Addressing the Cognitive Difficulties of Expressing N-ary Relations in Semantic Web Data

Stephen Davies, Jessica Zeitz, Jesse Hatfield
University of Mary Washington
1301 College Ave
Fredericksburg, VA 22401
1-540-654-1317
{sdavies, jzeitz, jhatfiel}@umw.edu

ABSTRACT
We present results from an empirical study in which everyday users attempted to generate formal knowledge representations for use in the Semantic Web. In particular, we focus on one especially difficult aspect of knowledge creation: statements that embody n-ary relations and therefore require reification of the verb in order to be expressible in standard RDF. In a cognitive experiment performed on over 80 novices, participants were asked to author statements containing n-ary relations corresponding to textual passages they were given. Our study compares the results between visual and text-based representations, illustrates the extent of the problem, and offers an alternative syntax for such relations that relieves several difficulties users face in properly formulating these statements. Our results soundly demonstrate that by allowing the use of this alternate syntax in place of traditional approaches, non-initiates can achieve much greater accuracy and coverage in the knowledge they generate. Further, knowledge modeled with the syntax can be trivially converted to standard RDF triples "behind" the user interface, so that the knowledge a user generates constitutes valid Linked Data.

Categories and Subject Descriptors
H.5.2 [**User Interfaces**]: Interaction styles; H.5.4 [**Hypertext/ Hypermedia**]: User issues.

General Terms
Design, Experimentation, Human Factors.

Keywords
Semantic Web, Linked Data, User Interface Design

1. INTRODUCTION
A successful, global-scale Semantic Web presupposes large amounts of instance data available for machines to process. In addition to mining such data from databases and text, it seems desirable for humans to directly author at least some of this data. Yet formal knowledge representation is difficult and error-prone for most non-technical people, and few studies exist to determine why, or to shed light on how to address the cognitive barriers users may face.

I-SEMANTICS 2010, September 1-3, 2010 Graz, Austria

Constructing formal knowledge representations is a very different activity from writing in natural language, which is the way that most laypeople have contributed to the Web to date. Writing blog posts or composing Wikipedia articles involves a medium of expression intimately familiar to the non-technical user, since it is the way we communicate every moment of every day. Graph-based knowledge structures, on the other hand, are foreign soil. They demand a degree of precision to which everyday communicators are unaccustomed, tolerate no ambiguity, and often require restructuring of the sentences which natural language thinkers may instinctively form. Cognitive psychologists and knowledge engineers alike (e.g., [2,14,18]) have described semantic networks in ways that seem straightfoward to the computer scientist, and it has even been argued that human memory itself is built upon a similar structure. But in our experience, non-technical users have surprising difficulty in putting knowledge into this form. They do have successes, but in general their results are erratic, inconsistent, and certainly not in keeping with the level of quality that Semantic Web enthusiasts hope will characterize the global Web of Data.

If non-technical humans are to be meaningful contributors, then, we believe that this problem needs to be more fully understood. The specific strengths and weaknesses that users have in creating RDF-compatible knowledge must be identified and then used to guide user interface design.

To this end, we have begun conducting a series of paper-and-pencil based cognitive experiments to discover what kinds of difficulties users encounter when trying to encode everyday knowledge into a triple-based syntax. The advantage of paper-based experiments rather than studying a particular RDF editor or annotation tool is that all factors having specifically to do with the tool are eliminated. This allows us to study the problem of knowledge structuring itself, which is central to the editing process. While it is certainly true that for a user to successfully author RDF he or she must correctly understand and employ an electronic tool's mechanisms of expression, the pure cognitive step of breaking down the knowledge into a fundamentally graph-based paradigm (rather than natural langauge sentences) is a prerequisite to all else. Hence the need to empirically investigate where users encounter obstacles in this process.

One particularly revealing find, which we focus on in this paper, is that knowledge involving n-ary relations – that is, where three or more entities are related in some way – is especially problematic to express. It appears, in fact, that the vast majority of users simply cannot achieve any reasonable degree of accuracy in generating n-ary relations by the traditional technique of reifying the verb, even after seeing examples where this is done. By using an alternative syntax, however, which permits modifiers to the predicate rather than requiring it to be reified, we discovered that users' success rate can be substantially increased.

