

# Position Paper: Ontology Construction from Online Ontologies

Harith Alani  
Intelligence, Agents, Multimedia  
Electronics and Computer Science Dept.  
University of Southampton  
Southampton, UK  
h.alani@ecs.soton.ac.uk

## ABSTRACT

One of the main hurdles towards a wide endorsement of ontologies is the high cost of constructing them. Reuse of existing ontologies offers a much cheaper alternative than building new ones from scratch, yet tools to support such reuse are still in their infancy. However, more ontologies are becoming available on the web, and online libraries for storing and indexing ontologies are increasing in number and demand. Search engines have also started to appear, to facilitate search and retrieval of online ontologies. This paper presents a fresh view on constructing ontologies automatically, by identifying, ranking, and merging fragments of online ontologies.

## Categories and Subject Descriptors

I.2.4 [Knowledge Representation Formalisms and Methods]: Representation languages, Semantic networks; H.3.3 [Information Search and Retrieval]: Search process, Selection process

## Keywords

Automatic ontology construction, ontology reuse

## 1. INTRODUCTION

Based on current semantic web technology, ontologies are complex to build and understand, which hinders their wide adoption [9]. Constructing ontologies automatically has been the focus of some research to help reduce the high cost of building ontologies by hand. Several approaches have been investigated for extracting ontologies from existing knowledge bases (e.g. [24]), from legacy software systems (e.g. [30]), and from text corpora (e.g. [13, 5]).

One of the problems with such approaches is that background knowledge is not usually explicitly expressed in the knowledge sources [5]. This could seriously limit the ontologies produced with such approaches and may enforce searching for external knowledge sources to fill such knowledge gaps.

An alternative approach, or perhaps a complimentary one to the approaches above, is to search and reuse ontologies that already exist on the web when constructing new ones. It is often believed that one of the major factors for the initial rapid growth of the web was the ability and ease for users

Copyright is held by the author/owner(s).

WWW2006, May 22–26, 2006, Edinburgh, UK.

to copy HTML code from existing web pages and reuse it to create their own pages. Users were able to create web pages with little effort, and without the need for deep understanding of the underlying languages. Such ease of reuse needs to be replicated with ontologies to speed up their spread and use [21].

Facilitating reuse of other people's ontologies should encourage more individuals and organisations to participate in the semantic web. After all, ontologies are meant to provide an "easy to reuse library of class objects for modelling problems and domains" [27]. Wide dissemination of ontology technology would require methods and tools to facilitate ontology reuse [7]. However, no such tools are currently available, which renders ontology reuse a very hardship task [4, 12].

Harnessing online ontologies might be the first step towards achieving true reuse. There is an increasing number of online libraries for searching and downloading ontologies. Examples of such libraries include Ontolingua<sup>1</sup>, Protégé<sup>2</sup>, and DAML<sup>3</sup>. Few search engines have recently appeared that allow keyword-based search for online ontologies, such as Swoogle [6] and OntoSearch [31]. Swoogle currently contains over 4000 ontologies covering a wide range of domains<sup>4</sup>. However, the use of such tools is limited to giving back a list of potentially relevant ontologies, with no further support for reuse.

Providing support for reuse during ontology development from specific ontology libraries has been studied before (e.g. [9, 12]). However, the focus was mainly to enable users to reuse or import whole ontologies or ontology modules. They provided no support for ranking available ontologies, or for extracting and merging the ontology parts of interest, or for evaluating the resulting ontology.

This paper outlines a proposal for a new approach for automatic ontology construction. The approach is meant to encourage and support reuse of existing ontologies, using latest semantic web research technology. We propose to build a system that searches online ontologies for representations of certain concepts, ranks the retrieved ontologies according to some criteria, then extract the relevant parts of the top ranked ontologies, and merge those parts to acquire the richest domain representation as possible.

<sup>1</sup><http://www-ksl-svc.stanford.edu:5915/>

<sup>2</sup><http://protege.stanford.edu/pugins/owl/owl-library/>

<sup>3</sup><http://www.daml.org/ontologies>

<sup>4</sup><http://swoogle.umbc.edu/>

We believe that by relying on existing ontologies we can avoid the problems stated in [5], and bootstrap the process of ontology building by reusing existing ontology fragments.

