HELSINKI UNIVERSITY OF TECHNOLOGY
DIGITAL SYSTEMS LABORATORY
Series **A:** Research Reports ISSN 0783–5396
No. **23**; June 1993 ISBN 951–22–1639–6

A SHEAF SEMANTICS FOR PETRI NETS

Johan Lilius

Digital Systems Laboratory
Department of Computer Science
Helsinki University of Technology
Otaniemi, FINLAND

Helsinki University of Technology
Department of Computer Science
Digital Systems Laboratory
Otaniemi, Otakaari 1
SF–02150 ESPOO, FINLAND

HELSINKI UNIVERSITY OF TECHNOLOGY
DIGITAL SYSTEMS LABORATORY
Series **A**: Research Reports
No. **23**; June 1993

ISSN 0783–5396
ISBN 951–22–1639–6

A Sheaf Semantics for Petri Nets *

JOHAN LILIUS

Abstract: The semantics of Petri Nets are discussed within the "Objects are sheaves" paradigm. Transitions and places are represented as sheaves and nets are represented as diagrams of sheaves. Both an interleaving semantics, and a non-interleaving semantics are shown to arise as the limit of the sheaf diagram representing the net.

*This work was supported by the Information Technology Promotion Agency, Japan, as part of the R & D of Basic Technology for Future Industries "New Models of Software Architecture" project sponsored by NEDO (New Energy and Industrial Technology Developments Organization).

Printing: TKK Monistamo; Otaniemi 1993

Helsinki University of Technology
Department of Computer Science
Digital Systems Laboratory
Otaniemi, Otakaari 1
SF–02150 ESPOO, FINLAND

Phone: $\dfrac{90}{+358\text{–}0}$ 4511
Telex: 125 161 htkk sf
Telefax: +358–0–465 077
E-mail: lab@hutds.hut.fi

1 Introduction

Sheaf theory is a mathematical tool that has been successfully applied to the solution of difficult mathematical problems. In [6], Gray surveys applications to complex analysis, algebraic geometry, differential equations, and category theory.

Sheaf theory has also been used in a general formulation of systems theory by Goguen [2, 3, 5]. The basic building blocks of this categorical formulation of General Systems Theory besides sheaves are the categorical notions of diagram, limit and colimit. The behavior of a component or system is represented by a sheaf. A system is represented as a diagram of sheaves where the diagram describes how the different components are interconnected. The behavior of a system is calculated by taking the limit of the diagram of its subcomponents, and colimits are used to connect systems. This paradigm was later extended to models of concurrency in [4] and used to give a semantics for the FOOPS language in [10].

This paper is an application of the "Objects are sheaves" paradigm as formulated in [4] to one specific model of concurrency, namely Petri Nets. The main result of this paper is that two well known semantics for Petri Nets arise through the categorical process of taking the limit.

We will start with a very simple view of a transition in a Place/Transition net. This view will then be refined giving us an interleaving semantics with a global firing rule. Then we will focus our attention on Elementary Net systems. We will further refine our semantics so that it will a non-interleaving semantics for Elementary Net systems.

It is assumed that the reader has a basic knowledge of category theory and Net theory.

2 Sheaves and Transitions

The basic idea in the "Objects are sheaves" paradigm is that an object has an internal state that can change, and that this state can only be observed through slots that are called attributes. From an external point of view, an object consists of a set of attributes, and the only thing we know about the object is that what we can observe about it. Observations may be partial in the sense that at the current moment we have only observed certain parts of the object. An important requirement for observations is that partial or local observations can be "pasted" together to global observations.

Observations are formalized in terms of functions $f : U \to A$ where U is some space-time domain and A is the domain of the attributes of the object. The elements of the domain U are partially ordered by inclusion, and closed under finite intersection and arbitrary union, ie. form a topological space, but in this paper the following definition is sufficient:

Definition 1
A *base* $\underline{\mathrm{T}}$ for observations is a family of sets partially ordered by inclusion.

