



**DECSAI**

**Departamento de Ciencias de la Computación e I.A.**

Universidad de Granada



# Inteligencia Artificial

© Fernando Berzal, [berzal@acm.org](mailto:berzal@acm.org)

## Temario



### **Sistemas Inteligentes de Gestión: Inteligencia Artificial y Sistemas Expertos**

- **Introducción a la Inteligencia Artificial**
- Búsqueda en I.A.
- Sistemas expertos basados en reglas: CLIPS
- Lógica en I.A.
- PROLOG



# Inteligencia Artificial



- **Inteligencia Artificial**
  - Historia
  - Definiciones
  - Aplicaciones
- **Sistemas expertos**
  - Historia
  - Estructura de un sistema experto
  - Características de un sistema experto
- **Mecanismos de representación del conocimiento**
- **Métodos de razonamiento**
- **Sistemas expertos basados en reglas (S.E.B.R.)**
  - Motor de inferencia
  - Encadenamiento hacia adelante y hacia atrás



# Inteligencia Artificial



## Historia

### 1943 Circuitos booleanos como modelos del cerebro

McCulloch & Pitts: "A logical calculus of the ideas immanent in nervous activity." *Bulletin of Mathematical Biophysics*, 5:115-133.

### 1950 Test de Turing

Alan M. Turing: "Computing Machinery and Intelligence", *Mind* LIX (236): 433–460, doi:10.1093/mind/LIX.236.433

### 1956 "Inteligencia Artificial"

Dartmouth Summer Research Conference on Artificial Intelligence (John McCarthy, Marvin Minsky, Nathaniel Rochester, Claude Shannon, Ray Solomonoff, Oliver Selfridge, Trenchard More, Arthur Samuel, Herbert Simon & Allen Newell)



# Inteligencia Artificial



## Historia

1950s

- Primeros programas con IA:  
LT [Logic Theorist], GPS [General Problem Solver]...

1965

- Algoritmo para razonamiento lógico de Robinson.

1966-73

- Estudios sobre la complejidad computacional.
- La investigación en redes neuronales casi desaparece.

1969-79

- Primeros sistemas basados en el conocimiento:  
DENDRAL, MYCIN, PROSPECTOR...



# Inteligencia Artificial



## Historia

Años 80

- Industria de la I.A. (boom de los sistemas expertos)
- Poco después: El invierno de la I.A.
- Modelo conexionista: "backpropagation" (1986)  
Las redes neuronales retoman su popularidad.
- Aplicaciones, p.ej. SAT solvers (problemas de satisfacción de restricciones), modelos ocultos de Markov (procesamiento del lenguaje natural)



# Inteligencia Artificial



## Historia

Años 90

- Knowledge Discovery & Data Mining  
KDD workshop, Detroit, MI, August 20<sup>th</sup>, 1989.
- "Business rule engines"  
(versión moderna de los sistemas expertos de los 80)
- Agentes inteligentes



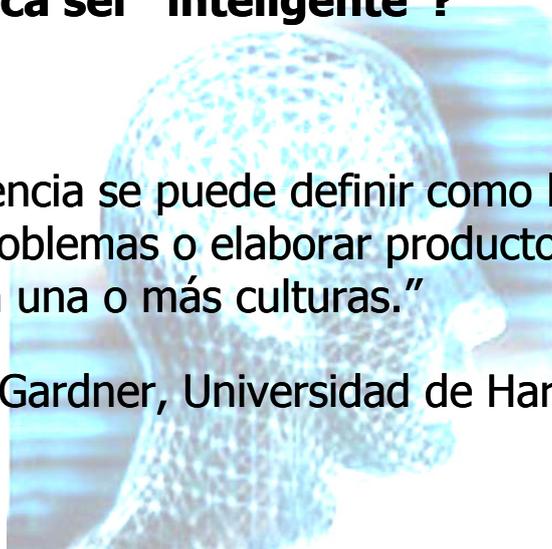
# Inteligencia Artificial



## ¿Qué significa ser "inteligente"?

"La inteligencia se puede definir como la capacidad de resolver problemas o elaborar productos que sean valiosos en una o más culturas."

-- Howard Gardner, Universidad de Harvard



# Inteligencia Artificial



## Definiciones de Inteligencia Artificial

**Sistemas que  
( piensan | actúan )  
( racionalmente | como humanos )**



# Inteligencia Artificial



## Definiciones de Inteligencia Artificial: Sistemas que piensan como humanos

- “El nuevo y excitante esfuerzo de hacer que los ordenadores piensen... máquinas con mentes en el más amplio sentido literal” (Haugeland, 1985)
- “[La automatización de] actividades que vinculamos con procesos de pensamiento humano, actividades como la toma de decisiones, resolución de problemas, aprendizaje...” (Bellman, 1978)



# Inteligencia Artificial



## Definiciones de Inteligencia Artificial: Sistemas que piensan racionalmente

- “El estudio de las facultades mentales mediante el uso de los modelos computacionales” (Charniak & McDermott, 1985)
- “El estudio de cálculos que hacen posible percibir, razonar y actuar” (Winston, 1992)



# Inteligencia Artificial



## Definiciones de Inteligencia Artificial: Sistemas que actúan como humanos

- “El arte de desarrollar máquinas con capacidad para realizar funciones que cuando son realizadas por personas requieren inteligencia” (Kurzweil, 1990)
- **“El estudio de cómo lograr que los computadores realicen tareas que, por el momento, los humanos hacen mejor”** (Rich & Knight, 1991)



# Inteligencia Artificial



## Definiciones de Inteligencia Artificial: Sistemas que actúan racionalmente

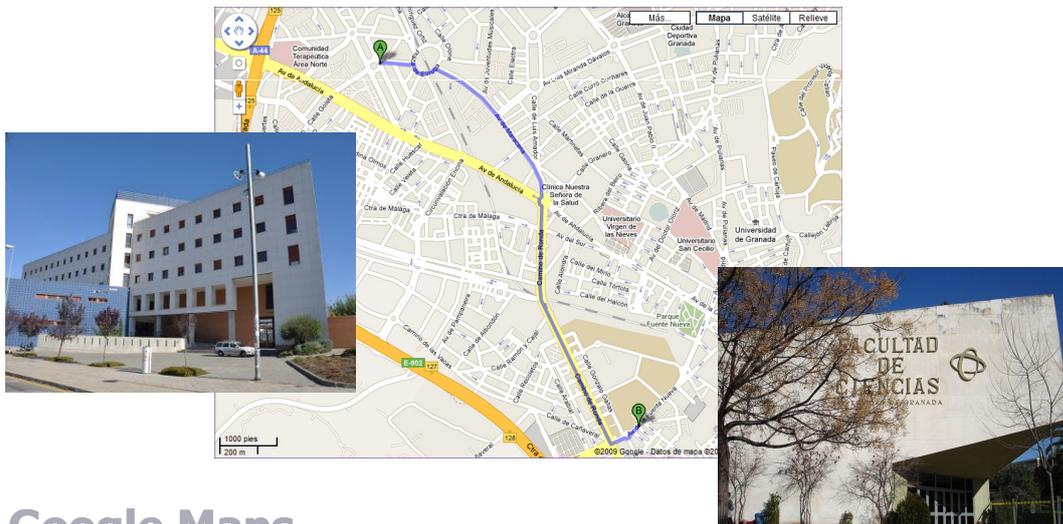
- “La inteligencia computacional es el estudio del diseño de agentes inteligentes” (Poole et al., 1998)
- “IA... está relacionada con conductas inteligentes en artefactos” (Nilsson, 1998)
- “IA... construcción de agentes que se comportan racionalmente (dados los recursos disponibles)” (Russell & Norvig, 1995)



