



INFORMATION MODELING SERVICE-ORIENTED ARCHITECTURE

Galen Mulrooney
J P Systems, Inc.

www.jpsys.com

June 16, 2009

V 1.2



SERVICE ORIENTED ARCHITECTURE DEFINITIONS

- ✘ “Service Oriented Architecture (SOA) is a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains”

OASIS Reference Model for Service Oriented Architecture 1.0 - (emphasis mine)

- ✘ “Set of services that a business wants to expose to its customers and partners or other portions of the organization”

Source: IBM - (emphasis mine)

INFORMATION MODELING IN SOA

- ✘ In order for a SOA to work, it is critical that service providers have a common understanding of the structure and meaning of the data being exchanged
- ✘ Therefore, a “model” of some kind is needed
 - + What form should this model take?
- ✘ Is it possible to have a single, agreed upon model?
 - + What about different technological requirements?
 - + What about different viewpoints?

INFORMATION ARCHITECTURE

- ✘ Information Architecture is the analysis and design of the data used in information systems
 - + Modeling semantics and reference knowledge common to a wide range of artifacts
 - + Leveraging the semantics to ensure consistency across models/artifacts
 - + Using the same semantics as a basis for logical and physical database model generation, software component and service generation, rule development (e.g., in production rule-based systems), etc.
- ✘ It's all about logical consistency, validation, and reuse
- ✘ It's also about ensuring interoperability within an enterprise and between that enterprise and its trading partners

INFORMATION ARCHITECTURE (CONT.)

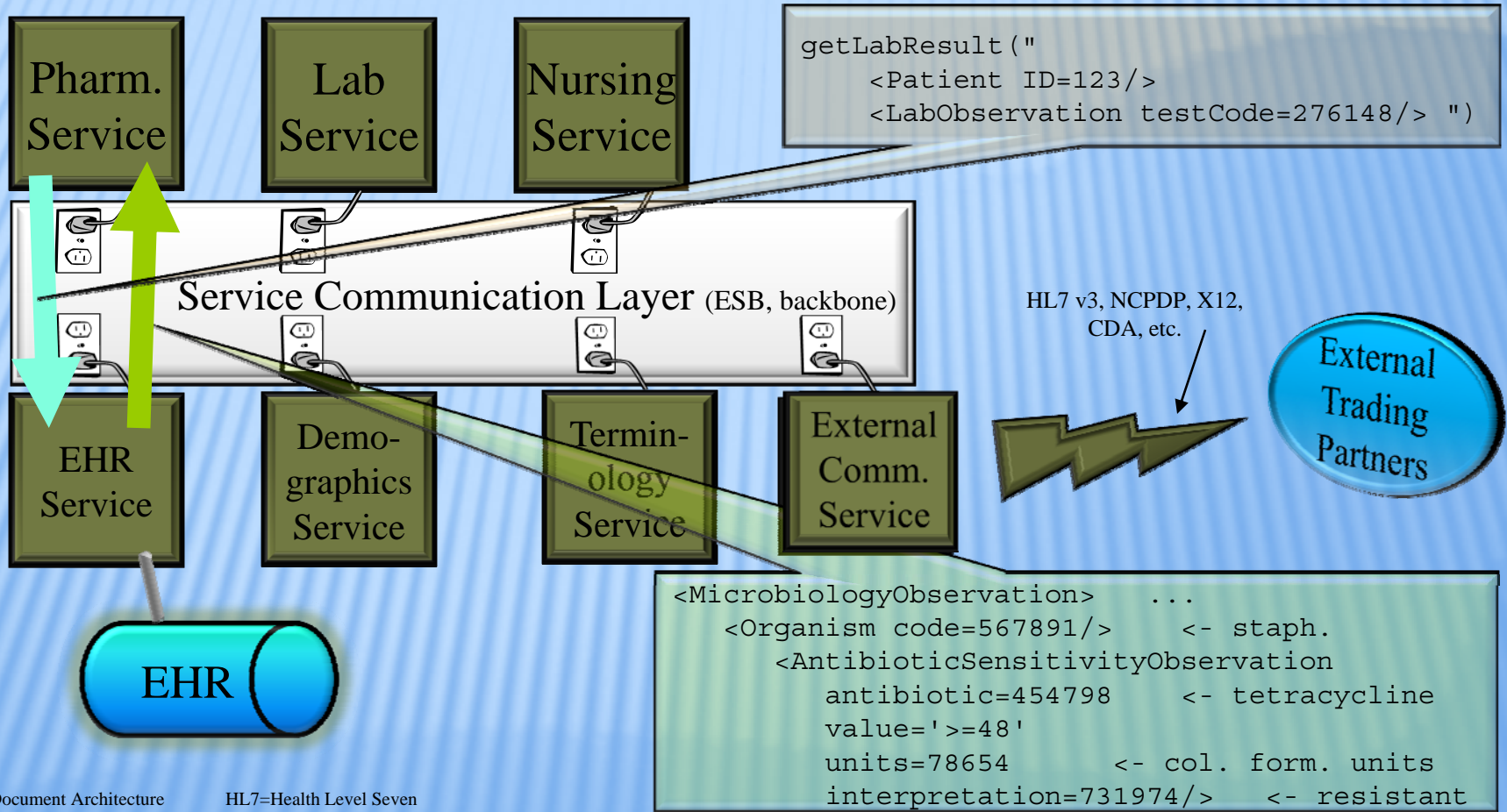
- ✘ The art and science of understanding the objectives of information in the context of the intended audience
- ✘ Information Architecture relates the Business Architecture to Information Technology
 - + It's the high level description of business information and communication
 - + Enables the translation of business perspectives to/from IT
- ✘ Business Strategy and Information Architecture are closely related
 - + Information Architecture enables the existing Business Strategy
 - + Information Architecture fosters new Business Strategies