□

Typically we will use *discrete linear time* as our base:

$$I_\omega = \{\varnothing, \{0\}, \{0,1\}, \{0,1,2\}, \ldots\} \cup \{\omega\}\ .$$

An object O consists of observations over a base $\underline{\mathrm{T}}$. Given an inclusion $U \subseteq V$ and an observation $f : V \to A$ we can form the restriction of f to U and we will denote it by $f \restriction U$. Let $O(U)$ denote the observations over U. Given the inclusion $i : U \rightarrowtail V$, we can form the map $O(i) : O(V) \to O(U)$ that maps an observation f over V to its restriction $f \restriction U$. O now satisfies the following two equations $O(i\,;j) = O(j)\,;O(i)$, $O(1_U) = 1_{O(U)}$, that make it into a contravariant functor.

Definition 2
A *presheaf* is a functor $O : \underline{\mathrm{T}}^{op} \to \underline{\mathrm{C}}$ where $\underline{\mathrm{T}}$ is the *base category* and $\underline{\mathrm{C}}$ is the *structure category*. If $i : U \rightarrowtail V$ is in $\underline{\mathrm{T}}$, then $O(i) : O(V) \to O(U)$ is called the *restriction* morphism induced by i. □

The idea that observations are closed under restriction can be seen as a categorical formulation of the *prefix closure* property of traces. A new "larger" observation must contain all previous observations; it cannot contradict the past behaviour of the system.

The structure category is the domain of the attributes. By chosing a suitable category many different examples can be modeled, as described in [4]. In this paper the attributes will have a very simple form, which allows us to give the definition of a preobject in a simple form.

Definition 3
A *preobject* O over a base $\underline{\mathrm{T}}$ with *attribute object* $A = A_1 \times \cdots \times A_n$ is a presheaf of the form

$$O(U) = \{h : U \to A_1 \times \cdots \times A_n \mid K(h)\}$$

where the morphisms $O(i)$ are restriction maps, and the relation K expresses some property of the functions, embodying the "laws" that O satisfies; the elements of A may be called states. □

The definition of an object is the following [4].

Definition 4
An *object* is a preobject O such that its base $\underline{\mathrm{T}}$ is a topological space, and such that given $U_i \in \underline{\mathrm{T}}$ and $f_i \in O(U_i)$ for all $i \in I$ such that $U = \bigcup_{i \in I} U_i$ such that $f_i \upharpoonright U_i \cap U_j = f_j \upharpoonright U_i \cap U_j$ for all $i \in I$, then there exists a unique $f \in O(U)$ such that $f \upharpoonright U_i = f_i$ for all $i \in I$. This condition is called the *sheaf condition*. If the index set I is restricted to be finite, then the corresponding condition is called *finite sheaf condition*. □

Informally, this means that if observations are consistent over a subinterval, then these observations can be pasted together into an observation that contains them all as subobservations. Depending on whether one compares a pair or a possibly infinite set of observations, one speaks of a finite and an infinite sheaf condition respectively.

The basic idea in the categorical formulation of General Systems Theory [2, 3, 5] is that the behavior of a component is represented by a sheaf. The whole system is then represented by a diagram of these sheaves where the interconnection of the components is described by the diagram.

To transfer this analogy to Net theory, we need to decide what our domain of attributes and what our components will be. It seems reasonable to choose markings of places as the observables and transitions as components. Thus transitions will be modelled by sheaves. In the next section we will show how a net corresponds to a diagram of sheaves and the behavior of the net to the limit of the diagram.

Before we go into details, we need some definitions from Net theory and some notation. The standard reference for Net theory is [8]. The following is a slightly massaged definition of a Place/Transition net [7] without capacities.

Definition 5
A *Place/Transition net* N is a quadruple $\langle T, P, \iota, o \rangle$ where T is the set of transitions, P is the set of places, and $\iota, o : T \rightarrow [P \rightarrow N]$ are the input and output weight functions and N is the set of natural numbers. We will assume that T and P are finite sets. □

A Place/Transition (P/T) net is a graph with arcs labeled by integers. The standard graphical representation of a net pictures places as circles and transitions as lines. The ι and o functions are represented by labeled arcs between the transitions and places. Figure 1 shows a net consisting of a single transition and three places. It is customary to assume that the functions ι and o are injective so as to rule out transitions with identical sets of input and output places. In the terminology of [1] P/T nets are "extensional transition systems".

