# Interaction of Processes and Tools to Meet the System Development Criteria

Citlalih Gutiérrez, Sergio Díaz, Eréndira Rendón, Itzel Abundez, Horacio Pacheco, Rafael Cruz

Instituto Tecnológico de Toluca, Av. Instituto Tecnológico s/n. Ex-Rancho La Virgen. Metepec, Estado de México, México
Contact: {citlalihg, sdiaz, erendon, iabundez, pacheco, rcruz}@ittoluca.edu.mx
*Paper received on 04/08/08, accepted on 10/09/08.*

**Abstract.** Proper information management, compliance with the purposes established and adequate communication derive all from the interest in having a complete set of knowledge and methods required in system development. Considering this, the research carried out for this paper defines a methodology useful in System Design. The method starts from a common-modeled platform and defines the way in which two processes closely linked to System Development interact: Product Design and Project Management. Currently, both processes occur independently from each other and each of them has its own tools and methods, therefore generating cost and time losses. The purpose of this research is to identify the technologies and knowledge obtained from both processes that will result in highly functional systems.

## 1 Introduction

The work herein described focuses on the interaction of processes and tools that meet the criteria of descending design [1], [2].

This paper is organized with the following structure: section Two describes the theoretical basis of this research; section Three explains how the Design and Management processes are carried out, and it also offers abundant information about the methodology that allows designing a common platform where the design and management processes interact with several tools under the context of a common platform; section Four describes a technical example corresponding to the area of robotics, which we have used to support our results.

## 2 Methodology Bases

The present research is based on the recommendations of the Electronic Industries Alliance-issued international standard EIA-632, with its main purpose being system development [3]. This standard methodology considers several processes to design systems, which range from an article's production up to its market deployment. It is organized into thirteen processes, grouped in 5 sets: *Technical Management, Purchase and Delivery, System Design, Product Manufacturing and Technical Evaluation*; from these, as per our research, we focused our interest in the processes

© M. G. Medina Barrera, J. F. Ramirez Cruz, J.H. Sossa Azuela, (Eds.)
*Advances in Intelligent and Information Technologies*
*Research in Computing Science 38, 2008, pp.365-375*

within these 5 groups and in the topics identified as: *Planning, Solution Definition and Verification*.

EIA-632 aims to allow interaction between its processes. Therefore, this is the most complete standard according to our field of interest. However, it does not define the methods and tools to make these processes interact. This is why this work and the problem to be solved are unique, because it proposes a methodology that clearly indicates the interaction between the different processes constituting the system development, as well as the tools that may be used.

The present work is also supported by the Model Driven Architecture (MDA) [4] proposed methodology, which offers a different manner to develop software systems by using the concept of "models". Such concept allows rapid identification of technical issues, as well as information reuse.

## 3 Common Platform

### 3.1 Product Design Process

This process integrates several stages that span from the product's specifications to its manufacturing. At the LAAS-CNRS laboratory in Toulouse, France, this process is carried out as follows:

The customer requirements stated in a specifications document are implemented in natural language in a word processor [5], [6]; then, a deep analysis is carried out that aims to a detailed study of the design, validating results in a LAAS laboratory-developed software, called TINA [7]; the final stage consists on carrying out a virtual prototype of the studied system. Such procedure eases the identification of technical restrictions and specifications upon which the system's operability will be based.

### 3.2 Project Management Process

This is a two-phase process, the first one consisting on the definition of every task required to manage the project, mainly identifying resources, materials and every necessary issue; the second phase consists on optimizing and selecting the tasks complying with the system's goals; this phase uses the GESOS [8] tool developed in the LESIA laboratory at Toulouse, France. The purpose of this tool is to build a number of scenarios that will provide viable solution alternatives to carry out the project. The Project Management procedure allows for the identification of the system's non-operational restrictions [9].

### 3.3 The Product Design - Project Management Association

This stage of our research dealt with studying the way to associate the product design and project management processes, which had been treated independently up until recently in the LAAS and LESIA laboratories. The purpose of this association was to define a modular architecture (with structural-operational modules) that would contain the system's functions and structure [2], [10].

The proposal of the design-management relation is based on the life-cycle Model in "Y" [11]. The system design derived from this model offers stages defining the possibility to describe the system according to different levels of abstraction. The "Y" methodology-guided proposal allowed for the integration of each internal process of the design and management steps, as well as for the tools used by each of these steps; such process is referred to as "*weaving*", as shown in Figure 1.

