Ontology Summit 2014 Communiqué
Semantic Web and Big Data Meets Applied Ontology

Lead Editors: Michael Gruninger, Leo Obrst
Co-Editors: Ken Baclawski, Mike Bennett, Dan Brickley, Gary Berg-Cross, Pascal Hitzler, Krzysztof Janowicz, Christine Kapp, Oliver Kutz, Christoph Lange, Anatoly Levenchuk, Francesca Quattri, Alan Rector, Todd Schneider, Simon Spero, Anne Thessen, Marcela Vegetti, Amanda Vizedom, Andrea Westerinen, Matthew West, Peter Yim.

Executive Summary

The role that ontologies play or can play in designing and employing semantic technologies has been widely acknowledged by the Semantic Web and Linked Data communities. But the level of collaboration between these communities and the Applied Ontology community has been much less than expected. And ontologies and ontological techniques appear to be of marginal use in Big Data and its applications.

To understand this situation and foster greater collaboration, Ontology Summit 2014 brought together representatives from the Semantic Web, Linked Data, Big Data and Applied Ontology communities, to address three basic problems involving applied ontology and these communities:

(1) The role of ontologies [in these communities],
(2) Current uses of ontologies in these communities, and
(3) Engineering of ontologies and semantic integration.

The intent was to identify and understand: (a) causes and challenges (e.g. scalability) that hinder reuse of ontologies in SW and LD, (b) solutions that can reduce the differences between ontologies on and off line, and (c) solutions to overcome engineering bottlenecks in current Semantic Web and Big Data applications.

Over the past four months, presentations from, and discussions with, representatives of the Semantic Web, Linked Data, and Applied Ontology communities have taken place across four tracks. Each Track focused on different aspects of this year's Summit topic: (Track A) Investigation of sharable and reusable ontologies; (Track B) Tools, services and techniques for a comprehensive and effective use of ontologies; (Track C) Investigation of the engineering bottlenecks and the ways to prevent and overcome them; (Track D) Enquiry on the variety problem in Big Data.

In addition to the four Tracks' activities there was a Hackathon. Six different Hackathon projects took place, all available at their individual project public repositories. An online Community Library and an online Ontology Repository have been created as freely accessible Community resources.

This Ontology Summit 2014 Communique presents a summary of the results, original in its attempt both to merge different communities' discourses and to achieve consensus across the Summit participants with respect to open problems and recommendations to address them.

1. Introduction, Scope, Motivation

Since the beginnings of the Semantic Web, ontologies have played key roles in the design and deployment of new semantic technologies. Yet over the years, the level of collaboration between the Semantic Web and Applied Ontology communities has been much less than expected. Within Big Data applications, ontologies appear to have had little use or impact.

Ontology Summit 2014 provided an opportunity for building bridges between the Semantic Web, Linked Data, Big Data, and Applied Ontology communities. On the one hand, the Semantic Web, Linked Data, and Big Data communities bring a wide array of real problems (performance and scalability challenges and the variety problem in Big Data) and technologies (like automated reasoning tools) that make use of ontologies. There is a particular emphasis on the Web in making sense of data and information distributed over the Web. This is in contrast to, say, using local reasoners on small ontologies, where the only "Web" aspects are using IRIs as symbol names, and employing inference rules based on an open (or sometimes closed) world assumption. On the other hand, the Applied Ontology community brings a large body of ontological analysis techniques and reusable ontologies.

Three focus areas arose from the Summit:

1. How are ontologies actually being used in Semantic Web and Big Data applications, and how does this differ from existing applications of ontologies?
2. How can the Semantic Web and Big Data communities share and reuse the wide array of ontologies that are currently being developed?
3. To what extent can automation and tools help overcome ontology engineering bottlenecks?

2. Using Ontologies with Big Data and the Semantic Web

Semantic technologies such as ontologies and related reasoning play a major role in the Semantic Web and are increasingly being applied to help process and understand information expressed in digital formats. Indeed, the derivation of assured knowledge from the connection of diverse (and linked) data is one of the main themes of Big Data.

One challenge in deriving assured knowledge is to build and use common semantic content while avoiding silos of concepts in different ontologies. Examples of such content are whole or partial ontologies, ontology modules, ontological patterns and archetypes, vocabularies, and common, conceptual theories related to ontologies and their fit to the problem space. However, crafting of whole or even partial common semantic content via logical union, assembly, extension, specialization, integration, alignment, and adaptation has long presented challenges. Achieving commonality and reuse in a timely manner and with manageable resources remain key ingredients for practical development of interoperable ontologies of quality.

