

SBML Level 3 extension Qualitative Models

Why this extension?

The number of groups using qualitative modeling is increasing.

Therefore the number of software is increasing.

There is a need for an exchange format.

Previous works tried unsuccessfully to use SBML.

Why this extension?

The number of groups using qualitative modeling is increasing.

Therefore the number of software is increasing.

There is a need for an exchange format.

Previous works tried unsuccessfully to use SBML.

Which formalism should we cover?

Why this extension?

The number of groups using qualitative modeling is increasing.

Therefore the number of software is increasing.

There is a need for an exchange format.

Previous works tried unsuccessfully to use SBML.

Which formalism should we cover?

We should cover:

- Formalisms not already covered by SBML itself.
- Formalisms used by several bioinformatics studies.

Why this extension?

The number of groups using qualitative modeling is increasing.

Therefore the number of software is increasing.

There is a need for an exchange format.

Previous works tried unsuccessfully to use SBML.

Which formalism should we cover?

We should cover:

- Formalisms not already covered by SBML itself.
- Formalisms used by several bioinformatics studies.

For the moment, we should cover:

- Logical models
- Petri nets

Videoconference

The videoconference has taken place on the 12 and 13 of August 2008.

Participants	Project
Denis Thieffry, Claudine Chaouiya, Aurelien Naldi	GINsim (logical model)
Ioannis Xenarios, Alessandro Di Cara	SQUAD (logical model)
Tomas Helikar	ChemChains (logical model)
Claudine Chaouiya	Petri nets
Mathias John, Dagmar Koehn	PiML (Pi-calculus)
Nicolas Le Novère, Nicolas Rodriguez, Sarah Keating, Duncan Berenguier	SBML

