

# ERP Services Effort Estimation Strategies Based on Early Requirements

Pierre Erasmus<sup>1</sup>, Maya Daneva<sup>2</sup>

<sup>1</sup>SAP B.V. The Netherlands  
pierre.erasmus@sap.com

<sup>2</sup>University of Twente  
m.daneva@utwente.nl

**Abstract.** ERP clients and vendors necessarily estimate their project interventions at a very early stage, before the full requirements to an ERP solution are known and often before a contract is finalized between a vendor/consulting company and a client. ERP project estimation at the stage of early requirements in a contractual setting has been challenging as it requires the inclusion and the availability of multiple experts, each being a specialist in a particular domain, e.g. solution architect, technical architect, consultant project manager. Today's highly volatile and customized ERP landscape demands a new approach to estimate effort by leveraging ERP professionals' tacit knowledge and expert judgments, as well as information from historical project databases (if available). This paper presents three ERP Service estimation strategies that leverage the strengths of both Functional Size Measurement (FSM) and expert-judgment-based estimation techniques. These strategies use a more structured approach to reduce the effects of expert bias and avoid common pitfalls associated with FSM and expert judgment-based estimation.

**Keywords:** Enterprise resource planning, requirements-based effort estimation, early requirements, contract-based project delivery, empirical study

## 1 Introduction

Estimation of the effort of professional experts in implementing enterprise resource planning (ERP) projects in client organizations often takes place in face of early, incomplete, and even unknown requirements [1]. Such estimates are critical for various decisions concerning the contractual agreements between ERP clients and ERP consulting partners (or vendors). For the last five years, the ERP domain has experienced rapid change due to a new technological shift and innovation, e.g. realized via in-memory optimized databases [2] which reduced the need to separate ERP services from the ERP suite. This major shift in the ERP industry brought three new challenges from project estimation perspective in contract-based ERP delivery projects:

(1) Traditionally, ERP effort estimation methods were often based on sizable and pre-defined ERP modules [26]. Due to the fixed nature of the client organization's functional groups that adopted those modules and their standard business processes, project managers and project estimators often relied on making reuse of functional

size measurements (e.g. [27]). With a new shift to an ERP service platform [2], there is a need of estimation approaches that are able to account for more innovative, customizable and flexible solutions, which not always relies on functional reuse.

(2) Projects within the domain of ERP Services are characterized with a higher degree of uncertainty, which often includes more innovation and adaptability to newer technologies in a faster rate. In turn, the projects embracing rapid change render any estimation figures obsolete, relatively quickly. Investing effort in achieving a precise functional size number (e.g. a function point count [1]) can be therefore impractical as benefits are short-lived.

(3) ERP Services projects are more explorative in nature, adapting to the natural flow of business which constitute continuous change and improvement, where the traditional ERP projects were more descriptive, governing business by suggesting standard content. This increases the interest on project managers' side in searching for more cost-effective project estimation techniques, beyond the realm of function size measurement (FSM) based estimation methods.

In light of these challenges many organizations realized that the reliance of fixed estimation rules and functional size estimations (such as the function-point-based methods in [1] or in [27]) shown to be useful for certain requirements which often occurred in the previous SAP Netweaver landscape [3]. These methods might no longer be a good fit for the ERP services domain, due to enormous scope of creating customized content for all possible scenarios covered, e.g. by the flexible SAP HANA platform [2]. To cope with this new project context, the project managers working on ERP Service projects, more often than not adopt an approach that relies on experts to evaluate project effort based on their experience and judgment [23] as well as on the available early requirements at the pre-contract stage. This observation motivated us to initiate research on designing three commonly used estimation strategies that could be used with expert-judgments and FSM methods depending on the need and fit within the ERP project context. The question used to guide this research was: *How can practitioners increase confidence and the accuracy of effort estimations by injecting more realism alongside the use of expert based judgments in combination with FSM methods in the ERP domain?* Following O'Brian's action research method guidelines [25], we engaged practitioners from SAP Netherlands, SAP AG, Germany, and SAP US, as part of our research process to formulate strategies for ERP Service project estimation.

In the next sections, we first presents related work on expert judgment in effort estimation (Sect. 2). and our research method (Sect. 3). We then summarize the approach to estimation based on early requirements, used at SAP (Sect. 4). We also describe the two ERP Service contexts for which we want to devise estimation strategies so that expert bias is reduced. We present the strategies (Sect. 5) and provide examples of how to validate the estimates in real-world projects (Sect. 6). Last, we draw conclusions in Sect. 7.

## 2 Related Work

Using expert judgments is the preferred method for estimating effort in ERP projects, despite overwhelming amount of evidence suggesting that expert judgment is flawed [4]. In our research, we set out to learn from the challenges and the failures in expert-judgment-based estimation and use this learning to introduce improvements in the form of a more structured approach. As there is no one best estimation method for all project requirements in all landscapes applicable for all conditions we rather focus this article in providing and grouping the main problems associated to effort estimation scenarios.

