

# **Process modelling analysis: comparison between activity-oriented models, product-oriented models and decision-oriented models**

M. Benabdellatif

*Ecole Militaire Polytechnique, Algeria.*

## **Abstract**

This paper presents a comparison between three classes of process models in management information systems (M I S) development.

The activity-centred models are the most numerous. Based on the “Waterfall” generic model, they include well-known process models defining the development process by finding and executing a plan of actions leading to the solution without mentioning a particular connection to the resulting product. A product is a specification of the solution to develop produced by an engineer.

The product-oriented process models represent the development process through the evolution of the product. Their advantage is to put the product definition in the foreground and push the process definition in the background. But the process is defined as an organized set of transformation activities that do not allow the capturing of the semantic attached to them.

The purpose of the more recent approach gathering process models is to emphasize the decisional aspect of the process. The successive transformations of the product are consequences of human decisions. Such models are semantically more powerful than previous ones because they explain not only how the process proceeds but also why transformations happen. Considering the limitations of the first two classes of process models, we think that the decision-oriented ones are the best adapted to describe the MIS process modelling. The paper shows an instantiation of the “Nature” model to Object-Oriented Modelling and Design (OMT) methodology process.

## 1 Introduction

### 1.1 Definition

Rolland [1] defines a management information system (MIS) “ as a set of:

- Collections of data ...
- Collections of rules that determine the informational functioning ...
- Processes for acquisition, memorization, transformation, retrieval and restitution of information,
- Human and technical resources integrated in a system, co-operating and contributing to its functioning and to its objectives.

This artifact supports a data flow network used for organizing, managing and keeping the company’s activities. It is a tool that allow communication between the company’s partners and increase their efficiency”.

### 1.2 MIS development

There are three main phases:

- **Analysis or Requirements Engineering (RE):** The initial input to the analysis phase is a problem statement that describes the problem to be solved. The output is a conceptual scheme (representation of the system) that captures the essential aspects of the system.
- **Design:** The overall architecture of the system is determined during this phase.
- **Implementation:** This phase leads to the operational MIS.

MIS development needs a methodology. The existing methodologies use two main components:

- Sets of **models** defining concepts that allow the description of the system, *e.g.* E/R model [2]. They let designers reason about the system and validate its representation according to users’ requirements;
- A **way-of-working** showing the process to follow from the problem statement to the conceptual scheme. It prescribes what kind of activities can be performed in the development and when they can be performed.

### 1.3 Problematic

MIS developments suffer from the unbalanced description of their modelling formalisms and the way developers must follow to obtain the expected computerized system.

Up to now, a large amount of research effort has been dedicated to the MIS modelling. This research has resulted in the development of a great number of modelling formalisms adapted to the particularities of the system to be built. Contrarily, current methodologies hardly define the way-of-working designers can or must follow to proceed in the development. Most of the process semantics is missing. This weakness can be explained by the little amount of research

work that has been devoted to the modelling of the development process, relatively to the MIS modelling effort.

This imbalance between models and way-of-working definition has also notable consequences on the CASE tools that more and more often support MIS developments. The CASE tools are efficient in recording, retrieving and manipulating the schemes representing the system undergoing development. But they are almost unable to provide assistance to the developers in deciding and performing the activities transforming these schemes.

Therefore, the methodologies do not describe their way-of-working so clearly, precisely and strictly as they describe their modelling formalisms.

The paper is structured as follows. The next section presents an overview of process modelling approaches. Section 3 presents the selected model to be applied on the OMT way-of-working. The application is shown in section 4.

## **2 Process modelling approaches**

According to [3], the process models, on which methodologies ways-of-working specifications are based, can be classified in three main categories:

- Activity-oriented models,
- Product-oriented models,
- Decision-oriented models.

### **2.1 Activity-oriented models**

Activity-oriented models come from analogy with problem solving, *i.e.* finding and executing a plan of actions leading to the solution. They are sequential in nature and provide a frame for manual management of projects developed in a linear fashion. Such a linear view of the design process is inadequate for methodologies that support backtracking, reuse of previous designs and have to support parallel engineering activities. The first widely used model, the waterfall model [4] falls into this category, along with the spiral model [5] and the fountain model [6] that try to eliminate the well-recognized lack of flexibility of the waterfall model. At the end of the eighteenth century, the research works provide other models such as Osterweil model [7] and HFSP model [8]. Limitations and drawbacks of these models come from their representation of development processes like programs which do not integrate at all the interactive aspects of MIS development.

