Progressive Formalization and Evolving Narrative as Preconditions for Knowledge Reuse

Sidney C. Bailin

Knowledge Evolution, Inc., 1221 Connecticut Ave. NW, STE 3B, Washington DC 20036, sbailin@kevol.com

Abstract. We argue that the forms of knowledge representation familiar from artificial intelligence and software reuse are inadequate for systematic, pervasive knowledge reuse. While formalization has an important role, it is equally important to provide for reusing informal knowledge and gradually migrating from informal to formal.

Keywords. domain model, ontology, knowledge reuse, features, informal knowledge.

1 Introduction

This position paper argues that the forms of knowledge representation familiar to us from the fields of artificial intelligence (AI) and software reuse are inadequate to meet a vision of systematic, pervasive knowledge reuse. In particular, we argue that ontologies barely scratch the surface of the kinds of knowledge that one would, ideally, want to reuse. We argue, more generally, that while formalization has an important role to play in knowledge reuse, it is equally important to provide a technical framework for reusing informal knowledge and for accommodating the gradual migration of knowledge from informal to formal.

2 Two Scenarios

Here are two scenarios, one reflecting the current state of practice, the other suggesting a richer and more integrated environment for knowledge reuse:

2.1 Scenario 1

David, a software engineer at Better Solutions, Inc., (BSI) has been tasked with evaluating a commercial off-the-shelf (COTS) web services framework named WSPR. The management of BSI have heard that service oriented architecture (SOA) is the wave of the future [1]. Dave knows about SOA and has heard of WSPR but has never worked with it. Nor have any of his close colleagues. So he does what anyone in such a situation would do in order to learn more: he types WSPR into Google. Among the top results returned are:

- The home page for WSPR, with subpages for technical specifications, user documentation, and a couple of white papers
- Several third-party sites that offer solutions based on WSPR
- A Wikipedia page on WSPR, with an official caution that the page may need revision because the information is contested
- User experience reports on various Question and Answer web sites
- Threads on technical aspects of using WSPR in various online forums
- Job postings for WSPR experts

Dave visits the WSPR home page first, and navigates to some of the descriptions offered. He gets little from the material other than “SOA is good for web applications” and “WSPR is good for SOA.” So he tries the Wikipedia page. This page has been written by developers who favor a RESTful approach to web application development and who appear to be biased against state-based frameworks like WSPR. So he moves on to read some of the user experiences. Half of them report success, half are less positive, but what is worse is that they seem to contradict each other on specifics. Dave proceeds to the more technical forums. The questions are mostly from new users who are struggling with deployment options; the replies are from more experienced users who provide detailed answers, such as the particular tags required in the webinf/web.xml file when running WSPR in conjunction with Apache’s Tomcat server.

Ultimately, Dave reports back that WSPR may well be useful to BSI but that it may not be easy to get it working. BSI places an order for WSPR, and at the same time posts a job requisition for a WSPR expert, emphasizing the requirement that applicants have at least 3 years of direct experience building applications with WSPR in the particular environment — platforms, operating systems, database management systems, and networking — already in use at BSI. They receive many responses, but none fully matches this requirement.

2.2 Scenario 2

The setting is the same: Dave, BSI, and WSPR. But this time the Google search highlights a semantic wiki [2] devoted to web service architectures and frameworks. From the wiki’s main page, Dave easily finds a page on WSPR. That page, in turn, points to several story lines concerning WSPR:

- What is WSPR?
- Relation to other SOA frameworks
- Vendor claims
- User experience
- It’s great — here’s why
- It’s horrible —here’s why
- Things you might want to consider

Hyperlinks connect much of this information within the wiki. For example, the user experience pages reference various vendor claims. Since this is a *semantic* wiki, there is ontological information residing in the background of these narratives, including:

- Taxonomy of SOA frameworks
- Taxonomy of user experiences
- Taxonomy of risks
- Taxonomy of common tasks

Of course, an ontology is more than a taxonomy [3]. Classes have instances, and there are relationships between instances of various classes. For example, the user-experience pages are tagged as instances of the taxonomy’s user-experience classes. They, in turn, are linked to instances of risks, to vendor claims, and to common tasks. Dave is impressed with the taxonomies of SOA frameworks and risks: clearly people have been hard at work developing a domain model not just of SOA but of SOA frameworks. He is even more impressed with the taxonomy of user experiences and common tasks, as he has never before encountered that form of modeling, which digs deeper into the nuts and bolts of practical software development.

