

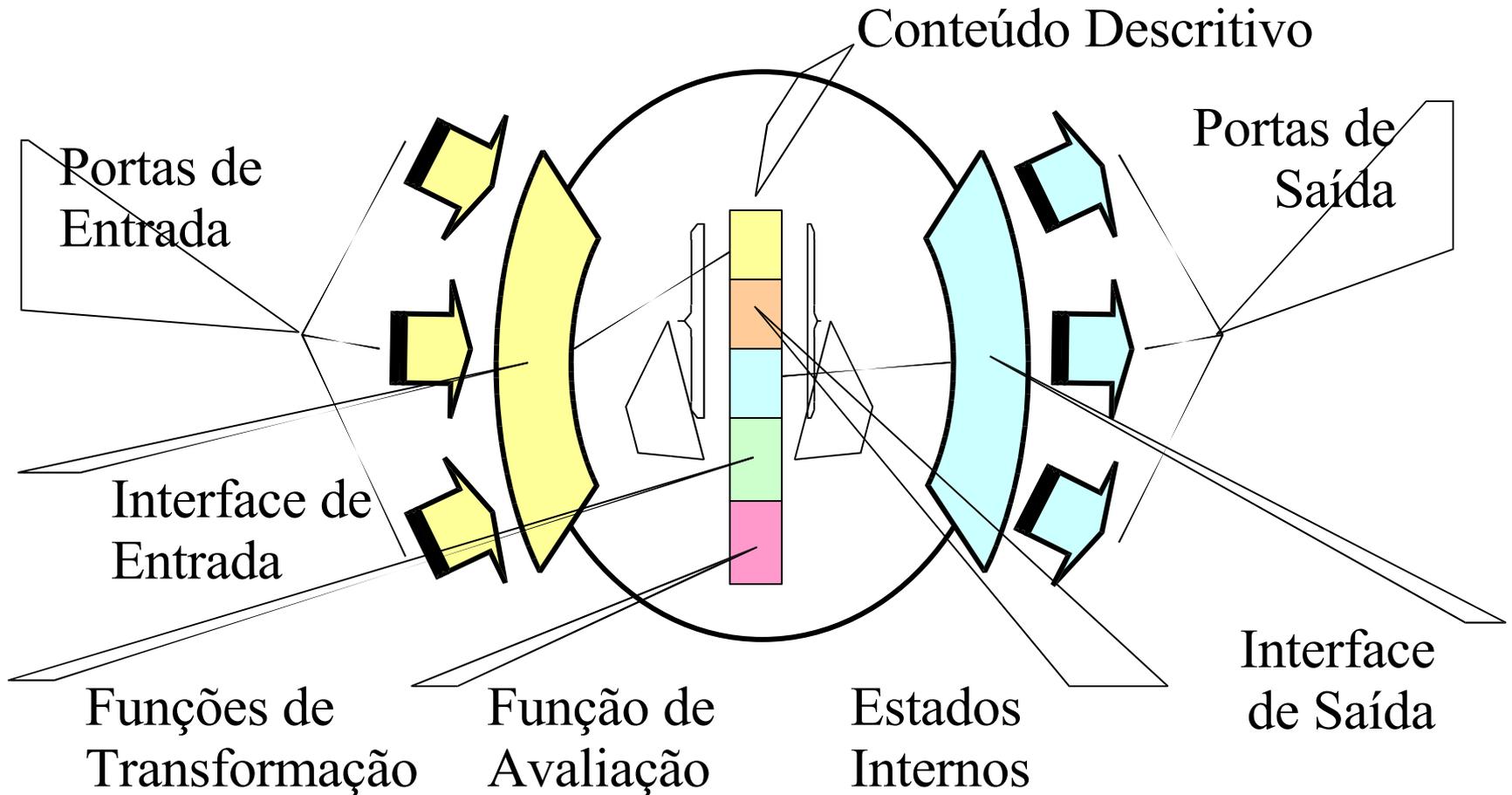


# Redes de Agentes

- Origem
  - Teoria Matemática de Objetos
- Utilização
  - Semiótica Computacional
    - Representação e Processamento de Conhecimento
  - Sistemas Dinâmicos a Eventos Discretos
  - Modelagem Organizacional e Operacional
  - Integração entre Elementos Processadores de Informação
    - Modelos Híbridos
- Agente Semiônico
  - **Sêmion**: Agente elementar por meio do qual um sistema semiótico pode ser construído



# Agente Semiônico





# Agente Semiônico

## ■ Agentes Semiônicos

- são agrupados e classificados em **classes**, da mesma maneira que objetos

## ■ Classes

- Variáveis de Entrada do Agente
- Variáveis de Saída do Agente
- Variáveis Internas do Agente
- Funções de Transformação do Agente
- Função de Avaliação do Agente

## ■ Diferença entre agentes semiônicos e objetos

- possuem um ciclo de atividade contínuo
- possuem uma função de avaliação que orienta o comportamento dinâmico do agente



# Comportamento do Agente Semiônico

- Dividido em cinco diferentes fases:
  - Fase de Avaliação
  - Fase da Atribuição
  - Fase da Assimilação
  - Fase da Transformação
  - Fase da Consumoção
- Sequência de Fases
  - Responsável por regular a interação entre sêmions e objetos



# Fase da Avaliação

## ■ Fase da Avaliação

- Começa quando um dado sêmions determina quais os objetos ou outros sêmions com os quais ele deve interagir
- O Sêmion deve avaliar cada objeto ou outro sêmion disponível para interação, e decidir o que deve acontecer a este objeto/sêmion após a interação
- Para cada função de transformação disponível no sêmion
  - Um conjunto de objetos interagentes do tipo correto é determinado
  - O Sêmion testa todas as possíveis combinações de objetos que sejam compatíveis com os tipos de entradas de suas funções de transformação