Qualitative model representation

Qualitative model representation

A

B

C

D

E

Qualitative model representation

A

B

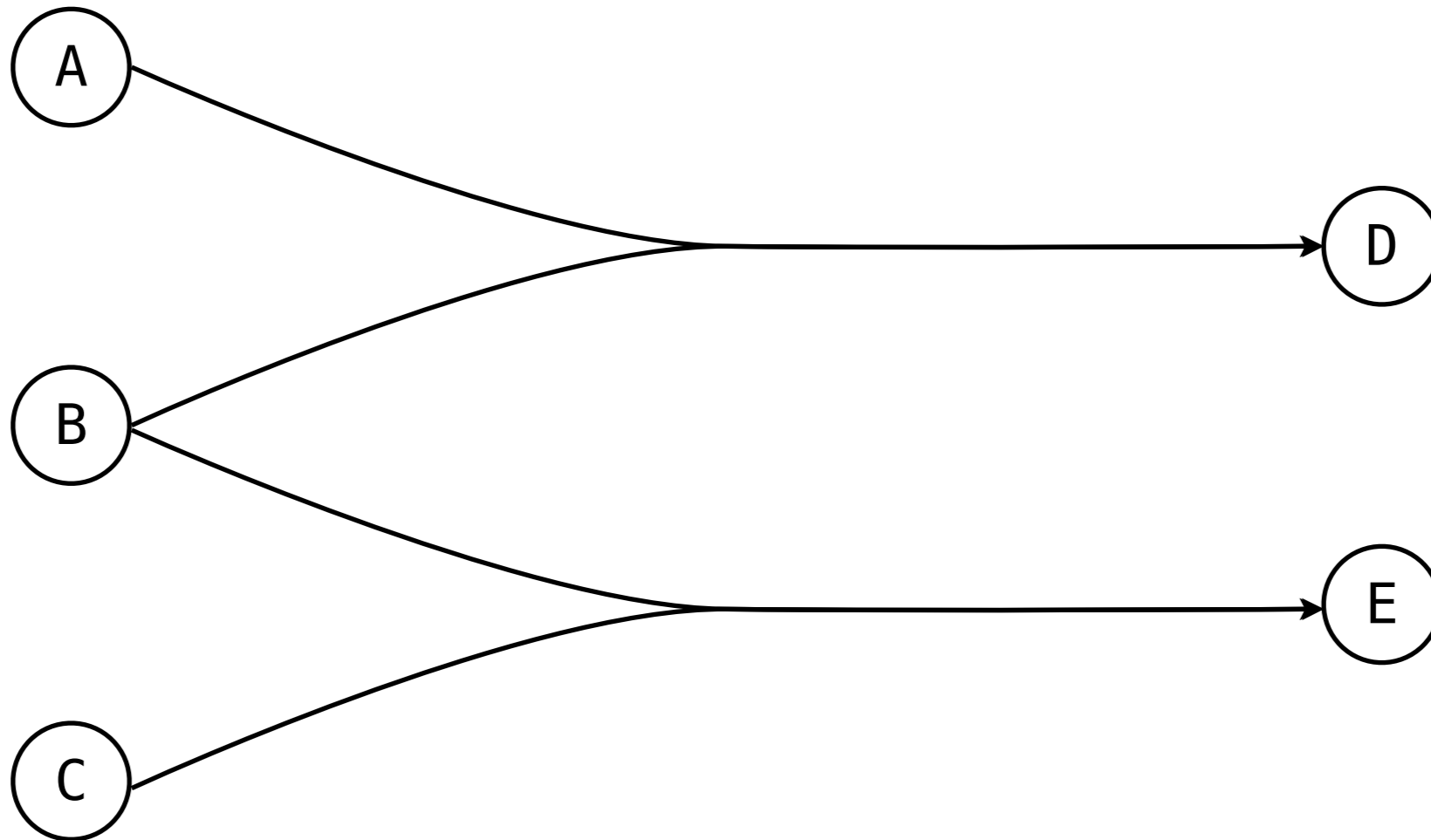
C

D

E

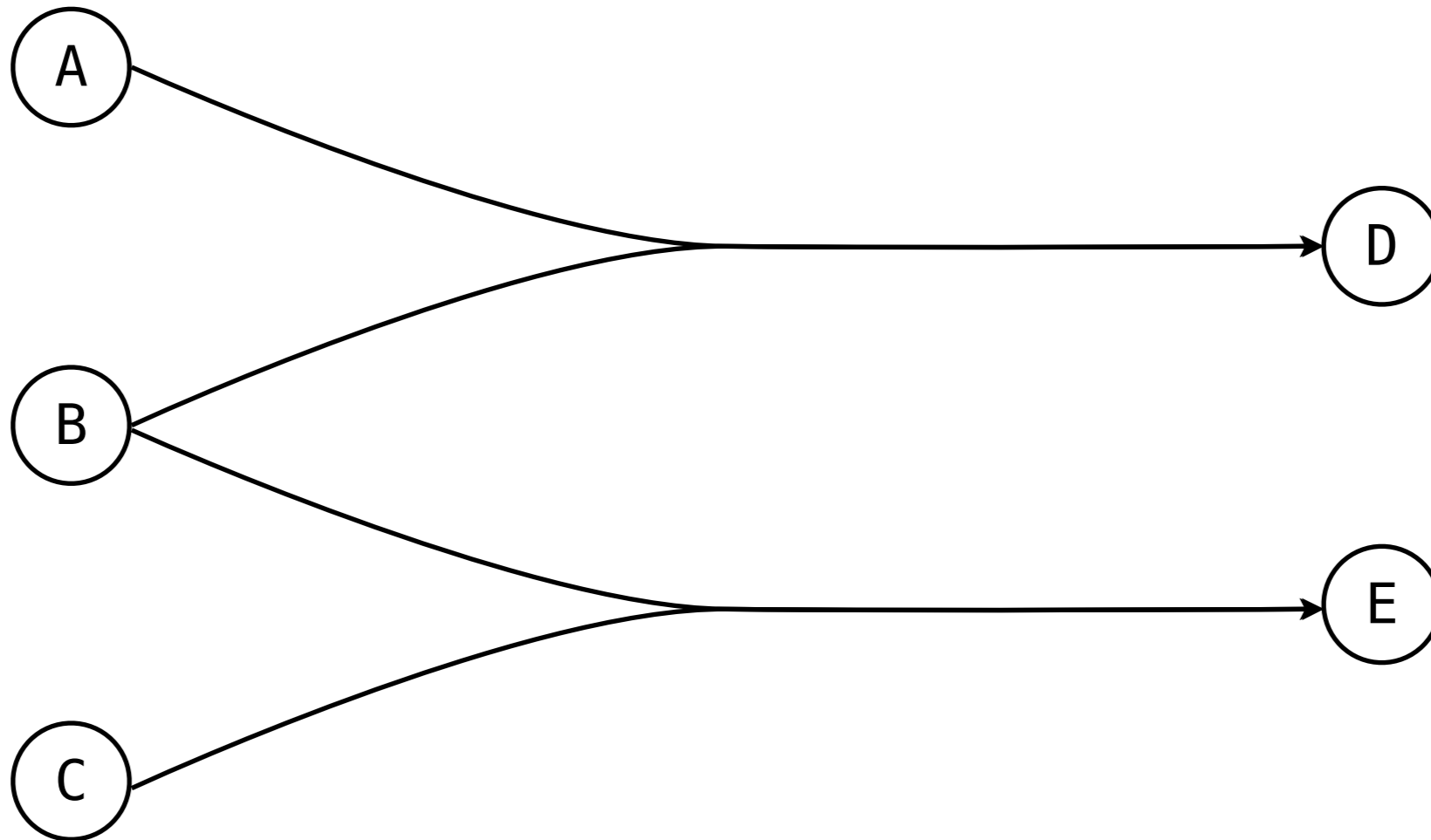
A Species

Qualitative model representation



A Species

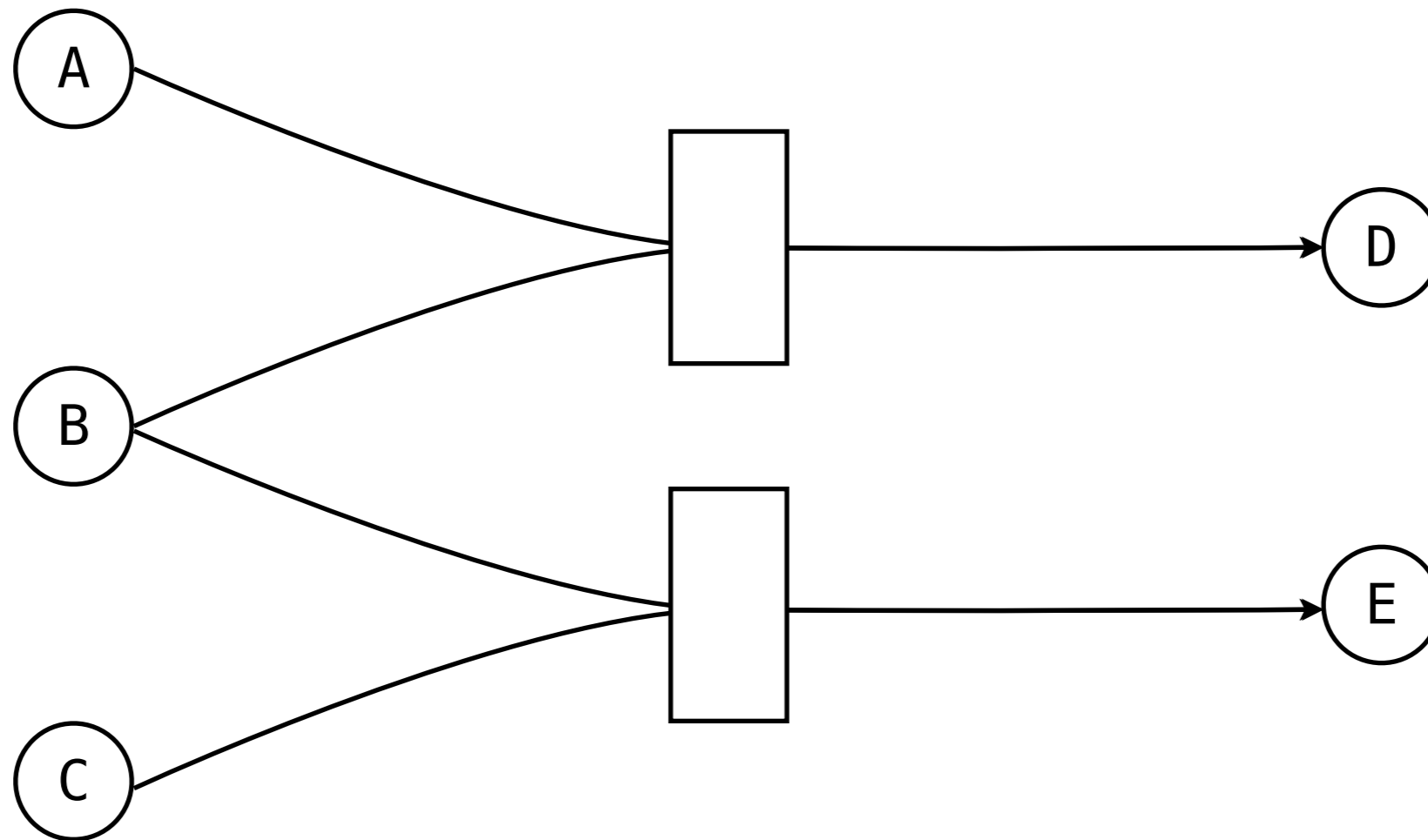
Qualitative model representation



(A) Species

Arc

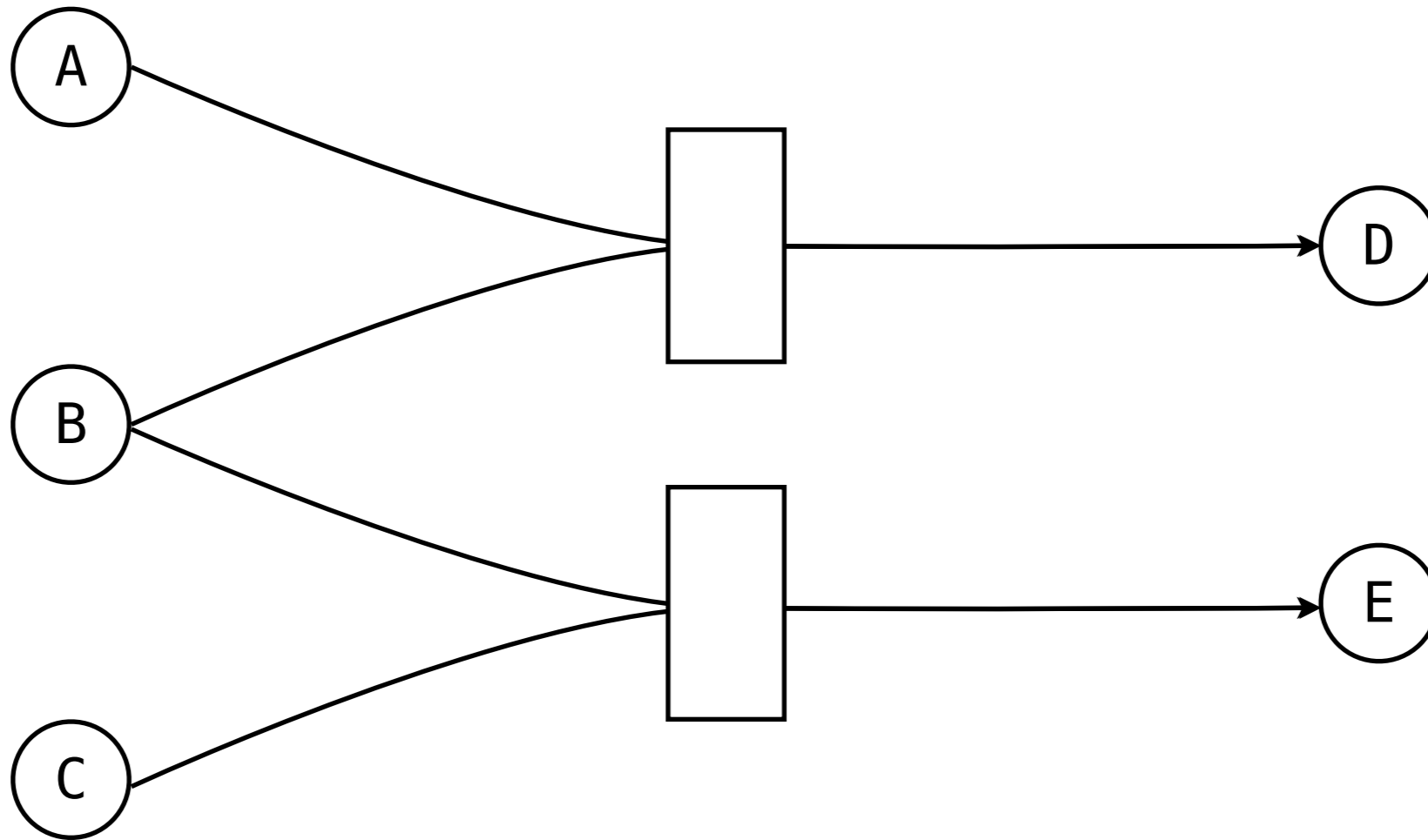
Qualitative model representation




(A) Species

Arc

Qualitative model representation

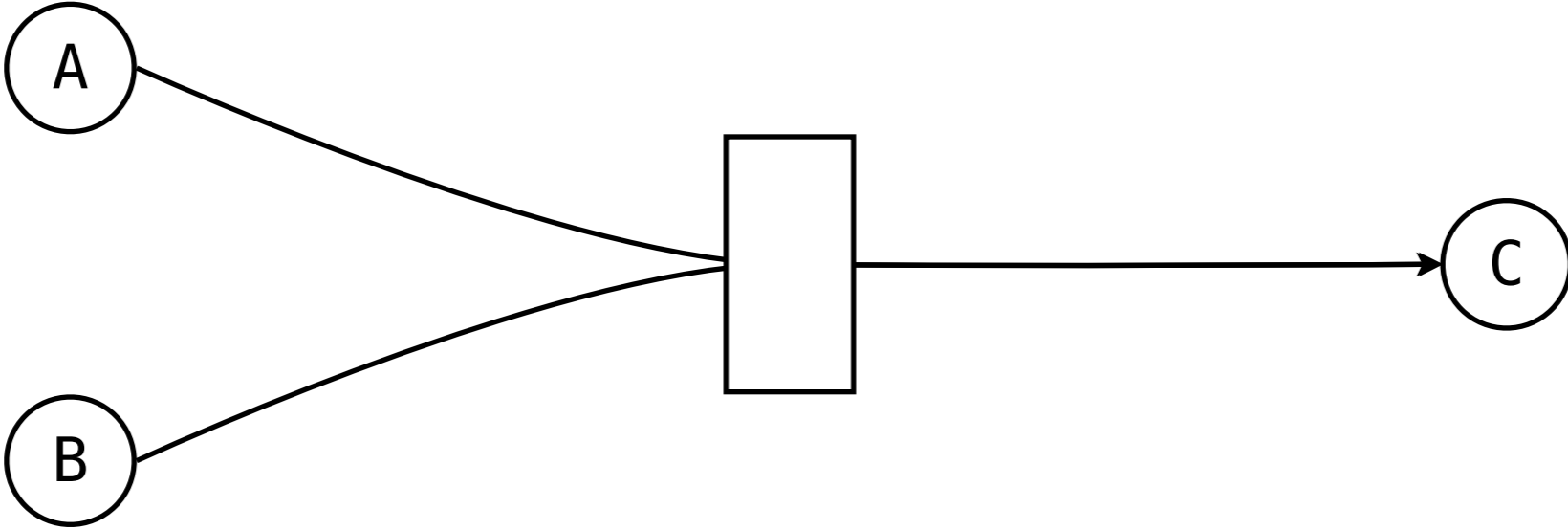


 Species

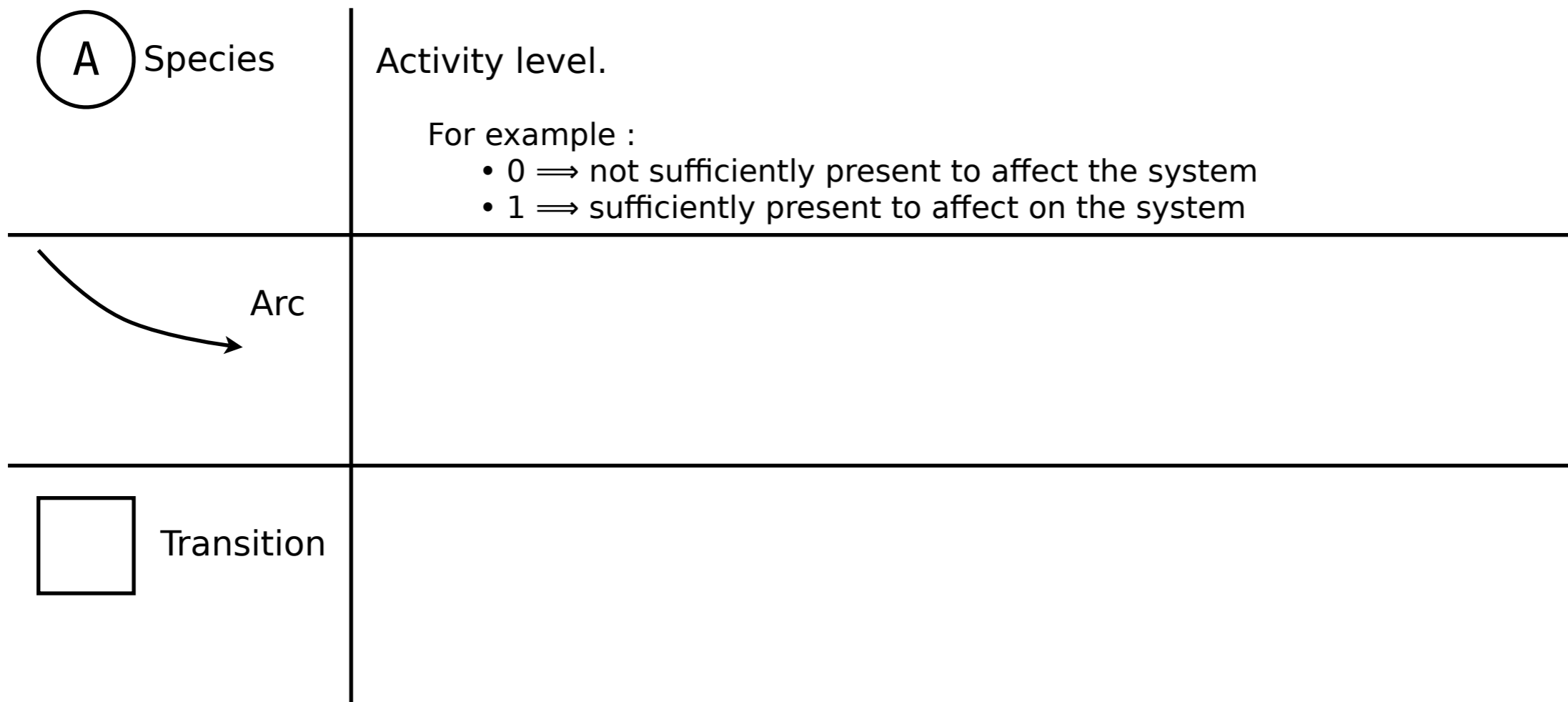
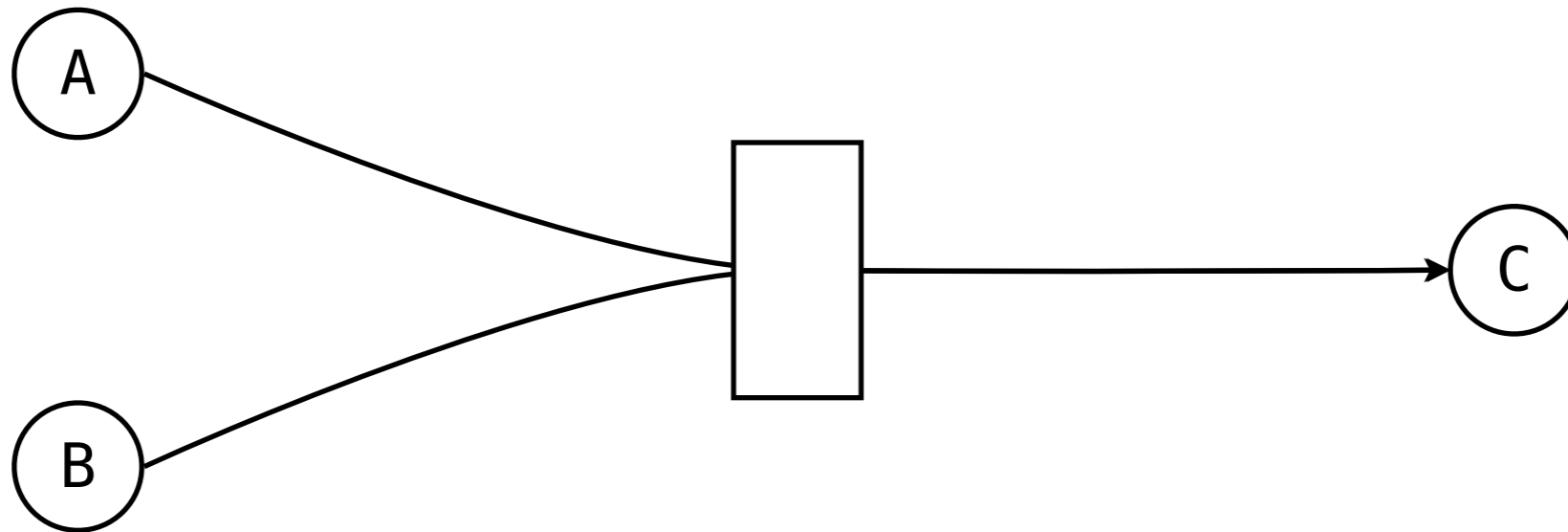
 Arc

 Transition

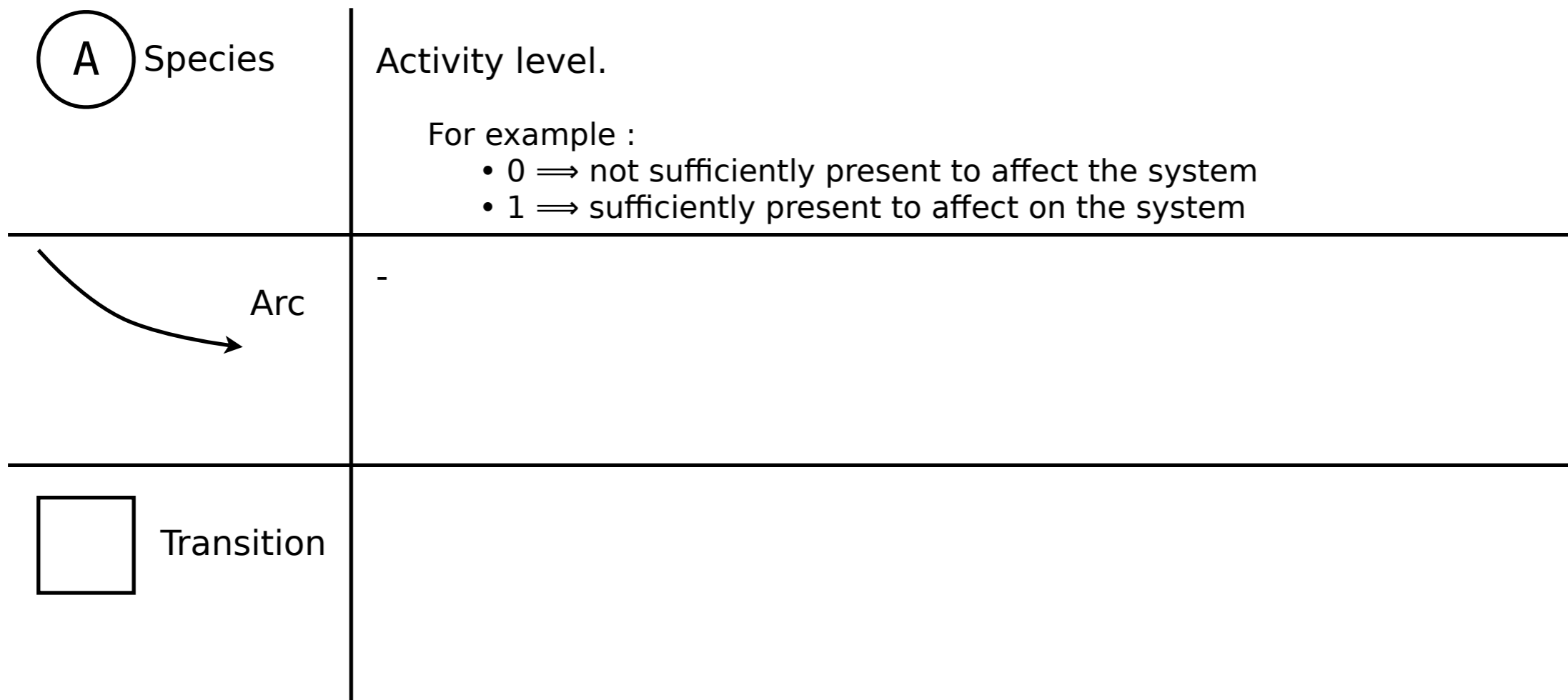
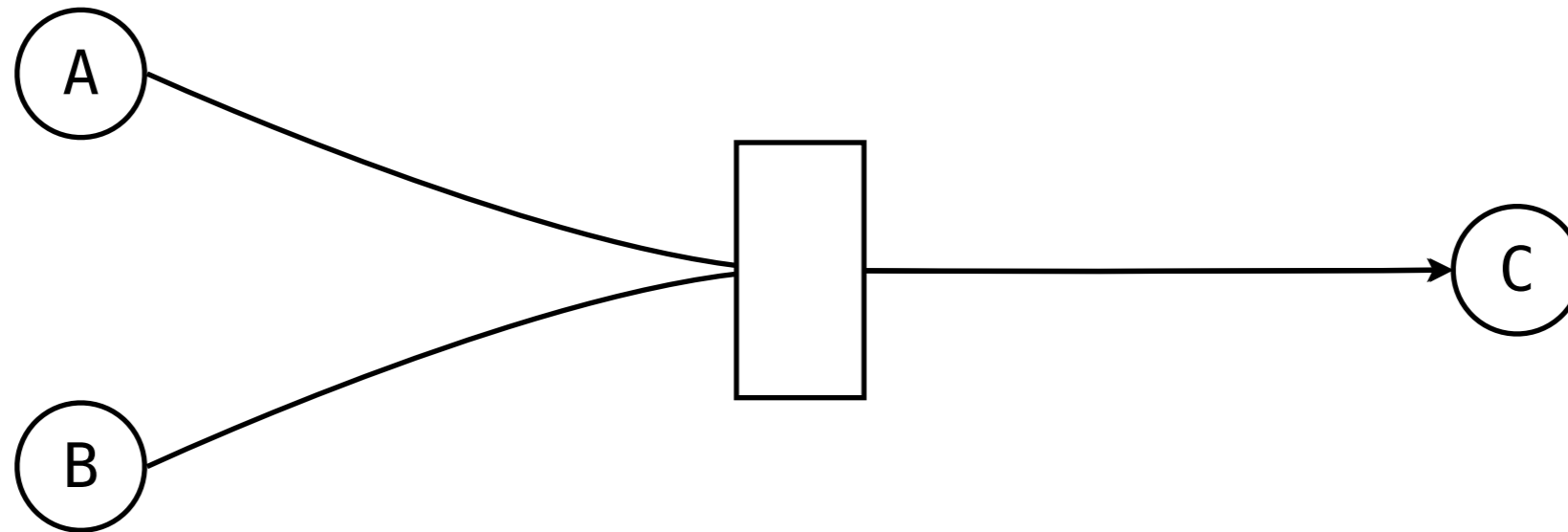
Qualitative model representation — Logical Model



