A Project Management Ontology

Roberto Casadei
April, 2014

Course: Semantic Web
Computer Science and Engineering LM
Alma Mater Studiorum – Università di Bologna

Summary

1) WHAT and WHY: the Domain of Interest

2) The Approach

3) Evaluating Possibilities for Reuse

4) Domain Modelling

5) Development of the Ontology

6) Validation

7) Usage

WHAT: a project management ontology

An **ontology** is a

formal,

explicit specification

of a shared

conceptualization

Project management is the application of knowledge, skills, tools, and techniques to project activities to meet the project requirements [5]

My Project Management Ontology (MPMO): a (domain-specific) ontology for project management

- Defining **concepts** common to the majority of project management efforts
- Defining **relationships** among those concepts
- Specifying a **well-defined semantics** for both concepts and properties

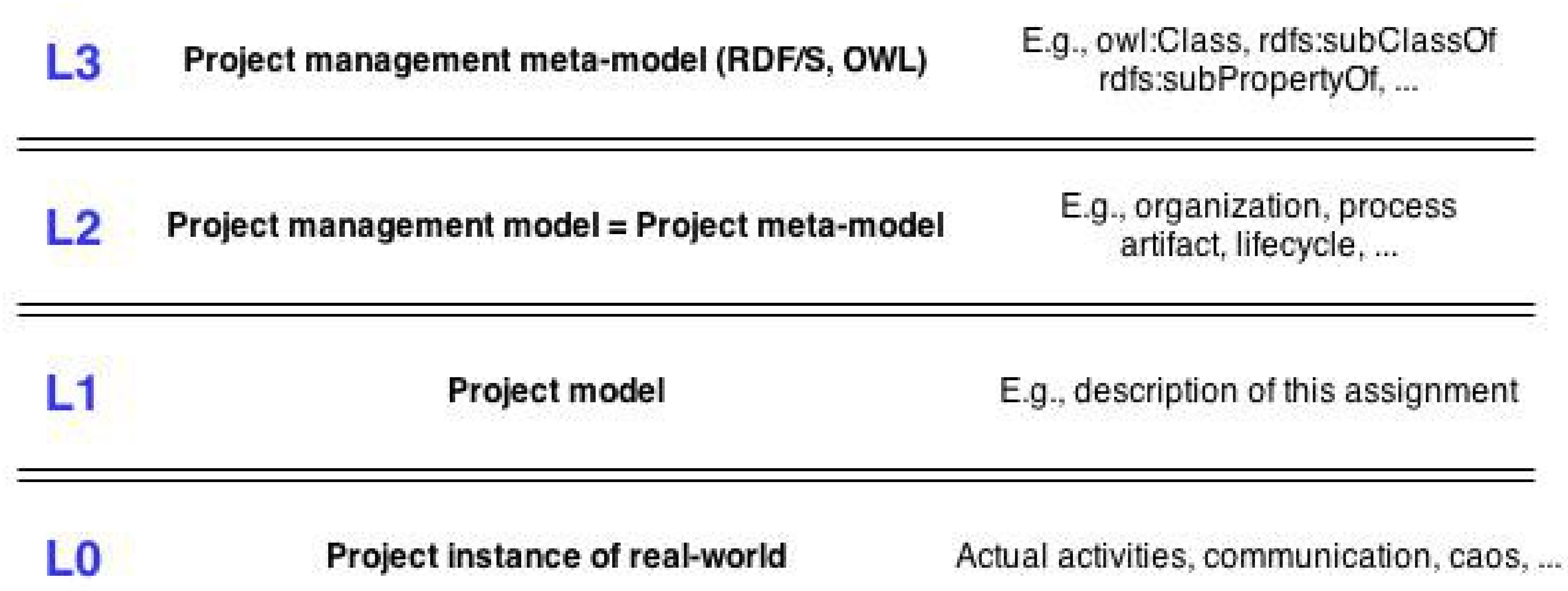
using **RDF** and **RDF Schema**

WHY: the driving forces

The expected value of such an ontology

- Providing the basis for infrastructures which support an **effective management of project data**
  - E.g., for documentation or monitoring purposes
  - E.g., keeping track of events, supporting traceability, etc.
- Using a **common format for project management data**
- **Supporting the implementation of project management software**
- Why not simply using standardized data models? [1]
  - ''While a standardized data model is good for the exchange of project data between PM-Software, it can also help to build a **common understanding of terms and definitions** in the field of project management, thus fostering interoperability not only for the exchange of data but also on a business process and organizational level''
  - ''A project management ontology is a valuable extension for the representation of project management data since it is **semantically more powerful than a data model** with explanatory text'

Model layers



L3	Project management meta-model (RDF/S, OWL)	E.g., owl:Class, rdfs:subClassOf rdfs:subPropertyOf, ...
L2	Project management model = Project meta-model	E.g., organization, process artifact, lifecycle, ...
L1	Project model	E.g., description of this assignment
L0	Project instance of real-world	Actual activities, communication, caos, ...

