Value Proposition Modelling for Service Innovation

Category:
Research paper

Purpose:
Insufficient or ambiguous description of services can lead to problems during service design and innovation. To lower the costs associated with such inefficiencies, we propose to utilize a semantics-oriented conceptual description of services which is built upon a semi-formal view of value proposition. This semi-formalization enables to understand and model more accurately the value proposition itself, its role in a service system, and the expectations of the service client.

Design/methodology/approach:
First, the semi-formalization of value proposition, along with the related semi-formalization of the concepts 'service system' and 'service' is recalled. This semi-formalization blends together the Service Science perspective with specific semantics-oriented conceptual modelling technique.

The main contribution of this paper is the proposal of the utilization of the semi-formalized value proposition to further support value proposition-driven service innovation. The proposed approach is illustrated on examples from telecommunication services billing domain.

Findings:
The introduced utilization of the semi-formalized value proposition results in the so-called 'conceptual perspective' which can be employed in the service design and specification process. This perspective allows for uniform modelling of the value proposition on the one hand, and the service client expectation on the other.

Research implications:
This paper presents an application of the semi-formalized view of service systems and illustrates the proposed method of modelling the value proposition semantics.

Practical implications:
The proposed 'conceptual perspective' can be utilized in service design and specification process to facilitate the work with the value proposition, for instance during service portfolio management.

Originality/value:
This paper presents an original contribution to the possible ways of modelling the value proposition. The found results are being developed in accordance with the implementation of a concrete system.

Key words: value proposition, service systems, service innovation, conceptual modelling, SSMED

Introduction

The importance of services and their systematic study is nowadays well recognized in the multidisciplinary research field called *service science* (Alter, 2008), (Spohrer et al., 2007). The perspective of service science has already impacted thinking about services in *computer science* as well (Katzan, 2009), (Qiu, 2009), (Staníček and Winkler, 2010), (Winkler and Dosoudil, 2011). This paper aims to continue in the computational thinking of services by proposing the so-called conceptual perspective of services, modelling the information which the services provide and need for their execution. The conceptual perspective is a practically usable application of the ideas of semi-formalized value proposition (Winkler and Dosoudil, 2011) and the semantic decomposition of services (Staníček and Winkler, 2010) published in our previous work. We believe that our proposed approach can help better describe and understand customer's needs, map those needs onto existing services in order to support their innovation, and improve design of new services.

We can meet several problems when trying to perform a service for our customer benefit and/or to build up our business on such services. One of them is to build a new service by assembling existing services possibly amended by a new part or component service. This is necessary to reach a similar effect as economy of scale in goods selling business.

Service provision is an interaction with a service client. The more clients the more interactions; this leads to linear cost increase with a number of clients. What can help us to avoid this scenario? Our opinion is we have two possibilities:

- technology, and
- organization.

A combination of these two possibilities is the approach of service system building. The first rule is not to build a new service from scratch. Contrarily, it is necessary to find an appropriate subset in the set of existing services and to build up the desired service by assembling services from this subset possibly glued by a newly completed service or its component. The challenge is to grow up a set of prime services from which it is easy to assemble a required service and particularly to enrich this set by such new items which will be usable in various future situations. The client is served on a higher quality this way as the majority of assembled services is validated by previous practice. The time to client's benefits is shorter as assembling known components is faster than building a new service from scratch.

On the other hand, the decomposition of a service into subservices is not a question of finding elementary and not more decomposable services: such a detailed decomposition is not comprehensive for a customer, thus it is non-beneficial. The aim is to find the proper subservices from a pragmatic point of view. A service explication by means of these subservices has to be understandable for key stakeholders and the subservices have to be usable repeatedly in other cases.

Value proposition is recognized as a proper means of explicating a service to various stakeholders. This implies the need to work with value proposition when processing a decomposition or a composition of services. Hence, value propositions have to be formulated more formally and/or precisely than it is done currently by means of natural language sentences. On the other hand, the formalization and precision must be balanced with comprehensibility and practical usability. We believe the answer to these requirements comes from well handled semantics.

Our approach is based on 30-year experience with semantics modelling using a functional approach based on Transparent Intensional Logic. A method of conceptual modelling—called HIT method (Duží, 2002), (Duží et al., 1986) (Staníček, 2009)—was developed in eighties of the last century and successfully proved in a hundred of projects. A common opinion that semantics can be handled in computers only through appropriately chosen syntax of artificial languages was broken. Instead, a kind of Mention-Use logic was established and proven: things we do (= Use) in real world have to be appropriately and non-ambiguously described (= Mention) to be grasped in minds, elaborated, combined and again put into the process of utilization (Use).

The paper is organized as follows: after a brief discussion of related work, we recall some of the background concepts and notation used in semantics-oriented conceptual modelling method HIT (Duží, 2002), (Duží et al., 1986). Then we mention the semi-formalized value proposition and its modelling by defining a set of service *characteristics*. The next section introduces the proposed conceptual perspective on the case study describing an existing service providing advanced billing of telecommunication services. The paper concludes with our planned future work.

Related Work

We build our perception of *service* and *service system* on the work of (Vargo and Lusch, 2004), (Maglio and Spohrer, 2008), and (Maglio et al., 2006). In addition, the semi-formalized value proposition requires more technical definitions of these two concepts, and therefore we adopt these technical definitions from (Winkler and Dosoudil, 2011).

This paper aims to continue in the computational thinking of services by further elaborating on the ideas published in our previous work, particularly the semi-formalization of the term value proposition (Winkler and Dosoudil, 2011), the semantic decomposition of services (the initial idea was published under the term *Service Breakdown Structure* in (Staníček and Winkler, 2010)), and utilization of semantics-oriented conceptual modelling methods for the purpose of modelling services and service systems (Staníček and Winkler, 2010), (Winkler and Dosoudil, 2011).

We mentioned the semi-formalized value proposition. What is usually understood by a value proposition? This term was introduced by Michael Lanning (Lanning, 1998):

> *"The combination of resulting experiences, including price, which an organization delivers to a group of intended customers in some time frame, in return for those customers buying/using and otherwise doing what the organization wants rather than taking some competing alternative."*

However, the original meaning of value proposition has shifted, and nowadays the value proposition often refers to *the attributes of offered service/goods* instead of *delivered experiences*. Both interpretations are manifested in terms of a *value proposition* and an *acceptable value proposition* defined in (Winkler and Dosoudil, 2011).

In the field of marketing there are resources (such as (Barnes et al., 2009)) that offer a comprehensive guide on creating and delivering value propositions. (Osterwalder and Pigneur, 2003) present an elaborated conceptual approach to modelling value propositions.

Our approach is oriented more computationally: we intend to connect a value proposition to the conceptual view of service (or services) realizing this value proposition. What do we mean by talking about *conceptual* perspective and modelling *the information which the services provide and need for their execution*? We do not attempt to develop another formal description of operations, i.e. processes or *services*, running in a service system; we try to describe the qualities of these services and propose a semi-formal method of mapping these service qualities onto client's needs. Such mapping can be used to identify what service qualities are demanded, while the semi-formal description of service qualities can be used to improve service provider's knowledge of available services. A semi-formal description is usually comprehensible quite easily for a businessperson or a domain expert while providing pragmatically chosen level of structuring the information contained within the description.

