

Don't make a mess of software subcontracting!

on MASS: a method for subcontractor selection in software application development

Danilo Assmann and Teade Punter

Fraunhofer IESE, Sauerwiesen 6, D - 67661 Kaiserslautern, Germany

Email: {assmann, punter} @iese.fhg.de

Abstract

As the amount of software in today's products increases also the need for subcontracting assessment arises. Frequently questions are posed like: 'What is the quality of my subcontractor? Or, is this company able to produce part(s) of our software?' Although, posing the question is easy, answering it is confronted the problem: 'How to assess the supposed subcontractor while its product does not exist yet?' A range of assessment methods has been defined over the years as the result of in-house as well as public standardization projects. In this paper the authors analyse these existing methods and propose a 'best-off' approach: the so-called MASS: Method for Assessing Software Subcontractors.

Keywords

Subcontracting, software component development, subcontractor selection, procurement, Euro Method.

INTRODUCTION

Outsourcing is a trend in design and manufactory for a longer time and it will be the adage for the next years too. It is worldwide a fast growing market. The overall value was around \$99 billion in 1998. By 2003 it is expected that an overall value of \$151 billion will be spent [13]. More and more companies in the software domain use subcontractors or work as subcontractors to provide pieces of a system solution. *Application Service Providers (ASP)* is a specific type of subcontractors involved in outsourcing. The ASP-market will grow from \$1 billion in 1999 to more than \$25 billion by 2004 according to Dataquest [4] and \$49 billion in 2006 according to Ovum [18].

In general three categories of outsourcing are distinguished [13]:

- Outsourcing of IT infrastructure – is the most general kind of outsourcing. Application Service Providing (ASP) is an example.
- Maintenance of existing legacy systems – is especially relevant for companies, that are faced with a change in the business focus, but that have existing contracts that require service for the complete life cycle of the software, see e.g. [26].
- Application development – is the development of new software applications in e.g. automotive industry and consumer electronics. Often these are complex applications that consist of many different hardware as well as software components. Managing such complex development projects is really a challenge.

This paper focuses on the third type: subcontracting in application development, during which software engineering principles [23] are applied. We specifically address the relationship between a contractor – outsourcing its development – and its subcontractors – doing the development or providing Commercial-Off-The-Shelf (COTS) or Modified-Off-The-Shelf (MOTS). An example of levels in subcontracting relationships is expressed by the 3-Tier relationship in Figure 1. These relationships are quite normal in e.g. subcontracting in automotive industry and may result in complex project management.

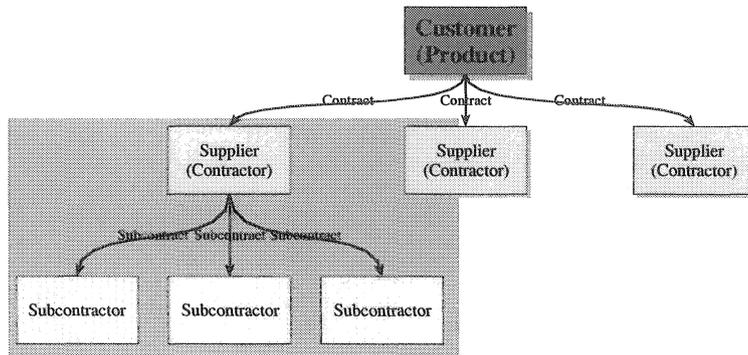


Figure 1 - Tiers in contracting - subcontracting relationships

Software development enables as well as requires subcontracting. Because of its intangibility and its easy reproducibility, different subcontractors can principally develop software products without having high transport costs and other barriers that other products normally face in subcontracting. Software development requires subcontracting because of its specific development knowledge as well as its way of time-to-market driven developing. It is obvious that the customers of the application – that might be a solution provider, system integrator as well as the manufacturer – have a need for reliable contractors and subcontractors. It has become crucial to know subcontractor's capability and maturity before selecting the company as a subcontractor – and it is even more crucial for software than in other domains, as will be shown later.

However, in practice the awareness of companies for the risks that they are faced in with subcontracting is rather low. This is true for contractors as well as for

subcontractors. The selection of subcontractors and also the monitoring of those subcontractors during their contract are often not performed according to defined and understood processes. Two problems are identified based upon literature²³ study and analyzing industrial cases, namely:

- Subcontracting projects are finishing too late and not within budget. They are not conducted within time constraints as well as that they are often more expensive than expected. For example because of unrealistic estimation of cost, time and manpower requirements or because of insufficient managing the risk associated with those complex software projects.
- The expectations about its outcomes do often not match its actual results, so that it will be hard to apply the results. For example because the requirements were too complex and difficult to identify.

