

INE 5410

PROGRAMAÇÃO CONCORRENTE

Prof. Vitorio Bruno Mazzola
mazzola@inf.ufsc.br



Capítulo 4 REDES DE PETRI



Introdução

- Engenharia de Software

Engenharia de Requisitos

Projeto

Implementação

Testes

Introdução

- Métodos Formais

– Necessidade de se ter técnicas de representação de aspectos inerentes a sistemas distribuídos que envolvem aspectos de concorrência

- Paralelismo
- Concorrência
- Não determinismo
- Comunicação
- Temporização
- Aspectos de arquitetura



Introdução

- Métodos Formais

- Vantagem dos métodos formais é permitir não apenas que se faça “síntese” de problemas e de suas soluções
- Possibilidade de se poder realizar a “análise” e obter resultados relevantes sobre a operação do sistema, antes de ter atingido a fase de implementação é um dos aspectos mais atraentes da utilização dos métodos formais

Introdução

- Métodos Formais

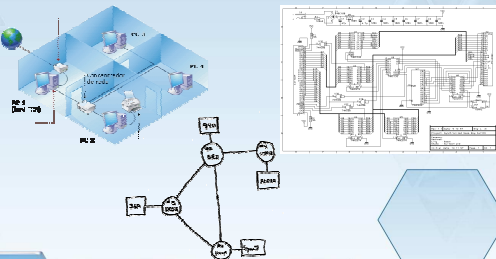
- Importância da Modelagem

- A modelagem é sempre uma aproximação da realidade... um modelo nunca reproduz fielmente o sistema real
- Razões são as mais diversas:
 - Falta de conhecimento do sistema real
 - Limitações da técnica de representação
 - Detalhes sobre o sistema são irrelevantes para o tipo de estudo ou desenvolvimento

Introdução

- Métodos Formais

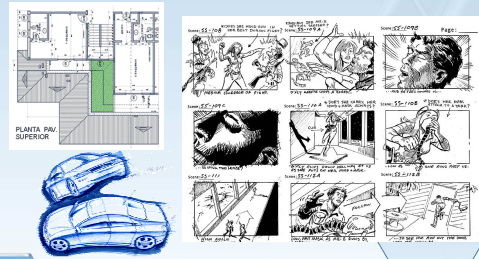
- Modelagem é adotada em muitas áreas



Introdução

- Métodos Formais

- Modelagem é adotada em muitas áreas



Introdução

- Métodos Formais

- Estados e transições

- Estados representam "patamares" de operação de um dado sistema
 - Conjuntos de valores de diferentes variáveis de um sistema podem compor um único estado
 - **Por exemplo:** o estado "ligado" de um televisor pode agrupar diferentes níveis de volume, cor, brilho, contraste e canais sendo recebidos

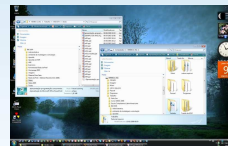
Introdução

- Métodos Formais

- Estados e transições

- Estados possuem uma duração significativa na evolução do comportamento de um sistema

- **Por exemplo:**



Introdução

- Métodos Formais

- Estados e transições

- Transições representam eventos que podem causar alterações no estado de um sistema



Clique num botão do mouse

Término de um processo químico



Ultrapassagem do valor de uma grandeza física

Introdução

- Origem das Redes de Petri

- Desenvolvidas por Carl Adam Petri em 1962 para permitir a modelagem de sistemas a eventos discretos
 - Aplicação-alvo eram os sistemas automatizados para manufatura
 - Potencial das RdP foi motivando a aplicação a outros tipos de sistemas

- Protocolos de comunicação
 - Algoritmos distribuídos
 - Sistemas químicos



Apresentação das RdP

- O que são as Redes de Petri

- Método para modelagem que permite avaliar sistemas a eventos discretos para fins de aprendizagem ou desenvolvimento
- É qualificado como um método formal, uma vez que, além da sintaxe, existe um formalismo matemático sobre o qual se apóia a semântica de funcionamento das RdP
- É possível simular ou avaliar propriedades de sistemas cujos modelos tenham sido realizados em RdP

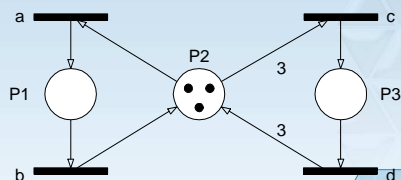
Apresentação das RdP

- Elementos da sintaxe de uma RdP

- **Lugar**, que representa uma pré-condição e/ou pós-condição de eventos de um sistema
- **Transição**, que representa os eventos relevantes de um sistema
- **Arcos**, que permitem relacionar lugares e transições
- **Fichas**, que indicam o estado corrente de um modelo construído em Rede de Petri, informando sobre a possibilidade de ocorrência de um ou mais eventos do sistema

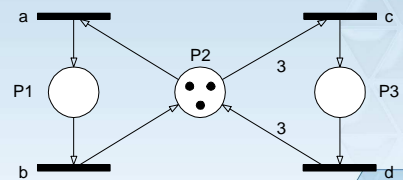
Apresentação das RdP

- Elementos da sintaxe de uma RdP



Apresentação das RdP

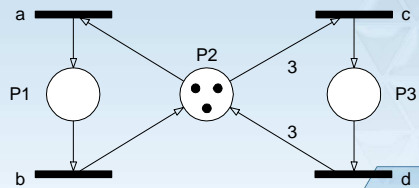
- Lugares de entrada e lugares de saída de uma transição



Lugar de entrada da transição a: P2
Lugar de saída da transição a: P1

Apresentação das RdP

- Lugares de entrada e lugares de saída de uma transição

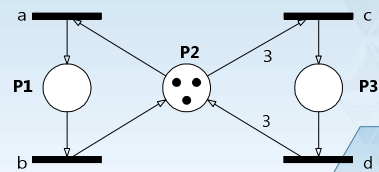


Lugar de entrada da transição d: P3
Lugar de saída da transição d: P2

Apresentação das RdP

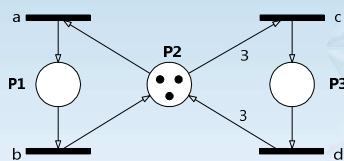
Grafo e Notações Matriciais

$P = \{p1, p2, p3\}$
 $T = \{a, b, c, d\}$



Apresentação das RdP

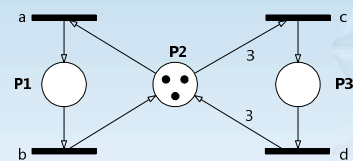
Grafo e Notações Matriciais



$$\text{Pre} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Post} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Apresentação das RdP