Semantic Modelling for Services

Conceptual modeling can be described as the process of conceptual understanding and rendering the modeled part of reality. The modeler tries to find out which objects of interest the system should keep and provide. Conceptual model should be totally independent of intended implementation and intelligible for users (Duží, 2001). We use specific semantics-oriented approach to conceptual modelling called HIT data model (Zlatuška, 1986), (Duží et al., 1987). Although the benefit of models containing semantics lies above all in the presence of definitions of modeled concepts, the better the quality of definitions, the more likely readers of the model are able to understand correctly the modeled domain. Here the HIT data model comes in, offering method for writing quality concept definitions, thus improving the techniques of conceptualization.

Exactness of the proposed way of conceptualization lies in its anchoring in natural language expressions analysis using a functional approach of Transparent Intensional Logic. Almost everything is a function over epistemic base formed by four sets: a set of possible worlds, a set of time moments, a set of individuals, and a set of truth values (Tichý, 88), (Duží et al., 2010). Thus our main construct will be a function. We distinguish analytical and empirical functions. Analytical are functions assigning a value to a given argument values independently of a state of world (which is simply a pair < possible world, time moment>). Analytical functions are commonly used and we deal with them using math and logic apparatus or using algorithms. The empirical function assigns a value to a given argument values dependently of a state of world; this means a function value cannot be computed from the arguments values by an algorithm but it is evaluated by processing data which records a particular state of world. The functions we use for state of affairs explication are of the following types:

(a) $W \rightarrow ((T_1, ..., T_n) \rightarrow (S_1, ..., S_m))$, or
(b) $W \rightarrow ((T_1, ..., T_n) \rightarrow ((S_1, ..., S_m) \rightarrow Bool))$,

where W denotes the set of all states of world, T_i, S_j, denote entity or descriptive sets (see below), and Bool = {True, False}.

When creating a database schema, a visualization or a conceptual model, we need to determine the nodes of this schema/model which stand for some basic classes of objects of our interest. When creating a process model in a business area, we need to determine nodes of this model which stand for elementary processes, events, branching and joining blocks. These are again basic classes of objects of our interest in this case. When creating a service model or a service system model, we again need to determine nodes of this model which stand for particular agents (clients, providers, possibly targets of a service), functions, processes, events, reports, value propositions, and other services or service systems. These again are basic classes of objects of our interest in this case. The variety of such classes is very big one. All the mentioned classes are usually called entity sets within a course of conceptual modelling.

It is very practical to distinguish between entity sets and descriptive sets. Entity sets contain objects of a real world that cannot be stored in a computer directly. We will denote them in a form (#Name of set). Descriptive sets contain strings of symbols and/or numbers that can be stored in a computer directly. We will denote them in a form (Name of set). Thus the symbol "#" helps us to distinguish between entity and descriptive sets at the first sight.

Each entity set must be precisely defined in a form which enables to decide whether a given item belongs to the set or not. Such definitions are based on properties which itself are functions assigning to each state of world $w \in W$ the set of all items possessing this property. The entity set definition can have the following form:

An object of category (#Name of set) is any object having the property P.

According to the Diamond of Attention Focusing (see Figure 1) introduced in (Procházka, Staníček, 2004) and explained in (Staníček, Winkler, 2010) we can speak of sets as of categories. Entity set definitions are then nothing more than definitions of appropriate categories to which particular objects are assigned using respective connection. Examples of entity sets definitions follow in next chapters.

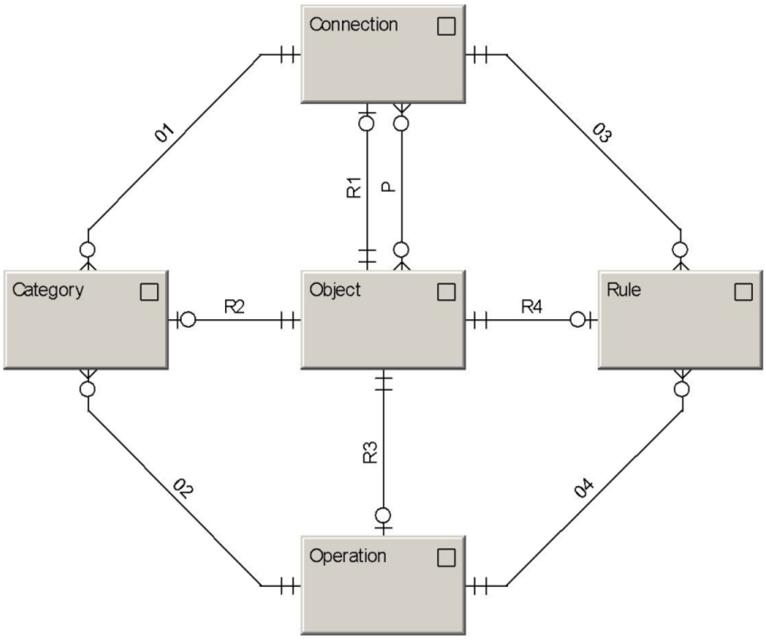

Figure 1: Diamond of Attention Focusing

Relations between the individual elements of the sets are described by the so-called HIT-attributes. HIT-attributes are empirical functions of type (a) or (b), see above. Such function can either map a Cartesian product of sets into another Cartesian product of sets or into the set of all subsets of the other Cartesian product. The sets can be both entity sets and descriptive sets; at least one of these sets must be entity set. If a HIT-attribute maps into Cartesian product, we call it a singular attribute, if it maps into the set of all subsets of the Cartesian product we call it a plural attribute.

As HIT-attributes are empirical functions they depend on a state of the world. Using lambda-terms such a HIT-attribute can be expressed as

$\lambda w \lambda x_1 ... x_n \iota y_1 ... y_m$ (C) or $\lambda w \lambda x_1 ... x_n \lambda\ y_1 ... y_m$ (C),

where w ranging over all possible states of the world, x_i, y_j ranging over respective (entity or descriptive) set T_i, S_j, and C is boolean expression containing as free variables w, x_i, i=1, .., n, y_j,
j=1, .., m, only. This form of HIT-attribute expression is convenient for the purpose of general statements formulation and their proofs. To work with HIT-attributes in a given domain we use standardized semantic names of these attributes. Such standardized name for a singular attribute is

$text_0$ (name_of_set S) $text_1$ (name_of_set T_1) $text_2$ (name_of_set T_2) ...
... $text_n$ (name_of_set T_n)$text_{n+1}$

where $S = S_1 \times ... \times S_m$, only $text_0$ and $text_{n+1}$ can be omitted as they serve for more natural expression only, $text_i$, i = 1 .. n, expresses a role of particular argument, and the whole natural language expression read in natural way gives us a notion of function. Similarly standardized name for a plural attribute is

$text_0$ (name_of_set S)-s $text_1$ (name_of_set T_1) $text_2$ (name_of_set T_2) ...
... $text_n$ (name_of_set T_n)$text_{n+1}$

The element “-s“ expresses the plural, i.e. mapping into set of all subsets of S.