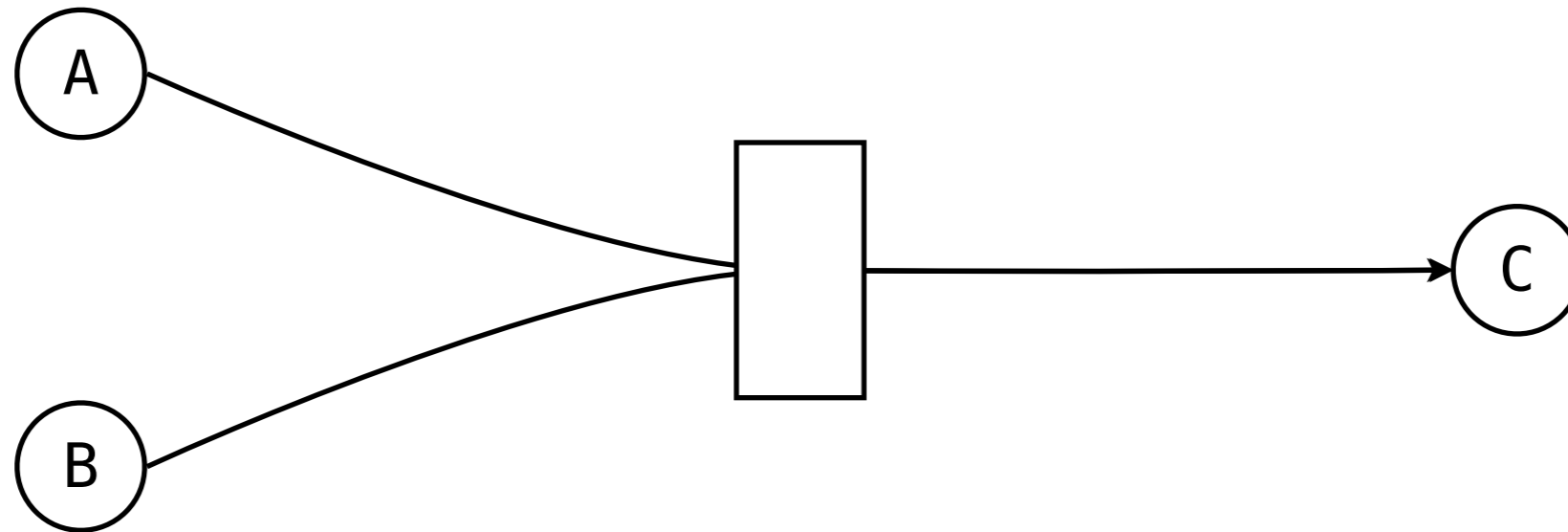
Qualitative model representation — Logical Model

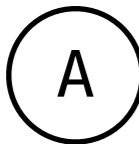

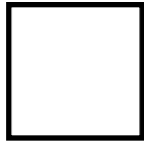


Qualitative model representation — Logical Model

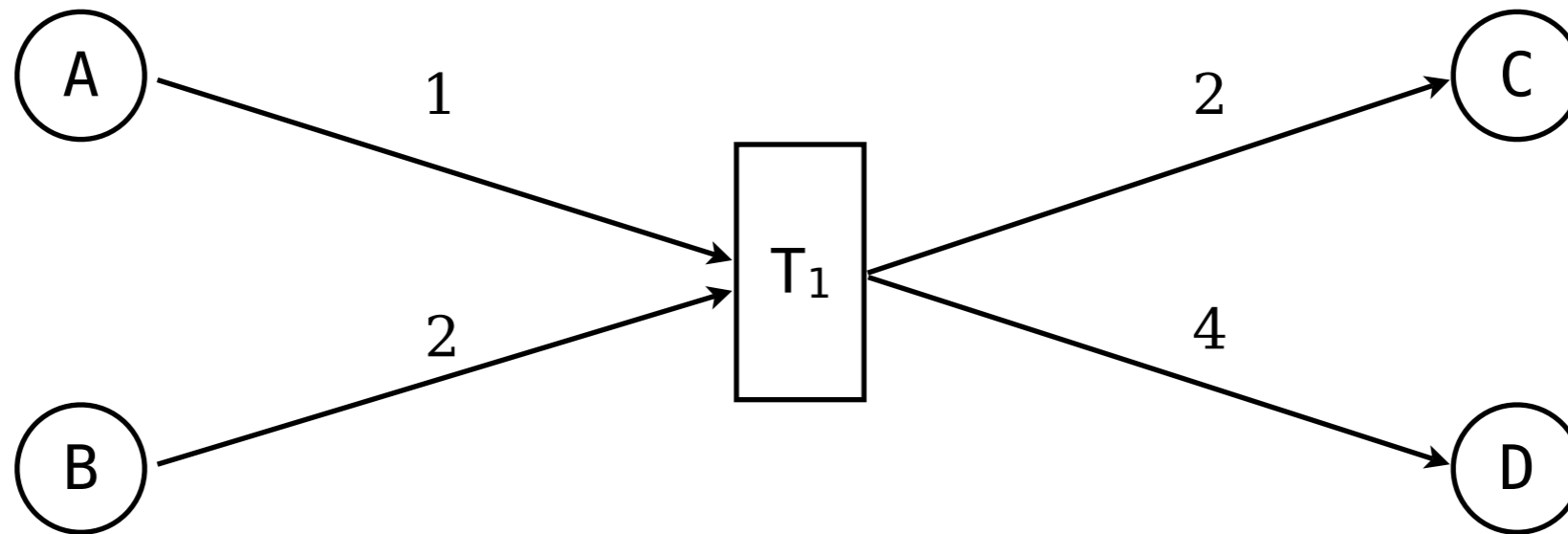


Qualitative model representation — Logical Model

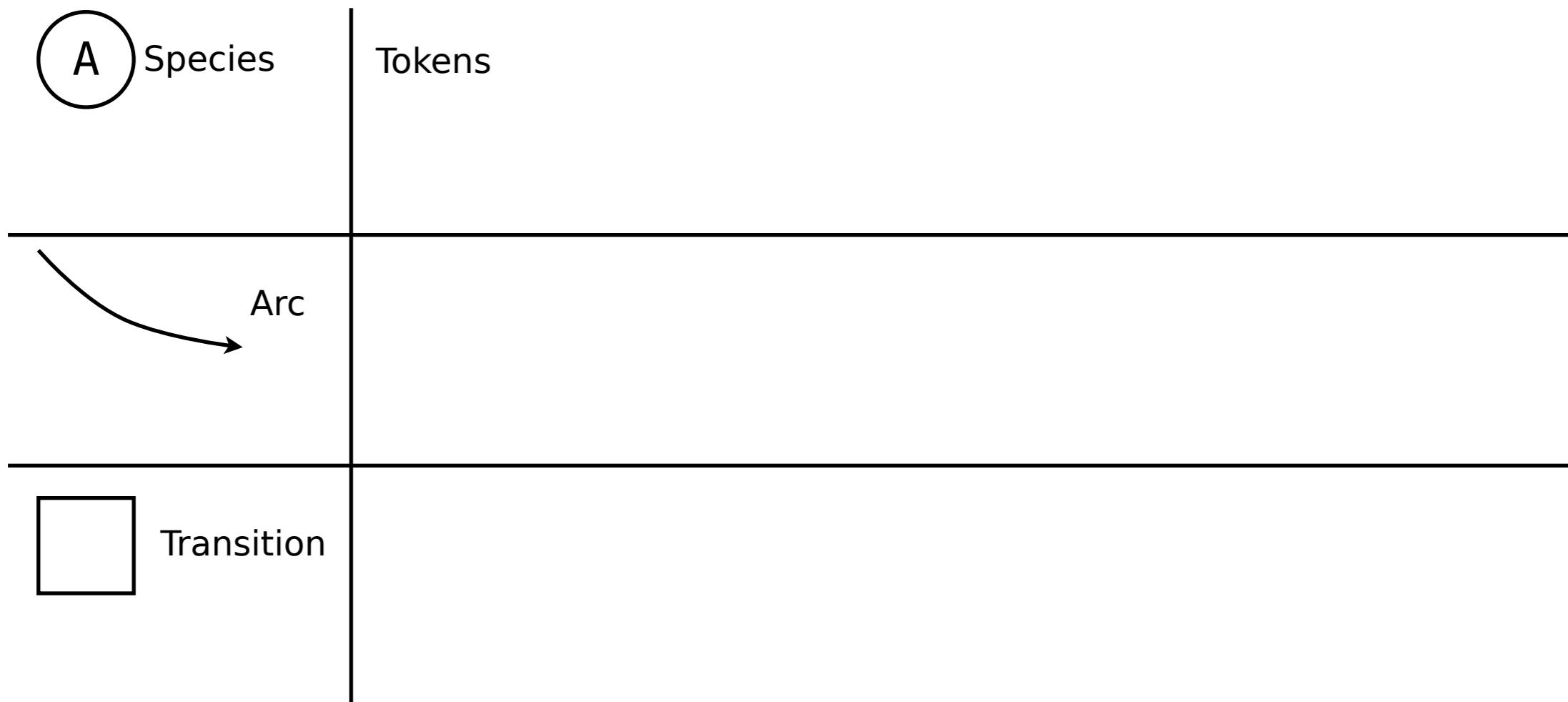
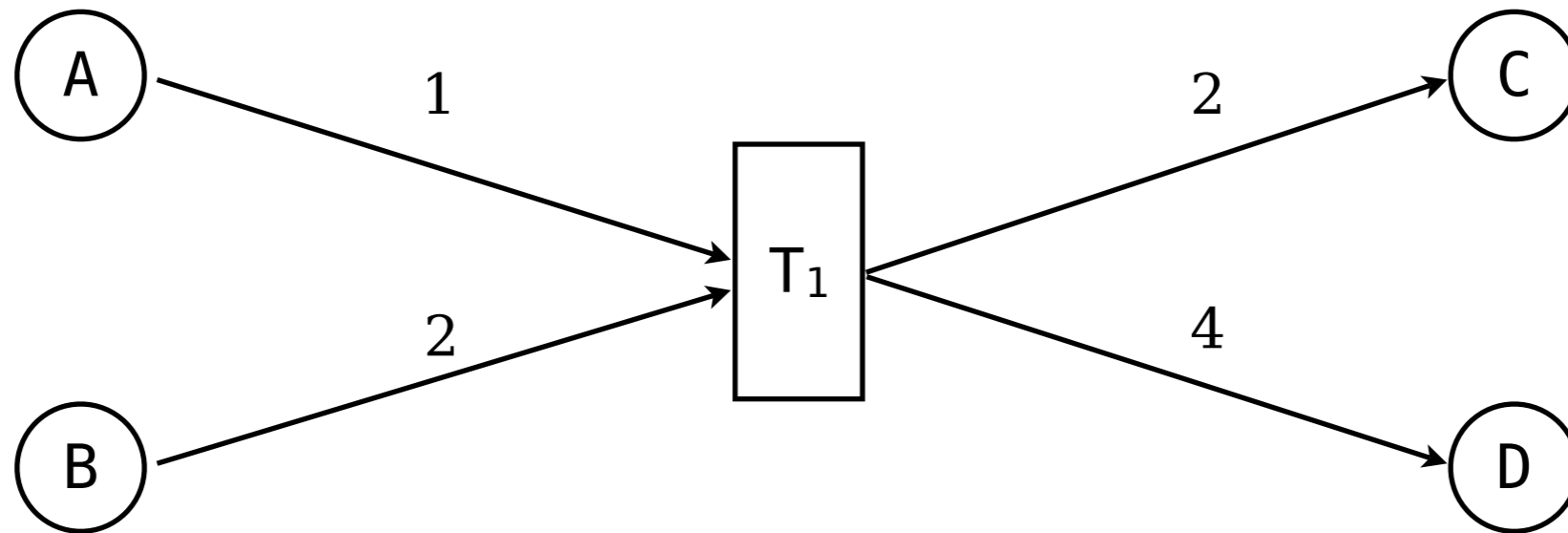