We argue that defining and implementing a well-defined software acquisition process can avoid a lot of these problems. Such an acquisition process consists mainly of three phases, namely: selection, monitoring and completion. The selection is synonymous with tendering and is about selecting the subcontractor. This phase ends with the contract signing. The monitoring starts with the signed contract and follows the subcontractors' work till the product is delivered. The last phase of the acquisition, and also ending the acquisition is completion, or completion of contract. This includes acceptance and installation of the product, and in many cases also the maintenance of the product over its lifetime. Figure 2 presents the three stages in the acquisition process.

In this paper we focus on the selection of subcontractors. The paper analyses existing methods for subcontracting processes. Based upon this analysis we conclude that a more detailed approach is necessary. This approach, the so-called MASS-method (Method for Assessing Software Subcontractors).

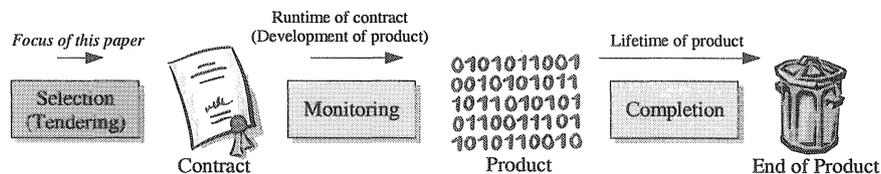


Figure 2 - Acquisition consists of selection, monitoring, and completion.

EXISTING STANDARDS AND METHODS

This section analyses existing standards for subcontracting and points out the problems in applying them for software subcontractor selection. A considerable amount of standards on subcontracting of software has been published over the years. Main standards in general as well are: SW-CMM [19, 20], SA-CMM [3], ISO 9000 [ISO9001a,b], [ISO9000-3], ISO 15504 [ISO15504-1/2/8], Bootstrap,

²³ The listed problems are mainly taken from [6, 7, 15, 27], but verified through many other sources.

EuroMethod [5], IEEE Recommended Practice for Software Acquisition [8], and Software Program Managers Network Model.

Analysis

We have analyzed those standards and methods by comparing them to a reference process that was defined by taking the elements that existing standards or methods applied successfully for some time, namely [1]:

- Identification of Need – be sure that there is a real need for the acquisition. This criterion is so obvious that it has to be stated in standards.
- Definition of goals – goals for the acquisition have to be described. If there is a need, it will be likely to define the goals easily.
- Planning – set up a high level plan for the acquisition.
- Development of strategy – the strategy is the continuation of the acquisition plan, but on a lower level. It concerns about the concrete implementation of the acquisition.
- Definition of work – develop and compile the statement of work (SOW).
- Preselection – make a first selection of probable candidates. This selection should be reliable, but overall cheap!
- Preparation of RFP – develop the request for proposal.
- Prepare responses – this task focuses on the subcontractor. Here the subcontractors have to prepare the responses on the RFP.
- Evaluation – evaluate the responding subcontractors. Use the method and criteria defined in the acquisition plan and refined in the strategy.
- Selection – make a decision based on the evaluation and inform all subcontractors.
- Definition of acceptance criteria – set-up the acceptance criteria for the product.
- Preparation of contract – develop and negotiate the contract. Both parties sign it.

Based upon this reference process, the existing standards and methods were analyzed. The results are presented in the next table (Table 1). In Table 1 X implies yes/fulfils the requested criterion, and (X) fulfils the requested criterion implicitly/is very close to the criterion.

Our analysis shows that the standards cover important elements of the reference process. Standards that are often used to conduct process assessments on subcontracting organisation, like SW-CMM, ISO 9000 and CMMi, cover a considerable number of elements but not as much as the others do. Meanwhile the ISO 15504 –that can be seen as the reference for these process assessment approaches-covers almost all elements.

EuroMethod and IEEE 1062 are two standards that can be regarded as methods to implement subcontracting assessment. However, each has its own drawback for subcontractor selection. EuroMethod covers any type of acquisition in the IT domain and thus becomes complicated in usage. IEEE 1062 has a good and ‘crisp’ process structure, but its description and information is too much on a checklist level. Therefore it is more a guideline than a method.

The majority of the approaches focus on a general customer supplier relationships. Even for this field there is no general approach for the selection and management of a supplier that covers nearly all of the suggested aspects. EuroMethod has the advantage over the other approaches that it includes the subcontractor perspective as well. The disadvantage of EuroMethod is its complexity.

