

# Programação Concorrente e Paralela

Capítulo 9 – Redes de Petri

Marcial Porto Fernández  
marcial@larces.uece.br

Semestre 2018.2

# Sumário

- Rede de Petri:
  - Lugares, transições, marcas, entradas, saídas
  - Disparo de transições
- Modelagem:
  - Concorrência e sincronização
- Propriedades das Redes de Petri:
  - vivacidade, limitada
- Exemplos

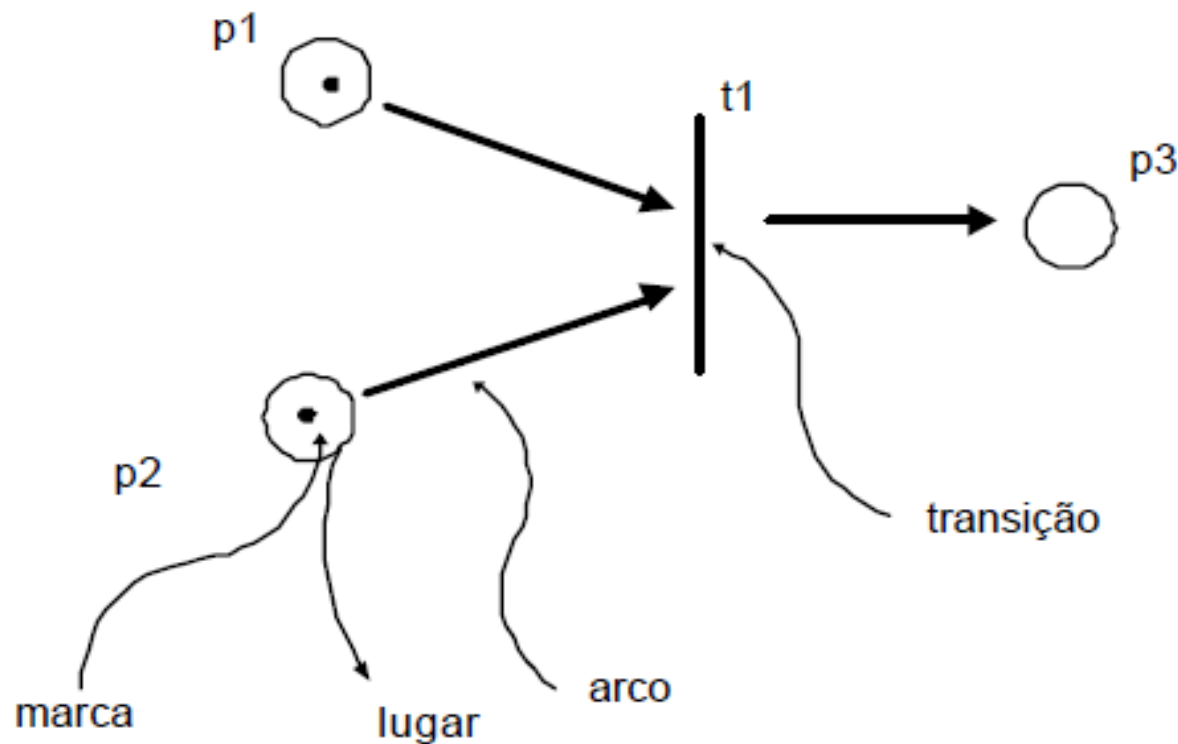
# Sumário

- Rede de Petri:
  - Lugares, transições, marcas, entradas, saídas
  - Disparo de transições
- Modelagem:
  - Concorrência e sincronização
- Propriedades das Redes de Petri:
  - vivacidade, limitada
- Exemplos

# Redes de Petri

- Proposta em 1962 por Carl Adam Petri, é uma técnica baseada em máquina de estados para analisar o funcionamento de sistemas concorrentes.
- Ferramenta usada em outros campos como: economia, biologia, engenharia, computação, etc.
- Na computação especificam as operações em ambientes de multiprocessamento.
- Aplicadas na avaliação do comportamento e do desempenho de um sistema computacional.
- Tratam concorrência, controle, conflitos, sincronização e compartilhamentos.
- Permitem analisar restrições temporais.
- Sistemas podem ser descritos graficamente.

# Elementos de Redes de Petri



- Lugar (place), Transição (transition), Arco (arc) e Marca (token)

# Redes de Petri: definição formal

- A rede de Petri  $\mathbf{C} = (\mathbf{P}, \mathbf{T}, \mathbf{I}, \mathbf{O})$  consiste em:
  - Um conjunto finito  $\mathbf{P}$  de lugares
  - Um conjunto de finito  $\mathbf{T}$  de transições
  - Uma função de entrada  $\mathbf{I}: \mathbf{T} \Rightarrow \mathbf{Nat}(\mathbf{P})$  (marcas em lugares)
  - Uma função de saída  $\mathbf{O}: \mathbf{T} \Rightarrow \mathbf{Nat}(\mathbf{P})$
  - A marcação  $\mathbf{m}$  de  $\mathbf{C}$  é um mapeamento inicial:  $\mathbf{P} \Rightarrow \mathbf{Nat}$

- Exemplo:

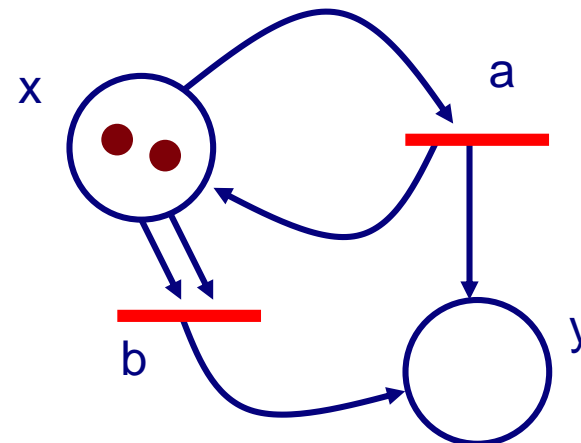
$$\mathbf{P} = \{x, y\}$$

$$\mathbf{T} = \{a, b\}$$

$$\mathbf{I}(a) = \{x\}, \quad \mathbf{I}(b) = \{x, x\}$$

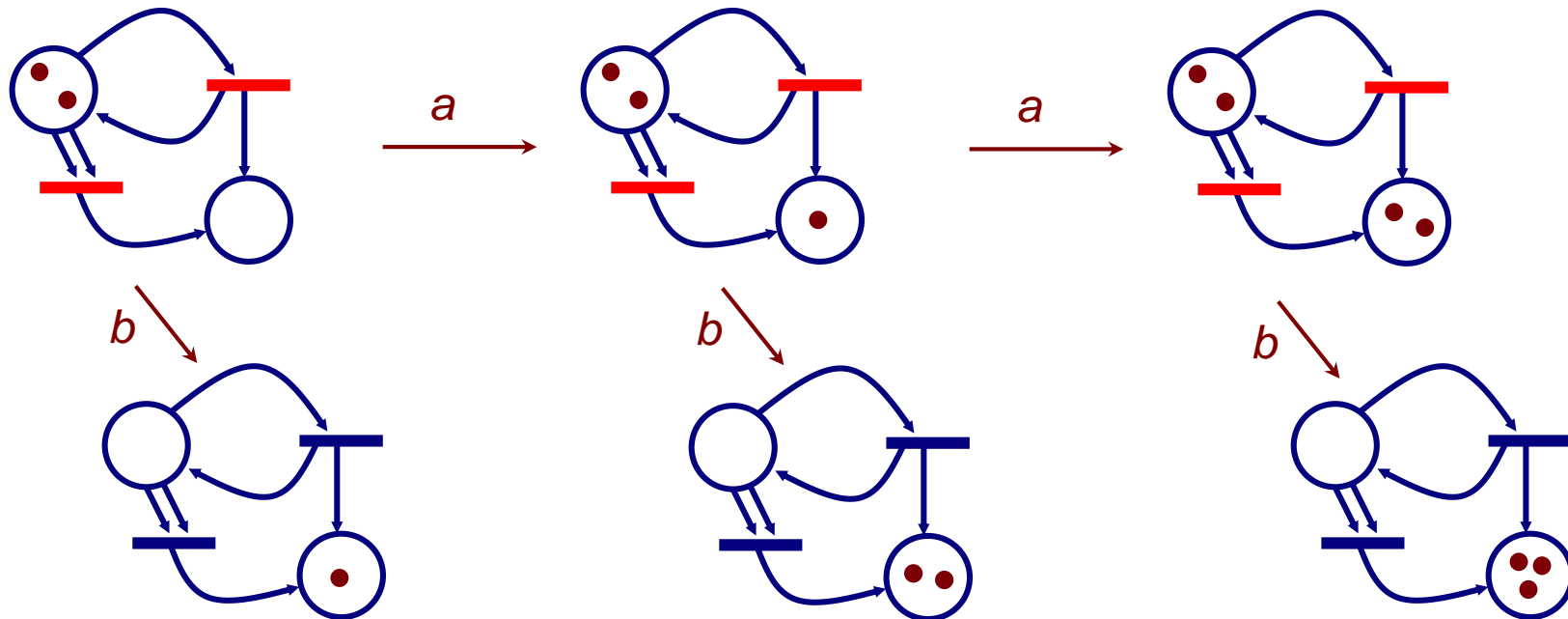
$$\mathbf{O}(a) = \{x, y\}, \quad \mathbf{O}(b) = \{y\}$$

$$\mathbf{m} = \{x, x\}$$



# Disparo de transições

- Para disparar uma transição  $t$ :
  - $t$  deve estar habilitado:  $m \geq I(t)$
  - consumir insumos e gerar saídas:  $m' = m - I(t) + O(t)$



# Sumário

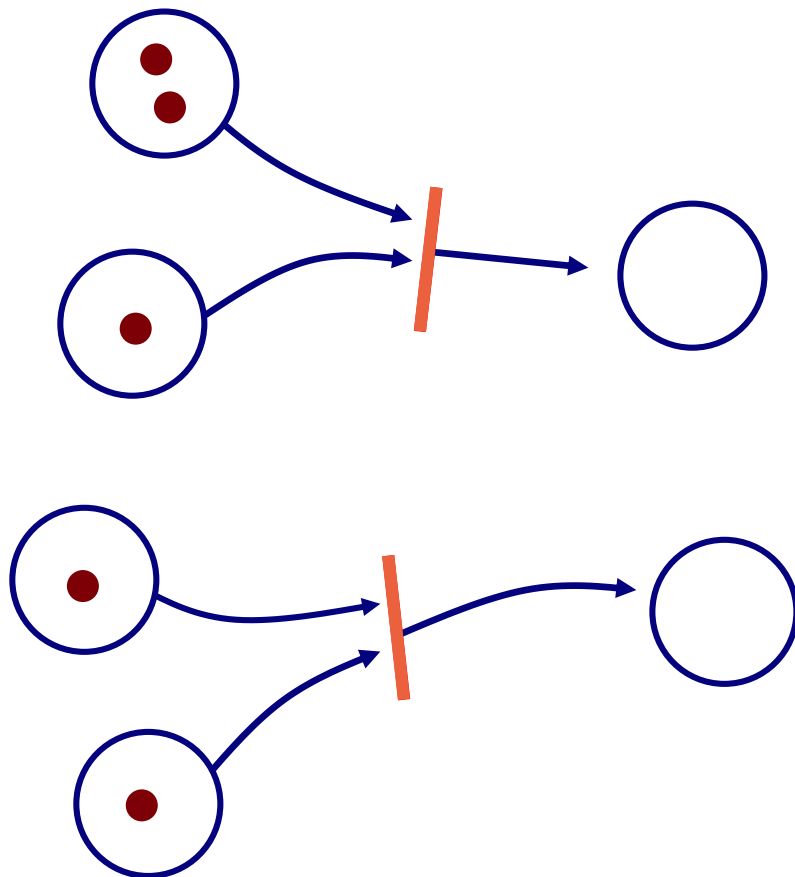
- Rede de Petri:
  - Lugares, transições, marcas, entradas, saídas
  - Disparo de transições
- **Modelagem:**
  - Concorrência e sincronização
- Propriedades das Redes de Petri:
  - vivacidade, limitada
- Exemplos



