

# Future and Emerging Technologies FET-Open



Phoenix

665347

## Technology analysis

Deliverable D3.3



V0.2



**KU LEUVEN**

**RWTH**AACHEN  
UNIVERSITY

**TU/e**

## Document history – List of changes

| Version | Date       | Author name                            | Scope                |
|---------|------------|--|----------------------|
| v0.2    | 30/09/2016 | Anil Yaman, Matt Coler, Giovanni Iacca | Final version        |
| v0.1    | 28/09/2016 | Anil Yaman, Matt Coler, Giovanni Iacca | First complete draft |

## Contents

|     |                                 |   |
|-----|---------------------------------|---|
| 1   | Technology Analysis .....       | 3 |
| 2   | References .....                | 5 |
| 3   | Appendix .....                  | 7 |
| 3.1 | A Simplified Class Diagram..... | 7 |



Funded by the European Union

## 1 Technology Analysis

This document reports on the technical aspects of the Knowledge Representation (KR) technology. **Table 1** presents tasks involved into the development, use and distribution of the Phoenix Knowledge Base. Each row identifies a task, an objective, and techniques that can be used. The final row serves as a reference to additional details including implementation status which follow in the corresponding section below the table.

**Table 1:** Tasks involved into the development, use and distribution of the Phoenix Knowledge Base.

|  | <i>Objective</i>  | <i>Technique</i>  | <i>Details</i> |
|--|---|---|----------------|
| <b><i>KR Strategy 1</i></b>                    | Representing domain knowledge   | Uncertain graph representation  | See (A) below. |
| <b><i>Use of existing knowledge</i></b>        | Making use of available open KR in context of Phoenix                           | Ontology integration and mapping  | See (B) below. |
| <b><i>KR Strategy 2</i></b>                    | Representing procedural knowledge   | Uncertain graphs and/or Frames and Scripts, Hierarchical Task Networks                      | See (C) below. |
| <b><i>KR Access</i></b>                        | Accessing all formalized knowledge  | Query evaluation, incl. methods Monte-Carlo algorithms, indexing and search space reduction | See (A) below. |
| <b><i>Experiment planning strategy</i></b>     | Generating procedural knowledge plans for exploration tasks                     | AI planner and reasoner (case-based and analogical)   | See (D) below. |
| <b><i>Integration [with Matlab]</i></b>        | Accessing formalized and serialized knowledge stored on computer                | Matlab object oriented programming classes  | See (E) below. |
| <b><i>Coding of the Knowledge Base</i></b>     | Serializing all formalized knowledge  | RDF, OWL and/or Neo4j   | See (E) below. |
| <b><i>Online visualization, monitoring</i></b> | All formalized knowledge, coded in OWL, RDF or similar for online visualization | Protégé or Neo4j  | See (F) below. |

**Human Interface Layer**

A GUI that is capable of obtaining user's question, request additional knowledge (if available), and provide results back to the used Matlab Software Tool See (G) below.

*(A). KR Strategy 1 & KR access - Under development (see D3.2). Estimated completion: M24.* The KR strategy we adopt for now is uncertain graph representation, a fundamental mathematical model that is used in applications to represent uncertain knowledge/data. It is suitable for representing domain knowledge and, to a lesser extent, procedural knowledge. The knowledge stored in the graph structure is accessed via queries. These queries are evaluated using one or a combination of following methods: Monte-Carlo algorithms (Karp, Luby, & Madras, 1989), indexing (Khan, Bonchi, Gionis, & Gullo, 2014) and search space reduction (Zhu, Zhang, Zhu, Zhang, & Lin, 2011) for fast and reliable results. This will be investigated further after the implementation of this strategy.