2. RELATED WORK

Precious few studies have been conducted to discover what makes the process of properly creating instance data so difficult. Semantic Web researchers have produced a plethora of tools for RDF creation, many with impressive features (e.g., [4,13]), and some specifically claiming to be well-suited for the non-technical user (e.g., [3,7]). But with very few exceptions, no compelling studies have evaluated the level of effectiveness of such tools, nor what aspects in particular make them effective (or ineffective.) The great majority have provided no user studies at all; a few (e.g., [8]) point to a user community as evidence of effectiveness; occasionally (e.g., [16]) a case study is performed illustrating use in a limited setting, often by Semantic Web experts. But the process of practical knowledge creation by everyday users is not being given much attention.

The most helpful study of human graph-based knowledge generation is by Staab *et al*.[15]. They performed an in-depth analysis of the behavior of nine experimental subjects who used the OntoAnnotate semantic annotation tool to add machine-processible metadata to web pages. Their primary measure was inter-annotator agreement; that is, the degree to which different subjects independently annotated a page in the same way. Their conclusion, roughly speaking, was that novices to the Semantic Web, operating in a general domain (where they are not experts), will not in general produce high-quality structured knowledge, or at least not knowledge that agrees with one another. This confirms what we have seen in the general case. Staab, *et al,* did not, however, study n-ary relations in particular.

Another study of interest involved military professionals who were given structured knowledge generation tasks to perform using a domain-specific version of Protege.[10] In addition to studying retention skills and efficiency rate, Noy *et al*. examined the users' resulting knowledge bases for correctness and completeness. One of the key findings was that the domain-specific extension to the tool (providing a custom display for military combat units) provided a great enhancement to the average user's knowledge acquisition rate, and a modest but still significant enhancement to the correctness of knowledge formulation. This study did not cover n-ary relations specifically but it does testify to the difficulty of open-ended knowledge formulation. The more that users are channeled into prescribed structures, the more likely they are to succeed. One of the purposes of our study is to determine how well novices can succeed when they are *not* so channeled.

Joseph Novak and his colleagues have performed numerous studies (e.g., [9]) of students' tendencies in creating "concept maps." A concept map is a graph-based knowledge structure, consisting of labeled nodes connected by labeled and directed links, which resembles an RDF graph in many ways. The act of constructing a concept map for some field of knowledge can be a tool for learning and intellectual exploration, and a finished concept map can be evaluated to assess the learner's understanding (see the work of Ruiz-Primo and Shavelson [12] for a survey of evaluation techniques). These concept map studies do not meet our need for understanding the RDF generation process, however, for two reasons. First, these efforts use concept maps to evaluate a student's knowledge of some difficult domain, not whether the concept map itself is an adequate expression. In our work, we are not testing whether the user has understood the knowledge correctly (since the knowledge itself is trivial) but rather whether this well-grasped knowledge can be formulated legally according to the structure demanded by RDF. Secondly, concept maps are ultimately too informal to meet the requirements of the Semantic Web. They are evaluated according to the standards of a forgiving human reader who supplies some degree of interpretation, rather than the rigorous standards of unguided machine processibility. One noteworthy manifestation of this is that concept maps encourage the use of "chains" (see below) which do not represent the independently meaningful statements mandated by RDF.

Note that all of the above remarks are concerned with generating data at the instance level, not the schema (ontology) level. The latter process has indeed been studied in detail; Cristani and Cuel [5] and Abar [1] summarize some of the prevalent methodologies. This is a very different problem from creating instance data, however, since in ontology creation the participants are typically domain experts with considerable incentive to engage in a lengthy, collaborative process to produce a lasting domain description. By contrast, we study the behavior of less-skilled, "everyday" users who we hope could enrich the Semantic Web without much training.

3. EXPRESSING N-ARY RELATIONS

A cardinal feature of RDF is that any encoded knowledge must be broken down into subject/predicate/object triples each of which can stand alone. This works well in many cases, since often we have a resource (say, "John") and want to relate to it just one other object (as in "John reportsTo Sue") or attribute (as in "John salaryUSD 65000"). Much RDF knowledge is easily expressible in this way.

But for a more complex sentence, the solution is not nearly as obvious. Consider the following fact:

> Lt. Gen. Ramsden led the 9th Australian Division near the Ruweisat Ridge between August 30th and Sept. 5th, 1942, sustaining 200 casualties.

Here a number of different concepts are tied together, and in a way that cannot be separated without losing information. We cannot simply have "Ramsden led 9thAustralian" and "9thAustralian near RuweisatRidge," because although both triples are indeed true, the two taken together do not imply that Ramsden led the division near the ridge; only that at some time, the division was located near the ridge (quite possibly when Ramsden was not in command at all). The same is true for the date range and the casualty count. Ramsden's command of these troops at this time is an act for which several supplementary pieces of data must be made to converge coherently.