# Aplicaciones



## Sistemas de planificación



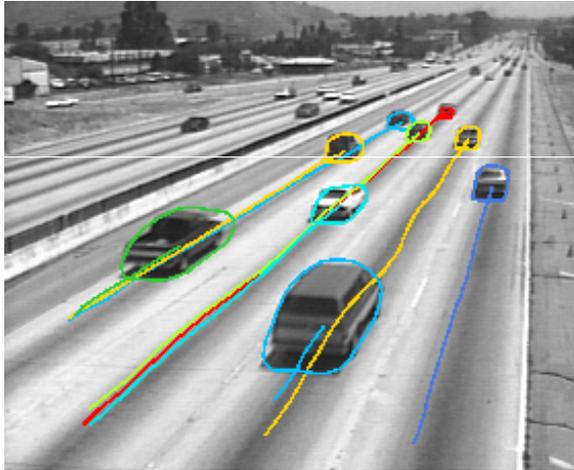
Google Maps



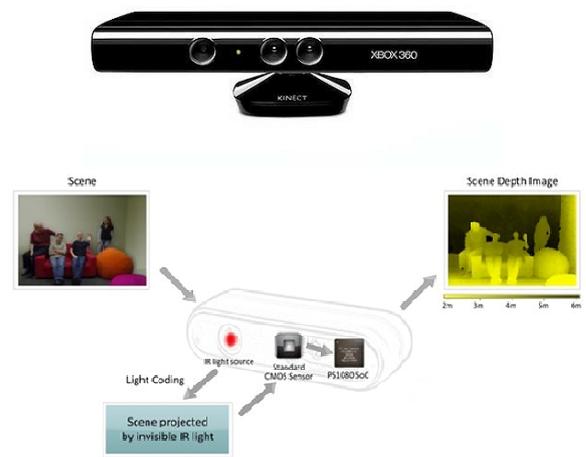
# Aplicaciones



## Visión artificial



UC Berkeley project  
(funded by Caltrans)



Microsoft Kinect



# Aplicaciones



## Robótica



Roomba © iRobot



QRIO & AIBO © Sony



# Aplicaciones



## Vehículos autónomos



Stanley — 2005 DARPA Grand Challenge Winner



# Aplicaciones



## Juegos



IBM Deep Blue, 1997



# Aplicaciones



## Juegos



RoboCup



# Aplicaciones



## Reconocimiento de voz



Android Voice Search



KINECT  
for XBOX 360.

Kinect (mic array)



# Aplicaciones



## Procesamiento del lenguaje natural



IBM Watson, 2011 — Jeopardy!



# Sistemas Expertos



Encargados de realizar (o asistir en la realización de) tareas ejecutadas por expertos:

- Ingeniería (diseño, detección de fallos...)
- Análisis científico
- Diagnóstico médico
- Análisis financiero
- Configuración de componentes
- Sistemas de control y monitorización
- Educación [intelligent tutoring systems]
- ...



# Sistemas Expertos

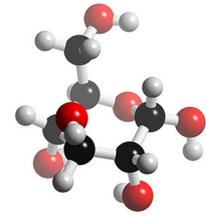


## Historia

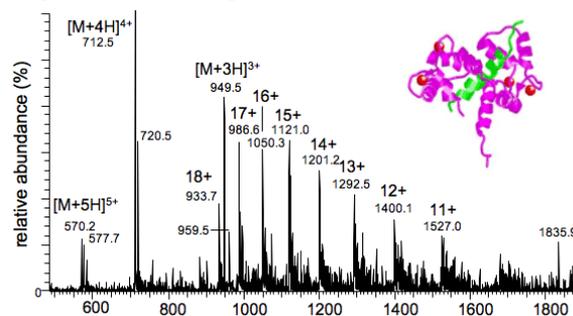
### Dendral ["Dendritic Algorithm"]

Universidad de Stanford, 1965-1975

Edward Feigenbaum, Bruce Buchanan, Joshua Lederberg & Carl Djerassi



- Primer sistema experto, programado en LISP para la identificación de compuestos orgánicos analizando datos de espectroscopia de masas.



# Sistemas Expertos



## Historia

### MYCIN

Stanford Research Institute, 1970s

Tesis doctoral de Edward Shortliffe bajo la dirección de Bruce Buchanan.



- Diseñado en LISP para identificar las bacterias causantes de infecciones severas (p.ej. meningitis) y recomendar antibióticos, con una dosis ajustada al peso del paciente [NOTA: el nombre de muchos antibióticos termina con el sufijo "-mycin"].
- Separación entre datos y conocimiento (reglas).
- Manejo de incertidumbre mediante factores de certeza [CF: Certainty Factors].



# Sistemas Expertos

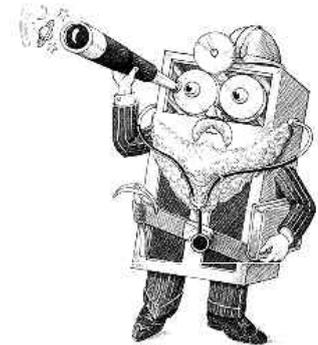


## Historia

### PROSPECTOR

Stanford Research Institute, 1974-1983

R. Duda, P.E. Hart, N.J. Nilsson, et al.



- Evaluación del potencial minero de una localización geológica (lugares de exploración o prospección).
- Representación del conocimiento del experto (mediante una red de inferencia) y de su proceso de razonamiento (mediante el uso de técnicas de tipo probabilístico).



# Sistemas Expertos



## Historia

### R1 / XCON [eXpert CONfigurer]

Carnegie Mellon University, 1978

John P. McDermott



- Escrito en OPS5 para ayudar en la selección de componentes para la configuración de máquinas DEC VAX, de Digital Equipment Corporation (adquirida por Compaq en 1998, que se fusionó con HP en 2002).
- Puesto en marcha en 1980, en 1986 había procesado 80,000 pedidos y se estima que le ahorraba a DEC más de \$40M al año.



# Sistemas Expertos



## Historia

### CLIPS

NASA Johnson Space Center, 1985-

Gary Riley

<http://clipsrules.sourceforge.net/>



### C Language Integrated Production System.

Sintaxis y nombre inspirado en OPS ("Official Production System"), 1977, creado por Charles Forgy durante su doctorado con Allen Newell en CMU.

Otros "expert system shells", descendientes de CLIPS:

- **Jess** [Java Expert System Shell] <http://www.jessrules.com/>
- **FuzzyCLIPS**, NRC, Canada



# Sistemas Expertos



## Algunos sistemas expertos "clásicos"

Medicina	MYCIN, PUFF, ABEL, AI/COAG, AI/RHEUM, CADUCEUS, ANNA, BLUE BOX, ONCOCIN, VM, INTERNIST-I, CASNET
Química	CRYALIS, DENDRAL, TQMSTONE, CLONER, MOLGEN, SECS, SPEX...
Matemáticas	AM (Automated Mathematician), EURISKO, SMP, MATHPERT... CCH-ES, ExperTAX...
Informática	PTRANS, BDS, R1/XCON, XSEL, XSITE, DART, SNAP, YES/MVS, TIMM...
Electrónica	ACE, IN-ATE, NDS, EURISKO, PALLADIO, IDEA, REDESIGN, CADHELP, SOPHIE...
Ingeniería	REACTOR, DELTA (GE), JETA, STEAMER, SACON, CALLISTO, G2, SHARP, MARVEL, Pile Selection...
Geología	DIPMETER, LITHO, MUD, PROSPECTOR...





## Algunos "shells" para sistemas expertos

- **E-MYCIN** ["Essential/Empty MYCIN"], Stanford Research Institute, 1973-1980.
- **OPS** ["Official Production System"] Carnegie Mellon University, 1977.
- **KEE** ["Knowledge Engineering Environment"] para máquinas Lisp, IntelliCorp, 1983
- **CLIPS** ["C Language Integrated Production System"] NASA Johnson Space Center, 1985
- **ESDE** ["Expert System Development Environment"], para máquinas MVS y VM, IBM, 1986
- **JESS** ["Java Expert System Shell"] Sandia National Labs, 1995
- **Drools** [business rules engine] JBoss, 2001 <http://www.jboss.org/drools/>



## ¿En qué áreas resultan útiles los sistemas expertos?

En aquellas en las que haya expertos [humanos] que nos puedan proporcionar el conocimiento necesario (de ahí el nombre de sistemas expertos).





