



Universidad  
Francisco de Vitoria  
**UFV** Madrid

## *Ingeniería del Conocimiento*

---

### ***Tema 8: Otros métodos de Representación del Conocimiento***



- Ubicación
  - Unidad 3: **EL CONOCIMIENTO Y SU REPRESENTACION**
  - *Tema 8: Otros métodos de Representación del Conocimiento*
  
- Objetivos generales
  - Enumerar y definir *otros métodos de representación* del conocimiento adicionales a las lógicas formales
  - Describir las *Reglas* y su relación con la lógica de predicados.
  - Describir las *Redes Semánticas*
  - Describir los *Frames*
  - Comprender los *mecanismos de representación* con cada uno de los métodos
  - Comprender los *mecanismos de inferencia* con cada uno de los métodos



1. Introducción
2. Lógica de Predicados de Primer Orden
3. Reglas
  1. Reglas de Producción
  2. Proceso de Inferencia
4. Redes Semánticas
5. Frames
6. Resumen



1. **Introducción**
2. Lógica de Predicados de Primer Orden
3. Reglas
  1. Reglas de Producción
  2. Proceso de Inferencia
4. Redes Semánticas
5. Frames
6. Resumen

# 1. Introducción



- La lógica como lenguaje de representación tiene una dificultad práctica: Demasiado formal para los usuarios
  - Se necesitan mecanismos mas intuitivos y fáciles de usar y que permitan también su implementación computacional
  - Cada mecanismo lleva asociado un modelo de representación y un modelo de razonamiento (inferencia)
  
- Mecanismos de representación e inferencia de conocimiento
  - Lógicas
  - Reglas
  - Redes Semánticas
  - Marcos (Frames)



1. Introducción
2. Lógica de Predicados de Primer Orden
3. Reglas
  1. Reglas de Producción
  2. Proceso de Inferencia
4. Redes Semánticas
5. Frames
6. Resumen

## 2. Lógica de Predicados de Primer Orden



- La Lógica Proposicional es débil: no puede representar todo tipo de conocimiento.
  - No se puede representar *Luis es el hermano de Pedro*
  
- Es necesario usar Lógica de Predicados de Primer Orden para representaciones más complejas
  - Amplía el vocabulario de la Lógica Proposicional
  - Añade parámetros a los enunciados simples (átomos), que pasan a representar propiedades o relaciones (predicados)
    - Las relaciones entre objetos se representan como tuplas
      - *Hermano(x,y)*
    - Las propiedades son relaciones unitarias
      - *Viejo(x)*

# 2. Lógica de Predicados de Primer Orden



## Sintaxis

- Alfabeto de la Lógica de Primer Orden
  - Átomos:  $T$  (verdadero) y  $\perp$  (falso).
  - Términos
    - Constantes: *elementos identificables del dominio*  $\mathbf{a}, \mathbf{b} \dots$
    - Variables: *elementos que podemos cuantificar o substituir por valores*  $\mathbf{x}, \mathbf{y} \dots$
    - Funciones (con su aridad): *expresiones que denotan el valor resultante de aplicar la función a variables y constantes*  $\mathbf{f}(\mathbf{x}, \mathbf{y}, \dots)$
  - Símbolos de predicado (con su aridad): *Representan propiedades o relaciones definidas sobre términos*  $\mathbf{P}(\mathbf{x}, \mathbf{y}, \dots)$
  - Conectivas:  $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
  - Cuantificadores:  $\forall$  (cuantificador universal),  $\exists$  (cuantificador existencial)
  - Símbolos extralingüísticos:  $( ) [ ] \{ \} ,$



## 2. Lógica de Predicados de Primer Orden



### ■ Cuantificación

- Los cuantificadores cierran las fórmulas abiertas construidas con variables → cuantifican el valor de las variables

- Universal ( $\forall$ ): Enunciados que cumplen todos los elementos de un conjunto. Válido para todas las interpretaciones de  $x$

- Todo rey es persona

$\forall x \text{ Rey}(x) \rightarrow \text{Persona}(x)$  donde  $x$  es una variable

- Existencial ( $\exists$ ): Existe un conjunto de elementos que cumplen cierta propiedad. Es verdadero para "al menos un"  $x$

- El rey Juan tiene una corona sobre su cabeza

$\exists x \text{ Corona}(x) \wedge \text{SobreCabeza}(x, \text{Juan})$

- Ejemplo

- $a = \text{Quevedo}$                        $M(x)$   $x$  es mamífero

- $L(x, a)$   $x$  Lee  $a$                        $D(x)$   $x$  disfruta

$\exists x [M(x) \wedge L(x, a)]$                       Algunos mamíferos leen a Quevedo

$\forall x [L(x, a) \rightarrow D(x)]$                       Todos los lectores de Quevedo disfrutan

$\therefore \exists x [M(x) \wedge D(x)]$                        $\therefore$  Algunos mamíferos disfrutan

## 2. Lógica de Predicados de Primer Orden



- Por convención
  - Sentencias atómicas: letras mayúsculas a partir de la  $P$
  - Fórmulas: letras mayúsculas a partir de la  $A$
  - Para los términos,
    - Constantes: letras minúsculas a partir de la  $a$
    - Variables: letras minúsculas a partir de la  $x$
    - Funciones: letras minúsculas a partir de la  $f$

