Apache Tuscany SCA

Luciano Resende
Senior Software Engineer, IBM Websphere Application Server, Apache Tuscany
lresende@apache.org

Haleh Mahbod
Program Director, IBM Websphere Application Server, Apache Tuscany
hmahbod@gmail.com

Apache Tuscany

- **Apache Tuscany Open Source (http://tuscany.apache.org)**
  - ✓ Anyone can participate in Apache open source projects

- **Tuscany supports SCA Standards (http://www.oasis-opencsa.org/) created by key Industry leaders/vendors**
  - ✓ Many vendors offer products that support SCA Standards (e.g. IBM, Oracle, Tibco, SAP, Roguewave, etc.)

- **Tuscany Goes beyond the standard**
  - ✓ Fosters innovation in SOA and SCA
  - ✓ Provides real user feedback and innovative ideas to Standards

- **Universities are using Tuscany/SCA to teach SOA across the world**
  - ✓ Many students are actively participating in Tuscany
    - o GSoC, Research Projects, Contributing thesis work, etc

- **You can participate in Tuscany**
  - ✓ Use the software and provide feedback: Ideas to enhance the software, bugs, scenarios,…
  - ✓ Help develop the infrastructure that eases development of SOA solutions and addresses IT challenges

Service Oriented Architecture (SOA)
What is SOA ?

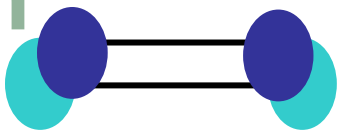
Service Oriented Architecture (SOA)

- **SOA is based on the concept of Services**
  - Services are reusable pieces of business functionality

- **SOA is an architectural approach that promotes**
  - Reusability
    - o Business functions (services) are composed together to form new business solutions.
  - Flexibility
    - o Business solutions can adopt to change

Service Component Architecture (SCA) role in SOA

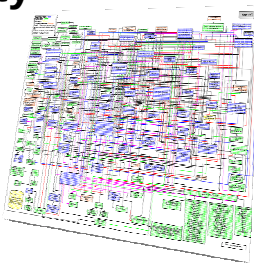
SOA Programming Model Evolution



More Flexibility

Greater Agility

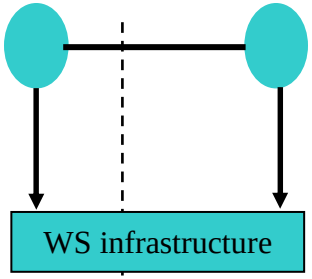
Higher ROI



Home Grown, Proprietary

X Business logic mixed with Service call API and protocol handling
X Single protocol enforced
X Cannot manage as part of the enterprise solution

Brittle application



Standards Based, JAX-WS

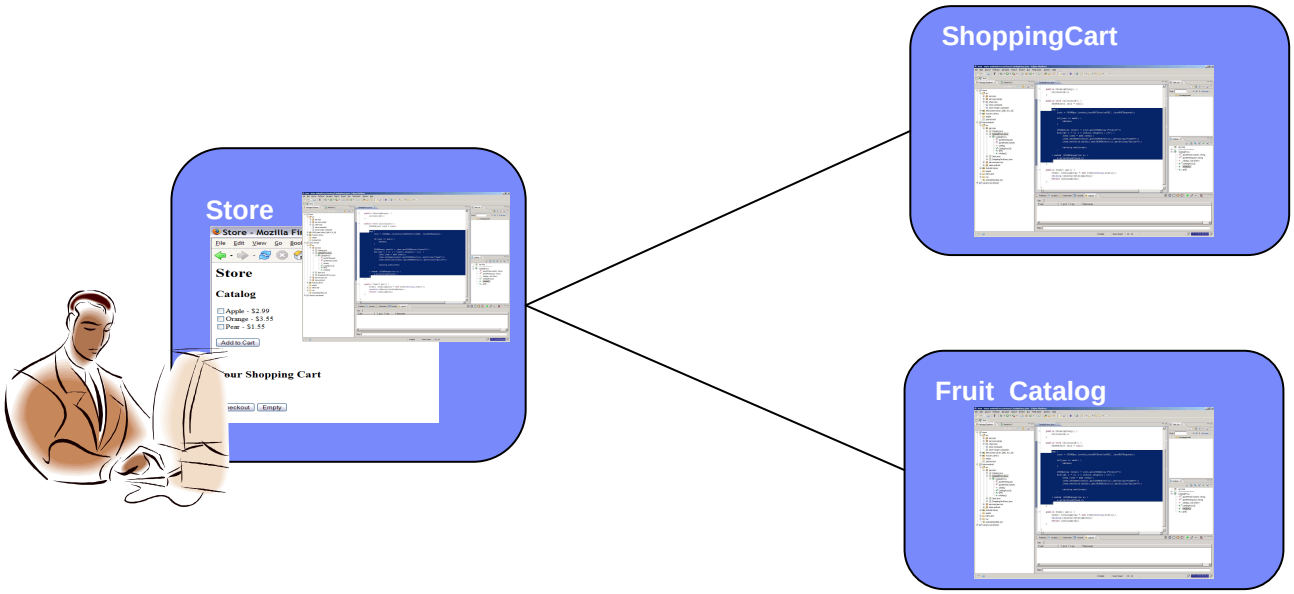
✓Business logic separate from Service call API, handled by infrastructure
X Business logic mixed with protocol handling
X Single protocol enforced, for example HTTP
X Single technology stack enforced for a solution
X Cannot manage as part of the enterprise solution

Service Component Architecture (SCA)

✓ Business logic separate from Service call API, handled by infrastructure
✓Business logic separate from protocol handling, handled by infrastructure
✓Multiple protocols supported. Protocol is a choice
✓Multiple technologies can be used in one solution
✓ Manage as part of the enterprise solution

