DETERMINING THE COSTS OF ERP IMPLEMENTATION

Rob J. Kusters[1,2], Fred J. Heemstra[2,3] and Arjan Jonker[3]
University of Technology, The Netherland
[2]Open University The Netherlands
[3]KWDResultaatmanagement The Netherlands

Keywords: ERP implementation, cost drivers, project size.

Abstract: The key question of the research reported here is 'which factors influence Enterprise Resource Planning (ERP) implementation costs'. A 'theoretical' answer to this question has been designed by studying the sparsely available literature on ERP implementation costs, and adding to this relevant items from the related fields of software cost estimation, COTS implementation cost estimation, and ERP implementation critical success factors. This result has been compared with empirical data that have been obtained from two large corporations. The combined result is a first attempt to define ERP implementation cost drivers.

1 INTRODUCTION

More and more organisations decide to acquire and implement Enterprise Resource Planning (ERP) systems. Those that have implemented such a system are regularly confronted with upgrades and new module implementations. In 2000 60% of the Fortune 1000 companies had already taken an ERP implementation decision of at least one ERP basic module or was in a decision process regarding such an implementation (Stein, 1999). Despite existing success stories regarding the many business benefits of ERP usage, the ERP market is confronted with a number of relevant problems. One of these, ERP implementation costs, will be the focus of this paper.

Both literature and practice seem to agree that ERP implementation costs often outstrip original estimates. Zuckerman (1999) e.g. finds that ERP implementations in companies with a turnover above $500.000 show an average budget overshoot of 17%. The Gartner Group (Hunter, 1999), based on 1300 organisations, found that 32% of ERP projects is more costly than planned. Many other authors report budget overshoots in ERP implementation projects (Williamson, 1997), (Pang, 2001), (de Koning, 2004), (Bothof en Götte, 1998), (Wijkstra, 1999).

Budget overshoots in ERP implementation can therefore be considered to be a serious problem that warrants further attention, specifically since investments in ERP tend to be substantial (Sumner, 2000). This paper describes a research project aimed at answering the following question: 'which factors substantially influence ERP implementation costs'. Knowledge of such factors supports structured estimation of ERP implementation costs and can be used to influence an implementation project.

In the remainder of this paper we will refer to these factors as 'cost drivers'. The word 'substantially' has been added, since it is to be expected that the total number of potential cost drivers is significant. Research by Kretzschmar and Noth (1984) resulted in the identification of over 1200 cost drivers influencing the costs of software development. Since ERP implementation, which often includes additional software development, is at least as complex as software development, there is every reason to assume that a similar number of cost drivers can be identified. In the field of software cost estimation it is recognized (see e.g. Boehm 1983) that such a number of cost drivers is too large to be gainfully employable as a management tool and an approach focusing on the most relevant cost drivers is adopted. We will follow this example.

In section two the results of a literature review regarding ERP implementation cost drivers is presented. To our surprise this subject has so far attracted little attention. Much of what could be found is aptly characterised by Klaus et.al. (2000) as “the ERP cost literature is still largely anecdotal and a-theoretical”. Only a few publications directly address ERP implementation cost drivers. Fortunately, a substantial body of literature focuses

J. Kusters R., J. Heemstra F. and Jonker A. (2007).
DETERMINING THE COSTS OF ERP IMPLEMENTATION.
In *Proceedings of the Ninth International Conference on Enterprise Information Systems - DISI*, pages 102-110
DOI: 10.5220/0002363001020110

at ERP implementation critical success factors (CSF's), an issue that is closely related to cost drivers and can support their identification. We also looked at the software cost estimation (SCE) literature. In this research field, that has matured during the previous decennia (Boehm and Sullivan, 2000), the issue of cost drivers has been studied in a structured way. Software development is not identical to ERP implementation but has many similarities (e.g. the involvement of diverse disciplines, complexity, etc.) it is reasonable to assume that this field can be a source of inspiration.

