Software process management supported for simulator: an application to transition workflow of the RUP

G.A. Dawilmar[1,2] and S. Nilson[1]
[1]Laboratory for Computing and Applied Mathematics - LAC
Brazilian National Institute for Space Research - INPE
C. Postal 515 – 12245-970 – São José dos Campos - SP
BRAZIL

[2]University of Taubaté – UNITAU
R Daniel Danelli, S/N – J. Morumbi – 12031-000 – Taubaté – SP
BRAZIL

E-mail: dawilmar@lac.inpe.br, nilson@lac.inpe.br

The software process defined as product is a results of a team effort very important for organizations life. With strong link with the develop environment, the software process is a key activity for quality assurance of organizations products. Than, management decision that involve process management (definition, performance and controls) are considered difficult and important actions to the maturity of developing organizations which adopted process *framework* as CMMI or ISO. In order, is "good-definition" of software process and an adequate follow-up during theirs performance has several implications in Software Development Environment-SDE. Development teams have the compromise to use very often methods, technique and tools that help to setup the management activities of the software process and the support of the development theirs products. This article presents an application of the management model from Araújo (Araújo, 2005) where it is possible to see the integration of the simulation and management in an unique system, as an example, bigger and better control of the performance and optimization of process (such as resources, activities sequences, job the scripts, teams definition for development etc). The methodology includes the paper revision about management and simulation of software processes, and a discuss of the implementation of the processes of RUP – Rational Unified Process (Transition phase). The viability of the integration of the model proposed is discussed looking forward the environment requirements of the management activities, estimating the different aspects of software project such as cost, effort, timing etc). This article shows some results in the application of the chosen model in the environment proposed, comparing with those presented by Boehm (Boehm, 2000). The comparison shows the viability of the simultaneous use of the simulation with the management of software processes with support of modeling, performance and control of process. Software processes developments are very dynamic and are influenced by the pressure of the competitiveness for delivery products of software with quality in time and reasonable cost. To implement alterations in the software processes they finish being difficult, expensive and they can introduce errors. In this direction this work sample that the simulation comes to provide with viable form and cheap application to evaluate and to reduce this risk through you analyze quantitative of modification of processes considered in terms of performance of the process in diverse experimental scenes. The article shows also that the simulation gives support, cheap and viable for the application to evaluate and to reduce the risk thorough the quantitative analysis of the change of the proposed process looking forward performance in different situations.

Keywords: software engineering, software process management, modeling and simulation of software process, software process technology, RUP.

1. INTRODUCTION

The organizations that develop software are interested for improvement of its processes. The results of the efforts found in literature can be mainly seen by the discernment of models of maturity and improvements of processes[1]. For example of models they can be cited CMMI-Capability Maturity Model Integration of SEI-Software Engineer Institute[2], or standard ISO-International Organization.

The adequate EDS-Environment of Development of Software must offer conditions to define, manage, measure and control the software processes (SANT'ANNA, 2000), and as example of an environment model that propitiates these

[1] www.pbqps.mct.gov.br
[2] http://sei.cmu.edu , 2005

easiness can cite the Environment of Development of Engineering of Software Centered in the Process, of the PSEE-Process-Centered original Software Engineering Environment.

Gift in the models of inserted maturity and in these environments SDE, one of the practical key of success of the managements of the Engineering of Software very evidenced is the estimate of parameters as cost, stated period, effort etc. Controlling must carefully deal with this decision. This influences directly, for example in the definition of the process of adopted development and in the product quality, which have developed the software.

One of the points key of success of the management of the Engineering of Software it is definition of the team. Controlling must also deal with the decision of the choice of the team. This influences in the product quality of software to be developed, in the process of adopted development, the routine operational results of the management, and in the dimensions of the parameters as costs, stated period, effort, productivity and quality.

To define a team is one of the stages of the management of software projects that must be carried with priority, and is a sufficiently critical task. The good definition includes the choice of the certain people for the amount and attributes of quality (knowledge, experience, ability etc), for the certain tasks, at the certain moments (defined in a good model of development process).

For this managing activity, the sufficiently specific knowledge that the manager must have of the participant elements and of the activities she must enough be adjusted and for the success of its work. Criteria as time and ability in projects, experiences, pointers of answers to the quality and productivity etc are part of the quantity of items that they go to participate of the decision taking. In a general way, to summarize, it can be said that the great challenges of the Engineering of Software are; among others, in the following points:

a) There is a concern and interest for the improvement of the software processes, a time that these are related with the product quality of software;

b) There is a great difficulty of establishing metric, estimation and conferring them for diverse aspects as cost, stated period, risk, effort and quality in the SDE;

c) The SDE are dynamic, what it takes the involved ones to establish diverse models of processes demanding different techniques, tools for the success of its implementation; e

d) The SDE depend strong on human resources. The good choice of the team and its integration to the process anticipates risks, uncertainty and increase the productivity with quality and success of the managements.

