BASE/X business agility through cross-organizational service engineering : the business and service design approach developed in the CoProFind project

Research School for Operations Management and Logistics

BASE/X
Business Agility through Cross-Organizational Service Engineering
The Business and Service Design Approach
developed in the CoProFind Project

Beta Working Paper series 414

BETA publicatie	WP 414 (working paper)
ISBN	
ISSN	
NUR	982
Eindhoven	March 2013

BASE/X

Business Agility through Cross-Organizational Service Engineering

The Business and Service Design Approach developed in the CoProFind Project

Paul Grefen, Egon Lüftenegger
School of Industrial Engineering
Eindhoven University of Technology

Eric van der Linden, Caren Weisleder
De Lage Landen International B.V.

Abstract

Many business domains are currently characterized by a move from an asset-orientation to a service-orientation: customers recognize that business value is not in owning assets, but in using the services offered by assets (which they do not need to own). This creates service-dominant business markets. The not-so-physical characteristics of these markets give rise to high levels of dynamism. This places high demands on the agility of service providers operating in these markets. These providers find their agility, however, heavily constrained by the business and IT platforms they use to deliver their services.

This paper proposes the *BASE/X* framework for agility in service-dominant business, which has been developed in the CoProFind project, a strategic collaboration between Eindhoven University of Technology and De Lage Landen International B.V. This framework makes two fundamental distinctions in the setup of a service-dominant business environment. Firstly, the framework separates two concepts in business conception: business strategy and business models. Business strategy is linked to the identity of an organization and hence is relatively stable in time. Business models specify the operation in the here and now of changing markets, and hence change frequently. Secondly, the framework separates two concepts in business implementation: service capabilities and service offerings. Service capabilities are relatively stable in time as they are based on the resources of an organization (both human and non-human). Service offerings change over time – they revolve with market dynamics as operationalization of network-centric business models of a provider. The framework provides a basis for structural agility by on the one hand coupling the stable elements to each other and on the other hand the fluid elements.

The concepts of *BASE/X* are combined in a service-dominant business structure that provides an operationalization of the framework. As this is a structure with stable outer layers and flexible inner layers, we call this the *business sandwich model*. Applying this sandwich model to the way business is organized thoroughly changes the traditional way of thinking in terms of decision horizons where it comes to implementing agility.

The business sandwich model describes the business view on agility. To enable automated support of agile business, the sandwich is mapped to an information system stack. This mapping how the business is connected to automated business applications at each of the layers of the sandwich. The information system stack is in its turn mapped to an infrastructure stack, which describes the basic IT platforms that form the implementation basis for the business applications.

The *BASE/X* framework described in this document is a structure for the development of new service-dominant business: business strategy, business models, their operationalization in service compositions, business services, and their implementation in state-of-the-art automated service management platforms. To aid in the use of this structure, the framework embeds tools that are tailored to business design in a service-dominant context. Application of the framework and its tools lead to well-structured management of the complexity of service-dominant business and short time-to-market of new business models. The approach is illustrated by a case study of an advanced travel industry service orchestrator.

Table of Contents

List of Figures

List of Tables

1 Introduction: agile, service-dominant business networks

In this introduction, we set the scene for the rest of this document. First, we discuss the rise of the service-dominant (SD) business paradigm. This business paradigm implies a high level of agility from business models. Next, we discuss the rise of dynamic business networks that are required for the realization of solution-oriented, service-dominant business. Then, we explain how this document contributes to the implementation of the SD paradigm in such a way, that a high level of agility is reached in a complex business environment without putting structure at risk. This *BASE/X*[1] approach is centered on the concept of a layered structure that we call the service-dominant *business sandwich*. We end this introduction with an outline of the structure of this document and the running case we use to illustrate concepts and techniques throughout the document.

1.1 The rise of the service-dominant business paradigm

Many business domains are currently characterized by a move from an asset-orientation to a service-orientation. This means that customers recognize that business value is not in owning assets, but in using the services offered by assets (which they do not need to own). Examples can easily be found in a wide spectrum of domains. In the goods logistics domain, a focus on physical transportation vehicles is being replaced by a focus on logical transportation services. In the personal mobility domain, a focus on providing cars (e.g., in lease constructs) is replaced by services that enable users to arrive at the right place at the right time. In the entertainment industry, a focus on physical entertainment carriers (such as CDs and DVDs) is replaced by compound services that offer ubiquitous entertainment experiences. A last but certainly not least example can be found in the IT industry, where physical computing assets (such as server parks) are being replaced by virtualized server facilities (a development commonly known as Cloud Computing [Arm09]). In all these examples, the value is not in *owning* the asset, but in *using* it – referred to as *value-in-use*. As such, given a business problem, owning an asset is typically more *means-oriented*, where a value-in-use is more *solution-oriented*.

Business in which service-orientation is the basis for full operation is termed *service-dominant* business. Service-orientation means taking services as the basic mechanism for interaction. We define the service-dominant business paradigm as follows:

> *The service-dominant business paradigm is the style of defining and implementing business models such that the following three characteristics apply:*
> *(1) value-in-use is the main entity that is exchanged (traded),*
> *(2) this value-in-use is encapsulated in a set of services, and*
> *(3) these services are offered to a market through a service delivery mechanism.*

[1] *BASE/X* is the acronym for **B**usiness **A**gility through **S**ervice **E**ngineering in a **Cross**-Organizational Setting

As such, the service-dominant business paradigm is a way of thinking about doing business that uses the service-oriented paradigm as an underlying way of thinking about delivering business value. In other words: where the service-oriented paradigm is mainly about operations (*how* to do things, often related to information technology such as Web services [Alon04]), the service-dominant business paradigm adds thinking about business (*why* to do things, related to market dynamics).

1.2 Agility in service-dominant business

The service-dominant business paradigm can replace the traditional asset-dominant business paradigm in many situations[2]. The asset-dominant paradigm implies relatively stable markets, as business models are at least partly dictated by ownership and whereabouts of physical assets – and the need to change these two characteristics in the enactment of business processes. The fact that physical assets are not so easy to acquire and relocate makes business relatively slow. In service-dominant business, however, not the asset but its use is traded, eliminating the need for ownership exchange and often also the need for relocation of the asset because of the ownership exchange. These not-so-physical trading characteristics of service-dominant business create highly dynamic markets: service consumers expect providers to swiftly react (or even pro-act) to new developments. These markets therefore place high demands on the agility of their service providers. Where in traditional markets, providers might have years to react to changes in these markets, in service-dominant markets, this reaction time may be reduced to months, weeks or even days only.

However, service providers in these dynamic markets often find their agility heavily constrained by the limitations of the automated platforms they use to deliver these services. This especially is problematic, as increasing dynamism in a market typically requires higher levels of efficiency in dealing with changes. These higher levels of efficiency require higher level of automation, which obviously are strongly dependent on the capabilities of the automated platforms available.

1.3 The rise of dynamic business networks

Above, we have seen that service-dominant markets require high levels of agility from the players in these markets, i.e., these players must be able to adapt dynamically to changes in these markets. A way to deal with this high level of dynamism is to not see the services delivered as monoliths that are completely produced in-house, but as flexible compositions of sub-services, part of which are produced in-house and part of which are produced by third parties in the market (which become partners for this reason). In other words, players in service-dominant markets typically engage into dynamic business networks in which parts of offered services can be delivered by business partners. These networks evolve as market conditions evolve. Examples of this relationship are: new requirements to a value-in-use in a market may lead to the inclusion of a new service provider into a network that can help meet these requirements; an emphasis shift in a

[2] In some situations, the service-dominant paradigm may augment the asset-dominant paradigm, thereby creating two lines of business - this line of thinking is further discussed in Section 12.2.

market from quality to price of a value-in-use may lead to the replacement of a high-quality service provider by a high-efficiency service provider.

Apart from the agility requirement discussed, we also see the trend from individual services to complete *solutions*. Where services provide consumers of those services with some specific value-in-use that contributes to the operation of those consumers, solutions provide consumers (users) with a value-in-use that solves a problem for them in an integrated way. As such, a solution consists of an integrated set of services. As problems to be solved get more and more complex in the modern economy, their solutions become more and more complex too. Hence, in many situations, not all services in a solution can be delivered by a single provider. This leads to necessity to have services provided by multiple providers, and hence to the creation of service networks. Where problems evolve in markets, so do solutions, and service networks become dynamic service networks.

Finally, we see the emergence of multi-sided business models [McKin12] that often have a service-dominant character. Multi-sided business models generate multiple revenue streams through collaboration with several classes of partners (may rely on multiple, interrelated supply chains, in traditional speak). The collaboration with multiple classes of partners gives rise to complex business networks – combined with service-dominance, this can give rise to dynamic business networks.

Concluding, we observe the rise of dynamic business networks for a number of main reasons: to deal with the agility of service-dominant business, to deal with the complexity of solution-oriented business, and to deal with multi-sided business models.

1.4 Contribution of this document

This paper proposes the *BASE/X* framework for agility in service-dominant business that explicitly addresses the issues outlined above. The framework does so by making two fundamental distinctions in the setup of a service-dominant business environment:

- Firstly, *BASE/X* separates two concepts in business conception: business strategy and business models. Business strategy is linked to the identity of an organization and hence is relatively stable in time. Business models specify the operation in the 'here and now' of dynamic markets, and hence have to be changed (or replaced) frequently. The two concepts are therefore fundamentally different, but are explicitly linked.
- Secondly, *BASE/X* separates two concepts in business implementation: service capabilities and service offerings. Service capabilities are relatively stable in time – they evolve with the identity of a provider as defined by its business strategy in the context of business networks. Service offerings are fluid in time – they revolve with market dynamics as operationalization of network-centric business models of a provider. Also here, the two concepts are fundamentally different, but explicitly connected.

The framework provides a basis for structural agility in service-dominant business by arranging the above four concepts in a novel way:

- On the one hand, the stable concepts from business conception and business implementation are coupled to each other and on the other hand, the dynamic concepts are coupled to each other. The concepts are operationalized as elements in a

novel service-dominant business framework called the *sandwich model*, in which the dynamic elements are sandwiched in between the stable elements. This sandwich model thoroughly changes the traditional way of thinking in terms of decision horizons and business agility.

The developed model is linked to automated support in a clearly structured way. In doing so, each of the four concepts above is mapped to classes of information systems (including but not limited to automated applications). These classes are again related to automated platforms, i.e., general-purpose systems that support specific applications.

The *BASE-X* framework integrates business engineering with information systems engineering in a clear structure that allows for separation of concerns to deal with the inherent complexity of modern, service-dominant business. As such, the *BASE-X* framework can be seen as a contemporary, service-dominant operationalization of strategic business-IT alignment models (such as the well-known model by Henderson and Venkatraman [Hend93]). *BASE-X* is an operationalization because it covers a description of a way of working including a unified tool set used in this way of working.

1.5 Structure of this document

The structure of this document is as follows. In Chapter 2, we discuss how we move from a traditional pyramid of decision making dynamics in a business environment to a new pyramid. This pyramid is the basis for our service-dominant business sandwich model, which we discuss in Chapter 3. In the next four chapters, we pay more attention to the four main parts of this model. Chapters 4 and 5 discuss the bread of the sandwich, i.e., the stable part of the sandwich model. Chapters 6 and 7 discuss the filling of the sandwich, i.e., the part of the sandwich model that can follow market movements in an agile way. Chapters 8 and 9 add information system and platform support to the four-layer model. We conclude the main text of this document in Chapter 10. The appendices to this document further elaborate some of the elements.

1.6 Running case: TraXP

The concepts in this document are illustrated by a running case from the travel industry: TraXP. TraXP is a service-dominant spin-off of a major player in the international travel market. TraXP will offer the complete **Tra**vel e**XP**erience (hence their name) in both the B2C and B2B travel markets. Their working slogan is: 'TraXP does not support a trip but a traveler'. In the travel market, TraXP compares to companies like Expedia[3] and Booking.com[4], like in the music market, Spotify[5] compares to companies like Apple's iTunes music store[6] and Amazon[7] (at least in their original setup, as business models are changing continuously).

[3] See www.expedia.com

[4] See www.booking.com

[5] See www.spotify.com

[6] See www.apple.com/itunes

The high-level business vision of TraXP is shown in Figure 1. They realize that the travel market is a very dynamic market, in which offerings change fast. Therefore, the mission of TraXP is to realize their vision in a highly agile way.

- Offer a seamless, complete travel experience to various customer groups
- Base that on advanced customer profiles that record preferences
- Use a broad spectrum of travel services that can be flexibly combined
- Execute the combinations in a highly networked, real-time fashion
- Be the central orchestrator in these networks

Figure 1: TraXP vision

[7] See www.amazon.com

2 The conceptual background: from an old to a new pyramid

In this chapter, we move from a traditional way of looking at decision horizons in business design and operation to a new way of looking at these horizons that is more suitable for a service-dominant world. We present both approaches in the well-accepted form of decision pyramids that span the strategy to operations spectrum.

2.1 The old pyramid

In this section, we take a look at a well-known pyramid that distinguishes three levels in business decision making. This pyramid has been used for decades in asset-dominant business, but creates problems in a service-dominant business. Below, we first explain the structure of the pyramid. Then, we discuss its problems.

2.1.1 The layers in the pyramid

A traditional way of looking at layered decision horizons in business management is shown in Figure 2. Here, we see the well-known separation of decision making into strategic decision making, tactical decision making, and operational decision making.

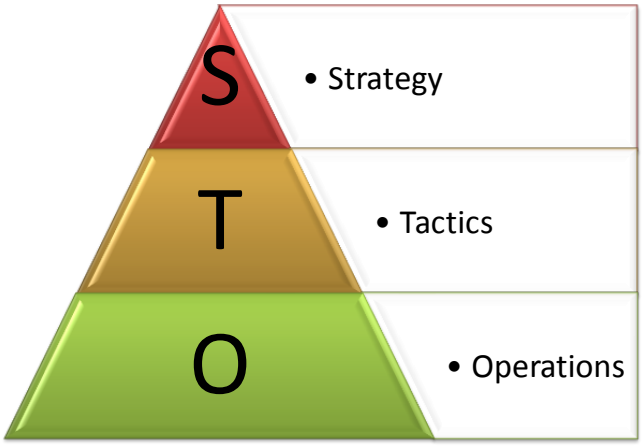

Figure 2: the old pyramid

Strategic decision making is aimed at long-term issues, typically with a horizon in terms of years. In strategic decision making, large investments are considered, such as the creation or acquisition of new physical business infrastructures. The definition of new business models is typically also considered at this layer – in traditional contexts, business models do not change very often. It sets the context for tactical decision making.

Tactical decision making aims at medium-term issues, typically with a horizon in terms of weeks or months. In this layer, setting the parameters for deployment of business infrastructures is considered. Also, parameterization of business models (the conception of

which has takes place at the strategic level) takes place at this level. Tactical decision making sets the context for operational decision making.

Operational decision making aims at day-to-day business operation, typically with a horizon of hours (or even shorter) to days or weeks (depending on the business domain). Here, individual customer orders are taken into consideration.

2.1.2 Problems in using the old pyramid

In modern-day economy, this traditional pyramid faces a number of challenges that often lead to problems.

Firstly, we see that the frequency of strategic and tactic decision making increases. This is caused by the fact that business markets change at an increasing pace that is fueled by new technologies, global competition, more customer demands, etcetera. This development 'pushes down' issues at the strategic and tactic levels in the pyramid: from a decision-making point of view, part of strategic decision making becomes tactic and part of tactic becomes operational. Long-term strategic decision making remains only at a high level of abstraction. Operational decision making becomes more and more automated.

Secondly, we see that strategic and tactic decisions have to become more and more flexible in their content nature to reflect swiftly changing market environments. Hence, these decisions may not match the possibilities offered in the operational layer: we are dealing with legacy structures in business that are based on historically taken decisions at the strategic and tactic levels. Operations are not able to change "on demand" and therefore they slow down the company in development.

Thirdly, the traditional pyramid is not tuned to the service-dominant business paradigm, which is the basis of modern business thinking. Using the service-dominant business paradigm means thinking in terms of delivering business services instead of delivering business products. The primary focus in service-dominant business is on composing usable functionality (which may be delivered by business products) rather than on constructing business products themselves (the functionality of which may be used). Bringing service-dominant business to the asset-oriented business world, the emphasis completely shifts from (physical) business assets to the value-in-use of these assets.

2.1.3 Navigating the old pyramid

Given the problems discussed above, one may consider how to take decisions in the old pyramid, i.e., how to navigate its layers to deal with these problems. Putting things black and white, we can see two alternative approaches: top-down and bottom-up (as illustrated in Figure 3).

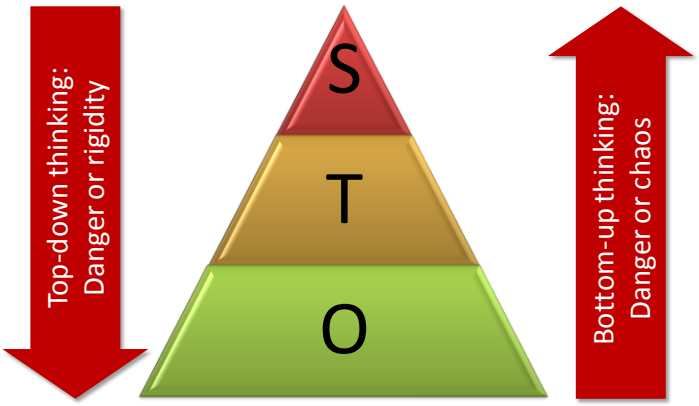

Figure 3: traversing the old pyramid

In the traditional approach of structured design, one goes from requirements to implementation, i.e., strictly top-down from strategy to operations in the pyramid. Given the complexity of modern business operation, however, this process takes long. This creates an imminent danger of rigidity in business design.

As a reaction to this, many organizations resort to strict bottom-up design, i.e., thinking from small capabilities at the operational layer into new possibilities at the strategic layer. Given the fact that operations are often quite diverse and silo-oriented, this process can lead to either disconnected business or many one-to-one business interfaces. In other words, this creates an imminent danger of chaos in business design.

So one can conclude that 'the truth is somewhere in the middle'. Working in a structured way with this new truth requires a new pyramid. We introduce this pyramid below.

2.2 A new pyramid

Above, we have seen that the traditional view on business decision making (organized into the three-layer pyramid) leads to a number of problems in modern-day, service-dominant business management (i.e., the design and operation of service-dominant business models). These problems pose challenges for the design of business in this modern context. To attack these challenges, we propose a new four-layer business pyramid that is specifically designed for agile, service-dominant business.

2.2.1 The layers in the pyramid

The pyramid is shown in Figure 4. In this pyramid, we have introduced a new layer (in between the traditional strategic and tactic layers) that captures a blend of strategic and tactic decision making. Further, we have specialized all layers towards the service-dominant (SD) business paradigm, i.e., made the SD concept explicit in them.

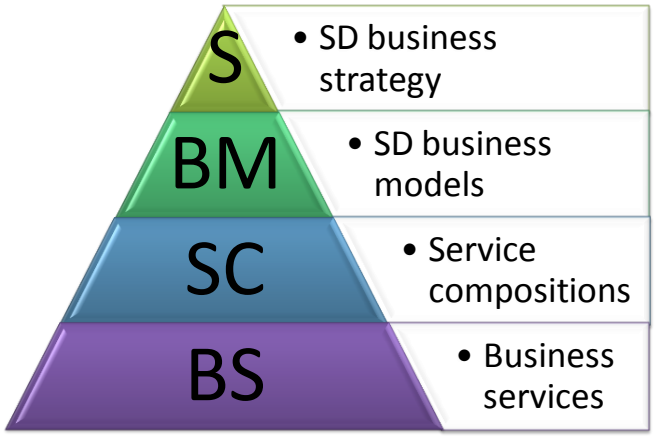

Figure 4: the new pyramid

The service-dominant business strategy layer contains decision making with respect to long-term business strategy, i.e., it defines the identity of a business organization in terms of the high-level services the organization delivers to its context (including but not limited to its market).

The service-dominant business models layer contains decision making with respect to the medium-term business course, which is expressed in business models defined in terms of services. Note that business strategy and business model are essentially different concepts: a business model is derived from a business strategy by making a set of choices [Shaf05]. Making different sets of choices, multiple business models can be derived from a single business strategy. We revisit this relation later in this document in more detail.

The service compositions layer contains decision making with respect to the implementation of business models. A business model is implemented by composing a number of services from the business services layer. The service compositions implement the high-level services that are offered to a market and that are used to exchange value-in-use (as defined in Section 1.1). In other words, the service compositions define the service-oriented, operational business offerings.

The business services layer contains the set of elementary services a business organization can perform, i.e., the service-oriented encapsulation of its core business capabilities (also referred to as business competences).

Summarizing, we can state that the new pyramid coupled two kinds of engineering (as shown in Figure 5): business engineering in its top half (about the *what* of service-dominant business) and operations engineering (about the *how* of service-dominant business) in its bottom half.

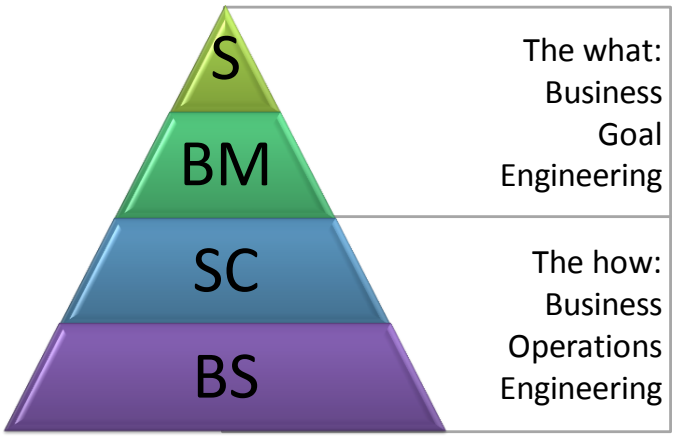

Figure 5: two kinds of engineering in the pyramid

2.2.2 Decoupling decision cycles

Given the layers in the new pyramid, two kinds of decision making about new business design can be decoupled, as illustrated in Figure 6.

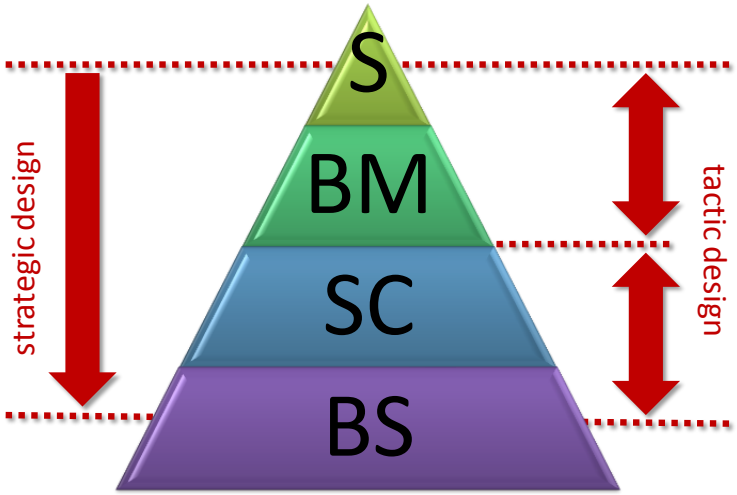

Figure 6: two forms of business design in the new pyramid

On the one hand (shown in the left side of the figure), the long-term strategy is the basis for the design of the set of elementary services in the business services layer – the strategic business design. As these elementary services are based on the core capabilities of an organization (and hence on its infrastructure in human and non-human resources), they should evolve rather than revolve, based on the organizational identity defined in the strategy.

On the other hand (shown in the right side of the figure), there is the design of new business models and their service compositions, which should follow the dynamism of market contexts. These business models are based both on the strategy (top-down decision making) and on the possibilities of business services (bottom-up decision making). In other words: new business models are designed by matching business interest and relevance (we call this *desirability*) on the one hand and practical business possibility (we

call this *feasibility*) on the other hand. This means that common ground has to be found between what an organization *wants* to achieve and what it actually *can* achieve. Focusing too much on desirability can lead to designing castles in the sky. Focusing too much on feasibility can lead to sub-optimal performance, i.e., mediocrity in a market. Business feasibility can be greatly increased by including business services offered by external parties into possible service compositions – we address this issue in more detail in the sequel of this document.

We can represent the above approach in two decoupled business design cycles (illustrated in Figure 7): a strategic design cycle going through the business strategy and the business service layers, and a tactical design cycle going through the business model and the service composition layers. The two design cycles are synchronized by two confrontation points: a confrontation of goals between business strategy and business models (about the 'what' of business), and a confrontation of means between the service compositions and the business services (about the 'how' of business).

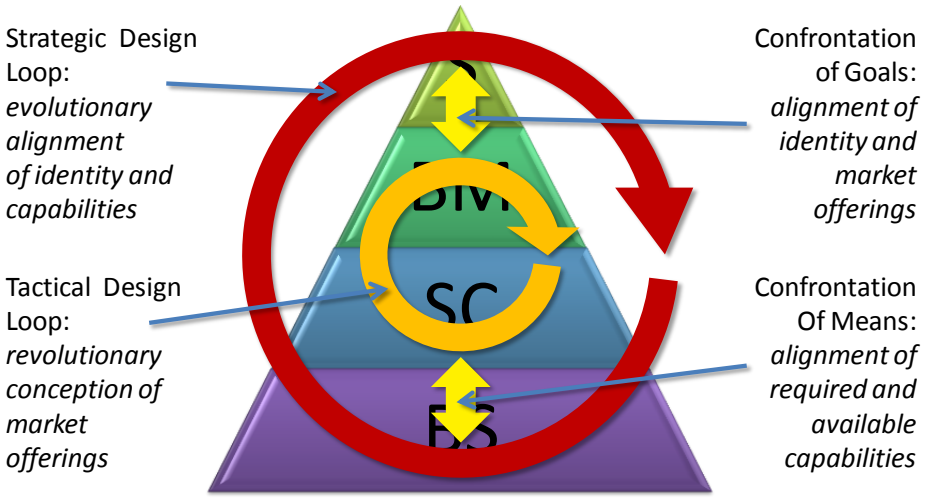

Figure 7: strategic and tactical design cycles in the new pyramid model

The strategic business design loop has a relatively low frequency, but contains decisions with a relatively high impact: they are long-term and may relate to large investments. The tactical design loop has a relatively high frequency, but contains decisions with a relatively low impact: they are medium-term and typically do not require large investments.

