Let's Speak Graphic!

Development of a Web-based Editor for Graphical Ontology Creation

(Bachelor Thesis Proposal)

"*People can't share knowledge if they don't speak a common language.*"
Thomas H. Davenport, 1998 [1]

1 Introduction and Motivation

Today's industry is highly interconnected. The success and development of technological evolutions depend on the cooperation of a growing number of expertises from different knowledge domains. Against this backdrop Thomas H. Davenports quote from the late 90's is more prominent than ever before. His book the "Working Knowledge: How Organizations Manage What They Know" [1] is a strong argument for the importance of managing, integrating and using different accessible knowledge resources in companies in order to strive forward.

One approach to tackle this problem is the use of ontologies as a means to standardize and exchange knowledge in terms of concepts and relations over different domains. Ontologies are "formal, explicit specifications of shared conceptualizations" [2]. Formerly, they were a philosophical concept, adapted by engineers in 1991 [3] as a possibility to solve the problem of knowledge representation and sharing in artificial intelligence. The concept got even more prominent in the context of the "Semantic Web" introduced by Tim Berners-Lee and colleagues in 2001 [4]. Today ontologies are widely used in different disciplines [5].

The ongoing project GENIAL! is also relying on ontologies to bridge the gap between original equipment manufacturers and suppliers in the context of technology roadmap development for microelectronics along the automobile value chain. Roadmapping allows for strategic and long-term planning by matching changing short-term and long-term technology goals with technological solutions. However, this process needs experts from different companies and knowledge domains to pull together. This in turn, requires two important things. Firstly, a possibility for distributed working, as the involved companies are geographically dispersed. Secondly, a common ground for knowledge communication - experts need to speak a common language. GENIAL! aims at providing the possibility to create ontologies in the early stages of roadmap development in order to ensure standardized, consistent and reliable knowledge flows between different companies. This means that ontologies should be built and maintained by domain experts. Although they are a prominent user group, the creation of ontologies is in general left to specialized ontology engineers. Hence, domain experts are rarely proficient in ontology creation and do not know the formal languages or logic which express ontological concepts. GENIAL! recognizes the

recent trend in ontology engineering towards integrating and empowering domain experts in the ontology creation process [6]. Therefore, the development of a graphical ontology editor adjusted to the specific requirements emerging from the inexperience of the domain experts and from the project context is proposed. In GENIAL!, the possibility of distributed working is ensured by a web-based platform. For this reason, the resulting editor needs to be a web-based application.

2 Problem Statement

An ideal ontology creation tool should be well usable, comprehensible and effective to use by both ontology and domain experts. However, existing tools are typically general-purpose tools with a variety of capabilities and possible applications, at the price of being very complex. These tools require that at some point the user has a deeper understanding of the underlying ontological concepts and logic, otherwise the user would not be able to efficiently use and benefit from all the functionalities of the tool. For this reason, there is a trend in scientific research to rethink the design of such editors. Different approaches do already exist to reduce complexity [7] and to make ontology creation more accessible to domain experts [8], [9], [10].

The project GENIAL! faces the same challenge. Ontology creation should be made possible for domain experts without any prior knowledge to ontologies in general and formal languages or logic. Thereby emphasis should not only be put on ontology creation but also on enabling domain experts to efficiently maintain and manage already created ontologies. Thus there are some concrete requirements to the solution of the editor which are going to be further explained below.

- The editor should provide a graphical interface.
- The editor should be web-based.
- The editor should use a suitable graphical notation for visualization.
- The editor should support the domain experts in understanding the underlying concepts of ontology creation.
- Finally, it is desirable that for domain experts the editor makes it possible to learn about and deepen the understanding of ontologies and their creation.

In GENIAL! the desire for a graphically based ontology editor exists, since the target group is not necessarily used to working with formal languages or logic, but is familiar with graphical modeling applications such as Enterprise Architect or IBM Rational Rhapsody to support work organization and workflow. Therefore the graphical editor is supposed to fit into the familiar working methods and the hurdle to use should be low.

Further on, as stated in the introduction the editor must allow for distributed working. This is best ensured by setting up a web platform implying a web-based solution for the editor. So far, there exists only one web-based graphical ontology editor - the WebVOWL Editor [11]. This editor represents an evolution of the WebVOWL ontology visualizer and is still in its beta phase. It might not serve as a basis for this work, but it might certainly hint a few interesting ideas.

