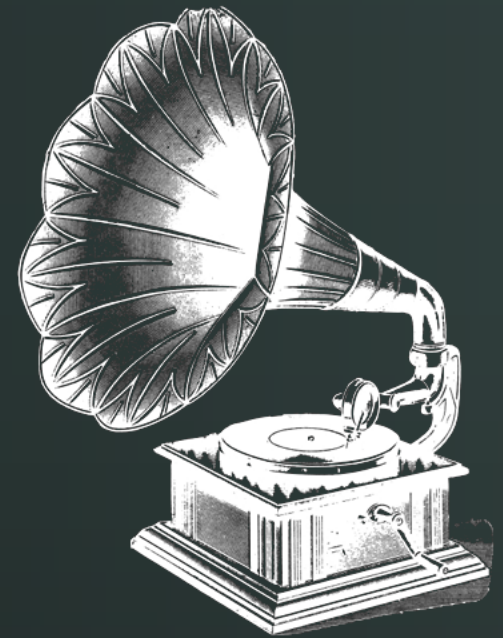


Petri Nets Quick Intro

ITBA 2010



Petri Nets

- Originally developed in the 60'ies and the 70'ies
- Language for description and analysis of synchronisation, communication and resource sharing between concurrent processes

Petri Nets are composed by

Four symbols:

- Circles: or places -> a condition in the graph
- Bars: are transitions -> are actions that can occur
- Arcs: directional connections between places and transitions
- Dots: also called tokens, are used to control the flow of the net

Petri Nets Base

- There are a lot of extensions of petri nets for different applications
- In the standard form of petri nets there is no passage of time. All actions occur instantaneously, take zero time.
- Marking Petri nets -> put tokens inside the net (in a place)
- The amount of tokens (k) in one place that the net support determine the order of the net. If $k == 1$ -> safe net.

Petri nets are directed graphs

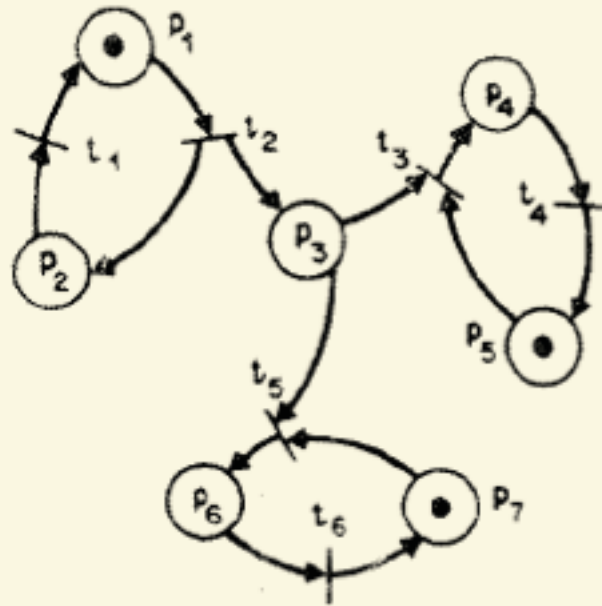
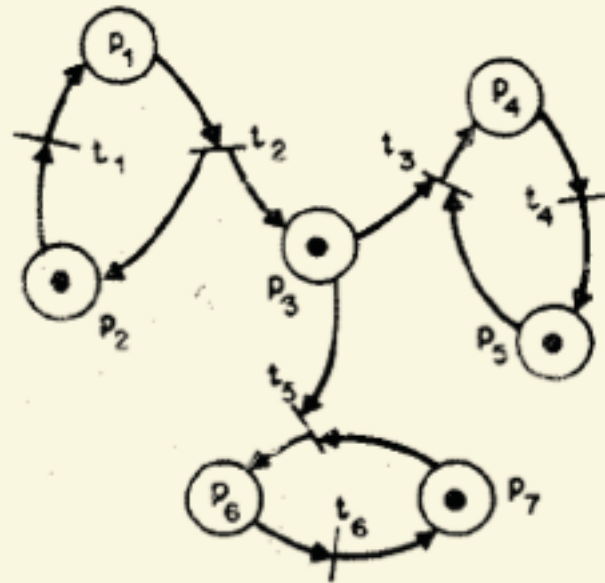


FIGURE 1 A MARKED PETRI NET

Firing a Transition

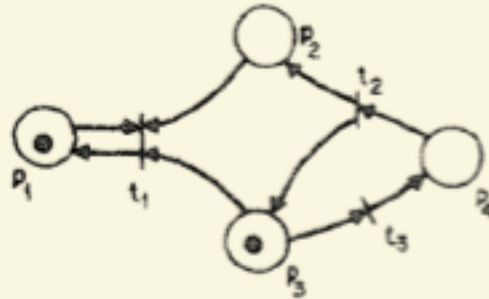


THE MARKING RESULTING FROM FIRING TRANSACTION "t2" IN FIGURE 1. NOTE THAT THE TOKEN IN "p1" WAS REMOVED AND TOKENS WERE ADDED TO "p2" AND "p3"

