

Agentes Inteligentes

Modelos y Arquitecturas de Agentes

Juan Pavón Mestras

Dep. de Ingeniería del Software e Inteligencia Artificial



<http://grasia.fdi.ucm.es>

¿Cómo se diseña un agente?

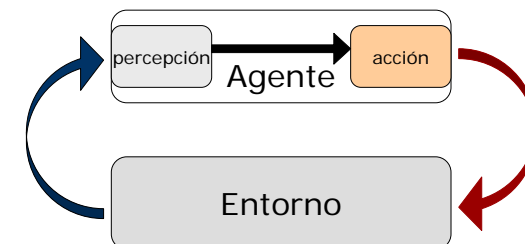
- Considerando el agente como entidad que interactúa con su entorno el diseño de un agente requiere estudiar:
 - Cómo percibir el entorno
 - Cómo representar el entorno
 - Cómo definir los actuadores
- Utilizando la definición de Newell:
 - Cómo representar los objetivos del agente
 - Cómo describir la toma de decisiones del agente
 - Cómo representar el conocimiento
- Y si consideramos un SMA, según Ferber, los mismos elementos de antes y además:
 - Leyes que controlan el entorno
 - Objetos ubicados
 - Coordinación de los agentes
 - Acciones permitidas

La tecnología se asienta

- El estudio de estos elementos ha derivado en
 - Arquitecturas
 - Desde la experimentación en la construcción de sistemas
 - Lenguajes
 - Desde el estudio teórico de los agentes, principalmente con lógicas modales
- El avance en la experimentación de diseño de sistemas ha progresado hacia soluciones más orientadas a la industria
 - Plataformas de desarrollo de agentes
 - Arquitecturas reusables
 - Entornos de desarrollo
 - Metodologías

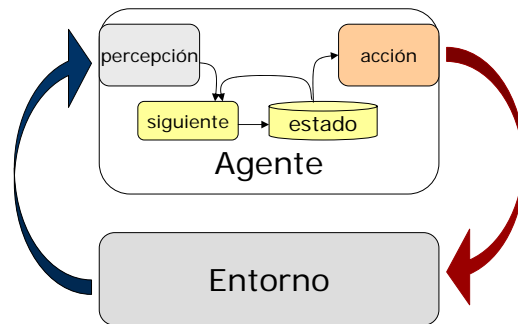
Agentes reactivos

- Modelo de agentes puramente reactivos
 - El proceso del agente es un ciclo percepción-acción (estímulo/respuesta)
 - Reacciona a la evolución del entorno
 - No hay una representación explícita del entorno, de los otros agentes, sus capacidades, etc.
 - Las decisiones no tienen en cuenta ni el pasado (no hay historia) ni el futuro (no hay planificación)



Agentes reactivos

- Agentes reactivos que mantienen su estado interno
 - Deciden la acción a realizar teniendo en cuenta su historia de interacciones con el entorno
 - Secuencia de estados del entorno o secuencia de percepciones
 - Se guarda como estado interno del agente



UCM 2006-07

Modelos y arquitecturas de agentes

5

Agentes reactivos

- Ejemplo de ciclo de ejecución de un agente reactivo
 - Reglas: situación-acción
 - Conjunto de percepciones

```

while (true) {
    estado = interpretar_entrada (percepcion);
    regla = correspondencia (estado, Reglas);
    ejecutar (regla, accion);
}
    
```

UCM 2006-07

Modelos y arquitecturas de agentes

6

Agentes reactivos (subsunción)

