Systemic Differences between SaaS- and On-Premise-ERP: An Overview of a Qualitative Option Calculation Scheme

Björn Link, Andrea Back

Andrea Back works as a leading Professor at the department of "SoM" School of Management at the institut of business computing at the university of St. Gallen. Her primary focus in research has been Business 2.0, Mobile Uni-Apps and Learning Center. She is the co-editor of newspapers such as "Zeitschrift für e-learning-lernkultur und bildungstechnologie", "Learning Waves-Blog zur Zukunft von Lernen und Wissen", "Blog Business 2.0", "WissensWert Mitmachzeitschrift über Enterprise 2.0", Knowledge Management and E-Learning" and "Fallstudiennetzwerk Enterprise 2.0".

Björn Link is a PhD-candidate at the department of "SoM" School of Management at the institut of business computing at the university of St. Gallen.

A research-based calculation scheme of IT options will be developed based on systemic differences between SaaS- and On-Premise-ERP. The calculation scheme can compute relative cost differences between SaaS- and On-Premise, and thus better determine which ERP operating mode is more financially affordable in the particular case. The systemic differences have been researched through a multiple case study with four ERP producers. The data obtained has been substantiated by ERP literature in general and SaaS literature in particular. The comparative total ERP cost was calculated by applying a total cost of ownership approach, which was modified to sum up only all relative cost differences rather than all absolute costs. This relative total cost approach enables the reader to compare the relative cost differences for each operation mode and includes financing aspects; the fixed costs were discounted and interest rates (by debt and equity) were included to compensate for the investment differences between On-Premise- and SaaS-ERP. The classification and calculation scheme is limited in that the research method is qualitative. A quantification of the adoption factors determined by building strict relations between the systemic differences and the customer's characteristics, as well as standardization and weighting of the importance of the adoption factors is not the focus of this contribution and is left to other publications.

1. Introduction

Relatively complex IT systems such as ERPs could until recently only be operated as licensed products on local servers. The Software as a Service (SaaS) innovation, drawing on existing technology, made it possible for the first time for providers not only to offer a more complex system but also to deliver it over the Internet. Each new operating model allows additional application options; the question for research is, then, which of the operating modes, SaaS, On-Premise, or perhaps ASP or some hybrid form, offers the best long-term value in a particular ERP case. It is then left to each company to select those solutions that offer the lowest cost with the best possible support for their operational processes. An ERP cost calculation method would best support the ERP operation mode-selecting end user, but a search of the extant literature yielded no results pertaining to ERP cost calculation methods or ERP delivery strategy selection.

To develop the ERP cost calculation method it is necessary to identify the systemic differences between SaaS- and On-Premise-driven ERP systems, so that a calculation of the two options can provide decision support for or against one of the two operation modes. The classification of the systemic differences, which has been researched and published in a previous article, will hence be the point of departure of the option scheme to be constructed. This paper, therefore, lays as a qualitative foundation in section 3 a comparison of the main differences between SaaS- and On-Premise-ERP and an option scheme, which will be used for calculation in section 4. The 5th section states some management strategies based on the previous calculation scheme, and the last section concerns the outlook and limitations.

To step into the field, the related work and the research method are briefly discussed in the next section.

2. Related Work and Research Method

2.1 Related Work

The extant literature on ERP systems and the SaaS operation mode in general, as well as on SaaS-ERP in particular, was the starting point of the research. Little scientific literature about calculation-based models of IT options with respect to operation modes was found, because most of the papers on ERP

systems are concerned with the strategic and functional selection processes, e. g. 1.8, 1.8. With regards to the economics of SaaS-ERP systems, there is no indicator at all in the extant literature for such a comparative calculation between SaaS- and On-Premise-ERP (c.f. 1.8 with further references). There is an approach to calculating the value added by the IT in the case of service-oriented architecture (SOA, 1.8, 1.8, 1.8) or in the case of value-oriented process modeling 1.8. This approach enables the user to calculate the in- and out-payments of a specific process and to compare the different IT options under consideration.

Another approach to investigating SaaS is the cost and the pricing of the service. In Katzan Jr. and Dowling the operation mode and characteristics of SaaS, as well as, inter alia, the costs of a SaaS-ERP system are presented 1.8. Studies that look intensively at pricing models can also be found: 1.8, 1.8, 1.8. Choudhary takes a comparatively detailed look at SaaS and On-Premise in order to use pricing models to estimate the differences in quality 1.8. While common cost calculation criteria for a SaaS-software can be transferred to a cost effectiveness calculation between SaaS- and On-Premise-ERP, price modeling has no place in a cost effectiveness calculation, since the conditions and bases of calculation are specified by the ERP producer.

2.2 Method

A two-step approach was used to construct the option calculation scheme. First, a classification of the systemic differences enabled the identification of all the general differences between the two extremes of ERP operation modes to be found. Second, the classification was used to build the option calculation scheme by relative comparison. In detail:

2.2.1 Data acquisition, data sampling and classification of systemic differences.

A "rigorous literature review" as described by vom Brocke et al. had been conducted to gain insight into what already exists 1.8. This literature review enabled a detailed exploration of all current general operation mode differences between cloud computing and On-Premise, which may be applied to the more specific domain of ERP operation modes and further identifying the remaining research gap with respect to the more specific ERP operation mode differences 1.8. The data from the literature were analyzed using both open coding and operation mode-contrasting meta-matrices 1.8, 1.8. The matrix thus obtained, when analyzed using pattern coding, revealed 6 main pattern clusters and many systemic differences.

The applicability of the general differences found in the literature was investigated and verified through case study research at ERP producers' premises. The inappropriate differences were discarded and the general operation mode differences were supplemented with further ERP-specific systemic differences. These case studies, moreover, provided more background information, allowing explanations or a better understanding of the contexts of the systemic differences to be found. In total, 15 interviews with 4 different ERP producers were conducted and transcribed.

The data collection was supplemented by document analysis (websites, informational material, pricing lists, internal documents, etc.), researcher's notes and real artifacts (ERP systems, test accounts, instructional videos; c.f. 1.8, 1.8, 1.8). These case data were analyzed using selective and open coding and were structured into a contrasting meta-matrix where operation-mode specific explanations and contextual information had been assigned to the respective systemic difference criteria 1.8, 1.8. The meta-matrices for each case had been condensed to an aggregate contrasting meta-matrix; the most important systemic differences are available in 1.8.

The main meta-matrix, which contains all identified ERP operation mode differences, was used as the starting point for constructing the option calculation scheme. All classified systemic differences can be assigned as an advantage of either SaaS- or On-Premise-ERP, which may reduce the overall cost of the ERP system in a particular case 1.8. This previous research enables a calculation scheme to be built using differential calculation methods, thereby eliminating absolute cost raising, which is usually perceived as too time-consuming.

