

LABORATORIO DI INFORMATICA

0011

“La disumanità del computer sta nel fatto che, una volta programmato e messo in funzione, si comporta in maniera perfettamente onesta.”
(Isaac Asimov)

Giorgio Poletti

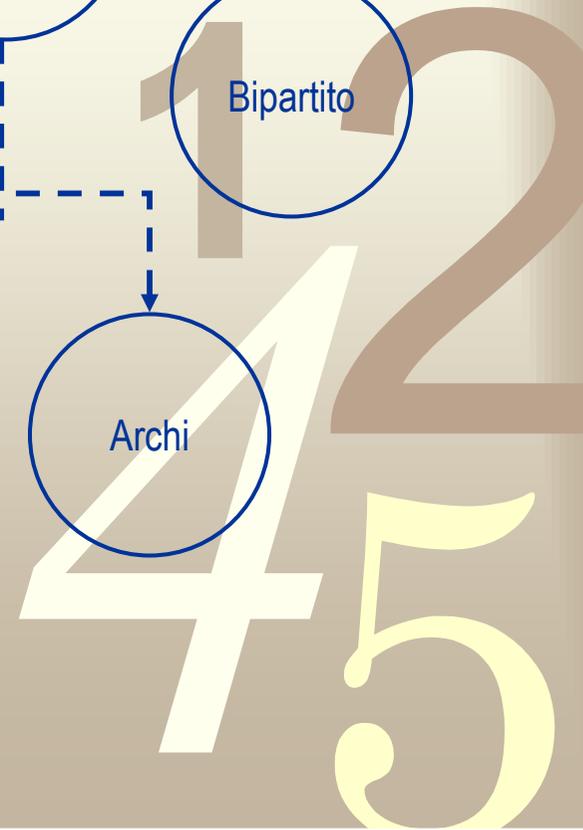
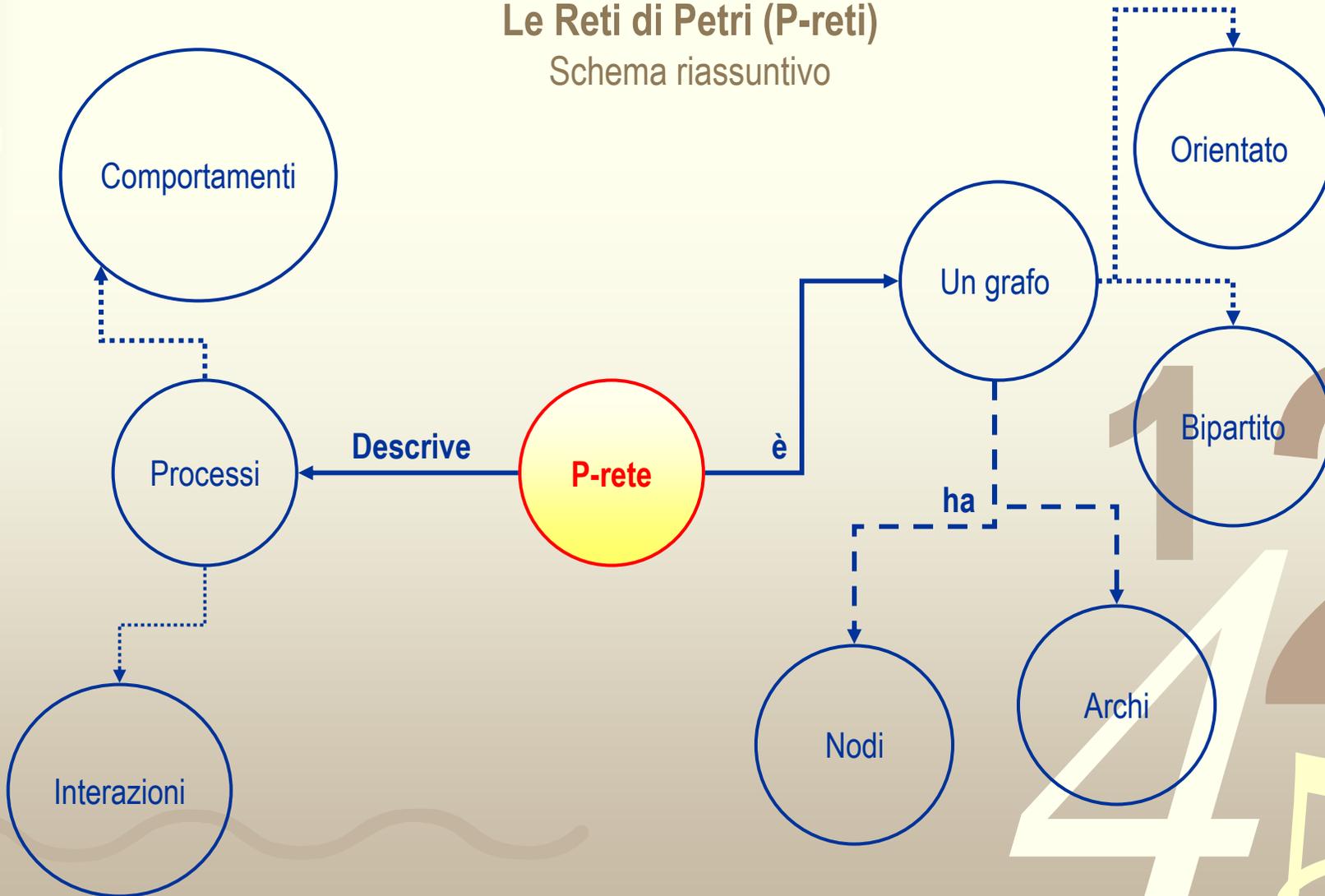
giorgio.poletti@unife.it – <http://docente.unife.it/giorgio.poletti>



Le Reti di Petri (P-reti)