With regard to the visualization, a form should be chosen that requires as little additional learning effort as possible while at the same time covering the relevant constructs of the underlying ontology language. Since the user group in GENIAL! is used to working with standards such as SysML, it should

be investigated to what extent, for example, visualization approaches based on UML notation [12], [13] can be considered.

Taken all of this together the editor should enable the transfer of existing competencies in abstraction and graphical modeling on ontology creation while on the same time minimizing tool based cognitive load on domain experts [14]. This means that an editor should be developed which is specifically tailored to the most important functionalities for creating and editing ontologies. In line with the research trends mentioned above the editor should additionally provide domain experts with necessary support to gain deeper understanding in the ontology creation process.

3 Purpose of the Study

This research addresses the following three main challenges:

1. Extraction of most important functionalities for ontology creation and editing from literature and existing editors.
2. Find out how to realize tool-based support which promotes understanding of underlying ontology concepts for inexperienced users.
3. Implementation of the identified core functionalities and evolved design concept in a web-based ontology editor.

4 Review of the Literature

4.1 Ontology Basics

For this work it is indispensable to deal more closely with the concepts and foundations of ontologies in information science. Therefore, the following literature is going to be considered among others.

Handbook on Ontologies [15]
The first part of the book provides a detailed overview of ontology concepts from an introduction to description logic to the specific ontology language OWL.

Ontologien [16]
This book chapter gives a deeper insight into the differentiation from other knowledge representations like taxonomies and explains the relevant concepts with well chosen examples.

4.2 Ontology Development Methodologies

Building on the knowledge of ontology basics, it is also important to gain understanding of the ontology creation process, as this is the task to be solved with the editor proposed in this work. The below research has two main emphases. On the one hand literature is selected which gives a general overview of existing methodologies on the other hand literature is presented which focuses on the practical applicability of the theoretical methodologies in the context of inexperienced users.
Ideally, the literature search in this area will result in an ontology creation method which is an easy to learn sequence of steps for non-experts.

Ontology Development 101: A Guide to Creating Your First Ontology [17]
This is only one example for the first group of literature. Noy and McGuinness provide a good step-by-step guide along the tool Protégé. Thereby giving a rough overview of the involved phases.

User-Friendly Ontology Creation Methodologies - A Survey [18]
This research compares 16 different development methodologies and serves as an example for the second group of literature. The authors evaluate the methodologies for their user-friendliness for domain experts, concluding a substantial lack of the latter. This work can serve as a starting point for an overview of existing methodologies and the assessment of strengths and weaknesses.

4.3 Recent Trends Towards Non-Expert Friendly Ontology Editors

In order to make ontology creation accessible to the masses, the scientific world has been rethinking the development of ontology editors for some time. The literature presented in this section provides some of the more interesting approaches which could be valuable during the design and implementation process of the editor to be developed in this work.

Human-centered Ontology Engineering: The HCOME Methodology [6]
This work stresses the need for an ontology engineering methodology centered on non-experts and gives information about the emergence of this need. A prototype of a human-centered engineering environment is presented. Unfortunately, no further evolvement of the proposed methodology and tool could be found. The project website was last updated in September 2004 [19].

Supporting Domain Experts to Construct Conceptual Ontologies: A Holistic Approach [10]
This paper presents ROO (Rabbit to OWL Ontology authoring) an ontology creation environment specifically designed to support domain experts. They use a controlled natural language approach to establish ontology concepts since they see it as more familiar to inexperienced users than OWL constructs. Furthermore, they have an in-tool creation methodology guideline which should assist the workflow of the users. ROO was developed as a Plugin for Protégé 4 but unfortunately it is no longer listed in the list of supported plugins (see [20]).

Overcoming the Pitfalls of Ontology Authoring: Strategies and Implications for Tool Design [21]
The authors present results from interviews with 15 ontology engineering experts and derive recommendations to enhance ontology creation tools. Furthermore, they develop some guidelines for non-experts towards ontology creation.

Towards Self-explanatory Ontology Visualization with Contextual Verbalization [9]
The authors present a feature extension to the OWLGrEd-editor. They implement controlled natural language prompts which explain defined relations between concepts and properties in the created ontology in order to make domain experts adapt faster to the ontology creation process.

