International Workshop on Visual Analytics (2012)
K. Matkovic and G. Santucci (Editors)

Ontology Visualization: One Size Does Not Fit All

I. C. S. da Silva [1,2], G. Santucci[2] and C. M. D. S. Freitas[1]

[1] Federal University of Rio Grande do Sul, Brazil
[2] Sapienza Universita di Roma, Italy

Abstract

Visualization techniques have been used for ontology representation to allow the comprehension of concepts and properties in specific domains. Due to the complexity and size of ontologies such techniques need to be efficient in showing all the concepts and relationships in a intuitive visualization. We propose a novel use of the Degree of Interest notion in order to reduce the complexity of the representation itself and draw the user attention to the main concepts for a given task,. Through an automatic analysis of the ontology aspects, we place the main concept in focus, distinguishing it from the unnecessary information and facilitating the analysis and understanding of correlated data. This new Degree of Interest calculation can be easily adapted to different user tasks. Besides, we extended a multiple coordinated views approach proposed in previous works for exploring the intensional structure of an ontology. We also present a tool implementing these ideas as a proof-of-concept prototype.

Categories and Subject Descriptors (according to ACM CCS): H.5.2 [INFORMATION INTERFACES AND PRESENTATION]: Graphical user interfaces—Interaction styles;

1. Introduction

According to Gruber [Gru93], an ontology is a formal, explicit specification of a conceptualization that refers to the way people think about some part of the world one needs to represent for some purpose. This explicit specification relates concepts and relationships, which must be supplied in accordance with specific and well-defined terms. An ontology allows the representation of knowledge about some domain and as such allows obtaining information about this.

Visualization systems can help in the extraction of information from an ontology; a challenging task is to limit the amount of information that users receive, while keeping them "aware" of the total information space and reducing cognitive effort. Ontologies are usually represented as static 2D graphs where nodes and edges often overlap and cause cognitive overload depending on the size and complexity of the graph. Katifori et al. [KHLV07] confirm that it is not simple to create a visualization that displays effectively all the information, and, at the same time, allows the user to perform easily various operations on the ontology.

Visual Analytics can improve both quality and efficiency of ontology visualization systems, providing automatic means for driving the visual exploration. Based on these issues, this work presents a system for the visual exploration of an ontology. The system relies on multiple coordinated views [BWK00] based on different hierarchical visualization techniques in order to help users to understand complex relationships among different features and aspects of an ontology.

Moreover, in order to cope with very large ontologies, we employ a suppression technique [Fur86] based on the notion of *Degree of interest (DoI)* that, from the automatic analysis of an ontology's intension and extension, extracts knowledge about the relevance of concepts and relationships according to the user task. That technique allows exploring large ontologies focusing on a main concept and having the view of the most relevant concepts and relationships automatically computed and displayed.

Summarizing, the contribution of this paper is twofold:

- it extends the propose of multiple coordinated views, presented in previous works, in order to improve the visualization of the ontology hierarchy, classes, attributes, and

DOI: 10.2312/PE/EuroVAST/EuroVA12/091-095

relationships. To this aim both a focus+context and an overview+details views are provided;
- an automatic analysis of the ontology's concepts, relationships, and instances is performed, allowing a novel definition of DoI that can be easily adapted to different users' tasks. The DoI is used to automatically compute a task-oriented view of the ontology.

The paper is organized as follows. Section 2 reviews related work. Section 3 presents the formal model. Section 4 describes the implemented prototype, OntoViewer, and Section 5 draws some conclusions.

2. Related work

Different approaches for interaction, coordinated visualizations, and automated analysis applied to ontologies have been proposed. Katifori el al. [KHLV07] discuss different techniques that could be adapted for ontology representation, such as indented lists, trees and graphs, zooming, space subdivision (treemaps, information slices), focus+context and landscapes. The authors review tools for ontology visualization and interaction as well as methods for clustering or hiding nodes although automated analysis is not mentioned.

Baehrecke et al. [BDBS04] proposed the use of treemaps, together with color, size, and grouping as a means to visualize an ontology. Other works focused on semantic aspects. For example, Amaral [Ama08] proposes a semantics-based framework for visualizing descriptions of concepts in OWL [W3C09]. The framework aims at allowing users to obtain deep insights about the meaning of such descriptions, thereby preventing design errors or misconceptions. Other proposals, more close to our work, combine different information visualization techniques, as in the work by Schevers et al. [STD06], where the user interacts with the ontology in the Protégé tool. Classes representing spatial information (like polygons, points, etc.) are presented in a second graphical interface that is used to mimic the functionality of a GIS (Geographic Information System).

