Imagining the future: A social science approach to the importance of visions in the software development process

Nils Kubischok

adesso AG, Germany
nils.kubischok@adesso.de

Abstract. A vision or some other form of future concept is crucial for the development of a new software system as the software that is to be developed does not exist yet. So far, the role of visions in Requirements Engineering (RE) is generally acknowledged but underestimated. A possible reason is the lack of a theoretical framework to understand the role of future concepts and their inscribed expectations. This article presents a brief introduction to the importance of future concepts for various technological fields. It introduces theoretical concepts from different disciplinary backgrounds, e.g. sociology of expectations as well as science and technology studies and applies them to the world of software development and the task of RE. The aim is to show the increased need for creativity techniques in order to develop reliable visions for IT projects as well as to spark a broader discussion on the role of future concepts in RE itself.

Keywords: Requirements Engineering, Sociology, STS, Visions, Expectations, Scenarios

1 Introduction

A core element of software development is to create software that does not exist yet. It exists however in ideas, visions, scenarios or other forms of future concepts. What is common to all types of future concepts is that they comprise the expectations of different stakeholders. How detailed these future concepts are varies from project to project and depends on several factors, e.g. if the software to be is an iterative development from an old system or radically new (where the unknowns are much broader). Also the social setting, e.g. how many stakeholders are involved, can have an impact on the clarity of a vision. Envisioning a new software system and adding the necessary degree of detail is a fundamental task in Requirements Engineering (RE) where creativity is the prerequisite for the creation of (new) visions [1]. A rather vast and broad body of literature covers the importance of visions and scenarios for the software development process, however with an emphasis on scenarios [2-4]. Yet, one essential element is missing so far: a theory on how future concepts work in social and organizational settings. The basis of any form of a future concept in software

development is shaped by a stakeholder's expectation about the future features of a software system. In order to better understand the role of these expectations and their associated future concepts in RE, this article draws on approaches originating from the sociology of expectations. It aims to outline a basic theoretical framework that stems from sociology and takes first steps to make it useful for the RE community. This article contributes to a better understanding of the underlying social processes when it comes to different stakeholders and their expectations and thereby paves the path for an improved assessment of future concepts and management of expectations. By outlining a theoretical framework this article hopes to contribute to a better understand of the earliest stages of RE and how important it is for the constitution of a social setting that integrates all stakeholders.

2 Future Concepts

Future concepts exist in different forms. What is common to all of them is that they exist in language or images alone. Future concepts can differ in their level of detail, which is rather low in visions but much higher in scenarios. What is common to all future concepts is that they comprise the expectations of their authors.

Science-Fiction is a genre in literature that had and still has an enormous impact on reality by influencing scientists and entrepreneurs alike [5]. The most popular examples are scientists who used Star Trek references when explaining their research motivation or even concrete research interests. This role of future concepts in science has been widely debated [6-8].

One particularly interesting example is the rise of nanotechnologies and their retrospective founding moment in 1959, when physicist Richard Feynman gave his famous speech titled "There is plenty of room at the bottom" [9]. In his speech, Feynman claimed that we will soon be able to write the content of the whole Encyclopedia Britannica on the head of a pin and that small machines would be able to build successively smaller machines until they reach the nanoscale and basically build new artifacts atom by atom. Four decades later the National Science Foundation started its vast funding campaign of nanotechnologies under the theme "shaping the world atom by atom" [10]. The vision of Feynman still resonates with scientists and policy makers alike more than 40 years after his speech. The pictures of nanorobots building structures or patrolling our blood were and still are common themes in popular science and science alike [11].

Nanotechnology is a strong example for the influence of future concepts, but there are also examples that connect much better to a more recent and timely discourse in software development: the Internet of Things (IoT). The very idea of the IoT exists in many different wordings: Ambient Intelligence, Pervasive Computing or Ubiquitous Computing. One of the key inventors of the concept of Ubiquitous Computing was Mark Weiser, who in 1991 published his now famous article "The Computer for the 21st Century". In this paper Weiser formulated a vision and a narrative scenario, the Sal-scenario. The most important statement in Weiser's paper is a vision:

"The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it." [12]

Weiser elaborates his vision in a detailed narrative scenario about a woman called Sal and her daily routine in which she is surrounded by invisible computers assisting her in her tasks at home and at the workplace. Weiser's vision and his scenario back then have now partly become reality. Others however are not yet realized and we are still on our way to fulfill the vision. Nevertheless, Weiser's work and particularly his vision has had and still has a great impact on the modern discourse revolving around IoT technologies [13, 14].

