

FOUST II – 2nd Workshop on Foundational Ontologies

Pizza & Wine: The need for educational tools for foundational ontologies

Stefan Schulz

Institute for Medical Informatics, Statistics and
Documentation, Medical University of Graz, Austria

Martin Boeker

Faculty of Medicine and Medical Center,
University of Freiburg, Germany

José A. Vera Ramos

Faculty of Biology, University of Murcia, Spain

Ludger Jansen

Institute of Philosophy, University of Rostock, Germany

Ontology education vs. Foundational Ontologies

- "Popular" educational ontologies (EOs) and tutorials ignore foundational ontologies (FOs)



Ontology education vs. Foundational Ontologies

- "Popular" educational ontologies (EOs) and tutorials ignore foundational ontologies (FOs)
- Questions on ontology education
 - Educational goals
 - Typical learners
 - The role FOs should play in OE
- Can popular EOs be "modernized"?
 - to meet with FO requirements
 - to enforce the understanding of FOs
 - to meet other ontology quality requirements

Material

PIZZA

- Created 2004
- DAML+OIL → OWL
- Manchester training courses
- Pizzas and pizza ingredients and qualities

WINE&FOOD

- Created 2001 (1991)
- CLASSIC → FRAMES → OWL
- OWL + Protégé tutorials
- Food, wines, their origins and meals constituted by them

Material

PIZZA

- Created 2004
- DAML+OIL → OWL
- Manchester training courses
- Pizzas and pizza ingredients and qualities
- 100 classes, 8 object properties, 6 individuals
- 259 subclass axioms
- 15 equivalent class axioms
- Expressivity *SHOIN*

WINE&FOOD

- Created 2001 (1991)
- CLASSIC → FRAMES → OWL
- OWL + Protégé tutorials
- Food, wines, their origins and meals constituted by them
- 137 classes, 16 object properties, 194 individuals
- 228 subclass axioms
- 87 equivalence class axioms
- Expressivity *SHOIN(D)*

PIZZA

The screenshot displays the Protege ontology editor interface for the 'Pizza' ontology. The left-hand pane shows a hierarchical tree of classes, starting from `owl:Thing` and descending through `DomainConcept`, `Country`, `Food`, `IceCream`, and `Pizza`. Under `Pizza`, there are subclasses like `CheeseyPizza`, `InterestingPizza`, `MeatyPizza`, and `NamedPizza`. `NamedPizza` further branches into `American`, `AmericanHot`, `Cajun`, `Capricciosa`, `Caprina`, `Fiorentina`, `FourSeasons`, `FruttiDiMare` (highlighted), `Giardiniera`, `LaReine`, `Margherita`, `Mushroom`, `Napoletana`, and `Parmense`.

The right-hand pane shows the details for the selected class, `FruttiDiMare`. It includes a list of annotations (currently empty), a description (currently empty), and a list of class axioms. The axioms include:

- `Equivalent To` (empty)
- `SubClass Of` (empty)
- `hasTopping only (GarlicTopping or MixedSeafoodTopping or TomatoTopping)`
- `hasTopping some GarlicTopping`
- `hasTopping some MixedSeafoodTopping`
- `hasTopping some TomatoTopping`
- `NamedPizza`

Below these, there are sections for 'General class axioms', 'SubClass Of (Anonymous Ancestor)', 'Instances', 'Target for Key', and 'Disjoint With'. The 'SubClass Of (Anonymous Ancestor)' section shows the axiom `hasBase some PizzaBase`.

The bottom-left pane shows the 'Object property hierarchy' with properties like `hasCountryOfOrigin`, `hasIngredient`, `hasSpiciness`, `isIngredientOf`, `isBaseOf`, and `isToppingOf`.

<https://protege.stanford.edu/ontologies/pizza/pizza.owl>

WINE&FOOD

owl:Thing

- ConsumableThing
 - EdibleThing
 - Dessert
 - Fowl
 - Meat
 - NonSweetFruit
 - OtherTomatoBasedFood
 - Pasta
 - Seafood
 - SweetFruit
 - Meal
 - MealCourse
 - BlandFishCourse
 - CheeseNutsDessertCourse
 - DarkMeatFowlCourse**
 - DessertCourse
 - FishCourse
 - FruitCourse
 - LightMeatFowlCourse
 - NonBlandFishCourse
 - NonOysterShellfishCourse
 - NonRedMeatCourse
 - NonSpicyRedMeatCourse