Ontologies have a wide range of applications, including semantic integration, decision support, search, annotation, and systems design, as can be seen in the Ontology Usage Framework from Ontology Summit 2011 [3]. A key question to consider is how Big Data and Semantic Web applications fit into this framework -- what is the role of ontologies in these applications, and how is the semantic content being used? There are also two general issues that take more prominence when tackling Big Data and Semantic Web

problems. The first is a characteristic of the ontology, namely, the representation language of the ontology and the tradeoff that exists between the expressiveness of this language and the efficiency of reasoning with the ontology in this language. The second feature, which is a characteristic of the problems encountered in Big Data and the Semantic Web, appears in the novel ways in which ontologies are used on a large scale.

2.1 How are Ontologies Being Used and How Could They be Used?

Within Big Data, semantic integration addresses the variety problem insofar as any software that uses multiple datasets needs to ensure that there are no semantic mismatches. Ontologies can also mitigate variety in Big Data by aiding the annotation of data and its metadata. Data sets will differ in completeness of metadata, granularity and vocabulary used. In this way, ontologies can reduce some of this variety by normalizing terms and providing for absent metadata.

A more recent use of ontologies for data analytics that has potential for high impact is for managing of data provenance. This includes any transformations, analyses, or interpretations of the data that have been performed. Currently, most Big Data projects handle provenance in an ad hoc, rather than systematic manner. Ontologies for describing data provenance do exist, such as the PROV-O ontology [4]. Developing standard ontologies for commonly used, but non-formalized, process models such as the OODA loop [5] and JDL/DFIG fusion models [6] could have a significant impact on data analytics. The KIDS [7] framework is an example of such a formalization. Standard statistical reasoning ontologies are another area that has the potential for having a high impact.

At the global level (e.g., the Web), there are too many domains to have very deep semantics common to all of them. Nevertheless, Schema.org has been tackling the formidable problem of developing a generally accepted vocabulary that is now being used by over five million internet domains, and gradually introducing deeper semantics. Incorporation of ontologies into the Schema.org framework is challenging but has the potential for significant benefits.

It is unlikely that there will be the ability to make Web-wide ontological commitments. Where projects such as Watson (IBM) [8] limit themselves to a few simple taxonomies, other large collaboration efforts may agree on a limited subset of ontologies, such as parts of some molecular biology ontologies, the Gene Ontology [9], and other Open Biological and Biomedical Ontologies (OBO) Foundry ontologies [10]. One question is whether it makes sense, or is feasible, to turn complete collections of Big Data into ontologies. It seems feasible, but is difficult. Manually building ontologies is labor intensive. Mining data for reusable semantic content suffers from the potential inconsistency, incompleteness, and irrelevance of data “out there”. Use of machine learning for harvesting semantic content from Big Data may require further research to enable learning ontologies from Big Data.

The current use of data- and text-mining, statistical, and other analytic techniques on Big Data to discover correlations and patterns can be combined with semantic content that provides some semantic interpretation of those patterns. The associated semantic content will aid greatly in further dissemination of the results, and then in turn, can be correlated and linked into larger, ever-growing semantic patterns -- providing the multi-layered richness of so-called “deep learning”.

2.2 The Role of Expressiveness

The notion of expressiveness refers to the logical properties of an ontology representation language. The Ontology Spectrum characterizes the range of different languages from RDF, OWL, and the Rule Interchange Format (RIF) through to Common Logic and modal logics. A critical question for both ontology users and developers is the selection of the appropriate language and the ability to reason effectively with it. In fact, many of the earlier debates about the nature of ontologies (i.e. what is an ontology?) have their roots in the different expectations that users have for the expressiveness of the underlying representation language.

There is widespread recognition that different applications will require different levels of expressiveness. For applications of ontologies related to decision support systems in which the queries are not known at design time, expressiveness is very important. On the other hand, if the queries are known beforehand, it is often possible to construct a more restrictive ontology that will answer those queries with improved performance.

Multiple axiomatizations of ontologies, in each of the standard ontology languages, will be needed to meet all requirements in a domain. Ontology developers in general recognize this condition, and so, some foundational ontologies such as the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [11] and Basic Formal Ontology (BFO) [12] have first-order logical representations, but also corresponding lighter-weight OWL representations with less restrictive (because less expressive) axiomatizations.

The expressiveness of an ontology representation language is closely related to the requirements for any ontology that is intended for a particular application. RDF, the native language of linked data, goes a long way in Big Data settings, because of the low ontological commitment it enforces, while still allowing the linking to complex descriptions. On the other hand, many traditional applications of ontologies, such as semantic integration and decision support, have required more expressive languages such as the Rule Interchange Format (RIF), Common Logic, and logic programming.

