Embedded Systems 4

BF - ES

Petri nets

Introduced in 1962 by Carl Adam Petri in his PhD thesis.

Different “Types” of Petri nets known

- Condition/event nets
- Place/transition nets
- Predicate/transition nets
- Hierachical Petri nets, …

BF - ES

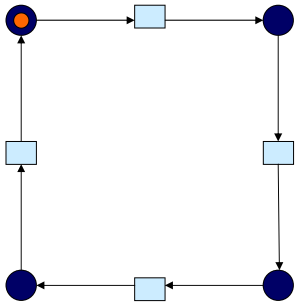
Used for Modelling, Analysis, Verification of Distributed Systems

(other) application areas:

- automation engineering
- business processes

Focus on modeling causal dependencies;
no global synchronization assumed (message passing only).

Example 1: The four seasons



Key Elements

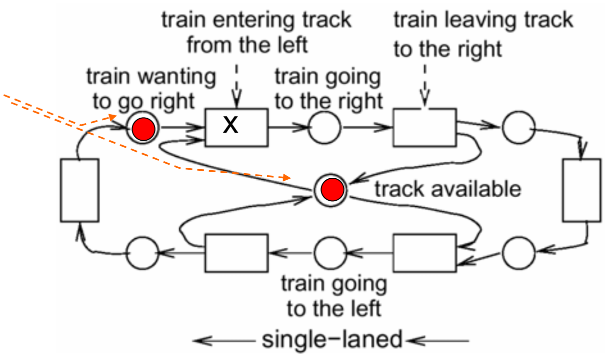
- **Conditions**
  Either met or not met. Conditions represent “local states”. Set of conditions describes the potential state space.
- **Events**
  May take place if certain conditions are met. Event represents a state transition.
- **Flow relation**
  Relates conditions and events, describes how an event changes the local and global state.
- **Tokens**
  Assignments of tokens to conditions specifies a global state.

Conditions, events and the flow relation form
a **bipartite graph** (graph with two kinds of nodes).

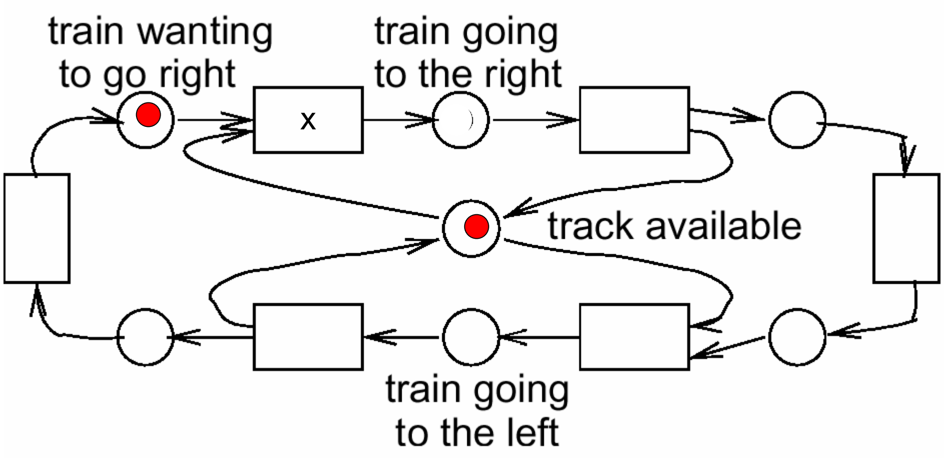
Example 2: Synchronization at single track rail segment

