
      <outbound-endpoint address="vm://Purchasing" transformer-refs="PPO.RestToCmf"
        responseTransformer-refs="PPO.CmfToRest" synchronous="true" />
     <restlet:uri-template-filter pattern="/ProcessPurchaseOrder" verbs="POST" />
    </filtering-router>

    <filtering-router>
      <outbound-endpoint address="vm://GetPurchaseOrder"/>
      <restlet:uri-template-filter pattern="/purchaseOrder/{poId}" verbs="GET"/>
    </filtering-router>

  </outbound>

</service>
```

Enterprise Service Bus Example – Mule ESB



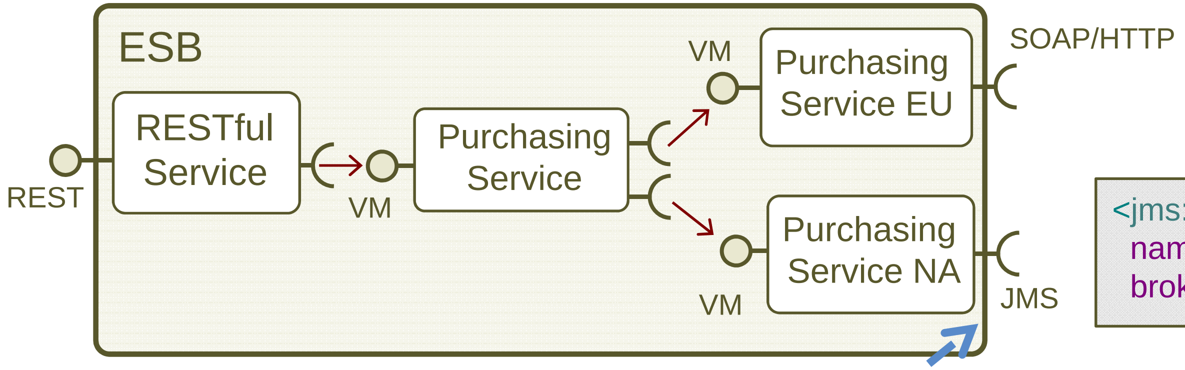
```
<service name="PurchasingService">
  <inbound>
    <inbound-endpoint address="vm://Purchasing" synchronous="true"/>
  </inbound>
  <outbound>
    <filtering-router>
      <outbound-endpoint address="vm://S1_PurchasingService_EU" synchronous="true"/>
      <message-property-filter pattern="ROLE=ROLE_USER_EU"/>
    </filtering-router>

    <filtering-router>
      <outbound-endpoint address="vm://S2_PurchasingService_NA" synchronous="true"/>
      <message-property-filter pattern="ROLE=ROLE_USER_NA"/>
    </filtering-router>
  </outbound>
</service>
```

Enterprise Service Bus Example – Mule ESB



```
<service name="PurchasingService_EU">
  <inbound>
    <inbound-endpoint address=”vm://S1_PurchasingService_EU" synchronous="true"/>
  </inbound>
  <outbound>
    <pass-through-router>
      <cxf:outbound-endpoint
        address=http://localhost:19000/Purchasing_EU
        wsdlLocation="classpath:/schemas/business/purchase/purchase-1.0.wsdl"
        wsdlPort="PurchasingPort"
        operation="ProcessPurchaseOrder"
        clientClass="se.callista.soalab.purchase.wsdl.v1.PurchasingService”
        synchronous="true"
      />
    </pass-through-router>
  </outbound>
</service>
```

Enterprise Service Bus Example – Mule ESB

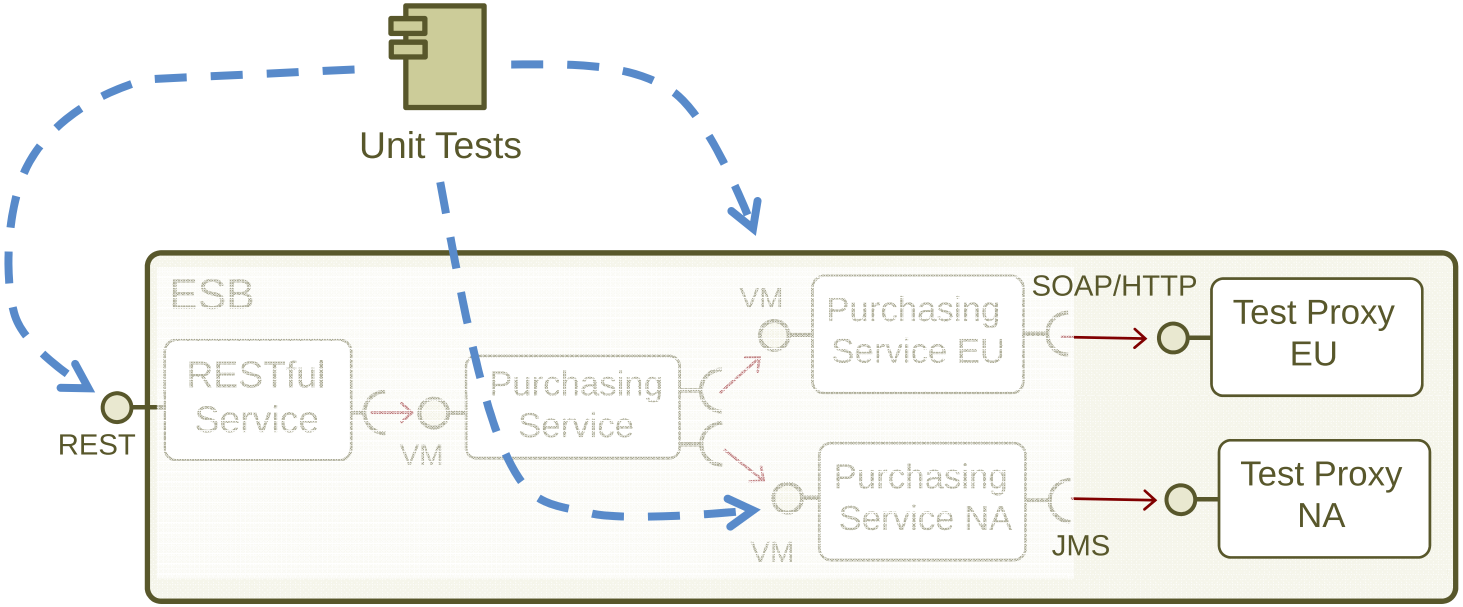


```
<jms:activemq-connector
  name="jmsConnector"
  brokerURL="tcp://localhost:61616”/>
```

```
<service name="PurchasingService_NA">
  ...
  <outbound>
    <pass-through-router>
      <outbound-endpoint address="jms://Soalab.PurchasingService_NA.Request"
        transformer-refs=”..." synchronous="true"/>
      <reply-to address="jms://Soalab.PurchasingService_NA.Response" />
    </pass-through-router>
  </outbound>

  <async-reply timeout="10000">
    <inbound-endpoint address="jms://Soalab.PurchasingService_NA.Response"
      transformer-refs=”..."/>
    <single-async-reply-router/>
  </async-reply>
</service>
```

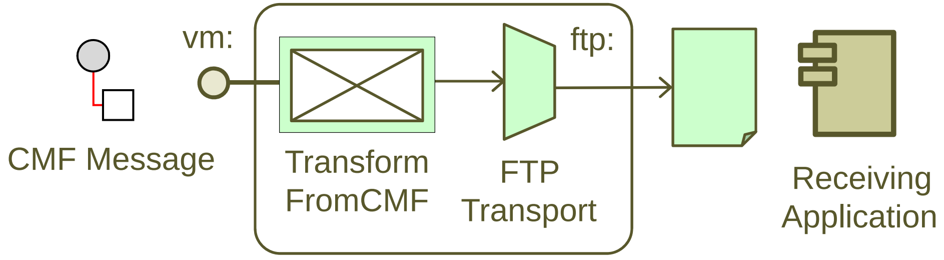
Enterprise Service Bus Example – Mule ESB

- Unit testing services in Mule ESB
  - Test Proxy Services
    - – Separate config-file
  - JUnit Tests
    - – Mule base class initiates Mule ESB per test run



CALLISTA

Mule ESB Connectivity - FTP



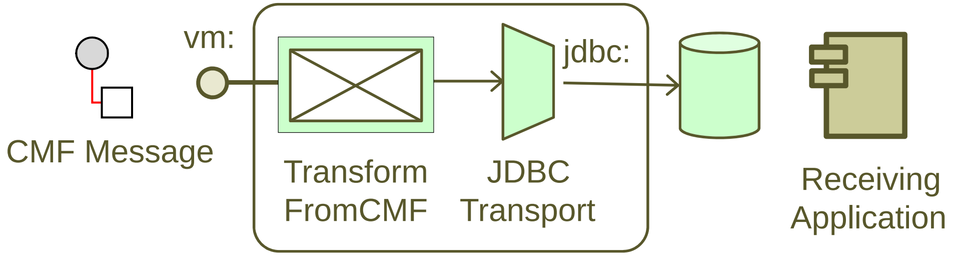
- Declare a FTP Connector

```
<ftp:connector
  name="FtpConnector"
  pollingFrequency="1000"
  binary="true"
  passive="true"
  outputPattern="${ORIGINALNAME}_${DATE}.txt" />
```

- Send file to a FTP Server

```
<outbound-endpoint address="ftp://usr:pwd@server/dir"/>
```

Mule ESB Connectivity - JDBC



- Declare a JDBC Connector

```
<jdbc:connector name="jdbcConnector"
  pollingFrequency="500" dataSource-ref="jdbcDataSource">

  <jdbc:query key="insertImportData"
    value="INSERT INTO IMPORT_TB (ID, VALUE)
           VALUES (${map-payload:ID}, ${map-payload:VALUE})"/>

</jdbc:connector>
```

- Insert records into staging table

```
<jdbc:outbound-endpoint queryKey="insertImportData”
  transformer-refs=”ToMap"/>
```

Mule ESB – Transaction and Error Handling

- Configure transactional endpoints

```
<jms|jdbc|vm:transaction action="ALWAYS_BEGIN"/>
```

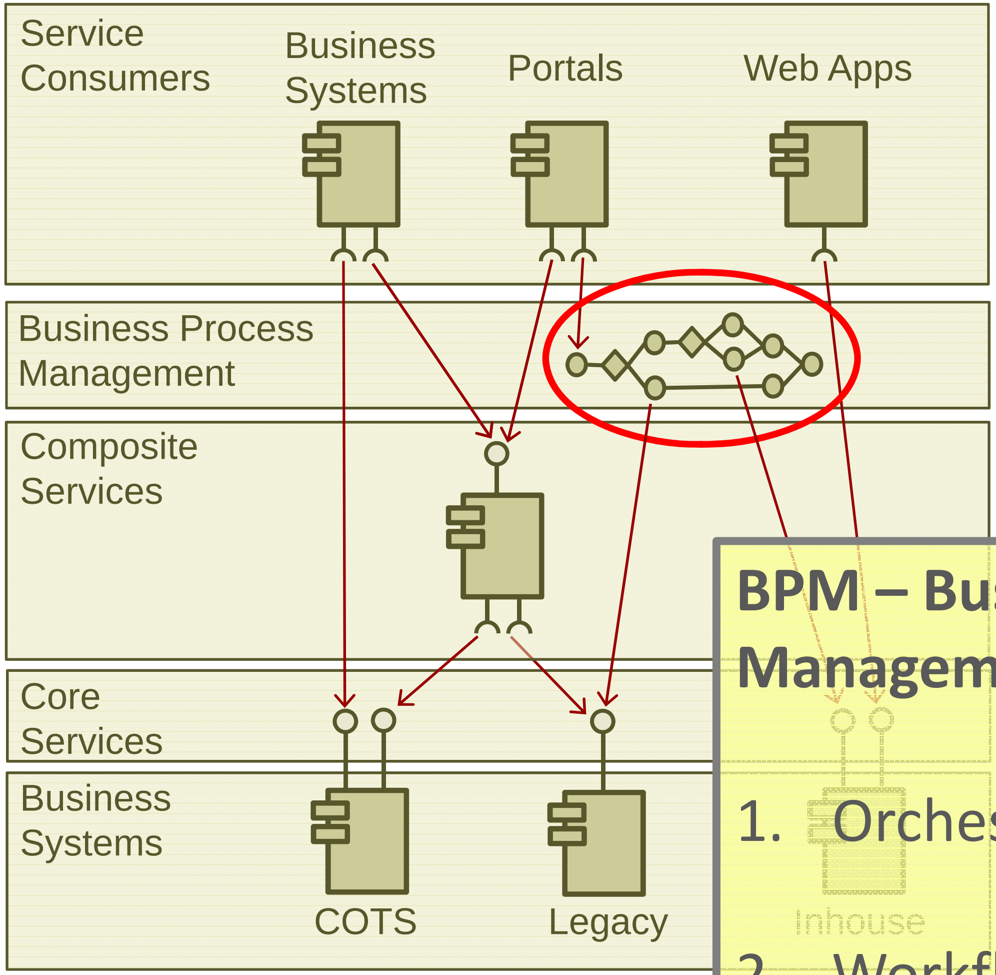
- Global XA transactions

```
<xa-transaction action="ALWAYS_JOIN"/>
```

- Error handling including commit/rollback control

```
<default-service-exception-strategy>

  <vm:outbound-endpoint path="S0_SystemErrorHandler"/>

  <commit-transaction
    exception-pattern="se.callista.soalab.BusinessException"/>