Based on this literature survey a 'theoretical' answer to the research question is formulated in the format of an overview of relevant cost drivers that may impact ERP implementation effort. Since literature only provided limited leads, it was decided to conduct in parallel an investigation in practice. The research question: "which factors substantially influence ERP implementation costs" was put to a number of experts from two large corporations. This is described in section three. The results from literature and practice are discussed in section four.

2 LITERATURE SURVEY

2.1 Cost Drivers

In this section the result of the literature survey on ERP implementation cost drivers is presented. The issue is discussed explicitly by Stensrud (2001), Francalanci (2001) and Von Arb (1997). A relevant basis for this research has been developed by Stensrud (2001). In his research, he wondered if the existing body of knowledge developed for software cost estimation was applicable to estimation of ERP implementation effort. His conclusion was, since most approaches are based upon the use of the number of lines of source code or some synthetic variable such as function point to assess the size of the project, these approaches are not applicable. An ERP implementation project may contain some software development, but will also contain substantial modelling, installation and reorganization efforts. It seems unlikely that a one-dimensional measure of software size will capture this complexity. Typical measures of size for an ERP implementation project would likely use a combination of measures such as the number of users, of reports to be designed, of systems to be adapted, and of ERP modules. In short, Stensrud suggest using a multi-dimensional measure of size.

Stensrud further concludes, based on a screening of existing SCE tools , that the concepts provided by parametric estimation models such as COCOMO II (1997) provide a good starting point for the development of a model to support estimation of ERP implementation effort. Key in this models is the existence of a size metric that can be used to estimate 'normal' costs, and the existence of cost drivers that adjust for project specific issues. He also concluded that emergent models for estimation of implementation effort of standard software (above all the COCOTS model (COCOTS, 2000)) may provide support. Stendsrud bases his conclusions on a logical analysis of the characteristics of the models, but is supported by empirical work by Francalanci (2001) and Von Arb (1997).

The research by Francalanci (2001) is focused at identification of a usable measure of size. In agreement with Stensrud, she deduces that such a measure should be multi-dimensional. Based on an extensive field study, using data from 43 (European) SAP R/3 implementation projects, she identifies three constituting elements for such a size metric:

- Size of the organization. This reflects its inertia, its ability to resist change. The assumption is, that the larger and more cumbersome an organization, and in consequence, the more inert or slow it is, the more effort an implementation will take. As measures for organizational size she tested number of employees and revenue. Both were found useful.
- Size of the configuration. This is expressed in the number of modules or sub-modules that is to be implemented. The logical assumption is that implementation effort will increase with the number of modules to be implemented.
- Size of the implementation. This is expressed with number of users involved, since these indicate training and reorganization effort.

Francalanci, in her research, focuses solely on size. Other cost drivers are left out. This approach is also taken by Von Arb in his PhD. thesis (1997), which focuses on the relationship between size and cost. His research, based on extensive data from practice, also identifies a multi-dimensional measure made up of number of users and number of (sub-) modules.

This is a result that is fairly similar to that of Francalanci. Literature, as far as it is available, seems to agree to the usefulness, the nature (multi-dimensional), and at least some of the components (number of users, and number of (sub-) modules) of the concept of ERP implementation project size to be used as the basis for cost estimation in 'normal' conditions. It should be mentioned however, that

ERP implementation projects often target a single module. The notion of number of modules as a size metrics therefore leaves something to be desired.

Less agreement can be found regarding the question which other relevant cost drivers should be taken into account to adjust this 'normal' estimation to specific local circumstances. Many authors that focus on ERP implementation overshoots indicate where these costs can be found or single out some specific cost types. Main cost types mentioned by all authors are staffing costs and costs of external advisors. Costs of developing / tailoring additional software is mentioned regularly (Wijkstra, 1999) (Bernroider and Koch, 2000). Hardware- en license costs can add up too (Robinson, 1998), as can costs for training (Davenport, 2000) and user support (Saunders, 1999). However, no single comprehensive reference was found that presents a structured overview of cost drivers that affect the amount of costs required (Mello, 2002).

