SYSTEMS ANALYSIS
LECTURE 6
PETRI NETS

Zuzana Bělinová

Petri nets – PN

- Graphical tool
- Used for system modelling
- Mathematical construct based on graph theory

- Easily managed by computers
- Useful for description of processes

Petri nets

- Used in 2 forms – graph or matrix
- Tool for process modelling
- Can be used for conflict situation analysis
- Created by Carl Adam Petri (German)
  - Based on his Ph.D. thesis from 1962
  - At the beginning used for flow modelling

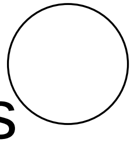
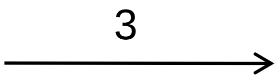
Petri nets - usage

- Useful for description of:
  - Concurrent processes
  - Parallel processes
  - Non-determenistic processes

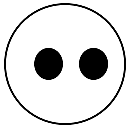
Petri nets - usage

- Used in many areas, e.g.
  - work-flows
  - flexible manufacturing
  - hardware structures
  - real-time systems
  - operations research
  - embedded systems
  - telecommunications
  - trading
  - logistics
  - transport networks
  - biological systems
  - etc.

Special type of oriented graph

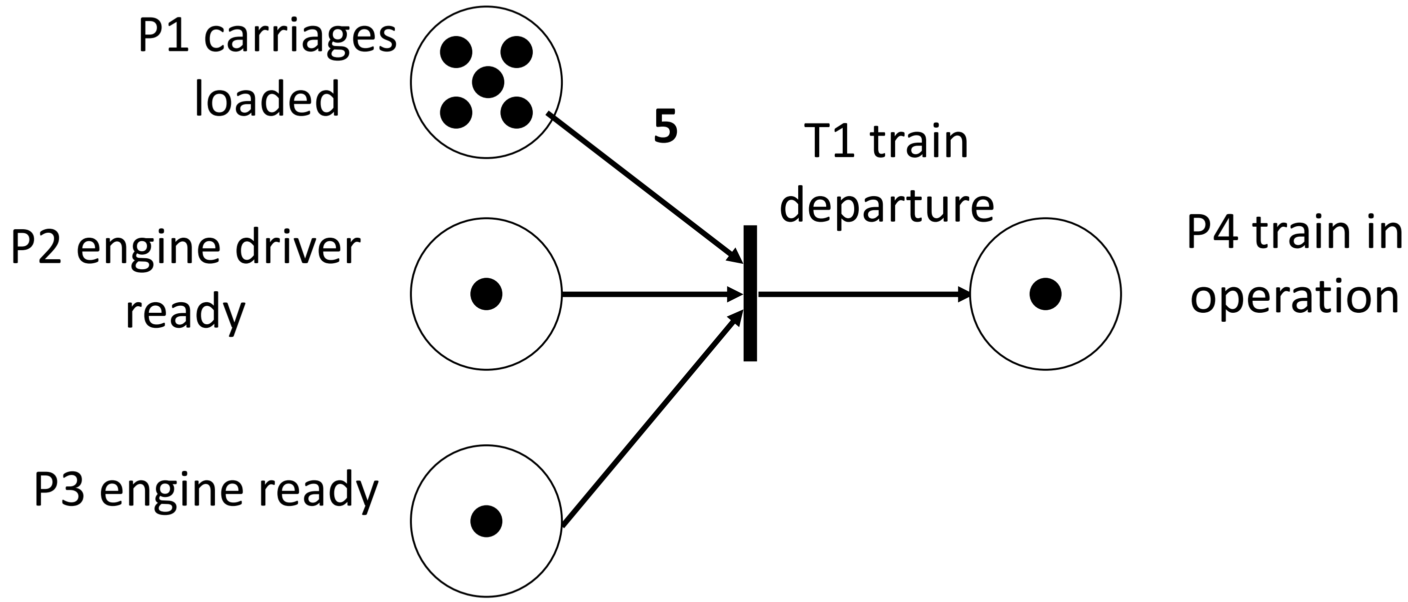
- 2 types of nodes
  
  - **P** – places – describe states
  
  - **T** – transitions - describe events
  

- Directed (oriented) arcs with weight
  - Arcs connect ALWAYS different nodes (from place to transtion of from transiton to place)
  - Arcs can fork both in places or in transitions

- Tokens in places marking active states
  

Petri nets – principle - example

- Places represent system‘s states
- Transitions represent actions



- *Note: do not confuse the Petri net graph for the graph of systems structure!*
- *Places to not represent system elements*

Petri nets - definition

PN := (P,T,F,B,M⁰)

