

# AceWiki: A Natural and Expressive Semantic Wiki

Tobias Kuhn  
Department of Informatics  
University of Zurich  
tkuhn@ifi.uzh.ch

## ABSTRACT

We present AceWiki, a prototype of a new kind of semantic wiki using the controlled natural language Attempto Controlled English (ACE) for representing its content. ACE is a subset of English with a restricted grammar and a formal semantics. The use of ACE has two important advantages over existing semantic wikis. First, we can improve the usability and achieve a shallow learning curve. Second, ACE is more expressive than the formal languages of existing semantic wikis. Our evaluation shows that people who are not familiar with the formal foundations of the Semantic Web are able to deal with AceWiki after a very short learning phase and without the help of an expert.

## Author Keywords

Controlled Natural Language, Attempto Controlled English (ACE), Semantic Wiki, Semantic Web, Ontology, Usability Test, Experiment

## ACM Classification Keywords

H.5.2 User Interfaces; I.2.4 Knowledge Representation Formalisms and Methods.

## INTRODUCTION

Ontologies are mostly defined within communities. Semantic wikis [16] could support the community processes for building and maintaining an ontology. Unfortunately, existing semantic wikis are often hard to understand for novices and do not have sufficient support for expressive ontology languages.

AceWiki<sup>1</sup> is a prototype that tries to solve both problems and demonstrates the use of controlled natural language for semantic wikis. Being a semantic wiki, it combines the ideas and technologies of the Semantic Web with the concepts of wikis. The use of controlled natural language allows ordinary users — who are not familiar with the concepts of logic and ontologies — to understand, modify, and extend the formal content of the wiki. Figure 1 shows a screenshot of the AceWiki prototype.

## BACKGROUND

In this section, we refer to existing semantic wikis and explain their concepts. Next, we introduce Attempto Controlled English (ACE) which is the controlled natural language that is used for AceWiki.

<sup>1</sup><http://attempto.ifi.uzh.ch/acewiki>



Figure 1. The web interface of the AceWiki prototype.

## Semantic Wikis

*Platypus Wiki* [6] was one of the first semantic wikis. It was introduced in 2004 and influenced many of the subsequent systems. The formal statements are represented in RDF and *Platypus Wiki* does not try to hide the technical details of this language. Thus, it is aimed at users who have experience with RDF. *WikSAR* [3] is another semantic wiki that has a similar design, but in contrast it does not show the RDF-specific details and has a very simple interface. *SemperWiki* [12] is a third example following a similar approach. All those three systems are very RDF-centric.

*Rhizome* [15] belongs to another category of semantic wikis, making use of RDF as well. Its goal is to “represent informal, human-authored content in a semantically rich manner”. It allows the users to organize the informal knowledge with the help of formal structures. The focus is rather on management of knowledge than on advanced reasoning.

*IkeWiki* [13] is a rich and sophisticated semantic wiki. It has a special focus on reusing existing informal articles (e.g. from Wikipedia) and to augment them with formal annotations. *Semantic MediaWiki* [17] is probably the most popular and most mature semantic wiki. It relies on the same wiki engine as the well-known Wikipedia and the goal of

the authors is even to integrate their extended wiki syntax into Wikipedia. In both cases, RDF and OWL are used in the background.

*SweetWiki* [5] is another semantic wiki approach. It is intended specifically for clearly identified communities, e.g. in intranets.

All those semantic wikis have in common that they rely on RDF and support only simple *subject-predicate-object* structures. Some of them make use of the more expressive language OWL, but only to define the background model in a very static way. For example, none of the wikis allows the users to define general concept inclusion axioms like “every person who writes something is an author”. For many applications simple RDF-triples may be sufficient, but there are cases where more expressive languages are needed. For example, there are several existing ontologies that exploit the expressivity of OWL, e.g. GALEN<sup>2</sup> and the Ordnance Survey Hydrology ontology<sup>3</sup>. It would be very convenient if these ontologies could be managed collaboratively within a wiki.

It is interesting that all the wikis addressed above organize their formal relations in “annotations” or as “metadata”. This indicates that the formal statements are not considered the main content but rather an enrichment thereof.

In contrast, the approach of the *myOntology* project [14] is to build and maintain ontologies within a dedicated wiki. But again, it focuses on lightweight (i.e. relatively inexpressive) ontologies. They claim that “... most users can not be expected to be able to add axioms”. We show that this is not true if controlled natural language is used.