</default-service-exception-strategy>
```

Building a SOA Reference Model…

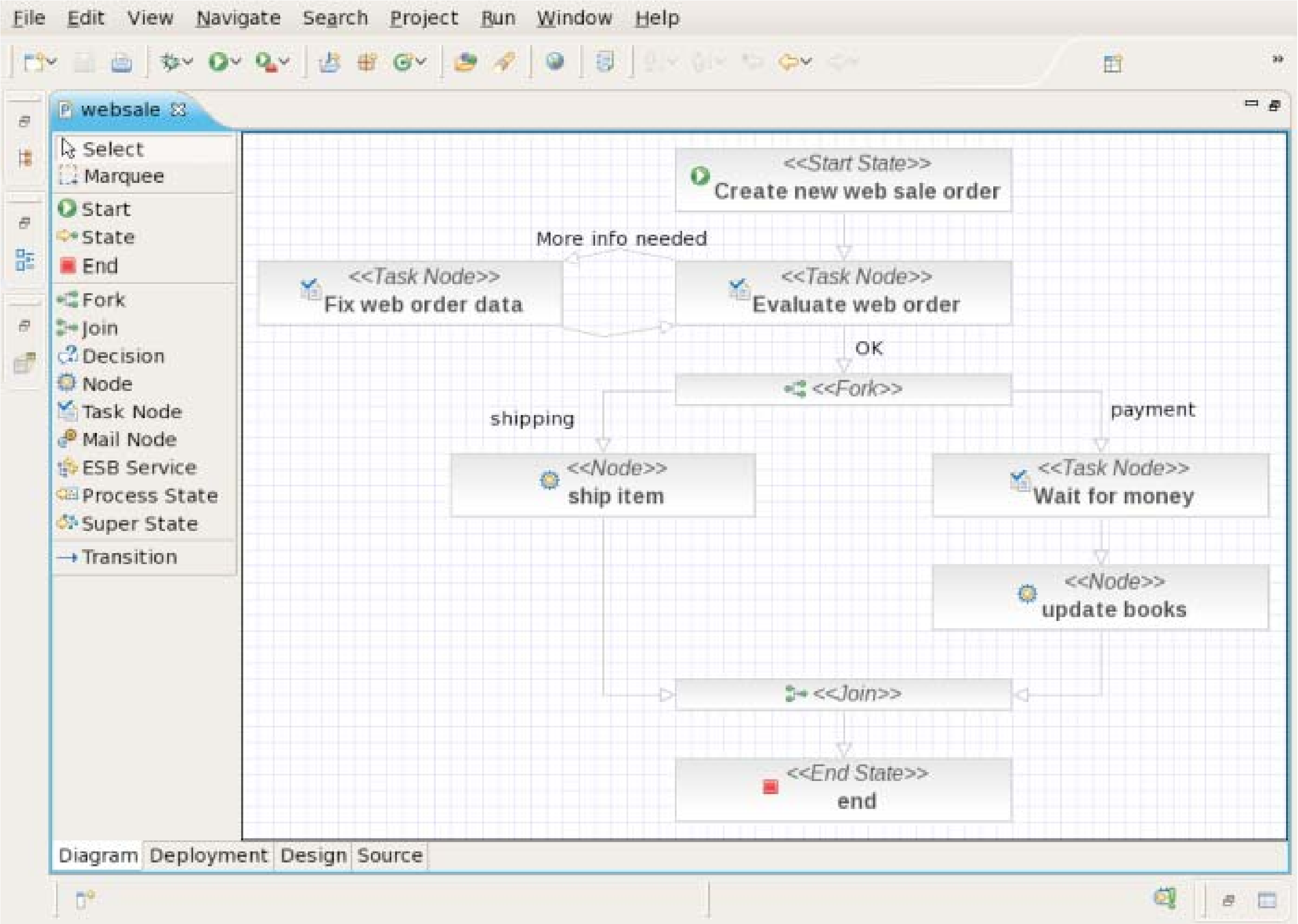


BPM – Business Process Management

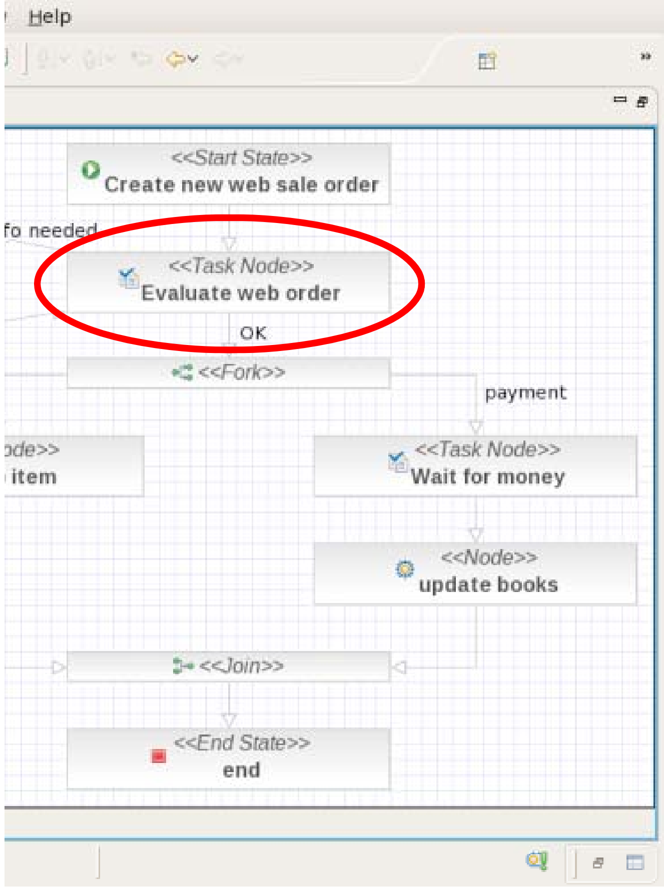
1. Orchestration
2. Workflow

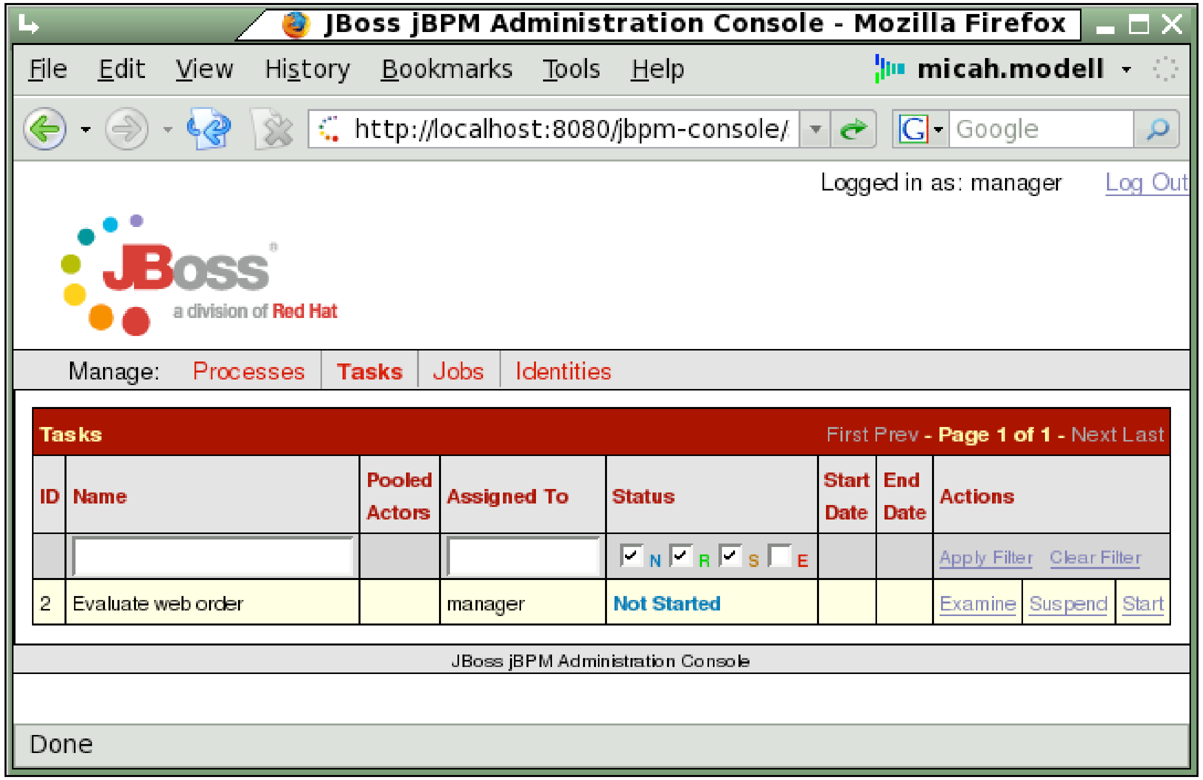
CALLISTA

BPM (JBoss jBPM)



BPM (JBoss jBPM)




JBoss jBPM Administration Console - Mozilla Firefox
File Edit View History Bookmarks Tools Help
micah.modell
http://localhost:8080/jbpm-console/
Google
JBoss a division of Red Hat
Done

Logged in as: manager Log Out

Manage: Processes Tasks Jobs Identities

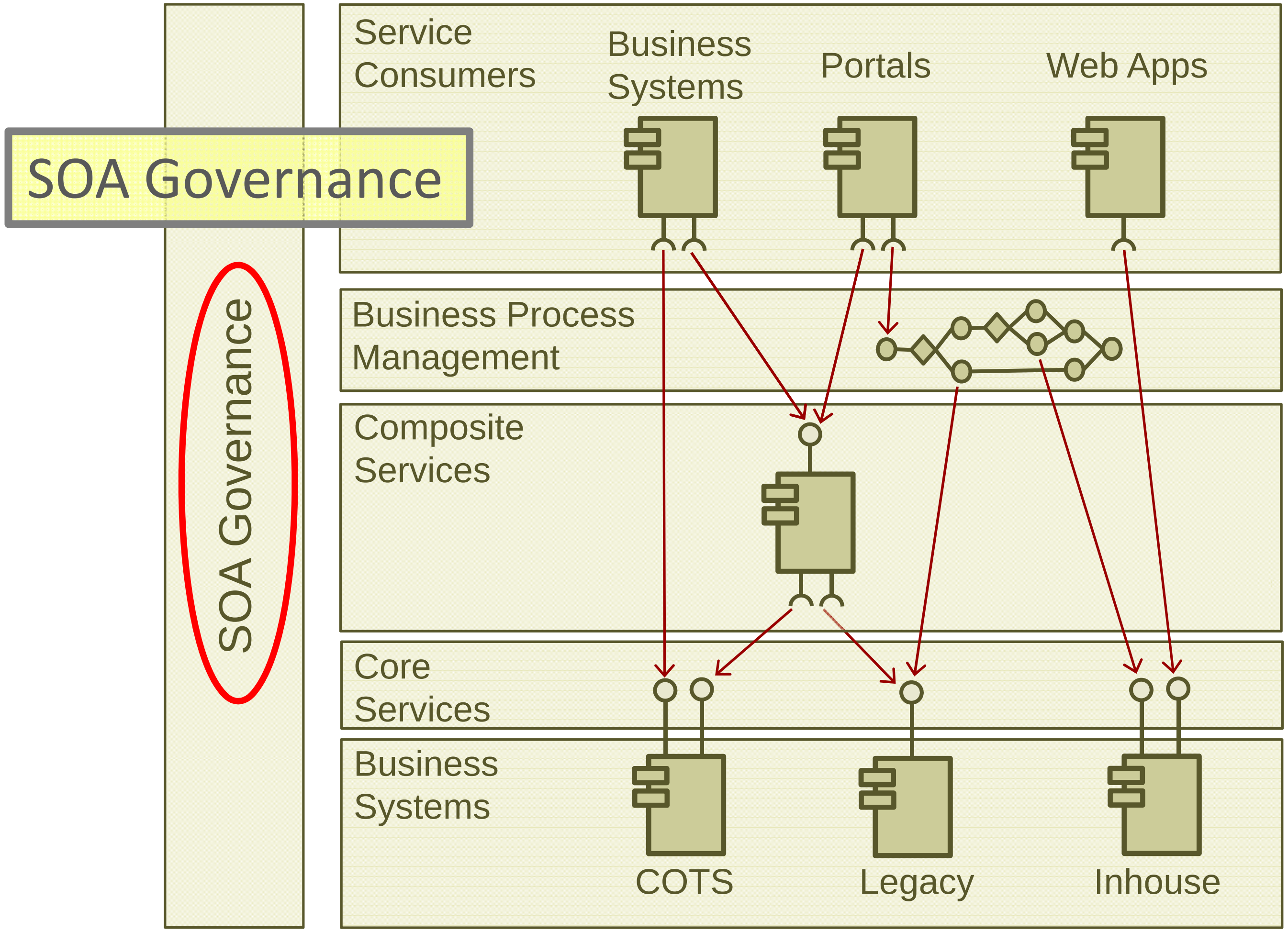
Tasks First Prev - **Page 1 of 1** - Next Last

ID	Name	Pooled Actors	Assigned To	Status	Start Date	End Date	Actions
				☑ N ☑ R ☑ S □ E			Apply Filter Clear Filter
2	Evaluate web order		manager	Not Started			Examine Suspend Start

JBoss jBPM Administration Console

CALLISTA

Building a SOA Reference Model...



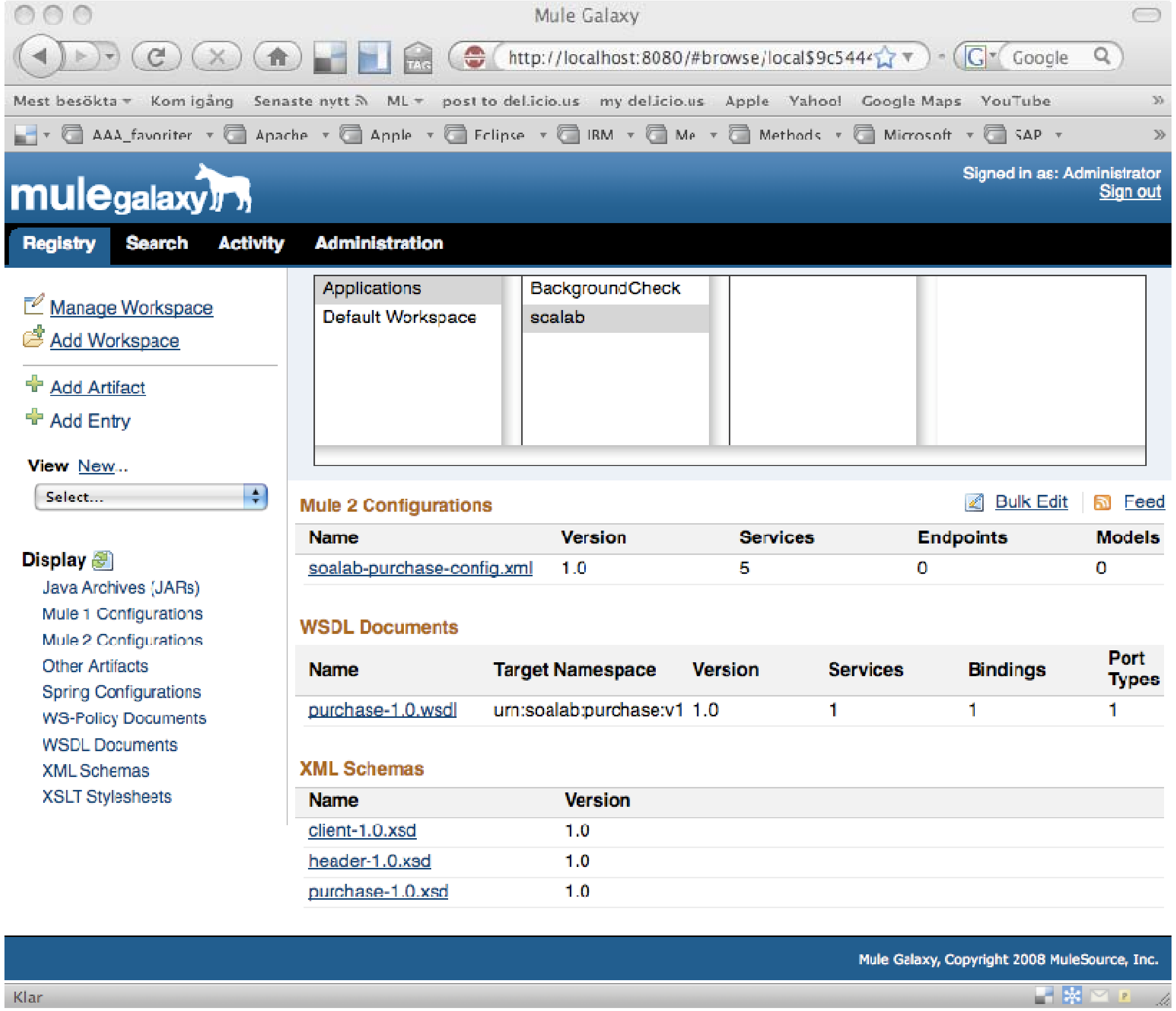
CALLISTA

What is SOA Governance tools?

From **http://wso2.org/library/articles/soa-governance-wso2-registry-v1-1**

- To enable SOA governance, software industry responded with two different categories of tools, **registries** and **repositories**.
- A **registry** is a tool that keeps a list of services with their locations.
- On the other hand a **repository** functions as a tool that keeps information on how the services are used, how they interact, who is using the services and why they are used.
- These tools are considered as key enablers of **SOA Governance**. Usually these two technologies come together as an "Integrated" Registry-Repository, thus avoiding data duplication.

SOA Governance (Mule Galaxy)



Mule 2 Configurations

Name	Version	Services	Endpoints	Models
soalab-purchase-config.xml	1.0	5	0	0

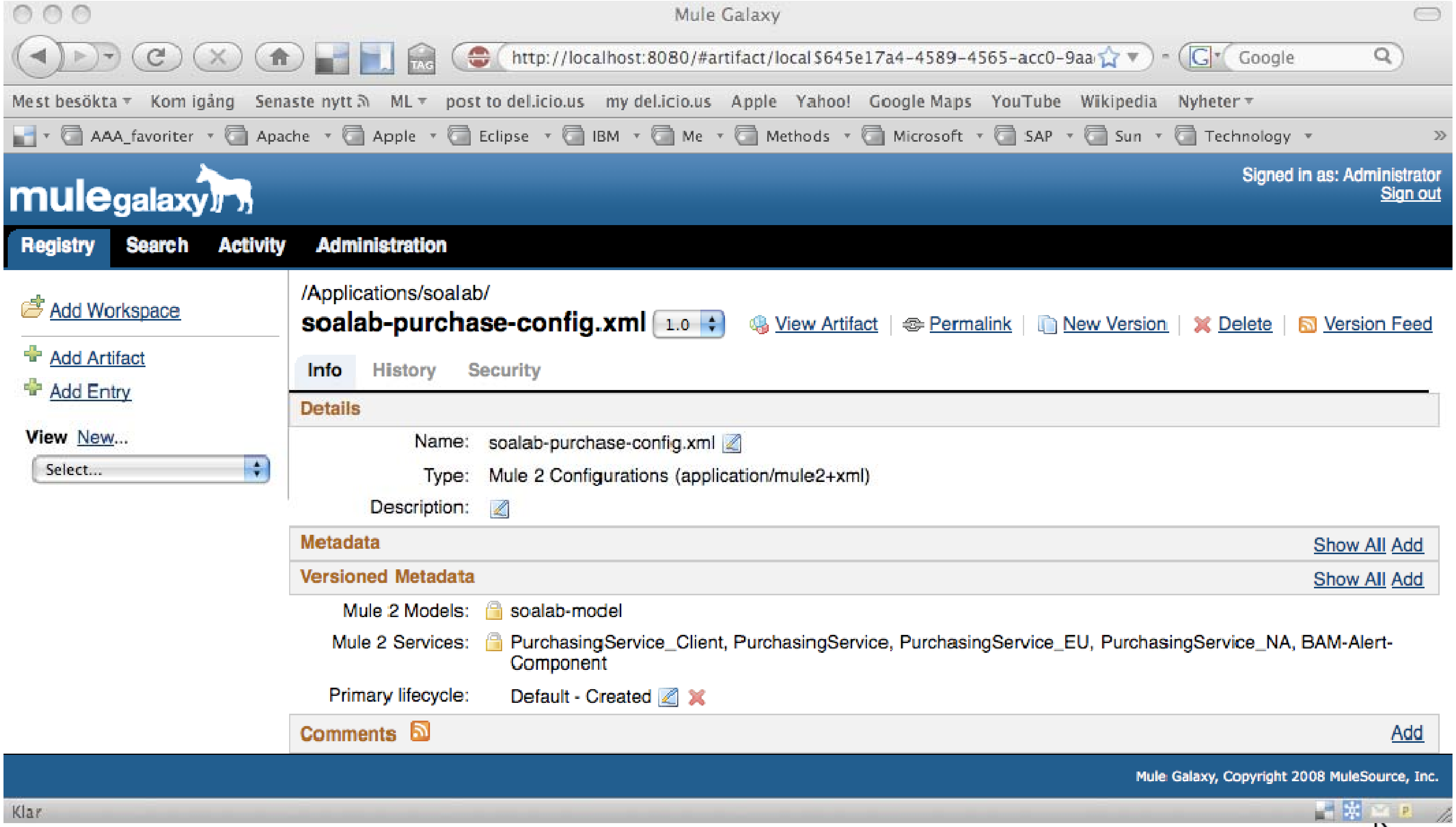
WSDL Documents

Name	Target Namespace	Version	Services	Bindings	Port Types
purchase-1.0.wsdl	urn:soalab:purchase:v1	1.0	1	1	1

XML Schemas

Name	Version
client-1.0.xsd	1.0
header-1.0.xsd	1.0
purchase-1.0.xsd	1.0

SOA Governance (Mule Galaxy)



/Applications/soalab/

soalab-purchase-config.xml 1.0

Details

Name: soalab-purchase-config.xml

Type: Mule 2 Configurations (application/mule2+xml)

Description:

Metadata

Versioned Metadata

Mule 2 Models: soalab-model

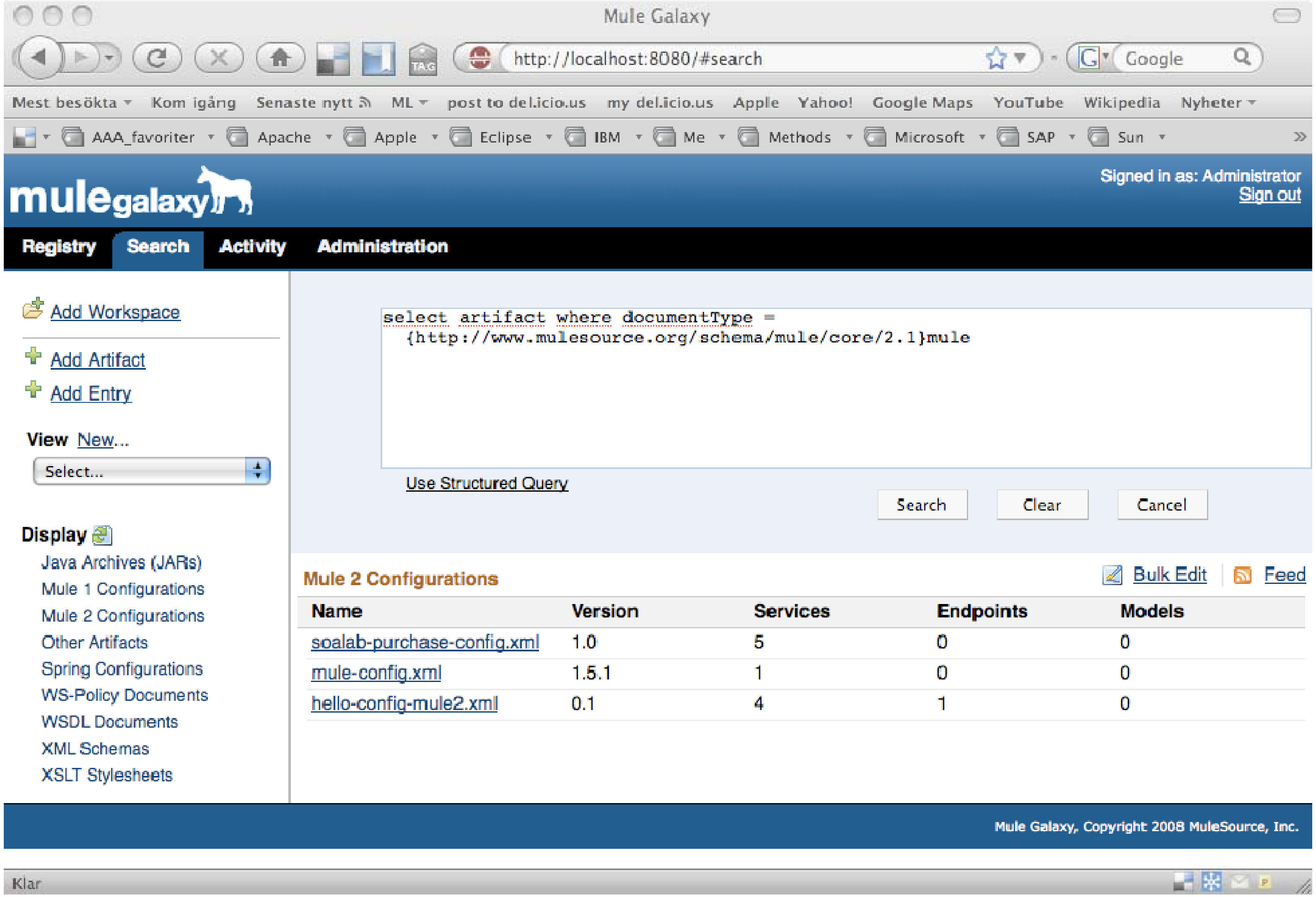
Mule 2 Services: PurchasingService_Client, PurchasingService, PurchasingService_EU, PurchasingService_NA, BAM-Alert-Component

Primary lifecycle: Default - Created

Comments

Mule Galaxy, Copyright 2008 MuleSource, Inc.

SOA Governance (Mule Galaxy)



Signed in as: Administrator
Sign out

