

Technology-Transfer Requirements Engineering (TTRE) – on the Value of Conceptualizing Alternatives

Blagovesta Pirelli^[0000–0001–9890–5227] and Alain Wegmann

EPFL, Switzerland
{firstname.lastname}@epfl.ch

Abstract. In this paper, we describe a requirements engineering method with a focus on the conceptualization of alternative service offerings. The practical context for our project is based on the first author’s work in a startup. Our proposed method is suitable for exploring market opportunities while specifying a service offering. Our method helps requirements engineering practitioners understand the business and technology worlds by modeling business needs and technical capabilities in the same model.

Keywords: technology transfer · market-driven requirements engineering · conceptualization · modeling · service design.

1 Introduction

One of the key roles of requirements engineering (RE) during software development projects is to achieve consensus and a common understanding between business and technology. A major challenge of bringing business and technology together is the different conceptualization of the world that people from the two domains hold. The business domain requires understanding the inner workings of an enterprise, its strategy, operations, and organization. Technology, and in particular research technology, requires specialized knowledge to understand and use it. To find a solution to a business problem and an application for a piece of technology, RE practitioners need to elicit requirements from the business and understand the capabilities of the technology.

Technology-intensive startups (e.g., research spin-offs) have to undergo a technology-transfer process that is a type of market-driven RE (MDRE) process [17]. Startups scout the market space for opportunities that they can address with their technology, instead of producing a customized product for a single customer. They implement a technology push strategy, which involves collaboration with potential customers to find applications for the research technology. The collaboration and feedback from the interaction with potential customers is valuable, as service offerings require information to shape the technology in a usable and profitable way.

The RE community currently explores the area of RE for technology transfer [2, 6]. Technology transfer projects lack direct applicability to an immediate

market, unlike most traditional software-development projects. Yet, technology transfer projects have a potential for a high impact. For example, Popek and Golberg developed the hardware virtualization principles in 1974 [15]. It was only 30 years later that the founders of VMWare¹ used these principles to advance the research and to create the market for virtualization [4]. We believe that RE can accelerate the process of finding a technology-market fit for new research projects.

In this paper, we present the artifact from the first iteration of a design science research (DSR) [10] project - a proposal for a technology transfer RE method to systematically specify alternative service offerings based on research technology in the process of market exploration. The DSR project is based on the experience of the first author who worked at technology-intensive startup for over a year. The method is an explicit prescription [9] to the following question: *How to productize a research technology in a service?*

The structure of the paper is the following. In Section 2, we present our research methodology. In Section 3, we present the knowledge base of the DSR project. In Section 4, we present our RE method. In Section 5, we present the business environment with the example of a pilot project in secure data exchange for an e-health platform. In Section 6, we conclude and give an outlook for future work.

2 Research Methodology

In our research, we use the design science research (DSR) framework as described by Hevner et al. [10]. To both solve a practical problem and contribute an abstract theory, DSR gives guidelines to researchers on how to build socio-technical artifacts, such as system design, methods to design systems, and requirements of systems [14].

A DSR project has three main components: the environment, the knowledge base, and the IS research artifact. Our artifact, a requirements engineering method for early-stage technology-intensive ventures, falls under the category of methods for designing systems [9]. Our method for technology transfer RE is prescriptive: it gives explicit guidelines on how to achieve a task, rather than provide an analysis, an explanation, or a prediction [9, 10]. Our contribution is a theory for design and action [9].

Environment: We obtained our data from the environment via a qualitative study that examines the real-life settings of a startup. The first author was the first employee of the startup (denoted as XYZ), and, as such, could follow the evolution of the company from the beginning. The observed period was fifteen months. Our practical question stems from the environment.

Qualitative research methods have a number of characteristics [13]: (1) the data are rich and contextual; (2) the emerging themes are validated often with informants; and (3) the researcher gains an integrated (holistic) overview of

¹ www.vmware.com

the study. This data collection method is an ethnographic method. We use the Gold’s classification of roles: a complete observer, an observer-as-participant, a participant-as-observer, and a complete participant [7]. The first author assumed a participant-as-observer role, with the observer role being secondary to the participant role. It enables researchers to gain access to more information, as they become part of the environment they study [8]. The drawback is that the collected data are subject to interpretation, reflection and reflexivity [20].

Knowledge Base: The knowledge base consists of foundations and methodologies. The foundations are previously published theoretical contributions. The foundations for our method are requirements engineering theories, service science, and conceptual modeling literature. The methodologies from the knowledge base are formalisms such as models and code.