Besides the coverage of the reference process, we have also looked at the usability of the approaches. For this reason the extent to which the approaches providing guidelines and other supporting information was analyzed. The results are depicted in table 1.

Table 2 shows that the support of most of the standards for software subcontracting is quite low. Even EuroMethod and IEEE 1062, which were detected as methods, are not impressive. None of the methods is applicable at once or 'out-of-the-box'. This mainly originates from the specific characteristics of software subcontracting.

Table 1 - Standards and methods for subcontractor selection compared to reference process

Acquisition Elements	SW-CMM	SA-CMM	ISO 9000	ISO 15504	EuroMethod	IEEE 1062	CMMI
Identification of Need				x			
Definition of Goals				(x)	x		
Planning	x	x		x	x	x	x
Development of Strategy		x		x	x		
Definition of Work (SOW)	x	(x)	(x)	x	(x)	x	x
Preselection						x	
Preparation of RFP		(x)		(x)	x		
Prepare Responses					x	x	
Evaluation	x	x	x	(x)	x	x	x
Selection	x	x	x	x	x	x	x
Definition of Acceptance Criteria		(x) ²⁴	x	x		x	
Preparation of Contract		(x)	x	x	x	x	

Table 2 - Standards and methods for subcontractor selection: support with guidelines

Acquisition Elements	SW-CMM	SA-CMM	ISO 9000	ISO 15504	EuroMethod	IEEE 1062	CMMI
Resources and Funding	x	x					x
Acquisition Management							

²⁴ Separate Element-KPA: Evaluation. It is not performed in the time frame of selection and has a different meaning.

Acquisition Elements	SW-CMM	SA-CMM	ISO 9000	ISO 15504	EuroMethod	IEEE 1062	CMMI
Organizational Policy/Objectives	x	(x)				(x)	x
Organizational Strategy						x	
Templates for Documents							
General Practices					x	x	
General Terminology					x		
Include External Help					x	x	
Use of Pilot Projects							
Tailoring of Process	(x)	(x)				x	(x)
Measurement		x					
Package Experience in Process	(x)	x					(x)
Quality Records			x				
Performance Prediction		x					

Characteristics of software subcontracting

When software engineering is compared to regular production engineering there are a lot of similarities when distinguishing phases like requirements engineering, design, and implementation. Despite of a similar structure on a high level, software application development is different from that in regular production. An important difference is that software is intangible and that its production is mainly compiling of source code. This has no equivalence with producing physical goods.

The distinction between regular production engineering and software engineering causes differences in the challenges for subcontracting in both fields. The challenge of subcontracting in 'normal' production engineering is to find a subcontractor that provides stable production quality of the component, reliable availability of the components for assembly in right numbers, flexibility for changes in component and in numbers, stable prices, after-production-run supply. For example in automotive industry the product is constructed and consists of multiple components. Components that do not belong to the core competence of the (prime) contractor will be subcontracted. Many of the components are standardized. Other components are customer specific, but even a number of them is only a customer specific adaptation of existing material and know-how. Contractor and subcontractor do development and construction of components jointly. Sometimes the cooperation is very tight and/or restricted by the contractor. The components become mature in the '0-series'. Changes and small adaptations will follow through the lifecycle of the product. For many components the exchange of subcontractors is possible in the production run.

In software development the contractor designs the product and identifies the components that will be subcontracted. Those components are not as standardized as in regular production. The subcontractor that will develop the component has much more influence on its final shape. Requirements can be changed easy and its impact can be shown with prototypes, but this has normally exceeded the project budget

dramatically. Any exchange of the subcontractor during the development is expensive and a high risk. Later changes or adaptations on the product have to be considered as project failures. The challenge for subcontracting in software application development is to find and select those subcontractors that provide the components according to specified quality, in time and within budget, without having seen any product by that time. The focus is on organisation and the processes and the subcontractor should be able to show that management of time, quality and cost is performed in a way that matches contractor's requirements.

The difference in focus for subcontractor selecting between software application development regular production engineering is not sufficiently supported by existing standards for subcontracting. Together with the other shortcoming presented before, see Analysis, we conclude that there is a need for a process and guidelines for subcontractor selection in software application development specifically. This has led to the development of MASS.