Some of the main pain points reported in literature [4,5,6,7,8,9,10,11,12,13,13,14,15,16,17] and in the first author's experiences are listed below:

First, expert judgments is an approach that often takes for granted the availability of experts. Experts (just as is in the ERP domain) are often in short supply and their tacit knowledge is hard to transfer without having experts available.

Second, experts often focus on a very specific set of topics and rarely consider obtaining knowledge of the bigger picture. Therefore we can differentiate between different types of expert judgments required to derive the complete scope. Experts (who are specialized on a specific topic) often show to be overconfident on the broader scope and typically provide confidence intervals that are too narrow [5].

Third, just like any occupation, experts make many reasoning mistakes [6]. These mistakes are often unintentionally repeated until corrected by feedback from others. Of course, one might consider some of these mistakes to be intentional due to the expert's involvement of the suggested project and inherent conflicts of interests [6]. The quality of professionals' judgment and decision-making is rarely improved through experience, mainly because professionals do not receive accurate and timely feedback [7].

Fourth, some inconsistencies arise when project estimates are done differently by different experts. The study in [8] showed that stakeholders in the estimation process had differing objectives than those objectives affected by their estimates [9].

Fifth, estimators find it difficult to use previously completed projects as analogies [10]. There is little knowledge about how people select and use historical information in the absence of formal estimation tools [11].

In this research, our position is that the problems associated with using expert judgment methods should not discourage professionals from considering it, but rather should trigger a conversation on how to leverage its strengths and reduce its weaknesses. Tactics for countering the weaknesses can be used to decrease the negative effects of expert judgments during the effort estimation process. The literature on expert judgment [12,13,14,15,16,17,18,19,20,21,22] does suggest a few tactics that could be good candidates to be considered for adoption or adaptation within the ERP domain. Examples include the following:

(1) Involving expert in further enquiry (use of other techniques/methods) for an 2nd or 3rd opinion where uncertainty is high.

(2) Risk of misleading estimation results provided by an expert can be mitigated or reduced by practices encouraging the self-awareness of this expert's own biases and

communicating the probability where uncertainty is high. This uncertainty could then be dealt with as further enquiry throughout the estimation process.

(3) Novices can be used if there is a shortage of experts as shown in [12]. Therein, the authors found that experts show less overconfidence than novices. With a little training, novices could improve their self-insight "calibration" [12].

(4) To counter expert's reliance on their past experiences and, in turn, their too narrow scope of estimation, there is a suggestion [13] to create a collection of alternative occurrences to widen the scope of possibilities within an estimate, making use of cause-and-affect scenarios while keeping an open mind for "extrimistan" (explained as black swans by Nassim Taleb) or things that could go wrong. Generating thoughts about how things can turn out can also influence metacognitive feelings, which could be useful in a controlled and consistent basis [14]. Another suggestion is to widen up the estimation focus by creating optimistic and pessimistic estimates. Best guess estimates were more comparable with the optimistic scenarios than with the pessimistic ones [15].

(5) Historical data can be used to generalize formulas or rules-of-thumb, e.g. the knowledge that a small software program module with high complexity requires about 30 work-hours [16].

While the above examples include counter-measures concerning a single expert, other solution tactics focus on grouping experts together for the purpose of effort estimation. Numerous studies emphasize the advantages of such group work, e.g.:

(1) Cross assessment of professionals (e.g. experts) was found to reduce the chance of overconfidence [17]. This study also indicated that when software professionals assessed the uncertainty of other fellow professionals' estimates, they were better calibrated and showed no systematic tendency toward overconfidence.

(2) Judgment-based (group discussion-based) combination strategies were superior to mechanical combination strategies in a software development context [18]. Estimates produced after the discussions were higher and more realistic than the initial individual estimates [19].

(3) When experiencing some uncertainty, experts could use simple cognitive strategies as, e.g. '*take an outsider's view*', and '*consider the opposite*' which have turned out efficient yet inexpensive in many circumstances. To improve the realism of uncertainty estimates is to first ask for the historical distribution of estimation error and then – for the minimum and maximum percentage deviation reflecting the confidence level [20].

Clearly, the plethora of counter-measures suggests that there is no one best way to conduct an accurate estimate. Therefore in our work, we take as a premise that making use of a combination of estimation methods [21,22] would help reduce experts' bias and increase estimation accuracy.

### **3 Research Method**

As indicated earlier, we used the Action Research method of O'Brien's [25] to structure and organize our research process. A purposive sampling technique is used as a type of non-probability sampling that is most effective when one needs to study a

certain domain with knowledgeable experts within. It consisted in a total of 11 semi-structured interviews and 3 project observations. The participants were instrumental to formulating the effort estimation strategies. The participants in the study were SAP professionals who worked on ERP Services projects. The participants were chosen by the first author based on his knowledge of their typicality. Action research cycles [25] were used to investigate the activities, interaction between stakeholders and existing practices used within the effort estimation process, and to use this knowledge in the design of a new method that leverages expert-based judgments centric effort estimation purposes. It consisted a total of 11 semi-structured interviews and 3 project observations and took place by conducting five cycles [25]. The first cycle was to determine the problems related to using experts' knowledge for effort estimation purposes, according to the literature.