The relations of individual objects dwell in the container Connection of the Diamond of Attention Focussing. The full semantic names of these relations denote particular categories of connections from the Category container.

Let’s understand that an entity set can be any grouping of any objects from the container Object in Figure 1. Due to the recursive nature of the Diamond of Attention Focusing (each connection, operation, category, or rule can be conceived as an object when we intend to mention it) we obtain a very powerful and expressive sub-language of the natural language, which is moreover very exact and not ambiguous.

Examples of HIT-attribute full semantic names follow in next chapters.

The last construct used to describe a part of reality is a construct of consistency constraint. A consistency constraint is a proposition, i.e. a function of type ($W \rightarrow Bool$) giving True in those states of world in which the constraint is fulfilled, constructed over a set of HIT-attributes. Again, examples will be found in next chapters. Consistency constraints can be found in the container *Rule* in Figure 1.

A crucial point of our semantics modelling approach is distinguishing analytical and empirical issues. The above explained concepts belong to the empirical world (what is given by a particular function depends on a state of the world). In fact, this dependence means that a function is in each state of the world *w* given as a table of arguments values and function values. In case of HIT-attributes we call such tables defining tables of these attributes in a particular state of world. Consistency constrains are then simply the constraints of selected values in those defining tables.

To be able to work with such constructs and to build something reasonable from them we need to employ analytical functions. If [AB] denotes an application of a function A to an argument value B we can define:

A HIT-attribute A is definable over a set of HIT-attributes $\{B_1, ..., B_n\}$ iff there exists an analytical function F (algorithm F), which is not a contradiction and which is not a tautology, such that for all $w \in W$ it holds

$[Aw] = [F([B_1w], ..., [B_nw]\ )]$,

i.e., which computes the A-defining table in a state of world w from the B_1-, ..., B_n-defining tables in the same state of world w. A set of HIT-attributes **M** is definable over a set of HIT-attributes **N** iff each attribute from **M** is definable over some subset of **N**.

Analytical functions helping us to compute something interesting from empirical functions dwell in container *Operation* in the Diamond of Attention Focusing, see Figure 1.

Using definability, we can compare particular sets of HIT-attributes, introduce partial ordering into sets of HIT-attributes, and, mainly, we can support the service decomposition and/or composition using an appropriate expression of their value propositions.

Semi-formalized Value Proposition

The definition of semi-formalized value proposition is built upon several concepts introduced in (Winkler and Dosoudil, 2011). In this paper, we mention only those related directly to the value proposition definition. Their definitions as well as the examples are adopted from (Winkler and Dosoudil, 2011).

For the purpose of the definitions given later in this section, we identify a service *S* by the following formula $S = (c, p, t, vp)$, where *c* is the service client, *p* the service provider, *t* the target of the service, and *vp* the value proposition. Please refer to (Winkler and Dosoudil, 2011) in case you intended to read the complete definition.

We build the semi-formalized value proposition on the concept of a *utility function* already known from economics. We have chosen this method because it combines very naturally with the value proposition definition proposed later in this paper. However, once a value proposition is formulated, other measures for evaluating if a service meets its value proposition can still be used.

The utility function is able to compute the amount of utility of a specific entity for some other entity. In this paper, we apply this general concept in services, and for that reason we use the utility function as a “device” which yields the amount of utility of a given service for a particular service client.

Definition 1 (utility function) *Let us consider a service* $S = (c, p, t, vp)$. *We denote* U_S^k *to be the k-dimensional utility function of the service S for its client:*

$$U_S^k = Q_1 \times \ldots \times Q_k \to R, \text{ where}$$

Q_j *is the set of values of the j-th objective characteristic relevant for the service* $S, j \in N, 1 \le j \le k$,

k *is the number of objective characteristics relevant for the client c,*

N *is the set of natural numbers,*

R *denotes the set of real numbers.*

We use the term the utility function of the service S, denoted by U_S, *for a k-dimensional utility function of the service S for some* $k \in N$.

The utility function computes the amount of utility of the service *S* for the client c according to a given collection of service characteristics. These characteristics can be different for various services and should be defined carefully in co-operation of both the service provider and the client, for example, by means of an appropriate part of a conceptual model.

Examples of objective characteristics of a service include both quantitative properties (performance, reliability, etc.) and qualitative properties (i.e. what the service actually does). We will explain the characteristics and their values in the following section.

The last concept necessary to define the value proposition is the *domain* $D(U_S^k)$ of the utility function U_S^k of a service *S*. We understand the *domain of a function f* as the set of all values *x* such that *f(x)* is defined, therefore $D(U_S^k) \subseteq Q_1 \times \ldots \times Q_k$.

Modelling Characteristics of a Service

Let us now discuss what the service objective characteristics look like and how to model them and their semantics. Taking a car rental service as an example, its basic characteristic would be that this service offers cars for rent to its clients. We can model this situation by using the following simple conceptual model (see Figure 2 for the diagram of the model):

- An object of category **(#Provider)** is any agent (person or company) acting as a service provider.
- An object of category **(#Car model)** is any particular design or version of a car.

- Relationship 1: ***(#Car model)****-s offered for rent by a given* ***(#Provider)****. / 0,M:0,M*

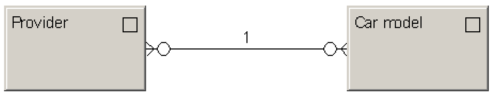

Figure 2: Car rental service — a provider offers cars (of particular car models) for rent (model diagram).

According to the HIT conceptual modelling method, the Relationship 1 is understood as a *function* taking an entity from the entity set **(#Provider)** as the input, and returning those entities from the entity set **(#Car model)**, which are offered for rent by the input argument (provider entity). The expression *p,q:r,s* defines the cardinality of the relationship as follows:

- the $p \in \{0,1\}$ determines if the function is partial or total (in the case of the relationship 1, there can be a provider that offers no car model for rent),
- the $q \in \{1, M\}$ says if the function returns at most one or more results (in the case of the relationship 1, there can be a provider that offers more than one car model for rent),
- the pair *r,s* is the analogy of the *p,q* for the "rotated"[1] function, in the case of the relationship 1 the rotated function is ***(#Provider)****-s offering for rent a given* ***(#Car model)***. The cardinality *0,M* says that zero or more providers that offer for rent a given car model can be returned by the function.

More detailed and rigorous definitions of the HIT method concepts are beyond the scope of this paper and can be found in (Duží, 2002), (Duží et al., 1986) or (Staníček, 2009).

Another characteristic of car rental service might express that the service provider rents a car (of a particular model) to the client. Let us have a look at the model (see Figure 3 for the diagram):

- An object of category **(#Provider)** is any agent (person or company) acting as a service provider.
- An object of category **(#Client)** is any person holding a valid driving license interested in the service.
- An object of category **(#Car model)** is any particular design or version of a car.

