Appendix 2

National Research Program

"Cyber-physical systems, ontologies and biophotonics for safe&smart city and society" (SOPHIS)

Project No.2
"Ontology-based knowledge engineering technologies suitable for web environment"

Scientific report

Periods 1–4

2014–2017

Introduction

The goal of the project No.2 is to develop the scientific expertise of the next generation IT systems by researching and further developing novel competitive model-based information technologies and their applications in modern web environment and to transfer the created expertise and technologies to concrete domains of Latvia's economics, as well as introducing them into the higher education study process.

The main tasks of the stage 4 are:

- Further development of data ontology depiction methods and approbation on real medical data.
- Research of ontology-based linked data and further development of their application in e-Government and e-Medicine domains.
- Fast query language usage testing on Children's Clinical University Hospital data.
- Further development of web-based methods for modeling of hard-to-formalize systems
- FrameNet micro-relation ontology formalization in the form of AMR (Abstract Meaning Representation) and development of appropriate machine learning methods for semantic analysis of text. Research of application of these innovative methods in other areas, e.g., text generation and robotics.
- Development of methods for transformation of knowledge structures and approbation of the prototype of intelligent structural modelling tool I4S 2.0 in the study process for working with knowledge structures (concept maps)
- Development, combination, and usage of knowledge structure models for decision making in multi-agent systems and intelligent tutoring systems
- Model, process, enterprise architecture and other knowledge/artefact amalgamation in FREEDOM framework and development of the methodics for requirements engineering knowledge/artefact maintenance and distribution
- Improvement and approbation of the framework and methodology for integration of semantic web services in traditional web portals for usage in various problem domains.
- Development of experimental data visualization and browsing software accordingly to the capabilities of the infrastructure of the monitor wall.
- Further development of the runtime verification methods by building the ticket reservation system's runtime verification model.
- Long-term technological prognosis for R&D directions developed in the project

The goal of this report is to summarize the main scientific and practical results of the project's stages 1-3 and describe the results of the current reporting period (stage 4) in more details. The parts of scientific results which are adequately presented in the corresponding publications will not be described in detail in this report.

2.1. Ontology based tools for knowledge analysis and mining semantics of natural language

The Section 2.1 describes scientific results of SOPHIS program Project No. 2 that were obtained by the researchers of the **Institute of Mathematics and Computer Science (IMCS), University of Latvia (UL)**.

The studies were focused on the research and development of:

- the ontology-based modelling technologies and tools for knowledge analysis suitable for web environment;
- methods of semantic web and computational linguistics for understanding data collected in a natural language, such as FrameNet situation formalization together with CDC (Cross Document Co-reference) approach.

2.1.1. Research and development of the ontology-based modelling technologies and tools for knowledge analysis suitable for web environment.

2.1.1.1. Development of the ontology- and web technology-based ad-hoc query language (stages 1-3).

In the year 2014 (stage 1) the development of the theoretical background for the ontology- and web technology-based ad-hoc query language has been carried out. So called "3-How" problem has been explored:

1) how to depict a data ontology for it to be easily understandable by a domain expert;

2) how to use such ontology as a base, on which one can build easy-to-use query language that can be exploited by the domain expert directly (without involving a programmer);

3) how to implement such a language efficiently so that one can get answers to typical queries in time less than a second (on data volume of several GBs, e.g., Children's Clinical University Hospital's (CCUH) one year data). The results have been described in [1].

The development of the ontology- and web technology-based controlled natural ad-hoc query language which can be used directly by end-user (without involvement of the programmer) has been started. The first and most essential result: we have found six controlled natural language query templates (supplemented with a formal concept of scalar expression) that covers practically all ad-hoc queries one can think of for needs of hospital management (we assume the managers have sufficient MS Excel skills). We have tested this hypothesis on real CCUH data (year 2014) and real queries that were needed to generate the review and analysis of year 2014 in one particular CCUH clinic (intensive therapy clinic). The experiment approved the hypothesis – 100% of necessary query coverage was achieved. In the yeara 2015 and 2016 (stage 2 and 3) of the project the development the fast ad-hoc query language continued. The ad-hoc querying process is slow and error prone due to inability of business experts of accessing data directly without involving IT experts. The problem lies in the complexity of means used to query data. We have proposed a new natural

language- and semistar ontology-based ad-hoc querying approach which lowers the steep learning curve required to be able to query data. The proposed approach shortens the time needed to master the ad-hoc querying and to gain the direct access to data by business experts, thus facilitating the decision making process in enterprises, government institutions and other organizations. We have also proposed an efficient implementation architecture for the parallel execution of ad-hoc queries based on distributed granular ontologies. Approbation of the language was performed on data of year 2015 of Riga Children's Clinical University Hospital for the needs of intensive care ward. Answers to all ad-hoc queries that were formulated for the needs of analysis of operation of the intensive care ward in year 2015 were obtained using the proposed query language in online mode. When the necessary question was formulated in natural language it took couple of minutes to reformulate it in the proposed language. All the queries were executed in less than 0.3 seconds on average for the data amount of one typical hospital in Latvia. This would match the performance of about 1 second per query if we took data from all the hospitals in Latvia (and take into account the potential to execute the query calculation process in parallel).These results are described in more detail in [2, 3, 4]. Additionally, we have introduced a new construct within the fast query language – the view definition mechanism –, and we have implemented it efficiently. This new feature allows end-users to create new subclasses of ontology classes by defining them using only constructs of the query language. The prototype of the fast ad-hoc query system is available upon request in the IMCS, UL. In the year 2016 (stage 3) the development of data access control mechanism based on data ontologies and web technologies, to be used for the implementation of the fast ad-hoc query language has been carried out. This mechanism uses an extended concept of a user role, where access rights are being defined using slightly extended facilities of the query language itself. The proposed access control mechanism permits to define in a simple way all typical access constraints in medical domain, that e.g. a Responsible Physician can see only data of the patients which have been treated by him, a ward manager can see data for patients being treated in the ward managed by him etc. A research has been done also to evaluate the impact of the access right application on query system performance, the expected slowdown is no more than 0.5 seconds on a query execution.

References.

[1].J. Barzdins, E. Rencis and A. Sostaks, "Fast Ad Hoc Queries Based on Data Ontologies." In: H.M. Haav, A. Kalja, T. Robal (Eds.), Frontiers of AI and Applications, Vol. 270, Databases and Information Systems VIII, IOS Press, pp. 43-56, 2014.

[2] J.Barzdins, E.Rencis, A.Sostaks. "Self-service Ad-hoc Querying Using Controlled Natural Language" Submitted to the 12th International Baltic Conference on Databases and Information Systems, Riga, Latvia, 4-6 July 2016.

[3] J.Barzdins, M.Grasmanis, E.Rencis, A.Sostaks, A.Steinsbekk, Towards a More Effective Hospital: Helping Health Professionals to Learn from their Own Practice by Developing an Easy to use Clinical Processes Querying Language. // J.E.Q. Varajão et. al. (Eds.), Procedia Computer Science, Vol. 100, Elsevier, pp. 498-506, 2016.

[4] J.Barzdins, M.Grasmanis, E.Rencis, A.Sostaks, J.Barzdins, Ad-Hoc Querying of Semistar Data Ontologies Using Controlled Natural Language. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Frontiers of AI and Applications, Vol. 291, Databases and Information Systems IX, IOS Press, pp. 3-16, 2016., http://ebooks.iospress.com/volumearticle/45695

2.1.1.2. Development of the web technology-based tool building technologies and methods for modeling of complex, hard-to-formalize systems (stages 1-3).

In the year 2014 (stage 1) the development of the web technology-based tool building technologies and methods for modeling of complex, hard-to-formalize systems has started. We have developed experimental graphical tool-building platform that can be used to modelling in web such systems that are difficult to formalize. The platform provides interactivity, collaboration, different machines support (computers, tablets, and smartphones), reactivity and live HTML. Development of metamodel specialization methods and their application to building of domain specific language tools for web environment. In the year 2016 (stage 3) a new metamodeling method – the metamodel specialization method – has been developed. This method is based on standard UML facilities – class diagrams, class and association specialization and OCL constraints. An application of metamodel specialization method to building graphical DSL tools has been developed. This application results in a new kind of a platform for building DSL tools. It permits to build a complete definition of the chosen DSL tool by adding appropriate OCL constraints. To compare, for traditional metamodel instantiation applications when building a more complicated DSL tool, as a rule it is necessary to dive into the internal implementation model of the corresponding universal engine, thus making the platform usage much more complicated. Research results have been published in [1, 2, 3, 4, 5]

References.

[1] A.Kalnins, J.Barzdins, Metamodel Specialization for DSL Tool Building. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Databases and Information Systems, 12th International Baltic Conference, DB&IS 2016, Riga, Latvia, July 4-6, 2016, Proceedings, Communications in Computer and Information Science Vol. 615, Springer, pp.68-82, 2016. (SCOPUS)

[2] A.Sprogis, DSML Tool Building Platform in WEB. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Databases and Information Systems, 12th International Baltic Conference, DB&IS 2016, Riga, Latvia, July 4-6, 2016, Proceedings, Communications in Computer and Information Science Vol. 615, Springer, pp.99-109, 2016. (SCOPUS)

[3] A.Kalnins, J.Barzdins, Metamodel specialization for graphical modeling language support. // In: Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems. ACM, pp.103-112, 2016. (SCOPUS)

[4] A.Kalnins, J.Barzdins, Metamodel Specialization for Diagram Editor Building. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Frontiers of AI and Applications, Vol. 291, Databases and Information Systems IX, IOS Press, pp. 87-100, 2016. (to be indexed SCOPUS), http://ebooks.iospress.com/volumearticle/45702

[5] A.Sprogis, ajoo: WEB Based Framework for Domain Specific Modeling Tools. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Frontiers of AI and Applications, Vol. 291, Databases and Information Systems IX, IOS Press, pp. 115-126, 2016. (SCOPUS), http://ebooks.iospress.com/volumearticle/45704

2.1.1.3 Research of the ontology-based linked data technologies for applications of e-government and e-health (stage 1).

In the year 2014 (stage 1) the research of the ontology-based linked data technologies for applications of e-government and e-health has started. Conclusion: Use-cases of ontology-based linked data grow fairly rapidly in the world (around 30% in a year). These use-cases are generally based on open data in form of RDF, they mainly refer to providing different (not known a priori) research, statistics, reviews.

2.1.1.4 Further development of data ontology depiction methods and approbation on real medical data (stage 4).

When we think about the various data representation formats from which to choose, several options can be available. One of the most exploited data storage formats is the relational database, because usually the data can indeed be represented in the form of ER model. If we choose to use this data storage format, we are later able to query the ontology using the SQL as a query language. This is a common solution, and thus it is quite easily to implement it.

However, we must take into account not only the ease of implementation of the chosen solution, but also its friendliness to end-users that are not IT specialists. In this project we are focusing our attention on healthcare professionals (managers and physicians) as main types of users of our system, so we cannot assume that for them the ER model would be the best representation of healthcare data in a natural and understandable way. Since ER model is almost never granular (naturally dividable into data slices), it is usually not easily understandable by non-programmers [1]. Moreover, regardless of the fact that the SQL language was initially designed to be used by standard end-users, hardly any non-programmer has nowadays acquired the necessary skills to be able to understand SQL queries, not to mention writing them himself.

Therefore we have to cope with at least two challenges: 1) how to depict the data ontology to be easily understandable by healthcare professionals; 2) how to develop a query language based on this representation of the underlying data ontology, such that a healthcare professional could formulate queries himself and understand their

answers. Finally, if we had developed such a user-friendly query language, we would then encounter also the third challenge: how to implement the query language efficiently enough in order to get the answer to a sufficiently wide class of queries in a reasonable time. These three challenges together form the so-called "3How" problem which we have described in more detail in our previous work [1].

If we now think about the most suitable format for storing healthcare data, we should look at how these data were stored before they were digitalized. When the information about patients was filled in by hand, hospitals used so-called patient cards where each patient had his separate card and each card contained information about each occurrence of this patient in the hospital, and each occurrence contained information about the treatments provided for the patient in this particular occurrence and so on. This division into smaller and smaller subdivisions is a very natural way of storing healthcare data, and it is also very familiar to healthcare professionals. Therefore we have chosen exactly such structure to be the basis of the data ontology that solves the first challenge of the abovementioned "3How" problem. The described structure is known in literature as the reversed star data schema, because in it "certain key characteristics of the classic star schema is 'reversed'" [2]. Indeed, in typical situations we always have one central class (the class "Patient" in this case), from which several paths can lead to other connected classes having the relation one-to-many, as can be seen in Fig. 2.1.1.1 It is also known that any database organized in the third normal form can be converted to a reversed star schema thus making the reversed star ontologies very powerful [3]. In addition to classical reversed star ontologies we also allow adding classes outside the star devoted for registers and classifiers (classes "CPhysician", "CDiagnosis" and "CManipulation" in Fig. 2.1.1.1). We call such enriched ontologies the semistar ontologies, and we have described them in detail in several publications [1, 4-7]. The simplified version of a semistar ontology seen in Fig. 2.1.1.1 has been introduced for use in Riga Children's Clinical University Hospital (RCCUH).

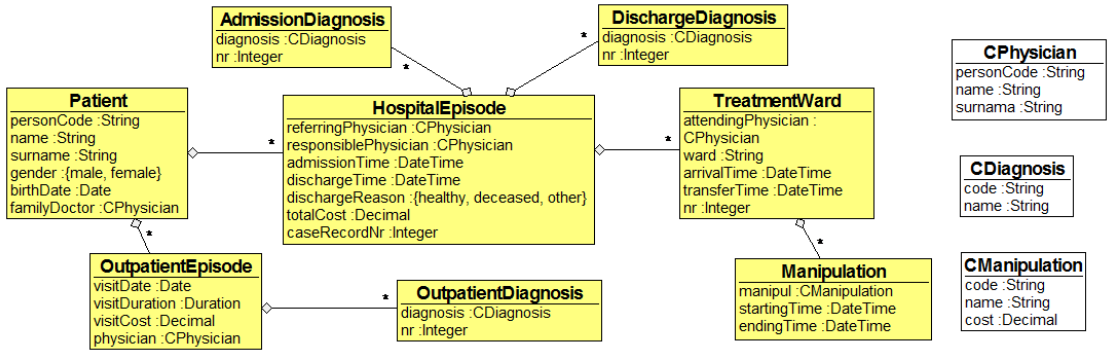


Figure 2.1.1.1. Semistar ontology exploited in Riga Children's Clinical University Hospital.

Generally speaking the semistar ontologies have only one type of associations between basic classes (the ones forming the star – depicted with yellow background in Fig. 2.1.1.1) – the "has" relation (e.g. Patient has HospitalEpisodes, HospitalEpisode has TreatmentWards etc.). As mentioned above, besides basic classes a semistar ontology usually also contains other classes called the classifiers (depicted with white background in Fig. 2.1.1.1). Associations between basic classes and classifier classes are coded as attributes (e.g. familyDoctor: CPhysician).

Semistar ontology is a practically important and expressive subset of all data ontologies, and practical use-cases often exploit exactly this type of ontologies. As can be seen in Fig. 2.1.1.1, hospital ontology viewed from patients' and physicians' point of view comes out to be a semistar ontology. Our experience shows that even in more general cases, when some ontology is not a semistar ontology, one can usually find an important subset of it to comply to principles of semistar ontology. We can always think of a semistar ontology as of a subject-oriented ontology where the role of the subject can be performed by a patient (in case of the medical domain), a customer (in case of some service domain), etc.

We allow attributes of basic classes to have two kinds of data types – the primitive types and the classifiers. We use the following predefined data types and operations:

- Integer (e.g. 75, -75), Real (e.g. 0.75, -75.0), operations: +, -, *, /;
- Boolean (true, false), operations: and, or, not;
- String (e.g. "abc"), operations: substring (e.g. "abcde".substring(2,3)="bc");
- Date (e.g. 2015.06.17), unary operations: year(), month(), day(), dayOfWeek(), binary operation: - (e.g. 2015.06.17-2015.05.12 = 1M5D);
- DateTime (e.g. 2015.06.17T10:45), unary operations: year(), month(), day(), hour(), minute(), second(), date(), binary operation: - (subtraction);
- Duration (e.g. 3Y4M5DT6H7M30.25S), unary operations: years(), months(), days(), hours(), minutes(), seconds().

A very important concept here is the *attribute expression*. In the simplest case the attribute expression is just any attribute found in the ontology. If some attribute *a* has a classifier class as its data type and this classifier class has some attribute *k*, then also *a.k* denotes a valid attribute expression, and its data type will be that of attribute *k*. If *x* is an instance of some class, for which attribute *a* is defined, then also *x.a* (or *x.a.k*, if type of *a* is a classifier class) denotes a valid attribute expression. We can build more complex attribute expressions from simpler ones using the abovementioned operations allowed for the given data types. Some examples of attribute expressions: *personCode*, *x.personCode*, *x.familyDoctor.surname*, *x.admissionTime.month()*, *(dischargeTime-admissionTime).days()*, etc.

We can now compare two attribute expressions (or constants) to obtain *attribute conditions*, e.g. *personCode=250285-10507*, *x.personCode=250285-10507*, *personCode.substring(1,4)=2502*, *dischargeTime-admissionTime>25d* (meaning – 25 days), *x.birthDate.year()>=1985*, *familyDoctor<>nil* (a family doctor exists), etc.

Attribute expressions and attribute conditions are one of the most important concepts in the process of developing the controlled natural language-based user-friendly query language that exploits the underlying semistar data ontology.

The ontology seen in Fig. 2.1.1.1 is a very simplified version of data ontology really used in Riga Children's Clinical University Hospital. The actually used ontology consists of 25 classes and 142 attributes, and it can be seen in Fig. 2.1.1.2 (this is the actual ontology, therefore all the names are in Latvian).

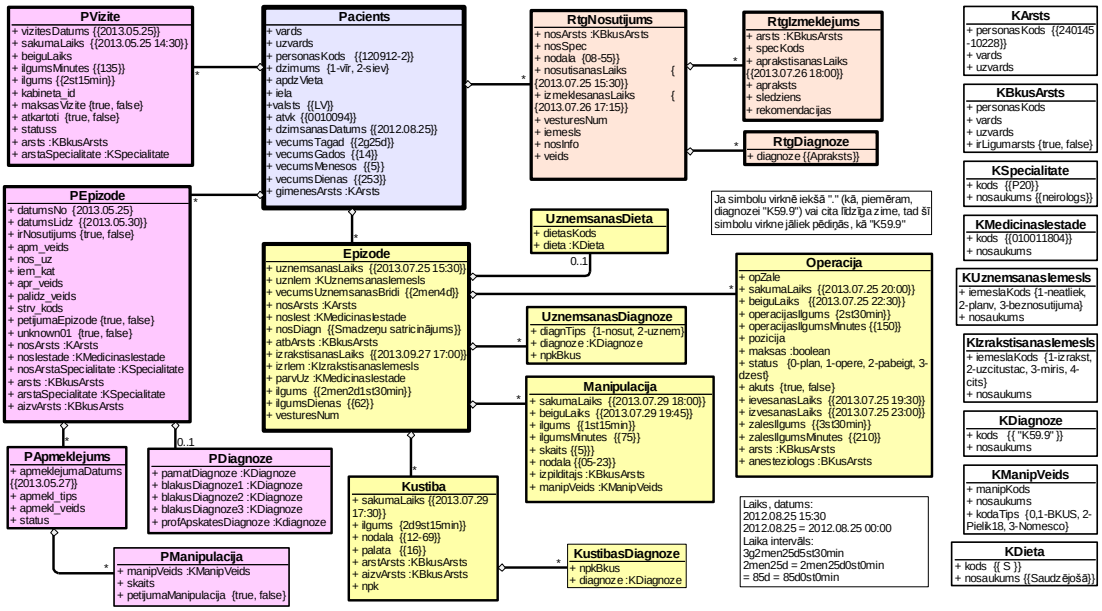


Figure 2.1.1.2. Real ontology used in Riga Children's Clinical University Hospital.

Semistar ontologies are by their nature granular, that is – they can be naturally divided into slices [1, 4] where each slice contains concluded information about one particular patient. This feature allows developing a new kind of querying language that would solve the second challenge of the abovementioned "3How" problem. As mentioned above, the language would be based on a controlled natural language, and it would use concepts from the underlying semistar ontology which is familiar to the domain experts who will later work with the language.

References

1. Barzdins, J., Rencis, E., & Sostaks, A. (2014). Data Ontologies and Ad Hoc Queries: a Case Study. In: H.M. Haav, A. Kalja, T. Robal (Eds.) Proc. of the 11th International Baltic Conference, Baltic DB&IS, 55-66, TUT Press.
2. Kasprzyk, A., Keefe, D., Smedley, D. et al. (2004). EnsMart: a generic system for fast and flexible access to biological data. In: Genome Res., 14, 160-169.
3. Zhang, J. et al. (2011). BioMart: a data federation framework for large collaborative projects. In: Database: The Journal of Biological Databases and Curation, bar038, http://doi.org/10.1093/database/bar038.
4. Barzdins, J, Rencis, E., & Sostaks, A. (2013). Granular Ontologies and Graphical In-Place Querying. Short Paper Proceedings of the PoEM, CEUR-WS, 1023, 136-145.
5. Barzdins, J., Grasmanis, M., Rencis, E., Sostaks, A., & Barzdins, J. (2016). Self-service Ad-hoc Querying Using Controlled Natural Language. In: G. Arnicans et al. (Eds.) Proc. of the 12th International Baltic Conference, Baltic DB&IS, 18-34, CCIS 615.
6. Barzdins, J., Grasmanis, M., Rencis, E., Sostaks, A., & Barzdins, J. (2016). Ad-hoc Querying of Semistar Data Ontologies Using Controlled Natural Language. In: Frontiers in Artificial Intelligence and Applications, Vol. 291 (DOI: 10.3233/978-1-61499-714-6-3), Databases and Information Systems IX, 3-16, IOS Press.
7. Barzdins, J., Grasmanis, M., Rencis, E., Sostaks, A., & Steinsbekk, A. (2016). Towards a more effective hospital: helping health professionals to learn from their own practice by developing an easy to use clinical processes querying language. International Conference on Health and Social Care Information Systems and Technologies, Procedia Computer Science Journal, 100, 498-506. doi:10.1016/j.procs.2016.09.188

2.1.1.5 Research of ontology-based linked data and further development of their application in e-Government and e-Medicine domains (stage 4).

2.1.1.5.1 Linked Data

Linked Data is a set of principles for publishing machine-readable information on the web while enabling information interlinking, uniform data access and information integration. This is achieved using commonly accepted standards such as RDF (Cyganiak et al., 2014).

This report uses the original formulation of Linked Data principles defined by Tim Berners-Lee (Berners-Lee, 2006):

1. Use URIs as names for things (Masinter et al., 2005);
2. Use HTTP URIs so that people can look up those names (Fielding et al., 1998);
3. When someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL);
4. Include links to other URIs. so that they can discover more things.

Principle 1 recommends the use of global, standardised identifiers when referring to something. Principles 2 and 3 say that HTTP URIs need to be used so that data is published on the web and that, when asked, the server should provide useful information about the "thing" identified by a given URI, expressed as RDF data. Principle 4 recommends including (referencing) other HTTP URIs in the data so that these links can be followed and additional information can be discovered.

The use of RDF (Resource Description Framework) graph model for representing Linked Data allows data consumers to seamlessly integrate information coming for multiple sources or distributed over the web (Cyganiak et al., 2014). When Linked Data is applied to data published on the web the result is a Web of Data, based on standards and a common data model. This makes it possible to implement generic applications that operate over the complete data space (Heath and Bizer, 2011).

The information represented in RDF can be queried using the SPARQL query language (Harris, S., & Seaborne, 2013). There exist multiple tools that can help users formulate SPARQL queries without writing query language code, for example, by defining queries visually in ViziQuer (Čerāns et al., 2017) or by using controlled natural language and faceted browsing in Sparklis (Ferré, 2017).

This study examines the use of Linked Data in e-Government and e-Medicine and challenges associated with it. Since many of these challenges are independent of a particular domain this study examines them separately before going into the details of any particular domain.

By the use of Linked Data we understand the publishing and consuming of information according to Linked Data principles. As a part of the study into the use of Linked Data we also refer to other, related Semantic Web technologies and standards. However, detailed examination of Semantic Web technologies other than Linked Data is outside the scope of this study.

Use of Linked Data: Open vs. Closed

When discussing Linked Data most publications and web articles refer to the open, public use of Linked Data – publishing information on the web according to Linked Data principles. Information on open Linked Data projects and applications is readily available and, where relevant, is included in this document.

There are, however, domains where not all information may be public or even where most of the information is limited access and may have privacy and confidentiality implications. This includes both areas covered by this report – e-Government and e-Medicine.

In these domains, we should distinguish between two kinds of Linked Data:

- open Linked Data – the use of Linked Data on the public web;
- closed Linked Data – the use of Linked Data in a closed, limited access environment such as government or corporate intranet.

Linked Data principles can be applied in both cases but only open datasets will be publicly available and useable by everyone. In the closed scenario data is still published and consumed according to Linked Data principles but access to it is limited to authorised users and applications. To ensure this, organisations will need to apply access control techniques to Linked Data. These techniques are discussed further in report (Section Since these validation languages are new, there is a potential for developing tools that support them. The existing expertise of IMCS in visual notations and tools for ontology- and RDF-based systems such as ViziQuer may also be applied to RDF data validation tools (Čerāns et al., 2017). An example of such application would be visual tools for defining and visualizing RDF data shape constraints.
Linked Data Access Control).

Further implications of closed Linked Data are discussed in (Cobden et al., 2011). Its authors recognise that closed Linked Data may be necessary in some cases (where datasets may not be published openly) and warn that poor implementations of closed Linked Data and access control may break URI resolvability and affect the interlinking and reuse of such datasets.

2.1.1.5.2. The Web of Data and Linking Open Data initiative

A significant number of individuals and organisations have adopted Linked Data as a way to publish their data and to link these datasets to one another. The result is a "distributed web-scale database" commonly referred to as the Web of Data. In contains large amounts of information on all sorts of topics including information about people, creative works, healthcare (drugs, genes, clinical trials) and online communities (Heath & Bizer, 2011).

The Web of Data can be illustrated as a graph of connected datasets published as Linked Data. Its origins lie in the W3C Linking Open Data (LOD) project[1] during

[1] http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData

which a "grassroots" Linked Data community self-organized in order to identifying existing data sets available under open licenses, convert them to RDF according to the Linked Data principles, and to publish them on the Web (Heath & Bizer, 2011). As a result, initially, a small core set of Linked Data datasets appeared as can be witnessed in the LOD "cloud" graph from 10 years ago (May 2007)[2]:

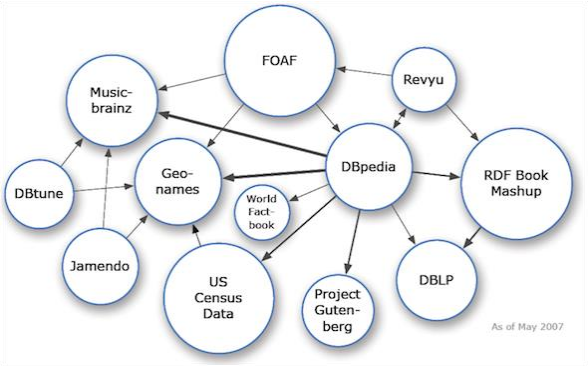


Figure 2.1.2.1: Linking Open Data cloud diagram, May 2007, by R.Cyganiak and A.Jentzsch. http://lod-cloud.net/

The LOD project is open to anyone who publishes data according to Linked Data principles and interlinks their dataset with other LOD datasets. This openness is probably a factor in the success of this project in bootstrapping the Web of Data and in its phenomenal growth.
According to a 2014 survey, in April 2014 the web of Linked Open Data consisted of 1014 datasets covering various domains. The most popular domains by the number of datasets are shown in Table 2.1.2.1. More detailed information can be found in Schmachtenberg et al., 2014.

Topic	Datasets	%
Government	183	18.0%
Publications	96	9.5%
Life sciences	83	8.2%
User-generated content	48	4.7%
Cross-domain	41	4.0%

Table 2.1.2.1: Top five LOD datasets by domain, in 2014 (Schmachtenberg et al., 2014).

The LOD "cloud" diagram, shown in Figure 2.1.2.2, depicts the largest connected LOD component as of April 2014. It consists of 570 datasets connected to one another and covering various domains (indicated by the color of nodes).

[2] http://lod-cloud.net/

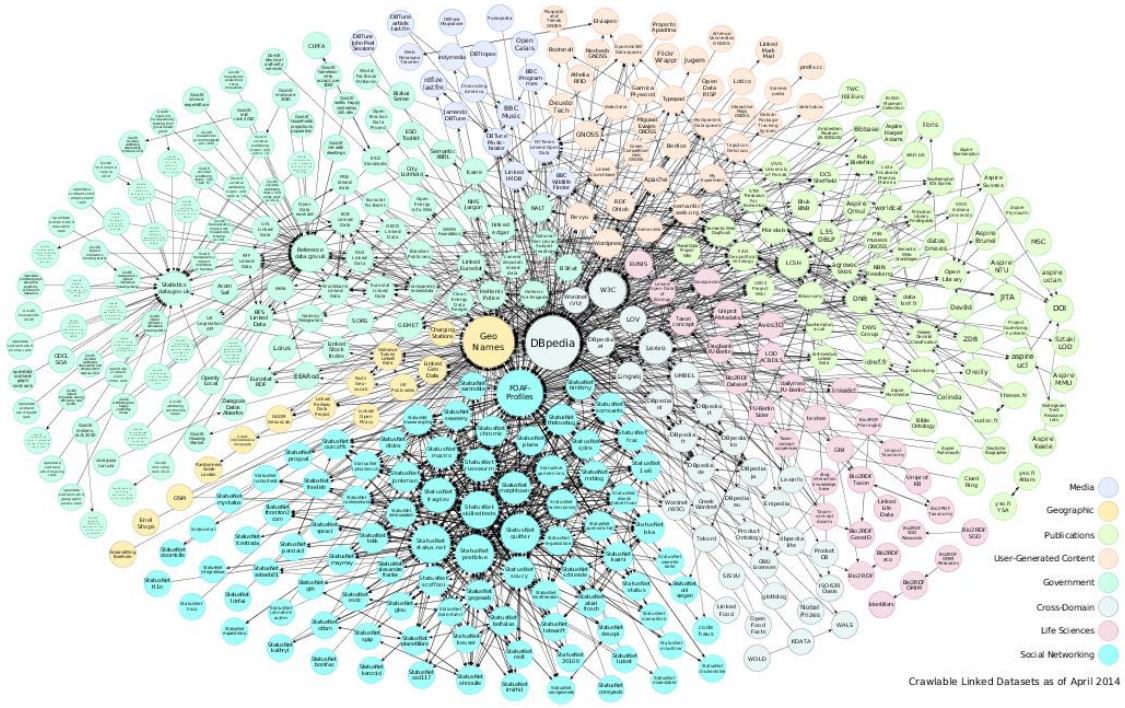


Figure 2.1.2.2: Linking Open Data cloud diagram, April 2014 (Schmachtenberg et al., 2014) [3].

At 500+ datasets, the LOD "cloud" diagram is no longer readable and, if interested, we recommend the reader to refer to the SVG version of the diagram[4] where every node is "clickable" and leads to the relevant dataset description (in datahub.io open data catalogue).

The LOD "cloud" is formed around core datasets that the project was started with. In 2014, the largest (in terms of size) and most popular (in terms of dataset interlinking) LOD datasets were:

- DBPedia[5] – a large, community-generated knowledge base that is automatically extracted from Wikipedia (Bizer et al., 2009). It is the most popular Linked Data resource and contains information about various topics (mirroring the topics covered in Wikipedia);
- GeoNames[6] - a large, open geographical database (available under a creative commons attribution license). It covers more than 10 million geographical names. Other projects often refer to GeoNames as a reference for placenames and their geographical coordinates.

Wikidata[7] is another large Semantic Web resource related to Wikipedia. It is a community-developed knowledge base of Wikipedia and central data management system used by Wikipedia and its sister projects. By 2014 it had collected data on

[3] http://lod-cloud.net/state/state_2014/
[4] http://lod-cloud.net/versions/2014-08-30/lod-cloud_colored.svg
[5] http://wiki.dbpedia.org/
[6] http://www.geonames.org/
[7] https://www.wikidata.org

more than 15 million entities and over 34 million statements about these entities (Erxleben et al., 2014). Unlike DBPedia, which is a secondary source derived from Wikipedia (thus its data quality is dependent on the quality of information on Wikipedia), Wikidata is aimed to be a central knowledge base that can be used by Wikipedia and other projects. It provides a number of data interfaces including a SPARQL query service[8].

At the time of writing, the most recent LOD "cloud" diagram was from August 2017 containing 1163 datasets. It is available in a number of formats including PNG and SVG versions and also as a JSON and TSV dataset[9].

Although the August 2017 LOD "cloud" diagram is not accompanied by domain statistics, we can observe that the most "popular" domains by the number of datasets are life sciences, government information, linguistic information and publications. This includes the two topics of this report (life sciences and government information) and illustrates that a large number of datasets from these domains are available as Linked Open Data.

2.1.1.5.3. General considerations

There is a number of Semantic Web research and development areas that apply to RDF and Linked Data in general, and are also important in the specific areas explored in this document – the use of Linked Data in e-Government and e-Medicine.

RDF Data Validation

When building RDF-based information systems their developers must take into account that the graph-based RDF model allows us to express any information about resources, their properties and inter-relations.

Developers of such systems need to be able to verify data for conformance to some criteria or constraints that the system is built upon. Therefore, a way to formally validate this data conformance is needed.

Until recently, there was no standard way to validate RDF data. The Semantic Web stack includes ontology languages RDF schema and OWL but their purpose is defining terms in RDF vocabularies and ontologies, and not validating RDF data. OWL ontology language is based on the Open World Assumption (OWA) which means that something can be true regardless of if it is stated to be true or not (Hitzler et al., 2012). This expresses the worldview that we may not have a complete knowledge of the domain and there can be assertions that may be true but do not know that.

OWA together with the non-unique-name assumption (i.e. that different URIs may refer to the same thing even if we do not know that) meant that OWL is not well-suited for RDF validation. There were initiatives to express integrity constraints in OWL by interpreting them using the Closed World Assumption but these initiatives

[8] https://www.wikidata.org/wiki/Wikidata:SPARQL_query_service
[9] http://lod-cloud.net/#history

did not replace the need for standard mechanisms for RDF data validation (Tao et al., 2010; Patel-Schneider, 2015).

The need for standardized RDF data validation approaches was addressed in 2017 with the publication of two standards:

- SHACL (Shapes Constraint Language)[10], a W3C Recommendation, is a language for validating RDF graphs against a set of conditions. These conditions are defined as data shapes and other constructs. SHACL is written in RDF (Knublauch and Kontokostas, 2017);

- ShEx 2.0 (Shape Expressions)[11] is a schema language for describing requirements for RDF graph structures (Prud'hommeaux et al., 2017). ShEx has three syntaxes – a compact, human-readable syntax (ShExC), JSON-LD syntax (ShExJ) that acts as an abstract syntax and the RDF representation of JSON-LD syntax (ShExR). ShEx standards are published by the W3C Shape Expressions Community Group.

Both validation languages can be worth exploring. While they both are meant for validating RDF graphs they differ in their principles, syntax and expressivity. A detailed description and comparison of these languages is provided in Labra Gayo et al., 2017.

Since these validation languages are new, there is a potential for developing tools that support them. The existing expertise of IMCS in visual notations and tools for ontology- and RDF-based systems such as ViziQuer may also be applied to RDF data validation tools (Čerāns et al., 2017). An example of such application would be visual tools for defining and visualizing RDF data shape constraints.

Linked Data Access Control

Access control becomes a necessity when information is published and exchanged in a closed setting and is not meant to be publicly accessible. This is also the case when using Semantic Web and Linked Data technologies in e-Government and e-Medicine.

Access control may also be necessary for open Linked Data when users are permitted write operations (adding, modifying and deleting information). Linked Data Platform (LDP) is a W3C Recommendation that defines a set of rules for implementing read-write Linked Data using HTTP operations on web resources (Speicher et al., 2015). It defines a special kind of resources called Containers which, in addition to the general HTTP mechanisms, is able to respond to requests to create new resources within them. When a resource is added to a Container, containment information is recorded and preserved as a link between the Container and the new entry added to it.

Two core functions of access control are *authentication* (ensuring that the party requesting access is who it claims it is) and *authorisation* (granting access to resources based on the access control policy).

[10] https://www.w3.org/TR/shacl/
[11] http://shex.io/shex-semantics/

While LDP specification is deliberately narrow in scope and does not directly cover access control it recognizes the need for authenticating users and controlling access to LDP resources when that is necessary. The Solid (Social Linked Data) initiative[12] that builds on LDP recommends choosing between two authentication protocols:

- WebID-TLS protocol[13] that uses cryptographic certificates stored in the client application (e.g. a web browser) to prove a user's identity;
- WebID-OIDC protocol[14] which is an authentication delegation protocol based on OAuth2/OpenID Connect and adapted to SOLID decentralised use cases.

Even though the authentication techniques adopted by the Solid project are aimed at decentralised use case they can also be applied to more centralised use cases such as government Linked Data. OAuth 2.0, in particular, is used by large social networking sites (Facebook, Twitter, etc.) for authenticating requests to their HTTP APIs (Hardt, 2012).

Linked Data is an HTTP API for RDF data and therefore authentication and authorisation techniques that work with other types of HTTP-based APIs are applicable to it as well. In e-Government and e-Medicine use cases the organisations involved are likely to have existing authentication and authorisation practices in place. We would recommend that the Linked Data applications they implement are integrated with the existing access control systems and standards adopted at these organisations.

The foundation of Linked Data – HTTP protocol – is a "clear text" protocol where information is exchanged unencrypted and thus vulnerable to eavesdropping and modification while in transit (man-in-the-middle attacks). This poses an obstacle for Linked Data access control as unauthorised users might access information that they should not have access to. In order to protect from such attacks HTTP data exchange should be encrypted. The standard method for encrypting HTTP traffic is the HTTPS (HTTP over Transport Layer Security) protocol. It protects HTTP connections from eavesdropping and tampering, and may also be used to authenticate participants of these connections using public key encryption (Rescorla, 2000).

Basic access control may be implemented using computer networking techniques by allowing access to Linked Data HTTP servers only from those networks and IP addresses that are authorized to access the information on these servers. In this case access control is using IP addresses and does not uniquely identify the user or application requesting access.

There are multiple methods for *authorisation* - granting access to resources once users have been authenticated. A widely used method is access control lists (ACLs) that define which users or groups of users may access what resources. ACLs are used by the Web Access Control (WAC) specification[15] which provides a simple vocabulary

[12] https://github.com/solid/solid-spec
[13] https://www.w3.org/2005/Incubator/webid/spec/tls/
[14] https://github.com/solid/webid-oidc-spec
[15] https://www.w3.org/wiki/WebAccessControl

for defining access control lists for web resources. It is further elaborated in the WAC specification adopted by the Solid project which takes into account the requirements of LDP systems such as inheriting ACLs in LDP containers[16].

There are situations when ACLs are not sufficient and more expressive access control methods are required. One such method for controlling access to Linked Data resources is proposed by Costabello et al., 2013. It introduces two RDF vocabularies that are used for describing access control policies and user attributes, and adapts the Shi3ld attribute-based authorization framework (originally developed for SPARQL endpoints) for the Linked Data Platform.

Further information about relevant access control methods and requirements can be found in the RDF access control survey published in 2017 (Kirrane et al., 2017). It contains a thorough review of the existing models and standards, and defines a set of access control requirements for Linked Data.

2.1.1.5.4 Applications of Linked Data to Target Research Areas

This section describes how Linked Data is used in e-Government and e-Medicine.

When addressing this question, we first look at the use of Linked Data principles in these respective areas. This includes publishing Linked Data, consuming and integrating existing Linked Data sources and other applications of Linked Data principles.

We looked for both open and closed uses of Linked Data in these domains. It should be noted that after careful examination of published research we found only limited evidence of closed use of Linked Data. Typically, Linked Data principles in these and other domains are applied as open Linked Data that the involved parties either publish, consume or do both.

Linked Data in e-Government

Linked Data principles can be applied in a domain (including e-Government and e-Medicine) in a number of ways:

a) publishing Linked Data;
b) consuming Linked Data;
c) developing information systems built on Linked Data.

The majority of cases of using Linked Data in the government information domain that we found are in publishing open data. This is demonstrated by the number of government datasets in the LOD "cloud" and in the Datahub.io data catalogue that this graph is based upon[17] (221 government LOD datasets as of November 2017).

By publishing information as open Linked Data, public sector (government, municipal, etc.) organisations not just make this information available for reuse but

[16] https://github.com/solid/web-access-control-spec – version v.0.4.0, retrieved 2017-11-02.
[17] https://old.datahub.io/dataset?tags=lod&tags=government&_tags_limit=0

also make it possible to interlink datasets (by using relevant URI identifiers) and for other parties to use these datasets as a reference point. While it is not the task of this report to present detailed analysis of all available government LOD datasets, this section will present information about interesting and useful applications of Linked Data in the e-Government domain.

