11-28-2011

Service Elements Valuation Using an Enterprise Architecture Language

Fons Wijnhoven
University of Twente, a.b.j.m.wijnhoven@utwente.nl

茅|Sprouts Working Papers on Information Systems ISSN 1535-6078

Service Elements Valuation Using an Enterprise Architecture Language

Fons Wijnhoven
University of Twente, Netherlands

Abstract
This article defines a service as an architecture of processes, software and infrastructure elements for serving people. An architecture language therefore is a means to structure and analyze the values of service elements and the service as a whole. We provide a value-based perspective, which first includes a review of the concept of value in the context of service architectures for services. Here we conclude that multiple, even competing, values are at stake for different parts of a service. Second, the paper discusses a method for the valuation of a service using competing value constructs. We also demonstrate by a case how a formal architecture language can be used to calculate service values. Finally, the results are discussed and suggestions for further research are given.

Keywords: Service value, Value centers, Service architecture, Architecture language

Permanent URL: http://sprouts.aisnet.org/11-164

Reference: Wijnhoven, F. (2011). "Service Elements Valuation Using an Enterprise Architecture Language," Proceedings > Proceedings of SIGSVC Workshop . *Sprouts: Working Papers on Information Systems*, 11(164). http://sprouts.aisnet.org/11-164

INTRODUCTION

A service is "… any act or performance that one party can offer to another that is essentially intangible and does not result in the ownership of anything" (Kotler, 1988). Services may deliver intangible values as a prime value generator and may be accompanied with material goods. For instance for information services, a key result of using it is having meaningful data (this is what Kotler (1988: 477) names "a major service with accompanying minor goods"; the major service is "informing" and the minor good is "data and information technology"). Sometimes services accompany material good supplies, like in the case of computer services and car services. In the information systems discipline a service is often defined as "… a unit of functionality that some entity (e.g., a system, organization, or department) makes available to its environment, and which has some value for certain entities in the environment (typically the "service users")" (Lankhorst, 2009: 85-86; (Lankhorst, 2009)). This broad description is applicable to all kind of services, such as people-based services (like professional services) and equipment-based services (like automated services) (Kotler, 1988: 478). Before the extensive use of computers for informing, information services like libraries and newspapers were mainly people-based and mostly professional services. With the growing use of computers, however, information services have mainly become equipment-based. Related to information services are web services. Such services aim at delivering software applications from different supplier locations that match the use context needs ((Pires, van Oostrum, & Wijnhoven, 2010), whereas information services aim at informing (Wijnhoven, 2012). Web services, though, can also deliver useful applications to support informing processes of information services.

Many value constructs can be used to value services. The information systems literature gives several useful approaches, like Venkatraman's value centre approach (Venkatraman, 1997), which proposes to extend the dominant cost-driven IT project assessment *criteria* by internal client satisfaction, business capability payoff, and profit and market credibility. Several techniques also exist to calculate values of information technologies, like constrained optimization techniques (e.g., linear programming), relevance tree models which use hierarchical goal models to compute a relevance score for IT (Bedell, 1985; Buckl, Matthes, & Schweda, 2009; Schuurman, Berghout, & Powell, 2008; Sen, Barach, Sen, & Basligil, 2009), multi-criteria decision making models (Chen & Cheng, 2009; Zandi & Tavana, 2010), and pair-wise comparison method (AHP) (J. Lee & Kim, 2000; Saaty, 1990; Wei, Chien, & Wang, 2005). Nevertheless, to the best of our knowledge there is no methodology available for the selection of values that apply to services-*elements*. Furthermore, values are inter-linked among service elements, which implies that improvements of one element could be beneficial but also detrimental to other elements (e.g. an improved database can cause troubles to the operation of several application when the interface with them have not been redesigned). These interlinkages are causal relations (e.g. used-by or realizations) that contribute to the value of a dependent service element. These inter-linkages thus link higher level business values with lower level application and infrastructure values in the service architecture. Therefore, value of a service architecture element is defined as *the contribution of the architecture element to the value of dependent architecture elements at the same or a higher level in the enterprise architecture*. Service elements and their positions in the architecture language are given in Table 1 (based on (Lankhorst, 2009). The architecture aspects of ArchiMate (which is an enterprise architecture language) refer to three types of services: information services, software services, and infrastructure services. Note that software and infrastructure service, obviously, also need a definition of its business (and application) layer to become successful.

	Information aspect: Information services	Behavior aspect: software services	Active structure aspect: Infrastructure services
Business layer	Business objects and representations	Business services and processes	Actors and roles
Application layer	Data objects	Application services and functions	Application elements
Technology layer	Artifacts	Technology services and system software	Devices and networks

Table 1. ArchiMate layers, aspects and elements

In ArchiMate, architecture elements are given a position in layers and several causality constructs are used to formalize the contributing values of an element to higher layers (e.g. realization and used-by) or contributions to adjacent elements in the same layer (like information flows and triggers). We distinguish focal and dependent elements in a service architecture. A *focal element* is a service element of which we aim to calculate its value, and which is contributing to the value of other elements in the same layer or in a higher layer. A *dependent element* is any other architecture element that receives a value contribution from the focal element. A dependent element may be in a higher layer of the enterprise architecture or in the same layer as the focal element (see Figure 1). We also identify contributing elements, which deliver value to a focal element. These contributing elements are costs and will be excluded from this paper on value.

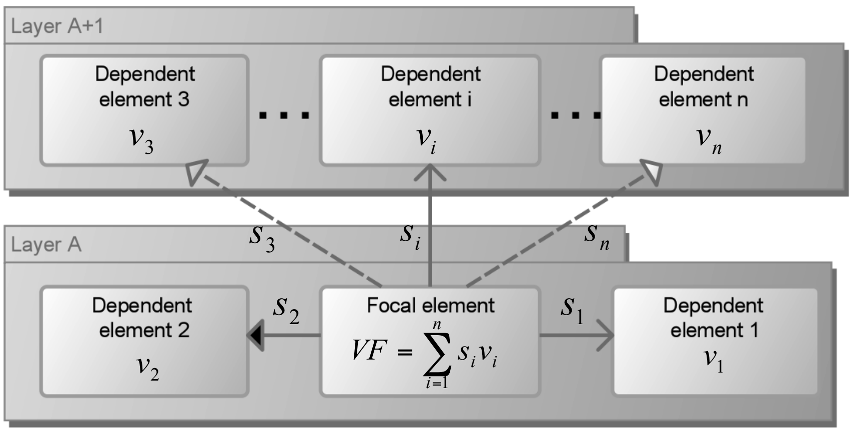

Figure 1. Focal and contributing architecture elements and value calculation. Note: v=vector of parameters.

For service valuation we need to know the relevant value constructs for the elements (which differ a lot, as will be argued later on) and we need to know the strengths of the causal relations to be able to know how much an element contributes to values of an adjacent or higher layer element in the enterprise architecture. This is the challenge of this design science paper: aiming at the development of a method and constructs (Hevner, March, Park, & Ram, 2004; Wand & Weber, 2002). The paper is further organized as follows. The next section presents the main research questions and research methodological issues for this study, after which a review of different value approaches from the literature is given. Next, we discuss how the values and their metrics can be allocated to elements of a service, and we will demonstrate the resulting service valuation method by a case. Finally, we discuss the practical implications of our approach and we draw conclusions and give pointers to future work.