- **mutual exclusion:**
  there is at most one train using the track rail

„Preconditions“ of x fulfilled

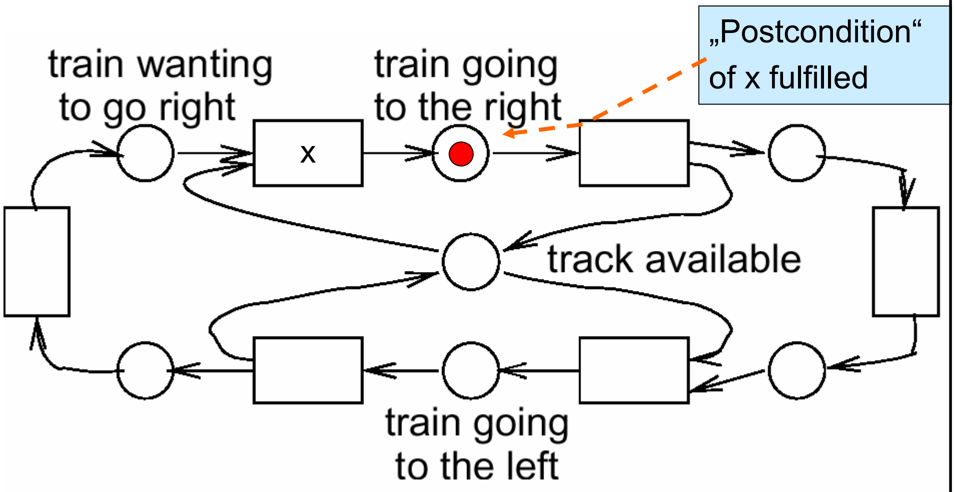


Playing the „token game“: dynamic behavior



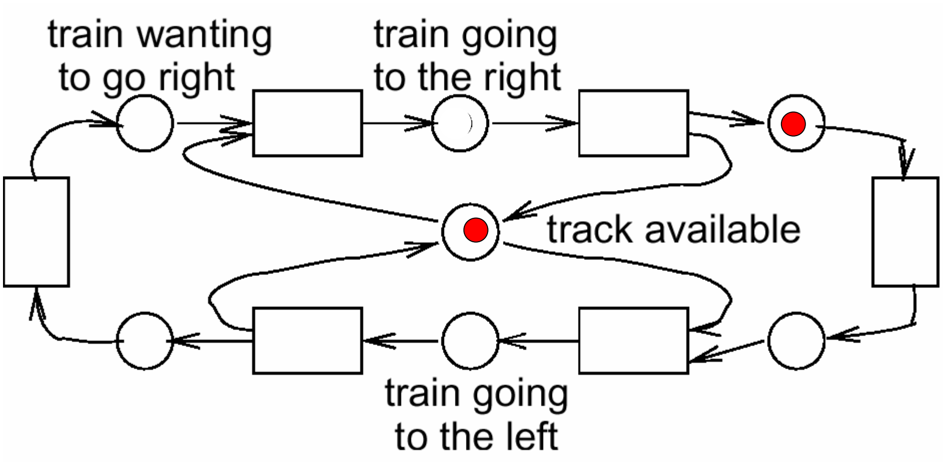
BF - ES

Playing the „token game“: dynamic behavior



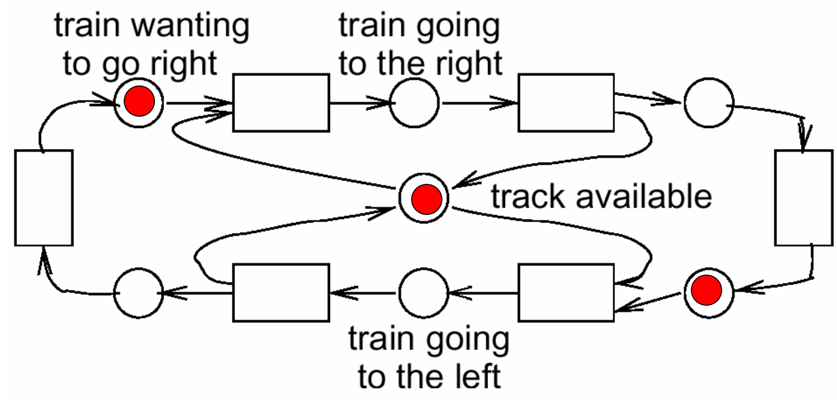
BF - ES

Playing the „token game“: dynamic behavior



BF - ES

Conflict for resource „track“: two trains competing



BF - ES

Petri Nets

Def.: *N*=(*C*,*E*,*F*) is called a **Petri net**, iff the following holds

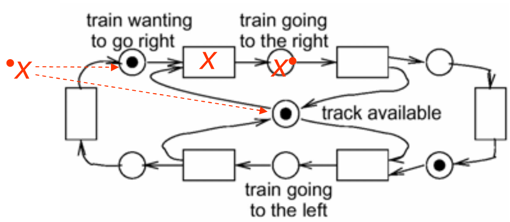
1. *C* and *E* are disjoint sets
2. $F \subseteq (C \times E) \cup (E \times C)$; is binary relation, („**flow relation**")