1.1 Objectives

From this context, one searched to think about a technology that was able to support the related activities the particular management and in the accomplishment of estimates, as a contribution for the definition and standardization of software processes, increase of the trustworthiness of the results of the estimates by means of gauging of pertinent parameters etc. To get a support model the management for gauging of estimates of the cited parameters already and the aspects related to the activities (decomposition in tasks as well as its characterization) by means of the use of the technique of agreed simulation..

2. SOFTWARE PROCESSES

Process of Software is "joint of organizational politics, structures, technologies, procedures and devices to conceive, to develop, to construct and to keep a software product". (CANHADAS, 2005).

The qualification in software development estimates the existence of a process of definite software and the application of a model of improvement of processes. In this direction, the qualification in software development reflects the degree of institutionalization of an infrastructure and the culture related to the methods, practical and procedures of development of software of an organization, or either, of the processes of software and its maturity.

Practical the described ones for the maturity models, for example CMM, also identified as practical-key, as much lead to the improvement of the processes as they serve of reference for the evaluation of the maturity of the same ones, what it allows to project improvements. Already on the other hand, a definite process has documentation that it details what is fact, when, for who, what is used and what is produced. The maturity of the process indicates until point a process is shaped, as well as its too much on aspects its management, metric, control and effectiveness is carried through.

2.1 Simulation in software processes

Simulation of software processes can be seen of general form in three classic models: event-discrete, continuous and hybrid (RAFFO, 2004). For the first model, event-discrete simulation, understands that the interest events happen promptly in the time and it does not matter what happens between them. Thus for example, to initiate a codification, to finish the code, to initiate an interpretation or analysis of requirements and many others can easily be characterized in the time. Already as the model, of the continuous simulation, searches to capture dynamic aspects that the previous model does not make it. The activities of the software processes are developed by human agents and consume a certain time that for happiness causes modification in its answers how much to the productivity, product quality that generates etc. Factors as stress can diminish these answers, or motivation can increase. Models of continuous simulation aim at the capture and agreement of these phenomena. The calls hybrid models appear when they exist changeable (continuous and discrete) of the two types of involved models in the study and that they are considered jointly. Also when from any modification that if makes in one or another model and that they modify the conceptual structure of the original model. Independently of the choice of the simulation model, it must the desenvolvedores possess the two main abilities:

a) To write models and to rewrite in simulation environments (to know the process of simulation, the used tool and everything that the fence); and

b) To know processes of software development (inquiry object, with its asset, rules and all more than the fence).

3. ESTIMATES AND GAUGING OF THE ESTIMATES IN SOFTWARE PROCESS

According the work of estimates could be carried through for example, for historical data, through analysis statistics, for estimates of specialists, methods and techniques guided for parameters of the projects through the acceptance of similar projects etc. Estimates can be seen in macro and micron estimates (IOANA, 1999). Fitting first a forecast of all. For this estimate evaluation for specialists is known (subjective, citizens the modifications), for mathematical model (using given historical), for comparison and analogy (projects with attributes to simulate and the main analogous attributes). This first form is characterized by being simpler in terms of registers; more commonly is found and is gifts in the first phases of development of maturity of the organizations. Already in the vision micron of the estimate an effort associated with each component or activity of the process adds separately. In this in case that, a decomposition in lesser parts of the project becomes. This form is characterized for being more complex, demands more maturity, disciplines of manages, experiences, standards of collection, registers of data of systematic and continuous form. They are gifts in the maturity levels highest.

Picture 1. Types of esteem models (Ioana, 1999).

Níveis de abstração	
High	Linear equations (simple) Determinísticos models. COCOMO Example. Defects = αKSLOC +βEffort
	Relationship Multi-variable (complex) Ex. Models of regression and statisticians Ex. Monte Carlo.
	Models of processes (static) Ex. Historical data of the activities of each cycle of life. Ex. Comparison between the processes and 0 variable.
	Models of processes (dynamic) Dynamic modeling of systems. Ex. Representation continues of interaction of 0 variable.
Low	Models of processes with simulation Ex. Evento-Discreto, continuum and hybrid. Ex. Event-discrete and combined. Ex. Comparison product x captured process.

To understand these relations between some models of existing estimates, a classification is presented in summary by Ioana (IOANA, 1999) and is presented in Picture 1. In this classification the degree of validation of the model is observed in accordance with that these models can be segmented and be based on abstraction levels. Its groupings are presented in a preliminary version, in one another more complete work carried through by the author of this article, find if in development a more systematic study. Of Picture 1, it is had that the high levels of abstraction (high) produce more similarity between the models. Simple equations exist that tell parameters of exits for some independent 0 variable (esteem). This boarding searches to see software process as a black box. In a detailed level more (low) it is observed use of models of processes based on simulation, where if it locates this proposal, to use the simulation for estimates.

