Ontolog Project Report

November 5, 2003

What is an Formal Ontology?

- An ontololgy is a shared conceptualization of a domain
- An ontology is a set of definitions in a formal language for terms describing the world.
- Different ontologies may differ in terms of their level of formalization

Origins of the Ontolog Forum

- **March 2002: Peter Yim and the UBL LCSC**
  - Majority of interest in learning about ontologies
  - Timetables and deadlines limited attention
- **September 2002: Reconstituted Ontolog Forum**
  - Open community
  - Charter
    - Discuss practical issues and strategies associated with the development of both formal and informal ontologies used in business
    - Indentify ontological engineering approaches that might be applied to the UBL effort

Ontolog Membership

- **> 70 subscribers, in 10 countries**
- **Overlap with UBL committee**
  - Bill Burcham
  - Sally Chan
  - Eduardo Gutentag
  - Monica Martin
  - Tim McGrath
  - Bill Meadows
  - Sue Probert
  - Marion Royal
  - Peter Yim
- **Join us! (As either observers or active members)**

Ontolog Logistics

- Infrastructure provided by CIM3.net
- Archived mailing list
- Shared, web-accessable work space
- Community Wiki
- Real-time screen and application sharing
- Weekly phone meetings (Thursday, 10:30 Pacific)

UBL-Ontology Project

- **Mission: Create a formal ontology based on the UBL schemas**

- **Aligns with general Ontolog community objectives**
  - Learn about ontologies (concepts, language, best practices)
  - Identify lifecycle process for developing ontology-based systems
  - Increase awareness and understanding of ontology tools
  - Work with a group of people on a common ontology
  - Apply ontologies to real-world applications, especially eBusiness
- **Participation: ~23 individuals (~10 active participants)**

Expected Relationship to UBL

- **UBL schemas are starting point for formalization**
- **Resulting ontology expected to**
  - Extend and formalize UBL English definitions
  - Formalize relationship semantics (hierarchical and non-hierarchical)
- **Ontolog team may provide "early warnings" to UBL teams (e.g., Context Methodology, or when get stuck)**
- **Input to UBL biased towards "actionable feedback"**
- **Anticipates an accurate modeling of the UBL domain that could result in some level of validation, acceptance, approval, or adoption by the UBL committee.**

Initial Technical Goals

- Leverage as much of the UBL committee's work as possible (don't reinvent the wheel)
- Leverage open processes, technologies, content, and philosophy
- Map to multiple upper ontologies (currently de-emphasized)
- Demonstrate multiple tools and methodologies (currently de-emphasized)
- Implement a real-life, public-domain application in parallel with the development of the ontology

Project Management Strategy

- **Demonstration project**
- **Heavily based on consensus (very voluntary)**
- **Iterative project management model**
  - Settled key technology and methodology questions ahead of full requirements
  - Some issues with alignment and shared understanding (project goals, drivers, and constraints)

High-Level Methodology

- Determine the domain and scope of the ontology
- Consider reusing existing ontologies
- Enumerate important terms in the ontology
- Define the classes and class hierarchy
- Define the properties of the classes
- Define the additional properties related to or necessary for classses (i.e., cardinality, bidirectionality/inverse, etc.)
- Create instances
- Create axioms/rules

Technology Selection

- **Base ontology: Suggested Upper Merged Ontology (SUMO)**
- **Normative Representation Language: Knowledge Interchange Format (SUO-KIF)**
- **Other derivative representations will be considered**
  - OWL, SQL, XML, “Protege”, etc.
- **Tools**
  - Text editors
  - Adam Pease’s SIGMA knowledge engineering environment
  - Sevcenko’s SUO-KIF ontology browser
  - Protege

Determine Domain and Scope

Ongoing

- **Initial doctype targets: Purchase Order, Invoice, Shipping Documents**
- **Use case articulation started in May**
  - Automated reconcilliation of Purchase Orders and Invoices
  - Identifying proper structure of an address from context
  - Mapping between different standards and representation languages
- **Primary workshop objective: Identify detailed utilization scenarios that can drive modeling**

Considering Ontology Reuse

Complete

- **Base ontology: Suggested Upper Merged Ontology (SUMO)**
  - Open
  - Rich representational language (KIF)
- **Implications**
  - Limited tools support
  - Can't be used natively in Protege (different levels of richness)
  - Raging "Protege vs KIF" debate in August/September
    - Protege well-developed and simple but not as expressive as KIF
    - Modeling process cannot rely on Protege unless bidirectionality can be demonstrated
  - **Elevated training & knowledge transfer requirements**

Enumerate Important Terms

Commencing

- **Some intial, exploritory modeling of UBL terms in July timeframe**
- **Hampered by**
  - Limited understanding of UBL
  - Lack of modeling principles (What's the best/agreed upon way to model a given concept?)
- **Primary Workshop Objectives:**
  - Understand UBL modeling philosophy
  - Understand / agree upon definition construction rules
  - Increase familarity with UBL deliverables
  - Focus on behavioral specifications ("use cases")

Define Classes and Hierarchy

Commencing

- **Adam Pease conducted an Ontological Engineering Tutorial in May**
  - Similar to this evening's tutorial
- **Primary Workshop Objectives**
  - Establish and demonstrate "fine-grained" methodology for translating "UBL terms" into KIF expressions
  - Establish an approach for dealing with the relationship between real world entities and their XML analogs

Feedback and Guidance

- Does this make sense (why / why not)?
- Could UBL be used differently with additional semantic formalization?
- What (additional) semantic properties should be modeled?
- How is UBL expected to evolve in the future?
- Are there any existing / expected gaps or issues with UBL?
- When should the Ontolog team look beyond UBL (e.g., to Core Components) to support semantic formalization?