Enterprise-SOA with UML+SoaML For Healthcare

Cory Casanave

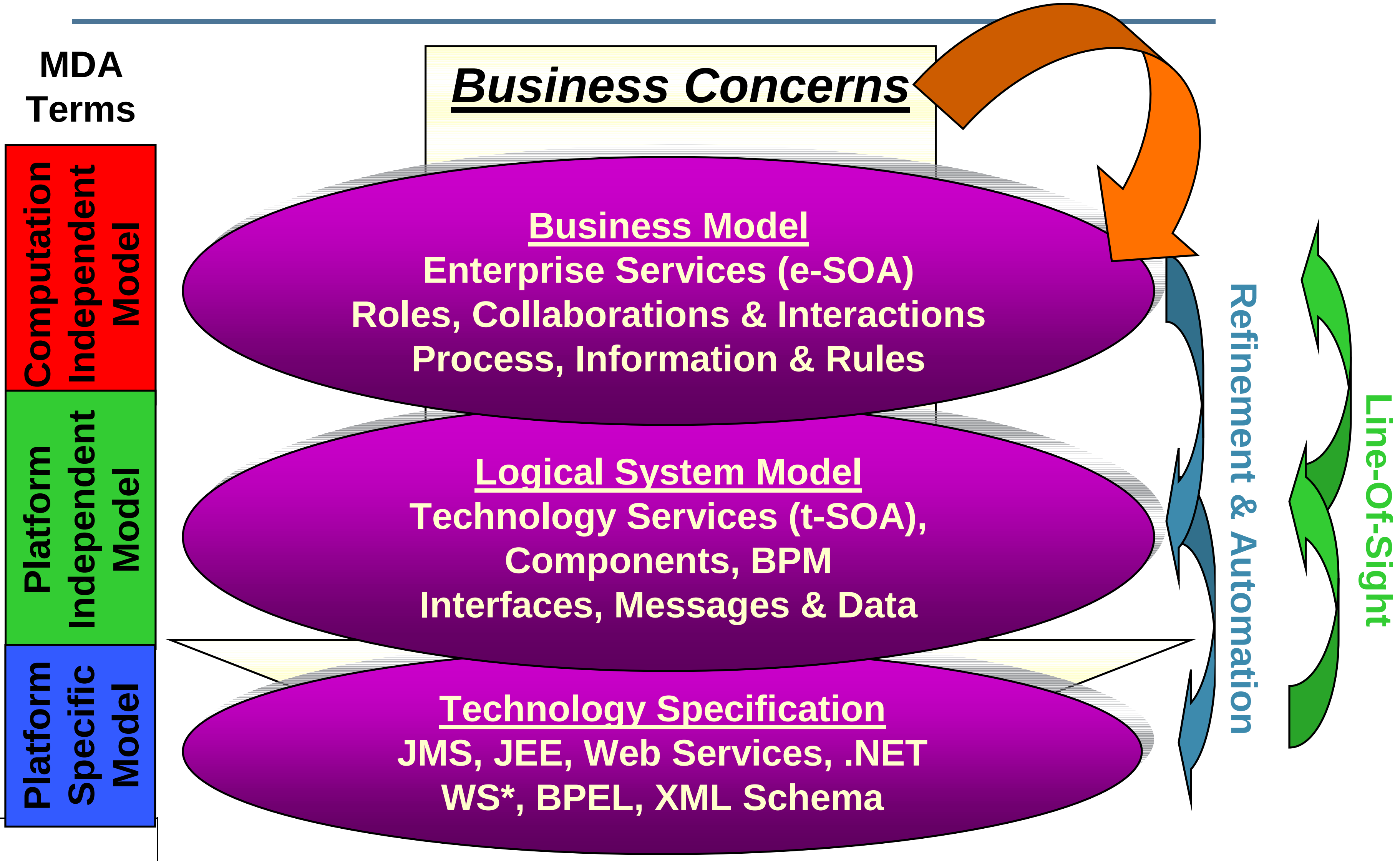
What is SoaML?

- An OMG Standard for Modeling Service Oriented Architectures
  - Adopted from the UML® Profile for Modeling Services (UPMS) RFP
  - SoaML supports the "A" in SOA
  - Used for modeling SOA at the business, enterprise and technology levels
  - Leverages Model Driven Architecture
- A "Profile" of the Unified Modeling Language™
  - Can be used with off-the-shelf UML tools as well as customized tooling
- In the "finalization" stage of the OMG process – essentially an adopted "beta" specification
  - Finalization with minor clean-up expected to complete this year
- Tool support & implementations already exist
  - Tool support – making it easy to create services models
  - MDA Implementations – provisioning web services, business artifacts and implementations from SoaML models

UML+SoaML & RIM

- UML Provide a broad rangge of modeling capabilities supported by standards, products and a community
- There are a lot of existing resources (produces, training, books, experienced modelers)
- RIM provides healthcare specific capabilities, but less support
- We are supporting the activities to evaluate a RIM profile of UML
- From what we know now UML+SoaML can provide most if not all of the RIM capabilities when extended with a profile
- There is no reason for Healthcare to live in a silo!

Context for Enterprise SOA



MDA Terms

Computation Independent Model

Platform Independent Model

Platform Specific Model

Business Concerns

Business Model
Enterprise Services (e-SOA)
Roles, Collaborations & Interactions
Process, Information & Rules

Logical System Model
Technology Services (t-SOA),
Components, BPM
Interfaces, Messages & Data

Technology Specification
JMS, JEE, Web Services, .NET
WS*, BPEL, XML Schema

Refinement & Automation

Line-Of-Sight

What does UML+SoaML do for me?

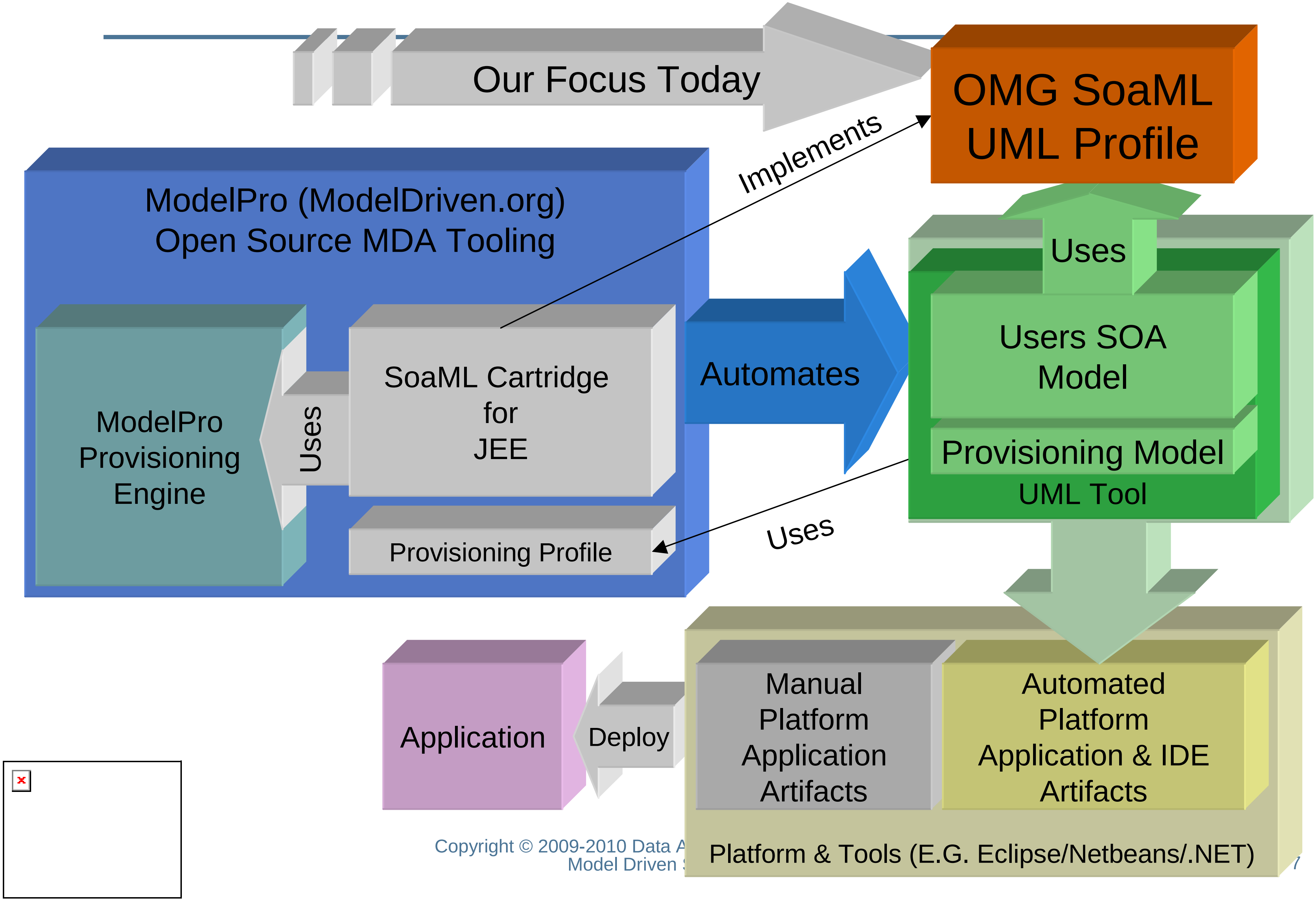
- What SoaML does for you depends on who you are!
  - an industry group, community or standards organization
  - a CEO, CFO or LOB Executive
  - a CIO or CTO
  - an I.T. Program Manager
  - a Business or Enterprise Architect
  - a SOA Solutions Architect
  - a Developer (Individual, Team or Contractor)

- This illustrates a strong point of UML+SoaML
  - It has scope to address stakeholders across the business and I.T. landscape using the services paradigm
  - SoaML has the capability to relate viewpoints meaningful to these diverse stakeholders
  - Only the parts of SoaML meaningful to the stakeholders need be used

How does SoaML Accomplish these goals?

- Representation of the Services Architecture
  - Roles of participants and the services the provide and use for a business purpose – gives services a context
  - Participants can be organizations, people or system components
  - Capabilities provided and used
- Specification of Services at the business and technical level
  - Including simple, bi-directional and n-ary services
  - Abstract information model connected to the messaging model
  - Choreography of service interactions
- Specification of Components linked to business architecture
  - Including composites (Compose applications)
  - Can be implemented with business processes & process execution
  - Implement with new capabilities or by adapting existing systems
- Model Driven Architecture Provisioning
  - Map the SoaML model to technology and other artifacts to avoid manual work and errors

Relating the Parts for Model Driven SOA



Our Focus Today

OMG SoaML UML Profile

Implements

ModelPro (ModelDriven.org) Open Source MDA Tooling

ModelPro Provisioning Engine

Uses

SoaML Cartridge for JEE

Provisioning Profile

Automates

Uses

Users SOA Model

Provisioning Model

UML Tool

Uses

Application

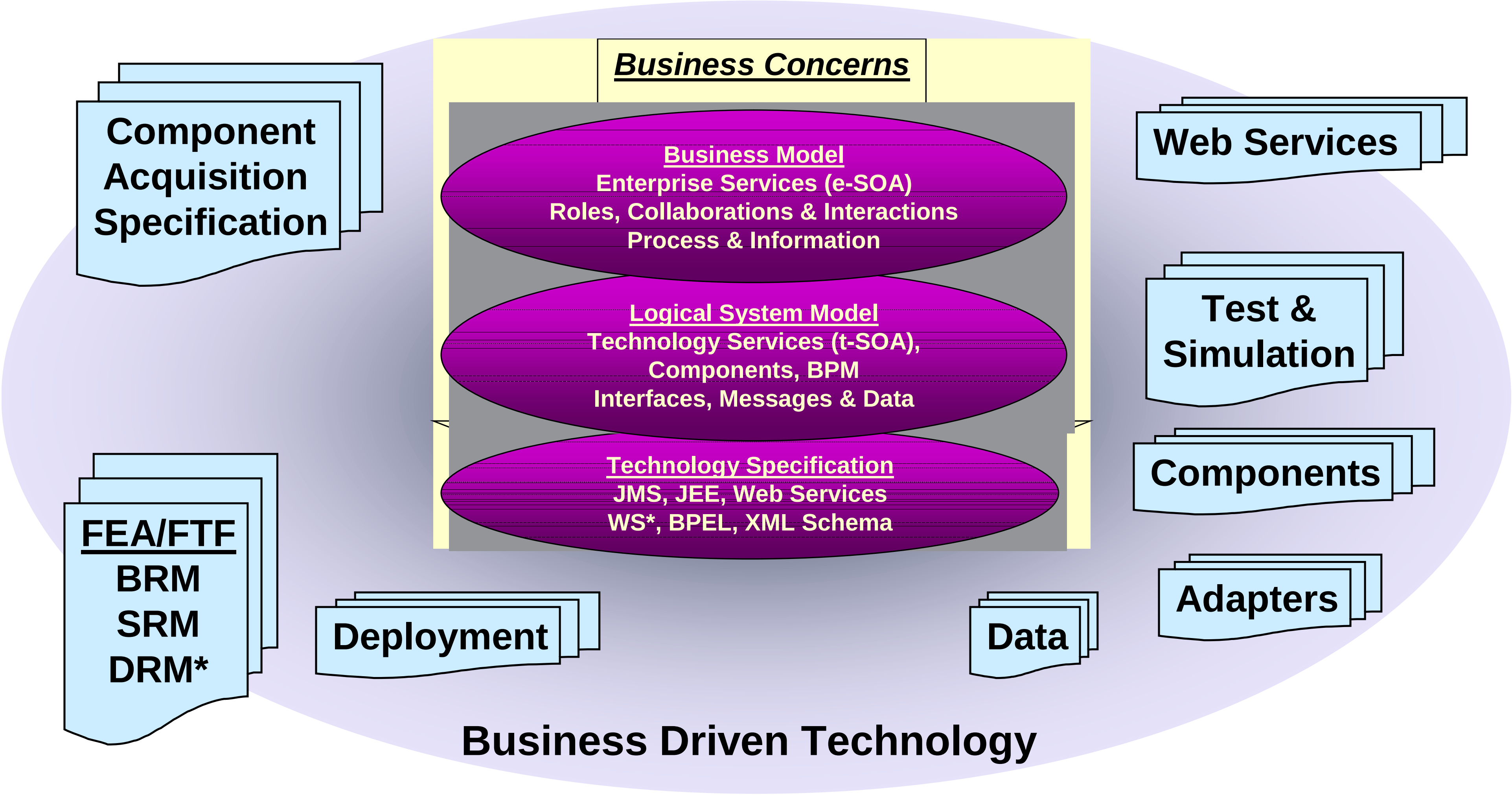
Deploy

Manual Platform Application Artifacts

Automated Platform Application & IDE Artifacts

Platform & Tools (E.G. Eclipse/Netbeans/.NET)

Value derived from the architecture with MDA



Business Concerns

Business Model
Enterprise Services (e-SOA)
Roles, Collaborations & Interactions
Process & Information

Logical System Model
Technology Services (t-SOA),
Components, BPM
Interfaces, Messages & Data

Technology Specification
JMS, JEE, Web Services
WS*, BPEL, XML Schema

Component Acquisition Specification

FEA/FTF
BRM
SRM
DRM*

Deployment

Data

Web Services

Test & Simulation

Components

Adapters

Business Driven Technology

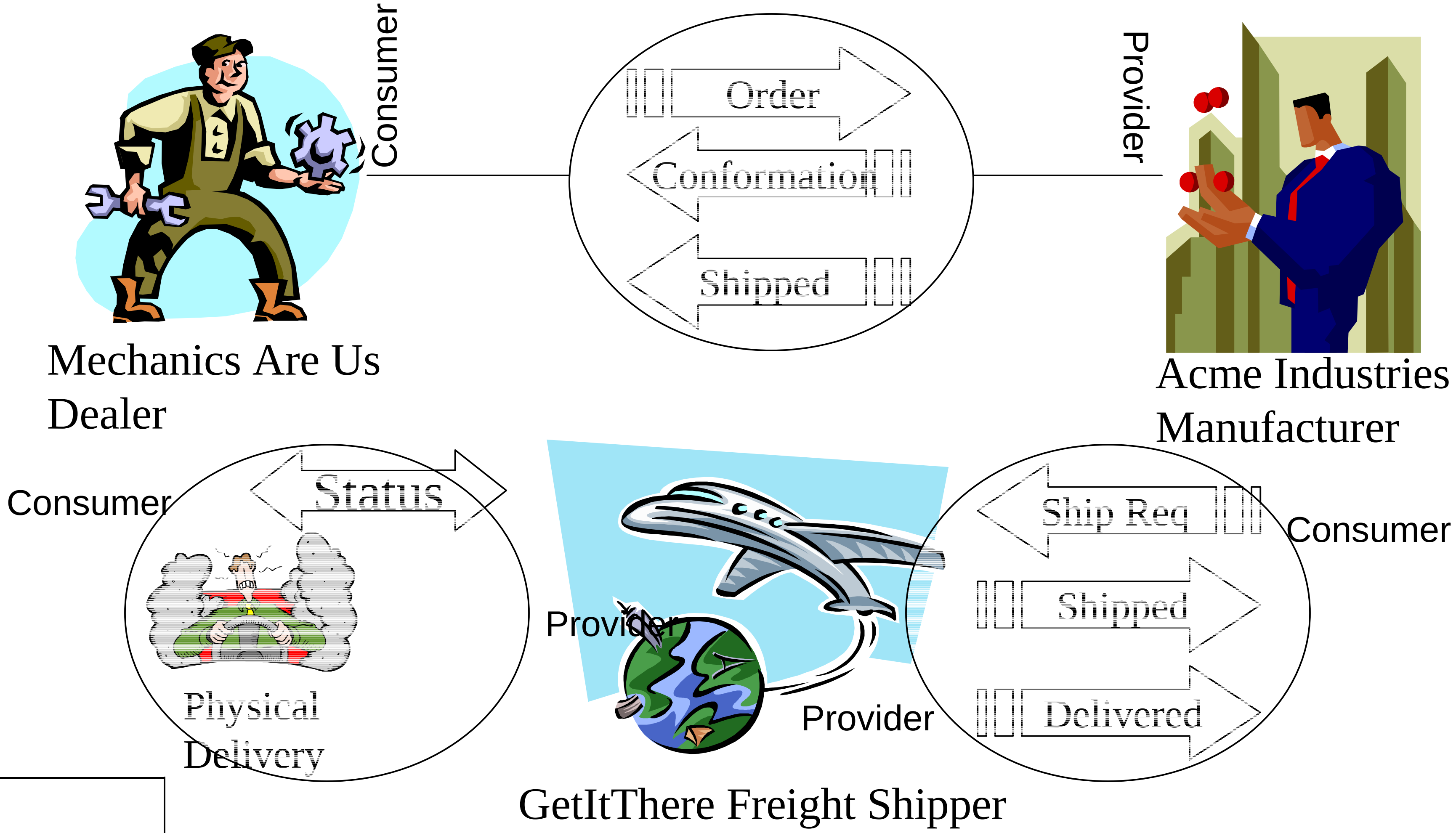
Facilitating Business Processes

SoaML Example – The “Dealer Network”

The dealer network models an “industry community” of dealers, shippers and manufacturers. The community defines the SOA architecture by which they all work together.

