



**SEP**  
SECRETARÍA  
DE EDUCACIÓN  
PÚBLICA



Instituto Politécnico Nacional  
"La Técnica al Servicio de la Patria"



# Inteligencia Artificial



## Unidad III: Modelos de representación del conocimiento

## 3.1 Conocimiento

- ¿Qué es el conocimiento? Esta es una cuestión que ha sido discutida desde los antiguos griegos, y aún no está totalmente desmitificada. Parte del misterio que rodea al conocimiento se debe a la naturaleza de las proposiciones. Lo que importa acerca de las proposiciones es que son entidades abstractas que pueden ser verdaderas o falsas, correctas o incorrectas.
- Davenport y Prusak concluyen que el conocimiento corresponde a un conjunto de experiencias, valores, información contextualizada, procedimientos y modelos mentales que sirven como marco para la acción y el aprendizaje.
- Milton ha dicho que el conocimiento equivale a la experiencia en cierto campo o dominio, a la habilidad que tienen las personas para hacer de forma efectiva y eficiente su trabajo y resolver situaciones complejas. Para Galinsky el conocimiento se refiere a las conceptualizaciones representadas que explican la naturaleza tanto del mundo real como del mundo abstracto.

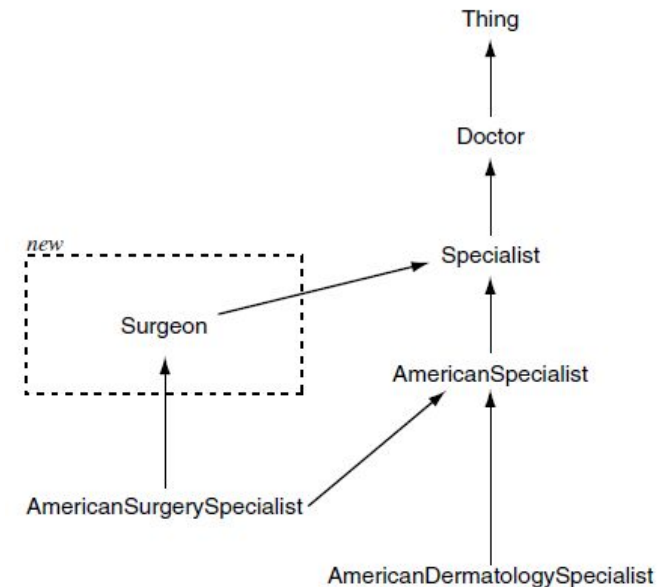
### 3.1.1 Características y taxonomía

- La adquisición del conocimiento debe llevarse a cabo de la forma más eficientemente posible, por lo que se hace necesario hacer una buena planeación para que se puedan maximizar los resultados obtenidos y usar la menor cantidad de recursos.
- En la práctica, hay una pequeña cantidad de preguntas clave que normalmente se le harían a una base de conocimientos de lógica de descripción. Debido a que se asemejan a bases de datos, donde los conceptos corresponden aproximadamente a elementos de un esquema y las constantes corresponden a registros.
- Una de las razones clave para usar una lógica de descripción en primer lugar es explotar el hecho de que los conceptos se consideran naturalmente organizados jerárquicamente, con los más generales en la parte superior y los más especializados más abajo. Esta taxonomía nos permitirá responder consultas de manera mucho más eficiente, lo que requiere un tiempo que en muchos casos crece linealmente con la profundidad de la taxonomía, más que con su tamaño.

### 3.1.1 Características y taxonomía (Cont.)

Una vez que tengamos una taxonomía de conceptos, podemos considerar agregar una oración para algún nuevo concepto atómico o constante. Esto implica la creación de algunos enlaces desde el concepto nuevo o constante a los existentes en la taxonomía, y tal vez redirigir algunos enlaces existentes. Este proceso se llama clasificación.

- Debido a que la clasificación en sí misma explota la estructura de la taxonomía, el proceso puede ser eficiente. Además, podemos pensar en construir toda la taxonomía por clasificación: comenzamos con un solo concepto en la taxonomía y luego le agregamos nuevos conceptos atómicos y constantes de forma incremental. En la figura se muestra una taxonomía que parte de definiciones.





### 3.1.1 Características y taxonomía (Cont.)

- Si construimos de esta manera una taxonomía correspondiente a una base de conocimientos, estamos en condiciones de responder a las preguntas clave de la lógica descriptiva con bastante facilidad. Para encontrar todas las constantes que satisfacen un concepto de consulta,  $q$ , simplemente clasificamos  $q$  y luego recopilamos todas las constantes en la periferia del árbol debajo de  $q$ . Esto implicaría un simple paseo por el árbol solo en la parte de la taxonomía subterendida por  $q$ . De manera similar, para encontrar todos los conceptos atómicos que se satisfacen con una constante  $c$ , comenzamos en  $c$  y subimos por el árbol, recopilando todos los nodos de conceptos a los que se puede llegar siguiendo los enlaces que representan la subsunción.

### 3.1.2 Modelos y sus características

- Los modelos de inteligencia artificial ayudan a automatizar la inferencia lógica y la toma de decisiones. Ayuda a hacer que los análisis sean más inteligentes y rápidos, con la capacidad de escalar al mismo tiempo que cantidades de datos siempre en crecimiento.
- Tras la recogida de datos y la preparación de estos, se crean modelos inteligentes de aprendizaje automático para admitir análisis avanzados. Estos modelos utilizan diversos tipos de algoritmos, como la regresión lineal o logística, para reconocer patrones en los datos y sacar conclusiones de una forma que emula la pericia humana. En pocas palabras, el modelado de inteligencia artificial es la creación de un proceso de toma de decisiones que sigue tres pasos básicos:

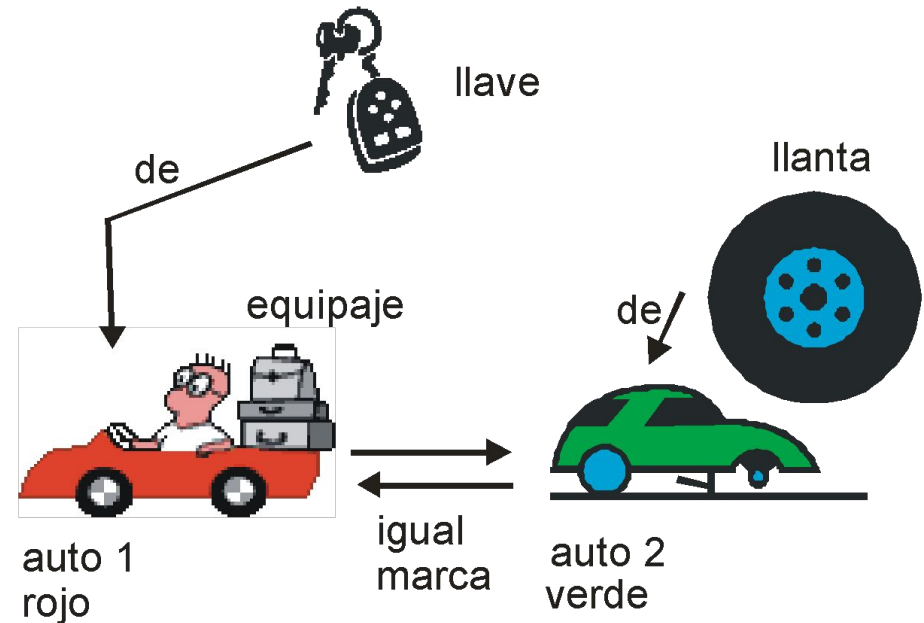


- **Modelado:** el primer paso es crear un modelo de inteligencia artificial que utiliza un algoritmo complejo o capas de algoritmos que interpretan los datos y toman decisiones basadas en ellos. Un modelo de inteligencia artificial exitoso puede actuar como un sustituto de la pericia humana en cualquier caso de uso.
- **Entrenamiento de modelo de inteligencia artificial:** el segundo paso es entrenar el modelo de inteligencia artificial. La mayoría de las veces, el entrenamiento implica el procesamiento de grandes cantidades de datos a través del modelo de inteligencia artificial en bucles de prueba iterativos y la comprobación de los resultados para garantizar la precisión y que el modelo se comporte como se espera y desea. Durante este proceso, los ingenieros están presentes para modificar y mejorar el modelo de inteligencia artificial a medida que aprende.
- **Inferencia:** Este paso se refiere al despliegue del modelo de inteligencia artificial en su caso de uso en el mundo real, donde el modelo de inteligencia artificial infiere habitualmente conclusiones lógicas basadas en los datos disponibles.

## 3.1 Lógica de primer orden

La lógica de primer orden supone:

- Objetos
- Propiedades
- Relaciones
- Funciones



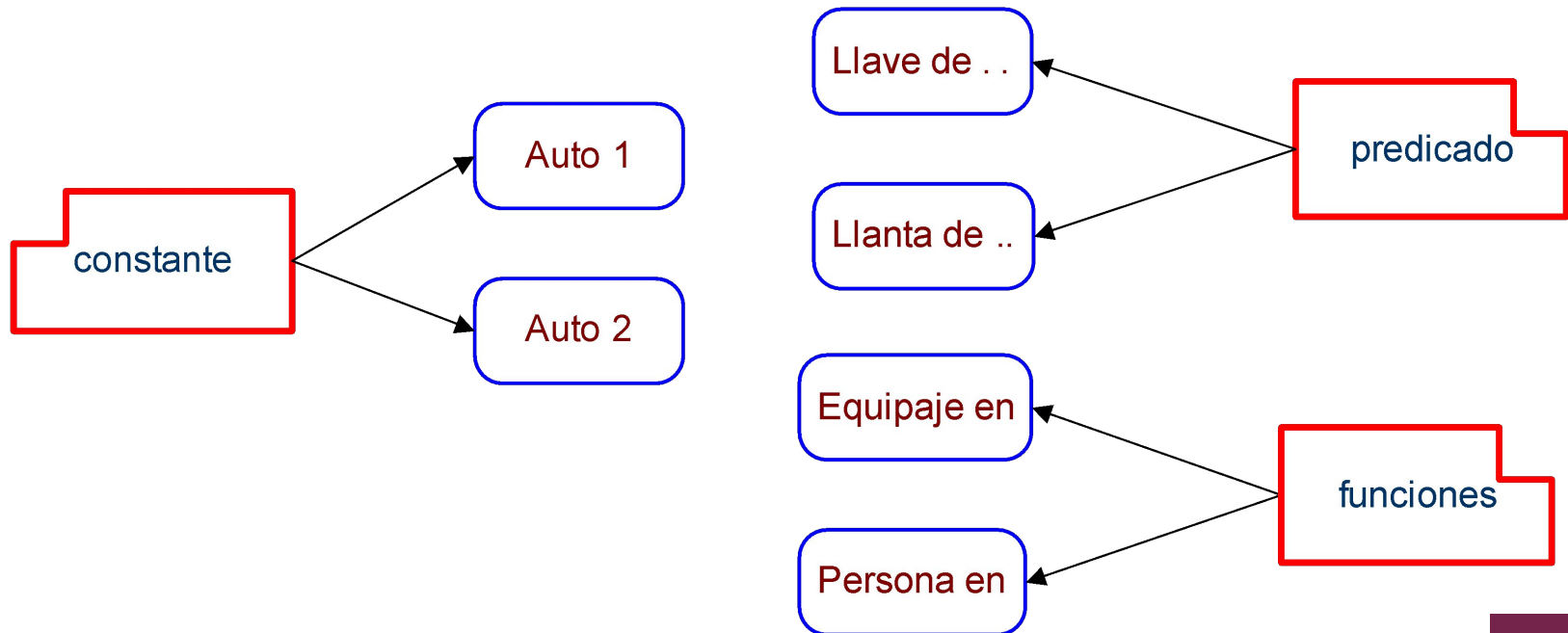
“existe un punto de salto si en su adyacencia existe aumento de la gravedad”



