

Adopting to Agile Software Development

Gusts Linkevics, *Riga Technical University, Latvia*

Abstract – Agile software development can be made successful, but there is no well-defined way how to achieve this. The problem is that the successful adoption of agile methods and practices is a complex process and this process should be customizable for a particular organization or a team. This research focuses on identification of agile methods and practices. Result of the research is the ranking of agile methods, practices and their usage trends. As some terms of agile software development are interpreted differently in different organizations and teams, terminology used is analyzed. Results of the research can be used as a reference material for those who are planning to adopt agile methods or are looking for the next agile practice to implement. Results will be used for the development of an expert system to support agile adoption.

Keywords – Agile, agile methods, agile practices, expert system, Scrum.

I. INTRODUCTION

Agile software development has become more and more popular during the last decade and offers an opportunity to deal with software development using other approaches rather than traditional development methodologies like waterfall. The need to do things differently arose from problems related to traditional software development. Clients would like to have feedback quicker than it was possible in traditional development. For this reason, an iterative agile methodology was introduced, where it is possible to get feedback more quickly and to make changes when necessary with smaller costs. It could seem that the agile approach could handle most of the problems in software development, but this assumption could not be right as the agile approach itself does not guarantee successful results [3][7].

Companies are trying to adopt the agile approach and in some cases it works, but there are many cases when the agile approach is not working well and organizations are failing using the agile approach. The problem is that organizations and teams are not able to find out what they are doing incorrectly and why the agile method is not working for them. The agile approach looks very good and convenient in theory, but why is it so hard to correctly adopt it? These questions need some answers and an appropriate solution should be found, and this research is the starting step towards answering these questions.

The aims of this paper are to create a list of most promising methods, practices and to determine a list of guidelines other teams and organizations are interested in. This information is required to build a proper expert system, which could help organizations and teams to be more successful with the agile approach.

To achieve the aims, the following tasks have been accomplished:

- Information from conferences about agile development in the period from 2008 to 2012 has been collected and stored in a database [3][7]. The database created contains information about 1,378 articles, 1,257 article identification keywords and 8,735 article-to-keywords associations;
- Terminology issues have been analyzed;
- Actuality index for agile methods has been calculated;
- Agile practice actuality index has been calculated;
- Agile trends have been analyzed;
- Opportunity to use the expert system in agile software development has been described and analyzed.

This paper consists of 7 sections. Section I is the introduction and describes the problem and the aims of the paper. Section II introduces the agile terminology map and analyzes the terminology problem. Section III describes the actuality index of agile methods for the adoption of a more smooth agile approach. Section IV describes creation of the agile practice actuality index as different agile practices should be used in different situations. Section V focuses on trends of agile development by using keyword dynamic analysis by year and Section VI is about the possibility of using the expert system in agile methodology adoption and transformation process. Section VII concludes the paper and gives a brief outline of future research.

II. TERMINOLOGY PROBLEM

Agile software development is used and adopted by many enterprises of different sizes with different teams, which have different levels of knowledge. The point is that various sources use different terminology to describe the same problems. For example, some sources state that Scrum is a method, and other sources suggest that it is a methodology and this is only one example of misleading terminology usage in agile software development that can lead to problems during the adoption of agile methodology. Terminology needs to be analyzed and organized to achieve common understanding of used terms.

It was decided to approach the terminology problem by collecting term definitions from 28 different sources [1][2], [9][10] (see ref. list for a complete list), categorizing them by distinct explanations and visualizing them and their relationships in order to create a common picture of the core terms. Core terms used are “methodology”, “method”, “principle”, “practice” and “procedure” (Fig. 1).

The term “methodology” is described as a “particular discipline or area”, “offers theoretical explanation”, “framework”, “documented process” etc. “Methodology”

consists and is a study of methods, practices and is regulated by various principles.

The term “method” is described by such phrases as “tool of scientific investigation”, “way of solving problem”, “logically ordered plan”, “generalized concept”, “established way” etc. “Method” can consist of procedures which in particular order can lead to reaching of a specific goal.

Term “practice” is well described as “routine”, “habit”, “contrast to theory”, “could be not documented”. Practices can consist of various procedures, which are not generalized and are systematic.

