Ontology-Based Photogrammetric Survey in Underwater Archaeology

Pierre Drap[1(✉)], Odile Papini[1], Jean-Chrisophe Sourisseau[2], and Timmy Gambin[3]

[1] Aix-Marseille Université, CNRS, ENSAM, Université De Toulon, LSIS UMR 7296, Domaine Universitaire de Saint-Jérôme, Bâtiment Polytech, 13397 Marseille, France
{Pierre.Drap,Odile.Papini}@univ-amu.fr

[2] Aix-Marseille Université, CNRS, Ministère de la Culture et de la Communication, CCJ UMR 7299, 13094 Aix En Provence, France
Jean-Christophe.Sourisseau@univ-amu.fr

[3] University of Malta, Msidi, Malta
Timmy.Gambin@um.edu.mt

Abstract. This work addresses the problem of underwater archaeological surveys from the point of view of knowledge. We propose an approach based on underwater photogrammetry guided by a representation of knowledge used, as structured by ontologies. Survey data feed into to ontologies and photogrammetry in order to produce graphical results. This paper focuses on the use of ontologies during the exploitation of 3D results. JAVA software dedicated to photogrammetry and archaeological survey has been mapped onto an OWL formalism. The use of procedural attachment in a dual representation (JAVA - OWL) of the involved concepts allows us to access computational facilities directly from OWL. As SWRL The use of rules illustrates very well such 'double formalism' as well as the use of computational capabilities of 'rules logical expression'. We present an application that is able to read the ontology populated with a photogrammetric survey data. Once the ontology is read, it is possible to produce a 3D representation of the individuals and observing graphically the results of logical spatial queries on the ontology. This work is done on a very important underwater archaeological site in Malta named Xlendi, probably the most ancient shipwreck of the central Mediterranean Sea.

Keywords: Underwater archaeology · Photogrammetry · Ontology · JAVA

1 Introduction

Recent developments in computer vision and photogrammetry, make the latter technique a near-ideal tool, or at the very least an essential one, for archaeological survey. In underwater context, it is undeniably a must as there is no efficient alternative.

The main idea of this project is based on the fact that the survey, whether it takes place in the scope of underwater archaeology, relies on a complex well-established corpus, even though it evolves over time. A formalisation of the archaeological knowledge involved, is used to guide the survey.

E. Blomqvist et al. (Eds.): ESWC 2017 Satellite Events, LNCS 10577, pp. 3–6, 2017.
https://doi.org/10.1007/978-3-319-70407-4_1

The photogrammetry survey carried out is based on an original approach of underwater photogrammetry that was deployed with the help of a specific instrumental infrastructure provided by COMEX, a partner in the GROPLAN[1] project [1]. This photogrammetry process, as well as the body of surveyed objects, were formalized in an ontology expressed in OWL2. Our approach is based on procedural attachment; the ontology being seen as a dual of the JAVA class structure that manages the photogrammetric survey and the measurement of artefacts. This allows the establishment of a reasoning for the ontologies as well as intensive calculations using JAVA programming language with the same interface. Furthermore, the ontology used to describe the archaeological artefacts from a measurement point of view is aligned with CIDOC-CRM ontology used for museo-graphical objects [2, 3].

The focus of this experimental project is the *Xlendi* shipwreck, named after the place where it was found off the Gozo coast in Malta. The shipwreck was located by during an offshore survey in 2008. The shipwreck is located near a coastline known for its limestone cliffs that plunge into the sea and whose foundation rests on a continental shelf at an average depth of 100 m below sea level. The shipwreck itself is therefore exceptional; first due to its configuration as well as its state of preservation which is particularly well-suited for our experimental 3D modelling project. The examination of the first layer of amphorae also reveals a mixed cargo, consisting of items from Western Phoenicia and Tyrrhenian-style containers which are both well-matched with other archaeological excavations from the Archaic, that is between the end of the VIII and the first half of the VII centuries BC. The historical interest of this wreck, which has been highlighted by our work, is the first to be performed on this site and creates real added-value in terms of innovation. In turn, this contributes to the international reputation of the project.

2 The Use of Ontologies

The ontology has been developed to represent the photogrammetric process used for the survey, the process identification, measure and representation of visible archaeological objects. The final ontology is, on the one hand, an ontology built from a JAVA program modeling the entire photogrammetric process. On the other hand, we also present an ontology describing the archaeological artefacts from the point of view of the photogrammetric measure.

Our goal is to link the measured artefacts with all the observations used to measure and identify them. One of the main advantages of the photogrammetric process is to provide several 2D representations of the measured artefacts. We build this ontology from an existing JAVA code in order to represent the concepts used in photogrammetry and to be able to use a reasoner on ABox photogrammetric data. Initial mapping from an Object Oriented (OO) formalism to a Description Logic (DL) is relatively easy according to the fact that we have to map a poor semantic formalism toward a richer one [4]. In order to do this, we need to manage both the computational aspects (often heavy in photogrammetry) implemented in the artefacts and measurable by

[1] http://www.groplan.eu.

photogrammetry and the ontological representation of the same photogrammetric process and surveyed artifacts.

Our implementation is based on a double formalism, JAVA, used for computation, photogrammetric algorithms, 3D visualization of photogrammetric data and patrimonial objects, and OWL for the definition of ontologies describing the concepts involved in the measurement process and the link with the measured objects.

The ontology construction in OWL - dual to the JAVA taxonomy - cannot be produced automatically. Each concept of ontology has been constructed with concern for the representation of fine knowledge from a specific point of view: measurement. Indeed, the same point of view presides over the development of the JAVA taxonomy, but software engineering constraints are superimposed on a point of view strictly linked to knowledge of concepts.