2.2.2 Option calculation scheme construction method.

According to vom Brocke et al. alternative process designs can be compared in two different ways: using a total or a differential calculation 1.8. In the total calculation, each payment amount is assessed and calculated independently, whereas in the differential calculation only the additional payments that are relevant to the comparison of two alternatives are considered. In principle the latter approach could be transferred to the comparison of ERP operation modes, even though the ERP operation modes are not themselves process designs. The operation modes have an impact on the business processes and their valuation, e. g., maintenance cost differences or assessment of the flexibility and adaptability to the lived processes. So there is some evidence supporting the use of this approach for comparing ERP operation modes. The calculation method of vom Brocke et al. is based on the assessment of Event-driven Process Chains at the operational level, Visualization of Financial Implications (VOFI) at the budgeting level and the Total Cost of Ownership (TCO) and Return on Investment (ROI) at the corporate level 1.8. This 3-stage model, which is strictly aligned to evaluating process designs, can be simplified when used to evaluate operation modes. On the one hand, several cost factors are exogenous

and therefore already known on the corporate level, e. g. the license or the subscription costs. So no further investigation of these cost factors will be necessary on the operational and budgeting level. On the other hand, ERP systems that are the same from a functional perspective have the same supporting functions; therefore when comparing such systems, one can estimate that the time and money savings will be the same. The in-payments from applying the ERP system, defined as savings of money and time, can therefore be excluded from the TCO balance, when comparing using the differential calculation. In the exceptional case of the SaaS- and On-Premise-ERP systems not being identical, e. g., with SAP Business by Design vs. SAP Business One, then the user must investigate all the differential functional in-payment factors as well and should include them as in-payment amounts in the cost calculation scheme.

In contrast, internal costs to keep the ERP system operational, e. g. maintenance or updating, are typically not directly known at the corporate level. These internal costs need to be registered at the operational level and have to be budgeted for each year to calculate the total costs over the whole expected lifetime of the ERP system.

Measuring the gains in flexibility typically associated with SaaS in the event that the ERP system must be adapted to a changed situation is more difficult. These flexibility gains will indeed arise insofar as optimal resource management is possible, with costs minimized by renting neither too many nor too few modules, user accounts or infrastructure capacity. Further, the effect size with flexibility gains is directly dependent on the probability or frequency of the expected business change 1.8. The cost amount in the calculation scheme therefore has to change depending on how often the business will change and how much flexibility is required. The respective probabilities therefore must be estimated in advance for each year and budgeted over the whole expected ERP lifetime.

The method applied in constructing the operation mode option calculation scheme will be a differential total cost of ownership approach at the corporate level, which refers to data similar to that investigated in the first part with regards to the systemic differences of ERP operation modes. The operation mode option calculation scheme must include all the differential costs of acquisition, operation, and use. Ellram 1.8 and Ellram and Siferd 1.8 declare the TCO method to be best suited, inter alia, for outsourcing decisions and supplier selections, especially when a high monetary value is at stake, although Ellram 1.8 declared that the main difficulty of the applied method is the complexity of taking all costs into account for the respective object. The applied differential approach eliminates exactly this difficulty by looking only at all cost factors which differ between the two operation mode options. This reduces the complexity considerably and matches best with the present study's aim of determining the operation mode cost differences in a particular ERP case. Furthermore, this method enables ERP-selecting end users to compute operation mode related cost differences with little effort and without much prior know-how.

3. Option Calculation Scheme

The option calculation scheme is limited to general differences identified by the literal replication logic 1.8. So neither producer-specific differences, nor customer-specific characteristics that induce specific needs are considered here. When applying this scheme, the user should add in any additional case-specific differentiating out-payments, especially with respect to functional differences, before calculating the results as described in the next section. As already mentioned, the option calculation scheme is intended to compare all out-payments exceeding those of the respective other operation mode, so no absolute option scheme cost calculation is provided here.

The option calculation scheme is subdivided into the three following subsections: First, the general nonrecurring differentiating costs are presented; then, the recurring differentiating cost factors are described. The last subsection looks at event-driven differentiating cost factors.

3.1 Initial and Nonrecurring Costs of ERP Operation Mode Options

License and hardware costs: The most obvious and important difference between SaaS- and On-Premise-ERP is that with SaaS, no instance is installed and maintained by the customer. This has the fundamental consequence that with SaaS, no licenses have to be bought, nor does hardware have to be provided (cf. 1.8, 1.8, 1.8, 1.8, 1.8). That means, no initial ERP and operating system installation costs arise in a SaaS-ERP and no IT professionals have to be hired to install or maintain the systems (cf. 1.8, 1.8, 1.8, 1.8, 1.8). The operation and provisioning of the SaaS-system is settled by a monthly subscription fee, allowing fixed costs to be replaced by variable recurring costs. In contrast, On-Premise offers entail high initial and nonrecurring license and hardware costs.

Preliminary project costs: The selection, initiation and implementation stages seem to be similar for the two operation mode options, with minor differences. The similarities arise from the fact that SaaS solutions are not less complex than On-Premise solutions and both systems must be implemented to reproduce and comply with the company's business processes 1.8, 1.8, 1.8. A preliminary project therefore has to be carried out for SaaS as well. But the scope of the preliminary project may be smaller in a SaaS-ERP than in an On-Premise. To check the functionalities of the ERP system, SaaS customers needs only to open a test account to try the features of the ERP

system, and have all their questions about the features and functions answered immediately; this facilitates the process of selecting all required modules. However, if the systems are identical, the SaaS test account with its advantages of immediate access and the possibility of testing the ERP functionalities may also be used by those selecting On-Premise.

After the selection of the ERP system, SaaS requires no installation, but only the creation of an account. Furthermore, SaaS is preconfigured and typically has self-configuration mechanisms enabling the user to modify the configurations of the system himself, e. g. with a wizard or with user-friendly tables. While this does require learning how to change the configurations, it increases the flexibility and reduces both the service costs and the dependence on the ERP partner. In an On-Premise system, the partner has to understand the business needs first, to get to know exactly what settings have to be changed. There will be some hermeneutic circling between the On-Premise ERP customer and the ERP partner before the configuration settings match the needs of the customer exactly. Hence preliminary project costs and time expenditures may be higher with On-Premise-ERPs than with SaaS, meaning that the same cost categories arise with an On-Premise-ERP as with a SaaS, but not the same costs. All categories, with their respective costs if known or estimable, should be included in the option calculation with respect to module selection, configuration, data migration and costs of conducting a pilot. Hence, ERP selecting customers are strongly encouraged to ask their ERP partners for the respective preliminary project costs and time expenses for each ERP operation mode. This sensitizes the customer to the respective cost factors and may help to fix the right price for the preliminary project service. If the prices for the preliminary project are fixed at the same amount for both ERP operation modes, then the customer can delete these costs from the option calculation scheme.

Training: Training is required for any ERP system[1], irrespective of the operation modes (cf. 1.8). Any expenditures will be directly dependent on the learning forms provided. The singular instance approach of SaaS facilitates the implementation of self-learning sections such as Web learning directly into the ERP system. These new learning methods and self-explanatory mechanisms in SaaS-ERP systems may reduce the costs for conventional classroom learning lessons or even make them obsolete. Additionally, the new learning methods enable immediate training, while classroom learning sessions must be scheduled in advance. Similar mechanisms might also be available in the future for On-Premise ERPs. The trainee's effort will often be similar irrespective of the training concept. The ERP implementing customer should ask his partner which different training concepts exist and what costs they entail.