This literature presents only a small selection. However, all these literatures have a decisive difference to the research work presented here - namely the way in which the improved accessibility of ontology creation for domain experts is defined and measured. It is often evaluated whether inexperienced users can more easily and clearly create valid ontologies using the new approaches and whether they are more satisfied with the use than with other tools. All of these results are very important. Yet, they do not give much indication as to whether the users have actually experienced a learning effect. The question if

this new approach will actually deepen the understanding of ontology concepts is answered only to a limited extent. Just because a tool makes it clear to the user what his next step should be and how he achieves it with the tool, it is not necessarily guaranteed that the user has understood why this step must be carried out in the first place and which effects it might have in the larger context. However, this statement should also be discussed in more detail within the framework of the work.

4.4 Ontology Editors

Only four editors which follow a consistent graphical approach have been found so far: WebVOWL Editor [11], EDDY - The Graphol Editor [22], OWLGrEd [12] and Hozo [23]. The first one is still in beta phase and is also the only web-based application. The other three are all desktop applications existing in several versions. However, all of the editors are freely available and there are scientific publications at hand which are going to be consulted. Studying existing ontology editors is important in order to be able to identify necessary functionalities which need to be implemented in the proposed editor.

Functionality
Evaluation and Ranking of Ontology Construction Tools [24]
This technical report postulates several evaluation criteria for ontology tools in general. Thereby the authors provide a comprehensive overview of desired features addressing among others, functionalities, learnability, efficiency, complexity and user-satisfaction.

Usability
An important point in the development of any technical system, which is actually going to be used, is to ensure good usability and a positive user experience. Only then can it be guaranteed that users will continue to use the system. Unfortunately in the context of ontology editors, not much objective and reliable literature is found. Most usability evaluations are qualitative studies based on the comparison of specific tool features aiming at identifying the better suited tool. However, such results are not of great interest for this work. The focus in this research is more on understanding user needs and capturing gaps in existing tool support in order to either profit from previously proposed solutions or to avoid already known problems. However, the literature found so far is not specifically related to any of the above mentioned graphical editors. Additionally, the studies are rather old (approximately about 10 years). This requires that the results need to be critically questioned and opposed to the standards of the editors in question. Nevertheless, some points might still be of interest.

Usability Evaluation of Ontology Editors [25]
The authors evaluate the usability and adequacy of seven ontology editors for people with no specific ontology creation skills but with a general knowledge about domain modeling. They conclude that the tested editors seem to be “fairly adequate” (p. 12) for the target group. Additionally, they derive several improvement areas including context-aware help systems, tool-language and navigation and filtering mechanisms.

Developing Ontologies in OWL: An Observational Study [26]
This paper presents and observational study which focuses on eliciting weaknesses in the user support of then prominent tools.

4.5 Modeling Approaches

With evolution of the model-based software engineering approach formerly made distinctions between ontologies and models are now more and more converging towards one another. Atkinson and colleagues

stated that "all ontologies are models, but not all models are ontologies" [27]. Following their line of reasoning every knowledge representation that is said to be an ontology should also fulfill the conditions to be a model. Hence, everything that applies to models should also apply to ontologies. Since the target group in GENIAL! is versed in the application and use of modeling techniques based on SysML a closer look at modeling approaches and tool support may lead to useful insights for the proposed ontology editor.

All in all, the presented research selection does not claim to be complete. During the work process, there may be changes in the overarching topics as well as changes concerning certain literatures. In addition, literature that is not specifically associated with the problem, but with its implementation or methodology, will also be consulted.

5 Research Questions and/or Hypotheses

The overarching question of this work is the development of a web-based ontology editor tailored to the specific needs emerging from the project context of GENIAL!. In addition to the results of the implementation and the subsequent evaluation, this work also covers the following theoretical questions.

RQ1 What are necessary and what are sufficient editor features concerning functionality and usability for ontology creation?

RQ2 What are existing ontology creation methodologies?

RQ2 a). How are they evaluated with regard to adequacy for non-ontology experts?

RQ2 b). What can a recommendable methodology for non-experts look like?

RQ3 How can support possibilities in the task of ontology creation be derived for non-experts from RQ2?

