Business Impact Assessment for Autonomic Network Management

Identifying Business Impact Indicators From Use Case Requirements

Sander Spek, Vânia Gonçalves, Simon Delaere
IBBT-SMIT, Vrije Universiteit Brussel
Pleinlaan 2, 1050 Brussels, Belgium
{alexander.spek, vania.goncalves, simon.delaere}@vub.ac.be

***Abstract*—In this paper, a methodology is presented to identify business impact indicators from the use case requirements in an early phase of the development stage. It is based on the Quality Function Deployment method and the business model design framework. This methodology is applied to the design for a unified management framework to facilitate autonomics in complex, ubiquitous and large-scale networks. The analysis concludes that this innovation will be particularly disruptive in the functional architecture and value proposition aspects of the business model.**

Keywords—autonomics; self-x; business impact; business model

I. Introduction

Determining business impacts of systems in pre-commercial development is a challenge that requires both a clear view of technical results and limitations of the system in development as well as a grasp of business insights. Technical objectives are often explicated in an early stage of the development by means of functional, non-functional and business requirements. While business requirements might already give some indication on the eventual business impacts of the system once implemented they are often developed by technical partners and thus limited in scope. Considering business requirements in an early stage is important to anticipate on the market impact of the innovation and possible conflicts between actors. However, it is far from straightforward to relate the functional and non-functional requirements to business impacts.

This paper explores an endeavor to link the identified requirements of autonomic management systems to business impacts by creating a matrix inspired by the Quality Function Deployment (QFD) method (see e.g., [1–5]), which allows for the mapping of individual requirements to business model design parameters. This is applied in the early design phase of the UniverSelf project [6][7], a collaborative project with over 15 partners, aiming to develop a management framework for autonomics in existing and emerging network architectures.

In the next section, the details of the UniverSelf framework to be developed will be explained in more detail, including the six use cases that form the basis for the services' requirements. Section III describes the methodology used, including an elaboration on its two building blocks: the Quality Function Deployment and the business model design framework. The results will be presented in Section IV, while a discussion on the findings takes place in Section V. Finally, the conclusion and future work is presented in Section VI.

II. Unified Management Framework

The goal of the UniverSelf project is to overcome "the growing management complexity of future networking systems, and to reduce the barriers that complexity and ossification pose to further growth" [6][7]. It designs a Unified Management Framework (UMF) in order to enable autonomic principles in current and emerging networking architectures. This framework should constitute a cross-technology, common substrate for both systems and services, and include the necessary functions to achieve self-management in the autonomic network, its systems and its network equipment. The project partners have formulated six use cases, which are considered to be representative and complementary, reflecting the network operator's desires to reduce its costs caused by complexity and reducing the dependency on human operation for operational tasks. This desire includes both the reduction of operational costs (OPEX) as well as network equipment and infrastructures (CAPEX). These use cases are labeled as follows:[1]

1. *Self-Diagnosis/Healing for IMS VoIP & VPN:* self-diagnosis and healing features with applications for IP networks and IMS services as well as VPN networks.
2. *Networks' Stability and Performance:* simulation and emulation results about stability and performance of a network (with a great number of nodes and real impairments) with cross-layer and cross-domain self-configuration mechanisms.
3. *Dynamic Virtualization and Migration of Contents and Servers:* the dynamic virtualization and migration of data/content and network entities (gateways and servers) nearer to users.
4. *SON and SON collaboration according to operator policies:* to design novel SON to improve network operation and performance, and to demonstrate the operation of a mobile network empowered by SON entities within a general management framework.
5. *Operator-governed, end-to-end, autonomic, joint network and service management:* to enable

[1] For historic reasons, use case 5 and 6 are referred to as use case 6 and 7 respectively in the UniverSelf project. For optimal clarity, we use a consecutive numbering in this paper.

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-236-3

operators to describe their goals and objectives, through high-level means and govern their network, to achieve policy-based operation of Radio Access Network and backhaul/core network segments, and to achieve coherence between these segments through cooperation, negotiation and federation.

