Worked Example: Veterinary Syndromic Surveillance

Methods and Tools for Modular Ontology Modeling, Part 3

Karl Hammar
The Project

- What is Veterinary Syndromic Surveillance?
- Animal Health Surveillance Ontology, AHSO
- AHSO Purposes:
  - Early warning systems
  - Data integration for reporting to EU agencies
  - Data entry assistance in various support systems
- Development team: 1 core member, 2-3 co-devs, 2-3 hangarounds (all part-time)
- Developer skills: veterinary science, biology, statistics
- Ontology Engineering novices at the outset but learning rapidly
AHSO Development 2015-2018

- Year 1-2:
  - Developing an understanding of tools/technologies/tradeoffs, ontology engineering, building network, finding stakeholders, etc.
  - Studying existing ontologies, their advantages and disadvantages.
  - Prototyping through eXtreme Design workshops.
  - Initial use cases: only in epidemiology.

- Year 3-4:
  - Real ontology development started.
  - Additional national and EU funding obtained, additional use cases developed.
  - eXtreme Design and ODPs used as aids in development but not followed in detail.
  - Karl comes on board to help veterinarians with development work (for both the ontology and tooling to construct/consume it).
AHSO Today

- 37 classes, 31 object properties, 7 datatype properties
- DL expressivity: ALIF(D)
  - I.e., base language + inverses + functional properties + datatype properties
- In need of refactoring
  - But work so far has clarified requirements and been useful all the same
- https://github.com/SVA-SE/AHSO
- http://w3id.org/ahso
- https://nandadorea.gitbooks.io/ahso/
owl:Thing
- 'Geographical Information'
  - 'Health and surveillance actors'
    - 'Animal owner'
    - Veterinarian
- Observation
  - Birth
  - Death
    - 'Fallen stock'
      - 'Fallen stock destroyed on farm'
      - 'Fallen stock not destroyed on farm'
    - Declared movement
    - Pathological examination
      - 'Ante Mortem Pathological Examination'
      - 'Post Mortem Pathological Examination'
    - Slaughter
      - 'Home slaughter'
      - 'Stillbirth event'
    - Observation context
      - Necropsy
      - 'Registered Animal Movement'
      - 'Surveillance activity'
      - 'Surveillance system'
    - Population unit
      - Herd
      - Individual
    - Registry
      - 'Movement Registry'
      - Sample
    - Zoographical Information
      - 'Age Category'
      - Breed
      - 'Production Type'
    - Sex
      - Female
      - Male
        - 'Castrated male'
    - Species
- owl:topDataProperty
  - hasAgeInformation
    - hasAgeUnit
    - hasAgeValue
    - hasDOB
    - hasID
    - isRecordedOnDateTime
    - occursOnDateTime
- owl:topObjectProperty
  - containsRecordsOf
  - 'has context member'
    - 'contains observation'
    - 'contains sample'
    - 'has member activity'
    - hasAgeCategory
    - hasBreed
    - hasGeographicInformation
    - hasMember
    - hasProductionType
    - hasRecordInRegistry
    - hasSex
    - hasSpecies
    - hasSubject
      - hasSubjectAnimal
      - hasSubjectHerd
      - hasSubjectAgeCategory
      - hasSubjectBreed
      - hasSubjectProductionType
      - hasSubjectSpecies
    - 'is a member of context'
      - 'is sample of'
      - 'observed in observation context'
    - 'is member of system'
    - isBreedOfSpecies
    - isMemberOf
    - isObservedBy
    - isOwnedBy
    - isOwnerOf
    - isRecordedBy
Modelling Issues

- T-box / A-box conflation
- Multi-species Agents
- Observations/Samples/Contexts
T-box / A-box conflation

**Symptom:** expressing data requires that classes be treated as values.

**Causes (?):** human “isA” insufficiently formal, not differentiating between subset and set membership. Lack of tree visualisation including members. Projects with unclear use cases. Reuse of existing taxonomies wholesale.

**Resolution:** Read W3C WG Note *Representing Classes As Property Values* from 2005 (Noy, Uschold, Welty). Consider OWL2 punning if needed.
Multi-species Agents

“Population unit” class and subclasses based on Martin Fowler’s *Accountability* pattern (from the book *Analysis Patterns*, highly recommended). *

**Question:** How do we ensure that herds only contain individuals of the same species?

**Answer:** We cannot: property chains cannot be used as cardinality restrictions (or be functional)
Observations/Samples/Contexts

**Goal:** representing attributes associated with animals/herds but recorded at some particular observation event.

**Solution:** Extend on Observation pattern (Blomqvist).

**Problems:** Boundary between Observation Context and Observation unclear - several object properties shared.

Shared property domain/range definitions incorrect (intersection, not union).
Reflections

- Underlying joint causes of errors:
  - Lack of hierarchy visualisation tooling
  - Lack of modular/folding ontology engineering tooling
  - Focus on formal correctness misses usability and common-sense correctness
  - Unintuitive RDFS domain/range semantics
  - Property chain limitations

- Some of these issues can be improved by better tooling: OPLa, ODPs, Protégé plugins, etc.
- Some require QA processes and user testing.
- Some might require new or modified standards.
- Some are unsolvable.