# Modelagem com Redes de Petri

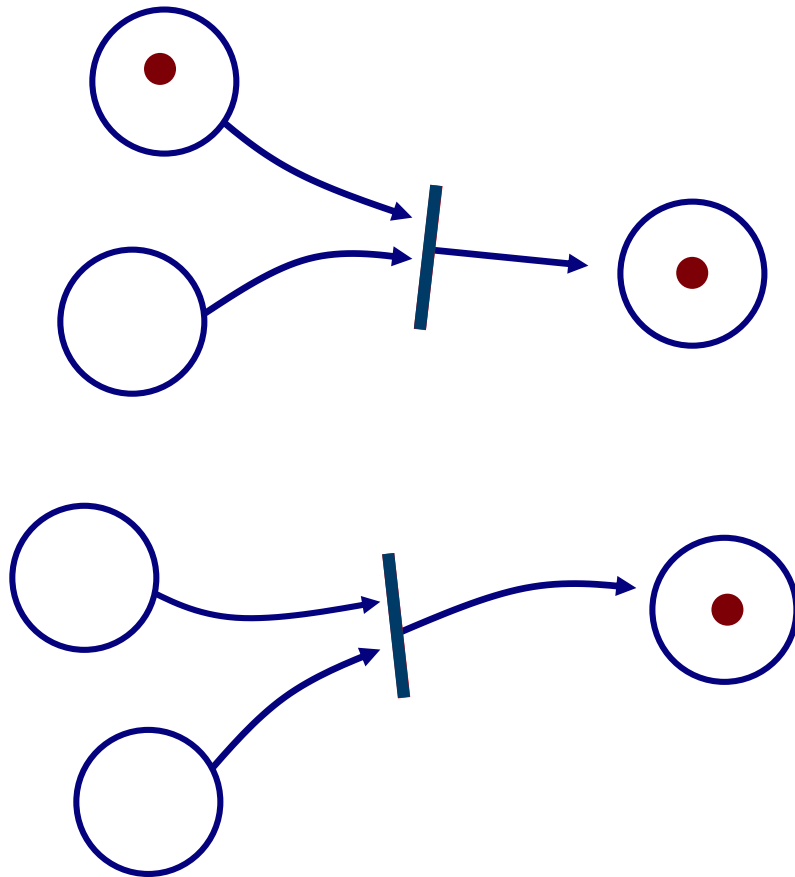
- Redes de Petri são boas para a modelagem:
  - simultaneidade
  - sincronização
- As marcas (tokens) podem representar:
  - disponibilidade de recursos
  - tarefas para realizar
  - fluxo de controle
  - condições de sincronização ...

# Simultaneidade



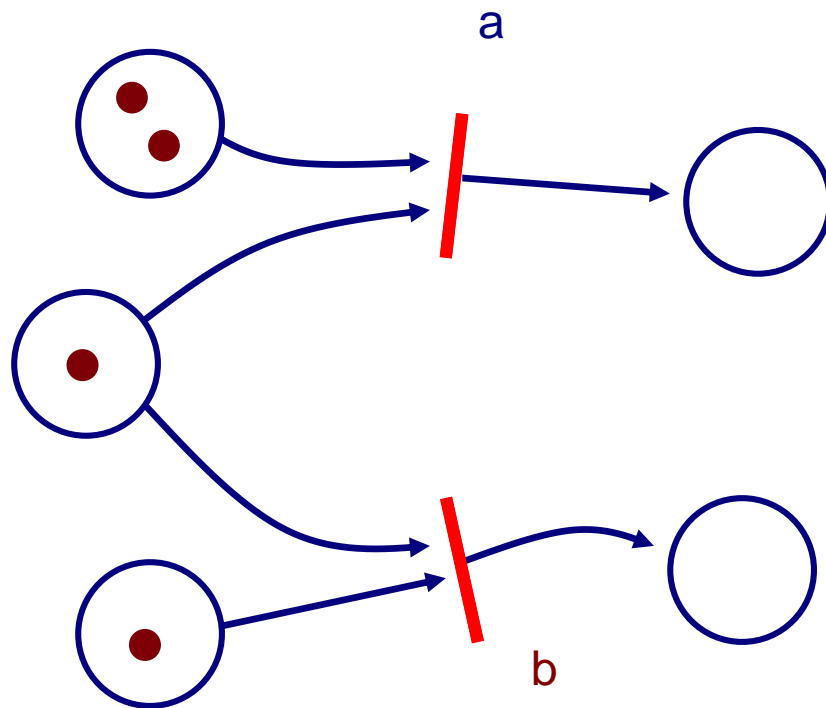
- Entradas independentes permitem o disparo "concorrente" de transições

# Simultaneidade



- Entradas independentes permitem o disparo "concorrente" de transições

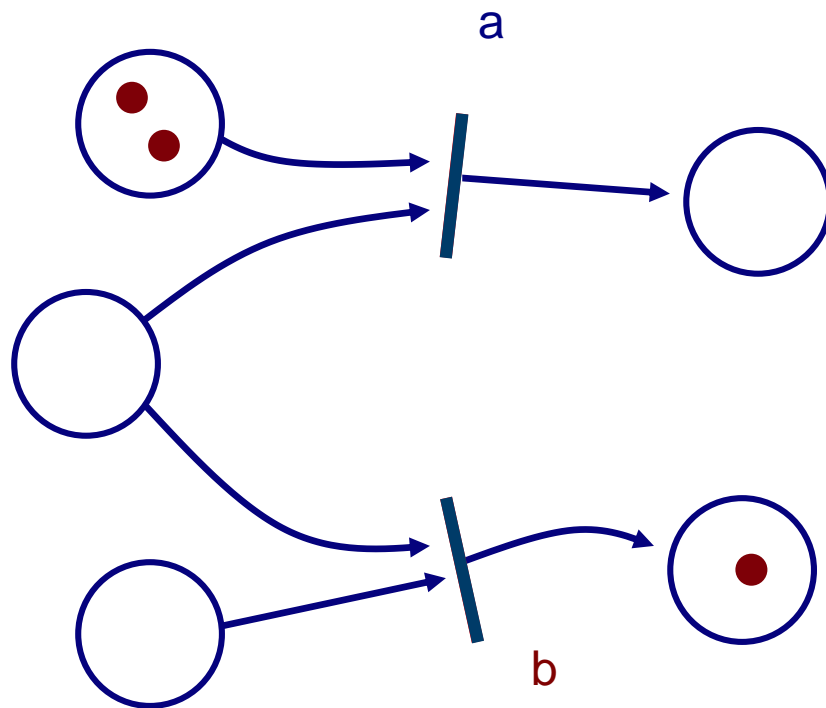
# Conflito



- Entradas sobrepostas colocam transições em conflito...

*Apenas uma transição, a ou b, pode ser disparada*

# Conflito

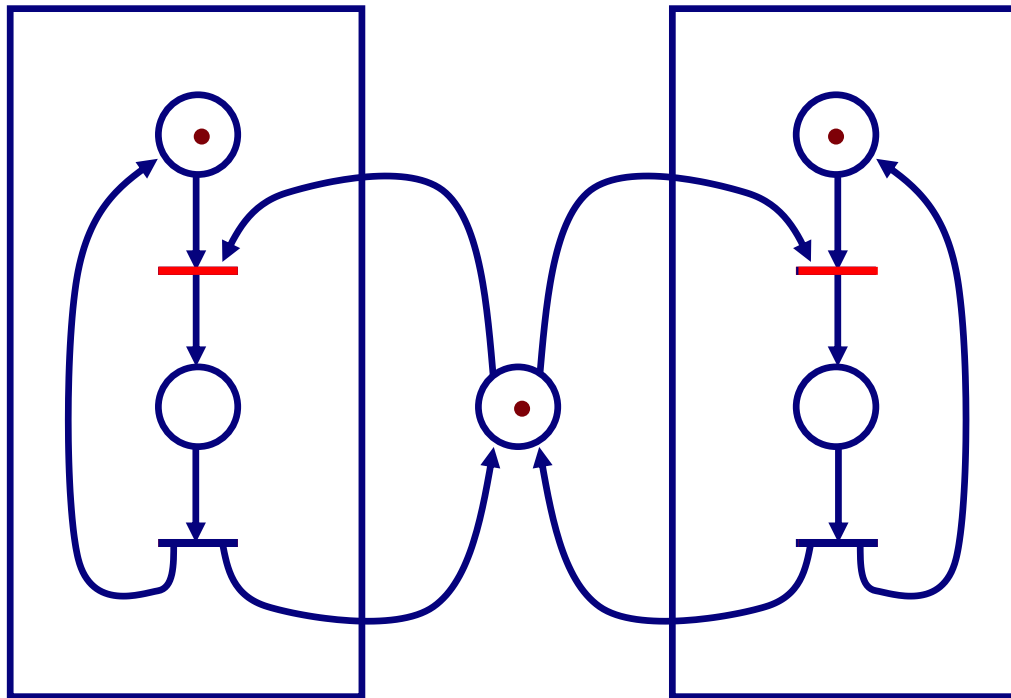


- Entradas sobrepostas colocam transições em conflito...