 Species	Activity level. For example : <ul style="list-style-type: none"> • 0 \Rightarrow not sufficiently present to affect the system • 1 \Rightarrow sufficiently present to affect on the system
 Arc	-
 Transition	Logical formula For example : $C := \begin{cases} 2 & \text{if } A = 2 \wedge B = 1 \\ 1 & \text{if } A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0 \\ 0 & \text{otherwise} \end{cases}$

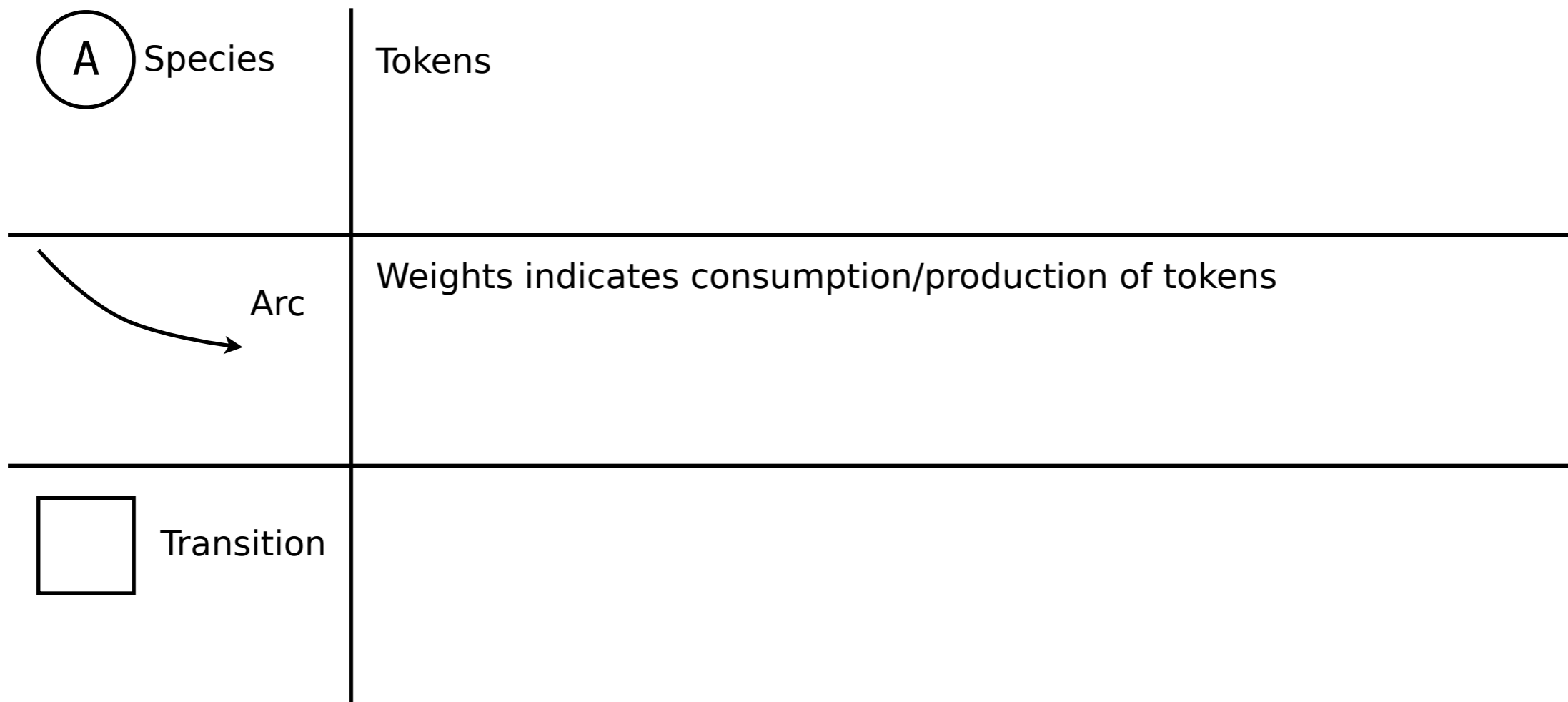
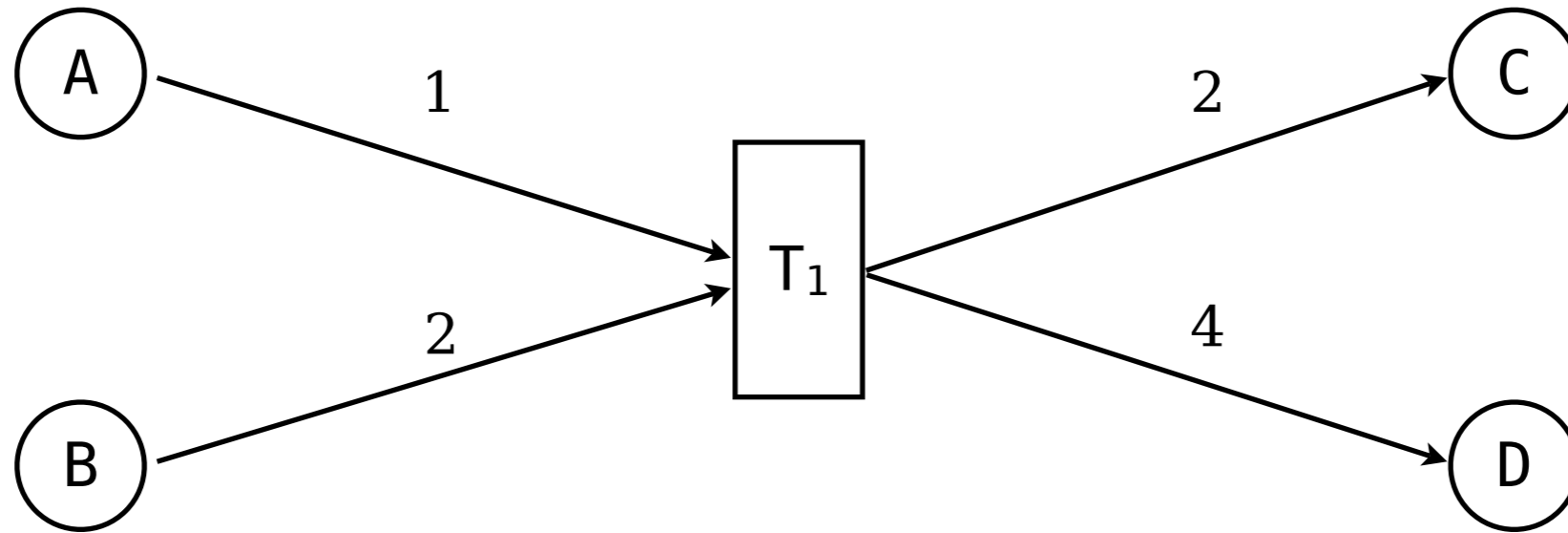
Qualitative model representation — Petri nets



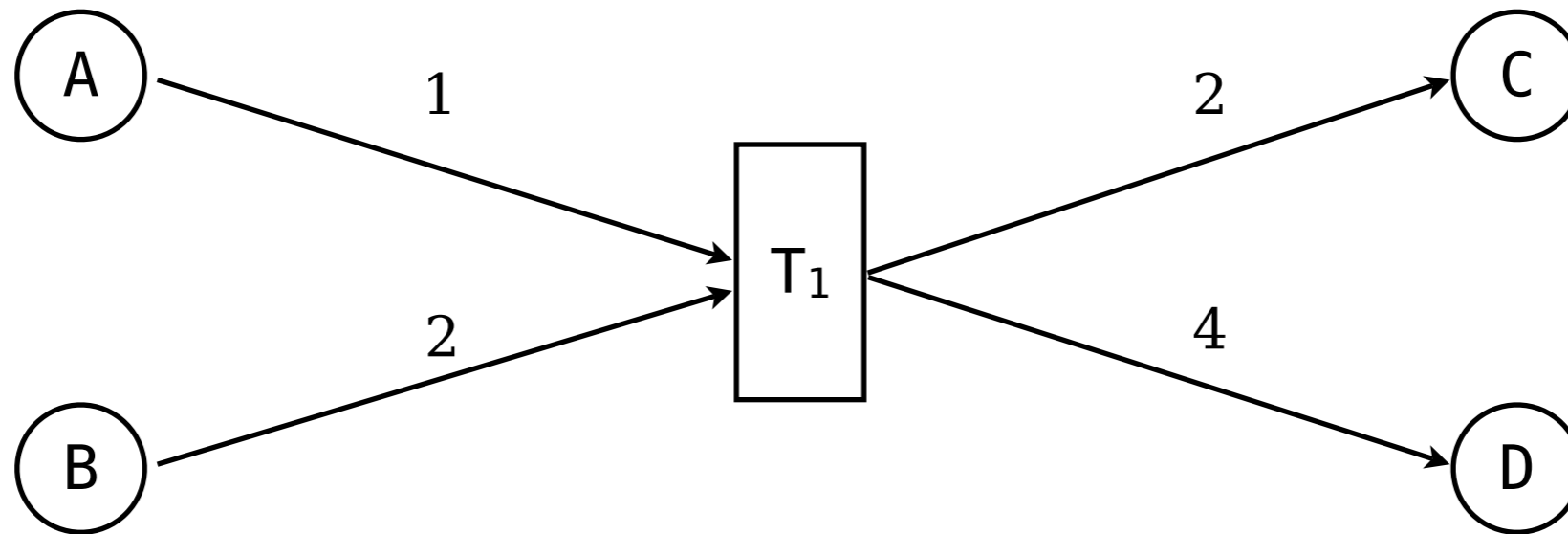
Qualitative model representation — Petri nets

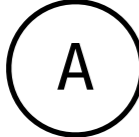
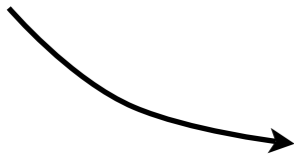



Qualitative model representation — Petri nets

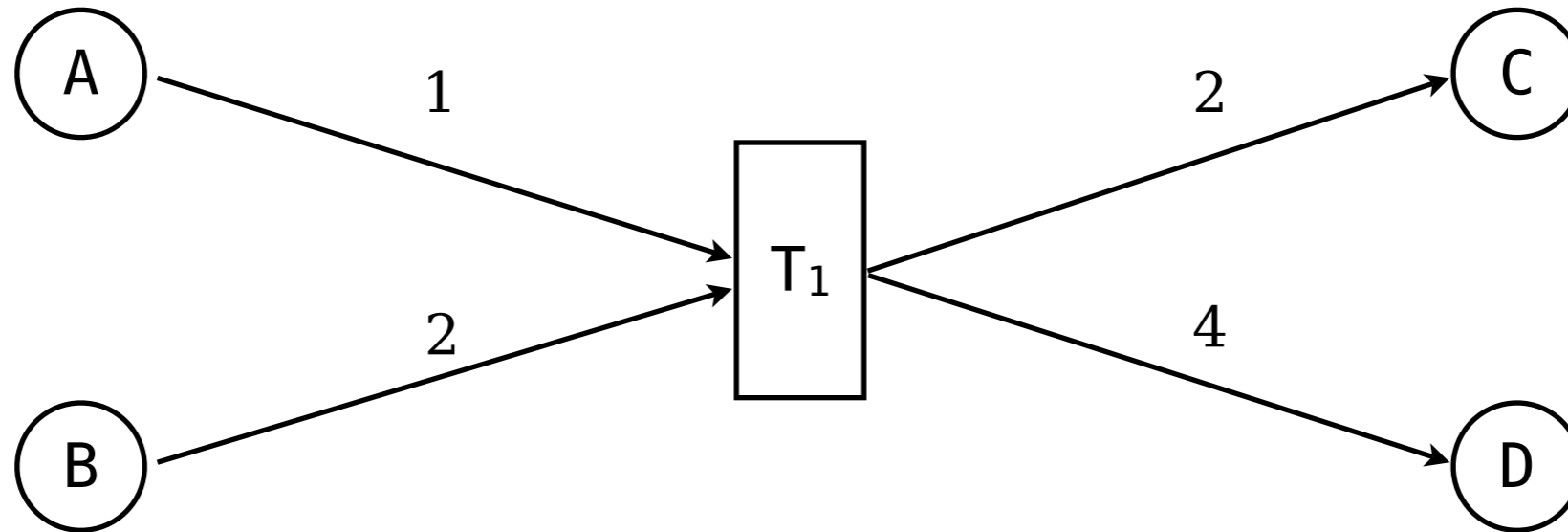


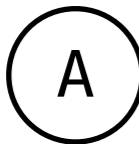
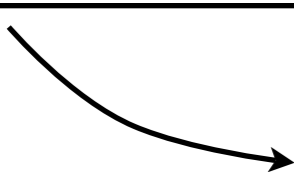
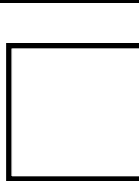
Qualitative model representation — Petri nets



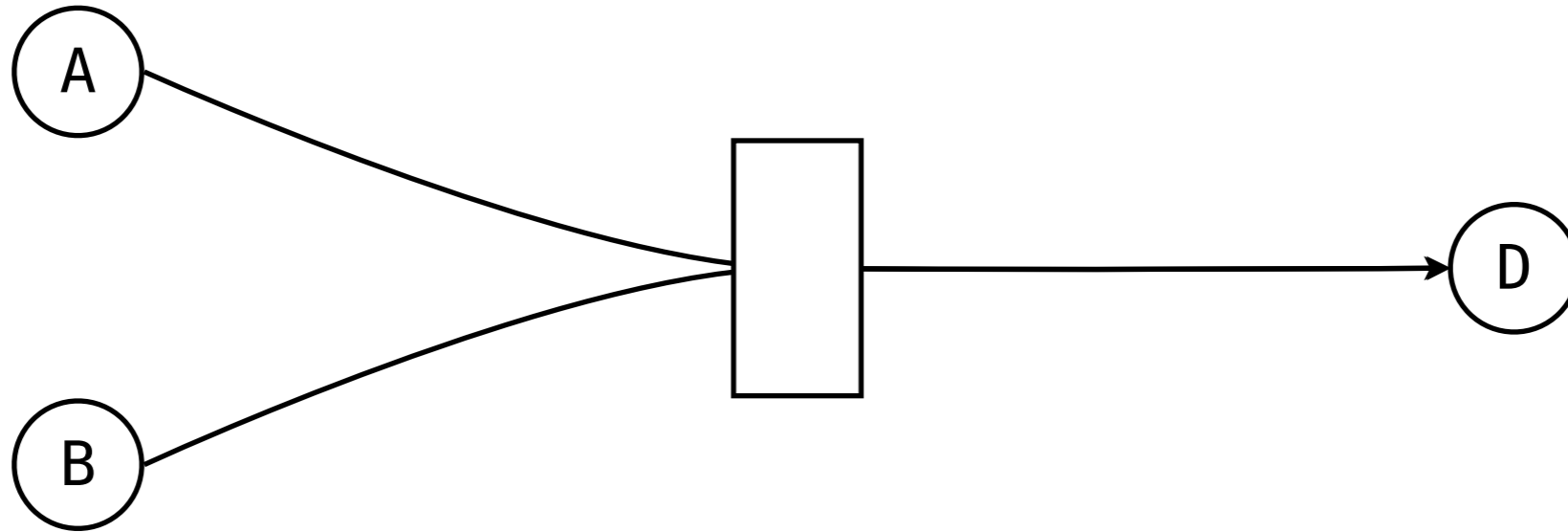
 Species	Tokens
 Arc	Weights indicates consumption/production of tokens
 Transition	Logical formulae are deduced from the graph topology and arcs weights For example :

