

Long-term technological prognosis for R&D directions developed in the project

2.1. Ontology-based modeling methods and tools for knowledge extraction and analyses

2.1.1. Development of technologies which would allow domain experts to perform data exploration without involving the third party – an IT experts.

The significance of the problem is well-illustrated by the citation from [1]: “The growth of available information in enterprises requires new efficient methods for data access by domain experts whose ability to analyse data is at the core of making business decisions. Current centralised approaches, where an IT expert translates the requirements of domain experts into Extract-Transform-Load (ETL) processes to integrate the data and to apply predefined analytical reporting tools, are too heavy-weight and inflexible. **In order to support interactive data exploration, domain experts therefore want to access and analyse available data sources directly, without IT experts being involved.**”

There is currently no universally accepted solution for this problem. There are so called business intelligence tools which offer wide possibilities of data analysis, but in the same time they require a significant amount of investment and IT expertise. Needless to say that business intelligence processes involve IT experts whose task is to translate business requirements and queries into a language which is understandable by a computer. There are some tools (e.g. Qlik Tool [2] and SAP Adhoc Query Tool [3]) which provide features for data analysis, but they offer quite a limited set of possible actions.

There have also been attempts to make wrappers for technical querying languages like SQL and SPARQL, e.g. graphical query builders like Graphical Query Designer for SQL Server [4], ViziQuer [5] and Ontology-Based Data Access (OBDA) approach [6], particularly the OptiqueVQS [1] for SPARQL and RDF databases, or form-based tools using wizards and standard GUI elements (e.g. tables and lists) like SAP Quick Viewer SQVI [7]. None of them, however, goes as far as is typically needed in real life situations with data access by non-programmers.

There are, of course, many attempts to use the natural language instead of SQL to formulate queries. The most famous ones are the Google Search Engine and the Wolfram Alpha Computational Knowledge Engine. They have mastered the query analysis for open-domain information extraction use-cases. Such type of querying would be very beneficial also in specialized domains like the hospital management domain. However, such type of queries usually returns a set of answers while the domain experts are typically looking for one correct answer to their specific queries. There are also other

approaches in the field of natural language processing, but none of them has still succeeded to cover deep querying with nontrivial calculations that involve deep inclusive quantifiers.

One approach to cover deep querying with nontrivial calculations is studied in the given project – Self-service ad-hoc querying using **controlled** natural language [8 - 12]. This is an important step towards solving of the above mentioned problem. But the problems remains – how to make this query language more convenient for domain experts. One way how to do this, is to replace a “**controlled** natural language” with a simply “natural language” (i.e. without “controlled”). Until recent time it was not clear how to do this, although in the field of natural language processing there appear some useful techniques for processing unstructured natural language texts (e.g. sentence segmentation, tokenization, part of speech tagging, noun phrase chunking etc.) that could be exploited in the process of developing the query language for specialized domains. But it was insufficient for a general solution of this problem. The problem is that queries in a natural language were not formally understandable (in the sense of semantics). But methods of deep machine learning provide such techniques. It is necessary to have a big enough corpus of queries in natural language. Since there is a hope that by means of deep machine learning it will be possible to “learn” the semantic of a natural language, there is also a hope that it will be possible to understand precisely the queries in a natural language which contain subordinate clauses with relevant conditions of a query. Therefore, there is a hope that in the coming decade it will be possible to find an appropriate solution for the given problem.

