INFLUENCE AND SELECTION OF BASIC CONCEPTS ON ONTOLOGY DESIGN

Tomasz Boinski, Piotr Orlowski, Piotr Szpryngier and Henryk Krawczyk
Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology
11/12 Gabriela Narutowicza Street, Gdansk, Poland

Keywords: Ontology, Set of basic concepts, Ontology learning.

Abstract: Ontologies as entities representing individual point of view on surrounding world introduce heterogeneity to knowledge representation. Common set of core concepts can introduce similarity big enough for further interoperability between ontologies. In this paper an experiment is presented proving that despite differences in detail ontologies stay similar in regard of core concepts. During the experiments NOIA methodology enhanced by OCS methodology was used to create ontologies from three significantly different sources of knowledge about risk management.

1 INTRODUCTION

Ontology creation is a complex task dependent on methodology and a set of basic concepts. Ontologies presenting different points of view limit interoperability, making mapping or merging a tedious task.

To be usable by a wide range of recipients the created ontology needs to represent a common view of the described problem. Humans, however tend to have a distinct way of perceiving the surrounding reality. Different sets of basic concepts can be used and utilized. Furthermore a common set of basic concepts not always means a common set of basic definitions. Even slight differences can lead to potentially different ontologies. There are some upper ontologies proposed (Niles and Pease, 2001) (Masolo et al., 2003) but it’s highly unlikely that a global agreement will be met upon a common set of concepts or the shape of an ontology. Without such agreement upon basic concepts and their definitions, interactions between systems using different ontologies will be difficult - a need for costly manual mapping arises.

The purpose of this paper is to test whether using distinct sets of definitions for chosen sets of basic concepts can provide ontologies similar enough to interoperate with each other. In other words, how the choice of a knowledge source implies the final structure of the ontology.

Objectives for this test were as follow:
1. To choose the ontology domain and three different sources of knowledge.
2. To build three ontologies using chosen knowledge sources.
3. To evaluate prepared ontologies.
4. To compare result ontologies and conclude the research.

The following section will present the initial assumptions. Next, in section 3, methodology and groupwork model used for ontology creation will be described. In section 4 the basic set of concepts and their definitions are described. In section 5 the process of creating ontologies and achieved results are presented. Section 6 shows achieved results.

2 INITIAL ASSUMPTIONS

2.1 Ontology Domain

Choice of a vague domain of concepts was needed to allow use of more diversified knowledge sources. The risk has many, often contrary, definitions i.e. (Waste, 2006) (Hall and Hulett, 2002) (Knight, 2002) and that was the main reason of choosing it as the ontology domain.

2.2 The Ontology Purpose and Scope

The prepared ontology has a strictly research purpose. It should answer the question, what are the main rela-
tions between the concepts of risk, threat, vulnerability and asset. These terms appear in most risk definitions.

2.3 Knowledge Sources

Basing on a search of the literature, three sources of knowledge were chosen:

- NIST Glossary of Key Information Security Terms (Kissel, 2006),
- ENISA Risk Management Glossary (Enisa, 2010),
- Software Engineering, Ian Sommerville (Sommerville, 2006).

Two of them are glossaries prepared by recognized organizations: one American, NIST - National Institute of Standards and Technology and one European, ENISA - European Network and Information Security Agency. The third one is a recognized software engineering university coursebook from The United Kingdom, but it also contain definitions of used terms. All are texts written in natural language and describing security domains. NIST glossary contains definitions used in USA and gives us a view to terminology used in America. ENISA Glossary was prepared for European Union Institutions and Member States. Consequently it presents terminology used in Europe and often utilises ISO (International Organization for Standardization) definitions. The book written by Ian Sommerville presents risk terminology in a more objective way, according to concepts used by software engineers.