2.2.3 Practical business design sequences

Above, we have seen how we can (and even need) decouple strategic from tactical business design (see Figure 7). The question now is what the practical order of design is to arrive at business models and their implementation in business compositions – as here the value is from the service-dominant perspective of thinking. Therefore, we here concentrate on how to proceed in tactical design once its context in strategic design has been set. In other words: given the fact that we have a stable strategy layer and a stable business services layer, we explore how business models and service compositions can come into existence by navigating the two middle layers of our pyramid. The purpose of this section is to provide a complete overview of design sequences. The applicability of

each of the approaches depends on a specific business context (both the kind of business and the stage of SD business development an organization is in).

We can analyze the set of design sequences in a well-structured way by observing that business design can in principle start by thinking from any of the four layers in the pyramid. We can start thinking from one of the two stable layers to arrive at new temporary models (structurally deducing the new from the existing, so to say), or we can start designing new models in one of the two flexible layers more freely and then massage them towards the stable layers (converging the new to the existing, so to say).

As observed, we can start designing from any of the four layers of the pyramid, so we have four possible design sequences. Each sequence is related to a specific design goal, as we will see a bit later. The four design sequences are shown in Figure 8. Each of the four sequences is rooted, though, in either the strategy or the services layer, as these are the stable layers of the pyramid. Therefore, the sequences are labeled as two pairs of two variations. We explain these four sequences below.

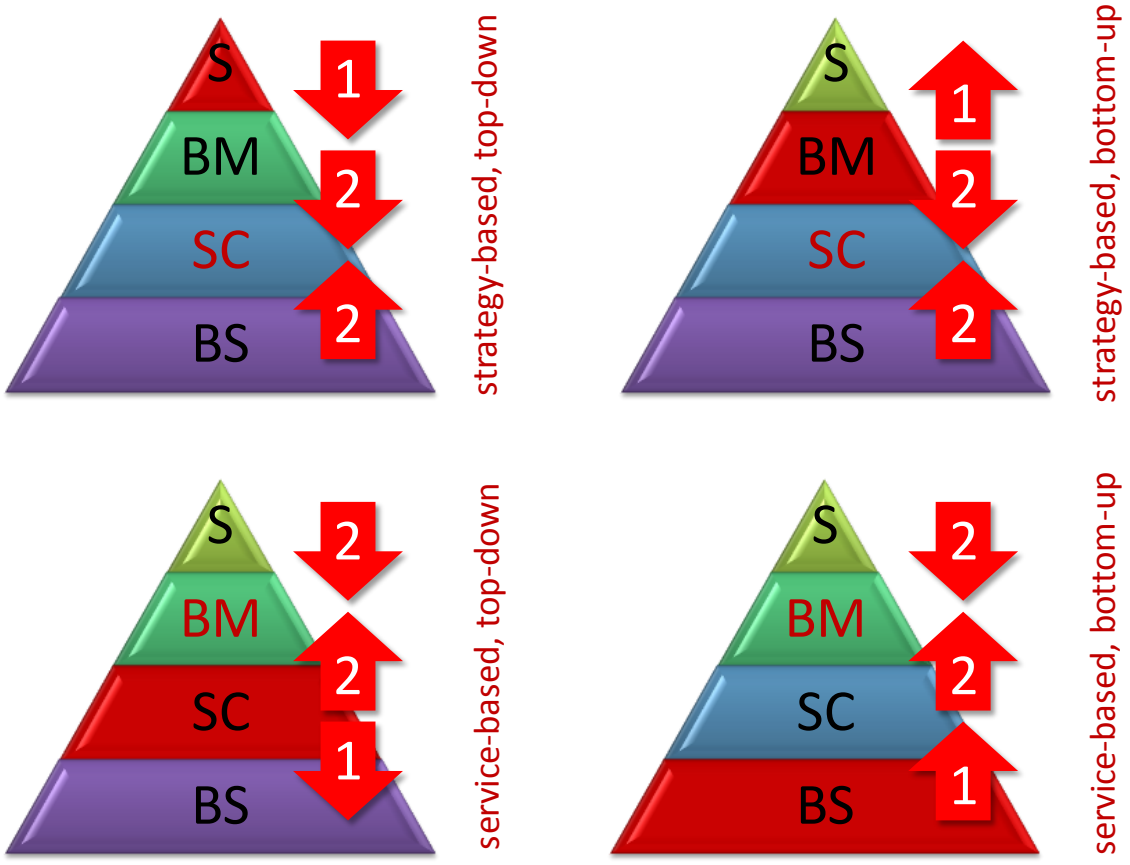

Figure 8: four practical sequences for business design

Strategy-based, top-down: This sequence is rooted in and starts business design from the strategy layer. This sequence is identity-oriented: it starts from the identity of a business organization in a market. In the first step, it derives new business models from a strategy (e.g., triggered by the fact that the strategy has evolved). In the second step, identified business models are translated to their operationalization by deriving the service compositions. This means confronting the requirements of the business models

with the capabilities of the business services – hence, the confrontation is in the service composition layer (does the *what* indeed map correctly to the *how*?).

Strategy-based, bottom-up: This sequence is rooted in the strategy layer, but starts from the business model layer. It is goal-oriented. In the first step, it tries to find new business models (goals) and consolidate these models in the strategy layer (where conflicts may trigger thinking about evolution of the strategy). In the second step, consolidated business models are translated to their operationalization in service compositions, which implies a confrontation in the service composition layer.

Service-based, top-down: This sequence is rooted in the business service layer, but starts from the service composition layer. It is operations-oriented, as the focus is on how to sequence business capabilities: it identifies new possibilities for service composition (e.g., by transforming existing compositions in a structured way to test for new ideas). In the first step, compositions are mapped to the business service layer. In the second step, compositions are mapped to business models, which are checked with the strategy layer for desirability in the context of strategy and identity of the organization. Hence, confrontation takes place at the business model layer (does the *how* indeed map correctly to the *what*?).

Service-based, bottom-up: This sequence is rooted in and starts from the business service layer – hence this sequence is capability-oriented. In the first step, business services are combined into possibly interesting service compositions – this is thus a bottom-up process. Making combinations can be performed free-form, or by going structurally through a service catalog (we discuss service catalogs in Section 5.6). The service compositions are mapped to business models, which are checked with the strategy layer. Confrontation takes place at the business model layer.

As one can observe from the above descriptions of the sequences, the first step in any sequence is always the creative step. In this step, new models are designed with few constraints. The second step is always the engineering step: here, a confrontation of two layers takes place (confronting the *how* with the *what* or vice versa) to assess the desirability (*how* to *what*) or feasibility (*what* to *how*) of a business design. Note that the layer identified for a confrontation is one of the two dynamic layers of the pyramid – these two confrontation layers coincide with the 'inner ends' (pointing to the tactical design cycle) of the two confrontation arrows in Figure 7.

The use of a specific design sequence is chosen depending on the goal of the business (re)design. In practice, several design sequences may need to be used in an iterative, interleaved way to keep external (with a market) and internal (between strategy and capabilities) alignment of business. An overview of the main characteristics of the four design sequences is given in Table 1.

Sequence (root, direction)	*Orientation*	*Goal*	*Start*	*Confrontation*	*Activity*
Strategy-based, top-down	Identity	Alignment of business with strategy	Strategy layer	Service composition layer	Structural deduction
Strategy-based, bottom-up	Goal	Exploration of new business opportunities	Business model layer	Service composition layer	Organized idea creation
Service-based, top-down	Operations	Exploration of new business configurations	Service composition layer	Business model layer	Organized idea creation
Service-based, bottom-up	Capabilities	Alignment of business with capabilities	Business service layer	Business model layer	Structural compo-sition

Table 1: overview of business design sequences

Note that the four design sequences are in fact ways to align business and organization capabilities (offered as services, which may again encapsulate assets – see Section 1.1). As such, we can see this as a business-resources alignment approach, in some ways comparable to more traditional ways of business-IT alignment (see e.g. [Hend93]).

2.3 Conclusion

In this chapter, we have observed that the traditional three-layer pyramid (strategy, tactics, operations) does not provide the right structure for agile, service-dominant business design. We have therefore replaced this pyramid with a new, four-layer pyramid which is centered at services thinking. We have shown how business design can actually navigate the pyramid.

The new pyramid has remained abstract, however, in this chapter: the layers of the pyramid are black boxes. To operationalize the business design approach based on the pyramid, we have to open the black boxes and see what is inside. To do this, we transform the pyramid into a service-dominant business sandwich model in the next chapter.

3 The main structure: the service-dominant business sandwich

In the previous chapter, we have constructed a new decision pyramid for a service-dominant world, consisting of four layers. We have shown that the top and bottom layers of this pyramid relate to stable structures: respectively the identity and the core capabilities of an organization. In between the two stable layers, we identified two layers that relate to changeable, i.e., agile structures that define and implement the external business offerings: business models and service compositions. This structure resembles a sandwich, consisting of two slices of bread with a selection of two fillings in between (see Figure 9). Hence, we paraphrase this conceptual business structure as the *service-dominant business sandwich*.

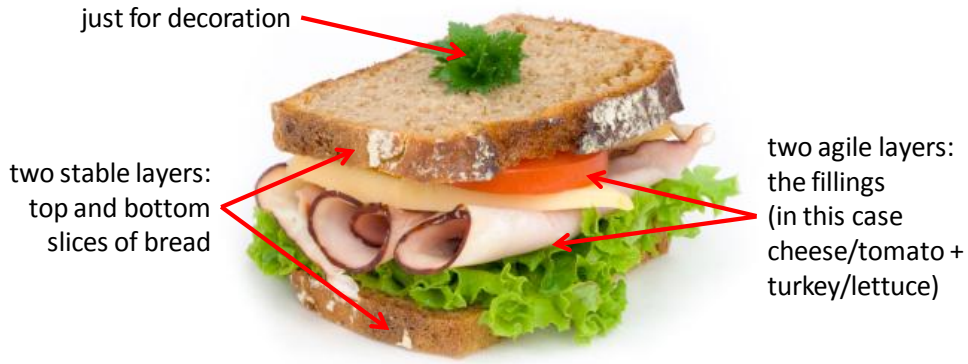

Figure 9: illustration of the service-dominant business sandwich

In this chapter, we take a closer look at this sandwich. We distinguish between the build-time view, focusing on decisions with respect to the contents of the four layers, and the run-time view, focusing on executing business using the four layers. The build-time view is related to the business design cycles and sequences we have discussed in the previous chapter (see Section 2.2.2). After this chapter, we discuss further details of the four stable respectively agile layers in the next chapters.

Note that the concepts of build-time view and run-time view refer to aspects of the business life cycle (as illustrated in Figure 7). They do not refer to sequential time periods. In an agile, service-based environment, design and modification of the service structure will generally be performed in parallel with the enactment (execution) of this service structure.

3.1 The build-time view

The build-time view of the service-dominant business sandwich concentrates on designing and defining the contents of the four layers of the sandwich (as discussed in Section 2.2). The build-time view of the sandwich is illustrated in Figure 10. The stack 'in the boxes'

indicates a business organization. Outside the 'boxed stack', there are external entities (shown in the right of the figure).

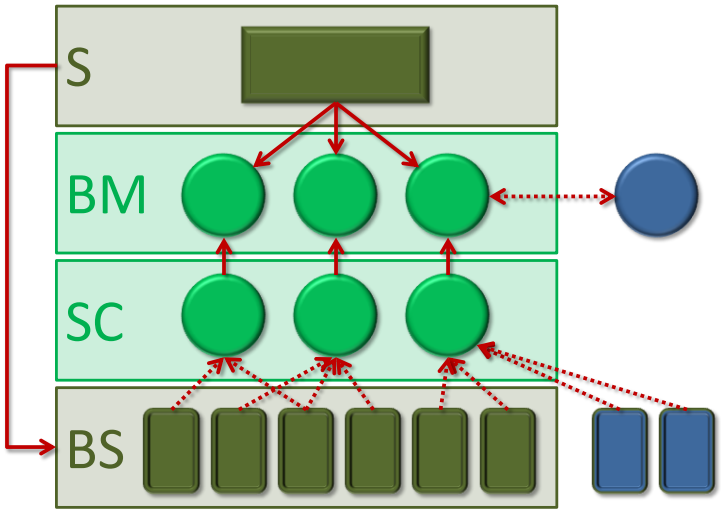

Figure 10: service-dominant business sandwich – build-time view

Below, we discuss each of the four boxed layers in detail. After that, we pay attention to the role of the external entities shown in Figure 10.

3.1.1 Business strategy

The overall starting point in the build-time view is the business strategy definition. This means that in defining a service-dominant business sandwich from scratch, the strategy layer is the point to take off. Once a sandwich exists and we want to modify it, several alternative approaches can be used (as discussed in Section 2.2.3).

Defining a business strategy can be a greenfield exercise, for example when setting up a completely new business organization. More often, however, it is a redefinition or reorientation of an existing strategy.

The strategy defines the identity of a business organization. Therefore, one organization has one strategy. Note that the definition of 'organization' depends on the scoping of the business view: in case of organizations with an explicit hierarchical structure (such as holdings), the 'organization' can be chosen at different levels of hierarchy. In case several strategies are identified for a single organization, we have a problem that requires a solution. A first possible cause of this problem can be that the organization consists of a number of unrelated parts and is therefore not a single coherent organization[8]. In this case, splitting up the organization should be considered. A second possible cause of the problem is that the strategy is not analyzed at the right level and hence business models are mistaken for strategies. In this case, the strategy discussion needs to be elevated to a higher level of abstraction.

We discuss details of the service-dominant business strategy layer in Chapter 4.

[8] This may, for example, be the result of the fact that an organization has been developing along two or more divergent courses, for example keeping one part of the business in the asset-oriented paradigm and moving one part into the service-oriented paradigm.

3.1.2 Business services

As the strategy defines the identity of an organization, it is the basis for the definition of its core capabilities in the business services layer. This is indicated by the arrow in the left-hand side of Figure 10, which is the operationalization of the arrow in the left-hand side of Figure 6.

Capabilities are seen as business services, i.e., encapsulated elemental modules of core business functionality. This means that they have a clear business-level interface[9] to be used (invoked), that they are elements that can be combined in several configurations, and that they represent those competences of a business organization that define its essence.

We discuss details of the business services layer in Chapter 5.

3.1.3 Business models

Below the strategy layer sits the business model layer. This layer contains one or more (typically more) service-dominant business models, which are context-dependent, specialized operationalizations of the strategy. They are context-dependent because their definition is based on a current business context in terms of (among other factors) customer group, geographic situation, availability of business partners, and time period. They are specialized from the strategy as they implement part of the strategy in a more specific way (as shown by the arrows in the figure, which are an operationalization of the arrow in the top right-hand side of Figure 6). They are operationalizations of the strategy as they are more concrete.

A business model may be related to a business model of a (potential) collaborator (as shown in Figure 10 for the right-most business model). This relation is necessary because the desire to use an external service should – obviously – coincide with the desire of an external party to offer this service. If an internal business model is related to the business model of an external collaborator, this implies that one or more business services offered by that external collaborator are used in the internal service compositions.

We discuss details of the business model layer in Chapter 6.

3.1.4 Service compositions

Between the business model and service layers sits the service composition layer. This layer contains one or more (typically more) service compositions, which are aggregations of a set of elementary services from the services layer plus possibly a set of external elementary services (as shown for the right-most composition in the figure).

Each service composition is related to one specific business model: it forms the operationalization of that specific business model. A service composition completely defines the operation of a business model. Hence, if a composition would implement

[9] Note that this business-level interface defines the ways in which a service can be activated (invoked) and which information (invocation parameters) need to be passed to the service. At this level, this is completely independent from any implementation in information technology.

multiple business models, these models would be operationally exactly the same[10]. If a specific business model requires operational flexibility (various options with respect to the way it is executed), this flexibility is created by having flexibility inside its service composition, not by having alternative service compositions.

We discuss details of service compositions in Chapter 7 – here we will also see that there are two main types of compositions, based on the nature of the business model involved.

3.1.5 The network view

Note that Figure 10 shows only a single external party (with one relevant business model and two services) for reasons of clarity. In practice, a business organization exists in a context of multiple (possibly a large number) of other organizations (external parties). Such a context can be called a *service-dominant market*.

The relations between organizations in a service-dominant market at the strategy level are implicit: organizations define their strategies independently without taking explicit links between strategies into account.

At the business model level, the relations between organizations in a market are explicit. A single internal business model of an organization can rely on a (potentially large) number of external business models in the business network. This is illustrated in Figure 11, which is the network-centric view of one of the models in the business model layer of Figure 10. In this figure, we see an internal business model that relies on eight external business models that are developed in the same market by other organizations. Note that the relations are explicit, but not operational: business models must match, but they are not executed (this happens in the service composition layer and the business service layer).

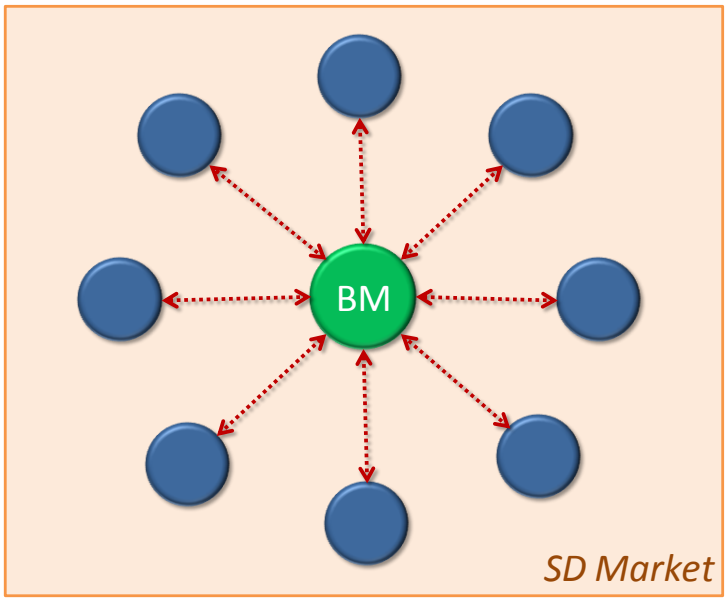

Figure 11: SD market at the business model layer

[10] Obviously, one single business model can be marketed in various ways (e.g., using different commercial labels). This, however, is just a representation issue, not a business engineering issue.

Given a networked business model as illustrated in Figure 11, its service composition uses external business services from the same eight external parties of which the business models are coupled to the internal business model. This is illustrated in Figure 12, which is the network-centric view of one of the compositions in the service composition layer of Figure 10, plus the relevant services at the business service layer. Here we see that the service composition uses four internal services and one or more services from each of the eight external parties the business models of which are linked in the business layer. The links in this figure are explicit and operational (they need to 'executed' to make business work).

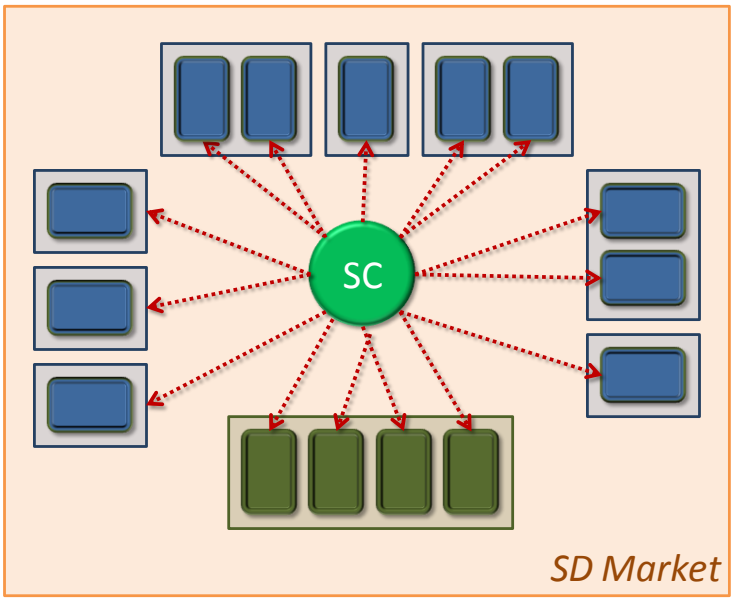

Figure 12: SD market at the service composition and business services layers

3.2 The run-time view

The run-time view of the service-dominant business sandwich concentrates on the execution of service compositions and the generation of management information to the business model and strategy layers – this based on the design made in the build-time view as discussed in the previous section. The run-time view is shown in Figure 13.

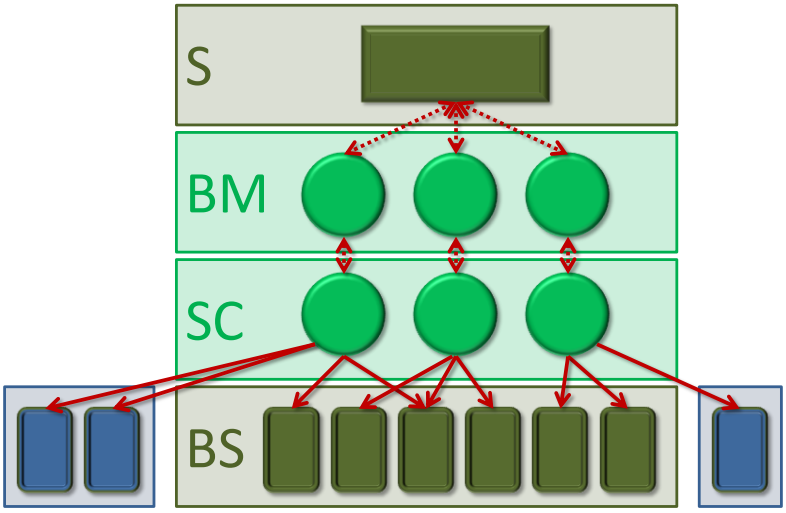

Figure 13: service-dominant business sandwich – run-time view

The core of the run-time view is the service composition layer. Here, the actual business execution is controlled. Service compositions invoke sets of elementary business services, where these services can be internal or external. Where they are external, they are invoked in the context of a specific collaborator – this is indicated by boxes around the external services in the figure. Preferably, this specific collaborator is chosen in a dynamic way during run-time. This means that at build-time, only the characterization of possible collaborators is specified (that constrains the selection at run-time). The characterization is specified in terms of functional attributes (what business services must a collaborator provide) and of non-functional attributes (how are these business service provided in terms of quality of service).

The execution of compositions generates management information, which is fed to the business model layer. This information can be used in two ways. In the first way, it is used as input for the build-time view. In the second way, it is used to feed management dashboards at the business model layer, where parameters can be chosen that influence the execution of the compositions.

The business model layer in its turn generates management information for the strategy layer. This management information is used in the build-time view as input for the evolution of the service-dominant strategy.

3.3 The concept model

In this chapter, we have introduced the sandwich model, which contains a new way to look at the relations between concepts at different levels in agile service-dominant business. In this section, we become more precise about these relations by placing the major concepts into one concept model (a simplified class model in UML terms). This concept model is shown in Figure 14.

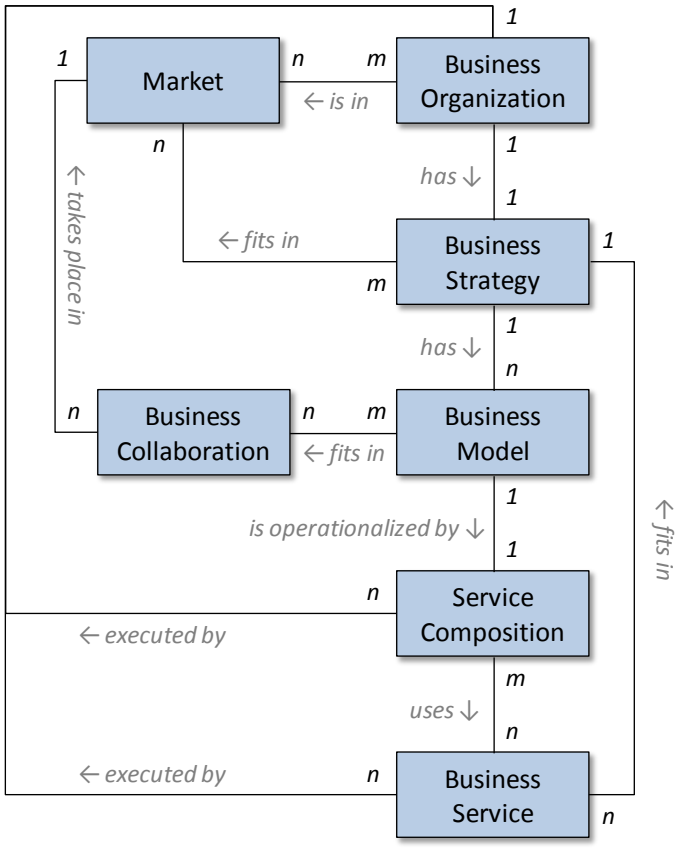

Figure 14: concept model

In the right-hand side of the concept model, we see the four main elements of the sandwich model: business strategy, business model, service composition, and business service. The concept model shows that one business strategy is related to multiple (denoted as *n*) business models[11]. This 1:n relation corresponds with the earlier made observation that a business model can be derived from a business strategy by making a set of choices [Shaf05] - each set of choices results in a different business model. Business models and service compositions have a 1:1 relation: one specific composition realizes one specific business model. A service composition uses multiple business services and a business service can be used in multiple compositions. Hence, compositions and services have an *m:n* relationship. The link between business strategy and business services is explicitly included (this link coincides with the left-hand side arrow in Figure 10). This relation is *1:n*, as one strategy relies on a set of services, and one service is realized by one organization.