Def.: Let *N* be a net and let $x \in (C \cup E)$.

$•x := \{y \mid y\ F\ x\}$ is called the set of **preconditions.**

$x• := \{y \mid x\ F\ y\}$ is called the set of **postconditions.**

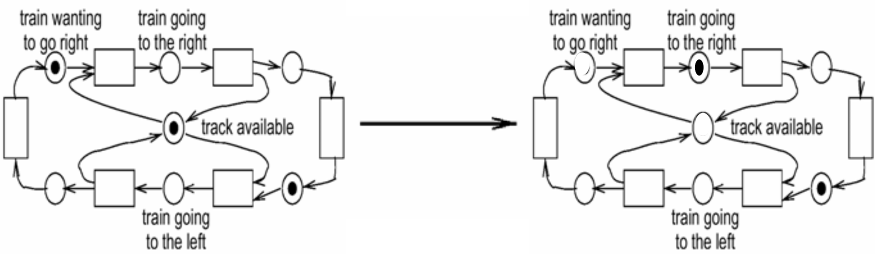
Example:



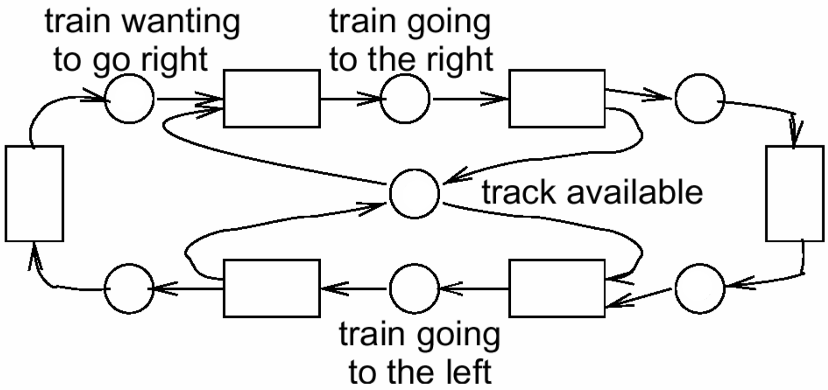
Boolean marking and computing changes of markings

- A Boolean marking is a mapping M: C → { 0,1 }.
- „Firing" events *x* generate new markings on each of the conditions *c* according to the following rules:

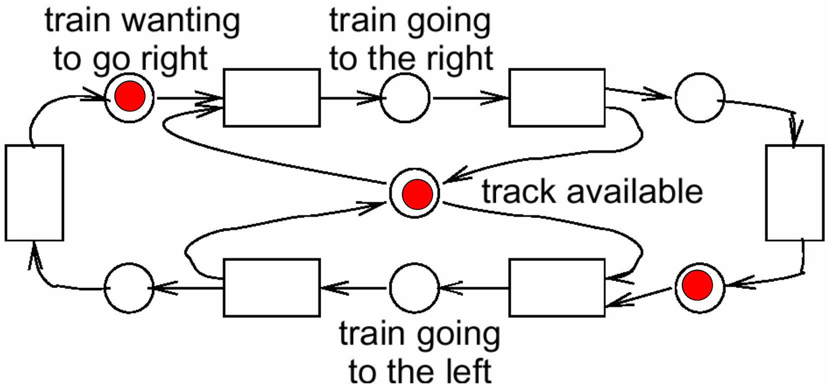
  a transition at *x* can be fired, iff •*x, i.e.* all preconditions of *x* are marked and *x*• is not marked, after firing •*x is* unmarked and *x*• is marked
- M → M', iff M' results from M by firing exactly one transition