2.4 Used Groupwork Model and Software

Ontologies have been created using a methodology initially designed for Ontology Creation System (OCS) (Boinski et al., 2009) developed by The Department of Computer Architecture of Gdansk University of Technology. Opposed to methodology available in Protegeeditor (Gennari et al., 2002) (Noy et al., 2000) with Collaborative Protege (Tudorache et al., 2008) extension, this methodology ensures that the creator of the ontology will retain control over it. Whereas Protege resolves conflicts by means of voting, OCS methodology always leaves the final word to the ontology owner.

Any registered user can propose some changes in any ontology. A privileged user, i.e. ontology creator or an expert designated by him, can accept or reject proposed changes. After that, a new version is created and accepted changes are available for other developers. The whole process is showed in Figure 1. When an ontology is public and its development is not restricted in any way, the process of submitting propositions of changes can be combined with creating a new version.

For the purpose of this paper we distinguish a security expert, who was assigned the role of an ontology owner and all changes needed to be accepted by him.

Table 1: Aggregated results of ontology comparison using Falcon-OA (Jian et al., 2005).

<table>
<thead>
<tr>
<th></th>
<th>ENISA</th>
<th>NIST</th>
<th>Sommerville</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENISA</td>
<td>X</td>
<td>26</td>
<td>17</td>
</tr>
<tr>
<td>NIST</td>
<td>26</td>
<td>X</td>
<td>18</td>
</tr>
<tr>
<td>Sommerville</td>
<td>17</td>
<td>18</td>
<td>X</td>
</tr>
</tbody>
</table>

3 RESEARCH METHODOLOGY

3.1 Ontology Learning

There are many methodologies of learning ontologies from natural text, for example (McGuiness and
Table 2: Similarity between ENISA and NIST based ontologies.

<table>
<thead>
<tr>
<th>No.</th>
<th>Concept name in ENISA</th>
<th>Concept name in NIST</th>
<th>Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Group</td>
<td>Group</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>Operation</td>
<td>Operation</td>
<td>0.9989048162438352</td>
</tr>
<tr>
<td>3</td>
<td>Safeguard</td>
<td>Safeguard</td>
<td>0.9984604434491676</td>
</tr>
<tr>
<td>4</td>
<td>Mission</td>
<td>Mission</td>
<td>0.9979042801363289</td>
</tr>
<tr>
<td>5</td>
<td>Denial_Of_Services</td>
<td>Denial_Of_Service</td>
<td>0.9900581108148236</td>
</tr>
<tr>
<td>6</td>
<td>Attack</td>
<td>Attack</td>
<td>0.9805473494555519</td>
</tr>
<tr>
<td>7</td>
<td>Weakness</td>
<td>Weakness</td>
<td>0.964168640783707</td>
</tr>
<tr>
<td>8</td>
<td>Harm</td>
<td>Harm</td>
<td>0.9301309595602091</td>
</tr>
<tr>
<td>9</td>
<td>Circumstance</td>
<td>Circumstance</td>
<td>0.9177295758968978</td>
</tr>
<tr>
<td>10</td>
<td>Potential</td>
<td>Potential</td>
<td>0.9140700859364972</td>
</tr>
<tr>
<td>11</td>
<td>Threat</td>
<td>Threat</td>
<td>0.8307676666186594</td>
</tr>
<tr>
<td>12</td>
<td>Event</td>
<td>Event</td>
<td>0.811927478336268</td>
</tr>
<tr>
<td>13</td>
<td>Modification_Of_Data</td>
<td>Unauthorized_Modification_Of_Data</td>
<td>0.8098784893405487</td>
</tr>
<tr>
<td>14</td>
<td>Implementation_Error</td>
<td>Implementation</td>
<td>0.797691170284476</td>
</tr>
<tr>
<td>15</td>
<td>Risk</td>
<td>Risk</td>
<td>0.7695468416659597</td>
</tr>
<tr>
<td>16</td>
<td>Vulnerability</td>
<td>Vulnerability</td>
<td>0.756619672704696</td>
</tr>
<tr>
<td>17</td>
<td>Procedure</td>
<td>Security_Procedure</td>
<td>0.7487199546707506</td>
</tr>
<tr>
<td>18</td>
<td>Application</td>
<td>Major_Application</td>
<td>0.7344763918983326</td>
</tr>
<tr>
<td>19</td>
<td>Control</td>
<td>Security_Control</td>
<td>0.713245540577225</td>
</tr>
<tr>
<td>20</td>
<td>Unauthorized_Access</td>
<td>Unauthorized_Access</td>
<td>0.708416887969344</td>
</tr>
<tr>
<td>21</td>
<td>Destruction</td>
<td>Unauthorized_Destruction</td>
<td>0.7079624012576905</td>
</tr>
<tr>
<td>22</td>
<td>exploits</td>
<td>exploits</td>
<td>0.999738325227409</td>
</tr>
<tr>
<td>23</td>
<td>isMemberOf</td>
<td>isMemberOf</td>
<td>0.9994788690128461</td>
</tr>
<tr>
<td>24</td>
<td>hasMember</td>
<td>hasMember</td>
<td>0.9992263192460275</td>
</tr>
<tr>
<td>25</td>
<td>causes</td>
<td>causes</td>
<td>0.989844720077993</td>
</tr>
<tr>
<td>26</td>
<td>isPotentialOf</td>
<td>isPotentialThat</td>
<td>0.9075646311625767</td>
</tr>
<tr>
<td>27</td>
<td>hasPotentialToImpact</td>
<td>hasPotentialTo</td>
<td>0.7609824888801703</td>
</tr>
</tbody>
</table>