Artifact: Our artifact is a method for early-phase technology transfer RE. To refine and evaluate the artifact, we use concepts from the knowledge base and interact with the environment. Our assumptions are that with the help of existing RE methods, the conceptual modeling techniques, the service science lens, and technology road-mapping concepts, we can accelerate the technology-transfer process of technology-intensive startups, such as research spin-offs.

3 Literature Review

Our literature review presents the knowledge base of our DSR project. The different streams of literature are conceptual modeling, service science, and RE. In this section, we outline the main relevance of these theories and the portion of them that we use to design our DSR artifact.

3.1 Conceptual Modeling

We base our understanding – of how people conceptualize and model their observations – on the literature of conceptual modeling. A universe of discourse (UoD) is an observed portion of the “reality” during conceptual modeling [1]. A UoD is a set of objects, such as people, physical objects, and technology. The observer establishes semantics (i.e., gives meaning) to the UoD, and with a predefined syntax (formalisms, models) expresses observations [1]. Different purposes for modeling result in different levels of detail and of precision of the models.

An observer, after perceiving the UoDs, *conceptualizes* and *models* the observed portion of the reality. The conceptualization is possible after the observer is able to break down the observed UoDs into nameable, i.e., meaningful, objects (or entities), such as a service, a graphical interface, a data storage, a department [3]. Conceptualization gives meaning to the observed entities and their relationships, whereas modeling gives a structure to the conceptualization.

3.2 Service Science

We use service science to structure our observations. Service science recognizes the value of combining different theories to achieve a holistic view of the service

environment [11]. Service-dominant logic provides a perspective over the combination of elements into service systems, and it is the philosophical foundation of service science [18, 19]. Services are the application of competences for the benefit of others hence the means to create value for others [19]. Service systems are configurations of resources - people and technology [11]. Service science assumes a service system to be the basic unit of analysis of the value exchange [12]. In these value networks, the value added by the socio-technical elements determine their place in the configuration of components.

3.3 RE methods

We use guidelines from the RE literature to understand how to conduct RE in a technology-intensive project. Some of the challenges that RE faces are to ground its findings in the reality in which the technical system will operate and instead of describing the system to be built [23]. There are differences between RE for market-driven and bespoke development, e.g. how to elicit, prioritize, negotiate and analyze the impact of requirements [5, 17]. The main goal of market-driven RE (MDRE) is to gauge a product for a broad market [17].

We use SEAM, an RE method: it enables service designers to model and design systems [21]. SEAM recognizes the importance of an observer of the reality that is being modeled. SEAM includes methods for analyzing explicitly the relationship between different stakeholders [22] and to conduct goal-oriented RE [16]. This enables service designers to define terms concretely (“What is a market/benefit/value?”) and to perform analysis of the stakeholders’ goals in the environment in which an IT system would operate. The explicit positioning of technology within a service system model enables us to also reason about the benefits for service adopters, or the market position of new customers (either market re-segmentation, or customers of competitors within the same segment). The service models use the same ontology to represent different parts of the reality – both business and technology.

4 Technology Transfer Requirements Engineering (TTRE)

We choose to model the service offerings with the help of SEAM and to further develop its applicability to designing new service offerings based on research technology. We formulate explicit hypotheses about potential benefits directly stemming from the technology for service adopters, and we build proof-of-concept prototypes to test the assumptions we have regarding the market propensity. After multiple feedback cycles, we keep only the requirements for the features that show high impact for a broad market. The distillation from concrete steps to conduct such early-phase RE lead to TTRE:

1. Observe - Examine the existing technology and the potential business settings in details.

2. Conceptualize and model - Define **explicitly** a set of alternative services that emerge from the observations.
3. Construct - Build proof-of-concept prototypes for the current prospects of the company to validate that services are of actual value for them.
4. Contextualize - Collect feedback from the prospects to make adjustments to the service alternatives and to the market assumptions.

The TTRE method is iterative and the expected outcome is to be able to outline a general profile of a customer and extrapolate the industry and the market segments that the service addresses.

5 TTRE in Context

In this section, we present the environment of the DSR project and the way TTRE applies in it, based on the first author’s experience. We describe the four different steps (Observe, Conceptualize and model, Construct, Contextualize) with the example of XYZ and a pilot project with E-Health Cisco Switzerland (Cisco for short). We show only one industry and illustrate TTRE with alternative technical service offerings. Another application for TTRE, and area for future work, is to show the same service alternatives in different industries.

5.1 Observe

Technology. The initial technology of XYZ enables operations such as search and share on end-to-end encrypted data. In this context, end-to-end refers to the notion of encryption at the point of creation of the data and storage of the data only in an encrypted form (cipher text) when it is outside of the boundaries of the creating unit. XYZ’s technology consists of a set of various components, each of which has a different function. The purpose of the components is to provide searchable and sharable encrypted data. The features that the technology can offer are (1) libraries - encryption, searchable encryption, and sharable encryption libraries, (2) storage - data and search index storage, and (3) graphical user interface for end-user applications.

