

USDL and the Web of Data

Dr. Carlos Pedrinaci
(The Open University)

Web of Data

“in recent years the Web has evolved from a global information space of linked documents to one where both documents and data are linked. Underpinning this evolution is a set of best practices for publishing and connecting structured data on the Web known as Linked Data.”

Bizer et al. Linked Data - The Story So Far.
International Journal on Semantic Web and Information Systems (IJSWIS) (2009)

Linked Data Principles

1. Use URIs as names for things
2. Use HTTP URIs so that people can look up those names.
3. When someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL)
4. Include links to other URIs. so that they can discover more things.

On Applying Linked Data Principles

- The first principle is catered for by RDF. Good care must be taken on exposing every entity with proper URIs, reusing existing instances in as much as possible. (See identifier patterns in [Linked Data Patterns book](#))
- This indeed relies on a (set of) proper vocabularies for modelling the data. (See next analysis on vocabulary and Modelling patterns in the [Linked Data Patterns book](#))

On Applying Linked Data Principles

- The second and third principles should be catered for by the server exposing USDL data. Content negotiation, SPARQL, etc are easily supported by state of the art tools. See more details on Publishing Patterns in [Linked Data Patterns book](#)
- The fourth principle requires analysing existing data sources to identify possible links and asserting these

1st Step:
Representing USDL in RDF

1st Step: Representing USDL in RDF

Foundation Module

Foundation Module

- The foundation module basically covers:
 - Time representation
 - Location
 - Agents
 - Resources
 - Artifacts

General Aspects

- In RDF every element has a URI
 - IdentifiableElement is not necessary
- RDF(S) and OWL provide a good deal of the aspects captured by Element Description
 - Name -> rdfs:Label (could also use skos:prefLabel)
 - Description -> rdfs:Comment
 - Concept -> rdf:type
 - Language -> part of the literals, e.g., @en
 - Synonyms -> see [WordNet RDF](#)
- Tags covered by [Tag ontology](#)

Time

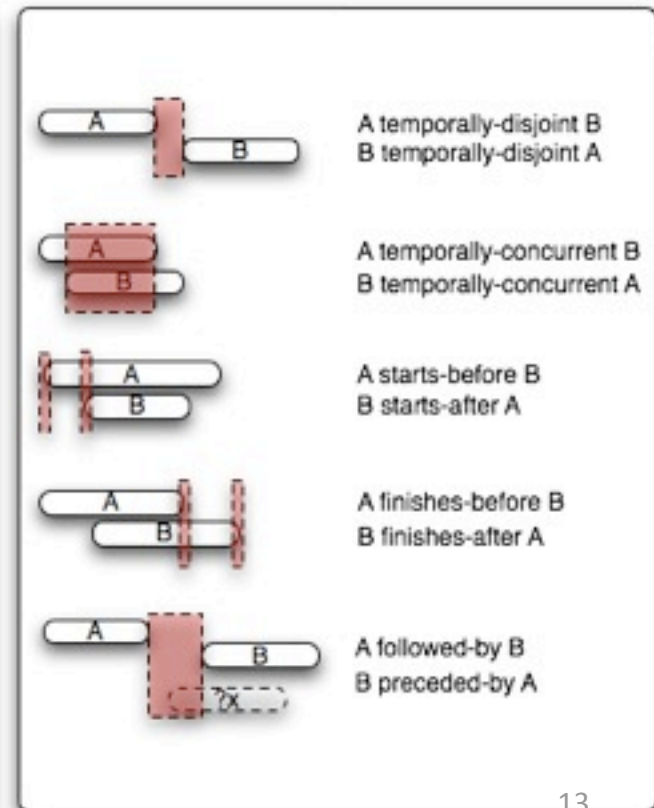
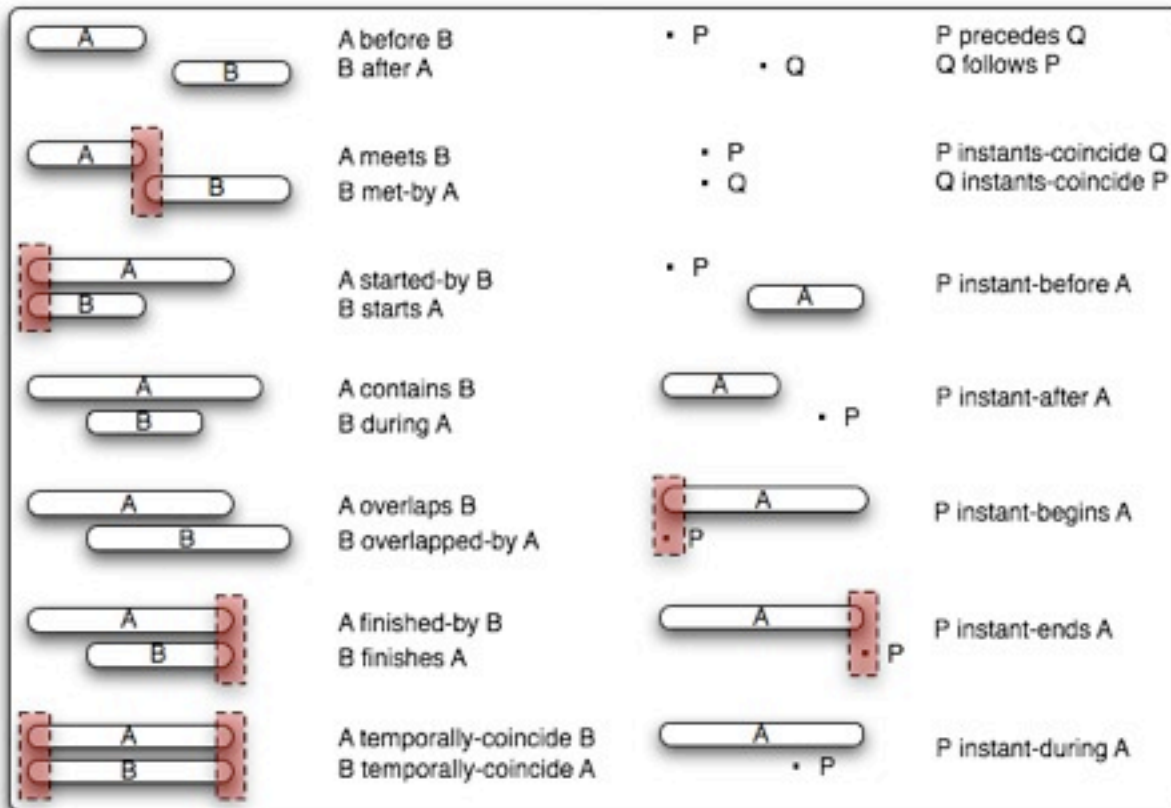
- Most of the time can be handled directly with [W3C OWL Time](#)
- The notion of duration is therein defined explicitly as composed out of a number of values for different units of time.
- In other approaches like [Time Ontology - OCML](#) this is dealt with via functions on the basis of its duration.

