Whose “Fault” is This?
Untangling Domain Concepts in an Ontology of Resilient Computing

Bene Rodriguez-Castro    Hugh Glaser
Dependable Systems and Software Engineering Group
Intelligence, Agents, Multimedia Group
School of Electronics and Computer Science
University of Southampton, UK

May 7th-9th, 2008
Fast Abstracts Session at the EDCC-7
Outline

1. Introduction
   - Ontologies and Ontology Design Patterns (ODPs)

2. Motivation
   - The ReSIST Network of Excellence
   - The Fault Domain Concept

3. Characterizing Role and Reusability
   - Comparative Analysis of 2 ODPs
   - Roles of Domain Concepts in ODPs
   - Reusability of Domain Concepts in ODPs
   - Role and Reusability of Fault in ReSIST
Ontologies have emerged as one of the key components for the realization of the Semantic Web.

Ontology Engineering involves a broad range of activities focused on the development of ontologies.

Ontology Design Patterns (ODPs) have evolved from the preceding success of design patterns in software engineering, (and known as “archetypal solutions to design problems in a certain context”).
The ReSIST Project

- The ReSIST Knowledge Base: an ontologically mediated web portal that enables the end-user to browse and search different type of information in the area of resilient systems. ([http://www.rkbexplorer.com/explorer/](http://www.rkbexplorer.com/explorer/))
- The ReSIST Knowledge Base features an **ontology** in the field of **resilient computing**.
- The representation of the **Fault domain concept** in the ontology for ReSIST is challenging due to:
  - The complexity of its **definition**.
  - The **number of roles** that it supports in the ontology.
  - The **relationships** with other domain concepts in the ontology.
Matrix representation of Fault from (Avizienis 2004)
Tree representation of Fault from (Avizienis 2004)

The ReSIST Network of Excellence
The Fault Domain Concept

Whose “Fault” is This?
Representing Specified Values in OWL: “value partitions” and “value sets”
Representing Classes As Property Values on the Semantic Web
Similarities between the 2 ODPs examined

- Both use a **hierarchy of classes** to provide **anonymous individuals** as **property values** for other concepts in the ontology.
  - In Pattern 2-Variant 2 from (Rector 2005) the hierarchy is used as a representation of **features, attributes, or modifiers** that describe other concepts in the ontology.
  - In Approach 5 from (Noy 2005) the hierarchy is used as a **subject index** to annotate other domain concepts in the ontology.

- Both keep ontology expressivity within **OWL-DL**
Differences between the 2 ODPs examined

- Regarding the **hierarchy of classes**:
  - In Pattern 2-Variant 2 (Rector 2005) it conforms to the definition of **value partition**.
  - In Approach 4 (Noy 2005) it does **not** and classes could be organized in any hierarchical structure.

- Regarding the **anonymous individuals**:
  - In Pattern 2-Variant 2 (Rector 2005) they are of the **same type** of the other individuals in the class.
  - In Approach 4 (Noy 2005) they are of **different type** of the other individuals in the class. Anonymous individuals are subjects while the others are actual animals.
A Terminology for Roles of Domain Concepts

- **Generic Class Hierarchy (GCH):** refers to a set of classes organized in any hierarchical structure (e.g. a single class or a set of classes organized in a list, a tree or a directed acyclic graph).

- **Domain Class Hierarchy (DCH):** refers to any GCH that contains the classes corresponding to the domain concepts that the ontology is intended to represent.

- **Value Class Hierarchy (VCH):** refers to any GCH that is used to provide anonymous individuals as values to properties for other domain concepts in the ontology.
A Terminology for Roles of Domain Concepts II

- **Value Partition Class Hierarchy (VPCH):** refers to a GCH that: a) is a Value Class Hierarchy and b) conforms to the definition of a value partition.

- **Domain Concept Space (DCS):** identifies the subset of the ontology model that contains all the classes that belong to a Domain Class Hierarchy.