In the top and left-hand side of the model, we have included organizational concepts: business organization, market, and business collaboration. A business organization is an autonomous entity that operates in a market according to its business strategy (so there is

[11] Note that this *1:n* relation between business strategy and business models may appear the same as the concept of business model unbundling [Oste10]. This is not precisely the case, however. The relation between strategy and business models as described in this document advocates making a fundamental difference between a single long-term strategy and multiple medium-term business models. The concept of unbundling advocates unraveling hybrid business models into multiple business models at the *same* level where it comes to business design.

a *1:1* relation here). A market is a collection of business organizations that may collaborate[12], as discussed in Section 3.1.5. A market can include many organizations. A market may be completely conceptual (i.e., have no physical appearance), or it may be based on a physical central infrastructure (such as a central business broker). A business collaboration is the combination of a number of business models. Each business model can be part of multiple business collaborations, so this is an *m:n* relationship. Figure 11 shows an example of a business collaboration between 9 parties (the central party plus 8 external parties).

Note that in an agile world, business collaborations are linked to business models, not to business strategies. This coincides with the observations we made about implicit and explicit relation in service-dominant markets (see Section 3.1.5).

In the sequel of this document, we extend the concept model with further concepts to complete describe the domain of agile, service-dominant business.

[12] As such, a service-dominant market can be compared to collaborative networks [Cama10], collaborative business ecosystems and breeding environments for instant virtual enterprises [Meha10].

4 The top sandwich slice: service-dominant business strategy

In the previous chapter, we have constructed the service-dominant business sandwich and shown that it consists of the two 'stable' layers (the bread of the sandwich) and two agile layers (the fillings). In this chapter, we concentrate on details of the top slice of bread: the service dominant strategy layer.

4.1 Service-dominant strategy concept

The top slice of the SD business sandwich is the strategy layer. In this layer, the identity of an organization is defined, which is closely linked to its long-term mission. The strategy evolves over time because the context of the organization evolves. Strategy is coupled to the long-term position of a business organization in a market.

The strategy *does not* change drastically to follow short- to medium-term movements in the market (this is handled by the business model layer).

In this document, we focus on service-dominant business strategies. These SD strategies are in their essence different from non-SD strategies in two major aspects:

1. The strategy is built around a mission that focuses on delivering value-in-use to the customers of the organizations, rather than focuses on assets that underlay the value-in-use. This value-in-use is of an abstract kind – it is operationalized into concrete value-in-use at the business model level (discussed in Section 0).
2. The strategy is explicitly defined in the context of business networks, i.e., collaborations with business partners. In this collaboration, we find co-production relationships. This may mean that the boundaries blur between the roles of producer and consumer of services[13], thus supporting co-creation.

4.2 The service-dominant strategy canvas

The design and specification of service-dominant business strategies is a field that is not yet well-developed in science or practice. We therefore use a design tool that is rooted in three well-established strategy approaches, based on respectively business competences [Hunt04], business resources [Hunt04], and industries or market relationships [Port80, Port85]. From these approaches, we take the elements relevant to service-orientation and

[13] The boundaries between the roles blur, but not the boundaries between the organizations, which are laid down in (dynamically established) contracts. The more dynamic the collaboration, the more explicit these contracts need to be, as there is less and less of an implicit, relational business structure. As dynamism increases, the number of contracts will increase. A more explicit nature of contracts implies more complex contracts. Both of these aspects increase the need for automated support for contract handling (both for establishing contracts, enacting them, and monitoring the enactment).

blend them into a new approach that is tailored to the SD world, yet understandable to the non-SD expert. This approach is illustrated in Figure 15.

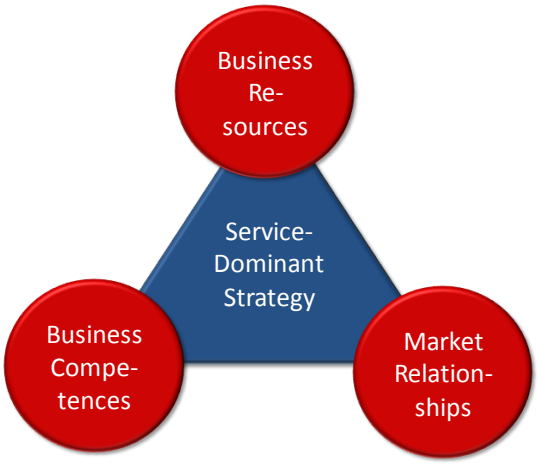

Figure 15: traditional strategy approaches blended into an SD business strategy

The approach has been materialized in a service-dominant business strategy canvas [Lüft12a, Lüft12b]. This canvas is shown in Figure 16. The canvas first organizes the essential elements of an SD strategy according to three main categories coinciding with the three traditional strategy approaches shown in Figure 15 – these are the three columns of the strategy canvas.

Each of the three main categories is divided into two elements sets, resulting in six categories (the rounded rectangles in Figure 16). The *exogenous relationships* category contains elements related to business relationships that have an origin external to the organization. The relationships in the *endogenous relationships* category have an origin internal to the organization. The *value* category contains elements directly related to the value-in-use. The *collaboration* category contains elements that are related to the collaboration required to produce the value-in-use. In the *actors* category, we find three types of business actors. Finally, in the *infrastructures* category, we find infrastructural elements required to make a service-dominant organization work.

Each category contains two or three elements that characterize an SD business strategy. The total canvas contains the fifteen elements shown with text labels and pictograms in Figure 16. The complete background of the choice of elements and the elements themselves are explained in detail in [Lüft12a].

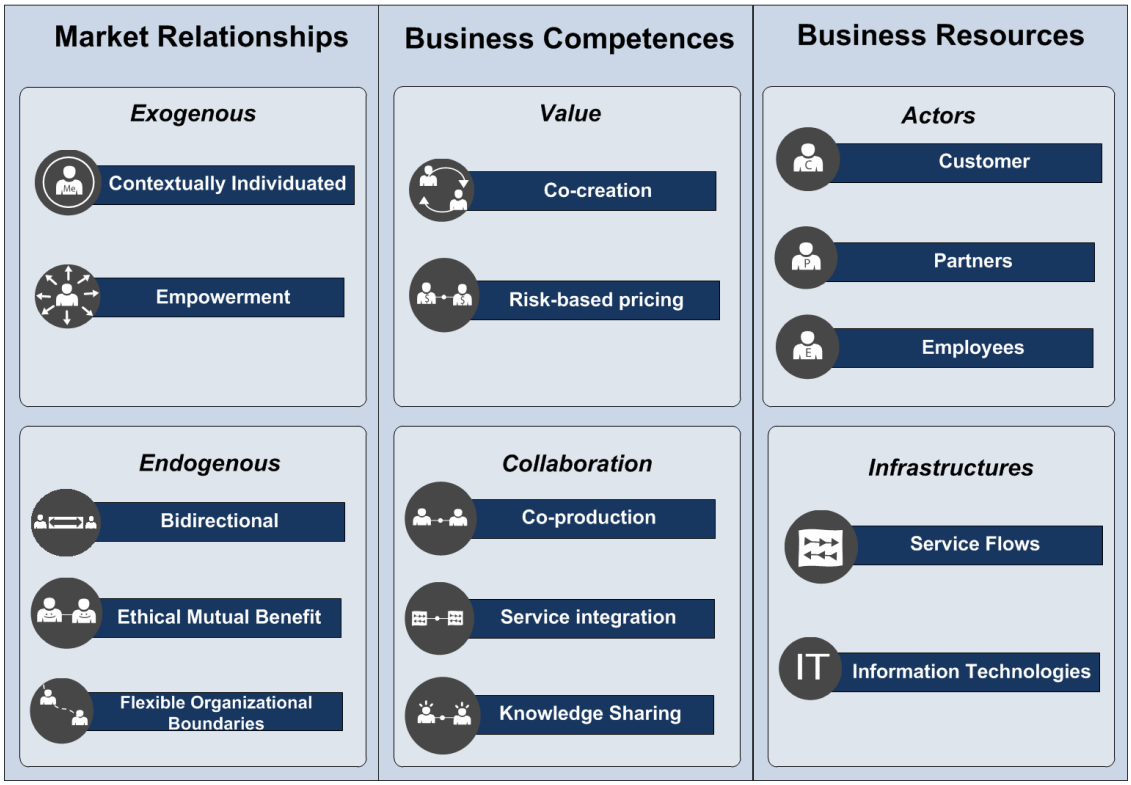

Figure 16: SD business strategy canvas (adapted from [Lüft12a])

An SD strategy is defined by filling in the canvas for a specific organization, using both domain experts and a canvas expert. Typically, this is done in an iterative way as defining an SD business strategy is a complex process.

We explain the canvas by applying it to the TraXP case below.

4.3 TraXP

Given the radical nature of its vision and mission (as introduced in Section 1.6), TraXP performs a detailed strategy analysis using the service-dominant business strategy canvas, resulting in a detailed strategy specification. A high-level summary is included below.

- Business Competences
    - Value: The value offered by TraXP is a seamless and complete travel experience that possibly spans both private and professional life, takes place in a context that can have personal, corporate and social elements, and that is supported by first-of-class service providers and up-to-date, first-hand information. This implies that co-creation has to take place between TraXP, its primary service providers, and its traveler customers. TraXP offers a transparent pricing scheme to its customers, based on the 'volume and intensity of travel experience'. Details of this pricing scheme are elaborated per business model.
    - Collaboration: As TraXP will implement mostly its integrating services, it will rely heavily on co-production with its primary service providers for

many of the ‘hard’ travel services. Tight service integration is important to offer a transparent and real-time experience to customers. Knowledge sharing is required with its primary service providers to allow detailed information-based integration of services.

- Market Relationships
    - Exogenous: TraXP customers (both individuals and corporations) will require a completely customized travel experience, based on elaborate profiles. Customization includes state-of-the-art pre- and post-trip information provisioning. Customers require empowerment as to be able to always overrule any travel decision made by TraXP.
    - Endogenous: TraXP will set up bidirectional relations with its primary service providers to operate on a peer-to-peer basis. This guarantees more collaborative stability than strict outsourcing relations. TraXP uses an open collaboration network, enabling new partners to join. TraXP promotes a transparent scheme in the use of travel information of customers, such that open mutual benefit between travelers and TraXP is clear.
- Business Resources
    - Actors: Customers are an important resource in the TraXP strategy, as it relies on a traveler community. TraXP sees the primary service providers as important resources. Secondary service providers should be interchangeable in a flexible way. Business development employees are seen as critical resources, as the travel market is highly fluid in nature. As trust from customers is mission-critical as well, customer service employees are highly listed as well.
    - Infrastructures: TraXP sees two kinds of service flows in its strategy: the kind supporting simple, one-shot travel organization (such as a package holiday, which is buy-and-go) and the kind requiring complex, real-time process management (such as multi-person, multi-location business trips, which need execution in well-synchronized phases). TraXP’s information systems will be mostly Internet-based and service-oriented to allow flexible communication with partners and customers. Local information systems are used to analyze and predict traveler behavior. To cater for elasticity in service demand, its IT infrastructures will be as much as possible cloud-based.

The essentials of the above are briefly summarized in TraXP’s strategy canvas, as shown in Figure 17. Here, we see the six categories of elements discussed above. Within each, categories, we have the elements defined in the canvas of Figure 16.

Market Relationships	*Business Competences*	*Business Resources*
Exogenous Profile-based, corporate & individual, state-of-the-art experience customization Empowerment of customer overrule of travel decisions	*Value* Co-creation with 1st SPs & customers of multi-context, 1st class, real-time travel experience Volume & intensity pricing scheme	*Actors* Customers as info-generating community Primary SPs as partners, secondary SPs as providers Emphasis on business developers and customer service
Endogenous Bidirectional P2P Relations with primary SPs Open collaboration network Mutual information benefit with and between customers	*Collaboration* Tightly Integrated Coproduction with primary SPs Knowledge Sharing with primary SPs	*Infrastructures* Dual logical infrastructure for one-shot and process-based travel support Service, Web & cloud-based infrastructure

Figure 17: SD strategy canvas of TraXP

5 The bottom sandwich slice: business services

The bottom slice of the SD business sandwich contains the elementary business services. They define the capabilities of an organization – like the business strategy, they are of a stable kind. Below, we first discuss the anatomy of a business service, answering the question what its structure is. Next, we pay attention to important characteristics of services. Then, we turn to our TraXP running example to practically illustrate the business service concept.

5.1 The anatomy of a business service

In this section, we discuss the anatomy, i.e. the internal structure, of a business service. We need to understand this anatomy to see how the service-dominant business view connects to the physical (asset-oriented) business view.

Each business service encapsulates an elementary business capability of the organization and makes this capability accessible through a well-defined, business-level interface (see Figure 18).

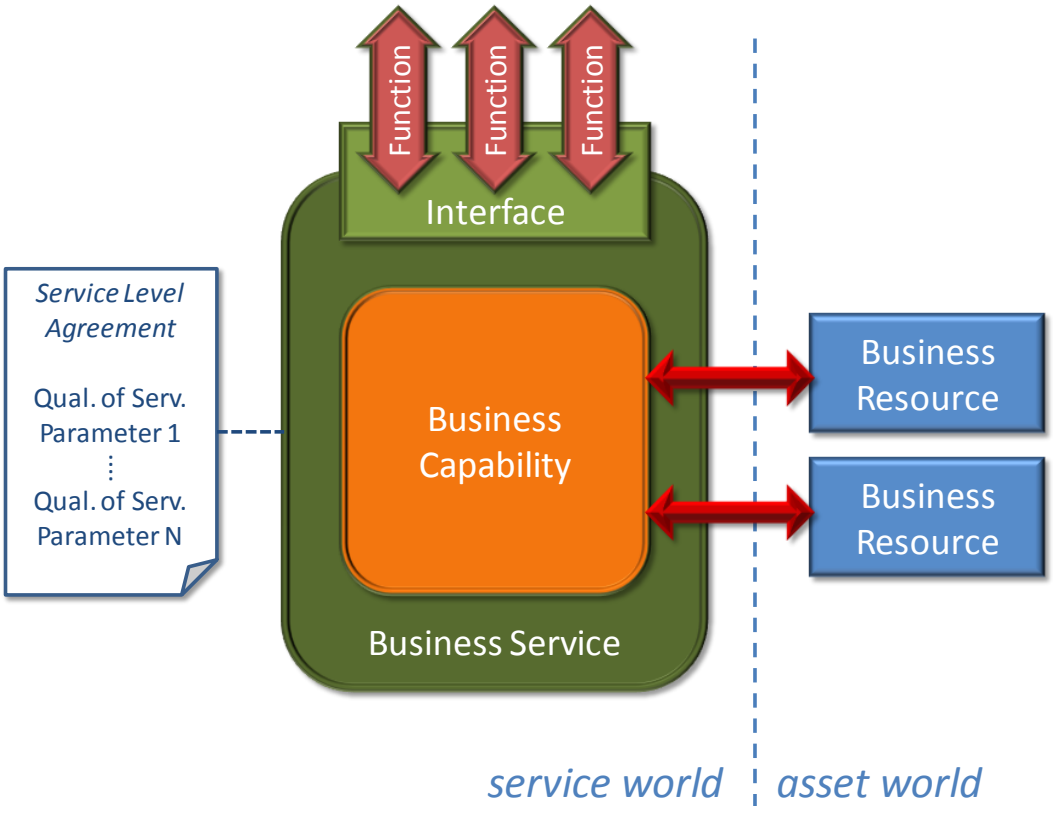

Figure 18: business service anatomy

The interface to a business service contains a number of service interface functions (as shown in the figure), each of which provides business-level access to a specific aspect of the service. Important classes of interface functions are:

- Query service status (availability, quality level, current cost level)
- Reserve service capacity (if the service encapsulates scarce resources)
- Invoke service (consume service capacity)
- Cancel/rollback service while in execution (if applicable)
- Acknowledge successful consumption of service (if applicable)

The functional specification of the service should preferably be coupled to a service level agreement (SLA) that specifies the non-functional behavior of the functions of the service in terms of quality of service (QoS) parameters (this is shown in the left-hand side of Figure 18). Important QoS parameters are:

- Average response time of a function (as a time period)
- Availability of a function (as a percentage of scheduled business up-time)
- Cost of a function (as a monetary figure per function call)

A business capability may rely on physical business resources (as illustrated in Figure 18) or may be completely digital in nature. Physical resources are located in the asset world, whereas the service is located in the service world.

A function of a business service is associated with a cost per use (a QoS parameter as indicated above). This cost is related to the use of resources encapsulated by the service and to the use of the automated platform for the execution of the function. Services can also produce a financial benefit, for example by including invoicing functionality. Profitability in a market context does not play a role at the individual service level, but at the level of service compositions, as these implement business models that are offered to a market.

5.2 Business resources

Business resources used in business services come in two important kinds: material resources and human resources.

Material resources are physical business assets, such as capital goods. In a traditional, non-SD, business scenario, we would trade the ownership of these – in an SD scenario, we make their value-in-use available through business services (as we have discussed in Section 1.1).

Human resources contain the capabilities (both physical and mental) of the employees of an organization that work in the primary process of that organization (not those in the secondary processes).

We can add business resource concepts to the concept model that we have introduced in Section 3.3 to show how they interrelate with the other concepts. This results in the model shown in Figure 19. Note that business network concepts (as included in Figure 14) have been omitted in Figure 19 to not make the model too complex.

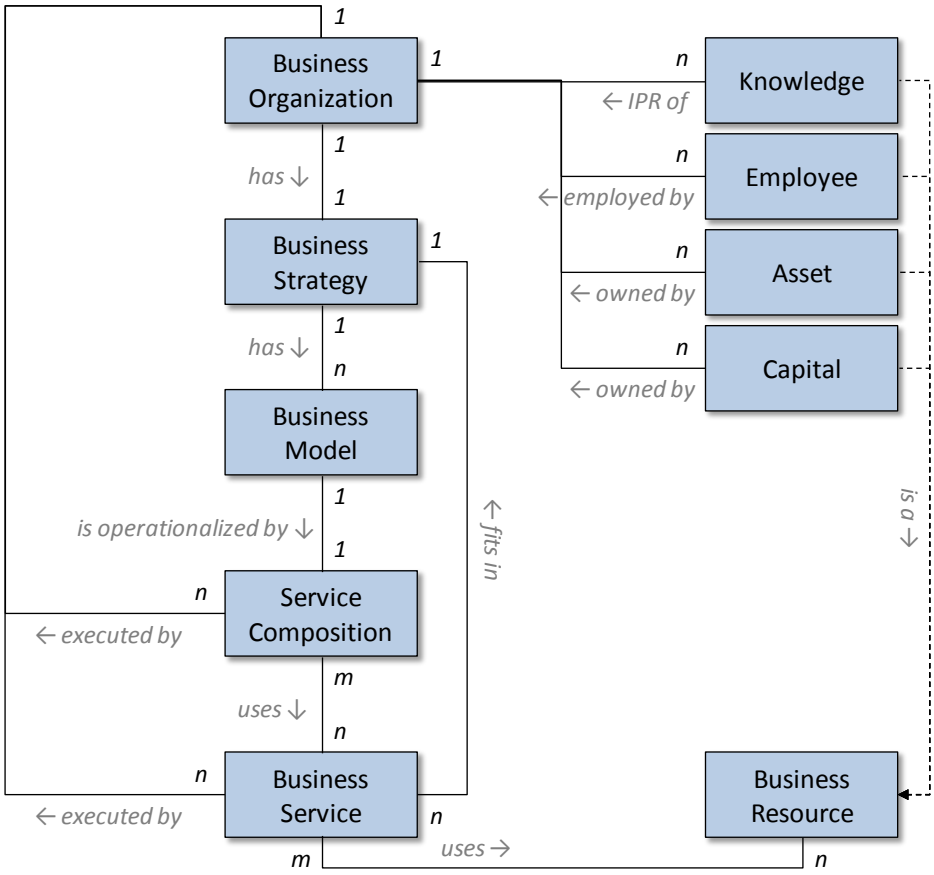

Figure 19: concept model with resources (dotted lines denote specializations)

5.3 Internal versus external business services

It is important to distinguish between internal and external services. This distinction actually strongly influences the composition of business networks (as discussed in Section 3.1), as it determines what an organization can do itself and what needs to be outsourced.

Internal services are provided by the focal organization and implement the core capabilities of this organization. These core capabilities (and hence the services that implement them) should be differentiating in a market. Often, this differentiation is based on the resources used in the services (as discussed above and shown in Figure 18) – though this is certainly not always the case: differentiation may also be based on knowledge, which is of a non-physical character.

External services are provided by partners and complement the core competences of the focal organization by implementing common-of-the-shelve capabilities in specific service compositions. Note that these external services should be core capabilities to the organizations that offer them to obtain the best value in service outsourcing.

As discussed in Section 3.1, the set of internal services is determined by the business strategy. Nevertheless, it can be worthwhile to classify services in the process of determining whether they will be internal. This can both be helpful in the case of testing

the match between an existing strategy and an existing set of internal services and in the process of actually delineating a strategy in a business transformation process.

Classification of business services can be performed using the strategic partitioning tool shown in Figure 20[14]. This tool uses two dimensions. The first dimension distinguishes services that are differentiating in a market from those that are not (commodity services). Commodity services do typically not contribute to the unique characteristics of a value-in-use proposition, so should usually be external. The second dimension distinguishes services that are mission critical from those that are not. Mission critical services must always be operated with guaranteed quality of service. This implies that commodity services that are not available with the right SLA may be internal (until they become available externally, possibly even by explicitly externalizing them from an organization). Differentiating services for which no internal support yet exists may be external (until they become internal, possibly even by explicitly merging the service provider into an organization).

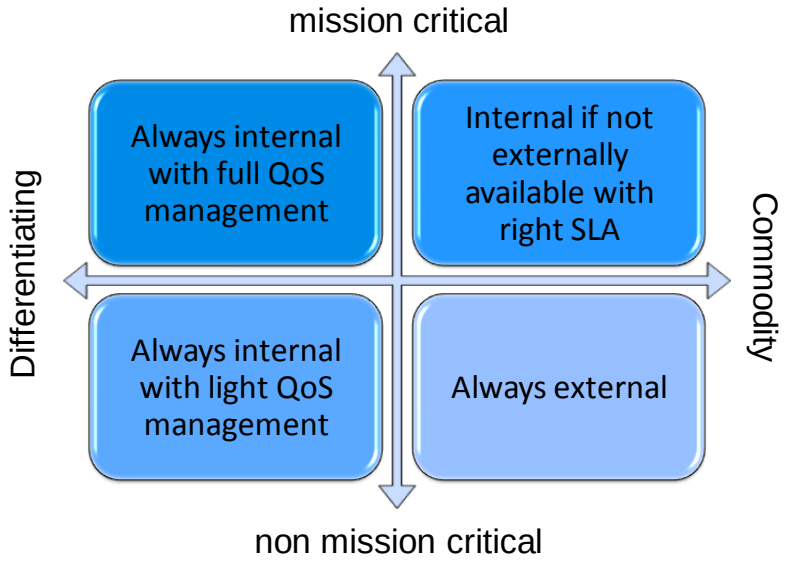

Figure 20: strategic partitioning tool for business service classification

5.4 Granularity of business services

The granularity of business services is chosen such, that flexible combination (in the service composition layer of the sandwich) is possible. Note that too 'large' services will result in too few composition possibilities, whereas too 'small' services will result in overly complex compositions. The right granularity of services is heavily domain-dependent.

[14] The matrix in this tool is based on a similar matrix using the dimensions 'core vs. context' and 'mission critical vs. non-mission critical' [Moor05].

There are several rule-of-thumb principles that can be used to determine the granularity of services:

- A service should be associated with a business process step observable by a business partner (including the customer, obviously), preferably directly related to the offered value-in-use – in other words: a partner must be able to clearly relate the service to (the life cycle of) the business offering. If this is a problem, the granularity may be too small.
- A service should have a clear actor (service provider) and a clear beneficiary (service consumer): it should be clear who is doing something for whom. If there are multiple, the granularity may be too large.
- A service should be associated with a clearly delineated period in time: it should have a clear starting point and ending point. If this is a problem, the granularity may be too large (for instance, because an iteration of a service is perceived as a service).

Too small services need to be combined into larger services – this is related to hierarchical service realization as discussed in Chapter 13. Too large services may in fact be (part of) service compositions, so belong in the Service Composition layer of the sandwich model.

Table 2 contains an overview of criteria for the determination of 'right' business services. The criteria are grouped into interrogative-based classes. The right-most column of the table indicates actions that can be performed if a candidate service does not meet a specific criterion.

Class	Name	Criterion	If fails
Why	Right context	Does the service fit in the context of the defined business strategy?	Redesign service
	Right goal	Does the service transform the state of the customer perception of the value-in-use?	Delete as business service, consider as building block service[15]
What	Right size	Is the service easily combinable in multiple service compositions?	Too large: split up service Too small: combine services
	Right scope	Isn't there any functional overlap with existing service(s)?	Re-scope service(s)
Who	Right actor	Is there a clear single actor (role) performing the service?	Split up service per actor
	Right beneficiary	Is there a single beneficiary (role) for whom the service is performed?	Split up service per beneficiary
When	Right start	Is there a clear starting point in time for the execution of the service?	Scope service down to delimited period
	Right end	Is there a clear ending point in time for the execution of the service?	Scope service down to delimited period

Table 2: criteria for right service determination

Note that it may appear as a good idea to design services on the basis of individual resources or assets (i.e., a *With What* class in the structure of Table 2). This conflicts, however, with the fact that service-dominant business is centered at value-in-use, not around assets.

Stability (both syntactically and semantically) of services and their interfaces is an aspect to be considered in the identification of services.

5.5 TraXP

Based on its service-dominant strategy, TraXP has determined a set of business services. These business services implement the core competences of TraXP (and are hence of a stable nature that evolves over time) and are complemented by external services that are insourced in a more dynamic way (we will see that when discussing service compositions).

[15] See Chapter 13 on hierarchical composition of services.

In designing their set of essential services, TraXP has used their strategy as an explicit starting point and later reference point for consistency checking. As such, they have elaborated a mapping from the elements of their strategy to the business services that support these elements. Figure 21 shows a small part of this mapping in a graphical way.