*b foi disparada*

# Exclusão Mútua

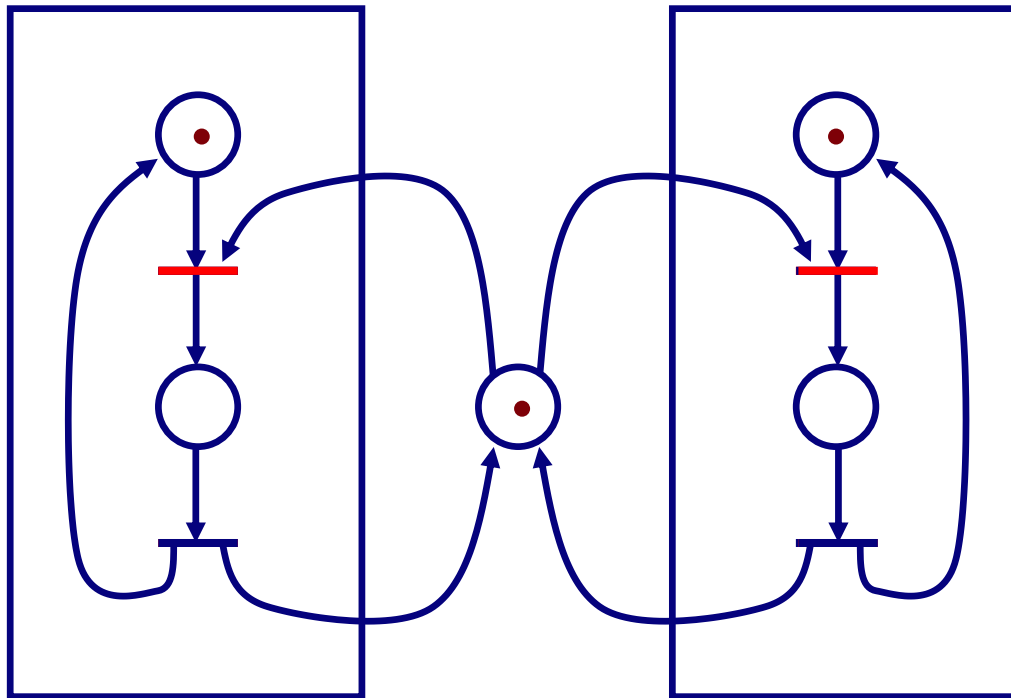
*As duas sub-redes são forçadas a se sincronizar*





# Exclusão Mútua

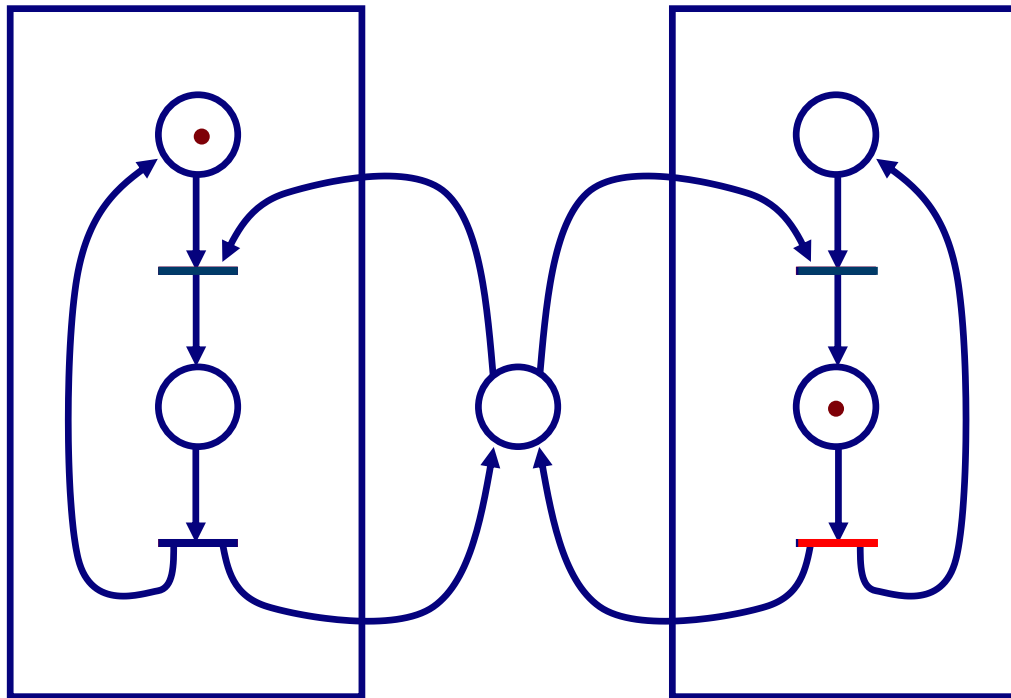
*As duas sub-redes são forçadas a se sincronizar*