- **P** := $\{p_1,p_2, . . . ,p_n\}$ set of all places in the PN (they represent possible states of the system)
- **T** := $\{t_1,t_2, . . . ,t_l\}$ set of all transitions in the PN (they represent possible state changes, possible events)
- **F** is forward matrix – describes relation between places and following transitions
  - If there is arc $p_i \rightarrow t_j$, than element of matrix F: $f_{i,j}$ means the weight of the arc
  - if the arc $p_i \rightarrow t_j$ does not exist, $f_{i,j} = 0$
- **B** is backward matrix – describes relations between places and antecedent transitions ($P \leftarrow T$)
  - If there is arc $t_j \rightarrow p_i$, than element of matrix B: $b_{i,j}$ means the weight of the arc
  - if the arc $p_i \leftarrow t_j$ does not exist, $b_{i,j} = 0$
- **M** = $\{m_1,m_2, . . . ,m_n\}$ is the marking vector (contains list of places that are active in the current state of PN
- **M^0** – is the vector of initial marking (contains list of places that are active at the beginning

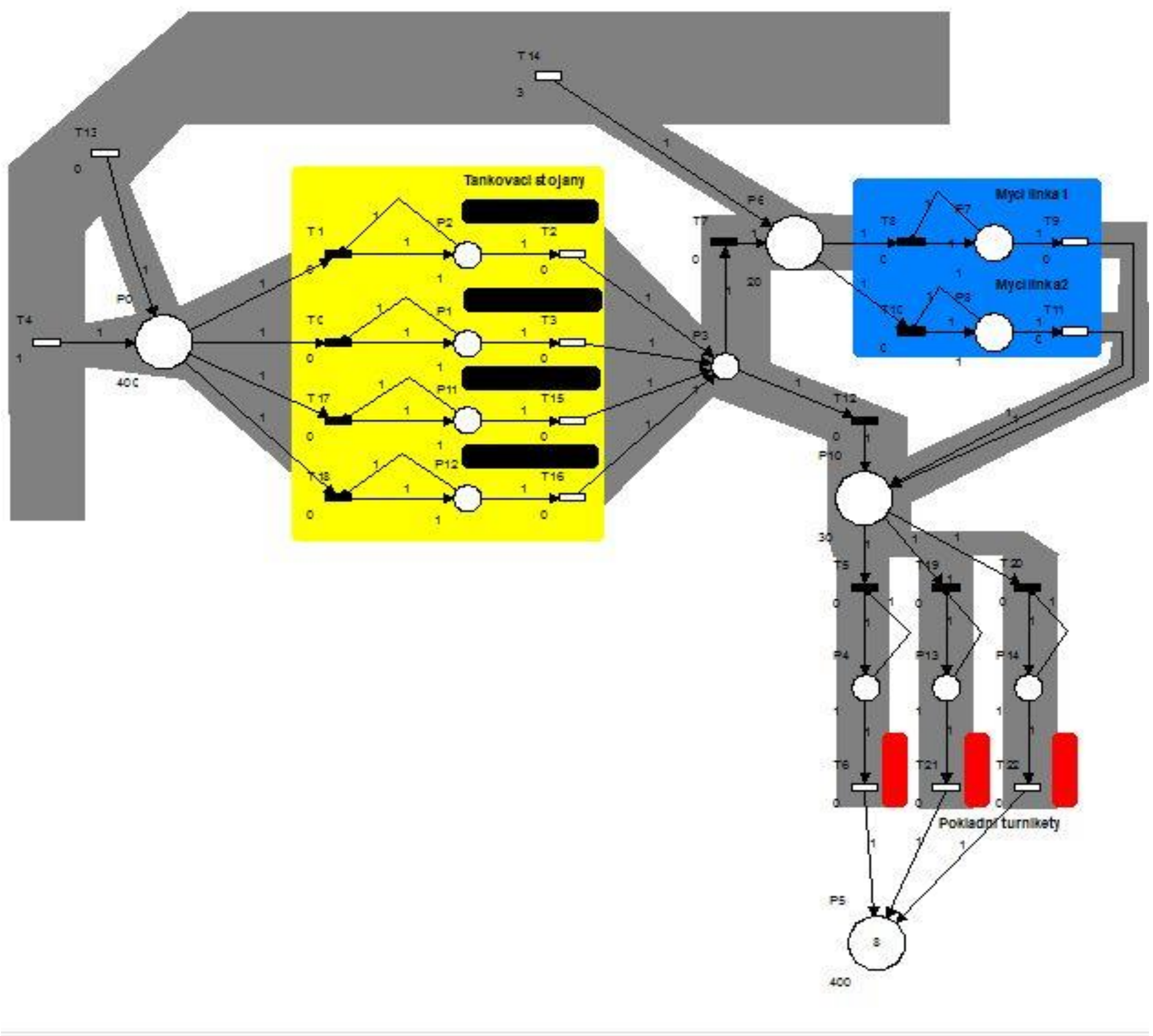
Example

- P=(P1- carriages loaded, P2 – engine driever ready, P3 – engine ready, P4 – train in operation)
- T=(T1- train departure)