## 2. Lógica de Predicados de Primer Orden



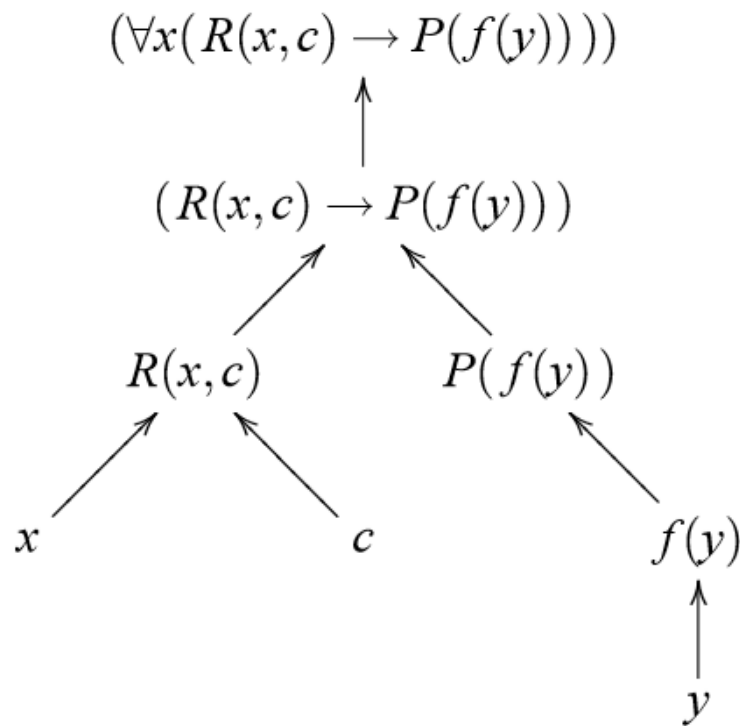
### ■ Gramática de la Lógica de Primer Orden

- Término: expresión obtenida por aplicación de las reglas
  - Cada constante  $c$  es un término
  - Cada variable  $x$  es un término
  - Si  $t_1, \dots, t_n$  son términos y  $f$  es una función  $n$ -aria, entonces  $f(t_1, \dots, t_n)$  es un término
- Fórmula atómica: predicado con el número de términos adecuado
  - Si  $t_1, \dots, t_n$  son términos y  $P$  es un predicado atómico  $n$ -ario, entonces  $P(t_1, \dots, t_n)$  es una Fórmula atómica
- Formulas bien formadas *fbf*:
  - Toda fórmula atómica es una *fbf*
  - Si  $A$  y  $B$  son *fbf*, la unión por conectivas también lo son ( $\wedge, \vee, \rightarrow, \leftrightarrow, \neg$ )
  - Si  $A$  es una *fbf* y  $x$  una variable,  $\forall xA(x)$  y  $\exists xA(x)$  también lo son

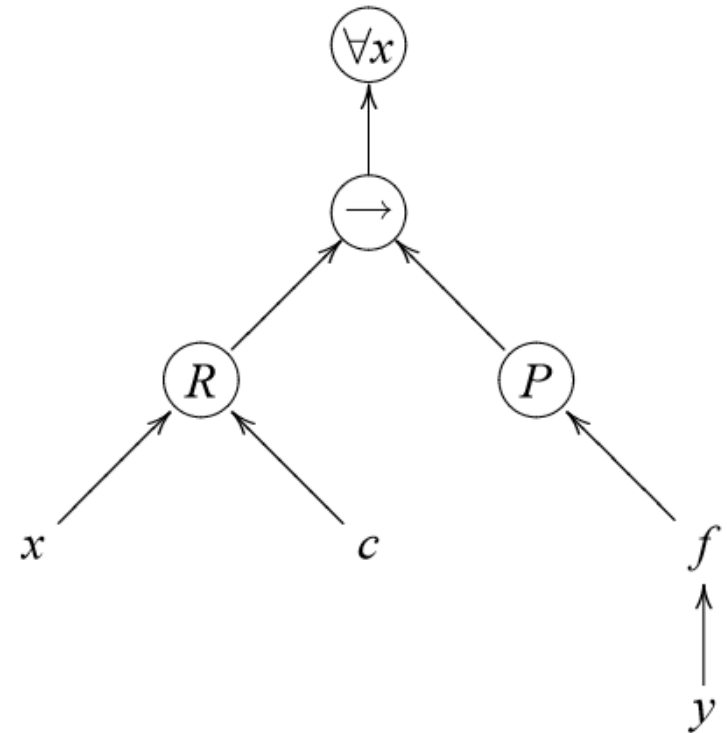
## 2. Lógica de Predicados de Primer Orden



- Árbol sintáctico para una *fbf* en Lógica de Primer Orden



(a) árbol sintáctico



(b) árbol, más esquemático

## 2. Lógica de Predicados de Primer Orden



### ■ Inferencia en la Lógica de Primer Orden

- Es necesario redefinir los mecanismos de razonamiento para que respondan a preguntas basadas en LPPO (versiones elevadas)
  - *Modus Ponens*
  - *Unificación*
  - *Resolución*
- La versión *elevada* del Modus Ponens (Lógica Proposicional) para la LPPO se llama *Modus Ponens Generalizado*
  - Una interpretación asigna un valor de verdad a cada formula
  - Una fórmula Q es *consecuencia lógica* de un conjunto de formulas  $\{P_1, P_2, \dots, P_n\}$

$$\{P_1, P_2, \dots, P_n\} \models Q$$

- Si para  $\forall n, \forall V_I(P_n) = \mathbf{V}$
- Entonces  $V_I(Q) = \mathbf{V}$



1. Introducción
2. Lógica de Predicados de Primer Orden
3. Reglas
  1. Reglas de Producción
  2. Proceso de Inferencia
4. Redes Semánticas
5. Frames
6. Resumen

# 3. Reglas



- En Ciencias de la Computación se usan reglas como mecanismo de programación:
  - Para compilación y traducción,
    - Compiladores de lenguajes (LEX, YACC)
    - XSLT (eXtensible Stylesheet Language Transformations)
    - Automatización de tareas: Make, ANT, ...
  - Como reglas de negocio en aplicaciones
    - En herramientas comerciales: SAP, IBM, Oracle, Microsoft, ...
    - En paradigmas de desarrollo de aplicaciones (SOA, SaaS)
  
- Pero además,
  - Casi todos los expertos son capaces de expresar su conocimiento en forma de REGLAS que pueden resolver problemas a partir de HECHOS
  - La representación en forma de reglas es intuitiva

