SOA, ROA and the Semantic Web

How the Common Terminology Services (CTS_2) Packages Ontology for Service Oriented and RESTful Architecture

Outline

Terminology Services – what is it and why do we care?

Common Terminology Services – quick background

CTS2 – PIMs, SOA, ROA and RDF

“Terminology”

A generic “term” that covers a broad spectrum of resources:

- Code / value pairs – Country codes, airline codes, fare codes, …
- Controlled vocabularies – when United Airlines uses “confirmed reservation”, they mean “…”
- Thesauri – If you are looking for information on the Tiger Swallowtail, you might also look for information on butterflies, moths, insects, … (BT, NT, RT, definition, note, …)

“Terminology” (continued)

- Classifications – “buckets” for various statistical and categorization purposes. Example: Classification of Diseases Edition 9 with clinical modifiations (ICD-9-CM) – 250.01 “Diabetes mellitus without complication type 1 not stated as uncontrolled”
- Formal Logic Systems – Web Ontology Language (OWL), OBO, Common Logic, others. - classes, individuals, relationships. SNOMED-CT, Gene Ontology, FMA, … as well as CYC, SUMO, and others

Why Terminology Services?

Start with the country codes:

- Where do you download them?
- What format are they in?
- How are they organized? How do I list just EU countries? African countries?
- Oh – they change???? How do I know when and what?

Why Terminology Services?

Now I've got them loaded…

- I construct some SQL queries to create a dropdown list for a form…
- … I write some more to create an Ajax App on a web page
- … and some more to print country names on a report
- …
- The new accounting package uses three character codes? Ack. I write a conversion
- Oh – they haven't updated their tables? What do we do with "YU"?
- ISO *reuses* codes?????

Why Terminology Services?

- We need to export our data to our new owner – they have completely different tables…
- … or they've never heard of ISO and they use 01 – 299 for countries…
- … and they want to know what currencies are (and were) used in a given country …

Why Terminology Services

Now, take the problems above and figure that a reasonably complex business such as a medical practice has to deal with *hundreds* of tables / classifications / logic systems…

… each with its own formats, labels, rules, …

… *many* separate application providers that create and consume terminologies..

… and data exchange requirements with competing, complementary businesses

… and an ever increasing load of reporting / analysis / billing requirements..

And you need to stop and ask – isn't there some way we can manage all of this stuff in one place (or at least in one way…)

Sample Terminology Service Applications

- Maintenance of choices for forms and data
- Translation / mapping between values in data
- Indexing – both NLP and manual
- Classification
- Subsumption testing
- Model granularity translation
- Pick list translation
- Definitions / examples / neighborhood (hoverbox type work)

Terminology Services

Goal is to specify a common set of interfaces to:

- Import
- Query
- Update
- Create Updates
- Export
- Reason about

Terminology *and* its relationships to data

CTS_2 Background

Common Terminology Services Edition 2

Derived from:

- OMG LQS Specification (1999)
  - OO Model, read only, but laid most of the groundwork
- HL7 CTS Specification (2004)
  - ANSI and ISO Standard
  - SOA Model, read only, reduced scope from LQS

CTS_2

Will consist of:

1) A “Service Functional Model” – a CIM and a collection of requirements
2) An OMG RFP – done through the Ontology PSIG
3) A submission with a PIM and at least one PSM
4) An implementation (!)

CTS_2
Additional Drivers and Requirements

- NCI/Mayo LexEVS compatibility
- Semantic Web / Ontology community buy-in
- BioPortal compatibility
  - RESTful compatible architecture
- Alignment w/ II4SM model
  - Reasoning
  - Z representation
- OMV alignment
- API4KB Alignment
- Addl: Phin VADS, HL7 MIF

CTS_2

Mayo/II4SM submission (Apelon, Sandpiper, Visumpoint as supporters) is targeting both SOA and ROA

ROA leads to some really interesting (and useful) modeling decisions…

ROA

That's the Resource-Oriented Architecture. It's just four concepts:

- Resources
- Their names (URIs)
- Their representations
- The links between them

and four properties:

- Addressability
- Statelessness
- Connectedness
- A uniform interface

CTS_2 Resource "Profiles"

Abstract Resource

- Code System
- Map
- Value Set
- Concept Domain

Resource Version

- Code System Version
- Map Version
- Value Set Definition
- Concept Domain Binding

Changeable

Entity
Association
Map Entry
Statement

CTS_2 Function "Profiles"

- Read
- Query/Search (w/ iterators)
- Import / Export (external formats)
- Incremental Update
- Resource History Query
- Update Creation
- Temporal Query
- Resource specific (reasoning, value set comparison, etc.)