Competing Trains Example:



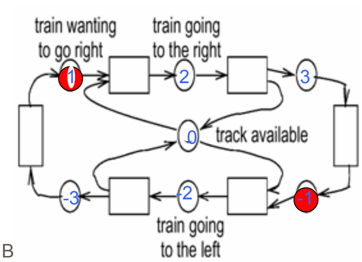
Competing Trains Example: Conflict for resource „track“

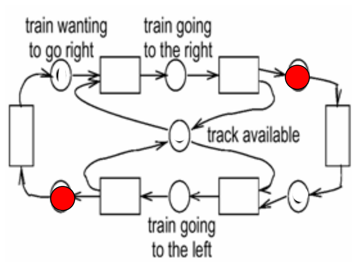


Competing Trains Example: Boolean marking and computing changes of markings

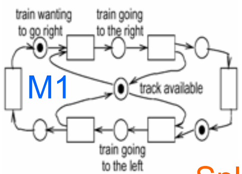
Competing Trains example

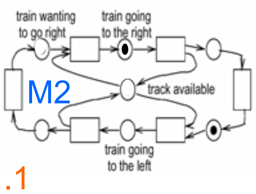
- Consider the mapping sym: C → C
  with sym(c)=-c for all c=0,1,2,3,-1,-2,-3
- We call two markings M1, M1s symmetric,
  iff M1 can be transformed to M1s by changing the marks from a node c to node -c , i.e M1s(sym(c)):= M1(c).
- It follows easily: M1→M2 iff M1s → M2s

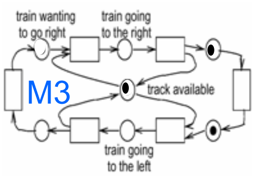




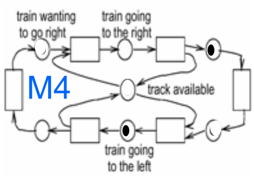
Competing Trains Example: Boolean marking and computing changes of markings
Reachable markings

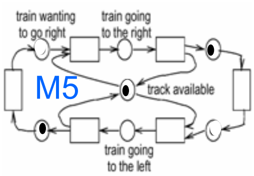


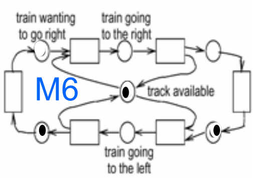




Split 1.1

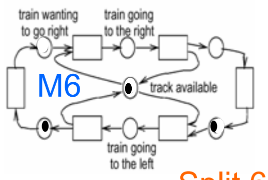


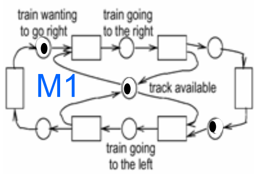




Split 4.1 Split 5.1

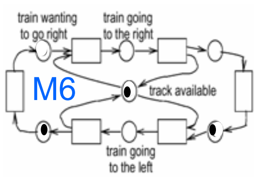
Competing Trains Example: Boolean marking and computing changes of markings
Reachable markings

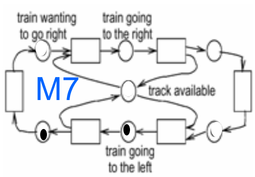


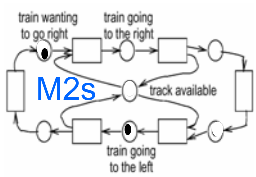


Split 6.1

Competing Trains Example: Boolean marking and computing changes of markings
Reachable markings