The approach: goals and desired properties

Goals

- **Producing a model for the project management domain** which supports the properties defined below
- **Producing an ontology using RDF and RDF Schema** which exhibits the properties defined below
- **Producing a model for a project management effort using the ontology**

Desired properties for the PM ontology

- Generality
- Simplicity
- Flexibility & Extensibility

Towards domain modelling: analysis

Some core concepts come to mind

- Organization
- Project
- Project lifecycle
- Process
- Team
- Stakeholder
- ...

What do these terms (informally) mean?

- The PMBOK (Project Management Body Of Knowledge) as a reference

The matter is complex... have others faced similar issues?

Existing ontologies related to project-management

- **Description of a Project (DoaP)**
  - An XML/RDF vocabulary to describe open source and other software projects
  - https://github.com/edumbill/doap/wiki
- **Project Documents Ontology (PDO)**
  - Models the inherent structure and concepts of various documents in a project-specific setting, like meeting minutes, status reports etc.
  - http://vocab.deri.ie/pdo#
- **SUPER (Semantics Utilised for Process management within and between EnteRprises)**
  - Focused on BPM
  - http://www.ip-super.org/
- **Software Process Control Model**
  - http://spi-fm.uca.es/spdef/models/deployment/spcm/1.0#
- **Another Project Management Ontology (APMO)**
  - https://code.google.com/p/apmo/
- **FP research project ontologies**
  - http://mayor2.dia.fi.upm.es/oeg-upm/index.php/en/ontologies/81-research-proj-ontologies
- **PROMONT ontology**
  - A project management ontology as a reference for virtual project organizations
  - Only publications?
- …

Reusing ontologies

Project management is about concepts... → The SKOS ontology

...and about how orgs execute projects... → The ORG ontology

...by driving people and resources... → The FoaF ontology

...in order to deliver some result and ultimately business value

- Other ontologies that will be reused include
  - The W3C Time Ontology (**OWL-Time**), e.g., in order to define time-bounded resource allocations

- The following ontologies are not considered at the general level, but may be in specializations
  - **DoaP** has a narrower scope (software projects, descriptive-level)
  - **FP Research project ontologies** (Documentation, Event, Organization, Person, Project) are very detailed but seem to have a slightly different focus
  - **SUPER ontologies** are part of a complex ecosystem focused on processes

  Moreover, some of them seem to be minor ontologies
  - **PDO** has a narrower scope with respect to deliverables (documents)
  - **Software Process Control Model** has a narrow scope and provides only some general classes
- Also, I'd like to be free in taking decisions about core concept modelling

The FoaF (Friend of a Friend) ontology

- [2], http://xmlns.com/foaf/0.1/
- FoaF is a project devoted to linking people and information using the Web

