Connecting Business Models with Service Platform Designs – Quantitative, Scenario-Based Framework

Josip Zoric

Telenor R&I, and Norwegian University of Science and Technology,
Trondheim, Norway
josip.zoric@telenor.com

Abstract. Heterogeneity of technical and business designs, complexity of collaborations, and incentives are just some of the consequences of service platform evolution that complicate their business analysis. Business models are usually on a higher abstraction level than service platform designs, which requires detailing prior to their financial analysis. This work proposes a framework for quantitative analysis, which "reinterprets" the business models by underlying service platforms technical and business entities, processes and scenarios. In such a way it prepares them for business analysis and valuation, focusing also on incentives of collaborating business actors. We explain the approach theoretically and demonstrate its use on the proof-of-the-concept service platform.

Keywords: techno-business modeling, service platforms, scenario approach, service, enabler, quantitative analysis, valuation.

1 Introduction

Converged, cross-media platform models and mobile communication mashups and platforms are increasing opportunities for new – and more dynamic value propositions. Heterogeneity of technical and business designs, complexity of collaborations, diversity of roles and incentives are just some of the consequences of service platform evolutions, which make their techno-business analysis complex. Service platforms are complex technical and business systems, hosting multiple service portfolios (bundles), influenced by dynamics from four spheres: user, business, system and technological [1-3]. All of them can dramatically influence the quality of experience, quality of service and thus the business value. A good quantitative business estimate is dependent on the input of all four spheres. The business analyses should not oversimplify any of them.

Technical and business models of services and platforms revolve around two knowledge sets:

(1) *At a higher abstraction level* - business models (BM) and qualitative business analyses specifying the value propositions and underlying value configurations (e.g. [4-8]). There is neither consensus nor a standard approach in business

C. Hesselman and C. Giannelli (Eds.): Mobilware 2009 Workshops, LNICST 12, pp. 33–44, 2009.

modeling [4-8]. Pateli and Osterwalder [14,6] suggested: whoever wishes to build a BM should select one of several published approaches, justify why using that instead of another similar, and use that as foundation for building her/his model.
(2) *At a higher detail level* – a plethora of technical and business designs of services and platforms (difficult to analyze and compare).

But how to map the BMs to their realizations, and how to estimate their technical and business impacts. In business terms, how to valuate the underlying asset (in our case service platforms) and quantitatively analyze business actor incentives in collaborative service provision? We offer a framework for techno-business analysis (TBA) of services and platforms, which might serve this purpose. The TBA [1,2] engages in service and SP analysis with respect to four important aspects affecting its technical and business performance: user, business, system and technical, which we discuss below.

User analysis in our approach contains two aspects: service customer and service user. Customer aspect focuses on user value concepts, their business counterparts and contribution of services in their realization. Focusing on service user means mapping the user value concepts to goals, requirements and usage scenarios, and analyzing which service can support their realization. We call it a service usage modeling. We focus on service support in delivering the value to the user (assisting the user in goal realization that is represented by scenarios). A user scenario describes the end-user behavior when interacting with the service.

Business analysis considers value proposition, value configuration, corresponding business models [4-8], and business process models [2] specifying collaborations in service delivery.

System and technical analyses focus on system models and system process models needed for analyzing system interactions in collaborative service provision. In this part of the TBA we take over the business process models and map them to their system and technical counterparts, with help of scenario techniques. We use (Message Sequence Charts (MSCs) [10] and Unified Modeling Language (UML) [11]. For scenario representation we start with scenario stories (modeled with help of templates [12,13]), which we map to interaction diagrams (MSCs).

We detail the TBA framework in the following text. In section 2 we present the SP approximate (used to simplify / prepare the SPs for the techno-business analysis), and explain the main phases of the TBA. In section 3 we discuss some additional analytical opportunities the TBA framework can offer. Namely, model-based mapping and projection techniques can also be used for qualitative and quantitative comparison of various, business, system and technological solutions. Section 4 exemplifies the use of the TBA on a proof-of-the-concept case, while section 5 concludes this work.