Table 3: Similarity between ENISA and Sommerville based ontologies.

<table>
<thead>
<tr>
<th>No.</th>
<th>Concept name in ENISA</th>
<th>Concept name in Sommerville</th>
<th>Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Value</td>
<td>Value</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>Weakness</td>
<td>Weakness</td>
<td>0.9564709904052531</td>
</tr>
<tr>
<td>3</td>
<td>Potential</td>
<td>Potential</td>
<td>0.9271158225611422</td>
</tr>
<tr>
<td>4</td>
<td>Harm</td>
<td>Harm</td>
<td>0.92097843867451191</td>
</tr>
<tr>
<td>5</td>
<td>Circumstance</td>
<td>Circumstance</td>
<td>0.9171607954083738</td>
</tr>
<tr>
<td>6</td>
<td>Attack</td>
<td>Attack</td>
<td>0.8580363492692699</td>
</tr>
<tr>
<td>7</td>
<td>Control</td>
<td>Control</td>
<td>0.8539265029954892</td>
</tr>
<tr>
<td>8</td>
<td>Asset</td>
<td>Asset</td>
<td>0.820426411517303</td>
</tr>
<tr>
<td>9</td>
<td>Computer_system</td>
<td>Computer_Based_System</td>
<td>0.7963994582890108</td>
</tr>
<tr>
<td>10</td>
<td>Information_Resource</td>
<td>Resource</td>
<td>0.795493114871415</td>
</tr>
<tr>
<td>11</td>
<td>Threat</td>
<td>Threat</td>
<td>0.782041135306344</td>
</tr>
<tr>
<td>12</td>
<td>Event</td>
<td>Event</td>
<td>0.781572930842786</td>
</tr>
<tr>
<td>13</td>
<td>Risk</td>
<td>Risk</td>
<td>0.758555685194784</td>
</tr>
<tr>
<td>14</td>
<td>Vulnerability</td>
<td>Vulnerability</td>
<td>0.73883024261744</td>
</tr>
<tr>
<td>15</td>
<td>reduces</td>
<td>reduces</td>
<td>0.999737166396932</td>
</tr>
<tr>
<td>16</td>
<td>hasValue</td>
<td>hasValue</td>
<td>0.998231905180453</td>
</tr>
<tr>
<td>17</td>
<td>exploits</td>
<td>canBeExploited</td>
<td>0.777660801540454</td>
</tr>
</tbody>
</table>