## 2. SCENARIO

To help explaining how our suggested system might work, consider the following scenario.

Imagine there is a knowledge engineer who is in need of an ontology representing the academic domain. The ontology is to be used for creating a knowledge-base to hold information on staff, projects, conferences, publications, etc. There are many ontologies online that covers various portions of this domain. It would certainly be beneficial if the engineer can quickly and efficiently reuse some of these third-party ontologies, to at least bootstrap the ontology construction task.

One of the recommended first steps towards building an ontology is to write down a list of terms to represent in the ontology [23, 27, 17]. This helps scoping the domain, reaching agreement, and building the class hierarchy. Lets assume that one of the terms that our knowledge engineer wrote was “Conference”. There could be many ontologies out there that covers this concept to some extent, that our engineer is not aware of. It might speed up his task if some existing representations can be easily gathered and represented to him to accept, modify, or at least learn from.

At the time of writing, when **searching for owl ontologies** on the term “Conference” in Swoogle [6], a list of 34 ontologies was returned. Our system could analyse these ontologies to acquire as much representational knowledge as possible about the given term. In some cases, if there are too many ontologies to analyse (such as in our example) then the system could greatly benefit from an **ontology ranking** service that can order the ontologies according to some criteria. The system could be set to only analyse and reuse, say, the top five ontologies from the ranked list, but of course such threshold can be modified based on what has been found.

Now the system could commence its analyses of the *top ranked* ontologies, starting with rank number 1. Based on the size and scope of that ontology, the system may decide to take the ontology as a whole, or only take the section that describes “Conference”. To achieve the latter, the system will need to use an **ontology segmentation** service to extract the required section from the ontology. In our case, the first ontology on Swoogle’s results list is *conference.owl*<sup>5</sup>. This ontology comprises of only one class “Conference” and a number of datatype properties, such as *title*, *paperDueOn*, *startDate*, etc. (figure 2).

Even though the above owl file does represent the concept “Conference”, but it might not be sufficient, in terms of coverage, detail, etc. However, many other ontologies were found by the search engine when searching for this concept. Our system could start analysing some of those ontologies to enrich our first ontology with more knowledge. For example, the second ontology found by Swoogle was *web04photo.owl*<sup>6</sup>. A section of this ontology is shown in figure 2. The class “Conference” in this ontology has more detail than in *conference.owl*. For example, we now have a superclass (*Event*), some siblings (e.g. *Workshop*, *Presentation*, etc.), and new

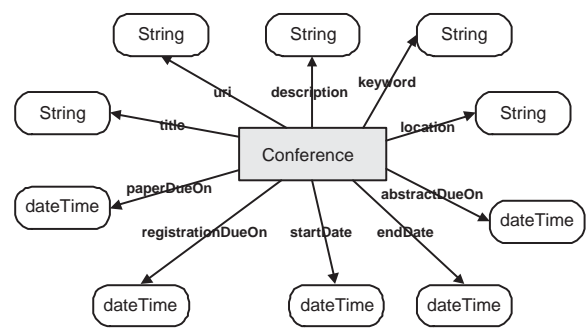


Figure 1: The whole ontology of conference.owl

properties (e.g. *subEventOf*, *hasSubEvent*, *hasOrganizer*, etc.). However, note that this ontology does not have all the properties found in *conference.owl*.

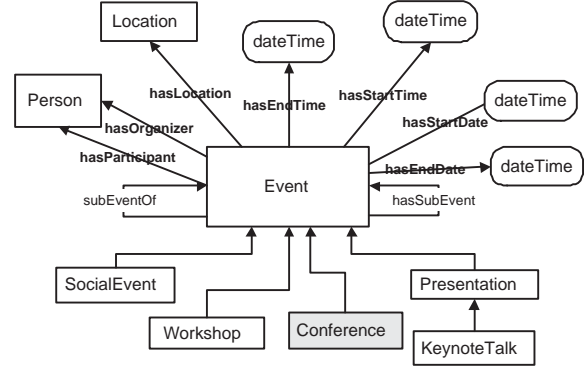


Figure 2: www04photo.owl

Next, the system will need to **compare the two ontologies** (or segments of ontologies) to find any additional representations that can be **merged** into the first ontology. This may result in the creation of new classes and properties in the first ontology to cover these additional knowledge representations.