2 Techno-Business Modeling Framework

The TBA framework is based on: *(1)* a service platform approximate, *(2)* scenario-based modeling and simulation of SP's technical and business processes and *(3)* portfolio analysis and valuation of services and enablers. Universality of the TBA is based on uniting these analytical techniques in a common framework, which can be used for the techno-business analyses of services and SP solutions, varying from simple service provisions to provisions of complex service and enablers' portfolios. The TBA enlightens simultaneously roles and responsibilities (specified by interactions and collaboration scenarios) of SP entities and business actors: services, enablers, end-users and service and enabler providers. This requires well structured and methodical analyses, which we discuss in the following subsections.

2.1 Service Platform Approximate

A structural service platform approximate, so called Generic Service Platform Model (GSPM) [1,2], hides a complexity of the SPs, simplifies system and technical designs and prepares them for techno-business analyses. The GSPM, shown in Fig.1, abstracts the service platform designs by the following entities (classified according to the set of both technical and business criteria [1,2]): services, enablers, service platform mechanisms (e.g. service discovery, composition, brokering and mediation), service platform capabilities and resources. Due to a high abstraction level the GSPM is also used as a framework for comparison of technical and business solutions, as discussed in section 3. We can conclude that we approximate the SP by two portfolio sets: *(1)*

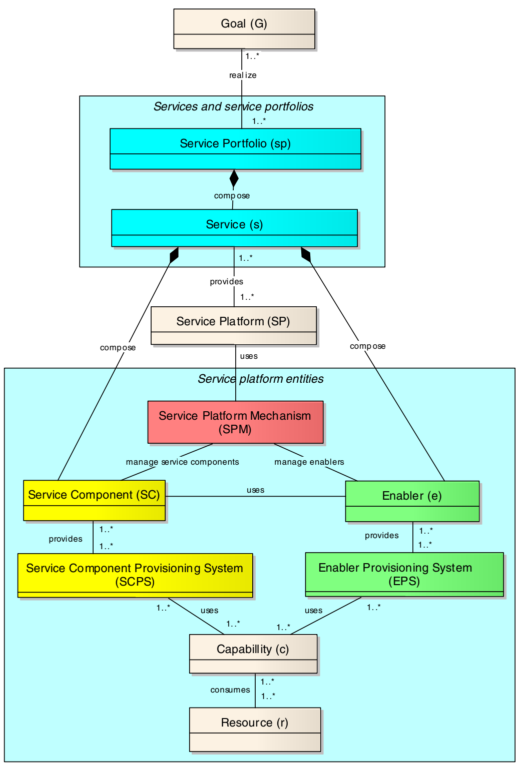

Fig. 1. Generic Service Platform Model (GSPM)

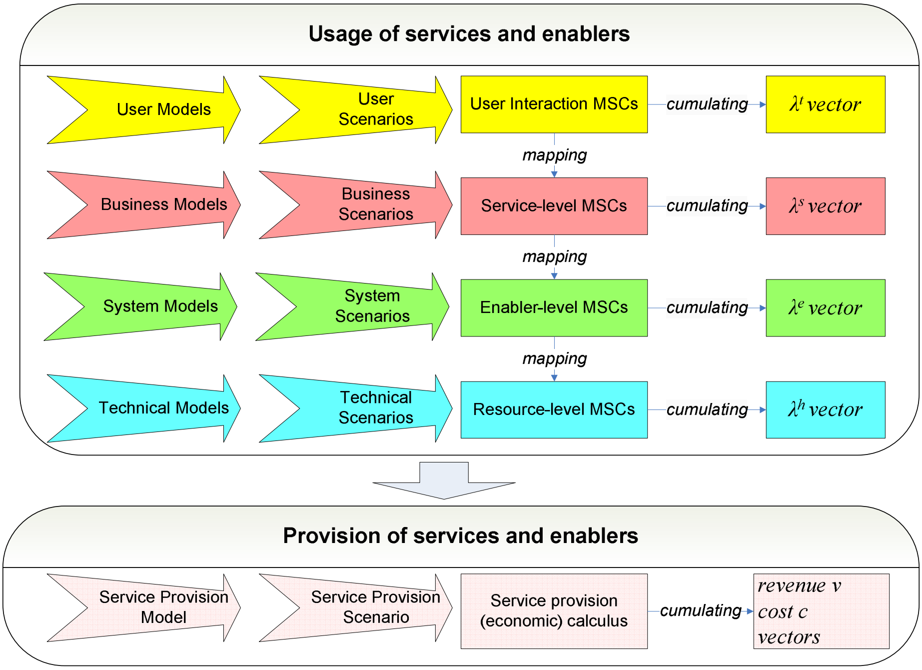

Fig. 2. Techno-business modeling approach (contains two major phases: *usage* of services and enablers, and *provision*). Scenario mapping and projection (cumulating) techniques are illustrated in each phase of the model, and explained below.

