

Los sistemas de **inteligencia artificial** son **sistemas software** (posiblemente también hardware), que dado un **objetivo complejo**, **actúan** en la **dimensión física** o **digital** al percibir su entorno a través de la adquisición de datos. Al **interpretar** y **razonar** sobre estos datos, **decide** la(s) mejor(es) acción(es) a tomar para lograr el objetivo dado. Los sistemas AI pueden usar **reglas simbólicas** o aprender un **modelo numérico**.

Llamamos **arquitecturas cognitivas** a aquellas que utilizan **símbolos** para representar el entorno. Se basan en **entidades computacionales**, pero esto no implica que la inteligencia deba provenir siempre de la computadora.

También especifica la **infraestructura** para un **sistema inteligente** que permanece constante en diferentes dominios y bases de conocimiento. Esta infraestructura incluye un compromiso con los **formalismos** para **representar el conocimiento**, las **memorias para almacenar el conocimiento** de este dominio y los **procesos que utilizan y cómo adquieren el conocimiento**.

Finalmente la investigación sobre la arquitectura cognitiva está ligada al llamado **modelado cognitivo**, en el sentido de que a menudo intentan explicar un **conjunto de comportamientos humanos**.

Un **agente** es un **sistema informático** que es capaz de **actuar de manera autónoma** en un **entorno específico** para cumplir con sus **objetivos de diseño**. Por lo general, es capaz de **percibir y actuar** sobre el entorno, que tiende a ser **complejo y dinámico**.

Mente, yo y sociedad de George H. Mead muestra un **análisis psicológico** a través del comportamiento y la interacción del yo de un individuo con la realidad. El comportamiento se desarrolla principalmente a través de **experiencias** y **encuentros sociológicos**. Estas experiencias conducen a comportamientos individuales que conforman los factores sociales que crean las comunicaciones en la sociedad. La comunicación se puede describir como la comprensión de los gestos de otra persona. Mead explica que la comunicación es un acto social porque requiere que dos o más personas interactúen. También explica que el yo es un proceso que el yo es un proceso social con comunicación entre el "yo" (la forma pura del yo) y el "yo" (la forma social del yo). **La mente humana surge únicamente a través de la experiencia social. Representa el proceso de pensamiento de la comunicación internalizada.**

La **mente humana** está compuesta por muchas neuronas, pero cada neurona actuando de manera individual no tiene mucho impacto, pero actuando de manera conjunta puede llegar a hacer cosas remarcables (como por ejemplo, una colonia de hormigas). Es decir, actuando individualmente sólo pueden llegar a realizar acciones simples, mientras que en conjunto pueden llevar a cabo acciones complejas.

La sociedad de la mente de Marvin Minsky, Marvin Minsky ve la mente humana como cualquier otro sistema cognitivo naturalmente evolucionado como un conjunto de procesos individualmente simples conocidos como agentes. Estas son las entidades de pensamiento fundamentales a partir de las cuales se constituye una mente, y juntos producen las muchas habilidades que atribuimos a la mente. La mente, según Minsky, es una sociedad constituida por la interacción de infinitud de diminutos agentes, los cuales carecen, a su vez, de toda actividad mental.

Tendencias que han marcado la historia de la informática:

1. **Ubicuidad**: la reducción continua en el costo de la capacidad informática ha hecho posible introducir potencia de procesamiento en lugar y dispositivos que antes eran inviables. A medida que se extiende la capacidad de procesamiento, la computación se vuelve omnipresente.
2. **Interconexión**: los sistemas informáticos de hoy en día ya no están solos, sino que están conectados en red en grandes sistemas distribuidos. Los datos en la nube.
3. **Inteligencia**: la complejidad de las tareas que somos capaces de automatizar y delegar en ordenadores ha ido creciendo de forma sostenida, hasta el punto de que podemos identificar esos sistemas como inteligentes.
4. **Delegación**: estamos dando cada vez más control a las computadoras, que hacen cada vez más tareas por nosotros.
5. **Orientación humana**: los programadores conceptualizan e implementan el software en términos de abstracciones de más alto nivel, más orientadas al ser humano.