The following notations will be useful, where $t \in T$ and $p \in P$. The *preset* or set of *input places* of a transition t is defined as ${}^\bullet t = \{p \mid \iota(t)(p) \geq 1\}$. The *postset* or the set of *output places* of a transition t is defined as $t^\bullet = \{p \mid o(t)(p) \geq 1\}$. Finally the restrictions of the input and output weight functions ι, o to the pre- and postsets are defined as $\iota_t = \iota(t) \restriction {}^\bullet t$ and $o_t = o(t) \restriction t^\bullet$ respectively.

The intuition behind P/T nets is that a place can be occupied by tokens. The distribution of tokens in the net represents the state of the system and it is called a *marking*. If the number of tokens in the input places of a transition exceeds the input weight of that transition, then the system can change state by *firing* that transition. This process is formally defined through the firing rule:

Definition 6
Given a P/T net $N = \langle T, P, \iota, o\rangle$, a *marking* M is a function $M : P \to N$ mapping places to integers. In a net N we say that a transition $t \in T$ is *enabled at* M iff $\iota(t) \leq M$. If a transition is enabled it can fire, producing a new marking through the *firing rule*:

$$M' = M - \iota(t) + o(t),$$

where the addition and subtraction of integers is extended to addition and subtraction of integer valued functions by defining $(f+g)(x) = f(x) + g(x)$ and $(f-g)(x) = f(x) - g(x)$. □

If we want to observe the behaviour of a transition, then the only observable values are the markings on the places. This gives the following notion of preobject:

Definition 7
Given a transition t in a P/T net we define a preobject over $t : I_\omega \to Set$ as follows:

$$t(U) = \left\{ f : U \to N^{|{}^\bullet t| + |t^\bullet|} \mid f(i+1) = f(i) - \iota_t + o_t \quad \text{if } \iota_t \leq f(i) \right\},$$

with $i, i+1 \in U$ and $U \in I_\omega$. □

The definition describes those observations that arise from the firing of a transition. It gives a relational semantics to a transition, saying that the

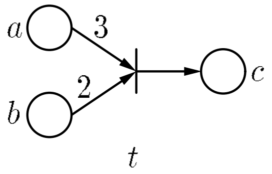

Figure 1: A transition t.

markings at times i and $i+1$ are related by the firing equation if $\iota_t \leq f(i)$. It is a very weak restriction on the relationship between the input and output places.

Example 1
Take the transition in figure 1. The corresponding sheaf is:

$$\begin{aligned} t_1 = \{ & f : I \to N \times N \times N \ \mid \\ & f(i+1) = f(i) - (2,3,0) + (0,0,1) \quad \text{if } (2,3,0) \leq f(i) \} \end{aligned}$$

□

One possible sequence of observations is

i	0	1	2	3	4
f	$(0,0,0)$	$(1,0,1)$	$(2,3,0)$	$(0,0,1)$	$(0,0,1)$

where the transition fires at $i = 2$.

The above example does not include capacities on the places, but it is not difficult to see how they could be incorporated.

A more interesting example is given by nets with time. There are several different ways for incorporating time into nets. We have chosen to attach an auxiliary place to each transition that expresses the number of clock ticks it takes for the transition to fire. This is a very simple model of nets with time. A more elaborate model could have a time interval associated with duration of the firing or one could attach a time interval to the transition that say within which time span the transition will fire after it has become enabled. To model the duration of the firing we add a stream that stores the firing time of the transitions:

Definition 8
Given a transition t in a P/T net with firing duration τ_t we can define a sheaf as follows (we abbreviate $n = |{}^\bullet t| + |t^\bullet|$).

$t(I) = \{ f : I \to N^n \times N \ \mid$

$$f(i+1) = \left\{ \begin{array}{ll} \langle f_{\{1,n\}}(i) - \iota(t), \tau_t \rangle & \text{if } \iota(t) \le f_{\{1,n\}}(i) \text{ and } f_{n+1}(i) = 0 \\ \langle f_{\{1,n\}}(i), f_{n+1}(i) - 1 \rangle & \text{if } f_{n+1}(i) > 1 \\ \langle f_{\{1,n\}}(i) + o_t, 0 \rangle & \text{if } f_{n+1}(i) = 1 \end{array} \right\}$$

$\square$

The idea behind this construction is that the $n+1$-st component stores the remaining time before the firing ends. The first clause of the case expression corresponds to the beginning of the firing of the transition. The tokens are removed from the input places and the timer is set. The second clause takes care of the countdown by decreasing the counter. After the firing time has expired the third clause outputs the required tokens to the output places and resets the timer to 0. Transitions for other kinds of nets can be treated in an analogous manner. Let us now look at the interconnection of transitions to form nets.