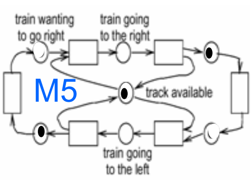


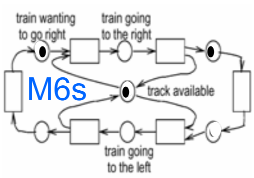




Split 6.2

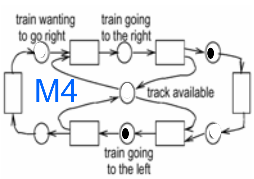
Competing Trains Example: Boolean marking and computing changes of markings Reachable markings

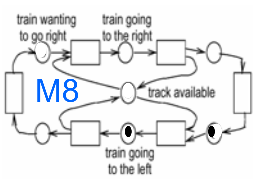


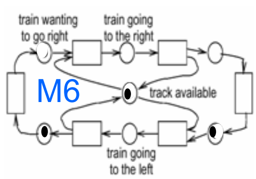


Split 5.2

Competing Trains Example: Boolean marking and computing changes of markings Reachable markings

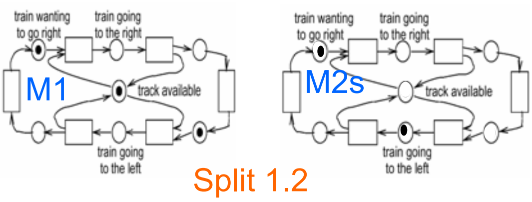






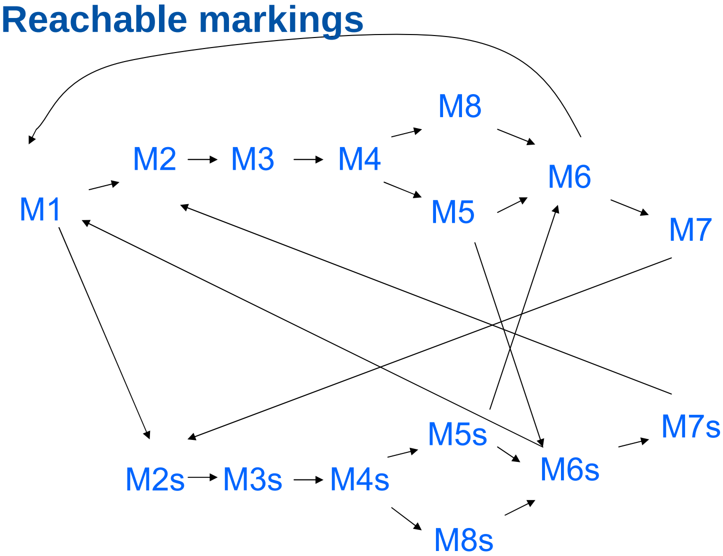
Split 4.2

Competing Trains Example: Boolean marking and computing changes of markings Reachable markings



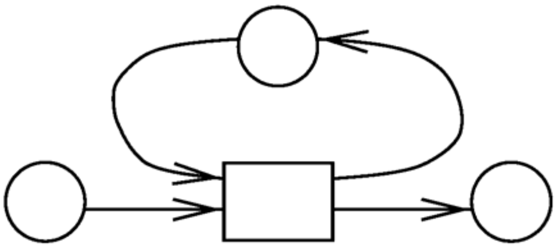
Split 1.2

Competing Trains Example: Boolean marking and computing changes of markings Reachable markings



Basic structural properties: Loops and pure nets

Def.: Let $(c,e) \in C \times E$. (c,e) is called a **loop** iff $cFe \wedge eFc$.



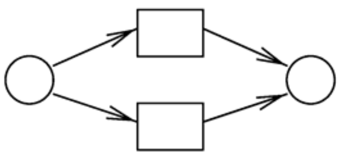
Def.: Net $N=(C,E,F)$ is called **pure**, if F does not contain any loops.