RESEARCH QUESTIONS AND APPROACH

A method by which service valuation is made using an enterprise architecture language can avoid sub-optimization tendencies in service development decisions (Hale, Haseman, & Groom, 1989). More precisely, decision impacts may move from locally optimizing specific elements to a more globally focused optimization of a service (Henderson & Venkatraman, 1993). This, however, requires both an understanding of the dependencies among enterprise elements, and an enterprise architecture language that specifies these relationships in terms of (causal) contributions of one entity to the other. Therefore, the first research question is:

Question 1: What values have to be linked to which service architecture layers?

The goal of our research is to develop a method for systematic service valuation that is architecture-driven, structured by a formal causal architecture language, and incorporating multiple relevant values. Following design science objectives (Hevner et al., 2004), we do not only construct this method but we also assess its relevance for practice by demonstrating the method in a case. This is why our second research question is:

Question 2: What are the characteristics of a usable architecture-driven service valuation method?

As stated before, we aim at the development of *constructs* for service valuation, a *model* for linking value and enterprise architecture elements, and a *method* for service valuation, which are three of the key products of design science (March & Smith, 1995; Wand & Weber, 2002). We also illustrate our method by an application in a case as an instance (the fourth and final product of design science). Walls et al (Walls, Widmeyer, & El Sawy, 1992) and Markus et al (Markus, Majchrzak, & Gasser, 2002) state that effective design processes require the application of micro or kernel theories from other disciplines (like psychology, economics or mathematics) to be able to ground design decisions. The value constructs that we develop are based on management kernel theories (R. Kaplan & Norton, 1996b; Quinn & Rohrbaugh, 1983). The model that links values with enterprise architecture elements are based on product-oriented IT valuation kernel theories present in the literature on information systems (J. D. Kaplan, 2005; Venkatraman, 1997). This model is also implemented in a structured spreadsheet application. The ambitions of design science are not limited to academically sound reasoning and creative literature reviews (i.e., rigor), but also to adopt the problems of a real organization (Hevner et al., 2004). Consequently, we introduce a case to evaluate the appropriateness of our insights for solving practical problems.

A REVIEW OF VALUE CONSTRUCTS AND SERVICE VALUATION

Enterprise architecture is a system involving layers from high-level business values and goals to operational means. For enterprises, Beer (Beer, 1979, 1984) recognizes at least five management systems that together are included in a viable system of any kind. The first system (Systems One) includes the enterprise's operational system that delivers products and services to clients or market segments. Enterprises have mostly multiple Systems One that all can profit from mutual adjustments (e.g. load balancing, sharing of resources and coordination) and hierarchical control and command which brings the systems one in connection to higher level strategic objectives. Beer names mutual adjustments management System Two and hierarchical command is named System Three. System Three is informed by a business intelligence system (System Four) about the wider environment of the enterprise and its consequences, and by System Five about the identity and scope philosophy of the enterprise. System Five has the task to realize pre-conditions for the enterprise's survival through the definition of a proper identity definition, i.e. one that aligns the enterprise to future needs of the environment. For this, System Five uses information gained from System Four about the outside world of the enterprise, and information from System Three about the internal capabilities, resources and possibilities. The validity of the viable systems model (VSM) for enterprise architecture development has been recognized before (Buckl et al., 2009; Lewis & Millar, 2010; Peppard, 2005). Obviously, each viable subsystem uses different values. For System One, operational quality and efficiency is important. System Two focuses on internal coherence and consistency. Systems Three evaluates Systems One and Two on effective use of its resources given System's Three understanding of its objectives. For System Four, learning and innovation are the key values. System Five focuses on the longer strategic goals and survival of the enterprise.

A service, like any viable system, (e.g., a human body) has to cope with all values, which often compete for limited resources (Quinn & Rohrbaugh, 1983). To solve the resulting conflicts of values, the Systems 1-5 are connected to each other through crucial interfaces but have loose coupling where possible.

Organization scientists also have empirically identified multiple values for organizational enterprises. Quinn and Rohrbaugh (1983) have identified four *competing* organizational values: rational goals, internal processes, human relations, and open systems. These four value areas are distinguished via the internal-external and flexibility-control dimensions of organizations. This model emphasises four core value models and their related goals and means, with output quality as an integrative dimension. Such value diversity for organizations is also identified by economists and accountants (Norreklit, 2000). The literature on organizational performance has concluded that organizational objectives are multi-dimensional, i.e., that different measures have to be applied and that these measures are related to different and mutually *unrelated*, or even competing values (Cameron & Quinn, 2006; Quinn & Rohrbaugh, 1983). Some values may reduce others, e.g., more emphasis on process efficiency may be at the expense of some operational flexibility. Because different stakeholders have different stakes and thus can hold different values, a dominant coalition (top management) that solely values its own values, may come into conflict with other stakeholders and consequently ruins the effectiveness of the cooperative system (Barnard, 1968). In other words, values may compete for the same limited resources.

Because of the existence of competing values, Kaplan and Norton have developed the Balanced Score Card (BSC) method (R. Kaplan & Norton, 1996a, 1996b) for performance management practice. The balanced score card method identifies financial, customer, business process, and growth and learning as major values for business valuation (R. Kaplan & Norton, 1996b). It has also become popular for IT management and IT portfolio management (Van Der Zee & De Jong, 1999). Consequently, we can prioritize values and objectives over different types of service elements. Decisions taken from the perspectives of these values may be conflicting. Thus, service management has to make ethical decisions by prioritizing values as an outcome of power balances and political processes (Earl, 1996; Van Der Zee & De Jong, 1999).

In the field of IT project and portfolio valuation, Bedell (Bedell, 1985; Schuurman et al., 2008) proposes a method for computing an IT portfolio's value using the contribution relationships between IT and business artifacts as input. The underlying idea of the method is that for information systems a balance is needed between their level of *effectiveness* and their level of *strategic importance*. Depending on the question addressed, the importance and effectiveness of IT support can be analyzed for (1) the organization as a whole, (2) business processes, or (3) business activities. Investments are more crucial if the ratio between the effectiveness of an information system and its importance is worse. The following information needs to be determined to calculate this ratio in Bedell's hierarchical goal approach to information systems assessment: (1) the importance of each *b*usiness process to the *o*rganization (IBO); (2) the importance of each business *a*ctivity to the *b*usiness processes (IAB); and (3) the *e*ffectiveness of an information system (*s*oftware *a*pplication) in supporting business activities (ESA). The work of Bedell is built around two important ideas: one should consider different types of values for different enterprise architecture layers and value propagates through the enterprise architecture layers from technology towards the business layers. This idea is highly applicable to services. However, Bedell *does not propose precise definitions of importance and effectiveness and it remains unclear how the quantitative inputs for these measures should be obtained*. Furthermore, Bedell only works with two generic value types, while other approaches (as it is explained below) identify many others. Not discussing the multitude of value constructs would ignore the discussion on value types and business goals that has been carried out in economics and management science (Anthony, 1965; Mitchell, Agle, & Wood, 1997; Quinn & Rohrbaugh, 1983). Buschle and Quartel (Buschle & Quartel, 2011) therefore have extended Bedell's work by a value construct extension on top of the ArchiMate business layer. See Figure 2.