3.1 Traditional approach: predicate reification

The standard solution for modeling this in RDF is to reify the predicate into a resource in its own right, name it, and attach the other facts to it. Hence we might have:

RamsdenEarlySept1942Command ledBy Ramsden
RamsdenEarlySept1942Command involved 9thAustralian
RamsdenEarlySept1942Command located RuweisatRidge
RamsdenEarlySept1942Command began 30-Aug-1942
RamsdenEarlySept1942Command ended 5-Sep-1942
RamsdenEarlySept1942Command numCasualties 200

These triples, taken in tandem, solve the problem. (The same information is depicted pictorially in figure 1.) We now have a single unifying resource ("RamsdenEarlySept1942Command") that unites the peripheral facts, so that there is no ambiguity that the dates, places, and statistics all apply to this particular historical event.

But the problem is "solved" only from the standpoint of the system. From a human factors point of view, it is far from satisfactory. Human encoders, as we will show presently, have an enormously difficult time structuring knowledge in this way. It is so counterintuitive, in fact, that the chances of a non-technical user doing it successfully for a sentence like the one above are almost neglibigle.

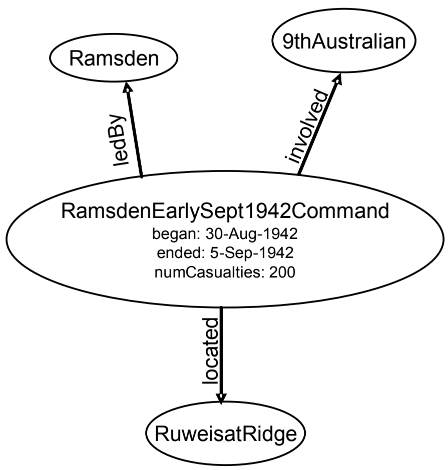

Figure 1. Predicate reification (traditional method). Attributes of an object – i.e., "literals," or "data type properties" – are shown inside that object's bubbles, while relations to other objects are shown as lines connecting bubbles.

There are at least three reasons why this standard solution poses such insurmountable difficulties:

1. It forces the user to model as a noun what is more naturally a verb. In the sentence above, the idea that "Ramsden led the 9th Australian" is an obvious and intuitive way of thinking about an activity in history. Yet in this case the user is forced to artificially recast the knowledge in a form which does not correspond to his or her original thinking pattern.
2. The user must give a name to this awkwardly extracted pseudo-noun, which can be difficult to generate. "RamsdenEarlySept1942Command" as a name is strained at best, yet the alternatives are no better. Occasionally a sensible name presents itself ("Booth shot Lincoln" can become "TheLincolnAssassination"; "Montgomery fought Rommel" becomes "TheBattleOfAlamElHalfa") but this seems to arise only in cases where the action in question is so famed as to have genuinely earned its own moniker.
3. Lastly, this solution makes knowledge refactoring difficult. Suppose a user, in building a knowledge base, initially has no ancillary facts for an event they wish to record. Perhaps "Lt. Gen. Ramsden led the 9th Australian" is all that is needed on that score; the dates and other details are not known at the time, and there is no reason to suppose they will ever be important. But then later, as the knowledge base evolves, it becomes evident that these other details are actually relevant. Ideally this would involve simply adding additional facts, not reworking old facts to make new facts fit. But that is precisely what this solution entails. The user must take a perfectly acceptable fact from the knowledge base and either replace it or duplicate it with this alternate structure. Modeling the verb as a predicate, which worked so well initially, is now suddenly an obstacle to augmenting the knowledge base, which forces an uncomfortable restructuring operation.

3.2 Alternative approach: predicate modifiers

All three of these obstacles can be overcome, however, by equipping the user with an alternate scheme for representing n-ary relations: namely, by allowing them to attach modifying properties to the original (binary) predicate. This scheme was first introduced as a "more natural solution" in conjunction with the Yago ontology[17], though not in the context of user interfaces or ease of knowledge construction by end users. The equivalent expression of the "Ramsden" facts in this new scheme would be thus:

Ramsden led 9thAustralian
  located RuweisatRidge
  began 30-Aug-1942
  ended 5-Sep-1942
  numCasualties 200

(See Figure 2 for a diagrammatic equivalent.) Here, the supplementary facts that modify the verb are positioned under it, indicating that they provide additional information applicable to the "Ramsden led 9thAustralian" statement as a whole.

