4-22-2010

Ontology and Knowledge Base of Brittle Deformation Microstructures for the San Andreas Fault Observatory at Depth (SAFOD) Core Samples

Cynthia Marie Broda
Georgia State University

Recommended Citation

Broda, Cynthia Marie, "Ontology and Knowledge Base of Brittle Deformation Microstructures for the San Andreas Fault Observatory at Depth (SAFOD) Core Samples." Thesis, Georgia State University, 2010.
https://scholarworks.gsu.edu/geosciences_theses/26

ONTOLOGY AND KNOWLEDGE BASE OF BRITTLE DEFORMATION MICROSTRUCTURES FOR THE SAN ANDREAS FAULT OBSERVATORY AT DEPTH (SAFOD) CORE SAMPLES

by

CYNTHIA M. BRODA

Under the Direction of Hassan Babaie

ABSTRACT

The quest to answer fundamental questions and solve complex problems is a principal tenet of Earth science. The pursuit of scientific knowledge has generated profuse research, resulting in a plethora of information-rich resources. This phenomenon offers great potential for scientific discovery. However, a deficiency in information connectivity and processing standards has become evident. This deficiency has resulted in a demand for tools to facilitate and process this upsurge in information.

This ontology project is an answer to the demand for information processing tools. The primary purpose of this domain-specific ontology and knowledge base is to organize, connect, and correlate research data related to brittle deformation microstructures. This semantically enabled ontology may be queried to return not only asserted information, but inferred knowledge that may not be evident. In addition, its standardized development in OWL-DL (Web Ontology Language-Description Logic) allows the potential for sharing and reuse among other geologic science communities.

INDEX WORDS: Ontology, Web Ontology Language, OWL, Knowledge base, Structural geology, Brittle deformation, Microstructures, San Andreas Fault Observatory at Depth, SAFOD

ONTOLOGY AND KNOWLEDGE BASE OF BRITTLE DEFORMATION MICROSTRUCTURES FOR THE SAN ANDREAS FAULT OBSERVATORY AT DEPTH (SAFOD) CORE SAMPLES

by

CYNTHIA M. BRODA

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Science

in the College of Arts and Sciences

Georgia State University

2010

ONTOLOGY AND KNOWLEDGE BASE OF BRITTLE DEFORMATION MICROSTRUCTURES FOR THE SAN ANDREAS FAULT OBSERVATORY AT DEPTH (SAFOD) CORE SAMPLES

by

CYNTHIA M. BRODA

Committee Chair: Hassan A. Babaie

Committee: W. Crawford Elliott
Seth E. Rose

Electronic Version Approved:

Office of Graduate Studies
College of Arts and Sciences
Georgia State University
May 2010

This thesis is dedicated to my husband and children who have so lovingly supported me throughout this endeavor.

I would also like to thank my parents and my mother-in-law, Betsy Broda, for their frequent assistance while I have been occupied with this project.

I am deeply grateful for all the encouragement I have received from both family and friends on this journey.

ACKNOWLEDGEMENTS

The success of this project has depended largely on the help and guidance I received from Dr. Hassan Babaie, my thesis director. His enthusiasm for the topic of ontology sparked my interest and greatly influenced my decision to pursue this topic of study. I would also like to acknowledge the great technical input and endless patience of Anuj Kumar. Without his assistance and technical contributions this project would not have had such potential.

In addition, I would like to acknowledge and thank my department head and committee member, Dr. Crawford Elliott. His leadership, advice, and confidence in my abilities have meant a great deal to me throughout my graduate studies.

I also wish to thank Dr. Seth Rose whose dedication to his classes and students is unquestionable. I admire his independent thinking and candid dialogue. His feedback and willing assistance is greatly appreciated.

TABLE OF CONTENTS

CHAPTER

APPENDICES

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The quest to answer fundamental questions about Earth features, phenomena, and processes, in order to solve complex Earth system problems, and to engender knowledge are among the principal tenets of Earth science. As scientific knowledge in these areas expands, an increase in disconnected segments of information continues to develop. Information becomes difficult to manage, and therefore, knowledge becomes less accessible and more challenging to integrate. Scientists are looking to technology at an ever increasing pace to solve these information processing dilemmas.

Ontology, a formal representation of knowledge, is a system that allows for the bridging of disconnected islands of information. Domain ontologies are applied to a particular conceptual representation, such as an ontology applied to the domain of structural geology. This type of domain oriented ontology consists of a set of classes that represent concepts about real geological features and processes, and the relationships among these classes. This approach is distinguished from other ways of organizing knowledge by the richness and expressiveness of these relationships [Larson and Martone, 2009]. A knowledge base representation system goes one step further than a basic ontology in that it also stores and processes the data defined by the ontology.

Ontologies have become relatively common on the World Wide Web. The National Science Foundation's Office of Cyberinfrastructure has announced calls for proposals in greater ontology development for the physical sciences [Raskin, 2007]. The objective of this thesis project is to develop an ontology and knowledge base for the brittle microstructures in the cores of the San Andreas Fault Observatory at Depth (SAFOD) project located in Parkfield, CA.

SAFOD Project Overview

SAFOD is created by the need to answer fundamental questions about the physical and chemical processes controlling faulting and earthquake generation within a major plate-boundary fault. The observatory project involved the drilling and instrumentation of a borehole through the active earthquake region of the San Andreas Fault Zone at 3.2 kilometers depth. The project has produced a collection of rock samples for the physical and chemical investigation of the active earthquake zone [Earthscope, 2009].

The SAFOD project is a collaborative effort funded by multiple resources. Support for funding programs and education has been primarily generated by: The National Science Foundation (NSF), The Earthquake Hazards Program of the United States Geological Survey (USGS), The International Continental Scientific Drilling Program (ICDP), and The German Science Foundation (ICDP of Germany).

The cores for the SAFOD project were extracted using directional drilling through the San Andreas Fault within an active portion of the fault zone. The initial pilot hole was drilled over the summer of 2002 to a depth of just over 2 km. The pilot hole was located 1.8 km southwest of the San Andreas Fault, near Parkfield, CA. The purpose of the preliminary pilot hole was to install instrumentation in the borehole in order to assist in determining exact hypocenters of repeating microearthquakes. It will also help to characterize the surrounding shallow crust region (i.e., physical properties, stress) and provide seismic imaging. This knowledge will guide investigations for the main SAFOD borehole.

The main borehole was drilled vertically for 1.5 km, before being angled at approximately 55° from horizontal and drilled to a depth of 3.2 km (Figure 1.1). Drilling of the main borehole took place in three phases. (Phase 1: June 11 – October 6, 2004; Phase 2: June 8 – August 28, 2005; Phase 3: June 14 – September 15, 2007). The borehole passed beneath the surface trace of the San Andreas Fault, where many repeating microearthquakes define the trace at depth [Zoback et al., 2005].

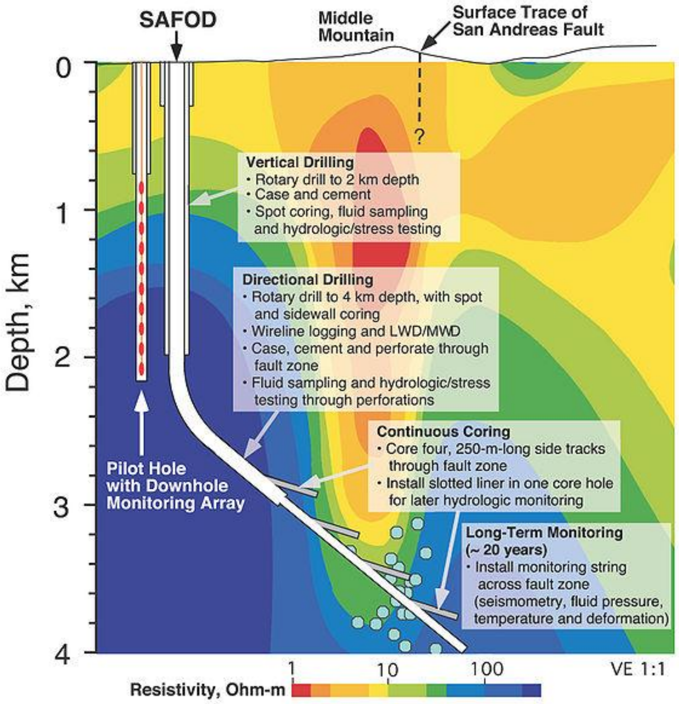


Figure 1.1. The SAFOD pilot hole and borehole with blue dots that represent approximate locations of microearthquakes [http://quake.usgs.gov/drillhole/figures/].

Approximately 135 feet of core was retrieved from the drilled boreholes. Scientifically valuable samples were distributed to selected scientists, as decided by the SAFOD sample committee. One-third of the core was distributed for non-destructive research. The remaining one-third will be available for future requests, and the final one-third was archived. Information regarding the application and selection process for the cores can be obtained from the Earthscope website at http://www.earthscope.org (Core Distribution link).

SAFOD Ontology Overview

The purpose of the main products of this thesis project, the brittle microstructure ontology and knowledge base, is to provide a way to organize, input, share, and query the core sample research data submitted by specific teams of SAFOD scientists (see list of investigators at: http://codd.cs.gsu.edu:9999/safod/). Additionally, the ontology enables the correlation of data through powerful inferencing capabilities of the Web Ontology Language (OWL). Inferencing (extracting new information from existing knowledge) allows for the sophisticated integration of data so that patterns and associations may be discovered. This intelligent system returns knowledge as well as data.

The domain (subject) of this ontology encompasses brittle deformation microstructures observed in the cores of the SAFOD project which formed in the active San Andreas Fault zone. The primary objective of this system is to allow for the classification of brittle microstructure concepts into a taxonomic hierarchy (ontology) and to define the relationships between these concepts. The ontology allows population of real data, as well as the ability to query the data on the web by the SAFOD investigators.

The knowledge base (based on the ontology) stores individual research investigator's data regarding a specific core sample. This allows an investigator to record, through Web submission, information about his or her analyses, experiments, methodology, along with detailed assumptions and results (to include images). The investigator may update (add, modify or delete) data input on a sample. An investigator may view the work and results of other scientists on the same sample or on other samples. This sharing of information is particularly useful because access to core thin sections is considerably limited due to high demand by scientists of other SAFOD teams. In essence, the Web accessible knowledge base virtually allows for greater access to the cores in a technologically elegant manner.

CHAPTER 2

THE SAN ANDREAS FAULT

San Andreas Fault Overview

The San Andreas Fault is an approximately 800 mile (1300 km) long transform fault that extends from northwest California to the Gulf of California. The fault is dominated by right-lateral strike-slip movement and forms the boundary between the Pacific oceanic and North American plate (Figure 2.1). Some major United States coastal cities, such as Los Angeles and San Diego, lie on the Pacific plate. Most of California's major cities, such as San Francisco and Sacramento, lie on the North American plate.

Figure 2.1. Surface trace of the San Andreas Fault [Lynch, 2007].

Distinctive fault zone landforms are evident in the region. These include escarpments, mountain ridges, and blocked ponds of water within the zone. Total accumulated displacement since the fault came into existence over 20 million years ago is approximately 350 miles. Displacement has occurred as a result of plate movement, earthquakes, and aseismic creep.

The San Andreas is predominantly a single vertical fault in the crust extending to the top of the mantle at 25 km depth [Ben-Zion and Rice, 1993]. While the San Andreas is the most prevalent and well known fault trace in California, other faults exist in the region. On average, the total horizontal displacement between the North American plate and the Pacific plate is approximately 3.5 to 4.6 cm per year. According to Nester [2008], the majority of this displacement is accommodated by the San Andreas Fault zone (2.0 to 3.5 cm per year), while the remainder is expressed by displacement along other subparallel faults.

Along the San Andreas Fault system are numerous exotic terranes. The pattern of movement and distances traveled by these terranes is a significant characteristic of the role of the San Andreas Fault in the overall scheme of global tectonics [Wallace, 1990]. The repositioning of continental land mass by plate tectonics has had a tremendous impact on the coastal region of western North America.

History of the San Andreas Fault

Approximately 30 million years ago, the westward moving North American Plate began to override the spreading ridge between the Farallon Plate and the Pacific Plate. This subduction process divided the Farallon Plate into two smaller plates, the northern Juan de Fuca Plate and the southern Cocos Plate. As the North American plate came into contact with the Pacific plate, subduction was replaced with right-lateral transform faulting [Stoffer, 2006]. This process resulted in the formation of the San Andreas Fault.

At the northern and southern edges of the transform fault, the Mendocino and Rivera Triple Junctions formed, respectively, and began to migrate along the plate boundaries (See Figure 2.2). The Mendocino Triple Junction migrated northward until reaching its present location off the coast of northern California, west of Cape Mendocino. The southern Rivera Triple Junction migrated southward until its present location in the Pacific Ocean, south of Baja California. Currently, these triple junctions are approximately 2,500 km apart.

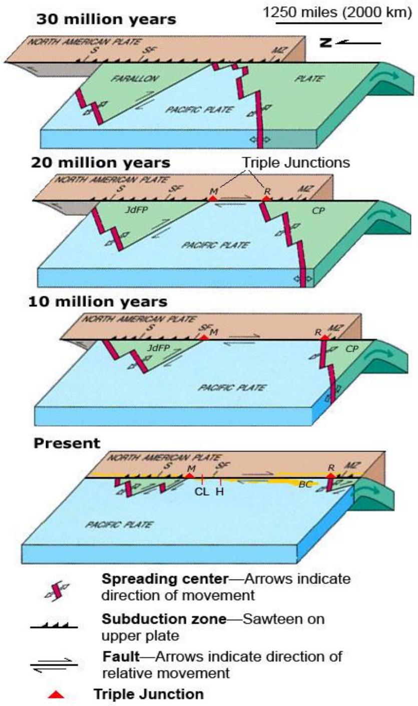


Figure 2.2. Block diagram series showing formation of the San Andreas Fault from 30 million years ago to the present [USGS: http://geomaps.wr.usgs.gov/socal/geology/geologic_history/san_andreas_history.html].

The ridge separating the Pacific and Farallon plates had an approximate north-northeast trend, resulting in very oblique convergence between the North American margin and the approaching ridge [Furlong and Schwartz, 2004]. Pacific–North American relative plate motion was subparallel to the North American margin so that convergence across that segment was replaced with the San Andreas transform. Simultaneously, the lithosphere on the southern (Farallon) flank of the ridge continued to be subducted [Furlong and Schwartz, 2004].

The geologic effects of these tectonic processes on western North America have been profound. The migration of the Mendocino Triple Junction resulted in a thickening (ahead of the junction) and then thinning (behind the junction) of the crust. This crustal fluctuation from the amalgamation of asthenosphere and crust may have converted a subduction accretion complex, such as the Franciscan Terrane (a Mesozoic- to- Cenozoic-aged, uplifted, subduction zone accretionary wedge), into a more typical continental crust [Furlong and Schwartz, 2004].

The Franciscan Terrane scraped off the subducting Pacific Plate and was thrust eastward. This process resulted in a shingling effect of blocks of rocks against western North America's Great Valley Sequence [www.nps.gov/goga/forteachers]. The boundary between Franciscan rocks and the Great Valley sequence is a low-angle, eastward-dipping thrust fault called the Coast Range Thrust [Howard, 1979]. This continual process of wedging and thrusting has resulted in a complex zone of folds and faults (Figure 2.3.).

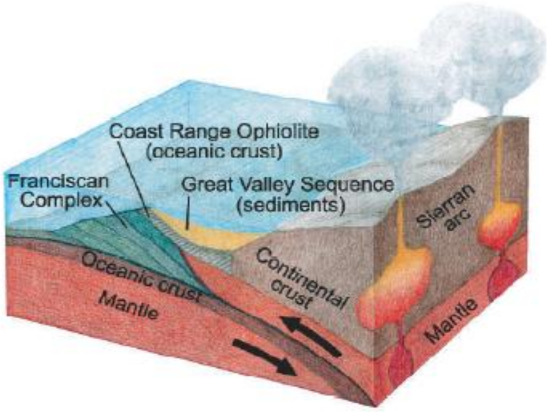


Figure 2.3. Block diagram showing subduction zone that developed along the California coast during the formation of the Franciscan Complex [www.nps.gov/goga/forteachers].

Earthquake History

Earthquake activity along the San Andreas Fault was first recorded by the Franciscan missionaries as early as 1769. The earliest seismographs capable of consistently detecting earthquakes were developed by John Milne in 1896. However, instrumental measurements did not fully supplant non-instrumental magnitudes and epicentral location detectors until the development and deployment of the Wood-Anderson seismograph in 1926 [Ellsworth, 1990]. The extensive earthquake history and immense damage that has been generated from the San Andreas Fault zone has resulted in extensive seismic study.

From 1812 to 1906, the fault generated four major earthquakes of $M > 7$ (where M is moment magnitude) [Toppozada et al., 2002]. The largest of these historical earthquakes were those in 1857 (Fort Tejon) and 1906 (San Francisco). The 1857 event ruptured the area from Parkfield, in central California, to Cajon Pass, in southern California. The 1906 event ruptured approximately 430 km of the fault (from San Juan Bautista to the triple junction at Cape Mendocino) [Ellsworth,1990] (Figure 2.4). These two earthquakes were of approximately the same magnitude, close to $M \sim 8$.

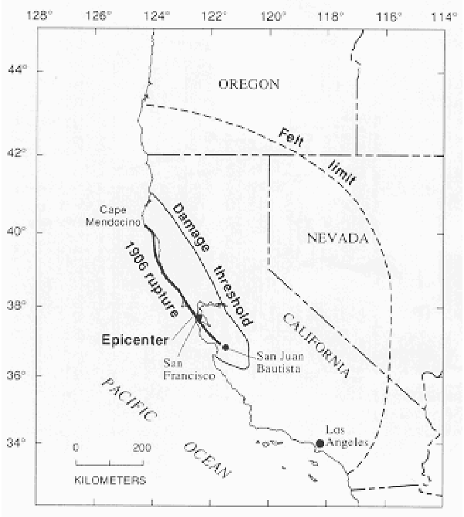


Figure 2.4. Extent of 1906 fault rupture (epicenter near San Francisco). [Modified from Lawson (1908) and Toppozada and Parke (1982)].

Earthquakes of *M* ~ 7.1 occurred in the Imperial Valley area in 1940 and 1979 [Schulz and Wallace, 1997]. More recent major eruptions (*M* > 6) have occurred in San Fernando (1971), Mammoth Lakes (1980), North Palm Springs (1986), Loma Prieta (1989), Los Angeles (1994), and Parkfield (2004). Some areas of the fault have experienced more than one major rupture event along the same segment of the fault. The 1857 faulting overlapped the 1812 earthquake faulting, and the 1906 faulting overlapped the 1838 earthquake faulting [Toppozada et al., 2002]. Time intervals between major eruptions have been variable, perhaps owing to complexity of the zone and variation in geologic features along the fault zone. This unpredictability emphasizes the critical necessity for an increase in knowledge of earthquake behavior along this heavily populated fault zone.

Geomorphology and Geologic Formations

Geomorphic landforms are valuable indicators in the study of active plate tectonics. These mostly surficial features can aid in determining such things as the rate of plate movement, direction, and where intricate fault patterns are located. An understanding of these features can contribute to the assessment of potentially hazardous areas.

The San Andreas Fault can easily be detected from the air and from aerial or satellite imagery. In many places it appears as a great scar-like feature across the landscape. This NW-SE trending trough is dispersed with mountain ranges, valleys, lakes, bays, sag ponds and offset stream channels**.** Some range-size blocks have risen or dropped hundreds of meters, or been shuffled one block against another by lateral displacement within the broad shear zone [Wallace, 1990]. This displacement, along with erosional phenomena, has probably led to the formation of many of the valleys that flank the fault.

Drainage patterns in most geologic settings flow roughly perpendicular to the trend of mountainous regions, and according to rock type in lower regions. However, along the active San Andreas Fault trace the drainage flow from highlands is diverted subparallel to the trends of the highlands, or is interrupted or blocked completely [Wallace, 1990] (Figure 2.5). Erosional activity tends to be the dominant factor in carving geomorphic forms of less active areas, but rapid displacement within the fault zone has had tectonic effects that overwhelm erosion, and so the geomorphic features directly express fault movement [Wallace, 1990].

Figure 2.5. Photo of stream offset in area of the Carrizo Plain. (Wallace Creek center is offset about 130 m.) [Wallace, 1990].

Landscape features, such as streams and geologically similar bedrock, are offset along the fault. Young rocks are offset less than older rocks. Due to the complexity of the fault system, the amount and rate of offset along the fault is not consistent from place to place [Stoffer, 2005]. The variation is most likely a result of the dendritic geometry of the faults and because some sections of the fault are creeping, while other sections are locked.

Many of the features such as mountain ridges and valleys are elongated due to the nature of right-lateral slip movement. For the same reason, depressions form in some areas from extensional stretching of the continental plate. This 800-mile-long fault is complex and variable from area to area making specific areas targets for detailed study.

Segments of the San Andreas Fault

Because the San Andreas Fault zone is exceptionally complex with much behavioral variability, geologists commonly divide the fault into segments. The boundaries of these segments can be defined by the rate of creep or offset due to earthquakes [Elam, 2000]. Most typically, three major segments are identified: the northern segment, the central segment, and the southern segment. The northern and southern portions of the fault are considered to be "locked," or not currently slipping. In these zones the top 12.5 km of the fault is locked against slip, except in great earthquakes such as the 1857 and 1906 events [Ben-Zion and Rice, 1993]. The central area of the fault zone is of particular interest because it experiences aseismic creep, or displacement without earthquakes (Figure 2.6).

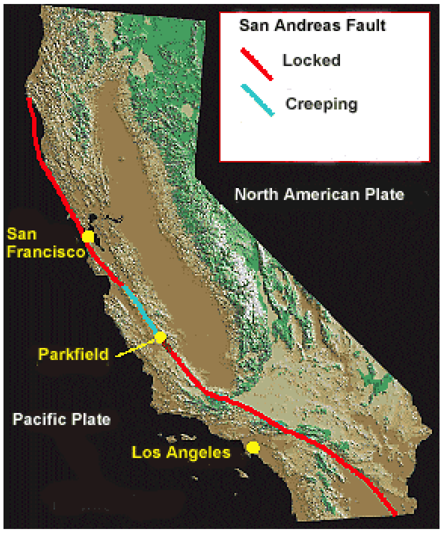


Figure 2.6: USGS image of the segments of the San Andreas Fault [USGS, 2009].

The northern segment of the fault begins offshore of Cape Mendocino in Northern California at the Mendocino Triple Junction. Here, the Pacific plate strains to move northwest, while the North American plate presses southeast, and the Gorda plate is being subducted to the northeast. The fault continues southeastward offshore, approximately mimicking the western coastal North American plate, until arriving onshore at the San Francisco Peninsula. From here it traverses northern California at a strike of approximately N40°W until it reaches Hollister [Wallace,1990]. This is the youngest section of the fault.

The great 1906 San Francisco and 1989 Loma Prieta earthquakes occurred on the northern segment of the San Andreas Fault. The 1906 earthquake ruptured 430 km of the fault, whereas the 1989 earthquake ruptured only 35 km of the fault (Figure 2.7). Therefore, the 1906 quake is considered to be the last major rupture event on this locked segment of the fault since it comprised the entire northern segment. This northern segment experiences infrequent large earthquakes with an average recurrence interval of approximately 240 years [Nester, 2008].

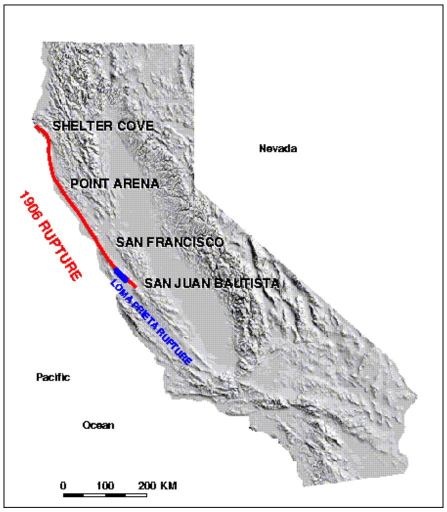
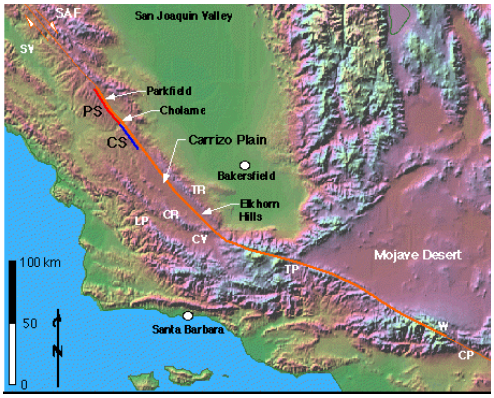


Figure 2.7: Shows the extent of the 1906 San Francisco rupture and, for comparison, the 1989 Loma Prieta rupture [http://earthquake.usgs.gov/regional/nca/1906/18april/howlong.php].

The central section of the fault continues a generally straight-line strike of N40°W from Hollister to just south of Parkfield, approximately 140 km long. The boundary of this nearly straight section of the San Andreas Fault is bordered to the north by the infamous 1906 rupture event (San Juan Bautista / Hollister), and to the south by the almost equally infamous 1857 rupture epicenter at Cholame. While the segments to the north and south of the central segment are locked against slip, the central segment experiences predominantly right-lateral aseismic slip. The actively creeping section of the fault has historical slip rates as high as 3.4 cm per year [Brown, 1990]. The city of Parkfield, the site location for this thesis project, is located on the central segment of the fault. Information regarding this segment of the fault is covered in greater detail in chapter 3.

The southern section of the fault becomes locked in the Cholame area, then becomes quite visible as it traverses the Carrizo Plain (Figure 2.8). According to Arrowsmith [1995], this section of the fault displays the largest accumulated post-early Miocene offset and is the oldest reach within the fault system. Just beyond the plain the fault changes sharply in strike to roughly N60°W. This 120 km-long area is referred to as the Big Bend.



Map Key:

TR is Temblor Range;
CR is Caliente Range;
CV is Cuyama Valley;
LP is La Panza Range;
SV is Salinas Valley
CP is Cajón Pass
TP is Tejón Pass
W is Wrightwood

Figure 2.8: Shaded relief map of Carrizo Plain in the Coast Ranges. (Original relief map is from *Relief map of the world*: cylindrical projection; elevation data from ETOPO5 dataset; by Ray Sterner, Johns Hopkins University Applied Physics Laboratory) [Arrowsmith, 1995].

Left-lateral faults splay from the main fault at the location of the bend, and clusters of earthquakes occur at depths as great as 20 km. This area of the fault splits into branches that accommodate about two-thirds of the slip motion between the North American and Pacific plates [Wallace, 1990]. This pronounced bend produces compressional forces that are most likely the cause of the east-west directional axis of the Transverse Mountain Ranges that were formed in this area. The majority of mountain ranges along the San Andreas Fault follow a general northwest-southeast trend.

The southern section of the fault continues southeastward of the Transverse Mountain Ranges until it terminates near the Salton Sea. This section of the fault exhibits evidence of the right-lateral motion of the fault with geologically similar terranes on opposite sides of the fault now separated by 150 miles. Some of these crustal blocks have moved through more than 20 degrees of latitude. [http://pubs.usgs.gov/gip/earthq3/safaultgip.html]

CHAPTER 3

THE STUDY AREA

The small town of Parkfield, CA. lies on the southern margin of the creeping segment of the San Andreas Fault. This town marks the transition of the fault between the creeping segment to the north and the locked segment to the south. It is located approximately mid-way between San Francisco and Los Angeles (Figure 3.1).

Parkfield is known as the earthquake capital of California, due to the large number of moderate sized earthquakes (M6.0 – M6.5) that have occurred there in the past 150-years [http://earthquake.usgs.gov/research/parkfield/shake/]. Historical earthquake recurrence has had somewhat regular intervals of 15-30 years, e.g., 1857, 1881, 1901, 1922, 1934 and 1966. These ruptures occurred on the same area of the fault. The most recent, and much anticipated, event occurred Sept, 2004 (*M* = 6). The remarkable pattern of seismicity has resulted in this being one of the most heavily studied earthquake zones in the world.

Figure 3.1. The town of Parkfield, located on the San Andreas fault in central California [http://earthquake.usgs.gov/research/parkfield/shake/].

Furthermore, Parkfield experiences a high volume of repeating microearthquakes. According to studies published by Nadeau and McEvilly [1997], more than half of the 4000+ seismic events in their study exhibited patterns of regularly occurring microearthquakes (0.5 to 2 yr.). While these microearthquakes (magnitudes approximately 0.2 to 1.3) relieve some of the seismic strain buildup, much of the strain is relieved through aseismic creep. The creep may be slow and steady, or episodic. The numerous small earthquakes that occur at this location define the fault to be a narrow (approximately 10- m wide), near-vertical zone to about 3- km depth [Hickman et al., 2007] (Figure 3.2). Serpentinite is considered a potential cause of, or contributing factor to, the creep and for the low strength of this section of the fault [Moore and Rymer, 2007].

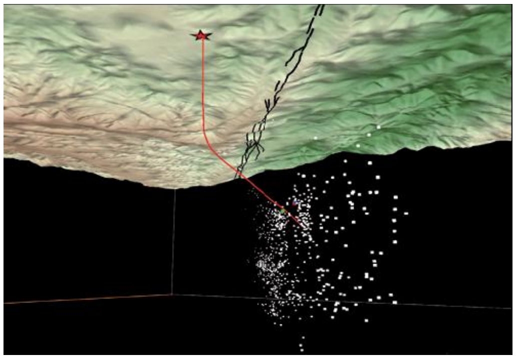


Figure 3.2. Seismicity of the San Andreas fault as seen from a vantage point in the earth looking to the northwest. The SAFOD main hole is shown in red, extending downward from the surface facility (red star). The surface trace of the fault is shown as black line (3-D EarthVision plot by Luke Blair, USGS).