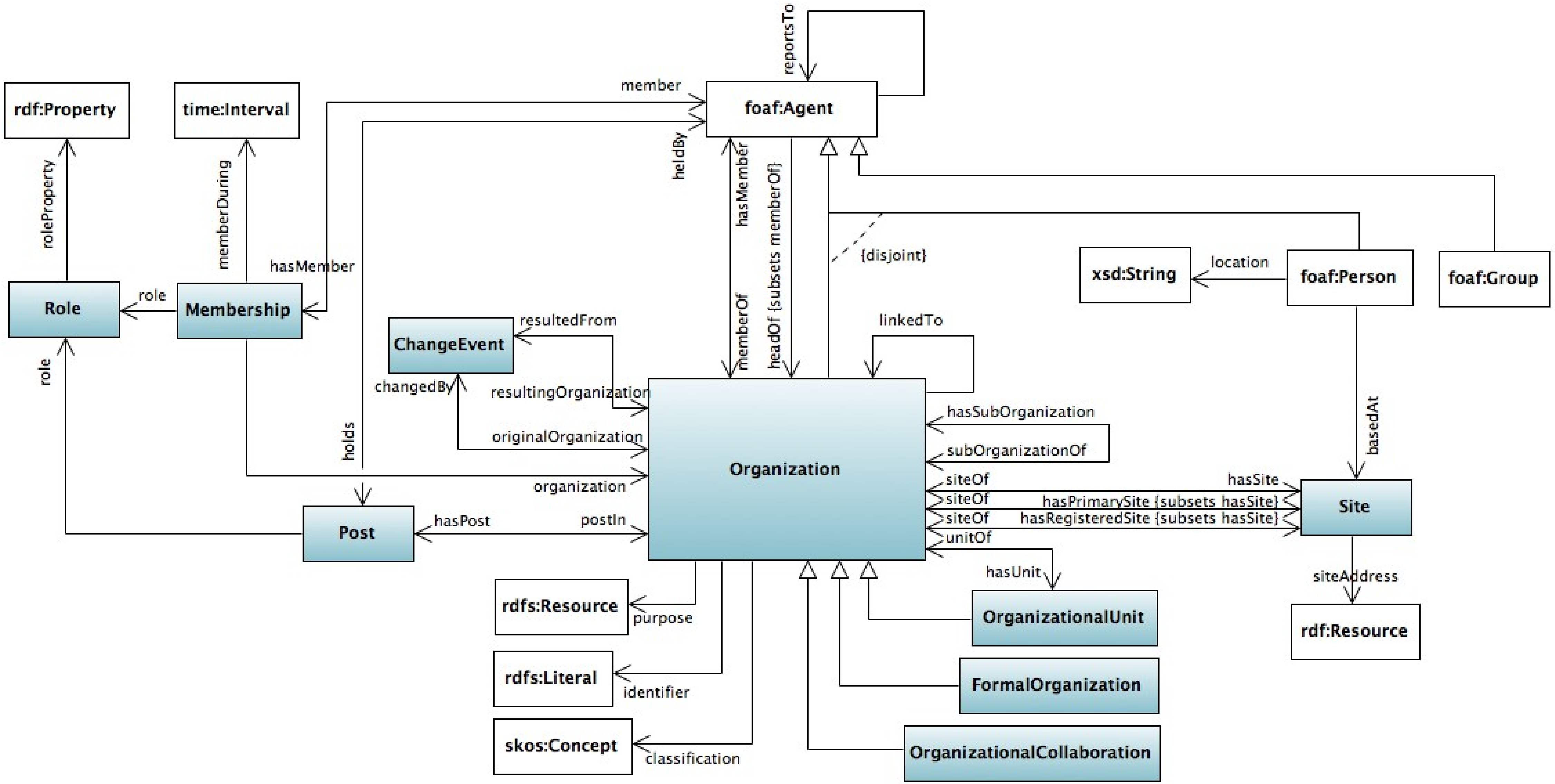
Elements of interest

- **foaf:Agent** class represents things that do stuff (e.g., person, sw or physical artifact)
  - Subclasses: **foaf:Person**, **foaf:Organization**, **foaf:Group**
  - May be a choice to represent project participants
    - What if I engage a robot in my next SW assignment?

The ORG ontology (1)

- [3], http://www.w3.org/ns/org#
- W3C Recommendation, January 2014
- Core ontology ''**for organizational structures**, aimed at supporting linked data publishing of organizational information **across a number of domains**''
  - So, it respects our requirement for generality
- ''It is **designed to allow domain-specific extensions** to add classification of organizations and roles, as well as extensions to support neighbouring information such as organizational activities''
  - So, it supports our requirement for flexibility and extensibility

The ORG ontology (2)



The ORG ontology (3)

Elements of interest

- **org:Organization**: represents a collection of people organized together into a community or other social, commercial or political structure
  - May be an **org:subOrganizationOf** another organization (which in turn **org:hasSubOrganization**)
  - Or may have more **org:OrganizationalUnit**s (which only have meaning within the context of the containing organization)
- Membership
  - Direct: an individual (represented as a **foaf:Agent**) is **org:memberOf** an organization
    - Subproperties of org:memberOf to represent specific roles that the person plays
  - Membership n-ary relationship: an **org:Membership** of an **org:member** in an **org:organization**, with possibly an **org:role** during **org:memberDuring**
- **org:Post** represents some position in the organization that may or may not be currently filled.
  - Posts enable reporting structures and organization charts to be represented independently of the individuals holding those posts
- **org:Site**s represent locations at which organizations exist
- When Organizations change substantially […] then the new Organization will typically be denoted by a new URI. In that case we need some vocabulary to describe that change over time and the relationship between the original and resulting resources. ORG provides **org:ChangeEvent** and associated properties as a foundation for this