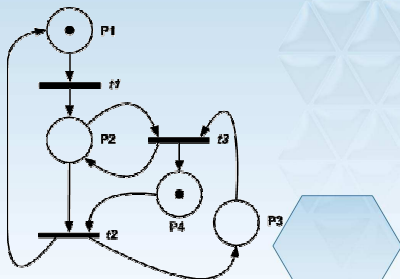
Grafo e Notações Matriciais



$$C = \begin{bmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & -3 & 3 \\ 0 & 0 & 0 & -1 \end{bmatrix} \quad M = \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix}$$

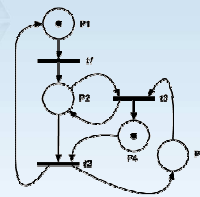
Apresentação das RdP

- Definir a representação matricial para as Redes de Petri a seguir...



Apresentação das RdP

- Definir a representação matricial para as Redes de Petri a seguir...



$$\text{Pre} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\text{Post} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$M0 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Funcionamento das RdP

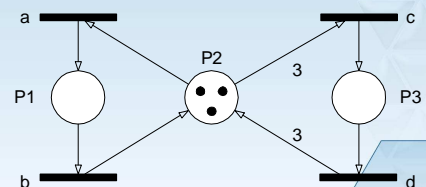
- Regras de Disparo das transições de uma RdP**

- Para que uma transição possa disparar, ela precisa estar habilitada
- Uma transição habilitada é aquela em que todas as pré-condições são verificadas, ou seja, para cada lugar de entrada da transição, deve-se verificar uma quantidade de fichas igual ou superior ao peso do arco que conecta o lugar à transição
- Num dado momento, diversas transições podem estar habilitadas numa dada RdP

Funcionamento das RdP

- Regras de Disparo**

- Existem transições habilitadas? Qual ou quais?



Funcionamento das RdP

• Regras de Disparo das transições de uma RdP

- Uma transição habilitada pode disparar ou não
- Em Redes de Petri, apenas uma transição pode disparar a cada vez
- Duas ou mais transições habilitadas numa RdP sugerem a existência de paralelismo ou conflito entre eventos do sistema
- O disparo de uma dada transição causa alterações no posicionamento das fichas nos lugares, segundo uma regra bem definida

Funcionamento das RdP

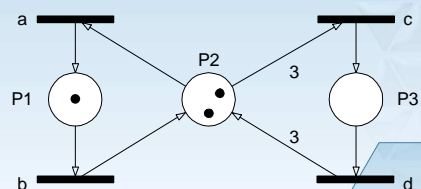
• Regras de Disparo das transições de uma RDP

- O disparo de uma transição é caracterizado pelas seguintes ocorrências em relação ao posicionamento de fichas nos lugares:
 - De cada lugar de entrada da transição, são retiradas tantas fichas quanto seja o peso do arco que o conecta à transição
 - Em cada lugar de saída, são depositadas tantas fichas quanto seja o peso do arco que o conecta à transição

Funcionamento das RdP

• Regras de Disparo

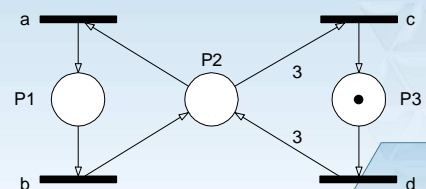
- Disparo da transição **a**



Funcionamento das RdP

• Regras de Disparo

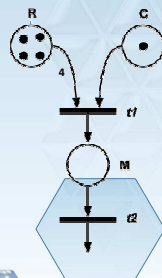
- Disparo da transição **b**



Funcionamento das RdP

• Regras de Disparo

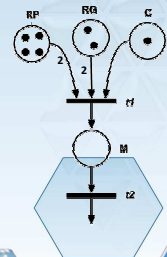
- Significado dos pesos dos arcos



Funcionamento das RdP

• Regras de Disparo

- Significado dos pesos dos arcos



Funcionamento das RdP

• Grafo de Marcação

- Marcação é o termo utilizado para designar o posicionamento das fichas nos lugares da Rede de Petri
- O disparo de qualquer transição numa Rede de Petri pode resultar em modificação na sua marcação
- Exemplo: observar a mudança na marcação da Rede de Petri do slide anterior para os disparos de *a* e de *b*

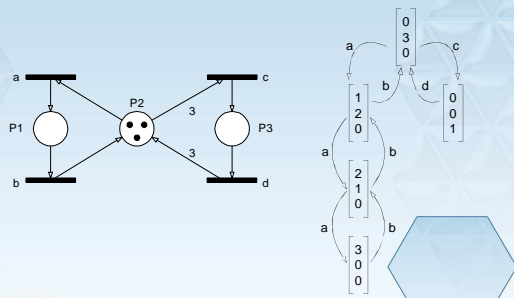
Funcionamento das RdP

• Grafo de Marcação

- À configuração inicial das fichas numa Rede de Petri chamamos de **marcação inicial**
- Cada **marcação** de uma Rede de Petri corresponde a um estado de operação da mesma
- Um **grafo de marcações** permite visualizar todas as marcações que podem ser atingidas para uma dada Rede de Petri, a partir da marcação inicial, considerando todos os disparos possíveis de cada transição

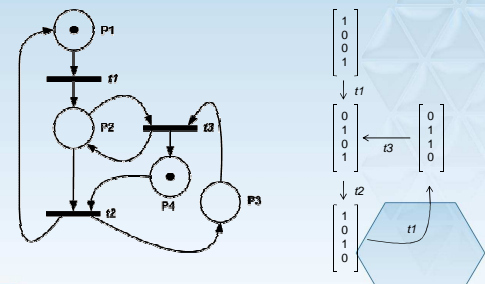
Funcionamento das RdP

• Grafo de Marcação - Exemplo



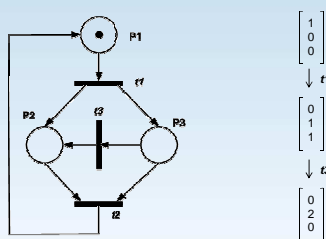
Funcionamento das RdP

• Grafo de Marcação - Exercício



Funcionamento das RdP

• Grafo de Marcação - Exercício



Análise das Redes de Petri

• Propriedades dependentes da marcação

- O estabelecimento de propriedades a serem verificadas numa Rede de Petri é um aspecto fundamental para a análise de um sistema
- As propriedades dependentes da marcação são assim denominadas porque, para uma mesma RdP, dependendo da marcação inicial, podem ou não ser verificadas