```
select artifact where documentType =
  {http://www.mulesource.org/schema/mule/core/2.1}mule
```

Mule 2 Configurations

Name	Version	Services	Endpoints	Models
soalab-purchase-config.xml	1.0	5	0	0
mule-config.xml	1.5.1	1	0	0
hello-config-mule2.xml	0.1	4	1	0

Mule Galaxy, Copyright 2008 MuleSource, Inc.

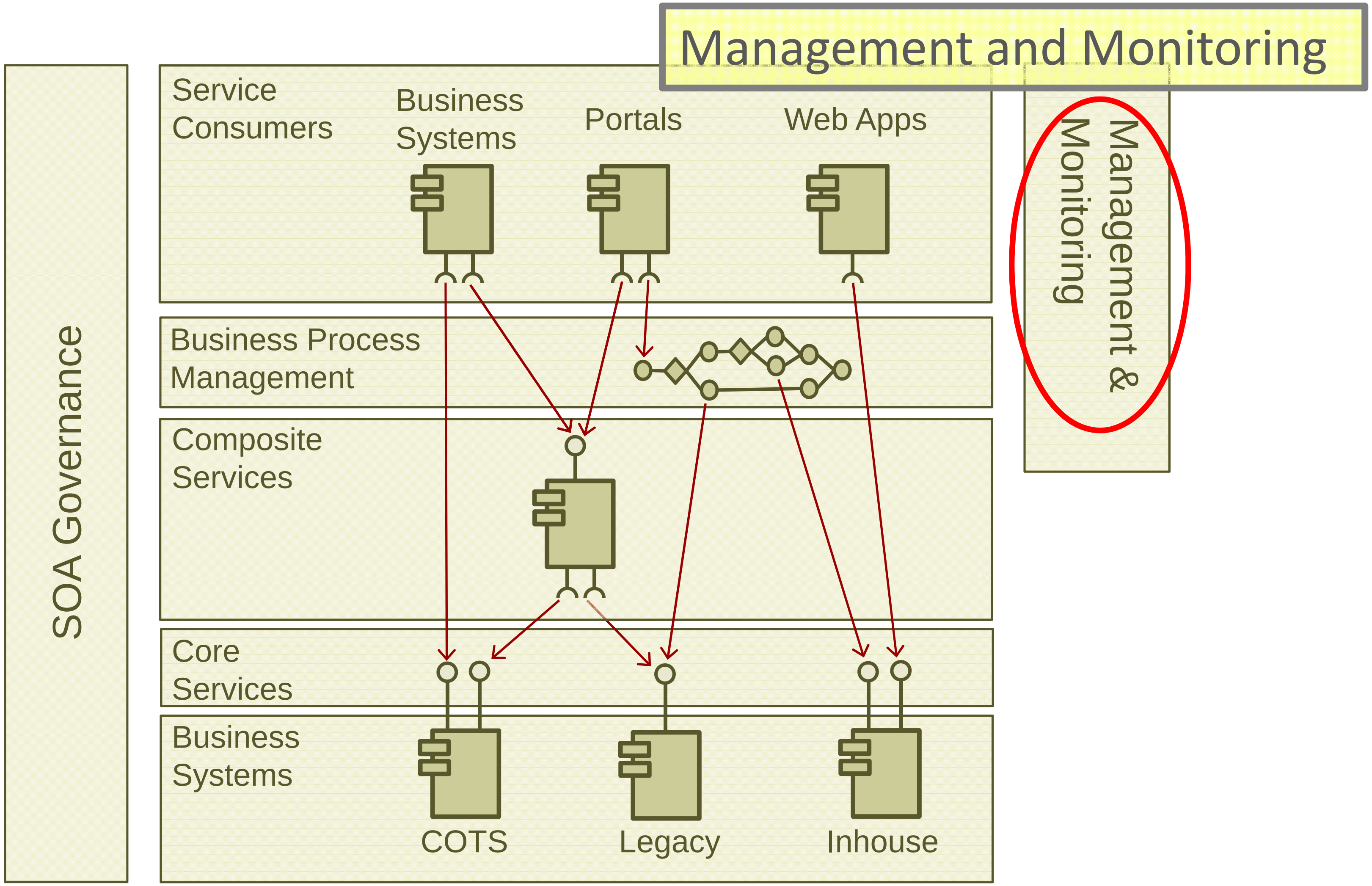
SOA Governance (Mule Galaxy)

- ATOM API – Query example

