

TOWARDS A UNIFIED VIEW OF PETRI NETS AND OBJECT ORIENTED MODELING

Pedro M. Gonzalez del Foyo
Facultad de Engineria Eletrica, University of Oriente, Cuba
pgf@fie.uo.edu.cu

José Reinaldo Silva
Dept. Of Mechatronics, University of São Paulo, Brazil
reinaldo@usp.br

Abstract. *The present work presents a preliminary proposal of unification between classical Petri Nets and objects based on two classes: the activity class and the box class. Eventually one of them can be interpreted as passive or active object, but both can equally contain attributes and methods specified by net formalism. So far, the result is a classical net formalism where each component is an object belonging to a class diagram. We claim that such proposal can be instantiated by several extended net approaches and also preserves some analytical properties of the classic net formalism. Specific examples are showed in workflow analysis.*
Keywords. *object oriented modeling, Petri Nets, workflow analysis*

1. Introduction

Petri have being used with great success in the design and modeling of discrete systems, including discrete control and supervisors, in spite of problems with combinatory explosion. Among the attempts to face this problem figure some proposals to unify object oriented methods for design and analysis of systems with the classical approach for Nets. Those attempts are well classified in the literature as belonging to two classes: including objects inside Petri Nets and including Petri nets inside objects (Lakos et al, 1994 and Bernardinello et al, 1992). There is not a truly unified proposal for any of these approaches so far, and mix of them is still more attractive. Therefore, the synthesis of a net should be made object oriented maintaining the conventional Net formalism and conversely, object nets can be taken using the net formalism to encapsulate behavior directly or in defined methods.

The proposal in this paper is guided by the belief that analytic properties are a key issue of Petri Nets, mainly for large or huge discrete systems. Thus, the introduction of objects is not a good excuse to loose the algebraic description of nets, except if another sound formalism is provided in substitution. On the other hand, we seek for an object definition of the nets and not just a process of object oriented synthesis for nets.

Thus, a real proposal of an object net should preserve composicionality, that is, a net could be defined as an object that is composed of other similar objects which inherits some of its mother class general properties adding other specific aspects. The identity of the objects are provided by the unicity of the set of attributes, which, for its turn, derives from modeling and design needs.

Also, nets (objects) should be identified with individual entities (passive or actives) of the universe of discourse of the modeling. That principle reinforce the restriction we made in the definition of object nets as a proper system (Linger et al, 1979), taking the risk of imposing a too strong restriction to Petri Nets, although it is not so significant to object formalism. In the worst case we would come out with a good synthesis of extended nets for a class of applications such as work flow and planning systems, but that is still a subclass of Petri Nets.

However this is still better than to reduce the general object oriented approach generally applied to discrete systems, software, organizations, and any holistic system. We claim that this can be a contribution to the global discussion of object nets even if very localized.

2. A tentative definition of Elementary Object Petri Nets

As a first approach let us take an introductory definition of an Elementary Net:

Def. 1] An elementary object net is equally identified by any of the two classes: the event class or the condition class.

We announce without proving that for distributed engineering systems each component is compositional if it is a Mealy single automaton with only one input element and a unique output, both represented by an object net. Also, it should be possible to show that this elementary component is live of class two (Murata, 1989), or equivalently, each element (or event) participates in at least one step that goes from the input to the output.

Therefore, it is time do define the contents of an object net.

Def. 2] Any elementary object net is composed of a particular arrangement of elementary object nets such that,

i)the arrangement is a bipartite directed graph composed of object nets satisfying to Def. 1;

ii)neighborhood relation in the graph is interpreted as communication channels between objects;

iii)the arrangement is a proper element;

iv)the input and output of the arrangement are events if the net superclass is identified as a condition;

v)the input and output of the arrangement are conditions if its net superclass is identified as an event;

vi)in any stage of inheritance the overall object net satisfy all the properties of an elementary net.

It is not necessary to go on with the definition of an elementary object net (EON) to conclude that it is in fact a hierarchical reinterpretation of Elementary Nets, maybe reducing topologic possibilities of arrangements for the components but not reducing the expressiveness, at least for a huge class of problems. In fact further definitions would restate the classical definitions for elementary Petri Nets as given by Thiagarajan (1986) applied to each level of hierarchy.