The first collision of terms is between terms “method” and “procedure”. For example, “You must first complete A before proceeding to B and then to C”. Is it a method or procedure? When we say the method of solving this problem is to complete A, then B and then C, it is actually a procedure we are referring to. “Method” thus covers the whole procedure of doing it. The method of solving this problem is to follow this procedure, so the method has a wider scope than the procedure and the procedure is a subset of method. Also as mentioned previously, the method is a generalized concept while the procedure is not generalized [34].

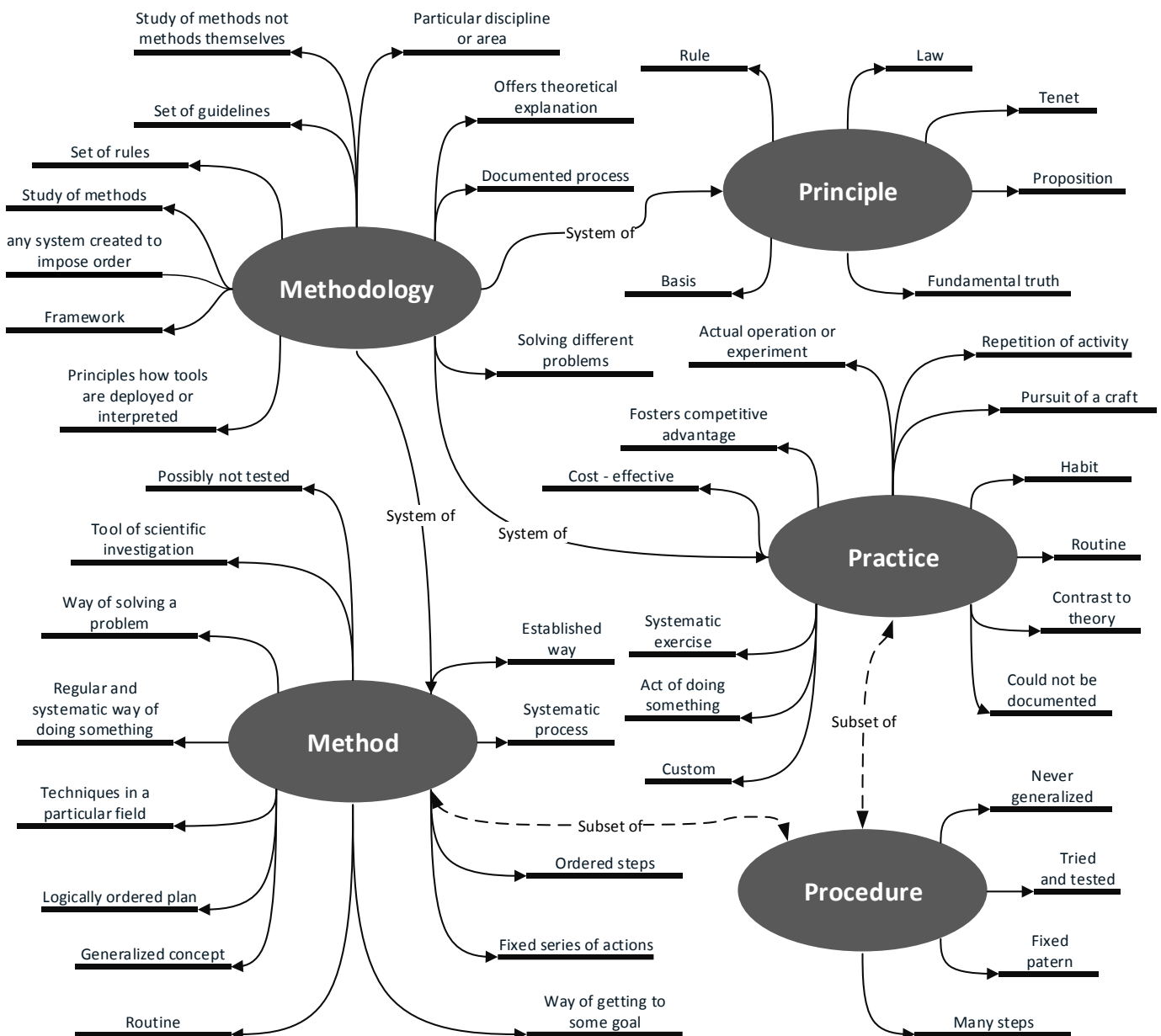


Fig. 1. Core terms and their relations.