*(B) Use of existing knowledge - Under Investigation.* We note the possibility of making use of existing resources for knowledge in the Phoenix context. There are various formal and informal knowledge sources that can contribute to the domain knowledge in Phoenix. Informally, terminologies, dictionaries and taxonomies, and formally, open template ontologies can be useful. For example, the Semantic Web for Earth and Environmental Terminology (SWEET) ("SWEET Ontology", 2016) is an open ontology for defining environment types, and the Semantic Sensor Network ontology (SSN) (Compton, Barnaghi, & Bermudez, 2011) which represents the domain knowledge in sensor networks. The KR technologies we propose are capable of integrating with other external knowledge sources.

*(C) KR Strategy 2 - Under investigation. Estimated completion: M36.* We investigate the most appropriate KR technology suitable for representing procedural knowledge in Phoenix. Procedural knowledge can be represented as a graph structure that can indicate the procedural flow. Moreover, the choice of the KR should allow for modeling uncertainties in the procedures. Therefore, it is possible to use our uncertain graph representation (see A). However, query evaluation method might not be the best way to process procedural knowledge. We first assess the possibility of using uncertain graph representation and query evaluation methods, and investigate other possibilities such as frames and scripts (Minsky, 1974; Schank & Abelson, 1975), and hierarchical task networks (Erol, Hendler, & Nau, 1994).

*(D) Experiment planning strategy - Under investigation. Estimated completion: M48.* We investigate the possibility of using an AI planner (Stuart & Peter, 2003) and/or reasoning methods (case-based and analogical (Liao, 2005)) to plan and reason about procedural knowledge. It is not possible to represent explicitly all possible exploration processes. Therefore, we investigate using methods for generating procedural knowledge for the exploration processes that are not explicitly defined in the Knowledge Base. The situation may arise when the procedure for answering a user's question is not present in the Knowledge Base.

*(E) Integration with Matlab & Coding of the knowledge base - In development. Estimated completion: M36.* We started a joint effort on the example exploration task provided in D3.1 with the other components of the system to integrate the Phoenix Knowledge Base (completion date was estimated for all formalized knowledge; for prototype estimated completion date is M18). This integration has been initiated on a simplified class diagram in the system provided in 3.1. This integration is being done over the Matlab software tool. Accordingly, we are currently investigating methods to integrate our uncertain graph representation to work with Matlab software environment. One possibility is adopting the Web Ontology Language (OWL) (McGuinness, Deborah L and Van Harmelen, 2004); however, it may be insufficient for representing our uncertain graph model; because, we introduce features that may require improvements in the representation models that current tools use. Simpler uncertain graph databases were represented using Resource Description Framework (RDF) model (Udrea, Subrahmanian, & Majkić, 2006), and probabilistic ontologies were modeled in OWL (Ding, Peng, & Pan, 2006). Another possibility is graph databases, such as: Neo4j (Tudorache, Nyulas, Noy, & Musen, 2013).

*(F) Online visualization - Under investigation. Estimated completion: M36.* We are also investigating open-source tools that can be used to code and visualize the Knowledge Base. For example, platforms like Protégé (Gennari et al., 2003) and Neo4j provide the framework for coding a knowledge graph. These tools can also be used for online visualization purposes (Tudorache et al., 2013), where the state of the knowledge graph can be shared and monitored online.

*(G) Human Interface Layer (HIL) - Under investigation. Estimated completion: M24.* The task and deliverables dedicated to the development of HIL starts in M13, and ends in M24. During this task we will develop a graphical user interface (GUI) to allow interaction with the user. The GUI will obtain the initial question of the user, and request additional knowledge if available. It will allow user driven selection during co-evolutionary cycle. Since it has been proposed to use Matlab as a common platform between all the components of the Phoenix system, we are planning to use Matlab software tool for the development of the GUI.

