Using New Standards to Develop IC Ontologies

STIDS 2010

Richard Lee

lee_richard@bah.com

Booz | Allen | Hamilton

Overview

- Previous Work: METS
- Recent Work: SCSP
- Universal Core
- OWL 2
- Future Work

Metadata Extraction and Tagging Service (METS)

- A DoDIIS service for processing documents:
  - Nomalization
  - Metadata detection
  - Entity extraction
  - Geotagging
- Produces XML:
  - IC PUBS
  - OWL/RDF

Ontology Work on METS

- Industry (OWL and KIF) Standards and near-Standards:
  - SUMO / MILO / etc (converted from KIF)
  - W3C's GML (supplemented) and Time
- Government (XML) Standards:
  - TWPDES (for person data)
  - ISM (for security markings)
  - DDMS (for other metadata)

New Project: SCSP

- Analysts needed to work with:
  - multiple data sources
  - multiple process models
  - multiple analysis models
- Analysts needed to perform:
  - search
  - discovery
  - correlation
  - presentation

New Ontology Work

- Building ontologies by converting models
  - Palantir
  - MIDB (partial), TIDE, Artemis, et al
  - CIA World Fact Book, ProMED, et al
  - PMESII, CTAF
  - various process models (maybe)
- Building a “master” ontology
  - Defines all the concepts of interest to the analysts
  - Defines the relationship between the concepts in all the others

Universal Core (UCore)

- A government-sponsored XML standard for information sharing
- Universal Core 2.0 included an OWL ontology
  - Class taxonomy only – no properties
  - Very minimal
- Universal Core Semantic Layer (UCore SL)
  - Started from the taxonomy and other elements in UCore
  - Used ideas from BFO to enhance them into a full upper-level ontology

Building on UCore SL

- SCSP master ontology was built on UCore SL
- Linked its concepts to those in all the others (plus METS)
  - Using subClassOf, equivalentClass, (ditto for Property), etc
- SUMO leveraged for many concepts
  - A few new classes wedged into the upper levels
  - Lots of new classes underneath

Example of Borrowing from SUMO: the Problem

- Equipment vs Sensor vs Vehicle vs Weapon vs etc
  - Artemis: Communication (Device), Equipment are distinct
  - MEPED: Equipment types include Communication Device, Sensor, Vehicle and Weapon, but only covers Military Equipment
  - Palantir: Equipment has children Communication Device, Sensor, et al, but Vehicle and Weapon are siblings of Equipment
  - UCore SL: Sensor, Equipment disjoint; Vehicle, Equipment unspecified; Weapon "in principle" would be a child of Equipment
  - (bio) DB: Equipment even includes the Kitchen Sink, and Lab Animals

Example of Borrowing from SUMO: the Solution

```
<owl:Class rdf:ID="Device">
  <rdfs:subClassOf rdf:resource="#Artifact"/>
  <owl:disjointWith rdf:resource="#Facility"/>
  <rdfs:subClassOf rdf:resource="&sumo;Device"/>
</owl:Class>

<owl:Class rdf:ID="Equipment">
  <rdfs:subClassOf rdf:resource="#Device"/>
  <owl:disjointWith rdf:resource="#ExplosiveDevice"/>
  <owl:disjointWith rdf:resource="#Sensor"/>
  <owl:disjointWith rdf:resource="#Vehicle"/>
  <owl:disjointWith rdf:resource="#Weapon"/>
  <rdfs:subClassOf rdf:resource="&art;Equipment"/>
</owl:Class>

<owl:Class rdf:ID="MeasuringDevice">
  <rdfs:subClassOf rdf:resource="#Device"/>
  <owl:disjointWith rdf:resource="#Communi.Device"/>
  <owl:disjointWith rdf:resource="#ExplosiveDevice"/>
  <owl:disjointWith rdf:resource="#Vehicle"/>
  <owl:disjointWith rdf:resource="#Weapon"/>
  <rdfs:subClassOf rdf:resource="&sumo;M.Device"/>
</owl:Class>
```

```
<owl:Class rdf:ID="Sensor">
  <rdfs:subClassOf rdf:resource="#M.Device"/>
  <owl:disjointWith rdf:resource="#Equipment"/>
  <owl:equivalentClass rdf:resource="&ucsl;Sensor"/>
  <owl:equivalentClass rdf:resource="&pal;Sensor"/>
</owl:Class>

<owl:Class rdf:ID="Bomb">
  <rdfs:subClassOf rdf:resource="#Weapon"/>
  <rdfs:subClassOf rdf:resource="#ExplosiveDevice"/>
  <owl:equivalentClass rdf:resource="&sumo;Bomb"/>
  <owl:equivalentClass rdf:resource="&pal;Bomb"/>
</owl:Class>

<owl:Class rdf:about="&ucsl;Equipment">
  <rdfs:subClassOf rdf:resource="#Device"/>
</owl:Class>

<owl:Class rdf:about="&meped;Equipment">
  <rdfs:subClassOf rdf:resource="#Device"/>
</owl:Class>
```