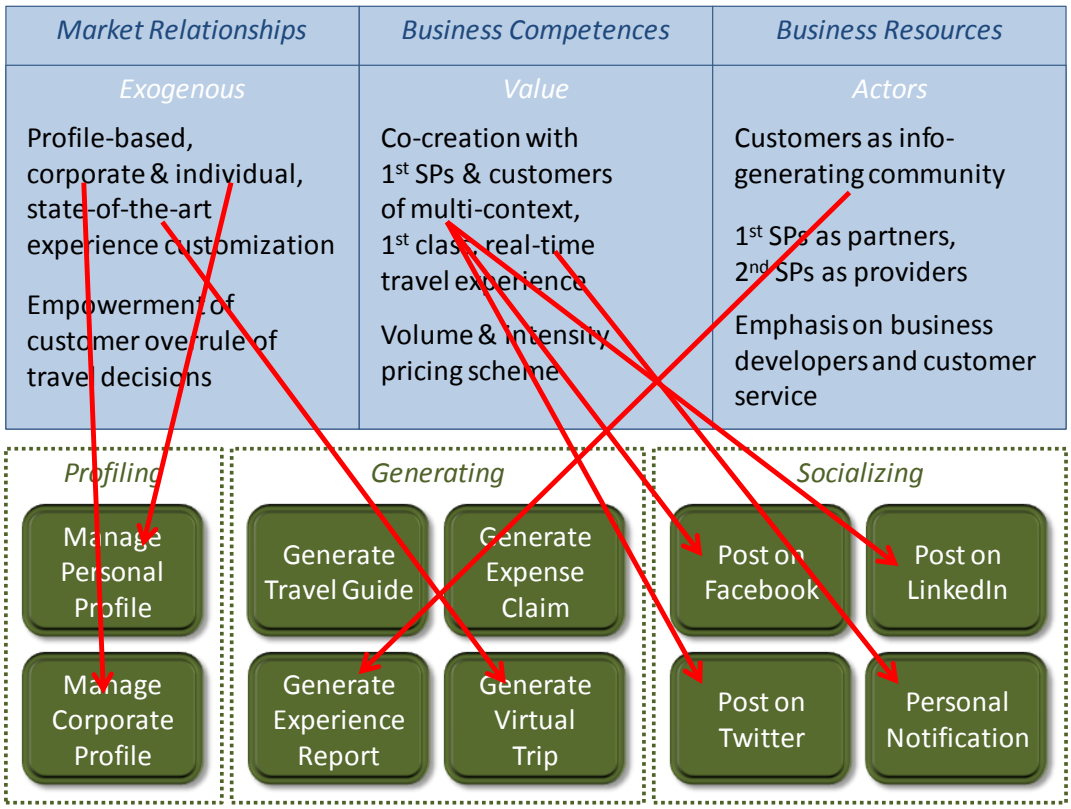

Figure 21: partial mapping of strategy to services for TraXP

The elementary business services have been organized in subsets, representing the major areas of the business capabilities of TraXP – we call these areas the business service domains. A heavily simplified version of the complete set of the TraXP business services is shown in Figure 22.

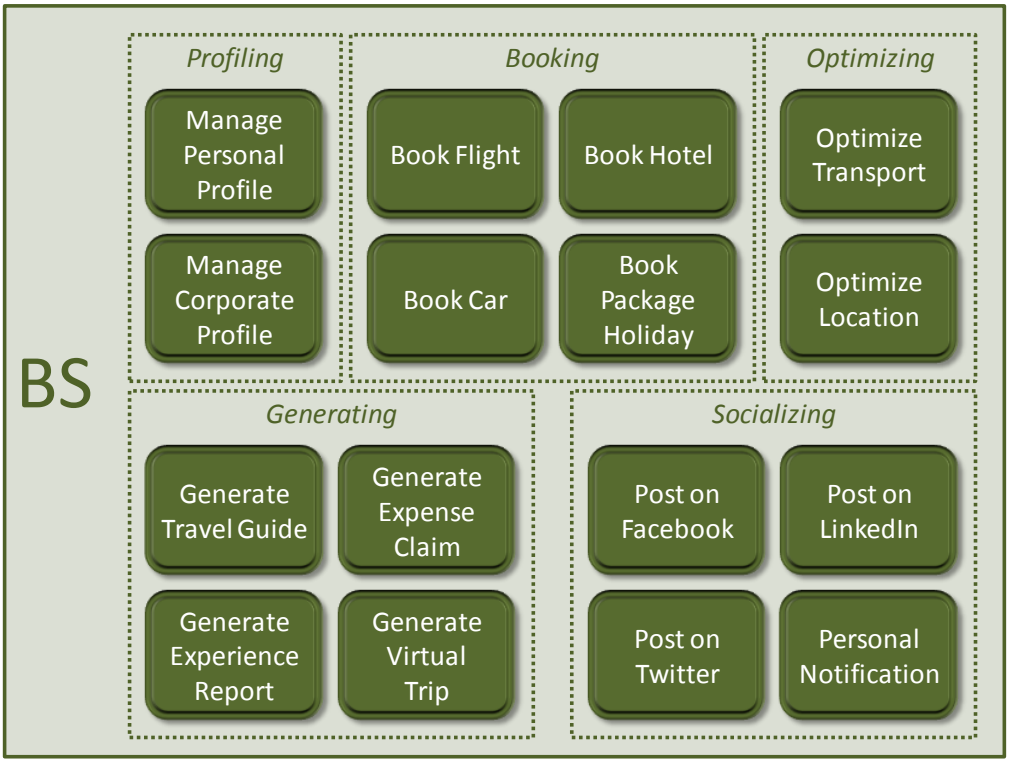

Figure 22: TraXP business services in their business service domains

We briefly discuss the subsets of business services to give the reader an impression of their functionality:

Profiling Services: Enable the creation and management of profiles that drive automated decision making in all the other services. Profiles exist on the personal and corporate profile. Corporate profiles can include complete business travel regulations and preferences which are automatically applied to individual trips.

Booking Services: Enable the interface to the actual booking of travel elements, such as air and rail transport, hotel accommodation, rental cars and complete package holidays. These services internally rely on booking services of external integration partners such as Expedia and Booking, but can also make direct use of travel industry services such as Amadeus.

Optimizing Services: Enable the optimization of travel plans in two ways. Transport optimization will determine the optimal way to travel between two set points, taking settings in relevant profiles into consideration. Location optimization will determine the optimal physical location to meet for a set of travelers that needs to convene, taking settings in relevant profiles into consideration. Both types of optimization services are 'self-learning' as they take historical experiences of TraXP users into account.

Generating Services: Enable the generation of various types of pre-trip and post-trip reports, as well as generation of virtual trips for pre-trip experience or social sharing. Pre-trip reports can be complete (digital) travel guides with integrated e-tickets, which provide a one-stop solution to both business and private travelers. Virtual

trips can be generated as static trips (such as photo presentations) or dynamic trips (such as tours in virtual worlds like Second Life).

Socializing Services: Enable posting travel plans and real-time travel updates on social media (pull socializing) and enable direct notification of selected relations by automated phone or text message (push socializing). These services serve a higher-level of real-time travelling and meeting behavior, which is especially interesting for the business traveler. These services rely on external services offered by social media and telecommunication organizations, and on external services offered by transport operators (such as Lufthansa's MySkyStatus).

Note that the TraXP service catalog includes business services that TraXP considers own core capabilities and business services that rely on core capabilities of external partners in its business network. Business services that rely on external parties can be considered placeholders in the TraXP catalog for external services. This difference between internally and externally implemented services is illustrated for TraXP in Figure 23.

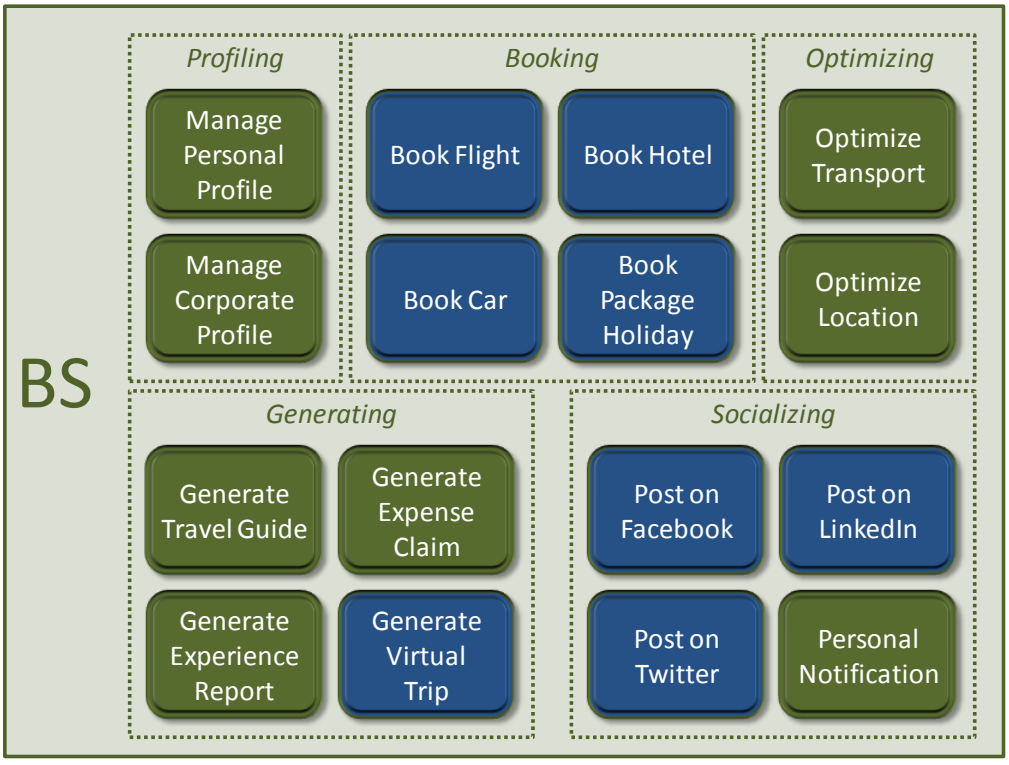

Figure 23: internal (green) and external (blue) business services for TraXP

5.6 Creating a business services catalog from a business strategy

As discussed before in this document, a set of elementary business services is deduced from the service-dominant business strategy – this is the strategic design shown in Figure 6. In making this deduction, a number of aspects have to be taken into account:

- What is the functionality that must be provided by the set of business services? In other words: how can the mission-like character of a strategy be mapped to operational business capabilities?
- What is right granularity of the business services? In other words: 'how much functionality' should be covered by one service? We have discussed this for individual services in Section 5.4.
- Which services are considered essential to keep internal, and which can be assumed to be outsourced to collaboration partners? This issue we have discussed in detail in Section 5.3.
- How can we organize services to make them a manageable set? Here, qualities of the set like completeness, non-overlap, and orthogonality play a role. This issue we address further below.

There is no general recipe available for the mapping of a business strategy to a set of business services, so it requires some old-fashioned craftsmanship by a team of business architects (who understand structure) and business domain experts (who understand content). Domain-specific function reference models do come in handy, though: these provide an overview of functionality to be covered in a domain and hence can be used as a starting point or checklist. In [Gref10], an example is included of deducing high-level e-business functions from the well-known supply chain model designed by Porter [Port85].

To address the aspect of manageability mentioned above, the set of business services is documented in a business service catalog. In this catalog, services are grouped with respect to their business service domains (as illustrated in Figure 22 for TraXP). Note that a business service domain is a cluster of related business capabilities, not a business model. Each service is described in a standardized way, listing all its important characteristics. Note that in practice, a catalog can be large – for a large company, it can easily consist of hundreds of business services.

To facilitate the creation of a business service catalog, templates can be used. In Table 3, a template is shown for the specification of a business service domain (set of related business services).

Business Domain / Service Cluster	*Enter domain name here:*
Domain Description	*Enter brief domain description here:*
Person Responsible for Definition of Domain	*Enter name and contact details here:*

Table 3: catalog template for business service domain description

In Table 4, a template is shown for the specification of an individual business service. A proper classification of a service is essential – for this, we use the characteristics that are discussed in Section 5.3. Each service function is described in detail using the lower part of the template: each row is a service (so the number of rows varies per service).

Service name				
General Service Description				
Service cluster service belongs to				
Service functionality in terms of value-in-use				
Business resources used by service				
Service Classification				
Mission-critical or Non-mission-critical		Remarks		
Commodity or Differentiation		Remarks		
Internal or External		Remarks		
Service Functions				
Function	Functionality	Input	Output	SLA

Table 4: catalog template for business service description

5.7 TraXP

In Table 5, we see an example business service domain description from the TraXP catalog, following the template in Table 3. In Table 6, we find an example business service description from the TraXP catalog, following the template in Table 4.

Business Domain / Service Cluster	Generating Business Services
Domain Description	Cluster of business services that generate information (in the broad sense) for TraXP customer
Person Responsible for Definition of Domain	S. Omeone Office: TraXP HQ H.17 Skype: s.omeone

Table 5: TraXP business service domain description

Service name	Generate Virtual Trip		
General Service Description			
Service cluster service belongs to	Generating Business Services		
Service functionality in terms of value-in-use	Generates a virtual trip specification that can be enacted in Second Life. Allows customer to have a virtual, pre-real-life trip experience.		
Business resources used by service	Access to HP compute cluster.		
Service Classification			
Mission-critical or Non-mission-critical	NMC	Remarks	If becoming popular, may become MC for /F BM
Commodity or Differentiation	DIFF	Remarks	No competitor in the market has this
Internal or External	INT	Remarks	May be outsourced to Linden Labs later

Service Functions				
Function	Functionality	Input	Output	SLA
Create_VT	Creates VT specification	Travel plan	VT specification	Execution < 30 sec, Availability 90%
Check_TP	Checks travel plan for conformance	Travel plan	Yes/No	Execution < 10 sec, Availability 99%
Check_Status	Checks current online status of service	None	0 if online, time to online in secs if offline	Execution < 1 sec, Availability 99.9%

Table 6: TraXP business service description

6 The upper sandwich filling: service-dominant business models

In this chapter, we address the top half of the filling of the service-dominant business sandwich: the business model layer. The business model layer defines the *what* of external service offerings, providing the business side of business agility in the more stable context of the strategy and business service layers.

First, we address the importance of business model design. Then, we present a design tool that suits the service-dominant business context.

6.1 Service-dominant business model design approach

Business model design is currently getting proper attention, for instance using tools like the Business Model Canvas [Oste10]. Existing approaches, however, are typically not focusing on service-dominant business. This is reflected in the fact that services and service compositions are not central concepts in the approaches, and the fact that these approaches are typically organization-centric, not network-centric.

Therefore, we propose a business model design approach [Lüft12c] that has a service-dominant starting point and is conceptually tied to the service-dominant strategy canvas. Unlike more traditional approaches (like the Business Model Canvas mentioned above), services and business networks are native elements in our business model design approach.

The pillars of our business modeling approach are shown in Figure 24. We see the service concept as the first horizontal layer. The service concept is the delivery operationalization of the value-in-use concept as discussed in Section 1.1. Services need to be managed in a business model, so this is the second horizontal layer in Figure 24. This incurs costs for involved parties, but also yields benefits - this is the third layer. The actors (the vertical pillar) compose the business network. Note that the customer is one of the actors, as a customer can be a co-creator in the SD business paradigm. The actors interact through the three horizontal elements.

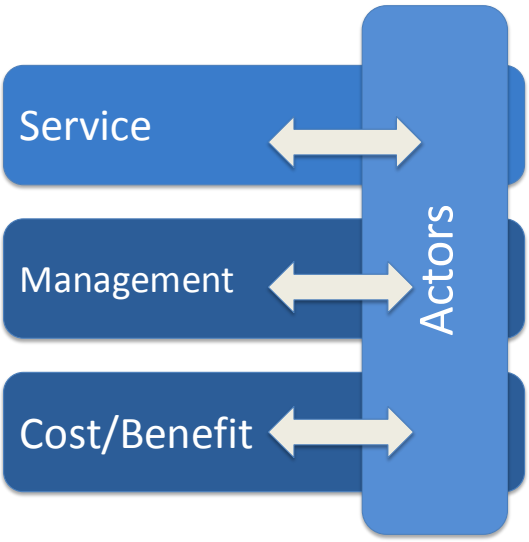

Figure 24: SD business model pillars

6.2 Service-dominant business model radar

The service dominant business model radar (SDBM/R) is the central tool in the design of service-dominant business models [Lüft12c]. An SDBM/R template is shown in Figure 25 (in an updated version of the one presented in [Lüft12c]).

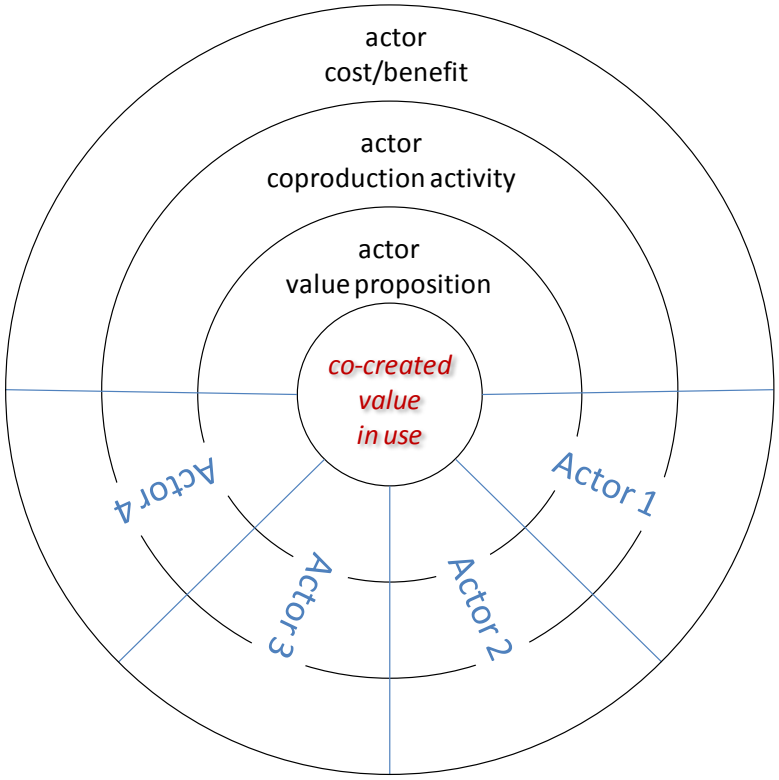

Figure 25: service-dominant business model radar

In the radar, we see the three horizontal pillars of Figure 24 as three concentric rings and a core. The concept of service is central, presented as the core notion of a co-created value-in-use (the core of the radar model), which is composed of service-based value propositions of the actors participating in a business model (in the inner concentric ring). The service management pillar is represented by coproduction activities of the actors (in the middle concentric ring). The cost/benefit pillar is represented by the costs and benefits of the actors (in the outer concentric ring).

The actors in a business model are represented by radial regions of the radar model. The figure shows four actors, but the approach accommodates an arbitrary number, hence suiting the network-centric characteristic of service-dominant business (see Section 3.1.5). This means the model is explicitly designed for the representation of multi-sided business models [McKin12]. As stated before, the customer is one of the actors in a service-dominant business model.

6.3 The role of value in use

A business model defines a concrete value-in-use that is needed by a customer group (this is the external business offering). A customer group is a set of parties operating in a market, where the set is defined by specific characteristics of the parties. The concrete value-in-use is coupled to a customer group through the customer profile. This is shown in the concept model in Figure 26 (based on the core concepts from Figure 14).

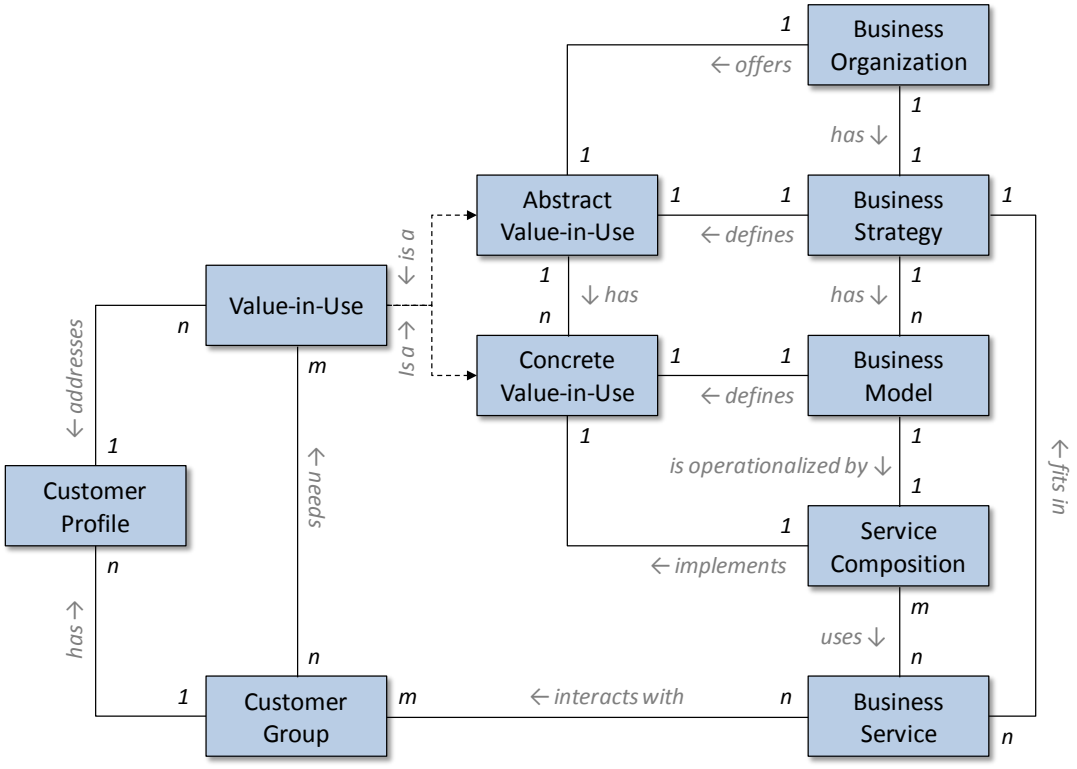

Figure 26: concept model with value-in-use

The concrete value-in-use related to a business model has to be consistent with the abstract value-in-use that is related to the business strategy – quite like a business model

has to be consistent with a business strategy. Both relations are indicated by the *has* relation between the pairs of concepts in Figure 26.

Going from an abstract value-in-use to a concrete value-in-use requires making a set of concrete choices, among which the choice for the intended customer group. These choices are related to the choices made to get from business strategy to business model, as we have discussed before (see Section 3.3).

6.4 Designing concrete business models

The design process to arrive at a concrete business model is strongly related to the specification of concrete value-in-use. The actual steps in the design process are closely related to the choice of sequencing as discussed in Section 2.2.3.

A typical sequence is going from business strategy to business model, i.e., using the strategy-based, top-down sequence from Table 1. Thinking from a value-in-use-centric perspective, this means explicitly scoping the abstract value-in-use associated with the business strategy towards a specific customer profile (see Figure 26). This scoping can be performed based on different dimensions:

- Geographical location of the customer group
- Socio-economical status of the customer group
- Profession of the customer group
- Etcetera

6.5 TraXP

TraXP performs a business model design using the strategy-based, top-down sequence from Table 1. From their abstract value-in-use "seamless travel experience" (STE) they derive four concrete cases of value-in-use by focusing on a specific customer group. In other words: they explicitly choose a customer group per business model and base further decisions on the specifics of concrete values-in-use on the characteristics (needs) of these customer groups. An overview of these choices is shown in Table 7.

Abstract value-in-use	Customer group	Concrete value-in-use	Label
Seamless Travel Experience (STE)	'Poor, young' budget traveler	STE without orchestration but high social exposure	Freebee /F
	'Wealthy, mature' individual traveler	STE with partial orchestration	Individual /I
	Standard corporate traveler	STE with full orchestration	Corporate /C
	Executive traveler	STE with location optimization and full orchestration	Executive /X

Table 7: TraXP cases of concrete value-in-use

The four business models related to the four concrete values-in-use can be informally and briefly described as follows:

- TraXP/F: the freebee business model, aiming at the student-like category budget traveler; traveler gets a free subscription; TraXP gets a small kickback fee from partners for bookings at them, but more importantly so gets exposure in a growth market; targets hit-and-go travel, including free partners like CouchSurfing.org; does not include provider-based travel orchestration; does include tight coupling to social media to enable high social exposure of travelling (follows from user group profile).
- TraXP/I: the individual business model, aiming at the individual business traveler and high-end leisure traveler; TraXP charges by consumption in a (volume x intensity) model measured in XPs (TraXP's unit of experience-delivering service), based on a credit system; includes partial provider-based travel orchestration (i.e., advanced elements of the orchestration remain traveler's responsibility).
- TraXP/C: the corporate business model, aiming at organizations with a substantial amount of business travelers that travel within the corporate regime; TraXP charges on a per-contract basis, both with customer and travel partners set in the corporate regime (two-sided earnings model); includes full provider-based travel orchestration.
- TraXP/X: the executive business model, aiming at the high-end executive business traveler; TraXP charges a fixed monthly fee; includes full provider-based travel orchestration; also includes meeting location optimization as an exclusive (platinum-level) service.

Each of the four business models is modeled using the business model radar introduced in Section 6.2. Each business model specifies a different concrete value-in-use, shown as ViU/F (freebee), ViU/I (individual), ViU/C (corporate) and ViU/X (executive). The four radar diagrams are shown highly simplified stylized in Figure 27. The figure illustrates that usually a more complex business model implies more partners (shown in the figure as *Actor 2* through *Actor 6*) in the business network.