Customization: If the ERP system has to be customized, there will be different cost factors to be taken into account in the cost calculation. SaaS has only limited customizability compared to On-Premise, because SaaS-ERPs can only be operated in the standard. Thus, the adaptability of SaaS-ERP is limited to the configuration options and interfaces that have been thought of in advance 1.8, 1.8. Further changes can only be made by the ERP provider when including the changes to the standard at an agreed price or with interconnected third-party systems, which must be integrated to the adaption and translation layers, if any. On the other hand, almost everything can be modified in an On-Premise-ERP by changing the source code, but changing the source code requires much know-how and time, and also incurs high costs. Moreover, the customized part is not always supported, especially when major adaptations are carried out. Customization may result in higher maintenance costs than with the SaaS-ERP working on the standard. So SaaS enjoys the advantage of a more reliable system and a guaranteed maintainability in the future, as well as lower customization cost, but lacks unlimited customization capability. Additionally, extensive configuration options are very complex; the customer will need guidance from the ERP partner. SaaS is therefore best suited for low-level customization, whereas a customer who needs a highly specific ERP program is, according to the transaction cost theory, best served by On-Premise 1.8, 1.8. Taking this into account, customization cost differences are very difficult to estimate, because the real costs are rarely foreseeable. In SaaS all estimated wages and service costs for configuring the system, for the ERP provider to adapt the standard and for programming the interface- or third-party system should be considered in the option calculation. Customizing an On-Premise ERP may incur programming and adaption costs to change the source code, to integrate the new function into the system, to test the section and adapt the interfaces. All wages of the internal personnel and all service costs should be considered. The service costs for all adaptations and programming services should be negotiated with the ERP partner; requesting a quote and fixing the price for the necessary adaptations are strongly recommended. With the quotes in hand, the customer should compare the additional costs of the adaptations with the additional recurring costs due to a higher labor effort in case of non-customization (see recurring cost section). These additional labor costs will add up to a significant amount over the ERP's lifetime, if the lack of customization means that important internal processes are

1 Smith et al. categorized training as operations costs, which certainly seems to be evident in the case of general IT, where new software or updates require training each year 1.8. This does not occur in the same way with respect to ERP systems, where the operating principles will be kept constant over the years in most cases. Some recurring training costs arise when new employees are trained and occasionally a new release makes it necessary to retrain the employees. These costs may be included in the next sections as recurring training costs.

misaligned. The investigation of all these customization options and their concomitant costs is necessary to the development of the right customization strategy.

for the internal operation and maintenance. Unfortunately, most of the companies studied for this paper do not separate the ERP from the other IT costs. One way to overcome

SaaS-ERP					On-Premise-ERP			
Class Cost Factors	occur.	Depend. Variable	Exemplary Types	(N)RC	Cost Factors	occur.	Depend. Variable	Exemplary Types	(N)RC
Initiation- and Nonrecurring Costs (NRC)									
ERP-System Costs									
					License:				
					licenses			ERP-system, operating system, database	NRC
					ERP Installation Costs:				
					setup		hours	wages, service	NRC
					installation		hours	wages, service	NRC
Operation, Hardware and Software Maintenance, Updates									
					Internal IT Professionals / 3rd-Party Professional IT Service:				
					hardware deployment		hours	wages, service	NRC
					software deployment		hours	wages, service	NRC
					Infrastructure Costs:				
					hardware provisioning		hours	server, network, etc.; wages, service	NRC
					software provisioning		hours	wages, service	NRC
					hosting, firewalls, etc.		hours	wages, service	NRC
Initiation and Implementation									
Preliminary Project Costs:					**Preliminary Project Costs:**				
selection of required modules		hours	wages	NRC	selection of required modules		hours	wages	NRC
configuration		hours	wages, service	NRC	configuration		hours	wages, service	NRC
data migration		hours	wages, service	NRC	data migration		hours	wages, service	NRC
costs of conducting a pilot					costs of conducting a pilot				
Training Costs:					**Training Costs:**				
training		hours	wages, service	NRC	training		hours	wages, service	NRC
Customization, Configurability and Adaption									
Individualization Cost:					**Individualization Cost**				
interface programming		hours	3rd party program change; wages, service	NRC	programming		hours	wages, service	NRC
adaption fee standard SaaS-ERP				NRC	adaption		hours	integration, interface, etc.; wages, service	NRC
Σ NRC_{SaaS}					Σ $NRC_{On\text{-}Premise}$				

Table 1: Option calculation scheme; first part: nonrecurring costs

3.2 Variable ERP Operation Mode Costs

Maintenance fee vs. subscription: In addition to the higher initial and nonrecurring costs of an On-Premise-ERP system, recurring costs, which must not be underestimated, will arise as maintenance and updating fees 1.8, 1.8. Predictable maintenance fees of 12% to 22% in respect to the license costs have to be paid to the ERP producer each year. Furthermore, less predictable maintenance services or wages of internal IT professionals have to be taken into consideration. All costs or expenditures for maintaining, backing up, and updating the ERP system and the infrastructure have to be estimated and should be included in the calculation. Customers who are replacing their existing outdated or obsolescent ERP systems may separate the current ERP costs from the other IT costs, obtaining a clearer and more business-related comparison between the operation modes. Separating the internal ERP costs from the other IT helps to get a clear statement about the current and future costs this problem would be to estimate the proportion of the costs of the obsolescent ERP out of the total IT costs, without having a strict separation by using different accounts. The proportion may be estimated by recording all the work done for a short period of time and extrapolating for a longer period.

SaaS-ERP customers, in contrast, pay only a clear and definable subscription fee, which includes all maintenance, backup, and infrastructure costs. The subscription fee is often dependent on the number of users, sometimes with a required minimum; this simplifies the estimation of the amount to be paid. But the subscription costs may add up to a not-insignificant amount each year.

Service and support: In On-Premise-ERP systems, service contracts may include all additional services and support by the ERP partner up to an agreed limit (on top of the maintenance and update contract). This may reduce the service and support costs, which otherwise would be charged by effort. In a SaaS-ERP, a service contract is not necessary and the support

is always included, but may be limited as well. So different cost calculation values may be applicable, which should be considered in the calculation scheme.

