Towards a Reverse Engineering Ontology

Bart Du Bois
Lab On REengineering (LORE)
Dept. of Mathematics-Computer Science
University of Antwerp, Belgium
Bart.DuBois@ua.ac.be

Abstract

In a review of existing reverse engineering literature, it became clear that there is no shared vision on key topics in reverse engineering. As a result, the many research contributions appear to lack uniformity with regard to the way results are reported.

In this short paper, we present "the Reverse Engineering wiki"[1], an initiative started at ICPC 2006 in the working session on Experimental Settings in Program Comprehension. This wiki allows reverse engineering researchers and practitioners to collaborate freely on a demarcation and description of the field of reverse engineering.

First, we address the need for this initiative. Second, we advocate the use of an ontology to satisfy this need. We present our current results in implementing this solution and describe valuable usage scenario's. Finally, we enlist key activities in the continuation of this initiative, stressing that collaboration will be a key success factor in this initiative.

1 Introduction

Recently, we reviewed the existing reverse engineering literature in an attempt to characterize the state of of the art in empirical reverse engineering studies [Tonella et al., 2006]. We found it remarkably difficult to position research contributions in the whole of reverse engineering. Most often, the reviewed research papers tended to focus their discussion to a particular analysis technique, without clarifying its relationship to the problem of reverse engineering in general. As a result, it is hard to index and retrieve knowledge on existing research contributions to reverse engineering. E.g., we have found it difficult to:

- relate different research contributions.
- create an overview of literature addressing similar issues.
- compose reusable evaluation methods for alternative solutions to a single problem.
- devise check lists for reviewing reverse engineering papers.

In this short paper, we discuss an initiative targeted at facilitating the indexing and retrieval of knowledge about research contributions in the domain of reverse engineering. The definition of reverse engineering used in this paper is reused from [Tonella et al., 2006]:

> "*Every method aimed at acquiring knowledge about an existing software system in support to the execution of a software engineering task.*"

2 Facilitating Indexing and Retrieval of Research Contributions

An *ontology* is a hierarchically structured body of knowledge about things, composed by subcategorizing them according to their essential (or at least relevant and/or cognitive) qualities [Howe (Editor), 2005]. In this paper, we propose the composition of a reverse engineering ontology as a means to "*provide a common, referencable set of concepts for use in communication*", which is the main goal of an ontology [van Rees, 2003].

The difference between an ontology and a taxonomy is clarified by [van Rees, 2003]. A taxonomy classifies items based upon their relationships, typically in a hierarchical manner. An ontology also classifies items, but additionally provides detailed information about the items and their relationships. Typically, ontologies organize these items in classes and subclasses in a hierarchical manner.

The main criterion for the structural organization of such an ontology is that it should clarify how different research contributions are related to each other. I.e., the ontology should allow to recognize (i) whether different solutions are

[1] http://lore.cmi.ua.ac.be/reWiki

addressing the same problem; and (ii) which characteristics of the solution are shared between alternative solutions.

The structural hierarchy introduced in [Tonella et al., 2006] satisfies this criterion, differentiating between three levels: methods, techniques and tools. *Methods* are general classes of solutions to known problems. *Techniques* are specific realizations of methods, based on particular algorithms, assumptions and approaches. At the lowest hierarchical level, *tools* are implementations of techniques.

The hierarchical organization in methods, techniques and tools facilitates evaluations of how different research contributions are related. Reverse engineering solutions addressing the same problem would be described as different techniques or tools within the same reverse engineering method. By comparing these alternative techniques, shared characteristics can be easily recovered. Thus, we propose to structure the body of knowledge concerning reverse engineering using by identifying and characterizing methods, techniques and tools.

3 Towards an Ontology

In [Du Bois, 2005], we summarized the construction of an ontology as consisting of the following steps: (i) *specification* of the purpose, usage, scope and degree of formality of the ontology; (ii) *data collection* using various elicitation methods; (iii) *conceptualization* of domain terms, resulting in a preliminary ontology; (iv) *integration* with other ontologies; (v) *formalization* in an ontology language; and (vi) *evaluation* of completeness, consistency and redundancy.

As initial steps towards the construction of a reverse engineering ontology, a website has been developed, entitled *the Reverse Engineering Wiki*[2]. This website has been the result of initial conversations started at ICPC 2006 in the working session on Experimental Settings in Program Comprehension. Through informal conversations, the steps of specification, data collection and conceptualization have been initiated.

