

Bounds: Expressing Reservations about Incoming Data

Martin G. Skjæveland Audun Stolpe

W3C RDF Validation Workshop
10–11 September 2013



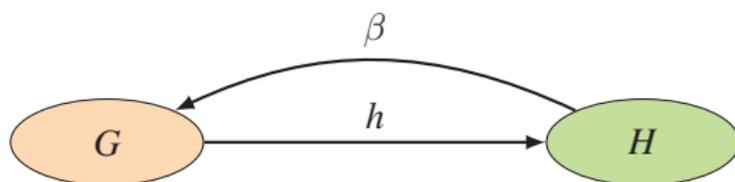
- Control additions to a dataset relative to its contents
- Dataset protection
- Bounded homomorphisms between RDF graphs
- Different bounds give different restrictions
 - Bounds \approx Triple Patterns \rightsquigarrow SPARQL
- RDF vocabulary
- Prototype implementation

Validation relative to receiving dataset

- Focus: Validate data to be added to an existing dataset
- Cannot do “traditional” validation, e.g.,
 - x is an integer and $0 \leq x \leq 100$
 - an x must have n y 's
 - $x \leq y$
- but, validation of data relative to receiving dataset, e.g.,
 - Do not add more superclasses to my dataset
 - New `f oaf : knows` relationships must relate new persons only
 - Write-protecting (parts of) a dataset
 - Adding new data must not rearrange existing data
- Difficult/Impossible to express with existing tools

- Prevent *ontology hijacking* (i.e., adding statements that effect the logical properties of existing ontology), to some degree
- Safeguarding particular vocabulary, and uses of it
- Write-protect subsets of the target, while keeping other parts of a dataset open
- Keep incoming data partitioned from current data

Theoretical foundation



- Homomorphism $h : G \rightarrow H$: $\langle s, p, o \rangle \in G \Rightarrow \langle h(s), h(p), h(o) \rangle$
 - Preserves the structure of G into H
- $h(u) = u$ for all elements u
 - Preserves data – no transformations allowed

- Bound (β)

$$\langle h(s), p, o \rangle \in H \Rightarrow \langle s, p, o \rangle \in G \quad (\text{S})$$

$$\langle s, h(p), o \rangle \in H \Rightarrow \langle s, p, o \rangle \in G \quad (\text{P})$$

$$\langle s, p, h(o) \rangle \in H \Rightarrow \langle s, p, o \rangle \in G \quad (\text{O})$$

- Prevents G from interfering, to different degree, with H
- Bounds can be combined conjunctively and/or disjunctively
- 20 different non-equivalent bounds

Examples

- Do not add more superclasses to my dataset (but new subclasses allowed)

$\langle n, P, a \rangle$ and $P = \text{rdfs:subClassOf}$

Prevents ontology-hijacking, and protects vocabulary

- New `foaf:knows` relationships must relate new persons

$\langle n, P, n \rangle$ and $P = \text{foaf:knows}$

Keeping amendments separated from receiver's data

- Write-protecting (parts of) a dataset

\top and, e.g., $P = \text{foaf:knows}$

- Adding new data must not re-arrange existing data

$\langle n, a, a \rangle$

$\langle a, n, a \rangle$

$\langle a, a, n \rangle$

Avoids hampering with the receiver's data

Features and “Features”

- Implements “conservative extensions” (CE) for RDF graphs
 - $CE \approx$ “No new statements allowed with old language only”
 - Weakest non-trivial bound ensures syntactic CE
- Works currently only for single triple statements
 - OWL LD (Linked Data) = OWL RL \cap single triple statements
- Composition
 - Combine (partial) bounds to new bounds
New bound as strong as the infimum/meet

- Generic constraints on graphs
- Restrictions
 - Restrict protection to subset of receiver:
S-, P-, O-values; -namespaces and/or -classes
- Exceptions
 - Ignore violating sources, i.e., abort
 - Ignore violating source
 - Ignore violating triples
- Specify Data exchanges, Payloads, Validation results
- Publish, re-use and combine bounds
- Prototype implementation: checks bounds, computes payloads
- Complexity: P

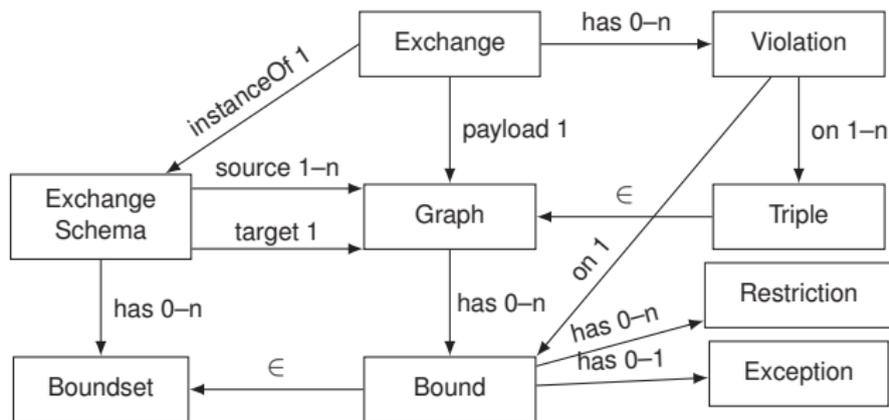
Summary:

- Constraints relative to receiving dataset
 - Motivation
 - Theory
 - RDF Vocabulary

Further reading:

- <http://sws.ifi.uio.no/project/boundz/>
- Paper accepted for ISWC workshop COL2013

Boundz vocabulary



Simplified overview of:

<http://sws.ifi.uio.no/vocab/boundz>

BBC Music dataset restrictions:

1. The vocabulary that the BBC uses must not be hijacked by adding new superclasses or superproperties.
2. Adding new `foaf:made` relationships is not tolerated, unless both artist and record is new to the BBC dataset; their current library is regarded as complete with respect to the albums of enlisted artists, but is open for extensions with new artists.
3. More fanpages may be added, but an existing fanpage cannot be related to more artists.
4. No new information about existing genres may be added.
5. Also, assume the BBC keeps a special dataset about the Beatles which is not under their management, so they want to disallow any new information using only elements from this dataset. However, new information may *relate* to the Beatles dataset.

```
1  ex:bbcmusic a bz:BoundedGraph ;
2    bz:hasBound bz:RDFS ,
3      [ a bz:Aso ; bz:predicateValue foaf:made ] ,
4      [ a bz:o ; bz:predicateValue mo:fanpage ;
5        bz:hasException bz:ignoreViolations ] ,
6      [ a bz:T ; bz:subjectClass mo:Genre ] ,
7      [ a bz:T ; bz:objectClass mo:Genre ] .
8  ex:beatles a bz:BoundedGraph ;
9    bz:hasBound [ a bz:KKspo ] .
```

Boundz output example

```
1 <file:///test/test>
2     a          :ExchangeSchema ;
3     :hasBound _:b1 , _:b2 , _:b3 , _:b4 , _:b5 , _:b6 , _:b7 , _:b8 ;
4     :hasSource <file:///test/uni1_0.ttl> , <file:///test/uni1_1.ttl> ;
5     :hasTarget <file:///test/uni500_2.ttl> ;
6     :outputPayload "false"^^xsd:boolean ;
7     :outputViolations "false"^^xsd:boolean ;
8     :sourceReasoning "false"^^xsd:boolean ;
9     :targetReasoning "false"^^xsd:boolean .
10
11 <http://test/test/1372168337206/31381>
12     a          :Exchange ;
13     :hasPayload [ :noOfTriples "6498"^^xsd:long ] ;
14     :hasViolation [ a          :Violation ;
15                     :hasSource <file:///test/uni1_1.ttl> ;
16                     :noOfTriples "566"^^xsd:long ;
17                     :onBound _:b7 ] ;
18     :instanceOf <file:///test/test> ;
19     :timestamp "1372168337206"^^xsd:long .
20
21 _:b7 a          :T ;
22     :classRestriction ub:University ;
23     :hasException :ignoreViolations ;
24     :hasRestriction ub:University ;
25     :objectClass ub:University ;
26     :objectRestriction ub:University .
```

Excerpt of <http://sws.ifi.uio.no/project/boundz/impl/outputExchanges.ttl>

SPARQL representation of bz:KKspo:

```
1 CONSTRUCT
2   { ?s ?p ?o .}
3 WHERE
4   { GRAPH <SOURCE>
5     { ?s ?p ?o
6       GRAPH <TARGET>
7         { { { { ?s ?_5 _:b0 } UNION { _:b1 ?s _:b2 } UNION { _:b3 ?_6 ?s } }
8           { { ?p ?_3 _:b4 } UNION { _:b5 ?p _:b6 } UNION { _:b7 ?_4 ?p } } }
9         { { ?o ?_1 _:b8 } UNION { _:b9 ?o _:b10 } UNION { _:b11 ?_2 ?o } }
10        }
11      }
12  MINUS
13    { GRAPH <TARGET>
14      { ?s ?p ?o }
15    }
16 }
```

- Aidan Hogan, Andreas Harth, and Axel Polleres. “Scalable Authoritative OWL Reasoning for the Web”. In: *International Journal on Semantic Web and Information Systems* 5.2 (2009), pp. 49–90
- Audun Stolpe and Martin G. Skjæveland. “Preserving Information Content in RDF Using Bounded Homomorphisms”. In: *ESWC*. Ed. by Elena Simperl et al. Vol. 7295. LNCS. Springer, 2012, pp. 72–86
- Birte Glimm et al. “OWL: Yet to arrive on the Web of Data?” In: *Proc. of the WWW2012 Workshop on Linked Data on the Web (LDOW 2012)*. 2012