3.1. Attainment and analysis of changeable candidates

On the basis of given and trustworthy information the estimates can be established. To get data is expensive, spends time and demands care. A relative planning to the collection, analysis and everything that the use of data involves must be made to guarantee the validity of the data (ARAÚJO, 1999). To get data efficiently, some principles must be followed (not necessarily in this order of importance):

a) To have objectives and planning before the taking of data;
b) To choose the data to be gotten established in the model of the process that is being analyzed;
c) Data of the proper process must carefully be defined and be managed, and
d) To have a plan of collection of data that must subsidize the management.

The objective of the retaining data and analysis is to use as well as possible, the data of form objective, absolute, and explicit. To be more significant a measure must be robust, understandable, explicit the properties of the process, consider strategically improvements, to be natural how much to the result of the process, to be simple, predictable and treatable. The parameters of cost, stated period and effort perhaps are focused on literature, this evidence how much the software process is characterized to stay undone. It is also verified that they are narrowly dependents of the human resources of environments SDE. For example, estimate of software cost is made measuring the time of required effort to complete a software product. Effort usually is represented by the human measure (HM). Estimate of cost is important because it offers to a basic parameter (essential) for the planning and accuracy of this parameter, controlling of projects would not have base to determine how much time and effort each phase of software and activities would be expenses, and them they would not have cash skill to monitor, to control the projects. They would be without parameters.

An estimate method can be cited as for example: COCOMO (Constructive Cost Model) of Boehm (BOEHM, 2000). Some other hybrid methods are considered esteem other different aspects, as example the aspect of the interpretabilities of the Diagrams of States of UML (Unified Modeling Language) carried through by Piattini (PIATTINI, 2002). Piattini presents a specific methodology that combines the Fuzzy Logical use with mathematical model of prediction that searches the average time expense for a person to interpret a Diagram of States of the UML. As it happens with other methods, each one has its limitations, advantages and disadvantages, and none of the cited alternatives is better of the one than another one considering all the aspects globally and nor considering separately.

4. THE INTERACTIVE ENVIRONMENT OF ESTEEM

Some identified characteristics already for this environment lead the decision of the choice and the definition of the structure of the environment and the tool most adequate, as well as some necessary elements (and characterized) that it will complement this environment, with for example the language of interface, database, definition of the tool of simulation etc.

The definition most efficient das requisite activities as das identified activities no process of to be defined software, and the allocation of executors that stops Engineering of Software is actors das activity for the tasks of agreement with profiles and attributes, will have to be evidenced and will guide the construction do model do process, had answered for its responsibilities and diverse resources could be controlled (private, defrayed etc), placed.
Aspects related to the activities are part of the process model, and will introduce modifications in the elements of the model of the process. From the simulation resulted of the carried through introduction it can be known by means of the curves of estimates and gauging.

Obviously this process is dynamic, being thus continuous improvements they are identified and waited. The USA is the proper simulation for alternating choices that will lead ahead in the definition of the accepted process in that instant. Ahead of this the results of the simulation answers had started to be the new entrances in a new scene of simulation. Part of this proposal is being developed in one another work and will be tied with the research of the group of Simulation and Management of LAC (NEMESIS/LAC)[3].

5. PROCESS RUP IN THE SIMPROCESS - CASE STUDY

The choice of the RUP for the accomplishment of the experiment that will be detailed to follow elapses of the following facts:

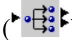
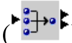
[3] Núcleo de Estudos em Modelagem e Simulação de Sistema/Laboratório de Matemática e Computação Aplicada

- The RUP presents a set of processes whose practical application is consecrated in great number of companies
- The used processes well are detailed through diagrams of activities.

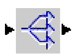
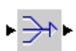
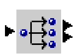
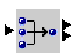
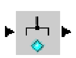
Studies exist that use the methodology proposal in COCOMO II (Constructive Cost Model) (BOEHM, 2000), for the accomplishment of estimates of effort and time of development of the projects developed through the RUP. These studies will be used later in this work during the simulation of the execution of projects. In the diagram of activities presented in Figure 2 (of illustrative character), it can be observed the involved flow of processes and activities in the phase of transition of software using the RUP. In it they appear the some involved actors in the accomplishment of each process, as well as, its chaining during the execution.

5.1. Conventions used for modeling of the processes in the environment of simulation - Simprocess

The Simprocess presents a series of easiness for modeling of processes, activities and tasks, as well as, of the flows of control between them. However, the notation for the conversion of the flows of processes and activities considered for the RUP, to the Simprocess environment do not occur in direct way. For its execution it was established a series of rules and conventions that better are detailed to follow. Through them, it is intended that in the future other processes can be mapped using criteria uniforms, in way to allow the comparison of the gotten results.

The use of this technology is illustrated in way simplified in Figure 3, that it presents the structure of activities of model RUP modified and shaped using the diagrams of workflow in hierarchic levels. The shown points of separation in Figure 3 are two types: And-Split () and Or-Split (). The And-Split is a point of control where the way of execution of the activities is broken up in two or more lines with activities executing in parallel. The Or-Split will go will make possible that only one of the ways of activities parallel bars is executed. The return of these activities parallel bars will give through points of control of And-Join () and Or-Join (). Some other details of interpretation are presented in Picture 2.