# 3. Reglas



- Ejemplo: procedimiento para determinar si puedo o no sentarme en el autobús:
  - IF (en la parada AND bus llega) THEN **action**(coger el bus)
  - IF (en el bus AND no he pagado AND tengo abono) THEN **action**(enseñar el abono) AND **add**(pagado)
  - IF (en el bus AND pagado AND hay asiento) THEN sentarme
- Este “razonamiento” se estudia formalmente mediante su formulación con Lógica de Predicados de Primer Orden



## 3.1 Reglas de producción



- Podemos representar un problema con LPPO como si fuera un proceso de razonamiento descomponiendo el problema en problemas más sencillos que se resuelven paso a paso.
    - Restricción: Solo se pueden usar un conjunto restringido de expresiones de la lógica
    - Justificación: la representación se ejecuta como una demostración y el coste computacional está ligado a la expresividad empleada en su descripción
    - Expresiones que se pueden usar:
      - Únicamente fórmulas cuantificadas universalmente que se puedan transformar en cláusulas disyuntivas en las que solo hay un átomo afirmado
      - Llamadas cláusulas de Horn o Reglas de Producción
- $\forall x \forall y [Persona(x) \wedge Edad(x,y) \wedge Mayor(y,18)] \rightarrow Mayor\_de\_edad(x)$



## 3.1 Reglas de producción

- **Regla de Producción** (en IA): *estructura IF-THEN que relaciona hechos o información de la parte IF con alguna acción en la parte THEN.*
  - Proporcionan una descripción de como resolver un problema
  - Examinan un conjunto de datos y producen (inferen) nueva información
- **Estructura**
  - Una *regla* es una **cláusula de Horn** expresada como
$$P_1 \wedge P_2 \wedge \dots \wedge P_n \rightarrow Q$$
donde  $P_1, \dots, P_n, Q$  son fórmulas atómicas
    - Cabeza de la regla:  $Q$
    - Cuerpo de la regla:  $P_1 \wedge P_2 \wedge \dots \wedge P_n$
  - Un *hecho* es una fórmula atómica
  - Una *base de conocimiento* es un conjunto de hechos y reglas



## 3.1 Reglas de producción

- Una regla consta de dos partes
  - Parte IF: antecedente, premisa o condición
  - Parte THEN: consecuente, conclusión o acción
- Una regla puede tener múltiples antecedentes combinados por los operadores AND (conjunción), OR (disyunción) o una combinación de ambos, pero solo un consecuente

```
IF    <antecedente 1>
AND   <antecedente 2>
:
OR    <antecedente n>
:
THEN  <consecuente>
```

*“Toda persona mayor de 18 años es mayor de edad”*

```
SI    ?x ES Persona
Y     Edad ?x ES ?Y
Y     ?Y > 18
ENTONCES ?x ES MayorEdad
```

## 3.1 Reglas de producción



- El antecedente de una regla incorpora dos partes:
  - Un objeto (objeto lingüístico)
  - Su valor
  - Ambos están unidos por un operador
- El operador identifica el objeto y le asigna el valor
  - Operadores como *es*, *no\_es*, *son*, etc. se usan para asignar valores simbólicos a objetos lingüísticos.
  - También se usan operadores matemáticos para definir un objeto como numérico y asignarle un valor numérico

```
IF 'edad del cliente' < 18  
AND 'retirada de efectivo' > 1000  
THEN se requiere 'firma del padre'
```

# 3.1 Reglas de producción



- Ejemplos de Reglas

REGLAS	HECHOS
SI        la luz del semáforo es verde Y        no hay peatones cruzando ENTONCES continúa la marcha	La luz del semáforo es verde
SI        x es número natural Y        x es par ENTONCES x es divisible por 2	
SI        el reactivo toma color azul Y        la morfología del organismo es alargada Y        el paciente es un posible receptor ENTONCES hay evidencia (0.7) de infección debida a pseudomonas.	El reactivo toma color azul

## 3.1 Reglas de producción



- Mecanismo de Razonamiento: Resolución por Inferencia usando el *Modus Ponens Generalizado*
- Dada una base de conocimiento **BC** y una formula atómica **P**, decimos que **BC**  $\models$  **P** (P se infiere de BC)
  - si existe una secuencia de formulas  $F_1, \dots, F_n$  tales que  $F_n = \mathbf{P}$
  - y cada  $F_i$  esta en **BC** o se obtiene de formulas anteriores mediante aplicación del *Modus Ponens Generalizado*
  - La obtención se puede realizar mediante
    - Razonamiento hacia adelante (Forward Chaining  $\rightarrow$  de las causas a los efectos) partiendo de **BC**
    - Razonamiento hacia atrás (Backward Chaining  $\rightarrow$  de los efectos a las causas) partiendo de **P**

# 3.1 Reglas de producción



## Base de Conocimiento

**R1:**  $\text{bueno}(x) \wedge \text{rico}(y) \wedge \text{quiere}(x,y) \rightarrow \text{hereda-de}(x,y)$

**R2:**  $\text{amigo}(x,y) \rightarrow \text{quiere}(x,y)$

**R3:**  $\text{antecesor}(y,x) \rightarrow \text{quiere}(x,y)$

**R4:**  $\text{progenitor}(x,y) \rightarrow \text{antecesor}(x,y)$

**R5:**  $\text{progenitor}(x,z) \wedge \text{progenitor}(z,y) \rightarrow \text{antecesor}(x,y)$

**H1:**  $\text{progenitor}(\text{padre}(x),x)$

**H2:**  $\text{rico}(\text{Pedro})$

**H3:**  $\text{rico}(\text{padre}(\text{padre}(\text{Juan})))$

**H4:**  $\text{amigo}(\text{Juan},\text{Pedro})$

**H5:**  $\text{bueno}(\text{Juan})$

- a partir de
  - $\text{bueno}(\text{Juan})$
  - $\text{rico}(\text{Pedro})$
  - $\text{amigo}(\text{Juan},\text{Pedro})$
  - **R2:**  $\text{amigo}(x,y) \rightarrow \text{quiere}(x,y)$
  - **R1:**  $\text{bueno}(x) \wedge \text{rico}(y) \wedge \text{quiere}(x,y) \rightarrow \text{hereda-de}(x,y)$
- se deduce
  - $\text{hereda-de}(\text{Juan},\text{Pedro})$
- Por sustitución de  $x \rightarrow \text{Juan}$ ,  $y \rightarrow \text{Pedro}$