Time

- Time zones are imported from the tzonto at W3C which actually covers part of the regional descriptions used in the Location part [Timezone Ontology](#)
- The Time Ontology covers everything except:
 - RecurrentTime definitions (concerns TimePattern as well)
 - RelativePointInTime
 - DurationInterval (I'm not clear on the meaning of this one)
- Note: the most fine grained value in Time W3C are seconds. Would need fixing.

Time

- Time manipulation would support implementing Allen's interval temporal algebra



Location

- [W3C WGS84](#)
 - Very widely used to identify places (lat, long, altitude)
- Ordnancesurvey [Spatial Relations](#)
 - Covers main spatial relations
- Ordnancesurvey [Administrative Geo](#)
 - Focussed solely on the UK but could be used as the basis for a more general vocabulary

Location

- Ordnancesurvey [Postcode](#)
 - Covers UK postcodes. Could be reapplied in some sense for other countries
- Ordnancesurvey [Abstract Geometry](#)
 - Covers abstract geometry notions such as polygons, etc
- Some more info from W3C Geo XG:
 - [Vocabulary](#)

Location

- These ontologies would cover to a great extent the location specific requirements for the module, e.g., PhysicalLocation, AdministrativeArea, Polygon, GeographicalPoint.
- Still requires the notion of Address and Virtual Address (see vCard next)

Units of Measure

- Time and Location rely on the handling of units of measure
- Currently USDL handles these as separate and only nominal units
- No knowledge is explicitly captured about the relationships between different units for the same magnitude, different magnitudes and their manipulation, etc

Units of Measure

- Possible approaches:
 - Ontolog [UoM ontology](#)
 - Still in rather early stages it seems and agreement process is rather slow
 - My own set of ontologies based on EngMath by T. Gruber and modelled in OCML
 - See some info at [\[1\]](#)
 - Handles [Prefixes](#), [Dimensions](#), [Physical Quantities](#), [International System of Units](#), [their manipulation](#)

Units of Measure

- Handling of Units would be supported better than currently, supporting mathematical operations with quantitative values
- NOTE: **The use of UnitOfLengthType in concept Area in USDL is incorrect. For these the unit should be of area (m², etc). This is a different dimension**

Addressing

- W3C Submission [vCard RDF](#) covers all of it
- FOAF covers part of it
 - Equivalence statements may be necessary for additional compatibility
- Some aspects would require further analysis
 - Virtual Region ?
 - Messaging System?

Classifications

- [WSMO-Lite](#) FunctionalClassificationRoot:
 - For strict inferencing classifications
- [SKOS](#)
 - For more relaxed Knowledge Organisation Systems

Agents, Resources, Artifacts

- The notion of Agent is covered by [Dublin Core](#) and [FOAF](#)
- The notion of Natural Person is covered by FOAF
- Organisation
 - Covered entirely by [Organisation Ontology](#)
 - Integrated with FOAF, GoodRelations, vCard

Agents, Resources, Artifacts

- Artifacts
 - Copyright supported by Dublin Core
 - Dc:rights and dc:license
 - MimeType to point to one of [RFC3870](#)
- Artifact types to be defined as a hierarchy or with SKOS

Remaining Concepts

- Resources to be refined at a later stage based on their use in other modules