### Attempto Controlled English

AceWiki uses the controlled natural language Attempto Controlled English (ACE) [8]. ACE looks like English but avoids the ambiguities of natural language by restricting the syntax [1] and by defining a small set of interpretation rules [2]. The ACE parser<sup>4</sup> translates ACE texts automatically into *Discourse Representation Structures* [7] which are a syntactical variant of first-order logic. Thus, every ACE text has a single and well-defined formal meaning. ACE supports a wide range of natural language constructs, e.g. singular and plural noun phrases, active and passive voice, relative phrases, anaphoric references, existential and universal quantifiers, negation, and modality. (However, for AceWiki we use only a subset of ACE and support not all of those constructs.) ACE has successfully been applied for different tasks, e.g. as a query language for ontologies [4], as a knowledge representation language for the biomedical domain [10], and as a rule language for a multi-semantics rule engine [11].

Furthermore, ACE has been used as a natural language front-end to OWL with a bidirectional mapping of ACE to OWL [9]. This mapping covers all of OWL 1.1 except data prop-

<sup>2</sup><http://www.co-ode.org/galen/>

<sup>3</sup><http://www.ordnancesurvey.co.uk/ontology/>

<sup>4</sup><http://attempto.ifi.uzh.ch/ape>

erties and some very complex class descriptions. AceWiki relies on this work for translating ACE sentences into OWL, which allows us then to do reasoning with existing OWL reasoners.

### DESIGN

The main goal of AceWiki is to improve knowledge aggregation and representation. AceWiki should be easier to use and understand than other semantic wikis. In addition, it should support a higher degree of expressivity.

Unlike other semantic wikis, the formal statements are not contained in “annotations” and are not considered “metadata”, but they are the main content of our wiki.

In order to achieve a good usability and still support a high degree of expressivity, AceWiki follows three design principles: *naturalness*, *uniformity*, and *strict user guidance*.

By *naturalness* we mean that the formal semantics has a direct connection to natural language. *Uniformity* means that only one language is used at the user-interface level. *Strict user guidance*, finally, means that a predictive editor ensures that only well-formed statements are created by the user. We will now discuss these three principles and show how they are achieved in AceWiki.

### Naturalness

AceWiki is natural in the sense that the ontology is represented in a form that is very close to natural language. This requires a direct mapping of ontological entities to natural language words. In AceWiki, individuals are represented as proper names (e.g. “Switzerland”), concepts<sup>5</sup> are represented as nouns (e.g. “country”), and roles<sup>6</sup> are represented as transitive verbs (e.g. “overlaps-with”) or as of-constructs (e.g. “part of”). Using those words together with the predefined function words of ACE (e.g. “every”, “if”, “then”, “something”, “and”, “or”, “not”), we can express ontological statements as ACE sentences. Since every ACE sentence is a valid English sentence, those ontological statements can be immediately understood by any English speaker.

We believe that ontological terms like “property”, “range”, or “subclass” are unknown or unclear to most potential users of a semantic wiki. Such terms do not comply with our principle of naturalness. We show that it is possible to avoid them, not only for the knowledge representation itself but also for captions, labels, help pages, etc. For example, instead of saying something like “man is a subclass of human” that uses the ontological term “subclass”, we can simply say “every man is a human” which does not use any special terms. In the cases where there is no such solution, we use linguistic terms like “noun”, “verb phrase”, or “sentence” instead of ontological terms. Those linguistic terms should be familiar to most users, since they are taught even in elementary schools.

A minor problem arises when using controlled natural lan-

<sup>5</sup>in OWL called *classes*

<sup>6</sup>in OWL called *properties*

# protein

[references](#) [delete](#)

**Word**

---

**word class:** *noun*  
**noun form:** *protein* [edit...](#)

**Hierarchy**

---

**upward:**  
▶ Every protein is a molecule .

**Individuals**

---

▶ Act1 is a protein .  
▶ Bub1 is a protein .  
▶ Cav1 is a protein .  
▶ Cbl is a protein .  
▶ CHES1 is a protein .  
... [show all](#)

- ▶ every protein interacts-with a protein .  
▶ no protein interacts-with every protein .  
▶ IRAK-1 is a protein that is a isoform of IRAK .  
[add...](#)

