Meta-Process Used for Production Process Modeling of a Software Factory: The Unitech Case

José Augusto Fabri, Faculdade de Tecnologia de Ourinhos, Brazil, & Fundação Educacional do Município de Assis, Brazil; E-mail: fabri@femanet.com.br
Alba Couto Falcao Scheible, Unitech, Brazil; E-mail: ascheible@unitech.com.br
Paulo Marcelo Lessa Moreira, Unitech, Brazil; E-mail: paulo.marcelo@unitech.com.br
André Luiz Presende Trindade, Faculdade de Tecnologia de Ourinhos, Brazil; E-mail: altrindd@yahoo.com.br
Luiz Ricardo Begosso, Fundação Educacional do Município de Assis, Brazil; E-mail: begosso@femanet.com.br
Ana Paula Braun, Unitech, Brazil; E-mail: paula@unitech.com.br
Marcelo S. de Paula Pessôa, Universidade de São Paulo, Brazil; E-mail: mpessoa@terra.com.br

ABSTRACT

This paper proposes the utilization software meta-process theory in a factory context. A case study verifying the adherence of the proposal to the factory in operation also is presented.

1. INTRODUCTION

Brazilian software industry has been constantly working for increasing its quality and productivity. This information can be verified by analyzing the incentive programs promoted by Science and Technology Ministry (MCT) in which Brazilian government established that software is a priority (Software, Semiconductors and Industry). One of these programs is the SOFTEX (Society for Promotion of the Excellency of Brazilian Software). The objectives of SOFTEX are: to place Brazil into the top 5 biggest world software exporters and producers and to reach international standard of quality and productivity in this sector.

Beyond of those programs, the Ministry develops, periodically, a research to verify quality and productivity attributes of Brazilian software development sector. In Brazil exists about eleven thousand companies with activities related to the development and commercialization of software, employing about 160.000 workers, 25% of these companies have a quality program defined, and other 26% feels the need to establish this kind of program. This fact shows that Brazil is conscious about the need of improving the quality of its products in IT sector, in this case, the software sector. (www.mct.gov.br/sepin - November, 2006).

Parallel to the these facts, COSTA (2003) presents a research involving the 31 most significant companies, which work at Brazilian market using the Software Factory model. From these, only 41% apply a software development complete cycle; 45% apply proper methodology; 16% use projects control tools; 14% possess CMMI certification; 13% use CASE tools and 10% apply quality metrics.

Based on the context presented above, it is possible to affirm that to reach the objectives traced by SOFTEX, it is necessary an effort of the UNIVERSITIES, COMPANIES and of the GOVERNMENT with the intention of becoming aware of the market about the existence of the quality and productivity models for the theme software factory.

So, the objective of this paper is to propose a meta-process for the production of a software process. The meta-process will be used in the modeling of the process of UNITECH[1] Software Factory.

2. THEORY ABOUT META-PROCESS

Sommerville (2003) defines that a software process has activities, methods, practices and transformations set which, partially, reach a determined objective.

Reis (2002) indicates that the meta-process provides the development of executable processes for the software production.

Feiler and Humphrey (1993) propose a group of activities to configure the basic structure of a meta-process:

- Analysis of process requirements: social and technological aspects of a certain environment configure the process requirements.
- Process modeling: results in a process abstract model that should be materialized by a process modeling language (PML).
- Instantiation: in that meta-activity the process engine is configured. That engine should act, directly, into the software process and it has the following functions:
  - to support and to monitor the cooperative development of the stakeholders in the process;
  - to register data generated during the execution process;
  - to guarantee the composition of the defined activities in the process;
  - to manage the versions of the information generated by the process;
  - to collect, automatically, the metrics generated with the process execution;
  - to make possible the process changing during its execution and;
  - to manage the resources allocation inside the process.
- Simulation: allows to foresee the problem and to esteem the process duration when it is under production. If problems are detected in that activity, the process planner should return to the previous phases so that fittings can be done.
- Execution: a real project is executed with the process created.