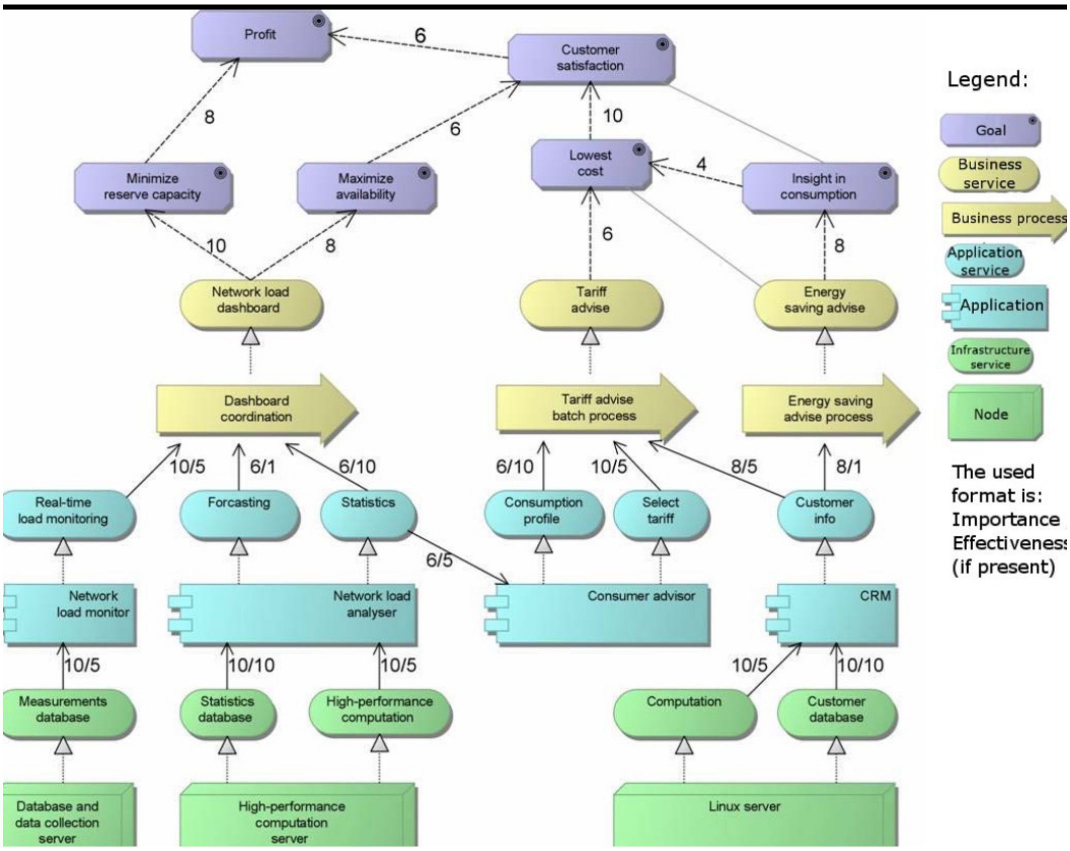

Figure 2. Goals as extensions on ArchiMate. Source: Buschle and Quartel (Buschle & Quartel, 2011) p. 377.

Note that the two top layers in this figure denote goals and values as an extension of Archimate diagram. The two layers below that are the Archimate business layers. Below that the diagram includes the two Archimate application layers. Finally, the two lowest layers are the Archimate technology layer (sometime incorrectly also named infrastructure layer). The causal links represent the Importance/Effectiveness ratio as defined by Bedell (1985).

This is indeed a valuable contribution to Bedell's framework and ArchiMate, although values are only attached on top of the business layer. Following Beer (1979, 1984), however, values can be present at each layer and place in an enterprise, and thus values can also be directly linked to application and technology layers. This is very true for services, where sometimes software, technology and databases are treated as services themselves in an agile architecture that easily allows the replacement of service elements by others and allows the sales and availability of services as elements for other service architecture (O. K. D. Lee et al., 2006). The loose-coupling among layers and elements also requires to approach value in a more decentralized manner. The practice in the management of larger firms has shown that multiple values at different layers in organizations are present and functional (Mintzberg, 1993, 2000).

Venkatraman (1997) has also addressed this multi-value phenomenon in IT projects and portfolios. He identified four value systems for value centers named 1) costs for systems, maintenance and employees, which can be benchmarked among organizations (Doll, Deng, & Scazzero, 2003; Tardugno, DiPasquale, & Matthews, 2000; Weill, 2004) for cost centers, 2) client satisfaction, competiveness and service levels measured through e.g. downtime and reliability for service centers, 3) business capabilities and investments payoff (including investment risks and uncertainties) for investment centers, and 4) realized market success (not only in financial terms, but also in intangibles like network positions and complementary skills and resources) for profit centers.

ASSIGNING VALUES TO SERVICE ELEMENTS

In order to assign values to service elements, one needs the specification of values relevant for each layer.

Important for the *business layer* is what the stakeholders want (Kulkarni, 2008). When the *stakeholder interest* is identified, a set of *strategic goals* can be derived (Sikdar & Das, 2009). *Strategic goals*, the concern of System Five, are defined as the ability of the elements to improve the competitive position of the service in its environment. Such a position can be related to strategic objectives and identity, its market position (market leader versus periphery), the decline of market communication and interaction costs, market information collection and analysis (business intelligence) (Davern & Kauffman, 2000; Weill & Broadbent, 1998). Bedell's method (Bedell, 1985) as described by Schuurman et al (Schuurman et al., 2008) is also very useful for *cost-benefit* calculation, because it is able to help in consolidating the costs of a set of elements for different organizational values (see also Figure 1 and its explanation). However, this method has considerable risks, because it does not specify value clearly and thus may easily slip into a cost focussed (cost centre) approach, thus lacking support for other business values. But, cost-benefit is a central value for the management of operational processes, i.e. the task of System One. *Service satisfaction* expresses the contribution of a service element to another element. This is similar to viable System Two, which is not a layer but the causal logic between elements included in a service architecture. Service satisfaction can be measured by criteria like information completeness, accuracy and correctness, format, ease of use, and timeliness. Proposals for measuring service satisfaction of IT are given by (Doll, Deng, Raghunathan, Torkzadeh, & Xia, 2004). *Profit payoff* is defined as the contribution of an element to the goals of a higher level (System Three) in the service architecture. IT profit payoff measures have been developed by (Kohli & Devaraj, 2003; Thatcher & Oliver, 2001). *Innovation and learning*, the key concerns of System Four, is the extent to which elements or a set of elements enable new processes and their efficiency in achieving these innovations. The opportunity of achieving higher flexibility and realizing new insights has often been seen as conflicting with IT investments (Davenport, 1998), but recently research on agile service architectures have indicated opportunities of increasing business flexibility (Byrd & Turner, 2000; O. K. D. Lee et al., 2006; Ross & Westerman, 2004; Sher & Lee, 2004).