Regional Geophysical Properties

The western side of the San Andreas Fault comprises a Salinian (Paleozoic era) granitic basement juxtaposed against the Franciscan complex on the eastern side of the fault [McPhee et al., 2004]. The Salinian granite is overlain by Tertiary and Quaternary sediments. The

Franciscan complex, a melange consisting primarily of metamorphosed accretionary prism, and the Coast Range ophiolite are covered by Cretaceous and younger sediments of the Great Valley sequence [Unsworth et al., 1997] [Liu et al., 2007] (figure 3.3). According to Irwin [1990], faulting has likely caused the Coast Range ophiolite to be highly sheared and thinned to the point that it is locally missing in many places. Where it does exist, it only exhibits a thickness of about a few km.

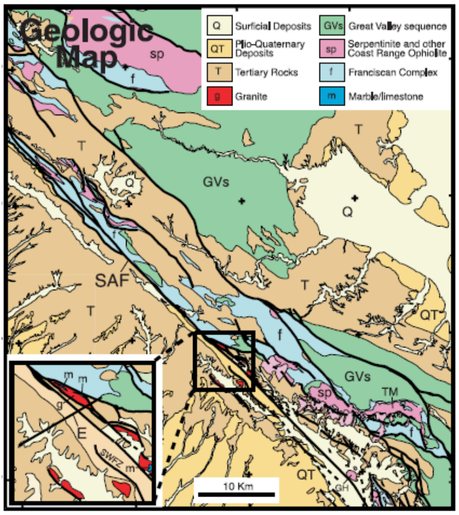


Figure 3.3. Map of San Andreas Fault lithology in Parkfield vicinity [McPhee et al., 2004].

Mapping by [Dibblee et al., 1999] show that sediments sandwiched between the Salinian granite to the west and the Cretaceous Franciscan subduction complex to the east form a syncline that plunges N30W, subparallel to the San Andreas fault [Park and Roberts, 2003]. According to Champion et al. [1984], paleomagnetic data indicate the Salinian block may have been displaced approximately 2,500 km (from Central America) since Cretaceous time. The sequences were thrust together during the historical subduction process of the Farallon plate beneath western North America. During this process, the Franciscan accretionary prism was uplifted and thrust eastward, resulting in emplacement of the overlying Coast Range ophiolite from the west.

The Franciscan assemblage consists predominantly of graywacke and shale, along with some volcanics, chert, and limestone. Fossils reveal that this assemblage was formed in a marine environment. These rocks also include serpentinite and tectonized blueschist that are the locus of much shearing [Irwin, 1990]. The Franciscan complex is broadly deformed and faulted, while overlying strata are more planar and less disturbed.

Site Selection

The Parkfield area was ideal for the SAFOD project study area for many reasons. Numerous studies had already taken place in the area (i.e., the Earthquakes Hazards Program by the USGS), providing a foundation with voluminous information available. This segment of the fault is of particular interest because of the combination of aseismic creep and repeating microearthquakes. Due to the repetitive nature of the earthquakes, the potential to find patterns for earthquake generation exists.

Seismographs from Parkfield earthquake study programs have shown the similarity between the first few cycles of P and S wave generation. This indicates that these events ruptured the same parts of the fault [http://earthquake.usgs.gov/research/parkfield/repeat.php] (Figure 3.4).

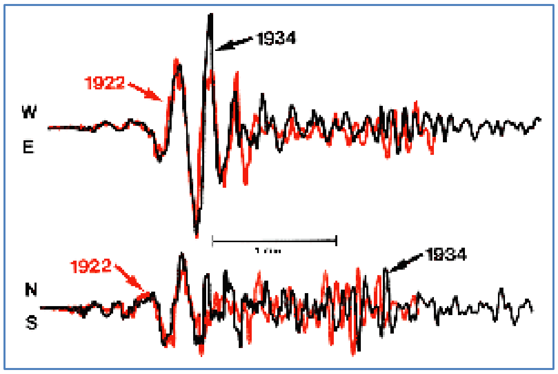


Figure 3.4. Seismograph recordings of S and P waves show East-west and north-south components of ground motion for the 1922 and 1934 Parkfield events [http://earthquake.usgs.gov/research/parkfield/repeat.php].

The position for the SAFOD main drill hole site was selected in an area of the fault segment that moves through a combination of aseismic creep and repeating microearthquakes. It lies just north of the rupture zone of the 2004 magnitude ~ 6 Parkfield earthquake. The borehole penetrates the fault in the area near Parkfield where ongoing earthquake activity has been identified.

The research and discoveries of the SAFOD investigators will be thought-provoking, and will certainly uncover new information about earthquakes. The ontology for this thesis project will be ideal for the storing and processing of portions of this information. (Initial research findings from core sample retrieval will be discussed in Chapter 6.)

CHAPTER 4

EVOLUTION OF ONTOLOGY AS A TRADITION AND ITS APPLICATIONS

Philosophical Roots

‘Ontology’ is a term borrowed from analytical philosophy. It is an area of study that has been evolving for over 2,000 years. The term “ontology” was coined from Aristotle’s *First Philosophy* (of “what exists”). Aristotle endeavored to classify beings in so far as they exist, or may exist. According to Sowa [1999], Aristotle did not use the term “being”, but rather the term “category,” for the classification of anything that can be said or predicated about anything. Aristotle referenced this categorization of beings as his First Philosophy. He believed that cognitions (perceptions of beings) were based on first principles of experience [Corazzon, 2009], rather than an intuitive consciousness. This line of thought alludes to ontology as conveying a subjective view of the world.

Philosophers since Aristotle have continued to study ontology, unifying it with theories of logic (a basis for philosophy). Christian Wolff (1679 – 1754) believed that philosophy was dependent on ontology in that it (ontology) is a system that organizes all aspects of life (i.e.: essence, attributes, order, etc.). He stated that “Even the art of discovery takes its principles from ontology.” [Corazzon, 2009]

While some philosophers believed that anything could be quantifiable in ontology, including principles that may be metaphysical, others believed that only objects that can be subjected to the physical senses were applicable to the science of ontology. Kant (1724 – 1804) thought of ontology as a resolution of knowledge into concepts based on experience, analogous to grammatical rules as the resolution of speech [Corazzon, 2009].

Husserl (1859 – 1938) used what he called “Mathesis Universalis” (pure logic) to facilitate his classification of objects into a formal ontology. He stated that all objects lead to more primary categories that are “fundamental truths which function as axioms in the discipline of pure logic” [Corrazon, 2009]. Husserl analyzed issues of category as they pertain to the establishment of hierarchical order in his analysis, especially as related to the determination of parts and wholes. This degree of granularity is a disputable question that dates back to the origins of *First Philosophy*.

Whereas the philosophers of Aristotle’s time believed anything (physical or metaphysical) could be categorized, Husserl’s views were more aligned with Kant in that tangible objects (outside of the abstract world) were to be the ones subject to formal logic. However, Husserl expanded his views of pure logic in his 1929 work, “Pure and Transcendental Logic.” Here, he expands the use of logic to include an “ideal structure” that expresses our propositional meanings [Corrazon, 2009]. This is a progression toward the inclusion of semantics in logic and ontology. Semantics convey the nuances of internal meaning and are dependent on structure.

Theories of ontology and theories of science naturally intertwine and complement each other. For example, 20th century scholar Quine (1908 – 2000) proposed that ontology (as a tradition) studies the theories of natural science, and that the natural sciences are our best sources for knowing reality [Mark et al., 2003]. He thought of objects as discrete and quantifiable, and therefore, entities that could be categorized in ontology. According to Quine, if one determines the variables used in scientific theory and these are the ontological commitments [Mark et al., 2003]. To him, ontology was the foundation and methodology for scientific theory.

If ontology is a sort of methodology for science, then one must determine the scientific domain for the ontological system. What categories, or objects, are to be included? What are the boundaries, what is quantifiable, and what vocabulary is to be used? Within the science of ontology this would be referred to as the “universe of discourse.” The universe would be the framework and its objects, or vocabulary, is what embodies the knowledge within the domain.

Ontology Today

According to Webster’s 3rd International Dictionary [http://www.merriam-webster.com], ontology is defined as “A science or study of being: specifically, a branch of metaphysics relating to the nature and relations of being; a particular system according to which problems of the nature of being are investigated.” Furthermore, as related to computer science or a knowledge based system, ontology is a “description (like a formal specification of a program) of the concepts and relationships that can exist for an agent or a community of agents” [Gruber, 1993] (parenthesis mine). Ontologies formally specify a domain theory (e.g., deformation) and knowledge about the world (e.g., shear zone) [Babaie, 2009]. In essence, ontology is about categorizing or defining things that exist, and the relationships between them.

In a philosophical sense, the purpose of an ontology is fulfilled with a simple categorization of reality. However, the computer science community has expanded the role of ontology to include the use of a formal vocabulary that is enhanced with the establishment of relational meanings. These relations facilitate data interoperability and exploration. Although ontology has progressed to include many disciplines, such as artificial intelligence, medicine, science, and commerce, today’s ontologists may enhance their work by drawing on historical philosophical principles. According to Smith [2003], information scientists can improve their ontological development by drawing on the work of various philosophies, such as the part-whole relationship analysis.

In essence, a synthesis has taken place in which the philosophical endeavors of categorizing reality has evolved to become a potential benefit for enterprises that are faced with an explosion of information and data, such as science. A distinction between developmental approach to ontology is given by [Sowa, 1999] when he states that “Philosophers often build their ontologies from the top down with grand conceptions about everything in heaven and earth. Programmers, however, tend to work from the bottom up. For their database and AI (artificial intelligence) systems, they often start with limited ontologies, or *microworlds,* which have a small number of concepts that are tailored for a single application.” These microworlds, or domains of interest, are the concepts to be included in a domain-specific ontology, such as a structural geology ontology.

According to Gruber [1993], ontology is a “specification of a shared conceptualization.” This conceptualization is dependent upon the domain one wishes to represent. Analysis, and the specification of its vocabulary regarding this domain, is dependent on the subjective view of the developer. Specifications are the vocabulary as used to define the area of knowledge. This vocabulary is influenced by the level of expertise of the developer regarding the domain of interest. Within the ontological science field an ontologist’s credo has emerged: “To create effective representations it is an advantage if one knows something about the things and processes one is trying to represent” [Mark et al., 2000]. The selected vocabulary must be structured in a way, and with adequate detail, so that the semantics of the subject matter are captured.

Ontologies are not limited to domain specificity. An ontology may serve a general purpose by means of a wide vocabulary of concepts that is analogous to a taxonomy. These are "upper level" ontologies that allow for extensive interoperability of many domain-specific ontologies. While the domain-specific ontologies describe the concepts and relationships of a particular area of interest, they have access to the shared concepts in the upper level ontology (e.g., temperature, time).

Because concepts within an ontology are organized in a specific structure, using a formalized grammar, computers are able to analyze the information within the ontology. For example, an application can determine whether a relationship is transitive ("ancestral inheritance", for example if a→b and b→c, then a→c), or whether it is the inverse of another relationship. These capabilities enable computers to perform operations on data [Alexander, 2007]. These computer-processable relationships will be discussed further in chapter 5.

Current Work in Ontology

The National Science Foundation's Office of Cyberinfrastructure stated that today's scientists need new information technology capabilities that can result in greater ability to answer complex questions [NSF, 2006]. Although ontological work in the physical sciences is still in an early stage, many projects have been completed, or are being developed.

The Semantic Web for Earth and Environmental Terminology (SWEET) is an upper-level ontology that provides a common semantic framework for various Earth science initiatives [Raskin, 2007]. This large scale project has been undertaken by NASA so that Earth science data and information will be more accessible and understandable. This ontology contains formal definitions for terms used in earth and space sciences and it contains associated properties with appropriate units to provide an extensible upper level terminology [Sinha et al., 2007]. The SWEET ontology is part of the Federation for Earth Science Information Partners (ESIP). ESIP (which includes the USGS in its network) collects, interprets, and develops applications for remotely sensed Earth observation information in order to make Earth science issues more available and understandable to researchers, educators, policy makers and the general public [http://esipfed.org/]. Numerous projects that involve ontology utilize the SWEET framework, such as: DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering), GEON (Geosciences Network), LEAD (Linked Environments for Atmospheric Discovery), GENESIS (Global Environmental & Earth Science Information System), and IRI (International Research Institute for Climate and Society).

Development of Geo-ontologies includes some unique characteristics and challenges. In addition to the customary ontological components (such as vocabulary, hierarchy, part-whole relationships, granularity), one must consider the complexity of the reality one is attempting to capture within the objects of a geologic domain. For example, geology contains not only real world objects, but also objects of “non-existence” (i.e., open fractures, pore). A geo-ontology must account for spatial considerations of objects, and that many of these objects exist in a dynamic state. As pointed out by [Mark et al., 2000], geo-ontological research is higher than standard ontological research in that it represents “a world of constant change, of multifarious causal processes at different levels of scale and granularity.”

In spite of the challenges, geoscience-related ontological development is significantly on the rise. For example, a collaborative effort is underway between the USGS and the University

Consortium for Geographic Information Science (UCGIS) to develop a National Map ontology. According to the USGS, this endeavor will improve and deliver topographic information to the Nation, in part to be utilized for many things ranging from recreation to scientific analysis to emergency response [USGS: http://nationalmap.gov/]. The National Map web service will provide geographic information such as aerial photographs, elevation, geographic names, hydrography, boundaries, transportation, structures, and land cover. Data from the National Map ontology can be merged with other data and can generate maps of various types or with varying views. Within the framework of this high level ontology, specific domain communities can map their own ontologies (vocabularies) to this upper level vocabulary.

David Warren and Fernando Pereira created a natural-language query system (ontology) that included a microworld of geographic concepts [Sowa, 1999]. This system was used for querying and reasoning through the use of hierarchies and their inheritance of properties (i.e., what is true for the parent class is true for the subclass), as well as through constraints on these properties, domain-database usage, and other methodologies. The following figure shows the hierarchy of concepts used for a section of their ontology (Figure 4.1):

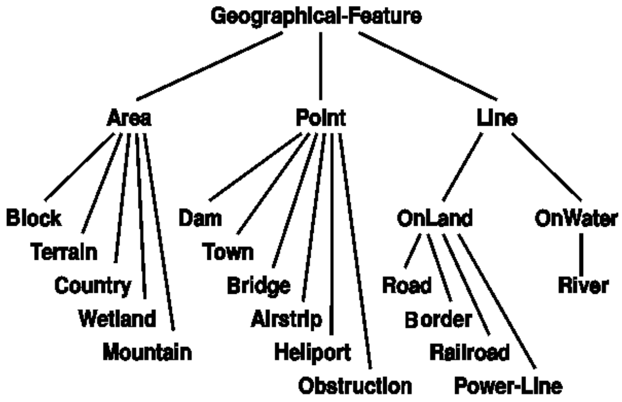


Figure 4.1. Geographical categories in the Chat-80 system [Sowa, 1999].

A Geospatial Trade Study was released in June, 2007 to the National Geospatial-Intelligence Agency. This study evaluated 45 ontologies that apply to spatial, event, and temporal granularity concepts in order to find the characteristics that are best suited to the geospatial intelligence community for uses in annotation, qualitative reasoning and interoperability. Note the following Geospatial Ontology Trade Study website for information regarding this study: [http://projects.semwebcentral.org/docman/?group_id=84#].

A search on the NSF website for funding that has been awarded to proposals with the word “ontology” in the title returned 34 matches. Some of these awards exceeded one million dollars in the amount granted for these projects. When the search was expanded beyond title, to include awards granted for subject matter dealing with ontology, the results showed 244 approved grants. These awards were granted solely from the NSF and did not include collaborative efforts, or awards from other institutions. The demand for research in ontology is evident.

The Motivation for Ontology

A well formed ontology accurately represents the concepts in its domain. The content must be inclusive enough to adequately express the reality of the domain, and the structure arranged in a manner that maximizes functionality and cohesion. If these criteria are met, then ontology is a means of clarifying the structure of knowledge within the domain [Staab, 2000]. This clarification of knowledge facilitates the communication of semantic nuances within the theories of the domain.

Large, upper-level ontologies can integrate existing domain-specific ontologies that describe the high level ontology [Noy and McGuinness, 2001]. Likewise, as pointed out by Noy and McGuinness [2001], the upper-level ontology can be used to describe the domain-specific ontology. Sharing of knowledge is reciprocal. For example, an upper-level ontology may have a component that represents time. In a structural geology ontology this could be used to facilitate the expression of causal events. This ability to integrate technologies, and to share these technologies among various communities, nourishes the ontological impetus.

Because an ontology represents a developer's view of some portion of the world, assumptions within the domain are made. These explicit assumptions may be evaluated, and the knowledge within the domain may be analyzed. If errors are detected, or new knowledge develops, changes are easy to incorporate (unlike working with many existing programming languages). Once a well developed ontology is in place, it may be open and accessible for other communities to use. These communities may commit to the ontology as it is written, alter portions of the content, or use only some section of the ontology. According to Gruber [1993], an ontological commitment is an agreement to use the vocabulary along the lines of using a theory specified by this vocabulary. The ontology may be used by people, databases, or applications that need to share in its domain information [W3C: http://www.w3.org/TR/webont-req/].

An ontology is capable of being more than a simple taxonomic hierarchy of concepts. It may also convey knowledge through inference capabilities and through logical axioms. According to the World Wide Web Consortium (W3C), "Using ontologies, tomorrow's applications can be "intelligent," in the sense that they can more accurately work at the human conceptual level." [W3C: http://www.w3.org/TR/webont-req/].

Ontologies are written in a machine readable format that computer programs can understand. The computer-understandable configuration makes for efficient information organization and retrieval, such as for search engine processes. Application of this resource can be used not only in domain specific fields, such as science, but also for uses such as searching the World Wide Web or for digital libraries. Additionally, ontologies can be used to provide semantic annotations for images or other non-textual objects [http://www.w3.org/TR/webont-req/]. Inclusion of these non-textual objects is further enhanced if the developer includes supplementary information or data about these objects (i.e., type of instrument used to capture an image, resolution, etc ..). From a pragmatic perspective, the utility of ontology is defined by its use [Gruber, 1993].

Challenges to Ontology

Ontologies developed by different communities will inevitably vary in quality. If inadequate analysis was performed on the domain then the ontology may be organized in an inefficient or incorrect manner, for example. Prior to the decision to commit to an ontology, it needs to be inspected to determine if it is correct and if it is appropriate for the user. Movement toward good ontology can be an incremental process of theory construction, criticism, testing, and amendment [Smith, 2003].

A problem for upper-level ontologies, or in the case of shared ontologies, is what Smith [2003] refers to as the Tower of Babel problem. He explains that this occurs when different groups use identical terms with different meanings, or when different terms are used to identify the same object. Smith goes on to state that methods are needed, such as standards, or else this problem increases geometrically. For this reason many upper-level ontologies are domain-independent and include only neutral concepts such as time, measurements, quantity, etc.

According to Noy and McGuinness [2001], ontology has been touted as too arcane for ordinary users to understand and, therefore, has failed to gain popularity in the mainstream until recently. One of the major problems was the lack of standards for approaching ontological development. There were many different proprietary ways to approach ontology and an inadequate number of tools with which to develop systems [Noy and McGuinness, 2001]. Interoperability issues have become apparent due to challenges of semantic agreement, system transferability, and application language compatibility. All of these issues underscore the need for standards. To address these types of challenges, the World Wide Web Consortium (W3C) has come together to establish standards in Web technology. The W3C has suggested specific formalisms for encoding ontologies [W3C: http://www.w3.org/TR/owl-features]. This will be discussed in detail in the following chapter.

CHAPTER 5

SEMANTIC WEB TECHNOLOGY ARCHITECTURE AND STANDARDS

The World Wide Web has provided a tremendous resource for humanity in recent years. A particular benefit has come from the availability and advancement of information. Along with this growth has come the challenge of organizing and effectively retrieving the information. For this reason significant attention has shifted to understanding scientific principles of organizing the vast amount of information [Little, 2003]. Ontology and knowledge based systems serve as the perfect vehicle for accommodating this organization of information.

The growing popularity of ontology is evident in many areas, such as business (i.e., categorical ontology, such as for Amazon.com), medicine (i.e., the Gene Ontology), science, and many other disciplines. Ontological interoperability and reusability are critical features in the expansion of ontological use. These features are some of the key focus issues addressed in the evolution of the next generation World Wide Web, referred to as the semantic web. The semantic web has broadened the task of classifying, labeling, defining, locating, integrating, and using everything on the World Wide Web (WWW), and has begun the process of integrating knowledge [Sowa, 1999].

The Semantic Web

The focus of the current WWW is to serve as a publication resource predominantly for simple documents in human-readable form. These documents are transferred to the web and connected with hyperlinks through the use of Hypertext Transfer Protocol (HTTP) and Hypertext

Markup Language (HTML), respectively. The documents are accessible through the use of Uniform Resource Locators (URLs) (e.g., http://www.gsu.edu). Although the WWW is a rather new technology, its growth has been remarkable and many people depend on its availability.

Functionality of the web is tremendous, but holds potential much greater than merely human readable documents in web browsers [Lacy, 2005]. The goal of the semantic web is to shift from document-centered operability toward data-driven operability that software systems can understand and interpret. In the same way as the WWW is a huge distributed hypertext system, the semantic web is intended to form a huge distributed knowledge based system [Obitko, 2007]. As noted by [Beckett, 2005], this will require the creation of a common framework that allows data to be shared and reused across applications through the use of tools that may reveal new relationships among data. With the explosion of growth in many fields, such as science, great benefit could be wrought from the potential to integrate and automatically process information content. However, syntactic format in the traditional markup languages does not allow for machine-processable interpretation, or the ability to integrate information.

In order to accommodate machine-readable, automatic processing by computers, information must be represented in a structured manner for computer parsers [Lacy, 2005]. This progression of structured applications in technology enables information to be enhanced with semantic descriptions. It also allows for inference, or the ability to garner knowledge that is not evident from explicit facts. While the current internet operates using search engine technology based on keyword matching, the semantic web will allow for direct answers to queries based (in part) on their semantic interpretations and inference capabilities. These innovative technologies of the semantic web are evolutionary extensions of the current web and do not require an overhaul of the tremendous work that has already transpired on the web.

Challenges to the evolution of the semantic web exist, such as the interpretability of languages used to present information. A language that is machine-understandable and provides syntax, semantics, and standards is required [Lacy, 2005]. In an attempt to resolve this issue, the World Wide Web Consortium (W3C) (directed by Sir Tim Berners-Lee, inventor of the current web) has recommended semantic markup language standards. The architecture of the semantic web system, as defined by the W3C, is affectionately known as the "layer cake" (Figure 5.1). Each layer in the architectural stack ("layer cake") is dependent on the one below it. The first layer, URI and Unicode, follows the important features of the existing WWW [Obitko, 2007]. These layers will be discussed in detail in the sections that follow.

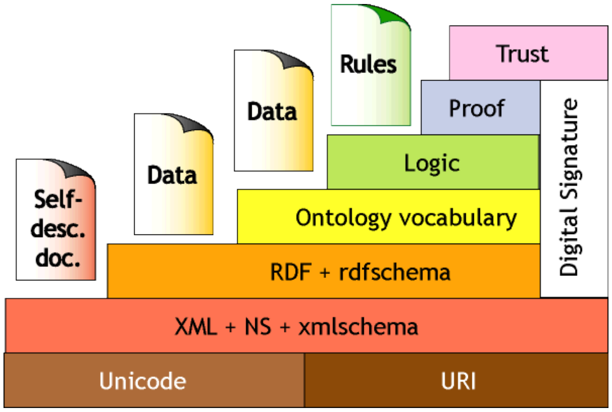


Figure 5.1. W3C's semantic web stack architecture, referred to as the "layer cake", in which each layer is a foundation for the one above it [W3C, XML slide, 2000].

Resource Identification

Unicode is a standard way of assigning a number for every character in any language. It serves as the foundation for most platforms and languages in today's technological arena. Because computers basically operate using numbers, this standard makes computer interaction global.

Uniform Resource Identifiers (URIs) enable identification of resources on the web. Regardless of what one wants to discuss on the web, it must first be identified. A uniform system of identification allows for each resource to be uniquely identified. URIs are the foundation of the web [Swartz, 2002]. They are a set of character strings that use a generic syntax in order to identify resources in a manner that can be used to network with other applications.

A resource is conceptual and can be anything one wishes to identify, real or arbitrary. Level of granularity can be vague or precise. For example, a vague identification of a resource could be "the stratigraphy within the fault zone." More precise would be, "the Jurassic-Cretaceous Franciscan formation of the San Andreas Fault at Parkfield." Functionality of the URI depends entirely on the context of the application.

In order to locate the resource named by the URI, a Uniform Resource Locator (URL) is used. The URL, a type of URI, is a resource locator. Whereas a URI can name anything, web accessible or not, a resource for the URL must be located on the web. In essence, it is the address where the resource resides on the web, such as http://codd.cs.gsu.edu:9999/safod/. URLs are most commonly used to retrieve web documents (e.g., HTML). Both URIs and URLs are a foundational layer of the current web and the semantic web.

XML, Namespaces, and XML Schema

XML, eXtensible Markup Language, is the syntactical layer in the semantic web architecture. It provides a consistent, standardized, computer-parseable format that is needed to underpin a Semantic Web language [Lacy, 2005]. It is rapidly taking the place of HTML to provide a uniform way to send documents across the web. It allows everyday users to "mark up" documents to the web in a more standardized syntax. Several of today's mainstream corporations (i.e., Microsoft, Adobe, Netscape) have come to adopt these progressive new standards.

Documents can be formatted in XML in a manner that allows for (machine-readable) emphasis on certain words. This is done by providing tags along with the content one would like to emphasize. There is a start tag and an end tag, formatted in brackets, with the domain content between the brackets. For example, '<mineral type> plagioclase feldspar </ mineral type>' tells the computer that plagioclase feldspar is a mineral type, though the computer only interprets this as it does any other character string set. The tags, along with the content, form what are called elements in XML. The content within the tags can consist of nested tags, essentially creating a hierarchy of content information.

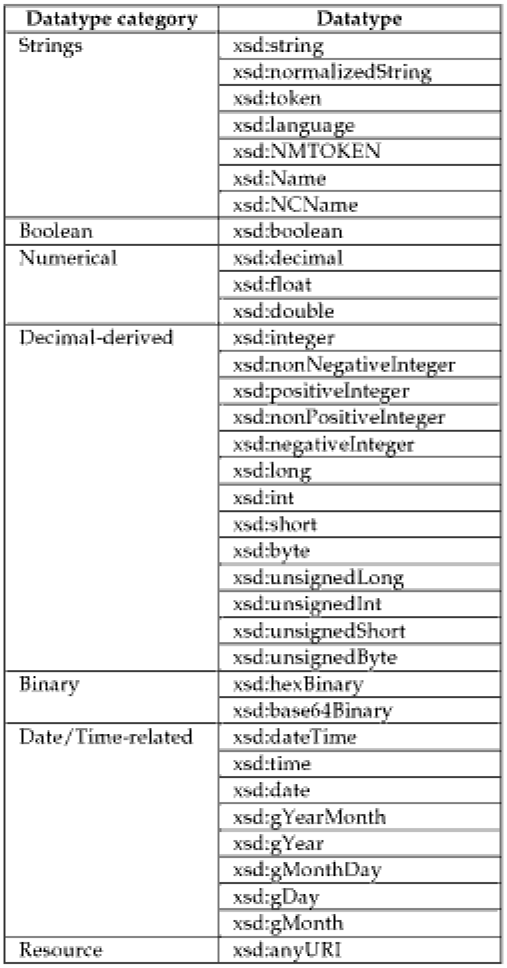
XML is actually a metalanguage (a language about a language) because it consists of tags (domain information) about the document content. It is machine-readable, making data interchange possible. The documents created in XML can be used by many applications, with each application only using the markup (content) it understands [Swartz and Hendler, 2001].

XML statements can be further enhanced with the addition of attributes, or values. Attributes provide additional information that may not be included in the data. These are distinguished in XML code by the use of quotes. For example, '<rock type = "igneous">'. Unique identifiers are assigned to each element and attribute through the use of XML Namespaces that assign unique URIs. A namespace (contains prefix 'XMLNS') is simply a way to identify a "space" on the Web from which meaning of the names is derived [Swartz, 2002].

XML provides the syntax for structured documents, but it does not provide semantic meaning within the documents. In other words, XML allows users to add structure to their documents but says nothing about what the structure means [Berners-Lee, 2001]. According to Jon Bosak ("father of XML"), it simply "clears away some of the syntactical distractions so that we can get down to the big problem: how we arrive at common understandings about knowledge representation." [Lacy, 2005]

XML Schema (XMLS) is an extension of XML. It provides further structure and other important features geared toward ontology and the semantic web. XMLS defines datatype features (e.g., string, decimal, float, Boolean) [Lacy, 2005]. These are identified with the prefix of "XSD." The following table shows the XMLS datatypes that are used by the Web Ontology Language discussed later in this chapter (Table 5.1).

Table 5.1. XML Schema Datatypes [Lacy, 2005].