```
http://localhost:8080/api/registry?
  q=select artifact where documentType =
  {http://www.mulesource.org/schema/mule/core/2.1}mule
```

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <entry>
    <link
      href="/api/registry/Applications/soalab/soalab-purchase-config.xml;atom”/>
    <title type="text">soalab-purchase-config.xml</title>
    ...
</entry>
  <entry>
    <link
      href="/api/registry/Default%20Workspace/hello-config-mule2.xml;atom" />
    <title type="text">hello-config-mule2.xml</title>
    ...
  </entry>
  ...
</feed>
```

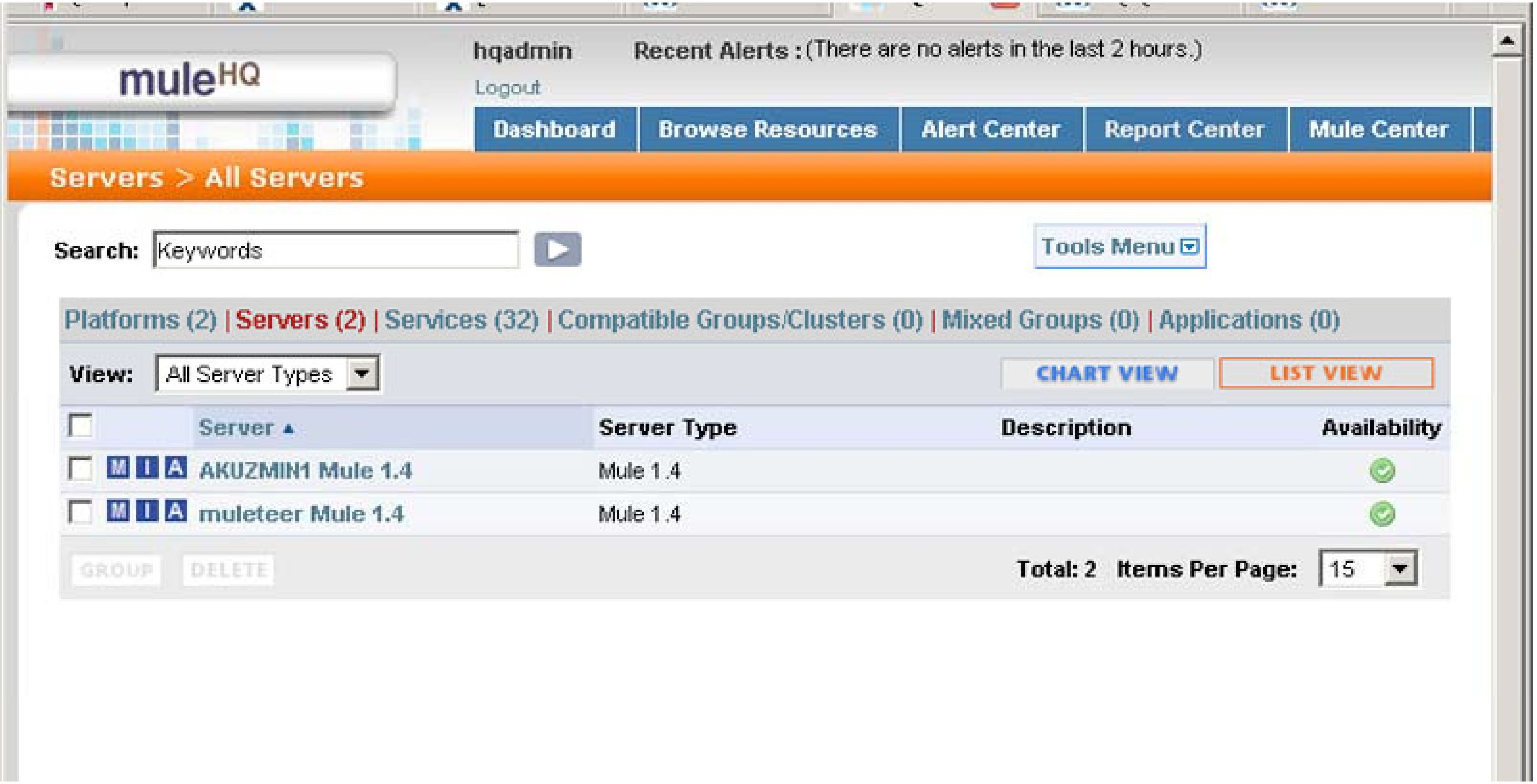
Building a SOA Reference Model...



SOA and Open Source

CALLISTA

Management and Monitoring (Mule HQ)



Recent Alerts :(There are no alerts in the last 2 hours.)

Servers > All Servers

Search: Keywords

Platforms (2) | Servers (2) | Services (32) | Compatible Groups/Clusters (0) | Mixed Groups (0) | Applications (0)

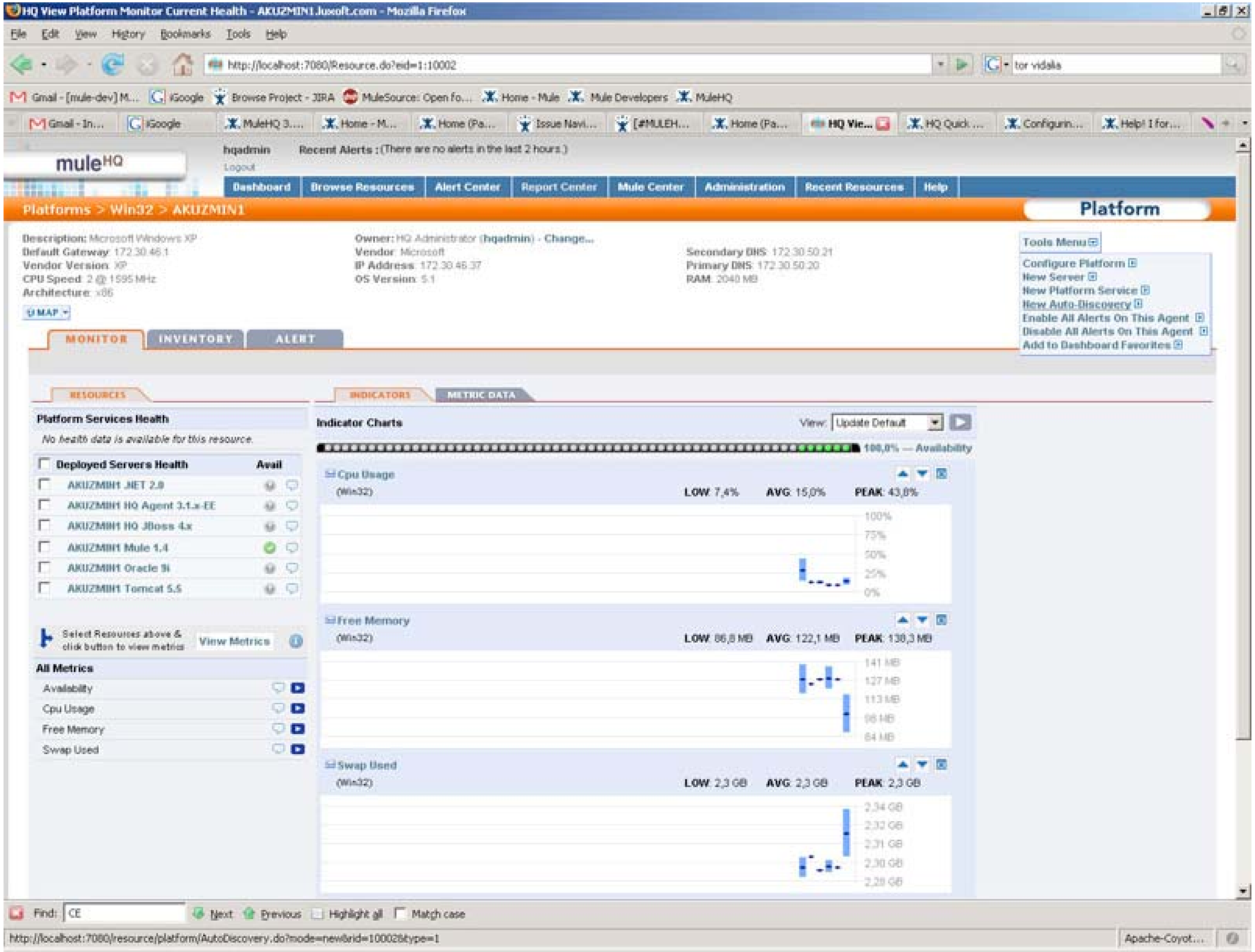
View: All Server Types

□		Server ▲	Server Type	Description	Availability
□	M I A	AKUZMIN1 Mule 1.4	Mule 1.4		✓
□	M I A	muleteer Mule 1.4	Mule 1.4		✓

Total: 2 Items Per Page: 15

SOA and Open Source

Management and Monitoring (Mule HQ)



HQ View Platform Monitor Current Health - AKUZMIN1.Juxoft.com - Mozilla Firefox

http://localhost:7080/Resource.do?eid=1:10002

hqadmin Recent Alerts : (There are no alerts in the last 2 hours.)

Platforms > Win32 > AKUZMIN1

Description: Microsoft Windows XP
Default Gateway: 172.30.46.1
Vendor Version: XP
CPU Speed: 2 @ 1595 MHz
Architecture: x86

Owner: HQ Administrator (hqadmin) - Change...
Vendor: Microsoft
IP Address: 172.30.46.37
OS Version: 5.1

Secondary DNS: 172.30.50.21
Primary DNS: 172.30.50.20
RAM: 2040 MB

Platform Services Health

No health data is available for this resource.

