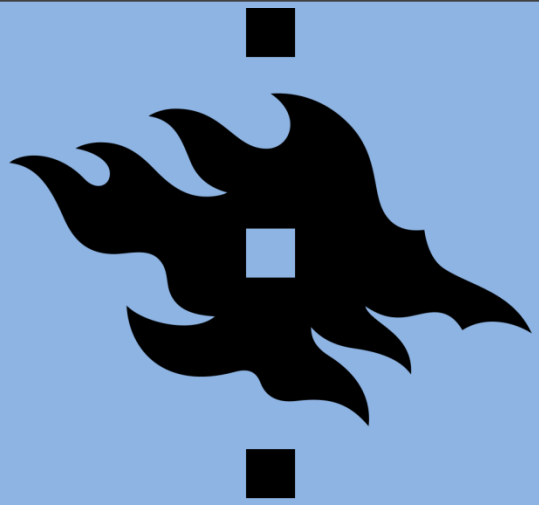


DISTRIBUTED SYSTEMS

Lecture 6 – Coordination (Part II)

Huber Flores, Xiang Su, Pan Hui
{firstname.lastname}@helsinki.fi

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Helsinki, Finland, 2019.

Recap

- Place: passive element
- Transition: active element
- Arc: causal relation
- Token: elements subject to change

The state (space) of a Petri net is a distribution of tokens across its places. Transition firing move the net from one state to another.

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

COORDINATION

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Mapping a system to a petri net

- Role of a token
- Role of a place
- Role of a transition

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Role of a token

Tokens can play the following roles:

- A **physical object**, for example a product, a part, a drug, a person;
- An **information object**, for example a message, a signal, a report;
- A **collection of objects**, for example a truck with products, a warehouse with parts, or an address file;
- An **indicator of a state**, for example the indicator of the state in which a process is, or the state of an object;
- An **indicator of a condition**: the presence of a token indicates whether a certain condition is fulfilled.

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Role of a place

- A type of **communication medium**, like a telephone line, a middleman, or a communication network;
- A **buffer**: for example, a depot, a queue or a post bin;
- A **geographical location**, like a place in a warehouse, office or hospital;
- A possible **state or state condition**: for example, the floor where an elevator is, or the condition that a specialist is available.

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Role of a transition

- An **event**: for example, starting an operation, the death of a patient, a change seasons or the switching of a traffic light from red to green;
- A **transformation of an object**, like adapting a product, updating a database, or updating a document;
- A **transport of an object**: for example, transporting goods, or sending a file.

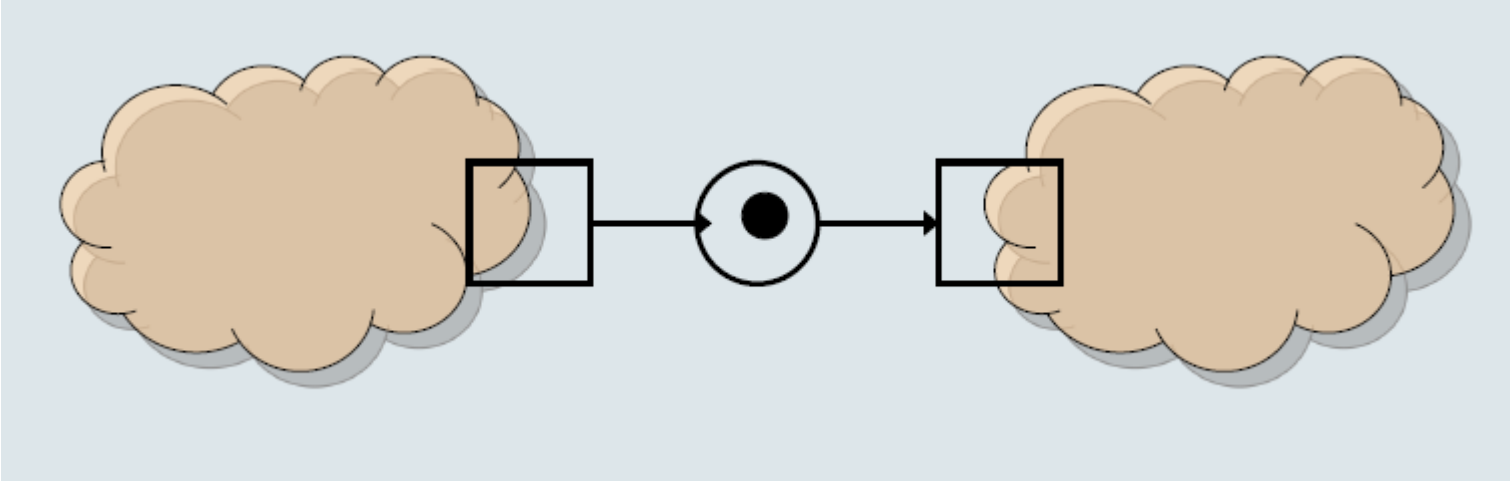
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Typical network structures

- Causality
- Parallelism (AND-split - AND-join)
- Choice (XOR-split – XOR-join)
- Iteration (XOR-join - XOR-split)
- Capacity constraints
  - Feedback loop
  - Mutual exclusion
  - Alternating

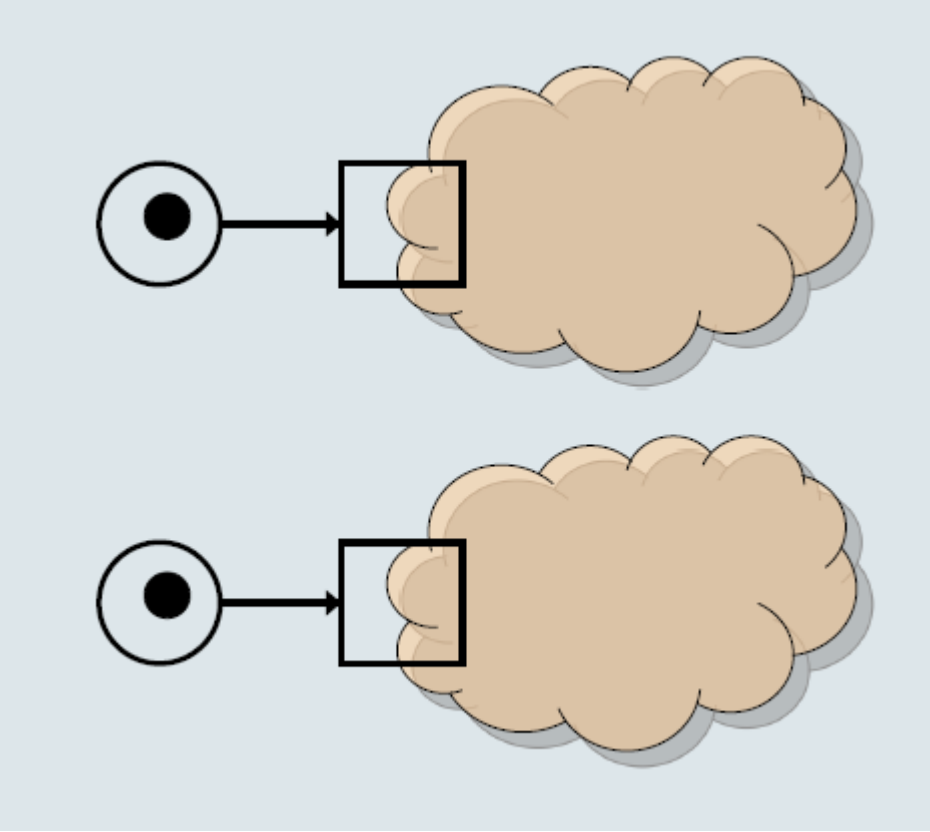
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Causality