Fortunately, extensive literature is available in the related field of research that focuses on critical success factors (CSF) for ERP implementation, see e.g. Akkermans and van Helden (2002), Al-Mashari et.al. (2003), Holland and Light (1999), and Umble et.al. (2003). We consider this field to be related, since factors that affect project success will also impact project costs. This CSF literature is therefore a good source of inspiration when determining relevant cost drivers. CSF's are 'things' that are a precondition for a successful implementation. Success can be determined by the degree to which a project is finished on time and within budget and the resulting product fulfils expectations / requirements. Furthermore these 'things' are such that they can be addressed adequately in advance (i.e. before the start of the implementation project). A CSF has to be handled before work starts. If e.g. insufficient management commitment is present, this can seriously impede an implementation project, resulting in budget and/or time overshoots, a product of insufficient quality or even abandonment of the entire project. A bad choice of an external implementation partner may have an identical result.

This implies that if a CSF has been dealt with insufficiently in advance, it will become a cost driver. The difference between CSF and cost driver is gradual. Management commitment is a precondition for success, but the degree of management commitment available (none, low / insufficient, sufficient, whole hearted) is a factor that will influence implementation costs.

In principle all CSF's mentioned in literature can, following this line of reasoning, be transformed into cost drivers. In particular issues dealing with implementation approach, vendor relationship / contract, conversion, involvement external consultants and project organization are obvious candidates.

Summarizing, on the basis of ERP specific literature the following conclusions can be drawn:

- Size is an important cost driver.
- Measuring size requires a multi-dimensional measure.
- The most specific research focuses on size, with little regard for other cost drivers.
- The number of modules as a measure for size is extremely coarse, allowing little or no nuance.
- CSF literature provides an interesting additional source of inspiration.

Stensrud (2001) advised to take a closer look at the field of software cost estimation. This will be done in the next section.

2.2 Software Cost Estimation

A widely adopted approach for software cost estimation was developed by Barry Boehm in his Constructive Cost Model (CoCoMo) (Boehm, 1983). This model has the basic format:

$$\textit{Development costs} = (a * cd[size]^{b}) * cd_1 * cd_2 \ldots\ldots * cd_{14}$$

(where cd stands for cost driver)

The cost driver 'size' (cd[size]) is viewed as the most dominant cost driver, not only in the CoCoMo model but also in many other models (Heemstra, 2005). Here 'size' is often measured by metrics such as number of Lines of code and number of function points (FP). FP is a measure of size based on software functionality (Albrecht en Gaffney, 1983). To facilitate discussion of cost drivers, Heemstra (1989) developed a framework for categorizing cost drivers that has been well received in literature. In this framework cost drivers are classified as:

- **Size.** How 'big' is the software (usually expressed in Lines of code or function points).
- **What with.** Which resources are used when developing software. Usually three types of resource are distinguished:
  - People (e.g. quality of project management and knowledge, experience, and availability of the development team),
  - Organisation (stability, work methods, etc.),
  - Systems (characteristics of the target platform such as CPU-time and memory capacity).
- **How.** How is software being developed. This incorporates cost drivers that deal with the type

of programming approach taken as well as project management approaches.

- **For whom** is the software developed, with cost drivers such as the number of users, the degree of user involvement and IT knowledge and experience of the users.
- **What** is being developed; e.g. the cost drivers required quality, complexity, specification stability and required documentation.