Building lightweight ontologies and vocabularies for Semantic Web and Big Data applications requires new, agile engineering techniques. The recent Linked Open Terms (LOT) [13] approach starts with reuse, taking advantage of the great number of vocabularies that already exist on the Web. Where the terms needed to describe the data at hand cannot be found in existing vocabularies, the knowledge engineer will have to create new ones, but is encouraged to link them to existing ones.

The Watson developers did not build a formal ontology of the world, with which they would try to unify formal logical representations of the questions. Instead, they locally learned ontologies on demand, drawing on formal as well as informal sources, using different reasoning techniques. First, hypotheses are generated. Second, evidence is retrieved for those (approaches include keyword matching against as-is natural language text sources). The challenge is to disambiguate types (e.g., “person” vs. “place”) of entities and predicates. This can be partly solved using existing ontologies and knowledge bases such as YAGO [14].

A swing back to lightweight approaches has also occurred in the field of web services. Generally, a service consumer finds a web service that a service provider has registered in a central registry, and then communicates with the web service in order to execute it. Semantic web service descriptions, in addition to the basic syntactic WSDL description, is required for finding and comparing service providers, for negotiating and contracting services, for composing, enacting and monitoring them, and for mediating heterogeneous

data formats, protocols and processes. Traditionally, the semantics of web services would have been described using heavyweight ontologies such as the Web Service Modeling Language (WSMO) [15] or OWL-S [16] based on expressive ontology languages, and these services would have been assumed to communicate by heavyweight XML messages according to SOAP. As this semantics-first modeling approach was not taken up in practice, and as the majority of web services is nowadays implemented using lightweight REST interfaces, more recent activities are instead promoting more lightweight semantic descriptions of web services: a bottom-up annotation and interlinking approach called “Linked Services”. Linked Services are described with lightweight ontologies mainly using RDFS and a few OWL constructs; e.g., the Linked Unified Service Description Language (USDL) [17], a linked data reimplementation of USDL [18], which itself generalizes generalizes the Web Services Description Language (WSDL) [19].

2.3 Scalability

One aspect in which both Big Data and Semantic Web applications differ from other applications of ontologies is in the scale of the problems which are being addressed. Together with performance constraints, scalability has a profound impact in how the required ontologies are represented and used. The joint demands of volume and velocity necessitate tradeoffs between expressiveness of the ontology language and the efficiency of reasoners for that language. The development of large-scale reasoning techniques should alleviate some of these concerns. Another approach is to use hybrid methods which incorporate the semantic content of an ontology without requiring an explicit axiomatization of the ontology to be used with a reasoning engine. A further approach is to use lightweight ontologies that in turn are linked to heavier-weight ontologies, to enable on-demand (and optional) more precise reasoning over more finely grained semantic content, i.e., putting into pragmatic practice the notion of ontology modularity.

Scalability can refer to the use of ontologies on Big Data sets, but it can also refer to problems in which the ontologies themselves are too large for conventional reasoning systems. Even editing and visualization of large-scale ontologies poses new challenges for existing ontology tools.

2.4 Questions

- What combination of ontology engineering and reasoning techniques will be used for Big Data problems?

- Should one even try to represent large amounts of knowledge using ontologies? Do even lightweight ontologies scale to Big Data? Or would it rather suffice, as use cases in biology suggest, to use ontologies for annotating Big Data with terms?

3. Sharable and Reusable Semantic Content

Reuse of semantic content can be defined as the ability to include content from one source in another, or simply to be inspired by the content in a source. The reuse may directly align with the original intentions of the developers, or may extend in totally unexpected directions. The notion of semantic reuse is very similar to reuse in software engineering. It requires that the concepts (including relationships, axioms and rules),

assumptions, and expression(s) of the included content meet a need, and can be fit into the implementation of the re-user's development activities. Reuse seems to be done for similar reasons across all development-related disciplines: to reduce the development effort and cost (by developing less), to expand the benefit (improve the return on investment) of the original content, and to improve the quality of the original content. Since increased reuse suggests that bugs are identified and eliminated, we have a virtuous cycle, especially when the different uses are diverse and any defects and changes are fully documented and explained.

3.1 What Limits Ontology Reuse?

From its inception, the development of sharable reusable ontologies has been a focus in the field of Applied Ontology. Much effort has gone into developing foundational (upper) ontologies (such as DOLCE and BFO) or creating broad domain models (such as Semantic Web for Earth and Environmental Technology (SWEET) [20]) as a means of enabling reuse. In addition, we currently see a massive proliferation of (sometimes overlapping) vocabularies described in the Linked Open Vocabularies (LOV) ecosystem [21]. Yet the amount of reuse of existing vocabularies and ontologies seems quite low in practice. In this section, we examine several possible reasons for this situation, and determine whether or not they present fundamental obstacles to reuse.