Picture 2. Interpretation of the elements in the Simprocess environment

	A complex process can be detailed in the Simprocess in some levels, being enough this to only clicar on estereótipo used to represent it. This easiness also can be used to add in one same level some similar processes, or that they play complementary actions. This resource must widely be used as it forms to diminish the complexity of the shaped flows.
	A simple task can be shaped through the interval of necessary time for its execution (delay).
	The control flows are represented in way to indicate the direction of the flow of the process.
	Flows that represent executed cycles of iteration by means of a finite sequence of operations, where the object of each one is the result of that it precedes it. Each one of these flows must have an execution condition that must be evaluated before its início(branches).
	The set of flows gone off through a ramification normally is locked up by a meeting (merge).
	They represent the separation of a flow in some competing flows, and must be initiated at the same time (splits).
	The set of flows gone off through a division normally is locked up by a junction (join).
	Allocation of resources to one determined process.
	Release of resources placed to one determined process.

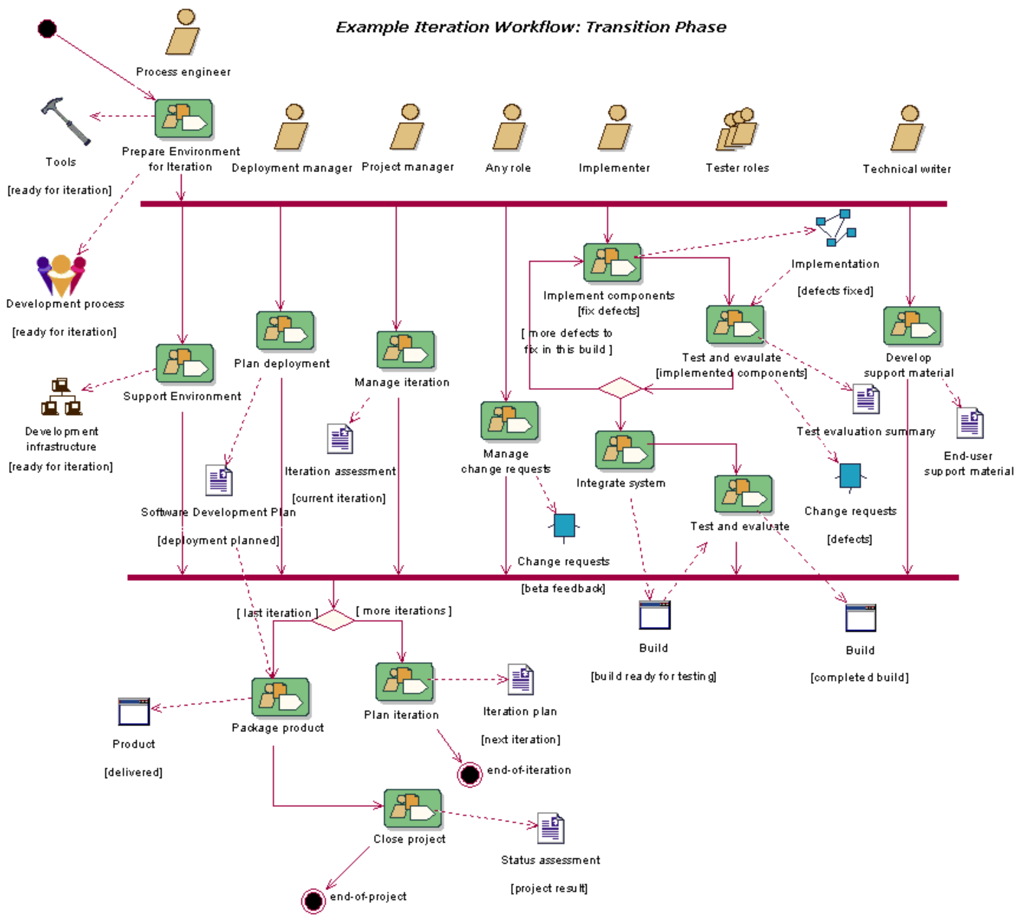


Figure 2. Workflow of the process of transistion in the RUP (http://rational.ibm.com , 2005).