Modeling Approach

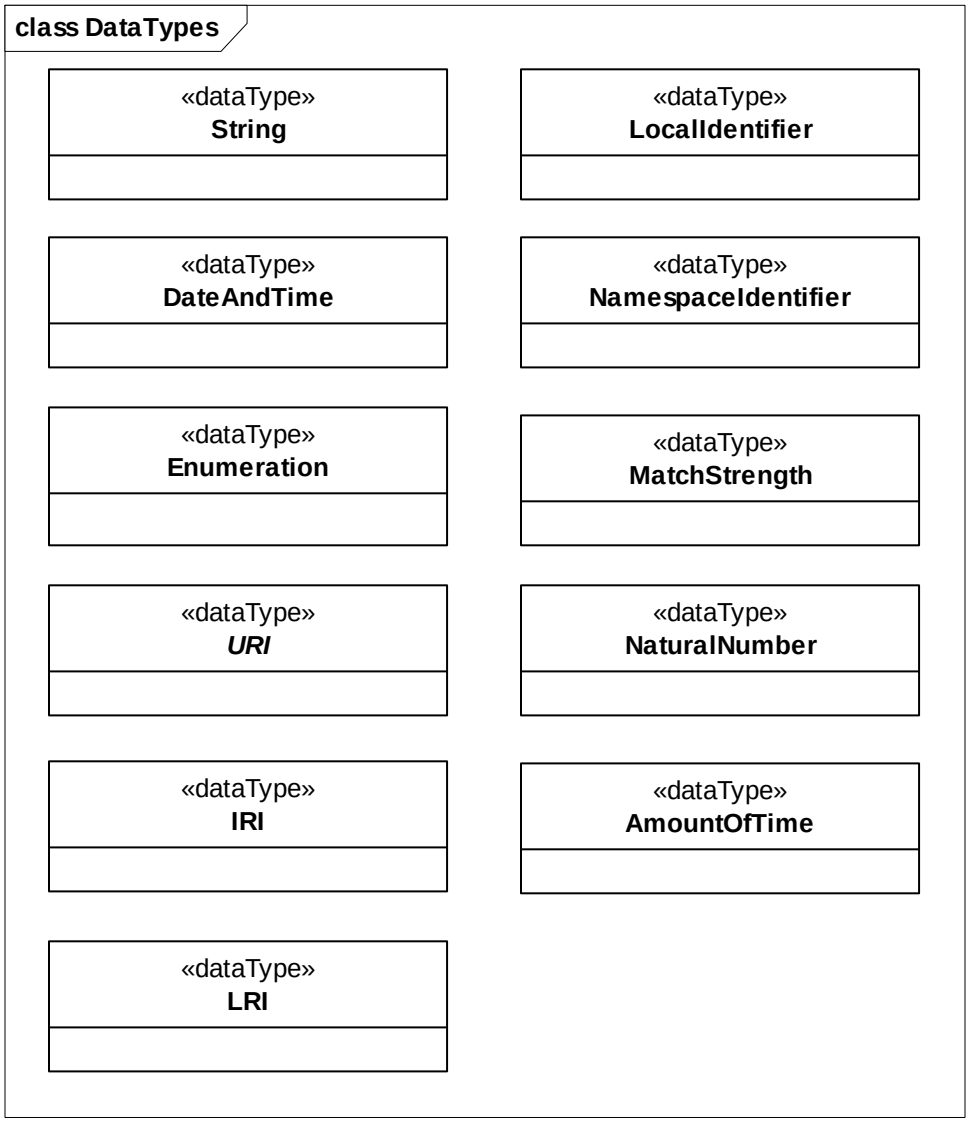
- UML Structure
- Z Semantics (invariants, preconditions, postconditions)
- Text in the middle

Experience, Issues, Questions

- What, exactly *is* a PIM?
  - Not obvious
  - Z helps ("you need a precise way to describe business processes")
  - What of URIs (IRI, interestingly is probably not PIM?)
- SOA, ROA, Pojo requirements
  - Pre- vs. post-orchestration
  - Choreography – in scope?
  - Error handling