Customization: The variable costs of customization are driven by the additional maintenance expenses that occur when the standard system is adapted. Especially when the system has to be updated, the individual source code may produce incompatibilities. So in On-Premise systems, further programming or problem-fixing costs may come along with the customization of the ERP system. The customer should therefore ask his ERP partner how often such incompatibilities typically arise, because the additional maintenance fee for the customized part will typically not cover

3.3 Change of Requirements and Stability of the ERP Systems

Comparing ERP operation modes make it necessary to contrast and evaluate the differences in flexibility. Most customers who face high volatility and who require fast changes, growth, or flexibility, or are project-driven, are best served by the SaaS operation mode. The flexibility may allow cost reductions in the future, so the customer should assess the value of the additional flexibility and should take these reductions into account in the option scheme calculation. But the key consideration with respect to the cost calculation is that the costs of change arise only when the changes are put into effect. So each cri-

SaaS-ERP					On-Premise-ERP				
Class Cost Factors	occur.	Depend. Variable	Exemplary Types	(N)RC	Cost Factors	occur.	Depend. Variable	Exemplary Types	(N)RC
Recurring Cost (RC)									
ERP-System Costs:									
Subscription Contract:					**Maintenance Contract:**				
subscription fee		user, space, other		RC	maintenance fee		%-age of license		RC
					Service Contract:				
					service fee		%-age of license		RC
					AND / OR				
					service		hours		RC
Operation, Hardware and Software Maintenance, Updates									
Support Costs:					**Internal IT Professionals / 3rd-Party Professional IT Service:**				
additional support service		hours	wages, service	RC	maintenance expense: operating system, firewalls, database, etc.		hours	update, upgrade; wages, service	RC
					Infrastructure Costs:				
					maintenance expense: hardware, network, ups, electricity, etc.		hours	extension, exchange, migration; wages, service	RC
					Support Costs in Addition to Service Contract:				
					additional support service		hours	wages, service	RC
					Backup Costs:				
					maintenance		hours	wages, service	RC
					hardware: extension / exchange		space	hard disks	RC
Customization, Configurability and Adaption									
Individualization Cost:					**Individualization Cost:**				
add. maintenance expense		hours	interface, 3rd party program	RC	add. maintenance expense		hours	customized sections	RC
AND / OR									
add. labor expense		hours	for non-optimal requirements	RC					
Σ RC_{SaaS}					Σ $RC_{On\text{-}Premise}$				

Table 2: Option calculation scheme; second part: recurring costs

major problems. Further, it is necessary to get information about the additional maintenance fee, the additional rates per hour for cases which are not covered by the maintenance contract and how much time is typically estimated to fix the major problems. This information makes calculating the average additional recurring costs of customization for On-Premise-ERP systems possible.

The limited customizability of SaaS-ERP, in contrast, leads to fewer special codes and thus to lower additional maintenance expenses. But sometimes essential customizations are not possible, making necessary a compromise that will lead to higher labor expenses. These expenses may be included in the calculation and have to be compared to the additional On-Premise maintenance costs.

terion mentioned in this section will be included in the cost calculation only with its respective probability of occurrence. The ERP customer should therefore estimate how likely each of these change criteria below is and can as a result omit all irrelevant criteria. Whether each cost is added to the recurring or the nonrecurring section can be seen in Table 3.

Functional and infrastructure change: On-Premise systems are generally more fixed, especially in the sense of the scalability of infrastructure and modules. Modules can be increased by paying the license and recurring maintenance fees for the additional module, but not decreased. The module increase sometimes comes along with a change of the package, necessitating the reinstallation and remigration of the ERP system, and incurring high costs of change. Further, increasing

modules triggers implementation costs, but as these are identical between the ERP operation modes, they are not considered here.

In the case of a module reduction only the maintenance contract may be reduced - if at all. But the option of reducing the maintenance contract to save money over a limited period is not made available. Most often, extra reactivation fees will be charged to catch up to the latest version. In a SaaS-ERP, modules can always be decreased or increased at least monthly among all the subscription bundles offered by the provider. Companies with a typically volatile business, e. g. seasonal or project-oriented, can therefore benefit most from SaaS-ERPs 1.8, 1.8. When modules are added, the subscription cost will be increased, but no additional costs are incurred relative to the On-Premise-ERP. But with the elimination of a module, data migration costs will accrue, because the reduction means that the data history of the module can no longer be accessed.

Peak-loads, capacity change and scalability: The load of an ERP system is directly dependent on the number of users and on the course of business. So when the business works well, the load of the ERP system will be higher. The On-Premise system always has to be aligned with the highest level of load, even when the loads will last for only a short time, e. g. Christmas season or during stocktaking. Exceeding the infrastructure limits will result in slow response times, which can only be overcome by expanding the infrastructure and migrating the system to the new infrastructure. In contrast to this scenario, SaaS has unlimited scalability, because a professional provider will have enough hardware available to serve their customers. Furthermore, peak-loads can be balanced, because customer peaks often arise at different times 1.8. Indeed, the more user accounts are rented, the more expensive the SaaS system is, but the number of user accounts can be rented on a monthly basis, with the unused user accounts reduced after the season is over. This frees up unused resources and reduces the cost to the customer. Customers who wish to change the number of users frequently could insert the average number of users instead in the subscription cost of the variable cost section.

Dependence on the provider: SaaS-ERP systems are generally more dependent on the ERP partner, especially on the provider 1.8, 1.8, 1.8, 1.8. Should the provider discontinue the service, whether the discontinuation is planned or unplanned

SaaS-ERP					On-Premise-ERP				
Class Cost Factors	occur.	Depend. Variable	Exemplary Types	(N)RC	Cost Factors	occur.	Depend. Variable	Exemplary Types	(N)RC
Change of Requirements and Stability									
Flexibility, Changeability									
Module Increase with $P_{ModuleIncrease(MI)}$:					**Module Increase with $P_{ModuleIncrease(MI)}$:**				
subscription fee increase	P_{MI} x	number of modules		RC	additional license costs	P_{MI} x	package increase		NRC
					installation costs	P_{MI} x	hours	wages, service	NRC
					system migration costs	P_{MI} x	hours	wages, service	NRC
Module Decrease with $P_{ModuleDecrease(DI)}$:					**Module Decrease with $P_{ModuleDecrease(MD)}$:**				
subscription fee decrease	P_{MD} x	number of modules		RC	maintenance contract cost reduction	P_{MD} x	package decrease		RC
data migration costs		hours	wages, service	NRC	reactivation fee	$P_{reactivation}$		extra fee for reactivation	NRC
Scalability with P_{Scale}:					**Scalability with P_{Scale}:**				
change of subscription fee	P_{Scale} x	user, space, etc.	increase, decrease	RC	infrastructure expansion	P_{Scale} x	limits {space, cpu, etc.}	server, hard disks, etc.	NRC
					implementation and migration	P_{Scale} x	hours	wages, service	NRC
Dependence and Stability									
Dependence on Provider with $P_{ProviderChange(PC)}$:					**Dependence on ERP Partner with $P_{ERPVersionChange(EVC)}$:**				
residual value {Σ NRC_{SaaS}}	P_{PC} x			NRC	continuous use without support:				
total cost of ERP exchange	P_{PC} x		Implementation cost, migration, etc.	NRC	Σ $RC_{On-Premise}$ - fees {maintenance, service,…} + estimated add. proceeding & maintenance costs	P_{EVC} x	hours	wages, service	RC
					OR				
					ERP-system change				
					residual value {Σ $NRC_{On-Premise}$}	P_{EVC} x			NRC
					total cost of ERP exchange	P_{EVC} x		Implementation cost, migration, etc.	NRC
Dependence on Internet Provider with $P_{Outage-n-Lines(OnL)}$:					**Dependence on Internet Provider with $P_{Outage-n-Lines(OnL)}$:**				
singular connection:					singular connection:				
Downtime cost	P_{O1L} x	time		NRC	Downtime cost {locations}	P_{O1L} x	time		NRC
OR					OR				
multiple connections:					multiple connections:				
Downtime cost	P_{O2L} x	time		NRC	Downtime cost {locations}	P_{O2L} x	time		NRC
Second internet line fee				RC	Second internet line fee				RC
Σ $NRC_{Change\ SaaS}$; Σ $RC_{Change\ SaaS}$					**Σ $NRC_{Change\ On-Premise}$; Σ $RC_{Change\ On-Premise}$**				

