

Semantic Representation of Domain Knowledge for Professional VR Training

Jakub Flotyński¹, Paweł Sobociński¹, Sergiusz Strykowski¹, Dominik Strugała¹, Paweł Buń², Filip Górski², Krzysztof Walczak¹

¹Poznań University of Economics and Business

Niepodległości 10, 61-875 Poznań, Poland

e-mail: flotyński@kti.ue.poznan.pl

²Poznań University of Technology

Piotrowo 3, 60-965 Poznań, Poland

Abstract. Domain-specific knowledge representation is an essential element of efficient management of professional training. Formal and powerful knowledge representation for training systems can be built upon the semantic web standards, which enable reasoning and complex queries against the content. Virtual reality training is currently used in multiple domains, in particular, if the activities are potentially dangerous for the trainees or require advanced skills or expensive equipment. However, the available methods and tools for creating VR training systems do not use knowledge representation. Therefore, creation, modification and management of training scenarios is problematic for domain experts without expertise in programming and computer graphics. In this paper, we propose an approach to creating semantic virtual training scenarios, in which users' activities, mistakes as well as equipment and its possible errors are represented using domain knowledge understandable to domain experts. We have verified the approach by developing a user-friendly editor of VR training scenarios for electrical operators of high-voltage installations.

Keywords: semantic web, ontologies, virtual reality, training, scenarios

1 Introduction

Progress in the quality and performance of graphics hardware and software observed in recent years makes realistic interactive presentation of complex virtual spaces and objects possible even on commodity hardware. The availability of diverse inexpensive presentation and interaction devices, such as glasses, headsets, haptic interfaces, motion tracking and capture systems, further contributes to the increasing applicability of virtual (VR) and augmented reality (AR) technologies. VR/AR applications have become popular in various application domains, such as e-commerce, tourism, education and training. Especially in training, VR offers significant advantages by making the training process more efficient and flexible, reducing the costs, liberating users from acquiring specialized equipment, and eliminating risks associated with training in a physical environment.

Training staff in virtual reality is becoming widespread in various industrial sectors, such as production, mining, gas and energy. However, building useful VR training environments requires competencies in both programming and 3D modeling, as well as domain knowledge, which is necessary to prepare practical applications in a given domain. Therefore, this process typically

involves IT specialists and domain specialists, whose knowledge and skills in programming and 3D modeling are usually low. Particularly challenging is the design of training scenarios, as it typically requires advanced programming skills, and the level of code reuse in this process is low. High-level componentization approaches commonly used in today's content creation tools are insufficient because the required generality and versatility of these tools inevitably leads to high complexity of the content design process. Therefore, the availability of user-friendly tools for domain experts to design VR training scenarios using domain knowledge becomes essential to reduce the required time and effort, and consequently promote the use of VR in training.

A number of solutions enabling efficient modeling of VR content using techniques for domain knowledge representation have been proposed in previous works. In particular, the semantic web provides standardized mechanisms to describe the meaning of any content in a way understandable to both users and software. The semantic web is based on description logics, which permit formal representation of concepts, roles and individuals. Such representations can be subject to reasoning, which leads to the inference of implicit knowledge based on explicit knowledge, as well as queries including arbitrarily complex conditions. These are significant advantages for the creation and management of content by users in different domains. However, usage of the semantic web requires skills in knowledge engineering, which is not acceptable in the practical preparation of VR training. Thus, the challenge is to elaborate a method of creating and managing semantic VR scenarios, which could be employed by users who do not have advanced knowledge and skills in programming, 3D modeling and knowledge engineering.

In this paper, we propose a new method of building and managing VR training scenarios based on semantic modeling techniques with a user-friendly editor. The editor enables domain experts to design scenarios in an intuitive visual way using domain knowledge described by ontologies. Our approach takes advantage of the fact that in concrete training scenes and typical training scenarios, the variety of 3D objects and actions is limited. Therefore, it becomes possible to use ontologies to describe available training objects and actions, and configure them into complex scenarios based on domain knowledge.

The work described in this paper has been performed within a project aiming at the development of a flexible VR training system for electrical operators. All examples, therefore, relate to this application domain. However, the developed method and tools can be similarly applied to other domains, provided that relevant 3D objects and actions can be identified and semantically described.

The remainder of this paper is structured as follows. Section 2 provides an overview of the current state of the art in VR training applications and a review of approaches to semantic modeling of VR content. Section 3 describes an ontology of training scenarios. The proposed method of modeling training scenarios is described in Section 4. An example of a VR training scenario along with a discussion of the results is presented in Section 5. Finally, Section 6 concludes the paper and indicates possible future research.

