OUTSHORE Maturity Model: Assistance of Software Offshore Outsourcing Decisions

Stefanie Betz[1], Juho Mäkiö[2], Andreas Oberweis[1]

[2]Forschungszentrum Informatik FZI
Haid-und-Neu-Str. 10-14
76131 Karlsruhe
maekioe@fzi.de
[1]Institute of Applied Informatics and Formal Description Methods
Universität Karlsruhe (TH), 76128 Karlsruhe, Germany
{betz, oberweis}@aifb.uni-karlsruhe.de

Abstract: *Offshore outsourcing software development is increasingly used by software industry. An outsourced software development project is geographically distributed software development. Compared with traditional ways of software development (in-house), the geographically distributed software development is more risky making the actual attainment of the expected results uncertain. Although the failure in an offshore outsourcing software development project is usually caused by various reasons and one of the major challenges is the geographical distance, we argue that the risk avoidance of an outshore software development project should start before the software development begins. This may be done by measuring the offshore outsourcing suitability of the software project and of the offshore outsourcing company. Therefore we have developed the so-called OUTSHORE Maturity Model, OMM.*

Introduction

Outsourcing of software development means that an organization wholly or partially contracts out its software development activities to another organisation. If the partner organisation is located abroad, this is referred to as "offshore outsourcing of software development". Inland or abroad, in both cases outsourcing may be a risky undertaking. However, many companies use offshore outsourcing in order to speed up the time-to-market, to access the global resources, to profit from the around-the-clock development, and to reduce the costs.

The goal of outsourcing software development activities is to uphold competitiveness in the global market. The goal is to be achieved by tactfully implementing all resources, information technology, skills and knowledge to guarantee a successful offshore outsourcing project. This is done in order to assist the company in its further global establishment. However, the global distribution of the development activities causes a number of unanswered questions about realisation and successful implementation. There actually exists a huge gap between the targets and the realised results. Thus, outsourcing studies report that 47 percent of buyers terminated their offshore outsourcing relationship abnormally. The reason for the termination was in 45 percent of the cases the poor provider performance and in 47 percent missing cost savings [Di06]. Furthermore, the quality of the work done by the offshore team is reported to be worse than the quality of in-house development projects in about 45 percent and unusable in 9 -15 % of cases [Mo04].

Risk is commonly inherent in business decisions. This is indeed the case in offshore outsourcing software development projects. Boehm states that the post-mortems of most unsuccessful software projects indicate that their failure could have been avoided, "if there had been an explicit early concern with identifying and resolving their high-risk elements" [Bo91]. However, risk avoidance during software development is directly coupled with the software development process. This is insufficient in outsourcing projects. Although the failure of an offshore outsourcing software development project may be caused by various reasons and one of the major challenges is geographical distance [CR06], we argue according to Boehm that the risk avoidance of a distributed software development project should start before the project begins. This may be done by measuring the offshore outsourcing suitability of the software project and of the offshore outsourcing company. Therefore we have developed the so-called OUTSHORE Maturity Model, OMM.

Related Works

In software industry maturity models are used to improve organisations, processes, technology and people in order to achieve better performance. Generally, these models measure a starting maturity level for an organization and indicate how next higher levels may be reached (cf. [Hum90], [Cur95]). The importance of maturity models is reasoned by the fact that product quality depends directly on the process maturity. Therefore, a number of maturity models have been developed to support process improvements in software development and describe the evolution of large and complex IS organizations (e.g. [Gr72], [GS03]. Depending on their intended purpose, various maturity models focus on different aspects of software development. Their common goal, however, is to help organizations to improve their software engineering management practices. Table 1 gives an overview about the commonly used maturity model CMM.

Maturity Model	**Focus**
P-CMM (People Capability Maturity Model)	The kind of attention team members should get from management during the systems development process [Cur95]
DMM (Documentation Maturity Model)	Assessment of the quality of software system documentation used in aiding program understanding [Pie02].
MMAST (Maturity Model for Automated Software Testing)	The improvement of an effective software verification and validation [Hum89].
IDEAL (Initiating – Diagnosing – Establishing – Acting – Leveraging)	Provides a path of actions that constitute a software process improvement program [Kau00]. The path defines five phases to go when an organization gathers a higher maturity level.
SE-CMM (Systems Engineering Capability Maturity Model)	Assistance in coordinating and publishing a model that would foster improvement in the systems engineering process [SEI95].
SA-CMM (Software Acquisition Capability Maturity Model)	The software acquisition capability of an organization [SEI02]. Defines five maturity levels and several Key Process Areas (KPAs) to be considered.
CMMI (Capability Maturity Model Integration)	Defines goals that must be fulfilled by a set of processes within the organization in order to increase the productivity of software development as they become disciplined and controlled. CMMI consists of five maturity levels and 22 process areas.

