INTERNATIONAL STANDARD

ISO/IEC 15909-3

First edition
2021-11

Systems and software engineering — High-level Petri nets —

Part 3:
Extensions and structuring mechanisms

Ingénierie du logiciel et des systèmes — Réseaux de Petri de haut niveau —

Partie 3: Extensions et mécanismes de structuration

Reference number
ISO/IEC 15909-3:2021(E)

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*.

A list of all parts in the ISO/IEC 15909 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

Petri nets have been used to describe a wide range of systems since their invention in 1962. The technique is mathematically defined and can thus be used to provide unambiguous specifications and descriptions of applications. It is also an executable technique, allowing specification prototypes to be developed to test ideas at the earliest and cheapest opportunity. Specifications written in the technique can be subjected to analysis methods to prove properties about the specifications, before implementation commences, thus saving on testing and maintenance time and providing a high level of quality assurance.

A problem with Petri nets is the explosion of the number of elements in their graphical form when they are used to describe complex systems. High-level Petri nets were developed to overcome this problem by introducing higher level concepts, such as the use of complex structured data as tokens, and using algebraic expressions to annotate net elements. The use of “high-level” to describe these Petri nets is analogous to the use of “high-level” in high-level programming languages (as opposed to assembly languages), and is the usual term used in the Petri net community. Two of the early forms of high-level nets that this document builds on are predicate-transition nets and coloured Petri nets, first introduced in 1979 and developed during the 1980s. It also uses some of the notions developed for algebraic Petri nets, first introduced in the mid-1980s. It is believed that this document captures the spirit of these earlier developments (see Bibliography).

The technique has multiple uses. For example, it can be used directly to specify systems or to define the semantics of other less formal languages. It can also serve to integrate techniques currently used independently such as state-transition diagrams and data flow diagrams. The technique is particularly suited to parallel and distributed systems development as it supports concurrency. The technique is able to specify systems at a level that is independent of the choice of implementation (i.e. by software, hardware (electronic and/or mechanical) or humans or a combination). This document may be cited in contracts for the development of systems (particularly distributed systems) or used by application developers or Petri net tool vendors or users.

The ISO/IEC 15909 series is concerned with defining a modelling language and its transfer format, known as high-level Petri nets. ISO/IEC 15909-1 provides the mathematical definition of high-level Petri nets, called the semantic model, the graphical form of the technique, known as high-level Petri net graphs (HLPNGs), and its mapping to the semantic model. It also introduces some common notational conventions for HLPNGs.

ISO/IEC 15909-2 defines a transfer format for high-level Petri nets in order to support the exchange of high-level Petri nets among different tools. This format is called the Petri net markup language (PNML). Since there are many different types of Petri nets in addition to high-level Petri nets, ISO/IEC 15909-2 defines the core concepts of Petri nets along with an XML syntax, which can be used for exchanging any kind of Petri nets. Based on this PNML core model, ISO/IEC 15909-2 also defines the transfer syntax for the types of Petri nets that are defined in ISO/IEC 15909-1: place/transition nets, symmetric nets[1)], high-level Petri nets, Petri nets with priorities, and Petri nets with time. Place/transition nets and symmetric nets can be considered to be restricted versions of high-level Petri nets. Petri nets with priorities and Petri nets with time are considered as extensions of the other types.

This document defines extensions to the types of Petri nets that are defined in ISO/IEC 15909-1. These extensions comprise enrichments of Petri net types and definitions of new Petri net types. This document also defines structuring mechanisms for these Petri net types.

In this document, the semantics which is considered is always the interleaving semantics.

This document provides an abstract mathematical syntax and a formal semantics for the technique. Conformance to the document is possible at several levels. The level of conformance depends on the class of high-level net chosen. The usual graphical notations are depicted in Annex A.

1) Symmetric nets have been first introduced as well-formed nets and are currently standardized as ISO/IEC 15909-1.

Systems and software engineering — High-level Petri nets —

Part 3: Extensions and structuring mechanisms

1 Scope

This document defines enrichments, extensions and structuring mechanisms of Petri nets, applied on the definitions proposed in ISO/IEC 15909-1. This document facilitates the definitions of new kinds of Petri nets and their interoperability, while remaining compatible with those defined in ISO/IEC 15909-1.

This document is written as a reference for designers of new Petri net variants, by defining common enrichments, extensions and structuring mechanisms, as well as a generalized process for defining new ones.

This document is applicable to a wide variety of concurrent discrete event systems and in particular distributed systems. Generic fields of application include:

— requirements analysis;

— development of specifications, designs and test suites;

— descriptions of existing systems prior to re-engineering;

— modelling business and software processes;

— providing the semantics for concurrent languages;

— simulation of systems to increase confidence;

— formal analysis of the behaviour of systems;

— and development of Petri net support tools.