2 Related Work

2.1 Training in VR

VR training systems enable achieving a new quality in employee training. With the use of VR, it becomes possible to digitally recreate real working conditions with a high level of fidelity. Currently available systems can be categorized into three main groups: desktop systems, semi-immersive systems and fully immersive systems. Desktop systems use mainly traditional presentation and interaction devices, such as a monitor, mouse and keyboard. Semi-immersive systems use advanced VR/AR devices for presentation, e.g., head-mounted displays (HMD), and interaction, e.g., motion tracking. Immersive systems use advanced VR/AR devices for

both presentation and interaction. Below, examples of VR training systems within all of the three categories are presented.

The ALEn3D system is a desktop system developed for the energy sector [1]. The system enables interaction with 3D content displayed on a 2D monitor screen, using a mouse and a keyboard. Scenarios implemented in the system mainly focus on training the operation of power lines and consist of actions performed by line electricians. The system includes two modules: a VR environment and a course manager. The VR environment can operate in three modes: virtual catalog, learning and evaluation. The course manager is a browser application that allows trainers to create courses, register students, create theoretical tests and monitor learning progress.

An example of a semi-immersive system is the IMA-VR system [2]. It enables specialized training in a virtual environment aimed at transferring motor and cognitive skills related to the assembly and maintenance of industrial equipment. The specially designed IMA-VR hardware platform is used to work with the system. The platform consists of a screen and a haptic device. This device allows a trainee to interact and manipulate virtual training scenes. The system records accomplished tasks and statistics, e.g., time, required assistance, errors made and correct steps.

An example of a fully immersive AR system is the training system for repairing electrical switchboards developed by Schneider Electric in cooperation with MW PowerLab [3]. The system is used for training in operation on electrical switchboards and replacement of their parts. The system uses the Microsoft HoloLens HMD. After a user puts on the HMD, the system scans the surroundings for an electrical switchboard. The system can work in two ways: providing tips on a specific problem to be solved or providing general tips on operating or repairing the switchboard.

2.2 Semantic modeling of VR content

A number of works have been devoted to ontology-based representation of 3D content, including a variety of geometrical, structural, spatial and presentational elements. A comprehensive review of the approaches has been presented in [4]. Existing methods are summarized in Table 1. Five of the methods address the low (graphics-specific) abstraction level, while six methods address a high (general or domain-specific) abstraction level. Three of those methods are general—may be used with different domain ontologies. For the methods that address a high abstraction level in specific application domains, the domains are indicated.

Table 1. Comparison of semantic 3D content modeling methods

Approach	Level of Abstraction	
	Low (3D graphics)	High (application domain)
De Troyer et al. [5]–[9]	✓	general
Gutiérrez et al. [10], [11]	✓	humanoids
Kalogerakis et al. [12]	✓	-
Spagnuolo et al. [13]–[15]	-	humanoids
Floriani et al. [16], [17]	✓	-
Kapahnke et al. [18]	-	general
Albrecht et al. [19]	-	interior design
Latoschik et al. [20]–[22]	-	general
Drap et al. [23]	-	archaeology
Trellet et al. [24], [25]	-	molecules
Perez-Gallardo et al. [26]	✓	-

The presented review indicates that there is a lack of a generic method that could be used for creating interactive VR training scenarios in different application domains. The existing

ontologies are either 3D-specific (with focus on static 3D content properties) or domain-specific (with focus on a single application domain). They lack domain-independent conceptualization of actions and interactions, which could be used by non-technical users in different domains to generate VR applications with limited help from graphics designers and programmers. In turn, the solutions focused on 3D content behavior, such as [27], [28], do not provide concepts and roles for representation of training scenarios.

3 Ontological Representation of VR Training Scenarios

A *scenario ontology* has been designed to enable semantic representation of VR training scenarios. The scenario ontology consists of a TBox and an RBox. The TBox is a specification of classes (concepts) used to describe training scenarios. The RBox is a specification of properties (roles) of instances (individuals) of the classes. A particular training scenario is an ABox including instances of TBox classes described by RBox properties. The scenario ontology and particular training scenarios are separate documents implemented using the RDF, RDFS and OWL standards. RDF is the data model for the ontology and scenarios. In turn, RDFS and OWL provide vocabularies, which enable expression of such relations as concept and role inclusion and equivalence, role disjointness, individual equality and inequality, and negated role membership.