- Limitação
- Vivacidade
- Reinicialização

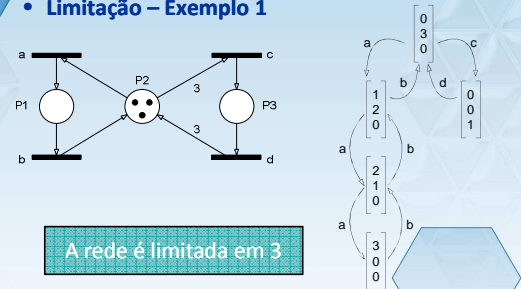
Análise das Redes de Petri

• Limitação

- A propriedade da limitação está associada aos lugares
- Um lugar é dito ***k-limitado*** se para todas as marcações possíveis da RdP, o número de fichas no lugar será, no máximo, igual a ***k***
- Uma Rede de Petri é ***k-limitada*** (ou ***limitada em k***) se todos os seus lugares são ***k-limitados***

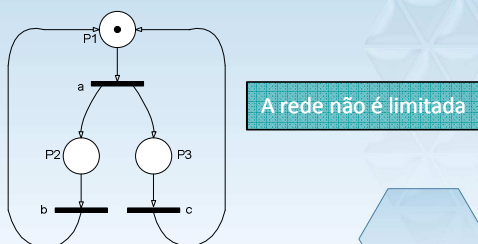
Análise das Redes de Petri

• Limitação – Exemplo 1



Análise das Redes de Petri

• Limitação – Exemplo 2



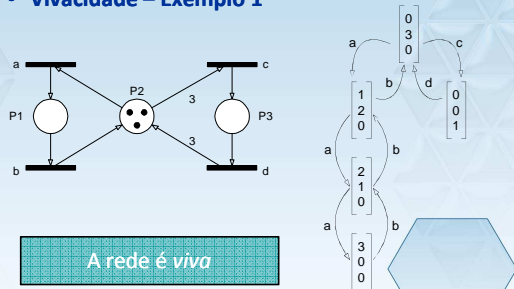
Análise das Redes de Petri

• Vivacidade

- Uma transição para a qual existe pelo menos um caminho a partir da marcação inicial que leve a seu disparo é dita ***quase-viva***
- Uma transição para a qual existe pelo menos um caminho a partir de qualquer marcação que leve a seu disparo é dita ***viva***
- Uma Rede de Petri é ***viva*** se todas as suas transições são ***vivas***

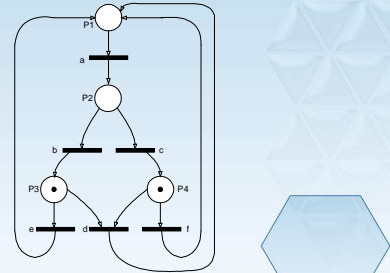
Análise das Redes de Petri

• Vivacidade – Exemplo 1



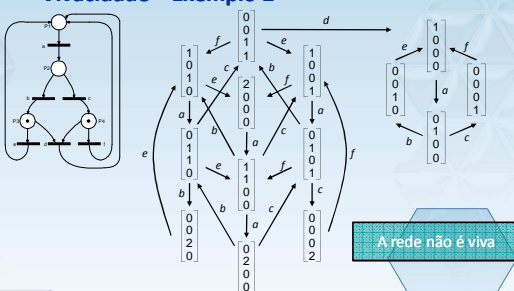
Análise das Redes de Petri

• Vivacidade – Exemplo 2



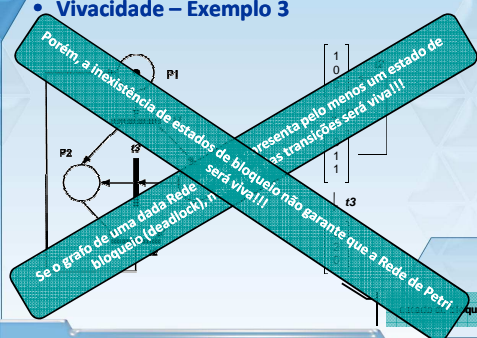
Análise das Redes de Petri

• Vivacidade – Exemplo 2



Análise das Redes de Petri

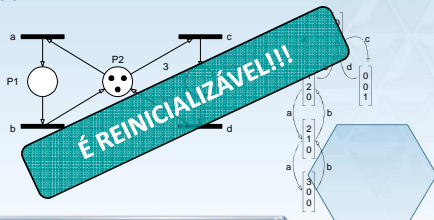
• Vivacidade – Exemplo 3



Análise das Redes de Petri

• Reinicialização

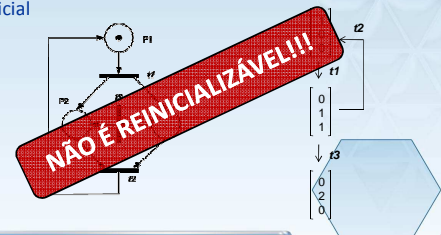
- Uma Rede de Petri é **reinicializável** se, de qualquer marcação existe um caminho que leve à marcação inicial



Análise das Redes de Petri

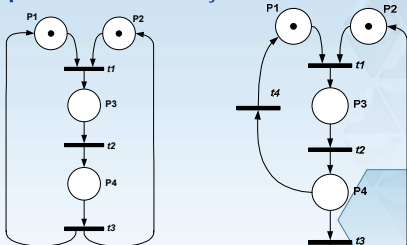
• Reinicialização

- Uma Rede de Petri é **reinicializável** se, de qualquer marcação existe um caminho que leve à marcação inicial

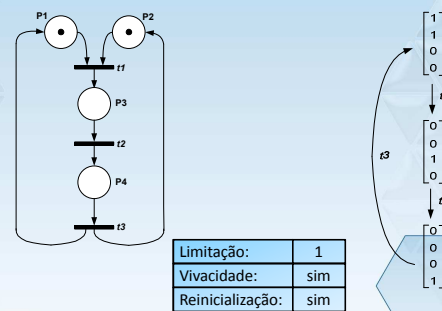


Exercício

- Desenvolver o grafo de marcações e analisar os modelos do ponto de vista das propriedades dependentes da marcação ...