Table 3: Option calculation scheme; third part: change of requirements

(e. g. due to bankruptcy), then all the customer can do is to change the system and export the data to a common format (e. g. Excel tables or SQL DB). This incurs all the fixed costs of replacing an ERP.

The On-Premise system, on the other hand, may become outdated and therefore not supported anymore, but this does not imply an immediate requirement to change the system. The system can still be used without support from the ERP partner until it no longer meets the requirements of the accounting standards or the company's own needs. Proceeding with the outdated system saves all further contractual maintenance and service costs, but incurs further internal continuation and maintenance costs for the work that had been done by the provider before the system was outdated. Otherwise, in the case of a premature change, the residual value of the old system which is written off has to be considered in addition to all the fixed costs incurred in replacing the On-Premise system. A SaaS system, in contrast, will never be outdated and will always run on the newest version, because a service and not a system is sold. There is no need to hold new features back for new ERP software versions as in an On-Premise ERP. Just the opposite is true: the new feature can only be rented, e. g. as a new module, when it has been made available in the SaaS-ERP system.

So, on one side, there is the replacement of the SaaS system if it is discontinued and on the other side there is the On-Premise becoming outdated, with two options: proceeding with the system or writing it off and replacing it. In both operation modes, any incurring costs may be estimated and multiplied by the respective probability of their occurrence. A risk assessment of the discontinuation or the outdating may help to get a more precise estimation of the likelihood of their occurrence.

Dependence on the internet: SaaS-ERP is, in addition, more dependent on the internet provider than an On-Premise-ERP, causing higher downtime costs when the internet service is interrupted. Highly downtime-critical customers can compensate by using multiple connection lines (fixed or mobile) from different providers. This diminishes the probability of downtime, but incurs the extra costs of two internet lines. On-Premise-ERP is only dependent on the internet for offices or plants outside its premises, reducing the internet downtime costs of the main site to zero. So when comparing SaaS- vs. On-Premise-ERP, the difference in downtime costs as triggered by an internet outage should be included, provided that the downtime costs are high, internet outages happen, and the probability of internet outage is known.

This scheme does not claim to be complete, especially because the criteria mentioned here result from general differences between SaaS and On-Premise. Many producer-specific differences between the operation modes may exist that are not captured here. In particular, to consider all differentiating factors, whenever this option scheme is applied, it should be supplemented by any situational specific differences and by any missing criteria. When applying a cost comparison between different operation modes, e. g. between ASP and SaaS, then the same method of comparison can be applied, but diverse systemic differences may be relevant. Therefore different costs have to be assessed and estimated with different operation modes before the calculation in the next section can be conducted.

4. Calculating the Differential Total ERP Costs

In principle, there are diverse methods of calculating total costs and each method has its advantages and drawbacks. This section uses a more process-related perspective based on the work of vom Brocke & Simons et al., which facilitates the variation of depreciation periods and interest rates, as well as enabling the inclusion of imputed interests 1.8. To get to the total cost, all recurring costs and all yearly depreciation values of the nonrecurring costs will be summed up. In addition to this, interest costs for financing the current residual values of the nonrecurring costs by debts or equity are added. When applying the total cost approach using the absolute costs, no distortion arises from calculating the interest costs. But this is not the case when using the differential TCO approach, except for the case where the same proportions of the nonrecurring costs are financed by equity and debts as in the absolute total cost approach. Hence, to avoid distorting the interest cost amounts, a debt-equity financing ratio that funds the nonrecurring costs by debts and equity in the same relative proportion as in the absolute calculation will be applied here. The ratios will most likely be different for each operation mode.

The yearly differential TCO is calculated by adding up the recurring costs with the nonrecurring costs, which will be distributed by linear depreciation. This will be achieved by discounting the nonrecurring costs over the average ERP lifetime; the residual value, which is the basis for the yearly interest, will decrease each year by an nth part per year[2]. Secondly, the interest is calculated on the basis of the residual values and is assigned to the respective yearly recurring costs. Last but not least, all yearly total costs until the end of the average ERP lifetime have to be added up to obtain the differential total ERP operation mode costs.

4.1 Yearly Nonrecurring Cost Proportion and Yearly Recurring Costs

The recurring and nonrecurring costs identified in the previous chapter should be used to compare the ERP operation

2 n is the average expected ERP lifetime in years.

mode options. This section calculates the relative additional costs of each operation mode, but the same method could be applied using the absolute total costs. To get the yearly operation mode option costs the following parameters are used:

$\Sigma NRCT = \Sigma NRC + \Sigma NRC_{Change}$	differential add. nonrecurring costs (for each operation mode)
$\Sigma RCT = \Sigma RC + \Sigma RC_{Change}$	differential add. recurring costs (for each operation mode)
EFI	interest rate of equity financing
DFI	interest rate of debt financing
DFr	ratio of debt financing (percentage of all costs financed by debts)
n	depreciation period; ERP lifetime

The total additional recurring costs of each operation mode for each year can be determined by summing up the respective recurring costs. The nonrecurring costs should be depreciated over the whole lifetime of the ERP system, to get the yearly nonrecurring cost proportion (1). Interest rates of the residual nonrecurring cost value will be incurred, along with the depreciation amount to correct for the financing aspect of the nonrecurring cost residuals. The nonrecurring cost amount may be financed by equity or debt, but often a mixture of both financing forms is used. The relative cost accounting does not cover all costs, so it is necessary to keep the equity to debt financing ratio equal to finance the same proportions by equity and debt as in the absolute total cost accounting (2), (3):

Yearly expense of depreciation: (1)

$$dNRCT = \Sigma NRCT / n$$

Interest financed by equity[3]: (2)

$$I_{E\,Y(x)} = EF_I \bullet \Sigma NRC_T \bullet \left(1 - \frac{(x-1)}{n}\right)$$

Additional debt financing costs: (3)

$$I_{aDFY(x)} = (DF_I - EF_I) \bullet \Sigma NRC_T \bullet DF_r \bullet \left(1 - \frac{(x-1)}{n}\right)$$

4.2 Yearly Total Costs

The yearly relative total cost amount for each operation mode can be obtained by adding up the recurring costs with the nonrecurring cost proportion (depreciation expense) and the interest:

Total relative cost amount for the year Y(x): (4)

$$TCcomp.Y_{(x)} = \Sigma RC_T + dNRC_T + I_{E\,Y(x)} + Ia_{DF\,Y(x)}$$

In principle this formula can be applied to calculate the relative cost difference for each particular year and operation mode. Using the yearly total costs makes it possible for the user to change the interest rates or the ratio of debt financing for each year. The debt redemption is often faster than the depreciation period, so the ratio of debt financing may diminish from year to year until the whole ERP system is financed by equity. In this case, a different ratio of financing ($DF_{r\,Y(x\,/\,x=\{1,2,\ldots,n\})}$) has to be applied for each year. Furthermore, interest rates, especially for debt financing, can vary over the depreciation period and have to be adjusted. Recurring or nonrecurring costs may change for each year as well. When nonrecurring costs are varied, then the depreciation fixed cost proportion has to be adapted to the new situation. The formula (1), which is the basis for interest calculation in (2) and (3), changes for the remaining years to:

$$dNRC\_(T\ (new)) = \left(\frac{(n-x)\ \cdot \Sigma NRC\rfloor\_T}{n} + \Sigma NRCChange(new)\right)/(n-x) \quad (5)$$

The changes in nonrecurring and recurring costs, as well as interest rates can rarely be foreseen, so in the option calculation the real costs must be estimated. Thus it often makes the most sense to keep as much as possible constant and to adjust these costs only when the predicted course of progression requires adaption to the real situation in the future. Therefore all the parameters will be kept constant in the next section, although the ratio of debt financing, which can be predicted in advance, e. g. with a redemption plan or by the expiration of the loan, could vary.

4.3 Total Comparative Operation Mode ERP Costs

The comparative total costs over the whole depreciation period have to be calculated to compare the options between the ERP operation modes. The final result tells the user which of the two operation modes is cheaper. To get the relative total costs per operation mode over the whole period, one needs only to sum up the yearly total costs identified in the last section. So the following formula can be stated:

3 Total amount financed by equity. Because DFI > EFI always, the total amount can be added when only the additional interest (DFI-EFI) is used for the additional debt finance costs. See (3).

$$TCcomp.Y_{(x)} = \sum_{x=1}^{n} \quad (6)$$

$$= \sum_{x=1}^{n} (\Sigma RCT + \frac{\Sigma NRC_T}{n} + IEY_{(x)} + IaDF\,Y_{(X)})$$

$$= n \bullet \Sigma RC_T + \Sigma NRC_T + \sum_{x=1}^{n} (I_{EY(x)} + I_{aDF\,Y(X)}) \quad *$$

*{ΣRCT; ΣNRCT = constant}

This formula cannot be simplified in the case of variable interest rates, variation in nonrecurring or recurring costs, or a change in the ratio of debt financing. Each year has to be calculated by its respective parameters, so that the results of each year can be summed up to the comparative total costs.

In the special case where all parameters are presumed to be constant, and the ratio of debt financing is kept constant until the end of the depreciation period, then the sum of the interest amounts can be rearranged by the Gaussian sum formula:[4]

$$\sum_{x=1}^{n} I_{EY(x)} = EF_I \bullet \sum_{x=1}^{n} \Sigma NRC_T \bullet \left(1 - \frac{(x-1)}{n}\right) \quad (7)$$

$$= EF_I \bullet \Sigma NRC_T \bullet \left(\frac{n+1}{2}\right)$$

$$\sum_{x=1}^{n} I_{aDFY(X)} =$$

$$= (DF_I - EF_I) \bullet \Sigma NRC_T \bullet DF_r \bullet \sum_{x=1}^{n} \left(1 - \frac{(x-1)}{n}\right) \quad (8)$$

$$= (DF_I - EF_I) \bullet \Sigma NRC_T \bullet DF_r \bullet \left(\frac{n+1}{2}\right)$$

This allows a rearrangement and simplification of the formula in (6):

$$\sum_{x=1}^{n} T_{Ccomp.Y(x)} = \quad (9)$$

$$= \Sigma NRC_T \bullet \left(1 + \frac{(n \bullet \Sigma RC_T)}{\Sigma NRC_T} + \left(\frac{n+1}{2}\right)\right)$$

$$\bullet (EF_I + DF_R \bullet (DF_I - EF_I)$$

*{ΣRCT; ΣNRCT; EFI; DFI; DFR = constant}

5. Cost Comparison: Management Strategies and Conclusions

Functional selection first: The selection of ERP systems does not stop at functional criteria, especially when several ERP operation mode options are available. It is evident that the leading ERP candidate should be selected on the basis of an evaluation of the ERP candidates' coverage of the functional requirements (cf. 1.8), because the ERP's embedded structures have to comply with the organization's embedded structures, so the ERP system has to be aligned to the institutional context of the company 1.8. Given a choice between a SaaS- and On-Premise-ERP system of the same preferred ERP candidate the main question, answered in this contribution, comes into effect: which of these operation mode options is the best choice from a cost perspective in the long run in this particular case? This answer can be obtained through an option calculation scheme, which is not reduced purely to SaaS subscription costs. Diverse fixed and variable costs, which arise during the implementation and operation of the ERP system, were uncovered.

Internal ERP costs: The first step toward an option calculation scheme is to look at the current internal ERP costs, if the selecting ERP customer has one that is to be replaced. With this procedure the selecting customer is able to estimate all the internal maintenance and updating costs by listing all expenditures in a non-aggregated manner. Another strategy to get more precise cost estimations for the ERP system operation is to visit one of the preferred EPR partner's customers, especially when the company has never had an ERP system before. During the visit, the customer may ask about the internal maintenance costs and the time spent in keeping the system operational. This information may help the customer get an idea of the cost dimensions. Further internal costs to be collected are the cost of the hardware, server, and server software, as well as the backup system, especially if they are to be renewed as well.

ERP and ERP integration costs: When all the internal costs of an On-Premise system are clear, a meeting should be arranged with the currently preferred ERP partner. In the meeting, the ERP partner should clearly explain the cost differences between the two ERP operation modes, emphasizing the license costs, maintenance fees, SaaS subscription costs, training costs, and implementation or preliminary project costs. Further discussion points should be the implications of customizations on future updates and maintenance and their concomitant costs, as well as the flexibility of each operation mode in light of changing requirements: module or user extension or reduction, bundles and scaling, minimum number of user accounts, waiting time before the next change period, etc. The customer should know as precisely as possible what

4 The proof for this rearrangement is given in the Annex.

information he needs from the future ERP partner to complete his option calculation scheme so that the comparison between the SaaS and On-Premise operation mode can proceed by calculating the relative cost difference.

Calculation: The developed calculation method enables ERP customers to identify the less expensive operation mode option by inserting the required numbers. Given the most important systemic differences between ERP operation modes as noted in this contribution, a method for calculating them has been demonstrated. The criteria mentioned here act as a guide and can open a discussion as to whether SaaS or On-Premise should be selected on financial grounds, although this contribution does not support the selecting customers with standardized cost estimates. The option calculation scheme sensitizes the ERP customer to all the cost differences, which can be seen as a good basis for negotiation between the ERP customer and the ERP partner to mitigate the costs for the preferred operation mode.

