Chapter 8.17
Benchmarking in the Semantic Web

Raúl García-Castro
Universidad Politécnica de Madrid, Spain

Asunción Gómez-Pérez
Universidad Politécnica de Madrid, Spain

ABSTRACT

The Semantic Web technology needs to be thoroughly evaluated for providing objective results and obtaining massive improvement in its quality; thus, the transfer of this technology from research to industry will speed up. This chapter presents software benchmarking, a process that aims to improve the Semantic Web technology and to find the best practices. The chapter also describes a specific software benchmarking methodology and shows how this methodology has been used to benchmark the interoperability of ontology development tools, employing RDF(S) as the interchange language.

INTRODUCTION

The Semantic Web technology has considerably improved since the 1990's, when the first tools were developed; although it has mainly been applied in research laboratories, in recent years companies have started to be interested in this technology and its application.

To transfer the Semantic Web technology from the academia, its current niche, to the industrial world it is necessary that this technology reaches a maturity level that enables it to comply with the quality requirements of the industry. Therefore, the Semantic Web technology needs to be thoroughly evaluated both for providing objective results and for attaining a massive improvement in its quality.

Until recently, the Semantic Web technology was seldom evaluated; now, however, this technology is widely used and numerous studies concerning its evaluation have appeared in the last few years. So now it seems quite necessary that researchers increase the quality of their evaluations and improve the technology collectively by benchmarking it, employing for this a methodological process.

Evaluating and benchmarking this technology within the Semantic Web can be quite costly because most of the people involved do not know how to carry out these processes and also because no standard nor agreed methods to follow now exist. On the other hand, since it is quite difficult to reuse the results and put into practice the lessons learnt in previous activities, it is necessary to develop new methods and tools every time this technology has to be evaluated or benchmarked.

Software benchmarking is presented in this chapter as a continuous process whose aim is to improve software products, services, and processes by evaluating and comparing them with those considered the best. Although software evaluations are performed inside the benchmarking activities, benchmarking provides some benefits that cannot be obtained from evaluations, as for example, the continuous improvement of software, or the extraction of the best practices used to develop the software.

Within the Knowledge Web[a] European Network of Excellence a new methodology for benchmarking Semantic Web technology has been developed; this methodology is now being adopted in different benchmarking studies and applied to the different types of Semantic Web technologies (ontology development tools, ontology alignment tools, ontology-based annotation tools, and reasoners). The methodology focuses on the special interests of the industry and research fields and on their different needs. At the end of the chapter, we describe how we have followed this methodology during one of the activities performed to benchmark the interoperability of ontology development tools, employing RDF(S) as the interchange language.

EVALUATION AND BENCHMARKING IN THE LITERATURE

Software Evaluation

Software evaluation plays an important role in different areas of Software Engineering, such as Software Measurement, Software Experimentation or Software Testing. In this section, we present a general view of these areas.

According to the ISO 14598 standard (ISO/IEC, 1999), software evaluation is *the systematic examination of to which extent an entity is capable of fulfilling specified requirements*; this standard considers software not just as a set of computer programs but also as a set of procedures, documentation and data.

Software evaluation can take place all along the software life cycle. It can be performed during the software development process by evaluating intermediate software products or when the development has finished.

Although evaluations are usually carried out inside the organisation that develops the software, other independent groups such as users or auditors can also make them. When independent third parties evaluate software, they are usually very effective, though their evaluations can become very expensive (Rakitin, 1997).

The goals of evaluating software vary since they depend on each specific case, but in general, they can be summarised (Basili et al., 1986; Park et al., 1996; Gediga et al., 2002) as follows:

- To **describe** the software in order to understand it and establish baselines for comparisons.
- To **assess** the software with respect to some quality requirements or criteria and determine the degree of quality required from the software product and its weaknesses.