TECHNICAL
RESEARCH
REPORT

SYSTEMS
RESEARCH
CENTER

*Supported by the
National Science Foundation
Engineering Research Center
Program (NSFD CD 8803012),
Industry and the University*

UPN: A Petri Net Based Graphical Representation of Company Policy Specifications in CIM

*by G. Harhalakis, C.P. Lin, and
P.R. Muro-Medrano*

TR 92-16

UPN: A Petri Net Based Graphical Representation of Company Policy Specifications in CIM

G. Harhalakis, C.P. Lin
Systems Research Center
University of Maryland

P.R. Muro-Medrano
Electrical Engineering and Computer Science Department
University of Zaragoza, Spain

Abstract

A graphical representation schema - Updated Petri Nets (UPN) - has been developed to model rule based company policy specifications, in the context of computer integrated manufacturing systems. UPN facilitates the modeling of relationships between operations of various related application systems and the database updates and retrievals among various CIM databases. Based on this representation, a hierarchical modeling technique which includes refining and aggregating rules has also been developed. Application of the UPN is demonstrated in desinging rule based systems for controlling and integrating the information between manufacturing applications, including Computer Aided Design, Computer Aided Process Planning, Manufacturing Resources Planning, and Shop Floor Control.

1 Introduction

In a modern factory, besides *parts* being produced, there is also a tremendous amount of *data* being processed. For an efficient operation, it is necessary not only to control the manufacturing processes of products but also to manage and control the information flow among all the computerized manufacturing application systems that exist in a modern factory. The emphasis of most of the previous and current research projects is on individual aspects of CIM, such as developing a generic CIM architecture, creating a global database framework, or interfacing shop floor activities. However, the future in automation of modern factories will be based on a distributed environment which needs not only a generic database framework but also a controller, usually a knowledge rule-based system, to control the relationships between activities within all computerized manufacturing application systems. Our approach is to develop such a control mechanism, in the form of a rule based system, and to fill the gap between the high level production management and the low level factory automation [HARH 90] [LIN 91].

Petri Nets [PETE 81] [MURA 89], which waere adopted in the development of the UPN, are ideal for modeling dynamically and formally analyzing complex dynamic relationships of interacting systems. They were initially developed and used mainly for advanced computer integrated systems design, both in hardware and software, such as artificial intelligence in network systems [COUR 83], and for flexible manufacturing systems [CROC 86]. Most recent applications of Petri Nets in manufacturing systems are focusing again on the shop floor level, with a large number of work stations, robots, and transportation systems, to be handled by a central controller. Colored Petri Nets [KAMA 86] allow the model designer to work at different aggregation levels. The main advantage of Colored Petri Nets over General Petri Nets is the possibility of obtaining a compact representation of a large and complex system.

This paper is structured as follows. The second section defines the problem domain and research approach. Third section describes the features of UPN. The fourth section discuses UPN validation capabilities. The last section presents our conclusions with recommendations for future work.

2 Company Policy Specification for CIM

The objective of developing UPN is to be able to model and validate a set of complex rules and procedures that constitute a company policy and to apply it in a Computer Integrated Manufacturing environment. This paper, aiming at linking product and process design, manufacturing operations and production management, focuses on the control of information flow between each of the key manufacturing applications at the factory level, including Computer Aided Design (CAD), Computer Aided Process Planning (CAPP), Manufacturing Resource Planning (MRP II), and Shop Floor Control (SFC) systems. This linkage between manufacturing application systems involves both the static semantic knowledge of data commonalities and the dynamic control of functional relationships. The common data entities, which form the basis of the integrated system, include: *Parts, Bills of Material* in CAD, *Parts, Bills of Material, Work Centers, Routings* in CAPP, *Parts, Bills of Material, Routings, Work Centers, Manufacturing Orders* in MRP II, *Parts, Routings, Work Centers, Manufacturing Orders* in SFC. The functional relationships deal with the inter-relationships of functions within those applications.

