The *SoLaSoTe* ontology for software languages & technologies

Ralf Lämmel, Martin Leinberger, and Andrei Varanovich
Software Languages Team
University of Koblenz-Landau, Germany
http://softlang.wikidot.com

© 2014, Software Languages Team, University of Koblenz-Landau
Ontologies in software engineering — some data points
Ontologies in software engineering

@article{GasevicGTW10,
  author    = {Dragan Gasevic and Giancarlo Guizzardi and Kuldar Taveter and Gerd Wagner},
  title     = {Vocabularies, ontologies, and rules for enterprise and business process modeling and management},
  journal   = {Inf. Syst.},
  volume    = {35},
  number    = {4},
  year      = {2010},
  pages     = {375–378},
}
Ontologies in software engineering

@inproceedings{SouzaFV13,
  author    = {Erica F. Souza and
               Ricardo de Almeida Falbo and
               N. L. Vijaykumar},
  title     = {Ontologies in Software Testing: A Systematic Literature Review},
  booktitle = {ONTOBRAS},
  publisher = {CEUR-WS.org},
  series    = {CEUR Workshop Proceedings},
  volume    = {1041},
  year      = {2013},
  pages     = {71–82},
}
Ontologies in software engineering

@inproceedings{CarvalhoAG14,
  author     = {Victorio Albani de Carvalho and Jo\~{a}o Paulo A. Almeida and Giancarlo Guizzardi},
  title      = {Using Reference Domain Ontologies to Define the Real-World Semantics of Domain-Specific Languages},
  booktitle  = {CAiSE},
  year       = {2014},
  pages      = {488–502},
  publisher  = {Springer},
  series     = {LNCS},
  volume     = {8484},
}
Ontologies in software engineering

@inproceedings{BarcellosF13,
  author    = {Monalessa Perini Barcellos and Ricardo de Almeida Falbo},
  title     = {A software measurement task ontology},
  booktitle = {SAC},
  publisher = {ACM},
  year      = {2013},
  pages     = {311-318},
}
Ontologies in software engineering

@article{WongthongthamCDS09,
  author    = {Pornpit Wongthongtham and Elizabeth Chang and Tharam S. Dillon and Ian Sommerville},
  title     = {Development of a Software Engineering Ontology for Multisite Software Development},
  journal   = {IEEE Trans. Knowl. Data Eng.},
  volume    = {21},
  number    = {8},
  year      = {2009},
  pages     = {1205–1217},
}
Ontologies in software engineering

@inproceedings{DobsonLS05,
  author    = {Glen Dobson and
               Russell Lock and
               Ian Sommerville},
  title     = {QoSOnt: a QoS Ontology for Service–Centric Systems},
  booktitle = {EUROMICRO–SEAA},
  year      = {2005},
  pages     = {80–87},
  publisher = {IEEE Computer Society},
}
Ontologies in software engineering

@article{Henderson-SellersGML14,
  author    = {Brian Henderson-Sellers and
               Cesar Gonzalez-Perez and
               Tom McBride and
               Graham Low},
  title     = {An ontology for ISO software engineering standards: 1) Creating
               the infrastructure},
  journal   = {Computer Standards \\& Interfaces},
  volume    = {36},
  number    = {3},
  year      = {2014},
  pages     = {563–576},
}
Ontologies in software engineering

@inproceedings{WongthongthamCDS05,
    author    = {Pornpit Wongthongtham and
                  Elizabeth Chang and
                  Tharam S. Dillon and
                  Ian Sommerville},
    title     = {Software Engineering Ontologies and Their Implementation},
    booktitle = {IASTED Conf. on Software Engineering},
    publisher = {IASTED/ACTA Press},
    year      = {2005},
    pages     = {208–213},
}
Ontologies in software engineering