3.1.1 Mismatches and Misunderstandings

One potential reason for little reuse is that the required ontologies simply do not exist, that is, the ontologies that have been designed do not satisfy the needs of users with new applications. Determining whether or not an existing ontology meets the needs of a user leads to the discussion of the ontology lifecycle -- the topic of Ontology Summit 2013 [22], in which ontologies were considered as engineered artifacts in the context of requirements developments, ontology analysis, design, evaluation, and deployment. In particular, users need to understand how the requirements for an ontology can be captured using techniques such as competency questions. There are many opportunities for reuse, but a domain and its competency questions must be understood first. Often, reuse fails because it is attempted before the requirements, underlying concepts, and assumptions (driving the creation of the content) are fully understood. In this case, there is a perceived, rather than a real mismatch -- there may be ontologies that can be reused, but users do not recognize that the existing ontologies do in fact meet their needs.

Ontologies that do exist may themselves not be designed for reuse, and may be implemented in ways that make reuse difficult (e.g., mismatches between actual generality/specificity of concepts and their labels and names). What is appropriate for a specific application may be more or less specific to the way in which one intends to re-use the concepts. Labels, in particular, may cause misunderstandings since the developer of an ontology may have used a very general label for a concept that is framed in a way that is very specific to the application context.

3.1.2 Finding Mr. Right Ontology

Another possibility is that the ontologies exist but are difficult to find. Where can users find this content? Efforts such as LOV and the Open Ontology Repository (OOR) [23] are beginning to address this issue. Of course, more than a simple registry of ontologies is needed -- there must also be ways of organizing and

annotating the ontologies with the appropriate metadata so that users can find the ontologies that match their requirements (as discussed in the preceding section). In addition to notions such as provenance (captured by efforts such as Ontology Metadata Vocabulary (OMV) [24], such metadata will also need to include a wider range of features. From the development perspective, metadata should include the competency questions, ontological commitments and design decisions which were used in the development of the ontology, and existing mappings and alignments with other ontologies. From an implementation perspective, features should include supported reasoning, languages, rules, and conformance to external standards, systems or applications in which it has been used. Finally, from an engineering perspective, it is important to include information about evaluations which have been performed on the ontology. In this way, ontology metadata can help guide useful selection of what to reuse from the supply of ontologies available in repositories.

Even when a potential ontology has been found for reuse, issues of evaluation, verification, quality, and trust arise. Reusing an ontology simply because it uses a particular set of keywords for its concepts will undoubtedly lead to problems.

3.1.3 This Ontology Doesn't Fit ...

Like Goldilocks and the Three Bears, perhaps appropriate ontologies exist, but they have issues that prevent them from being reused for particular efforts. In some cases, the ontologies that do exist are themselves not designed for reuse, and may be implemented in ways that make reuse difficult, including insufficient semantic explicitness and mismatches between actual generality/specificity of concepts and their labels and names.

An ontology may be incomplete, that is, it may not satisfy all the requirements for a particular application. Existing ontologies are usually insufficient for a new domain or application and must be extended or modified in some fashion. In this regard, it is important to remember the role of competency questions in the selection of what to reuse. If users are able to match their competency questions with the competency questions supported by the existing ontologies, they can better determine how the ontologies can be reused or extended to satisfy all of the requirements.

Finally, an ontology may not be in the knowledge representation language that a user needs, so that even if the ontology meets all the requirements as captured by competency questions, it may not meet the additional requirements that arise from the use of the ontology in the overall system design and deployment, and operation. In this case, it is important to recognize that reuse of an ontology can occur across multiple representation languages. For example, given an ontology in an expressive language such as Common Logic, we can specify less expressive versions, or fragments of the ontology in other representation languages, such as RIF, OWL, and RDF. Each of these fragments can then be reused by a wider variety of applications. In particular, applications on the (big) Web of Data can profit from using lightweight ontologies and methods. These lightweight definitions can provide focused ontological commitment, and still afford the benefits of adequate semantics while supporting reasoning for their intended usage. The idea is to find and reuse ontology parts that are appropriately expressive.

3.1.4 Modularity

In many cases, the user only needs parts of the ontology, and this leads to the problem of supporting partial

reuse. An obvious approach to this problem is modularity, but the modularization of existing ontologies itself remains a difficult problem. The assembly, extension, specialization, integration, alignment, and adaptation of small modular ontologies needs to become part of the ontology development methodology. Approaches that support the specification of relationships between the modules of an ontology, such as OntoIOp [25], address these issues. Ontology repositories may also be able to provide more explicit support for the modularization of ontologies as they are uploaded. Ontology development, editing, and browsing tools can then support modularity by better enabling effective views of, and work with, collections of ontology modules.

