Advanced Studies in Contemporary Mathematics
29 (2019), No. 2, pp. 179 – 189

www.jangjeon.or.kr
http://dx.doi.org/10.17777/ascm2019.29.2.179

GENERALIZED NET REPRESENTATIONS OF ELEMENTS OF SERVICE SYSTEMS THEORY

VELIN ANDONOV, STOYAN PORYAZOV, EMILIYA SARANOVA

Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Acad. G. Bonchev Str., bl. 8, 1113 Sofia, Bulgaria
E-mails: velin_andonov@math.bas.bg, stoyan@math.bas.bg,
emiliya@cc.bas.bg

Abstract. A use of base and fundamental model concepts for representation of constructions from the languages for computer modeling and simulation is proposed. These concepts are used for representation with the apparatus of the Generalized nets of widespread modeling objects which are used in other known approaches to the conceptual modeling of service systems. The results allow for a comparison of different languages and would make the use of the perspective method of the Generalized nets easier.

2010 Mathematics Subject Classification. 68Q85, 68M10, 90B22.

Keywords and phrases. service systems theory, generalized nets, conceptual modeling.

1. Introduction

In this section the base and fundamental model concepts are discussed which are necessary for the definition and comparison of modeled objects in the contemporary languages for computer modeling and simulation. The Generalized Nets (GNs) (see [2]) are a perspective method for conceptual modeling [11] of complex systems. For this reason in the present paper GNs are used for representation of widespread modeling objects which are often used in other languages for computer modeling and simulation. These objects are Generator, Terminator, Transportation (simple transition), Delay, Information gathering, Unifying Transition, Distributive Transition and Queue.

1.1. **A base for comparison of systems for modelling of service systems.** The conceptual models of service systems are built of concepts on several levels. The lowest level consists of elementary (basic) concepts for which it is assumed that they do not contain other basic concepts. The

The work of V. Andonov was supported by the Bulgarian NSF Project DM 12/2 - "New Models of Overall Telecommunication Networks with Quality of Service Guarantees". The work of E. Saranova was supported by the National Scientific Program "Information and Communication Technologies for a Single Digital Market in Science, Education and Security (ICT in SES)", financed by the Bulgarian Ministry of Education and Science.

defining of basic concepts is necessary in the process of comparison of different languages for informatical modeling, as well as for the design of new such languages.

In the present paper, base and fundamental concepts are considered which can be used for comparison of the various approaches to the construction of conceptual models of service systems. A comparison of two approaches is presented. On one hand, the GNs approach (see [2]) and, on the other hand, the classical languages for computer modeling such as GPSS [5], SIMSCRIPT [6] and their successers SLX [8] and SIMSCRIPT III [12], as well as and newer ones such as Ptolemy II [7].

1.2. **Characteristic and state.** For all model concepts used in the informatical modeling there are specific characteristics which are used for representation of the model concepts. Characteristics can be qualitative and quantitative. In both cases the characteristics are represented in the languages for computer modeling through parameters. For instance, in a given model concept the quality "volume" can have the values "exists" and "does not exist". The qualitative parameter "volume" (if it is pointed out as existing) can have as a value a single non-negative number. In this sense, every characteristic in the general case can have qualitative and quantitative state represented explicitly or not. All characteristics of a modeling object in one moment of the modeling time we call object's current state.

1.3. **Base model concepts.** We assume that the base model concepts for modeling of dynamical service systems are generally devided into two types: entities and changes.

Entities have roles, for example requests (call attempts, bids), device, characteristic, condition, objective, command and many more.

Each change is a modification of a state of an entity and can be reduced to two modifications (events) – generation and termination of an existing object in the model. Changes are regarded as operations and processes which are carried out at the time when a request for service enters a device. Examples of such devices are generator, terminator and server.

Requests are the objects which are serviced in the devices and are usually assumed movable in the process of simulation. The service is often interaction – the states of the requests and the characteristics of the devices are modified. Requests can have names such as token, call attempt, call [4], transaction (GPSS) etc. Requests can be point, local and distributed. Point requests can occupy a single device during one moment of time while the distributed requests usually occupy more than one device. Examples of point requests are the postal letters, internet data packets etc. Examples of distributed requests are the telecommunication requests (in systems with channel switching and messages), fluids (gases, liquids) considered as requests etc.

The characteristics of the requests include: locality and quantity of the occupied place by them (traffic, [4]) in the servicing devices; next objective operation depending on the state of the request and the objective device.

The difference between the objective and the occupied place by one request in one device is expressed through qualifiers of the term "traffic" depending on the objective and the result of the occupation – "demand", "offered", "carried", "lost" and others [4]. The maximum size of the spase used by the requests in a single device is called capacity of the device.

