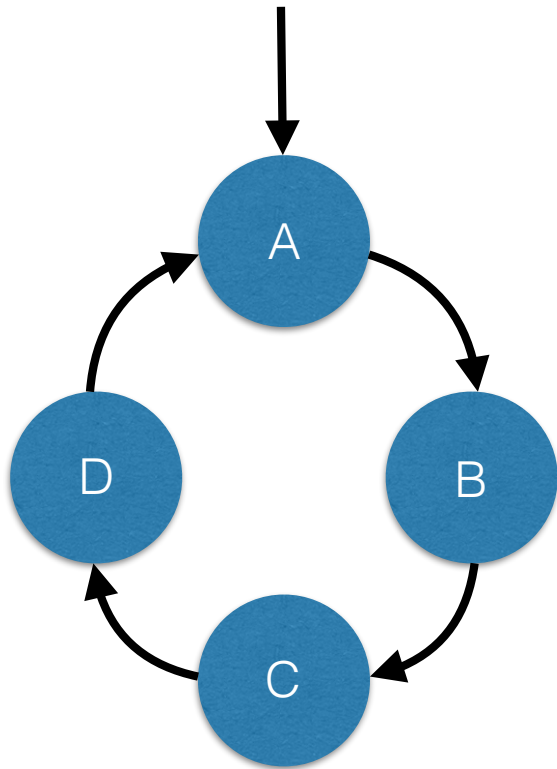


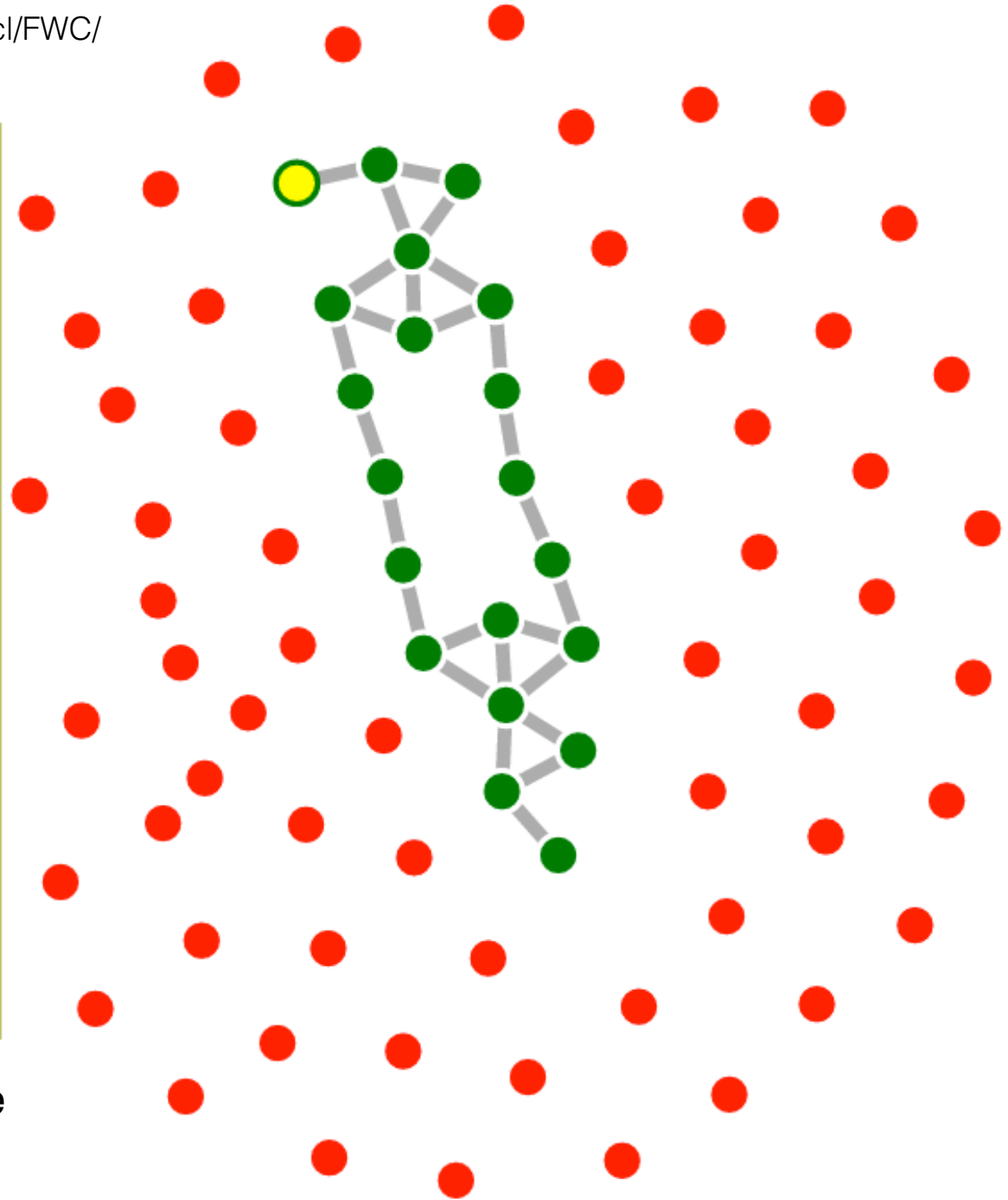