- An object of category **(#Rents)** is any representation of relationship among a provider, a client and a car model, with the following meaning:
  (#Client)*-s who have rented a car of given* ***(#Car model)*** *from given* ***(#Provider)****. / 0,M:0,M*

[1] The definition can be found as the term *HIT attribute rotation* in (Duží et al., 1986) or (Staníček, 2009).

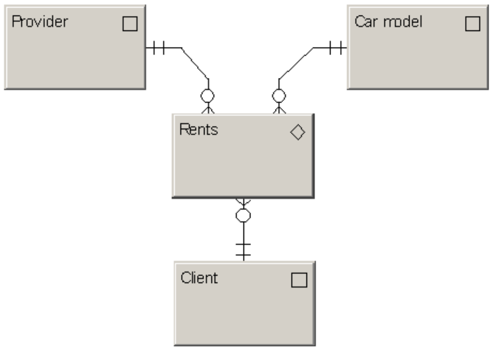

Figure 3: Car rental service – a client rents a car from the provider (model diagram). The rectangles with boxes represent primary entities, while the rectangle with the diamond ((#Rents)) denotes an entity representing a relationship.

Formal definition of service characteristics and their distinction into qualitative and quantitative is beyond the scope of this paper and can be found in (Winkler and Dosoudil, 2011).

Value Proposition definition

We mentioned that a utility function returns the amount of utility of the given service for the specific service client. The service can be (not necessarily completely) described by a set of objective characteristics and their values – the sets Q_j that we defined as inputs for the corresponding utility function. We claim that a collection of the values of such characteristics (which are used for evaluation of the utility of the service) represents a value proposition.

Using previously introduced notions, we can define a general value proposition as follows:

Definition 2 (value proposition) *Consider a service S. Let* $D(U_S)$ *be the domain of the utility function of the service S for its client. Then any* $VP_S \subseteq D(U_S)$ *is a value proposition of S for its client.*

Please consider the following two special cases of VP_S. Both are quite impractical but they can give you further insight into the proposed formalism:

- the case $VP_S = D(U_S)$ means that *VP* proposes *anything* in the domain defined by the characteristics that U_S is applied to,
- the case $VP_S = \varnothing$ represents *VP* which proposes *nothing*.

Although we have defined the value proposition, this definition is very general because, in fact, it allows for any value proposition, even those that no client would accept. This situation is amended by defining the concept of *acceptable value proposition*, which corresponds to a value proposition which is considered useful by a client. The amount of usability, expressed as a real number, is determined by applying the client's utility function to the value proposition. The acceptable value proposition is further described in (Winkler and Dosoudil, 2011).

We are going to illustrate the proposed method of value proposition formalization on the example of a car rental service.

Semi-formalized Value Proposition Example

This section presents an example of concrete value proposition semi-formalization. We define common concept dictionary, then use it for the description of service characteristics, and finally present an example of a value proposition defined over the service characteristics.

Dictionary

We begin with the definition of terms used in the later characterization of the service. Let us define the parts involved in the service first:

- An object of category **(#Provider)** is any agent (person or company) acting as a service provider.
- An object of category **(#Client)** is any person holding a valid driving license interested in the service.

Next, we describe the domain of the service by means of the following conceptual model *M* (see Figure 4 for its diagram):

- An object of category **(#Car model)** is any particular design or version of a car.

- An object of category **(#Circuit)** is any course suitable for car racing.
- An object of category **(#Location)** is any particular place.

- An object of category **(#Rents)** is any representation of relationship among a provider, a client and a car model, with the following meaning:
  ***(#Client)**-s who have rented a car of given **(#Car model)** from given **(#Provider)**. / 0,M:0,M*

- An object of category **(#Allows)** is any representation of relationship among a provider, a circuit and a car model, with the following meaning:
  (#Circuit)-s which can be attended with a car of given **(#Car model)** rented by given **(#Provider)**. */ 0,M:0,M*

- An object of category **(#Drop-off)** is any representation of relationship among a provider, a location and a car model, with the following meaning:
  (#Location)-s where a car of given **(#Car model)** rented by given **(#Provider)** can be dropped off. */ 0,M:0,M*

- Relationship 1: ***(#Car model)**-s offered for rent by given **(#Provider)**. / 0,M:0,M*

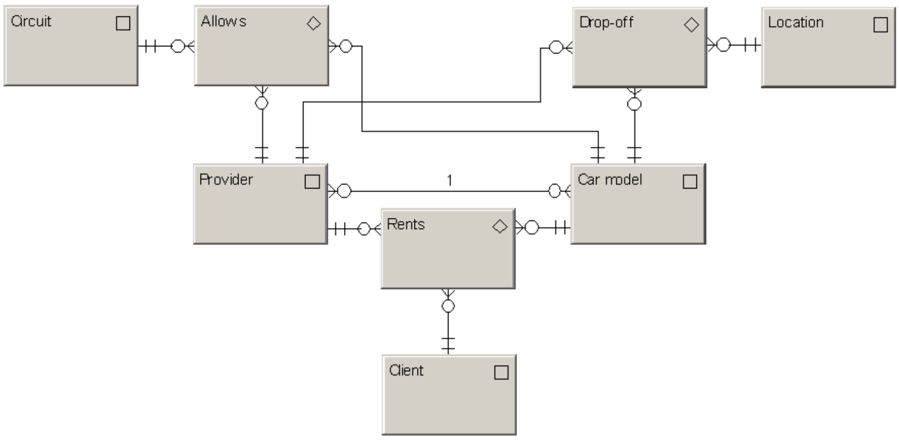

Figure 4: Car rental service – the diagram of the model M

Service Characteristics

Next, we need to define service characteristics important for the intended service client (or clients). It means we probably have to determine these characteristics in co-operation with the client(s).

Let us assume that the provider aims to focus on clients who intend to use the rented car for any of the following purposes:

- travelling (with the starting location same as the drop-off location),
- one-way travelling (with the drop-off location different from the starting location),
- renting a car to experience driving a racing car on a race circuit.

Each purpose may determine several requirements on the proposed service. These requirements indicate which of the possible service characteristics are important for the client when deciding whether or not to rent a particular car from the provider. The characteristics in our example are defined by referencing the corresponding part of the conceptual model *M* introduced above. We have derived the following characteristics:

Qualitative characteristics:

- Q_1: The characterized service is a car rental service, i.e. this characteristic is defined by the whole model *M*.
- Q_2: **(#Circuit)**-s which can be attended with a car of given **(#Car model)** rented by given **(#Provider)**.
- Q_3: **(#Location)**-s where a car of given **(#Car model)** rented by given **(#Provider)** can be dropped off.

You can notice that the characteristic Q_1 is formulated with a consistency constraint while the other two characteristics are defined by HIT attributes.

Quantitative characteristics:

- Q_4 (max speed): maximum **(Speed)** in miles per hour achievable by a car of given **(#Car model)**.
- Q_5 (rental rate): total rental **(Rate)** in USD, including taxes, of a car of given **(#Car model)** for one day requested by given **(#Provider)**.