The *application layer* includes any structural entity in the application layer, which can be a complete software system. For the application layer *Service satisfaction* is important because this layer has elements which the user might directly influence (Schmitz et al., 2008). Furthermore, *flexibility* is important for this layer because application elements' connections might change. Flexibility can be measured by the number of connections with different services (Wan, Wei, Song, & Zhong, 2007). Finally, for the application layer cost-effectiveness (*cost benefits*) is important because it needs to enable the business layer in a cost efficient way. For the data aspect of services, we propose the following metrics: data integrity, quality, reliability and controllability (Pipino, Lee, & Wang, 2002). Information *availability* is defined as the extent of useful, high-quality information accessible for a dependent purpose. This can be measured by the number of times information is not available when trying to access it (Tamm, Seddon, Shanks, & Reynolds, 2011). According to Närman et al. data *accuracy* (number of errors in the information) is an important factor for information quality (Närman, Johnson, Ekstedt, Chenine, & König, 2009), Zhu and Wu add *completeness* and *relevancy* of the information as indirect metrics for the quality (Zhu & Wu, 2010). Completeness can be measured by the number of times missing information is found and relevancy by the extent of redundant information. Tamm et al. use the same values for information quality, but they add *timeliness*, *interpretability* and *accessibility* of information. Timeliness can be measured by the number of times information is available too late, interpretability can be measured by the number of times information is misinterpreted, and accessibility can be measured by the number of times permission is denied (Tamm et al., 2011). Motro argues that data integrity is based on the *validity* and the *completeness* of data. Validity should guarantee that all false information is excluded and completeness should guarantee that all true information always is included (Motro, 1989). For the behavioural aspect or software services we identify security, precision and compliance (Delone & McLean, 2003), and completion time (Jonkers & Iacob, 2009). Performance can be measured by the *response time* of a service, the *processing time* of a service, and the *completion time* of a service . Another indicator for performance is the *failure rate* (Lei, Xue, & Jia, 2010). Furthermore, it is important that only the right purposes have access to the right functions (*security and privacy*). This can be measured by the number of times someone accesses something that he was not supposed to. For the infrastructure aspect of services we identify availability, capacity, satisfaction, resource utilisation, throughput (Jonkers & Iacob, 2009) and capability (Feeny & Willcocks, 1998; Ravichandran & Lertwongsatien, 2002).

The main purpose of the technology layer is enabling the other layers so it should be cost effective (*Cost-effectiveness)* and should not fail (*robustness*). Robustness can be measured with the number of breakdowns a year. Also the technology layer should be easy extendible if needed (*scalability*). Scalability can be measured by how well the system can handle increasing load (Buyya, Pandey, & Vecchiola, 2009; Gupta & Moitra, 2004). The interfaces and devices should enable the behavior that is performed by the actors, so there should not be any bottlenecks. Important values to enable the productivity are requirements for user interfaces such as *consistency* and *usability*. Usability can be measured by the number of clicks needed for an action and consistency can be measured by the number of inconsistencies in the user interface (Dix, Finlay, Abowd, & Beale, 2004). Other values for this aspect are *utilization*, *throughput* and *capacity* of the hardware. These can be measured by indicators like the server load, memory usage, processor speed, bandwidth and disk space (Lankhorst, 2009; Roy, Dubey, Gokhale, & Dowdy, 2011).

Table 2 extends Table 1 (Architecture layers and aspects) by summarizing the values identified for each layer. The actual contributions of lower layer elements thus dependent on the values of the dependent layers and their impact (strength).

Layer	Metric
Business	Accessibility; Accuracy; Agility; Availability; Capacity; Completeness; Completion time; Cost-efficiency; Failure rate; Flexibility; Interpretability: Privacy; Processing time; Relevancy; Response time; Robustness; Scalability; Security; Strategic goals; Timeliness; Usability; Utility; Validity

Appli-cation	Accessibility; accuracy; availability; Capacity; completeness; Consistency; Cost-effectiveness; interpretability; relevancy; timeliness; validity; cost-efficiency; Failure rate (lower); Privacy; Processing time; Response time; Robustness; Scalability; Security; Flexibility; Throughput; Utility
Tech-nology	Infrastructure Value. Accessibility; Availability; Capacity; Completion time; Cost-efficiency; Failure; Processing time; Response time; Scalability; Security; Timeliness; Cost-efficiency; Robustness; Scalability

Table 2. Value metrics for service layer elements

CASE STUDY OF A SERVICE ASSESSMENT

To validate the method, we have applied it to Osiris, i.e. the student- and education administration service used by the University of Twente. It is the main service that realizes the student registration and course administration.

Osiris was introduced by the University of Twente to replace several old systems: ISIS, FASIT, TAST, TOST, VIST and MAST. ISIS managed the student registrations, FASIT managed the education information, TAST was used for the exam registrations, TOST managed the study results, VIST dealt with course information and MAST managed the minors. The other application elements important for this case are RUN (connection between FASIT and Blackboard) and Blackboard (online learning environment). Blackboard also is a replacement of a similar older element called Teletop. Figures 3 and 4 show the information and software services at the application and business layers. The models focus on one business process that contains the steps a student must follow in order to take courses. The service architecture is modelled at two different moments before and after the introduction of the new Osiris system.

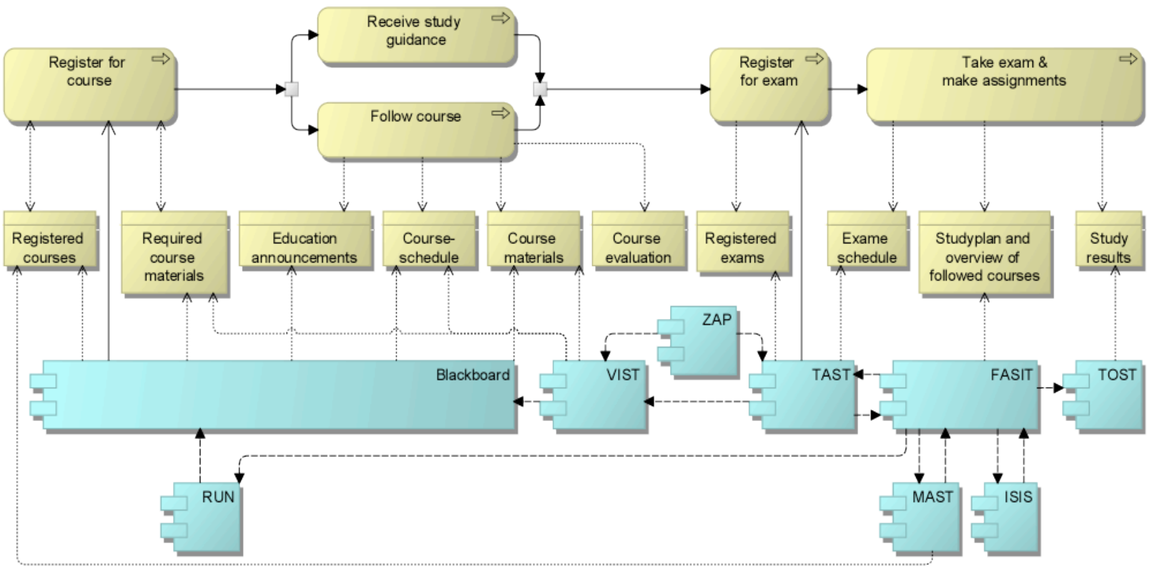

Figure 3. Student registration and course administration service before Osiris

Figure 4 shows the architecture of educational applications after the Osiris implementation.

