RIDU
REVISTA DIGITAL DE INVESTIGACIÓN
EN DOCENCIA UNIVERSITARIA

REVIEW ARTICLE

Modelling as a Competency for the Training of Computer Professionals

La modelación como competencia en la formación del profesional informático

A modelagem como competência na formação do profissional da informática

Walfredo González Hernández
Departamento de Informática, Facultad de Ciencias Económicas e Informática,
Universidad de Matanzas, Matanzas, Cuba

Received: 08/09/16
Reviewed:02/11/16
Accepted: 15/11/16

ABSTRACT. Modeling is one of the most important activities in engineering activity. In the article the authors positions on the model and its classifications is taken. Furthermore, the essential elements of computer curriculum in the world and the importance of modeling them are assumed. In the last paragraph it discusses how to structure modeling as an essential element in their mode of action. In this section the structure of competence unfolds modeling analysis taking support current definitions of competence from the knowledge, skills, values and self-regulated learning processes. It is assumed in the article for the development of this competence the teaching of software engineering must precede to programming teach.

Keywords:
Informatics teach, Competence, Modelling.

RESUMEN. La modelación es una de las actividades más importantes en la actividad ingenieril. En el artículo se toma posiciones de los autores sobre el modelo y sus clasificaciones. Por otro lado, se asumen los elementos esenciales del currículo informático en el mundo y la importancia de la modelación en ellos. En el último acápite se analiza el cómo estructurar la modelación como elemento esencial en su modo de actuación. También se explica acerca de la estructura de la competencia modelar tomando como sustento el análisis de definiciones actuales de competencia a partir de los conocimientos, habilidades, valores y los procesos de autorregulación del aprendizaje. Se asume en el artículo que para el desarrollo de esta competencia la enseñanza de la ingeniería del software debe anteceder a la programación.

Palabras clave:
La enseñanza de la Informática, Competencias, Modelar.

Cite as: González Hernández, W. (2016). La modelación como competencia en la formación del profesional informático [Modelling as a Competency for the Training of Computer Professionals]. *Revista Digital de Investigación en Docencia Universitaria, 10*(2), 57-69. **doi:** http://dx.doi.org/10.19083/ridu.10.493
E-mail: Walfredo.glez@umcc.cu

RESUMO. A modelagem é uma das atividades mais importantes na atividade de engenharia. No artigo, os autores posições sobre o modelo e suas classificações é tomada. Além disso, os elementos essenciais do currículo de computador do mundo e a importância da modelagem deles são assumidas. No último parágrafo, ele discute como estruturar a modelagem como um elemento essencial no seu modo de ação. Nesta secção, a estrutura da concorrência se desenrola a análise de modelagem tomando apoio actuais definições de concorrência dos conhecimentos, habilidades, valores e processos de aprendizagem auto-regulada. Supõe-se no artigo para o desenvolvimento desta competência ensinando engenharia de software deve preceder programação.

Palavras chave: Ensino Tecnologia da Informação, habilidades, Modelagem.

The accelerated development of ICTs in this period has led to consider their introduction in society as a strategic process. However, highly trained professionals must conduct this process to carry out the functions it demands. For the training of this professional, it is necessary to investigate a set of problems inherent to the teaching-learning process of Computing.

In the historical development of computing, several scientific disciplines can be structured as part of the curriculum of the professionals in this science, while others cannot. However, there is certain content that is essential for their training and that are not framed in a single discipline. One of these contents is precisely the modeling of computerization processes.

Modeling is one of the most important activities in the training of computer engineers because of the need for an anticipated representation of the computerization processes to be performed. To do this, the analysis must occur in the order of the psychological personality structures that must be regarded as modeling in the training of the computer professional.

A first stage of analysis may lead to defining modeling as a skill. However, the scope of modeling also implies knowledge, values, self-regulation processes that suggest it is a competency for the components that comprise it and that are now accepted in the literature (Ramírez Oyarzo, 2013; Torra Bitlloch & Esteban Moreno, 2012). A starting point in this analysis is the role of the model and modeling in the computing activity and its essential form of organization. A second stage of analysis is modeling as part of the curricular design for computer professionals. Finally, determining the personality structure to which modeling in the training of this professionals corresponds.

MODELING AS PART OF THE COMPUTING ACTIVITY AND THE PROJECT

Models have been one of the most recurrent results of human activity in the history of humanity. The first man-made models, although not considered as such, are found in caves, whose walls were used for representations and as a means to express their wishes; at least that is what is known until now. Models have been used to pierce into the essence of phenomena related to man's cognitive and transforming activity, using them as an instrument of communication and anticipation of future activities. It is fair to say that models are a result of human activity.

Computing also uses models in the various activities it performs. Several authors (Epstein, Ping, Fogarty & Munson, 2015; Jayapandian et al., 2014) suggest the need to use models in computing in all its scientific disciplines.

For other authors (Pardo, Pino, García, Baldassarre, & Piattini, 2013) "... it is a representation of a computer process using formal elements. The model captures the important aspects of the process from an angle that is essential in that moment and omits that which is not essential." (p. 127).

ISNN 2223-2516

A positive aspect of this definition is that the model is acting as an abstract representation of a process that is necessary to computerize, but does not consider all the interrelations that are produced in it, but only those of interest. Therefore, it reproduces a certain reality depending on the interests of those who ask for the intervention of computing in their processes. It is also necessary to emphasize another model as an idea of computerization actions, as it will be analyzed next.