Figure 2. This is a screenshot of an article of an exemplary AceWiki instance. The title denotes the topic of this article, i.e. the concept “protein”. The first box shows the linguistic information of this entity. The other two boxes contain sentences that follow a certain pattern, i.e. hierarchy statements and individual assignments. The unrestricted sentences are at the bottom of the page. All the text that is not in italics is ACE. The blue triangle in front of an ACE sentence denotes that this sentence is within OWL. Otherwise a red triangle is shown.

guage. Since informal (uncontrolled) natural language is still needed sometimes (e.g. for labels, help pages, introductory notes, etc.), we have to make sure that the user does not confuse informal natural language with ACE. For example, an informal introductory note could be misinterpreted as a formal statement, or a formal statement could be misinterpreted as an informal explanation. In order to overcome this problem, we use a very simple convention: normal font is used for formal statements and terms, whereas italics are used for informal statements and terms in uncontrolled language. In this way, a user can immediately find out whether a certain statement or term is part of the formal ontology or not.

Figure 2 shows a screenshot of an example wiki about proteins. All the formal representations appear in ACE, and all the text that is not ACE is displayed in italics.

## Uniformity

The Semantic Web community defines three categories of languages on the logic level of the Semantic Web stack: ontology languages (e.g. OWL), rule languages (e.g. SWRL), and query languages (e.g. SPARQL). Most languages cover only one of those categories, and languages of different categories look usually very different.

This distinction makes sense from a knowledge engineer’s

point of view: Ontology languages need a different kind of reasoning than rule languages, and query languages ask for knowledge whereas ontology and rule languages assert knowledge. But this should not lead us to the conclusion that the end-users have to learn three different languages.

We claim that at the user-interface level ideally one single language should cover all those categories. In the background, there might be several internal languages, but the users should need to learn only one. For many users who are not familiar with formal conceptualizations, learning one formal language is already a hard task. We should not make this learning effort harder than necessary.

ACE is able to represent those different kinds of formal statements in a very natural way. In the case of queries, this distinction does not need to be made explicit: If a sentence ends with a question mark then it is clear for the user that this is a query and not an assertion. However, queries are still future work for AceWiki.

AceWiki classifies declarative ACE sentences into three categories: Some can be translated into OWL, others can be translated into SWRL, and finally there are ACE sentences that have no representation in OWL or SWRL at all. In ACE, this distinction is not visible and we think that users should not bother about it. The only thing they need to know is that if using an OWL reasoner only the OWL-compliant sentences are considered.

For that reason, AceWiki marks statements that are within OWL with a blue triangle, and all the other statements with a red triangle (as it can be seen on Figure 2). Statements that are within SWRL could be marked in a similar way.

## Strict User Guidance

Learning a new formal language is normally accompanied by frequent syntax error messages from the parser. Wikis are supposed to enable easy and quick modifications of the content, and syntax errors can certainly be a major hindrance in this respect, especially for new users.

This problem can be solved by guiding the users during the creation of new statements in a strict manner. By strict we mean that the creation of syntactically incorrect sentences is simply made impossible. This can be achieved by a predictive editor that guides the user step by step and ensures the syntactic correctness. Syntactic correctness can be subdivided into lexical correctness and grammatical correctness. Lexical correctness means that only the words that are defined in a certain lexicon are used. Grammatical correctness means that the grammar rules are respected.

To some degree, a predictive editor could also take care of the semantic correctness. It could prevent the users from adding statements that introduce inconsistency into an underlying ontology. If the verb “meets”, for example, is defined in the ontology as a relation between humans then the predictive editor could prevent the user from writing sentences like “a man meets a car”, assuming that the ontology

says that “car” is not human.

AceWiki has a predictive editor that is used for the creation and modification of ACE sentences. It ensures lexical and grammatical correctness of the resulting sentences. The semantic correctness is not enforced, but the words that seem to be semantically suitable are shown first in the list. The suitable words are retrieved on the basis of the hierarchy of concepts and roles and the domain and range restrictions of roles. For example, if a user creates the incomplete sentence “Limmat flows-through” and there is a range restriction that says “if something flows-through something Y then Y is a city” then the individuals that are known to be cities are shown first in the list.

In order to be convenient for both novices and advanced users, the stepwise creation of a sentence can be done either by clicking on lists of proposed words (for novices) or by typing the words in a text field (for advanced users). Both alternatives are supported by a single graphical interface allowing the users to switch from one to the other at any time. Figure 3 shows a screenshot of the predictive editor of AceWiki.