The entities specified in the scenario ontology as well as the relations between them are depicted in Fig. 1. The entities encompass classes (rectangles) and properties (arrows) that fall into three categories describing: the workflow of training scenarios, objects and elements of the infrastructure, and equipment necessary to execute actions on the infrastructure.

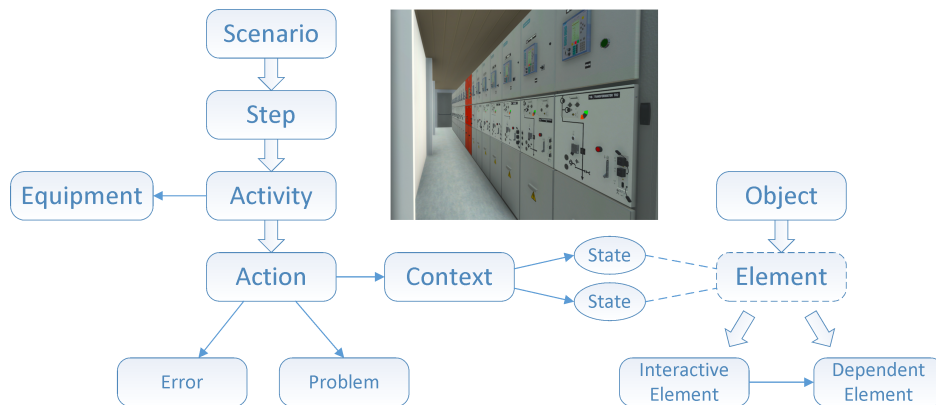


Figure 1. Ontology of VR training scenarios

Every *scenario* is represented by an individual of the *Scenario* class. A scenario consists of at least one *Step*, which is the basic element of the workflow, which consists of at least one *Activity*. Steps and activities correspond to two levels of generalization of the tasks to be completed by training participants. Activities specify equipment required when performing the works. In the VR training environment, it can be presented as a toolkit, from which the user can select the necessary tools. Steps and activities may also specify protective equipment. *Actions*, which are grouped into activities, specify particular indivisible tasks completed using the equipment specified for the activity. Actions are executed on infrastructural components of two categories: *Objects* and *Elements*, which form two-level hierarchies. A technician, who executes an action, changes the *State* of an object's element (called *Interactive Element*), which may affect elements of this or other objects (called *Dependent Elements*). For example, a control panel of a dashboard is used to switch on and off a transformer, which is announced on the panel and influences the infrastructure. N-ary relations between different entities in a scenario are represented by individuals of the *Context* class, e.g., associated actions, elements and states. Non-typical situations in the workflow are modeled using *Errors* and *Problems*.

While errors are due to the user, e.g., a skipped action on a controller, problems are due to the infrastructure, e.g., a controller's failure.

4 Designing VR Training Scenarios

The concept of the method of modeling VR training scenarios is depicted in Fig. 2. The method consists of two main stages, which are accomplished using two modules of the editor we have developed. At the first stage, electricians who directly train new specialists provide primary information about scenarios using the Scenario Editor tool. At the second stage, the information collected from the first stage is used by the managers of technical teams to refine, manage and provide scenarios in their final form using the Semantic Scenario Manager. Next, the final scenarios are used to train specialists with the VR application.