F	T1		B	T1
P1	5		P1	0
P2	1		P2	0
P3	1		P3	0
P4	0		P4	1

- M0=(3,1,0,0)

Example – petrol station model

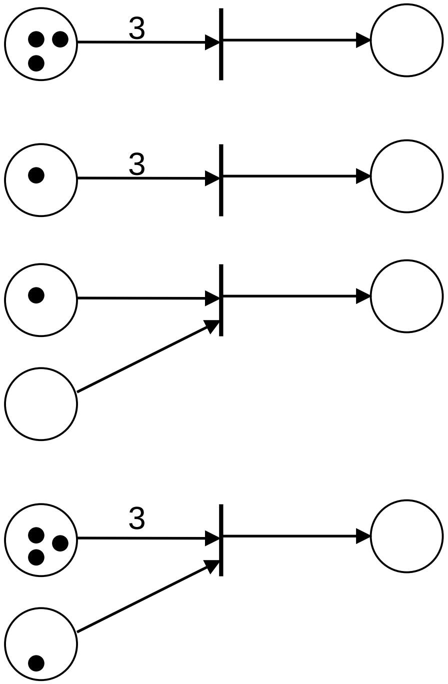


Petri nets modelling rules

- Rules for adding and removing of tokens:

- Tokens are removed from places in front of transition and added to places behind a transition
- It is possible to „fire“ a transition only when it is enabled
- A transition is said to be enabled if each input place has at least as many tokens as the weight of the arc connecting them.
- When a transition is fired, from each input place the number of tokens equal to the weight of the arc connecting them is removed and to each output place number of tokens equal to the weight of the arc joining them is added.
- A special kind of arc, the inhibitor arc, is used to reverse the logic of an input place. With an inhibitor arc, the absence of a token in the input place enables, not the presence (or less number than is the weight of the arc) 

Are these transitions enabled?



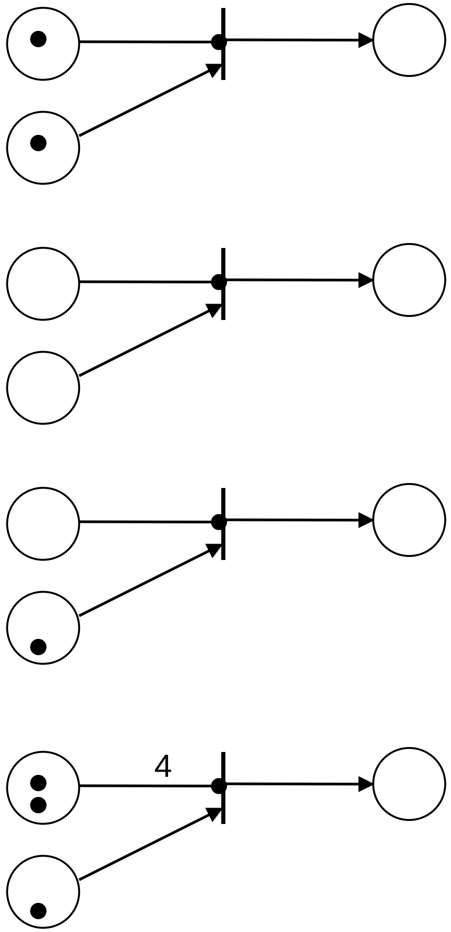
Yes

No

No

Yes

Inhibitor arc
Are these transitions enabled?