The second collision is between the terms “methodology” and “method”. During the last decade, authors started to use “methodology” as a substitute for the term “method”, and it has led to the incorrect perception of terms. Even now authors have been arguing about the correct definition of the terms “methodology” and “method”, and sometimes in scope of one document or article authors use both terms describing the same problem. Decomposition of term definitions to separate phrases and their visualization around terms have allowed one to indicate important trends in distinguishing terms “methodology” and “method”. One such criterion is “methodology is study of methods” [1]. Unfortunately, there are also counter criteria “Any system created to impose order” [13], which can lead to further discussion about the definition. As proper definitions of terms are required for further work on expert system development, it has been decided to settle on a more common definition of the term “methodology” which is defined as “the system of methods and principles used in a particular discipline” [1] and the definition of term “method”, which is defined as “a way of proceeding or doing something, especially a systematic or regular one” [13].

The created terminology map is used in the next sections, where agile methods and practices are analyzed. The next section focuses on agile method actuality index.

III. METHOD ACTUALITY INDEX

Index of actuality (*IOA*) is an indicator that indicates how organizations and developers are interested in a particular agile method, practice or keyword, and index is represented as percentage from all articles published in a particular year and defined by (1), where IOA_m is the index of actuality of the method, A_m is article count, where a particular method is referred to (each article is included only once) and A_y is article count in a particular year.

$$IOA_m = \frac{100A_m}{A_y} \quad (1)$$

Equation (1) is also used for practice and keyword *IOA* calculation, where appropriate article data are used in equation and A_m is replaced by A_p — particular practice referred to during the year or A_k — a keyword referred to during the year.

In accordance with terms described in Section 2, agile methods are Scrum, Extreme programming, Lean, Dynamic System Development Method, Crystal, Feature Driven Development, and Agile Model Driven development.

This section focuses on methods that have the highest *IOA* at conferences about agile methodology hosted by “Agile Alliance” in the period from 2008 to 2012. “Agile Alliance” is the core organization of agile movement and hosts conferences about the agile software development, where business people, researchers and agile method practitioners meet to share their findings; conferences include experience reports, research papers and other findings in the agile field [35]. *IOA* is a good indicator to present what teams are interested in and what problems they are dealing with. This

information can help in transformation and adoption process by indicating which approaches are used by other teams and organizations.

IOA calculation consists of 5 steps. The first step is the creation of an agile method list; the second step is to compare methods in the list against terminology map to verify whether a particular item in the list is a method or not (Fig. 1.). The third step is the creation of a database and software for storing and handling the conference data. The fourth step is the collection of conference data about actuality of a particular method. The fifth step is the ordering and grouping of the data collected.

Data stored about articles include:

- Title of an article;
- Description;
- Year;
- Identified keywords – keyword identification is a manual process where all keywords are identified from context of an article, not from simple appearance;
- Group/Area information;
- URL (Uniform resource locator) of an article.

As a result, the list of method *IOA* is created and represented in Table I.

TABLE I
METHOD *IOA*

	2008	2009	2010	2011	2012
Scrum	18.75	16.78	15.00	13.51	16.82
Extreme Programming	8.16	5.77	3.26	1.93	1.66
Lean	7.91	5.77	7.39	8.69	11.14
Dynamic System Development Method	0.51	0.70	0.00	0.00	0.00
Crystal	1.02	0.35	0.00	0.00	0.00
Feature Driven Development	0.00	0.52	0.22	0.00	0.24
Agile Model Driven Development	0.26	0.00	0.43	0.00	0.00

Results in Table I show that the most common agile method is Scrum followed by a Lean approach. Scrum method has the highest *IOA* during the whole period from 2008 to 2012, while a Lean approach is slowly increasing its index. Extreme programming at the same time is losing its positions year by year. All other methods are very low profile, so it is more reasonable for an organization to adapt to agile development by using the Scrum or Lean approach in conjunction with particular practices. The *IOA* of methods is mostly decreasing during the research period and it is explainable with the age of a particular method, where older methods are used less, because newer exist. Also some important role is in the fact that methods already described in great detail are not widely presented during conferences as information about them is known. Organizations and teams are concentrating their attention now on the agile practices. Practices and their *IOA* are described in more detail in the next section of this paper.