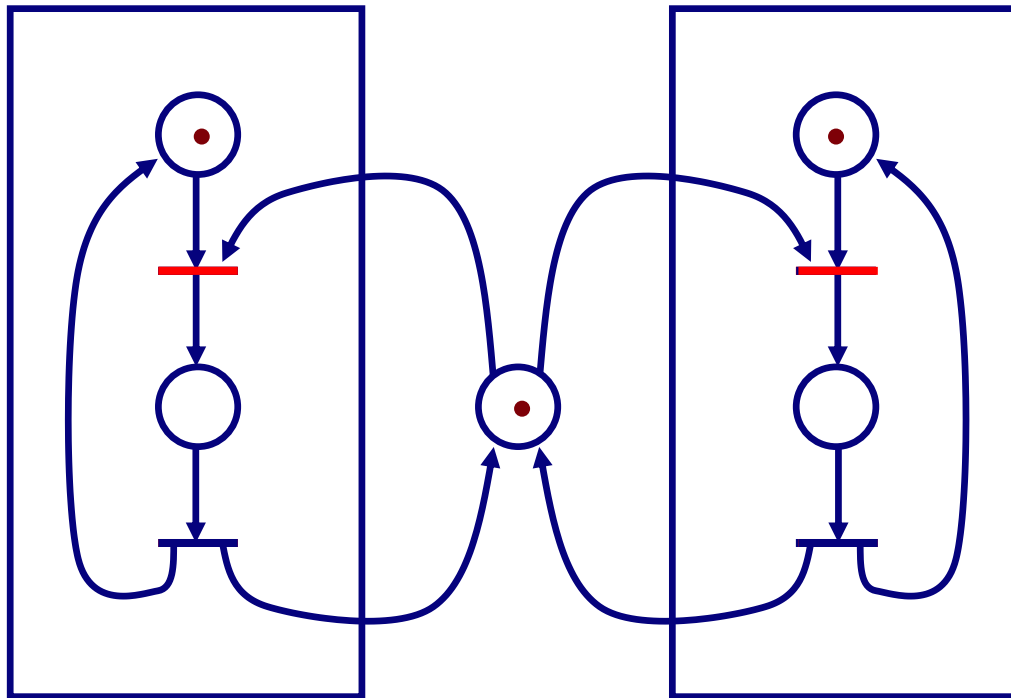
# Exclusão Mútua

*As duas sub-redes são forçadas a se sincronizar*

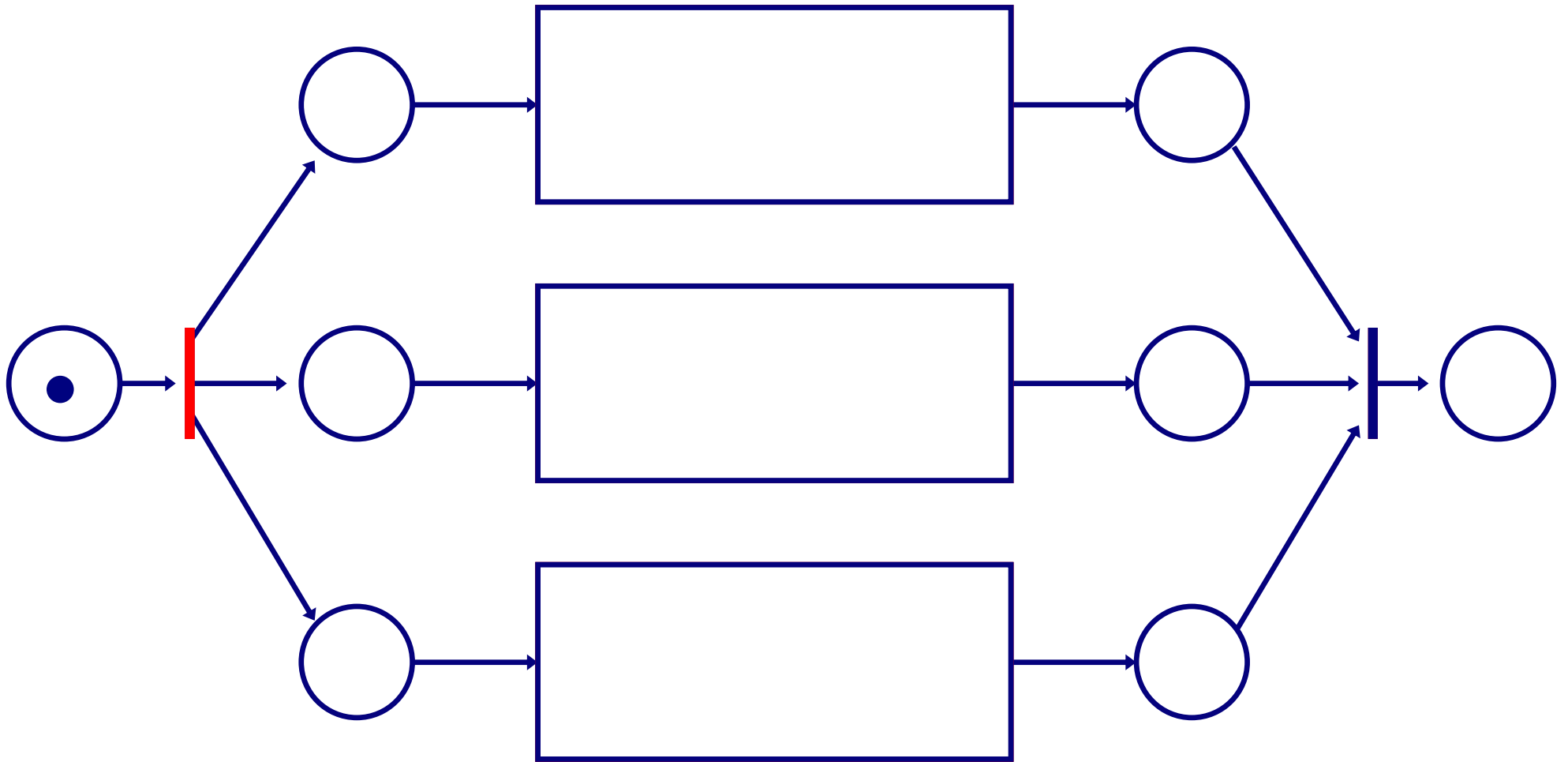


# Exclusão Mútua

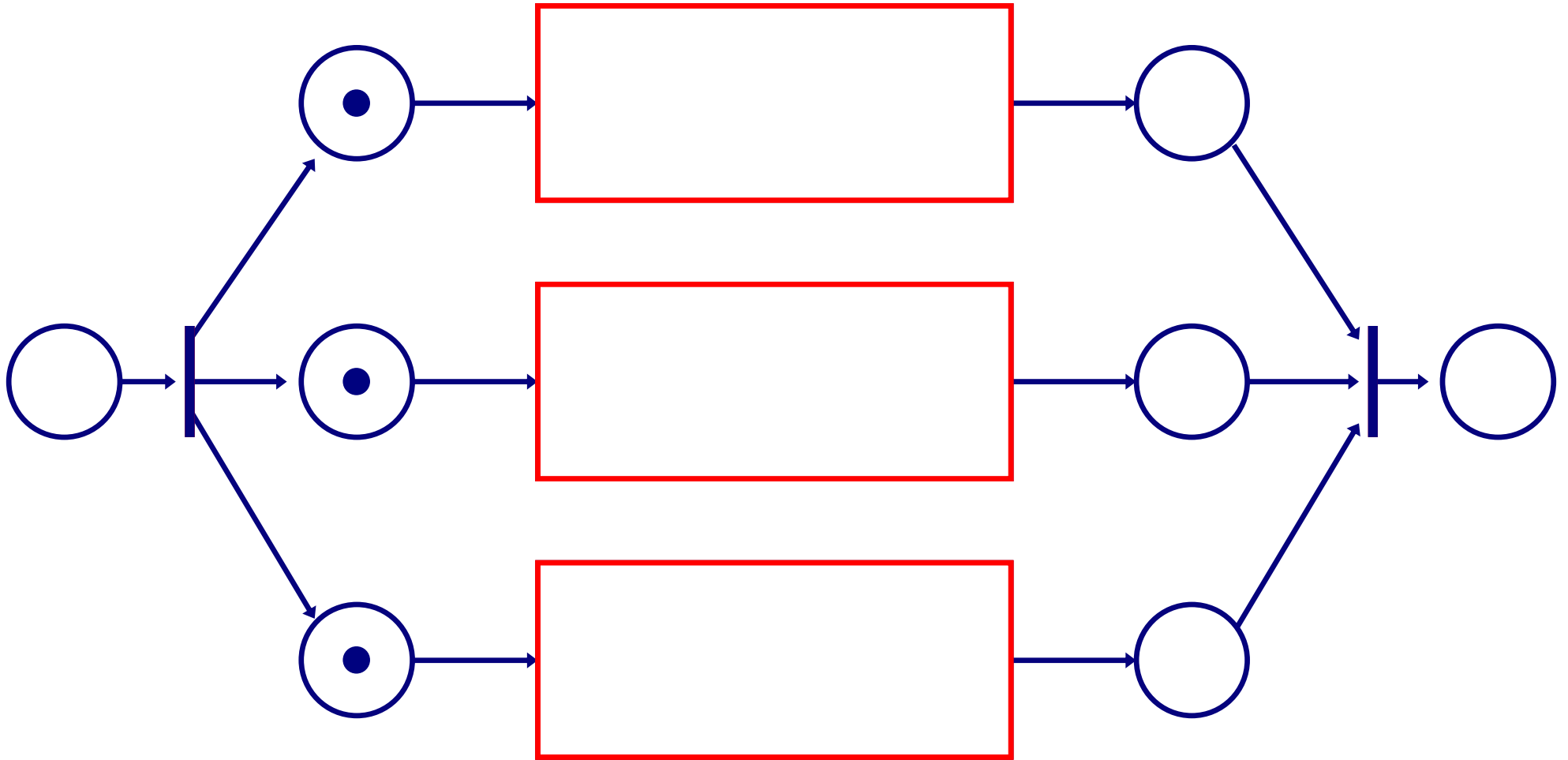
*As duas sub-redes são forçadas a se sincronizar*