□ Deployed Servers Health	Avail
□ AKUZMIN1 .NET 2.0	
□ AKUZMIN1 HQ Agent 3.1.x-EE	
□ AKUZMIN1 HQ JBoss 4.x	
□ AKUZMIN1 Mule 1.4	✓
□ AKUZMIN1 Oracle 9i	
□ AKUZMIN1 Tomcat 5.5	

Select Resources above & click button to view metrics

All Metrics

- Availability
- Cpu Usage
- Free Memory
- Swap Used

Indicator Charts

View: Update Default

100,0% — Availability

Cpu Usage (Win32) LOW: 7,4% AVG: 15,0% PEAK: 43,0%

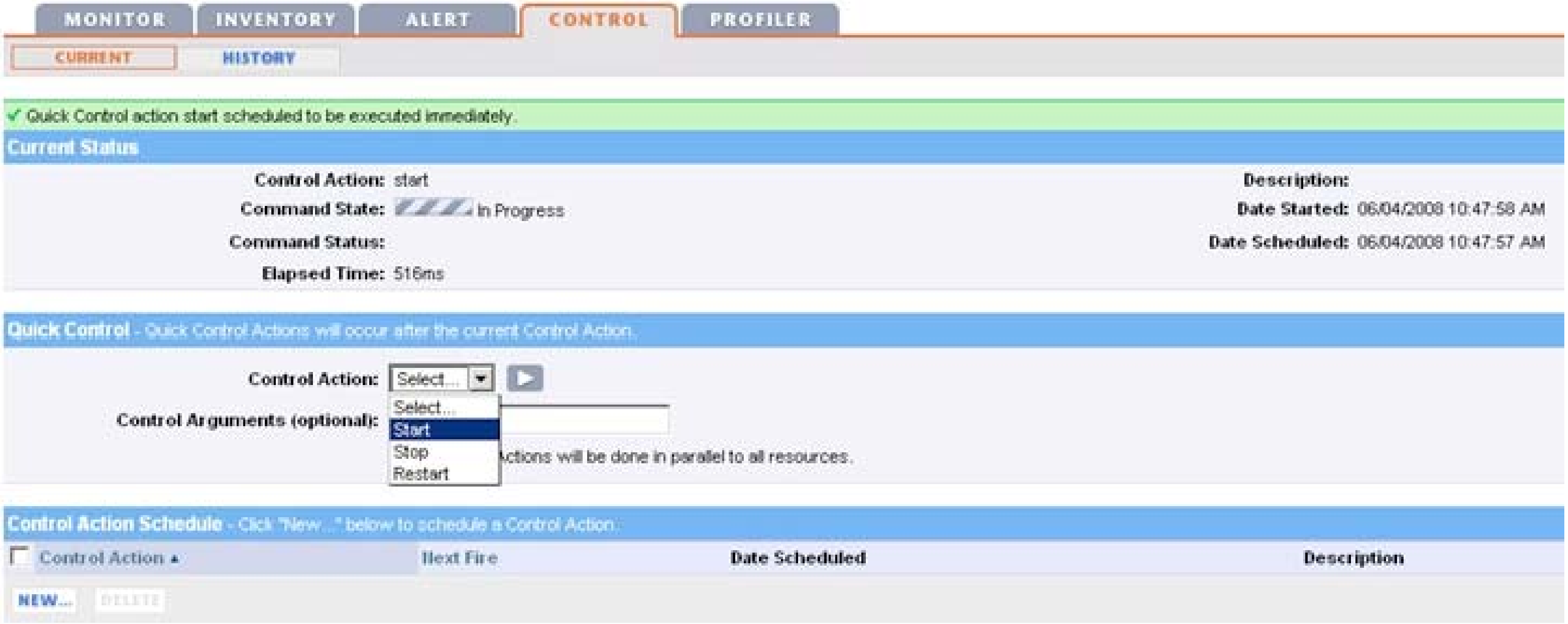
Free Memory (Win32) LOW: 86,8 MB AVG: 122,1 MB PEAK: 138,3 MB

Swap Used (Win32) LOW: 2,3 GB AVG: 2,3 GB PEAK: 2,3 GB

Find: CE

http://localhost:7080/resource/platform/AutoDiscovery.do?mode=new&rid=10002&type=1

Management and Monitoring (Mule HQ)



✓ Quick Control action start scheduled to be executed immediately.

Current Status

Control Action:	start	**Description:**	
Command State:	In Progress	**Date Started:**	06/04/2008 10:47:58 AM
Command Status:		**Date Scheduled:**	06/04/2008 10:47:57 AM
Elapsed Time:	516ms		

Quick Control - Quick Control Actions will occur after the current Control Action.

Control Action: Select...

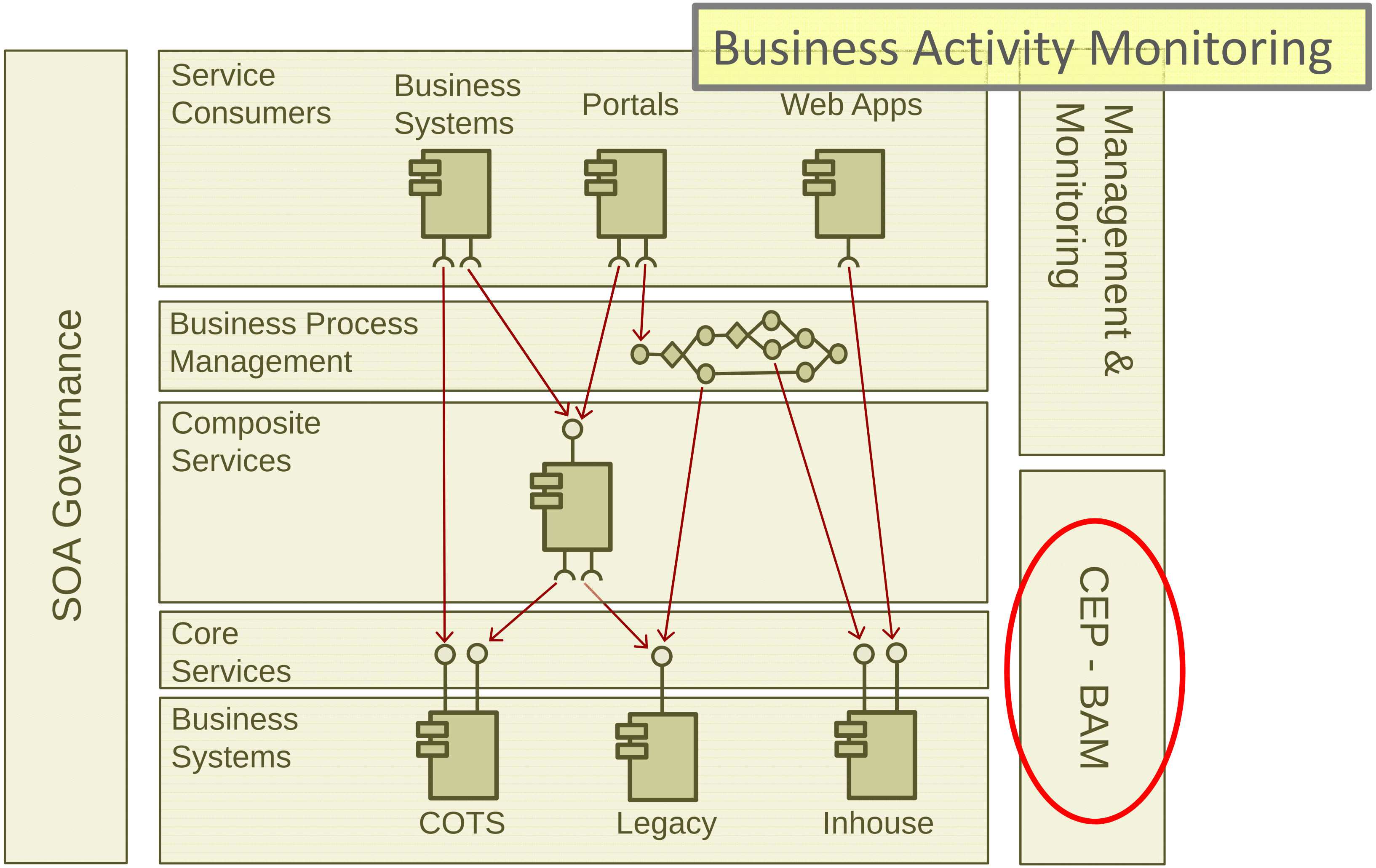
Control Arguments (optional):

…ctions will be done in parallel to all resources.

Control Action Schedule - Click "New..." below to schedule a Control Action.

□ Control Action ▲	Next Fire	Date Scheduled	Description

Building a SOA Reference Model...



SOA and Open Source
Copyright 2009, Callista Enterprise AB

What is BAM & CEP?

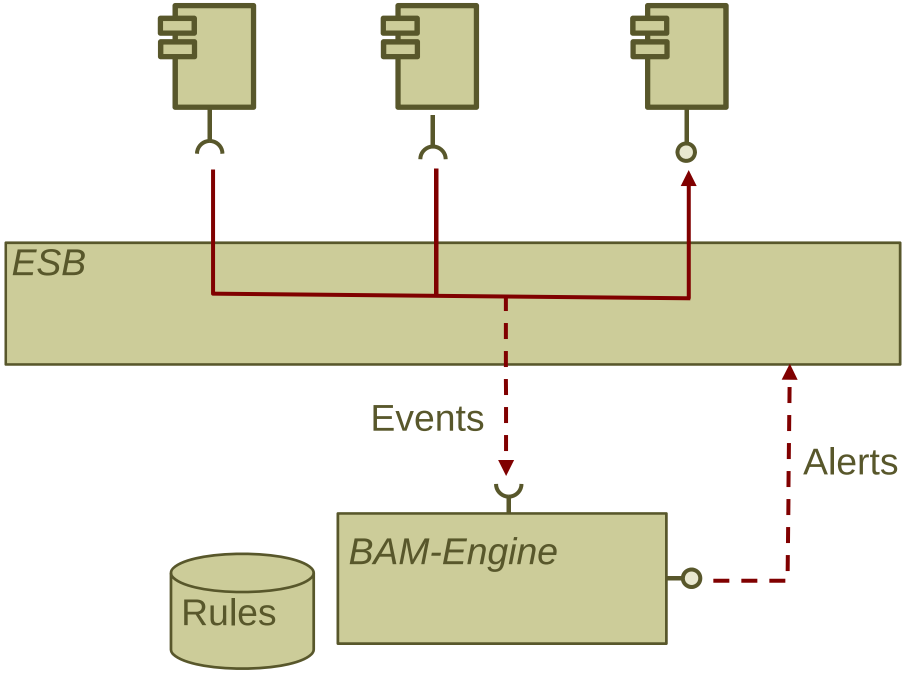
- **Business Activity Monitoring** (BAM) and

  Complex Event Processing (CEP)

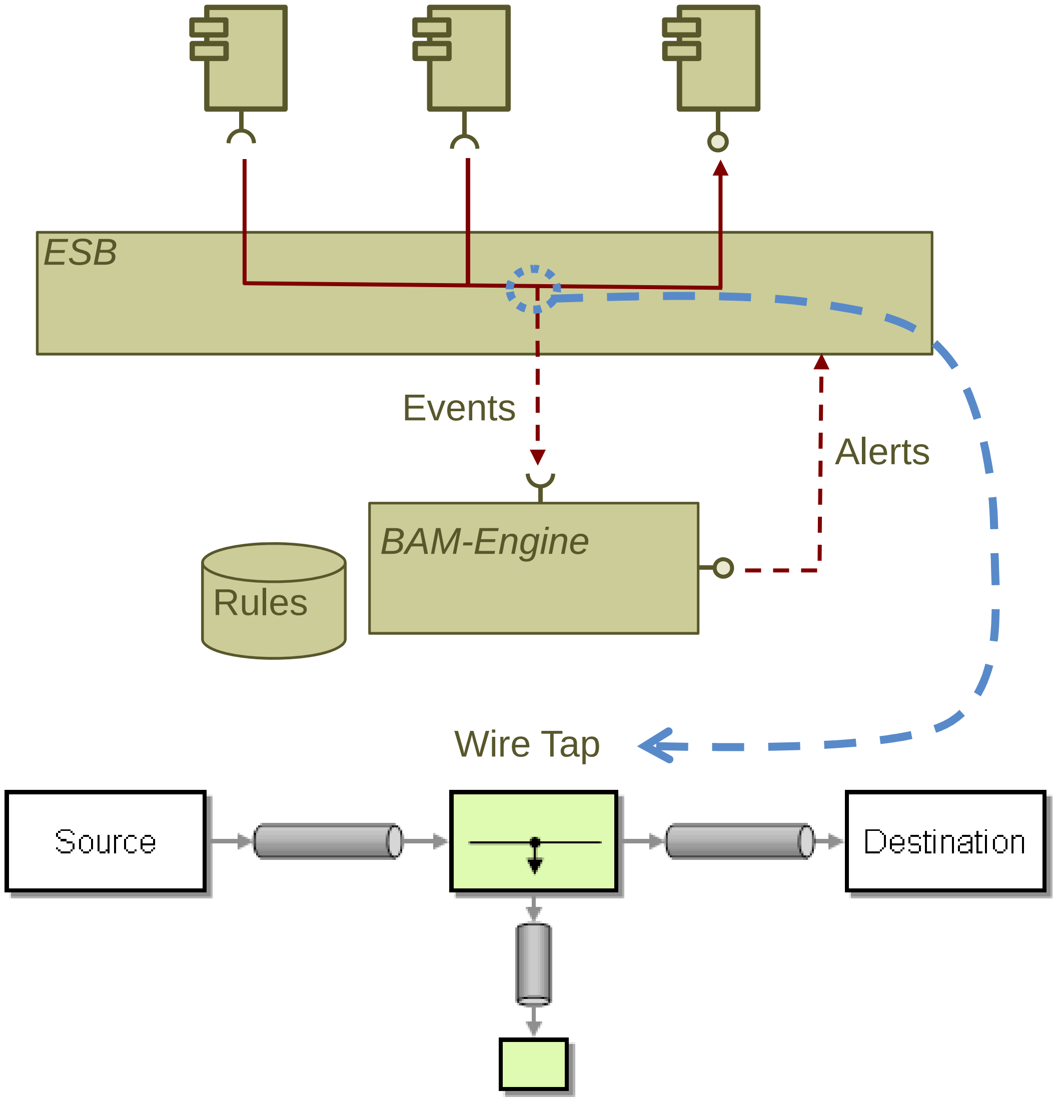
  can be defined as (from **`www.eventstreamprocessing.com`**):