Unlike the miscellaneous web pages in Scenario One, the semantic wiki shows Dave *why* the different user experiences are, in fact, different: they involved different tasks, or different environments, or differently skilled practitioners. The risk model, in particular, is culled from all of these experiences and represents a relatively neutral picture of the benefits and pitfalls of WSPR. Dave wonders who has done all this work, and who is responsible for maintaining it. Navigating back to the WSPR main page, and then to the wiki main page, he finds guidelines for contributing to the wiki. These include guidelines for posting information, experience, or analysis that appears to conflict with existing information.

Getting back to his task, Dave is able to make specific recommendations to BSI management about WSPR. He is confident about being able to work with WSPR using the community knowledge base he has found. Thus, no job posting is necessary. BSI gradually migrates its transactional applications to WSPR, and Dave becomes an active participant in that portion of the semantic wiki.

3 Analysis

The salient differences between the two scenarios are:

- Integration — all of the information is linked and *related* to all other relevant information
- Formalization — the accumulated knowledge has been abstracted into classes, instances, and relationships
- Narrative — the abundance of knowledge has been culled into a small number of stories, each of which has a primary message or throughline

Let us consider why each of these is important. The virtues of integration are familiar to us, at some level, already today. The abundance of links in the World-Wide Web allow us to discover information that might otherwise elude us. Scenario Two, however, illustrates a far stronger form of integration, which lies in the explicit relationships between linked pages. For example, let us suppose there are two pages describing WSPR. Do they reinforce each other? *Why?* Do they contradict each other? *Why?* Do they compliment each other? *How?* If there were only one page on WSPR, the knowledge to be reused would simply be the content of that page. Otherwise, *the real value resides in the "Why?" "Why?" and "How?"* Those answers constitute the knowledge to be reused, even more than the page contents themselves, because without those answers the pages collectively become an undifferentiated mass of data.

Formalization is a key to achieving this deeper level of integration. To see this, consider one of the questions that integration answers: Why do two experience reports on WSPR contradict each other? Suppose one page recommends a particular way to deploy WSPR, while the other reports a failed attempt to do just that. Obviously there is something different in the two cases. Perhaps the second case did not configure WSPR exactly the same way as the first case. Perhaps the environment was slightly different. Or perhaps the goals or success criteria were different, or the skill levels of the developers were different. If such information is tacit, then the *Why?* question is not answered. But even if explicit, if it is buried somewhere in the text, the answer may still not be readily apparent. If, however, the similarities and differences of the two cases are modeled ontologically, then the answer becomes both explicit and exposed, thus: The cases are not, in fact, contradictory, but rather they are distinct in respect to particular properties modeled in the ontology. These distinctions explain the difference in outcome.

But formalization without narrative is not terribly useful. A complex ontology is almost impossible to navigate without commentary. At the very least, a narrative tells us *Start here*. Better would be: *Start here in order to accomplish a particular goal.* It would then lead us through the taxonomic distinctions — Why do they exist? What is their significance? — and the properties of each class — Why are they there? This is a narrative because each explanation tells us, *We solved* this *problem* this *way*. In other words, the narrative is a succession of problems and their solutions, starting from an initial context and leading to an eventual goal. That is the basic structure of a story [4].

In summary, Scenario Two improves over Scenario One by turning a morass of information (uncoordinated pages all over the web) into a set of stories whose crucial turning points are plotted in a formal ontology. That, we suggest, is the optimal structure for knowledge reuse.

4 Curatorship and Evolution

Underlying the differences between the two scenarios — integration, formalization, and narrative — is a difference in *process*. The contents of the semantic wiki are subject to a community curatorship, in contrast to the anarchy of the world-wide web at large as reflected in Scenario One.

The knowledge base must contain both formal and informal information. To see why this is so, let us consider the different kinds of knowledge that appear in the two scenarios. We can identify the following kinds of knowledge:

- Taxonomic
- Structural
- Relational
- Propositional
- Procedural

We have already identified several taxonomies, or class hierarchies, present in the semantic wiki and implicit in the Scenario One. Structural knowledge, consisting of whole-part relationships, would likely occur in the technical descriptions of WSPR, and possibly also in the descriptions of common tasks (if they are decomposed into subtasks). Relational knowledge is embodied in the links mentioned in Scenario Two, and is surely present also in Scenario One in the form of any hyperlinks that exist between the pages.

Ontologies, in their current manifestation as encoded in the Web Ontology Language (OWL), are good for representing these three types of knowledge. In substance, they are not much of a breakthrough: they simply provide a common representation for the kinds of knowledge familiar to us from various domain modeling methods.

Ontologies are less suited for expressing propositional and procedural knowledge. Procedural knowledge is knowledge about how to accomplish a goal. It includes programming and algorithmic knowledge, but also plans and strategies. While one can argue about the best formalisms in which to express such knowledge, we hope it is not controversial to say that ontologies are not especially well suited. But reasonable formalisms do exist: certainly for the kinds of procedural knowledge we would expect to see in our scenarios.