Schema riassuntivo

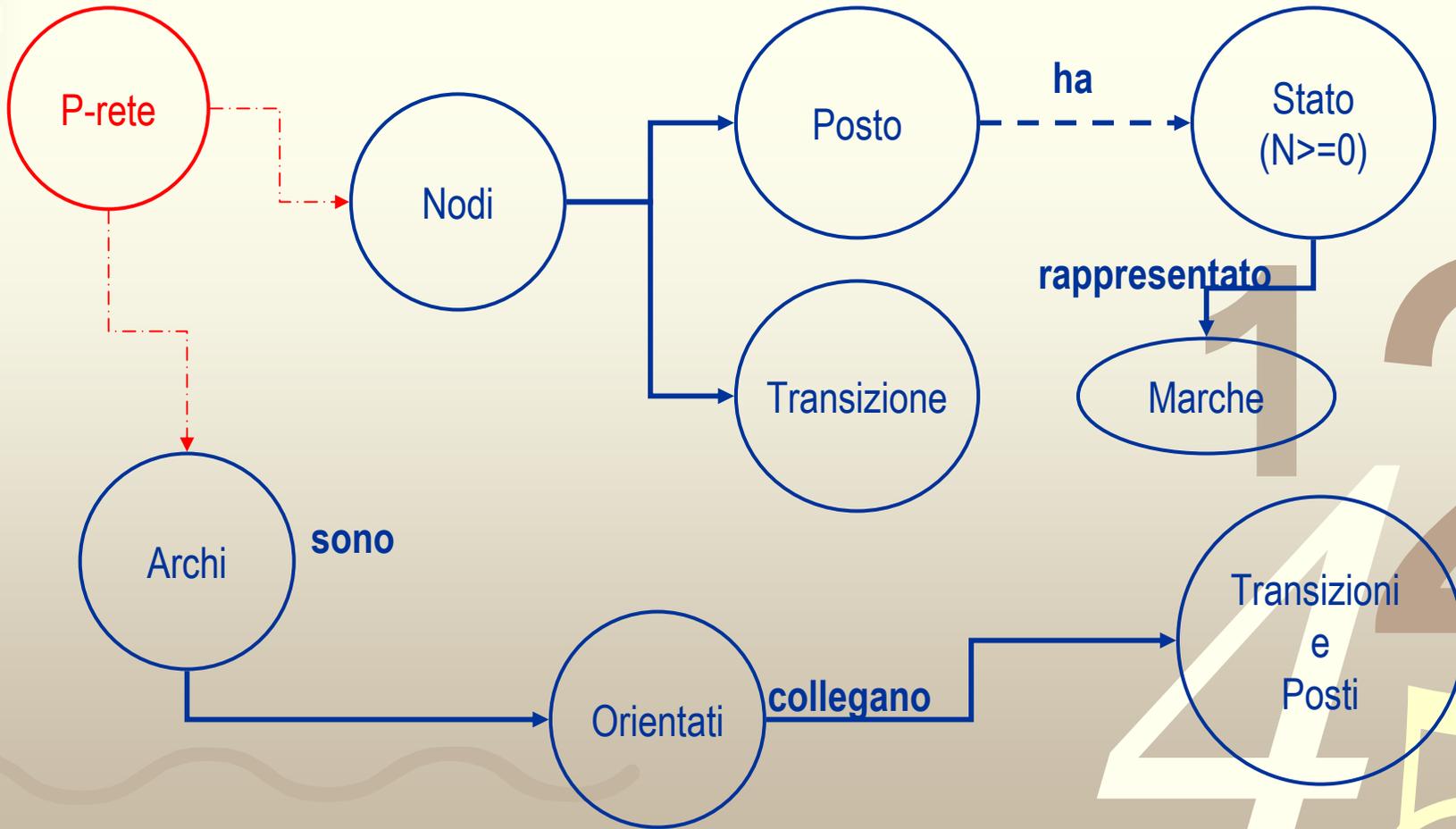
0011



Le Reti di Petri (P-reti)

Nodi e Archi

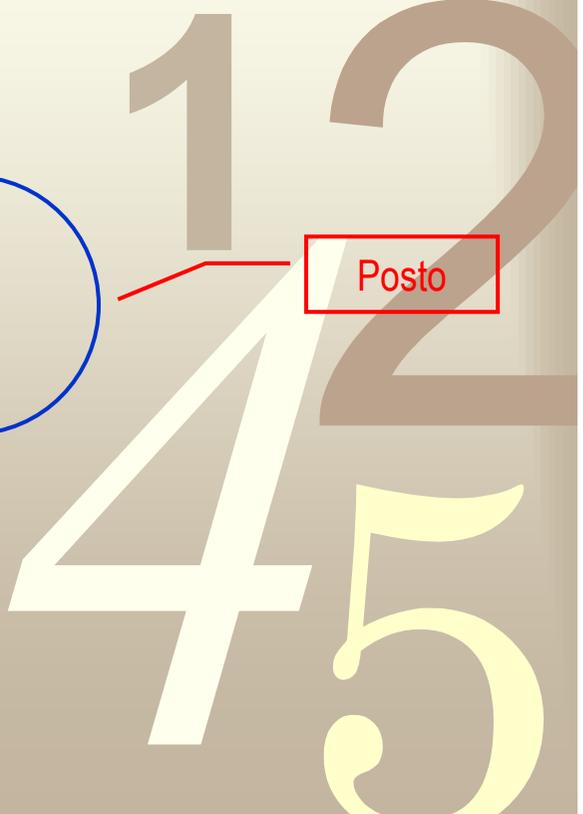
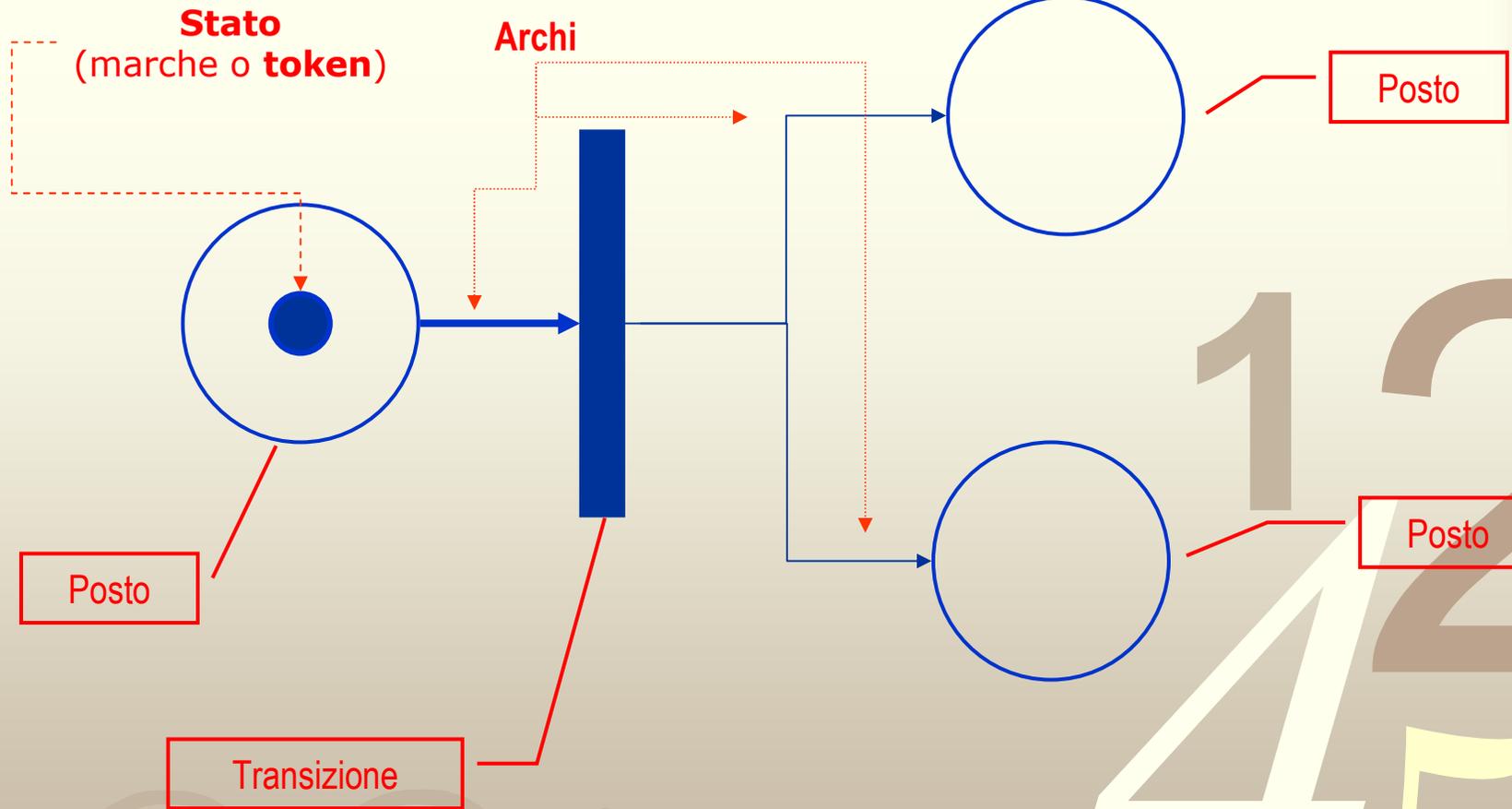
0011