INFORMATION ARCHITECTURE IN HEALTH CARE

- ✘ Legal Electronic Medical Record
 - + Definition and additional data about “metadata” elements that are deemed to be officially part of the Legal Electronic Medical Record
- ✘ Message “payload” design
 - + Definition of the contents of data sent to/from software systems as they fulfill business workflows
- ✘ Database and warehouse design
- ✘ Data standardization and data quality efforts
- ✘ Data requirements for quality improvement programs
- ✘ Medical research
- ✘ Clinical decision support

A HEALTH CARE SOA SCENARIO

Note that these services might be "home-grown" or commercial products



CDA=Clinical Document Architecture
 EHR=Electronic Health Record
 ESB=Enterprise Service Bus

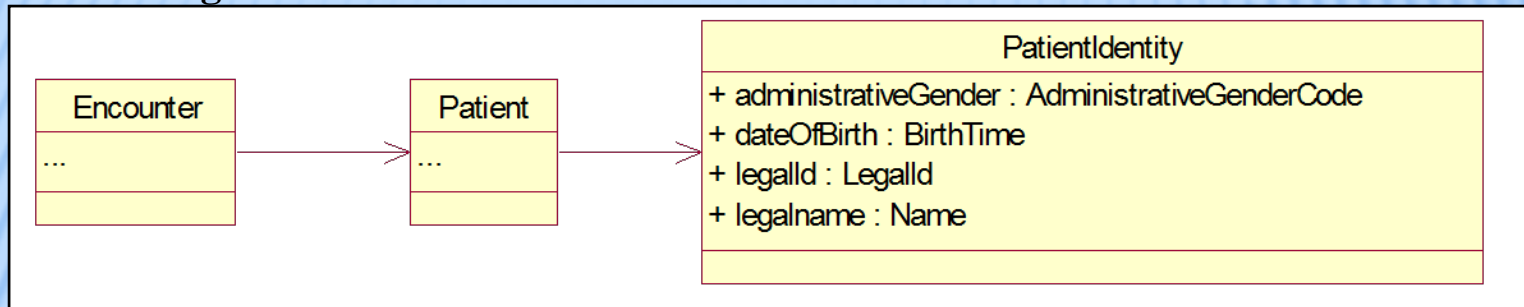
HL7=Health Level Seven
 NCPDP=Natl. Council of Prescription Drug Programs
 X12=Accredited Standards Committee X12