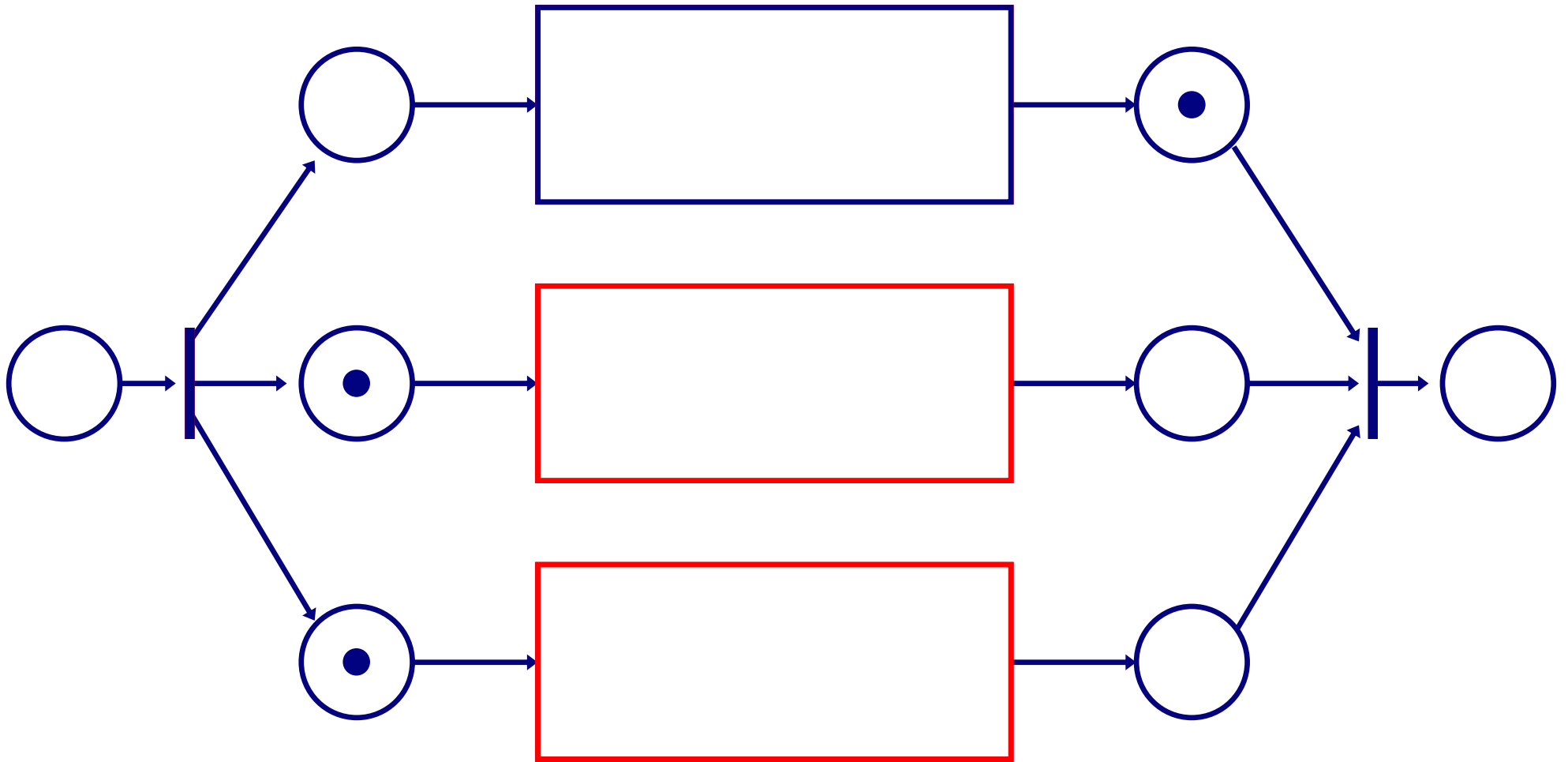
# Fork e Join



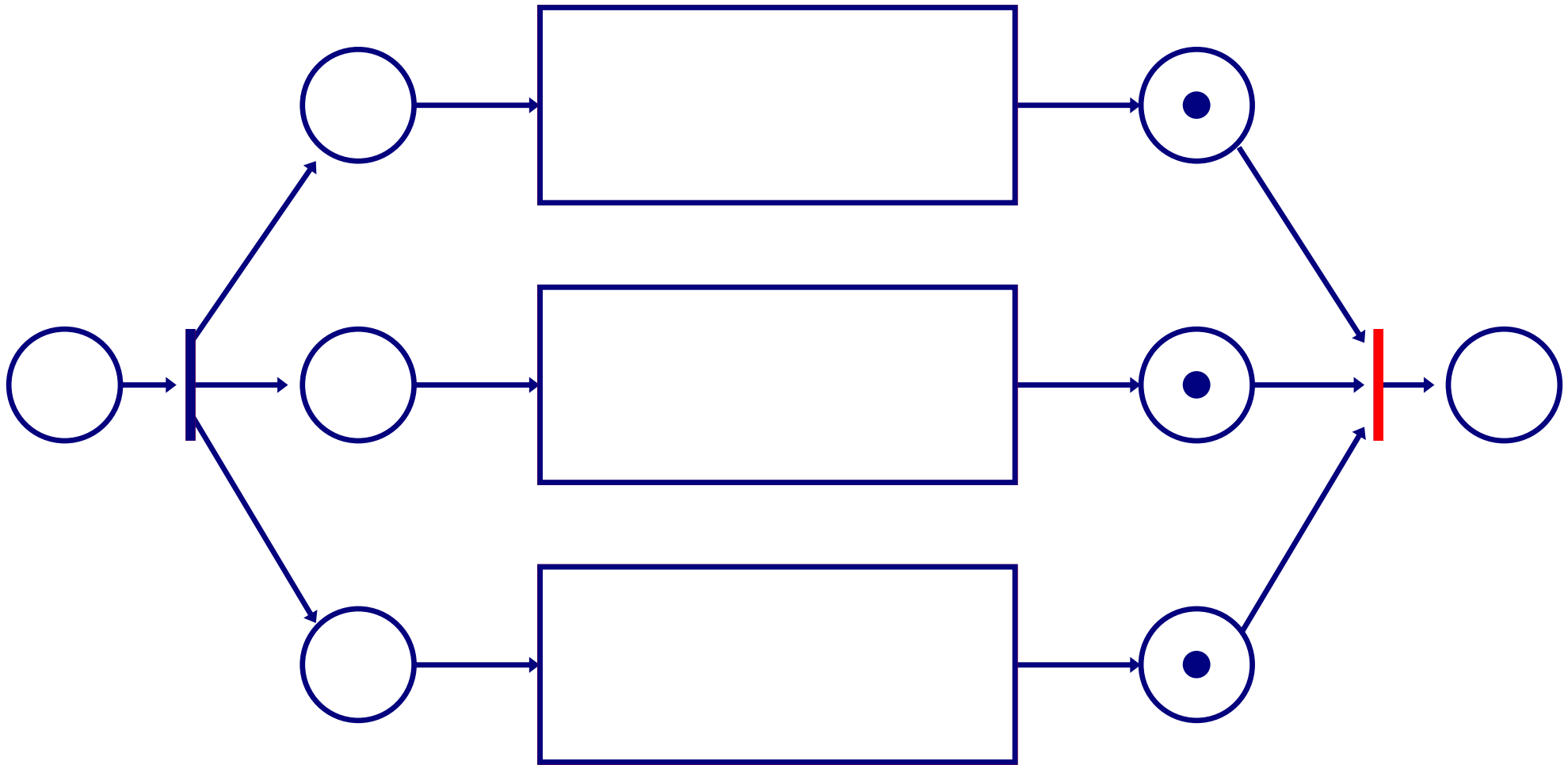
# Fork e Join



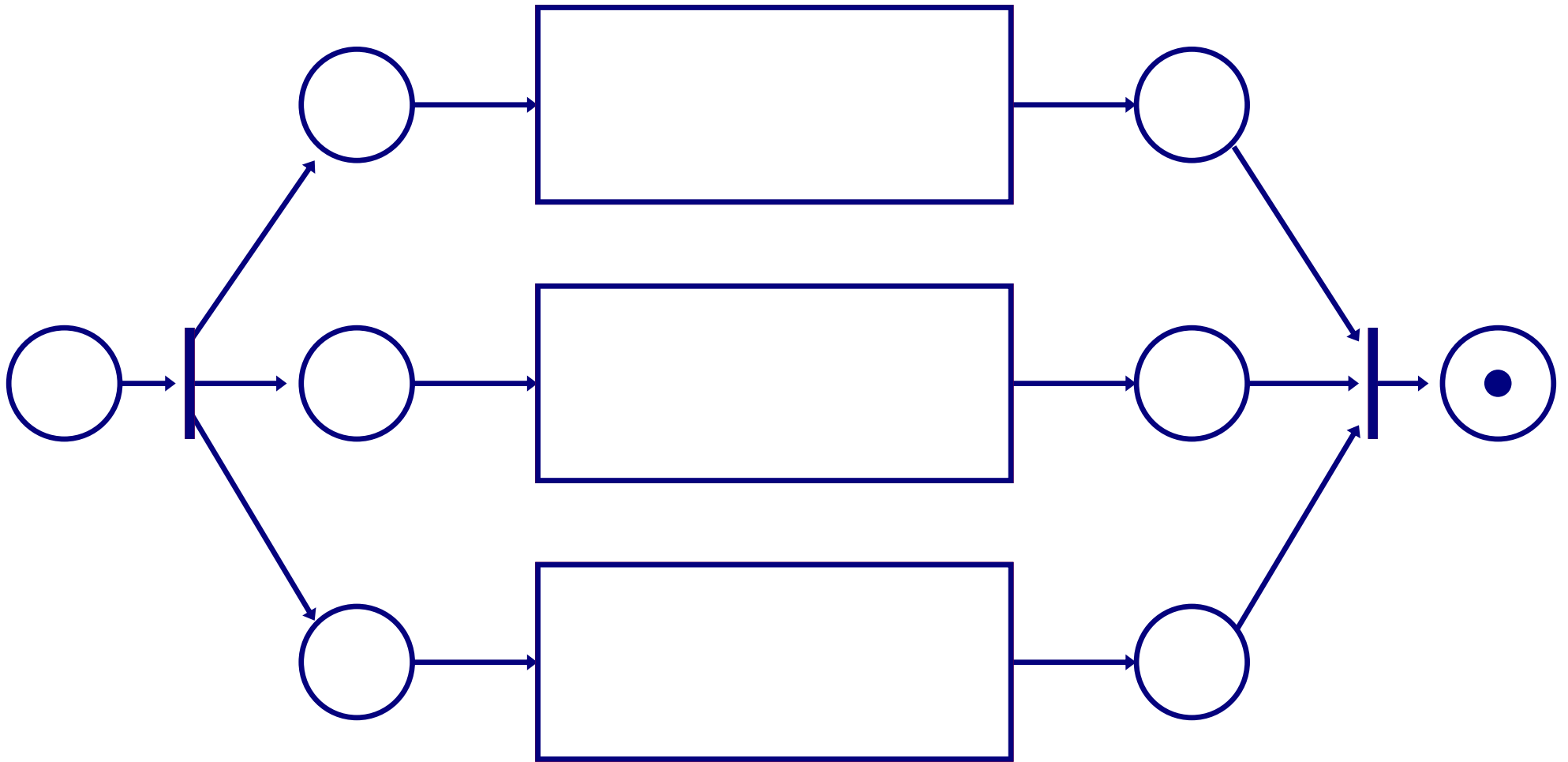
# Fork e Join



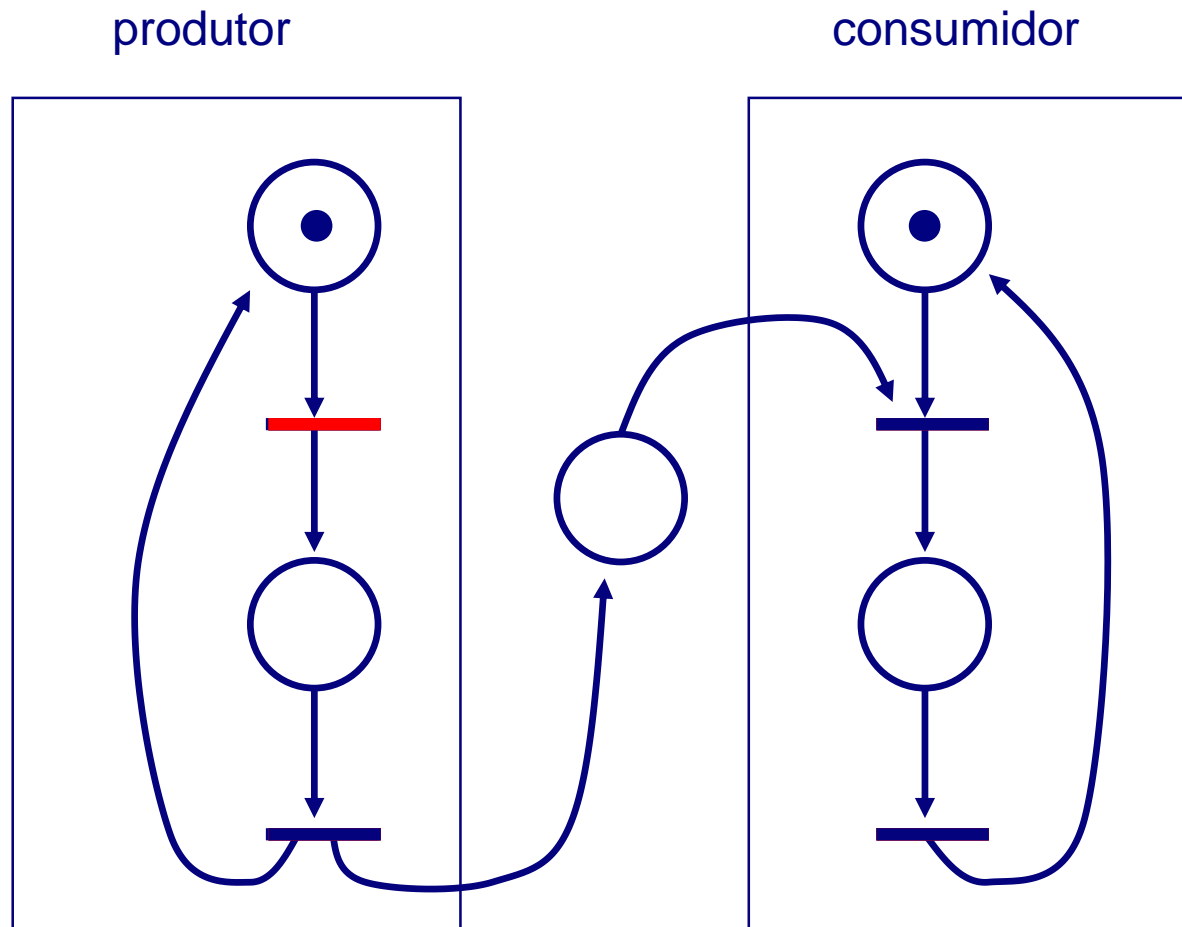
# Fork e Join



# Fork e Join

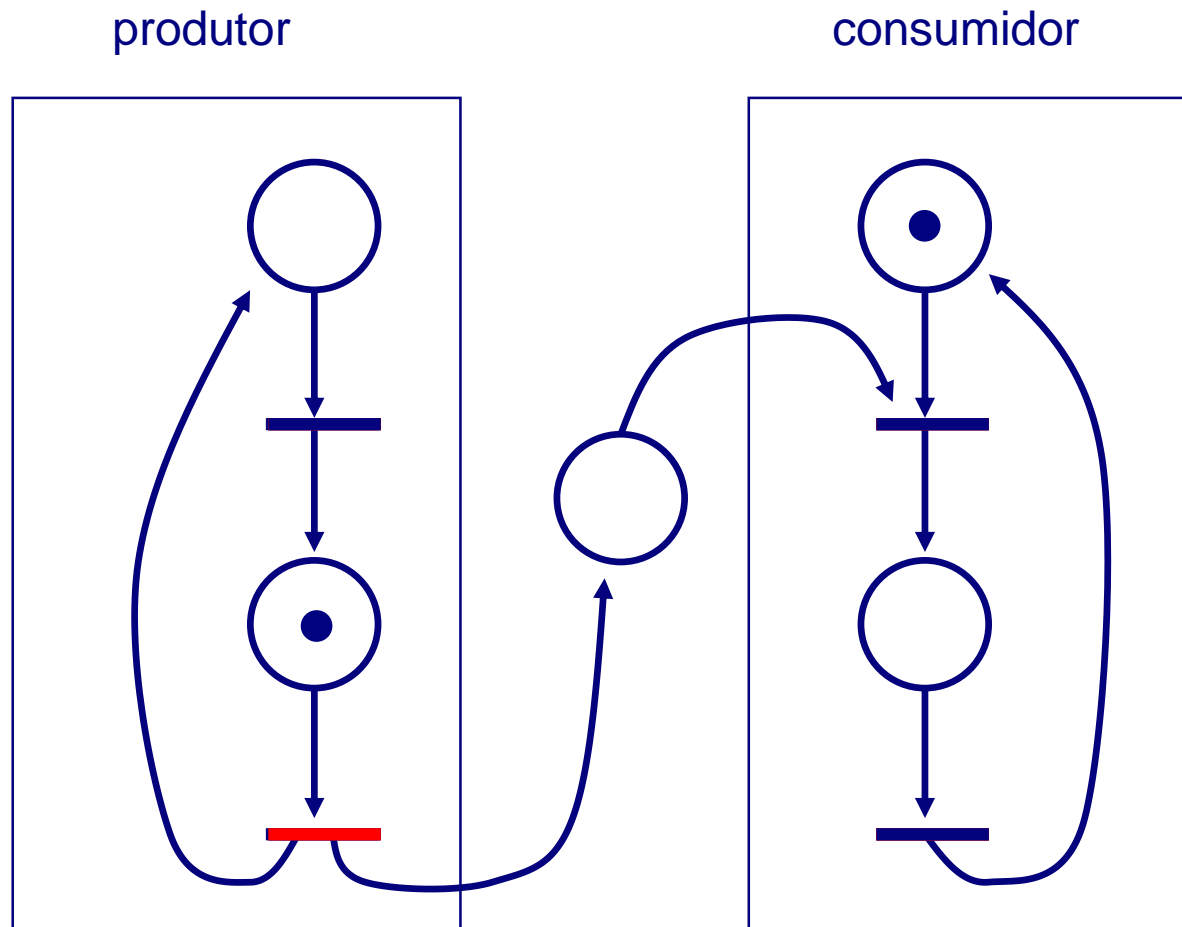


# Produtor e Consumidor

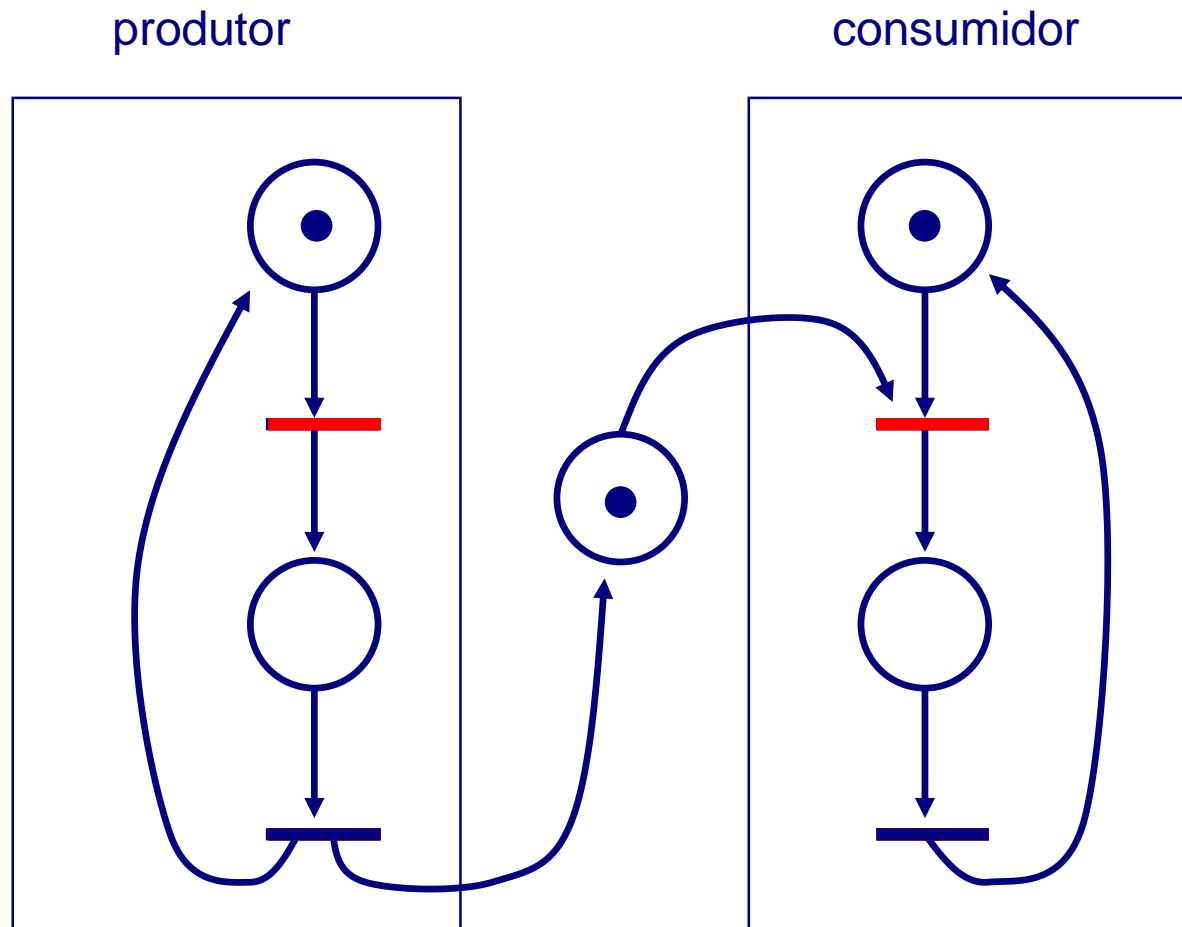




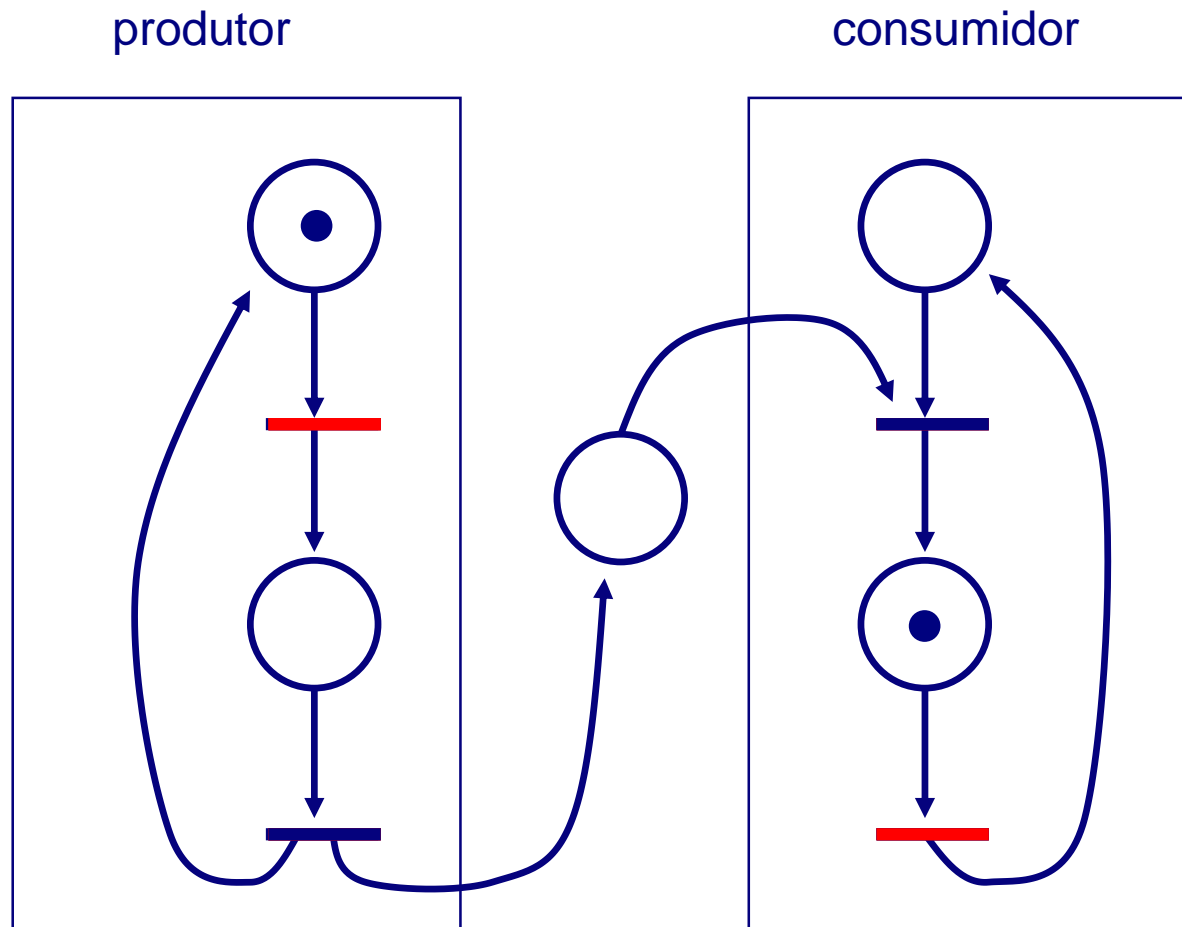
# Produtor e Consumidor



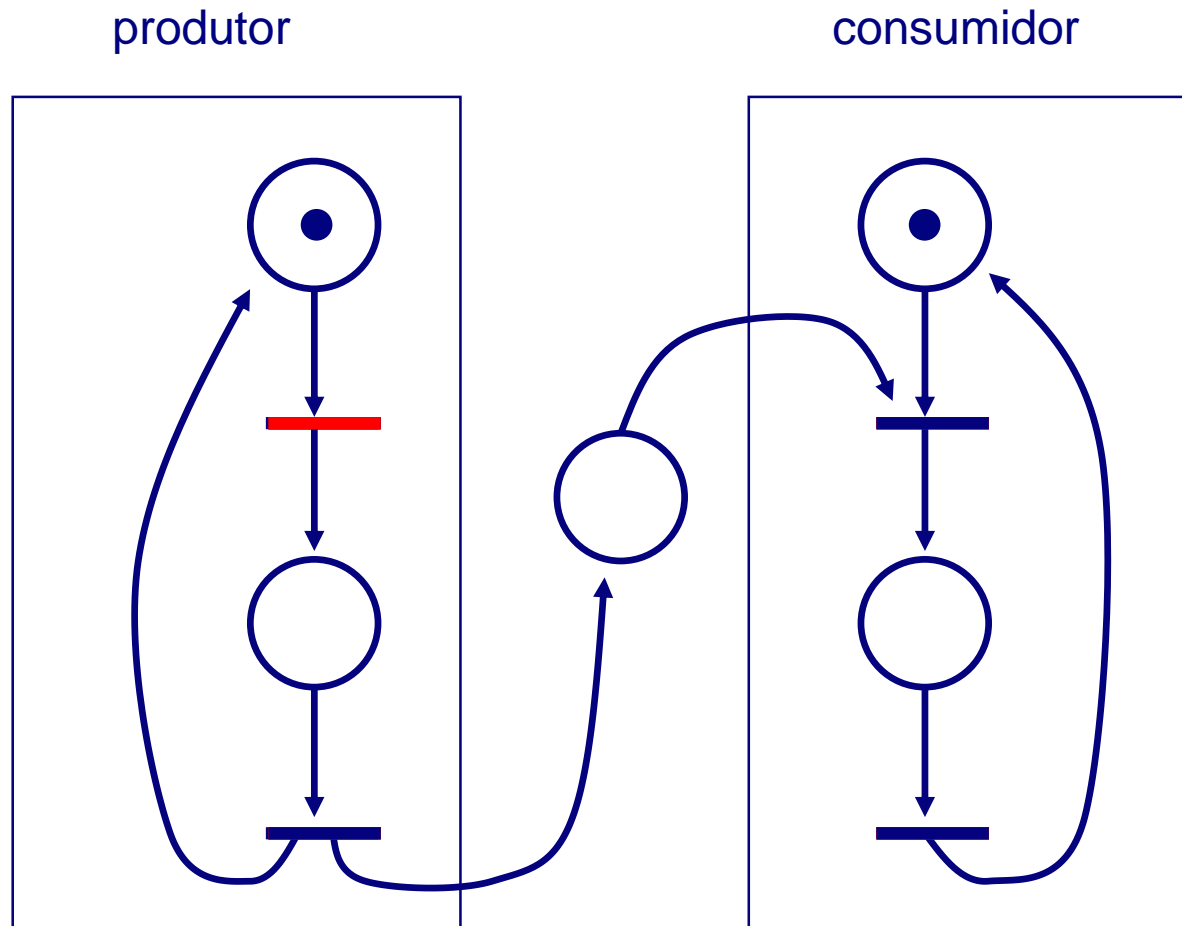