# Fase da Avaliação

## ■ Escopo Habilitante

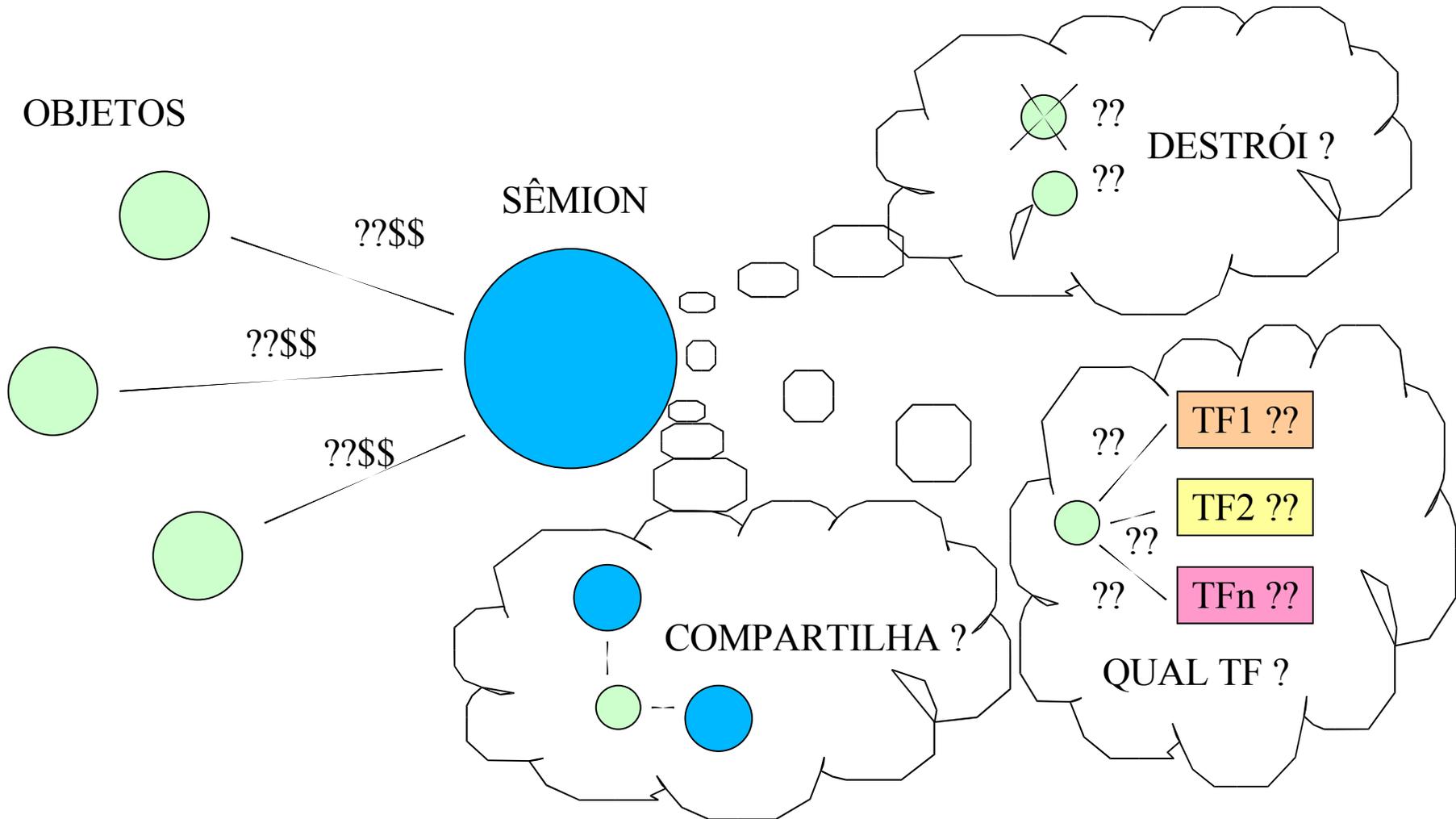
- Cada possível combinação compatível com uma função de transformação particular
- Lista de objetos/outros sêmions potencialmente disponíveis para interação
- Avaliado por meio da função de avaliação
- Determina se os objetos/outros sêmions serão modificados, simplesmente devolvidos a seus lugares ou destruídos

## ■ A Fase Acaba quando

- O sêmion avalia todos os escopos habilitantes possíveis e determina para cada um valor de avaliação e um modo de acesso pretendido
- O Modo de Acesso pretendido descreve as intenções do sêmion para com cada um de seus objetos de entrada. Ele deve informar se sêmion permite o compartilhamento do objeto com outros sêmions e se pretende destruir o objeto depois da interação



# Fase da Avaliação





# Fase da Atribuição

## ■ Fase da Atribuição

- Um algoritmo supervisor central coleta os desejos de interação de cada sêmion ativo e um sistema semiônico, e atribui a cada sêmion ativo um escopo habilitante
- Esta atribuição deve evitar qualquer tipo de conflito com o desejo de outros sêmions
- Vários algoritmos diferentes podem ser usados nesta fase
- Nosso grupo desenvolveu para testes (Guerrero et. al. 1999), um algoritmo que chamamos de BMSA (Best Matching Search Algorithm),
  - Atribui um objeto ao sêmion ativo que melhor o avaliou, respeitando-se os modos de acesso pretendidos de cada sêmion



# Fase da Assimilação

## ■ Fase de Assimilação

- Sêmions ativos assimilam o conteúdo descritivo de seus objetos interagentes, efetuando uma cópia destes em sua interface de entrada, por meio das portas de entrada
- Depois disso, a interface de entrada é copiada novamente nos estados internos do conteúdo descritivo do sêmion
- A seguir, os objetos são devolvidos ao ambiente, destruídos ou transferidos para as portas de saída do sêmion para uma nova inoculação, dependendo-se do modo de acesso prescrito.



# Fases da Transformação e da Consumo

## ■ Fase da Transformação

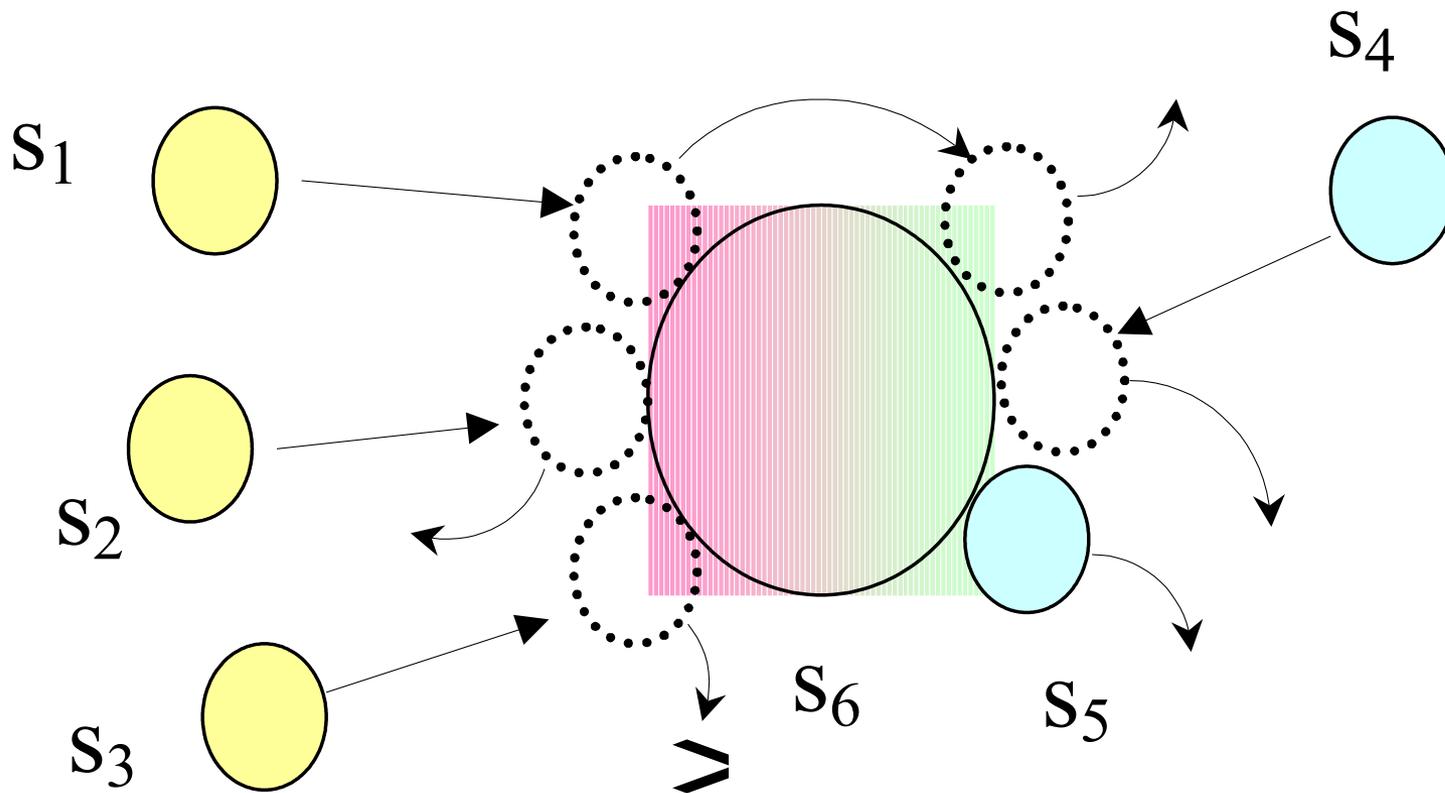
- A função de transformação atribuída é invocada, determinando os novos valores dos estados internos dos sêmions e dos estados correspondentes às saídas