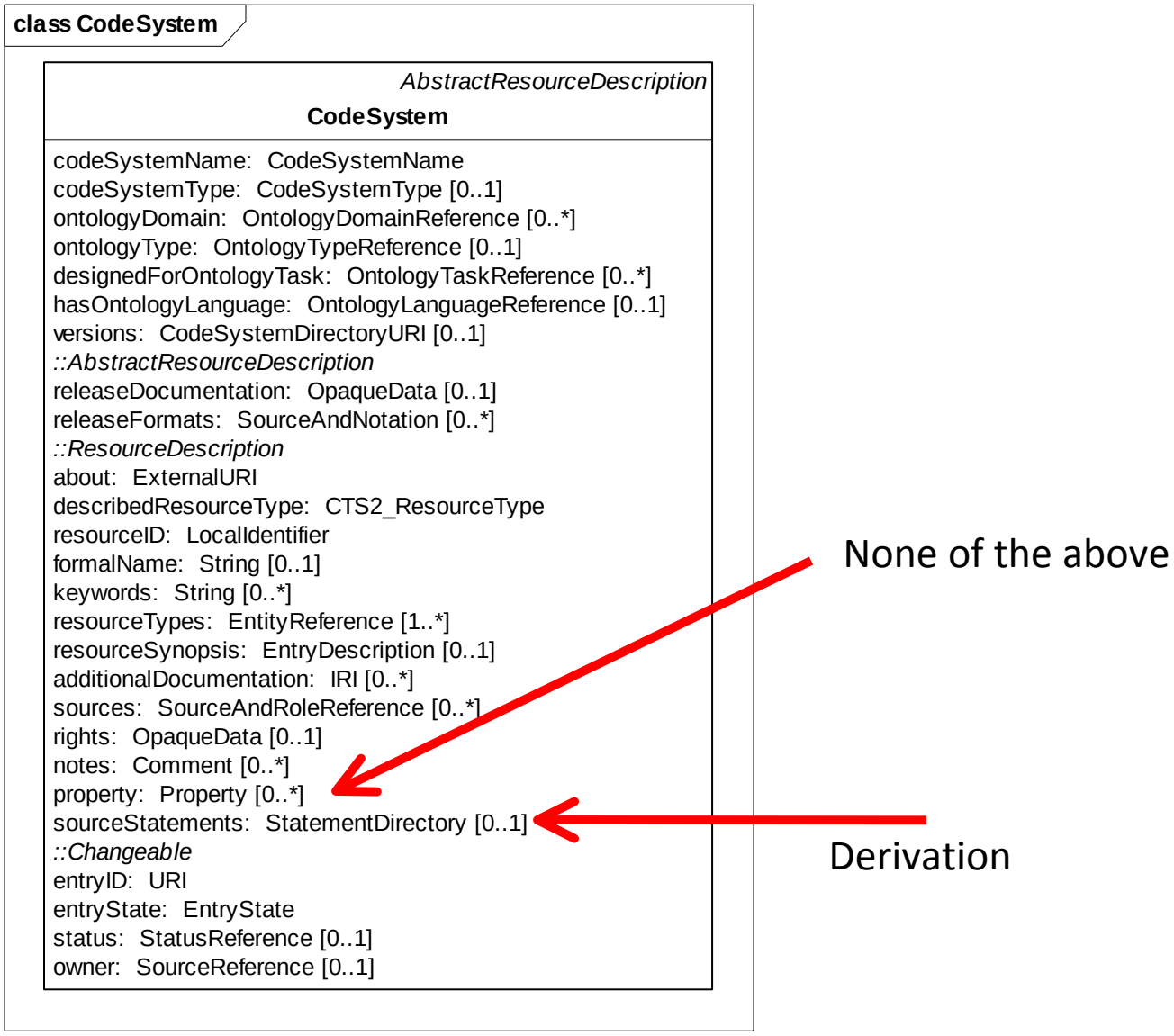
Experience, Issues, Questions

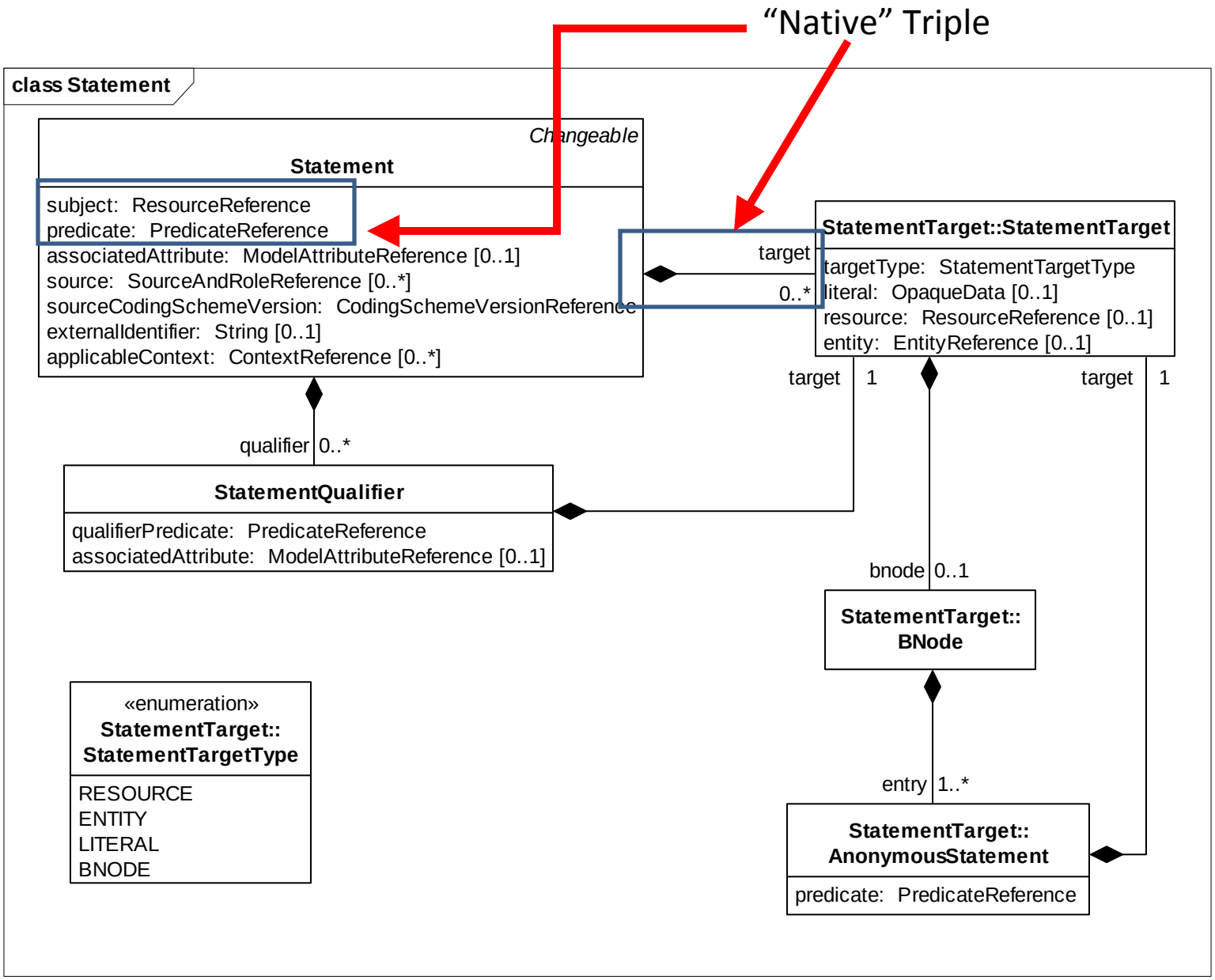
- Z issues
  - Precise vs. Detailed (Kilov)
  - Precision has a cost (and benefit?)
  - Allergies, LaTeX and machinability…
- REST issues
  - Identity for triples (arm part of body)
- RDF/OWL vs. traditional model
- Agility – this is (sort of) waterfall writ large…