What is not reasonable is to expect ordinary users to employ OWL when posting taxonomic, structural, or relational knowledge. Taxonomic knowledge in the scenarios appears as distinctions between cases. These distinctions may be implicit or explicit, but they will rarely be posted in the form of taxonomic structures. Nor should we expect users to post procedural knowledge in a procedural formalism; at best, some kind of pseudocode might be used to achieve clarity. This begins to illustrate the need for curators; but the need becomes clearer when we consider propositional knowledge.

Propositional knowledge consists of assertions. It thus subsumes taxonomic, structural, and relational knowledge — all of which can be expressed in OWL as RDF triples, i.e., subject-predicate-object, hence as assertions — but it is much broader. For example, propositional knowledge includes *n*-ary relations, not just the binary relations of RDF triples. More significantly, propositional knowledge includes logic, i.e., quantifiers. Furthermore, propositional knowledge includes various kinds of logic assertion, e.g., higher-order [5], modal [6], temporal [7], and fuzzy [8] or other multi-valued logic [9], and, if one really wants to be inclusive, other forms of logic such as deontic (the logic of obligations) [10], defeasible (the logic of refutation) [11], and abductive logic (reasoning to causes) [12].

It is not difficult to imagine at least some of these non-standard logical assertions appearing in our scenarios. For example, a discussion of risks is likely to contain informal modal assertions (indicating that an undesirable outcome *might* occur) or fuzzy or probabilistic assertions. This is true also for discussions of design tradeoffs and rationale. Knowledge about troubleshooting could easily involve informal defeasible reasoning (ruling out explanations) or abductive reasoning (arriving at an explanation). We could even imagine deontic logic implicitly underlying a discussion of software contracts.

Posters to a semantic wiki on SOA frameworks are not going to use formal versions of these logics in order to express such knowledge. But we might imagine the curators of a semantic wiki drawing out some formal assertions along these lines from the informal experience and expertise that has been posted — *if* the underlying formal mechanism (the semantic part of the semantic wiki) permitted it. This would require a language richer than OWL. The expanded richness might entail a reduction in computational tractability; but here, ironically, we intend formalization not for purposes of computation, but rather for human consumption, as concise distillations of voluminous discussions. The formal entities are the plot points of the narrative.

Curators are required in order to cull the underlying structures from the ongoing discussions and represent them formally. This is progressive formalization of evolving knowledge, and it suggests, as well, an evolution path from Scenario One to Scenario Two. Because formalism for human consumption requires narrative, the narratives too must evolve. While the most proficient or literary posters might develop these narratives themselves, this cannot be assumed. Curators can and should be members of the community. Widely used wikis and open source projects have shown how this can work.

5 Conclusion

Through two scenarios — one representing the state of current practice, the other an imagined future — we have illustrated how knowledge reuse requires a dual kind of semantic richness, comprising both formalism and narrative. We have shown, further, how both of these must be extracted and abstracted from the raw knowledge posted by members of a knowledge community. This means that neither the underlying formal structures nor the explanatory narratives are fixed; rather, they evolve. Effective knowledge reuse requires an infrastructure and processes to support such evolution.

References

[1] Erl, T. Service-Oriented Architecture: Concepts, Technology, and Design. Prentice Hall, 2005.
[2] http://semwiki.org
[3] Allemang, D., Hendler, J. Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL. Morgan Kaufmann, 2008.
[4] Herman, D., Jahn, M., Ryan, M-L. (eds) The Routledge Encyclopedia of Narrative Theory. Routledge, 2007.
[5] Lambek, J., Scott, P. Introduction to Higher-Order Categorical Logic. Cambridge University Press, 1988.
[6] Cocchiarella, N., Freund, M. Modal Logic: An Introduction to its Syntax and Semantics. Oxford University Press, 2008.
[7] Kröger, F., Merz, S. Temporal Logic and State Systems. Springer, 2008.
[8] Tanaka, K., Niimura, R. An Introduction to Fuzzy Logic for Practical Applications. Springer, 2007.
[9] Schumann, A., Smarandache, F. Neutrality and Multi-Valued Logics. A.R. Press, 2007.
[10] Horty, J. Agency and Deontic Logic. Oxford University Press, 2009.
[11] Antonelli, A. Grounded Consequence for Defeasible Logic. Cambridge University Press, 2005.
[12] Josephson, J., Josephson, S. Abductive Inference: Computation, Philosophy, Technology. Cambridge University Press, 1996.