Annotation property hierarchy | Datatypes

Data property hierarchy | Individuals by type

Object property hierarchy:

- owl:topObjectProperty
 - adjacentRegion
 - course
 - hasDrink
 - hasFood
 - hasMaker

Annotations: DarkMeatFowlCourse

Description: DarkMeatFowlCourse

Equivalent To

- MealCourse and (hasFood only DarkMeatFowl)

SubClass Of

- hasDrink only (hasBody value Light)
- hasDrink only (hasColor value Red)
- hasDrink only (hasFlavor value Delicate)
- hasDrink only (hasSugar value Dry)

General class axioms

SubClass Of (Anonymous Ancestor)

- hasFood min 1 owl:Thing
- hasDrink min 1 owl:Thing
- hasDrink only Wine

Instances

Target for Key

Our desiderata for good EOs

- General
 - Demonstrate guidelines
 - Explicate design patterns
 - Demonstrate naming conventions
 - Provide good documentation
 - Show limitations of what ontologies can /should express
- Domain
 - Understandable, common sense, attractive
 - Exemplify broad range of ontological categories (time, space, qualities, realizables) and formal relations (spatial, temporal, participation, inheritance,...)

Foundational ontologies in EO

- FOs
 - EOs to be built under a FO
 - Make use of FO plausible
 - User-friendly and intuitive FO, providing exhaustive set of upper level classes and properties
 - Demonstrate how FO-centred ontology design facilitates modelling / interoperability / reusability
- Users
 - In the first place: domain experts, standards developers
 - In second place: ontologists

Scrutiny of PIZZA and WINE&FOOD against desiderata for educational ontologies

Some Shortcomings of PIZZA

- No reference to any FO, top level bipartition into "Domain concepts" and "Value partitions"
- Non-rigid classes, e.g. *P:Food*
- Idiosyncratic classes and properties:
 - *P:Country* extensionally defined by exactly five countries
 - Unclear meaning of **P:hasCountryOfOrigin**.
- Imprecise naming
 - *P:TabascoPepperSauce* under *P:PizzaTopping*

Some Shortcomings of WINE&FOOD

- No FO, top level: consumable, non-consumables, regions, vintages, wineries, wine descriptors
- Highly specific object properties: **W:madeFromGrape**
- Labelling issues
 - *W:Loire* (Wine, Region, River)?
 - *W:MealCourse* (does not allow non-alcoholic meals)
- Unprincipled instance / class division:
 - class *W:Chianti*, individual **W:ChiantiClassico**
- No metadata, no text definitions
- Tutorial uses frame terminology (“concept”, “slot”)

Redesign

- $P: \rightarrow P^+:$ $W: \rightarrow W^+:$
- Goal: merged ontology $PW:$
- Based on
 - GoodOD guideline
 - BTL2 (BioTopLite version 2)
upper level ontology
(import btl2:)
- Attempts:
 - everything under BTL2 toplevel
 - no extension of R box

Guideline on Developing Good Ontologies in the Biomedical Domain with Description Logics

URL: <http://www.purl.org/goodod/guideline>

Version 1.0
December 2012

Send feedback to:
martin.boeker@uniklinik-freiburg.de
ludger.jansen@uni-rostock.de

Schulz S^{1,3}, Seddig-Raufie D¹, Grewe N², Röhl J²,
Schober D¹, Boeker M¹, Jansen L²