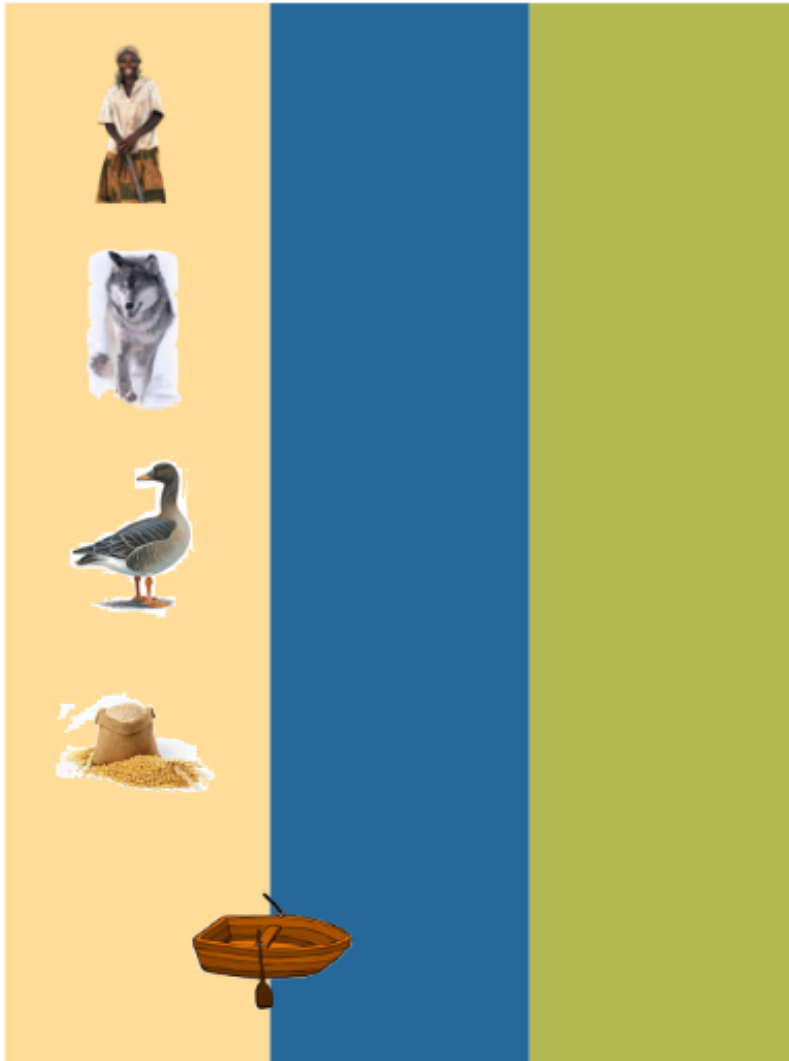
# Finite-State Machines (Automata) lecture 12