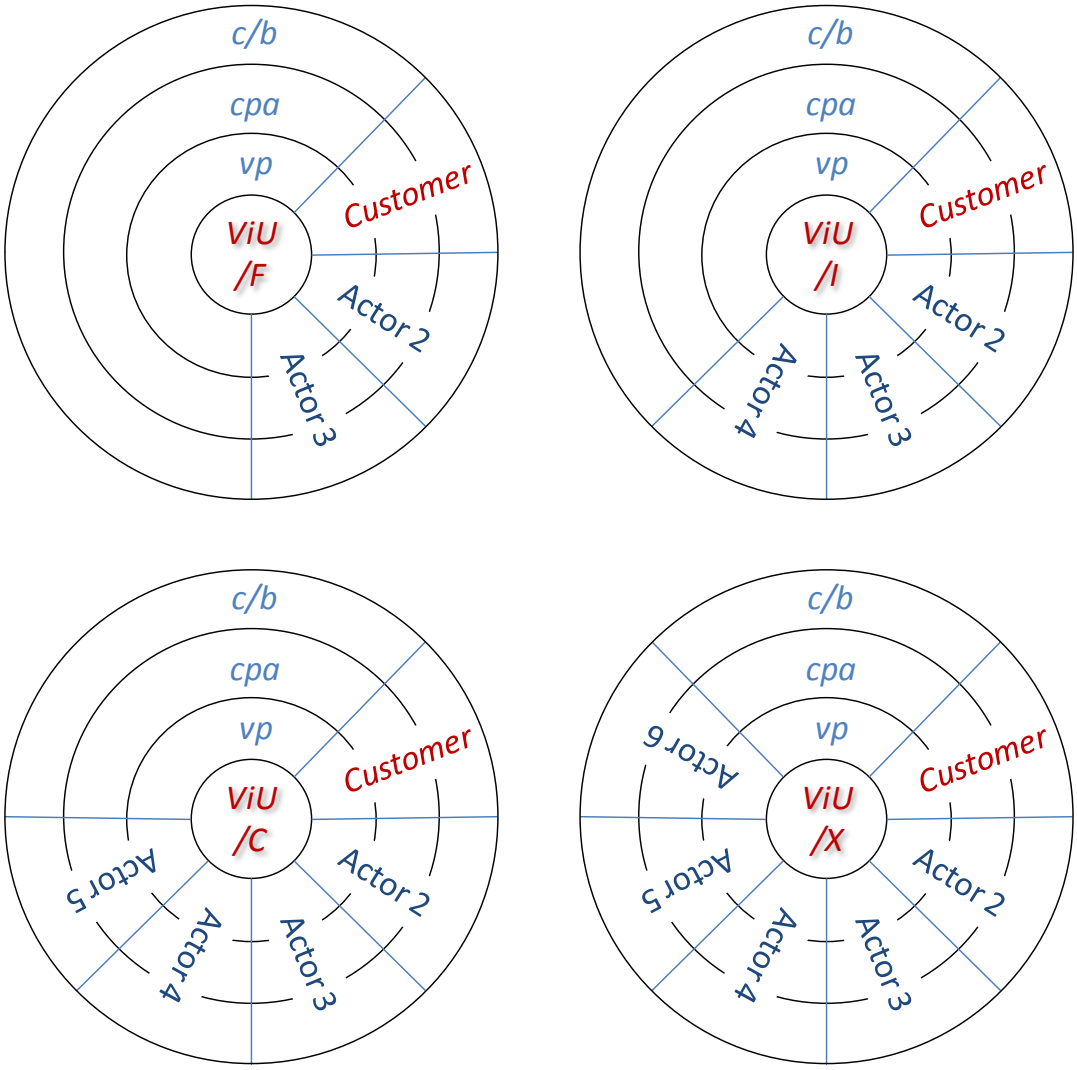

Figure 27: four TraXP business models in business model radar design

A simplified version of the business model radar for the TraXP/X business model is shown in Figure 28. For the other three business models discussed above, similar radar diagrams can be constructed as well. We discuss the TraXP/X business model radar below.

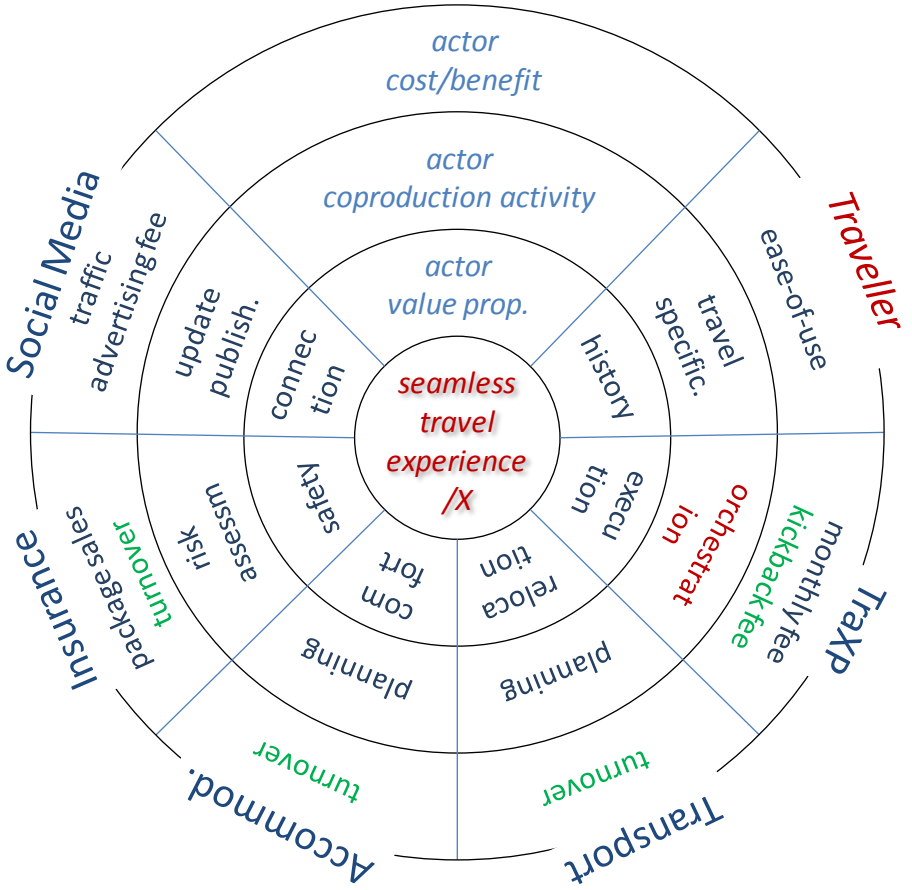

Figure 28: business model radar for TraXP/X business model (simplified)

In Figure 28, we see that the TraXP/X business model (in its simplified version) relies on six actors: TraXP (as the main actor and orchestrator), the traveler (as the consumer), the transport provider, the accommodation provider, the insurance provider, and the social media operator (as the main service providers). For each actor, its value proposition to the network, its main coproduction activity, and its main cost/benefit is specified (note that the latter is not complete in the diagram, as each actor has both costs and benefits). We explain this in more detail for three actors below.

The value proposition of TraXP is its ability to execute the processes that generate the value-in-use (seamless travel experience in the executive style). Its main coproduction activity is the orchestration of the overall process. Its benefits are a monthly fee from the traveler and kickback fees from the transport, accommodation, and insurance providers. The kickback fees are parts of the turnover TraXP generates for the service providers – this relation is indicated by color in the diagram.

The value proposition of the traveler is the travel history generated (in the broad interpretation), including travel plans that can serve as templates for others, transport and accommodation reviews, travel tips, etcetera. The coproduction activity is travel specification (both the plan and the history). The main benefit is ease of use in traveling.

For the transport provider, the value proposition to the network is its ability to provide relocation of travelers, i.e., transport them. Its main coproduction activity is planning this

relocation (as the actual relocation is performed by third parties, such as airlines). Its main benefit is turnover, which generates kickback fees from the third parties (not shown as these third parties are outside the network for this business model).

6.6 Consistency of business models

Consistency of business models is important in two ways. Firstly, business models of an organization should be consistent with its business strategy. Secondly, business models should be consistent with respect to each other.

Consistency of a business model with the business strategy can be checked by relating the concrete value-in-use of the business model to the abstract value-in-use of the business strategy. Lack of consistency can be observed in the following cases:

- The concrete value-in-use is not covered by the abstract value-in-use. This means that the business model specifies things that do not fit the specified identity of the organization.
- The concrete value-in-use is covered by the abstract value-in-use, but only fills a minimal space in this. This means that the business model hardly as any relevance (may be outdated because of evolution of the business strategy).

Consistency within a set of business models can be checked by combining (superimposing) the business radar diagrams of the business models. In this way, potential problems like the following can be detected:

- Overlapping business models (e.g., similar concrete value-in-use, similar user group, but different cost/benefit model).
- Business models with conflicting business networks (e.g., business models that rely on strong competitors in a market).

To analyze business models for internal and external consistency, tools from other frameworks can be used as well. A business model can for instance be projected onto the well-known Business Model Canvas [Oste10] to allow discussion in the BMC community – note however, that the BMC is not tuned towards service-dominant business. A business model can also be analyzed using the business model aspect of the BOAT framework [Gref10]. This latter option is further elaborated in Sections 14.3 and 14.4.

7 The lower sandwich filling: service compositions

In this chapter, we discuss the details of the service composition layer of the sandwich model. As discussed before, the service compositions provide the agile operationalization for business models as discussed in the previous chapter.

7.1 The essence of service compositions

The essence of service compositions is to make complex functionality available to a market by combining a set of simpler functionalities realized as services. The realization of the combination should be light, such that the composition approach is agile. The simpler functionalities should be standardized within an organization (or even across its boundaries), such that these functionalities can be reused. This can be clearly illustrates by taking the analogy with Lego blocks: using the same simple, standardized building blocks, one can create all kinds of buildings (the compositions).

7.2 Two types of compositions

Service compositions exist in two basic types: the process type and the mash-up type. In practice, we also find hybrid types, which have characteristics of both basic types. For reasons of clarity, we restrict ourselves to the two basic types in the sequel of this chapter.

The process type is typically used for strictly sequenced business interactions in which the activities of multiple actors need to be synchronized in time and information needs to passed between these activities. In the process type, there is an explicitly managed (and possibly complex) state of a service delivery, where the management of the state is the responsibility (or even the added value) of the service orchestrator.

The mash-up[16] type is typically used for free-form business interactions in which a single actor invokes the functionalities of a number of other actors. In the mash-up type, there is an implicitly managed (and usually simple) state of a service delivery, where the management of the state is the responsibility of the service consumer.

7.3 Business processes: sequenced under provider control

The process type is used for compositions where multiple actors (business organizations) go through a well-defined sequence of steps. As such, business process management

[16] The term 'service mash-up' often has a technological connotation (see for example [Carl08]), but we use it here to indicate a 'free-style' combination of business services from the business point of view (i.e., functionality point of view).

(BPM) is used to operationalize service composition in an agile way: “BPM enhances the ability of businesses to react to changes in the business environment” [CapG12].

A process definition is required to make sure the individual actors remain well-synchronized in the execution of the composition. Performing this sequence may take considerable time. After the sequence is completed according to definition, the composition execution is terminated.

Each defined service composition corresponds to a *business process type*. Each invocation of a service composition corresponds to a *business process instance*. The state of an individual service composition invocation is the state of that business process instance (which is typically managed automatically by a business process management system, as we will see in Chapter 9).

7.4 TraXP

The TraXP/X business model is implemented by a business process composition. This is illustrated in Figure 29, where the */X* module in the SC layer contains the composition orchestration, i.e., the specification of the business process that controls the properly sequenced execution of the services in the BS layer.

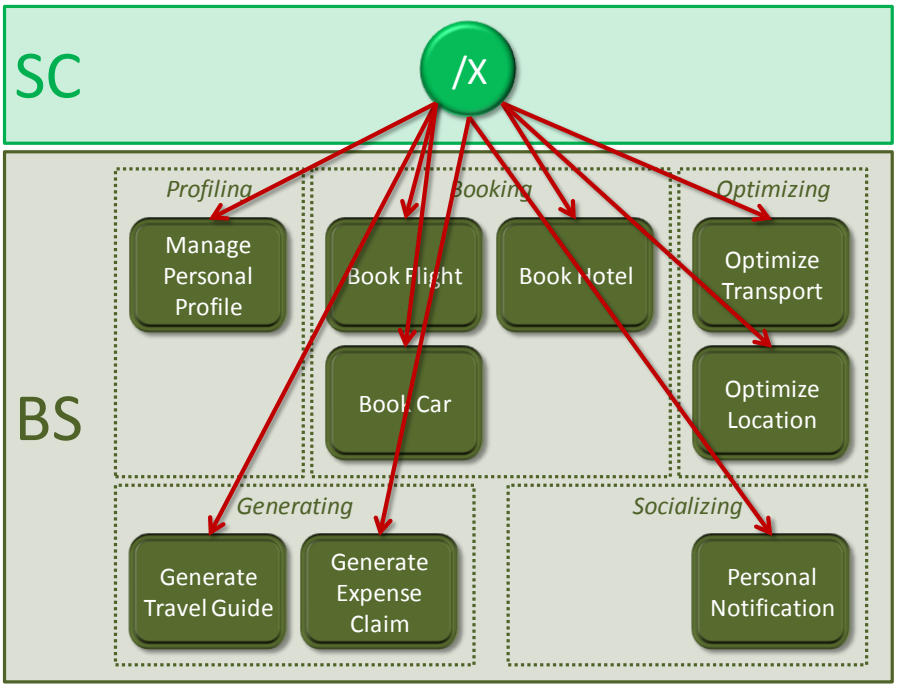

Figure 29: service composition for TraXP/X business model (external services omitted)

The business process executed by the TraXP/X service composition is shown (in a simplified form) in Figure 30.

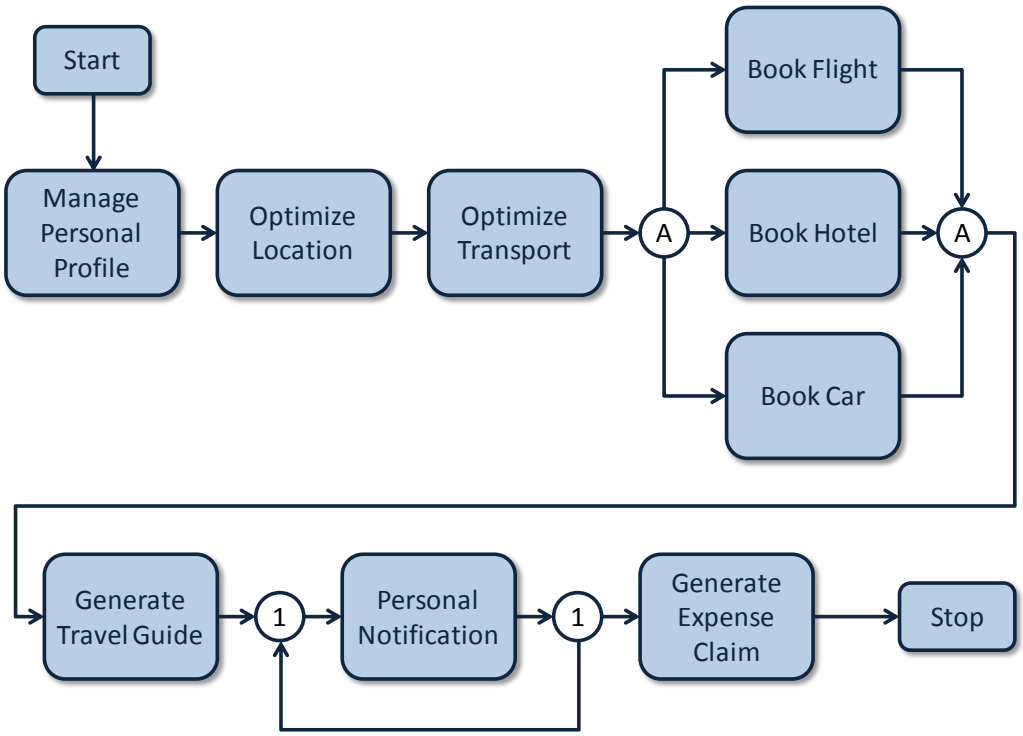

Figure 30: business process executed by TraXP/X service composition

7.5 Service mash-ups: free-form under consumer control

The mash-up type is used for compositions where a single actor can activate individual services in the composition at own will, i.e., without a sequence predefined by the service provider. The composition execution is terminated when the actor uses a termination service.

This type of service composition can be considered a simple form of user-driven case management: the user manages his own case by activating services in a non-prescribed way.

Comparable to the situation with process-based service compositions, each defined service composition corresponds to a *mash-up type*. Each invocation of a service composition corresponds to a *mash-up instance*. A mash-up instance is very light-weight from a provider point of view, as there is no state management by the provider (but the provider may want to be aware that a mash-up instance is active, e.g. for CRM purposes).

7.6 TraXP

The TraXP/F business model is implemented using a service mash-up, as there is no explicit business process management at the provider side.

An implementation as a UI-based service mash-up is shown simplified in Figure 31. Here we see a window interface, where all business services are offered to the user on a stand-alone basis. The connection between the services has to be maintained by the user.

Figure 31: TraXP/F UI-based service mash-up

7.7 Deducing service compositions from business models

In a business model radar diagram (as discussed in Section 6.2), we specify the key activities per partner that contribute to the realization of the value-in-use. This is done in the activities ring of the radar diagram, as highlighted in Figure 32.

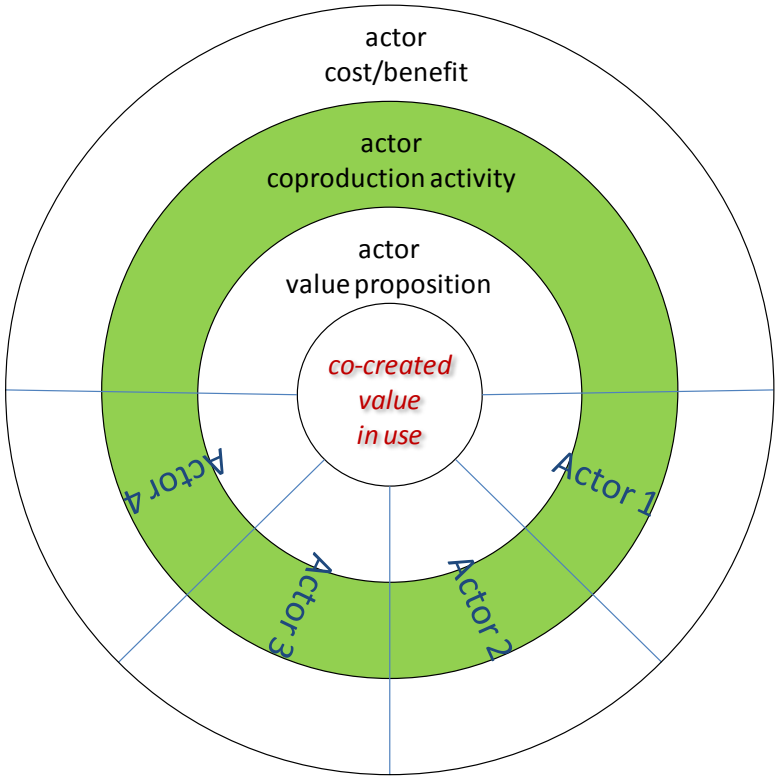

Figure 32: business model radar with activities layer highlighted

We can create the basis for a service composition by mapping these key activities to business services in a business service catalog. Typically, one key activity can require a number of business services, as the aggregation level of key activities can be higher than that of business services (never lower – this would be faulty business model design). In other words, key activities in a business model and business services have a *1:n* mapping. In mapping key activities to business services, we create a list of business services required for the realization of a specific business model.

After we have created the list of used business services, we decide on the type of composition (process-based or mash-up-based) and, if so required, on the execution sequence of the services (i.e., on the business process definition in case of a process-based composition).

7.8 TraXP

We illustrate the mapping of a business model to a service composition by means of the TraXP/X business model. The radar diagram of this model is described in Section 6.5 – we repeat it in Figure 33 with the activities ring (i.e., the key activities) highlighted.

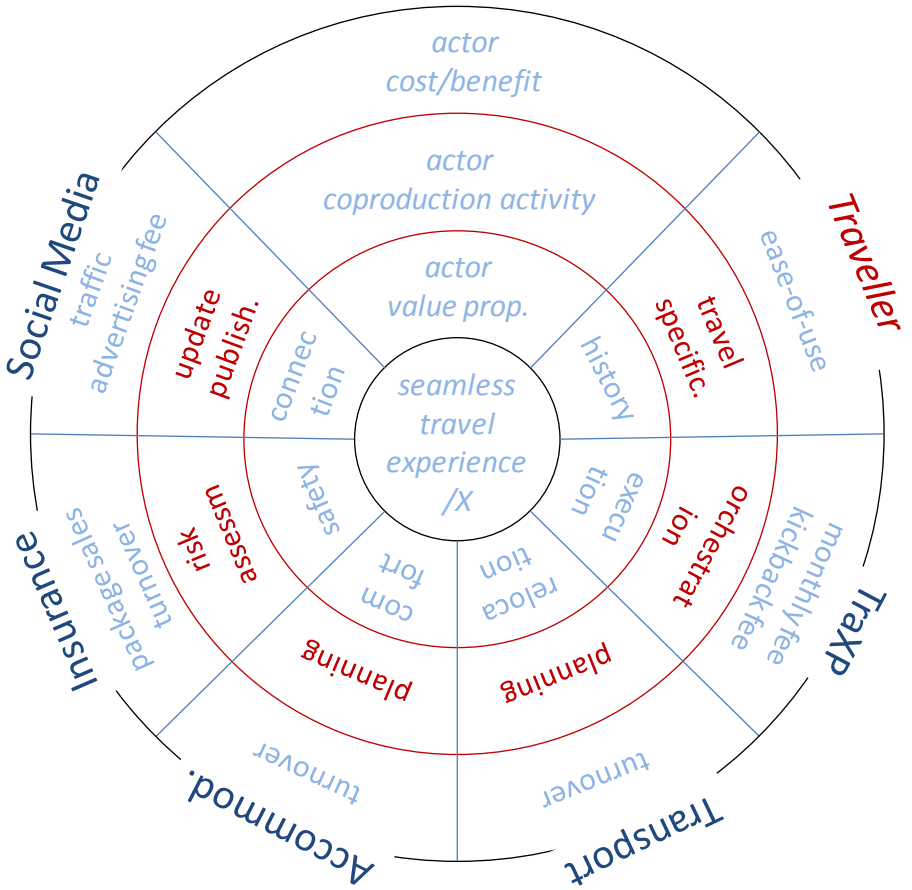

Figure 33: TraXP/X business model radar with activities highlighted

In Figure 34, these key activities are mapped to the service catalog of TraXP (this figure is a refinement of Figure 29 – the mapping is simplified for illustration purposes).

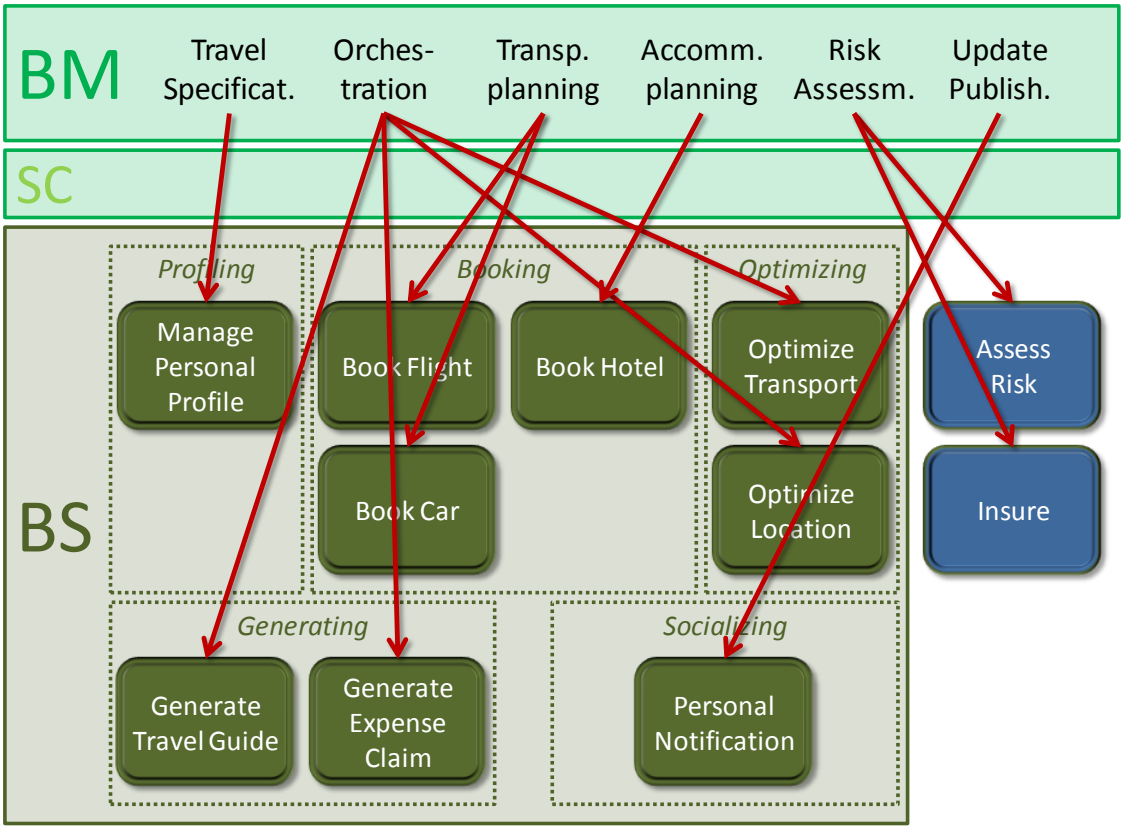

Figure 34: mapping key activities of TraXP/X business model to business services

As shown in the figure, key activities for which no business services exist in the TraXP business service catalog (as shown in Figure 22) are mapped to external services. Other services are in the catalog, but may be located internally to TraXP for orchestration purposes (and make use of functionality of external partners, i.e., be placeholders) or may be located externally to TraXP (i.e., they need to be classified still in the way described in Section 5.3).

7.9 Cost/benefit analysis of a service composition

Once we have a fully specified service composition, we can make a detailed cost/benefit analysis of this service composition. This includes the following:

- Per used service invocation, a cost or benefit can be specified, ideally based on the service level agreement for that service (as discussed in Section 5.1).
- Based on the specification of a mash-up or business process, an estimate can be made of the invocation pattern per composition invocation (typically, this coincides with a client order).
- Based on the per-invocation figures and the invocation pattern, a cost/benefit figure can be constructed per composition invocation.
- The per-composition invocation figure can be translated on a per-period figure by making an estimate of the number of composition invocations in that period (i.e., the number of customer orders per period in a business model).