Catenazzi et al. [CSM09] propose the OWLeasyViz tool that combines textual and graphical representations for displaying ontology entities. Interaction techniques such as zooming, filtering and search are available. Kriglstein and Wallner [KW11] presented Knoocks, a visualization tool focused on the interconnections between the ontology concepts and instances. This tool employs the overview + details approach. Bach et al. [BPL11] proposed OntoTrix, a visualization technique designed to visualize large OWL ontology instance sets that employs both node-link and adjacency matrix representations of graphs to visualize ontology data.

In the works referenced above, there has been little or no concern to automate the extraction and display of concepts and properties of ontologies. Regarding this, Card and Nation [CN02] and Spence [Spe07] describe the application of the DoI concept for tree layouts as logical filtering of nodes, and Husken and Ziegler [HZ07] discuss the use of DoI in visualization and exploration of ontologies where nodes are automatically displayed or elided according to the user's computed DoI.

D'Entremont and Storey [d'E09] also apply DoI in their work and present a plug-in for ProtÉgé, called Diamond. This tool consists of two components: a mechanism to continuously calculate the user's DoI and a dynamic display of the information that uses the DoI calculation to draw users' attention to interesting elements in order to reduce navigation overhead. The results obtained from DoI are displayed over views existing in the Protégé (Class Browser and Jambalaya - this latter is discontinued). Chan et al. [CKL10] presents an interactive visual technique for analyzing and understanding hierarchical data, which they have applied for analyzing a corpus of technical reports. The analysis consists of selecting a known entity and then incrementally add other entities to the ontology graph based on known relations.

In our previous works, we investigated ontology creation and visualization [SNF09], performed requirements analysis and proposed a visualization tool based on interviews with experts who work with conceptual modeling and ontologies [SF11a], and proposed a multiple views ontology visualization tool that aims at systematizing and transmitting knowledge more efficiently [SF11c, SF11b]. In this work, we extend the multiple views with coordinated interaction and apply concepts of Visual Analytics in order to automate the analysis of concepts and properties of the ontology.

3. The formal model

Developing an ontology (see [NM01]), includes four main aspects: defining classes, arranging such classes in a hierarchy, defining relationships among classes, and defining instances of classes and relationships. According to this, we model an ontology as a tuple $O = (C, H, R, I_C, I_R, A)$ (adapted from [ES04]). Concepts C, which are classes of real-world objects, are organized in a hierarchy H; relationships R exist between pairs of concepts, describing properties of classes and instances. I_C is the set of the instances of all concepts and A are the concepts' attributes (also referred as classes' properties); I_R are the instances of the relationships.

We represent an ontology as a graph $G = (V, E \cup OE)$, where vertices V are the concepts C, edges $E \subseteq V \times V$ are the relationships R and the oriented edges $OE \subseteq V \times V$ are the classes' hierarchy H ($E \cap OE = \emptyset$). Moreover, to model the intensional part of the ontology and the *A Priori Importance (API)* of classes and relationships we introduce the following functions (where $v \in V$ and $e \in E$):

- $att(v)$, $att : V \to 2^A$, where A is the set of all concepts' attributes. Such a function returns the attributes of v;
- $inst(v)$, $inst : V \to 2^{I_C}$ where I_C is the set of all concepts' instances. Such a function returns the instances of the class v;

- $inst(e)$, $inst : E \rightarrow 2^{I_R}$ where I_R is the set of all relationships' instances. Such a function returns the instances of the relationship e;
- $rel(v)$, $rel : V \rightarrow 2^E$. Such a function returns all edges in E that involve v;
- $dep(v)$, $dep : V \rightarrow N^+$. Such a function returns the depth of v in the ontology hierarchy.

The cardinalities $|att(v)|$, $|inst(v)|$, and $|rel(v)|$, $|inst(e)|$, and $\frac{1}{dep(v)}$ are linearly combined to compute the API of concepts and relationships. Using such APIs and the distance of a concept from the user selected *Main Concept (MC)* it is possible to compute the DoI of classes, i.e., $DoI = f(API, D)$.