Le Reti di Petri (P-reti)

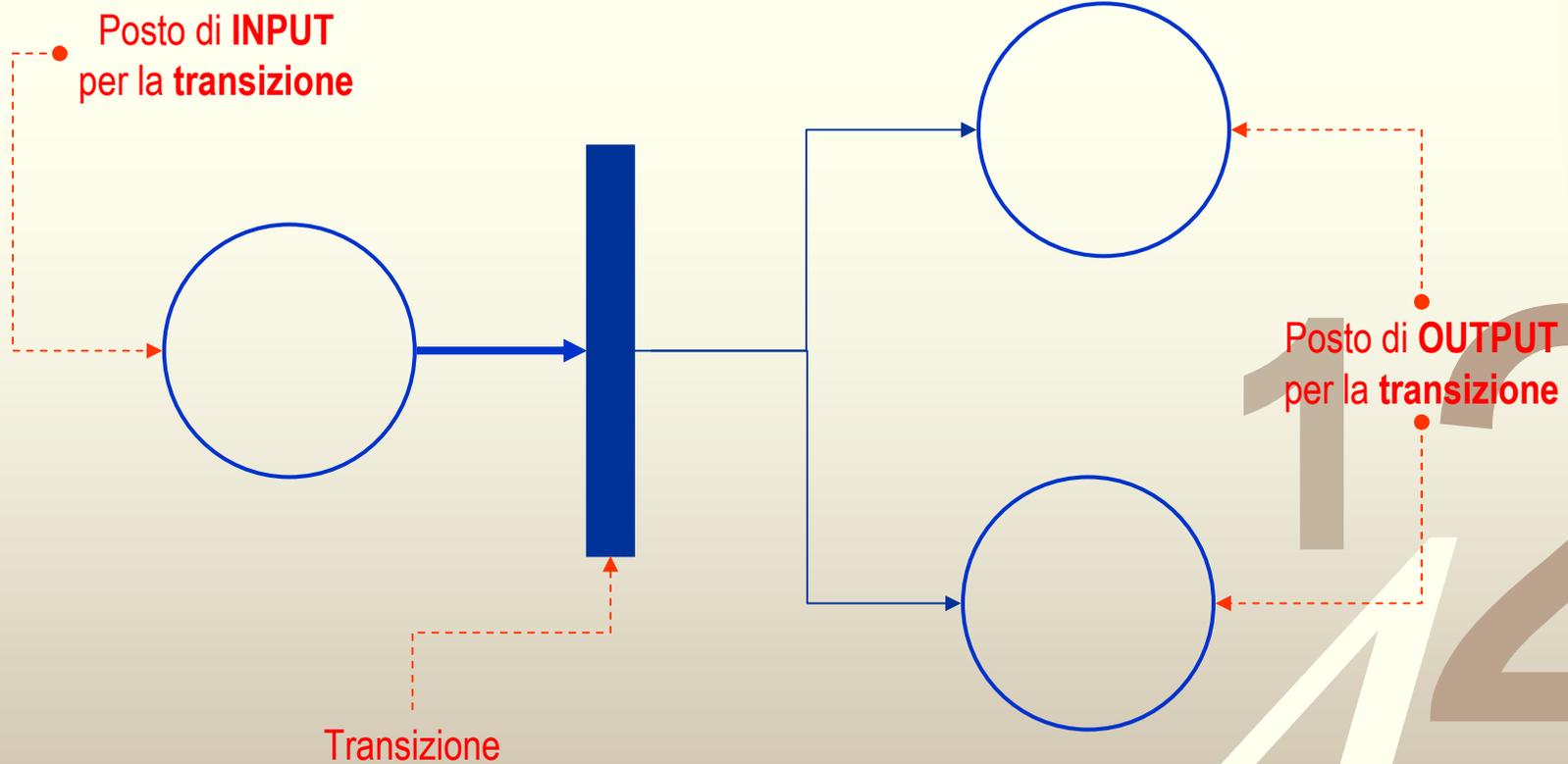
Modello di rappresentazione

0011



Le Reti di Petri (P-reti) Transizione: Input e Output

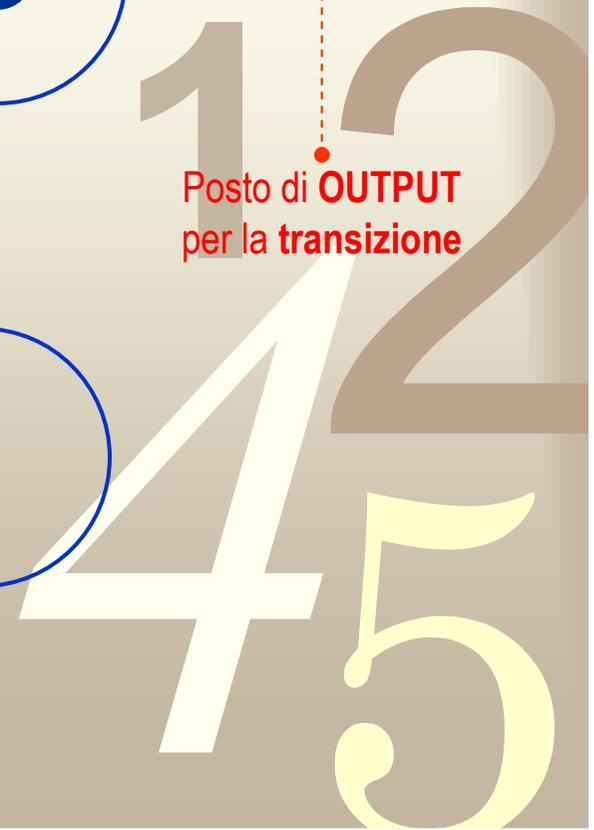