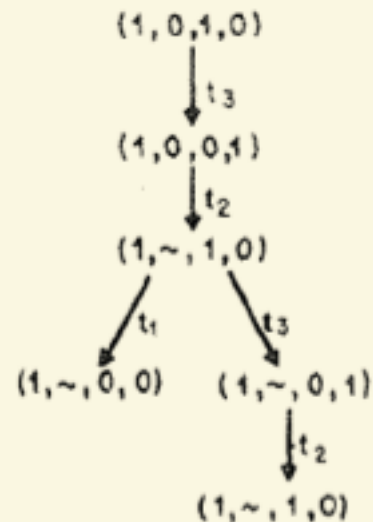
Transitions State

- "Dead", \rightarrow for an initial marking, there is now way for that transition to fire
- "Potentially friable" \rightarrow if exist at least one sequence of firings which can enable a transition
- or "Live" \rightarrow for all initial markings the transition will be enabled

Reachability Tree



A PETRI NET WITH MARKING $(1,0,1,0)$ AND INFINITE REACHABLE STATE - SPACE



THE REACHABILITY TREE OF THE PETRI NET OF FIGURE 8

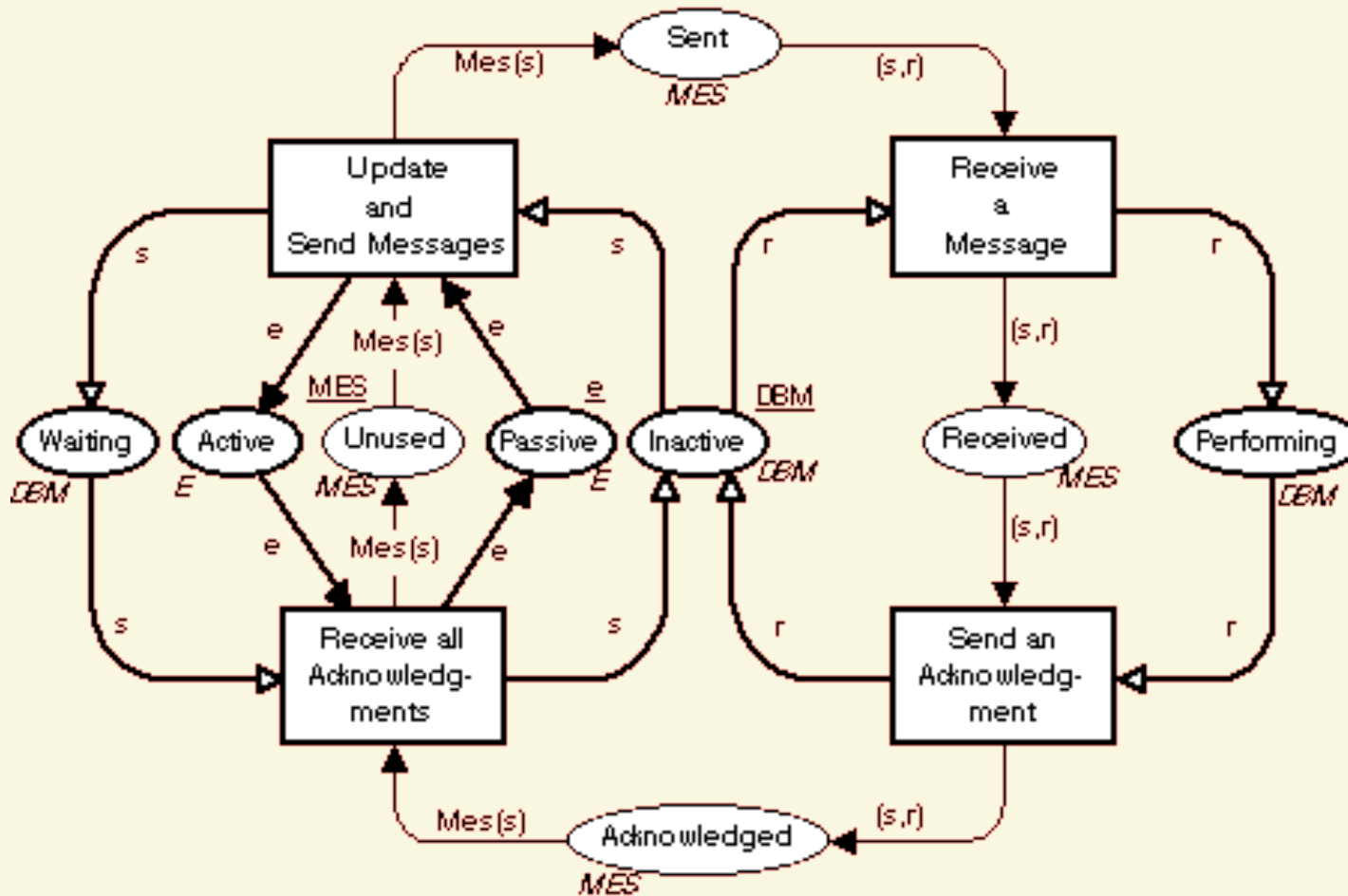
Other types of Petri Nets

- Token Machine
- Time Petri Nets
- Evaluation Nets
- Coloured Petri Nets (CPN)

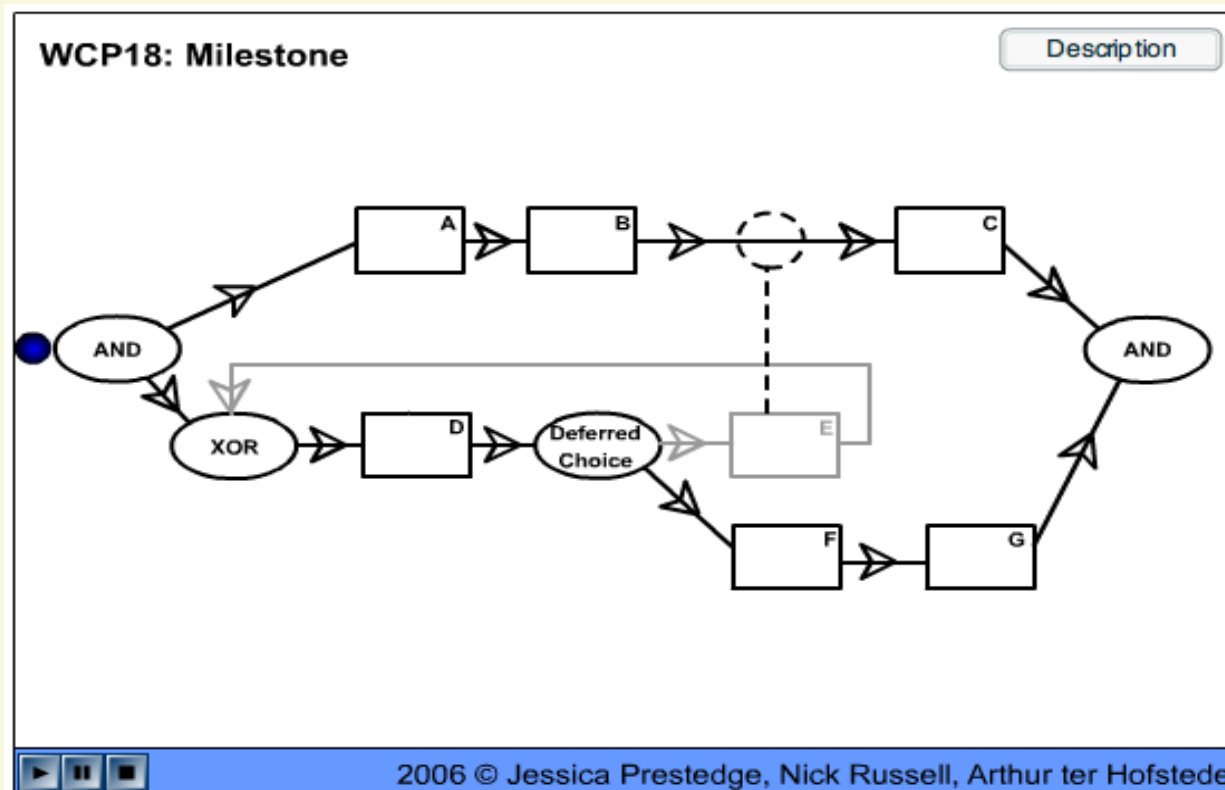
Coloured Petri Nets (CPN)

- CPN combine the strengths of ordinary Petri nets with the strengths of a high-level programming language.
- Petri nets provide the primitives for process interaction, while the programming language provides the primitives for the definition of data types and the manipulations of data values.
- CPN also offers more formal verification methods, known as state space analysis and invariant analysis. In this way it is possible to prove, in the mathematical sense of the word, that a system has a certain set of behavioural properties.

Coloured Petri Nets (CPN)



- CPN based workflow patterns
- Describe a full set of real life patterns and how those patterns can be modeled using CPN
- Compare commercial and open source projects



Questions?