RQ3 a). How can the proposed support options be translated into implementable editor features?

RQ3 b). To what extent can these options serve in gaining deeper understanding of underlying ontology concepts?

6 The Design - Methods and Procedures

In the context of information system science there exists the methodology of "Design Science Research" (DSR) which is focused on the development of new artifacts. DSR is designed as a problem solving process and has been used in many research fields for almost a decade now. In [28] the authors provide the necessary steps of an iterative DSR process. This work orients its methodology to these guidelines.

1. Analysis
   This phase builds the foundation of the following ones. It includes two subphases.
   a) Problem Identification
   During this phase the research problem is stated and put into a broader research context. This has already been done in sections 2 and 4.

   b) Problem Elaboration and Deepening
   This phase requires a more in-depth literature examination of the subject areas elaborated

in section 4. The extent to which existing work can serve to resolve the first two research challenges identified in section 3 must be explored. Gaps in existing literature should be identified and it should be made clear whether these can be filled by this work. A first approach can be found at the end of 4.3. Additionally, the author of the presented work is going to use the four existing graphical editors presented in section 4.4 to create an excerpt from the well-known pizza ontology. This example exists in all four editors as a tutorial file, which can serve as reference for the author. The actual usage should identify important features regarding functionality and usability or even the lack of such.

Ideally, with regard to the first research challenge the outcome of this phase should be an overview of identified important functionalities which need to be implemented in the proposed editor and have a justification based on existing literature and tools. With regard to the second challenge the outcome should cover two aspects. Firstly, an understanding of existing ontology creation methodologies and how they can be applied for inexperienced users should be established. A recommendable procedure model for domain experts should be developed based on existing literature. Secondly, an evaluation of existing approaches towards ontology editors designed for non-experts should be made. The evaluation should concern towards adequacy and applicability in the context of this work resulting in a set of favorable concepts.

2. Design
   The second phase centers on the conceptualization and implementation of the graphical ontology editor. It is composed into three different subtasks.

   a) Conceptualization
      During this phase the editor features defined in the previous phase are translated into actual requirements for the implementation. It needs to be paid attention on how the functionalities mainly resulting from the study of desktop applications and literature can be transferred into a web-based environment. Furthermore, it must be evaluated to what extend and how the ontology creation procedure model adapted to domain experts can be translated into design patterns. Existing usability and user interaction patterns need to be consulted.

   b) Preparation
      After defining the features and supports which should be implemented, an adequate architecture for the editor to be developed must be designed.

      Furthermore, since the work is focused on developing a graphical web-based editor a respective drawing framework which enables graphical modeling functionalities must be chosen. This decision is crucial as the possibilities for extension provided by the framework determine the scope and quality of the resulting implementation. The tradeoff between the importance of the function to be implemented and the possible feasibility in the framework has to be considered carefully. Therefore, the application programming interfaces of possible frameworks have to be studied.

   c) Implementation
      Following the latter two phases the implementation is realized during this phase.

3. Evaluation
   Closing the research cycle, the result of the above two phases is assessed. It needs to be evaluated to what extent the implementation has been accomplished. Furthermore, the implication that the resulting editor not only assists domain experts during the creation of ontologies but also fosters deeper understanding of the underlying concepts needs to be verified. Hence, an evaluation study is proposed.

   a) Evaluation Study
      The assumptions need to be tested against a sample group which matches the attributes of later users in the project GENIAL!. As stated in section 2 domain experts in GENIAL! are familiar to graphical modeling but are inexperienced in ontology creation. Thus, employees of the Institute of Software Engineering and Programming Languages at Ulm University seem to be an appropriate target group, as they are used to various graphical modeling approaches in the context of software development. However, most of them have little specialized knowledge on ontologies.

      The evaluation should take the form of a usage test. However, the concrete procedure needs to be determined after the functions and manner of user support have been implemented in the editor. So that it can be ensured that the evaluation actually captures the correct indicators and measures the desired variables.

      Ideally, the results of the evaluation study should support the intentions of the developed editor. If this is not the case then the evaluation should help in eliciting existing problems and hurdles and therefore indicate the direction for further developments.

4. Diffusion
   This last phase is concerned with the promotion of the results of this work in scientific discourse. This is going to be done in the form of the final thesis and the possible further development in the course of the GENIAL! project.