## ■ Fase da Consumo

- Os estados de saída do conteúdo descritivo são copiados nas portas de saída, de forma a inocular os objetos de saída, que se conectam às portas de saída
- Estes objetos de saída podem ser
  - Objetos de entrada que foram transferidos às portas de saída
  - Um objeto externo que se conectou à porta de saída
  - Um objeto completamente novo, criado durante esta fase
- O conteúdo descritivo destes objetos é mudado para incorporar os novos valores armazenados na interface de saída, e depois desta inoculação, os objetos de saída são enviados ao ambiente



# Interação entre Agentes Semiônicos





# Interação entre Agentes Semiônicos

- Seleção de Agentes para Assimilação
  - **Função de Avaliação** - todos os outros agentes disponíveis para assimilação são avaliados
    - Múltiplas Funções de Transformação
      - Avaliação é feita considerando-se cada função de transformação
  - Escopos Habilitantes
  - **Algoritmo de Seleção** - Baseado na avaliação, um algoritmo de seleção deve fazer a escolha
    - escolha deve evitar conflitos com outros agentes querendo interagir com um mesmo agente
    - **algoritmo BMSA** (Best Matching Search Algorithm)
- Assimilação dos Agentes Escolhidos
  - Absorção do conteúdo descritivo do agente
  - Transporte, Liberação ou Destruição do Agente



# Interação entre Agentes Semiônicos

- Processamento do Conteúdo Descritivo
  - **Funções de Transformação:** processam o conteúdo descritivo dos agentes assimilados podendo
    - | **alterar** o conteúdo descritivo de algum agente assimilado
    - | **alterar** o conteúdo descritivo de algum outro agente
    - | **gerar** um novo agente no sistema
- **Casos Especiais**
  - **Agente Fonte**
    - | utilizado para **introduzir** novos agentes no sistema
    - | agente não tem interface de entrada, e a função de avaliação simplesmente escolhe a função de transformação a ser utilizada
  - **Agente Vertedouro**
    - | utilizado para **retirar** agentes do sistema
    - | não tem função de transformação



# Sistemas de Agentes Semiônicos

- **Sistemas de Agentes Semiônicos**
  - Conjunto de Agentes Semiônicos interagindo entre si
- **Sistemas Fechados**
  - normalmente um sistema de agentes semiônicos é um sistema fechado
- **Sistemas Abertos**
  - podem ser emulados por meio de agentes-fonte e agentes-vertedouro
- **Agentes-Fonte**
  - internamente coletam informações de alguma fonte externa
- **Agentes-Vertedouro**
  - internamente enviam informações para fontes externas



# Sistemas de Agentes Semiônicos

## ■ Problema

- a medida que o tamanho da população de agentes aumenta, a demanda computacional aumenta exponencialmente
  - | agentes precisam avaliar todos os agentes disponíveis para interação
  - | custo computacional aumenta exponencialmente com o aumento do tamanho da população de agentes
  - | nem todos os agentes são interessantes para interação
    - tipos inadequados
    - conteúdo indesejado

## ■ Solução

- encontrar alguma maneira de agrupar os agentes de forma que somente os agentes que têm realmente algum interesse sejam sondados para interação



# Redes de Agentes Semiônicos

## Agentes

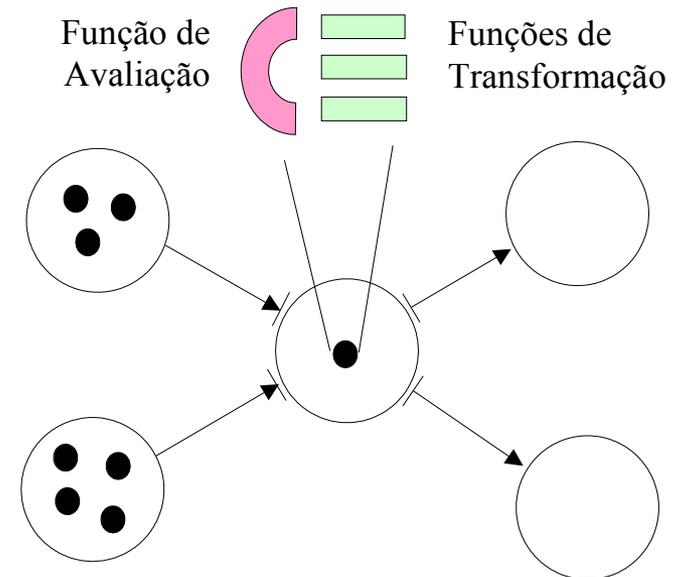
- confinados a lugares

## Lugares

- conectados por arcos
- arcos entram e saem de portas
- portas de entrada e saída
- agentes do mesmo tipo

## Vantagens

- agentes disponíveis para assimilação podem ser agrupados e localizados, o que evita a avaliação de toda uma população de agentes





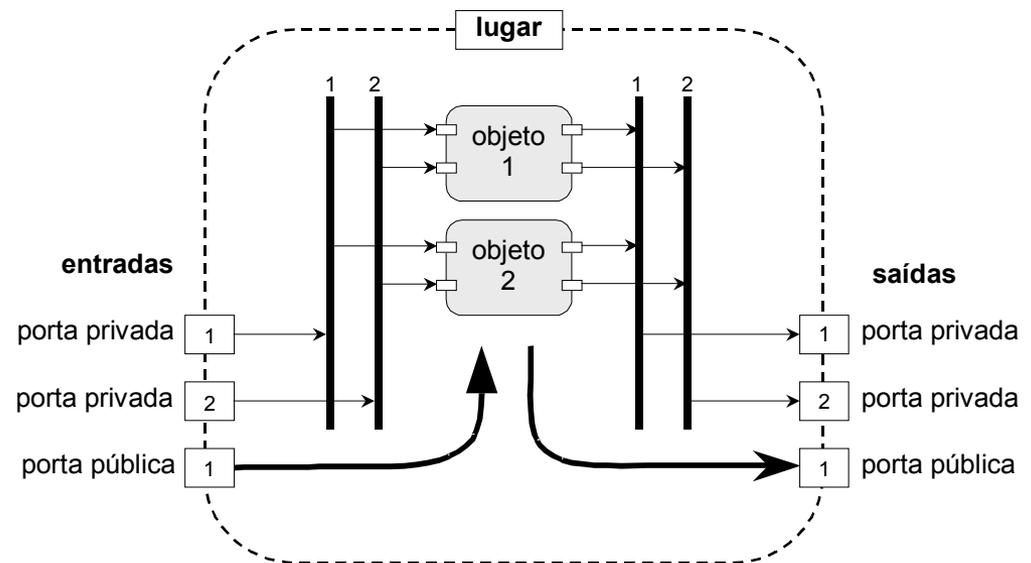
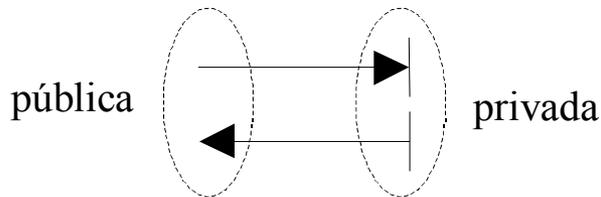
# Redes de Agentes Semiônicos