We stop making the observations on the research technology at the level where the components are meaningfully divided into different-purpose components. The components contribute to achieving the main goal of the technology: in XYZ’s case, search and share end-to-end encrypted remotely-stored data.

Business. Our running example is a project between Cisco and XYZ. The current offering to the healthcare organizations is a health-data exchange platform. Cisco’s role is to provide a secure medical-data exchange infrastructure that enables different stakeholders to communicate in a unified way.

While in the process of delivering treatment to patients, certain data is generated, for example, prescribed medications, test results, courses of treatment, and check-up dates. All of these data are associated with the patient and stored

in their record. The electronic health record is sensitive information and requires a secure exchange process between different actors. Additionally, different stakeholders have access to only a subset of the records.

We stop the observations on the business domain at the level of service exchange between our potential customer (Cisco) and their customer(s) (hospitals, medical practitioners). These observations give us information on what value we can bring to the value network of our customer to enable them to offer a feature desired by their customer.

5.2 Conceptualize and Model

Observers form concepts about what they have observed, as a precursor for expressing ideas in concrete formal models. Conceptualization is an interpretative step of translating the observed signals into semantically coherent entities and relationship between those entities. Using these observations, we compose different configurations (alternatives) of the XYZ’s technical components.

Technology. The observed features of the technology are grouped together in three possible services. There are different ways to group features. For example, at random, with the core functionality in mind (share data or search data), and based on the abstraction level (software libraries, development toolkit, end-user graphical interface). We choose to group the features of the alternatives service offerings, based on their abstraction level. Table 1 shows the grouping of features that form the three alternative services for XYZ’s technology: an encryption package, a secure data-storage platform, and an encrypted cloud-storage system.

Table 1. Features and services.

Feature	Encryption package	Secure data-storage platform	Encrypted cloud-storage system
Encryption libraries	✓	✓	✓
Shareable encryption libraries	✓	✓	✓
Searchable encryption libraries	✓	✓	✓
Data storage		✓	✓
Search index storage		✓	✓
GUI			✓

Encryption Package. The shareable encryption package is a library that offers encryption that preserves operations (search and share) on the data. The user of this package is a software developer, who develops their own applications and already has mechanisms in place to interact with their storage servers.

Secure data-storage platform. In this package, we propose a service: a platform that handles the interaction with the data and the search servers, as well

as the data encryption and decryption. The user is a software developer who develops end applications for their users and makes use of the platform to enhance the security of users' data.

Encrypted cloud-storage application. This service alternative bundles all the features of XYZ's technology. The user of this package is an end user.

Business. With the help of formalized models, we are able to reason about the different actors in the service system, the process in which they participate and the possible integration points for XYZ's service offerings. Figure 1 represents the model of the e-health data exchange ecosystem. We choose to model the ecosystem in three levels; each of them represents the grouping of features based on Table 1. On the topmost level, the model depicts the e-health data-exchange service system in Switzerland. The actors (medical practitioners, pharmacies, post-hospital facilities, and the Swiss Post e-health unit) all contribute their services to the process of "secure exchange of electronic patients data". These are the customers of our customer.

One of the alternative service offerings - the encrypted cloud-storage system - can take the place of the electronic medical record (EMR) system. This requires working with chief officers (CxOs) of the organizations to further define the requirements of the system and working with the end users from the first level. However, there is an EMR system provided by the "software vendor" already in place, hence the competition with the incumbent software vendor would be hard.

The next integration point for XYZ is within the software vendor's value network on Level 2. The software vendor's value network provides the EMR system as a service. This value network is on the same level of granularity as XYZ's platform. The EMR system of the "Software vendor" provides the infrastructure for storing and searching through medical records and the mechanisms to share them with other members of the system. XYZ's platform service offering has a limited but well-defined role. The main feature of the service is encrypting the most sensitive parts of the data that contains details about person's health, e.g., EMRs. XYZ has to communicate with the people (project managers and software developers) who choose and configure the different systems that will work with each other.

The last alternative, the encryption package, would require going to Level 3, where the system is built. On this level, the service offering is a software library for the software developers who build the EMR system. XYZ works with the developers to understand their requirements.

5.3 Construct

During the construct, we take as input the models from the previous conceptualization step and build prototypes. The construction has a two-fold purpose: to test the market propensity towards the service offering by placing in the hands of prospective users a concrete implementation and to check the technical feasibility of the service offering.

