



---

# **Ingeniería Conocimiento Grado Informática**

---

Introducción





---

# Contenido

---

1. Caracterización y perspectiva de agente.
2. Paradigmas principales.
3. Áreas de aplicación y ejemplos.



---

# 1. Caracterización y perspectiva de agente.

---



---

# Caracterización

---

The field of artificial intelligence (AI) is concerned with the design and analysis of autonomous agents. These are software systems and/or physical machines, with sensors and actuators, embodied for example within a robot or an autonomous spacecraft. An **intelligent system** has to perceive its environment, to act rationally towards its assigned tasks, to interact with other agents and with human beings.

*Computer Science*

*Curriculum 2008: An Interim Revision of CS 2001*

*December 2008*

*Association for Computing Machinery*

*IEEE Computer Society*



---

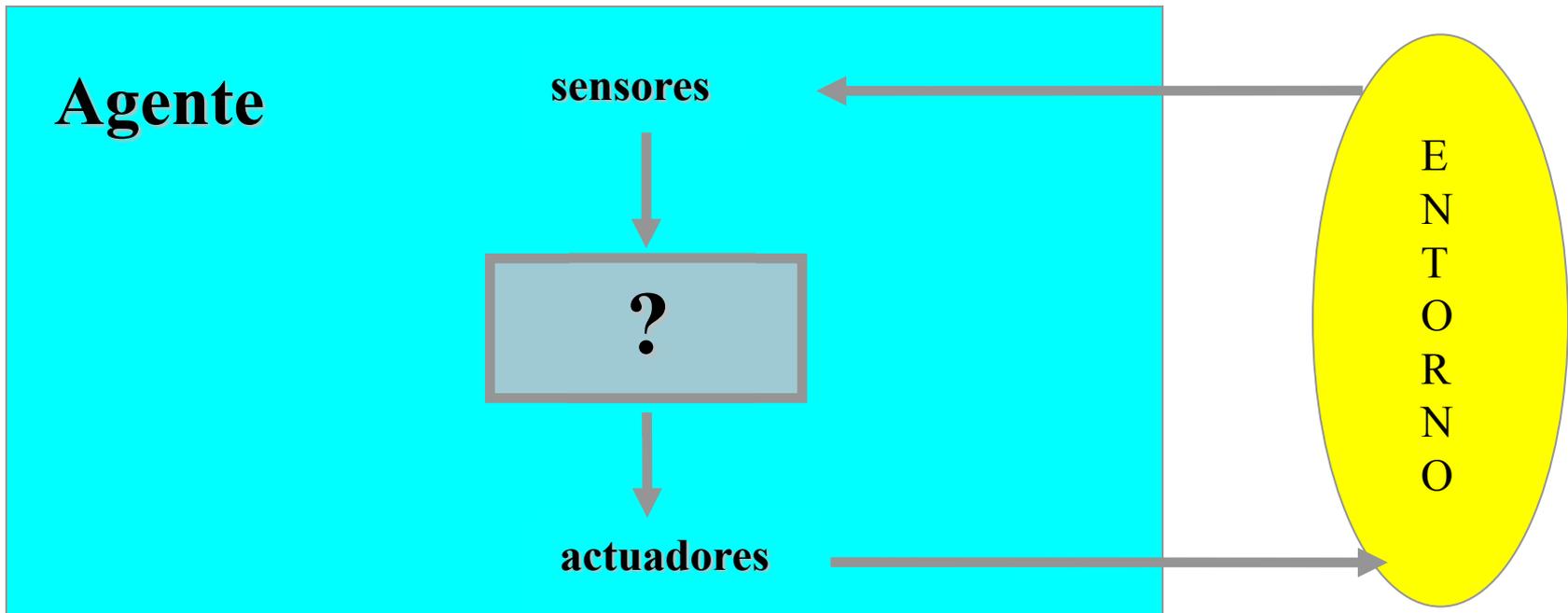
# ¿Qué es un sistema inteligente?

---

- Sistema Inteligente
  - Agente inteligente o, mejor, racional

# Concepto de Agente

- Entidad que:
  - percibe su entorno a través de sensores.
  - modifica el entorno mediante actuadores.

