



PONTIFICIA
UNIVERSIDAD
CATÓLICA
DEL PERÚ

INTELIGENCIA ARTIFICIAL (INF371)

AGENTES INTELIGENTES (PARTE 2)

Dr. Edwin Villanueva Talavera

□ Estructura de Agentes

Bibliografía:

Capítulo 2.4 del libro:

Stuart Russell & Peter Norvig “[Artificial Intelligence: A modern Approach](#)”,
Prentice Hall, Third Edition, 2010

*** Todas las figuras de las presentes laminas han sido extraídas de este libro**

Estructura de Agentes

- Un agente queda completamente especificado si se define la **función del agente** (mapeamiento de secuencias de percepciones en acciones)
- En la práctica es difícil especificar explícitamente la función del agente
- Al diseñar agentes se busca entonces encontrar una forma de representar concisamente la función racional del agente, esto es, se busca implementar un **programa de agente** para una arquitectura dada

Agente = arquitectura + programa

Estructura de Agentes

Programa de Agente Dirigido por Tabla

```
Function TABLE_DRIVEN_AGENT(percept) return action
```

Variables estáticas:

- *percepts*, una secuencia, inicialmente vacía
- *table*, tabla de acciones, indexada por secuencias de percepciones, de inicio completamente especificada

```
append percept to the end of percepts
```

```
action ← LOOKUP(percepts, table)
```

```
return action
```

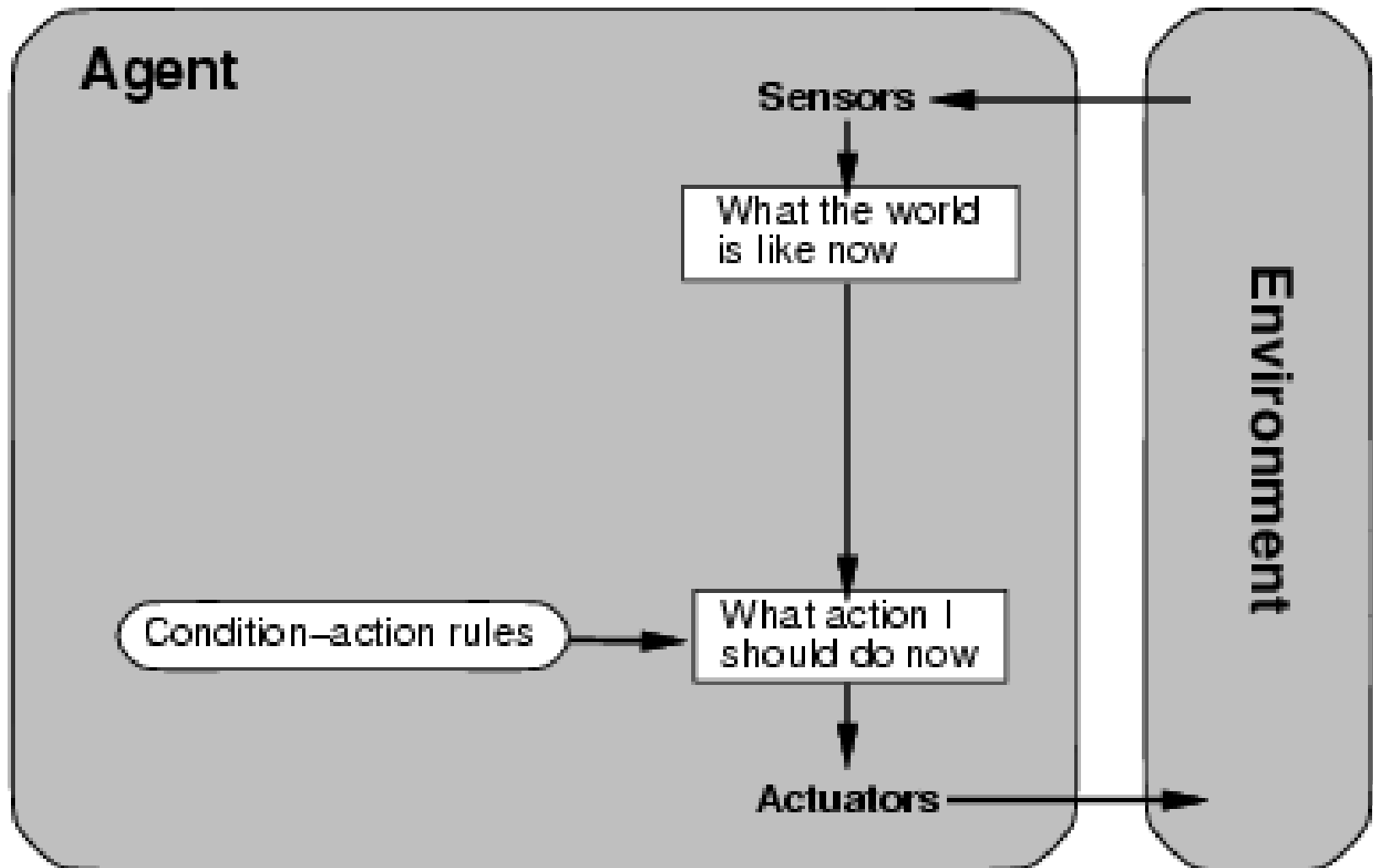
- Desventajas:
 - Tabla gigante (ajedrez = 10^{150} entradas)
 - Mucho tiempo para construir la tabla
 - No tiene autonomía
 - Aunque use aprendizaje automático, tardaría mucho en aprender la tabla.