Time

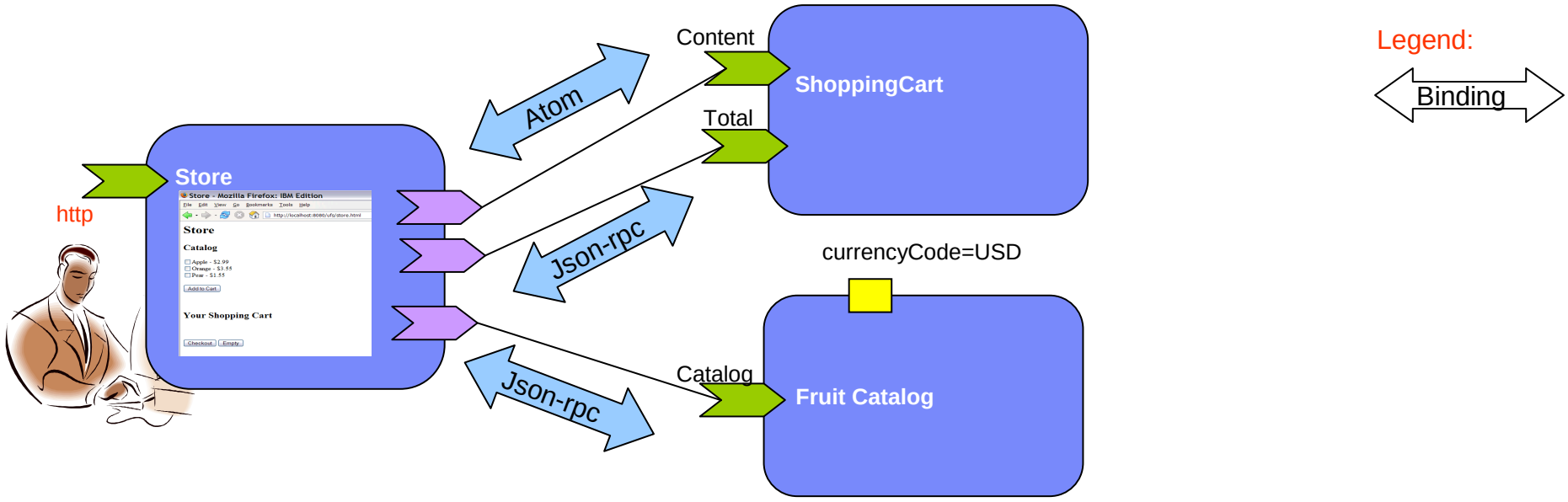
The Store Scenario



Challenges

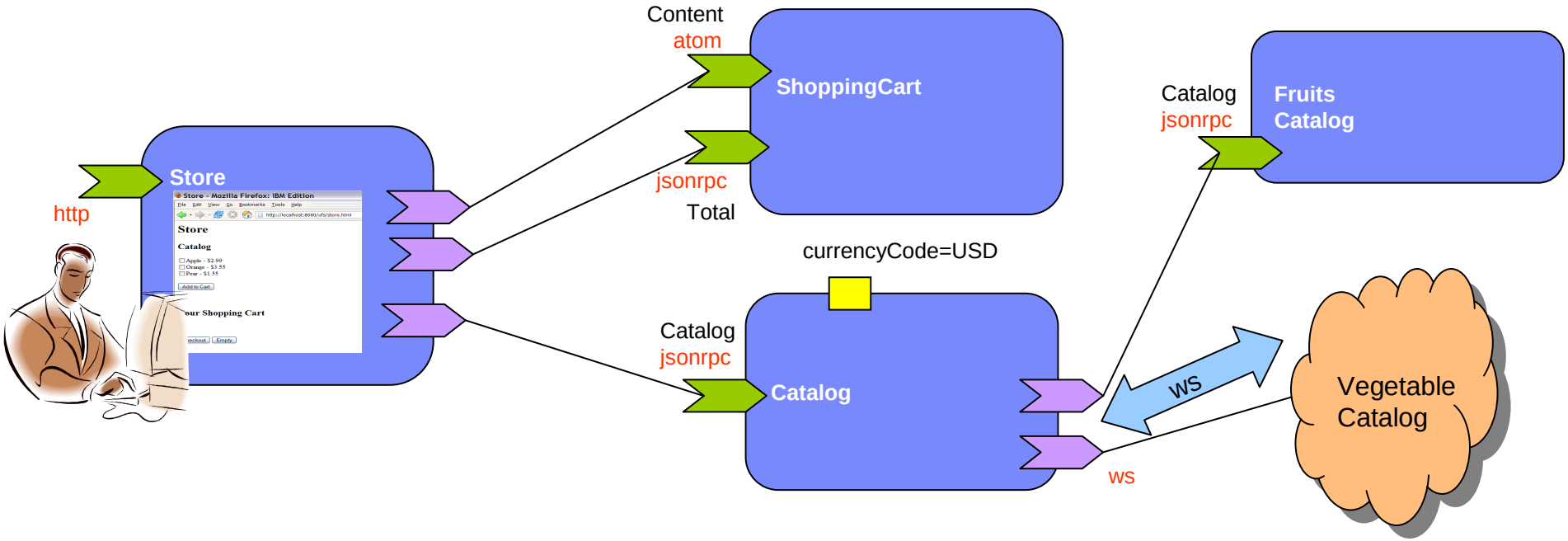
- Business logic mixed with protocol handling
  - Change in protocol requires change to business logic
  - Logic more complicated than it needs to be
- Cannot make the same service available trough different protocols without code change

The Store Scenario using SCA



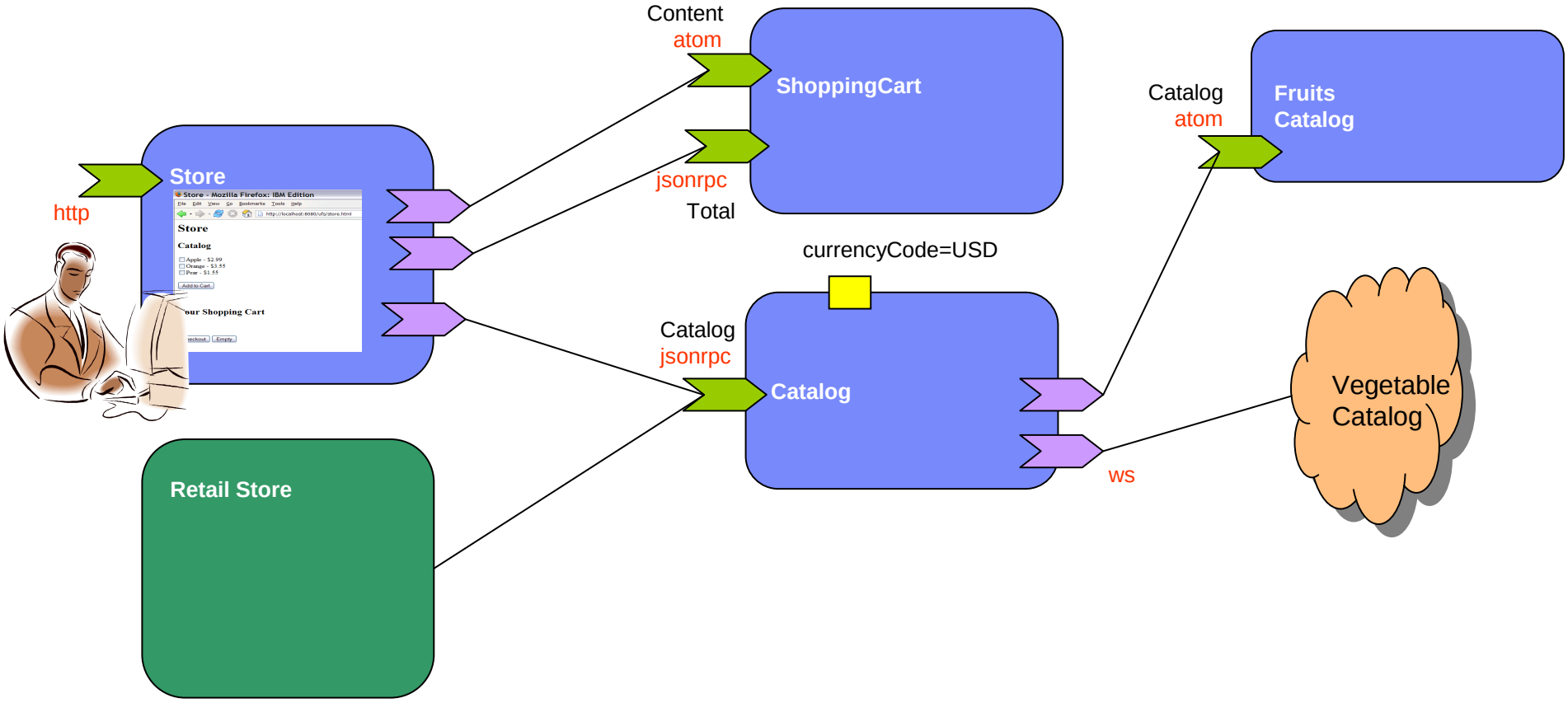
- **SCA separates business logic from infrastructure details.**
  - Services are reusable.
  - Protocol is handled by the infrastructure (Tuscany SCA Runtime in this case) through pluggable bindings
    - Bindings can change with minimal configuration change