No

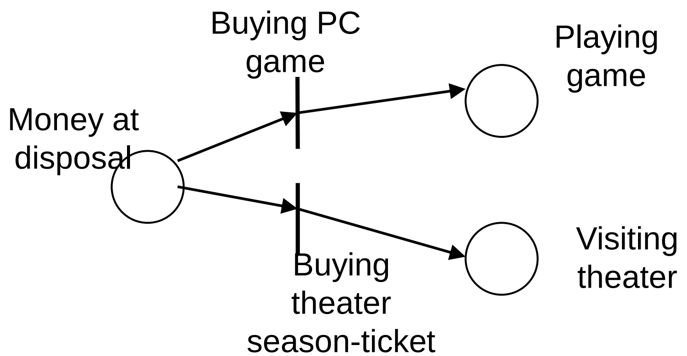
No

Yes

Yes

Special Petri nets types

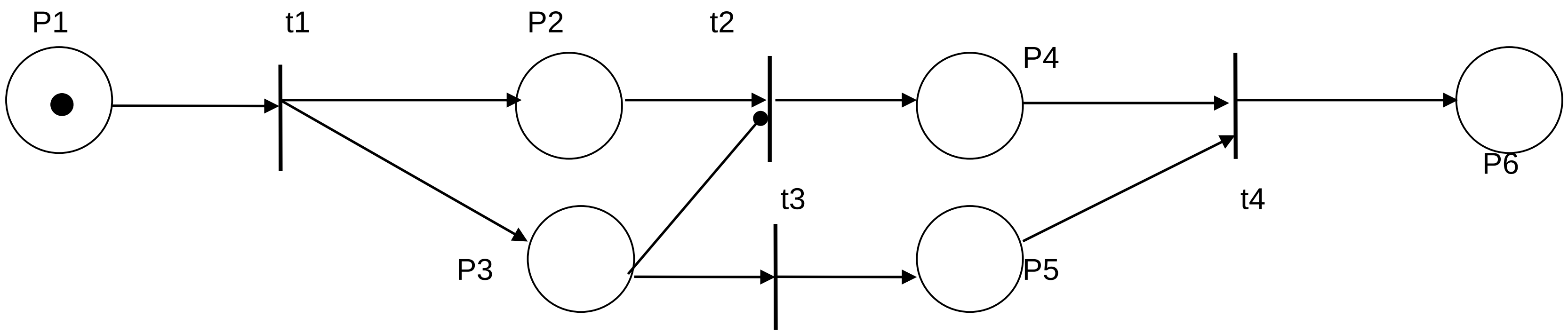
- Conflict net – more than one transition enabled at the same time

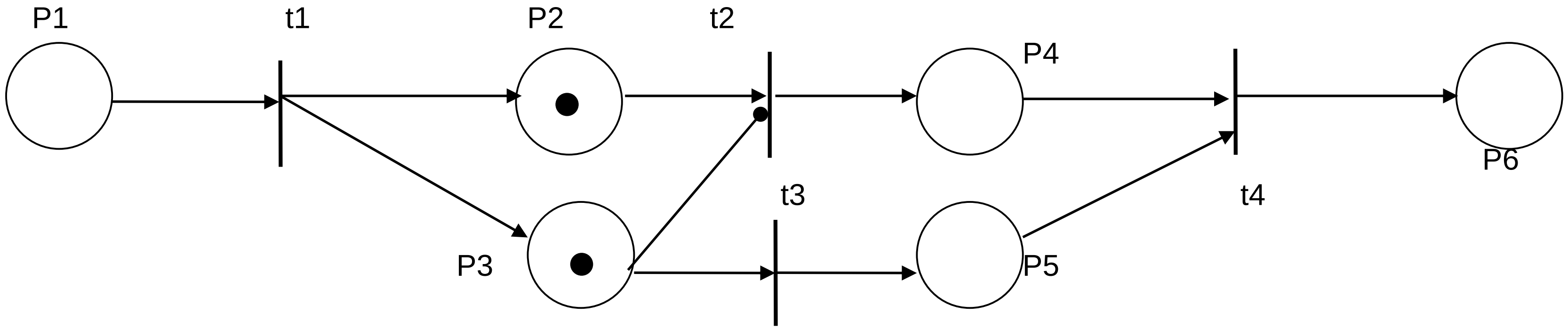


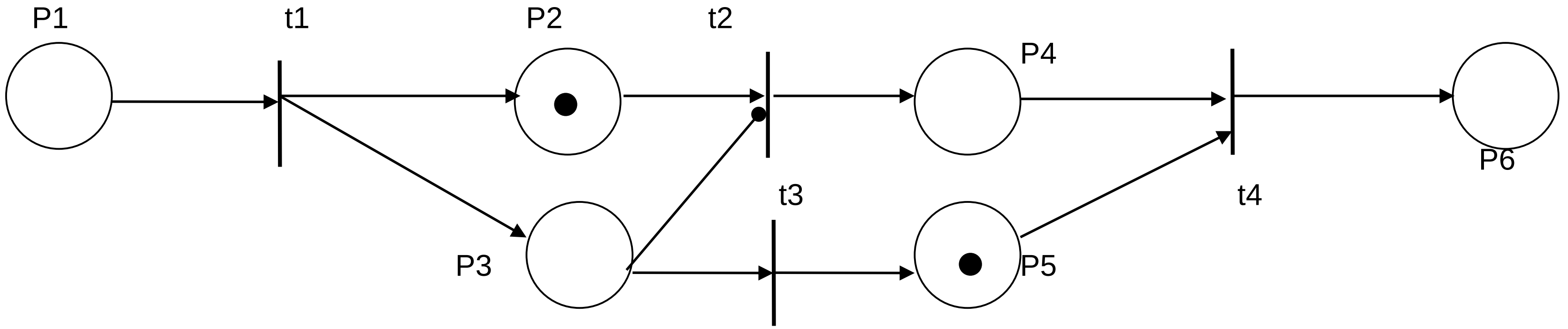
- Safe net – in every place there is at maximum 1 token (1-bounded)
- k-bounded net – in every place there are at maximum k tokens
- Live net – it is possible to fire all transitions (for particular initial state M_0)
- Conservative net – constant number of tokens

