Has Time Stood Still in Requirements Engineering?

Richard Berntsson Svensson
Department of Computer Science, Lund University, Sweden
rbsv@cs.lth.se

Introduction

Since the inception of requirements engineering in the 1970s, most requirements engineering research has been devoted to techniques of modeling and specification of requirements [1]. Also, since the 1980s we have been doing requirements acquisition where we have developed dozens of different techniques, each shown to be effective in different situations. And yet, most requirements analysts do not use them, instead we keep using the same old techniques in projects to elicit, acquire, and discover our customer's requirements. However, since the 1970s and 1980s, a major shift is currently taken place in how requirements engineering tasks needs to be approached. This shift is not only related to changes in globalization and distribution of software development; web services and virtualization of application services, but also to changes in economic activities and the merge of traditional industries and technology industries. In this 'new' world, we need to empirically investigate how creative elements and creativity theories (creativity focuses on the generation of novel and useful ideas) applies to RE, which may lead to more innovative products (innovation is defined as the practical application of the results from creativity thinking), and it is important that it is based on real world challenges and that it scales.

The merge of industries is described in Forbes Magazine [2]:

"Ford sells computers-on-wheels. McKinsey hawks consulting-in-a-box. FedEx boasts a developer skunkworks. The era of separating traditional industries and technology industries is over—and those who fail to adapt right now will soon find themselves obsolete"

Another example of an industrial change has occurred in the medical domain. Before 2010, medical devices were primarily hardware (perhaps with some embedded software). However, based on the 2010 Medical Device Directive [3], software applications are classified as active medical devices. Regarding the evolution of economic activities:

"Creativity will be the next economic activity, replacing the current focus on information" - Nomura Research Institute

According to the Nomura Research Institute, Japan, the main stages of human civilization are agricultural, industrial, informational, and creative. Just as the industrial revolution replaced agriculture as the dominant economic activity, the 'creativity age' is replacing the 'information age' as the next dominant global economic focus [4]. *"This is the age of creativity because the subtext of global competition is increasingly about a nation's ability to mobilize its ideas, talents, and creative organizations"* [4].

Creativity in Requirements Engineering

So, was the Nomura Research Institute right in their prediction? Well, there are several indications and 'evidences' that supports the Nomura Research Institute's prediction. *First*, in 2009 the NEST report [7] was released. According to the NESTA report, 78% of 850 respondents reported that innovation was 'very' or 'extremely important' to their organizations, and 58% of the respondents indicated that the importance of innovation has increased over the past 12 months as a result of the recession. Policymakers have recognized that that to remain competitive and to survive in the unprecedented economic, companies need to encourage innovation [7]. *Second*, it seems to be difficult to find a market domain in which the innovation does not depend on software, hence, where software, traditionally seen as secondary in many sectors, has moved to a center-stage. This is supported by, e.g. [5], [8], which reports that most of a products' innovation (requirements are the key abstraction that encapsulates the results of creative thinking about the vision of an innovative product) lies in its software components. And *third*, based on my own interactions with 20-odd companies in Sweden in the past few years, creative thinking and innovation stands out as an extremely important challenge from small startup companies to large multinational companies in Sweden.

The shift of information technology organizations toward the creative sector and companies striving to design innovative products that combine and use existing technologies in unanticipated ways seems to be an important real-world need. Although the importance of aligning creativity in requirements engineering is argued, both by empirical evidence and in [9], and that creativity has received more attention in requirements engineering research in the last couple of years, relatively little requirements engineering research has addressed creativity [9] despite its importance for the industry. Very little is known about how current requirements engineering processes support creativity practice, and according to Maiden and Robertson [10], there is a lack of creativity theories and models in current requirements engineering research and practice.

An even more interesting challenge lies in combining creative thinking with agile development. In creative thinking, incubation and illumination are the most important phases. Incubation handles complexity; during this relaxing period, the designers unconsciously and consciously combines ideas with freedom that denies linear and rational thought. During the subsequent and shorter illumination phase, a creative or innovative idea suddenly emerges, often at the most unlikely time in the most unlikely place. However, there are many barriers to creativity, for example, by placing stress on the value of efficiency, effectively, moneymaking, and the organizational climate [11]. One obvious weakness, in terms of supporting creativity, in agile development methods is not to exploit working software for creative thinking about new requirements and opportunities. Indeed, the short durations of sprints with a focus on delivering efficiently and effectively may hinder creative thinking.

I do believe that requirements engineering already today is a highly creative process. However, I also believe that economic and market trends imply that requirements engineering will have to become **significantly more** creative to realize the potential of future software applications and highly productive software organizations. Unfortunately, current work on requirements engineering does not recognize the importance of creative thinking in requirements engineering.