## ■ Portas

- privadas e públicas

## ■ Arcos

- entre portas públicas e privadas



## ■ Modos de Acesso

- **compartilhamento** de objetos : exclusivo ou não-exclusivo
- **destruição** : consumo ou não-consumo

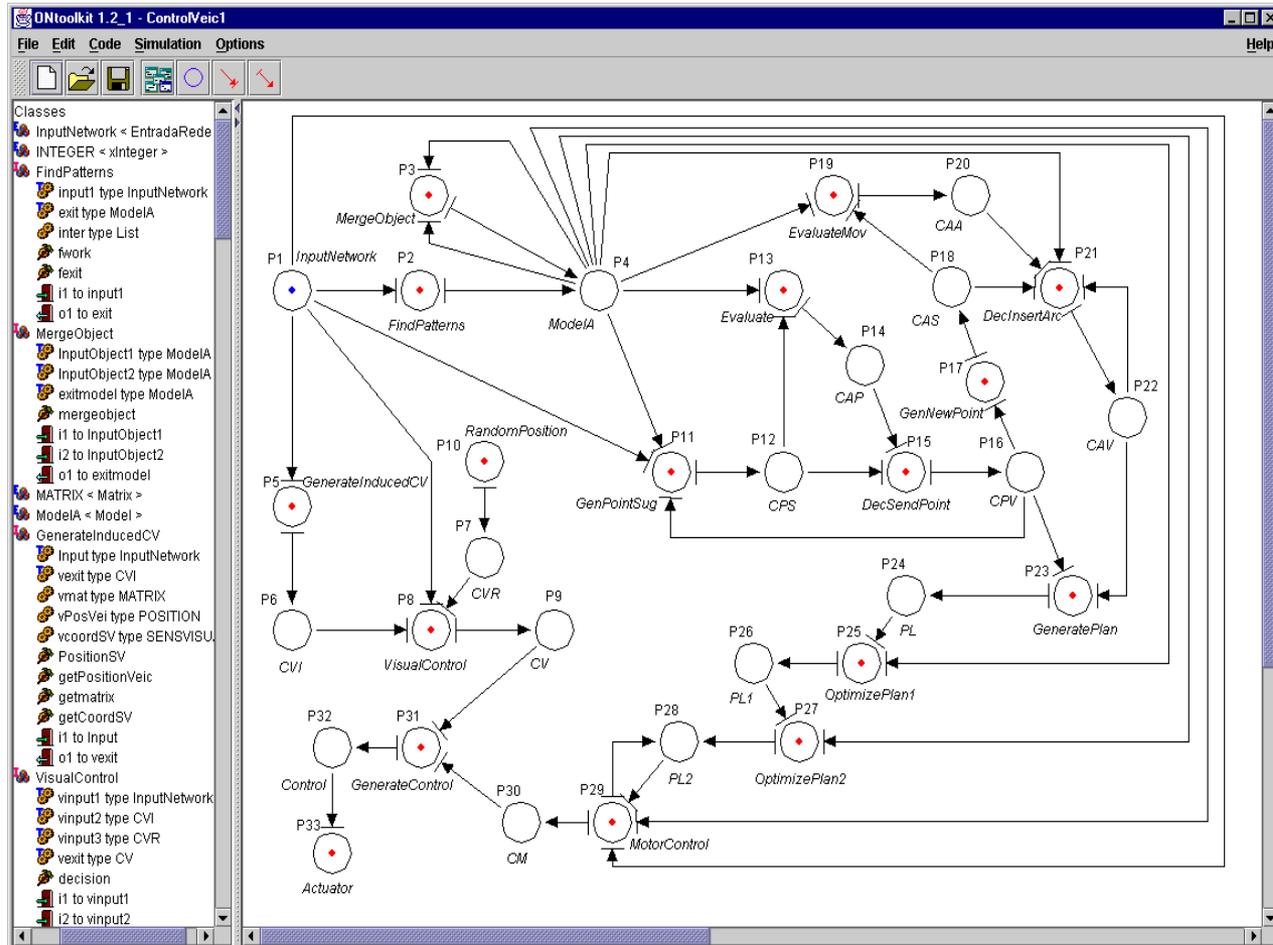


# ONToolkit

- **ONtoolkit** (*Object Network toolkit*)
  - auxiliar no *design* e simulação de redes de agentes
  - fornece um *engine* que implementa os mecanismos necessários para a execução de redes de agentes