Future concepts differ when it comes to their level of detail. Visions for example are considered low on technical details but are regarded as highly influential, not despite their low level of detail but because of it. Weiser's vision contains the idea that technology ideally becomes invisible and weaves into the fabric of everyday life. The rather low level of detail allows for different ways of realization and thereby more actors can potentially contribute to the realization of the vision. How technology is supposed to disappear is not mentioned at all and is therefore open for different paths of realization. The lack of technical detail potentially mobilizes more actors. Scenarios on the other side exist in many different typologies for different use-cases, but generally have a higher level of detail than visions. They contain not only technological features but their socio-technical context [15, 16]. Weiser's scenario is several pages long and describes the everyday life of a woman but in this example technological features are presented as well. The scenario clarifies the vision. Whatever the level of detail in future concepts is, they all have in common that they carry the expectations of different stakeholders.

The role of future concepts is not limited to research and research policy. It is also well established in the context of innovation and product design [17] and of course in software development. In RE the role of visions in developing an image of a future system is acknowledged as well:

"Each requirement engineering process starts with an aim to change the current reality. Regardless of the complexity of the project, the essence of the desired change should be defined briefly and precisely. We call this definition of the envisioned change the system "vision"." [2]

In this quote, Pohl also places importance on the associated social processes, as he goes on to note that the vision guides the stakeholders in the development process and allows them to "align their activities with the defined vision" [2]. This is almost a sociological remark, and we will see, that his idea resonates well with other sociological research approaches.

3 Sociology of Expectations

While most of the literature in context with the sociology of expectations is based on empirical work from research and innovation policy, a lot can be applied to the

more concrete and practically orientated field of software development. The underlying difficulty in research and innovation policy and in product development is the same: whatever is to be developed does not exist yet and one can only communicate about it through future concepts, e.g. visions. Visions comprise expectations which are providing a structure for legitimation, steering of activities, generation of attention and resources while also defining roles, duties and offering a way to prepare for opportunities and risks [18]. Expectations can coordinate actors on the horizontal axis (e.g. in between group members) and on the vertical axis (e.g. in between different hierarchies) within an organization. The main difference between expectations in research policy and expectations in product development is the fact, that expectations in research policy are collective expectations where the outcome will most likely differ to a larger degree from the vision. In product development however the set of actors involved in formulating expectations is smaller and the expectations are much more focused on a clear outcome.

Expectations are prospective structures that allow actors to define their own roles and the roles of others within a narrative [19]. Other actors have to react to the attribution of roles and position themselves in relation to other actors as well. Expectations and promises on fulfilling them lead to a process called "mutual positioning". Mutual positioning allows for the development of different options of action and strategies. Expectations and strategies together allow for agenda building, a process where definitions and specifications are outlined [20]. Expectations therefore are not only descriptive:

"A central theme is that expectations are statements that do something, rather than being descriptive statements that may be true or false. An expectation is not just a description of a (future) reality, but rather a change or creation of a new reality (Guice 1999). In other words, expectations are performative: they do something." *[21]*.

Expectations, according to van Lente, do not just describe something, they create something. Here lies the importance of all different kinds of future concepts that transport expectations. They are a first step in creating a future reality and they lay the foundation for the alignment of heterogeneous actors, the mobilization of resources and the development of strategies. Therefore they are not mere rhetoric, they are a way from rhetorics to social reality [20].