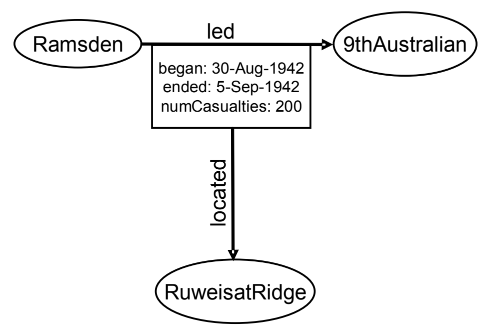

Figure 2. Predicate modifiers (alternate method.)

This seemingly minor change will prove to have a tremendous impact on users' ability to correctly generate semantic data containing n-ary relations.

4. HYPOTHESES

The purpose of our experiment was to come to a conclusion about how well novice users are able to formulate data that includes n-ary relations. In particular, we tested three hypotheses:

- H1 – Without having studied previous examples of statements with properly constructed n-ary relations, novice users would find formulating such statements very difficult, certainly more difficult than binary relations. Predicate reification is a non-trivial concept and is not likely to be intuitive for users.

- H2 – Even after participants are shown an example of predicate reification, and then asked to use that technique to express other knowledge, they would still exhibit poor performance. Some might exploit similarities between an example and another sentence that closely mimicked the example, but for sentences that were structurally dissimilar to the example, participants would have difficulty reproducing predicate reification.
- H3 – On the other hand, participants who are shown examples of the alternate technique of predicate modifiers, and asked to formulate n-ary relations using that technique, would outperform those having only seen, and required to use, predicate reification.

5. THE EXPERIMENT

To test these hypotheses, we composed a written test that would require participants to read short paragraphs (in some cases, single sentences) of English text, and then convert the key knowledge they contained into an RDF-compatible syntax.

5.1 Participants

Our participants consisted of 86 college students ranging from 18 to 22 years of age, with a roughly even split between genders. All of the students were enrolled at the University of Mary Washington during the Fall 2009 semester, and were of many diverse majors. We believe this group is appropriate since it fits a key demographic of potential Semantic Web contributors: young (and hence typically comfortable with technology) and educated (and hence likely to have non-trivial knowledge to contribute.) None of the participants had received any special training in formal knowledge representation, RDF, or the Semantic Web, and therefore we believe that our results are representative of novice users to a substantial degree.

5.2 Procedure and materials

Each participant received a written test with five parts, in this order:

Part I – Practice with binary relations: Knowledge formulation examples and exercises that contained no n-ary relations. Each participant saw two example paragraphs with accompanying "answers" – i.e., RDF-compatible knowledge representations of the knowledge contained in those paragraphs – one in visual format (with bubbles and arrows) and one as a set of textual triples. Each participant was also given two other paragraphs (again without n-ary relations) and asked to provide knowledge representations of them; one visually, and one as triples. The purpose of Part I was to acquaint participants with the concept of expressing knowledge inherent in paragraphs in formalized terms. To give a flavor of the material, one of the paragraphs was:

> "Barack Obama (from Illinois) and Joe Biden (from Delaware) are currently the president and vice president of the United States. They are both members of the Democratic party, an organization currently holding 57 Senate seats, 257 House seats, and boasting about 72 million members nationwide. Obama is a 48-year-old African-American with two children, Malia Ann and Natasha."

Note that none of the information in the above paragraph requires an n-ary relation to express; all facts can be put in the form of binary relations ("BarackObama bornIn Illinois"; "JoeBiden memberOf DemocraticParty," etc.)

Part II – N-ary relation attempt with no examples: The participants were then asked to read and express the following two sentences that do contain n-ary relations:

> (1) Muhammad Ali boxed against Joe Frazier in Detroit.
> (2) General William Barksdale led the Fifth Brigade on May 1st, 1863 in Chancellorsville, Virginia.

Note that these sentences can only be properly represented using n-ary relations, of which at this point no example had been provided. Half of the participants were instructed to express sentence (1) as text triples and sentence (2) as a visual diagram; the other half did the opposite. Part II thus comprised a simple 1x2 experimental design in which the visual mode could be compared against the textual mode, and in which hypothesis H1 could be tested.
After Part II was completed, these materials were collected to ensure that no participant could go back after seeing Part III and change their solutions in Part II.

Part III – N-ary relation example: Participants were presented with an example paragraph, and associated "correct answer," that did contain an n-ary relation: "Ernest Hemingway received the Pulitzer Prize in 1953 for The Old Man and the Sea." Like sentences (1) and (2), n-ary relations are required here, and participants were shown a solution that properly used them. The participants were split into four groups, however, and each saw the solution in a different form:

> Group A – triples, predicate reification
> Group B – triples, indented predicate modifiers
> Group C – diagram, predicate reification
> Group D – diagram, indented predicate modifiers

For the remainder of the test (including parts IV and V) participants remained in these groups. All further work they produced was in the format associated with each group (for example, group A always produced texual triples using predicate reification, and never saw predicate modifiers nor was asked to produce a diagram.)