Structural properties: Simple nets

Def.: A net is called **simple**, iff

$$[x,y \in (C \cup E) \wedge ({}^\bullet x = {}^\bullet y) \wedge (x^\bullet = y^\bullet)] \rightarrow x = y$$

Example (not a simple net):

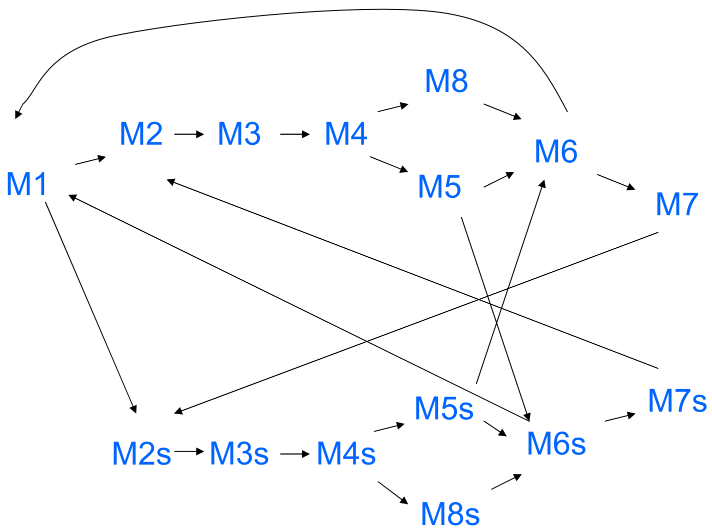


Condition/event nets (C/E nets)

Def.:
A Petri net *N*=(*C*,*E*,*F*) together with a set of Markings *M* is called **condition/event net (C/E net)**, iff

- *N* is simple and has no isolated elements
- *M* is closed w.r.t. “firing” and “inverse firing”
- two markings in *M* can be transformed into each other by “firing” and “inverse firing”
- for each event $e \in E$, there exists a marking in *M*, that allows firing at *e*

Competing Trains Example is C/E net:



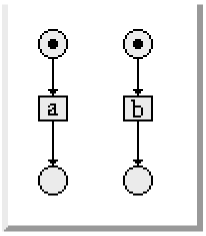
Properties of C/E

Def.:

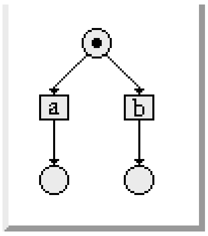
- Marking M' is **reachable** from marking M, iff there exists sequence of firing steps transforming M into M' (Not.: M➲M')
- A C/E net is **cyclic**, iff any two markings are reachable from each other.
- A C/E net fulfills **liveness,** iff for each marking M and for each event e there exists a reachable marking M' that activates e for firing