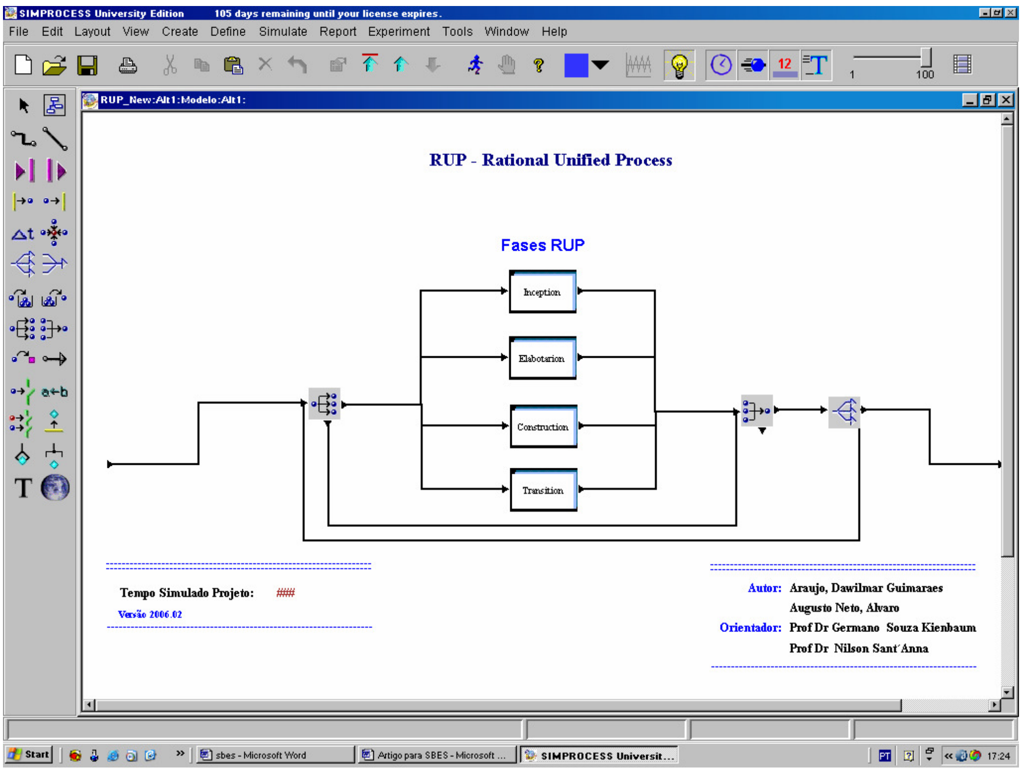


Figure 3. Structure of activities (Phases) of model RUP implemented in the Simprocess environment

5.1 Used parameters esteem effort and time of development

Its last version, called COCOMO II (BOEHM, 2000), represents an evolution on the works and allows that it is applied in the development of software that the objects use modern techniques as orientation, with aggregation of suitable or reused code, metric based in function points or object points, based processes of development in evolving models as the RUP. Thus, he adopted himself for this work from model COCOMO II (BOEHM, 2000); esteem the effort necessary to develop software:

$$\text{PMNS}= \text{A x SizeE x} \prod_{i=1}^{n} EMi \qquad (1)$$

Where:
PMNS is the people-month number that must be placed to the project, working during the normal period of work (nominal schedule) it is a factor that must be calibrated through Bayesian analysis of multiple regression.
Size is an estimate of size of software, evaluated in thousand of lines of code-source (SLOC-Source Lines of Code).
EMI they are multiplying of effort attributed in function of the characteristics of the process and the model of development that will be used. Its values are attributed from data calibrated for the author. In the Annex one brief description of each one of these multipliers and the values attributed to each one of them meets it.
E = is a factor calculated through the following formula:

$$\text{E}= \text{B} + 0{,}01 \text{ x} \sum_{j=1}^{5} SFj \qquad (2)$$

Where:
B is a factor that must be calibrated through Bayesian analysis of multiple regressions.
SF is factors of scale that they are attributed in function of the possible economies or diseconomies of scale that can occur in the development of the project, caused for the experience of the team with similar systems, involved degree of maturity and cohesion of the same one, risks and necessity of conformity with preexisting requirements. Esteem the time necessary to develop software, COCOMO II (BOEHM, 2000), as it uses the following expression:

$$\text{TDEVNS} = \text{C x (PMNS)F} \qquad (3)$$

Where:
TDEVNS is the number of months that must last the development do project, working during the normal period of work (nominal schedule)
C is a factor that must be calibrated through Bayesian analysis of multiple regressions
F = is a factor calculated through da following formula:

$$\text{F} = \text{D} + 0{,}2 \text{ x (E-B)} \qquad (4)$$

In studies taken for calibration of the model the author collected given in 161 projects of software development, whose size varied between 2 and 512 a thousand lines of code, and arrived at the following average values for the constants A= 2,94, B = 0.91, C=3,67 and D=0,28. The alert author who stops getting more necessary estimates, at least the factors and the C need to be calibrated in the development environment (Boehm 2000).
The functions are described in the Simprocess environment by means of scripts of appropriate functions Figure 4.