- **Value Space (VS):** identifies the subset of the ontology model that contains all the classes that belong to a Value Class Hierarchy or Value Partition Class Hierarchy.
Example 1: Roles of Domain Concepts in (Rector 2005)

Value Partition Class Hierarchy:
- HealthValue
  - Poor_health_value
  - Medium_health_value
  - Good_health_value

Domain Class Hierarchy:
- Person
  - Healthy_Person
  - Johns_Health
  - John

Comparative Analysis of 2 ODPs
Roles of Domain Concepts in ODPs
Reusability of Domain Concepts in ODPs
Role and Reusability of Fault in ReSIST
Example 2: Roles of Domain Concepts in (Noy 2005)

- Value Class Hierarchy:
  - Animal
  - Lion (rdfs:subclassOf Animal)
  - African Lion (rdfs:subclassOf Lion)

- Domain Class Hierarchy:
  - Book About Animals
    - "Lions: Life in the Pride"
    - "The African Lion"

Bene Rodriguez-Castro, Hugh Glaser

Whose “Fault” is This?
Scenario 1:

Let us consider two ontologies $O_1$ and $O_2$, with two Domain Class Hierarchies $DCH_1$ and $DCH_2$ in their Domain Concept Space respectively.

In the context of (Noy 2005) and (Rector 2005) we can **reuse** $DCH_2$ from $O_2$ to support the role of a Value Class Hierarchy in ontology $O_1$. 
Example Scenario 1: Premise

Ontology 2: Animals

- **Animal**
  - rdfs:subclassOf
  - **Lion**
    - rdfs:subclassOf
    - **African Lion**
      - rdf:type
      - **An Actual African Lion**

Ontology 1: Books

- **Books**
  - rdfs:subclassOf
  - **Books About Animals**
    - rdfs:subclassOf
    - **Books About Lions**
      - rdf:type
      - **"Lions: Life in the Pride"**
      - **"The African Lion"**

Bene Rodriguez-Castro, Hugh Glaser

Whose “Fault” is This?
Example Scenario 1: Conclusion

Bene Rodriguez-Castro, Hugh Glaser

Whose “Fault” is This?
Scenario 2:

- Let us consider a single ontology $O_1$, with two Domain Class Hierarchies $DCH_{11}$ and $DCH_{12}$ in its Domain Concept Space.

- In the context of (Noy 2005) and (Rector 2005) we can **reuse** $DCH_{12}$ to support the role of a Value Class Hierarchy for $DCH_{11}$ in $O_1$. 
Example Scenario 2: Premise

Ontology 1 – IT Department

DCH12

- Issue
  - Software
    - Linux
      - rdf:type
        - Driver
          - Wi-Fi
    - Windows
      - rdf:type
        - Virus
          - Found
  - Hardware
    - rdf:s subclassOf
  - rdf:s subclassOf

DCH11

- Person
  - Employee
    - rdf:s subclassOf
  - Customer
    - rdf:s subclassOf
  - Technician
    - rdf:type
      - John
      - Larry

Whose “Fault” is This?
Example Scenario 2: Conclusion

DCH12 in O1 reused as a VCH in O1

Ontology 1 - IT Department

Issue

rdfs:subclassOf

Software

Hardware

rdfs:subclassOf

Linux

rdfs:subclassOf

Windows

Unidentified Hardware Issue

Unidentified Linux Issue

Unidentified Windows Issue

Person

rdfs:subclassOf

Customer

Employee

Technician

John

Larry

hasExperienceWith

someValuesFrom

hasExperienceWith

someValuesFrom

Driver

WI-FI

Virus

Found

Value Space

Domain Concept Space

Domain Concept Space

Whose “Fault” is This?

Bene Rodriguez-Castro, Hugh Glaser
Representation of **Fault** in ReSIST fits Scenario 2
Performed a comparative analysis of 2 ODPs.

Characterized the role of domain concepts in the 2 ODPs examined.

Characterized two reusability scenarios for domain concepts in ODPs.

Made explicit certain potentially implicit modeling decisions previously taken in ontology building.

Applied findings to the representation of the Fault domain concept in the ontology for ReSIST.
Questions

References II

N. Noy.
Representing Classes As Property Values on the Semantic Web.
*W3C Note*, 2005.
http://www.w3.org/TR/2005/
NOTE-swbp-classes-as-values-20050405

A. Rector.
Representing specified values in owl: “value partitions” and “value sets”.
*W3C Note*, 2005.
http://www.w3.org/TR/2005/
NOTE-swbp-specified-values-20050517