IV. AGILE PRACTICE IOA

Agile methods include various procedures and define a generalized or conceptual way how to achieve a result, but do not guarantee a good result. To achieve better results in agile development, it is recommended using some subset of agile practices. It is the responsibility of teams to decide which practices they would like to follow on a daily basis.

The intension of this section is to analyze the information from “Agile Alliance” conferences about agile practices during the period from 2008 to 2012 and to create *IOA* of practices and analyze dynamics of practices by year. The result of this process is a list consisting of 35 agile practices with highest *IOA* (Table II).

TABLE II
PRACTICE IOA

	2008	2009	2010	2011	2012
Test Driven Development	12.88	10.84	3.04	6.56	5.69
Retrospective	11.48	15.73	3.91	2.70	4.74
Review	8.55	8.22	0.87	1.74	2.61
INVEST	8.42	7.34	0.87	1.16	5.69
Code Refactoring	7.14	5.59	1.96	2.90	3.79
User Stories	6.76	6.12	3.48	4.44	6.64
Continuous Deployment	6.63	8.04	1.30	1.54	1.42
Backlog Usage	6.38	7.34	4.13	1.93	6.64
Pair Programming	6.12	5.77	2.61	3.47	3.08
Measurements	5.36	6.47	3.48	1.54	3.32
Automated Testing	4.08	1.31	0.43	1.54	0.00
Automated Unit Testing	3.95	2.45	1.30	1.16	1.90
Continuous Integration	3.83	6.99	0.87	1.93	1.42
Acceptance Testing	3.57	2.27	1.30	2.32	0.95
Release Planning	3.57	2.27	0.00	0.39	0.95
Iteration Planning	3.32	2.80	1.30	0.00	0.47
Estimation	2.81	1.40	1.74	0.77	2.37
Daily Stand-up Meeting	2.81	1.40	1.30	0.39	0.47
Behavior Driven Development	2.81	3.50	1.74	1.35	1.42
Planning Game	2.30	1.57	0.65	0.00	0.95
Working Software	1.79	2.10	1.30	0.77	1.42
Source Control	1.53	1.40	0.00	0.77	0.47
Active Stakeholder Participation	1.28	2.80	0.43	0.39	1.90
Definition of Done	1.02	0.52	0.22	0.19	2.13
Emergent Design	1.02	1.57	0.00	0.19	1.66
Exploratory Testing	1.02	0.70	0.43	0.77	0.47
Facilitation	0.89	2.10	0.22	0.39	1.18
Cross-functional Team	0.26	1.40	0.00	0.77	3.32

Usability Testing	0.38	1.05	1.30	0.39	0.00
Code Review	0.26	1.05	0.43	0.00	0.47
Story Mapping	0.26	0.00	0.43	0.39	0.95
Sustainable Pace	0.26	0.70	0.43	0.00	0.47
Kanban Board	0.26	0.70	0.43	0.00	0.95
Acceptance Test Driven Development	0.38	0.52	0.00	1.16	1.66
Automated Build	0.38	0.87	0.43	0.00	1.42

Creation of the list consisted of several steps. The first step was the creation of an agile practice list. The initial agile practice list consisted of 176 agile practices and was created using 22 sources. The second step was eliminating duplicate practices from the list as some practices in various sources were defined under different names, but represented the same practice. After merging the list of agile practices, list was reduced to 111 practices and this is the final list used for *IOA* calculation. The third step of analysis consisted of two sub-steps where the first was comparing practice list items against keywords, which were identified and stored in the database during data gathering step and the second part was comparing practice list items against stored article content.

As an outcome, two result sets were generated. The first result represented practices-to-keywords mapping and the second represented practices-to-article content mapping. Both result sets were merged together and, as a result, the list of 70 agile practices was created. As the list of 70 practices is too big to be represented in this paper, it was decided to add selection criteria and represent only the top 30 practices with the highest *IOA* in each year. As some practices from the top 30 in one year are not in top 30 next year, information about them is also included in the final result (Table II), which consists of 35 practices.