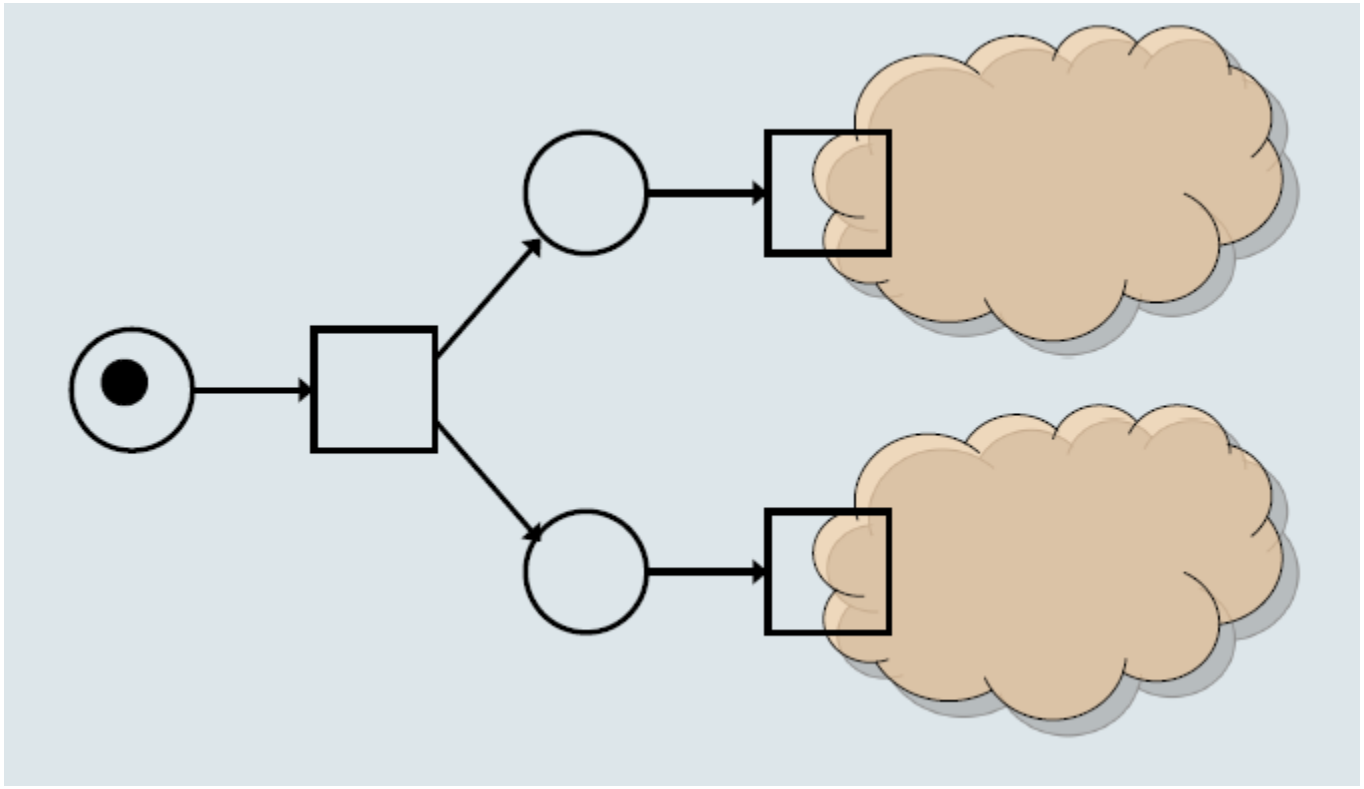
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Parallelism