cl

- a simple form of computation
- used widely
- one way to find patterns



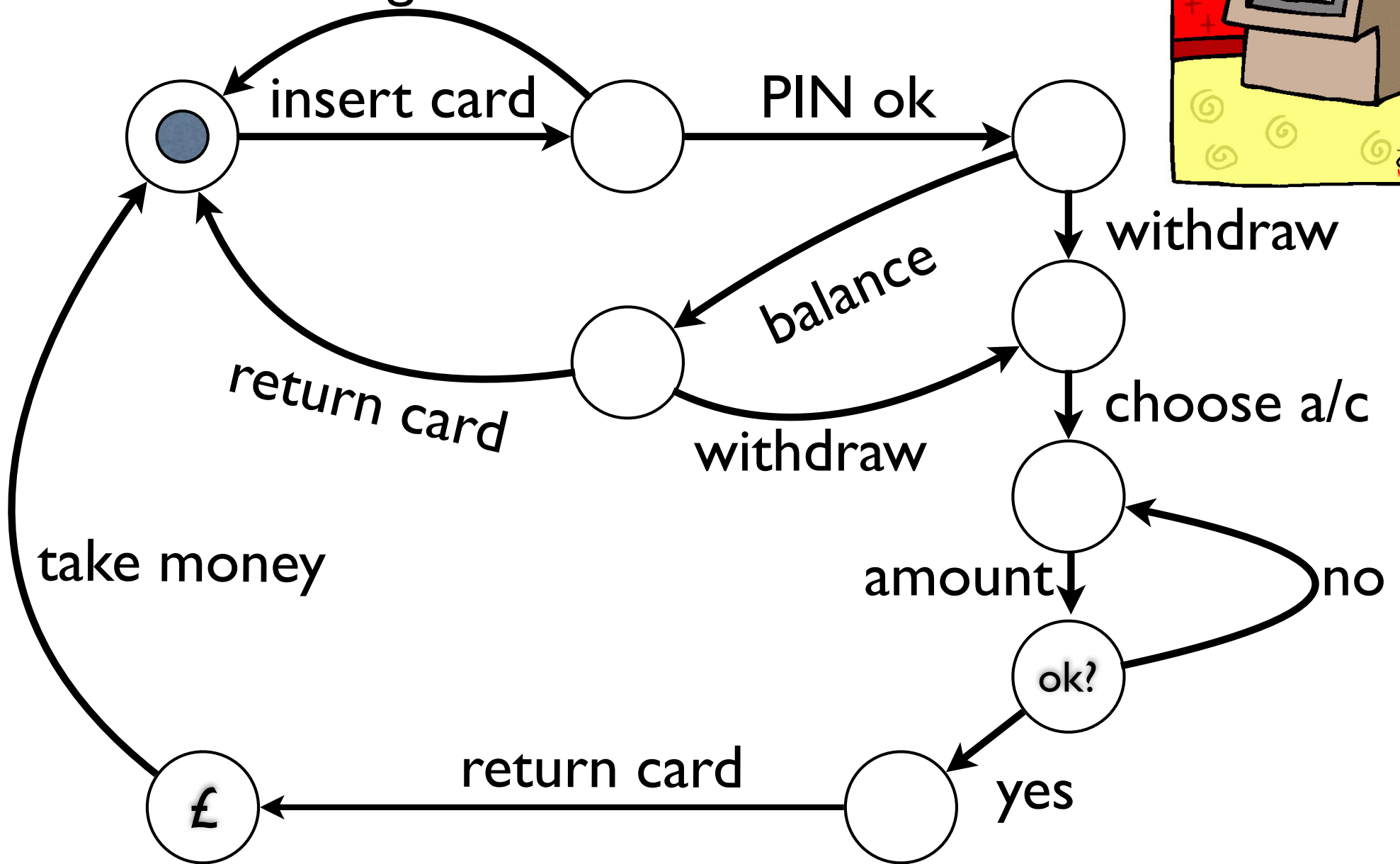
| current |   |   |   |
|---------|---|---|---|
| A       | B | C | D |
| <hr/>   |   |   |   |
| B       | C | D | A |
| next    |   |   |   |



How can we use logic to specify the transitions?

This is a **non-deterministic** system. We define a next state **relation**.