Continuing with the design issue, a system consists of organs (or components) as per its internal structure, and of functions as per its performance; therefore, it is normally divided into two: the functional and the organic-organization scopes [12], described according to a group of levels. These two points of view identify restrictions at different domain levels, like the functional, technical, regulatory, organizational, logistical and security levels, among others. Following, we offer details of each stage constituting the "Y" model.

Both the functional and the organic scopes derive from a specifications document that allows identifying requirements and exigencies.

### 3.3.1. Functional scope (left branch of scheme in Figure 1)

1. Extraction of functional exigencies: at this stage, the features to be performed by the system are defined.
2. Push-down design: hierarchical decomposition of the features required to comply with the functional exigencies.
3. Functional architecture: it details the system functional architecture by offering a hierarchical tree flowchart.

### 3.3.2. Organic scope (right branch)

1. Extraction of non-functional exigencies: this process details which issues will be covered, specifically regarding cost and time.
2. Inventory of organs and technical support: at this stage, a study of the existing physical elements required to manufacture the selected physical components is carried out.
3. Organs architecture: details the system physical architecture in a hierarchical tree flowchart.

In order to comply with the specific features of the "Y" model, fusion of the functional and organic data (central branch) is considered through a process of partition, resulting in three supplementary stages:

1. Structural-functional modules: this stage allows for the visualization of the first block (module) solutions integrating the system's features and physical organs (corresponding to the building blocks recommended in standard EIA-632).
2. Task definition: at this stage, each structural-functional block defines a specific task to be carried out.
3. Virtual prototype: this stage is relevant because it encompasses the system's test and validations phases before construction.

The most significant contribution of the present research work is at the stage that goes from functions to models. We have referred to this procedure as "weaving", which initiates with the projection of functional blocks over the organs blocks; such operation requires "block aggregation" tools. For this, we use the standard tool "HiLeS" (described in section 4.2) and the "projection" process, resulting in structural-functional blocks [11]. Let us see in detail how these processes are carried out.

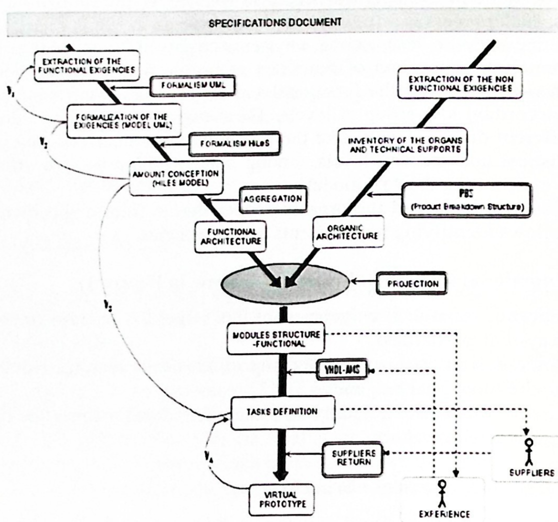


**Fig. 1.** Details of the cycle in "Y" associating the Organic description and the Conception Process (Common Platform)

The association of both Design/Management processes can be seen in Figure 1:

1. representation of technical design and
2. representation of management process

Stages in Figure 1 derive from certain processes, which are detailed below:

a. *Derivation of $Y_C$ towards $Y_M$*, the purpose consists on defining physical organs (structural-functional models) that will be used as the basis to define the project's technical tasks and up to the suppliers bid. The initial level in $Y_C$ derives from design, preliminary phase proposed by HiLeS, while the entry level in $Y_M$ corresponds to the pre-planning level.

Entry organs at this stage will be: the features defined in the functional model (HiLcS models) and the organic model components. Association of these two elements results in the structural-functional models.

b. *Pre-planning*, a first double, internal evaluation of proposals (technical and economic) will be carried out in structural-functional models based on the acquired experience and the company's experience. For such evaluation, we will integrate the

acquired knowledge (knowledge capitalization) obtained from structural-functional modules. This will allow for:
• the anticipation of technology choices,
• the anticipation of complementary tasks of data compression,
• the anticipation of costs to apply this pre-planning.

Pre-planning considers the technical specifications defined in the specifications document, which is then submitted to suppliers to be considered at each module to be carried out. The entry or input at this stage is a list of structural-functional modules, with each of these to be translated as a task to be taken care of and the product's organs of visual representation. The output will be a list of alternatives to carry out tasks with the information complying with requirements.

c. *The Bidding*, stage to receive the suppliers' responses.
• technical proposals, accompanied by physical models required to simulate the manufacturing of virtual prototypes,
• bids with delivery terms and costs.