3.1.5 Integration

Reuse usually requires integration of multiple ontologies, and the integration problem can be just as difficult as developing a new ontology. A key technique is the creation of "integrating" modules that merge the semantics of the reused components.

Ontology mapping plays a key role in reuse when there are multiple ontologies that can potentially be reused. Understanding how different ontologies in the same domain (e.g., multiple time, units, or process ontologies) are related to each other is an essential part of determining whether or not one ontology can be integrated with others, even in cases where the terminologies are not the same.

Integration arises most acutely in the variety problem with Big Data, where ontologies can tackle variety by aiding the annotation of data and metadata. Data sets usually differ in completeness of their metadata, granularity, and terminology used. Ontologies can reduce some of this variety by normalizing terms and providing absent metadata. An additional problem in many Big Data applications is that terminology used at one time for one set of data might have a different interpretation from another dataset that appears to use the same terminology but which is used at a different time. For ontologies to deal with this effectively, they must not only evolve over time but also map terms and previous interpretations to the new ones. Ontologies have the potential to greatly ease this problem, by providing a standard model, independent of particular data representations and terminologies, to which those various representations and terminologies can be mapped

3.1.6 Just Do It Yourself

It may be easier to design a new ontology for an application rather than spend time to find possible ontologies for reuse and then to understand them sufficiently well enough to determine whether or not they satisfy the user's requirements. If this is indeed the case, then it will be important to create new ontology development environments that better support design for reuse.

Ontology design patterns are an approach that can be used to directly incorporate reuse into the ontology development methodology. By explicitly capturing the reusable aspects of an ontology, a design pattern allows the designer to more effectively specify the commonalities among otherwise disparate components.

There may also be situations in which weaker forms of reuse are more appropriate. For example, in “reuse by inspiration”, the terms, relations, or axioms of a particular ontology are not explicitly reused, but they serve to guide the designer of a new ontology with respect to the design decisions that need to be made. In this approach, ontology modification becomes a technique for ontology design.

Many times there are barriers and bottlenecks to the use of ontologies, both in terms of reuse of existing content or in developing new content. These barriers and bottlenecks can be due to a multiplicity of factors including:

- the cost of development and deployment of ontologies,
- inadequate understanding of the uses of ontologies,
- the timeliness of being able to deliver solutions,
- incomplete knowledge about or skills in ontological engineering on the part of the ontology developers,
- a mismatch between the application requirements and the intended domain coverage and reasoning requirements of the ontologies,
- the use of inadequate tools at different stages of the ontology development lifecycle,
- sociological, cultural, and motivational issues involving the stakeholders, application developers, domain experts, and ontologists.

Realistically, all of the above factor into the cost of development and deployment of ontologies, and so reuse of existing semantic content is the potential cost-saver. However, non-ontological solutions are often done faster and cheaper as one-offs using other technology, because the value proposition of ontology reuse -- vastly cheaper development and maintenance costs amortized over multiple ontology application lifecycles -- is not understood nor communicated to, and thus not understood by the supporting community.

3.1.7 Social Factors

Many ontologies intended for reuse are designed in English and it is assumed all users will use English; however, this is not a valid assumption in many applications. Although it is pragmatic that identifiers should be in the language of the developer (since this helps the development and debugging process), identifiers should be hidden from end users, who should be able to choose the language for the labels they see. This can be even more problematic when the intended semantics of concepts in the ontology are primarily specified in the documentation instead of being formally captured in the axiomatization of the ontology. In any case, the use of both vocabularies (terms) and ontologies (concepts) that are linked together enable language-specific terms to be mapped to their logical concepts.

3.2 Where Reuse Happens

Despite the dour nature of the previous section, we have examples of successes with sharing and reusing vocabularies and ontologies. For example, consider Schema.org. It defines a widely used (and extensible) vocabulary for describing the contents of a web page. The concepts contained in Schema.org are thoroughly documented, as well as how to use and extend the vocabulary. In addition, users are supported via blogs and discussion groups. The approach taken in developing Schema.org addresses the issues of finding reusable content, managing the size and complexity of the content, integrating the various pieces and extensions together, and maintaining quality and trust. All of these are important issues that were raised in the previous section.

Other examples of successful reuse are based on small ontologies and design patterns. These can be generally applicable or specific to a domain. Examples of both general and specific patterns can be found at Ontology Design Patterns (ODP) [26], while the OceanLink [27] project (within the NSF's EarthCube

initiative) is defining more domain-specific patterns. The goal is to succinctly capture basic concepts, such as collections, lists, events, or in the case of OceanLink, the trajectory of a cruise ship.