MASS: OVERVIEW AND MAIN STEPS

MASS-Method for Assessing Software Subcontractors - is a method that integrates the best off existing standards and methods in subcontractor selection and builds. The advantage is that MASS is compliant to all standards. MASS looks from a perspective of contractor to the subcontractor selection process, but recognizes at several points in the process that also the subcontractors should be taken into account. Every strategy for selecting, contracting and managing a supplier can only be highly successful if it used and can be used by both parties.

The main activities of the MASS process are presented in the Figure 3. Two levels are distinguished in those activities, namely: preparation and selection activities. The preparation tasks are meant to prepare the actual selection. The selection activities or main steps are presented in the current section.

Pre-select

Potential subcontractors should be identified at this stage in process. This is in fact the definition of a long list that contains all potentially subcontractors. The pre-selection is a quick and cheap assessment of all of these identified potential subcontractors. The result will be a selection of no more than 7-9 potential subcontractors that become members of the short list. Only these candidates will receive the Request for Proposal (RFP).

During the pre-selection the following characteristics of the subcontractors might be taken into account: its financial soundness, its experience, history and capabilities, its past Performance, its development and control processes, its technical assistance, its quality practices, its maintenance services, its product usage, its product warranty, its costs, its contracts.

One remark on subcontractor's past performance: "Contractors with a good record of delivering quality software within cost and on schedule are more likely to do the same in the future." This is a reasonable hypothesis only if there are a large number of similarities between the contract under consideration and the past performance being considered. This is called an implicit *ceteris paribus* (all other things being equal) assumption, and it is common in logical reasoning processes.

Whenever this type of implicit assumption is encountered, it is useful to enumerate the items that must be equal for the assumption to be valid and our hypothesis to be correct. The following seven necessary areas of similarity between the contractor's past performance and the contract under consideration should come to mind: Problem Domain; Software Processes; People; Management; Development Environment; Requirements Maturity; Cost Constraints. [16]

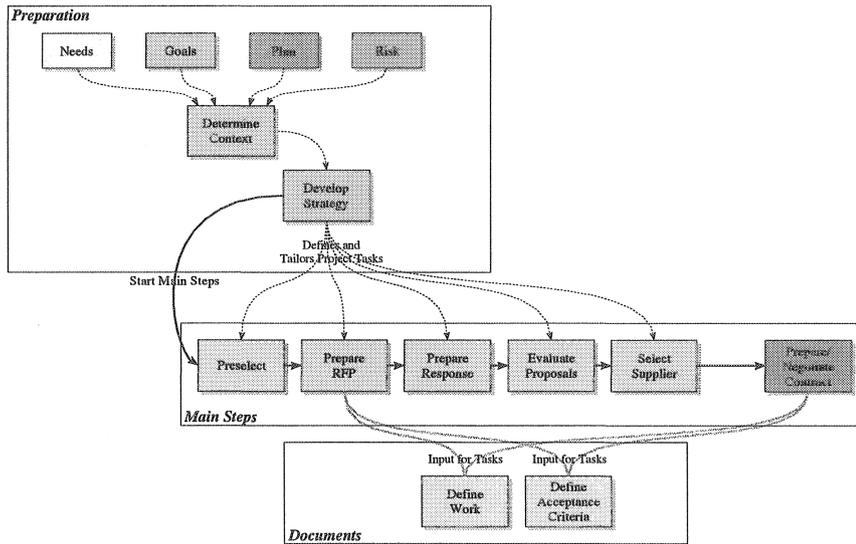


Figure 3 - Two levels in the MASS process: preparation and selection activities & relation to documentation

Prepare Request for Proposal (RFP)

Prepare and/or compile the Request for Proposal (RFP), and send it to the subcontractors on the short list. The following elements should be included in the RFP:

- Target domain of the procurement
- Objectives of the contract and the requirements to systems and services
- Organizational constraints and procedures
- Delivery Plan
- Constraints likely to affect the project (plans)
- Financial constraints/budget
- Legal clauses of the contract
- Subcontractor evaluation approach
- Instructions for bidding subcontractors
- Requirements to subcontractors' organization characteristics (derive criteria from standards or process capability and evaluation methods—this is important!)

Main input for the RFP is the statement of work, system and software requirements, and required standards and procedures. The statement of work will be

normally the design or the software architecture. Domain analysis and a set of scenarios will define this. The used criteria are divided in to types: Evaluation criteria for the evaluation and selection of the subcontractor before the contract. Acceptance criteria are applied to monitor the subcontractor after the contract has been signed. All criteria will be defined from three perspectives, namely: product quality, project development and process quality perspective (see Figure 4). For product quality the distinction between system and software should be taken into account. To specify product quality the ISO 9126 standard can be used as a vocabulary. Meanwhile this standard can be used to define system quality. For process quality the focus will be on the quality of the way of working by the subcontractor. The CMMI or ISO 15504 are standards that provide base practices, and with that criteria, for software processes. The project as such is determined by how the subcontractors' process will result into a quality product. Additional criteria for project concern time, cost and benefit. Next Figure provides a graphical of this taxonomy for both classes of criteria.