This document can be applied to the design of a broad range of systems and processes, including aerospace, air traffic control, avionics, banking, biological and chemical processes, business processes, communication protocols, computer hardware architectures, control systems, databases, defence command and control systems, distributed computing, electronic commerce, fault-tolerant systems, games, hospital procedures, information systems, Internet protocols and applications, legal processes, logistics, manufacturing systems, metabolic processes, music, nuclear power systems, operating systems, transport systems (including railway control), security systems, telecommunications and workflow.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 15909-1, *Systems and software engineering — High-level Petri nets — Part 1: Concepts, definitions and graphical notation*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 15909-1 and the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

3.1
capacity
maximum multiset of tokens a *capacity place* (3.2) can hold

3.2
capacity place
special kind of place that can hold no more than a specified *capacity* (3.1)

3.3
FIFO
special kind of queue that is operated first in first out

3.4
inhibitor arc
special kind of arc that reverses the logic of an input place

Note 1 to entry: Instead of testing the presence of some tokens in the related place, it tests the lack of these.

3.5
read arc
special kind of arc that tests the presence of some tokens in the related place, without consumption

3.6
reset arc
special kind of arc that empties an input place

3.7
sort generator
generator of new sorts and operators built from a given signature (passed as a parameter)

4 Conformance

4.1 General

There are different levels of conformance to this document.

4.2 Enrichment process

A Petri net model is conformant to the enrichment process level when it contains a subset of the features defined as enrichments by Clause 5.

4.3 Extension process

A Petri net model is conformant to the extension process level when it contains a single feature defined as extension by Clause 6.

4.4 Structuring mechanism

A Petri net model is conformant to the structuring mechanism level when it embeds a structuring feature as defined by Clause 7.

5 Enrichment process

5.1 General

This clause defines enrichments of Petri nets. For each enrichment, it first provides a syntactic definition, then its individual semantics, and finally a formulation of the semantics allowing for composition with other enrichments or extensions.

5.2 Instances of enrichment

5.2.1 General

This subclause defines commonly used enrichments. The list provided here is not exhaustive. The general mechanism to define a new enrichment is provided in 5.3.

NOTE Enrichments are orthogonal one to another. Hence, they can be used in combination among themselves, and to enrich the different types of nets defined in this document.

5.2.2 Inhibitor arcs

5.2.2.1 Definition of Petri nets with inhibitor arcs

A Petri net with inhibitor arcs is a Petri net N together with a matrix $I \in AN^{|P| \times |T|}$ of inhibitor arcs.

5.2.2.2 Definition of enabling rule for Petri nets with inhibitor arcs

A transition $t \in T$ is enabled in marking M, denoted by $M[t\rangle$, iff:

$$\forall p \in \bullet t, (M(p) \geq \mathrm{Pre}(p,t)) \wedge (I(p,t) = 0 \vee M(p) < I(p,t))$$

5.2.2.3 Filtering function for inhibitor arcs

The filtering function $F_\mathrm{i}(N \in \mathcal{N}, t \in T_N)$ returns true when there are less tokens than the value specified on the inhibitor arc ($I(p,t) \neq 0$).

$$F_\mathrm{i}(N \in \mathcal{N}, t \in T_N) : \begin{cases} \text{True, iff } \forall p \in \bullet t : M(p) < I(p,t) \\ \text{False, otherwise} \end{cases}$$

The enabling function for place/transition nets with inhibitor arcs is thus:

$$E_{\mathrm{pti}}(N,t) = E_{\mathrm{pt}}(N,t) \wedge F_\mathrm{i}(N,t)$$

The enabling function for symmetric nets with inhibitor arcs is thus:

$$E_{\mathrm{sni}}(N,t) = E_{\mathrm{sn}}(N,t) \wedge F_\mathrm{i}(N,t)$$

The enabling function for high-level Petri nets with inhibitor arcs is thus:

$$E_{\mathrm{hlpni}}(N,t) = E_{\mathrm{hlpn}}(N,t) \wedge F_\mathrm{i}(N,t)$$

NOTE Multiple combinations of filtering functions can be defined (see 5.3 and 6.3), e.g. for Petri nets with priorities and inhibitor arcs.

5.2.2.4 Firing rule for Petri nets with inhibitor arcs

The firing rule for Petri nets with inhibitor arcs is the same as for the corresponding Petri net type. Hence, $\mathrm{A}_{\mathrm{pre}}$ and $\mathrm{A}_{\mathrm{post}}$ are those of the corresponding Petri net type.