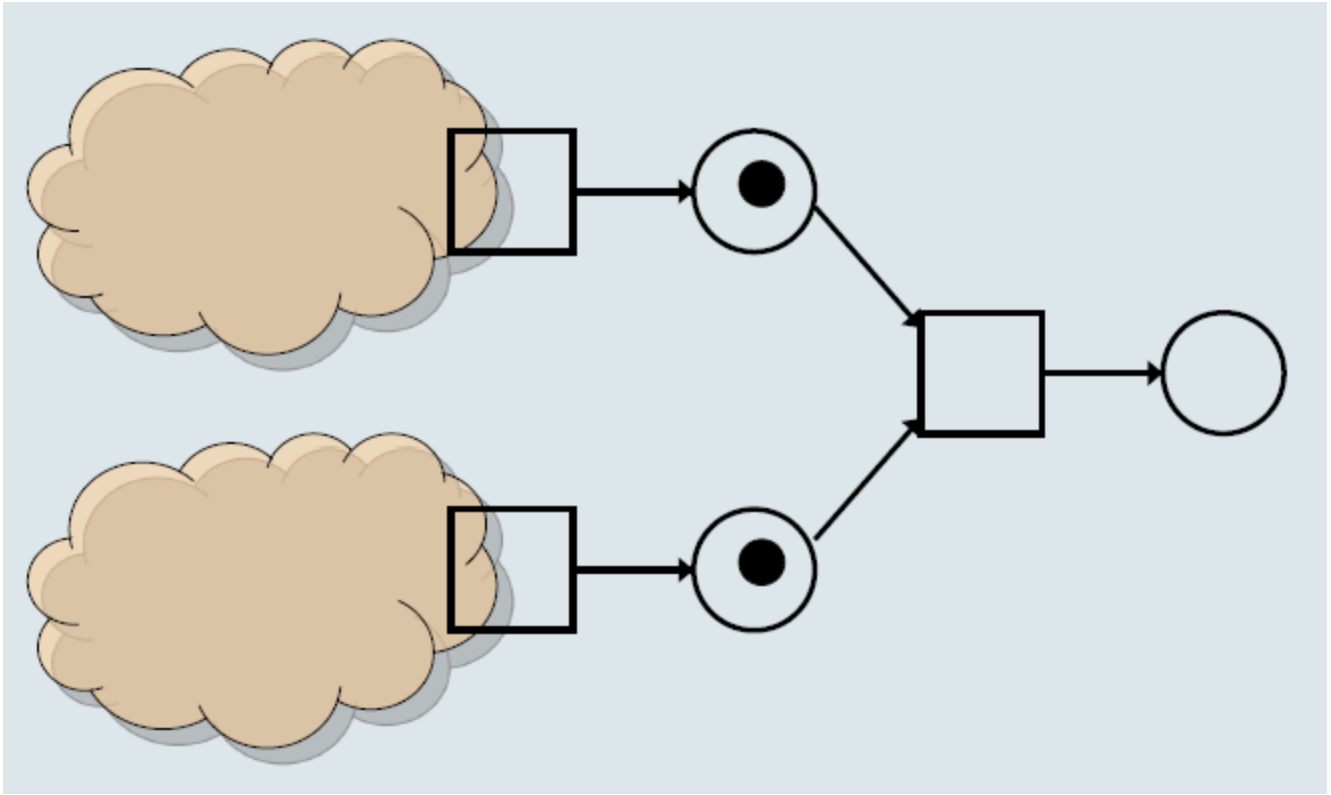


HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Parallelism: AND-split

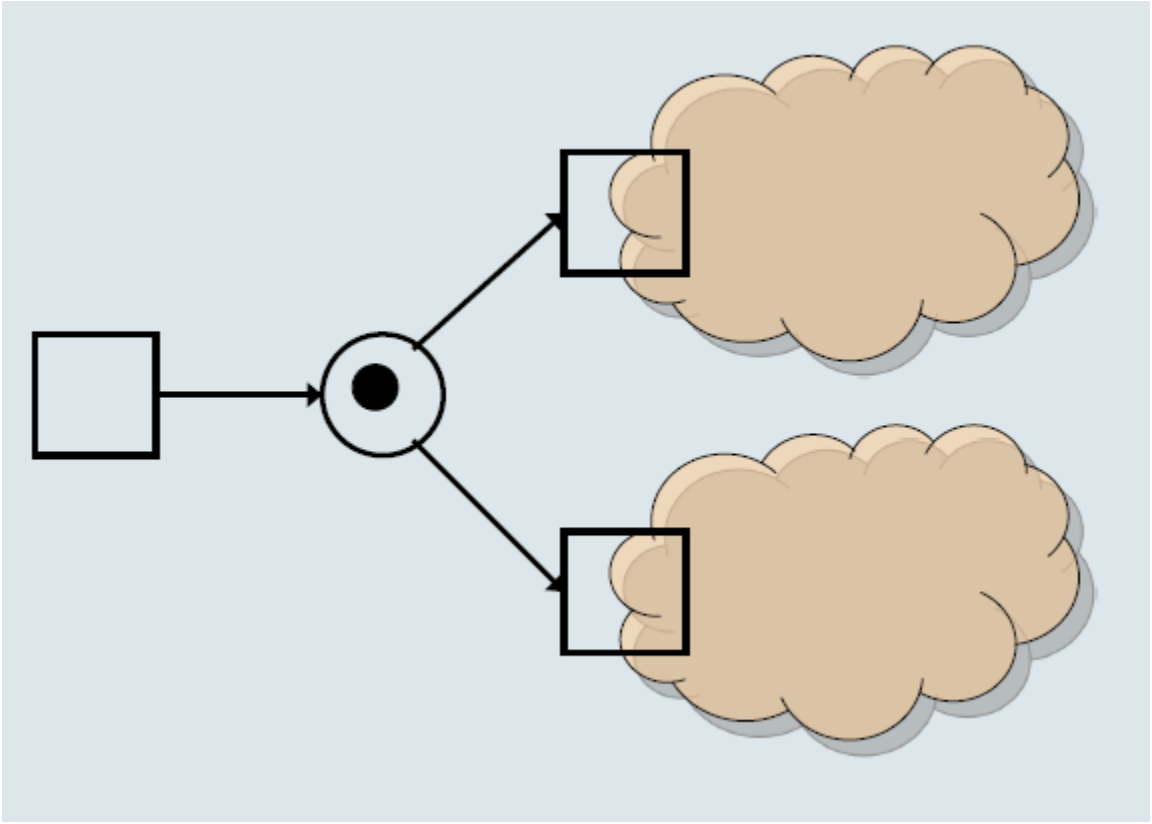


Parallelism: AND-join



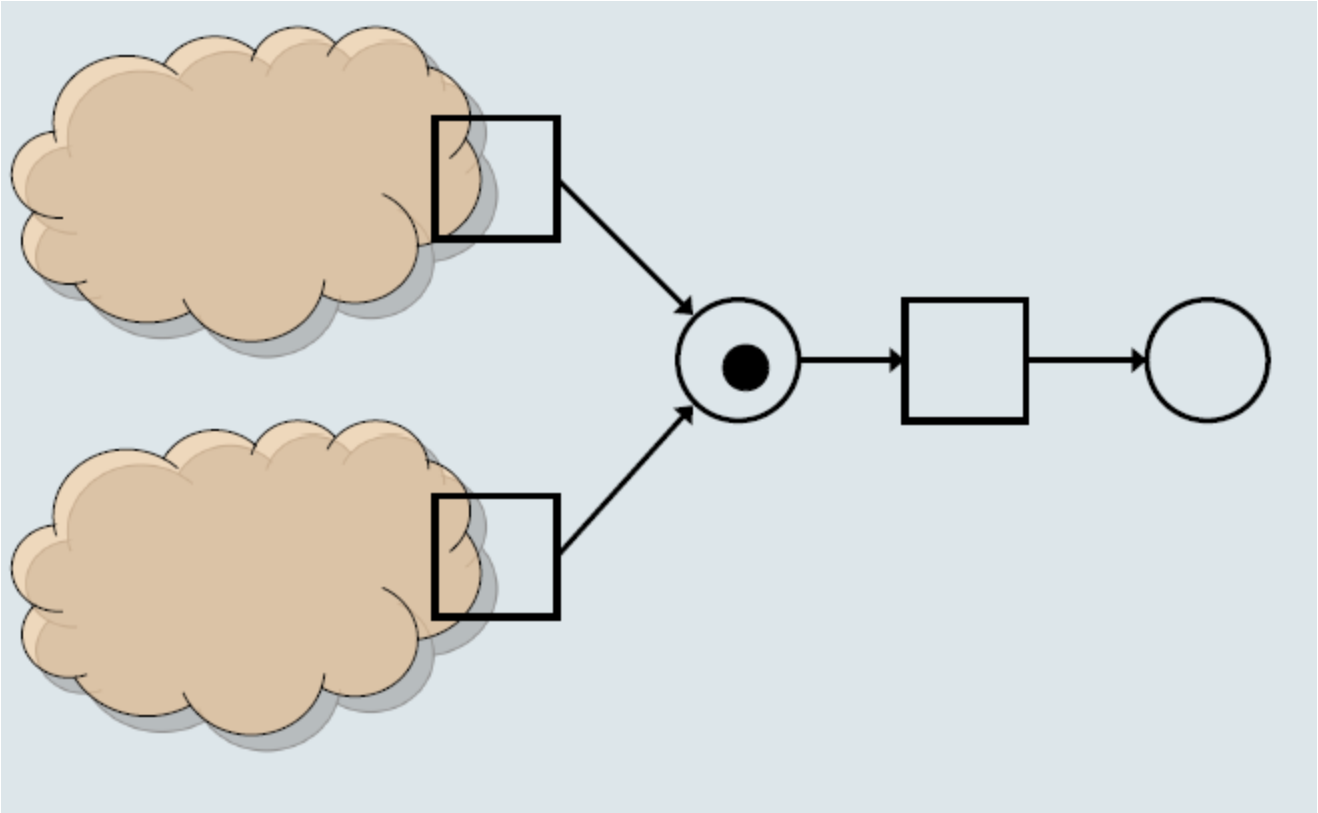
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Choice: XOR-split