Prepare Response

During this activity the subcontractors and contractors should prepare the response on the RFP for their own perspective. The subcontractor should give all requested information. The contractor should prepare everything for the evaluation task, i.e., complete all needed documents.

The subcontractor should define the target domain. The same holds true for the definition of the proposed delivery plan that describes initial and final states, deliverables and decisions. Another issue is the project and service organizations and plans for this should be described as well. Charges and prices should be determined. This can be based on analogy to previous experience, mathematical models, and task decomposition or expert opinion. The response should also contain legal clauses concerning the liability of the subcontractor. Additionally the subcontractor should describe organization characteristics, like certifications, profiles (include staff), and assessment results. The advantages of the subcontractors proposal should be in the conclusion of the response

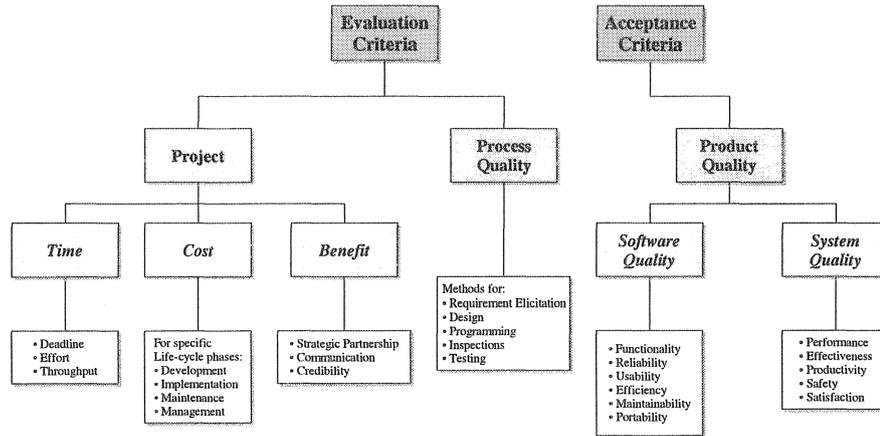


Figure 4 - Taxonomy of evaluation and acceptance criteria for subcontractor selection

Evaluate Subcontractors

This activity is the analysis of the response made by the potential subcontractors. The evaluation might be enriched by a demonstration or presentation of the subcontractor. Also a detailed (on site) assessment might be part of it. The process will include agreement on an evaluation score for each criterion, filling in the Response Evaluation Matrix (REM) and some other steps. We give some examples for evaluation aspects:

- Include proposals submitted for the planned subcontract.
- If available consider prior performance records on similar work.
- Take the geographic locations of the subcontract bidders' organizations relative to the prime contractor into account.
- Consider software engineering and software management capabilities.
- Include availability of staff to perform the work.
- Include availability of resources.
- Consider prior experience in similar applications, including software expertise on the subcontractor's software management team.
- If the subcontractor proposes the use of existing software products, then assess the product and get a demonstration.
- Consider any results observed during subcontractor demonstrations, and subcontractor facility visits.
- Determine for whom the subcontractor has produced work. Solicit comments from the subcontractor's prior customers.
- Costs should be compared to other subcontractor's prices and schedules.
- Caution should be exercised when the subcontractor's proposed costs are much higher or lower than the average of all costs received.
- Subcontractors that are not completely responsive to the requirements in the RFP should be eliminated from further considerations.

The evaluation will be conducted on basis of the evaluation criteria that were specified during Planning and in this task. The focus during the evaluation will be especially on the criteria concerning the project and the process quality perspective.

Process assessments, like Bootstrap or CMMi, can be applied specifically to assess process maturity.

Select Subcontractor

In this activity the 'final' decision is made. It is time to select the right subcontractor basing on the Subcontractor Evaluation Matrix (SEM), including the REM results, additional assessment data from presentations and assessments, and the interpersonal aspects and impressions. Also the Subcontractor Evaluation Report (SER) has to be prepared. Notify all subcontractors on the short list about the result of the selection. All of the evaluation assessment results and conclusions should be described within a SER. The purpose of the report is to:

- Document with whom the contract should be done
- Describe the way in which the evaluation has been conducted
- Provide a summary of the key points in the evaluation supporting the decision
- Provide a history and audit trail for the information system procurement

The subcontractor evaluation report provides the basis for the notices informing subcontractors about their selection or rejection.