end-user service portfolios (in further text service portfolios), and *(2)* portfolios of enablers and service components, used to compose the end-user services and their portfolios.

When the SP is approximated with help of the GSPM (Fig.1), we engage in the TBA. As illustrated in Fig.2, the TBA is divided in two phases: *(1) usage* of services, enablers and other SP entities (analyzing user interaction scenarios, business process, system process and technical process scenarios, needed for service composition and delivery) and *(2) provision* of services and enablers (specifying business aspects of service provision and analyzing its economical consequences). These two models are used to "simulate" technical and business processes and estimate their business and technical performance, as detailed in the following subsections. After these two models have been presented, we discuss scenario-based simulation of periodic use of services and enablers, and focus on the service and enabler portfolio analysis and valuation. To stress once more, simulations analyze two usage aspects: *(1)* how service portfolios support end-user scenarios and *(2)* how enablers are used in service composition and delivery.

2.2 Modeling Usage of Services and Enablers

Scenario Modeling and Mapping

As mentioned before, we engage in process modeling (sketched in Fig.1), starting with user scenarios u_u (indexed by $u = 1 : N_u$), focusing on user activities and interactions t_m (indexed by $m = 1 : N_t$)). We continue with business process scenarios (specifying the support of services s_j (indexed by $j = 1 : N_s$)), system scenarios (determining the usage of enablers e_i (indexed by $i = 1 : N_e$)), and finish with technical scenarios (mapping the system process scenarios to various technological solutions and its resources h_l (indexed by $l = 1 : N_h$)).

Firstly we identify *a set of user / usage scenarios* u_u, representing the typical usage patterns (aiming at realizing user goals). Scenarios, modelled as template-based scenario stories, are mapped to MSC-based usage interaction sets, i.e. sequences of interactions invoked by the user (Eq. 1). Some of these interactions t_m can be supported by the SP functionality (services and enablers). Not all user interactions result in service interactions. Careful scenario analysis has to resolve these mapping issues.

$$u_u \xrightarrow{map} msc\,(t_1,\cdots,t_{N_t}) \xrightarrow{cum} \lambda_u^t = (\lambda_{1u}^t, \lambda_{2u}^t, \cdots, \lambda_{mu}^t, \cdots, \lambda_{N_t u}^t) \quad (1)$$

The λ parameters are cumulative message counts, summing up the number of message instances of the same message type as in a MSC diagram (Eqs. 1-4), convenient for further processing. In such a way we project the MSCs into their message totals, as sketched in Fig.1.

Mapping scenario interactions to services

It should be noted that a user interaction t_m could be mapped to multiple services s_j, representing alternative service supports. E.g. it is possible to send an instant message by various solutions, depending on the service context (SMS, MMS or e-mail). Mapping and cumulating scenario interactions t_m to service invocations s_j is shown in Eq. 2.

$$msc\,(t_1,\cdots,t_{N_t}) \xrightarrow{map} msc(s_1,\cdots,s_{N_s}) \xrightarrow{cum} \lambda_m^s = (\lambda_{1m}^s, \lambda_{2m}^s, \cdots, \lambda_{jm}^s, \cdots, \lambda_{N_s m}^s) \quad (2)$$