Besides calculating the financial advantage of the preferable operation mode, diverse soft and non-monetary factors should be included to align the more advantageous systemic difference criteria to the particular company characteristics. These diverse soft and non-monetary factors, as well as the cost comparison results are at least necessary to develop an overall ERP delivery strategy. Further information about the more qualitative operation mode selection is available in 1.8.

6. Limitations and Outlook

A method should serve a specific purpose, opting for advantages and disadvantages with the method selection. This method was aimed at ERP selecting end users who have little prior know-how and few human resources. The method is the only one directly aligned to this problem, but is not the only way of calculating ERP operation mode differences. Most of the investment calculation methods, e. g. Net Present Value or Discounted Cash Flow, could be applied as well. These investment methods are based on an absolute comparison, taking all costs into consideration. Using these methods for calculating the ERP operation mode cost differences may result in more accurate and better comparable results, but they are far more expensive and complex to apply. Hence, the main limitations of the proposed option calculation scheme come principally out of the method used. The differential cost approach enables comparisons only of pairs, not of third options. The exclusion of all the similarities carries the risk of overlooking company-specific differences, which are not included in this proposed general calculation scheme. And last but not least, the biggest difficulty remains the often time-consuming assessment or estimation of each single cost factor discovered here as a systemic difference between the operation modes 1.8. The lack of accessible data is one major disadvantage of the total cost of ownership approach 1.8. Neither the total costs nor their components can be determined in general, because the amounts depend on the company characteristics and requirements and therefore differ from case to case. The better the costs can be assessed, the more accurate results this cost calculation scheme can be expected to provide. Data from the experience of the ERP partner or customers working on the same system could help to fill this gap, but to overcome this weakness, research to find generalizable cost amounts in relation to the company's characteristics is also encouraged.

Furthermore, the total cost of ownership method is not the best method to assess the costs and advantages of flexibility, imputed risks and gains, because the farther in the future the probability estimates are predicted, the rougher the assessments are. The method can therefore provide a feeling about the cost advantage, especially about the value of greater flexibility, but the real cost advantage may not be determined accurately with this method. Non-monetary risk assessment methods may be perhaps more constructive. Further non-monetary criteria such as ubiquity or collaborative gains are also excluded from the cost calculation scheme. Hence, to fill this gap, other methods are required, especially to be able to construct the qualitative part of the ERP delivery strategy, e. g. 1.8.

Another limitation of the calculation method is that the result is not tax-adjusted. An On-Premise system lengthens the balance sheet with the purchase and has thereafter to be taxed as an asset; this is not true for a SaaS-ERP. This cost difference was not considered due to multiple tax laws and rates of each state.

As mentioned, further research has to be qualitative, capturing not only the systemic differences but also the criteria, needs and requirements of the ERP customers by looking at the company's characteristics. Since each customer will benefit from the systemic ERP operation mode differences only in relation to their company's characteristics, the systemic differences should be compared with the company characteristics to build general claims about the situations in which each ERP operation mode is preferable. Additionally, this option calculation scheme, as well as all further qualitative findings and criteria will be applied to a concrete case, which will allow this scheme to be validated and refined. Any adjustments and refinements after the appliance are possible; the findings of the case study will form the basis for a system that will provide evidence supporting a decision between SaaS- and On-Premise-ERP on both strategic and financial grounds. For this purpose, more research and several further publications will be required.

Keywords
SaaS-ERP; On-Premise-ERP; Differences between SaaS and On-Premise; Calculation Scheme of IT-Options

References

[1] Benbasat, I., Goldstein, D.K. and Mead, M. (1987), "The Case Research Strategy in Studies of Information Systems", MIS Quarterly, Vol. 11 No. 3, pp. 369-386.

[2] Benlian, A., Hess, T. and Buxmann, P. (2009), "Drivers of SaaS-Adoption – An Empirical Study of Different Application Types", Business and Information Systems Engineering, Vol. 1 No. 5, pp. 357-369.

[3] Buxmann, P., Hess, T. and Lehmann, S. (2008), "Software as a Service", Wirtschaftsinformatik, Vol. 50 No. 6, pp. 500-503.

[4] Choudhary, V. (2007), "Comparison of Software Quality under Perpetual Licensing and Software as a Service", Journal of Management Information Systems, Vol. 24 No. 2, pp. 141-165.

[5] Cusumano, M.A. (2007), "The changing labyrinth of software pricing", Communications of the ACM, Vol. 50 No. 7, pp. 19-22.

[6] Darke, P., Shanks, G. and Broadbent, M. (1998), "Successfully completing case study research: Combining rigour, relevance and pragmatism", Information Systems Journal, Vol. 8 No. 4, pp. 273-289.

[7] Dubé, L. and Paré, G. (2003), "Rigor in Information Systems Positivist Case Research: Current Practices, Trends, and Recommendations", MIS Quarterly, Vol. 27 No. 4, pp. 597-635.

[8] Ellram, L.M., and Siferd, S.P. (1998), "Total Cost of Ownership: A Key Concept in Strategic Cost Management Decisions", Journal of Business Logistics, Vol. 19 No. 1, pp. 55-84.

[9] Ellram, L.M. (1995), "Total cost of ownership", International Journal of Physical Distribution and Logistics Management, Vol. 25 No. 8-9, pp. 4-23.

[10] Ellram, L.M. (1994), "A taxonomy of total cost of ownership models", Journal of Business Logistics, Vol. 15 No. 1, pp. 171-191.

[11] Farah, P. (2010), "Cloud computing or software as a service – which makes the most sense for HR?", Employment Relations Today (Wiley), Vol. 36 No. 4, pp. 31-37.

[12] Fulford, R. and Love, P.E. (2004), "Propagation of an alternative enterprise service application adoption model", Industrial Management and Data Systems, Vol. 104 No. 6, pp. 450-456.

[13] Gephart, R.P. (2004), "Qualitative Research and the Academy of Management Journal", Academy of Management Journal, Vol. 47 No. 4, pp. 454-462.

[14] Gill, R. (2011), "Why Cloud Computing Matters to Finance", Strategic Finance, Vol. 92 No. 7, pp. 43-47.

[15] Höss, O., Weisbecker, A. and Spath, D. (2009), "Software as a Service - Potenziale, Risiken und Trends", Information Management and Consulting, Vol. 23 No. 4, pp. 6-11.

[16] Katzan Jr., H. and Dowling, W.A. (2010), "Software-As-A-Service Economics", The Review of Business Information Systems, Vol. 14 No. 1, pp. 27-37.

[17] Lehmann, S., Draisbach, T., Koll, C., Buxmann, H. and Diefenbach, P. (2010), "SaaS-Preisgestaltung: Bestehende Preismodelle im Überblick", in: Benlian, A., Hess, T., Buxmann, P. (Eds.), Software-as-a-Service, Gabler, Wiesbaden, pp. 155-169.