These responses are aimed to the design job ($Y_C$ process) to then head pointedly towards the manufacturing of virtual prototypes; besides, they examine planning ($Y_M$ process) in detail.

The suppliers offer (technological bid) first provides every technology solution (maximum usage of the results from virtual prototypes) and economic considerations, where the GESOS procedure (described in section 4.3) takes place; through the alternative situations we propose (technical and organizational selections).

d. *Two-stage planning*:
• Stage to generate every alternative scenario. These scenarios are constituted by the best proposals from suppliers. The stage to select the best of these situations is carried out by GESOS according to "ad-hoc" criteria corresponding to the demands of the specifications document.
• The entry (input) level of this planning stage must have task alternatives with information updated according to the specifications proposed by suppliers; the result is made up of the most-interesting scenarios according to the project manager's point of view.

Finally, in a way similar to the "weaving" process in programming, according to the MDA, the design and management processes that we propose to associate will offer the elements to integrate (weaving operations) as output for each described stage to obtain a more complete product. This is then a set of transformations: each time the information is integrated, the process will result in a transformed output.

## 4 Demonstration supported by a Robot of automatic manufacturing of Biochips

This section has been divided as follows: we will first explain the technical example we have used to validate the process followed in our methodology, referred to as system of a Biochip-manufacturing robot, designed in LAAS. Secondly, we will detail the results obtained through the platform-based methodology upon which the design process is carried out, defining the four diagrams resulted from this process, as well as the information provided by each of them.

A biochip is a tool for genetic analysis. It is used to determine particular DNA sequences if a specific DNA is to be analyzed. A biochip is constituted by a miniaturized glass substrate over which several thousands of DNA strands can be deposited. The Biochip-manufacturing robot was built in the LAAS with an innovation [13]: micro-injectors, using 4 injection matrixes by acting on each nucleotide constituting the DNA (thiamine, guanine, cytosine and adenine). Each matrix is composed of a number of individual injectors required by hybridization units to properly function.

According to this description, we must focus our observations in the functional behavior to be complied with by the robot, particularly regarding specifications contained in the specifications document, from which the following functional requirements were identified:

- Make Biochips functional.
- Carry out a totally-automatic process.

Performance requirements are as follows:

- create a robot with a deposit density of 1000 / $cm^2$,
- high number of probes (1000) to be produced,
- fast and economic manufacturing,
- interface exigencies,
- free criteria to carry out deposits,
- possible manual control of process.

At this design stage, the functional and organic decomposition process was carried out with UML and with the HiLeS tool to get a system model that defines its functionality and behavior. The next step had to identify the tasks that would allow planning activities to completely set the system.

### 4.1 Result of UML Formalism

To formalize exigencies, the UML formalism [14] was used; it allows modeling systems by means of graphic diagrams, which in this work only used four out of the thirteen diagrams proposed. Results of these four diagrams are described below:

*Context diagram*: allows for a first useful representation of the system. With it, actors and their interactions with the system and the environment are identified.

*Diagram of cases of usage*: allows enriching the information coming from the context diagram by explaining the type of iteration between actors and system. Several usage cases were considered according to the document specifications [13] and others were the result of the developer's reflections and experience. Usage cases serve to model the system's operation.

Regarding the biochip-manufacturing robot application, reference is made to exigencies set for in the robot's document of specifications. Each usage case is developed with the assistance of a verbal syntax expressing the goal to accomplish.

*Sequence diagram*: describes the system's dynamic behavior in the context of specific operation. Such diagrams must depict delays of functions and they must also contain temporary exigencies indicated in specifications.

*Diagram of activities*: allows for the definition of a terminal functional architecture and a global logic of the system operation.

Our contribution in this process of exigencies formalization has allowed for the proposal of a process depicting the method to chain the already mentioned UML four diagrams. In summary, each diagram defines the following information:

- *The context diagram* defines the main function and interactive environment.
- *The diagram of cases of usage* carries out an inventory of every function.
- *The sequence diagram* considered time and usage sequence (for each case of usage)
- *The diagram of activities* synthesized preceding formalism data of the diagram of activities.

The first transformations of models in the four diagrams were carried out to comply at all times with the EIA-632 and the MDA.

### 4.2 Results of HiLeS Formalism

HiLeS is part of the methodologies used for product design and it was created at the LAAS. It offers an assistance language to accomplish high-level, push-down designs of systems. The process carried out by HiLeS shows the contribution where it is possible to see the transformation of detailed requirements in UML context and case of usage diagrams.