From the above it is evident that the notions "spase" and "time" are a base for individual and common characteristics of the requests for the devices and the changes, i.e. they are fundamental concepts in the modeling.

1.4. **Fundamental modeling concepts.** Fundamental modeling concepts are those which are used for defining of modeled objects and which are common for the modeled and the modeling system. Such concepts are spase, time and order which are closely related.

The notions "path" and "direction" of requests belong to the characteristics of the spase. They are usually represented graphically by lines and arrows. The movement of the requests and the direction of the movement can be changed by different devices of type "transition".

In every computer model requests are ordered by various criteria – time (of generating, arrival etc), priority, decision taken by the modeller, random order etc. The order of the requests can be determined explicitly for the devices of type queue.

Each change in the state is made in moments of time which are ordered (before, after, simultaneously). Simulation systems differ from each other in the models for representation of time [7].

In the present paper a comparison of modeling objects is made using the mentioned above base and fundamental modeling concepts.

2. Generalized net representations of elements of Service Systems Theory

The theory of GNs [2] and their applications [3] provide an alternative approach to the conceptual modeling of telecommunication systems. Here, for the first time, a more systematical study of GN representations of elements of the Service Systems Theory is presented. We consider the following elements:

- Generator;
- Terminator;
- Transportation (simple transition);
- Delay;
- Server;
- Information gathering;
- Unifying Transition;
- Distributive Transition;
- Queue.

The following standard numerical characteristics and standard logical characteristics of the devices:

- Number of call attempts;
- Coefficient of usage;
- Capacity;

- Volume and Intensity of the device traffic;
- Intensity of the incoming flow into the device;
- Free or Occupied device (standard logical characteristic of the device);
- Offered traffic;
- Mean time of service of the call attempts;

are represented by characteristics of special tokens in the GN.

2.1. **Generator.**

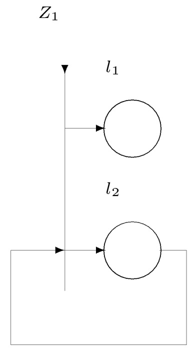

Fig. 1. Generalized net representation of Generator.

The function of the Generator is to create requests, for instance call attempts. Its execution does not increase the model time. The device has characteristics: capacity and time interval between two consecutive requests – usually a pseudo random variable. The requests generated belong to a certain type.

In terms of GNs, the Generator can be represented by the transition in Fig. 1. Its formal description is

$$Z_1 = \langle \{l_2\}, \{l_1, l_2\}, r_1 \rangle, \text{where } r_1 = \begin{array}{c|cc} & l_1 & l_2 \\ \hline l_2 & W_{2,1} & W_{2,2} \end{array}$$

and

- $W_{2,2}$ = "true";
- $W_{2,1}$="A request must be generated at the current time step".

In place l_2 a token stays permanently. All characteristics of the Generator described above must be kept in the token's characteristics. When the truth value of the predicate $W_{2,1}$ is "true" the token in place l_2 splits into two tokens one of which remains in place l_2 and the other enters place l_1 with characteristic *"Number of requests generated during the current step"*.

In this way, the requests generated during a single time step are represented by one token. This includes the case of ordinary flow, when one token represents one request. If we have more requests generated during a single time step and we want different requests to be represented by different tokens, we should add more output places like l_1 in Fig. 1 and the token in place l_2 splits into more tokens each of which corresponds to a different request.

2.2. **Terminator.**

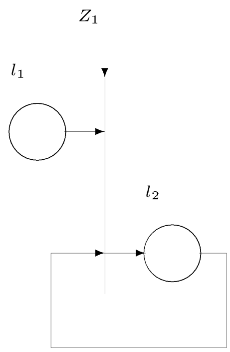

Fig. 2. Generalized net analogue of Terminator.

The Terminator removes the requests from the model. It has capacity. The duration of its execution is 0, i.e. it does not change the model time.

A GN representation of the Terminator is shown in Fig. 2. The requests that should be terminated are represented by tokens in place l_1.

$$Z_1 = \langle \{l_1, l_2\}, \{l_2\}, r_1, \square_1 \rangle \text{, where } r_1 = \begin{array}{c|c} & l_2 \\ \hline l_1 & W_{1,2} \\ l_2 & W_{2,2} \end{array}$$

and

- $W_{1,2}$ = "The current request should be terminated";
- $W_{2,2}$= "true".

$\square_1 = \wedge(l_1, l_2)$.

