Petri nets II
Extensions and Problems

Rushikesh K Joshi
CSE
IIT Bombay

Elementary Petri-Nets

At most one token per place

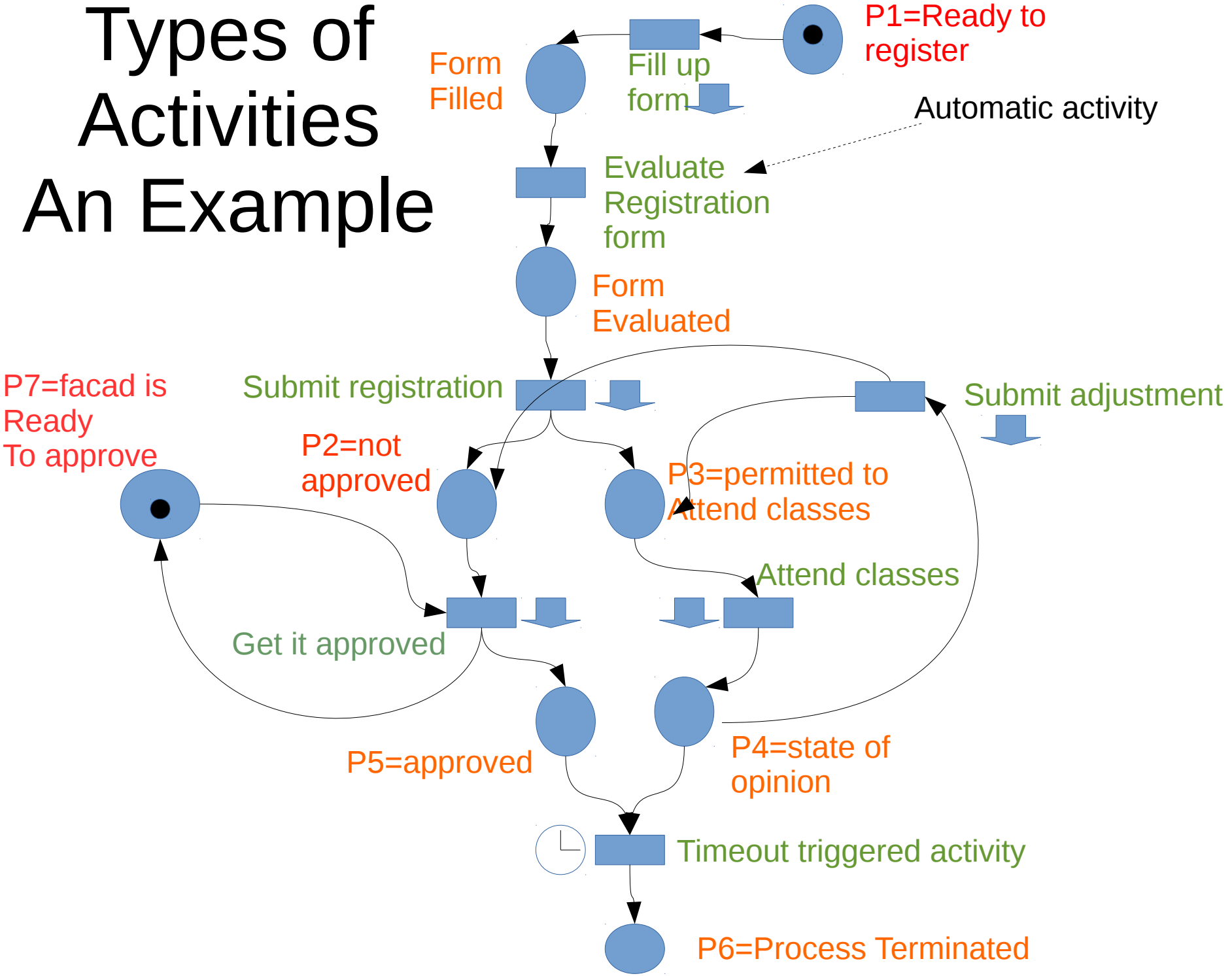
Workflow Nets

- Unique source place
- Unique sink place
- Connected
- Unique initial marking, unique terminal marking
- Well-formed – every transition is reachable, every marking is reachable, every marking terminates

Types Annotations for activities

- Automatic Activity
  - Computer can execute it fully
    - (when enabled, it is executed automatically such as by an algorithm, script task etc.)
- User Activity
  - A human being executes it
    - (though enable, it is done manually)
- Message Activity
  - An external message triggers the task instance
    - (though enabled, it requires a message to trigger it)
- Time Triggered Activity
  - Task needs to be trigged at a particular time, or after a certain period of timeout
    - (though enabled, time has to trigger it)