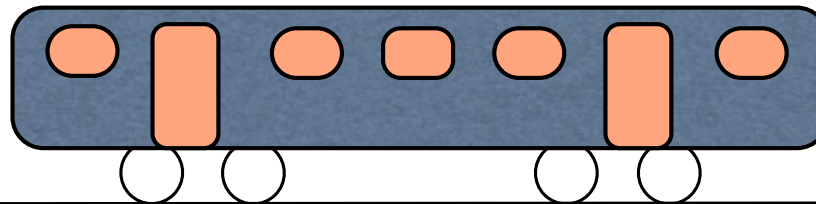
# ATM





# Counting trains

A

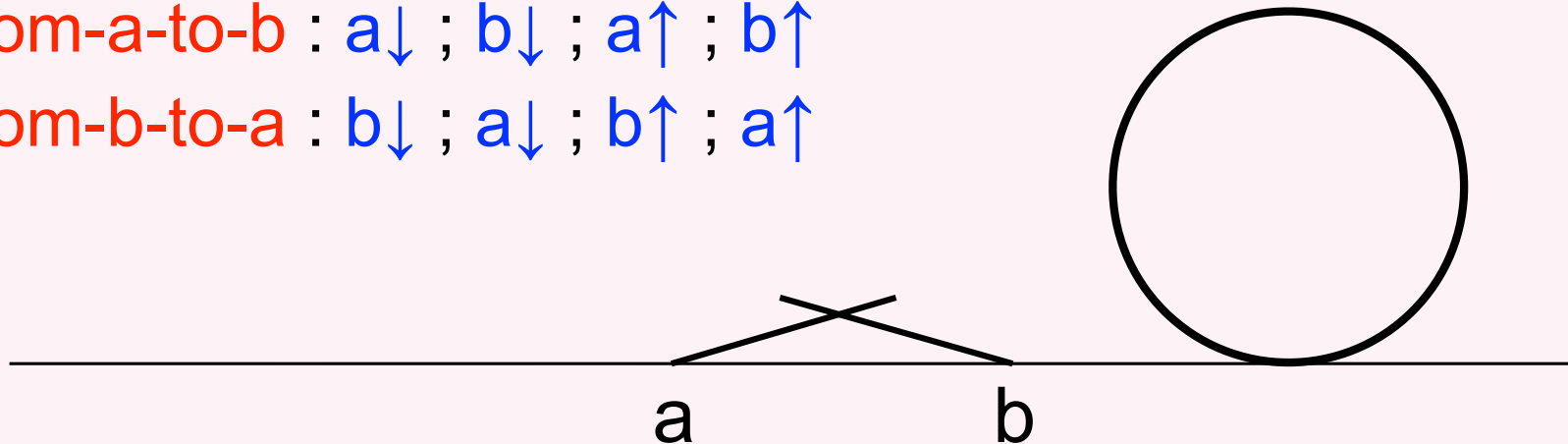


B

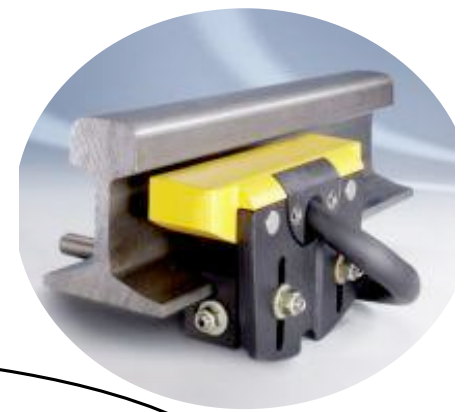
axle sensor (detects passing wheels)

from-a-to-b :  $a \downarrow$  ;  $b \downarrow$  ;  $a \uparrow$  ;  $b \uparrow$

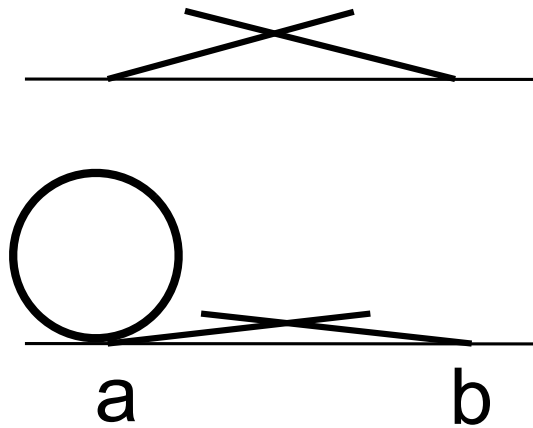
from-b-to-a :  $b \downarrow$  ;  $a \downarrow$  ;  $b \uparrow$  ;  $a \uparrow$