Looking at this collection of cost drivers from the Point of view of ERP implementation cost estimation the following may be noticed:

1. As Stensrud indicated, no suitable size indicator is offered,
2. Most of the costs drivers in the ‘what’ category differ from application to application. Degree of required reliability, application complexity, amount of required documentation are examples of cost drivers that differ from application to application. In the case of ERP no such differences are apparent, since the ‘application’ is a single standard system. The values of this type of cost driver will be more or less fixed per type of module, although differences between modules may be envisaged. Relevant cost drivers in this category will therefore only those that refer to sort or type, e.g. brand of ERP system or type of module (finance, HRM, etc.).
3. ERP implementation will be influenced by such cost drivers as knowledge, experience and involvement of the users in a way similar to software development. This implies that the cost drivers form the ‘for whom’ category are prime candidates for ERP cost drivers.
4. Similarly, the cost drivers from the category ‘with what’ are directly applicable in an ERP setting, specifically those from the sub-categories people and organization.
5. Most of the cost drivers from the ‘how’ category are specific to software development and therefore not applicable in an ERP implementation setting. However, the cost driver ‘project management approach’ can be used when translated in ERP terminology, leading to ideas as ‘implementation approach’.

A second source was found in the literature regarding estimation of standard software implementation, in particular the estimation model COCOTS (Agarwal et.al., 2001).

The name COCOTS stands for Constructive COTS Model. Constructive refers to CoCoMo (Constructive Cost Model) because it follows this models approach and also because it is an open / transparent model. "COTS" stands for Commercial-Off-The-Shelf, indicating the models aims at standard software components. Within COCOTS cost drivers are grouped into three categories:

- cost drivers related to staff experience, quality and availability,
- costs drivers related to COTS components, such as component maturity, complexity, and update frequency,
- cost drivers related to the application, such as application reliability, portability, number and complexity of interfaces, and limitations in technical performance.

For the determination of ERP cost drivers the following lessons can be drawn from COCOTS:

1. For identical reasons used in the made to measure situation, factors regarding knowledge, availability, and experience of staff are relevant.
2. COCOTS defines cost drivers that specifically target the aspect 'integration', the interfacing of the COTS component with other software. These are candidates for the ERP context.
3. Some cost drives are related to the COTS component itself and can be translated to ERP. Specifically 'training', 'maturity', and 'frequency of new releases' can be mentioned.
4. Some cost drives are related to the COTS component itself and can be translated to ERP. Specifically 'training', 'maturity', and 'frequency of new releases' can be mentioned.
5. From the final 'application' category of COCOTS costs drivers, 'number and complexity of interfaces' is a prime candidate cost driver.

Based on this literature survey we selected a number of cost drivers that we feel are factors that 'substantially influence ERP implementation costs'. The factors are classified according to the framework that Heemstra (1989) developed for categorizing software cost estimation cost drivers. For the category 'size' we conformed to the empirically tested proposals of Francalanci. Other choices were based on considerations as presented above. The result is presented in table 1 (the heading 'theory'). This table also contains the results from the empirical studies that will be discussed next.

3 EMPIRICAL RESULTS

Since the literature results were derivative at best, two empirical studies were conducted to provide an additional source of material. Two large Dutch companies participated in this study, an insurance company (Interpolis) and a mail and parcel distributor (TNT). Both companies have recently

executed a number of ERP implementation projects, so it is reasonable to assume that relevant knowledge is available within the organisation. Obviously, no data with regard to cost drivers was maintained during the project. Some relevant information might be obtained by studying the project documentation. Such documentation is difficult to access for an outsider and, given that this outsider is not aware of local specifics, interpretation of the data would at least be difficult. Having insiders study the project documentation was infeasible due to the prohibitive costs this would entail. So it was decided to 'ask' knowledgeable persons what they experienced as cost drivers.

This process of 'asking' was carried out in a three step approach consisting of a meting, a survey and an additional meeting. First in a meeting with a number of experts from the organisation is held to identify possible cost drivers. This meeting consists of a divergent step in which in a brainstorm setting as many as possible relevant cost drivers where generated. This was followed by a second convergent step in which the results were classified. For this a group protocol was required that could deal effective and efficient with a significant number of items, since a large number of potential cost drivers could be expected to be generated. The metaplan protocol (Härtl en Kemmerer, 2002) fulfils this requirement.