Part IV – Short n-ary relation attempts: After studying the example solution in Part III, participants were asked to read two more sentences (with n-ary relations) and express them using the technique illustrated:

> (3) Beyonce won an MTV Video Music Award for Best Female Video in 2003 for her video "Crazy in Love."
> (4) On Sept. 23rd, 2009, Joe Williams bought a paperback copy of Harry Potter on Amazon.com with his Visa card (card number 1234567812345678.)

Our rationale behind sentence (3) was its structural similarity to the Hemingway example. Presumably, participants would easily make the connection between all parts of each sentence and mimic the example. Sentence (4) was expected to be more difficult because it was less similar to the example.

Part V – Long n-ary relation attempts: Lastly, a longer paragraph with multiple n-ary relations was given to the participants. This consisted of information about medals won by various swimmers in the 2004 and 2008 Summer Olympic

Games. It contained six n-ary relations, and was intended to be more complex than the individual sentences. This was to discover how well participants could gather the important information within a complex paragraph and correctly use n-ary relations to express it.

6. QUANTITATIVE RESULTS

The Part II sentences were presented before the participants had been given an example of a properly handled n-ary relation. In both cases, the vast majority of participants were unable to express the statement accurately, regardless of whether they used a diagram or textual triples to do so. Only eight participants out of eighty-six (9%) were able to accurately represent sentence (1) (seven as a diagram, one as text.) And only one (1.2%) got a correct answer for sentence (2) (as text). This performance was strikingly poorer than that for the previous exercises that involved only binary relations.

These data clearly indicate that, not surprisingly, it is hopeless to expect novices to formulate n-ary relations without being shown an explicit example of how to reify a verb. Hypothesis H1 was dramatically confirmed.

Sentence (3)	A		B		C		D		**Total**	
Correct	1	(4%)	17	(77%)	5	(25%)	9	(45%)	32	(37%)
Partial	1	(4%)	1	(5%)	2	(10%)	9	(45%)	13	(15%)
Incorrect	20	(83%)	2	(9%)	11	(55%)	1	(5%)	34	(40%)
Invalid	2	(8%)	2	(9%)	2	(10%)	1	(5%)	7	(8%)
Total	24	(100%)	22	(100%)	20	(100%)	20	(100%)	86	(100%)

Sentence (4)	A		B		C		D		**Total**	
Correct	0	(0%)	8	(36%)	2	(10%)	4	(20%)	14	(16%)
Partial	0	(0%)	5	(23%)	2	(10%)	14	(70%)	21	(24%)
Incorrect	20	(83%)	4	(18%)	11	(55%)	1	(5%)	36	(42%)
Invalid	4	(17%)	5	(23%)	5	(25%)	1	(5%)	15	(17%)
Total	24	(100%)	22	(100%)	20	(100%)	20	(100%)	86	(100%)

Figure 3. Scores for responses for each of the four groups on Part IV sentences (3) and (4).

The Part IV sentences, presented to the participants after seeing such an example in Part III (Hemingway), were more promising. We classified responses to these sentences into four categories. Correct responses represented the complete n-ary relation. Partial responses correctly represented an n-ary relation, but at least one fact that should have been present was missing. (For instance, "Beyonce won an MTV Video Music Award for 'Crazy in Love'" omitted 2003). Incorrect responses failed to express an n-ary relation, expressing only binary relations, in such a way that information was lost. Invalid responses failed to represent the information because of invalid syntax. Figure 3 contains a summary of these scores.

Four reviewers independently scored these items in an effort to reduce bias and promote consistency. The "majority vote" for a given item was taken as the score for that item. Note that a double-blind approach was not possible here, since the format used by the participants varied by group (solutions expressed visually were obviously visual; solutions using predicate modifiers obviously had those modifiers.) We remark anecdotally that the level of certainty among reviewers was quite high.

Sentence (3) relates four primary resources: Beyonce, MTV Video Music Award, "Crazy In Love," and 2003. Among the 13 "partial" responses, "Crazy In Love" was by far the most commonly omitted resource (11 times, including all nine participants from Group D.)

Sentence (4) relates five primary resources: Joe Williams, Amazon.com, Harry Potter, Joe Williams' Visa card, and the date 9/23/2009. Here, the resources omitted by the "partial" responses were more scattered, usually omitting Amazon.com, the Visa card, or both.