References

1. Kharlamov E., Solomakhina N., Ozcep O., Zheleznyakov D., Hubauer T., Lamparter S., Roshchin M., Soylyu A.: How Semantic Technologies can Enhance Data Access at Siemens Energy. In Proc. of International Semantic Web Conference (ISWC'14), pp. 601-619 (2014).
2. Qlik Tool, <http://www.qlik.com>
3. SAP Adhoc Query Tool, <http://scn.sap.com/docs/DOC-45972>
4. Microsoft. Graphical Query Designer User Interface. Retrieved 3 February 2016, <https://msdn.microsoft.com/en-us/library/ms365414.aspx>
5. Zviedris M., Barzdins G. *ViziQuer: A Tool to Explore and Query SPARQL Endpoints*. In *The Semantic Web: Research and Applications*, LNCS, Vol. 6644, pp. 441–445 (2011)
6. Kogalovsky M.R.: *Ontology-Based Data Access Systems*. In: *Programming and Computer Software* 38.4 (2012)
7. SAP. QuickViewer. <http://help.sap.com>
8. Barzdins J., Grasmanis M., Rencis E., Sostaks A., Barzdins J. Self-service Ad-hoc Querying Using Controlled Natural Language. In: 12th International Baltic Conference on Databases and Information Systems 2016, CCIS 615, pp. 18-34 (2016).
9. Barzdins J., Rencis E., Sostaks A. *Data Ontologies and Ad Hoc Queries: a Case Study*. In: *H.-M. Haav, A. Kalja and T. Robal (Eds.), Proc. of the 11th International Baltic Conference, Baltic DB&IS 2014, TUT Press, pp. 55-66 (2014)*.
10. Barzdins J., Rencis E., Sostaks A. *Fast Ad Hoc Queries Based on Data Ontologies*. In: *Frontiers in Artificial Intelligence and Applications*, Vol. 270, Databases and Information Systems VIII, IOS Press, pp. 43-56 (2014).

11. Barzdins J., Grasmanis M., Rencis E., Sostaks A., Barzdins J. *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, IOS Press, pp. 3-16 (2016).
12. Barzdins J., Grasmanis M., Rencis E., Sostaks A., Steinsbekk A. *Towards a more effective hospital: helping health professionals to learn from their own practice by developing an easy to use clinical processes querying language*. In: *International Conference on Health and Social Care Information Systems and Technologies*, *Procedia Computer Science Journal*, Vol. 100 (DOI: 10.1016/j.procs.2016.09.188), pp. 498-506 (2016).

2.1.2. Development of ontology-based and deep-learning methods for natural language semantics extraction

In this project we studied ontology-based and deep-learning methods for natural language semantics extraction. This has led to further insights resulting in visually grounded language learning concept as the most promising further research direction, described our paper:

G.Barzdins et.al. dBaby: Grounded Language Teaching through Games and Efficient Reinforcement Learning. In: *Visually-Grounded Interaction and Language (ViGIL)*, NIPS 2017 Workshop. <https://nips2017vigil.github.io>

2.2. Development of approaches, methods and algorithms for knowledge structure transformations and analysis, and design methodology of semantic network services

2.2.1. Development of methods for knowledge structure merging and decomposition as well as for models' transformations and analysis of models' syntax, semantics and structure

In this research direction, various methods and algorithms for the automatic mutual transformation of the knowledge structures (knowledge representation schemas) will be developed in the next decade. Developed algorithms will be embedded into multi-agent based systems, in which different technologies, including intelligent software agents, personal knowledge management and semantic web, will be integrated into. In addition, the usage of general and problem-specific domain ontologies will provide possibilities to implement heterogeneous, autonomous and affective systems based on distributed artificial intelligence and affective computing methods and tools. The modern intelligent distributed systems in 10 years will be capable to learn based on their experience and extend its knowledge, so ensuring long-term adaptivity to the changes in the environment and in the system itself. Such a long-term adaptivity is named viability. The research in the field of affective computing that has been carried out within SOPHIS project is aimed at tackling currently identified challenges, in particular, long-term interaction among humans and computers. For this purpose, new methods were created and verified

on both user's and system's side to enable advanced user modelling, design of believable artificial agents as well as system's adaption to user's emotions.

Developed methods belong to a set of tools that most probably will be used to enable two main directions:

1. Enhancing virtual environments that will be used in multiple fields, including entertainment, serious games as well as creating fully believable non-playable characters in these environments.
2. Support for people with special needs. With development of smart technologies and current society processes, such as aging and increasing tolerance, it is a matter of time before affective computing methods and tools will be used to support people with both physical and mental illnesses. Moreover, technology can be used to prevent health problems. This application area was highlighted also in 7th International Conference on Affective Computing and Intelligent Interaction (ACII2017), which is held once every two years and is arguably the most influential conference in affective computing. In the future, the smart assistants will obtain physical (robotic) body and thus will be able to assist in everyday activities, for example, teach students who for some reason are not able to attend classes.

The main technologies that will enable acquiring these goals will be:

- deep learning for the affective data processing, in particular, for the affect recognition in various scenarios, e.g., in the wild;
- expressive knowledge representation schemes and models, including ontologies, for representing uncertain and fuzzy knowledge;
- advanced planning algorithms for enabling high-level adaptation in uncertain circumstances.