Mapping services to enablers

Enablers provide the SP functionality to the services and are measured in units of functionality (invocation counts). The service composition mechanisms [1,2] determine to a great extent the service and enabler level MSCs (Eqs. 2 and 3). In praxis the service composition can be a static software system (producing the same service and enabler composition sets in all scenario instances), or a dynamic (and respond to service context changes or changes in service domains (e.g. availability of enablers and services in service domains. Service composition might vary, resulting in different coefficients in Eq. 3. Each service s_j, is represented by an enabler MSC, representing a set of interactions that enablers e_j and service platform mechanisms must complete in order to deliver service s_j. So we can approximate the *service* s_j as a composition of a set of service components and *enablers* e_i expressed by:

$$s_j \xrightarrow{map} msc\,(e_{1j},\cdots,e_{N_e j}) \xrightarrow{cum} \lambda_j^e = \left(\lambda_{1j}^e, \lambda_{2j}^e, \cdots, \lambda_{ij}^e, \cdots, \lambda_{N_e j}^e\right) \quad (3)$$

s_j is mapped to enablers by using the vector $\lambda^e{}_{ij}$ – representing the needed amount of enablers for delivering the service s_j.

Mapping enabler level MSCs to SP capabilities and resources

Each enabler requires a dedication of a set of capabilities, which in turn consume resources, expressed by:

$$e_i \xrightarrow{map} msc\,(h_{1i}, \cdots, h_{N_h i}) \xrightarrow{cum} \lambda_i^h = \left(\lambda_{1i}^h, \lambda_{2i}^h, \cdots, \lambda_{li}^h, \cdots, \lambda_{N_h i}^h\right) \tag{4}$$

Such model transformations give us a possibility to specify interaction sets in such a way that they can be analyzed: qualitatively (including structural model analyses), and quantitatively - with help of various projections, e.g. financial (cost and revenue analysis) and resource and capability projections (analysis of usage of service platform capabilities and resources).

Simulation of Periodic Usage of Services and Enablers
Simulation of periodic use of services and other SP entities (enablers and resources) is the next step in our techno-business analysis. Important inputs are vectors specifying: contribution of services λ^s_{jm} to scenarios, enablers λ^e_{ij} to services and consumption of resources λ^h_{li} (by enablers). All the scenario instances are cumulated into the vectors λ^s_m,λ^e_j, λ^h_i, representing the totals of their contributions in scenarios. When projections are calculated for single instances of scenarios we continue with their periodisation. Interesting period units for this work are days, weeks, months and years. Simulation of periodic use of services, enablers and resources (denoted as *use* in Eq. 5) is made in the following way.

$$\begin{aligned} use_{period} &= \sum_{subperiod} use_{subperiod} \\ use_{subperiod} &= \sum_u w_u \cdot use_u \\ use_u &= f\,(u_u, \lambda_j^s, \lambda_i^e, \lambda_l^h) \end{aligned} \tag{5}$$

Where w_u is a weighting coefficient describing usage variation in scenario u, compared to the normative use, while *period* and *subperiod* are period units. We proceed toward valuation of roles, responsibilities and contributions after completion of simulations. A modeling entity under consideration determines the most convenient valuation method. In this analysis we focus on services and enablers. For them a cash flow-based valuation is the most appropriate. We explain it in the following section.

2.3 Modeling Service Provision

Business characteristics of services and service portfolios (e.g. business scale and scope parameters, network effect features, pricing, utility and price elasticity, service life-cycle effects, revenue and cost structure of services) have to be included in the techno-business models and scenarios. Majority is included either in service usage modeling or in service provision modeling (cash flow – based analyses of various business scenarios), which is shortly discussed in this subsection. The commercial service life-cycle is divided in three zones [9]: growth, maturity, and decline. In order to get realistic modeling estimates various information sources are combined, such as: market analyses, field work (empirical studies, field trials), survey techniques and expert analyses. In the backbone of all these analyses is a simple service provisioning model, shortly presented in the text below (described in [1,3]).