## 3.2.1 Razonamiento en la lógica proposicional

### Sintaxis y semántica

- Símbolos de constantes      objetos
- Símbolos de predicados      relaciones
- Símbolo de funciones      funciones



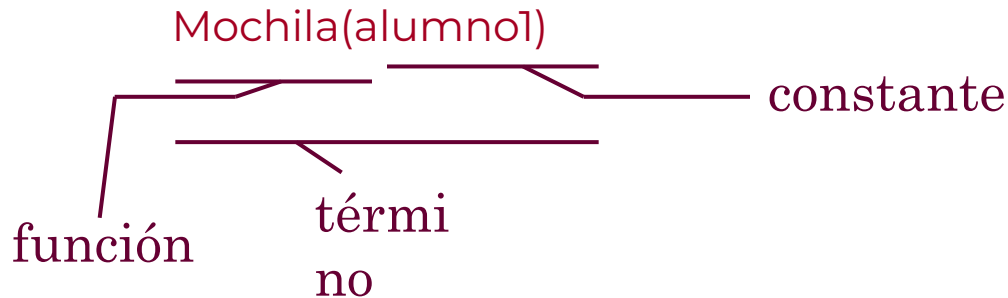
## Continuación...

Sentencia	→	SentenciaAtómica (Sentencia Conexión Sentencia) Cuantificador Variable, ... Sentencia ¬ Sentencia
SentenciaAtómica	→	Predicado(términos,...)
Término	→	Función(término,...) Constante Variable
Conexión	→	$\Rightarrow$ $\wedge$ $\vee$ $\Leftrightarrow$
Cuantificador	→	$\forall$ $\exists$
Constante	→	A $X_1$ Juan ...
Variable	→	a x s ...
Predicado	→	antes de Color lloviendo ...
Función	→	madre hoja derecha ...

## Continuación ...

### Término

Un término es una expresión lógica que se refiere a un objeto. Los símbolos constantes son términos. Los términos también se pueden construir a partir de símbolos de funciones y símbolos de constantes.





## Continuación...

- Sentencia Atómica
  - Una sentencia atómica está formada por un símbolo predicado seguido por una lista entre paréntesis de términos
    - **Compañeros(alumno1,alumno2,...)**
    - **Grupo(muj(alumna1, alumna2,...), hom(alumno1,alumno2,...))**
- Sentencia compleja
- Se pueden usar conexiones lógicas para construir Sentencias complejas.
  - **Grupo(alumno1)  $\vee$  Grupo(alumno2)**

### 3.2.2 Lenguaje de predicados

- La lógica de proposiciones es conveniente para representar conocimiento en donde no sea necesario formalizar propiedades entre individuos o relaciones entre ellos. Por ejemplo, la sentencia: "Algunos animales son domésticos", carece de significado hasta que no se determine a qué animales nos referimos. Si dicha sentencia se formaliza con el lenguaje de proposiciones tenemos la fórmula:  $p$ , expresión que no expresa conocimiento acerca de los individuos que pueden ser animales domésticos. De manera similar la proposición: "Todos los animales son domésticos" se representaría por la fórmula:  $q$ , que tampoco nos permite referirnos a todos los elementos de un dominio.
- El lenguaje de predicados, también conocido como lenguaje de la lógica de primer orden, generaliza al lenguaje proposicional introduciendo nuevos elementos del lenguaje con los que se describen con más detalle los elementos sintácticos de una proposición; ésta se formaliza atendiendo a los individuos.

### 3.2.2 Lenguaje de predicados (Cont.)

Cuantificadores:

Universal ( $\forall$ ): se utiliza para afirmar que todos los elementos de un conjunto cumplen una determinada propiedad o relación.

Existencial ( $\exists$ ): se utiliza para indicar que hay uno o más elementos de un conjunto que cumplen una determinada propiedad.

Ejemplo En el MC =  $\{N(x): x \text{ es número natural}; Po(x): x \text{ es positivo}\}$  las sentencias:

a) "Todos los números naturales son positivos", se formaliza como:  $\forall x[N(x) \rightarrow Po(x)]$ .

b) "Algunos números naturales son positivos", se formaliza como:  $\exists x[N(x) \wedge Po(x)]$ .

## Continuación...

### Cuantificadores

#### Cuantificador universal $\forall$

“todos los alumnos son personas”

$\forall x \text{ Alumno}(x) \Rightarrow \text{Persona}(x)$

(para todo  $x$  si  $x$  es un alumno, entonces  $x$  es una persona)

$x$  es una variable

$\forall x P$  es verdadero en un modelo dado bajo una interpretación  
si  $P$  es verdadero en todas sus posibles interpretaciones  
extendidas, donde cada interpretación especifica una  $x$ .

$x \rightarrow \text{alumno1}$

$x \rightarrow \text{mochila}$

$x \rightarrow \text{alumna3}$

## Continuación...

Cuantificador existencial  $\exists$

"El alumno tiene un compañero que es una persona"

$\exists x \text{ Compañero}(x, \text{alumno}) \wedge \text{Persona}(x)$

$\exists x P$  es verdad si  $P$  es verdad para algún objeto en el mundo.

Anidación de cuantificadores

"Todos somos buenos para alguna cosa"

$\forall x \exists y \text{ Buenopara}(x,y)$

"Para todo  $x$ ,  $x$  no acepta alumnos flojos"

$\forall x \neg \text{Acepta}(x, \text{AlumnosFlojos})$

"No existe un  $x$  que acepte alumnos flojos"

$\neg \exists x \text{ Acepta}(x, \text{AlumnosFlojos})$



## Continuación...

### Igualdad

"el compañero del alumno1 es el alumno 2"

Compañero (alumno1)=alumno2

"Hay un "x" y un "y" que son compañeros de alumno1 y no son el mismo individuo."

$\exists x,y \text{ Compañero}(\text{alumno1},x) \wedge \text{Compañero}(\text{alumno1},y) \wedge \neg(x=y)$

Para restringir número de objetos con una propiedad:

"Todo par de objetos con la propiedad P son iguales."

$\forall x,y P(x) \wedge P(y) \Rightarrow x=y$

Esta afirmación los restringe a ser un objeto con la propiedad P.  
o bien

$\exists x P(x) \wedge y P(y) \Rightarrow x=y$

## Continuación...

### Equivalencias

$$\neg ( \forall \Omega ) A(\Omega) \equiv ( \exists \Omega ) \neg A(\Omega)$$

$$\neg ( \exists \Omega ) A(\Omega) \equiv ( \forall \Omega ) \neg A(\Omega)$$

Renombrando variables

$$( \forall x ) A (x) \equiv ( \forall y ) A(y)$$

Unicidad  $\exists!$

$$\exists!x \text{ Rey}(x)$$

Existe una persona x que es Rey

### 3.2.2 Lenguaje de predicados (Cont.)

- Alfabeto: Conjunto de símbolos formado por términos, que pueden ser constantes, variables o funciones, predicados, cuantificadores y símbolos heredados del lenguaje proposicional (conectivas, variables proposicionales y paréntesis).
- Constantes: Representan objetos concretos como Juan, Sócrates.... Las constantes son los elementos llamados dominio de referencia o universo de discurso, que son distinguibles entre sí por sus propiedades. Se formalizan con letras, que pueden estar numeradas, empezando por minúscula: a, b, c, a1, a2, a3
- Variables: Se refieren a objetos indeterminados que pertenecen a un conjunto de referencia donde se define el razonamiento que se debe estudiar. Se formalizan con letras típicas de variables: x, y, z, x1, x2...

### 3.2.2 Lenguaje de predicados (Cont.)

Función n-aria (aridad<sup>2</sup> n): Representan a sujetos constantes que se obtienen aplicando una función a otro objeto constante. Pueden ser:

- Monádicas ( $n=1$ ): representan un individuo en función de otro.  
Ej: " $f(x) = x$ ", es una función monódica que calcula el cuadrado de un número real " $x$ ".
- Poliádicas ( $n>1$ ): representan un individuo en función de otros.  
Ej: " $f(x,y) = x + y$ ", es una función binaria que calcula la suma de dos números reales.
- Toda función tiene un número  $n \in \mathbb{N} \cup \{0\}$  de argumentos, que se conoce como la aridad de la función. Las funciones se formalizan con letras típicas de función:  $f, g, h, \dots$
- Constantes ( $n=0$ ): representan a las proposiciones atómicas.
- Ej: "María es estudiosa";  $Fbf: p$ , siendo  $MC = \{p: \text{María es estudiosa}\}$ .

### 3.2.2 Lenguaje de predicados (Cont.)

Predicados o relaciones n-arias: Según la aridad los predicados pueden ser:

- Constantes ( $n=0$ ): representan a las proposiciones atómicas.
- Ej: "María es estudiosa"; Fbf:  $p$ , siendo  $MC = \{p: \text{María es estudiosa}\}$ .
- Monádicos ( $n=1$ ): representan a las propiedades de los individuos.
- Ej: "María es estudiosa"; Fbf:  $Es(ma)$ , siendo  $MC = \{ma: \text{María}; Es(x): x \text{ es estudiosa}\}$ .
- Polliádicos ( $n>1$ ): representan a las relaciones entre individuos. Ej: "María es novia de Carlos"; Fbf:  $Nov(ma, ca)$ ,  $MC = \{ma: \text{María}; ca: \text{Carlos}; Nov(x,y): x \text{ es novio de } y\}$

Al igual que las funciones, los predicados tienen un número  $n \in \mathbb{N} \cup \{0\}$  de argumentos que determinan el carácter de dicho predicado. Los predicados se formalizan con letras o expresiones cuya primera letra debe estar en mayúscula:  $P, Q, R_a$  ....

### 3.2.2 Lenguaje de predicados (Cont.)

Ejemplo : "Todos los hombres son mortales"

P2: "Sócrates es un hombre"

Q: "Sócrates es mortal" Si formalizamos este razonamiento con el lenguaje de proposiciones con el marco conceptual:

MC = {Todos los hombres son mortales:  $p$ ; Sócrates es un hombre:  $q$ ; Sócrates es mortal:  $r$ }.

El razonamiento  $p, q \Rightarrow r$ , es, a todas luces, incorrecto. La conclusión " $r$ " no tiene ninguna relación sintáctica con las premisas ya que al ser éstas atómicas, no existe ninguna forma de extraerla de ellas; por otro lado, semanticamente, podemos interpretar las fórmulas  $p$  y  $q$  como verdaderas y la fórmula  $r$  como falsa, por lo que el razonamiento dado no es correcto; pero dicho razonamiento es correcto.

Se puede demostrar con otra formalización del lenguaje de predicados: Es decir si MC = { $s$ : Sócrates;  $H(x)$ :  $x$  es hombre;  $Mo(x)$ :  $x$  es mortal}, se tiene:  $\forall x [H(x) \rightarrow Mo(x)]$ ,  $H(s) \Rightarrow Mo(x)$ , como podemos ver ahora la relación sintáctica es evidente, luego ya podremos averiguar si la conclusión se deduce de las premisas.