3.1 The Reverse Engineering wiki

The Reverse Engineering wiki presents a repository in which reverse engineering methods, techniques and tools can be described. In an initial phase, we will focus on a breadth-first description. This strategy stresses the description of problems and their essential solution classes before detailed assumptions and algorithms are taken into account. In an initial attempt to ensure a uniform description of reverse engineering methods, we have proposed the following template:

1. **Problem –** This part of the template focuses on the essence of the knowledge acquisition problem for which the method defines a particular class of solutions.
   (a) **Context –** In which context does the problem arise? What are the key characteristics of the problem? Why is the problem relevant? These questions help to situate the problem in the larger context of software engineering.
   (b) **Lacking knowledge –** Which knowledge is lacking or not readily available? The answer to this question helps characterizing the goal of the reverse engineering method, namely to acquire knowledge.
   (c) **Evaluation criteria –** Which criteria can be used to evaluate the results of solutions that claim to acquire the lacking knowledge? This aspect focuses on non-functional aspects of the knowledge acquisition, both with regard to properties of the end-result as of the acquisition process itself.
   (d) **Known usage of lacking knowledge**
2. **Solution –** This part of the template focuses on the essence of the knowledge acquisition solution that is common to realizations of the reverse engineering method.
   (a) **Key to the solution –** What is the key to the solution of the problem, and is therefore common between alternative solutions? The answer to this question opens up the search for alternative solutions by focusing on the minimal set of solution characteristics, thereby clarifying the degrees of freedom in alternative realizations of the method.
   (b) **Factors enduring acquisition**
   (c) **Common process definition –** Which phases can be typically distinguished in the acquisition of the lacking knowledge? This question allows to decompose the knowledge acquisition problem in a sequence of more fine-grained problems, i.e. subclasses of the main class of solutions.
   (d) **Towards techniques –** Which assumptions can be made about the acquisition of the lacking knowledge, and which techniques incorporate these assumptions? In this aspect, the bridge to techniques realizing the reverse engineering methods is made.

3.2 Envisioned Usage Scenario's

We envision the use of the Reverse Engineering wiki as an ontology of reverse engineering methods, techniques and tools in at least the following three scenario's.

[2] http://lore.cmi.ua.ac.be/reWiki

1. *Introduction to the domain of reverse engineering* – Researchers and practitioners new to the domain can study the set of reverse engineering methods to grasp the set of problems addressed in reverse engineering.

2. *Expansion of the set of realizations of reverse engineering methods* – By clarifying the intrinsic properties of the class of solutions to known problems, alternative variants of current techniques will become apparent.

3. *Support for empirical studies* – The description of key evaluation criteria for realizations of reverse engineering methods can guide researchers in evaluating their techniques and tools in a standard format.

   Across research contributions, the enumeration of a set of techniques addressing the same problem allows to identify the maturity of research on particular problems in the domain of reverse engineering. This identification is essential to the evaluation of the community's readiness for introducing benchmarks [Sim et al., 2003].

4 Future Work

The most critical next step is the expansion of the set of reverse engineering methods described in the ontology. Complementary, the quality of these descriptions needs to be assured, by evaluating typical ontology criteria as (i) completeness; (ii) validity; (iii) clarity; and (iv) redundancy. The implementation of the ontology in a wiki platform minimizes the effort for correcting such quality deficiencies.

We sincerely hope that this initiative will enjoy the participation of an enthusiastic community of reverse engineering researchers.

5 Acknowledgements

We would like to thank the researchers participating in discussions on this initiative.

References

[Du Bois, 2005] Du Bois, B. (2005). Towards an ontology of factors influencing reverse engineering. In *STEP '05: Proceedings of the 13th IEEE International Workshop on Software Technology and Engineering Practice*, pages 74–80, Washington, DC, USA. IEEE Computer Society.

[Howe (Editor), 2005] Howe (Editor), D. (2005). The free on-line dictionary of computing. http://www.foldoc.org.

[Sim et al., 2003] Sim, S. E., Easterbrook, S., and Holt, R. C. (2003). Using benchmarking to advance research: a challenge to software engineering. In *ICSE '03: Proceedings of the 25th International Conference on Software Engineering*, pages 74–83, Washington, DC, USA. IEEE Computer Society.

[Tonella et al., 2006] Tonella, P., Torchiano, M., Du Bois, B., and Systä, T. (2006). Empirical studies in reverse engineering: State of the art and future trends. Submitted to Empirical Software Engineering: An International Journal.

[van Rees, 2003] van Rees, R. (2003). Clarity in the usage of the terms classification, taxonomy and ontology. In *Proceedings of the 20th International Conference on Information Technology for Construction*.