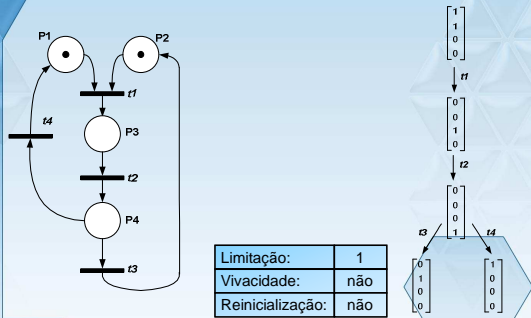


Exercício



Limitação:	1
Vivacidade:	sim
Reinicialização:	sim

Exercício



Redes de Petri Interpretadas

• Lugares, Fichas e Transições

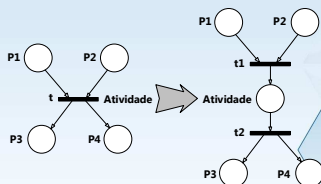
- Os elementos de um modelo a base de Redes de Petri são definidos de modo a representar algum aspecto concreto do sistema
- O significado específico vai depender da natureza do modelo sendo construído, mas é possível estabelecer correlações genéricas destes elementos com a realidade dos sistemas

- **LUGARES:** atividades de um sistema a eventos discretos
- **TRANSIÇÕES:** eventos "instantâneos"
- **FICHAS:** objetos físicos ou lógicos

Redes de Petri Interpretadas

• Transições

- Transições podem ser consideradas uma simplificação de um trecho de Rede de Petri, caracterizado por uma transição de início, um lugar que representa a atividade e uma transição de fim da atividade



Redes de Petri Interpretadas

• Controle, Dados e Ambiente

- Um Sistema Aberto é um sistema que se comunica frequentemente com o ambiente
- Para representação de um sistema aberto, a Rede de Petri deve ser estruturada em duas partes, uma que representa o **sistema** e outra que corresponde ao **ambiente**
- Ainda, deve-se tomar o cuidado de explicitar as partes da Rede que correspondem ao controle do sistema e aquelas relacionadas aos dados

Redes de Petri Interpretadas

- **Representação do Tempo**

- É um aspecto importante para expressar requisitos e restrições associados ao tempo nos sistemas
- A expressão de aspectos temporais pode estar relacionada aos lugares ou às transições
- Foram definidos novos modelos de Redes de Petri para levar em conta estes aspectos

Redes de Petri Interpretadas

- **Análise**

- Uma Rede de Petri Ordinária descreve apenas a parte “controle” de um sistema
- Numa Rede Interpretada, o estado deve também levar em conta os dados do sistema
 - **Estado do Sistema = Marcação + Estado dos Dados**
 - **Estado dos Dados = Variáveis Internas + Tempo**

Redes de Petri Interpretadas

- **Análise**

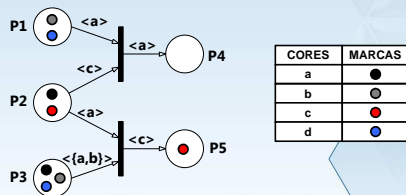
- Conjunto das Marcações de uma **Rede de Petri Interpretada** é um subconjunto das Marcações de uma **Rede de Petri Ordinária**
- Para que uma transição possa ser disparada numa **Rede de Petri Interpretada** é preciso que ela possa também ser disparada na **Rede de Petri Ordinária**
- Por outro lado, uma transição que pode ser disparada numa **Rede de Petri Ordinária** pode não ser disparada num dado instante na **Rede de Petri Interpretada** devido às restrições impostas no modelo

Redes de Petri Coloridas

- **Objetivo principal:** reduzir o tamanho do modelo, permitindo que marcas individualizadas (cores) representem diferentes processos ou recursos em uma mesma sub-rede.
- Inicialmente as marcas eram representadas por cores (ou padrões) que possibilitassem a distinção das marcas. Em trabalhos mais recentes (a partir de 1990) as marcas são representadas por tipos de dados complexos.

Redes de Petri Coloridas

- **Arcos:** rotulados com uma ou mais das possíveis cores ou ainda com variáveis.
- **Habilitação da transição:** os lugares de entrada devem ter marcas da cor associada ao arco que interliga esses lugares a transição.



Redes de Petri Coloridas

• Características

- A separação entre dados e controle num sistema nem sempre é evidente
- Um sistema onde a parte de controle é excessivamente rica pode conduzir a uma Rede de Petri com estrutura demasiadamente complexa
- A replicação de processos pode ser uma estratégia interessante (representar conjuntos de processos que possuam estruturas de controle idênticas)

Redes de Petri Coloridas

• Associando Cores

- A associação de “cores” às fichas é uma estratégia para a distinção das fichas numa Rede de Petri
- As “cores” são representadas por números inteiros ou por conjuntos de etiquetas
- A cada lugar associa-se um subconjunto de “cores” que podem ser posicionadas nele
- A cada transição associa-se um conjunto de cores que correspondam a um possível disparo

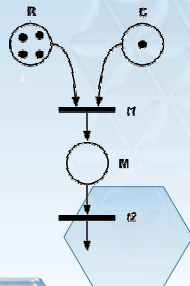
Redes de Petri Coloridas

• Associando Cores

- No caso da replicação de processos com estruturas idênticas, o conjunto de cores será idêntico para todos os lugares e transições
- As “cores” estarão indicando as diferentes instâncias de um processo
- No caso de alguns lugares serem comuns a diferentes processos, pode ser necessário introduzir o conceito de “cores compostas”

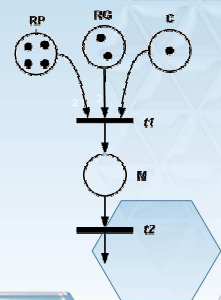
Redes de Petri Coloridas

- Exemplo: Montagem de um carro



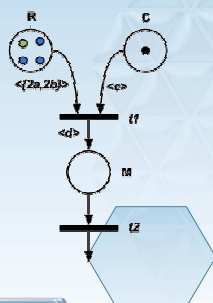
Redes de Petri Coloridas

- Exemplo: Montagem de um buggy



Redes de Petri Coloridas

- Exemplo: Montagem de um dragster



Cores	Marcas	Significado
a		Rodas grandes
b		Rodas pequenas
c		Chassi
d		Buggy

Redes de Petri Coloridas

Elementos

– As **RdP Coloridas** oferecem um maior poder de expressão, permitindo representar escolhas determinísticas... Para isso, contam com os seguintes elementos:

- Estrutura
- Declarações
- Inscrições

Redes de Petri Coloridas

- **Elementos / Estrutura**

- Corresponde à forma como os lugares, transições e arcos são dispostos para representar o sistema, consistindo de um grafo orientado que possui como vértices os lugares e as transições



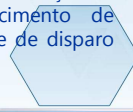
Redes de Petri Coloridas

- **Elementos / Declaração**

- A parte de declaração consiste da descrição dos elementos, em particular, dos conjuntos de cores, assim como das variáveis que serão utilizadas no modelo

- **Elementos / Inscrições**

- Podem ser associadas a lugares, transições e arcos e permitem o estabelecimento de condições iniciais, de habilitação e de disparo das transições da RdP.



Redes de Petri Coloridas

- **Inscrições Lugares**

- Nomes
- Conjuntos de cores
- Expressões de inicialização

- **Inscrições Transições**

- Nomes
- Expressões guarda

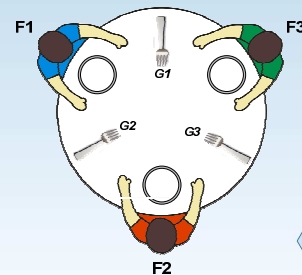
- **Inscrições Arcos**

- Expressão



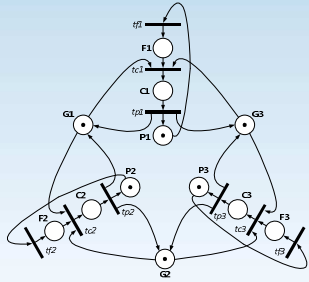
Redes de Petri Coloridas

- **Replicação de Processos (exemplo)**



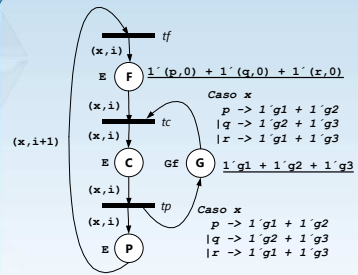
Redes de Petri Coloridas

• Replicação de Processos (exemplo)



Redes de Petri Coloridas

• Replicação de Processos (exemplo)



Redes de Petri Temporizadas

• Considerações

- Os Modelos de Redes de Petri estudados estabelecem uma relação de causalidade aos eventos
- Se um evento **a** é considerado ser causa de um evento **b**, então **a** precede **b** e, portanto, **a** e **b** estão ordenados no tempo
- Estes modelos não permitem expressar, porém, o tempo de forma quantitativa

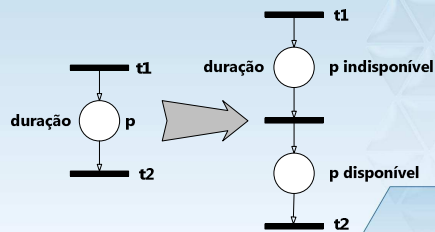
Redes de Petri Temporizadas

• Associação de Tempo aos Lugares

- Em praticamente qualquer modelo de Redes de Petri, os lugares representam atividades
- Associar tempo a um lugar corresponde a descrever a duração de uma atividade
- Um lugar com duração associada pode ser descrito por uma sequência **lugar-transição-lugar**, onde o primeiro lugar corresponde à atividade em curso, a transição representa o evento passagem do tempo e o segundo lugar corresponde a uma espera pelo final da atividade

Redes de Petri Temporizadas

• Associação de Tempo aos Lugares



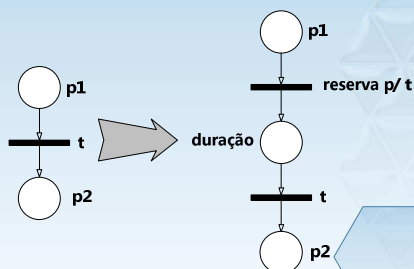
Redes de Petri Temporizadas

• Associação de Tempo às Transições

- Quando se reflete em associar um tempo a uma transição, esta não representa mais um evento instantâneo
- Uma transição com duração associada pode ser descrita por uma sequência **transição-lugar-transição**, onde a primeira transição corresponde a um evento de início de atividade, o lugar representa a atividade em curso e a última transição corresponde ao final da atividade

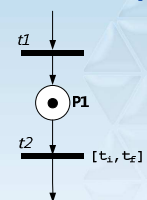
Redes de Petri Temporizadas

• Associação de Tempo às Transições



Redes de Petri Temporais

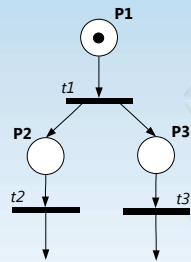
• Associação de um intervalo de tempo às transições



- t_i - define o instante de tempo mínimo (a contar da habilitação da transição, antes que ela possa disparar)
- t_f - define o instante de tempo máximo que a transição deve estar habilitada para que possa disparar

Redes de Petri Temporais

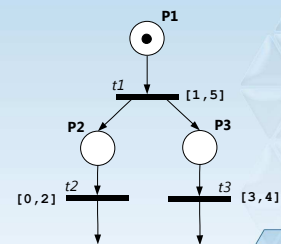
• Exemplo:



- Disparos possíveis: $t1 \rightarrow t2 \rightarrow t3$ ou $t1 \rightarrow t3 \rightarrow t2$

Redes de Petri Temporais

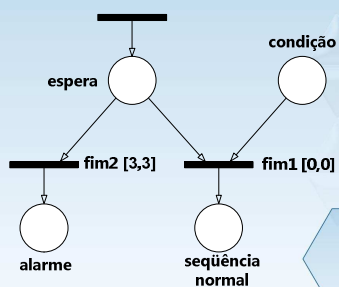
• Exemplo:



- Disparos possíveis: $t1 \rightarrow t2 \rightarrow t3$

Redes de Petri Temporais

• Timeouts



Ferramenta de Suporte a Redes de Petri

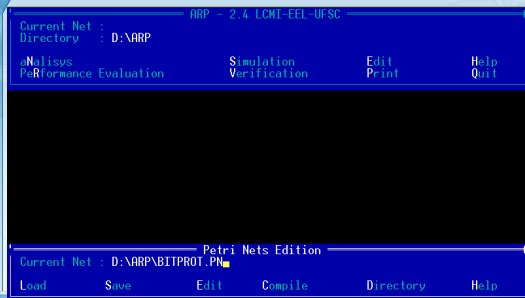
• ARP — Analisador de Redes de Petri

— Funções Possíveis

- Edição da Rede de Petri (em arquivo texto)
- Análise de Invariantes de Lugar e de Transição
- Construção do Grafo de Marcação
- Redução do Grafo por Equivalência de Traço
- Simulação
- Avaliação de Desempenho (Redes Estocásticas)

Ferramenta de Suporte a Redes de Petri

• ARP — Analisador de Redes de Petri



Ferramenta de Suporte a Redes de Petri

• ARP — Analisador de Redes de Petri

```

NET nome_rede; { comentários}
CONST
  const1, const2, ... = valor;

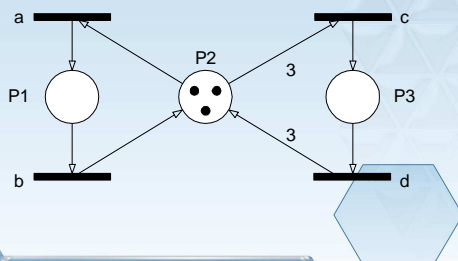
NODES
  t1, t2, ... : TRANSITION {[Tmin,Tmax]} {EXPON (1000)}
  p1, p2, ... : PLACE { (marcação inicial) };

STRUCTURE
  { entrada }          { saída }
  t1 :              (lugar, peso*lugar, ...)   (lugar, ... );
  ...
ENDNET.

```

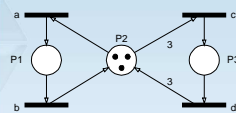
Ferramenta de Suporte a Redes de Petri

• ARP — Analisador de Redes de Petri



Ferramenta de Suporte a Redes de Petri

• ARP — Analisador de Redes de Petri



```

NET Ex1;

NODES
  p1,p3 : PLACE;
  p2 : PLACE(3);
  a,b,c,d : TRANSITION;

STRUCTURE
  a : (p2),(p1);
  b : (p1),(p2);
  c : (3*p2),(p3);
  d : (p3),(3*p2);

ENDNET.

```

Ferramenta de Suporte a Redes de Petri

• ARP — Analisador de Redes de Petri

```

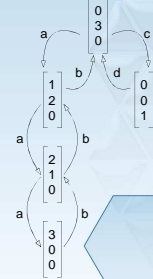
Reachability Graph
States Enumeration : net Ex1.
Reachability graph for this net:
*
M0 : (a: M1) (c: M4)
M1 : (a: M2) (b: M0)
M2 : (a: M3) (b: M1)
M3 : (b: M2)
M4 : (d: M0)
*
    
```

Ferramenta de Suporte a Redes de Petri

• ARP — Analisador de Redes de Petri

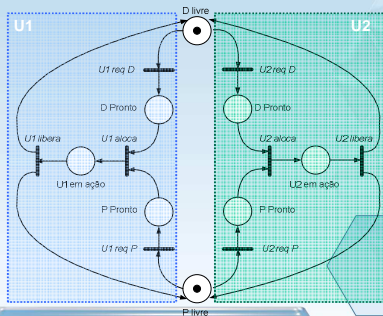
```

M0 : (a: M1) (c: M4)
M1 : (a: M2) (b: M0)
M2 : (a: M3) (b: M1)
M3 : (b: M2)
M4 : (d: M0)
    
```



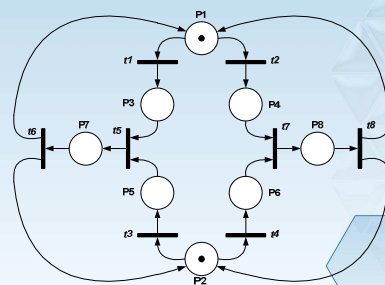
Análise das Redes de Petri

• Propriedades - Exercício



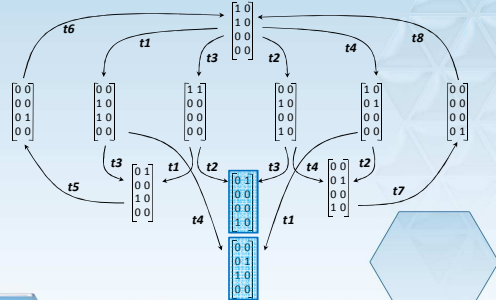
Análise das Redes de Petri

• Propriedades - Exercício



Análise das Redes de Petri

• Propriedades - Exercício



Análise das Redes de Petri

```

program recursos1;
var
  P,HD,tela : semaphore;
process type tpusuario (pid : integer);
var i : integer;
begin
  repeat
    sleep (random(4));
    if (random(2-1)+1) = 1
    then
      begin
        wait(tela);
        writeln('PROCESSO ',pid:2,' - PEDINDO IMPRESSORA');
        signal(tela);
        wait(P);
        wait(tela);
        writeln('PROCESSO ',pid:2,' - IMPRESSORA ALOCADA');
        signal(tela);
      end;
    end;
  end;
end;

```

Análise das Redes de Petri

• Implementação Pascal FC

```

wait(HD);
wait(tela);
writeln('PROCESSO ',pid:2,' - DISCO ALOCADO');
signal(tela);
end
else
begin
  wait(tela);
  writeln('PROCESSO ',pid:2,' - PEDINDO DISCO');
  signal(tela);
  wait(HD);
  wait(tela);
  writeln('PROCESSO ',pid:2,' - DISCO ALOCADO');
  signal(tela);
  sleep(random(4));
  wait(tela);
  writeln('PROCESSO ',pid:2,' - PEDINDO IMPRESSORA');
  signal(tela);
end;