The second cycle in this research focus on semi-structured interviews whereby different experts in the ERP industry are interviewed in two stages. First, our initial interviews are used to get a better understanding regarding the current challenges in practice. These interviews have less structure and are more open for interpreting and exploring current practices and argumentation from different types of experts. In a second stage, follow-up interviews were conducted with more structure and with closed and open questions, in order to understand how successful these experts consider the practices mentioned in the first interviews.

The third cycle focused on observations of the first author in three real life projects, where he was involved in expert-judgment-based estimation process. The fourth cycle focused on the Estimation Strategies design, described in Sect.5 The fifth cycle focused on the empirical evaluation of the Estimation Strategies validation, which forms part of a larger scope not covered in this article.

## **4 ERP Effort Estimation**

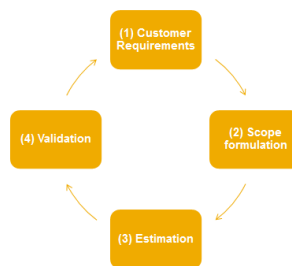
### **4.1 Current estimation practices at SAP**

The early stage of an SAP project is usually characterized by strong involvement of experts available from all domains such as development, solution management, architecture, sales, presales and consultancy. They are specialists on different topics and work together on creating solution roadmaps, scoping documents, applying rules of thumb and creating an architectural design for integration possibilities. The artefacts that serve as inputs to the effort estimation process are tested during different quality gates. Before a new solution is introduced to market, a special project with its own supported processes called Ramp-Up is offered. A ramp-up project is the last quality gate for SAP projects before the release of an ERP service solution to the general market. These reconfigurable rules-of-thumb and scoping documents are made available for reuse for non-experienced individuals (novices) and project managers upon the launching of specific ERP Services to the wider market. However, there are a number of problems with these current estimation practices: identifying an incomplete project scope, a lack of business and/or technical insights, and missing critical components of the targeted ERP solution, ultimately lead to inaccurate

estimates. Project managers who did estimate their projects in isolation often relied on outdated or incomplete documentation, such as past SAP Work Breakdown Structures (WBSs) and were unable to identify the projects in which it's more meaningful to rely on functional size estimation (e.g. Function Points counts), expert estimates or historical estimates. Making reuse of past estimates was useful in projects in which little or no experience was available but there was no way to tell if those estimates were accurate or not as they were never validated. In cases where past estimates were valid, they might have been specifically tailored for a specific situation at a specific customer. It was at this point of time, when it became clear to many project managers at SAP that within the radical shift in the ERP Service domain (from the traditional pre-fixed ERP modules such as those in the SAP R/3 packages) some reliance on expert judgment principles needed to be adopted as part of the SAP effort estimation approach and needs to be combined with other approaches when applicable. This was exactly the gap in the SAP ERP industry, how to combine different estimation approaches and when to use the different approaches for a certain situation.

### 4.3 ERP Services Estimation Method

In a previous study we designed a method known as the ERP Services Estimation Method [23] forming part of research conducted in a scope larger than the focus of this article. This method is designed specifically for facilitating effort estimations of ERP Services. The suggested method includes tactics and lessons learned (addressed in Sect. 3) and designed to decrease participation or memory bias while improving estimation accurateness. The estimation method (Fig. 1) includes four steps: (1) customer requirements, (2) scope formulation, (2) estimation and (4) validation. Depending on the project's complexity or risk, the cycles could be repeated iteratively to increase the level of detail in the expert knowledge needed to derive project estimates.



**Fig. 1.** ERP service effort estimation method [23].

**1. Customer requirements elicitation:** This involves the elicitation and documentation of customer requirements. It resembles the evaluation phase of the ASAP methodology for rapid ERP implementation [26] often used in the industry for SAP projects. More in detail, as ERP Service projects are often associated as part of a

change management initiative, therefore engineering the customer requirements also includes the investigation (analysis) and documentation of business processes or refinement of these business processes (re-modeling) to present the suggested process changes. It is worthwhile noting that reuse of requirements is a common practice at SAP, to determine dependency and completeness. At early project stages, the point of 'just enough' requirements is reached when both the customer and their business have reached a desired degree of understanding and consensus. Typical 'just enough' requirements elicitation techniques practiced at SAP, include design thinking workshops [29], storyboarding and high level mockup prototyping with the customer's business stakeholders involved. The outcomes of these elicitation techniques are a combination between prioritizing requirements, their relationship with each other (traceability) and sometimes a high-level design.

**2. Scope formulation:** This is concerned with scoping of ERP Service projects. It resembles the business blueprint phase [26] used in the industry for SAP projects. More in detail, this step takes as input the requirements from the previous step, and then results in prioritized requirements. The architect can often determine the best fit (high level design) and best choice of technologies based on previous experience. The architecture (now representing groups of functional requirements) is advanced for further refinement (as per enquiry) to specific solution experts representing specific group of functionalities. Important to good expert judgments estimation in this step is to incorporate different stakeholders during the scoping process and to decompose estimates where uncertainty is high.