The solution to these problems starts mainly with the creation of an appropriate programming and people's tasks assignment before the computerization process. To achieve this purpose, the need to define computerization actions that use models and formal design and implementation structures emerges.

In (Pardo et al., 2013), models in computing are recognized as being diverse and needing integration. This integration responds to the needs of the process that is intended for representation, the organization and the needs of the customer that may derive in a single model. From this statement we can deduce that a model in computing may comprise several elaborate models based on a criterion. Each one of these models, hereinafter sub-models, represents a partial aspect of the computerization process that is intended for development. Each one of these sub-models must be integrated into a single model to achieve the total representation of what is intended to be modeled.

In general, computerization actions establish the need to consider a preliminary process design and the actions to undertake that collect–in different abstract models– system specifications to be developed, if necessary, rationalizing the construction process, facilitating the subsequent maintenance and reuse, which ensures the final quality of the product, its design and verification. From the previous information, it is deduced that the model obtained is the product of a prior concept of the computerization process, which specifies its essential characteristics.

In the research carried out, (Li, Dey & Forlizzi, 2010) suggest that the methodologies for the development of systems must be used depending on the project and the product to be developed. For this reason, the need to use a methodology in relation to the social conditions under which the students and potential customers will develop is assumed. In the case of the process this research addresses, the students will be integrated in the development team.

Considering the aspects already mentioned in this paper, the idea that the model of a computer system is the mediator between the needs that the user and/or customer has and the development team is defended, using different computer symbols that depend on the methodology used. The essential relations in the definition are expressed in Figure 1.

From the previous figure, it can be deduced that the model obtained is the product of the fundamental organization form in computing that is the project (González Hernández, 2013). The actions and the work that computer professionals can carry out are structured in the project. In this development process, team members structure the communication with their peers and master the typical technical language of the profession, the ways in which information is represented and the processes to computerize, as well as the results of computing activities. Therefore, with the concept assumed in this paper, the representations of the different components of a system, as well as the activities to develop, are parts of the model and constitute one of the key products of the computing activity. When assuming the model as the mediator between the customer and the development team, it is also necessary to emphasize that modeling can be understood as the model development process. It is necessary to recognize that, in modeling, development methodologies and the modeling languages play a key role.

Modeling languages–at its highest expression: UML–constitute a computing formalization of the construction of models without ambiguity. These languages arise for development processes in which the development teams are large and the projects that use them are considered complex. In these projects, the exhaustive documentation of the project and of the resulting system must be done with a single and understandable language for all members of the project. These aspects must be considered for the training of computer professionals, since all the roles