How to record the processes in PN?

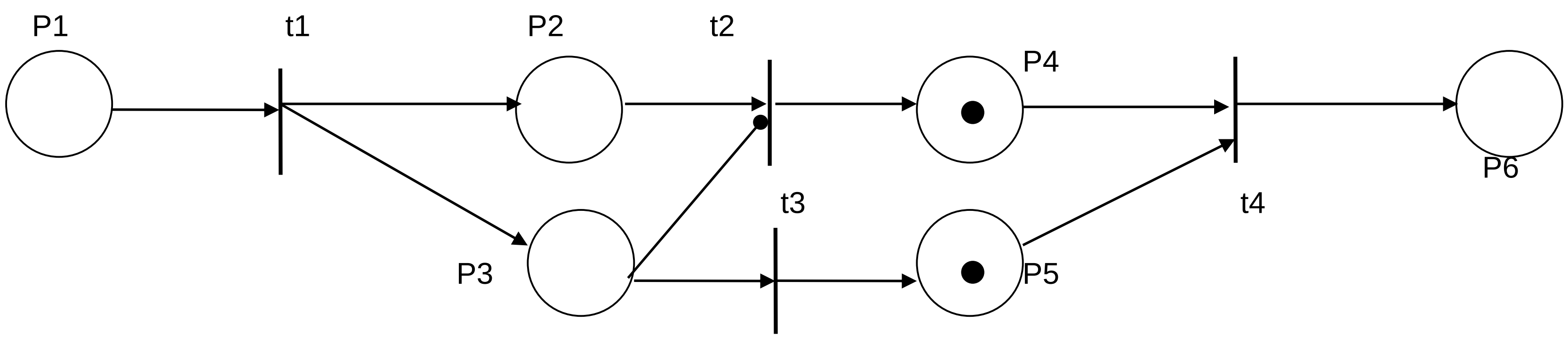
- Graphically

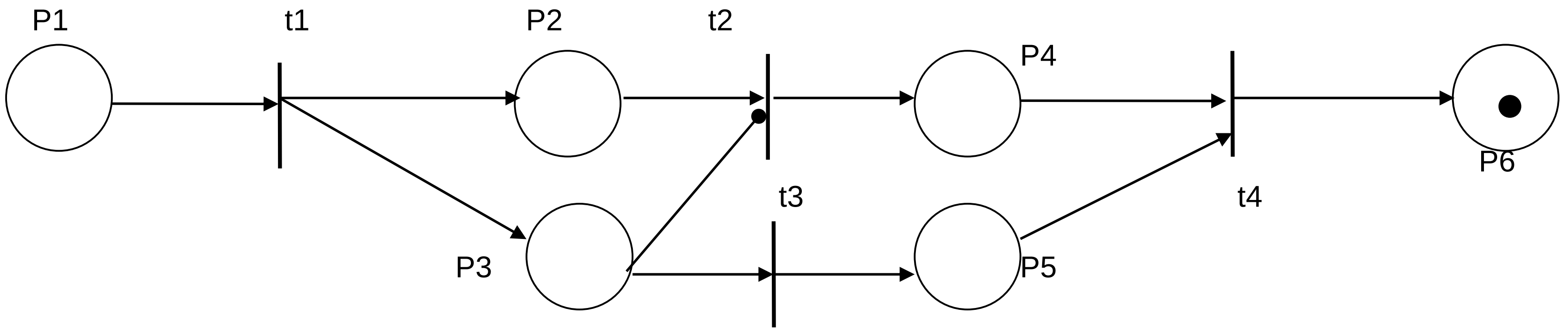



P1
t1
P2
t2
P4
P6
t3
t4
P3
P5



P1
t1
P2
t2
P4
P6
t3