**3. Estimation:** The effort estimation approach is initially top down followed by a bottom up approach. The estimation is initially rolled down with a top down approach which starts with the architecture and design of the project are defined (or refined in the case of a consecutive iteration). The level of abstraction is determined on the level of uncertainty or risk associated to the project. A project with a high degree of uncertainty or risk is often required to carry out estimation on a more thorough level of abstraction. The bottom-up approach is often carried out in this case when a detailed estimation is required of the effort (time) to implement specific solution objects or functions. Linking common casualty as a common practice by creating rules of thumb is often created for reusable estimation purposes but does not fit every estimation situation. This is considered as good practice as long as there is feedback provided to validate these rules. Inaccurate configurable rules may persist because experts get little or no feedback. FSM methods should also not be used for all situations and should or could only be used when there is an advantage to do so. Therefore the ERP Effort Estimation method could be supported by the use of estimation strategies to indicate when to use these methods in which situation.

**4. Validation:** The validation phase represents activities associated with the validation of the estimate. Among the techniques include creating a process whereby the effort estimates are validated. The validation process itself can make use of expert judgment, actual recorded values, function points and group based validation.

### 4.3 The Two Contexts of ERP Service Estimation

The estimation happens in two types of SAP Services projects (Fig. 2 and Fig. 3), respectively:

(1) Innovation and ramp-up projects (Fig. 2) are also known as time and material projects. Therein, an SAP Service project is estimated with the biggest configuration effort customized for the customer specific needs.



Fig. 2. SAP Innovation and Ramp-Up Projects.

(2) Fixed priced or engineered solutions (Fig. 3) also known as Rapid Deployment Solutions (RDS), which are pre-packaged and engineered for quick implementation. A RDS provides a customer with the baseline solution and demo configuration, plus predefined process. There is also a small part of the project where the customer can select to enable optional scenarios, engineered for quick customization of the most commonly known customization scenarios.

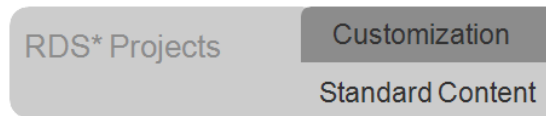


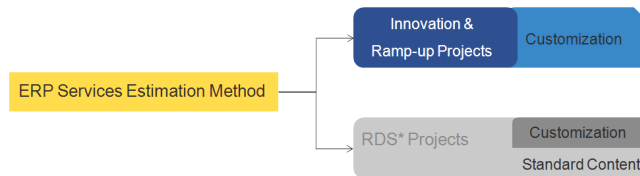
Fig. 3. SAP Rapid Deployment Solution (RDS) Projects.

Type 1 (Fig.1) projects are where a solution is often modified and customized for achieve a high degree of integration to accommodate a customer's current processes whereas RDS projects (Fig. 2) are pre-engineered and implemented as-is with a small customizing piece. While type 1 projects are often focused on interoperability, introducing new technology into a customer's eco system, type 2 (RDS) is more focused on quick implementation (shorter implementation times, adding immediate value) for already matured and proven solutions within a matured eco system, to enable quick results.

In the first phase of creating the ERP services estimation method, our past research [23] mainly focused on the context that is described in point (1), namely innovation and ramp-up projects. Therein, the biggest challenge is still how to effectively and accurately provide SAP services effort estimates for innovation and ramp-up project whereby the emphasis is on customization. Once project type 1 has reasonable solution we can apply the lessons learned from using our method in these projects and adopt it to project types 2. The main difference between the two project types is that project type 1 includes the core baseline and a bigger part of the effort related to customization, whereas project type 2 consists mostly of implementing the baseline and optional customization scenarios (pre-configured customization carried out in a lab like environment).



As indicated earlier, even though the ERP Service estimation method is mainly created for typical innovation and blueprint projects it could also be used for RDS (rapid deployment solutions).



**Fig. 4.** ERP Services Estimation Method used for SAP Projects.

#### 4.4 Work Breakdown Structure Templates

As part of our observations of the current requirements-based effort estimation practices at the bidding stage at SAP, it is worthwhile noting that a work breakdown structure (WBS) template is available for each SAP solution and ERP Service offering. The WBS forms part of SAP’s Accelerated SAP (ASAP) methodology [26]. Typically, the WBS represents the activities and sub-activities related to core implementation and baseline of a solution. The WBS is commonly used for project management and effort estimation. A WBS is often complemented with a scoping questionnaire that is used to determine the baseline scenarios outlined within a WBS. In Section 5, we will see how the practice of using the WBS forms part of an effort estimation strategy that uses the ERP Services estimation method [23].