5.2.3 Reset arcs

5.2.3.1 Definition of Petri nets with reset arcs

A Petri net with reset arcs is a Petri net N together with a matrix $Rt \in \{0,1\}^{|P| \times |T|}$ of reset arcs.

5.2.3.2 Definition of enabling rule for Petri nets with reset arcs

A transition $t \in T$ is enabled in marking M, denoted by $M[t\rangle$, iff:

$$\forall p \in \bullet t, M(p) \geq \mathrm{Pre}(p,t)$$

5.2.3.3 Filtering function for reset arcs

The filtering function $F_{\mathrm{reset}}(N \in \mathcal{N}, t \in T_N)$ returns true for all places.

5.2.3.4 Augmented firing rule for reset arcs

The firing rule for Petri nets with reset arcs relies on the one of the corresponding Petri net type. $\mathrm{A}_{\mathrm{pre}}$ is the one of the corresponding Petri net type. $\mathrm{A}_{\mathrm{post}}$ is the one of the corresponding Petri net type augmented by removing all tokens referenced in AN from the connected place.

5.2.4 Read arcs

5.2.4.1 Definition of Petri nets with read arcs

A Petri net with read arcs is a Petri net N together with a matrix $Rd \in AN^{|P| \times |T|}$ of read arcs.

5.2.4.2 Definition of enabling rule for Petri nets with read arcs

A transition $t \in T$ is enabled in marking M, denoted by $M[t\rangle$, iff:

$$\forall p \in \bullet t, \ (M(p) \geq \mathrm{Pre}(p,t) \ \wedge M(p) \geq Rd(p,t)$$

5.2.4.3 Filtering function for read arcs

The filtering function $F_{\mathrm{read}}(N \in \mathcal{N}, t \in T_N)$ is the same as the enabling function, applied to Rd.

5.2.4.4 Augmented firing rule for read arcs

The firing rule for Petri nets with read arcs is the same as for the corresponding Petri net type. Hence, $\mathrm{A}_{\mathrm{pre}}$ and $\mathrm{A}_{\mathrm{post}}$ are those of the corresponding Petri net type.

NOTE Tokens in read arc annotation $Rd(p,t)$ are not consumed.

5.2.5 Capacity places

5.2.5.1 Definition of Petri nets with capacity places

A Petri net with capacity place is a Petri net N together with a vector $C \in (\mathbb{N} \cup \{\infty\})^{|P|}$ of place capacities.

5.2.5.2 Definition of enabling rule for Petri nets with capacity places

A transition $t \in T$ is enabled in marking M, denoted by $M[t\rangle$, iff:

$$[\forall p \in \bullet t : M(p) \geq \mathrm{Pre}(p,t)] \wedge [\forall p' \in t\bullet : M(p') + \mathrm{Post}(p',t) - \mathrm{Pre}(p',t) \leq C(p')]$$

NOTE For all Petri net types, only the number of tokens in a place is limited by its capacity.

5.2.5.3 Filtering function for capacity places

The filtering function $F_c(N \in \mathcal{N}, t \in T_N)$ returns true when the firing would not lead to exceeding the capacity for output places.

$$F_c(N \in \mathcal{N}, t \in T_N) : \begin{cases} \text{True, iff } [\forall p \in \bullet t : M(p) \geq \mathrm{Pre}(p,t)] \wedge [\forall p' \in t\bullet : M(p') + \mathrm{Post}(p',t) - \mathrm{Pre}(p',t) \leq C(p')] \\ \text{False, otherwise} \end{cases}$$

The enabling function for place/transition nets with capacity places is thus:

$$E_{\mathrm{ptc}}(N,t) = E_{\mathrm{pt}}(N,t) \wedge F_c(N,t)$$

The enabling function for symmetric nets with capacity places is thus:

$$E_{\mathrm{snc}}(N,t) = E_{\mathrm{sn}}(N,t) \wedge F_c(N,t)$$

The enabling function for high-level Petri nets with capacity places is thus:

$$E_{\mathrm{hlpnc}}(N,t) = E_{\mathrm{hlpn}}(N,t) \wedge F_c(N,t)$$

NOTE Multiple combinations of filtering functions can be defined (see 5.3 and 6.3), e.g. for Petri nets with priorities and capacity places.

5.2.5.4 Firing rule for Petri nets with capacity place

The firing rule for Petri nets with capacity places is the same as for the corresponding Petri net type. Hence, A_{pre} and A_{post} are those of the corresponding Petri net type.

5.3 Generalized enrichment process

5.3.1 General

This subclause defines the template used to define a new enrichment to an existing Petri net type.

5.3.2 Definition of Petri nets with enrichment

A Petri net with ε is a Petri net N together with a definition D of the enrichment according to the net elements.