Such systems will be able to acquire necessary knowledge and process it, as well as to use it for various purposes, e.g., to communicate knowledge between them, for example, with an aim to divide tasks, monitor their execution and evaluate results. In the context of SOPHIS project, two following examples, namely, intelligent tutoring systems and smart houses, illustrate possible future systems.

Intelligent tutoring systems as a research direction will experience a growth in the future since integration of previously mentioned technologies will allow developing autonomous virtual and affective tutoring systems, which will search required learning material in the semantic web based on a problem-domain ontology. Such tutoring systems after processing semantic web will be able to create learning objects and study courses, generate learning content and tasks in an adaptive manner (depending on learner's knowledge level, emotional states, needs, etc.) and carry out knowledge assessment. Thus, an adaptive system for personal knowledge management will be developed based on different type of intelligent agents to provide a support for each learner individually. Virtual tutor will also possess its own personal knowledge management system with an aim to improve teaching effectiveness.

In a production plant, smart home or smart office multi-robot system, which is doing various routine tasks to free humans for more creative activities, can be mentioned as another example. New robots can come into such systems to do new tasks or existing tasks in new ways. Also, some robots can be removed from the system. In case a new robot is added, the system must extend its knowledge structure and knowledge about the capabilities of new robots and new tasks delegated to the system. System is also capable to autonomously organize a workflow that involves robots. As a result, when a system encounters a new task, it is capable to decompose it, find the most appropriate way to perform each of the subtasks and build a workflow for all the subtasks to be executed. Even if the system generates or receives completely new (sub-)task, it is capable to find an appropriate performer for it. The research done in SOPHIS project is one of the first steps to build systems like this.

Despite the fact that majority of intelligent and affective computing systems will be based on deep learning, the neural network based approach has one significant limitation – it does not allow humans to understand the knowledge encoded in the network and adapt it manually if necessary. As a consequence, there will still be a need for possibility to both manually and automatically change the knowledge and its structures. Many applications will still need knowledge structures that can be easily changed. In distributed systems, joint knowledge structure will still be necessary. It will be based on the current notion of ontology, but will be easily adaptable and extendable. Both the system itself and its user/designer can change the knowledge structure.

2.2.2. Continuous requirement engineering

In ten years, the enterprises will have a semi-self updating enterprise knowledge management system that will be based on the FREEDOM framework thus exploiting built in flexibility and handling potential of the internal and external variability. The system will reflect current enterprise at different levels of detail and from different perspectives, as well as provide an insight into enterprise's future alternatives. The system will have cognitive services for external and internal analytics, monitoring and auditing. It will have rich feedback mechanisms that will be related with self-updating mechanisms and recommender systems. The system will continuously maintain the compliance between enterprise hardware, software and social subsystems achieving viable internal socio-cyber-physical ecosystem and promote its viability in the external environment. The system will have interfaces to all employees of the enterprise with well balanced information exposure and security. It will also have regulation analysis services that will allow controlling the compliance between regulations (such as state laws, agreements, internal regulations) and the states of information and physical objects of the enterprise. The system will provide a transparency which will give an opportunity to eliminate internal and external reporting, thus giving an opportunity to use human intelligence mainly only for creative and value adding activities.

2.2.3. Development and integration of different services in Web portal using open Semantic Web resources and enhancement of software configuration management methods

An EAF (Environment-Action-Framework) approach partially developed during VPP project could be improved in many different ways. The influence for improvement process could come from many different trends, which are popular nowadays and will be actual during next 5 – 10 years. One of them is machine learning. Currently EAF approach provides reusable library of automation functions, which could be used to apply different automation workflows. Machine learning could help to automate development and selection process of such functions. Currently, the user should have strong knowledge about problem domain to select needed functions from the library and this is the case where automation should be applied. Combination together with machine learning could automate function selection process and could produce ready to use workflow without mentioned knowledge as well it could be done much faster. Another trend could be an improvement of visual representation and touch screen technology. Currently, during last stages of EAF implementation, users should works with generated source code. After 2 – 3 years, source code could be fully replaced by visual flexible model and could be managed by touch screen. It means that each user will have ability to draw any automation workflow using tablet. After some minutes, automation will be available on as a cloud service. Machine learning will select automatically needed automation functions from the library as well parameters will be set without additional efforts from the user side. If users will need to modify something, no problem, just take a

tablet and make changes on automation picture. Press “regenerate” button and enjoy solution on the cloud.