### **2.2 Product-oriented models**

The product-oriented process models represent the development process through the evolution of the product. A product is a specification of the solution to develop produced by an engineer. They promote a view of development processes which is centred around the notion of development activity but present the advantage to link development activities to their output: the product. The EPM model [9] and the viewpoints model [10] belong to this category.

Therefore, apart from the differences of their ways-of-working formalisms, these two classes of models have a similar expression power.

### 2.3 Decision-oriented models

The main concept of this category of models is “decision”. They consider processes not only as executions of activities but also as decisions leading to them. The successive transformations of the product are looked upon as consequences of decisions. Such models are semantically more powerful than the previous ones because they explain not only how the process proceeds but also why transformations happen. They provide a more complete knowledge about the process. The Potts model [11], the DAIDA model [12] and the NATURE model [13] belong to this category.

## 3 Overview of the NATURE process modelling approach

From now on, the focus is put on the analysis (RE) phase.

RE phase is the most intellectual and creative activity of MIS development process. Its initial input is a problem statement that describes the problem to be solved. The output is a conceptual scheme (representation of the system) that captures the essential aspects of the system.

NATURE is an acronym for Novel Approaches and Theories Underlying Requirements Engineering. Its paradigm is contextual. The idea is as follows: when analysing a problem statement, the analyst is in front of a given situation. To progress in the analysis, he has to take a decision that will modify the part of the problem statement he is reasoning about. The alteration may be added, changed or cancelled information. The result is a new situation and the process loops until reaching the conceptual scheme.

The main concepts of the NATURE process model are *situation*, *decision*, *context*, *action* and *argument*.

A *situation* is a part of the product it makes sense to take a decision on it. It can be a statement about the application domain or a requirement.

A *decision* reflects a choice made at a given point in time of the RE process.

A *context* is the association of a situation and a decision made on it. A context can be “*plan*”, “*choice*” or “*executable*”:

- A plan context is in general a high-level statement or requirement. Generally it is decomposed in other contexts which can be plan, choice or executable.
- A choice context gives the opportunity of exploring different ways according to arguments supporting or objecting the context.
- An executable context is the simplest one. It cannot be decomposed and implements the decision through a deterministic action.

An *action* performs a transformation on the MIS description. It is a materialisation of the decision. Performing an action changes the description and may generate new situations, subjects to new decisions.

*Arguments* are statements supporting or objecting to decisions made in a given situation, *i.e.* support or object to contexts.

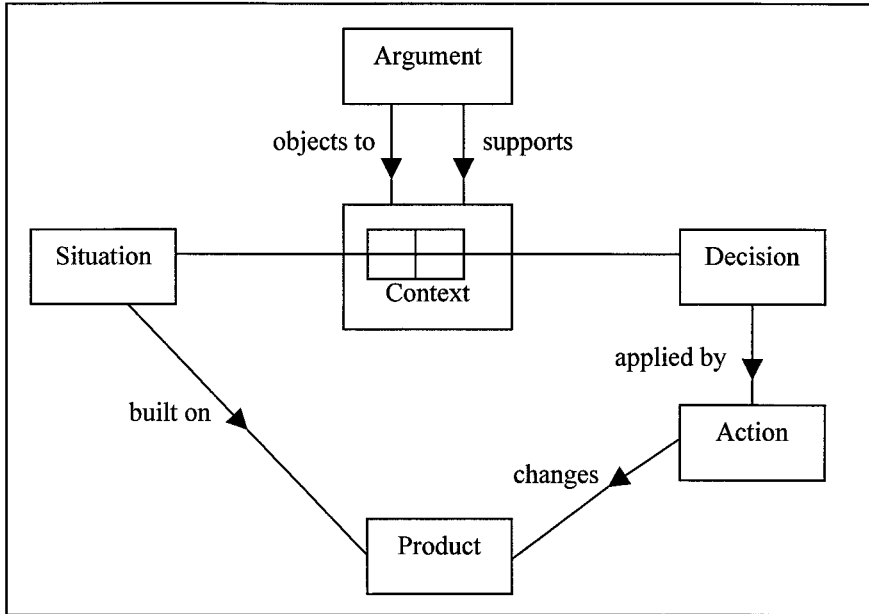


Figure 1: Simplified overview of the NATURE process model.

[14] details the NATURE process model and the reader is invited to refer to it for a complete presentation of these concepts.

#### 4 Application of the NATURE process model to OMT

The OMT methodology [15] supports the entire life cycle, from problem formulation through analysis, design and implementation. It consists in three phases: the analysis, the design and the object design also called implementation phase. The development process is iterative: the development steps are repeated to progressively describe lower levels of details. Each iteration aims at clarifying the features rather than modifying work that has already been done.