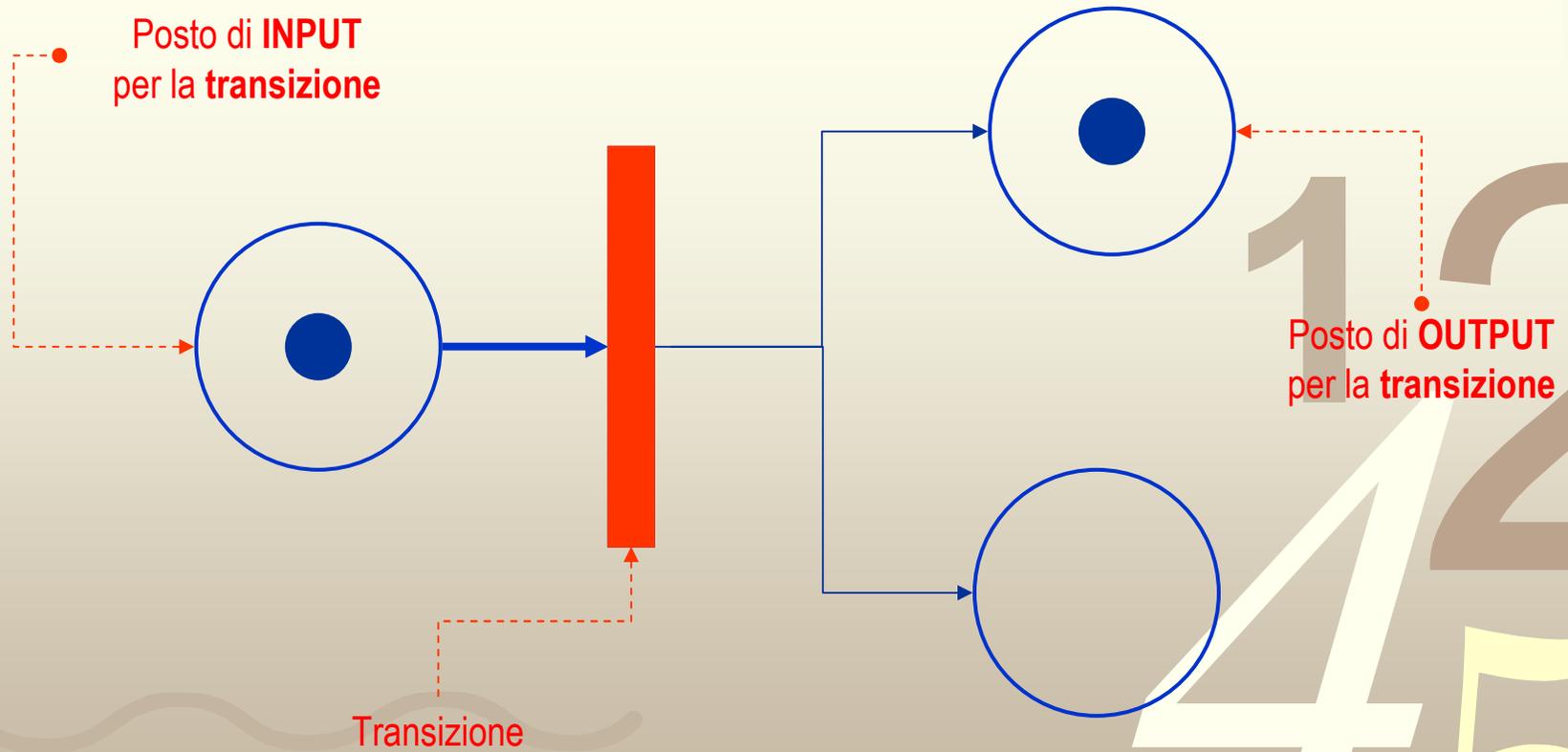
0011



Le Reti di Petri (P-reti)

Un esempio

0011

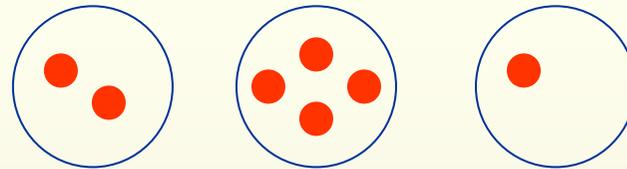


Le Reti di Petri (P-reti)