### 5 Solution: ERP Services Estimation Strategies

The ERP Services estimation method [23] consist of an open and extendible framework with different sets of guidance’s and instructions. The open framework makes the method more useful for the general ERP industry, independent on the current experience, organizational type or specialization within a specific market. The overall intent behind the method is to enable different organizations, e.g. consulting groups, development houses and internal self-grown R&D departments to make use of a structured effort estimation approach with explicit inclusion of both quantitative and qualitative information derivable from the early requirements for a project. The open framework allows the project estimators to configure the method to their current situation, depending on their contextual settings such as the presence of a historical database with information on past projects, the available resources (expert’s vs. novices), the maturity of the project teams, and their experience in a certain ERP area. E.g., in a case where experts are in short supply, it’s only logical to select an estimation strategy that uses a high degree of FSM or preconfigured rules of thumb. On the flipside, if experts are available, then we could rather leverage on their knowledge and expertise where uncertainty or risk is to a higher degree.

The following matrix (Fig. 5) provides an estimator with an overview of ‘best practices’ regarding the degree to which he/she should rely on different estimation techniques. We note that there is not one estimation strategy suitable for every situation, therefore it’s important to identify the situation, resources available, knowledge base of the resources, the characteristics of the implementation eco system and historical project information available. The idea is not to select a single strategy for the complete estimation, but incorporate different strategies within a single estimate, depending on the topic, situation, eco-system and resources. E.g., relying on the **Baseline** estimation strategy as shown in the matrix in Fig. 5, is wise when covering the topics related to implementation of the foundation or baseline of the solution. But using this estimation strategy for tasks related to customer specific customization (depending on the resources available) is not the most optimal and the estimator could rather use the **Tailored** strategy.

The three main ERP services estimation strategies include:

**Baseline:** This estimation strategy includes those requirements and activities related to the baseline which represent the implementation of the core features, widely implemented within most projects. As this is the core functionality, project management should have FSM data available to repeat baseline estimations.

**Configurable:** As the name of this strategy indicates, the estimators could rely on a set of configurable estimates or rules of thumb. E.g. rules reflecting the amount of customization commonly appearing for a specific industry (e.g. the telecommunication sector [14]). These activities are often repeatable customization scenarios but still require customer customization and fine tuning to a certain degree. These groups of pre-configured scenarios could also be integrations scenarios of one well known solution to another but still require expert skill during implementation depending on the choice of architecture.

**Tailored:** This strategy should be used for those activities related to customer specific customization and rely on the knowledge and experience of experts. As these cases are very specific to a customer’s demands it’s often tailored specifically to fit a customer’s processes.

We make the note that the strategies in Fig. 5 can be used in two situations: first, when no estimation exists and a team wants to initiate the estimation task and get together experts do come up with the very first estimation on the bases on high-level and only partly understood requirements. Second, the ERP estimation strategies can also be used as an approach for refining/confirming (or validating) already existing estimates of solution implementation and solution maintenance at the time before submitting a bid. In Section 6, we focus our discussion explicitly on this second setting.

1	2	3	4	5	6	7	8
Estimation Strategy	Project type	Implementation	Configuration Component	Estimation Component	Knowledge base	Effort Estimation	Estimation Reference

Baseline	Time & Material (Innovation & Ramp-Up projects)	Rapid Deployment Solutions (Engineered projects)	Baseline	Core	Activities and Sub activities	Explicit	Function Point & FSM Estimation	Work Breakdown Structure Template
Configurable			Customization	Optional	Pre- Configured Scenarios		Expert Judgment	Rules of thumb
Tailored				Specific	Expert Judgment	Tacit	Experience & Expertise	

**Fig. 5.** The ERP Services estimation strategy matrix.

The ERP Services estimation strategy matrix above should be used to assist an estimator to decide where and what estimation strategy with the right mix of methods and material to use, depending on the type of components and projects. The leftmost column of the matrix shows the three estimation strategies. Each strategy is represented by a set of characteristics and a recommended method/s to be used. Identifying the characteristics of each major component of the project and comparing it to the matrix above will assist the estimator with which approach to follow. The second column consists of the project types: the first type is Rapid Deployment Solution, which is considered as baseline because these components form the basic platform for all or most projects and include little or no variable aspects. This type of projects follows a predetermined implementation script which is in line with a straightforward baseline “vanilla” implementation (see the 3<sup>rd</sup> column in Fig. 5) consisting of core configuration components (4<sup>th</sup> column) commonly found in most projects. In this case most estimators are accustomed to use Function Points or another functional size estimation method (7<sup>th</sup> column) with explicit predefined estimation reference materials such as WBS’s (8<sup>th</sup> column).

The Innovation or Ramp-up project type (see the left side of the 2<sup>nd</sup> column) means projects that differentiate themselves from one implementation to another. The Innovation or Ramp-up project often demands a higher degree of customization. Moreover, customization implementation (3<sup>rd</sup> column) consists of configuration components (4<sup>th</sup> column) which is in line with configuration options for different industries, for example and optionally selecting implementation components which correlate with preconfigured scenarios (5<sup>th</sup> column). Estimation of optional configuration components (5<sup>th</sup> column) using configured scenarios might have been

executed in previous projects but often not documented officially as functional count or WBS. Making use of previous estimates might be a valuable source for estimators, deriving and relying on rules of thumb (column 8), derived from experts during previous projects. Specific installation components (4<sup>th</sup> column) is a one-time implementation, where component of the overall solution are very specific to a customer and, therefore, expert judgments (5<sup>th</sup> column) often provide the only feasible way to derive estimates. Expert judgments imply reliance on experts experience and expertise (8<sup>th</sup> column). This information is usually tacit (6<sup>th</sup> column) and not documented anywhere with no possibility to reuse rules of thumb.