AceWiki provides special support for often used sentence patterns, i.e. concept and role hierarchies, domain and range of roles, and individual assignments. Such sentences are kept in separate boxes in the articles. The same predictive editor is used to create and modify them but with a reduced grammar that covers just the respective subset of ACE.

## EVALUATION

In order to evaluate AceWiki, we set up a small-scale end-user experiment. The hypothesis to be tested was whether average people (i.e. people who are not familiar with ontologies and logic) are able to learn how to deal with AceWiki within a short amount of time and without the help of an expert.

### Experiment Design

The experiment was performed through the internet and it had a very simple design. The prerequisites for participation were only basic English skills and access to a computer with a broadband internet connection. We recruited 20 participants.

The participants received an instruction sheet which they read before they started with the experiment. These instructions explained the procedure and the task, but they did not explain how to interact with the AceWiki interface.

After reading the instructions, the participants were ready to start with the experiment. In the end, they filled out a questionnaire which asked for their background and their experiences with AceWiki.

The task for the participants was to visit AceWiki and to add knowledge to it. They were free to choose what kind of knowledge to add, as long as they followed three restrictions:

- The participants should add only knowledge that is true or at least true in most cases.
- The knowledge should be general, i.e. verifiable by others.
- The participants were allowed and encouraged to change or even delete the contributions of other participants if they found them violating one of the first two restrictions.

Furthermore, the participants were encouraged to add a couple of complex sentences starting with “every”, “no”, or “if”.

Altogether, each participant should spend between 30 minutes and two hours (possibly split into several intervals) within a time-frame of 14 days.

## Results

The basis for the evaluation of the experiment were the questionnaire and the log files from the server.

Most of the 20 participants were students or graduates. Two participants had a computer science background, but they were not experts in the fields of Semantic Web or logic. The table below shows the exact distribution.

participants in total	20 (100%)
students in computer science	1 (5%)
graduates in computer science	1 (5%)
other students	8 (40%)
other graduates	8 (40%)
no academic background	2 (10%)

In the questionnaire, the participants were asked how familiar they are with different technical terms. Figure 4 shows the result. The term *OWL* was almost completely unknown. The majority of the participants have never heard the terms *first-order logic* or *UML*. Only in the case of the terms *Semantic Web* and *ontology*, we have a substantial minority knowing those terms. The results show that the participants had no considerable background in Semantic Web technologies or similar fields.

Figure 5 shows two examples of AceWiki articles how they came out of the experiment. Those screenshots intuitively show that the participants understood the ideas of AceWiki.

In order to measure how well the participants managed to work with AceWiki, we evaluated the sentences that they created or modified. For each of these sentences, we checked manually whether they represent a correct and sensible fact of the real world under the ACE semantics. The results are shown in the table below.

	total	average	median	min	max
$S$	186	9.3	7.0	1	31
$S^+$	148	7.4	6.0	1	22
$S^-$	38	1.9	1.0	0	9
$S^+ / S$	0.796	0.854	0.912	0.5	1.0

The overall number of sentences  $S$  is 186. 148 of them are considered correct and sensible ( $S^+$ ), whereas the remaining 38 are not ( $S^-$ ). We do not count the sentences that have