OWL 2 New Features

- Enhanced Annotation Property Usage
- New construct: AllDisjointClasses
- New construct: PropertyChainAxiom

AllDisjointClasses

- Problem: Listing, for each class, all its DisjointWith declarations is bulky and error-prone
- Solution: AllDisjointClasses

OWL 2 AllDisjointClasses enables simplification

```
<owl:Class rdf:ID="Device">
  <rdfs:subClassOf rdf:resource="#Artifact"/>
  <rdfs:subClassOf rdf:resource="&sumo;Device"/>
</owl:Class>

<owl:AllDisjointClasses>
  <owl:members rdf:parseType="Collection">
    <owl:Class rdf:about="#Equipment"/>
    <owl:Class rdf:about="#ExplosiveDevice"/>
    <owl:Class rdf:about="#Sensor"/>
    <owl:Class rdf:about="#Vehicle"/>
  </owl:members>
</owl:AllDisjointClasses>

<owl:Class rdf:ID="Equipment">
  <rdfs:subClassOf rdf:resource="#Device"/>
  <owl:disjointWith rdf:resource="#Weapon"/>
  <rdfs:subClassOf rdf:resource="&art;Equipment"/>
</owl:Class>
```

```
<owl:Class rdf:ID="Sensor">
  <rdfs:subClassOf rdf:resource="#M.Device"/>
  <owl:equivalentClass rdf:resource="&ucsl;Sensor"/>
  <owl:equivalentClass rdf:resource="&pal;Sensor"/>
</owl:Class>
```

PropertyChainAxiom

- Problem: Models differ in what they define as first-class objects, and what is simply represented by name/code strings
  - Model A says *Location* is a first-class object, linked to by other objects
  - Model B says *Location* is represented in other objects by its name or country code string
  - How to relate the two?
- Solution: PropertyChainAxiom

OWL 2 PropertyChains for cross-ontology links

- a's *countryOfBirth* is an ObjectProperty linking *Person* to *Country*
- b's *birthCountry* is a DatatypeProperty of *Person* simply naming a country

```
<rdf:Description rdf:about="&b;birthCountry">
  <owl:propertyChainAxiom rdf:parseType="Collection">
    <owl:ObjectProperty rdf:about="&a;countryOfBirth"/>
    <owl:DatatypeProperty rdf:about="&a;name"/>
  </owl:propertyChainAxiom>
</rdf:Description>
```

(It was not clear from the OWL 2 specification whether using PropertyChainAxiom to relate DatatypeProperties was allowed, but it's been confirmed)

AnnotationProperty

- Problem:
  - Capturing security, provenance, confidence, etc is critical
  - Agreeing on how to represent that is needed for secure handling and effective data sharing
  - ICS500-21 requires all XML representations to use the ISM XML schema for security markings
  - This is not workable for many pre-defined XML schemas such as RDF/XML
  - We need to propose a representation which can be ratified as a standard in lieu of ISM XML
- Solution: PropertyChainAxiom