CMMI –DEV (Capability Maturity Model Integration for Development)	Assistance in product and service development processes [SEI06]
CMMI-ACQ (Capability Maturity Model Integration for Acquisition)	Assistance in supply chain management, acquisition, and outsourcing processes in government and industry [SEI07]

Table 1: An overview of maturity models

Common to the maturity models is that they help software organizations progress along an evolutionary path from ad hoc, chaotic processes to mature, disciplined software processes [Her97]. Note that the focus is clearly on the improvement of processes.
It is well known that software project management is affected by multiple factors, such as communication, project schedules and plans, and personnel, as stated in [Bo81]. A strong attention is also given for well defined, managed and documented software processes in an organization as they are thought to have a significant payoff in terms of project success ([Cur92]). Therefore an entire development process with individual views across all organization levels is suggested (c.f. [Cur92]).

However, besides the software processes, communication and coordination mechanisms play a central role as success factors for distributed software development ([CR06], [Kra95]) as they may reduce the project uncertainty and improve performance. We argue that problems in communication and coordination are both not solely interrelated to the ongoing processes. These problems as well depend on the given buyer maturity and the particular given project. The other factor is the ability of the participating organisations on the structural level. Software offshoring maturity models are used to measure the ability of suppliers to deliver the promised performance. Some work has been made in the area of the maturity of buyer of distributed development services, e.g. the Sourcing of IT Work Offshore (SITO) stage model [CA02] and the model from [Mc03], which likewise describes a four-stage migration path for organizations that offshore IT work. For both the subject matter is the buyer's organization. However, insufficient work has been done to measure the buyers' ability to cooperate offshore at all as well as to measure the capability of one particular project for offshoring. The aim of the OUTSHORE Maturity Model, OMM, is to fill this gap.

The OMM Phase Model

An empirical study was carried out to assist in the understanding of the components of risk. In the first phase 29 experts in the field of distributed software development projects were interviewed. The analysis of the collected data is based on the Grounded Theory [Gl67]. Based on the results from the first two phases risk and decision factors were formulated. They were subsequently evaluated in respect to their importance in offshore software development and with reference to literature (cf. [AD99], [Mo05]) as well as with the aid of expert opinion. Based on that a number of risk, success and decision factors (e.g. communication problems, cultural distance, experience in offshoring, process maturity (c.f. [AD99], [Mo05]) were derived from the hypothesis.
Generally we may come to the conclusion that the companies are fully aware of the existing distributed software development project risks. Nonetheless, companies especially on the first few projects were not in a position to name or estimate the existing challenges or risks within their current situation. As stated in [CA02] and in [Mc03] offshore IT sourcing follows a stage model based on increasing maturity in the offshore outsourcing effort. At the beginning the buyer has no distributed development skills and at the highest level, global sourcing is a core competence.

Within the OMM the risk factors are assigned to the distributed software development process phases and the maturity of the buyer, vendor and project are measured. In the following we introduce the phases a distributed software development process goes through. Subsequently the OMM is presented.
The process consists of seven major phases (cf. Figure 1). In each phase different tasks requiring various kinds of information are executed. The depicted process is based upon our empirical study and literature review (e.g. [AW05], [Ba08], [So06]).

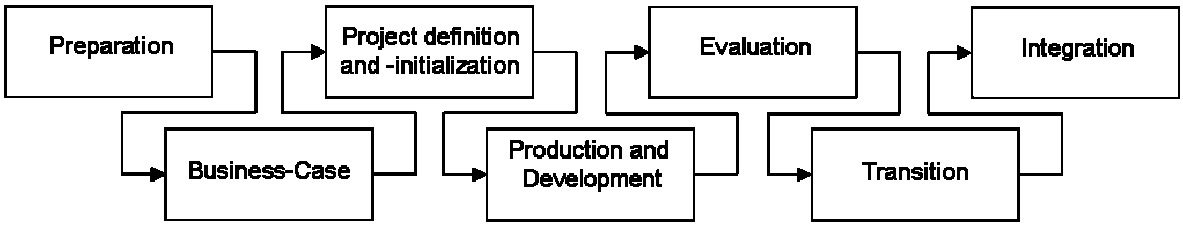

Figure 1: The OMM phase model

The first phase, *"Preparation"*, is devoted for the project preparations. The preparations conclude, among others, the search of a suitable project partner, check of the internal structures of the buyer organization with respect to its own abilities to start a distributed project, check of communication structures and skills of employees as well as the clear formulation of expected results. Furthermore, the main motivation for the outsourcing/offshoring has to be clarified and the employees need to be informed. Also the metrics to measure the success must be defined.