An example, which represents the release of a work center in MRP II, is explained in a natural language. Invoking the work center release transaction in MRP II triggers a set of consistency checks, which are as follows: the WC I.D. provided must exist in MRP II with a hold status; all the required data fields should have been filled, and any data fields left out by users are requested at this stage. If all these checks are satisfied, the system changes the work center status code from 'hold' to 'released', and a skeletal work center record is automatically created in the work center file in CAPP, with its status set to 'working'.

The next step is to convert the UPN model into a set of General Petri Nets (GPN) for validation purposes, and feed the results back to the user to resolve (i) conflicting company rules and (ii) errors introduced during the modeling phase. After the model has been validated, a parser translates the UPN model into a rule specification language. In short, the input is a set of company rules and the output is an AI production system for controlling operations, accessibility and updates of data within the manufacturing applications involved.

3 Updated Petri Nets

We have developed the Updated Petri Nets (UPN), which is a specialized type of the Colored Petri Nets (CPN) [JENS 87], and a hierarchical modeling methodology with a systematic approach for the synthesis of separate nets. The use of UPN allows the model designer to work at different levels of abstraction. Once we have this net we can selectively focus the analysis effort on a particular level within the hierarchy of a large model.

3.1 UPN: Definition and Example

An UPN is a directed graph with three types of nodes: places which represent facts or predicates, primitive transitions which represent rules or implications, and compound transitions which represent meta-rules (sub-nets). Enabling and causal conditions and information flow specifications are represented by arcs connecting places and transitions.

Formally, an UPN is represented as: $UPN =< P, T, C, I^-, I^+, M_0, I_o, MT >$, where:

1. P, T, C, I^-, I^+, M_0 represent the classic Color Petri net definition. They identify the parts of the information system that provide the conditions for the control of information flow. Only this part of the UPN net is used in the validation process. These terms are defined as follows [JENS 87]: $P = \{p_1, ..., p_n\}$ denotes the set of places (circles) and $T = \{t_1, ..., t_m\}$ the set of primitive transitions (black bars), where $P \bigcap T = \emptyset$ and $P \bigcup T \neq \emptyset$. C is the color function defined from $P \bigcup T$ into non-empty sets. It attaches to each place a set of possible token-data and to each transition a set of possible data occurrences. I^- and I^+ are negative and positive incidence functions defined on $P \times T$, such that $I^-(p,t), I^+(p,t) \in [C(t)_{MS} \rightarrow C(p)_{MS}]_L \ \ \forall (p,t) \in P \times T$ where S_{MS} denotes the set of all finite multisets over the non-empty set S, $[C(t)_{MS} \rightarrow C(p)_{MS}]$ the multiset extension of $[C(t) \rightarrow C(p)_{MS}]$ and $[\ldots]_L$ denote a set of linear functions. The net has no isolated places or transitions. The initial marking, M_0, is a funtion defined on P, such that: $M_0(p) \in C(p), \forall p \in P$.

2. I_o is an inhibitor function defined on $P \times T$, such that: $I_o(p,t) \in [C(t)_{MS} \rightarrow C(p)_{MS}]_L, \ \ \forall (p,t) \in P \times T$.

3. $MT = \{mt_1, ..., mt_l\}$ denotes the set of compound transitions (represented graphically as blank bars); these are transitions which will be refined into more detailed subnets.

We have divided the representation of the domain knowledge in the following four groups: *Data, Facts, Rules, Metarules. Data* and relations between different data are used in relational database management systems. *Facts* are used to declare a piece of information about some data, or data relations in the system. The control of information flow is achieved by *Rules.* Here, we are considering domains where the user specifies information control policies using "if then" rules. Rules are expressed in UPN by means of transitions and arcs. Metaknowledge, in the form of metarules, is represented by net aggregation and hierarchical net decomposition (compound transitions).

The example shown in section 2, which represents the release of a work center in MRP II, is now modeled in UPN, as shown in figure 1 to illustrate the corresponding component of UPN.