Datatype category	Datatype
Strings	xsd:string
	xsd:normalizedString
	xsd:token
	xsd:language
	xsd:NMTOKEN
	xsd:Name
	xsd:NCName
Boolean	xsd:boolean
Numerical	xsd:decimal
	xsd:float
	xsd:double
Decimal-derived	xsd:integer
	xsd:nonNegativeInteger
	xsd:positiveInteger
	xsd:nonPositiveInteger
	xsd:negativeInteger
	xsd:long
	xsd:int
	xsd:short
	xsd:byte
	xsd:unsignedLong
	xsd:unsignedInt
	xsd:unsignedShort
	xsd:unsignedByte
Binary	xsd:hexBinary
	xsd:base64Binary
Date/Time-related	xsd:dateTime
	xsd:time
	xsd:date
	xsd:gYearMonth
	xsd:gYear
	xsd:gMonthDay
	xsd:gDay
	xsd:gMonth
Resource	xsd:anyURI

Although XMLS compliments XML with added structure and functionality, it is not adequate for interpreting semantics. Semantics are necessary in order to interpret the constructs in a software language, and to develop reasoners that will perform uniform inferences [Gil and Ratnakar, 2002]. Computer applications may parse the standardized content, but another layer is needed in order to merge information (data) and establish relationships. The next layer in the Semantic Web stack addresses these requirements.

RDF

Resource Description Framework (RDF) provides a standardized data model of resources and their relationships, to include basic semantics. It provides a methodology for expressing knowledge in the decentralized sphere of the World Wide Web. Its manner of relationship establishment and interpretability enable the computer to process information in a way that makes the computer appear to understand content. Utilizing this standard, applications are able to make use of distributed, structured information spread throughout the Web [http://www.rdfabout.com/quickintro.xpd]. In essence, RDF advances application functionality from "machine-readable" to "machine-*processable*." It provides *meaning*.

According to the W3C [http://www.w3.org/TR/owl-features], RDF was developed from the motivation to: provide information about web resources (i.e., web metadata, such as author), to do for application data what the web did for hypertext (i.e., allow data to be processed outside the environment in which it was created), allow for interworking among applications and automated processing of the information or data. To meet these goals RDF is based on a simple data model that can accommodate the structure of most data. It uses a formal semantics that provide a dependable basis for reasoning about the meanings within the RDF expressions, with reliable rules for inference within the data [http://www.w3.org/TR/rdf-concepts/].

The previously mentioned technologies and languages (URI and XML) allow a user to make statements and uniquely identify them. However, this means nothing to a computer. RDF makes it possible to make statements that are *machine-processable* [Swartz, 2002]. For example, several investigators may post data regarding serpentinite found in their samples. Another application may then use this information to order the samples according to strata and construct a profile of the region based on serpentinite location.

The standard format that RDF uses to make statements (usually in XML syntax) about resources is expressed in simple sentences called *triples*. Triples are written as subject-predicate-object, often through the use of XML tags. A triple is also known as an RDF graph. Usually there are sets of triples because there are multiple resources (objects) to be described for a specific object. An XML tag is used to specify the transition to RDF. All code between the RDF tags conforms to RDF standards (e.g., <rdf:RDF> … </rdf:RDF>).

This approach provides a general method to decompose knowledge into small pieces, with rules about the semantics of those pieces [Tauberer, 2005]. Within an RDF triple, the subject is the resource, the predicate is the relationship description, the object is a datatype or another resource, and the framework is the combination of Web-based protocols (i.e., URI, XML) that RDF is built upon [Pollock, 2009]. For example, ThinSection (subject) has_mineral (predicate) Serpentinite (object).

The predicate is a type of property that describes the relationship between the subject and object. The object may be a literal (e.g., data value) or a resource value of the property for the subject resource [Lacy, 2005] (Figure 5.2). If an object is a literal, it is an actual value rather than a resource. Literals are specified using the lexical form (Unicode characters) of literals for the datatype [Pollock, 2009]. These datatypes are a defined standard, such as the one for an integer [W3C: http://www.w3.org/TR/xmlschema-2/].

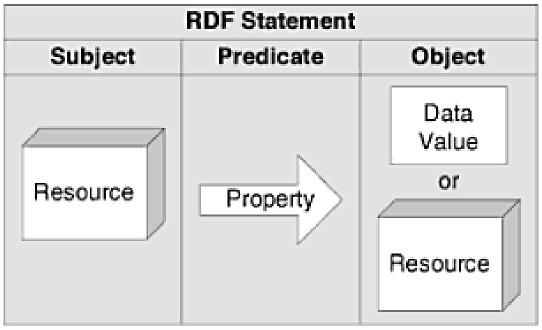


Figure 5.2. RDF subject-predicate-object statement components [Lacy, 2005].

Any resource can be written about and sometimes multiple terms can have the same meaning, therefore, resources need unique identifiers to reference them. URIs are associated with each resource or property identified and allow for global referencing. They ensure that concepts are tied to a unique definition [Berners-Lee, 2001]. In the prior example, each part of the RDF statement can be described with the URIs in Table 5.2. A URI may coincidentally be the same name path as a URL, but does not have to be. Its primary purpose is to be a unique identifier.

Table 5.2. Examples of URIs that could point to the RDF triple “ThinSection has_mineral Serpentine.”

Subject	http://www.owl-ontologies.com/SafodOntology.owl#ThinSection
Object	http://www.owl-ontologies.com/SafodOntology.owl#has_mineral
Predicate	http://www.owl-ontologies.com/SafodOntology.owl#Serpentine

URIs can be abbreviated using namespaces, called qnames. A URI expressed as a qname has two parts: namespace and identifier, separated by a colon [Allemang and Hendler, 2008]. For example, the qname for FaultRock (identifier) could be 'frock', mapped to 'http://www./SafodOntology/FractureRock.owl#'. The URI is separated from the resource using the # character. This is useful for assigning multiple resources to an 'http//www…' notation (which represents a triple). The XML namespace feature establishes a standard identifier. With *rdf* namespace, a property of 'rdf:type' provides an elementary typing system in RDF [Allemang and Hendler, 2008]. These features can be used to express a relationship among multiple subjects (resources) that have the same object (Table 5.3).

Table 5.3. Using rdf:type to describe fault rocks (adapted from Allemang and Hendler, 2008, Table 3-9).

Subject	**Predicate**	**Object**
frock:Breccia	rdf:type	frock:FaultRock
frock:Cataclasite	rdf:type	frock:FaultRock
frock:Mylonite	rdf:type	frock:FaultRock

There are instances when it is desirable to represent a resource that does not have a Web space. For example, an unconformity in the strata of a fault zone has missing (unidentified) bedding. This is addressed with a blank node, using the '?' character. For example, '? rdf:type strata:Bedding'. The subject is represented as a blank node. This code establishes that there is an unknown existence (of Bedding).

Although RDF may be represented in a variety of formats, XML is the preferable representation scheme for standardized interoperability [Lacy, 2005]. The combination of XML syntax, tags, namespaces, and URIs for unique referencing allow data to be mapped using the relationships within the RDF triples. This facilitates interoperability and reuse of data. The machine-readable code in the RDF standard allows other intelligent applications to locate and use the resources within this application.

RDF provides a fundamental layer toward knowledge representation on the Semantic Web. Its inherent purpose is to model data that may or may not come from multiple sources. It exploits XML to allow for namespaces, and the linking between namespaces, to provide a methodology for sharing machine-processable knowledge [Staab et al., 2000]. It is a rapidly growing, easily accessed format that allows for sharing across many applications. It provides a basic understanding of semantics through limited use of inference.

RDF provides the foundation for the Semantic Web. However, a more complete vocabulary is needed with more detailed relationship establishment to realize the next generation web. With RDF, there remains a limitation for computers or humans to figure out what specific terms mean or how they should be used [Swartz, 2002]. The RDF Schema provides an extensive vocabulary that further describes those relationship meanings.

RDFS

RDFS (Resource Description Framework Schema) is a vocabulary description language. It offers a vocabulary to model class and property hierarchies that can be referred to from RDF models [Staab et al., 2000]. It describes properties and classes of RDF resources, along with the general hierarchies of the properties and classes. RDFS expands the capabilities of RDF through its formal concept of classes within the modeling system. While RDF is a language that models resources on the web, it does not describe what these resources or properties are. RDFS describes the relationships between resources and between properties.

The vocabularies, based on RDF/XML syntax, describe domain-specific vocabularies using collections of RDF resources with explicit semantics [Lacy, 2005]. With RDFS a class-subclass type hierarchy can be created within the domain, upon which reasoning can be performed to make inferences. These inferences can uncover facts or establish relationships that are not explicitly stated.

Within a class-subclass hierarchy, some of the properties of subclasses are inherited from the parent class(es). For example, if StrikeSlip is a subclass of the class Fault, then all properties of Fault are inherited by the subclass StrikeSlip. In other words, if some properties of StrikeSlip are also complete properties of Fault, then StrikeSlip is a subclass of Fault. Furthermore, if RightLateral is a subclass of StrikeSlip, then some of its properties are inherited from both StrikeSlip and Fault. Members of a class are called instances of the class.

Classes are specified with the '<rdfs:Class>' notation and subclasses with '<rdfs:subClassOf>. Classes are considered to be resources. Classes can be further described by their properties, and properties may contain subproperties. Properties are denoted by the '<rdf:property>'. These describe the relationship between subject and object (predicate), as in RDF. Within RDFS, however, properties may be constrained by a domain and a range.

Properties can be described in terms of the classes of resource to which they apply. This is the role of the domain and range mechanisms [W3C: http://www.w3.org/TR/rdf-schema/]. The domain of a property states that any resource that has a given property is an instance of the class, while range of a property states that the values of a property are instances of the class [Obitko, 2007]. An intersection of classes is allowed when multiple classes of a property share the same domain and range. For example, the property has_mineral may have multiple classes (e.g., Rock, Gouge, and Particle) all of which have the class Mineral as their range.

Overall, RDFS is a vocabulary for describing the classes and properties of resources with some general semantics established in the hierarchical structure. However, according to W3C, RDF and RDFS are intended to provide a basic foundation for more advanced assertional languages. The RDF(S) vocabulary for describing classes and properties can be built upon with more facilities for expressing meaning and semantics through the Web Ontology Language (OWL) [W3C: http://www.w3.org/TR/rdf-mt].

Building the Vocabulary with Web Ontology Language (OWL)

In order for ontologies to be widely accessible, interoperable, and reusable, standards are essential. The World Wide Web Consortium (W3C) has recommended Web Ontology Language (OWL) as a standard for representing information and deploying ontologies onto the web. According to the W3C, OWL is a language for defining structured, Web-based ontologies which enable richer integration and interoperability of data across application boundaries [Daly, 2003]. OWL provides the logical layer in the Semantic Web's architecture and builds on RDF(S) with additional language features that describe ontologies [Lacy, 2005]. With OWL, applications can perform functional reasoning on resource concepts.

In order to be web-compatible, OWL uses URIs for naming conventions that identify a resource, and the RDF language to provide a standard syntax so that data within the URIs can be in exchangeable, machine readable format. RDF provides descriptions of resources, and RDFS combines those descriptions into a vocabulary, and ontology (OWL) develops the subject (domain) specific to the vocabulary [Daly, 2003].

OWL adds more vocabulary for describing properties and classes, such as greater specialization of the relationships between classes or values. For example, disjointness (a instance can be a member of one class or the other, but not both), cardinality (i.e., “exactly one”), equality, and characteristics of properties (i.e., inverseOf, symmetry). According to IBM’s Alphaworks [http://www.alphaworks.ibm.com/contentnr/semanticsfaqs], OWL goes beyond RDFS with the following features: may limit properties with respect to number and type, may infer that item with various properties are members of a particular class, may determine if all members of a class will have a particular property or if only some of them might, may distinguish one-to-one from many-to-one or one-to-many relationships, may express relationships between classes defined in different documents across the Web, may constrain range and domain to specific class or property combinations, and may construct new classes out of the unions, intersections, and complements of other classes.

Three varieties of OWL exist with a varying level of expressivity. From least expressive to most, these dialects are: OWL Lite, OWL DL (Description Language), and OWL Full. OWL Lite supports reasoning, but offers minimal features. It is a good choice for users who mostly want a classification hierarchy with simple constraints, and who do not need complex semantics. OWL DL offers a great degree of expressiveness with some degree of restriction on which OWL constructs may be used. OWL DL is based on Description Logics, a field of research that has studied the logics that form the formal foundation of OWL [W3C: http://www.w3.org/TR/owl-features]. OWL Full is the complete OWL language and offers maximum expressiveness. RDF constructs and OWL Full are completely interchangeable with no restrictions between the two. However, most software application reasoners would be unable to support the complete package of features within OWL Full. Additionally, computation time could be greatly increased with the use of OWL Full. For this thesis project, OWL DL was selected for development and will be the focus of this chapter.

The complete vocabulary of OWL Full is also available in OWL DL. OWL DL incorporates certain restrictions that make reasoning easier. For example, a class cannot be an instance of another class. Also, a class cannot be a property and a property cannot be a class. Properties must be either datatype properties or object properties. Datatype properties are relations between instances of classes and RDF literals, while object properties are relations between instances of two classes [W3C: http://www.w3.org/TR/owl-features].

OWL DL shares some features with RDFS, such as class, subclass, property, subproperty, domain, range, and individual. Classes are meant to represent groups of individuals that share characteristics (or properties). Properties are used to establish relationships either between individuals (subject and object) or between an individual (subject) and a data value. The subclass, subproperty, domain, and range features were described in the previous RDFS section.

Once a property has been defined, it may be desirable to describe further characteristics about the property. Some supplementary features for this purpose are: ObjectProperty, DatatypeProperty, inverseOf, TransitiveProperty, SymmetricProperty, FunctionalProperty, and InverseFunctionalProperty. (ObjectProperty and DatatypeProperty were described previously.)

If one property is the inverse of another property, the feature '<owl:inverseOf>' may be used. Properties are simply binary relations between a domain and range. An inverse relationship relates members of the range back to the domain [Lacy, 2005]. It is a symmetrical relationship (figure 5.3). For example, if has_instrument is the inverse of instrument_for, and

Petrographic Microscope instrument_for Aachen University, then a reasoner can deduce that Aachen University has_instrument Petrographic Microscope.

Figure 5.3. The inverseOf property [Lacy, 2005].

If a property is declared to be transitive ('<owl:TransitiveProperty>'), then its characteristics are inherited from its ancestors. This is a geometric concept that states that if a transitive function relates X to Y, and Y to Z, then X is related to Z by the same function [Lacy, 2005] (Figure 5.4). This feature can only be applied to object property types and cannot have a maxCardinality = 1 restriction (explanation of maxCardinality below).

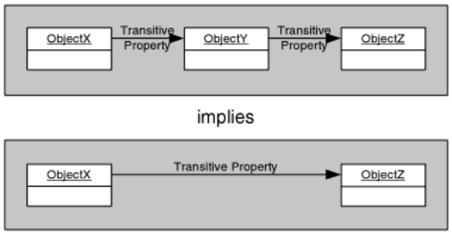


Figure 5.4. Transitive properties [Lacy, 2005].

A property relates a subject to an object. If the property is symmetric ('<owl:SymmetricProperty>'), then it also relates the object back to the subject (Figure 5.5). This is an inference shortcut, and only one direction needs to be specified [Lacy, 2005]. For example, if Babaie has_CoAuthor Andreani, then a reasoner can deduce that Andreani has_CoAuthor Babaie.

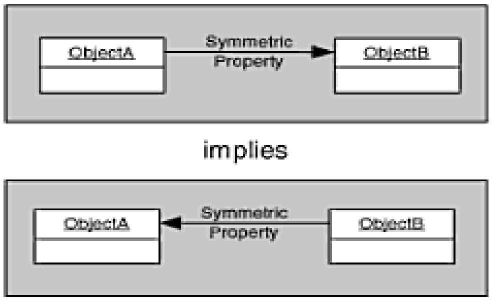


Figure 5.5. Symmetric property [Lacy, 2005].

If a property contains a unique value, then it can be restricted by the Functional Property, '<owl:FunctionalProperty>'. This means that it can have at most one value for each individual. It may have no values, but cannot have more than one. This characteristic has been referred to as having a unique property [W3C: http://www.w3.org/TR/owl-features]. For example, the has_ID property may be restricted as Functional. A reasoner may then deduce that no individual may have more than one ID. This type of restriction may be applied to both object and datatype properties (Figure 5.6).

Figure 5.6. Functional property [Lacy, 2005].

The opposite property restriction of Functional is the InverseFunctional Property '<owl:FunctionalProperty>'. If a property is functional then the inverse of the property is functional. This is used to identify object properties whose values uniquely identify the subject instance of the property [Lacy, 2005]. If the value of this type of property is known, then a reasoner can deduce the subject (Figure 5.7).

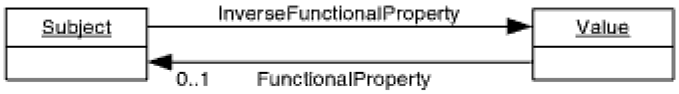


Figure 5.7. InverseFunctional property [Lacy, 2005].

In addition to describing additional property characteristics, OWL also allows restrictions to be placed on properties. The restrictions may be such that a limitation is placed on which values can be used for a property, or how many values may be used. For a restriction on which values may be used, there are two options: allValuesFrom or someValuesFrom. These options restrict a properties range with respect to its class. For a restriction on how many values may be used, a property could have cardinality restrictions. These cardinality restrictions are: minCardinality, maxCardinality, and cardinality. For a quick overview of these property restrictions, refer to Table 5.4 below.

Table 5.4. Property Restriction Overview.

Property Restriction	Definition
allValuesFrom	All property values belong to the associated class. Allows for no property instances as well.
someValuesFrom	At least one of the property values belongs to the associated class.
minCardinality	The associated class has the minimum value specified by the property requirement.
maxCardinality	The associated class has the maximum value specified by the property requirement.
cardinality	Useful for stating an exact value (i.e., min and max are = 1).

OWL DL also allows for logical class axioms, to include Boolean expressions. Axioms provide logical information upon which properties and classes can be reasoned. Examples of class axioms are disjointWith and equivalentClass. Boolean expressions deal with class combinations and sets, such as unionOf, complementOf and intersectionOf.

A class is sometimes described as disjoint if it is an “either/or” situation [Lacy, 2005]. An individual may belong to one class or the other, but not both. For example, the class SinistralStrikeSlip would be disjoint with DextralStrikeSlip. An equivalent class shares the same instances, and is useful if more than one term may be used for the same resource (concept). For example, sinistral and left-lateral strike slip.

The Boolean logical operator ‘<owl:intersectionOf>’ is used when multiple classes are combined to form a new class concept. The new class includes all attributes and values of the included classes. It is equivalent to “AND” in logic, and can be used to specify a necessary condition [Lacy, 2005] (Figure 5.8). If ‘<owl:unionOf>’ can be used to state that a class contains instances that are either in one class, in the other class, or both. The ‘<owl:complementOf>’ feature is used to identify instances that do not belong to a class.

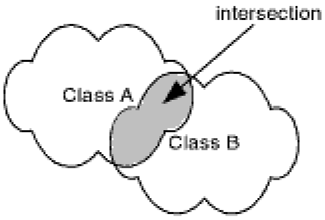


Figure 5.8. The intersectionOf Boolean function [Lacy, 2005].

The property characteristics, restrictions, and logical operators that OWL builds on the XML/RDF(S) provide the essential components for the Semantic Web vision. These languages, designed by Web experts, are distributable and accessible. Development in these languages has been made easier by tools that facilitate their use. One such tool, the Protégé editor, was used to construct the SAFOD ontology. Protégé provides a suite of tools to construct domain models and knowledge-based applications with ontologies [http://protege.stanford.edu/overview]. The use of Protégé is covered in detail in the Development section of this thesis (chapter 6).

CHAPTER 6

THE SAFOD ONTOLOGY PROJECT
DESIGN AND DEVELOPMENT

Prior to development of the ontology, the SAFOD project had been underway since initial drilling began in 2002. As previously mentioned, drilling of core samples continued through the summer of 2007. The cores samples addressed by the ontology for this project were taken from sidetrack E (10,306 ft region) and sidetrack G (10,480 ft and 10,830 ft regions).

As a result of the successful retrieval of core samples, EarthScope/SAFOD received 28 proposals (representing 98 scientists from around the world) to work on SAFOD Phase 3 samples. According to the Earthscope website for distribution of core samples, these proposals contained approximately 790 requests for SAFOD core and cuttings samples [http://www.earthscope.org/es_doc/safod/SAFOD_Core_Sample_Distribution.pdf]. Demand for samples greatly outweighed availability.

SAFOD core and sample requests were sent to the SAFOD Sample Committee (SSC) for consideration. The members of SSC were selected because of their expertise in microstructures, mineralogy/geochemistry, rock mechanics and core handling and curation. These members had no personal involvement in SAFOD core sample research. The committee determined which samples would be allocated for current research, and which ones would be reserved for future study.

The SSC decided that not more than 1/3 of the core would be distributed at this time, with priority for non-destructive research. From the requests, approximately 190 samples were distributed to 18 investigator groups. Of the approved investigators, several were interested in the SAFOD ontology for use during their scientific research.

Prior to development of the ontology, Dr. Hassan Babaie, who was supported by the NSF to develop the ontology and related knowledge base is a member of one such group of SAFOD investigators. The funded NSF proposal (0545472) is entitled: 'Microstructural analyses of gouge from the San Andreas Fault Observatory at Depth (SAFOD) borehole in relation to brittle fault mechanics'. He was able to ascertain initial requirements in relation to the objectives of the rest of the Gratier Group investigators in Grenoble, France in 2008. The list of the investigators in this group can be found at the SAFOD web-interface: http://codd.cs.gsu.edu:9999/safod/. This requirement acquisition conducted by Babaie was the initial step toward determining how to best design the ontology that would meet the needs of the investigators. Some of these investigators also provided feedback via email after the Grenoble meeting. In addition, online access was available for the reported objectives and current study results of some investigators. These reports had mostly been presented at an Earthscope meeting in Boise, Idaho during May, 2009 and were available on the Earthscope National meeting website [http://www.iris.edu/hq/esreg/page/abstracts]. These initial study results and objectives were carefully considered when developing concepts for the ontology. The lithology and properties of the core samples approved for their research was also taken into consideration. These descriptions were used to facilitate the establishment of concepts within the domain of the ontology.

Design and development of the ontology occurred in incremental phases. These phases included, from initiation of the SAFOD ontology project to the final completion: research on the study site for the SAFOD project (Parkfield, CA), research of SAFOD documentation (i.e., on core samples described in the Atlas), determination of objectives and results of investigators, determining concepts (classes) from the research, documentation, and investigator objectives, arranging the classes (defined concepts) in a taxonomic hierarchical structure, determining the properties and property types related to each class (as well as the relationships between the concepts), using the Protégé editor tool to develop classes and their respective properties in the ontology application system, enriching the classes and properties with particular property characteristics, and population of the ontology with investigator data through the application interface. It should be noted that at the time of writing of this thesis, most of the data set from the investigators is still not available. This is mainly due to the technical issues to make the core available to the investigators. Through the interface, the users were then able to view data (their own as well as the data input by other investigators), manipulate their own previously entered data, and query the system to find relationships among the data.

The Design Phase

Information about the cores that our investigators would be researching was collected from the Photographic Atlas of the SAFOD Phase 3 Cores public data records on the ICDP website [http://www.icdp-online.org/upload/projects/safod/phase3/Core_Photo_Atlas_v3.pdf]. This information was used in an initial effort to gain an understanding about the lithologic characteristics of the core samples. This knowledge was used to establish an initial model of class concepts for the ontology. It also provided further information regarding the geology of the study area, as well as a foundation to better understand the user objectives (that would be considered next).

The cores were obtained from the Phase III drilling project that occurred in 2007. These boreholes were drilled in multilateral directions off of the main borehole (Figure 6.1). The intention of the selected locations was to retrieve continuous core samples from the active fault zone. These cores were then differentiated by Hole, Run, and Section, based on their respective coring location. Two successful coring runs were conducted within sidetrack 'Hole E'. Within sidetrack 'Hole_G', three coring runs were made across actively creeping faults.

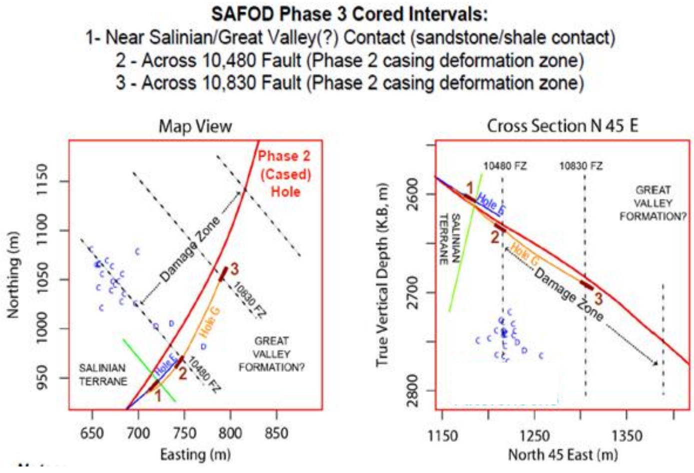


Figure 6.1. Phase 3 sidetrack holes E and G, taken from the SAFOD Core Atlas [http://www.icdp-online.org/upload/projects/safod/phase3/Core_Photo_Atlas_v3.pdf].

The first samples described were taken from Hole E, Runs 1 and 2. These core runs were extracted from measured depths between 10,306 ft and 10,346 ft. and were then divided into sections. An image of the core from Hole E, Run 1, Section 1 is given as an example in Figure 6.2. Images of all cores retrieved for the SAFOD project can be viewed on the Earthscope website at [http://www.earthscope.org/data/safod_sidetrack_e].

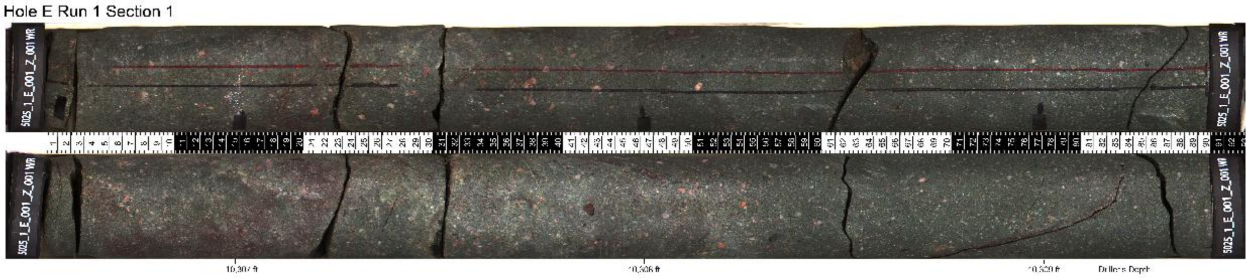


Figure 6.2. Screen shot of image of one core run sample from Hole E, Run 1, Section 1 from the Earthscope Core Viewer [http://www.earthscope.org/data/safod_sidetrack_e].

This section of the thesis report provides a basic summary of the lithologic descriptions of the core runs. These summaries are from the Photographic Atlas on the Earthscope website at [http://www.icdp-online.org/upload/projects/safod/phase3/Core_Photo_Atlas_v3.pdf]. The purpose of these summaries is to make more clear how the ontology classes were formulated. The cores described are from Phase 3 core run drilling holes E and G.

Hole E, Runs 1 and 2 were delineated into sections within the core runs. SAFOD scientists determined that the top of section 1 to the middle of section 4 (10,306.5 – 10,316.8 ft) consisted mostly of a greenish gray pebbly sandstone with coarse, subangular to subrounded sand. The grains were equant to slightly elongate, dominantly feldspathic, and up to 2.5 cm in diameter.

The lower half of Section 4 to nearly the bottom of Section 5 (10,316.8 – 10,320.9 ft) was dark grayish-black siltstone. The siltstone was fractured and displayed a weak, scaly fabric. The majority (~90%) of this region was mesoscopically homogeneous silt and clay size particles. The remainder was composed of two subunits of fine to medium sands with pebbles less than .5 cm in diameter. The coarser subunit (approximately 10 cm thick within the core section) was greenish-black in color. This subunit was from a fault contact between a light olive-gray unit and underlying grayish-red pebbly sandstone. The clasts in the coarse units were predominantly feldspathic, with a few thin non-quartz silicate veins (< 1 mm).

A grayish-red to brownish-gray pebbly sandstone existed between the fault contact near the base of core Run 1, Section 5 and the bottom of core Run 2, Section 6 (10,320.9 – 10,343.2 ft). The matrix consisted of coarse to very coarse subrounded, elongate sand with clasts up to 3 cm in diameter. Again, these were dominantly feldspathic with some iron-oxide staining. Subparallel bedding (20 – 30 degrees to the core axis) was defined by grain size variation. The unit was crosscut by fractures and mesoscale faults consisting of layers of cataclasite (up to .5 cm thick).

The second core interval was from Hole G, core Runs 1, 2 and 3. From the top of Hole G, Run 1, Section 1 to the middle of Run 2, Section 4 (10,455.2 – 10,478.8 ft) was a foliated siltstone-shale cataclasite. In the finer-grained sections the cataclasite foliation had a scaly fabric with cm-thick color banding. The fabric was formed by elongate, irregular-shaped lenses and porphyroclasts of siltstone with fine-to very-fine grained sandstone. The lenses and the porphyroclasts were in a fine-grained calcite cement matrix, possibly including pyrite. There were many thin, short calcite veins oriented at a high angle to the foliation.

The hardest rock retrieved from Phase 3 coring was a massive grayish-black shale that extended from the middle of Run 2, Section 4 to the top of Run 2, Section 7 (10,478.8 – 10,487.0). It is cut by numerous thin calcite veins (< 1 mm) and calcite-bearing mesoscale faults. The base of the unit grades into a cataclastic siltstone and shale that is similar to the unit above this section.