Thus, before defining in details the class of problems suitable to be represented by EON, let us enhance our object net by introducing the GHENeSys system, that is an enhanced kind of object net that encapsulates several extended nets, although is not in the scope of this paper to present a proof of that. However one can see the elements of several extended nets such as channel-agent, mark flow graph, and others.

The idea of GHENeSys is to concentrate everything in the side of conventional nets to jump to an object oriented approach that could preserve the analytical properties before doing a similar approach with a high level net. A plausible reason for that is that high level net by itself already is detached of the simple schema of Petri Nets that is the start point to use it in modeling and design. This point was already explored in a similar article by some of the authors of this paper where Elementary Petri Nets is used to represent elicited requirements from several viewpoints (Murata, 1989). That is an abstract domain to use net schemas that shows the potential of this formalism when applied to general design representation and modeling.

In the following we will describe briefly the basis of GHENeSys, a more thoroughly definition could be found in del Foyo (2001) and Silva (1998)

2. A Brief definition of the General Hierarchical Enhanced Net System (GHENeSys)

.

As the elementary system the idea of defining GHENeSys is to have a powerful representation of abstract discrete distributed systems. Thus, the key issue of the representation is just the introduction of partial ordering in a sequence of events directed by cause-effect relationship. However, there is a felling of lack of important features in the classical net representation to practical problems.

First of all, one would like to fold some easy a repetitive features that make the net to grow in size without adding anything to interpretation and understanding of the problem. After that, when representing workflow, there is some strategies like pipeline flow that would be better represented if multiple marks would be granted in places while maintaining the flow of one mark at a time in the arcs, as in Elementary Net.

Besides, to represent flexible and integrated systems it is necessary to compose subsystems (or subnets) that are developed including the interaction with other (external) systems. In spite of that, there is no way to represent interaction of integrated subparts even when the main property of the target system is just that.

Therefore, the proposition of GHENeSys is not just a theoretical exercise but a practical need, or at least a tentative of answer such demands which we synthesize as the representation of holonic integrated distributed discrete system by abstract schemas. Of course the objective is to face large and complex systems, what make the methodological approach very important ant so the object oriented paradigm.

Similarly to the EON, an Enhanced Object Net (EnON) is defined as,

Def. 3] An Enhanced Object Net (EnON) is equally identified by any of the two classes: the activity class or the box class.

Def. 4] Any EnON is composed of a particular arrangement of EnON´s such that,

i)the arrangement is a bipartite directed graph composed of EnON´s satisfying to Def. 1;

ii)neighborhood relation in the graph is interpreted as communication channels between objects;

iii)the graph arrangement is a proper element;

iv)the input and output of the arrangement are activities if the net superclass is identified as a box;

v)the input and output of the arrangement are boxes if its net superclass is identified as an activity;

vi)in any stage of inheritance the overall object net satisfy all the properties of an extended net.

Notice that in both object approaches only simple inheritance is considered in order to preserve the duality that is the source of expressiveness in Petri Nets.

Def. 5] A generic box class is defined by the following attributes : name, capacity, pre-set, pós-set, behavior subnet and a method that transfer calls and messages to this element to its subnet, where name is a string, capacity is a positive integer, pre-set and pos-set are proper subsets of A, the set of all activities.

Def. 6] A subclass of the class box is called pseudo-box, where capacity=1, pre-set=Ø and pos-set $\subset$A. A subnet is not defined for this class and its relation with its pos-set does no involve a token transfer. This relation could be positive or enabling or negative or disabling.

Def. 7] A generic class is defined by the following attributes: name, time step, pre-set, pós-set, behavior subnet and a method that transfer calls and messages to this element to its subnet and a method called internal clock, where name is a string, time step is a positive integer, pre-set and pós-set are proper subsets of B, the set of all boxes.