The *DoI* is used to automatically compute an ontology view containing the most relevant vertices and edges with respect to the *main concept*. More precisely, to compute the DoI we follow four steps:

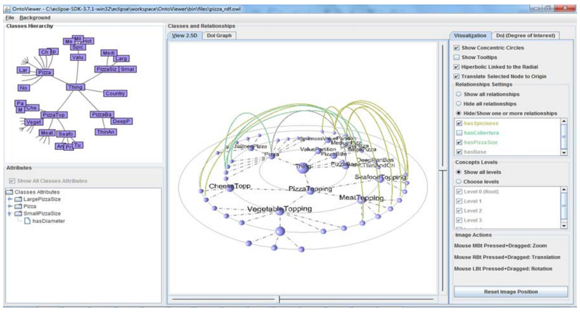
1. we assign an *API* value to each vertex in the ontology independently of the intended focus, i.e., the main concept MC selected by the user. In particular the *API* is computed using the following formula:
$API(v) = c_1|att(v)| + c_2|rel(v)| + c_3\frac{1}{dep(v)} + c_4|inst(v)|$
2. we assign an *API* value to each edges $e \in E$ using the following formula:
$API(e) = c_5\frac{|\{x|<x,y>\in inst(e)\}|+|\{y|<x,y>\in inst(e)\}|}{|inst(a)|+|inst(b)|}$
where a and b are the vertices connected by e and assuming $c_5 = 1$ it holds that $API(e) \in [0,1]$. Roughly speaking, we can say that $API(e)$ corresponds to the percentage of instances of a and b that are involved in the relationship e. Moreover we label e with $1 - API(e)$: such a label represents the *semantic distance* between a and b: if most of the instances of a and b are involved in the relationship, the classes are very related each other, the $API(e)$ is very close to 1 and the label is very close to 0;
3. we calculate the distances $D(v, MC)$ between the MC and each concept in V analyzing the different paths that exist between them. In the most general case, we have n_{oe} paths composed by OE edges and n_e paths composed by E. We label $oe \in OE$ edges with 1 and $e \in E$ edges with $1 - c_5API(e)$; the length of a path $l(p_i)$ is just the sum of its labels. In order to compute the overall distance we use a parallel resistor-like formula (the more the parallel paths the closer the two classes are):
$D(v, MC) = \frac{\prod_{i=1}^{n_{oe}+n_e} l(p_i)}{\sum_{i=1}^{n_{oe}+n_e} l(p_i)}$
4. we normalize D and API and we compute the DoI as:
$DoI(v, MC) = API(v) - c_6D(v, MC)$ and we normalize it.

Coefficients $c_1 \ldots c_6$ are set according to the user task: high c_1 and c_2 values are suitable when the user is interested in classes with high structural complexity (great number of attributes) and highly connected; high c_3 values are suitable when the user is looking for very abstract classes (close to the root); high c_4 and c_5 values allow for focusing on highly populated classes and relationships; and high c_6 values allow for exploring concepts that are far from the main concept. Initial values for these coefficients have been set during an informal user study involving expert ontology designers and undergraduate students; the actual version of the system allows changing such defaults and exploring the impact the changes have on API and DoIs (see Section 4).

Once DoI has been computed, it is sufficient to select a suitable threshold k and show on the view only the vertices where $DoI(v, mc) \geq k$. That results in a subgraph G' of the ontology induced by mc and k, where $G' = (V', OE' \cup E')$ is a subgraph of $G = (V, OE \cup E)$ with $V' \subseteq V$, $OE' \subseteq V' \times V'$ and $E' \subseteq V' \times V'$.

4. The prototype

In this section we describe OntoViewer, a tool that employs three integrated views showing different tree visualizations: an hyperbolic tree for representing the ontology hierarchy; a classic treeview for showing ontology entities, and an augmented radial tree for displaying relationships between classes (see Figure 1).