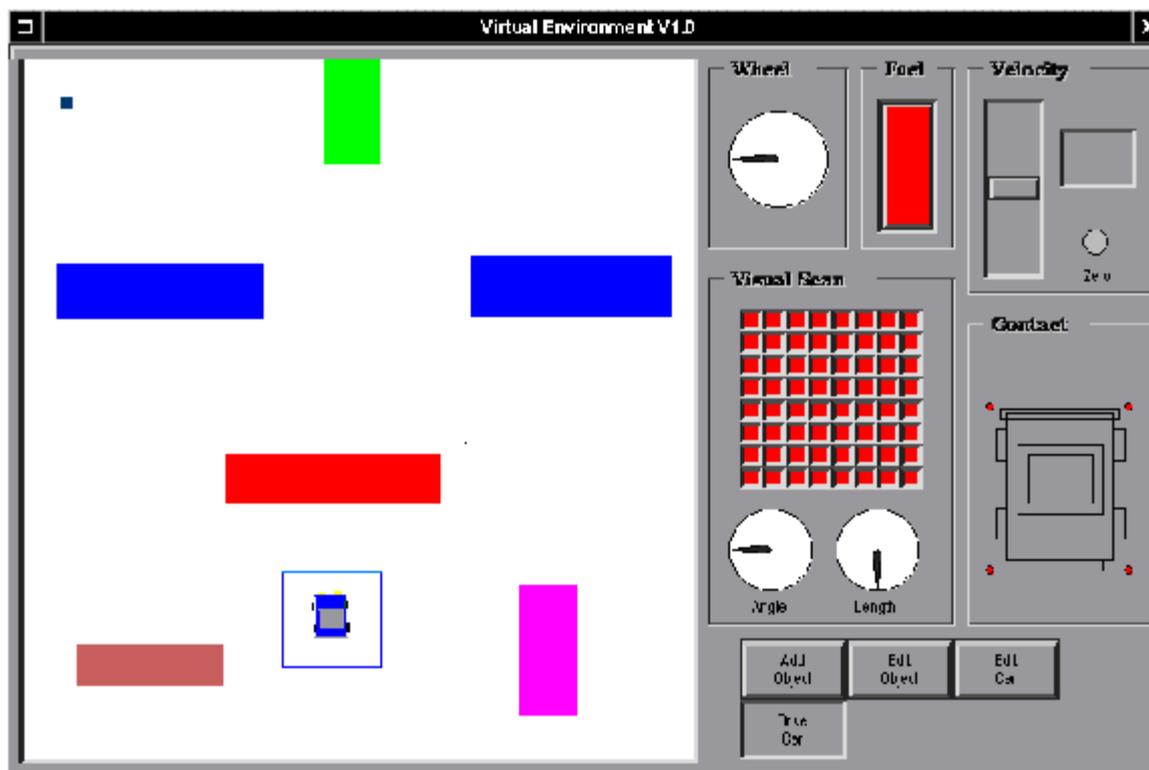
# ANToolkit





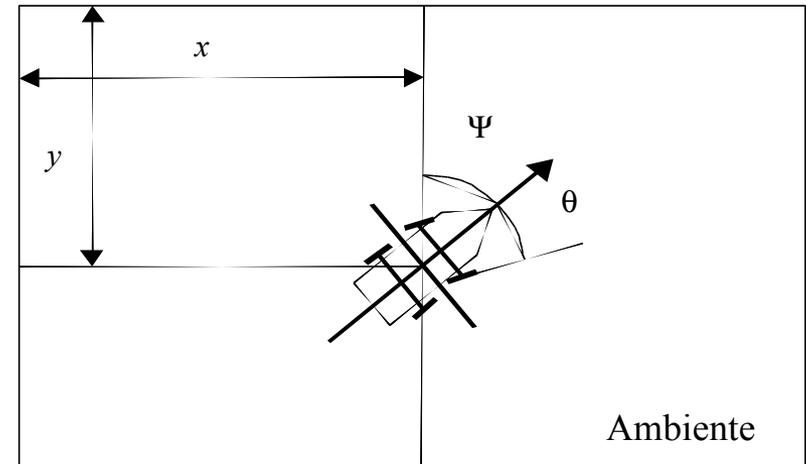
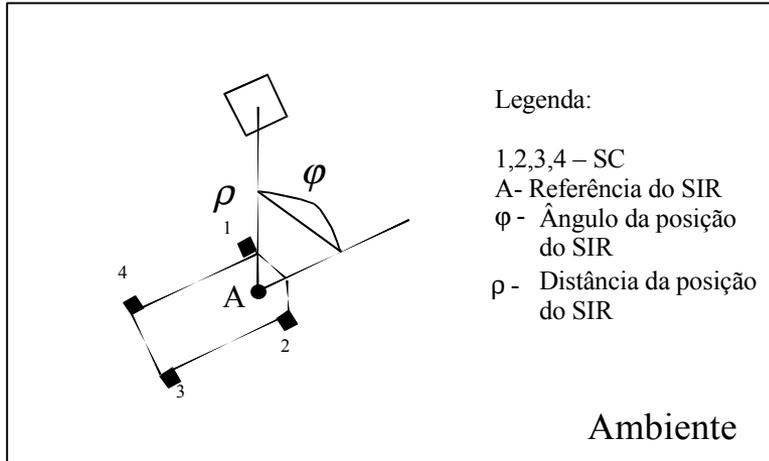
# Aplicações Desenvolvidas

- Simulação Robótica Autônoma em Mundos Virtuais





# Modelo do Veículo Autônomo



## Tipos de Sensores:

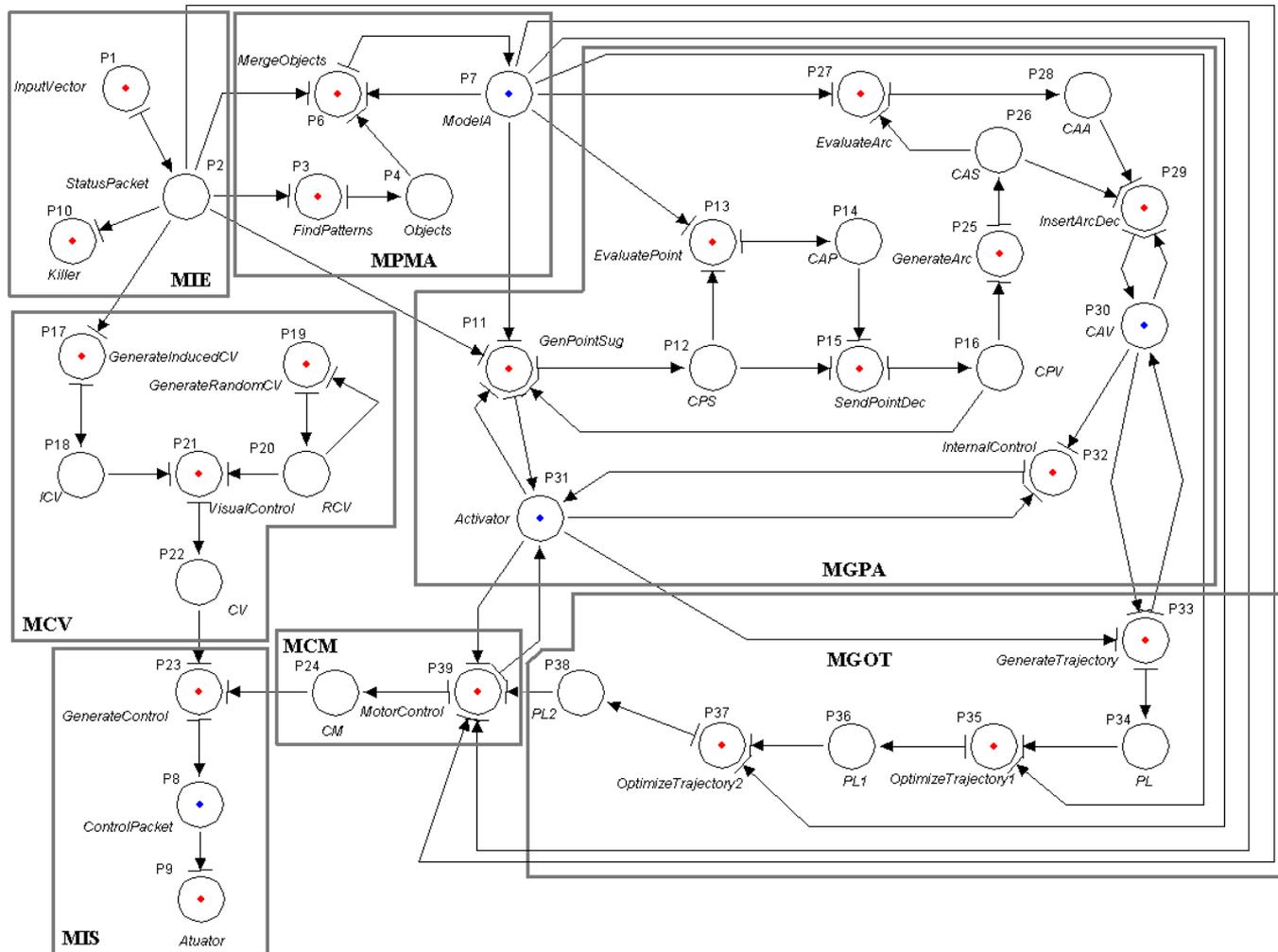
- Sensor de informação remota (**SIR**): simplificação de um mecanismo de visão.
- Sensores de contato (**SC**): informam quando existe contato com um objeto.