Tipos básicos de programas de agentes

- Agentes reactivos simples
- Agentes reactivos basados en modelos
- Agentes basados en objetivos
- Agentes basados en utilidad

Estructura de Agentes

Agente Reactivo Simple



Estructura de Agentes

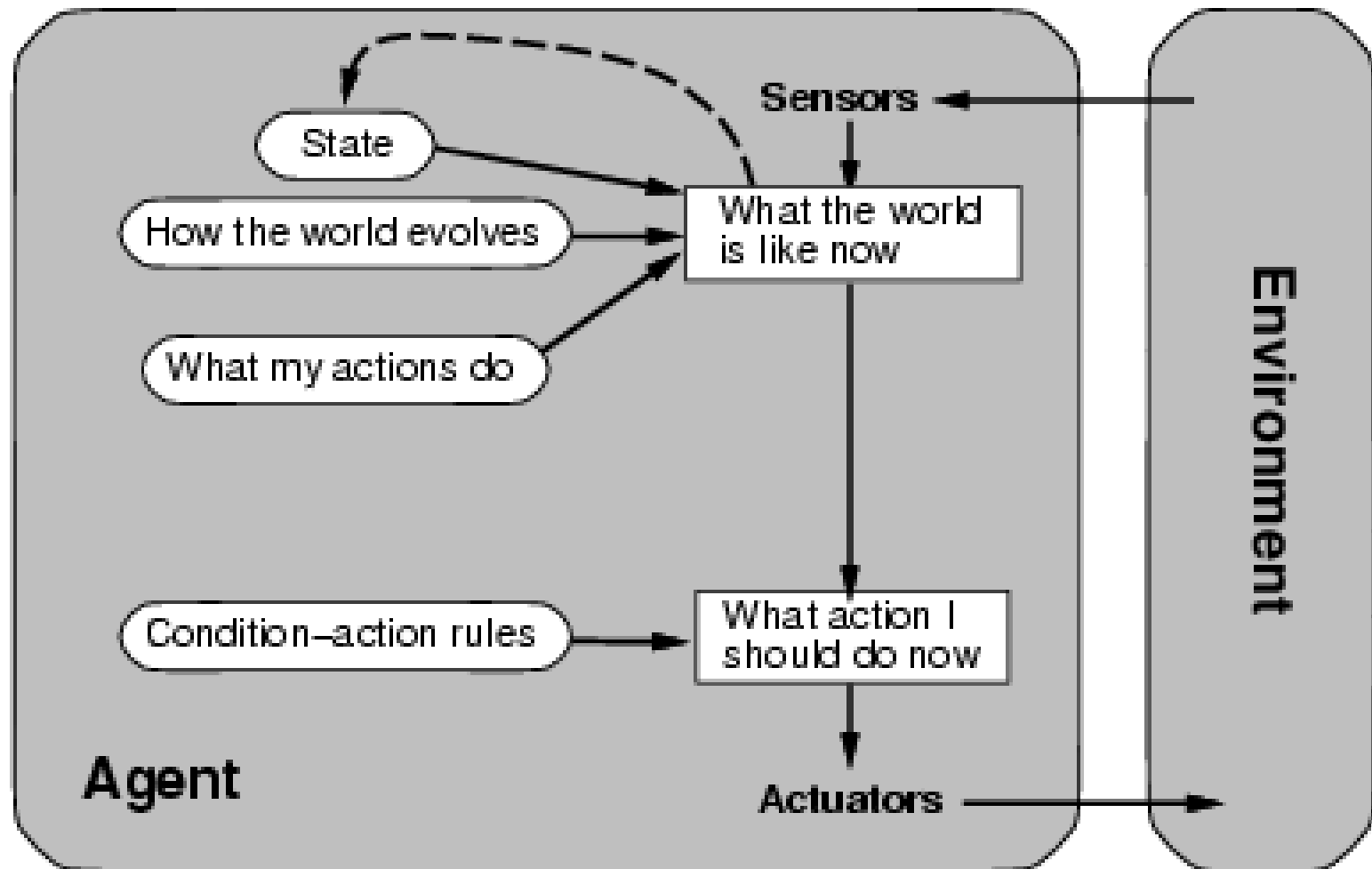
Programa del Agente Reactivo Simple

```
function SIMPLE-REFLEX-AGENT(percept) returns an action  
  persistent: rules, a set of condition–action rules  
  
  state  $\leftarrow$  INTERPRET-INPUT(percept)  
  rule  $\leftarrow$  RULE-MATCH(state, rules)  
  action  $\leftarrow$  rule.ACTION  
  return action
```