The initial input to the analysis phase is a problem statement which describes the problem. The result should be a precise, concise and understandable model of the real world enabling to understand the problem as a preparation for the design.

The analysis model addresses the three aspects of the objects: static structure (object model), sequencing of interactions (dynamic model), and data transformations (functional model). Those three sub models are not equally important in every problem. However, the object model is always very important. The dynamic model is important for problems concerning real time

systems and user interface. The problems containing significant computation have important functional models.

In [15], the analysis phase is made up of five steps. Contextually, it is represented in the following figure.

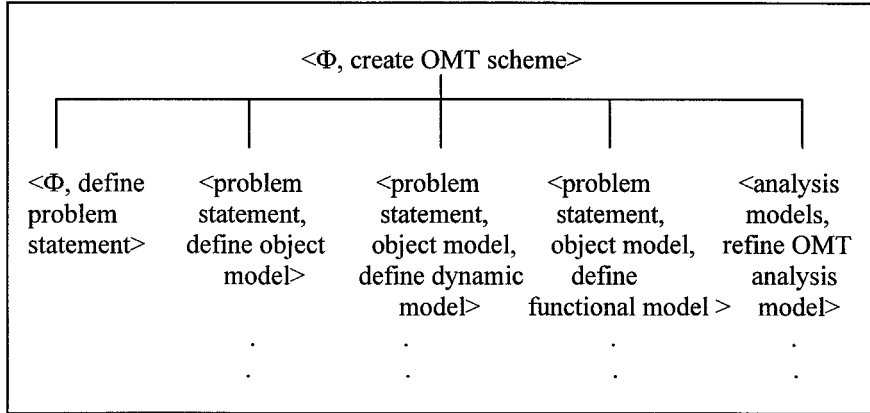


Figure 2: The analysis phase modelled with contexts

In this figure, the *analysis phase* is represented with a plan context. From left to right, we can see that a *problem statement* is created first. Then, the *object model* is defined (“*problem statement*” is the situation of the context defining the object model). Afterwards, the *dynamic* and *functional models* are defined and finally those three analysis models are refined by iterating in the last context *<analysis models, refine OMT analysis model>*. Each context may be decomposed in others contexts which can be plan, choice or executable. So, each step mentioned in the figure 2 will be modelled with a tree. Therefore, *<creating an OMT schema>* will be modelled with a forest composed of four trees. *<Defining a problem statement>* can’t be modelled because it is just a starting for understanding the problem.

#### 4.1 Object modelling

The object model describes the real-world object classes and their relationships. The way-of-working for the object modelling can be decomposed into the following steps [15]:

- Identify objects and classes
- Prepare a data dictionary
- Identify associations (including aggregation) between objects
- Identify attributes of objects and links
- Organise and simplify object classes using inheritance

Benabdellatif [16] details the object model tree which root context is *<problem statement, define object model>*.

Figure 3 presents a hierarchy of contexts describing the object modelling process.