Noy, 2005) or (Fernandez et al., 1997). The modified “Noy and McGuinness” approach was used here. Original “Noy and McGuiness” Knowledge Engineering Methodology consist of 7 steps:

1. To determine the domain and the scope.
2. To consider reusing of existing ontologies.
3. To enumerate important terms.
4. To define classes and class hierarchy.
5. To define properties of classes.
6. To define properties restrictions.
7. To create instances.

Steps 2 and 7 were not performed. As the knowledge source in step 3 we used 3 different documents. Terms were taken from different definitions of Risk, Threat, Vulnerability, Asset and Safeguard. Step 6 was extended by adding definitions to classes (to change these classes into protege defined classes), according to definitions found in glossaries. All steps were enhanced by groupwork possibilities introduced by OCS methodology.

### 3.2 Ontology Evaluation

Evaluation was performed with use of advice given in (Staab and Studer, 2009). By evaluation we mean verification and validation of our ontology. The verification answers the question “did we build ontology in a correct way?”. The validation answers the question “did we build the proper ontology?”. The following quality criteria were considered during evaluation phase:

- accuracy - does the ontology present knowledge given in knowledge source?
- clarity - is the ontology understandable? is it documented?
- completeness - does it cover the domain of interest?
- consistency - does it match the specification?

Chosen methodologies made both the ontology owner and the participants responsible for the evaluation of the ontology. OCS methodology required double checking of created ontologies asking the aforementioned questions multiple times - once by normal users suggesting changes in ontology, and later by the ontology owner during the process of changes acceptance. Using Protege and Pellet (the reasoner plug-in used by Protege) provided the proper structure of OWL file.

### 4 ACHIEVED RESULTS

Three ontologies were constructed. Their inferred hierarchies are depicted on Figure 2. Final ontologies had different number of classes. ENISA based ontology had 43 classes (Figure 2 a), Sommerville based had 38 classes (Figure 2 b) and NIST based had 71 classes (Figure 2 c). They were compared using Falcon-AO (Jian et al., 2005) Ontology Matching tool. Falcon is based on linguistic matching for ontologies and uses promising (Euzenat and Shvaiko, 2007) graph modeling algorithms (GMO) with support of WordNet technology. Similarity of two entities from two ontologies comes from the combination of similarities of involved statements (triples) taking the two entities as the same role (subject, predicate, object) in the triples, while the similarity of two statements comes from the accumulation of similarities of involved entities of the same role in the two statements being compared (Hu et al., 2005). Compacted results are presented in Table 1. Values in the table represent the number of common concepts found in ontologies based on knowledge sources designated by NIST and Sommerville.
Full results (Tables 2, 3 and 4) show that main concepts are very similar. Differences occur mainly in leaf classes differentiating ontologies in details but not in main elements. Five core concepts (Risk, Threat, Vulnerability, Asset and Safeguard), and some other meaningful concepts, i.e. Attack, Weakness, Event and Harm were found similar in over 70% of all created ontologies.

5 CONCLUSIONS

Performed experiments show that despite differences in details, ontologies proved to be similar in regard of core concepts. Choosing a common set of basic entities enables users to design their own ontologies, yet creating a common base for interoperability at execution time. With the introduction of semantic dictionaries like WordNet (Fellbaum et al., 1998) such undertakings became possible and widely accepted. Common sets of core concepts defined by such dictionaries allow merging or mapping of ontologies on reasoning.
time, using knowledge from both ontologies. Concepts from chosen core sets are used as a bridge between those ontologies, opening new opportunities in knowledge integration and creating new possibilities when interoperating with other parties.

As a next step, research on ontology merging and integration will be performed examining new possibilities emerging from usage of a common set of core concepts.

REFERENCES