Government and other public-sector datasets are typically listed in open data catalogues. There are various such catalogues maintained both by governments (e.g. data.gov.uk) and open data communities (e.g. Datahub.io). These catalogues are typically hosted using the CKAN open source software[18], although there are also other systems available.
In order to ensure data catalogue interoperability, W3C has published the Data Catalog (DCAT) RDF vocabulary[19] (Maali et al., 2014). It allows systems to exchange information about data catalogues, datasets and their distributions. The EU has published an adapted version of DCAT – the DCAT-AP Application profile for data portals in Europe v1.1 [20] aimed at describing information about data catalogues deployed in the EU (DCAT, 2017). This profile further refines the DCAT vocabulary by indicating how its terms should be used when describing EU data portals and what controlled vocabularies to use as term values. The DCAT-AP initiative has adopted the SHACL constraint language and has published SHACL validation rules that DCAP-AP 1.1 documents must comply to[21].

The OpenDataMonitor project[22] has collected a comprehensive list of open data platforms (such as CKAN), open data catalogues and reports[23]. It covers 11 different platforms, 217 open data portals and 503 open data reports and resources (Open Data Institute, 2015).

Many open data portals publish their metadata (about the portal and its datasets) as Linked Open Data. In most cases it is done using W3C DCAT vocabulary and, in the case of EU portals, the DCAT-AP 1.1 application profile. Support for DCAT-AP 1.1 metadata output can be added to CKAN using the ckanext-dcat extension[24].

Once this extension is enabled, dataset information is available as RDF data (in a number of RDF syntaxes) by adding the relevant RDF syntax extension (.xml, .ttl, .n3 or .jsonld) to the dataset URI. This example shows how to retrieve dataset metadata from Latvia's national open data portal (data.gov.lv):

> Uzņēmumu reģistrs (Enterprise register data)
>
> Dataset URI: https://data.gov.lv/dati/lv/dataset/uz
> Turtle RDF metadata: https://data.gov.lv/dati/lv/dataset/uz.ttl

[18] https://ckan.org/
[19] https://www.w3.org/TR/vocab-dcat/
[20] https://joinup.ec.europa.eu/release/dcat-ap-v11
[21] https://github.com/SEMICeu/dcat-ap_shacl
[22] https://project.opendatamonitor.eu/
[23] https://docs.google.com/spreadsheets/d/12aptaQdVrd37QH40nbT2YpTZqd2fWe-V8NQXRsU0Leo/edit#gid=0
[24] https://github.com/ckan/ckanext-dcat

Some data portals also provide SPARQL endpoints that allow users to make advanced queries for open datasets. The EU currently maintains two open data portals (with different goals each):

- Open Data Europe Portal (ODP) is the data portal of the European Union containing datasets that are collected and published by the European Institutions[25];
- European Data Portal (EDP) is a European portal that harvests metadata from public sector portals throughout Europe. EDP therefore focuses on data made available by European countries. In addition, EDP also harvests metadata from ODP[26].

Both these portals publish metadata as Linked Data and provide advanced SPARQL query functionality along with query examples. For example, this SPARQL query allows users to find information about ODP portal datasets published after a given date:

```
SELECT  ?DatasetTitle  ?DatasetPublisher
WHERE { graph ?g  {
  ?DatasetURI a <http://www.w3.org/ns/dcat#Dataset>;
  dc:publisher ?DatasetPublisher;
  dc:title ?DatasetTitle ;
  dc:modified ?DateModified
  FILTER(xsd:dateTime(?DateModified)
    > "2016-01-01"^^xsd:dateTime)
  }
}
```

Government institutions often publish tabular data (CSV, Excel files, etc.). These files are just tables and do not contain additional information about their content, semantics and relations to other data. W3C CSV on the Web (CSVW) working group has published Semantic Web standards for addressing this issue and describing detailed information about tabular datasets[27]:

- Model for Tabular Data and Metadata on the Web[28] defines the core model for describing information about tabular data and related metadata (Tennison & Kellogg, 2015b);
- Metadata Vocabulary for Tabular Data[29] is an vocabulary of terms for describing tabular datasets (Tennison & Kellogg, 2015a). It defines the format and structure of metadata documents, expressed in JSON-LD syntax.

By describing machine-readable information about datasets, data publishers can help users understand, interpret and process these datasets. The CSVW standard also makes it possible to create mappings from tabular data to RDF classes, properties and

[25] https://data.europa.eu/euodp/en/linked-data
[26] https://www.europeandataportal.eu/sparql-manager/en/
[27] When referring to all these documents together we will use the term "CSVW standard".
[28] http://www.w3.org/TR/tabular-data-model/
[29] http://www.w3.org/TR/tabular-metadata/

resource URIs, and defines a standard procedure for transforming tabular data to RDF (Tandy et al., 2015).

The standards described above (DCAT and CSVW) are used by the Open Data Portal Watch framework[30] – a scalable open data monitoring and quality assessment framework that harvests and analyses metadata from about 260 open data portals. Its authors have observed that data portals vary by the portal platform used (CKAN, Socrata, etc.) and by the amount and the quality of metadata published. In order to provide a unified, homogenized view of these dataset descriptions, the Portal Watch framework collects metadata in portals' original formats and maps these descriptions to DCAT. It also enriches CSV dataset descriptions by analyzing file contents (e.g. detecting datatypes) and converting this information into CSVW metadata (Neumaier et al., 2017). The resulting dataset providing a unified view on these open data portals and their datasets (approx. 120 million RDF triples per weekly snapshot) is available via the Portal Watch SPARQL endpoint[31].

The use of Linked Data for integrating government datasets is demonstrated by Shi et al., 2017, where authors describe the Norwegian State of Estate (SoE) dataset containing information about real estate owned by the central government in Norway. This dataset is produced by integrating government datasets from different sources such as the business entity register, cadastral system and the previous SoE report. The dataset is made available as Linked Data. Detailed information about the Linked Data generation process, including data cleaning, conversion, augmentation and interlinking with other LOD datasets, is presented in the paper (Shi et al., 2017).

After examining publications on Linked Data use in e-Government, including the research mentioned above, we could not find evidence of the closed e-Government use of Linked Data. The projects that we are aware of publish and/or consume Linked Data as a part of public Linked Open Data initiatives.

This observation may partly be explained by strong incentives for government organizations to publish open data and by the fact that open data initiatives often recommend Linked Data as a good practice for publishing open data. Also, we can expect information about open data projects (due to their public nature) to be more widely disseminated compared to closed, non-public projects.

In a closed and controlled setting, such as e-Government data exchange, there are many alternative technologies that can be used for linking information systems. With a limited number of systems to connect and with centralized decision-making that can require all parties to use the same technologies, there may be less incentives to use open, web-oriented technologies such as Linked Data. At the same time, modern information systems often use REST-like web services for exchanging information. In case if these services truly follow REST principles (Fielding, 2000), they are already very close to following Linked Data principles.

This section surveyed the use of Linked Data in e-Government. Currently, the main use of Linked Data in this domain is data publishing and integration. Public sector

[30] http://data.wu.ac.at/portalwatch/
[31] http://data.wu.ac.at/portalwatch/sparql

organisations have many datasets that can be published as structured and linked data. Particular attention should be paid to publishing Linked Data datasets that are often referenced from other government datasets. While publishing open data is a worthwhile effort, it is also important to make the next step and to develop applications that use this Linked Data.

Linked Data in e-Medicine

Semantic Web technologies, including RDF, Linked Data and ontologies, are widely used in healthcare and life sciences. This is demonstrated both by the large number of studies about this topic, some of which are mentioned in this section, and by the by the number of datasets in the Datahub.io data catalogue (245 biomedicine LOD datasets as of November 2017) that the LOD "cloud" graph is based upon[32].

Ontologies and RDF vocabularies are important components of semantic information systems that help ensure information interoperability. While it is possible to publish Linked Data using a custom vocabulary (e.g. derived from a database schema), the data is more valuable when it can be interpreted and reused by other applications and users. Ontologies and RDF vocabularies help in this regard by providing a common set of terms (classes, properties, ...) for describing knowledge related to a given domain.

Ontologies play a very important role in the biomedical domain. Many ontologies used in this domain are extremely large, with tens and hundreds of thousands of classes. BioPortal[33] is the largest repository of biomedical ontologies that by November 2017 contained more than 650 ontologies. It contains the ontologies themselves, metadata about these ontologies and the mappings between terms in different ontologies. For access to this information, BioPortal provides a SPARQL endpoint[34] and de-referenceable Linked Data URIs for whole ontologies and for individual terms in these ontologies (Salvadores et al., 2013).

BioPortal supports access control (ontology submitters to may set access to "private" and limit access to an ontology to a list of users), therefore access control is also implemented in its SPARQL endpoint. Users are provided with a user API key that, in order to get access to private ontologies, they must include in HTTP headers of the SPARQL request (Salvadores et al., 2013).

The W3C Health Care and Life Sciences Interest Group (HCLS IG) that was active from 2005 to 2016 was aimed at developing, advocating for, and supporting the use of Semantic Web technologies in health care, life sciences, clinical research and translational medicine[35]. The scope of HCLS IG included, among other things, creating health care and life sciences Linked Data and guidelines to help others create Linked Data, as well as developing RDF vocabularies and creating mappings between HCLS vocabularies.

---

[32] https://old.datahub.io/dataset?tags=lod&_tags_limit=0&tags=biomedicine
[33] http://bioportal.bioontology.org/
[34] http://sparql.bioontology.org/
[35] https://www.w3.org/2011/09/HCLSIGCharter

Recent outcomes of the HCLS IG[36] include the addition of RDF to the HF7 FHIR standard, the development of RDF Shape Expressions (described earlier in this report) and the use of HCLS best practices in the Open PHACTS standards initiative[37].

FHIR (Fast Healthcare Interoperability Resources) Specification[38] is a standard framework developed by the clinical standards organization HL7 for exchanging healthcare information.
This framework adopts a resource-centric approach, with all exchangeable content being represented as Resources, and defines data formats and APIs for exchanging healthcare information.

FHIR Release 3.0, published in 2017, adds support for RDF and Linked Data[39]. This makes it possible to use FHIR data with RDF applications, to support inference, shared semantics across multiple standards, data integration, SPARQL queries and other uses. In order to support data validation, FHIR uses ShEx for defining a grammar to validate FHIR/RDF data[40] (Solbrig et al., 2017).

The best practices for mapping and interlinking HCLS data using RDF, developed inside this W3C Interest Group, are described by Marshall et al., 2012. This paper presents a workflow for mapping HCLS sources to RDF and linking them to other Linked Data sources, and includes four case studies that demonstrate the workflow. The best practices for creating and publishing HCLS Linked Data are further elaborated in the W3C HCLS Linked Data Guide[41] (Marshall & Boyce, 2012).

The HCLS IG community has also developed important HCLS Linked Data resources such as Bio2RDF[42] (Callahan et al., 2013) and Linked Open Drug Data (Samwald et al., 2011). Other important HCLS Linked Data resources include Linked Life Data[43] and the EBI RDF Plaform[44]. The EBI RDF Platform integrates RDF resources and services available at the European Bioinformatics Institute (EBI), and allows its users to run SPARQL queries over all these EBI RDF resources together (Jupp et al., 2014). The platform uses Linked Data (HTTP) URIs for all datasets, either making use of the existing Linked Data URIs (where available) or minting new URIs where necessary.

OpenPHACTS[45] (the Open Pharmacological Concepts Triple Store) is a EU public-private partnership project that involves academia, pharmaceutical companies and other businesses. The Open PHACTS Discovery Platform integrates pharmacological data from a variety of sources (Uniprot, Drugbank, WikiPathways, etc.) and provides services for querying this integrated data. By providing open access to integrated pharmacological data, the project aims to reduce barriers to drug discovery (Gray et

[36] More information about the outcomes of HCLS IG: https://www.w3.org/wiki/HCLSIG/Products
[37] https://www.openphacts.org/
[38] http://hl7.org/fhir/
[39] https://www.hl7.org/fhir/linked-data-module.html
[40] https://www.hl7.org/fhir/fhir.shex
[41] https://www.w3.org/2001/sw/hcls/notes/hcls-rdf-guide/
[42] http://bio2rdf.org/
[43] http://linkedlifedata.com/
[44] https://www.ebi.ac.uk/rdf/
[45] http://www.openphacts.org

al., 2012). Further information about the OpenPHACTS Linked Data architecture can be found in Groth et al., 2014.

There are numerous projects and studies that make use of HCLS Linked Data, for example, by using this information for federated querying of life sciences Linked Data (Hasnain et al., 2017) or assessing drug target associations (Chen et al., 2012).

The use of Linked Data in e-Medicine also include cases where some data sources that are closed (i.e. not public) and require access control. For example, clinical research may require the integration of medical data from multiple sources (e.g. hospitals and research institutes) where medical data may contain sensitive or proprietary information. The Linked Medical Data Access Control (LiMDAC) framework proposes to address the issue of controlling access to medical data with diverse access constraints by using Linked Data technologies to (Katameri et al., 2014).

The issues related to combining open and restricted-access data sources were examined in a pilot study incorporating commercial and private datasets into the OpenPHACTS platform. In this study, Linked Data access control was ensured by defining access rights on the level of named RDF graphs and graph groups (Goble et al., 2013). Another issue related to including restricted-access sources in a collection of Linked Data resources is to determine if the metadata describing these datasets should be private or public. The conclusion of this pilot study was to make all metadata public whenever possible (Goble et al., 2013).

This section demonstrated that Linked Data and other Semantic Web technologies are widely used in the domain of e-Medicine and bioinformatics. Linked Data sources and principles are applied in this domain in a number of different ways such as data publishing, data mapping and integration, and making further use this information (e.g. for drug discovery).

References

Berners-Lee, T., 2006. Linked Data - Design Issues. W3C Design Issues.
URL http://www.w3.org/DesignIssues/LinkedData.html

Callahan, A., Cruz-Toledo, J., Ansell, P., & Dumontier, M. (2013). Bio2RDF Release 2: Improved Coverage, Interoperability and Provenance of Life Science Linked Data. In The Semantic Web – ISWC 2016 (Vol. 7882, pp. 200–212). Berlin, Heidelberg: Springer Berlin Heidelberg.
http://doi.org/10.1007/978-3-642-38288-8_14

Chen B, Ding Y, Wild DJ (2012) Assessing Drug Target Association Using Semantic Linked Data. PLOS Computational Biology 8(7): e1002574. https://doi.org/10.1371/journal.pcbi.1002574

Christian Bizer, Jens Lehmann, Georgi Kobilarov, Soren Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. Dbpedia - a crystallization point for the web of data. Journal of Web Semantics: Science, Services and Agents on the World Wide Web, 7(3):154–165, 2009.
http://dx.doi.org/10.1016/j.websem.2009.07.002

Cobden, M., Black, J., Gibbins, N., Carr, L., & Shadbolt, N. (2011). A Research Agenda for Linked Closed Data. Proceedings of the Second International Workshop on Consuming Linked Data (COLD2011). Bonn, Germany, October 23, 2011.

Costabello L., Villata S., Rodriguez Rocha O., Gandon F. (2013) Access Control for HTTP Operations

on Linked Data. In: Cimiano P., Corcho O., Presutti V., Hollink L., Rudolph S. (eds) The Semantic Web: Semantics and Big Data. ESWC 2013. Lecture Notes in Computer Science, vol 7882. Springer, Berlin, Heidelberg.

Cyganiak, R., Wood, D., & Lanthaler, M. (2014). RDF 1.1 Concepts and Abstract Syntax. 25 February 2014. W3C Recommendation. URL: http://www.w3.org/TR/rdf11-concepts/

Čerāns, K., Bārzdiņš, J., Šostaks, A., Ovčiņņikova, J., Lāce, L., Grasmanis, M., & Sproģis, A. (2017). Extended UML Class Diagram Constructs for Visual SPARQL Queries in ViziQuer/web. Proceedings of the Third International Workshop on Visualization and Interaction for Ontologies and Linked Data (VOILA '17), Vienna, Austria. CEUR Workshop Proceedings, vol. 1947.

DCAT Application Profile for data portals in Europe Version 1.1. (2017). EU ISA^2 programme publication, 2017-02-24. URL https://joinup.ec.europa.eu/release/dcat-ap-v11

Erxleben F., Günther M., Krötzsch M., Mendez J., Vrandečić D. (2014) Introducing Wikidata to the Linked Data Web. In: Mika P. et al. (eds) The Semantic Web – ISWC 2014. ISWC 2014. Lecture Notes in Computer Science, vol 8796. Springer, Cham.

Ferré, S., 2017. Sparklis - An expressive query builder for SPARQL endpoints with guidance in natural language. Semantic Web 8(3): 405-418.

Fielding, R., Gettys, J., Mogul, J. C., Frystyk, H., Masinter, L., Leach, P. & Berners-Lee, T. (1998). Hypertext Transfer Protocol -- HTTP/1.1. Internet Standard, RFC 2616.
URL https://tools.ietf.org/html/rfc2616

Fielding, R. T. (2000). REST: architectural styles and the design of network-based software architectures. Doctoral dissertation, University of California.

Gray A., Groth P., Loizou A., et al. (2012) Applying Linked Data Approaches to Pharmacology: Architectural Decisions and Implementation. Semantic Web Journal, April 18, 2012

Groth, P. T., Loizou, A., Gray, A. J. G., Goble, C. A., Harland, L., & Pettifer, S. (2014). API-centric Linked Data integration - The Open PHACTS Discovery Platform case study. J. Web Sem., 29, 12–18. http://doi.org/10.1016/j.websem.2014.03.003

Hardt, D. (2012). The OAuth 2.0 Authorization Framework. Internet Standard, RFC 6749.
URL https://tools.ietf.org/html/rfc6749

Harris, S., & Seaborne, A. (2013). SPARQL 1.1 Query Language. W3C Recommendation, 21 March 2013. URL http://www.w3.org/TR/sparql11-query.

Hasnain, A., Mehmood, Q., Sana e Zainab, S., Saleem, M., Warren, C., Zehra, D., et al. (2017). BioFed: federated query processing over life sciences linked open data. Journal of Biomedical Semantics, 8(1), 599. http://doi.org/10.1186/s13326-017-0118-0

Heath, T., & Bizer, C. (2011). Linked Data: Evolving the Web into a Global Data Space (1st edition). Synthesis Lectures on the Semantic Web: Theory and Technology, 1:1, 1-136. Morgan & Claypool.

Hitzler, P., Krötzsch, M., Parsia, B., Patel-Schneider, P. F.; Rudolph, S. (2012). OWL 2 Web Ontology Language: Primer (Second Edition). W3C Recommendation, 11 December 2012. URL http://www.w3.org/TR/owl2-primer/

Jupp, S., Malone, J., Bolleman, J., Brandizi, M., Davies, M., Garcia, L., et al. (2014). The EBI RDF platform: linked open data for the life sciences. Bioinformatics, 30(9), 1338–1339. http://doi.org/10.1093/bioinformatics/btt765

Kamateri, E., Kalampokis, E., Tambouris, E., & Tarabanis, K. (2014). The linked medical data access control framework. Journal of biomedical informatics, 50, 213-225.

Kirrane, S., Mileo, A. & Decker, S. (2017). Access control and the Resource Description Framework - A survey. Semantic Web, 8(2), pp.311–352.

Knublauch, H., & Kontokostas, D. (2017). Shapes Constraint Language (SHACL). W3C Recommendation, 20 July 2017. URL https://www.w3.org/TR/shacl/

Labra Gayo, J. E., Prud'hommeaux, E., Boneva, I., & Kontokostas, D. (2017). Validating RDF Data. Synthesis Lectures on Semantic Web: Theory and Technology, 7(1), pp. 1-328.

Maali, F., Erickson, J., & Archer, P. (2014). Data Catalog Vocabulary (DCAT). W3C Recommendation, 16 January 2014. URL https://www.w3.org/TR/vocab-dcat/

Marshall, M.S., & Boyce, R.D., (Eds). (2012). Health Care and Life Science (HCLS) Linked Data Guide. World Wide Web Consortium (W3C). November 2012. http://www.w3.org/2001/sw/hcls/notes/hcls-rdf-guide/.

Marshall, M. S., Boyce, R., Deus, H. F., Zhao, J., Willighagen, E. L., Samwald, M., et al. (2012). Emerging practices for mapping and linking life sciences data using RDF — A case series. Web Semantics: Science, Services and Agents on the World Wide Web, 14, 2–13. http://doi.org/10.1016/j.websem.2012.02.003

Masinter, L., Berners-Lee, T., & Fielding, R. T. (2005). Uniform resource identifier (URI): Generic syntax. Internet Standard, RFC 3986. URL https://tools.ietf.org/html/rfc3986

Neumaier, S., Umbrich, J., & Polleres, A. (2017). Lifting Data Portals to the Web of Data. WWW2017 Workshop on Linked Data on the Web (LDOW2017), Perth, Australia, April 3-7, 2017. URL http://ceur-ws.org/Vol-1809/article-03.pdf

Open Data Institute (2015). Open data platforms, catalogues and reports: resources from OpenDataMonitor. OpenDataMonitor, EU grant number 611988. Available at http://project.opendatamonitor.eu/project/ and www.opendatamonitor.eu

Patel-Schneider, P. F. (2015). Using description logics for RDF constraint checking and closed- world recognition. In Proc. of the 29th Conference on Artificial Intelligence, AAAI'15, pp. 247–253. AAAI Press, 2015. http://dl.acm.org/citation.cfm?id=2887007.

Prud'hommeaux, E., Boneva, I., Labra Gayo, J. E., & Kellog, G. (2017). Shape Expressions Language 2.0. W3C Draft Community Group Report, 13 July 2017. URL http://shex.io/shex-semantics/

Rescorla, E. (2000). HTTP Over TLS. Internet Standard, RFC 2818. URL https://tools.ietf.org/html/rfc2818

Salvadores, M., Alexander, P. R., Musen, M. A., & Noy, N. F. (2013). BioPortal as a Dataset of Linked Biomedical Ontologies and Terminologies in RDF. Semantic Web, 4(3), 277–284.

Samwald, M., Jentzsch, A., Bouton, C., Kallesøe, C., Willighagen, E., Hajagos, J., et al. (2011). Linked open drug data for pharmaceutical research and development. Journal of Cheminformatics, 3(1), 19. http://doi.org/10.1186/1758-2946-3-19

Schmachtenberg, M., Bizer, C., & Paulheim, H. (2014). State of the LOD Cloud 2014. University of Mannheim, Data and Web Science Group. August, 30. URL http://lod-cloud.net/state/state_2014/

Shi, L., Sukhobok, D., Nikolov, N., & Roman, D. (2017). Norwegian State of Estate Report as Linked Open Data. In: Panetto H. et al. (eds) On the Move to Meaningful Internet Systems. OTM 2017. Lecture Notes in Computer Science, vol 10574. Springer, Cham. https://doi.org/10.1007/978-3-319-69459-7_30

Solbrig, H. R., Prud'hommeaux, E., Grieve, G., McKenzie, L., Mandel, J. C., Sharma, D. K., & Jiang, G. (2017). Modeling and validating HL7 FHIR profiles using semantic web Shape Expressions (ShEx). Journal of Biomedical Informatics, 67, 90–100. http://doi.org/10.1016/j.jbi.2017.02.009

Speicher, S., Arwe, J., Malhotra, A. (2015). Linked Data Platform 1.0. W3C Recommendation, 26 February 2015. URL http://www.w3.org/TR/ldp/

Tandy, J., Herman, I., & Kellogg, G. (2015). Generating RDF from Tabular Data on the Web. W3C Recommendation, 17 December 2015. URL: http://www.w3.org/TR/csv2rdf/

Tao, J., Sirin, E., Bao, J. & McGuinness, D. L. (2010). Integrity constraints in OWL. In Proc. of the 24th Conference on Artificial Intelligence (AAAI'10), 2010.

Tennison, J., & Kellogg, G. (2015a). Metadata Vocabulary for Tabular Data. W3C Recommendation, 17 December 2015. URL: http://www.w3.org/TR/tabular-metadata/

Tennison, J., & Kellogg, G. (2015b). Model for Tabular Data and Metadata on the Web. 17 December 2015. W3C Recommendation. URL: http://www.w3.org/TR/tabular-data-model/

2.1.1.6 Fast query language usage testing on Children's Clinical University Hospital data (stage 4).

Usability testing of Fast query language, which is developed in the previous project stages, has been performed on CCUH data for years 2015 and 2016. Practical application of the system has proved that domain experts (physicians) after approximately 2 hours long training were able to write independently (without assistance of programmer) queries of such complexity which were actual for the analyses of hospital processes of last two years. Usability testing also showed that the developed system provides very high speed of performance – in average less than 0.5 seconds for traditional queries. In general, usability testing has proved that the developed system can be practically used for the analysis of hospital's processes.

Besides that, a publication is accepted for the publication in journal "Software and System Modeling (SoSyM)" (Scopus, SNIP>1)

2.1.1.7 Further development of web-based methods for modeling of hard-to-formalize systems (stage 4).

The UL IMCS team already has a significant experience in creating a support for the building of sufficiently rich domain specific graphical modeling languages using the local tool building platform TDA developed by the team. The experience has shown that such domain specific languages are appropriate for modeling of hard-to-formalize systems. Now a web-based tool building platform version has been developed. This platform enables the online development of both a graphical modeling language and its editor using the Configurator tool, and supports on-the-fly testing of the language on examples and modifying it if required. The platform runs on a server, but each user working in a web browser can participate in the development of the language and its editor, and use the already developed language for modeling. All the modifications are synchronized between the concurrent users within a team. Since the user accesses the platform only via a standard browser, various hardware can be used – desktops, laptops, tablets and even smartphones.

The functionality of the web-based platform is similar to that of the existing local TDA. The architecture (Fig. 2.1.4.1) of the new platform is also similar to that of the TDA.

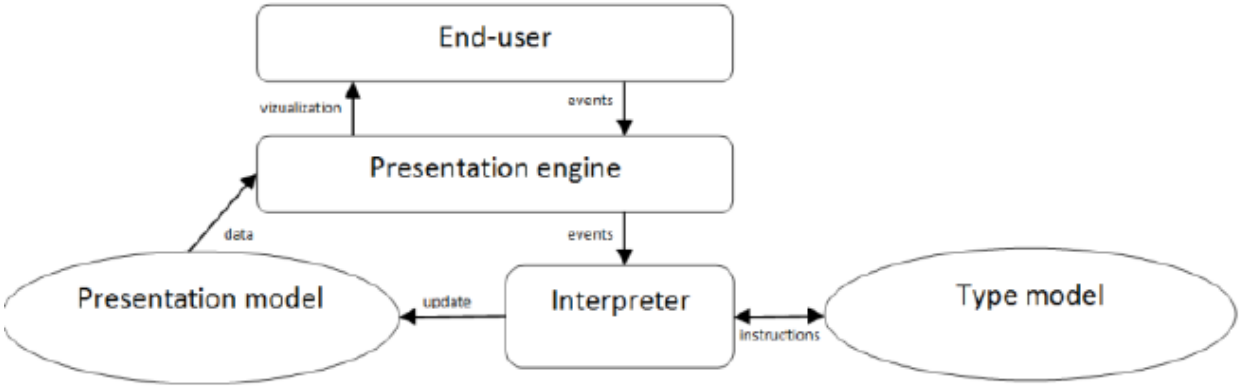


Figure 2.1.4.1. Architecture of the platform.

The presentation engine renders the current diagram according to its Presentation model and accepts events generated by the user actions in the diagram. The interpreter processes these events and updates the Presentation model according to the instructions stored in the Type model for the given tool definition. The Type model in turn is an instance of the Type metamodel (Fig. 2.1.4.2). The Type model for a graphical language and its tool is built using the Configurator tool in the platform. It should be noted that the Configurator itself is a fixed tool based on a specific graphical language and therefore could be built using the initial version of the platform (the bootstrapping principle). The principles of the platform have been published in two papers:

1. A.Sprogis, DSML Tool Building Platform in WEB. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Databases and Information Systems, 12th International Baltic Conference, DB&IS 2016, Riga, Latvia, July 4-6,

2016, Proceedings, Communications in Computer and Information Science Vol. 615, Springer, pp.99-109, 2016. (SCOPUS)

2. A.Sprogis, ajoo: WEB Based Framework for Domain Specific Modeling Tools. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Frontiers of AI and Applications, Vol. 291, Databases and Information Systems IX, IOS Press, pp. 115-126, 2016. (WebOfScience),

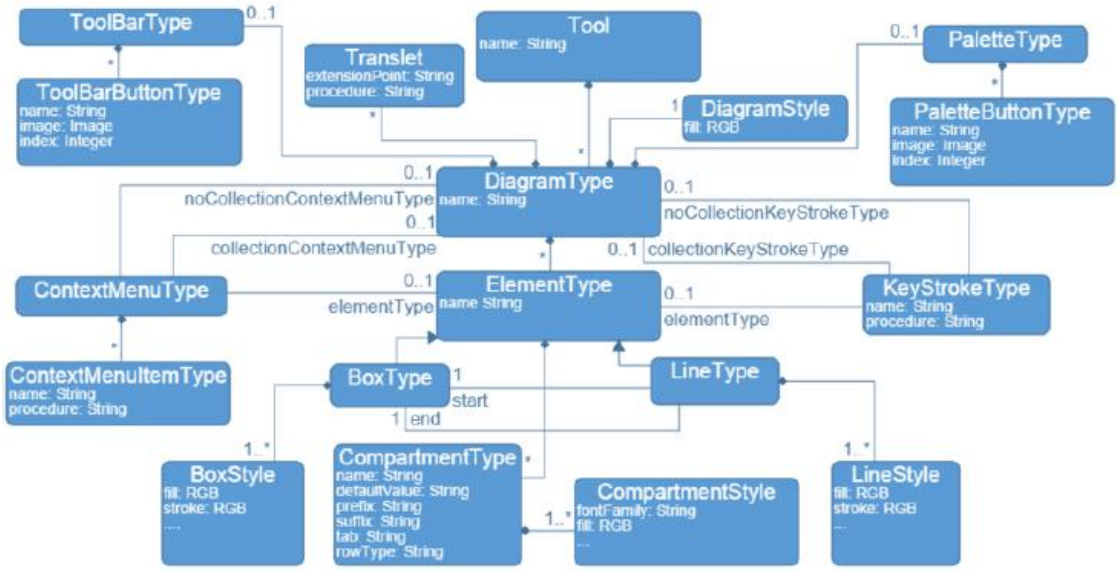


Figure 2.1.4.2. Type metamodel

Currently an executable platform prototype has been developed. It is built using the JavaScript language-related technologies: Meteor (web application development framework), MongoDB (database), KonvaJS (diagram rendering), and Bootstrap (HTML 5, CSS). The Meteor framework is of special value here since it is directly oriented towards building such kind of server-based systems and e.g. automatically supports user data synchronization.

A platform usage methodology containing several examples has been developed as well. Figures 2.1.4.3 and 2.1.4.4 show one such example – a simplified class diagram definition (a Type model) in the Configurator and a class diagram example built using the defined tool. The element texts in the Configurator are defined using textual dialogs (not shown here).

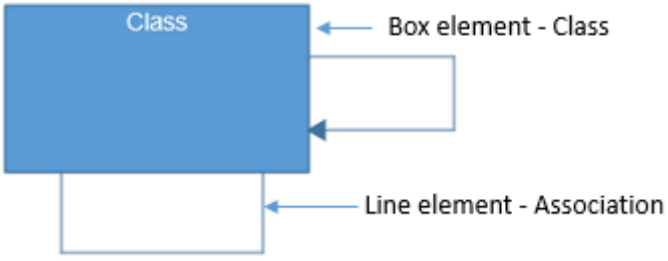


Figure 2.1.4.3. Simplified class diagram definition (a Type model) in the Configurator

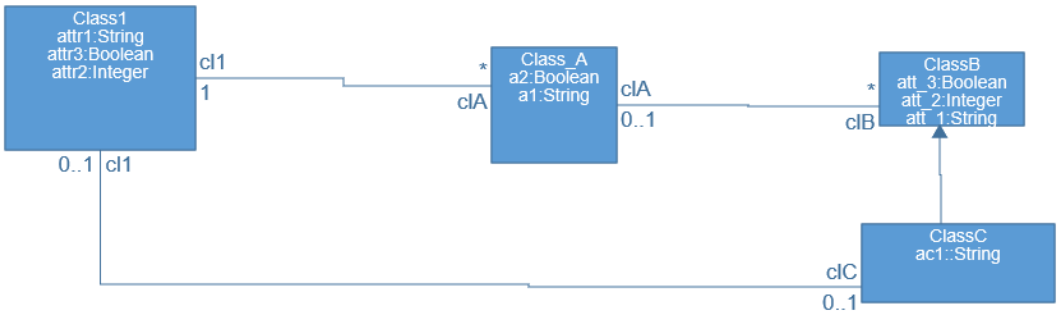


Figure 3. Class diagram example

The next step in the platform development is to migrate from the currently used metamodel instantiation to metamodel specialization. For simple diagram kinds (such as the simplified class diagram) the creation of the required Type model is quite straightforward. However, for more complicated diagrams the definition requires certain knowledge of internal system details, since various procedural extensions frequently are required for the interpreter.

On the contrary, the new proposed specialization approach does not require such a knowledge. There only subclasses of the relevant Universal metamodel need to be created basing on standard class diagram features. For some cases declarative OCL constraints also need to be added to the defined subclasses. This specialized metamodel completely defines the given language and its tool. Thus the specialization approach would ease the definition of a language and its editor and make this activity accessible to experts in various problem domains for whom such domain specific modeling languages are really required. At the same time all the possibilities of the existing platform are retained. Currently the basic principles of the specialization approach have been developed, and its implementation principles in the platform have been chosen.

The principles of the metamodel specialization method have been described in the following publications:

1. A.Kalnins, J.Barzdins, Metamodel Specialization for DSL Tool Building. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Databases and Information Systems, 12th International Baltic Conference, DB&IS 2016, Riga, Latvia, July 4-6, 2016, Proceedings, Communications in Computer and Information Science Vol. 615, Springer, pp.68-82, 2016. (**SCOPUS**)

2. A.Kalnins, J.Barzdins, Metamodel specialization for graphical modeling language support. // In: Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems (**MODELS 2016**). ACM, pp.103-112, 2016. (**SCOPUS**)

3. A. Kalnins, J. Barzdins. Metamodel Specialization for Diagram Editor Building, Databases and Information Systems IX, Selected Papers from DB&IS 2016, Frontiers in Artificial Intelligence and Applications, Vol. 291, IOS Press, pages 87-100, 2016 (**WebOfScience**)

4. J. Barzdins and A. Kalnins, Metamodel Specialization for Graphical Language and Editor Definition, Baltic J. Modern Computing, Vol. 4 (2016), No. 4, pp. 910-933 (**WebOfScience**)

5. A. Kalnins, J. Barzdins. Metamodel Specialization for Graphical Language Support. Revised version accepted for **SoSyM** journal (**Springer**), for the special issue of the SoSyM journal containing the best papers from MODELS 2016

2.1.2 Developing methods of semantic web and computational linguistics for understanding data collected in a natural language (stages 1-3).

In the year 2014 (stage 1) by integrating FrameNet n-ary relation extraction and BabelNet inspired Named Entity Linking approaches we have developed a unified linguistic ontology suitable for extracting Curriculum Vitae like semantic information (a semantic graph) about persons and organizations mentioned in unstructured newswire texts. We have also developed a new classification algorithm nicknamed C6.0 and used for implementing a semantic parser for Latvian, English, Czech and Chinese, with which we participated in SemEval-2015 competition where it performed on par with other state-of-the-art parsers and was among the three winning parsers in various testing categories. These scientific results are described in our SemEval-2015 paper [1].

After a successful participation in SemEval-2015 competition in the year 2015 (stage 2), we were able to integrate these semantic graph parsing technologies and also the approaches used by other competitors in the Latvian language semantic analysis toolchain developed in the 1-st period of this project. Applying this research enabled a significant improvement in the accuracy of semantic frame extraction – an improvement of F1-score from 57.6% to 74.6% for semantic frame target word selection, and 70.4% to 77.0% for frame element classification. A prototype of the system was approbated by LETA news agency. Based on the SemEval-2015 competition results and their practical application in the LETA media monitoring automation, we were able to join an international consortium submitting and winning a Horizon-2020 project "SUMMA" under H2020-ICT-16 BigData-research call. Building on the C6.0 classification algorithm expertise we have developed a character-level neural translation methodology [2] SD and automatic thesaurus corpus-sample selection methodology [3]. In the year 2016 (stage 3) we continued research in information retrieval and semantic parsing with an application of our earlier Semeval-2015 approach to the formalism of Abstract Meaning Representation (AMR). AMR parsing extends the FrameNet micro-relations concept and attempts to build a semantic graph of all relations within a sentence. We managed to achieve excellent AMR parsing accuracy, resulting in the first place in Task 8 of Semeval-2016 shared task competition [4]. In addition to integration of the C6.0 0 classifier with the AMR SMATCH scoring tool to improve accuracy of the CAMR parser, we implemented an ensemble with a character level sequence-to-sequence neural network model for semantic parsing with methods inspired by neural machine translation. Exploration of these technologies also resulted in publications about applications of AMR in text summarization [6] and deep neural networks for Latvian tagging [5], and the development of several master's thesis on these technologies.

References.

[1] Barzdins G., Paikens P., Gosko D.. Riga: from FrameNet to Semantic Frames with C6.0 Rules. SemEval 2015 Task 18: Semantic Dependency Parsing. Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), Association for Computational Linguistics, pp. 960–964. (http://www.aclweb.org/anthology/S15-2160).

[2] G.Barzdins, S. Renals, D. Gosko. (2016). Character-level Neural Translation for NextGen Media Monitoring in the SUMMA Project. Accepted by LREC-2016, 23-28 May 2016, Portorož (Slovenia). (Web of Science)

[3] A.Spektors, I. Auzina, R. Dargis, N. Gruzitis, P. Paikens, L. Pretkalnina, B. Saulite. (2016). Tezaurs.lv: the Largest Open Lexical Database for Latvian, Accepted by LREC-2016, 23-28 May 2016, Portorož (Slovenia). (Web of Science)
[4] Guntis Barzdins, Didzis Gosko. RIGA at SemEval-2016 Task 8: Impact of Smatch Extensions and Character-Level Neural Translation on AMR Parsing Accuracy. Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), Association for Computational Linguistics, pp. 1143-1147. URL https://aclweb.org/anthology/S/S16/S16-1176.pdf. (to be indexed ACL)
[5] Peteris Paikens. Deep Neural Learning Approaches for Latvian Morphological Tagging. Frontiers in Artificial Intelligence and Applications, Volume 289: Human Language Technologies – The Baltic Perspective, I. Skadiņa and R. Rozis (Eds.). IOS Press, 2016, pp 160-166. DOI 10.3233/978-1-61499-701-6-160 URL http://ebooks.iospress.nl/volumearticle/45531 (SCOPUS)
[6] N. Gruzitis and G. Barzdins. The role of CNL and AMR in scalable abstractive summarization for multilingual media monitoring. Controlled Natural Language, Controlled Natural Language 5th International Workshop, CNL 2016, Davis, Brian, Pace, Gordon J., Wyner, Adam (Eds.), LNAI, Volume 9767, pp. 127-130, Springer 2016. doi = "10.1007/978-3-319-41498-0" (SCOPUS)

2.1.2.1 FrameNet micro-relation ontology formalization in the form of AMR (Abstract Meaning Representation) and development of appropriate machine learning methods for semantic analysis of text. Research of application of these innovative methods in other areas, e.g., text generation and robotics (stage 4).

In this project stage the following AMR (Abstract Meaning Representation) research has been carried out:

- Participation in SemEval-2017 workshop Task 9: Abstract Meaning Representation Parsing and Generation (co-located with ACL 2017). We won the Generation track by integrating AMR and GF (Grammatical Framework) approaches as described in our paper [2].
- Participation in TAC KBP Task 2016 at MNIST. A pre-defined ontology Knowledge Base there had to be populated with facts from English source text. Using AMR parser we achieved highest precision among participants, but due to low recall overall results were mediocre. Detailed description in our paper [3].
- Doctoral thesis by Peteris Paikens (adviser prof. Guntis Barzdins) "LATVIAN SEMANTIC PARSING TOOLCHAIN" has been completed and successfully defended 5/12/2017. It describes AMR and FrameNet micro-ontology approach for semantic parsing of Latvian.

In parallel we transferred our deep learning expertise to the field of robotics:

- Master thesis by Nauris Dorbe (adviser prof. Guntis Barzdins) "DRIVERLESS CARS TUITION USING REINFORCEMENT LEARNING ARTIFICIAL DEEP NEURAL NETWORKS WITHIN COOPERATE DRIVING SYSTEM" has been completed and achieved 1.place in the Latvian Master thesis competition ZIBIT.
- Doctoral thesis by Uldis Locans (adviser Guntis Barzdins) "Future Processor Hardware Architectures for the Benefit of Precise Particle Accelerator Modeling". Among other topics it explores GPU computing architectures relevant for training deep neural networks.