## Variáveis de interesse:

- posição do veículo ( $x, y, \psi$ ).
- ângulo das rodas em relação ao eixo longitudinal do veículo ( $\theta$ ).
- velocidade nominal do veículo ( $v$ ).

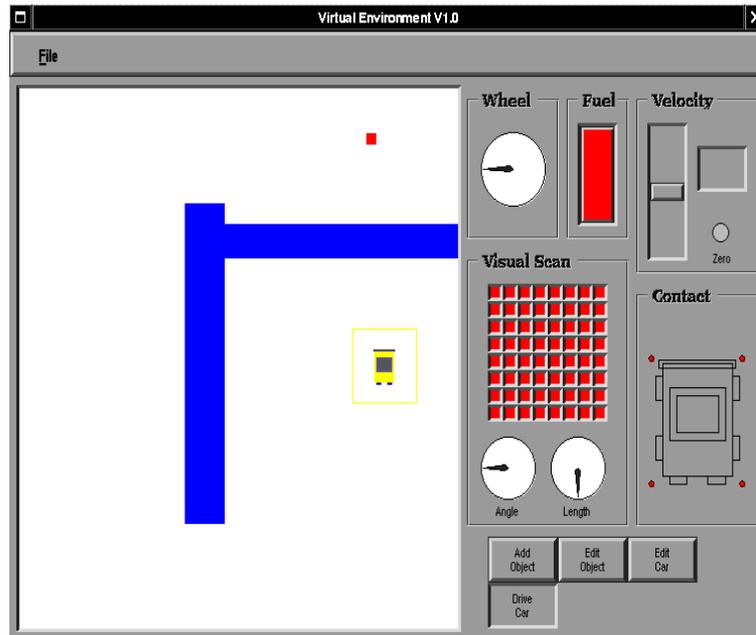


# Controle do Veículo Autônomo por Redes de Agentes

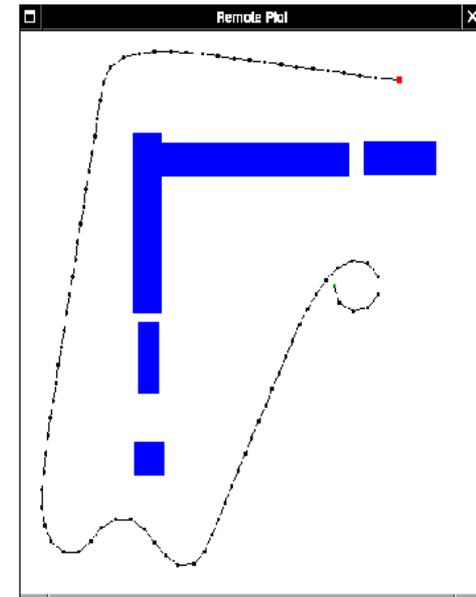




# Resultados de Simulação



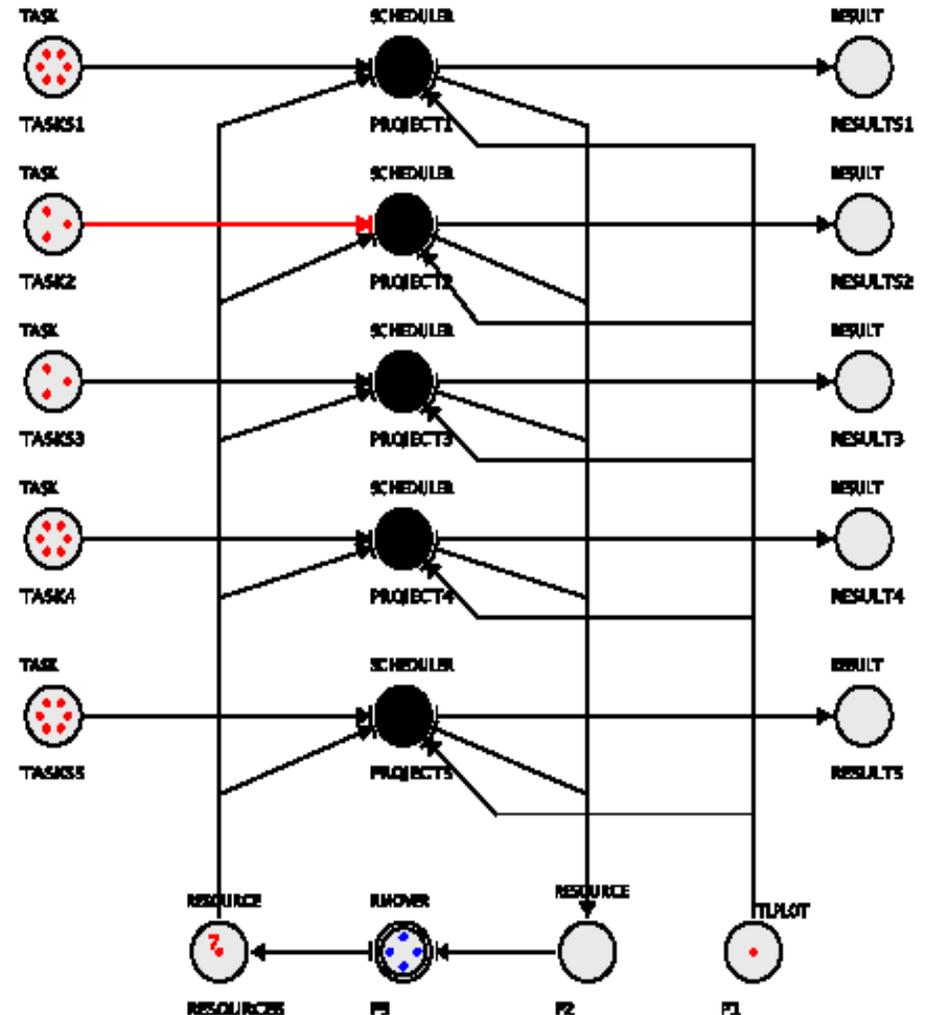
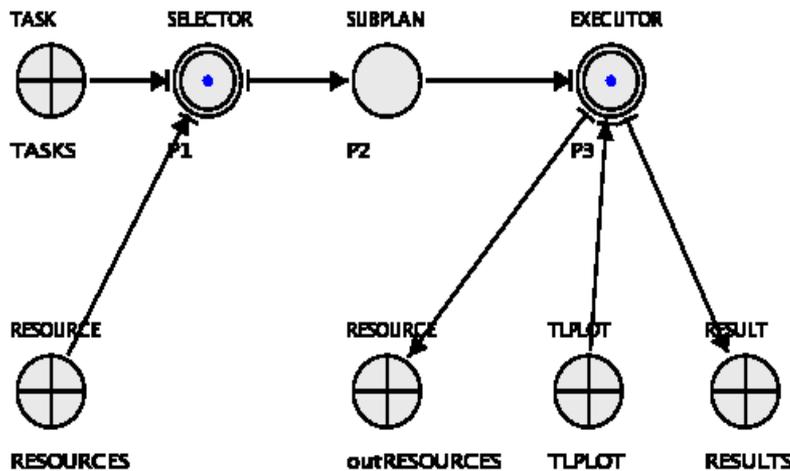
Ambiente de Simulação