To propose GHENeSys schema of Def. 4] vi) we assume that there is a net which is a bipartite graph G=((B,A), F), where the couple (B, A) is composed by a non-intersecting sets of passive and active object nets, $B \cap A = \emptyset$, and a flux relationship $F = (B \times A) \cup (A \times B)$. This graph is a schema representation for the relationship that exist among the objects that compose the set B and those belonging to A.

Recursively, any hierarchical element from B or A can have his behavior total or partially described by a GHENeSys net.

As an illustration let us consider the following net[1],

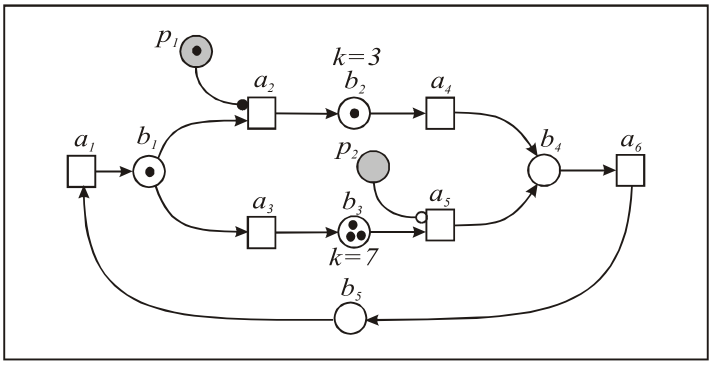

Figure 1. An small example of a net with pseudoboxes and hierarchical elements.

The structure of this net can be algebraically described by a state equation (Murata, 1989),

$$\mathbf{M} = \mathbf{M}_0 + \mathbf{C}\,\sigma$$

where M and M_0 are vectors describing the marking of passive elements belonging to L, and σ is the habilitation vector. The incidence matrix C_{nxm} is a mapping of the relations between the *n* activities and the *m* passive elements.

In the GHENeSys approach all these terms must be reviewd in order to be adapted to its new meaning, although we expect to have a similar equation and preserve the properties analysis from the classical representation of Petri Nets.

In GHENeSys, the incidence matrix for the example above is done by analysig each relation between passive and active elements in the net: $C:(X \times Y)\rightarrow\{-1,0,1\}$, where $C(x,y)$ is given by,

$$C(x,y) = \begin{cases} 1 \text{ if } (x,y) \notin \mathbf{F} \text{ y } (y,x) \in \mathbf{F} \\ -1 \text{ if } (x,y)\ \mathbf{F} \text{ y } (y,x) \notin \mathbf{F} \\ 1 \text{ if } (p,y) \in \mathbf{F} \text{ and } p \text{ is an inhibitor arc} \\ -1 \text{ if } (p,y) \in \mathbf{F} \text{ and } p \text{ is an enabling arc} \\ 0 \text{ for the remaining cases} \end{cases}$$

To cover the relationships depicted in Fig. (1), we would have the incidence matrix,

$$\mathbf{C}= \begin{array}{c} \\ \end{array} \begin{array}{cccccc|c} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & \\ 1 & -1 & -1 & 0 & 0 & 0 & b_1 \\ 0 & 1 & 0 & -1 & 0 & 0 & b_2 \\ 0 & 0 & 1 & 0 & -1 & 0 & b_3 \\ 0 & 0 & 0 & 1 & 1 & -1 & b_4 \\ -1 & 0 & 0 & 0 & 0 & 1 & b_5 \\ 0 & -1 & 0 & 0 & 0 & 0 & p_1 \\ 0 & 0 & 0 & 0 & 1 & 0 & p_2 \end{array}$$

However, we have the presence of the pseudoboxes, which stand for formal representations between a sub-net (possibly the behavior of an object) and the outside world, eventually another sub-net. Such relations are associated with non controllable events.

