Chapter #

SEMIONICS: A PROPOSAL FOR THE SEMIOTIC MODELING OF ORGANIZATIONS

Ricardo R. Gudwin
DCA-FEEC-UNICAMP gudwin@dca.fee.unicamp.br *Brazil.*

Abstract: In this paper, we present Semionics, a contribution to the field of Computational Semiotics, and propose its use in order to build and simulate models of organizations. Computational Semiotics refers to a research area where semiotic techniques are used in order to synthesize semiotic processes in computers and computer-based applications. Semionics is the main technology developed by our research group, based on Peircean semiotics, with the aim of providing both modeling and simulation artifacts for the design of such semiotic systems. Here we present the main backgrounds of semionics - the semionic network - what it is and how it works. Further, we show an application example of a semionic network for the modeling and simulation of a small business organization.

Key words: Semionics, Computational Semiotics, Organizations

1. INTRODUCTION

The notion of "organization" is a fairly abstract concept that can be applied to many kinds of physical systems. This notion comes from the greek word "organon", which means "tool". Tools are artifacts or systems which have a purpose, or functionality associated to them. In this sense, we may think of organizations as special kinds of systems, where there are purposes for their existence, and these systems continuously work doing their best in order to achieve these purposes. Many different things can be classified as organizations: biological systems as cells, organs, organisms, societies, etc., or economical systems as business organizations, markets and even national and international economies. But from these examples we may guess that it is not so easy to define what an organization really is. Let's take the example of a business organization (even though the following conclusions are valid also for other kinds of organizations). This organization is not simply the sum of its employees, installations and resources. It is much more. It is its brand, its name, its connection to its market, its customers and its suppliers. And this is not all ! If we change all employees, move to a different installation and renew all its resources, it will keep being the same organization. So, this is not an easy task defining what an organization is.

Many different models of organizations were attempted (Sterman 2000) . One particular approach that proved to be of special interest is to model them in terms of the signs being processed during its behavior. In this sense, the semiotic modeling of organizations (Van Heusden & Jorna 2002) leaded the way to the creation of a new area of research that was called organizational semiotics (Alderson *et al*. 1999; Liu et.al 2000). But how can we pragmatically do such semiotic modeling of organizations? What are the artifacts that allow us to consider organizations as flows of signs in semiotic systems ? The main purpose of this work is to present Semionics, a pragmatical proposal for a both formal and computational model of sign systems, and to apply it on the semiotic modeling of organizations.

2. SEMIOTICS - THE STUDY OF SIGNS

Semiotics is the science which studies the phenomena of signification, meaning and communication in natural and artificial systems (Noth 1995). Its main artifact is the notion of *sign,* and its main approach is to explain different kinds of phenomena as being *sign processes.* The study of sign processes is documented in literature since the works of Plato and Aristotle (Noth 1998), but Semiotics, as an independent area of research was organized and structured only with the work of Charles S. Peirce, an american philosopher, during the middle of the 19th century (Peirce 1960). Even though we consider Peirce as the great exponent on developing semiotics, there are many different approaches developed in order to account for the notion of sign, and many others have contributed to the development of semiotics, like Saussure, Hjelmslev, Jakobson, Greimas and Morris - more recently Eco, Sebeok, Merrell and others (Noth 1996; Morris 1947; Morris 1964; Morris 1971; Sebeok 1997).

Both natural and artificial systems can be modeled semiotically. There are some constraints, though. When we are considering natural systems, i.e., systems that are already working in nature, the only way of semiotic modeling is due to semiotic analysis. Now, considering artificial systems, we can apply both semiotic analysis and semiotic synthesis. We use semiotic synthesis in order to artificially create semiotic processes. In this case, we are not mere expectants of the miracles of nature, but actants in order to fully synthesize devices where semiotic processes do occur. Of course, after making the synthesis, we are also able to employ semiotic analysis on the synthesized systems, but we will see that the synthesis problem is sometimes harder than its analysis counterpart.

Many different strategies may be employed for semiotic synthesis. On the next section, we will present Semionics, our proposal for semiotic synthesis.

3. SEMIONICS

We may understand Semionics as a particular way of implementing the notion of a sign in a formal and computational way. So, before describing the details of semionics, it is important to analyze the different models of signs available within semiotics, in order to characterize the power and constraints of each available option.

Let's start with the dyadic sign as proposed within structuralist semiotics, presented in figure 1.

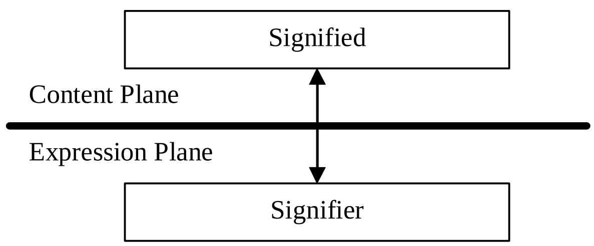

Figure 1 - The Structuralist Model of the Sign (dyadic)