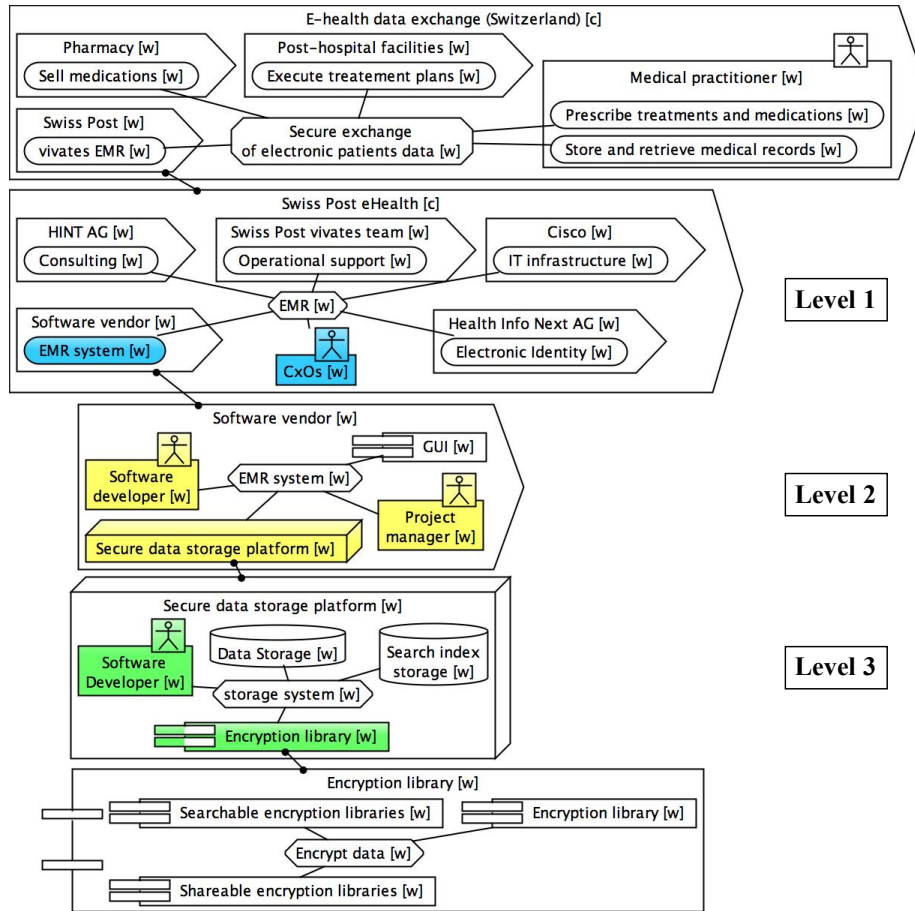


Fig. 1. E-health data-exchange service system (Switzerland)

In the case of XYZ, there were prototypes for all three alternatives. XYZ’s software developers, guided by the identified dependencies between the service offerings, used the service offering alternatives to package the components.

5.4 Contextualize

The last step is to “place” the service offering in the environment of the customers. At this stage, the service offering prototypes serve both to sell to customers (generate revenue) and to orient the technology development during the next iteration. The service alternatives transfer a limited vocabulary from the technology to the business application domain. Little by little searchable and shareable encryption becomes a part of the business vocabulary. Also, the business vocabulary influences the technology UoD with terms such as electronic

health/medical records. The RE practitioners' role is to carry these concepts from one UoD to the other and to show the potential to the other is crucial.

Also, the explicit position of the alternatives points towards a different approach in the service-integration process, depending on who the customer is and who the user is. For example, the CxOs are the customers of the storage system but the hospital practitioners are the users of it, hence the technical development has to satisfy the needs of both.

5.5 Findings

The value of the explicit conceptualization is that XYZ categorizes the user and the customer profile of each alternative and collects systematized feedback. This categorization serves to test the viability of alternatives by cross referencing feedback from different industries.

In the case of XYZ, the search for systematically conceptualizing alternative service offerings resulted in a catalog of opportunities that linked the market, the market segment and the layering of technology components. XYZ worked on projects for various industries, namely insurance, e-health, banking compliance. In each industry, XYZ had a pilot project using one of the service alternatives. The conceptualization of the services helped the development team develop a stack of services to offer to the different projects and to reuse as many components as possible.

6 Conclusion and Future Work

In this paper, we have addressed the question on how to transfer technology with RE methods. We have presented the results of a DSR project for technology transfer RE. We have described a method highlighting the value of conceptualizing alternative technical service offerings. We have presented a startup in its search for a service/market fit and a project that we observed.