Types of Activities An Example

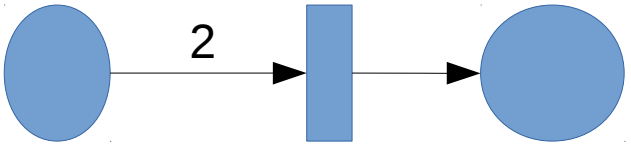


Classical Petri Nets

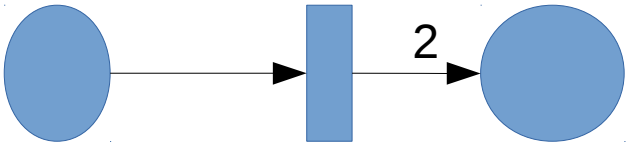
- A place can contain 0 or more tokens (unlimited number of them)
  - Infinite capacity net
  - A state in state space then needs to mention the count of tokens held in places
    - e.g. {1 p1, 2 p3, 4 p4} or in another notation (1,0,2,4) enumerating the numbers in each place
      - Note, this state is different from {2 P1, 2 P3, 4 p4}
- Arcs between places and transitions can have weights
  - Place to transition: requires those many tokens in the place for transition to fire
  - Transitions to place: produces those many tokens if the transition fires
- Source transition: no input place- is unconditionally enabled all the time
- Sink tansition: no output place – consumes token.
- Pure petri net: without self loops
- Ordinary petri nets: arc weight is 1 (default)
- Finite capacity net: places are marked with capacities

Showing Arcs with weights

Default arc weight is 1
Similarly default capacity is inf
But if the net is elementary, capacity
Is 1, and arc weight is 1

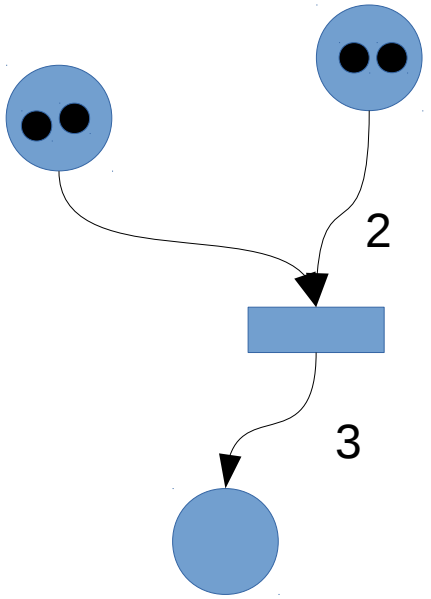


2 tokens required to enable the transition

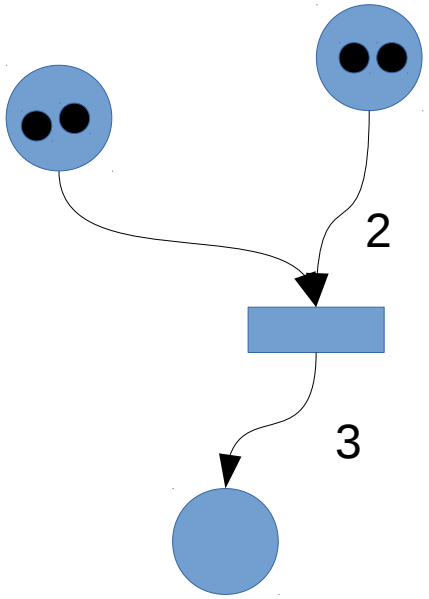


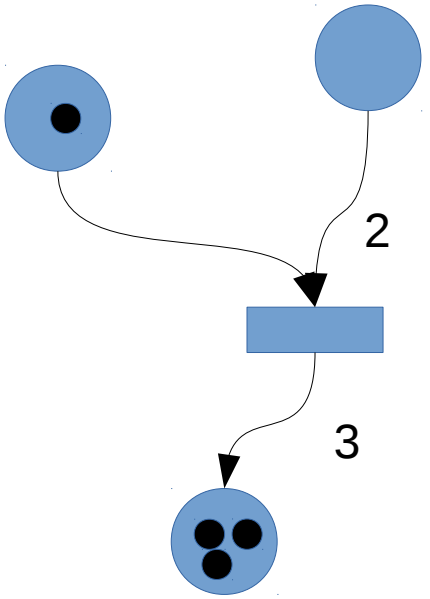
2 tokens are produced after firing the transition

Example
what would be the marking after firing the transition?

