

# Web Operational Analysis Through Performance-Related Ontologies in OWL for Intelligent Applications

Isaac Lera, Carlos Juiz, and Ramon Puigjaner

University of Balearic Islands,  
Carretera de Valldemossa, km. 7,5, 07122 Palma de Mallorca, Spain  
{cjuiz,putxi}@uib.es  
<http://www.uib.es>

**Abstract.** Web performance engineering techniques are classically based on the augmented description of the model regarding performance annotations. However, these annotations are only related with the syntactical view of the system. The next generation of performance assessment tools for intelligent systems would be capable of acquiring knowledge and even reasoning about performance as the systems will work. The use of ontologies with performance-related information may be used to build on-line performance brokers that assess the performance of the system during its execution. We present the web operational analysis as an example of its future utilization by a framework for building intelligent applications based on semantic web.

## 1 Performance-Related Ontologies in OWL

There are several issues to be considered when defining a framework for the performance assessment of intelligent applications that seems to be similar to traditional software performance engineering techniques: (i) It must be decided about the way the intelligent system is modelled and therefore, how to add the performance-related information into the software specification with the minimal interference; (ii) Once the performance aspects of the system are depicted in the model, how to transform the architectural options onto performance models and finally; (iii) the way to evaluate every performance model derived from previous steps. This strategy does not consider the semantic representation of the information on the model. On the other hand, an ontology is an explicit formal description of concepts in the domain composed of classes, properties of each class, and restrictions on properties [6]. OWL (Web Ontology Language [2]) ontologies express the set of terms, entities, objects and classes and also the relations among them with formal definitions. Therefore, performance-related information may be also declared through this new web engineering approach, not only for performance evaluation of the different components of the system, but also in scenarios where it is possible to reason about the performance activity in an intelligent application environment and even take actions based on it.

## 2 Annotation of Performance-Related Information in OWL

An ontology may consider performance-related information as description of a performance view of the application as measurable parameters. However, these performance parameters have been discovered from the relationship among several entities:

client-web browser utilization or process-cpu demand, etc. This example shows the OWL classes involved in the description of the performance-related information model:

```
<owl:Class rdf:ID="PerformanceMeasure"/>
<owl:Class rdf:ID="Client"/>
<owl:Class rdf:ID="Device"/>
```

Each client or device class member possess different functional or non-functional attributes, however, we are mainly interested on computing certain operational values that are observed during a time interval for the ulterior performance analysis of the application. Clearly, it is possible to reuse existing temporal ontologies which consist on a good example of the ontology information sharing. The response time of a transaction, the service time of a process or even the interval observation time are parameters belonging to the time domain. Therefore, the efficiency of the ontological representation resides on using specific classes for time units, i.e. the OWL-Time ontology in [3] and [5]. This example shows a simple performance object property in OWL, the response time, reusing the time ontology:

```
<owl:FunctionalProperty rdf:ID="ResponseTime">
  <rdf:type
    rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
  <rdfs:range rdf:resource="http://www.isi.edu/~pan/damlttime/time-
    entry.owl#DurationDescription"/>
  <rdfs:domain rdf:resource="#PerformanceMeasure"/>
</owl:FunctionalProperty>
...
<ResponseTime>
  <time-entry:DurationDescription rdf:ID="BusyTimeID1">
  <time-entry:seconds
    rdf:datatype="http://www.w3.org/2001/XMLSchema#decimal">
    20.0
  </time-entry:seconds>
  </time-entry:DurationDescription>
</ResponseTime>
```

### 3 Working with a Performance-Related Ontology in OWL

Once the OWL code has been implemented it is necessary to build some external application (broker) for interpreting, computing and updating performance values on instantiations of clients and devices [7]. Jena [1], is a Java framework for building Semantic Web [4] applications that support inference engines over OWL. We built a tool to operate with the performance-related ontology in order to compute interesting performance magnitudes taken from the operation of web server monitorization. This simple application emulates the future broker implementation which will provide information for ulterior reasoning. Figure 1 shows three instantiations of web clients and two web servers on the ontology. The performance values have been obtained during benchmark operation of the web server *deviceID2*. From this information is possible to compute the throughput of the device applying operational analysis by the broker at any instant. The result of the analysis could be take actions on clients or devices whether the intelligent application design provides this feature. For example, it should be used to compare the computed average throughput with the maximum number of transactions per second (tps) that the device supports and take actions when the application will reach to overload the quality of service required.

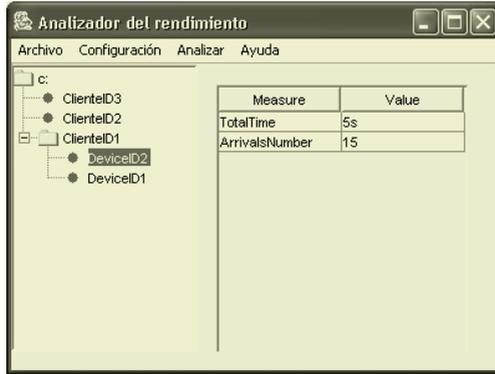


Fig. 1. Performance broker-emulator during operation

## References

1. Jena, Java framework for building Semantic Web applications <http://jena.sourceforge.net/>
2. OWL, Web Ontology language, <http://www.w3.org/TR/owl-features/>
3. DAML Ontology of Time, <http://www.cs.rochester.edu/~ferguson/daml/>
4. Berners-Lee, T., Hendler, J. and Lassila, O.: The Semantic Web. Scientific American, May (2001)
5. Hobbs, J.R., Pan, F.: An Ontology of Time for the Semantic Web. ACM Transactions on Asian Language Information Processing, Vol. 3, No. 1, March (2004) 66-85
6. Mesina, E.R., Meystel, A.M. (eds.): What is the Role of Ontology in Performance Evaluation? Panel discussion in Proceedings of 2002 PerMIS Workshop (2002)
7. Haring, G., Juiz, C., Kurz, C., Puigjaner, R., Zottl, J.: Framework for the Performance Assessment of Architectural Options on Intelligent Distributed Applications. In Proceedings of the Conference on Performance Metrics for Intelligent Systems, Gaithersburg (2005)