The SKOS (Simple Knowledge Organization System) ontology

- [4], http://www.w3.org/2004/02/skos/core#
- SKOS is a common data model for KOSs such as thesauri, classification schemes, subject heading systems and taxonomies

Elements of interest

- A **skos:Concept** can be viewed as an idea or notion; a unit of thought. However, what constitutes a unit of thought is subjective, and this definition is meant to be suggestive, rather than restrictive
  - The notion of a SKOS concept is useful when describing the conceptual or intellectual structure of a knowledge organization system, and when referring to specific ideas or meanings established within a KOS
- A **skos:ConceptScheme** can be viewed as an aggregation of one or more SKOS concepts
  - Is disjoint with skos:Concept
  - Semantic relationships (links) between those concepts may also be viewed as part of a concept scheme
  - Related properties: **skos:inScheme, skos:hasTopConcept, skos:topConceptOf**
  - There are no conditions preventing a SKOS concept from taking part in zero, one, or more than one concept scheme

Domain modelling (1)

Core concepts

- **mpmo:PMModel**
  - Subclass of **skos:ConceptScheme**
- **mpmo:PMEffort**
- **mpmo:PMConcept**
  - Subclass of **skos:Concept**
- An mpmo:PMModel is a scheme of mpmo:PMConcepts
  - **skos:hasTopConcept**
- An mpmo:PMModel **mpmo:governsExecutionOf** an mpmo:PMEffort

Domain modelling (2)

Main project management concepts

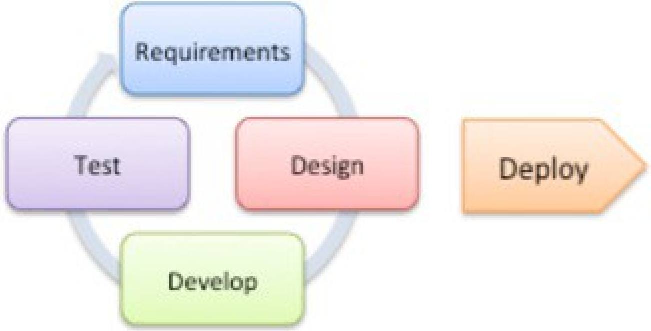
- They are classes (i.e., instances of *rdfs:Class*)
- They are instances of mpmo:PMConcept
- **mpmo:Process**: a set of interrelated actions and activities performed to create a pre-specified product, service, or result. Each process is characterized by its inputs, the tools and techniques that can be applied, and the resulting outputs [5]
  - Subclass of mpmo:PMEffort
- **mpmo:Project**: temporary endeavor undertaken to create a unique product, service, or result [5]
  - Subclass of mpmo:Process
- **mpmo:ValueResource**: anything expected to produce value through interaction with other entities
- **mpmo:Deliverable**: any unique and verifiable product, result or capability to perform a service that is required to be produced to complete a process, phase, or project [5]
  - Subclass of mpmo:ValueResource
- **mpmo:Organization**: a group of individuals organized to work for some purpose or mission
  - Subclass of org:Organization

Domain modelling (3)

- **mpmo:Stakeholder**: a person which has an interest in the project
  - Subclass of **foaf:Person**
- **mpmo:Participant**: an agent (which may be a person but not necessarily) which actively does something within a project
  - Subclass of **foaf:Agent**
- **mpmo:Role**: a role within a project
  - Subclass of **org:Role**
- **mpmo:Membership**: a membership relationship (agent, role, organization, project/process) within a project
  - Subclass of **org:Membership**
  - **mpmo:withinProcess** allows to specify that such membership relationship applies to given process