Data in Table II clearly indicate that the highest actuality index during 2008 is for Test Driven Development, Retrospective, Review, INVEST (Independent, Negotiable, Valuable, Estimable, Small, Testable) and Code refactoring practices. During 2009, the highest actuality index was for Retrospectives, Test Driven Development, Review, Continuous Deployment, Backlog Usage and INVEST practices. Some of those practices were also among the top practices during 2010, for example, Backlog Usage, Retrospective, Test Driven Development, User Stories and Measurement System practices. During 2011 the situation at the top was very similar, the highest actuality index was for Test Driven Development and User Stories practices. The situation during 2012 was also very similar, practices like User Stories, Test Driven Development, INVEST and Backlog Usage were at the top of the list. Changes were minor and only order varied slightly during the years.

When *IOA* of practices is determined, it is easier to decide which practices to utilize during next sprint or sprints.

Some additional information can be obtained from conferences and this additional information are keywords of articles presented. Keyword *IOA* is described in more details in the next section.

V. KEYWORD IOA

Keyword *IOA* is a good way to determine in what aspects of agile development teams and organizations are interested. Focus of this section is the calculation of previously defined keyword *IOA*.

Keyword data preparation consisted of 3 steps. The first step was the identification of article related keywords, which was a manual process, where keywords were extracted from the context of an article. Totally 1,257 keywords were identified. Each article depending on context could have as many keywords as needed. As some of the identified keywords were very similar in meaning, the second step was required to store additional data about keyword-to-keyword alias relations. The third step was the grouping of all keywords by year and selecting the top 30 keywords from each year. Sometimes keyword dynamics by year was such that a particular keyword was at the top 30 during one year and not in the top 30 during the next period, in such cases data about the keyword were added to each year. This approach allows following the dynamics of the keyword year by year (Table III).

TABLE III
KEYWORD IOA

	2008	2009	2010	2011	2012
Agile Adoption	16.58	21.33	13.48	20.08	23.70
Experience Report	16.33	11.19	2.61	9.65	10.90
Agile Team	16.07	6.29	13.91	11.58	18.01
Practices	14.80	14.69	31.74	20.08	21.33
Testing	14.80	14.34	9.13	7.34	9.00
Leadership	14.54	13.99	12.61	6.56	8.53
Organizational Culture	14.29	13.64	8.26	10.04	13.74
Tools	11.99	11.54	9.57	15.83	8.53
Business Value	10.97	8.74	3.04	3.47	9.95
Customer	10.46	6.64	2.61	6.95	11.85
Distributed Agile	10.20	8.04	7.83	1.93	3.79
Development	9.69	11.54	9.13	6.18	8.53
Learning	8.93	7.69	3.48	12.74	6.16
Large Scale Agile	7.40	6.64	10.00	5.79	8.53
Transition	7.40	6.29	15.22	15.44	16.59
Quality	7.40	4.55	2.61	7.34	7.58
Collaboration	6.89	8.39	7.83	17.37	20.85
Communication	6.89	5.59	5.22	5.41	5.21
Organization	6.63	5.94	11.74	4.25	1.42
Coaching	4.08	11.54	5.22	10.04	12.32
Enterprise	3.57	6.99	16.96	9.27	12.32
User Experience	5.10	6.99	6.96	5.41	8.06
Environment	4.85	6.64	7.83	5.79	7.58
Planning	3.06	6.29	3.04	3.47	4.27
Product Management	0.77	5.94	6.09	0.39	0.47
Requirements	1.79	4.90	10.00	3.86	4.27
Teambuilding	1.02	2.45	6.96	1.93	2.84

Project Management	2.81	2.45	6.96	6.56	0.95
Problems	4.08	4.20	6.52	5.79	5.69
Research	2.55	1.75	5.65	5.79	4.74
Hands on Labs	0.26	2.10	3.04	8.49	6.64
Business	0.51	1.05	1.30	7.34	1.42
Mentoring	0.77	2.10	0.43	6.18	7.58
Innovation	1.53	4.55	2.17	5.79	5.21
Principles	0.00	0.70	2.17	1.93	7.11
Scaling Agile	2.30	5.59	4.35	3.47	6.64
Agile Adoption	16.58	21.33	13.48	20.08	23.70

The final top keyword list contained 37 keywords. Keywords with the highest *IOA* in 2008 were “agile adoption”, “experience report”, “agile team”, “practices” and “testing”. Experience reports are also known as information sharing topics when organizations or developers are sharing their knowledge about agile development. *IOA* for “agile teams” and “testing” is also very high.