- Prepared a paper [1] for NIPS 2017 Workshop on Visually-Grounded Interaction and Language (ViGIL) exploring language and robotics relationship via grounded reinforcement learning.
- Masters degree course "Deep Learning" updated with a lecture on deep reinforcement learning.

References.

1. G. Barzdins, R. Liepins, P. F. Barzdins, D. Gosko. dBaby: Grounded Language Teaching through Games and Efficient Reinforcement Learning. NIPS 2017 Workshop on Visually-Grounded Interaction and Language (ViGIL), 2017
https://nips2017vigil.github.io/papers/2017/dBaby.pdf

2. N. Gruzitis, D. Gosko, G. Barzdins, RIGOTRIO at SemEval-2017 Task 9: Combining machine learning and grammar engineering for AMR parsing and generation Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval), 2017
http://www.aclweb.org/anthology/S17-2159

3. P. Paikens, G. Barzdins, A. Mendes, D. Ferreira, S. Broscheit, M. S. C. Almeida, S. Miranda, D. Nogueira, P. Balage, A. F. T. Martins, SUMMA at TAC Knowledge Base Population Task 2016
Proceedings of the 9th Text Analysis Conference (TAC), 2017
https://tac.nist.gov/publications/2016/participant.papers/TAC2016.summa.proceedings.pdf

2.2. Knowledge engineering and semantic web technologies for e-learning, multi-agent systems, business process modelling, e-logistics, and software development

The Section 2.2 describes scientific results of SOPHIS programme Project No. 2 that were obtained by the researchers of the **Faculty of Computer Science and Information Technology, Riga Technical University (RTU FCSIT)**. The studies were focused on the research and development of knowledge engineering and semantic web technologies for knowledge formalization, reuse and sharing in the context of e-learning, multi-agent systems, business process modelling, e-logistics, and software development.

2.2.1. Development of automated methods and algorithms for the system's structural model analysis and their implementation in I4S

In 2014 the most important scientific and practical result was the development of several methods and algorithms for system's structure formalization and transformation necessary for knowledge structure models used in structural modelling, as well as implementation of a prototype of I4S software tool for model representation and analysis. The first year students of RTU doctoral study programme "Computer Systems" M. Pudāne, S. Šķēle and H. Grīnbergs participated in testing of the prototype.

In 2015 a novel formal knowledge structure transformation method and algorithm was developed. According to this method, a morphological structure model (MSM) is transformed into a functional structure model in behaviour space (FSM BS), which is used as an intermediate model supporting the next step of transformation – the construction of functional structure model in parameter space (FSM PS). The latter supports evaluation of functional state of complex industrial control systems on the basis of experts' knowledge about changes of parameter values caused by different faults. The method was implemented in the I4S software tool, and this new functionality was tested by the first year student of RTU doctoral study programme "Computer Systems" Ē. Urtāns. Using the I4S, structural models (MSM and FSM) of control system for winch handling system for the company ICD Software (Norway) was developed. However, the final goal – evaluation of functional state of the abovementioned system was not reached because cooperation with experts from ICD Software was interrupted (the new management of the company decided to postpone it). Therefore, a decision to widen the research of knowledge structures was made to enable new applications of the approach and the I4S. For this purpose an initial research phase on knowledge structures in the form of concept maps (CMs) was carried out towards the development of formal method for evaluation of CM complexity based on criteria used in Systems Theory. It was proposed to interpret and use for CMs the four criteria applied for estimation of systems complexity – the number of system's elements and relationships between them, attributes of systems and their elements, and the organizational degree of systems.

In 2016 the functionality of the I4S tool was further extended. A novel method was developed for assessment of the importance of elements for knowledge structures of different types and granularities. The initial model that it used for this purpose is the aggregated model of morphological structure (MSM), for which the importance of each element in the whole structure is assessed using the structural modelling

approach. Three criteria are used – local, global, and causal connectedness. After transformation (homomorphism of models is ensured) of the initial model into a new model that has deeper level of granularity, the assessment of element importance is repeated. The obtained results are summed for subsets of elements that correspond to each element of initial model. For implementation of the method, the corresponding algorithm has been developed.

The work started during the previous year, which was focused towards the development of a formal method for evaluation of concept map complexity from the systems viewpoint, was continued. Evaluation of complexity of concept maps as one kind of knowledge structure representation is based on the criteria used in Systems Theory. The set of criteria was extended by a new formula for calculation of structural complexity, degree of centralization of structure, and relative weight of hierarchical levels. As a result, a framework for multicriterial evaluation of concept map complexity was created.

As the development and analysis of different types of systems is essential to classification of knowledge structures, a repository of knowledge structures was built using the I4S software tool. The repository serves as a central part where knowledge about different types of systems is stored. The entry of knowledge structures of systems is made according to a previously developed methodology, and these descriptions are stored as different models in the I4S. The functionality of software ensures a convenient way for transition between different type of models, which, in turn, is essential for carrying out appropriate research models and causal analysis.

Testing of methods for assessment of element importance in knowledge structures of different types and granularities using various criteria was started using the developed repository of knowledge structure models. Research concerning the possibilities of combining different knowledge structures used in distributed artificial intelligence with the focus on network schemas was carried out as well.

In 2017 studies were focused on the approbation of previously developed methods and tools, on the development of new methods for the transformations and processing of knowledge structures as well as on increasing the autonomy of concept map-based knowledge assessment system by utilizing a relations replacement web that contains information about possible replacement of one linking phrase with another. The obtained scientific results are described in next subsection.

2.2.2. Development of methods for transformation of knowledge structures and approbation of intelligent structural modelling tool I4S in study process (period 4)

Introduction

The main objective or research carried out during the period 4 is to approbate those methods and tools which were developed in previous periods as well as to develop new methods for knowledge structures' transformations and processing.

2.2.2.1. Approbation of the prototype of the tool I4S with extended functionality for the evaluation of concept map complexity and importance of its elements

The complexity of concept maps is evaluated in accordance to the framework which development was started during the period 3. The basic principles of the framework follow next (more details can be found in [1]).

The framework for the evaluation of complexity of concept maps

The central idea of the approach is based on interpretation of CMs as systems (as a whole) and application of criteria used in Systems Theory for estimation of complexity of systems to CMs. In Systems Theory, as a rule, two quantitative parameters are used – the number of system's elements and the number of implemented relationships. Logically, it is declared that simple systems have a small number of elements and relationships, while complex systems consist of a large number of elements and relationships [2]. These parameters are relative and only shallowly evaluate the complexity of systems. Some improvements are known which suggest using expert's knowledge who evaluates the complexity of each element and then summing up these evaluations to get a conjunctive parameter of complexity. An awkward attempt to ask experts to evaluate complexity of relationships comparing them with complexity of elements is also proposed despite the fact that such approach is useless in practice. Authors in [3] have shown that complexity also depends on other aspects, such as the knowledge about organization of system and attributes of its specific elements, which may substantially change the evaluation of system's complexity, so that a very complex system at first sight, in fact, is simple for an expert.

Taking abovementioned into account, the following criteria are proposed [3]:

- The number of elements
- The number of relationships
- The attributes of specific elements of the system
- The organizational degree of the system

The one-to-one correspondence between these criteria, which are supplemented with introduced additional ones, namely, attributes of the system and attributes of a relationship, in case of systems and in case of CMs is defined in Table 1.

Table 1. Correspondence of complexity criteria

No	System	Concept map
1	The number of elements	The number of concepts
2	The number of relationships	The number of arcs
3	Attributes of the system	Linking phrases (their number and variety of types and/or the number of synonyms of concepts)
4	Attributes of an element	The structural importance of a concept
5	Attributes of a relationship	The weight of an arc
6	The organizational degree of the system	The topological features of the corresponding graph

All results and considerations without loss of generality are shown using two incoming trees $T(V,Q)$ and $T^*(V^*,Q^*)$, where $|V| = 4$ and $|V^*| = 5$. These trees have been chosen as a trade-off between very simple trivial cases ($|V| = 2$ or 3) and more complicated ones ($|V| = 6,7,\ldots$). Figures 1 and 2 represent $T(V,Q)$ and $T^*(V^*,Q^*)$, respectively (different topologies of trees are divided into categories T_i and T_j^*).

It is obvious that similarly with the general case, the first two criteria help nothing because all trees shown in Figure 1 or Figure 2 have the same complexity, correspondingly. As a consequence, one can obtain a more complex CM only by adding new concepts and increasing in such a way the number of concepts and arcs.

The situation changes if the third criterion, namely, system's attributes is introduced. Interpretation of system's attributes in case of CMs is tightly connected with semantics of concepts and linking phrases. For example, if CMs can be constructed with free vocabulary, different learners can use different words or linking phrases for the same concept and arc, respectively. The CM is complex comparing with a practically identical CM with the only difference being that all concepts and linking phrases are predefined unambiguously. This conclusion refers to the both cases – construction of a CM and its assessment by comparison with an expert's CM. The latter task leads to the graph matching problem [4].

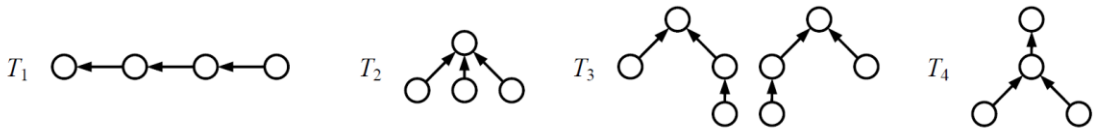


Fig. 1. Categories of topologies of T(V;Q)

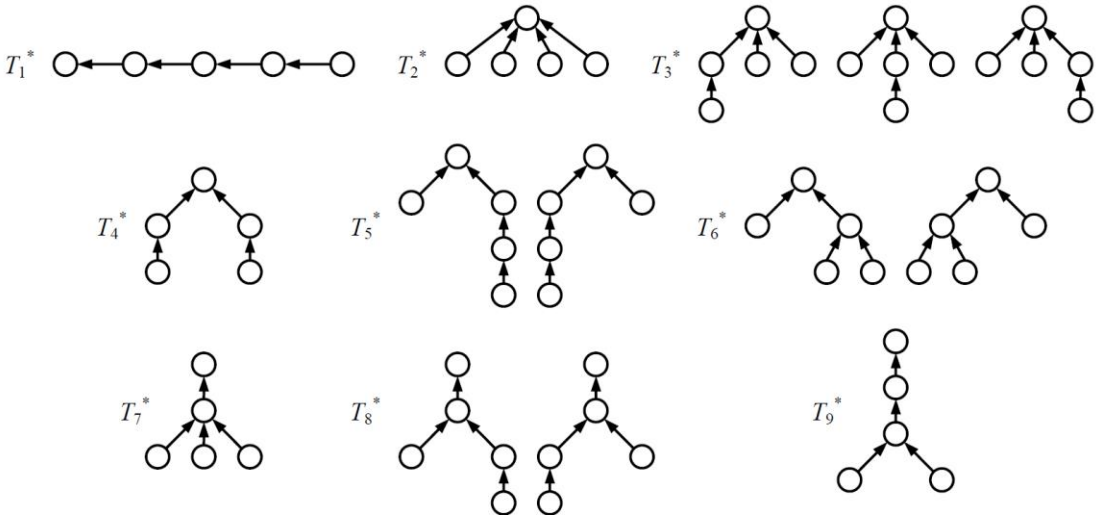


Fig. 2. Categories of topologies of T*(V*;Q*) (Adapted from [8])

Thus, concerning concepts, the conclusion is that the complexity of a particular CM increases if the number of synonyms is growing. For example, in cybernetics "the summation point" is also termed "the comparator", "the measurement point", and "the error detector". As a consequence, the presence of synonyms brings in more complexity both during the task solution and the CM assessment. Linking phrases also are expressed in a natural language, which is not unambiguous. If linking phrases are not given to the CM creator then he/she may use any expression that seems appropriate according to his/her understanding of how concepts are related in a particular domain. Moreover, the semantics of relationships of the same two concepts can vary depending on the context in which they are used [5], as well as there can even be cases when it is meaningful to represent more than one relationship between two concepts [5, 6]. Such situation is not inspiring because the variance of linking phrases theoretically is indefinite. For example, research completed by Strautmane [7] shows that for inheritance relationship alone, there are more than 50 ways how to label it. That is the reason why researchers of semantic networks and CMs have defined typical linking phrase types, such as "is a", "part of", "kind of", "is an

example of", "is an attribute of", "is the value of", "characterizes", etc. along with others.

The working hypothesis is that the complexity of CM (and, as a result, the complexity of CM-based task) increases if the variety of linking phrases increases and vice versa. It is obvious that the number of arcs may restrict the maximum number of used linking phrase types. For the examples represented here, the maximum number of linking phrase types is 4 (for *T*(V*;Q*)*).

For estimation of CM complexity using only the third criterion, for attribute graphs the following formula is used:

$$C^{\text{att}} = N_{\text{LPT}} \cdot \sum_{i=1}^{N_{\text{LPT}}} w_i \cdot n_i \,, \qquad \textbf{(1)}$$

where N_{LPT} is the number of linking phrase types, w_i is the weight of the *i*-th linking phrase type, and n_i is the number of linking phrases of the *i*-th type in a CM.

The hard point is evaluation of weights, the values of which may be different in different areas. Experience obtained by working with the IKAS has shown that students of engineering courses had more difficulties with the "part of" relationship than with the "is a" relationship. Accordingly, weights may be defined as follows: 2 for the "part of" relationship and 1 for all other linking phrase types. Applying the Eq. (1), the complexity of CM depicted in Figure 3a is 4, while the complexity of CM shown in Figure 3b is 20.

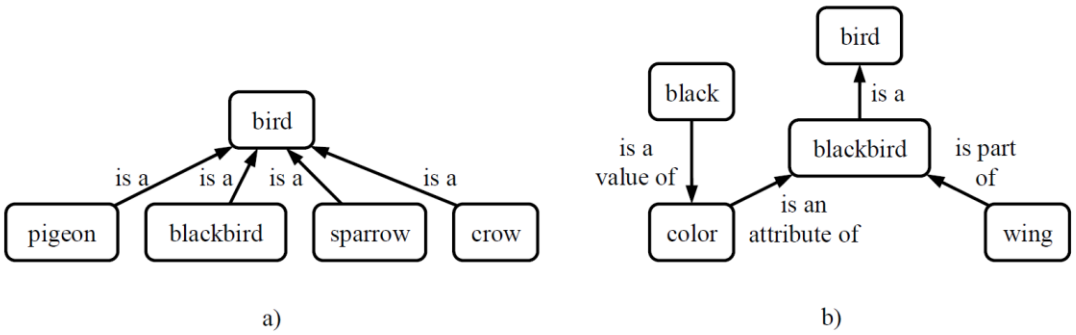


Fig. 3. Examples of CMs with different complexity (Source: [8])

In general, if the weights of linking phrase types are ignored, the minimal complexity has a CM with only one type of linking phrases, while the maximal complexity has a CM with unique types of linking phrases. The rationale of this statement can be founded on the fact that the frequency to make wrong decisions in the second case is significantly higher comparing with the first case.

It is necessary also to point that the weight of an arc representing the importance of the relationship may be taken into account in a similar way.

Now, let's consider how to evaluate the complexity of CMs according to the topological features of the corresponding underlying graphs. It is proposed to use one criterion which is borrowed from scoring systems used for CMs – the number of valid levels of hierarchy N_H, which shows where on the general–specific continuum each concept lays in respect to the domain being represented. The number of levels of hierarchy is related to the extent to which the learner subsumes more specific knowledge under more general knowledge [9]. In graph theory, the number of hierarchy levels is equal to the diameter of the tree. Such criteria as the complexity of

structure, the relative weight of each hierarchy level, and the degree of centralization of structure are borrowed from [10]. Other graph theory criteria which already are used in structural modeling [11] will be applied for determination of the structural importance of a concept in the next section.

First, it is worth to stress that the following parameter for evaluation of complexity of systems, which is based on the consideration of complexity of structural analysis [10], is not applicable:

$$C_S = \frac{1}{|V_{in}| \cdot |V_{out}|} \sum_{i=1}^{V_{in}} \sum_{j=1}^{V_{out}} P_{ij} - 1 \,, \qquad \textbf{(2)}$$

where $|V_{in}|$ and $|V_{out}|$ are the number of system's inputs and outputs, correspondingly, and P_{ij} is the path from any input to any output of the system. It is easy to see that for each incoming tree with one root node, the complexity is always equal to zero independently of the number of apex nodes.

For this reason, for calculation of the complexity of structure of CM, the following modification of Eq. (2) is suggested:

$$C_S = \frac{1}{|V_{apex}|} \sum_{i=1}^{V_{root}} P^W_{i,root} - 1 \,, \qquad \textbf{(3)}$$

where $|V_{apex}|$ is the number of apex nodes in the incoming tree, V_{root} is the root node, and $P^W_{i,\,root}$ is the weighted path from any apex node to the root. The $P^W_{i,\,root}$ is found as follows:

$$P^W_{i,root} = d_{i,root} + \sum_{j=1}^{S_i} d_{j,root} \,, \qquad \textbf{(4)}$$

where $d_{i,root}$ is the distance from the apex node to the root, S_i is the number of descendants of apex node *i*, and $\Sigma_{j=1}^{S_i} d_{j,root}$ is the sum of distances from all descendants of apex node *i* to the root. For example, the complexity of structure of the incoming tree T^*_5 (see Figure 2) is

$$C_S^{T_5^*} = \frac{1}{2} \cdot (1 + (3 + 2 + 1)) - 1 = 2.5$$

The T^*_5 has four valid hierarchy levels (the root node is always placed at 0-level), and the relative weights of hierarchy levels are the following: 0-level – 0.2, 1st level – 0.4, 2nd level – 0.2, 3rd level – 0.2.

The degree of centralization of structure is calculated as follows [10]:

$$D_C = \frac{1}{(n-1)(\rho_{max} - 1)} \sum_{i=1}^{n} (\rho_{max} - \rho_\Sigma(V_i)) \,, \qquad \textbf{(5)}$$

where *n* is the number of nodes, $\rho_\Sigma(Vi) = \rho^+(Vi) + \rho^-(Vi)$, where $\rho^+(Vi)$ and $\rho^-(Vi)$ denote outdegree and indegree [12] of the node V*i*, and ρ_{max} is the maximum value of ρ_Σ for the given structure. For example, the degree of centralization of T^*_5 is:

$$D_C^{T_5^*} = \frac{1}{(5-1)(2-1)} ((2-2) + (2-1) + (2-2) + (2-2) + (2-1)) = 0.5$$

The structural modeling approach [11] offers also other parameters for evaluation of topological characteristics of structure, for instance, redundancy of arcs (not applicable for trees, which have the minimum number of arcs), compactness of structure, and the dispersion of ranks of nodes. Now, according to the approach

described above, the topological features of each category of the underlying graph of CM are evaluated using the three criteria N_H, D_C, and C_S. For integration of these criteria into one parameter, the procedure of ranking is suggested. The essence of it is the following: first, graph categories are ranked using only one selected criterion (others are neglected), taking into account the corresponding CM task difficulty. Second, the ranking is carried out for each criterion. Third, the sum of ranks is calculated, and the graph categories are ordered in compliance with the presumption of the degree of CM task difficulty.

The following assumptions are accepted:

1. The degree of task difficulty is higher if the value of N_H is greater because a learner must subsume a greater number of more specific knowledge under more general knowledge.
2. The degree of task difficulty is higher if the value of D_C is smaller because each relationship (linking phrase) has relatively greater impact on correctness of CM as a whole.
3. The degree of task difficulty is higher if the value of C_S is greater because propositions are more interrelated.

The relative weights of hierarchy levels of *T(V,Q)* given in Figure 1 are shown in Table 2. The results of calculations of criteria values and ranking for topological categories of *T(V,Q)* are collected in Table 3, while the ordering of topological categories of T*(V,Q)* according to the sum of ranks is given in Table 4.

Table 2. Relative weights of hierarchy levels of *T(V,Q)*

Category	0-level	1st level	2nd level	3rd level
T_1	0.25	0.25	0.25	0.25
T_2	0.25	0.75		
T_3	0.25	0.5	0.25	
T_4	0.25	0.25	0.5	

Table 3. Values of criteria and ranking of categories of *T(V,Q)*

Category	N_H	Rank N_H	D_C	Rank D_C	C_S	Rank C_S
T_1	4	1	0.67	1	5	1
T_2	2	3	1	2	0	4
T_3	3	2	0.67	1	1	3
T_4	3	2	1	2	1.5	2

Table 4. Ordering of topological categories of *T(V,Q)*

Category	T_1	T_3, T_4	T_2
Sum of ranks	3	6	9
	← the most difficult		the easiest →

The relative weights of hierarchy levels of *T*(V*,Q*)* given in Figure 2 are shown in Table 5. The results of calculation of criteria values and ranking for topological categories of *T*(V*,Q*)* are collected in Table 6, while the ordering of topological categories of *T*(V*,Q*)* according to the sum of ranks is given in Table 7.

The number of underlying graphs of CMs grows rapidly. For example, incoming tree with 6 nodes has 19 topological categories, while if there are 7 nodes, the number of categories is 37. At first it seems that calculations and ranking may be very time consuming, but that is not so. Apart from the topological structure, all categories with the same ρ_{max} have equal numerical values of D_C. The value of C_S is growing if the value of N_H is growing, but in case if several graphs have the same N_H, the greatest value of C_S depends on the relative weight of the hierarchy level (compare, for instance, T^*_5, T^*_8, and T^*_9 in Table 6).

Table 5. Relative weights of hierarchy levels of *T*(V*,Q*)*

Category	0-level	1st level	2nd level	3rd level	4th level
T_1^*	0.2	0.2	0.2	0.2	0.2
T_2^*	0.2	0.8			
T_3^*	0.2	0.6	0.2		
T_4^*	0.2	0.4	0.4		
T_5^*	0.2	0.4	0.2	0.2	
T_6^*	0.2	0.4	0.4		
T_7^*	0.2	0.2	0.6		
T_8^*	0.2	0.2	0.4	0.2	
T_9^*	0.2	0.2	0.2	0.4	

Table 6. Values of criteria and ranking of categories of *T*(V*,Q*)*

Category	N_H	Rank N_H	D_C	Rank D_C	C_S	Rank C_S
T_1^*	5	1	0.5	1	9	1
T_2^*	2	4	1	3	0	8
T_3^*	3	3	0.875	2	0.67	7
T_4^*	3	3	0.5	1	2	5
T_5^*	4	2	0.5	1	2.5	4
T_6^*	3	3	0.875	2	1.33	6
T_7^*	3	3	1	3	2	5
T_8^*	4	2	0.875	2	3.5	3
T_9^*	4	2	0.875	2	5	2

Table 7. Ordering of topological categories of *T*(V*,Q*)*

Category	T_1^*	T_9^*	T_5^*, T_8^*	T_4^*	T_6^*, T_7^*	T_3^*	T_2^*
Sum of ranks	3	6	7	9	11	12	15
	← the most difficult						the easiest →

A very shallow analysis of the results collected in Tables 4 and 7 clearly shows that the CM task with the highest degree of difficulty always is the task that corresponds to the underlying graph known as a chain (the deepest hierarchy – T_1 and T^*_1), while the bipartite graph (T_2 and T^*_2) represents the CM task with the lowest degree of difficulty (of course, it is true if only the fourth criterion is taken into account).

Determination of structural importance of concepts

The main idea is borrowed from the structural modeling [11], but in case of CMs, the method must be modified.

The determination of structural importance of concepts in a CM is based on the usage of local and global information contained in the corresponding underlying graph. Three parameters are used. The first is $P_1 = \rho\Sigma(V_i)$, which takes into account local information, that is, how many direct relationships has the selected node (a concept in a CM, respectively). All nodes are ranked using so called first rank R_1. The node V^*_i with the maximum $\rho\Sigma(V^*_i)$ is ranked as the first.

The second parameters P_2 is the number of paths from all apex nodes to the root node that include the selected node. This global information represents connectedness of the selected node with all other nodes and indirectly shows the role of the corresponding concept in understanding the whole CM. The nodes are ranked in accordance with this parameter using so called second rank R_2 (the highest rank receives the node with the greatest number of paths).

The third parameter P_3 is the number of nodes in the reachability component that includes the selected node and all its descendants. This parameter also represents global information and in case of CMs may be interpreted in the following way: the lack of knowledge of the corresponding concept may cause the lack of knowledge of all other concepts in the reachability component. So, the larger is the reachability component, the higher rank is assigned to this node (so called third rank R_3).

The next step is to calculate the sum of ranks $R_\Sigma = R_1+R_2+R_3$. In essence, the $R_{\Sigma j}$ is the sum of places according to the parameters P_1, P_2, and P_3. That is the reason why the common rank $R_{com\,j}$ of the node is obtained following the principle that the less is the value of $R_{\Sigma j}$ the higher is $R_{com\,j}$ of the node V_j.

The R_{com} is used for calculation of the structural importance of node

$$SI_j = 1 + \frac{1 - R_{\mathrm{com}_j}}{R_{\max}} \qquad \textbf{(6)}$$

where SI_j is the structural importance of the node V_j (a concept in a CM), $R_{com\,j}$ is the common rank of the node V_j, and R_{max} is the maximum value of R_{com} in the graph. For better understanding of the proposed method, two categories of incoming trees *T*(V*,Q*)* are chosen (see Figure 4).

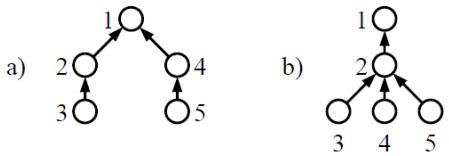


Fig. 4. Examples of two categories of incoming trees *T*(V*,Q**

Results of calculations and ranking as well as the structural importance of nodes are collected in Table 8.

Table 8. Values of parameters P_1, P_2, P_3, ranking and structural importance of nodes

	Graph in Figure 4a									*Graph in Figure 4b*								
Node	P_1	R_1	P_2	R_2	P_3	R_3	R_Σ	R_{com}	*SI*	P_1	R_1	P_2	R_2	P_3	R_3	R_Σ	R_{com}	*SI*
1	2	1	2	1	1	3	5	1	1	1	2	3	1	1	3	6	3	0.33
2	2	1	1	2	2	2	5	1	1	4	1	3	1	2	2	4	1	1
3	1	2	1	2	3	1	5	1	1	1	2	1	2	3	1	5	2	0.67
4	2	1	1	2	2	2	5	1	1	1	2	1	2	3	1	5	2	0.67
5	1	2	1	2	3	1	5	1	1	1	2	1	2	3	1	5	2	0.67

The results collected in Table 8 are rather interesting. For the graph in Figure 4a, all nodes have the same structural importance. So, each concept in a CM with such structure is equally important for knowledge acquisition. The graph in Figure 4b has one most important node (node 2). Thus, mastering the corresponding concept is the key for good assessment results.

Each of 114 students' developed concept maps were processed using the intelligent structural modeling tool I4S that supports the framework and the method for determination of structural importance of concepts described in this section. The comprehensive complexity of concept map of each student was found which may be used for his/her knowledge assessment (maximum complexity has an expert's concept map and it corresponds to maximum of points given for such concept map). Besides, determination of structural importance of concept map elements reveals a student understanding of what he/she thinks to be relevant in the learned topic.

References

1. Grundspenkis J. (2016). Towards the Formal Method for Evaluation of Concept Map Complexity from the Systems Viewpoint. Databases and Information Systems IX, G. Arnicans et al. (Eds.), Frontiers in Artificial Intelligence and Applications, Vol. 291. IOS Press, pp. 341-354.
2. Skyttner, L.: General Systems Theory: Problems, Perspectives, Practice. World Scientific, Singapore (2005)
3. Schoderbek, P.P., Schoderbek, C.G., Kefalas, A.G.: Management Systems: Conceptual Considerations. Richard D. Irwin, Burr Ridge (1990)
4. Souza, F.S.L., Boeres, M.C.S., Cury, D., Menezes, C.S., Carlesso, G.: An Approach to Comparison of Concept Maps Represented by Graphs. In: 3rd International Conference on Concept Mapping, pp. 205–212 (2008)
5. Jonassen, D.H.: Computers in the Classroom: Mindtools for Critical Thinking. Prentice Hall, Upper Saddle River (1996)
6. Shavelson, R.J., Lang, H., Lewin, B.: On Concept Maps as Potential "Authentic" Assessments in Science. Technical report, National Center for Research on Evaluation, Standards, and Student Testing (1994)
7. Strautmane, M.: Usage of Semantics of Links as the Basis for Learner's Knowledge Structure Assessment: The Pros and Cons. In: Joint International Conference on Engineering Education and International Conference on Information Technology, pp. 294–303 (2014)

8. Grundspenkis, J.: Initial Steps towards the Development of Formal Method for Evaluation of Concept Map Complexity from the Systems Viewpoint. In: 12th International Baltic Conference on Databases and Information Systems, pp. 366–380. Springer Verlag (2016)
9. Borda, E.J., Burgess, D.J., Plog, C.J., DeKalb, N.C., Luce, M.M.: Concept Maps as Tools for Assessing Students' Epistemologies of Science. Electronic Journal of Science Education 13(2), 160–185 (2009)
10. Nikolayev, V.I., Bruk, V.M.: Sistemotehnika: Metodi i Prilozeniya (in Russian). Masinostroyeniye, Leningrad (1985)
11. Grundspenkis, J.: Reasoning Supported by Structural Modelling. In: Wang, K., Pranevicius, H. (eds.) Lecture Notes of the Nordic-Baltic Summer School "Intelligent Design, Intelligent Manufacturing and Intelligent Management", pp. 57–100. Kaunas University of Technology, Kaunas (1999)
12. Agnarsson, G., Greenlaw, R.: Graph Theory: Modeling, Applications, and Algorithms. Pearson Education, Upper Saddle River (2007)

2.2.2.2. Development of method for knowledge structures transformations

Research on linking and common use of different knowledge structures used in distributed artificial intelligence (DAI) allows formulation of two basic principles of the developed method: 1) the fundamental schema for knowledge representation is ontology; and 2) ontology is transformable (directly or indirectly) into traditional AI knowledge representation schemas – semantic networks and their equivalent – concept maps, frame system, production rules, and knowledge structures formalized using first order logic language. The developed method which is focused on network structures includes 5 mutual transformation pairs as it is shown in Figure 1.

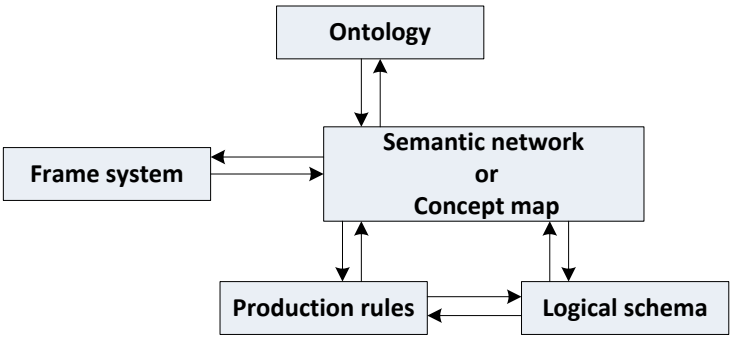


Fig. 1. Mutual transformations of knowledge representation formalisms

The mutual transformation pairs are the following:

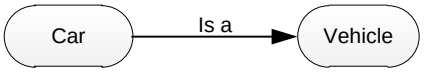
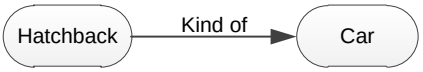
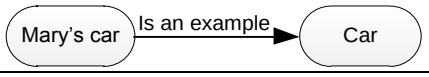
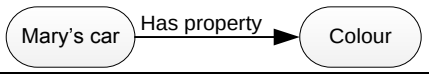
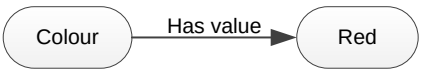
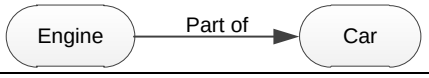
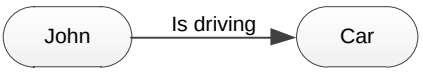
1. ontology – semantic network or concept map;
2. semantic network or concept map – production rules;
3. semantic network or concept map – frame system;
4. semantic network or concept map – logical schema (propositional calculus as knowledge representation language);
5. production rules – logical schema.

The method consists from 10 algorithms for semiautomatic mutual transformations. All transformations follows one-to-one mapping on syntactic elements of source schema on syntactic elements of target schema. For example, in transformation of ontology to semantic network it is one-to-one mapping of class concept of ontology to a node representing this class in semantic network. All ten algorithms are described in "Description of method for transformations of knowledge structures" which is available on request at the Department of Artificial Intelligence and Systems Engineering of Riga Technical University. Algorithms of mutual transformation of ontology to concept map and vice versa are given below.

Algorithm of concept map transformation into ontology

The algorithm was designed before the current project started and for implementation into intelligent concept map based tutoring system IKAS [1]. The IKAS supports the range of predefined linking phrases as well as ones defined by a user. In Table 1 all linking phrases and elements together with examples are shown.

Table 1. Types of linking phrases supported in the IKAS

#	Linking phrase	Linked elements	Example
1	Is a	Subclass->class	
2	Kind of	Subkind->kind	
3	Is an example	Instance->class	
4	Has property	Class/instance->property	
5	Has value	Property->Value	
6	Part of	Part->whole	
7	<linguistic linking phrase defined by user>	Class/instance-> class/instance	

An example of CM using these linking phrases is shown in Fig. .

CMs in the IKAS are stored as XML files which contains a list of all relations together with connected nodes (see Fig.):

- **element name="relation"** defines a relation with the following name of the linking phrase **value="Linking phrase 1"**;
- **element name="source"** is the initial concept of the relation followed by a particular value **value="Concept name 1"**;
- **element name="target"** is the terminal concept of the relation followed by a particular value **value="Concept name 2"**;
- **element name="synonym"** defines the synonym for the name of the concept or the linking phrase followed by particular values.

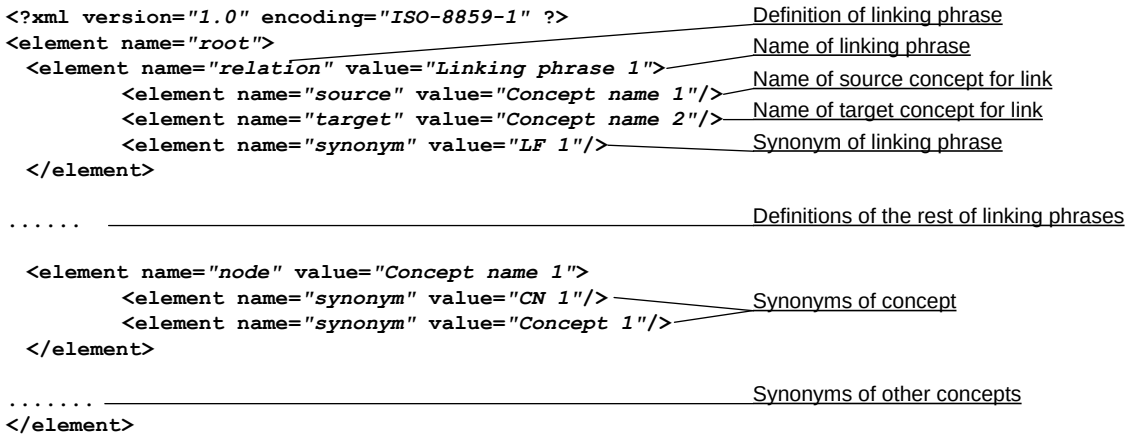


Fig. 2. Structure of XML file storing CM in the IKAS

- **Similarities between elements of concept maps and OWL ontology**

Both ontologies and CMs represent some domain, and both have classes or concepts and relations between them. Unlike CMs ontologies have also attributes for classes, their values and restrictions on them. Correspondence between elements of the CM and main elements of OWL ontology are shown in Fig. , as it was identified in **Error! Reference source not found.** in case of CM generation from OWL ontologies. A concept in the CM can correspond to a class, an instance, a datatype property and its value in the OWL ontology depending on the linking phrase which relates two concepts. The linguistic and "part-whole" linking phrases correspond to the object property in the ontology.

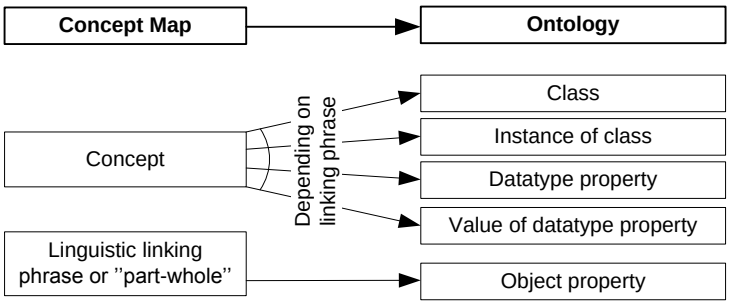
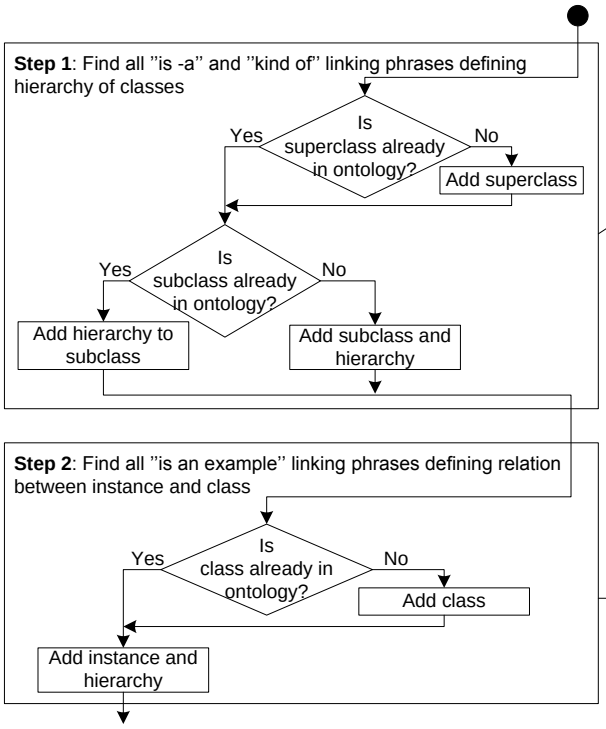


Fig. 3. Correspondence between elements of CM and elements of ontology

- **Algorithm for concept map transformation**

The proposed algorithm for CM transformation to ontology consists of 6 steps during which all elements of CM are proceeded to determine their role in the ontology and appropriate constructs to the ontology are added. First five steps are dedicated to analysis of all types of linking phrases used in the CM. The type of related concepts is determined according to linking phrases. The 6th step is for discovering synonyms defined in the CM. The simple flowcharts (actions and IF statements) of the proposed algorithm together with comments and OWL ontology fragments are shown on Fig. , 5, and 6.



All concepts that are related with theses phrases are added as classes to the ontology. Direction of the link shows direction of the hierarchy. A source concept corresponds to a subclass and a target concept to a superclass. All superclasses are defined with `<owl:Class rdf:about="Super_concept_name"/>` and subclasses together with superclass as follows:

```
<owl:Class rdf:about="Subclass_name">
  <rdfs:subClassOf rdf:resource=" Superclass_name"/>
</owl:Class>
```

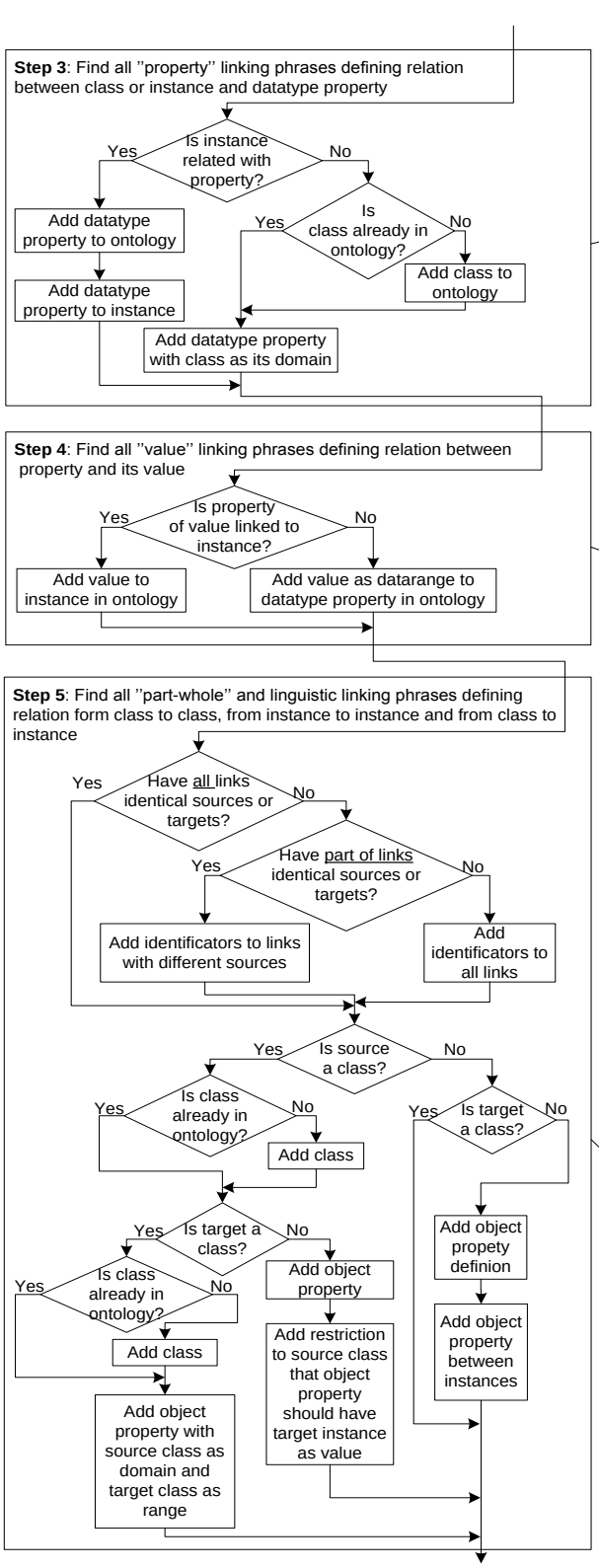
After this step all classes related in hierarchies are added to the ontology.