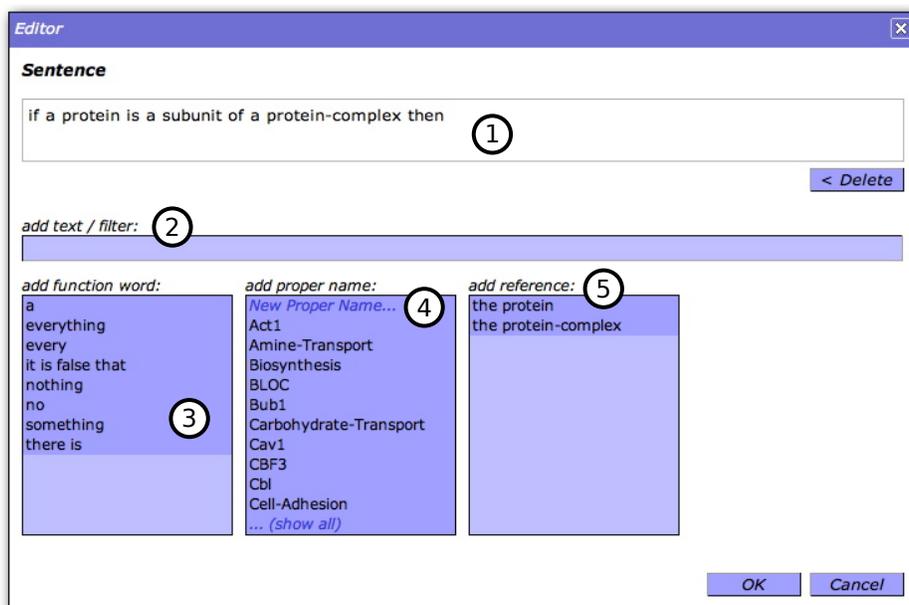


Figure 3. This is a screenshot of the predictive editor of AceWiki. (1) is a read-only text field that shows the beginning of an ACE sentence. This beginning has been entered by the user and it has been accepted by the predictive editor as a correct sentence beginning. The button “Delete” can be used to undo the last step. The text field (2) is used for entering the next words of the sentence. If they are accepted then they are moved to the text field (1). The tab key triggers autocompletion. (2) can also be used to filter the entries of the menus (3). Clicking on the entries of the menu boxes (3) is an alternative way to construct a sentence. There is a menu box for each word class that is allowed at the current position. If a word is not yet known then it can be added on-the-fly by clicking on the respective menu entry (4). Furthermore, references can be introduced that point to objects occurring earlier in the sentence (5).

been created and later removed by the same participant. If someone modified a sentence that was created by himself then we count only the last version. If a sentence was modified by another participant then the respective versions of the sentence count for each of the participants. Thus, this table shows the achievements of the individuals, not of the community.

Let us first explain how we judged whether a sentence is correct and sensible. The main criteria was that the sentence is a true statement (under the ACE semantics) of the real world using the common interpretations of the natural language words. If this is not the case, e.g. for “every musician is a man”, then the sentence counted for  $S^-$ . In the case of 24% of the incorrect sentences, words were used in the wrong category, for example “every London is a city” where “London” was introduced as a common noun instead of a proper name. Another 24% of the incorrect sentences are statements like “a city is a landscape-element” which is interpreted in ACE as having only existential quantification: “there is a city that is a landscape-element”. Even though this is a correct statement about the real world, the user probably wanted to say “every city is a landscape-element”. For that reason, such sentences were considered incorrect. (The remaining 52% of the incorrect sentences do not show specific patterns for further categorization.) On the other hand, sentences like “every country is a part of a continent” were considered correct, even though it depends on the interpretation of “part of” and “continent”. One could say that Russia is not part of a continent, but only overlaps with Europe and

Asia. But in this case, there is no reason to believe that the participant wanted to say something different than what the ACE semantics defines.

The results show that almost 80% of the created sentences were correct, which is — we think — a very good result. Furthermore, every participant created at least one correct sentence. Another interesting result is that the ratio of correct sentences was in the worst case 50%. Thus, no participant created more wrong sentences than correct ones. Altogether, we can conclude that all of the participants managed to deal with AceWiki.

Another interesting point to investigate is whether there was a fruitful community process. If we look at the results from a community perspective then we should not consider the sentences that have been removed later by someone else and we should count sentences that have been edited by different participants only once. Under these conditions, we get slightly different values.

$S_c$	179
$S_c^+$	145
$S_c^-$	34
$S_c^+ / S_c$	0.810

The ratio of correct sentences is slightly higher than the one from the individual data. This means that the community interaction — i.e. the deletion and modification of other’s sentences — increased the quality of the content in this par-

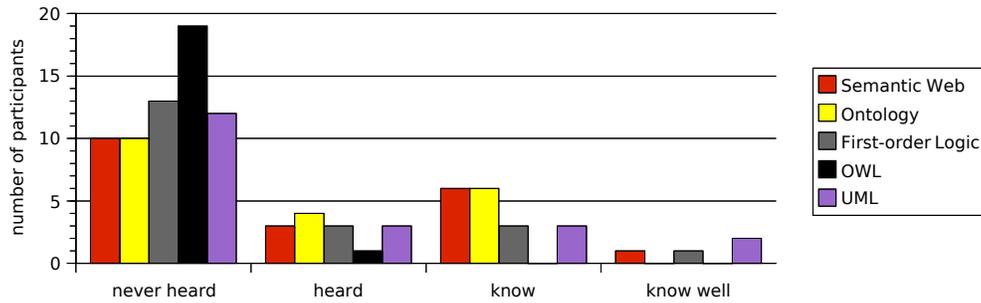


Figure 4. This chart shows the familiarity of the participants with the terms *Semantic Web*, *ontology*, *first-order logic*, *OWL*, and *UML*. This data was retrieved from five questions of the questionnaire. For each of the terms the question was “How familiar are you with this term?” and there were four choices: “I have never heard it”, “I have heard it but I don’t really know what it means”, “I know more or less its meaning”, “I know this term (very) well”.