6. *Network and Service Governance:* facilitates network and service governance through the use of IPTV services running on top of both fixed and mobile networks.

These use cases have been formulated from a technical perspective, complete with functional, non-functional and business requirements, about 200 in total. While these requirements might represent valid technical goals, one also needs to assess their feasibility and viability from a business perspective. Since the systems are complex, the networks are heterogeneous and the stakeholders are multiple, this is not a straightforward exercise. In this paper an endeavor is made, based on the requirements from the use cases. While the cost reduction is the main aim of the framework and its use cases, this analysis is an effort to discover secondary impacts that without this analysis might be overlooked.

III. Methodology

The methodology is aimed at deriving business-impact indicators from the requirements that have been developed by technical partners at an early stage of the design. Two analytical frameworks are taken as the basis. The Quality Function Deployment provides a method of analyzing requirements to derive characteristics and controls from them, although in the original QFD the results are not business impacts. The business model design framework, on the other hand, provides clear business design choices for development in telecommunications and other systems; but, so far has had no connection to the technical requirements outlined at an early stage in a development project. The last subsection contains our synthesized method that takes elements from both.

A. *Quality Function Deployment*

Quality Function Deployment is a method developed in the late 1960s by Mizuno and Akao. It can be defined as "an over-all concept that provides a means of translating customer requirements into the appropriate technical requirements for each stage of product development and production (i.e., marketing strategies, planning, product design and engineering, prototype evaluation, production process development, production, sales)" [8] (via [4]). The underlying idea is that (potential) customers have valuable

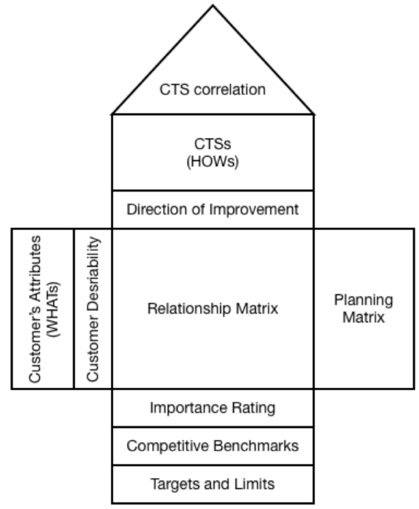

Figure 1. The House of Quality. (Taken from [4].)

input for the design of the product, but that cannot or will not express this in a technical terminology [5]. The total QFD process is described in what is called the *House of Quality*, see Figure 1, but the essence is its four-phase approach, consisting of the following steps: (1) *Product planning,* transforming customer demands into quality characteristics; (2) *Product design,* transforming quality characteristics in product characteristics; (3) *Process planning,* transforming product characteristics into a manufacturing process; and (4) *Process control,* transforming the manufacturing process into quality controls. These phases are presented visually in Figure 2.

Interesting about the four phases of QFD is that they all provide a transformation of certain requirements or characteristics of one kind into requirements or characteristics of a different kind by using a scorecard matrix. Typically this is done by placing both input and outcome on one of the axes, and to add scores to the intersections. These scores can be weights (e.g., to indicate importance) or categories (e.g., strong relationship (9), medium relationship (3), weak relationship (1); example taken from [1]).

B. *Business model design framework*

The business model design framework is a model developed by Ballon (see a.o., [9–10]). It "follows the multi-parameter approach by defining four levels on which business models operate, and by identifying three critical design parameters on each level" [9]. These refer to a whole ecosystem for a product or service rather than to a specific organization within that ecosystem. In Ballon's view, the essence of a business model is the (re)configuration of