If the class is not already added to the ontology do it like in Step 1 and then add the instance to the ontology, or add only the instance if the class have been added in the previous step as follows:

```
<owl:NamedIndividual rdf:about="Instance_name">
  <rdf:type rdf:resource="Superclass_name"/>
</owl:NamedIndividual>
```

After this step all instances are added to the ontology.

Fig. 4. Transformation steps 1-2



The concept is an instance if it is linked with "is an example" to any other concept else it is a class. If a property is linked to the instance then, firstly, definition of datatype property is added to the ontology and, secondly, datatype property is added to the instance description:

```
<owl:DatatypeProperty
rdf:about="Property_name"/>

<owl:NamedIndividual rdf:about="Instance_name">
  <rdf:type rdf:resource="Superclass_name"/>
  <Colour></Colour>
</owl:NamedIndividual>
```

If a property is linked to the class then, firstly, add the class to the ontology, if it isn't there, then add datatype property with the class as the domain:

```
<owl:DatatypeProperty
rdf:about="Datatype_property">
  <rdfs:domain rdf:resource="Class_name"/>
</owl:DatatypeProperty>
```

If the property is assigned to the class then add the value as a datarange of this property.

If the property is assigned to the instance then add the value to the instance:

```
<owl:NamedIndividual rdf:about="Instance_name">
  <rdf:type rdf:resource="Concept_name"/>
  <Datatype_property>Value</Datatype_property>
</owl:NamedIndividual>
```

If the phrase is between classes then, firstly, check is classes already in the ontology, if not add them; secondly, define the object property with the source class as the domain and the target class as the range:

```
<owl:ObjectProperty rdf:about="Linking_phrase">
  <rdfs:domain rdf:resource="Source_class"/>
  <rdfs:range rdf:resource="Target_class"/>
</owl:ObjectProperty>
```

If the phrase is between instances then add the object property definition and assign it between instances:

```
<owl:ObjectProperty rdf:about="Linking_phrase"/>

<owl:NamedIndividual
rdf:about="Source_instance">
  <Linking_phrase
rdf:resource="Target_instance"/>
</owl:NamedIndividual>
```

If the phrase is between the class and the instance then define the object property and add a restriction to the source class that the object property should have the target instance as the value:

```
<owl:ObjectProperty rdf:ID="Linking_phrase"/>b

<owl:Class rdf:ID="Target_class">
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty
rdf:ID="Linking_phrase"/>
      </owl:onProperty>
      <owl:hasValue>
        <Other_class rdf:ID="Source_instance"/>
      </owl:hasValue>
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>
```

It is denied to have a relation from an instance to a class.

Fig. 5. Transformation steps 3-5

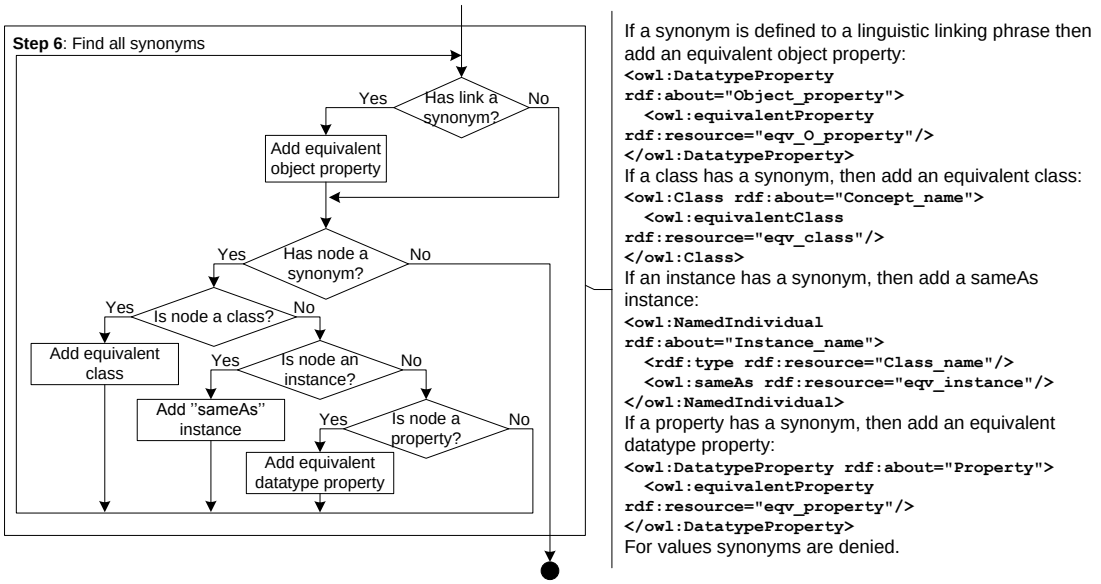


Fig. 6. Transformation step 6

- **Example of ontology obtained from the Concept Map**

The example of concept map containing all types of linking phrases supported in the IKAS is developed. In Fig. the CM is shown together with pieces of OWL ontology which corresponds to particular elements of the CM.

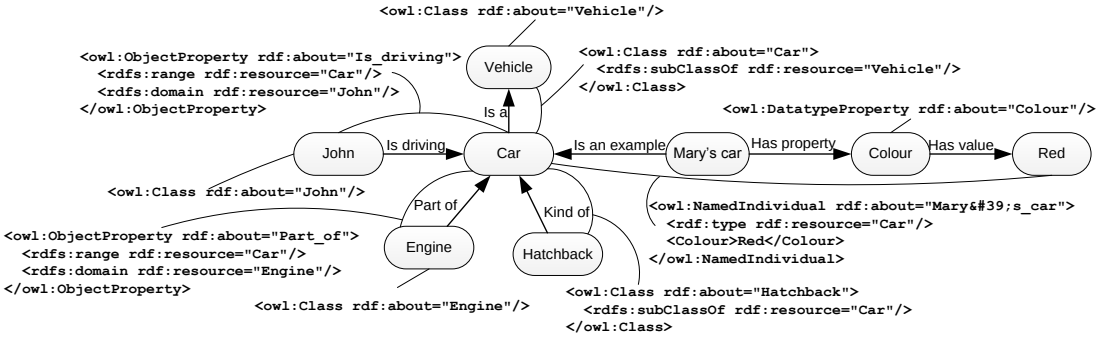


Fig. 7. Example of CM together with the corresponding OWL code

Algorithm of ontology transformations into concept map

General correspondence between OWL ontology and a concept map is shown in Figure 8, where it is illustrated which ontology elements directly corresponds to concept map elements, i.e., all ontology classes, instances, data type properties and values of data type properties are concepts and object properties correspond to links in concept maps **Error! Reference source not found.**. Besides these elements there are found and summarized also other OWL constructs which describe different properties of these elements that influence element transformation into a concept map, for example, a construct defining that one class is a subclass of other class **Error! Reference source not found.**.

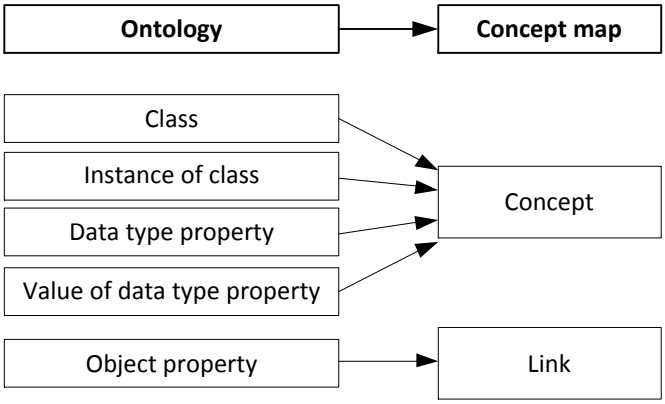


Fig. 8. Correspondence between elements of ontology and elements of CM **Error! Reference source not found.**

All cases of ontology transformation have been divided into 3 groups **Error! Reference source not found.**:

- Hierarchal relations between classes and instances – include cases related to finding of ontology classes and their instances and establishing hierarchal relations, i.e., relations between a class and a subclass, between a class and its instances, Boolean relations between classes, synonyms of classes and instances, distinctions of classes and instances.
- Semantic relations between classes and instances – include cases related to finding of object properties which define semantic relations between classes/instances.
- Property relations for classes and instances – include cases related to finding of data type properties for classes/instances.

Examples of the identified mappings between OWL ontology and concept map are overviewed in **Error! Reference source not found.**. The listings of OWL code are followed by graphical representation of corresponding concept map elements.

During operation of the algorithm from the ontology saved in the text file an extended incidence matrix is obtained, where names of concepts and their interrelations are stored showing the name of the link (linking phrase) and its direction. In addition, this matrix is extended with one more column where data about a type of concept or a label of concept type (root class – class without superclasses, subclass, instance, property or value) are stored. Basic steps for concept map generation from an ontology are the following **Error! Reference source not found.**:

Step 1: Read an ontology file and check OWL syntax.

Step 2: Find all classes (begin creation of an incidence matrix).

Step 3: Find subclasses of each class (for particular class add the link "is a" which goes from a subclass to a superclass in the matrix, add labels to root classes).

Step 4: For each class check intersection, union and collection with other classes (add the link "is a" in the matrix between appropriate classes).

Step 5: For each class check complement relations to other classes (add the link "is not" in the matrix).

Step 6: Find instances of each class (add instances and links "is instance of" between appropriate classes and instances which go from an instance to a class in the matrix, add labels to instances).

Step 7: Find data type properties for each class and instance (add properties and links "has property" between appropriate class/instance and a property in the matrix, add labels to properties).

Step 8: Find values for each data type property (add properties' values and links between a data type property and its value "has value" in the matrix, add labels to values).

Step 9: For each class, instance and data type property check equivalence (add the link "is synonym" in the matrix between appropriate elements).

Step 10: Find object properties for each class/instance (add appropriate links between classes or instances in the matrix).

Step 11: Check if an object property is inverse, symmetric or transitive (extend the matrix with appropriate links).

Step 12: Find hidden relations (relations which can be inferred using reasoners and are not directly defined in the ontology).

Step 13: Perform correction of concept and link names (replace understrike sign "_" with space). This is needed because spaces between words in the names of ontology elements are not allowed and usually they are replaced with "_".

Step 14: Display completed incidence matrix as a graph and save in XML accordingly to the format used in the IKAS.

References

1. Anohina-Naumeca, A., Grundspenkis, J., Strautmane, M. The Concept Map Based Assessment System: Functional Capabilities, Evolution, and Experimental Results. International Journal of Continuing Engineering Education and Life-Long Learning (ISSN: 1560-4624), Vol. 21, No. 4, 2011, Inderscience Enterprises Ltd., pp.308-327.
2. Cañas, A.J., J.D. Novak. Concept Mapping Using CmapTools to Enhance Meaningful Learning. In Okada A., S. Buckingham Shum S., T. Sherborne (Eds.). Knowledge Cartography: Software Tools and Mapping Techniques, London: Springer-Verlag, 2008, pp. 25-46.
3. Gouli, E., A. Gogoulou, A. Alexopoulos, M. Grigoriadou. Exploiting COMPASS as a Tool for Teaching and Learning. In Proceedings of the 3rd International Conference on Concept Mapping, September 22-25, 2008, Tallinn, Estonia & Helsinki, Finland, Vol. 2, pp. 383-390. Available at: http://cmc.ihmc.us/cmc2008papers/cmc2008-p181.pdf (accessed 10.02.2011).
4. Graudina, V., J. Grundspenkis. Concept Map Generation from OWL Ontologies. In Proceedings of the 3rd International Conference on Concept Mapping, September 22-25, 2008, Tallinn, Estonia & Helsinki, Finland, Vol. 1, 2008, pp. 173-180. Available at: http://cmc.ihmc.us/cmc2008papers/cmc2008-p263.pdf.
5. Graudina V., Grundspenkis J. Concept Map Generation from OWL Ontologies. The 3rd International Conference on Concept Mapping, September, 22-25, 2008, Tallinn, Estonia and Helsinki, Finland, 2008, pp. 173-180.
6. Graudina V. OWL Ontology Transformation into Concept Map. In Scientific Proceedings of Riga Technical University, 5th Series, Computer Science. Applied Computer Systems, Vol. 34, 2008, pp.80-91.

2.2.2.3. Development and approbation for knowledge structure processing

Two methods have been developed and approbated in the study process, i.e., in the study course "Fundamentals of Artificial Intelligence" (*certificate of RTU Department of Academic Affairs in Appendix 5*):

1. method of concept map processing for development of study materials and automated task output;
2. method for automated knowledge assessment.

Method for concept map processing

The first method which is modification of earlier developed (before the current project started) method is based on analysis of expert's (teacher's) concept map with objective to find so called graph patterns, i.e., subgraphs. For implementation of method 3 algorithms were developed.

- **The algorithm for construction of learning object content**

The algorithm is based on the principle that each learning object should contain one type of linking phrase and at least two concepts related with this link, as well as each concept should be included in at least one learning object.

Each learning object belongs to one of following types of subgraphs which include:

- A concept and all its direct subconcepts (the linking phrase "is a");
- A concept an all its instances (the linking phrase "is instance of");
- Two or more concepts related with linking phrases "has property" or "has value";
- Two or more concepts related with the linking phrase "has synonym" or a linguistic linking phrase, i.e., any other linking phrase which is defined by the teacher.

To identify subgraphs for learning objects 5 steps are performed in a loop while all linking phrases are handled:

1. Find a linking phrase in the concept map and determine if it is a linguistic one. If it is the case then determine all concepts connected with this linking phrase.
2. If the linking phrase is "is a", "is instance of" or "is part of" then find all source concepts and the single target concept connected with this linking phrase.
3. If the linking phrase is "has property" then find the single source concept and all target concepts. For all target concepts find all successors connected with the linking phrase "has value".
4. If the linking phrase is "has synonym", then find all subgraphs where concepts are connected with this linking phrase.
5. After finding links and related concepts they are added to a list which stores information about concepts and linking phrases needed to be described in each learning object.

For each type of the linking phrase different cardinalities between the source and the target concepts are determined:

- For linking phrases "is a", "is instance of" and "is part of" searches N:1 (for each target concept find all source concepts because the direction of the link is

from the specific to the general concept). An example of information included in the learning object of this kind is shown in the Fig. **1**.

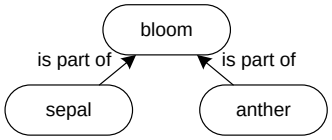


Fig. 1. A subgraph with linking phrases "is part of"

- For a linguistic linking phrase searches N:M. An example of information included in the learning object is shown in the Fig. **2**.

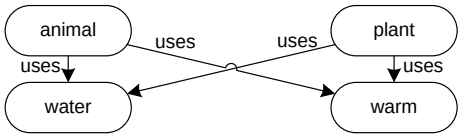


Fig. 2. A subgraph with semantic linking phrases

- For a linking phrase "has property" searches 1:N (find all properties of one source concept). Besides, the learning object should include all values of each property. Therefore for the linking phrase "has value" also searches 1:N. Thus, the learning object includes a concept with all its properties and values. An example of information included in the learning object is shown in the Fig**. 3**.

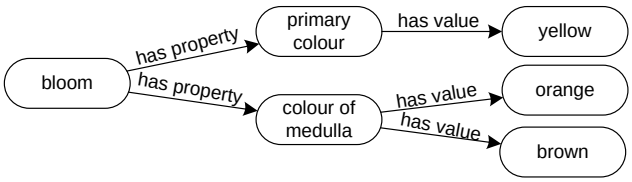


Fig. 3. A subgraph with linking phrases "has property" and "has value"

- For the linking phrase "has synonym" searches a subgraph which includes the source and the target concepts and other concepts connected with them are also interrelated with the same linking phrase. An example of information included in the learning object is shown in the Fig. **4**.

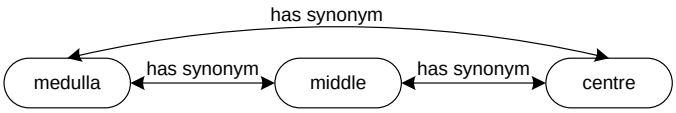


Fig. 4. A subgraph with linking phrase "has synonym"

When the algorithm finishes a teacher receives a list of learning objects together with concepts and linking phrase(s) which should be included in each of them.

- **The algorithm for generation of course learning path**

After learning objects have been authored they should be ordered in a concept map's learning path. A concept map can be developed for a whole study course or for a single topic. This learning path will be further used for searching personalized learning paths for students. A learning path or the order of learning material delivery is related with a place of learning material for learning of particular topic. Learning

materials are ordered taking into consideration prerequisite sequence of concepts. The algorithm generates an initial learning path which can be changed by a teacher if other prerequisites are needed or it demands a used pedagogical strategy.

The idea of a learning path is based on sorting of learning objects from general to specific. Learning objects with root concepts (which have not superconcepts) are located in the upper levels, after that follows learning objects with subconcepts, continuing with instances, properties and values. Steps of the algorithm are shown in the

Fig. **5**.

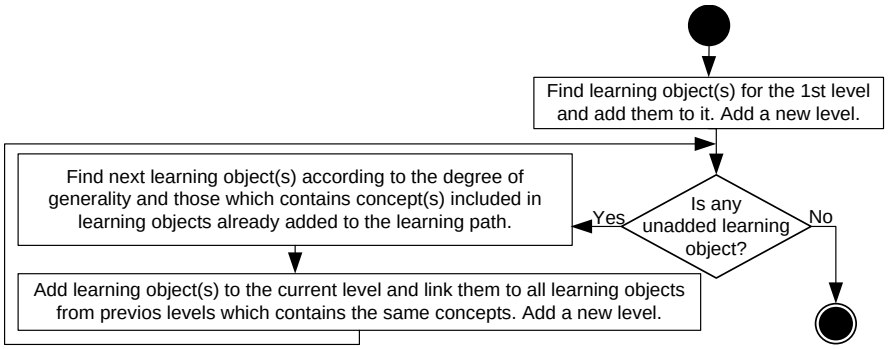


Fig. 5. The algorithm for course learning path generation

A degree of generality of a learning object is based on generality of concepts and linking phrases included in particular learning object. The most general learning object contains linguistic linking phrases and most of all concepts linked with it. For the next level in the learning path goes a learning object with next lower degree of generality and containing any of concepts from learning objects added in the previous levels.

To determine the degree of generality of a learning object there is experimentally found coherence. Learning objects are ordered according to the expression (1).

LO Order By *(R, (T+L+N))*, i.e., a set of learning objects LO is ordered twice, (**1**) firstly, sorted by R, secondly, sorted by (T+L+N), where

R – a number of root concepts in the learning object;
T – a coefficient of the relation type given below;
L – a maximal local degree of concepts included in a learning object, taking into account only links between concepts included in this learning object;
N – a number of concepts in the learning object.

Values for a coefficient of the relation type T are determined as following:

6 – for a linguistic relation between classes,
5 – for linking phrases “is a” or “is part of”,
4 – for the linking phrase “is instance of”,
3 – for a linguistic relation between instances,
2 – for linking phrases “has property” and “has value” with classes,
1 – for linking phrases “has property” and “has value” with instances.

A small example of concept map together with identified subgraphs for learning objects is shown in the Fig. **6** and the Table 2 contains calculus of the degree of generality for it.

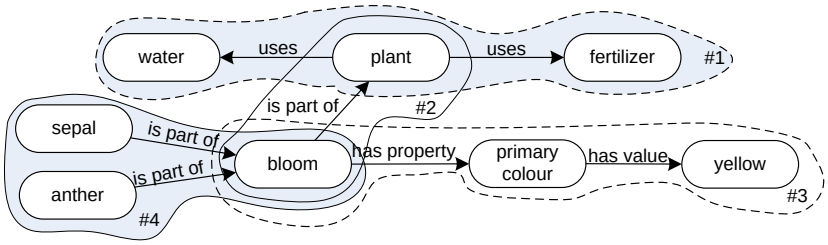


Fig. 6. A concept map with subgraphs for learning objects

Table 2. A calculus of the degree of generality for learning objects from the Fig. **6**

Learning object	Number of root concepts (R)	Coefficient of relation type (T)	Local degree (L)	Number of concepts (N)	Result
#1	3	6	2	3	(3, 9) →1
#2	1	5	1	2	(1, 8) →2
#3	0	2	2	3	(0, 7) →4
#4	0	5	2	3	(0, 10) →3

The highest rank receives a learning object which has the greatest number of root concepts. In case if there are several learning objects with the same number then the second component is taken into consideration. Full learning path generated using the proposed algorithm and concept map shown in the Fig. **6** is depicted in the Fig**. 7**.

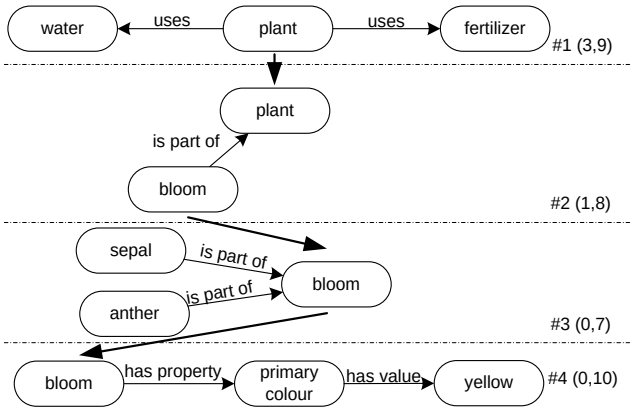


Fig. 7. A learning path for concept map from the Fig. 6

- **The algorithm for generation of personalized learning path**

When a student has submitted his/her concept map, the IKAS system evaluates it and according to detected errors generates a personalized learning path. To perform it the algorithm takes into account erroneous propositions (concept-link-concept). Based on previously generated learning path, the algorithm finds the longest subpath which contains the learning object needed for remedial action. If there are no paths, i.e., needed learning objects are isolated then learning objects are delivered to a student separately beginning with the most general. If between two deliverable learning objects exist one whose content is determined as mastered it also has been added to the personalized learning path. For example, if a student has the incorrect propositions (anther-has property-primary colour) and (primary colour-has property-yellow). The system delivers to the student learning materials #4 and #3. Learning objects included in the personalized learning path allow automatically generate tasks for knowledge remediation.

Method for automated knowledge assessment

This method is based on conception of relations replacement network (RRN) which contains information about possible replacement of one linking phrase with another semantically equivalent linking phrase.

Necessity of semantic analysis when assessing concept maps

The main aim of this research is to increase the autonomy of a concept map (CM) assessment system so that it is able to assess knowledge adequately and minimises the workload of a tutor. Current knowledge assessment systems that utilize CMs as a form for representing knowledge and assess the quality of propositions in an automated way usually employ an expert's map – a CM created by one or several experts and/or tutors. Linking phrases that explain the nature of a relationship between concepts are expressed in natural language and several phrases can be used to express the same meaning. The problem is that an expert's map does not contain all the possible linking phrases that could possibly be used by students. Current CM-based knowledge assessment systems perform simple matching of linking phrases, and if the student's chosen linking phrase does not exactly match the one used in expert's map, it is considered erroneous. No expert's map is omniscient and it contains only the subset of all knowledge in the domain (see Figure 8). Therefore it is possible for a student to demonstrate some correct knowledge that was not included in the expert's map and thus is falsely classified as erroneous when it is not.

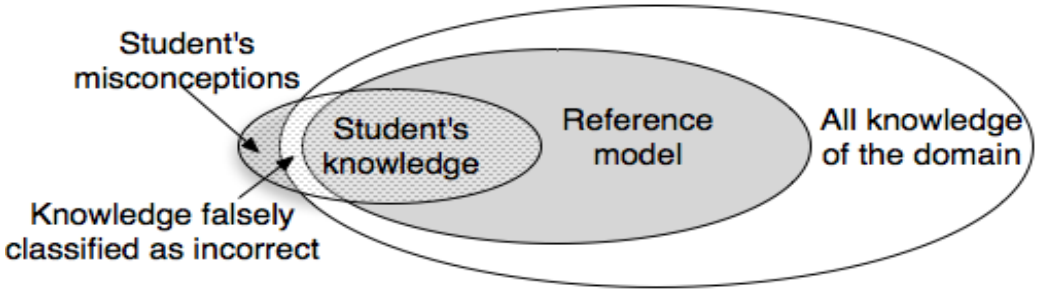


Fig. 8. The relationship between student's knowledge, reference model and all correct knowledge of the field

One of the possible solutions for this problem could be using such assessment criteria that do not involve analysis of linking phrases. However previous studies (see, for example [1]) prove that analysis of linking phrases allow evaluating such aspects of a student's knowledge that cannot be assessed otherwise. Most research on using CMs as a knowledge assessment tool also suggest including the quality of a proposition (which is made up of two concepts and a relationship between them) among other assessment criteria (more on CM assessment methods and criteria see [2]).

Creation of such system that could reliably assess the quality of propositions could be quite difficult and most likely would include some natural language analysis tools. There are several attempts to overcome the multiplicity of possible linking phrases by adding synonyms to the expert's map in systems like IKAS [3], HPKMT [4] and C-TOOLS [5] or adding alternative relationships like in [6]. This approach could work just fine if the variety of possible correct linking phrases was small. Unfortunately for creators of automated knowledge assessment systems it is said that each individual's knowledge structure contains at least some unique components [7], [8], thus the variety of linking phrases could be rather high.

During the development of the catalogue of linking phrases (more on the catalogue and its possible use see in [9]) it was revealed that the same meaning was indeed expressed using wide variety of linking phrases. For example, to describe the relationship between the whole and its part more than 55 phrases were used. For this reason adding all possible valid linking phrases to the expert's map in advance could be impossible. To my knowledge there are also no publications that would report the results of use of this approach.

The previous research did not give enough proof that CMs on the same topic created by students that attended the same course taught by the same tutor would also use wide variety of linking phrases so a study was carried out and the results were presented in a paper (see [10]) that was presented in 58th International Scientific Conference of Riga Technical University. 186 CMs on topic "Fundamentals of artificial intelligence" were analyzed to determine how often students use linking phrases that are different from the ones used in expert's map (a CM that is constructed by a teacher or a domain expert) and how wide is the variety of used phrases. Every student's CM contained on average 34% propositions with different linking phrase than the one used in the expert's map, 38% propositions that connected concepts that were not connected in expert's map and only for about 21% of propositions students used exactly the same linking phrase as expert. In the second part of the study I took a closer look at three of the propositions and it was revealed that the variety of linking phrases is rather wide and only 4% to 25% used linking phrases that matched with expert's used ones while 18% to 39% of propositions were labeled with a different linking phrase that had a semantically close meaning. Results of this study prove that current automated CM assessment systems would not able to adequately assess a rather large part of students' CMs and it is necessary to create mechanism that could analyze the semantics of linking phrases in an automated way.

Relationship replacement network

Propositions are sentence like structures thus at first it was hypothesized that existing natural language processing tools could be used to analyze them. The problem is that in some languages simple combination of first concepts label, linking phrase and second concepts label make grammatically incorrect structures thus the analysis of parts of the speech would be encumbered. For this reason, an alternative solution is proposed- relations replacement network (RRN) that contains linking phrases that can replace or be replaced by other phrases with a certain belief.

To build the RRN first 186 CMs used in the study were digitalized using the CMapTools (https://cmap.ihmc.us) and converted to CSV (comma separated values) format for automated processing. After that the program code was written to compile the list of unique propositions. The resulting array contained 2176 propositions. To each of these propositions an evaluation was assigned so the result of this stage was an array where each element contains 4 components: the label of the first concept, the linking phrase, the label of the second concept, and the evaluation of this proposition.

The RRN is also stored as an array and each element in this array consists of 5 components:

- the linking phrase used by student (replacement);
- the linking phrase used by an expert (the one being replaced);
- the belief – number between 0 and 1 that is an numeric estimation of how likely it is that the replacement phrase can be used instead of the one used by the expert;

- the number of examples – number of propositions where the first element was replaced with the second;
- the number of correct examples – number of propositions where the proposition with a replacement phrase was evaluated as correct.

A schematic representation of the algorithm for building RRN is presented in Figure 9. This algorithm was implemented and the preliminary structure of an RRN was built. Currently the belief is being calculated as a division of the correct examples by the number of all examples. A fragment of the result (in Latvian) in an array form is presented in Figure 10.

RRN can be visualized as a graph where each node is a linking phrase and oriented edges between nodes show which phrases can be used as a substitutes of other phrases. Each of the edges has a number attached which shows how likely it is that a proposition with the substitute phrase also represents correct knowledge in the domain. A fragment of RRN (in Latvian) represented as graph can be seen in Figure 11.

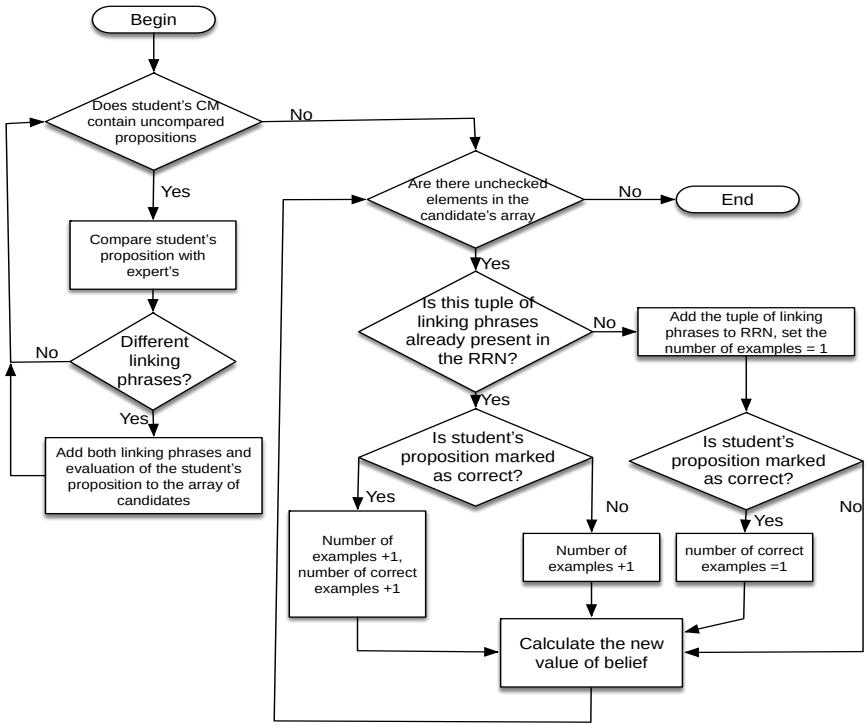


Fig. 9. Algorithm for building RRN

```
ir daļa no      veic    1.0     3       3
ir veids, kā radīt      rada    1.0     1       1
ir piemērs      nodrošina       0,86    7       6
ir piemērs      ir veids        1.0     31      31
ir daļa no      atspoguļo       0       2       0
ir veids        veic    0       6       0
ir piemērs      ir apakškopa    1.0     194     194
ir īpašība      ir apakškopa    0       18      0
```

Fig. 10. Fragment of the preliminary RRN

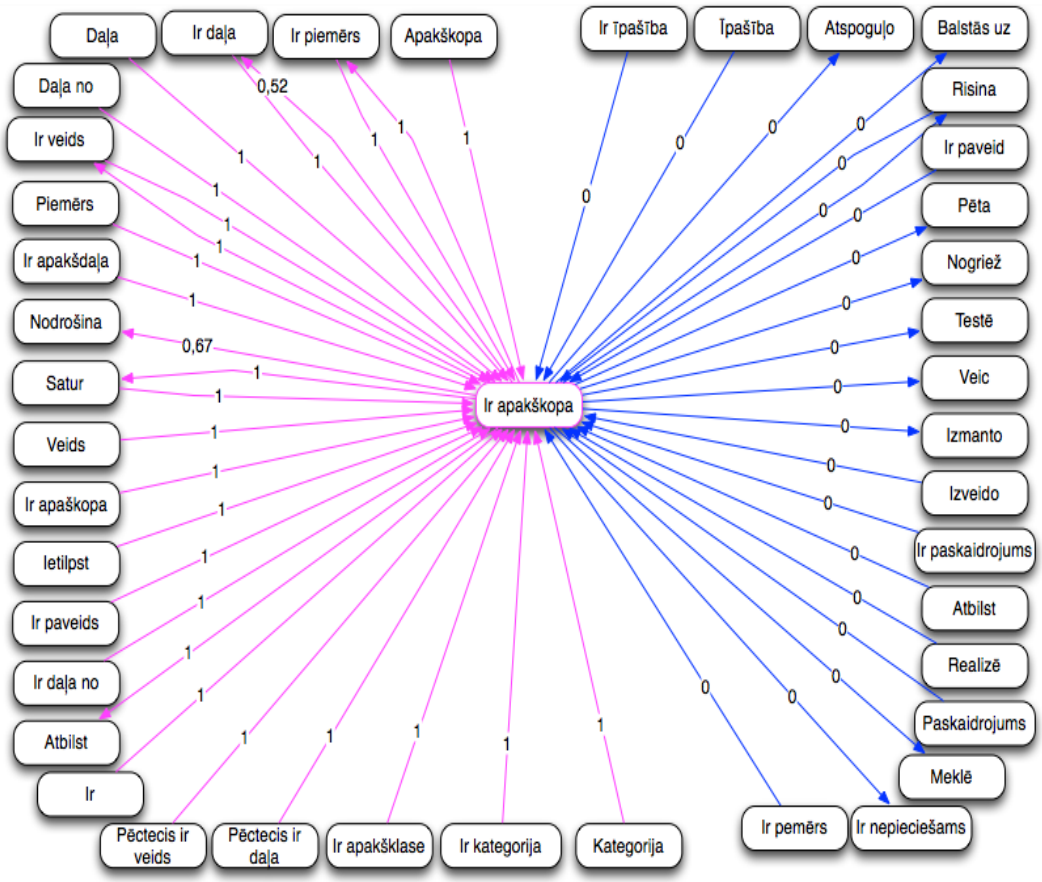


Fig. 11. A fragment of the preliminary RRN represented as a gap

Conclusions and further research

The usage of RRN for knowledge assessment seems promising because, as was proved conducting the study on necessity of semantic analysis when evaluating knowledge represented as CMs, on average 34% of students created propositions use different linking phrase than the one in the expert's map. And almost 40% out of these phrases are with a semantically close meaning. Presumably these propositions form the major part of propositions that are falsely classified as incorrect by current CM-based knowledge assessment systems.

The augmentation of an expert's CM with synonyms for linking phrases does not solve situations when different linking phrases are created due to mistyping or usage of grammatically incorrect forms. Besides synonyms are rarely absolute ones- words in natural language can have multiple, sometimes quite distinct meanings. Also words that can be used as synonyms in one domain may not be acceptable substitutes in another domain. The process of obtaining synonyms for linking phrases would be complicated also due to the fact that linking phrase is often more than a single word. These flaws could be overcome by using RRN.

At the same time it is clear that RRN will not solve all the problems in CM-based knowledge assessment. The usage of RRN has following limitations:

- RRNs could be domain- dependent which means that a separate RRN should be created to assess knowledge in different domains;
- RRNS are natural language dependent because as words can have multiple meanings in most cases it is not possible to find a translation that would cover them all;

- Rather large set of CMs is necessary to build a RRN;
- RRN cannot with a 100% certainty determine which are suitable replacements of a particular linking phrase and which are not, the prediction is more or less precise depending on the examples that were provided for building RRN.

Building the preliminary structure of RRN is only the first step towards its usage for knowledge assessment purposes, additional examples have to be supplemented whenever possible to increase the accuracy of classification. RRN has been built but it has yet to be tested in real knowledge assessment situations. Also different formulas could be used for calculating the belief of pairs of linking phrases. More details about these results can be found in [10].

References

1. Strautmane, M. (2014) Usage of Semantics as a Part of Links as the Basis for Learner's Knowledge Structure Assessment. The Pros and Cons. In proceedings of ICEE/ICIT 2014.
2. Strautmane M. Concept Map-Based Knowledge Assessment Tasks and Their Scoring Criteria: an Overview // Concept Maps: Theory, Methodology, Technology: Proceedings of the Fifth International Conference on Concept Mapping, Malta, Valletta, 17.-20. septembris, 2012. - 80.-89. lpp.
3. Anohina-Naumeca, A., Grundspenkis, J. (2012), Concept maps as the tool for extended support of intelligent knowledge assessment. In Concept Maps: Theory, Methodology, Technology. Proceedings of the Fifth International Conference on Concept Mapping, A. J. Canas, J. D. Novak, J. Vanhear, Eds. September 17-20, 2012, Valleta, Malta, Vol.3., pp.57-60.
4. Chung, G.K. W. K., Baker, E.L., Brill, D.G., Sinha, R., Saadat, F., Bewley, W.L. (2006) Automated Assessment of Domain Knowledge With Online Knowledge Mapping. CSE Technical Report 692, http://www.cse.ucla.edu/products/reports/r692.pdf
5. Harrison, S. H. J. L. Wallace, D. Ebert-May, D. B. Luckie. C-TOOLS Automated Grading for Online Concept MAps Works Well with a Little Help from "Wordnet". Proceedings of the 1st International Conference on Concept Mapping. A. J. Canas, J. D. Novak, F. M. Gonzalez, Eds. September 14- 17, 2004, Pamplona, Spain.
6. Dennis, R.A., Stamboltsian. V., Huang K. (2002) The Mind-Matrix: A Learning, Instruction, and Assessment System. Proceedings of the 2002 American Society for Engineering Education Annual Conference & Exposition, http://www.mind-matrix.org/public/Mind-Matrix-asee.pdf
7. Schau, C., Mattern, N., Weber, R.J., Minnick, K., Witt, C. (1997). Use of fill-in concept maps to assess middle school students connected understanding of science. Paper presented at the Annual Meeting of the American Educational Research Association. https://eric.ed.gov/?id=ED408200
8. Calafate, C.T., Cano, J.-C., Manzoni P. (2009) Improving the Evaluation of Concept Maps: a Step-by-step Analysis. Proceedings of EAEEIE Annual Conference, pp. 1-6.
9. Strautmane M. Increasing the Flexibility of Automated Concept Map Based Knowledge Assessment, Proceedings of the 15th International Conference on Computer Systems and Technologies, Bulgaria, Ruse, 27-27 June, 2014. New York: ACM, 2014, pp. 349-356.
10. Strautmane, M. (2017). On Necessity of Semantic Analysis when Evaluating Concept Maps. Applied Computer Systems, Vol. 21, pp. 52-60.

Conclusion

Practical approbation of developed methods and algorithms as well as the intelligent structural modelling tool I4S showed that analysis of properties of knowledge structures can help to ensure personalized adaptive teaching and learning process. From this point of view the obtained results may be considered as significant step towards development of really autonomous intelligent tutoring and assessment systems. It is foreseen that such type of systems will integrate semantic web, multi-agent and personal knowledge management techniques and technologies. Of course, a lot of work must be done to get the solution when autonomous intelligent tutoring and knowledge assessment system will be able after monitoring an individual learner's successes and failures to search semantic web for needed knowledge and process it using problem domain ontology for construction of individual learning materials and tasks.

2.2.2. Development of methods for the knowledge structures combination and separation, as well as the development of methods for the models' transformation and for the analysis of models' syntax, semantics, and structure

In 2014 during the research on knowledge structures used in intelligent agents, a mechanism for introduction of knowledge structure changes was developed. This includes a general conceptual mechanism for introduction of changes, an ontology for experimental purposes, and a mechanism for adapting existing rules to new/ incomplete data. For achievement of research goals and providing experiments, a prototype of room cleaning multirobot system simulator, which supports manual introduction of changes of knowledge structure into ontologies and rule bases of agents, was developed. Also some research was performed to carry out comparative analysis of knowledge structures used in knowledge bases and deductive data bases. A method for identifying the source of knowledge change in pedagogical model (part of the pedagogical module of an intelligent tutoring system) was developed. This method allows to choose and to adapt tutoring strategies more precisely.