For the purpose of a value proposition, the values of qualitative characteristics are defined as pairs *(b, c)*, where $b \in Bool$ is 1 if the service described by the value proposition conforms to the model defining the characteristic, and 0 otherwise; *c* is an optional set of additional consistency constraints on the population of the model. The more detailed description can be found in (Winkler and Dosoudil, 2011); this paper explains this topic on the example of a value proposition in the next paragraph.

Value Proposition

The formalized value proposition of the Car Rental service is based on the aforementioned characteristics, more precisely:

$$VP_S \subseteq Q_1 \times Q_2 \times Q_3 \times Q_4 \times Q_5$$

The value proposition is, for instance, the following (in case there are no model population constraints for a given characteristic, we omit them for better readability):

$$VP_S = \left\{ \begin{pmatrix} 1 \\ 0 \\ (1, C_1) \\ 90 \\ 45 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ (1, C_1) \\ 80 \\ 40 \end{pmatrix}, \begin{pmatrix} 1 \\ (1, C_2) \\ 0 \\ 120 \\ 399 \end{pmatrix} \right\}, \qquad \text{where}$$

C_1 = (**(#Location)** = {any city in Georgia, U.S.})

C_2 = (**(#Circuit)** = {Road Atlanta})

How do we interpret this value proposition? VP_S offers three variants of the service:

- renting a car which can be dropped off at different than pick-up location, more specifically at any city in Georgia, U.S., with maximum speed of 90 mph for $45 a day,
- the same as the previous variant, except the maximum speed is 80 mph and the rental rate $40 a day,
- renting a race car with allowed access to Road Atlanta road course with maximum speed of 120 mph for $399 a day.

The value proposition semi-formalization is explained in more detail, as well as an example of determining acceptability of a value proposition by a client, in (Winkler and Dosoudil, 2011).

Case Study – Advanced Billing of Telecommunication Services

The semi-formalized value proposition described in the previous section may look rather complicated for any practical utilization. The goal of the section which you are reading right now is to persuade you that the practical utilization is possible, and even brings benefit.

The practical utilization of the semi-formalized value proposition and the semantics-oriented conceptual modelling method HIT is the *conceptual perspective* already mentioned in the introduction. This perspective uses service characteristics to define value proposition of the service and to decompose the service according to the information that is required by the service to deliver its value proposition. The service characteristics have to be defined by the appropriate conceptual model.

This section illustrates the conceptual perspective on an existing service called Advanced Billing of Telecommunication Services (ABTS). The service combines the information provided by the service client with billing data produced by client's telecommunication services provider. As the result, the client receives billing reports enhanced with information enabling the client, for instance, to optimize their costs.

The problem statement

As we mentioned, the ABTS service provides its client with a set of billing reports helping the client to achieve cost and revenue optimization, for instance by providing reports for departmental and employee chargeback. The service is delivered by a trained operator using specialized software application automating as many tasks as possible.

During the development of software support for this service, the service was being transformed from the pure “manufacture“, performed by several operators, into a more optimized form of one operator using specialized software application. The ABTS manager faced, as far as we know, a very common issue: the whole process had been very underspecified, the information how to create the reports for current customers was present only in the “minds“ of several operators of the original service. The cooperation with them was rather demanding, as they had no up-to-date specification, their answers were often incomplete or changed.

We are sure that if the original service had been described by means of a conceptual perspective view, which will be presented later in this section, the costs for the development of optimized service would have been lowered significantly.

Another issue is even more interesting because it occurs each time the service is being set up for a new client. The ABTS project manager has to deal with the following situation:

- the service is already capable of producing reports which are already being delivered to existing clients; we can imagine that the reports are being produced by some subservices,
- some of the existing reports (or subservices) can be reused for the new client,
- some of the reports required by the new client may not be derived from the existing ones and additional programming is needed,
- every additional programming means extra costs for the service provider.

The project manager has to decide which reports will be derived from the existing ones, and which reports have to be programmed. In case of small amount of reports this task is easy; however, for running this service on a larger scale, a tool capturing the mapping of existing service components onto clients' expectations and needs would increase the manager's efficiency.

The conceptual perspective of ABTS service

We are going to illustrate the introduced approach to value proposition modelling on ABTS service or, more precisely, on a selected part of the ABTS service. To illustrate the benefits for the problem mentioned in the introduction to this section, we then present the process of adding a new service for another client. We show the mapping of client needs and then illustrate the finding of the decomposition of the new service onto the existing subservices.

When introducing a new service, we can basically follow three approaches: top-down analysis, bottom-up construction of a service from existing service portfolio, or a combination of the previous two. The last option, the combined approach, is usually the most frequent in practice (see *Goal service modelling* and service modelling method *SOMA* by IBM (Arsanjani and Allam, 2006)).

We are going to describe a variant of the combined approach, starting with the statement of the value proposition directly targeted to client needs (we will refer to this value proposition as to the *client value proposition* later):

Costs and revenue are optimized by using departmental and employee chargeback.

This value proposition is so general that it is in fact service provider independent. It reflects the fact that the client needs not to be interested in the way how the value proposition is actually delivered. However, the provider has (or should have) an intention how to fulfil the client value proposition. This provider intention will be captured by another value proposition called *provider value* proposition. The provider usually formulates this value proposition as an answer to the question what he/she needs to deliver to be able to fulfil the client value proposition.

The provider value proposition will be modelled by defining the characteristics which are fulfilled by this value proposition. The characteristics are defined by providing the relevant part of a model (because we are describing a *conceptual* perspective, the model type used is the *conceptual* model):

C1. *(#Billing report) which contains the value of departmental chargeback for every department, and the value of employee chargeback for every employee, of given (#Organisation) for given (Time period) / 0,1:1,M*[2]

Using this characteristic, we can define a service *S1* with value proposition stating that the service delivers the C1, i.e. the *(#Billing report) which contains the value of departmental chargeback for every department, and the value of employee chargeback for every employee, of given (#Organisation) for given (Time period) / 0,1:1,M.*

We must not forget that we have defined the service *S1* on the *conceptual level*. A conceptual service needs not to

2 To avoid very verbose definitions repeatedly defining the same concepts over and over again, the referenced concepts (#Billing report) and (#Organisation) can be defined in a separate concept dictionary.

correspond directly to a physical implementation of this service. This case is the same as the relationship of an entity in a conceptual data model to an entity of a logical or physical data model.

So far, we have defined the service delivering the *provider value proposition* which is still rather high level. Next, we attempt to find services which are needed by the currently found service for the delivery of its results. But first, let us stop and think about what we are doing just now.

We are building a conceptual view of a service; in accordance with this intention, the resulting view should tell us what concepts and relationships are related to the modelled service. Particularly, it has to answer the following two questions:

- what information can be retrieved from a service output,
- what information a service needs to retrieve from other services to be able to “compute“ its results.

The answer to the first question is given by the value proposition of S1 which states that S1 delivers the C1. C1 is a characteristic defined by a part of a conceptual model which is exactly the tool that is used to define concepts and their relationships.