Note – This example is presented top-down, integrating both the business and technology viewpoints. SoaML can be used top-down, bottom up or middle out. It can be focused on the business and/or the technology based on the users needs.

Marketplace Services



Consumer

Order

Conformation

Shipped

Provider

Mechanics Are Us
Dealer

Acme Industries
Manufacturer

Consumer

Status

Physical
Delivery

Provider

Provider

Ship Req

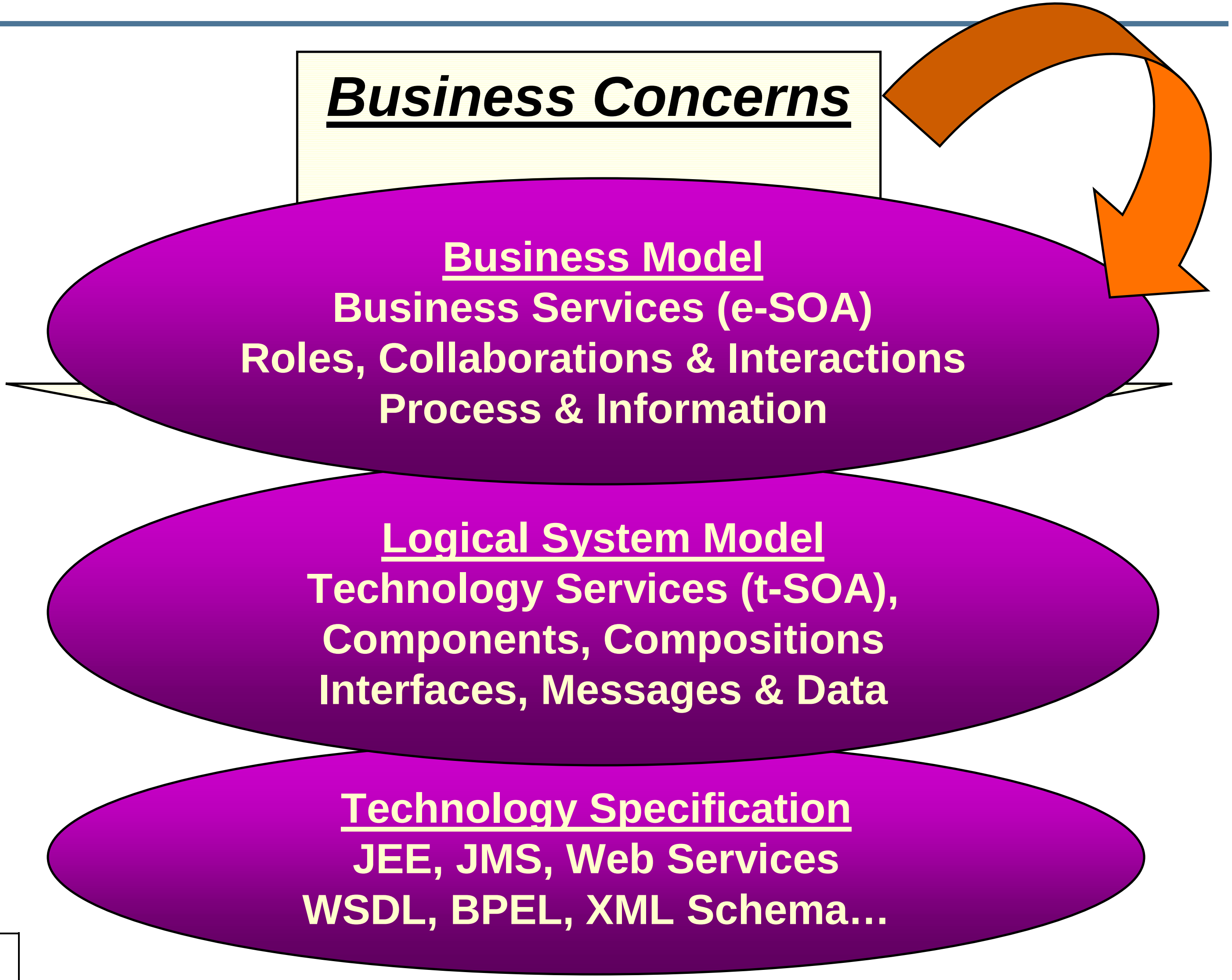
Shipped

Delivered

Consumer

GetItThere Freight Shipper

Focus on the Business Model



Business Concerns

Business Model

Business Services (e-SOA)

Roles, Collaborations & Interactions

Process & Information

Logical System Model

Technology Services (t-SOA),

Components, Compositions

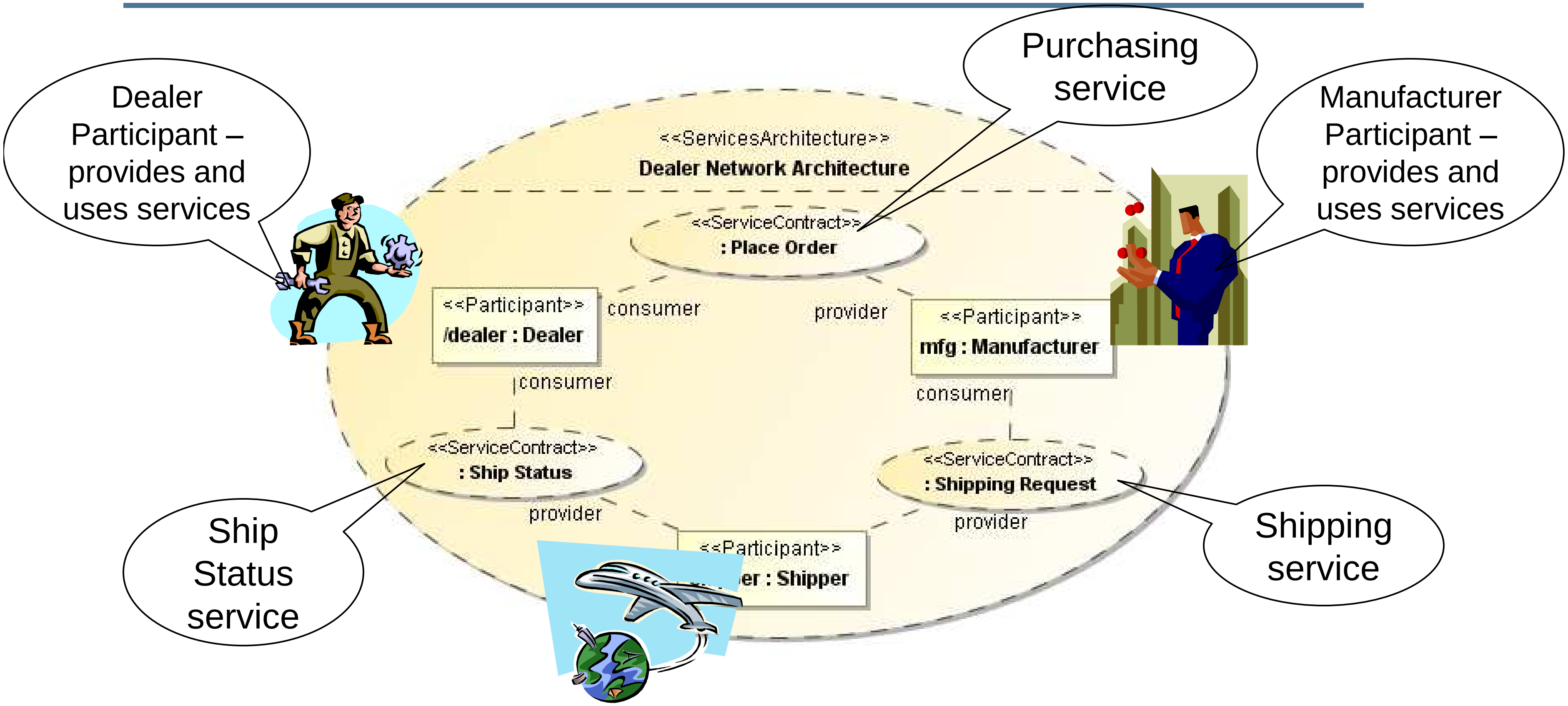
Interfaces, Messages & Data

Technology Specification

JEE, JMS, Web Services

WSDL, BPEL, XML Schema…

Services Architecture for the Dealer Network



Dealer Participant – provides and uses services

Purchasing service

Manufacturer Participant – provides and uses services

Ship Status service

Shipping service

<<ServicesArchitecture>>
Dealer Network Architecture

<<ServiceContract>>
: Place Order

<<Participant>>
/dealer : Dealer

<<Participant>>
mfg : Manufacturer

consumer

provider

consumer

consumer

<<ServiceContract>>
: Ship Status

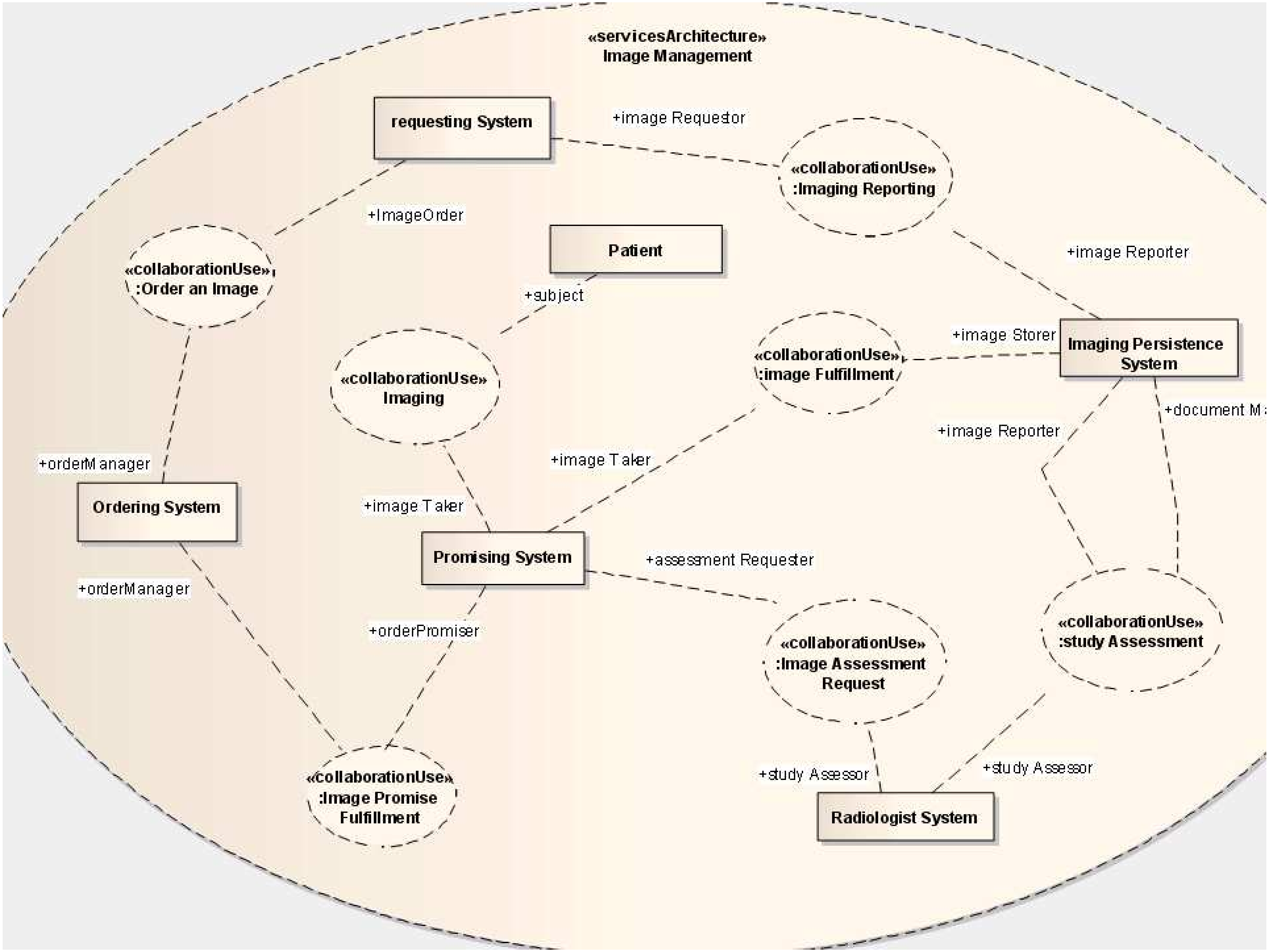
<<ServiceContract>>
: Shipping Request

provider

provider

<<Participant>>
er : Shipper

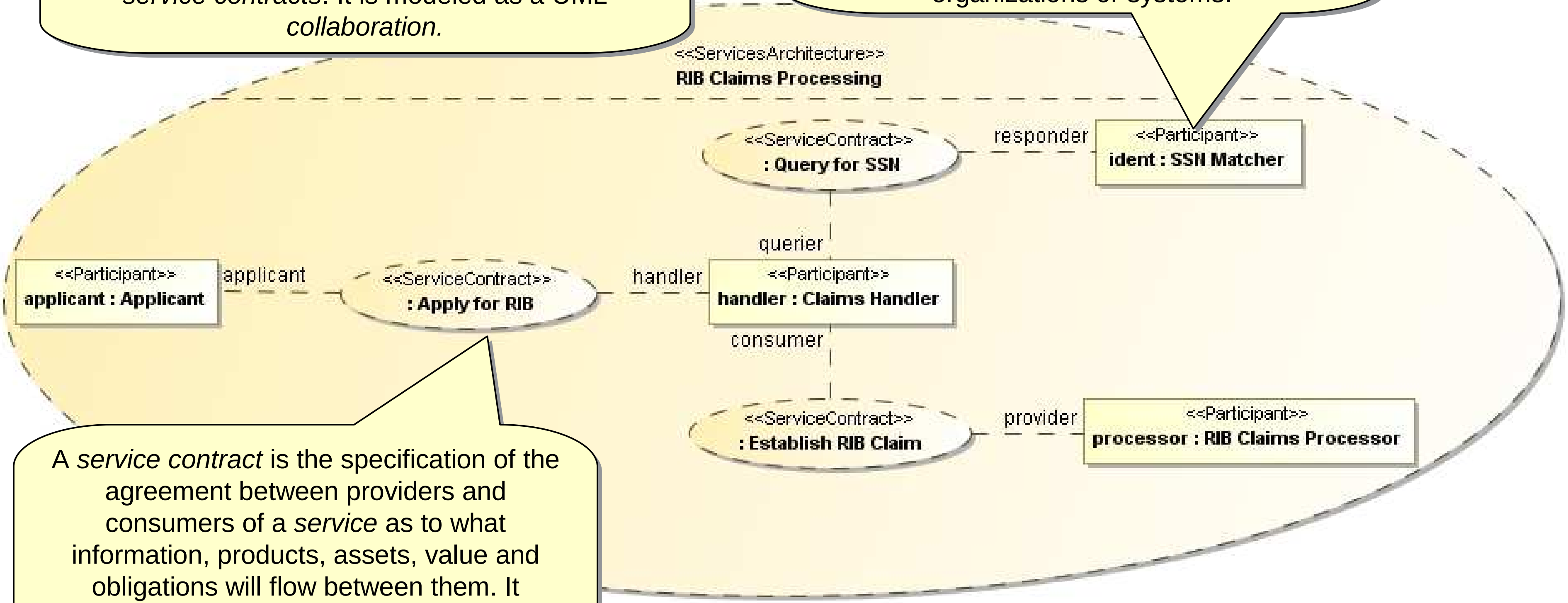
A ServicesArchitecture (or SOA) is a network of participant roles *providing* and *consuming services* to fulfill a purpose. The services architecture defines the requirements for the types of participants and services that fulfill those roles.