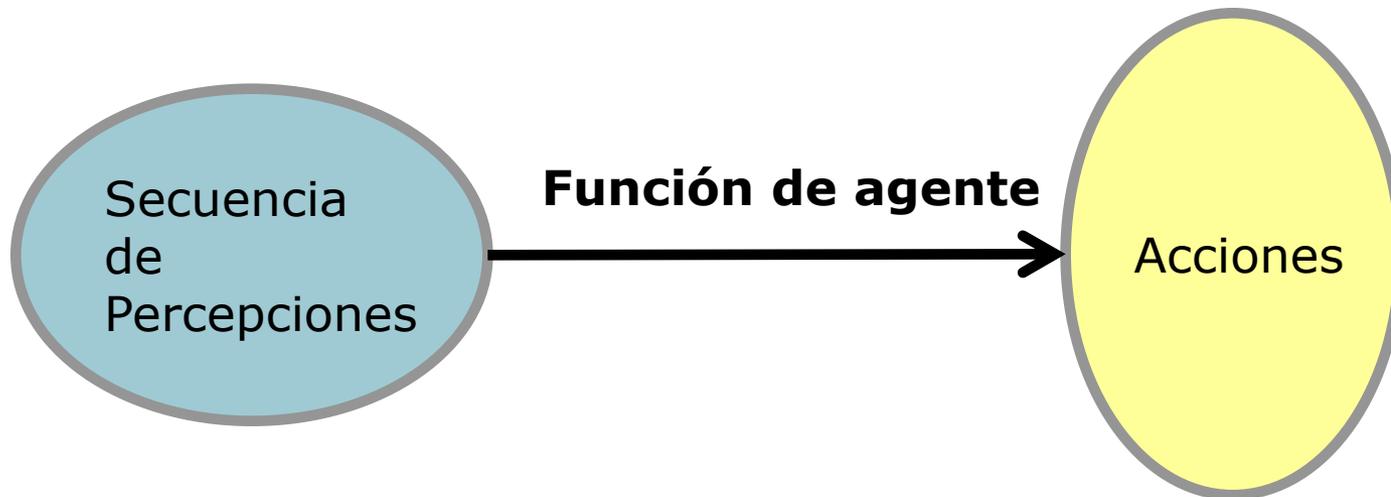


---

# Comportamiento de un agente

---

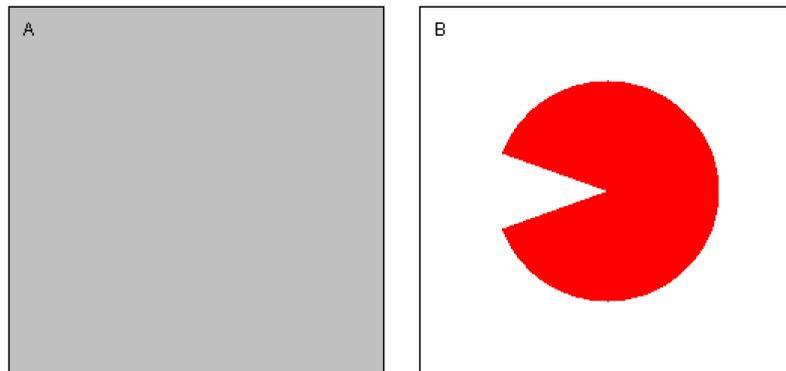
- Teóricamente: **función de agente**



---

# Mundo del aspirador

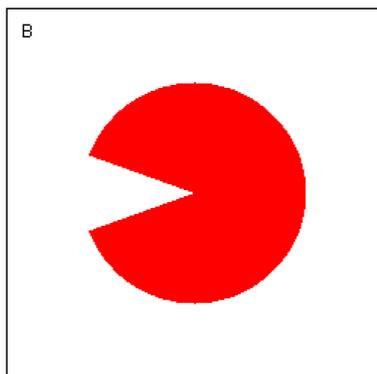
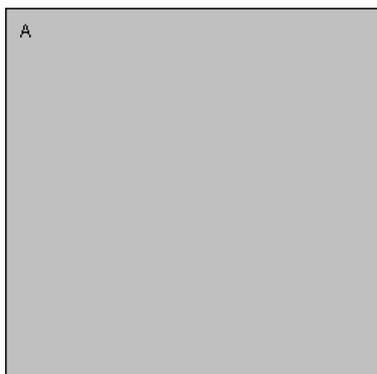
---



- Percepción: Situación, Suciedad en la casilla.
- Acciones: Derecha, Izquierda, Aspirar, NoOp
- Función de agente:

Si la casilla actual esta sucia entonces aspirar, sino ir a la otra casilla.

# Mundo del aspirador



| <b>Secuencia Percepciones</b>         | <b>Acción</b> |
|---------------------------------------|---------------|
| [A, Limpia]                           | Derecha       |
| [A, Sucia]                            | Aspirar       |
| [B, Limpia]                           | Izquierda     |
| [B, Sucia]                            | Aspirar       |
| [A, Limpia], [A, Limpia]              | Derecha       |
| [A, Limpia], [A, Sucia]               | Aspirar       |
| ...                                   | ...           |
| [A, Limpia], [A, Limpia], [A, Limpia] | Derecha       |
| [A, Limpia], [A, Limpia], [A, Sucia]  | Aspirar       |
| ...                                   | ...           |



---

# Hasta ahora

---

- Secuencia de Percepciones.
- Acciones.
- Función de Agente.
  
- Pero también: entorno.
  
- Entorno sencillo: discreto (continuo), estático (dinámico), determinista (estocástico), único agente (multi) , episódico (secuencial) pero parcialmente observable (totalmente).



---

# ¿Cómo determinar la calidad de un agente?

---

- Medida de rendimiento que cuantifique el grado de "éxito".
- Objetiva y externa.
- Desde el punto de vista del entorno.
- Mundo del aspirador:
  - Suciedad aspirada en turno de 8 horas.
  - Tiempo total en que el suelo está limpio.
  - Suelo limpio sin demasiado consumo eléctrico.

---



# Agente racional

---

- Agente racional ideal, Russell, Norvig
  - “Aquel que para cada posible secuencia de percepciones, realiza la acción que se espera que maximice su medida de rendimiento, basándose en la evidencia proporcionada por su secuencia de percepción y el conocimiento que el agente mantiene almacenado.”

---



# ¿Es el agente aspirador racional?

---

- Depende...
- Suponer:
- Entorno:
  - Geometría conocida, distribución inicial de suciedad desconocida.
  - Aspirar limpia una casilla y permanece limpia.
  - Derecha e Izquierda mueven al agente a la derecha y a la izquierda, salvo si le sacan del entorno (permanece).
- Percepciones: situación y suciedad (correctas).
- Acciones: Derecha, Izquierda, Aspirar, NoOp.
- Función de agente: Si la casilla actual esta sucia entonces aspirar, sino ir a la otra casilla.
- Rendimiento: 10 puntos por casilla limpia por instante de tiempo.



---

# ¡Sí, es racional!

---

- Ningún otro agente obtendrá una mejor rendimiento esperado.
- ¿Os parece que el comportamiento es defectuoso?
- Entonces, definir otra medida de rendimiento.
- Suponer que no queremos al agente moviéndose en un entorno limpio.



---

# Modificar la medida de rendimiento

---

- Modificar ligeramente la medida anterior:  
10 puntos por casilla limpia por instante de tiempo.



---

# Nueva medida de rendimiento

---

- 10 puntos por casilla limpia por instante de tiempo
- -1 punto por cada movimiento.
  
- Para maximizar el rendimiento esperado, tenemos que cambiar la Función de agente:
  1. Si la secuencia de percepciones incluye [A, Limpia] y [B, Limpia], en cualquier orden y posición, entonces seleccionar NoOp.
  2. Si la casilla actual esta sucia entonces Aspirar, sino ir a la otra casilla.