```

Análise das Redes de Petri

```

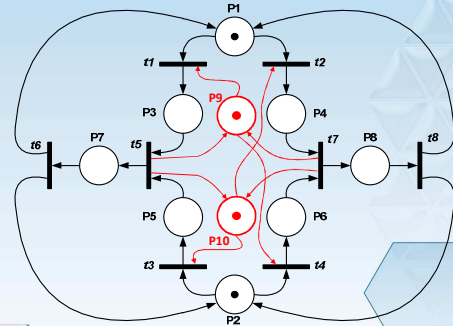
wait(tela);
writeln('PROCESSO ',pid:2,' - IMPRESSORA ALOCADA');
signal(tela);
end;
sleep(random(4));
wait(tela);
writeln('PROCESSO ',pid:2,' - TRABALHANDO');
signal(tela);
sleep(random(4));
wait(tela);
writeln('PROCESSO ',pid:2,' - DEVOLUCAO RECURSOS');
signal(tela);
signal(HD);
signal(P);
wait(tela);
for i:= 1 to 10 do write(' -- P',pid);
writeln(' --');
signal(tela);
forever;
end;

```

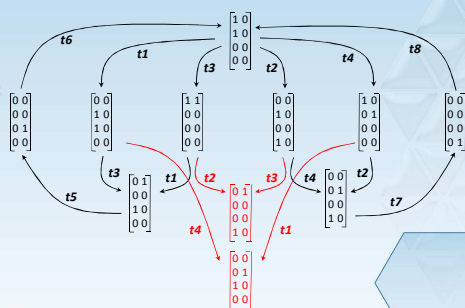
Análise das Redes de Petri

```
var
  u1,u2 : tpusuario;
begin
  initial(HD,1);
  initial(P,1);
  initial(tela,1);
  cobegin
    u1(1);
    u2(2);
  coend;
end.
```

Análise das Redes de Petri



Análise das Redes de Petri



Análise das Redes de Petri

```
program recursos2;
var
  P,HD,tela,mutex1,mutex2 : semaphore;
  process type tpusuario (pid : integer);
  var i : integer;
begin
  repeat
    sleep (random(4));
    if (random(2-1)+1) = 1
    then
      begin
        wait(mutex1);
        wait(tela);
        writeln('PROCESSO ',pid:2,' - PEDINDO IMPRESSORA');
        signal(tela);
        wait(P);
        wait(tela);
        writeln('PROCESSO ',pid:2,' - IMPRESSORA ALOCADA');
        signal(tela);
      end
    end
  repeat
```

Análise das Redes de Petri

```
sleep(random(4));
wait(mutex2);
wait(tela);
writeln('PROCESSO ',pid:2,' - PEDINDO DISCO');
signal(tela);
wait(HD);
wait(tela);
writeln('PROCESSO ',pid:2,'- DISCO ALOCADO');
signal(tela);
end
else
begin
wait(mutex1);
wait(tela);
writeln('PROCESSO ',pid:2,' - PEDINDO DISCO');
signal(tela);
wait(HD);
wait(tela);
writeln('PROCESSO ',pid:2,' - DISCO ALOCADO');
signal(tela);
sleep(random(4));
```

Análise das Redes de Petri

Implementação Pascal FC

```
wait(mutex2);
wait(tela);
writeln('PROCESSO ',pid:2,' - PEDINDO IMPRESSORA');
signal(tela);wait(P);
wait(tela);
writeln('PROCESSO ',pid:2,' - IMPRESSORA ALOCADA');
signal(tela);
end;
sleep(random(4));
wait(tela);
writeln('PROCESSO ',pid:2,' - TRABALHANDO');
signal(tela);
sleep(random(4));
wait(tela);
writeln('PROCESSO ',pid:2,' - DEVOLUCAO RECURSOS');
signal(tela);
signal(mutex1);
signal(mutex2);
```

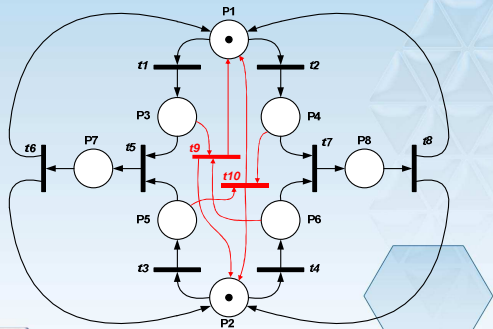
Análise das Redes de Petri

```
signal(HD);
signal(P);
wait(tela);
for i:= 1 to 10
do
write(' -- P',pid);
writeln(' --');
signal(tela);
forever;
end;
```

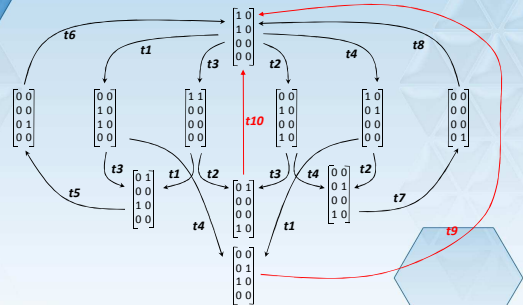
Análise das Redes de Petri

```
var
u1,u2 : tpusuario;
begin
initial(HD,1);
initial(P,1);
initial(tela,1);
initial(mutex1,1);
initial(mutex2,1);
cobegin
u1(1);
u2(2);
coend;
end.
```

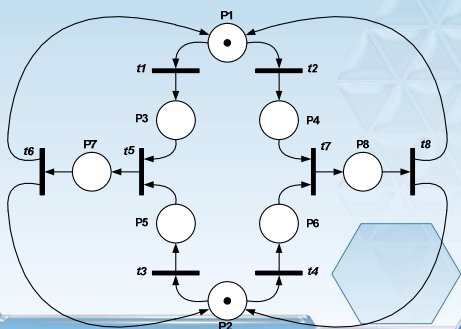
Análise das Redes de Petri



Análise das Redes de Petri



ARP — Analisador de Redes de Petri



ARP — Analisador de Redes de Petri

```

NET RDP1;

NODES
  p3,p4,p5,
  p6,p7,p8 : PLACE;
  p1,p2 : PLACE(1);
  t1,t2,t3,t4,
  t5,t6,t7,t8 : TRANSITION;

STRUCTURE
  t1 : (p1),(p3);
  t2 : (p1),(p4);
  t3 : (p2),(p5);
  t4 : (p2),(p6);
  t5 : (p3,p5),(p7);
  t6 : (p7),(p1,p2);
  t7 : (p4,p6),(p8);
  t8 : (p8),(p1,p2);

ENDNET.
    
```