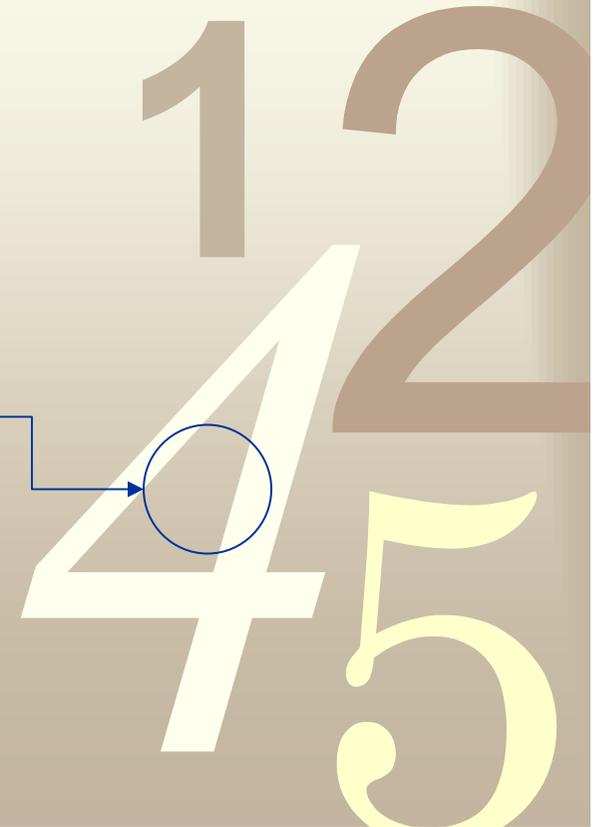
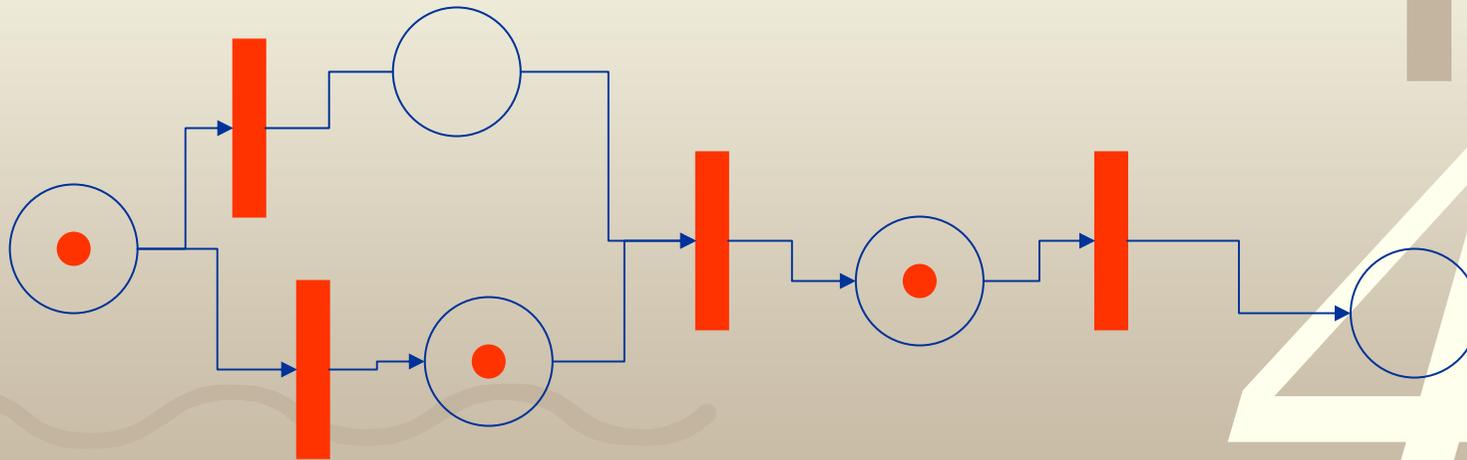
Alcune premesse

0011

I posti possono contenere un certo numero di token



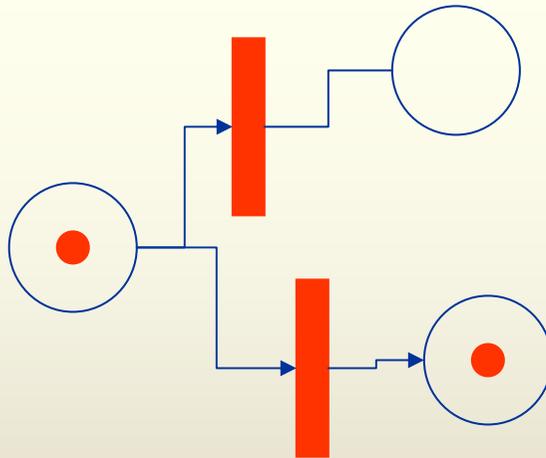
Marcatura è una distribuzione di token nei posti della rete



Le Reti di Petri (P-reti)

Alcune premesse

001 Le transizioni agiscono sui token in ingresso secondo una regola, detta **REGOLA di SCATTO** (*firing*)



Le transizioni si dicono **ABILITATE** se possono scattare

→ Ci sono tutti i token necessari nei posti di INPUT



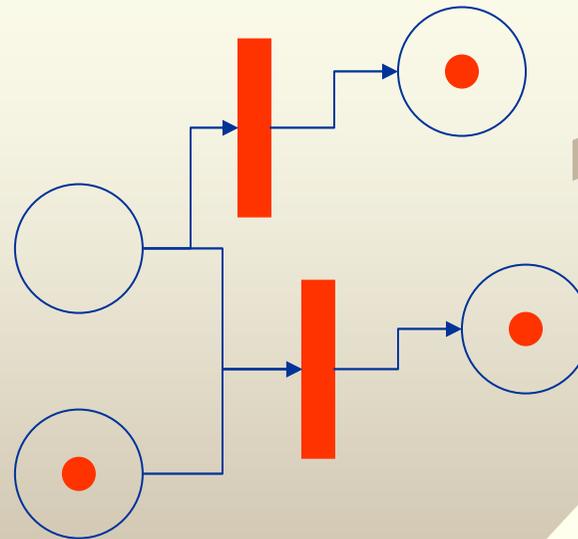
Le Reti di Petri (P-reti) Alcune premesse

0011

Le transizioni che scatta:

1. *Consuma* tutti i token presenti nei suoi posti di INPUT
2. *Esegue* i suoi compiti (TASK)
3. *Posiziona* nei posti di uscita i token

Processi automatici



Le reti di Petri (P-reti) sono reti **non-deterministiche**

1
2
4
5

Le Reti di Petri (P-reti)

Rete non-deterministica

0011

Rete non-deterministica

Se ci sono più transizioni abilitate, allo stesso tempo, qualsiasi può scattare

Non è garantito che una transizione abilitata scatti

Una transizione abilitata **può** scattare

Immediatamente

Dopo un periodo di attesa indefinito (purché permanga abilitata)

Mai



Le Reti di Petri (P-reti)