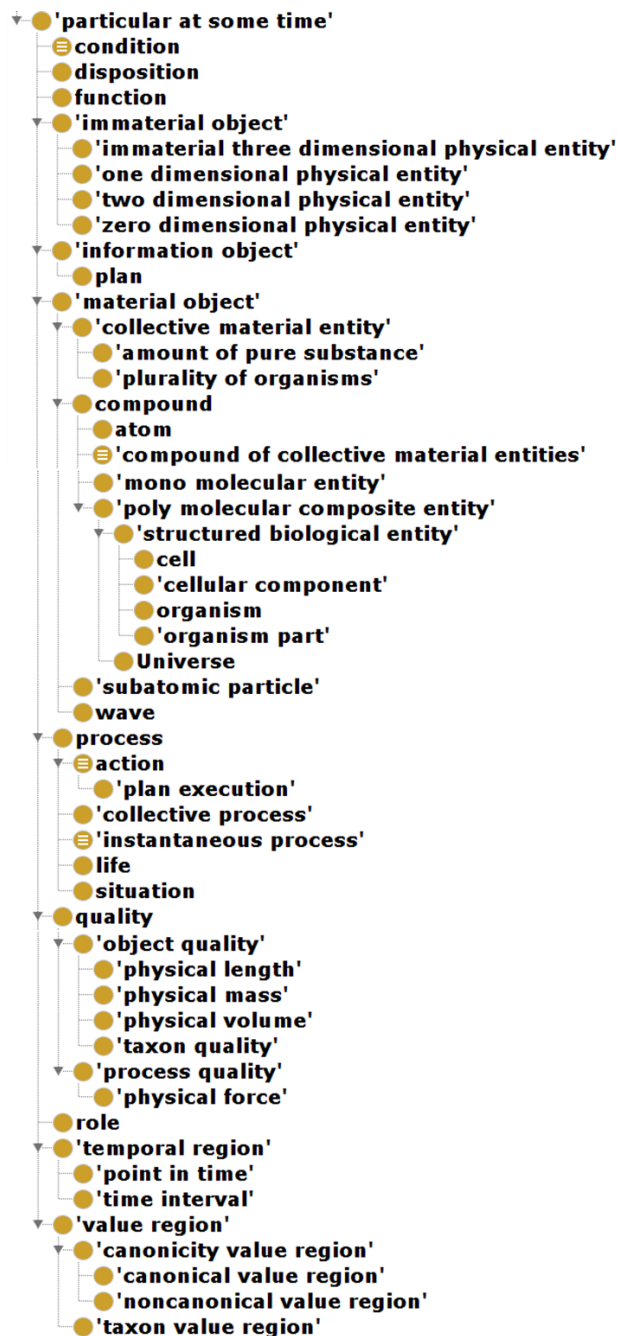
¹: Institute of Medical Biometry and Medical Informatics,
University Medical Center Freiburg

²: Institute of Philosophy, University of Rostock

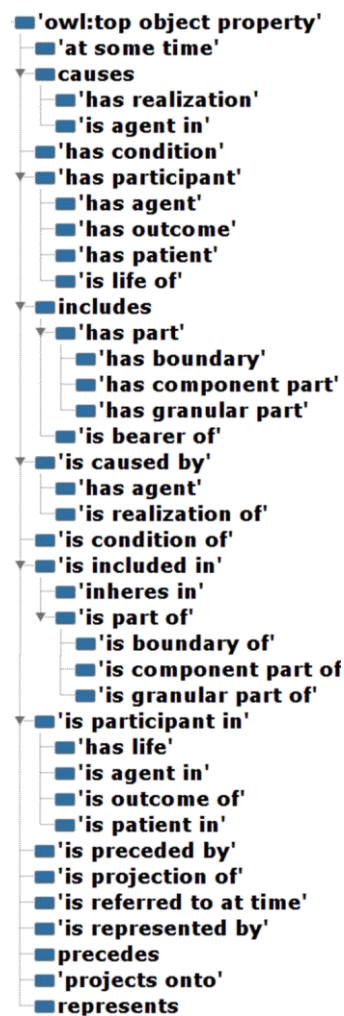
³: Department of Medical Informatics, University of Graz

11th December 2012

BTL2 Classes



BTL2 Relations



BTL Axioms (examples)

Description: MaterialObject
Equivalent To +
SubClass Of +
- atSomeTime only MaterialObject - hasBoundary only TwoDimensionalPhysicalEntity - hasGranularPart only MaterialObject - hasLife only Life - hasLife some Life - hasPart only (ImmaterialObject or MaterialObject) - hasPart some MaterialObject - hasPart some SubAtomicParticle - isBearerOf only (Disposition or Function or InformationObject or ObjectQuality or Role) - isBearerOf some PhysicalMass - isBearerOf some PhysicalVolume - isParticipantIn only Process - Particular - projectsOnto only ImmaterialThreeDimensionalPhysicalEntity - projectsOnto some ImmaterialThreeDimensionalPhysicalEntity
Description: InformationObject
Equivalent To +
SubClass Of +
- atSomeTime only InformationObject - Disposition or Function or InformationObject or ObjectQuality or Role - hasLife only Life - hasLife some Life - hasPart only InformationObject - hasRealization only Process - inheresIn some MaterialObject - isIncludedIn only (InformationObject or MaterialObject) - isParticipantIn only Process - isPartOf only InformationObject not (isBearerOf some PhysicalMass) not (isBearerOf some PhysicalVolume) - Particular