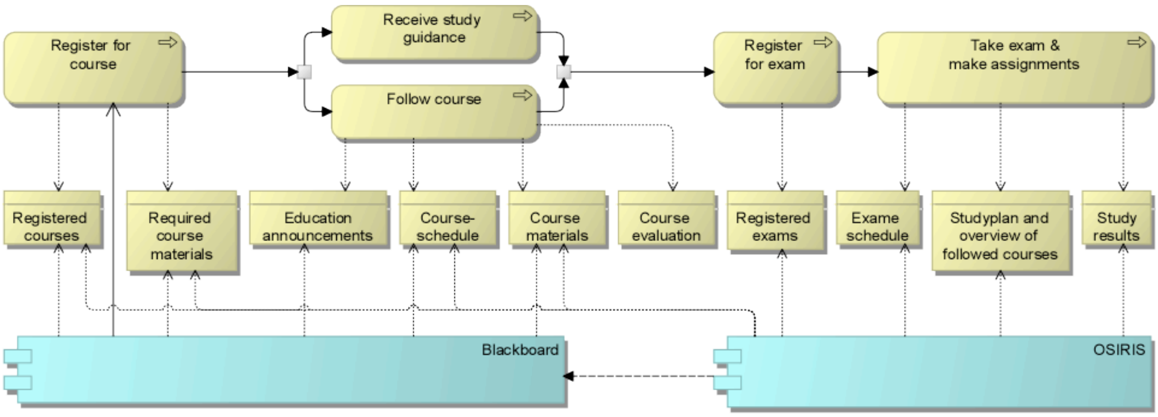

Figure 4. Student registration and course administration with Osiris

Application elements support the business processes mostly through business objects. For example, Blackboard supports the process step "follow course" through the "course-schedule", "course materials" and "education announcements" business objects. Figure 5 gives the affected services and their related matrics in a causal digram.

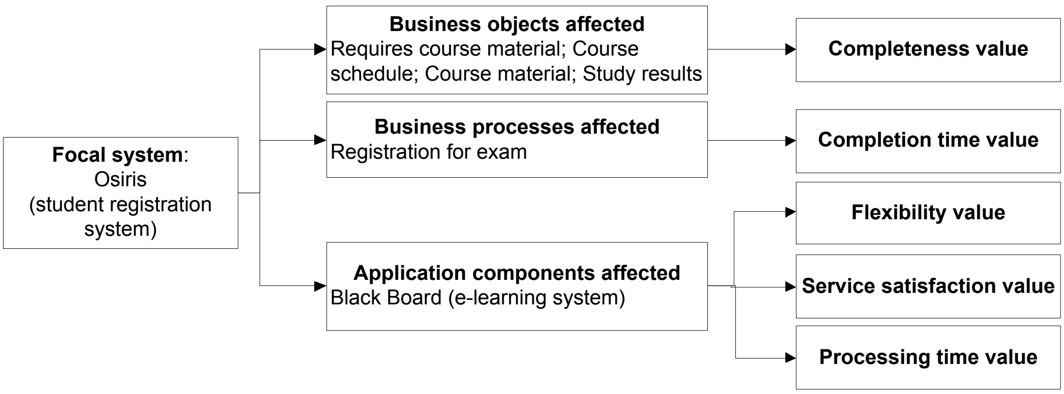

Figure 5. Causal relation between focal system and values in the Osiris project

To see if Osiris adds value to the services, we have to compare the value of the old elements with the new elements of the new service architecture. So first we calculate the values of the objects from figures 3 and 4. In order to apply our method we select the important value metrics for this case from the values we identified in the previous sections. We have three types of elements in this case: business objects, business processes, and application elements.

For the business *objects* we use accessibility, availability and completeness because in this case it is important that the business objects are useful, and high-quality information objects have to be accessible to the right elements. We measure accessibility by the cost per permission denied, availability by the cost per unavailable object and completeness by the cost per incomplete object. For the business *processes* we use completion time, failure rate and security because in this case it is important that the processes are completed quickly without errors and that only the actors who are allowed to access have the right access privileges. We measure completion time by cost per hour, failure rate by the cost per failure, and security by the cost per security breach. For the *application* elements, we use flexibility, service satisfaction and processing time because in this case it is important that the elements can easily change or add connections with new services or functions and the processing time is important for evaluating completion time of the business layer. We measure flexibility by the cost for new connections, service satisfaction by the cost of the added time needed of the users to overcome unnecessary problems, and processing time by the cost per hour. Osiris as focal element affects many other services elements.

The spreadsheets with prime data (weights and value per construst) for the calculations are given Figures 6.1 and 6.2. Please note that each value has its own scale, so the values cannot be compared. The calculations are based on a number of 9,000 enrolled students which all do on average four courses every quarter. All the other data is based on our own estimates, except the number of connections with other elements.

6.1. Value calculations in the situation before Osiris

		Accessibility			Availability			Completeness		
Focal element	Dependent component	Times accessed per year	Relation weight	Component contribution	Minutes of downtime per year	Relation weight	Component contribution	Number of missing data	Relation weight	Component contribution
Registered courses	Register for course (BP)	144000	0,8	115200	120	-0,8	-96	0	-0,9	0
	MAST (AC)	3000	0,8	2400	120	-0,8	-96	0	-0,9	0
	Blackboard (AC)	2880000	0,8	2304000	120	-0,8	-96	0	-0,9	0
	Total Value:			**2421600**			**-288**			**0**
Required course materials	Register for course (BP)	144000	0,2	28800	120	-0,1	-12	50	-0,25	-12,5
	VIST (AC)	144000	0,2	28800	120	-0,1	-12	100	-0,25	-25
	Blackboard (AC)	144000	0,2	28800	120	-0,1	-12	50	-0,25	-12,5
	Total Value:			**86400**			**-36**			**-50**
Course-schedule	Follow course (BP)	144000	0,2	28800	120	-0,4	-48	20	-0,25	-5
	VIST (AC)	144000	0,2	28800	120	-0,4	-48	20	-0,25	-5
	Blackboard (AC)	288000	0,2	57600	120	-0,4	-48	20	-0,25	-5
	Total Value:			**115200**			**-144**			**-15**
Course materials	Follow course (BP)	144000	0,2	28800	120	-0,1	-12	50	-0,25	-12,5
	VIST (AC)	144000	0,2	28800	120	-0,1	-12	100	-0,25	-25
	Blackboard (AC)	144000	0,2	28800	120	-0,1	-12	50	-0,25	-12,5
	Total Value:			**86400**			**-36**			**-50**
Registered exames	Register for exame (BP)	144000	0,8	115200	120	-0,9	-108	0	-0,9	0
	TAST (AC)	144000	0,8	115200	120	-0,9	-108	0	-0,9	0
	Total Value:			**230400**			**-216**			**0**
Study plan and overview of followed courses	Do exam and assignment (BP)	9000	0,4	3600	120	-0,3	-36	0	-0,4	0
	FASIT (AC)	9000	0,4	3600	120	-0,3	-36	0	-0,4	0
	Total Value:			**7200**			**-72**			**0**
Exame schedule	Do exam and assignment (BP)	144000	0,8	115200	120	-0,6	-72	0	-0,8	0
	TAST (AC)	144000	0,8	115200	120	-0,6	-72	0	-0,8	0
	Total Value:			**230400**			**-144**			**0**
Study results	Do exam and assignment (BP)	144000	0,75	108000	120	-0,9	-108	50	-0,7	-35
	TOST (AC)	288000	0,75	216000	120	-0,9	-108	50	-0,7	-35
	Total Value:			**324000**			**-216**			**-70**