# Produtor e Consumidor



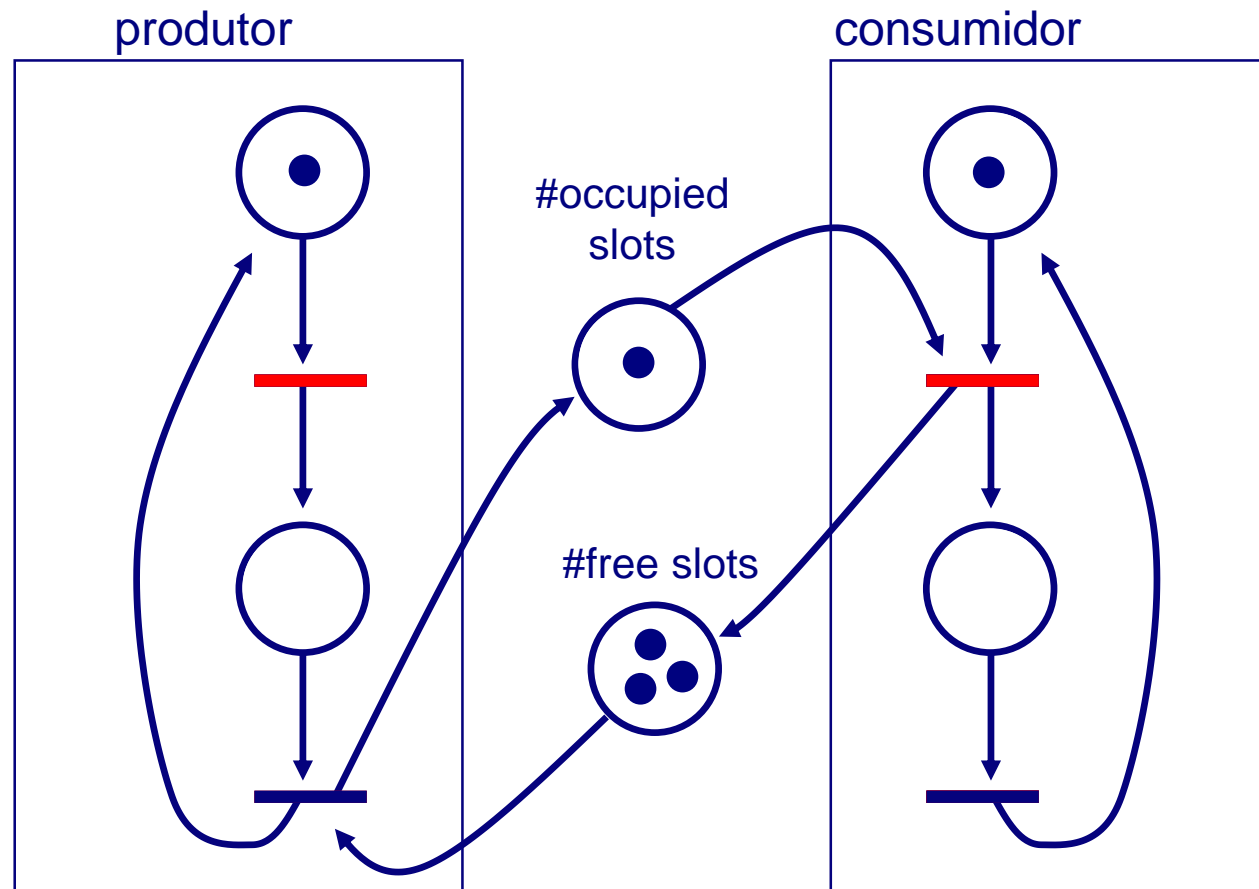
# Produtor e Consumidor



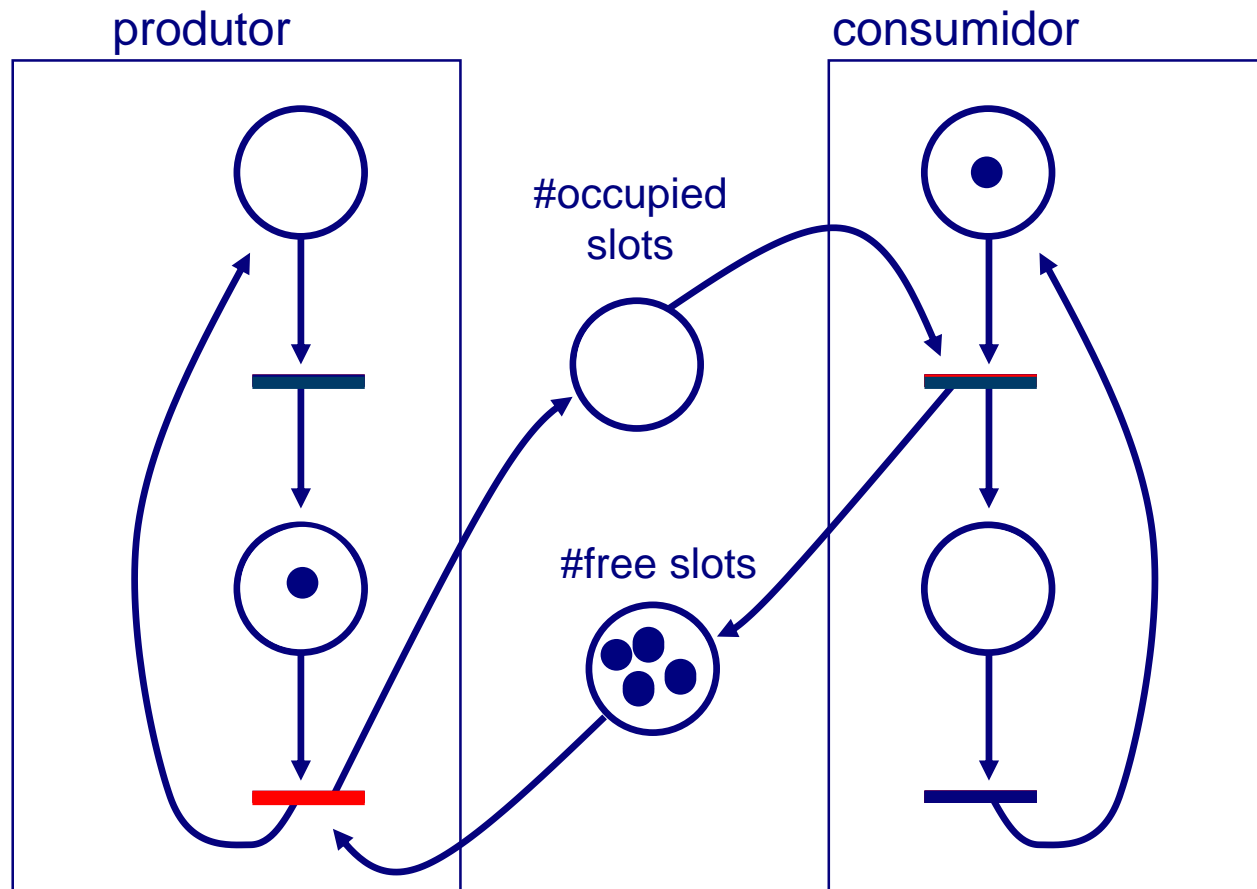
# Produtor e Consumidor



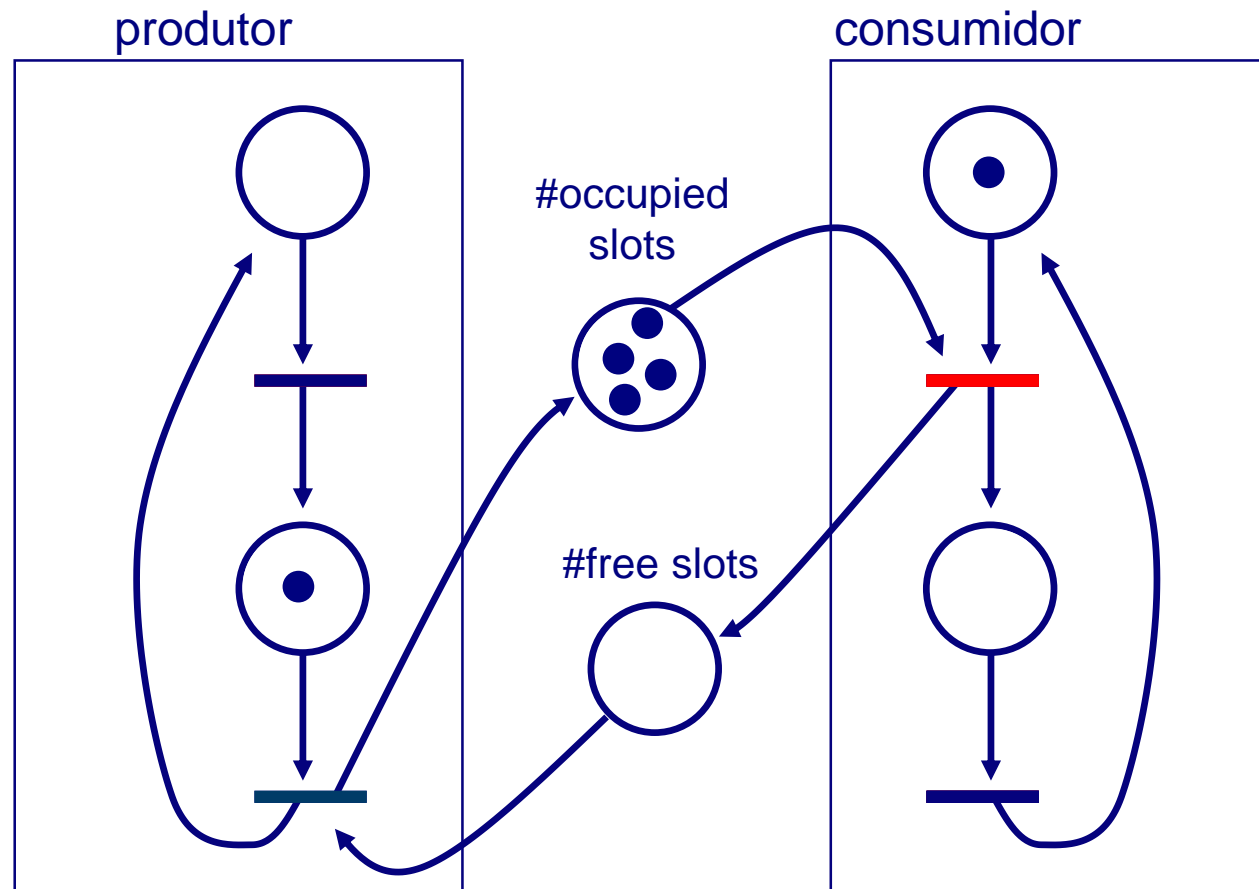
# Buffer Limitado



# Buffer Vazio



# Buffer Cheio



# Sumário

- Rede de Petri:
  - Lugares, transições, marcas, entradas, saídas
  - Disparo de transições
- Modelagem:
  - Concorrência e sincronização
- **Propriedades das Redes de Petri:**
  - **vivacidade, limitada**
- Exemplos



# Alcançabilidade e Limitação

- Alcançabilidade:
  - O grafo de alcançabilidade  $R(C,m)$  de uma rede de Petri  $C$  é o conjunto de todas as marcas  $\mu$  alcançáveis a partir da marcação inicial  $m$ .
- Limitação:
  - A rede  $C$  com marcação inicial  $m$  é segura se todo lugar em qualquer  $t$  sempre tenha no máximo 1 token.
  - Uma rede é chamada K-limitada se em qualquer lugar nunca tenha mais de  $K$  tokens.
  - Uma rede é chamada conservativa se o número de tokens é constante.