Because the concepts are common, they can be easily understood and integrated into ontology development activities. In addition, they can be mapped to data in disjoint, disconnected repositories, and used to integrate that data.

3.3 Best Practices

What can we learn, then, from both our successes and failures? The following bullets summarize some of the key best practices.

- Wise reuse possibilities follow from knowing the project requirements. Competency questions should be used to formulate and structure the ontology requirements, as part of an agile approach. The questions help contextualize and frame areas of potential content reuse.
- Be tactical in formalization. Reuse content based on your needs, represent it in a way that meets your objectives, and then consider how it might be improved and reused. Clearly document your objectives so that others understand why you made the choices that you did.
- Small ontology design patterns provide more possibilities for reuse because they have low barriers for creation and potential applicability, and offer greater focus and cohesiveness. They are likely less dependent on the original context in which they were developed.
- Use "integrating" modules to merge the semantics of reused, individual content and design patterns.
- Separately consider the reuse of classes/concepts, from properties, from individuals and from axioms. By separating these semantics (whether for linked data or ontologies) and allowing their specific reuse, it is easier to target specific content and reduce the amount of transformation and cleaning that is necessary.
- RDF provides a basis for semantic extension (for example, by OWL and RIF). But, RDF triples without these extensions may be underspecified bits of knowledge. They can help with the vocabulary aspects of work, but formalization with languages like OWL can more formally define and constrain meaning. This allows intended queries to be answerable and supports reasoning.
- Better metadata (providing definitions, history and any available mapping documentation) for ontologies and schemas is needed to facilitate reuse. Also, it is valuable to distinguish constraints or concepts that are definitive (mandatory to capture the semantics of the content) versus ones that are specific to a domain. Domain-specific usage, and "how-to" details for use in reasoning applications or data analytics are also valuable. Some work in this area, such as Linked Open Vocabularies and several efforts in the Summit's Hackathon, is underway and should be supported.
- Better ontology and schema management is needed. Governance needs a process and that process needs to be enforced in tooling. The process should include open consideration, comment, revision and acceptance of revisions by a community.
- The explicit specification of ontology fragments should be incorporated into development methodologies in the ontology lifecycle.

4. Automation and Tools

The Web of Data (Semantic Web, Linked Data, and Big Data) provides great opportunities for ontology-based services, but also poses challenges for tools for editing, using, and reasoning with ontologies, as well as techniques that address bottlenecks for the engineering of large-scale ontologies. It is sensible to start with lightweight tools, but large complex ontologies cannot be managed with such tools. Inferencing tools can help with logical consistency, but there are many more errors that can be made beyond logical consistency, and tool support that can identify and resolve such errors is still in its infancy.

4.1 Automated Ontology Acquisition

The Holy Grail in the use of ontologies is the acquisition of ontologies by automated means. This is a very complex task because it tries to capture and represent the semantics that human beings possess, from arbitrary or sometimes domain-specific data. Ontology extraction and automated acquisition is still in its infancy and requires much more robust machine learning (sometimes termed “deep learning”) than exists today. Current state of the art in automated ontology acquisition typically consists of using existing machine-learning, text-analytic, and natural language processing techniques (and often all three) on annotated or un-annotated data to provide candidate ontology classes, relations, and properties to a human being, who often adjudicates the candidates.

Notwithstanding the above paragraph, information extraction certainly feeds ontology-provisioning of the semantics of data, especially for unstructured data, but in turn can be greatly assisted by existing ontologies, with the result that data becomes semantically annotated or indexed and thus accessible to semantic search and navigation -- with the resulting ontology-described triples of Linked Data and the Semantic Web able to be added to triple stores to more directly facilitate reasoning over the data. At Internet scale, navigation, search and discovery (via free-text search or querying using SPARQL, for example), and aggregated semantics may reasonably be provided. Automated reasoning (deductive, inductive, abductive, and probabilistic) at scale over the data using ontologies can be partitioned, distributed, and parallelized but may require special tools (such as ontology registries with services, and more specialized hardware) and longer time-scales.

4.2 Tools for Engineering Large-Scale Ontologies

The tools needed for engineering large-scale ontologies and supporting the semantic enrichment of Big Data at Internet scale range from distributed collaborative ontology development and maintenance tools (an example is WebProtege), connected islands of ontology repositories (such as the Open Ontology Repository from Ontology Summit 2008 [28], BioPortal [29], etc.) and the services provided by those, to more modular ontology architectures, and distributed and parallel reasoning and ontology/vocabulary mapping services. Along with promulgating increased knowledge about ontologies and semantic technologies and their value proposition especially to development and stakeholding communities, such tools are needed to help overcome the recognized barriers and bottlenecks described in the previous sections.