# 3.1 Reglas de producción



- Ventajas
  - Las reglas son fáciles de entender
  - Los mecanismos de inferencia son relativamente sencillos
  - Se puede introducir incertidumbre fácilmente
- Desventajas
  - Dificultad de establecer relaciones
  - Representar conocimiento complejo requiere muchas reglas
  - Tener muchas reglas puede crear problemas de usabilidad, mantenimiento y eficiencia



## 3.2 Procesos de Inferencia



- Procesos de inferencia:
  - Razonamiento hacia delante (forward chaining)
    - Dirigido por datos
    - Teniendo todos los hechos de partida podemos ir analizándolos para derivar conclusiones y así sucesivamente...
  - Razonamiento hacia atrás (backward chaining)
    - Dirigido por objetivos
    - Teniendo en conjunto limitado de posibles conclusiones, intentamos demostrar una conclusión probando las premisas de esa regla, y así sucesivamente



### Forward chaining

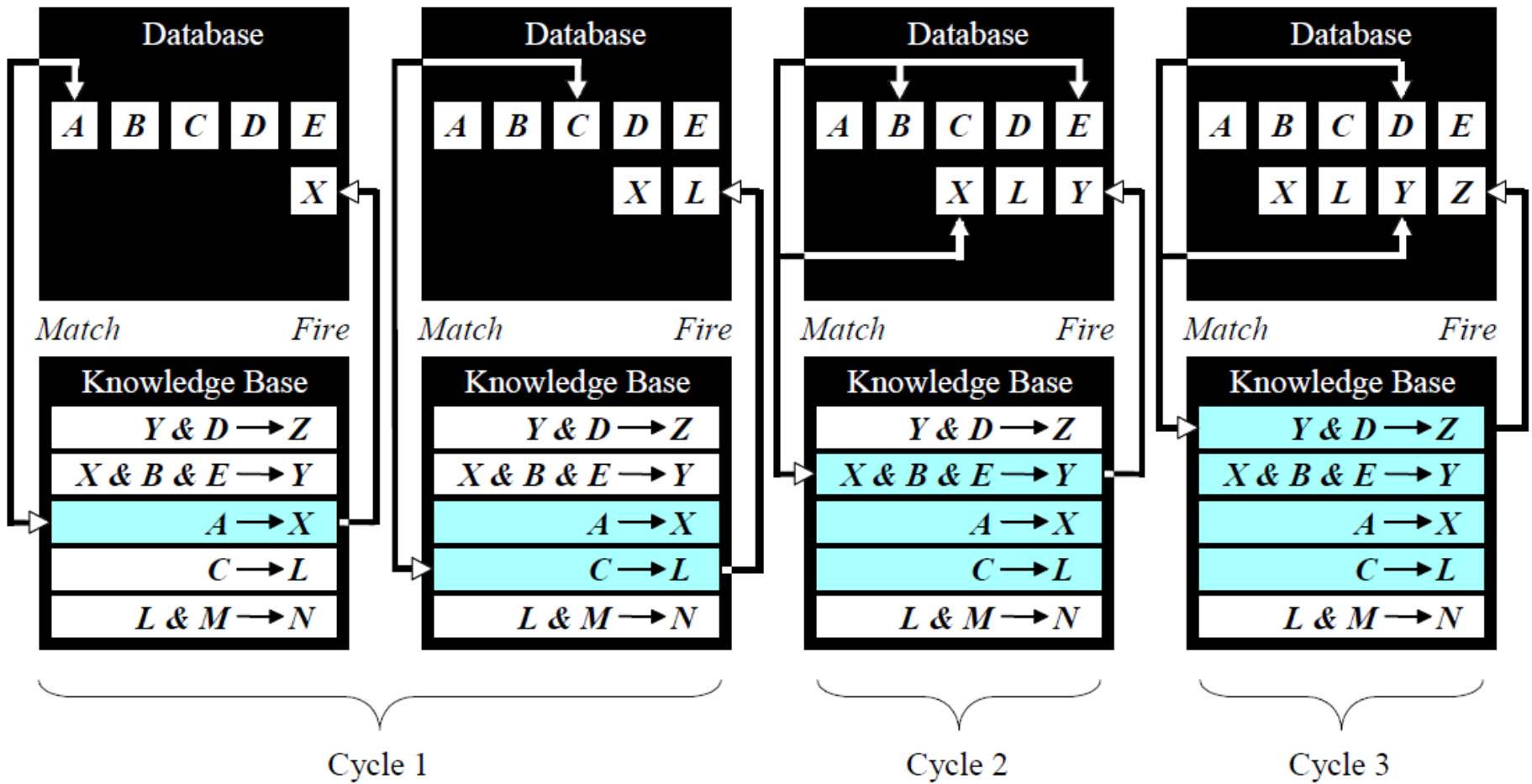
- Razonamiento basado en datos
- BC con solo cláusulas de Horn positivas → evitamos utilizar resolución
- Partiendo de **BC** (sentencias atómicas de la base de conocimiento)...
  - Se aplica el *Modus Ponens Generalizado* hacia delante, añadiendo las sentencias atómicas nuevas hasta que no se puedan realizar más inferencias.
- Ventaja:
  - Con esta técnica se infiere todo lo que puede ser inferido
- Inconveniente
  - Muchas de las reglas que se disparan pueden no tener nada que ver con el objetivo que se busca

## 3.2 Procesos de Inferencia



- Pasos
  - Mantener una lista de todos los hechos conocidos.
  - Se buscan todas aquellas reglas cuyos antecedentes podamos validar con cualquier posible combinación de los hechos que tenemos en ese momento.
  - Tras la selección de la regla más adecuada se añade a la base de hechos el consecuente de la regla como un hecho nuevo.
    - Los hechos deducidos permiten obtener nuevos hechos
    - Cada vez solo se dispara la regla más “activa”
    - Cada regla solo se puede disparar una sola vez
  - El ciclo se detiene cuando
    - No hay más reglas que disparar
    - El objetivo es deducido