Modelo Interno do Robô  
e sua Trajetória até a Meta

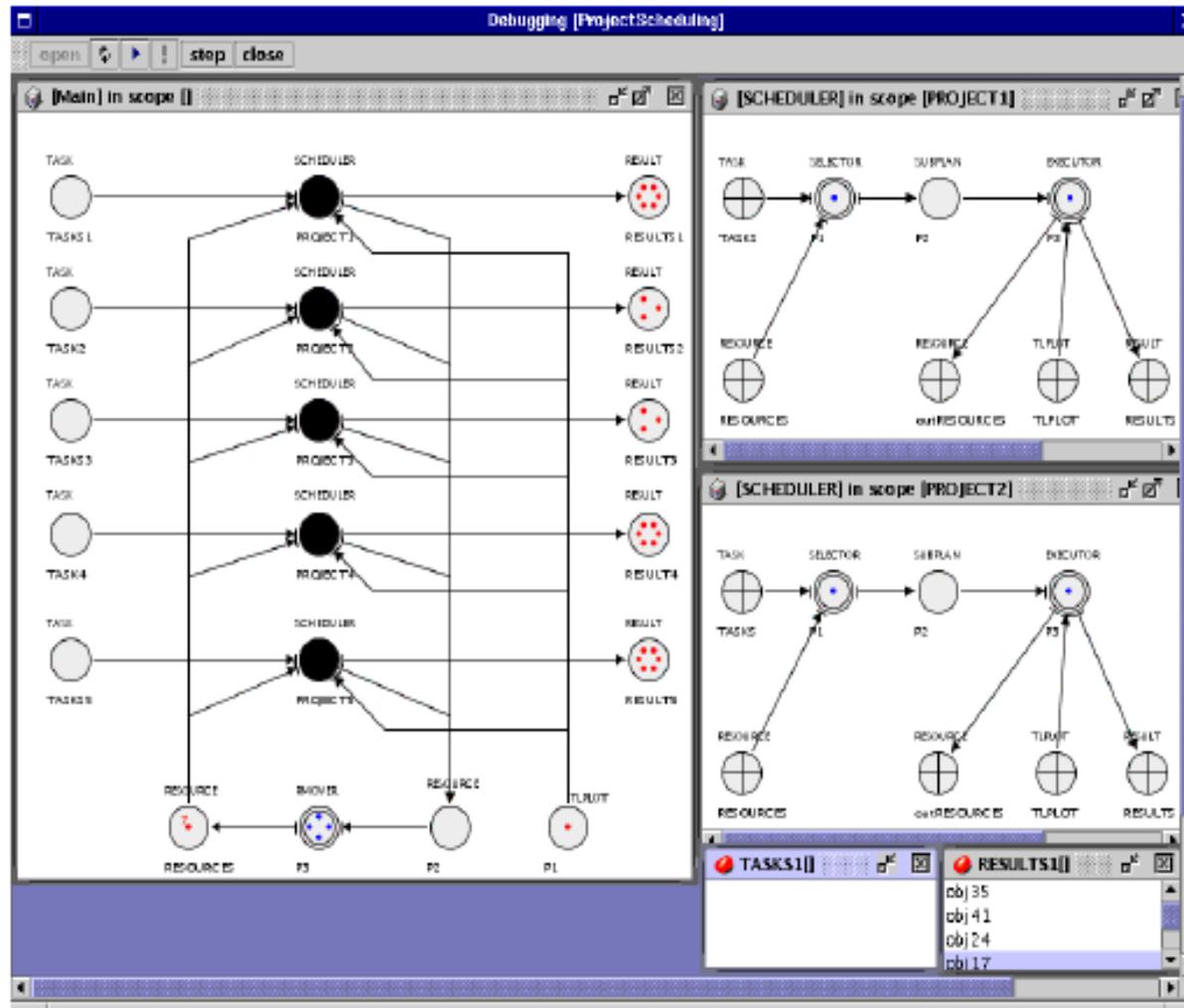


# Semiótica Organizacional: Gerência de Projetos



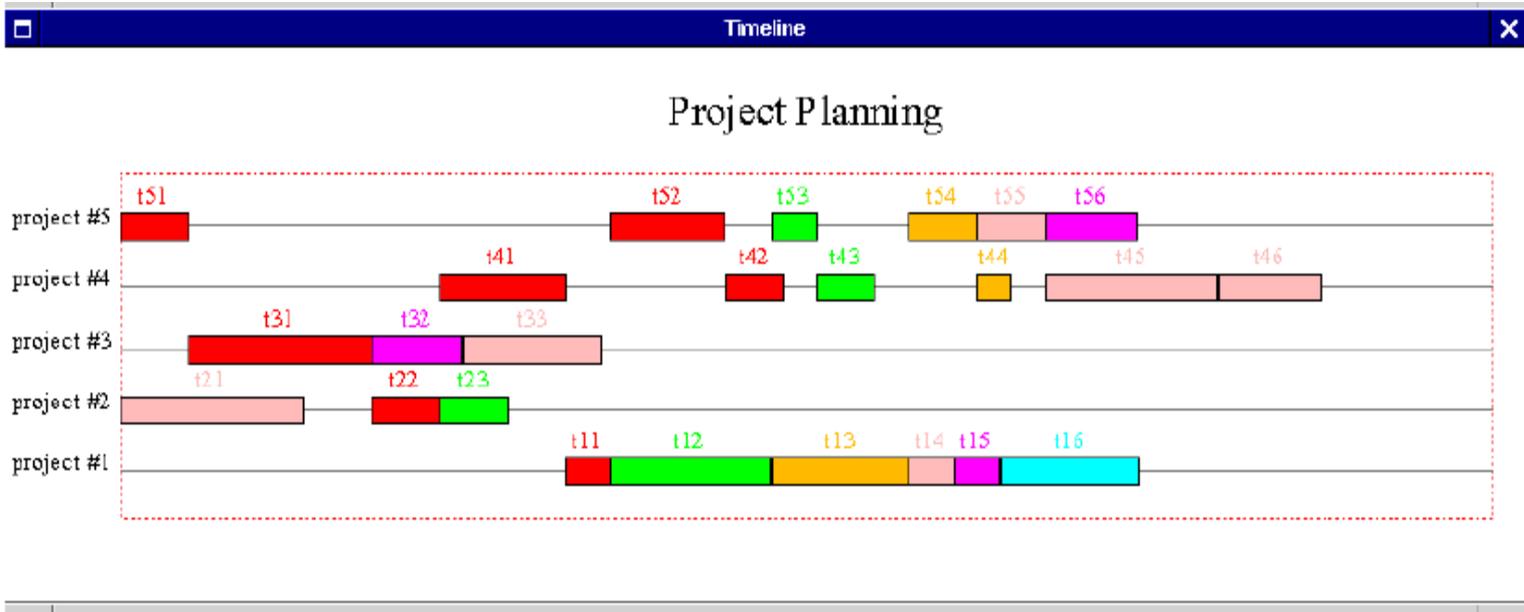


# Simulação da Gerência de Projetos





# Resultados de Simulação: Gerência de Projetos



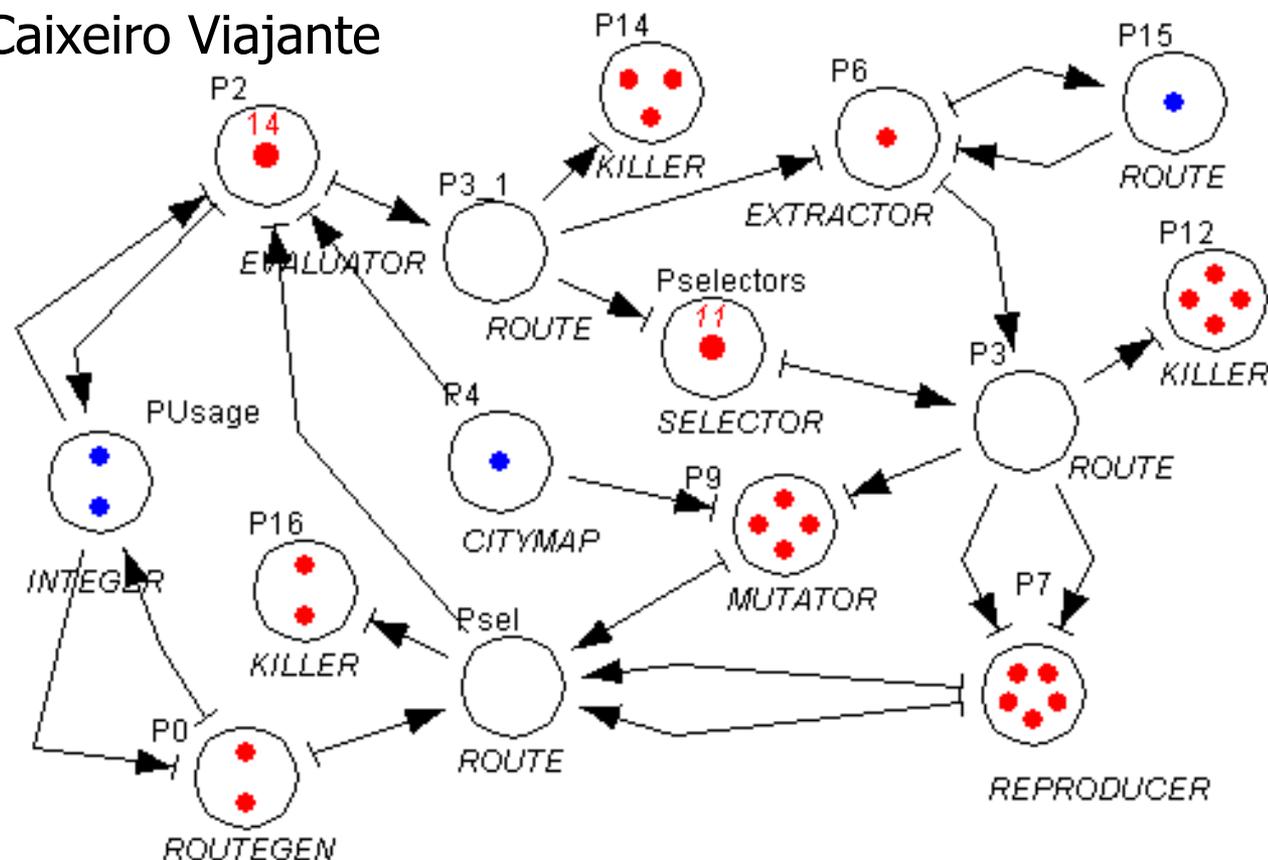


# Outros Exemplos de Modelos

## Algoritmo Genético (Modelo Interno)

### Problema do Caixeiro Viajante

- Classes
  - ROUTE <is> Route
  - CITYMAP <is> CityMap
  - EVALUATOR
  - ROUTEGEN
  - EXTRACTOR
  - REPRODUCER
  - DOUBLE <is> xDouble
  - MUTATOR
  - INTEGER <is> xInteger