Attibute	*Color set*	*DB data type*	*Description*
wcid	$WCID$	identification	identification number
des	DES	text	description
dep	DEP	text	department
cap	CAP	integer	capacity
sts	$MSTS$	$\{h, r\}$ (hold, release)	work center status code
ste	$MSTE$	$\{na, av\}$ (not avail., avail.)	work center state code
res	RES	text	Resource code
esd	ESD	date	Effectivity start date
Complete data structure for work center in MRP II			
$Mwc(wcid, des, dep, cap, sts, ste, res, esd)$			

Table 1: Data structures and color sets of work centers.

3.2 UPN: Hierarchical Modeling Approach

Generally speaking, any "company policy" starts from the specification of general global rules which describe aggregate operations for a given entity within the system. These rules are then further refined into more detailed specifications on a step by step basis, until no aggregate operations are left. Following a similar concept, a hierarchical modeling method using UPN has been developed which incorporates:

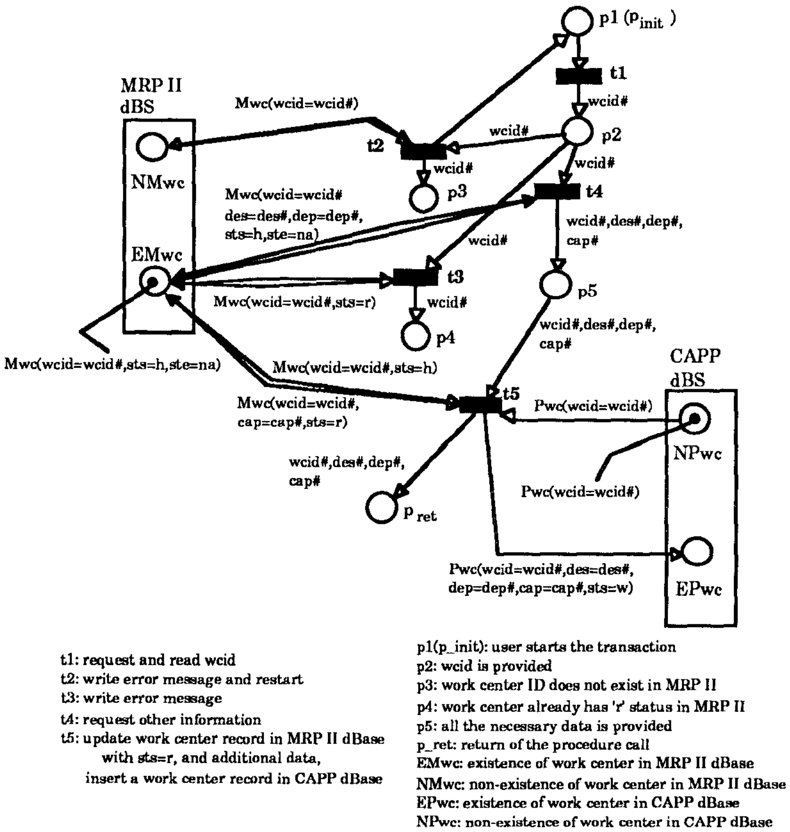

Figure 1: Subnet of the work center creation scenario "Release of a work center in MRP II".

Top-down stepwise refinement technique for the modeling of each scenario from an abstract and aggregate level to a detailed level. This approach necessitates the development of new Petri Net modeling entities which include two types of transitions as mentioned above; one to represent primitive rules, and the other to represent metarules which can be further refined into sub-nets. The connections are represented by calls from one compound transition of the net at the abstract level to the sub-nets at the more detailed level.

Synthesis technique for synthesizing separate nets, which represent different scenarios of the system, to form a coherent net. Our modeling approach is capable of incorporating the modeling of the databases of the manufacturing application systems involved by defining the database states as global variables and interfacing the application procedures (company policy) through the default modification procedure (system dependent) and places representing database states, and synthesizing nets through them systematically. More details can be found in [LIN 91]