## 3.2 Procesos de Inferencia





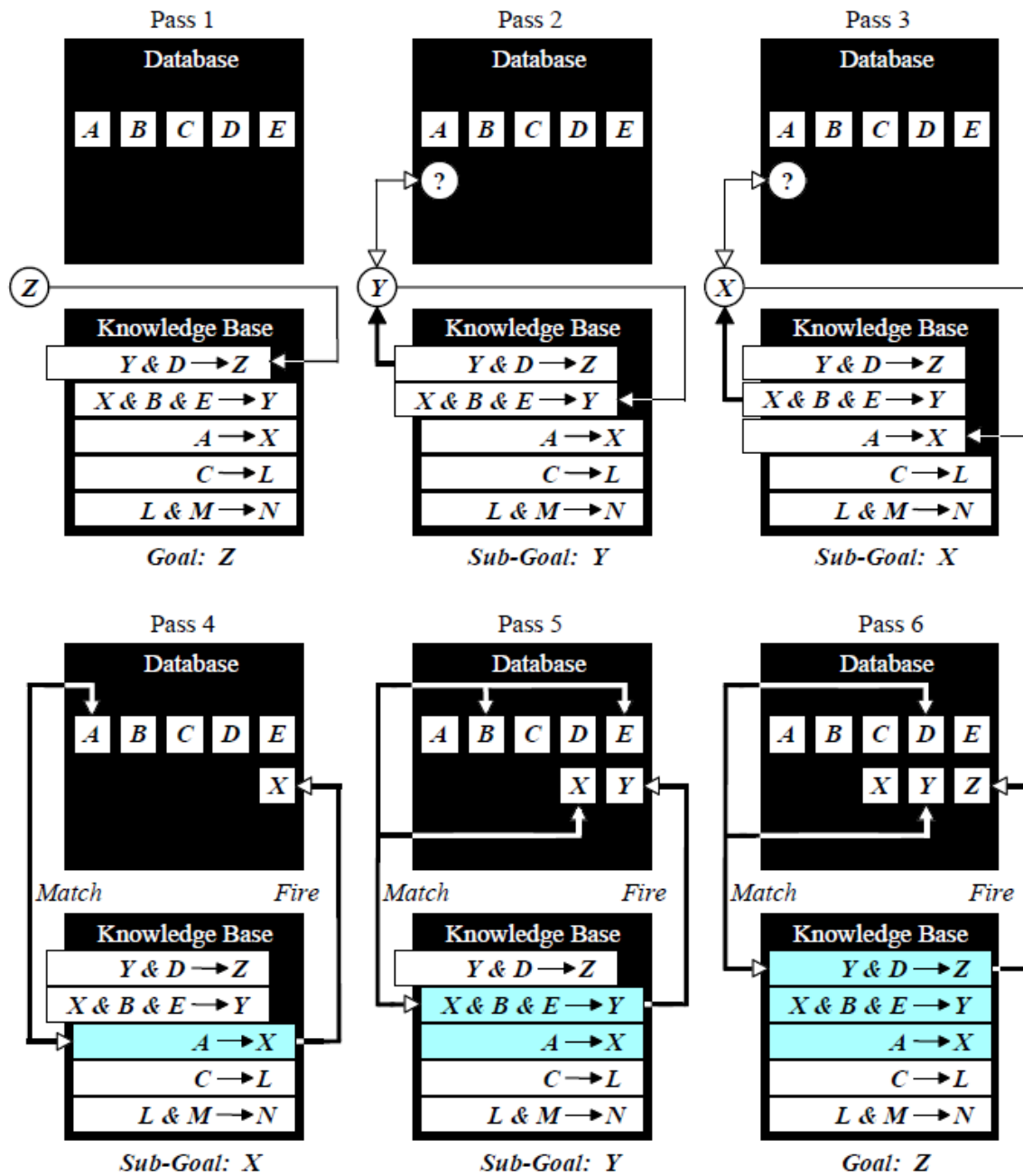
### Backward chaining

- Razonamiento basado en objetivos
- Estos algoritmos trabajan hacia atrás desde el objetivo (Hipotético), encadenando a través de las reglas hasta encontrar los hechos conocidos que soportan la demostración
- Partiendo de **P**...
  - Se aplica el *Modus Ponens Generalizado* hacia atrás
- Ventaja:
  - Todo el trabajo que se realiza se encamina a la resolución del problema y no se hace trabajo extra
- Inconveniente:
  - Plantear un conjunto de reglas para backtracing es menos intuitivo

## 3.2 Procesos de Inferencia



- Pasos
  - Mantener una lista de objetivos pendientes de demostrar que se inicializa con el objetivo del problema.
  - Se mira si entre los hechos está el objetivo actual
    - Si es así lo eliminamos y pasamos al objetivo siguiente.
    - Si no es así, se seleccionan todas aquellas reglas de la BC que tengan el objetivo como conclusión (acción de la parte THEN)
  - Si esa regla existe y su parte izquierda IF (condición) se satisface con los hechos de la Base de Hechos, entonces la regla se dispara y se da por alcanzado el objetivo
  - Si no se satisface la condición, los predicados del antecedente de la regla pasan a añadirse a la lista de objetivos y se buscan reglas que lo prueben (*backtracking*)
  - El proceso continúa hasta que
    - No se encuentran reglas que prueben los objetivos
    - Todos los objetivos que han aparecido han sido cubiertos mediante hechos del problema.