## 6 ERP Estimation Strategies for Refining or Validating Estimates

The key idea behind validating an early estimate is that a team reiterates the steps of our approach [23] repeatedly. How many iterations could this include before submitting a bid in a request-for-proposal process, would depend on the time available before the deadline. We suggest at least two iterations, however, in our early experience it's our perception that four would be ideal. We note that in the context of the ERP Service estimation method, the term 'validation' means more than just 'double-checking' the effort estimates and repeatedly applying and re-applying the experts' reasoning that produced the estimates in the first place. It also include the activities that help make the tacit expert knowledge explicit in order to enable a high degree of reuse and to flag the level of accuracy between already proven FSM and suggested rules of thumb which still need to be validated. Below are examples of what could be used as part of this validation process. As these are just examples, we do not claim that our list is exhaustive. Indeed, it cannot be exhaustive because SAP experts involved in estimation might come up with additional techniques that help de-bias their estimates.

1) Feedback from others: This could be initiated via the use of post mortem reports or a forecasted versus actual comparison. Traditionally, ERP projects use the WBSs as project planning vehicles. Such WBSs can also contain a field for actual values which could be maintained by a project manager. These WBSs should then be provided back to the Project Management Office group or estimation team. This process can also be automated as the input field can be uploaded into a database.

2) Effort Variance: A calculated value can be created to indicate the accurateness of the FSM estimates and rules of thumb delivered via expert judgment when available. This could provide more insight during estimation and validation of effort estimates.

$$EffortVariance = \frac{ActualEffort - EstimatedEffort}{EstimatedEffort}$$

3) Group based effort estimation: Group based effort estimation also help increasing the accuracy of an estimate by consolidating expert's experiences while validating the project scope. Often, making use of an estimation counsel provides a deeper insight to the possible scenarios and this could all them be accounted for. This

also includes using a second expert for an second opinion, and even a third expert for a third opinion.

4) Top-down and bottom-up estimation validation: The estimates provided for the top-down components is often improved by rolling up the effort from its subcomponents. Once the leaf nodes of a subcomponent have been verified it could then be used and rolled up to its high-level components for future use.

Depending on the estimation strategy used to create an estimate for a certain scenario (Baseline, Configurable or Tailored), one could either decide to rely solely on the FSM or expert judgment (rules of thumb) or a combination of both during validation phase. The main idea is to use, consider and leverage both estimation techniques (FSM or expert judgment) during validation, therefore leveraging the feedback from both techniques. The general rule is when an activity is commonly repeated, project after project, FSM could in theory be more accurate as there is little unknown in these cases and the repeated averages for projects over time build a good foundation for an accurate FSM and estimation. In cases where your estimate is built on a Configurable or Tailored strategy one could shift the emphasis on using expert advice (rules of thumb). Saying that it should also become clear that as these customized scenarios are reused and repeated, an estimator could indicate and trigger the transformation of implicit into explicit information, therefore been converted from expert judgment into FSM's.

Fig. 6 provides an example of a database which contains historical project estimates per scenario. The example is about the application domain of processing data for materials in a data warehouse, a standard SAP business process for interfacing ERP master data into a Business Intelligence data warehouse.

Fig. 6 presents three examples labeled as **A**, **B** and **C**. Each one takes one row, respectively. In the figure, the columns in blue indicate the case when the FSM of COSMIC [3] is used and there is a database that stores size and effort information measured in COSMIC Function Points (CFP). The columns in red are about the case when expert judgment is used and there is a database storing expert judgments. We used "null" in some cells in Figure 6, to indicate that data is not available. Last, the blue and red columns named 'Effort variance' indicate how great the deviation is between the initial estimates and the 'validated' estimates.

Example A (Fig. 6) would be a scenario forming part of the solution core implementation and being repeated in multiple projects. The estimation strategy chosen to provide the estimate could have been a Baseline. Therefore, it is suggesting the use of FSM if available. During validation we could decide to compare the estimate against the rule of thumb, while having a high degree of confidence that the frequency count and validation key figure (actual project key figure in hours divided by function points, e.g. COSMIC FP per hour [3]).

Example B (the next row in Figure 6) would be a typical case where a configurable strategy was used. Therefore a FSM was created due to the trigger of the Expert Judgments frequency count, (indicating it's a scenario often being used). But as the FSM is still pre-mature (Frequency of 1 and representing a un-validated estimate) the main validation would then be done based on the rules of thumb. In some cases you can use the average between the rule of thumb and FSM in case the frequency count is more representable ( $> 3$ ). In certain cases you can exclude the outliers. This is

indicated by the effort variance, 0.1 indicating a wide spread between the individual estimates and 0.9 a narrow spread (variance) between estimates.