The Store Scenario using SCA



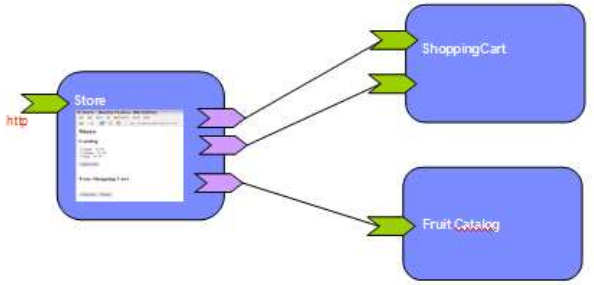
- **Solutions composed with SCA are extendible.**
- **Add a Vegetable Catalog service to the solution without recoding**
  - SCA composites can work with non-SCA services

The Store Scenario using SCA



- **Solutions composed with SCA can be accessed as a service.**
- **Make Fruit/Vegetable store available as a service to another company**
  - SCA composites can be called by non-SCA applications

SCA Assembly



Service Interface
- Java
- WSDL

Property

Reference Interface
- Java
- WSDL

Composite A

property setting

Service

Component A

Component B

Reference

promote

wire

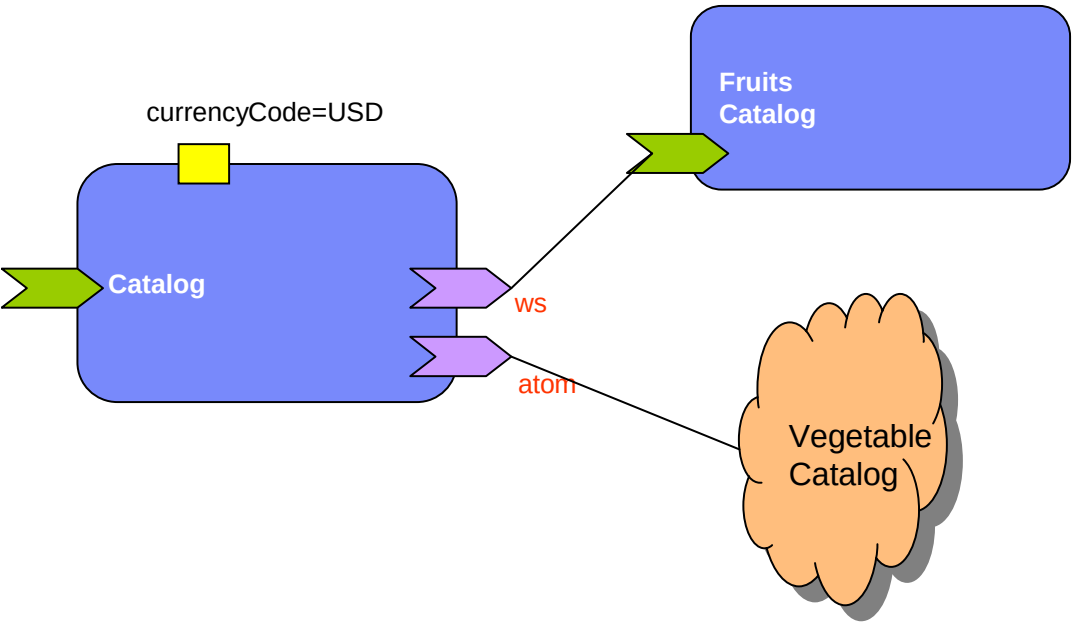
promote

Service Binding
- Web Service
- JMS
- JCA
- SLSB
- JSONRPC
- ATOM
- ...

Implementation
- Java
- BPEL
- SCA composite
- Spring
- EJB Module
- Java EE
- Scripting:Groovy, Jscript, PHP, Python,Ruby
- XQuery