3 Nets, Systems and their Behavior

The intuition underlying this section is that the interconnection of objects should be achieved through morphisms. The system is represented as a diagram and the composite behavior of the system will arise as the limit of the diagram. The process of taking the limit selects mutually consistent behaviors of the subsystems into the behavior whole system. In this section we will show how the behavior of a Petri net arises as the limit of its corresponding sheaf diagram.

We start by defining a notion of morphism for pre-objects and objects:

Definition 9
Given preobjects O and O' over the same base $\underline{T}$, a *morphism* $\phi : O \to O'$ is a family $\phi_U : O(U) \to O'(U)$ of maps, one for each $U \in \underline{T}$, such that for each $i : U \to V$ in $\underline{T}$ the diagram

$$\begin{tikzcd} O(V) \arrow[r, "\phi_V"] \arrow[d, "O(i)"'] & O'(V) \arrow[d, "O'(i)"] \\ O(U) \arrow[r, "\phi_U"'] & O'(U) \end{tikzcd}$$

commutes in $\underline{C}$. When O and O' are objects, we may also call ϕ an *object morphism* or *sheaf morphism*. This gives rise to categories $\underline{\mathrm{PreObj}(\underline{T}, \underline{C})}$, and

$\underline{\mathrm{Obj}}(\underline{\mathrm{T}}, \underline{\mathrm{C}})$ of preobjects and objects respectively, over a given base $\underline{\mathrm{T}}$ and structure category $\underline{\mathrm{C}}$ □

A system consists of a diagram of pre-objects or objects.

Definition 10
A *system* $\mathcal{S}$ consists of a graph with nodes $n \in \mathrm{N}$ labeled by (pre-)objects S_n and with edges $e : n \rightarrow n'$ labeled by morphisms $\phi_e : S_n \rightarrow S_{n'}$. □

Morphisms can be used to represent sharing and inheritance of attributes between objects. In the sequel we shall mostly be interested in interconnecting objects through special morphisms called projection morphisms.

Definition 11
Given preobjects O and O' with attribute objects $A = \Pi_{j \in J} A_j$ and $A' = \Pi_{j \in J'} A_j$ respectively with $J' \subseteq J$, if $a : A \rightarrow A'$ sending $\langle a_j | j \in J \rangle$ to $\langle a_j | j \in J' \rangle$ induces a morphism, then it is called a *projection morphism.*

□

A projection morphism "picks" the necessary information from the attribute object. A typical use for projection morphisms is in connection with so called *event streams* $E = \{f \rightarrow A | M(f)\}$. An event stream contains the observations pertaining to an attribute. Examples of event streams include wires, communication channels, and places in Petri nets. The observations may be constrained by a condition $M(f)$ on the function f.

Given a system as a diagram, we can construct its behavior as the limit of the diagram. The limit of the diagram will contain all the mutually consistent behaviors of the system. The limit of a diagram can be understood as a completion of the diagram with an object that makes the whole diagram commute.

Definition 12
Given a diagram whose nodes are labeled by sheaves $S_n : \underline{\mathrm{T}}^{op} \rightarrow \underline{\mathrm{Set}}$ for $n \in \mathrm{N}$, select a fixed $U \in \underline{\mathrm{T}}$. Construct a new diagram D' where the nodes $n \in \mathrm{N}$ are labelled with $S_n(U)$ and edges $e : n \rightarrow n'$ are labelled with $\varphi_e : S_n(U) \rightarrow S_{n'}(U)$. Then the behavior object L of the diagram D' is

$$L(U) = \{\{f_n | n \in \mathrm{N}\} | f_n \in S_n \wedge (\phi_e : S_n \rightarrow S_{n'} \Rightarrow \phi_e(f_n) = f_{n'}\}$$

□

Theorem 1
$L(U)$ is the limit of the diagram D' in $\underline{\mathrm{Set}}$.