### 3.2.2 Lenguaje de predicados (Cont.)

#### Ejemplo

- a) La variable proporcional “p” es un predicado constante (aridad cero), luego es una fórmula atómica.
- b)  $P(x)$ , es un predicado monádico que atribuye al sujeto “x” la propiedad P. Si el conjunto de referencia es:

$D = \{\text{ana, luis}\}$ , al sustituir la variable “x” por “ana” se obtiene la fórmula atómica:  $P(\text{ana})$ .

- c)  $P(x,y)$ , es un predicado poliádico que relaciona a los sujetos “x” e “y”. Al igual que en b), si “ $x = \text{ana}$ ” e “ $y = \text{luis}$ ”, tenemos la fórmula atómica:  $P(\text{ana, luis})$ , que representa que los sujetos “ana” y “luis” están relacionados mediante el predicado “P”.

### 3.2.2 Lenguaje de predicados (Cont.)

Ejemplo: Un predicado no es una función. Dado el conjunto  $D = \{\text{personas}\}$ . El predicado binario " $\text{Ma}(x,y)$ :  $x$  es la madre de  $y$ " no es una función porque dependiendo del valor que tomen las variables " $x$ " e " $y$ ", la fórmula correspondiente será verdadera o falsa. Por ejemplo, si " $x=\text{ana}$ " e " $y=\text{luisa}$ ", la fbf:  $\text{Ma}(\text{ana}, \text{luisa})$  será verdadera si el sujeto " $\text{ana}$ " es la madre del sujeto " $\text{luisa}$ ". Si hubiésemos querido usar una función podríamos haber puesto " $f(x)=y$ ", donde " $f$ ", calcula la madre " $y$ " del sujeto " $x$ ".

Ejemplo "Félix es el padre de Clara"

- 1.- Formalización mediante funciones:  $\text{padre}(\text{clara})$ , da como resultado: Félix.
- 2.- Formalización sin funciones:  $\text{Pa}(\text{felix}, \text{clara})$ , da como resultado verdadero o falso.



### 3.2.2 Lenguaje de predicados (Cont.)

Símbolos del lenguaje proposicional: variables proposicionales, conectivas y paréntesis.

Reglas para construir fórmulas de la lógica de primer orden bien formadas (fbf).

Las expresiones bien formadas de la lógica de predicados, es decir las fórmulas predicativas, se construyen a partir de los términos (que representan objetos) y de los predicados cuyos argumentos son, precisamente, los términos.

Definición recursiva de término:

Todo símbolo de variable y de constante es un término.

Si  $f$  es un símbolo de función de aridad  $n > 0$  y  $t_1, t_2, \dots, t_n$  son términos, entonces  $f(t_1, t_2, \dots, t_n)$  es un término.

Sólo son términos los definidos por 1 o 2.



### 3.2.2 Lenguaje de predicados (Cont.)

Ejemplo Las siguientes expresiones son términos:  $x$ ;  $a$ ;  $f(x)$ ;  $g(x, y)$ ;  $g(x, f(x))$ ; donde  $x$  es una variable,  $a$  es una constante,  $f$  es una función monádica y  $g$  es una función binaria.

#### • Definición de Fórmula:

Una fórmula predicativa puede ser atómica o molecular. Las fórmulas atómicas del lenguaje son las expresiones  $P(t_1, \dots, t_n)$ , donde  $P$  es un símbolo de predicado de aridad  $n > 0$  y  $t_1, \dots, t_n$  son términos constantes o de función.

Las fórmulas moleculares del lenguaje están formadas por fórmulas atómicas conectadas o por fórmulas en donde aparecen predicados con argumentos variables cuantificados. Si  $A$  es una fórmula y  $x$  es una variable, entonces  $(\forall x A(x))$  y  $(\exists x A(x))$  son fórmulas.

### 3.2.2 Lenguaje de predicados (Cont.)

- Ejemplo Formalizar la sentencia S: "María es rubia", con el lenguaje de predicados
- $MC = \{ma: \text{María}; R(x): x \text{ es rubia}\}$
- Fbf-S:  $R(ma)$
- Explicación de los símbolos usados:
  - -  $ma$  es un término que representa al sujeto: "María".
  - -  $R(ma)$  es una fórmula atómica que representa al hecho: "María es rubia".
- **Reglas de construcción de fórmulas bien formadas de la lógica de primer orden (fbf):**
  - **R.1.** Toda variable proposicional es una fbf.
  - **R.2.** Si  $P$  es un predicado, entonces  $P(t_1, t_2, \dots, t_n)$  es una fbf, siendo  $t_i$  términos.
  - **R.3.** Si  $F$  es una fbf entonces:  $\forall x_i F [x_1, x_2, \dots, x_i, \dots, x_n]$  ;  $\exists x_i F [x_1, x_2, \dots, x_i, \dots, x_n]$  son fbfs. La variable  $x_i$  queda ligada al cuantificador introducido y las otras variables s libres.
  - **R.4.** Si  $A$  y  $B$  son fbf entonces  $\neg A$ ,  $A \wedge B$ ,  $A \vee B$ ,  $A \rightarrow B$  y  $A \leftrightarrow B$  son fbf.
  - **R.5.** Sólo son fbfs las construidas desde R1 hasta R4.
  - **R.6.** Son válidas todas las reglas de la gramática proposicional.

### 3.2.2 Lenguaje de predicados (Cont.)

- Ejemplo La siguiente fórmula es sintácticamente correcta:  $\forall x [P(x) \wedge Q(z) \rightarrow \exists y R(x, y)]$  ? (justificar)
- ya que todas las subfórmulas de la fórmula dada son válidas
- $R(x, y)$  por R2
- $\exists y R(x, y)$  por R3
- $P(x)$  por R2
- $Q(z)$  por R2
- $P(x) \wedge Q(z)$  por R4
- $P(x) \wedge Q(z) \rightarrow \exists y R(x, y)$  por R4
- $\forall x ( P(x) \wedge Q(z) \rightarrow \exists y R(x, y) )$  por R3.

### 3.2.2 Lenguaje de predicados (Cont.)

#### Fórmula abierta y cerrada

Las fórmulas lógicas de primer orden, que denotaremos por  $\phi$ , pueden ser abiertas o cerradas según la "forma" en que aparezcan las variables que conforman los argumentos de los predicados declarados en la formalización. Si las variables están afectadas por algún cuantificador se dice que son variables ligadas y si no lo están, son variables libres.

- Si toda variable que aparece en una fórmula es ligada, diremos que la fórmula es cerrada. En este caso la fórmula puede ser evaluada como verdadera o falsa.
- Si en una fórmula aparece una variable libre, diremos que la fórmula es abierta. En este caso la fórmula no puede evaluarse ni a verdadera ni a falsa.
- Ejemplo 8
- 1) En la fórmula:  $\forall xP(x, y)$  la variable  $x$  aparece ligada a  $y$  la variable  $y$  aparece libre. La fórmula es abierta.
- 2) En la fórmula:  $\forall x[P(x) \rightarrow Q(x) \wedge R(x)]$  la variable  $x$  aparece ligada al cuantificador existencial en todos los predicados. La fórmula es cerrada.

## Continuación ...

### Dominio de referencia o Universo de Discurso

Es el conjunto no vacío de individuos distinguibles entre sí en el que se definen sus relaciones y propiedades, es decir es el conjunto de cosas acerca de las cuales se habla en un determinado contexto. Dependiendo del dominio elegido una misma proposición puede ser verdadera o falsa. Por ejemplo, al decir "Todos los números son positivos", en el conjunto  $D = \{1,2,3,4\}$  la proposición  $\forall x P_o(x)$ , siendo  $P_o(x)$  el predicado que se refiere a que el sujeto  $x$  es positivo, es verdadera, pero en el conjunto  $D_2 = \{-2,-1,1,2\}$  la proposición es falsa.

Por convenio, el dominio de discurso es siempre un conjunto no vacío.

Cuando tenemos un dominio finito el cuantificador universal se puede considerar como una generalización de la conjunción y el cuantificador existencial como una generalización de la disyunción.

Sea el dominio  $D = \{a^1, a^2, \dots, a^n\}$ , tenemos que la fbf:

- $\forall x P(x)$  es equivalente a:  $P(a^1) \wedge P(a^2) \wedge \dots \wedge P(a^n)$
- $\exists x P(x)$  es equivalente a:  $P(a^1) \vee P(a^2) \vee \dots \vee P(a^n)$ .



## Ejercicios propuestos

1. La mochila es rosa.
2. IA es interesante.
3. Pooh es un oso.
4. Pooh es un oso, los osos son animales, los animales son seres vivos.
5. El agua es líquida entre 0 y 100°C.
6. El pescado es fresco si no tiene los ojos hundidos, si no tiene los ojos opacos, si no huele feo.
7. Los alumnos de IA son del grupo 6CM5 pero estos alumnos toman clase de otras asignaturas en otros grupos.
8. Los alumnos de IA no deben entregar las tareas tarde como lo hace Juanita.
9. El libro es de IA, el libro es para estudiar y debe de ser cuidado, si no lo cuidas se deshojará.
10. Eres un tonto, no tu eres el tonto, no tu eres una vaca, las vacas son tontas, tu eres la vaca

## Continuación...

### Más ejemplos

Se tiene un robot que reparte paquetes en una oficina

Paquete(x) Término de propiedad de una cosa (un paquete)

EnSala(x,S) Término de relación que denota que un objeto esta en una sala.

MásPequeño(x,y) Término de relación que denota que un paquete es más pequeño que otro.

Hallegado(x,z) Término de relación de tiempo, un paquete llevo un tiempo z antes que otro

Antes(z1,z2) Término de relación de tiempos.

### Sentencias:

Todos los paquetes que están en la Sala 1 son más pequeños que los que están en la Sala 2

$(\forall x,y) \{ [ \text{Paquete}(x) \wedge \text{Paquete}(y) \wedge \text{EnSala}(x,S1) \wedge \text{EnSala}(y,S2) ] \Rightarrow \text{MásPequeño}(x,y) \}$



## Continuación...

Cada paquete en la Sala 1 es más pequeño que uno de los paquetes de la Sala 3

$$(\exists y)(\forall x)\{[\text{Paquete}(x) \wedge \text{Paquete}(y) \wedge \text{EnSala}(x,S1) \wedge \text{EnSala}(y,S3)] \\ \Rightarrow \text{MásPequeño}(x,y)\}$$

$$(\forall x)(\exists y)\{[\text{Paquete}(x) \wedge \text{Paquete}(y) \wedge \text{EnSala}(x,S1) \wedge \text{EnSala}(y,S3)] \\ \Rightarrow \text{MásPequeño}(x,y)\}$$

Expresar que un paquete llega antes que otro:

$$(\exists z1\ z2)[\text{Hallegado}(A,z1) \wedge \text{Hallegado}(B,z2) \wedge \text{Anterior}(z1,z2)]$$



## Ejercicios propuestos

1. No todos los estudiantes cursan IA y Compiladores.
2. Sólo uno de los estudiantes reprobó IA.
3. La mejor de las calificaciones de IA es mejor que la mejor calificación de las obtenidas en Compiladores.
4. Toda persona a quien le desagrade los vegetarianos es poco inteligente.
5. A nadie le gusta un vegetariano poco inteligente.
6. Hoy por la noche en la TV pasan a House por si te interesa.
7. Quizás venga a la fiesta y quizás no venga.
8. Si vivieras aquí, podrías estar en casa ahora. Si ahora no estuvieras en casa, no estarías aquí. Por lo tanto, si vivieras aquí, no estarías aquí.