There is a token which stays permanently in place l_2. When the truth value of the predicate $W_{1,2}$ is "true" the token in place l_1 representing the requests that should be removed from the model enters place l_2 and merges with the token there. The new token does not obtain new characteristic. The transition becomes active only if there is at least one token in each of the places l_1 and l_2.

Another way to represent the Terminator in terms of GNs is by using an output place of the net, i.e. place with no outgoing arc. When tokens enter such places they leave the net.

2.3. **Transportation.**

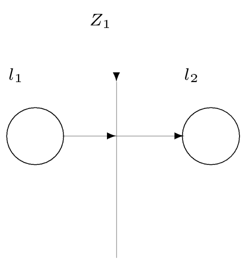

Fig.3. Generalized net representation of Transportation.

The function of the Transportation is to represent the movement of objects from one part of the model to another. It reflects the dynamics of the modeled process. A GN representation of Transportation is shown in Fig.

3. The places of the transition represent two different parts of the modeled system for which there is a flow of information in one direction – from the input place l_1 to the output place l_2.

$$Z_1 = \langle \{l_1\}, \{l_2\}, r_1, \square_1 \rangle, \text{where } r_1 = \begin{array}{c|c} & l_2 \\ \hline l_1 & W_{1,2} \end{array}$$

and

- $W_{1,2}$ =“The current token in place l_1 must be moved to l_2.”

The transition's type $\square_1$ is such that the transition becomes active if there is at least one token in place l_1. When the truth value of the predicate $W_{1,2}$ is “true” the token in place l_1 enter place l_2. The new coordinates of the object represented by the token are preserved in the characteristic obtained by the token in l_2.

2.4. **Delay.**

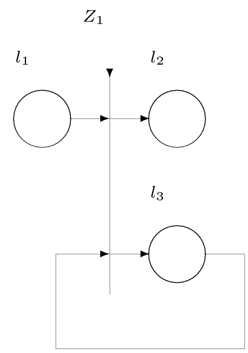

Fig. 4. Generalized net representation of Delay.

The Delay is used to represent situation in the modelled process when requests must wait for a certain period of time until the condition for their transfer is satisfied. The GN representation of the Delay is shown in Fig. 4. The tokens in place l_1 representing requests that must wait enter place l_3. They stay there until the condition for transfer is satisfied. The evaluation of the truth value of the predicate corresponding to the transfer of the tokens from input to output places does not change the model time, i.e. it is performed outside of it.

$$Z_1 = \langle \{l_1, l_3\}, \{l_2, l_3\}, r_1 \rangle, \text{where } r_1 = \begin{array}{c|cc} & l_2 & l_3 \\ \hline l_1 & W_{1,2} & W_{1,3} \\ l_3 & W_{3,2} & W_{3,3} \end{array}$$

and

- $W_{1,2}$ = “The current token should not wait”;
- $W_{1,3}$= “The current token must wait for a certain number of steps”;
- $W_{3,2}$=“The current token can be transfered”;
- $W_{3,3} = \neg W_{3,2}$.

When the truth value of the predicate $W_{1,2}$ is “true” the token from place l_1 enters l_2. The characteristic which it obtains there depends on the modeled process. When the truth value of $W_{1,3}$ is “true” the token from l_1 enters l_3 where it obtains the characteristic *“condition for further transfer”*. This characteristic can be the number of time steps that the token must wait or another condition for transfer which should be checked on the next steps to determine the waiting period for the current token. When the truth value of the predicate $W_{3,2}$ is "true" the current token in l_3 enters l_2 where it obtains a characteristic that depends on the modeled process. When the truth value of the predicate $W_{3,3}$ is "true" the current token stays in place l_3 and its next characteristic is *"condition for further transfer"*.

The GN representation can be used not only when requests must wait a given number of steps but also when their further transfer depends on some logical condition and the delay must be determined at any time step. For example, the delay could be as a result of accumulation of many events within the model.

2.5. **Server.**

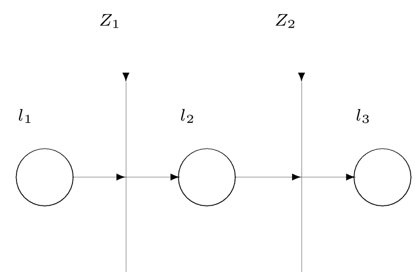

Fig. 5. Generalized net representation of Server.