Let us move on to the second question – what information does the service need to fulfil its value proposition? A possible answer is given in the following two characteristics:

C1.1. *(Value) of departmental chargeback for every department of given (#Organisation) for given (Time period) / 0,1:0,M*

C1.2. *(Value) of employee chargeback for every employee of given (#Organisation) for given (Time period) / 0,1:0,M*

These two characteristics and their relationship to service S1 are shown by Figure 5: blue rectangles depict *services*, the callouts represent *service characteristics*, and the topmost text over the arrow is the *client value proposition*. Using these characteristics, we can define two services, *S1.1* and *S1.2*, each of them having a value proposition stating that the service *S1.1/S1.2* delivers the *C1.1/C1.2*.

We can go on and find out what information is required by services *S1.1* and *S1.2* to perform their calculations. Let us decompose services *S1.1* and *S1.2* further; the results are displayed in Figure 6 and Figure 7 respectively.

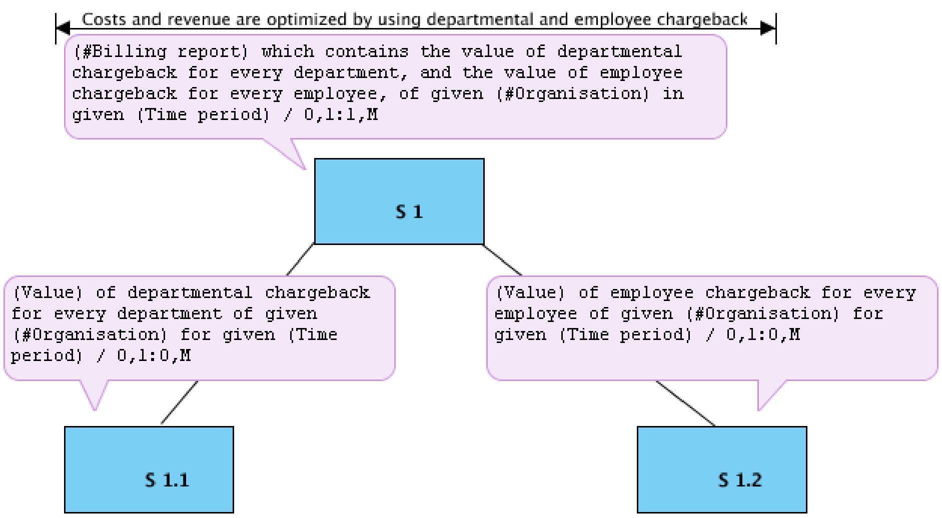

Figure 5: Conceptual view of the service – top level with client value proposition

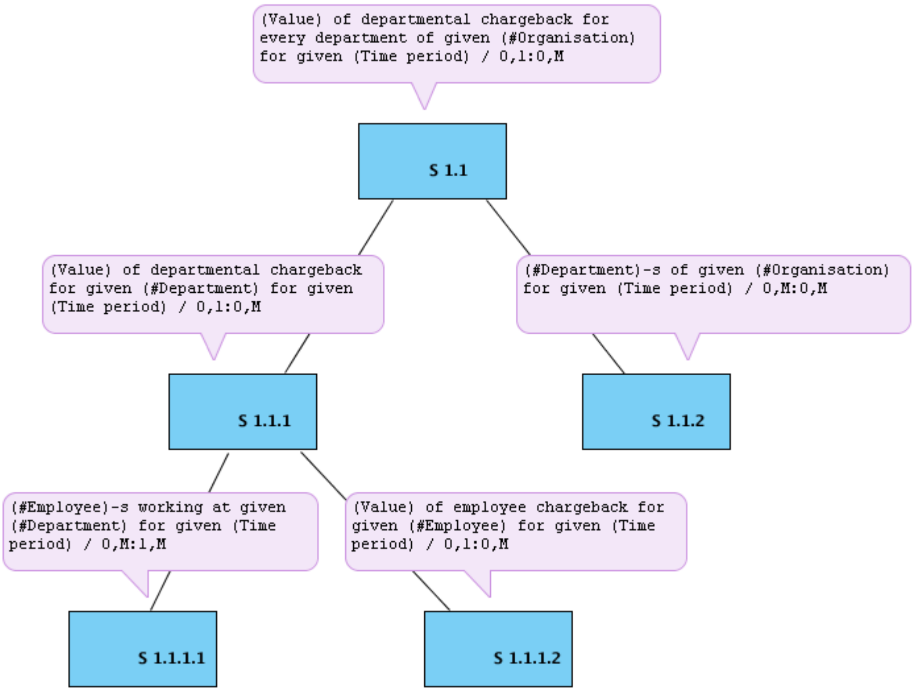

Figure 6: Conceptual service decomposition – departmental part

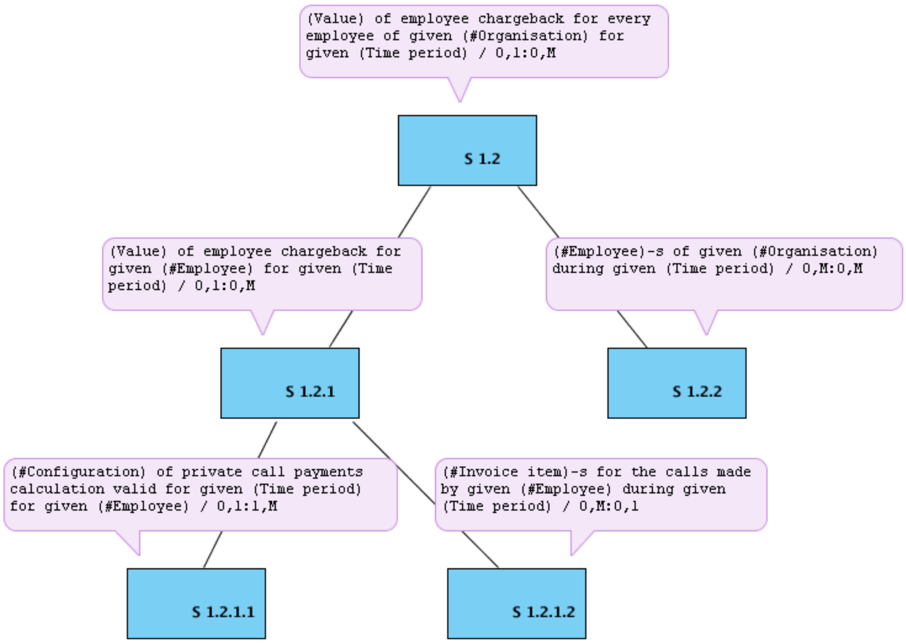

Figure 7: Conceptual service decomposition – employee part

Next, have a look at the overall diagram as shown in Figure 8. The interesting point is indicated by the blue arrow labelled *definable*. It states that the service S1.1.1.2 can reuse the service S1.2.1 for the calculation of its results.