In this model, there are two planes, bi-univocally connected - the so called expression plane and the content plane. Within expression plane, we find discriminable unities which we call "signifiers". On the content plane, we find discriminable unities we call "signifieds". Unities on the expression plane and on the content plane are related to each other, forming cartesian pairs of the type (signifier, signified). A sign (according to the structuralist view) is then defined as being such a pair (signifier, signified). So, in this model, a sign is viewed as a dyadic relation that connects a signifier to a signified. This model has its origins on the work of Saussure (Nöth 1996), being further enhanced by Hjelmslev and others. In its original acception, both signifier and signified were supposed to be mental units. So, an example of a signifier would be the sound of the word "car", defined on an expression plane of sound waves, encoded accordingly to become a mental term, and its signified will be the idea we have of a car, also encoded in a mental way. Some variations on this model may associate the expression plane to an inner world, and the content plane to the external world. Then, to each signifier on the inner world (mental world), there should be a natural correlate on the external world (content plane). The problem with this view is that such correlation is totally arbitrary. This is the vision proclaimed by cognitivism within cognitive sciences, which says that if in a computer memory a given set of signals represent the proposition "Socrates is mortal", the connection between these signals with the historic Socrates (the individual), and the fact that this man has a property of "being mortal" would be "automatic". This totally arbitrary connection between signifier and signified is apparently the fragility of this model, being the origin of the symbol grounding problem in artificial intelligence (Harnad 1990). With this model, it is also impossible to model the so said natural signs - the icons and the indexes, but only symbols.

As a contraposition to the dyadic model of the sign, Peirce developed a more elaborated, triadic (Noth 1995) model, which splits the notion of "signified" into two different parts, one of them connected to an element of real world - the so called "object" of the sign, and the other connected to the effect of the sign on the mind of a potential interpreter, called the "interpretant" of the sign. A sign, according to Peirce, is something which, under some aspect or mode, represents something else to someone. This sign will create on the mind of this "someone" a second sign, equivalent to itself - that is, a more developed sign, which is called its "interpretant". Both sign and interpretant refer to exactly the same object (Peirce 1960; Santaella 2000). So, the process of semiosis (or a meaning process), is a triadic relation that bounds a sign, an object and an interpretant, as shown in figure 2:

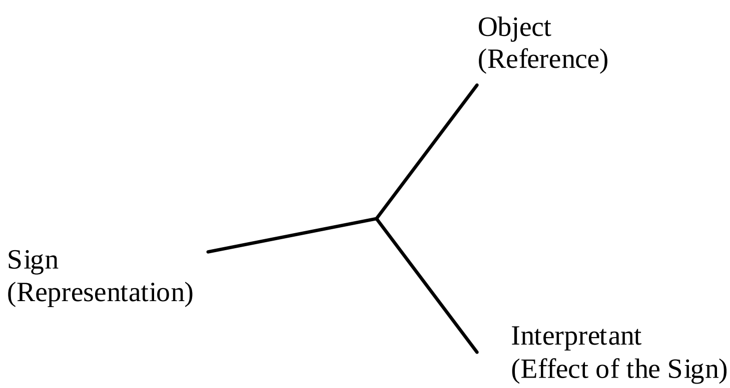

Figure 2 - The Peircean Model of the Sign (triadic)

In this process, the object, by means of its relation to the sign, confers to the sign the power to represent it. This power is consolidated during the generation of the interpretant, within the mind of a potential interpreter (even though the notion of interpreter is not necessary, according to Peirce). Note that in this case, there is no "automatic" connection between a signifier and a signified, as in the dyadic model, but a process in which a sign only becomes a sign, when it possess this capacity of generating an interpretant, and it will only have this capacity, in virtue of the relation that it have with its object. This allows the definition of natural signs, i.e., the icons and indexes, which were not allowed on the dyadic model.

The Peircean model (Peirce 1960; Santaella 2000) is elaborated and full of details (which we omit in this article, as they are out of scope here). But let us better appreciate the relation between a sign and its object, which is what gives the sign its character of being interpretable. So, according to the nature of the relation between a sign and its object, signs can be divided into icons, indexes and symbols.

Icons are signs which have in their structures some relation of similarity to his designated object. In other words, they have in themselves the same qualities (or a subset of them) that the object itself has. This is the main reason they are entitled to represent their object. Icons can be divided into three different sub-categories - images, diagrams and metaphors. Images have in themselves, the same qualities as their objects. This happens, for example, when we use a picture to represent something that was photographed. So, by viewing the picture of a house, and appreciating its qualities, we know which house is this. This meaning is so split into two parts. From one side, we have the real house, the one which was photographed, and which is referred by the picture. On the other side, there is the idea that we have of such a house, idea that is triggered in our mind, due to the presence of the sign. Diagrams, opposed to images, do not possess in themselves directly the same qualities as their object. But they present a relation between their parts which are equivalent to the relations that hold for the object’s parts. So, the relation among the parts of a diagram is equivalent to the relations among the parts of the object on real world. This is a different kind of icon. The metaphors, which are another kind of icon, are connected to their objects by means of abstractions that we are able to make to both the sign and the object. So, even though sign and object do not share qualities, the abstractions we make for both sign and object do share these qualities. This is so, a more sophisticated kind of icon. In a general way, we say that icons do not depend on their objects (that is, they do not depend on the simultaneous presence of their objects for the interpretation), because they hold in themselves the object’s qualities, which allows them to be interpreted as signs of it. On the contrary, we will see that indexes and symbols will depend on their objects and interpretants, respectively, in order to be interpreted (Peirce 1960).