The figures resulting from the above analysis form the variable costs/benefits of a service invocation. These figures can be used in two ways:

1. As a service invocation corresponds *1:1* with a business model, the precise quantitative figures of the service invocations can be confronted with the qualitative or imprecise quantitative figures of a business model specification (resulting from the cost/benefit layer of the business model radar diagram as discussed in Section 6.2).
2. Service invocations make use of resources encapsulated by the invoked services (as discussed in Section 5.1, which are partly associated with fixed costs). As such, an analysis can be made of the percentage that a service invocation makes use of these resources and consequently of the percentage that the fixed costs of a resource are covered by a specific service composition and hence a specific business model.

8 Supporting business: the information system pyramid

In this document so far, we have looked at the business side of a service-dominant business framework. The actual implementation of business has not yet been discussed. For the implementation, we require support in the form of organizational processes and automated systems. Agile, service-dominant, business in networks typically requires automated support to be run both effectively (for example, on a 24x7 basis) and efficiently (swift and cost-effective). Therefore, in this chapter, we add support in the form of business applications (information systems in the broad meaning of the term) to our approach.

We first show that the – perhaps deceitfully obvious – approach of adding another layer to the pyramid is not the right way to go. Then, we present the proper way to look at things: we extend the framework with an additional pyramid covering business support in the form of information system elements. Next, we go into details of this new pyramid.

8.1 Don't extend the business pyramid

To extend the service-dominant business pyramid model we have developed in Chapter 2 (see Figure 4) with (automated) support in the form of information systems, we might consider extending this pyramid with another layer to its bottom. This is, however, not a good idea for two reasons:

1. Firstly, going from high-level business concepts to low-level business concepts is something different than going from business concepts to information system concepts. In other words, we are talking about distinct design dimensions: aggregation/abstraction on the one hand and realization on the other hand [Gref12].
2. The information systems do not only support the business services layer, but also the service compositions layer (and maybe even more, as we will see in the sequel of this chapter). This means that a support layer cannot interface only with the business services layer (as would be implied by a new layer below the business services layer).

So, we can conclude that an extended pyramid as shown in Figure 35 is incorrect (as illustrated by the subtle, red cross). Consequently, we should not add another 'slice of bread' to our SD business sandwich. In the next section, we will see how to extend the pyramid model in a better way.

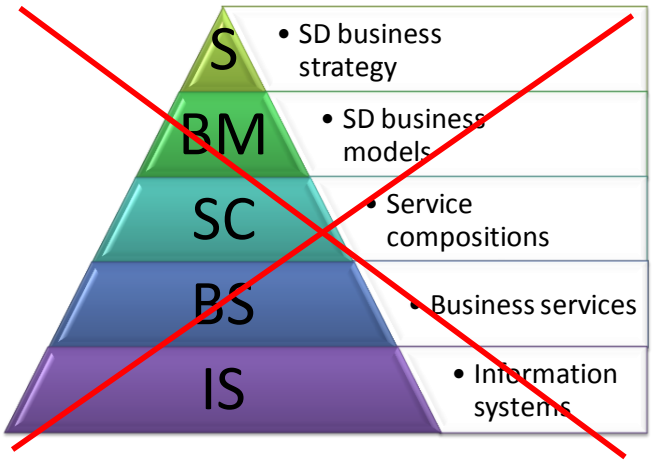

Figure 35: automated support is not *another layer*

8.2 A pyramid for business support

Instead of adding another layer to the business pyramid, we add another dimension to the pyramid model. This dimension is the realization dimension [Gref10]: the dimension that 'stretches' from business requirements to implementation in technology. In this dimension, we add a second pyramid 'behind' the one we have been discussing so far (as illustrated in Figure 36). We call this pyramid the Information System pyramid. Using the sandwich metaphor (see Figure 9), the Information System pyramid can be seen as a plate supporting the business sandwich[17].

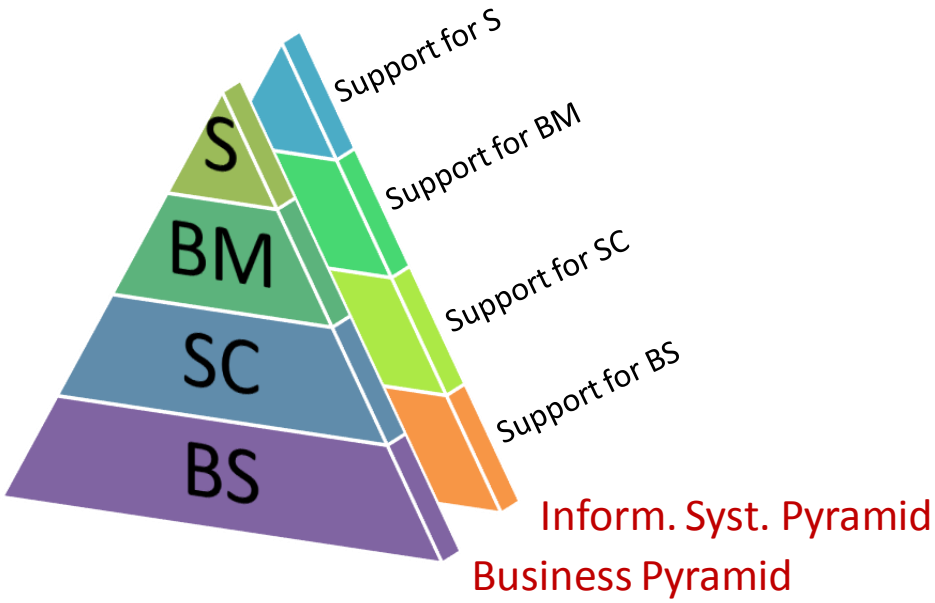

Figure 36: business and information system pyramids

[17] This metaphor is slightly imperfect, as we typically don't put sandwiches on their side on plate, but the metaphor does convey the 'support' notion.

The Information System pyramid describes the support for the Business Pyramid by information systems. The term 'information system' is used here in the broad sense: it covers both automated applications and manual information processing.

The Information System pyramid has four layers like the Business Pyramid, where each layer contains (automated) support for the business functionality in the corresponding layer of the business pyramid (as also shown in Figure 36).

8.3 The layers of the Information Systems pyramid

Each of the four layers in the Information System pyramid caters for different types of business activities (in the corresponding layers of the business pyramid):

- The business strategy layer of the IS pyramid contains support for development and operation of business strategies. As strategies are not 'directly executable', the emphasis is here on design-time support.
- The business model strategy layer contains support for development and operation of business models. As business models are not 'directly executable', the emphasis is here on design-time support.
- The service composition layer contains support for development and operation of service composition. Service compositions are executable, so the run-time aspect is important here.
- The business services layer contains support for development and operation of business services. Business services are executable, so the run-time aspect is important here.

This means that each layer contains different types of information systems, as illustrated in the information system application typology overview of Figure 37. We discuss the four layers in more detail in the sequel of this section.

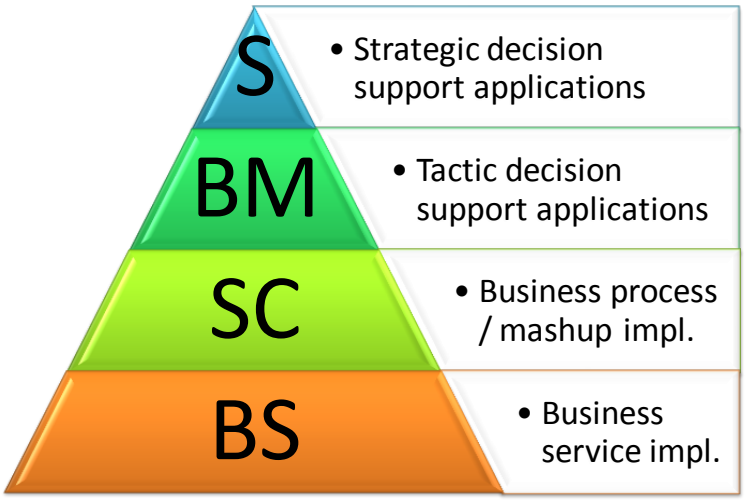

Figure 37: application typology

Note that we are talking here about an information system application typology (i.e., about the support of specific business activities), not about an information system

platform typology (i.e., about generic platforms supporting specific applications). We address the platform issue in Chapter 9.

8.3.1 Business strategy layer

The business strategy layer of the IS pyramid contains information system applications (and organizational procedures) that support design and evolution of a business strategy. These applications are for example strategic decision support applications. These can be used, for example, for long-term what-if analyses. We may also find tools for qualitative analysis, for example to support the identification of new long-term trends in specific business domains.

8.3.2 Business model layer

The business model layer contains information system applications that support the conception of business models. Here, we find for example decision support applications, targeted at providing (quantitative) information used in the design and analysis of business models. Advanced tools for identifying and analyzing potential business partners are also of great importance here.

Apart from tools that relate to the creation of individual business models, we may also find tools that help managing a set of business models. These tools help in maintaining consistency between business models, help in analyzing possible overlap in business models, and help in reusing existing business models in the design of new business models.

8.3.3 Service composition layer

The service composition layer contains implementations of service compositions described in the corresponding layer of the Business pyramid. These implementations can have the form of executable specifications of business processes or service mash-ups.

This layer also contains tools to create and deploy these implementations. Prototyping tools may also be important to quickly create prototypes or mock-ups of new service compositions, such that new business models can easily be 'visualized' in an operational way. In advanced situations, even simulation tools (more specifically business process simulation tools) may be used to assess quantitative behavior of business processes before detailed implementation and deployment of a service composition.

8.3.4 Business service layer

The business service layer contains the implementations of business services described in the corresponding layer of the Business pyramid. These implementations preferably have the form of executable service specifications.

Similar to the service composition layer, the service layer also contains tools to create and deploy business service implementations. The possibility to create easily functional mock-ups of new services contributes to the ability to create prototypes of mock-ups of service compositions, as mentioned above.

8.4 Agility in the IS pyramid

The business pyramid as discussed in Section 2.2 has been designed with business agility as a main starting point. As the information systems pyramid supports the business pyramid, agility is also a main issue for the information systems pyramid.

Agility in the IS pyramid should be completely driven by the agility of the business pyramid. In other words: changes in the business pyramid should cause changes in the IS pyramid, not the other way around[18]. This is illustrated in Figure 38.

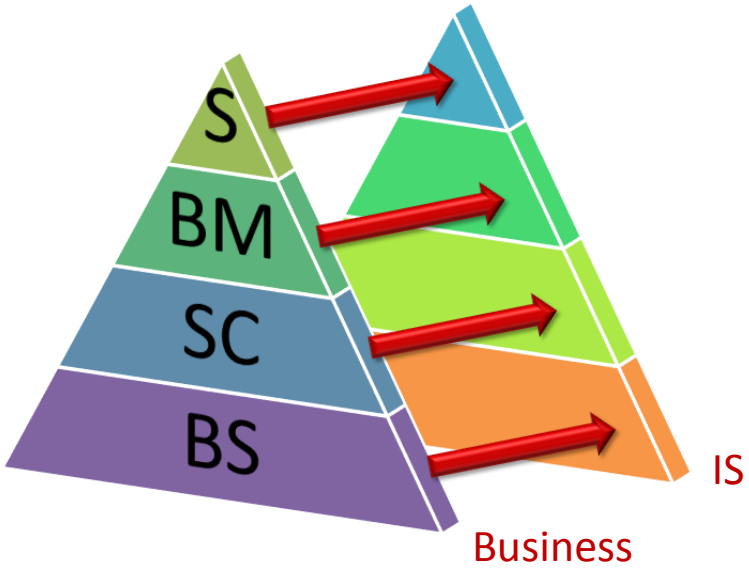

Figure 38: agility relation between business and IS pyramids

Note that independent changes in the IS pyramid (i.e., not triggered by changes in the business pyramid) are allowed as long as these changes do not require changes in the business pyramid. Typically, this concerns the introduction of more effective or efficient information system applications (or organizational procedures) with the same business goal.

8.5 Levels of support

In current business practice, we can find structured (automated) support for business services (business functions in non-service-oriented contexts) and for service (or function) composition – the latter often hardwired into applications and sometimes flexibly composed into business processes.

This mean that structured support for the layers of the sandwich model in current practice is often limited to the service layer of the business pyramid, or the service composition and business service layers of the pyramid. This is illustrated in Figure 39.

[18] Note that new possibilities in IT can have an influence on a business strategy or on the development of business models, but this is a contextual influence (located rather in the platform pyramid discussed in Chapter 9 than in the IS pyramid discussed here), not a reverse version of the arrows shown in Figure 38.

Figure 39: one and two levels of the pyramid supported

This situation implies that development of business strategy and business models is currently often performed in an ad-hoc fashion without structured support (by defined processes and/or automated applications): these levels in the business pyramid have no counterparts in the information systems pyramid.

Obviously, the more attention goes to agility, the more effort goes into the design of business models, and the more support it will require. This means that – leveraging the situation in Figure 39 – we need to provide structured support for business model design as well. This situation is illustrated in Figure 40.

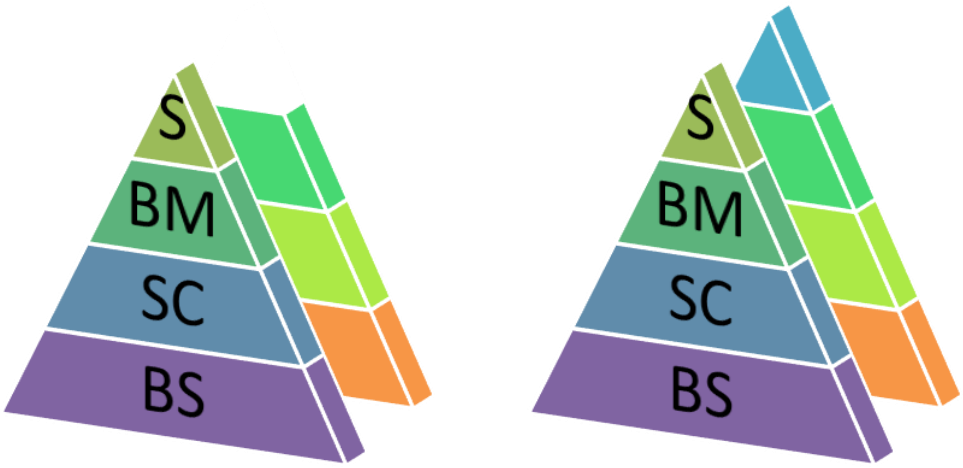

Figure 40: three and all four levels of the pyramid supported

In the completely supported situation, also the strategy level is covered by structured support.

9 Supporting information systems: the platform pyramid

In Section 8.2, we have added the Information System pyramid to our approach to provide support for the layers in the Business Pyramid: it is a 'plate for the sandwich'. The support provided in the IS pyramid, however, also requires support itself. For this reason, we add a third pyramid to the framework in this chapter.

9.1 The platform pyramid

Automated applications require underlying platforms – business process management (BPM), for example, requires a business process management system (BPMS), which can vary from a traditional workflow management system [Leym99] to a system supporting business process management in the cloud [Stoit12]. Manual applications require platforms as well, for example collaboration platforms like document management systems or email systems.

For this reason, we add a third pyramid to our approach which we call the Platform pyramid (as shown in Figure 41). The third pyramid extends the realization dimension [Gref10] that we started by adding the Information Systems pyramid to the Business pyramid. Continuing our sandwich metaphor, the Platform Pyramid can be seen as 'the table that supports the plate' (being the IS Pyramid).

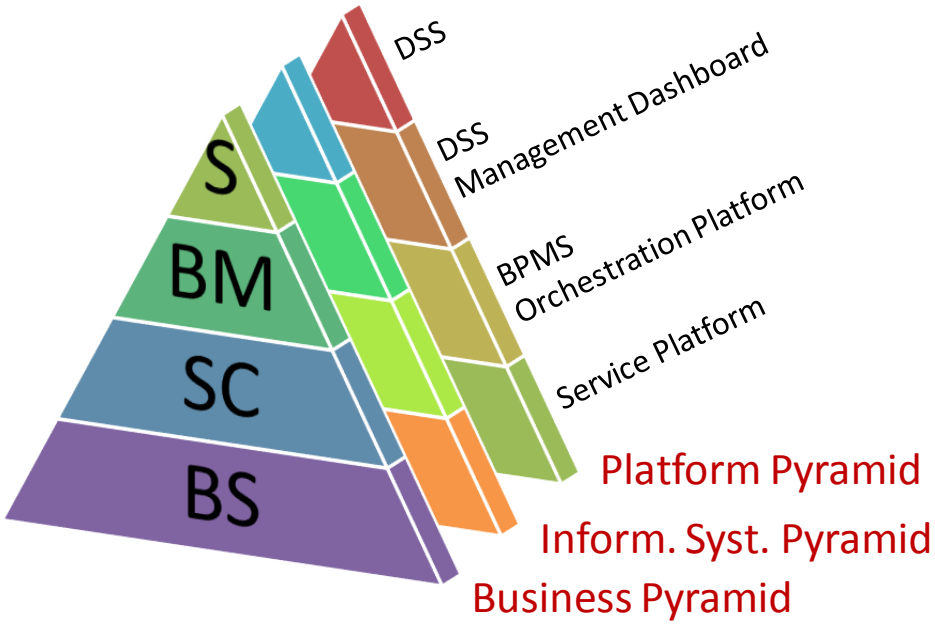

Figure 41: adding the Platform pyramid

9.2 Agility in the platform pyramid

Where it comes to agility, there is an important difference between the Information System pyramid and the Platform pyramid. In the Information System pyramid, we create agility by having middle layers in the pyramid with flexible structures: we create and dismiss business process implementations and their supporting service composition implementations as the corresponding structures in the Business pyramid demand. In the Platform pyramid, however, we do not want to have flexible structures in the middle layers, as this would mean adopting the system landscape architecture continuously. Instead, we want platforms that support application flexibility (but have stable structures themselves). This typically is a reason to explicitly decouple application architecture from platform architecture. Concluding, we often don't require *functional* agility in the platform pyramid at all.

We may, though, require flexibility in the platform pyramid where it comes to capacity or performance: as we revolve business models, the functional requirements to the platform pyramid don't change that much (they are more dictated by the requirements of the stable business functions), but the non-functional requirement may change dramatically. A successful new business model, for example, may temporarily lead to high throughput requirements to the platforms used. A modern way to deal with this required *capacity elasticity* is the use of cloud computing [Arm09].

9.3 A platform reference architecture

In this section, we map the platform Pyramid to a corporate reference architecture, which is a high-level blueprint for the organization of automated systems and data sets in a service-dominant organization. First, we present an overview with four layers conforming to the four layers of the Platform pyramid. Then, we discuss details of each of these four layers. We start at the bottom layer and move up from there, as layers depend on the functionality of the underlying layer.

9.3.1 Overview of the reference architecture

Given the four layers in the information system and platform pyramids, we can design a corporate reference information system architecture showing the main information system structure for the support of agile, service-dominant business. This architecture is shown in Figure 42. In the figure, we see the four layers of our service-dominant business sandwich. We briefly discuss each layer below, moving from bottom to top.

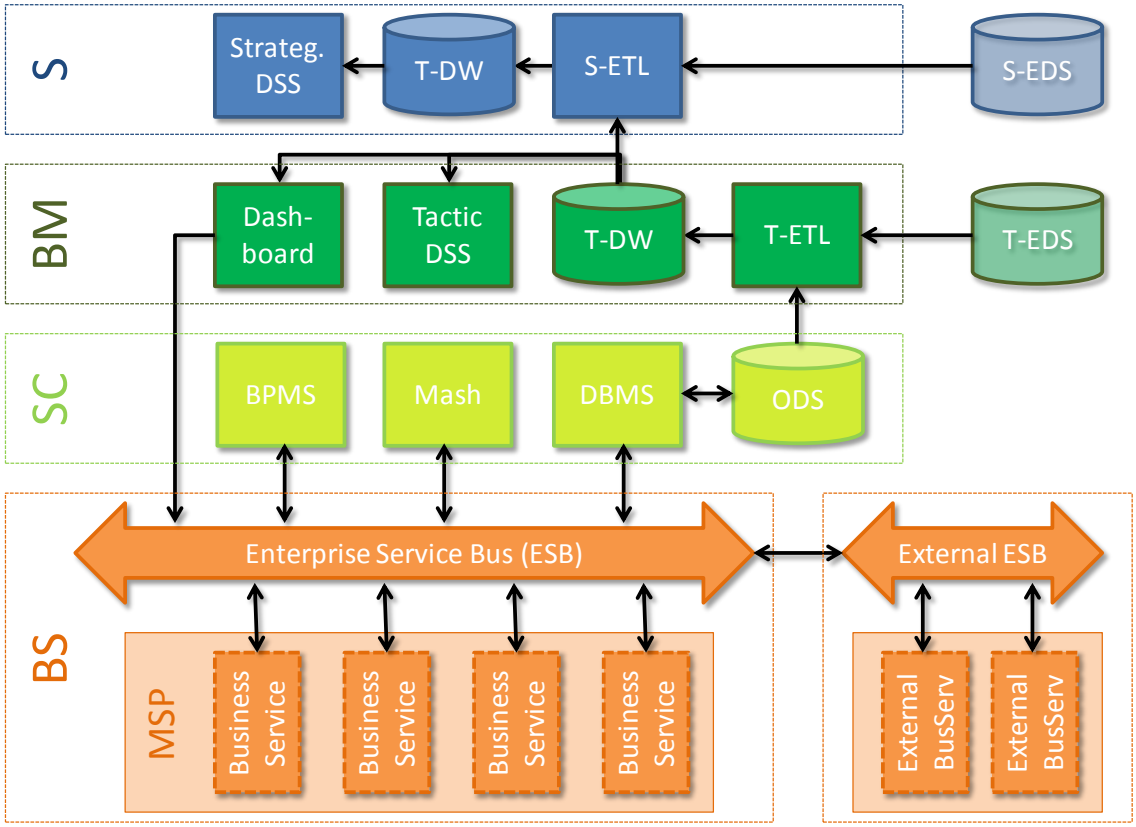

Figure 42: corporate reference architecture for agile, service-dominant business

9.3.2 The business services layer

The business service layer (BS) contains the support for the implementations of the elementary business services (note that the actual services belong in the Information Systems pyramid – they are included in Figure 42 for clarity, but dotted for this reason). This includes the (user) interfaces to customers[19]. The services are implemented in a managed service platform (MSP) that facilitates development, maintenance and evolution of the services. All services are connected to the enterprise service bus (ESB). There are *no* other connections to services. The ESB is connected to external ESBs for two reasons. The first reason is to allow internal service compositions to invoke external services (as discussed in Chapter 1) using the same mechanisms as invoking internal services. The second reason is to allow services of external customers to invoke internal services that start service compositions. Typically, each external party operates a single ESB. Note that it is not relevant whether an external party operates an MSP (because we are only interested in the run-time behavior of external services).

9.3.3 The service composition layer

The service composition layer (SC) contains the mechanisms for composing elementary services into high-level services that are offered to customers. In this layer we find a

[19] Note that cross-function user interfaces, such as business dashboards, are in fact elementary services of their own that retrieve data from other elementary services.

business process management system (BPMS) and a service mash-up platform (Mash) to support the two kinds of business service composition that we have discussed in Section 7.2. A database management system (DBMS) takes care of data management across elementary business services. The data is stored in the operational data store (ODS) managed by the DBMS. Note that the ODS contains only operational data directly related to the execution of business services and service compositions.

9.3.4 The business model layer

The business model layer (BM) supports the definition and control of business models. The automated tools found here form the basis for the agility of the business organization. A tactic decision support system (T-DSS) is used to explore the possibilities (also in quantitative ways) of new business models. A business dashboard is used to monitor the performance of business models implemented as service compositions, and adjust composition parameters where necessary. The dashboard has a user interface that is closely coupled to the concepts in service-dominant business models, as discussed in Chapter 6. New parameter values are communicated by the dashboard to the composition engines through the ESB. Both the T-DSS and the dashboard use data from a data warehouse (T-DW). This T-DW contains management data that is distilled from operational data in the ODS using a module with extraction and transformation logic (T-ETL). Also, data from external data sources (T-EDS) can be used. The T-DW contains historical information to support exploration of time series.

9.3.5 The strategy layer

The strategy (S) layer contains a strategic decision support system (S-DSS) to support decision making related to the evolution of the service-dominant business strategy as discussed in Chapter 4. To enable making quantitative analyses (such as strategic what-if analyses), the S-DSS uses data in a data warehouse at the strategic layer (S-DW). The S-DW is filled with data using an ETL module at the strategic layer (S-ETL). Also, data from external sources (S-EDS) can be used. Decisions made in the S-DSS influence the set of services in the BS layer (as illustrated before in Figure 10). This is not an automated coupling, however. Therefore, no interface is included in the reference architecture for this purpose.

9.4 Levels of support

As discussed for the information systems pyramid in Section 8.5, we can also have different levels of support in the platform pyramid, i.e., several situations where it comes to the number of levels of the pyramid covered.

For the platform pyramid, the number of levels covered by actual systems is completely dependent on the number of levels covered in the information systems pyramid: they need to be the same (as illustrated in Figure 43 and Figure 44). If there is a level covered in the information systems pyramid, it requires a platform to run on (even if the covering in the information systems pyramid is not highly automated, it typically requires general purpose automated tools). Having a level covered in the platform pyramid that has no counterpart in the information systems pyramid is (obviously) a waste.

