INFORMATICS ON FRESH CURRICULUM OF ENGINEERING AND COMPUTER SCIENCE - A SYNTHESIS APPROACH

Clara Amelia de Oliveira[1], Isaias Camilo Boratti[2] and Joao Bosco da Mota Alves[3]

Abstract — *The present paper analyses some pedagogical aspects of Computer Programming Disciplines on Fresh Curriculums of Engineering and Computer Sciences Courses. The discussion is centered on the possibilities of synthesis of several disciplines under the academic title of Inverted Curriculum, as a pedagogical axis. The Object Oriented Computer Modeling Paradigm seams to facilitate the migration to this synthesis approach. This is just aligned with globalization professional needs in modern techno - science carriers. Another aspect to be enhanced is the recent discussion about Visual Programming for Introductory Computer Programming Disciplines. Nowadays, visual programming is usually placed as priority value on those pedagogical environments. Is it quite correct ? or there is another needs to be integrated, with interface aspects, to get harmony on computer modeling introductory teaching? Suggestions are made, where synthesis and theoretical think are considered as a high level value, in terms of pedagogical aspects, applied to Engineering and Computer Science innovative curriculums.*

Index Terms ¾ Introductory Computer Programming Discipline, Inverted Curriculum, Object Oriented Paradigm, Synthesis Approach, Visual Programming.

INTRODUCTION

"Object as abstraction of the real world provide a focus on significant aspects of the problem domain… this perspective has always been important, even it has not been a terrible popular one.." Peter Coad and Edward Youdon [1]

The present paper analyses some pedagogical aspects related with the presence of Computer Programming Disciplines on Fresh Curriculums of Engineering and Computer Sciences Courses. Historical experience on university teaching process indicates that computer programming disciplines contents evolutes from syntax and logical structures to solve practical cases, problems, or even immediate tasks, towards modeling complex tasks, named yet, project. So, in this sense, a project is a kind of more complex problem, in terms of thematic domain field, or generalization level of thematic approach. Yourdon at alli [2], suggest "it should be a genuinely practical approach which really can be used to develop applications in the real world". The term real world is interpreted as a complex vision approach.

Some elements that are supposed to be an important income, to built, nowadays desired open multi focused educational environments are analyzed in the present work. It is pre recognized the presence of computer programming as an obvious support tool, on the techno – sciences courses curriculum. But, here, what is important to perceive is that there are several ways to implement this potent tool. The best way is the one, that considers a tool submission, to pedagogical values respecting different kinds of contexts. This ensure a right way walking, toward increasing complexity, looking to future evolution. In this sense, the contribution from Computer Science, specifically from Object Oriented paradigm, added to the Engineering Project Oriented axis, promotes a welcome synthesis approach, in terms of innovative curriculum proposals.

For this discussion, they are presented three items. To begin, an historical perspective from curriculum trends, helps to understand the dynamic process occurring on education. Following, it emerges a *first aspect*, related with Inverted Curriculum pedagogic proposal, associated with inter disciplinary concepts. In this sense, engineering domain, constitute itself, a natural source of high complexity level projects, to be implemented by computer science applied domain.

A *Second step* looks to a synthesis experience involving Object Oriented paradigm, from Computer Sciences domain, and, Project Oriented perspective, from Engineering domain. This perspective contributes open multi focus teaching environment creation. A practical example on the Meteorological Study Project will be presented.

A *third, and last step* of discussion is about the Visual, or graphics, Programming Orientation, associated with visual computer environments, as the Microsoft Windows. The most recent computer environments have this kind of interface model. It is important to define a set of pedagogic aspects, related with Visual Programming to preserve pedagogic goals, broad sense. What kind of teaching/learning context is focused? System development, or professional training ? What kind of customers/student profile is focused ? Advanced or junior skill? What kind of problem level is proposed to solve? Complex system/ project or concrete aspects/ problem? Those considerations help to built some kind of work hierarchy, avoiding a lack of values, over a natural value hierarchy to support pedagogic choices.