In 2015 a conceptual approach for learning and knowledge structure update was developed to enable changes in knowledge structures based on both user inputs and the results of machine learning. A rule-based learning approach was added to the previously created ontology-based knowledge structure, and a concept of a multi-agent system (MAS) management tool was introduced. A basic functionality of the tool was implemented, namely, the definition of the environment and ontology, as well as the configuration of the MAS [1]. Additionally, the architecture of agent-based affective tutoring systems was proposed that involves emotion ontology sharing among agents simulating human-tutors and students for the evaluation of tutoring strategies adaptation [2].

In 2016 the development of autonomous multi-agent system knowledge structure update mechanism was continued. The previously developed conceptual framework was implemented in an ontology- and rule-based multi-agent system management tool [3]. The tool implements a simplified simulation of a multi robot system for cleaning tasks. The previously developed rule-based priority mechanism was implemented in the tool and was applied to the task allocation in multi-robot systems (currently working in a simulated environment). The functionality of the tool at this stage was the following: defining and updating the knowledge structure (ontology),

configuration of the environment, configuration of the multi-robot system as well as the knowledge base.

Research related to the development of knowledge structure for dynamic adaptation of emotion-based instructional process was carried out to continue the studies started in the previous year. A multi-agent system was selected as a basic approach to enable simulation of affective student-tutor interaction. Architecture of a pedagogical agent was designed supporting not only the usage of pedagogical knowledge (e.g., tutoring strategies, including game-based learning) but also emotion ontology, which includes both the descriptions of emotions and their possible causes. The communication in the multi-agent system is enabled by using fragments of previously researched ontologies. The knowledge of personality is included in the model by transforming it into core mood, thus ensuring communication of emotional state among agents. Furthermore, the reasoning mechanism of a pedagogical agent utilizes the knowledge about student's personality, which serves for various purposes, e.g., for prediction of student's emotions and behavior, for generation of appropriate tutor's personality and teaching actions, as well as for selection of suitable teaching methods [4].

In 2017 the research focused on rule- and ontology-based adaptation in autonomous multi-robot systems, as well as on game-based adaptive education and affective interaction based on emotion ontology was continued. The results of these studies are described in the subsections 2.2.2.1–2.2.2.3.

2.2.2.1. Development of adaptive multi-agent system and validation in virtual environment based on previously developed toolset and learning mechanism

Architecture of an adaptive multi-agent system, a toolset consisting of Ontology Learning Tool and Multi-Agent System Management (MAS) Tool, as well as machine learning algorithm for learning to match capabilities with corresponding tasks were developed during the first stages of the project. The task of the fourth stage of the project was to validate the toolset and learning algorithm by developing a MAS prototype. The developed prototype has been implemented in a virtual environment and simulates a heterogeneous multi-robot system for cleaning tasks. The prototype validates the toolset and learning algorithm.

The prototype of a multi-agent system has been developed for cleaning domain. It has been implemented in a virtual environment where agents simulate actions done by cleaning robots. The user interface gives information to the user in terms of actions done by each robot. The cleaning robots are equipped with different cleaning methods. The task solved in terms of the project is to dynamically find the most appropriate performer for each task based on the appropriateness of each capability (in the particular example, cleaning method) to a particular task (cleaning particular object). This task is depicted in Fig. 2.2.2.1.

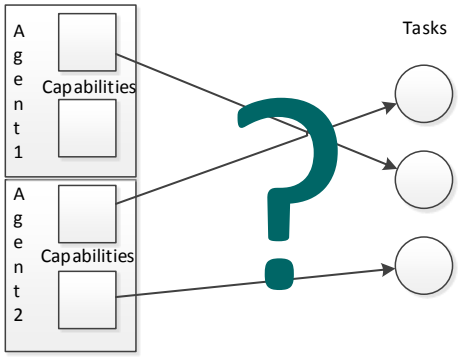


Fig. 2.2.2.1. Learning task for multi-agent system's prototype

The prototype was developed in the previously developed toolset. The system consists of the following agents:

- Sweeper with method Sweeping;
- VacuumCleaner with methods VacuumCleaning withBrush and VacuumCleaningWithoutBrush;
- WetCleaner with methods CleaningWithFoam, HighPressureWashing and CleaningWithPowder;
- SteamCleaner with method SteamCleaning;
- Mopper with methods WetMopping ans MoistMopping.

Screenshot with the MAS management tool with the details of the developed prototype is given in Fig. 2.2.2.2.

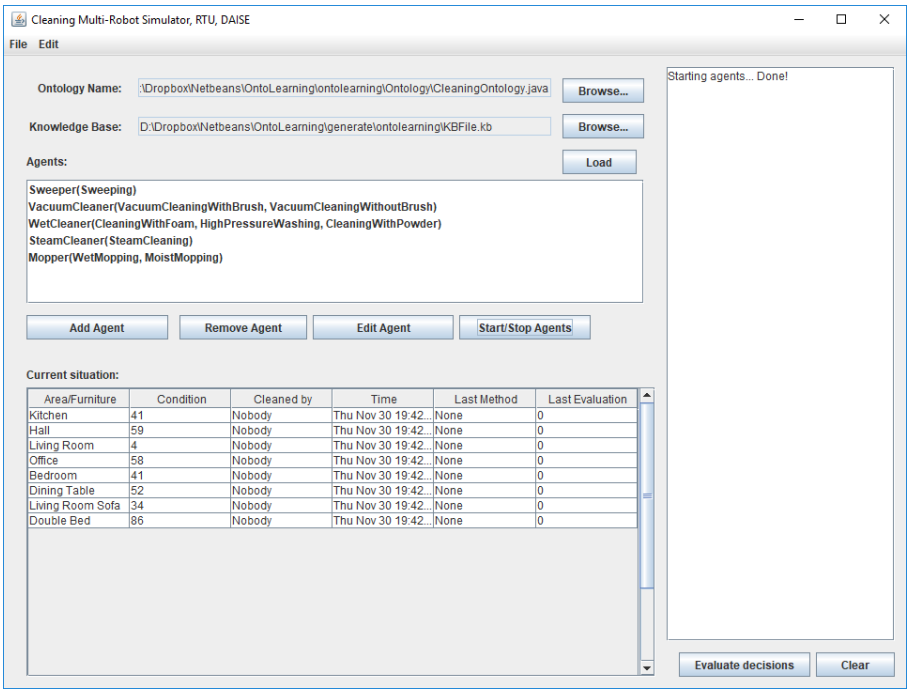


Fig. 2.2.2.2. The prototype of the multi-agent system in the MAS management tool

After deployment of the prototype, it is capable to find a cleaning robot that can clean any of the objects. Still the resulting allocation is far from optimal, because during the design phase the set of the rules was on purpose designed to be very limited. Therefore, there is a need for learning.

The learning method previously implemented in the MAS management tool is used. The system starts operating based on the set of rules defined by the knowledge engineer in the Ontology Learning Tool and stored in the rule base. After completing every action the user may give a feedback about its result. The feedback is collected in the example set in the traditional form for the inductive learning, namely the values of the attributes (particular action and capability used to do it and their classes) and the evaluation. The evaluation is provided in the scale from (−3 as very bad to +3 as the best possible choice; 0 is used in case no feedback is available).

One of the purposes of implementing the MAS prototype was to evaluate the previously developed learning mechanism. To do so, the following approach was used:

1. Deploy the system;
2. Let it operate without giving any feedback until 10 examples are collected;
3. Evaluate the examples collected in Step 2;
4. Calculate the average evaluation and plot it in the learning curve;
5. Return to Step 2 and start over by collecting 10 new examples.

The learning curve obtained as a result of the validation process is given in Fig. 2.2.2.3. It shows how the average evaluation changes based on collecting new examples and using their evaluations to improve the decision making process. The curve does not get to the theoretical maximum (in our case perfect evaluation is +3) due to the fact that for some objects there are no performers that can do it with the highest posisble evaluation.

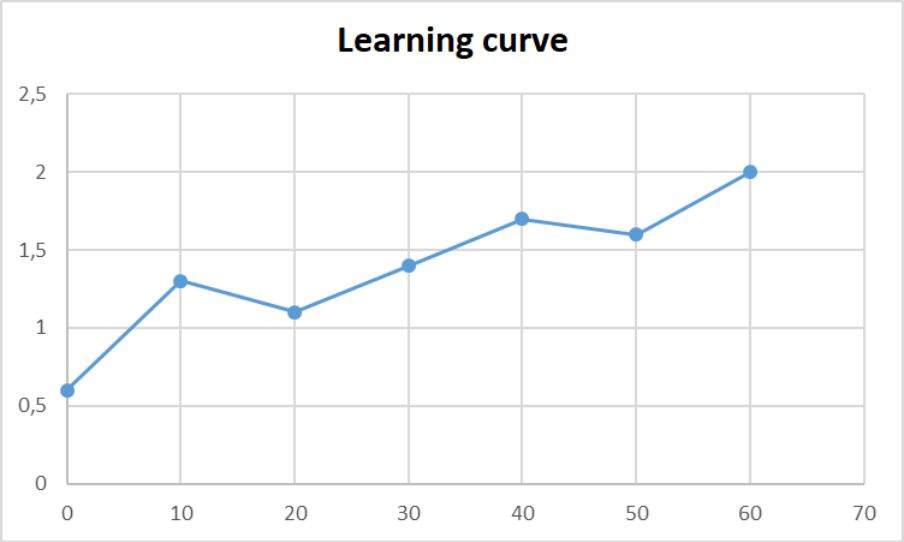


Fig. 2.2.2.3. Learning curve – average evaluation depending on the number of examples

The learning curve proves that the learning process is successful. If the system were working without learning, the average evaluation would be 0.6 due to the fact that the initial rule set created during the design process is way too small and too general. Based on the examples and feedback, the average evaluation grows rapidly. After 30-50 examples there are 2 objects (out of 7 total) for which there is no good solution found yet. The other objects have an appropriate solution already after 20 examples, while for these two objects appropriate solution is found only after 60 examples or in iteration 7. Minor oscillations in the average evaluation are explained by two factors. First, random factor – the average evaluation depends on the requests for particular

tasks that must be cleaned by the robots. Some of them already have an appropriate performer, while for others it is sought. Secondly, in some cases at the moment when the request comes, the most appropriate performer is already busy with some other tasks. In such case the system will try some other agent/cleaning method which may lead to lower evaluations.

One should note that the average evaluation does not reach the theoretical maximum of +3 because not all objects have a corresponding cleaning method with evaluation of exactly 3.

If the requests would come sequentially one by one without any tasks coming concurrently, then once an appropriate method for a particular task is found, it would always be used for the particular object. Still, in practical setups, the request can come also during the execution of any task. In that case the system may decide to try new solutions. If such situations are often enough then there is no need for any additional mechanisms to explore unseen solutions, because local maxima will be avoided based on such trials. Otherwise there might be a need to introduce some kind of randomizer that would lead away from local maxima by trying new and possibly better solutions. In case of introducing a new method this problem can be solved by artificially increasing its evaluation for some period to get it tried.

Further details about ontology, multi-agent system and the environment where it is working, as well as details about the collected data and calculations are available in the technical report: Lavendelis, E. Prototype of adaptive multi-agent system. Technical report, Riga Technical University, 2017.

2.2.2.2. Development of behaviour and interaction method based on pedagogical knowledge and emotion ontology

The popularity of digital games has drawn researchers' attention to the development of game-based educational settings to enhance learner's knowledge, skills, and other characteristics. The reasons behind this interest are different, e.g., games increase enjoyment, involvement, and motivation, as well as they influence emotions. Playing games is an emotional process and, at the same time, emotions are an important factor for successful learning. However, learners usually differ regarding their interests, preferences, and personality. All of the mentioned aspects could be analysed to provide adaptation in learning environments. Educational games like other learning environments have a potential to adapt to characteristics of an individual player by matching together game elements to player's actions, performance and emotions. However, the analysis of recent studies still proves that learners' differences are seldom taken into account in educational games, and emotions as an adaptation source are used even rarer. For this purpose, general ideas from intelligent tutoring systems are adopted to ensure game adaptation. In general, instructional adaptation can be provided at two levels:

1. at macro-level that is realized prior to learning based on static player's data available to system (or game) before learning (gameplay);
2. at micro-level that is ensured during the learning/teaching process based on dynamic player's data acquired in a real-time during the gameplay, e.g., based on responses, actions, emotions, etc.

Macro-level Adaptation

To provide the macro-level adaptation, before the game starts, the learner's prior knowledge is considered in the combination with a learner's personality represented

as Big Five personality traits. The personality traits give information not only about the learner's personality but also about his/her default mood or tendency to some specific emotions, preferred learning style, teacher's type, and teaching approach, as well as they allow identifying tendency towards goal achievements. Overall, the macro-level adaptation process is represented in Fig. 2.2.2.4.

To ensure adaptation at the macro-level, personality inventory is offered to the learner if he/she enters the system (game-based environment) for the first time. The system calculates personality traits based on learner's answers and feeds this data to two neural network models, which assign learning style and achievement goal for the learner. For example, if personality traits are already identified (see Fig. 2.2.2.5) then using models corresponding learning style and achievement goal can be identified. Both models were developed on the basis of collected surveys on learners' personalities, learning styles, and achievement goals. After the macro-level adaptation, the game initiates a micro-level adaptation.

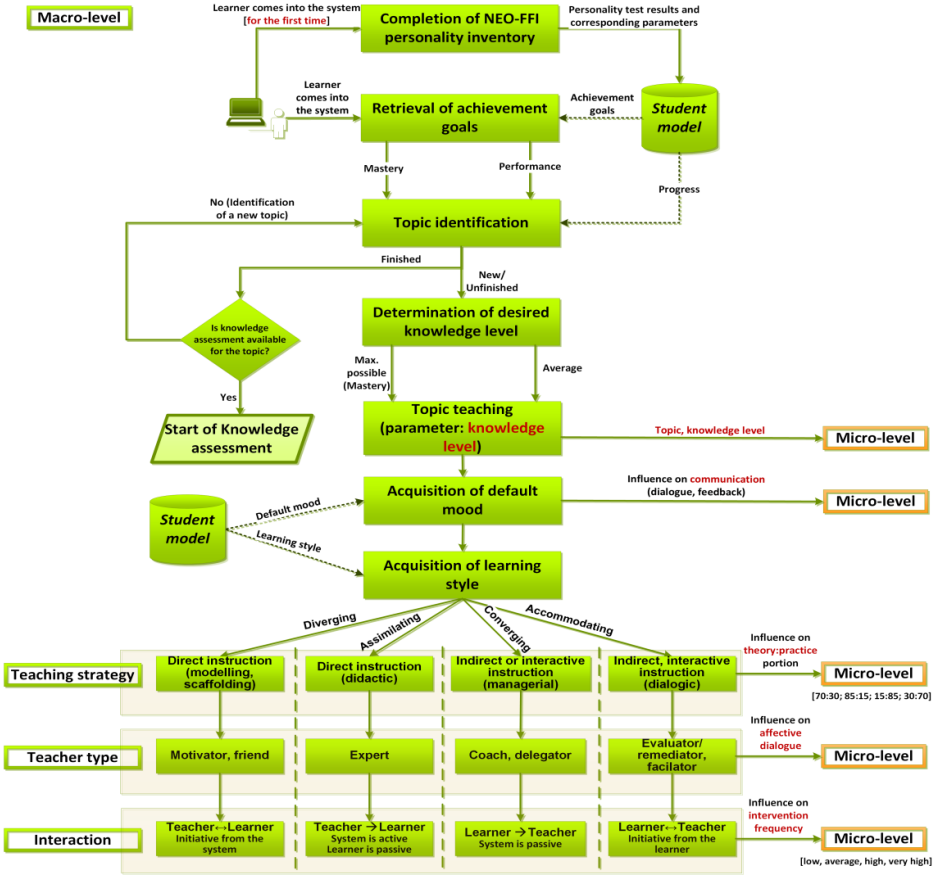


Fig. 2.2.2.4. Adaptation at the macro-level

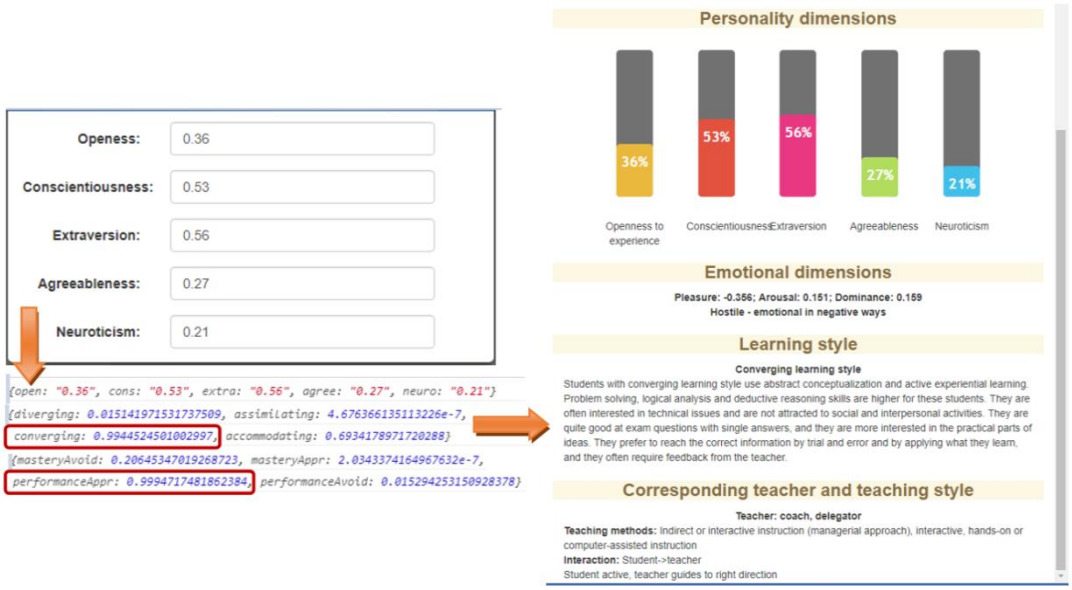


Fig. 2.2.2.5. Identification of learning style and achievement goal based on personality

Micro-level Adaptation

As a gameplay starts, the micro-level adaptation is implemented on the basis of already stored static parameters (personality, learning style, teacher's type, and achievement goals) and dynamic parameters. Overall, four different types of teachers (Friend, Expert, Coach, and Remediator/Evaluator) are integrated in this approach corresponding to the particular learning style of a student and interacting at the micro-level based on developed reacting rules. This influences teacher's (system's) decisions and behaviour or interventions (frequency, intervention type and content) at the micro-level and the achievement goal set for a learner during the assessment process. Emotions are considered as one of the parameters, since they are occurring and changing during the gameplay. They are identified with an aim of recognizing potentially problematic learning situations and providing timely teacher's interventions in case of negatively valenced emotions or low attention level. Adaptation approach is provided at all three steps of any learning/teaching process – topic teaching, practice, and knowledge assessment. Example for knowledge assessment and developed reacting rules (*for teacher-friend*) are demonstrated in Fig. 2.2.2.6. More detailed description of the developed multi-level adaptation approach is available in the scientific publication: Petrovica, S. Multi-level adaptation of an educational game to individual student's gameplay, knowledge and emotions. Proceedings of the 9th Annual International Conference on Education and New Learning Technologies. Barcelona, Spain, July 3-5, 2017, pp. 2220-2230.

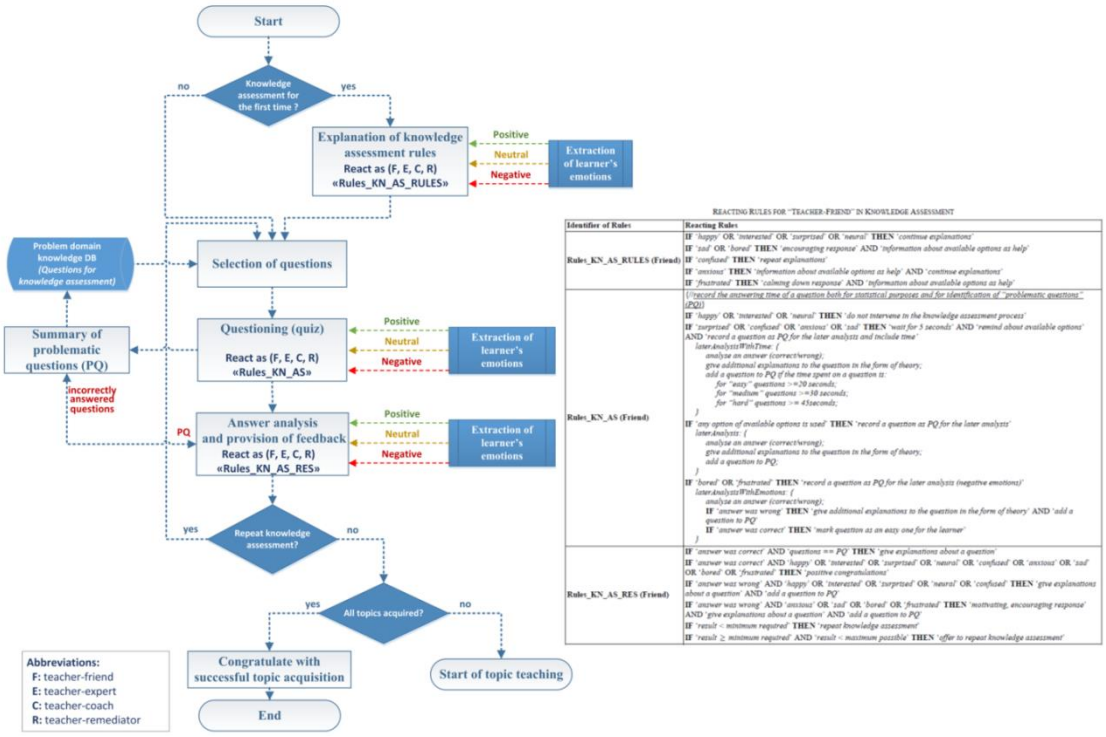


Fig. 2.2.2.6. Activities of knowledge assessment and reacting rules

Currently, the development of a 2D game is carried out as part of an affective tutoring system called "ELIA" (***E**motions for **L**earning and **I**ntelligent **A**ssessment*) created for the study course "Fundamentals of Artificial Intelligence". The game is designed to provide double functionality – learning/practice through topic teaching and game levels (see Fig. 2.2.2.7a) and knowledge assessment (see Fig. 2.2.2.7b).

Game is created with an aim to test the proposed adaptation approach. Currently, the first experiments with the system were conducted for its testing purposes and comparing assessment results of learners, who were using the game, with learners, who had paper-based version of the same test without involving the management of learner's emotions. Next step will be related to the effectiveness evaluation of the proposed approach in terms of learning outcomes (including exam results) and affective outcomes or learners' emotional experiences.

a) b)

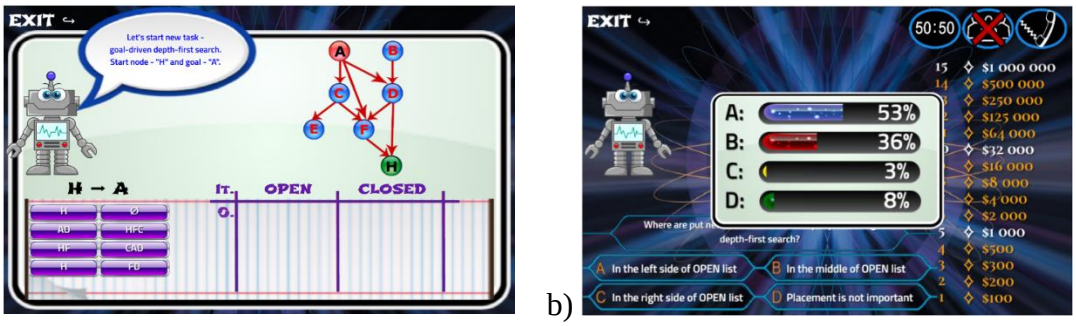


Fig. 2.2.2.7. Game interface for learning/practice activities and knowledge assessment

2.2.2.3. Development of emotion ontology based method for emotion flow modelling between agents

Agent based modelling is a paradigm that is widely used to simulate and research systems where each entity is autonomous and behaves independently. Models that

intend to simulate human group corresponds to these characteristics thus it is convenient and logical to implement human group affective behaviour simulation as an agent based model.

There are several approaches for modelling human group in an affective system, it can be either biological model (agents are biological beings with parameters like energy level), cognitive beings (agents are able to make decisions) or social beings (agents consider other agents' states). It is also possible to have heterogeneous or homogeneous agents. The classes and their compatibilities were used as guidelines for several design decisions: network topology (preferential attachment was chosen), agent type (human as a social being with according parameters) and architecture (layered architecture to match multiple information processing levels). These and other classes and their compatibility is described, as a part of larger and more general classification, in the scientific publication: Pudane, M. Classification of Agent-Based Models from the Perspective of Multi-Agent Systems. Proceedings of the 5th IEEE Workshop on Advances in Information, Electronic and Electrical Engineering AIEEE'2017, Riga, Latvia, November 24-25, 2017. Riga, 2017.

Based on human as a social being, the method for modelling information flow among agents in group that may or may not have common tangible goal was created. Method consists of five types of emotional interaction mechanisms that need to be implemented:

1. direct communication, i.e., highly cognitive process with aim to express the emotions and change other agent's strategy or to follow own social or cognitive strategy; it uses ontology to express emotions;
2. manipulation: also cognitive process that aims to alter others' behaviour or follow own plans, yet the agent that manipulates does not have affective state;
3. primary emotional contagion that results in experiencing the same emotion as the party that has expressed it without cognition;
4. secondary emotional contagion that includes social appraisal (i.e., re-appraising the result of primary contagion depending on social links) and perspective taking (the process of understanding how other person feels);
5. patterns that are needed to invoke complementary states, such as if one feels anger, other person might feel fear.

The mechanisms were then mapped on three level architecture of agents to create formal model of mechanisms that need to be implemented. Since the mechanisms are not the same for the sender and the receiver (e.g., manipulation and direct expression does not differ on the receiver's side; secondary contagion as a phenomenon does not exist on sender's side) both sides were mapped separately.

Further, the mechanisms were detailed to pseudocode and implemented in a game "Snakes and Ladders" by using multi-agent system framework JADE. More detailed information on the method, its implementation and results can be found in the scientific publication: Pudane, M. Affective Multi-Agent System for Simulating Mechanisms of Social Effects of Emotions. Proceedings of Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW), San Antonio, United States of America, October 23-26, 2017. San Antonio, 2017, pp. 129-134.

References

1. Lavendelis, E. A Conceptual Approach for Knowledge Structure Update and Learning in Multi-Agent Systems. International Journal of Computers, Vol. 1, 2016, pp. 141-147.
2. Petrovica, S., Pudane, M. Simulation of Affective Student-Tutor Interaction for Affective Tutoring Systems: Design of Knowledge Structure. International Journal of Education and Learning Systems, Vol. 1, 2016, pp. 99-108.
3. Lavendelis, E. A Cloud Based Knowledge Structure Update and Machine Learning Framework for Heterogeneous Multi-Agent Systems. International Journal of Artificial Intelligence, Vol. 14, No. 2, 2016, pp. 157-170.
4. Petrovica, S., Pudane, M. Emotion Modeling for Simulation of Affective Student-Tutor Interaction: Personality Matching. International Journal of Education and Information Technologies, Vol. 10, 2016, pp. 159-167.

2.2.3. Development of approaches and methods for the control of models, processes, enterprise architectures, and knowledge structural compatibility

In 2014 several new theoretical results were achieved while analyzing the role of time dimension and knowledge structures that describe it, which allow introducing time dimension in enterprise architectures and their management. New theoretical results have been reached, including extended and enriched time model which could be linked with enterprise architectures and Bunge ontology. During studies on processes, important links that are necessary for providing continuous requirement engineering have been identified, which is an important practical result for software development.

In 2015 on the basis of research done in the previous year, understanding of knowledge structure and process compatibility was extended by the analysis of related works where four compatibility criteria were identified. While these criteria concern both tacit and explicit knowledge, in this period the main focus was on explicit knowledge, when working on the compatibility identification approach that concerns the identified criteria and well known workflow data patterns. For moving towards compatibility identification methods, the FREEDOM framework was developed that relates operational business processes to its development and management functions, thus forming the basis for requirements engineering for continuous process development. The main requirements problems in multi-project environment were analyzed and an approach for their handling was developed in order to enable continuous requirements engineering. In the context of operational business processes, possibilities to reflect knowledge about business object states were investigated. A method for document compatibility analysis was developed and approbated in educational processes. Experiments with identification of security requirements patterns as a knowledge structure and business process compatibility, as well as possibilities to introduce continuous information security audit were done in cooperation with a parallel international project, thus gaining research synergy and promoting further collaboration with Tartu University. As the FREEDOM framework was intended to be related to the integrated model of business processes and enterprise architecture based on Bunge, Wand, and Weber information systems ontology that was developed under the Latvian Council of Science grant No. 342/2012, a conceptual connection was made between the integrated model and the time dimension, which in related works and in our research was recognized as an important factor in knowledge structure and process compatibility.

In 2016 an approach for aligning requirements engineering based knowledge flows (structures)/information artifacts was proposed, based on the continuous requirements/systems engineering framework FREEDOM that was developed in the previous year. The details of the framework that influence the knowledge/information flow were addressed; in parallel the FREEDOM framework was compared to other frameworks developed for similar purposes, and the variants of the FREEDOM framework were identified and illustrated to demonstrate its flexibility.

Aligning requirements/systems engineering based knowledge flows (structures)/ information artifacts requires addressing the existing gap between business process models and states of business objects. Therefore an approach was developed for explicit definition of states of business objects, automatic generation of conceivable state space at a process model design-time, automatic generation of lawful state space, and compliance checking at a process run-time. In this regard, time aspects play an important role; therefore, in this phase of the project, also time aspects were analyzed with respect to the FREEDOM framework. Additionally an approach for identification of system's external knowledge structures/information artefacts and the changes of their state was proposed. The approach is rooted in the analysis and representation of the structure and content of the documents capturing valuable knowledge for the enterprises. Taking into consideration that large amount of requirements are elicited and analyzed during information systems development process and that those requirements can be processed, stored, and managed in various tools and communicated via various channels, the appropriate requirement distribution approach was developed. To establish the background for unification of abovementioned aspects of artifact alignment in the next phase of the project, the possibility to use graph algorithms in information flow analysis was analyzed.

In 2017 studies of various aspects of requirements engineering were continued based on the results obtained in the preceding stages of the project. Within the developed FREEDOM framework, the researchers studied representation of time intervals and moments, handling of external information sources, and representation of artefact flows using graphs. During this stage of the project, a method for maintenance and distribution of requirements engineering knowledge and artefacts was developed (see Subsection 2.2.3.6), which was one of the stated goals of the Activity A1.8. After the Subsection 2.2.3.1, which contains a short overview of the FREEDOM framework, the Subsections 2.2.3.2–2.2.3.6 describe these scientific results in more detail.

2.2.3.1. Overview of the FREEDOM framework

The research in this topic applies the previously developed FREEDOM framework for establishing the methodics for knowledge/artefact maintenance and distribution during requirements engineering. It assumes that FREEDOM framework can be used as a basis for continuous requirements engineering tool development that can support the methodics.

FREEDOM is a continuous requirements engineering framework, which has emerged as a result of the application of viable systems model in requirements engineering. FREEDOM framework is flexible with respect to the number of its units and links between units, as some of units may be merged together. The framework is represented in Fig. 2.2.3.1.

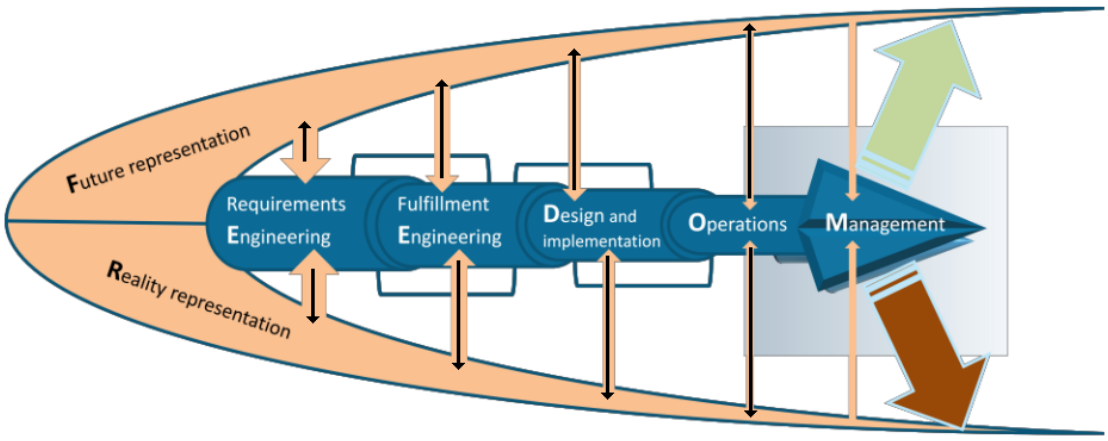


Fig. 2.2.3.1. Constituents of FREEDOM framework

FREEDOM framework has the following constituents: F – Future representation, R – Reality representation, E_1 – Requirements Engineering, E_2 – Fulfillment Engineering, D – Design and implementation, O – Operations, and M – Management. FREEDOM framework constituents have the following functions:

- Future representation F represents vision of the target system in its context (To-be situation). Artifacts that are developed by this function are mainly different enterprise models, enterprise development artifacts, project plans, design documents and other artifacts, that represent and characterize an envisioned future situation. These artifacts may be developed by F, or contributed by other constituents of the FREEDOM framework (in Fig. 2.2.3.1 black and brown arrows);
- Reality representation R represents the present (As-is situation). Types of artifacts are similar to those of F, just information they represent is about current situation. These artifacts may be developed by R, or contributed by other constituents of the FREEDOM framework (in Fig. 2.2.3.1 black and green arrows). Information available in databases, warehouses and other IT systems also may belong to R;
- Requirements Engineering E_1 is the function dedicated to the model and tool based acquisition and management of requirements that can be used by functions on to the right from E_1;
- Fulfillment Engineering E_2 is the function that takes care of handling project portfolios that would lead to the fulfillment of stated requirements. In simpler cases E_2 can be merged with either E_1 or D, as it is common to put design next to requirements engineering;
- Design and implementation D is the function that produces the design and handles implementation of the target system accordingly to fulfillment strategies, methods, chosen lifecycles and guidelines established in E_2;
- Operations O regard the actual operation on the implemented system, including maintenance;
- Management M refers to all levels of management under which the target system operates. The management can influence both the reality and its representation function R and the future vision and its representation function F.

The framework assumes that knowledge/artefacts continuously propagate from E_1 towards O in a managed and transparent way and each function can acquire

information from other functions and provide feedback to other functions. The management can provide direct requests to all other functions. All functions can acquire information from wider environment.

Constituents mentioned above are not monolithic blocks; they are consisting of various functions themselves, for example, Requirements Engineering (E_1) may consist of Requirements acquisition, Requirements analysis, Requirements representation, Requirements validation and Requirements management functions. Other constituents of FREEDOM framework may also be viewed as consisting of smaller functions.

2.2.3.2. Time intervals and moments in the FREEDOM framework

In modeling, time dimension has two main concepts: interval and instant [1]. To obtain the grouping (or taxonomy) for elements of the time dimension of FREEDOM framework these two concepts must be mapped to FREEDOM framework concepts. Table 2.2.3.1 shows that mapping, which represents how time intervals and moments are grouped with respect to the FREEDOM framework.

Table 2.2.3.1. Time dimension and FREEDOM framework concepts' mapping

Time dimension element	**FREEDOM framework element**	**Description**
Interval	E_1	Requirements engineering
	E_2	Fulfillment Engineering
	D	Design and implementation
	O	Operations
	M	Management
	SystemV_n (obtained from Reality representatio)	System version N – current system state
	SystemV_{n+1} model (obtained from Future Representaion)	System version N+1 – artifacts that represents system future states.
Instant	RequirementDefined	Day when first requirement definition or idea about system's new functionality is documented
	RequirementSpecified	Day when requirements specification is completed
	SpecificationDesigned	Day when requirements detailed specification is prepared
	AgreementSigned	Day when development of new functionality starts
	AcceptanceActSigned	Day when development of new functionality ends and usage of new functionality starts

Every interval includes many instants. Two instants *start* and *end* limit the interval. These two instants are described as follows *start(i)* and *end(i)*, where *i* – interval and *end(i)* > *start(i)*. For example, design and implementation step is limited by start moment – *start(D)* and end moment – *end(D)*. Using every interval start and end instants the rules that link different FREEDOM framework time concepts were identified (Table 2.2.3.2) and corresponding ontology developed.

Table 2.2.3.2. Linkage between FREEDOM framework temporal concepts

Rules	**Description**
RequirementDefined ≡ Start(E_1) ≡ Start(SystemV_{n+1} model)	Requirements engineering step start is equivalent to the instant when idea about new functionality is raised and equivalent to system's future representation model interval start.
End(E_1) ≡ RequirementSpecified ≡ Start(E_2)	Requirements engineering interval's end is equivalent to fulfillment engineering interval's start. Ideas, documented during E_1 stage, are taken into consideration and fulfillment requirements are developed.
End(E_2) ≡ SpecificationDesigned ≡ AgreementSigned ≡ Start(D)	Fulfillment engineering ends when specification is ready. Specification is part of agreement. Day when agreement is signed is equivalent to design and implementation interval's start.
End(D) ≡ End (SystemV_{n+1} model) ≡ AcceptanceActSigned ≡ Start(SystemV_{n+1}) ≡ Start(O)	Day when acceptance act is signed means that design and implementation is ended, system's version V_{n+1} future representation or model becomes part of the system. Now this functionality is system's reality and operations stage starts.
End(O) ≡ Start(M)	Operations interval's end is equivalent to management interval's start.
SystemV_{n+1} model ≡ E_1 + E_2 + D	During E_1, E_2 and D steps, artifacts that describe systems future state are prepared. This means that sum of E_1, E_2 and D intervals are equivalent to SystemV_{n+1} model.
SystemV_{n+1} ≡ O + M	During O and M steps, users work with SystemV_{n+1} current (present) version. This means that the sum of the length of operations and management intervals is equal to SystemV_{n+1} current state.

Described ontology was entered in Protégé [2] tool (Fig. 2.2.3.2).

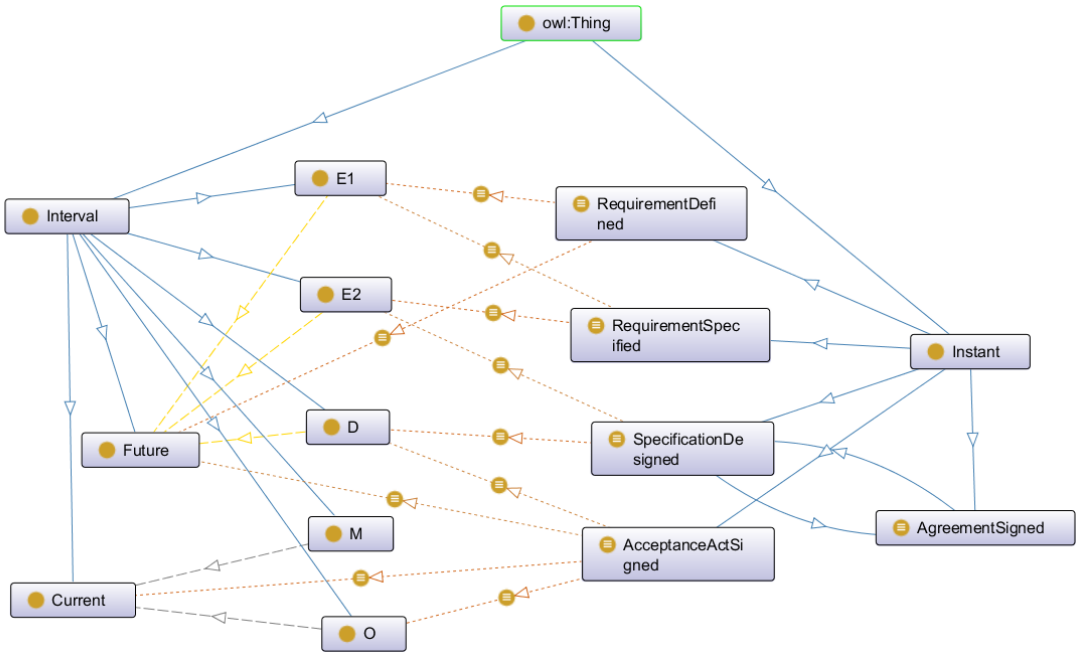


Fig. 2.2.3.2. FREEDOM framework temporal concept ontology

Fig. 2.2.3.3 describes relationship types between ontology classes.