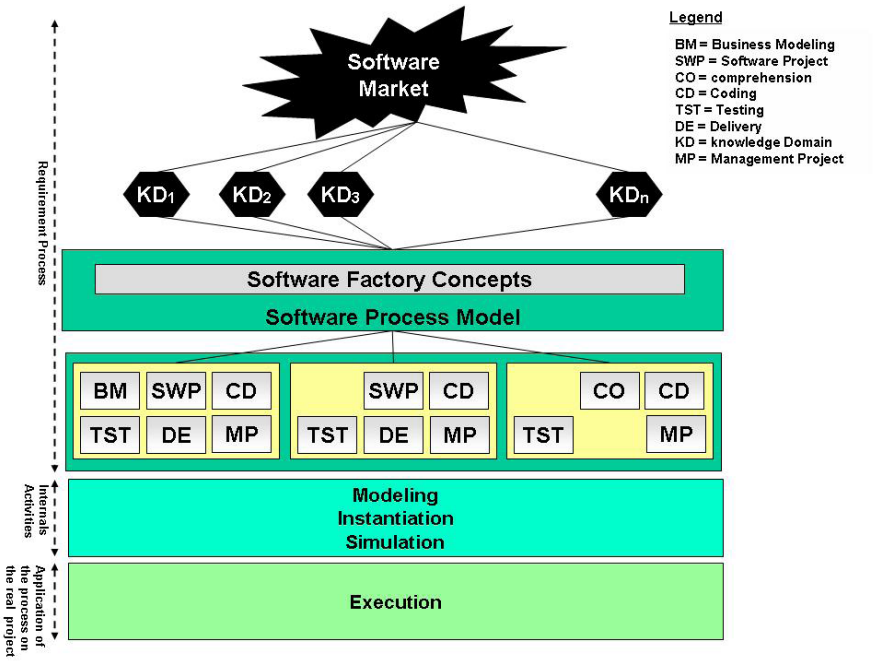
3. THE META-PROCESS PROPOSAL

The meta-process proposed by this paper is divided into five meta-activities: requirements, modeling, instantiation, simulation and execution.

In the requirement meta-activity it is necessary to define:

- Production cycle: The cycle presents the type of products generated by the factory. It can be classified as:
  - Long: The software factories which are framed in this concept should accomplish the business modeling, software project, coding, testing (modular or unitary, integration), project management and, finally, the software delivery (installation and training, for example). The long cycle factories should have a strong standardization to the activities of business modeling and software project. It is important to point out that activities depend on the creativity of system analyst and software architect.
  - Medium: The software factories which are on this cycle do not care about the business modeling; the responsibility of the factory begins with the execution of the activities of software project. It also accomplishes code, tests and project management.
  - Short: The software factories which are on this cycle make the activities of comprehension (the comprehension has the objective to verify if the

Figure 1. Structure of the meta-process proposal



project specifications is correct, consistent and intelligible, in order to reduce the stop risks in the production or the breaks of productivity), coding, testing (component and integrated) and project management. The business modeling, the software project, should be accomplished by long cycle software factory.

- Knowledge domain in which the factory will operate: the definition of domain is, specifically, the line products concept. Which products will the factory produce? This question should be answered.
- Process model (waterfall, incremental) that the factory will operate: The definition of the model will influence the dynamic behavior of the software process, knowing that the model defines the activities relations about the temporary optics.

The modeling, instantiation, simulation and execution activities are described on Section 2.

Figure 2 represents the meta-process used in the modeling of a software process. The figure shows the software development market divided into several segments (knowledge domains), the cycles (long, medium, short) and the conceptual aspects about software factory which are used taking into consideration the chosen model. The modeling, instantiation, simulation and execution activities, also, are contemplated in the figure.

4. ADHERENCE OF THE META-PROCESS TO THE SOFTWARE FACTORY: THE UNITECH CASE

The information presented in this section has been based in a case study realized in the UNITECH Software Factory.

Figure 2. UNITECH process modeling: PML used: IDEF-0 notation

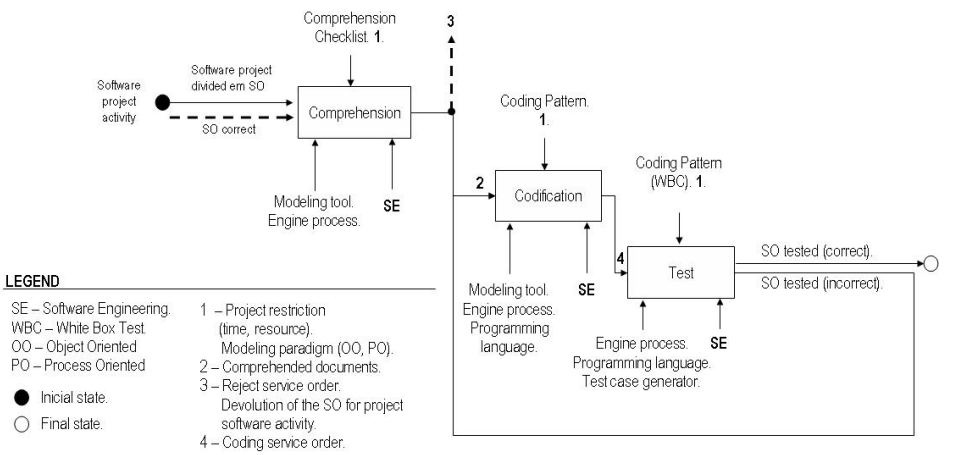
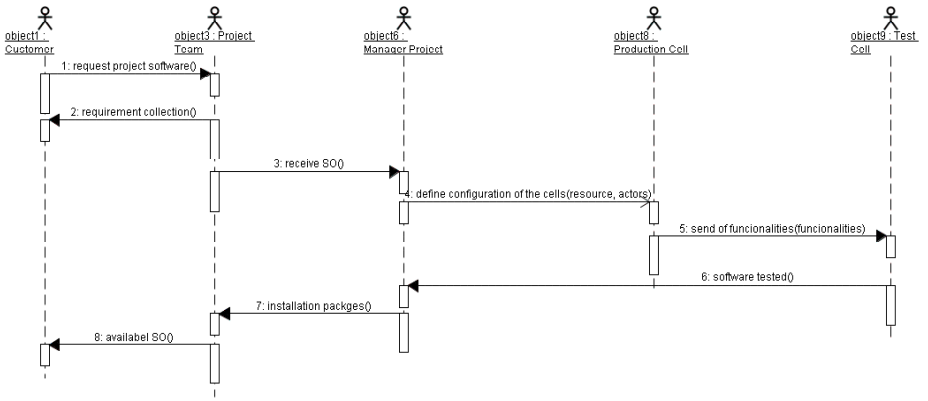