«servicesArchitecture»
Image Management
requesting System
+image Requestor
«collaborationUse»
:Imaging Reporting
+ImageOrder
«collaborationUse»
:Order an Image
Patient
+image Reporter
+subject
«collaborationUse»
:image Fulfillment
+image Storer
Imaging Persistence System
«collaborationUse»
Imaging
+document M
+image Reporter
+orderManager
+image Taker
Ordering System
+image Taker
Promising System
+assessment Requester
+orderManager
+orderPromiser
«collaborationUse»
:study Assessment
«collaborationUse»
:Image Assessment Request
+study Assessor
+study Assessor
«collaborationUse»
:Image Promise Fulfillment
Radiologist System

Example Enterprise Level SOA
Claims Processing Services Architecture

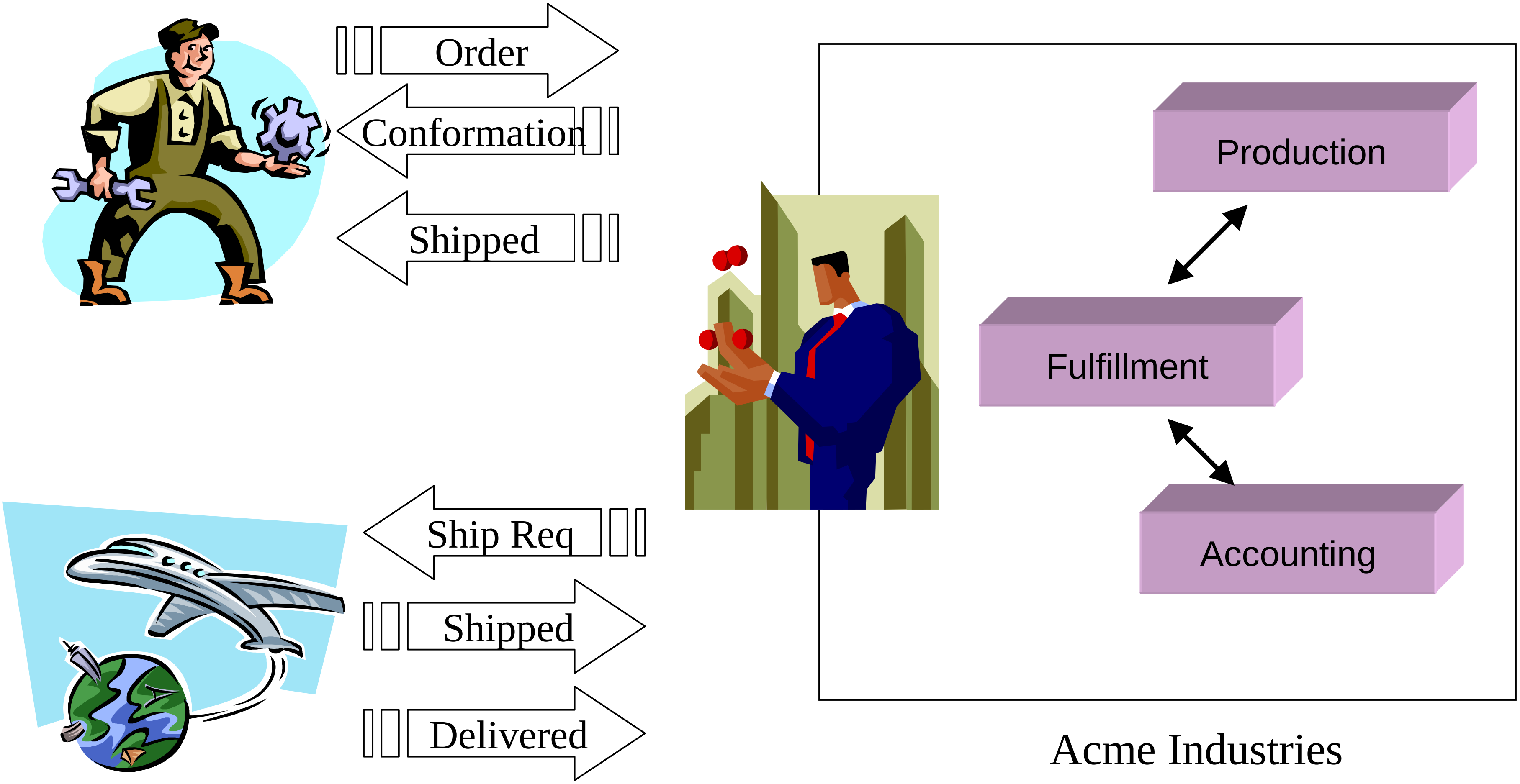
A *services architecture* describes how *participants* work together for a purpose by providing and using services expressed as *service contracts*. It is modeled as a UML *collaboration.*

A *participant* represents some party that provides and/or consumes services. Participants may represent people, organizations or systems.



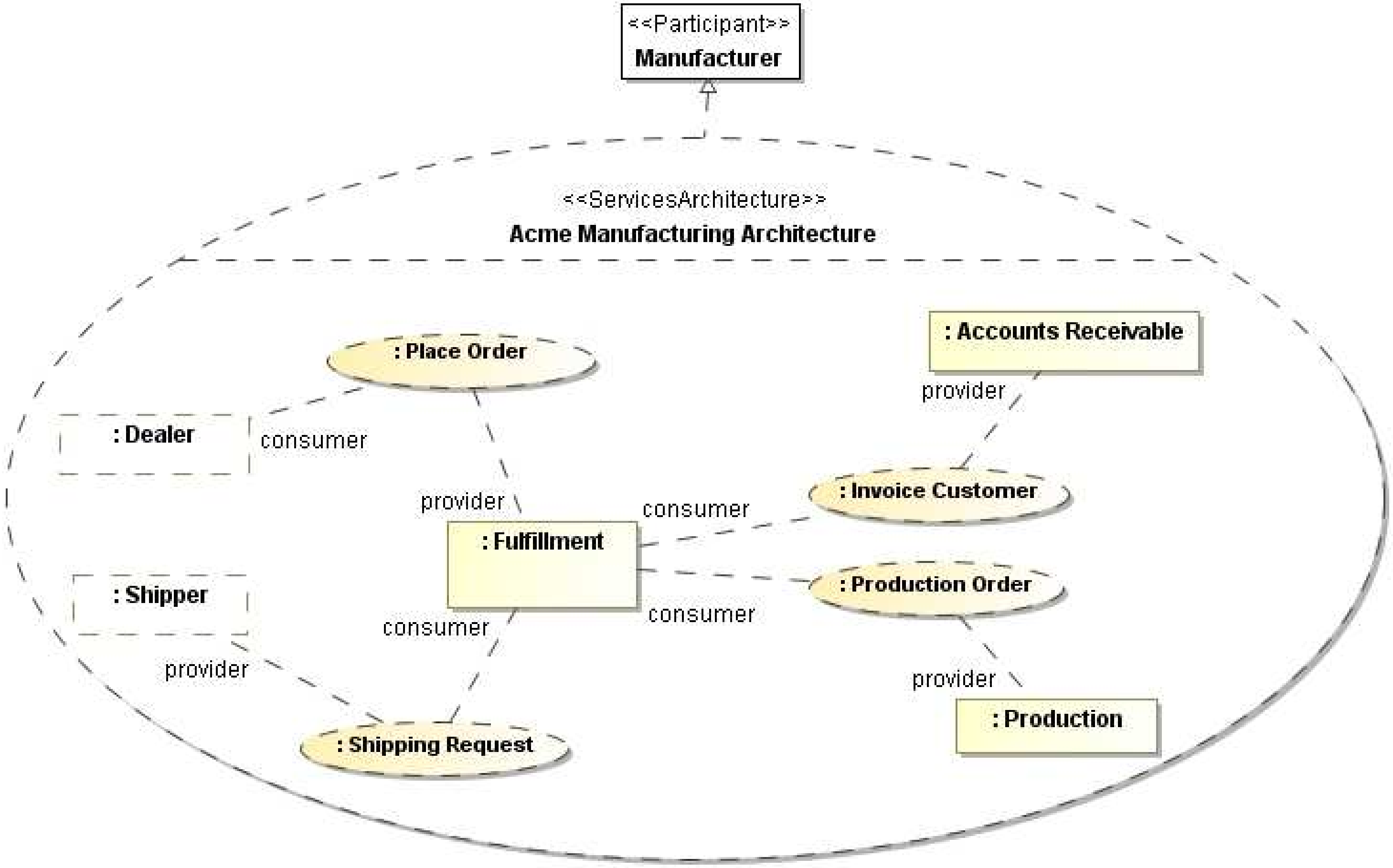
A *service contract* is the specification of the agreement between providers and consumers of a *service* as to what information, products, assets, value and obligations will flow between them. It specifies the service without regard for realization, capabilities or implementation.

Drilling down - Inside a Manufacturer



Not every manufacturer is going to be the same inside – this shows some of the internals of “Acme”

Architecture Inside of Acme

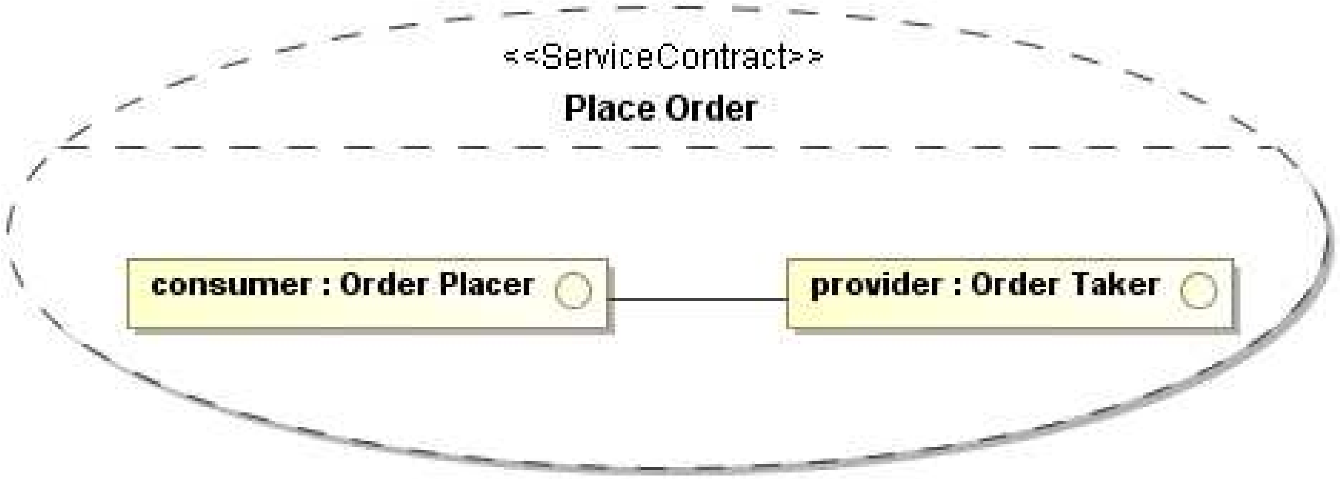


SOA architectures are able to “drill down” in more detail – this shows the architecture inside of a particular manufacturer, Acme. Other manufactures may have different internal architectures and processes.

Specifying Services

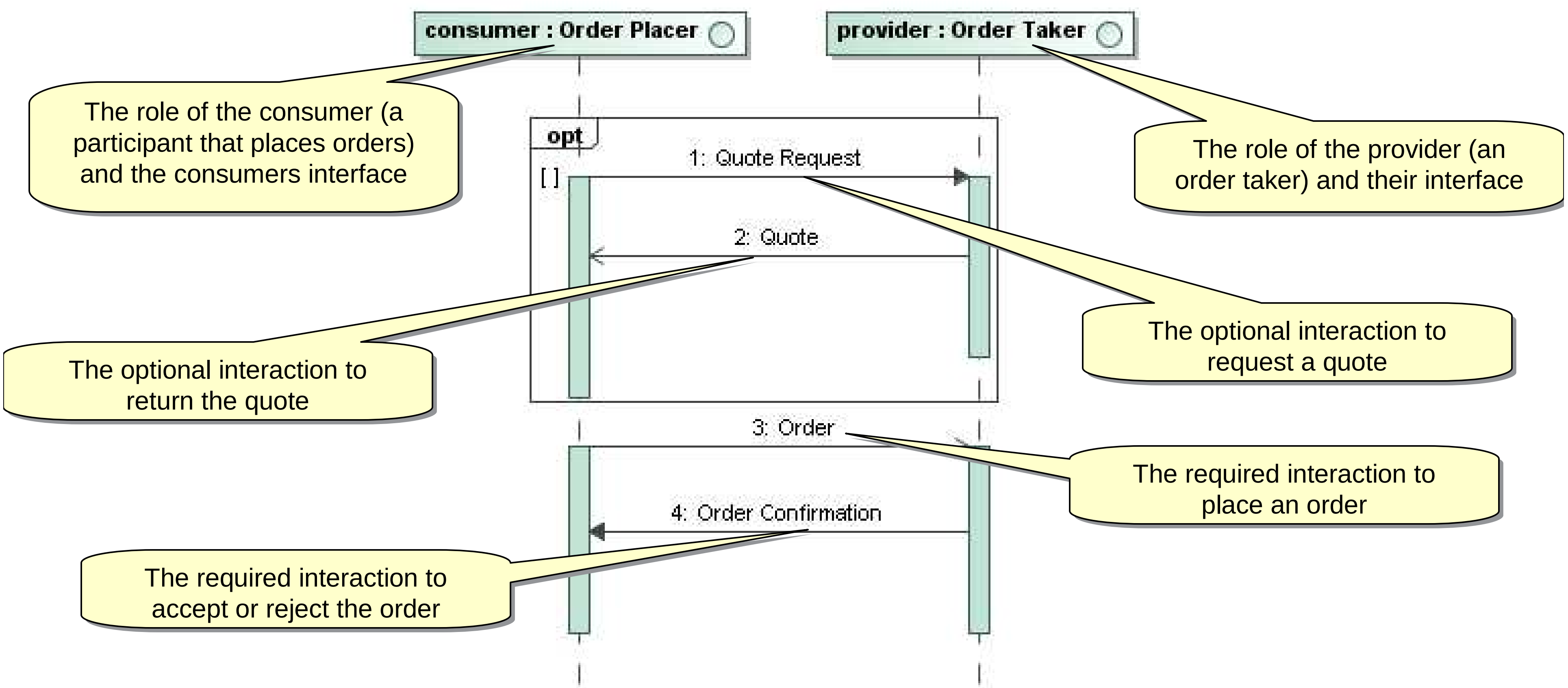
- Specification of services includes
  - The roles each participant plays in the service, such as provider and consumer
  - The message types that go between the participants when the service is enacted
  - The interfaces provided and used by each participant for the service
  - The choreography of the interactions between the participants while enacting the service
  - Placeholders are provided for service policies and motivation
- Modeling services
  - Services are modeled using “Service Contracts” and “Service Interfaces” in SoaML. These use UML interfaces, classes and behaviors.

High level view of a service



This view of a service only identifies the service name and the roles each participant plays in the service. This is a high-level summary view.

Service Choreography for “Place Order”



The role of the consumer (a participant that places orders) and the consumers interface

The role of the provider (an order taker) and their interface

The optional interaction to return the quote

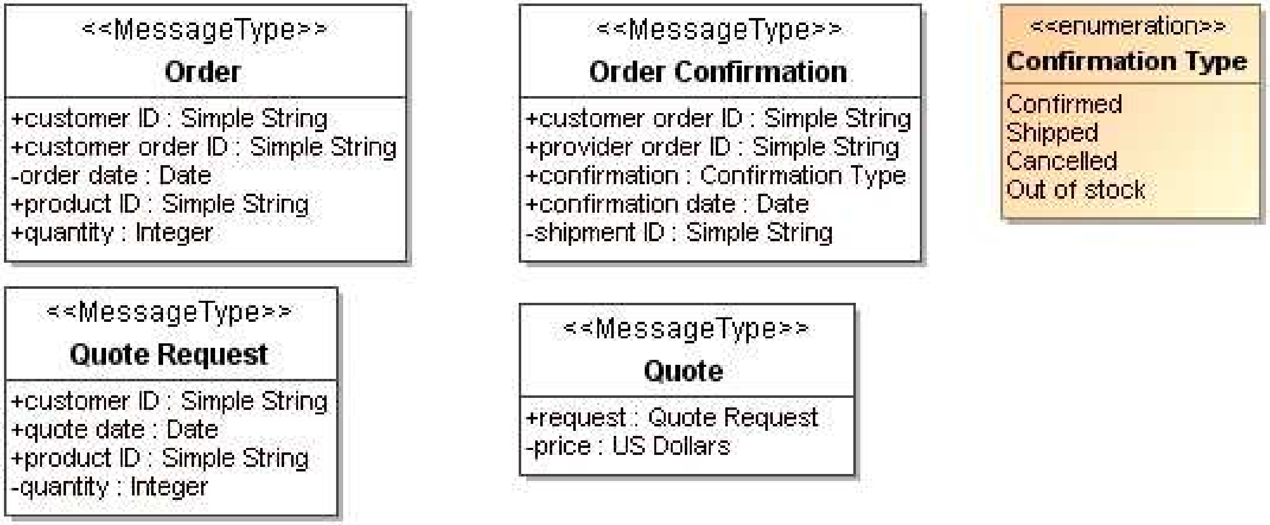
The optional interaction to request a quote

The required interaction to place an order

The required interaction to accept or reject the order

A more detailed look at the same service. Note that this models a fully asynchronous SOA – like most business interactions, the document message types are detailed on the next page.