- El agente funciona apenas si el ambiente fuese completamente observable y la decisión correcta pudiese ser tomada basada apenas en la percepción actual.

Estructura de Agentes

Agente reactivo basado en modelo



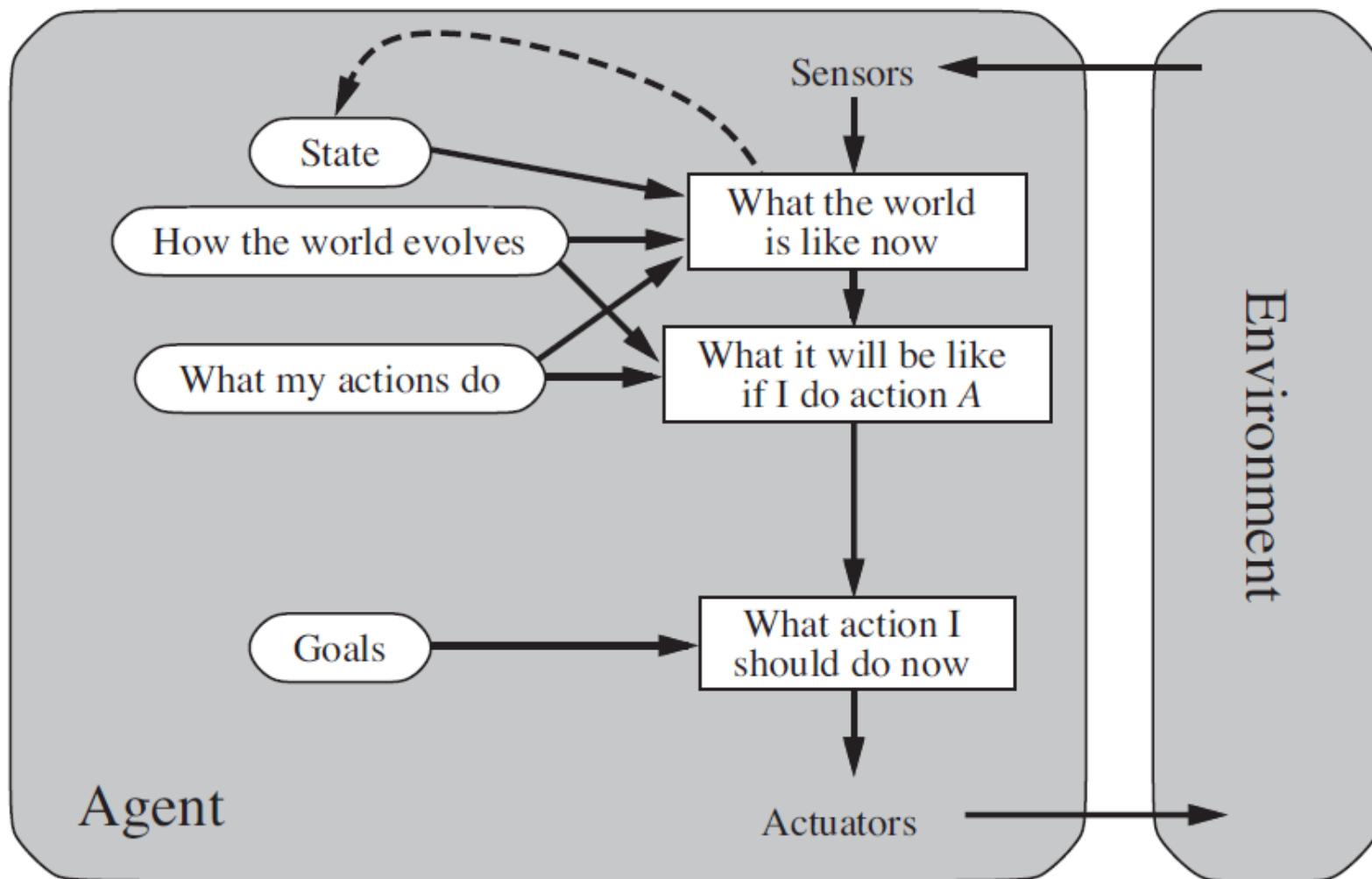
Programa del Agente Reactivo Basado en Modelo

```
function MODEL-BASED-REFLEX-AGENT(percept) returns an action  
  persistent: state, the agent's current conception of the world state  
               model, a description of how the next state depends on current state and action  
               rules, a set of condition–action rules  
               action, the most recent action, initially none  
  
  state ← UPDATE-STATE(state, action, percept, model)  
  rule ← RULE-MATCH(state, rules)  
  action ← rule.ACTION  
  return action
```

- Es efectivo en entornos parcialmente observables.