## Limmat

[references](#) [delete](#)

### Word

**word class:** proper name  
**name:** Limmat [edit...](#)

### Assignment

▶ Limmat is a river .

- ▶ Limmat flows-through Zurich .
- ▶ Limmat is a river that is a part of Zurich and that is a part of Switzerland .
- ▶ if Limmat is a part of Zurich then it is false that Limmat is a part of Uster .
- ▶ Limmat does not flow-through Uster .
- ▶ Limmat flows-into Rhein and does not flow-into Spree .
- [add...](#)

## Karlsruhe

[references](#) [delete](#)

### Word

**word class:** proper name  
**name:** Karlsruhe [edit...](#)

### Assignment

▶ Karlsruhe is a city .

- ▶ it is false that Karlsruhe is Zurich .
- ▶ Karlsruhe has a Football-Club .
- ▶ Karlsruhe is a part of Germany .
- ▶ Karlsruhe does not belong-to Schwaben .
- ▶ Karlsruhe is a part of Baden .
- [add...](#)

Figure 5. This figure shows the screenshots of two exemplary articles how they looked at the end of the experiment. All the sentences have been created by the participants of the experiment.

ticular experiment. However, we do not have enough data for making a general statement.

The participants were encouraged to create not only simple sentences, but also some complicated ones. We can now find out whether they managed to do so. Figure 6 shows the most frequent sentence patterns and reveals the occurrence of negation (i.e. “does not”, “is not”, “no”, or “it is false that”) and implication (i.e. “if ... then”, “every”, or “no”). Only the correct sentences are considered here.

The two most frequent sentence patterns were superconcept statements (positive e.g. “every canal is a waterbody” or negated e.g. “it is false that every animal is a mammal”) and individual assignments (positive e.g. “Zurich is a city” or negated e.g. “Bob-Dylan is not a woman”). Also quite frequent were role instances (positive e.g. “Limmat flows-through Zurich” or negated e.g. “it is false that Winston-Churchill is a prime-minister of Denmark”) and superrole statements (e.g. “if something X protects something Y then X shelters Y”). All the examples are sentences that the participants created during the experiment.

It is remarkable that there is a long tail of other sentence patterns and that 61% of the sentences contained a negation or an implication or both. This shows that the participants made use of the high expressivity of ACE.

In order to evaluate the performance of the participants, we have to take the time dimension into account. The following table shows the time values (in minutes) that we retrieved from the log files.

	total	average	median	min	max
$t$	931.2	46.6	39.1	7.7	132.5
$t_f$		11.0	8.0	2.5	33.8
$t/S^+$	6.3	8.2	7.2	2.9	24.7

The first line shows the overall time  $t$  of the participants. This is the time they spent on AceWiki, not counting the time for reading the instructions and for filling out the questionnaire. The second line shows the time  $t_f$  needed for creating the first correct sentence. The final line contains the time per correct sentence and shows how well the participants performed. Thus, they needed on average 11.0 minutes to create

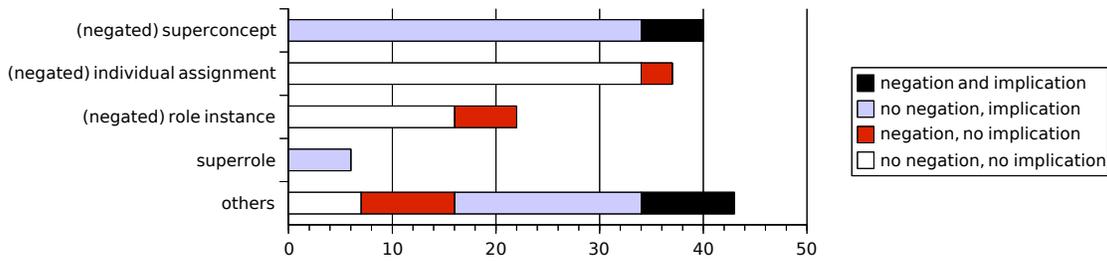


Figure 6. This chart shows the most frequent sentence patterns in absolute numbers. Only the correct sentences are considered.

their first correct sentence, and overall the time per correct sentence was 8.2 minutes.