---

# ***Racionalidad no es perfección y requiere aprendizaje***

---

- Un comportamiento perfecto requiera saber por adelantado el resultado de las acciones del agente: sólo se pide maximizar el rendimiento esperado.
- Excepto para entornos sencillos, invariables, el conocimiento inicial del agente no puede garantizar maximizar el rendimiento a largo plazo.
- Racionalidad normalmente requiere aprendizaje para proporcionar autonomía al agente.



---

# Ejemplos de agentes y descripción REAS

---

| <b>Tipo de Agente</b>                  | <b>Medida de Rendimiento</b>                                  | <b>Entorno</b>                               | <b>Actuadores</b>                                    | <b>Sensores</b>                         |
|--|---|--|--|---|
| Robot para la selección de componentes | Porcentaje de componentes clasificados en los cubos correctos | Cinta transportadora con componentes, cubos  | Brazo y mano articulados                             | Cámara, sensor angular                  |
| Controlador de una refinería           | Maximizar la pureza, producción y seguridad                   | Refinería, operadores                        | Válvulas, bombas, calentadores, monitores            | Temperatura, presión, sensores químicos |
| Tutor de Inglés interactivo            | Maximizar la puntuación de los estudiantes en los exámenes    | Conjunto de estudiantes, agencia examinadora | Visualizar los ejercicios, sugerencias, correcciones | Teclado de entrada                      |



---

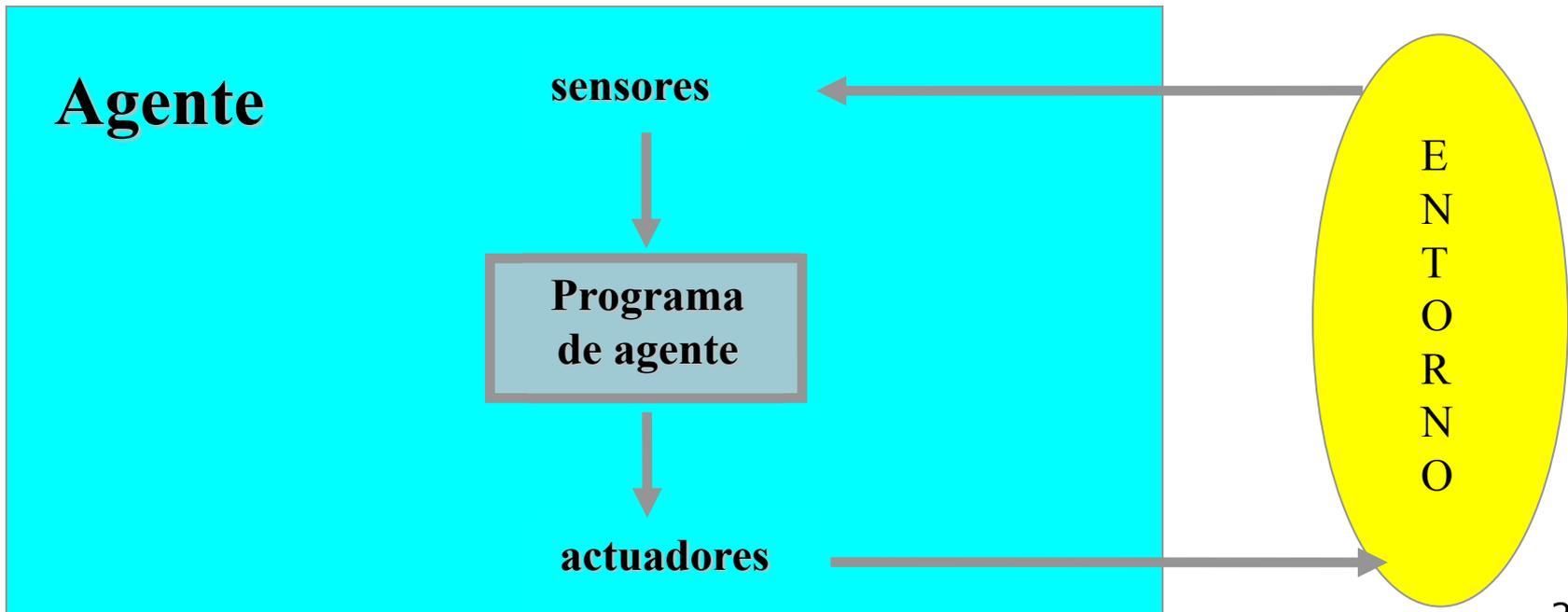
# Estructura de agente

---

- La Función de agente no se implementa, pues no se suele conocer.
- Se implementa: **Programa de agente.**
- El Programa de agente utiliza solo la percepción actual del entorno.
- Utiliza la percepción actual del entorno y el conocimiento y memoria que pueda tener para seleccionar la acción actual.

# Concepto de Agente y Programa de Agente

- Entidad que
  - percibe su entorno a través de sensores
  - modifica el entorno mediante actuadores