Message Detail for Place Order



<<MessageType>>
Order

+customer ID : Simple String
+customer order ID : Simple String
-order date : Date
+product ID : Simple String
+quantity : Integer

<<MessageType>>
Order Confirmation

+customer order ID : Simple String
+provider order ID : Simple String
+confirmation : Confirmation Type
+confirmation date : Date
-shipment ID : Simple String

<<enumeration>>
Confirmation Type

Confirmed
Shipped
Cancelled
Out of stock

<<MessageType>>
Quote Request

+customer ID : Simple String
+quote date : Date
+product ID : Simple String
-quantity : Integer

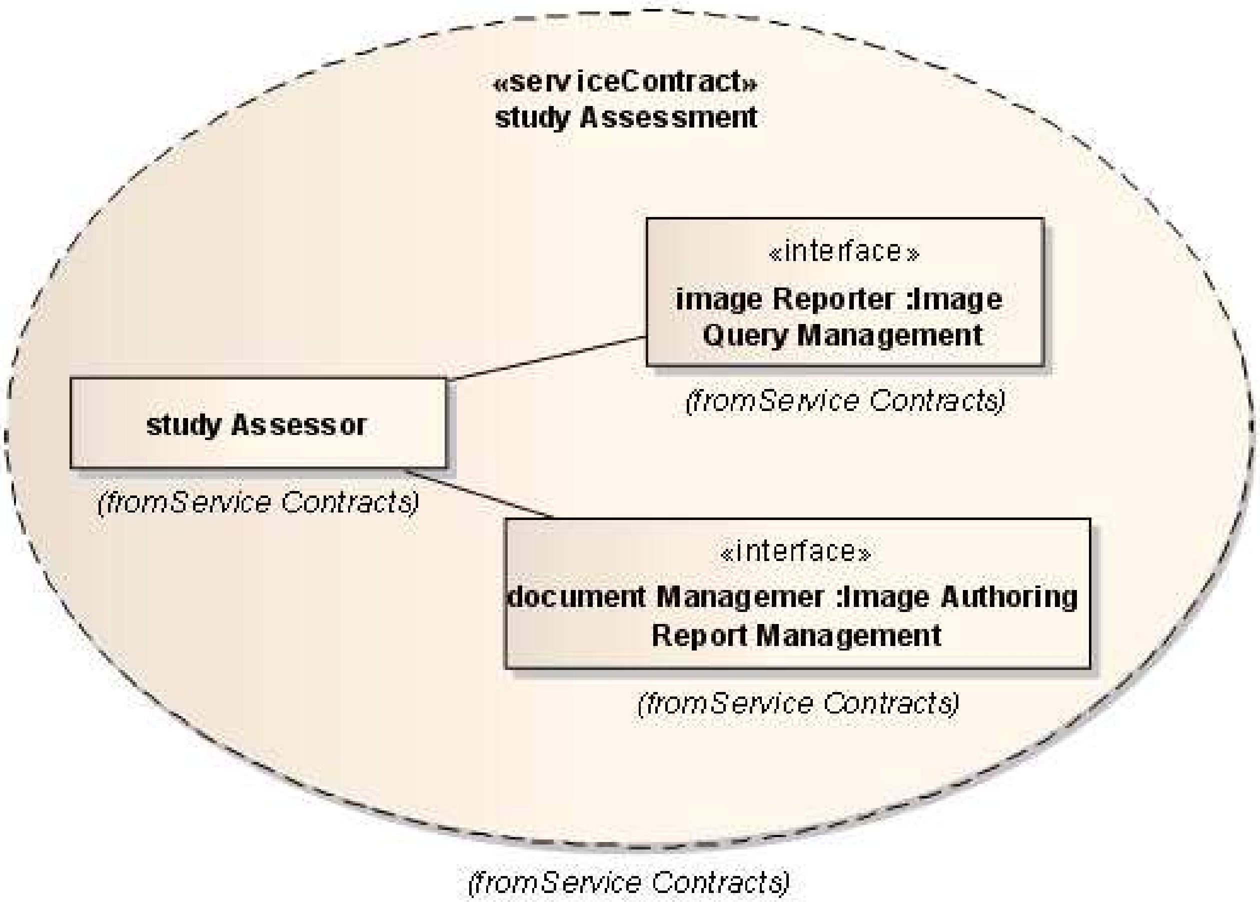
<<MessageType>>
Quote

+request : Quote Request
-price : US Dollars

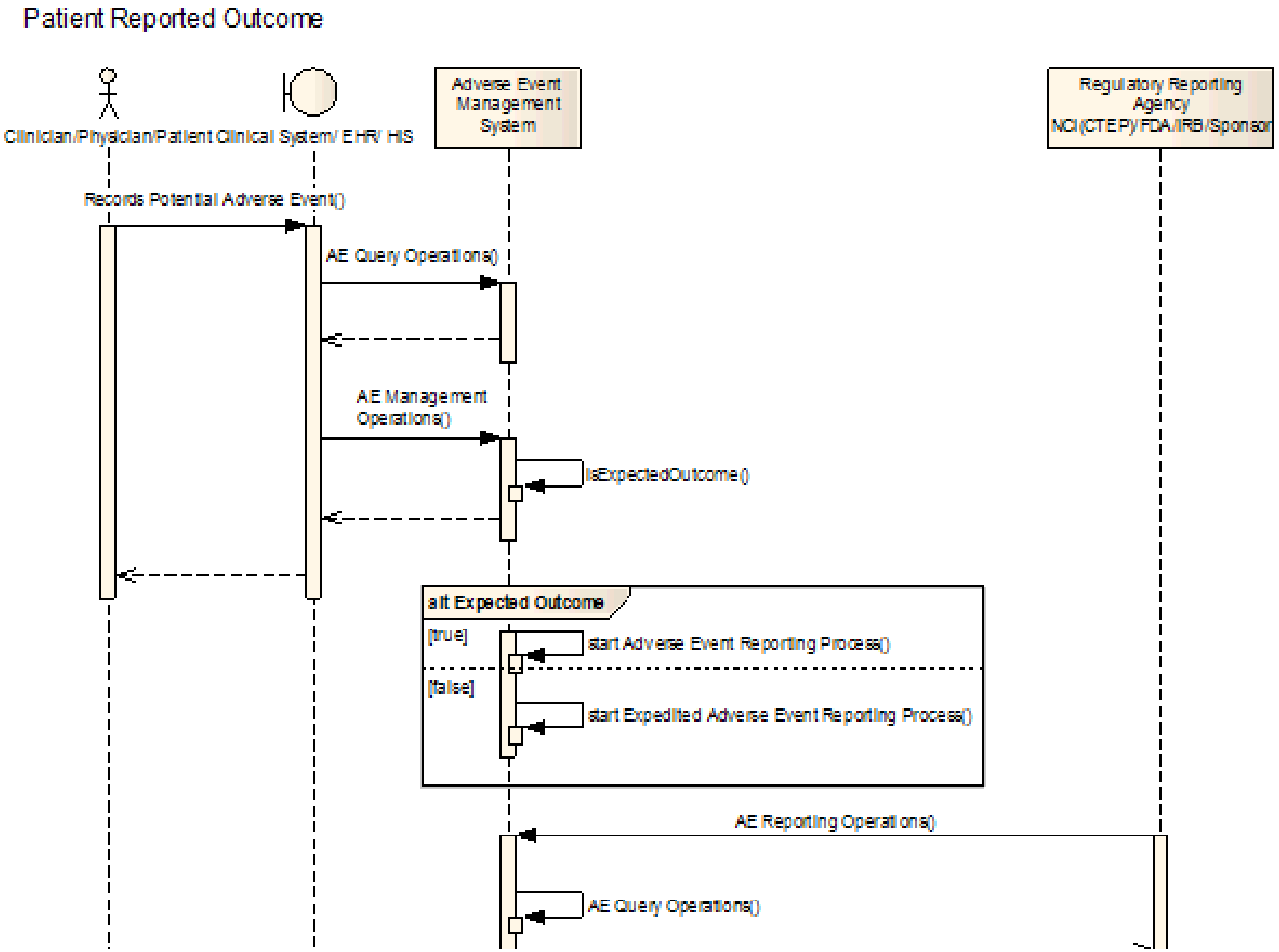
This is the detail for the message types that correspond to the interactions for the place order service.

Note that at the technology level this can produce XML schema for the messages.

Services can be multi-party

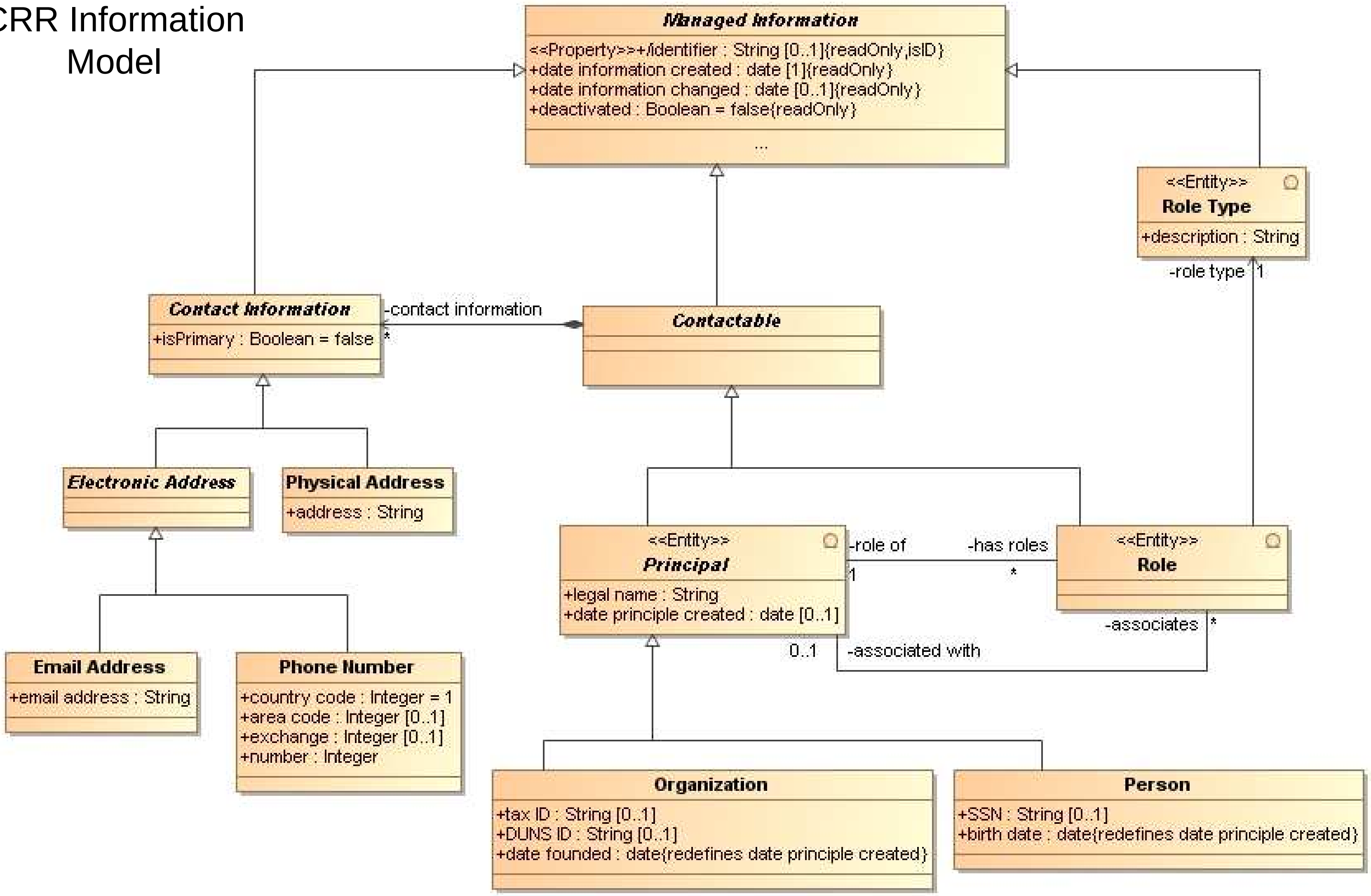


Service Interaction Detail


Patient Reported Outcome
Clinician/Physician/Patient
Clinical System/ EHR/ HIS
Adverse Event Management System
Regulatory Reporting Agency NCI(CTEP)/FDA/IRB/Sponsor
Records Potential Adverse Event()
AE Query Operations()
AE Management Operations()
IsExpectedOutcome()
alt Expected Outcome
[true]
start Adverse Event Reporting Process()
[false]
start Expedited Adverse Event Reporting Process()
AE Reporting Operations()
AE Query Operations()

Example Information Model

CRR Information Model



Managed Information
<<Property>>+/identifier : String [0..1]{readOnly,isID}
+date information created : date [1]{readOnly}
+date information changed : date [0..1]{readOnly}
+deactivated : Boolean = false{readOnly}
...

<<Entity>>
Role Type
+description : String

-role type 1

Contact Information
+isPrimary : Boolean = false

-contact information *

Contactable

Electronic Address

Physical Address
+address : String

Email Address
+email address : String

Phone Number
+country code : Integer = 1
+area code : Integer [0..1]
+exchange : Integer [0..1]
+number : Integer

<<Entity>>
Principal
+legal name : String
+date principle created : date [0..1]

-role of 1

-has roles *

<<Entity>>
Role

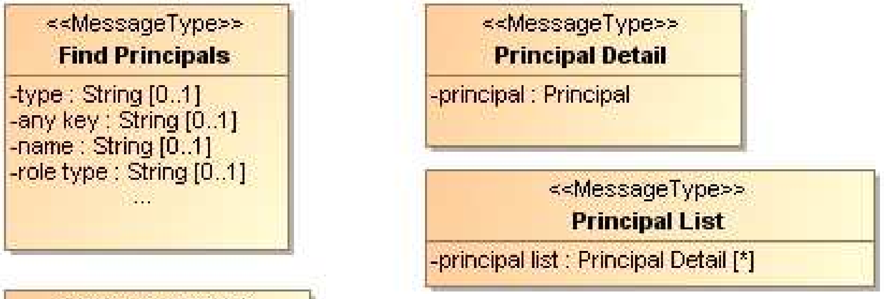
-associates *

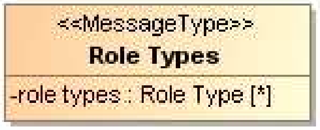
0..1 -associated with

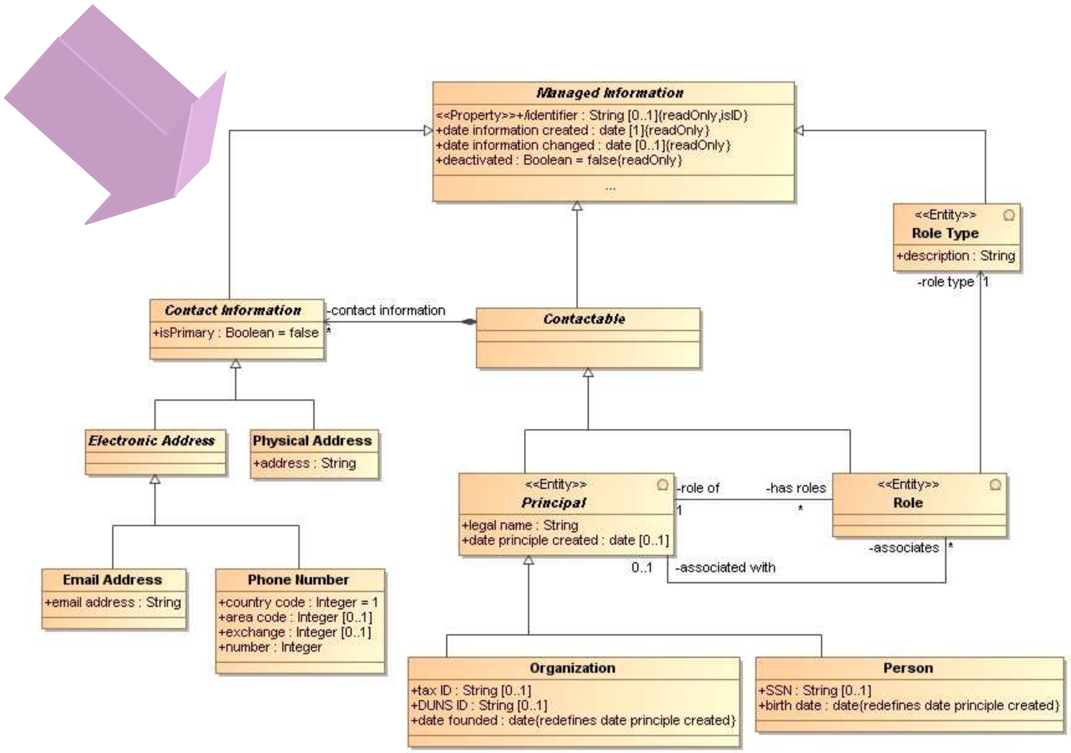
Organization
+tax ID : String [0..1]
+DUNS ID : String [0..1]
+date founded : date{redefines date principle created}