In Part V, which included six n-ary relations, we categorized each participant's expression of each fact as correct, partially correct, or not correct. A participant's expression of a fact would be considered not correct if it was incorrect, invalid, or omitted entirely. Remarkably, none of the 24 participants from Group A either partially or completely represented any of the six facts. And only one participant from Group C was able to do so. In total, then, for Part V, only one out of 44 participants (2%) in the predicate reification groups produced even a single partially correct result. (For this reason, we do not include these groups further in the Part V analysis.)

	Group B (text)				Group D (diagram)			
	Date Modifier Only		Multiple Modifiers		Date Modifier Only		Multiple Modifiers	
Correct	4	(18%)	8	(36%)	15	(75%)	3	(15%)
Partial	0	(0%)	6	(27%)	0	(0%)	4	(20%)
Not Correct	18	(82%)	8	(36%)	5	(25%)	13	(65%)
Total	22	(100%)	22	(100%)	20	(100%)	20	(100%)

Figure 4. Scores for the predicate modifier groups for two of the n-ary relations in the final paragraph. The first relation involved only a date as a modifier, while the second involved multiple modifiers, some of which had to be object properties (instead of attributes.)

Groups B and D (using predicate modifiers) performed significantly better. Figure 4 compares their results with respect to two of the n-ary relations. (The results from the other four were similar.) One of these n-ary relations contained only a date in addition to the participant and object; the other was more complex.

7. ANALYSIS

7.1 "Chains" and "triangles"

In evaluating the participants' attempts to express n-ary relations, we observed several recurring tendencies. When confronted with n-ary relations, many participants struggled to express the information. Several made no attempt to express sentences (1) and (2) at all, suggesting that they were completely unable to determine a reasonable way to represent the information.

Another problem was more common, however. The Semantic Web requires that each fact (triple) be independently meaningful. Participants often attempted to express n-ary relations in a way that violated this principle. In textual responses, these participants included triples that seemingly depend on previous information, for instance:

Ramsden led 9thAustralian
led near RuweisatRidge

Many participants made a related mistake in diagrams, modeling the entire statement as a linear sequence of connections between the subject, the object, and the direct objects of the original sentence. Figure 5 depicts a prototypical example for the "Ramsden" sentence. We term this type of construction a "chain." It is a valid expression of a complex statement in a concept map[9] because the human viewer can interpret which combinations of triples should be associated together. But a valid RDF graph cannot contain chains because the connections will not represent independent triples.

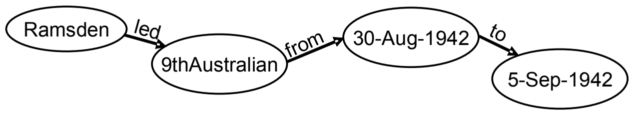

Figure 5. A "chain": an error which violates the principle of RDF that all triples must be independently meaningful.

Another common error was to express only binary relations between three or more parts of a given complex statement. We term this kind of pattern a "triangle":

Ramsden led 9thAustralian
9thAustralian near RuweisatRidge
Ramsden near RuweisatRidge

These assertions are all true, but do not contain all of the information in the original statement. The triples express the facts that Ramsden once led the 9th Australian Brigade, that Ramsden was once near Ruweisat Ridge, and that the 9th Brigade was once near Ruweisat Ridge, but does not express the fact that these were all true *simultaneously*. Hence information is lost. A correct representation would express that Ramsden led the 9th Brigade while the 9th Brigade was near Ruweisat Ridge.

All of these tendencies persisted even after the participants were shown an example of a valid representation of an n-ary relation (but were less frequent).

7.2 Predicate reification: text vs diagram

It is clear from the data in figure 3 that when using the traditional technique of predicate reification to represent n-ary relations, participants were more successful dealing with a visual format than text. Group A (text) struggled to correctly express n-ary relations even after being presented with an example. This suggests that textual predicate reification is non-intuitive for the layman. Sentence (3) was written specifically to resemble the example, but still only 8% of participants were able to even partially represent the n-ary relation. And none succeeded at representing sentence (4) or any part of the final paragraph. Seeing an example of predicate reification was evidently not sufficient instruction to result in significant improvement, confirming hypothesis H2, at least for text. Group C (diagram) was more successful, with a total of 35% of participants able to partially or completely express sentence (3), and 20% able to express sentence (4). A chi-square test of independence between groups A and C yields $\chi2$ (df=1, N=44) = 18.66, $p < .001$, ϕ = .651 for sentence (3) and $\chi2$ (df=1, N=44) = 17.57, $p < .001$, ϕ = .632 for sentence (4), confirming a difference between formats (visual or textual). Therefore, participants were significantly more successful using visual predicate reification than textual reification. Neither technique proved very impressive, however, and we feel that a success rate of only 20-35% even for diagrams confirms H2 in both cases.