A service j generates a revenue v_j per unit of service. It is assumed that revenue v_j generated by a unit of service j is distributed among the actors who participate in creation of service j, performed using a vector of revenue shares γ_j. As detailed in [3] the expected return r_i of the service portfolio (containing M services s_j) is given by Eq.6, where x_{ij} specifies the amount of enabler e_i dedicated to provision of service s_j, and c_i includes both variable and projected/discounted fixed costs of enabler provision. More about the model is given in [1,3].

$$\bar{r}_i = \sum_{j=1}^{M} x_{ij} (\gamma_{ij} \mathrm{E}(\frac{v_j}{c_i \lambda_{ij}}) - 1) \tag{6}$$

3 Scenario-Based Analysis and Valuation

Model-based mapping and projection techniques can also be used for qualitative and quantitative comparison of various user, business, system and technological solutions. When *single scenario instances* or classes are analyzed the vectors $\lambda^s{}_s$, $\lambda^e{}_j$ and $\lambda^h{}_i$ are used. However, often the analysts direct their attention to the scenario sets (scenario trees), representing the complete solutions / designs, as discussed in this section. In these cases they group scenarios according to various modeling and analytical criteria, and develop the *scenario group representatives* λ^s, λ^e and λ^h. Scenario trees, illustrated in Fig.2, are used to compare alternative scenario branches (each with its own set of λ^s, λ^e and λ^h vectors), the difference of which we compare by traversing the branches and applying mappings (*map*) and projection techniques (*project*). In such a way we can compare alternative technical and business designs, as exemplified in the text below.

Comparison of alternative service supports of a user scenario

Various service portfolios can support the same user scenarios. That might change a user experience, utility of services and scenarios, quality of service, and overall value of the service support, but if the difference from the user expectations is not too big, these service portfolios could be considered as alternative service portfolios, capable of supporting the same usage scenarios. Alternative service portfolios can imply alternative business collaborations (resulting in different business roles and responsibilities for business actors). We can assume that the revenue share part for a business actor corresponds to the roles / responsibilities in the service provision, which can be interpreted by the amount of delivered / provided SP entities (services, enablers and resources), in these business and system collaborations. An example comparison of alternative service support of the same user scenarios will traverse two branches with the following nodes in Fig.2: *(a) 1→3→7→12→17* and *(b) 1→4→8→13→18* and analyze them by the following mappings and projections:

$$\lambda^t \xrightarrow{map} \lambda^s \xrightarrow{map} \lambda^e \xrightarrow{map} \lambda^h \xrightarrow{project} r \tag{7}$$

Comparison of various system collaboration patterns

Another interesting analysis can be a comparison of various system solutions, modeled as various system collaboration patterns (scenarios). Vectors λ^e represent various system solutions for delivering the same service (so the same service can be composed of different sets of enablers or components). It can in some cases also represent various technical solutions, where some enablers are replaced with others. Alternative service implementations (by alternative enabler portfolios) in Fig. 2 are represented by the following scenario paths: *(a) 2→5→9→14* and (b) *2→6→11→16*, and analyzed by the following mapping and projections:

$$\lambda^s \xrightarrow{map} \lambda^e \xrightarrow{map} \lambda^h \xrightarrow{project} r \tag{8}$$

Comparison of various technological realizations
Alternative technical implementations (technological variants) of enablers can also be compared with help of enabler-level MSCs and corresponding cumulatives, as illustrated in Fig. 2 by two branches: *(a) 5→9→14* and *(b) 5→10→15*:

$$\lambda^e \xrightarrow{map} \lambda^h \xrightarrow{project} r \tag{9}$$

4 Practical Case – SPICE Mobile Service Platform

We have used our scenario-based modeling approach in several practical cases. Here we will present one of them: a third party service provision platform – based on the SPICE service platform technology [1,7]. Four business actors collaborate in service provision. Each business actor is a separate business unit (enabler or service component provider), as illustrated in Fig. 3: *s*ervice provider a_1 (responsible for: e_1 – service composition and delivery enabler, and e_3 – A4C enabler), context provider a_2 (providing e_2 – service context enabler), network provider a_3 (delivering e_4 – network enabler) and content provider a_4 (responsible for e_5 – content enabler). The SP model