[1] Clara Amelia de Oliveira, Federal University of Santa Catarina, Dept. of Informatics, INE-CTC-UFSC, Cx. P.476, Fpolis, SC, Brazil , clara@inf.ufsc.br
[2] Isaias Camilo Boratti, Federal University of Santa Catarina, INE-CTC-UFSC, Cx.P.476, Fpolis, SC, Brazil, isaias@inf.ufsc.br
[3] Joao Bosco da Mota Alves, Federal University of Santa Catarina, INE-CTC-UFSC, Cx.P.476, Fpolis, SC, Brazil, jbosco@inf.ufsc.br

HISTORICAL EVOLUTION ON CURRICULUM TRENDS

There are several kinds of curriculum orientation, specially looking to the historical evolution, on those referenced engineering and techno – science carriers. In this direction, curricular organization and orientation is close related with a epistemological paradigm, deep sense. It will be presented, here, a resume from each orientation, under an historical perspective, till the recent Inverted Curriculum [3] that is occurring, anyway, under different proposals, all over the world [4],[5]. This last curricular proposal is, recently, appearing, as an international tendency, also enhanced by ABET – Accreditation Board for Engineering and Technology from USA. At first, it search for common attributes involving several curriculum techno carriers. The present proposal, enough generic to be applied on all engineering carriers, involves attributes that are specially abstracts, and associated with complex pedagogical context needs. In practice, this is traduced by the engineering project central pedagogic axis. Those projects constitute, themselves, into an interesting source of complexity, specially if it is desired to integrate them, in a wide range of practical, and theoretical intrinsic aspects. And, it can occur a natural inclusion of biological, social, climatic, and other broad aspects.

Disciplinary Orientation

Traditional curriculums are, usually, Disciplinary Oriented. There is to say, they normally include a range of sequential disciplines, specially on fresh curriculum period, just to cover several fundamental domains. They concern to contents as math, physics, chemistry, electronic, computer science programming, computational math, and so on. Those disciplines are considered as a kind of income to be integrated with middle and professional academic period. They follow a linear sequence from part categories to whole categories of knowledge contents. Those last categories are normally more close related with engineering project under complex approach modeling.

For example, a part knowledge category could be some mathematical methods contents, implemented through computer programming, potentially applied on engineering projects. A discipline of computer numerical methods is usually placed on sophomore undergraduate curriculum disciplines, as the case of Santa Catarina University engineering courses.

A whole knowledge category, could be just a practical engineering project, involving all the parts and also socio – economic- ethical aspects. An Engineering Project discipline is, usually, placed on more advanced periods, on professional curriculum. Some pedagogical aspects emerge from this disciplinary paradigm.

The most important pedagogic aspect, embedded on this paradigm, is, that, student receive some income, constituted by basic contents, looking to future practical application, in his carrier specific field. There is to say, on the curricular period of professional level disciplines, it is supposed that he is able to make inferences using those previous disciplines contents. This has been called, pos disciplinary process. This curricular modality was very up to date till the years sixties. And, of course, depending on complex combination of many other aspects, as resources, labs, relation teacher/ students and so on, it can be successful till today. This paradigm is still popular on traditional techno – sciences curriculums.

Multi – disciplinary Orientation

Later, on the years seventies, it appears a tendency of block disciplines creation- the multi – disciplinary tendency. It does not change so much from the disciplinary orientation but it increase a kind of sensibility towards relationship between several disciplines, normally under responsibility of different teachers. So, a relation between parts are a little more enhanced, aggregating disciplines in a block structure. This inter dependence stays walking in the sequential direction, from part to whole categories, in terms of curriculum perspective. But, it is already perceived, more clear, that, somehow, some pieces of basic contents are lost in the middle of the way. This occurred with both disciplinary and multi – disciplinary orientation. It exists an effort towards bigger categories of knowledge aggregation. The multi disciplinary curricular structure tendency is a kind of horizontal aggregation. It is direct related with increasing complexity projects towards inter – disciplinary times.

Inter – disciplinary orientation, or even, Inverted Curriculum Orientation

Inter disciplinary Orientation and Inverted Curriculum can be considered synonymous? Yes, they are, because under complexity parameters, claimed from interdisciplinary abstractions models, there is an *indissoluble interdependence* of the categories of knowledge. And this is, in practice, achieved by an inversion, just putting, before, the complex idea, and, later, the details of its implementation. The double expression above was already applied in 1997 by A. Rapoport [7] on the concept of world order and it is, also, welcome on the present pedagogical analysis. Inter disciplinary curricular trend is more recent and it has been practiced, also, under isolated disciplines context. But, of course, its potentiality increase a lot, if it is supported by a global curricular proposal, putting all together, under a course Inverted Curriculum. Deep sense, it is an extremely simple idea. It is inverted the direction, yet, from generic – complex categories of knowledge, synonymous of engineering project, on those carries courses, to part categories of knowledge, involving from math till informatics practices, for example. The thematic of study can look, the more close as possible, to the real world

complexity. At the first time, you can abstract complexity in the implementation aspects, but at same time, you can discuss with your students about essential aspects, demanded for a typical multi focused discussion.