By iterating the above comparison and merging processes for some of the top ranked ontologies, an ontology might be produced that is perhaps richer and more detailed than any of the already existing ontologies. When the process ends, the resulting ontology can be presented to the knowledge engineer to study and modify or even correct as required.

## 3. ARCHITECTURE

The system we are planning to build combines many of the hot topics in current semantic web research, such as ontology ranking [1, 6, 20], segmentation [25, 14, 22], mapping and merging [11, 8], and automatic evaluation [28]. Figure 3 shows the envisaged architecture of this system. The processes required by this system are briefly described below.

### 3.1 System Processes

The process of reusing online ontologies to construct new ontologies should include the following steps:

1. **Search for Ontologies:** First step is to find some potentially relevant ontologies to reuse. This may be

<sup>5</sup><http://ebiquity.umbc.edu/ontology/conference.owl>

<sup>6</sup><http://www.mindswap.org/~golbeck/web/www04photo.owl>

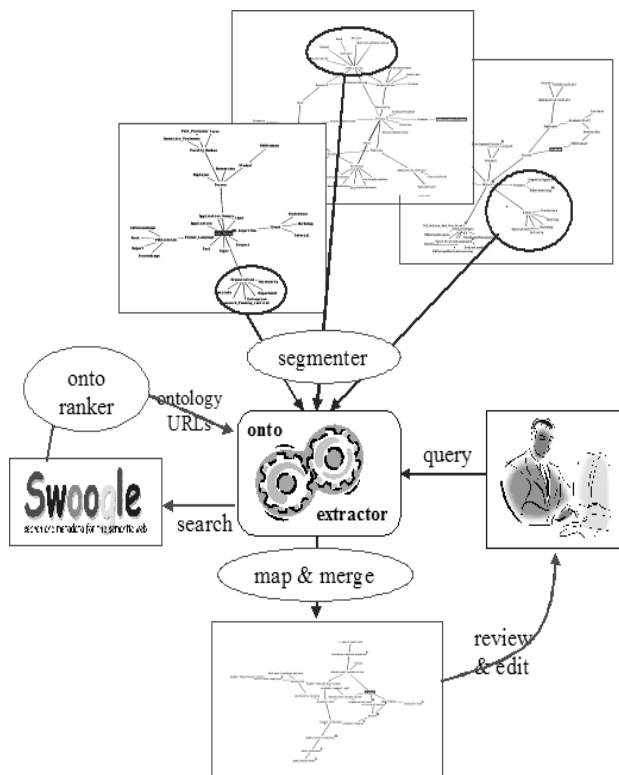


Figure 3: System architecture.

done by searching for specific keywords (as in Swoogle[6]) or perhaps using a more complex query mechanism (eg metadata search [12], structure-based queries, query expansion techniques, etc). This process should produce a list of ontology URLs that are potentially relevant to the user’s needs.

2. **Ontology Ranking:** As shown in section 2, a search for ontologies may return too many hits. To enable the system to start with the *best* ontology, the returned list of ontologies must be ranked according to, for example, how well they represent the concepts in the search query [1], how the ontologies have been rated by users [26], or how well they meet the requirements of certain evaluation tests (e.g. OntoClean [10]). The outcome of this process will be a ranked list of ontologies for the system to analyse and reuse.

3. **Ontology Segmentation:** Depending on the size of the ontology that is being analysed for reuse, and on the desired scope for the final ontology, the system may deploy some tools to *extract* only certain parts of the ontology in hand. This could be necessary to limit the size and domain of the resulting ontology to only what the user is interested in. Users can be given some control on the level of generosity of the segmentation methods.