  - MOVER
  - SELECTOR
  - KILLER
  - Network TSP



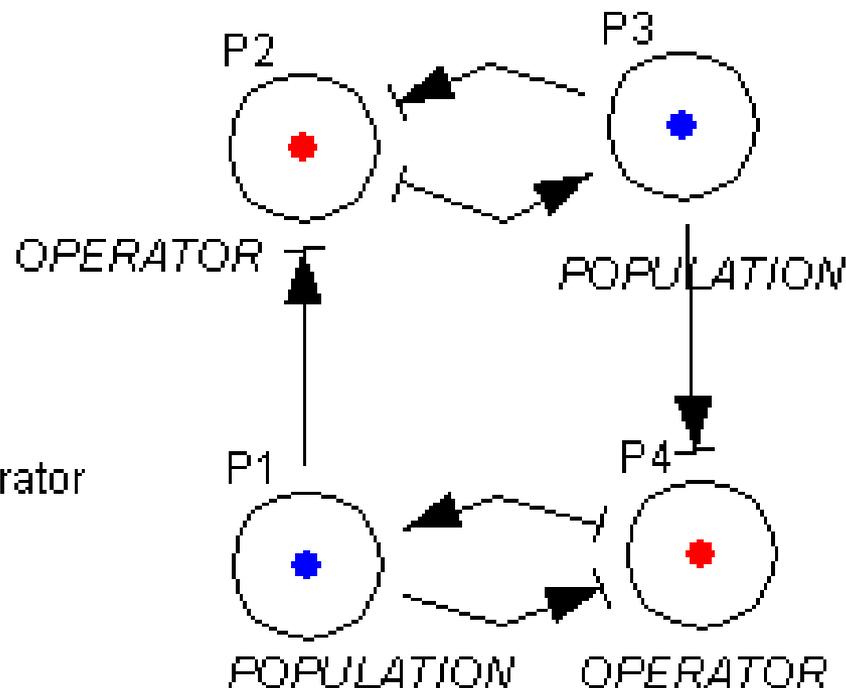


# Outros Exemplos de Modelos

## Algoritmo Genético (Modelo Embutido)

### Problema do Caixeiro Viajante

- Classes
  - POPULATION <is> MTONPopulation
  - ROUTE <is> Route
  - POP\_OPERATOR <is> MTONPopOperator
- OPERATOR
- CITYMAP <is> CityMap
- Network TSP





# Outros Exemplos de Modelos

## ■ Controlador Fuzzy

### ■ Controle de um Veículo A

- Classes
  - INPUTS
  - VALUE <is> xDouble
  - RULEBASE <is> RuleBase
  - INFERENCE\_ENGINE
  - FUZZYSET <is> FuzzySet
  - DEFUZZYFIER
- Network FuzzyController