Scalability

The complexity and size of software-intensive systems continues to increase. Not only have there been major changes since the 1970s, e.g. in the transition from hardware, mechanics and manual labor to software solutions, but we have also moved from the 'information age' to the 'creativity age', which in turn adds to the increasingly large and complex sets of requirements. It is difficult to conceive of any organization in any sector that is not dependent on software. This dramatic change in importance of software is a major challenge for organizations. Hence, scaling software in a controlled and efficient way may become a crucial competitive advantage. How many requirements can an industrial system development organization manage with available requirements engineering processes, methodology, techniques and tools? This is hard to know as requirements engineering research often falls short in characterizing the scalability of proposed methods [14].

One interesting characteristic of requirements engineering is the ability to abstract large parts of the source code and merge them into a concise name of a feature. Depending on the abstraction level, 25,000 lines of source code may be represented as a single market feature, or as a set off 100 system level requirements with an additional 100 quality requirements. This ability of compression may lead to a situation where requirements engineering research reported in a large-scale context actually operates in a small-scale environment, which simplifies the problem of scalability. As a result, reported methods in the literature do not have to be fully scalable, unless they only operate on this high abstraction level [12]. When looking into the requirements engineering literature, it is tempting to make a statement that most reported research follows the mentioned abstraction simplification. When the over-simplification across the abstraction ladder is made, addressing the scalability of achieved results becomes difficult.

The scalability issue can't be addressed by evaluating and testing our solutions on 'toy examples' (according to [6], many reported case studies are based on small toy examples) and then expect is to scale and be useful in an industrial environment, and that our results will be transferred to industry. If we do want to have any impact of our research results outside of the academic world (according to [5], software engineering research has very little impact in industries), I do believe that research in software engineering in general, and requirements engineering in particular, should enable product development organizations to improve, both short terms and long term. This implies that research should be pragmatic and based on industry needs, solving real problems for the real world, and testing the results and assumptions in the real world (e.g. like the QUPER model [13]). That is, the results should be validated and evaluated in a non-simulated environment for their usefulness and scalability. To identify the real world problems and challenges we can either conduct, e.g. case studies (as described in Runeson et al. [6]) to identify the real needs and problems, or, as suggested by Briand [5], establishing tight collaborations with industry or public institutions.

Conclusions

So what's the remedy? We need to recognize the importance of creativity, innovation

and scalability in requirements engineering. I do believe we need to empirically investigate how creative elements and creativity theories apply to requirements engineering and integrate them within requirements engineering methods to foster and support creativity. Creativity techniques and tools can be adopted to improve the requirements engineers' creative abilities. One important issue is the integration of creative techniques within requirements engineering processes and to make sure that they scale, which I believe could advance the requirements engineering domain.

References

[1] van Lamsweerde, A., Requirements Engineering in the Year 00: A Research Perspective, in proceedings of the 22nd International Conference on Software Engineering, pp. 5-19, 2000.

[2] Forbes Magazine, November 2011.

[3] MDD 93/42/EEC (Medical Device Directive)

[4] Kao J, Jamming: the art and discipline of business creativity, Harper Collins, 1996.

[5] Briand, L. Embracing the Engineering Side of Software Engineering, Computing now, vol. 29, July/August, 2012, pp. 93-95.

[6] Runeson, P., Höst, M., Rainer, A., Regnell, B., Case Study Research in Software Engineering: Guidelines and Examples, John Wiley & Sons, USA, 2012.

[7] Patterson, F., Kerrin, M., Gatto-Roissard, G., and Coan, P., Everyday innovation: How to enhance innovative working in employees and organisations, NESTA research reports, pp. 1- 54, 2009.

[8] Quinn, J.B., Baruch, J.J., Zien, K.A., Software based innovation. Sloan Management Review, vol. 37, pp. 11– 24, 1996.

[9] Maiden, N., Jones, S., Karlsen, K., Neill, R., Zachos, K., and Milne, A., Requirements Engineering as Creative Problem Solving: A Research Agenda for Idea Finding, in proceedings of the 18th IEEE International Requirements Engineering Conference, pp. 57-66, 2010.

[10] Maiden, N., and Robertson, S., Integrating creativity into requirements processes: experiences with an air traffic management system, in proceedings of the 2005 13th IEEE International Conference on Requirements Engineering, pp. 105-114, 2005.

[11] Victor, R., Vidal, V., Creativity for problem solvers, AI & Soc, vol. 23, pp. 409-432, 2009.

[12] Wnuk, K. Visualizing, Analyzing and Managing the Scope of Software Releases in Large-Scale Requirements Engineering, PhD Thesis, ISSN: 1404-1219, Lund University, 2012.

[13] Berntsson Svensson, R., Sprockel, Y., Regnell, B., Brinkkemper, S., Setting quality targets for coming releases with QUPER – an industrial case study, Requirements Engineering Journal, in print, 2011, doi: 10.1007/s00766-011-0125-0.

[14] Regnell, B., Berntsson Svensson, R., Wnuk, K., Can We Beat the Complexity of Very Large-Scale Requirements Engineering? in proceedings of the 14th International Working conference on Requirements Engineering: Foundation for Software Quality, pp. 123-128, 2008.