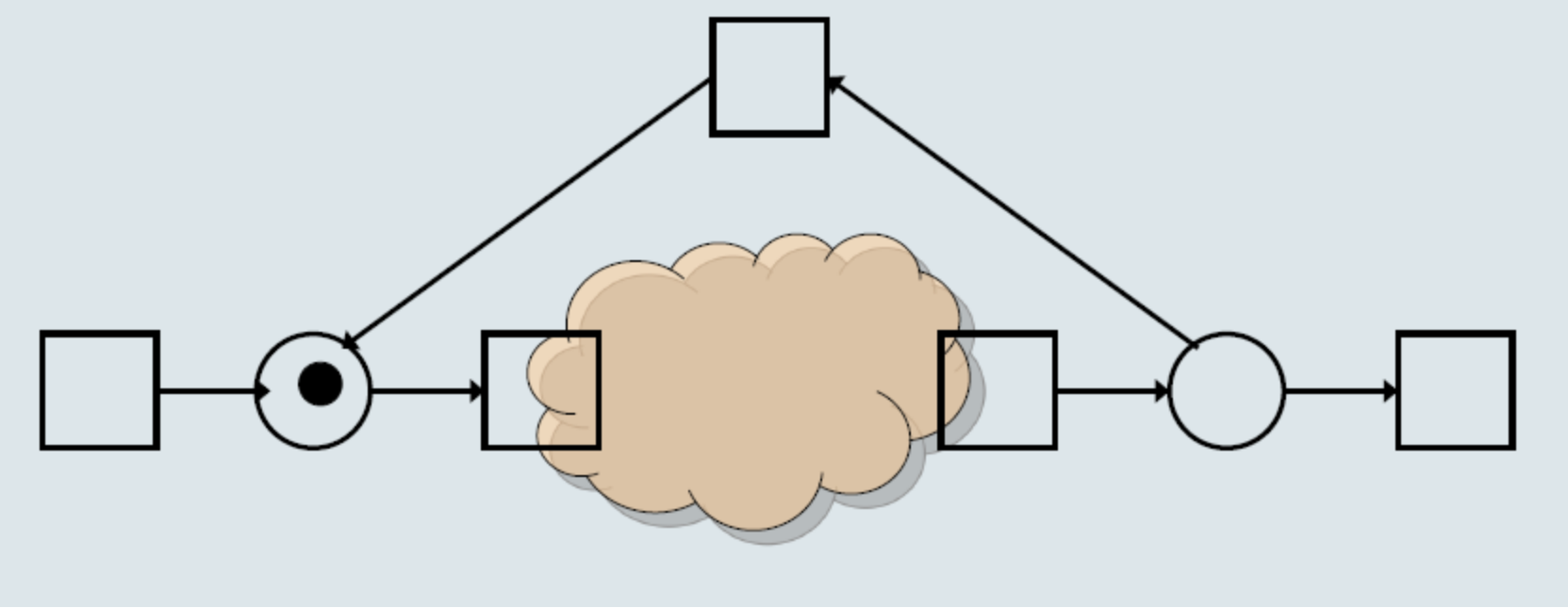
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Choice: XOR-join



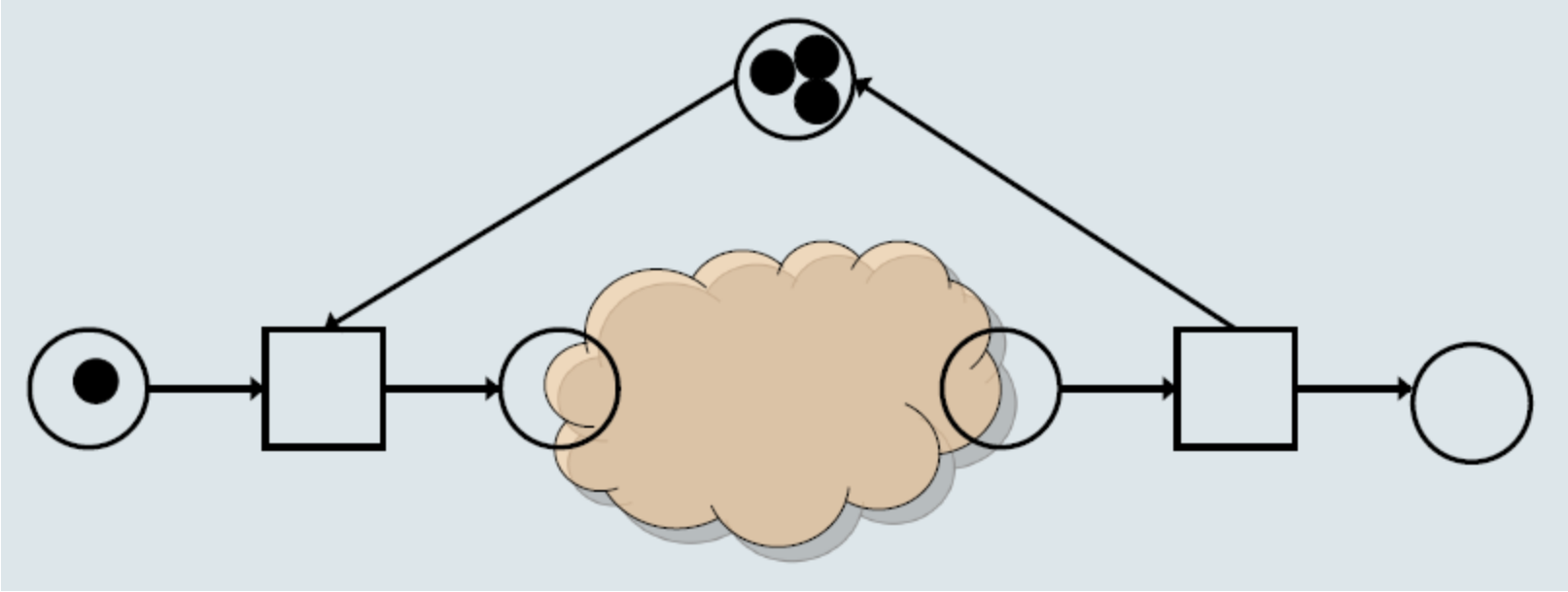
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Iteration: 1 or more times



HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

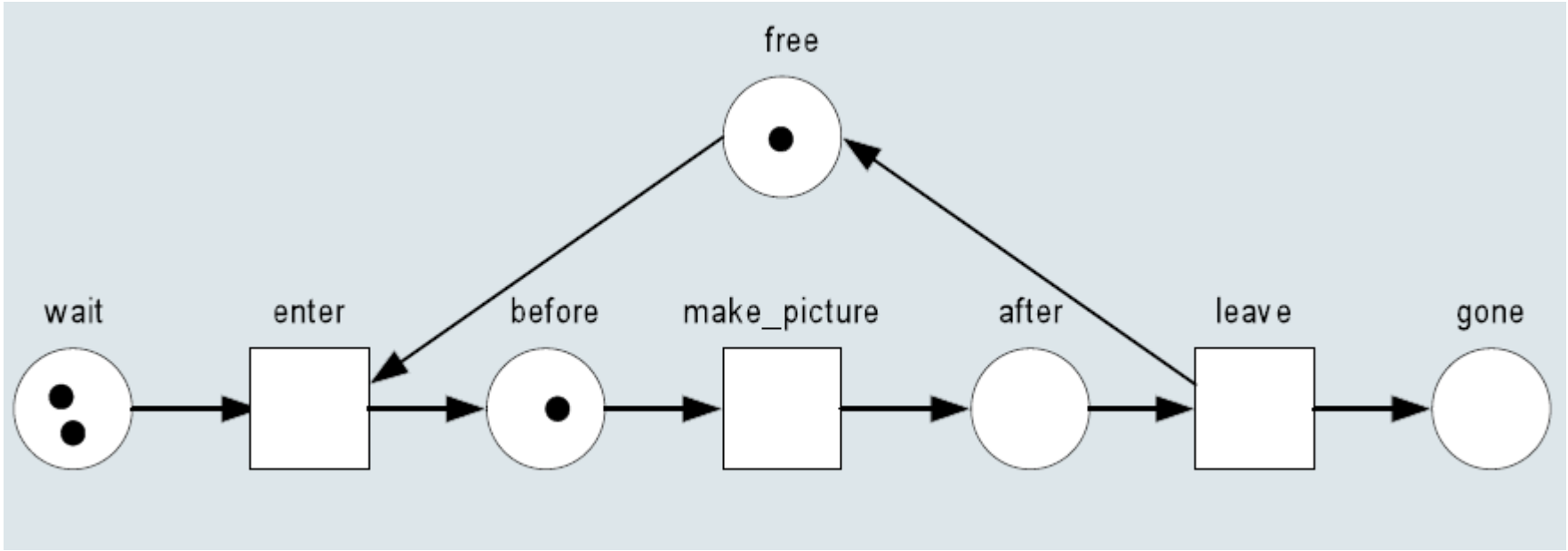
Capacity constraints: feedback loop



HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

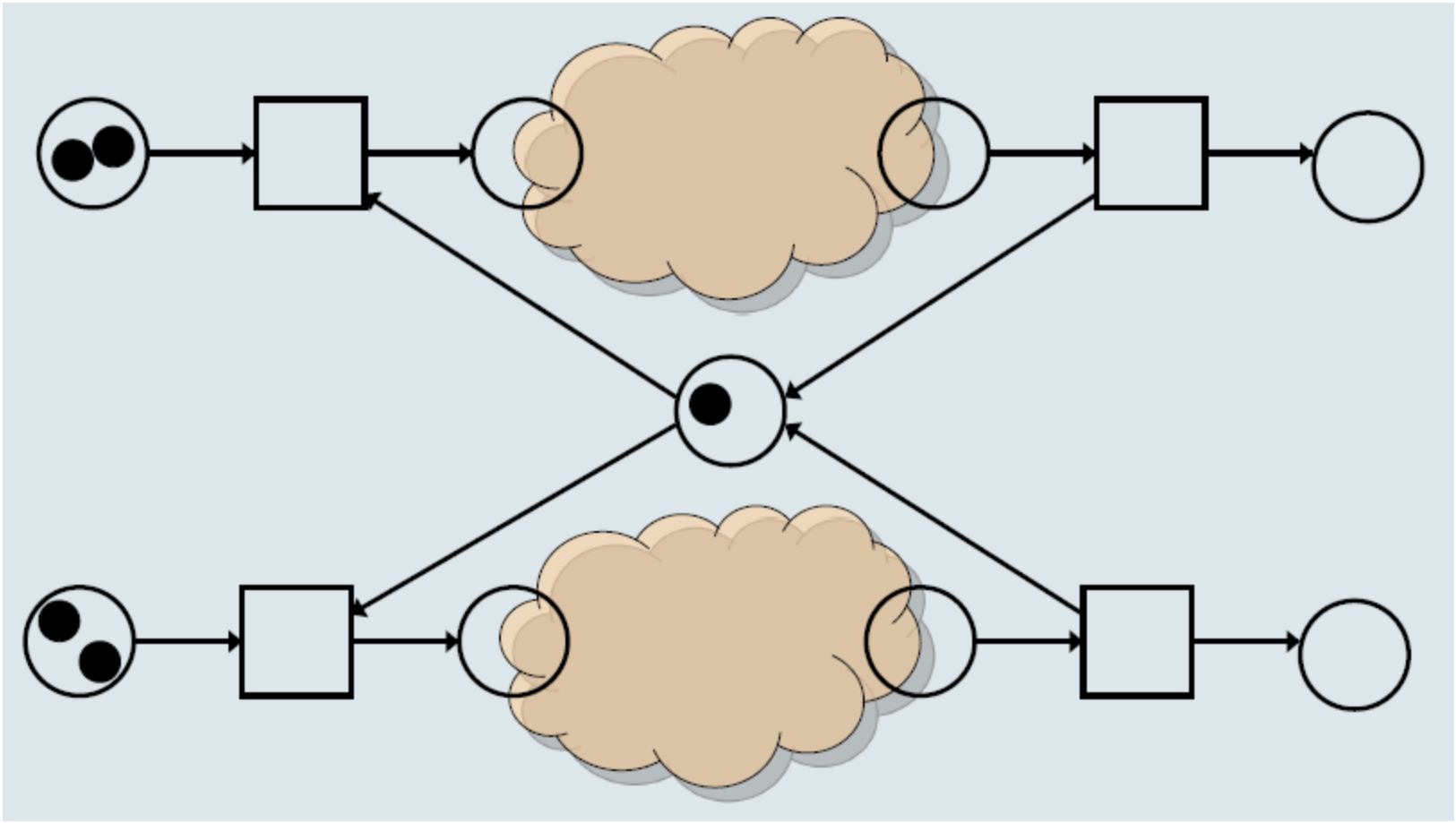
Capacity constraint: example

- The room has a capacity of 2, 1 person is in the room

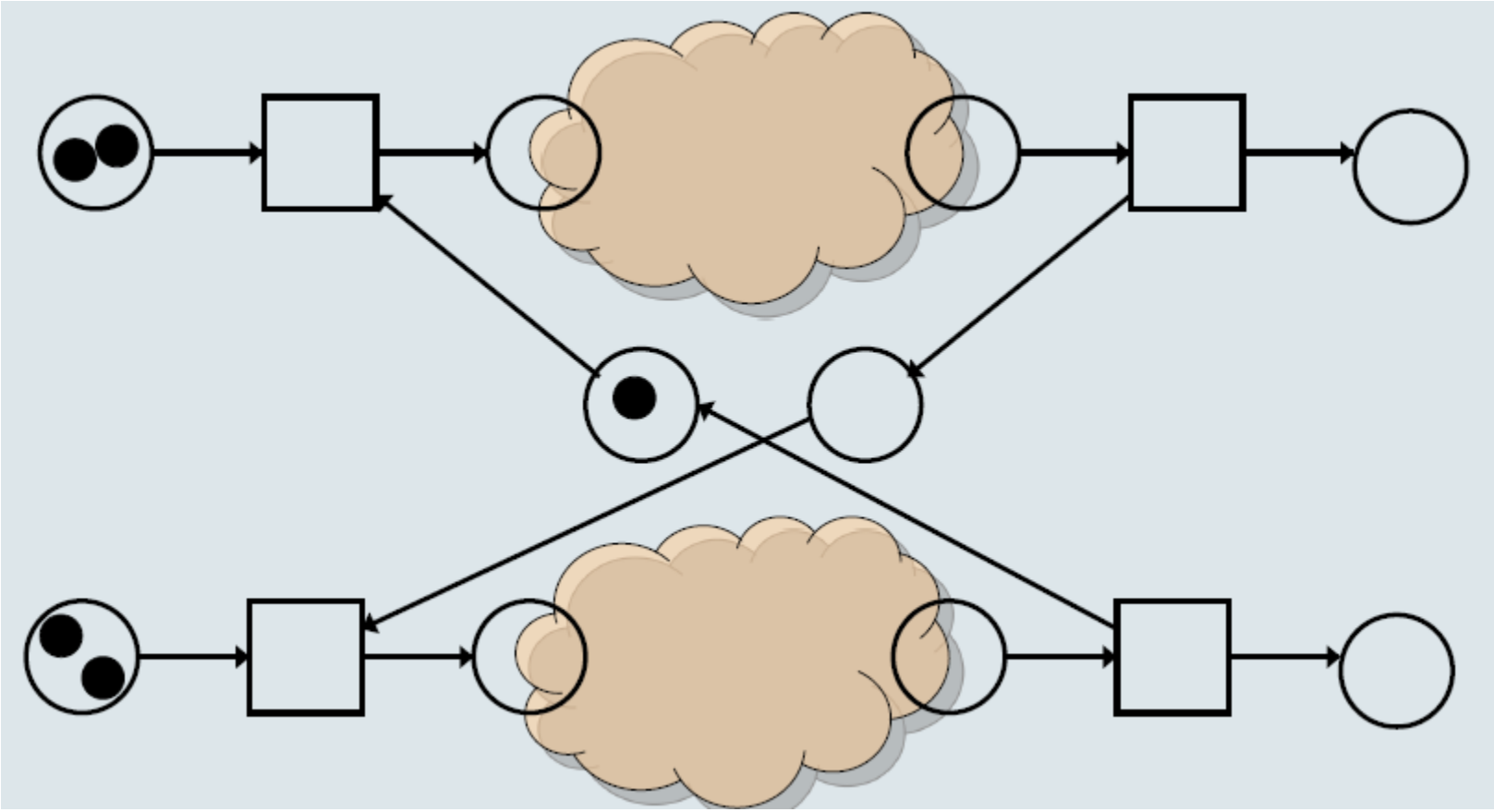


HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Capacity constraints: mutual exclusion



Capacity constraints: alternating



Mutual exclusion pattern: example

- Synchronized traffic light
  - How to make them alternate?

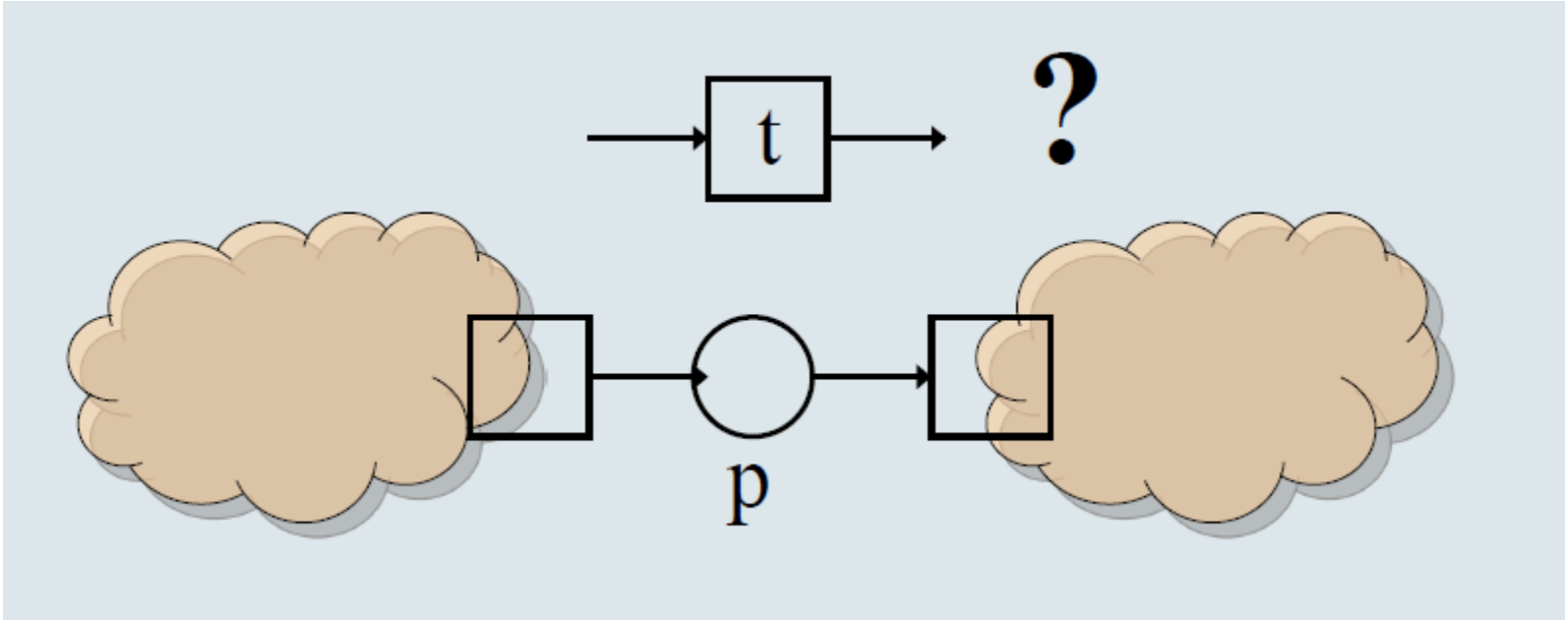
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Hands-on session

- Download the solution
  - https://drive.google.com/file/d/1e777Z1CHterNoI4J693l4-ElJvxb98-s/view?usp=sharing
  - goo.gl/syeEUz