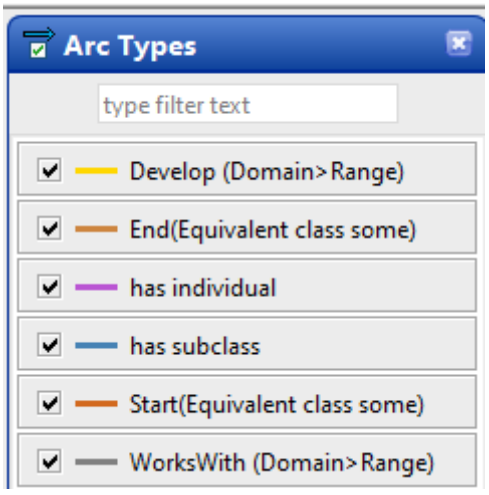


Fig. 2.2.3.3. Relationship types

The ontology shown in Fig. 2.2.3.2 is one possible, generic, option of time interval linkages. It is a matter of further research to see how flexible the time ontology should be to accommodate different variants of FREEDOM framework configuration. Another topic of further research is the extension of developed ontology for inclusion time intervals and instants that are related to knowledge/artifacts external to FREEDOM framework.

2.2.3.3. Requirements engineering artifacts and FREEDOM framework

Requirements engineering knowledge and artifacts were analyzed in general (by considering most popular sources where requirements engineering artefacts are described [23], and by detailed analysis of specific continuous requirements engineering cases, such as continuous requirements engineering for mobile application development [16], DevOps for very small entities [24], [25], continuous

requirements engineering with Scrum [17], and continuous requirements engineering for start-ups [26]. The continuous requirements engineering for mobile application development, continuous requirements engineering with Scrum, and continuous requirements engineering for start-ups were further analyzed and mapped to the fractal version of FREEDOM framework transferred into enterprise architecture representation language Archimate [14]. The analysis showed that the use of FREEDOM framework gives an opportunity to improve continuous requirements engineering processes in all three cases.

2.2.3.4. External information sources relevant to FREEDOM framework

The success of continuous requirements engineering depends not only on the artefacts that are created inside the FREEDOM framework. They depend also on the knowledge/artefacts outside of the framework [3], [4], [5]. External knowledge/artefacts can appear and change with or without notifying about these events the FREEDOM framework's functions. Knowledge about external knowledge/artefacts supports a possibility to be aware of the external environment in requirements engineering.

To identify the relevant groups of external knowledge/artefacts the following method was applied. First the external artefacts for each function of the FREEDOM framework were identified on the basis of related works about these functions. Second the identified knowledge/artefacts were grouped vertically to obtain a generic grouping of external knowledge/artefacts.

Table 2.2.3.3 represents the external artifacts for each function of the FREEDOM framework. Only knowledge/artefacts that are external to the whole FREEDOM framework are considered, i.e. the artifacts that are internal in one FREEDOM framework's function, but (internally in the framework) external to another function are not taken into account.

Table 2.2.3.3. Identified external artifacts

FREEDOM framework's function	**External knowledge/artefacts**
F – Future representation	Enterprise architecture standards [3], Enterprise architecture frameworks [3], Enterprise modeling methods [3], Domain models (ontologies) [3], Information technology advances [3], Quality standards [4], Information on modeling methods and tools [4], Text books [4]
R – Reality representation	Enterprise architecture standards [3], Enterprise modeling methods [3]
E_1 – Requirements Engineering	Software development lifecycle models [3], Competitor information [6], Customer information [6], Market information [6], Product information [6], Supplier information [6], Ontologies [7], Standards [8]
E_2 – fulfillment Engineering	Market information [9], Customer information [13]
D – Design and implementation	Software development lifecycle models [3], New technologies [10], Ontologies [11], Standards [12]

O – Operations	Market information [9], Technology information [9]
M – Management	Competitor information [6], Customer information [6], Market information [6], Product information [6], Supplier information [6]

As seen from Table 2.2.3.3, there are types of external information that are required by several FREEDOM framework functions, e.g. ontology. Assuming that, at least to some extent, each identified external knowledge source can be useful for all FREEDOM functions, the grouping of the external information is made by amalgamating the source types reflected in Table 2.2.3.3.

The grouping of external information, i.e. knowledge/artifacts, by the content of the source is given in Fig. 2.2.3.4. The grouping of external information by its type is shown in Fig. 2.2.3.5.

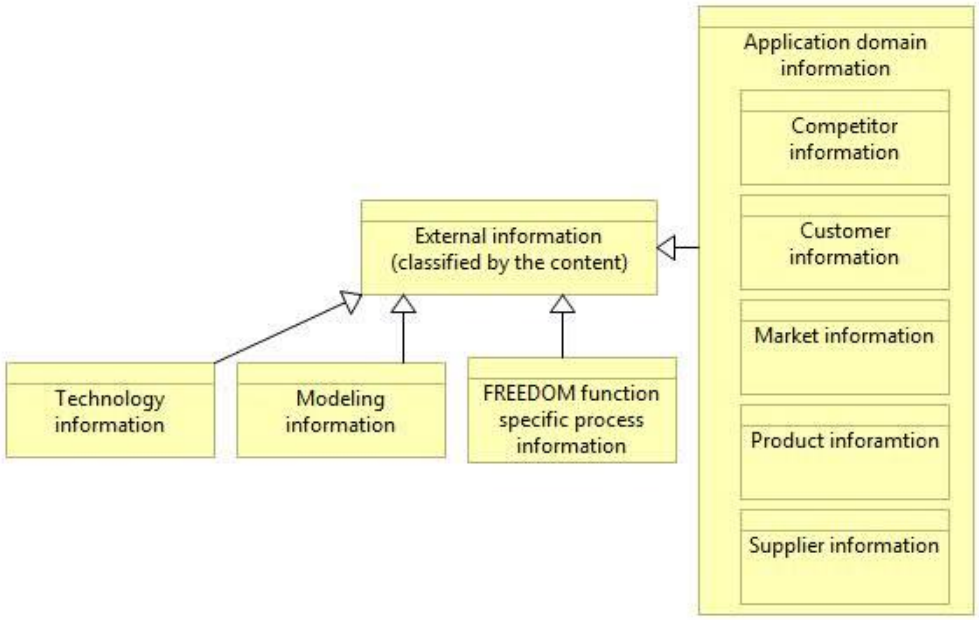


Fig. 2.2.3.4. External information grouped by its content

As seen from Fig. 2.2.3.4, the external information should be gathered for new technology, new modeling approaches, different processes (such as design process, implementation processes, etc.), and different types of domain knowledge.

As seen from Fig. 2.2.3.5, the basic types of external information to be considered are text books and articles; ontologies; frameworks, models, and methods; and standards.

To handle these sources manual methods shall be combined with artificial intelligence tools. The first insight on using analytics methods in requirements engineering has been obtained in this project and is amalgamated in [21]. Development of particular algorithms for handling different types of eternal information is a matter of further research.

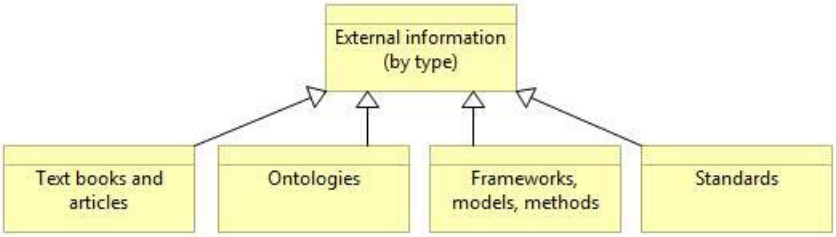


Fig. 2.2.3.5. External information grouped by its type

2.2.3.5. Graph based method for artefact flow representation in FREEDOM framework

The constituents of FREEDOM framework produce various requirements engineering artifacts, which then are used in other functions of the framework. This flow of artifacts may be represented using graphs.

The following method for representation of knowledge/artefact flow in the FREEDOM framework is proposed:

1. Construct a directed graph G(V, E). The Graph consists of vertices that are constituents of FREEDOM framework (F, R, E_1, E_2, D, O, M), where F – Future representation, R – Reality representation, E_1 – Requirements Engineering, E_2 – Fulfillment Engineering, D – Design and implementation, O – Operations, and M – Management. The edges of the graph are links between the vertices.
2. Use the edges of the graph to represent flow of generic knowledge/artifact flow between units of FREEDOM framework. Use a specific color for representation of these links.
3. Taking into account, that between constituents of FREEDOM framework there exist not only a general flow of artefacts (knowledge forward propagation), but also the information that is obtain via feedbacks, and monitoring, audit and analytics functions [27]; represent this information flow by edges of different color.
4. Assign to edges direction, which allows seeing which unit initiated transaction of information or produced artifact.
5. Assign the name of knowledge/artefact to the edges if appropriate.

Below the method is illustrated on the Waterfall and Scrum system development processes.

Flow of knowledge/artifacts in Waterfall development process

Waterfall is a linear method of development that is done by planning every single application's requirement in advance and then following strict set of guidelines in a sequential order as it is created from scratch [16]. Fig. 2.2.3.6 describes step-by-step life cycle of software development using Waterfall model. The process starts with requirements discovery and research phase, when information is gathered and project goals are determined. This is done by FREEDOM functions E_1 and E_2 monitoring, auditing and applying various analytics methods to information gathered from F, R and M. Requirements documents are then produced, and D builds Software according to requirements, which is then are used by O to provide value to customers.

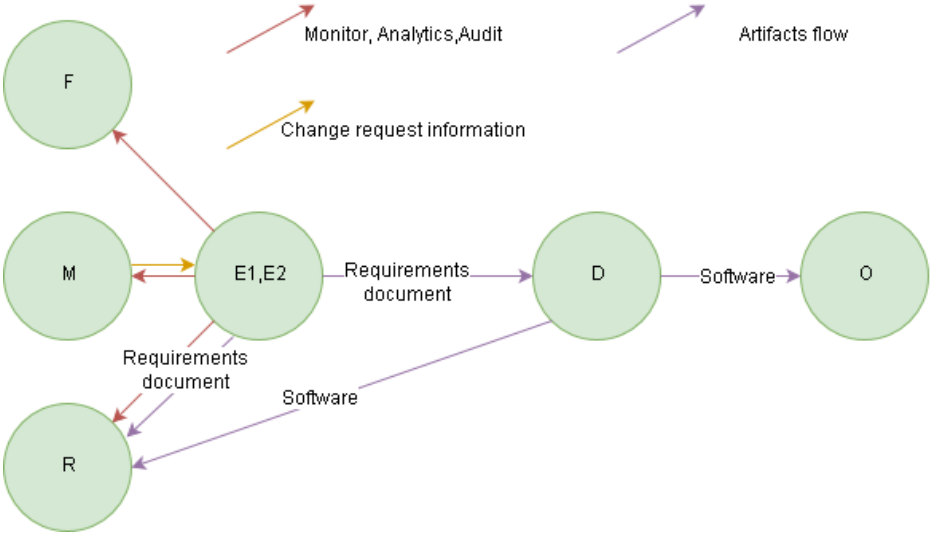


Fig. 2.2.3.6. Artifact flow in FREEDOM framework using Waterfall model

This process is very linear, and any unexpected changes in F, R or M after Requirements document is produced lead to costly re-engineering of requirements, as the already done work may have to be discarded [16].

Flow of artifacts in Scrum development process

Scrum is a framework within which people can address complex adaptive problems, while delivering products with highest possible value. Development as well as requirements engineering process is done iteratively, using continuous communication and collaboration with customer, reviewing what is done and accordingly adjusting requirements. In [14] and [17] application of requirements engineering to Scrum development process is described, together with various artifacts produced during requirements engineering. List of artifacts produced (middle column) and processes that produced them (left column) are as follows [17], see Table 2.2.3.4.

Table 2.2.3.4. Artifacts produced during Scrum development

Function	**Artifact**	**FREEDOM constituent**
Envisioning product	Product roadmap	F
Requirements elicitation	Requirements backlog	E_1
Requirements elicitation	Product backlog	E_1
Backlog grooming	Story map	E_2
Prioritization	Sprint backlog	E_2
Development (Sprint)	Increment	D
Inspect and adapt	Wiki	O

According to Table 2.2.3.4, graph showing artifact flow can be produced – see Fig. 2.2.3.7. Here artifact flow from FREEDOM functions is clearly visible, as one function produces an artifact, and others use it as an input. For example Story map produced in requirements engineering is used by E_2 to produce Sprint backlog, which is further used in developing software. All constituent units also contribute to reality representation, and possibly to future representation, which will be used in next sprint to determine new requirements for the sprint.

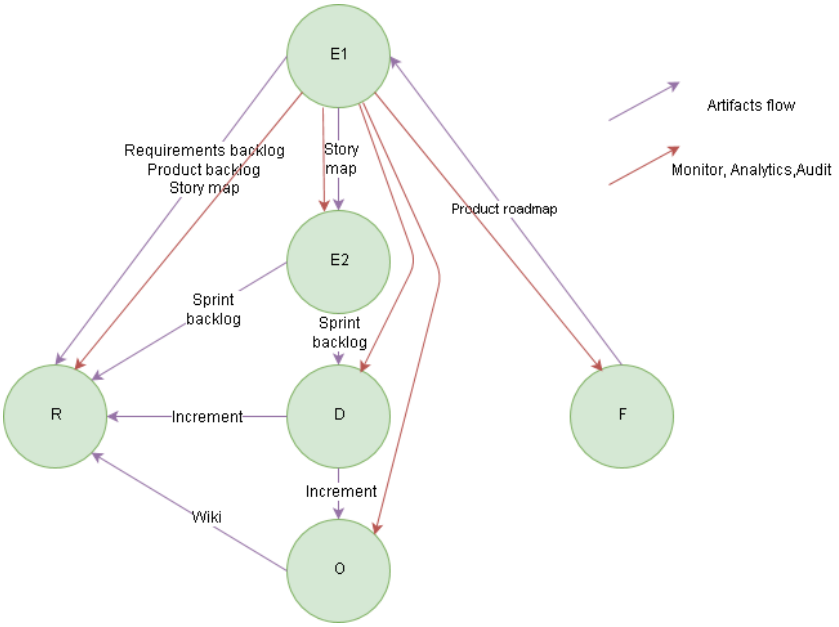


Fig. 2.2.3.7. Artifact flow in Scrum

In Fig. 2.2.3.7 one iteration – sprint is displayed; however, every sprint follows the same process, and this cycle can be repeated until project is done. This allows rapid discovery of new requirements and implementation of new features.

Taking into account that E_1 may be viewed as multiple smaller functions, it is possible to develop more detailed graph, describing interaction of smaller functions within large scale FREEDOM framework constituents (see Fig. 2.2.3.8).

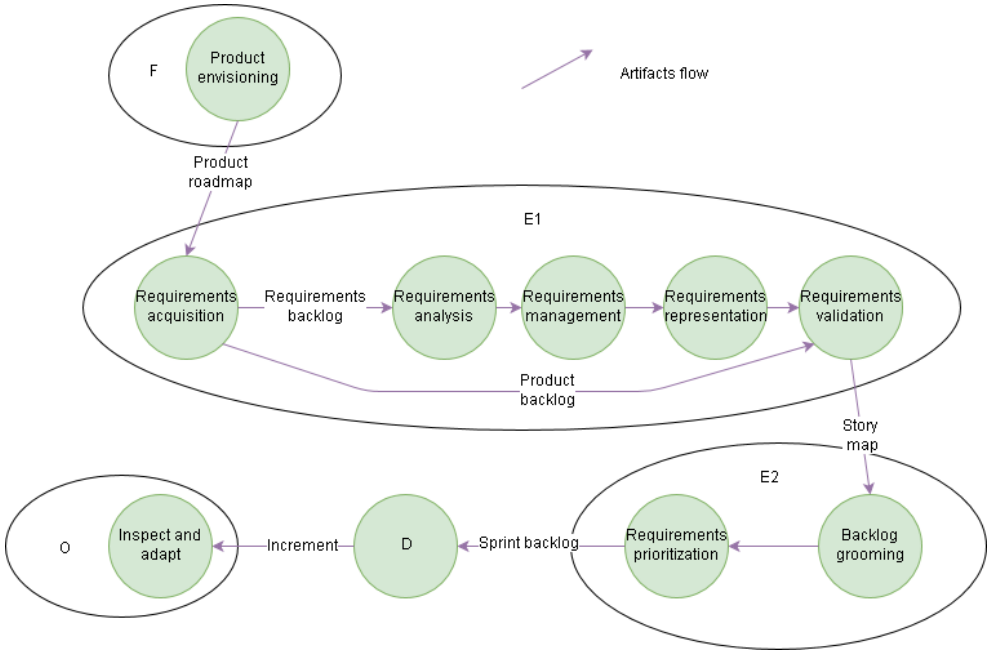


Fig. 2.2.3.8. Detailed artifact flow in Scrum.

The artifact flow shown in Fig. 2.2.3.8 basically is the same as the one in Fig. 2.2.3.7; the only difference is in the level of detail. Here sub-functions of FREEDOM constituent units (E_1, in particular) are shown.

Generic flow of knowledge/artifacts in FREEDOM framework

In continuous requirements engineering, beside the knowledge forward propagation, additional information flow must exist between E_1 and E_2, D and O, as requirements engineering continuously monitors, audits and performs analytics as well as receives feedback from all other elements. Feedback is evaluative information about activities or artifacts of E_1 [15].

To ensure continuity of requirements engineering the following information relationships must exist in the framework (see Fig. 2.2.3.9):

- Knowledge forward propagation from E_1 to E_2, D, O, M, R and F. This direct transfer of knowledge is represented using violet color in Fig. 2.2.3.9.
- Knowledge supply from F and R, both future representation and reality representations should be available for E_1.
- Feedback should be available from all constituents of the framework. In continuous requirements engineering, the feedback is evaluative information about activities or artifacts of E_1. However, as the feedback should be available from all constituents to any other, it is not shown in Fig. 2.2.3.9.
- Information can be acquired by monitoring, applying analytics to and auditing other constituents of framework, shown by red lines in Fig. 2.2.3.9.
- Management can issue change requests to all constituents of framework (depicted by yellow color).

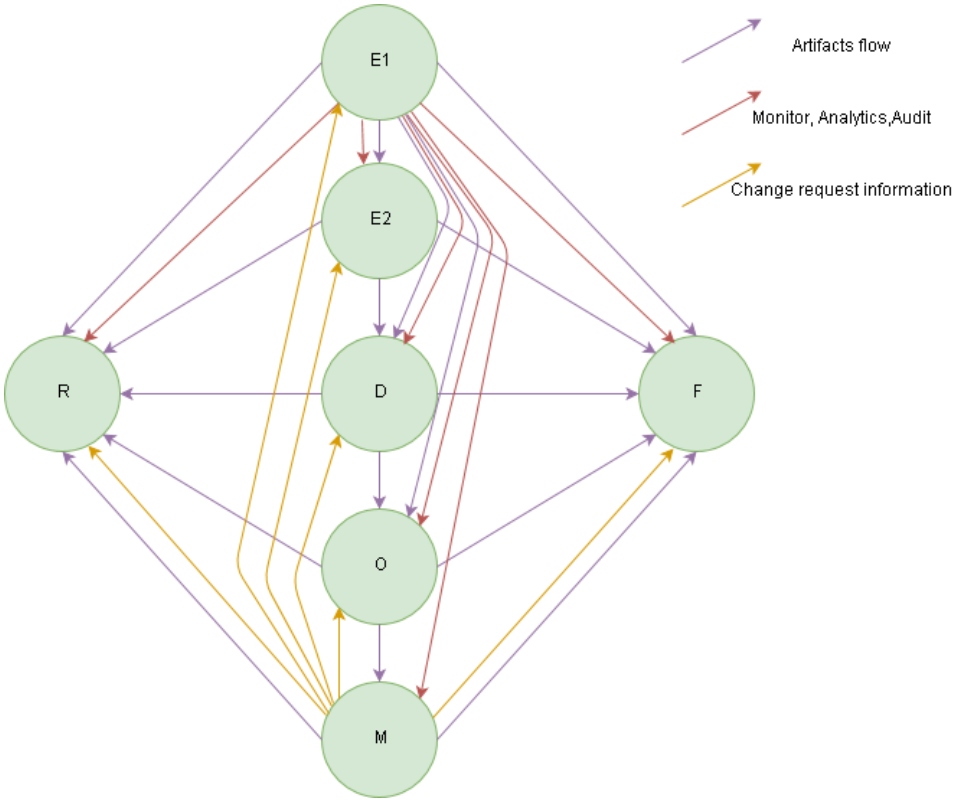


Fig. 2.2.3.9. Information relationships in FREEDOM framework

In Fig. 2.2.3.9 high abstraction level relationships between FREEDOM framework functions are displayed. Feedback information flows are not displayed, as they exist implicitly between all elements. However, should the need arise then feedback information flow also may be displayed, using differently colored edges.

The proposed method uses directed graphs with color coded edges to create user-friendly and rich with information representation of flows of information in various use cases of FREEDOM framework. Using developed method, four graphs were created, which contained all relevant information about flow of artifacts in chosen situations. Developed graphs may be further elaborated using more detailed functions of FREEDOM constituents and analyzed using various graph analysis methods.

The above described method does not restrict changes of the states of the knowledge/artifacts in the representation of the knowledge/artifact flow in the FREEDOM framework. However the state changes of the artifacts have to be controlled. The possibilities to control the states of knowledge/artifacts were analyzed earlier and are reflected in [18]. Integration of the methods of direct graph based knowledge/artefact flow representation and knowledge/artefact state control, in the requirements engineering tool based on FREEDOM framework, is a matter of further research.

2.2.3.6. Methodics for requirements engineering knowledge/artefact maintenance and distribution

The methodics (a simplified methodology) for requirements engineering knowledge/ artefact maintenance and distribution is based on a particular requirements management model. The proposed model and an initial version of the methodics are published in [19]. The model is represented in Fig. 2.2.3.10.

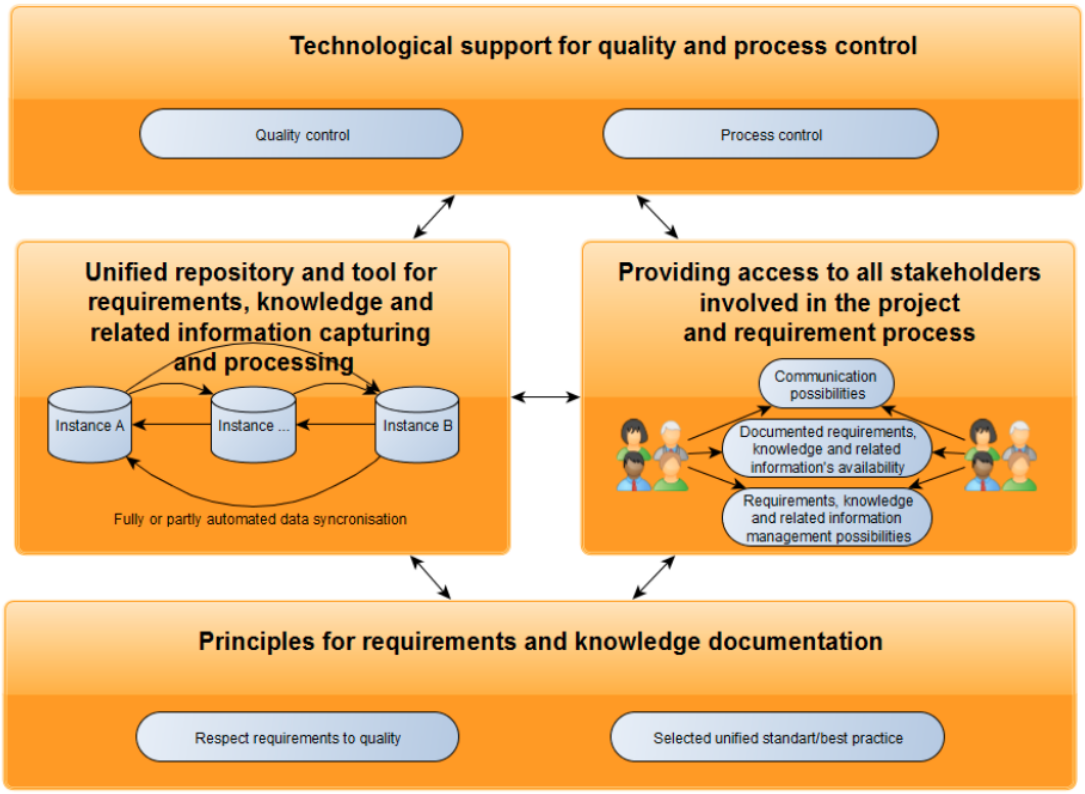


Fig. 2.2.3.10. Model of requirements management (source: [19])

The methodics includes two types of blocks, namely (1) generic blocks that *can* refer to any knowledge/artifact handled during requirements engineering and system development and (2) sequential blocks that *must* apply to any artefact. Bellow the description of the methodics is provided. For the sake of simplicity, the notion

"knowledge/artifacts" is represented with "artifacts" in the description of the methodics. The methodics assumes that (1) the maintenance and distribution of artefacts is based on the FREEDOM framework, and (2) the application of the framework is supported by the dedicated requirements engineering tool or a combination of tools that can support the FREEDOM framework.

Generic blocs of the methodics

Generic block 1: *Artifact and process quality management*. There are no restrictions on the use of quality management methods in the methodics. They can be chosen according to the traditions and legal obligations of a particular requirements engineering situation. However, it is strongly recommended that with respect to the processes that are supported by information technology solutions the method for ensuring compliance between normative documents and the process is applied. One of such methods has been developed during this project and is exemplified in [20].

Generic block 2: *Ensuring awareness of external environment*. The methodics does not prescribe specific methods for ensuring awareness of external environment. However, it recommends monitoring the states of relevant external artefacts and applying data analytics methods were appropriate. Some possibilities of application of data analytics are amalgamated in [21].

Sequential blocks of the methodics

Sequential blocks of the methodics are visualized in Fig. 2.2.3.11.

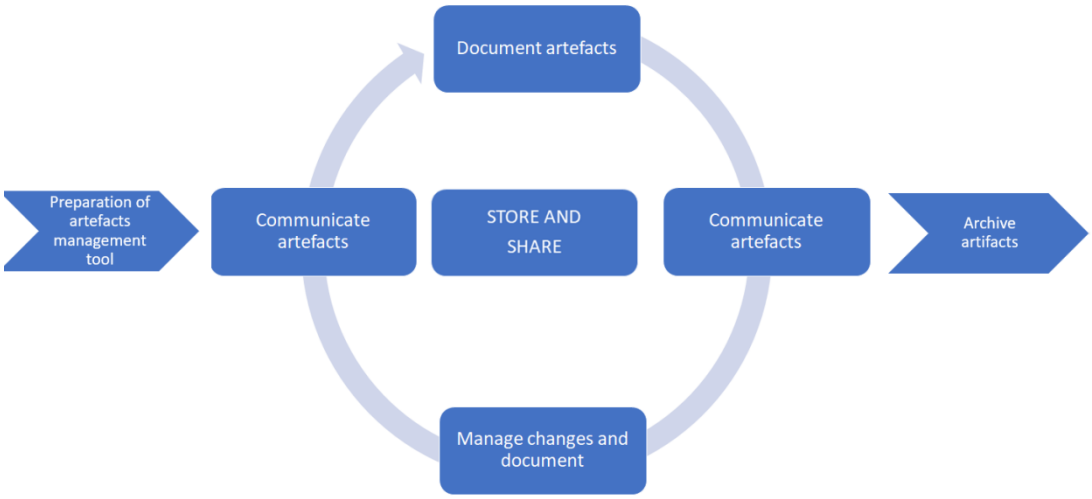


Fig. 2.2.3.11. Sequential blocks of artefact maintenance and distiribution methodics

In Fig. 2.2.3.11 we can see a graphical representation of sequential blocks of requirements engineering artefact maintenance and distribution methodics that comply with the model of requirements engineering represented in Fig. 2.2.3.10. The blocks comprise the method that includes activities that provide possibility to maintain artefacts and distribute them within the basic constructs represented in the model of requirements management represented in Fig. 2.2.3.10 [19]. Bellow the description of the method is provided.

Sequential Block 1. Activity 1. The first activity is the artefacts management tool's preparation – we need to choose a tool which will provide all possibilities to store and

share artefacts and manage them during the project and during the information technology solution life cycle.

When selecting the requirements engineering tools it is important to consider the following issues:

- Tool possibilities – functionality – is it possible to store information in a long term, is it possible to share, to comment, to track changes, etc. It is recommended that the tool can store the information about time intervals and instants (moments).
- Tool usability through the whole project and even during all information technology solution life cycle and beyond it. It is strongly recommended that the tool has a possibility to track artefact state changes and reflect artefact flow during requirements engineering.
- Tool variability capacity – does the tool support multi project management.
- Tool flexibility and simplicity – is it possible to perform artefact management, including their maintenance and distribution, with reasonably small effort.
- Possibility to customize the tools.

Sequential Block 2. The next four activities are represented in the circle that can involve several iterations. In the cycle Activity 3 (Artefact communication) appears twice to emphasize that the changes in the artifacts must be communicated. It is strongly recommended to visualize the artifact flow and control artifact states according to the graph based method of artefact representation in FREEDOM framework proposed in this project.

Activity 2. Document artefacts – it is a necessary condition for completion of further activities and providing necessary availability of the artefacts.

- This activity can be executed repeatedly. First, it is necessary to document the first artefact versions. After that it is necessary to document changes or additions to the existing artefacts.
- The form of documentation can vary depending on project management methods and other factors. However it must comply with quality management methods chosen in the Generic Block 1.

Activity 3. Communicate artefacts – requirements communication is a mandatory activity. It allows ensuring whether the artifacts are correctly documented and facilitates getting necessary approvals. It also helps to provide information about projects progress. There are two basic types of artifact communication: communication via tool(s) and communication via socialization.

- *Communication via tool(s).* The possibility to use one tool for artefact storage and communication can minimize necessary time for artefact management in several tools. Therefore having just one tool is preferable. Communication includes artefact distribution to stakeholders The result of this communication or distribution can be documented as links to developer tasks, to testing tasks and testing results, etc.
- *Communication via socialization*. Communication via socialization does not exclude tool usage, however, it involves several other aspects that are amalgamated and discussed in [22].

Activity 4. Manage changes and document them – during the project and during the information technology solution life cycle, there will be changes in artefacts and there will also be new artefacts. It is very important to manage them and to document them. It allows to provide actual information during all phases.

Sequential Block 3. Activity 5. All results must be stored in the tool and shared, i.e. all artefacts have to be available to all stakeholders according to their access rights.

Sequential Block 4. Activity 6. The last activity is old artefact archiving – in the phase when the project ends or information technology solution is retired, it is necessary to decide where to store the "old" artefacts. Advanced tools provide possibility to archive information with the possibility to access it and use it if necessary.

Although Fig. 2.2.3.11 does not show it, the method described above includes the goal to get a fully provided historically sorted information related to artefact life cycle (artefact creation or documentation, changes, realization etc.). In Fig. 2.2.3.12 we can see a visualization of possible results – a number of related artefacts and information that we can get if we continually perform all described activities and use the assumed functionality that tool(s) provide.

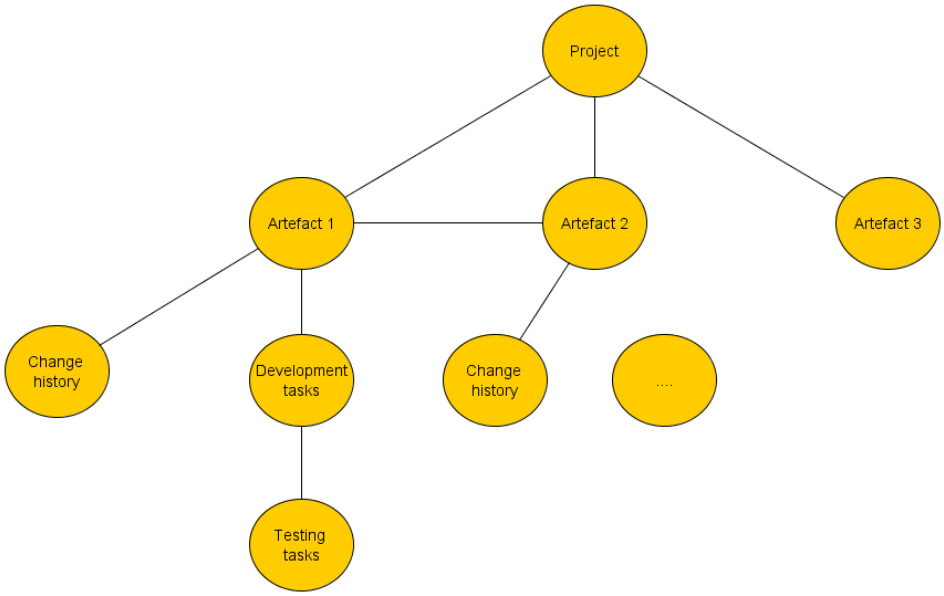


Fig. 2.2.3.12. Requirements Engineering artefacts, links, and related information

The linkage of different elements, e.g. development tasks, testing tasks, etc. provides a possibility to get the whole picture of the usage process and feedbacks. For instance, the links between artefacts, development tasks and testing tasks can provide testing results as a feedback to documentation of artefacts – respectively, it is possible to analyze and make conclusion if the documented artefacts are complete and do not contains mistakes or gaps.

The method suggests communication and sharing of artifacts with as few tools as possible. Fewer tools with rich functionality can bring in a possibility to exclude, e.g. e-mails for information's sharing and delivering comments. It has to be taken into account that having fewer places to manage artefacts will require less time for the

management process. For socialization, such tools as WhatsApp, Messenger, and Skype can be considered.

As described above, the methodics for requirements engineering knowledge/artefact maintenance and distribution is developed according to the FREEDOM framework. The main idea is to provide possibility to distribute requirements engineering artifacts from one requirements engineering function to others. The method does not prescribe a specific tool for requirements engineering. There are advanced requirements engineering tools that can support most of activities of the sequential blocks of the methodics (e.g. DOORS ng). However, the tools scarcely support visualization of artefact flow, artefact state control, and artefact and activities monitoring, analytics and audit prescribed by the FREEDOM framework; and cannot support all variations of the FREEDOM framework.

The proposed methodics has the following strengths:

1. It does not contradict to existing requirements engineering approaches and thus gives an opportunity to utilize existing requirements engineering body of knowledge and tools.
2. It is based on the FREEDOM framework, which is flexible and can be applied in various configurations; and has been validated with respect to requirements engineering relevant theories, project management methods and problem domains.
3. The methodics prescribes having awareness of external environment.
4. The method suggests having control of the states of the artefacts.
5. The methodics combines toll based artefact communication with artefact communication via socialisation.
6. The methodics incorporates data analytics for handling external and internal artefacts and processes.
7. The methodics supports continuous requirements engineering.

At the current stage of development, the methodics is complete to the extent that it can be applied in information technology projects. However the methodics still has the following limitations:

1. The dedicated tool for methodics support is not yet designed and developed.
2. The method for integrating developed FREEDOM framework temporal concept ontology with the developed graph based artefact flow visualization method is not yet designed, and thus is not included in the methodics.
3. The algorithms for data analytics application in FREEDOM framework are not yet developed and thus are not included in the methodics.
4. The further research will be focused on overcoming of above mentioned limitations by designing tool prototype in a one of the tool development environments (such as e.g., ADOxx or META EDIT++); by integrating FREEDOM framework temporal concept ontology into developed artifact handling methods; and by incorporating data analytics algorithms in the FREEDOM framework and in the prototype of the tool that supports the framework.

References

1. Hobbs, J.R., Pan, F. Time Ontology in OWL. Available at: https://www.w3.org/TR/owl-time/
2. Protégé. Software, available at: https://protege.stanford.edu/

3. Snoeck M. Enterprise Information Systems Engineerig, Springer, 2014.
4. Karlsen A. Enterprise modeling practice in ICT-enabled process change. Proceedings of the IFIP Working Conference on The Practice of Enterprise Modeling PoEM 2011, Springer, pp. 208-222.
5. van Osch W. and Steinfield Ch. Team boundary spanning: strategic implications for the implementation and use of enterprise social media. Journal of Information Technology, June 2006, Volume 31, Issue 2, pp. 207-225.
6. Chouder M.L., Chalal R., Setra W., Requirements analysis in data warehouses to support external inforamtion. Proceedings of the Computer Science and its applications, IFIP AICT 456, Springer, 2015, pp. 243-253.
7. Fraga A., Llorens J. Alonso L., Fuentes J. Ontology-assisted systems engineering process with focus in the requirements engineering process. Complex System Design and Management, Springer, 2015, pp. 149-161.
8. Ross H-L. System engineering for development of requirements and architecture. Functional Safety for Road Wehicles, Springer 2016.
9. Shepherd D.A., Patzel H. Trailblazing entrepreneurship, Springer Nature, 2017.
10. Ramburn A., Seymour L.F., Gopaul A. Understanding the role of knowledge management during the ERP implementation lifecycle: preliminary research findings relevant to emerging economies. Proceedings of the IFIP WG 8.9 Working Conference, CONFENIS 2012, Revised selected papers (G. Poels, ed.), Springer, 2013, pp. 234-241.
11. Yang L., Qian L., Ding Sh., Xu Y. Ontology-based design knowledge representation for complex product, Knowledge engineering and managment, Springer, 2014, pp. 277-287.
12. Laporte C. Y., Munoz M., Miranda J.M., O'Connor R.V., Applying software engineering standards in Very Small Entities: From startups to grownups, IEEE software, 2017, Vol. 35, Issue 1, pp. 99-103.
13. Zhang L. L., Lee C.K.M., Xu Q. Towards product customization: An integrated order fulfillment system. Computers in Industry, Vol. 61, Issue 3, 2010, pp. 213-222.
14. Kirikova, M. Continuous Requirements Engineering. Proceedings of the 18th International Conference on Computer Systems and Technologies, CompSysTech'17, Ruse, Bulgaria, June 23-24, 2017. ACM Press, 2017, pp. 1-10.
15. Kirikova, M. Continuous requirements engineering in the FREEDOM framework: A position paper. *CEUR Workshop Proc.*, vol. 1564, 2016.
16. Stepanova, E. Continuous Requirements Engineering Method for Mobile Application Development. Master Thesis, Riga Technical University, 2017.
17. Ivanova, A. Requirements Engineering in Agile Scrum Software Development. Master Thesis, Riga Technical University, 2017.
18. Peņicina, L. Controlling Business Object States in Business Process Models to Support Compliance. No: PoEM-DC 2016 : Doctoral Consortium Papers at PoEM 2016 : Proceedings of the Doctoral Consortium at the 9th IFIP WG 8.1 Working Conference on The Practice of Enterprise Modeling (PoEM-DC 2016) : CEUR workshop proceedings, Volume 1765, 2016, pp. 6.-13. ISSN 1613-0073.
19. Finke, A. Continuous Requirements Engineering Support Environment Model in Methodologically Heterogeneous Projects. Business Information Systems

Workshops: BIS 2017 International Workshops: Revised Papers, Poznań, Poland, June 28-30, Lecture Notes in Business Information Processing, Vol. 303. Springer, 2017, pp. 207-215.
20. Peņicina, L. Towards Supporting Business Process Compliance with Policies. Perspectives in Business Informatics Research: 16th International Conference BIR 2017: Proceedings, Copenhagen, Denmark, August 28-30, 2017. Lecture Notes in Business Information Processing. Vol. 295. Springer, 2017, pp. 93-107.
21. Kirikova, M., Purmalietis, K. Analytics in Continuous Requirements Engineering. Joint Proceedings of REFSQ-2017 Workshops, Doctoral Symposium, Research Method Track, and Poster Track co-located with the 22nd International Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2017), Essen, Germany, February 27, 2017. Ceur-ws.org, Vol. 1796, pp. 1-9.
22. Finke, A. Socialization Aspect in Requirements Engineering. REFSQ 2017 Joint Proceedings of the Co-Located Events: Joint Proceedings of REFSQ-2017 Workshops, Doctoral Symposium, Research Method Track, and Poster Track co-located with the 22nd International Conference on Requirements Engineering: Foundation for Software Quality, Germany, Essen, February 27, 2017. CEUR-WS.org, Vol. 1796, pp. 1-6.
23. Kirikova M. Variable Contents of Enterprise Models. Procedia Computer Science, Vol. 104, ICTE 2016. Elsevier, 2017, pp. 89-96.
24. Būcena, I. DevOps Adoption for Very Small Entities. Master Thesis, Riga Technical University, 2017.
25. Bucena, I., Kirikova, M. Simplifying the DevOps Adoption Process. Joint Proceedings of the BIR 2017 pre-BIR Forum, Workshops and Doctoral Consortium co-located with 16th International Conference on Perspectives in Business Informatics Research (BIR 2017), Copenhagen, Denmark, August 28-30, 2017. Ceur-ws.org, Vol. 1898.
26. Purmalietis, K. Continuous Requirements Engineering for Start-Ups. Master Thesis, Riga Technical University, 2017.
27. Rocque, S.C. Towards Supporting Switching from Traditional System to Cloud ERP for SME. Master Thesis, Riga Technical University, 2017.