t4
P3
P5

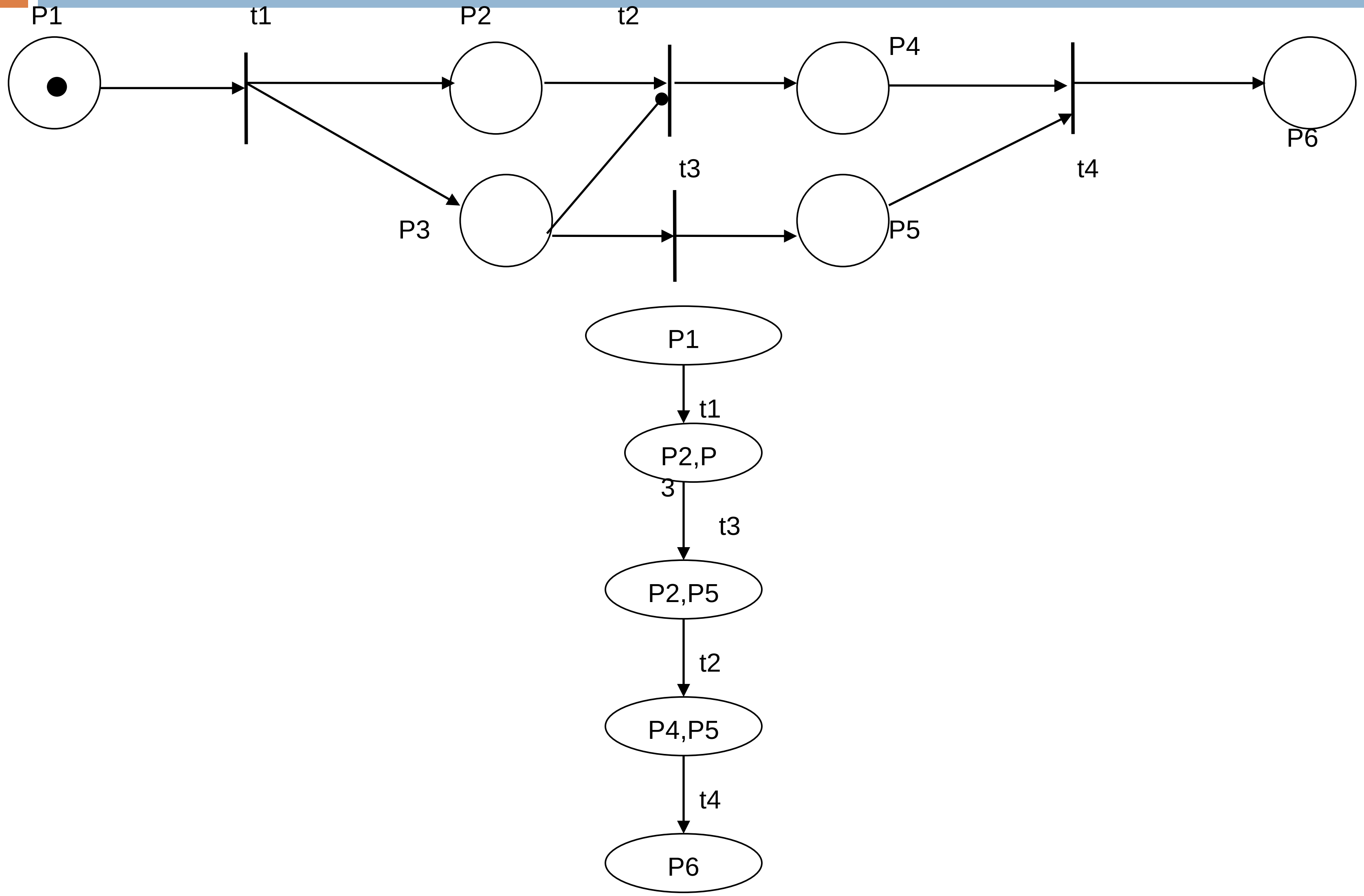

P1
t1
P2
t2
P4
P6
t3
t4
P3
P5


P1
t1
P2
t2
P4
P6
t3
t4
P3
P5

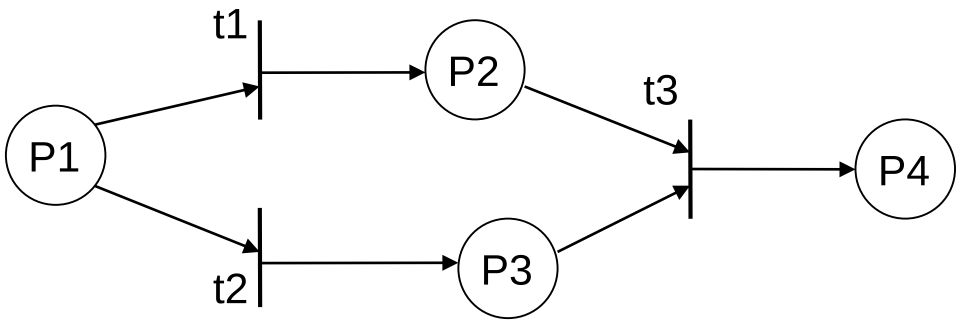
How to record the processes in PN?