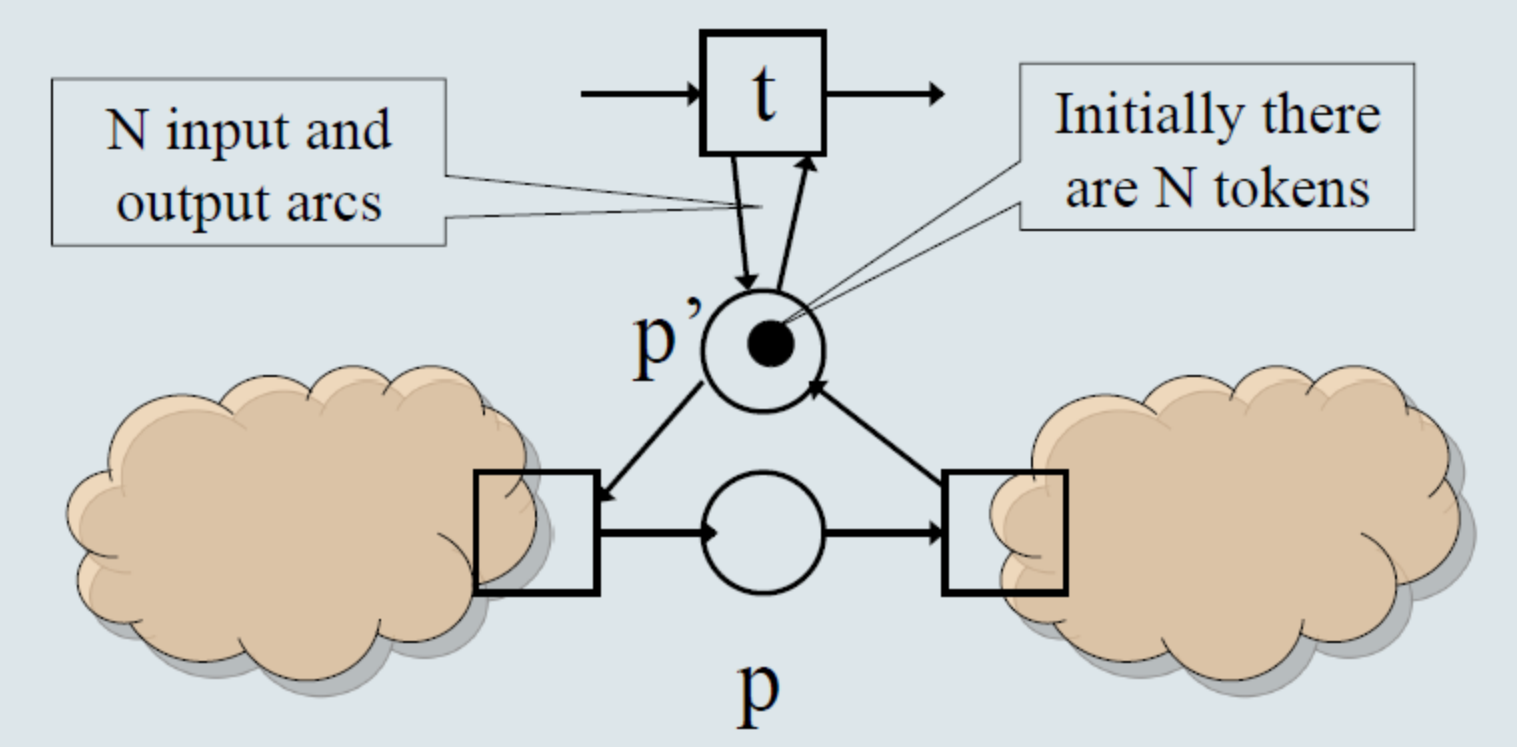
Modeling problem (1): Zero testing

- Transition *t* should fire if place *p* is empty.



HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

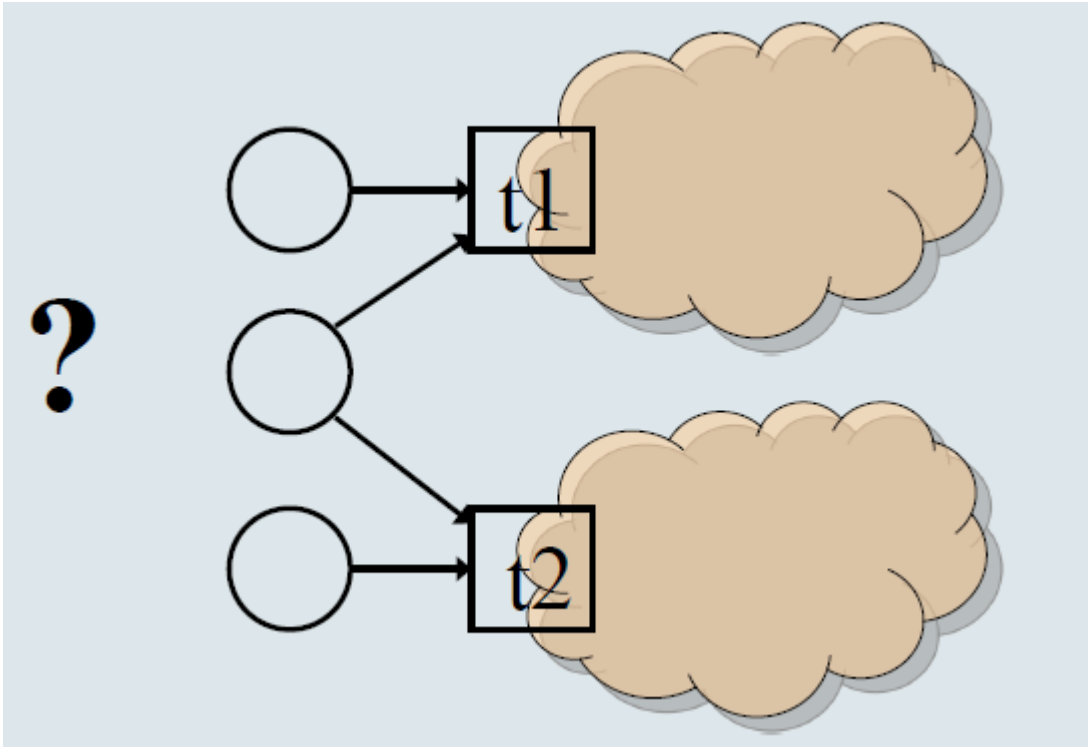
Solution



HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Modeling problem (2): Priority

- Transition t1 has priority over t2



Hint:
Similar to
Zero
testing

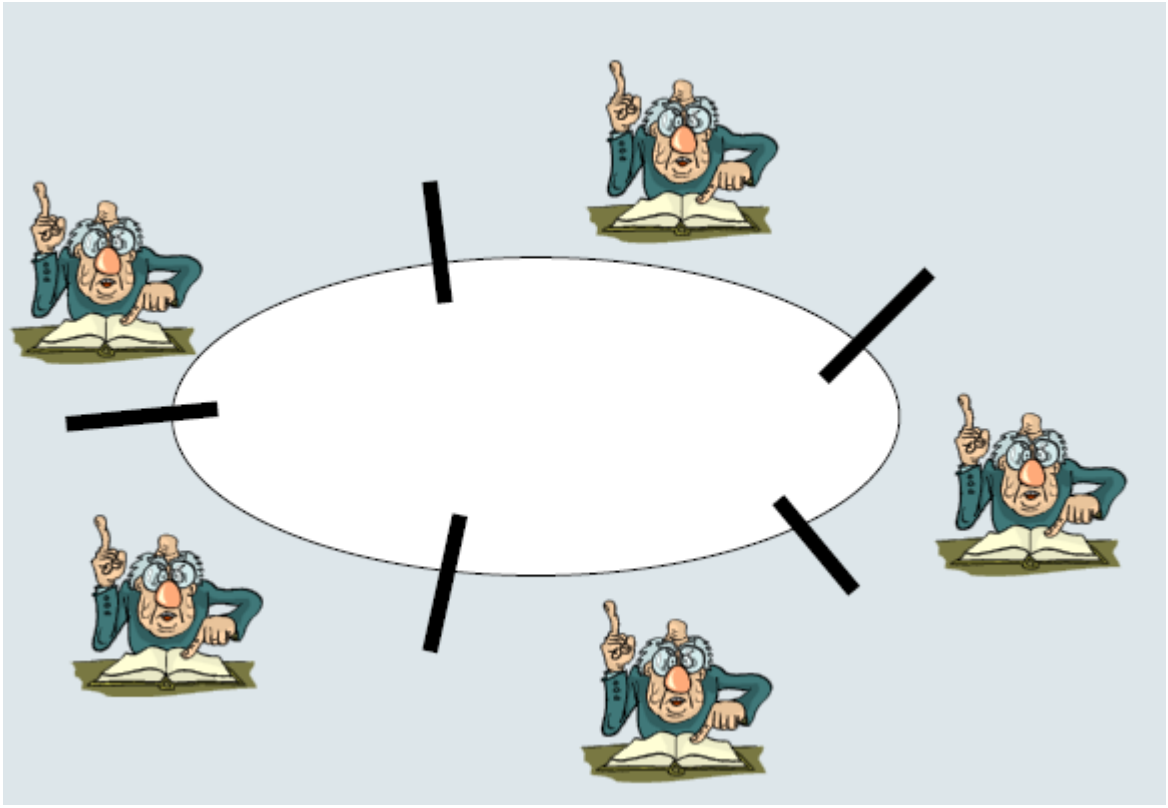
Theoretical results

- Extensions have been proposed to tackle these problems, e.g., inhibitor arcs.
- These extensions extend the modeling power (Turing completeness*).
- Without such an extension not Turing complete.
- Still certain questions are difficult/expensive to answer or even undecidable (e.g., equivalence of two nets)

Exercise: Dining philosophers (Basic version)

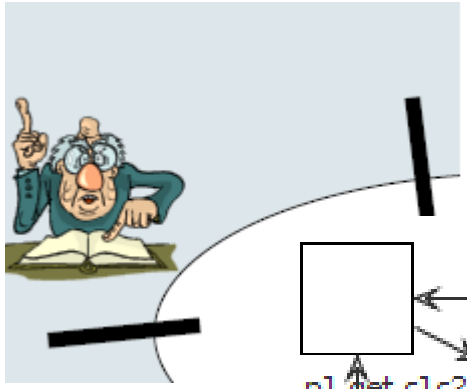
- 5 philosophers sharing 5 chopsticks: chopsticks are located in-between philosophers
- A philosopher is either in state eating or thinking and needs two chopsticks to eat.
- Model as a Petri net.