MODEL TRANSFORMATIONS

- ✘ How can one static data model meet all the implementation needs (database, domain model, payload structure)?
- ✘ Answer: Model transformations
 - + Computational Independent Model (CIM)
 - ✘ Conceptual model, aka Domain Analysis Model
 - + Platform Specific Model (PSM)
 - ✘ Adds concepts needed by a type of platform, e.g., keys
 - + Platform “Model”
 - ✘ Language used by platform, e.g., DDL, WSDL, CORBA
- ✘ Model Driven Architecture (MDA) and Unified Modeling Language (UML) provide tools for automating model transformations

UML TRANSFORMED TO OTHER LANGUAGES

Class Diagram



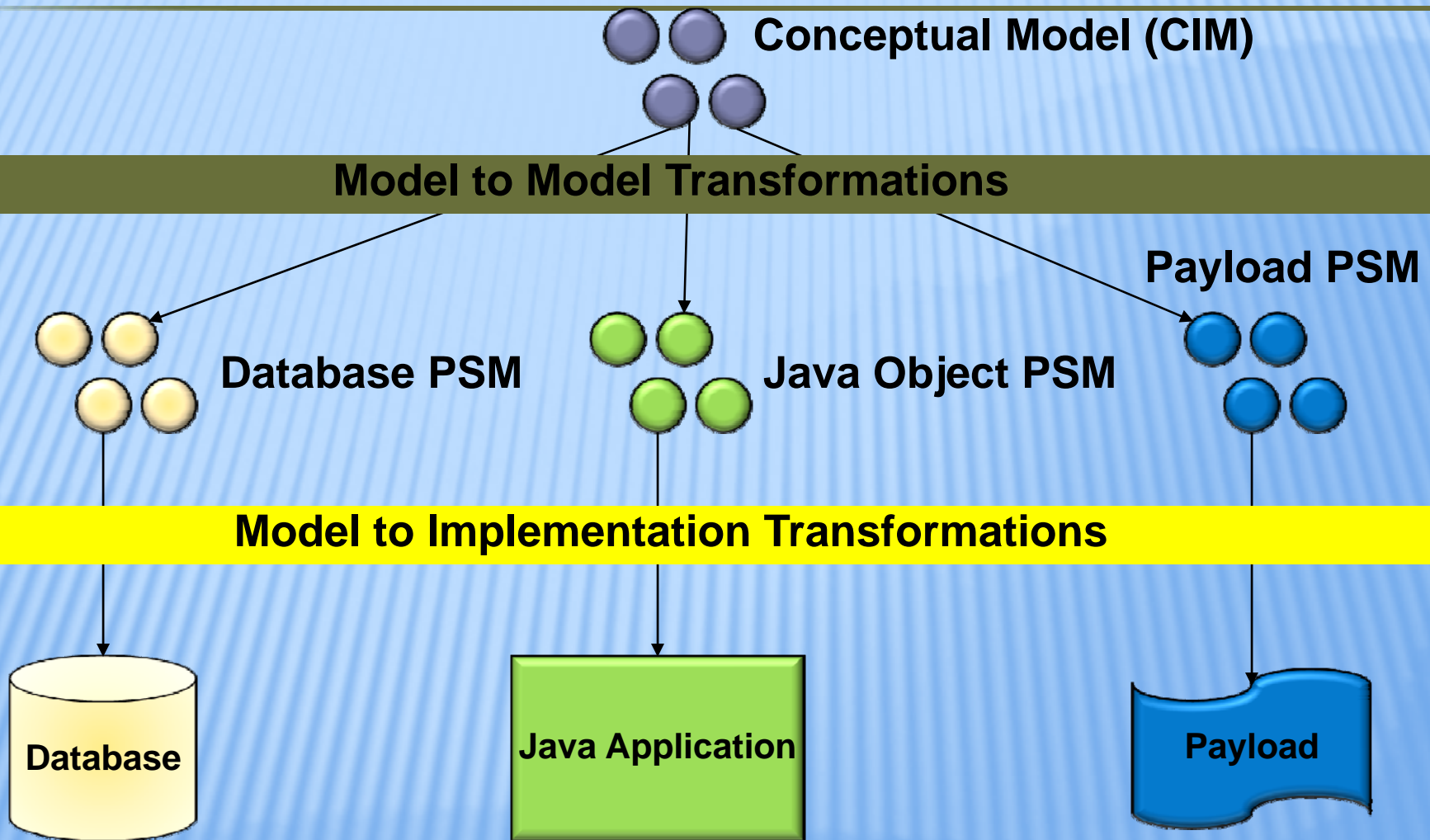
XML Schema Definition

```
<xs:element name="PatientIdentity" type="Validentity"
substitutionGroup="personIdentity" />
- <xs:complexType name="PatientIdentity">
- <xs:complexContent>
- <xs:extension base="PersonIdentity">
- <xs:sequence>
- <xs:element name="administrativeGender"
type="AdministrativeGenderCode">
- <xs:annotation>
<xs:documentation>A value representing the gender (sex) of a person.
The allowable values for this field as specified by the DS DAT for
Demographics are: F (Female), M (Male) and UN
(unspecified).</xs:documentation>
</xs:annotation>
...
```