Rete non-deterministica

0011

Scatto di transizioni **non predicibile**

P-Reti

adatte per

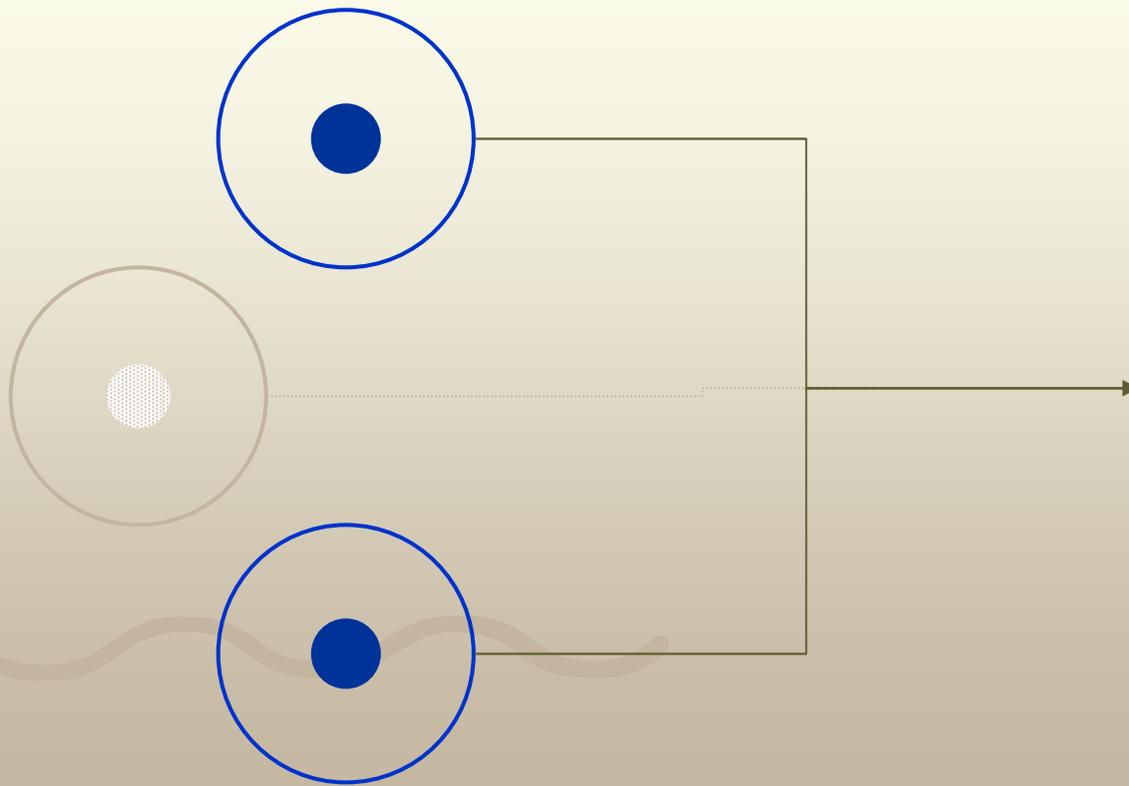
Modellare il comportamento di un sistema
CONCORRENTE DISTRIBUITO

In informatica sistemi che prevedano processi computazionali eseguiti contemporaneamente

Le Reti di Petri (P-reti) Concorrenza

0011

Si parla di **CONCORRENZA** in una **P-rete** quando sono possibili più di un input in una transizione



Le Reti di Petri (P-reti)

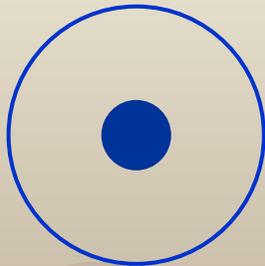
Conflitto

0011

Si parla di **CONFLITTO** in una **P-rete** quando è possibile avere più di una transizione come output

?

Dove deve andare il token di un posto? In quale transizione?



1

2

4

5

Le Reti di Petri (P-reti)

Le proprietà

0011

- **Raggiungibilità**
- **Limitatezza** (*Boundedness*)
- **Sicurezza** (*Safe*)
- **Vitalità** (*Liveness*)



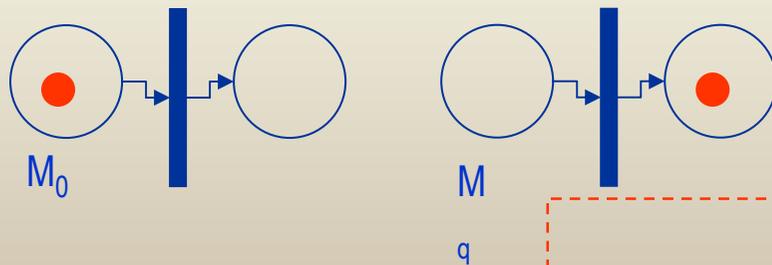
Le Reti di Petri (P-reti)

Raggiungibilità

00

Nota: data una marcatura iniziale M_0 di una rete G si indicano con $R(G, M_0)$ tutte gli stati raggiungibili dalla rete a partire da M_0

Una marcatura M_q è raggiungibile se esistono scatti che la rendono raggiungibile a partire da M_0



Problema della raggiungibilità:
 $M_q \in R(G, M_0)$

In quale situazioni M_q è uno stato sbagliato, non deve (quindi non può) essere raggiunto

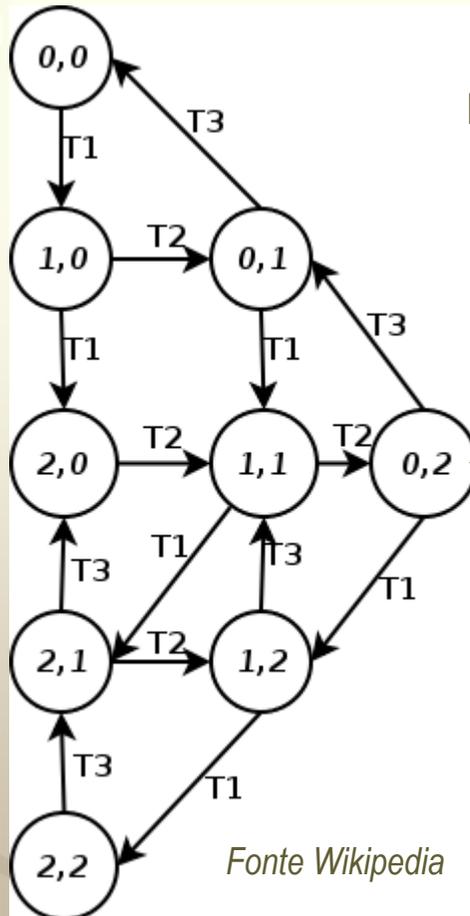
ad esempio

Porte aperte con l'ascensore non presente!!!!