## Diferencias con respecto a la programación convencional (imperativa)

Se separa el conocimiento

de los mecanismos que permiten manipularlo:

- Apenas existen instrucciones en el sentido clásico.
- El "programa" consiste, básicamente, en declarar conocimiento (usualmente, en forma de reglas).
- Una "caja negra" (motor de inferencia) infiere nuevo conocimiento y determina el flujo de control.



## Datos

- En lenguaje natural:

Los padres de Elena son Carlos y Belén.

Los padres de Carlos son Juan y María.

- Declaración de hechos PROLOG:

```
padres ('Carlos', 'Belén', 'Elena').
```

```
padres ('Juan', 'María', 'Carlos').
```



# Sistemas Expertos



## Conocimiento

- En lenguaje natural:

Los padres de los padres son los abuelos.

- Programa en "PROLOG":

```
abuelos (Abuelo, Abuela, Nieto) if
  padres (Abuelo, Abuela, Hijo) and
  padres (Hijo, MujerDelHijo, Nieto) .
```

```
abuelos (Abuelo, Abuela, Nieto) if
  padres (Abuelo, Abuela, Hija) and
  padres (EsposoHija, Hija, Nieto) .
```



# Sistemas Expertos



## Uso de un sistema experto

- Consulta (objetivo):

```
?- abuelos ('Juan', 'María', Nietos).
donde Nietos es una variable.
```

- Respuesta del intérprete/compilador de PROLOG:

```
Nietos = 'Elena'
```

**Conocimiento inferido (deducido) a partir de los hechos y las reglas conocidos por el sistema.**



# Sistemas Expertos



## Ejemplo

### Sistema de control de una planta industrial

- **Datos:**

Temperatura actual del reactor 1 =  $75^{\circ}$

- **Conocimiento:**

Si la temperatura de cualquier reactor supera el umbral establecido, entonces activar el mecanismo de emergencia.



# Sistemas Expertos



## Ejemplo

### Sistema de control de una planta industrial

- El conocimiento suele ser más estático que los datos, pero también puede cambiar ("refinarse"):

Si la temperatura de cualquier reactor supera el umbral establecido, entonces activar el mecanismo de emergencia, **hacer sonar la alarma y evacuar al personal.**



# Sistemas Expertos



## Datos vs. Conocimiento

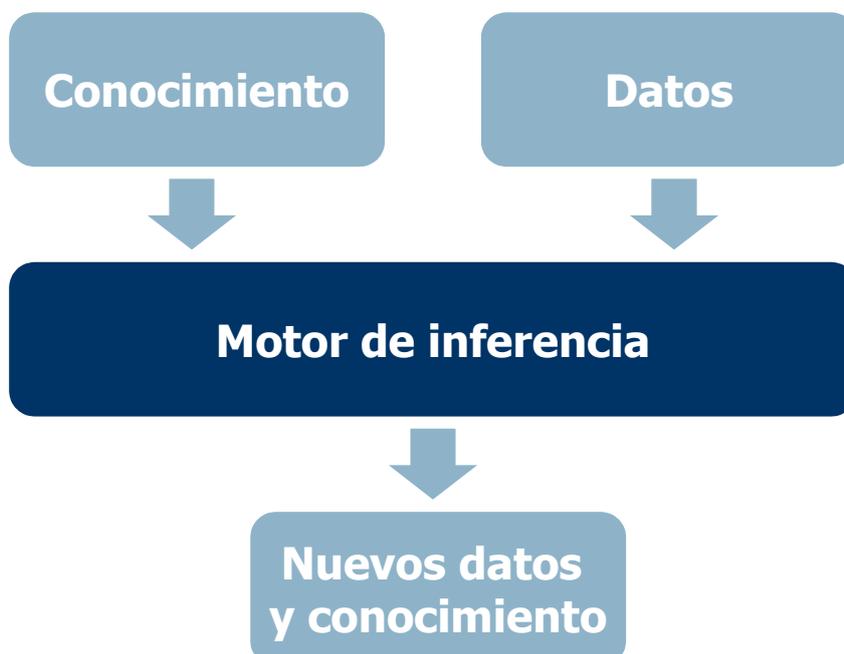
Datos	Conocimiento
Afirmaciones puntuales	Afirmaciones generales
Suelen ser dinámicos	Suele ser estático
Declaración extensiva	Declaración intensiva
Gran volumen	Pequeño volumen
Almacenamiento secundario	Almacenamiento en RAM
Representación eficiente	Representación simbólica



# Sistemas Expertos



## Estructura de un sistema experto



# Sistemas Expertos



## Términos habituales en Psicología



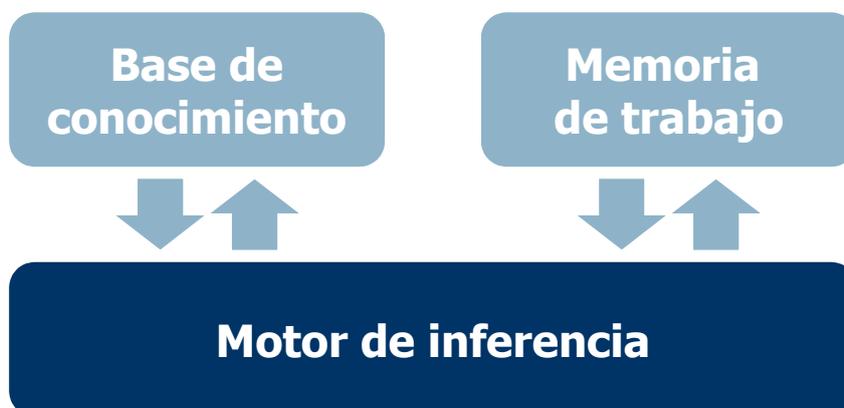
- LTM [long-term memory]: Memoria a largo plazo.
- STM [short-term memory]: Memoria a corto plazo.



# Sistemas Expertos



## Arquitectura de un sistema experto

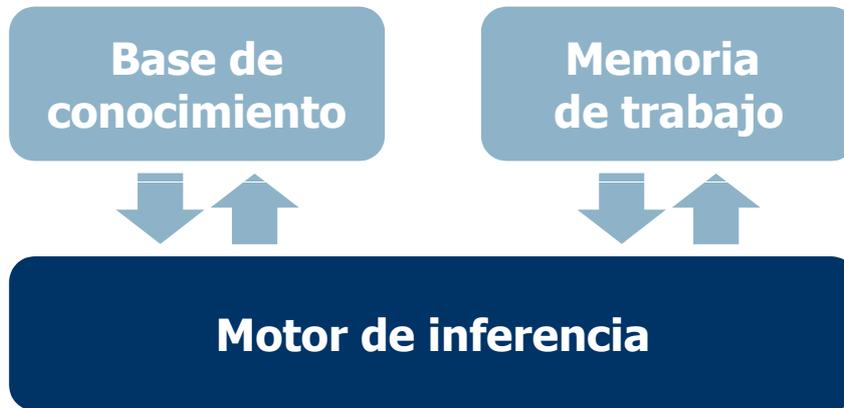


- KB [knowledge base]: Base de conocimiento.
- WM [working memory]: Memoria de trabajo.





## Arquitectura de un sistema experto



El motor de inferencia [inference engine] determina cuáles son las reglas aplicables en cada momento y se encarga de ejecutarlas.



## Ejemplo

Regla 1. Si el coche no arranca,  
realice una comprobación de la batería

Regla 2. Si el coche no arranca,  
compruebe el indicador de combustible

...

Regla 75. Si se comprobado la batería y el voltaje  
de la batería es inferior a 10 voltios, entonces  
tiene que cambiar la batería.

...

Regla 120. Si ha comprobado el indicador de  
combustible y el depósito de combustible está  
vacío, entonces hay que llenar el depósito.