Java

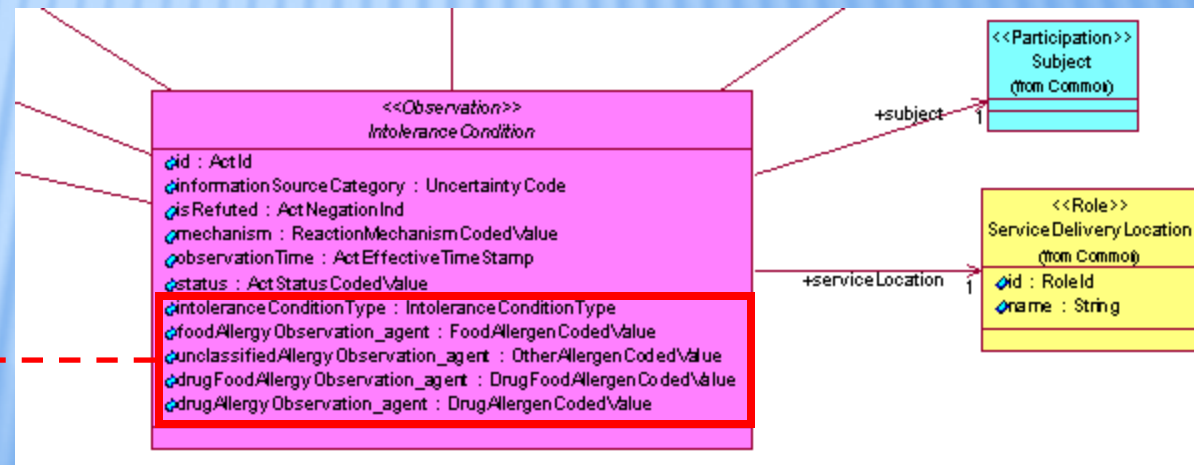
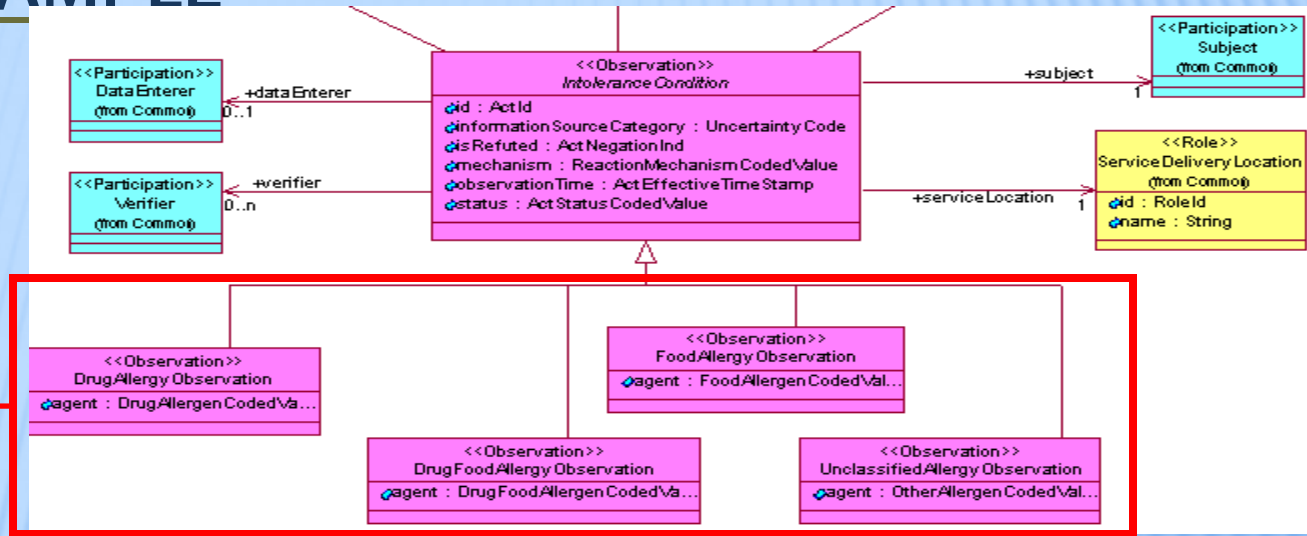
```
public interface PatientIdentity extends personSRDTs.PersonIdentity
{
    livingSubject.AdministrativeGenderCode getAdministrativeGender();
    void setAdministrativeGender(livingSubject.AdministrativeGenderCode
administrativeGender);
    livingSubject.AdministrativeGenderCode
addNewAdministrativeGender();
    livingSubject.BirthTime getDateOfBirth();
    void setDateOfBirth(livingSubject.BirthTime dateOfBirth);
    livingSubject.BirthTime addNewDateOfBirth();
    ...
}
```