OWL 2 Annotations for Security (ISMv3.owl)

```
<owl:Class rdf:ID="CVE_Classif">
  <owl:oneOf rdf:parseType="Collection">
    <owl:Thing rdf:about="#U">
      <rdfs:comment>UNCLASSIFIED</rdfs:comment>
      <ism:security rdf:resource="#U-USA"/>
    </owl:Thing>
    ...
  </owl:oneOf>
</owl:Class>

<ism:Security rdf:ID="U-USA">
  <ism:classification rdf:resource="#U"/>
  <ism:ownerProducer rdf:resource="#USA"/>
</ism:Security>

<owl:Class rdf:ID="Security">
<owl:intersectionOf rdf:parseType="Collection">
  <owl:Restriction>
    <owl:onProperty rdf:resource="#classification"/>
    <owl:cardinality>1</owl:cardinality>
  </owl:Restriction>
   <owl:Restriction>
     <owl:onProperty rdf:resource="#ownerProducer"/>
    <owl:minCardinality>1</owl:minCardinality>
  </owl:Restriction>
  <owl:Restriction>
    <owl:onProperty rdf:resource="#classification"/>
    <owl:allValuesFrom rdf:resource="#CVE_Classif"/>
  </owl:Restriction>
  <owl:Restriction>
    <owl:onProperty rdf:resource="#ownerProducer"/>
    <owl:allValuesFrom rdf:resource="#CVE_OP"/>
  </owl:Restriction>
  <owl:Restriction>
    <owl:onProperty rdf:resource="#relTo"/>
    <owl:allValuesFrom rdf:resource="#CVE_Rel"/>
  </owl:Restriction>
  …
 </owl:intersectionOf>
</owl:Class>
```

OWL 2 Annotations for Security (Usage)

```
<owl:Axiom>
  <owl:annotatedSource rdf:resource="#ID1"/>
  <owl:annotatedProperty rdf:resource="&example;memberOf"/>
  <owl:annotatedTarget rdf:resource="#ID2"/>
  <ism:security rdf:resource="#Sec1"/>
</owl:Axiom>

<ism:Security rdf:ID="Sec1">
  <ism:classification rdf:resource="#S"/>
  <ism:ownerProducer rdf:resource="#USA"/>
  <ism:disseminationControls rdf:resource="#REL"/>
  <ism:relTo rdf:resource="#USA"/>
  <ism:relTo rdf:resource="#CAN"/>
  <ism:relTo rdf:resource="#GBR"/>
  <ism:relTo rdf:resource="#ISAF"/>
</ism:Security>
```

- Asserts the triple
- Annotates the triple with its security

AnnotationProperty

- Problem:
  - Palantir ontology had class hierarchies cut off at a fairly high level; used various *type* properties below that
  - Palantir's lack of multiple inheritance also led to use of *type* properties
  - Palantir provides "Composite Properties" where OWL needs to use simple classes
  - We needed to have a reasonably formal specification of the mapping from that ontology to the OWL one
- Solution: AnnotationProperty

OWL 2 Annotations for Mapping (Usage)

```
<owl:Class rdf:ID="Acoustics%20Sensor">
  <rdfs:label>acoustics sensor</rdfs:label>
  <translationClass>Equipment</translationClass>
  <translationProperty>
    <Property>
      <pname>Equipment Type</pname>
      <pvalue>Sensor</pvalue>
    </Property>
  <translationProperty>
  <translationProperty>
    <Property>
      <pname>Sensor Type</pname>
      <pvalue>Acoustics Sensor</pvalue>
    </Property>
  </translationProperty>
  <rdfs:subClassOf rdf:resource="#Sensor"/>
</owl:Class>
```

Future Work

- Add mappings in METS ontology to the concepts in UCore SL
- Continue data source and process modeling in SCSP
- Continue retrofitting OWL 2 constructs into the ontologies of both
- Work with the community on standardizing ontologies such as ISM

Contact Info

Richard Lee

lee_richard@bah.com cnleera@dodiis.ic.gov

571-482-7809

METS info

http://purl.org/mets http://mets.dodiis.ic.gov