  – *"Software technology that enables applications to monitor multiple streams of event data, analyze them in terms of **Key Performance Indicators** (KPI) that are expressed in event rules, and act upon opportunities and threats in real time, potentially by creating derived events, or forwarding raw events.".*

Business Activity Monitoring (Esper)

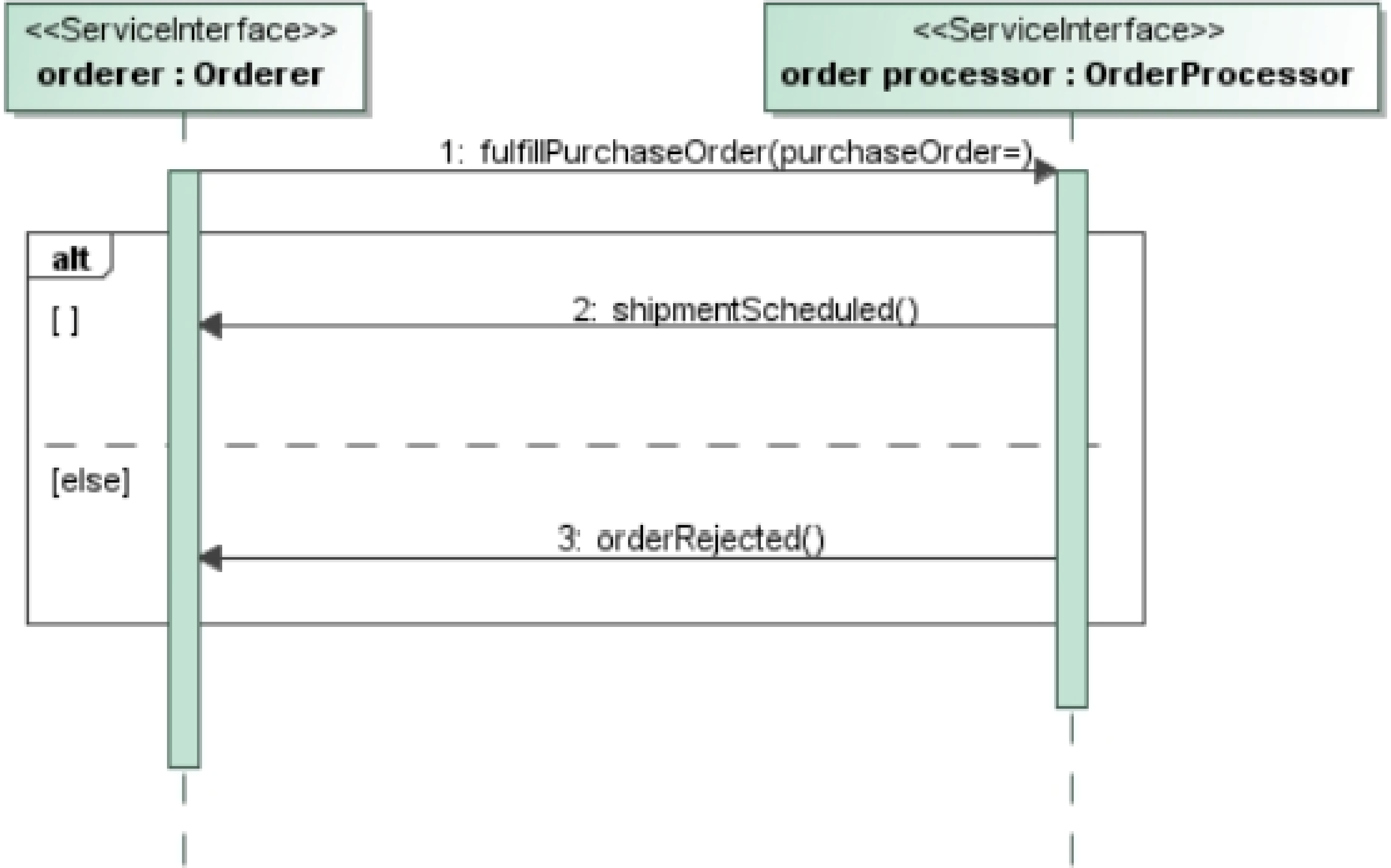


Business Activity Monitoring (Esper)

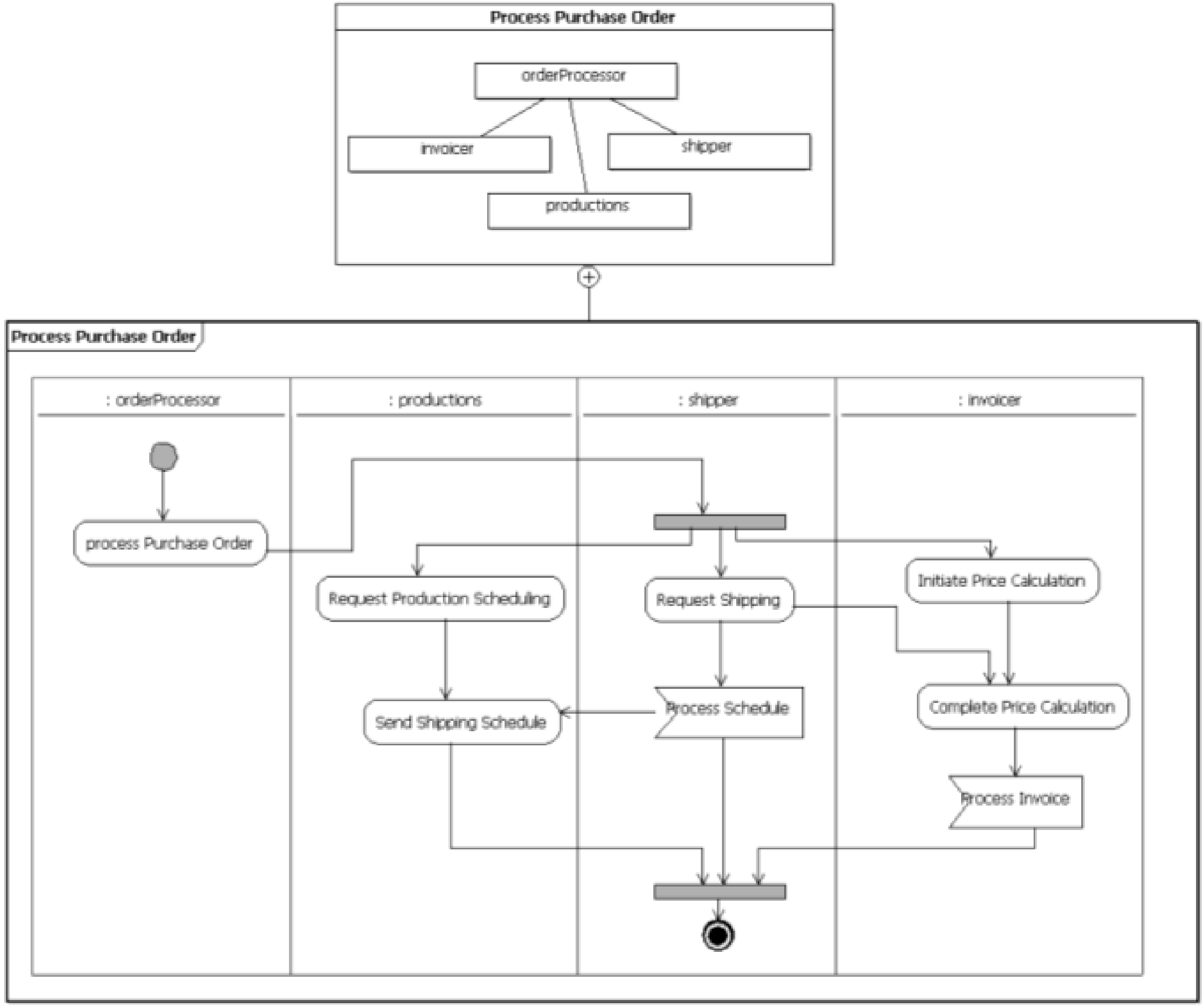


SOA and Open Source

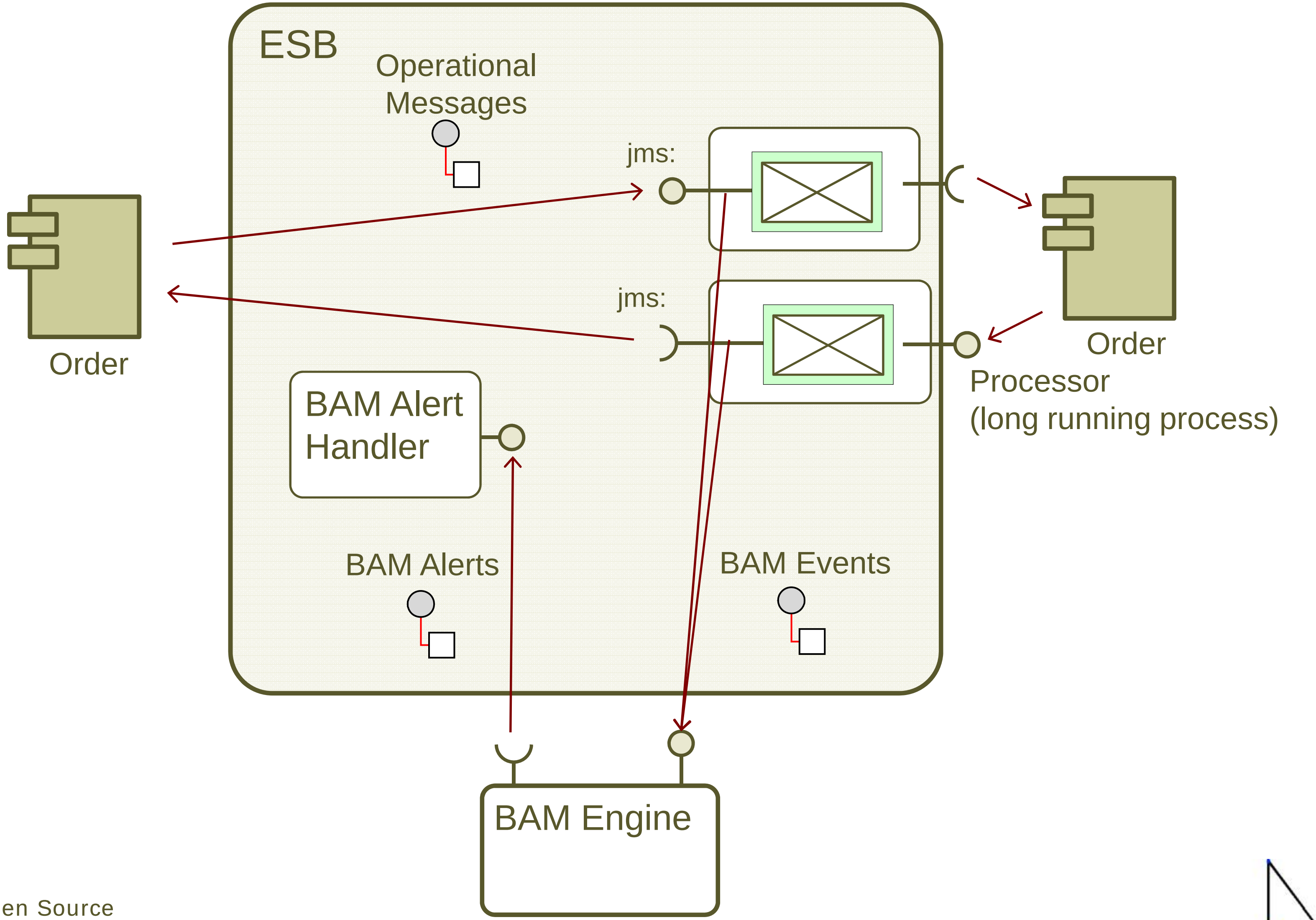
Business Activity Monitoring (Esper)



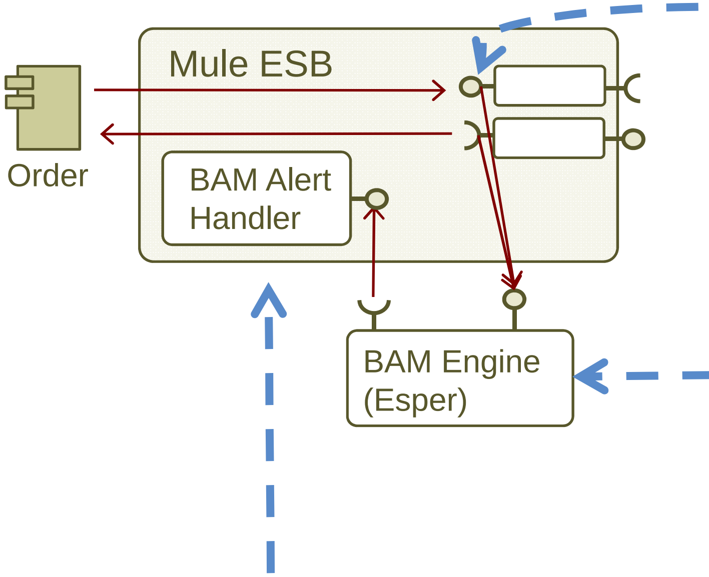
Business Activity Monitoring (Esper)



Business Activity Monitoring (Esper)



Business Activity Monitoring (Esper)