[18] Link, B. (2013), "Considering the Company's Characteristics in Choosing between SaaS vs. On-Premise-ERPs", Wirtschaftsinformatik Proceedings 2013, pp.°261-277.

[19] Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J. and Ghalsasi , A. (2010), "Cloud Computing – The Business Perspective", Decision Support Systems, Vol. 51 No. 1, pp. 176-189.

[20] Mathew, M. and Nair, S. (2010), "Pricing SaaS Models: Perceptions of Business Service Providers and Clients", Journal of Services Research, Vol. 10 No. 1, pp. 51-68.

[21] Matros, R., Rietze, C. & Eymann, T. (2010), "SaaS und Unternehmenserfolg: Erfolgskategorien für die Praxis", in: Benlian, A., Hess, T., Buxmann, P. (Eds.), Software-as-a-Service, Gabler, Wiesbaden, pp. 239-254.

[22] Miles, M.B. and Huberman, A.M. (1994), Qualitative Data Analysis. An Expanded Sourcebook (2nd ed.), Sage Publications, Thousand Oaks, London, New Delhi.

[23] Palanisamy, R., Verville, J., Bernadas, C. and Taskin, N. (2010), "An empirical study on the influences on the acquisition of enterprise software decisions: A practitioner's perspective", Journal of Enterprise Information Management, Vol. 23 No. 5, pp. 610-639.

[24] Poppo, L. and Zenger, T. (1998), "Testing Alternative Theories of the Firm: Transaction Cost, Knowledge-Based, and Measurement Explanations for Make-or-Buy Decisions in Information Services", Strategic Management Journal, Vol. 19 No. 9, pp. 853-877.

[25] Pozzebon, M. and Pinsonneault, A. (2005), "Global–local negotiations for implementing configurable packages: The power of initial organizational decisions", Journal of Strategic Information Systems, Vol. 14 No. 2, pp. 121-145.

[26] Ragowsky, A., and Somers, T. M. (2002), "Special Section - Enterprise Resource Planning", Journal of Management Information Systems, Vol. 19 No. 1, pp. 11-15.

[27] Rai, A. and Sambamurthy, V. (2006), "The Growth of Interest in Services Management: Opportunities for Information Systems Scholars", Information Systems Research, Vol. 17 No. 4, pp. 327-331.

[28] Sääksjärvi, M., Lassila, A. and Nordström, H. (2005), "Evaluating the software as a service business model: From CPU time-sharing to online innovation sharing", Perspective 2000, pp. 177-186.

[29] Sammon, D., Adam, F. and Carlsson, S. (2009), "CSF Relationships in ERP Project Implementations: A Collective Case Study", in AMCIS 2009 Proceedings, paper 744.

[30] Smith, J., Schuff, D., and Louis, R.T. (2002), "Managing Your IT Total Cost of Ownership", Communications of the ACM, Vol. 45 No. 1, pp. 101-106.

[31] Soh, C. and Sia, S.K. (2004), "An institutional perspective on sources of ERP package–organisation misalignments", Journal of Strategic Information Systems, Vol. 13 No. 4, pp. 375-397.

[32] Sontow, K. and Kleinert, A. (2010a), "ERP-as-a-Service zwischen Euphorie und Skepsis", Industrial Engineering, Vol. 3, pp. 10-18.

[33] Sontow, K. and Kleinert, A. (2010b), "Software as a Service – Die schlanke Zukunft für ERP-Lösungen? Ergebnisse eine Anwenderbefragung", ERP Management, Vol. 6 No. 4, pp. 24-27.

[34] Strauss, A.L. and Corbin, J. (1998), Basics of Qualitative Research: Procedures and Techniques for Developing Grounded Theory (2nd ed.), Sage Publications, Thousand Oaks, London, New Delhi.

[35] Thomas, O. and vom Brocke, J. (2009), "A value-driven approach to the design of service-oriented information systems - making use of conceptual models", Information Systems and E-Business Management, Vol. 8 No. 1, pp. 67-97.

[36] Umble, E.J., Haft, R.R. and Umble, M.M. (2003), "Enterprise resource planning: Implementation procedures and critical success factors", European Journal of Operational Research, Vol. 146 No. 2, pp. 241-257.

[37] vom Brocke, J., Recker, J.C. and Mendling, J. (2010), "Value-oriented process modeling: integrating financial perspectives into business process redesign", Business Process Management Journal, Vol. 16 No. 2, pp. 333-356.

[38] Vom Brocke, J., Simons, A., Niehaves, B., Riemer, K., Plattfaut, R. and Cleven, A. (2009), "Reconstructing the Giant: On the Importance of Rigour in Documenting the Literature Search Process", 17th European Conference on Information Systems, pp. 3226-3238.

[39] vom Brocke, J., Sonnenberg, C. and Simons, A. (2009), "Value-oriented Information Systems Design: The Concept of Potentials Modeling and its Application to Service-oriented Architectures", Information Systems and E-Business Management, Vol. 8 No. 3, pp. 223-233.

[40] vom Brocke, J. (2007), "Service Portfolio Measurement: Evaluating Financial Performance of Service-Oriented Business Processes", International Journal of Web Services Research, Vol. 4 No. 2, pp. 1-32.

[41] Waters, B. (2005), "Software as a Service: A look at the customer benefits", Journal of Digital Asset Management, Vol. 1 No. 1, pp. 32-39.

[42] Watjatrakul, B. (2005), "Determinants of IS sourcing decisions: A comparative study of transaction cost theory versus the resource-based view", Journal of Strategic Information Systems, Vol. 14 No. 4, pp. 389-415.

[43] Weiping, L. (2009), "An analysis of new features for workflow system in the SaaS software", in ICIS 09 Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human, pp. 110-114.

[44] Xin, M. and Levina, N. (2008), "Software-as-a-Service Model: Elaborating Client-Side Adoption Factors", in ICIS 2008 Proceedings, paper 86.

[45] Yin, R.K. (2003), Case Study Research, Design and Methods, Sage Publications, Thousand Oaks, London, New Delhi.

Annex: proof for formula conversion

$$\sum_{x=1}^{n}\left(1-\frac{(x-1)}{n}\right)$$ (A1)

$$=\sum_{x=1}^{n}\left(1+\frac{1}{n}-\frac{x}{n}\right)$$ rearrangement

$$=\sum_{x=1}^{n}\left(\frac{n+1}{n}-\frac{x}{n}\right)$$ isolation of x by extraction of all constant terms out of the sigma

$$=n\bullet\left(\frac{n+1}{n}\right)-\sum_{x=1}^{n}\frac{x}{n}$$

$$=(n+1)-\frac{1}{n}\sum_{x=1}^{n}x$$

$$=(n+1)-\frac{1}{n}\bullet\left(\frac{n(n+1)}{2}\right)$$ replacement by Gaussian sum formula

$$=\left(\frac{n(n+1)}{2}\right)$$