The symbol, on the same way as the icon, carries in itself an absolute meaning. But, opposed to it, it doesn't need to have in itself the same qualities present on the object. Its power to represent an object is instead related to an arbitrary convention, or law, that binds the sign to the object. The interpretation of a symbol comes in two steps. In a first step, something presented to the system is recognized due to its attributes as an icon, corresponding to the identification of the sign in itself (as something known). In a second step, a convention linking this already known icon to something else is invoked and used to connect the preliminary sign to the final object. We see that an important element on the interpretation of a symbol is the arbitrary convention that binds the sign to the object. This convention occurs due to a personal decision of the interpreter (e.g. ... "from now on I will call this xyzt" ...), or due to a pact with other interpreters, to whom this interpreter wants to communicate. The celebration of this pact involves a sophisticated protocol, still not known in its entirety, involving icons and indexes.

The third kind of sign, the index, does not have an absolute meaning, as the icon and the index. Its meaning is otherwise relative to some existing connection to the object. A good hint to understand what is an index is to think on it as a "key" to a procedure which will point to the object. An example will be referential indexes, like "this" or "that". The meaning of "this", or "that" is not absolute, but will depend on the context where these words appear.

Now, in order to understand "Semionics", let's relate these concepts well known within semiotics to a computational procedure that is proposed in order to materialize semiotics within computers. We will start with the Peircean model of the sign, as presented in figure 3.

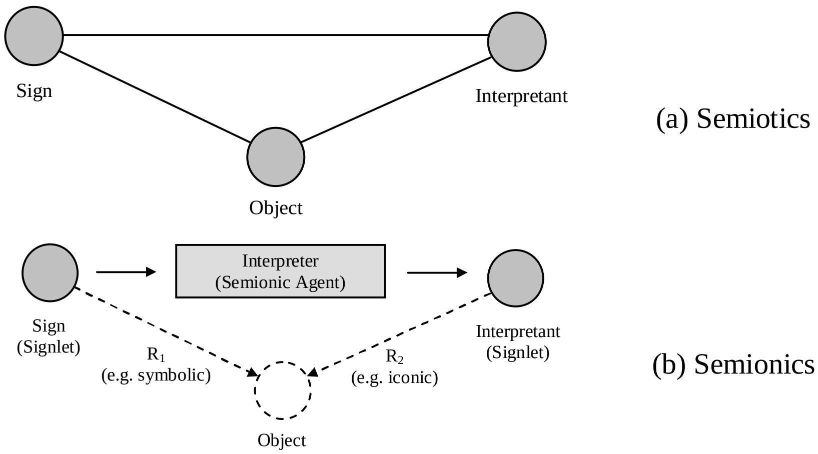

Figure 3 - Linking Semiotics to Semionics

In figure 3a, we have the Peircean model of the sign, as a triadic relation among sign, object and interpretant. The idea is that we have three distinct entities that are related to each other in the sense that there is a relation between the sign and the object, which confers to the sign its power to be turned into an interpretant. At the same time, this interpretant needs to be related to the same object, allowing the possibility of a potentially infinite chain of successive transformations like that, in which each new interpretant keeps a relation to the same original object. On figure 3b, we have our proposal for an equivalent relation between three distinct entities, in a computational version of the same triadic relation found in figure 3a. In this case, each entity in figure 3a is turned into a computational entity that we call a "signlet". We also propose the notion of a "semionic agent", which performs the role of an interpreter, translating a given signlet, performing the role of a sign, into another signlet, performing the role of an interpretant. In order to generate the signlet performing the role of an interpretant, the semionic agent supposes that the other signlet (the one which performs the role of a sign) should have a relation to a third element, their object (which is also supposed to be a signlet). Based on this supposed relation, it tries to propose a signlet that should maintain some kind of relation with this same object. In the example on figure 3b, an input signlet has a symbolic relation to a presumed object, and after the interpretation, the semionic agent generates a signlet which has an iconic relation to this same object.

This process, of generating an interpretant from a sign, based on a supposed relation of both of them to a same object can be a very complicated procedure. Even though it appears to be a sequential process, in order to better understand it, we need to decompose this external, or exosemiotic view, into an internal, or endosemiotic view (figure 4).

From an internal, endosemiotic view, the same process of interpretation, where a given semionic agent takes a signlet and transform it into other signlet can be performed by a great number of other (internal) semionic agents, creating a whole network of interpretations, that will result, from an external perspective, into an exosemiotic behavior. Different semionic agents would make different guesses on the supposed object related to both sign and interpretant, resulting into different potential interpretants, which will compete to each other in order to generate the final interpretant appearing at the exosemiotic process. So, from the point of view of Semiotic Synthesis, this endosemiotic understanding of the behavior of the interpreter is very much important, as the exosemiotic process can be a composition of many intricate endosemiotic processes, becoming a complex network of semiosic processes occurring in parallel and in real time. Now, if we want to model (and build) such an endosemiotic system, we will need an artifact that should be able to support these requisites: it needs to model the dynamics of discrete event systems (Cassandras 1993), which are also concurrent processes. If we go to literature, we will see that these requisites are well supported into a mathematical tool that is called Petri Nets (Murata, 1989).