Overall, EAF will take influence from trends like machine learning, cloud computing and visual programming. People still use reusable things, but mentioned trends could help implement all these things using small and nice tablet produced by Apple. No source code, no powerful laptops with huge HDD or SSD disks or keyboards. Just tablet, where anybody could draw everything, will be implemented and available on cloud from any device at any time.

2.3. Model-based data visualization and business process run-time verification

Rapid development of information technologies has created systems of unprecedented complexity; some authors [1] refer to as „computing systems with complexity approaching boundaries of human ability”. They indicate that the ultimate dream of a pervasive computing – billions of computing systems simultaneously connected to the internet – can quickly become unmanageable and may soon turn into evil “nightmare”. The authors predict even further increase of information systems’ complexity that would almost eliminate human ability to perform software installation, configuration, optimization and maintenance. Solution to this problem certainly lies within prospects of information technologies. In complex systems operations that are too sophisticated to be managed by a regular user should be entrusted to the system itself. This can be executed by implementing respective components into software and setting environment, in which the system is used. IBM has proposed a solution described in its autonomic computing manifesto in 2001. The main statement implies targeted development of information systems that were able to self-management thus overcoming gap between users and increasingly complex world of information technologies.

The manifesto listed four aspects of autonomic computing: Self-configuration, Self-optimization, Self-healing and Self-protection. Achievements of autonomic computing movement during its first decade after publication of the manifesto have been explicitly demonstrated in [2], as well as in [3]. As of now, manifesto’s targets have been met only to some extent. The concept of smart technologies was created by authors [4]-[11], and its main objectives are similar to those of autonomic computing. The approach contains a set of practically applicable improvements of non-functional features to simplify the maintenance and daily use of information systems. As to date smart technologies incorporate version updating, execution environment testing, self-testing, runtime verification and business process execution. The proposed smart technologies cover only part of requirements outlined in the autonomic computing manifesto.

During previous years scientist of Computing faculty in collaboration with IT companies developed and approbated 5 mentioned technologies. Also constructive approach of data quality management is found. It is made precondition for data quality management incorporation into information systems according principles of smart technologies. Information systems security solutions can be made similarly. However, in the light of past research, data quality solutions are preferred.

Today the business working environment is vastly utilizing many computer technologies for the means of increasing the efficiency of presentation and collaboration. There are several distinct kinds of information that either takes up much space when presented or requires much other contextual information to provide full meaning to the user. Thus to efficiently present this information either to

a co-worker or a wider audience large display surfaces like large displays, multi-monitor displays, projectors and projector matrixes are used.

Moreover, such large display surfaces need advanced means of interaction to use their advantages fully. Touch-based devices like tablets, notebooks with touchscreens and even desktops with touchscreens are used to present and share information and interact with bigger presentation surfaces – projectors, monitor walls, etc., Technological limits in bandwidths, resolutions and other areas have reduced during the last years Nevertheless, the issues with large display surfaces remains. The most important problems are stated by Ni et al [12]: 1) truly seamless tiled displays; 2) stereoscopic large high-resolution displays; 3) easily reconfigurable large high-resolution displays; 4) high-performance cluster rendering; 5) scalability; 6) design and evaluate large high-resolution display groupware; 7) effective interaction techniques; 8) perceptually valid ways of presenting information on the large displays; 9) empirical evidence for the benefits of large high-resolution displays; 10) integrating large high-resolution displays into a seamless computing environment.

Bundulis and Arnicans proposed a new architectural solution for building monitor walls [13, 14, 15]. This architecture in contrast to most of the current systems does not host the frame-buffer outside of the client system. Instead, it uses virtualization for the framebuffer implementation. It implements a virtualized GPU that works on top of one or many physical GPUs. This approach allows removing a hard dependency among physical outputs on the physical GPUs and the size of the display surface available. The proposed architecture does this in a manner which hides this fact from the visualization software and sets no specific requirements on it. Any visualization software running on the client system interacts with a virtualized GPU that works just like a normal GPU and exposes all the standard 2D drawing and 3D rendering APIs. Underneath a custom display wall software stack implements these calls by using the physically available GPUs in the system thus allowing efficient scaling by adding more GPUs to the system in the case if the previously described gap between the computing power and rendering power is encountered. The rendered data is then encoded in a video stream that is transmitted over Ethernet to a display endpoint system where it is decompressed and displayed on a connected display or projector. To satisfy the needs of a multi-client environment, the display endpoint can receive different independent streams and display them in a layered mode.