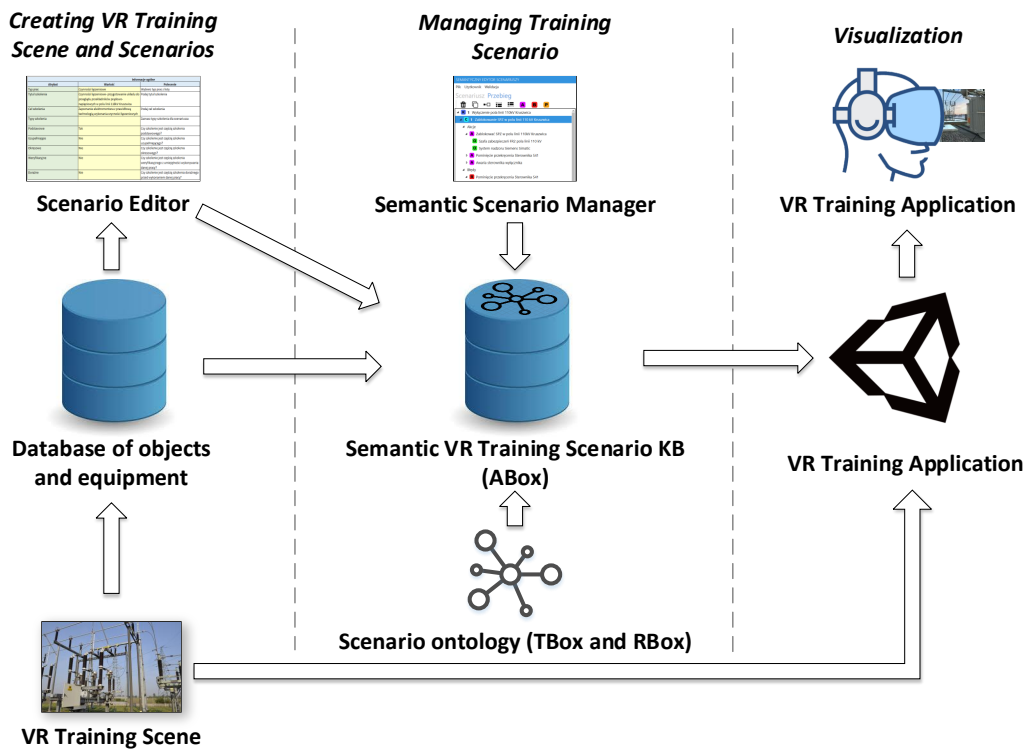


Figure 2. Knowledge base driven design of VR training scenarios

4.1 Scenario Editor

The Scenario Editor is a visual tool based on MS Excel. Its main goal is to enable efficient and user-friendly collection of data about training scenarios by electricians who directly work with trainees and the high-voltage installations.

Scenarios are stored as Excel files based on a specific scenario template. A single scenario is represented by several worksheets, each worksheet contains numerous rows with data. Data in a row is organized in a pair $\langle \text{attribute}, \text{value} \rangle$. Rows containing data relating to the same topic are grouped into sections, where each section is identified by a header. The Scenario Exporter has been implemented as an Excel extension using C# programming language. Its class diagram is presented in Fig. 3.

The *OntologyStore* class is responsible for managing mappings between scenario content (scenario sections and rows within the sections) and elements of the scenario ontology (classes and properties). The mappings are stored in a template file—the same file which is used by the Scenario Editor. While instantiating, the *OntologyStore* class parses the template file and builds in-memory object-oriented representation of the mappings.

mappings. Because, in general, a data tuple can have several mapping units assigned, each data tuple can result in more than one RDF triple generated.

The generated RDF triples are stored in a form of a semantic graph represented by the *Graph* class. An RDF triple is represented by the *Triple* class and consists of three entities: subject, predicate and object. These entities are included within the graph as its nodes and are represented by various classes being implementations of the *INode* interface:

- the *UriNode* class: a node with a full identifier (a name), used to uniquely represent an RDF triple entity within the whole graph,
- the *LiteralNode* class: a node with a literal text value, enriched with optional metadata: data type and language, used to store single data values of scenario content,
- the *BlankNode* class: an anonymous node (without a public identifier), used to group a set of other nodes into a subgraph.

The *IdGenerator* interface defines a method for generating RDF triples with domain-specific identifiers for individuals of objects, elements and states included in a knowledge base. The *IdGenerator* class, which implements this interface, first uses the *IQueryManager* implemented as the *QueryManager* class to query the semantic graph for all mentioned above individuals. Next, it uses the *IdProvider* implemented as the *IdProviderDatabase* class to retrieve the appropriate identifiers from the database of objects and equipment. Finally, RDF triples with the identifiers are generated and asserted into a semantic graph implemented through the *Graph* class.

A semantic graph can be serialized to a text file or saved to a remote triple store. The *TurtleWriter* class is used to serialize a graph to a text file compliant with Turtle syntax.

4.2 Semantic Scenario Manager

The Semantic Scenario Manager is an intuitive visual tool based on Windows Presentation Foundation, which is used by the managers of electricians' teams. Its main goal is to enable refinement and management of the particular training scenarios on the basis of data provided by the electricians using the Scenario Editor.

The Semantic Scenario Manager presents a user with a number of simple and intuitive forms enabling modification of scenario elements. The forms include the names of attributes as well as textboxes or drop-down lists, where the user can provide the necessary information (Fig. 4). The values presented in the drop-down lists are acquired from the scenario ontology. The user needs to provide general information, such as the type of work and a scenario title. Also, the scenario must be classified as elementary, complementary, regular, or verifying. Next, based on the type of work, the user gives information about the works: their category, symbol, technology used and workstation number. The last step is to provide which elements of protective equipment are necessary to complete the training. The user can choose the equipment from a list.