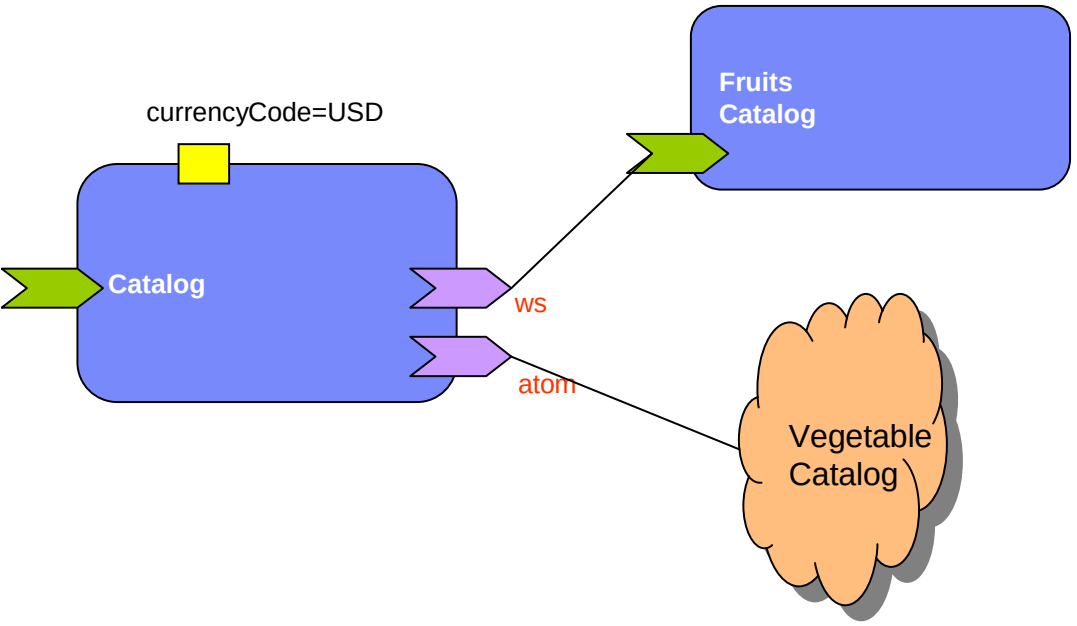
Reference Binding
- Web Service
- JMS
- JCA
- SLSB
- JSONRPC
- ATOM
- ...

Store Composition



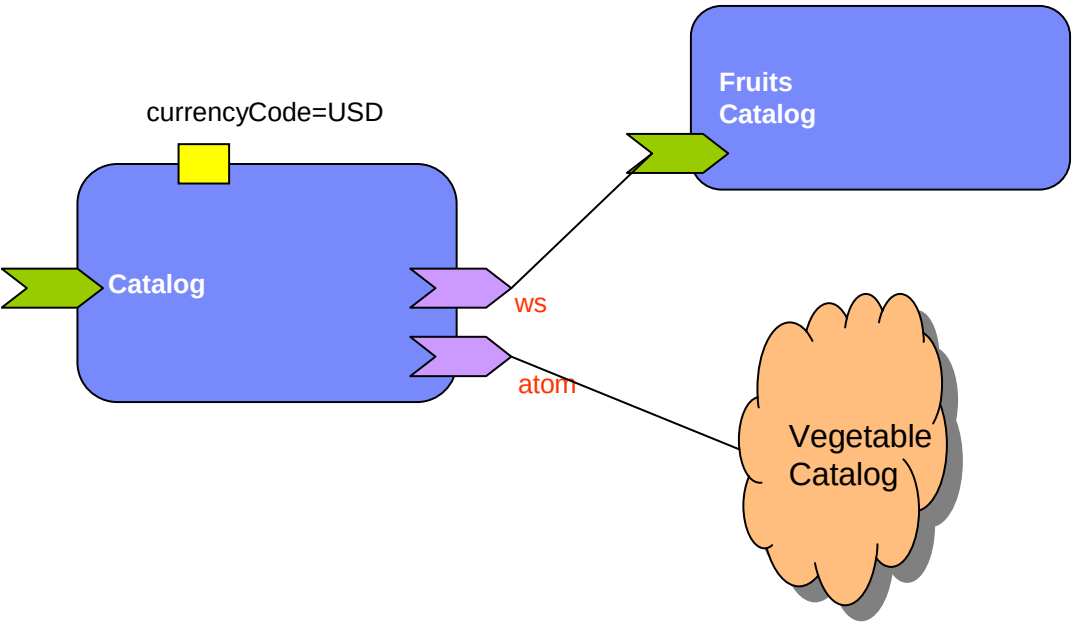

```
<composite      xmlns=http://www.osoa.org/xmlns/sca/1.0
    xmlns:t=http://tuscany.apache.org/xmlns/sca/1.0
    targetNamespace=http://store    name="catalog">
    <component name="Catalog">
            <implementation.java class="services.CatalogImpl"/>
        <service name="Catalog">
            <t:binding.jsonrpc/>
        </service>
        <reference name="fruitsCatalog" target="FruitsCatalog">
            <binding.ws/>
        </reference>
        <reference name="vegetablesCatalog"
    target="VegetablesCatalog">
            <t:binding.atom/>
        </reference>
     </component>
    <component name="FruitsCatalog">
       <implementation.java class="services.FruitsCatalogImpl"/>
       <service name="Catalog">
          <binding.ws/>
          <t:binding.rss/>
       </service>
    </component>
    <component name="VegetablesCatalog">
       <implementation.java
    class="services.VegetablesCatalogImpl"/>
       <service name="Catalog">
          <t:binding.atom/>
          <t:binding.jsonrpc/>
       </service>
    </component>
</composite>
```

Store Composition



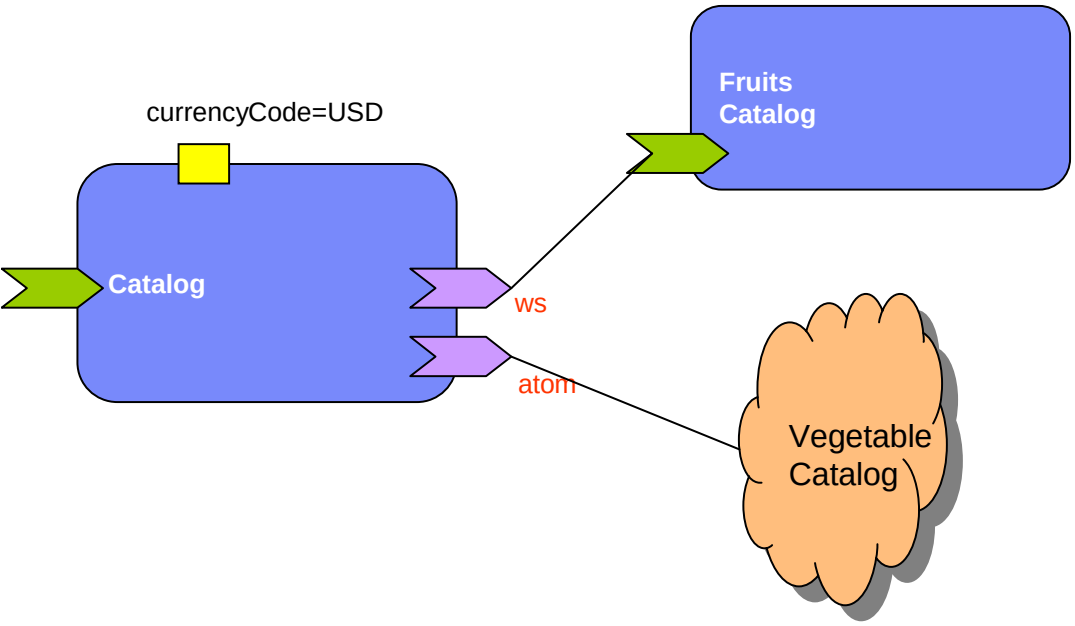

```
<composite     xmlns=http://www.osoa.org/xmlns/sca/1.0
    xmlns:t=http://tuscany.apache.org/xmlns/sca/1.0
    targetNamespace=http://store   name="catalog">
    <component name="Catalog">
          <implementation.java class="services.CatalogImpl"/>
        <service name="Catalog">
           <t:binding.jsonrpc/>
        </service>
        <reference name="fruitsCatalog" target="FruitsCatalog">
           <binding.ws/>
        </reference>
        <reference name="vegetablesCatalog"
    target="VegetablesCatalog">
           <t:binding.atom/>
        </reference>
     </component>
    <component name="FruitsCatalog">
       <implementation.java class="services.FruitsCatalogImpl"/>
       <service name="Catalog">
          <binding.ws/>
          <t:binding.rss/>
       </service>
    </component>
    <component name="VegetablesCatalog">
       <implementation.java
    class="services.VegetablesCatalogImpl"/>
       <service name="Catalog">
          <t:binding.atom/>
          <t:binding.jsonrpc/>
       </service>
    </component>
</composite>
```