Attributes

☑ Show All Classes Attributes

- Classes Attributes
  - LargePizzaSize
  - Pizza
  - SmallPizzaSize
    - hasDiameter

☑ Show Concentric Circles
☐ Show Tooltips
☑ Hiperbolic Linked to the Radial
☑ Translate Selected Node to Origin

Relationships Settings
○ Show all relationships
○ Hide all relationships
● Hide/Show one or more relationships
☑ hasSpiciness
☐ hasCobertura
☑ hasPizzaSize
☑ hasBase

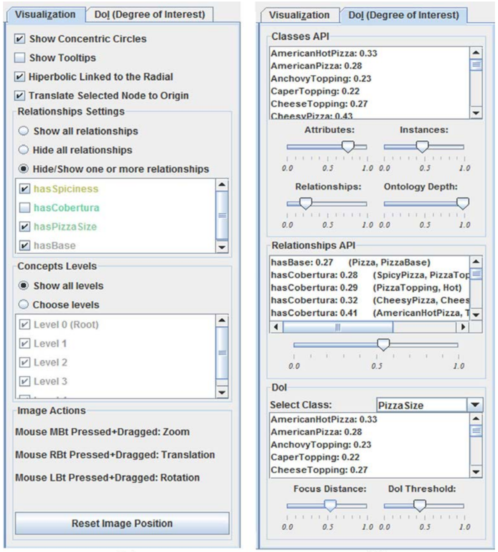
Concepts Levels
● Show all levels
○ Choose levels
☑ Level 0 (Root)
☑ Level 1
☑ Level 2
☑ Level 3

Image Actions
Mouse MBt Pressed+Dragged: Zoom
Mouse RBt Pressed+Dragged: Translation
Mouse LBt Pressed+Dragged: Rotation

Figure 1: *Ontology visualization: 2D hyperbolic tree, treeview, and augmented 2.5D radial tree.*

Treeview is an intuitive visualization while the 2D hyperbolic tree is a focus+context technique which reduces the cognitive overload and the user disorientation during the interaction with large ontologies. Concerning relationships, we display the classes hierarchy in a radial tree on the XZ-plane and selected relationships are represented as curved lines in space (thus yielding 2.5D), connecting the related classes without interfering with the display of the hierarchical structure.

While exploring the structure of the classes with the hyperbolic tree and the treeview representation, the user can interact with the 2.5D view by choosing to display one or more relationships at the same time or hiding them, choosing which levels of the tree view are to be shown or hidden, performing rotations around the axes X, Y and Z, zoom and pan, i.e., providing full 3D navigation. Moreover, when the user selects a class in the 2.5D view or hyperbolic tree, this node is placed in the center of both visualizations and the attributes of the selected class are displayed in the treeview in a coordinated form. These functionalities are available directly on the views or through the tab "Visualization" (Figure 2 (a); right panel of Figure 1).



☑ Show Concentric Circles
☐ Show Tooltips
☑ Hiperbolic Linked to the Radial
☑ Translate Selected Node to Origin

Relationships Settings
○ Show all relationships
○ Hide all relationships
● Hide/Show one or more relationships
☑ hasSpiciness
☐ hasCobertura
☑ hasPizzaSize
☑ hasBase

Concepts Levels
● Show all levels
○ Choose levels
☑ Level 0 (Root)
☑ Level 1
☑ Level 2
☑ Level 3

Image Actions
Mouse MBt Pressed+Dragged: Zoom
Mouse RBt Pressed+Dragged: Translation
Mouse LBt Pressed+Dragged: Rotation

Classes API
AmericanHotPizza: 0.33
AmericanPizza: 0.28
AnchovyTopping: 0.23
CaperTopping: 0.22
CheeseTopping: 0.27
CheesyPizza: 0.43

Attributes:
Instances:
Relationships:
Ontology Depth:

Relationships API
hasBase: 0.27 (Pizza, PizzaBase)
hasCobertura: 0.28 (SpicyPizza, PizzaTop
hasCobertura: 0.29 (PizzaTopping, Hot)
hasCobertura: 0.32 (CheesyPizza, Chees
hasCobertura: 0.41 (AmericanHotPizza, T

DoI
Select Class: PizzaSize
AmericanHotPizza: 0.33
AmericanPizza: 0.28
AnchovyTopping: 0.23
CaperTopping: 0.22
CheeseTopping: 0.27

Focus Distance:
DoI Threshold:

(a) (b)

Figure 2: *Tabs panel: (a) Visualization controls; (b) Visual Analytics controls.*

The DoI tab (Figure 2 (b)) shows the API values of classes and relationships, and the DoI calculation parameters. These values can be changed through sliders that control the coefficients $c_1 \dots c_6$ (see Section 3). The interaction with the sliders generates new results that can be analyzed in the lists of this tab and in a 2D plot as shown in Figure 3. The slider DoI threshold allows filtering classes and relationships in the 2.5D view (Figure 4), in order to reduce the complexity of the visualization according to the user task.