## 3.2 Procesos de Inferencia



- Ejemplo Backward Chaining: MYCIN
  - Desarrollado en Standford (LISP)
  - Sistema basado en reglas con factor de confianza
  - Razonamiento backward chaining
  - Interfaz con el usuario en inglés

- Ejemplo de regla:

IF stain of organism is gram negative

AND

morphology is rod

AND

is anaerobic

THEN

suggestive that class is *enterobacteriaceae* (CF 0.8)



## 3.2 Procesos de Inferencia



*¿Nombre, edad, sexo, raza?*

» Roberto, 55, Masculino, Caucásico

*¿Se han obtenido cultivos positivos en el lugar donde ROBERTO presenta la infección?*

» Sí

*¿De qué infección se trata?*

» Bacteriemia primaria

*¿En qué fecha aparecieron los primeros síntomas de bacteriemia primaria?*

» 01/01/2000

*Llamaré Cultivo-1 al cultivo más reciente asociado con bacteriemia primaria.*

*¿De qué lugar se tomó el cultivo?*

» Sangre

*¿En qué fecha?*

» 10/01/2000

*Llamaré Organismo-1 al microorganismo más significativo asociado con este cultivo de sangre. ¿Cuál es la identidad del Organismo-1?*

» Desconocida

## 3.2 Procesos de Inferencia



*¿Se sabe si es un bastón, coco, etc?*

» Es un bastón

*¿Qué tinción tiene el Organismo-1?*

» Gram-negativo

*Mis recomendaciones terapéuticas van a basarse en las siguientes infecciones y microorganismos posibles:*

*Infección-1: bacteriemia primaria*

<1> El Organismo-1 puede ser PSEUDOMONAS-AERUGINOSA

<2> El Organismo-1 puede ser KLEBSIELLA-PNEUMONIAE

<3> El Organismo-1 puede ser ESCHERICHIA COLI

<4> El Organismo-1 puede ser BACTEROIDES-FRAGILIS

<5> El Organismo-1 puede ser ENTEROBACTER

<6> El Organismo-1 puede ser PROTEUS-NON-MIRABILIS

*Expresa el grado de gravedad de ROBERTO en una escala creciente de 0 a 4*

» 3

*¿Padece ROBERTO de reacciones alérgicas a algún agente antimicrobiano?*

» No

## 3.2 Procesos de Inferencia



*¿Cuál es el peso de ROBERTO?*

» 70

*(Rec. 1) Mi recomendación terapéutica es:*

*Para cubrir <1,2,3,5,6>*

*Administrar GENTAMICINA*

*Dosis: 119 mg (1,7 mg/kg) c/8h durante 10 días*

*Comentarios: Modificar dosis en caso de complicación renal*

*Para cubrir <4>*

*Administrar CLINDAMICINA*

*Dosis: 595 mg (8,5 mg/kg) c/6h durante 14 días*

*Comentarios: Si aparece diarrea u otro síntoma gastrointestinal, evaluar al paciente por posible colitis pseudomembranosa*



1. Introducción
2. Lógica de Predicados de Primer Orden
3. Reglas
  1. Reglas de Producción
  2. Proceso de Inferencia
- 4. Redes Semánticas**
5. Frames
6. Resumen

## 4. Redes Semánticas



- La psicología cognitiva afirma:
  - La representación y recuperación del conocimiento se realiza a partir de sus relaciones
  - Si se puede representar las relaciones mutuas entre conceptos se puede crear una estructura de conocimiento que sea un modelo cercano a la mente humana y así poder hacer deducciones y adquirir conocimiento como lo haría una persona
- Las redes semánticas trasladan esa afirmación a un Formalismo: *Son un método de representación declarativa de conceptos y relaciones mediante grafos orientados*
- Basadas
  - en modelos de psicología cognitiva
  - en la idea de que los objetos o los conceptos pueden ser unidos por alguna relación

## 4. Redes Semánticas



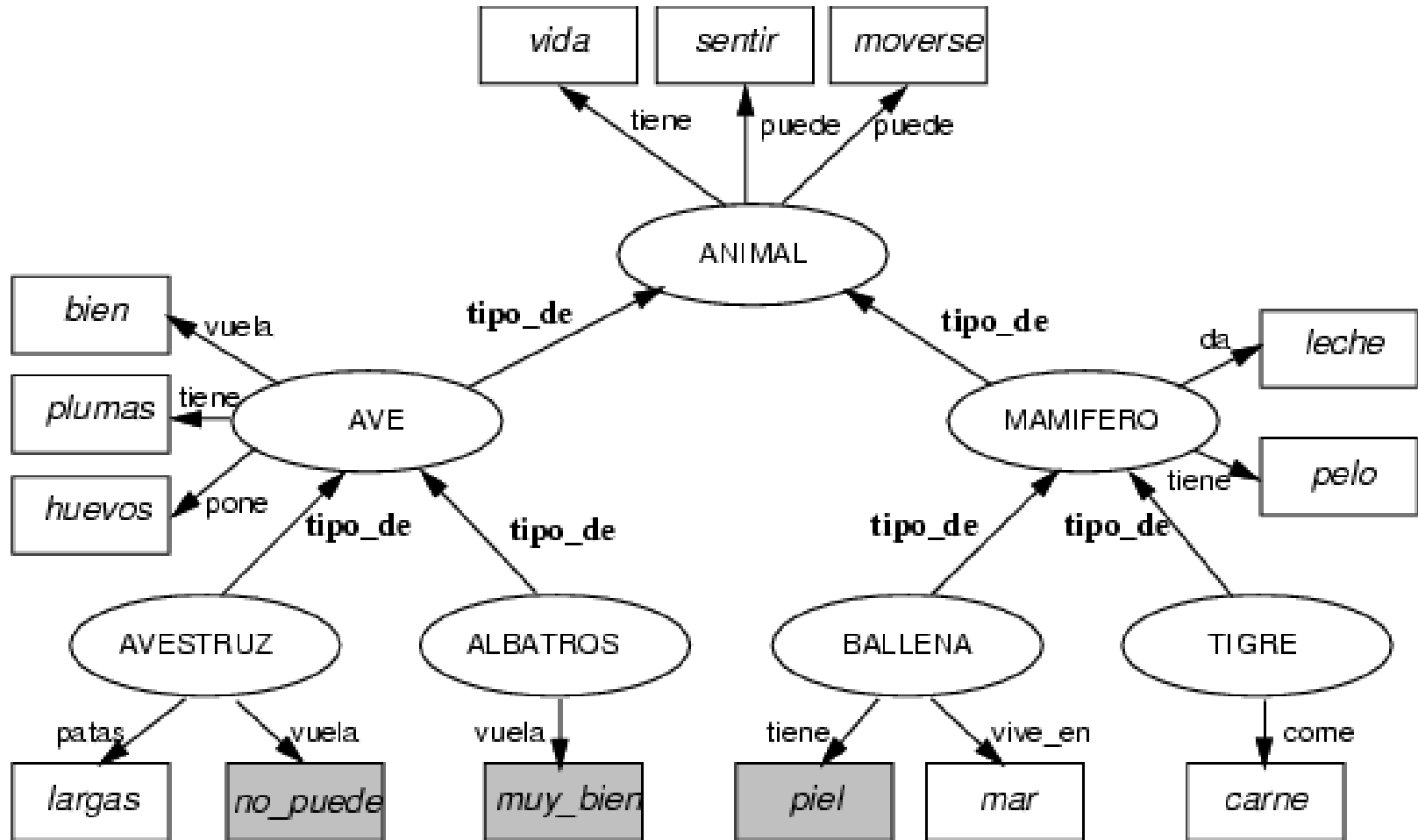
- Los primeros esquemas de representación formalizados fueron usados por Quillian (1968) y Shapiro & Wodmansee (1971) para representar el *significado semántico* de oraciones en inglés.
- Actualmente, se utiliza el término **redes asociativas** (una forma más amplia) ya que también se usa para representar
  - asociaciones físicas o causales entre varios conceptos u objetos.
  - mapas conceptuales
- Fundamentación psicológica muy sólida, por lo que se han realizado numerosos esfuerzos por llevar a cabo implementaciones importantes basadas en ellas.