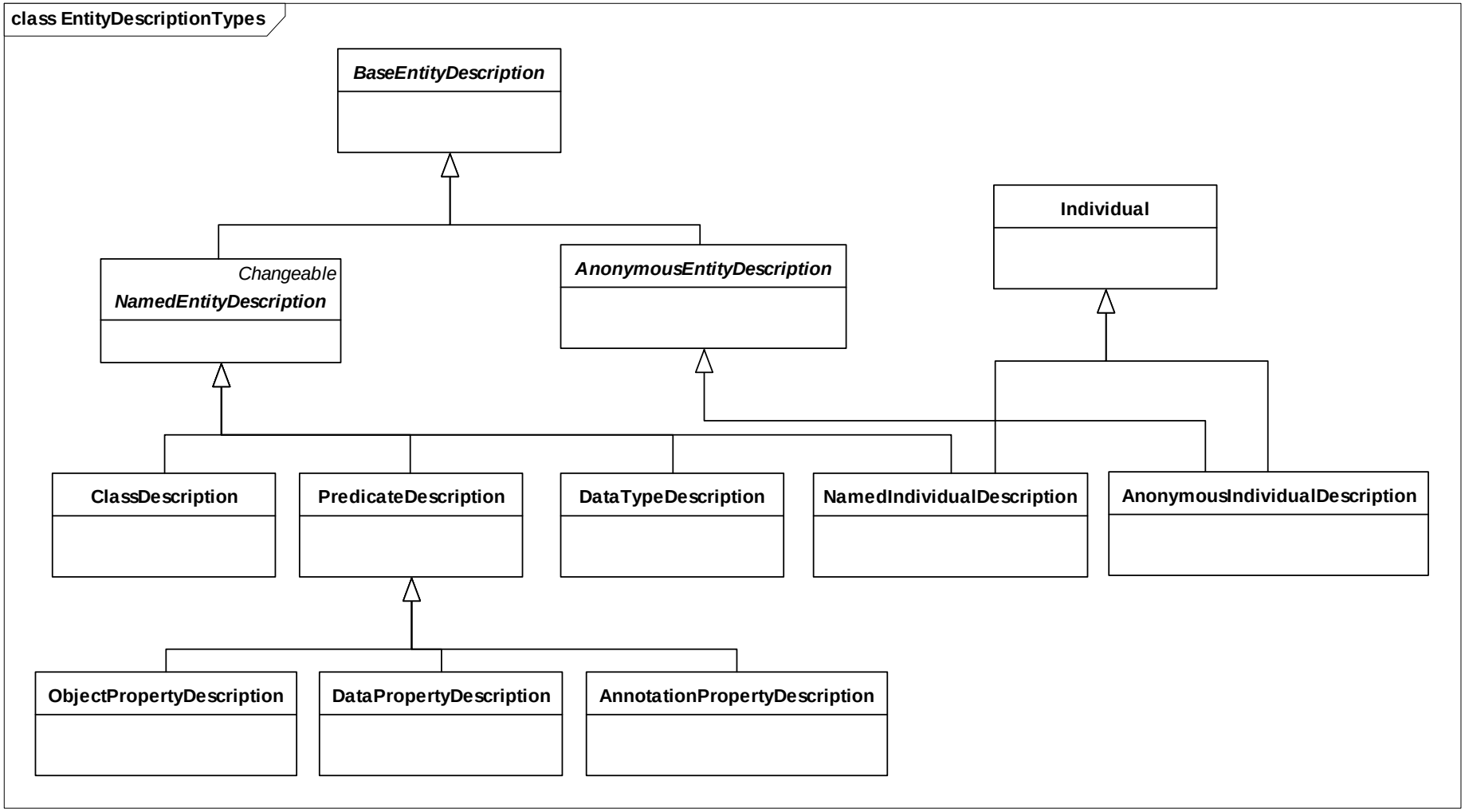


CTS_2 and RDF

- (i)RDF – a “canonical” map from CTS_2 to RDF tags.
- Statement Model







One more note

Terminology forms a component of both the information and behavioral semantics of a model. Even if you don't use terminology services, or anything more than enumerated dropdown lists, you need to agree on the meaning and behavior of the terminology

CTS_2 provides a meta-model of static and dynamic terminology behavior that allows reasonable terminology model analysis (!)

References

- http://informatics.mayo.edu/cts2 - (under construction, but useful references)
- http://www.iso.org/iso/english_country_names_and_code_elements
- http://www.ics.uci.edu/~taylor/documents/2002-REST-TOIT.pdf (fielding paper)
- http://en.wikipedia.org/wiki/Representational_State_Transfer

©

Solbrig.harold@mayo.edu