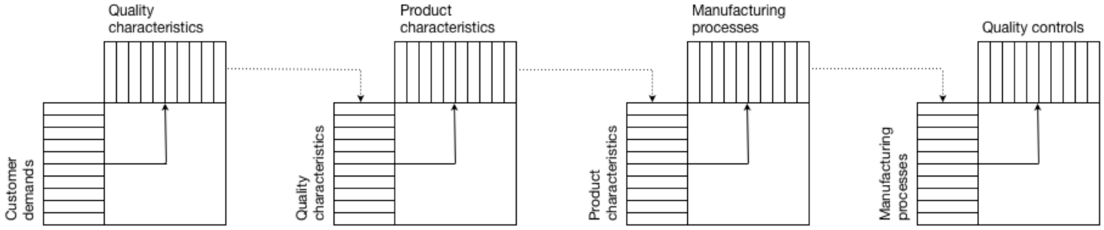

Figure 2. The QFD four-phase approach. Based on [1].

TABLE I. THE BUSINESS MODEL DESIGN PARAMETERS BY BALLON [9–11].

Control parameters		**Value parameters**	
A. Value network parameters	***B. Functional architecture parameters***	***C. Financial model parameters***	***D. Value proposition parameters***
A1. Combination of assets	B1. Modularity	C1. Cost(-sharing) model	D1. Positioning
A2. Vertical integration	B2. Distribution of intelligence	C2. Revenue model	D2. User involvement
A3. Customer ownership	B3. Interoperability	C3. Revenue-sharing model	D3. Intended value

control and value [9][10]. The value reflects the traditional view of a business model as an elaboration of a value proposition, while the control aspect raises attention to the questions on who controls the value network and the system design. As displayed in Table I, the framework reflects this by being divided into two parts: control parameters and value parameters. The control parameters consist of three value network parameters and three functional architecture parameters. The value parameters consist of three financial model parameters and three value proposition parameters. This business model approach based on the (re)configuration of two elements, each providing a more detailed set of parameters, dissolves the disadvantages of both two-parameter schemes and multi-parameter approaches as they existed during the development of this approach. [10]

The design framework consists of two sets of three configuration parameters each. For the control parameters, these are value network parameters and functional architecture parameters. For the value parameters, these are financial model parameters and value proposition parameters. The parameters will be used as indicators of business impacts in this analysis of the requirements in a system design.

C. Synthesized method

A matrix has been created for scoring the functional, non-functional and business requirements to the business model design parameters. Since the requirements per use case vary, this exercise is performed for every use case separately. The scoring entails that a point is given in case the requirement impacts the said design parameter. When counting the points per design parameter, one can gain insights into the importance of that parameter given the requirements of a specific use case. It might feel as a self-fulfilling prophecy to also score the business requirements on the business parameters to determine their business impact. This, however, is justified because the same persons who developed the use cases and also extracted the other requirements have formulated them, and did so from a technical perspective. In other words, they only concern direct business effects of the technology, and do not yet make an analysis of the business consequences, hence the use of them as inputs in the matrix.

In the QFD-methodology, one can give a score to a field in the matrix to quantify the impact it has. In this exercise, only a binary assessment is used; as 197 requirements were scored on 12 parameters, thus creating a 2364 field matrix, this has provided sufficient granularity for the purpose of this method.

A comparable method has been developed in [11], which analyzes the business impacts for mobile self-organizing networks. Technical parameters have been identified on the basis of technical functionalities and key performance indicators (KPIs). The business model design framework was operationalized in the form of business parameters, which were then linked to the technical parameters to create a scorecard of business impacts. Other efforts in extracting business impacts from technical documentation also exist. For instance, Raju *et al.* [12] utilize a framework based on seven business model parameters extracted from the business model design framework, specifically adapted to energy aware self-growing business ecosystems. The method in this paper distinguishes itself on several aspects. First of all, it starts from the requirements, a document that is a part of many development tracks already. Also, it arrives at the business model design framework, which is a suitable and objective general framework for further analysis. As a final argument, the scoring of the use cases is a semi-formalized process. Several persons can perform it in parallel, after which the results can be easily compared and discrepancies can be discussed. In the present case, this also happened.

IV. RESULTS

After scoring the requirements of the six use cases on the business parameters, a matrix is created that can be interpreted generally, as well as aggregated in two directions.