The second phase, "*Business-Case*", functions as a self-check for the buyer. The development processes are checked with respect to their capability to be executed by the vendor. The availability of personals skills within the buyer's organization and the suitability of the project for the outsourcing are alike to be checked. The required roles and responsibilities need to be clarified.

During the third phase, "*Project definition and initialization"*, the project deadlines and the teams are defined together with the outshore partner, and the functional and non-functional software requirements are communicated. The communication structures within the project are defined and, if possible, face-to-face meetings with enough time to know each other with all relevant people are planned in this phase. The main focus is on the downsizing of social barriers.

The following phase – "*Production and development*" – is devoted to the software development. During this phase control mechanisms and communication structures with the outsourcing partner and clear functioning communication links between the two teams have to be rechecked.

During the "*Evaluation*"-phase the results of the development phase are evaluated and tested. The information flow from the developers to the buyers and vice versa has to function properly. Lacking trust between the teams complicates this phase.

The "*Transition*"-phase aims at the knowledge transfer from the outshore provider to the buyer and it serves the buyer to avoid his dependency from the outshore provider. The knowledge transfer requires working communication mechanisms and trust between the employees on both sides.

During the "*Integration*"-phase the developed software is to be integrated with the existing systems. The integration contains the final testing of the new software and requires therefore a careful planning.

The OMM phase model specifies major phases for distributed software development and indicates tasks for each phase. However, the phase model does not specify the responsibilities of the tasks and it does not say anything about the dimensions of the maturity: buyer maturity, vendor maturity and project maturity (cf. Figure 2). In the following Section this aspect of OMM is discussed in more details.

The OUTSHORE Maturity Model

The OMM targets at the measurement of the company's outsourcing capability with respect to its suitability for the vendor and the suitability of the project to be outsourced. The OMM may be used to measure both offshore outsourcing and onshore outsourcing. The OMM provides a tool for risk management [BM08]. The gradual improvement should however be upheld, because the required activities can be derived from the model. OMM rates the buyers' maturity using the dimensions: strategy, experience in offshoring, operational structuring, maturity level to build up the risks, and hardware assistance (see Figure 2). The project maturity is measured using the dimensions modularity, size, duration, requirements stability, number of interfaces, type of the project, dependency of the company's internal knowledge and architecture. The third aspect – vendor - is measured by cultural and geographical distance, experience in offshoring / outsourcing, and fluctuation.
The goal of the OMM is to predict by means of various criteria whether a planned project that is to be outsourced, may be completed successfully. A successful completion means that the resulting software works as expected and the development is finished timely without financial overrun. An outshore offshoring specific maturity model is needed as the risks of offshore development differ strongly from in-house or onshore development. As aforementioned, the OMM measures the maturity in three dimensions. Each dimension contains elements that are essential for a successful offshore project. The OMM is organized into five maturity levels for each dimension, as depicted in Table 2. The calculated value indicates the ability of an organisation to use software offshoring with respect to that dimension. By doing so, we believe that an organisation recieves a better overview of its weak points. The values may be summarised into a combined OMM maturity.

Level	**Buyer Maturity**	**Project Maturity**	**Vendor Maturity**
Level 1: Initial	Almost no distributed software development project experience	Very complex and mission-critical projects	Grand cultural and geographical distance and no domain knowledge, almost no distributed software development project capability
Level 2: Started	Some exposure to distributed software development projects	Complex and critical projects	Considerable cultural and geographical distance, ad-hoc capabilities
Level 3: Aware	Distributed software development projects mechanism emerge	Non-core projects	cultural and geographical distance, strong capabilities
Level 4: Managed	Distributed software development projects	Mitigation or isolated new development	Some cultural and geographical distance, domain knowledge

	are understood and controlled		
Level 5: Mature	Global player, distributed developments a core competence	Simple, independent and large projects	Marginal cultural and geographical distance, no fluctuation, multi-shore supplier

Table 2: Level – Buyer, Vendor and Project Maturity

The maturity level is assessed by means of questionnaires. The questions are arranged into coherent areas and attached to the phases of the OMM phase model. This eases the delivery of hints about specific areas requiring improvement and the identification of the responsible persons or areas.