Qualitative model representation — Petri nets

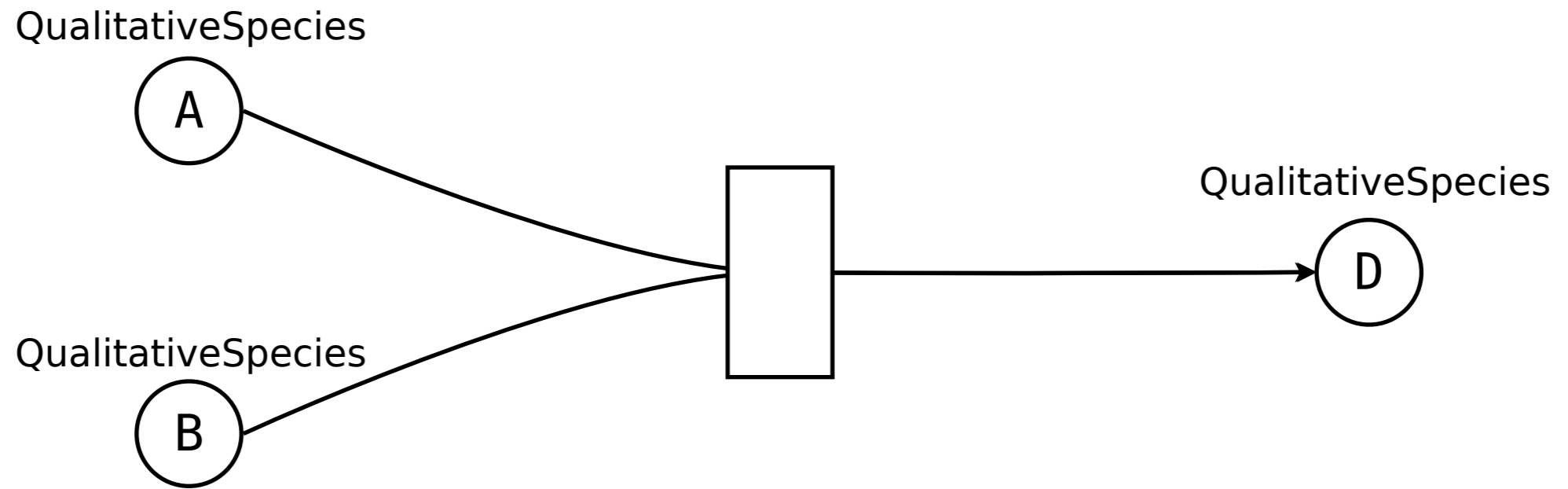


 Species	Tokens
 Arc	Weights indicates consumption/production of tokens
 Transition	Logical formulae are deduced from the graph topology and arcs weights For example : $\text{if } A \geq 1 \wedge B \leq 2 \text{ then}$ $A := A - 1$ $B := B - 2$ $C := C + 2$ $D := D + 4$

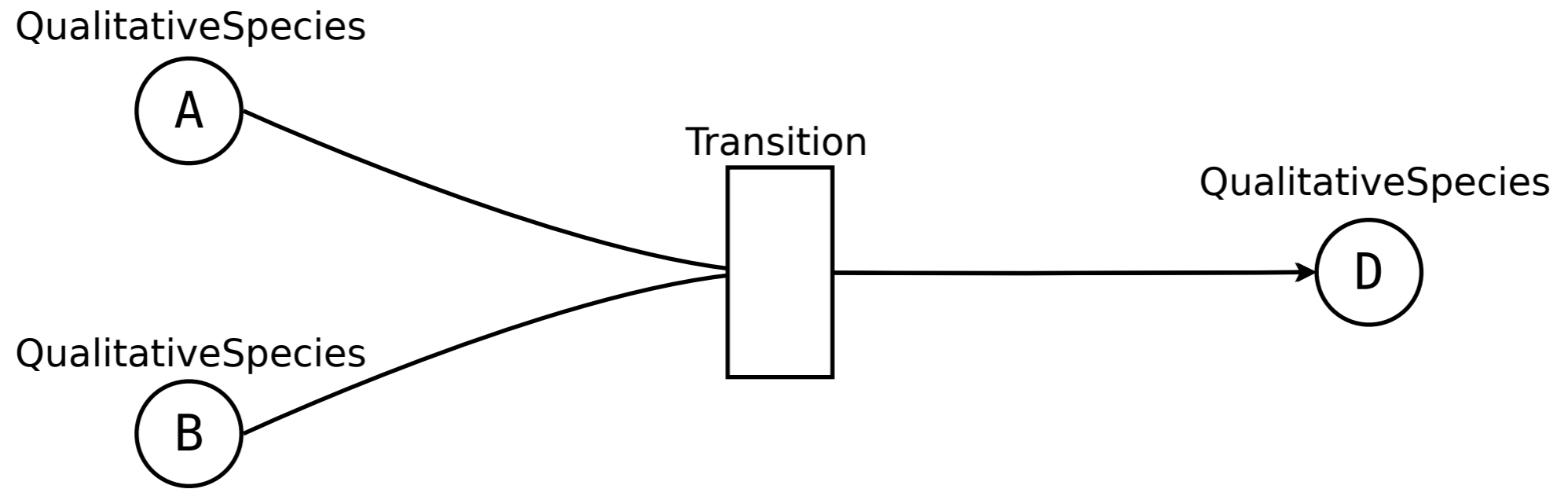
Introduction to our proposal



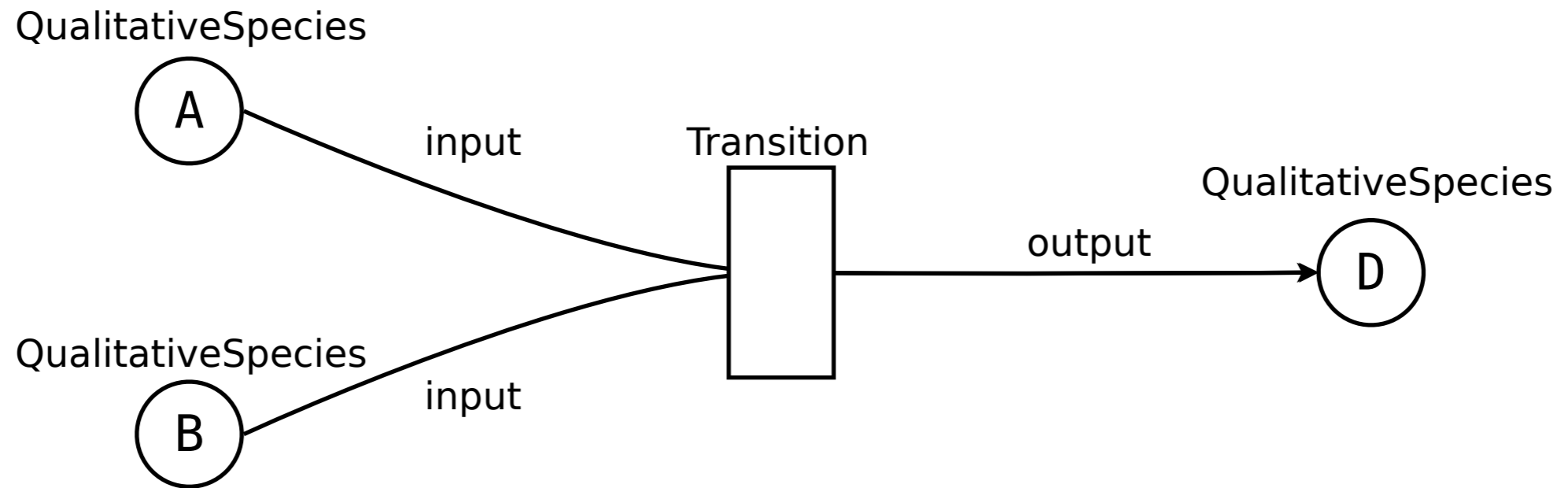
Introduction to our proposal



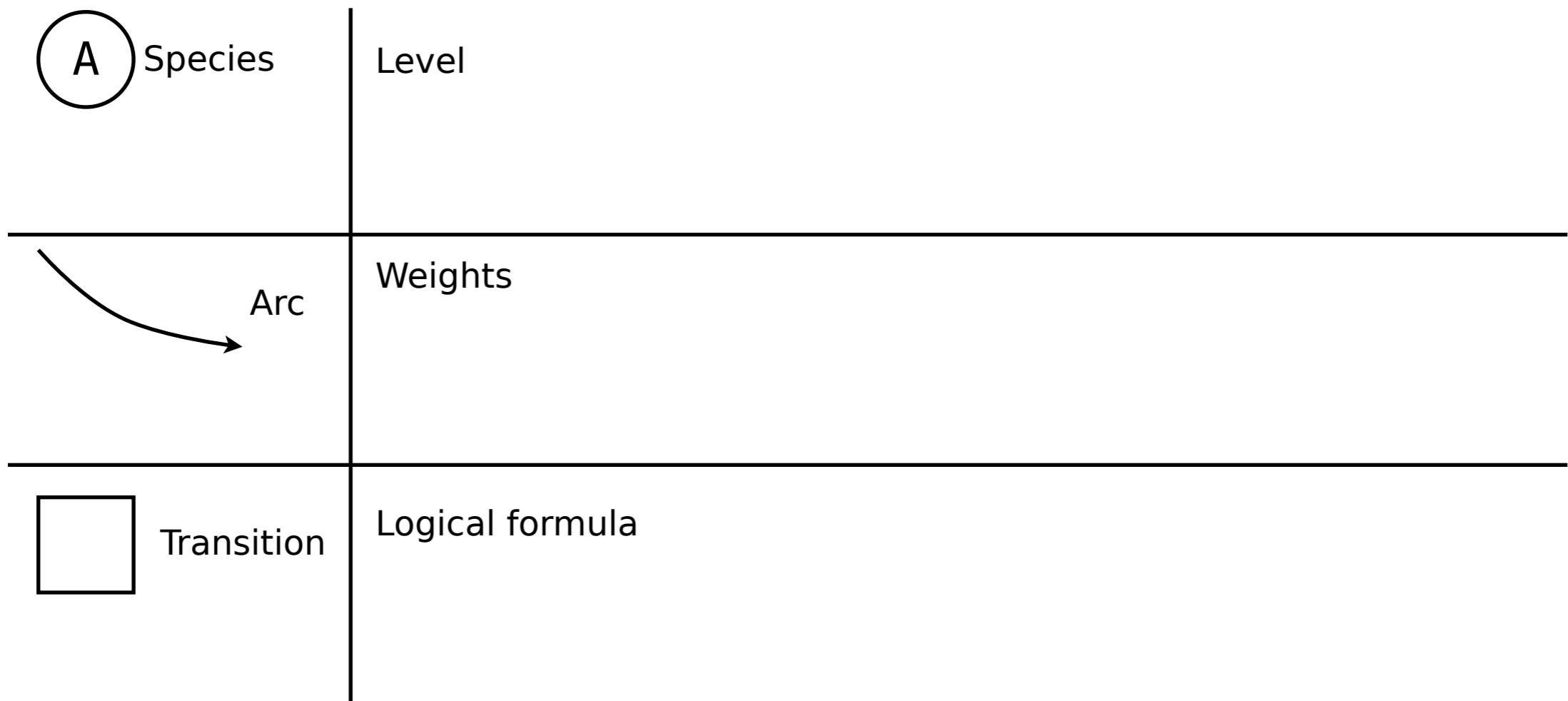
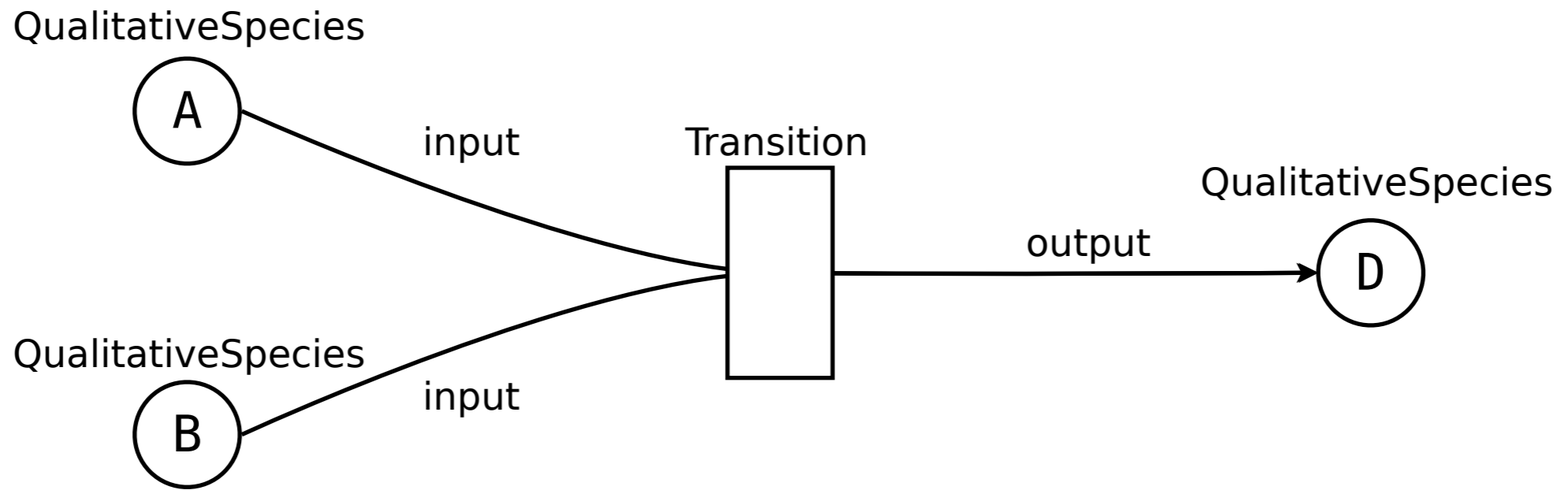
Introduction to our proposal