After completing the general information about the scenario, the manager can review and modify the particular steps, activities and actions that trainees need to perform in this scenario. In each scenario, at least one step with at least one activity with at least one action must be specified (cf. Section 3). Actions are associated with interactive and dependent objects' elements as well as possible problems and errors that may occur during the action.

The manager can refine and manage the details of the scenario by editing its tree view, which is a widespread and intuitive form of presentation of hierarchical data (Fig. 5). The hierarchy encompasses the scenario steps, activities, actions, problems, errors and objects, which are distinguished by different icons. The user can expand and collapse the list of sub-items for every item in the tree. The user can also visually add, modify and delete the items in

Figure 4. Basic scenario data in the Semantic Scenario Manager

the tree using the toolbar and the context menu. The order of the steps, activities and actions can be altered by dragging and dropping.

Figure 5. Scenario details tab with the tree view and a form for an activity

During the scenario design, the manager can potentially make a mistake leading to unexpected results in the VR training scene. For that reason, the Semantic Scenario Manager validates the entire scenario against the Scenario Ontology (cf. Section 3) to check whether the scenario is correct. The validation is the consistency checking process on the Scenario Ontology combined with the ABox describing the scenario. It verifies multiple elements of the scenario, including mandatory fields and permitted values, the number of steps, activities and actions, as well as relations between individual instances of classes. The Semantic Scenario Manager highlights the incorrect attributes and the encompassing tree items.

5 Demonstration and Discussion

Training of employees in practical industrial environments requires designing new and modifying existing training scenarios efficiently. In practice, the number of scenarios is by far larger than the number of training scenes. One of the possible applications of our approach is the representation of the training of operators of high-voltage installations. In this case, typically, one 3D model of an electrical substation is associated with at least a dozen different scenarios. These scenarios include learning daily maintenance operations, reactions to various problems that may occur in the installation as well as reactions to infrastructure malfunction.

In the presented approach, all scenarios are knowledge bases structured according to the generic scenario ontology. The scenario ontology consists of 343 axioms, 18 classes, 34 object properties and 47 datatype properties, which can be used in different scenarios. A scenario knowledge base is an ABox specifying a concrete training scenario consisting of steps, activities and actions, along with its elements and infrastructure objects, which are described by classes and properties specified in the scenario ontology (Fig. 6). Scenario knowledge bases are encoded in OWL/Turtle.

To perform training, a scenario knowledge base is imported into the VR Training Application by an importer module, which – based on the scenario KB – generates the equivalent object model of the scenario. An example view of a user executing the "Karczyn" VR training scenario action is presented in Fig. 7.

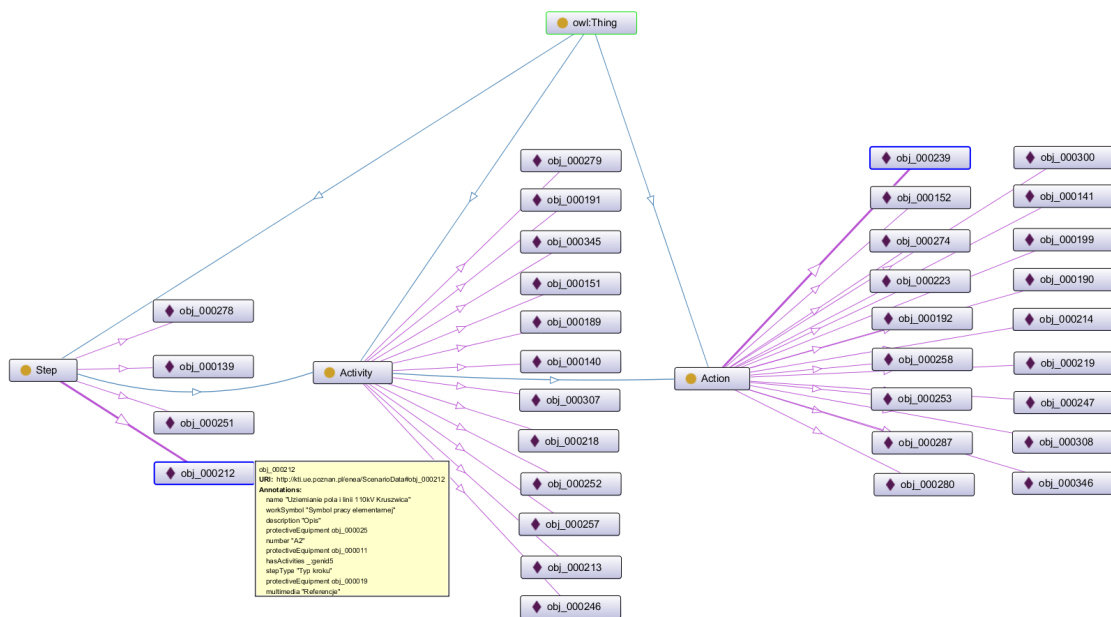


Figure 6. VR training scenario represented as a semantic knowledge base (fragment)