Those results do not look very spectacular at first sight, but we have to recall the situation of the participants. When we start counting the minutes, the participants see AceWiki for the very first time. The instructions contained no explanation whatsoever of the AceWiki interface. In order to get familiar with this unknown interface, the participants started to navigate around, searched for terms, and explored the predictive editor. Some of them added new words without adding a sentence yet, and some added a sentence but removed it again. And then, after only eleven minutes, on average, the participants managed to create their first correct sentence. Over the complete duration, they created a correct sentence approximately every eight minutes, and — as Figure 6 shows — most of those sentences were quite complicated. We think that these are very good results under the given circumstances, and they show that AceWiki has indeed a shallow learning curve.

Restricting our attention to the time when the editor window was open, we can evaluate the usability of the predictive editor alone. The following table shows these values.

	total	average	median	min	max
$t_e$	573.2	28.7	24.7	1.7	85.6
$t_{ef}$		5.3	3.0	0.9	28.6
$t_e/S^+$	3.9	4.6	3.3	1.7	12.5

The editor window was on average open for 5.3 minutes (not necessarily continuously) before the first correct sentence was created, and overall the time per correct sentence was 4.6 minutes. Note that this includes also the time for wrong sentences, the time for sentences that have been canceled during their creation, and the time that was needed to introduce new words.

Again, we think that these are very good results considering that the users were not trained how to interact with the predictive editor.

Finally, we can take a look at the participants’ feedback after the experiment. In the questionnaire we asked how easy or how difficult the handling of AceWiki was. Figure 7 shows the result. The responses are distributed symmetrically and have a peak at “medium”. On the one hand, this is a good result since only 25% of the users found it hard to

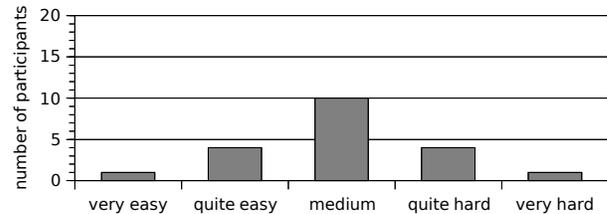


Figure 7. This chart shows how the participants answered the question “How easy/difficult was the handling of AceWiki?” after the experiment.

use AceWiki. We have to consider that the participants experienced only the costs of formal knowledge representation, but not the benefits (since reasoning features were missing). Furthermore, knowledge representation is inherently a difficult task. Probably, it will never be possible to make this very easy for everybody. On the other hand, the results show that there is certainly room for improvement.

## FUTURE WORK

There are various possibilities to improve AceWiki. ACE supports a wide range of word classes and we plan to support more of them in AceWiki. Intransitive verb (e.g. “flows”) or intransitive adjectives (e.g. “large”) could be used to represent concepts. Transitive adjectives (e.g. “located-in”) and comparative adjectives (e.g. “larger than” or “as large as”) could be used to represent roles. Furthermore, roles could be represented by adverbs together with transitive verbs (e.g. “directly interacts-with”).

Another important extension would be to support number restrictions like “every car has exactly 4 wheels” or “every person has at most 2 parents”. Such sentences are already supported by ACE and also by the ACE-to-OWL translator.

In order to enable reasoning with the formal content of AceWiki, we plan to tightly integrate an existing OWL reasoner. The interaction with the reasoner should be done via ACE questions, e.g. “which river flows-through Zurich?” or “is Zurich a part of Europe?”. A very nice feature would be to support inline queries, as shown by other semantic wikis (e.g. WikSAR and IkeWiki).

AceWiki is still a prototype and it probably will stay so for the near future. It lacks a safe concurrency management, has only a rudimentary support for persistent storage, and it

does not feature a history and undo facility. All those points would of course be crucial for a real-world application. So far, we focused on the aspects that are interesting from a scientific rather than an industrial point of view.

## CONCLUSIONS

We presented the AceWiki prototype and an experimental evaluation of its usability. The fact that the participants of the experiment created on average a correct statement every eight minutes shows that AceWiki is easy to learn. Furthermore, the users were able to create complex statements that go beyond the expressivity of other semantic wikis. The results indicate that our three design principles — naturalness, uniformity, and strict user guidance — are beneficial for enabling unexperienced users to effectively interact with a semantic wiki.

