Solving the Data Problem

Cory Casanave

Cory Casanave
Email: Cory-c at modeldriven dot com
CEO, Model Driven Solutions
Founder, ModelDriven.org (Open Source)
OMG, Board of Directors
Architecture Ecosystem SIG, Co-chair

Model Driven Solutions
Where Business Meets Technology
A Division of Data Access Technologies, Inc.
Fundamental questions for solving the data problem

- What does the information mean?
- Who do I want to, and not want to, share and collaborate with?
- How is information for sharing and collaboration structured?
- What technologies enable secure sharing and collaboration?
- Where do I vest my community and institutional knowledge?
Exchange Data Structures (e.g. XML Schema)
Common Implementation (e.g. Cobol System)
Reusable Data Structures (e.g. NIEM XML)
Model Based Structures (e.g. NIEM UML)
Ontological (e.g. Common Logic)
Shared Concepts (e.g. Vocabularies)

Physical
How is it implemented & structured

Conceptual
What does it mean?

Scope of Applicability & Stakeholder Relevance

Generalize / Reference
Provision / Implement
User Needs

Common Implementation (e.g. Cobol System)
Reusable Data Structures (e.g. NIEM XML)
Model Based Structures (e.g. NIEM UML)
Ontological (e.g. Common Logic)
Shared Concepts (e.g. Vocabularies)

Scope of Applicability & Stakeholder Relevance

Generalize / Reference
Provision / Implement
User Needs

Common Implementation (e.g. Cobol System)
Reusable Data Structures (e.g. NIEM XML)
Model Based Structures (e.g. NIEM UML)
Ontological (e.g. Common Logic)
Shared Concepts (e.g. Vocabularies)

Scope of Applicability & Stakeholder Relevance

Generalize / Reference
Provision / Implement
User Needs

Common Implementation (e.g. Cobol System)
Reusable Data Structures (e.g. NIEM XML)
Model Based Structures (e.g. NIEM UML)
Ontological (e.g. Common Logic)
Shared Concepts (e.g. Vocabularies)
Where do I vest my community and institutional knowledge? e.g. What is the canonical representation?

- Have a wide scope, so I can collaborate easily
- Be applicable across a wide range of usage profiles
- Be applicable across a wide range of technologies
- Be easy to understand for stakeholders
- Be easy to implement in systems and exchange protocols
NIEM Conformant XML Schema

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <xsd:import namespace="http://niem.gov/niem/appinfo/2.1" schemaLocation="../../../niem/appinfo/2.1/appinfo.xsd"/>
  <xsd:import namespace="http://niem.gov/niem/structures/2.0" schemaLocation="../../../niem/structures/2.0/structures.xsd"/>
  <xsd:import namespace="http://www.modeldriven.org/niem/examples/PetAdoptionExtension" schemaLocation="../../../niem/examples/PetAdoptionExtension"/>
  <xsd:import namespace="http://niem.gov/niem/appinfo/2.0" schemaLocation="../../../niem/appinfo/2.0/appinfo.xsd"/>
  <xsd:import namespace="http://niem.gov/niem/proxy/xsd/2.0" schemaLocation="../../../niem/proxy/xsd/2.0/xsd.xsd"/>
  <xsd:import namespace="http://niem.gov/niem/niem-core/2.0" schemaLocation="../../../niem/niem-core/2.0/niem-core.xsd"/>
  <xsd:complexType abstract="false" name="PetAdoptionExchangeType">
    <xsd:annotation>
      <xsd:appinfo>
        <i:Base i:name="Object" i:namespace="http://niem.gov/niem/structures/2.0"/>
      </xsd:appinfo>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element maxOccurs="unbounded" minOccurs="1" ref="tns:People"/>
        <xsd:element maxOccurs="unbounded" minOccurs="1" ref="tns:Pets"/>
        <xsd:element maxOccurs="unbounded" minOccurs="1" ref="tns:PetAdoptions"/>
        <xsd:element maxOccurs="unbounded" minOccurs="0" ref="tns:PersonContactInformationAssociations"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:complexType>
</xsd:schema>
```

Very automatable but overly technology focused and tied to a specific application’s purpose.
Vocabularies

Person

*noun*

A human being regarded as an individual.
"the porter was the last person to see her"

Very reusable but not sufficiently precise and formalized to enable automated collaboration
A reasonable middle-ground, can define vocabularies but also produce implementation. Still not perfect!
The NIEM connection

NIEM-XML includes

- A vocabulary for information to be shared: **7000+ concepts**
- An XML based technical architecture, **300+ design rules**
- Reusable XML data structures – the reference schema
- Specific XML structures (IEPDs) for specific exchanges

NIEM-UML adds standards based

- A higher level of abstraction for all of the above, separation of concerns
- A user-friendly diagrammatic representation
- Ability to map to implementation as well as ontologies [not standardized]

NIEM as an asset

- The current asset is applied to XML sharing of specific data structures
- The **vocabulary** can be used and reused across a wide community and other technologies