Estas tendencias marcan la necesidad de construir sistemas informáticos que puedan actuar eficazmente en nuestro nombre, es decir, de forma independiente y que represente nuestros mejores intereses. También es importante el hecho de que puedan cooperar y llegar a acuerdos con otros sistemas que tienen intereses diferentes.

Inteligencia ambiental, dirigido a la entrega fluida de servicios y aplicaciones, se basa en la comunicación ubicua y las interfaces de usuario inteligentes. Requiere que los agentes puedan interactuar con muchos otros agentes en el entorno que los rodea para lograr sus objetivos.

Edge computing, tiene como objetivo acercar la computación y los datos a la fuente (usuario final). **Edge** (el más cercano al usuario, teléfonos inteligentes, sensores...), **Fog** (nodos fijos de baja latencia, torres de telefonía, enrutadores...) y **Cloud** (nubes públicas, privadas...).

Un **agente** es un sistema informático que es capaz de realizar acciones independientes en nombre de su usuario o propietario. Un **sistema multiagente** es aquel que consta de una serie de agentes, que interactúan entre sí. Para **interactuar con éxito**, requerirán la capacidad de **cooperar, coordinar y negociar** entre sí.

En el **diseño de agentes** intentan resolver el problema sobre cómo construir agentes capaces de actuar de manera independiente y autónoma.

Y en el **diseño de la sociedad** se intenta resolver el problema sobre cómo construir agentes que sean capaces de interactuar con otros agentes para llevar a cabo con éxito esas tareas delegadas.

El **campo de los sistemas multiagentes** está **influenciado** por muchos otros campos (filosofía, teoría de juegos, economía,...), esto puede ser tanto una fortaleza (infundir metodologías de otros campos) como una debilidad (muchos puntos de vista diferentes sobre cómo tratar el campo).

Propiedad de los agentes:

- **Autonomía:** capaz de actuar de forma independiente (control sobre su estado interno).
- **Flexibilidad:** es flexible si es reactivo (con capacidad de respuesta), proactivo (tomar la iniciativa) y social (interactuar con otros).
- **Reactividad:** un sistema reactivo es aquel que mantiene una interacción continua con su entorno y responde a los cambios que se producen en él.
- **Proactividad:** generar y tratar de alcanzar metas, no impulsado únicamente por el entorno, tomando la iniciativa y reconociendo oportunidades.
- **Habilidad social:** capacidad de interactuar con otros agentes a través de algún tipo de lenguaje de comunicación de agentes, y tal vez cooperar con otros.

Hay otras propiedades como: **movilidad** (capacidad de un agente para moverse por una red telefónica), **veracidad** (un agente no comunicará información falsa), **benevolencia** (no tienen objetivos contradictorios), **racionalidad** (actuará para lograr sus objetivos) y **aprendizaje** (mejoran el rendimiento con el tiempo).

Como queremos que nuestros agentes sean reactivos (respondiendo a las condiciones cambiantes de manera apropiada) y queremos que nuestros agentes sistemáticamente hacia objetivos a largo plazo (deliberación), esto puede suponer un desacuerdo entre sí.

Propiedades del entorno:

- **Accesible vs inaccesible:** **entorno accesible** es aquel en el que el agente puede obtener información completa, precisa y actualizada sobre el estado del entorno (**entorno totalmente observable**). **Entorno inaccesible** (**entorno parcialmente observables**). Cuanto más accesible sea el entorno, más fácil es construir agentes en él.
- **Determinista vs no determinista:** **entorno determinista** es aquel en el que cualquier acción tiene un único efecto garantizado: no hay incertidumbre sobre el estado después de la acción. Pero el mundo físico es considerado **no determinista**

(estocástico). Es más complicado diseñar agentes en entornos no deterministas. El entorno es **estratégico** si es determinista excepto por las acciones de otros agentes.