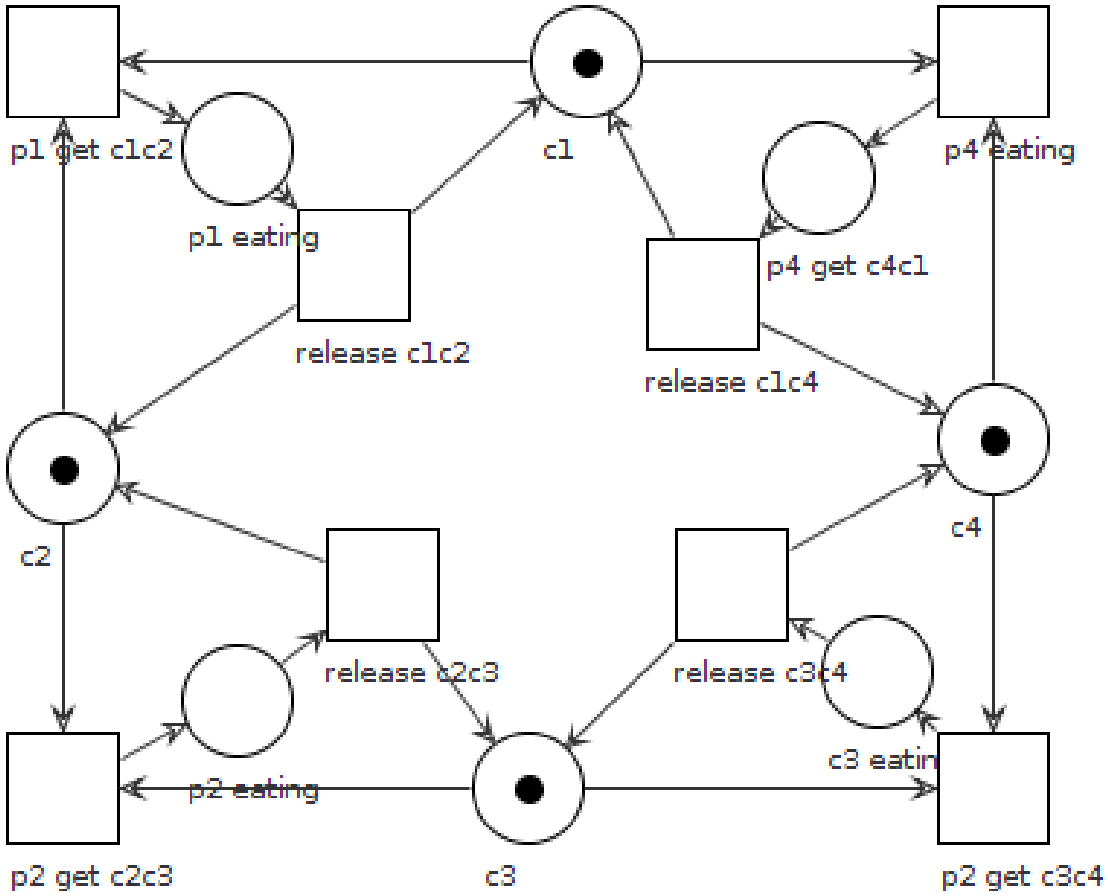
Exercise: Dining philosophers (Basic version)



HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Solution





HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Hands-on session

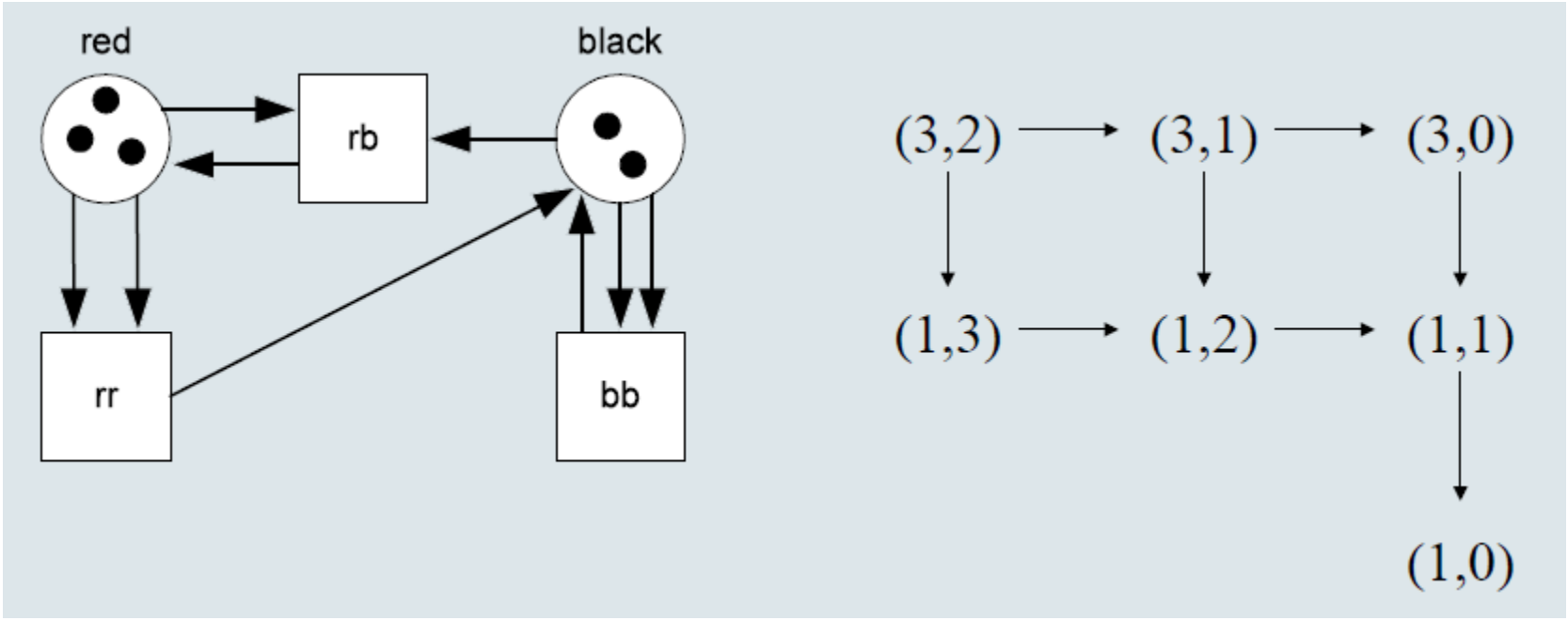
- Download exercise solution
  - https://drive.google.com/file/d/1FB9dp6S-jbQvmcQHolD7RZPkJDDe5gQB/view?usp=sharing
  - goo.gl/ezvnbS

Petri net analysis: Reachability Graph

- Graph containing one node for each reachable state; an edge indicates that the system can move from the source state to the target state through one transition firing
- The reachability graph can be calculated as follows:
  - Let X be the set containing just the initial state and let Y be the empty set.
  - Take an element x of X and add this to Y. Calculate all states reachable for x by firing some enabled transition. Each successor state that is not in Y is added to X.
  - If X is empty stop, otherwise goto 2.