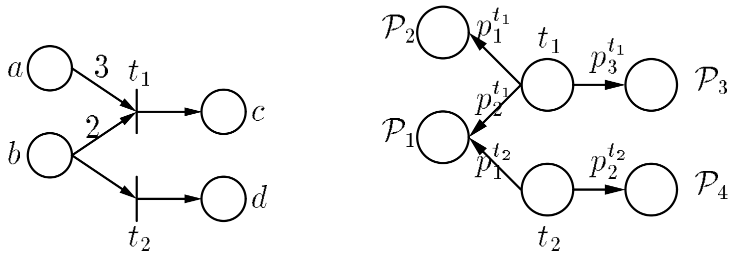

Figure 2: The net of the example and its corresponding diagram.

Proof:
$L(U)$ together with the projections $p_n : \{f_n | n \in \mathbf{N}\} \mapsto f_n$ form a cone over the diagram D'. Given some other cone $q : L'(U) \to D'$ with projections $q_1, \ldots, q_n$ the universal morphism $L'(U) \to L(U)$ is given by $\langle q_1, \ldots, q_n \rangle$.

□

To construct a net from a set of transitions, some way of expressing that two transitions are connected to the same place is needed. This is where event streams and projection morphisms come in handy. A place is represented by an event stream $P = \{f : I \to N\}$. If there is a capacity k attached to the place, then the event stream is of the form $\{f : I \to N | f(i) \leq k \;\; i \in I\}$. Each transition is connected to its input and output event streams by projection morphisms. The righthand side of figure 3 depicts the diagram obtained from the net on the lefthand side.

The limit object of this diagram is of the form

$$L = \{I \to N^9\} \,.$$

There will be projections $p_i : L \to P_i$ for $i \in \{1, 2, 3, 4\}$ and projection tuples $\langle p_5, p_6, p_7 \rangle : L \to t_1$ and $\langle p_8, p_9 \rangle : L \to t_2$. The limit object is isomorphic to the object $L' : I \to N^4$ which is the state space of the system.

Actually, the approach presented above is too simplistic, because it does not take into account the global nature of the notion of state. To see this consider the net in figure 3. For simplicity all the arcs have a weight of 1. Given the marking $M = \{a \mapsto 1, b \mapsto 1, c \mapsto 0, d \mapsto 0\}$, there is a choice or conflict between the firing of t_1, giving the marking $M' = \{a \mapsto 0, b \mapsto 0, c \mapsto 1, d \mapsto 0\}$, and the firing of t_2, resulting in the marking $\{a \mapsto 0, b \mapsto 0, c \mapsto 0, d \mapsto 1\}$. So from the marking $\{a \mapsto 1, b \mapsto 1, c \mapsto 0, d \mapsto 0\}$ it is impossible to get to the marking $\{a \mapsto 0, b \mapsto 0, c \mapsto 1, d \mapsto 1\}$. But if we take the limit of the corresponding sheaf diagram, the limit object will contain this marking, because we have no way of specifying the fact that firing t_1 and t_2 at the same time is an inconsistent behavior. Consequently the integrity of the tokens is destroyed.

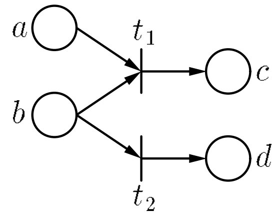

Figure 3: The net of the example.

To see how to approach a solution to this problem, let us first briefly look at the solution adopted in Net theory. The general assumption there is that a system can resolve a conflict by itself. A consequence of this is that a semantics for Petri nets must define an execution policy is some way, explicitly or implicitly. The firing rule for P/T nets establishes such a policy. It says that one should look at one transition at a time to see if it is enabled. During this process of examination each enabled transition is fired. The resulting reachability graph gives an interleaving semantics for the system, where at each instant only one transition is allowed to fire. What the firing rule does not say is in what order the transitions are to be examined.

The net effect of this is that we have a global state and we are able to observe the effect of a transition on the global state. As defined in Definition 7, the firing rule is "distributed" into each sheaf, and thus gains a local character resulting in the unfaithful rendering of the semantics of choice.

To rectify this problem we need to introduce a global arbiter, scheduler or observer. This sheaf will be connected to each transition of the system and allows only one transition to fire at each instant.

We use the shorthand $T_\perp$ for $T \cup \{\perp\}$.