Several approaches for segmenting ontologies have been investigated and covered in the literature. The type of segmentation required by our proposed system could be very simplistic, such as the spreading activation-based extraction discussed in [19], which cuts an ontol-

ogy view to a fixed graph length around a selected concept. More complex approaches that take the ontology structure into account can also be deployed, such as those discussed in [3, 22]. Other approaches on ontology segmentation include the use of classical clustering algorithms to break an ontology into a fixed number of parts [25], the use of queries to create specific ontology views [29, 14], and the segmentation of an ontology based on analysing its applications’ queries [2].

4. **Ontology Mapping and Merging:** Once ontology segments are obtained, the system will need to compare and merge these segments to gather more domain representation for the concepts in question. This needs to be handled carefully to avoid overwhelming the user with oversized representations.

Ontology mapping and merging are hot topics in AI research, aiming to bridge the communication gap between applications that rely on different, but overlapping, knowledge representations. A number of tools have been developed to map and/or merge ontologies semi-automatically, where the system compares the given ontologies and reports back to the user some mapping suggestions. Examples of such systems are the PROMPT Suite [18] which is integrated into the Protégé ontology editor [16] and Chimeara [15], which was built on Ontolingua [9].

5. **Ontology Evaluation:** The system may need to evaluate the ontology that it constructed from fragments taken from other ontologies, to make sure that the resulting ontology meets some minimum quality checks. The evaluation process may help the system to, for example, resolve some inconsistencies, identify semantic gaps which needs to be filled, etc. OntoClean is a well known methodology for evaluating ontologies based on meta-property tags [10]. The tagging process can be too costly if performed manually, but some attempts to automate it are already under way [28].

The user may want to repeat the steps above to search for additional concepts, or to rerun the process using different thresholds (e.g. analyse more ontologies, use larger segments) then merge the resulting ontology with the one that the system produced so far.

## 4. CHALLENGES

The system described in this paper will definitely be a challenge to build and run successfully. Most of the processes this system builds on (section 3.1) are dependent on semantic web technologies that are still rather immature and far from perfect. However, technologies can only mature once they are applied to practical and real world tests, which is what this system is hoping to provide. Bringing all these technologies together and integrating them to form a single production line, is one of the indirect aims of this proposed system.

Apart from the well known challenges of searching, ranking, segmentation, merging, and evaluation of ontologies, which are well covered in the literature, there are other fundamental issues that the proposed system may need to deal with.

One of the crucial factors when building new ontologies from existing ones is obviously the availability of ontologies

to reuse, in terms of numbers and domain variety. Many of the ontologies constructed by semantic web researchers and developers are never put on the web. This will hopefully change once ontology search engines become more popular, and the benefits of making ontologies available for others become more apparent.

There is also the danger of ending up with a very large and messy ontology as a result of using automated systems, like the one we are suggesting, to construct the ontology. This could happen if many large ontologies are used as a resource. Users might find it hard to clean or modify the resulting ontology if its too large, rendering the whole process less useful. However, this problem can be avoided to some extent using system cut-off thresholds, which can be set based on the size of the ontology to be constructed, where the system stops augmenting the ontology once it reaches a certain size limit. Another possible approach is to enable users to interact with the system throughout the whole process. Users could monitor the construction and augmentation of the ontology, and either permanently stop the system if they are satisfied with the result, or pause the system while they clean the ontology before allowing the system to analyse and merge further ontologies.

Because the system is reusing existing ontologies, the quality of those ontologies will certainly affect the quality of the output ontology. Users might want to restrict the system to only those ontologies that pass certain quality tests, or are provided by specific organisations or authors. Applying proper ontology ranking and evaluation processes might help reducing this problem. However, it can not be guaranteed that the segments extracted from the original ontologies will retain the quality and consistency of their source.

## 5. CONCLUSIONS AND FUTURE WORK

This paper presented a new approach for automatic construction of ontologies. The idea is based on reusing the increasing number of online ontologies to build new ontologies, rather than the current costly habit of starting from scratch. The proposed system intends to make use of a number of technologies to complete its task, such as ontology searching, ranking, segmentation, mapping, merging, and evaluation. To the best of our knowledge, this will be the first time that these technologies are put together to achieve a common goal.