At modernity times, it is important to face global aspects, literal sense. To a first look, students emit ecological, ethic and socio political opinions, not necessary implemented, for each one of the discussed aspects. Depending on curricular orientation, this central project can stay alive during all the course period, focusing different aspects, at different times. This perspective, in engineering education, is just coincident with Object Oriented modeling paradigm, an integrative perspective to modeling complex systems in computer science domain. This paradigm was derived on the years sixties to solve complex systems, by professor Kristen Nygaard, from Department of Informatics from Oslo University. On the years nineteen, Bertrand Meyer paid homage to him describing Object Oriented principles as 'a computational wave that born in the ice-blue waters of the festooned Norwegian coast',[8].

OBJECT ORIENTED PARADIGM ON FRESH CURRICULUM DISCIPLINES – A SYNTHESIS APPROACH

"The substance of the system is organized as objects, building the system's components". Kristen Nygaard, Dept. of Informatics, University of Oslo [9]

The present proposal point out, the perspective from a discipline of introduction to computer science in the first semester of university, specifically on engineering or computer sciences courses. Normally, this discipline contents don't specify anything, direct related with engineering applied domain aspects. Usually, there is no curricular exigency, because such discipline content is related with specific aspects of computer science programming, as logical structures, data structures, and other computational specificities. But, of course, computer science programming teaching is, always, oriented to some kind of computer paradigm, as programming representational model.

After the, called, Imperative programming times, popular on the years sixties, it came the age of Structured paradigm. In engineering courses, this paradigm is, still popular today. The present proposal suggests the migration towards Object Oriented paradigm, also for Engineering courses, but this is not all. The most important aspect is to integrate this one with a Project Orientation axis idea, synonymous of Inverted Curriculum orientation tendency. And this, complementary way, is facilitated by a computer science paradigm implementation- the Object Oriented one. There is to say, to synchronize the computer paradigm, with a enough complex problem praxis - the Engineering Project under high level of complexity approach.

Short way, the Project Oriented methodology suggest the creation of a set of project versions, from easy to complex, in terms of details and specific/specialized knowledge.

The Project Oriented perspective was applied by Oliveira [6], under both Structured and Object Oriented computational paradigms. The following examples, from this pedagogical experience occurred, first way, at Sanitary and Environmental Engineering course, from Santa Catarina University. The chosen project example is from Meteorological Study of Weather thematic.

Figure 1 shows a class diagram from a first project version, named version 0. There is a unique category/ class, titled, Meteorological Work Station. Here, it is focused, just only one attribute, the Existence_Time from the Meteorological Work Station, and, also just only one method, Atualize_Existence_Time. It is too easy to do this operation? But, remember that the maximum goal, at this moment, is to arrive to a first contact with the Object Oriented terminology, and with the computational environment. There is enough theoretical details, for this moment. Modeling aspects will increase, as times goes by.

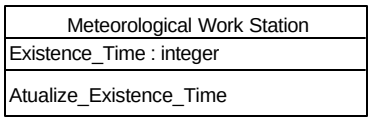

FIGURE. 1
CLASS DIAGRAM FROM METEOROLOGICAL STUDY PROJECT - VERSION 0

Figure 2 shows a class diagram from the project next step, or its version one. It appears, yet a specialized class of Meteorological Station, sub class from a generic Work Station one. The goal here is, to present, new Object Oriented concepts, associated with methods, applied over new data structures, as the array one.

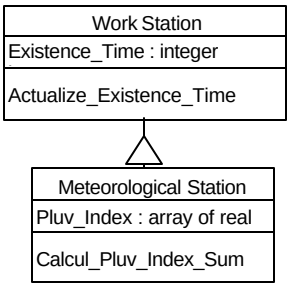

FIGURE. 2
CLASS DIAGRAM FROM METEOROLOGICAL STUDY PROJECT - VERSION 1

It is remarkable that each version, from the unique project, is considered as an independent example. So, the project is rewritten many times, simultaneous with increasing student skill with each version acquired experience. Looking to the set of versions of the project, it is possible to perceive, classes structure, as a template place, for future expansion,

and specification, in terms of technical aspects implementation. For example, a basic math aspect, just a very easy application, plays a role of place reservation for future math developments. It is not important to present all together on the initial versions of the project. This idea promotes the evolutionary vision and it is very interesting to unify complexity with first aid tools, in pedagogic environments.