## 3.3 Razonamiento basados en reglas

### 3.3.1 Reglas de inferencia

Reglas

Si entonces

- condiciones - acciones
- antecedentes - consecuentes
- premisas – conclusiones
- Las reglas modelan el conocimiento general del dominio y constituyen:
  - Base de conocimiento (BC)
  - Memoria a largo plazo
  - Implicaciones

Ejemplo: Si x es un gato entonces x es un animal doméstico

- El motor de inferencia o mecanismo de control está compuesto de:
- Intérprete de reglas o mecanismo de inferencia
- Estrategia de control o estrategia de resolución de conflictos Función

### 3.3.1 Reglas de inferencia

Fases del ciclo básico:

#### 1. Detección (filtro): Reglas pertinentes

- Obtención, de la BC, del conjunto de reglas aplicables a una situación determinada (estado) de la BH; formación del conjunto de conflictos (intérprete de reglas). Construcción del conjunto de reglas aplicables. El intérprete de reglas realiza los cálculos necesarios para obtener las instanciaciones que son posibles en cada estado de resolución del problema (comparación o *matching*). Una regla puede instanciarse más de una vez, en caso de existir variables que lo permitan.

#### 2. Selección: ¿Qué regla?

- Resolución de conflictos: selección de la regla a aplicar (estrategia de control). Las reglas son o no aplicadas dependiendo de la estrategia de control:
  - estrategia fija
  - estrategia dinámica prefijada
  - estrategia guiada por meta-reglas

### 3.3.1 Reglas de inferencia (Cont.)

- Selección de la “mejor” instanciación
- Ejemplos de estrategia de control:
  - 1ª regla por orden en la base de conocimiento
  - la regla más/menos utilizada
  - la regla más específica (con más literales) / más general
  - la regla que tenga el grado de certeza más alto
  - la instanciación que satisfaga los hechos:
    - más prioritarios
    - más antiguos (instanciación más antigua)
    - más nuevos (instanciación más reciente)

- 3. Aplicación

Aplicación de la regla sobre una instanciación de las variables: modificación de la memoria de trabajo.

- Ejecución de la regla ⇒
  - Modificación de la base de hechos (en el razonamiento hacia delante)
    - – Nuevos cálculos, nuevas acciones, preguntas al usuario
    - – Nuevos sub-objetivos (en el razonamiento hacia atrás)

### 3.3.1 Reglas de inferencia (Cont.)

- Propagación de las instanciaciones
- Propagación del grado de certeza
- El proceso de deducción acaba cuando:
  - se encuentra la conclusión (el objetivo) buscado  $\Rightarrow$  éxito
  - no queda ninguna regla aplicable  $\Rightarrow$  éxito? / fracaso?
  - aplicación de todas las reglas (sólo si se quieren todas las soluciones posibles)
  - la regla usada más recientemente
  - meta-reglas, indican dinámicamente como seleccionar las reglas a aplicar:
- Posible combinación de criterios
- Tipos de razonamiento
  - Deductivo, progresivo, encadenamiento hacia delante, dirigido por hechos
    - evidencias, síntomas, datos  $\Rightarrow$  conclusiones
  - Inductivo, regresivo, encadenamiento hacia atrás, dirigido por objetivos
    - conclusiones  $\Rightarrow$  datos, evidencias, síntomas
  - Mixto, encadenamiento híbrido

### 3.3.1 Reglas de inferencia (Cont.)

- 4. Vuelta al punto 1, o parada si el problema está resuelto  
Si no se ha encontrado una solución y no hay reglas aplicables: fracaso.

### 3.3.2 Encadenamiento hacia adelante y hacia atrás

- Encadenamiento hacia adelante
- Basado en *modus ponens*:  $A, A \Rightarrow B \vdash B$
- La base de hechos (BH) se inicializa con los hechos conocidos inicialmente.
- Se obtienen las consecuencias derivables de la BH:
  - se comparan los hechos de la BH con la parte izquierda de las reglas; se seleccionan las reglas aplicables: las que tienen antecedentes conocidos (que están en la BH);
  - las nuevas conclusiones de las reglas aplicadas se añaden a la BH (hay que decidir cómo);
  - se itera hasta encontrar una condición de finalización.

Problemas:

- – No focaliza en el objetivo
- – Explosión combinatoria

Ventajas:

- – Deducción intuitiva
- – Facilita la formalización del conocimiento al hacer un uso natural del mismo



### 3.3.2 Encadenamiento hacia adelante y hacia atrás

- Encadenamiento hacia atrás

Basado en el *método inductivo*: va guiado por un objetivo que es la conclusión (o hipótesis) que se trata de validar reconstruyendo la cadena de razonamiento en orden inverso. Cada paso implica nuevos sub-objetivos: hipótesis que han de validarse.

- Funcionamiento:

- se inicializa la BH con un conjunto inicial de hechos;
- se inicializa el conjunto de hipótesis (CH) con los objetivos a verificar;
- mientras existan hipótesis a validar en CH se escoge una de ellas y se valida:
  - se comparan los hechos de la BH y la parte derecha de las reglas con las hipótesis;
  - si una hipótesis está en BH eliminarla de CH;
  - si no: buscar reglas que tengan como conclusión la hipótesis.
- Seleccionar una, añadir las premisas no satisfechas a CH como subobjetivos.

### 3.3.2 Encadenamiento hacia adelante y hacia atrás

- Encadenamiento híbrido
- Partes de la cadena de razonamiento que conduce de los hechos a los objetivos se construyen deductivamente y otras partes inductivamente: exploración bi-direccional
- El cambio de estrategia suele llevarse a cabo a través de meta-reglas. Ejemplos:
  - – función del número de estados iniciales y finales
  - – función de la dirección de mayor ramificación
  - – función de la necesidad de justificar el proceso de razonamiento
- Se evita la explosión combinatoria del razonamiento deductivo.

### 3.3.3 Ontologías (Cont.)

- Son Conceptos seleccionados para describir un dominio Vocabulario de predicados, funciones y constantes. Seleccionar alternativas - como nombrar los conceptos, cuando usar funciones, variables o constantes, etc. Decidir cómo organizar dichos conceptos (relaciones, jerarquías)
- Puede haber desde ontologías muy específicas para un problema, ontologías más amplias para un dominio, hasta ontologías genéricas

### 3.3.3 Ontologías (Cont.)

Elementos de una Ontología

Categorías – incluyen objetos con propiedades comunes arregladas en taxonomías jerárquicas.

- Se puede inferir la categoría de un objeto con base a sus propiedades, y hacer predicciones de otras propiedades del objeto.
- Se pueden obtener propiedades de un objeto de sus súper-clases a través de herencia.
- Las categorías permiten organizar y simplificar la adquisición del conocimiento.

Medidas – relaciona objetos a cantidades de tipos particulares (v.g., masa, edad, precios, etc.). Las medidas cuantitativas son en general fácil de representar. Otras medidas no tienen una escala de valores única (problemas, sabor, belleza, etc.)

Objetos compuestos – objetos complejos que se describen por su estructura constitutiva (partes). Se pueden tener jerarquías de tipo partes-de. Se pueden tener objetos compuestos sin estructura.

.

### 3.3.3 Ontologías (Cont.)

Tiempo, Espacio y Cambio – para permitir acciones y eventos con diferentes duraciones y que puedan ocurrir simultáneamente. La noción general es que el universo es continuo tanto en tiempo como en espacio.

Eventos y Procesos – eventos individuales ocurren en un tiempo y lugar particular. Los procesos son eventos continuos y homogéneos por naturaleza.

Objetos Físicos – al extender las cosas en tiempo y espacio, los objetos físicos tienen mucho en común con los eventos. A veces les llaman “fluentes”(fluents)

Substancias – temporales y espaciales. Existen propiedades intrínsecas que son de la sustancia del objeto más que del objeto mismo (color, temperatura en que se derrite, etc.), y propiedades extrínsecas (peso, forma, etc.)

Objetos Mentales y Creencias – se tiene que razonar acerca de creencias del mundo.

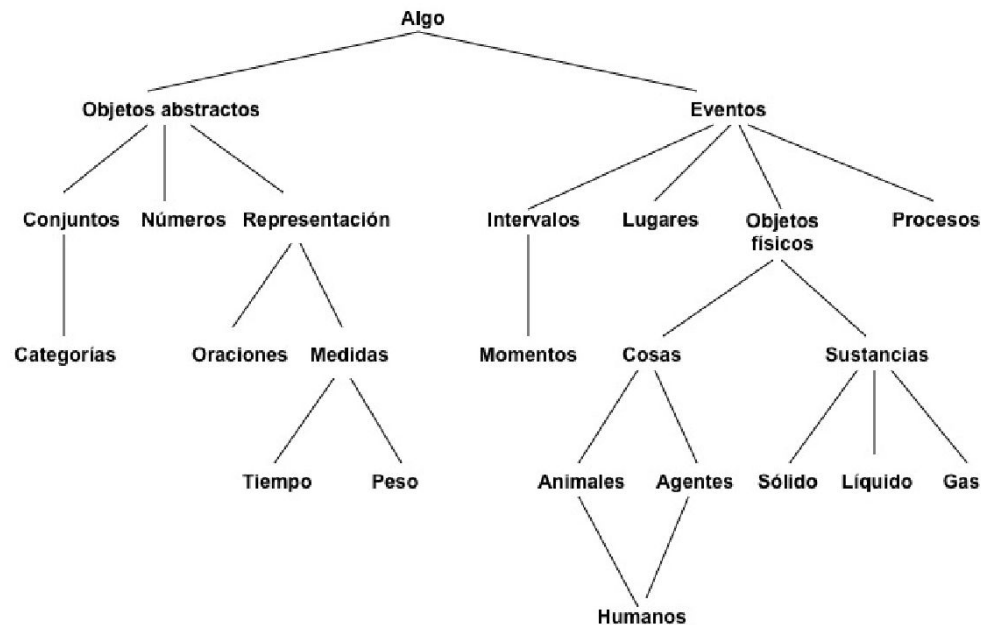
Eventos y Procesos – eventos individuales ocurren en un tiempo y lugar particular. Los procesos son eventos continuos y homogéneos por naturaleza.

Objetos Físicos – al extender las cosas en tiempo y espacio, los objetos físicos tienen mucho en común con los eventos. A veces les llaman “fluentes”(fluents)

### 3.3.3 Ontologías (Cont.)

Substancias – temporales y espaciales. Existen propiedades intrínsecas que son de la sustancia del objeto más que del objeto mismo (color, temperatura en que se derrite, etc.), y propiedades extrínsecas (peso, forma, etc.)

Objetos Mentales y Creencias – se tiene que razonar acerca de creencias del mundo.



### 3.4 Modelos de llenado de ranuras

- El *Llenado de ranuras* es un concepto de comprensión del lenguaje natural que significa guardar una entidad extraída en un objeto.



### 3.4.1 Redes semánticas y Marcos

Una red semántica es un grafo donde:

- los nodos representan conceptos
- los arcos (dirigidos) representan relaciones entre conceptos
- Mecanismos de razonamiento específicos permiten responder a preguntas sobre la representación:
  - ¿Están relacionados dos conceptos?
  - ¿Que relaciona dos conceptos?
  - ¿Cuál es el concepto mas cercano que relaciona dos conceptos?