Definition 13
Given a set of transitions T define an observer sheaf by:

$$T(I) = \{I \rightarrow T_\perp\} \; .$$

The possible observations are constrained by the rules:

$$\begin{array}{ccl} \perp & \mapsto & \perp \\ t & \mapsto & \perp \text{ with } t \in T \\ \perp & \mapsto & t \text{ with } t \in T \end{array}$$

where the lhs is the observation $f(i)$ and the rhs is the observation $f(i+1)$.

□

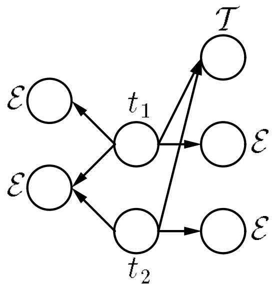

Figure 4: The diagram corresponding to the net in figure 3.

The sheaf T is used to store the name of the transition that is currently firing, and the token $\perp$ is used to denote that no transition is in the process of firing. The first rule expresses the fact that it is not necessary for any transition to fire; the system can idle. The second rule says that the firing process is over in one time step. The last rule is applicable when a transition starts to fire.

The definition of a transition sheaf as given in definition 7 has to be modified to accommodate the observer.

Definition 14
Given a transition t in a P/T net we define a sheaf as follows:

$$t(I) = \{f : I \to T_\perp \times N^{|{}^\bullet t|+|t^\bullet|} \mid$$
$$f(i+1) = \begin{cases} \langle t, f_{\{2,n\}}(i)\rangle & \text{if } \iota_t \leq f_{\{2,n\}}(i) \text{ and } f_1(i) = \perp \\ \langle \perp, f_{\{2,n\}}(i) - \iota_t + o_t\rangle & \text{if } f_1 = t \end{cases} \Bigg\}$$

□

The definition arranges things so that firing is only possible if the attribute T has the value $\perp$. The first clause insures that transition will only fire if no other transition is firing, while the second clause does the actual firing. Notice that this definition does not rule out the possibility that the firing process takes an infinite amount of time. This possibility is ruled out by the the definition of the sheaf T which has a rule that says that the firing takes only one time tick.

Now the net in figure 3 is translated to the diagram in figure 4. The limit-object of this diagram will have of the form

$$\{f : I \to N \times N \times N \times N \times T_\perp \times (T_\perp \times N \times N \times N) \times (T_\perp \times N \times N)\},$$

where the first four N:s correspond to the $\mathcal{E}$:s in figure 3 the last two expressions in parenthesis to the transitions and the $T_\perp$ to $\mathcal{T}$. This limit-object is isomorphic to the following simpler object

$$\{f : I \rightarrow N \times N \times N \times N \times T_{\perp}\},$$

because the arrows in the diagrams represent projections. From the structure of the object it is easy to see that only one transition is able to fire at each instant. The limit object corresponds to what is more commonly called the state space of the net. From this object one can see the state of the system by projecting on all but the last component.

Some possible observations are:

i	0	1	2	3
f	$(0,0,0,0,\perp)$	$(1,1,0,0,\perp)$	$(1,1,0,0,t_1)$	$(0,0,1,0,\perp)$
g	$(0,0,0,0,\perp)$	$(1,1,0,0,\perp)$	$(1,1,0,0,t_2)$	$(0,0,0,1,\perp)$

A more interesting semantics can be obtained by coding the arbitration of choice locally. To simplify the presentation we will restrict our nets so that all places have a capacity of one and all arcs have a weight of one, making the nets essentially Elementary Net (EN) systems (cf. [9]). When talking about EN-systems, it is customary to call places conditions, and transitions events. Elementary Net systems are essentially a modern version of Condition/Event systems (cf. [8] chapter 2). The very basic form of the formalism makes it possible to study some of the more fundamental phenomena that arise in distributed and parallel systems. These phenomena include contact, conflict and concurrency.

The following two definitions set up the basic framework where we assume that we are given an elementary net with set of conditions C and set of events E:

Definition 15
A condition is represented by the sheaf:

$$C(I) = \{f : I \rightarrow E \cup \{\mathbf{true}, \mathbf{false}\}\},$$

which is governed by the rules:

$$\begin{array}{rcl} m & \mapsto & m' \text{ with } m, m' \in \{\mathbf{true}, \mathbf{false}\} \\ e & \mapsto & m \text{ with } e \in E \text{ and } m \in \{\mathbf{true}, \mathbf{false}\} \\ m & \mapsto & e \text{ with } e \in E \text{ and } m \in \{\mathbf{true}, \mathbf{false}\} \end{array}$$

where the lhs is the observation $f(i)$ and the rhs is the observation $f(i+1)$.

