

Petri Nets – Principles and Practice

- Basic Concepts of Petri Nets
- Definition
- Practice
- Appendix

Basic Concepts of Petri Nets

- initiated by Carl Adam Petri (*1926) in 1962
- formal and graphical language used for modelling, analysis and verification
- suitable for discrete *parallel* systems with asynchronous events
- used for dynamic systems with a static structure
 - computers
 - operating systems
 - office processes
 - production lines

Definition

■ triple $N = \{S, T, F\}$:

(1) $S \cup T \neq \emptyset$ and $S \cap T = \emptyset$

(2) $F \subseteq (S \times T) \cup (T \times S)$

(3) $\text{dom}(F) \cup \text{ran}(F) = S \cup T$:

$$\text{dom}(F) = \{x \in S \cup T : (x, y) \in F \text{ for some } y \in S \cup T\}$$

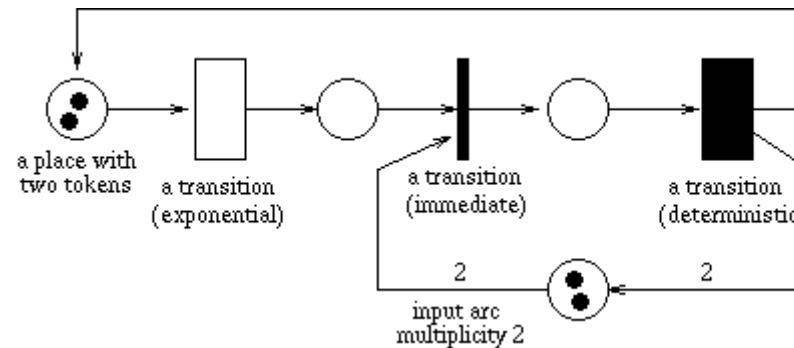
$$\text{ran}(F) = \{y \in S \cup T : (x, y) \in F \text{ for some } x \in S \cup T\}$$

S – States

T – Transitions

F – flow relation

Practice



- places can contain tokens → state is determined by the number and type (if token can be distinguished) in each place
- transitions are active components, they *fire* → change of state
- transitions are only allowed to fire if they are enabled (each precondition is fulfilled)

- firing means removing tokens from their input places and adding them to the output places, depending on the cardinality of each flow relation (often called “arc”)

- two main models:
 - capacity of places limited to 1 (boolean) → “condition-event system”
 - multiple token allowed in one place → “place-transition system”

- tokens may represent different objects: numbers, materials, persons etc.

- properties which are of interest:
 - does the petri net *terminate* ?

 - will the petri net be *alive* ?

Appendix

- used resources:

<http://pdv.cs.tu-berlin.de/~azi/petri.html>

<http://www.daimi.au.dk/PetriNets/>

Duden Informatik