# Vivacidade e Deadlock

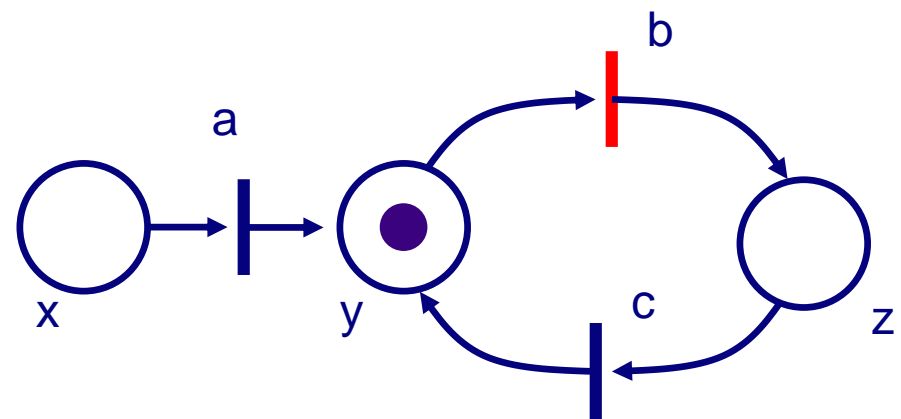
- Vivacidade:
  - Uma transição é *deadlocked* se nunca puder ser disparada.
  - Uma transição é *viva* se nunca ficar em impasse.

Esta rede é ao mesmo tempo  
*segura e conservativa*.

Transição a é *deadlocked*.

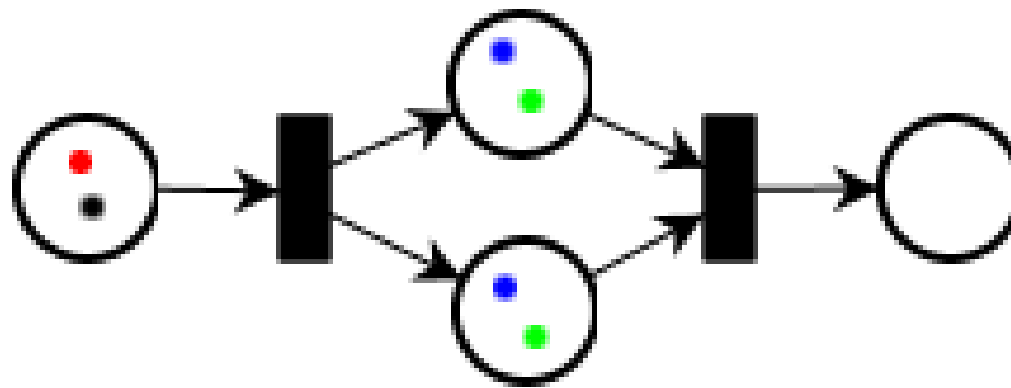
Transições b e c são *viva*.

O conjunto de alcançabilidade é  
 $\{\{y\}, \{z\}\}$ .



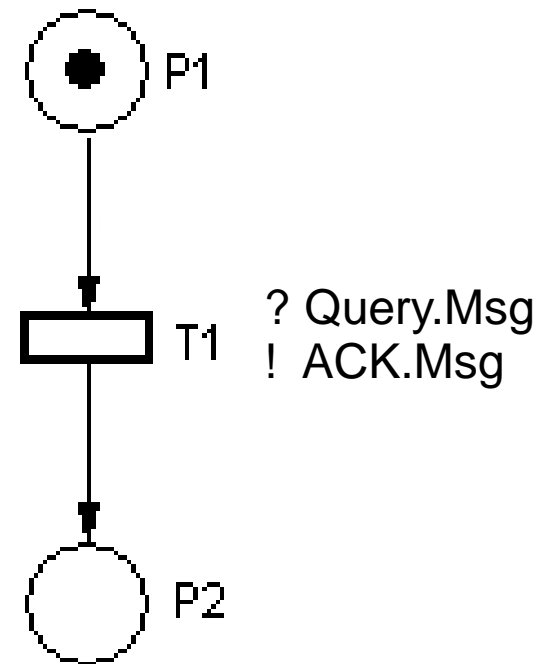
# Extensões de Redes de Petri

- Redes de Petri Coloridas:
  - Os tokens são "coloridos" para representar diferentes tipos de recursos



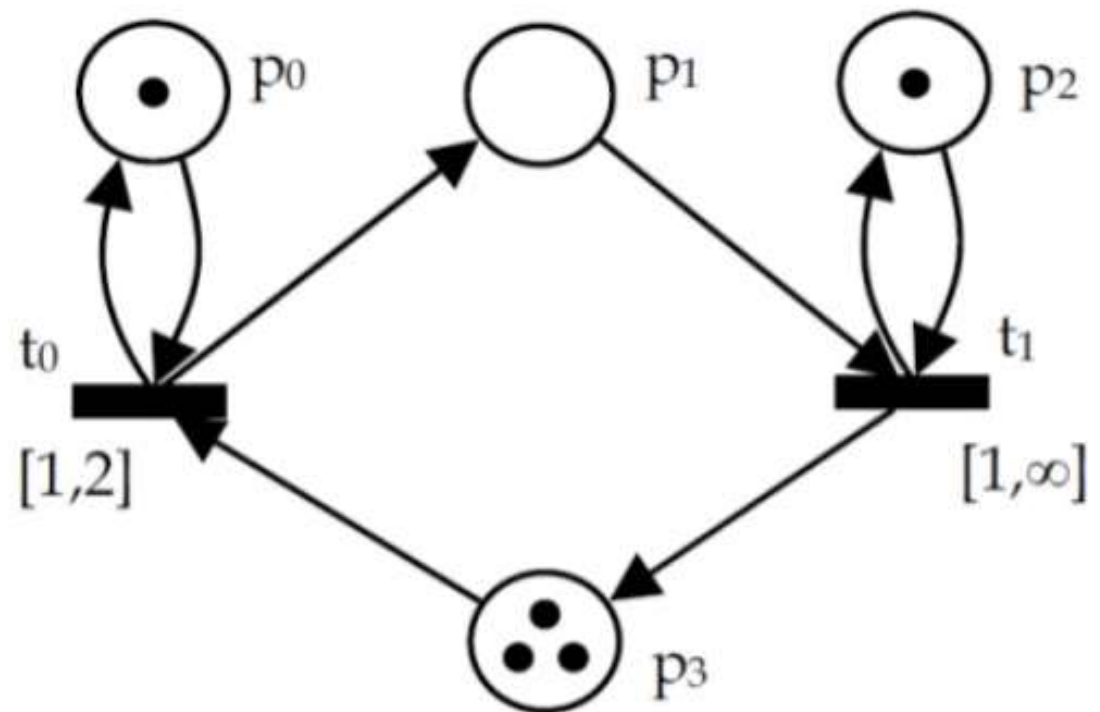
# Extensões de Redes de Petri

- Redes de Petri Condição-Ação:
  - Transições dependem adicionalmente de condições externas



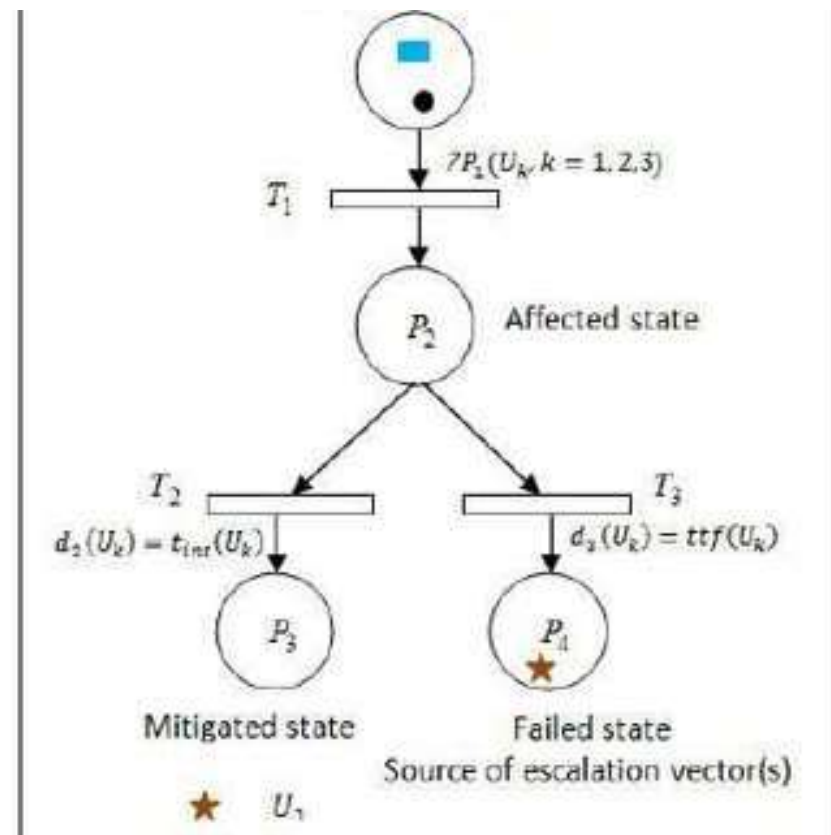
# Extensões de Redes de Petri

- Redes de Petri Temporizadas:
  - A cada transição está associado um tempo para ocorrer o disparo



# Extensões de Redes de Petri

- Redes de Petri Estocástica:
  - O disparo de cada transição está associado uma variável aleatória de tempo



# Sumário

- Rede de Petri:
  - Lugares, transições, marcas, entradas, saídas
  - Disparo de transições
- Modelagem:
  - Concorrência e sincronização
- Propriedades das Redes de Petri:
  - vivacidade, limitada
- Exemplos

# Exemplos

- Uso de ferramenta gráfica:

PIPE <http://pipe2.sourceforge.net/>