Example C (the last row in Figure 6) would be a very specific scenario which is not commonly found in projects and addresses a special requirement for a specific customer. Therefore the strategy tailored would be used and the validation solely based on historical rule of thumb. In this case we would not rely on FSM as it is not representable (frequency count < 3 or null). In general the higher the frequency count and closer to 1 for the effort variance, the more accurate the estimate.

Scenario example	FSM (COSMIC FP - CFP)			Rule of thumb (expert judgment)		
	FSM (COSMIC FP - CFP)	Frequency count	Effort variance	Rule of thumb (expert judgment)	Frequency count	Effort variance
A.Currency conversion	23 CFP (3 hours)	12 (frequently used)	0.9	Range (1-2 hours)	1 (seldom used)	null
B.Extracting product material data from a SAP DSiM via RFC connection	52 CFP (8 hours)	1 (seldom used)	null	Range (6-7 hours)	5	0.6
C.Extracting product material data from a 3 <sup>rd</sup> party ERP	null	null	null	Range (6-8 hours)	7 (frequently used)	0.8

**Fig. 6.** Validation Example.

## 7 Conclusions

This paper argues that requirements-based effort estimation in the ERP domain at in the contract stage, can be improved by deploying a set of appropriate methods and using them accordingly aligned with the needs and pressures within this industry. It's also important to create the conditions most favorable in a controlled, structured and consistent process to ensure we leverage on the strengths of these methods and reduce the problems associated with the method, such as in the case of Expert Judgment [16]. We acknowledge that one-size-fits-all mentality would not be a realistic solution to the early-requirements-based effort estimation in ERP projects. Instead, we argue for the complementary use of a variety of techniques and for choosing the combination of techniques based on contextual information, e.g. presence of database with FSM information, or a database with expert judgments. Our three estimation strategies depend on the project characteristics and resources available to the ERP estimating team or the project manager.

The Baseline strategy should be used for those requirements, scenarios and activities related to the baseline which represent the implementation of the core features relying on FSM.

The Configurable strategy is meant for a set of configurable activities using rules of thumb. These groups of pre-configured scenarios are often examples such as integrations scenarios. The rules of thumb can conceptually be converted into FSM estimates.

The tailored strategy should be used for those activities related to customer specific customization and rely on the knowledge and experience of experts. As these cases are very specific to a customer's demands it's often tailored specifically to fit a customer's processes. In this case we rely on expert judgment rules of thumb.

The ERP services effort estimation method [23] and the three strategies propose in this paper differentiate themselves from previous work [1,28,19] by:

- (1) taking both an objective and subjective views on the topic of expert judgment based effort estimation.
- (2) combining the learnings from three competing paradigms and schools of thought, namely expert judgment based estimation, effort estimation based on analogies, and FSM-based estimation [16].

Moreover, the method leverages the important lessons of social sciences into account, such as how experts play a role in effort estimation and how memory bias could affect estimations and how reconstructive memory could be managed to a certain degree. From software project estimation standpoint, the method also includes the lessons learned from the past research on expert judgment estimation, estimation by analogy and function point based estimation.

Using the three estimation strategies as described and suggested in this paper could make a contribution by improving the overall quality and accurateness of early estimates addressing the immediate problems of project managers (as indicated in Sect. 4.1). The ERP estimation strategies and toolset provide guidance to an estimator to ensure that the right estimation is used depending on the component or project type. Each type of project has its own biases and problems, e.g. Innovation and Ramp Up projects often lack an accurate and representable scope during early estimation cycles. Therefore relying on the guidance of the suggested strategy can encourage estimators and project managers to rely on rules of thumb and expert judgment where formal explicit documentation falls short. Reiterating the process steps several times (four times ideally) would ensure that previously overlooked components are identified with the right level of detail, addressing the necessary level of complexity.

This research has some limitations: First, the strategies have been tried out in a small number of projects. Although the results and the feedback by practitioners in the circle of colleagues of the first author, has been positive, we would need further application so that we could claim their wider applicability across a variety of contexts.

Second, the strategies have been formulated and tried out in the organization of one ERP vendor (SAP). A key question then would be whether would be a good idea for other vendors to consider using the strategies? Would they provide value in contexts of other ERP packages? We acknowledge that we cannot and should not claim universal generalizability of the usefulness of the strategies for estimation, because there is no way we could know in advance all possible scenarios in which ERP projects could be setup in all ERP adopting organizations in the ERP marketplace. However, as research methodologists suggest [24], while we cannot claim universal generalizability we could possibly expect that similar ERP implementation

organizations, e.g. big ERP service project organizations that engage multiple experts and share process-oriented thinking, would possibly make observations similar to those that we have in our research. To substantiate this by means of empirical evidence however, we plan to carry our further studies. Only then, we could provide more insightful lessons learned for practitioners outside our research organizations.

Third, as part of preparing for our research, we searched for previously published ERP effort estimation studies. We realize that even if we want to know the evidence related to those methods that were applied to ERP systems, we should not have limited our search for methods already applied to ERP systems. It might well be possible that there are methods not yet applied in this domain that could be helpful nevertheless. We consider this an important line for future research. We would like to invite other researchers working at the intersection of RE and contract-driven project delivery for large business systems to take part in such a research initiative.