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

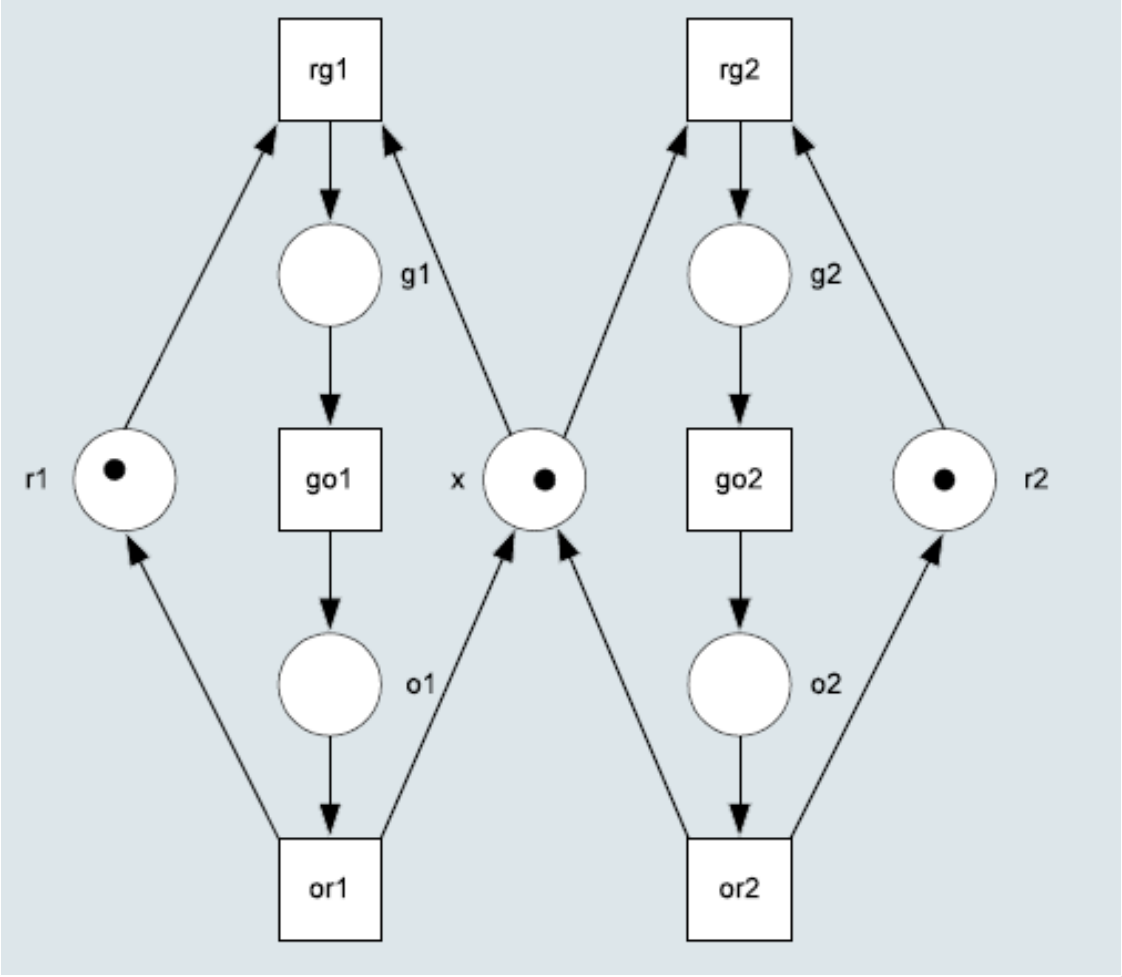
Example



- Nodes in the reachability graph can be represented by a vector “(3,2)” or as “3 red + 2 black”. The latter is useful for “sparse states” (i.e., few places are marked).

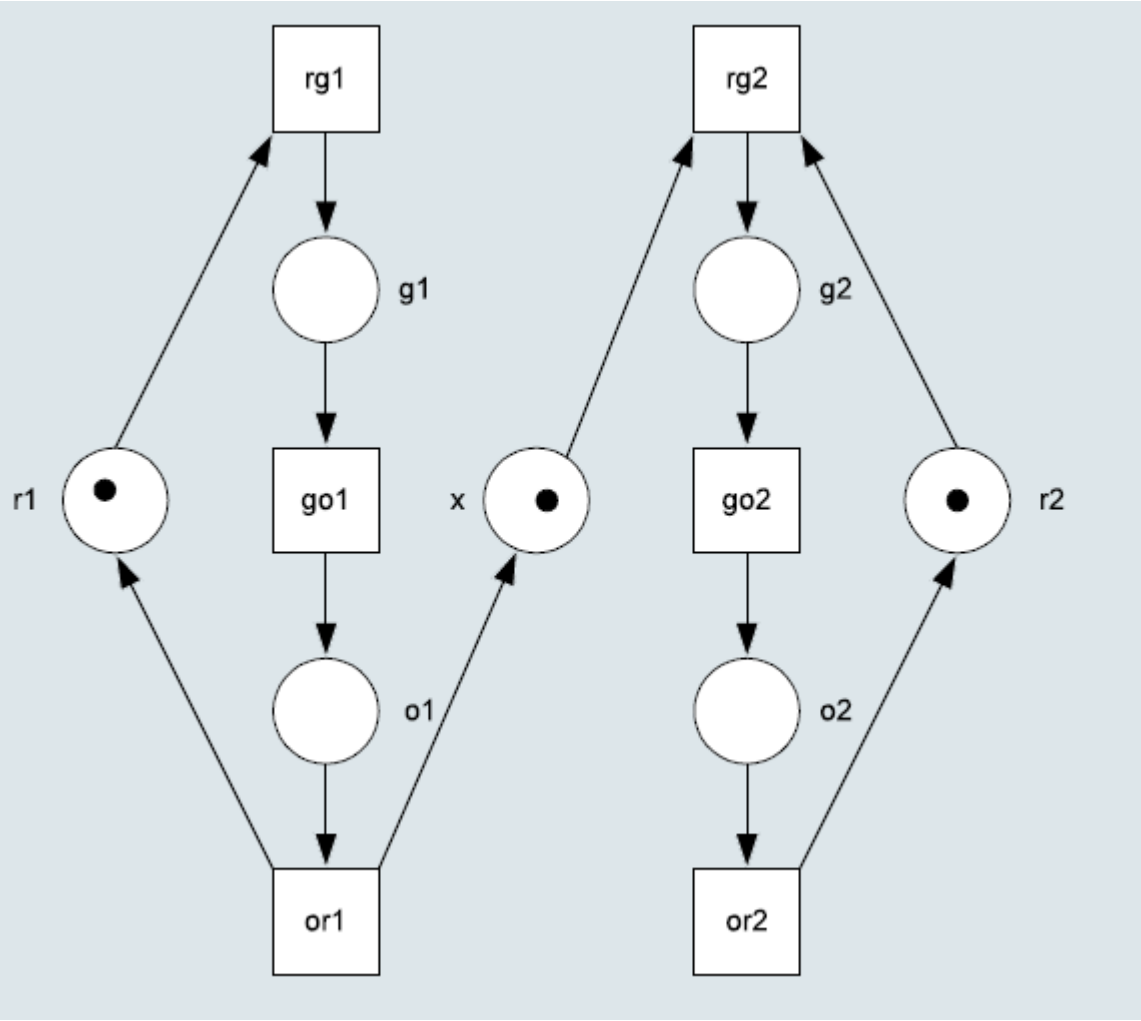
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Exercise: Give the reachability graph using both notations



HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Infinite reachability graph



HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Different types of states

- **Initial state**: Initial distribution of tokens.
- **Reachable state**: Reachable from initial state.
- **Final state** (also referred to as “dead states”): No transition is enabled.
- **Home state** (also referred to as home marking): It is always possible to return (i.e., it is reachable from any reachable state).

How to recognize these states in the reachability graph?

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

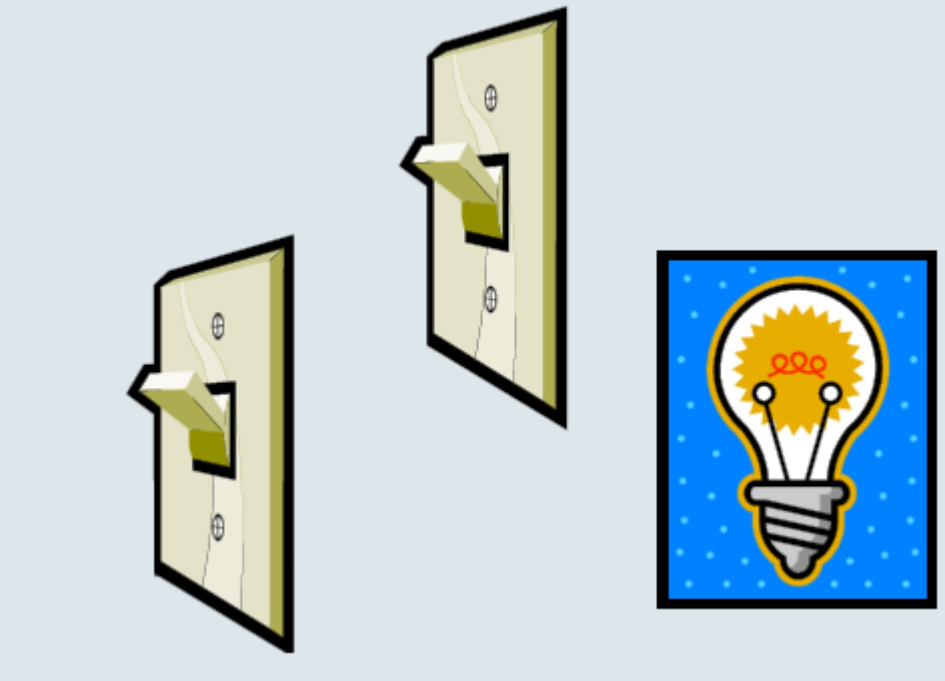
Exercise: Witness statements

- As part of the process of handling insurance claims, we need to handle witness statements.
- There may any number of witnesses per claim. The number is determined when the claim is lodged. After an initialization step (one per claim), each of the witnesses is registered and contacted (N witnesses per claim in parallel). Only after all witnesses have contacted a report is made (one report per claim).
- Can you model this using a Petri net?

Exercise: Two switches

- Consider a room with two switches and one light. The light is **on** or **off**. The switches are in state **up** or **down**. At any time any of the switches can be used to turn the light on or off.
- Model this as a Petri net.
- Give the reachability graph

Exercise: Two switches



Your tasks

- For next lecture session:
  - Resolve the exercises (Use your practical session time for this also)
  - Include exercises with ✓ in your deliverables for Task 2

References

- Part of this material is from lectures given by
  - M. Dumas, University of Tartu
  - W. van der Aalst, Eindhoven University of Technology

QUESTIONS ☺