Person
+SSN : String [0..1]
+birth date : date{redefines date principle created}

Linking messages to business information



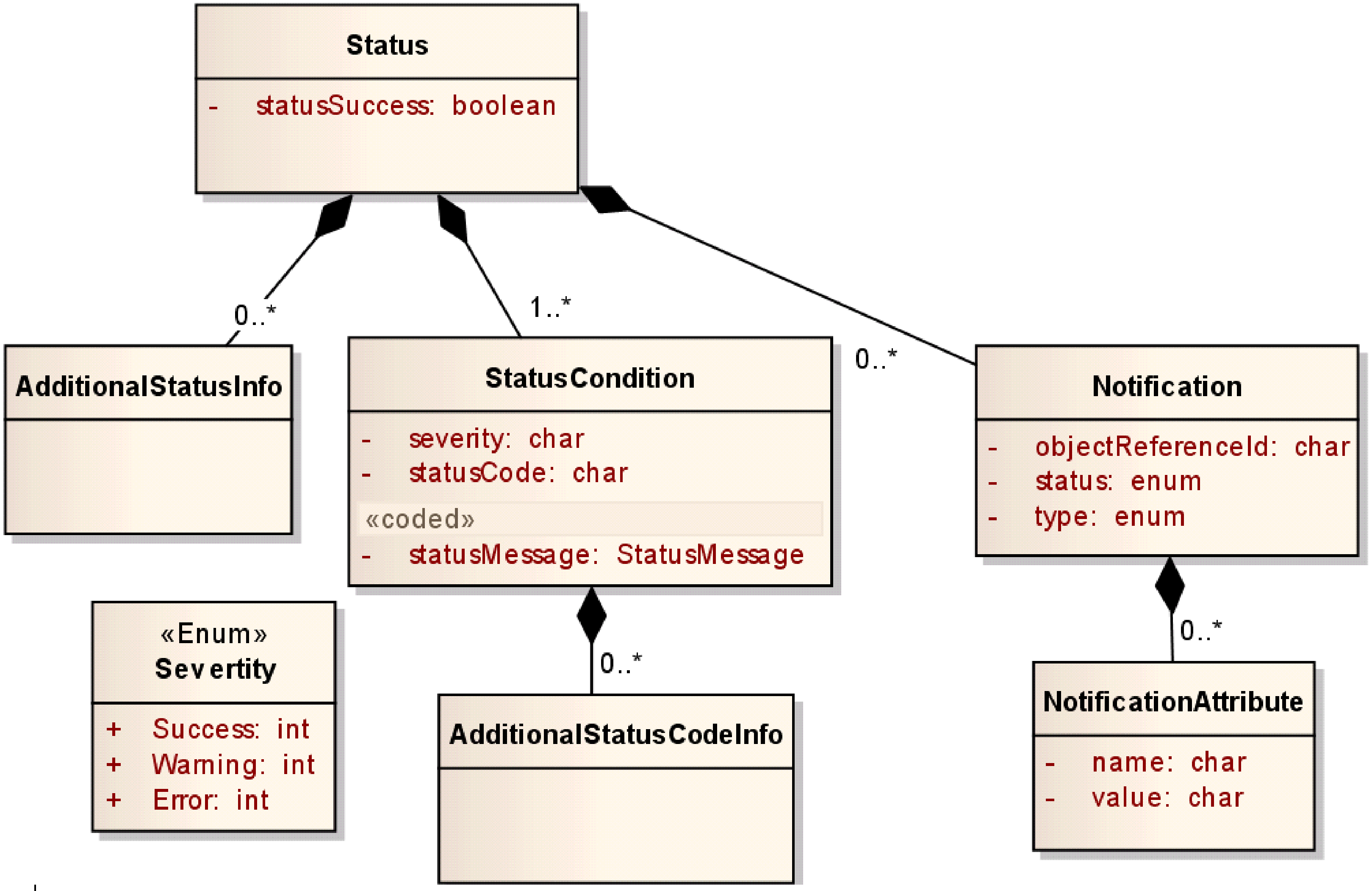




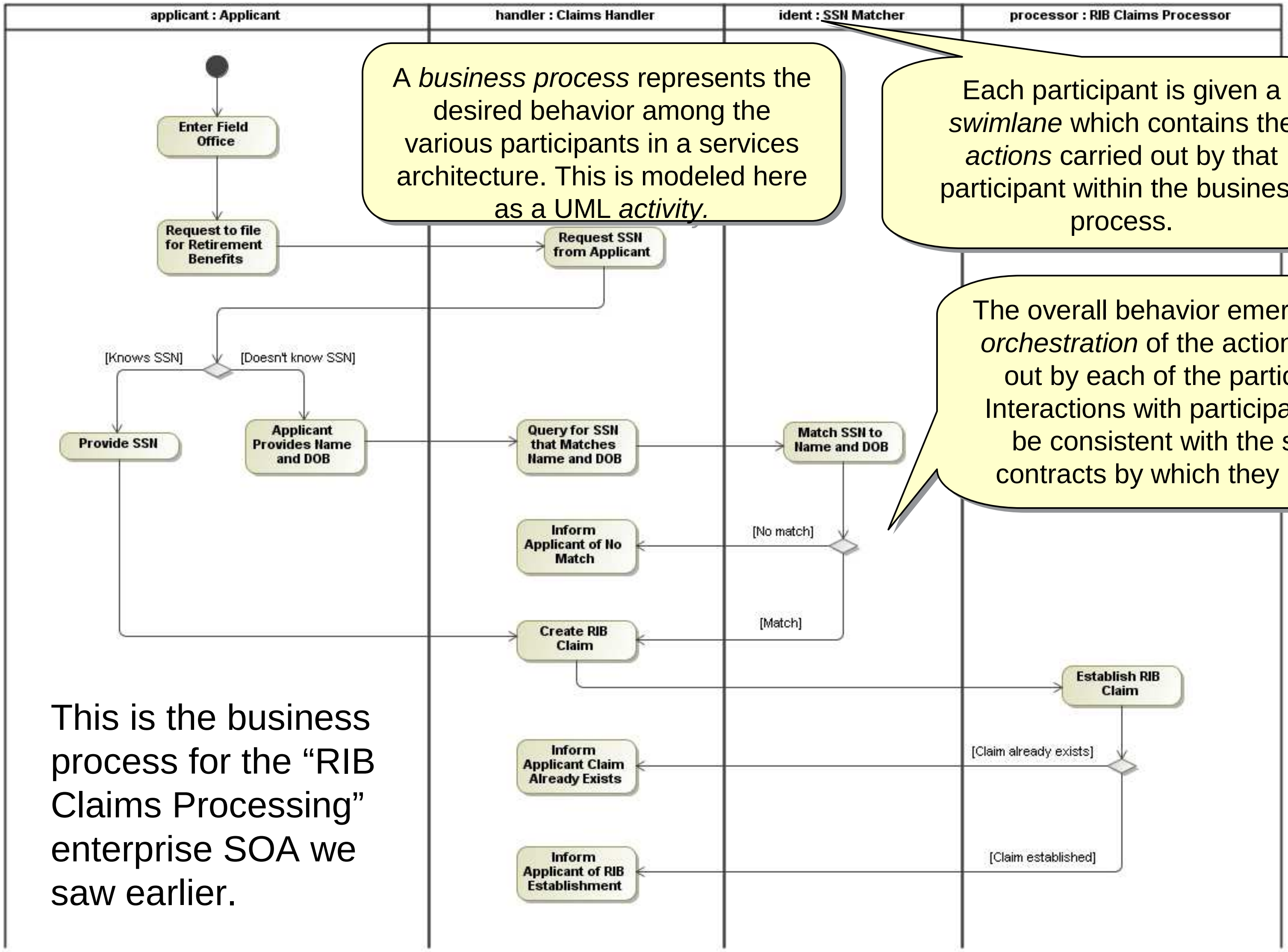
Managed Information
<<Property>>+/identifier : String [0..1]{readOnly,isID}
+date information created : date [1]{readOnly}
+date information changed : date [0..1]{readOnly}
+deactivated : Boolean = false{readOnly}
...
<<Entity>>
Role Type
+description : String
-role type
1
Contact Information
+isPrimary : Boolean = false
-contact information
*
Contactable
Electronic Address
Physical Address
+address : String
<<Entity>>
Principal
+legal name : String
+date principle created : date [0..1]
-role of
1
-has roles
*
<<Entity>>
Role
-associates
*
0..1
-associated with
Email Address
+email address : String
Phone Number
+country code : Integer = 1
+area code : Integer [0..1]
+exchange : Integer [0..1]
+number : Integer
Organization
+tax ID : String [0..1]
+DUNS ID : String [0..1]
+date founded : date{redefines date principle created}
Person
+SSN : String [0..1]
+birth date : date{redefines date principle created}

SOA Messages can reference and include parts of the logical information model – forming a connection between SOA and enterprise data

Example Information Model



Linking the Business Process


applicant : Applicant
handler : Claims Handler
ident : SSN Matcher
processor : RIB Claims Processor
Enter Field Office
Request to file for Retirement Benefits
Request SSN from Applicant
[Knows SSN]
[Doesn't know SSN]
Provide SSN
Applicant Provides Name and DOB
Query for SSN that Matches Name and DOB
Match SSN to Name and DOB
[No match]
Inform Applicant of No Match
[Match]
Create RIB Claim
Establish RIB Claim
[Claim already exists]
Inform Applicant Claim Already Exists
[Claim established]
Inform Applicant of RIB Establishment

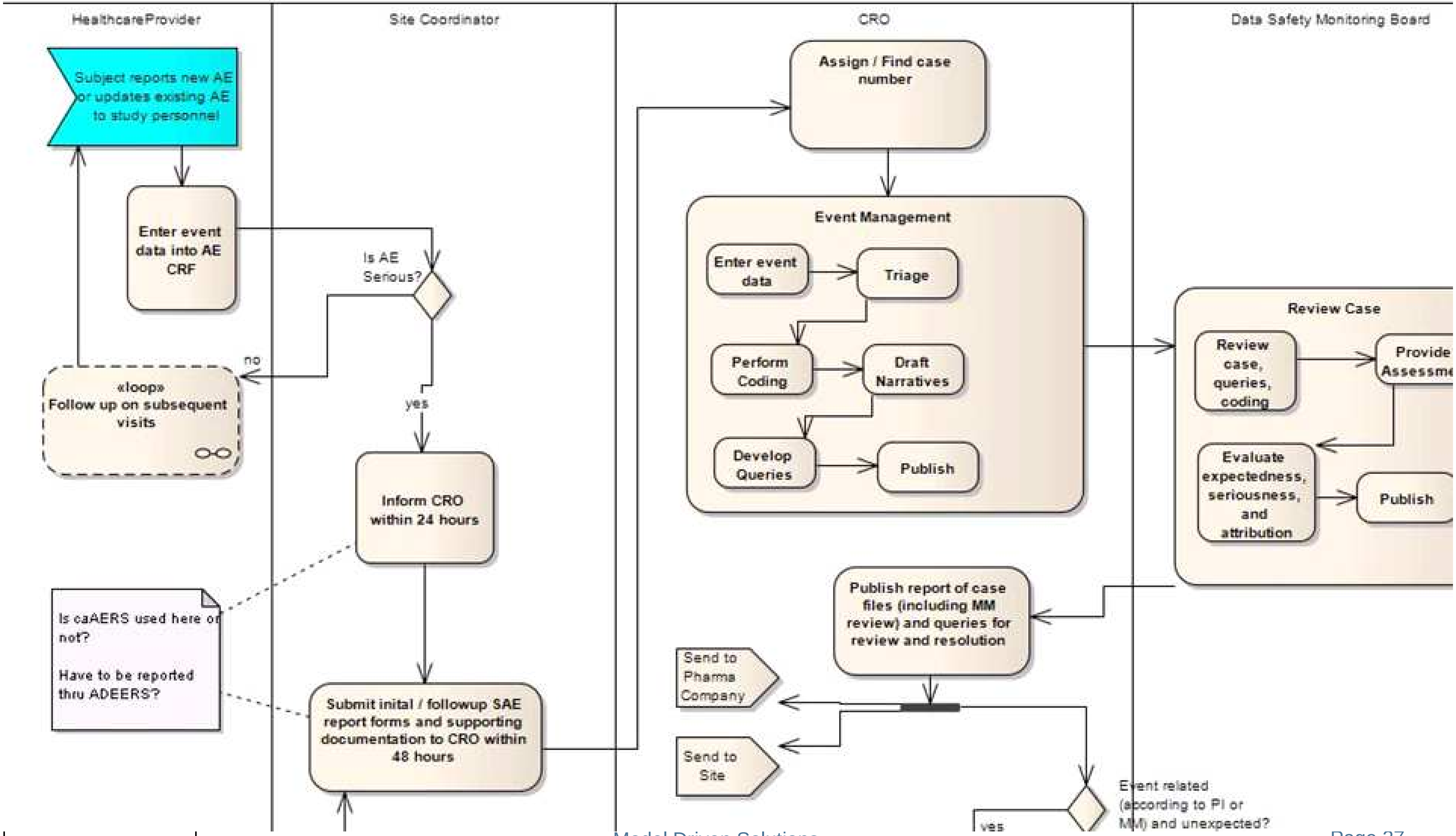
A *business process* represents the desired behavior among the various participants in a services architecture. This is modeled here as a UML *activity.*

Each participant is given a *swimlane* which contains the *actions* carried out by that participant within the business process.

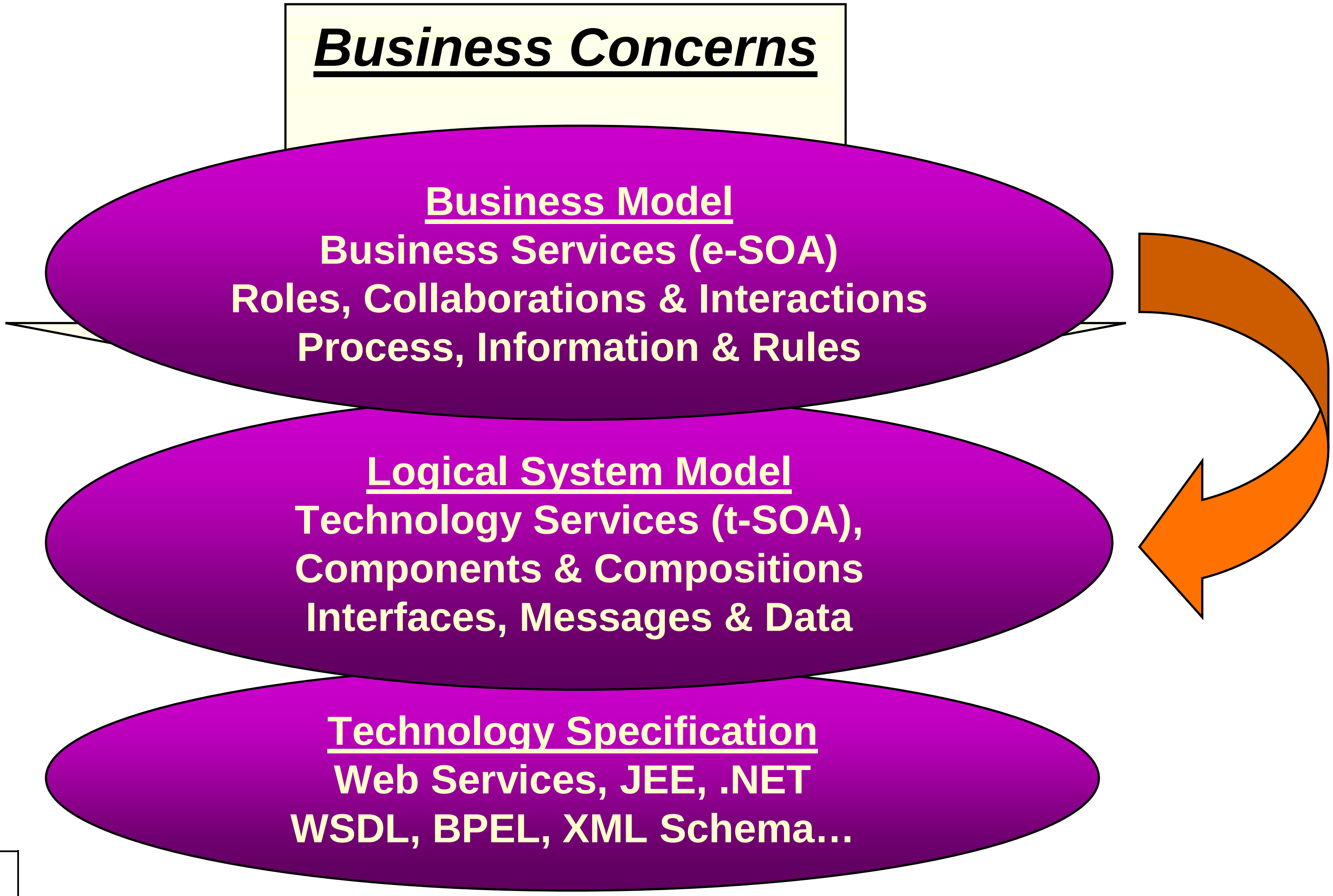
The overall behavior emerges as an *orchestration* of the actions carried out by each of the participants. Interactions with participants must be consistent with the service contracts by which they interact.

This is the business process for the “RIB Claims Processing” enterprise SOA we saw earlier.

Example Healthcare Process


HealthcareProvider
Site Coordinator
CRO
Data Safety Monitoring Board
Subject reports new AE or updates existing AE to study personnel
Enter event data into AE CRF
«loop» Follow up on subsequent visits
no
Is AE Serious?
yes
Inform CRO within 24 hours
Is caAERS used here or not?
Have to be reported thru ADEERS?
Submit inital / followup SAE report forms and supporting documentation to CRO within 48 hours
Assign / Find case number
Event Management
Enter event data
Triage
Perform Coding
Draft Narratives
Develop Queries
Publish
Review Case
Review case, queries, coding
Provide Assessme
Evaluate expectedness, seriousness, and attribution
Publish
Publish report of case files (including MM review) and queries for review and resolution
Send to Pharma Company
Send to Site
Event related (according to PI or MM) and unexpected?
yes

Producing the logical systems model



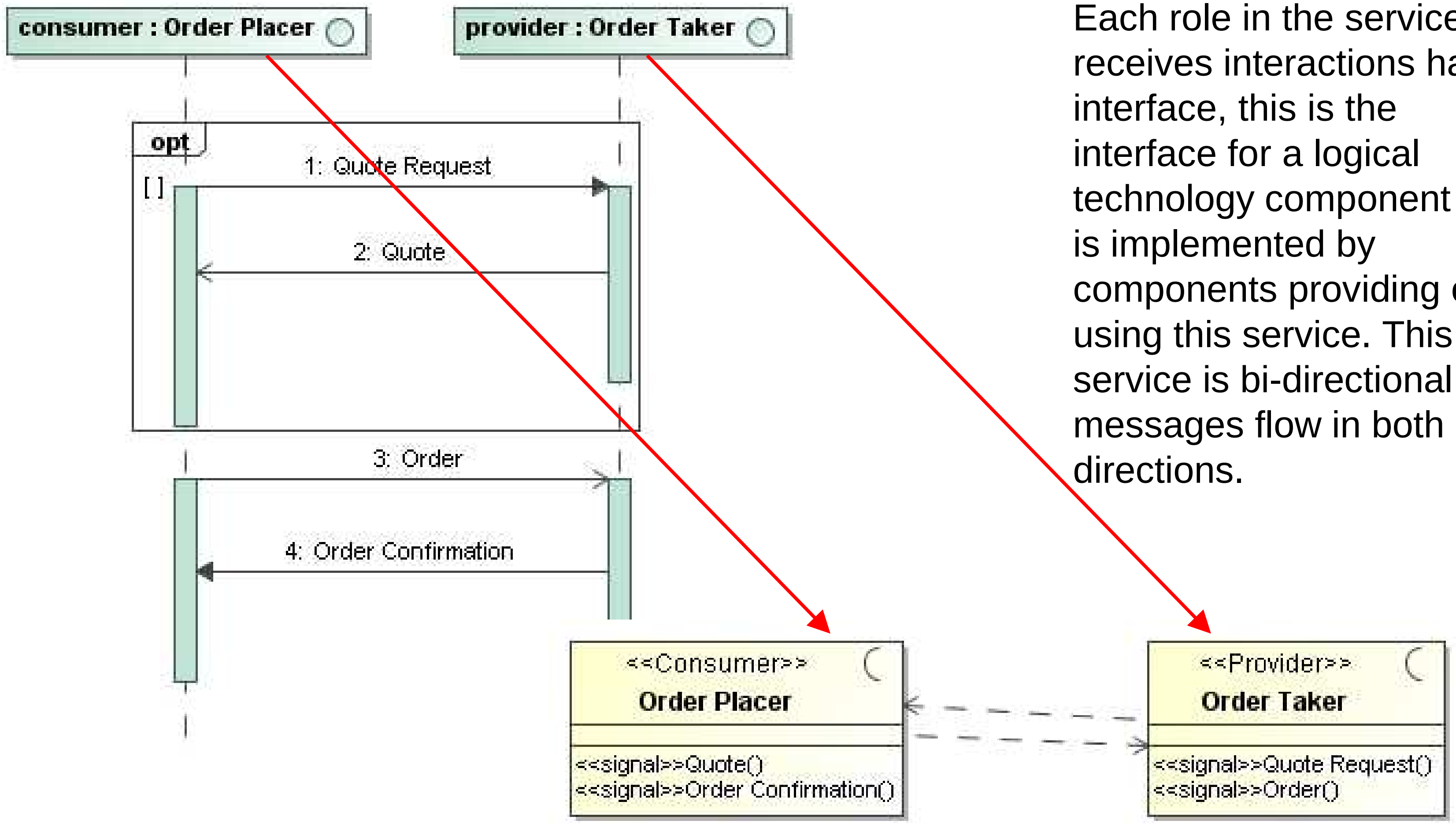
Business Concerns

Business Model
Business Services (e-SOA)
Roles, Collaborations & Interactions
Process, Information & Rules