Generally, this process is divided into three phases (see Figure 2). The first one deals with translating and meeting the formalities of natural language-implemented specifications; after that, these are represented in a graphic editor identified by structural and functional blocks, as well as by Petri networks. The next phase consists on validating and simulating the formal representation using the TINA tool [7]. Then, a complete representation of the structure over the VHDL-AMS language is generated to execute a temporary simulation of the whole system.

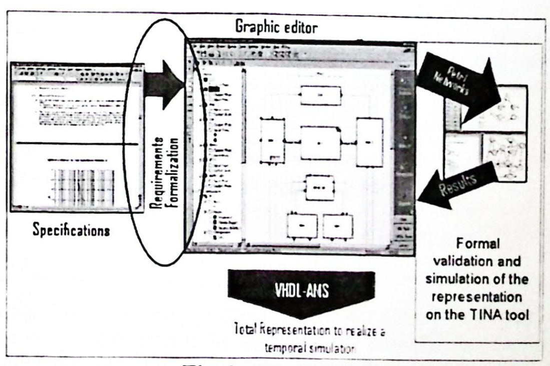


**Fig. 2.** HiLeS Platform

At this point, it is important to say that the transition of UML diagrams into the HiLeS formalism (Figure 3) is presently carried out directly and manually; for this, it is necessary to apply several rules of transformation according to the demands and recommendations of the MDA process.

The logical-functional modeling of the HiLeS-manufactured Biochip robot allows getting a logical representation at different levels, which will be verified and validated with the TINA tool.

### 4.3 Planning Results

Because system functions that will constitute modules are already defined according to the MDA recommendations, it seemed necessary to implement a tool to carry out such procedure; for this purpose, such tool was integrated to HiLeS, keeping several specifications, like the limitation of internal interfaces between functions and consideration of functions, also taking into account the same type or nature of the function, etc. This process allows for future capitalization and enhancement of processes, with a strong impact over future project costs.

In order to ease the process, we worked with the organic decomposition of the Biopuce robot, which clearly shows the different components at all decomposition levels, and then we proceeded with the task of recognizing each component.

Once system components were visualized and functions were defined, the next phase consisted on associating the functional representation with components; such process is carried out in compliance with initial goals and according to the stated non-functional exigencies and criteria of the system.

This process result is to obtain the scheme of Structural-Functional Modules, which we identified as Structural-Functional Modular Representation.

To verify the association process, we are now proposing two rules to be complied with:

- The first rule estimates that association of functions to components must be made considering the behavior of the functional block.
- The second rule requires association of functions with components to be carried out at all levels of abstraction.

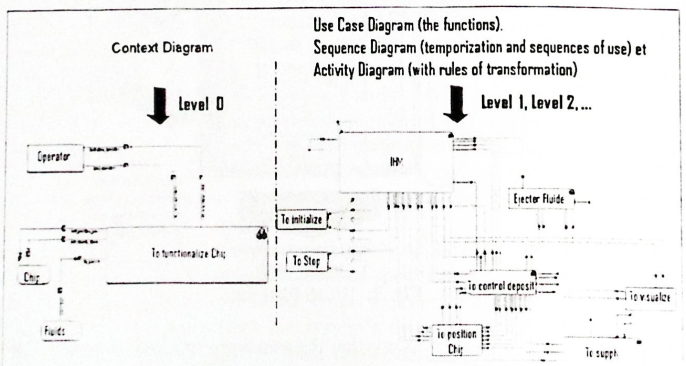


**Fig. 3.** Logical, formal and verifiable representation (HiLeS Tool)

The final stage is defined as work planning, consisting on the definition of tasks; such phase must consider that each structural-functional module is linked to a task that must be carried out. Therefore, each module describes production of the system in its environment, which serves as specification, and each defined task describes the method to carry this out.

The mentioned process offers as a result all the information required to determine the writing of project's tasks, because at this point there is a complete list of every task that must be carried out, precedence among them and a person responsible for each task. The project's writing shows relations of orders among every project task, linked by precedence restrictions; writing is always made right-to-left to reflect the project's chronology.

This process results in a set of task alternatives that comply with the goals specified for the project. If we alternate the combination of all these alternatives, we will obtain every alternative scenario required to carry out the project.

GESOS [8] is a tool that uses genetic algorithms to offer the project manager a reduced subset of good scenarios, which the manager will use to choose those that will allow him/her to design the system. Such tool works by using as a basis the information obtained from the representation of coded scenarios described in the previous stage and, along with genetic algorithms, economic and time criteria, the tool uses the proposed rules of Taguchi as rules of economic optimization, according to the goals stated in the construction of the Biochips-Manufacturing Robot.