- **Episódico vs no episódico:** en un **entorno episódico**, el desempeño de un agente depende de una serie de episodios discretos, sin vínculos entre el desempeño de un agente en diferentes escenarios.
- **Estático vs dinámico:** **entorno estático** es aquel que se puede suponer que permanece sin cambios, excepto por la realización de acciones por parte del agente. Y un **entorno dinámico** es aquel en el que operan otros procesos y, por lo tanto, cambia de maneras que escapan del control del agente. El mundo físico es dinámico.
- **Discreto vs continuo:** un **entorno es discreto** si hay un número fijo y finito de acciones y percepciones sobre él. (por ejemplo, los movimientos del ajedrez). Los **entornos continuos** tienen un cierto nivel de desajuste con los sistemas informáticos.

Tipos de agentes:

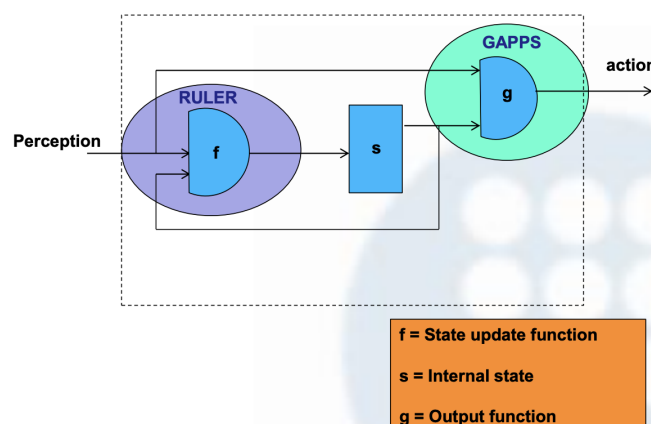
- **Agentes físicos:** interactúan con el mundo real (robots).
- **Agentes software:** **agentes de Internet** (búsqueda y extracción de información desde internet), **agentes colaborativos** (se coordinan con otros agentes para resolver una tarea común), **agentes de interfaz** (colaboran con un humano para resolver una tarea o para actuar en nombre del usuario) y **agentes SW móviles** (los que pueden mover de una computadora a otra).

Agentes puramente reactivos (sin estado interno), **agentes reactivos con estado interno**, **agentes deliberativos** (comportamiento orientado a objetivos) y **agentes híbridos** (combinan el comportamiento reactivo y deliberativo).

Arquitectura de agentes: propone una metodología particular para construir un agente autónomo.

Agentes puramente reactivos: algunos agentes deciden qué hacer sin hacer referencia al pasado: basan su toma de decisiones completamente en el presente.

Un agente se especifica en términos de dos componentes: **percepción (RULER)** y **acción (GAPPS)**.



RULER toma como entrada **3 componentes**:

- La semántica de las entradas del agente (“siempre que el bit 1 está encendido, está lloviendo”).
- Un conjunto de hechos estáticos (“siempre que llueve, el suelo está mojado”).
- Una especificación de las transiciones de estado del mundo (“si la tierra está mojada, permanece mojada hasta que sale el sol”).

GAPPS toma como entrada:

- Un conjunto de reglas de deducción de objetivos.
- Un objetivo de primer nivel.

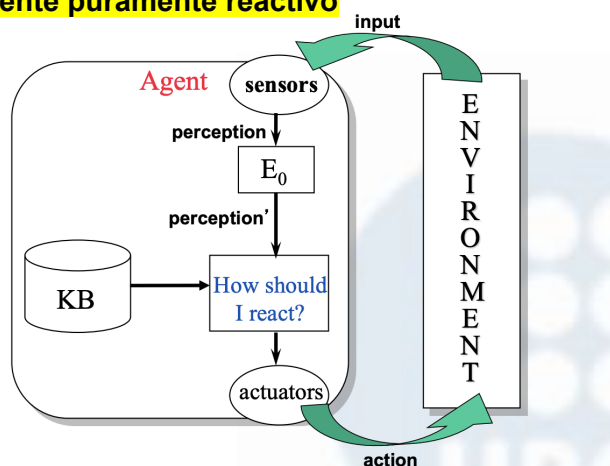
Ventajas de los agentes reactivos:

- Simplicidad de los agentes individuales.
- Flexibilidad y adaptabilidad.
- Manejabilidad computacional.
- Robustez contra fallas.
- Elegancia.

