Competency Assessment: Integrating COCOMO II and People-CMM for estimation improvement.

Ricardo Colomo Palacios, Marcos Ruano Mayoral, Juan Miguel Gómez Berbis, Ángel García Crespo

Dpto. de Informática, Universidad de Carlos III de Madrid

Leganés, 28911 Madrid, España

E-mail:

{ricardo.colomo, marcos.ruano, juanmiguel.gomez, angel.garcia}@uc3m.es

Abstract:

'Human factor' is one of the most relevant and crucial aspects of software development projects management. Aiming at the performance improvement for software processes in organizations, a new model has been developed to diagnose people related processes. This new model is People-CMM and represents a complementary solution to CMM. On the other hand, existing estimation models in Software Engineering perfectly integrate those aspects related to personnel's technical and general competence, but fail to integrate competence and performance measurement instruments when it comes to determine the precise value for each of the factors involved in the estimation process. After reviewing the already deployed initiatives and recommendations for competence measurement in the industrial environment and the most relevant estimation methods for personnel factors used in software development projects, this article presents a recommendation for the integration of each of the 'human factor' related metrics in COCOMO II with the management tools proposed by People-CMM, which are widely implemented by existing commercial tools.

Keywords: COCOMO II, Software Estimation, Competence Assessment.

1

1.   Introduction.

Presently, software is a critical factor. Failure rate in software projects are high and qualified software engineers pertaining to software development teams are key factors in the software development process and their shortcomings and caveats (Pressman, 2005). These teams are composed by professionals with a heterogeneous training, background and expertise (McConnell, 2003) that Human Resources management must be able to evaluate and provide with a professional view, with the ultimate goal of improving the competences of the workforce and their results (Curtis, Hefley & Miller, 2001).

Individual differences have been identified as one of the paradigms for the research of human factors in software development (Curtis, 2002). Research in this area goes back to the 1960s (Sackman, Erikson & Grant, 1968) and continues actively in the 1980s (Curtis, 1981), (De Marco & Lister, 1985). During that decade, tools and models for software estimation, such as COCOMO (Boehm, 1981), started spreading. Some of the previously quoted models allow to identify productivity differences among human resources calculating time and effort with some parameters provided by the user. Actually, this is very close to the work presented in this paper. The main objective of this paper is to identify a correspondence between the estimation and the evaluation of human resources from the point of view of competence management. This correspondence would allow the integration of both instruments into a unified framework, providing managers highly reliable estimation mechanisms.

The remainder of this paper is organized as follows. Section 2 presents related work and a conceptual model for competency, competency management and competency assessment. Section 3 describes personnel factors in software Project estimation and more concretely, the COCOMO II approach. In Section 4 an integration proposal between competency assessment and COCOMO II personnel factors is presented. Finally, section 5 concludes the paper.

2.   The Competency Paradigm, Competency Management & Assessment.

The competency approach to human resources management can be found in early origins. The early Romans practiced a form of competency profiling in attempts to detail the attributes of a "good Roman soldier" (Draganidis & Mentzas, 2006, p.52). In recent history, competence is a concept used by early 20th century scientific management (Taylor, 1911) and later revised and redefined by McClelland, former Hay Group director, in the early seventies. According to this author competency is comprehended as the relation between humans and work tasks, that is, the concern is not about knowledge and skills in itself, but what knowledge and skills are required to perform a specific job or task in an efficient way (McClelland, 1973). From the sixties, and due to the ever-growing adoption in business environments of the competency paradigm, a number of definitions emerged for the term, to the extent that multiple authors adhere to the opinion of facing a "competency pandemonium" (DeHaro, 2004). In a recent analysis of the term in the scientific literature, Draganidis & Mentzas (2006) state that a competency must be defined in terms of:

- **Category**. A group to which homogeneous and/or similar competencies belong.
- **Competency**. A descriptive name for the specific competency.
- **Definition**. Statement(s) that explains the basic concept of this competency.
- **Demonstrated behavior**. Behavior indicators which an individual should demonstrate if the specified competency is possessed.

The aforementioned authors point out, based on the survey of a number of competency-based management systems, that competencies are capital in the following employee management applications: Workforce planning,

Recruitment management, Learning management, Performance management, Career development and Succession planning. According to People CMM (Curtis, Hefley & Millar, 2001), competency management is a collection of workforce practices used to enhance the capability of the workforce to perform their assigned tasks and responsibilities, and to achieve specific competency growth objectives. Competency management is becoming more and more important in today's organizations. Workers competency levels are important for the achievement of company goals, complimentary to, for instance, core business processes, customer relationships and financial issues (Norton & Kaplan, 1996).