## Marcos (frames)

- Ante un problema nuevo nadie empieza directamente un análisis exhaustivo, y desde cero. Por el contrario, el primer paso suele consistir en recuperar experiencias anteriores y tratar de razonar por “semejanza”.
- Las *frames*, se pueden describir como redes semánticas complejas que tratan el problema de la representación desde la óptica del razonamiento por semejanza. Describen clases de objetos y pueden definirse como representaciones estructuradas de conocimiento estereotipado.
- Estructuralmente, una frame consta de una *cabecera*, que le da nombre al frame (y que es representativa de la clase de objetos que se describen), y de un conjunto de *slots*, cada uno de los cuales representa una propiedad o atributo del elemento genérico representado por el frame. Cada slot puede tener distintos slots anidados y sin limitación de profundidad. De esta forma se habilitan posiciones concretas para ubicar sistemáticamente los componentes de nuestras experiencias anteriores en relación a la clase de elementos representados.

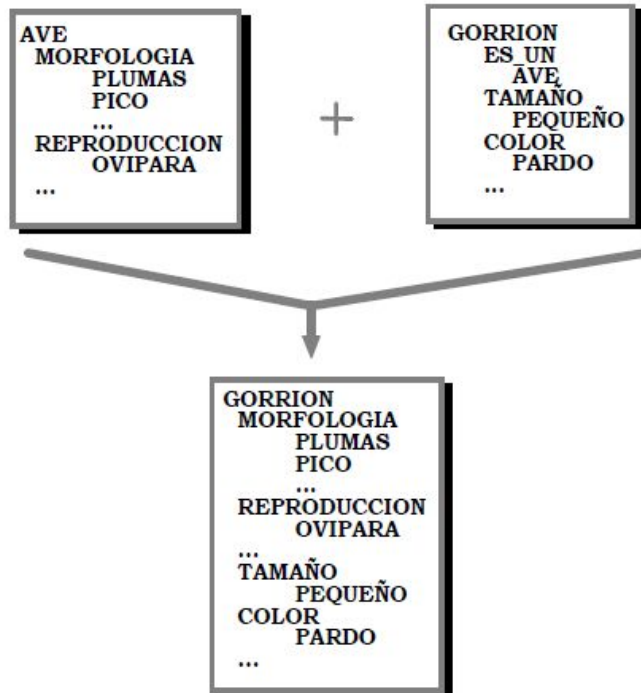
## Marcos (frames)

AUTOMOVIL  
TIPOS  
    TODO\_TERRENO  
    DEPORTIVO  
    UTILITARIO  
COMPONENTES  
    CARROCERIA  
    PUERTAS  
    CAPO  
    ....  
    ....

Cada una de las indentaciones de los slots representa un nivel de conocimiento o nivel epistemológico, y su contenido es una especialización del nivel anterior. Un sistema en el que el conocimiento esté representado por medio de frames utiliza con profusión el concepto de herencia. Para ello se emplean slots de tipo ES\_UN, que permiten la entrada de información a una frame, en un nivel de conocimiento determinado, y a partir del cual la información de la clase correspondiente pasa al objeto considerado.

## Marcos (frames)

- Las frames suelen incorporar también información procedimental. Para ello, ciertos slots llevan asociados procedimientos que la mayor parte del tiempo están inactivos, pero que cuando son activados desencadenan acciones concretas. Algunos de tales procedimientos denominados *demons*.



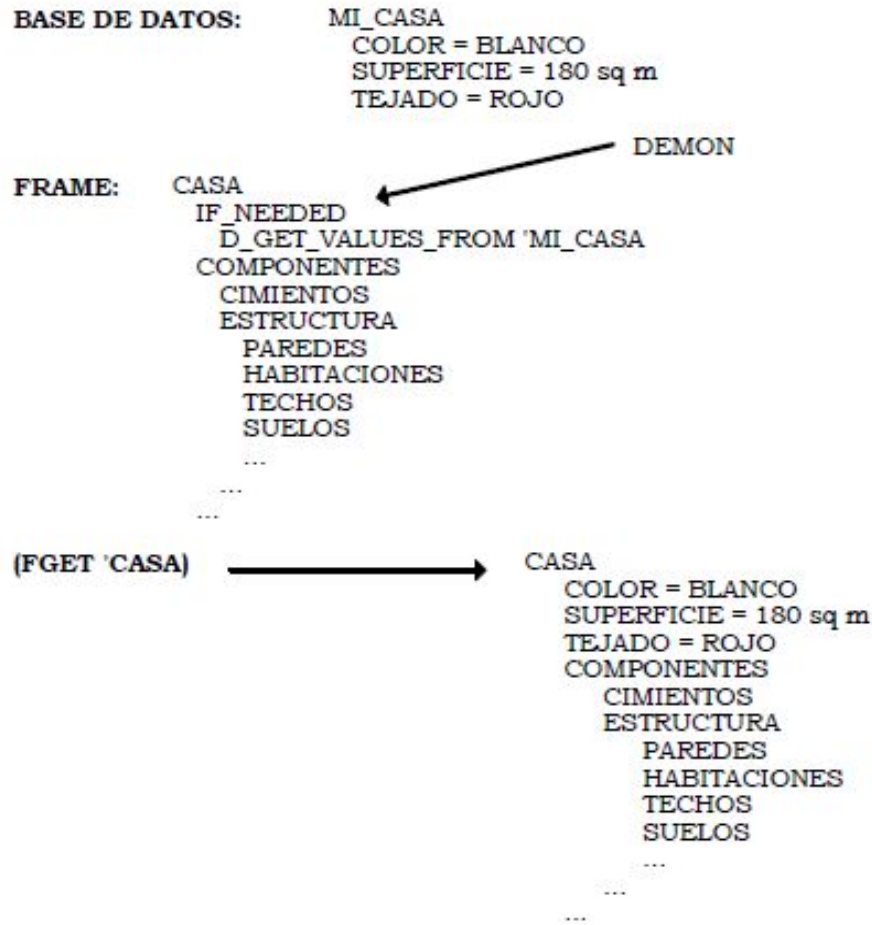
Cuando un demon es activado (por una entrada en la frame, al nivel correspondiente), el procedimiento que sigue al demon es ejecutado y, normalmente, el demon (cumplida su misión), es eliminado inmediatamente.

Los demons son estructuras muy útiles ya que: Proporcionan uniones procedimentales entre distintos frames y posibilitan la ejecución de rutinas externas. Imprimen un cierto carácter dinámico a la representación del conocimiento con frames. Como estructuras de representación, las frames pueden emplearse como:

Elementos descriptivos

Elementos de control del conocimiento

## Marcos (frames)



La Figura muestra una frame diseñada para controlar el formato de entrada de un cierto parámetro.

Con lo visto, podemos ya resaltar que: Las frames permiten definir procesos de razonamiento con información incompleta.

Las frames permiten inferir rápidamente hechos no representados de forma explícita.

Las frames imprimen cierto carácter dinámico a la representación al definir procesos que establecen relaciones entre otras frames y conexiones con el mundo exterior.

Las frames utilizan con profusión el concepto de herencia.

## Marcos (frames)

```
JOSE_PEREZ
  EDAD = 57
  HISTORIAL_CLINICO
    HEPATITIS
    TIPO = B
  COMPLICACIONES
    FALLO_RENAL
    BRONQUITIS_CRONICA
  MEDICACION
    DIURETICOS
```

Una frame como elemento descriptivo

```
HIPERTENSION
  ESPECIFICACIONES_DE_ENTRADA
    VALOR_TIPO = LITERAL
    MULTIPLICIDAD = NO
    IMPRESCINDIBILIDAD = NO
    INCERTIDUMBRE = NO
  ESTRATEGIA
    PREGUNTAR
    USAR_REGLAS
  ...
  ...
```

Una frame como elemento de control

El razonamiento con frames suele comenzar con la selección de una frame determinada que se ajuste a nuestra situación actual. Dado que en la mayoría de los casos no tendremos ninguna frame que describa exactamente nuestro estado inicial, comenzaremos seleccionando la más ajustada en base a la evidencia parcial disponible. A continuación, se produce la *ejemplificación* de la frame seleccionada tras considerar ciertas condiciones específicas actuales.

En general, el proceso de ejemplificación asocia un individuo particular con una clase. En otras palabras, obtenemos una descripción individual del problema actual considerando, por un lado, la descripción de la clase genérica y, por otro lado, las características específicas actuales.



## Marcos

- Un marco está generalmente dividido en:
  - – una parte **declarativa** (atributos o *slots*)
  - – una **procedimental** (métodos o *demons*)
- La parte procedimental permite obtener más información o hacer cálculos sobre sus características o las relaciones que pueda tener con otros marcos.
- La descripción de los atributos también está estructurada: un atributo puede tener propiedades (*facets*).
- En el caso más general se pueden tener taxonomías de atributos.

- Las **relaciones** poseen una descripción formal que establece su semántica y su funcionamiento.
- Dividimos las relaciones en dos simples clases:
  - **taxonómicas**:
  - enlace ES-UN (subclase/clase)
  - enlace INSTANCIA-DE (instancia/clase)
  - **no taxonómicas**

Los **atributos** poseen un conjunto de propiedades que permiten establecer su semántica: dominio, rango, cardinalidad, valor por defecto, métodos.

Permiten definir procedimientos de manera que se realizan cálculos bajo ciertos eventos (a través de los métodos).

## Marcos Relaciones

- Permiten conectar los marcos entre sí.
- Se define su semántica mediante un conjunto de propiedades: dominio, rango, cardinalidad, inversa, transitividad, composición
- • Se pueden establecer **métodos** que tienen efecto ante ciertos eventos (demonstrations):
  - *if-needed* (se activan al consultar el atributo);
  - *if-added* (se activan al asignar valor al atributo y se establece la relación entre instancias);
  - *if-removed* (se activan al borrar el valor del atributo y se elimina la relación entre instancias);
  - *if-modified* (se activan al modificar el valor del atributo).
- Se puede declarar como, el mecanismo de herencia afecta a los atributos.
- Se puede establecer el comportamiento de la relación respecto al mecanismo de herencia (que atributos permite heredar).

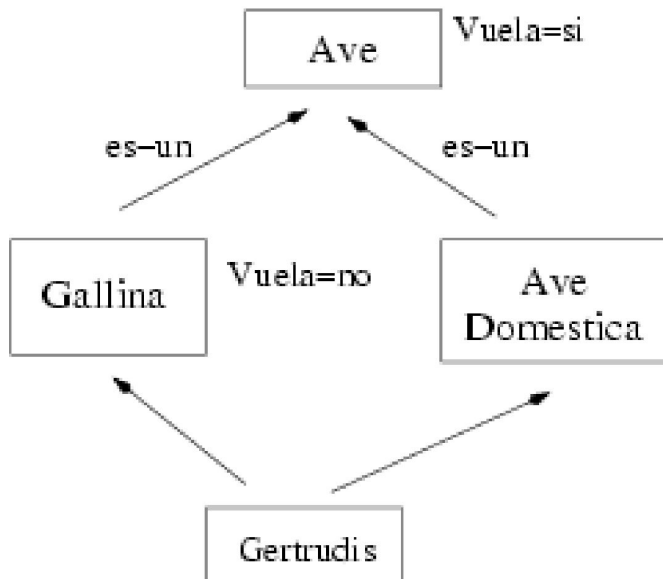


## Marcos: herencia

- La herencia permite obtener en un marco el valor o los valores de un atributo a través de otro marco con el que esta relacionado.
- En el caso de las **relaciones taxonómicas** (caso más común) la herencia (de atributos y valores) se da por defecto. En el **resto de las relaciones** se ha de establecer de manera explícita.
- Hay atributos no heredables. Ejemplo:
  - *tiene-por-instancia*
- Dado un marco es posible que la representación permita heredar un valor a través de múltiples relaciones (**herencia múltiple**): hay que establecer **criterios** (ejemplo: camino más corto).