ISNN 2223-2516

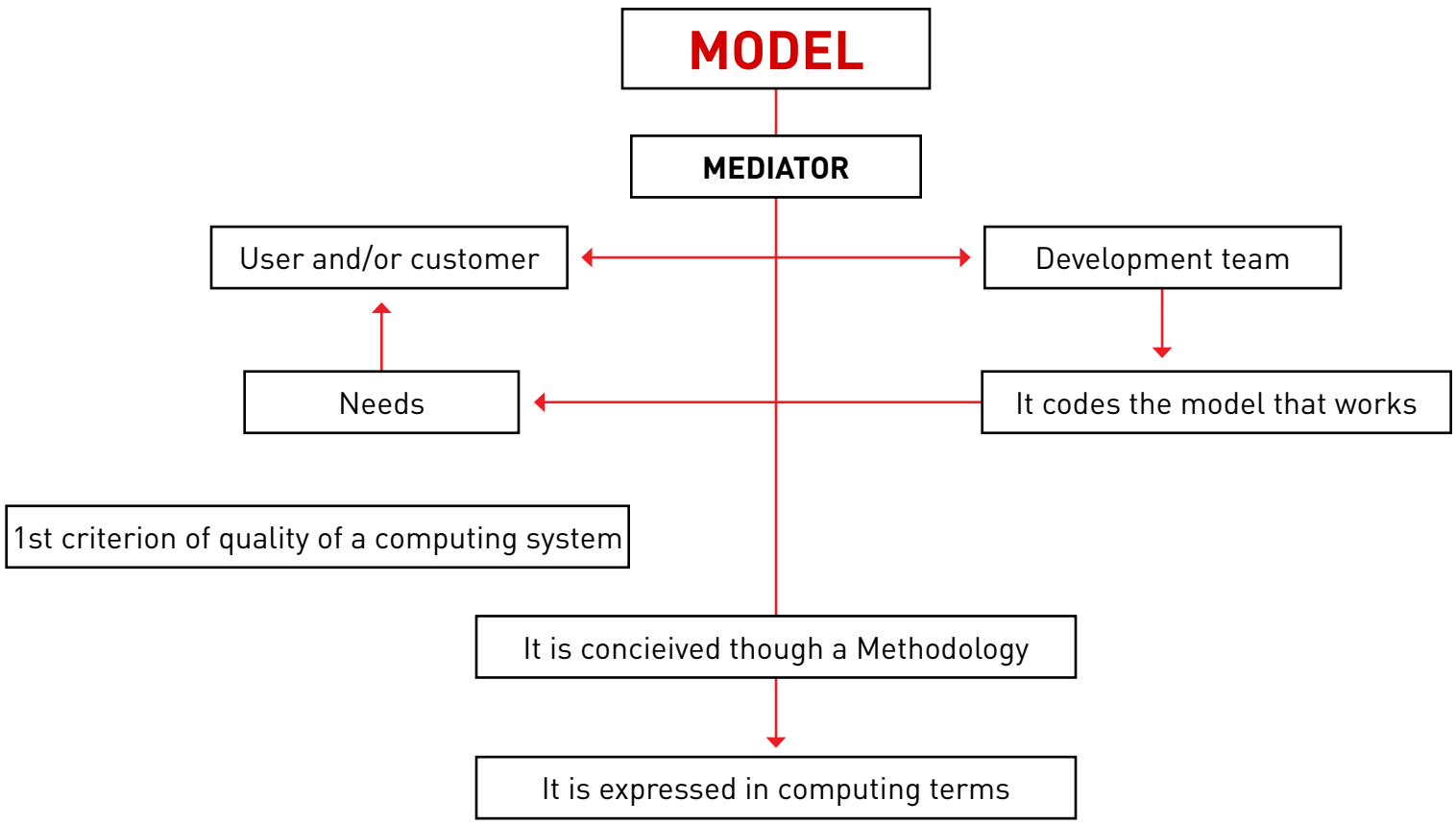

Figure 1: *Diagram of the position of models in the computing activity. From "La habilidad modelar multimedia en los procesos formativos de los Joven Club", by J. Segura Montero & W. González Hernández, 2015, Didasc@alia: Didáctica y Educación, 6(2) (p.31), Copyright 2015 by Centro de Estudios de Didáctica Universitaria de Las Tunas, Cuba.*

they can carry out in a software process are related in one way or another to modeling. Thus, a line in the training of these professionals is development methodologies that incorporate this type of languages. On the other hand, other methodologies for different projects have been introduced in the last few years. There are software projects, especially for small and medium sized computing companies, in which these so-called traditional methodologies create difficulties. These methodologies oriented towards small projects, of fast variation of the initial conditions–called requirements in the computing activity–and of few members are part of the agile methodologies. However, modeling is also necessary in these methodologies.

Even when graphical representations are not built in agile methodologies for the representation of the product, it is important to emphasize that models are obtained. If it is assumed that models are the abstract representation, the result of the computing activity and the mediator between the development team and the customer; then it is fair to consider the description of the products in these methodologies as a model. From these considerations, it is assumed that user stories are the model of the product and the delivery plan, iterations and tests may well be the models of the processes where the activity of obtaining the product is represented.

In these development processes, there is a first stage to establish requirements, where system functionalities and the conditions in which it must operate are determined. These early stages in the development of the system are abstractions that can be written in two different ways: system use cases or user stories. Being consistent with the definition of model assumed in this paper, it is considered that these two mechanisms that may result from the process of establishing

ISNN 2223-2516

requirements are models of projective nature, because they anticipate reality and project the new system to be developed.

The same happens in the case of the need for information storage once the execution of the product finishes. In this case, it is important to start with the application domain, as in the case of functionalities modeling. From the application command, the class diagram follows as a means to represent the internal structure of the system through its structural components, if an object-oriented model is used. From this model, the entity-relation model follows, as a relational representation of the information to be stored through computing transformation means. These transformations include symbolic processes among which operations on sets and other mathematical and computer operations are included. From this database modeling, the subsequent codification is performed in the corresponding language. In this way, other modeling actions are included when transformation actions of this model are made to obtain another one. Databases play an essential role in the data storage for the informational representations of natural processes (Mungall et al., 2015).

Within this modeling knowledge system, it is important to consider the symbol system for each methodology and the documentation this generates to express the processes and structures. Symbology developed in the historical process of the development of computing as a science expresses the essential idea of eliminating ambiguities in representations, for all the members of the project to understand it. However, not only the mastery of symbology for each project ensures the correct representation of the process or structure, but also the relations between them. Each symbol and its relation with the others express ideas that must be represented without ambiguity. The right positioning of symbols depending on the ideas represented is essential for the other members of the project to understand it.

One of the most comprehensive stages for the computer professional's training in the development process of a system is focused on the estimation process of a project. In this process, different abstractions of the system to develop are applied, depending on the estimation model selected. If it is estimated using function points, then the system functionalities written in one of the three models known are considered fundamental. Later, a set of transformation actions of these models is followed to obtain the results required.

Hence, we conclude that modeling is an important activity within the project by pointing out the guidelines of the system development. However, systems are not the only result of the computing activity. When dealing with the computerization process of an organization, information flows play an essential role. In this computerization project, the access levels, the roles of the people who can access information, as well as the routes of information traffic must be represented. The representations of the network along which the information must travel, the points of traffic, as well as the nodes for information exchange are required for the communication in this type of project. This representation also has its special signs and symbols that the computer professional must command to express the essential ideas of the process.

Following this order of ideas, it is possible to find modeling processes in other less popular branches of computing, though with large applications in current technologies. One of them is Artificial Intelligence, by trying to provide the necessary tools and techniques for the modeling of human thought. To mention two examples, descriptive programming is based on the statement of the rules and the facts and from these the solution of the problem is obtained. On the other hand, case-based reasoning systems are just abstractions of the inductive processes that occur in human thought.
From the considerations about the role of models in computing activities, two directions may be regarded: (a) as a starting point and (b) as a result. In the first case, a set of required processes is carried out based on a model, and in the second case, when the actions system of the project members is aimed at obtaining the model. This role of models in the computing activity has a projective nature oriented towards the computerization of organizational processes. Another essential characteristic of computer models is their comprehensive nature. Depending on the project to carry out, several models are integrated into one