Users of the proposed system will be expected to modify, delete from, and add to the automatically built ontology as they see fit. The purpose here is to avoid reinventing the wheel by provide users with a tool to help them gather and learn from existing domain knowledge representations, thus bootstrapping their ontology construction task.

We aim to start building the system described in this paper soon and experiment with it to evaluate how well such an approach may work in real life scenarios.

## 6. ACKNOWLEDGMENTS

This work is supported under the Advanced Knowledge Technologies (AKT) Interdisciplinary Research Collaboration (IRC), which is sponsored by the UK Engineering and Physical Sciences Research Council under grant number GR/N15764/01. The AKT IRC comprises the Universities of Aberdeen, Edinburgh, Sheffield, Southampton and the Open University. The views and conclusions contained herein are

those of the authors and should not be interpreted as necessarily representing official policies or endorsements, either express or implied, of the EPSRC or any other member of the AKT IRC.

## 7. REFERENCES

- [1] H. Alani and C. Brewster. Ontology ranking based on the analysis of concept structures. In *3rd Int. Conf. Knowledge Capture (K-Cap)*, pages 51–58, Banff, Canada, 2005.
- [2] H. Alani, S. Harris, and B. O’Neil. Ontology winnowing: A case study on the akt reference ontology. In *Proc. Int. Conf. on Intelligent Agents, Web Technology and Internet Commerce (IAWTIC’2005)*, Vienna, Austria, 2005. IEEE.
- [3] M. Bhatt, C. Wouters, A. Flahive, W. Rahayu, and D. Taniar. Semantic completeness in sub-ontology extraction using distributed methods. In *Proc. Int. Conf. on Computational Science and its Applications (ICCSA)*, pages 508–517, Perugia, Italy, 2004. LNCS, Springer Verlag.
- [4] E. P. Bontas, M. Mochol, and R. Tolksdorf. Case studies on ontology reuse. In *5th Int. Conf. on Knowledge Management (I’Know’05)*, Graz, Austria, 2005.
- [5] C. Brewster, F. Ciravegna, and Y. Wilks. Background and foreground knowledge in dynamic ontology construction. In *Semantic Web Workshop, SIGIR’03*, Toronto, Canada, 2003.
- [6] L. Ding, T. Finin, A. Joshi, R. Pan, R. S. Cost, Y. Peng, P. Reddivari, V. C. Doshi, and J. Sachs. Swoogle: A semantic web search and metadata engine. In *Proc. 13th ACM Conf. on Information and Knowledge Management*, Nov. 2004.
- [7] Y. Ding and D. Fensel. Ontology library systems: The key to successful ontology re-use. In *Int. Semantic Web Working Symposium (SWWS)*, Stanford, CA, USA, 2001.
- [8] A. Doan and A. Halevy. Semantic integration research in the database community: A bried survey. *AI Magazine*, 26(1):83–94, 2005.
- [9] A. Farquhar, R. Fikes, and J. Rice. The ontolingua server: A tool for collaborative ontology construction. In *Proc. 10th Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff, Canada, 9–14 Nov 1996.
- [10] N. Guarino and C. Welty. Evaluating ontological decisions with ontoclean. *Communications of the ACM*, 45(2):61–65, 2002.
- [11] Y. Kalfoglou and M. Marco Schorlemmer. Ontology mapping: the state of the art. *The Knowledge Engineering Review*, 18(1):1–31, 2003.
- [12] A. Maedche, B. Motik, L. Stojanovic, R. Studer, and R. Volz. An infrastructure for searching, reusing and evolving distributed ontologies. In *The Twelfth International World Wide Web Conference (WWW’03)*, pages 439–448, Budapest, Hungary, 2003. ACM.
- [13] A. Maedche and S. Staab. Ontology learning for the semantic web. *IEEE Intelligent Systems*, pages 72–79, March/April 2001.