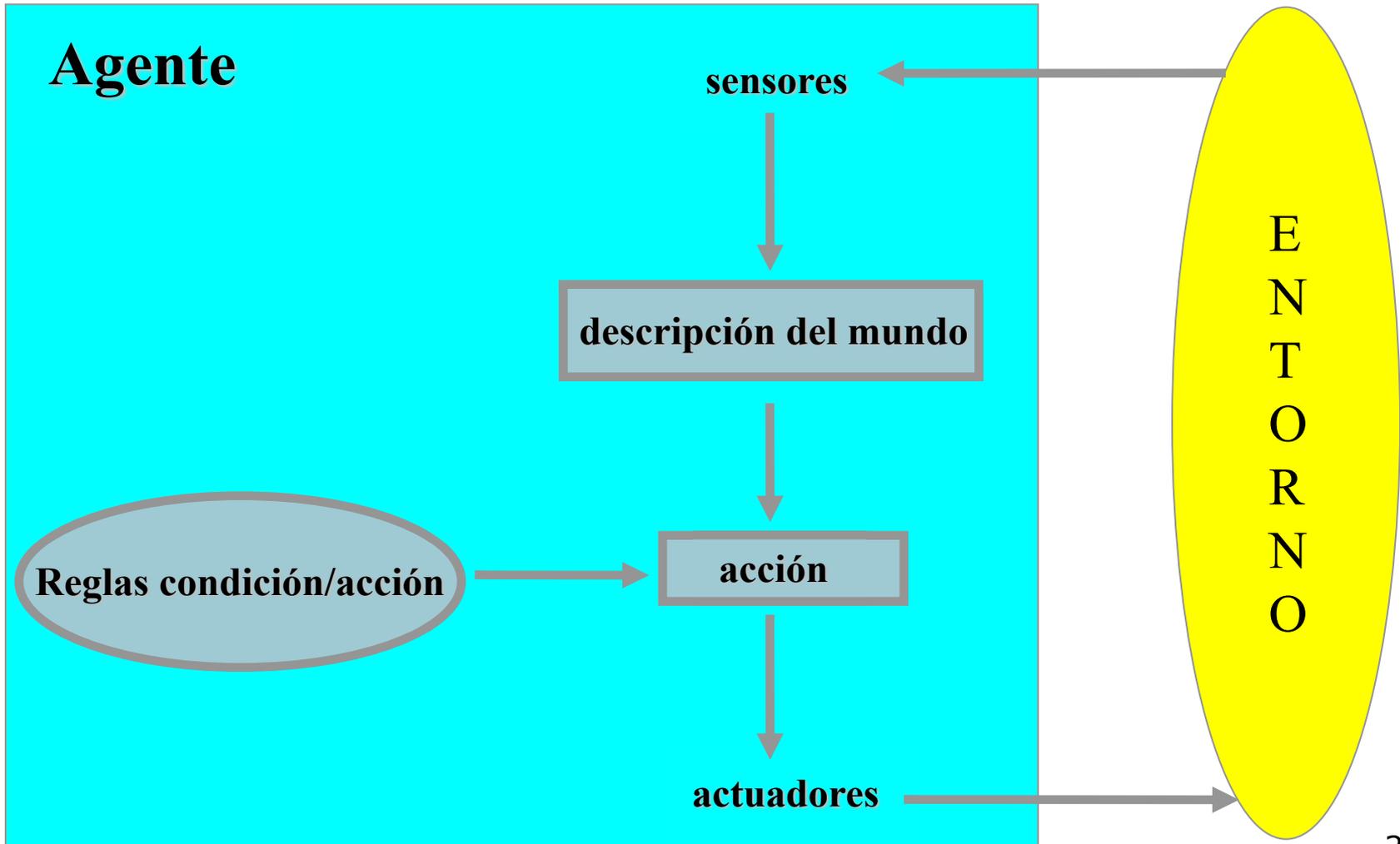
---

# Estructura de agente

---

- La mayoría de los principios subyacentes a los sistemas inteligentes se pueden describir con 5 tipos básicos de Programas de agente.
- Agente reactivo simple.
- Agente reactivo basado en modelos.
- Agente basado en metas.
- Agente basado en utilidad.
- Agentes que aprenden

# Agente reactivo simple





---

# Agente reactivo simple

---

- Suficiente si la acción actual se puede tomar a partir de la percepción actual.
- Muchas tareas de clasificación:
  - Riesgo crediticio
  - Etiquetado de imágenes
  - Comportamiento anómalo en aeropuertos
  - Sistemas de diagnosis de fallos sencillo