BO before OSIRIS | Blad2 | Blad3

		Completion time			Failure rate			Security		
Focal element	Dependent component	Time it takes to register (in MS)	Relation weight	Component contribution	Times it fails per year	Relation weight	Component contribution	Number of unauthorized access	Relation weight	Component contribution
Register for exam	Registered exames (BO)	100	-0,4	-40	100	-0,7	-70	0	-0,9	0
	Tast (AC)	500	-0,4	-200	100	-0,7	-70	0	-0,9	0
	Total Value:			**-240**			**-140**			**0**

BP = Business process

AC = Application component

BO = Business object

		Flexibility			Service satisfaction			Processing time		
Focal element	Dependent component	Number of connections with components	Relation weight	Component contribution	User rating (1-10)	Relation weight	Component contribution	Time it takes to process (in MS)	Relation weight	Component contribution
Blackboard	Registered courses (BO)	3	0,5	1,5	6	0,4	2,4	100	-0,5	-50
	Required course materials (BO)	2	0,5	1	7	0,4	2,8	100	-0,5	-50
	Education announcements (BO)	2	0,5	1	7	0,4	2,8	100	-0,5	-50
	Course-schedule (BO)	3	0,5	1,5	6	0,4	2,4	100	-0,5	-50
	Course materials (BO)	3	0,5	1,5	7	0,4	2,8	100	-0,5	-50
	Register for course (BP)	3	0,5	1,5	6	0,4	2,4	300	-0,5	-150
	Vist (AC)	6	0,5	3	6	0,4	2,4	400	-0,5	-200
	Run (AC)	2	0,5	1	7	0,4	2,8	600	-0,5	-300
	Total Value:			**12**			**20,8**			**-900**

BP = Business process

AC = Application component

BO = Business object

6.2. Value calculations in the situation with Osiris

Focal element	Dependent component	Accessibility: Times accessed per year	Accessibility: Relation weight	Accessibility: Component contribution	Availability: Minutes of downtime per year	Availability: Relation weight	Availability: Component contribution	Completeness: Number of missing data	Completeness: Relation weight	Completeness: Component contribution
Registered courses	Register for course (BP)	144000	0,8	115200	120	-0,8	-96	0	-0,9	0
	Osiris (AC)	3000	0,8	2400	120	-0,8	-96	0	-0,9	0
	Blackboard (AC)	2880000	0,8	2304000	120	-0,8	-96	0	-0,9	0
	Total Value:			**2421600**			**-288**			**0**
Required course materials	Register for course (BP)	144000	0,2	28800	120	-0,1	-12	50	-0,25	-12,5
	Osiris (AC)	144000	0,2	28800	120	-0,1	-12	20	-0,25	-5
	Blackboard (AC)	144000	0,2	28800	120	-0,1	-12	50	-0,25	-12,5
	Total Value:			**86400**			**-36**			**-30**
Course-schedule	Follow course (BP)	144000	0,2	28800	120	-0,4	-48	20	-0,25	-5
	Osiris (AC)	144000	0,2	28800	120	-0,4	-48	10	-0,25	-2,5
	Blackboard (AC)	288000	0,2	57600	120	-0,4	-48	20	-0,25	-5
	Total Value:			**115200**			**-144**			**-12,5**
Course materials	Follow course (BP)	144000	0,2	28800	120	-0,1	-12	50	-0,25	-12,5
	Osiris (AC)	144000	0,2	28800	120	-0,1	-12	10	-0,25	-2,5
	Blackboard (AC)	144000	0,2	28800	120	-0,1	-12	50	-0,25	-12,5
	Total Value:			**86400**			**-36**			**-27,5**
Registered exames	Register for exame (BP)	144000	0,8	115200	120	-0,9	-108	0	-0,9	0
	Osiris (AC)	144000	0,8	115200	120	-0,9	-108	0	-0,9	0
	Total Value:			**230400**			**-216**			**0**
Study plan & overview of courses done	Do exam and assignment (BP)	9000	0,4	3600	120	-0,3	-36	0	-0,4	0
	Osiris (AC)	9000	0,4	3600	120	-0,3	-36	0	-0,4	0
	Total Value:			**7200**			**-72**			**0**
Exame schedule	Do exam and assignment (BP)	144000	0,8	115200	120	-0,6	-72	0	-0,8	0
	Osiris (AC)	144000	0,8	115200	120	-0,6	-72	0	-0,8	0
	Total Value:			**230400**			**-144**			**0**
Study results	Do exam and assignment (BP)	144000	0,75	108000	120	-0,9	-108	50	-0,7	-35
	Osiris (AC)	288000	0,75	216000	120	-0,9	-108	20	-0,7	-14
	Total Value:			**324000**			**-216**			**-49**

BP = Business process

BusObjectsBeforeOsiris | BusObjectsAfterOsiris | BusProcessBeforeOsiris | BusProcessAfterOsiris | ApplCompE

Focal element	Dependent component	Completion time: Time it takes to register (in MS)	Completion time: Relation weight	Completion time: Component contribution	Failure rate: Times it fails per year	Failure rate: Relation weight	Failure rate: Component contribution	Security: Number of unauthorized access	Security: Relation weight	Security: Component contribution
Register for exam	Registered exames (BO)	100	-0,4	-40	100	-0,7	-70	0	-0,9	0
	Osiris (AC)	300	-0,4	-120	100	-0,7	-70	0	-0,9	0
	Total Value:			**-160**			**-140**			**0**

BP = Business process
AC = Application component
BO = Business object

Focal element	Dependent component	Flexibility: Number of connections with components	Flexibility: Relation weight	Flexibility: Component contribution	Service satisfaction: User rating (1-10)	Service satisfaction: Relation weight	Service satisfaction: Component contribution	Processing time: Time it takes to process (in MS)	Processing time: Relation weight	Processing time: Component contribution
Blackboard	Registered courses (BO)	3	0,5	1,5	6	0,4	2,4	100	-0,5	-50
	Required course materials (BO)	2	0,5	1	7	0,4	2,8	100	-0,5	-50
	Education announcements (BO)	2	0,5	1	7	0,4	2,8	100	-0,5	-50
	Course-schedule (BO)	3	0,5	1,5	6	0,4	2,4	100	-0,5	-50
	Course materials (BO)	3	0,5	1,5	7	0,4	2,8	100	-0,5	-50
	Register for course (BP)	3	0,5	1,5	6	0,4	2,4	300	-0,5	-150
	Osiris (AC)	10	1	10	8	0,8	6,4	200	-1	-200
	Total Value:			**18**			**22**			**-600**

BP = Business process
AC = Application component
BO = Business object

Figure 6. Service value calculations before and after Osiris

As a summary, Osiris helped increasing the completeness, the completion time, the flexibility, the service satisfaction and the processing time of the other elements in the enterprise architecture.