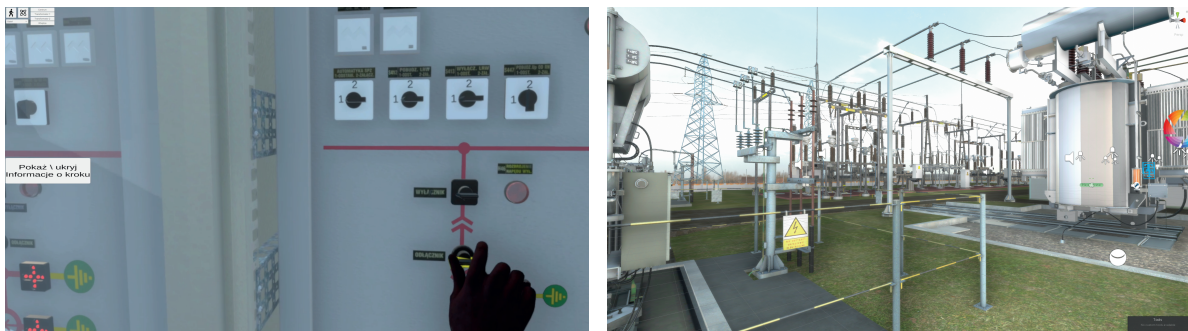


Figure 7. VR training scenario – control room view (left), outside view (right)

The example scenario "Karczyn" covers the preparation of a trainee for specific maintenance work and consists of 4 steps, 11 activities and 17 actions. For each action, there are dependent objects (44 in case of this scenario). For each step, activity, action and object, the scenario provides specific attributes (9-10 for each item). For each attribute, the name, value, command and comment are provided. In total, the specification of the course of the scenario consists of 945 rows in Excel. In addition, there are 69 rows of specification of errors and 146 rows of specification of problems. The scenario also covers protective equipment, specific work equipment and others.

The generic scenario ontology (TBox) encoded in OWL takes 1,505 lines of code and 55,320

bytes in total. The "Karczyn" scenario saved in Turtle (which is a more efficient way of encoding ontologies and knowledge bases) has 2,930 lines of code and 209,139 bytes in total.

Implementation of the "Karczyn" scenario directly as a set of Unity 3D C# scripts would lead to very complex code, difficult to verify and maintain even by a highly-proficient programmer. The design of such a scenario is clearly beyond the capabilities of most domain experts dealing with the everyday training of electrical workers.

An important aspect to consider is the size of the scenario representations. The total size of the "Karczyn" Unity 3D project is 58 GB, while the size of the executable version is only 1.8 GB. Storing 20 scenarios in editable form as Unity projects would require 1.16 TB of disk space. Storing 20 scenarios in the form of semantic knowledge bases requires only 4MB of storage space (plus the size of the executable application).

The use of semantic knowledge bases with a formal ontology described in this paper enables the concise representation of training scenarios and provides means of editing and verifying scenarios correctness with user-friendly and familiar tools.

6 Conclusions and Future Works

The approach proposed in this paper enables the semantic representation of training scenarios, which is independent of particular application domains. The representation can be used in various domains when accompanied by domain-specific knowledge bases and 3D models of objects. In this regard, it differs from the approaches summarized in Table 1, which are not related to training, even if they permit representation of 3D content behavior.

The approach enables flexible modeling of scenarios at a high level of abstraction using concepts specific to training instead of forcing the designer to use low-level programming with techniques specific to computer graphics. The presented editor, in turn, enables efficient and intuitive creation and modification of the scenarios by domain experts. Hence, the method and the tool make the development of VR applications, which generally is a highly technical task, attainable to non-technical users allowing them to use the terminology of their domains of interest in the design process.

Future works include several elements. First, the environment will be extended to support collaborative creation of scenarios by distributed users. Second, we plan to extend the training application to support not only the training mode, but also the verification mode of operation with appropriate scoring based on user's performance. Finally, we plan to extend the scenario ontology with concepts of parallel sequences of activities, which can be desirable for multi-user training, e.g., in firefighting.