Logical System Model
Technology Services (t-SOA),
Components & Compositions
Interfaces, Messages & Data

Technology Specification
Web Services, JEE, .NET
WSDL, BPEL, XML Schema…

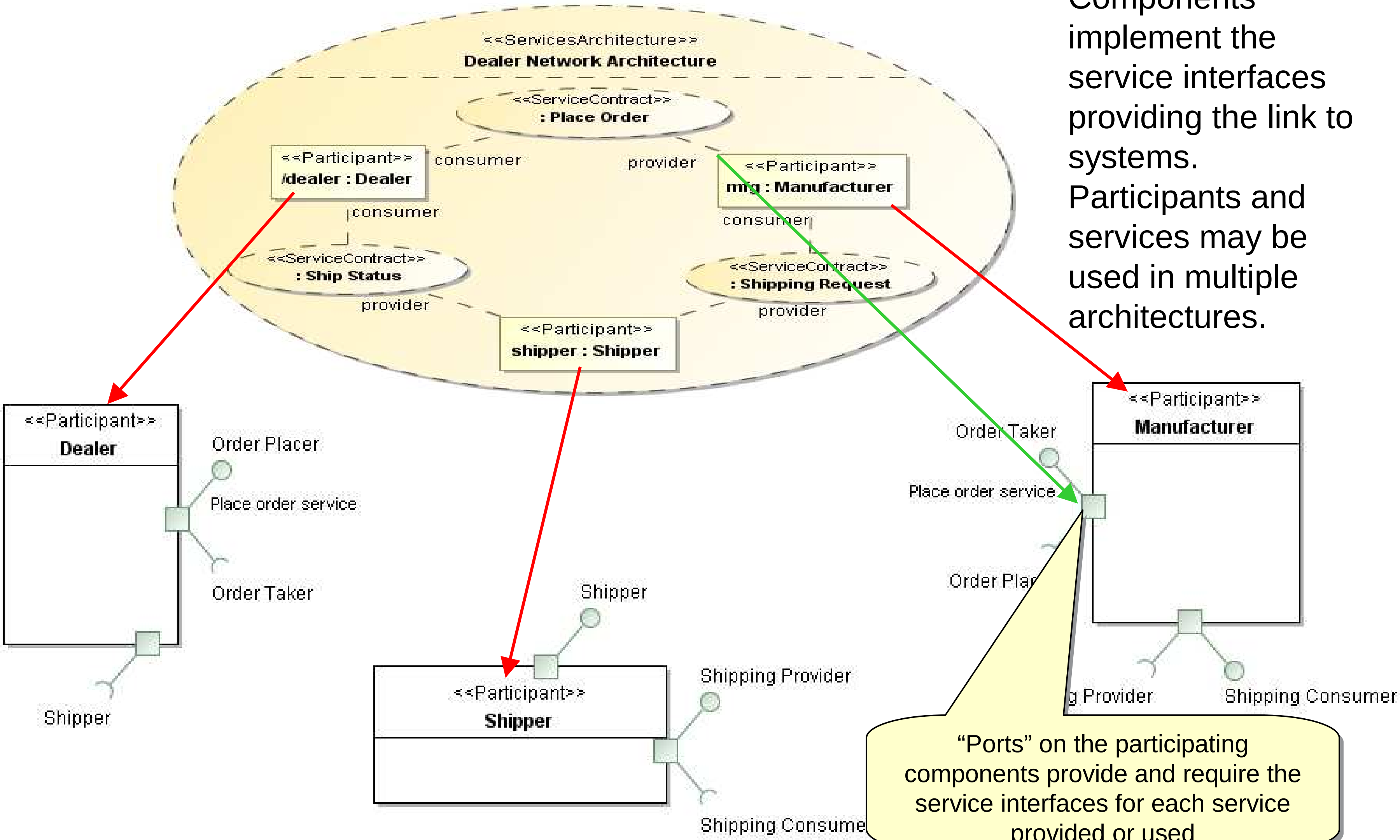
Interfaces for Participants



Each role in the service that receives interactions has an interface, this is the interface for a logical technology component and is implemented by components providing or using this service. This service is bi-directional - messages flow in both directions.

Interfaces will correspond with parts of WSDL in a web services mapping of SoaML

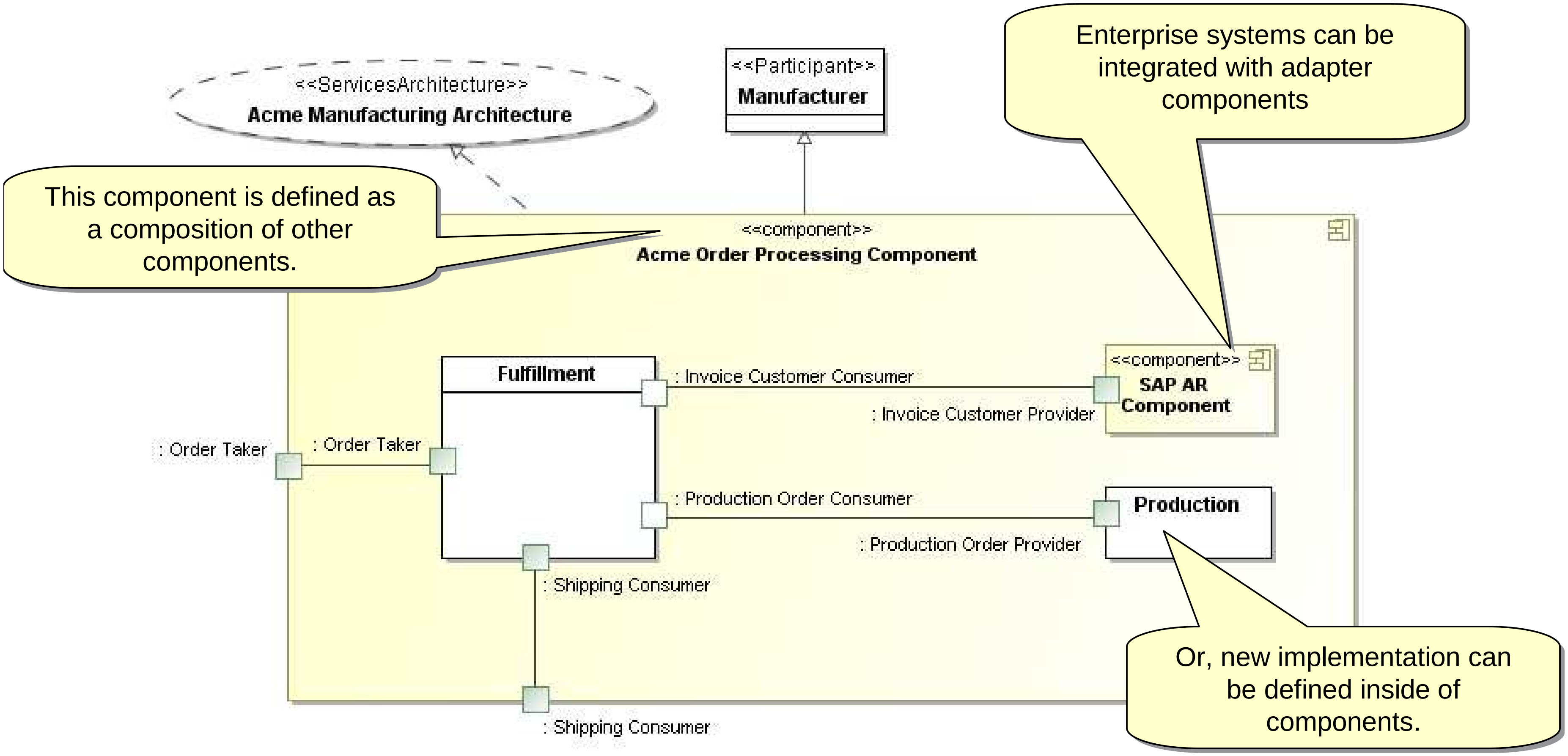
Logical System Components



“Ports” on the participating components provide and require the service interfaces for each service provided or used

Components implement the service interfaces providing the link to systems. Participants and services may be used in multiple architectures.

Composite Application Components



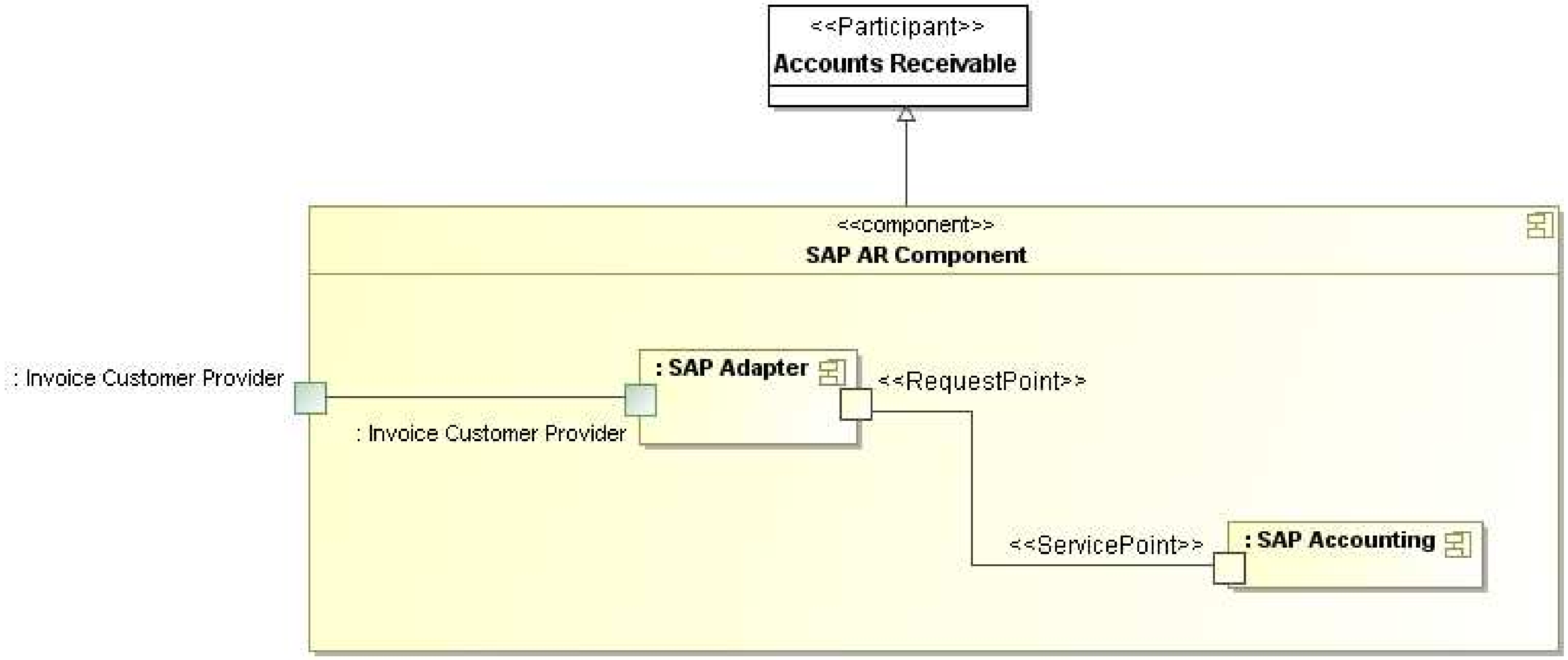
This component is defined as a composition of other components.

Enterprise systems can be integrated with adapter components

Or, new implementation can be defined inside of components.

Components can be assembled from other components by linking their services. This corresponds to the architecture for Acme.

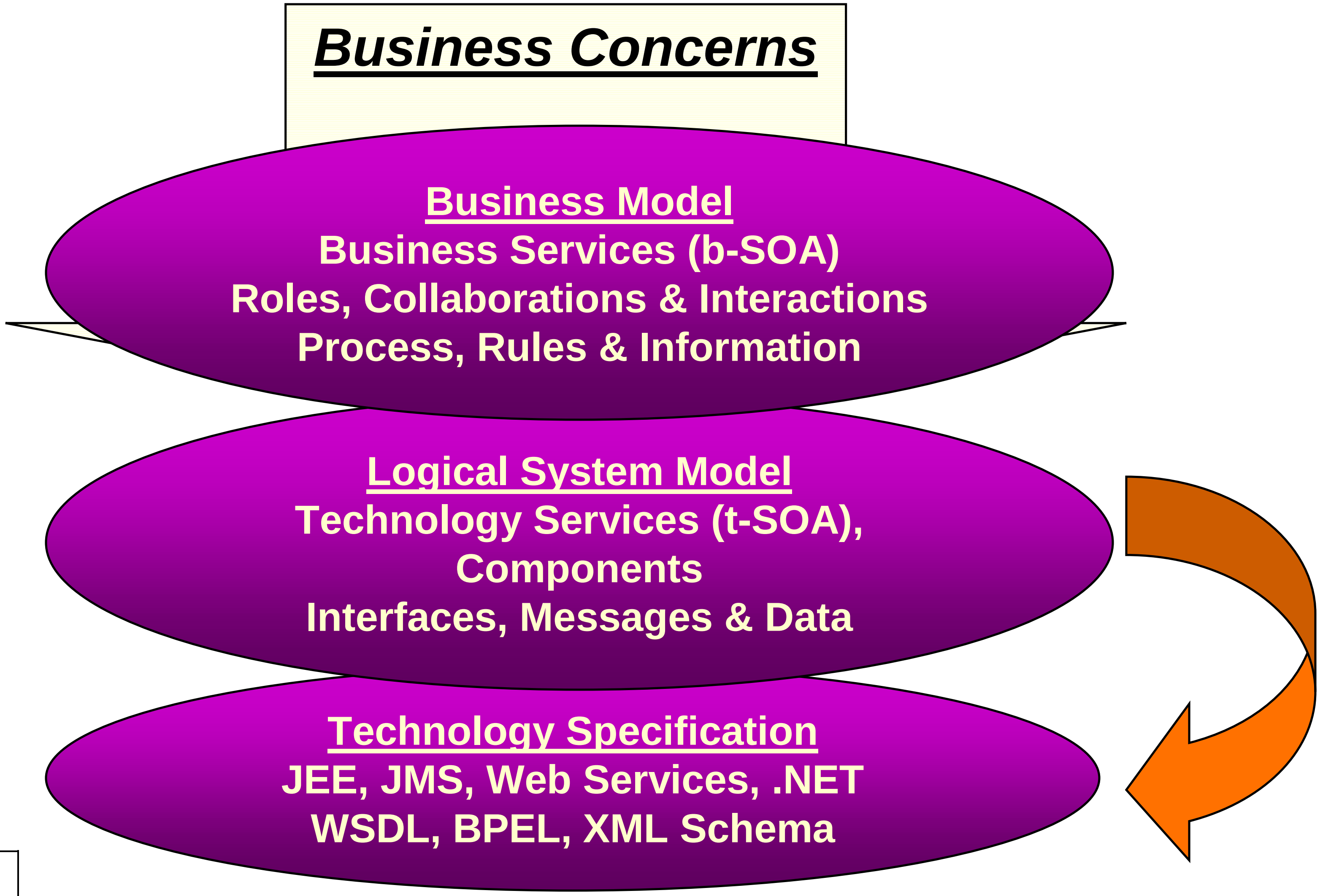
Adapting Enterprise Systems


<<Participant>>
Accounts Receivable
<<component>>
SAP AR Component
: Invoice Customer Provider
: SAP Adapter
<<RequestPoint>>
: Invoice Customer Provider
<<ServicePoint>>
: SAP Accounting

This is the inside of the SAP AR component – also a composition, it uses the existing SAP interfaces and adapts them to the enterprise contract.

This separates the concerns of a particular enterprise system from the enterprise SOA. Sometimes the system interfaces are used directly or adapted by an ESB.

