

The Wheel and the Hub

A simple and empty way
towards semantic integration

This version : 2005-12-27

Previous version : http://www.mondeca.com/lab/bernard/hubjects_v1.pdf

Latest version : <http://www.mondeca.com/lab/bernard/hubjects.pdf>

Editor : Bernard Vatant

Background image : <http://www.bigfoto.com/miscellaneous/photos-11/cart-wheel-photo.jpg>



Prologue

Thirty spokes share the wheel's hub;
It is the center hole that makes it useful.

Shape clay into a vessel;
It is the space within that makes it useful.

Cut doors and windows for a room;
It is the holes which make it useful.

Therefore profit comes from what is there;
Usefulness from what is not there.

Tao Te Ching, Chapter 11

Translation source : <http://www.nokama.com/tao/index.cfm?fuseaction=chapter&ch=11>



Subjects are what the conversation is about

- Wordnet 2.0
 - The subject matter of a conversation or discussion.
- Dictionary.reference.com
 - Logic: The term of a proposition about which something is affirmed or denied.
 - Philosophy: The essential nature or substance of something as distinguished from its attributes.
- Topic Maps Reference Model
 - Anything whatsoever, regardless of whether it exists or has any other specific characteristics, about which anything whatsoever may be indicated by any means whatsoever.



Subjects are known through descriptions

- Unformal descriptions enable human communication
 - Textual : Names, descriptions, definitions ...
 - Multimedia : Pictures, icons, sounds, video ...
- Formal descriptions enable computer management
- The type of descriptions depends on the framework used
 - In RDF they are 'Resources'
 - In OWL they are 'Things'
 - In Topic Maps they are 'Topics'
 - In OO Programming they are 'Objects'
 - In Conceptual Graphs they are 'Concepts'
 - In Thesauri they are 'Terms'
 - ...



Descriptions are defined in perspectives

- A perspective provides rules used to handle descriptions
 - Type of descriptions
 - Logical structure of descriptions
 - Rules of use

- A perspective generally includes
 - A description framework
 - Language, logical rules, protocols, specific tools ...
 - Example : RDF, Topic Maps, Conceptual Graphs, UML ...
 - A description scheme
 - Structured and hopefully consistent set of descriptions, generally defined using a single consistent framework
 - Example : Taxonomy, Thesaurus, Ontology ...

- Perspectives are often implicit ...



Identification is perspective-specific

- Identity is conferred through descriptions ...
 - Assignment of identifying properties
 - Identification protocols
- Therefore identification is perspective-specific ...
 - Identification uses rules defined by the perspective
- Therefore subjects have no absolute identity ...



So what about semantic integration?

- Using the same perspective
 - Identification, mapping, merging, consistency checking, inference are possible
 - Shared notion of true/false
 - Shared axioms and inference rules
 - Shared identification process

- Across different perspectives
 - Using different frameworks and schemes
 - Identification, mapping, merging, consistency checking, inference, are real challenges
 - Unless a meta-perspective is provided ...
 - Or specific rules of mapping are defined ...



So why not define a single meta-perspective ?

- Because it 's a silly, endless and childish game
 - A: My perspective is more meta than yours.
 - B: No, MINE is more meta than yours !
 - C: MINE is more meta than both of you !!
 - D: Prove it, but use MY perspective, please !!!
- People have played that game for centuries
 - With no visible results except religion wars



Semantic diversity is Life

- No description is exhaustive
 - A description only provides a limited set of properties which are useful and possible to express in a given perspective
- All perspectives have limited expressivity
 - Bound to their specific language and logical rules
 - Relative to their specific purposes
- Anybody can describe anything anyway ...
 - So we 're bound to live with various, inconsistent descriptions of the 'same' subject
 - But we would like to be able to assess this 'same-ness' and use it efficiently across different perspectives like in natural language and thought



Re-discovering the Wheel ...

■ Hub

- The central piece of a wheel, binding spokes together
- A center of activity or interest or commerce or transportation; a focal point around which events revolve
- To hub : to connect through a hub



Characteristics of hubs

- Hubs are neutral binding points

Thirty spokes share the wheel's hub;

- Hubs are hollow

It is the center hole that makes it useful.

*Therefore profit comes from what is there;
Usefulness from what is not there.*



Hubs in RDF : blank empty nodes

- Subjects are hubs
 - For different descriptions pointing at them
 - From the same or different perspectives
- Subjects are hollow
 - They have neither type, nor property whatsoever
 - They do not provide any more semantics than the descriptions pointing at them
 - Various descriptions of the same subject might bear conflicting semantics
- Their representation in RDF should be the same
 - Blank empty nodes are natural candidates
 - No absolute identity, no name, no type, no property whatsoever ...