Acknowledgments

The research work presented in this paper has been supported by the European Union from the European Regional Development Fund within the Smart Growth Operational Programme 2020-2024. The project is executed within the priority axis "Support for R&D Activity of Enterprises" of the National Centre for Research and Development under the contract POIR.01.01.01-00-0463/18.

References

- [1] R. Patoni. (). "Alen 3D," [Online]. Available: <https://www.scribd.com/document/245796121/ALEN-3D>.

- [2] N. Gavish, T. Gutiérrez, S. Webel, J. Rodríguez, M. Peveri, U. Bockholt, and F. Tecchia, "Evaluating virtual reality and augmented reality training for industrial maintenance and assembly tasks," *Interactive Learning Environments*, vol. 23, no. 6, pp. 778–798, 2015. DOI: 10.1080/10494820.2013.815221. eprint: <https://doi.org/10.1080/10494820.2013.815221>.
- [3] Schneider Electric. (2017). "HoloLens Application on Premset," [Online]. Available: <https://www.youtube.com/watch?v=RpXyagutoZg>.
- [4] J. Flotyński and K. Walczak, "Ontology-Based Representation and Modelling of Synthetic 3D Content: A State-of-the-Art Review," *Computer Graphics Forum*, vol. 35, pp. 329–353, 2017. DOI: 10.1111/cgf.13083.
- [5] W. Bille, O. De Troyer, B. Pellens, and F. Kleinermann, "Conceptual modeling of articulated bodies in virtual environments," in *Proceedings of the 11th International Conference on Virtual Systems and Multimedia (VSMM)*, H. Thwaites, Ed., Archaeolingua, Ghent, Belgium: Archaeolingua, 2005, pp. 17–26.
- [6] F. Kleinermann, O. De Troyer, H. Mansouri, R. Romero, B. Pellens, and W. Bille, "Designing semantic virtual reality applications," in *In Proceedings of the 2nd INTUITION International Workshop, Senlis, 2005*, pp. 5–10.
- [7] B. Pellens, O. De Troyer, W. Bille, F. Kleinermann, and R. Romero, "An ontology-driven approach for modeling behavior in virtual environments," in *Proceedings of On the Move to Meaningful Internet Systems 2005: Ontology Mining and Engineering and its Use for Virtual Reality (WOMEUVR 2005) Workshop*, R. Meersman, Z. Tari, and P. Herrero, Eds., Springer-Verlag, Agia Napa, Cyprus: Springer-Verlag, 2005, pp. 1215–1224.
- [8] O. De Troyer, F. Kleinermann, B. Pellens, and W. Bille, "Conceptual modeling for virtual reality," in *Tutorials, posters, panels and industrial contributions at the 26th Int. Conference on Conceptual Modeling - ER 2007*, J. Grundy, S. Hartmann, A. H. F. Laender, L. Maciaszek, and J. F. Roddick, Eds., ser. CRPIT, vol. 83, Auckland, New Zealand: ACS, 2007, pp. 3–18.
- [9] O. De Troyer, F. Kleinermann, H. Mansouri, B. Pellens, W. Bille, and V. Fomenko, "Developing semantic VR-shops for e-Commerce.," *Virtual Reality*, vol. 11, no. 2-3, pp. 89–106, 2007.
- [10] M. Gutiérrez, "Semantic virtual environments, EPFL," 2005.
- [11] M. Gutiérrez, D. Thalmann, and F. Vexo, "Semantic virtual environments with adaptive multimodal interfaces.," in *MMM*, Y.-P. P. Chen, Ed., IEEE Computer Society, Jan. 26, 2005, pp. 277–283, ISBN: 0-7695-2164-9.
- [12] E. Kalogerakis, S. Christodoulakis, and N. Moutoutzis, "Coupling ontologies with graphics content for knowledge driven visualization," in *VR '06 Proc. of the IEEE conference on Virtual Reality*, Alexandria, Virginia, USA, Mar. 2006, pp. 43–50.
- [13] M. Attene, F. Robbiano, M. Spagnuolo, and B. Falcidieno, "Semantic Annotation of 3D Surface Meshes Based on Feature Characterization," in *Proceedings of the Semantic and Digital Media Technologies 2nd International Conference on Semantic Multimedia*, ser. SAMT'07, Genoa, Italy: Springer-Verlag, 2007, pp. 126–139, ISBN: 978-3-540-77033-6.
- [14] F. Robbiano, M. Attene, M. Spagnuolo, and B. Falcidieno, "Part-Based Annotation of Virtual 3D Shapes," *2013 International Conf. on Cyberworlds*, vol. 0, pp. 427–436, 2007.
- [15] M. Attene, F. Robbiano, M. Spagnuolo, and B. Falcidieno, "Characterization of 3D Shape Parts for Semantic Annotation," *Comput. Aided Des.*, vol. 41, no. 10, pp. 756–763, Oct. 2009, ISSN: 0010-4485. DOI: 10.1016/j.cad.2009.01.003.