4 Knowledge verification

One of the major objectives of creating a KBS using Petri Nets is the ability of validating the KBS mathematically and systematically. Several analysis methods [MURA 89] [MART 82] and validation techniques [NGUY 87], [LOPE 90] have already been developed. Typically they were developed for Generalized Petri Nets (GPN), and do not apply to Colored Petri Nets (CPN) which are characterized by a great diversity of linear functions that are associated to their arcs. Therefore, unlike analysis algorithms for GPN that use integer matrices, analysis algorithms for CPN need to manipulate matrices composed by linear functions. We have developed an *abstraction* and *unfolding* technique to convert UPN into GPN before they can be analyzed [LIN 91]. The first step is to perform an abstraction on the identification number of a database entity, which if unfolded will create a large number of duplicated nets, one for each possible value of its identification number. For example, unfolding the *wcid* of the work center record in MRP II will create a large number of nets, as many as the number of work centers which can be inserted into the database. In reality, we need only one net (for one "generic" work center) to represent the company policy for any work center. The second step is to apply the formal Colored Petri net unfolding algorithm on a UPN, which consists of the information that is aggregated in the color sets of its places and transitions, and the functions of its arcs. The result of this conversion process is an analyzable GPN.

5 Conclusions

A formal structured representation schema for rule based systems has been developed and demonstrated with information integration for manufacturing applications. The representation schema, called UPN, is based on the graphical and formal capabilities of colored Petri nets to express and validate if-them rules. The UPN is capable of representing user specification rules as well as database updates and retrievals, which is necessary for controlling and integrating information within current and future distributed database systems. Related rules can be aggregated at the same level of abstraction and the relation between one rule at a given level of abstraction and a set of aggregated rules at a lewer level of abstraction is also allowed. These facilities provide a mechanism for step wise refinement in modeling and validation. A rule base of the CAD/CAPP/MRP II/SFC integrated system has been developed using UPN and implemeted on a Sun Sparc station.

References

[COUR 83] M. Courvoisier, R. Vallette, et al. "A Programmable Logic Controller Based on A High Level Specification Protocol", *Proceedings of IECON Conference on Industrial Electronics, pp. 174-179*, 1983.

[CROC 86] Crockett, D.H. and Desrochers, A.A., "Manufacturing Workstation Control Using Petri-Nets", *Technical Report 83, Robotics and Automation Laboratory, Department of Electrical, Computer, and System Engineering, Rensselaer Polytechnic Institute*, August 1986.

[HARH 90] Harhalakis, G., Lin, C., Hillion, H., and Moy, K., "Development of a Factory Level CIM Model", *Journal of Manufacturing Systems, vol. 9, no. 2, pp. 116-128*,1990.

[JENS 87] Jensen, K., "Colored Petri Nets", *Petri Nets: Central Models and Their Properties. Advances in Petri Nets 1986, Part I. Proceedings of an Advanced Course, Bad Honnef, 8-19. September 1986, pp. 248-299*, Edited by G. Goos and J. Hartmanis. Springer-Verlag Berlin Heidelberg 1987.

[KAMA 86] Kamath, M. and Viswanadham, N., "Applications of Petri net based models in the modeling and analysis of flexible manufacturing systems". *Proc. IEEE Int. Conf. on Robotics and Automation., pp. 312-316*, 1986.

[LIN 91] Lin, C.P., "Design, Verification and Implementation of Rule Based Information System for Integrated Manufacturing", PhD Thesis, Department of Mechanical Engineering, University of Maryland, College Park, Maryland, 1991.

[LOPE 90] Lopez B., Meseguer P. and Plaza E.: "Knowledge Based Systems Validation: A State of the Art". *AI Communications, Vol.3, No. 2, pp. 58-72*. June, 1990.

[MART 82] Martinez, J. and Silva, M." A Simple and Fast Algorithm to Obtain All Invariants of A Generalised Petri Net",. *Second European Workshop on Application and Theory of Petri Nets, pp. 301-310*, 1982.

[MURA 89] Murata, T., "Petri Nets: Properties, Analysis and Applications", *Proceedings of The IEEE, vol. 77, no. 4, pp. 541-580*, April 1989.

[NGUY 87] Nguyen T.A., Perkins W.A., Laffey T.J. and Pecora D.: "Knowledge Base Validation". *AI Magazine, summer, pp. 67-75*. 1987.

[PETE 81] Peterson, J.L., "Petri Net Theory and the Modeling of Systems", Prentice Hall, Englewood Cliffs, New Jersey, 1981.