Berio & Harzallah (2005) provide a very simplified view, but very intuitive of the Competency Management process which is depicted in the following figure:
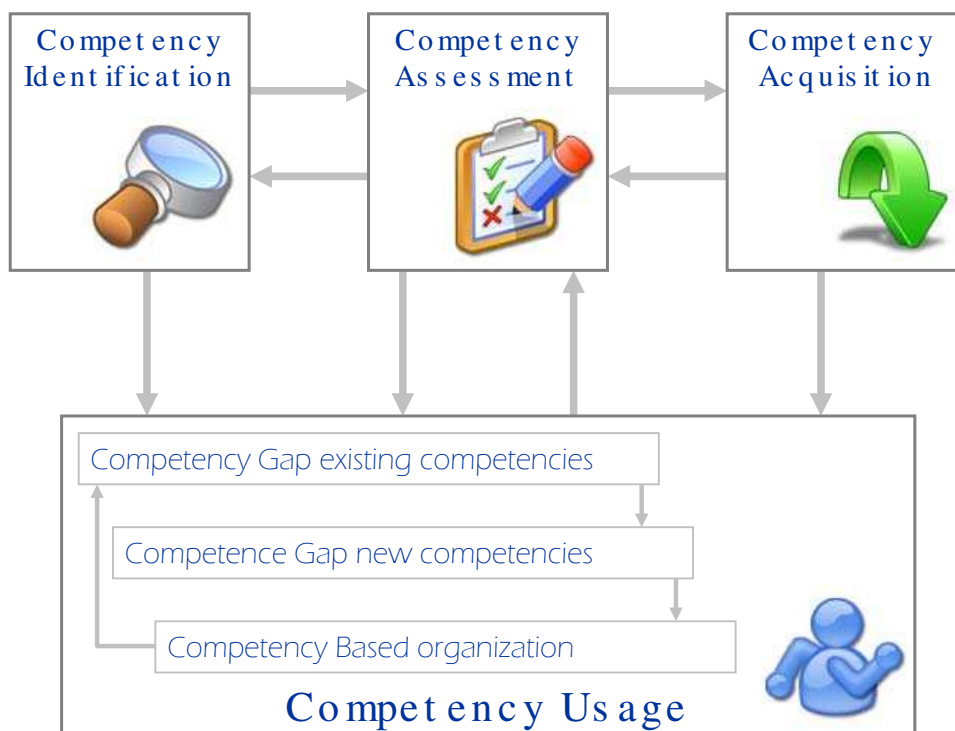

Figure 1: Competence Management Process Architecture

4

- **Competence identification**, i.e. when and how to identify and to define competencies required (in the present or in the future) to carry out tasks, missions, strategies; how competence is represented is included here.

- **Competence assessment**, i.e. (i) when and how to identify and to define competence acquired by individuals and/or (ii) when and how an enterprise can decide that an employee (or an individual) has acquired specific competencies; how the relationships between individuals and required competencies are represented is included here.

- **Competence acquisition**, i.e. how an enterprise can decide about how to acquire some competencies in a planned way and when;

- **Competence usage**, i.e. how to use the information or knowledge about the competencies produced and transformed by identification, assessment and acquisition processes; for instance, how to identify gaps between required and acquired competencies, who should attend required training, how finding key employees (i.e. holding key competencies) and so on.

In the scientific literature, a number of initiatives have been for implementation have been considered in Knowledge Management systems and, particularly, for Competency Management in Software Engineering (Rus & Lindvall, 2002), (Lindgren, Stenmark, & Ljungberg, 2003), (Aurum, Parkin, & Cox, 2004) & (Dingsøyr, Djarraya & Røyrvik, 2005).

Performance evaluation can be defined as the systematic and periodical process of measuring, in an objective way, the efficiency of an employee or a team in their job (Pereda & Berrocal, 2001). Typically, employees with certain seniority are evaluated yearly by their superior about the set of competences involved in their roles. The evaluation is made under a stable and defined framework applicable to the whole set of human resources of the company. The results of the evaluation can influence the salary and the training plan of the evaluated employee among others.