- [16] L. De Floriani, A. Hui, L. Papaleo, M. Huang, and J. Hendler, "A semantic web environment for digital shapes understanding," in *Semantic Multimedia*, Springer, 2007, pp. 226–239.
- [17] L. Papaleo, L. De Floriani, J. Hendler, and A. Hui, "Towards a semantic web system for understanding real world representations," in *Proceedings of the Tenth International Conference on Computer Graphics and Artificial Intelligence*, 2007.
- [18] P. Kapahnke, P. Liedtke, S. Nesbigall, S. Warwas, and M. Klusch, "ISReal: An Open Platform for Semantic-Based 3D Simulations in the 3D Internet," in *International Semantic Web Conference (2)*, 2010, pp. 161–176.
- [19] S. Albrecht, T. Wiemann, M. Günther, and J. Hertzberg, "Matching CAD object models in semantic mapping," in *Proceedings ICRA 2011 Workshop: Semantic Perception, Mapping and Exploration, SPME*, 2011.
- [20] M. Fischbach, D. Wiebusch, A. Giebler-Schubert, M. E. Latoschik, S. Rehfeld, and H. Tramberend, "SiXton's curse - Simulator X demonstration," in *Virtual Reality Conference (VR), 2011 IEEE*, M. Hirose, B. Lok, A. Majumder, and D. Schmalstieg, Eds., 2011, pp. 255–256. [Online]. Available: <http://dx.doi.org/10.1109/VR.2011.5759495>.
- [21] M. E. Latoschik and H. Tramberend, "Simulator X: A Scalable and Concurrent Software Platform for Intelligent Realtime Interactive Systems," in *Proceedings of the IEEE VR 2011*, 2011.
- [22] D. Wiebusch and M. E. Latoschik, "Enhanced Decoupling of Components in Intelligent Realtime Interactive Systems using Ontologies," in *Software Engineering and Architectures for Realtime Interactive Systems (SEARIS), proceedings of the IEEE Virtual Reality 2012 workshop*, 2012, pp. 43–51.
- [23] P. Drap, O. Papini, J.-C. Sourisseau, and T. Gambin, "Ontology-based photogrammetric survey in underwater archaeology," in *European Semantic Web Conference*, Springer, 2017, pp. 3–6.
- [24] M. Trellet, N. Férey, M. Baaden, and P. Bourdot, "Interactive visual analytics of molecular data in immersive environments via a semantic definition of the content and the context," in *Immersive Analytics (IA), 2016 Workshop on*, IEEE, 2016, pp. 48–53.
- [25] Trellet, M., Férey, N., Flotyński, J., Baaden, M., Bourdot, P., "Semantics for an integrative and immersive pipeline combining visualization and analysis of molecular data," *Journal of Integrative Bioinformatics*, vol. 15 (2), pp. 1–19, 2018.
- [26] Y. Perez-Gallardo, J. L. L. Cuadrado, Á. G. Crespo, and C. G. de Jesús, "GEODIM: A Semantic Model-Based System for 3D Recognition of Industrial Scenes," in *Current Trends on Knowledge-Based Systems*, Springer, 2017, pp. 137–159.
- [27] J. Flotyński, M. Krzyszkowski, and K. Walczak, "Semantic Composition of 3D Content Behavior for Explorable Virtual Reality Applications," in *Proc. of EuroVR 2017, LNCS*, J. Barbic, M. D'Cruz, M. E. Latoschik, M. Slater, and P. Bourdot, Eds., Springer, 2017, pp. 3–23. DOI: 10.1007/978-3-319-72323-5_1.
- [28] J. Flotyński and K. Walczak, "Knowledge-based representation of 3D content behavior in a service-oriented virtual environment," in *Proceedings of the 22nd International Conference on Web3D Technology, Brisbane (Australia), June 5-7, 2017*, ACM, New York, 2017, Article No 14, ISBN: 978-1-4503-4955-0. DOI: 10.1145/3055624.3075959.