Le Reti di Petri (P-reti)

Grafi di raggiungibilità

0011



Buon metodo per trovare gli stati "sbagliati"

La realtà presenta però troppo stati differenti

Utilizzo di metodi per dimostrare l'irraggiungibilità di alcuni stati

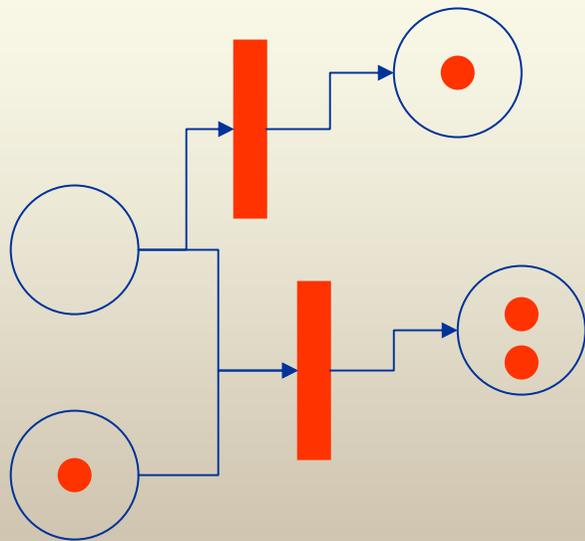
4
5

Le Reti di Petri (P-reti)

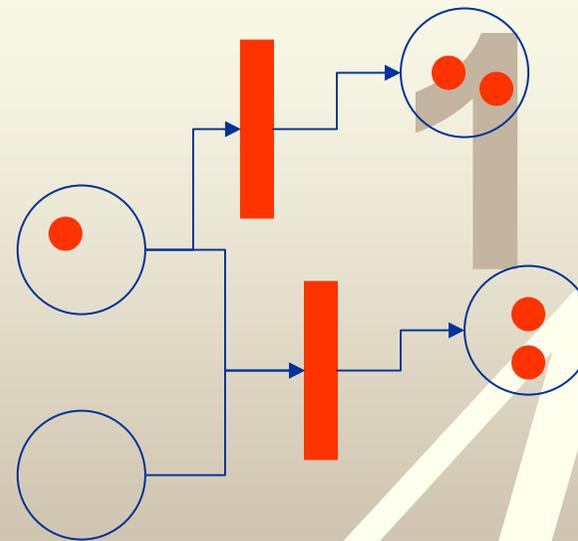
Limitatezza (Boundedness)

0011

- Un posto si dice k -limitato quando K è il numero massimo di token presenti in un posto per una qualsiasi marcatura possibile
- Una rete si dice LIMITATA se tutti i suoi posti sono limitati



M_0



M_q

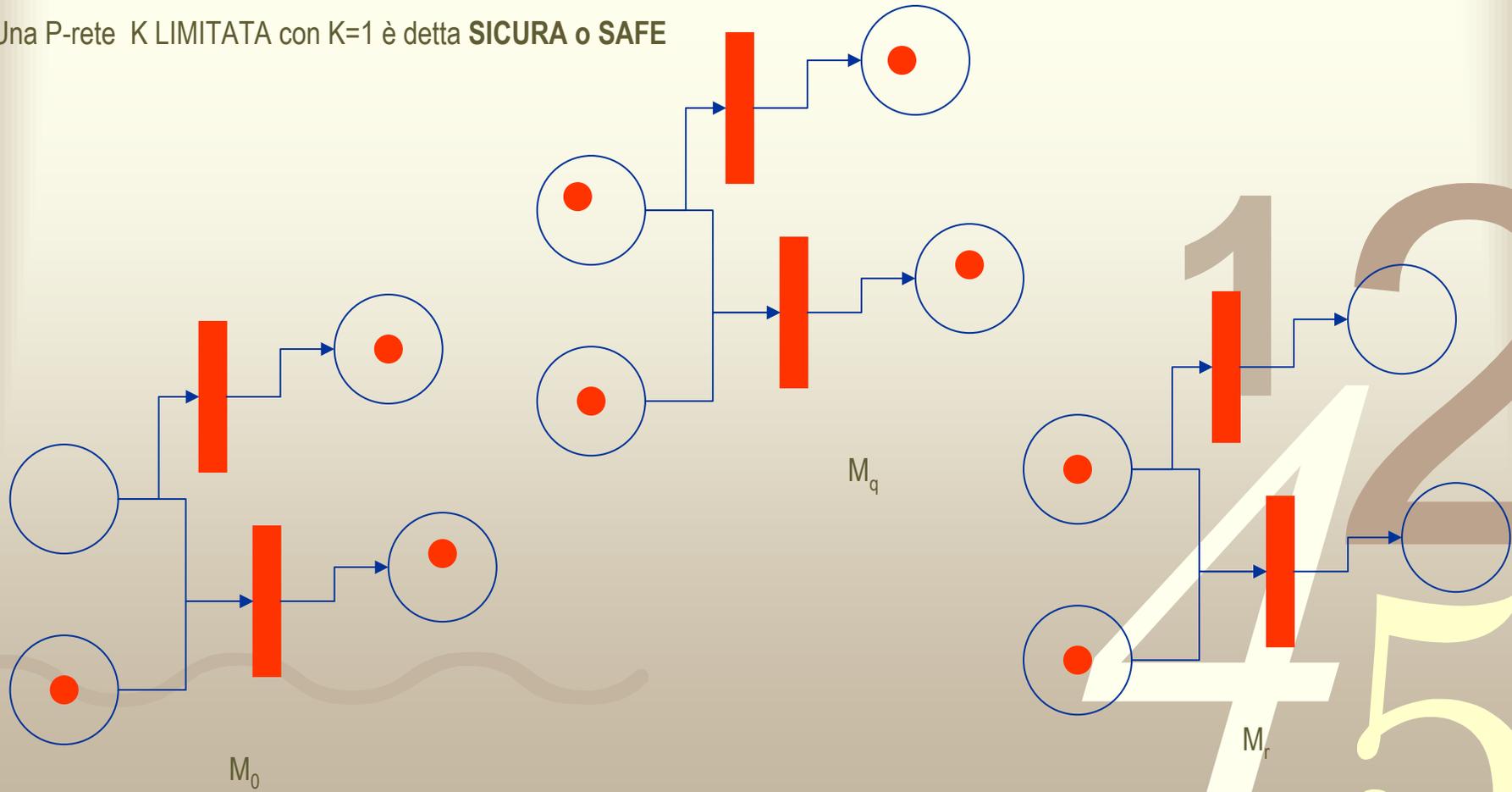
Esempio di P-rete 2-limitata

Le Reti di Petri (P-reti)

Sicurezza (Safe)

0011

Una P-rete K LIMITATA con $K=1$ è detta **SICURA** o **SAFE**



Esempio di P-rete 1-limitata

Le Reti di Petri (P-reti)

Vitalità (Liveness)

0011

La vitalità è una proprietà delle P-reti relativa alle transizioni. Si stabilisce il grado di vitalità dall'analisi dell'attivabilità di una transizione all'interno di una marcatura raggiungibile.

Una rete è viva se, qualunque sia la marcatura M_q raggiunta a partire da M_0 , da M_q è sempre possibile far scattare qualunque transizione (T) della rete a seguito di una ulteriore sequenza di scatti.