Following this meeting a survey was send out to a larger number of knowledgeable persons within the company to validate these results. A final step consisted of a second meeting, with the same participants from the first meeting, to analyse the results of the survey and to draw conclusions from it. This resulted in the following approach that was adopted in both organisations.

Meeting 1: cost driver generation.
Before the meeting participants were selected, based on their participation in recent SAP implementation efforts. Active participation was required since only motivated participants could be expected to contribute to the result. To insure this, participants attended on a voluntary basis. At TNT seven staff members participated in the research, at Interpolis this number was nine. At the start of the meeting the research objectives were introduced further and the approach that would be taken was outlined. The results of the literature survey were not presented, to prevent undue influence on the results. However, the basic concepts of estimation: size and cost drivers were introduced, using the field of software cost estimation as a reference. Again, apart from some obvious examples, no software cost estimation cost drivers were presented to avoid undue influence.

Next the divergent, generating brainstorm was executed. Participants were asked to generate cost drivers in groups of two or three, thus combining the motivational aspects of brainstorming while at the same time avoiding pitfalls of production blocking (Nijstad et.al, 2003) and groupthink (Janis, 1982). The participants were asked the following questions:

- what do you consider to be a good measure of size for an ERP implementation project, and
- what are relevant cost drivers impacting ERP implementation within your organisation,

and to write the answers on large post-it stickers in preparation for the next (convergent) step.

Following this, the convergent metaplan protocol was followed, which is a two step procedure. The first step in essence consists of a fast interactive process in which all participants take part. A facilitator states the objective of the meeting, which is in this case, is identical to the questions asked for the brainstorm. Then a first item (written on a sticker) is placed on a large empty wall. A second item is than taken and the group decides if this item is similar to the first of if it should form a separate group. If it is similar, it is placed next to the first item, otherwise it is placed further away on the wall. This procedure is followed by all remaining items. If discussion takes to long, an item is set aside. As more and more items are sorted, clusters of items start to appear that each identifies a potential cost driver. At the end, cards that have been set aside are handled again. Given the emergence of recognisable clusters of items, placement should be easier. This part of the protocol ends when all items are placed.

The second part of the metaplan protocol consists of looking at the resulting clusters and naming them to facilitate handling and discussion. Next clusters that have become too big and contain several relevant concepts are split up. Also, clusters that are sufficiently similar are joined together. Again the resulting clusters are named. This process might be repeated several times until the participants are satisfied with the result, thus ending the metaplan protocol. After this, the resulting clusters were placed in the Heemstra (1989) framework (size, what, how, for whom, what with) in order to facilitate further usage.

The survey
In order to validate these results, a survey was developed and send out to a number of knowledgeable stakeholders in the ERP implementation process. The survey contained all

cost drivers that had been identified in the first meeting, together with a definition that had been developed by the researchers. For each size measure / cost driver the respondents were asked to determine (on a yes/no scale) if is an important size measure / cost driver.

Furthermore, the respondents were invited to contribute additional size measures and/or cost drivers. For Interpolis the survey was distributed among 32 persons, of which 18 responded. For TNT these figures are 33 participants with a less satisfactory number of 10 respondents.

Meeting 2: discussion of results

The participants of the first meeting were invited to a second meeting in order to discuss the results of the survey. These results were presented in a PowerPoint presentation. Each cost driver, whether originally proposed in the first meeting or added by a respondent of the survey, was discussed separately in order to decide whether or not it should be accepted as a relevant size measure / cost driver. The resulting list was finalised and accepted.

4 DISCUSSION

Table 1 contains the over all results of this is research. The column 'theory' contains the results from literature. Columns 'TNT' and 'Interpolis' contain the results of the empirical investigation.