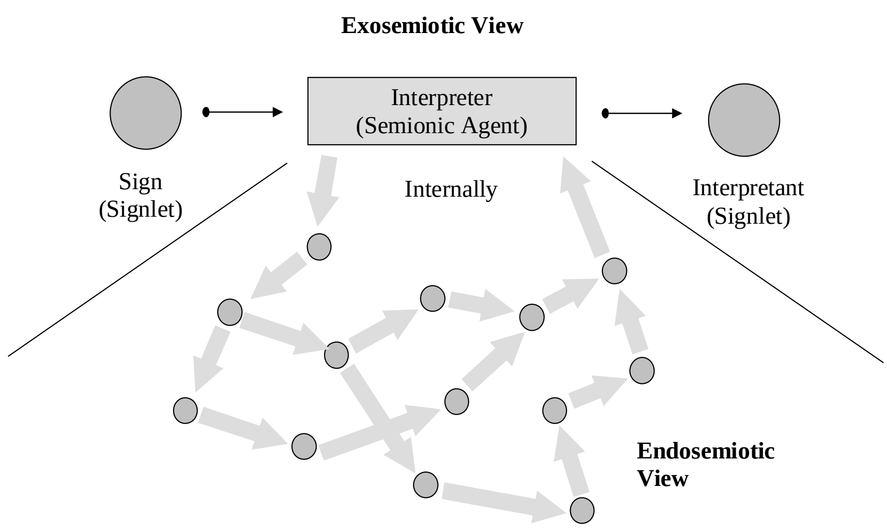

Figure 4 - Exosemiotic and Endosemiotic View of an Interpretation

But standard Petri Nets are not enough for our purposes, because tokens are unstructured and transitions have no processing capabilities. We may seek then into extensions and variations of Petri Nets, like Higher Level Petri Nets (Genrich & Lautenbach 1981), Coloured Petri Nets (Jensen 1990) or Object-based Petri Nets (Lakos 2001), where tokens are structured, and transitions have (at least some) processing capabilities. But, again, Coloured Petri Nets and Object-based Petri Nets are not enough, because we have two different kinds of entities in our system, signs and interpreters, both of them structured, something that is not supported within these extensions of Petri Nets. So, we decided to create a new extension of a Petri Net, contemplating the requirements we envisioned for semionics, reaching a model we called "Semionic Networks". Semionic networks (Gudwin 2002) are a development that came after many other previous developments, like Object Networks (Gudwin 1996; Gudwin & Gomide 1997a,b,c, 1998) and Agent Networks (Guerrero et. al. 1999).

4. SEMIONIC NETWORKS

An example of a semionic network can be viewed in figure 5, below.

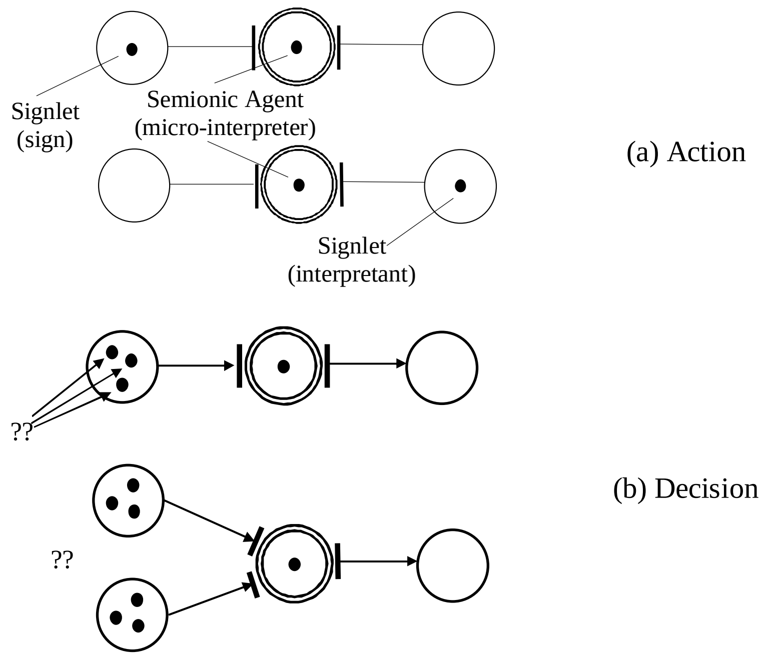

Figure 5 - Action and Decision in a Semionic Network

Signlets and semionic agents are distributed within a network of places, where each place may have different ports, and places are connected by means of arcs linking two ports in different places. Semionic agents perform their role by taking a signlet and generating a newer signlet. This is shown in figure 5a. The semionic agent in the place with the double line takes a signlet from the place on its left, and generates a newer signlet that is put into the place on its right. This could be a very simple procedure, if we consider that there is only one signlet and one semionic agent. But in fact, if we are going to use this to model the endosemiosic processing we suggested in figure 4, we need to cope with the existence of many signlets and eventually many semionic agents competing to each other. This is the situation presented in figure 5b. In the first case, we show the situation in which a semionic agent in the middle place has many options of signlets to process. Which one it will process ? The second situation shows that this can be either more complex, if we consider that we need to compose many signlets in order to perform the interpretation. So, we need to better elaborate the behavior of our semionic agents.

As suggested by the examples in figure 5, a semionic agent needs to perform two main tasks: decision and action. In the decision task, it needs to choose which sign (or which signs, in the case of a composition) it is going to interpret, and also what is going to happen to this sign, if it is going to be preserved or not. In the action task, the semionic agent needs to materialize the interpretation, generating the interpretant, based on the chosen sign.

The *decision task* is performed by means of two distinct phases, the evaluation phase and the assignment phase. The *action task* is also performed by means of two other phases, the assimilation phase and the generation phase. In order to understand how those phases work together, we need first to dig into the structure of signlets.