□

The constants **true** and **false** are used to tell whether the condition holds or not. The name of the event is used to mark which event is occurring.

Definition 16
An event is modeled by the sheaf:

$$E(I) = \{f : I \to (e \cup \{\mathbf{true}, \mathbf{false}\})^{|{}^\bullet e|+|e^\bullet|},$$

which is governed by the transition rules:

$$\begin{array}{rcl} \langle x_1, \ldots, x_n \rangle & \mapsto & \langle x'_1, \ldots, x'_n \rangle \text{ with } |\langle x_1, \ldots, x_n \rangle| = |\langle x'_1, \ldots, x'_n \rangle| = n \\ & & \text{and with } e \neq e' \text{ for all } e' \in \{x_1, \ldots, x'_n\} \\ \langle \mathbf{true}^k, \mathbf{false}^l \rangle & \mapsto & \langle e^k, e^l \rangle \\ \langle e^k, e^l \rangle & \mapsto & \langle \mathbf{false}^k, \mathbf{true}^l \rangle \end{array}$$

where $k = |{}^\bullet e|, l = |e^\bullet|$ and $n = k + l$ and where the lhs is the observation $f(i)$ and the rhs is the observation $f(i+1)$. □

An event is represented by the product of its pre- and post-conditions. An event can only occur when all of its preconditions hold and all of its post-conditions don't (second rule, lhs). This ensures that events cannot occur when there is contact. When an event occurs it replaces all pre- and post-conditions with its name (second rule, rhs). This prevents other events that are in conflict from occurring. This implements conflict resolution on a local level. The lack of a global scheduler also means that causally independent transitions may fire in parallel. A net is now constructed by interconnecting event and condition sheaves with morphisms to a diagram. The limit object of this diagram gives a non-interleaving semantics for the net.

4 Conclusions

In this paper we have given a sheaf semantics for Place/Transition Systems and Elementary Net Systems. The two semantics have distinct flavors; the semantics for Place/Transition systems is an interleaving semantics, while the semantics for Elementary Net Systems is a non-interleaving semantics. This indicates that the "Objects as Sheaves" paradigm is powerful enough to be used in the study of some of the more fundamental properties of concurrency. Also it is interesting to note, that although we work with linear time, it is possible to obtain a non-interleaving semantics.

Acknowledgements I am grateful to Prof. Joseph Goguen for encouragement and comments on the manuscript.

References

[1] Corradini, A. *An algebraic semantics for transition systems and logic programming.* Technical Report TD-9/90. Università degli Studi di Pisa, Dipartimento di Informatica, 1990.

[2] Goguen, J. *Categorical foundations for general systems theory.* Advances in Cybernetics and Systems Research. Transcripta Books, 1973, pp. 121–130.

[3] Goguen, J. *Objects.* International Journal of General Systems, 1(1975), pp. 237–243.

[4] Goguen, J. *Sheaf semantics for concurrent interacting objects.* Mathematical Structures in Computer Science, 11(1992).

[5] Goguen, J. and Ginali, S. *A categorical approach to general systems theory.* Applied General Systems Research. Plenum, 1978, pp. 257–270.

[6] Gray, J. *Fragments of the history of sheaf theory.* Applications of Sheaves. Lecture Notes in Mathematics 753. Springer Verlag, Berlin, 1979, pp. 1–79.

[7] Reisig, W. *Place/transition systems.* Petri Nets: Central Models and Their Properties. Lecture Notes in Computer Science 254. Springer Verlag, Berlin, 1986, pp. 117–141.

[8] Reisig, W. *Petri nets: An introduction.* Springer-Verlag, 1985.

[9] Thiagarajan, P. *Elementary net systems.* Petri Nets: Central Models and Their Properties. Lecture Notes in Computer Science 254. Springer Verlag, Berlin, 1986, pp. 26–49.

[10] Wolfram, D. A. and Goguen, J. A. *A sheaf semantics for FOOPS expressions.* Proceedings, ECOOP'91 Workshop on Object-based Concurrent Computation. 1991.