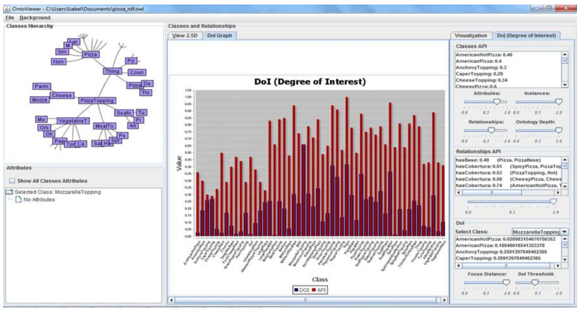


Figure 3: *2D Plot of APIs and DoI values.*

At present we are testing this prototype with different ontologies sizes and domains; moreover we have performed an informal user study involving expert ontology designers and undergraduate students, and a usability inspection method based on cognitive walkthrough simulating different analysis tasks on the ontologies.

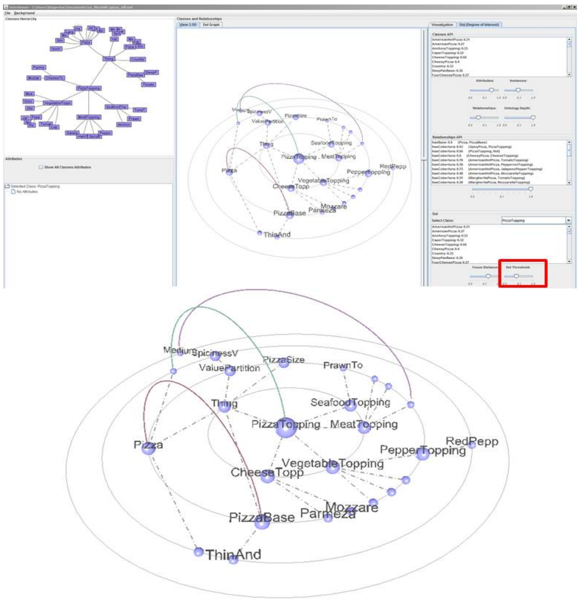


Figure 4: *Results of DoI threshold calculation.*

5. Conclusions

Multiple and coordinated views can help users to understand different aspects of data sets particularly when coupling two or more views showing different patterns that reveal hidden relationship. However, we have few studies exploring the visualization of ontologies using more than one synchronized view. In this sense, Information Visualization and Visual Analytics techniques amplify cognition and reduce exploration time of a data set, allowing the recognition of patterns and facilitating inferences about different concepts.

We have designed a visual and interactive way to explore an ontology, improving the process of insight from such data by applying multiples coordinated views and automatic analysis. Our visualization method combines aspects of both 2D and 3D techniques in a intuitive interaction based in hierarchical views and focus+context concepts. For the data analysis, we calculate the degree of interest (DoI) and show the results in two dynamic views to reduce cognitive overload and amplify the understanding of the analyzed data according to different tasks.

This is a preliminary study involving visualization and analysis of ontologies structures. Moreover, we intend to realize formal evaluation studies with experts and investigation of alternative display of ontology instances and their relationships, ontologies with different sizes and different domains.

6. Acknowledgements

We would like to thank the CAPES, Brazil, for financial support (Process 5341-11-6).

References

[Ama08] AMARAL F. N.: Visualizing the semantics (not the syntax) of concept descriptions. In *VI Workshop em Tecnologia da Informacao e da Linguagem Humana (TIL 2008)* (2008). 2

[BDBS04] BAEHRECKE E., DANG N., BABARIA K., SHNEIDERMAN B.: Visualization and analysis of microarray and gene ontology data with treemaps. *BMC Bioinformatics* (2004). 2

[BPL11] BACH B., PIETRIGA E., LICCARDI I.: Ontotrix: A hybrid visualization for populated ontologies. In *20th Internacional Word Wide Web Conference.* (2011). 2

[BWK00] BALDONADO M., WOODRUFF A., KUCHINSKY A.: Guidel ines for using multiple views in information visualization. In *AVI '00 Proceedings of the working conference on Advanced visual interfaces* (2000), ACM New York. 1