The Server represents control or comparison of results with their standard or expected values. This includes checking of requests' quality or quantity parameters, control of the results of experiments, reading documents before taking a decision etc. A GN representation of Server is shown in Fig. 5. The change of the of the characteristics of the requests is modelled through the characteristic function of the places of the GN. The two transitions can be formally described in the following way:

$$Z_1 = \langle \{l_1\}, \{l_2\}, r_1 \rangle \text{, where } r_1 = \begin{array}{c|c} & l_2 \\ \hline l_1 & W_{1,2} \end{array}$$

and

- $W_{1,2}$ = "The current token can be handled by the server";

When the truth value of the predicate $W_{1,2}$ is "true" the token enters place l_3 where it obtains the characteristic *"data related to the control of the process"*.

$$Z_2 = \langle \{l_2\}, \{l_3\}, r_2 \rangle \text{, where } r_2 = \begin{array}{c|c} & l_3 \\ \hline l_2 & W_{2,3} \end{array}$$

and

- $W_{2,3}$ = "A decision related to the control of the process is taken".

When the truth value of the predicate $W_{2,3}$ is true the current token enters place l_3 where it obtains the characteristic *"Decision taken about the current token"*.

2.6. **Information gathering.**

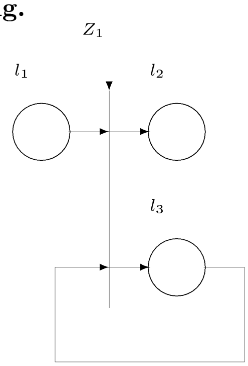

Fig. 6. Generalized net representation of Information gathering.

The Information gathering denotes the accumulation and storage of data obtained within the model. In particular it describes the passive storage of data. The GN representation of the Information gathering is shown in Fig. 6. The information that must be stored is preserved in the form of characteristics of a token which stays permanently in place l_3. Its initial characteristic is *"Description of the types of data which will be stored"*.

$$Z_1 = \langle \{l_1, l_3\}, \{l_2, l_3\}, r_1, \square_1 \rangle, \text{where } r_1 = \begin{array}{c|cc} & l_2 & l_3 \\ \hline l_1 & W_{1,2} & W_{1,3} \\ l_3 & W_{3,2} & W_{3,3} \end{array}$$

and

- $W_{1,2}$ = "true";
- $W_{1,3}$= "The characteristics of the current token must be stored.";
- $W_{3,2}$="false";
- $W_{3,3}$= "true".

$$\square_1 = \wedge(l_1, l_3).$$

When the truth value of the predicate $W_{1,3}$ is "true" the current token in place l_1 splits into two identical tokens one of which enters place l_2 without obtaining new characteristic and the other one merges with the token in place l_3. The token in place l_3 obtains together with its previous characteristic those of the characteristics of the token coming from l_1 which must be stored.

2.7. **Unifying Transition.**

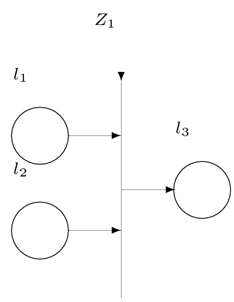

Fig. 7. Generalized net representation of Unifying Transition.

The Unifying Transition denotes two channels that merge to form a single channel. Its GN representation is shown in Fig. 7.

$$Z_1 = \langle \{l_1, l_2\}, \{l_3\}, r_1 \rangle, \text{where } r_1 = \begin{array}{c|c} & l_3 \\ \hline l_1 & W_{1,3} \\ l_2 & W_{2,3} \end{array}$$

and

- $W_{1,3}$ = "true";
- $W_{2,3}$= "true";

The tokens entering place l_3 merge to form a single token which preserves their characteristics.

2.8. **Distributive Transition.**

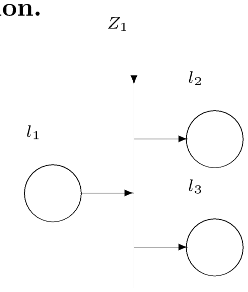

Fig. 8. Generalized net representation of Distributive Transition.

The Distributive Transition represents one channel that splits into two. Its GN representation is shown in Fig. 8.

$$Z_1 = \langle \{l_1\}, \{l_2, l_3\}, r_1 \rangle, \text{where } r_1 = \begin{array}{c|cc} & l_2 & l_3 \\ \hline l_1 & W_{1,2} & W_{1,3} \end{array}$$

and

- $W_{1,2} = W_{1,3}$;

When the truth value of the predicate $W_{1,2}$ is "true" the token in place l_1 splits into two identical tokens one of which enters place l_2 and the other one enters place l_3. They do not obtain characteristics in the output places.