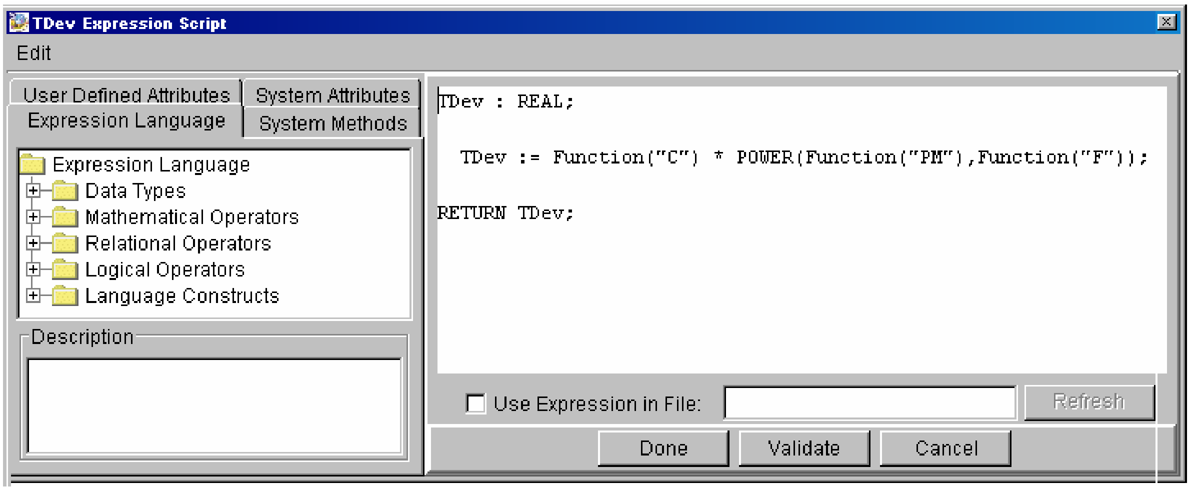


Figure 4. Script of the function of I calculate of development time

5.3. Exercised initial experiments for validation of the integration

Steps of the experimentation (considered for the exercises):

1.Definition of an only entrance in the system: number of projects as entities of the system to be executed simultaneously.

i. This value is given in the model as parameter of entrance in the icon (generator)

ii. For each project (entity) the system generates automatically: one size (2k the 512 k) corresponding to the size of the project and uses it as attribute III. For each project a value of IN (effort) is calculated on the basis of the table proposal for Boehm (BOEHM, 2000). This value also is referred as attribute of the entity. The table proposal for Boehm was inserted in the Simprocess environment. These tables if find in the Annex the IV. The values of SF for each project also are calculated in function of the table proposal for Boehm and also inserted in the Simprocess.

iii. These entrances are carried through in the system for May of the menu of services illustrated in 5 Figure and Figure 6 by means of scripts and discrete distribution of probability.

2.With the generated values the model calculates p.m. and TDev for the project.

3.On the basis of the results of item 2, calculate the values of p.m. and TDev for each phase of the RUP. 4, Wheel the system until the corresponding values to each phase to reach the percentile proposals of Boehm (BOHEM, 2000), as Table 1. 5. It was observed, one gotten excited the parameters of the model to exercise the management and was looked to know as for example: i) Number of iterations necessary to reach values of reference of the phases; ii) Number of carried through iterations focusing more efforts in determined phase; e iii) Modification of characteristics of tables of references (SF for example) modifying the behavior of the model. As for example LTEX-Capability and knowledge of the team in Tools and Languages. Vide Prices A1 and A2, Annex.

Tabela 1. Interpretation of the values of reference for each phase of the RUP (BOEHM, 2005) versus experiment

Phase	Effort (ref)	Time of development (ref)	Exp. (ref)	Experiment (ref)
Conception	5%	10%	5%	10%
Elaboration	20%	30%	18%	28%
Construction	65%	50%	64%	53%
Transition	10%	10%	13%	7%