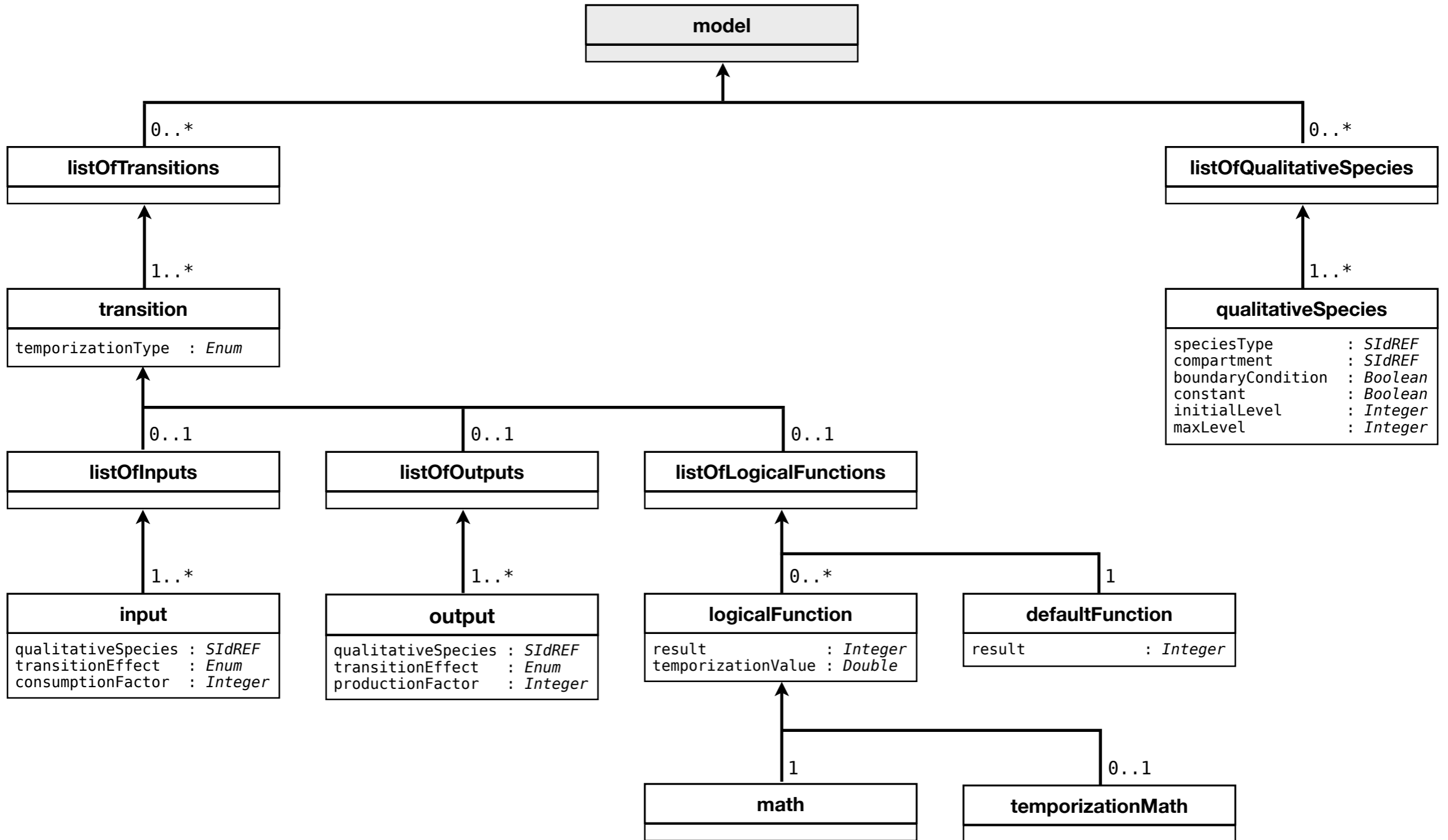
Introduction to our proposal



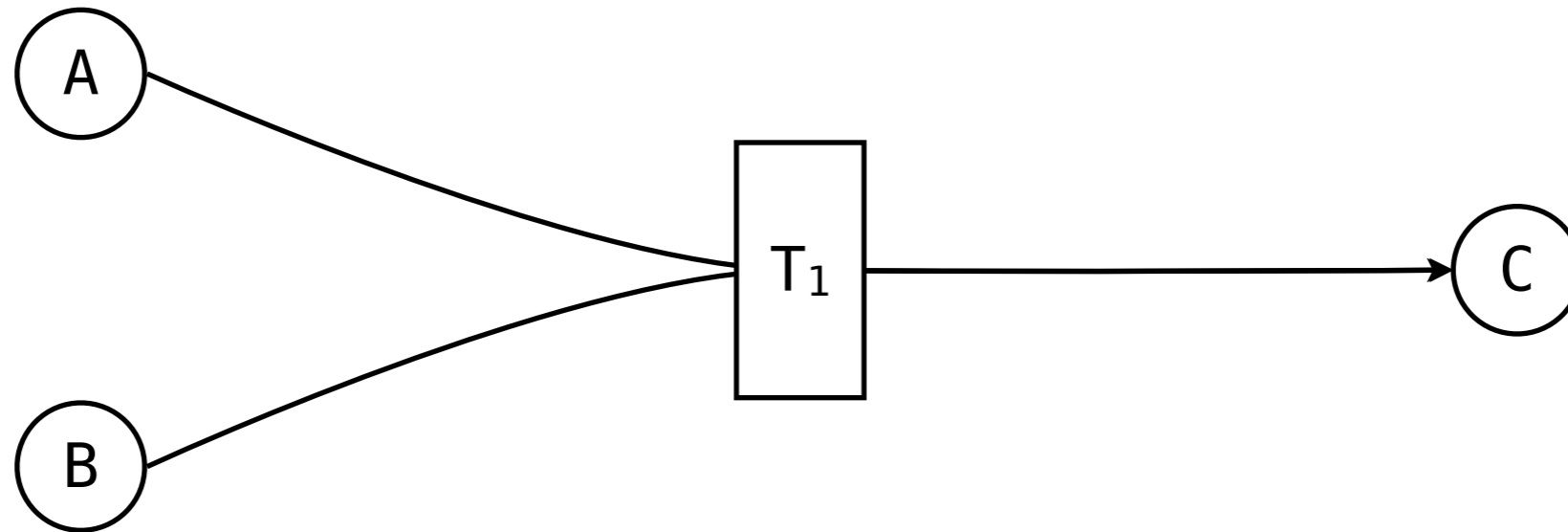
Introduction to our proposal



Diagram



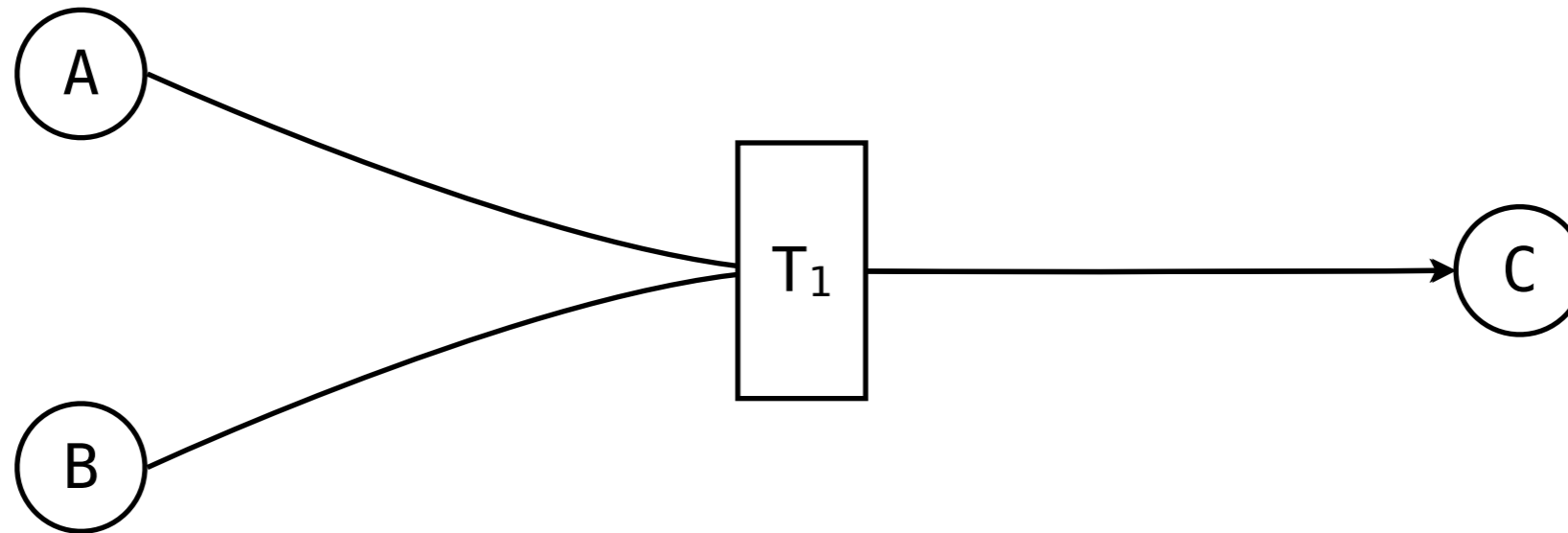
Dynamics — Logical model



C's level is defined by the following formula :

$$C := \begin{cases} 2 & \text{if } A = 2 \wedge B = 1 \\ 1 & \text{if } A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0 \\ 0 & \text{otherwise} \end{cases}$$

Dynamics — Logical model



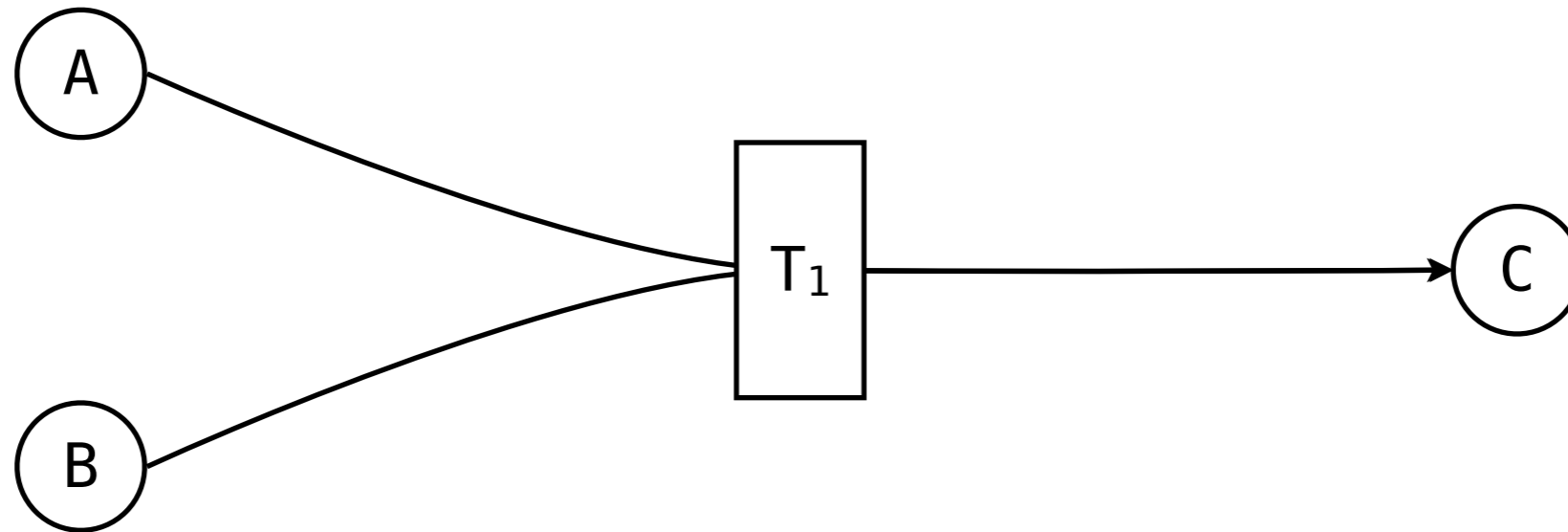
C's level is defined by the following formula :

$$C := \begin{cases} 2 & \text{if } A = 2 \wedge B = 1 \\ 1 & \text{if } A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0 \\ 0 & \text{otherwise} \end{cases}$$

\Rightarrow

```
<listOfLogicalFunctions>
  <logicalFunction result="2">
    <math>
      A = 2 \wedge B = 1
    </math>
  </logicalFunction>
  <logicalFunction result="1">
    <math>
      A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0
    </math>
  </logicalFunction>
  <defaultFunction result="0" />
</listOfLogicalFunctions>
```

Dynamics — Logical model



C's level is defined by the following formula :

$$C := \begin{cases} 2 & \text{if } A = 2 \wedge B = 1 \\ 1 & \text{if } A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0 \\ 0 & \text{otherwise} \end{cases}$$