# Sistemas Expertos



## Ejemplo

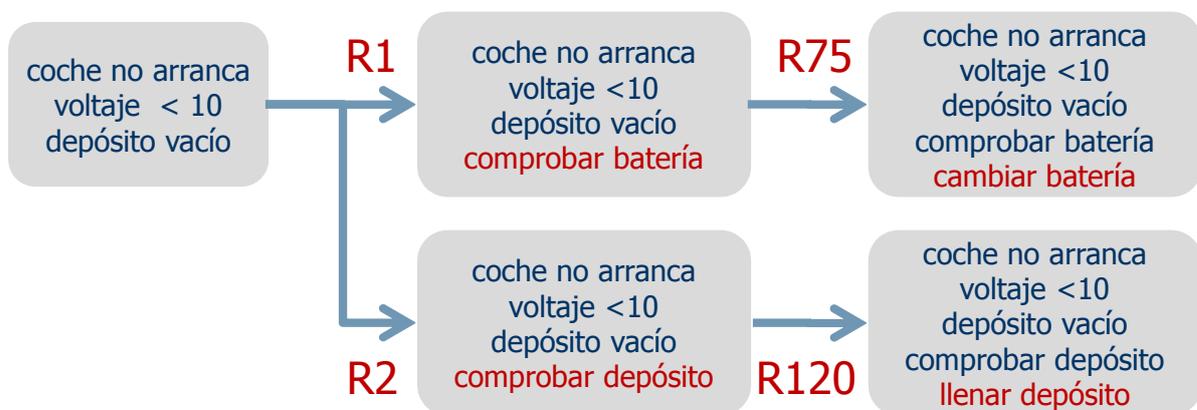
- Regla 1. IF coche no arranca,  
THEN comprobar batería
- Regla 2. IF coche no arranca  
THEN comprobar combustible
- ...
- Regla 75. IF comprobar batería  
AND voltaje batería < 10V  
THEN cambiar batería
- ...
- Regla 120. IF comprobar combustible  
AND depósito de combustible vacío  
THEN llenar depósito.



# Sistemas Expertos



## Ejemplo



El motor de inferencia determina el orden en el que se aplican las reglas "activas" (aquellas para las que se cumple su antecedente).



# Sistemas Expertos



## Características de un sistema experto

- Separación de los datos y el conocimiento de su manipulación, lo que permite actualizar fácilmente la base de conocimiento o la base de datos sin tener que modificar el motor de inferencia ("shell").
- Implementación del conocimiento explícito de un experto en un dominio concreto ["domain knowledge"]



# Sistemas Expertos



## Características de un sistema experto

- Objetivo:  
Ayudar en la toma de decisiones (pero no sustituir al experto!!!).
- Consejo práctico:  
No intentar cubrir un área excesivamente grande.  
Dividirla en subproblemas  
y construir, para cada uno de ellos,  
un sistema experto específico que lo resuelva.



# Sistemas Expertos



## Características de un sistema experto

### Razonamiento simbólico

- si una persona tiene fiebre y no es alérgica al ácido acetil salicílico (AAS) entonces suministrar aspirina 500 mg
- Pedro no tiene fiebre
- Pedro no es alérgico al AAS

- $\forall x \text{ Fiebre}(x) \wedge \neg \text{AlergiaAAS}(x) \rightarrow \text{Terapia}(x, \text{Aspirina500})$
- $\text{Fiebre}(\text{Pedro})$
- $\neg \text{AlergiaAAS}(\text{Pedro})$



# Sistemas Expertos



## Características de un sistema experto

### Razonamiento heurístico

- Reglas heurísticas basadas en la **experiencia** de los expertos (que pueden fallar en situaciones concretas):
  - Ante un problema de arranque, descartar que sea un fallo de carburación y chequear primero el sistema eléctrico.
  - Si el tipo de interés está bajo, considerar invertir en acciones. Si el tipo de interés está alto, mejor invertir en bonos.
  - Las personas no suelen coger una gripe en verano.
  - Si se sospecha cáncer, comprobar el historial familiar.



# Sistemas Expertos



## Características de un sistema experto

Razonamiento inexacto (con incertidumbre)

- PROSPECTOR ('likelihoods' [verosimilitudes])
- MYCIN ('certainty factors' [factores de certeza])

Si el paciente es un huésped de riesgo  
y existen reglas que mencionan a las pseudomonas  
y existen reglas que mencionan a las klebsiellas,  
entonces **es plausible (0.4)**  
que deban considerarse primero las segundas.



# Sistemas Expertos



## Resumen de características

Programas convencionales	Sistemas expertos
Programación imperativa	Programación declarativa
Razonamiento algorítmico	Razonamiento heurístico
Control definido por el programador	Control determinado por el motor de inferencia
Difíciles de modificar	Fáciles de modificar
Información precisa	Información no precisa
Se ofrece una solución como resultado final	Se ofrece una recomendación razonada
Solución "óptima"	Solución aceptable

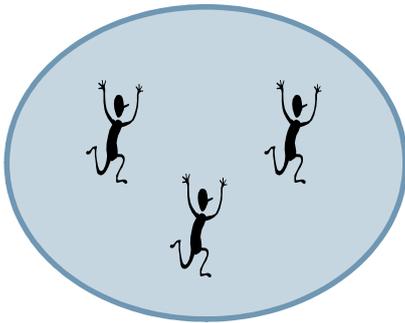


# Representación del conocimiento

## Modelos de memoria

Origen: Psicología

¿Cómo estructuramos en nuestra mente datos y conocimiento (eminentemente estático)?



EJEMPLO

Dominio de discurso:  
Conjunto de clientes



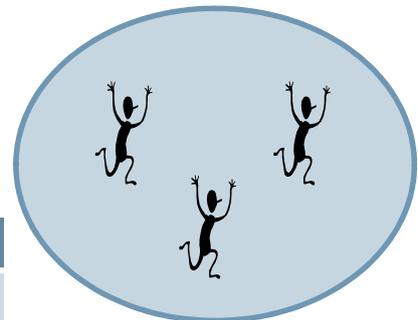
# Representación del conocimiento

## Modelos de memoria

Modelos informáticos

### ■ Representación relacional

DNI	Nombre	Edad	Saldo
12345678Z	Pedro	32	3000€
44444444A	Juan	40	10000€



### ■ Representación orientada a objetos

**Clase Cliente:**

- Subclase de Persona.
- Propiedades: Las heredadas de persona (DNI, nombre, edad) y saldo.
- Instancias: Pedro y Juan



# Representación del conocimiento

## Modelos de memoria

### Tripletas OAV (objeto-atributo-valor)

Representan conocimiento simple en el que se enfatiza la relación entre un objeto y sus propiedades

#### EJEMPLO

Juan - Altura - 185cm

Juan - EsHijoDe - María

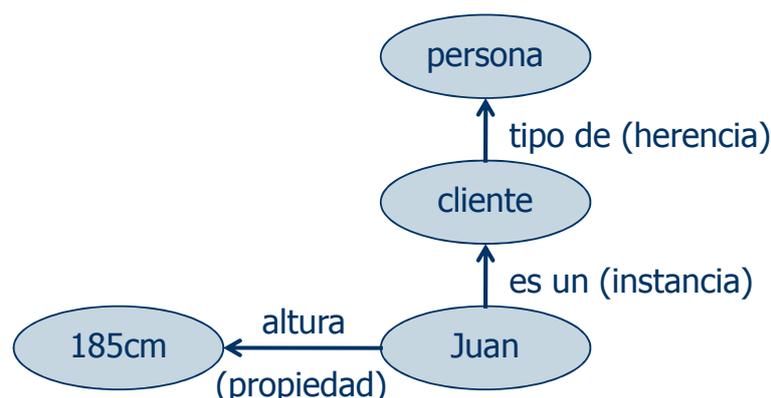


# Representación del conocimiento

## Modelos de memoria

### Redes semánticas / ontologías

Representación gráfica del conocimiento en la que se representan como arcos las relaciones entre objetos:



# Representación del conocimiento

## Modelos lógicos

Representación formal de las relaciones existentes entre objetos (y entre los objetos y sus propiedades):

- Lógica Proposicional
- Lógica de Predicados
- ...