TABLE II. RESULTS OF THE SCORING

	A. Value Network			**B. Functional Architecture**			**C. Financial Model**			**D. Value Proposition**		
	A1	***A2***	***A3***	***B1***	***B2***	***B3***	***C1***	***C2***	***C3***	***D1***	***D2***	***D3***
Use case 1	1	1	1	5	17	20	4	1	1	0	9	11
Use case 2	0	0	0	3	7	2	2	1	0	1	7	7
Use case 3	2	1	0	2	9	8	4	2	0	2	1	8
Use case 4	0	0	0	1	4	6	3	2	0	2	4	4
Use case 5	5	2	0	14	11	11	9	3	0	2	3	13
Use case 6	1	0	0	6	12	2	4	4	0	1	8	8
Total	**9**	**4**	**1**	**31**	**60**	**49**	**26**	**13**	**1**	**6**	**28**	**47**

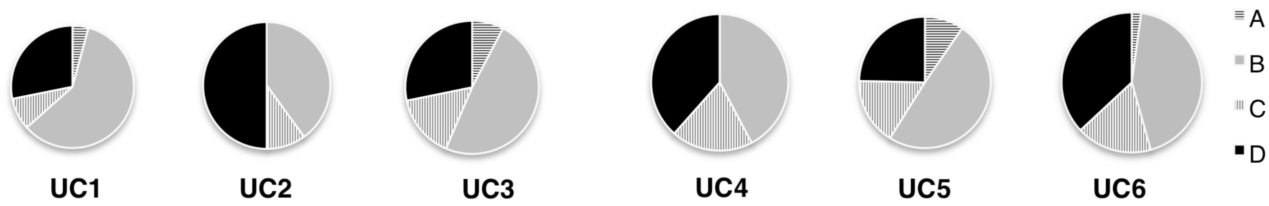

Figure 3. Results of the scoring per use case.

First, the results per use case will be considered, to see if the method gives significantly different outcomes for these different cases. Second, a reverse approach is taken, where the scores are aggregated per business parameter, in order to analyze for single business parameters which of the use cases have a large impact on it.

A. General results

Many requirements have scored on multiple business parameters, while some did not have an impact on any of the parameters. For the 197 requirements, in total 275 scores have been given. The results are displayed in Table II. The individual requirements have been grouped per use case. The business parameters that the requirements have been scored on can be found in the top. One can observe that the system, as designed in this stage, will especially have an impact in sets B and D of the business parameters, namely the functional architecture and the value proposition. It seems less obtrusive in terms of value network and financial model. The biggest emphasis will be on the distribution of intelligence (B2), followed by interoperability (B3) and intended value (D3). Also the modularity (B1) and the user involvement (D2) are stressed. These results can be interpreted as an indication that the service as currently designed puts great emphasis on its functional architecture. Even though distribution of intelligence receives the greatest attention, it scores high on all three parameters. The emphasis on distribution of intelligence results from the aim to build contextual knowledge and monitoring information from both control and data planes gathered from elements distributed over core, distribution and access network levels, which are essential for the orchestration of autonomic functionalities. Since the management framework targets the integration between current and future networks and interfaces with NMSs and OSSs, interoperability also stands out. In terms of value proposition, the service will have a high impact on the user involvement and the intended value. This emphasis on user involvement is a reflection of the efforts to reduce human (operator) involvement, as tasks will be taken over by the autonomic network mechanisms and elements. The system is thus clearly designed to facilitate the reduction of OPEX, although the limited scoring on the cost model parameter indicates that the requirements do not point directly to lowering OPEX; it is often implicit. The emphasis on intended value is a reflection of the requirements focused on improving performance and stability through self-x functionalities, thereby enhancing Quality of Service (QoS) and Quality of Experience (QoE) for end-users.