Competence-based performance evaluation systems are integrated into global competence management systems that implement competence paradigm to the management of human resources. Competence-based performance evaluation systems assess the possession of generic and technical competences. Generic competences, such as leadership and effective teamwork, are not related with the development of any specific activity. Technical competences are those destined to the development of a specific activity and they are really close to that knowledge put into practice and capabilities. On the market there are several commercial solutions that support both types of competences but, nevertheless, the in-depth management of technical competences is usually made by means of skills management systems that are closer to knowledge management.

3. Software project estimation: COCOMO II & personnel factors.

According to (Boehm, Abts & Chulani, 2000), Software Estimation Techniques can be divided into six categories:

- **Model Based** (SLIM (Putnam & Myers 1992), COCOMO (Boehm, 1981), SEER (Jensen, 1983), ESTIMACS (Rubin, 1983), PRICE-S (Park, 1988) y Checkpoint (Jones, 1997)).
- **Expertise Based** (COCOTS (Constructive COTS) (Abts, 1997)).
- **Learning Oriented** (Case-Based (Shepperd & Schofield, 1997), Artificial Neural Networks).
- **System Dynamics Based** (Abdel-Hamid & Madnick (1991)).
- **Regression Based** (OLS, Robust).
- **Composite- Bayesian** (COCOMO II 2000 (Boehm, Horowitz, Madachy, Reifer, Clark, Steece, Winsor Brown, Chulani, & Abts, 2000)).

Due to its relevance and commercial success, our work will analyze and upgrade COCOMO II 2000 (Boehm et al., 2000). The rest of the estimation techniques, described in (Saliu, Ahmed & AlGhamdi, 2004), will not be tackled in this article and its in-depth analysis is proposed as future work.

COCOMO (COnstructive Cost MOdel) is the most developed algorithmic model since it was first created. Its first version was published in l981 (Boehm, 1981) and updated in 1987 with an extension to ADA language. In 1994 started the development of COCOMO II that was first published in 1995. According to the previous classification of estimation techniques, COCOMO II and COCOMO II 2000 (Boehm et al., 2000) belong to the composite techniques group, due to the use of Bayesian techniques for the calibration of the model.

The effect of personnel factors in algorithmic models is extremely relevant and many models include this kind of factors in their algorithms (SLIM, Checkpoint, PRICE-S, ESTIMACS, SEER-SEM, COCOMO II, …). Nonetheless, the latter is the model that is considered to provide a deeper support.

In COCOMO II, the factors taken into account when capturing the impact of the project's environment in its cost are classified into four big groups: product, platform, project and personnel. Personnel related factors are, just after the size of the product, the most influential factor when determining the required effort for the development of a software product. In particular, six personnel related factors are identified:

- ACAP: Analyst capability.
- PCAP: Programmer capability.
- PCON: Personnel continuity.
- APEX: Applications experience.
- PLEX: Platform experience.
- LTEX: Language and tool experience.

7

AEXP, PEXP and LTEX factors, related to experience, are measured in periods of time (months and years). Personnel continuity is measured as a yearly percentage of workers. For ACAP and PCAP a percentile scale over the capacity is used to determine the value of each factor according to the perception of the manager. Table 1 shows the effect on the effort multipliers for personnel factors according to COCOMO II.

	Very Low	Low	Average	High	Very High	Extra High
ACAP	*15 %*	*35%*	*55%*	*75%*	*90%*	
	1.42	1.19	1.00	0.85	0.71	n/a
PCAP	*15%*	*35%*	*55%*	*75%*	*90%*	
	1.34	1.15	1.00	0.88	0.76	n/a
PCON	*48%/year*	*24% / year*	*12% / year*	*6% / year*	*3% / year*	
	1.29	1.12	1.00	0.90	0.81	
APEX	*<= 2 months*	*6 months*	*1 year*	*3 years*	*6 years*	
	1.22	1.10	1.00	0.88	0.81	n/a
PLEX	*<= 2 months*	*6 months*	*1 year*	*3 years*	*6 years*	
	1.19	1.09	1.00	0.91	0.85	n/a
LTEX	*<= 2 months*	*6 months*	*1 year*	*3 years*	*6 years*	
	1.20	1.09	1.00	0.91	0.84	

Table 1. COCOMO II Personnel Factors effort multipliers.