Expressiveness: basic examples

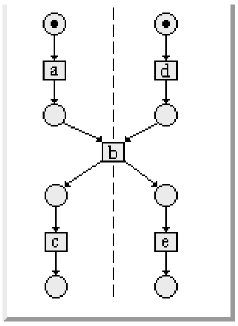
- concurrency of transitions



- alternative or conflict

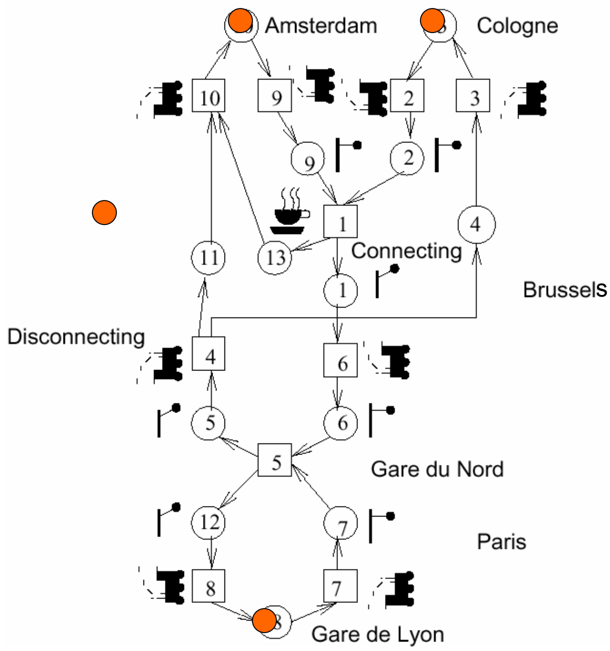


- synchronization



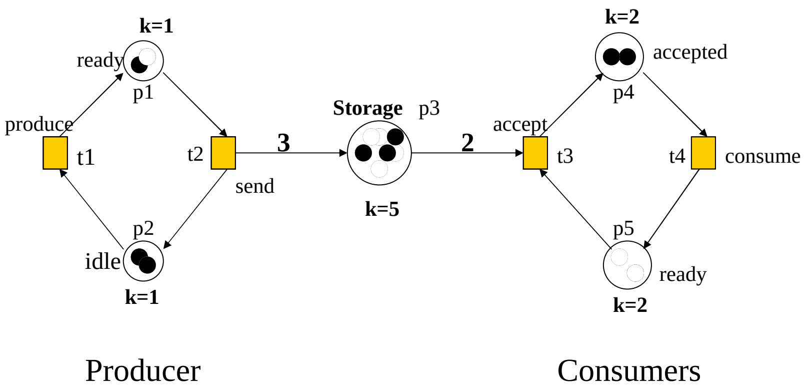
Example Thalys trains: more complex

- Thalys trains between Cologne, Amsterdam, Brussels and Paris.
- Synchronization at Brussels and Paris



Realistic scenarios need more general definitions

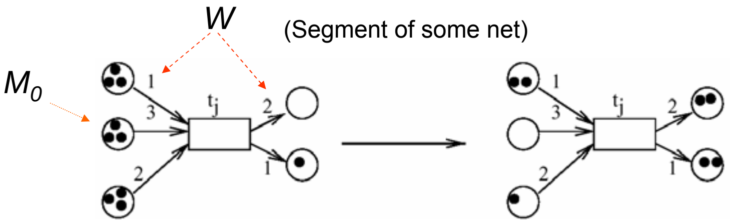
- More than one token per condition, capacities of places
- weights of edges
- state space of Petri nets may become infinite!



Place/transition nets

Def.: (P, T, F, K, W, M_0) is called a **place/transition net (P/T net)** iff

1. $N=(P,T,F)$ is a **net** with places $p \in P$ and transitions $t \in T$
2. $K: P \rightarrow (\mathbf{N_0} \cup \{\omega\}) \backslash \{0\}$ denotes the **capacity** of places (ω symbolizes infinite capacity)
3. $W: F \rightarrow (\mathbf{N_0} \backslash \{0\})$ denotes the **weight of graph edges**
4. $M_0: P \rightarrow \mathbf{N_0} \cup \{\omega\}$ represents the **initial marking** of places



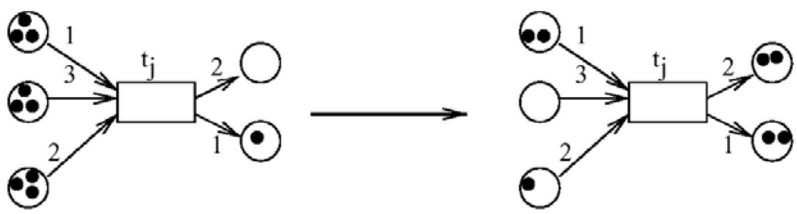
(Segment of some net)

defaults:
K = ω
W = 1

Computing changes of markings

- „Firing“ transitions *t* generate new markings on each of the places *p* according to the following rules:

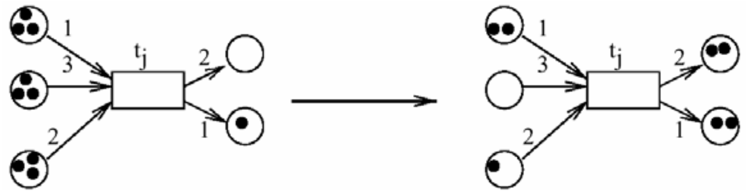
$$M'(p) = \begin{cases} M(p) - W(p,t), & \text{if } p \in {}^\bullet t \setminus t^\bullet \\ M(p) + W(t,p), & \text{if } p \in t^\bullet \setminus {}^\bullet t \\ M(p) - W(p,t) + W(t,p), & \text{if } p \in {}^\bullet t \cap t^\bullet \\ M(p) & \text{otherwise} \end{cases}$$