The most noticeable difference between theory and empirical results can be found under the heading 'SIZE'. As was discussed, 'size' is expressed in literature as a multidimensional measure containing according to Francalanci number of modules / sub modules, size of the organisation (expressed as number of employees or revenue) and number of users. The empirical data gave different results. The notion that size is multidimensional was supported in both organisations, but the composing metrics were different. It appeared that size was perceived as a combination of:

a) A measure related to the amount of work that is involved in configuring the ERP system. This measure included items such as the *number and complexity of transactions, interfaces, and reports*, and at *the amount of data and data conversion*.
b) A measure that indicates system implementation and business reorganisation costs. Francalanci refers to this as implementation size. If this size increases, more staff needs to be trained and also more people are involved in organisational change efforts. This measure includes items such as *number of users, number f user groups, number of departments and number and complexity of business processes.*

It follows that people in practice perceive 'size' at a more detailed level of abstraction (e.g. number of interfaces) than Francalanci and Von Arb who look at the rather coarse measure of number of modules.

In the category 'WHAT' differences between literature and practice were less significant. A number of cost drivers from literature were classified in another category (e.g. number of interfaces and data conversion to 'size'), declared to be not applicable (e.g. the type of ERP system involved, which is of course fixed within one company) of relabelled (e.g. maturity technology versus frequency of releases). It is remarkable the type of module is in practice not recognised as a relevant cost driver although in both organisations the SAP CRM module was mentioned as a module that was more difficult to implement.

For the category 'FOR WHOM' differences are also small and explainable. The item of branch is not applicable, since this is not something that can be noticed within a single company. This is also true for the cost driver national/international since in both companies ERP implementation efforts took place within a national setting. Agreement existed as to the relevance of the cost driver fit. A similar agreement exists for the cost driver process maturity. However, when interpreting the meaning of this concept the aspects may be considered:

- the degree to which an organisation has shifted from a functional to a process orientation,
- the degree to which business processes have been documented (standardised, described, and modelled), and
- the degree to which process execution and process description are consistent. The notion 'consistent' refers to consistency in execution (is the process – over time – executed consistently) as well as to the required consistency between reality and documentation (is the process executed in line with its description).

Both organisations mention a number of organisational characteristics as cost driver. TNT mentions in this context *number of organisational units involved* whereas Interpolis mentions *stability, willingness to change,* and *ability to change*. This type of cost driver was not found in literature.

In the category 'HOW' from literature a number of cost drivers were identified that were not mentioned by the organisations. Both organisations used Accelerated SAP as implementation method,

meaning that the cost driver *method* cannot be recognised. Similar arguments might explain why *implementation approach* and *implementation strategy* are not mentioned. The theoretical cost driver *contract* is also not recognised, maybe because it was handled correctly throughout?

In the final category 'WHAT WITH' the differences between theory and practice are limited, mainly a matter of description detail. An important 'theoretical' cost driver *training* is not mentioned in practice. A possible explanation is that this driver is included in the size measure *number of users*.

The main conclusions from the research are:

- The state of the art of identifying ERP implementation cost drivers is not yet very far advanced. Both in scientific literature and in practice this issue receives limited attention.
- The limited knowledge available on ERP implementation cost drivers is mainly focussed on determining a suitable measure for ERP

Table 1: Overview of cost drivers from theory and from the empirical studies.