CONCLUSIONS AND FURTHER RESEARCH

Here we review the answers to our research questions.

Question 1: What values have to be linked to which service architecture layer and dependent element?

We identified five value constructs that match the four layers and associations in a service architecture. For each of these value constructs several metrics are identified.

Question 2: What are the characteristics of a usable architecture-based service valuation method?

We have described a technique for the calculation of contributions of focal elements to adjacent dependent elements. The proposed architecture-driven service valuation method evaluates elements in the broader context of the service architecture. There are, however, still difficulties that have to be addressed, such as that values are competing and thus loose-coupled thinking is needed, and that if values are competing, choosing for one is at the expense of another. We apply the concept of strength-of-relationships in architectures to make enterprise architectures a useful instrument for service valuation. Such strengths are hard to quantify and no clear rules exist of doing this. Interviews with domain experts (e.g., people directly involved in the processes), monitoring and log information of the system usage, and realistic estimates are the most probable sources of information for such quantitative input.

Our allocation of value constructs over the different service layers -although extensive- is not exhaustive. Further research is needed regarding the selection of other value constructs. Furthermore, in each case values for which the "business case" is made are very much dependent on the particular context. Despite these limitations and challenges, the idea of valuing services using enterprise architecture information is a substantial improvement over any approach that values without considering their context. In this line of thinking, enterprise architecture modelling languages, such as ArchiMate, provide the necessary level of formalism, rigor and consistency to facilitate the evaluation of project proposals using project context information. Following Ross and Westerman (Ross & Westerman, 2004), a well-developed enterprise architecture is essential for effective IT outsourcing and utility computing (i.e., IT services). We agree on this, because it allows to carefully analyse the impact of a component on the whole service architecture of a firm.

REFERENCES

Anthony, R. N. (1965). *Planning and control systems: a framework for analysis*: Division of Research, Graduate School of Business Administration, Harvard University, Boston.

Barnard, C. I. (1968). *The functions of the executive* (Vol. 11): Harvard Univ Pr.

Bedell, E. F. (1985). *The computer solution: strategies for success in the information age*. Homewood, Ill.: Dow Jones-Irwin.

Beer, S. (1979). *The heart of enterprise*. Chichester: Wiley.

Beer, S. (1984). The Viable System Model: its provenance, development, methodology and pathology. *Journal of the Operational Research Society, 35*(1), 7-25.

Buckl, S., Matthes, F., & Schweda, C. M. (2009). A Viable System Perspective on Enterprise Architecture Management *International Conference Proceedings on Systems, Man and Cybernetics* (pp. 1483-1488): IEEE.

Buschle, M., & Quartel, D. (2011). *Extending the method of Bedell for Enterprise Architecture Valuation.* Paper presented at the 15th ieee International Enterprise Distrbuted Object Computing Conference.

Buyya, R., Pandey, S., & Vecchiola, C. (2009). In M. G. Jaatun, G. Zhao & C. Rong (Eds.), *Proceedings Cloud Computing* (Vol. 5931, pp. 24-44). Heidelberg: Springer.

Byrd, T. A., & Turner, D. E. (2000). Measuring the Flexibility of Information Technology Infrastructure: Exploratory Analysis of a Construct. *Journal of Management Information Systems, 17*(DAVENPORT, T. H. 1998. Putting the enterprise into the enterprise system. Harvard business review, 76.), 167-208.

Cameron, K. S., & Quinn, R. E. (2006). *Diagnosing and changing organizational culture: Based on the competing values framework*: Jossey-Bass Inc Pub.

Chen, C., & Cheng, H. (2009). A comprehensive model for selecting information system project under fuzzy environment. *International Journal of Project Management, 27*(4), 389-399.

Davenport, T. H. (1998). Putting the Enterprise into the Enterprise System. *Harvard Business Review, 76*, 121-131.

Davern, M. J., & Kauffman, R. J. (2000). Discovering potential and realizing value from information technology investments. *Journal of Management Information Systems, 16*(4), 121-143.

Delone, W. H., & McLean, E. R. (2003). The DeLone and McLean model of information systems success: A ten-year update. *Journal of Management Information Systems, 19*(4), 9-30.
Dix, A., Finlay, J., Abowd, G., & Beale, R. (2004). *Human-Computer Interaction*. Harlow, England: Pearson.
Doll, W. J., Deng, X., Raghunathan, T., Torkzadeh, G., & Xia, W. (2004). The meaning and measurement of user satisfaction: a multigroup invariance analysis of the end-user computing satisfaction instrument. *Journal of Management Information Systems, 21*(1), 227-262.
Doll, W. J., Deng, X., & Scazzero, J. A. (2003). A process for post-implementation IT benchmarking. *Information & Management, 41*(2), 199-212.
Earl, M. J. (1996). *Information management: the organizational dimension*. New York: Oxford University Press.
Feeny, D. F., & Willcocks, L. P. (1998). Core IS Capabilities for Exploiting Information Technology. *Sloan Management Review, 39*, 9-21.
Gupta, P., & Moitra, D. (2004). Evolving a pervasive IT infrastructure: a technology integration approach. *Personal and Ubiquitous Computing, 8*(1), 31-41.
Hale, D., Haseman, W., & Groom, F. (1989). Integrating islands of automation. *MIS Quarterly, 13*(4), 433-445.
Henderson, J., & Venkatraman, N. (1993). Strategic Alignment: Leveraging Information Technology for Transforming Organizations. *IBM Systems Journal, 32*(1), 4-16.
Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly, 28*(1), 75-105.
Jonkers, D., & Iacob, M. E. (2009). Architecture Analysis. In M. Lankhorst (Ed.), *Enterprise Architecture at Work* (pp. 199-230). Heidelberg: Springer.
Kaplan, J. D. (2005). *Strategic IT portfolio management: governing enterprise transformation*: Pittiglio Rabin Todd & McGrath Inc.
Kaplan, R., & Norton, D. (1996a). *The balanced scorecard: translating strategy into action*. Boston (MA): Harvard Business Press.
Kaplan, R., & Norton, D. (1996b). Using the balanced scorecard as a strategic management system. *Harvard Business Review*(1), 75-85.
Kohli, R., & Devaraj, S. (2003). Measuring information technology payoff: A meta-analysis of structural variables in firm-level empirical research. *Information systems research, 14*(2), 127-145.
Kotler, P. (1988). *Marketing management: Analysis, planning, implementation, and control* (six ed.). Englewood Cliffs, NJ: Prentice-Hall.
Kulkarni, V. (2008). A Conceptual Model for Capturing Stakeholders' Wish List *Proceedings of CSSE* (Vol. 2, pp. 275-278): IEEE.
Lankhorst, M. (Ed.). (2009). *Enterprise architecture at work: Modelling, communication and analysis*. New York: Springer-Verlag.
Lee, J., & Kim, S. (2000). Using analytic network process and goal programming for interdependent information system project selection. *Computers & Operations Research, 27*(4), 367-382.
Lee, O. K. D., Banerjee, P., Lim, K. H., Kumar, K., Hillegersberg, J., & Wei, K. K. (2006). Aligning IT components to achieve agility in globally distributed system development. *Communications of the Acm, 49*(10), 48-54.
Lei, X., Xue, K., & Jia, Y. (2010). A comprehensive metric to evaluating software aging *Proceedings International Conference on Computer Application and System Modeling (ICCASM)* (pp. V11-518-V511-520): IEEE.