A signlet is a computational entity that is basically a tuple of compartments, just like in figure 6 below:

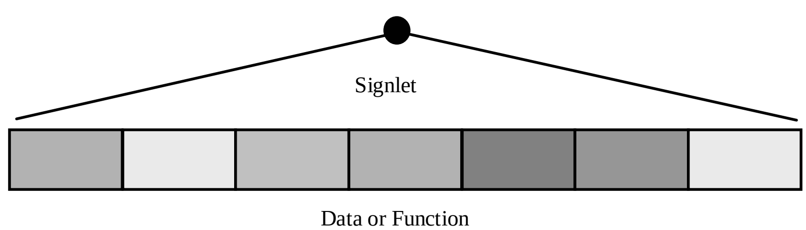

Figure 6 - A Signlet

Signlets can be organized into classes, according to the types of its compartments, which can be either data or functions. In this sense, we will see that semionic agents can be defined also as signlets, but with a special arrangement of compartments, just like in figure 7 below:

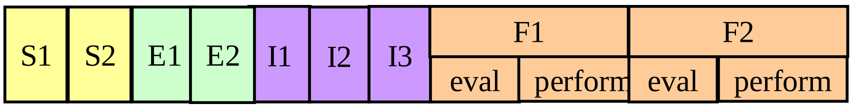

Figure 7 - Semionic Agents are Signlets

In the case of a semionic agent, the compartments are divided into 4 regions. They can be sensors, effectors, internal states and mediated transformation functions. Each mediated transformation function is described by two different functions, the eval function and the perform function. They are called mediated transformation functions, because the perform functions are executed only mediated by the result of the eval functions, according to the phases described in the sequence.

4.1 The Decision Task

In the decision task, the semionic agent needs to decide which signlet (or signlets) it is going to interpret, and what is going to happen to this signlet. This is not an easy task, because there may be many different signlets available for the semionic agent, and also many possible semionic agents interested in the same signlet. So, the decision task must be implemented in a coordinated way, in order to allow multiple chains of semiosis to happen in parallel an concurrently. The decision task is split into two different phases, the evaluation phase and the assignment phase.

The evaluation phase starts when a given semionic agent is faced to the many available signlets, and considering the different transformations it is able to apply to them, it needs to evaluate each available signlet, and at the same time decide what is going to happen to it after the interpretation. This last step is necessary because signs may be perennial or not, and also semionic agents may require exclusive rights or not in the process of interpreting the sign.

A pictorial illustration of what happens during the evaluation phase is given in figure 8.

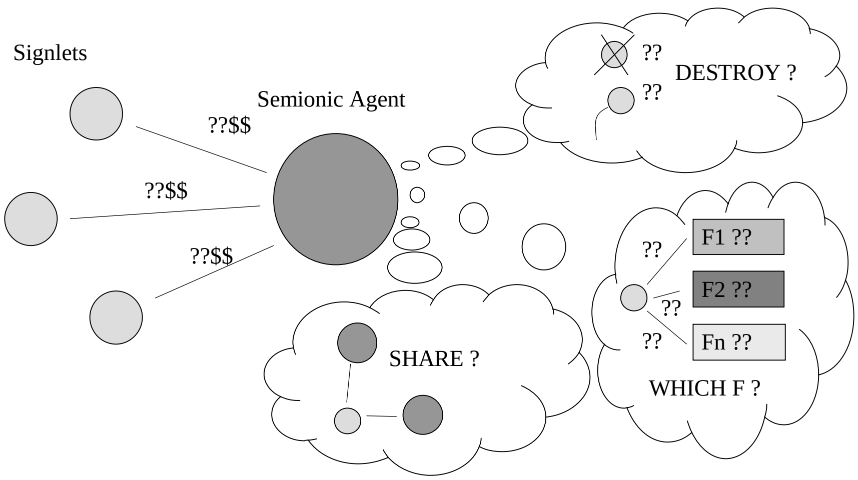

Figure 8 - The Evaluation Phase

Basically, for each transformation function available within the semionic agent, a set of signlets required to perform it is determined among the available signlets. All possible combinations of available signlets that match the function requirement must be evaluated. Each possible combination, in the form of a list of potential signlets, is called an enabling scope. Each possible enabling scope must be evaluated by means of an evaluation function, which should provide a score to the enabling scope, and also a destiny to it. The possible destinies are:

– a signlet should be modified and sent to a different place
– a signlet should be dropped back to its original place
– a signlet should be destroyed after the interaction

The evaluation phase ends when the semionic agent evaluates all available enabling scopes and attributes to each one an interest value and a pretended access mode.

The pretended access mode describes the semionic agents planned actions to each input signlet. It should inform if the semionic agent plan to share or not the signlet with other semionic agent and if it plans to destroy the signlet after its use.

After all enabling scopes are evaluated and rated, a second phase starts, the assignment phase. The assignment phase is responsible for solving possible conflicts with the plans of each semionic agent in the network. So, to solve that, a central supervisory algorithm computes the plans of each semionic agent and selects for each of them an enabling scope. This selection should avoid any kind of conflict with the plans of other semionic agents. Many different algorithms can be used in this phase to solve this scheduling problem. For test purposes, our group developed an algorithm (Guerrero et.al. 1999), which we called BMSA (Best Matching Search Algorithm), which attributes a signlet to the semionic agent that best rated it, respecting the pretended access modes of each semionic agent.