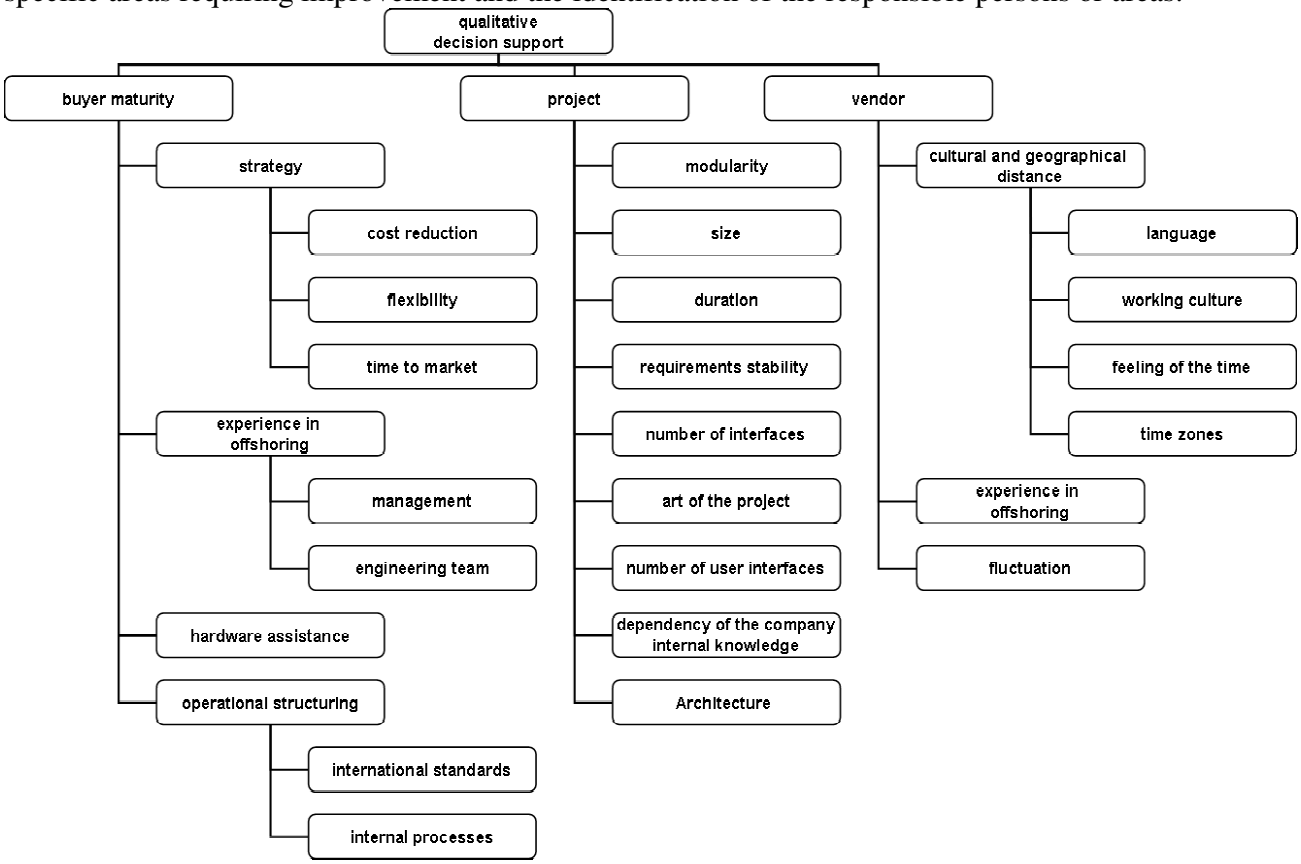

Figure 2: Dimension of the OMM for the qualitative decision support of software development projects

Figure 2 depicts the dimensions of the OMM. These dimensions are further subdivided into several risk factors. These factors have been weighted according to experts' experience as shown in Table 3.

risk factors	Weight of the influence
Technical questions	2,5
Infrastructure	2,5
Time zones	3
Political and legal framework	4,5
“Way of thinking”	5,5
Communication	6

Cultural differences	6,5
Project management	6,5

Table 3: Weight of the influence on the offshore outsourcing development project

The maturity level is evaluated from all dimensions and it is conducted as follows:

1. Verification through the response of questions →
2. Creation of a base for the evaluation of the offshore outsourcing activities →
3. Evaluation of the possible influence on the offshore outsourcing project →
4. Aggregation to a maturity level.

