Ontology Summit 2012

Track 4: Large-Scale Domain Applications

Co-Champions

Steve Ray
Trish Whetzel

Thursday, April 12, 2012
Mission Statement

• This track will help to ground the discussions in the other tracks and bring key challenges to light by describing current large-scale systems and systems of systems that either use, or could use, ontologies in their deployment. "Large-scale" can mean either very large data sets, very complex data sets, federated systems, highly distributed systems, or real-time, continuous data systems.

• Examples of large data sets might include scientific observations and studies; complex data sets could be technical data packages for manufactured products, or electronic health records; federated systems could include information sharing to combat terrorism, highly distributed systems includes items such as the smart electrical grid (aka Smart Grid), and real-time systems include network management systems. Of course, some big systems might include all five aspects.
Large-scale Domain Applications

• Smart electrical grid (UML to OWL)
• Geography
• DoD system building (using OntoUML)
• Civilian government applications
• Oil
• Clinical genomics
• Plant science
• Hydrology
• Earth Sciences
Speakers

• Dr. Andrew Crapo (General Electric)
  – "Overcoming Challenges Using the CIM as a Semantic Model for Energy Applications"

• Dr. Krzysztof Janowicz (UCSB)
  – "Data-Intensive Geospatial Semantics"

• Mr. Bruce Bauman (DoD)
  – "Separating Semantics and Implementation: From a Single Ontologically Sound Conceptual Model to Multiple Physical Schema Languages"

• Mr. Mills Davis (Project10X)
  – "What if Everything You Know about System Engineering is Wrong?"
Speakers

• Mr. **David Price** (TopQuadrant)
  – "Experiences from a Large Scale Ontology-Based Application Development for Oil Platforms"

• Dr. **Mike Kellen** (Sage Bionetworks)
  – "Collaborative Clinical Genomics Data Analysis with Sage Bionetworks Synapse"

• Dr. **Damian Gessler** (iPlant Collaborative) & Dr. **Blazej Bulka** (Clark & Parsia)
  – "The iPlant Collaborative Semantic Web Platform: Using OWL and SSWAP (Simple Semantic Web Architecture and Protocol) for On-Demand Semantic Pipelines"

• Dr. **Ilya Zaslavsky** (SDSC)
  – "Managing observation semantics in CUAHSI Hydrologic Information System"

• Dr. **Line Pouchard** (ORNL)
  – "Linked Earth Science: a producer and consumer of Big Data"
Observations / Lessons learned

- Converting UML to OWL is a common requirement for upgrading legacy systems
  - Starting from scratch is rare.
- Ontology patterns are very helpful, and encourage model reuse
- Semantic techniques work best when not compromised by implementation tradeoffs
- Semantic methods are faster to implement and easier to maintain
- Semantic approaches are particularly suited to systems with many complex constraints, rules, laws, with frequent changes
- Incremental implementation is possible through federation of datastores
- Ontologies are not always applied to enable reasoners - sometimes just as a more rigorous data modeling approach
- Engineers turned ontologists often don't have the necessary background/skills
- Existing infrastructure supports traditional software development far better than large-scale ontology development
- There are many ontologies of dubious quality
- Service-oriented architectures allow separation of code and ontology updates
- Reasoning and query engine performance is highly dependent upon the exact formulation of rules and queries
- No single technology/tool currently provides the best solution across all large system use cases
Recommended Practices

• Look for the 80-20 rule of semantic development
• Use well-defined and narrow use cases to demonstrate benefits of semantic approaches
• Having explicit vocabularies (classifiers) is a must in a distributed system;
• Community should be included in the development and evolution of vocabularies
• It is critical to capture and evolve domain knowledge in a form that the community is comfortable with
• Transition from implicit domain knowledge to explicit encoding requires community consensus - and an organization to manage the consensus
In implemented systems, ontologies are...

- Strong for:
  - Supporting change and aggregation
  - Enabling community aggregation, annotation
  - Automated data ingestion
  - Data validation
  - Ensuring consistency of terms across many data sets (Distributed systems)
  - Supporting reasoning
  - Self describing systems
  - Systems with many complex constraints, rules, laws, with frequent changes (Dynamically changing systems)
  - Data mining / semantic signature extraction
  - Rapid system building

- Weak for:
  - Being understandable by software engineers and customers
  - Query performance (compared to relational databases)
Needs

• Need better standards for common elements:
  – Datatypes
  – Ontology patterns (e.g. whole/part patterns)
  – Collect ontological primitives from observation data
• Need repositories
  – Repositories of ontological patterns as well as repositories of ontologies
• Need industrial strength semantic services resident in the cloud
• Need better visualization tools and approaches
• Need better tools to help interpret legacy systems, transform into semantic systems.
• Need to establish feedback mechanisms from end users to ontology designers directly from point of use.