Prepare and Negotiate Contract

Prepare the Requirements for the Contract, the Acceptance of the product, and the Management of the contract. Some general preparations have to be made including payment and non-performance remedies. Agreement between contractor and subcontractor, and ensure the mutual understanding between both parties.

Although project and the process criteria are taken into account during this activity the focus during the contract preparation and negotiation is especially on the acceptance or product criteria. Paying attention to these criteria means that actions for future monitoring of the subcontractors' work are formulated. This means that evaluation techniques, like inspections and code measurement, are defined. The applicability of those techniques depends on the information sources or artifacts that become available during the development process. Principally, three information sources can be distinguished. The first is the software code. This may be source code like if-then-else statements in software modules. But also the compiled system, and therefore an executable system, is regarded as this. The second type of information sources is product documentation, like: requirements specifications, design-, technical-, functional and user documentation. Also test reports; problem reports and change requests might be interesting as information sources for future monitoring. The third category is the people. These are the engineers, testers, maintainers that have knowledge about the system. It may be important to have access to them. The techniques for future monitoring will be specified on behalf of the expected availability of information sources during the life-cycle phases of the product as well as the required quality of the product. Next table presents an overview about when for selecting appropriate techniques according to these dimensions.

Quality focus	Information source	Life-cycle phase
---------------	--------------------	------------------

	Functionality	Reliability	Usability	Efficiency	Maintain.	Portability	Code	Document	People	Req. elic.	Design	Coding	Implemen.
Inspections/reviews	X							X		X	X	X	
Testing	X		X				X	X				X	X
Architecture analysis [12]	X	X		X	X	X		X	X	X	X		
Code measurement, conformance checking		X			X		X					X	
Walkthrough / Heuristic evaluation									X	X	X		
Audit (checklists)		X	X	X	X	X		X	X		X	X	X

Figure 5 - Selecting techniques by taking quality focus, information source and life-cycle phase into account

After both parties sign the contract the subcontractor selection process has finished. This is also the moment that MASS ends. Now the monitoring of the subcontractor starts.

MASS: PREPARATION

The central idea during the preparation of MASS is that the perfect subcontractor does not exist. Instead there are one or more best matching subcontractors in the given project context. Therefore the selection process is context dependent. Evaluation is the central activity of the main steps of MASS. The way the evaluation is determined-and especially the evaluation criteria and the methods - is impacted by the strategy. During the preparation it is tried to capture the context and thus prepare a reliable evaluation.

Acquisition strategy

Central element for the preparation is the determination of the acquisition strategy. The basic intention of the acquisition strategy is to answer the question 'How to acquire?' The strategy tailors the chosen acquisition method to the concrete need of this acquisition and addresses therefore explicitly the risks that might lead to failures in the acquisition. Concerning MASS, the strategy is included to simplify tailoring and adaptation of MASS. The strategy will, for example, describe which of the elements of bulleted list in the task Evaluation will be executed in a specific project. Further it is separated form the Acquisition Plan to be better maintainable. Acquisition Plans (conforming the standards) tend to be huge document, which are not easy to edit. Thus we propose a linked Acquisition Plan consisting of multiple documents and the usage of a separate strategy, which will be adapted more often.

The strategy –and with it the context for a subcontractor assessment - is mainly influenced by three factors, namely: needs, goals and risks. Besides the strategy is constructed by the acquisition plan.

Needs – it is important to start an acquisition only if there is a real need, otherwise you can save the money by not purchasing anything. Further, these needs have to be documented, to be able to revise a decision, if necessary.

Goals – the acquisition should be based on a (set of) goals for the future subcontractor relationship that produces the application at reasonable cost as soon as possible.

Principally, these goals include derivations of business goals (for the contractor and the subcontractor, at least if there is a chance for a win-win relation). Furthermore, there should be project specific goals that cover and describe the expectations of the project manager and team. Also resources and constraints that might affect the acquisition (for both parties) should be formulated.

Syntactical requirements for goal formulation are helpful to explicit the goal formulation. Often goals are formulated e.g. too vague being not operational or are conflicting to other formulated goals. Therefore the checkpoints in have to be regarded during goal formulation (additional the relations between goals should be checked).