		THEORY	TNT	INTERPOLIS
COST DRIVERS	**SIZE**	- # of (sub)modules - # of users - size organization	- # of transactions, - # of interfaces, - # of reports - amount of data conversion - # of users - # of user groups	- # + complexity interfaces - # + complexity transactions - # + complexity reports - # + complexity business processes - size + complexity data - # of departments - # of users
	WHAT	- type of system (SAP, …) - type of module (CRM, …) - degree of tailoring - number of interfaces - frequency releases - data quality/conversion	- # of modules - maturity of the technology - degree of tailoring	- maturity of the technology - degree of tailoring - Module connectedness
	FOR	- # of sites - National / international - Process maturity - Fit between organization and product - Branch	- # of stakeholders - # of organizational units - Process maturity - Fit	- Stability organization - Willingness to change - Ability to change - process complexity - Insight in the processes - Fit
	HOW	- Method (e.g. ASAP) - Team composition - Implementation strategy (per module / per site / big bang) - Implementation approach (degree of BPR) - Contract (clarity responsibilities and authorities)	- Vision - Management	- Vision management - Commitment management - Steering management
	WHAT WITH	- Staff availability (degree of, continuity), - Staff quality (technical, social, ..), - Tool availability (degree of, continuity), - Tool quality, - Quality, continuity and availability of training	- Team continuity - Team composition - Team quality - Quality Business users - Availability business users - Consultant quality - Availability management	- team composition - team quality - team maturity - consultant quality - consultant knowledge - User quality - Critical attitude users - Quality developers / /business analysts - tools quality - Test approach - Infrastructure - Experience project manager

implementation project size.

- The ERP implementation CSF literature can provide valuable insights.
- The contribution of the field of software cost estimation is limited. Cost drivers for software development differ markedly from those influencing ERP implementation. Only for 'general' drivers such as staff knowledge, experience, and availability, and involvement of users, staff and management, clear parallels emerge.
- The COCOTS model, aimed specifically at characteristics of standard software such as degree of integration, frequency of releases and system maturity, provided solid inputs.
- Results from practice show a much more detailed approach to measuring size than was found in literature, although the notion of multidimensionality for such a size measure was supported both in theory and in practice.
- A number of potential cost drivers cannot be expected to vary within a single company. identification in a practical setting is therefore unlikely. Examples of such potential cost drivers are implementation approach and type of system. These cannot be confirmed or repudiated on the basis of this research.
- Of the remaining 'theoretical' cost drivers only a limited number (contract, training and type of module) were not confirmed in practice. Only a limited number of cost drivers (mainly organisational characteristics) were mentioned in practice that had not been mentioned in the theoretical list. On the whole to a large degree theory and practice identify identical cost drivers, although small differences in formulation and level of detail may be noticed.
- Summarising it may be stated that table 1 gives a reasonable first approach towards an answer of the research question: 'which factors substantially impact ERP implementation costs'.

The organisations involved recognised that the project provided a solid basis for further learning. As a direct benefit was mentioned that the information obtained was already considered to be useful for:

- better planning and monitoring of projects,
- better control of vendors.

Further research is firstly aimed at determining a proper size metric. Next steps include determining the relative impact each of these cost drivers may have, development of an estimation mechanism, and identifying ways of handling these data.

REFERENCES

Akkermans, H.A., and Helden, K. van, Viruous and vicious cycles in ERP implementation, *European journal of Information Systems* (2002) 11. 35-46.

Albrecht, A.J., and Gaffney, J.E. Software Function, Source Lines of Code, and Development Effort Prediction, *IEEE Transactions on Software Engineering*, vol. SE-9, no. 6, 1983.

Al-Mashari, M., Al-Mudimigh, A., and Zairi, M., ERP: a taxonomy of critical factors, *European J. of Operational Research*, vol 146, 352-364, 2003.

Arb von, R**.** *Vorgehensweisen und Erfahrungen bei der Einführung von Enterprise-Management-Systemen dargestellt am Beispiel von SAP R/3*, Ph.D.-Thesis Universität Bern, 1997.

Agarwal, R., Manish Kumar, Yogesh, T., Mallick, S., Bharadwaj, RM., Anantwar, D. Estimating Software Projects, *ACM SIGSOFT, Software Engineering Notes*, vol. 26 no 4, July 2001, pp. 60-67.