The concept of *definability* is borrowed from the HIT conceptual modelling method[3] – we can utilize it by saying that a service A with the characteristic C1 is *definable* from a service B with the characteristic C2, if the set of HIT attributes defined in C1 is *definable over* the set of HIT attributes defined in C2.

Using definability, we can define the semantics of the connections in the diagram: imagine node S1 connected to node S1.1 as shown in Figure 5. The node S1 provides its client with information which is definable over the

[3] see section *Semantic Modelling for Services*

information provided by the node S1.1. In general, the diagram displays the following:

- what information can be retrieved from a node,
- what information a node needs to retrieve from other nodes to be able to „compute“ its results.

Definability has one more important consequence: the diagram emphasizes the distinction between information provided by an empirical function (needed/provided by a service) and the information computed analytically from the previous one. The reason is that while analytical computation can be implemented by programming or other algorithmical procedures, the empirical information must be recorded somewhere in the system (e.g. in a database). When empirical facts are missing it is often harder (and therefore more expensive) to incorporate them into the service compared to implementing a new calculation method.

This kind of perspective can be useful for services which provide information. First, this could look as a too hard limitation; however, lots of services (e.g. all IT services) deal with information and their purpose indeed is to provide information.

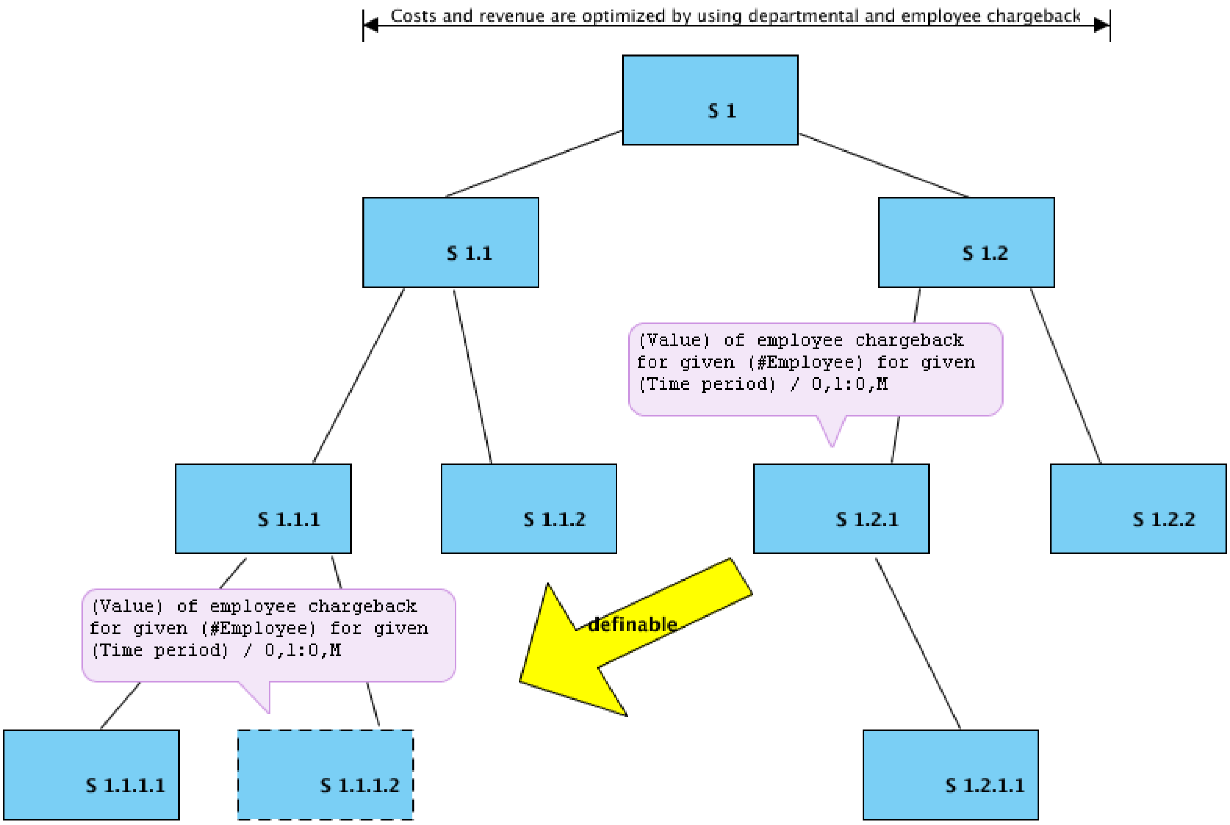

Figure 8: Conceptual view of the service – overall view

Let us now consider the problems mentioned in the first part of this section and evaluate the conceptual perspective on them:

The first use case was the refactoring of the original laborious service into a more efficient form needing less human resources because of utilization of specialized software support. Since the original service was lacking up-to-date specification documents, the analysis and design parts of the refactoring took considerable amount of time. In fact, the analysis had to be done almost from the scratch. As usual in (software) development, the most painful moments occurred every time we had found out that some earlier assumption was wrong:

- another (or different) information was needed for the calculation of the output,
- we had to change the computation of some outputs for one client without affecting the outputs for the other clients.

The conceptual perspective shows exactly the information needed for the service to carry out its outputs. Unfortunately, we did not have this tool at our disposal when we had been working on the analysis; however, having mapped selected parts of ABTS service by the conceptual perspective is turning out useful for further development because it defines what (empirical) information is required during delivery of the service.

The second use case mentioned setting up the ABTS service for a new client. Naturally, the more existing services are simply reused, the faster (and cheaper) the service set-up and initiation. The conceptual perspective can partially help in this process by the following:

- providing an organized mapping of client value propositions onto the services and their „subservices“,
- using a semi-formal semantics oriented approach developed in conceptual modelling HIT method, it allows to describe the information needed by particular (sub)services in uniform and structured manner; consequently, the communication between different people engaged in the project is fostered and the project manager can identify relevant existing services (or their subservices) faster,
- the semantic-oriented description enables using of several techniques defined by the HIT conceptual modelling method, such as looking for (sub)services providing information which is *definable* over information provided by another services, or decomposing services providing complex information (see the section about *decomposability/reducibility* in (Winkler, 2011)); using these techniques can lead to some sort of a *normal form* of the conceptual perspective removing any undesired redundancies[4].

The semantic description has been applied to services realized by computer software, however, we should emphasize that there is no restriction to do so. The same form of description could be applied to services realized entirely by human operators or, in the spirit of service systems, by a configuration of people and technology. The description is also quickly readable by a non-expert, on the contrary to formal models.

Notice that the case study does not define any of the services or value propositions by directly using the semi-formal definitions presented in the previous sections. The important feature of a modelling technique should be its ease of use with respect to its users, therefore we had intentionally kept the most of the theory “outside“ this case study, and we expect to do so in other applications of this modelling technique. However, the definitions allowed making clear what we understand by value proposition and what its relationship to service systems is, providing solid ground for further studies and potential enhancements of the model. The key principle behind the described approach is recursive utilization of the Diamond of Attention Focusing—see Figure 1. The services and subservices are recursively taken as objects to which our attention is focused to recognize necessary connections, to apply suitable operations (analytical functions) and/or integrity constraints (rules), and to categorize these objects. This allows relating services with provider value propositions to appropriate client value propositions.