B. Results per use case

The results per use case are displayed in Figure 3. What becomes immediately clear is that there are significant differences between the different use cases. Especially use case 2 seems to stand out, as it is the only use case that has the biggest emphasis in the value proposition parameters. Both use case 2 and 4 seem to have no impact on the value network parameters, which means that the innovations in these use cases are not likely to alter or disrupt the current value network configuration. Looking at the goals of the use cases, this seems right. Use case 1, *Self-Diagnosis/Healing for IMS VoIP & VPN*, is a technical use case, so it should not come as a surprise that the requirements reflect this by putting most emphasis on the functional-architecture parameters. Similarly, use case 2, *Networks' Stability and Performance*, focuses on the quality of the service to the end-user, and as expected it scores high on the value proposition parameters. Use case 6, *Network and Service Governance*, relies on both the functional architecture and value proposition parameters, and hardly has any impact on the value network and financial model parameters. This can be explained by the fact that this use case is of a supportive nature, improving the functionality and the value proposition of the service by setting the orientations for a governance tool that will contribute to the translation of human high-level service business goals into network policies.

To study this example in more depth, Figure 4 contains the breakdown of use case 6 again, but this time up to the level of the individual parameters. For clarity, the value network and financial model parameters have been blanked out, to focus on the functional architecture (B) and value proposition parameters (D). What becomes clear is that the impact in this section should be mostly attributed to B1 and B2, modularity and distribution of intelligence, as they make up the major share of the solid-line bordered slice of the pie. The impact of the third functional architecture parameter, interoperability, is limited in this use case. In the value proposition parameter set, one can see a similar effect, with the impact being attributed to D2 and D3, user involvement and intended value. There is little impact related to the positioning of the product or service in this use case. One can make such a detailed description for each of the use cases, but space restrictions prevent doing so in this paper.

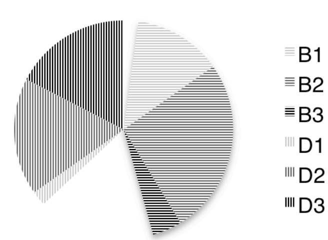

Figure 4. Results of use case 6, specified to individual parameters. Parameter sets A and C have been blanked out in the chart.

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-236-3

C. *Results per business parameter*

The previous analysis can be reversed: taking the design parameters as a basis and study the matrix to determine which of the use cases contribute the most or the least to a specific parameter. These results can be derived from Table II by looking at the columns rather than the rows. Examples are given in Figure 5, with parameters broken down into the use-case contributions. The main contributor for D2, *User Involvement*, is use case 1, with use case 6 and 2 following closely. The interpretation is that the decreasing human operator involvement will be largely determined by the result of the first use case, which deals with self-diagnosis and self-healing. The same reasoning also applies to use case 2 and 6, albeit with a slightly smaller impact. From the use case descriptions, one could also have expected use case 4 to score high here, but judging on the requirements the user involvement impact is present but limited. The parameter *Modularity* shows a different picture. Here it becomes clear that especially use case 1, 5 and 6 contribute to the weight of modularity—an indication that issues concerning integration, interfacing and standardization might occur in these use cases. This is an indication that issues concerning the syntax and semantics of external manifestation can become of high importance in this use case. The results do not provide any information on the specificities of these issues, but they do stress that there is a significant impact of these use cases on the business configuration concerning modularity — something that can serve as valuable input for further analysis. The last two use cases deal mostly with the management and governance of the system, so the stress on modularity issues is a plausible result. The large impact of use case 1 is more surprising, but is largely justified by this use case's aim to integrate several sources of information (service and network alarms, configurations, messages, performance indicators, etc.) spread across different network elements, service components and NMS which will be the primary source for self-healing and diagnosis decisions.

V. DISCUSSION

In this section, a discussion on the methodology will be given as well as a discussion on the results.