Desventajas de los agentes reactivos:

- Necesidad de información suficiente del entorno local.
- No está claro cómo incorporar información no local.
- Sin capacidades de planificación o replanificación a largo plazo.
- La aplicabilidad es limitada.
- Difícil de hacer agentes reactivos que aprenden.
- Las interacciones con el medio ambiente pueden tener consecuencias impredecibles a largo plazo.

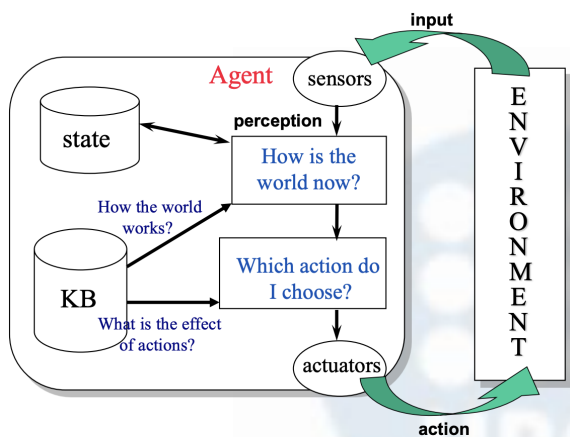
Agente puramente reactivo



function pra(percept) **returns** (action)
static rules

```
state ← interpret-input(percept)
rule ← rule-match(state, rules)
action ← rule-action[rule]
return action
```

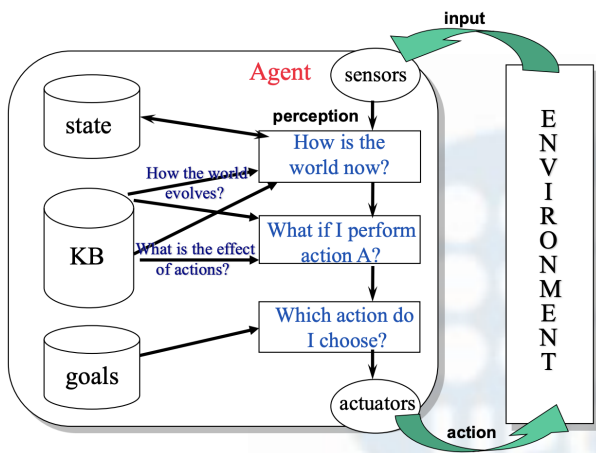
Agentes reactivos con estado interno



function reactive-agent-with-state(percept) **returns** action
static state ; a world description
 rules ; a set of, e.g., if-then rules
state ← **update-state**(state, percept)
rule ← **rule-match**(state, rules)
action ← **rule-action**[rule]
state ← **update-state**(state, action)
return action

Agentes deliberativos

Aquellos que tienen comportamientos orientados a objetivos. La gran desventaja es que es difícil tener una visión a largo plazo.



function reactive-agent-with-goals(percept) **returns** action
static state ; a world description
 rules ; a set of, e.g., if-then rules
 goals ; a list of goal states
state ← **update-state**(state, percept)
applicable-rules ← **rule-match**(state, rules)
possible-actions ← **rule-action**[rule]
action ← **goal-oriented-selection**[possible-actions]
state ← **update-state**(state, action)
return action

2 tipos de métodos de comunicación entre agentes dentro un sistema multiagente:

1. **Sistemas de pizarra:** los agentes comunican información a través de una estructura de datos común, accesible para todos.
2. **Paso de mensajes:** los agentes se comunican directamente por medio de mensajes, la plataforma de agentes suele actuar como enrutador de mensajes. Tienen que tener lenguaje de comunicación y protocolo de comunicación común.