The new equation is given by,

$$\mathbf{M}_{k+1} = \mathbf{M}_k + \mathbf{DC}\nu_k$$

where M is an extended marking vector that includes the marks in the box and in the pseudoboxes in this order, and D is the matrix,

[1] Since it is a schema we do not be worried about what real systems it could possibly stands for.

$$D=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

which is the square matrix generated by the passive identification vector,

$$[d]_{1j} = \begin{cases} 1, \text{ if } j \text{ is a box} \\ 0 \text{ . if } j \text{ is a pseudobox} \end{cases}$$

For the example above,

$$d = [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 0]$$

Once defined a state equation all remaining properties of conventional Petri Nets can also be derived observing the unique difference between the equation proposed here and the conventional one: the matrix D which solves the problem of permanent marking in the pseudoboxes.

The class diagram bellow shows the main components of GHENeSys what follows the principles suggested in Batiston et al (2001) to object-oriented nets, where a general class ROOT are introduced as the general class T.

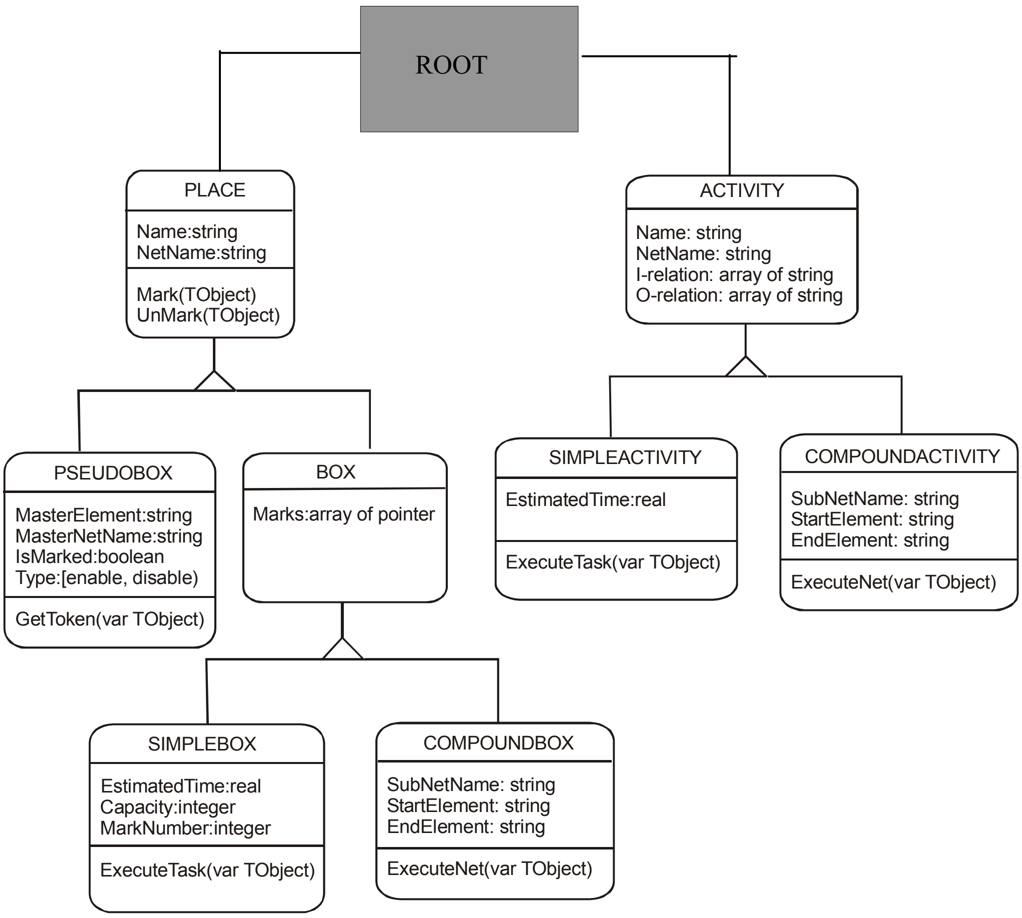

Figure 2: The Main Components of GHENeSys.

With this definition of GHENeSys we built a token player in Delphi 6.0 strictly associated with the state equation and to derivations of that to calculate the complete net, detect conflicts, analyze properties, and so on. Of course this result does not implies in an achievement to object oriented modeling using Petri Nets (at least not yet), but it is already comfortable to have the algebraic analysis and the object approach together. In the following section we will show an application to workflow where the object paradigm can be applied.