The situation in 2009 was quite similar to 2008 as keywords with the highest *IOA* were “agile adoption” followed by “practices” and “testing”. Newcomer to the list was “leadership” and “organizational culture”. Organizational culture in this context is a change process in an organization when it transforms to agile development and struggles with internal cultural changes which are triggered by adapting agile approaches, for example, resistance of the team, business or other people in the organization. Articles about practices are usually concentrating on describing the agile practices, their usage and also possibility to combine them.

The keyword with the highest *IOA* during 2010 was “practices” followed by “enterprise”. Enterprise in this context is enterprise level agile development, where the agile approach is implemented on an enterprise level. New keyword at the top during 2010 was “transition”, which described the transition process to the agile approach. When a decision is made to adopt the agile approach, this adaption is taking some period of time, which is called “transition period”. One more newcomer to the list was “large scale agile”, which described working with large projects. This keyword is also closely linked with “enterprise agile”.

It is not a big surprise with 2011; the first position was taken by “agile adoption” that was followed by “practices”. During 2011, “collaboration”, “learning” and “coaching” increased their *IOA*. “Learning” and “coaching” have been at the top 30 for all previous years which means that those topics are important. More and more agile teams are looking for coaches or training to improve development process and to get improvement as quickly as possible.

Also during 2012, the keywords with the highest *IOA* were “agile adoption” followed by “practices”, “transition”, “agile team” and “organizational culture”.

Having performed the analysis of data on all 5 years, it is noticeable that teams and organizations are interested in the same things during the period specified, just the keyword in focus slightly changes. The keywords “agile adoption” and

“practices” maintain very high *IOA* throughout the period, which indicates the teams and organizations are very interested in adopting agile and finding right practices to follow. *IOA* for “learning” and “coaching” indicates that teams are trying to learn more about the agile process and are open to new knowledge. As agile experts are very expensive and it is almost impossible to hire them for a period, it is advisable to teach someone from an internal team to have knowledge in agile software development or to try using some advisory or expert system, which could collect knowledge about teams, practices, metrics and sprints. This knowledge would allow improving development process and could advise how to deal with some specific situations. The opportunity to use the expert system in agile development is described in the next section.

VI. EXPERT SYSTEM USAGE IN AGILE SOFTWARE DEVELOPMENT

Expert systems are used in different fields and can be used in agile software development as well. As it is described in the previous section “agile adoption”, “practices”, “learning” and “coaching” have high *IOA*. Teams and organizations are constantly searching for expertise in adopting the agile approach. Each team and organization is very different and sometimes things which are working great for one team do not work for another. Even two teams within one organization are different, and teams should find ways which work for them best. The role of the expert system at this point is to be as an advisor or a coach that can give appropriate advice during the agile adoption process or during the transition phase.

When organizations are starting to think about adopting the agile approach, they have many questions:

- Can we handle the agile approach?
- Are we ready for agile development?
- What will change in our development process?
- Are we doing agile development correctly?
- What to do if something does not work?
- Whom can we ask help and advice?
- Is adapting to the agile process complete or are we still in the process of adopting it?
- Are our new team members agile?

This is only a small part of the questions the teams and organizations could have, and the idea is to use expert system to answer those questions.

In the beginning, some areas of expertise for the expert system should be set as it is needed to settle on some scope in which the expert system can work. The scope of the system could be broadened at the following stages of development. It has been decided to implement in the beginning two modules of expertise:

- Organization agility evaluation module – as organizations are very different, and adopting the agile approach or transition process to agile development is not going in the same way for all organizations, it is necessary to evaluate the organization’s agility to suggest the appropriate adoption path. For example, it is possible

that organization’s resistance to change is very high and in such cases it is more advised to use “Kanban” practice, so that an organization does not need to change its daily process completely, but tries to slowly change to a more agile approach. The scope of this module is to deal with such problems as:

- Evaluating organization’s agility to identify organization’s readiness;
 - Identifying problematic areas of organization which need to be changed in order to adapt to the agile approach;
 - Advising appropriate agile practices;
 - Identifying how far we are in adapting the agile approach;
 - Some additional scopes could be added during the creation of the expert system.
- Team agility evaluation module – all teams are not equally agile and depending on the team some specific solutions are needed. The purpose of the team evaluation module is:
 - Evaluating team readiness to adopt agile approach, for example, people who have worked with the waterfall approach are adopting agile with a big resistance;
 - Evaluating new members of the team to understand the level of new members’ agility;
 - Detecting some problematic areas and recommending some specific agile practices;
 - Scope of this module could also be broadened during the creation of the expert system.