ISNN 2223-2516

to represent all the components of the process or structure. As a result of the modeling process in a computer project, two types of models may be obtained, as it is shown in Figure 2:

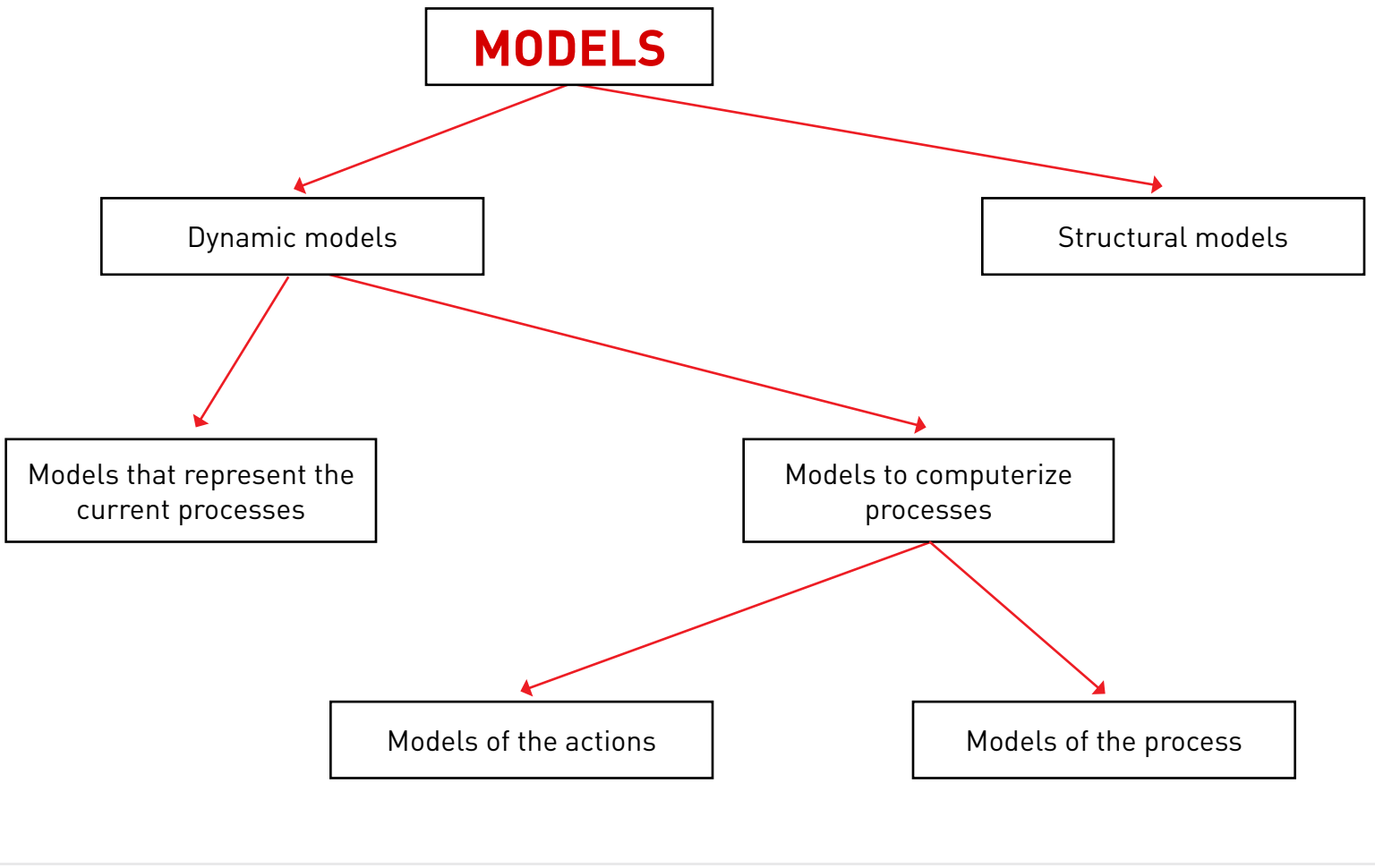

Figure 2: *Diagram of the models obtained in the computing activity. Preparation of the author.*

Each model will be explained below. The structural model does not represent processes but structures of the system or the result of the computerization process. Examples of them can be found in the entity-relationship model of a database or in the model of a network's components.

Dynamic models represent a set of actions comprising any process. First, models that represent current processes stand out as those that include the actions performed before computerizing. An example is in the business model in the RUP methodology. A process model represents the set of actions to be executed to change the current state to the desired state. An example of these models is the one where computerization process actions are represented and another one is the system actions such as CRC cards in an agile methodology.

Many of these actions can converge in models for a total representation of the process, as is the case of a system design using the RUP methodology with UML. In these comprehensive models, several sub-models must appear systemically, so that a more complete description is achieved. In summary, Figure 3 represents the essential aspects mentioned in this section.

This computerization of organizational processes can occur in several activities within the scientific disciplines of computing as a science and may be integrated or not in the solution of the problem.