4.2 The Action Task

In the action task, the semionic agent simply follows the plan assigned in the assignment phase, generating a new signlet, destroying signlets or modifying them. The action task is also divided into two sub-phases, the assimilation phase and the generation phase. In the *assimilation phase*, the semionic agent decides for a course of action, depending on the access mode given by the decision task. Depending on the access mode, the semionic agent will *read* or *get* the signlets on its inputs. In the case of a *read*, the semionic agent stays only with a reference to the signlet, so it can have access to its internal contents, but it is not supposed to change the internals of the signlet. In the case of a *get*, the semionic agent fully assimilates the input signlets, becoming the owner of it, and in this case, it is able to modify its contents and change it for further reuse. After assimilating the necessary information, the semionic agent then, depending on the given plans, may leave the signlet in its original place, destroy it permanently or take it from its original place, in order to process it. This is the end of the assimilation phase. After managing the future of input signlets, the semionic agent turns to the generation phase, where it will generate new signlets, if it is the case. In the generation phase, after getting the available information from input signlets, this information is used to generate new signlets, or to modify an assimilated signlet. This information is then processed, applying a transformation function that will generate new signlets, which are then sent to the proper places in the network.

4.3 Special Cases

There are two special kinds of semionic agents, which are useful to be pointed out. These are the *sources* and the *sinks*. Sources are special kinds of semionic agents that don't have inputs, only outputs. The result is that signlets are constantly being generated and being inserted into other places, in a semionic network. Sinks, on the opposite, are semionic agents that don't have outputs, but just inputs. These semionic agents are used to take signlets from places on the network and destroying them. Sources and sinks can be used in a semionic network to link the network to external systems.

4.4 The SNToolkit

In order to create computational models of semionic networks, and use them to simulate organizational processes, our group built the SNToolkit (Guerrero 1999), the Semionic Network Toolkit, a software tool for editing and simulating semionic networks. A view of this tool is presented in figure 9 below:

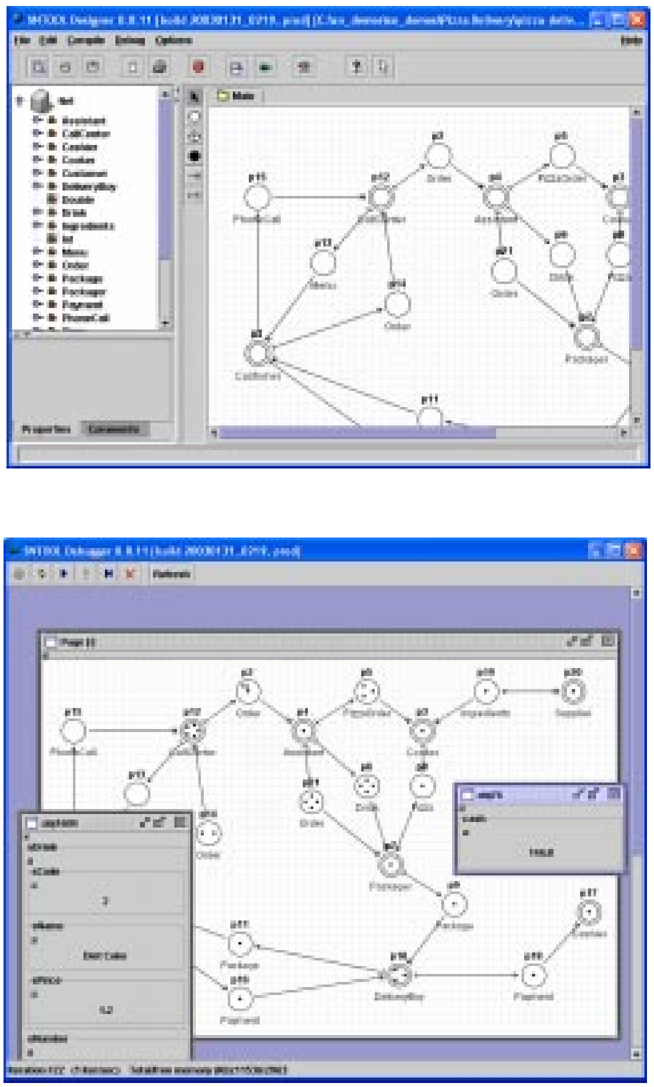

Figure 9 - Screenshots of the SNToolkit

5. MODELING ORGANIZATIONAL PROCESSES

Now, in order to understand how to semiotically model organizational processes, using semionics, we need to introduce some fundamental concepts. And the first concept we will start with is the own notion of what is an organization. We will define an organization as a network of resource processing devices performing a purposeful role. In this sense, we may understand a resource as a very abstract concept that can be applied to many different domains of knowledge. These resources may have an associated "value" or "cost", which can be used on the models being developed. A resource can be almost anything: something material, some information, a machine, a person, whatever plays a meaningful role into an organization. We differentiate between two major kinds of resources: passive resources and active resources. Passive resources are resources that are passively managed or manipulated during the organization activities. They can be material resources like objects, parts, products, raw-materials, money, etc... , or informational resources, like texts, documents, diagrams, data, sheets, tables, etc... Active resources are processual resources, or in other words, resources that execute activities of resource processing. Active resources can be mechanical resources (or processors without decision-making) or intelligent resources (or processors where there is some kind of decision-making). Examples of active resources are machines, human resources (workers), etc...