Introducing SPEK vocabulary

- A subject is defined by a blank empty node
- Descriptions point to this blank empty node
 - Using the property 'spek:hsubject'
- Each description can explicit the perspective it uses
 - Property 'spek:viewpoint'
 - Class 'spek:Perspective'



SPEK example : triples

- Two descriptions of the 'same subject'
 - Concept defined in a SKOS Concept Scheme
 - eg:Human
 - Class in a RDFS ontology
 - foaf:Person

scheme:eg
eg:Human
eg:Human

rdf:type
spek:viewpoint
spek:hubject

spek:Perspective
scheme:eg
_:b0

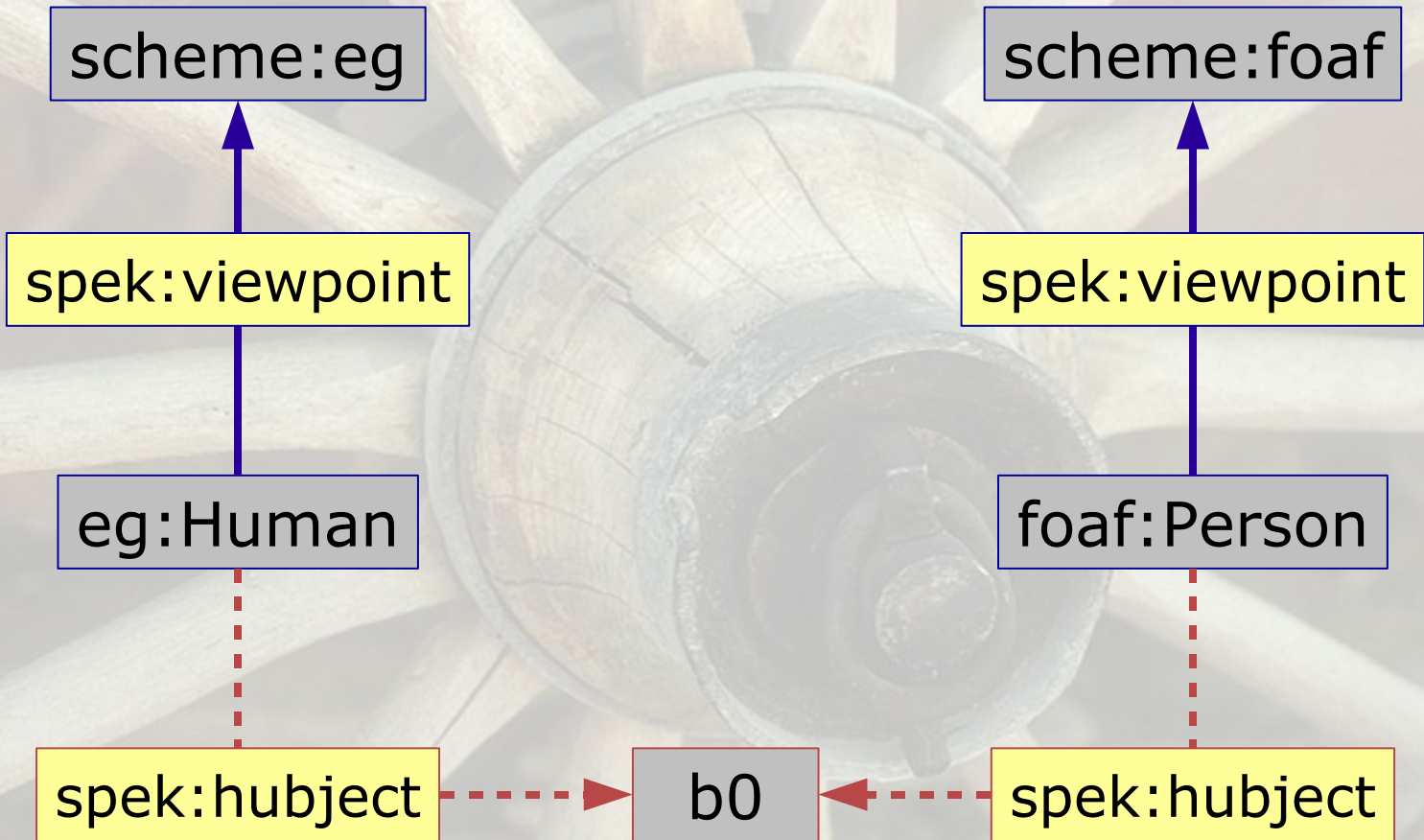
scheme:foaf
foaf:Person
foaf:Person

rdf:type
spek:viewpoint
spek:hubject

spek:Perspective
scheme:foaf
_:b0



SPEK graph



Further reading

- Specification of SPEK vocabulary with more examples
 - <http://www.mondeca.com/lab/bernard/spek.rdf>
- More about subject identification
 - <http://universimmedia.blogspot.com/>