4.2.1 Modular Ontology Architectures

In recent years, more modular ontology architectures and their supporting tools have emerged, at least as research threads and prototypes (e.g., Workshop on Modular Ontologies (WoMO) [30]). Because there are potentially multiple levels of granularity needed for large-scale ontology use, tools and practices that support modularity and granularity are needed.

4.2.2 Ontology Reasoning Tools

Semantic Web and Linked Data technologies are focused on providing semantic enrichment of data on the Internet, and use ontologies in multiple ways to support that. In many cases various kinds of ontology and rule reasoning are needed, ranging from classificational reasoning, to consistency checking of ontologies and triple representations that constitute knowledge base instances, to simple inference (e.g., materializing transitivity assertions) and SPARQL query aggregation and optimization, to more complex inference requiring finely discriminating rules for decision support and similar applications. For more complex reasoning, often hybrid reasoning tools are necessary, e.g., tools that support both description logic and first-order logic reasoning, and both logical and probabilistic reasoning (e.g., see the Ontolog Forum mini-series “Ontology, Rules, and Logic Programming for Reasoning and Applications” [31]).

4.2.3 Ontology and Vocabulary Mapping and Alignment Tools

Large-scale use of ontologies for the Internet and Big Data also require the use of tools to support ontology and vocabulary mapping and alignment. As mentioned previously, users and developers need to (naturally) use their own natural languages to both develop and use ontologies. In many cases, the same ontologies will have to be mapped to multiple vocabularies (represented, for example, in SKOS), possibly each in distinct natural languages or used by distinct communities. In addition, distinct ontologies, or modules of ontologies, will have to be mapped to other ontologies or otherwise aligned, to provide scalable semantics. Tools and services to support vocabulary-to-ontology and ontology-to-ontology mapping are needed (see: Workshop on Ontology Matching (OM-2013) [32]).

4.2.4 Ontological-Analytical Techniques and Hybrid Tools

Big Data present special requirements for tool support, because many of the analytical tools that work on large-scale data use statistical and probabilistic text-analytic methods and massive machine-learning, or hybrid algorithmic methods (e.g., IBM Watson). These methods must be combined with logical and ontological methods in reasonable fashion to make sense of the Big Data and to disseminate that sense to users and applications that provide decision support. Cloud and Grid architectures and infrastructure are often required to find significant correlations and patterns in the Big Data, which ontologies can be used to describe and enrich. However, in many cases, simple parallel architectures and computing resources are not sufficient for combining large amounts of data with the graph-structured representations that ontologies use. So more specialized hardware may be needed (e.g., Cray YarcData Urika graph machines [33]).

4.3 Questions

Among the questions that the Ontology Summit brought forward concerning automation and tools for ontologies are the following:

- Which ontology tools are needed and when are they needed?

- Can ontology acquisition, development, integration, and reuse be automated more?

5. Conclusions and Recommendations

Hector Levesque gave an invited talk last year at the IJCAI-13 conference in Beijing to the artificial intelligence community, and his concluding words may have bearing to our community [34]:

"We should avoid being overly swayed by what appears to be the most promising approach of the day. As a field, I believe that we tend to suffer from what might be called serial silver bulletism, defined as follows: the tendency to believe in a silver bullet for AI, coupled with the belief that previous beliefs about silver bullets were hopelessly naive."

5.1 Recommendations

1. Efforts to identify the values of ontologies within Big Data applications are of the highest priority, as gaps between the Big Data and Applied Ontology communities still exist. We should seek more opportunities to encourage cross-community interaction.
2. The community should converge and adopt best practices for sharable and reusable content.
3. Semantic Web and Big Data developers need to identify the ontology features that matter to them, i.e., those which they need in an ontology or which they need to know about an ontology when considering for reuse.
4. Ontology developers and providers should consider the above features and attempt to: (a) design and/or refactor their ontologies and methodologies with these needs in mind, when possible, and (b) provide metadata about their ontologies that indicates their status with respect to these needs.
5. The community should adopt the definition of standard metadata for reuse -- documentation of assumptions, requirements, scope, intent, use cases, history. Ontology repositories and other tools should support this metadata, and the addition of more applicable metadata by re-users and evaluators.
6. Tools should be developed to better support modular ontology development, integration, and reuse.
7. A wider array of functionalities should be incorporated into tools, including support for designing, publishing, finding, understanding, visualizing, verifying, maintaining, translating, integrating ontologies on the web.

5.2 Challenge Problems

We can also pose a number of challenge problems which can serve to focus and guide future collaboration among the three communities of Applied Ontology, Semantic Web (and Linked Data), and Big Data.

- What ontologies are required by Semantic Web and Big Data applications?