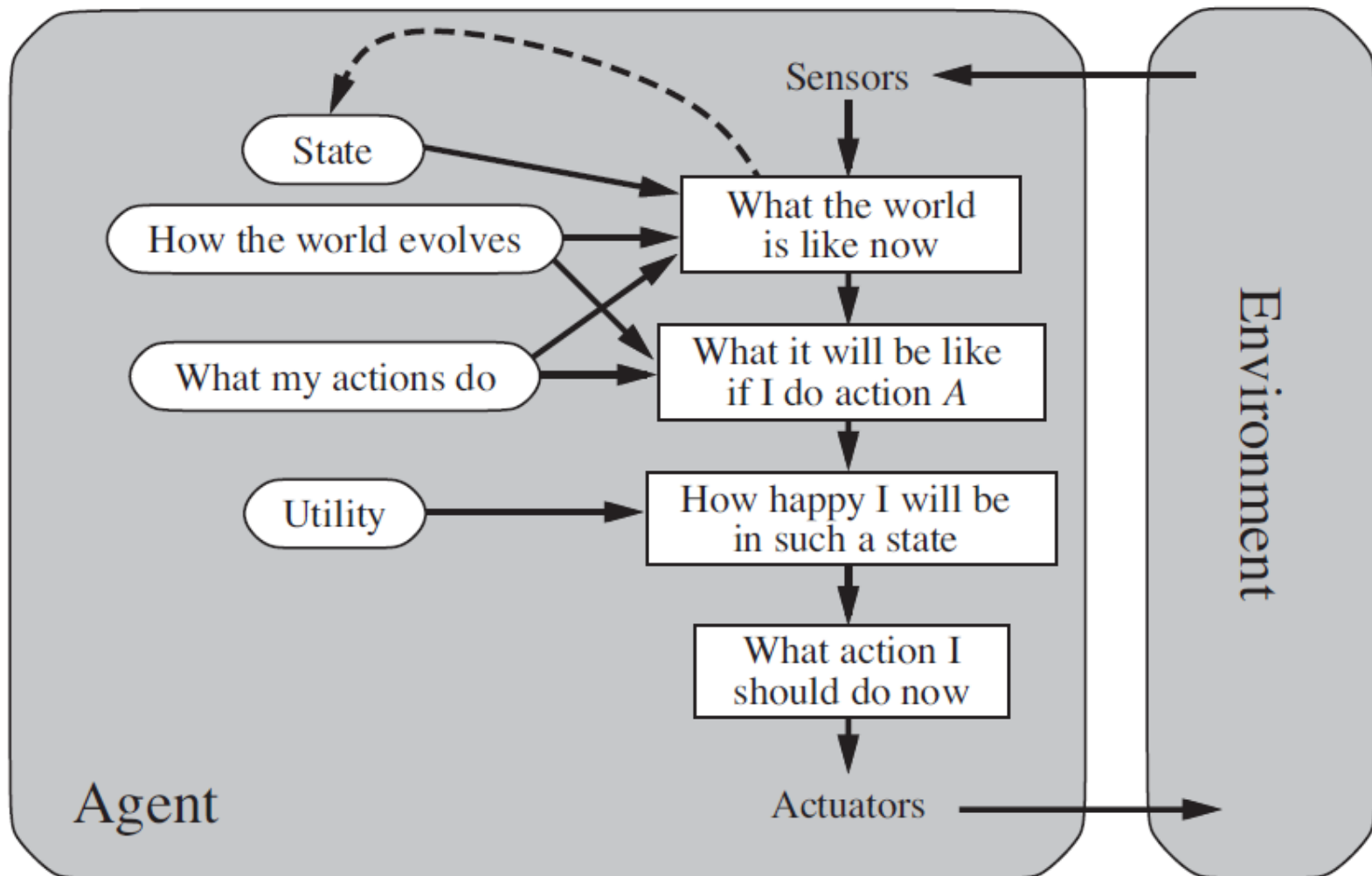
Estructura de Agentes

Agente basado en objetivo



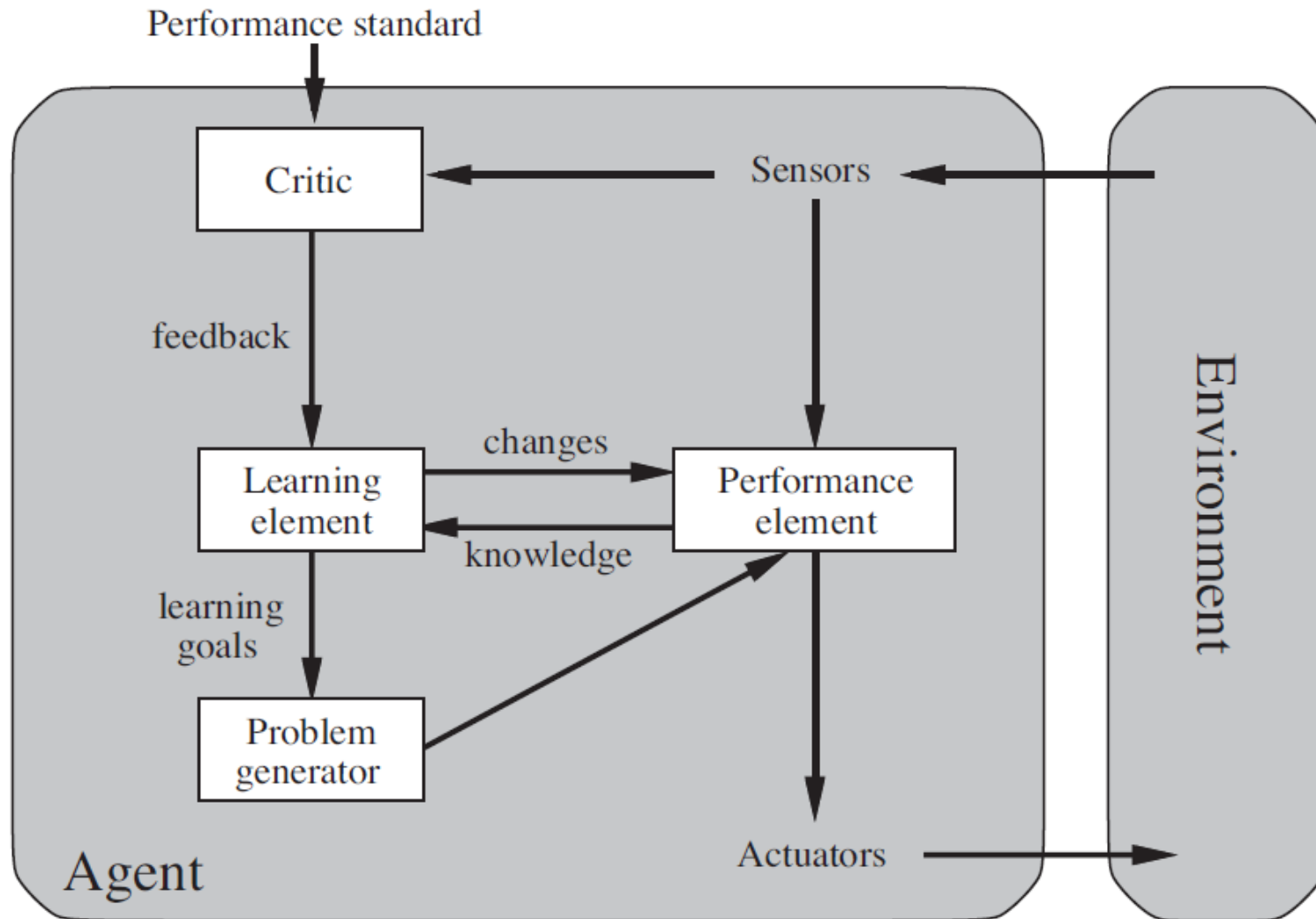
Estructura de Agentes

Agente basado en utilidad