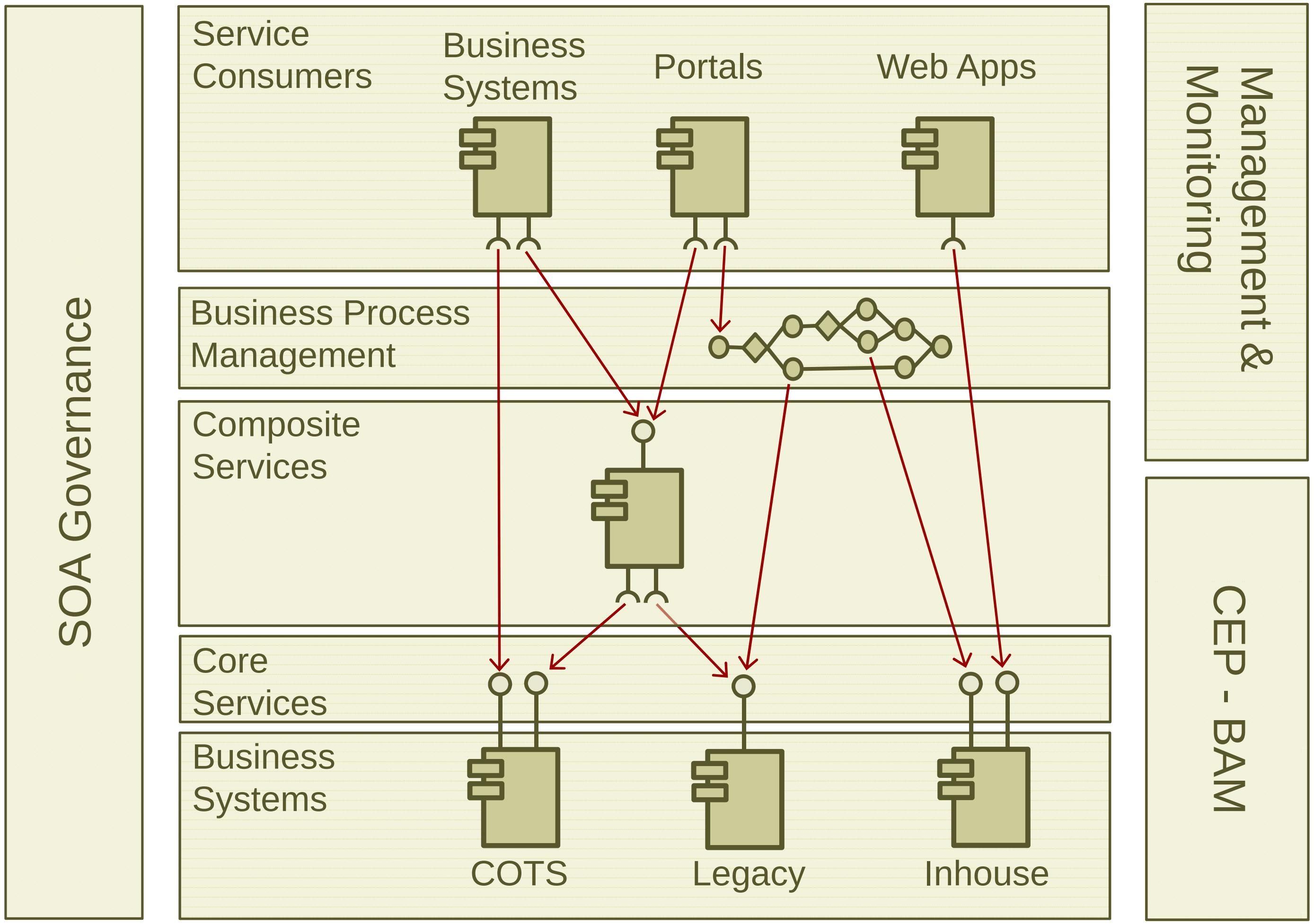


```
<service name="OrderProcessor_TestProxy">
  <inbound>
    <jms:inbound-endpoint queue="Soalab.OrderProcessor”/>
    <wire-tap-router>
      <jms:outbound-endpoint queue="Soalab.BAM"/>
    </wire-tap-router>
</inbound>
```

```
select po.id as orderno from pattern
  [every po=Created ->
    (timer:interval(3 sec) and
    not Completed(id = po.id))]
```

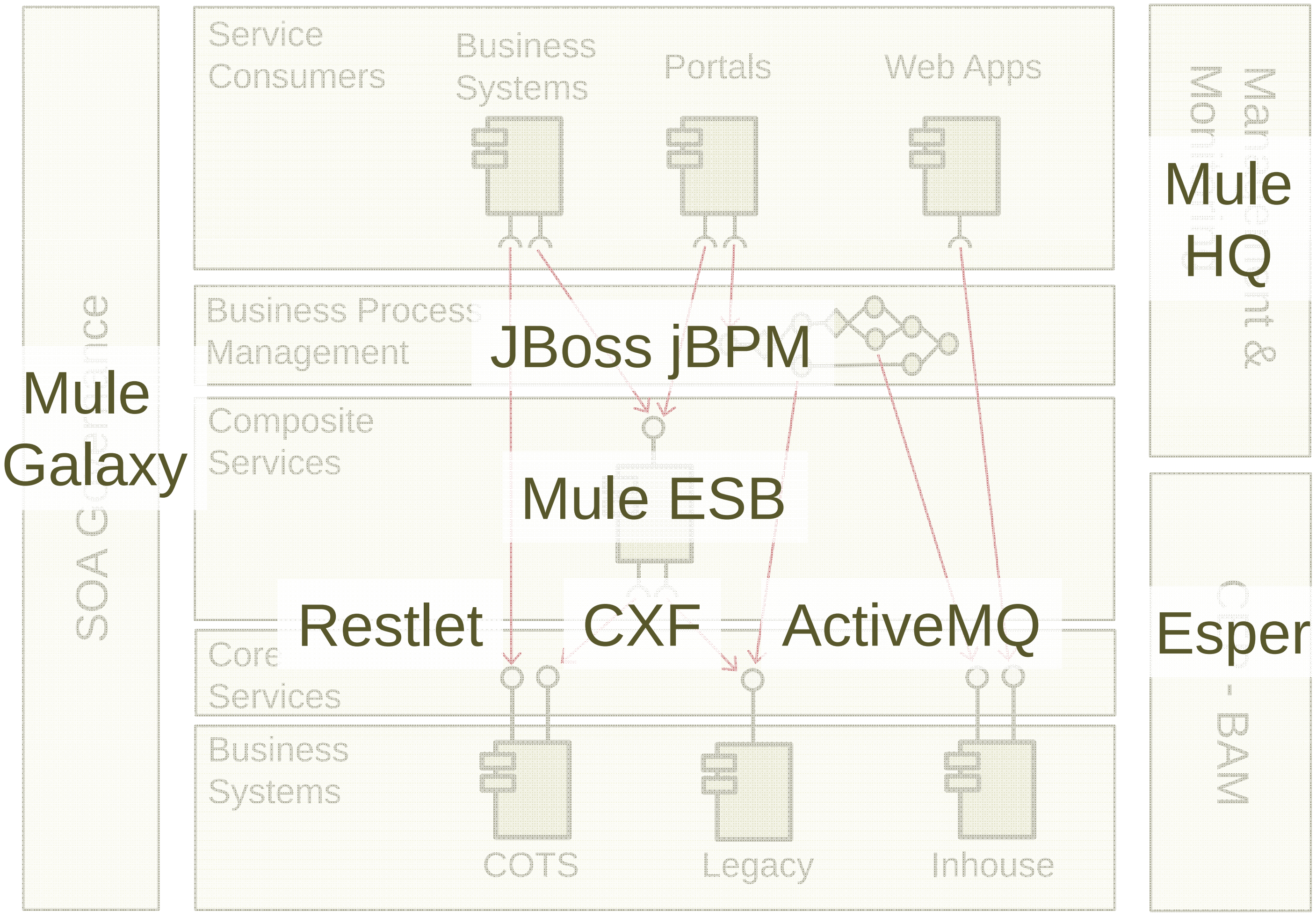
```
<service name="BAM-Alert-Component">
  <inbound>
    <jms:inbound-endpoint queue="Soalab.BAM.Alert"/>
  </inbound>

  <log-component/>
</service>
```