The main idea now, here, is that organizational processes can be described in terms of sign processes, which is the main idea behind organizational semiotics. Resources within an organization can be modeled in terms of signlets and semionic agents. Passive resources are modeled as signlets and active resources as semionic agents. So, a network of resource processing (an organization), can be modeled in terms of a semionic network. It is important to notice that both intelligent and mechanical active resources can be modeled in terms of a semionic agent. But the most interesting case, of course, is the case of intelligent active resources, as mechanical processes can be easily modeled by standard Petri Nets. From Peircean semiotics, we borrow the notions of abduction, deduction and induction as the elementary operators being applied on signs. Abduction refers to the generation of newer knowledge structures (signlets). Deduction is related to the extraction of explicit knowledge structures from implicit knowledge structures. And induction is the evaluation of a given knowledge structure in terms of the system purpose. We propose that semionic agents are able to perform decision-based actions, and that the coordination between evaluation and transformation functions in semionic agents allows a semionic agent to perform the three main semiosic steps: abduction, deduction and induction. So, the coordinated work of a network of semionic agents may allow the representation of full semiotic processes (the endosemiosic view) and in this sense, we say that actions performed by semionic agents are mediated actions, where the transformation function is mediated by the evaluation function. An example of a model of a business organization using a semionic agent is given in figure 10.

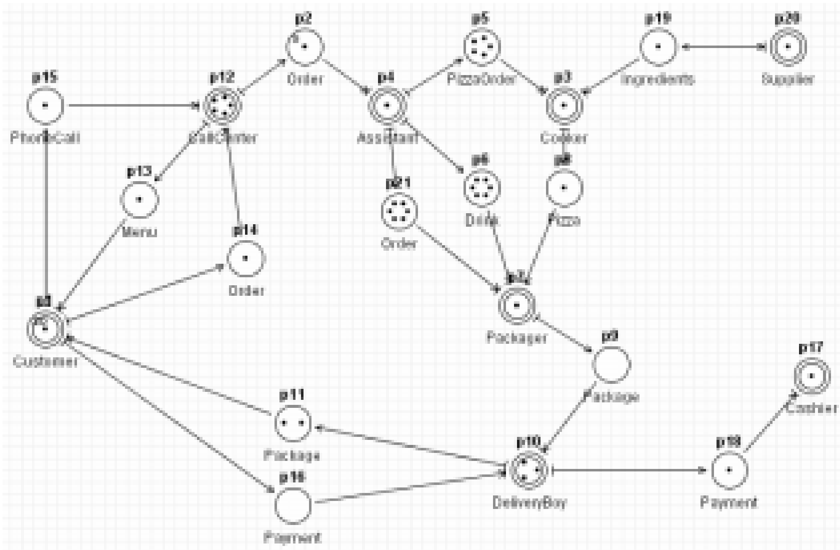

Figure 10 - An Example: A Pizza Delivery Organization

In this example, there are many active resources on the organization, performing the roles of customers, call center attendees, assistants, cookers, suppliers, packagers, delivery boys and cashiers. Each active resource is represented by means of a semionic agent and is placed into a place on the network. Passive resources are phone calls, menus, orders, ingredients, pizzas, drinks, packages, payments, and others, which are represented by means of signlets and put on different places around the network, according to its purpose on the organization. A semionic network like the one in figure 10 can be simulated on SNToolkit, and many different kinds of results can be collected. We may change the number of employees performing the different roles, give them different organizational procedures, and with that reengineer the whole organization and simulate what should be the results of these changes.

So, many different things can be done with this framework. We can use it to model and simulate multiple levels of abstraction of an organization, focusing on the resources processed and on the deliverables created and used. We may test and simulate multiple configurations, making a simulated reengineering of an organization. They can also be used as a both formal and computational model of the organization, which can be used to better understand the dynamics of such an organization. And we can also build information systems better suited to the organizational structure, and which better represent the control demands of the organization.

6. CONCLUSIONS

We presented in this work a new approach for the semiotic modeling and analysis of organizations, which we called semionics. The main artifact in semionics is the semionic networks, which are a potentially interesting tool for the semiotic modeling of organizations, as we pretended to have shown in the given example. Even though the main guidelines for this approach are already delineated in this work, we are conscious that there is still a lot of work that remains to be done. For example, a comparison between our approach and other approaches used in the study of organizations and workflows, is a must. In order to do that, there are many standards and proposals that need to be checked, like those from the Workflow Management Coalition Standards (Hollingsworth 1995), the Enterprise Distributed Object Computing (OMG 2002) and other models of business processes found in the literature. We also need a more complex study case of a real organization, as for today, only demonstrations and proof-of-concept implementations were generated until now. A real study case may suggest new features to be included on the approach, or even changes on the current features. We also need a better understanding of the semiotic contributions to this kind of modeling, which is an issue to be analyzed by the organizational semiotics community. This is just a preliminary presentation of a promising tool for the modeling and simulation of organizations, which still needs a lot of work.

REFERENCES

Alderson, A. Yap, J, Liu, K. and Shah, H.U. 1999, Relating Organisational Semiotics, Process Modelling and Stakeholder Viewpoints to Elucidate and Record Requirements, Proceedings of Systems Modelling For Business Process Improvement March 1999, pp. 168- 220

Cassandras, C.G. 1993, Discrete Event Systems: Modeling and Performance Analysis, Aksen Associates Incorporate Publishers.