- [14] A. Magkanaraki, V. Tannen, V. Christophides, and D. Plexousakis. Viewing the semantic web through rvl lenses. In *Proc. Second Int. Semantic Web Conf. (ISWC)*, pages 98–112, Sanibel Island, Florida, 2003.
- [15] D. L. McGuinness, R. Fikes, J. Rice, and S. Wilder. An environment for merging and testing large ontologies. In *Proc. 17th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR-2000)*, Colorado, USA, 2000.
- [16] M. A. Musen, R. W. Ferguson, W. E. Grosso, N. F. Noy, M. Y. Grubezy, and J. H. Gennari. Component-based support for building knowledge-acquisition systems. In *Proc. Intelligent Information Processing (IIP 2000) Conference of the International Federation for Processing (IFIP), World Computer Congress (WCC'2000)*, pages 18–22, Beijing, China, 2000.
- [17] N. Noy and D. L. McGuinness. Ontology development 101: A guide to creating your first ontology. Technical Report KSL-01-05, Stanford Medical Informatics, Stanford, 2001.
- [18] N. F. Noy and M. A. Musen. The prompt suite: Interactive tools for ontology merging and mapping. *International Journal of Human-Computer Studies*, 59(6):983–1024, 2003.
- [19] N. F. Noy and M. A. Musen. Specifying ontology views by traversal. In *3rd Int. Semantic Web Conf. (ISWC'04)*, Hiroshima, Japan, 2004.
- [20] C. Patel, K. Supekar, Y. Lee, and E. Park. Ontokhoj: A semantic web portal for ontology searching, ranking, and classification. In *Proc. 5th ACM Int. Workshop on Web Information and Data Management*, pages 58–61, New Orleans, Louisiana, USA, 2003.
- [21] D. Quan and D. R. Karger. How to make a semantic web browser. In *13th Int. World Wide Web Conf. (WWW04)*, New York, 2004.
- [22] J. Seidenberg and A. Rector. Techniques for segmenting large description logic ontologies. In *Workshop on Ontology Management: Searching, Selection, Ranking, and Segmentation. 3rd Int. Conf. Knowledge Capture (K-Cap)*, pages 49–56, Banff, Canada, 2005.
- [23] D. Skuce. Conventions for reaching agreement on shared ontologies. In *Proc. 9th Banff Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff Conference Centre, Banff, Alberta, Canada, 1995.
- [24] D. Sleeman, S. Potter, D. Robertson, and M. Schorlemmer. Ontology extraction for distributed environments. In B. Omelayenko and M. C. A. Klein, editors, *Knowledge Transformation for the Semantic Web*, pages 80–91. IOS Press, Amsterdam, 02 2003.
- [25] H. Stuckenschmidt and M. Klein. Structure-based partitioning of large concept hierarchies. In *Proc. 3rd International Semantic Web Conference (ISWC2004)*, Hiroshima, Japan, 2004.
- [26] K. Supekar. A peer-review approach for ontology evaluation. In *Proc. 8th Int. Protege Conference*, pages 7–122, Madrid, Spain, 2005.
- [27] M. Uschold and M. Gruninger. Ontologies: principles, methods and applications. *The Knowledge Engineering Review*, 11(2):93–136, 1996.
- [28] J. Volker, D. Vrandečić, and Y. Sure. Automatic evaluation of ontologies (aeon). In *Proc. 4th Int. Semantic Web Conf. (ISWC)*, Galway, Ireland, 2005.
- [29] R. Volz, D. Oberle, and R. Studer. Implementing views for light-weight web ontologies. In *Proc. IEEE Database Engineering and Application Symposium (IDEAS)*, Hong Kong, China, 2003.
- [30] H. Yang, Z. Cui, and P. O'Brian. Extractin ontologies from legacy systems for understanding and re-engineering. In *Proc. 23rd IEEE Int. Conf. on Computer Software and Applications (COMPSAC)*, Phoenix, AZ, USA, 1999. IEEE Press.
- [31] Y. Zhang, W. Vasconcelos, and D. Sleeman. Ontosearch: An ontology search engine. In *Proc. 24th SGAI Int. Conf. on Innovative Techniques and Applications of Artificial Intelligence*, Cambridge, UK, 2004.