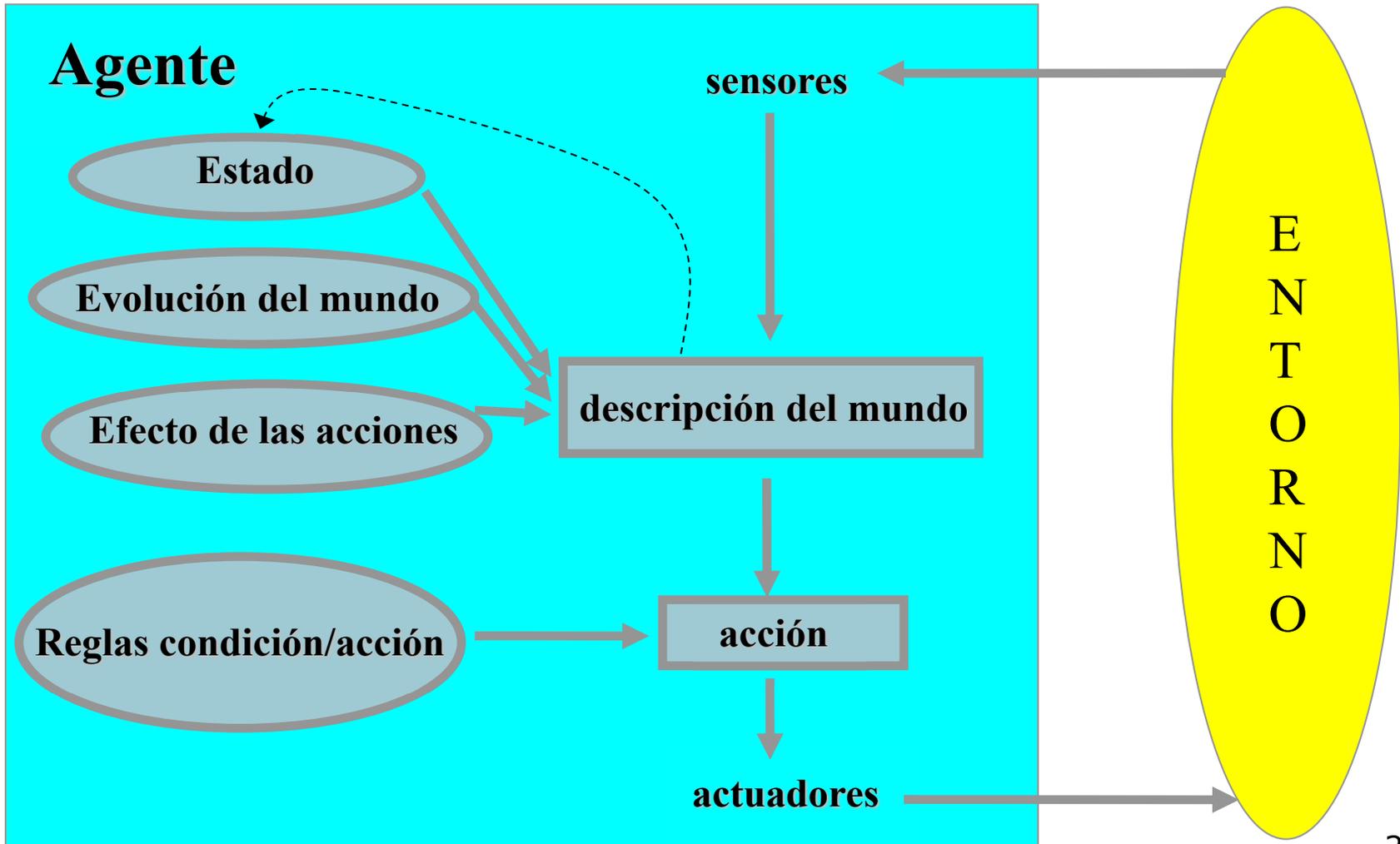
---

# Pero el agente reactivo simple es muy limitado

---

- Entorno no totalmente observable, secuencial
- Mundo aspirador:
  - 10 puntos por casilla limpia por instante de tiempo
  - -1 punto por cada movimiento.
- Un agente reactivo simple no puede ser racional con esta función de rendimiento

# Agente reactivo basado en modelos





---

# Capacidad agente reactivo basado en modelos

---

- Puede generar comportamiento racional en numerosas tareas analíticas
- Sistemas basados en conocimiento para tareas analíticas:
  - Clasificación
  - Evaluación (Assessment)
  - Monitorización
  - Diagnósis



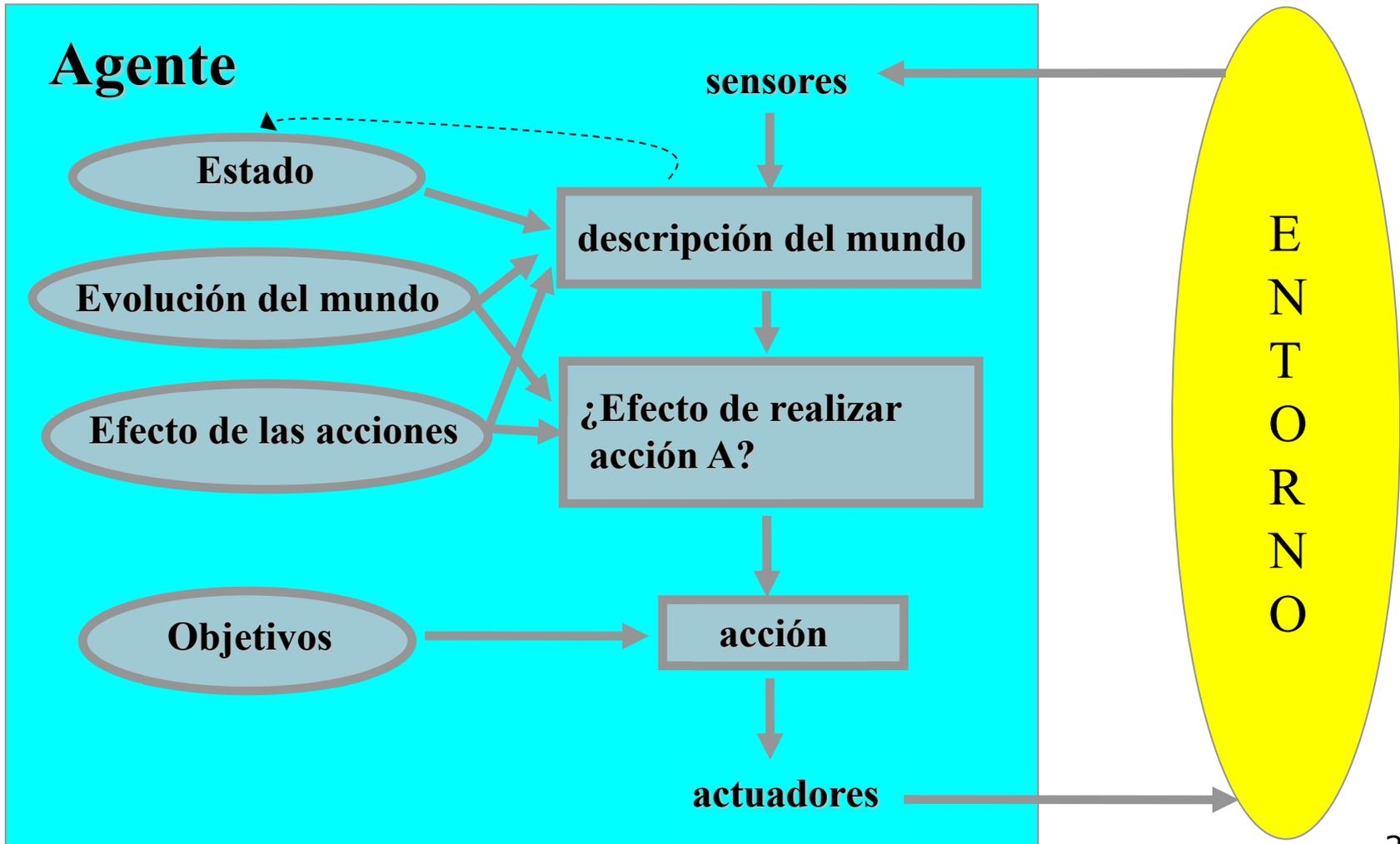
---

# Limitaciones agente reactivo basado en modelos

---

- A veces, el agente necesita considerar su meta actual para seleccionar la mejor acción
- Agente aspirador en un edificio docente.
- Dos metas:
  - Limpieza
  - Silencio
- Aproximación sencilla:
  - Horas de clase: silencio.
  - Fuera del horario de docencia: limpieza

# Agente basado en objetivos





---

# Selección de acciones basada en metas

---

- Puede ser sencilla, como el agente aspirador en el edificio de docencia.
- Generalmente, el agente tiene que considerar secuencias de acciones.
- Búsqueda: rutas, juegos
- Planificación: Horarios de clases, asignación de tiempos a tareas (Scheduling)