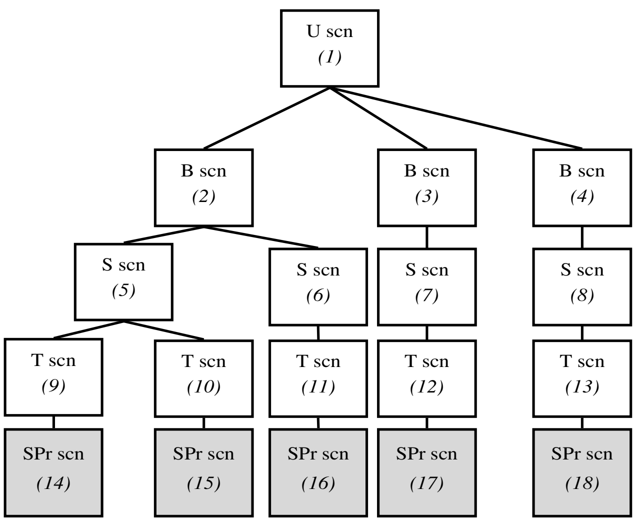

Fig. 3. Scenario tree, containing: user scenarios (*U scn*), business process scenarios (*B scn*), system process scenarios (*S scn*), technical process scenarios (*T scn*), and service provision scenarios (*SPr scn*). The nodes 1-13 focus on usage or services and enabler, while the nodes 14-18 focus on the business aspects of service provision.

contained service portfolios of 6 services, composed of 5 enablers. 5 user groups interacting in 8 usage scenarios have been simulated. Periodic service usage was simulated and calculated for a period of 10 years (anticipated service life cycle). Usage variations on daily, weekly and seasonal basis have been included in the simulations. Business case was limited to Norwegian telecom market and Norwegian users. Business risk estimates were obtained from comparable projects and service offerings. Valuation was based on cash flow projections, where various total and partial net present values have been calculated.

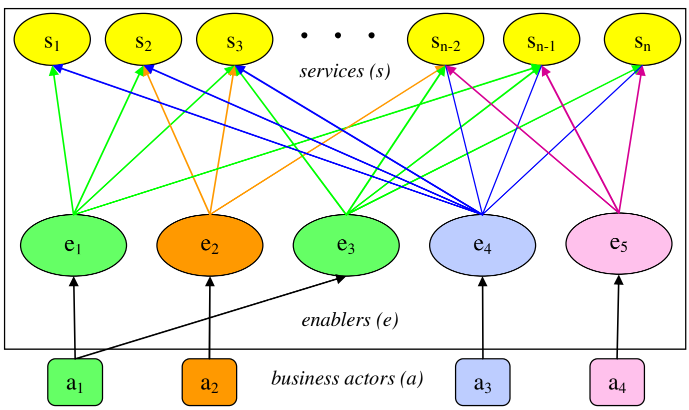

Fig. 4. Sketch of the SP entities, actors and their techno-business management structure

In the analyzed platform one part of enablers is already implemented, while the rest will be either developed (need investment estimation) or rented from other service providers. Three enablers required significant investment: service composition and

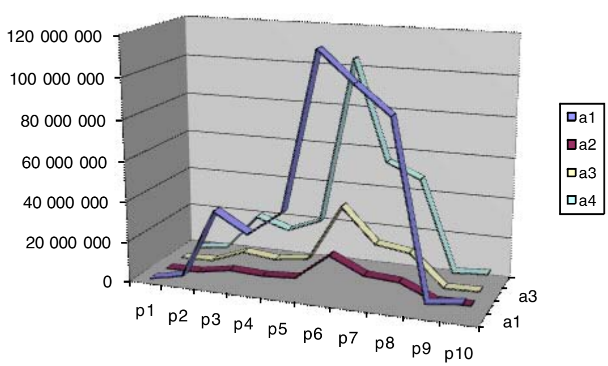

Fig. 5. Income profile for business actors a_i, providing one enabler each (as sketched in Fig. 4), with exception of service provider a_1, who provides two enablers: e_1 – service composition and delivery enabler, and e_3 – A4C enabler. The revenue units are anonymous. Anticipated commercial service life-cycle is 10 years ($p_1 - p_{10}$).

delivery enabler, and improvement of the existing context provisioning system. Investors typically seek the quantitative answers to the following questions. Which services users need in their scenarios (and to which extent)? Which enablers are used in services and to which extent? Are these scenarios equally profitable for enabler providers as for service providers? Which alternative service portfolios might support the same user scenarios? In the text below we give examples of results (with anonymized revenue values).

