Datatypes

Uli Sattler
<sattler@cs.man.ac.uk>
Information Management Group
School of Computer Science
University of Manchester
Datatypes - in OWL now

- XML Schema datatypes:

- User-defined datatypes
  - no provision in the OWL syntax for conveying what these datatypes are

- What we can do:
  - (range hasAge xsd:integer) or (range address xsd:string)
  - (John hasAge 18) or (Mary address “theBlueHouse”)
Datatypes - in OWL now

• What we cannot do:
  – No intervals, etc.: Adult SubClassOf (All hasAge >= 18)
    Person SubClassOf
    (All address In“[A-B]Letter*”)
  – No comparisons, etc.: Square SubClassOf (width, height, =)
    Person SubClassOf
    (SystBP, DiastBP, >=)
  – …n-ary datatype predicates in general
  – Have rationals, e.g. 1/3
  – In OWL DL: have datatype properties as keys
    • InverseFunctional does not work for datatype properties
  – Composite keys: neither on object nor on datatype properties
  – Many more: we post more use cases
Datatypes - resulting issues

• n-ary
  – Issue 5: Doubts about n-ary datatypes
    • Fragile, syntax missing
  – Issue 53: Linear inequality(n-ary) xsd related
    • a use case requiring n-ary datatypes
  – Issue 29: owl:DataRange vs rdfs:Datatype
    • how to call datatypes?
  – Issue 31: Canonical URI for externally defined datatypes
    • How to name datatypes?

• Others:
  – Issue 87: owl:rational
    • Why decimal isn’t sufficient and why we need rationals
  – Issue 74: xsd namespace for facets
    • Suggests to use “xsd:minInclusive” for <=
  – Issue 25: Range of values as extension of class
    • A use case for user defined datatypes/intervals
  – Issue 71: lang tag datarange
    • I didn’t understand
What we want/would like to give

The same as before, plus

1. A built datatype rationals,
   
   **ToDo**: accept

2. Comparisons with & without constants using schema “facet names” such as
   minInclusive, maxExclusive, equals, etc.
   
   • To be used in  
     
     | `Square SubClassOf (width, height, equals)` |
     
   and
     
     | `Adult SubClassOf (hasAge, 18, maxInclusive)` |
     
   • Similar for strings: length, minLength, infix, prefix, matches
   
   **ToDo**: list predicates & semantics, use as in OWL1.1 proposal
What we want/would like to give

The same as before, plus

3. A mechanism to define “fancy” datatype predicates
   - E.g., to define *Celsius-Fahrenheit* to be used in *PhysObject SubClassOf* (tempC, tempC, *Celsius-Fahrenheit*)
   - Mechanism for external only (not in-line), using BLD/SWRL/SPARQL syntax

**ToDo**: choose/specify mechanism & use as in OWL1.1 proposal
What we want/would like to give

The same as before, plus

4. Easy keys: a mechanism to
   • specify simple & composite keys for datatype & object properties
   • E.g., \((\text{SSNr } \text{keyFor USCit})\)
     \((\text{DoB, PoB keyFor GermanCit})\)
   • They will not affect classification/consistency, but only work on known individuals:

\[(x=y) \iff \text{USCit}(x), \text{USCit}(y), \text{SSNr}(x)=\text{SSNr}(y)\]

See http://code.google.com/p/owl1-1/wiki/EasyKeyProposal

**ToDo**: extend syntax of OWL1.1, specify semantics