Se tutte le transizioni di una P-rete sono L_k Live la rete è L_k Live

Gradi di vitalità di una transizione T di una P-rete

Una transizione T di una P-rete è:

- **0 - L_0 Live** solo se T non può scattare in nessuna marcatura raggiungibile;
- **1 - L_1 Live** solo se esiste almeno una marcatura raggiungibile in cui T può scattare;
- **2 - L_2 Live** solo se per ogni numero intero K esiste almeno una marcatura raggiungibile tale per cui T può scattare K volte;
- **3 - L_3 Live** solo se esiste almeno una marcatura raggiungibile tale per cui T può scattare infinite volte;
- **4 - L_4 Live** solo se T può scattare in tutte le marcature raggiungibili;

0 Transizione morta

4 Transizione viva

Le Reti di Petri (P-reti)

Sistemi concorrenti distribuiti - 2 problemi di esempio

00 •“**Il problema dei filosofi affamati**” (dining philosophers problem, *Dijkstra*)
(problemi di concorrenza e condivisione di risorse)

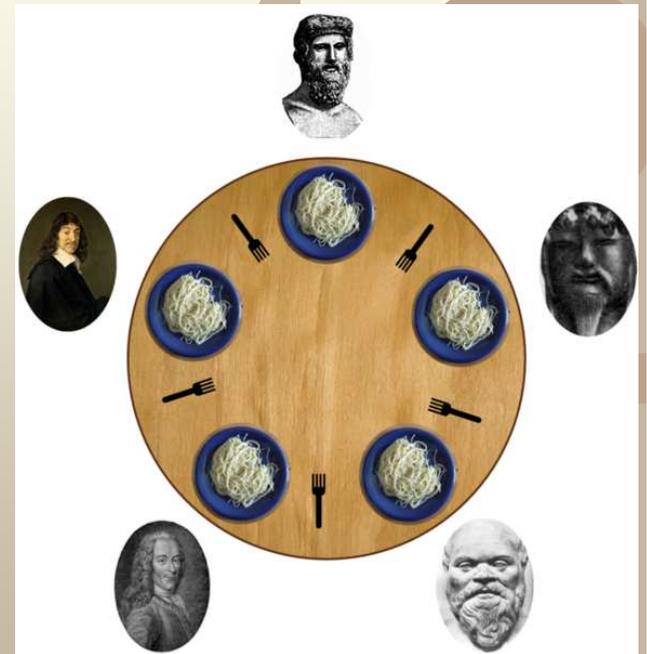
•“**Il problema del barbiere che dorme**”
(problemi analogo agli helpdesk informatizzati)

“Il problema dei filosofi affamati”

Alcuni filosofi (5 nel testo originale) sono seduti a tavola di fronte al loro piatto ed a due forchette (condivise con i loro vicini). I filosofi alternano momenti durante i quali meditare e momenti durante i quali mangiare. Per mangiare devono prendere le due forchette accanto al loro piatto e mangiare mentre durante la meditazione devono tenere le forchette sul tavolo. Risulta evidente che il numero di forchette impedisce a tutti i filosofi di mangiare contemporaneamente.

Una corretta programmazione concorrente deve essere in grado di far mangiare alternativamente tutti i filosofi evitando che qualcuno in particolare soffra di **starvation**¹ ed evitando che si verifichino stalli in fase di "acquisizione delle forchette".

(Fonte Wikipedia)



Starvation, letteralmente “inedia”, ma in informatica lo stato di un processo pronto per essere eseguito ma che non riesce ad ottenere le risorse di cui necessita.

Le Reti di Petri (P-reti)

Sistemi concorrenti distribuiti - 2 problemi di esempio

0011

- **“Il problema dei filosofi affamati”** (dining philosophers problem, *Dijkstra*)
(problemi di concorrenza e condivisione di risorse)
- **“Il problema del barbiere che dorme”**
(problemi analogo agli helpdesk informatizzati)

“Il problema del barbiere che dorme”

Alcuni un barbiere possiede un negozio con una sola sedia da lavoro e un certo numero limitato di posti per attendere. Se non ci sono clienti il barbiere dorme altrimenti, all'arrivo del primo cliente il barbiere si sveglia ed inizia a servirlo. Se dovessero sopraggiungere clienti durante il periodo di attività del barbiere, essi si mettono in attesa sui posti disponibili. Al termine dei posti di attesa, un ulteriore cliente viene scartato. Questa problematica è molto vicina al sistema di funzionamento degli helpdesk informatizzati dove l'operatore serve, uno per volta, tutti i clienti in coda oppure attende, senza effettuare alcuna operazione in particolare, l'arrivo di nuove chiamate.

Una corretta programmazione concorrente deve far "dormire" il barbiere in assenza di clienti, attivare il barbiere sul primo cliente al suo arrivo e mettere in coda tutti i successivi clienti tenendoli inattivi.

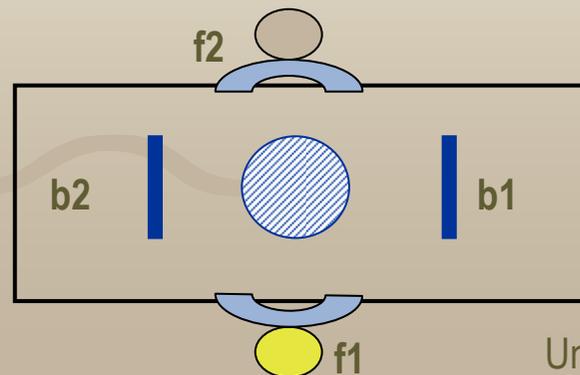
(Fonte Wikipedia)

Le Reti di Petri (P-reti)

Il problema dei 2 filosofi affamati

0011

- *Nietzsche ed Eraclito mangiano spesso assieme.*
- *Essi si siedono attorno ad un tavolo rotondo e hanno, ognuno, a disposizione un piatto di cibo e due singole bacchette sono collocate ai lati dei loro piatti.*
- *Essi, sempre, o pensano o mangiano.*
- *Quando uno dei due comincia ad avere fame cerca di prendere possesso delle bacchette alla sua destra e alla sua sinistra, in ordine arbitrario.*
- *Qualora riesca a prenderle entrambe, mangia per un po'. Successivamente depone le bacchette e si rimette a pensare.*
- *Si sottolinea che nessuno dei due è in grado di mangiare con una singola bacchetta o con le mani.*
- *Il problema consiste nel far in modo che entrambi i filosofi riescano a cibarsi periodicamente.*



Un possibile schema