⇒

```
<listOfLogicalFunctions>
```

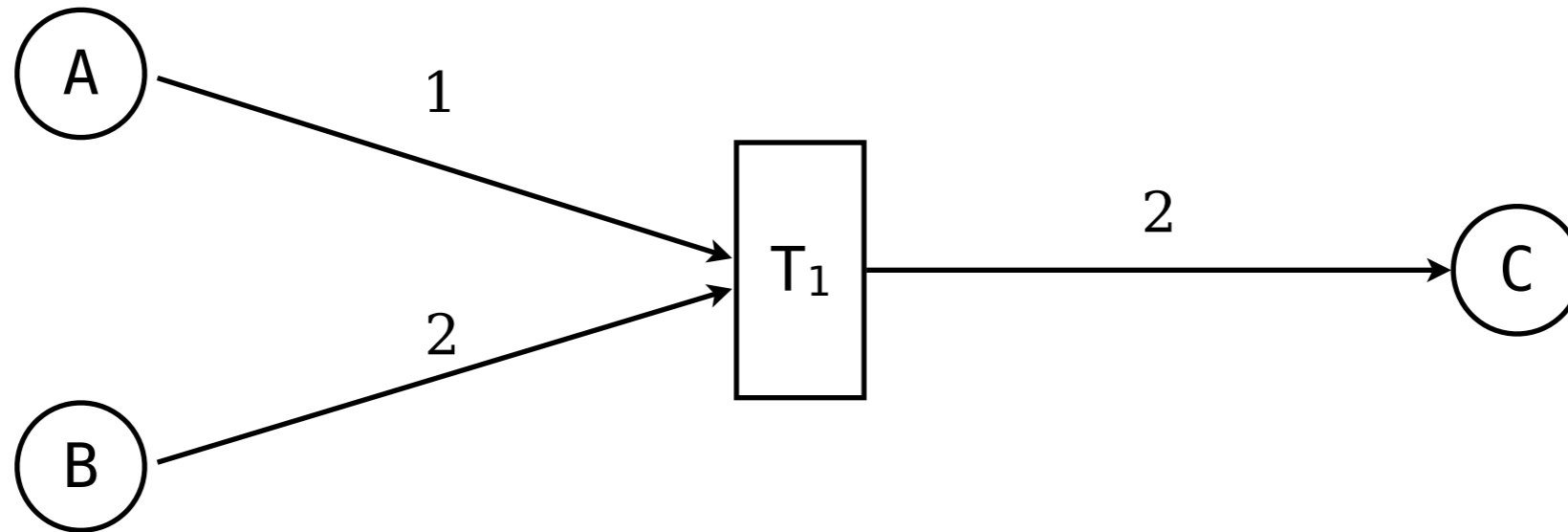
```
<logicalFunction result="2">
  <math>
    A = 2 \wedge B = 1
  </math>
</logicalFunction>
```

```
<logicalFunction result="1">
  <math>
    A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0
  </math>
</logicalFunction>
```

```
<defaultFunction result="0" />
```

```
</listOfLogicalFunctions>
```

Dynamics — Petri net ⁽¹⁾



In Petri net, the firing condition is derived from the network :

```
if  $A \geq 1 \wedge B \geq 2$  then  
   $A := A - 1$   
   $B := B - 2$   
   $C := C + 2$   
else  
  nothing
```

Dynamics — Petri net ⁽²⁾

if $A \geq 1 \wedge B \geq 2$ then

$A := A - 1$

$B := B - 2$

$C := C + 2$

else

nothing

Dynamics — Petri net ⁽²⁾

```
if  $A \geq 1 \wedge B \geq 2$  then  
   $A := A - 1$   
   $B := B - 2$   
   $C := C + 2$   
else  
  nothing
```



```
<listOfLogicalFunctions>  
  <logicalFunction result="1">  
    <math>  
       $A \geq 1 \wedge B \geq 2$   
    </math>  
  </logicalFunction>  
  
  <defaultFunction result="0" />  
</listOfLogicalFunctions>
```

Dynamics — Petri net ⁽²⁾

```
if  $A \geq 1 \wedge B \geq 2$  then
```

```
   $A := A - 1$ 
```

```
   $B := B - 2$ 
```

```
   $C := C + 2$ 
```

```
else
```

```
  nothing
```



```
<listOfLogicalFunctions>
```

```
  <logicalFunction result="1">
```

```
    <math>
```

```
       $A \geq 1 \wedge B \geq 2$ 
```

```
    </math>
```

```
  </logicalFunction>
```

```
  <defaultFunction result="0" />
```

```
</listOfLogicalFunctions>
```

```
<listOfInputs>
```

```
  <input
```

```
    qualitativeSpecies="A"
```

```
    consumptionFactor="1"
```

```
  />
```

```
  <input
```

```
    qualitativeSpecies="B"
```

```
    consumptionFactor="2"
```

```
  />
```

```
</listOfInputs>
```

```
<listOfOutputs>
```

```
  <output
```

```
    qualitativeSpecies="C"
```

```
    productionFactor="2"
```

```
  />
```

```
</listOfOutputs>
```

Dynamics — Petri net (2)

```
if  $A \geq 1 \wedge B \geq 2$  then  
   $A := A - 1$   
   $B := B - 2$   
   $C := C + 2$   
else  
  nothing
```



```
<listOfLogicalFunctions>  
  <logicalFunction result="1">  
    <math> $A \geq 1 \wedge B \geq 2$ </math>  
  </logicalFunction>  
  <defaultFunction result="0" />  
</listOfLogicalFunctions>  
  
<listOfInputs>  
  <input  
    qualitativeSpecies="A"  
    consumptionFactor="1"  
  />  
  <input  
    qualitativeSpecies="B"  
    consumptionFactor="2"  
  />  
</listOfInputs>  
  
<listOfOutputs>  
  <output  
    qualitativeSpecies="C"  
    productionFactor="2"  
  />  
</listOfOutputs>
```



```
 $A := A - 1 * \text{result}$   
 $B := B - 2 * \text{result}$   
 $C := C + 2 * \text{result}$ 
```

Dynamics

$$C := \begin{cases} 2 & \text{if } A = 2 \wedge B = 1 \\ 1 & \text{if } A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0 \\ 0 & \text{otherwise} \end{cases}$$



```
<listOfLogicalFunctions>
  <logicalFunction result="2">
    <math>A = 2 \wedge B = 1</math>
  </logicalFunction>
  <logicalFunction result="1">
    <math>A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0</math>
  </logicalFunction>
  <defaultFunction result="0" />
</listOfLogicalFunctions>
```

Dynamics

$$C := \begin{cases} 2 & \text{if } A = 2 \wedge B = 1 \\ 1 & \text{if } A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0 \\ 0 & \text{otherwise} \end{cases}$$



```
<listOfLogicalFunctions>
  <logicalFunction result="2">
    <math>
      A = 2 \wedge B = 1
    </math>
  </logicalFunction>
  <logicalFunction result="1">
    <math>
      A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0
    </math>
  </logicalFunction>
  <defaultFunction result="0" />
</listOfLogicalFunctions>
```

```
if transitionEffect="change"
  input := input - consumptionFactor * result
  output := output + productionFactor * result
```

```
if transitionEffect="assignment"
  input := consumptionFactor * result
  output := productionFactor * result
```

Dynamics

$$C := \begin{cases} 2 & \text{if } A = 2 \wedge B = 1 \\ 1 & \text{if } A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0 \\ 0 & \text{otherwise} \end{cases}$$

⇒

```
if transitionEffect="change"
  input := input - consumptionFactor * result
  output := output + productionFactor * result

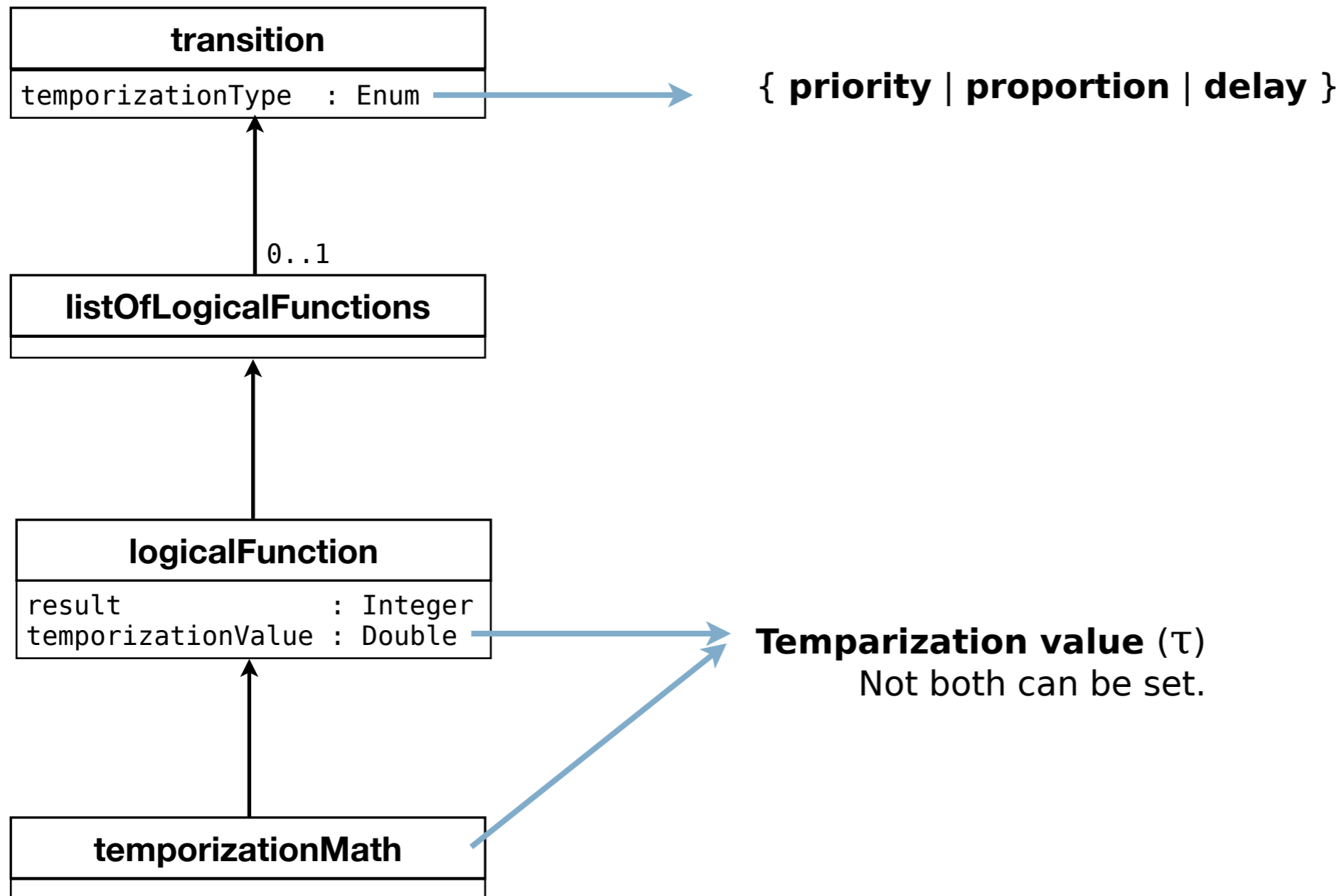
if transitionEffect="assignment"
  input := consumptionFactor * result
  output := productionFactor * result
```

```
<listOfLogicalFunctions>
  <logicalFunction result="2">
    <math>A = 2 \wedge B = 1</math>
  </logicalFunction>
  <logicalFunction result="1">
    <math>A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0</math>
  </logicalFunction>
  <defaultFunction result="0" />
</listOfLogicalFunctions>

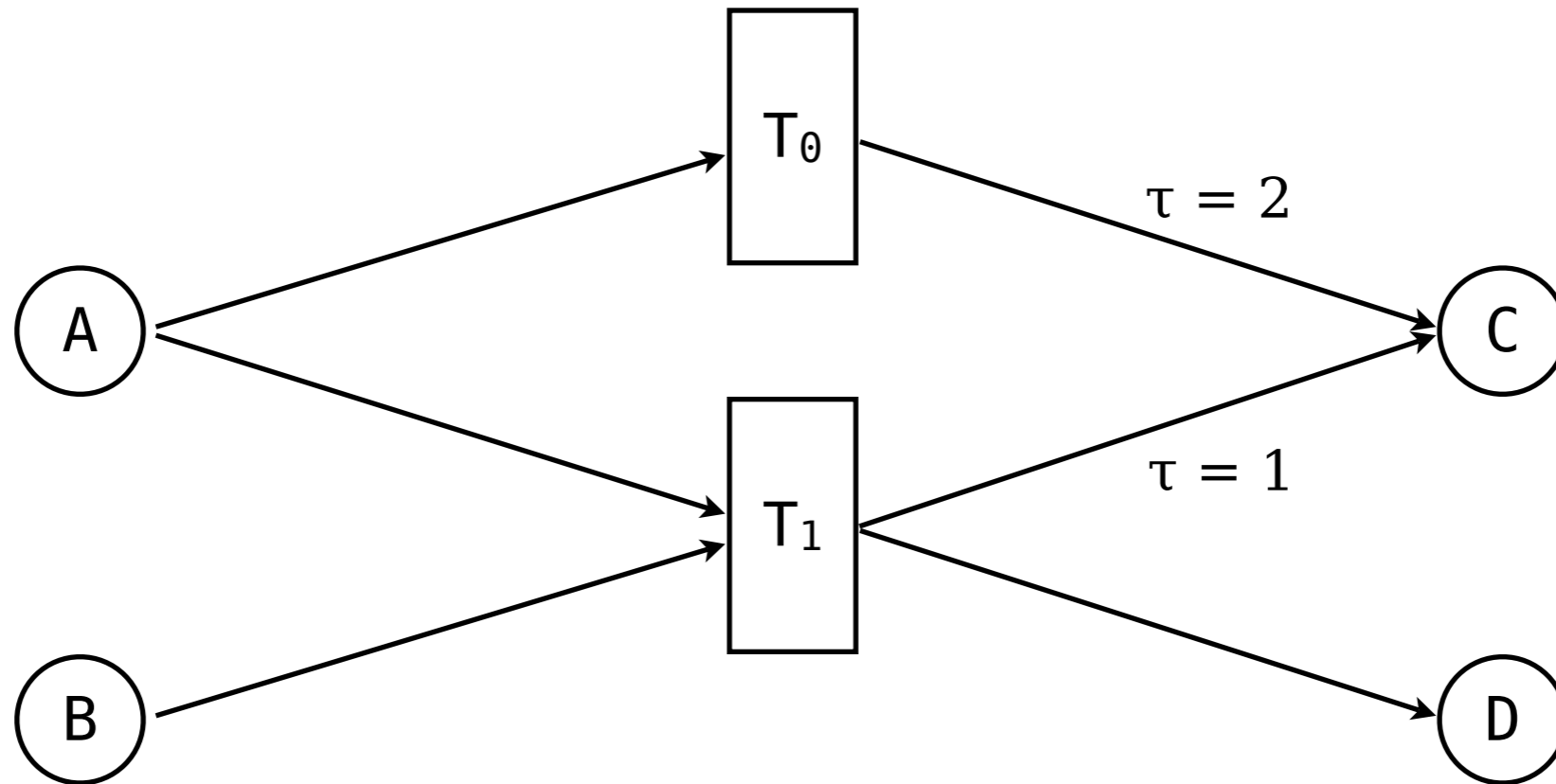
<listOfInputs>
  <input
    qualitativeSpecies="A"
    transitionEffect="change"
    consumptionFactor="0"
  />
  <input
    qualitativeSpecies="B"
    transitionEffect="change"
    consumptionFactor="0"
  />
</listOfInputs>

<listOfOutputs>
  <output
    qualitativeSpecies="C"
    transitionEffect="assignment"
    productionFactor="1"
  />
</listOfOutputs>
```