2.2.4. Development of the methodology and framework for semantic web services integration

In 2014 several solutions were studied and analyzed before starting development of Semantic Web services: OWL-S, Web Service Modelling Ontology (WSMO) and Semantic Automated Discovery and Integration (SADI). Following the results of study and analysis, SADI technology has been selected. A service preparation template was developed to provide automated generation of plug-ins and additional files, thus accelerating the development of services. Semantic Web services that semantically annotate the information received from traditional web services that are included in e-LOGMAR portal were developed for processing information related to the logistics domain (available routes and their types, cargo expenses, etc.). A logistics domain ontology, which was built using the OWL language, supports the annotation process. The OpenCalais web service was used in the developed services to integrate natural language processing methods with Semantic Web technologies, allowing to retrieve semantic metadata from the text and to return a semantically

annotated document to the client. Semantic Web technologies have also been integrated in the model driven software configuration approach.

In 2015 a modified MTM (Model – Transformation – Model) approach based on MDA (Model-Driven Architecture) and Semantic Web Technologies was developed. It makes it easier to implement repeated usage and integration of Software Configuration Management data.

In 2016 the developed EAF methodology, which is based on MDA approach and is aimed at software configuration management and automation of IT operations for software practical application, was further improved. A survey was implemented to determine the main continuous processes that ICT companies are going to automate, the current automation level of these processes, and the main challenges in the automation field. The survey had 42 respondents from more than 35 ICT companies in Latvia.

The semantic services development and integration methodology utilizes model-driven approach to design, develop, and maintain services with reusability in mind. One of the key elements of the approach – the Reusable Functions Library – was developed at this stage of the project. This library makes it possible to reuse implemented automation functions in different projects and different workflows.

The methodology for systematic development of RESTful Semantic Web services using SADI framework was further developed. The methodology was used to develop a set of Semantic Web services that are used in eLOGMAR logistic portal (available at http://www.elogmar.eu). The validation was implemented in collaboration with an industrial partner – company Logitrans Consult (Estonia).

In 2017 the research on development of Semantic Web services, software configuration management, and automation of IT activities was continued, and the researchers also started to study the questions of agile software testing as well as information technology transfer and commercialization. The obtained results are described in Subsections 2.2.4.1 and 2.2.4.2.

2.2.4.1. Testing of integration of Semantic Web Services and extented ontology-based approach

The growth of technology made software interaction to the user as an integral part of the day to day life. More humans, machine and robots depend on and work with software. The growth of software tools and technologies made software development fast and simplified. According to recent survey software development companies spending more budget on the software validation and verification to make application error free before it reaches to the customer and it created the market competition with several testing tools and practices. Software development becomes more advanced, fast, agile with continuous integration, continuous deployment, and test automation. This advancement also brings the new challenges to quality assurance teams with agile methodology adopted projects. Agile software development principles are individuals and interactions, working software, customer collaboration, responding to change, quality focus, iterative, incremental and evolutionary, very short feedback loop and adaptation cycle, efficient and face-to-face communication make it more popular.

Standard lean canvas is used for the evaluating the business. The lean canvas is a white board with several blocks with title names. It is simple, lightweight one-page design it shows the product design to market segmentation and market fit. Main

contribution is to find the proper design of lean canvas and identifying the adoption in agile software testing (Fig. 2.2.4.1):

- Finding the lean canvas design for the agile software testing;
- Improving the test planning;
- Simplifying test strategy;
- Indenting the best fit, lean canvas for the agile;
- Utilizing the lean principles in lean canvas design;
- Finding the most appropriate blocks for the lean canvas design.

Fig. 2.2.4.1. Prototype agile lean canvas design board for testing

The software configuration management (SCM – Software Configuration Management) is an information technology discipline which is responsible for tracking and managing any software changes. In order to ensure the quality of the final product, configuration management performs multiple tasks: identification of configuration units, version control, construction and installation governance, tracking of configuration units etc. The semantic web technology (RDF and OWL languages) in the model-guided approach is used to establish the formal, machine-readable SCM ontology describing the concepts of SCM and relationships between them. The model-guided approach is studied in order to develop the semantic web services for the logistics portal e-LOGMAR with the aim of:

- completing the detailed analysis of the underlying semantic web services;
- researching how the semantic web technologies can be effectively integrated into the existing systems;
- developing and improving a methodology for the systematic development of the semantic web services;
- integrating the semantic web technologies into the software configuration management approach EAF.

Five most common semantic web service development technologies were analysed during the research: SADI (Semantic Automated Discovery and Integration), OWL-S (Ontology Web Language for Services), WSMO (Web Service Modelling Ontology), WSDL-S (Web Service Description Language-Semantics) and SAWSDL (Semantic Annotation for Web Service Description Language). According to the results of the

technology analysis it was determined that, although the SADI technology offers fewer possibilities compared to the OWL-S and WSMO, it still facilitates the semantic web service significantly which could resolve the set objectives, development and maintenance. Another important SADI framework advantage is the open code. After the technology analysis, the methodology for systematic RESTFul semantic Web service development was advanced and improved through SADI (Semantic Automated Discovery and Integration) framework.

The methodology includes the following four main activities:

- Definition and placement of the domain and service (input and output classes) ontology;
- Generation of the skeleton service (Java class plug and other additional documents);
- Adding the business logic to a service (Java class and other required files);
- Compiling of the service and the placement of the acquired WAR (Web application ARchive) file in the application server.

This methodology was used to produce the semantic web services utilized in the Business portal e-LOGMAR in e-logistics. The developed services attempt to semantically annotate the information which is received from the services available in the portal. The information processed is related to logistics (available routes and their types, transport costs etc.). Several SADI services can be connected in a row and its completion results in receiving a continuous linked data (Linked Data) graph containing information regarding the initial input resource. The services use modified logistic ontologies LogiCO and LogiServ from TNO.nl (Fig. 2.2.4.2).

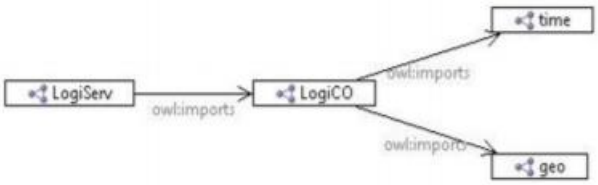


Fig. 2.2.4.2. LogiServ and LogiCO ontology

The methodology for development and integration of services uses the model-guided approach in order to design, develop and maintain the web services as well as to ensure their reuse in other future projects. One of the key elements of this approach is the development of the RFL (Reusable Functions Library). This library enables the reuse of the automation functions in other projects and workflows.

The provided methodology of SCM based on model-driven approach and semantic web has been validated in the process of development of Business portal e-LOGMAR in e-logistics.

The life cycle of each information technology begins with its development and continues with the transfer of the result to the end user. In reality, frequently, the prototypes that have been designed remain within the premises of the university and do not reach the public. Therefore it is crucial to draw attention to the sustainable development, improvement and the technology transfer process of the Semantic Web Portal e-LOGMAR.

The technology transfer provides the commercialisation of the products generated as well as the promotion, patenting and the market introduction of the applied research

results. According to the European Commission Regulation, which provides a common understanding of the state of technology development and its position in the chain of innovation, the Semantic Web Logistics Portal complies with the pilot development research type. When assessing the state of the portal according to the level of preparedness of the technology, it is assigned to the 5th or 6th Technology Readiness Level (TRL). In cooperation with the industrial partners from Estonia and Central Asia the semantic web services were examined and demonstrated in an artificial environment where the portal semantic services had been integrated with real aid elements.

In total, 44 partners from various transport and logistics companies participated in the study. The user feedback indicates that, in general, the portal functionality and user interface meet the requirements and are evaluated positively (Fig. 2.2.4.3 and 2.2.4.4).

Fig. 2.2.4.3. Assessment of the Semantic Web Logistics Portal functionality

Fig. 2.2.4.4. Assessment of the Semantic Web Logistics Portal user interface

For the elevation of the competitiveness of a new logistics portal and for the successful operation of the system, the prototype of the portal should be checked in a real environment (8 TRL).

By summarizing multiple sources it can be concluded that most of technology transfer patterns include the following common parts:

- the transfer object;
- two or more participants involved in the technology transfer process;
- the interaction between the participants;
- the transfer method which involves a number of steps;
- the environmental factors influencing the technology transfer process.

In order to transfer the Semantic Web Logistics Portal to end users, it is offered to use the process oriented information technology transfer pattern which will allow for verification of the technology's development and commercialisation processes as well as to assess and improve them (Fig. 2.2.4.5).

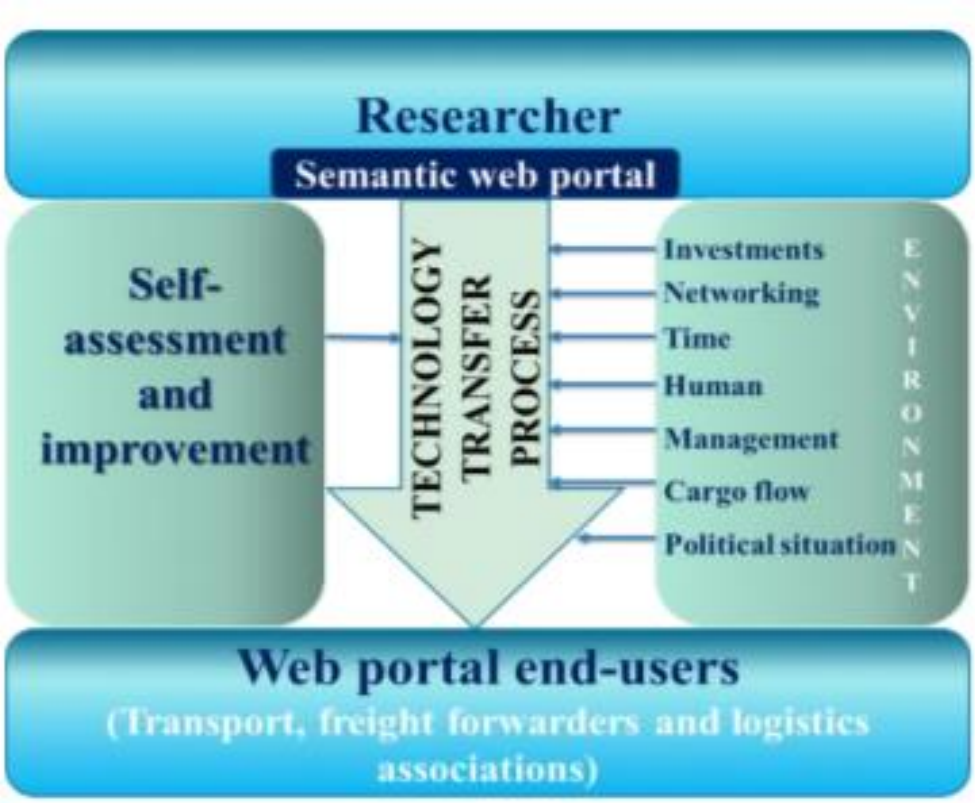


Fig. 2.2.4.5. Transfer pattern of the Semantic Web Logistics Portal

To to develop and improve the Semantic Web Portal, it is offered to use the process oriented information technology transfer pattern. The selfassessment system will ensure a common understanding of the technology transfer status and its position in the innovation chain. Given technology transfer approach is also applicable for any process oriented activity assessment and improvement and can be adapted to any organization needs.

2.2.4.2. Improvement and approbation of EAF approach

Short introduction to EAF approach

Let authors to start current paragraph with story about modern software development company. This company develops large and complex software, where many technologies integrated together: Oracle, Java, Ruby, .NET etc. Many bug tracking systems are going to manage development of mentioned software; huge procedure of software configuration management should be applied to manage source code in few repositories. Repositories are controlled by different version control systems, such as git and subversion. Frequent builds and deployments are required to support up to time testing process.

Firstly, it is extremely important to automate all these activities, which will called also IT operations, in context of current project. Secondly, automation of all mentioned IT operations should be prepared as soon as possible, because manual activities will waste time and will grow risks of human factor. Nowadays starts of new projects are like explosion and after few days customer wants to get the first release of software, however automated process, which can prepare release, is not ready.

Of course, mentioned software development company has a number of modern tools to automate IT operations: software configuration management, bug tracking, creation of software builds and deployments, continuous integration etc. [1]. Is it possible to manage all these operations by click of one button? For example, project, described before, has a tool and a script to manage versions of source code, a tool to make builds and deployments, scripts to manage issues in the bug tracking system. From one point of view it seems that all operations are automated, but really its automated separately and could not be managed by one click. Therefore, software configuration manager firstly should manage branches by version control tools and scripts, secondly, make software builds and deployments for test environments and finally, update information about related issues in the bug tracking systems.

Sometimes it is possible to achieve such automation level, which allows managing all mentioned operations by one click. However, are solutions for automation are reusable? How much time will takes implementation of similar automation in new projects? Are automations of different operations integrated together and could be controlled by one click? Usually, these questions become a challenge for software development companies and current report will describe some of possible answers of mentioned questions. The Fig. 2.2.4.6 summarizes the scope of this study and problems that will be justified.

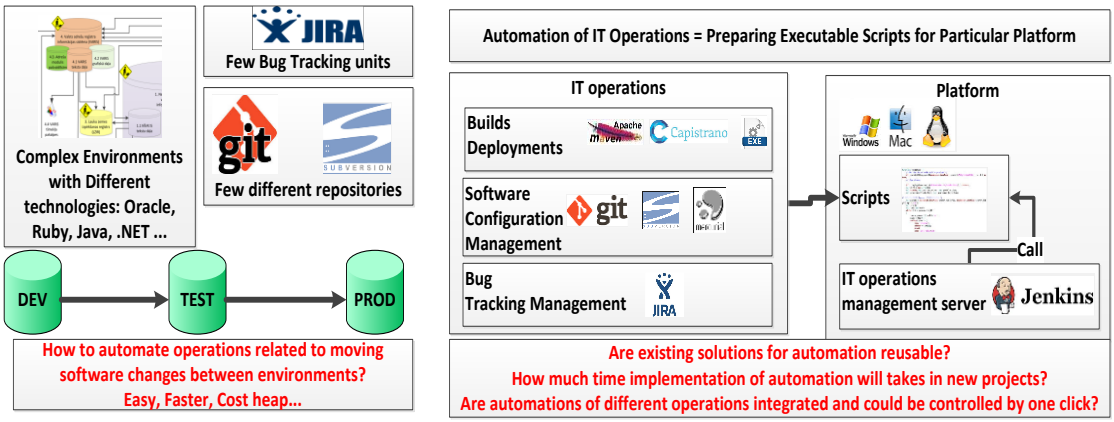


Fig. 2.2.4.6. The scope, definitions and problems of the study

Novelty of EAF approach
The following novel features are introduced in the EAF approach:

- A method for development of library of reusable source code for automation of different IT operations;
- Model-Driven approach how to generate source code for automation of operations in particular project using library of reusable source code;
- Meta-models for models, which implements new model-driven approach.

Background of EAF approach
Current research has been started four years ago. Then only software configuration management have been included in research topic. Firstly, during reading the books about best practices of software configuration management [2], [3], interesting problem have been found. Sometimes ready solution for automation of process of software configuration management is not in consequence with initial requirements of mentioned process. It means that purpose of some requirements have been lost during development of source code to automate process. Authors of mentioned books [2], [3] introduce to using models for initial requirements of process, for example, branching models, models of software builds and deployment process etc. After source code for automation of mentioned processes is ready, it should be checked for a consequence with models of requirements. It was one of the first attempts to use model-driven ideas in the field of software configuration management. Later, new ideas to use model-driven approach for software configuration management and automation of related processes are provided [4], [5]. There are some benefits using model-driven approach in the automation field:

- Generating source code for automation by model-driven approach could reduce manual efforts and save up time during development of code;
- Increase traceability between initial requirements and source code.

Later, ideas to use model-driven approach have been provided in few papers related to software configuration management [6], [7], [8]. Until year 2009, software configuration management and automation of related processes have been studied together with build and deployment management, integration with bug tracking, continuous integration etc. All these processes have been designated as IT operations, but huge methodology and research topic related to automation and improvement of mentioned operations have called DevOps [1], [9]. One of the main challenges of DevOps approach was statement that speed of IT operations should be quite high to get in time working software to customer during high level of agility [9], [10]. Nowadays many tools are related to automate IT operations and most of them use model-driven approach as base idea [1]. As an example, OpenMake tools could be mentioned. One of authors of OpenMake tools, Tracy Ragan told that novel solutions for automation of IT operations have not exist without model-driven approach, because static scripts could not be successfully applied in a cloud, where is no information about static addresses of servers or about platforms [9]. During research, described in this report, a number of tools and approaches related to automation have been studied. There are some disadvantages of mentioned tools:

- Sometimes, the scope of tools is only one particular IT operation, for example building and deployment. There are no recommendations how to integrate these tools with other tools related to automate other operations.

- Some tools require buying licences, installing a number of additional tools and refactoring a structure of existing projects. It could be a problem for managers of companies, because usually companies already have a number of trusted tools, script and best practices. Companies would like to increase reuse of existing and trusted tools and scripts, instead of buying new unknown and untrusted tools.

As a result, authors also have taken model-driven approach as main idea of new solution for automation. However, provided solution is not limited to using particular tools or automation of only some particular operations. Provided solution allows developing a library with reusable units of source code for automation of different operations. The library allows adding new units and allows working with existing and trusted tools and scripts. In addition, new model-driven approach allows generating source code for automation of operations in the particular project using mentioned library of reusable source code.

Researchers of this project have made a first attempt to adopt model-driven approach for software configuration management in 2014. Conception, described in [11] has names of models and steps for generating of source code for automation. However, meta-models are not developed. Later, the first meta-model of mentioned conception have been provided in the paper [12]. The first completed solution for automation of software configuration management have been provided in the paper [13], but first practical results described in [14] and [15]. Automation of software configuration management is extended by other IT operations, in addition library of reusable source code and meta-models are improved and a number of errors from previous versions are fixed. The Fig. 2.2.4.7 provides the main principles of provided approach. There is a method how to develop the library of reusable source code and model-driven approach, which could generate source code for automation of IT operations in particular projects.

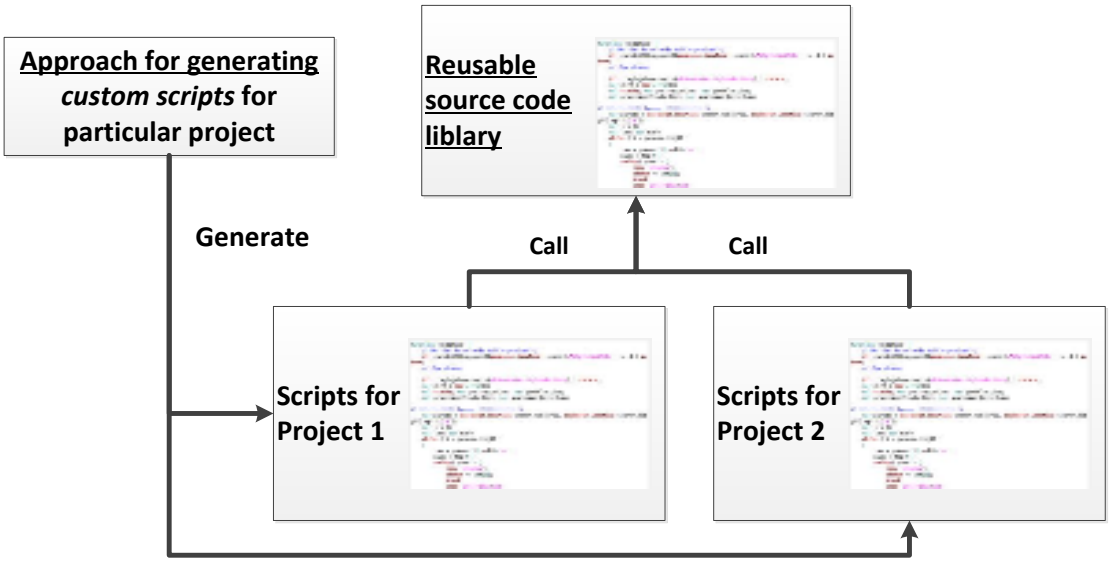


Fig. 2.2.4.7. Main principles of provided approach

Library of reusable source code is provided in Fig. 2.2.4.8.

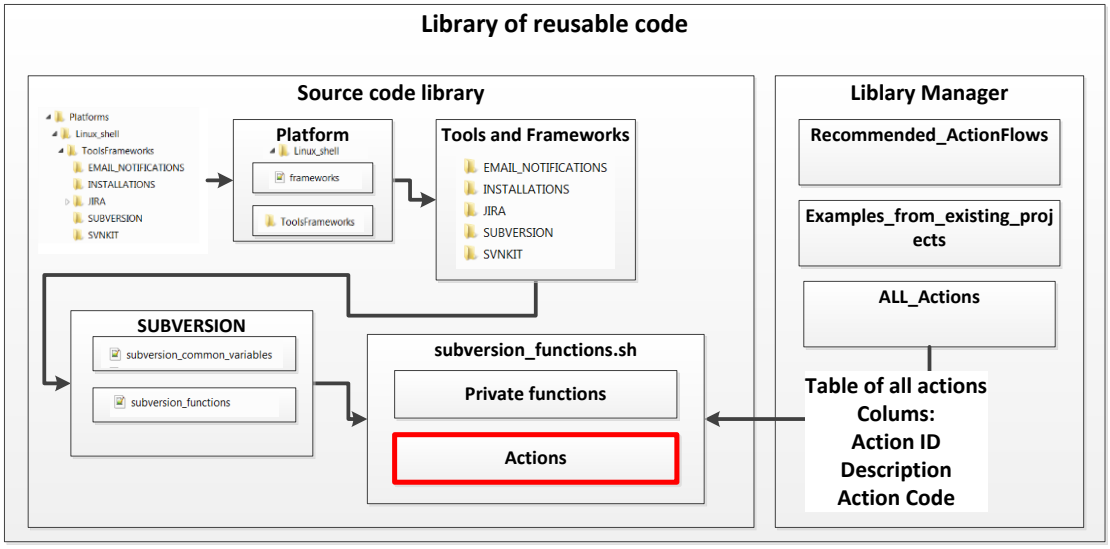


Fig. 2.2.4.8. Structure of library of reusable source code

General picture of EAF approach is represented on Fig. 2.2.4.9.

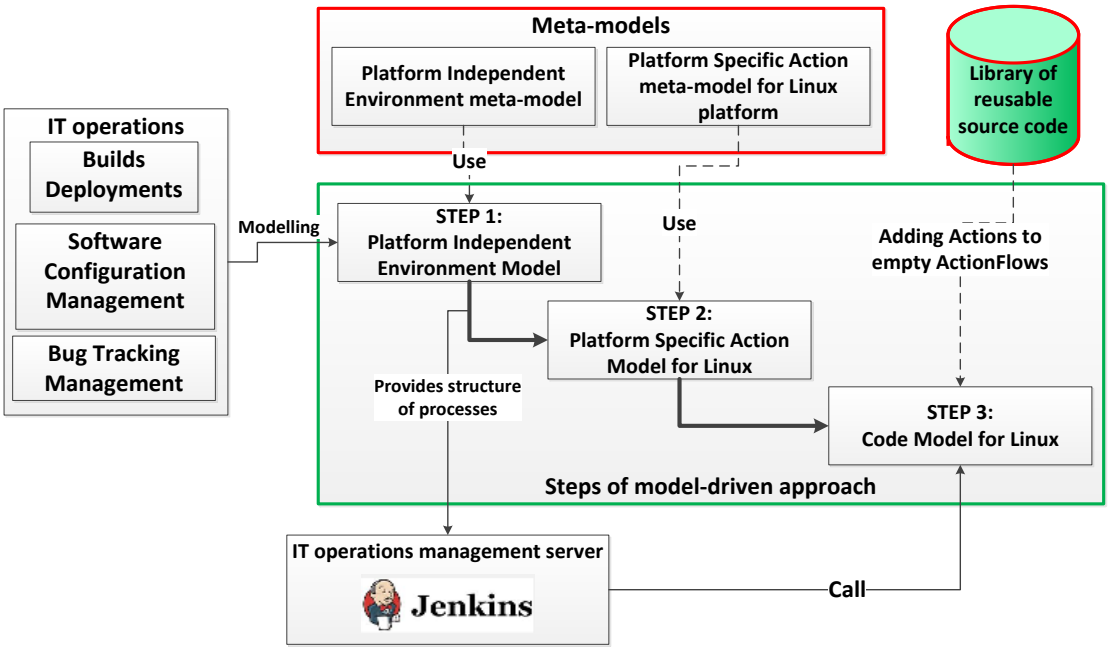


Fig. 2.2.4.9. Model-driven approach for generating source code for automation

Experiments and approbations of EAF

During the testing of provided model-driven approach, automation of IT operations has been implemented at five different software development projects. Implementation time has been fixed for each project. Then mentioned implementation time has been compared with related implementation time of automation, but before model-driven approach. The Fig. 2.2.4.10 shows comparison between automation time by old methods and by provided model driven approach.

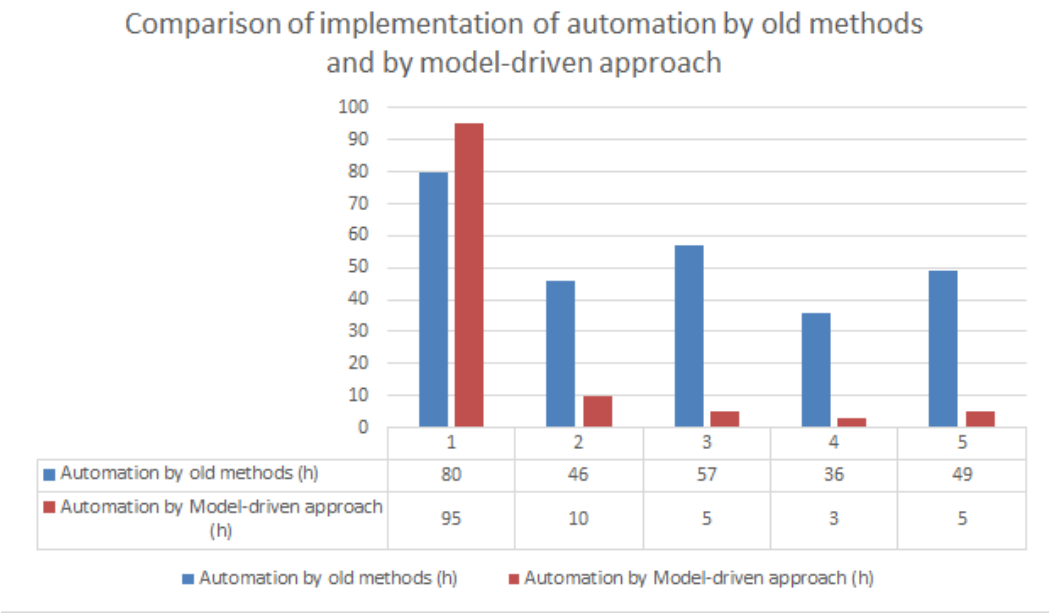


Fig. 2.2.4.10. Time of implementation of automation: comparison

Blue colour on graphic in Fig. 2.2.4.10 represents implementation time by old methods, but red colour represents implementation time by model-driven approach. Results of experiments shows, that until library of reusable source code is empty, implementation of automation by model-driven approach is not rational – it takes quite more time. However, after library of reusable source code contains all necessary Actions, implementation of automation by model-driven approach helps to save up time (project 2 – 36 hours, project 3 – 52 hours, project 4 – 33 hours, project 5 – 44 hours). To increase gains from provided model-driven approach, it is necessary to find how to fulfil the library of reusable source code as soon as possible. It will help to reduce implementation time at project 1.

Model-driven approach for implementation of automation of IT operations

Provided approach uses reusable source code for automation of single activities. A method for development of library of reusable source code also has been designed during current research. In context of provided model-driven approach, the following meta-models are designed:

- Meta-model for Platform Independent Environment Model;
- Meta-model for Platform Specific Action Model for Linux platform.

Practical experiments of provided model-driven approach shows that only fulfilled library of reusable source code could give benefits and save up time comparison of implementation of automation by old methods. It means that one of the most important further researches should be generating of source code for single actions and adding this code to library. It should decrease time of library development.

Now, software configuration manager, using knowledge about automation domain, manually writing source code for single Actions and adding it to library. Then library is fulfilled, it could be used by model-driven approach to generate source code for automation of operations in particular project.

In the future, source code for single actions should be generated by expert system. Expert system will use entered knowledge about particular domain (automation domain) and human experience. Such intellectual solution could provide a modern

approach for generating of reusable source code repositories for different domains. At the same time, models for generating custom source code could be easy because amount of reusable functions will be quite less than amount of elements of traditional programming languages.

References

1. Azoff, R. (2014). DevOps: Advances in Release Management and Automation. Available at: http://electric-cloud.com/wp-content/uploads/2014/06/EC-IAR_Ovum-DevOps.pdf
2. Stephen P. Berczuk. (2003). Software Configuration Management Patterns: Effective Teamwork. Practical Integration. Edition. Addison-Wesley Professional.
3. Aiello, R. (2010). Configuration Management Best Practices: Practical Methods that Work in the Real World. 1 Edition. Addison-Wesley Professional.
4. Giese, H. (2009). A Model-Driven Configuration Management System for Advanced IT Service Management. In International Conference on Model Driven Engineering Languages and Systems (MoDELS 2009). USA, October 4 - 9, 2009. IEEE Digital Library: 4th International Workshop on Models. 300-310.
5. Bird, C. & Zimmermann, T. (2012). Assessing the value of branches with what-if analysis. Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering, FSE 2012.
6. Buchmann, T., Dotor, A. & Westfechtel, B. (2013). MOD2-SCM: A model-driven product line for software configuration management systems. Information and Software Technology, vol. 55, no. 3, pp. 630-650.
7. Buchmann, T. & Westfechtel, B. (2014). Mapping feature models onto domain models: ensuring consistency of configured domain models. Software and Systems Modeling, vol. 13, no. 4, pp. 1495-1527.
8. Schwägerl, F., Buchmann, T., Uhrig, S. & Westfechtel, B. (2015). Towards the integration of model-driven engineering, software product line engineering, and software configuration management. MODELSWARD 2015 - 3rd International Conference on Model-Driven Engineering and Software Development, Proceedings, pp. 5.
9. Ragan T. (2014). 21st-Century DevOps--an End to the 20th-Century Practice of Writing Static Build and Deploy Scripts. Linux Journal, 230, pp. 116-120, Computers & Applied Sciences Complete, EBSCOhost, accessed on 22 October 2014.
10. Grzegrzólka, P. (2014). Configuration management in agile software development. BIR 2009 - 8th International Conference on Perspectives in Business Informatics Research.
11. Bartusevics, A., Novickis, L., Bluemel, E. (2014). Intellectual Model-Based Configuration Management Conception. Applied Computer Systems. Vol.15, 2014, pp. 22-27.
12. Novickis, L., Bartusevics, A. (2014). Model-Driven Software Configuration Management and Environment Model. In: Recent Advances in Electrical and Electronic Engineering: Proceedings of the 3rd International Conference on Systems, Communications, Computers and Applications (CSCCA"14), Italy, Florence, 22-24 November, 2014. Florence: WSEAS Press, 2014, pp. 132-140.

13. Bartusevics, A., Novickis, L. (2015). Model-based Approach for Implementation of Software Configuration Management Process. In: MODELSWARD 2015: Proceedings of the 3rd International Conference on Model-Driven Engineering and Software Development, France, Angers, 9-11 February, 2015. Lisbon: SciTePress, 2015, pp. 177-184
14. Bartusevics, A., Lesovskis, A., Novickis, L. (2015). Semantic Web Technologies and Model-Driven Approach for the Development and Configuration Management of Intelligent Web-Based Systems. In: Proceedings of the 2015 International Conference on Circuits, Systems, Signal Processing, Communications and Computers, Austria, Vienna, 15-17 March, 2015. Vienna: 2015, pp. 32-39.
15. Bartusevics, A., Novickis, L., Leye, S. (2015) Models and Methods of Software Configuration Management. Applied Computer Systems. Vol.17, 2015, pp. 53-59.

2.3. Model based data visualization and real-time verification of business processes

The Section 2.3 describes scientific results of SOPHIS program Project No. 2 that were obtained by the researchers of the Faculty of Computing (FC), University of Latvia (UL).

The studies were focused on the research and development of:

- Technologies for large scale NoSQL data base exploration and visualization.
- Business process runtime verification

2.3.1. Development of technologies for large scale NoSQL data base exploration and visualization.

New possibilities are researched in large scale data set analysis and visualization for new type of hardware – high resolution displays wall, consisting of many (more than 20) standard displays. In this research client-server environment is developed. This environment supports agent based modelling and relational data exploration and migration to NoSQL database with browser, that works with display wall.
Results of research of Development of technologies for large scale NoSQL data base exploration and visualization is published in 7 publications (DF1, DF2, DF3, DF4, DF5, DF6. DF7), 5 of them are indexed in Web of Science and Scopus data bases. Detailed information about rsearch is available in english report.
First stage of research was devoted to create a prototype of display wall. Main research problems were compatibility with popular operation systems and to keep cost of display wall as low as possible. Different solution architectures were analyzed and display wall prototype was developed partially according to raised requirements. Results of first stage are published in [DF1, DF2, DF3].
Second stage of the research was devoted to improvement of the prototype of display wall. Main research problem was to optimize amount of data transferred between computer and display wall. Different software and hardware compression methods were explored. One of possible usage of the display wall was explored – development of agent based modeling and simulating environment. Results of second stage are published in [DF4, DF5].
During stage 3 together with doctoral student R. Bundulis display wall architecture and data processing algorithms are improved. Prototype created is stable enough to start intensive research of usage of display wall. Created prototype is called Infiviz and is described in publications [DF6, DF7].
Different display wall development possibilities are explored with students of University of Latvia. It is impossible to work in traditional way with keyboard and mouse because of the large screen size. Display wall control with Android device is researched (bachelor thesis of student Mārtiņš Andersons). Even more effective control can be achieved, if display wall would be built with touch screen monitors. We found that it is possible to use high resolution monitors in display wall (bachelor thesis of student Laura Ķirsone). For Microsoft Windows applications to work fast, DirectX technologies must be implemented into Virtualbox virtual graphics card (course work of student Matīss Ķeiris).
During the development of monitor wall itself, there was research for the possibility to implement applications for monitor wall. However, using web-based tools on a real monitor wall resulted in a decrease in performance. The understand the reasons for the

performance decrease we study web technology guidelines, operating rules, and its internals.
Based on the results, a strategy has been created, and partially implemented, and its goal is to find the optimal combination of the right system configuration, tools, and web browser specific optimization, that would help in creating effective web solutions for high definition display walls (the master thesis of student Aleksandrs Rilins).
Also display of connected web pages or resourses on display wall is researched during third stage (bachelor thesis of student Kristīne Karlsone). Monitor edges interfere large graph display on display wall. First algorithms is developed that ensures that graph nodes are not split between monitors (bachelor thesis of student Jānis Kašs).
During stage 3 tree prototypes for large data visualisation was created. First prototype collects different spatial data of city (Riga) and shows them on map, showing which parts of city are more fitted to user criteria (the master thesis of student Jānis Peisenieks). Second prototype gathers information about paraplane flights near Baltic seeshore and visualizes model of adequacy of weather conditions for flying in different places based on weather forecast (the master thesis of student Vladislavs Maksimčuks). Third prototype ganerates different test pictures and tests adequacy of popular web browsers for displaying large complex pictures on display wall (the master thesis of student Normunds Pureklis).
Research was done in two directions during forth stage:

- Development of information visualidation infrastructure, during which display wall was improved technically and functionality was added,
- Development applications for display wall, during which was made approbation of display wall for two applications – model based data visualisation and visualisation of large graphs.

Positive references are got form partners who made approbation of prototypes.

2.3.1.1 Development of display wall

Efficiency and scalability issues was the main focus of the fourth stage. Display wall was built using medium power hardware, that provides 52 megapixel images Compared with Reality Deck solution that uses most modern technologies and provide 88 Mpix, our result is in line with project's goal of creating a relatively cheap and yet sufficiently powerful solution for large images. The authors of the work have successfully used both the standard (Google Chrome, PDF viewer) and domain-specific (video surveillance) software. This eliminates necessity to create a special display wall software - everything that runs on a standard PC will also work on the display wall.

2.3.1.2 Model based data visualisation

The display wall prototype was used to model database-based information. The main benefits are high display resolution, which allows you to display a lot of information at the same time, including simultaneously displaying different levels of database (physical, logical, conceptual), and relevant data, and also allows you to navigate through data using "traditional" navigation through data items while navigating in models. The prototype was designed as a versatile tool capable of running on various operating systems as well as browsing various types of databases. Particularly emphasizes that the method used allows browsing data from several related databases

at the same time (including databases may be in different database management systems). This option is not offered even in the tools developed by world's leading companies.

2.3.1.3 Visualization of large-scale graphs

The second approbation of the display wall was to capture a large graph (over 400 vertices) that illustrates the interconnection of the testing terminology. The concepts of the testing (terms), taken from the glossary, serve as graph vertices. Graph edges represent a reference from one term to another. Since glossary contains over 400 different terms then the corresponding graph reaches a considerable size. Such graph visualization, in order for a person to be able to understand the term relationshi, to identify the most important terms referred to by many other terms, can not be realized with a traditional one or two computer screens.

The proposed solution by using a display wall gives many benefits to the graphic representation of the graph: at the same time a large part or even the entire graph of the graphs can be seen, the graph can be scaled out and its individual parts can be viewed immediately, user can arrange graph vertices in the desired configuration. Large companies offer significantly smaller, but more expensive screens to offer analogue graphical visualization capabilities.

The development of the display wall can not be considered complete because interactive collaboration capabilities with display walls needed to be improved, layout must be improved according screen edges, as well as other improvements must be made.

2.3.2. Business process runtime verification

Computerized system analysis and operation correctness evaluation during runtime in operational environment is understood as runtime verification in this research. Correctness evaluation can be done by tools built into system or by system events external monitoring. This research focuses on the last one. Verification is done according to each processes' verification description – model, where is defined events that confirm correctness of each process step, their execution sequence and execution time restrictions.

Results of the resarch of business process runtime verification is published in 6 publications, 5 of them are indexed by Web of Science and 4 of them are indexed by Scopus [DF8, DF9, DF10, DF11], and 2 of them are published in published by IOS Press, but not indexed by Scopus yet [DF12, DF13]. Detailed description of research is available in english report.

Prototype for runtime environment controlling system was developed in the first stage of the research. It fixes runtime environment events and via autonomous agents sent them to the controller. Controller monitors environment events and verifies them according to verification model.

In second stage of the research prototype developed in first stage was used for real life business process verification to measure additional workload to information system

added by runtime verification process. Obtained measurements show, that additional workload for information system is negligible. It shows practical usability of proposed business process runtime verification mechanism.

Research started in first two stages was continued during third stage. All research made by Ivo Oditis in his doctoral thesis defended on 4th of September 2016 was integrated. Additional research of usage of developed methods was continued in two directions:

(1) Define set of systems to whom usage of such system is adequate. As the result of the research class of so called event based system was defined, as well as conditions was defined when models of theese systems are authomatically executable. It opens wide possibilities to use runtime verification in lot of real-world systems.

(2) Run time verification is included in the set of attributes of smart technologies. Smart technologies follows the same goal as IBM's autonomous systems – reduce complexity of system usage by including tools of system runtime support into the system. The main difference between smart technologies and autonomous systems – IBM try to find universal solution, but smart technologies try to reduce complexity by developing realization of some system properties, in this case – run time verification.

During the fourth stage of the research, the works started in the first three stages were continued, developing runtime verification protocols prototype and analyzing three practical applications. Different tasks of the economy, in which the approbation of the runtime verification was performed, have been deliberately chosen. These various examples of use serve as proof that low-resource runtime verification can be used to solve many real problems.

2.3.2.1 Runtime verification of the interbank settlement system

Within the framework of the study, a prototype of the runtime verification solution for the interbank clearing system was created. The main findings of the prototype approbation:

- The proposed multi-agent runtime verification mechanism allows dynamically defining new and modifying pre-defined runtime controls without changing the system being verified. The solution offered by this feature is fundamentally different from traditional, for example, document management system solutions, where verification solutions are planned by the developer and included in the system's functionality.
- Runtime verification is an important component of software quality assurance. With the increasing complexity of the system and its execution environment, it is increasingly important to automatically follow the correctness of the execution of business processes - to verify it during execution.
- The additionl workload caused by the controller and agents did not leave an influence on observed processes; The proposed mechanism provides

verification of processes whose execution rates are even higher than typical business processes.