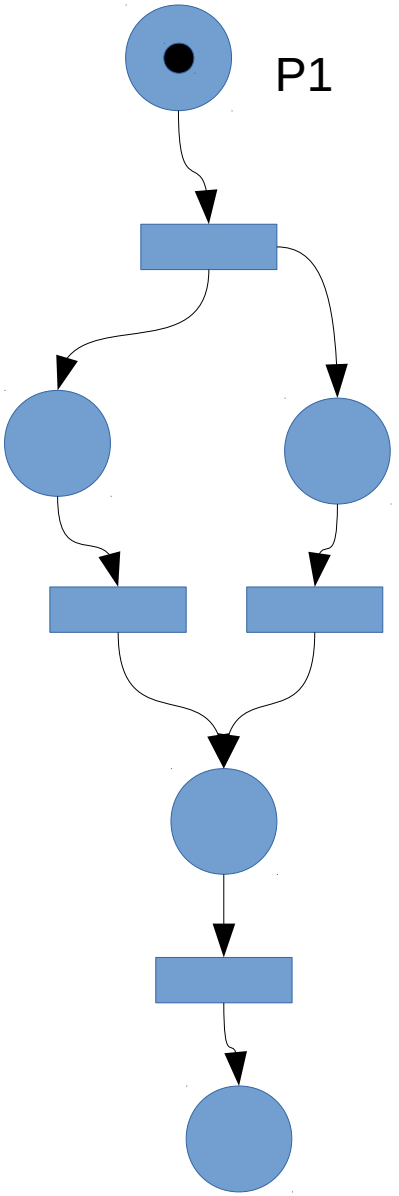


After this, no more progress





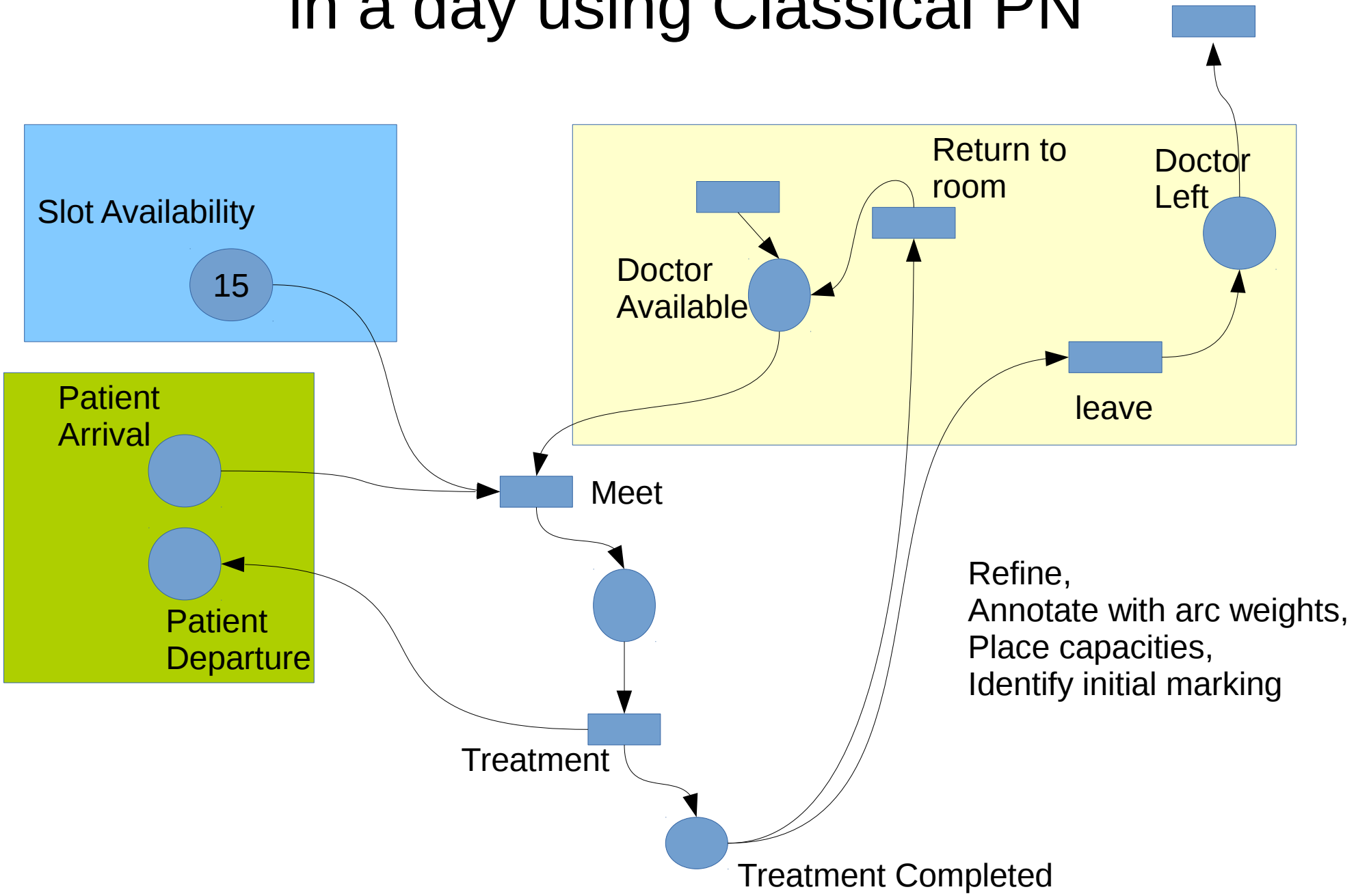