Those two modules are only the starting point for the expert system to be developed. Later it is possible to add some additional modules to the system. Possible additional modules are team building, sprint status evaluation and some more. Organization’ agility evaluation and team agility evaluation module implementations will be described in more detail in next research papers as the intention of this paper is only to define how the system could be used.

VII. CONCLUSION

Adopting the agile approach has never been a simple task and in the process of adopting the agile approach various questions arise. This research is the first step towards the creation of the expert system that can help evaluate organizations and teams to allow them to adopt the agile approach more smoothly and to assist during the transition phase.

Common understanding of terminology is required and, therefore, terminology issues have been analyzed and core misleading terms have been identified. A terminology map has been created to check easily the scope of a particular term. The most misleading terms are “methodology” and “method”. The core difference between those terms is that methodology is the study of methods not methods themselves.

When an organization is willing to adopt the agile approach it has to select some of the agile methods. *IOA* of agile methods is calculated and this result can be used when a decision is made to adopt the agile approach. Methods with the highest *IOA* are Scrum and Lean approach. Scrum has very high *IOA* throughout the whole period from 2008 to 2012, and the Lean approach is slowly increasing its *IOA* during the years.

Adoption of the agile method is not the only thing an organization and a team have to deal with. Methods should be used in conjunction with the agile practices. The analysis of conference materials and the calculation of *IOA* of practices have identified that practices with the highest *IOA* are Test Driven Development, Retrospective, Review, INVEST, Code Refactoring, and User Stories. In total, 111 agile practices have been identified and dynamics by year of the top 35 practices have been analyzed. The top 35 of agile practices varies a little during the period from 2008 to 2012, but same practices are at the top 35 each year.

Various aspects of agile development trends can be identified from keyword analysis. Keywords help to identify the dynamics of topics, in which organizations and teams are interested. Keywords with the highest *IOA* are “agile adoption”, “practices”, “learning” and “coaching”.

Expert system can be used during adoption of the agile approach and during the transition process, but is not limited to that usage. It has been decided to start with two modules: one of them will deal with the evaluation of organization’ agility and the second will deal with agile team agility evaluation. Information collected from agile conferences will allow creating basic facts and rules for the expert system development. Agile expert involvement will be required during next steps to add more information to the expert system.