---

# Limitaciones de los agentes basados en metas

---

- A veces el agente tiene que razonar sobre las metas.
- Porque hay metas contrapuestas (compromiso).
  - Agente aspirador en hospital: limpieza *frente* a silencio.
- Porque hay incertidumbre en la obtención de las metas: mejor jugada poker



---

## 2. Paradigmas principales

---



---

# Sistemas Inteligentes basados en conocimiento

---

- No sólo técnicas de búsqueda
  - De interés en numerosos entornos (juegos, optimización de rutas, etc...)
- Incluyen conocimiento, implícito o explícito, sobre el problema a resolver

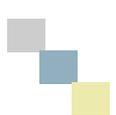


---

# Paradigmas principales (basados en conocimiento)

---

- Sistemas basados en conocimiento (sistemas expertos)
- Razonamiento basado en casos
- Métodos de aprendizaje y minería de datos
- Razonamiento basado en modelos



---

# Sistemas basado en conocimiento (sistemas expertos)

---

- Identificación y explotación del conocimiento *humano* para la resolución de problemas
- Suposiciones básicas (sistemas expertos)
  - El conocimiento proviene de la experiencia
  - Los expertos codifican el conocimiento mediante asociaciones (heurística)
  - El conocimiento se puede extraer del experto y codificar mediante un Lenguaje de Representación de Conocimiento



---

# Reglas de producción

---

- Estructura
  - SI antecedente ENTONCES consecuente
- Ejemplo

SI las diferencias de temperaturas son anormalmente altas Y las presiones se mantienen aproximadamente constantes ENTONCES sospechar fallo en sensores de temperatura



---

# Ejemplo de MYCIN

---

- Diagnósis y terapia de enfermedades infecciosas
- Base de conocimiento: Reglas de Producción
  - if (1) the stain of the organism is gram-negative
  - (2) the morphology of the organisms is coccus
  - (3) the growth configuration of the organism is chainsthen there is a suggestive evidence (0.7)  
that the identity of the organisms is streptococcus
- Control: encadenamiento hacia atrás, meta-reglas
- Razonamiento aproximado: factores de certeza



---

# Aplicaciones SE

---

- Numerosos dominios de aplicación:
  - Medicina, Mecánica, Electrónica, Control de Procesos, Aeronáutica, ...
- Numerosas tareas basadas en conocimiento:
  - Diagnósis, configuración, clasificación, ...
- Numerosos sistemas:
  - Representativos: MYCIN, INTERNIST, R1/XCONF...
  - Actuales: Expert systems with applications (An international Journal)



---

# Ventajas aproximación SE

---

- Aproximación bien establecida
  - Metodologías (Ingeniería de conocimiento, Ontologías), sistemas
- Adecuada si
  - Suficiente experiencia
  - No hay otras fuentes de conocimiento
  - Suficientes observaciones
  - El sistema permanece estable



---

# Inconvenientes aproximación SE

---

- Relacionadas con la experiencia
  - Dificultad tarea adquisición conocimiento
  - Disponibilidad experiencia/expertos
  - **Dependencia del dispositivo**
- Relacionado con el método de solución
  - **Situaciones no previstas**
  - Combinación de soluciones(ej: fallos múltiples)
  - Fragilidad
- Ingeniería de software
  - Obtención del conocimiento
  - Reutilización de conocimiento (tareas, dispositivos)
  - **Mantenimiento** de (la consistencia de ) la base de conocimiento



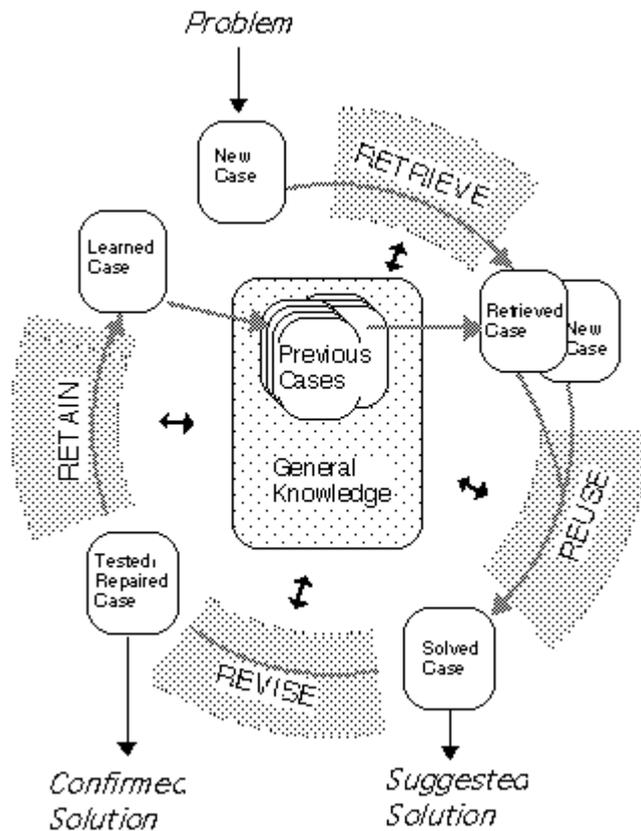
---

# Razonamiento basado en casos

---

- Caso: descripción de un problema y su solución
- Solución nuevo problema
  - Buscar caso(s) con descripción más similar
  - Adaptar solución(es) y revisar resultado
  - Almacenar nuevo caso si interesante

# Ciclo CBR



- Recuperar (**retrieve**) los casos más parecidos
- Reutilizar (**reuse**) la solución propuesta en los casos para tratar de resolver el problema
- Revisar (**revise**) la solución propuesta
- Almacenar (**retain**) la nueva solución como parte de un nuevo caso



---

# Ventajas/inconvenientes aproximación CBR

---

- Ventajas
  - No requiere representación explícita del conocimiento
  - Capacidad de aprendizaje
  - Adecuado si no se dispone de modelo explícito
- Inconvenientes
  - Necesidad de casos previos
  - Limitaciones de la etapa de REVISIÓN