7 Limitations and Delimitations

Regarding the evaluation study the validity of the results is depending on the matching of the selected sample to the attributes of the target group in GENIAL!. If the matching is not adequate then the results can not be generalized to the actual user group. Nevertheless, a sample of the actual user group is not available during this work. Furthermore, it should be noted that due to the limited time span, the results obtained from the study can only be considered as trends. It is not possible to carry out such an assessment that objective implications are possible.

In section 6 during the step 1b of the analysis phase it is proposed to work with existing graphical ontology editors to elicit important features. Other ontology engineering tools like for example Protégé [29] are not considered at all. This is on the one hand due to the limited amount of time in this work but on the other hand it is also assumed that the authors of the graphical ontology tools have probably already benchmarked and adapted their functions to other existing standards.

In addition, this work is limited in the time available for the selection of a suitable framework. So it may be that the selected framework is not the most suitable one (see also 9.4).

Furthermore, this work is conducted as part of the project GENIAL!. The resulting editor and its conceptualization are based on conditions emerging from the project context. Thus, it might not readily be applied in another context.

8 Significance of the Study

This study will complement existing research on ontology editors designed for domain experts. However, this study may also have the potential to expand existing research in this area. Since, it is desired to create an ontology editor which does not only assist inexperienced users in ontology creation but that does also have the aim to foster understanding of underlying ontological concepts. At this stage of knowledge, such an approach has not yet been proposed.

In the context of GENIAL! this work proposes an editor which is specifically tailored to the requirements emerging from the project context.

9 Planning

9.1 Own Background

As the proposed editor needs to be implemented as a web-based application, experiences in this area would be beneficial. Since July 2017 I have conducted two small website projects with two fellow students and since January 2016 I am working as Hiwi in the Hochschulsport and I am jointly responsible for the creation of web forms. Thus, I could already collect some experience values in web-based development, including programming with the scripting languages PHP and JavaScript and using the jQuery library. However, I have not implemented a web graphical editor yet, so I do not have any experience with potential frameworks.

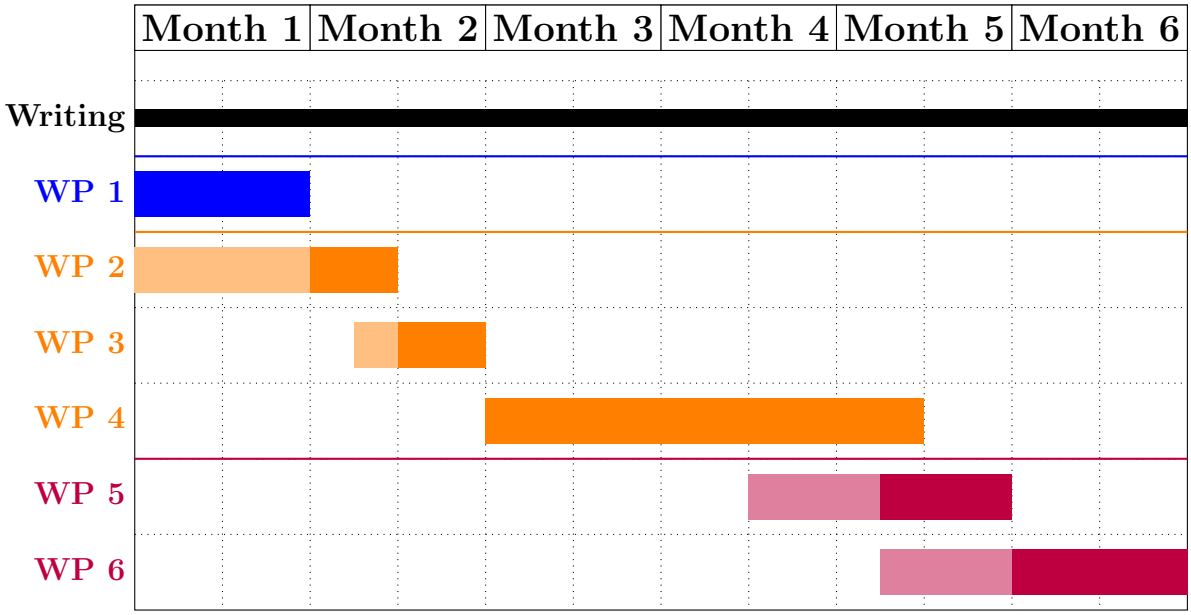
The presented work requires familiarization with ontology basics and creation methodologies.
Furthermore, it is necessary to take a closer look at qualitative research methods, such as usage analysis.