A. *Discussion on the methodology*

This paper provides work in progress on a novel method to identify business impacts in an early stage of the product or service design process by the use of analyzing the requirements drawn out by technical partners. The analysis of the results in this paper shows that in many cases the results of this exercise are plausible, as one can explain them from a business analysis. However, since this is a new type of analysis one should be careful for a bias, e.g., a bias of the technical partners creating the requirements, or a bias of the analyst(s) scoring the requirements on the business model design parameters. These biases cannot be excluded entirely, but can be contained by involving multiple parties in the exercise. Moreover, the value of the exercise is not so much in the exact numbers it produces, but rather in the indications that it gives. The results should not be interpreted rigorously, but should rather serve as an input for business analysis. It provides early indicators on where the design puts stress on certain business impacts that might turn into bottlenecks or require special attention later on during development. It could raise awareness on a multitude of issues that need to be tackled before implementation or commercial deployment: issues concerning backward compatibility, standardization, trust, the value network, and etcetera.

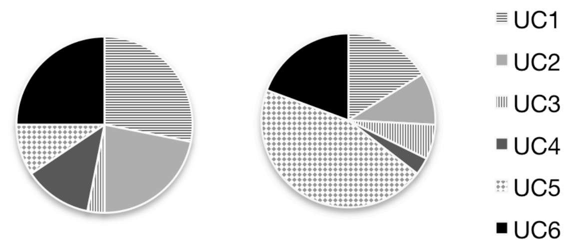

Figure 5. D2, User involvement (left), and B1, Modularity (right), broken down to the contributions of the different use cases.

Another bias might be in the method itself. The results indicate a strong emphasis on the functional architecture parameters. This could be a characteristic of this specific service, but it could also be a natural bias, since functional requirements make up a large part of the total set of requirements and they might favor the functional architecture more than the other sets of configuration parameters. By applying this method in more projects, such a bias could be discovered. A solution could then be to include an extra weighing factor to mute the discovered biases.

The weighing factors of the different types of requirements can also be considered. In this exercise, all types of requirements have been treated equal, despite there being 131 functional requirements and only 31 and 35 non-functional respective business requirements. Especially the business requirements already contain a first indication of the possible business impacts of the use cases, so one could consider weighing them in more than the other requirements.

B. *Discussion on the results*

The type of analysis as performed in this paper does not have a clear impact assessment as an outcome. It is rather an explorative exercise that provides valuable input for a qualitative analysis to follow. It explores the technical specificities from the requirements and indicates where they could have a business impact. It can be valuable input to discover impacts a straightforward analysis could have missed, and it provides some insights on which impacts are the heaviest, and which use cases specifically have impacts on a certain business parameter.

The main aim of the framework is a cost reduction, particularly in the form of OPEX. The use cases have been developed with this in mind, but this is not reflected in the results of the analysis. Most of the studied use cases are facilitating this higher goal, but do not refer directly to it. Therefore, the results should be interpreted as those business impacts that exist next to the cost reduction and that might facilitate this reduction.

The value network parameters are expected to have a relatively small impact. This means that the use cases, as currently formulated, will not be very disruptive in terms of

Copyright (c) IARIA, 2012.

the value network, although the requirements are very low level, and higher level disruptions could be discovered from other types of analysis. Therefore, there is not much impact on assets belonging to certain roles, on vertical integration and disintegration, and on customer ownership. Also the financial model is relatively stable. However, these results do not take into account the actual business scenarios, and the different value configurations that can arise. The matrix indicates some issues concerning the cost(-sharing) model and the revenue model, but these are less significant, or at least more straightforward, compared to other disruptions. Considering potential business scenarios (e.g., network virtualisation, network infrastructure sharing), the introduction of the management framework could however impact the value network and the split of costs and revenues.

The value proposition is subject to change. While positioning is stable, there is a high impact on user involvement and intended value. This is a reflection of the decreasing human involvement caused by autonomic configuration and interventions and the emphasis these use cases put on network performance and stability towards an improved QoS and QoE.