Figure 3. An instance of the real execution of the UNITECH process



The UNITECH was created in 1995 and, nowadays, has about 1266 professionals. In 2002, the company had 584 employees and annual revenue of 18 million dollars. The forecast for 2007 is to reach the 1500 employees with revenue of 50 million dollars. The UNITECH growth rate has been maintained in the last years around 25%. In relation to the quality certification, UNITECH got CMMI level 3.

The company produces software for the following segments: telecommunications, administration of engineering companies, customer relationship management (CRM) and tributary administration for municipal districts.

The process model used by the company is characterized as incremental. The software factory production cycle of UNITECH is defined as short, which is, the comprehension, coding, and testing activities are executed.

After the process requirements are defined, the next activities of the meta-process are modeling and instantiation. The first of them can be verified in Figure 2.

The instantiation demands the configuration of the process engine. In UNITECH such machine has these following functionalities:

- To maintain[2] the production process activities;
- To maintain the service orders, derived from the software project;
- To maintain the stakeholders of the software process;
- To maintain the customers and their software projects;
- To maintain the developed products, which are related to the services orders;
- To store the stakeholders, errors and the development time for each product in each version;
- To relate the products developed to the customers' projects.

Finally, it is important to point out that the UNITECH software process is already under execution, and so, it is not possible to execute a meta-activity simulation. An instance of the real execution of the process is presented in Figure 3, (in case of multiple instances, this is necessary to develop new diagrams). Through the illustration, we can observe that a **Customer** requests a software project to the **Project Team**. This **Team** begins the collection and analysis of the requirements, as well as logical and physical project. The **Project Manager** verifies if the Software Factory has conditions, specially, in relation to time and technology, to produce the requested software.

At this moment, the production cycle of the factory begins. The **Project Manager** receives the SO and defines the configuration of the **Project Team**. The specification comprehension is led; the software production begins and the functionalities implemented are sent to the tests activity. Tests are executed and, if there are no errors, the SO is liberated for delivery to the customer. The **Project Manager** packs such SO, sending it to the **Project Team**. Finally, the Project Team delivery the product requested in the SO to the customer.

5. CONCLUSIONS

This paper presented the possibility of using the meta-process theory for the definition of a software process.

We could verify that the requirements meta-activity has a great importance to the process definition, because it is where is defined the knowledge domain, production cycle and process model.

In the modeling meta-activity was used like PML, the notation derived of the technique IDEF-0, it is important to point out the efficiency of this notation for the modeling process.

The instantiation and simulation meta-activities were not verified, because UNITECH already has a software process implanted.

Finally, as future work authors intends to verify the meta-process under the view point of new cases.

REFERENCES

Costa, Ivanir; *Contribuição para o aumento da qualidade e produtividade de uma fábrica de software através da padronização do processo de recebimento de serviços de construção de softwares* - 174 pag.; Tese (Doutorado) Apresentada ao Departamento de Engenharia de Produção da Universidade de São Paulo; São Paulo: PoliUSP, 2003.

CUSUMANO. M. A. *Japan's Softwares Factories*. New York: Oxford University Press. 1991.

Feiler, P.; Humphrey, W. Software Process Development and Enactment: Concepts and Definitions. In: *International Conference on the Software Process, ICSP Proceedings*. Berlin, Germany: IEEE Computer Society Press, 1993.

REIS, R. Q. APSEE: *Um Meta-Modelo para Apoiar a Reutilização de Processo de Software*. Tese de doutorado apresentada ao Instituto de Informática da Universidade Federal do Rio Grande do Sul. 2002.

Schaefer, Wilhelm; Fuggetta, Alfonse; Gobart Claude, Jahnke Jens. Architectural Views and Alternatives. *Lecture Notes in Computer Science. Software Process: Principles, Methdology, and Techonology*. Derniame, Jean-Claude; Kaba, Badara Ali; Wastell David Eds. Vol. 1500.. ISBN. 3-540-65516-6. Pages 95-116. Springer-Verlag. 1999.

Sommerville Ian; *Engenharia de Software* – 6a. Ed.; Addison Wesley, 2003.

ENDNOTE

1 The UNITECH is a Brazilian company certified CMMI level 3.

2 In this paper the word maintain translates the idea of storing, deleting e consulting such data.