To make an estimation based on the principles of Engineering, data should be sufficiently reliable for every factor. Therefore, even when there are available numeric data for factors related to experience and continuity, ¿which are the criteria for the evaluation for the evaluation of the capacity of programmers and analysts?, ¿what should be the reference for the different experience measures?. Next chapter suggests the utilization of artifacts and methods typical of evaluation and management of competences as a support for the calculation of this kind of factors in software estimation.

4.   Competency assessment & COCOMO II: An integration proposal.

People-CMM (Curtis, Hefley & Miller, 2001) was developed by the Software Engineering Institute at the Carnegie-Mellon University. Encouraged by the mentioned institution, a number of Software Engineers and Human Resources

researchers developed a set of models to help organizations in the management and intellectual capital development. This initiative is based on the most successful strategies for Human Resources, knowledge management and organizational development, is deemed as a prominent set of guidelines to achieve a steady and ever-increasing improvement of the organization working force. The ultimate goal is to also improve the capability of the organizations to bring, train, motivate, organize, manage and hold their human resources.

As mentioned before, despite the relevance of COCOMO II in software estimation and the importance of competence management in knowledge-intensive environments, no correspondence has been established between both support tools for software development projects managers. The first step to build the desired correspondence is the analysis of those People-CMM practices that are necessary to determine the values of each of the personnel factors identified in COCOMO II. Secondly, it is proposed a recommendation to determine the value of each factor from the results of the different techniques provided by the maturity model developed by SEI.

4.1.   ACAP.

Analysts, according to the definition provided by the estimation method, are those practitioners related to requirements, high-level design and detailed design. The attributes that should be considered for the valuation of this factor include synthesis and analysis capacity, efficiency, thoroughness, and communication and cooperation capabilities (Boehm et al., 2000). The experience of the analysts should not be taken into consideration because it is valued by means of APEX, LTEX and PLEX.

Taking into account the provided definition, when determining the value of ACAP, technical competences related to requirements management and design and general competences related to teamwork and synthesis and analysis capacity should be analyzed.

9

For the integration of the human resources management practices included in People-CMM and the estimation methods for ACAP, it is regarded as necessary that the organization implements the process areas detailed next:

- Level 2.

  o Performance management, which includes the assessment of the performance of professionals.

- Level 3.

  o Competence analysis, which establishes the procedures and methods for the identification of the competences required for the attainment of the objectives of the organization.

  o Development of professional careers, which determines the different professional paths and should include those paths for analysts and programmers.

If the organization had a competence-based performance assessment system that included the aforementioned process areas, the value of ACAP could be determined from the combination of the values related to the following list of competences extracted from the Software Engineering Body of Knowledge, SWEBOK (Bourque & Dupuis, 2004):

- Generic competences:

  o Capacity for analysis and synthesis.

  o Teamwork.

  o Concern for quality.

  o Oral and written communication.

- Technical competences:

  o Software requirements.

  o Software design.

The provided competence list is a generic proposal. Following the dictates of People-CMM, the determination of this value should be carried on by means of

a detailed study of the necessary competences for analysts so that they can develop their duties with excellence in a particular organization.

The value of ACAP is the combination of the performance assessment scores of the analysts with the identified competences involved in the project (EVAL), weighed by the degree of participation in the project (DEDI) and the relative importance of each competence (POND). The following formula shows the summation of the different factors involved in the calculation of ACAP:

$$ACAP = \sum_{j=1}^{k} \sum_{i=1}^{n} Pond(j) \, x \, Ded(i) \, x \, Eval(i, j)$$

The numeric value for ACAP obtained after the calculation provides a value within the range of the scores used for the evaluation of competences, which are usually determined by a Likert scale. This value can be automatically converted to a percentiles scale, the unit used to express ACAP.

4.2. PCAP.

The capacity of programmers should also be taken into account for estimation. For this factor, just like ACAP, experience should not be considered but efficiency, thoroughness, and the capacity for communication and teamwork of the programmers should be measured (Boehm et al., 2000).

PCAP factor requires the same process areas of People-CMM as ACAP. However, according to the description of the evaluation that should be performed for the determination of PCAP, the set of involved technical competences is reduced to Software Construction. The value of PCAP is obtained from the following formula:

$$PCAP = \sum_{j=1}^{k} \sum_{i=1}^{n} Pond(j) \, x \, Ded(i) \, x \, Eval(i, j)$$

4.3.   PCON.

The personnel continuity factor (PCON) measures the annual personnel turnover as regards the total number of human resources (Boehm et al., 2000). It is the easiest factor to calculate, since it is based in a mere historical record of the new staff taking on. Even those organizations in the initial maturity levels should be able to calculate PCON, at least if the staff of the project is formed by internal resources. This is even simpler in Spain where it is compulsory to record the seniority of each worker in their payslip.