We have abandoned an automatic mapping using JAVA annotation and JAVA beans for a manual extraction even if this is a common way in literature [4–8]. The main advantage of our approach is that it is possible to perform logical queries on both the ontology and the JAVA representation. We can thus read an ontology, visualize the artefacts in 3D present in the ontology as well as the result of SQWRL queries in the

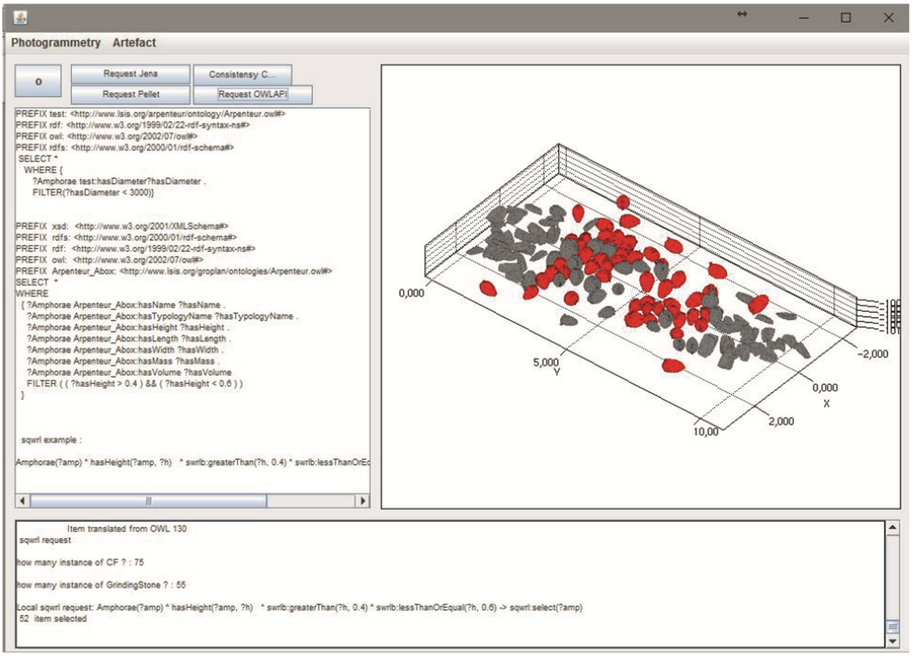

Fig. 1. 3D visualization of the answer to a spatial query in SWRL: `Amphorae(?a) ^ swrlArp:isCloseTo(?a, "IdTargetAmphora", 6.2) ^ hasTypologyName (?a, "Pitecusse_365") -> sqwrl:select(?a)`. Means select all amphorae with the typology *Pitecusse_365* and at a maximum distance of 6.2 m from the amphorae labelled *IdTargetAmphora*

JAVA viewer. A representation of the artefacts measured on the *Xlendi* wreck as well as an answer to a SWRL query is shown in Fig. 1.

The approach we have chosen so far, using OWLAPI and the Pellet reasoner, allows for handling SQWRL queries using an extension of SWRL Built-In[2] packages. SWRL provides a very powerful extension mechanism that allows for implementing user-defined methods in the rules. We have built some spatial operators allowing us to express spatial queries in SWRL, as for example the operator *isCloseTo* with three arguments which allows for selecting all the amphorae present in a sphere centered on a specific amphora and belonging to a certain typology (Fig. 1).

3 Conclusions and Future Work

Based on a procedural attachment approach we built a mechanism which allows for evaluating and visualizing spatial queries from SWRL rules. We are currently extending this approach in a 3D Information System dedicated to archaeological survey based on photogrammetric survey and knowledge representation for spatial reasoning.

Acknowledgement. This work is partially done in the framework of the projects GROPLAN (ANR-13-CORD-0014) and ASPIQ (ANR-12-BS02-0003) funded by ANR, the French agency for scientific research.

References

1. Drap, P., Merad, D., Hijazi, B., Gaoua, L., Nawaf, M., Saccone, M., Chemisky, B., Seinturier, J., Sourisseau, J.-C., Gambin, T., Castro, F.: Underwater photogrammetry and object modeling: a case study of xlendi wreck in Malta. Sensors **15**(12), 29802 (2015)
2. Gergatsoulis, M., Bountouri, L., Gaitanou, P., Papatheodorou, C.: Mapping cultural metadata schemas to CIDOC conceptual reference model, Athens, pp. 321–326 (2010)
3. Niccolucci, F., D'Andrea, A.: An ontology for 3D cultural objects. In: The 7th International Symposium on Virtual Reality, Archaeology and Cultural Heritage VAST (2006)
4. Roy, S., Yan, M.F.: Method and system for creating owl ontology from java. I.T. Limited, Google Patents (2012)
5. Kalyanpur, A., Pastor, D.J., Battle, S., Padget, J.A.: Automatic mapping of OWL ontologies into Java. In: SEKE (2004)
6. Ježek, P., Mouček, R.: Semantic framework for mapping object-oriented model to semantic web languages. Front. Neuroinform. **9**, 3 (2015)
7. Stevenson, G., Dobson, S.: Sapphire: generating java runtime artefacts from OWL ontologies. In: Salinesi, C., Pastor, O. (eds.) CAiSE 2011. LNBIP, vol. 83, pp. 425–436. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22056-2_46
8. Horridge, M., Knublauch, H., Rector, A., Stevens, R., Wroe, C.: A Practical Guide To Building OWL Ontologies Using the Protege-OWL Plugin and CO-ODE Tools. 1.0 (2004)

[2] https://github.com/protegeproject/swrlapi/wiki/SWRLBuiltInBridge#SWRL_BuiltIns.