@incollection{RuizH06,
author = "Francisco Ruiz, Jos'e R. Hilera",
title = "Ontologies for Software Engineering and Software Technology",
year = "2006",
pages = "49–102",
booktitle = "Using Ontologies in Software Engineering and Technology",
publisher = "Springer"
}
Ontologies in software engineering

@inproceedings{Ahmed08,
  author    = {Emdad Ahmed},
  title     = {Use of Ontologies in Software Engineering},
  booktitle = {SEDE},
  year      = {2008},
  publisher = {ISCA},
  pages     = {145–150},
  bibsource = {DBLP, http://dblp.uni-trier.de}
}
The *SoLaSoTe* ontology for software languages and technologies
SoLaSoTe’s cause:
Knowledge representation

- **Classification** of languages and technologies as well as related concepts.
- **Dependencies** between languages and technologies.
- **Concept-based characterization** of languages and technologies.
- **Links to existing knowledge resources** for languages and technologies.
- **Traceability** for language and technology usage in **shared software systems**.
http://101companies.org/wiki/Contribution:simplejdbc
What kind of ontology?

- Domain ontology
- Task ontology
- Application ontology
- Generic ontology

© 2014, Software Languages Team, University of Koblenz-Landau
SoLaSoTe’s perceived benefits

• **Unambiguous terminology** in the “domain” of languages and technologies.

• Identification of **commonalities and differences** of entities in ditto domain.

• Systematic **demonstration** of languages and technologies.

• **Integration** of otherwise scattered knowledge resources.
Querying SoLaSoTe to „infer“ knowledge

Paradigm-specific concepts Given a small set of programming paradigms, find the concepts that appear to be (more or less) uniquely associated with each paradigm—by means of collecting concepts being mentioned in the documentation of contributions, which are using programming languages of the different paradigms.

Simple baseline implementation Find the contribution that uses a given language and exercises a given concept such that there is no other contribution with less features, languages, technologies, and concepts involved.

Knowledge holder shortage Identify languages and technologies that are used infrequently by contributions without a proportional frequency of contributors who appear to be knowledgeable for these languages and technologies.
The \textit{SoLaSoTe} schema
## Top-level classification of entities

- **Entity**
  - Language
    - *Everything in the scope of the ontology*
    - *Software languages such as Java or XML*
  - Technology
    - *Software technologies such as JUnit or Eclipse*
  - Concept
    - *Software concepts such as Visitor or Unit testing*
  - Feature
    - *Features of 101’s imaginary system*
  - Contribution
    - *Implementations of 101’s imaginary system*
  - Contributor
    - *Contributors of code and documentation*
  - Theme
    - *Containers of related contributions*
  - Vocabulary
    - *Containers of domain-specific terms*
  - Resource
    - *External resources such as standards and specifications*

There are a few „less important“ types.
### Top-level classification of entities

- **Entity**
  - Everything in the scope of the ontology
  - **Language**
    - Software languages such as Java or XML
  - **Technology**
    - Software technologies such as JUnit or Eclipse
  - **Concept**
    - Software concepts such as Visitor or Unit testing
  - **Feature**
    - Features of 101's imaginary system
  - **Contribution**
    - Implementations of 101's imaginary system
  - **Contributor**
    - Contributors of code and documentation
  - **Theme**
    - Containers of related contributions
  - **Vocabulary**
    - Containers of domain-specific terms
  - **Resource**
    - External resources such as standards and specifications