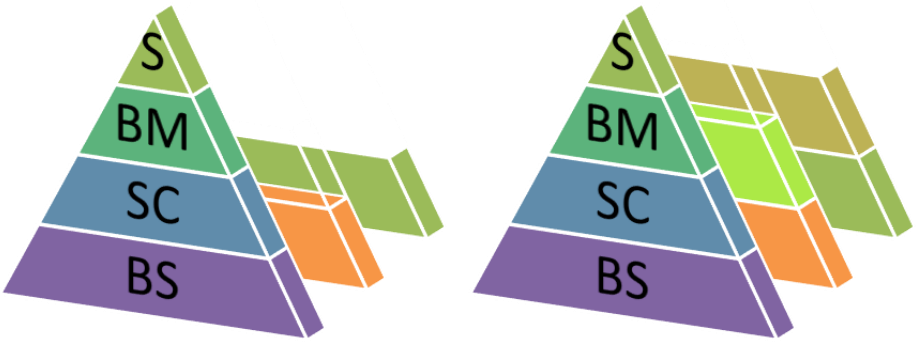

Figure 43: one and two levels of the pyramid supported

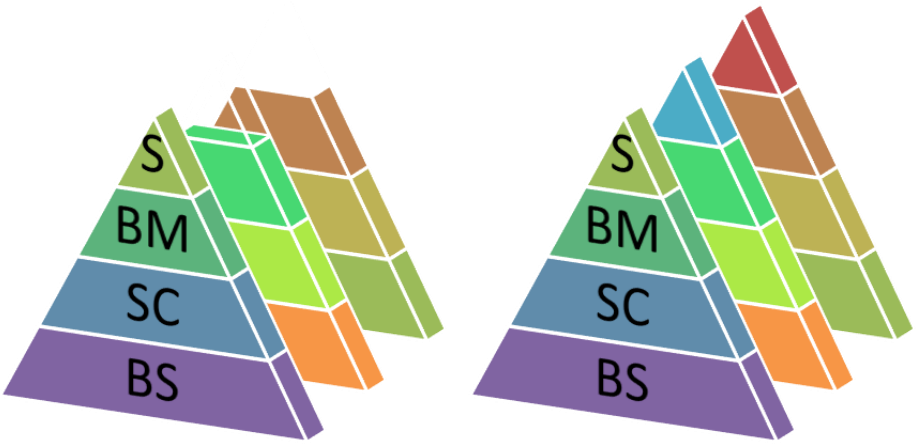

Figure 44: three and all four levels of the pyramid supported

Note that in the *BASE-X* approach, full support in the platform pyramid is not a cost center to a business organization, but rather an opportunity center: IT platforms enable the agile, service-dominant way of business. In other words: IT platforms enable profitability of an organization.

10 Conclusions

In this chapter, we conclude this document. We first summarize the essence of the *BASE/X* approach described in this document. Then, we show that the approach consists of two main aspects: method and tools. We end with acknowledgments of all the people who have contributed in one way or the other to the contents of this document.

10.1 Summarizing the essence of the approach

This document describes the *BASE/X* approach for the design of business following the service-dominant logic [Lusch07]. An overview of the main elements of this approach is shown in Figure 45.

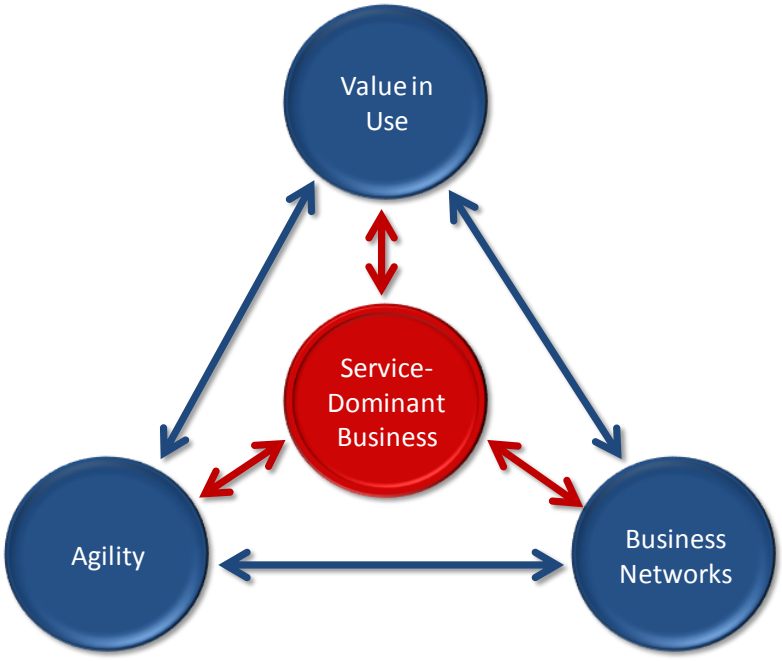

Figure 45: overview of main elements of approach

Value-in-use is the first main ingredient of service-dominant business: don't think in terms of resources (assets), but think about what the added value is of the *use* of these resources. Thinking in terms of value-in-use completely changes thinking about concepts like asset ownership or asset location. Becoming really service dominant requires a radical shift in business paradigm – where extreme thinking may set interesting new horizons and incremental thinking may only slowly move asset-orientation.

Agility is the second main ingredient of service-dominant business. On the one hand, service-dominant markets require higher levels of agility to survive (as their products are more 'fluid'). On the other hand, agility brings new advantages by itself: "agile organizations realize higher margins on their goods and services by taking advantage of market opportunities" [CapG12].

Business networks form the third ingredient of service-dominant business. As complexity of products (services) grows and market competition increases, companies are forced to rely on possibly large networks of other companies to deliver their market offering – a development that we have already observed in non-service-dominant markets leading to concepts like instant virtual enterprises [Meha10] or even plug-and-play factories [Schul12]. Obviously, companies that see networking as an opportunity instead of a necessity may gain substantial advantage here.

10.2 The two main aspects of the approach

The *BASE/X* approach presented in this document describes a well-structured way to address the analysis and design of service-dominant business. Note that the approach consists of two main aspects.

In the first place, there is the conceptual aspect. This aspect defines the way of analyzing and designing, separating strategic from tactical concerns. This is abstractly illustrated in Figure 6. This conceptual way of thinking is based on the concept model described in Figure 14 (and further extensions). This aspect of the approach be considered the methodological part of the approach: it describes in which order things can (or should) be addressed.

In the second place, there is the tooling aspect. This aspect defines tools that can be used in analyzing and designing. Tools can be used for filling in the four layers in the conceptual pyramid framework and for mappings between these layers. Tools are specification techniques, such as the Service-Dominant Strategy Canvas or the Service-Dominant Business Model Radar, or design aids such as the Business Service Criteria table.

The tools presented in this document have been designed to fit in well with the methodological aspect of the approach. They are, however, certainly not (yet) complete and certainly also not the only possible choice – there may be alternatives for several of the tools presented.

10.3 The added value of *BASE/X*

The 'added value' in practice of the *BASE/X* approach described in this document can be found in the following main aspects:

1. *BASE/X* has two main ingredients of modern business at its very core: service-orientation and networked business. These ingredients are not elements 'that should also be taken into consideration', but are essential starting points. Correct use of the approach in practice will hence lead to automatic adaptation of these two main ingredients in business design and operation.
2. *BASE/X* explicitly addresses the complexity of service-dominant business. Service-dominant business design is not a simple issue. If it would be, it would be easy to replicate and hence never be a unique selling point in a market. The approach does, however, provide a basis for well-structured management of this complexity by adopting an engineering approach to business design (in contrast to other approaches that have a focus on idea generation only).

3. *BASE/X* explicitly separates the more stable (business strategy and business services, which evolve) from the more volatile (business models and service compositions, which revolve). This provides a separation of concerns that addresses the complexity mentioned above and greatly adds to the agility of an enterprise.
4. *BASE/X* explicitly addresses the fact that a one-shot design of a new business model 'doesn't cut it' in modern markets to survive as a company: the power of modern business is not in a change, but in the ability to keep adapting to changing circumstances (as reflected in the cycles in Figure 7). Given a stable context (the bread of the sandwich), business model design and implementation is a continuous process (the fillings of the sandwich). The latter can be performed relatively light-weight (ultimately leading to plug-and-play business design), allowing for short times-to-market for new business models.
5. *BASE/X* explicitly decouples business design from the design of business support (be it in organizational processes or automated systems) and the underlying platforms. This is clearly reflected by the separation of concerns in the three-pyramid model (Figure 41). This strongly adds to the agility of business and its support.

10.4 Acknowledgments

Many people have contributed to the realization of the ideas presented in this document by participating in discussions about the approach and the application of the approach in industrial practice. Thanks go to many employees and associates of (in alphabetical order) De Lage Landen International B.V., Portbase, and University of Vienna. Special thanks go to Marco Comuzzi of TU/e for his feedback on this document.

11 References

[Alon04] G. Alonso, F. Casati, H. Kuno, V. Machiraju; *Web Services: Concepts, Architectures and Applications*; Springer, 2004.

[Arm09] M. Armbrust c.s.; *Above the Clouds: A Berkeley View of Cloud Computing*; Berkeley University, 2009.

[Cama10] L. Camarinha-Matos, X. Boucher, H. Afsarmanesh (eds.); *Collaborative Networks for a Sustainable World*; Proceedings 11th IFIP WG 5.5 Working Conference on Virtual Enterprises; Springer, 2010.

[CapG12] Global Business Process Management Report; CapGemini, 2012.

[Carl08] M. Carlson, A. Ngu, R. Podorozhny, L. Zeng; *Automatic Mash Up of Composite Applications*; Proceedings 6th International Conference on Service-Oriented Computing 2008; Springer, 2008; pp. 317-330.

[Gref10] P. Grefen; *Mastering E-Business*; Routledge; 2010; ISBN 978-0-415-55787-0.

[Gref12] P. Grefen; *Business Information System Architecture*; 1BM41 Lecture Reader; Eindhoven University of Technology, 2012.

[Hend93] J.C. Henderson, H. Venkatraman; *Strategic Alignment: Leveraging Information Technology for Transforming Organizations*; IBM Systems Journal, Vol. 32, Nr. 1, 1993.

[Hunt04] S. Hun, C. Derozier; *The Normative Imperatives of Business and Marketing Strategy: Grounding Strategy in Resource-Advantage Theory*; Journal of Business and Industrial Marketing, Vol. 19, Nr. 1, 2004; pp. 5–22.

[Leym99] F. Leymann, D. Roller; *Production Workflow: Concepts and Techniques*; Prentice Hall, 1999.

[Lüft10] E. Lüftenegger, X. Zhao, S. Angelov, P. Grefen; *SerVestMeNt: A Service Oriented Business Model for the Financial Service Industry*; CoProFind Deliverable #005; Eindhoven University of Technology & De Lage Landen, 2010.

[Lüft12a] E. Lüftenegger, P. Grefen, C. Weisleder; *The Service Dominant Strategy Canvas: Defining and Visualizing a Service Dominant Strategy through the Traditional Strategic Lens*; Beta Working Paper Series 383; Eindhoven University of Technology, 2012.

[Lüft12b] E. Lüftenegger, P. Grefen, C. Weisleder; *The Service Dominant Strategy Canvas: Towards Networked Business Models*; Proceedings 13th IFIP Working Conference on Virtual Enterprises; Bournemouth, UK, 2012.

[Lüft12c] E. Lüftenegger, P. Grefen, C. Weisleder; *The Service Dominant Business Model: A Service Focused Conceptualization*; under development; Eindhoven University of Technology, 2012.

[Lusch07] R. F. Lusch, S. L. Vargo, and M. O'Brien; *Competing through Service: Insights from Service-Dominant Logic*; Journal of Retailing, vol. 83, no. 1, pp. 5 – 18, 2007.

[McKin12] *The Age of Multi-Sided Business Models*; in: *Tech-Enabled Business Trends 2012*; McKinsey, 2012.

[Meha10] N. Mehandjiev, P. Grefen; *Dynamic Business Process Formation for Instant Virtual Enterprises*; Springer; 2010.

[Moor05] G. Moore; *Dealing with Darwin*; Portfolio Hardcover, 2005.

[Oste10] A. Osterwalder, Y. Pigneur; *Business Model Generation*; Wiley, 2010.

[Port80] M. Porter; *Competitive Advantage*; Free Press, 1980.

[Port85] M. Porter; *Competitive Strategy*; Free Press, 1985.

[Shaf05] S. Shafer, J. Smith, J. Linder; *The Power of Business Models*; Business Horizons, Vol. 48, No. 3, 2005; pp. 199-207.

[Stoit12] V. Stoitsev, P. Grefen; *Business Process Technology and the Cloud: defining a Business Process Cloud Platform*; Beta Working Papers; Vol. 393; Eindhoven University of Technology; 2012.

[Port85] M. Porter; *Competitive Advantage: Creating and Sustaining Superior Performance*; Free Press, 1985.

[Schul12] S. Schulte, D. Schuller, R. Steinmetz, S. Abels; *Plug-and-Play Virtual Factories*; IEEE Internet Computing, Vol. 16, No. 5, 2012; pp. 78–82.

12 Appendix A: From in-house to integrator-only

In this appendix, we take a closer look to the distribution of services over multiple organizations. First, we take this look from the perspective of one organization, i.e., the internal versus external services issue. Then, we see how this issue can give rise to splitting up an organization based on business strategies.

12.1 In-house versus integrator-only

In realizing business compositions from underlying business services, we can decide which part of these services is implemented internally and which part externally. In this paper, we have seen a case in Figure 13, in which part of the business functions is external. This case is an intermediate point in the spectrum from complete internal to completely external, as illustrated in Figure 46.

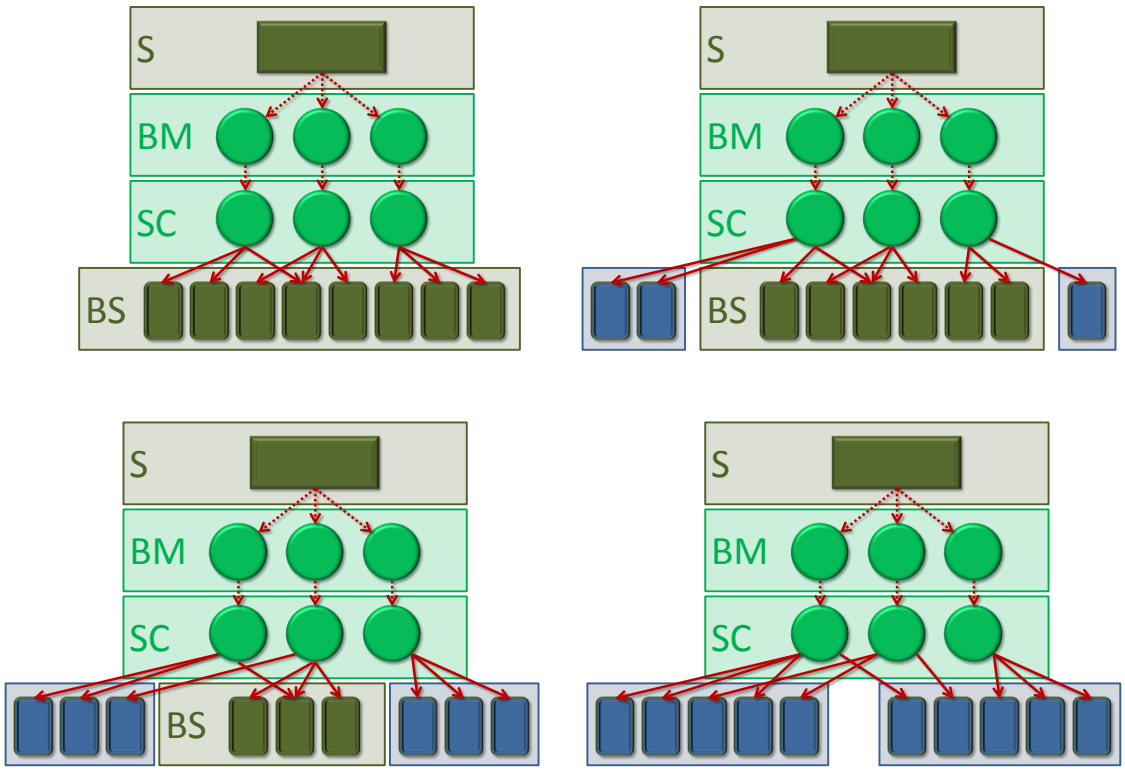

Figure 46: in-house (top-left) to integrator-only (bottom-right) spectrum

The situation in which all business services are implemented internally, we call In-House. This scenario can allow a high level of adaptation of the services to the requirements of the company, but typically at high cost, low speed and with little flexibility. This actually is quite a traditional way of looking at business, in which the distinctive value of the organization is derived from the qualities of the competences encapsulated by the

business services. We may find this in domains where the competences are highly asset-based.

The situation in which all business services are implemented externally, we call Integrator-Only [Gref10]. In this situation, the organization has no elementary business services of its own. This scenario can allow a high level of agility at low cost and high speed. The downside is that business is not based on hard-to-copy assets, and hence may be easily replicable – the distinctive value of the organization is derived from its ability to swiftly design and implement unique service compositions. This is an extreme e-business scenario. One step further is the business scenario where a business organization does not even implement specific business scenarios as an orchestrator only, but offers a service-dominant platform in which third parties can do this – and hence becomes a meta-orchestrator (this is reflected in the ideas behind the SerVestMent business model [Lüft10]).

12.2 Splitting up an organization

A hybrid business approach may be obtained by splitting an organization into two (or more parts) based on the business paradigm. One part is the traditional, not-so-agile In-House part, the other the very agile Integrator-Only part. This is illustrated in Figure 47.

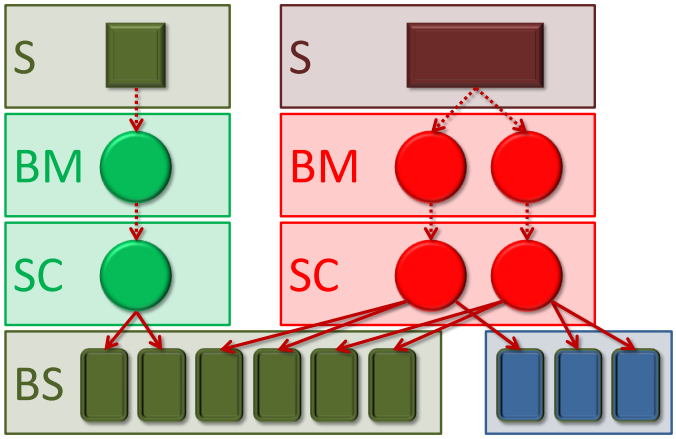

Figure 47: splitting an organization based on business paradigms

The In-House part is relatively stable and focuses on the elementary business services, on which it can build one or a few traditional business models. Hence, it is service-heavy and strategy-light. The Integrator-Only part uses business services of the In-House part (bottom-left in the figure) plus services from one or more third parties (bottom-right in the figure). It is service-light and strategy-heavy to reach the required agility.

This business split paradigm allows a business-wise separation of concerns. Each organization can focus on its own strengths, implemented in – among other things – its business culture and its physical rooting. This is comparable to the concept of 'unbundling a corporation' [Oste10], which advocates splitting up a business organization with a hybrid business model into several organizations. This can be done according to the 'fundamental' types of business models: customer relationship business, product innovation business, and infrastructure business [Oste10]. The split we have discussed

above coincides more or less with infrastructure business on the one side (left-hand side in Figure 47) and customer relationship/product innovation business on the other side (right-hand side in the figure). Note, however, that in the service-dominant approach advocated in this document, business splits are not necessarily based on the three mentioned business types, but are based on the existence of multiple business strategies (where multiple strategies can in principle be related to the same business type).

13 Appendix B: Realizing services in a hierarchical way

In this appendix, we take a short look at realizing business services in a hierarchical way. We first present the concept, then apply it to the TraXP case.

13.1 A stratified service organization

Elementary business services can be realized using the functionality of smaller services, which we call *building block services* (BBSs). Doing so, we use the principle of modularity to achieve maintainability, standardization and reuse of service functionality.

Building block service do not have external business semantics by themselves, they only contribute to creating external business semantics. Hence, using building block service is in the domain of (intra-organizational) service engineering, not (inter-organizational) business engineering.

Multi-layer structures of building block services are possible too in complex application domains, where larger building block services are realized using the functionality of smaller building block services. This approach leads to stratified (strictly layered) building block service organization.

13.2 TraXP

We can see an example in the use of building block services in the TraXP case. The elementary business service *Generate Travel Guide* is realized on the basis of the functionality of three building block services: *Retrieve Content Element*, *Compose Content Elements*, and *Generate Document Layout*. This is illustrated in Figure 48.

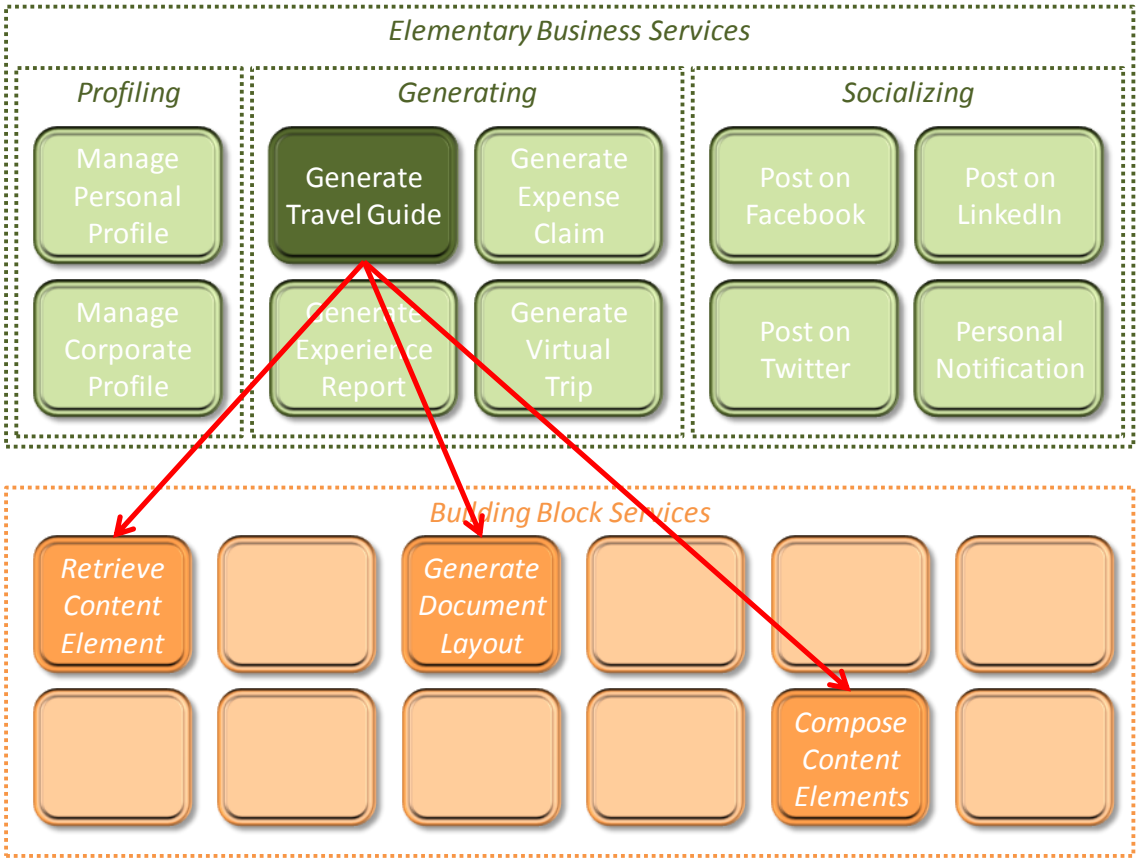

Figure 48: business service constructed from building block services

14 Appendix C: Sandwich in a BOAT?

In this appendix, we discuss the relation between the Sandwich framework described in this document and the BOAT framework [Gref10]. Both approaches aim at providing a clear structure to analyze or design complex e-business scenarios.

14.1 The BOAT framework

BOAT is a framework for structured analysis and design of e-business scenarios [Gref10]. The BOAT framework consists of four aspects (the first letters of which make the acronym BOAT):

Business (B): The business aspect describes the business goals of e-business. As such it answers the question why a specific e-business scenario exists or should exist or what should be reached. Topics can be access to new markets, reorientation of interaction with customers, leverage of efficiency levels, etcetera.

Organization (O): the organization aspect describes how organizations are structured and connected to achieve the goals defined in the B aspect. Organization structures, business processes and business functions are the main ingredients here.

Architecture (A): the architecture aspect covers the conceptual structure (i.e., the architecture) of automated information systems required to make the organizations defined in the O aspect work. As such, it describes how automated systems support the involved organizations in a conceptual way.

Technology (T): the technology aspect describes the technological realization of the systems of which the architecture is specified in the A aspect. In other words, the T aspect describes from what ingredients an e-business system is built. The T aspect covers the concrete ingredients from information and communication technology, including software, languages, communication protocols, and hardware where relevant.

14.2 Mapping the sandwich approach to BOAT

We map the BOAT framework to the most complete, high-level diagram of the Sandwich framework, which is the tri-pyramid model of Figure 41. The mapping is illustrated in Figure 49 and explained below.

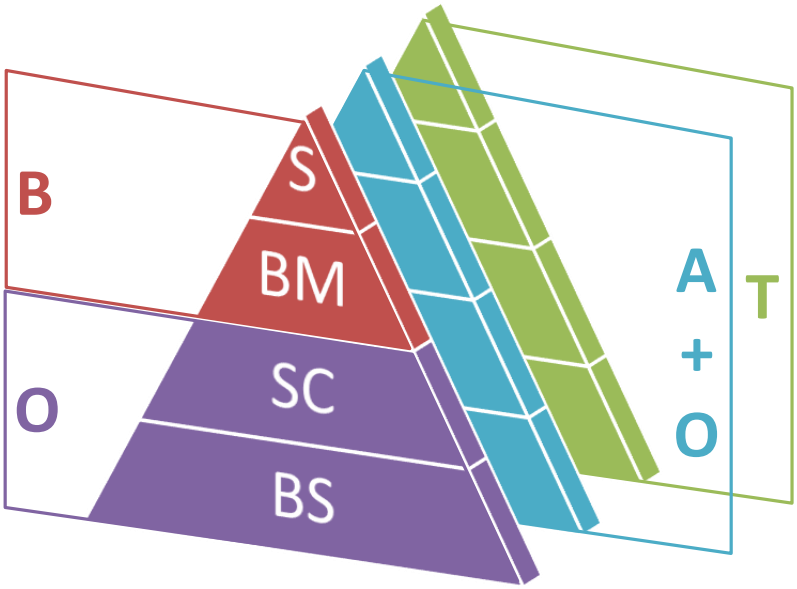

Figure 49: the BOAT framework mapped to the Sandwich tri-pyramid model

BOAT includes the Business (B) aspect, which is comparable to the Business Model layer of the Business Pyramid of the Sandwich framework. BOAT does, however, not explicitly distinguish a Strategy aspect. Hence, elements related to business strategy (e.g. in terms of business drivers) are implicitly included in the BOAT B aspect. Hence, we map the BOAT B aspect to the top two layers of the Business pyramid.