Figure 3 shows a class diagram from a more complex version from the project. Depending on pre defined model, the Work Station class can include, a Department of Human Resources - DHR. The DHR class can, also, include other classes, referring, for example, to Employers aspects manipulation, and so on. Technical departments can be represented by classes as Pluviometric , Thermometric, Higrometric and Barometric ones. The organizational structure of the project want to traduce a real world abstraction. The math class is very special one. It is an environment that opens infinite possibilities of inclusion of statistical and numerical computational methods. Under this point of view, mathematical methods are only functions applied over some kind of data structure, as the arrays example. Those specific math environment are developed in such independent way, that they can be easy reused by other systems that have the same input / output desired.

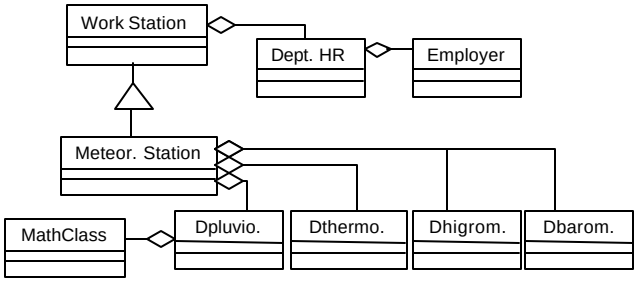

FIGURE. 3
CLASS DIAGRAM FROM METEOROLOGICAL STUDY PROJECT - VERSION N

This pedagogic proposal, Thematic Oriented or Project Oriented is independent from computer paradigm or even computer implementation system. But, if it is desired to develop a complex model, and to implement this model in a computer system, the suggestion of Object Oriented is the one, because it touch the complex problem with compatible resources. And, a practical simple reason is that all today's softwares have included the Object Oriented implementation. Considering the proposal of teach an Introductory Computer Science discipline to Engineering fresh curriculum, and adding the idea of generic Project Oriented axis, it comes, as consequence the synthesis approach from several basic disciplines.

VISUAL PROGRAMMING TEACHING, ON DIFFERENT CONTEXTS – A PEDAGOGIC INFERENCE

Above, it was presented a synthesis between Computer Paradigm and Project Oriented axis, as a innovative pedagogic approach, for fresh curriculum. After this, it comes, naturally the people-machine interface aspect, to be integrated with the other two roots, harmonic way. Visual interface is a term which defines a friendly interface, where the customer acts by manipulating input tools, as mouse one. Visual Programming is a double term, related with the interface modeling, from nowadays times. The most important change during interface development historical period is that, today, customer is less specialized and more active in terms of program control, because computer is a popular engine. The new computer interface conception is, that, machine is programmed, to wait from customers orders. It is remarkable the change on customer profile. Sometimes ago, customer and system developer role, was played by an unique actor. Usually, costumer role was played by engineering carrier people. Today, this is no more true. The roles are separated and customer can be someone else, not necessary, computer specialized. In terms of systems development, interface development and programming is, an even more deep, specialized computer modeling domain. The central point, in a pedagogic context, is the clear definition of pedagogic goals. Visual Programming is a step of the system development. A system / project has stages in accordance with its complexity. If the pedagogical goal is to treat with complexity aspects, interface aspects must wait for its specific moment. Abstracting visual aspects, attention focus turns towards modeling aspects. What is wanted in terms of pedagogic decisions? What is a pedagogic need ? They are present two needs: project needs, to obtain useful answers, and also product needs, in terms of visible project results. Anyway, it seams, somehow, out of discussion to reflect about advantages and disadvantages of those input / output visual aspects. But, the world arrives to some paradoxes when lack of values in terms of vertical hierarchy are practiced. But this is a complex affirmation because it is nothing negative by itself. It all depends on the application context. People with different skills level, and different autonomies, and consciences, leads with those aspects, defining, at the end, what is good and what is not, depending on self process context.