### Semantic properties grouped by subject entity

<table>
<thead>
<tr>
<th>Entity</th>
<th>Concept</th>
<th>Language</th>
<th>Technology</th>
<th>Concept</th>
<th>Language</th>
<th>Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>instanceof</td>
<td>Entity</td>
<td>An instance/type relationship</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>isA</td>
<td>Entity</td>
<td>A specialization relationship</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>partOf</td>
<td>Entity</td>
<td>A whole-part relationship</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dependsOn</td>
<td>Entity</td>
<td>Dependence relationship</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mentions</td>
<td>Entity</td>
<td>Nonspecific reference in documentation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sameAs</td>
<td>URL</td>
<td>Equivalence relative to external resource</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>similarTo</td>
<td>URL</td>
<td>Similarity relative to external resource</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>linksTo</td>
<td>URL</td>
<td>Nonspecific reference to external resource</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>documentedBy</td>
<td>Contributor</td>
<td>Authorship of documentation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>memberOf</td>
<td>Vocabulary</td>
<td>Membership in vocabulary of terms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Contribution</th>
<th>Concept</th>
<th>Language</th>
<th>Technology</th>
<th>Concept</th>
<th>Language</th>
<th>Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>uses</td>
<td>Language</td>
<td>Language usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uses</td>
<td>Technology</td>
<td>Technology usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uses</td>
<td>Concept</td>
<td>Concept usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>implements</td>
<td>Feature</td>
<td>Feature implementation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>developedBy</td>
<td>Contributor</td>
<td>Developer of contribution</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>reviewedBy</td>
<td>Contributor</td>
<td>Reviewer of contribution</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>memberOf</td>
<td>Theme</td>
<td>Membership in theme of contributions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>basedOn</td>
<td>Contribution</td>
<td>Indication of reuse</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>varies</td>
<td>Contribution</td>
<td>Indication of variation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>moreComplexThan</td>
<td>Contribution</td>
<td>Indication of complexity</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource</th>
<th>Concept</th>
<th>Language</th>
<th>Technology</th>
<th>Concept</th>
<th>Language</th>
<th>Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>describes</td>
<td>Language</td>
<td>Language definitions, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>describes</td>
<td>Technology</td>
<td>API specifications, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>describes</td>
<td>Concept</td>
<td>Textbook, white papers, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Technology</th>
<th>Concept</th>
<th>Language</th>
<th>Technology</th>
<th>Concept</th>
<th>Language</th>
<th>Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>uses</td>
<td>Language</td>
<td>Language usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uses</td>
<td>Technology</td>
<td>Technology usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uses</td>
<td>Concept</td>
<td>Concept usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>implements</td>
<td>Language</td>
<td>Parsers, compilers, interpreters, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>implements</td>
<td>Resource</td>
<td>Compliance with a standard, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>supports</td>
<td>Concept</td>
<td>Support of a protocol, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Semantic properties grouped by subject entity

<table>
<thead>
<tr>
<th>Entity</th>
<th>Instance</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>instanceOf</code></td>
<td>Entity</td>
<td>An instance/type relationship</td>
</tr>
<tr>
<td><code>isA</code></td>
<td>Entity</td>
<td>A specialization relationship</td>
</tr>
<tr>
<td><code>partOf</code></td>
<td>Entity</td>
<td>A whole-part relationship</td>
</tr>
<tr>
<td><code>dependsOn</code></td>
<td>Entity</td>
<td>Dependence relationship</td>
</tr>
<tr>
<td><code>mentions</code></td>
<td>Entity</td>
<td>Nonspecific reference in documentation</td>
</tr>
<tr>
<td><code>sameAs</code></td>
<td>URL</td>
<td>Equivalence relative to external resource</td>
</tr>
<tr>
<td><code>similarTo</code></td>
<td>URL</td>
<td>Similarity relative to external resource</td>
</tr>
<tr>
<td><code>linksTo</code></td>
<td>URL</td>
<td>Nonspecific reference to external resource</td>
</tr>
<tr>
<td><code>documentedBy</code></td>
<td>Contributor</td>
<td>Authorship of documentation</td>
</tr>
<tr>
<td><code>memberOf</code></td>
<td>Vocabulary</td>
<td>Membership in vocabulary of terms</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Contribution</th>
<th>Language</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>uses</code></td>
<td>Language</td>
<td>Language usage</td>
</tr>
<tr>
<td><code>uses</code></td>
<td>Technology</td>
<td>Technology usage</td>
</tr>
<tr>
<td><code>uses</code></td>
<td>Concept</td>
<td>Concept usage</td>
</tr>
<tr>
<td><code>implements</code></td>
<td>Feature</td>
<td>Feature implementation</td>
</tr>
<tr>
<td><code>developedBy</code></td>
<td>Contributor</td>
<td>Developer of contribution</td>
</tr>
<tr>
<td><code>reviewedBy</code></td>
<td>Contributor</td>
<td>Reviewer of contribution</td>
</tr>
<tr>
<td><code>memberOf</code></td>
<td>Theme</td>
<td>Membership in theme of contributions</td>
</tr>
<tr>
<td><code>basedOn</code></td>
<td>Contribution</td>
<td>Indication of reuse</td>
</tr>
<tr>
<td><code>varies</code></td>
<td>Contribution</td>
<td>Indication of variation</td>
</tr>
<tr>
<td><code>moreComplexThan</code></td>
<td>Contribution</td>
<td>Indication of complexity</td>
</tr>
</tbody>
</table>