ARP — Analisador de Redes de Petri

```

P:\Trabalho\INES41-1\FERRAM-1\ARP\ARP-2-4\ARP2-4.EXE

Current Net : P:\TRABALHO\INES41-1\FERRAM-1\ARP\ARP-2-4\INES410.PN
Directory : P:\TRABALHO\INES41-1\FERRAM-1\ARP\ARP-2-4

M:lags Simulation Edit Help
Performance Evaluation Verification Post Quit

States Enumeration : net RDPI (11 reachable states).
Verified properties:
Net under analysis is binary.
Null places (N = 0): {}
Binary places : {all}
k-Bounded places : {}
Unbounded places : {}
Net under analysis is not strictly conservative.
Net under analysis is not live.
Live Tr : {}
"Almost-live" Tr : {all}
Non-fired Tr : {}
States from which the net cannot go back to M0: M4 M6
No live-locks detected.
States (and fire sequences) in deadlock:
M4 :t1 t4
M6 :t2 t3
    
```

ARP — Analisador de Redes de Petri

```

P:\Trabalho\INES41-1\FERRAM-1\ARP\ARP-2-4\ARP2-4.EXE

Current Net : P:\TRABALHO\INES41-1\FERRAM-1\ARP\ARP-2-4\INES410.PN
Directory : P:\TRABALHO\INES41-1\FERRAM-1\ARP\ARP-2-4

M:lags Simulation Edit Help
Performance Evaluation Verification Post Quit

States Enumeration : net RDPI.
Reachability Graph for this net:
M0 :<t1: M1> <t2: M5> <t3: M9> <t4: M10>
M1 :<t3: M2> <t4: M3>
M2 :<t5: M3>
M3 :<t4: M0>
M4 :
M5 :<t3: M6> <t4: M7>
M6 :
M7 :<t7: M8>
M8 :<t8: M0>
M9 :<t8: M5> <t2: M6>
M10 :<t1: M5> <t2: M7>
    
```

ARP — Analisador de Redes de Petri

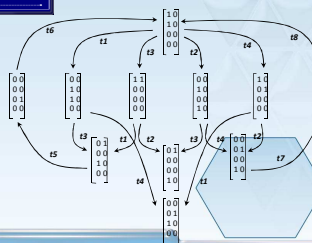
```

P:\Trabalho\INES41-1\FERRAM-1\ARP\ARP-2-4\ARP2-4.EXE

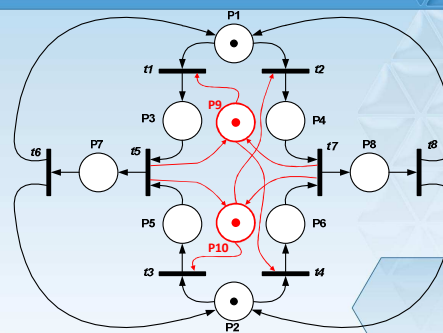
Current Net : P:\TRABALHO\INES41-1\FERRAM-1\ARP\ARP-2-4\INES410.PN
Directory : P:\TRABALHO\INES41-1\FERRAM-1\ARP\ARP-2-4

M:lags Simulation Edit Help
Performance Evaluation Verification Post Quit

States Enumeration : net RDPI.
Reachability Graph for this net:
M0 :<t1: M1> <t2: M5> <t3: M9> <t4: M10>
M1 :<t3: M2> <t4: M3>
M2 :<t5: M3>
M3 :<t4: M0>
M4 :
M5 :<t3: M6> <t4: M7>
M6 :
M7 :<t7: M8>
M8 :<t8: M0>
M9 :<t8: M5> <t2: M6>
M10 :<t1: M5> <t2: M7>
    
```



ARP — Analisador de Redes de Petri



ARP — Analisador de Redes de Petri

```

NET RDP2;
NODES
  p3,p4,
  p5,p6,
  p7,p8 : PLACE;
  p1,p2,
  p9,p10 : PLACE(1);
  t1,t2,
  t3,t4,
  t5,t6,
  t7,t8 : TRANSITION;
STRUCTURE
  t1 : (p1,p9),(p3);
  t2 : (p1,p10),(p4);
  t3 : (p2,p10),(p5);
  t4 : (p2,p9),(p6);
  t5 : (p3,p5),(p7,p9,p10);
  t6 : (p7),(p1,p2);
  t7 : (p4,p6),(p8,p9,p10);
  t8 : (p8),(p1,p2);
ENDNET.

```

ARP — Analisador de Redes de Petri

```

State Enumeration : net RDP2 <9 reachable states>.
Verified properties:
Net under analysis is binary.
Null places (N = 0): {}
Binary places : {all}
k-Bounded places : {}
Unbounded places : {}
Net under analysis is not strictly conservative.
Net under analysis is live.
Live fr... : {all}
Net can always go back to M0.
No live-locks detected.
No deadlocks detected.

```

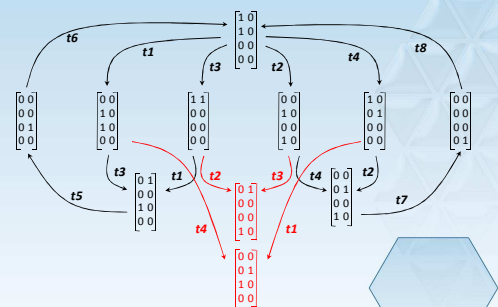
ARP — Analisador de Redes de Petri

```

States Enumeration : net RDP2.
Reachability Graph
Reachability graph for this net:
M0 : <t1: M1> <t2: M4> <t7: M7> <t8: M8>
M1 : <t7: M2>
M2 : <t3: M3>
M3 : <t4: M0>
M4 : <t8: M5>
M5 : <t5: M6>
M6 : <t6: M0>
M7 : <t1: M2>
M8 : <t2: M5>

```

ARP — Analisador de Redes de Petri



Aplicações Industriais das RdP

- Especificação do comportamento de um chip VLSI para uso em supercomputador – 1991
- Transações bancárias na HP – 1993
- Especificação de software de celulares Nokia – 1998
- Validação de arquiteturas de computadores na Universidade de Newcastle – 1998
- Validação de Rede para Eletrônica Embarcada (CAN) – 2008
- Validação de protocolo de roteamento para Redes Ad-Hoc – 2008
- Especificação e Análise do protocolo de negociação SIP (VoIP) - 2008