Income profile for business actors a_i is shown in Fig. 5 (corresponding well to income per enabler they provide). Namely, business actors provide one enabler each (as sketched in Fig. 5), with exception of service provider a_1, who provides two enablers: e_1 – service composition and delivery enabler, and e_3 – A4C enabler. Fig. 5 shows significant difference in cash flow profile of business actors. Service provider (a_1) and content provider (a_4) have much higher cash flow positions. It might indicate the business potential of delivering and integrating service platform solutions (and mechanisms), assuming that our pricing and revenue share mechanisms are used. It is still very unclear how the service discovery, composition, brokering and mediation (service platform mechanisms) will be priced. We believe that they should be treated equally to the other enablers (e.g. network and content provision) because of their importance for composition, delivery and provision of end-user services. At last, they wrap the SP functionality and provide it to the end-users.

Fig. 6 compares the income per user group, based on calculating the value of service support in user scenarios. The differences in profitability of various user groups can be noticed, and it partly reflects the importance and the role of service support for their activities (expressed as interaction patterns). User groups differ in: the services they choose, the way they use them, and the consumed quantities.

Fig. 7 shows differences in the income of various services. When we analyze Figs. 5-7 we have to keep in mind that the choice of service portfolio is not determined just by the business criteria. Additional important moments can play significant role: user preferences, service context, required services / enablers, and even government

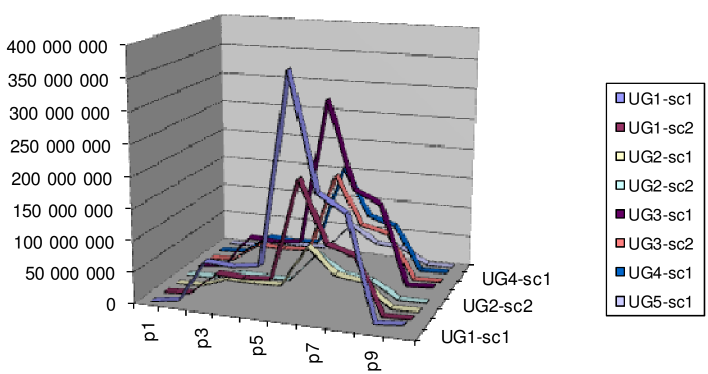

Fig. 6. Contribution of 4 user groups (UG_i) and their 8 usage scenarios (sc_i) to revenue creation (10 years service life-cycle). The revenue units are anonymous. Anticipated commercial service life-cycle is 10 years ($p_1 - p_{10}$).

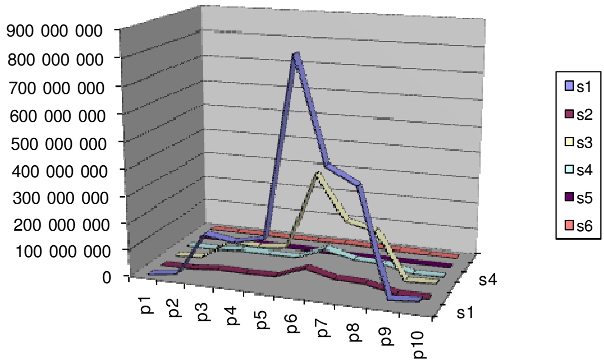

Fig. 7. Contribution of 6 services ($s_1 - s_6$) to revenue creation (10 years service life-cycle). The revenue units and service names are anonymous. Anticipated commercial service life-cycle is 10 years ($p_1 - p_{10}$).

regulation. Very often just a part of the service portfolio elements are profitable, some of them even create just costs, however their total contribution should result in a positive position for the business actor [3].

The TBA approach can also offer the sensitivity information for various entities: services and service portfolios, enablers and enabler portfolios, business actors, user groups and their usage scenarios, various business parameter sets (e.g. pricing schemas, cost factors and revenue models). This information complements well the business model information and gives an insight in some aspects of the risk of business actors.. This result requires that the scenario trees represent well the possible technical and business situations.