SOA Reference Model, the complete picture



CALLISTA

A sample Open Source SOA Sandbox



Service Consumers

Business Systems

Portals

Web Apps

Business Process Management

JBoss jBPM

Composite Services

Mule ESB

Core Services

Restlet

CXF

ActiveMQ

Business Systems

COTS

Legacy

Inhouse

SOA G... ance

Mule Galaxy

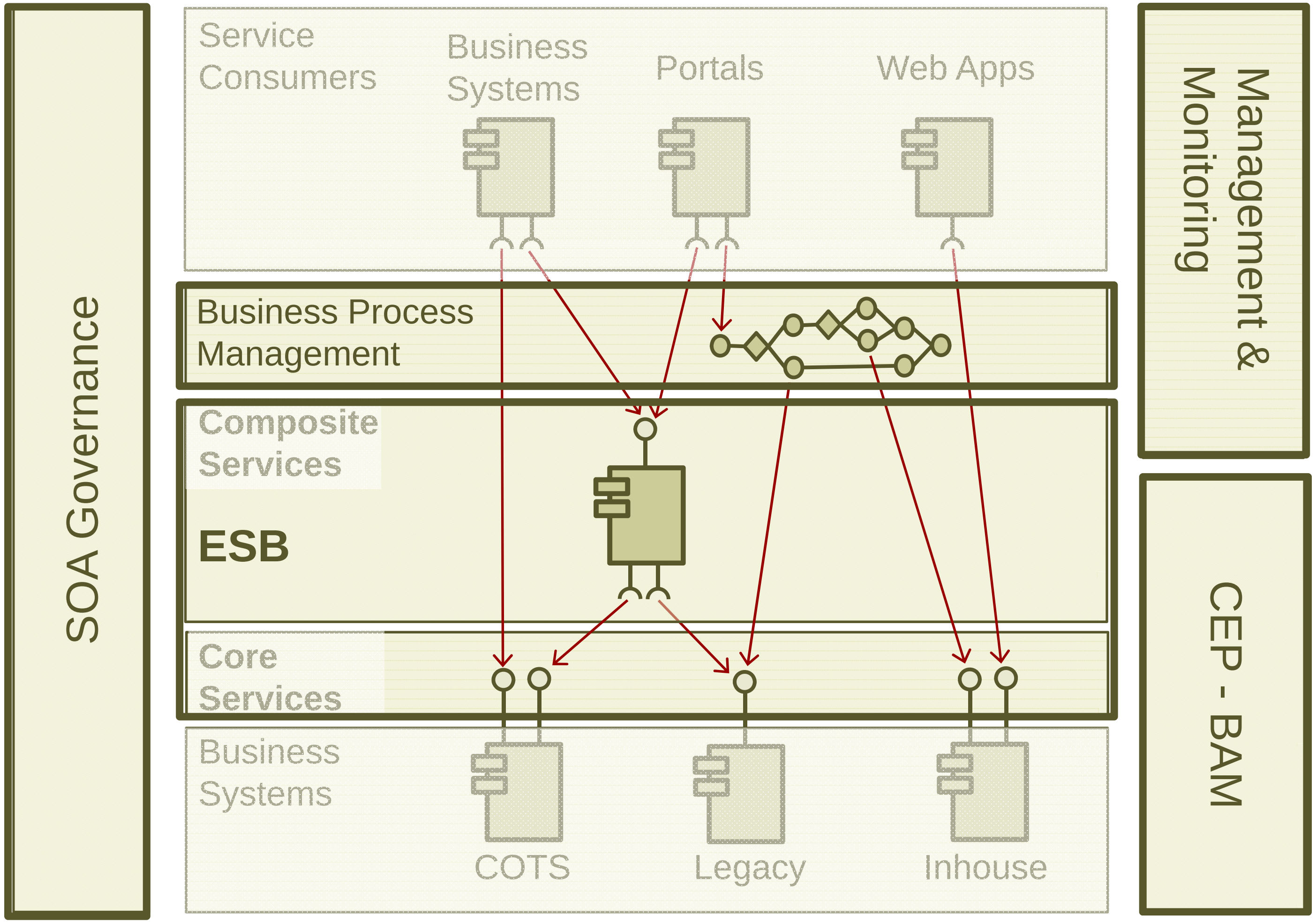
Management & Monitoring

Mule HQ

C... - BAM

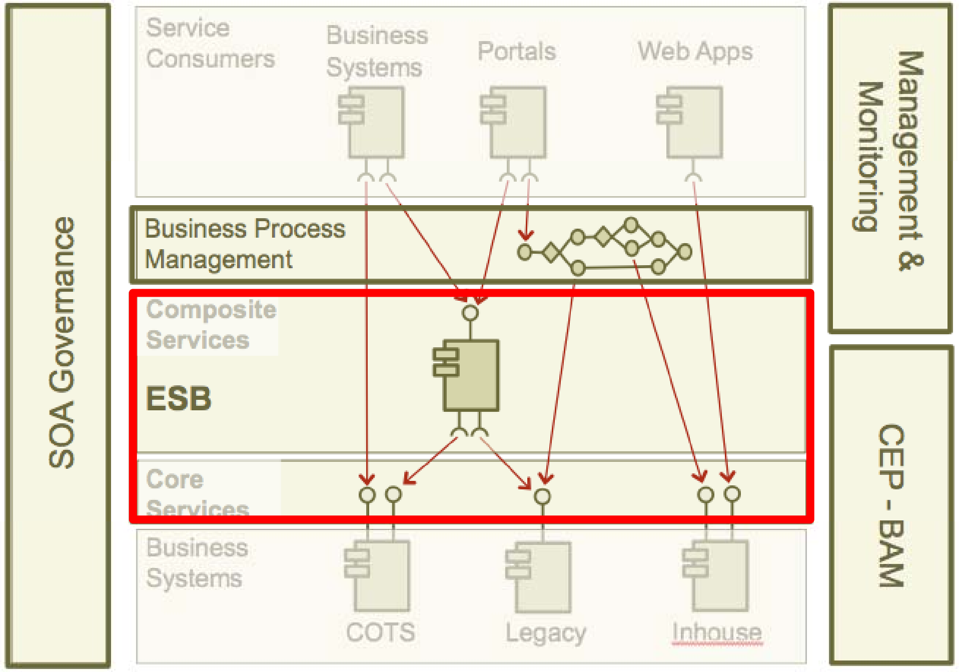
Esper

SOA Product Areas



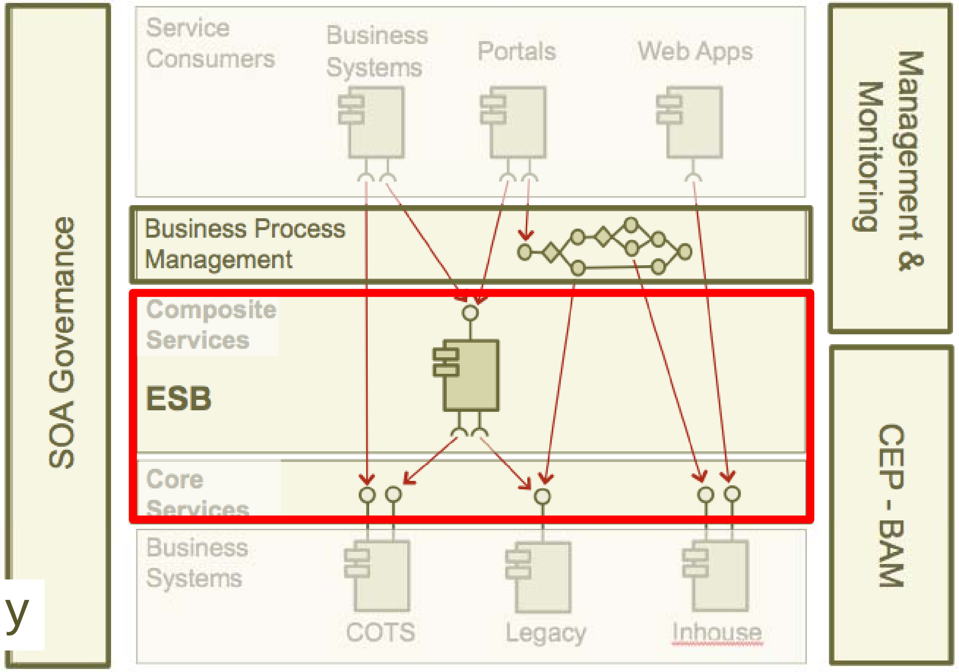
Open Source SOA

- ESB
  - MuleSource Mule ESB
  - Apache ServiceMix
  - Sun OpenESB
  - *”ESB buidling blocks”*
    - Apache Synapse (WS + XML focus)
    - Apache Camel (EIP library)



CALLISTA

Open Source SOA

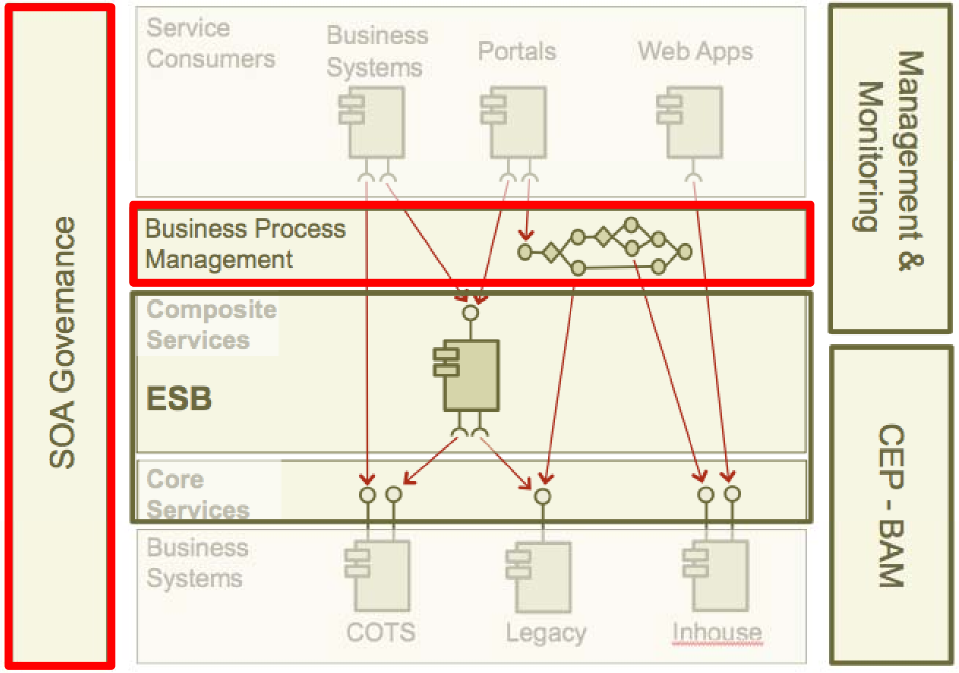


- ESB Connectivity - Web Services
  - Apache CXF
  - Apache Axis
  - Sun Metro (WS-IT)
    - MS .Net WCF interoperability

- ESB Connectivity - Messaging
  - Apache ActiveMQ
  - Sun OpenMQ
  - JBoss Messaging
  - Spring.NET Messaging API
    - JMS API for MS .NET

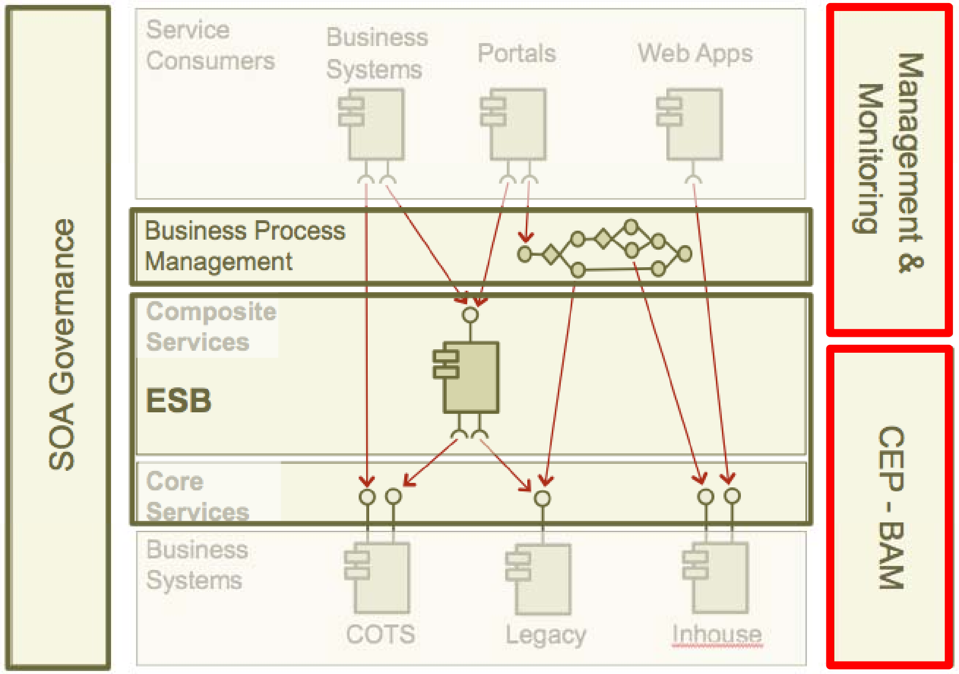
- ESB Connectivity – REST
  - Restlet
  - Jersey
    - JAX-RS (JSR 311)
  - Adbera
    - Atom Feeds

Open Source SOA



- BPM
  - Apache Ode
  - Sun BPEL Engine
  - JBoss jBPM
- SOA Governance Tools – Repository & Registry
  - MuleSource Galaxy
  - WSO2 Registry

Open Source SOA

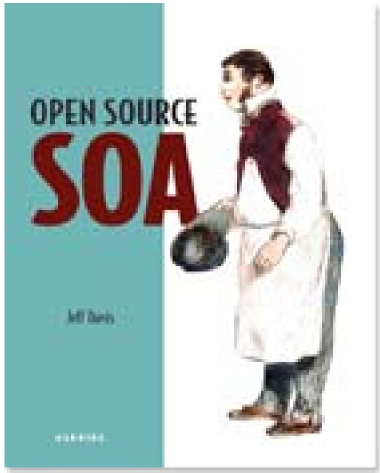


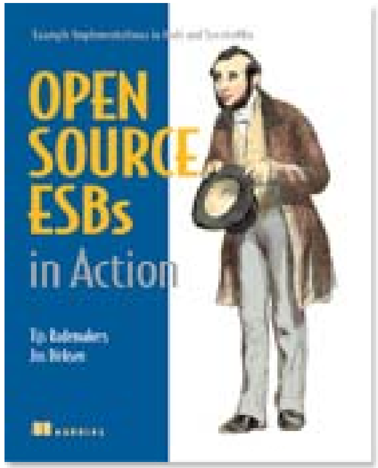
- BAM
  - Esper
  - JBoss Drools (since v5.0)
- Monitoring and Management
  - Hyperic HQ
  - MuleSource Mule HQ
  - Progress Fuse HQ
  - Builtin JMX access, e.g. in ESB products...

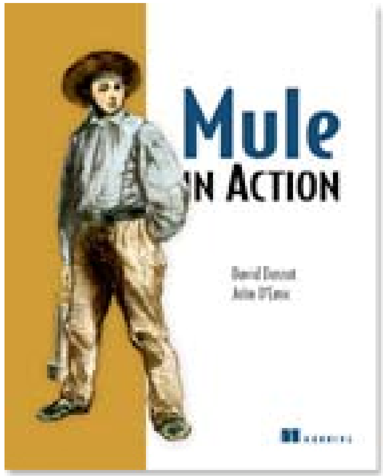
Summary

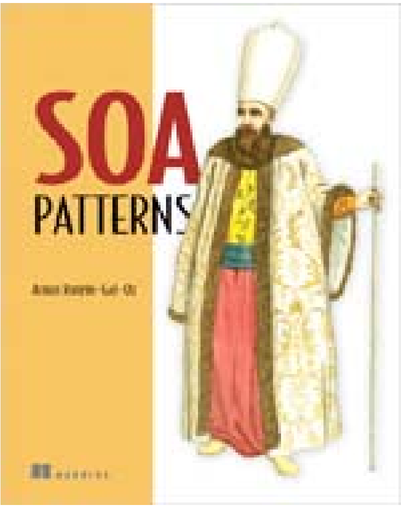
- Open Source products for SOA exists not only for core features such as Web Services, Messaging and ESB but also for
  - SOA Governance, Monitoring & Management and BAM
- Typical scenarios for service provisioning and consumption can be implemented very easy
- Advantages compared to commercial products
  - Lower license cost, obviously ☺
  - Lower complexity, at least for mainstream scenarios
  - Easier to extend
- Vendor service offerings for support, training and consulting
- Important to establish a reference model and guiding principles
  - SOA Governance, holistic view

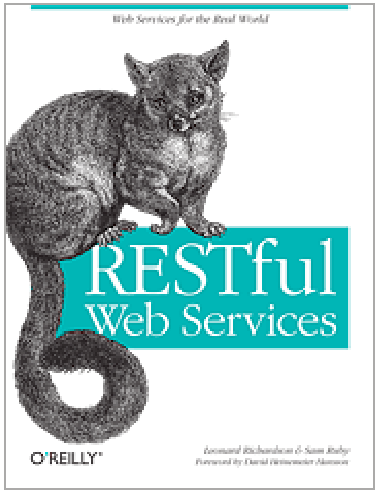
Recommended reading

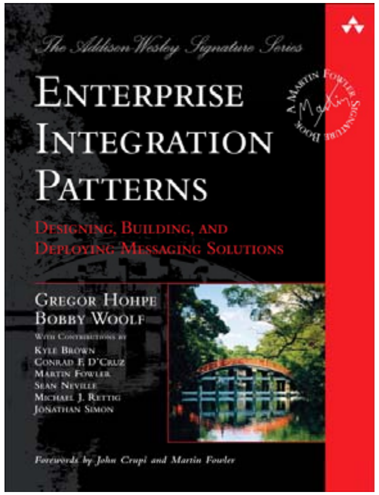



Example Implementations in Java and ServiceMix
OPEN SOURCE ESBs in Action
Tijs Rademakers
Jos Dirksen
MANNING






Web Services for the Real World
RESTful Web Services
O'REILLY
Leonard Richardson & Sam Ruby
Foreword by David Heinemeier Hansson


The Addison-Wesley Signature Series
A Martin Fowler Signature Book
Enterprise Integration Patterns
Designing, Building, and Deploying Messaging Solutions
Gregor Hohpe
Bobby Woolf
With Contributions by
Kyle Brown
Conrad F. D'Cruz
Martin Fowler
Sean Neville
Michael J. Rettig
Jonathan Simon
Forewords by John Crupi and Martin Fowler

Questions?

?

http://callistaenterprise.se

magnus.larsson@callistaenterprise.se