9.2 Required Resources

This work requires the availability of an appropriate sample for the evaluation study (see 6). Further on, a web server to run the editor is needed.

9.3 Work Packages

The present work is part-time and is designed for a period of six months. The following figure depicts the six specified work packages (WP) and their assumed durations spread over the months. Bars with a lower opacity indicate that preparatory work is being done for the respective work package. Bars with full opacity represent the duration over which the core work is conducted.



WP 1 covers the **analysis phase** of the presented work. It focuses on the activities listed in 1b. Main aspects are the in-depth literature research and actual usage of existing graphical ontology editors.

WP 2, **WP 3** and **WP 4** represent the work of the **design phase**. In **WP 2** the steps mentioned in 2a are performed. The knowledge gained from literature research and working with existing editors must now be translated into concrete implementation requirements. This can partly run parallel to the work package 1. **WP 3** is concerned with the architecture and framework choice described in 2b. Building on the steps of working packages 1 to 3, **WP 4** represents the implementation phase in 2c.

The last two work packages contain the work of the **evaluation phase**. In **WP 5**, the evaluation study (see 3a) is first prepared at the end of implementation and then carried out. In **WP 6** the last missing results are then written down.

In general, the results of the work packages are written down in parallel and continuously with a view to the final elaboration.

9.4 Risks and Contingency Plan

1. Risks in the analysis phase:
   a) With regard to the identification of important functions from existing literature and from the interaction with editors, it is possible that under certain circumstances too much focus is placed on “old” functions rather than thinking out of the box.
2. Risks in the design phase:
   a) As three out of the four graphic editors which are studied for features are desktop-based, it may not readily be possible to transfer some of the identified important functionalities to the context of a web-based application.
   b) Regarding the framework choice for graphical modeling functionalities it may be possible that none of the studied frameworks allows for implementation of enough of the identified

functionalities. In this case only a rather limited set of functions is implemented and the unworkable aspects and their implementation difficulties are critically discussed.

c) Since not all requirements for the editor will be known by the GENIAL! project during this work, it is possible that the choice of architecture may have to be adapted at a later point in time.

d) Due to the general time limitation it is possible that not all of the identified features can be implemented. Thus, the author proposes a categorization of the features into necessary and sufficient and possibly only the necessary features are implemented.

3. Generally, it may be that due to time problems not all points of the above work plan can be carried out. The author will then give a detailed description, guidance and evaluation in the written elaboration, so that they can be reproduced and carried out in future research.

References

[1] Davenport, T. H. and Prusak, L., *Working Knowledge: How Organizations Manage What They Know*. Harvard Business School Press, 1998.

[2] Gruber, T. R., "A Translation Approach to Portable Ontology Specification", *Knowledge Acquisition*, vol. 5, 1993.

[3] Neches, R. and Fikes, R. and Finin, T. and Gruber, T. and Patil, R. and Senator, T. and Swartout, W. R., "Enabling Technology For Knowledge Sharing", *AI Magazine*, vol. 12, no. 3, 1991. [Online]. Available: `https://www.isi.edu/isd/KRSharing/vision/AIMag.html`.

[4] Berners-Lee, T. and Hendler, J. and Lassila, O., "The Semantic Web: a new form of Web content that is meaningful to computers will unleash a revolution of new possibilities", *Scientific American*, vol. 284, no. 5, 2001. [Online]. Available: `https://www-sop.inria.fr/acacia/cours/essi2006/Scientific%20American_%20Feature%20Article_%20The%20Semantic%20Web_%20May%202001.pdf`.