Places with multiple tokens



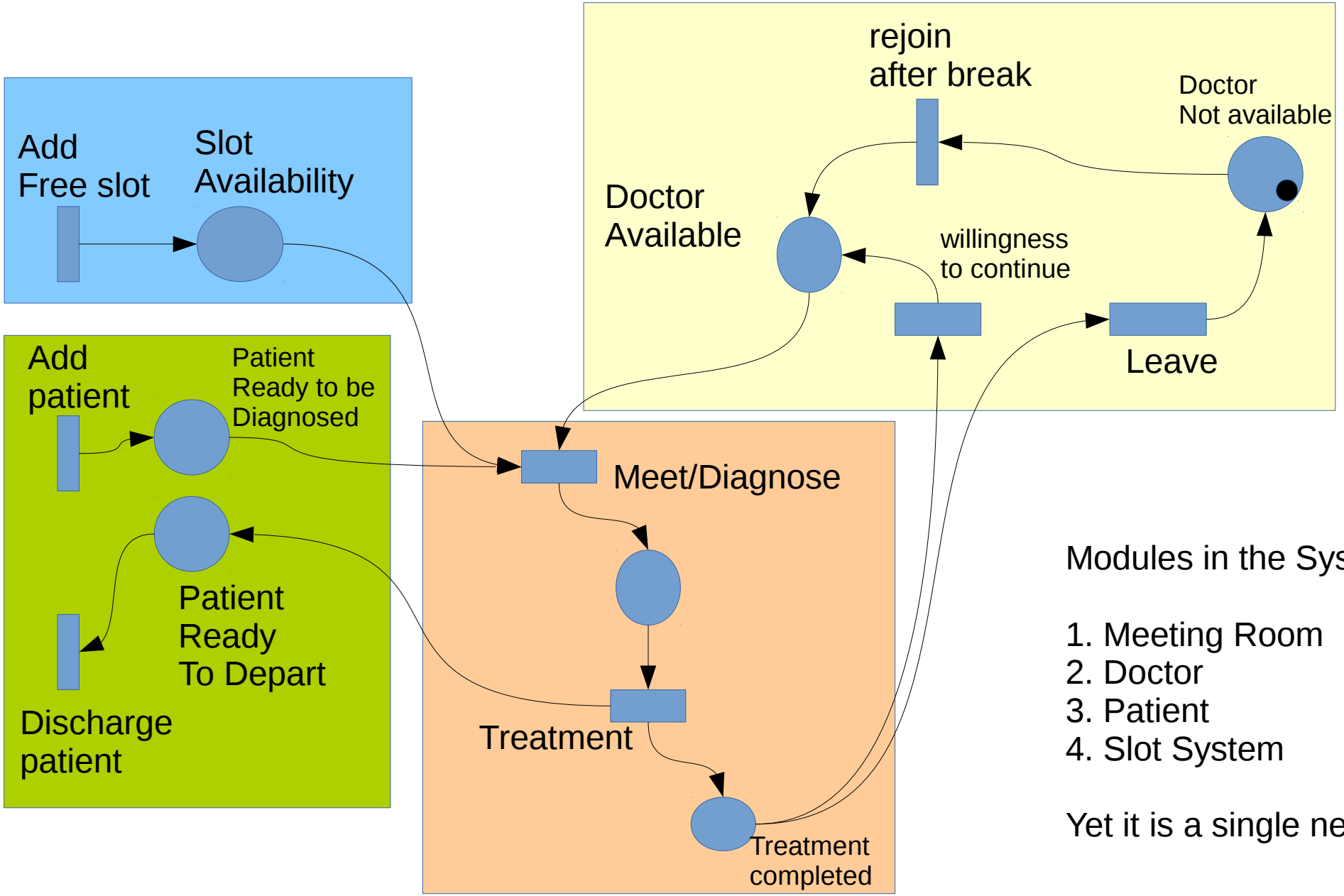
Build state space

Initial marking: {P1}

Exercise: Doctors and