# Finite-state machines



axle sensor

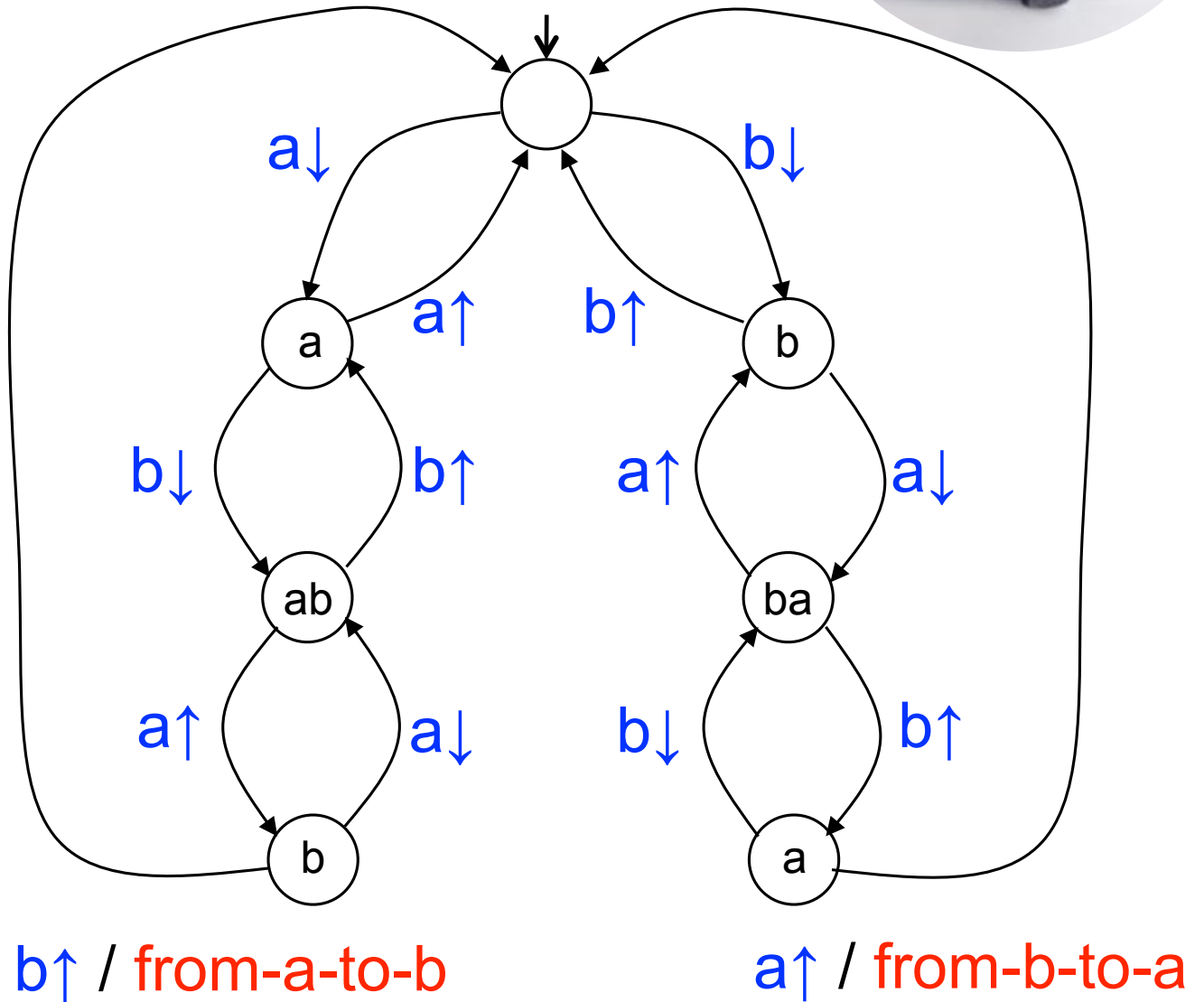


inputs :

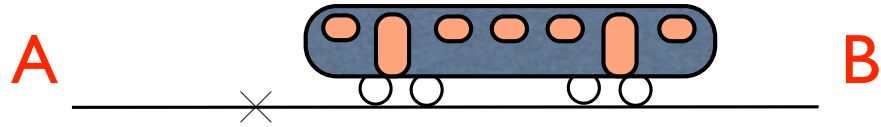
$a\uparrow$ ,  $a\downarrow$ ,  $b\uparrow$ ,  $b\downarrow$

outputs :