[5] Corcho, O. and Fernández-López, M. and Gómez-Pérez, A., "Ontologies for Software Engineering and Software Technology", in, Calero, C. and Ruiz, F. and Piattini, M., Ed. Springer-Verlag Berlin Heidelberg, 2006, ch. Ontological Engineering: Principles, Methods, Tools and Languages, pp. 1,48. [Online]. Available: `https://books.google.de/books?hl=de&lr=&id=zsWa1tBFVXkC&oi=fnd&pg=PA1&dq=ontolgies+in+engineering+science&ots=ceFdLpwacb&sig=zHx3isqPI4WIHvkXmdhDt1kqFCw#v=onepage&q=ontolgies%20in%20engineering%20science&f=false`.

[6] Kotis, K. and Vouros, G. A., "Human-centered ontology engineering: The HCOME methodology", *Knowledge Information Systems*, vol. 10, no. 1, 2006.

[7] Wieschnowsky, T. and Paulheim, H., "A Visual Tool for Supporting Developers in Ontology-based Application Integration", in *SWESE 2011 : The 7th International Workshop on Semantic Web Enabled Software Engineering; Co-located with ISWC 2011*, 2011.

[8] Westerinen, A. and Tauber, R., "Ontology development by domain experts (without using the "O" word)", *Applied Ontology*, vol. 12, pp. 299,311, 2017.

[9] Liepiņš, R. and Bojārs, U. and Grūzītis, N. and Čerāns, K. and Celms, E., "Towards Self-explanatory Ontology Visualization with Contextual Verbalization", in *International Baltic Conference on Databases and Information Systems*, 2016.

[10] Denaux, R. and Dolbear, C. and Hart, G. and Dimitrova, V. and Cohn, A. G., "Supporting domain experts to construct conceptual ontologies: A holistic approach", *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 9, 2011.

[11] Wiens, V. and Lohmann, S. and Auer, S., "WebVOWL Editor: Device-Independent Visual Ontology Modeling", in *Proceedings of the ISWC 2018 Posters & Demonstrations, Industry and Blue Sky Ideas Tracks. CEUR Workshop Proceedings*, 2018. [Online]. Available: `http://editor.visualdataweb.org/`.

[12] Bārzdiņš, J. and Bārzdiņš, G. and Čerāns, K. and Liepiņš, R. and Sproģis, A., *OWLGrEd*. [Online]. Available: `http://owlgred.lumii.lv`.

[13] *OntoUML*. [Online]. Available: `https://ontouml.org`.

[14] Neurauter, M. and Pinggera, J. and Martini, M. and Burattin, A. and Furtner, M. and Sachse, P. and Weber, B., "Information Systems and Neuroscience. Lecture Notes in Information Systems and Organisation", in, Davis, F. and Riedl, R. and vom Brocke, J. and Léger, P. M. and Randolph, A., Ed. Springer, Cham, 2015, vol. 10, ch. The Influence of Cognitive Abilities and Cognitive Load on Business Process Models and their Creation, pp. 107,115.

[15] Staab, S. and Studer, R., Ed., *Handbook on Ontologies*. Heidelberg: Springer Verlag, 2009.

[16] Weller, K., "Grundlagen der praktischen Information und Dokumentation", in, Kuhlen, R. and Semar, W. and Strauch, D., Ed. Berlin: Walter de Gruyter, 2013, ch. Ontologien, pp. 207,218.

[17] Noy, N. F. and McGuinness, D. L., "Ontology Development 101: A Guide to Creating Your First Ontology", Stanford University, Stanford, CA, Tech. Rep., 2001.

[18] Dahlem, N. and Hahn, A., "User-Friendly Ontology Creation Methodologies - A Survey", in *Proceedings of the Fifteenth Americas Conference on Information Systems, San Francisco, California*, 2009.

[19] University of the Aegean Department of Information and Communication Systems InCoSys Group, *The HCONE (Human Centered ONtology Environment)*. [Online]. Available: `http://www.icsd.aegean.gr/Postgraduates/kkot/hconeweb/default.htm`.

[20] *Protégé Plugin Library*. [Online]. Available: `https://protegewiki.stanford.edu/wiki/Protege_Plugin_Library`.

[21] Vigo, M. and Bail, S. and Jay, C. and Stevens, R., "Overcoming the pitfalls of ontology authoring: Strategies and implications for tool design", *International Journal Human-Computer Studies*, vol. 72, 2014.

[22] Lembo, D. and Pantaleone, D. and Santarelli, V. and Savo, D. F., *EDDY - The Graphol Editor*, online. [Online]. Available: `https://www.obdasystems.com/eddy`.