- What are the requirements for tools, services, and techniques that support ontology development within Semantic Web and Big Data applications?

- Is scalability the fundamental challenge for using ontologies on the Web?

- Is the design and application of ontologies on the Web fundamentally different from existing

techniques?

- What is the role of crowd-sourcing in ontology design?
- What are the requirements for the tools, services, techniques used for designing and implementing semantic content on the Semantic Web and in Big Data applications?
- Are we encountering new ontology engineering bottlenecks in Semantic Web and Big Data applications?
- Can the variety problem in Big Data applications be addressed using existing techniques for semantic integration, such as ontology mappings?
- What benchmark data sets can be used to guide future work in the integration of ontologies?

References

[1] Ontology Summit 2014 Recommended Readings and Ontology Repository. http://ontolog.cim3.net/OntologySummit/2014/readings.html.
[2] Ontology Summit 2014. http://ontolog.cim3.net/OntologySummit/2014/.
[3] Ontology Summit 2011: Making the Case for Ontology. http://ontolog.cim3.net/cgi-bin/wiki.pl?OntologySummit2011.
[4] PROV-O, Provenance Working Group. http://www.w3.org/TR/prov-overview/.
[5] OODA Loop. http://en.wikipedia.org/wiki/OODA_loop.
[6] Joint Directors of Laboratories (JDL) / Data Fusion Information Group (DFIG). http://en.wikipedia.org/wiki/Data_fusion#The_JDL.2FDFIG_model.
[7] Chan, Eric. 2014. Enabling Enhanced OODA Loop with Modern Information Technology. Ontology Summit 2014 presentation. http://ontolog.cim3.net/cgi-bin/wiki.pl?ConferenceCall_2014_02_13#nid466S.
[8] IBM's Watson. http://www.ibm.com/smarterplanet/us/en/ibmwatson/.
[9] Gene Ontology. http://www.geneontology.org/.
[10] Open Biological and Biomedical Ontologies (OBO) Foundry. http://www.obofoundry.org/.
[11] Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE). http://www.loa.istc.cnr.it/old/DOLCE.html.
[12] Basic Formal Ontology (BFO). http://www.ifomis.org/bfo/. Also: http://ncorwiki.buffalo.edu/index.php/Basic_Formal_Ontology_2.0.
[13] Linked Open Terms (LOT). http://lot.linkeddata.es/.
[14] YAGO. http://www.mpi-inf.mpg.de/yago-naga/yago/.
[15] Web Service Modeling Ontology (WSMO). http://www.wsmo.org/.
[16] OWL-S: Semantic Markup for Web Services. http://www.w3.org/Submission/OWL-S/.
[17] Linked Unified Service Description Language (Linked USDL). http://www.linked-usdl.org/.
[18] Unified Service Description Language (USDL). http://www.internet-of-services.com/index.php?id=288&L=0.
[19] Web Services Description Language (WSDL). http://www.w3.org/TR/wsdl.
[20] Semantic Web for Earth and Environmental Technology (SWEET). http://sweet.jpl.nasa.gov/.
[21] Linked Open Vocabularies. http://lov.okfn.org/dataset/lov/.

[22] Ontology Summit 2013: Ontology Evaluation Across the Ontology Lifecycle. http://ontolog.cim3.net/OntologySummit/2013/.
[23] Open Ontology Repository (OOR). http://oor.net.
[24] Open Metadata Vocabulary (OMV). http://omv2.sourceforge.net/.
[25] Ontology Integration and Interoperability (OntoIOp). http://ontoiop.org.
[26] Ontology Design Patterns (ODP). http://OntologyDesignPatterns.org.
[27] EarthCube OceanLink. http://workspace.earthcube.org/oceanlink.
[28] Ontology Summit 2008: Toward an Open Ontology Repository: http://ontolog.cim3.net/cgi-bin/wiki.pl?OntologySummit2008).
[29] National Center for Biomedical Ontology (NCBO) BioPortal. http://www.bioontology.org/BioPortal.
[30] Workshop on Modular Ontologies (WoMO): http://womo2014.bio-lark.org/.
[31] Ontology, Rules, and Logic Programming for Reasoning and Applications, Ontolog Forum mini-series. http://ontolog.cim3.net/cgi-bin/wiki.pl?RulesReasoningLP.
[32] Workshop on Ontology Matching (OM-2013): http://om2013.ontologymatching.org/.
[33] Cray YarcData Urika. http://www.cray.com/Products/BigData/uRiKA.aspx.
[34] Levesque, H. 2014. Artificial Intelligence, Volume 212, July 2014, Pages 27–35. http://dx.doi.org/10.1016/j.artint.2014.03.007.