2.9. **Queue.**

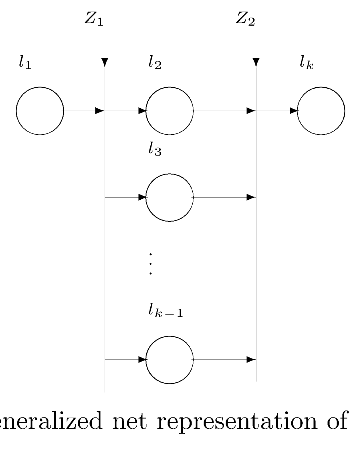

Fig. 9. Generalized net representation of Queue.

The Queue represents waiting lines. It is important to know where the information comes from and what is the management of the queue, i.e. FIFO, LIFO or some other rule. A GN representation of a Queue is shown in Fig. 9. The management of the Queue can be described in terms of GNs in different ways. For example, through the priorities of the places and the predicates of the transition's condition. If l_{k-1} has the highest priority among the output places of Z_1 and the input places of Z_2, the capacity of l_k is one and the tokens in l_k are transfered unconditionally to other places of the net (not shown in Fig. 9) and the transfer of tokens from the input places of Z_2 to l_k is also unconditional, then we have a GN analogue of FIFO. In a similar way a LIFO or some other management of the Queue can be described. The formal description depends on the modeled process.

Other GN representations of queuing systems in service networks are studied in [10, 13]

3. Conclusion

The results presented here allow for comparison of various possible representations of modeling objects in the languages for computer modeling and simulation. The GNs representations of the base elements of Service Systems Theory allow for an easier use of the promising method of the GNs.

The following problems for future research in this direction should be stated explicitly.

(1) In the present paper only a part of the used modeling objects are considered. It is necessary that the other used constructions be analyzed and represented in terms of the GNs.

(2) The important methodological problem for determining of base and fundamental model concepts is only pointed out here. A special and profound study of this problem is required.

The present paper is the first step towards the solving of these important problems.

References

[1] Andonov, V., Atanassov, K., Generalized nets with characteristics of the places. Compt. rend. Acad. bulg. Sci., 66, 2013, No. 12, 1673-1680.

[2] Atanassov, K., On Generalized Nets Theory. Prof. M. Drinov Academic Publ. House, Sofia, 2007.

[3] Atanassov, K., Applications of Generalized Nets. World Scientific Publ. Co., Singapore, 1993.

[4] ITU-T Recommendation E.600: Terms and Definitions of Traffic Engineering (Melbourne, 1988; revised at Helsinki, 1993).

[5] Gordon, G., A general purpose systems simulation program. In Proc. EJCC. Washington, D.C., Macmillan, NY, 1961, 87-104.

[6] Markowitz, H.M., Hausner B., W. Karr H., SIMSCRIPT: A Simulation Programming Language, Prentice Hall, Englewood Cliffs, N.J., 1962.

[7] Claudius Ptolemaeus (Editor). System Design, Modeling, and Simulation using Ptolemy II, Ptolemy.org, 2014. First Edition, Version 1.02, pp.690. (http://ptolemy.org/systems)

[8] Henriksen, J. O., An introduction to SLX. In: S. Andradyttir, K. J. Healy, D. H. Withers, and B. L. Nelson (Editors). Proceedings of the of the 29th Winter Simulation Conference, IEEE Computer Society, 1997, 559-566.

[9] Poryazov, S., Saranova, T., Some General Terminal and Network Teletraffic Equations in Virtual Circuit Switching Systems. Chapter 24 in: A.Nejat Ince, Ercan Topuz (Editors) Modeling and Simulation Tools for Emerging Telecommunications Networks. Springer Sciences+Business Media, LLC, USA 2006, 471-505.

[10] Poryazov, S., V. Andonov, E. Saranova, Comparison of four conceptual models of a queuing system in service networks. Proc. of the 26th National Conference with International Participation - TELECOM 2018, Sofia, 25-26 October, 2018, 71-77.

[11] Robinson, S., Brooks, B., Kotiadis, K., Van Der Zee, D., Conceptual Modeling for Discrete-Event Simulation. Taylor and Francis Group, LLC, CRC Press, 2011, ISBN -13: 978-1-4398-1038-5 (Ebook-PDF) 527 pages.

[12] Rice, S. V., Marjanski, A., Markowitz, H. M., Bailey, S. M., The SIMSCRIPT III Programming language for modular object-oriented simulation. In: Proceedings of the 2005 Winter Simulation Conference (IEEE Cat. No.05CH37732C), M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, eds., 621-630.

[13] Tomov, Z., Krawczak, M., Andonov, V., Dimitrov, E., Atanassov, K., Generalized net models of queuing disciplines in finite buffer queuing systems. Proc. of the 16th International Workshop on Generalized Nets, Sofia, 9-10 February, 2018, 1 - 9.