Bernroider, E., Koch, S. Ergebnisse einer empirischen Untersuchung der Entschei-dungsfindung bei der Auswahl von betriebs-wirtschaftlicher Standard software. *Wirtschafts-informatik* vol. 42, nr. 4, 2000.

Boehm, B.W., Sullivan, K.J., Software Economics: a Roadmap, *Proc. of ICSE 2000*, 319 – 34, 2000.

Boehm, B.W. *Software Engineering Economics*, Prentice Hall 1983.

Bothof, N.W.J., Götte, B.J. *Enterprise Resource Planning als omwenteling*, Giarte Research, 1998.

COCOMO II, *Model Definition Manual, Version 1.4*, http://sunset.usc/edu/COCOMOII/cocomo.html, 1997.

COCOTS, *Model Description*, http://sunset.usc.edu/ research/COCOTS/index.html 2000.

Davenport, TH. In search of ERP paybacks. *Computerworld*, 34 (8), 42. 22 August 2000.

Francalanci, C. Predicting the Implementation Effort on ERP Projects, *J. of Inf. Technology*, 2001, 33-48.

Härtl, J., Kemmerer, J. *Präsentation und Moderation*, Cornelsen Verlag, 2002.

Heemstra, F.J., *Hoe duur is programmatuur?* Kluwer bedrijfswetenschappen, 1989.

Heemstra, F.J. Software; what does it cost? *International J. of applied Economics and Econometrics*, 2005.

Holland, C.P. and Light, B., A critical success factors model for ERP Implementation, *IEEE Software*, may/june 1999, pp. 30-36.

Hong, K.K. and Kim, Y.G., The critical success factors for ERP implementation, Information & Management, vol. 40, 25-40, 2002.

Hunter, R. *Is ERP Delivery so bad?* Gartner. 1999.

Janis, I.J., *Groupthink, psychological studies of policy decisions and fiascos*, HoughtonMifflin, Boston, 1982.

Klaus, H., Rosemann, M., Gable, G. What is ERP? *Information Systems Frontier*, 2, Aug. 2000, 141-162.

Koning, F. De, ERP implementaties; management-probleem of softwareprobleem? *MAB*, 2004, 435-444.

Mello, A., ERP fundamentals - ERP's hidden costs, *Inside ERP ZDNet*, Februari 7, 2002.

Nijstad B.A., Stroebe W., Lodewijkx H.F.M., Production blocking and idea generation, *J. of Experimental Social Psychology*, Vol. 39, 2003, pp. 531-548.

Noth, T., Kretzschmar, M. *Schätzung von Software Einführungstrajecten*, Springer Verlag, Berlijn 1984.

Pang, L. Manager's guide to ERP Systems, *Information Systems Control J*. Vol. 4, 47-52.

Robinson, P. *Business Excellence, the integrated Solution to Planning and Control*, BPI, 1998.

Saunders J., Beware of Costs Lurking in ERP - Industry Trend or Event. *Computing Canada*, March, 1999.

Stein, T. Big Strides for ERP. *Information Week*, Jan. 4, 1999, 67-68.

Stensrud, E, Alternative Approaches to Effort Prediction of ERP Projects, *Inf. Softw. Techn.* 43, 413-423, 2001.

Sumner, M. Risk factors in enterprise-wide ERP projects. *J. of Information Technology*, nr. 15, 317-327, 2000.

Sumner, M. *Enterprise Resource Planning*, Prentice Hall, 2004.

Umble, E., Haft, R., and Umble, M., ERP: implementation procedures and critical success factors, *European J. of Operational Research*, vol. 146, pp. 241-257, 2003.

Williamson, M. From Sap to Nuts. *Computerworld*, 31, November 10, 1997, 68-69.

Wijkstra, J. Integraal karakter het grootste probleem, *Informatie Management*, april 1999, pp. 29-32.

Zuckerman, A. ERP, Pathway to the Future or Yesterday's Buzz? Transportation & Distribution, 40, 1999, 37-44.