A dark, grayish-black foliated fault gouge extends from the top of Run 2, Section 7 to the bottom of the core sample (10,487 – 10,492.3 ft). The contact between this unit and the unit above is sharply inclined. The fault rock is intensely sheared and contains a very fine particle, incohesive matrix that displays a wavy foliation. Microscale shears create a penetrative, micro-scaly fabric with striated split surfaces. Some visible porphyroclasts comprised of serpentinite, very fine-grained sandstone, and siltstone exist within the matrix. Fragments of white veins, thought to be calcite, are present. The foliations are approximately perpendicular to the core axis and the porphyroclasts are elongated parallel to the foliation.

The foliated fault gouge contains a block of serpentinite that is approximately 30 cm thick. The serpentinite is fractured by white veins (up to several mm thick) that are subparallel and subperpendicular to the core axis. A thin, inclined zone of sheared serpentinite marks the upper contact between the serpentinite block and the gouge. The lower boundary is also sheared serpentinite (4 cm thick) with fragmented and reoriented veins.

Core Run 3 contained a sheared siltstone and shale at measured depths of 10,493.5 ft to the end of the core at 10,497.2 ft. This unit was a thinly bedded, dark grayish-black shale, a grayish-black to olive siltstone, and a very fine-grained sandstone. Bedding was normal to the core axis and was offset along mesoscale faults and sheared shale. Coarse-grained layers and lenses were cut by thin calcite veins oriented at a high angle to the layers. The top and bottom of the unit contained cataclasite shale.

From these initial descriptions, a foundation of ideas for the concepts within the ontology began to take place. This foundation was augmented through the scrutiny of the investigator objectives. As previously mentioned, these could be found on the Earthscope National meeting website [http://www.iris.edu/hq/esreg/page/abstracts].

The principal objectives of the investigators can be summarized as follows: the study of fault surface roughness, the physical and chemical mechanics of the San Andreas Fault Parkfield segment, the study of pore space microstructure (i.e., pore fluids and chemicals, pore networks, and pore relationship to fabric and mineralogy) as it relates to fault zone mechanics, the chemical, cyclic and mechanical deformation in gouge, the interseismic period fault permeability and strength evolution, and the study of episodes of fluid flow within veins (particularly in relationship to the phenomena of pressure). These objectives were aimed toward a better understanding of the earthquake cycle, particularly at the microstructural level. The primary application of the ontology was for the domain of structural geology as it pertains to brittle fracture microstructures within an earthquake zone.

The objectives of individual investigators and their teams were taken into consideration. For example, Dr. Desbois and Dr. Urai of Aachen University in Germany obtained samples from Hole E, Run 1, Section 2: 37-39 cm and Hole G, Run 4, Section 2: 8-10 cm. Their poster, presented at the Earthscope meeting in Idaho, included information regarding their team's research methodology, initial study results, and objectives for current and future study.

Initial research results presented by the team of Dr. Desbois' [Desbois and Urai, 2009] was based on their unique application of argon-beam cross sectioning (Broad Ion Beam, BIB) to prepare well polished thin section surfaces used in conjunction with Scanning Electron Microscopy (SEM). The primary objective was to image the pore spaces at high resolution. This BIB-SEM approach allowed for the investigation of the pores and in-situ fluids, as well as reconstruction of the pore network in 3D (using serial cross sectioning). This methodology resulted in high quality images at the pore scale, along with detailed structural information (i.e., the distribution and morphologies of pores in relation to fabric and mineralogy) (Figure 6.3). Micro-chemical analysis was also performed using Energy Dispersive X-ray analysis (EDX) coupled with SEM. The overall objective of the team is a better understanding of the role of porosity in the mechanical behavior of fault zone mechanics. Further research plans include the use of cryogenic techniques to quench in-situ pore fluids, along with serial cross-sectioning. Future plans include the use of EDX techniques to obtain micro-chemical analysis. The purpose will be the study of the distribution of fluid occupying the pore space and the connectivity of the pore network [Desbois and Urai, 2009].

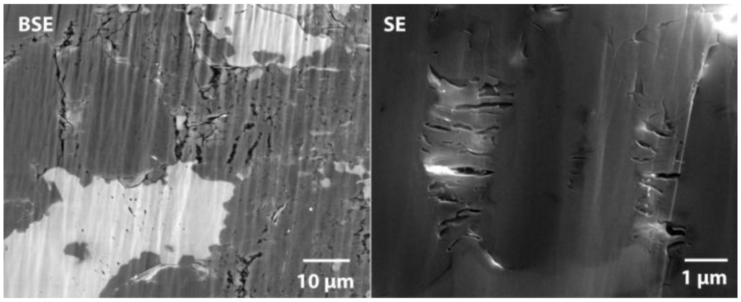


Figure 6.3. Shows vertical interconnected pathways cemented by clay material. Minerals in contact with these have rounded edges. Clay filling channels exhibits parallel and elongated pores [Desbois and Urai, 2009].

Dr. Hadizadeh’s team at the University of Louisville obtained core section samples from Hole G, Run 2, Section 4. The core sample from sidehole G consisted of foliated shale-siltstone cataclasites in contact with (lesser deformed) shale [Hadizadeh et al., 2009]. The team performed microstructural analysis of the foliated fault gouge using data acquired from high resolution SEM, Cathodoluminescence images, X-ray fluorescence (XRF) maps and EDX spectra. This analysis was used to correlate microstructural evidence (shear localization) with mechanical properties of the gouge. The preliminary studies of the group found that data indicated multiple episodes of veining and deformation (Figure 6.4). The team also found evidence of syn-deformation growth and brittle shear of pyrite, alteration of some host shale clays to Fe-rich smectite (stilpnomelane), and grain boundary corrosion of non-clay fragments. SEM image analysis indicated that gouge porosity varied between 0 and 18% with the highest porosity in clay gouge bands with siltstone fragments. EDX results also indicated that the high porosity areas were deficient in Ca. The gouge was banded into areas of low clay content with high porosity and high clay content with low porosity. The team plans to investigate the effects of this banding foliation on the gouge as it relates to mechanical behavior, fluid flow, and the role of pressure solution creep [Hadizadeh et al., 2009].

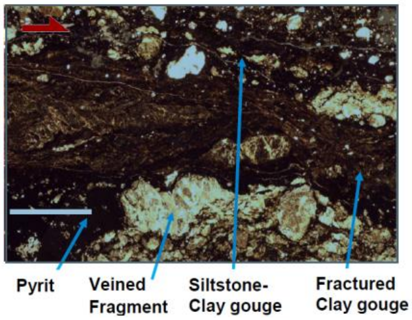


Figure 6.4. Microstructures of foliated clay gouge indicate multiple deformation episodes [Hadizadeh, et al., 2009].

Dr. Gratier's team of the University of Grenoble obtained six samples for study. These were from Hole E, Run 1 (at measured depths 10,307 to 10,323ft) and Hole G, Run 2 (at measured depth 10,814ft). This team posed some specific questions in relation to the seismic cycle [Gratier and Doan, 2009]: Does fracturing increase pressure solution creep in fault zones? Are fractures related to earthquakes? What is the mechanism of such fracturing? Which minerals are dissolved or passively concentrated by pressure solution creep? Which minerals seal fractures? What are the various mechanisms of healing and sealing? What are the mean sizes of the closed system for mass transfer?

Initial study results by Dr. Gratier's team indicated fan shaped fractures networks at the contact between grains, and the fragmentation of rigid grains (Figure 6.5a). Orientation of the fracture network appeared compatible with horizontal maximal and minimal stresses. Quartz minerals were not dissolved, but calcite and feldspar minerals were dissolved (Figure 6.5b). Phyllosilicates, Ti and Fe oxides, and sulfurs were passively concentrated along well-defined interfaces. All of these features were thought to attest to the pressure solution creep process. Additionally, various characteristics of times of healing were noted by sealing of fractures either by euhedral crystals growing in free fluid or by calcite fibers parallel to extension. Solution cleavage in sample 10, located not far from a creeping zone, was found to be so intense that it produced tectonic layering with large strain values (more than 20% of extension, see Fig. 6.5c) [Gratier and Doan, 2009].

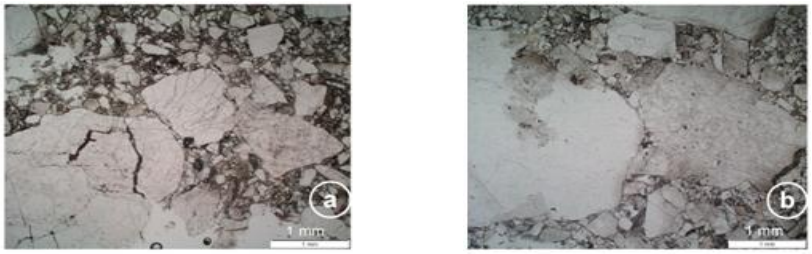


Figure 6.5 (a) Fracture network initiating at grain contact, (b) Dissolution of feldspar [Gratier and Doan, 2009].

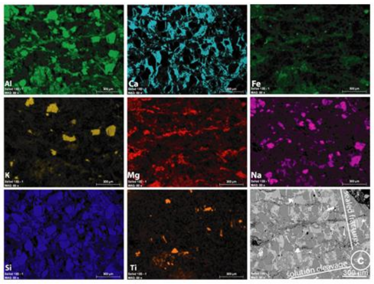


Figure 6.5 c. SEM chemical distribution maps: dissolution of feldspars, deposition of calcite and passive concentration of phyllosilicates [Gratier and Doan,2009].

In conclusion of the team's initial findings, Gratier and Doan [2009] stated that microstructures show healing mechanisms of both the fault gouge and the damage zone were related to mechano-chemical processes, such as pressure solution creep, that can explain post-seismic mechanisms of creep and sealing. Further study plans were indicated to complete the data on the parameters deduced (mechanism of sealing, diffusive path transfer geometry, PT stress conditions, nature of minerals and fluids) in order to evaluate the characteristics of fault healing in a given geological context.

Dr. Di Toro's team of investigators at the University of Padova in Italy studied a SAFOD core sample from the measured depth of 10,494 ft. This sample location contained the interface between sheared black shales and veined ribbons of arkosic sandstone (Figure 6.6). The team posed the question regarding the possibility that local patches of fluid overpressure within the fault zone might be an explanation for the "weak" behavior of the San Andreas Fault. This overpressure is thought to have developed from the localized zone of sheared shales and siltstones that act as a barrier to fluid circulation. This zone is the location of the currently creeping segment of the fault.

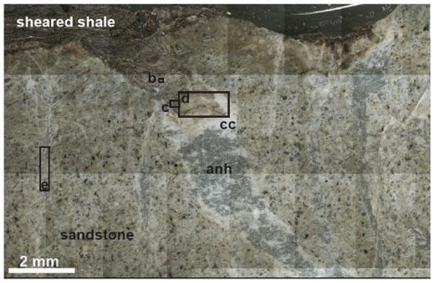


Figure 6.6. Microstructures from a sample slightly NE of the fault creep zone [Mittempergher, DiToro, Spiess, 2009].

X-Ray diffraction analysis indicated that the shales were made of 30% illite and mixed layer illite-smectite, plus 23% K-feldspar, 22% quartz, 20% plagioclase, 5% chlorite, 1% calcite, with pyrite and traces of anhydrite. Transmitted and reflected optical and scanning electron microscope investigations showed that the shales were pervasively foliated (S-C' foliation). Cross-cutting relationships indicated that the veins in sandstone underwent multiple opening and sealing episodes. Within the first veining episode were tension gashes in en-echelon array, filled by elongated anhydrite crystals, and sheared veins with anhydrite slickenfibres. The anhydrite veins showed a stress field orientation compatible with the S-C' foliation in the sheared shale. The second veining episode contained disarticulated anhydrite fibers in calcite veins, partly intruded by the shale layer. The calcite had a blocky texture (absence of shape preferred orientation, SPO) at the vein margins, and elongated habits towards the vein center (presence of SPO).

The team also used Electron Back Scattered Diffraction, which revealed a lack of crystallographic preferred orientation (CPO) in the calcite. Cathodoluminescence analysis suggested that growth of calcite in the vein center occurred by distributed fracturing and sealing. The team concluded by stating that the investigated sample from the San Andreas fault zone suggests multiple episodes of fluid flow and the occurrence of transient high fluid pressures alternated with dominant sub-lithostatic fluid pressures during the seismic cycle. Fracture sealing in the sub-lithostatic regime is thought to induce build up of fluid overpressure in isolated porous sandstone ribbons embedded in less permeable shales [Mittempergher et al., 2009].

Core sample #66 from Hole G, Run 4, Section 3 (interval 37 – 39 cm) was approved for research by another team of investigators on the SAFOD project. The study of fault surface roughness as it relates to the SAF earthquake cycle was the primary objective proposed by this team. The team's first order parameter was to establish the detailed topography of the fault surface in order to estimate the heterogeneity of the stress field in the fault.

Initial analysis on this sample determined that this section is a foliated fault gouge with clasts of serpentine and a striated surface. The team hypothesized that these striations are the result of aseismic creep and proposed to use White Light Interferometry to perform high resolution topography measurements of the roughness of the slip surface. Chemical analysis was proposed via the use of X-ray Fluorescence (XRF) chemical mapping. The chemical mapping would also be employed to identify mass transfer processes. Application of XRD would be utilized to characterize mineral assemblage of the sample. In addition, the team proposed to study the textures of the clasts of serpentine in order to compare them with textures and mineralogy found in previous outcrop studies conducted by other scientists.

Defining the Classes

The described objectives from each investigative team were utilized in order to establish a list of class concepts. Once classes were defined, their type-subtype hierarchy was determined.

This taxonomic process involved determination of whether or not the class or object may be an instance of another class. If a class is a superclass of a subclass, then every instance of the subclass is also an instance of the superclass. For example, if PressureSolutionSeam is a subclass of Fracture, then every instance of PressureSolutionSeam (e.g., styolite, splay) is also a Fracture.

This stage of ontological development employs the theory of mereology. Mereology is the study of the relationship of parts to a whole. It is an integral facet of ontology philosophy and, like ontology, dates back to early Greek philosophy. This aspect of ontology development is dependent on the world view of the developer. For example, would ‘texture’ and ‘fabric’ be separate classes, or would fabric be a superclass of texture? While fabric can be used to describe elements (i.e., grains, crystals) within a rock, such as their preferred orientation, texture can be used to describe the elements characteristics (i.e., shape, size). If a sample contains elongate anhydrite fibers (textural shape of element), oriented toward the vein center (exhibiting an SPO-type fabric of element), the SPO could be considered to be a sub-characteristic of the elongated fibers. Because the concept of fabric is typically a deformational feature, and often related to a planar structure, it would more appropriately be a separate class on its own.

The first approach in determining class concepts from the described samples and objectives was to seek out the nouns in the descriptions. The nouns served to reveal basic class concept descriptions for the domain. For example, the investigators of Dr. Gratier’s team presented information about the core section sample they would be studying. Afterwards they presented initial study results and concluded with a description of further study. The nouns used in Dr. Gratier’s team objectives were considered as potential class concepts. For example, the team’s study is based on Hole E, Run 1. This information would be stored in a class called Sample (with attributes for hole, run, and interval). The team studied a thin section and that information would be contained in a class called ThinSection (also related to Sample, as described later in this report section). The ThinSection class included information about the type of analysis used on it, whether or not it was polished (or had a coating material), its orientation, etc… The team could also store information about the fracture networks in their sample, along with descriptions of the grains and their features. This was made available in the ontology through classes such as Fracture (with subclass for PressureSolutionSeam, and attributes for orientation, geometry, healing mechanism, etc.), Microstructure (grains, structural features such as lineations and/or foliations, etc.), Texture (grain size, shape, contact network, etc.), and Mineral (type, concentrated or dissolved, etc.). Class concept establishment was done in this manner for each investigator.

Subclasses were then defined for some of these classes. For example, the class Fracture contained subclasses PressureSolutionSeam and Vein. Other classes, such as FaultRock, had subclasses for BrittleFaultRock and DuctileFaultRock. These two subclasses were developed in the ontology as disjoint from each other, meaning they would not share mutual instances. In addition, these subclasses contained subclasses of their own. For example, the class BrittleFaultRock had subclasses defined for Cataclasite, Breccia, and FaultGouge. The subclass DuctileFaultRock was further specialized with subclasses for Mylonite, ProtoMylonite, and UltraMylonite.

The class concept for Microstructure was appointed to the superclass Structure. The purpose of this design was so that other structural subclasses could be included, such as

LinearStructure and PlanarStructure. These sibling classes were restricted to disjoint because an instance may not simultaneously be both linear and planar. These subclasses were further defined with more subclasses (i.e., PlanarStructure had subclasses for Banding and Foliation).

A class concept for Analysis was also defined. Analysis included subclasses to accommodate the description of various types of analysis the investigators would be using. Some examples of these subclasses include Broad Ion Beam (BIB), Rotary Torsion Experiment, Cathodoluminescence, SEM (with subclasses for CryogenicSEM and EBSD), which are types of Analysis.

Once the general classes and subclasses were defined, they were grouped into four major subclasses for further organization. These included FaultComponent, GeologicProperty, GeologicStructure, and InvestigationalComponent. GeologicStructure, for example, comprised of classes with features pertinent to (upper) crustal structures in fault zones (ie: veins, pressure solution seams) and related concepts (e.g. lineation, foliation). A final list of all class concepts is provided in the ontology development section below.

Closely intertwined with the establishment of classes was determining which properties would establish relationships between those classes or other data values. The purpose of the properties was to describe the internal structure of the class concepts [Noy and McGuinness, 2001]. Delineating what concept or feature is to be a class or a property is a challenge that can be addressed by determining which classes have single arguments. According to Noy and McGuinness [2001], classes can be viewed as unary predicates, or questions that have one argument. For example, “Is this object a fault rock?” In contrast, properties can answer more than one argument. For example, “What is the texture of this fault rock?” (The rock has several attributes (properties) that describe texture, such as grain size, grain shape, etc …). The subclasses of a class inherit the properties of that class. For example, all the properties of the superclass FaultRock (e.g., has_mineral, has_structure) will be inherited by the subclass Cataclasite. In this example, the investigator may or may not include properties about a cataclasite's minerals and/or structure. (Note that when an investigator inputs data, they are not required to include information for every available property option.)

There are two predominant types of properties available for use within the Protégé-OWL environment. One is an object property, and the other a datatype property. An object property links sets of individuals together, such as individuals from the class Microstructure to the class Foliation. For example, the object property 'has_foliation' was used in the ontology to establish a relationship between the classes Microstructure and Foliation (Figure 6.7). (All properties were named using lowercase letters.)

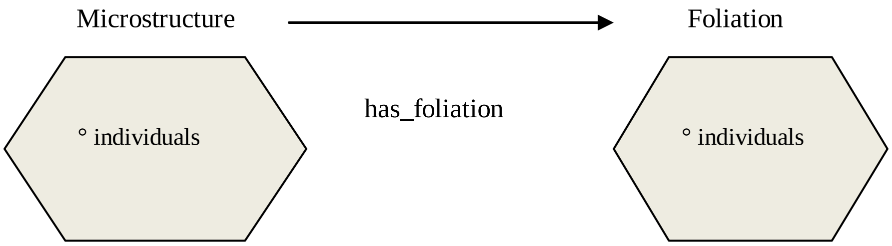


Figure 6.7. Individuals of the class Microstructure are related to the individuals of the class Foliation through the object property has_foliation.

The datatype property links an individual to a datatype value (XML Schema) or an rdf literal. Datatype properties relate instances to values rather than resources [Lacy, 2005]. (These property types were discussed in detail in chapter 5.) An example of a datatype property used in the ontology established a relationship for Analysis to analysis_id. The attribute (property) for analysis_id allows for a numeric value (and was restricted to a unique number, described later).

Object properties for the ontology were grouped into four main areas, based on the primary class concept groups. These were has_geologic_structure, has_geologic_property, has_fault_component, and has_investigational_component. Subproperties were then developed for each group, such as has_brittle_fault_rock under the has_geologic_component. Further subproperties specialized has_brittle_fault_rock, such as has_cataclasite and has_faut_gouge. If has_cataclasite is a subproperty of has_brittle_fault_rock, then every instance of a domain class which uses the has_cataclasite must also use the has_brittle_fault_rock property.

Ontology Development

Once the list of classes and subclasses were determined, Stanford University's Protégé tool [http://protege.stanford.edu/overview/protege-owl.html] was used to develop the ontology. The Protégé platform provided for the development of the ontology through a Protégé-OWL modeling editor. This editor provided the means to define the classes, establish their properties (along with corresponding characteristics), visualize the class structure, check consistency and perform reasoning on the classes, and edit all elements within the ontology. Protégé's Java plug-in architecture also allowed for the development of an application interface. These features will be discussed further in this section of the thesis.

The Protégé software tools were downloaded from Stanford's Protégé website at: http://protege.stanford.edu/download/protege/3.4/installanywhere. From the installation website, version 3.4 of Protégé for Windows was selected. (This was the most current version available when the ontology was initially developed.) The link for this selection provided a popup that asked to run the executable file 'install_protege_3.4.exe'. The executable installation application placed the files for the Protégé application in the folder for 'Documents and Settings.' The application could then be run by selecting the Protégé application program.

Execution of the Protégé application icon elicited a "Welcome to Protégé" screen (Figure 6.8). Next, "New Project" was selected and a "Create New Project" pop-up screen allowed for the choice of "OWL / RDF Files" to develop the web-standardized ontology. This step generated two files, the project file (.pprj - stores information relating to application interface) and the source file (.owl - contains the classes, individuals, and properties).

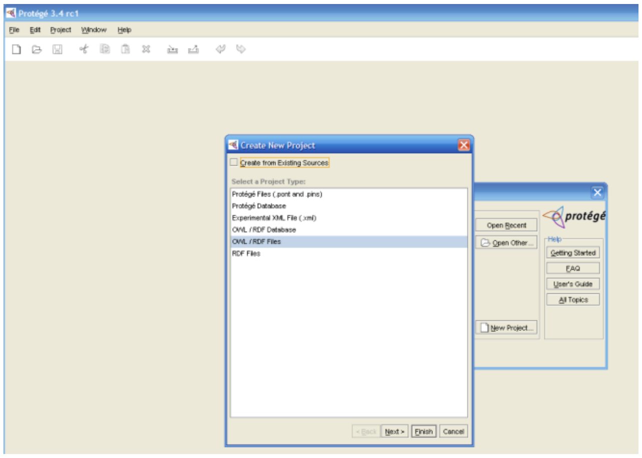
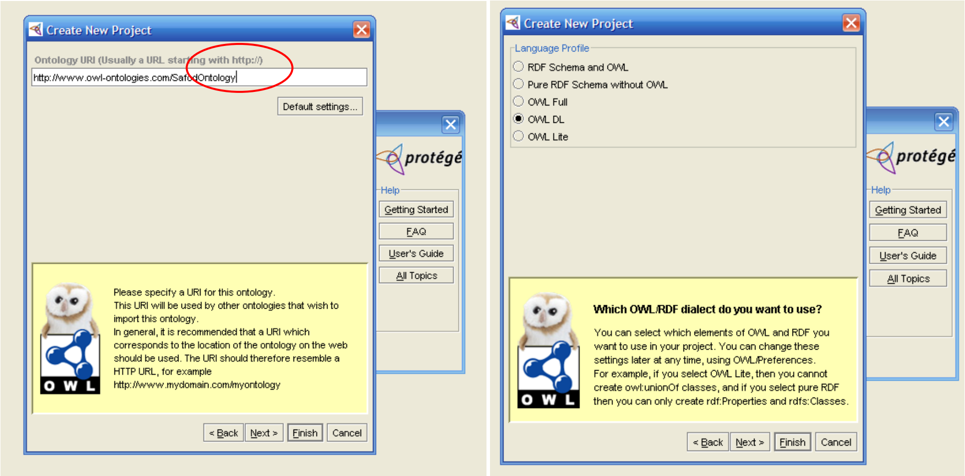


Figure 6.8. Protégé startup screen and selection.

The (Figure 6.9) shows the screens used to name the URI for the ontology and the selections available for the OWL language versions. On this screen the selection for OWL DL was made. As previously explained in chapter 5, the URI is the identifier for the ontology. It also contains the location where the ontology is stored.



Ontology URI (Usually a URL starting with http://)
http://www.owl-ontologies.com/SafodOntology

Please specify a URI for this ontology.
This URI will be used by other ontologies that wish to import this ontology.
In general, it is recommended that a URI which corresponds to the location of the ontology on the web should be used. The URI should therefore resemble a HTTP URL, for example
http://www.mydomain.com/myontology

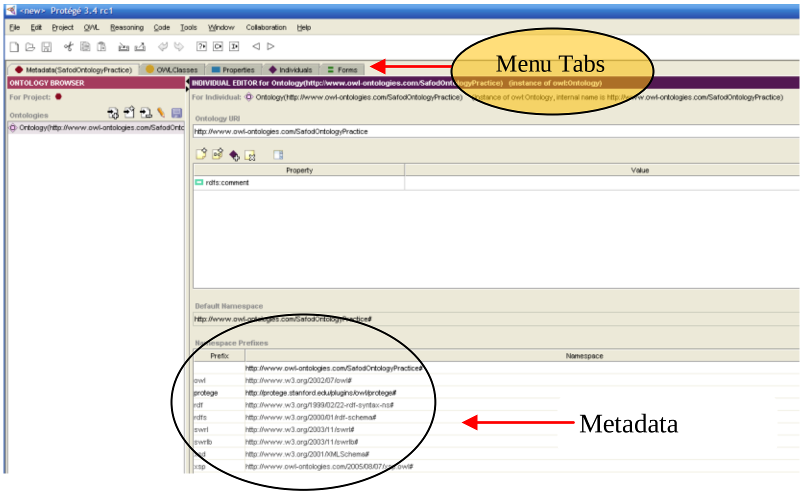
Language Profile
○ RDF Schema and OWL
○ Pure RDF Schema without OWL
○ OWL Full
● OWL DL
○ OWL Lite

Which OWL/RDF dialect do you want to use?

You can select which elements of OWL and RDF you want to use in your project. You can change these settings later at any time, using OWL/Preferences.
For example, if you select OWL Lite, then you cannot create owl:unionOf classes, and if you select pure RDF then you can only create rdf:Properties and rdfs:Classes.

Figure 6.9. Protégé assignment of URI and OWL-DL.

The initial screen for ontology development includes metadata information within the standardized URIs for the ontology. It also displays the menu tabs and tools that were used for creation of the SAFOD ontology (Figure 6.10). Protégé automatically assigns owl:Thing as the root class to represent all individuals. Any class created within Protégé is a subclass of owl:Thing. From this screen, the Classes tab was selected and the button for 'Create Subclass' was clicked to begin the process of creating classes (Figure 6.11).



INDIVIDUAL EDITOR for Ontology(http://www.owl-ontologies.com/SafodOntologyPractice) (instance of owl:Ontology)

For Individual: Ontology(http://www.owl-ontologies.com/SafodOntologyPractice) (instance of owl:Ontology, internal name is http://www.owl-ontologies.com/SafodOntologyPractice)

Ontology URI: http://www.owl-ontologies.com/SafodOntologyPractice

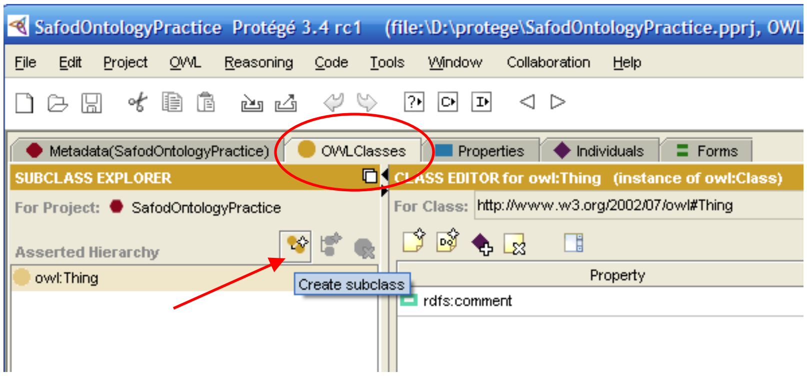
rdfs:comment

Default Namespace: http://www.owl-ontologies.com/SafodOntologyPractice#

Namespace Prefixes

Prefix	Namespace
	http://www.owl-ontologies.com/SafodOntologyPractice#
owl	http://www.w3.org/2002/07/owl#
protege	http://protege.stanford.edu/plugins/owl/protege#
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs	http://www.w3.org/2000/01/rdf-schema#
swrl	http://www.w3.org/2003/11/swrl#
swrlb	http://www.w3.org/2003/11/swrlb#
xsd	http://www.w3.org/2001/XMLSchema#
xsp	http://www.owl-ontologies.com/2005/08/07/xsp.owl#

Figure 6.10. Protégé screen shot for initial ontology development.

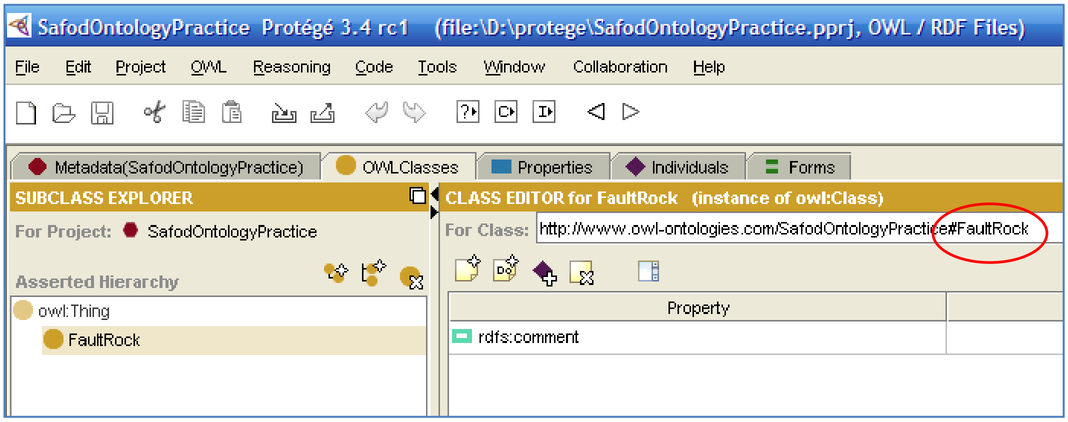


For Project: SafodOntologyPractice

For Class: http://www.w3.org/2002/07/owl#Thing

Figure 6.11. Protégé screen shot for creation of classes.