- Arquitectura de subsunción

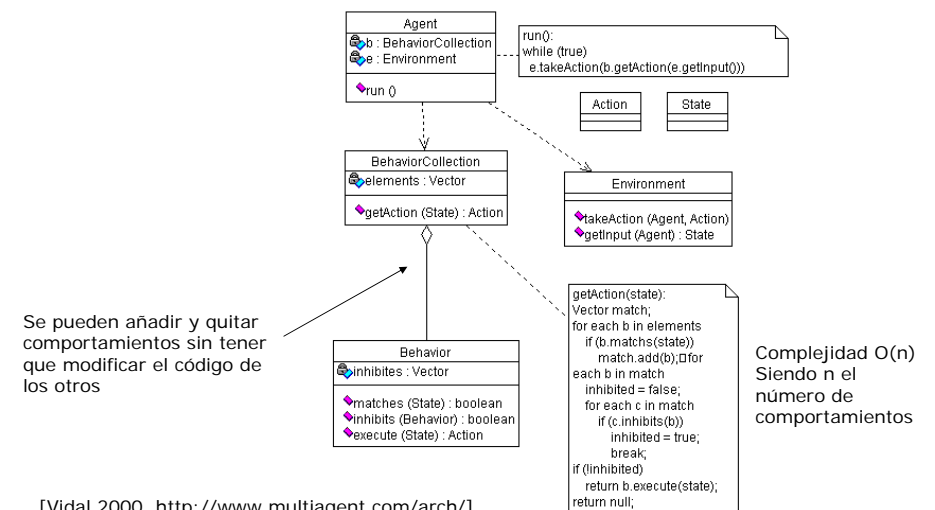
[A. Brooks, *A Robust Layered Control System for a Mobile Robot*, 1986]

 - Un conjunto de módulos de comportamiento que realizan tareas
 - No hay representación ni razonamiento simbólico
 - El comportamiento se puede implementar como un conjunto de reglas

situación -> acción

donde situación se toma directamente de la percepción (sin ningún tipo de transformación a representaciones simbólicas)
 - Jerarquía de subsunción
 - Se pueden ejecutar varios comportamientos simultáneamente
 - Para elegir entre ellos se usa la jerarquía de subsunción
 - Los comportamientos están ordenados por capas
 - Los comportamientos de las capas más bajas (mayor prioridad) inhiben a los de las capas superiores

Agentes reactivos (subsunción)



[Vidal 2000, <http://www.multiagent.com/arch/>]

UCM 2006-07

Modelos y arquitecturas de agentes

7

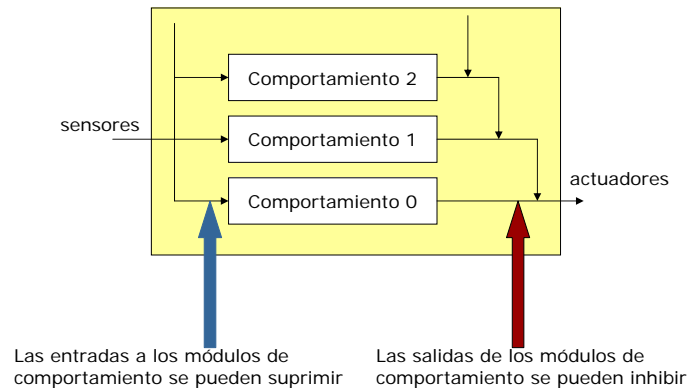
UCM 2006-07

Modelos y arquitecturas de agentes

8

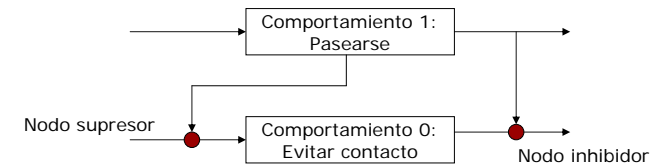
Agentes reactivos (subsunción)

■ Arquitectura de subsunción [Brooks 86]



Agentes reactivos (subsunción)

■ Nodos supresores e inhibidores



Agentes reactivos (Robots distribuidos)

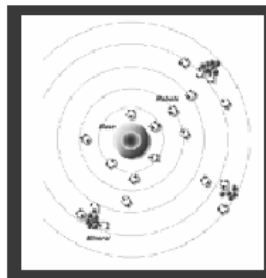
■ Ejemplo: robots distribuidos [Steels 89]

■ Problema

- Un conjunto de robots tienen que recoger minerales (cuya localización no se conoce de antemano) y llevarlas a una nave nodriza
- Los minerales se encuentran en cúmulos

■ Arquitectura de subsunción cooperativa

- La cooperación no usa comunicación directa. La comunicación se realiza a través del entorno:
 - Campo gradiente de la señal generada por la nave nodriza
 - Partículas radioactivas que pueden recoger, echar y detectar los robots al pasar



Agentes reactivos (Robots distribuidos)

■ Hay dos comportamientos que se ejecutan en paralelo:

■ Comportamientos de manejo de objetos

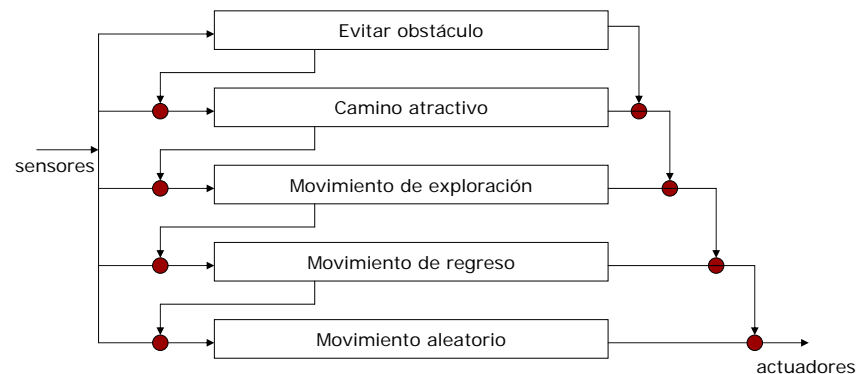
1. Si se detecta un obstáculo => cambiar de dirección.
2. Si se llevan muestras y se está en la base => descargar las muestras.
3. Si se llevan muestras y no se está en la base => dejar dos migas y moverse hacia mayor gradiente.
4. Si se detecta una muestra => recogerla.
5. Si se detectan migas => recoger una y moverse hacia menor gradiente.
6. Si cierto => moverse de manera aleatoria.

■ Comportamientos de movimiento

- organizados de acuerdo a una jerarquía de subsunción

Agentes reactivos (subsunción)

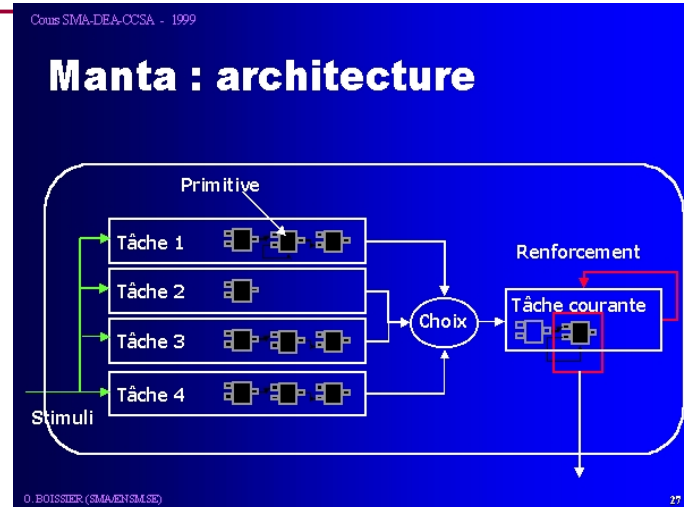
- Ejemplo de jerarquía de subsunción
 - Sistema de control de un robot móvil



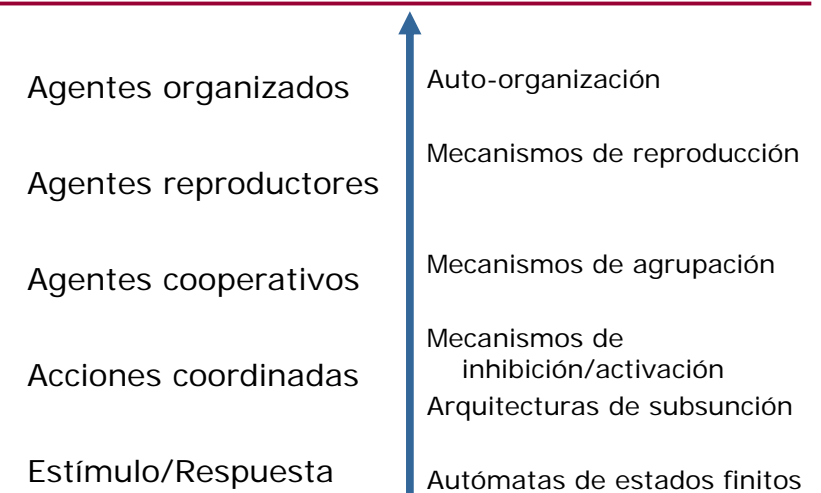
Agentes reactivos (MANTA)