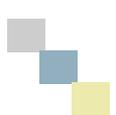


---

# Aprendizaje y Minería de datos

---

- Permiten inferir conocimiento nuevo
  - Descripción de conceptos a partir de ejemplos etiquetados (clasificación)
  - Descubrimiento de conceptos a partir de ejemplos no etiquetados (clustering)
  - Descubrimiento de regularidades en datos
  - Mejora de la eficiencia



---

# Ventajas/inconvenientes paradigma de aprendizaje

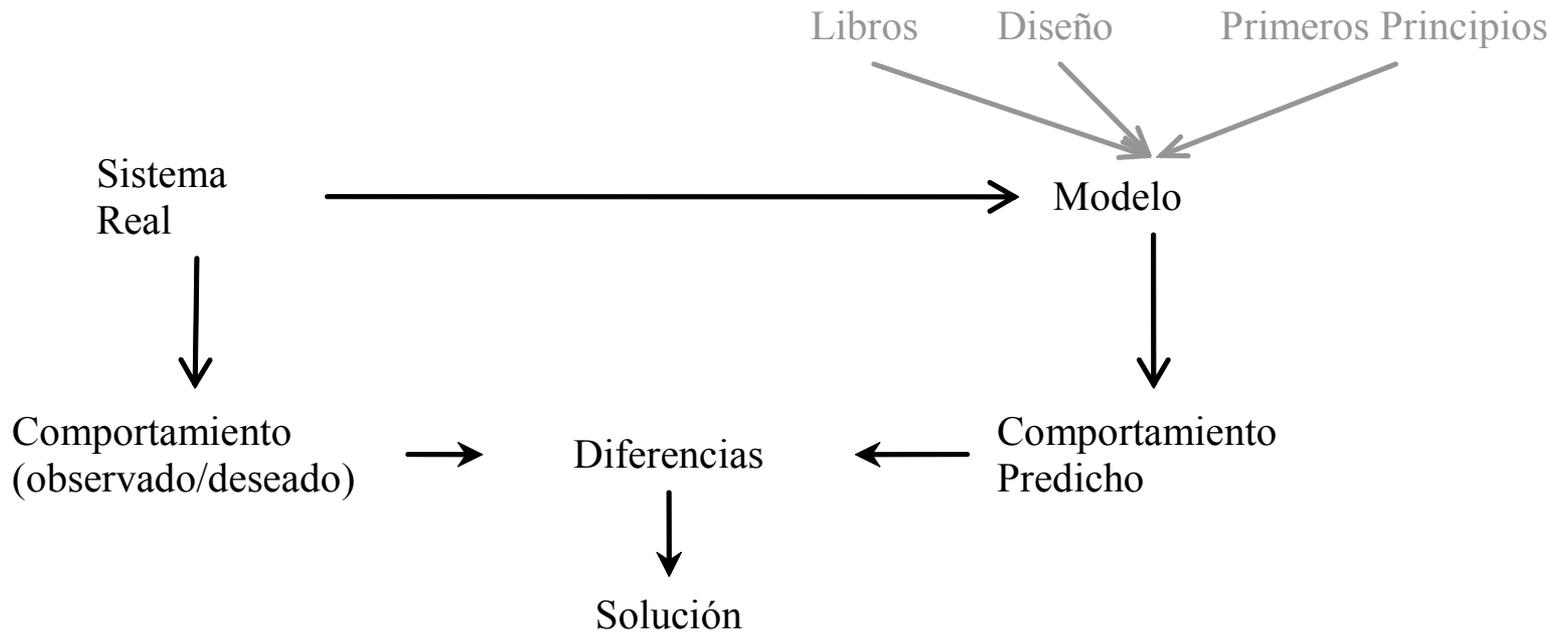
---

- Ventajas:
  - Permiten generar automáticamente conocimiento a partir de datos
  - Adecuado si no se dispone de modelo explícito
- Inconvenientes
  - Necesidad de disponer de datos
  - Habitualmente de utilidad *sólo* en alguna parte del proceso de solución

---

# Razonamiento basado en modelos

---



---



# Razonamiento basado en modelos

---

- Conocimiento: modelos (estructura, comportamiento) de los componentes del sistema
- Razonamiento: proceso de manipulación de los modelos hasta obtener solución del problema



---

# Ventajas RBM

---

- Independiente de la experiencia
  - Aplicable a dispositivos nuevos
- Independencia del dispositivo
  - Problema de las variantes
- Soluciones complejas (ej: Fallos múltiples)
- Sólido y completo
  - Respecto a los modelos
- Mantenimiento y reutilización del conocimiento
  - Biblioteca de modelos (disponibles desde el diseño)



---

# Inconvenientes RBM

---

- Dificultad de obtención de los modelos
  - Procesos poco conocidos
  - Sistemas con numerosos componentes
  - Comportamientos complejos: dinámica, no linealidades, rango de validez de modelo...
- Mayor carga computacional del proceso de razonamiento
  - Limitaciones aplicaciones en tiempo real



---

# 3. Áreas de aplicación y ejemplos

---



---

# Áreas de aplicación

---

- Planificación y Scheduling
- Configuración y diseño
- Diagnósis
- Control
- Visión
- Tecnologías del habla
- Robótica
- Lenguaje natural
- Análisis de transacciones
- Industria informática actual
  - navegadores, buscadores, gestor e-mail, reconocimiento matrículas aparcamientos, compras on-line ...