4 Lessons for Requirements Engineering

While the role of visions in RE is generally acknowledged and understood, it still lacks a broader and more profound understanding of their impact. Visions in RE ideally encapsulate the expectations of different stakeholders. The process of building visions itself is as important since visions do not only provide a goal that is to be achieved. Visions and their embedded expectations are more tacit than goals, which are usually clearly stated. However, visions provide a framework for mutual understanding, alignment of different stakeholders, mobilization of resources and the set-

ting of agendas. As Bjarnason et al. state, in order for software projects to be successful, they need functioning communication. Problems in software projects can often be traced back to social factors [22], e.g. diverging common views or (a lack of) mutual understanding. Also other authors focus on the role of visions for product innovation and the importance of a shared and attainable vision [23]. Within their work, it was acknowledged that not only do requirements but also expectations have to be managed [24]. Visions and the accompanying expectations are therefore much more than simple statements of goals, they touch many social aspects that are common through the whole project life cycle. The following key findings from the sociology of expectations can therefore also be applied to software development:

- **Future concepts**: Are well known in various forms. Visions are necessary prerequisites for a successful software project, a fact that is already acknowledged as a part of Requirements Engineering. Scenarios and Use-Cases are different types of future concepts also well known in software development.
- **Expectations**: Collecting the expectations from various stakeholders about a new software system is one of the key aspects of the elicitation process. Managing expectations throughout the whole project life cycle is as important as the management of requirements, as changing expectations certainly means changing requirements. Expectations are embedded in different kinds of future concepts.

The sociology of expectations provides a theoretical framework and vocabulary for some of the social processes concerning visions and expectations:

- **Legitimation**: Clear expectations offer a legitimation for actions concerning the software development process and thereby allow decision making without the need for further arguments.
- **Steering of activities**: By defining goals, roles and duties it becomes clear what is expected of every single stakeholder. Keep in mind that expectations for a new software system also allow for expectations that stakeholders have for themselves and other perceived stakeholders.
- **Mutual positioning**: Formulating expectations for a new system allows stakeholders to also define expectations for them and other stakeholders. Stakeholders have to react to each other's expectations and they thereby shape the vision and social context of a project.
- **Mobilization of attention and resources**: Visions in their incorporated expectations are important for communication as they are the only way to talk about something new to be developed. This allows for the generation of attention for a project and helps to include all important stakeholders and attaining a shared understanding of the vision. A generally shared vision allow for the mobilization of resources in form of funding, staffing and commitment by stakeholders.

- **Agenda building**: Having developed a shared vision and by going through a process of mutual positioning it is now possible to develop an agenda and strategies. Expectations are now hardened in project plans, goals and specifications.

The sociology of expectations allows for a better understanding on how visions and expectations work and why it is important to carefully shape visions in projects, as many different aspects are at stake. For this reason, RE should not only focus on the elicitation of requirements but also on the underlying social processes. Fostering communication, mutual positioning and agenda building can immensely contribute to the project's success beside the elicitation of requirements.

5 Conclusion

While the role of visions in software development is generally acknowledged in RE, the broad impact of visions is still underestimated. This article briefly introduced two fields that are largely impacted by visions. It further outlined a theoretical framework to better understand the role and impact of visions. By explaining and illustrating underlying social and organizational processes, this article contributes to a better understanding of one of the early tasks in RE: Vision building. Visions do not simply state goals; they play a far more crucial role especially in the early stages of a new project. This article shows the importance to competently guide the process of vision building, especially in regards to more radical innovations in software development, such as IoT applications. The implementation of different creativity techniques could be useful to adequately support stakeholders along the way. Developing visions for unprecedented software systems can be a very difficult task, but this task lays the foundation for a successful project as many social processes in the project life cycle are impacted by it.

However, it needs further theoretical and conceptual work and empirical studies in order to develop a reliable and useable framework that draws on the outlined concepts from the social sciences. From a theoretical and conceptual viewpoint, it is necessary to elaborate on what actually defines a vision and how visions relate to e.g. goals and scenarios. Empirical studies will be helpful to understand the process of vision building and how stakeholders shape and relate to them. It will then also be possible to see how visions are reflected in goals and scenarios. Future work can contribute to guidelines for a more successful vision building and thereby to fostering social processes like communication, mutual positioning and agenda building.