The BOAT Organization (O) aspect describes how business models are implemented without referencing technology. From an abstract point of view, this coincides with both the Service Composition and the Business Services layers of the Business pyramid. From a realization point of view, this coincides with business processes in the Information System pyramid. Hence, we map the BOAT O aspect to the two bottom layers of the Business pyramid and to the Information System pyramid.

The BOAT Architecture (A) aspect describes the application architecture of an e-business scenario. This is mapped to the complete Architecture pyramid of the Sandwich approach.

The BOAT Technology (T) aspect describes the information technologies used to embody application systems. This coincides with the complete Platform pyramid of the Sandwich approach.

14.3 Using the BOAT business model classification tool

The BOAT framework includes a business model classification tool for electronic business [Gref10]. As service-dominant business models are often highly oriented towards electronic business, we can use this tool here as well to analyze a business model or compare business models.

The tool is shown in Table 8 with short explanations – for a complete description, the reader is referred to [Gref10]. The tool is applied to TraXP in the sequel of this section.

Business model summary	
Parties	Describes the kind of parties collaboration in a BM
Objects	Describes the main object(s) traded in a BM
Time Scope	Describes the level of dynamism of collaboration in a BM
Drivers	Describes the basic reasons for a BM
Chains	Describes the way business chains are altered by a BM
Directions	Describes business directions used in a BM
Structures	Describes basic collaboration patterns in a BM

Table 8: BOAT business model classification tool

14.4 TraXP

In Table 9, we find the BOAT BM classification of two of the four TraXP business models (see Section 6.5). This classification gives a clear overview of the basic differences between these two business models. We give a short explanation below.

Business model summary		
BM	***TraXP/F***	***TraXP/X***
Parties	B2C	B2B
Objects	Electronic services	Electronic services
Time Scope	Semi-Dynamic	Static
Drivers	Increasing reach	Increasing richness
Chains	Reintermediation	Reintermediation
Directions	Completely automated business	Enhanced CRM
Structures	Dynamic service outsourcing	Dynamic service outsourcing

Table 9: business model classification of TraXP/F and TraXP/X

The two business models were designed for different customer profiles: /F for individual consumers (hence B2C) and /X for executive business travelers (who travel on corporate budget, hence a B2B classification). For /F, the time scope is dynamic as integration with customers is medium (there is customer profiling, but not very deep). For /X, the time scope is targeted at static through deep customer profiling to obtain customer retention.

The main driver for the /F model is increasing reach: attracting more customers to increase exposure of TraXP in the market. The main driver for the /X model is increasing richness for executive customers. This illustrated in Figure 50.

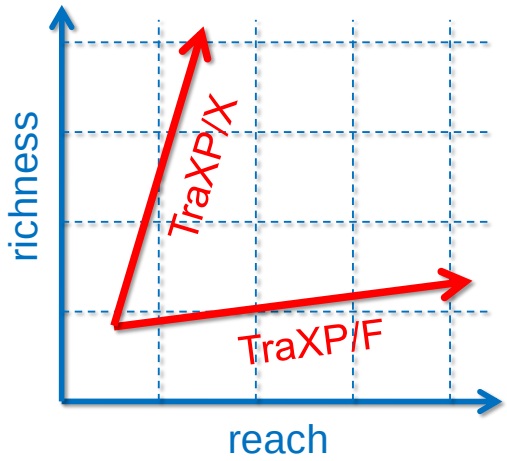

Figure 50: R&R analysis for two TraXP business models (adapted from [Gref10])

Both business models are clear cases of reintermediation: TraXP inserts itself as a new link in existing business chains. In business directions, the /F business model focuses on efficiency through complete automation. The /X models focuses on enhanced customer relationship management (this goes with the increasing of richness). Both business models rely on the dynamic service outsourcing business structure (as also show in their business model radars, illustrated in Figure 27 and Figure 28).

Working Papers Beta 2009 - 2013

nr.	Year	Title	Author(s)
414	2013	*BASE/X.* Business Agility through Cross-Organizational Service Engineering	Paul Grefen, Egon Lüftenegger, Eric van der Linden, Caren Weisleder
413	2013	The Time-Dependent Vehicle Routing Problem with Soft Time Windows and Stochastic Travel Times	Duygu Tas, Nico Dellaert, Tom van Woensel, Ton de Kok
412	2013	Clearing the Sky - Understanding SLA Elements in Cloud Computing	Marco Comuzzi, Guus Jacobs, Paul Grefen
411	2013	Approximations for the waiting time distribution In an *M/G/c* priority queue	A. Al Hanbali, E.M. Alvarez, M.C. van der van der Heijden
410	2013	To co-locate or not? Location decisions and logistics concentration areas	Frank P. van den Heuvel, Karel H. van Donselaar, Rob A.C.M. Broekmeulen, Jan C. Fransoo, Peter W. de Langen
409	2013	The Time-Dependent Pollution-Routing Problem	Anna Franceschetti, Dorothée Honhon, Tom van Woensel, Tolga Bektas, Gilbert Laporte.
408	2013	Scheduling the scheduling task: A time Management perspective on scheduling	J.A. Larco, V. Wiers, J. Fransoo
407	2013	Clustering Clinical Departments for Wards to Achieve a Prespecified Blocking Probability	J. Theresia van Essen, Mark van Houdenhoven, Johann L. Hurink
406	2013	MyPHRMachines: Personal Health Desktops in the Cloud	Pieter Van Gorp, Marco Comuzzi
405	2013	Maximising the Value of Supply Chain Finance	Kasper van der Vliet, Matthew J. Reindorp, Jan C. Fransoo
404	2013	Reaching 50 million nanostores: retail distribution in emerging megacities	Edgar E. Blanco, Jan C. Fransoo
403	2013	A Vehicle Routing Problem with Flexible Time Windows	Duygu Tas, Ola Jabali, Tom van Woensel
402	2013	The Service Dominant Business Model: A Service Focused Conceptualization	Egon Lüftenegger, Marco Comuzzi, Paul Grefen, Caren Weisleder

401	2012	Relationship between freight accessibility and Logistics employment in US counties	Frank P. van den Heuvel, Liliana Rivera, Karel H. van Donselaar, Ad de Jong, Yossi Sheffi, Peter W. de Langen, Jan C. Fransoo
400	2012	A Condition-Based Maintenance Policy for Multi-Component Systems with a High Maintenance Setup Cost	Qiushi Zhu, Hao Peng, Geert-Jan van Houtum
399	2012	A flexible iterative improvement heuristic to Support creation of feasible shift rosters in Self-rostering	E. van der Veen, J.L. Hurink, J.M.J. Schutten, S.T. Uijland
398	2012	Scheduled Service Network Design with Synchronization and Transshipment Constraints For Intermodal Container Transportation Networks	K. Sharypova, T.G. Crainic, T. van Woensel, J.C. Fransoo
397	2012	Destocking, the bullwhip effect, and the credit Crisis: empirical modeling of supply chain Dynamics	Maximiliano Udenio, Jan C. Fransoo, Robert Peels
396	2012	Vehicle routing with restricted loading capacities	J. Gromicho, J.J. van Hoorn, A.L. Kok J.M.J. Schutten
395	2012	Service differentiation through selective lateral transshipments	E.M. Alvarez, M.C. van der Heijden, I.M.H. Vliegen, W.H.M. Zijm
394	2012	A Generalized Simulation Model of an Integrated Emergency Post	Martijn Mes, Manon Bruens
393	2012	Business Process Technology and the Cloud: Defining a Business Process Cloud Platform	Vasil Stoitsev, Paul Grefen
392	2012	Vehicle Routing with Soft Time Windows and Stochastic Travel Times: A Column Generation And Branch-and-Price Solution Approach	D. Tas, M. Gendreau, N. Dellaert, T. van Woensel, A.G. de Kok
391	2012	Improve OR-Schedule to Reduce Number of Required Beds	J.T. v. Essen, J.M. Bosch, E.W. Hans, M. v. Houdenhoven, J.L. Hurink
390	2012	How does development lead time affect performance over the ramp-up lifecycle?	Andres Pufall, Jan C. Fransoo, Ad de Jong

389	2012	Evidence from the consumer electronics industry	Andreas Pufall, Jan C. Fransoo, Ad de Jong, Ton de Kok
388	2012	The Impact of Product Complexity on Ramp-Up Performance	Frank P.v.d. Heuvel, Peter W.de Langen, Karel H. v. Donselaar, Jan C. Fransoo
387	2012	Co-location synergies: specialized versus diverse logistics concentration areas	Frank P.v.d. Heuvel, Peter W.de Langen, Karel H. v.Donselaar, Jan C. Fransoo
386	2012	Proximity matters: Synergies through co-location of logistics establishments	Frank P. v.d.Heuvel, Peter W.de Langen, Karel H.v. Donselaar, Jan C. Fransoo
385	2012	Spatial concentration and location dynamics in logistics:the case of a Dutch province	Zhiqiang Yan, Remco Dijkman, Paul Grefen
384	2012	FNet: An Index for Advanced Business Process Querying	W.R. Dalinghaus, P.M.E. Van Gorp
383	2012	Defining Various Pathway Terms	Egon Lüftenegger, Paul Grefen, Caren Weisleder
382	2012	The Service Dominant Strategy Canvas: Defining and Visualizing a Service Dominant Strategy through the Traditional Strategic Lens	Stefano Fazi, Tom van Woensel, Jan C. Fransoo
381	2012	A Stochastic Variable Size Bin Packing Problem With Time Constraints	K. Sharypova, T. van Woensel, J.C. Fransoo
380	2012	Coordination and Analysis of Barge Container Hinterland Networks	Frank P. van den Heuvel, Peter W. de Langen, Karel H. van Donselaar, Jan C. Fransoo
379	2012	Proximity matters: Synergies through co-location of logistics establishments	Heidi Romero, Remco Dijkman, Paul Grefen, Arjan van Weele
378	2012	A literature review in process harmonization: a conceptual framework	S.W.A. Haneya, J.M.J. Schutten, P.C. Schuur, W.H.M. Zijm
377	2012	A Generic Material Flow Control Model for Two Different Industries	H.G.H. Tiemessen, M. Fleischmann, G.J. van Houtum, J.A.E.E. van Nunen, E. Pratsini

376	2012	Dynamic demand fulfillment in spare parts networks with multiple customer classes	K. Fikse, S.W.A. Haneyah, J.M.J. Schutten
375	2012	Improving the performance of sorter systems by scheduling inbound containers	Albert Douma, Martijn Mes
374	2012	Strategies for dynamic appointment making by container terminals	Pieter van Gorp, Marco Comuzzi
373	2012	MyPHRMachines: Lifelong Personal Health Records in the Cloud	E.M. Alvarez, M.C. van der Heijden, W.H.M. Zijm
372	2012	Service differentiation in spare parts supply through dedicated stocks	Frank Karsten, Rob Basten
371	2012	Spare parts inventory pooling: how to share the benefits	X.Lin, R.J.I. Basten, A.A. Kranenburg, G.J. van Houtum
370	2012	Condition based spare parts supply	Martijn Mes
369	2012	Using Simulation to Assess the Opportunities of Dynamic Waste Collection	J. Arts, S.D. Flapper, K. Vernooij
368	2011	Aggregate overhaul and supply chain planning for rotables	J.T. van Essen, J.L. Hurink, W. Hartholt, B.J. van den Akker
367	2011	Operating Room Rescheduling	Kristel M.R. Hoen, Tarkan Tan, Jan C. Fransoo, Geert-Jan van Houtum
366	2011	Switching Transport Modes to Meet Voluntary Carbon Emission Targets	Elisa Alvarez, Matthieu van der Heijden
365	2011	On two-echelon inventory systems with Poisson demand and lost sales	J.T. van Essen, E.W. Hans, J.L. Hurink, A. Oversberg
364	2011	Minimizing the Waiting Time for Emergency Surgery	Duygu Tas, Nico Dellaert, Tom van Woensel, Ton de Kok
363	2011	Vehicle Routing Problem with Stochastic Travel Times Including Soft Time Windows and Service Costs	Erhun Özkan, Geert-Jan van Houtum, Yasemin Serin

362	2011	A New Approximate Evaluation Method for Two-Echelon Inventory Systems with Emergency Shipments	Said Dabia, El-Ghazali Talbi, Tom Van Woensel, Ton de Kok
361	2011	Approximating Multi-Objective Time-Dependent Optimization Problems	Said Dabia, Stefan Röpke, Tom Van Woensel, Ton de Kok
360	2011	Branch and Cut and Price for the Time Dependent Vehicle Routing Problem with Time Window	A.G. Karaarslan, G.P. Kiesmüller, A.G. de Kok
359	2011	Analysis of an Assemble-to-Order System with Different Review Periods	Ahmad Al Hanbali, Matthieu van der Heijden
358	2011	Interval Availability Analysis of a Two-Echelon, Multi-Item System	Felipe Caro, Charles J. Corbett, Tarkan Tan, Rob Zuidwijk
357	2011	Carbon-Optimal and Carbon-Neutral Supply Chains	Sameh Haneyah, Henk Zijm, Marco Schutten, Peter Schuur
356	2011	Generic Planning and Control of Automated Material Handling Systems: Practical Requirements Versus Existing Theory	M. van der Heijden, B. Iskandar
355	2011	Last time buy decisions for products sold under warranty	Frank P. van den Heuvel, Peter W. de Langen, Karel H. van Donselaar, Jan C. Fransoo
354	2011	Spatial concentration and location dynamics in logistics: the case of a Dutch provence	Frank P. van den Heuvel, Peter W. de Langen, Karel H. van Donselaar, Jan C. Fransoo
353	2011	Identification of Employment Concentration Areas	Pieter van Gorp, Remco Dijkman
352	2011	BOMN 2.0 Execution Semantics Formalized as Graph Rewrite Rules: extended version	Frank Karsten, Marco Slikker, Geert-Jan van Houtum
351	2011	Resource pooling and cost allocation among independent service providers	E. Lüftenegger, S. Angelov, P. Grefen
350	2011	A Framework for Business Innovation Directions	Remco Dijkman, Irene Vanderfeesten, Hajo A. Reijers
349	2011	The Road to a Business Process Architecture: An Overview of Approaches and their Use	K.M.R. Hoen, T. Tan, J.C. Fransoo G.J. van Houtum
348	2011	Effect of carbon emission regulations on transport mode selection under stochastic demand	Murat Firat, Cor Hurkens

347	2011	An improved MIP-based combinatorial approach for a multi-skill workforce scheduling problem	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
346	2011	An approximate approach for the joint problem of level of repair analysis and spare parts stocking	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
345	2011	Joint optimization of level of repair analysis and spare parts stocks	Ton G. de Kok
344	2011	Inventory control with manufacturing lead time flexibility	Frank Karsten, Marco Slikker, Geert-Jan van Houtum
343	2011	Analysis of resource pooling games via a new extenstion of the Erlang loss function	Murat Firat, C.A.J. Hurkens, Gerhard J. Woeginger
342	2011	Vehicle refueling with limited resources	Bilge Atasoy, Refik Güllü, TarkanTan
341	2010	Optimal Inventory Policies with Non-stationary Supply Disruptions and Advance Supply Information	Kurtulus Baris Öner, Alan Scheller-Wolf Geert-Jan van Houtum
339	2010	Redundancy Optimization for Critical Components in High-Availability Capital Goods	Joachim Arts, Gudrun Kiesmüller
338	2010	Analysis of a two-echelon inventory system with two supply modes	Murat Firat, Gerhard J. Woeginger
335	2010	Analysis of the dial-a-ride problem of Hunsaker and Savelsbergh	Murat Firat, Cor Hurkens
334	2010	Attaining stability in multi-skill workforce scheduling	A.J.M.M. Weijters, J.T.S. Ribeiro
333	2010	Flexible Heuristics Miner (FHM)	P.T. Vanberkel, R.J. Boucherie, E.W. Hans, J.L. Hurink, W.A.M. van Lent, W.H. van Harten
332	2010	An exact approach for relating recovering surgical patient workload to the master surgical schedule	Peter T. Vanberkel, Richard J. Boucherie, Erwin W. Hans, Johann L. Hurink, Nelly Litvak
331	2010	Efficiency evaluation for pooling resources in health care	M.M. Jansen, A.G. de Kok, I.J.B.F. Adan

330	2010	The Effect of Workload Constraints in Mathematical Programming Models for Production Planning	Christian Howard, Ingrid Reijnen, Johan Marklund, Tarkan Tan
329	2010	Using pipeline information in a multi-echelon spare parts inventory system	H.G.H. Tiemessen, G.J. van Houtum
328	2010	Reducing costs of repairable spare parts supply systems via dynamic scheduling	F.P. van den Heuvel, P.W. de Langen, K.H. van Donselaar, J.C. Fransoo
327	2010	Identification of Employment Concentration and Specialization Areas: Theory and Application	Murat Firat, Cor Hurkens
326	2010	A combinatorial approach to multi-skill workforce scheduling	Murat Firat, Cor Hurkens, Alexandre Laugier
325	2010	Stability in multi-skill workforce scheduling	M.A. Driessen, J.J. Arts, G.J. v. Houtum, W.D. Rustenburg, B. Huisman
324	2010	Maintenance spare parts planning and control: A framework for control and agenda for future research	R.J.I. Basten, G.J. van Houtum
323	2010	Near-optimal heuristics to set base stock levels in a two-echelon distribution network	M.C. van der Heijden, E.M. Alvarez, J.M.J. Schutten
322	2010	Inventory reduction in spare part networks by selective throughput time reduction	E.M. Alvarez, M.C. van der Heijden, W.H. Zijm
321	2010	The selective use of emergency shipments for service-contract differentiation	B. Walrave, K. v. Oorschot, A.G.L. Romme
320	2010	Heuristics for Multi-Item Two-Echelon Spare Parts Inventory Control Problem with Batch Ordering in the Central Warehouse	Nico Dellaert, Jully Jeunet.
319	2010	Preventing or escaping the suppression mechanism: intervention conditions	R. Seguel, R. Eshuis, P. Grefen.
318	2010	Hospital admission planning to optimize major resources utilization under uncertainty	Tom Van Woensel, Marshall L. Fisher, Jan C. Fransoo.

317	2010	Minimal Protocol Adaptors for Interacting Services	Lydie P.M. Smets, Geert-Jan van Houtum, Fred Langerak.
316	2010	Teaching Retail Operations in Business and Engineering Schools	Pieter van Gorp, Rik Eshuis.
315	2010	Design for Availability: Creating Value for Manufacturers and Customers	Bob Walrave, Kim E. van Oorschot, A. Georges L. Romme
314	2010	Transforming Process Models: executable rewrite rules versus a formalized Java program	S. Dabia, T. van Woensel, A.G. de Kok
313	2010	Getting trapped in the suppression of exploration: A simulation model	
312	2010	A Dynamic Programming Approach to Multi-Objective Time-Dependent Capacitated Single Vehicle Routing Problems with Time Windows Tales of a So(u)rcerer: Optimal Sourcing Decisions Under Alternative Capacitated Suppliers and General Cost Structures	Osman Alp, Tarkan Tan
311	2010	In-store replenishment procedures for perishable inventory in a retail environment with handling costs and storage constraints	R.A.C.M. Broekmeulen, C.H.M. Bakx
310	2010	The state of the art of innovation-driven business models in the financial services industry	E. Lüftenegger, S. Angelov, E. van der Linden, P. Grefen
309	2010	Design of Complex Architectures Using a Three Dimension Approach: the CrossWork Case	R. Seguel, P. Grefen, R. Eshuis
308	2010	Effect of carbon emission regulations on transport mode selection in supply chains	K.M.R. Hoen, T. Tan, J.C. Fransoo, G.J. van Houtum
307	2010	Interaction between intelligent agent strategies for real-time transportation planning	Martijn Mes, Matthieu van der Heijden, Peter Schuur
306	2010	Internal Slackening Scoring Methods	Marco Slikker, Peter Borm, René van den Brink
305	2010	Vehicle Routing with Traffic Congestion and Drivers' Driving and Working Rules	A.L. Kok, E.W. Hans, J.M.J. Schutten, W.H.M. Zijm
304	2010	Practical extensions to the level of repair analysis	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
303	2010	Ocean Container Transport: An Underestimated and Critical Link in Global Supply Chain Performance	Jan C. Fransoo, Chung-Yee Lee
302	2010	Capacity reservation and utilization for a manufacturer with uncertain capacity and demand	Y. Boulaksil; J.C. Fransoo; T. Tan
300	2009	Spare parts inventory pooling games	F.J.P. Karsten; M. Slikker; G.J. van Houtum
299	2009	Capacity flexibility allocation in an outsourced supply chain with reservation	Y. Boulaksil, M. Grunow, J.C. Fransoo

298	2010	An optimal approach for the joint problem of level of repair analysis and spare parts stocking	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
297	2009	Responding to the Lehman Wave: Sales Forecasting and Supply Management during the Credit Crisis	Robert Peels, Maximiliano Udenio, Jan C. Fransoo, Marcel Wolfs, Tom Hendrikx
296	2009	An exact approach for relating recovering surgical patient workload to the master surgical schedule	Peter T. Vanberkel, Richard J. Boucherie, Erwin W. Hans, Johann L. Hurink, Wineke A.M. van Lent, Wim H. van Harten
295	2009	An iterative method for the simultaneous optimization of repair decisions and spare parts stocks	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
294	2009	Fujaba hits the Wall(-e)	Pieter van Gorp, Ruben Jubeh, Bernhard Grusie, Anne Keller
293	2009	Implementation of a Healthcare Process in Four Different Workflow Systems	R.S. Mans, W.M.P. van der Aalst, N.C. Russell, P.J.M. Bakker
292	2009	Business Process Model Repositories - Framework and Survey	Zhiqiang Yan, Remco Dijkman, Paul Grefen
291	2009	Efficient Optimization of the Dual-Index Policy Using Markov Chains	Joachim Arts, Marcel van Vuuren, Gudrun Kiesmuller
290	2009	Hierarchical Knowledge-Gradient for Sequential Sampling	Martijn R.K. Mes; Warren B. Powell; Peter I. Frazier
289	2009	Analyzing combined vehicle routing and break scheduling from a distributed decision making perspective	C.M. Meyer; A.L. Kok; H. Kopfer; J.M.J. Schutten
288	2009	Anticipation of lead time performance in Supply Chain Operations Planning	Michiel Jansen; Ton G. de Kok; Jan C. Fransoo
287	2009	Inventory Models with Lateral Transshipments: A Review	Colin Paterson; Gudrun Kiesmuller; Ruud Teunter; Kevin Glazebrook
286	2009	Efficiency evaluation for pooling resources in health care	P.T. Vanberkel; R.J. Boucherie; E.W. Hans; J.L. Hurink; N. Litvak
285	2009	A Survey of Health Care Models that Encompass Multiple Departments	P.T. Vanberkel; R.J. Boucherie; E.W. Hans; J.L. Hurink; N. Litvak
284	2009	Supporting Process Control in Business Collaborations	S. Angelov; K. Vidyasankar; J. Vonk; P. Grefen
283	2009	Inventory Control with Partial Batch Ordering	O. Alp; W.T. Huh; T. Tan
282	2009	Translating Safe Petri Nets to Statecharts in a Structure-Preserving Way	R. Eshuis
281	2009	The link between product data model and process model	J.J.C.L. Vogelaar; H.A. Reijers
280	2009	Inventory planning for spare parts networks with delivery time requirements	I.C. Reijnen; T. Tan; G.J. van Houtum
279	2009	Co-Evolution of Demand and Supply under Competition	B. Vermeulen; A.G. de Kok
			B. Vermeulen, A.G. de Kok

278	2010	Toward Meso-level Product-Market Network Indices for Strategic Product Selection and (Re)Design Guidelines over the Product Life-Cycle	R. Seguel, R. Eshuis, P. Grefen
277	2009	An Efficient Method to Construct Minimal Protocol Adaptors	
276	2009	Coordinating Supply Chains: a Bilevel Programming Approach	Ton G. de Kok, Gabriella Muratore
275	2009	Inventory redistribution for fashion products under demand parameter update	G.P. Kiesmuller, S. Minner
274	2009	Comparing Markov chains: Combining aggregation and precedence relations applied to sets of states	A. Busic, I.M.H. Vliegen, A. Scheller-Wolf
273	2009	Separate tools or tool kits: an exploratory study of engineers' preferences	I.M.H. Vliegen, P.A.M. Kleingeld, G.J. van Houtum
272	2009	An Exact Solution Procedure for Multi-Item Two-Echelon Spare Parts Inventory Control Problem with Batch Ordering	Engin Topan, Z. Pelin Bayindir, Tarkan Tan
271	2009	Distributed Decision Making in Combined Vehicle Routing and Break Scheduling	C.M. Meyer, H. Kopfer, A.L. Kok, M. Schutten
270	2009	Dynamic Programming Algorithm for the Vehicle Routing Problem with Time Windows and EC Social Legislation	A.L. Kok, C.M. Meyer, H. Kopfer, J.M.J. Schutten
269	2009	Similarity of Business Process Models: Metics and Evaluation	Remco Dijkman, Marlon Dumas, Boudewijn van Dongen, Reina Kaarik, Jan Mendling
267	2009	Vehicle routing under time-dependent travel times: the impact of congestion avoidance	A.L. Kok, E.W. Hans, J.M.J. Schutten
266	2009	Restricted dynamic programming: a flexible framework for solving realistic VRPs	J. Gromicho; J.J. van Hoorn; A.L. Kok; J.M.J. Schutten;

Working Papers published before 2009 see: http://beta.ieis.tue.nl