So, here, the effort is to define, previously, the Engineering course student specific context, and also, in this scope, the Computer Science course student context. In practice, a pedagogic context of developers is different from a training people context, or even, from a system final customer . In a pedagogic context, different goals / needs, affect interfaces models, different way. Visual programming is based on visual objects like forms, bottoms, boxes, and so on. But, an Object Based model is not necessary, the same as an Object Oriented model, in terms of embedded principles,

in accordance with Cardelli Wegner, apud Booch [10]. Of course, this is extensible to interface models and visual programming. A pedagogic approach, specially the complex project modeling, must take care about this point. What kind of interface is to be presented to students in fresh curriculum in disciplines related with computer tool? The answer, again, is: it depends on the pedagogic context.

Following, it will be presented two different cases of context affecting pedagogic tools choice.

First context example of Visual Interface utilization: Discipline of Introductory Programming and Engineering Project Modeling, under Project Oriented pedagogic approach. In this case, it is enhanced an hierarchy of values that goes from project model comprehension towards programming implementation aspects. As a system developer, student attention must be directed, at first, to project aspects. He must, first of all, understand the problem itself. He learns, also to discuss with project team, and other development pedagogical aspects. He learns something more than isolated technical or instrumental aspects. Next step, he begins to solve problems, applying mathematic, physics, logical and computer programming concepts. At this moment, we can imagine students facing a personal or portable computer, solving a project, already converted in a system. But, for system modeling focus approach, this is wrong. Student is not, at first moment, near from computer tool. Because, in real world, there are, also, planning stage, before execution stage. Development environments need people, creative people. Schemes, class diagrams, and many other support tasks must be ready, before programming aspects dealing. But, fresh curriculum student central point of interest is instrument, specially under Visual Interface. How to lead with this paradox ? Below, there are two approaches to solve this.

A first solution is to use Dos interface, also named text interface, at the course first period. This is interesting, in terms of a hierarchy of values, to be followed, specially under Project Oriented pedagogical proposal. In this case, pedagogic axis is oriented to an unique project thematic presented during all discipline period. After discussion of several versions of the project, basic modeling principles are, yet, clear. At this moment, student has got his skill in theoretical aspects, and, then, is he ready to enter on interface world aspects. A practical example of this approach, related with the Meteorological Study Project, was presented above. This pedagogical approach has been applied, by Oliveira [11], on Santa Catarina University, since eight semesters.

A second solution is to present, earlier, at the course first period, a visual interface modeling and programming, but, forcing the interface to be the most standard as possible, looking to avoid the excessive and attractive appearance of visual details, that causes the by passing of attention towards fragments, lacking the whole problem. In this case, pedagogic axis is oriented to several project thematics presented during a discipline period. This second pedagogic solution has been applied, by Boratti [12], since eight semesters.

Examples of both, textual and graphic interfaces, related with the Meteorological Study Project, are presented on Figure 4. In addition, it is important to register that both pedagogical experiences occurred, at first, on Computer Sciences under-graduation course. The migration from specialized computer science domain to other related applied domains, as all Engineering courses is a natural, and historical, occurrence.

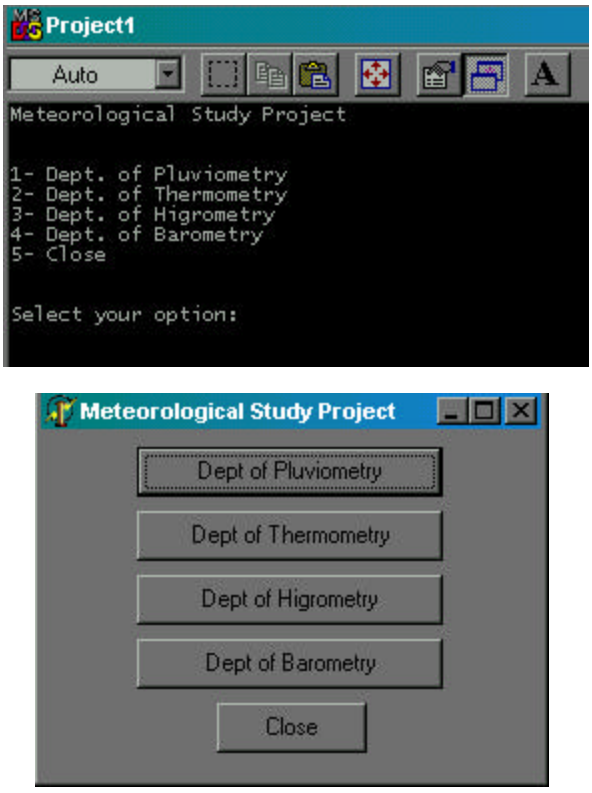

FIGURE. 4
EXAMPLES OF TEXT AND VISUAL INTERFACES ON THE METEREOLOGICAL STUDY PROJECT

Second context example of Visual Interface utilization: Any discipline that need Data Manipulation on Engineering Applied Problems. Here, student is a customer of a computer system. In this case, student don't look to programming aspects, but reacts, as a final customer, looking only to system input/ output aspects. Here, he finds a ready system, built by developers that create the *illusion of simplicity* [13]. They lead with closed software packets.