Time — encoding

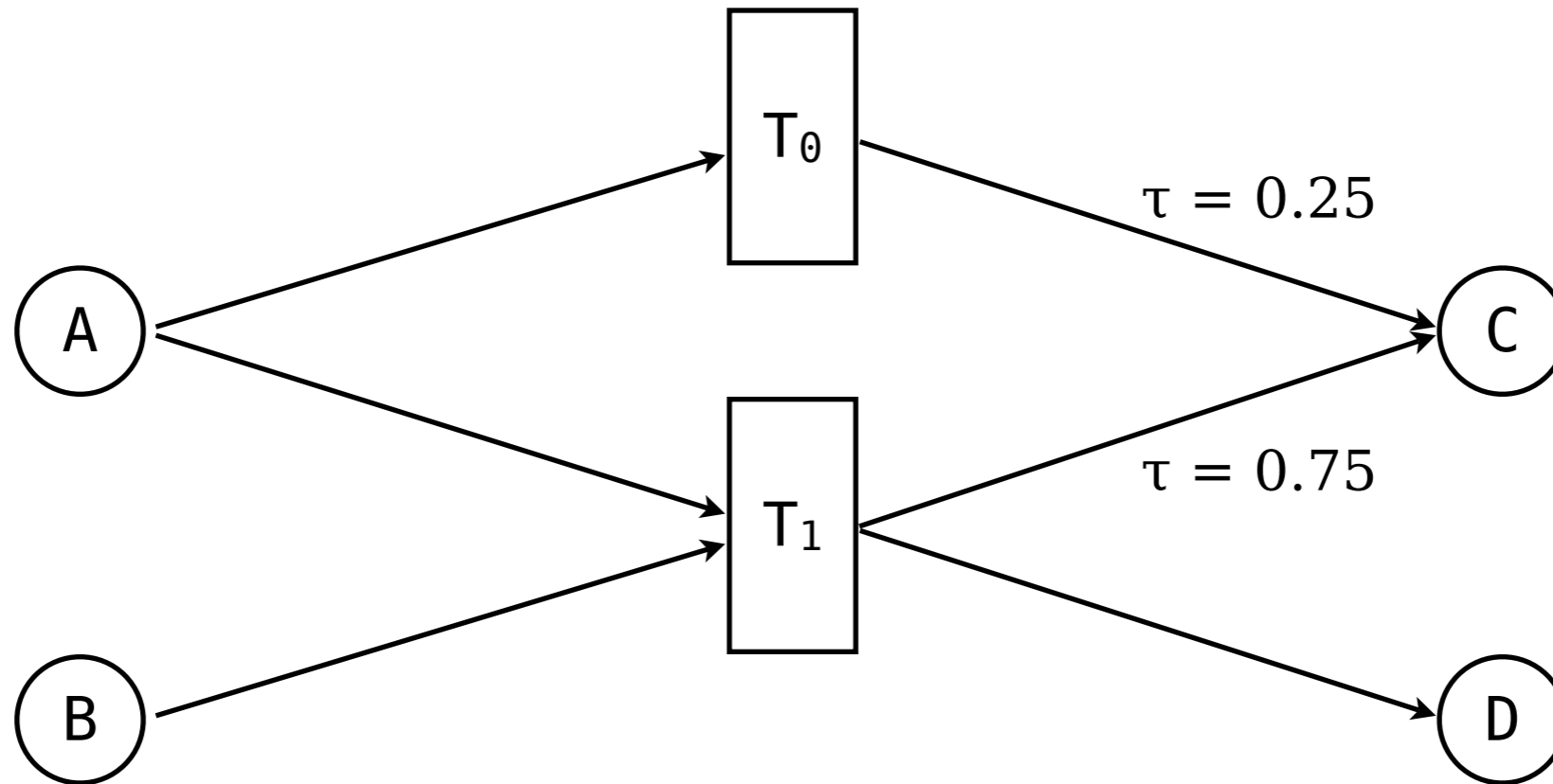


Time — priority



Lowest τ fire first

Time — proportion

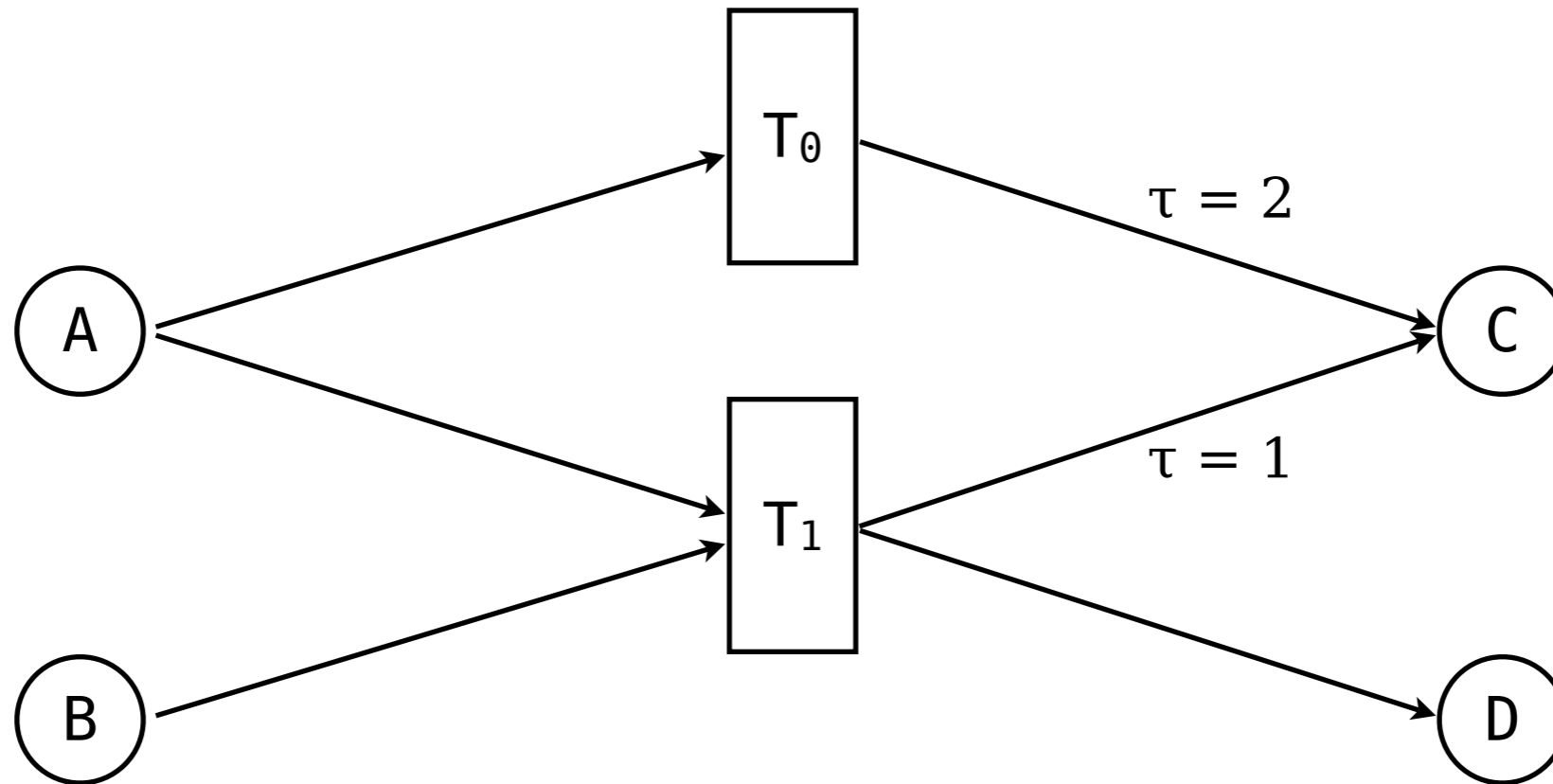


Each arc has a certain probability τ to fire.

How to run the simulation?

- run one simulation, where T_0 fire with a probability of 25%.
- run 100 simulations, where T_0 fire with a probability of 25%.
- run 100 simulations, where T_0 fire 25 times and T_1 75 times.

Time — delay



As long as the logicalFunctions are true, the timer run for τ iterations.
The first logicalFunction's timer to complete is choose to fire.
Did we reset the clock each time one is completed ?

Thank you for listening

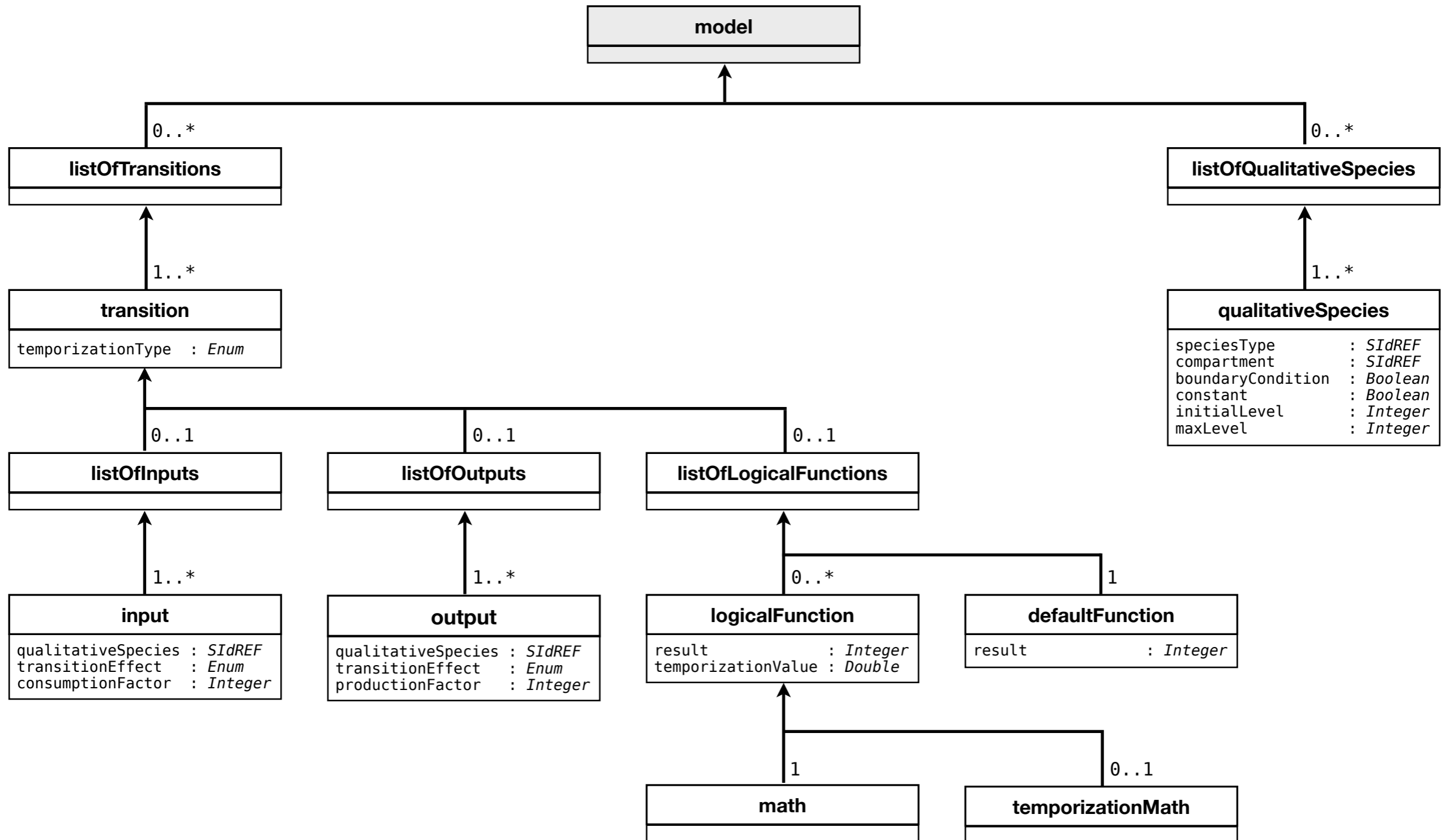
Any questions ?

Web site : <http://compneur.wiki.ebi.ac.uk/xwiki/bin/view/SBML/L3F>

Thanks to Nicolas Le novère, for having answer all my « little » questions.

Thanks to all the people involved in the discussions.

Annexe A – Diagram



Annexe B — Dynamics — Logical Model

$$C := \begin{cases} 2 & \text{if } A = 2 \wedge B = 1 \\ 1 & \text{if } A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0 \\ 0 & \text{otherwise} \end{cases} \Rightarrow$$

```

if result = 2
  A := A
  B := B
  C := 2

if result = 1
  A := A
  B := B
  C := 1

if result = 0
  A := A
  B := B
  C := 0
    
```

```

<listOfLogicalFunctions>
  <logicalFunction result="2">
    <math>
      A = 2 \wedge B = 1
    </math>
  </logicalFunction>

  <logicalFunction result="1">
    <math>
      A = 0 \wedge B = 1 \vee A = 1 \wedge B = 0
    </math>
  </logicalFunction>

  <defaultFunction result="0" />
</listOfLogicalFunctions>
<listOfInputs>
  <input
    qualitativeSpecies="A"
    transitionEffect="change"
    consumptionFactor="0"
  />
  <input
    qualitativeSpecies="B"
    transitionEffect="change"
    consumptionFactor="0"
  />
</listOfInputs>
<listOfOutputs>
  <output
    qualitativeSpecies="C"
    transitionEffect="assignment"
    productionFactor="1"
  />
</listOfOutputs>
    
```



Annexe B — Dynamics — Petri nets

```
if  $A \geq 1 \wedge B \geq 2$  then
```

```
  A := A - 1
```

```
  B := B - 2
```

```
  C := C + 2
```

```
else
```

```
  nothing
```



```
<listOfLogicalFunctions>
```

```
  <logicalFunction result="1">  
    <math> $A \geq 1 \wedge B \geq 2$ </math>  
  </logicalFunction>
```

```
  <defaultFunction result="0" />
```

```
</listOfLogicalFunctions>
```

```
<listOfInputs>
```

```
  <input  
    qualitativeSpecies="A"  
    transitionEffect="change"  
    consumptionFactor="1"
```

```
  />
```

```
  <input  
    qualitativeSpecies="B"  
    transitionEffect="change"  
    consumptionFactor="2"
```

```
  />
```

```
</listOfInputs>
```

```
<listOfOutputs>
```

```
  <output  
    qualitativeSpecies="C"  
    transitionEffect="change"  
    productionFactor="2"
```

```
  />
```

```
</listOfOutputs>
```



```
if result = 1
```

```
  A := A - 1
```

```
  B := B - 2
```

```
  C := C + 2
```

```
if result = 0
```

```
  A := A
```

```
  B := B
```

```
  C := C
```