- MANTA: *Modeling an ANThill Activity* [Drogoul 93]
 - Simulación de sociedades de hormigas para estudiar la emergencia del reparto de trabajo en el seno de la sociedad
 - Cada hormiga tiene operaciones de percepción, selección y activación que manipulan un conjunto de tareas
 - Cada tarea se define por:
 - Una secuencia de acciones primitivas directamente ejecutables por los actuadores del agente en su entorno
 - Peso (importancia para el agente)
 - Umbral y nivel de activación
 - Selección por competición de tareas
 - A nivel de sistema esta regulación permite la emergencia de especializaciones y de repartición de tareas entre agentes

Agentes reactivos (MANTA)



Clasificación de agentes reactivos [Demazeau 96]

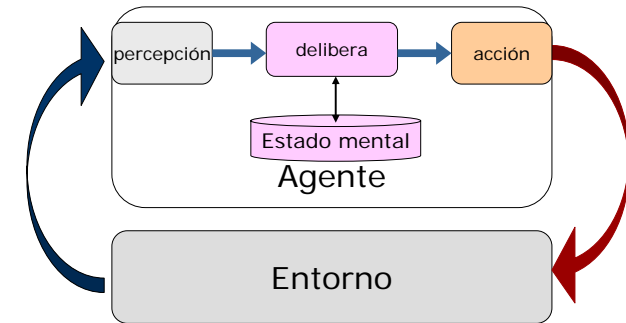


Agentes reactivos: conclusiones

- Sistemas constituidos de numerosos agentes homogéneos
 - Sencillos, flexibles, tratables computacionalmente, robustos, tolerantes a fallos
- La inteligencia emerge del SMA
- Problemas:
 - Los agentes necesitan conocer suficiente información sobre su entorno para actuar adecuadamente
 - La visión del agente es a corto plazo ya que está basada únicamente en información local
 - Es difícil el aprendizaje y la mejora de las capacidades de los agentes con el tiempo
 - Es difícil desarrollar agentes con muchas capas de comportamiento
 - La dinámica de las interacciones entre los comportamientos se hace cada vez más compleja
 - No hay metodología para crear este tipo de agentes: prueba y error

Agentes deliberativos

- Extienden arquitecturas cognitivas de la IA
- El proceso del agente introduce una función deliberativa entre la percepción y la ejecución para elegir la acción correcta



Agentes deliberativos

- Ejemplo de ciclo de ejecución de un agente deliberativo

```
EstadoMental s;  
ColaEventos eq;  
...  
s.inicializa();  
  
while (true) {  
    opciones = generar_opciones (eq, s);  
    seleccionado = delibera (opciones, s);  
    s.actualiza_estado(seleccionado);  
    ejecutar (s);  
    eq.mira_eventos();  
}
```

Agentes deliberativos

- Requieren dos procesos:
 1. Decidir qué objetivos perseguir: deliberación
 2. Decidir cómo alcanzar dichos objetivos: razonamiento basado en medios y fines
- Se basan en el razonamiento práctico (decidir en cada momento la acción a realizar para facilitar la consecución de los objetivos)
- Modelo BDI (Beliefs-Desires-Intentions)
 - Precursor: IRMA [Bratman 88]
 - Arquitectura PRS (Procedural Reasoning System) [Georgeff y Lansky 87]
 - Lógicas BDI [Rao y Georgeff]

Arquitectura BDI

- Las intenciones del agente juegan un importante papel en el razonamiento práctico:
 - Dirigen el razonamiento basado en medios y fines
 - Restringen las deliberaciones futuras
 - Persisten
 - Influyen las creencias sobre las que se basará el futuro razonamiento práctico
- Cada cierto tiempo el agente deberá replantearse sus intenciones, abandonando aquellas que considera que no va a alcanzar, aquellas que ya ha alcanzado y aquellas cuya justificación ha desaparecido