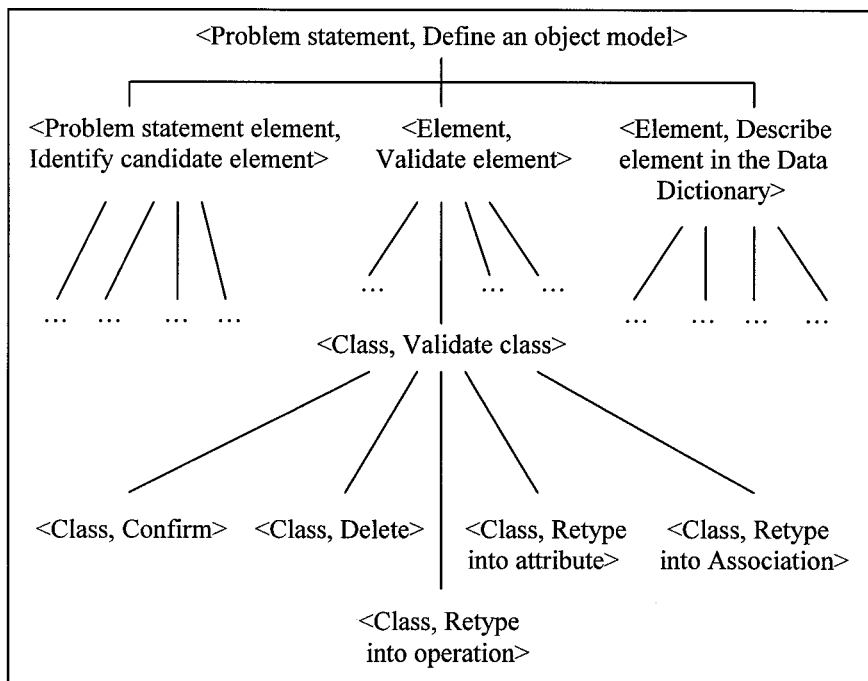


Figure 3: Hierarchy of contexts representing the object modelling process

We can see that the object modelling consists of identification, validation and description of elements. An OMT object model element can be an attribute, an association, an IS-A link or a class. Everyone is a branch of the tree. Each element can be subdivided in sub elements. For example, an OMT association is a choice context because it can be either ordinary, qualified or an aggregation. Figure 3 illustrates the class validation context. It's a choice one because class validation may be confirmation, deletion, or retyping into attribute, into operation, or into association. Each context, *e.g.* <Class, Confirm>, is supported or objected by arguments. For example, if the name of the class refers to a real world phenomenon relevant to the system, it is confirmed. Contrarily, if it expresses the same information than another class, and its name is less descriptive than the other, then it is deleted (and the other is confirmed) because of redundancy.

The same modelling approach, *i.e.* based on contexts, is applied to the definition of dynamic model, functional model and validating/refining the OMT analysis model.

## 5 Conclusion

Applying NATURE process model to the OMT way-of-working provides a flexible processing that guides the RE engineer throughout the analysis phase. We have developed a specification/validation tool at the Military Polytechnic School, and currently we try to apply the NATURE process model to OMT system design & object design phases.

## References

- [1] Rolland, C., Foucault, O. & Benci, G., *Conception des systèmes d'information: La Méthode REMORA*, Editions Eyrolles, 1988.
- [2] Chen, P.P.S., The Entity-Relationship Model: Towards an unified view of data, *ACM Transactions on Database Systems*, Vol. 1, N° 1, 1976.
- [3] Dowson, M., Iteration in the software process, *Proc. 9<sup>th</sup> Int. Conf. On Software Engineering*, 1987.
- [4] Royce, W.W., Managing the development of large software systems, *Proc. IEEE WESCON*, 08/1970.
- [5] Boehm, B. W., A spiral model of software development and enhancement, *IEEE Computer*, Vol. 21, N° 5, 1988.
- [6] Henderson-Sellers, B. & Edwards, J.M., The object-oriented systems life cycle, *Comm. Of the ACM*, 09/1990.
- [7] Osterweil, L., Software processes are software too, *Proc. 9<sup>th</sup> Int. Conf. On Software Engineering, IEEE Computer Society, Washington*, pp. 2-13, 1987.
- [8] Katayama, T., A Hierarchical and Functional software Process Description and its Enaction, *Proc. 11<sup>th</sup> Int. Conf. On software Engineering*, 1989.
- [9] Humphrey, W.S. & Kellner, M.I., Software process Modelling: Principles of Entity Process Models, *Proc. 11<sup>th</sup> Int. Conf. On software Engineering*, 1989.
- [10] Finkelstein, A., Kramer, j. & Goedicke, M., ViewPoint Oriented software Development, *Proc. Conf. "Le Génie Logiciel et ses Applications"*, Toulouse, pp. 337-351, 1990.
- [11] Potts, C., A Generic Model for Representing Design methods, *Proc. 11<sup>th</sup> Int. Conf. On software Engineering*, 1989.
- [12] Jarke, M., Mylopoulos, J., Schmidt, J.W. & Vassiliou, Y., DAIDA – An Environment for Evolving Information systems, *ACM Transactions on Information Systems*, Vol. 10, N° 1, 1992.
- [13] Jarke, M., Rolland, C. & Sutcliffe, A., *The NATURE of Requirements Engineering*, Springer-Verlag, 1996.
- [14] Rolland, C., Modelling the requirements Engineering Process, 3<sup>rd</sup> European-Japanese Seminar on Information Modelling and Knowledge Bases, Budapest, Hungary, 1993.
- [15] Rumbaugh, J., Blaha, M., Premerlani, W. Eddy, F. & Lorezen, W. *Object-Oriented Modelling and design*, Prentice Hall International Editions, 1991.
- [16] Benabdellatif, M. *M I S Process Modelling: Application to OMT Methodology*, Master thesis, Houari Boumediene Sciences & Technologies University (USTHB), Algiers, Algeria, 1998.