**Fig. 1.** The schema of SoLaSoTe (with some omissions)
<table>
<thead>
<tr>
<th>Entity</th>
<th>Language</th>
<th>Technology</th>
<th>Concept</th>
<th>Feature</th>
<th>Contribution</th>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>documentedBy</td>
<td>Contributor</td>
<td>Authorship of documentation</td>
<td>Membership in vocabulary of terms</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>memberOf</td>
<td>Vocabulary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contribution</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uses</td>
<td>Language</td>
<td>Language usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uses</td>
<td>Technology</td>
<td>Technology usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uses</td>
<td>Concept</td>
<td>Concept usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>implements</td>
<td>Feature</td>
<td>Feature implementation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>developedBy</td>
<td>Contributor</td>
<td>Developer of contribution</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>reviewedBy</td>
<td>Contributor</td>
<td>Reviewer of contribution</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>memberOf</td>
<td>Theme</td>
<td>Membership in theme of contributions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>basedOn</td>
<td>Contribution</td>
<td>Indication of reuse</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>varies</td>
<td>Contribution</td>
<td>Indication of variation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>moreComplexThan</td>
<td>Contribution</td>
<td>Indication of complexity</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Resource</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>describes</td>
<td>Language</td>
<td>Language definitions, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>describes</td>
<td>Technology</td>
<td>API specifications, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>describes</td>
<td>Concept</td>
<td>Textbook, white papers, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Technology</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uses</td>
<td>Language</td>
<td>Language usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uses</td>
<td>Technology</td>
<td>Technology usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>uses</td>
<td>Concept</td>
<td>Concept usage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>implements</td>
<td>Language</td>
<td>Parsers, compilers, interpreters, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>implements</td>
<td>Resource</td>
<td>Compliance with a standard, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>supports</td>
<td>Concept</td>
<td>Support of a protocol, et al.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Technicalities of SoLaSoTe
"Do you use OWL?"

Rarely
Validation versus reasoning

• XSD, JSON-SCHEMA are made for good old validation.
• RDFS and Owl are made for reasoning, not validation.
• Validation implies (some sort of) CWA.
• Semantic Web (for most part) assumes OWA.
Our validation process

- **Extract RDF triples from the semantic wiki:**
  - Entity becomes a root class.
  - Language, Technology, etc. become subclasses of Entity.
  - The isA properties give rise to rdfs:subClassOf properties.
  - The instanceOf properties give rise to rdf:type properties.
  - All other semantic properties are adopted, as is.

- **Analyze the integrity of the RDF triples:**
  - All resources have an rdf:type property.
  - The subjects and objects of properties agree with the schema.
  - No properties other than those of the schema are used.
  - An instance is never specialized (as in OWL DL).
BTW, such techniques are used elsewhere.

http://docs.stardog.com/icv/
SoLaSoTe in action
The End.