ISNN 2223-2516

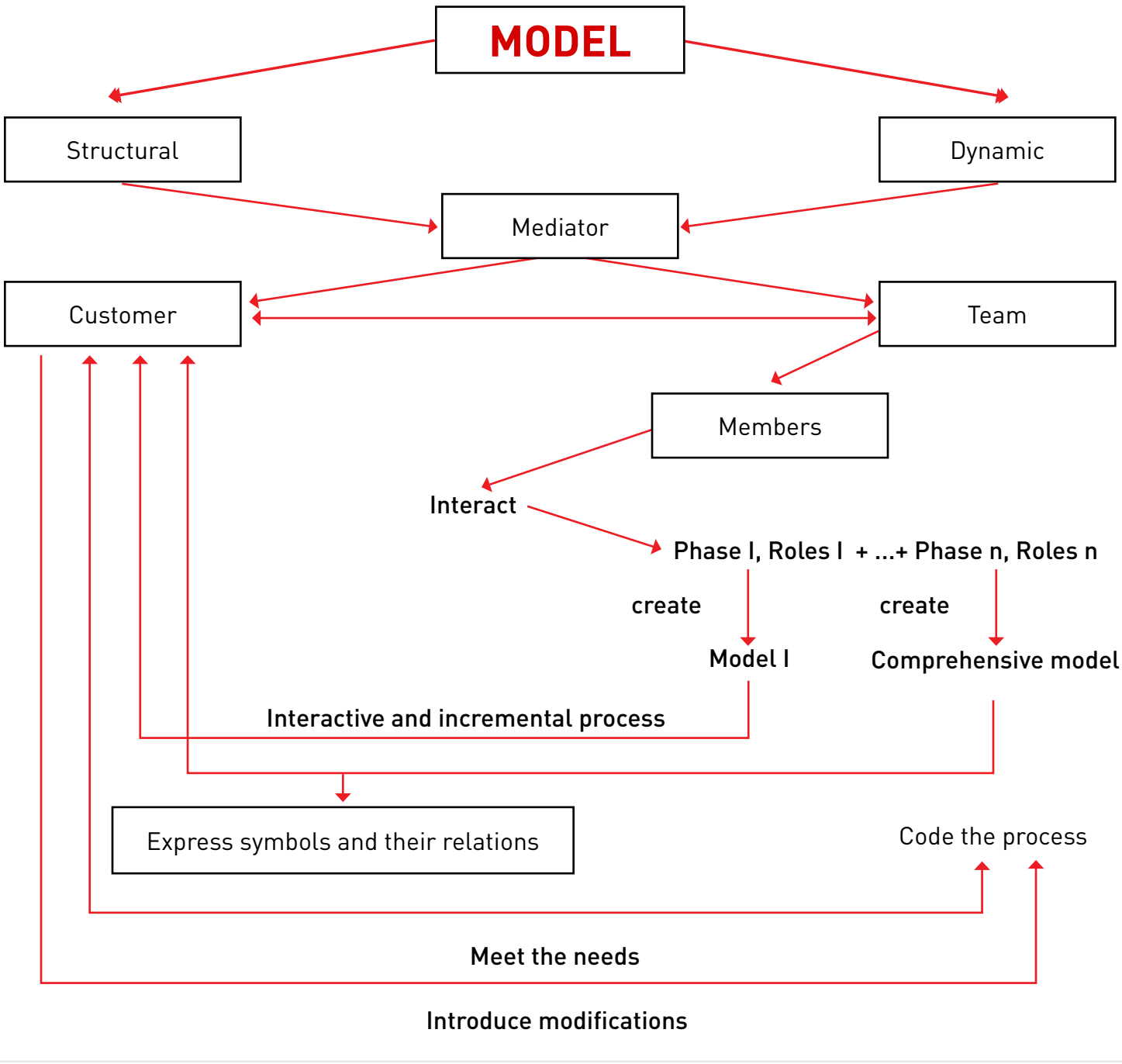

Figure 3: *Model structure in the computing activity. Preparation of the author.*

Considerations outlined up to now about computer models constitute part of the knowledge system that must be internalized by the student who is trained as a computer professional. These scientific disciplines reflect in the curriculum of the computer professional and contribute to their training to interact in increasingly multidisciplinary and complex computerization processes.

MODELING IN THE CURRICULUM OF THE COMPUTER PROFESSIONAL

Assuming curriculum "... is a comprehensive system that expresses relations of subordination and coordination among its organizational levels, the graduate profile is the guiding document that expresses the educational objective of the major" (Sahinoglu, 2016, p. 153). It is necessary to point out if modeling is within the graduate student profile of these professionals. An analysis of this type leads to a comparative analysis of curricula in prestigious universities in computing training in different countries.

A study conducted on the computing curriculum in different universities (Mulder, Lemmen, & van Veen, 2002) concludes that the essential contents are in algorithms and system development, computer

ISNN 2223-2516

systems and their architecture, computer formalities, and the representation and modeling of information. As already suggested in the first section, in these computing contents, fundamental actions result in a model.

At the University of Burgundy in France, programming, database and systems engineering subjects, among others that lead to computerization processes modeling have remained since 2007. At the University of Dublin, the University of New Delhi, the MIT, among others, subjects like Software Engineering, Artificial Intelligence, Computer Architecture, Educational Computing, Databases, and Programming have remained. These subjects of the curriculum design in the training of the computer professional imply modeling as one of the fundamental activities. Therefore, it is easy to understand that there is a total agreement between computing, its fundamental project organization, models as one of the fundamental results, modeling as actions to obtain a model and their presence as courses in the curriculum.

The statement of the previous paragraph allows considering that modeling is one of the forms of action of the computer professional with a great impact on its professional activities. It is then inferred that computer professionals create projective models as part of their activity for the computerization of the different organizational processes.