Estructura de Agentes

Agentes que aprenden



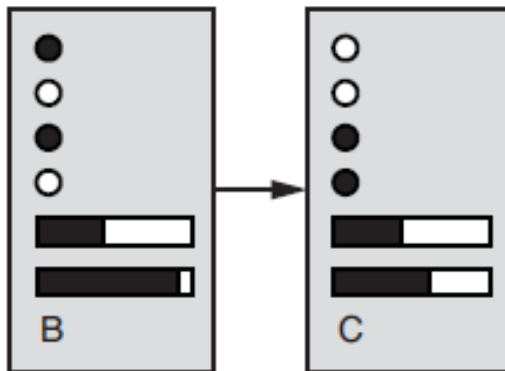
Estructura de Agentes

Tipos de Representación de Estados:



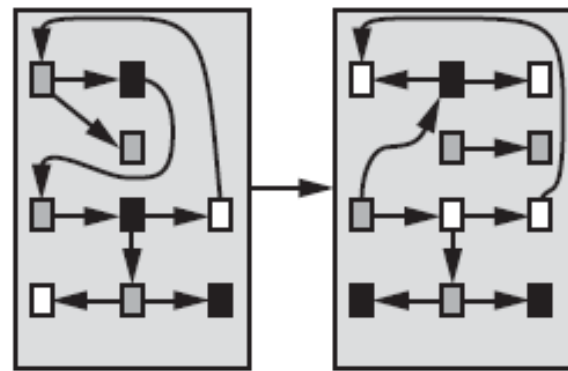
Atómica

Los estados son
cajas negras



Factorada

Los estados se
describen por atributos



Estructurada

Los estados son descritos
por la estructura interna de
sus componentes