Lewis, E., & Millar, G. (2010). The Viable Governance Model: A Theoretical Model for the Corporate Governance of IT. *International Journal of IT/Business Alignment and Governance (IJITBAG), 1*(3), 19-35.
March, S. T., & Smith, G. F. (1995). Design and natural science research on information technology. *Decision Support Systems, 15*(4), 251-266.
Markus, M. L., Majchrzak, A., & Gasser, L. (2002). A design theory for systems that support emergent knowledge processes. *MIS Quarterly, 26*, 179-212.
Mintzberg, H. (1993). *Structure in fives: Designing effective organizations*: Prentice-Hall, Inc.
Mintzberg, H. (2000). *The rise and fall of strategic planning*: Pearson Education.
Mitchell, R. K., Agle, B. R., & Wood, D. J. (1997). Toward a theory of stakeholder identification and salience: Defining the principle of who and what really counts. *Academy of Management Review*, 853-886.
Motro, A. (1989). Integrity= validity+ completeness. *ACM Transactions on Database Systems (TODS), 14*(4), 480-502.
Närman, P., Johnson, P., Ekstedt, M., Chenine, M., & König, J. (2009). Enterprise architecture analysis for data accuracy assessments *Proceedings of EDOC* (pp. 24-33): IEEE.
Norreklit, H. (2000). The balance on the balanced scorecard a critical analysis of some of its assumptions. *Management Accounting Research, 11*(1), 65-88.
Peppard, J. (2005). The application of the viable systems model to information technology governance. *Twenty-Sixth International Conferences on Information Systems.*, 45–58.
Pipino, L. L., Lee, Y. W., & Wang, R. Y. (2002). Data quality assessment. *Communications of the ACM, 45*(4), 211-218.
Pires, L. F., van Oostrum, A., & Wijnhoven, F. (2010). Service Registry Design: An Information Service Approach. *International Journal of Information Systems in the Service Sector (IJISSS), 2*(4), 1-21.
Quinn, R. E., & Rohrbaugh, J. (1983). A spatial model of effectiveness criteria: Towards a competing values approach to organizational analysis. *Management Science, 29*(3), 363-377.
Ravichandran, T., & Lertwongsatien, C. (2002). *Impact of Information Systems Resources and Capabilities on Firm Performance: A Resource-Based Perspective.* Paper presented at the Proceedings of 23rd International Conference on Information Systems B2 - Proceedings of 23rd International Conference on Information Systems, Barcelona, Spain.
Ross, J. W., & Westerman, G. (2004). Preparing for utility computing: The role of IT architecture and relationship management. *IBM Systems Journal, 43*(1), 5-19.
Roy, N., Dubey, A., Gokhale, A., & Dowdy, L. (2011). *A Capacity Planning Process for Performance Assurance of Component-based Distributed Systems.* Paper presented at the ICPE '11 Proceeding of the second joint WOSP/SIPEW international conference on Performance engineering.
Saaty, T. (1990). An Exposition on the AHP in Reply to the Paper "Remarks on the Analytic Hierarchy Process". *Management Science, 36*(3), 259-268.
Schmitz, D., Nissen, H. W., Jarke, M., Rose, T., Drews, P., Hesseler, F. J., & Reke, M. (2008). *Requirements engineering for control systems development in small and medium-sized enterprises.* Paper presented at the Proceedings of requirements engineering.

Schuurman, P., Berghout, E. W., & Powell, P. (2008). Calculating the Importance of Information Systems: The Method of Bedell Revisited. (1535-6078). from Sprouts Alliance http://sprouts.aisnet.org/8-37
Sen, C. G., Barach, H., Sen, S., & Basligil, H. (2009). An integrated decision support system dealing with qualitative and quantitative objectives for enterprise software selection. *Expert Systems with Applications, 36*(3), 5272-5283. doi: DOI 10.1016/j.eswa.2008.06.070
Sher, P. J., & Lee, V. C. (2004). Information technology as a facilitator for enhancing dynamic capabilities through knowledge management. *Information & Management, 41*(8), 933-945.
Sikdar, S., & Das, O. (2009). *Stakeholder Appropriate Requirement Development Approach.* Paper presented at the IEEE International Advance Computing Conference.
Tamm, T., Seddon, P. B., Shanks, G., & Reynolds, P. (2011). How Does Enterprise Architecture Add Value to Organisations? *Communications of the Association for Information Systems, 28*(10), 141-168.
Tardugno, A. F., DiPasquale, T. R., & Matthews, R. E. (2000). *IT services: costs, metrics, benchmarking, and marketing*: Prentice Hall.
Thatcher, M. E., & Oliver, J. R. (2001). The impact of technology investments on a firm's production efficiency, product quality, and productivity. *Journal of Management Information Systems, 18*(2), 17-45.
Van Der Zee, J., & De Jong, B. (1999). Alignment is not enough: integrating business and information technology management with the balanced business scorecard. *Journal of Management Information Systems*, 137-156.
Venkatraman, N. (1997). Beyond outsourcing: managing IT resources as a value center. *Sloan Management Review, 38*(3), 51-64.
Walls, J. G., Widmeyer, G. R., & El Sawy, O. A. (1992). Building an Information System Design Theory for Vigilant EIS. *Information Systems Research, 3*(1), 36-59.
Wan, S., Wei, J., Song, J., & Zhong, H. (2007). *A Satisfaction Driven Approach for the Composition of Interactive Web Services.* Paper presented at the Proceedings 31st Annual International Computer Software and Applications Conference.
Wand, Y., & Weber, R. (2002). Research commentary: information systems and conceptual modeling-a research agenda. *Information Systems Research, 13*(4), 363-376.
Wei, C., Chien, C., & Wang, M. (2005). An AHP-based approach to ERP system selection. *International Journal of Production Economics, 96*(1), 47-62.
Weill, P. (2004). Don't Just Lead, Govern: How Top-Performing Firms Govern IT *MIS Quarterly Executive* (Vol. 3, pp. 1-17).
Weill, P., & Broadbent, M. (1998). *Leveraging the New Infrastructure: How Market Leaders Capitalize on Information Technology*. Cambridge, MA: Harvard Business School Press.
Wijnhoven, F. (2012). *Information services design: A design science approach for sustainable knowledge*. New York: Routledge.
Zandi, F., & Tavana, M. (2010). A multi-attribute group decision support system for information technology project selection. *International Journal of Business Information Systems, 6*(2), 179-199.

Zhu, H., & Wu, H. (2010). Assessing Quality of Data Standards: Framework and Illustration Using XBRL GAAP Taxonomy. *Metadata and Semantic Research*, 288-299.

芽|Sprouts

芽|Sprouts
Working Papers on Information Systems | ISSN 1535-6078