3. Application of GHENeSys to analysis of workflow

GHENeSys is very suitable to workflow applications. First of all the object approach proposed here facilitates the definition of subclasses of object nets that fits the primitive workflows identified by the Workflow Management Coalition (WFMC) (WFMC, 1996). Also, a state space analysis is more comfortable than one event driven, which is more appropriated to discrete control: GHENeSys can do that.

In what follows we show the modeling and some analysis of the complain registration proposed by van der Alst (1998) which has an interesting interpretation problem: two possible flows are linked by an external occurrence what is very difficult to be represented in conventional Petri Nets.

The process starts with the presentation of a complaint which is received and registered. Two possible actions follow in parallel: a questionnaire is sent to the complainer, while the complaint is verified. If the complainer returns the questionnaire more information will be added to the process to finish the analysis, otherwise the whole process finish without a conclusion. Of course there is a deadline to return the questionnaire after what the process is closed.

An abstract modeling of the process is showed in GHENeSys representation,

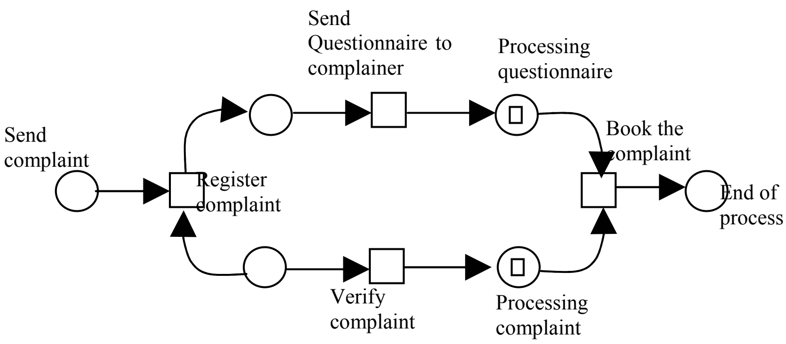

Figure 3: Generic plan for the workflow of complaints

In Fig.(3) above the two macro boxes (with a little box inside) represent the processing of the questionnaire and the processing of the complaint and must be synchronized according to an external event which models the return of a questionnaire. After some refinement that consists in replacing the macro boxes by its respective subnets the net Fig.(4) is obtained.

In this net the retrieving of information from the questionnaire and the processing of the complaint are both refined in proper object nets which were already substituted in the diagram of Fig.(3). The result is a new object net which is (dynamically) equivalent to the net in Fig.(3) but have each compounded box substituted by its proper subnet.

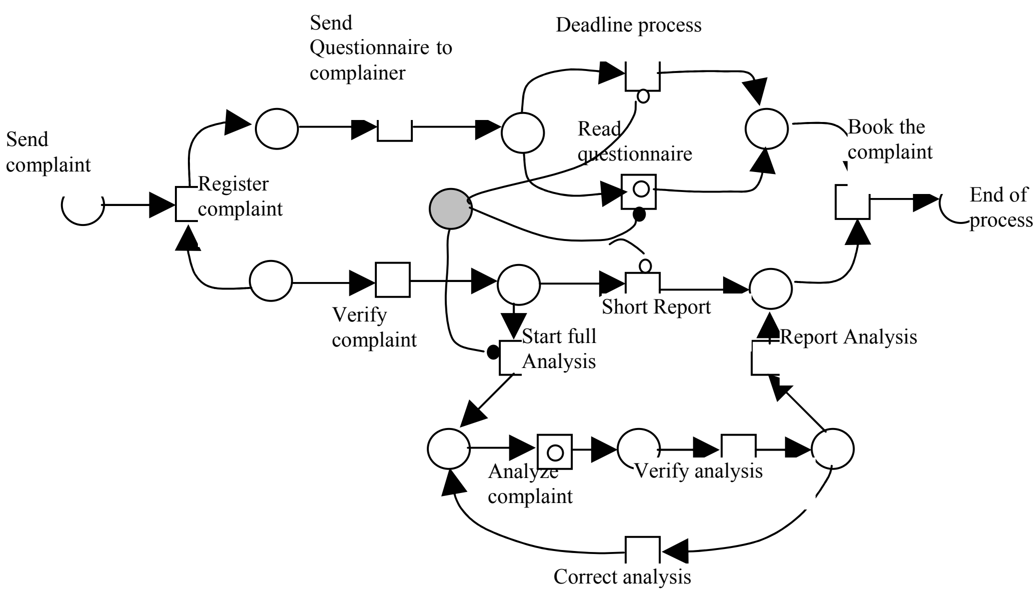

Figure 4: Model refined to include the analyzis of the complaint.