Provisioning Technology Artifacts



Business Concerns

Business Model

Business Services (b-SOA)

Roles, Collaborations & Interactions

Process, Rules & Information

Logical System Model

Technology Services (t-SOA), Components

Interfaces, Messages & Data

Technology Specification

JEE, JMS, Web Services, .NET

WSDL, BPEL, XML Schema

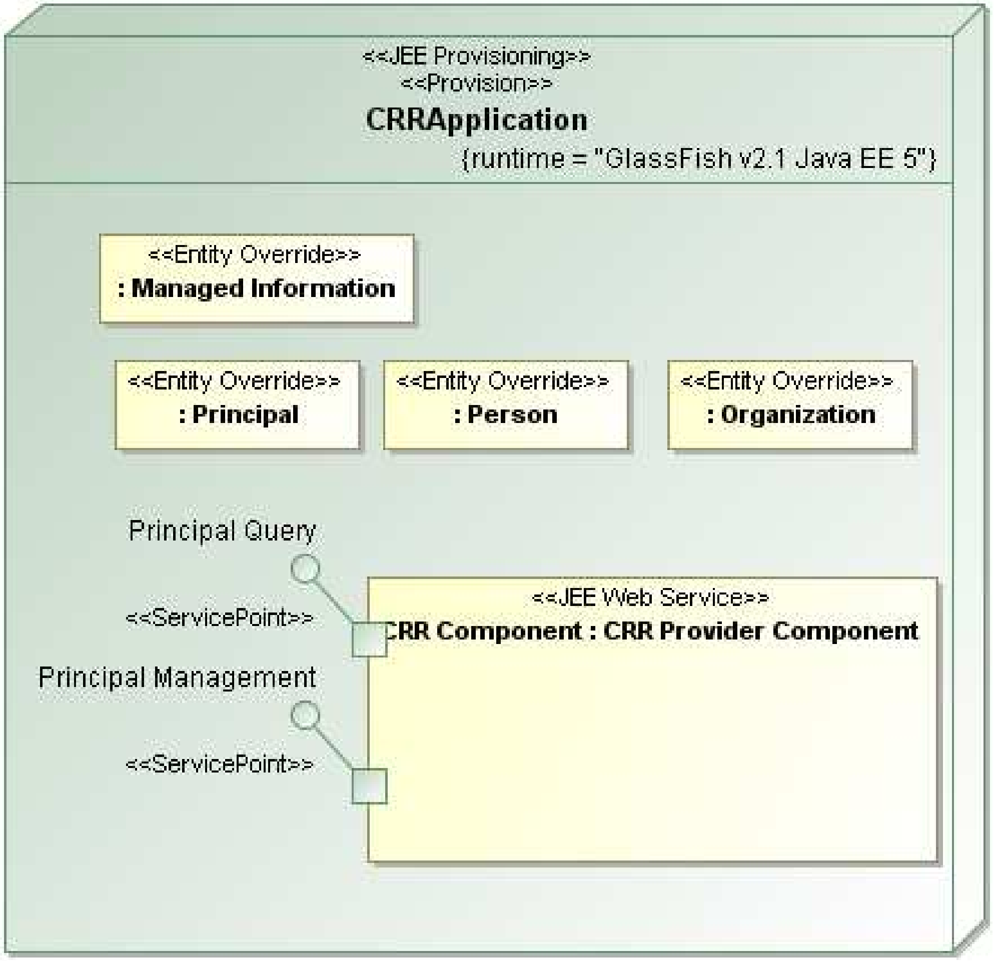
Realizing the Model

- How to we use I.T. to realize our processes and services?
  - Direct execution frameworks
    - The "no code" approach where the process and services execute directly from the model
    - May use other standards, such as BPEL
  - Wrapping and adapting existing capabilities
    - Automatic or manual creation of "adapter components" that use legacy systems, information or services to create the architected enterprise services
  - Creation of new application components and services
    - Build new capabilities by creating new components and creating composite applications
    - May be visual and declarative or code oriented
- Under the SoaML framework, all of these options can co-exist as a system of systems linked by services

Ok, but sometimes I write code

- Developers want to know how SoaML relates to the development environments they understand
- The following shows how the SoaML architecture can automate this development process

Example Provisioning to JEE Web Services

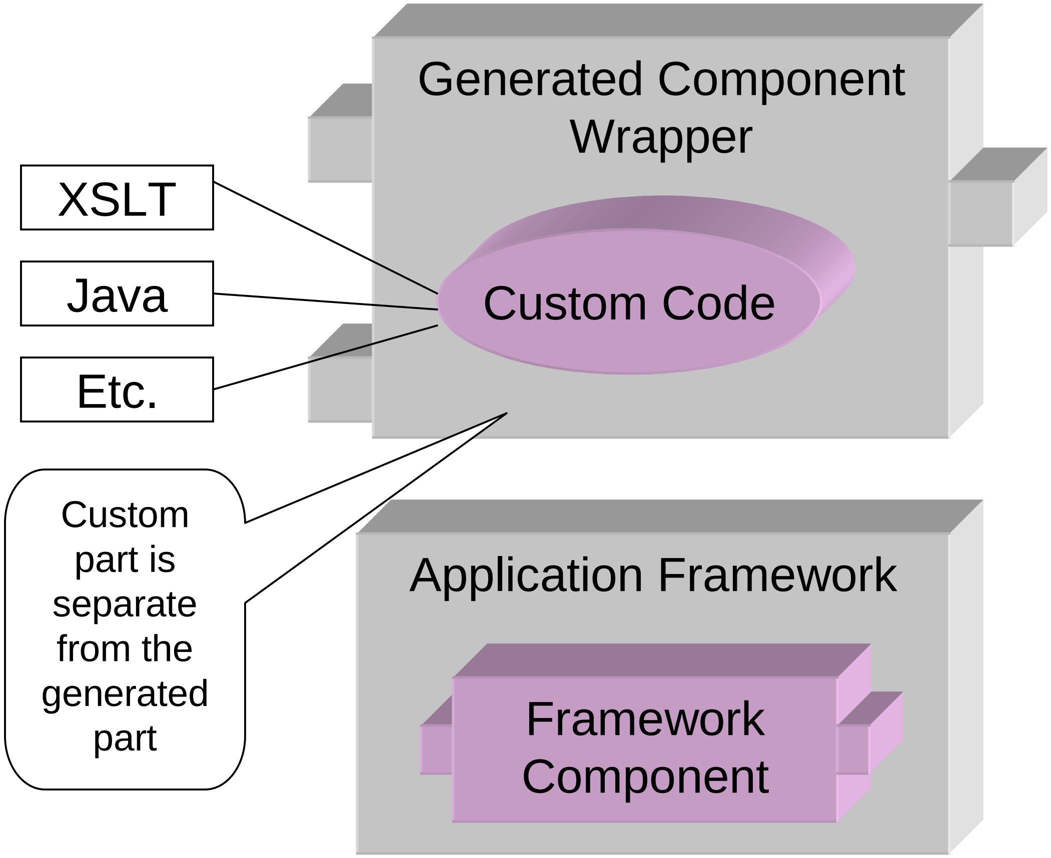


How SoaML is provisioned to technology artifacts, such as Web Services, is not yet standard.

This illustrates one approach.

The components to be provisioned are dropped into a technology specific provisioning node (in this case JEE and web services).

Custom Business Logic Components



Generated Component Wrapper

Custom Code

XSLT

Java

Etc.

Custom part is separate from the generated part

Application Framework

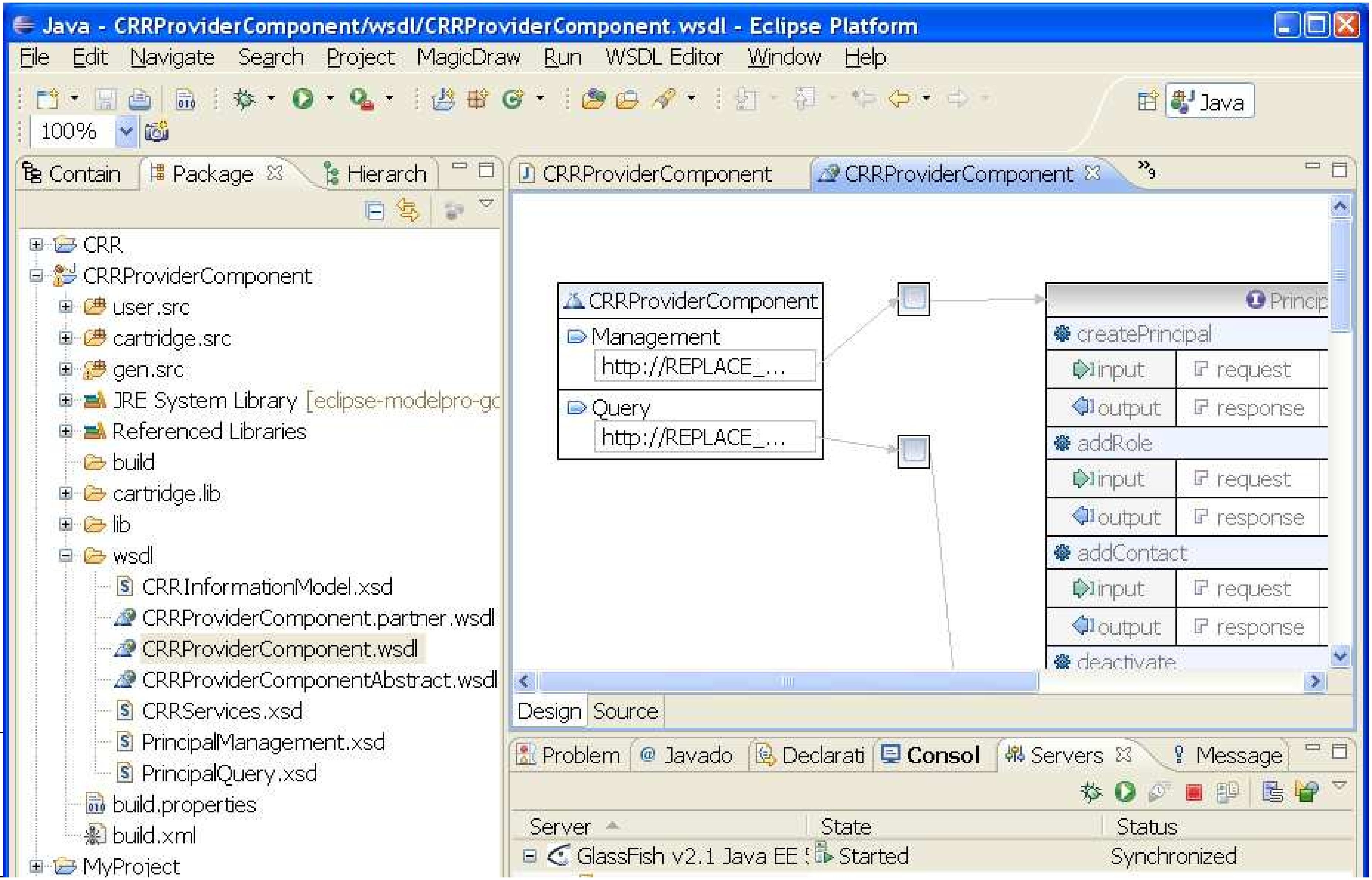
Framework Component

Application components provide service implementations with user supplied logic. These “plug into” the users architecture as composite application components

Framework components add infrastructural capabilities by extending the platform (E.G. JBI) and are called by the provisioned code or platform configuration

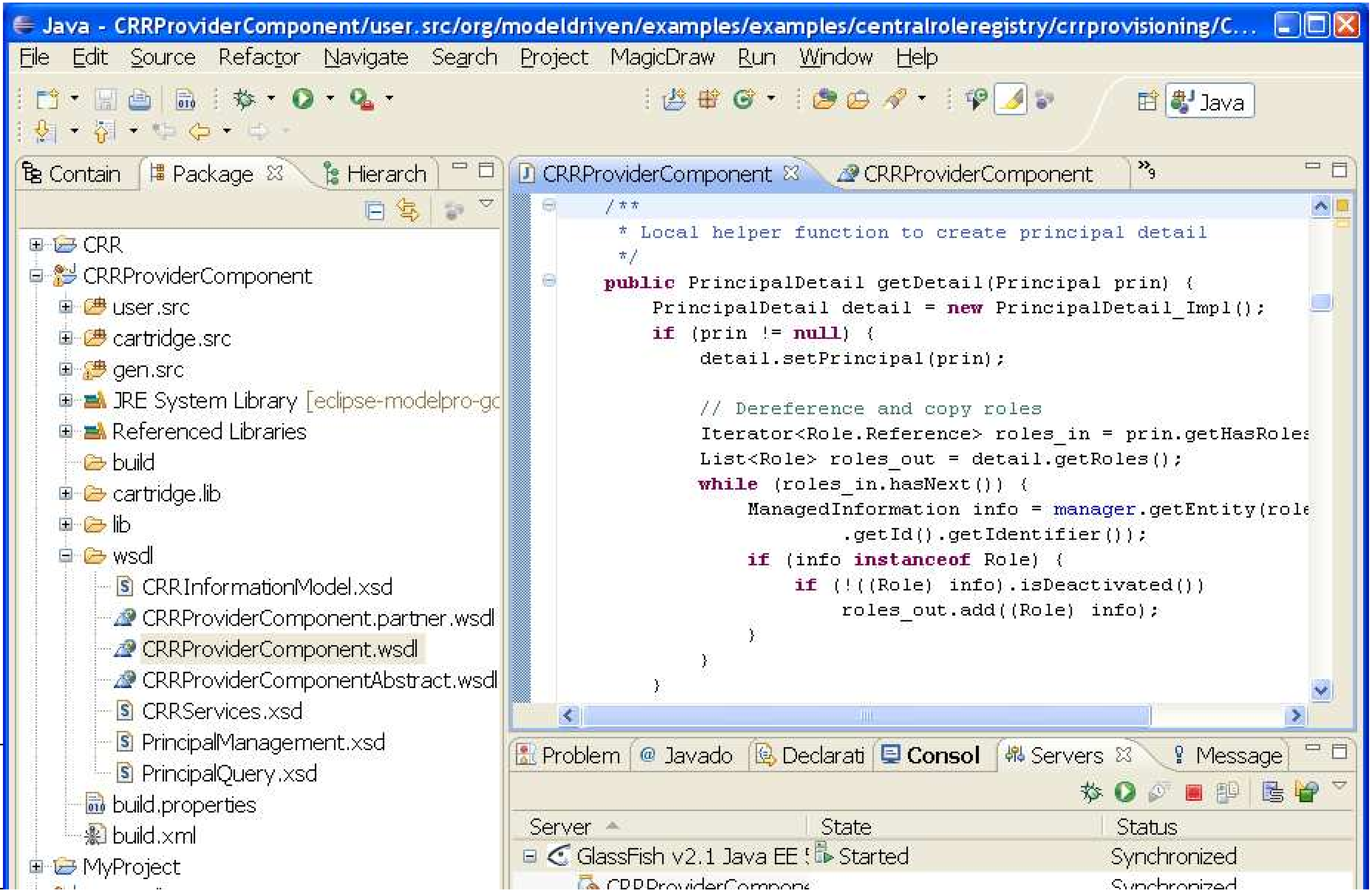
As MDA progresses, there will be less and less need for custom components, but the capability will remain.

Generated Artifacts in Java IDE



Server	State	Status
GlassFish v2.1 Java EE !	Started	Synchronized

Java Override Code

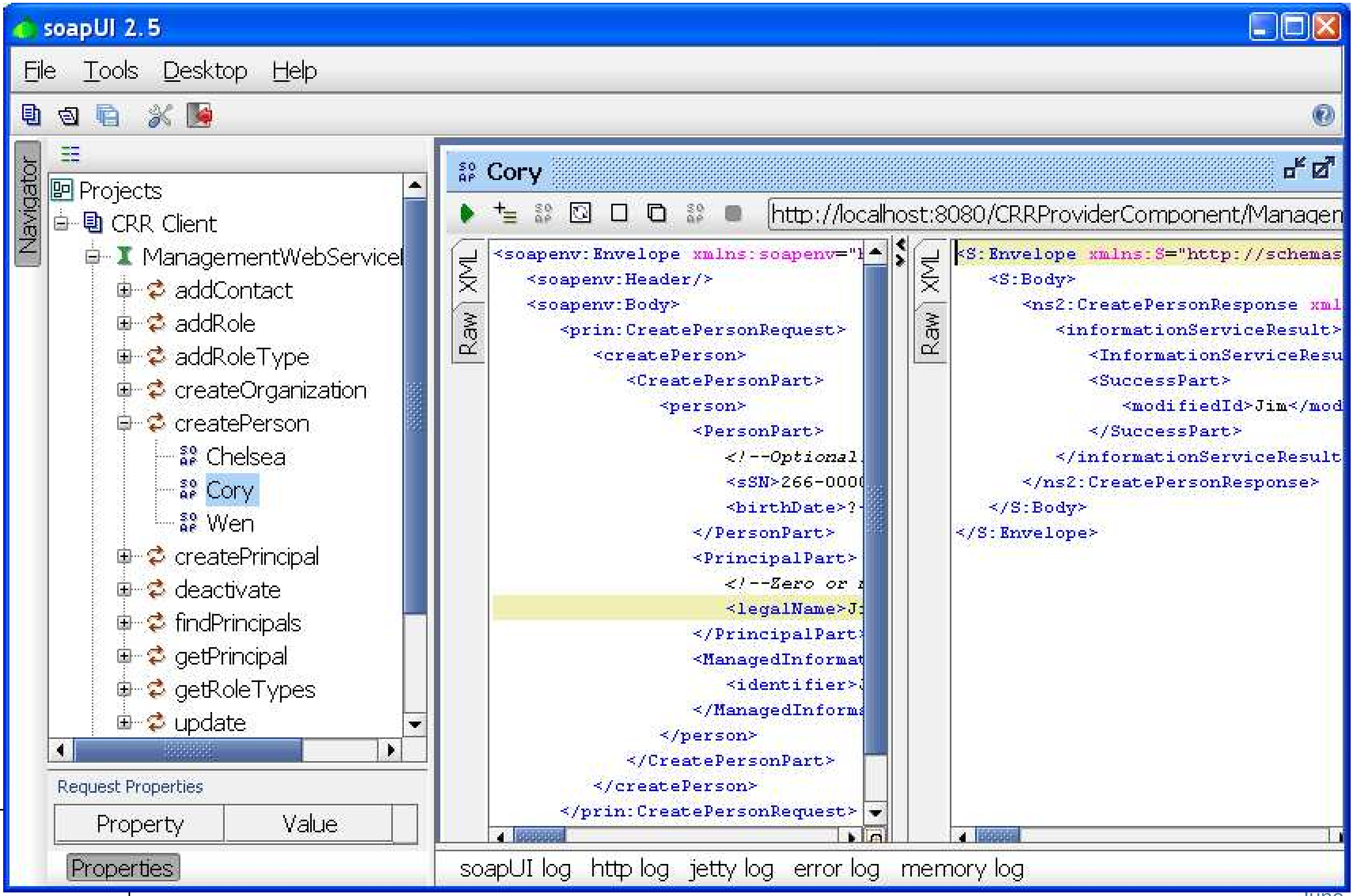


```
/**
 * Local helper function to create principal detail
 */
public PrincipalDetail getDetail(Principal prin) {
    PrincipalDetail detail = new PrincipalDetail_Impl();
    if (prin != null) {
        detail.setPrincipal(prin);

        // Dereference and copy roles
        Iterator<Role.Reference> roles_in = prin.getHasRoles
        List<Role> roles_out = detail.getRoles();
        while (roles_in.hasNext()) {
            ManagedInformation info = manager.getEntity(role
                    .getId().getIdentifier());
            if (info instanceof Role) {
                if (!((Role) info).isDeactivated())
                    roles_out.add((Role) info);
            }
        }
    }
```

Server	State	Status
GlassFish v2.1 Java EE !	Started	Synchronized
CRRProviderCompone		Synchronized