References

1. Nguyen, L., Shanks, G.: A framework for understanding creativity in requirements engineering. Information and software technology 51, 655-662 (2009)
2. Pohl, K.: Requirements engineering: fundamentals, principles, and techniques. Springer Publishing Company, Incorporated (2010)

3. Sutcliffe, A.: Scenario-based requirements engineering. In: Proceedings. 11th IEEE International Requirements Engineering Conference, 2003., pp. 320-329. (2003)
4. Alspaugh, T.A., Antón, A.I.: Scenario support for effective requirements. Information and Software Technology 50, 198-220 (2008)
5. Saage, R.: Konvergenztechnologische Zukunftsvisionen und der klassische Utopiediskurs. In: Nordmann, A., Schummer, J., Schwarz, A. (eds.) Nanotechnologien im Kontext, pp. 179 - 194. Akademische Verlagsgesellschaft, Berlin (2006)
6. Gammel, S., Ferrari, A., Gammel, S.: Nanotechnologie, Visionen und Science Fiction. Visionen der Nanotechnologie 79-101 (2009)
7. Kitzinger, J.: Questioning the sci-fi alibi: a critique of how 'science fiction fears' are used to explain away public concerns about risk. Journal of Risk Research 13, 73-86 (2010)
8. Dourish, P., Bell, G.: "Resistance is futile": reading science fiction alongside ubiquitous computing. Personal and Ubiquitous Computing 18, 769-778 (2014)
9. Feynman, R.P.: There's plenty of room at the bottom. Engineering and science 23, 22-36 (1960)
10. Roco, M.: Nanotechnology: Shaping the World Atom by Atom. National Science and Technology Council, The Interagency Working Group on Nanoscience, Engineering, and Technology, Washington, DC (1999)
11. Kubischok, N.: Assessing the future: past and present visions of nanomedicine. Nanomedicine 10, 3195-3197 (2015)
12. Weiser, M.: The computer for the 21st century. Scientific american 265, 94-104 (1991)
13. Schulz-Schaeffer, I.: Scenarios as Patterns of Orientation in Technology Development and Technology Assessment. Outline of a Research Program. Science, Technology & Innovation Studies 9, 23-44 (2013)
14. Bell, G., Dourish, P.: Yesterday's tomorrows: notes on ubiquitous computing's dominant vision. Personal and Ubiquitous Computing 11, 133-143 (2007)
15. Seyff, N., Maiden, N., Karlsen, K., Lockerbie, J., Grünbacher, P., Graf, F., Ncube, C.: Exploring how to use scenarios to discover requirements. Requirements Engineering 14, 91-111 (2009)
16. Van Notten, P.W., Rotmans, J., Van Asselt, M.B., Rothman, D.S.: An updated scenario typology. Futures 35, 423-443 (2003)
17. von Wartburg, I., Teichert, T., Rost, K.: Visioning for innovation-how firms inject knowledge about the future into innovative activities. In: Engineering Management Conference, 2003. IEMC'03. Managing Technologically Driven Organizations: The Human Side of Innovation and Change, pp. 301-305. IEEE, (2003)
18. Borup, M., Brown, N., Konrad, K., Van Lente, H.: The Sociology of Expectations in Science and Technology. Technology Analysis & Strategic Management 18, 285 - 298 (2006)
19. van Lente, H., Rip, A.: Expectations in technological developments: an example of prospective structures to be filled in by agency. In: Disco, C., van der Meulen, B. (eds.) Getting New Technologies Together: studies in making socio-technical order, pp. 203-229. de Gruyter, Berlin (1998)
20. Van Lente, H., Rip, A.: The rise of membrane technology from Rhetorics to social reality. Social Studies of Science 28, 221-254 (1998)
21. van Lente, H.: Navigating foresight in a sea of expectations: lessons from the sociology of expectations. Technology Analysis & Strategic Management 24, 769-782 (2012)

22. Bjarnason, E., Wnuk, K., Regnell, B.: Requirements are slipping through the gaps - A case study on causes & effects of communication gaps in large-scale software development. In: 2011 IEEE 19th International Requirements Engineering Conference, pp. 37-46. (2011)
23. Sarpong, D., Maclean, M.: Mobilising differential visions for new product innovation. Technovation 32, 694-702 (2012)
24. Boehm, B., Abi-Antoun, M., Port, D., Kwan, J., Lynch, A.: Requirements engineering, expectations management, and the Two Cultures. Proceedings IEEE International Symposium on Requirements Engineering (Cat. No.PR00188) 14-22 (1999)