It should be noticed that the dark box represents an external non-controlable event and not an internal passive element, which would make the incidence matrix undefined and so the state equation. This element has permanent marking, meaning that if it is unmarked the questionnaire corresponding to the complaint did not return. In such case the events enabled are the processing of the deadline and the short report for the complaint without a further analysis. Conversely, if the questionnaire arrives it must be read and the complaint analyzed and verified, resulting in a more detailed report, which is also archived, finishing the process.

Of course the object net in Fig. 2 is a modeling option (once the net is a schema with all inherent modularity problems). But it should be noticed that it is in fact a good option, where the general flow is proposed to be parallel, that is, the complaint is verified internally, and an answer will be given to the costumer even if he or she does not return the questionnaire. That is better to the linear option where all process would be waiting the deadline to be processed[2].

Also, to be a real alternative, the present model admits that information (messages) about the contents of the questionnaire would be passed between the activities *Read questionnaire* and *Analyze complaint.*

The example showed here is only an illustration and we do not intend to justify the approach by the convenience in the analysis of a particular case. It is important to stress that the discipline of modeling is quite different with object nets even for simple cases like the one showed here. This difference relies on the inclusion of external non-controlable signs, which marking is not controlled by the system (sub-system) being modeled, and by the inclusion of internal messages that flows among activities (so far we do not see any use for messages flowing among passive elements although it could theoretically occurs).

Property analysis, as well as deadlock detection algorithms (Hasegawa, 1996) can be applied to the object net depicted in Fig. (4), while the possibility of taken two parallel flow where already explored. That is very helpful, mainly in the early phases of design where the objective is to come up with an abstract functional model specification of the target system.

The next step is to improve the design process by reinforcing reusability. To achieve that we proposed to have a library of abstract defined object nets which rational could be reused in several different cases. Fig.(5) shows some of these object nets applied to workflow analysis.