Since AceWiki is still an emerging system and since we observed several possibilities for improvements during the experiment, we think that even better results will be possible in the future.

## ACKNOWLEDGEMENT

This research has been funded by the European Commission and the Swiss State Secretariat for Education and Research within the 6th Framework Programme project REWERSE, and by the University of Zurich within the research grant program 2006 (Forschungskredit). I would like to thank Norbert E. Fuchs and Kaarel Kaljurand for their inputs.

## REFERENCES

1. *ACE 6.0 Construction Rules*, 19 January 2008.  
[http://attempto.ifi.uzh.ch/site/docs/ace\\_constructionrules.html](http://attempto.ifi.uzh.ch/site/docs/ace_constructionrules.html)
2. *ACE 6.0 Interpretation Rules*, 19 January 2008.  
[http://attempto.ifi.uzh.ch/site/docs/ace\\_interpretationrules.html](http://attempto.ifi.uzh.ch/site/docs/ace_interpretationrules.html)
3. David Aumüller, Sören Auer. Towards a Semantic Wiki Experience – Desktop Integration and Interactivity in WikSAR. *Proc. First Workshop on The Semantic Desktop*, 2005
4. Abraham Bernstein, Esther Kaufmann, Norbert E. Fuchs, June von Bonin. Talking to the Semantic Web — A Controlled English Query Interface for Ontologies. *Proc. 14th Workshop on Information Technology and Systems*, 2004
5. Michel Buffa, Gaël Crova, Fabien Gandon, Claire Lecompte, Jeremy Passeron. SweetWiki: Semantic Web Enabled Technologies in Wiki. *Proc. First Workshop on Semantic Wikis — From Wiki To Semantics (SemWiki2006)*, 2006
6. Stefano Emilio Campanini, Paolo Castagna, Roberto Tazzoli. Platypus Wiki: a Semantic Wiki Wiki Web. *Proc. First Italian Semantic Web Workshop*, 2004
7. Norbert E. Fuchs, Kaarel Kaljurand, Tobias Kuhn. *Discourse Representation Structures for ACE 6.0*, Technical Report ifi-2008.02. Department of Informatics, University of Zurich, 2008
8. Norbert E. Fuchs, Kaarel Kaljurand, Gerold Schneider. Attempto Controlled English Meets the Challenges of Knowledge Representation, Reasoning, Interoperability and User Interfaces. *Proc. 19th International FLAIRS Conference (FLAIRS'2006)*, 2006
9. Kaarel Kaljurand. *Attempto Controlled English as a Semantic Web Language*. PhD thesis, Faculty of Mathematics and Computer Science, University of Tartu, 2007
10. Tobias Kuhn, Loïc Royer, Norbert E. Fuchs, Michael Schroeder. Improving Text Mining with Controlled Natural Language: A Case Study for Protein Interactions. *Proc. Third International Workshop on Data Integration in the Life Sciences 2006 (DILS'06)*, Springer, 2006
11. Tobias Kuhn. AceRules: Executing Rules in Controlled Natural Language. *Proc. First International Conference on Web Reasoning and Rule Systems (RR2007)*, Springer, 2007
12. Eyal Oren. Semperwiki: a semantic personal Wiki. *Proc. First Workshop on The Semantic Desktop*, 2005
13. Sebastian Schaffert. IkeWiki: A Semantic Wiki for Collaborative Knowledge Management. *Proc. First International Workshop on Semantic Technologies in Collaborative Applications (STICA06)*, 2006
14. Katharina Siorpaes, Martin Hepp. myOntology: The Marriage of Ontology Engineering and Collective Intelligence. *Proc. Bridging the Gap between Semantic Web and Web 2.0*, 2007
15. Adam Souzis. Building a Semantic Wiki. *IEEE Intelligent Systems*, 20:8791, 2005
16. Roberto Tazzoli, Paolo Castagna, Stefano Emilio Campanini. Towards a Semantic Wiki Wiki Web. *Proc. Third International Semantic Web Conference (ISWC2004)*, 2004
17. Markus Krötzsch, Denny Vrandečić, Max Völkel, Heiko Haller, Rudi Studer. Semantic Wikipedia. *Journal of Web Semantics*, 5/2007, pp. 251–261, Elsevier