The desired class name was typed at the URI location. The first letter of a Class is always capitalized. If the name of the class contains more than one word, the words are joined with each word beginning with a capital letter. For example, the class FaultRock was input as the class name in the following screen shot (Figure 6.12).

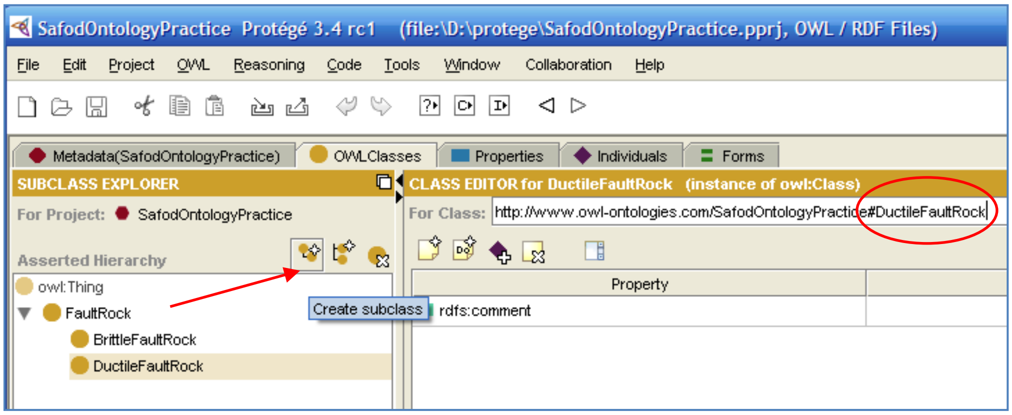


For Project: SafodOntologyPractice

For Class: http://www.owl-ontologies.com/SafodOntologyPractice#FaultRock

Figure 6.12. Protégé screen shot for naming of a
class with more than one word.

Further subclasses of this class were created by clicking on FaultRock and then the subclass icon (Figure 6.13). Examples of subclasses of FaultRock included BrittleFaultRock and DuctileFaultRock. These subclasses were then made disjoint to each other so that an individual cannot be an object of both classes simultaneously. This was accomplished by highlighting BrittleFaultRock and then pressing the "add disjoint class" icon in the disjoint widget box. A popup was then presented that allowed the disjoint class, DuctileFaultRock, to be selected (Figure 6.14).



For Project: SafodOntologyPractice

For Class: http://www.owl-ontologies.com/SafodOntologyPractice#DuctileFaultRock

Figure 6.13. Screen shot of defining subclasses BrittleFaultRock and DuctileFaultRock to the class FaultRock.

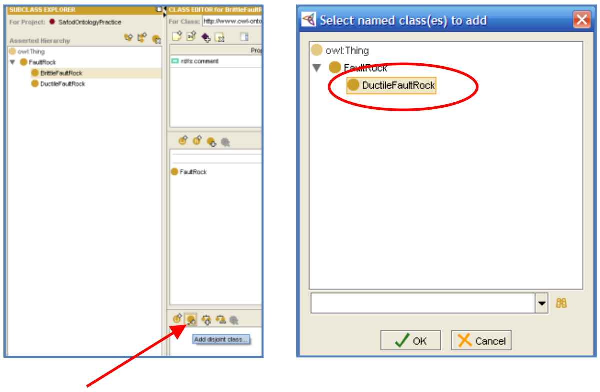
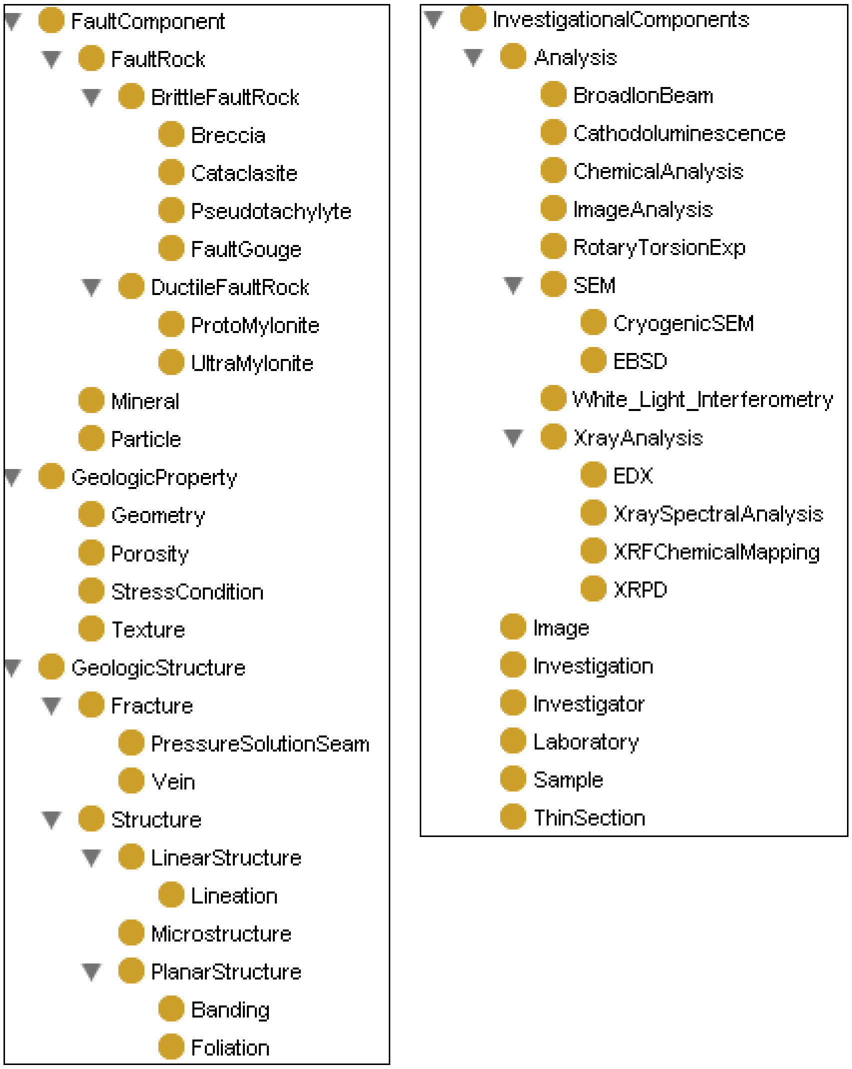


Figure 6.14. Screen shot of using the disjoint widget to make BrittleFaultRock and DuctileFaultRock disjoint from each other.

Further subclasses of these two subclasses were created in the same manner. For example, BrittleFaultRock was further specialized with the subclasses Breccia, Cataclasite, and FaultGouge. DuctileFaultRock was specialized with the subclasses Mylonite, ProtoMylonite, and UltraMylonite. The basic taxonomy of all classes defined is shown in (Figure 6.15).



- FaultComponent
  - FaultRock
    - BrittleFaultRock
      - Breccia
      - Cataclasite
      - Pseudotachylyte
      - FaultGouge
    - DuctileFaultRock
      - ProtoMylonite
      - UltraMylonite
  - Mineral
  - Particle
- GeologicProperty
  - Geometry
  - Porosity
  - StressCondition
  - Texture
- GeologicStructure
  - Fracture
    - PressureSolutionSeam
    - Vein
  - Structure
    - LinearStructure
      - Lineation
    - Microstructure
    - PlanarStructure
      - Banding
      - Foliation

- InvestigationalComponents
  - Analysis
    - BroadIonBeam
    - Cathodoluminescence
    - ChemicalAnalysis
    - ImageAnalysis
    - RotaryTorsionExp
    - SEM
      - CryogenicSEM
      - EBSD
    - White_Light_Interferometry
    - XrayAnalysis
      - EDX
      - XraySpectralAnalysis
      - XRFChemicalMapping
      - XRPD
  - Image
  - Investigation
  - Investigator
  - Laboratory
  - Sample
  - ThinSection

Figure 6.15. Screen shot of the basic taxonomy of classes for the SAFOD ontology.

Protégé development consists of classes, slots, and instances. These basically correspond to OWL's classes, properties, and individuals. While classes are a concrete representation of concepts, and properties represent the relationships, individuals are 'instances of classes' [Horridge et al., 2004]. A class is a set of individuals. For example, an individual quartz crystal or the San Andreas Fault is instance of the Mineral or Fault class, respectively. These Instances are the real objects within the domain. Properties were defined using the Protégé editor, while instances were populated using the application interface that will be described later.

Object properties link individuals from the domain class to individuals of the range class. For example, the property has_foliation would link individuals belonging to the class Microstructure to individuals belonging to the Foliation class. These property types were developed by selecting the “Properties” tab, followed by the button for “Create Object Property” (Figure 6.16). Property names begin with a lowercase letter. In order to clarify the purpose of the object property, the names used in this ontology are prefixed with the word “has” (e.g., has_foliation). The property name was given by typing the name at the URI string assignment location.

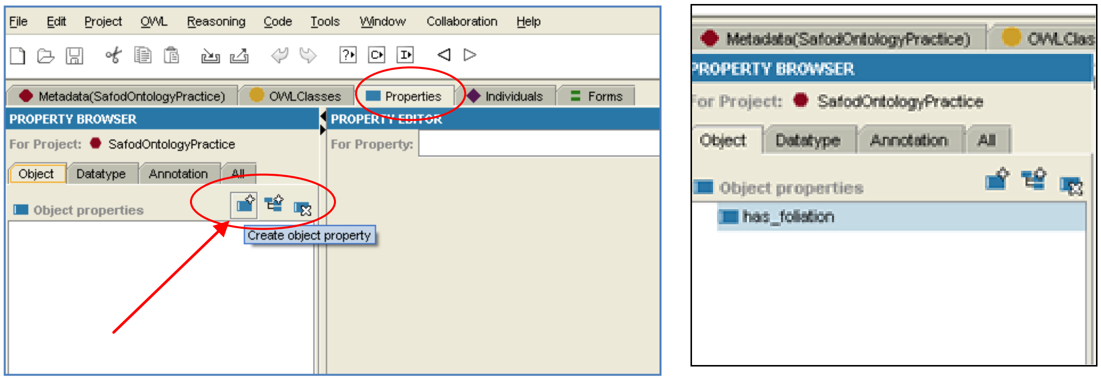
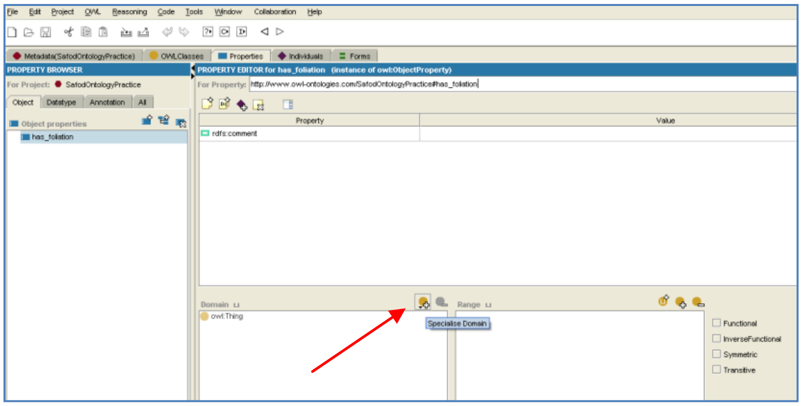


Figure 6.16. Screen shot of how to create an object property.

For the has_foliation property, the domain is Microstructure and the range is Foliation. Figure 6.17 shows the tab selected to assign Microstructure as the domain. In the domain widget box the button for “Specialize Domain” was pressed. This button elicits a popup that was used assign the class (Microstructure) for the domain (Figure 6.18).



For Project: SafodOntologyPractice

For Property: http://www.owl-ontologies.com/SafodOntologyPractice#has_foliation

Figure 6.17. Screen shot showing how to assign the domain class.

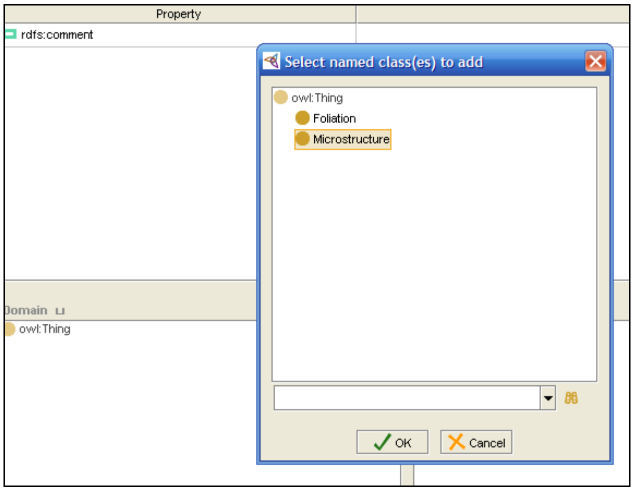


Figure 6.18. Screen shot of domain class widget.

The range should be thought of as the individuals that are members of the target class which provide value for the property. The same process for specifying the domain was used to specify the range. The ‘specialize range’ button in the range widget box was selected, followed by selection of the appropriate class (Foliation) in the popup (Figure 6.19). A final screen view is shown in (Figure 6.20).

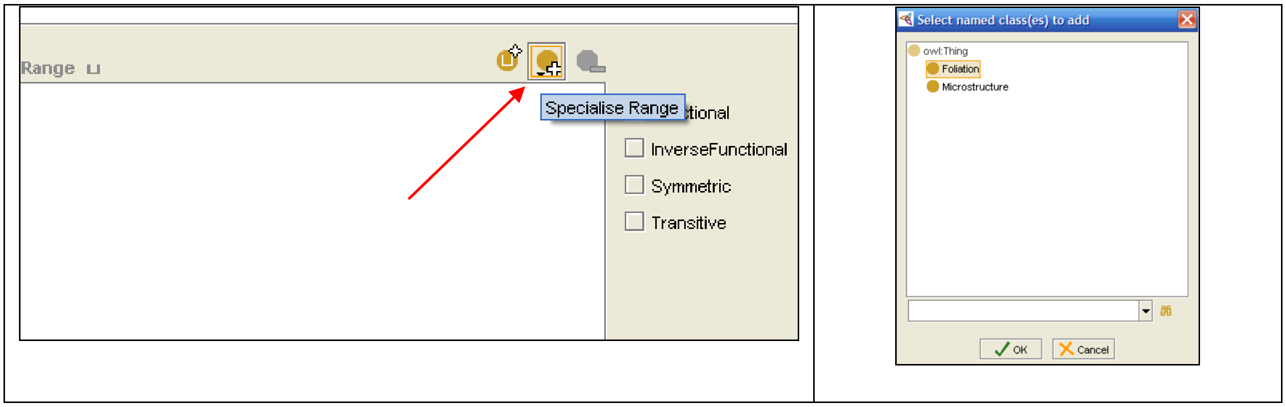


Figure 6.19. Screen shot showing how to assign the range class to the object property.

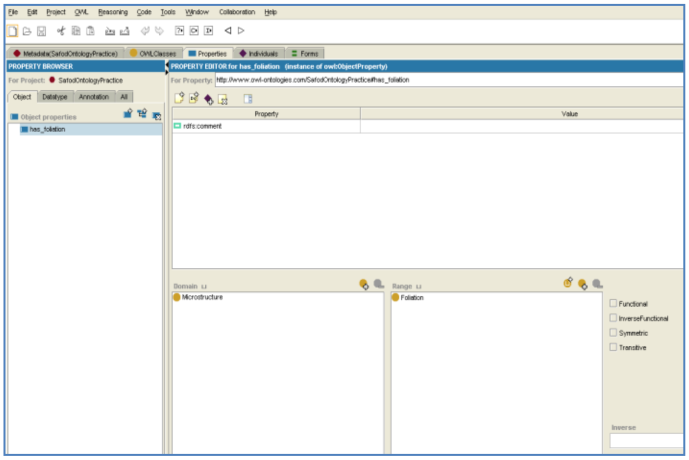


Figure 6.20. Final screen shot showing the property has_foliation and its domain and range.

This means that individuals that are used ‘on the left hand side’ of the has_foliation property will be inferred to be members of the class Microstructure. Any individuals that are used ‘on the right hand side’ of the has_foliation property will be inferred to be members of the class Foliation. We can infer from the assertion: ‘fault gouge’ has_foliation ‘compositional banding’ that ‘fault gouge’ is a member of the class Microstructure and that ‘compositional banding' is a member of the class Foliation. This will be the case even if 'fault gouge' has not been asserted to be a member of the class Microstructure and/or 'compositional banding' has not been asserted to be a member of the class Foliation.

Multiple classes may be specified as the domain (and/or the range) for a property. In this case, Protégé-OWL interprets the domain (and/or the range) of the property to be the union of the classes. OWL domains and ranges are logical axioms and should not be viewed as constraints to be checked [Horridge et al., 2004]. For example, if the property has_foliation has the domain of Microstructure (the individuals of that class) and the has_foliation property was applied to the class BroadIonBeam (individuals that are members of the class BroadIonBeam), this would not result in an error. (An error would only be generated (by a reasoner) if Microstructure was disjoint to BroadIonBeam.) There are multiple ways to check the structure of classes and properties which will be covered in the reasoner section of this chapter.

Datatype properties were created by selecting the "Datatype" tab, and then the "create datatype property" button (Figure 6.21). Again, the desired name was typed at the URI location (i.e., dip_angle). Datatype properties are associated with a domain (e.g., PlanarStructure), and the range is assigned a value type (Figure 6.22). Allowable value types are: Boolean, float, integer, string, date, and dateTime. The range for Boolean allows for a true-false type of input, such as for the datatype property is_polished (with domain specified as ThinSection). The option for float would be chosen for property types that may have large values, such as for exponential numbers. Integer was selected for numeric values, such as displacement_value. When specified as float or integer, these numbers can be used in calculations. String was used for any input that might have characters, such as description or dip_direction (input may not be azimuth, but instead as N30W, for example). An example of the range of date specified for a property in the ontology was for datatype property submitted_on. The dateTime and Time ranges were not specified within this ontology. (All properties for the ontology were created in this manner.)

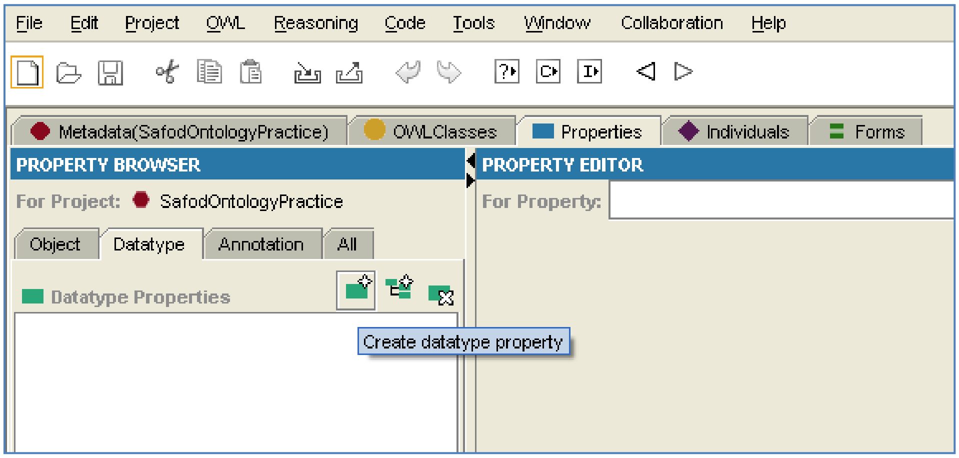


Figure 6.21. Screen shot of creating a datatype property.

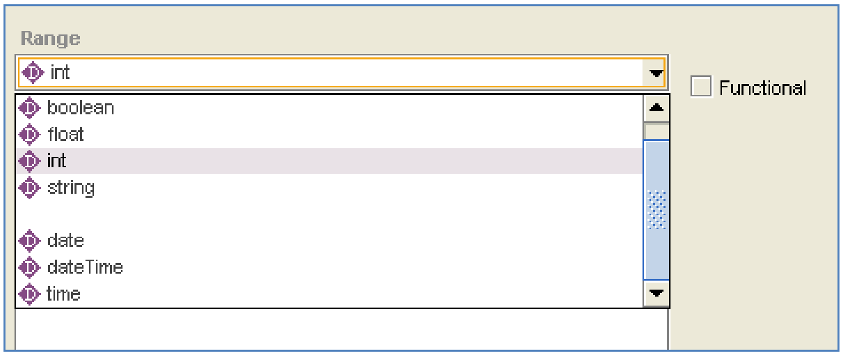


Figure 6.22. Screen shot showing a value type assigned to the range associated with the datatype property.

At this stage of ontology development a taxonomy of classes was in place with properties to link individuals or assign allowable range values. While this structure would allow for some significant ontological mechanisms, such as inheritance, additional features for specializing properties were used to further enhance capabilities. Some examples of these property characteristic features are inverse property and functional property.

The inverse property function may only be applied to object properties. If a property links individual a to individual b, then its inverse property will link individual b to individual a. For example, the properties has_investigation and has_investigator were set up as inverse. The individual for ‘investigation ID’ (within class Investigation) has_investigator ‘Dr. Gratier’ (individual within class Investigator). It can also be reasoned by inference that ‘Dr. Gratier’ has_investigation ‘investigation ID.’ (The reasoner capabilities will be discussed further later in this section.)

In general, the domain for a property is the range for its inverse, and the range for a property is the domain for its inverse [Horridge et al., 2004]. The domain specified for has_investigation is the class Investigator and the range is Investigation. The domain specified for has_investigator is Investigation and the range is Investigator. These inverses are similar to mirror images in their domain and range specification.

The inverse for these properties was accomplished by first defining the classes for Investigator and Investigation. Afterwards, object properties were defined for has_investigation and has_investigator. The ‘set inverse property’ button was then selected in the inverse widget (Figure 6.23). Afterwards, a popup appeared that allowed for the selection of the inverse property. The property browser then shows the inverse properties with arrows between them (Figure 6.24).

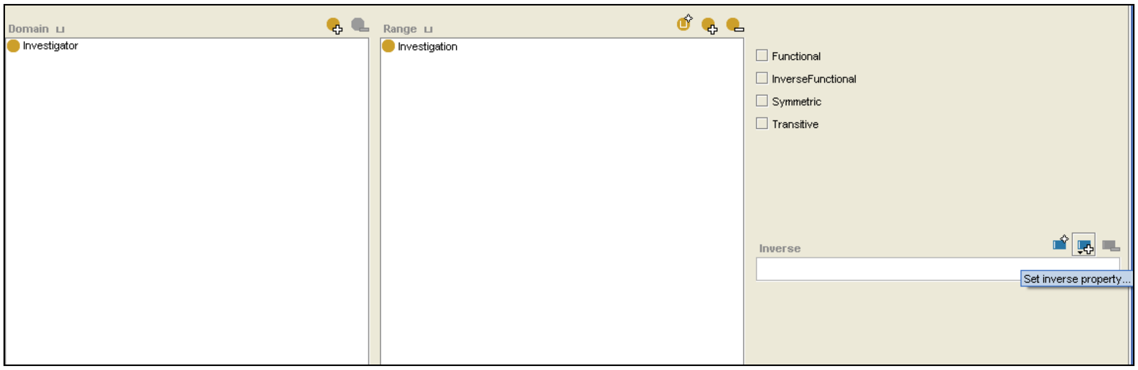


Figure 6.23. Screen shot of the inverse property widget.

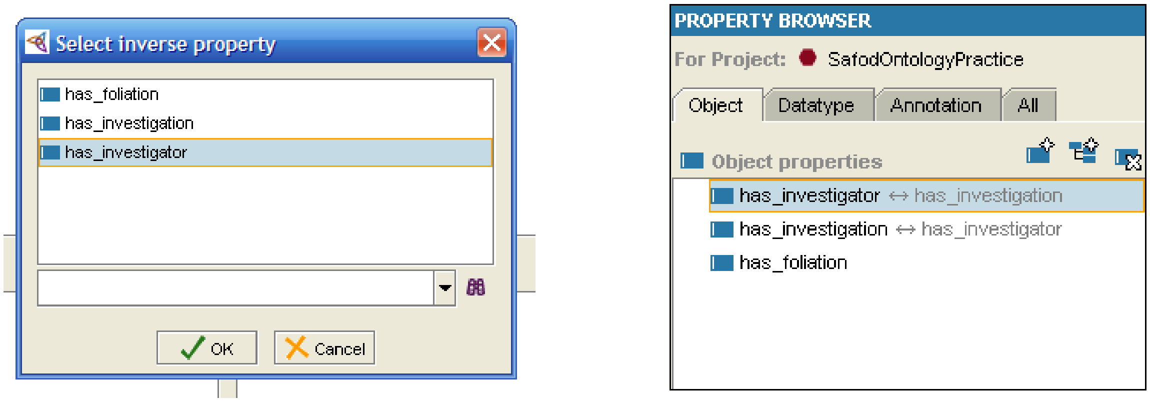


Figure 6.24. Screen shot using inverse property widget to make 'has_investigator' and 'has_investigation' inverse properties.

If a property is functional, then for a given individual there can be at most one individual that is related to the individual via the property [Horridge et al., 2004]. For example, a sample may only have one ID #. Functional properties are also known as single valued properties.

In Protégé-OWL, restrictions may be created through properties. These specifications restrict what individuals may belong to a class. These restrictions are: Quantifier Restrictions, Cardinality Restrictions, and hasValue Restrictions. Detailed descriptions of these restrictions were given in chapter 5. An example of how to apply a quantifier restriction on a property, for cases incorporated in this ontology, is described below.

The existential quantifier (symbolized with "∃"), which can be read as “some values from,” was applied to state that an Analysis will have at least one (or more) individual from the class Sample. For a set of individuals, an existential restriction specifies the existence of a (i.e. at least one) relationship along a given property to an individual that is a member of a specific class [Horridge et al., 2004]. To add this restriction, the class Analysis was highlighted. The “NECESSARY” header in the class conditions widget was then selected in order to make this a necessary condition (Figure 6.25). The “create restriction” button was then selected, which displayed the restriction dialog box (Figure 6.26).

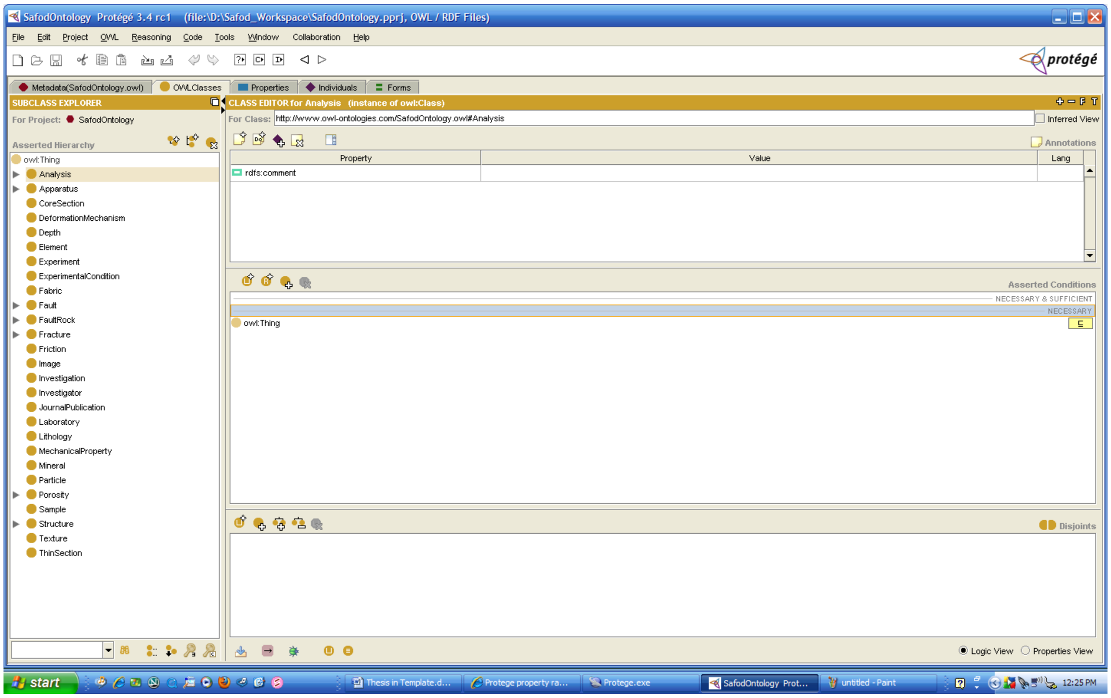
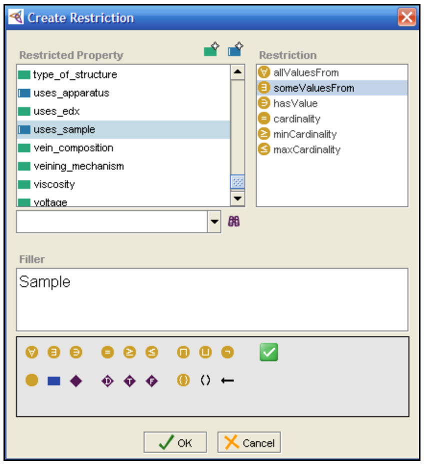


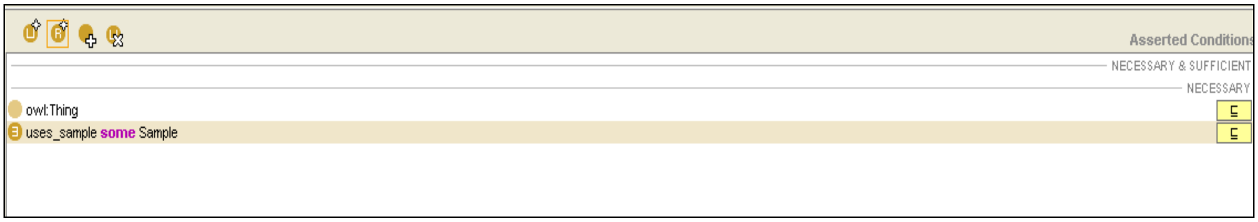
Figure 6.25. Screen shot to make a
‘Necessary’ condition on the class Analysis.