## References

1. M. Daneva, Balancing uncertainty of context in ERP project estimation: an approach and a case study. *Journal of Software Maintenance and Evolution: Research and Practice*, 22 (5), 2010, 310-335.
2. H. Plattner and A. Zeier, *In-Memory Data Management An Inflection Point for Enterprise Applications*, Heidelberg: Springer, 2011.
3. I. P. Erasmus, *The COSMIC EPC method - An ERP functional size measurement method delivering time and cost estimates*, Goteborg University/Chalmers University, 2012.
4. F. Bolger, G. Wright, Assessing the quality of expert judgment: Issues and Analysis, *Decision Support Systems*, vol. 11, pp. 1-24, 1994.
5. J. B. Soll, Overconfidence in interval estimates. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, p. 299 – 314, 2004.
6. A. Magazinius, S. Börjesson, R. Feldt, Investigating intentional distortions in software cost estimation – An exploratory study, *Journal of Systems and Software*, 2012.
7. P. G. Benson, The Effects of Feedback and Training on the Performance of Probability Forecasters, *International Journal of Forecasting*, vol. 8, pp. 559-573, 1992.
8. A. Lederer, J. Prasad, Causes of inaccurate software development cost, *Systems Software*, vol. 31, pp. 125-134, 1995.
9. A. Lederer, J. Prasad, A causal model for software cost estimating error, *IEEE Transactions on Software Engineering*, vol. 24, pp. 137-148, 1998.
10. R. Buehler, D. Griffin and M. Ross, Exploring the “Planning Fallacy”: Why people underestimate their task completion times, *Journal of Personality and Social Psychology*, vol. 67, pp. 366-381, 1994.
11. H. & H. H. A. J, Cost Estimation of Software Intensive Projects: A Survey of Current Practices, in *Proceedings of the 13th International Conference on Software Engineering*, Austin, Texas, 1991.
12. W. M. Goldstein, R. M. Hogarth, *Research on Judgment and Decision Making: Currents, Connections, and Controversies*, Cambridge University Press, 1997.
13. P. J. Hinds, The curse of expertise: The effects of expertise and debiasing methods on predictions of novice performance., *Journal of Experimental Psychology*, p. 205–221, 1999.
14. L. J. Sanna, C. J. Parks, E. C. Chang and S. E. Carter, The hourglass is half full or half empty: Temporal framing and the group planning fallacy. *Group Dynamics, Theory, Research, and Practice*, p. 173–188, 2005.



15. S. J. Byram, Cognitive and motivational factors influencing time predictions, *Journal of Experimental Psychology*, p. 216–239, 1997.
16. M. Jørgensen, Forecasting of software development work effort: Evidence on expert judgement and formal models., *International Journal of Forecasting*, vol. 23, p. 449–462, 2007.
17. M. Jørgensen and T. M. Gruschke, The Impact of Lessons-Learned Sessions on Effort Estimation and Uncertainty Assessments, *IEEE Transactions on Software Engineering*, vol. 35, no. 3, pp. 368 - 383, 2009.
18. U. Passing and M. Shepperd, An experiment on software project size and effort estimation, Paper presented at the International Symposium on Empirical Software Engineering, 2003.
19. K. Molokken and M. Jørgensen, A review of surveys on software effort estimation, *International symposium of empirical software engineering*, p. 223–230, 2003.
20. M. Jørgensen and K. Moløkken, Eliminating over-confidence in software development effort estimates, *Product focused software process improvement*, vol. 3009, p. 174 –184, 2004.
21. N. Bajaj, A. Tyagi and R. Agarwal, Software estimation: A fuzzy approach, *SIGSOFT Software Engineering Notes*, vol. 31, p. 1–5, 2006.
22. J. Hill, L. C. Thomas and D. E. Allen, Experts' estimates of task durations in software development projects, *International Journal of Project Management*, vol. 18, p. 13–21, 2000.
23. P. Erasmus, M. Daneva, ERP Effort Estimation Based on Expert Judgments, *IWSM/Mensura 2013*: 104-109.
24. Wieringa, R., Daneva M., Six strategies for generalizing software engineering theories, *Science of Computer Programming*, Nov 2014, in press:  
<http://www.sciencedirect.com/science/article/pii/S0167642314005450>
25. R. O'Brien, An Overview of the Methodological Approach of Action Research," University of Gothenburg, 1998. [Online]. Available: <http://www.web.ca/~robrien/papers/xx%20ar%20final.htm>. [Accessed 2013].
26. T. Keller, *Accelerated SAP*, Addison-Wesley, 1999.
27. M. Daneva, Measuring Reuse of SAP Requirements: A Model-Based Approach, *SSR 1999*: 141-150
28. M. Daneva, Understanding Functional Reuse of ERP Requirements in the Telecommunication Sector: An Empirical Study, *IWSM/Mensura 2014*: 216 – 221
29. T. Brown, *Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation*, Harper Business, 2009