However, the effects of subcontracts and the wealth of self-employed practitioners can considerably complicate the calculation of PCON. In this case the relationship between the employees and the organization has not labor character and the traceability is more complex. People-CMM does not include any explicit mention or recommendation for this kind of records, possibly due to its simplicity.

For this reason, organizations are recommended to implement any system to support the assignation of resources and projects to allow their traceability.

4.4.   APEX.

APEX factor reflects the experience, measured in units of time, of the staff of the project in the type of application being developed (Boehm et al., 2000).

Experience is approached within People-CMM across several knowledge areas. The first of the recommendations for the management of experience is provided at level 2, in practice no. 3 at the "Staffing" knowledge area. However, its proper management requires the implementation of techniques closer to Knowledge Management. This circumstance fosters the search of competency within the organization and the determination of the experience of its workers.

12

Having this in mind, the knowledge area that should be considered is "Assets based on competences", placed at level 4. The second practice in this knowledge area recommends the implementation of vehicles for sharing competences. It can be stated that the observance of this second practice is a necessary condition for the successful implementation of this kind of systems. Lindgren and Stenmark (2002) provide a technical guide for the implementation of competence search within organizations.

The adoption of these practices in the environment of software development has been abundantly documented in technical literature (Dingsøyr & Røyrvik, 2003), (Lindgren & Stebnmark, 2002), (Hardless, 2005) and has the support of several commercial tools such as Skillscape and Skillview among others.

In particular, APEX factor can be directly determined from the documented experience stored in the implemented system. The system provides, for each of the project staff members, an expertise report in the functional environment that can be specified as any other kind of knowledge. From this report, that indicates the total labour experience in days (EXPE), and weighing the different dedication of the professionals to the project (DEDI) it can be directly calculated the value for APEX as shown in the following formula:

$$APEX = \sum_{j=1}^{k} \sum_{i=1}^{n} Dedi(i) x Expe(i,j)$$

4.5. PLEX.

PLEX represents the experience in the technologic platform and reflects the relevance of the understanding and the use of the platform which includes GUI, databases, middleware, etc (Boehm et al., 2000).

Taking into account the aforementioned considerations about APEX, PLEX can be determined in a similar way with some modifications to the intrinsic

characteristics of the factor. PLEX, as pointed out in its definition, involves a set of technical factors that should be reflected in its estimation: knowledge on middleware, database management systems …

The proposal for PLEX is to specify each of the elements that make the technological platform up and weigh their relevance. The final estimation will be the combination of the values for the experience of each professional in the different technologies (EXPE), the relative weigh of each one (POND) and the dedication of each professional (DEDI), as expressed in the following formula:

$$PLEX = \sum_{j=1}^{k} \sum_{i=1}^{n} Pond(j) x Dedi(i) x Expe(i, j)$$

4.6. LTEX.

The last of the personnel factors provided by COCOMO II is the experience of the project team in the language and the tools to be used during its development (Boehm et al., 2000).

The value of LTEX can be determined following the same criteria used for PLEX. The estimation should include the experience of the professionals in the language and the tools, their relative weigh and the dedication of the professionals to the tasks of the project.

$$LTEX = \sum_{j=1}^{k} \sum_{i=1}^{n} Pond(j) x Dedi(i) x Expe(i, j)$$

5. Solution Testing.

To perform an empirical validation of the solution proposed in this paper, it has been implemented in two different environments. The first of them, Company A, represents a software development company focused on out-of-the-box solutions based in Java technologies. This company needs to add an isolated

functionality to one of the packages for the support of commercial Leads. The second one, Company B, is a company specialized in tailored developments that has received an order to develop a Java-based, integrated traceability control solution for a kettle farm. Both companies have implemented commercial Competence Management Systems that are able to provide all the required data for the implementation of the improvement proposal.

Managers in Company A estimate that the required development will approximately produce 8,000 SLOC, involving lots of references to objects already used in the current version. After providing all the parameters to the COCOMO II model used by the company, the effort estimates (in the interests of simplicity) indicate that the development will require a total effort of 16.01 man-month. On the other hand, after introducing the proposed competential measures, the obtained effort is slightly higher and represents 17.28 man-month.