Activated transitions

- Transition *t* is „activated" iff

$$(\forall p \in {}^{\bullet}t : M(p) \geq W(p,t)) \wedge (\forall p \in t^{\bullet} : M(p) + W(t,p) \leq K(p))$$



Activated transitions can „take place" or „fire",
but don't have to.
The order in which activated transitions fire is not fixed
(it is non-deterministic).

Shorthand for changes of markings

Firing transition:

$$M'(p) = \begin{cases} M(p) - W(p,t), & \text{if } p \in {}^{\bullet}t \setminus t^{\bullet} \\ M(p) + W(t,p), & \text{if } p \in t^{\bullet} \setminus {}^{\bullet}t \\ M(p) - W(p,t) + W(t,p), & \text{if } p \in {}^{\bullet}t \cap t^{\bullet} \\ M(p) & \text{otherwise} \end{cases}$$

Let

$$\underline{t}(p) = \begin{cases} -W(p,t) \text{ if } p \in {}^{\bullet}t \setminus t^{\bullet} \\ +W(t,p) \text{ if } p \in t^{\bullet} \setminus {}^{\bullet}t \\ -W(p,t) + W(t,p) \text{ if } p \in t^{\bullet} \cap {}^{\bullet}t \\ 0 \end{cases}$$

$\Rightarrow$ $\forall p \in P$: $M'(p) = M(p) + \underline{t}(p)$

$\Rightarrow$ $M' = M + \underline{t}$ +: vector add

Matrix $\underline{N}$ describing all changes of markings

$$\underline{t}(p) = \begin{cases} -W(p,t) \text{ if } p \in {}^{\bullet}t \setminus t^{\bullet} \\ +W(t,p) \text{ if } p \in t^{\bullet} \setminus {}^{\bullet}t \\ -W(p,t) + W(t,p) \text{ if } p \in t^{\bullet} \cap {}^{\bullet}t \\ 0 \end{cases}$$

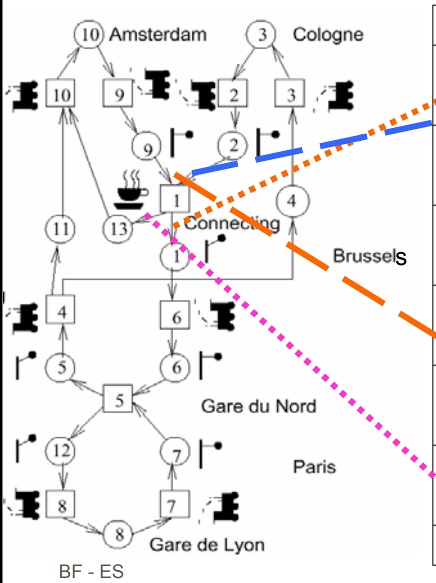
Def.: Matrix $\underline{N}$ of net N is a mapping

$\underline{N}$: $P \times T \rightarrow Z$ (integers)

such that $\forall\, t \in T$: $\underline{N}(p,t) = \underline{t}(p)$

Component in column t and row p indicates the change of the marking of place p if transition t takes place.

Example: $\underline{N}$ =



	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
p_1	1					–1				
p_2	–1	1								
p_3		–1	1							
p_4			–1	1						
p_5				–1	1					
p_6					–1	1				
p_7					–1		1			
p_8							–1			
p_9	–1							1	1	
p_{10}									–1	1
p_{11}				1						–1
p_{12}					1			–1		
p_{13}	1									–1

Computation of Invariants

We are interested in subsets *R* of places whose number of labels remain invariant under transitions,

•e.g. the number of trains commuting between Amsterdam and Paris (Cologne and Paris) remains constant

Important for correctness proofs, e.g. the proof of liveness

Competing Trains Example: Place Invariant 2,0,-2