Filler: Sample

Figure 6.26. Screen shot of assigning the 'someValuesFrom' restriction to indicate any class Analysis will also have at least one sample.

From the restriction dialog box, three things must be done. First, select the type of restriction from the restriction type list (i.e., the existential restriction for someValuesFrom). Then select the property to be restricted from the property list (i.e., uses_sample). Last, specify a filler for the restriction (i.e., the class Sample). The restriction will now be displayed in the conditions widget on the Classes tab page (Figure 6.27). Notice that this restriction does not imply that all of the uses_sample relationships must be related to members of the class Sample. To restrict the relationships for a given property to individuals that are members of a specific class we must use a universal restriction.

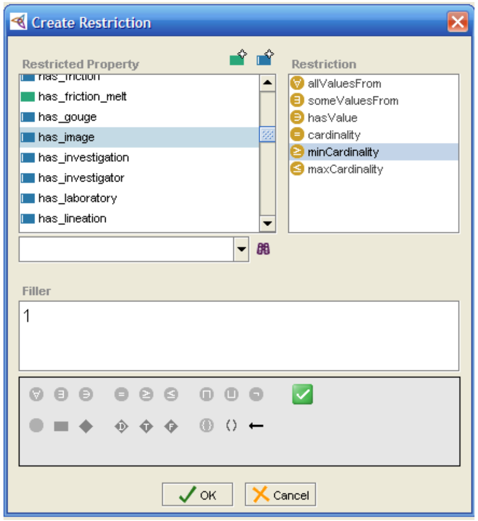


owl:Thing ⊑
uses_sample some Sample ⊑

Figure 6.27. Screen shot of the conditions widget showing a restriction on the property 'uses_sample'

In Protégé, cardinality restrictions describe the class of individuals that have at least, at most, or exactly a specified number of relationships with other individuals or datatype values [Horridge et al., 2004]. A Minimum Cardinality Restriction specifies the minimum number of relationships that an individual must participate in. A Maximum Cardinality Restriction specifies the maximum number of relationships that an individual can participate in. A Cardinality Restriction specifies the exact number of relationships that an individual must participate in. Cardinality restrictions were discussed in detail in chapter 5.

An example of a cardinality restriction used in the ontology was for the property has_image as it relates to Cathodoluminescence. This restriction defines Cathodoluminescence as a set of individuals that are members of Analysis with at least one has_image relationship. This was accomplished by highlighting the Cathodoluminescence class, clicking the Necessary conditions widget, and clicking the restriction button. In the restriction window, click the property value (has_image), then the value for minCardinality in the restriction box, and type “1” in the filler box (Figure 6.28).



Restricted Property: has_image

Restriction: minCardinality

Filler: 1

Figure 6.28. Screen shot creating restriction on the property ‘has_image’ to indicate it will have at least one associated image.

Appendix A contains a basic flowchart of all classes and their related properties. Appendix B provides a detailed depiction of each class and their respective relationships.

Reasoner

A valuable tool provided as a part of the Protégé-OWL editor package is the ability to process the ontology by a reasoner. For example, the reasoner incorporated with this ontology thesis project is Pellet. One function of the reasoner is to test whether or not a class is a subclass of another class, based on the conditions within the class, and to compute the inferred hierarchy [Horridge et al., 2004]. The reasoner checks for consistency within the classes based on whether or not it is possible for the class to have any instances. If it is not possible for a class to have any instances, then the class is inconsistent. An example of this would be a subclass that belongs to two disjoint superclasses. If the superclasses were disjoint, then there could not be any instances because no individual could simultaneously belong to both.

The consistency of the ontology was checked by selecting the “Reasoning” menu option within Protégé (Figure 6.29). This resulted in a pop-up that stating that the application was querying for inconsistent concepts (Figure 6.30). Any inconsistencies would have resulted in a large red “X” with the offending class named. Additionally, the problematic classes would be highlighted in red in the class hierarchy list.

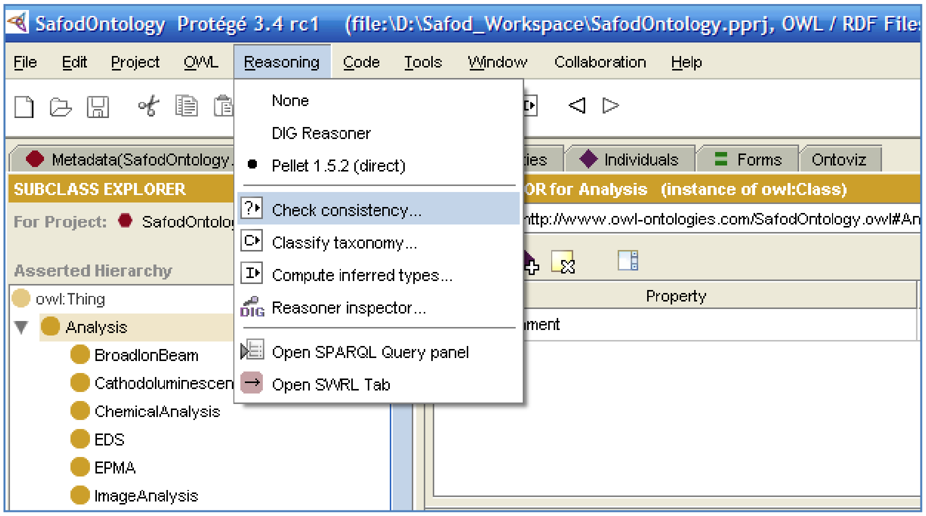
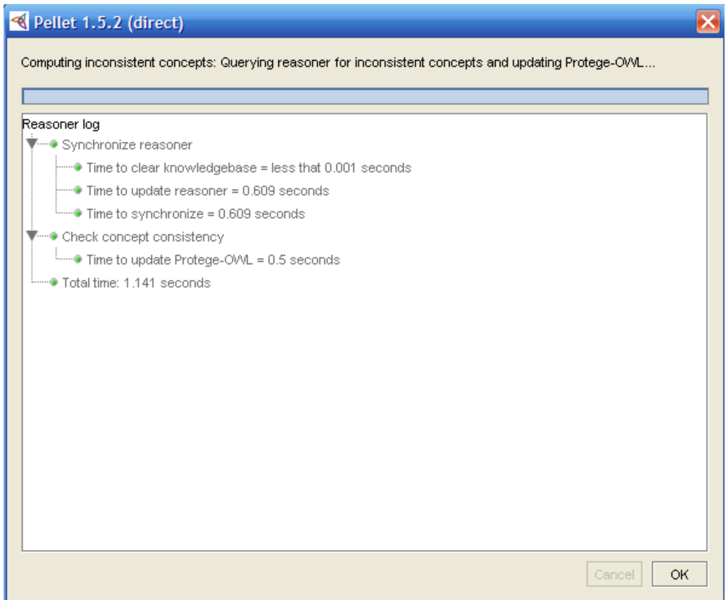


Figure 6.29. Screen shot of menu option to run the reasoner.



Computing inconsistent concepts: Querying reasoner for inconsistent concepts and updating Protege-OWL...

```
Reasoner log
Synchronize reasoner
    Time to clear knowledgebase = less that 0.001 seconds
    Time to update reasoner = 0.609 seconds
    Time to synchronize = 0.609 seconds
Check concept consistency
    Time to update Protege-OWL = 0.5 seconds
Total time: 1.141 seconds
```

Figure 6.30. Screen shot of execution of reasoner application.

Additionally, the Reasoner menu allows for an inferred hierarchy to be computed. The manually entered hierarchy is called the "asserted hierarchy." However, through this option, the application processes an inferred hierarchy. This was accomplished by selecting the "classify taxonomy" menu option. This results in an inferred hierarchy window that appears next to the existing asserted hierarchy window (Figure 6.31).

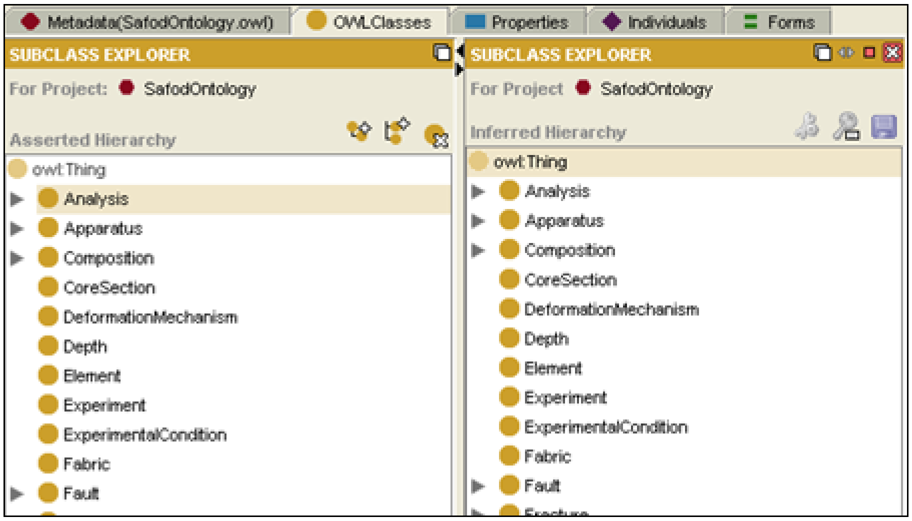
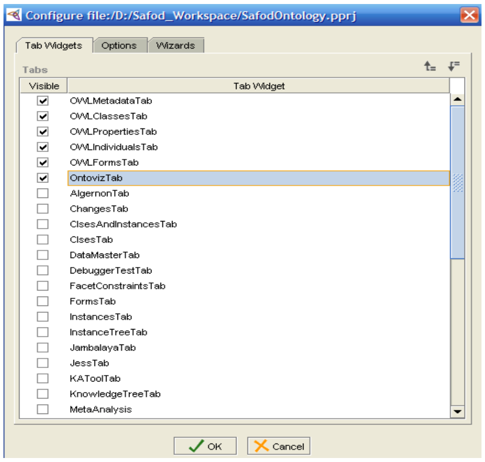


Figure 6.31. Screen shot including the inferred hierarchy window.

If a class has been reclassified (the associated superclasses have changed), then the class name will appear in a blue color in the inferred hierarchy. If a class has been found to be inconsistent its icon will be circled in red [Horridge et al., 2004]. In essence, a reasoner makes inferences (based on relationships such as inverse properties, disjoint classes, etc …) to deduce information that may not be apparent in the asserted hierarchy or relationships.

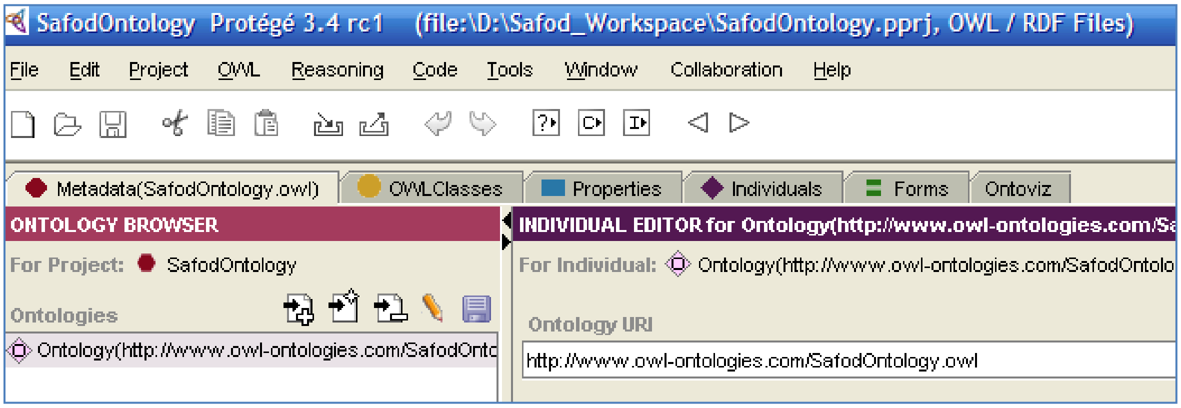
Ontoviz

The progression of ontology development was able to be graphically visualized through use of the Ontoviz option provided with the Protégé download. In order to use this feature, Graphviz (an AT&T application software product) had to be downloaded from the website [http://www.graphviz.org/Download.php]. This download installs a file, "protege.properties", that required modification. An addition was made to the properties file to point to the location of the graphviz program, as follows: “dot.command=C\:\\Program\ Files\\graphviz\\bin\\dot”. The Ontoviz tab was enabled by selecting the menu option “Project”, then “Configure”, and clicking the Ontoviz box (Figure 6.32). This added an Ontoviz tab to the main menu (Figure 6.33).



Visible	Tab Widget
☑	OWLMetadataTab
☑	OWLClassesTab
☑	OWLPropertiesTab
☑	OWLIndividualsTab
☑	OWLFormsTab
☑	OntovizTab
☐	AlgernonTab
☐	ChangesTab
☐	ClsesAndInstancesTab
☐	ClsesTab
☐	DataMasterTab
☐	DebuggerTestTab
☐	FacetConstraintsTab
☐	FormsTab
☐	InstancesTab
☐	InstanceTreeTab
☐	JambalayaTab
☐	JessTab
☐	KAToolTab
☐	KnowledgeTreeTab
☐	MetaAnalysis

Figure 6.32. Screen shot of selection of Ontoviz widget.



For Project: SafodOntology

Ontology(http://www.owl-ontologies.com/SafodOnto

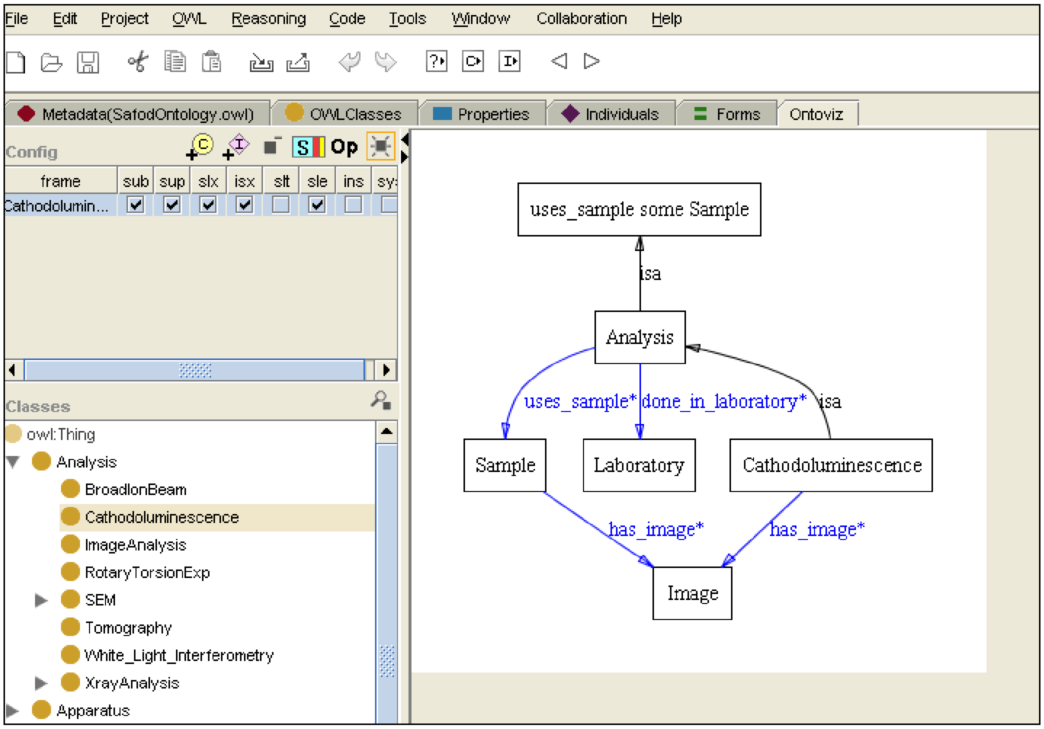
For Individual: Ontology(http://www.owl-ontologies.com/SafodOntolo

Ontology URI

http://www.owl-ontologies.com/SafodOntology.owl

Figure 6.33. Screen shot showing the Ontoviz tab.

To create a graphical overview of a class, select the Ontoviz tab. Highlight the desired class and select the “add class” button (the yellow circle with a “C”). Options such as "sub", "sup", "slx”, etc. may be checked, followed by the "Create Graph" button (Figure 6.34). The graph provides a visual of how classes are related through their respective properties.



frame	sub	sup	slx	isx	slt	sle	ins	sy:
Cathodolumin...	☑	☑	☑	☑	☐	☑	☐	☐

Figure 6.34. Screen shot showing graphing options in Ontoviz.

The available check boxes (to be graphed) and their meanings is as follows: sub - subclass closure, sup - superclass closure, slx - slot extension, isx - inverse slot extension, slt – slots, sle - slot edges, ins – instances, and sys - system frames. The same process can be used for creating graphs of instances by using the "add instance" instead of the "add class" button. To remove an entry from the Config table, use the "remove class" button.

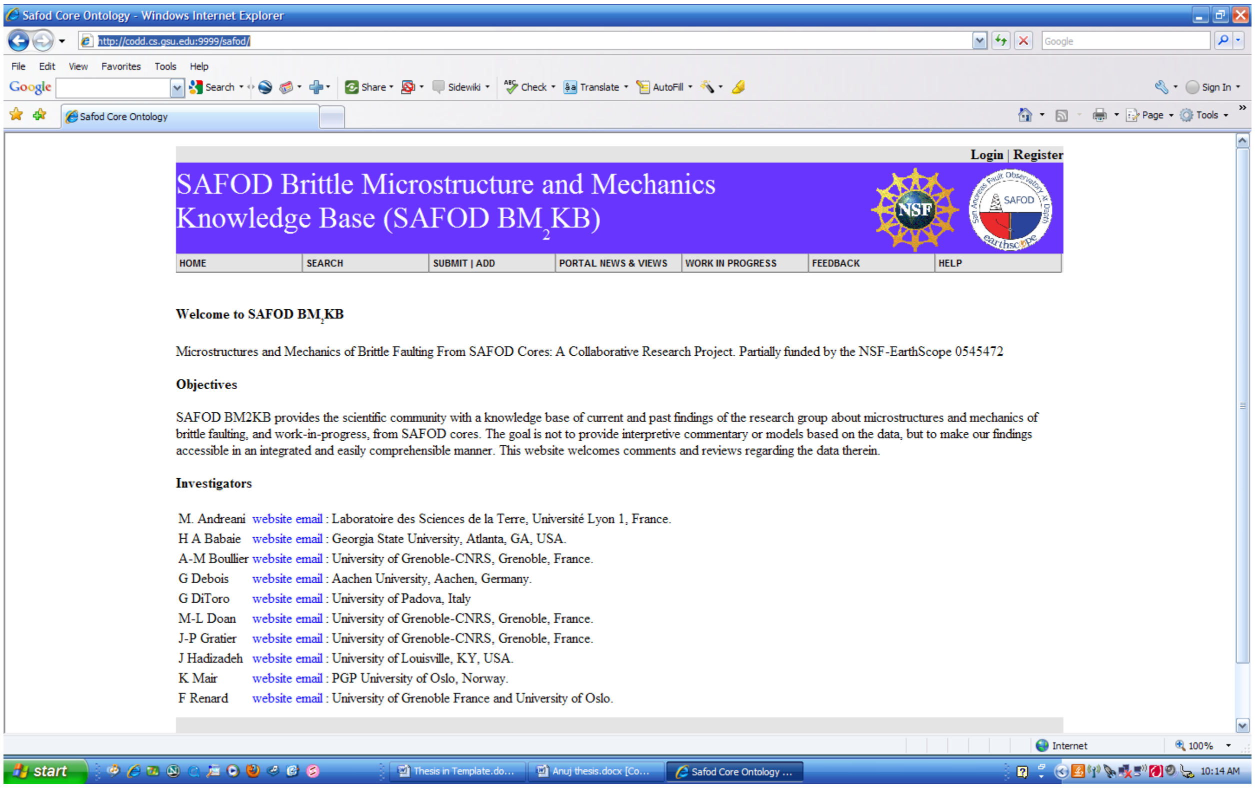
User Interface

The purpose of the application interface is to allow investigators the ability to submit, view, and query research data. Investigators also have the ability to include images and/or tables, and they may later modify or delete their own data entries. This interface was designed through a collaborative effort of Dr. Hassan Babaie, Anuj Kumar (computer science Masters student), and myself.

The application interface development was then performed by Anuj Kumar in an integrated development environment that included the Jena framework. The Jena framework supports semantic web technology (such as RDFS and OWL) and provides an application interface tool that supports inference through the Pellet reasoner.

Investigator data are submitted to me, as the administrator, for input or updates within the knowledge base (via the application interface). Data that are added through the application interface are stored on the Georgia State University server (path: /home/safod/SafodOntology) in the .owl application file (through the designated RDF URIs). Upon addition of the data to the knowledge base, the system may be queried via the search options within the application interface (described below). Query is performed using the SPARQL language which performs the query based on the data in the .owl file and the reasoners inference results.

The application interface is located on the GSU server and can be accessed through the website http://codd.cs.gsu.edu:9999/safod/. The initial login screen for an investigator to submit or search data is located at this website. Prior to submission of data, an investigator must first register with the ontology project. A ‘Register’ button is provided in the initial screen’s top right hand section (Figure 6.35).



SAFOD Brittle Microstructure and Mechanics Knowledge Base (SAFOD BM$_2$KB)

Welcome to SAFOD BM$_2$KB

Microstructures and Mechanics of Brittle Faulting From SAFOD Cores: A Collaborative Research Project. Partially funded by the NSF-EarthScope 0545472

Objectives

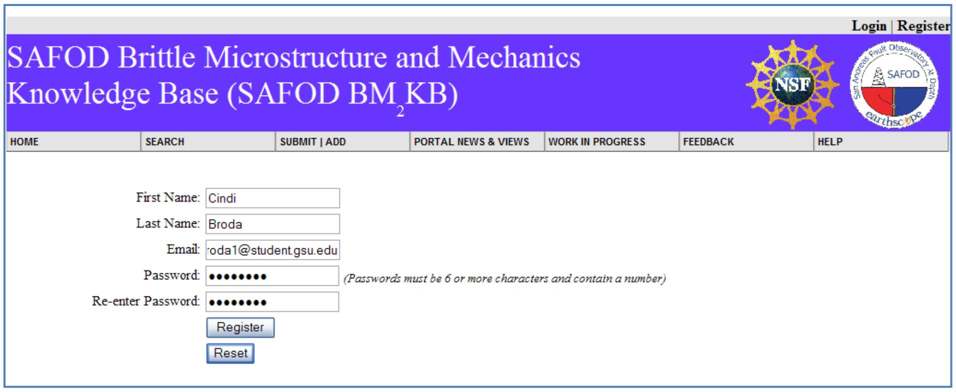
SAFOD BM2KB provides the scientific community with a knowledge base of current and past findings of the research group about microstructures and mechanics of brittle faulting, and work-in-progress, from SAFOD cores. The goal is not to provide interpretive commentary or models based on the data, but to make our findings accessible in an integrated and easily comprehensible manner. This website welcomes comments and reviews regarding the data therein.

Investigators

M. Andreani website email : Laboratoire des Sciences de la Terre, Université Lyon 1, France.
H A Babaie website email : Georgia State University, Atlanta, GA, USA.
A-M Boullier website email : University of Grenoble-CNRS, Grenoble, France.
G Debois website email : Aachen University, Aachen, Germany.
G DiToro website email : University of Padova, Italy
M-L Doan website email : University of Grenoble-CNRS, Grenoble, France.
J-P Gratier website email : University of Grenoble-CNRS, Grenoble, France.
J Hadizadeh website email : University of Louisville, KY, USA.
K Mair website email : PGP University of Oslo, Norway.
F Renard website email : University of Grenoble France and University of Oslo.

Figure 6.35. Screen shot of the intial startup screen for the SAFOD ontology. This is the location for user registration.

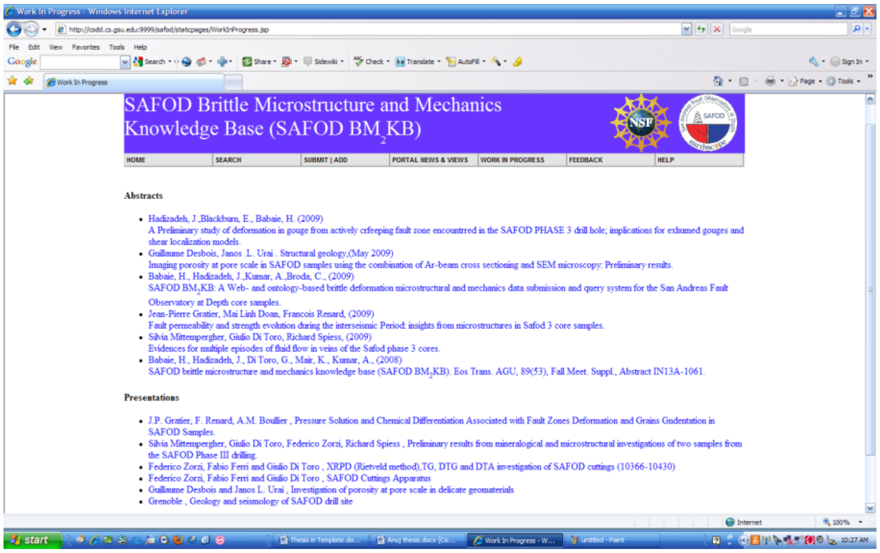
After selecting the ‘Register’ button, a screen is provided for the user to input their information (Figure 6.36). Once this information is entered, the user presses the final ‘register’ button and the information is sent to the administrator (e.g., myself). Upon receipt of the information, I enter it into the knowledge base (using the ‘submit’ → ‘add’ options). After the investigator is registered, he or she may submit and/or manipulate their research data.



First Name: Cindi
Last Name: Broda
Email: roda1@student.gsu.edu
Password: ●●●●●●●● *(Passwords must be 6 or more characters and contain a number)*
Re-enter Password: ●●●●●●●●

Figure 6.36. Screen shot of the registration process.

The initial screen provides menu options for Home, Search, Submit/Add, Portal News and Views, Work In Progress, Feedback, and Help. The menu for 'Work In Progress' provides a screen with links to the latest abstracts, publications and presentations related to the SAFOD investigators that are using this ontology project (Figure 6.37). The 'Feedback' menu option provides an area for investigators (or the community) to offer feedback or suggestions. The 'Help' menu provides an overview description of the available menu options.



Abstracts

- Hadizadeh, J.,Blackburn, E., Babaie, H. (2009)
  A Preliminary study of deformation in gouge from actively crfeeping fault zone encountred in the SAFOD PHASE 3 drill hole; implications for exlumed gouges and shear localization models.
- Guillaume Desbois, Janos .L. Urai . Structural geology,(May 2009)
  Imaging porosity at pore scale in SAFOD samples using the combination of Ar-beam cross sectioning and SEM microscopy: Preliminary results.
- Babaie, H., Hadizadeh, J.,Kumar, A.,Broda, C., (2009)
  SAFOD BM$_2$KB: A Web- and ontology-based brittle deformation microstructural and mechanics data submission and query system for the San Andreas Fault Observatory at Depth core samples.
- Jean-Pierre Gratier, Mai Linh Doan, Francois Renard, (2009)
  Fault permeability and strength evolution during the interseismic Period: insights from microstructures in Safod 3 core samples.
- Silvia Mittempergher, Giulio Di Toro, Richard Spiess, (2009)
  Evidences for multiple episodes of fluid flow in veins of the Safod phase 3 cores.
- Babaie, H., Hadizadeh, J., Di Toro, G., Mair, K., Kumar, A., (2008)
  SAFOD brittle microstructure and mechanics knowledge base (SAFOD BM$_2$KB). Eos Trans. AGU, 89(53), Fall Meet. Suppl., Abstract IN13A-1061.

Presentations

- J.P. Gratier, F. Renard, A.M. Boullier , Pressure Solution and Chemical Differentiation Associated with Fault Zones Deformation and Grains Gndentation in SAFOD Samples.
- Silvia Mittempergher, Giulio Di Toro, Federico Zorzi, Richard Spiess , Preliminary results from mineralogical and microstructural investigations of two samples from the SAFOD Phase III drilling.
- Federico Zorzi, Fabio Ferri and Giulio Di Toro , XRPD (Rietveld method),TG, DTG and DTA investigation of SAFOD cuttings (10366-10430)
- Federico Zorzi, Fabio Ferri and Giulio Di Toro , SAFOD Cuttings Apparatus
- Guillaume Desbois and Janos L. Urai , Investigation of porosity at pore scale in delicate geomaterials
- Grenoble , Geology and seismology of SAFOD drill site

Figure 6.37. Screen shot of work in progress related to the SAFOD project.