Store Composition




```
<composite     xmlns=http://www.osoa.org/xmlns/sca/1.0
    xmlns:t=http://tuscany.apache.org/xmlns/sca/1.0
    targetNamespace=http://store   name="catalog">
    <component name="Catalog">
        <implementation.java class="services.CatalogImpl"/>
        <service name="Catalog">
            <t:binding.jsonrpc/>
        </service>
        <reference name="fruitsCatalog" target="FruitsCatalog">
            <binding.ws/>
        </reference>
        <reference name="vegetablesCatalog"
    target="VegetablesCatalog">
            <t:binding.atom/>
        </reference>
    </component>
    <component name="FruitsCatalog">
        <implementation.java class="services.FruitsCatalogImpl"/>
        <service name="Catalog">
            <binding.ws/>
            <t:binding.rss/>
        </service>
    </component>
    <component name="VegetablesCatalog">
        <implementation.java
    class="services.VegetablesCatalogImpl"/>
        <service name="Catalog">
            <t:binding.atom/>
            <t:binding.jsonrpc/>
        </service>
    </component>
</composite>
```

Store Composition



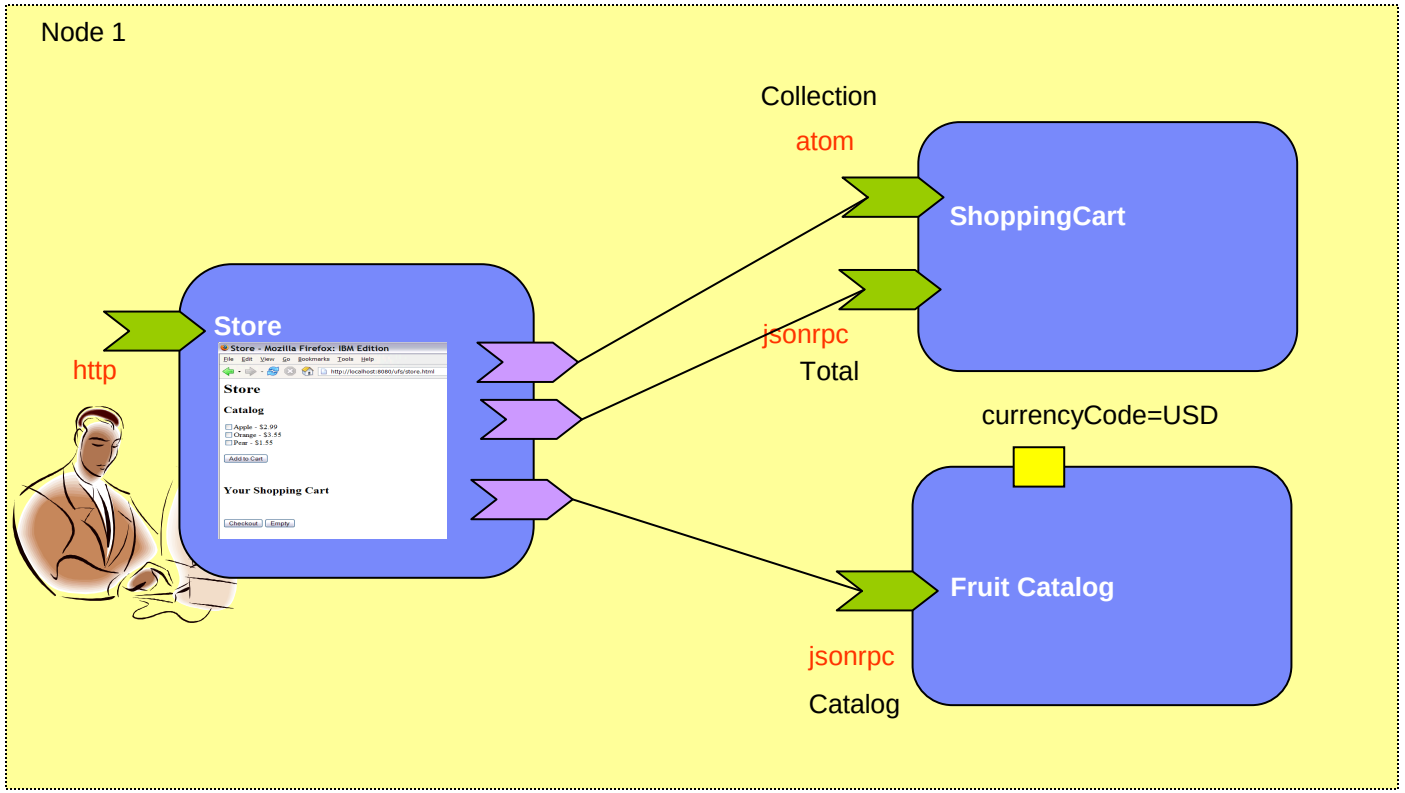

```
<composite     xmlns=http://www.osoa.org/xmlns/sca/1.0
    xmlns:t=http://tuscany.apache.org/xmlns/sca/1.0
    targetNamespace=http://store   name="catalog">
    <component name="Catalog">
        <implementation.java class="services.CatalogImpl"/>
        <service name="Catalog">
            <t:binding.jsonrpc/>
        </service>
        <reference name="fruitsCatalog" target="FruitsCatalog">
            <binding.ws/>
        </reference>
        <reference name="vegetablesCatalog"
    target="VegetablesCatalog">
            <t:binding.atom/>
        </reference>
    </component>
    <component name="FruitsCatalog">
        <implementation.java class="services.FruitsCatalogImpl"/>
        <service name="Catalog">
            <binding.ws/>
            <t:binding.rss/>
        </service>
    </component>
    <component name="VegetablesCatalog">
        <implementation.java
    class="services.VegetablesCatalogImpl"/>
        <service name="Catalog">
            <t:binding.atom/>
            <t:binding.jsonrpc/>
        </service>
    </component>
</composite>
```