# Representación del conocimiento

## Modelos lógicos

### Lógica Proposicional

- Se utilizan proposiciones que representan afirmaciones que pueden ser verdaderas o falsas.
- Las proposiciones se unen con operadores lógicos ( $\wedge$  [y],  $\vee$  [o],  $\neg$  [no]), y se construyen reglas con el operador de implicación lógica ( $\rightarrow$ ).
- Los mecanismos de inferencia permiten obtener nuevos datos a partir de los datos ya conocidos (por ejemplo, modus ponens, modus tollens...)



# Representación del conocimiento

## Modelos lógicos

### Lógica Proposicional

EJEMPLO: Modus ponens

$p$	si $p$
$p \rightarrow q$	y $p$ implica $q$
<hr/>	
$q$	entonces $q$

es un razonamiento válido porque  $q$  siempre será verdad, independientemente de lo que represente, cuando se cumplan  $p$  y  $p \rightarrow q$ .



# Representación del conocimiento

## Modelos lógicos

### Lógica Proposicional

EJEMPLO: Modus ponens

$p$  = "hace calor"  
 $q$  = "el profesor está incómodo"

Memoria de trabajo (datos):  $p$   
Base de conocimiento (reglas):  $p \rightarrow q$   
Deducción (aplicando modus ponens):  $q$

- $q$  pasa a formar parte de la memoria de trabajo.



# Representación del conocimiento

## Modelos lógicos

### Lógica Proposicional

EJEMPLO: Modus tollens

$\neg q$	si no q
$p \rightarrow q$	y p implica q
<hr/>	
$\neg p$	entonces no p

es un razonamiento válido porque p siempre será falso, independientemente de lo que represente, cuando se no se cumpla q y se verifique  $p \rightarrow q$ .



# Representación del conocimiento

## Modelos lógicos

### Lógica Proposicional

EJEMPLO: Modus ponens

p = "hace calor"  
q = "el profesor está incómodo"

Memoria de trabajo (datos) :	$\neg q$
Base de conocimiento (reglas):	$p \rightarrow q$
Deducción (aplicando modus tollens):	$\neg p$

¿seguro?



# Representación del conocimiento

## Modelos lógicos

### Lógica Proposicional

En un sistema experto que trabaje con lógica proposicional, el motor de inferencia será el encargado de decidir qué mecanismo de inferencia aplicar en cada momento.



# Representación del conocimiento

## Modelos lógicos

### Lógica de Predicados

La Lógica de Predicados añade la posibilidad de utilizar **cuantificadores**:

$\forall$  (para todo)

$\exists$  (existe)

Mecanismos de inferencia:  
modus ponens, modus tollens, resolución...

EJEMPLO: Lenguaje de programación PROLOG



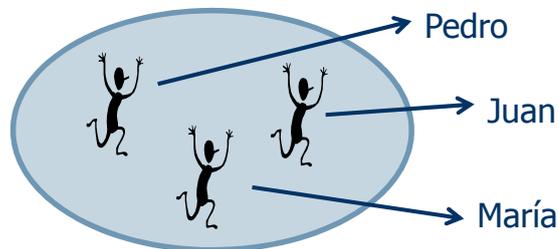
# Representación del conocimiento

## Modelos lógicos

### Lógica de Predicados

Formalización de un modelo en Lógica de Predicados:

- **Constantes** (lista de objetos): Pedro, Juan, María...  
El conjunto de todas ellas se denomina **dominio de discurso**.
- **Semántica** (proporcionada por el experto)



# Representación del conocimiento

## Modelos lógicos

### Lógica de Predicados

Formalización de un modelo en Lógica de Predicados:

- **Predicados** (relaciones entre objetos):

quiere-a, casado-con, es-padre, es-abuelo...

Formalmente, un predicado necesita un nombre y se define enumerando los objetos sobre los que dicho predicado es verdad (es decir, como un subconjunto del producto cartesiano del dominio):

$\text{quiere-a} = \{(\text{Pedro}, \text{María}), (\text{Juan}, \text{María})\}$



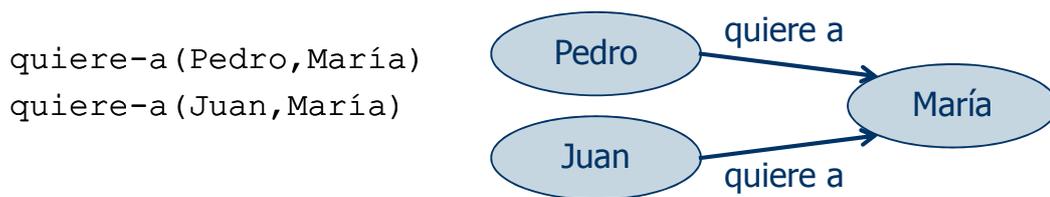
# Representación del conocimiento

## Modelos lógicos

### Lógica de Predicados

Formalización de un modelo en Lógica de Predicados:

Usualmente, los predicados se expresan como:



Ésta es la forma de representar **hechos** (datos) en Lógica de Predicados.



# Representación del conocimiento

## Modelos lógicos

### Lógica de Predicados

Formalización de un modelo en Lógica de Predicados:

Para representar el conocimiento, se utilizan **reglas**, usualmente cuantificadas universalmente:

$$\forall x \forall y \forall z \\ \text{es-padre}(x, y) \wedge \text{es-padre}(y, z) \rightarrow \text{es-abuelo}(x, z)$$

El motor de inferencia se encargará de emparejar los hechos con las reglas e ir aplicándolas.



# Representación del conocimiento

## Modelos lógicos

### Otros modelos lógicos

Existen otras lógicas que permiten incorporar aspectos como el tiempo, la incertidumbre, que hechos dejen de ser ciertos...

- Lógicas modales
- Lógicas temporales
- Lógica difusa [fuzzy logic]
- Lógicas no monótonas
- ...



# Representación del conocimiento

## Modelos mixtos

### Lógicas descriptivas [DL: Description Logic]

Más expresivas que la lógica proposicional, pero limitándose a fragmentos decidibles de la lógica de predicados de primer orden.



La base formal de los lenguajes de descripción de ontologías que se usan en la [Web Semántica](#), p.ej. **OWL** [Web Ontology Language].



# Representación del conocimiento

## Modelos mixtos

### Lógicas descriptivas [DL: Description Logic]

- **ABox** [assertion component]: Hechos.

A es una instancia de B.  
Juan es una persona.

- **TBox** [terminological component]: Descripción de un sistema mediante un vocabulario controlado (conjunto de definiciones y especializaciones).

Todos los estudiantes son personas.  
Hay 2 tipos de personas: estudiantes y profesores.



# Representación del conocimiento

## Modelos mixtos

### Lógicas descriptivas [DL: Description Logic]

#### Base de conocimiento = ABox + TBox

Desde el punto de vista lógico, la distinción ABox/TBox no es esencial, pero resulta útil en la práctica

- para resolver problemas concretos (comprobación de instancias en ABox, clasificación en TBox)
- para modelar correctamente un dominio particular (términos/conceptos en TBox [clases] y manifestaciones particulares de esos conceptos en ABox [instancias]).



# Razonamiento



hechos + conocimiento + estrategias  $\Rightarrow$  conclusiones

- En sentido amplio, facultad humana que permite resolver problemas, extraer conclusiones de los hechos y aprender de manera consciente de los hechos estableciendo conexiones causales y lógicas necesarias entre ellos.
- En sentido más restringido, se suele hablar de diferentes **tipos de razonamiento**:  
razonamiento deductivo, razonamiento inductivo,  
razonamiento abductivo, razonamiento por analogías,  
razonamiento no monótono...



# Razonamiento



## Razonamiento lógico

Proceso mental de realizar la inferencia de una conclusión a partir de un conjunto de premisas:

- Razonamiento deductivo
- Razonamiento inductivo
- Razonamiento abductivo

Hasta ahora, sólo hemos visto ejemplos de razonamiento deductivo.



# Razonamiento



## Razonamiento deductivo

Estrictamente lógico: A partir de premisas verdaderas (hechos, axiomas, reglas...), sólo se obtienen conclusiones que siempre son verdaderas.