# 4. Redes Semánticas



- Estructura
  - Los nodos se utilizan para representar conceptos
    - objetos
    - propiedades.
  - Los arcos (dirigidos) representan relaciones entre nodos
  - Sobre ellos operan procedimientos de inferencia .
    - El mecanismo de inferencia básico es el encadenamiento y la herencia de propiedades
  
- En caso de que no existan ciclos, estas redes pueden ser visualizadas como árboles.

# 4. Redes Semánticas





# 4. Redes Semánticas



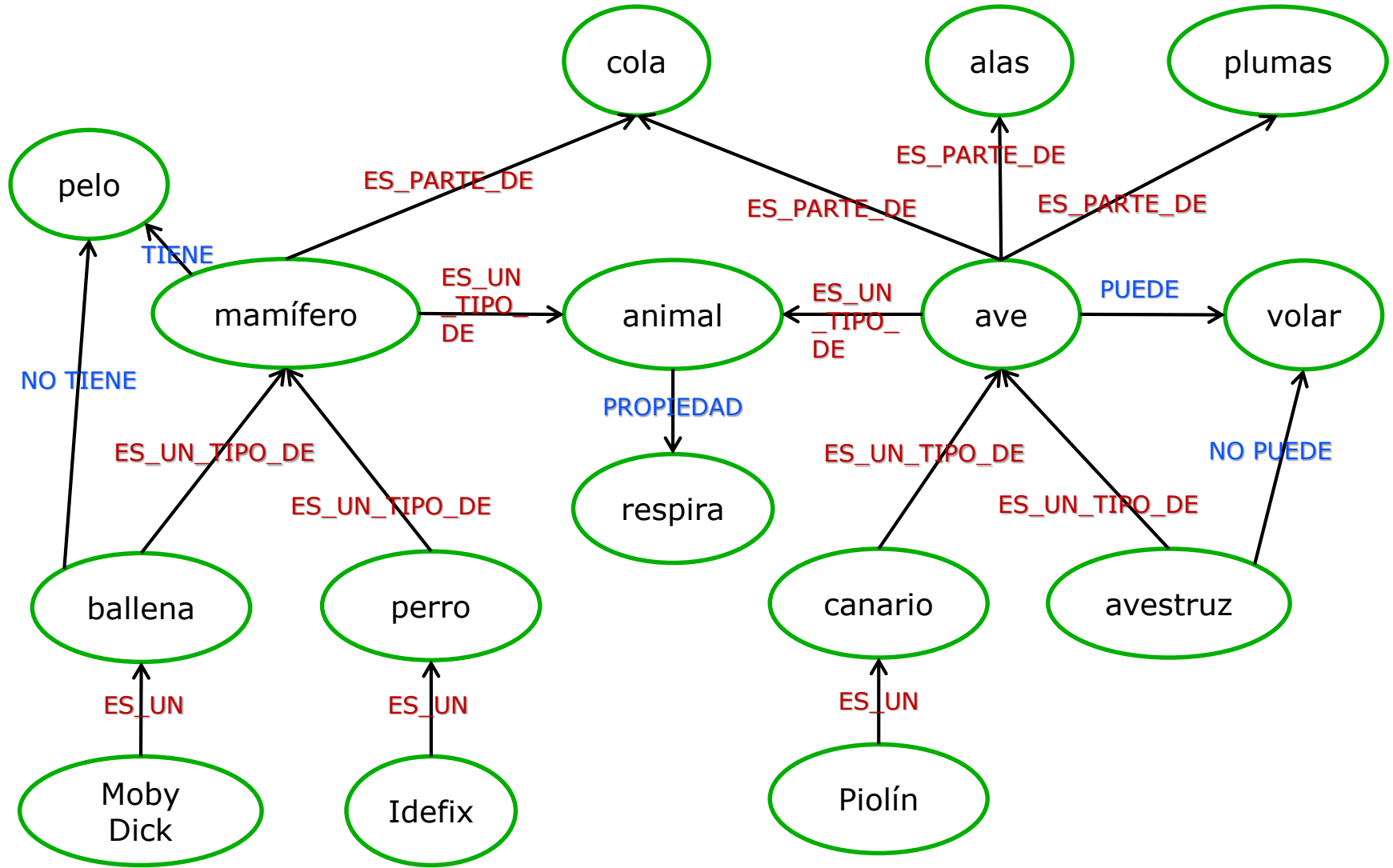
- La unidad elemental de una red semántica está formada por un par de nodos y un arco que los une
  - Nodos:
    - Nodo origen: objeto para el que se define el par atributo-valor
    - Nodo destino: representa los valores de los atributos
  - Arcos: atributos del par. Pueden ser de dos tipos:
    - Descriptivos: describen los conceptos mediante atributos propios del dominio.
    - Estructurales: enlazan los conceptos formando la estructura de la red. Son independientes del conocimiento del dominio que se está representando.
      - Instanciación: liga un objeto concreto con su tipo o clase genérica
        - Piolín **es\_un** Pájaro
      - Agregación: liga un concepto con sus componentes
        - Lente **es\_parte\_de** Cámara
      - Generalización: relaciona una clase con otra más general formando una red de nodos por especialización de conceptos.
        - Helicóptero **es\_un\_tipo\_de** Aeronave