[CKL10] CHAN Y. H., KEETON K., L. M. K.: Interactive visual analysis of hierarchical enterprise data. In *IEEE 12th Conference on Commerce and Enterprise Computing (CEC)* (2010), pp. 180 – 187. 2

[CN02] CARD S. K., NATION D.: Degree-of-interest trees: a component of an attention-reactive user interface. In *AVI '02 Proceedings of the Working Conference on Advanced Visual Interfaces* (2002), ACM New York, pp. 231–245. 2

[CSM09] CATENAZZI N., SOMMARUGA L., MAZZA R.: User-friendly ontology editing and visualization tools: the owleasyviz approach. In *13th IEEE International Conference on Information Visualisation.* (2009), pp. 283–288. 2

[d'E09] D'ENTREMONT T.; STOREY M.-A.: Using a degree of interest model to facilitate ontology navigation. In *IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC 2009* (2009), pp. 127–131. 2

[ES04] EHRIG M., SURE Y.: Ontology mapping - an integrated approach. In *Proceedings of the First European Semantic Web Symposium* (Heraklion, Greece, May 2004), Bussler C., Davis J., Fensel D., Studer R., (Eds.), vol. 3053 of *Lecture Notes in Computer Science*, Springer Verlag, pp. 76–91. 2

[Fur86] FURNAS G.: Generalized fisheye views. In *Human Factors in Computing Systems CHI Ś86 Conference Proceedings* (1986), pp. 16–23. 1

[Gru93] GRUBER T.: A translation approach to portable ontologies. *Knowledge Acquisition 5*, 2 (1993), 199–220. 1

[HZ07] HUSKEN P., ZIEGLER J.: Degree-of-interest visualization for ontology exploration. In *INTERACT'07 Proceedings of the 11th IFIP TC 13 international conference on Human-computer interaction* (Heidelberg, 2007), Baranauskas C., Palanque P., Abascal J., Diniz S., (Eds.), Springer-Verlag Berlin, pp. 116–119. 2

[KHLV07] KATIFORI A., HALATSIS C., LEPOURAS G., VASSILAKIS C.: Ontology visualization methods: a survey. *ACM Computing Surveys (CSUR) 39*, 4 (2007). 1, 2

[KW11] KRIGLSTEIN S., WALLNER G.: Development process and evaluation of the ontology visualization tool knoocks - a case study. In *International Conference on Information Visualization Theory and Applications IVAPP, 2011* (2011), SciTePress Science and Technology Publications, pp. 187–197. 2

[NM01] NOY N. F., MCGUINNESS D. L.: *Ontology Development 101: A Guide to Creating Your First Ontology*. Tech. rep., Standford University, 2001. 2

[SF11a] SILVA I. C. S., FREITAS C. M. D. S.: Requirements for interactive ontology visualization - using hypertree+2.5d visualization for exploring relationships between concepts. In *Proceedings of the International Conference on Imaging Theory and Applications and International Conference on Information Visualization Theory and Applications, IVAPP* (2011), SciTePress Science and Technology Publications, pp. 242–248. 2

[SF11b] SILVA I. C. S., FREITAS C. M. D. S.: Using multiple views for visual exploration of ontologies. In *IV Seminar on Ontology Research in Brazil (Ontobras) and VI International Workshop on Metamodels, Ontologies and Semantic Technologies (MOST)* (2011), CEUR-WS, pp. 25–36. 2

[SF11c] SILVA I. C. S., FREITAS C. M. D. S.: Using visualization for exploring relationships between concepts in ontologies. In *15th International Conference on Information Visualisation, IV* (2011), IEEE, pp. 317–322. 2

[SNF09] SILVA I. C. S., NETTO V. M., FREITAS C. M. D. S.: Novos caminhos para simulação urbana: integrando métodos de visualização de informações e modelagem de agentes e redes espaciais. In *XIII Congreso Iberoamericano de Gráfica Digital - Sigradi* (2009). 2

[Spe07] SPENCE R.: *Information Visualization: Design for Interaction*, second ed. Pearson, Prentice Hall, England, 2007. 2

[STD06] SCHEVERS H., TRINIDAD G., DROGEMULLER R.: Towards integrated assessments for urban development. *Journal of Information Technology in Construction (ITcon): Special Issue Decision Support Systems for Infrastructure Management 11* (2006), 225–236. 2

[W3C09] W3C: Owl 2 web ontology language document overview, 2009. 2