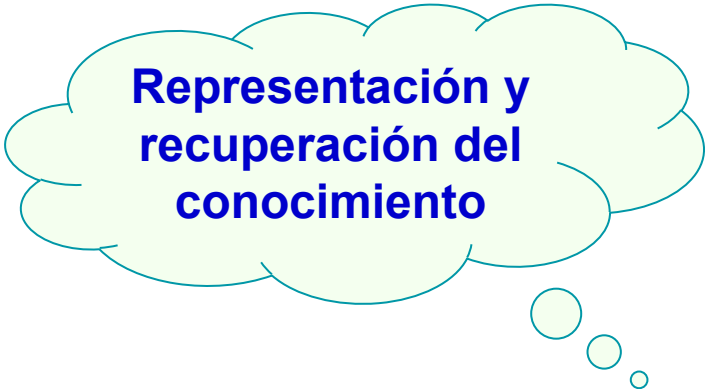
## Herencia simple y múltiple

- Si la taxonomía es un **árbol**, la herencia es **simple**.
- La herencia es **múltiple** si:
  - la taxonomía es un **grafo** (dirigido acíclico)
  - **hay otras relaciones** (no taxonómicas) que permiten herencia
- **¡Puede haber conflicto de valores!** (obviamente, sólo si hay herencia de atributo y valor)



- ¿Gertrudis vuela?
- Si considero el camino más corto: no. Pero, ¿si Gallina está sub-clasificado?
- El algoritmo de **distancia inferencial** permite establecer cuál es el marco del que se ha de heredar.

## 3.4.1 Redes semánticas y Marcos



**Representación y  
recuperación del  
conocimiento**

**Relaciones propias**

### **Redes IS-A.**

**Se basan en la relación de un subconjunto con uno más general.**

**Sentencias lógicas se pueden representar con una red de relación de subconjuntos semánticos con jerarquías.**

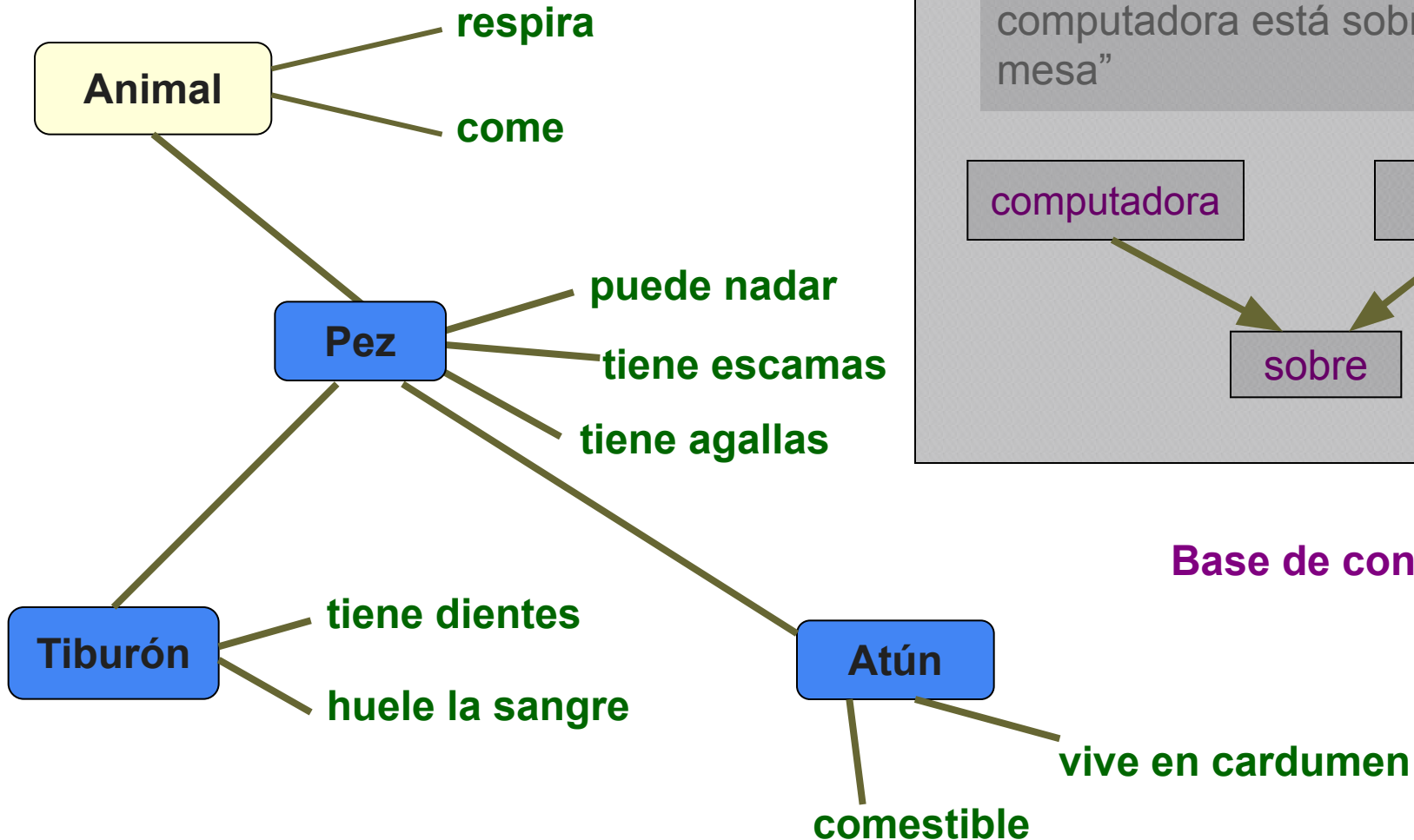
- ❖ **Representación restringida de la lógica de primer orden.**
- ❖ **Se establecen relaciones entre conceptos.**
- ❖ **Se puede establecer mecanismos de razonamiento.**

### **Tipos:**

- ❖ **Redes IS-A.- enlaces de nodos etiquetados.**
- ❖ **Grafos conceptuales.- nodos de concepto y de relación.**
- ❖ **Redes de marcos.- enlaces como parte de la etiqueta del nodo.**

## 3.4.1 Redes semánticas y marcos (cont.)

Ejemplo de una red semántica IS-A



Prototipo de red sem. de: “la computadora está sobre la mesa”

computadora

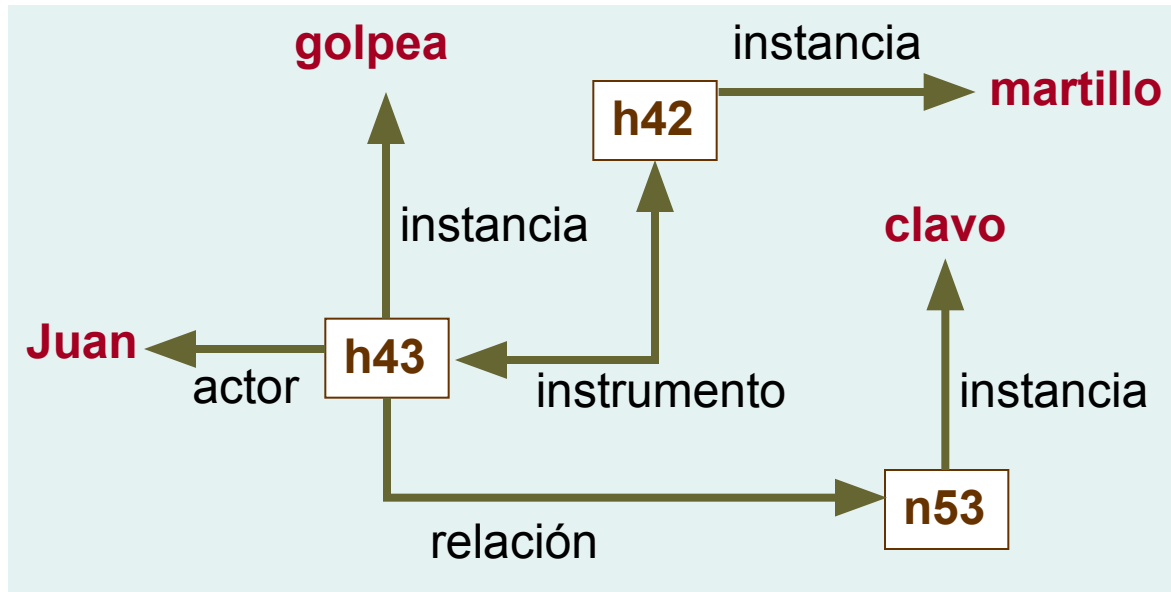
mesa

sobre

Base de conocimiento

## 3.4.1 Redes semánticas y marcos (cont.)

Red semántica para lenguaje natural  
"Juan golpea un clavo con un martillo"



Instancias:  
martillo  
clavo  
Golpea

h43 es una  
instancia de  
acción.

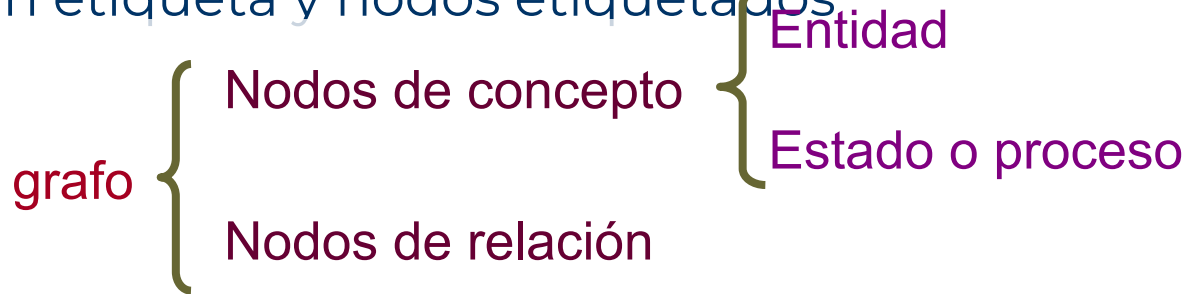
h43 y n53 crean  
una relación de  
instancias.