Genrich, H.J. and Lautenbach, K. 1981, System Modelling with High Level Petri Nets - Theoretical Computer Science 13, pp. 109-136.

Gudwin, R.R. 1996, Contributions to the Mathematical Study of Intelligent Systems - Ph.D. Thesis - DCA-FEEC-UNICAMP - (in portuguese) available on-line at http://www.dca.fee.unicamp.br/~gudwin/Publications/thesis.html

Gudwin, R. R. 2002, Semiotic Synthesis and Semionic Networks S.E.E.D. Journal (Semiotics, Evolution, Energy, and Development), Volume 2, No. 2, p. 55-83.

Gudwin, R.R. and Gomide, F.A.C. 1997, Computational Semiotics : An Approach for the Study of Intelligent Systems - Part I : Foundations" – Technical Report RT-DCA09 - DCA-FEEC-UNICAMP. available on-line at http://www.dca.fee.unicamp.br/~gudwin/ftp/publications/rep1_97.pdf

Gudwin R.R. and Gomide, F.A.C. 1997, Computational Semiotics : An Approach for the Study of Intelligent Systems - Part II : Theory and Applications" - Technical Report RT-DCA09 - DCA-FEEC-UNICAMP. available on-line at http://www.dca.fee.unicamp.br/~gudwin/ftp/publications/rep2_97.pdf

Gudwin, R.R.and.Gomide, F.A.C. 1997, An Approach to Computational Semiotics- Proceedings of the ISAS Conference, Gaithersburg, pp.467-470.

Gudwin, R. and Gomide, F. 1998, "Object Networks – A Modeling Tool" – Proceedings of FUZZ-IEEE98, WCCI'98 - IEEE World Congress on Computational Intelligence, 4-9, Anchorage, Alaska, USA, pp. 77-82

Guerrero, J.A.S Gomes, A.S.R. Gudwin, R.R. 1999, A Computational Tool to Model Intelligent Systems - Anais do 4o SBAI - Simpósio Brasileiro de Automação Inteligente, 8-10 September, São Paulo, Brasil , pp. 227-232

Harnad, S. 1990, The Symbol Grounding Problem, Physica D, 42:335-346.

Hollingsworth, D. 1995, Workflow Management Coalition - The Workflow Reference Model, Document Number TC00-1003, Issue 1.1, Workflow Management Coalition - http://www.wfmc.org.

Jensen, K. 1990, Coloured Petri Nets : A High Level Language for System Design and Analysis - Lecture Notes in Computer Science 483 - Advances in Petri Nets, pp. 342-416.

Lakos, C. 2001, Object Oriented Modelling with Object Petri Nets - In G. Agha et al. (Eds.): Concurrent OOP and PN, Lecture Notes in Computer Science, pp. 1-37.

Liu, K. Clarke, R.J. Andersen, P. B. Stamper, R.K. 2000, Information, Organisation and Technology - Studies in Organisational Semiotics (Information and Organization Design Series, Volume 1), Kluwer Academic Publishers, Boston.

Morris, C.W. 1947, Signs, Language and Behavior - New York : Prentice Hall.

Morris, C.W. 1964, Significant and Significance – New York – Prentice Hall.

Morris, C.W. 1971, Foundation for a Theory of Signs - in Writings on the General Theory of Signs - The Hague : Mouton.

Murata, T. 1989, Petri Nets : Properties, Analysis and Applications - Proceedings of the IEEE, vol. 77, n. 4.

Noth, W. 1995, "Handbook of Semiotics" - Indiana University Press - Bloomington & Indianapolis.

Noth, W. 1996, A Semiótica no Século XX - (Semiotics in the 20th Century - in portuguese), Annablume Editora, São Paulo, Brasil.

Noth, W. 1998, Panorama da Semiótica: De Platão a Peirce - (A Semiotic Panorama : From Plato to Peirce - in portuguese) Annablume Editora - São Paulo, Brasil, 1998 2d edition.

Object Management Group, 2002, UML Profile for Enterprise Distributed Object Computing Specification- OMG Adopted Specification ptc/02-02-05, http://www.omg.org/docs/ptc/02-02-05.pdf

Peirce, C.S. 1960, "Collected Papers of Charles Sanders Peirce" - vol I - Principles of Philosophy; vol II - Elements of Logic; vol III - Exact Logic; vol IV - The Simplest Mathematics; vol V - Pragmatism and Pragmaticism; vol. VI - Scientific Metaphysics - edited by Charles Hartshorne and Paul Weiss - Belknap Press of Harvard University Press - Cambridge, Massachussets.

Santaella, L. 2000, A Teoria Geral dos Signos, Editora Thomson Pioneira.

Sebeok, T.A. 1997, The Evolution of Semiosis, Tutorials 1 – Semiotics - ISAS Conference, Gaithersburg.

Sterman, J.D. 2000, Business Dynamics: Systems Thinking and Modeling for a Complex World - McGraw-Hill Companies.

Van Heusden, B. Jorna, R.J. 2002, "Reconsidering the Standard: A Semiotic Model of Organisations" In: Kecheng Liu, R.J. Clarke, P.B. Andersen & R.K. Stamper (Eds.) Coordination and Communication Using Signs: Studies in Organisational Semiotics 2. Dordrecht: Kluwer Academic Publishers, p. 153-67.