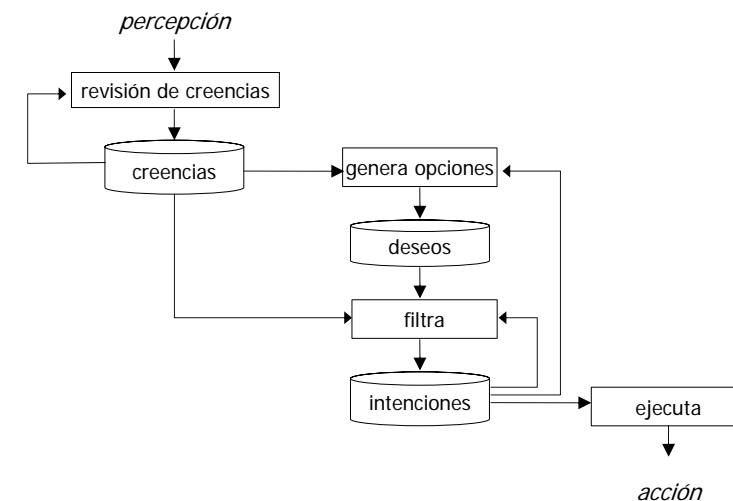
Arquitectura BDI

- Dos extremos:
 - El agente que no reconsidera suficientemente a menudo sus intenciones (atrevido)
 - Comportamiento dirigido por objetivos
 - El agente que continuamente reconsidera sus intenciones dedicando así un tiempo insuficiente a su consecución (precavido)
 - Comportamiento dirigido por eventos (reactivo)
- Lo difícil es encontrar el equilibrio entre ambos comportamientos [Kinny y Georgeff 91]
 - En entornos estáticos el comportamiento dirigido por objetivos es más adecuado
 - Los agentes no pierden tiempo reconsiderando sus objetivos y actúan para tratar de conseguir satisfacerlos
 - En entornos dinámicos es necesario tener cierta precaución
 - Los agentes pueden reconsiderar sus intenciones y aprovechar nuevas oportunidades

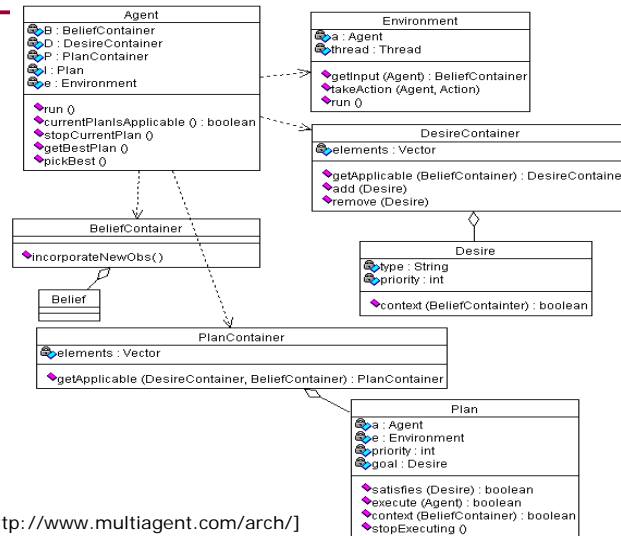
Arquitectura BDI

- Componentes de un agente BDI:
 - Conjunto de creencias actuales que tiene el agente acerca de su entorno
 - Función de revisión de creencias (frc) que actualiza las creencias en base a las percepciones
 - Función de generación de opciones (deseos) a partir de sus creencias e intenciones
 - Conjunto de opciones actuales (acciones disponibles)
 - Función filtro correspondiente al proceso de deliberación
 - Determina las nuevas intenciones en función de sus creencias, deseos e intenciones
 - Conjunto de intenciones actuales
 - Función de selección de acciones (ejecuta)
 - Determina la acción a ejecutar a partir de las intenciones actuales

Arquitectura BDI



Arquitectura BDI



[Vidal 2000, <http://www.multiagent.com/arch/>]

UCM 2006-07

Modelos y arquitecturas de agentes

25

Arquitectura BDI extendida (1)

- En las arquitecturas BDI el estado interno del agente (habitualmente conocido como “estado mental”) está formado por tres componentes: creencias, deseos e intenciones.
- La arquitectura BDI clásica ha evolucionado a lo que podríamos denominar arquitectura BDI extendida, en la cual el estado mental del agente integra cinco factores:
 - Creencias
 - Deseos
 - Objetivos
 - Intenciones
 - Planes

UCM 2006-07

Modelos y arquitecturas de agentes

26

Arquitectura BDI extendida (2)