REFERENCES

- [1] S. Anderson, *Collins English Dictionary – Complete and Unabridged*, New York, NY: HarperCollins Publishers, 2003.
- [2] *The American Heritage Dictionary of the English Language*, Boston, MA: Houghton Mifflin Harcourt, 2000.
- [3] Agile Alliance, “Agile Conference 2008,” *Agile Alliance*, [Online]. Available: <http://agile2008.agilealliance.org/>. [Accessed: May. 17, 2012].
- [4] Agile Alliance, “Agile Conference 2009,” *Agile Alliance*, [Online]. Available: <http://agile2009.agilealliance.org/>. [Accessed: May. 19, 2012].
- [5] Agile Alliance, “Agile Conference 2010,” *Agile Alliance*, [Online]. Available: <http://agile2010.agilealliance.org/>. [Accessed: May. 25, 2012].
- [6] Agile Alliance, “Agile Conference 2011,” *Agile Alliance*, [Online]. Available: <http://agile2011.agilealliance.org/>. [Accessed: May. 30, 2012].
- [7] Agile Alliance, “Agile Conference 2012,” *Agile Alliance*, [Online]. Available: <http://agile2012.agilealliance.org/>. [Accessed: June. 10, 2012].
- [8] Reverso, [Online]. Available: <http://dictionary.reverso.net/english-definition/methodology> [Accessed: June. 20, 2013]
- [9] Oxford Dictionaries, [Online]. Available: <http://oxforddictionaries.com/definition/english/methodology> [Accessed: June. 22, 2013].
- [10] Oxford Dictionaries, [Online]. Available: http://oxforddictionaries.com/definition/american_english/methodology. [Accessed: June. 22, 2013].
- [11] Yahoo Answers, [Online]. Available: <http://answers.yahoo.com/question/index?qid=20070819065707AAfHjsa> [Accessed: June. 22, 2013].
- [12] Business Dictionary, [Online]. Available: <http://www.businessdictionary.com/definition/methodology.html> [Accessed: June. 22, 2013].
- [13] The Free Dictionary, [Online]. Available: <http://www.thefreedictionary.com/methodology> [Accessed: August. 12, 2013].
- [14] Merriam Webster, [Online]. Available: <http://www.merriam-webster.com/dictionary/methodology> [Accessed: June. 22, 2013].
- [15] Wikipedia, [Online]. Available: <http://en.wikipedia.org/wiki/Methodology> [Accessed: June. 22, 2013].
- [16] Wikipedia, [Online]. Available: http://en.wikipedia.org/wiki/Software_development_methodology [Accessed: June. 22, 2013].
- [17] Wikipedia, [Online]. Available: <http://en.wiktionary.org/wiki/methodology> [Accessed: June. 22, 2013].
- [18] Dictionary, [Online]. Available: <http://dictionary.reference.com/browse/methodology> [Accessed: June. 22, 2013].
- [19] Microsoft Developer Network, [Online]. Available: <http://msdn.microsoft.com/en-us/library/dd997578.aspx> [Accessed: June. 22, 2013].
- [20] The Free Dictionary, [Online]. Available: <http://www.thefreedictionary.com/method> [Accessed: June. 22, 2013].
- [21] Vocabulary.com, [Online]. Available: <https://www.vocabulary.com/dictionary/method> [Accessed: June. 22, 2013].
- [22] Macmillan Dictionary, [Online]. Available: <http://www.macmillandictionary.com/dictionary/american/method> [Accessed: June. 22, 2013].
- [23] Merriam Webster, [Online]. Available: <http://www.merriam-webster.com/dictionary/method> [Accessed: June. 22, 2013].
- [24] Answers, [Online]. Available: http://wiki.answers.com/Q/What_is_the_difference_between_a_method_and_a_procedure [Accessed: June. 22, 2013].
- [25] Business Dictionary, [Online]. Available: <http://www.businessdictionary.com/definition/method.html> [Accessed: June. 22, 2013].
- [26] Wiktionary, [Online]. Available: <http://en.wiktionary.org/wiki/method> [Accessed: June. 22, 2013].
- [27] Dictionary.com, [Online]. Available: <http://dictionary.reference.com/browse/method> [Accessed: June. 22, 2013].
- [28] Oxford Advanced Learner’s Dictionary, [Online]. Available: <http://oald8.oxfordlearnersdictionaries.com/dictionary/method> [Accessed: June. 22, 2013].
- [29] Yahoo Education, [Online]. Available: <http://education.yahoo.com/reference/dictionary/entry/method> [Accessed: June. 22, 2013].
- [30] Collins, English Dictionary, [Online]. Available: <http://www.collinsdictionary.com/dictionary/english/method> [Accessed: June. 22, 2013].
- [31] Wiktionary, [Online]. Available: <http://en.wiktionary.org/wiki/practice> [Accessed: June. 22, 2013].
- [32] Dictionary.com, [Online]. Available: <http://dictionary.reference.com/browse/practice> [Accessed: June. 22, 2013].
- [33] Greenspun.com, [Online]. Available: http://greenspun.com/bboard/q-and-a-fetch-msg.tcl?msg_id=00CHDn [Accessed: June. 22, 2013].
- [34] Wiki Answers, [Online]. Available: http://wiki.answers.com/Q/What_is_the_difference_between_a_method_and_a_procedure [Accessed: June. 22, 2013].
- [35] Agile Alliance, [Online]. Available: <http://www.agilealliance.org/>. [Accessed: May. 10, 2014].

Gusts Linkevics, Riga Technical University, Master Degree in Business Administration, 2005. Riga Technical University, Engineer Degree in Computer Sciences, 2003. Riga Technical University, Bachelor Degree in Computer Sciences, 2001.

If P&C Insurance AS Latvia, Republikas sq. 2A, SENIOR DEVELOPER. Nexum Insurance Technologies, SENIOR DEVELOPER. Tilde Ltd., LEAD DEVELOPER. Researching agile software development.

Member of Latvian .NET user group. Member of Microsoft Student Partners. Microsoft Certified Solution Developer, Microsoft Certified Application Developer, Microsoft Certified Professional.

Phone: +371 29417442; E-mail: gusts.linkevics@rtu.lv