### EJEMPLO

Dato: Sócrates es un hombre.  
Conocimiento: Todos los hombres son mortales.  
Razonamiento: Deductivo.  
Inferencia: **Sócrates es mortal.**



# Razonamiento



## Razonamiento inductivo

Se formulan conjeturas a partir de la observación de ciertos hechos, aunque no siempre sean ciertas (p.ej. técnicas de aprendizaje en I.A.).

### EJEMPLO

Dato: Sócrates es un hombre.  
Dato: Sócrates es mortal.  
Razonamiento: Inductivo.  
Inferencia: **Todos los hombres son mortales,  
o bien, Todos los mortales son hombres !!**



# Razonamiento



## Razonamiento abductivo

Formula hipótesis de trabajo que luego se pueden aceptar o rechazar.

### EJEMPLO

Dato: Sócrates es mortal.  
Conocimiento: Todos los hombres son mortales.  
Razonamiento: Abductivo.  
Inferencia: **Sócrates es un hombre ¿?**



# Sistemas expertos basados en reglas



## Modelos conductuales

p.ej. Aprendizaje condicionado (perros de Pavlov)

### Antes



Comida



Salivación



Timbre



Sin respuesta

### Durante el condicionamiento



Comida + timbre



Salivación

### Después



Timbre



Salivación



# Sistemas expertos basados en reglas

## Modelos conductuales

p.ej. Aprendizaje condicionado (perros de Pavlov)



# Sistemas expertos basados en reglas

- **Producción:** Término utilizado en Psicología Cognitiva para describir relaciones entre situaciones y acciones.
- **Regla de producción:** Término utilizado en I. A. para denotar un mecanismo de representación del conocimiento que implementa una producción.

## SI situación ENTONCES acción

- La situación (condición ó antecedente) establece las condiciones que se han de satisfacer en un momento dado para que la regla sea aplicable.
- La acción ó consecuente establece las acciones que se han de realizar cuando la regla "se active".



# Sistemas expertos basados en reglas

La acción en una regla de producción puede involucrar:

- Añadir algún dato (hecho) a la memoria de trabajo.
- Suprimir algún dato de la memoria de trabajo.
- Ejecutar algún procedimiento.

## Ejemplo

Base de conocimiento:

Reglas de producción.

Mecanismo de inferencia:

Encadenamiento hacia adelante (p.ej. CLIPS).



# Sistemas expertos basados en reglas

## Ejemplo: Guía de reparación del automóvil

Regla 1. IF coche no arranca,  
THEN comprobar batería

Regla 2. IF coche no arranca  
THEN comprobar combustible

...

Regla 75. IF comprobar batería  
AND voltaje batería < 10V  
THEN cambiar batería

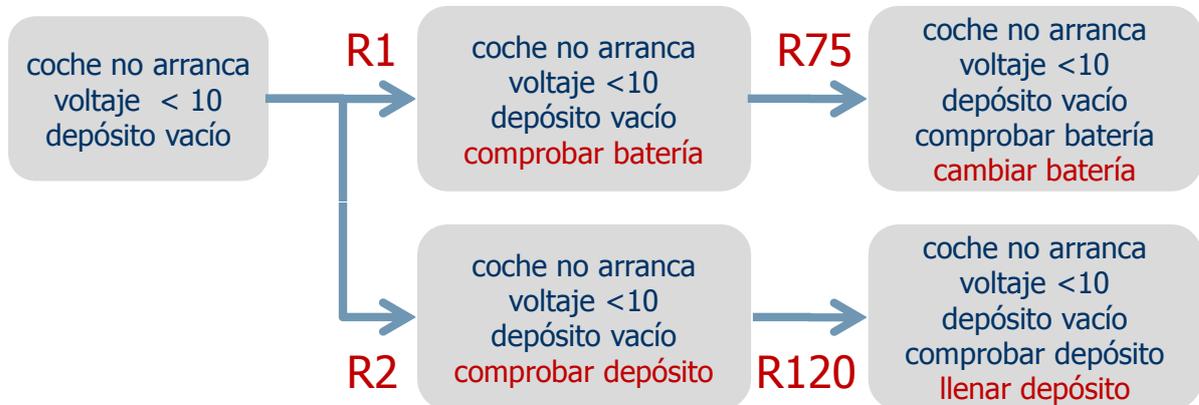
...

Regla 120. IF comprobar combustible  
AND depósito de combustible vacío  
THEN llenar depósito.



# Sistemas expertos basados en reglas

## Ejemplo: Guía de reparación del automóvil



El motor de inferencia determina el orden en el que se aplican las reglas "activas" (aquéllas para las que se cumple su antecedente).



# Sistemas expertos basados en reglas

## Ejemplo: Sistema experto de diagnóstico médico

IF el paciente tiene manchas violáceas  
THEN proceder a realizar un test de alergias

IF el paciente tiene manchas violáceas  
AND el resultado del test de alergias es negativo  
THEN añadir meningitis como posible diagnóstico

IF el paciente tiene manchas violáceas  
AND el resultado del test de alergias es negativo  
THEN añadir problema sanguíneo como posible diagnóstico



# Sistemas expertos basados en reglas

## Representación del conocimiento en un S.E.B.R.

- **Datos:** Cualquier mecanismo de representación vale.

vg. Tupla (Pedro, 27, 124000)

Registro (C) Empleado ( Nombre=Pedro,  
Edad=27,  
Salario=124000 )

Objeto (OO) Clase cliente (propiedades y métodos)

- **Conocimiento:** Reglas.

vg. SI condición ENTONCES acción  
SI antecedente ENTONCES consecuente



# Sistemas expertos basados en reglas

## Motor de inferencia

¿Cómo se van aplicando las reglas de la base de conocimiento sobre los hechos de la memoria de trabajo?

- El motor de inferencia es el encargado de hacerlo.
- Pueden utilizarse dos mecanismos básicos de inferencia:
  - Hacia adelante (p.ej. CLIPS)
  - Hacia atrás (p.ej. PROLOG).
- Se puede programar por completo (p.ej. en C) o utilizar un **shell de sistema experto** (que proporciona un lenguaje específico para declarar hechos y reglas y el motor de inferencia ya implementado).



# Sistemas expertos basados en reglas

## Motor de inferencia:

### Encadenamiento hacia adelante

Dada una regla **R: si A entonces C**

- La regla **R** se puede disparar, ejecutar o aplicar hacia adelante cuando se satisfacen las condiciones especificadas en su antecedente **A**.
- Cuando la regla se dispara, se procede a la ejecución de las acciones especificadas en su consecuente **C**.



# Sistemas expertos basados en reglas

## Motor de inferencia:

### Encadenamiento hacia adelante

Un motor de inferencia  
con encadenamiento hacia adelante:

- parte de unos hechos (datos en la memoria de trabajo),
- va cotejando (emparejando) los datos de la memoria de trabajo con los antecedentes de las reglas, y
- las va disparando hasta que se satisface algún objetivo o hasta que ninguna regla sea aplicable.



# Sistemas expertos basados en reglas

## **Motor de inferencia:**

### **Encadenamiento hacia adelante**

#### **Emparejamiento**

- Comparación de los antecedentes de cada regla con los hechos de la memoria de trabajo.
- Resultado: Conjunto de reglas que pueden aplicarse en cada momento ("conjunto conflicto").



# Sistemas expertos basados en reglas

## **Motor de inferencia:**

### **Encadenamiento hacia adelante**

#### **Resolución de conflictos**

- Selección de una regla del "conjunto conflicto" (cuando exista más de una).
- Resultado: Regla seleccionada para su aplicación.