In the following we present some example questions which belong to the first phase *Preparation* of the OMM phase model. Generally these questions are aimed at the risk factor project_management, more specific they are aimed at the skill and people management:
Effectiveness of a project team:

- How many participants do the particular project teams have?
- Do you know the skills of the staff members?
- Do you know the availability of the staff members?
- Do you have the required specialists at hand?
- Do you know who instructs and thematically trains the required specialist?

Contribution to Theory and Practice

The main contribution of OMM to practice is that it offers companies a decision support system that checks their preparation for offshore outsourcing development projects. A decision matrix on the planned project as well as the vendor preferences in addition to the buyers' maturity level is needed in order to provide a qualitative decision support tool which considers all relevant factors of an offshoring decision [cf. AW05]. The dimensions of this decision matrix are depicted in Figure 2. OMM provides a rigorous basis for the a priori detection of the outsourcing availability of a software development process. The main contribution to the theory is that it offers an objective evaluation tool for offshore outsourcing software development projects and hence provides a stable fundament for further studies of software offshore outsourcing.

Summary and Outlook

We have described a novel maturity model to support software offshore outsourcing. We have also introduced a set of risk factors for successful offshore outsourcing software development based on interviews. These interviews are a starting point for our future studies. The results are reassigned into the OUTSHORE Maturity Model for decision support. We argue that this model provides new chances for risk analysis and planning of offshore outsourcing software development projects. We build a decision matrix upon which a company decides whether a planned software development project is appropriate for offshoring or not. Additionally it eases the delivery of hints about specific areas requiring improvement and the identification of the responsible persons or areas.
The presented OUTSHORE Maturity Model is developed by expert opinion and reference to literature on outsourcing, offshoring, global and distributed software development as well as existing maturity models. It is a work in progress from which practical guidelines for practitioners will emerge.

Acknowledgements
This research was developed within the co-operation project OUTSHORE funded by the German Federal Ministry of Education and Research (BMBF). OUTSHORE is a collaboration of the Forschungszentrum Informatik (FZI), the Institute AIFB of the University of

Karlsruhe and several industrial partners. The research project has a runtime of 36 months. The goal of the OUTSHORE project is to determine the effect of critical success factors on an offshore outsource software development project. A decision model, based on this critieria, will be created for the project run-time simulation to enable risk analysis in distributed software development projects.

References

[AD99] Aubert, B.A.; Dussault, S.; Patry, M.; Rivard, S., "Managing the risk of IT outsourcing," System Sciences, 1999. HICSS-32. Proceedings of the 32nd Annual Hawaii International Conference on Systeme Sciences, 1999.

[AW05] Amberg, M. and Wiener, M. „Kritische Erfolgsfaktoren für Offshore Softwareentwicklungsprojekte – eine explorative Studie“, Friedrich-Alexander-University, Erlangen-Nürnberg, 2005.

[Ba08] Balzert, H., “Softwaremanagement: Lehrbuch der Softwaretechnik“. Spektrum Akademischer Verlag; Auflage: 2. Aufl. (Februar 2008)

[BM08] Betz, S., Makiö , J.“ Applying the OUTSHORE approach for risk minimisation in offshore outsourcing of Software Development projects”, Multikonferenz Wirtschaftsinformatik (MKWI) 2008, February 2008, München, Germany

[Bo81] Boehm, B.W. Software Engineering Economics. Prentice-Hall Inc., Englewood Cliffs, NJ, 1981.

[Bo91] Boehm, B. “Software risk management: principles and practices”, Piscataway: IEEE Software, v. 8, p. 32-41, Jan. 1991.

[CA02] Carmel, E., and Agarwal, R., “Offshore Sourcing of IT Work”, MIS Quarterly Executive Vol. 1 No. 2,June 2002.

[CR06] Casey, V. and Richardson, I. “Uncovering the Reality Within Virtual Software Teams”,Workshop on Global Software Development for the Practitioner, Shanghai, China, pp 66-72, 2006.

[Cur92] Curtis, B., Kellner, M.I. and Over, J. Process Modeling. Commun. ACM 35, 9 , 75–90. 1992.