Risks – Analyze and document risks for this acquisition. We will not propose a new method for risk management here. In most cases the risk management in place should be used. For an acquisition we separate two types of risks: risks that might occur during the selection phase and risks that might affect the monitoring. The risks of the selection are covered in the Acquisition Strategy via contingency plans. When the result of the selection is known a risk-list for the monitoring phase will be defined. This risk-list provides also valuable input for the contract. For example, depending on the maturity of the chosen subcontractor instruments for the monitoring will be chosen. Assuming a prediction scale from '0' (performs not good at all) to '5' (performs very good, at high probability) for the product of an subcontractor, based on the evaluation, a number '5' subcontractor requires little monitoring instruments. Whereas for a number '2' subcontractor multiple instruments, as participating on reviews, inspections, periodical test observation, required usage of certain tools (controlling code, code metrics and quality), strong communication (especially on change request), and process monitoring should be planned and integrated in the contract.

Table 3 - Checklist goal formulation.

- | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none"> 1. Realistic: The goal has to be realistic. The resources and conditions must allow the fulfilment of the goal. 2. Operationally: Goals have to be defined by Content, Extent, Time, and Responsibility as detailed as possible and necessary. This is important to avoid misunderstandings in communication and to allow the control of the fulfilment. 3. Order: The relationship between the goals and the priorities have to be defined. 4. Consistency: The goals have to be consistently defined. Nevertheless, there might exist some partial goal conflicts. |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

5. **Actuality:** The goal system should not include (sub -) goals that are old or have been given-up.
6. **Completeness:** The goal system should include all important goals at minimum. This should help to identify wrong priorities or hidden conflicts.
7. **Ability to Accomplish:** The goals should be composed in a way that they will be accepted and increase the motivation needed to implement them.
8. **Organizational Congruence:** Defined goals should fit to the organization in a bijective way. This means that goals should not offend against organizational conditions.
9. **Transparency and Means of Checking:** The goal system should be clear, comprehensible, well-structured, and checkable.

Looking at these three factors provides information to define the appropriate acquisition plan and development strategy. Other factors can be defined as well, but many of them e.g., architectural concerns, are already covered when defining the Acquisition Plan.

Plan – the acquisition plan includes the planning process that is guided by a document. The activity is also concerned about the analysis of risks and preparing the decisions. The following issues are addressed typically during this activity:

- Size of procurement
- What will be acquired?
- How much will be acquired?
- When will it be acquired?
- For how long will it be needed?
- How will it be acquired?
- How much will it cost?
- Do I have funding? Does it match the need?
- Description of the “point of contract”

Scenarios

Based upon the captured context, a tailoring of the main steps can be defined for the strategy. Principally, three levels of tailoring or also called scenarios can be distinguished to define the process for selecting subcontractors, namely:

- **Minimal process** – the project is easily managed and the purpose of the acquisition processes is achieved with a low level of rigor. The processes contain just the activities without which the project would really be in trouble. This level is suitable for low-risk projects.
- **Controlled process** – the project has a tangible complexity and a reasonable risk of going out of hands. To contain the complexity and risks, a higher level of formality is appropriate for the acquisition processes. The processes on this level contain the activities of the minimal processes and add some activities that are not absolutely necessary but make things more controlled and reliable. This level is suitable for medium-risk projects and more ambitious goals.
- **Robust process** – the project is highly complex and has a strong tendency to get chaotic unless it is managed appropriately and performed with discipline. The processes contain activities that increase the formality and administrative effort

but in return make things more robust and reliable. This level is suitable for high-risk projects and highly ambitious goals.

These scenarios should be taken as a guideline to when defining your own subcontractor selection process.

CONCLUSIONS

This paper has provided a comparison of the main standards on subcontractor selection. Existing approaches focus on general customer supplier relationships. Euro Method is the exception and has also the advantage over the other approaches that it includes the subcontractor perspective as well. However, the method is complex too. Another problem with the existing approaches is that the specific focus that is needed for subcontractor selection in application development is insufficiently supported. This makes the existing methods not particularly suitable for selecting subcontractors in software application development.

This analysis has led to the development of MASS: a Method for Assessing Software Subcontractors. The method aims at a simpler process than EuroMethod. This should guarantee that applying the method is feasible in practice. Meanwhile the method integrates the best-off features of the existing standards and is therefore compliant to those standards. The method is meant to be applied by prime contractors, although it takes also the subcontractor perspective into account.

