Ontologies in Requirement Engineering

Presented by Yetian Wang

July 4, 2017
RE Lifecycle and Activities

Fig. 1. Requirements Engineering Activities.

[Berry2004]  [Castaneda2010]
Problems in RE

- Factors that could lead to inadequate requirement elicitation [Castaneda2010]
  - Ambiguous requirement
  - Insufficient specifications
  - Requirements not completely defined, inconsistent
  - Dynamic and changing requirements

- RE needs a formal representation of the environment that deals with inconsistent, incomplete and evolving models [Nuseibeh2000].
Ontology and RE

- Ontologies can be used \cite{Lin1996}
  - Provides an formal, unambiguous and precise terminology
  - Are generic, reusable, and easy to extend
  - Enables automated reasoning that captures and stores traceability of, dependencies and relationships among requirements
  - Facilitates change management process

- In fact, an ontology is a sub-product of RE \cite{Breitman2003}
  - Particularity of a context of use of a system must be understood in detail before requirements can be derived.
What is an Ontology?

- “An ontology is a formal, explicit specification of a shared conceptualization” – Gruber, 1993

- An Ontology is the specification of:
  - Terminology, that specify the classes, properties and data types of the domain, and
  - Axioms, that define and constrain the interpretation of the terminology (in FOL, DL, OWL).
[Fox2016]
Ontology Languages [Fox2016]

- Declarative languages based on DL and FOL
  - Resource Description Framework (RDF/RDFS)
  - The Web Ontology Language (OWL)

- Description Logic is a family of formal knowledge representation languages.
  - Focus on defining classes and their properties.
  - Designed for efficient inference
    - Checking an instance satisfies a definition of a class it is a member of.
    - Checking if two concepts are related by a property, and no contradiction.
    - Subsumption: Determining whether one class is subsumed by another.

- First Order Logic
  - More expressive than Description Logic
  - Less efficient reasoning
Class Hierarchy [Fox2016]

- PrimaryStudent is a type of Student
  - PrimaryStudent $\subseteq$ Student
- SecondaryStudent is a type of Student
  - SecondaryStudent $\subseteq$ Student

Graph View [Fox2016]
Class Definition with Properties  

- A primary student is defined to be
  - the set of individuals that are “Students,”
  - attend “Public Primary School”, and
  - take “Primary Courses”.

\[
\text{Primary Student} \triangleq \\
\text{Student} \\
\Box \text{attends.}[\text{PrimarySchool} \sqcap \text{PublicSchool}] \\
\Box\forall \text{hasCourse. PrimaryCourse}
\]
Ontologies in RE

- According to Castaneda, et al (2010), three categories of ontologies in RE are identified [Castaneda2010]:
  - **Requirements Specification Document Ontology**
    - Describes structure of requirements specification documents
  - **Requirements Ontology**
    - Describes general requirement concepts such as FRs, NFRs, etc
  - **Application Domain Ontology**
    - Describes knowledge of specific application or domain
Scenario = ∃ has-goal.Goal ∧ ∃ has-goal.Goal

[Castaneda2010]
Requirements Specification Document Ontology

- Decker et al. [Decker2005] presented a wiki system for exchanging and reusing requirement specifications
  - Concludes that ontologies will be beneficial to this approach as the size of documents grow
  - Semantic Wiki
    - Wiki in RDF
    - Additional metadata to documents
- Groza et al. [Groza2007] described a solution for
  - Generating different representations of the same document based on metadata
    - Called “templates”
- Dragoni et al. [Dragoni2010] an ontological representation
  - Terms replaced with concepts
  - Calculate the importance of each concept
Requirements Ontology

- A requirement specification can be modelled using a generic ontology (e.g., upper level ontology)
- Can be used during elicitation process [Castaneda2010]
  - Reduces ambiguous requirements
  - Avoid incomplete requirements
  - Detect inconsistent requirements
A Requirement Ontology for Engineering Design

- Proposed a generic terminology with dependencies and relationships among requirement
  - It’s unambiguous, precise, reusable, and easy to extend
  - Able to detect redundant or conflicting requirements
  - Able to trace through relationships and revisions of requirements
- Explicit requirement, derived requirement, external, internal, etc.
- Axioms in FOL
- Implemented in Prolog
Decomposition

- \( r_1...r_n \) are decomposition of \( r \) (i.e., \textit{decomposition\_of}(r_1, r))..., the following is true.

\[
req(r_1) \land req(r_2) \land \ldots \land req(r_n) \rightarrow req(r)
\]

- However, the converse might not be true

\[
req(r) \rightarrow req(r_1) \land req(r_2) \land \ldots \land req(r_n)
\]
R

weight(Desk_spot_lamp) < 2

decomposition_of

R1
weight(Heavy_base) <= 1.3

R2
weight(Short_arm) <= 0.3

R3
weight(Small_head) <= 0.4

* Unit of measurement: pound

[Lin1996]
More rules

- Explicit Requirement
  \[(\forall r) \neg (\exists r') \text{decomposition_of}(r, r') \rightarrow \text{explicit}(r)\]

- Derived Requirement
  \[(\forall r) \text{derived}(r) \rightarrow (\exists r') \text{decomposition_of}(r, r')\]

- External Requirement
  \[(\forall r) \text{external}(r) \equiv (\exists a) \text{posted_by}(r, a) \land \text{customer}(a)\]
  \[(\forall r) \text{external}(r) \rightarrow \text{explicit}(r)\]

- Subsumption
  \[(\forall r_1 r_2) \text{subsume}(r_1, r_2) \equiv [\text{req}(r_1) \rightarrow \text{req}(r_2)]\]
Application Domain Ontology

- Contains knowledge of specific domain
- Relationship among domain concepts
- Can be used to identify dynamic and changing requirements
- Business information required for building software applications
- E.g. the student example, or a product ontology
Conclusion

- RE involves activities to generate consistent and complete requirement
- Ontologies could aid the RE process to provide formal representation of unambiguous, consistent, and traceable requirements
- Three types ontologies in RE
  - Requirement Document Specification Ontology
  - Requirement Ontology
  - Domain and Application Ontology
- An ontology should be a sub-product of a RE activity.
Thank you
Reference

- [Fox2016] M.S. Fox, “PolisGnosis Project Enabling the Computational Analysis of City Performance” (Powerpoint Presentation), University of Toronto (April 29, 2016)