[Cur95] Curtis, B., Hefley W.E., Miller S.A. People Capability Maturity Model [P-CMM]. UK: Addison-Wesley. 1995.

[Di06] Diamondcluster 2006 Global IT Outsourcing study, available on http://www.diamondconsultants.com/PublicSite/ideas/perspectives/downloads/Outsourcing2006_Diamond.pd

[Did04] Dibbern, J., Goles, T., Hirschheim, R., and Jayatilaka, B. 2004. Information systems outsourcing: a survey and analysis of the literature. *SIGMIS Database* 35, 4 (Nov. 2004), 6-102.

[Gl67] Glaser B. and A. Strauss, “The discovery of grounded theory: Strategies of qualitative research”, Wiedenfeld and Nocholson, London, 1967.

[Gr72] Greiner, L.E., “Evolution and revolution as organisations grow”, Harvard Business Review, Vol. 50, No. 4, 1972.

[GS03] Galliers, R., and Sutherland, A.R., “The Evolving Information Systems Strategy”, Strategic Information Management, Elsevier, 2003 (Third Edition), Oxford.

[Her97] Herbsleb, J., Zubrow, D., Goldenson, D., Hayes, W., and Paulk, M. “Software quality and the Capability Maturity Model”. Commun. ACM 40, 6 (Jun. 1997), pp. 30-40, 1997.

[Hum89] Humphrey WS, Managing the Software Process, Reading, MA, Addison-Wesley, 1989.

[Hum90] Humphrey, W.S. Managing the software process. Reading, Massachusettes: Addison-Wesley Publishing, 1990.

[Kau00] Kautz, K., Hansen, H. W., and Thaysen, K. 2000. Applying and adjusting a software process improvement model in practice: the use of the IDEAL model in a small software enterprise. In Proceedings of the 22nd international Conference on Software Engineering (Limerick, Ireland, June 04 - 11, 2000).

[Kra95] Kraut, R.E. and Streeter, L.A. ,“Coordination in large scale software development”, Commun. ACM 38, 7 (1995), 69–81.

[Ma88] Mahoney, D., “Confessions of a Street-Smart Manager”, New York: Simon & Shuster, 1988.

[Mc03] McCarthy, J., Ferrusi Ross, C., and Schwaber, C.E., “Users’ Offshore Evolution and Its Governance Impact”, Forrester Research, December 2003.

[Mo04] Morales, A. W., “Outshore by Numbers”, available on http://www.ddj.com/architect/184415074”

[Mo05] Moczadlo, R., “Chancen und Risiken des offshore-Development – Empirische Analyse der Erfahrungen deutscher Unternehmen”, Germany, 2005, available on: http://www.competencesite.de/offshore.nsf/8FB68EAB823EF285C1256D72005BBCD1/$File/studie_offshore_prof_moczadlo.pdf

[Pie02] Pierce, R. and Tilley, S. “Automatically Connecting Documentation to Code with Rose”. Proceedings of the 20th Annual International Conference on Systems Documentation (SIGDOC 2002: October 20-23, 2002; Toronto, Canada), pp. 157-163. ACM Press: New York, NY, 2002.

[SEI02] Carnegie Mellon University, Software Engineering Institute. Software Acquisition Capability Maturity Model Version 1.03. available on http://www.sei.cmu.edu/pub/documents/95.reports/pdf/mm003.95.pdf. Accessed May 2008.

[SEI02] Carnegie Mellon University, Software Engineering Institute. Software Aquisition Capability Maturity Model Version 1.03. available on http://www.sei.cmu.edu/pub/documents/95.reports/pdf/mm003.95.pdf. Accessed May 2008.

[SEI06] Carnegie Mellon University, Software Engineering Institute CMMI for Development available on http://www.sei.cmu.edu/publications/documents/06.reports/06tr008.html. Accessed May 2008.

[SEI07] Carnegie Mellon University, Software Engineering Institute. CMMI for Acquisition. available on http://www.sei.cmu.edu/publications/documents/07.reports/07tr017.html. Accessed May 2008.

[SEI95] Carnegie Mellon University, Software Engineering Institute. A Systems Engineering Capability Maturity Model Version^{SM} 1.1. available on http://www.sei.cmu.edu/pub/documents/95.reports/pdf/mm003.95.pdf. Accessed May 2008.

[So06] Sommerville, I. “Software Engineering: Update”. Addison-Wesley Longman, Amsterdam; Auflage: 8th ed. (15. Juni 2006)