Using the deployed service from an ugly client



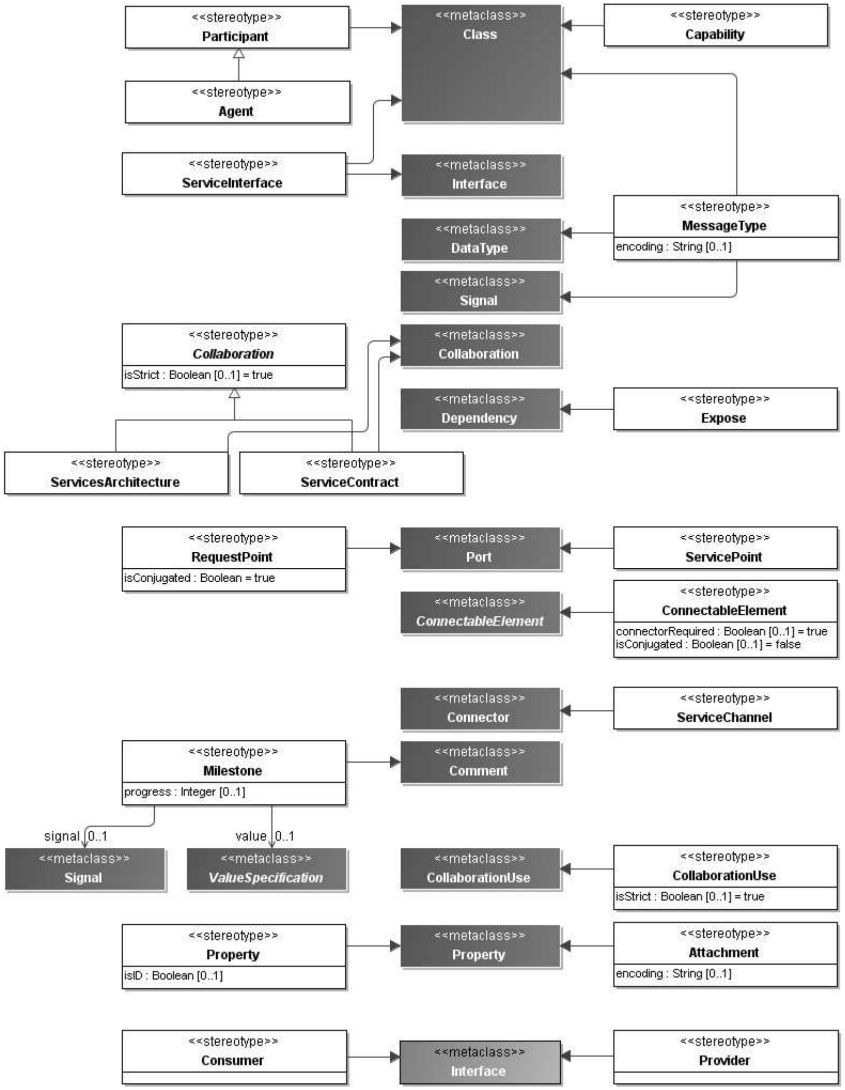
http://localhost:8080/CRRProviderComponent/Managen

```
<soapenv:Envelope xmlns:soapenv="I
   <soapenv:Header/>
   <soapenv:Body>
      <prin:CreatePersonRequest>
         <createPerson>
            <CreatePersonPart>
               <person>
                  <PersonPart>
                     <!--Optional:
                     <sSN>266-0000
                     <birthDate>?-
                  </PersonPart>
                  <PrincipalPart>
                     <!--Zero or m
                     <legalName>J
                  </PrincipalPart>
                  <ManagedInformat
                     <identifier>J
                  </ManagedInforma
               </person>
            </CreatePersonPart>
         </createPerson>
      </prin:CreatePersonRequest>
```

```
<S:Envelope xmlns:S="http://schemas
   <S:Body>
      <ns2:CreatePersonResponse xml
         <informationServiceResult>
            <InformationServiceResu
            <SuccessPart>
               <modifiedId>Jim</mod
            </SuccessPart>
         </informationServiceResult
      </ns2:CreatePersonResponse>
   </S:Body>
</S:Envelope>
```

The SoaML Profile

- SoaML is defined as a small set of UML stereotypes.
- These specialize a UML tool for use with SoaML.
- Standard UML can be used as well, as part of a SoaML model.
- Some tools provide enhanced SoaML support.



<<stereotype>> Participant
<<stereotype>> Agent
<<stereotype>> ServiceInterface
<<metaclass>> Class
<<stereotype>> Capability
<<metaclass>> Interface
<<stereotype>> MessageType
encoding : String [0..1]
<<metaclass>> DataType
<<metaclass>> Signal
<<stereotype>> Collaboration
isStrict : Boolean [0..1] = true
<<metaclass>> Collaboration
<<metaclass>> Dependency
<<stereotype>> Expose
<<stereotype>> ServicesArchitecture
<<stereotype>> ServiceContract
<<stereotype>> RequestPoint
isConjugated : Boolean = true
<<metaclass>> Port
<<stereotype>> ServicePoint
<<metaclass>> ConnectableElement
<<stereotype>> ConnectableElement
connectorRequired : Boolean [0..1] = true
isConjugated : Boolean [0..1] = false
<<metaclass>> Connector
<<stereotype>> ServiceChannel
<<stereotype>> Milestone
progress : Integer [0..1]
<<metaclass>> Comment
signal 0..1
value 0..1
<<metaclass>> Signal
<<metaclass>> ValueSpecification
<<metaclass>> CollaborationUse
<<stereotype>> CollaborationUse
isStrict : Boolean [0..1] = true
<<stereotype>> Property
isID : Boolean [0..1]
<<metaclass>> Property
<<stereotype>> Attachment
encoding : String [0..1]
<<stereotype>> Consumer
<<metaclass>> Interface
<<stereotype>> Provider

What you need beyond SoaML

- SoaML is a profile of UML, it needs a UML tool in which to be used – some tools may offer additional support to make creating a SOA easier
- MDA provisioning requires additional tooling which goes beyond the standard. E.G. ModelPro (ModelDriven.org) and RSA (IBM)
- Some SOA execution framework and/or ESB (I.E. “The Platform”)
- Optional, but highly desirable
  - Support for policies
  - Business process execution
  - Information modeling and implementation
  - Business rules
  - Requirements & motivation modeling
  - Security modeling and infrastructure
  - An IDE for elements that are not model driven
  - Application server and/or ESB

Current SoaML Support

- OMG Web site
  - SoaML Wiki: http://www.SoaML.org
  - Specification: http://www.omgwiki.org/SoaML/doku.php?id=specification

- Known SoaML Tooling
  - Cameo SOA+ (NoMagic) UML with SoaML Modeling and Provisioning
  - ModelPro (ModelDriven.org) Open Source MDA provisioning for SoaML
  - Enterprise Architect (Sparx) SoaML Profile for UML tool
  - Objecteering (Softeam) SoaML Profile for UML Tool
  - RSA (IBM) UML tool with SoaML & code generation [Not yet released]

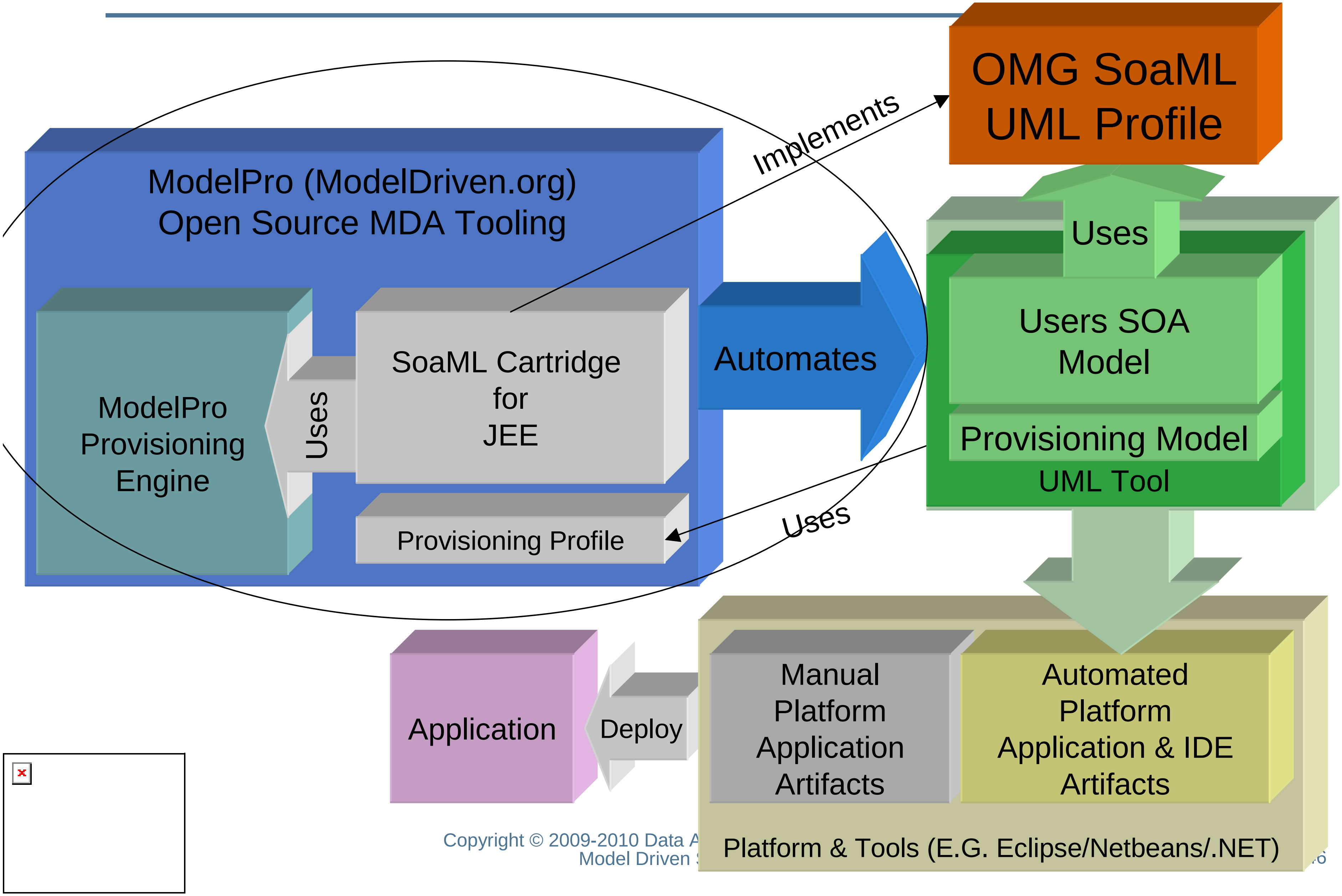
About ModelDriven.org, ModelPro

And the SoaML Cartridge

ModelDriven.org

- ModelDriven.org is a community sponsored by Model Driven Solutions that provides open source products that build on modeling, architecture and MDA
- This includes tools, infrastructure, solution oriented communities, models and ontologies
- Open source projects currently public on ModelDriven.org are:
  - **ModelPro** – Core MDA provisioning engine – derive value (of any kind) from models (of any kind)
  - **SoaML Cartridge** – The ModelPro provisioning specification and framework to support execution of SoaML models (Partnered with NoMagic for the SOA+ Modeling tool)
  - **Foundational UML** – Reference implementation of the executable UML standard
  - **Global Architecture & Information Network (GAIN)** – Community to support federation of architectures and information in support of the President’s Open Government Initiative

ModelPro & The SoaML Cartridge


OMG SoaML UML Profile
Implements
ModelPro (ModelDriven.org) Open Source MDA Tooling
ModelPro Provisioning Engine
Uses
SoaML Cartridge for JEE
Provisioning Profile
Automates
Uses
Users SOA Model
Provisioning Model
UML Tool
Uses
Application
Deploy
Manual Platform Application Artifacts
Automated Platform Application & IDE Artifacts
Platform & Tools (E.G. Eclipse/Netbeans/.NET)

ModelPro

- ModelPro is the open “provisioning engine” we sponsor on ModelDriven.org
- It is Eclipse based and uses “EMF” models as the basis for both input and output models
- Uses velocity templates to generate code, XML, HTML, descriptors, doc and other artifacts
- “Cartridges” plug into modelpro to map specific types of models to specific technology artifiacts
- The “SoaML Cartridge” can create a JEE implementation framework and eclipse project from a SoaML model
- Other modeling frameworks and other technoliogies can be supported
- The ModelPro roadmap includes support for other aspects of modeling (such as data modeling) and other technologies (such as spring), as required

NoMagic Cameo SOA+

- NoMagic Cameo SOA+ is the first tool to fully support SoaML with Web Services Provisioning
- Custom profile support makes creating SOA architectures easy
- Bundled with a commercial version of ModelPro for full MDA support
- See: http://www.nomagic.com/text.php?lang=2&item=338&arg=295

About Model Driven Solutions

http://www.modeldriven.com

Overview

- **Mission**
  - Develop the next generation of products and services that implement standards-based Object and Web technologies
- **Values: Our Beliefs**
  - Model-based service-oriented architecture, based on open standards that allow an open, agile, flexible and robust system to be built using enterprise level components
  - Automation using tools rather than “brute force coding”
  - Validation: A system and its components must be constantly and repeatedly validated
- **History**
  - Founded and Incorporated in 1996 by Cory Casanave
  - Joined OMG in 1996: Participated in development of CORBA, UML, EDOC, ebXML
  - Obtained a multi-million dollar NIST development grant in 1997: roots of Component-X
  - Expansion of services and customer base begins in 2001
  - First project with the US General Services Administration: 2003
  - First bid as prime contractor for a major US government contract: 2005

Specific Areas of Expertise

Enterprise & Solutions Architecture

- Enterprise architecture
- OMG MDA standards: UML, EDOC, SoaML, CCA, ADM, etc.
- Architecture Documentation
- Semantic Web/Ontology application to Architecture
- Enterprise SOA and process architecture

MDA based systems implementation

- SOA & BPM driven by business architecture
- Distributed systems, Web Services SOA
- Business Process Management

Open Source tools and infrastructure (ModelDriven.org)

- Component-X
- Eclipse based MDA provisioning of solutions for SOA
- Open source enterprise knowledge base (in progress)

Key Clients

Government & Government Contractors

GSA U.S. General Services Administration

NATIONAL ARCHIVES AND RECORDS ADMINISTRATION LITTERA SCRIPTA MANET 1985

U.S. ARMY

Raytheon

LMI GOVERNMENT CONSULTING

National Institute of Standards and Technology NIST ...working with industry to foster innovation, trade, security and jobs

Barquin International

LOCKHEED MARTIN We never forget who we're working for™

UNISYS

Commercial

KAISER PERMANENTE.

GM

DATA ACCESS WORLDWIDE

Copyright © 2009-2010 Data Access Technologies, Inc.
Model Driven Solutions

Partners

ASG®
Software Solutions

TACTICAL STRATEGY GROUP INC

Forestay
Ahead Forever

Booz | Allen | Hamilton
90 years delivering results that endure

ciber

INFOMANIACS

NO MAGIC

Synsyta
Synthesizing Systems of Systems

OmniSolve

TopQuadrant

Recent Projects

- GSA Human Resources Enterprise Architecture
- Executable UML Reference Implementation
- GSA/FAS ESB Evaluation
- GSA OSERA Registry & Enterprise Knowledge Base
- GSA Integrated Portfolio management (EA)
- GSA OCFO/OCIO Financial Management EA (and Prototype Implementation)
- Detailed Business Process Integration Model: Wireless Backbone
- UML Information Model of Domain Vocabulary for NARA
- Industry Expert Review of DoD's SIAP MDA Project
- GSA OCIO "One GSA" EA – "Mile wide & inch deep"
- Model & Simulation of US Army C4I Driver

Key Personnel

- Over 100 years Senior Management Experience and Expertise
- Cory Casanave, President and CEO, Founder
  - 20+ years product & application development management
  - Author of numerous standards: SoaML, BPDM, EDOC, ebXML, etc.
- Ed Harrington, Executive Vice President and COO
  - 30+ years technology market expertise: finance, marketing, etc.
  - Frequent speaker at industry conferences: Open Group, OMG, etc.
- Ed Seidewitz, Vice President MDA Services
  - 20+ years software development and systems architecture management
  - Active in standards development: SysML, fUML, UML
- Tom Digre, Vice President Development
  - 30+years designing and developing software tooling
  - Focus is on standards and MDA based architecture tooling

About Cory Casanave

- Object Management Group
  - Member since 1990
  - OMG Board Of Directors
  - SoaML Submitter and one of the primary authors
  - Participated in the foundations of MDA
  - Participated in multiple standards efforts over the years
    - SoaML, UML, MOF, EDOC, Corba, Etc.
- Other Community & Government Activities
  - Lead first “SOA Demo” as part of the “SoaCop” SOA Community Of Practice (a Federal Government Sponsored Community)
  - Chief Architect – U.S. GSA’s “OneGSA” Enterprise Architecture
  - Active in Open Group, Government and Semantic Communities
- 33 Years broad-based experience in software, systems, standards and architecture

Cory Casanave Bio (Continued)

- Founder, ModelDriven.org
  - Open community for everything model driven – supporting open standards, open source and open models. Embraces MDA, Semantic Web, SOA and BPM
  - ModelDriven.org hosts the “ModelPro” project supporting SoaML
- CEO and Founder of “Model Driven Solutions” since 1996
  - Professional services company for business architecture and model driven development focusing on SOA, Model Driven Architecture® (MDA®), Enterprise Architecture, Metadata, BPM, Semantic Web and closing the gap between business and technology. MDS primarily serves the U.S. federal government. Formerly operating as “Data Access Technologies”.
  - Methodology based on SoaML is part of our current practice

Thank You

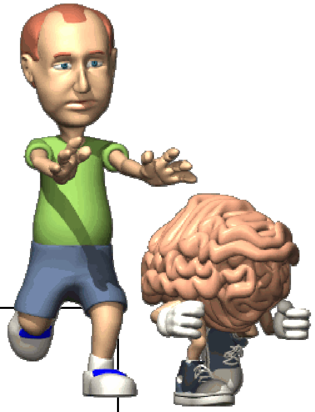
Additional Information:

SOAML at OMG OBJECT MANAGEMENT GROUP

http://www.SoaML.org

ModelPro™ : http://www.modeldriven.org

CAMEO™ SOA+ POWERED BY MODELPRO : http://soaplus.cameosuite.com/



Cory Casanave, CEO
Model Driven Solutions
Cory-c (at) modeldriven.com