REFERENCES

- [1] Danilo Assmann. MASS—a Method for Assessing Software Subcontractors. University Kaiserslautern and Fraunhofer IESE, 2000.
- [2] William J. Baumol, Alan S. Blinder. Economics – Principles and Policy. The Dryden Press; Harcourt Brace & Company, Seventh Edition, 1998.
- [3] Jack Cooper, Matthew Fisher, S. Wayne Sherer. Software Acquisition Capability Maturity Model (SA-CMM) Version 1.02. CMU/SEI-99-002. www.sei.cmu.edu
- [4] Dataquest, The State of Today's Application Service Provider Marketplace, 2000
- [5] EuroMethod Project. EuroMethod Version 1. July 1996. www.fast.de/EuroMethod.
- [6] Thomas Gartner, Gerhard Getto, Joachim Weisbrod. Software acquisition processes at DaimlerChrysler AG: Research activities for improvement. SPI – The European Conference on Software Process Improvement. December 1999.
- [7] Gerhard Getto. Evaluation von Prozessen und Methoden für Softwarebeschaffung und Management von Auftragnehmern. Presentation SQM Congress. 1999.
- [8] IEEE Std 1062-1993. IEEE Recommended Practice for Software Acquisition.
- [9] [ISO15504-1/2/8]: ISO/IES TR 15504-1: 1998(E). ISO/IES TR 15504-2: 1998(E). ISO/IES TR 15504-8: 1998(E).
- [10] [ISO9001a/b]: EN ISO 9001: 1994. ISO/CD2 9001: 2000. DIN EN ISO 9001: 1999-05. *Draft*.
- [11] [ISO9000-3]: EN ISO 9000-3: 1997. Normen zum Qualitätsmanagement und zur Qualitätssicherung — Leitfaden für die Anwendung von ISO 9001 auf die Entwicklung, Lieferung und Wartung von Software.

- [12] Kazman, R., et al, The Architecture Tradeoff Analysis Method, in: 4th International Conference on Engineering of Complex Computer Systems, augustus 1998.
- [13] Michael C. Mah. Managing Outsourcing Expectations: Productivity Benchmarks, Baselines, Service Levels, and Other Quandaries. Cutter IT Journal, Vol. 12, No. 10. October 1999.
- [14] Marty McCaffrey. Offshore Outsourcing: The Alternatives, Key Countries, and Major Challenges. Cutter IT Journal, Vol. 12, No. 10. October 1999.
- [15] Norman P. Moreau. Software Acquisition Strategies--A Look at Current Practices. 52nd Annual Quality Congress Proceedings, pp. 831 - 839. May 4-6, 1998. Philadelphia.
- [16] Ronald W. Morrison. Assessing Past Performance: When Is It Useful, When Is It Not? CrossTalk, August 1997. stsc.hill.af.mil/CrossTalk/1997/aug.
- [17] Jim Nielsen. A Subcontractor Management Tool Kit. Presentation, Software Engineering Consortium. 29. September 1999.
- [18] Hope, M & N. Khair, Overcoming Market Development Challenges, Ovum, July 2001
- [19] Mark C. Paulk, Bill Curtis, Mary Beth Chrisis, Charles V. Weber. Capability Maturity Model for Software, Version 1.1. Software Engineering Institute, February 1993. CMU/SEI-93-TR-24. www.sei.cmu.edu.
- [20] Mark C. Paulk, Charles V. Weber, Suzanne M. Garcia, Mary Bess Chrissis, Marilyn W. Bush. Key Practices of the Capability Maturity Model, Version 1.1. Software Engineering Institute, February 1993. CMU/SEI-93-TR-25. www.sei.cmu.edu.
- [21] Software Subcontract Management--Practitioner's Guidebook. 17. December 1997. www.sepo.nosc.mil/ssmpg/ssmpg.html.
- [22] Charles C. Poirier, Stephen E. Reiter. Die optimale Wertschöpfungskette -- Wie Lieferanten, Produzenten und Handel bestens zusammenarbeiten. Campus Verlag, 1997.
- [23] Ian Sommerville, Software engineering, Wokingham, Addison-Wesley, 1995
- [24] Daniel Svennberg. Software Acquisition Management Guidelines. University Linköping, 2001.
- [25] Bonnie R. Troup, Brian P. Gallagher. Using Contractor Evaluations to Reduce Software Development Risk. CrossTalk, August 1999. stsc.hill.af.mil/CrossTalk/1999/aug.
- [26] Jeffrey Voas, Maintaining Component-based systems, in: IEEE Software, pp. 22 - 27, July/August 1998.
- [27] Thimo Weiss. Experiences with Subcontractor Management. Presentation, Software Engineering Consortium. 27. September 1999.