Our future work is to show not only how technical alternatives, but also industry alternatives, influence the choice of technology and market evolution. For a next step, we will link TTRE with value-based RE to explicitly estimate market potential in monetary terms. Furthermore, we will expand our modeling tooling and use goal-oriented RE to delve into the motivational analysis. One of the limitations of our work is that it is based on one startup case. To test the generalizability of the TTRE method, we plan to evaluating the method in another context.

References

1. Boman, M., Bubenko Jr, J.A., Johannesson, P., Wangler, B.: Conceptual modelling. Prentice-Hall, Inc. (1997)
2. Cleland-Huang, J., Damian, D.: Ready-set-transfer! Technology transfer in the requirements engineering domain. In: 2011 IEEE 19th International Requirements Engineering Conference. pp. 327–328. IEEE (8 2011)

3. Daft, R.L., Weick, K.E.: Toward a Model of Organizations as Interpretation Systems. *Academy of Management Review* **9**(2), 284–295 (1984)
4. Devine, S.W., Bugnion, E., Rosenblum, M.: Virtualization system including a virtual machine monitor for a computer with a segmented architecture (5 2002)
5. Dos Santos, J.R.F., Albuquerque, A.B., Pinheiro, P.R.: Requirements Prioritization in Market-Driven Software: A Survey Based on Large Numbers of Stakeholders and Requirements. In: *Quality of Information and Communications Technology (QUATIC), 2016 10th International Conference on the*. pp. 67–72. IEEE (2016)
6. Duarte, C.H.C., Gorschek, T.: Technology transfer-Requirements Engineering research to industrial practice an open (ended) debate (panel). In: *Requirements Engineering Conference (RE), 2015 IEEE 23rd International*. pp. 414–415. IEEE (2015)
7. Gold, R.L.: Roles in Sociological Field Observations. *Social Forces* **36**(3), 217–223 (1958)
8. Gray, D.E.: *Doing research in the real world*. Sage (2013)
9. Gregor, S.: The nature of theory in information systems. *MIS quarterly* **30**(3), 611–642 (2006)
10. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science in Information Systems Research. *MIS Quarterly* **28**(1), 75–105 (2004)
11. Maglio, P.P., Spohrer, J.: Fundamentals of service science. *Journal of the Academy of Marketing Science* **36**(1), 18–20 (2008)
12. Maglio, P.P., Srinivasan, S., Kreulen, J.T., Spohrer, J.: Service systems, service scientists, SSME, and innovation. *Communications of the ACM* **49**(7), 81–85 (2006)
13. Miles, M.B., Huberman, A.M., Saldana, J.: *Qualitative data analysis: A sourcebook*. Beverly Hills (1984)
14. Offermann, P., Blom, S., Schönherr, M., Bub, U.: Artifact Types in Information Systems Design Science A Literature Review. In: Winter, R., Zhao, J.L., Aier, S. (eds.) *Global Perspectives on Design Science Research*. LNCS, vol. 6105. vol. 6105, pp. 77–92. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
15. Popek, G.J., Goldberg, R.P.: Formal requirements for virtualizable third generation architectures. *Communications of the ACM* **17**(7), 412–421 (7 1974)
16. Regev, G., Wegmann, A.: Defining early it system requirements with regulation principles: the lightswitch approach. In: *Requirements Engineering Conference, 2004. Proceedings. 12th IEEE International*. pp. 144–153. IEEE (2004)
17. Regnell, B., Brinkkemper, S.: Market-driven requirements engineering for software products. In: *Engineering and managing software requirements*, pp. 287–308. Springer (2005)
18. Vargo, S.L., Lusch, R.F.: Evolving to a New Dominant Logic for Marketing. *Journal of Marketing* **68**(1), 1–17 (2004)
19. Vargo, S.L., Lusch, R.F.: Service-dominant logic: Continuing the evolution. *Journal of the Academy of Marketing Science* **36**(1), 1–10 (2008)
20. Weber, R.: The Reflexive Researcher. *MIS Quarterly* **27**(4), v–xiv (2003)
21. Wegmann, A.: On the systemic enterprise architecture methodology (SEAM). In: *International Conference on Enterprise Information Systems* (2003)
22. Wegmann, A., Julia, P., Regev, G., Perroud, O., Rychkova, I.: Early requirements and business-IT alignment with SEAM for business. In: *Proceedings - 15th IEEE International Requirements Engineering Conference, RE 2007*. pp. 111–114. IEEE (2007)
23. Zave, P., Jackson, M.: Four dark corners of requirements engineering. *ACM transactions on Software Engineering and Methodology (TOSEM)* **6**(1), 1–30 (1997)