The tailored development by Company B is estimated to consist of 15,000 SLOC, which using the same model as in the previous case throws an effort estimation of 57.64 man-month. The effort estimation refined with the improvement proposal results in 55.12 man-month.

The actual human resources requirements as well as the different estimates are shown in Table 2.

	COCOMO II	Model	Reality
Company A	16.01	17.28	21.50
Company B	57.64	51.93	55.50

Table 2. COCOMO II effort estimation, new model effort estimation and resources consumpted.

The low complexity of the project developed by Company A has leaded to an estimation of the resources that is closer to real effort after applying the

improved model. On the other hand, for the project by Company B, the estimation obtained from the COCOMO II model is closer to reality that the estimation of the modified model. A possible reason for this dissimilarity is that the development of tailored solutions has got a higher level of uncertainty due to the lack of knowledge about the functional environment. Nonetheless, it is important to make clear that the estimation obtained from the modified model is very close to reality and that the decisive estimation factors can belong to other categories such as Scale Factors, pressure put by the release to an external client o some intrinsic problems related to competence evaluation.

Another cause for the variability of the estimation results might be the calibration of the model. Several authors have addressed the improvement of algorithmic estimation methods by the calibration of the model's coefficients to the local environment (Shepperd, 2007). Authors such as Gulezian (1991) and Jeffery & Low (1990) concluded that calibration is essential in models that were to be used outside those environments in which they were developed. This conclusion seems to be paradoxical since COCOMO was intended to be general purpose through its many drivers and parameters (Shepperd, 2007).

In any case, basing estimations on data collection and analysis in accordance to engineering criteria means one step forward for the consideration of Software Engineering as a remarkable discipline.

6. Conclusions and future work.

This article presents a correspondence between the most outstanding and widespread models for human resources management. In particular, those recommended by People-CMM and COCOMO II, that provide software estimation methods and include personnel related aspects.

This research interest on the improvement of software estimations is based on the specification of conversion criteria between the results of the

implementation of the practices proposed by People-CMM and the six personnel related factors identified by COCOMO II.

The integration of the different systems that a software project manager should use is a crucial factor towards their complete leveraging. This work has only focused in a small subset of the factors, but the integration possibilities are practically infinite. In current business environment, where information technologies control the management of corporate resources, transversal leveraging of information is an opportunity that managers and IT workers should not waste if they are committed to the improvement of business performance and the development of human capital.

The future works in the area of estimation should be directed to the calculation of the relative values of other factors included in COCOMO II using engineering criteria and the extension to other estimation techniques for software development projects both algorithmic and no algorithmic. It would also be really interesting to study the influence of the calibration of the integrated model in the estimation results as well as the determination of the set of practices or guidelines for the correct calibration if necessary.

References.

Abdel-Hamid, T. & Madnick, S., 1991, Software Project Dynamics, Prentice-Hall.

Abts, C., (1997), COTS Software Integration Modeling Study, Report prepared for USAF Electronics System Center, Contract No. F30602-94-C-1095, University of Southern California, 1997.

Aurum, A., Parkin, P. & Cox, K. (2004). Knowledge Management in Software Engineering Education. Proceedings of IEEE International Conference on Advanced Learning Technologies, pp. 370 – 374.

17

Berio, G. & Harzallah, M. (2005). Knowledge Management for Competence Management. Journal of Universal Knowledge Management, 0(1), pp. 21-28.

Boehm, B. (1981) Software Engineering Economics, Prentice Hall.

Boehm, B., Abts, C. & Chulani, S. (2000) Software Development Cost Estimation Approaches - A Survey, Annals of Software Engineering, 10(1-4), pp. 177-205.

Boehm, B.W., Horowitz, E., Madachy, R., Reifer, D., Clark, B.K., Steece, B., Winsor Brown, A., Chulani, S. & Abts, C. (2000) Software cost estimation using COCOMO II, Prentice-Hall.

Bourque, P. & Dupuis, R. editors. (2004) Guide to the Software Engineering Body of Knowledge, IEEE Computer Society Press, Los Alamitos, CA.

Curtis, B. (1981) Substantiating Programmer Variability, Proceedings of the IEEE, 69 (7).