7.3 Predicate reification vs predicate modifiers

The experiment revealed a staggering difference between predicate reification and predicate modifiers. (Refer again to figure 3.) In Group B, 82% of participants were partially or completely successful with sentence (3), and 59% with sentence (4). In Group D, 90% of participants were completely or partially successful with sentence (3), and 90% with sentence (4). Comparing the predicate reification groups (A and C) with the predicate modifier groups (B and D), $\chi2$ (df=1, N=86) = 34.12, $p < .001$, ϕ = .630 for sentence (3) and $\chi2$ (df=1, N=86) = 34.66, $p < .001$, ϕ = .635 for sentence (4). It appears that regardless of the paradigm (visual or text), novices were far more likely to construct valid n-ary relations by modifying the predicate rather than reifying it. Therefore, hypothesis H3 was soundly confirmed.

Interestingly, group B participants were more likely to produce a completely correct representation, while group D participants were more likely to be partially correct. Analyzing the specific omissions suggests an explanation for this effect. Empirically, participants using the visual representation were much more successful modifying a predicate with a date than modifying it with relationships to other objects. This could either be because putting attributes in a box adjacent to the relationship line (such as numCasualties in figure 2) is easier than connecting it to another object, or simply because date was used in the example.

The data from Part V (see figure 4) are consistent with this trend. The paragraph's first two facts both consist of a simple statement involving a subject, verb, object, and year. Only 18% of participants from Group B were able to represent either statement correctly, while 75% of participants from Group D represented both. ($\chi2$ (df=1, N=42) = 11.45, $p < .001$, ϕ = .522 for the first; $\chi2$ (df=1, N=42) = 13.70, $p < .001$, ϕ = .571 for the second.) The remaining four facts are more complex, involving several objects as well as attributes. The difference in success rates between groups B and D was not statistically significant in these cases.

We suspect that a strength of the visual technique we used is the ease of including attributes in the box, but that it is considerably more difficult for users to connect the relationship box to other objects. This would explain why group B (text) was somewhat more successful constructing the latter type of relationship.

8. COMPATIBILITY

As mentioned above, predicate modifiers are far easier for novices to employ than predicate reification. Yet in order for them to be a useful method for representing and authoring Semantic Web data, automatic conversion between the predicate modifier format and standard RDF must be possible. In order to facilitate this conversion, we describe a simple, generalized ontology for the representation of n-ary relations. This can be accomplished using three predicates, inspired by the standard reification predicates rdf:subject, rdf:predicate, and rdf:object[6]. Note that the standard reification vocabulary is not appropriate for handling n-ary relations, as argued by Noy and Rector[11]. These predicates all have the rdfs:domain of rdf:Statement. Using reification, the user can refer to the statement described, but not to the fact it represents. For example:

A rdf:subject Ramsden
A rdf:predicate led
A rdf:object 9thAustralian

The node "A" represents an RDF statement. It could be referred to in order to express provenance information, or assert the date the statement was made. However, it cannot be used to further describe the event. In order to express the event in such a way that it may be further modified (i.e., form the basis of an n-ary relation), a different set of properties must be used. For example:

B **nary:subject** Ramsden
B **nary:predicate** led
B **nary:object** 9thAustralian

(Note that the example namespace "nary" is used here for illustrative purposes, to distinguish it from the "rdf" namespace to which the reifcation vocabulary belongs.) These predicates are defined such that the node "B," rather than representing a statement, represents the actual relation between Ramsden and the 9th Division. Information can be added to the relation by referring to B as the event itself.

B began 30-Aug-1942
B ended 5-Sep-1942
B located RuweisatRidge
B numCasualties 200

Now B represents the full n-ary relation. In general, a statement with predicate modifiers can be converted to valid RDF as follows:

subject primaryPredicate primaryObject
predicate1 object1
predicate2 object2

would become:

relation **nary:subject** *subject*
relation **nary:predicate** *primaryPredicate*
relation **nary:object** *primaryObject*
relation predicate1 object1
relation predicate2 object2

This ontology allows n-ary relations to be handled algorithmically, facilitating tools that allow users to express them using more intuitive methods (such as predicate modifiers) than RDF allows.