The service conceptual perspective has been developed recently and has yet to be validated against a number of other services; however, the results obtained so far prove that this tool can bring benefit to the analysis and design of services.

Conclusion and Future Work

This paper introduced an original type of modelling technique based on semantics-oriented conceptual modelling method HIT and semi-formalized view of service systems, services, and value propositions. The outcome of the modelling technique, called conceptual perspective, connects a service to its value proposition and maps the service onto its partial “subservices” according to the characteristics which model the information that a service provides or requires for its operation.

Since the perspective is based on the conceptual modelling method HIT, it is able to utilize the HIT concepts of definability and decomposability to achieve normalization of the service described by the perspective. The normalization can result in the identification of redundant services. However, in some cases the normalization can be undesired, to separate service components, for instance. Therefore, the normalization cannot be applied “blindly” and it is yet to be investigated.

The conceptual perspective has been used to model the part of the ABTS service providing advanced billing of telecommunication services. Intensive validation of the perspective using other services is subject to future work.

Finally, we intend to work on a set of guidelines for using this approach in a concrete business case. The procedure must be rephrased into a form comprehensible to people in practice.

References

Alter, S. (2008), “Service system fundamentals: Work system, value chain, and life cycle”, *IBM Systems Journal*, Vol. 47 No. 1, pp. 71–87.

Arsanjani, A. and Allam, A. (2006), “Service-oriented modeling and architecture for realization of an SOA“, in *SCC '06: Proceedings of the IEEE International Conference on Services Computing*, IEEE Computer Society, Washington, DC, USA.

Barnes, C., Blake, H. and Pinder, D. (2009), Cr*eating and Delivering Your Value Proposition: Managing Customer Experience for Profit,* Kogan Page.

4 Not all redundancies are undesired. See Conclusion and Future Work section.

Duží, M., Krejčí, F., Materna, P. and Staníček, Z. (1986), "HIT Method of the database design (a functional approach to information representation)", Research report, Brno University of Technology.

Duží, M. (2002), *Logical Foundations of Conceptual Modelling*, Habilitation thesis, VŠB-Technical University of Ostrava.

Duží, M., Jespersen, B. and Materna, P. (2010), *Procedural semantics for hyperintensional logic*, Springer.

Katzan, H. (2009), "Cloud software service: Concepts, technology, economics", *Service Science*, Vol. 1 No. 4, pp. 256–269.

Lanning, M.J. (1998), *Delivering profitable value: a revolutionary framework to accelerate growth, generate wealth, and rediscover the heart of business*, Perseus Books.

Maglio, P.P. and Spohrer, J. (2008), "Fundamentals of service science", *Journal of the Academy of Marketing Science*, Vol. 36, pp. 18–20.

Maglio, P.P., Srinivasan, S., Kreulen, J.T. and Spohrer, J. (2006), "Service systems, service scientists, SSME, and innovation", *Communications of the ACM*, Vol. 49 No. 7, pp. 81–85.

Osterwalder, A. and Pigneur, Y. (2003), "Modeling value propositions in e-business", in Sadeh, N.M., Dively, M.J., Kauffman, R.J., Labrou, Y., Shehory, O., Telang, R. and Cranor, L.F. (Ed.), *ICEC*, volume 50 of *ACM International Conference Proceeding Series*, ACM, pp. 429–436.

Qiu, R.G. (2009), "Computational thinking of service systems: Dynamics and adaptiveness modelling", *Service Science*, Vol. 1 No. 1, pp. 42–55.

Spohrer, J., Maglio, P.P., Bailey, J. and Gruhl, D. (2007), "Steps toward a science of service systems", *IEEE Computer*, Vol. 40 No. 1, pp. 71–77.

Staníček, Z. (2009), "SSME*: Service Systems, Modeling, Execution, Education, Evaluation", available at: http://seslab.fi.muni.cz/documents/study_materials/SSMEstar_manuscript.pdf (accessed 30 April 2011).

Staníček, Z. and Winkler, M. (2010), "Service systems through the prism of conceptual modelling", *Service Science*, Vol. 2 No. 1/2, pp. 112–125.

Tichý, P. (1988), *The Foundations of Frege's Logic,* De Gruyter, Berlin-New York.

Vargo, S.L. and Lusch, R.F. (2004), "Evolving to a new dominant logic for marketing", *Journal of Marketing*, Vol. 68 No. 1, pp. 1–17.

Winkler, M., Staníček, Z. (2011), "Conceptual modelling using the HIT method: Practical tutorial", available at: http://seslab.fi.muni.cz/documents/study_materials/HIT_tutorial.pdf (accessed 30 April 2011).

Winkler, M. and Dosoudil, V. (2011), "On Formalization of the Concept of Value Proposition", *Service Science*, Vol. 3, No. 3, pp. 1-13.

Authors:

Marek Winkler

Faculty of Informatics, Masaryk University, Botanická 68a, 602 00 Brno, Czech Republic
Mycroft Mind Inc., Jundrovská 31, 624 00 Brno, Czech Republic
xwinkler@mail.muni.cz, marek.winkler@mycroftmind.com

Marek Winkler is a PhD student at the Faculty of Informatics at Masaryk University in Brno and works as a software developer with Mycroft Mind Inc. (Masaryk University spin-off company). He has 5 years of experience in the design and development of knowledge-intensive systems and their applications. His research interests include conceptual and service-oriented modeling, design and development of enterprise applications, and logic programming. In addition, Marek Winkler is a member of Service Systems Laboratory as well as Service Systems, Modeling and Execution research group at the Faculty of Informatics at Masaryk University in Brno.

Zdenko Staníček

Faculty of Informatics, Masaryk University, Botanická 68a, 602 00 Brno, Czech Republic
Mycroft Mind Inc., Jundrovská 31, 624 00 Brno, Czech Republic
stanicek@fi.muni.cz, zdenko.stanicek@mycroftmind.com

RNDr. Zdenko Stanicek, Ph.D. graduated at the Masaryk University in Brno, specializing in mathematics, receiving doctoral degree in applied informatics. He is a certified IS auditor;
a member of the Czech Association for System Integration and the Czech Project Management Association, and also a certified project manager according to the International Project Management Association (IPMA). Moreover, he is an assessor of the IPMA certification process for the Czech Republic, a member of Project Management Institute (PMI) and a member of the Chartered Institute of Personnel and Development, London, UK.
He has 35 years of experience in the design of large scale information systems and conceptual models, 28 years of experience in the project management of IS design and implementation, 23 years of experience in projects of strategic planning, project management and business process modeling; and he is also the author of about 80 articles and publications.
He is an associate and senior consultant of SHINE Consulting, Chief System Architect of Mycroft Mind Inc. (spin-off of Masaryk University). Further, he teaches at the Faculty of Informatics at the Masaryk University in Brno. He is the SSME master study branch guarantor and the head of Service Systems Laboratory at this faculty.