Finally, the functional architecture is the area with the highest business impact. All three parameters are impacted, which is partly a reflection of the design choice to create a modular, distributed system with autonomic mechanisms. The analysis recognizes this, and stresses the consequences this might have for the business model. It puts an emphasis on automatic collaboration and standardization of policy languages and interfaces. One consequence of this is that the roles responsible in the value network must have a good understanding between each other and must have a sufficient amount of trust. In networks where such an understanding and trust is not guaranteed, this design choice could cause severe problems when the system or service gets deployed. In such a case, a misalignment of business model configurations between the designed framework and the actual market, either one of the two has to be adapted.

VI. CONCLUSION AND FUTURE WORK

A methodology has been proposed to use a QFD-inspired matrix to derive potential business impacts from use case requirements. The outcome is a set of impact indicators belonging to parameters of the business model design framework, which can serve as the basis of a qualitative impact analysis. This methodology is applied to the six use cases of a design for a unified management framework in order to enable autonomic principles in current and emerging networking architectures. Cost reduction in the form of OPEX and CAPEX was the main objective for these use cases, but with this methodology one can find other impacts — either facilitating the cost reduction or unrelated to it.

The model indicates the highest impacts in the parameters relating to the functional architecture and the value proposition. The first is caused by the distributed design of the framework, which has major business impact consequences. The value proposition is impacted by the changes in user involvement due to automation, and the changes in the intended value due to improvements in the QoS and QoE. As it turns out, the designed framework already imposes a certain business model configuration. This analysis provides a first insight in whether this configuration matches the actual market situation or that adaptations need to be made.

The methodology thus provided a promising starting point for further analysis. It highlights the most important parameters to be considered in a following qualitative study. However, the methodology needs to be applied to more designs in order to study and contain possible biases. Also, practical guidelines about validations of the matrix by multiple actors need to be developed.

ACKNOWLEDGMENT

The research leading to these results has been performed within the UniverSelf project (www.UniverSelf-project.eu) and received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 257513.

REFERENCES

[1] A. T. Bahill and W. L. Chapman, “A tutorial on quality function deployment,” Engineering Management Journal, vol. 5, no. 3, 1993, pp. 24–35.

[2] C. P. M. Govers, “What and how about quality function deployment (QFD),” International Journal of Production Economics, vol. 46, 1996, pp. 575–585.

[3] Y. Akao, “QFD: Past, present, and future,” in International Symposium on QFD, 1997, vol. 97, pp. 1–12.

[4] L. K. Chan and M. L. Wu, “Quality function deployment: A literature review,” European Journal of Operational Research, vol. 143, no. 3, 2002, pp. 463–497.

[5] K. Yang and B. S. El-Haik, “Chapter 7: Quality Function Deployment (QFD),” in Design for Six Sigma: a roadmap for product development, Second Edition. Kindle ed., McGraw-Hill Professional, 2008.

[6] Univerself Project, “UniverSelf, realizing autonomics for Future Networks,” Oct-2010. [Online]. Available: http://cordis.europa.eu/fp7/ict/future-networks/documents/projects-univerself-presentation_en.pdf.

[7] Univerself Project, “About UniverSelf,” 2010. [Online]. Available: http://www.univerself-project.eu/about-univerself.

[8] L. P. Sullivan, “Quality Function Deployment,” Quality Progress, vol. 19, no. 6, 1986, pp. 39–50.

[9] P. Ballon, “Business modelling revisited: the configuration of control and value,” info, vol. 9, no. 5, 2007, pp. 6–19.

[10] P. Ballon, “Business Modelling as the Configuration of Control and Value,” in Proceedings of the 20th Bled eConference eMergence: Merging and Emerging Technologies, Processes, and Institutions, Bled, Slovenia, 2007.

[11] V. Gonçalves and S. Delaere, “Business Impact Assessment of Mobile Self-Organising Networks,” in New Frontiers in Dynamic Spectrum, 2010 IEEE Symposium on, 2010, pp. 1–12.

[12] A. Raju, S. Lindmark, O. Yaron, E. De Poorter, L. Tytgat, S. Delaere, and P. Ballon, “Business Model Assessment of Green Wireless Sensor Ecosystems,” in Proceedings of CTTE’12.