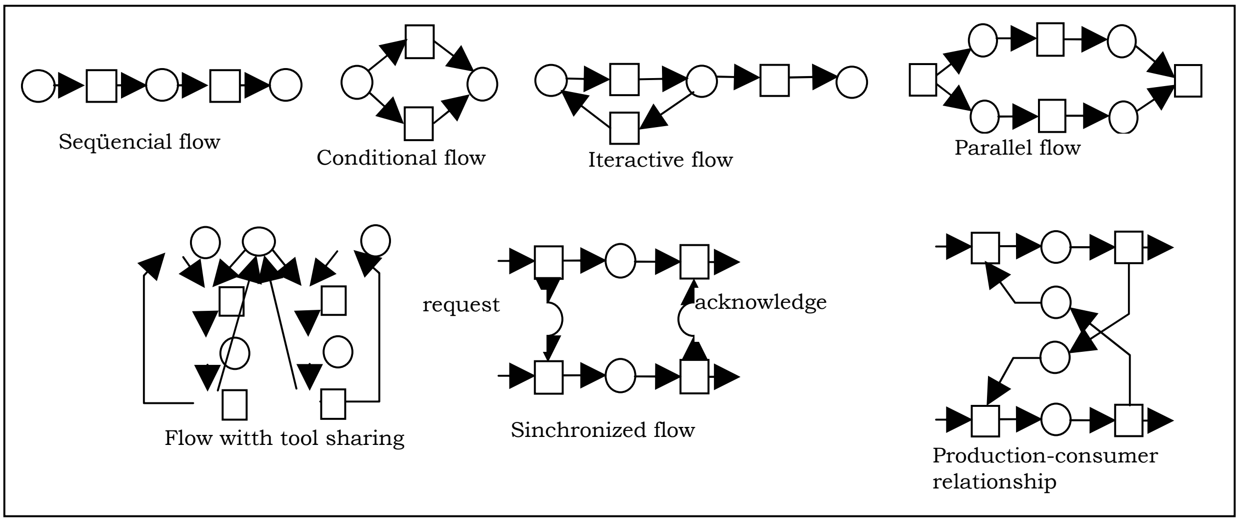

Figure 5: object nets relationales to workflow analysis

The above object subnets could be seen as patterns in workflow. They are also proper elements, that is, truly object nets in the sense discussed in this paper, so, can be used as rationales as well as net representation of the internal behavior of an object net.

4. Conclusion and further work

In conclusion we would say that, at least in what concerns workflow analysis there is a discipline for modeling and design based on object nets that could be applied with success to large systems. This discipline uses fairly the object oriented paradigm, in its practical sense, without wasting the net algebraic formalism what is always a good feature

[2] Unfortunately, such problems happens frequently not only in human organizations but also in manufacturing shop floor. To deal with that the pipeline strategy is commonly used. Notice however that in cases like the one showed here, where here is a dependency of an external occurrence, pipeline is not a solution.

when the system is large and complex and it is more advisable to relay on property analysis instead of simulation and marking gambling.

However this is only a partial result and a question still remains: it is possible to generalize this discipline to a broader class of applications or even to any application at all? We are working in to answer that question taking as a target application in planning and building and residence automation. This last one addresses the problem of discrete control which is very different from the workflow problem.

Also, the possibility to fold the net by the introduction of high level paradigms is attractive and should be always present in the general discussion. In fact, once the object orientation can also be considered a high level paradigm, the question is how to introduce types in the marks. We are now working in this topic.

5. References

Lakos C. A and Keen C.D., 1994, “LOOPN++: A new language for Object Oriented Petri Nets”, Proceedings of Modeling and Simulation (European Simulation Multiconference), pp369-374, Barcelona.

Bernardinello L and De Cindio F., 1992, “A Survey of Basic Models and Modular Net Classes”, Lecture Notes in Computer Science 609, Springer-Verlag.

Linger R. C., Mills H. D. and Witt B. I., 1979, “Structured Programming”, Addison-Wesley Publishing Company.

Murata T., 1989, “Petri nets: properties, analysis, and applications”, Proc. IEEE pp 541-580.

Thiagarajan, P.S.; 1986, “Elementary Net Systems”, Lecture Notes in Comp. Science, 254, Spring Verlag.

Santos, E.A. and Silva, J.R., 2003, “Requirements Engineering Validation Based on Viewpoints an Petri Nets”, to appear on 17h International Conference of Mechanical Engineering.

del Foyo, P.M.G., 2001, “GHENeSys : Uma Rede Estendida Orientada a Objetos Para Projeto De Sistemas Discretos”, masther thesis, (in Portuguese).

Silva J.R., 1998, “Interactive Design of Integrated Systems, Intelligent Systems for Manufacturing”, L.M.Camarinha-Matos, H. Afsarmanesh, V. Marik, (eds.), Kluwer Academic Pub.

Batiston, E., Chizzoni, A. and de Cindio, F., 2001, “CLOWN as a Testbed for Concurrent Object-Oriented Concepts, in Concurrent Object-Oriented Programming and Petri Nets”, Lecture Notes in Computer Science, State of Art Survey.

WFMC, Workflow Management Coalition Terminology and Glossary (WFMC-TC-1011), 1996, “Technical report, Workflow Management Coalition”, Brussels.

van der Aalst, W. M. P., 1998, “The Application of Petri Nets to Workflow Management”, Journal of Circuits, Systems, and Computers, Vol. 8, No 1.

Hasegawa, K. 1996, “Modeling, Control and Deadlock Avoidance of Flexible Manufacturing Systems”, invited paper, 11th Brazilian Congress of Automation, Brazilian Society of Automation, pp 37-51.

6. Copyright Notice