In this case, interface modelers, are people that have skill on specific aspects, as psychological, and human communication factors. They present specialized interfaces environments, close to real world vision, creating a, already commented, simplicity illusion, for users, as the student user case. In this context, Visual Programming seams to be ideal application, because here, central point is to obtain answers

from the ready system, instead of developing the system/project itself.

CONCLUSIONS

Three aspects related with synthesis pedagogic approach can be enhanced.

On the synthesis pedagogic approach for Engineering Courses fresh curriculum : several disciplines can be treated together under project approach axis. Engineering project is a natural axis which aggregate the other basic disciplines, as computer programming, math, physics, and so on. Synthesis pedagogic approach is enhanced by Project Orientation perspective.

O***n Oriented Paradigm as a contribution to synthesis pedagogic approach:*** this computer paradigm, for knowledge representational modeling, is not only useful on computer domain. It contributes, together with Project Oriented approach for a inter disciplinary pedagogic approach.

***On Visual Programming pedagogic approach*:** It is important to perceive the necessity of integration of Visual Programming aspects with project modeling aspects in accordance with pedagogic values, in open, multi focused educational environments. Visual programming times can cause a lack of value in terms of complex modeling, when theoretical thinking goes to be a secondary aspect, in face of the visual interface attractive appearance. Experimental and instrumental pedagogic aspects are very important but they must be thought carefully to maintain what they have as positive essential value - the final presentation from a high quality software system.

ACKNOWLEDGMENT

The authors want to recognize the relevant aid from docent Leandro Komosinski, educated under Object Oriented paradigm times, by his attention and friendship towards his colleges, by their sides, educated under Structured paradigm, at their migration starting point.

REFERENCES

[1] Coad, P., Yourdon, E., " Object Oriented Analysis ",*Yourdon Press*, USA,1991, pp.5.

[2] Yourdon, E. et alli, "Mainstream Objects – An analysis and Design Approach for Business", Yourdon Press, USA,1995, pp.311.

[3] Meyer, B. ,"Towards an Object Oriented Curriculum", *home page available from http://www.eiffel.com*, 1993, pp.7.

[4] De Oliveira, C. A ., Conte, M.F., Riso, B.G., " Aspects on Teaching/Learning with Object Oriented Programming for Entry Level Courses of Engineering ", *International Conference on Engineering Education – ICEE-98*, Brazil, 1998.

[5] Solen, K. A ., Harb, J.N., "A First– Year Course as a Foundation for engineering Education", *International Conference on Engineering Education-ICEE-98* , Brazil., 1998.

[6] De Oliveira, C. A., "Development and Application of an Integrative Methodology Derived from Object Oriented Paradigm for Entry Level Courses of Engineering and Computer Science", *International Conference on Engineering Education- ICEE-99* , Czech Rep., 1999.

[7] Rapoport, A ., "Conceptions of World Order", *Home Page from the Interdisciplinary Conference on the Evolution of World Order, available from http://www.pgs.ca/woc/rapoport.htm*, Canada, 1997, pp.7.

[8] Meyer, B., "Object Oriented Software Construction", *Prentice Hall*, USA, 1997, pp.v.

[9] Nygaard, K., "Object Oriented Modeling, System Development and Programming", *Lectures at Dina's Summer School, Dept. of Informatics, University of Oslo*, Norwege, 1998, pp.11.

[10] Booch, G., "Object Oriented Design with Applications", *B. Cummings*, USA, 1994, pp.38.

[11] De Oliveira, C. A., Boratti, I..C., Medeiros, S.J., " Uma Proposta para o Ensino da Programacao Visual – Estimulando a Visao Sistemica para alem do Simples Evento", Meeting 51 from Scientific Progress Brazilian Society – SBPC, Brazil, 1999.

[12] Boratti, I.C., "Programação Orientada a Objetos", Visual books, Brazil, 2001, pp.215.

[13] Booch, G., " Object Oriented Design with Applications", *B. Cummings*, USA, 1994, pp.6.