As this example and the theoretical part of this paper show, the TBA is a complex analysis, strongly dependent on realistic input coming from four domains (user, business, system and technical). With this limitation/assumption in mind, we believe that it offers a reasonable framework for integrating scenario information in a systematic quantitative analysis. Its further strength and practical relevance lies in the choice of standard and well accepted analytical techniques. However, we stress that we consider the TBA as a complement to other methodologies used in business modeling and analysis. It is particularly convenient for the cases of conceptually new, complex designs, without historical data, where the other methodologies exhibit high degree of speculation, and where some quantitative estimates might help.

5 Conclusion

We believe that the value of business models increases when supported by well founded quantitative (e.g. financial) estimates. This is particularly needed for the conceptually new designs, without historical data, and with difficulties in using comparables. It is difficult to create technical and business analyses for complex, distributed, heterogeneous and pervasive services. The TBA framework might offer some of the answers and estimates. The whole TBA is quite a complex analysis, strongly dependent on realistic input, coming from various analytical domains. Its strength and practical relevance lies in the choice of standard and well accepted analytical techniques. We are continuing improving the approach and simplifying it for the practical use in service and SP analyses.

Acknowledgement

Part of this work is supported and financed by the ISIS project (Infrastructure for Integrated Services), financed by the Norwegian Research Council (NFR # 180122). Author is grateful for the support.

References

1. Zoric, J., Strasunskas, D.: Techno-Business Assessment of Services and Service Platforms: Quantitative, Scenario-Based Analysis. In: Cunningham, P., Cunningham, M. (eds.) ICT-Mobile Summit 2008, Conference Proceedings, IIMC International Information Management Corporation (2008) ISBN: 978-1-905824-08-3
2. Zoric, J.: Practical quantitative approach for estimating business contribution of enablers and service platforms. In: Proceedings of ICIN 2008, Services, enablers and Architectures Supporting Business Models for a New Open World, Bordeaux (2008)
3. Gaivoronski, A.A., Zoric, J.: Evaluation and Design of Business Models for Collaborative Provision of Advanced Mobile Data Services: Portfolio Theory Approach. In: Proceedings of the 9th INFORMS Telecommunications Conference, Telecommunications Modeling, Policy, and Technology. Springer Verlag's Book of Conference Proceedings, Springer, Heidelberg (2008)
4. Gordijn, J.: Value-based Requirements Engineering – Exploring Innovative E-Commerce Ideas. PhD thesis, Vrije Universiteit, Amsterdam (2002)
5. Akkermans, H., Baida, Z., Gordijn, J.: Value Webs: Ontology-Based Bundling of Real-World Services. IEEE Intelligent Systems 19(44), 23–32 (2004)
6. Osterwald, A., Pigneur, Y.: E-Business Model Ontology for Modeling E-Business. In: Proceedings of the 15th Bled Electronic Commerce Conference E-Reality: Constructing the E-Economy, Bled (2002)
7. Ballon, P., Gaivoronski, A., Walravens, N., Zoric, J.: Structural and Quantitative Evaluation of Multi-Actor Business Models for Mobile Service Platforms. In: Proceedings of ICT-Mobile Summit (2008) ISBN: 978-1-905824-08-3
8. Weill, P., Vitale, M.R.: Place to space: Migrating to E-Business Models. Harvard Business School Press, Boston (2001)
9. Kotler, P., Armstrong, G.: Principles of Marketing, 10th International edn. Pearson Education Limited (2004) ISBN 9789582850972
10. ITU-T. Languages for Telecommunication Applications – Message Sequence Chart (MSC), ITU-T Recommendation Z.120, Geneva (1999)
11. Jacobson, I.: The Use Case Construct in Object-Oriented Software Engineering. In: Carroll, J.M. (ed.) Scenario-Based Design: Envisioning Work and Technology in System Development, pp. 309–336. John Wiley and Sons, Chichester (1995)
12. Leite, J.C.S.P., Hadad, G., Doorn, J., Kaplan, G.: A Scenario Construction Process. Requirements Engineering 5, 38–61 (2000)
13. Schneider, G., Winters, J.: Applying use cases: a practical guide. Addison-Wesley, Reading (1998)
14. Pateli, A., Giaglis, G.M.: A framework for understanding and analysing E-Business models. In: Proceedings of the 16th Bled Electronic Commerce Conference - E-Transformation, pp. 329–348 (2003)