In order for an investigator to submit data, he or she must first login. The button to login is in the upper right corner of home screen (next to the 'register' button). Once the user is logged in, he would select the 'Submit/Add' button and then press 'Submit Data' (Figure 6.38).

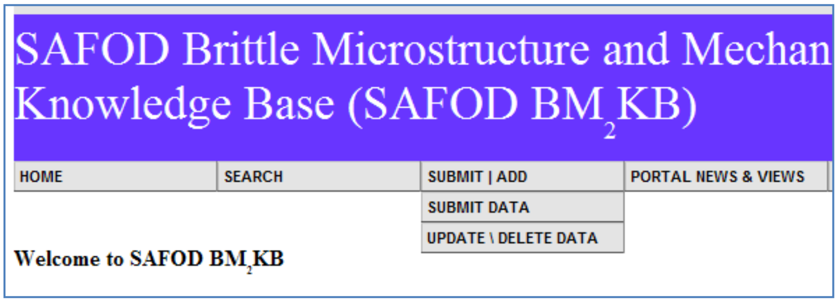
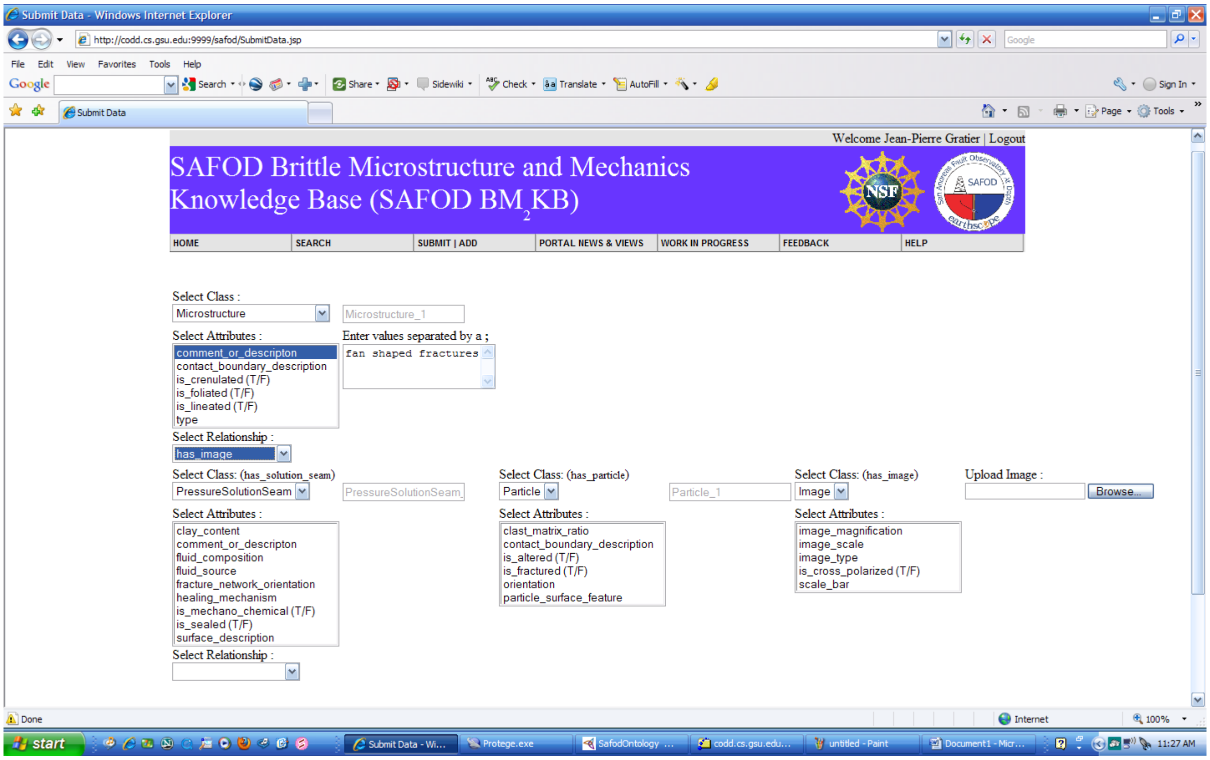


Figure 6.38. Screen shot of menu option to submit data.

The following screen allows the user to select a class from a drop-down list. The user selects the initial class concept based on the data that he or she would like to input information about. Other options will then be available based on the relationships associated with this class. Depending on the class concept chosen, a list of available property attributes and relationships appears next. The user may highlight as little or many attributes as he wishes and then input the values in the box to the right. The property value relationships are populated in the associated drop-down and the user may select one or multiple properties. The user may also upload an image or table associated with their data. These properties establish links to other class concepts, and the process continues until the user selects the 'submit' button.

As an example, suppose that Dr. Gratier intends to input the data from the following statement: a solution seam that is fractured and sealed by calcite fibers parallel to extension. The contact grains are fractured network in a fan-shaped network. Certain minerals were unaltered (quartz), dissolved (calcite and feldspar), or passively concentrated (phyllosilicates).

To achieve inputting this information, several options could be used, but the following screen-shots are one example of how Dr. Gratier may input his findings (Figure 6.39).



SAFOD Brittle Microstructure and Mechanics Knowledge Base (SAFOD BM$_2$KB)

Select Class : Microstructure — Microstructure_1

Select Attributes : comment_or_descripton; contact_boundary_description; is_crenulated (T/F); is_foliated (T/F); is_lineated (T/F); type

Enter values separated by a ; fan shaped fractures

Select Relationship : has_image

Select Class: (has_solution_seam) PressureSolutionSeam — PressureSolutionSeam_

Select Attributes : clay_content; comment_or_descripton; fluid_composition; fluid_source; fracture_network_orientation; healing_mechanism; is_mechano_chemical (T/F); is_sealed (T/F); surface_description

Select Relationship :

Select Class: (has_particle) Particle — Particle_1

Select Attributes : clast_matrix_ratio; contact_boundary_description; is_altered (T/F); is_fractured (T/F); orientation; particle_surface_feature

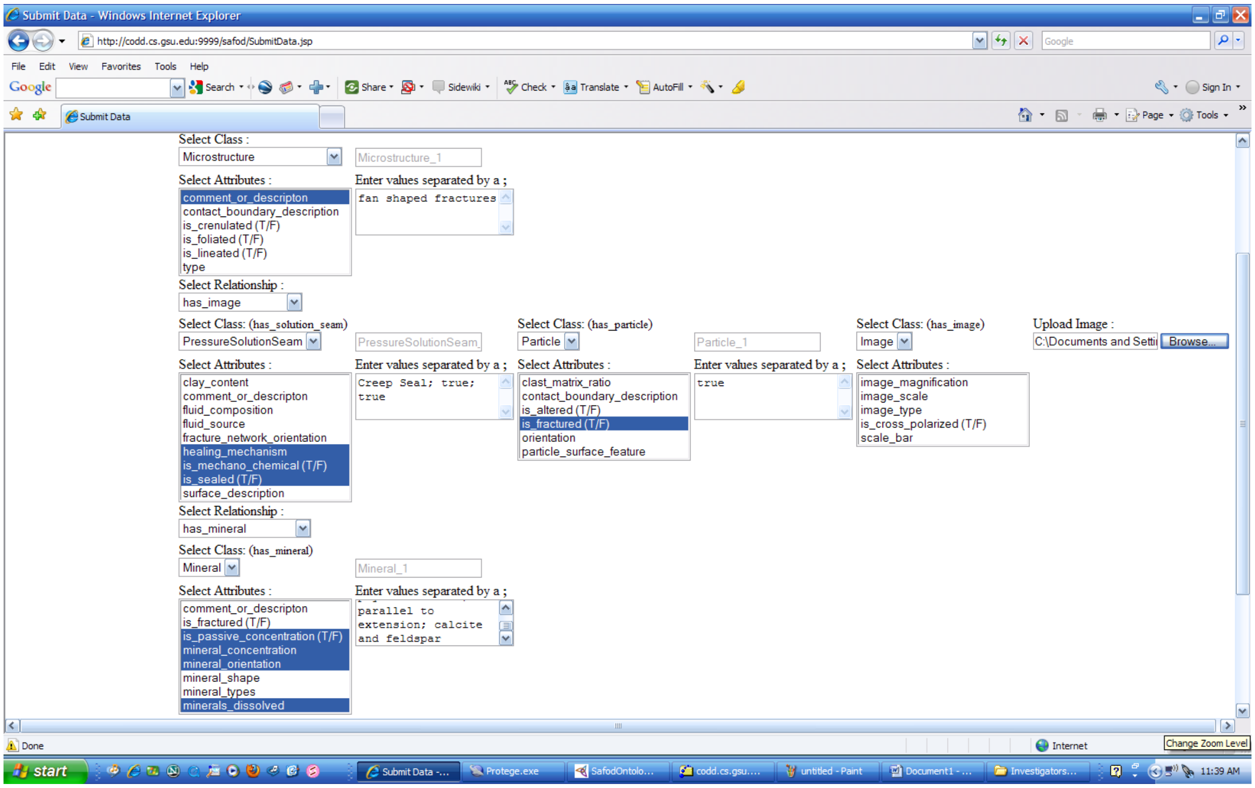
Select Class: (has_image) Image

Select Attributes : image_magnification; image_scale; image_type; is_cross_polarized (T/F); scale_bar

Upload Image :

Figure 6.39. Selection of Microstructure class and related properties for ‘has_solution_seam,’ has_particle,’ and ‘has_image.’

Dr. Gratier, as an investigator who is submitting data, would click on all attributes of classes he wishes to input data for, and then type the respective attributes in the box to the right. (A series of attributes are delimited using a semi-colon.) He may also continue to select relationships in further dropdown lists. For example, PressureSolutionSeam has a relationship with the class Mineral. He could then select ‘has_mineral’ from the list and enter the associated attributes. He could also click on the ‘Browse’ button to upload an image (Figure 6.40).



Select Class : Microstructure | Microstructure_1

Select Attributes :
comment_or_descripton
contact_boundary_description
is_crenulated (T/F)
is_foliated (T/F)
is_lineated (T/F)
type

Enter values separated by a ; fan shaped fractures

Select Relationship : has_image

Select Class: (has_solution_seam) PressureSolutionSeam | PressureSolutionSeam_

Select Attributes :
clay_content
comment_or_descripton
fluid_composition
fluid_source
fracture_network_orientation
healing_mechanism
is_mechano_chemical (T/F)
is_sealed (T/F)
surface_description

Enter values separated by a ; Creep Seal; true; true

Select Class: (has_particle) Particle | Particle_1

Select Attributes :
clast_matrix_ratio
contact_boundary_description
is_altered (T/F)
is_fractured (T/F)
orientation
particle_surface_feature

Enter values separated by a ; true

Select Class: (has_image) Image

Select Attributes :
image_magnification
image_scale
image_type
is_cross_polarized (T/F)
scale_bar

Upload Image : C:\Documents and Setti

Select Relationship : has_mineral

Select Class: (has_mineral) Mineral | Mineral_1

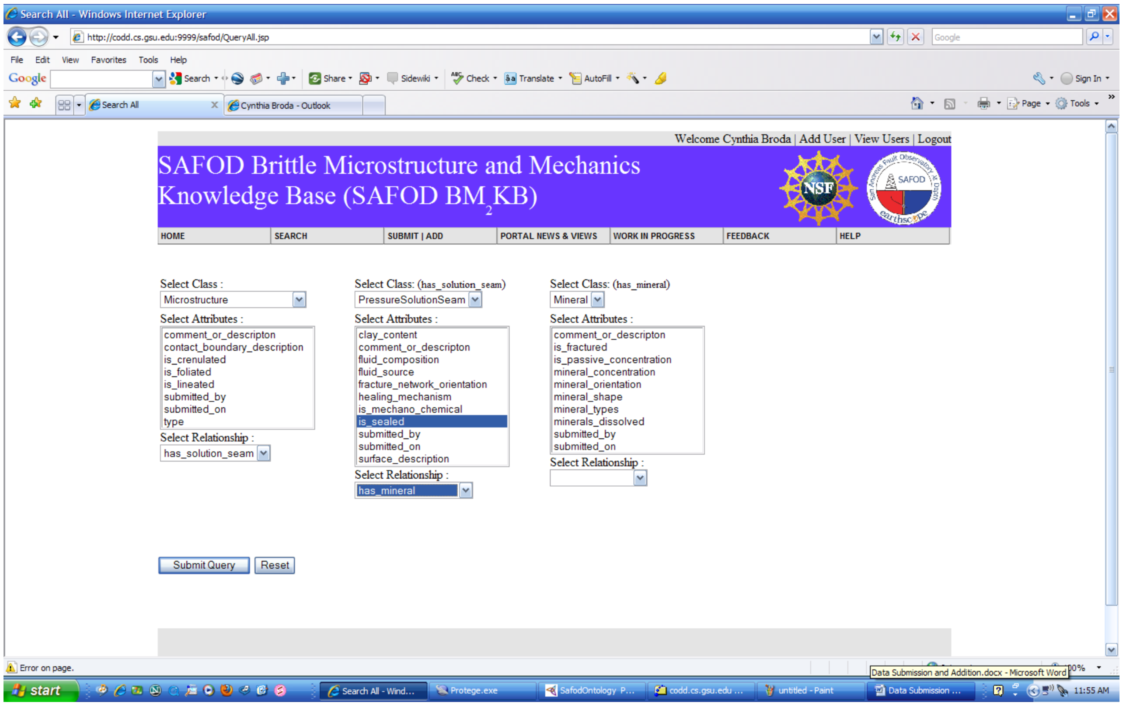
Select Attributes :
comment_or_descripton
is_fractured (T/F)
is_passive_concentration (T/F)
mineral_concentration
mineral_orientation
mineral_shape
mineral_types
minerals_dissolved

Enter values separated by a ; parallel to extension; calcite and feldspar

Figure 6.40. Screen shot of data being entered for the solution seam and associated mineral and particle information. The ability to upload an image is also shown.

A total of eight classes and associated attributes/properties may be entered at one time. If the user wants to add further data, he may later update the previous records with additional information. The investigator then selects the 'submit' button to send data to the administrator (e.g., Cindi Broda), at which time it may be added to the knowledge base. After the data has been added, it will be available for modification or query/search.

The menu option for 'Search' provides three methods on which to search: General Search, Search by Sample, and Search by Investigator. The General Search option allows the user to perform a general search based on classes and their respective attributes that are stored in the knowledge base. The user can select up to eight related classes for a single query. The user clicks the menu option for 'Search' and then selects the desired class from the drop down list. The user then selects all desired attributes or related properties that he or she would like to see. Based on the previous example, Dr. Gratier may wish to search for any microstructures that have a sealed pressure solution seam, and all associated mineral information. This could be accomplished by selecting "Microstructure", then its associated relationship with "PressureSolutionSeam". Within the attributes, select "is_fractured" (this will display only that attribute for PressureSolutionSeam), followed by selection of the relationship for "has_Mineral" (Figure 6.41). If data is available for the selection criteria, then the screen will return the query results. It will also provide a blue link with the name of the concepts that can be clicked on for further information (Figures 6.42).



Select Class : Microstructure
Select Attributes :
comment_or_descripton
contact_boundary_description
is_crenulated
is_foliated
is_lineated
submitted_by
submitted_on
type
Select Relationship : has_solution_seam

Select Class: (has_solution_seam) PressureSolutionSeam
Select Attributes :
clay_content
comment_or_descripton
fluid_composition
fluid_source
fracture_network_orientation
healing_mechanism
is_mechano_chemical
is_sealed
submitted_by
submitted_on
surface_description
Select Relationship : has_mineral

Select Class: (has_mineral) Mineral
Select Attributes :
comment_or_descripton
is_fractured
is_passive_concentration
mineral_concentration
mineral_orientation
mineral_shape
mineral_types
minerals_dissolved
submitted_by
submitted_on
Select Relationship :

Figure 6.41. Screen shot of search based on microstructures with sealed solution seams and their related mineral information.

SAFOD Brittle Microstructure and Mechanics Knowledge Base (SAFOD BM$_2$KB)

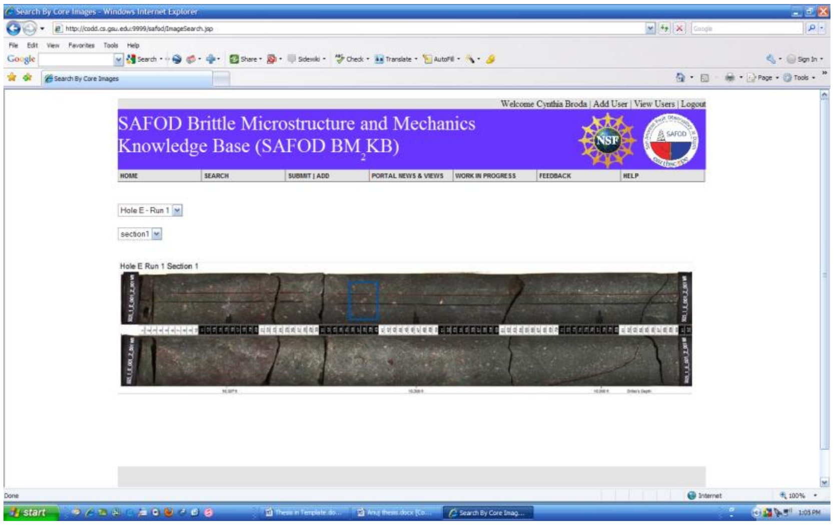
Detailed Query Result

Microstructure -- Microstructure_1	
Property	Value
is_crenulated	
submitted_by	Jean-Pierre Gratier
is_foliated	
submitted_on	2010-03-25
is_lineated	
type	
comment_or_descripton	fan shaped fractures
contact_boundary_description	

has_solution_seam	PressureSolutionSeam_1
has_mineral	Mineral_1

Figure 6.42. Screen shot of search results with concepts available to display further information (in blue).

For the 'Search by Sample' option, a core viewer is loaded. A dropdown box is provided so that the user may select the desired Hole, Run, and Section. Once the hole, run, and section are selected, the image of the related core appears. The user may have the option to view a particular section of the core if it is highlighted by a blue box (Figure 6.43).



SAFOD Brittle Microstructure and Mechanics Knowledge Base (SAFOD BM$_2$KB)

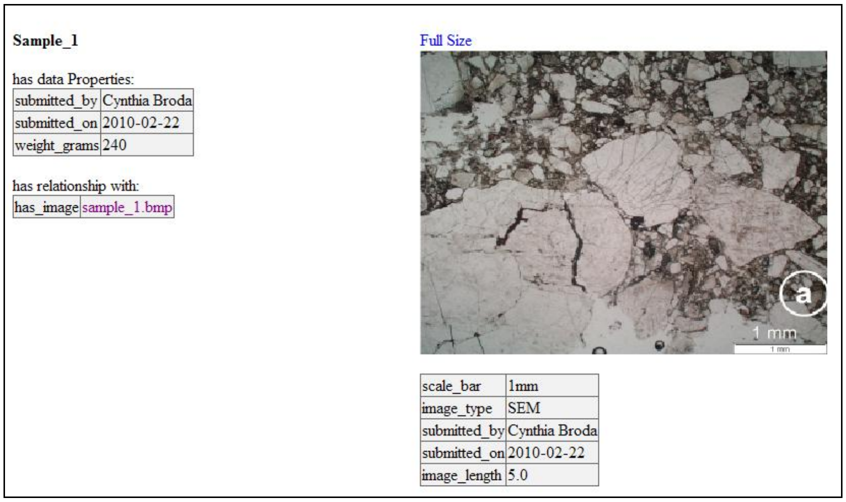
Hole E - Run 1

section1

Hole E Run 1 Section 1

Figure 6.43. Screen shot of image for Hole E, Run 1, Section 1 with blue box that may be clicked for detailed information on the related thin section.

The user may click on the blue box in order to view more information about the sample indicated on the core interval. Such information will include the investigator who submitted data regarding the sample, weight of the sample, and if there are other concept classes related to the sample (e.g., microstructural details or images). If there are other related concepts, these may also be clicked for related detail. For example, if the sample has an associated image, the user may click the (blue) image name for retrieval (Figure 6.44).



Sample_1

has data Properties:

submitted_by	Cynthia Broda
submitted_on	2010-02-22
weight_grams	240

has relationship with:

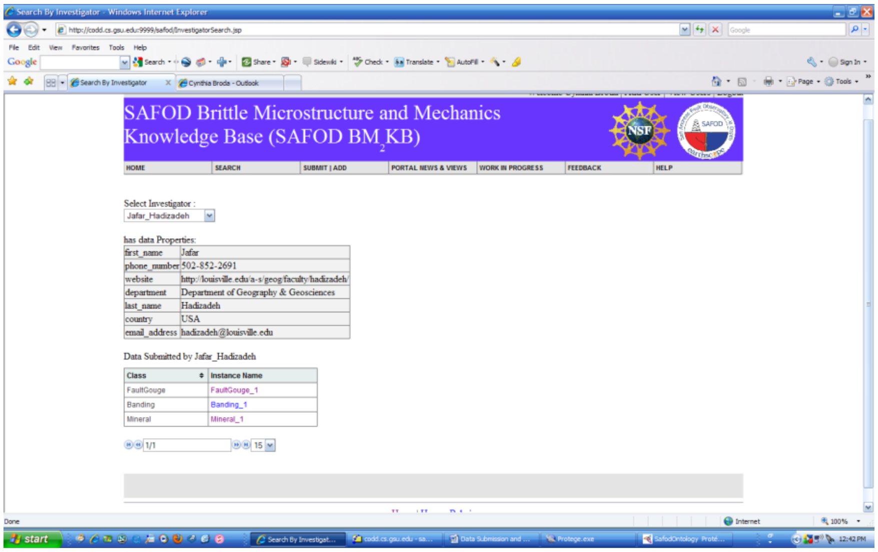
has_image	sample_1.bmp

Full Size

scale_bar	1mm
image_type	SEM
submitted_by	Cynthia Broda
submitted_on	2010-02-22
image_length	5.0

Figure 6.44. Screen shot of image associated image with core sample image.

The option ‘Search by Investigator’ allows a user to view the data input by other investigators. This menu leads to a drop-down list of investigators from the SAFOD project who have expressed interest in using this ontology. The user selects the investigator for whom they are interested, and that investigator’s data will appear on the next screen (Figure 6.45). Specific data concepts show under ‘instance name’ and these blue links may be clicked for further information.



SAFOD Brittle Microstructure and Mechanics Knowledge Base (SAFOD BM$_2$KB)

Select Investigator : Jafar_Hadizadeh

has data Properties:

first_name	Jafar
phone_number	502-852-2691
website	http://louisville.edu/a-s/geog/faculty/hadizadeh/
department	Department of Geography & Geosciences
last_name	Hadizadeh
country	USA
email_address	hadizadeh@louisville.edu

Data Submitted by Jafar_Hadizadeh

Class	Instance Name
FaultGouge	FaultGouge_1
Banding	Banding_1
Mineral	Mineral_1

1/1 15

Figure 6.45. Screen shot of data submitted for an investigator.

The SAFOD ontology provides a way by which investigators can remotely share data and view the ongoing work of other scientists. Hopefully this application tool will assist investigators in furthering their own research by locating patterns in data, or by gaining additional insights from a shared community.

CHAPTER 7

CONCLUSIONS

As knowledge grows and the amount of information available continues to increase, individuals and communities face an ever-increasing need for tools to harness these resources. Often, these precious resources (such as data or expert information) are difficult to access, connect, or correlate. The abundance of information, and the related details, can be overwhelming.

The ontology for this thesis project has made a direct attempt to provide a user-friendly tool on the Web that addresses the need for information storage, accessibility, and processing. It provides a way for a community of scientists (working on the SAFOD project) to share research progress and results, while allowing for inference connections to be revealed. This project is an instance in the evolutionary process of information connectivity. It is available for use in its existing structure, or it could be expanded or integrated into other systems.

Many forms of future work are possible for the use and application of this knowledge base project. In its current state, the ontology is static and accounts for data at a set point in time. It could be developed into a process-oriented ontology that would consider the space-time continuum. In order to incorporate the concept of time, the ontology would most likely benefit from an upper-level ontology connection. If the ontology were process-oriented, it could move beyond the concepts of brittle fracture and include concepts such as ductile shear, aseismic creep, mylonitization, etc. It could conduct operations based on a dynamic series of events, rather than instantaneous snapshots in time.

The ontology could also include other features related to structural geology. It could contain more detailed models regarding deformed landforms, for example. Expanded concepts related to faulting, and various types of faulting could be included. A more extensive definition of rock types would be simple to incorporate. There are many ways to build on the current ontology. Expansion is achievable within the current community of users, or by a other geologic communities. Other community of users could employ part or all of the SAFOD ontology.

In addition, features related to the application interface could be altered or enhanced. For example, the initial drop-down class list that displays all classes could be refined. It could show only superclasses in the first list, and then direct the user on a more specific path of input or query based on preliminary entries. This enhancement would hone in on the intent of the user, and could allow for multiple property options to be selected throughout the process (because less information would be presented on the screen, making the input more manageable).

There are a profuse number of ways that this project could assimilate further design and development applications. The objective of the user would be the basis for direction among the numerous enhancement possibilities. After all, the reusability and capacity to share ontologies are among some of their great benefits.

REFERENCES

Alexander, P., (07 Nov. 2008), "Document Domain Knowledge" In The MMI Guides: Navigating the World of Marine Metadata, <http://marinemetadata.org/guides/vocabs/ont/importance/domainknowledge>, (October 2009).

Allemang, D. and Hendler , J., 2008, Semantic Web for the Working Ontologist, United States, Morgan Kaufmann Publishers, 352 p.

Arrowsmith, J R., 1995, Coupled tectonic deformation and geomorphic degradation along the San Andreas Fault System, PhD. Dissertation (Appendix C), Stanford University, pp. 312-346.

Babaie, H., 2009, Depiction of Geological Processes with Ontologies, Geological Society of America *Abstracts with Programs*, Vol. 41, No. 7, p. 556.

Becket, D., (23 September 2005), Resource Description Framework (RDF) Resource Guide, <http://planetrdf.com/guide/>, (September, 2009).

Ben-Zion, Y. and Rice, J. R., 1993, Earthquake failure sequences along a cellular fault zone in a three dimensional elastic solid containing asperity and nonasperity regions, J. Geophys. Res., 98, 14,109–14,131.

Berners-Lee, T., 2001, The Semantic Web, Scientific American, Vol. 284, Issue 5, 10 p.

Brown, R. D., 1990, Earthquake history, 1769-1989, Contributing document to The San Andreas Fault System, California, U.S. Geological Survey Professional Paper 1515, Washington, United States Government Printing Office, pp. 155-156.

Champion, D. E., Howell, D. G., and Gromme, C. S., 1984, Paleomagnetic and geologic data indicating 2500 km of northward displacement for the Salinian and related terranes, California, Journal of Geophysical Research, Vol. 89 (B9), pp. 7736 – 7752.

Corazzon, R., (12 September 2009), Theory and History of Ontology. A Resource Guide for Philosophers, <http://www.formalontology.it> , (September, 2009).

Daly, J., (August 2003), World Wide Web Consortium Issues Web Ontology Language Candidate Recommendations, Emerging Ontology Standard, OWL, Strengthens Semantic Web Foundations, <http://www.w3.org/2003/08/owl-pressrelease>, (September, 2009).

Desbois, G. and Urai, J. L., 2009, Imaging porosity at pore scale in SAFOD samples using the combination of Ar-beam cross sectioning and SEM microscopy: Preliminary results, poster at Earthscope meeting, Boise, ID, USA.

Dibblee, T. W., Graham, T. M., Blissenbach, J. J., Mariant, and Wentworth, C. M., 1999, Regional geologic map of San Andreas and related faults in Carrizo Plain, Temblor, Caliente and La Panza Ranges and vicinity, California: A digital database, U.S. Geological Survey, Open File Report, 99-14.

Earthscope SAFOD Info, (30 April 2008), First SAFOD Core Samples Distribution, <http://www.earthscope.org/es_doc/safod/SAFOD_Core_Sample_Distribution.pdf>, (January, 2010).

Earthscope Website, 2009, San Andreas Fault Observatory at Depth, <http://www.earthscope.org/observatories/safod>, (August, 2009).

Elam, T., 2000, The San Andreas Fault, Buena Vista Museum of Natural History, <http://www.sharktoothhill.org/fault.html>, (September, 2009).

Ellsworth, W., 1990, Earthquake history, 1769-1989, Contributing document to The San Andreas Fault System, California, U.S. Geological Survey Professional Paper 1515, Washington, United States Government Printing Office, pp. 155-156.

Furlong, K. P., Schwartz, S. Y., 2004, Influence of the Mendocino triple junction on the tectonics of coastal California. Annual Review of Earth and Planetary Sciences, Vol. 32, pp. 403–433.

Gil, Y. and Ratnakar, V., 2002, A Comparison of (Semantic) Markup Languages, Published at Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference, Florida, USA, pp. 413-418.

Gratier, J. P., and Doan, M. L., 2009, Fault permeability and strength evolution during the interseismic period: insights from microstructures in SAFOD 3 core samples, poster at Earthscope meeting, Boise, ID, USA.

Gruber, T., 1993, What is an Ontology?, <http://www-ksl.stanford.edu/kst/what-is-anontology>, (August, 2009).

Hadizadeh, J., Blackburn, E., and Babaie, H.,2009, A preliminary study of deformation in gouge from actively creeping fault zone encountered in the SAFOD phase 3 drill hole; implications for exhumed gouges and shear localization models, poster at Earthscope meeting, Boise, ID, USA.

Hickman, S., Zoback., M., Ellsworth, W., et al., 2007, Structure and Properties of the San Andreas Fault in Central California: Recent Results from the SAFOD Experiment, Scientific Drilling, Vol.1, pp. 29-32.