Deployment

- **Deployment is Flexible**
  - Can deploy in a variety of different topologies
    - o Single JVM
    - o Different JVMs
    - o Single machine
    - o Distributed over physical hardware

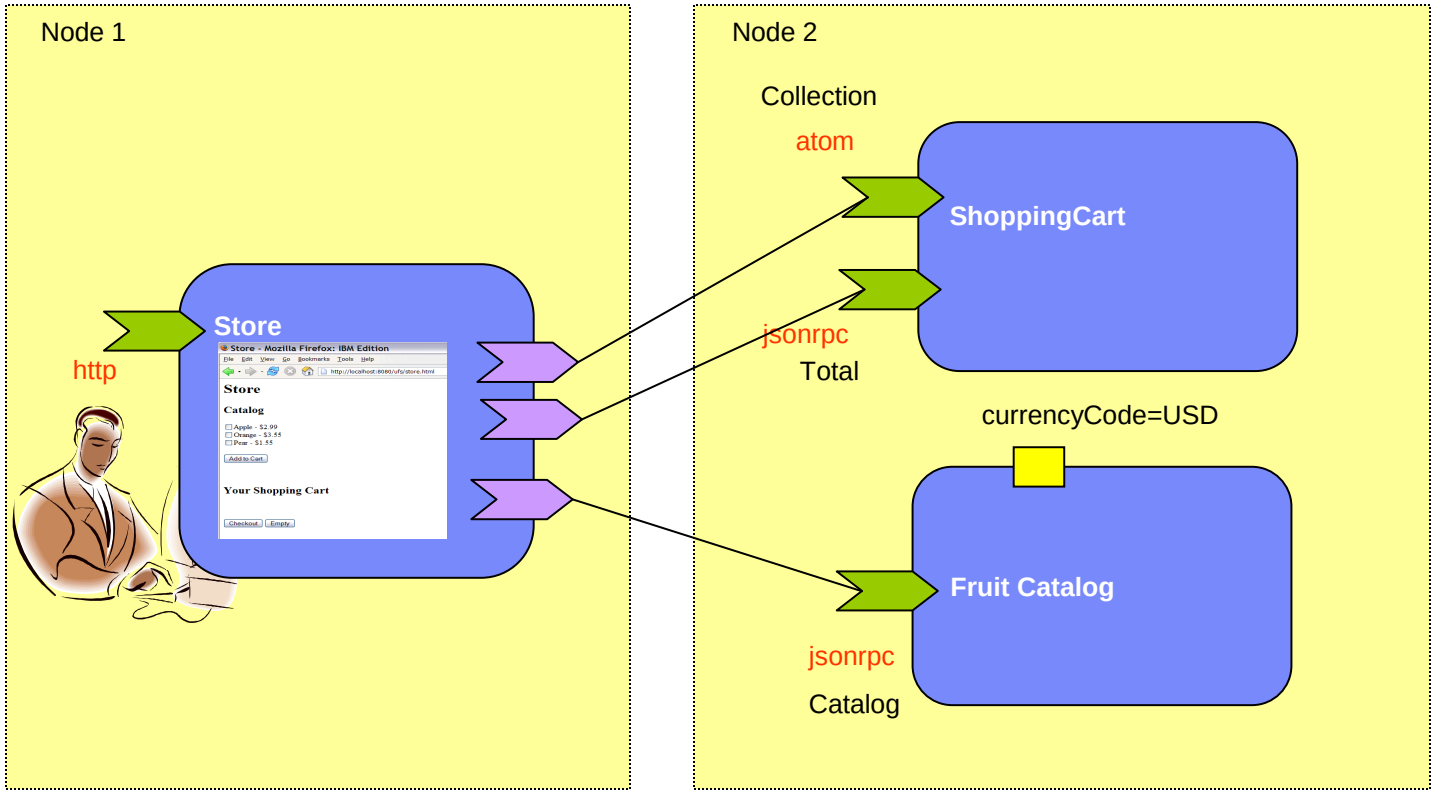
- **Solution becomes Scalable**
  - Redistribute application without code changes

Store - Single Node Deployment



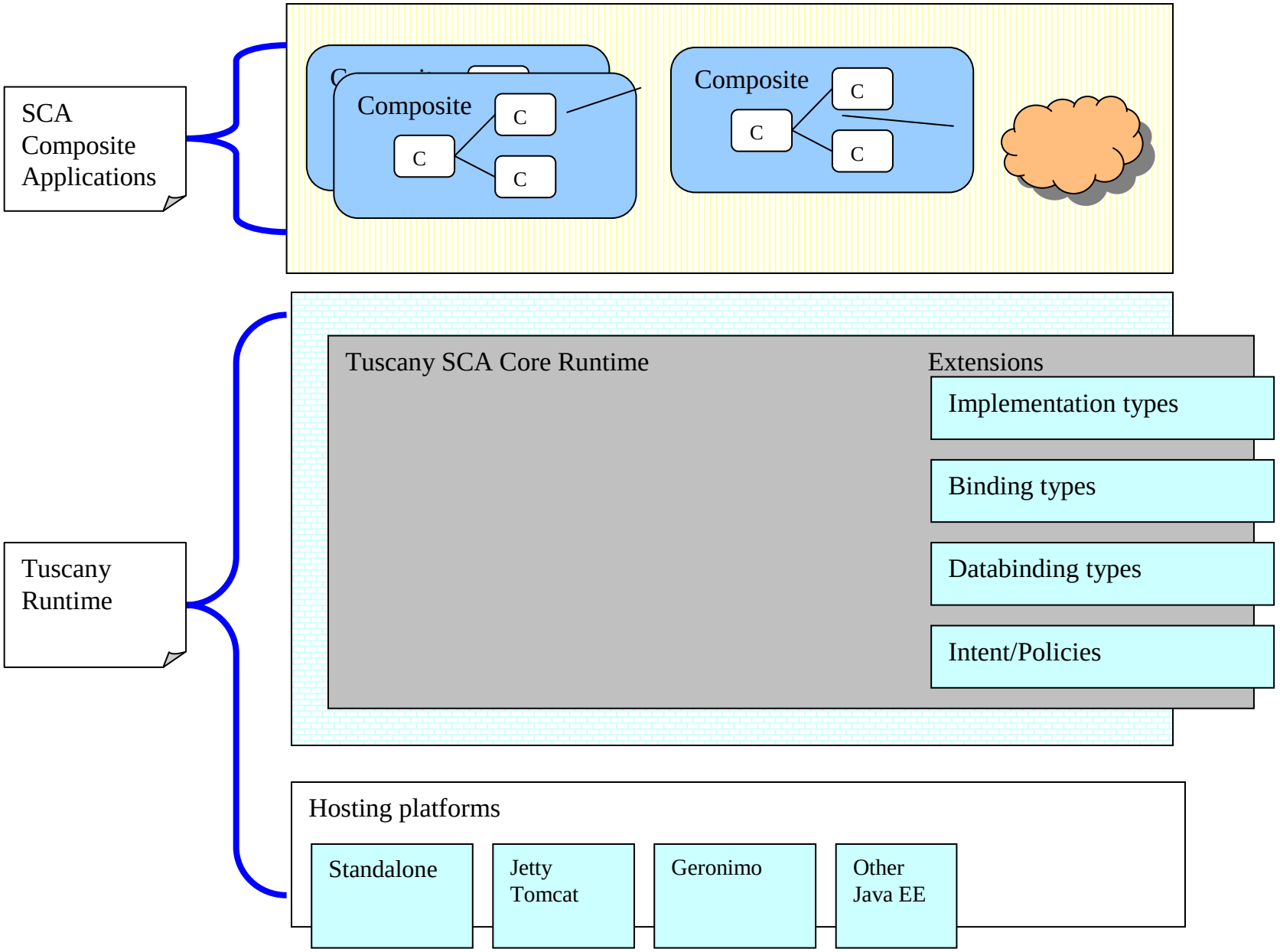
- **Physical topology is specified separately from the logical assembly model.**