---

# Áreas de aplicación

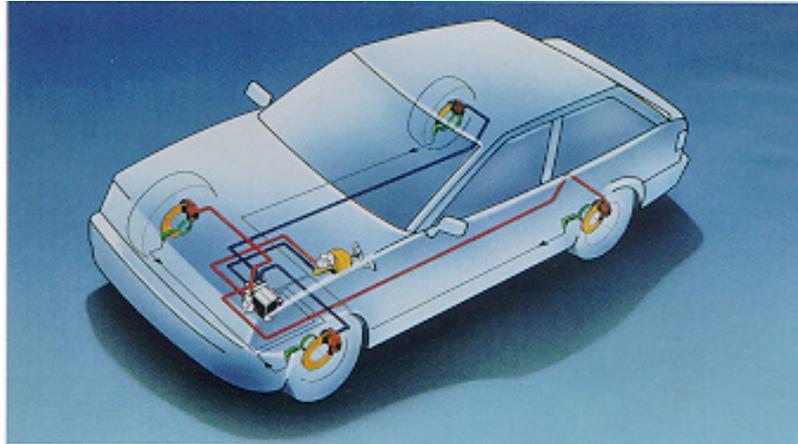
---

- Planificación y Scheduling
- Configuración y diseño
- **Diagnosis**
- Control
- Visión
- Tecnologías del habla
- Robótica
- Lenguaje natural
- Análisis de transacciones
- Industria informática actual
  - navegadores, buscadores, gestor e-mail, reconocimiento matrículas aparcamientos, compras on-line ...

---

# Industrial del automóvil

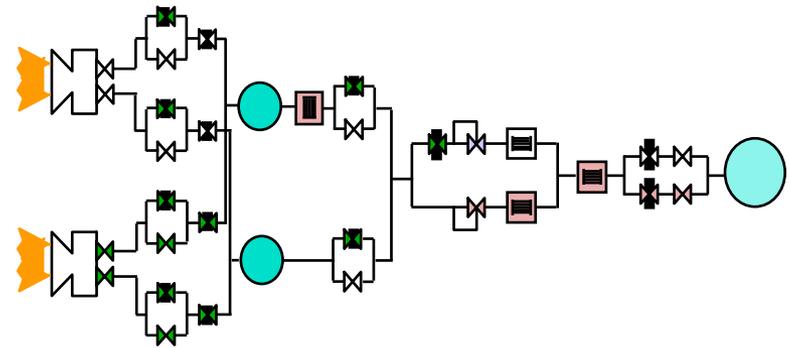
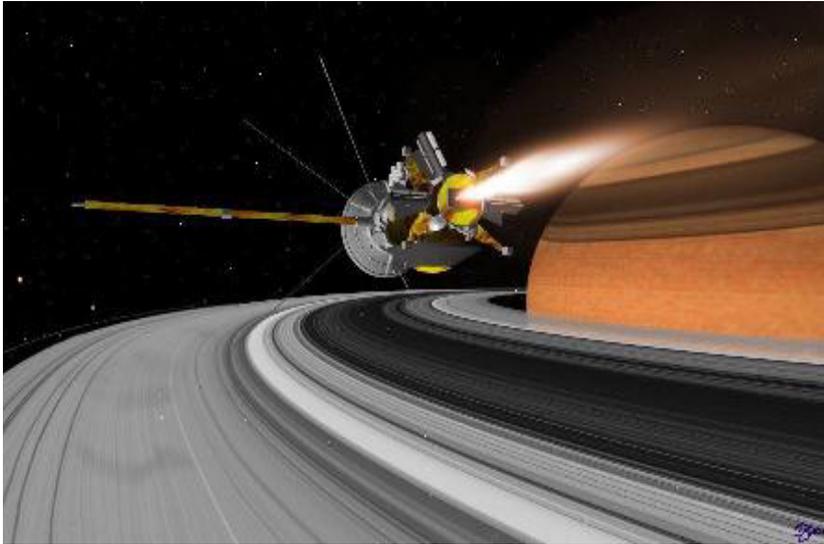
---



- Diagnóstico a bordo
- Diagnóstico en el taller
- Elaboración de manuales
- Diagnóstico, mantenimiento preventivo

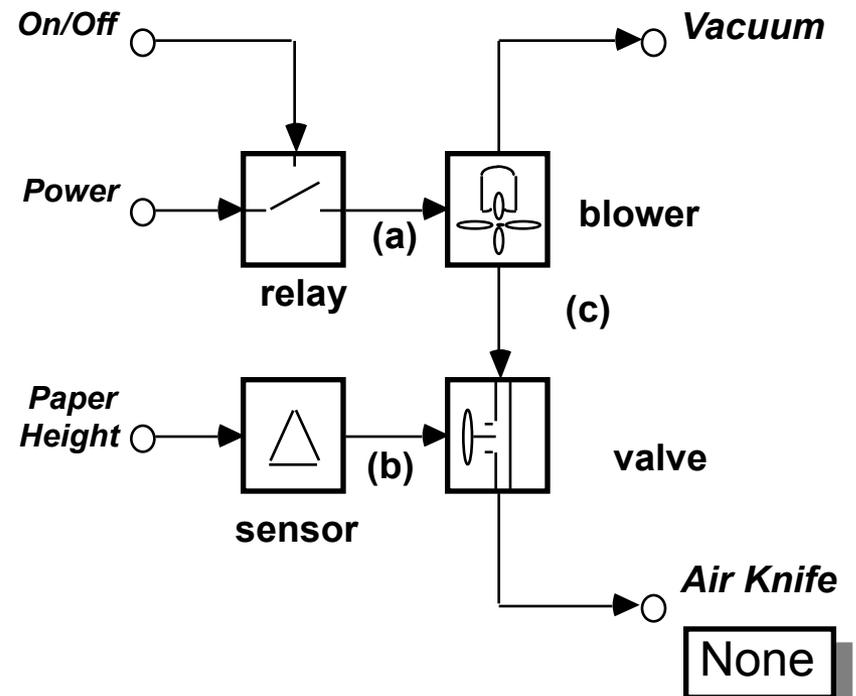
- De gran interés por
  - Seguridad
  - Medio ambiental
  - Económicos
- Varios proyectos Europa, USA

# Industria Aeroespacial



- Sistemas de Monitorización de las funciones básicas de naves espaciales (Health Management Systems)
  - subsistemas de propulsión, guiado, “de vida”...
- Para detección, localización y reconfiguración
  - Satélites, lanzaderas...
  - ESA (European Space Agency), NASA (National Space Agency), Deimos Space, IBERESPACIO, GMV Aerospaziale

# Localización de fallos en copiadoras



# Supervisión de procesos industriales



## Aporta

- Atención continuada
- Seguridad
- Calidad homogénea