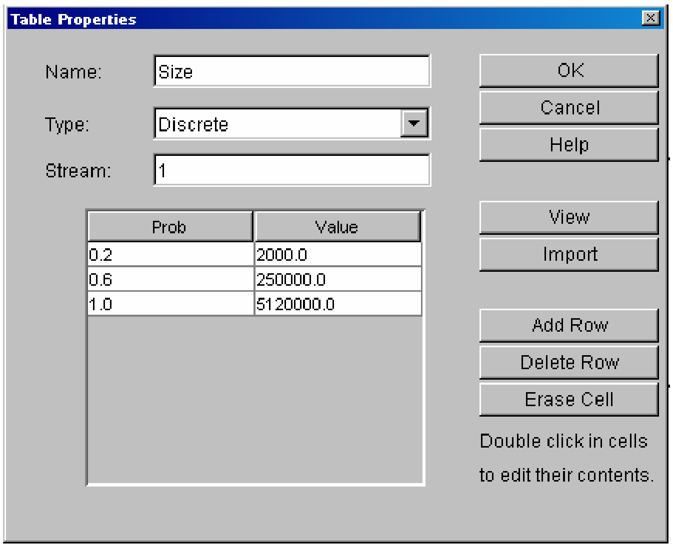


Figure 5. Discrete distribution of probability for the entrance parameter size

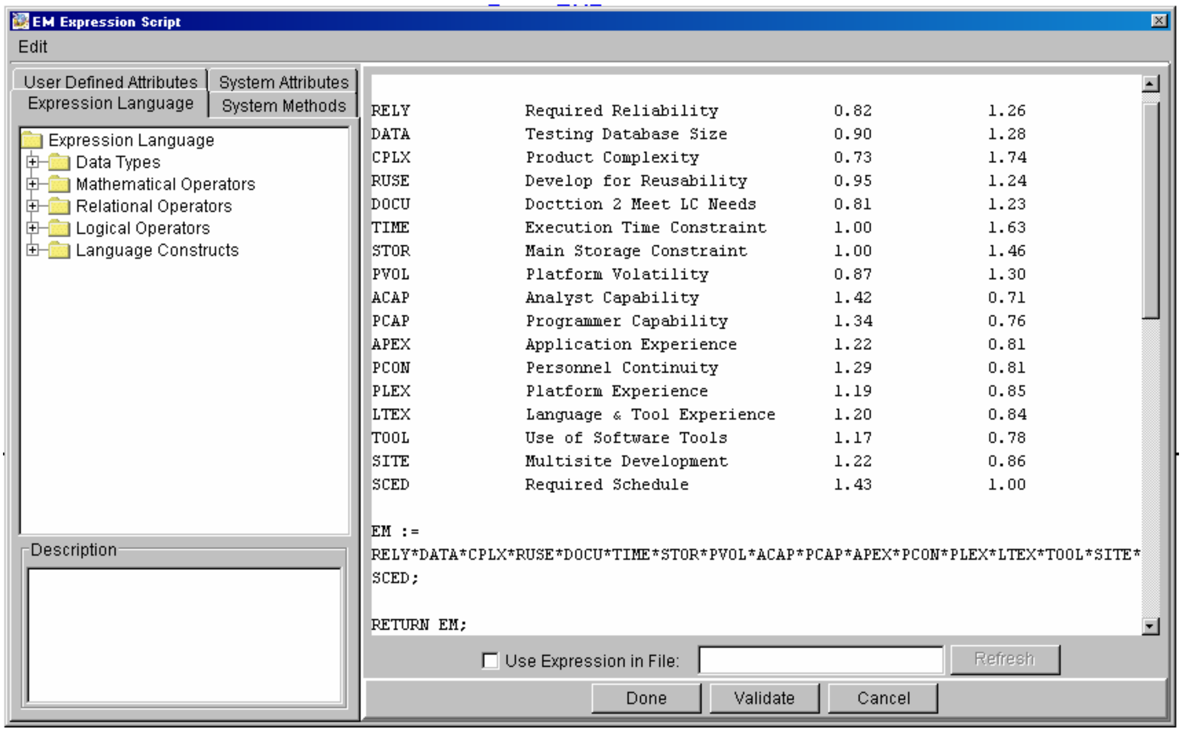


Figure 6. Way of insertion of parameters in the Simprocess. Annex

It was looked to keep the parameters sufficiently next to the experiments of Boehm with relation to COCOMO II, what it allowed for this scene (hypothetical) carried through studies to confirm the estimates of times and groups with the attributes of the characteristics in the average. The studies still meet in progress, as example, already a comparative result of the parameter can be presented time of development (Tdev) for each phase of the RUP.

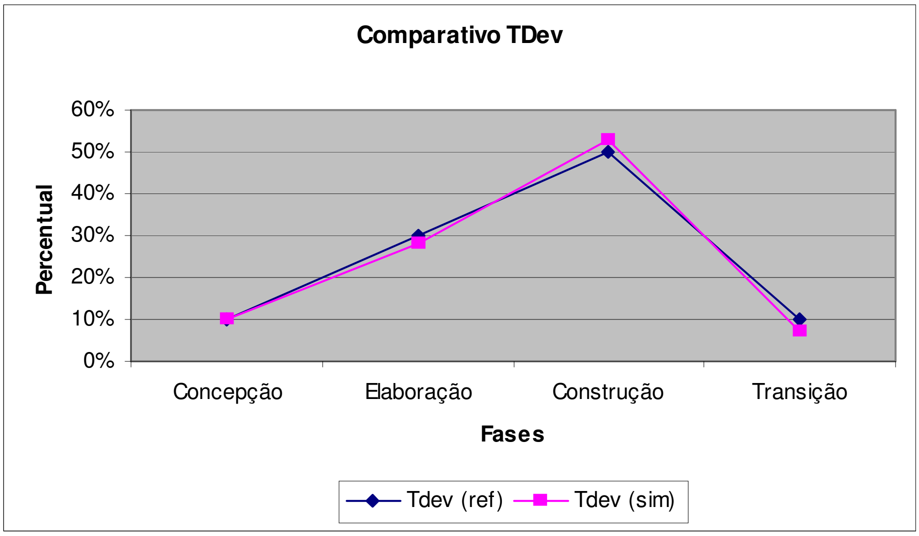


Figure 7. Partial result of the TDev parameter (Time of Development) of the Phases of the RUP:x Experiment.

6. CONCLUSIONS

Processes of software development are very dynamic, and influenced for the pressure of the competitiveness for delivery products of software with quality in time and reasonable cost. To implement changes in the processes they finish being difficult, expensive and they can introduce errors.