Patients in a day using Classical PN



An Improved Model

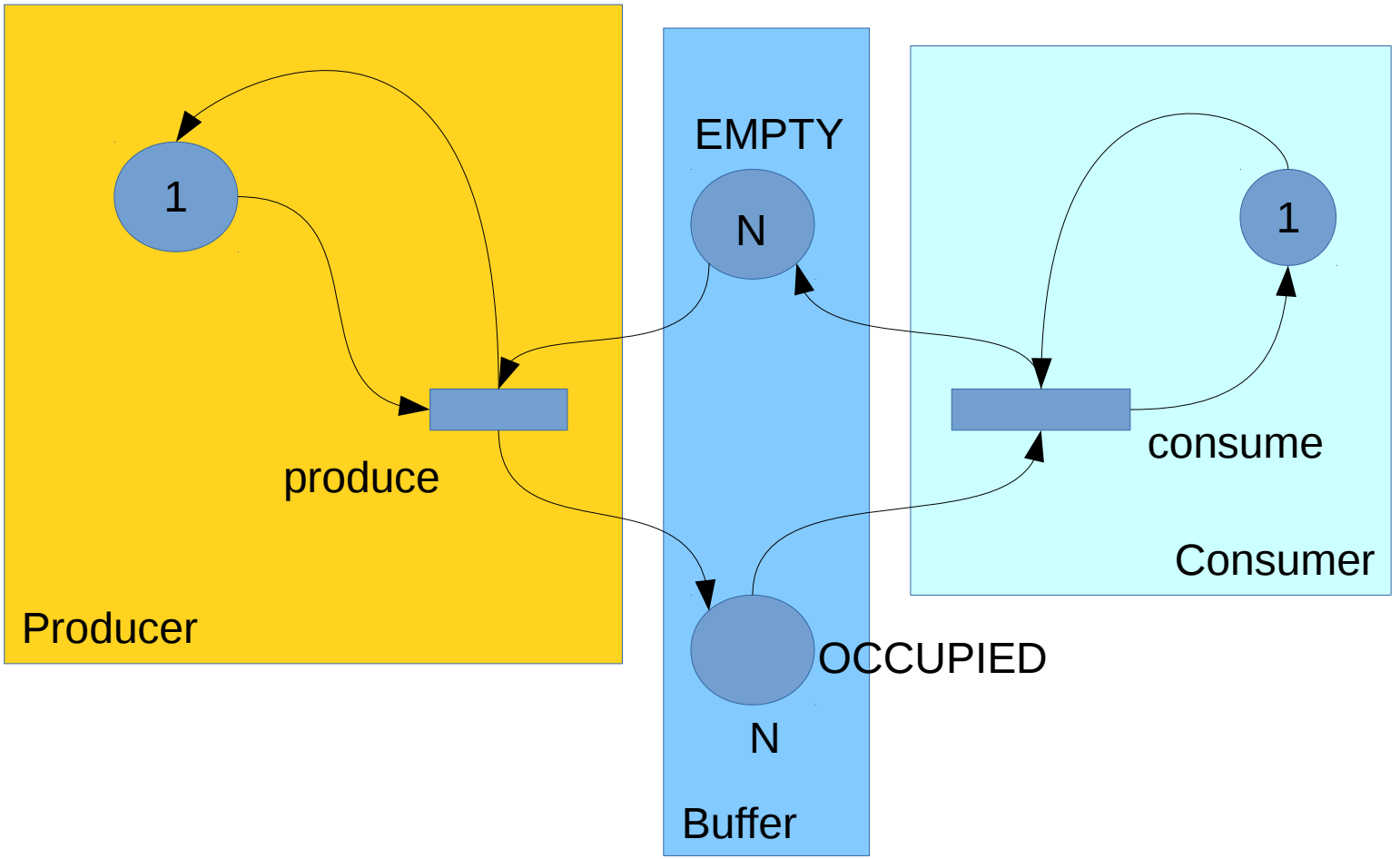


Modules in the System:

1. Meeting Room
2. Doctor
3. Patient
4. Slot System

Yet it is a single net

Producers and Consumers



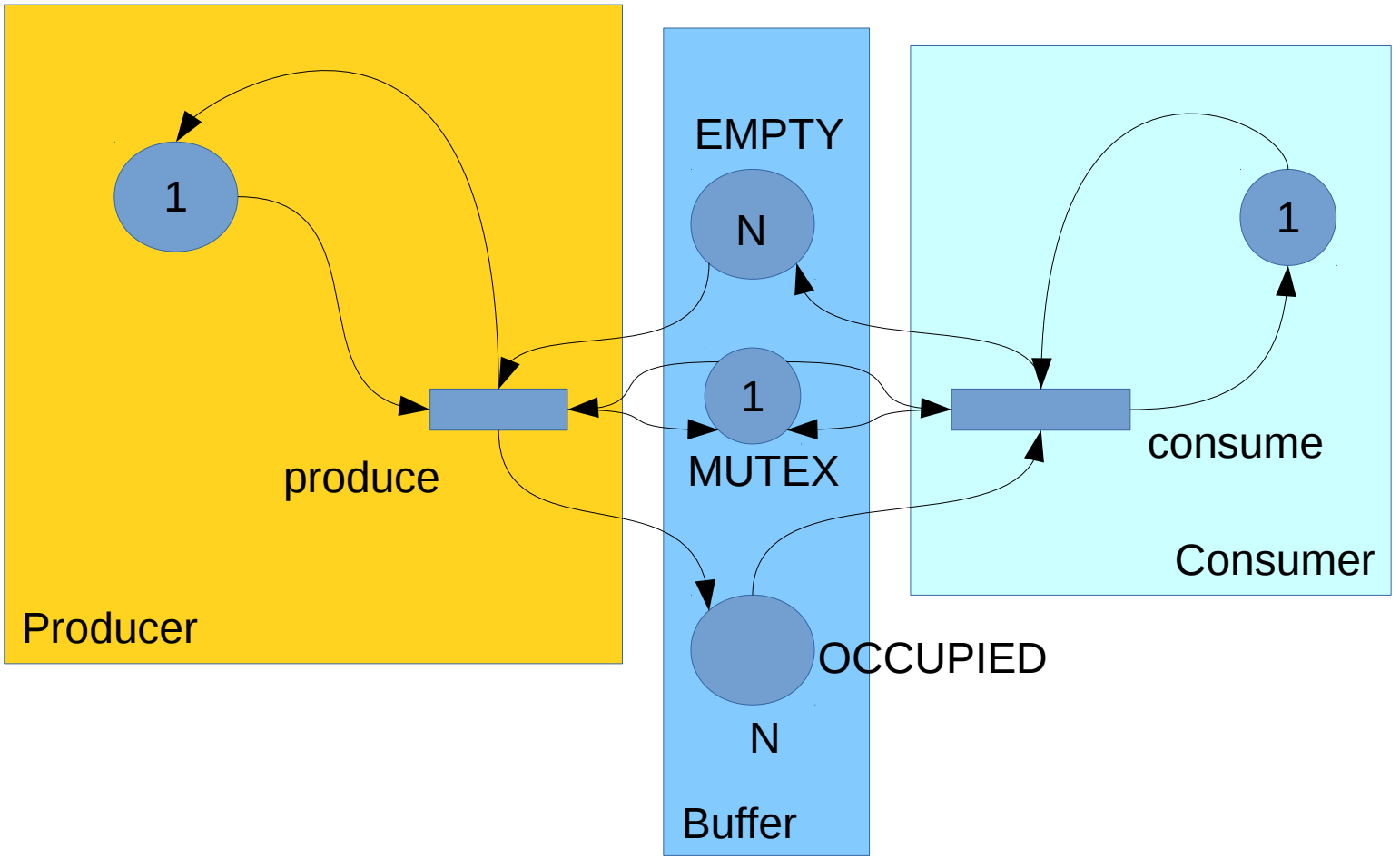
N

N Tokens available inside the place

N

Place can accept at most N tokens

Producers and Consumers with MUTEX For the buffer



N N Tokens available inside the place

N Place can accept at most N tokens