[23] Kouji KOZAKI, the Institute of Scientific and Industrial Research, Osaka University, *Hozo - Ontology Editor*. [Online]. Available: `http://www.hozo.jp`.

[24] Morshed, A. and Singh, R., "Evaluation and Ranking of Ontology Construction Tools", DIT-University of Trento, Tech. Rep., 2005.

[25] Barriocanal, E. G. and Sicilia, A. M. and Sánchez-Alonso, S., "Usability evaluation of ontology editors", *Knowledge Organization*, vol. 32, no. 1, 2005.

[26] Dzbor, M. and Motta, E. and Buil, C. and Gomez, J. and Goerlitz, O. and Lewen, H., "Developing ontologies in OWL: An observational study", in *OWL: Experiences and Directions Athens, Georgia, USA*, 2006.

[27] Atkinson, C. and Gutheil, M. and Kiko, K., "On the relationship of ontologies and models", in *Proceedings of the 2nd International Workshop on Meta-Modelling (WoMM). LNI 96,Karlsruhe, Germany*, 2006.

[28] Österle, H. and Becker, J. and Frank, U. and Hess, T. and Karagiannis, D. and Krcmar, H. and Loos, P. and Mertens, P. and Oberweis, A. and Sinz, E. J., "Memorandum on design-oriented information systems research", *European Journal of Information Systems*, vol. 20, no. 1, 2011.

[29] *Protégé*. [Online]. Available: `https://protege.stanford.edu/`.

[30] Borst, W., "Construction of Engineering Ontologies", PhD thesis, Institute for Telematica and Information Technology, University of Twente, Enschede, 1997.

[31] Lembo, D. and Pantaleone, D. and Santarelli, V. and Savo, D. F., "Easy OWL Drawing with the Graphol Visual Ontology Language", in *International Conference onPrinciples of Knowledge Representation and Reasoning*, 2016.

[32] Rastogi, N. and Verma, P. and Kumar, P., "Analyzing Ontology Editing Tools for effective semantic Information Retrieval", *International Journal of Engineering Sciences & Research Technology*, vol. 6, no. 5, 2017.

[33] Dahlem, N. and Hahn, A. and Guo, J. and Reinelt, M., "Towards an user-friendly ontology design methodology", in *International Conference on Interoperability for Enterprise Software and Applications China*, 2009.

[34] Lembo, D. and Pantaleone, D. and Santarelli, V. and Savo, D. F., "Eddy: A Graphical Editor for OWL 2 Ontologies", in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, 2016.

[35] Busse, J. and Humm, B. and Lübbert, J. and Moelter, F. and Reibold, A. and Rewald, M. and Schlüter, V. and Seiler, B. and Tegtmeier, E. and Zeh, T., "Actually, What Does "Ontology" Mean?A Term Coined by Philosophy in the Light of Different Scientific Disciplines", *Journal of Computing and Information Technology*, vol. 23, no. 1, 2015.

[36] Slimani, T., "A Study Investigating Typical Concepts and Guidelines for Ontology Building ", *Journal of Emerging Trends in Computing and Information Sciences*, vol. 5, no. 12, 2014.

[37] Malik, Z. H., "Usability Evaluation of Ontology Engineering Tools", in *In the Proceedings of the Computing Conference 2017 in London, UK*, 2017.

[38] Vaishnavi, V. and Kuechler, B. and Petter, S. (Dec. 5, 2018). Design research in information systems (Jan. 2004), [Online]. Available: `http://desrist.org/design-research-in-information-systems`.

[39] Guarino, N. and Oberle, D. and Staab, S., "Handbook on Ontologies", in, Staab, S. and Studer, R., Ed. Heidelberg: Springer Verlag, 2009, ch. What Is an Ontology?, pp. 1,21.

[40] Herre, H. and Heller, B. and Burek, P. and Hoehndorf, R. and loebe, F. and Michalek, H., "General Formal Ontology (GFO)", Institute of Medical Informatics,Statistics and Epidemiology (IMISE) University of Leipzig, Tech. Rep., 2006.

[41] Studer, R. and Benjamins, R. and Fensel, D., "Knowledge Engineering: Principles and Methods", *Data & Knowledge Engineering*, vol. 25, no. 1-2, 1998.