Curtis, B. (2002), Human Factors in Software Development en Marciniak, J.J., Encyclopedia of Software Engineering, pp. 598-610, Willey & Sons.

Curtis, B. Hefley, W.E. & Miller, S.A. (2001). People Capability Maturity Model. Version 2.0, CMU/SEI-2001-MM-01

De Haro, J.M. (2004). ¿Sabe alguien qué es una competencia? De McClelland a la ISO 9000. (Does anybody know what is competeny? From McClelland to ISO 9900) Revista de la Asociación Española de Dirección de Personal (Journal of Personal Management Spanish Association), 30, pp. 8-17

De Marco, T. & Lister, T. (1985) Programmer Performance and the effects of the workplace, Proceedings of the 8th International Conference on Software Engineering.

Dingsøyr, T. & Røyrvik, E. (2003) An empirical study of an informal knowledge repository in a medium-sized software consulting company, Proceedings of the 25th International Conference on Software Engineering, pp: 84 – 92,Portland.

Dingsøyr, T., Djarraya, H.K. & Røyrvik, E. (2005). Practical knowledge management tool use in a software consulting company. Communications of the ACM, 48(2). pp. 96-100.

Draganidis, F. & Mentzas, G. (2006). Competency based management: a review of systems and approaches. Journal: Information Management & Computer Security, 14(1), pp.51 - 64

Gulezian, R. (1991). Reformulating and calibrating COCOMO. Journal of Systems and Software, 16 (3), pp. 235-242.

Hardless, C. (2005). Designing competence development systems, Doctoral dissertation Goteborg University.

Jeffery, R. & Low, G. (1990). Calibrating estimation tools for software development. Software Engineering Journal, 5 (4), pp. 215-221.

Jensen R. (1983). An Improved Macrolevel Software Development Resource Estimation Model, Proceedings 5th ISPA Conference, pp. 88-92.

Jones, C. (1997). Applied Software Measurement, McGraw Hill.

Lindgren, R. & Stenmark, D. (2002) Designing competence systems: Towards interest activated technology, Scandinavian journal of Information Systems, 12(1), pp. 18-29.

Lindgren, R., Stenmark, D. & Ljungberg, J. (2003). Rethinking Competence Systems for Knowledge-Based Organizations. European Journal of Information Systems, 12(1), pp. 18-29.

McClelland, D.C., (1973) Testing for competence rather than for 'intelligence', American Psychologist, vol. 28, pp 1–14.

McConnell, S. (2003) Professional Software Development, Addison-Wesley.

Norton, D.P. & Kaplan, R.S. (1996). The balanced scorecard: translating strategy into action. Harvard Business School Press.

Park R., (1988) The Central Equations of the PRICE Software Cost Model, Proceedings of the fourth annual COCOMO Users' Group Meeting.

Pereda, S. & Berrocal, F. (2001). Técnicas de gestión de recursos humanos por competencias, Centro de estudios Ramón Areces, Madrid.

Pressman, R. (2005). Software Engineering: A Practitioner's Approach, McGraw Hill.

Putnam, L. & Myers, W. (1992), Measures for Excellence, Yourdon Press Computing Series.

Rubin, H.A. (1983) Macroestimation of software development parameters: The Estimacs system. In SOFTFAIR Conference on Software Deuelop- ment Took, Techniques and AIternatiues (Arlington, Va., July 25-28). IEEE Press, New York, 1983. pp. 109-118.

Rus, I. & Lindvall, M. (2002). Knowledge management in software engineering. IEEE Software, 19(3), pp. 26-38

Sackman H., Erikson, W.J. & Grant, E.E. (1968). Exploratory Studies comparing online and offline programming performance, Communications of the ACM, 11(1), pp. 3-11.

Saliu, M. O., Ahmed, M. A. & AlGhamdi, J. S., 2004, Towards Adaptive Soft Computing Based Software Effort Prediction, Proceedings of the North American Fuzzy Information Processing Society Conference, pp. 16-21.

Shepperd, M. & Schofield, M. (1997). Estimating Software Project Effort Using Analogies, IEEE Transactions on Software Engineering, 23(12), pp. 736-743.

Shepperd, M. (2007). Software project economics: a roadmap. FOSE '07: 2007 Future of Software Engineering. IEEE International Conference on Software Engineering. pp. 304-315.

Taylor, F., 1911, The Principles of Scientific Management. Harper & Row,New York.