## 2 References

- Compton, M., Barnaghi, P., & Bermudez, L. (2011). The SSN Ontology of the Semantic Sensor Networks Incubator Group. *Journal of Web ...*, 1–6. Retrieved from [http://www.w3.org/2005/Incubator/ssn/wiki/images/f/f3/SSN-XG\\_SensorOntology.pdf](http://www.w3.org/2005/Incubator/ssn/wiki/images/f/f3/SSN-XG_SensorOntology.pdf)
- Ding, Z., Peng, Y., & Pan, R. (2006). BayesOWL: Uncertainty modeling in semantic web ontologies. *Studies in Fuzziness and Soft Computing*, 204, 3–29. [http://doi.org/10.1007/3-540-33473-4\\_1](http://doi.org/10.1007/3-540-33473-4_1)
- Erol, K., Hendler, J., & Nau, D. (1994). UMCP: A Sound and Complete Procedure for Hierarchical Task-network Planning. *Aips*, 94, 249–254. Retrieved from <http://www.cs.umd.edu/~nau/papers/erol1994umcp.pdf>
- Gennari, J. H., Musen, M. a, Fergerson, R. W., Grosso, W. E., Crubezy, M., Eriksson, H., ... Tu, S. W. (2003). The evolution of Protege: an environment for knowledge-based systems development. *International Journal of Human-Computer Studies*, 58(1), 89–123. [http://doi.org/10.1016/S1071-5819\(02\)00127-1](http://doi.org/10.1016/S1071-5819(02)00127-1)

- Karp, R. M., Luby, M., & Madras, N. (1989). Monte-Carlo approximation algorithms for enumeration problems. *Journal of Algorithms*, 10(3), 429–448. [http://doi.org/10.1016/0196-6774\(89\)90038-2](http://doi.org/10.1016/0196-6774(89)90038-2)
- Khan, A., Bonchi, F., Gionis, A., & Gullo, F. (2014). Fast Reliability Search in Uncertain Graphs. *Edbt*, (c), 535–546. article.
- Liao, S. H. (2005). Expert system methodologies and applications-a decade review from 1995 to 2004. *Expert Systems with Applications*, 28(1), 93–103. <http://doi.org/10.1016/j.eswa.2004.08.003>
- McGuinness, Deborah L and Van Harmelen, F. and others. (2004). OWL Web Ontology Language Overview. *W3C Recommendation*, 1–22.
- Minsky, M. (1974). A Framework for Representing Knowledge. *MIT-AI Laboratory Memo 306*.
- Schank, R. C., & Abelson, R. P. (1975). Scripts, Plans, and Knowledge. *Proceedings of the 4th International Joint Conference on Artificial Intelligence*, 151–157.
- Stuart, J. R., & Peter, N. (2003). *Artificial Intelligence a Modern Approach*. *Artificial Intelligence* (Vol. 72). <http://doi.org/10.1017/S0269888900007724>
- SWEET Ontology. (2016). Retrieved from <https://sweet.jpl.nasa.gov/>
- Tudorache, T., Nyulas, C., Noy, N. F., & Musen, M. A. (2013). WebProtégé: A Collaborative Ontology Editor and Knowledge Acquisition Tool for the Web. *Semantic Web*, 4(1), 89–99. <http://doi.org/10.3233/SW-2012-0057>
- Udrea, O., Subrahmanian, V. S., & Majkić, Z. (2006). Probabilistic RDF. *Proceedings of the 2006 IEEE International Conference on Information Reuse and Integration, IRI-2006*, 172–177. <http://doi.org/10.1109/IRI.2006.252408>
- Zhu, K., Zhang, W., Zhu, G., Zhang, Y., & Lin, X. (2011). BMC: An efficient method to evaluate probabilistic reachability queries. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6587 LNCS(PART 1), 434–449. article. [http://doi.org/10.1007/978-3-642-20149-3\\_32](http://doi.org/10.1007/978-3-642-20149-3_32)

### 3 Appendix

#### 3.1 A Simplified Class Diagram

“A simplified class diagram designed for the Phoenix workshop 19/09/2016 (by Ahmed Hallawa, Stephan Schlupkoth, Giovanni Iacca and Anil Yaman).