Once modeling is understood as part of the profile of the computer professional and one of the most important activities as future computer professionals in organizations, it is necessary to clarify whether it is a skill or a competency. For some authors (Segura Montero & González Hernández, 2015), modeling is a skill. This conceptualization of modeling as a skill has a flaw that, in the view of this author, restricts its structure for the context of this paper. The flaw of this concept lies fundamentally on the contextualization of modeling in a basic training curriculum, which has substantial differences regarding the professional training curriculum in computing.

For other authors (Barrera Jiménez, Barrera Jiménez, & Hernández Amaro, 2015), generalized skills that have generalized actions and operations with an also general knowledge as an essential characteristic are structured. In this case, in this conception, there is a clear cognitive conception because it only includes knowledge and skills, leaving aside other subjective processes involved in the professional activity like imagination, self-regulation, the senses, and experiences that are not integrated in its conception.

Computing professionals must incorporate modeling in their performance as one of the essential requirements of planning and projecting of their activity. From this projection and planning of their activity as future professionals, a set of essential qualities is deduced, such as decision-making in the projection of computerization processes, honesty, and responsibility to assume one's own mistakes and the group's in filtering systems, among others. This implies that modeling exceeds skills as a systemic structure of the personality.

For several authors (Núñez Pérez, 2013), competencies may be defined as learning or complex outcomes that are part of cognitive, procedural and attitudinal aspects, skills, characteristics of the personality and values that– put into practice in a certain context–will have a positive impact on the results of the activity performed.

For another author (D´Angelo Hernández, 2015), competencies are *psychological integrational structures at intermediate level*, which complement or articulate the functions of the main structures of the personality in situations that demand a given performance as the expression of a person's behavior in a social context and in a specific action environment. Taking the definition of the last author, his three competency analysis dimensions are regarded.

The necessary differentiation of the modeling processes that each student performs and that is structured individually enables the expression of experiences, knowledge and skills related to modeling. This allows them to integrate their projects with others' and to learn from their peers. Considering these elements, this paper shows that modeling is a competency for the computer engineer and its structure will be analyzed below.

ISNN 2223-2516

We will try to show that modeling is a competency of the computer engineer, omitting the system of knowledge that has already been addressed in the first section of this paper, also addressing the skills within this competency. Two essential model-related skills for a computer professional can be mentioned. One is the representation of processes and structures using the appropriate symbol system that enable structuring the representations that have been analyzed so far. The second is to understand the models made by others in computerization processes and carry them out according to the conception of the people responsible for these processes: analysts and designers. Therefore, it is important to take up the project again, now as an articulating axis of the training processes for computer engineers. The integration of these two skills within a harmonious system together with the knowledge about modeling will allow students to complete a project with success.

Project-based learning allows the students to apply the appropriate contents and quickly analyze the relevance of these for their professional training. In a project environment, the students carry out the roles of their future professional performance and articulate performance methods within their life project for the future profession they will work on. This process has a special relevance in the youth because it is one of the fundamental characteristics of the social development situation in which they find themselves.

The learning of values for a project is also important to consider in this paper. In this order of ideas, the values of the profession such as responsibility, commitment, honesty and humility (de Castro & de Sá, 2002) are shaped. Responsibility is one of the most important values in the computer professional due to the characteristics of the activity, as well as the social impact that technologies have in society. In any of the roles professionals must perform, they must be responsible for their actions and be a real regulating axis of their actions, since others' actions in the project also depend on them. Modeling is one of the activities with the highest potential for the development of students' responsibility, given the role it has in the projection of the future system to be developed and its marked mediator character.

In the development of the project, honesty with colleagues in the modeling process is essential to understand the social relations that are established. In these relations, the project, the positioning of the company, and the trust among the members play a fundamental role.

Another of the fundamental values in training is industriousness. In a project-based educational environment, the students can understand the social significance of the work they perform. It raises their awareness in terms of team work and produces satisfaction for the work performed, which also contributes to regulating their behavior to achieve the project's success. In addition, a positive attitude toward the project and the actions that must be performed in it is achieved. In this order of ideas, the clarification of the roles they play in a project and the balanced social significance among them contributes to the love of work.

The teaching of these values must be based on a set of experiences that makes people understand their responsibility with the other members of the project and society for the result of the project they will obtain. This process must be focused on the importance of the role they have and the result of the activity for the organization. All roles are important in a project environment. The detailed explanation of errors has an essential role, as well as the opportunity to express and correct them without representing an opportunity for punishment, through dialog, confrontation, and constant and constructive discussions. It is also important for modeling to assign tasks of greater complexity that involve all members of the project to explain the specificities of their model.

The integration of these values–in the configurations that are structured in the computing activity–into the life project actually incorporate into the regulating potential of the professional's personality. Consequently, these values will become part of a person's subjectivity and they will not produce formalities.

Like values, the remaining components of the modeling competency are regarded. The integration of student - reality - teaching ensures that the students' work

ISNN 2223-2516

acquires a social character given the implication of the project results on organizations and the system of relations to develop with the rest of the group in problem solving. The previous idea leads to the analysis of the situation and a reflexive position before criticism and questioning.

However, there is still a pending issue regarding the competent performance of the professional addressed in this article. The structures that the competency has are organized in the professional's performance and are structured depending on the activity to perform. This means that structural components that intertwine depending on the type of model that the computing activity requires intervene in modeling. The experiences with the values and the metacognitive reflections guide and structure the interrelationship of the individual with the model that must be performed.