Arcos etiquetados


## 3.4.1 Redes semánticas y marcos (cont.)

Grafos conceptuales

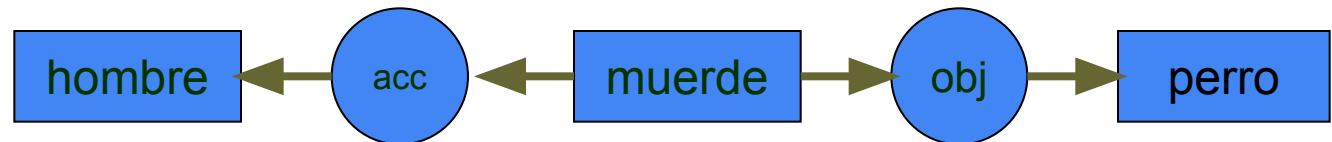
Arcos sin etiqueta y nodos etiquetados



 Nodos de concepto

 Nodos de relación

El hombre muerde al perro



## 3.4.2 Redes semánticas y marcos (cont.)

### Ejemplo de una red semántica

El pingüino y el albatros son aves que ponen huevos y tienen plumas y por lo tanto son animales. Ambos comen pescado, que también es un animal. El albatros es de color gris y puede volar, el pingüino es de color blanco y negro y no vuela pero puede nadar. Moni y Juan son albatros mientras que Pepe es un pingüino.

### Representación con lógica de primer orden:

**Albatros(Moni)**

$\forall x \text{ color}(x, \text{gris}) \Rightarrow \text{Albatros}(x)$

**Albatros(Juan)**

$\forall x \text{ color}(x, \text{negro}) \wedge \text{color}(\text{blanco}) \Rightarrow \text{pingüino}(x)$

**Pingüino(Pepe)**

$\forall x \text{ pescado}(x) \Rightarrow \text{animal}(x)$

$\forall x \text{ ave}(x) \Rightarrow \text{animal}(x)$

$\forall x \text{ ave}(\text{Albatros}(x))$

$\forall x \text{ ave}(\text{Pingüino}(x))$

$\forall x \text{ tiene}(x, \text{huevos}) \Rightarrow \text{ave}(x)$

$\forall x \text{ tiene}(x, \text{plumas}) \Rightarrow \text{ave}(x)$

$\forall x \text{ Albatros}(x) \Rightarrow \text{puede}(x, \text{volar})$

$\forall x \text{ Pingüino}(x) \Rightarrow \text{puede}(x, \text{nadar})$

$\forall x \forall y \text{ Albatros}(x) \wedge \text{Pingüino}(y) \Rightarrow \text{comer}(x, \text{pescado}) \wedge \text{comer}(y, \text{pescado})$

### 3. Tarea-Práctica

- 1.- Crear un programa orientado a objetos que pueda manejar una red semántica IS-A. con las opciones: 1.- introducir conocimientos, 2.-introducir palabra y 3.- salir.

El programa debe estar orientado al problema del albatros y el pingüino con la capacidad de introducir mas aves y peces con sus características y relaciones.

Con la opción 1 se introducirán nodos con nombres de peces, aves, o animales con sus atributos, el programa debe de ser capaz de relacionar atributos cuando se efectué una búsqueda de relaciones. Con la opción 2 se introducirá una palabra que se buscará en la base de conocimientos y se mostrará todo lo relacionado con dicha palabra.

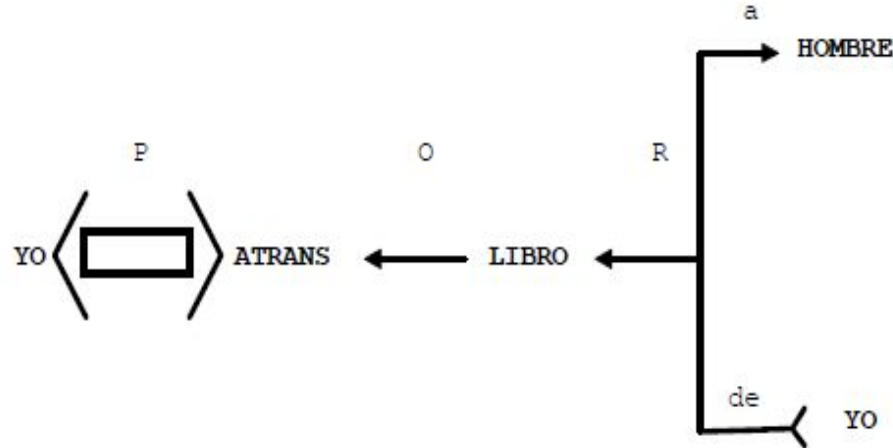
- 2.- Diseño de base de datos relacionales para el problema del albatros y pingüino.



### 3.4.2 Dependencia conceptual y guiones

- La dependencia conceptual intenta representar el significado de frases en lenguaje natural de forma que:
  1. Se posibilite la derivación de conclusiones.
  2. La representación sea independiente del lenguaje utilizado en las declaraciones originales.
- Una representación de dependencia conceptual no se construye a partir de primitivas que correspondan a palabras concretas de una declaración dada. Por el contrario, se utilizan primitivas conceptuales que pueden combinarse para formar significados de palabras en cualquier lenguaje.
- La dependencia conceptual proporciona simultáneamente una estructura y un conjunto específico de primitivas a partir de las cuales pueden construirse representaciones de elementos concretos de información. Así, el suceso representado por la frase: “Yo le di un libro al hombre”, se representaría en un modelo de dependencia conceptual del siguiente modo:

### 3.4.2 Dependencia conceptual y guiones (Cont.)



en donde:

Las flechas indican una dirección de dependencia

La flecha doble indica un enlace bidireccional entre la acción y el actor “p” indica tiempo pasado

ATRANS es una de las primitivas de acción permitidas por el modelo e indica una transferencia de posesión

“o” indica una relación causal con un objeto

“R” indica una relación causal relativa a “libro”, que va desde *Yo* hasta *hombre*

### 3.4.2 Dependencia conceptual y guiones (Cont.)

- El conjunto de primitivas que utiliza el modelo es el siguiente:
- ATRANS: Que indica transferencia de una relación abstracta (e.g., dar)
- FTRANS: Que indica transferencia de la localización física de un objeto (e.g. ir)
- IMPELER: Que indica aplicación de fuerza física a un objeto (e.g., empujar)
- MOVER: Que indica movimiento de parte de un cuerpo por su propietario (e.g., patear)
- ASIR: Que indica el hecho de tomar un actor un objeto (e.g., lanzar)
- INGER: Que indica ingerir un animal un objeto (e.g., comer)
- EXPEL: Que indica expulsar algo del cuerpo de un animal (e.g., llorar)
- MTRANS: Que indica transferencia de información mental (e.g., contar)
- MCONST: Que indica construir información nueva a partir de otra ya existente (e.g., decidir)
- HABLAR: Que indica producción de sonidos (e.g., decir)
- ATENT: Que indica centrar un órgano sensorial sobre un estímulo (e.g., oír)

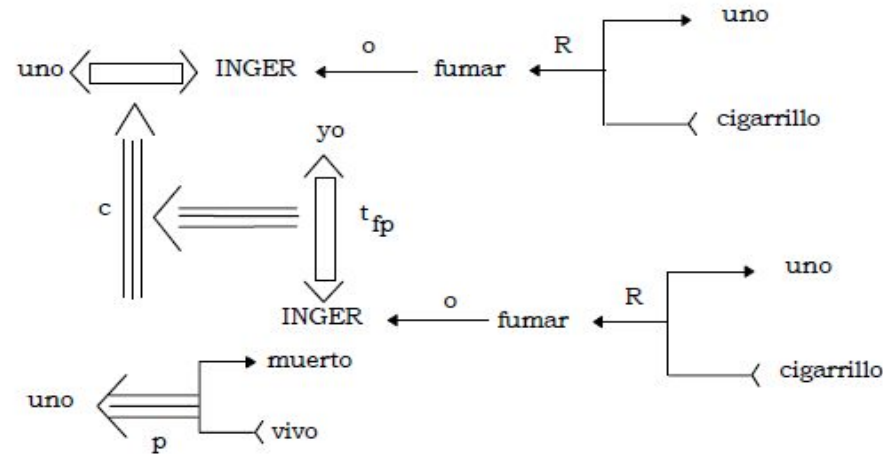
### 3.4.2 Dependencia conceptual y guiones (Cont.)

- Un segundo bloque de construcción es el conjunto de las dependencias permitidas entre las conceptualizaciones descritas en una frase. El modelo de dependencia conceptual incluye cuatro categorías conceptuales primitivas, a partir de cuyas dependencias pueden construirse las estructuras:
- ACC Acciones
- OB Objetos (que generan imágenes)
- MA Modificadores de acciones (ayudas a las acciones)
- MO Modificadores de objetos (ayudas a las imágenes)
- En este modelo, las dependencias entre conceptualizaciones corresponden a las relaciones semánticas entre conceptos subyacentes. Tales dependencias pueden resumirse en una lista de 15 reglas

### 3.4.2 Dependencia conceptual y guiones (Cont.)

- Las conceptualizaciones representan sucesos que pueden modificarse de varias formas para proporcionar información indicada normalmente en el lenguaje por medio de: tiempo, modo y forma verbal. El conjunto de tiempos conceptuales utilizado habitualmente es el siguiente:
  1. p pasado
  2. f futuro
  3. t transición
  4. ti inicio de transición
  5. tf final de transición
  6. k continuar
  7. ? interrogación
  8. / negativo
  9. nada presente
  10. delta atemporal
  11. c condicional
- El uso de tiempos conceptuales es importante en el modelo de dependencia conceptual. A continuación se representa la frase “*puesto que el fumar puede matarte, lo dejé*”, en donde se ilustra cómo este modelo emplea los tiempos conceptuales:

### 3.4.2 Dependencia conceptual y guiones



El modelo de dependencia conceptual facilita el razonamiento con el conocimiento representado ya que:

1. Al descomponer el conocimiento en primitivas se necesitan menos reglas de inferencia.
2. Muchas inferencias están autocontenidas en la representación.
3. La estructura inicial construida con la información disponible, facilita el enfoque de atención del programa que debe entender las frases, ya que contiene huecos que deberán rellenarse.

### 3.4.2 Dependencia conceptual y guiones (Cont.)

- Los guiones son especializaciones del concepto general de frame, que conforman estructuras capaces de representar prototipos de secuencias de sucesos. En los guiones el tiempo es siempre una variable implícita. Los elementos estructurales más habituales de los guiones son:
  1. Las condiciones de entrada o condiciones iniciales de aplicabilidad del guión.
  2. Los resultados o hechos que serán verdaderos una vez el guión sea ejecutado.
  3. Las herramientas u objetos relevantes en el desarrollo del guión.
  4. Los papeles o roles que representan los actores que actúan en el guión.
  5. Las escenas o secuencias típicas de eventos del guión.
  
- En los guiones no existen reglas absolutas que definan contenidos genéricos. Así, un mismo evento puede ser contemplado según:
  1. distintos factores de tiempo
  2. distintos lugares de ocurrencia
  3. distintos puntos de vista

### 3.4.2 Dependencia conceptual y guiones (Cont.)

- La Figura muestra el guion típico “viaje al teatro”:

Guion: Viale al teatro	
Herramientas: Automóvil Llaves Puertas Aparcamiento	Escena Primera: Arranque Propietario busca llaves Propietario abre puerta Propietario arranca coche Propietario sale
Papeles (roles): Propietario Conserje	Escena segunda: Conducción Incorporación al tráfico Dirección al teatro
Entradas: Propietario y automóvil	Escena tercera: Encuentro con el conserje Propietario para automóvil Propietario sale del coche Propietario da llaves al conserje Propietario entra al teatro
Resultado: Propietario en el teatro Y automóvil en el aparcamiento	Escena cuarta: Fin del guion Empleado entra al coche Empleado arranca Empleado encuentra aparcamiento libre Empleado aparca el coche Empleado sale del coche



### 3.4.2 Dependencia conceptual y guiones (Cont.)

- En los guiones el proceso de razonamiento está basado en que, si un determinado guión es apropiado para describir una situación dada, debe poder ser también utilizado para predecir acontecimientos no mencionados explícitamente. Así, para razonar sobre la base de un guión, éste debe ser previamente seleccionado mediante lo que se denomina proceso de *activación*.
- Existen dos tipos fundamentales de guiones:
  1. Los guiones instantáneos (i.e., aquellos a los que nos podemos referir siempre pero que no son el foco de atención principal de nuestro problema)
  2. Los guiones no instantáneos (que constituyen el foco de atención principal de nuestro problema).
- La activación de un guión instantáneo se realiza definiendo punteros en lugares estratégicos de forma que, cada vez que tal guión sea requerido, sepamos en dónde buscarlo.
- Por el contrario, en el caso de los guiones no instantáneos procede la activación total del mismo tras el proceso de emparejamiento de la situación inicial con el conjunto de guiones que describen secuencias en nuestro dominio. Así, la secuencia de eventos en un guión puede entenderse como una *cadena causal* gigante.