# Sistemas expertos basados en reglas

## Motor de inferencia:

### Encadenamiento hacia adelante

#### Ejecución de la regla

- La regla seleccionada se ejecuta (dispara); esto es, se ejecutan las acciones especificadas en su parte derecha.
- **Principio de refracción:** La regla ejecutada no volverá a ser aplicable hasta que no desaparezca alguno de los hechos que hicieron posible su aplicación y vuelvan a afirmarse ("asertarse")



# Sistemas expertos basados en reglas

## Motor de inferencia:

### Encadenamiento hacia adelante

#### EJEMPLO

R0: IF hay placas (puntos blancos) en la garganta  
THEN diagnóstico: posible infección de garganta

R1: IF garganta inflamada  
AND sospechamos infección bacteriana  
THEN diagnóstico: posible infección de garganta

R2: IF temperatura paciente > 39  
THEN paciente tiene fiebre

R3: IF paciente enfermo más de una semana  
AND paciente tiene fiebre  
THEN sospechamos infección bacteriana



# Sistemas expertos basados en reglas



## Motor de inferencia:

### Encadenamiento hacia adelante

#### EJEMPLO

M.T.	C.C.	Resolución
temperatura = 40 enfermo dos semanas garganta inflamada		

## Memoria de trabajo inicial:

Datos provenientes de la exploración del paciente, p.ej.

temperatura = 40  
enfermo dos semanas  
garganta inflamada



# Sistemas expertos basados en reglas



## Motor de inferencia:

### Encadenamiento hacia adelante

#### EJEMPLO

M.T.	C.C.	Resolución
temperatura = 40 enfermo dos semanas garganta inflamada	R2	R2
temperatura = 40 enfermo dos semanas garganta inflamada fiebre		

- Si no aplicamos el principio de refracción, podríamos estar añadiendo indefinidamente el hecho de que la persona tiene fiebre.



# Sistemas expertos basados en reglas



## Motor de inferencia:

### Encadenamiento hacia adelante

#### EJEMPLO

M.T.	C.C.	Resolución
temperatura = 40 enfermo dos semanas garganta inflamada	R2	R2
temperatura = 40 enfermo dos semanas garganta inflamada fiebre	R3	R3
temperatura = 40 enfermo dos semanas garganta inflamada fiebre posible infección bacteriana		



# Sistemas expertos basados en reglas



## Motor de inferencia:

### Encadenamiento hacia adelante

#### EJEMPLO

M.T.	C.C.	Resolución
temperatura = 40 enfermo dos semanas garganta inflamada fiebre posible infección bacteriana	R1	R1
temperatura = 40 enfermo dos semanas garganta inflamada fiebre posible infección bacteriana infección en la garganta		



**Diagnóstico: Posible infección en la garganta**

# Sistemas expertos basados en reglas

## Motor de inferencia:

### Encadenamiento hacia adelante

#### EJEMPLO

Si, en la exploración inicial, se hubiesen visto puntos blancos en la garganta, entonces tendríamos:

M.T.	C.C.	Resolución
temperatura = 40 enfermo dos semanas garganta inflamada puntos blancos en la garganta	R0, R2	???

Dependiendo del orden en que el motor de inferencia seleccione las reglas, podríamos llegar a un diagnóstico con menos pasos (si se elige R0 primero) o, incluso, puede que diferente.

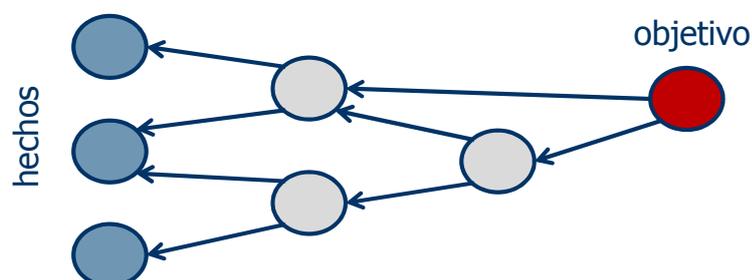


# Sistemas expertos basados en reglas

## Motor de inferencia:

### Encadenamiento hacia atrás

Es obligatorio incluir un objetivo inicial (lo que pretendemos demostrar), que se irá reemplazado por otros subobjetivos como resultado de aplicar las reglas hacia atrás.



# Sistemas expertos basados en reglas

## **Motor de inferencia: Encadenamiento hacia atrás**

Dada una regla **R: si A entonces C**

- La regla se puede aplicar hacia atrás cuando existe un objetivo **OBJ** que concuerda con el consecuente **C**.
- Cuando se aplica una regla hacia atrás, se procede a sustituir la demostración de **OBJ** por la demostración de los antecedentes **A** de la regla; esto es, el objetivo inicial **OBJ** se reemplaza por todos los objetivos **A**.



# Sistemas expertos basados en reglas

## **Motor de inferencia: Encadenamiento hacia atrás**

Un motor de inferencia  
con encadenamiento hacia atrás:

- parte de unos hechos (datos) y de un objetivo inicial,
- va cotejando (emparejando) los consecuentes de las reglas con la lista de objetivos, y
- va aplicando las reglas hacia atrás (aumentando así la lista de objetivos) hasta que todos ellos coincidan con hechos de la memoria de trabajo.

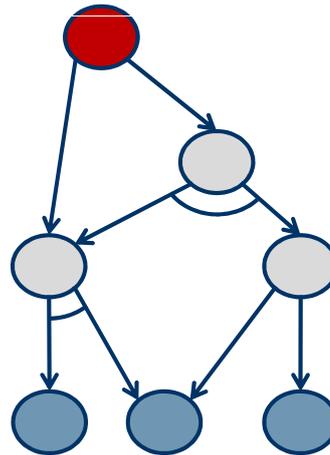


# Sistemas expertos basados en reglas

## Motor de inferencia: Encadenamiento hacia atrás

Para representar los objetivos que han de demostrarse, suele utilizarse un grafo Y/O:

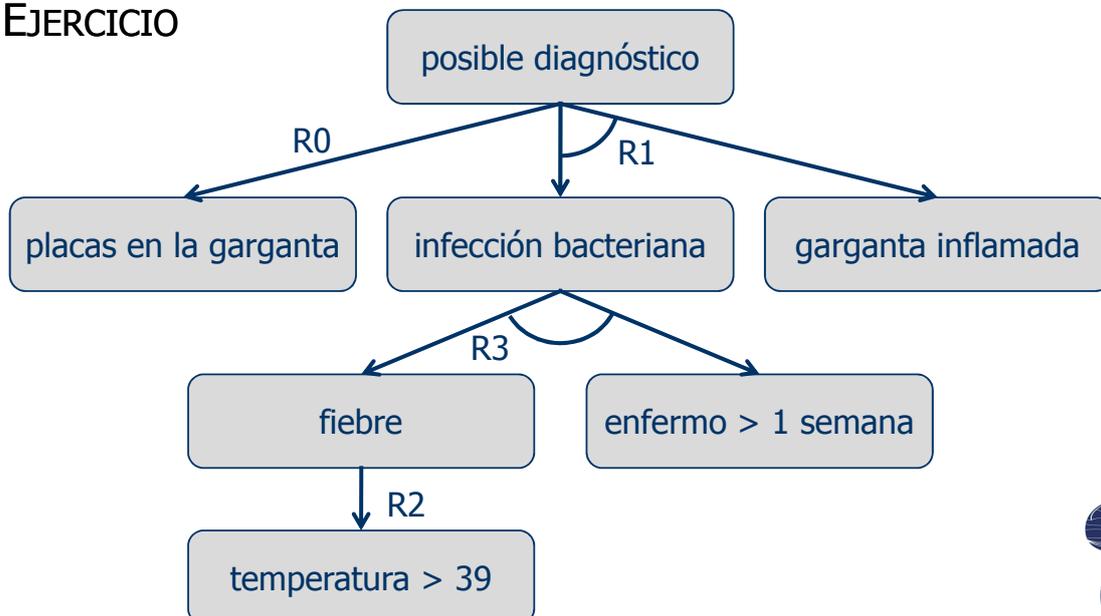
EJERCICIO  
Dibujar el grafo Y/O  
para el ejemplo anterior.