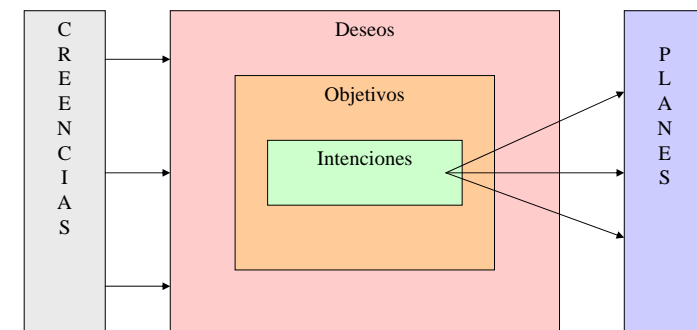
- Creencias: conocimiento del agente sobre su entorno
- Deseos: se derivan de las creencias. Hechos que el agente quiere que se cumplan en estados futuros. Puede haber conflictos y contradicciones entre ellos
- Objetivos: subconjunto de los deseos a cuya consecución podría dedicarse el agente. Tienen que ser realistas y no puede haber conflictos entre ellos
- Intenciones: subconjunto de los objetivos. Son los que el agente persigue en el momento presente
- Planes: combinan las intenciones del agente en unidades consistentes. Reflejan las acciones a desarrollar por el agente para conseguir sus intenciones. Habrá un plan global y un número suficiente de niveles de subplanes

UCM 2006-07

Modelos y arquitecturas de agentes

27

Arquitectura BDI extendida (3)



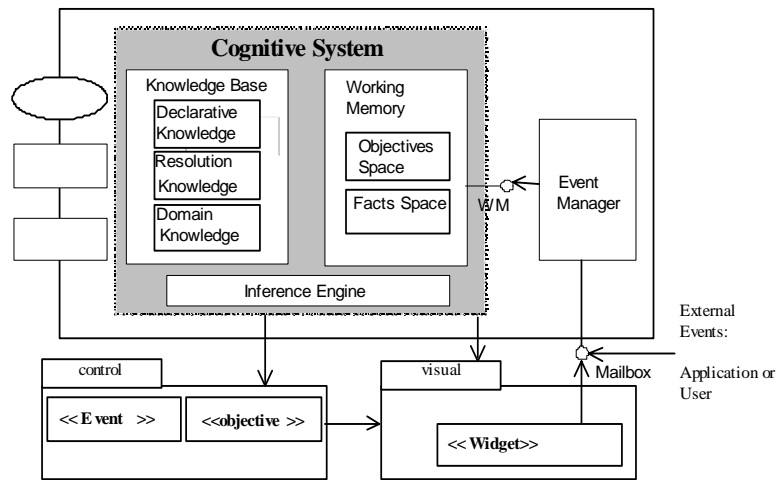
UCM 2006-07

Modelos y arquitecturas de agentes

28

Una arquitectura para agentes de interfaz

[Gomez 00]

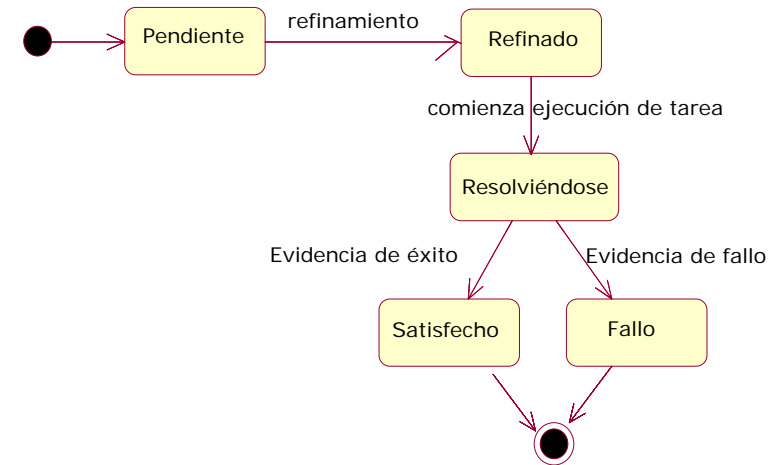


UCM 2006-07

Modelos y arquitecturas de agentes

29

Ciclo de vida de un objetivo



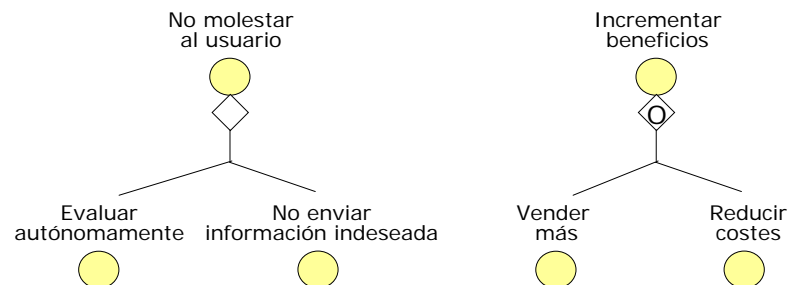
UCM 2006-07

Modelos y arquitecturas de agentes

30

Descomposición de objetivos

Árboles Y/O [Rich y Knight 90]