Store – Multiple node deployment



- **A distributed deployment of the assembly.**

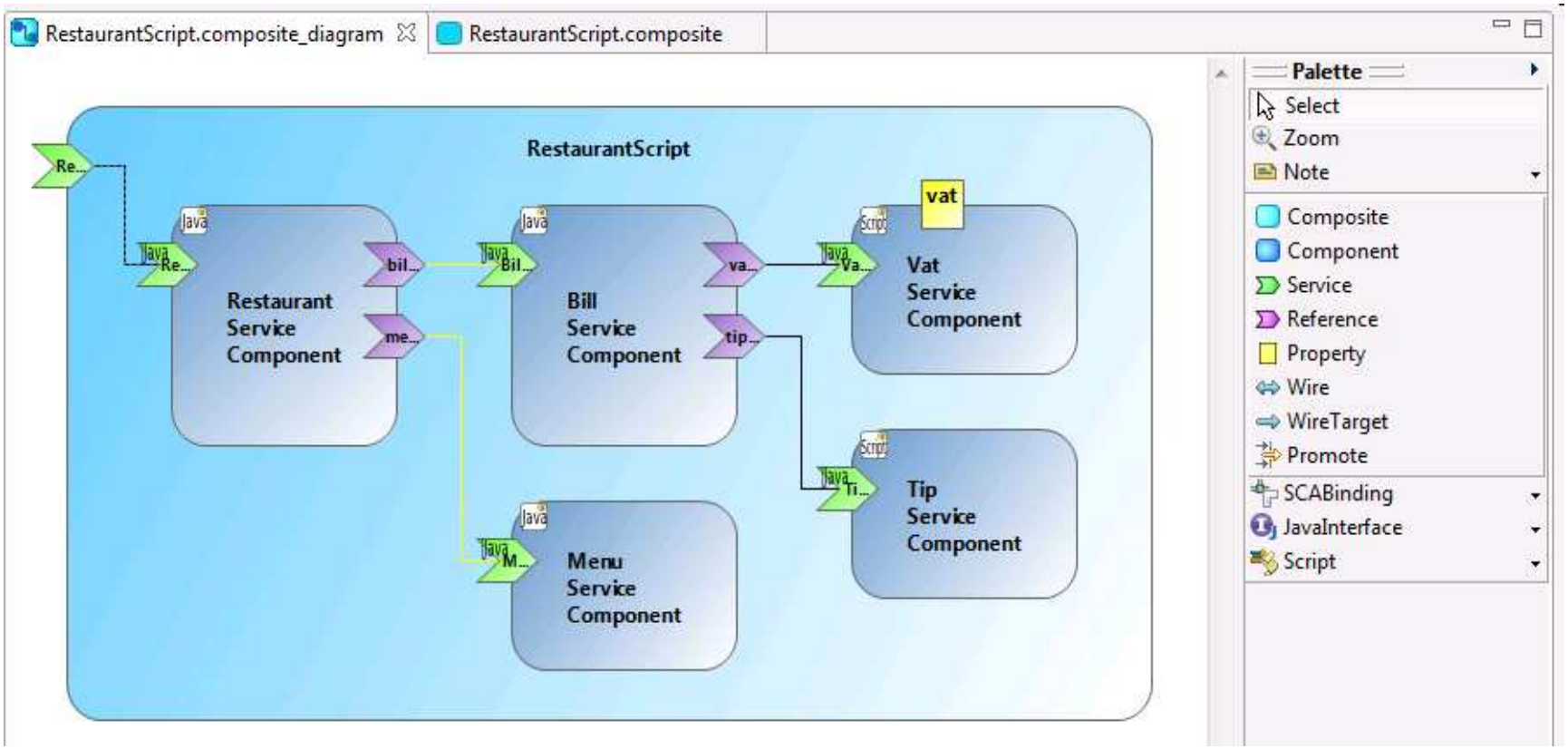
Tuscany Architecture



SCA Tooling

Eclipse Tools

- **Eclipse STP SCA project**



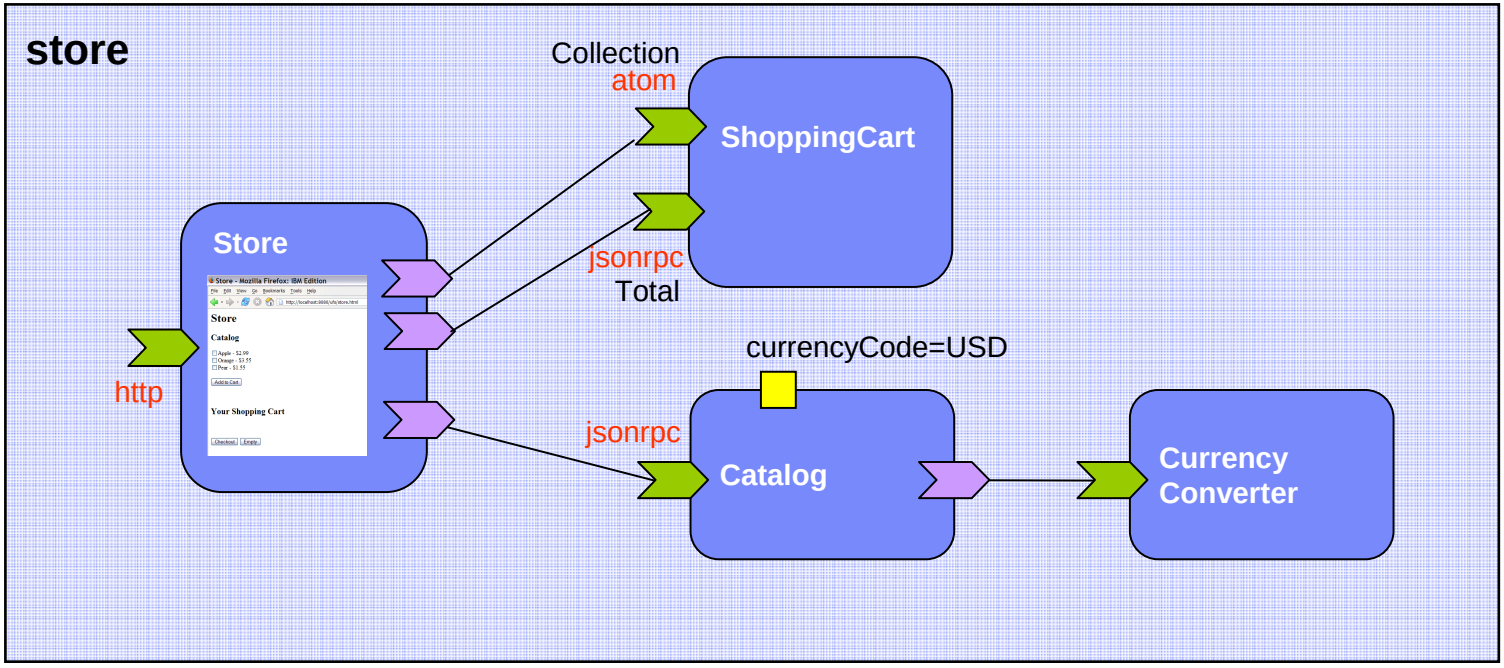
http://www.eclipse.org/stp/sca/index.php

Apache Tuscany Demo
The Raise of a Fruit Business

http://www.rarefruit.org/photos/fruit_store.jpg

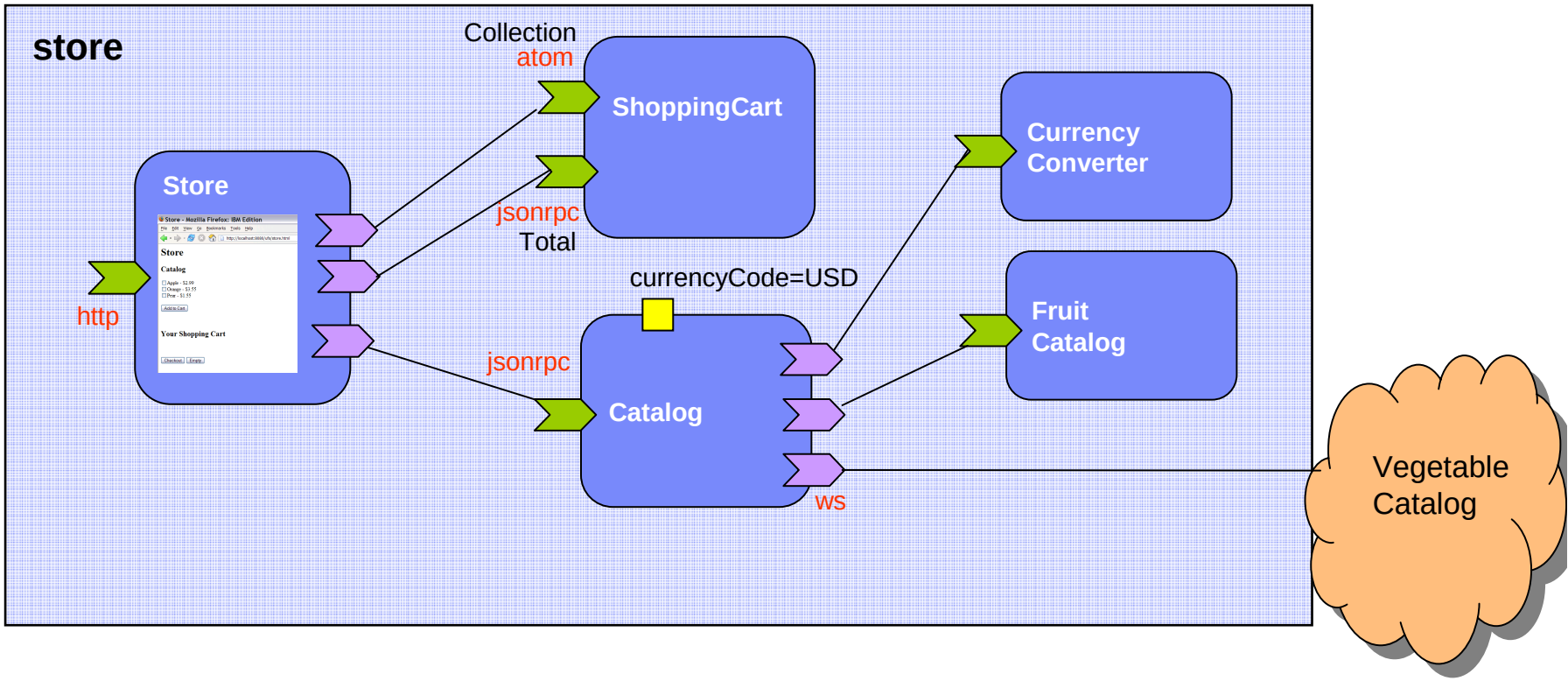
The Fruit Store

Creating an Online Business



The Fruit&Vegetable Store

Merger or Acquisition



How do I get Involved ?

- **Take a look at Apache Tuscany**
  - http://tuscany.apache.org/
- **Download latest release**
  - http://tuscany.apache.org/sca-java-releases.html
- **Join the active developer and user communities**
  - http://incubator.apache.org/tuscany/getting-involved.html
- **Learn more about Tuscany/SCA**
  - http://tuscany.apache.org/tuscany-dashboard.html
- **You are very welcome to get involved in the project in any way you want to, here are some examples.**
  - Try out the software and give us your feedback
  - Record bugs (JIRA) for any enhancements you want or problems you find
  - Suggest and develop new extensions
  - Provide those bits of documentation that you think are missing or can be improved
  - Get involved in the development of the core infrastructure

THANK YOU !!!