### 3.5 Modelos de conocimiento incierto e incompleto

La información que necesitamos para desarrollar un Sistema basado en Conocimiento tiene muchas veces las siguientes características:

- NO ES DEL TODO CONFIABLE
- IMPRECISO
- INCOMPLETO
- CONTRADICTORIO

#### Causas de inexactitud

- Generalmente no es del todo confiable (falta de evidencias, excepciones)
- Suele ser incompleta a la hora de tomar decisiones (faltan datos provenientes de mediciones, análisis)
- Diferentes fuentes pueden ser conflictivas, redundantes, subsumidas
- El lenguaje usado para transmitirla es inherentemente impreciso, vago

### 3.5.1 Factores de certidumbre

- El conocimiento se expresa mediante predicados precisos pero **no podemos establecer el valor de verdad** de la expresión
- **Ejemplos:**
  1. *Es posible que* mañana llueva  
Mañana llueve
  2. *Creo que* el auto era rojo  
El auto es rojo
- Cuando no podemos establecer la verdad o falsedad de la información Debemos evaluar la :  
PROBABILIDAD POSIBILIDAD NECESIDAD/PLAUSIBILIDAD GRADO DE  
CERTeza... De que la información sea verdadera.
- El conocimiento cuenta con **predicados o cuantificadores vagos** (no precisos)

#### **Ejemplos:**

- Pedro tiene entre 20 y 25 años.
- Juan es joven
- Mucha gente juega al fútbol
- El espectáculo es para gente grande.

### 3.5.1 Factores de certidumbre (Cont.)

- Se debe tomar decisiones a partir de información incompleta o parcial.
- Esto se suele manejar a través de supuestos o valores por defecto.

Ejemplo:

1. Si el paciente tiene S1, S2 y S3 entonces tiene una infección a Bacteria S3 ???

#### **RAZONAMIENTO APROXIMADO (RA)**

Trata como REPRESENTAR, COMBINAR y REALIZAR INFERENCIAS con conocimiento impreciso y/o incierto

#### **Distintos modelos**

- MODELOS PROBABILISTICOS
- MODELO EVIDENCIAL
- MODELO POSIBILISTICO

Todos tratan la incertidumbre en un sistema de producción  
Sólo el modelo posibilístico puede tratar la imprecisión.



## 3.5.2 Lógica difusa

- La lógica tradicional funciona con valores establecidos y nos dice si algo es verdadero o falso. Sin embargo, los seres humanos no trabajan con este tipo de lógica. Un sistema difuso trabaja con valores no exactos.
- La lógica difusa descende de los conjuntos difusos, que forman parte de la teoría de conjuntos. En los conjuntos tradicionales, podemos decir únicamente si objeto pertenece o no al conjunto. En los conjuntos difusos, podemos agregar que tanto pertenece un objeto al conjunto, es decir, su grado de membresía.
- En la lógica tradicional se tiene incertidumbre pero la lógica difusa nos permite trabajar con esto. En la lógica difusa, nos manejamos con grados de pertenencia a un conjunto y permite que sea parcial.

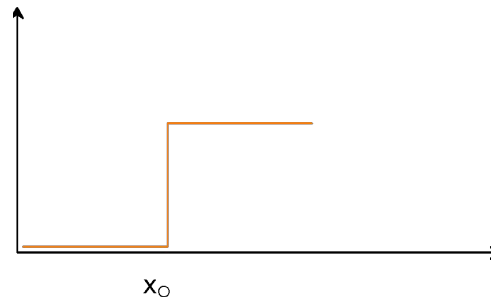
### 3.5.2 Lógica difusa (Cont.)

- En la lógica difusa, la membresía la damos por medio de un valor entre 0 y 1. Si el objeto no pertenece al conjunto, su valor de membresía es 0. Si el objeto pertenece en forma total al conjunto, su valor de membresía es 1. Si pertenece parcialmente se da un valor entre ellos.
- Para crear una aplicación en lógica difusa, necesitamos tres pasos:  
Fuzzificación: Se toma el dato del sistema y se convierte en un dato difuso. El dato del sistema puede ser un número real obtenido del ambiente por medio de un sensor, o calculado por alguna función o una entrada dada por el usuario. El transformar el dato a un dato difuso se lleva a cabo al encontrar el grado de membresía de ese valor contra una serie de conjuntos difusos.
- Reglas difusas: Estas reglas las construimos nosotros según las necesidades del sistema. Después de ser evaluadas, sabremos qué grado de membresía tiene cada una con relación a la salida.
- Defuzzificación: Regresar el valor resultante a un valor concreto, perteneciente a los reales.

### 3.5.2 Lógica difusa (Cont.)

Función de membresía

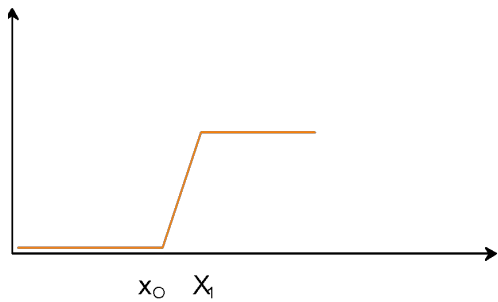
- Esta función recibe el dato concreto y regresa un valor que indica qué tanto pertenece ese dato a la función. El valor devuelto se encuentra en el rango de 0 a 1.
- Función de membresía booleana- Esta función regresa solamente dos valores 0 o 1. Supongamos que el dato que tenemos se encuentra en la variable  $X$ . Dependiendo del valor de  $X$ , el valor regresado es 0 o 1. Para poder indicar el punto a partir del cual cambia el valor regresado por la función, usamos la variable  $X_0$ .



## 3.5.2 Lógica difusa

### Función de membresía de grado

- Esta función tiene tres secciones. Una sección nos regresa el valor de 0; la siguiente sección nos regresa un valor comprendido entre 0 y 1, y la tercera sección nos regresa el valor de 1. Es decir que nuestro dato puede no tener membresía, algo de membresía o membresía completa. Las secciones la definimos por medio de dos variables  $x_0$  y  $x_1$ . Con esto, definimos la primera sección desde el inicio de los posibles valores de  $x$  hasta  $x_0$ ; la segunda de  $x_0$  a  $x_1$ , y la tercera desde  $x_1$  hasta el máximo valor posible de  $x$ .



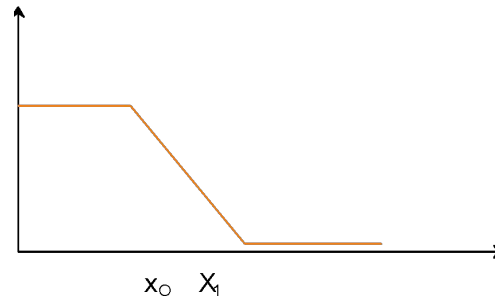
Dependiendo de la sección donde se encuentra, el valor de membresía es calculado y luego devuelto al finalizar la función. La segunda sección resulta interesante, pues aquí calculamos el valor de membresía usando una fórmula que podemos encontrar por medio de las ecuaciones de las rectas.



## 3.5.2 Lógica difusa

Función de membresía de grado inverso

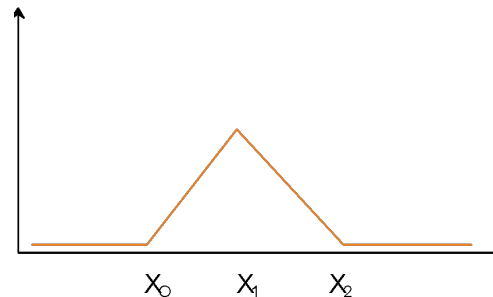
- La primera sección queda definida entre el inicio de los posibles valores de  $X$  hasta  $X_0$  y regresa una membresía de 1 en esta sección. La segunda sección se encuentra entre  $X_0$  y  $X_1$  y regresa un valor entre 0 y 1. La tercera parte va desde  $X_1$  hasta el valor máximo posible de  $X$ .



## 3.5.2 Lógica difusa

### Función de membresía de triángulo

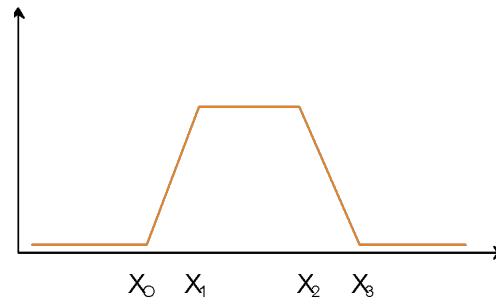
- En la función de membresía de triángulo necesitamos tres valores que son  $X_0$ ,  $X_1$ ,  $X_2$ . Estos tres valores nos definen cuatro secciones. La primera, desde el valor de inicio de  $X$  hasta  $X_0$ ; la función regresa 0. La segunda, desde  $X_0$  hasta  $X_1$ ; la función regresa un valor desde 0 hasta 1. En la tercera sección, el valor regresado va desde 1 hasta 0, y queda definida entre  $X_1$  y  $X_2$ . La última parte va desde  $X_2$  hasta el valor de  $X$  y regresa a 0.



## 3.5.2 Lógica difusa

### Función de membresía trapezoide

- En esta función se tienen cinco partes o secciones. Utilizaremos cuatro variables para delimitar las secciones. Estas variables son  $X_0$ ,  $X_1$ ,  $X_2$ ,  $X_3$ . La función regresa a 0 en el primer segmento que va desde el valor más pequeño de  $X$  hasta  $X_0$ . En la segunda sección, que se define desde  $X_0$  hasta  $X_1$ , el valor regresado por  $X$  va desde 0 hasta 1. Continuamos con la tercera sección, la cual regresa 1 y queda definido entre  $X_1$  y  $X_2$ . La cuarta parte, el valor va desde 0 hasta 1 y se encuentra entre  $X_2$  y  $X_3$ . Por último, la quinta sección regresa el valor de 0 y va desde  $X_3$  hasta el valor máximo posible para  $X$ .





### 3.5.2 Lógica difusa

- Podemos reunir varias funciones de membresía dentro de un conjunto al que se le llamara difuso. Todas las funciones contenidas dentro de este conjunto difuso deben utilizar la misma variable  $X$  para calcular la membresía de  $X$  en cada una de las funciones.
- Es recomendable que las funciones de los conjuntos difusos estén conceptualmente relacionadas entre sí.