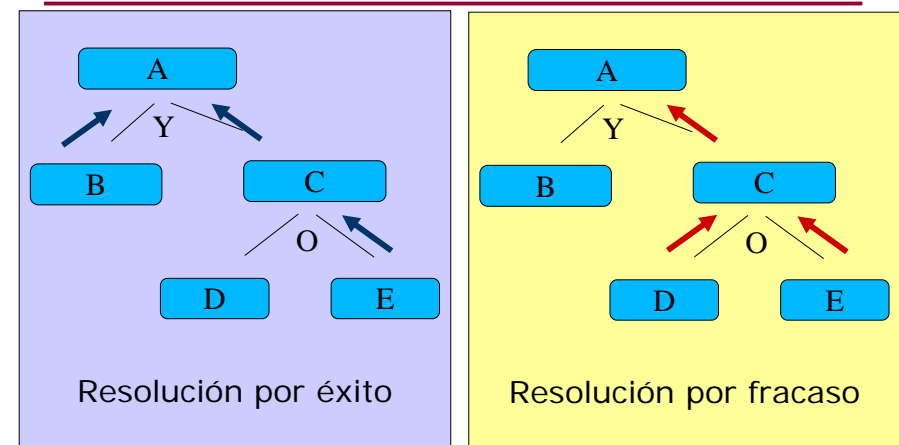
- Reglas de transmisión de éxito o de fallo

UCM 2006-07

Modelos y arquitecturas de agentes

31

Espacio de objetivos



UCM 2006-07

Modelos y arquitecturas de agentes

32

Cambio de estados de objetivos

Rule 1	If There is an event indicating proximity of a deadline for a task and there exists Analyze_situation in PENDING state then tell the objectives space to refine Analyze_situation
Rule 2	If there exists Obtain_data in PENDING state then activate Obtain_data and solve Obtain_data
Rule 3	If there exists data (reports and status) about a task and there exists Analyze_situation in SOLVING state then solve Analyze_situation

Arquitectura BDI: Conclusiones

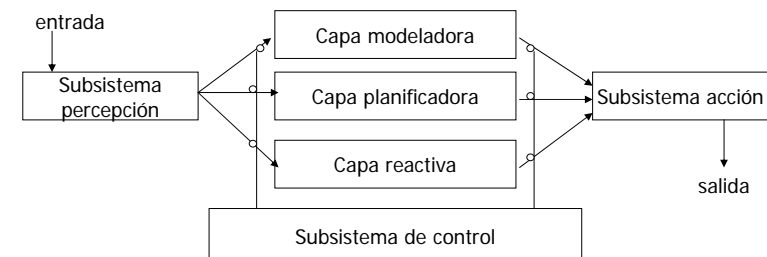
- El proceso de generación de opciones de un agente BDI es realmente un proceso recursivo de generación de una jerarquía de planes
 - En cada paso se generan intenciones más específicas hasta llegar a acciones directamente ejecutables
- El proceso de deliberación de un agente BDI debe ser ajustado en función de la variabilidad del entorno
- En la práctica el problema está en encontrar implementaciones eficientes de esta arquitectura

Agentes híbridos

- Arquitecturas basadas en capas
 - Unas capas o subsistemas implementan el comportamiento reactivo y otras el comportamiento deliberativo
 - *Capas horizontales*: todas las capas están conectadas a la entrada y salida del agente
 - *Capas verticales*: la entrada y la salida están conectadas a una única capa del agente
- Inicialmente en robótica
 - AuRA [Arkin 90, *Integrating behavioral, perceptual and world knowledge in reactive navigation*]
 - Sistema reactivo para el control de bajo nivel del robot
 - Planificación para la toma de decisiones
- Otros ejemplos:
 - RAP [Firby 89, *Adaptive execution in dynamic domains*]
 - Interrap [Muller et al. 95, *Modelling reactive behaviour in vertically layered agent architectures*]
 - Touring Machines [Ferguson 92]