- State Transition Diagram
  - Oriented graph, nodes are lists of enabled places, arcs are fired transitions
  - First node is equivalent to the vector of initial marking M^0
  - State Transition Diagram branches only for conflict situations – each branch models alternative of the process
  - State Transition Diagram contains ALL alternatives
  - For each M^0 different State Transition Diagram arises

State Transition Diagram - example



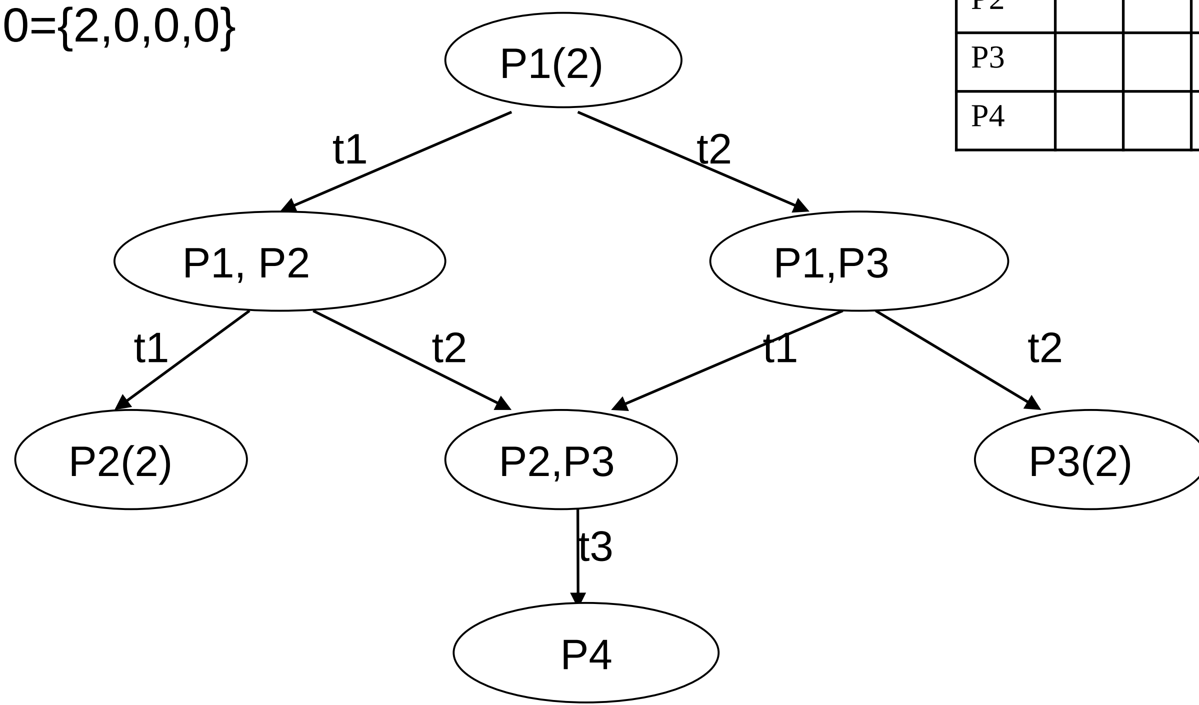
Example



Forward matrix			
	t1	t2	t3
P1	1	1	
P2			1
P3			1
P4			

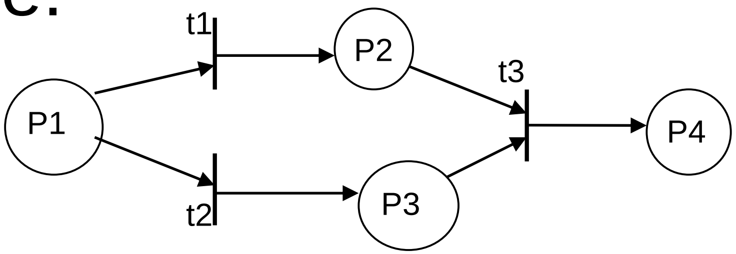
Backward matrix			
	t1	t2	t3
P1			
P2	1		
P3		1	
P4			1

M0={2,0,0,0}



State transition diagram by matrixes (easy for computer processing)

Example:



- B-F=A
- To every column in A add vector M. If the resulting column does not have negative value, it is new M vector.
- Add the new M vector again to the A matrix
- ...