# Sistemas expertos basados en reglas

## Motor de inferencia: Encadenamiento hacia atrás

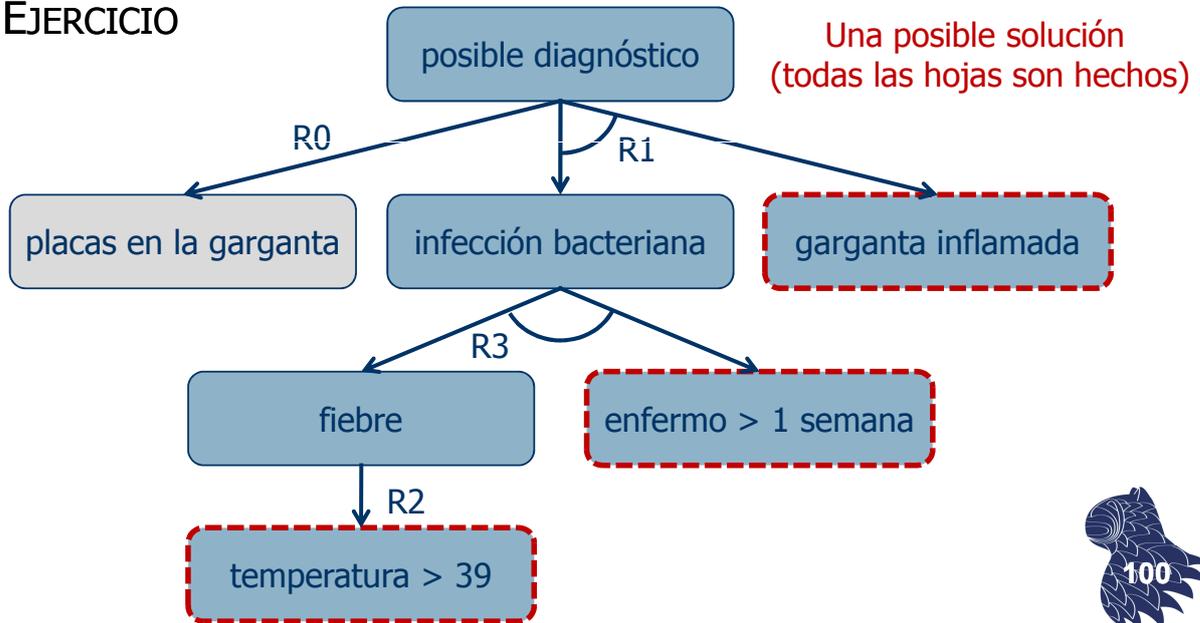
EJERCICIO



# Sistemas expertos basados en reglas

## Motor de inferencia: Encadenamiento hacia atrás

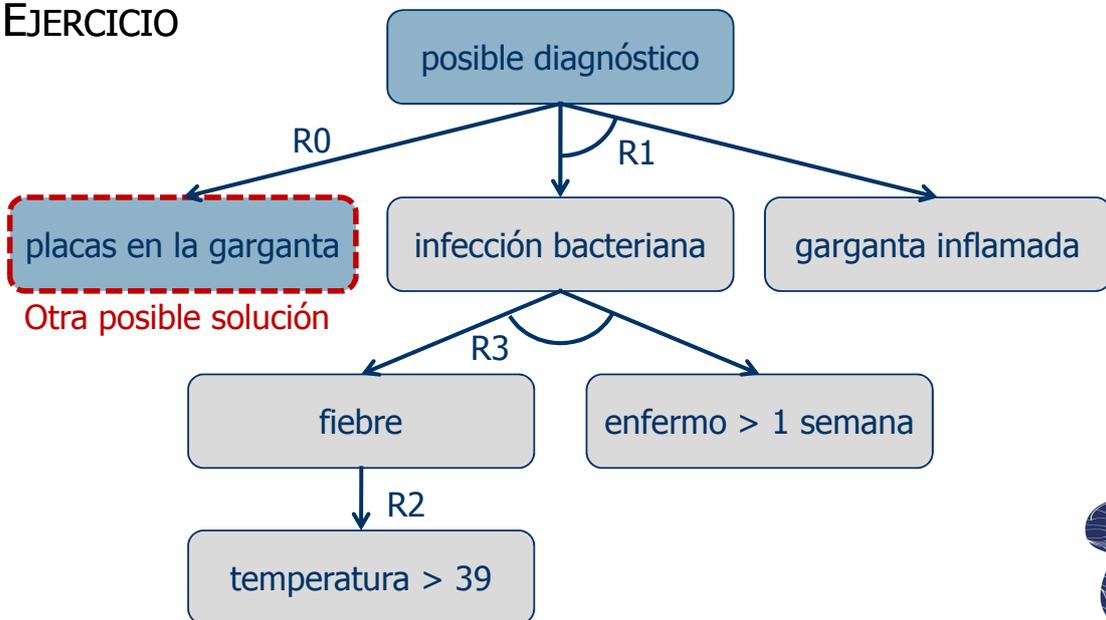
EJERCICIO



# Sistemas expertos basados en reglas

## Motor de inferencia: Encadenamiento hacia atrás

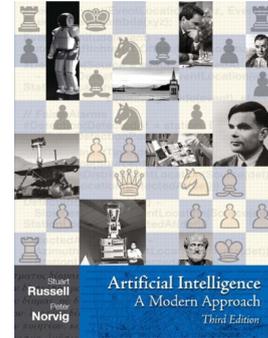
EJERCICIO



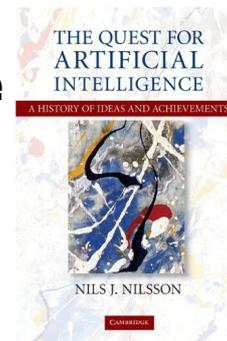
# Bibliografía



- Stuart Russell & Peter Norvig: **Artificial Intelligence: A Modern Approach**  
Prentice-Hall, 3<sup>rd</sup> edition, 2009  
ISBN 0136042597



- Nils J. Nilsson  
**The Quest for Artificial Intelligence**  
Cambridge University Press, 2009  
ISBN 0521122937



# Bibliografía



## Bibliografía complementaria

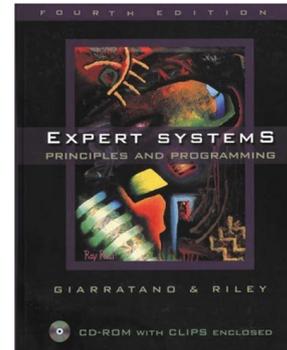
- Elaine Rich & Kevin Knight: **Artificial Intelligence**. McGraw-Hill, 1991.
- Patrick Henry Winston: **Artificial Intelligence**. Addison-Wesley, 1992.
- Nils J. Nilsson: **Principles of Artificial Intelligence**. Morgan Kaufmann, 1986.
- Daniel Jurafsky & James H. Martin: **Speech and Language Processing**. Prentice Hall, 2008.
- Yoav Shoham & Kevin Leyton-Brown: **Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations**. Cambridge University Press, 2008.



# Bibliografía: Sistemas Expertos

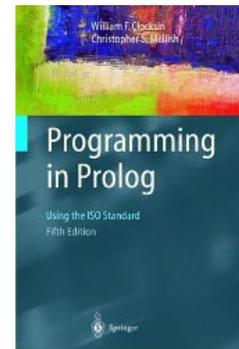
## CLIPS

- Joseph C. Giarratano & Gary D. Riley  
**Expert Systems:  
Principles and Programming**  
Thomson, 4<sup>th</sup> edition, 2005.  
ISBN 0534384471



## PROLOG

- William F. Clocksin  
& Christopher S. Mellish:  
**Programming in Prolog**  
Springer, 5<sup>th</sup> edition, 2003  
ISBN 3540006788



# Bibliografía: Sistemas Expertos

## Bibliografía complementaria

- John Durkin, **Expert Systems: Design and Development**. Prentice Hall, 1994.
- Peter Jackson, **Introduction to Expert Systems**. Addison Wesley, 1998.
- James P. Ignizio, **Introduction to Expert Systems: The Development and Implementation of Rule-Based Expert Systems**. McGraw-Hill, 1990.
- Ernest Friedman-Hill, **Jess in Action: Java Rule-Based Systems**. Manning Publications, 2003.
- Jay Liebowitz (editor): **The Handbook of Applied Expert Systems**. CRC Press, 1997.