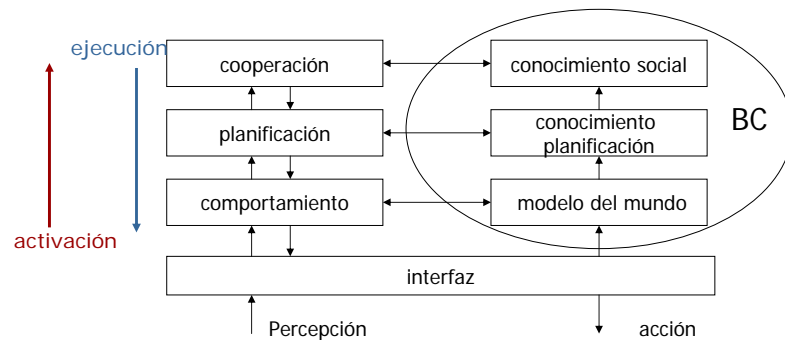
Arquitectura de capas horizontales: TouringMachines

- Arquitectura de 3 capas horizontales:
 - Reactiva (respuesta inmediata a cambios del entorno, como reglas situación->acción, o entrada de sensor-> salida de actuador)
 - Planificadora (comportamiento proactivo basado en esquemas de planes)
 - Modeladora (modelo del mundo para anticipar conflictos y genera nuevos objetivos para la capa planificadora)
- Embebidos en un subsistema de control basado en reglas que pueden inhibir entradas y salidas, y determina que capa tiene control sobre el agente



Arquitectura de capas verticales: InterRap

- Arquitectura de 3 capas verticales con 2 pasos:
 - Capa de comportamiento (reactivo): información del entorno
 - Capa de planificación (pro-activo): planes y acciones del agente
 - Capa de cooperación (interacción social): planes y acciones de otros agentes del entorno
- Más una base de conocimiento también organizada por capas



UCM 2006-07

Modelos y arquitecturas de agentes

37

Arquitecturas híbridas: Conclusiones

- Arquitecturas de carácter general
 - Las más utilizadas
 - La propia arquitectura de subsunción de Brooks es otro ejemplo de este tipo de arquitecturas
 - Utilizan una descomposición natural de la funcionalidad del agente
 - No tienen la claridad conceptual y semántica de otras arquitecturas
 - La gestión de interacciones entre capas puede ser compleja
- Capas horizontales:
 - Ventaja: simplicidad conceptual
 - Si necesitamos que un agente tenga n tipos de comportamientos, implementamos n capas diferentes
 - Inconveniente: el comportamiento global puede no ser coherente
 - Se suele añadir una función de mediación que decide qué capa tiene el control en cada momento → cuello de botella y complejidad de diseño del control de interacciones entre capas
- Capas verticales
 - Arquitecturas de uno o dos pasos
 - Ventaja: La complejidad de las interacciones entre capas se reduce
 - Inconveniente: Para tomar una decisión hay que pasar el control a cada una de las capas → No tolerante a fallos

UCM 2006-07

Modelos y arquitecturas de agentes

38

Referencias

- Bratman, M. E., Israel, D., and Pollack, M., Plans and Resource-bounded Practical Reasoning, Journal of Computational Intelligence, vol. 4, no. 4, pp. 349-355, 1988.
- Brenner, W., Zarnekow, R. Wittig, H. Intelligent Software Agents. Springer, 1998.
- Brooks, R.A., Intelligence without representation. Artificial Intelligence, 47, 1991.
- Carver, N. and Lesser, V. R.: The Evolution of Blackboard Control Architectures. Informe. Department of Computer Science, University Massachusetts. 1992
- Ferber, J. Multiagent systems : an introduction to distributed artificial intelligence, Addison-Wesley, 1999.
- Genesereth, M.R., Ketchpel, S.P. Software Agents. CACM, 37, 7, 1994
- Jennings, N.R., On agent-based software engineering. Artificial Intelligence, 117, 2000.
- Müller, J.P., *The design of intelligent agents: a layered approach*. Lecture Notes in Computer Science, Vol.1177, Springer-Verlag, 1996
- Shoham, Y. Agent-oriented programming. Artificial Intelligence, 60, 1993.
- Sycara K.P. Multiagent Systems. AI Magazine, Summer 1998.
- Weiss, G. Multiagent Systems. The MIT Press, 1999.

UCM 2006-07

Modelos y arquitecturas de agentes

39