from-a-to-b,  
from-b-to-a



# Hierarchical FSMs



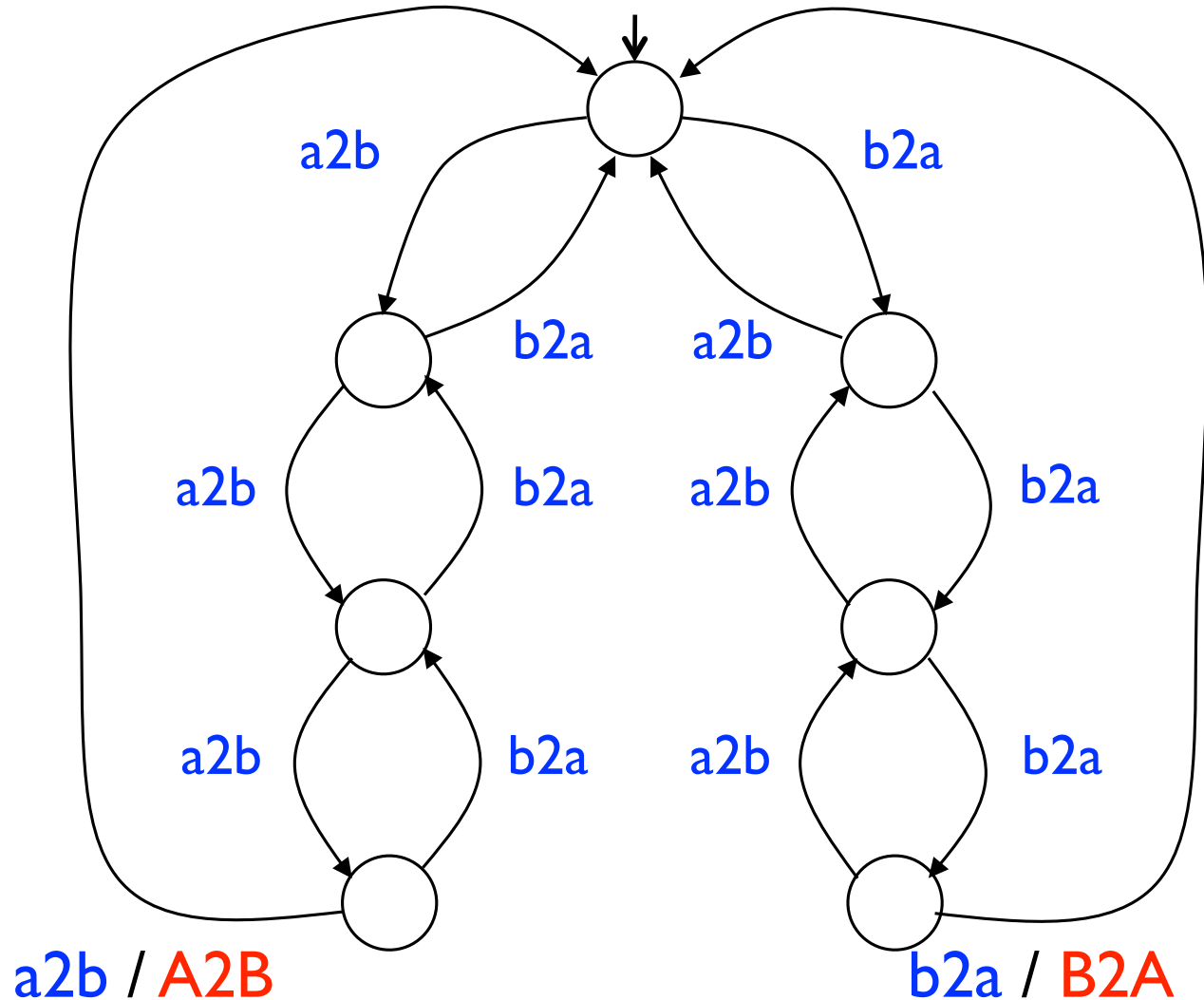
carriage counter

inputs :

a2b, b2a

outputs :

A2B, B2A



a2b = from-a-to-b

# Application Fields



## Industry

- real-time control, vending machines, cash dispensers, etc.

## Electronic circuits

- data path / control path
- memory / cache handling
- protocols, USB, etc.



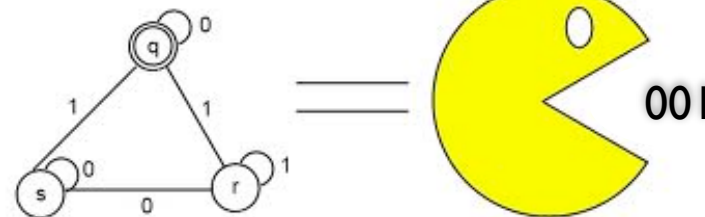
## Communication protocols

- initiation and maintenance of communication links
- error detection and handling, packet retransmission



## Language analysis

- natural languages
- programming languages
- search engines





# A Decimal Number