Horridge, M., Rector, A., et al., 2004, A Practical Guide to Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools, University of Manchester, USA, Vol. 27, 117 p.

Howard, D., 1979, Geologic history of middle California, California Natural History Guides, London, University of California Press, 113 p.

International Continental Scientific Drilling Program (ICDP), (19 December 2007), Photographic Atlas of the SAFOD Phase 3 Cores, <http://www.icdp-online.org/upload/projects/safod/phase3/Core_Photo_Atlas_v3.pdf>, (January, 2010).

Irwin, W. P., 1990, The San Andreas Fault System, California, U.S. Geological Survey Professional Paper 1515, Washington, United States Government Printing Office, 283 p.

Lacy, L. W., 2005, OWL: Representing Information Using the Web Ontology Language, USA, Trafford Publishing, 302 p.

Larson, S. and Martone, M., 2009, Ontologies for neuroscience: What are they and what are they good for?, Frontiers in Neuroscience, Volume 3, Issue 1, p. 60-67.

Little, E., 2003, A Proposed Methodology for the Development of Application-Based Formal Ontologies, Paper presented at the Workshop on Reference Ontologies vs. Application Ontologies, University of Hamburg, 14 p.

Liu, Y., Zhang, H., Thurber, C., and Roecker, S., 2007, Shear wave anisotropy in the crust around the San Andreas fault near Parkfield: spatial and temporal analysis, Geophysical Journal International, Vol. 172, Issue 3, pp. 957–970.

Mark, D., et al., 2003, Ontology, Natural Language, and Information Systems: Implications of Cross-Linguistic Studies of Geographic Terms. in AGILE Conference on Geographic Information Science, Lyon, France: AGILE conference.

McPhee, D., Jachens, R., Wentworth, C., 2004, Crustal structure across the San Andreas Fault at the SAFOD site from potential field and geologic studies, Geophysical Research Letters, Vol. 31, L12S03.

Mittempergher, S., Di Toro, G., and Spiess, R., 2009, Evidences for multiple episodes of fluid flow in veins of the SAFOD phase 3 cores, poster at Earthscope meeting, Boise, ID, USA.

Moore, D., and Rymer, M., 2007, Talc-bearing serpentinite and the creeping section of the San Andreas Fault, Nature, Vol. 448, pp. 795-797.

Nadeau, R. M. and McEvilly, T. V., 1997, Seismological studies at Parkfield V: characteristic microearthquake sequences as fault-zone drilling targets, Bulletin of the Seismological Association of America, Vol. 87, pp. 1463-1472.

National Parks Service U.S. Department of the Interior, (05 August 2009), Geology of the Golden Gate Headlands (Poster), <http://www.nps.gov/goga/forteachers>, (August, 2009).

National Science Foundation Cyberinfrastructure Council, 2006, NSF's Cyberinfrastructure Vision for 21st Century Discovery, Version 5.0, 32 p.

Nester, I., 2008, San Andreas Fault Zone, Global Tectonics at Emporia Education, <http://www.emporia.edu/earthsci/student/nester1/sandreas.htm>, (September, 2009).

Noy, N. F., and McGuinness, D. L., 2001, Ontology development 101: A guide to creating your first ontology, Technical Report SMI-2001-0880, Stanford Medical Informatics.

Obitko, M., 2007, Ontologies and Semantic Web, <http://www.obitko.com/tutorials/ontologies-semantic-web/semantic-web.html>, (November, 2009).

Park, S. K., Roberts, J. J., 2003, Conductivity structure of the San Andreas fault, Parkfield, revisited, Geophysical Research Letters, 30(16), 1842.

Pollock, J., 2009, Semantic Web for Dummies, NJ, USA, For Dummies, 432 p.

Raskin, R., 2007, Community Science Ontology Development, Geoinformatics Conference (17–18 May 2007), Paper No. 3-16.

Schulz, S. and Wallace, R. E., (24 June 1997), USGS: The San Andreas Fault, <http://pubs.usgs.gov/gip/earthq3/safaultgip.html>, (September, 2009).

Sinha, A. K., Malik, Z., Rezgui, A., et al., 2007, Ontology Packages for GeoScience, GEON Technology Innovations, Technical Report 01-2007, Version 1, pp. 20-21.

Smith, B., 2003, Ontology, Preprint version of chapter "Ontology", in L. Floridi (ed.), Blackwell Guide to the Philosophy of Computing and Information, Oxford: Blackwell, p. 155-166.

Sowa, J. F., 1999, Knowledge Representation: Logical, Philosophical, and Computational Foundations, Pacific Grove, CA, Brooks Cole Publishing Co., 512 p.

Staab, S., Erdmann, M., and M¨adche, A., 2000, An extensible approach for modeling ontologies in rdf(s), In First ECDL'2000 SemanticWebWorkshop, Lisbon, Portugal.

Stoffer, P.W., 2005, The San Andreas and San Gregorio Fault Systems in San Mateo County: In The San Andreas Fault in the San Francisco Bay Area, California: A Geology Fieldtrip Guidebook to Selected Stops on Public Lands (Chapter 8), USGS, Open-File Report 2005-1127.

Stoffer, P.W., 2006, Where's the San Andreas Fault? A guidebook to tracing the fault on public lands in the San Francisco Bay region: U.S. Geological Survey General Interest Publication 16, 123 p.

Swartz, A., 2002, The Semantic Web in Breadth, < http://logicerror.com/semanticWeb-long>, (November 5, 2009).

Tauberer, J., 2005, Quick Intro to RDF, <http://www.rdfabout.com/quickintro.xpd>, (November, 2009).

Toppozada, T. R., Branum, D., Reichle, M. S., et al., 2002, California M > 5.5 earthquake history, Bulletin of the Seismological Society of America, Vol. 92, no. 7, pp. 2555-2601.

Unsworth, M. J., Malin, P. E., Egbert, G. D., et al., 1997, Internal structure of the San Andreas fault at Parkfield, California, Geology, Vol. 25, pp. 359-362.

USGS Earthquakes Hazards Program, 2009, <http://earthquake.usgs.gov/regional/nca/1906/18april/howlong.php>, (September, 2009).

USGS Earthquakes Hazards Program, 2009, <http://earthquake.usgs.gov/research/parkfield/shake/>, (September, 2009).

USGS Earthquakes Hazards Program, (29 January 2009), Repeating Earthquakes, <http://earthquake.usgs.gov/research/parkfield/repeat.php>, (September, 2009).

USGS Western Earth Surface Processes Team, 2006, Geologic History of the San Andreas Fault System, <http://geomaps.wr.usgs.gov/socal/geology/geologic_history/san_andreas_history.html>, (August, 2009).

Wallace, R. E., 1990, The San Andreas Fault System, California, U.S. Geological Survey Professional Paper 1515, Washington, United States Government Printing Office, 283 p.

World Wide Consortium (W3C), 2000, XML slide, “The Layer Cake” Architecture < http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html>, (November, 2009).

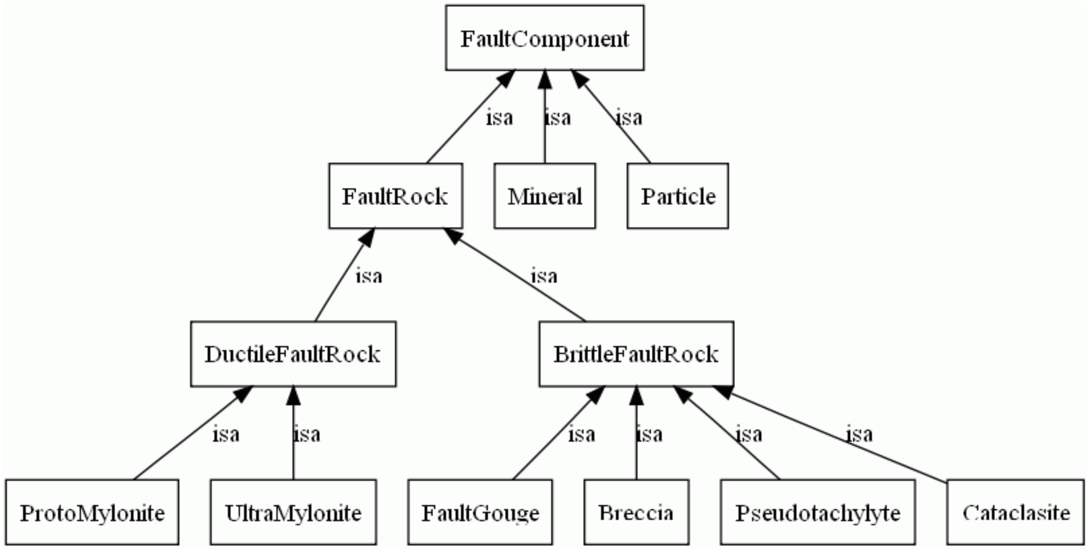
World Wide Consortium (W3C), (10 February 2004), RDF Semantics W3C Recommendation, <http://www.w3.org/TR/rdf-mt/>, (November, 2009).

Zoback, M., Hickman, S. & Ellsworth, W., 2005, Overview of SAFOD Phases 1 and 2: Drilling, Sampling and Measurements in the San Andreas Fault Zone at Seismogenic Depth, Eos Trans. AGU, Vol. 86 (52), Meeting Supplement, Abstract T23E-01.

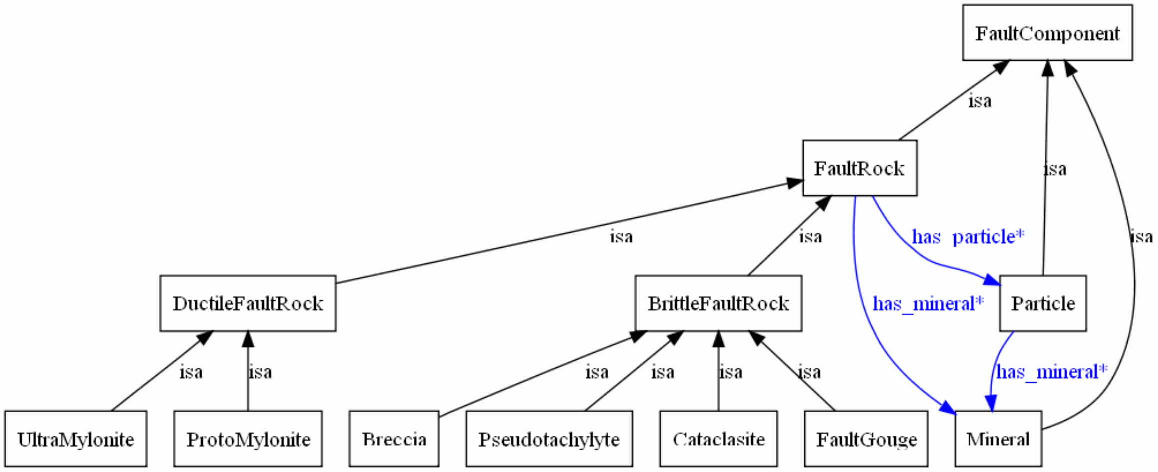
APPENDIX A

Class Hierarchies and Associated Properties

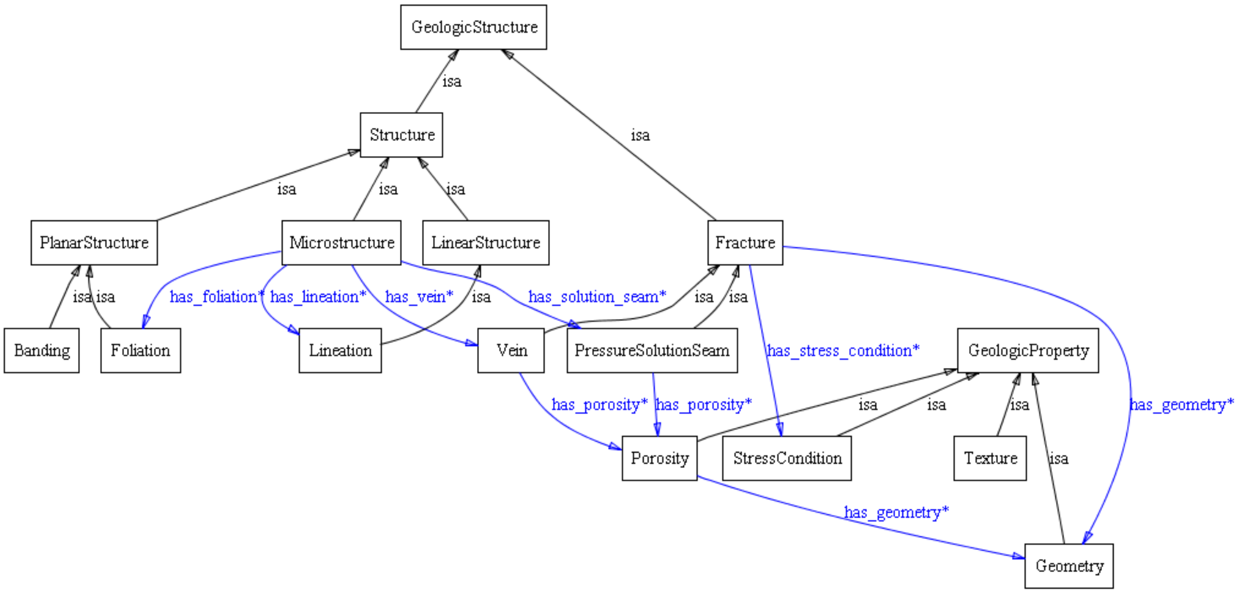
1) Subclass hierarchy for FaultComponent:



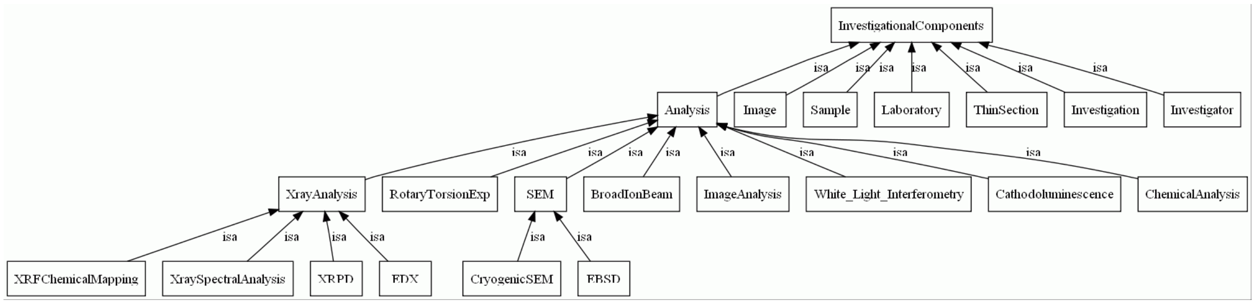
2) Properties associated with FaultComponent hierarchy: (Note that subclasses inherit properties. For example, BrittleFaultRock inherits the properties of FaultRock. However, the class Particle does not because it is not a subclass.)



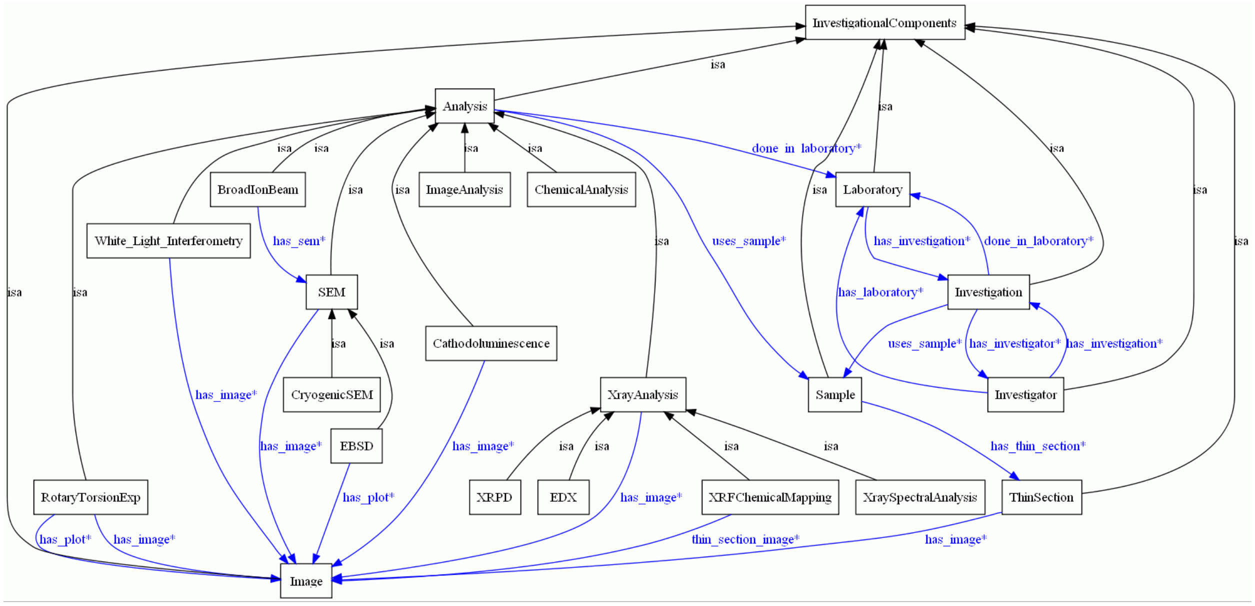
3) Class hierarchy and associated properties for GeologicStructure and GeologicProperty:



4) Hierarchy for InvestigationalComponents:



5) Properties related to InvestigationalComponents hierarchy:

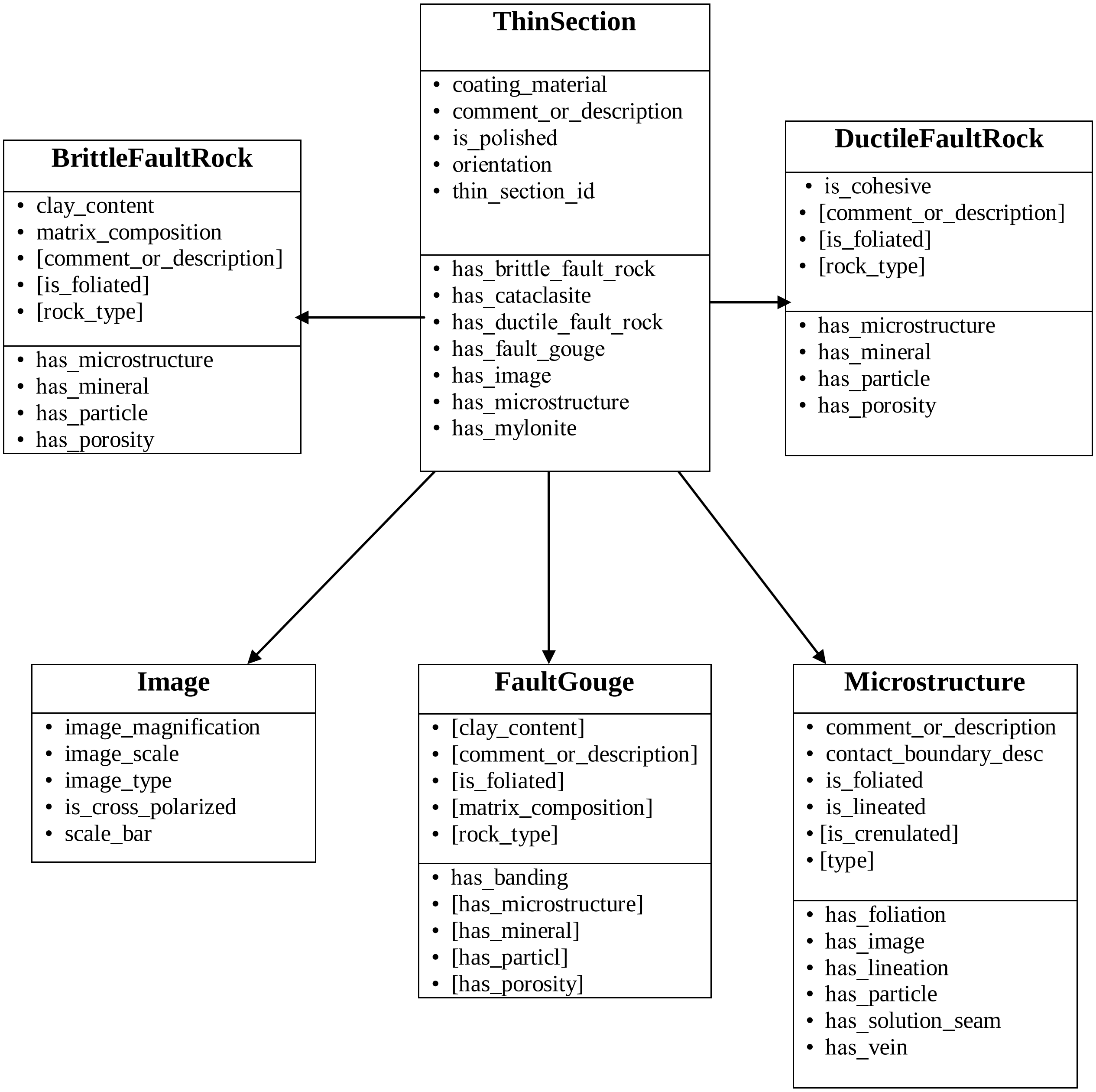


APPENDIX B

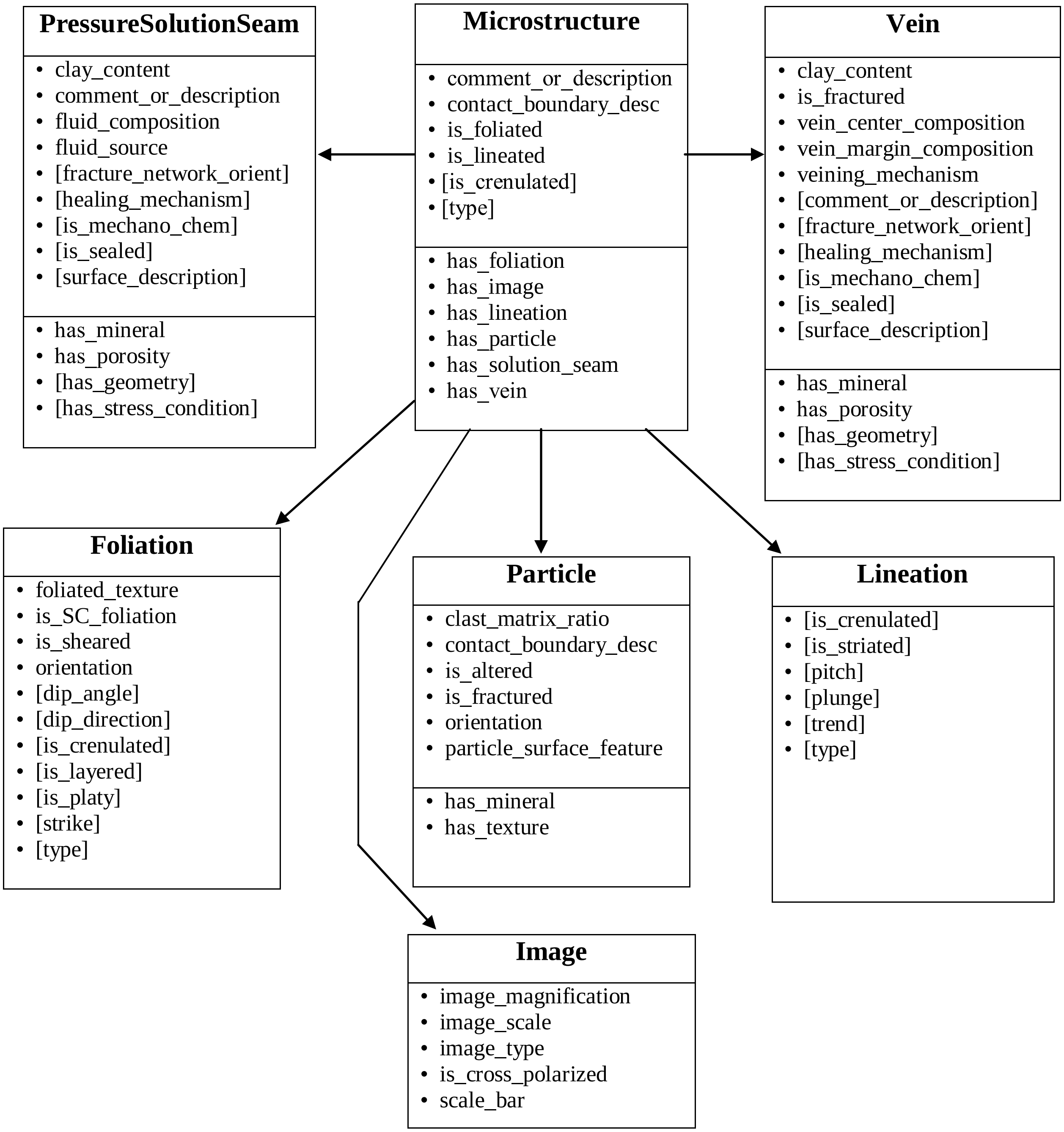
Examples of Detailed Class Information

- Note that inherited properties are in brackets. (This note all class diagrams included in Appendix B.)

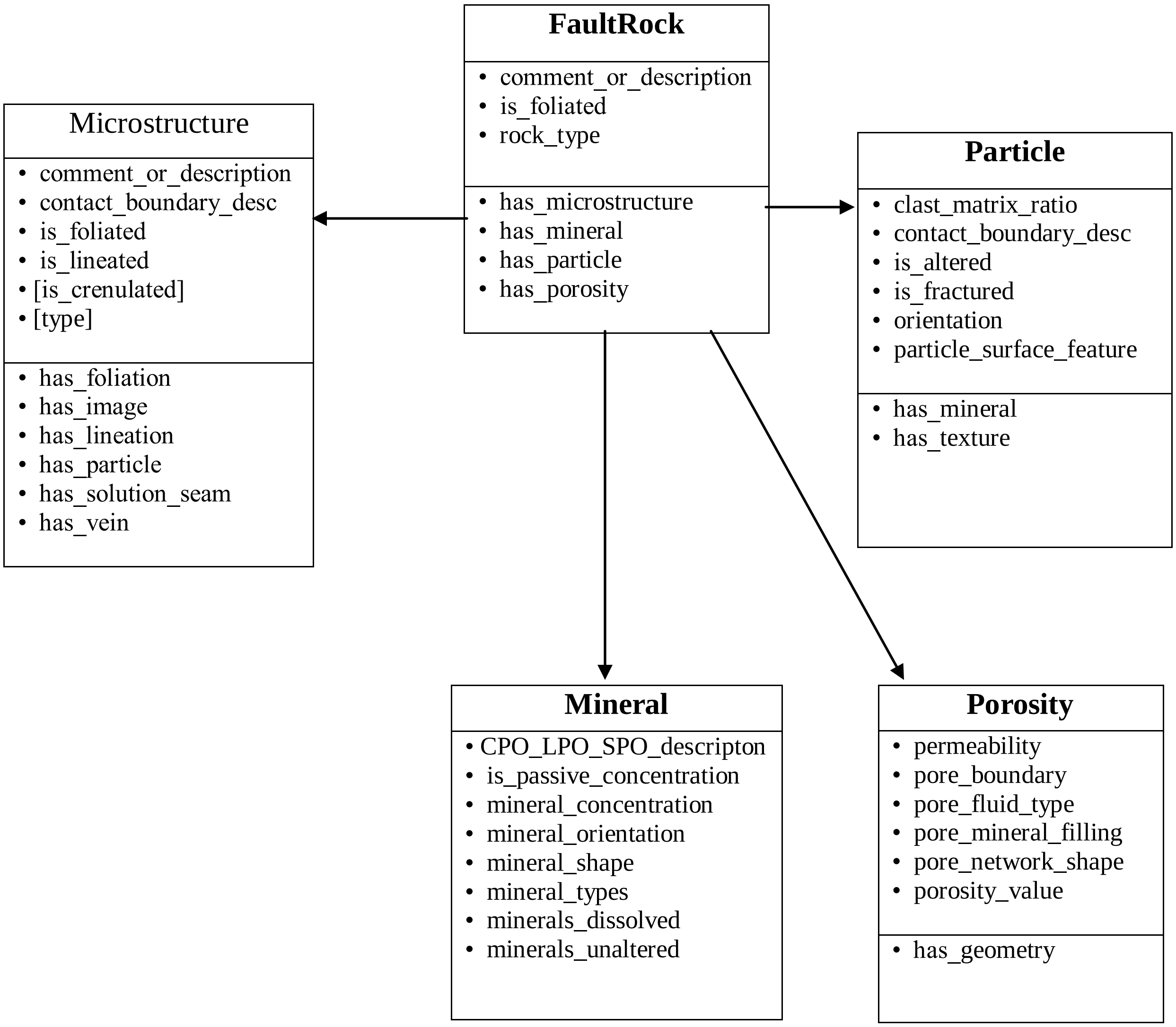
Thin Section



Microstructure



FaultRock


FaultRock
• comment_or_description
• is_foliated
• rock_type
• has_microstructure
• has_mineral
• has_particle
• has_porosity
Microstructure
• comment_or_description
• contact_boundary_desc
• is_foliated
• is_lineated
• [is_crenulated]
• [type]
• has_foliation
• has_image
• has_lineation
• has_particle
• has_solution_seam
• has_vein
Particle
• clast_matrix_ratio
• contact_boundary_desc
• is_altered
• is_fractured
• orientation
• particle_surface_feature
• has_mineral
• has_texture
Mineral
• CPO_LPO_SPO_descripton
• is_passive_concentration
• mineral_concentration
• mineral_orientation
• mineral_shape
• mineral_types
• minerals_dissolved
• minerals_unaltered
Porosity
• permeability
• pore_boundary
• pore_fluid_type
• pore_mineral_filling
• pore_network_shape
• porosity_value
• has_geometry