During the next decade, large high-resolution display surfaces will become more interactive (touchscreens, wearable devices for non-contact interacting), easily movable and installable in any place. This display surface will require only electricity and network for communicating with cloud services that will process data and produces content to display it on the display surface.

References

1. Kephart J., Chess D. The Vision of Autonomic Computing. Computer Magazine, IEEE, 2003
2. KEPHART, Jeffrey O. Autonomic computing: the first decade. In: ICAC. 2011. p. 1-2.
3. Lalanda P., McCann J. A., Diaconescu A. Autonomic Computing: Principles, Design and Implementation. Springer, 2013, 288 p.
4. Bičevska Z., Bičevskis J. Smart Technologies in Software Life Cycle. In: Proceedings of Product-Focused Software Process Improvement. 8th International Conference, PROFES 2007, July 2-4, 2007 (Münch, J., Abrahamsson, P., eds.), Riga, Latvia, vol. 4589/2007, 2007. pp.262-272
5. Bičevska Z., Bičevskis J. Application of Smart Technologies in Software Development: Automated Version Updating. In: Scientific papers, vol. 733 (Bārzdiņš, J., Freivalds, R.-M., Bičevskis, J., eds.), University of Latvia, 2008, pp.24 -37.

6. Rauhvargers, K. On the Implementation of a Meta-data Driven Self Testing Model. In: Software Engineering Techniques in Progress (Hruška, T., Madeyski, L., Ochodek, M., eds.), Brno, Czech Republic, 2008, pp.153-166
7. Rauhvargers K., Bicevskis J. Environment Testing Enabled Software – a Step Towards Execution Context Awareness. In: H.-M. Haav, A. Kalja (eds.), Databases and Information Systems, Selected Papers from the 8th International Baltic Conference, vol. 187, IOS Press, (2009), pp. 169–179.
8. Diebelis E., Bičevskis J. Software Self-Testing. In: Proceedings of the 10th International Baltic Conference on Databases and Information Systems, Baltic DB&IS 2012, July 8-11, 2012, Vilnius, Lithuania. IOS Press, vol. 249, 2013, pp. 249 – 262
9. Cerina-Berzina J., Bicevskis J., Karnitis G. Information systems development based on visual Domain Specific Language BiLingva. Selected Papers from the 4th IFIP TC 2 Central and East Europe Conference on Software Engineering Techniques, CEE-SET 2009, Krakow, Poland, LNCS 7054 Springer, 2011, pp. 124-135.
10. Oditis I., Bicevskis J. Asynchronous Runtime Verification of Business Processes. In Proceedings of the 7th International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN), Riga, 2015, pp. 103-108.
11. Bicevskis, J., Bicevska, Z., Rauhvargers, K., Diebelis, E., Oditis, I., Borzovs, J.: A Practitioner's Approach to Achieve Autonomic Computing Goals Baltic J. Modern Computing, Vol. 3 (2015), No. 4, 273-293 (2015)
12. Tao Ni, Greg S. Schmidt, Oliver G. Stadt, Mark A. Livingstonm, Robert Ball, Richard May, A Survey of Large High-Resolution Display Technologies, Techniques, and Applications, VR '06 Proceedings of the IEEE conference on Virtual Reality, IEEE Computer Society Washington, DC, USA ©2006, 223 – 23.
13. Bundulis, R., Arnicans, G.: Architectural and Technological Issues in the Field of Multiple Display Technologies. In : Databases and Information Systems VII: Selected Papers from the Tenth International Baltic Conference, DB&IS 2012, pp. 317-329
14. Bundulis, R., Arnicans, G.: 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, 2014.
15. R. Bundulis and G. Arnicans. Infiniviz – Virtual Machine Based High-Resolution Display Wall System. In: Arnicans, Guntis, et al. (eds.) Frontiers in Artificial Intelligence and Applications. Databases and Information Systems IX, Vol. 291, IOS Press, 2016, 225-238.