MPMO: ontology development

Iterative process



RDF and RDF Schema

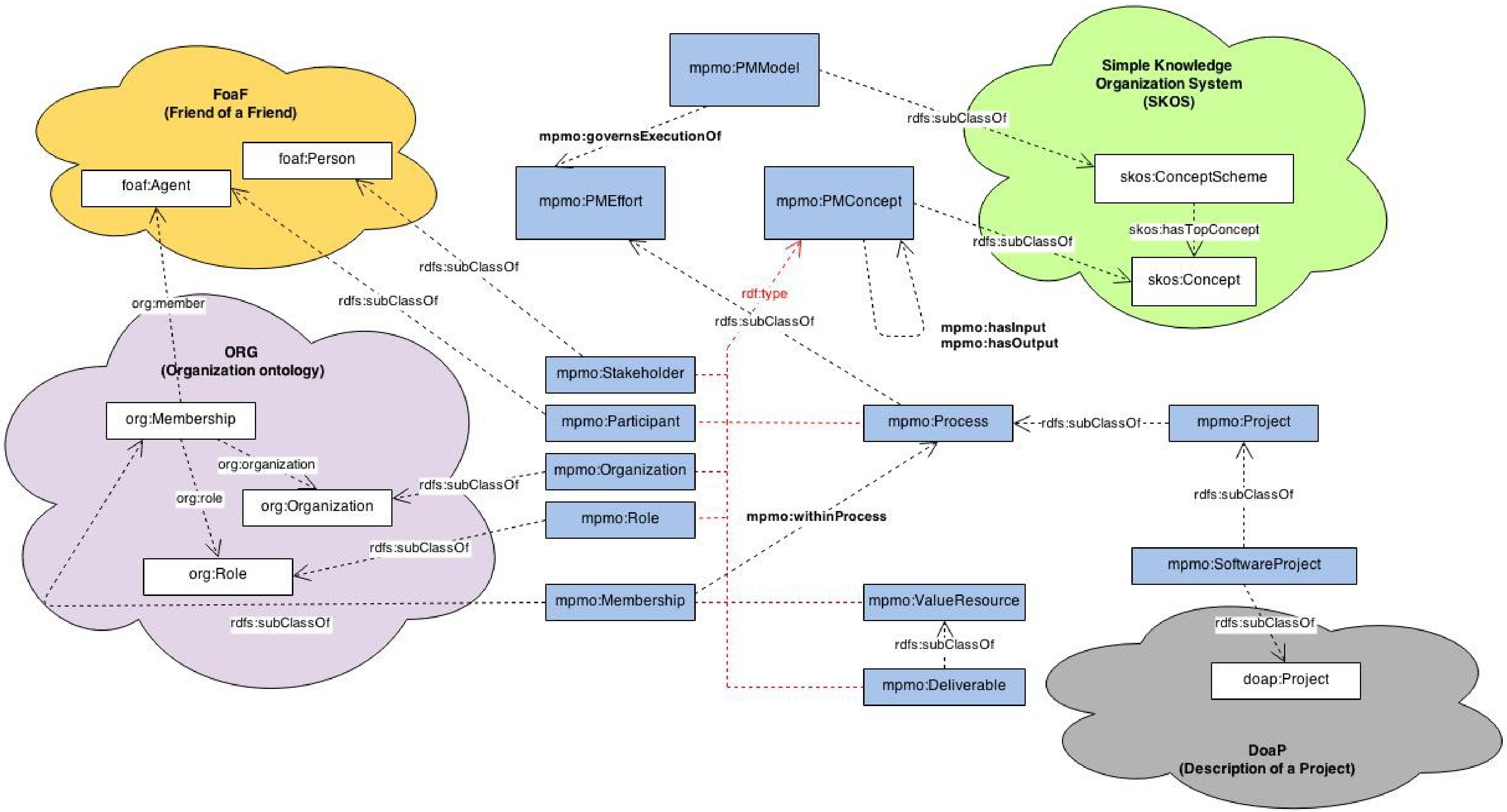
Modelling the domain

Validating the requirements

Validating the schema

The MPMO ontology

My Project Management Ontology (MPMO)



The ontology in use

Describing this assignment

In terms of

- Participants: me and the teacher
- Goals: what need to be delivered
- Approach: the project management model

Also, useful to generate feedback for the ontology modelling phase

- Especially with regards to the desired quality attributes

Conclusion

Lessons learned

- Modelling a domain may be very complex
- Building an ontology is an engineering process
  - A mature process is needed
  - Trade-offs
- The ontology is a living model

References

- [0]: Google & Wikipedia
- [1]: Sven Abels, Frederik Ahlemann, Axel Hahn, Kevin Hausmann, and Jan Strickmann. 2006. PROMONT – a project management ontology as a reference for virtual project organizations.
- [2]: http://xmlns.com/foaf/spec/
- [3]: http://www.w3.org/TR/2014/REC-vocab-org-20140116/
- [4]: http://www.w3.org/TR/skos-reference/
- [5]: 2004. A Guide to the Project Management Body of Knowledge (PMBOK Guides). Project Management Institute.
- [6]: https://github.com/edumbill/doap/wiki
- [7]: http://www.w3.org/TR/owl-time/