Example P / P⁺

PIZZA

P:AmericanPizza subClassOf

P:NamedPizza and **P:hasTopping** only (*P:MozzarellaTopping* or *P:PeperoniSausageTopping* or *P:TomatoTopping*)

P:AmericanPizza subClassOf *P:NamedPizza* and

P:hasTopping some *P:MozzarellaTopping* and

P:hasTopping some *P:PeperoniSausageTopping* and

P:hasTopping some *P:TomatoTopping*

PIZZA⁺

P⁺:AmericanPizza subClassOf *P⁺:NamedPizza* and

(btl2:hasComponentPart only (*P⁺:MozzarellaTopping* or *P⁺:PeperoniSausageTopping* or *P⁺:TomatoTopping* or *P⁺:PizzaBase*)) and

(btl2:hasComponentPart some *P⁺:TomatoTopping*) and

(btl2:hasComponentPart some *P⁺:MozzarellaTopping*) and

(btl2:hasComponentPart some *P⁺:PeperoniSausageTopping*)

Example P / P⁺

PIZZA

P:American subclassOf **P:hasCountryOfOrigin** value **P:America**

PIZZA⁺

P⁺:PreparingPizza EquivalentTo *P⁺:PreparingFoodAction* and **bt12:hasOutcome** some *P⁺:Pizza*

P⁺:AmericanPizzaRecipe EquivalentTo *P⁺:PizzaRecipe* and (**bt12:isOutcomeOf** some (*P⁺:CreatingRecipe* and (**bt12:isIncludedIn** value **P⁺:NorthAmerica**))))

P⁺:AmericanPizza subclassOf *P⁺:Pizza* and
bt12:isOutcomeOf some (*P⁺:PreparingPizza* and (**bt12:isRealizationOf** some *P⁺:AmericanPizzaRecipe*)))

Example W / W⁺

WINE

W:Beaujolais subclassOf **W:madeFromGrape** value **W:GamayGrape**

WINE⁺

W⁺:WineMaking equivalentTo *btl2:Action* and
btl2:hasOutcome some *W⁺:Wine* and **btl2:hasPatient** some *W⁺:Grape*

W⁺:BeaujolaisWine subclassOf *W⁺:Wine* and
(**btl2:isOutcomeOf** some
W⁺:WineMaking and **btl2:hasPatient** some *W⁺:GamayGrape*)

Discussion: Ontology education – which principles to subscribe to?

- Design of ontologies: two tendencies
 1. "reality representation": assumed consensus about reality as only criterion for design decisions (Smith, Ceusters)
 2. "conceptualization": a domain is represented in a way a concrete application requires (Noy, Gruber, McGuinness)
- 1. blinds out conflicting views / perceptions of reality
- 2. obviates interoperability and notion of ontology artefacts as standards
- Educational ontologies should start with a domain where consensus can be assumed
 - Tangible objects: pizzas, food items, wine
 - More difficult with social entities, cognitive entities, informational entities, dispositions etc.

Discussion: Pros and cons of P^+ / W^+ for ontology education

- Understanding description logics
 - 😐 Imposition of BTL2 narrows down breadth of modelling tasks, e.g. CGIs, algebraic properties of object properties, punning
 - 😐 Neither P/W nor P^+/W^+ include examples for datatype properties or advanced Abox reasoning
- Understanding ontology
 - 😊 Focus on FO understanding and use
 - 😊 Demonstration of validation by reasoners
 - 😊 Demonstration of ontology merging
 - 😊 Avoidance of trivial modelling workarounds

Outlook

- P^+/W^+ soon merged into a single ontology
 - Current efforts documented at purl.org/biotop
 - New tutorial to be written, with focus on good practice, naming, foundational ontologies
 - Assessment against competency questions
 - Reasoning benchmarks
- Future work:
 - Enhancing P^+/W^+ by more ontological categories (dispositions, processes, social entities,...)
 - More Abox reasoning and concrete domains (datatype property examples)

Thanks

- Contact:

- Stefan Schulz, Medical University of Graz:
stefan.schulz@medunigraz.at

- Cooperation welcome!

- Acknowledgements (BTL2):

Elena Beißwanger, Catalina Martínez-Costa, Udo Hahn, Filipe Santana da Silva, Daniel Schober, Holger Stenzhorn, Gustavo Uribe