- Business Process Verification Mechanism should be included as one of the smart technology components for achieving autonomous computing goals.
- The runtime verification solution proposed in the study is applicable to a wider range of problems than originally planned. Although the original goal was to provide business process verifications only, the proposed solution can also be used for a wider range of problems where process descriptions and process events are available.

The results of the research are reflected in 6 publications (DF8, DF9, DF10, DF11, DF12, DF13).

2.3.2.2 Drones flight monitoring

The second runtime verification prototype was created for controlling the drones (RUAV - Rotor Unmaned Autonom Vichle). As you know, flight control of program-driven autonomous equipment is necessary both for analyzing event-based events and for handling unforeseen events that can trigger emergency situations. There are a lot of different events that can cause emergency crashes. It is precisely in such situations that an external, easily configurable runtimetime event analysis mechanism, as developed in this study, is required.

The prototype of run-time verification for drones management was created within the ARTEMIS project R5-COP (Reconfigurable ROS-based Resilient Reasoning Robotic Cooperating Systems), Deliverable Number D34.42 “SIL-based Verification using on-line monitoring”. The development of the dron software was carried out according to the following scheme. First, a dron mission model was created in the MATLAB / Simulink environment, defining a mission using the Extended Finite State Machine (EFSM) formalization. From this base model, using Simulink's latest features, program code was generated that could be transmitted to the dron memory. Before executing programs on a real dron, it was tested in simulation mode in the MATLAB / Simulink environment, in addition to visualizing the dronal flight in virtual reality on the display screen. This allowed to check the operation of the dron control programs without risk of breaking the dron software due to errors.

The runtime verification component was added to the base process as a separate autonomous component that received flight event information from the base process. The steps in the implementation of the runtime verification process were tied to the steps of the underlying process and were executed independently of each other. This implementation mechanism allows you to check the compliance of the basic process with the verification description.

The Dron Flight Analysis in simulation mode confirmed the main findings described in the previous section. In addition, simplicity and transparency of the implementation of runtime verification process was shown. At the same time, the limitations of applying the proposed solution to real-time hardware and simulation model MATLAB / Simulink environment appeared..

The detailed study content can be found in the ARTEMIS project R5-COP (Reconfigurable ROS-based Resilient Reasoning Robotic Cooperating Systems) 2017.year report “Deliverable Number D34.42 SIL-based Verification using on-line monitoring”.

2.3.2.3 Ticket distribution process monitoring.

The theater ticketing system was selected as the third prototype of the run-time verification process, with a run-time control component added. The chosen ticket distribution system has been used for many years in the New Riga Theater. The operation of the system has shown that correct distribution of tickets faces serious problems in situations of unforeseen people reactions or interruptions to the Internet connection occur. A solution to these problems, when the business process is influenced by events outside the information system, was searched in runtime verification.

In order to provide the necessary verification, the verification mechanism was supplemented by databases events verification agents, thus making sure the extension of the runtime verification mechanism. This ensured the validation of existing agent interfaces, and also enabled identification of potential database event logging problems. Basically, they relate to timely identification of changes in record data.

Any significant inconsistencies in the processeas not detected during ticket sales verification. Since the payment of tickets and the shipment of purchased tickets are asynchronous, there may be situations when tickets are sent to the buyer with a certain delay in case of a large system load. In order to avoid confusion, the system owner has determined that the ticket must have been prepared and shipped within a maximum of five minutes. During the verification it was found that in most cases (98%) this condition is also observed.

2.3.3 Performance Indicators

2.3.3.1 Publications (13 publications)

[DF1] Rudolfs Bundulis, Guntis Arnicans. Concept of virtual machine based high resolution display wall. In Information, Electronic and Electrical Engineering (AIEEE), 2014 IEEE 2nd Workshop on Advances in, pp. 1-6. IEEE, 2014. (Web of Science, Scopus)

DOI: 10.1109/AIEEE.2014.7020317, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7020317

[DF2] Rudolfs Bundulis, and Guntis Arnicans. Virtual Machine Based High Resolution Display Wall: Experiments on Proof of Concept. International Conference on Systems, Computing Sciences and Software Engineering (SCSS 14) , Electronic CISSE 2014 Conference Proceedings, 2014. (It was promised that it will be published in Innovations and Advances in Computer, Information, Systems Sciences, and Engineering. LNEE, Springer)

[DF3] Rudolfs Bundulis, Guntis Arnicans, and Rihards Gailums. NVENC Based H.264 Encoding for Virtual Machine Based Monitor Wall Architecture, GPU

Technology Conference, San Jose, Mart 17-20, 2015. http://on-demand.gputechconf.com/gtc/2015/posters/GTC_2015_Visualization_Large_Scale___Multi_Display_01_P5174_WEB.pdf

[DF4] Rudolfs Bundulis, Guntis Arnicans. Use of H. 264 real-time video encoding to reduce display wall system bandwidth consumption. In Information, Electronic and Electrical Engineering (AIEEE), 2015 IEEE 3rd Workshop on Advances in, pp. 1-6. IEEE, 2015. (Web of Science, Scopus)

DOI: 10.1109/AIEEE.2015.7367298, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7367298

[DF5] Ingars Ribners, Guntis Arnicans. Concept of Client-Server Environment for Agent-Based Modeling and Simulation of Living Systems. In Computational Intelligence, Communication Systems and Networks (CICSyN), 2015 7th International Conference on, pp. 83-88. IEEE, 2015. DOI: 10.1109/CICSyN.2015.25 (Web of Science, Scopus)

[DF6] R.Bundulis and G.Arnicans. Conclusions from the evaluation of virtual machine based high resolution display wall system. Springer Verlag, Communications in Computer and Information Science. Volume 615, 2016, pp. 211–225. (Web of Science, Scopus).

DOI**:** 10.1007/978-3-319-40180-5_15

[DF7] Rudolfs Bundulis, Guntis Arnicans. Infiniviz – Virtual Machine Based High-Resolution Display Wall System. IOS Press. Frontiers in Artificial Intelligence and Applications. Volume 291: Databases and Information Systems IX. 2016. pp 225-238. (Web of Science, will be in Scopus)

DOI: 10.3233/978-1-61499-714-6-225

[DF8] Oditis, I., Bicevskis, J. Asynchronous runtime verification of business processes: Proof of concept. International Journal of Simulation: Systems, Science and Technology 16 (6), pp. 6.1-6.11(Scopus).

[DF9] Oditis, I., Bicevskis, J. Document Asynchronous Runtime Verification of Business Processes. Source of the Document - Proceedings - 7th International Conference on Computational Intelligence, Communication Systems and Networks, CICSyN 2015 (Web of Science, Scopus)

[DF10] Bicevska, Z., Bicevskis, J., Karnitis, G. Models of event driven systems. Springer Verlag, Communications in Computer and Information Science. Volume 615, 2016, pp. 83-98. (Web of Science, Scopus)

DOI: 10.1007/978-3-319-40180-5_6

[DF11] Bicevskis, J., Bicevska, Z., Oditis, I. Self-management of information systems. Springer Verlag, Communications in Computer and Information Science. Volume 615, 2016, pp. 167-180. (Web of Science, Scopus).

DOI**:** 10.1007/978-3-319-40180-5_12

[DF12] Janis Bicevskis, Zane Bicevska, Girts Karnitis. Executable Models of Event Driven Systems. IOS Press. Frontiers in Artificial Intelligence and Applications. Volume 291: Databases and Information Systems IX. 2016. pp 101-114. (Web of Science, būs Scopus)

DOI: 10.3233/978-1-61499-714-6-101

[DF13] Janis Bicevskis, Zane Bicevska, Ivo Oditis. Implementation of Self-Management. IOS Press. Frontiers in Artificial Intelligence and Applications. Volume 291: Databases and Information Systems IX. 2016. pp 169-182. (Web of Science, būs Scopus).

DOI: 10.3233/978-1-61499-714-6-169

2.3.3.2 Doctoral thesis (2 thesis)

Supervisor Jānis Bičevskis
Ivo Odītis Business proceses runtime verification (defended 4.09.2016)
Rūdolfs Bundulis (planed to defend in 2018)

2.3.3.3 Master thesis (7 thesis)

Supervisor Ģirts Karnītis
Normunds Pureklis Large-scale data visualization on a high-definition display wall
Jānis Peisenieks Assessment of the living conditions of Riga city using open source data
Vladislavs Maksimčuks Determining favorable weather conditions for paragliding

Supervisor Guntis Arnicāns
Aleksandrs Rilins Efficient web application development for high-definition monitor walls

Supervisor Jānis Bičevskis
Edgars Kirkiļevičs Testing of collaborative processes
Ilga Baiža Use of Priority Mechanism for Testing Analytical Algorithms
Agnese Semjonova Development of Integration Testing Performance Guidelines

Full list of publications

1. Barzdins G., Paikens P., Gosko D. Riga: from FrameNet to Semantic Frames with C6.0 Rules. SemEval 2015 Task 18: Semantic Dependency Parsing. Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), Association for Computational Linguistics, pp. 960–964. (http://www.aclweb.org/anthology/S15-2160)
2. J. Barzdins, E. Rencis and A. Sostaks, Fast Ad Hoc Queries Based on Data Ontologies. In: H.M. Haav, A. Kalja, T. Robal (Eds.), Frontiers of AI and Applications, Vol. 270, Databases and Information Systems VIII, IOS Press, pp. 43-56, 2014.
3. Rudolfs Bundulis, and Guntis Arnicans, Concept of virtual machine based high resolution display wall. Information, Electronic and Electrical Engineering (AIEEE), 2014 IEEE 2nd Workshop on Advances in. pp. 1-6, IEEE, 2014. DOI: 10.1109/AIEEE.2014.7020317, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7020317
4. I.Oditis, J.Bicevskis Asynchronous Runtime Verification of Business Processes 7th International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN'2015) pp. 103-108, 2015
5. Kirikova, M. Enterprise Architecture and Knowledge Perspectives on Continuous Requirements Engineering. Proceedings of REFSQ-2015 Workshops, Research Method Track, and Poster Track co-located with the 21st International Conference on Requirements Engineering: Foundation for Software Quality. Essen, Germany, March 23, 2015. CEUR-WS.org, Vol. 1342, ISSN 1613-0073, pp. 44-51.
6. Bartusevičs, A., Novickis, L., Lesovskis, A. Model-Driven Software Configuration Management and Semantic Web in Applied Software Development. No: Recent Advances in Telecommunications, Informatics and Educational Technologies: Proceedings of the 13th International Conference on Telecommunications and Informatics (TELE-INFO '14) , Turkey, Istambul, December, 2014. Istambul: WSEAS Press, 2014, 108.-116.lpp. ISBN 978-1-61804-262-0.
7. Bartusevičs, A., Lesovskis, A., Novickis, L. Semantic Web Technologies and Model-Driven Approach for the Development and Configuration Management of Intelligent Web-Based Systems. No: Proceedings of the 2015 International Conference on Circuits, Systems, Signal Processing, Communications and Computers, Austrija, Vienna, 15.-17. marts, 2015. Vienna: 2015, 32.-39.lpp. ISBN 978-1-61804-285-9. ISSN 1790-5117.
8. Rudolfs Bundulis, and Guntis Arnicans. Virtual Machine Based High Resolution Display Wall: Experiments on Proof of Concept. International Conference on Systems, Computing Sciences and Software Engineering (SCSS 14), Electronic CISSE 2014 Conference Proceedings, 2014. (in Innovations and Advances in Computer, Information, Systems Sciences, and Engineering. LNEE, Springer)
9. Posters: Rudolfs Bundulis, Guntis Arnicans, and Rihards Gailums. NVENC Based H.264 Encoding for Virtual Machine Based Monitor Wall Architecture, GPU Technology Conference, San Jose, Mart 17-20, 2015.
10. J.Barzdins, E.Rencis, A.Sostaks. Self-service Ad-hoc Querying Using Controlled Natural Language. Submitted to the 12th International Baltic Conference on Databases and Information Systems, Riga, Latvia, 4-6 July 2016

11. G.Barzdins, S. Renals, D. Gosko. (2016). Character-level Neural Translation for NextGen Media Monitoring in the SUMMA Project. Accepted by LREC-2016, 23-28 May 2016, Portorož (Slovenia). (Web of Science)
12. A.Spektors, I. Auzina, R. Dargis, N. Gruzitis, P. Paikens, L. Pretkalnina, B. Saulite. (2016). Tezaurs.lv: the Largest Open Lexical Database for Latvian, Accepted by LREC-2016, 23-28 May 2016, Portorož (Slovenia). (Web of Science)
13. Grundspenkis J. Initial Steps towards the Development of Formal Method for Evaluation of Concept Map Complexity from the Systems Viewpoint. Submitted to the 12th International Baltic Conference on Databases and Information Systems, Riga, Latvia, 4-6 July 2016.
14. Lavendelis E. A Conceptual Approach for Knowledge Structure Update and Learning in Multi-Agent Systems. In Proceedings of 16th International Conference on Applied Computer Science (ACS '16), Istanbul, Turkey, April 15-17, 2016.
15. Petrovica S., Pudane M. Simulation of Affective Student-Tutor Interaction for Affective Tutoring Systems: Design of Knowledge Structure. Submitted to 7th International Conference on Education and Educational Technologies (EET '16), Istanbul, Turkey, April 15-17, 2016.
16. Kirikova, M., Penicina, L., Gaidukovs, A. Ontology based Linkage between Enterprise Architecture, Processes, and Time. ADBIS 2015 short Papers and Workshops BigDap, DCSA, GID, MEBIS, OAIS, SW4CH, WISARD Proceedings. Poitiers - France, September 8-11, 2015. New Trends in Databases and Information Systems. CCIS, Vol. 539. Springer, pp.382-391. In database: (SCOPUS). DOI: http://dx.doi.org/10.1007/978-3-319-23201-0_39
17. Anita Finke. Requirements Inheritance in Continuous Requirements Engineering. Joint Proceedings of REFSQ-2016 Workshops, Doctoral Symposium, Research Method Track, and Poster Track co-located with the 22nd International Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2016), March 14-17, 2016, Gothenburg, Sweden. CEUR-WS.org, Vol. 1564. (SCOPUS)
18. Marite Kirikova. Continuous Requirements Engineering in FREEDOM Framework: a Position Paper. Joint Proceedings of REFSQ-2016 Workshops, Doctoral Symposium, Research Method Track, and Poster Track co-located with the 22nd International Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2016), March 14-17, 2016, Gothenburg, Sweden. CEUR-WS.org, Vol. 1564. (SCOPUS)
19. Dmitrijs Kozlovs, Kristine Cjaputa and Marite Kirikova. Towards Continuous Information Security Audit. Joint Proceedings of REFSQ-2016 Workshops, Doctoral Symposium, Research Method Track, and Poster Track co-located with the 22nd International Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2016), March 14-17, 2016, Gothenburg, Sweden. CEUR-WS.org, Vol. 1564. (SCOPUS)
20. Marite Kirikova, Raimundas Matulevičius, and Kurt Sandkuhl. The Enterprise Model Frame for Supporting Security Requirement Elicitation from Business Processes, submitted to the 12th Baltic Conference on Databases and Information Systems to be held 4-6 July 2016, Riga, Latvia.
21. Bartusevics, A., Lesovskis, A., Novickis, L. Semantic Web Technologies and Model-Driven Approach for the Development and Configuration Management of Intelligent Web-Based Systems. No: Proceedings of the 2015 International

Conference on Circuits, Systems, Signal Processing, Communications and Computers, Austrija, Vienna, 15.–17. marts, 2015. Vienna: 2015, 32.–39.lpp. ISBN 978-1-61804-285-9. ISSN 1790-5117 (Springer, Web of Knowledge).
22. Novickis L., Vinichenko S., Sotnichoks M., Lesovskis A., Graph Models and GeoData Based Web Portal in Cargo Transportation. In : Scientific Journal of Riga Technical University. Applied Computer Systems, 2015/17, RTU Press, Riga, 2015, pp. 34-39. ISSN 2255-8683 (EBSCO, VINITI).
23. Novickis L., Mitasiunas A., Ponomarenko V. Towards Knowledge and Information Technology Transfer Concept and Its Validation. In: Procedia Computer Science, ICTE in Regional Development , 2015 , Valmiera, Elsevier, Volume 77, 2015, Pages 48–55 (SCOPUS).
24. Bartusevics, A., Novickis, L. Model–Based Approach for Implementation of Software Configuration Management Process. In: MODELSWARD 2015. Proceedings of the 3rd International Conference on Model-Driven Engineering and Software Development, France, Angers, 9-11 February, 2015, pp.177-184. ISBN 978-989-758-083-3 (SCOPUS).
25. Rudolfs Bundulis, Guntis Arnicans. Use of H. 264 real-time video encoding to reduce display wall system bandwidth consumption. In Information, Electronic and Electrical Engineering (AIEEE), 2015 IEEE 3rd Workshop on Advances in, pp. 1-6. IEEE, 2015.
26. Ingars Ribners, Guntis Arnicans. Concept of Client-Server Environment for Agent-Based Modeling and Simulation of Living Systems. In Computational Intelligence, Communication Systems and Networks (CICSyN), 2015 7th International Conference on, pp. 83-88. IEEE, 2015. DOI: 10.1109/CICSyN.2015.25 (to appear in SCOPUS)
27. Guntis Barzdins, Didzis Gosko. RIGA at SemEval-2016 Task 8: Impact of Smatch Extensions and Character-Level Neural Translation on AMR Parsing Accuracy. Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), Association for Computational Linguistics, pp. 1143-1147. URL https://aclweb.org/anthology/S/S16/S16-1176.pdf. (to be indexed ACL)
28. Peteris Paikens. Deep Neural Learning Approaches for Latvian Morphological Tagging. Frontiers in Artificial Intelligence and Applications, Volume 289: Human Language Technologies – The Baltic Perspective, I. Skadiņa and R. Rozis (Eds.). IOS Press, 2016, pp 160-166. DOI 10.3233/978-1-61499-701-6-160 URL http://ebooks.iospress.nl/volumearticle/45531 (SCOPUS)
29. N. Gruzitis and G. Barzdins. The role of CNL and AMR in scalable abstractive summarization for multilingual media monitoring. Controlled Natural Language, Controlled Natural Language 5th International Workshop, CNL 2016, Davis, Brian, Pace, Gordon J., Wyner, Adam (Eds.), LNAI, Volume 9767, pp. 127-130, Springer 2016. doi = "10.1007/978-3-319-41498-0" (SCOPUS)
30. J.Barzdins, M.Grasmanis, E.Rencis, A.Sostaks, A.Steinsbekk, Towards a More Effective Hospital: Helping Health Professionals to Learn from their Own Practice by Developing an Easy to use Clinical Processes Querying Language. // J.E.Q. Varajão et. al. (Eds.), Procedia Computer Science, Vol. 100, Elsevier, pp. 498-506, 2016.
31. A.Kalnins, J.Barzdins, Metamodel Specialization for DSL Tool Building. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Databases and Information Systems, 12th International Baltic Conference, DB&IS 2016,

Riga, Latvia, July 4-6, 2016, Proceedings, Communications in Computer and Information Science Vol. 615, Springer, pp.68-82, 2016. (SCOPUS)
32. A.Sprogis, DSML Tool Building Platform in WEB. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Databases and Information Systems, 12th International Baltic Conference, DB&IS 2016, Riga, Latvia, July 4-6, 2016, Proceedings, Communications in Computer and Information Science Vol. 615, Springer, pp.99-109, 2016. (SCOPUS)
33. A.Kalnins, J.Barzdins, Metamodel specialization for graphical modeling language support. // In: Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems. ACM, pp.103-112, 2016. (SCOPUS)
34. J.Barzdins, M.Grasmanis, E.Rencis, A.Sostaks, J.Barzdins, Ad-Hoc Querying of Semistar Data Ontologies Using Controlled Natural Language. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Frontiers of AI and Applications, Vol. 291, Databases and Information Systems IX, IOS Press, pp. 3-16, 2016. (SCOPUS), http://ebooks.iospress.com/volumearticle/45695
35. A.Kalnins, J.Barzdins, Metamodel Specialization for Diagram Editor Building. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Frontiers of AI and Applications, Vol. 291, Databases and Information Systems IX, IOS Press, pp. 87-100, 2016. (SCOPUS), http://ebooks.iospress.com/volumearticle/45702
36. A.Sprogis, ajoo: WEB Based Framework for Domain Specific Modeling Tools. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Frontiers of AI and Applications, Vol. 291, Databases and Information Systems IX, IOS Press, pp. 115-126, 2016. (SCOPUS), http://ebooks.iospress.com/volumearticle/45704
37. Bicevska, Z., Bicevskis, J., Karnitis, G. Models of event driven systems. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Databases and Information Systems, 12th International Baltic Conference, DB&IS 2016, Riga, Latvia, July 4-6, 2016, Proceedings, Communications in Computer and Information Science Vol. 615, Springer, pp. 83-98, 2016. (SCOPUS)
38. Bicevskis, J., Bicevska, Z., Oditis, I. Self-management of information systems. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Databases and Information Systems, 12th International Baltic Conference, DB&IS 2016, Riga, Latvia, July 4-6, 2016, Proceedings, Communications in Computer and Information Science Vol. 615, Springer, pp. 167-180, 2016. (SCOPUS).
39. R.Bundulis and G.Arnicans. Conclusions from the evaluation of virtual machine based high resolution display wall system. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Databases and Information Systems, 12th International Baltic Conference, DB&IS 2016, Riga, Latvia, July 4-6, 2016, Proceedings, Communications in Computer and Information Science Vol. 615, Springer, pp. 211–225, 2016. (SCOPUS).
40. Rudolfs Bundulis, Guntis Arnicans. Infiniviz – Virtual Machine Based High-Resolution Display Wall System. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Frontiers of AI and Applications, Vol. 291, Databases and Information Systems IX, IOS Press, pp. 225-238, 2016. (SCOPUS)
41. Janis Bicevskis, Zane Bicevska, Girts Karnitis. Executable Models of Event Driven Systems. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Frontiers of AI and Applications, Vol. 291, Databases and Information Systems IX, IOS Press, pp. 101-114, 2016. (SCOPUS).

42. Janis Bicevskis, Zane Bicevska, Ivo Oditis. Implementation of Self-Management. // In: G.Arnicans, V.Arnicane, J.Borzovs, L.Niedrite (Eds.), Frontiers of AI and Applications, Vol. 291, Databases and Information Systems IX, IOS Press, pp. 169-182, 2016. (SCOPUS).
43. Grundspenkis J. (2016). Historical Retrospection on Success and Failures during the Development of Concept Map Based System IKAS. Proceedings of the 7th International Conference on Concept Mapping, Tallinn, Estonia, September 5-9, 2016, Vol. 2, pp. 113-119.
44. Grundspenkis J. (2016). Towards the Formal Method for Evaluation of Concept Map Complexity from the Systems Viewpoint. Databases and Information Systems IX, G. Arnicans et al. (Eds.), Frontiers in Artificial Intelligence and Applications, Vol. 291. IOS Press, pp. 341-354. (SCOPUS).
45. Jurenoks A., Novickis L. Adaptive Method for Assessing the Life Expectancy of a Wireless Sensor Network in Smart Environments Applications. In: Proceedings of 14th IFAC INTERNATIONAL CONFERENCE on PROGRAMMABLE DEVICES and EMBEDDED SYSTEMS PDES 2016, Brno University of Technology: 2016, pp.93-98.(IEEExplore)
46. Jurenoks A., Novickis L. Simulation-Based Experimental Research of Wireless Sensor Network Life Expectancy Reconfiguration Method in Transport Logistics Area. In: 2nd International Conference on Systems Informatics, Modelling and Simulation, Riga: 2016, pp.135-140. ISBN 978-1-5090-2693-7. (IEEExplore)
47. Businska L., Kirikova M., The Goal-Based Selection of the Business Process Modeling Language. 9th IFIP WG 8.1. Working Conference, PoEM 2016, Skövde, Sweden, November 8-10, 2016. Vol. 267 of the series Lecture Notes in Business Information Processing, pp 307-316. In database: SpringerLink. DOI: https:/doi.org/10.1007/978-3-319-48393-1_22
48. Kirikova, M. Towards Framing the Continuous Information Systems Engineering. Joint Proceedings of the BIR 2016 Workshops and Doctoral Consortium, co-located with 15th International Conference on Perspectives in Business Informatics Research (BIR 2016), September 14–16, 2016. Prague, Czech Republic. B. Johansson and F. Vencovský (Eds.). Managed Complexity. CEUR-WS.org, Vol. 1684.
49. Kirikova, M., Matulevičius, R., Sandkuhl, K.Application of the Enterprise Model Frame for Security Requirements and Control Identification. Databases and Information Systems IX, G. Arnicans et al. (Eds.). IOS Press. 12th International Baltic Conference, DB&IS 2016, Riga, Latvia, July 4-6, 2016, pp.129-142. (SCOPUS)
50. Lavendelis E. A Cloud Based Knowledge Structure Update and Machine Learning Framework for Heterogeneous Multi-Agent Systems. In: International Journal of Artificial Intelligence Vol 14 (2), October, 2016 CESER Publishing, pp. 157-170. **(Indexed in SCOPUS, SNIP: 1,159)**
51. Petrovica, S., Pudane, M. Emotion Modeling for Simulation of Affective Student-Tutor Interaction: Personality Matching. International Journal of Education and Information Technologies, 2016, Vol.10, pp. 159.-167. (Web of Science)
52. Peņicina L. Controlling Business Object States in Business Process Models to Support Compliance. PoEM 2016, Doctoral Consortium, Skövde, Sweden, November 8-10, 2016. (Ceur-ws.org)

53. Kozlovs, D., Kirikova, M. Auditing Security of Information Flows. Proceedings Perspectives in Business Informatics Research. 15th International Conference, BIR 2016, Prague, Czech Republic, September 15-16, 2016, Vol. 261 of the series Lecture Notes in Business Information Processing. Springer, pp. 204-219. (SCOPUS)
54. Grocevs A., Prokofjeva N. The Capabilities of Automated Functional Testing of Programming Assignments. In: Procedia – Social and Behavioral Sciences, Vol. 228, 2016, pp. 457-461. (Web of Science)
55. A.Kalnins, J.Barzdins. Metamodel Specialization for Graphical Language Support. To be published in the journal of "Software and System Modeling (SoSyM)" **(Scopus SNIP>1).**
56. G. Barzdins, R. Liepins, P. F. Barzdins, D. Gosko. dBaby: Grounded Language Teaching through Games and Efficient Reinforcement Learning. NIPS 2017 Workshop on Visually-Grounded Interaction and Language (ViGIL), 2017, Pieejams: https://nips2017vigil.github.io/papers/2017/dBaby.pdf
57. N. Gruzitis, D. Gosko, G. Barzdins RIGOTRIO at SemEval-2017 Task 9: Combining machine learning and grammar engineering for AMR parsing and generation Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval), 2017. http://www.aclweb.org/anthology/S17-2159 (SCOPUS)
58. P. Paikens, G. Barzdins, A. Mendes, D. Ferreira, S. Broscheit, M. S. C. Almeida, S. Miranda, D. Nogueira, P. Balage, A. F. T. Martins SUMMA at TAC Knowledge Base Population Task 2016 Proceedings of the 9th Text Analysis Conference (TAC), 2017. https://tac.nist.gov/publications/2016/participant.papers/TAC2016.summa.proceedings.pdf
59. Bartusevics, A., Lesovskis, A., Ponomarenko, V. Model-Driven Approach and Library of Reusable Source Code for Automation of IT Operations. Applied Computer Systems. Vol.21, 2017, pp.5-12. (indexed in ISI Web of Science)
60. Novickis, L., Ponomarenko, V., Mitasiunas, A. Information Technology Transfer Model as a Bridge between Science and Business Sector". Procedia Computer Science, Vol.104, 2017, pp.120-126. (indexed in Scopus)
61. Bartusevics, A. Novickis, L., Lesovskis, A. "An Approach for Development of Reusable Function Library for Automation of Continuous Processes", Procedia Computer Science, Vol.104, 2017, pp. 112-119. (indexed in Scopus)
62. Bartusevics, A. Automation of Continuous Services: What Companies of Latvia Says about it?. Procedia Computer Science, Vol.104, 2017, pp.81-88. (indexed in Scopus)
63. Ponomarenko, V., Novickis, L. A Review of Information Technology Transfer Process, Its Topicality, and Related Models. In: Rezekne Academy of Technologies. Rezekne: Rezekne Academy of Technologies, Vol. II, 2017, pp. 128.-132. (indexed in Scopus)
64. Petrovica, S. Multi-level adaptation of an educational game to individual student's gameplay, knowledge and emotions. Proceedings of the 9th Annual International Conference on Education and New Learning Technologies, 2017, pp. 2220-2230. (ISI Web of Science)
65. Pudane, M. Classification of Agent-Based Models from the Perspective of Multi-Agent Systems. Proceedings of the 5th IEEE Workshop on Advances in

Information, Electronic and Electrical Engineering AIEEE'2017, Latvia, Riga, November 24.-25., 2017. (Scopus)

66. Pudane, M. Affective Multi-Agent System for Simulating Mechanisms of Social Effects of Emotions. Proceedings of Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos, 2017, pp. 129-134. (Scopus)
67. Ponomarenko V., Novickis L. Sustainable development and improvement of the Semantic Web Portal. International Journal of Computers, Volume 2, 2017, pp. 74–79.

Full list of doctroal thesis

1. Armands Šlihte. Integrētā priekšmetiskās vides modelēšana: pieeja un rīku kopa topoloģiskā funkcionēšanas modeļa iegūšanai. (Aizstāvēts RTU promocijas padomē P-07 8.06.2015)
2. Pēteris Rudzājs. Development of Education Demand and Offer Information Monitoring System's Model. Defended RTU Promotion Council P-07, September 2015, Scientific supervisor - prof. M. Kirikova.
3. Artūrs Bartusevičs. The Development and Implementation of Model-Driven Software Configuration Management Solutions. Defended RTU Promotion Council P-07, September 2015, Scientific supervisor - prof. L.Novickis
4. Raita Rollande "Individualizētas studiju plānošanas kā pedagoģiskā moduļa sastāvdaļas izpēte un realizācija"/ The Research and Implementation of Personalized Study Planning as a Component of Pedagogical Module." Defended at RTU Promotion Council P-07 May 2015. Scientific supervisor Dr. habil. sc. ing., profesors J. Grundspeņķis.
5. Ivo Odītis. Business process run-time verification. Defended at UL Promotion Council, September 2016, Scientific supervisor: J. Bičevskis
6. Aleksejs Jurenoks. Development and Estimation of Wireless Sensor Network Life Expectancy Assessing Model And Methods. Defended at RTU Promotion Council P-07, June 2016, Scientific supervisor: L. Novicki
7. Sergejs Rikačovs (supervisor prof. Jānis Bārzdiņš) dissertation (planned to submit in 2018)
8. Pēteris Paikens, "RĪKU KOPA LATVIEŠU VALODAS SEMANTIKAS ANALĪZEI" // "Toolset for semantic analysis of Latvian language" Defended at UL Promotion Council, December 2017, Scientific supervisor: G.Bārzdiņš
9. Uldis Locāns, "Future Processor Hardware Architectures for the Benet of Precise Particle Accelerator Modeling" Defended at UL Promotion Council, October 2017, Scientific supervisor: G.Bārzdiņš

Full list of master thesis

1. Svetlana Otto. Daudzaģentu sistēmu un ontoloģiju projektēšanas metožu integrēšanas metodoloģiju analīze (Zinātniskais vadītājs prof. J.Grundspeņķis. Aizstāvēts RTU DITF 15.06.2015)
2. Emil Bikjanov. Zināšanu bāzu un deduktīvo datu bāzu salīdzināšanas analīze (Zinātniskais vadītājs prof. J.Grundspeņķis. Aizstāvēts RTU DITF 15.06.2015)
3. Andrejs Gaidukovs. Laika dimensija biznesa procesu un uzņēmumu arhitektūras modelēšanā (Zinātniskā vadītāja prof. M.Kirikova. Aizstāvēts RTU DITF 16.06.2015)

4. Edgars Kirkiļevičs. Sadarbojošos procesu testēšana (Zinātniskais vadītājs prof. J.Bičevskis. Aizstāvēts LU DF 02.06.2015)
5. Dace Damberga. Uzraudzītas mašīnmācīšanās klasifikatoru izpēte un empīriska salīdzināšana (Zinātniskais vadītājs prof. Guntis Bārzdiņš. Aizstāvēts LU DF 05.06.2015)
6. Zane Siliņa. Latviešu valodas sintaktiskā analizatora 'Čankeris' modernizācija (Zinātniskais vadītājs prof. Guntis Bārzdiņš. Aizstāvēts LU DF 01.06.2015)
7. Andrejs Gaidukovs. Time Dimension in Business Process and Enterprise Architecture Modelling. Master Thesis defended June 2015. Scientific supervisor - prof. M. Kirikova.
8. Artūrs Znotiņš, Jēdzientelpas uun to pielietojumi. // Word embeddings and their applications. (2nd place in Thesis competition "ZIBIT 2016") (Advisor: prof. Guntis Bārzdiņš. Defended at UL FC 06.2016)
9. Roberts Darģis, Liela apjoma datu kopu klasterēšanas algoritmi. // Clustering algorithms for large scale data sets. (Advisor: prof. Guntis Bārzdiņš. Defended at UL FC 06.2016)
10. Reinholds Pīrāgs, Automātiska teksta konspektēšana izmantojot jēdzientelpu. // Automatization of Text Summarization. (Advisor: prof. Guntis Bārzdiņš. Defended at UL FC 06.2016)
11. Aleksandrs Riļins, Efektīva tīmekļa lietotņu izstrāde augstas izšķirtspējas monitoru sienām. // Web application effective development for high-resolution monitor walls. (Advisor: prof. Guntis Arnicāns. Defended at UL FC 06.2016)
12. Normunds Pureklis, Liela apjoma datu vizualizācija uz augstas izšķirtspējas displeju sienas. // Big data visualization on the high-resolution display wall. (Advisor: prof. Ģirts Karnītis. Defended at UL FC 06.2016)
13. Elīna Kauķe, Testēšanas modeļa izveide testējamās sistēmas uzlabošanai. // Testing model development to improve the system under the test. (Advisor: prof. Jānis Bičevskis. Defended at UL FC 06.2016)
14. Jānis Peisenieks, Rīgas pilsētas dzīvojamības novērtēšana, izmantojot atvērtus datu avotus. // Livability evaluation of Riga city using open data. (Advisor: prof. Ģirts Karnītis. Defended at UL FC 06.2016)
15. Vladislavs Maksimčuks, Labvēlīgas laikapstākļu prognozes noteikšana lidošanai ar paraplānu. // Analysis and display of weather forecasts in iOS application. (Advisor: prof. Ģirts Karnītis. Defended at UL FC 06.2016)
16. Parshu Ram Dhungyel. Dzongkha valodas vārdu segmentēšanas metožu analīze. // Analysing the Methods of Segmentation of Dzongkha Words. (Advisor: prof. J. Grundspeņķis. Defended at FCSIT RTU 06.2016)
17. Royal Imanli. Spieta intelekta analīze robotizētām sistēmām un to vadība lietojot nestriktu loģiku.// Analysis of swarm intelligence for robotic systems and their control using fuzzy logic (Advisor: prof. J. Grundspeņķis. Defended at FCSIT RTU 06.2016)
18. Yan Kuchin. Urbumu ģeofizikālo datu interpretācija ar neironu tīkliem. // Interpretation of Drill Hole Geophysical Data with Neural Networks (Advisor: prof. J. Grundspeņķis. Defended at FCSIT RTU 06.2016)
19. Nail Karimov. Uz daudzaģentu pieeju balstītas siltumnīcas vadības un monitoringa sistēmas projektēšana un izstrāde.// Design and Development of Control and Monitoring System for Greenhouses Based-On Multi Agent Approach (Advisor: prof. J. Grundspeņķis. Defended at FCSIT RTU 06.2016)
20. Arjan Chakma. Datu informācijas un zināšanu plūsmas organizāciju informācijas sistēmās. // Data Information and Knowledge Flows in

Organizations Information Systems (Advisor: prof. M. Kirikova. Defended at FCSIT RTU 06.2016)

21. Sangay Dorji. Datu, informācijas un zināšanu modelēšana darbsistēmu tīklos. // Data, Information and Knowledge modeling in work system networks (Advisor: prof. M. Kirikova. Defended at FCSIT RTU 06.2016)
22. Rakesh Parthasarathy. Informācijas loģistikas atsauces modeļa izstrāde. // Development of Information Logistics Reference Model (Advisor: prof. M. Kirikova. Defended at FCSIT RTU 06.2016)
23. Elmārs Sirovatskis. Vispārīga ostas informācijas sistēmas modeļa izstrāde. // Development of Generic Ports Information Systems Model (Advisor: prof. M. Kirikova. Defended at FCSIT RTU 06.2016)
24. Nauris Dorbe, "BEZPILOTU MAŠĪNU APMĀCĪBA IZMANTOJOT STIMULĒTĀS MĀCĪŠANĀS MĀKSLĪGOS DZIĻOS NEIRONU TĪKLUS KOOPERATĪVĀS BRAUKŠANAS SISTĒMĀ" // Driverless cars tuition using reinforcement learning deep neural networks within coorporate driving system. (Advisor: prof. Guntis Bārzdiņš. Defended at UL FC 06.2017)
25. Ilga Baiža Prioritāšu mehānisma izmantošana analītisko algoritmu testēšanā. // Priorities in debugging for analytic algorithms (Advisor: prof. Jānis Bičevskis. Defended at UL FC 06.2017)
26. Agnese Semjonova Integrācijas testēšanas efektivitātes uzlabošanas vadlīniju izstrāde // Development of Guidelines for Integration Testing Efficiency Improvement. (Advisor: prof. Jānis Bičevskis. Defended at UL FC 06.2017)
27. Swathi Christina Rocque. Atbalsts MVU pārejai no tradicionālajām uz mākoņpakalpojumu URP.// Towards Supporting Switching from Traditional System to Cloud ERP for SME Model (Advisor: prof. M. Kirikova. Defended at FCSIT RTU 01.2017)
28. Elza Stepanova. Nepārtrauktas prasību inženierijas metode mobilo lietotņu izstrādei.// Continuous Requirements Engineering Method for Mobile Application Development Model (Advisor: prof. M. Kirikova. Defended at FCSIT RTU 02.2017)
29. Anna Ivanova. Prasību inženierija programmizstrādē ar spējo metodi Scrum.// Requirements Engineering in Agile Scrum Software Development Model (Advisor: prof. M. Kirikova. Defended at FCSIT RTU 06.2017)
30. Ineta Būcena. DevOps ieviešanas iespējas mazām izstrādes grupām.// DevOps Adoption for Very Small Entities (Zinātniskā vadītāja prof. M. Kirikova. (Advisor: prof. M. Kirikova. Defended at FCSIT RTU 06.2017)
31. Idania Lizbeth Rodriguez Moncada. Mācību analītika prasmju attīstīšanas atbalstam.// Learning analytics for skill development support (Advisor: prof. M. Kirikova. Defended at FCSIT RTU 06.2017)
32. Kaspars Purmalietis. Nepārtraukta prasību inženierija iesācējuzņēmumos.// Continuous Requirements Engineering for Start-Ups (Advisor: prof. M. Kirikova. Defended at FCSIT RTU 06.2017)
33. Vitālijs Pavlovs. Grafu algoritmu izmantošana informācijas plūsmu analīzē.// Information Flow Analysis Using Graph Algorithms (Advisor: prof. M. Kirikova. Defended at FCSIT RTU 01.2017)
34. Muhammad Naveez. Audzēja un imunitātes savstarpējo mijiedarbību izskaidrojoša tīkla rekonstrukcija sporādiska kolorektālā vēža audzēja invazīvajā frontē.// Reconstruction of a network accounting for the tumor immunity interaction in the invasive front of sporadic colorectal cancer tumors (Advisor: prof. J. Grundspeņķis. Defended at FCSIT RTU 06.2017)

35. Arnis Staško. Daudzaģentu sistēmas politiski nozīmīgu personu noteikšanai izstrāde.// Multi-agent System Development for Identification of Politically Exposed Persons (Advisor: prof. J. Grundspeņķis. Defended at FCSIT RTU 06.2017)
36. Himanshu Prakashbhai Patel. Daudzaģentu sistēmu efektīvas projektēšanas metodoloģiju novērtēšana.// Evaluation of methodologies for efficent multi agent system projects (Advisor: prof. J. Grundspeņķis. Defended at FCSIT RTU 06.2017)
37. Siyuan Gao. Izkliedēta mākslīgā intelekta realizācijas metožu analīze un novērtēšana.// Analysis and evaluation of methods for implementation of distributed artificial intelligence (Advisor: prof. J. Grundspeņķis. Defended at FCSIT RTU 06.2017)
38. Alīna Vasiļevska. Skaitļošanas intelekta paradigmu lietojums intelektuālo aģentu vadības mehānisma izstrādei.// Application of Computational Intelligence Paradigms in the Development of Intelligent Agent Control Mechanism (Advisor: prof. J. Grundspeņķis. Defended at FCSIT RTU 06.2017)