In this direction the Simulation comes to provide with viable form and cheap application to evaluate and to reduce this risk through you analyze quantitative of modification of processes considered in terms of performance of the process in diverse experimental scenes.

A good work of modeling and simulation also gives support to the processes of taking of decisions in the organizations.

The definition of distribution curve was seen as a great chance of this work, and with the use of simulation to support this intention for the vision micron-estimate of process it expects to get a greater and more insurance forms to deal with the aspects related with the process of software and the aspects of software environments as stated period, cost, effort etc and the definition of teams.

One still expects other benefits of this proposal from the conclusion of the development of an integrated environment of simulation capable to deal jointly with the cited complementary aspects to the two areas of study. It is in studies for future steps in the direction of this research to explore two perspectives horizontal and vertical for definition of processes specialized from characteristics of processes organizational standard. For the vertical perspective one adopts proposal presented it by Falbo (1999). Already for the horizontal perspective, one meets in study to detail the process illustrated in the Figure 1na form of a new boarding of process definition, combining with the previous one (Falbo, 1999).

REFERENCES

Araújo, D G-Kienbaum, G S-Sant´Anna, N. (2006), *Estimativas em processos de software através da simulação*. II UNEM-WorkShop de Integração Universidade-Escola: UNITAU-Alston, Set. Taubaté-SP.

Araújo, D G-Kienbaum, G S-Sant´Anna, N. (2005), *Proposta de um ambiente de apoio a estimativas em processos de software através da simulação*. V WorCap do INPE. São José dos Campos-SP.

Boehm, B W. e outros. (2000), *Software Cost Estimation with COCOMO II*. Upper Saddle River: Prentice-Hall.

Falbo, R A and Bertollo, G, (2003): *Apoio Automatizado à Definição de Processos de Software em Níveis*, 17 SBES-Simpósio Brasileiro de Engenharia de Software. Manaus-AM.

Hauston, D X at al. (1999), *Behavioral characterization: finding and using the influential factors in software process simulation models.* The Journal of Systems and Software, 59, 259-270, Mar.

Ioana, R A; Collofello, J A; Lakey, P B. (1999), *Software process simulation for reliability management*. The Journal of Systems and Software, 46, 173-182, Nov.

Kellner, M I; Raffo, D M and Madachy, R J. (1999), *Software process simulation modeling: Why? What? How?* The Journal of systems and software, 46, (2-3), 91-105, apr, 15.

Piattini, M at al. (2002), *Construcción de un modelo de predicción para el entendimiento de los diagramas de estados en UML.* Grupo ALARCOS. D Departamento de Informática, Universidad de Castilla-La Mancha. Ciudad Real (España).

Raffo, D M and Harrison, W. (2004), *Combining Process Feedback with Discrete Event Simulation Models to Support Software Project Management.* International Software Process Simulation Modeling Workshop (ProSim 2004), Edinburgh, Scotland, 24-25 May.

Raffo, D M; Harrison, W and Vandeville, J. (2002) *Software Process decision support: making process tradeoffs using a hybrid metrics, modeling and utility framework.* Proceedings of the 14th international conference on Software engineering and knowledge engineering. ACM International Conference Proceeding Series. pgs: 803 – 809. July.

Ruiz, M. (2003), *Modelado y Simulação para la Mejora de los Procesos Software*. Tesis Doctoral. Departamento de Lenguajes y Sistemas Informáticos. Universidad de Sevilla.

ANNEX

Tabela A1 - EM- Effort Multiplies - COCOMO II

Acrônimo	Descrição	Valor Mínimo	Valor Máximo
RELY	Required Reliability	0.82	1.26
DATA	Testing Database Size	0.90	1.28
CPLX	Product Complexity	0.73	1.74
RUSE	Develop for Reusability	0.95	1.24
DOCU	Documentation to Meet Life Cycle Needs	0.81	1.23
TIME	Execution Time Constraint	1.00	1.63
STOR	Main Storage Constraint	1.00	1.46
PVOL	Platform Volatility	0.87	1.30
ACAP	Analyst Capability	1.42	0.71
PCAP	Programmer Capability	1.34	0.76
APEX	Application Experience	1.22	0.81
PCON	Personnel Continuity	1.29	0.81
PLEX	Platform Experience	1.19	0.85
LTEX	Language & Tool Experience	1.20	0.84
TOOL	Use of Software Tools	1.17	0.78
SITE	Multisite Development	1.22	0.86
SCED	Required Schedule	1.43	1.00

Tabela A2 - SF- Scale Factors - COCOMO II

Acrônimo	Descrição	Valor Mínimo	Valor Máximo
PREC	Precedentedness	6.20	0.00
FLEX	Development Flexibility	5.07	0.00
RESL	Architecture/Risk Resolution	7.07	0.00
TEAM	Team Coesion	5.48	0.00
PMAT	Process Maturity	7.80	0.00