The transformation of reality by the students based on the process of computerization and the selection of the necessary tools to achieve it evidence the active nature of the regulatory function of their personality. The wide variety of tools for the same activity, as well as the selection of methodologies implies that decisions have been made regarding the project and the models involved. This process of transformation must happen in the mental plane, where imagination, which contributes to their development, plays an important role. Thus, the students integrate these participation experiences into activities that generate models while regulating their learning of computer modeling.

From the search of problems in reality, the software's life cycle starts until it concludes with the set-up and maintenance. The problems derived from the individual project motivate students toward solutions and the course contents knowledge generates–in turn– problem situations for the other students. Actions and operations associated with modeling– from the use of the set of symbols and signs associated with this– and the team work with the rest of more experienced peers structure different scenarios for their future professional profile.

In each meeting, based on the prior interaction between student - professor - group, the problem situations are determined for the remaining students. The professor decides the problem situations to present in the meeting based previously on the group's interaction and the development of the students' projects. Considering problematic situations related to modeling and their solutions–from a project perspective–makes it possible to consider the goals, projects, and expectations. In the extent that these problematic situations are structured and concatenated with their practice, it will lead them to reflect on this practice and to improve it. Through this continuous improvement, people grow as professionals, encouraging self-knowledge and self-regulation.

For many authors (Barrera Jiménez et al., 2015; de Castro & de Sá, 2002; Gutiérrez Alea, 2012; Mulder et al., 2002; Sánchez, 2009), skills training in computing occurs when a particular system is taught. In this paper, a systemic conception proposed by several authors (Hernández, Sentí, & Llantada, 2006; Mosquera, 2011) is assumed. In this conception of computing teaching, the process as a comprehensive whole is addressed, in which special attention is paid to the concatenation of the concepts and computing procedures that cannot be formed in a single class, as in the case mentioned in this paper: the modeling competency. Due to the interdisciplinary nature and the complexity of its structure, the training of the computer professional must be addressed from a systemic approach.

Strengthening the system approach in computing teaching means establishing computing content groups from which concatenation and the general guidelines to organize the teaching can be established. The systematic and integrating work proposed ensures a solid training in computing for the students as an important element for the development of the modeling competency.

Applying the system approach involves the analysis of implementation systems because they precede the teaching of programming in the computing preparation of the students. One of the matters emphasized in computing teaching given the importance of this science is information processing. Modeling of information and its flows can be structured from the beginning of computing training and in the different

ISNN 2223-2516

computing disciplines comprised in the professional training of this science.

To understand the position in this paper about the modeling competency, it is important to assume different positions. There are two divergent criteria regarding computing training and the programming and software engineering disciplines. Some argue it is preferable to start with programming teaching and others with software engineering. When computing teaching is structured based on the project as the core element, software engineering reaches prevalence because it is the projection of what is intended to be reached. It is the discipline that provides the symbols and their relations that allow reaching a first representation of the process to computerize. The skills to represent processes and structures and understand representations that were defined above are developed. Thus, the students are prepared based on their learning for their future professional activity.

Considering software engineering must precede programming, it is important to mention that databases must be taught as of the introduction of Software Engineering to determine the entity-relationship model based on the command of the application. Hence, the students internalize the forms of work of each course, but using modeling and the close relation that exists between these computing disciplines. The integration of these two subjects in the project provides the students with the necessary elements to make a first modeling analysis of the application.

Subsequently, when this first model of analysis is refined, object-oriented programming can be taught from the beginning. The dominant paradigm is object-oriented programming and the concepts of class, object, inheritance, and polymorphism can be analyzed. Then, the concepts of algorithm, variable and code based on these classes are addressed, as well as the methods included. This methodological procedure fosters the integration of knowledge as it takes place in a software development process. Following this order of ideas, the final evaluation of the courses must be integrated to assess accurately the objectives of each of the courses and their integration to solve a real project.

When programming is taught, the beginning consists on working from the modeling of the computer system using any of the structural elements of the two large methodology groups: heavy and agile.

In this process, the skill to understand representations is developed. The students analyze the representations made by others, they relate the symbols expressed in their model and then concatenate them for the representation on the mental plane of the process or structure to computerize. Now, the students can structure their actions better by implementing the actions included in the model, another process within their sphere of action as a computer professional.

This analysis places the students again in their role as future professionals. However, the treatment of modeling does not end with the teaching of programming. This is the moment to begin the treatment of algorithms as descriptive models of the actions that computers must perform. This process can be done using a pseudocode that allows an approach to the syntactic and semantic structures of the programming languages, but closer to reality. Then–using analogy–the syntactic and semantic structures of the most appropriate language are introduced for the solution of the problem of the process to computerize.

For this paper, modeling is one of the most systemic and inclusive competencies of the computer professional activity. It represents a central axis in two roles of the computing activity: project designer and analyst. In addition, it has an essential role as a starting point for the process computerization actions. Therefore, modeling training begins in the first year of the major and finishes when the students present their research process as a written memory, where all the models studied in the major are included to describe the project. The development process of this competency corresponds to their work environment from the diverse problems that these professionals must solve in the organizational environment.

CONCLUSIONS

Models are the result of the modeling activity that is developed in computing projects and establish

ISNN 2223-2516

the guidelines to be followed in the computerization process to be achieved. These models are a mediator between the activities of the development team and the customer to meet the needs of the customer.

In the curriculum of the computing professional, modeling plays an essential role and integrates several of the disciplines that are conceived in the training process of this professional. Assuming the training of this competency involves changes in curricular design to improve the computer professionals' performance.
It is shown that modeling is a competency based on the analysis of several current definitions. From them, a structure comprising knowledge, skills, values, and self-regulation processes of learning is assumed. Assuming the competency in this way, each one of the components was analyzed based on modeling to reveal their internal structure.