F			
	T1	T2	T3
P1	1	1	0
P2	0	0	1
P3	0	0	1
P4	0	0	0

B			
	T1	T2	T3
P1	0	0	0
P2	1	0	0
P3	0	1	0
P4	0	0	1

A=B-F				
	T1	T2	T3	M0
P1	-1	-1	0	2
P2	1	0	-1	0
P3	0	1	-1	0
P4	0	0	1	0

M0+A			
	T1	T2	T3
P1	**1**	**1**	2
P2	**1**	0	-1
P3	0	**1**	-1
P4	0	0	1

M(1,1,0,0)+A			
	T1	T2	T3
P1	0	0	1
P2	2	1	0
P3	0	1	-1
P4	0	0	1

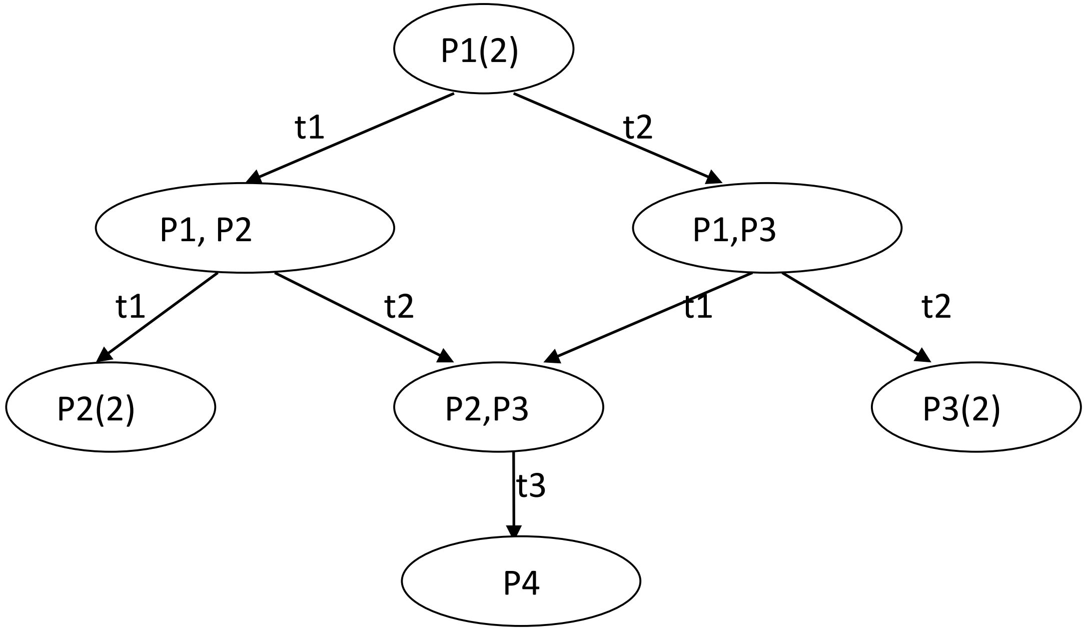
M(1,0,1,0)+A			
	T1	T2	T3
P1	0	0	1
P2	1	0	-1
P3	1	2	0
P4	0	0	1

M(0,2,0,0)+A			
	T1	T2	T3
P1	-1	-1	0
P2	3	2	1
P3	0	1	-1
P4	0	0	1

M(0,1,1,0)+A			
	T1	T2	T3
P1	-1	-1	0
P2	2	1	0
P3	1	2	0
P4	0	0	1

M(0,0,2,0)+A			
	T1	T2	T3
P1	-1	-1	0
P2	1	0	-1
P3	2	3	1
P4	0	0	1

Resulting state transition diagram



Petri nets with inhibitor arcs – evaluation by matrixes

1. We count the matrix A as usual A=B-F, but we have to mark specifically the position of the inhibitor arc (Aij)
2. We add vector M_k to every column of the matrix A
3. We assess the resulting sum of M and A with the following modification:
   a) Columns different from j (no inhibitor arc in the column) are new vector M if all the values are non-negative
   b) Columns j (containing inhibitor arc) show new marking vector when the value in position Aij (position of the inhibitor arc) is negative and the rest of the positions are non-negative. In the new marking vector on the position of the inhibitor arc we replace the counted value by the value of the previous marking vector (the one added to the matrix A), the rest of the column stays as counted.

Petri net extensions

- Additional types of arcs
- Addining places capacity
- Coloured Petri nets
- Prioritised Petri nets
- Etc.

Thank you for your attention