9. CONCLUSIONS

The construction of n-ary relations in RDF poses a tremendous challenge for novice users. The difficulties can be significantly alleviated, however, by empowering users with a scheme of modifying predicates rather than forcing them to be reified. This appears to be true whether the user is constructing knowledge visually or textually, although both representations seem to have different advantages. In particular, annotating a relationship line with attributes in an adjacent box seems to be an effective method for users, but connecting a relationship to other objects is problematic.

Further research may uncover specific ways in which the visual and textual notations may be improved to increase novice user performance. In any event, designers of Semantic Web editing tools would be wise to consider incorporating a predicate modifier scheme into their user interfaces to facilitate user knowledge construction.

10. ACKNOWLEDGEMENTS

We gratefully acknowledge the contributions of our colleagues Chris Donaher, Debra Hydorn, and David Rettinger in conducting and analyzing this experiment.

11. REFERENCES

[1] Abar, V. M. "An Ontological Approach to Representing Historical Knowledge." Unpublished PhD dissertation, University of British Columbia Department of Electrical and Computer Engineering, 2004.

[2] Anderson, J. R. *Cognitive Psychology and its Implications, (Seventh Edition.)* Worth Pub, 2009.

[3] Auer, S., S. Dietzold, and T. Riechert. "OntoWiki-A Tool for Social, Semantic Collaboration." *Lecture Notes in Computer Science*, 4273: pp. 736-749, 2006.

[4] Berners-Lee, T., J. Hollenbach, K. Lu, J. Presbrey, E. Prud'ommeaux, and M. Schraefel. "Tabulator Redux: Browsing and Writing Linked Data." In *1st Workshop about Linked Data on the Web* (LDOW 2008), April, 2008.

[5] Cristani, M., and R. Cuel. "A survey on ontology creation methodologies." *International Journal on Semantic Web & Information Systems*, vol. 1, no. 2, pp. 49–69, 2005.

[6] Hayes, Patrick. "RDF Semantics," http://www.w3.org/TR/rdf-mt/#ReifAndCont, 2004.

[7] Kalyanpur, A., J. Hendler, B. Parsia, and J. Golbeck. "SMORE-semantic markup, ontology, and RDF editor." Defense Technical Information Center, 2006.

[8] Krötzsch, M., D. Vrandecic, and M. Völkel. "Semantic MediaWiki." In *Proceedings of the 5th International Semantic Web Conference* (ISWC06), pp. 935–942. Springer, 2006.

[9] Novak, J. D, and A. J Canas. "The theory underlying concept maps and how to construct and use them." Florida Institute for Human and Machine Cognition, Pensacola Fl, http://cmap.ihmc.us/Publications/ResearchPapers/TheoryCmaps/TheoryUnderlyingConceptMaps.htm, 2008.

[10] N.F. Noy, W. Grosso, and M.A. Musen, "Knowledge-acquisition interfaces for domain experts: An empirical evaluation of Protege-2000." In *Proceedings of the 12th Internal Conference on Software and Knowledge Engineering*. Chicago, USA, July, 5-7, 2000.

[11] Noy, N. and A. Rector. "Defining N-ary Relations on the Semantic Web." http://www.w3.org/TR/swbp-n-aryRelations, 2006.

[12] Ruiz-Primo, M. A, and R. J Shavelson. "Problems and issues in the use of concept maps in science assessment." *Journal of Research in Science Teaching* 33, no. 6, pp. 569–600, 1996.

[13] Schaffert, S. "IkeWiki: A Semantic Wiki for Collaborative Knowledge Management." In *1st International Workshop on Semantic Technologies in Collaborative Applications*, STICA, 2006.

[14] Sowa, J. F. *Conceptual structures: information processing in mind and machine*. Reading, MA, 1984.

[15] Staab, S., A. Maedche, and S. Handschuh. "Creating metadata for the semantic web: An annotation framework and the human factor." Technical Report 412, Institute AIFB, University of Karlsruhe, 2001. Google Scholar.

[16] Stojanovic, N., A. Maedche, S. Staab, R. Studer, and Y. Sure. "SEAL: a framework for developing SEmantic PortALs." In *Proceedings of the 1st international conference on Knowledge Capture*, pp. 155-162, 2001.

[17] Suchanek, F. M, G. Kasneci, and G. Weikum. "Yago: a core of semantic knowledge." In *Proceedings of the 16th international conference on World Wide Web*, pp. 697-706, 2007.

[18] W.A. Woods, "What's in a link: Foundations for semantic networks," in D. Bobrow and A. Collins (eds.), *Representation and Understanding: Studies in Cognitive Science*, New York: Academic Press, 1975.