The modeling competency structure implies assuming the system approach in the teaching of computing for training during the major. Project-based learning is assumed together with this approach as the essential route for this competency training because professionals' performance methods are integrated in it. Assuming a project approach, software engineering and programming teaching are analyzed, emphasizing the first over the second in terms of the teaching sequence.

REFERENCES

Barrera Jiménez, R., Barrera Jiménez, A. D. & Hernández Amaro, L. E. (2015). Algunas consideraciones en torno al desarrollo de habilidades profesionales del ingeniero informático y el rol de la comprensión de texto en la modelación de algoritmos computacionales. *Revista Científica Pedagógica Mendive, 5*(2), 1-5.

D´Angelo Hernández, O. (2015). Formación para el desarrollo de proyectos de vida reflexivos y creativos en los campos social y profesional. *Revista Crecemos Internacional, 5*(2), 34-45.

de Castro, E. B., & de Sá, M. A. D. (July, 2002). *Habilidades, Competências, Valores e Atitudes-Um Perfil Para o Profissional de Computação e Informática.* Paper presented at the Anais do Congresso da Sociedade Brasileira de Computação, Florianópolis, SC, Brasil.

Epstein, D. A., Ping, A., Fogarty, J., & Munson, S. A. (September, 2015). *A lived informatics model of personal informatics.* Paper presented at the Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, Osaka, Japan.

González Hernández, W. G. (2013). Creativity Development in Informatics Teaching Using the Project Focus. *International Journal of Engineering Pedagogy, 3*(1), 22 - 30.

González Hernández, W. G., Sentí, V. E., & Llantada, M. M. (2006). El enfoque de sistema en la enseñanza de la Informática para el desarrollo de la creatividad. *Revista Enseñanza Universitaria, 27*, 7-21.

Gutiérrez Alea, M. (2012). *Una metodología para contribuir al desarrollo de la habilidad resolver problemas en la disciplina Lenguajes y Técnicas de Programación, en estudiantes de la carrera de Licenciatura en Educación, especialidad de Informática.* (Tesis doctoral), Universidad Pedagógica Enrique José Varona, Academia de Ciencias de Cuba.

Jayapandian, C., Chen, C.-H., Dabir, A., Lhatoo, S., Zhang, G.-Q., & Sahoo, S. S. (2014). Domain ontology as conceptual model for big data management: application in biomedical informatics. *Conceptual Modeling, 8824,* 144-157. doi: http://dx.doi.org/10.1007/978-3-319-12206-9_12

Li, I., Dey, A., & Forlizzi, J. (April, 2010). *A stage-based model of personal informatics systems.* Paper presented at the CHI '10 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Atlanta, Georgia, USA.

Mosquera, O. (2011). *El reconocimiento del concepto función en estudiantes de la carrera de Ingeniería Industrial.* (Tesis de maestría), Universidad de Matanzas "Camilo Cienfuegos", Matanzas, Cuba.

Mulder, F., Lemmen, K., & van Veen, M. (July, 2002). *Variety in views of university curriculum schemes for informatics / computing / ICT A comparative assessment of ICF-2000 / CC2001/ Career Space.* Paper presented at the ICTEM 2002 Florianopolis, SC, Brazil.

Mungall, C. J., Washington, N. L., Nguyen-Xuan, J., Condit, C., Smedley, D., Köhler, S., ... Haendel, M. A. (2015). Use of model organism and disease databases to support matchmaking for human disease gene discovery. *Human mutation, 36*(10), 979-984. doi: http://dx.doi.org/10.1002/humu.22857

Núñez Pérez, M. C. S. (2013). El docente en el enfoque por competencias. *Pensamiento, Papeles de Filosofía, 1*(1), 23-34.

Pardo, C., Pino, F. J., Garcia, F., Baldassarre, M. T., & Piattini, M. (2013). From chaos to the systematic harmonization of multiple reference models: A harmonization framework applied in two case studies. *Journal of Systems and Software, 86*(1), 125-143. doi: http://dx.doi.org/10.1016/j.jss.2012.07.072

Ramírez Oyarzo, R. R. (2013). *Estrategia metodológica para el desarrollo de la competencia comunicativa profesional en idioma inglés*

ISNN 2223-2516

en la licenciatura en periodismo (Tesis presentada en opción al Grado Científico de Doctor en Ciencias Pedagógicas.), Universidad Tecnológica Equinoccial del Ecuador.

Sahinoglu, M. (2016). *Cyber-risk informatics: Engineering evaluation with data science*. Hoboken, NJ: John Wiley & Sons.

Sánchez, B. I. (2009). *El concepto de función matemática entre los docentes a través de representaciones sociales.* (Tesis doctoral), Instituto Politécnico Nacional, México, D.F.

Segura Montero, J., & González Hernández, W. (2015). La habilidad modelar multimedia en los procesos formativos de los Joven Club [Multimedia Modeling Ability in the Training Process of Young Computer Club]. *Didascalia: Didáctica y Educación*, *6*(2), 45-56.

Torra Bitlloch, I., & Esteban Moreno, R. M. (2012). Presentación del monográfico: Competencias docentes en la Educación Superior. *REDU. Revista de Docencia Universitaria*, *10*(2), 17-19. doi: http://dx.doi.org/10.4995/redu.2012.6095