MDA TRANSFORMS TO OTHER MODELS



CIM TO PSM EXAMPLE

Note the transformation from one model that has four subclasses to distinguish the type of the allergy, to another model that has a single class which uses an allergy type to distinguish the type of allergy

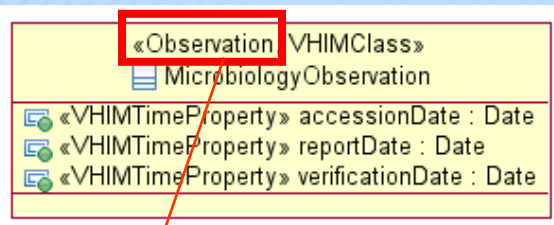


TRANSFORMATIONS AND STANDARDS

- ✘ The HL7 Reference Information Model (RIM) is used as a UML Profile, allowing a rigorous transformation to/from HL7 version 3 artifacts
 - + One can import and export HL7 Model Interchange Format (MIF) files to/from the UML model
- ✘ This can also be used to model other Standards
 - + NCPDP SCRIPT* being reverse-engineered
 - + Efforts underway at ASC X12 as well
- ✘ CDA authoring tooling is now being built

* NCPDP SCRIPT is a messaging standard used to convey pharmacy insurance claim information

ALIGN CLASSES WITH HL7 V3



The Class Code and Mood Code are explicitly identified in the model; allowing for computable transformations to/from HL7 v3 message structures

The screenshot shows the Properties window for the class «Observation, VHIMClass». The left sidebar has 'Stereotypes' selected. The main area shows the class name and a table of applied stereotypes. Below that, the 'Stereotype Properties' table is expanded to show the 'Observation' stereotype's properties, with 'classCode' and 'moodCode' highlighted in red boxes.

Stereotype	Profile	Required
Observation	RIM	False
VHIMClass	VHIM	False

Property	Value
Observation	
classCode	40 - OBS
moodCode	1 - EVN
VHIMClass	
client	Entries: 1
convertToChoice	False
hl7Mood	2 - event (occurrence)
isEntryClass	False
isMultiObservation	False
updateMode	Entries: 0

ALIGN ATTRIBUTES WITH HL7 V3 (CONT.)

«Observation, VHIMClass»
MicrobiologyObservation

- «VHIMTimeProperty» accessionDate : Date
- «VHIMTimeProperty» reportDate : Date
- «VHIMTimeProperty» verificationDate : Date

The HL7 RIM Attribute and constrained datatype are identified in stereotype properties, not in the model itself

Properties Tasks Console Bookmarks Problems

General

Stereotypes

Documentation

Constraints

Advanced

<Property> «VHIMTimeProperty» accessionDate

Keywords:

Applied Stereotypes:

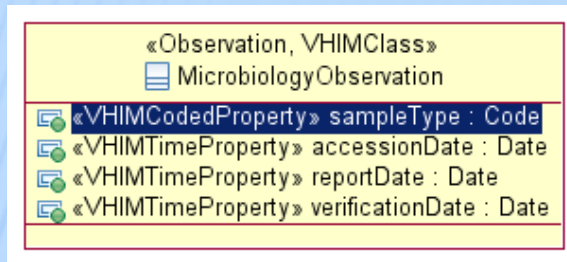
Stereotype	Profile	Required
VHIMTimeProperty	VHIM	False

Apply Stereotypes... Unapply Stereotypes

Stereotype Properties:

Property	Value
<input type="checkbox"/> VHIMTimeProperty	
canBeImprecise	False
client	Entries: 1
hl7Attribute	48 - effectiveTime
hl7Datatype	65 - TS
hl7ObservationClassName	
updateMode	Entries: 0

MECHANISM TO ALIGN WITH TERMINOLOGY



The model explicitly links to both the VHA Unique concept Identifier (VUID) and to the HL7 value set

Properties x Tasks Console Bookmarks Problems

General <Property> «VHIMCodedProperty» sampleType

Stereotypes Keywords:

Documentation Applied Stereotypes:

Stereotype	Profile	Required	
VHIMCodedProperty	VHIM	False	

Apply Stereotypes... Unapply Stereotypes

Stereotype Properties:

Property	Value
[-] VHIMCodedProperty	
client	Entries: 1
hl7Attribute	24 - code
hl7Datatype	12 - CD
hl7ObservationClassName	
updateMode	Entries: 0
valueSetVuid	12345
vocabularyDomainMnemonic	SMPLTYP
vocabularyDomainName	ActSampleType

UML STYLE BENEFITS

- ✘ The UML Style provides
 - + More computationally independent models; HL7-isms and XML-isms not in the diagrams
 - ✘ Easier for Subject Matter Experts to understand and validate
 - + Ability to automate Quality Assurance checks based on the semantics of the model, not just structure
 - ✘ This is done through Eclipse extensions using the Eclipse Modeling Framework (EMF) Validation Framework

LESSONS LEARNED

- ✘ There will be a time between the As-Is and the To-Be where you have a mixture of both
 - + Need to simultaneously maintain transforms to/from the UML model and HL7 v2 Electronic Data Interchange format, HL7 v3 XML, and CDA XML
 - + “As-Is” systems do not understand the “To-Be” concepts and constructs – these need to be removed from As-Is artifacts
- ✘ One challenge was the mapping to/from HL7 v2 data types and HL7 v3
 - + Certain v2 data types are needed for “backward compatibility”

LESSONS LEARNED (CONT.)

- ✘ Transformations require extensive testing
 - + Some HL7 messages might not conform to specifications
 - + Validation of mappings is manually intensive and time consuming
- ✘ Data migration/transformation will bring to light data quality issues
 - + Some data may be need to be cleansed
 - + Adjustments may be needed to the UML model or the transformations
- ✘ Some issues require the involvement of the authoritative source to make changes
 - + Important to have business input
 - + Need a “Data Stewardship” program to ensure authoritative business “owners” for semantics, governance, and error resolution/data quality

FUTURE WORK

- ✘ Research is being conducted to explore the linkage between UML and RDF, OWL, and SBVR*
 - + See the Object Management Group's (OMG) Ontology Definition Metamodel (ODM)
- ✘ Exploration of the use of ontologies and semantic web technologies
- ✘ Can the structural and ontological modeling worlds be bridged?

*

RDF=Resource Description Framework, a family of World Wide Web Consortium (W3C) specifications. See www.w3.org
OWL=Web Ontology Language, a family of knowledge representation languages for authoring ontologies. See www.w3.org
SBVR=Semantics of Business Vocabulary and Business Rules, an OMG standard. See www.omg.org

SUMMARY

- ✘ To design a SOA, you need
 - + Dynamic (Behavioral) Models
 - + Static (Information) Models
 - + Terminology Models
- ✘ UML can be used effectively for the first two
 - + Enables the benefits of Model Driven Architecture
 - + Link to terminology in UML Profile
 - ✘ Further exploration of the linkages between Information and Terminology modeling is needed
- ✘ Model Transformations allow a single model to satisfy multiple purposes