At last, the information offered by suppliers was integrated to the first order plan, so the project manager was able to propose the optimum information to carry out the project.

## 5 Conclusions and Future Work

The ever-increasing complexity of Systems compels us to deepen design methods and tools to cope with the double goal of reducing products time-to-market and of strengthening quality and reliability results. Our contribution in this work is based on the idea of sharing and causing the interaction of information and tools of two complicated and already complete processes, like the design and management processes. This work has received the support of a great number of French industries, among which we can mention Airbus, EDS, Thales, etc.

Thanks to the chosen example, the application of our proposal could deliver optimum results [15]; however, we are conscious about having to increase the number of application examples to release our first conclusions.

In prospective, we must offer new testing and adaptations obtained from experience. There are several remarks to be made in the relations to be implemented with suppliers: knowing how to propose specifications in structural-functional modules and knowing how to define, in retrospective, the modules to be built and evaluate the information to make virtual prototypes. Furthermore, this work must still consider reuse, which consists on integrating the ideas suggested by the "return of experiences" methodology [16], aimed to capitalize knowledge and technical know-how.

## 6 Acknowledgements

We would like to thank the National Science and Technology Council (CONACYT) and the Instituto Tecnológico de Toluca for their support and sponsorship in carrying out this project, which is part of two projects of the Repatriation Program with application number 076636 and 076803.

We thank the DGEST for their sponsorship to carry out the projects "Analysis and Design of Functional and Logical-Behavioral Modeling of Systems" with code 907.08-P and "Study and development of the pneumatic artificial minimuscle prototype for robotics applications", which have been used to get resources and continue with researches related to this project.

We would also like to thank the support of the LESIA and LAAS laboratories in Toulouse, France, for their kindness offered for works made before this project.

### References

1. C. Gutiérrez et al. (2005). How to interconnect Product Design and Project Management including Experience Feedback and Reusability Requirements. IEEE 2005 International Conference on Information Reuse and Integration. Las Vegas, NV, USA. 23-32.
2. C. Gutiérrez et al. (2006). Design and Project Management: Their coordination on a platform of interoperable tools. INCOM2006, Saint Etienne, France.
3. James N. Martin (2003). Processes for Engineering a System. Overview of the EIA 632 Standard.
4. J .Bézivin (2001). From object composition to model Transformation with the MDA. TOOLS'USA. Volume IEEE TOOLS-39 Santa Barbara.
5. J. C. Hamon (2005). Thèse : Méthodes et outils de la conception amont pour les systèmes et les microsystèmes. LAAS-CNRS.
6. J. C. Hamon et al. (2003). HiLeS Designer: A Tool For Systems Design. Aeronautics, Automobile & Space International Symposium. Paris (France).
7. Berthomieua & M. Menasche (1983). An Enumerative Approach for Analyzing Time Petri Nets. IFIP Congress 1983.
8. S. Rochet, C. Baron (2006). An Evolutionary Algorithm for Decisional Assistance to Project Management in Handbook of Research on Nature Inspired Computing for Economy and Management.
9. C. Baron, S. Rochet and C. Gutiérrez (2005). Proposition of a methodology for the management of innovative design projects. 5th annual International Symposium of the International Council on Systems Engineering.
10. C. Baron, D. Esteve, C. Gutiérrez (2004). Towards a Shared Process for Product Design and Project. 14th Annual International International Conference on System Engineering (INCOSE), Toulouse, France.
11. Citlalih Gutiérrez Estrada (2007). Thèse : Méthodes et Outils de la conception Système couplée à la Conduite de Projet. LESIA-INSA. Toulouse. France.
12. J-P Meinadier (1998). Ingénierie et intégration des systèmes. Éditorial HERMES. pag 31-45. Paris.
13. L. Piazza et al. (2004). Conception et réalisation de l'environnement d'un microsystème de dépôt localisé pour la synthèse de l'ADN à l'aide du logiciel HiLeS-Designer. Rapport LAAS N° 04141. LAAS-CNRS
14. C. Morley et al. (2006). UML2 pour l'analyse d'un système d'information : Le cahier des charges du maître d'ouvrage. Éditeur : Dunod. Édition: 3e édition.

15. Jean Verries (2006). Méthodes de conception systèmes appliquées à un robot de dépôt localisé pour la synthèse in situ de l'ADN. Rapport du master. Toulouse France.
16. P. Clermont et al. (2005). Instanciation d'un système de retour d'expérience pour un processus de maintenance. 6[ème] Congrès International de Génie Industriel 2005. Besançon France.