## 4. Redes Semánticas



- En una red semántica, los conceptos están organizados en una jerarquía en la que
  - Existe un nodo superior (*top*)
  - al que se le asigna uno o varios nodos hijos (**ES\_UN\_TIPO\_DE**)
  - y así sucesivamente hasta que se alcanza el final (*bottom*)
  - cuyos nodos ya no son conceptos sino *instancias*. (**ES\_UN**)

# 4. Redes Semánticas



## 4. Redes Semánticas



- Categorías de redes semánticas:
  - Redes IS-A: en las que los enlaces entre nodos están etiquetados.
  - Gráficos conceptuales: en los que existen dos tipos de nodos: de conceptos y de relaciones
- Representan un conjunto restringido de la lógica de predicados
  - Pueden representar acciones o frases complejas, pero puede ser más fácil de comprender para los algoritmos (que incluso pueden llegar a ejecutarse más rápidamente).
  - El problema para utilizarlos como redes lógicas es representar la cuantificación, la negación, la implicación (reglas), y la disyunción

## 4. Redes Semánticas



- Se pueden establecer mecanismos de razonamiento específicos que permiten responder a preguntas sobre la representación
  - ¿Están relacionados dos conceptos?
  - ¿Qué relaciona dos conceptos?
  - ¿Cual es el concepto mas cercano que relaciona dos conceptos?
  
- Si definimos una semántica más rica sobre las relaciones se pueden responder preguntas mas complejas
  - Taxonomías entre conceptos (clase/subclase/instancia)
  - Generalizaciones/Especializaciones

## 4. Redes Semánticas



- Mecanismo de Razonamiento: Herencia por defecto
  - *Sistema de razonamiento que permite deducir propiedades de un concepto basándose en las propiedades de conceptos más altos en la jerarquía*
  - Un nodo hereda las propiedades de los conceptos “más altos en jerarquía” a través de relaciones del tipo es\_un y es\_un\_tipo\_de.
  - Razonamiento no monotónico
    - Se pueden poner excepciones de manera natural
    - Ejemplo: Los avestruces no vuelan

# 4. Redes Semánticas



- Ventajas
  - Declaración de asociaciones en forma explícita y concisa
  - Debido a que los nodos relacionados están directamente conectados el tiempo de búsqueda para hechos particulares puede ser significativamente reducido
- Desventajas
  - Válidas solo para casos simples.
    - Difícil manejo de estructuras grandes
    - Explosión combinatoria del número de relaciones a examinar para comprobar una relación, sobre todo si la respuesta es negativa.
  - Falta una base teórica al modelo de razonamiento
    - Formalismos distintos con diferentes capacidades
    - No existe una interpretación normalizada para el conocimiento expresado por la red: depende del programa que la manipula.
  - No se puede usar heurísticas para explorar la red de manera eficiente.



1. Introducción
2. Lógica de Predicados de Primer Orden
3. Reglas
  1. Reglas de Producción
  2. Proceso de Inferencia
4. Redes Semánticas
5. Frames
6. Resumen



# 5. Frames



- Fundamentación psicológica:
  - Al enfrentarnos a una situación determinada, intentamos ajustarla a otra parecida de la que ya tenemos experiencia previa y
  - *Esperamos* que aparezcan un número de elementos comunes y se sucedan algunas situaciones.
    - Por ejemplo, si entramos en una habitación de hotel, esperamos encontrar una cama, un armario, un baño, etc.
  - Nuestra mente reconoce las instancias específicas de la nueva habitación y los acomodará al estereotipo que ya poseemos.
  - La base teórica son las situaciones estereotipadas.
  
- Es aplicar la Programación Orientada a Objetos a la IA (de hecho, son previos a la OOP)

# 5. Frames



- En Inteligencia Artificial, *estructura de datos que contiene una descripción general de un objeto derivada de conceptos básicos y de la experiencia.*
  - Propuestos por Minsky en 1974
  - Son redes semánticas estructuradas que además pueden incluir representaciones de conocimiento procedimental
  - Cada nodo correspondiente a un objeto o a una clase se convierte en un marco
  
- Cada frame representa un concepto dividido en
  - Parte declarativa (atributos)
    - Describe la semántica del concepto (características)
  - Parte procedimental (métodos)
    - Definir como obtener información o hacer cálculos sobre sus características o las relaciones que pueda tener con otros frames

## 5. Frames



- Un Frame es, por tanto, una colección de atributos y la descripción de sus características → modo natural de representar el conocimiento

### **QANTAS BOARDING PASS**

**Carrier:** *QANTAS AIRWAYS*  
**Name:** *MR N BLACK*  
**Flight:** *QF 612*  
**Date:** *29DEC*  
**Seat:** *23A*  
**From:** *HOBART*  
**To:** *MELBOURNE*  
**Boarding:** *0620*  
**Gate:** *2*

### **AIR NEW ZEALAND BOARDING PASS**

**Carrier:** *AIR NEW ZEALAND*  
**Name:** *MRS J WHITE*  
**Flight:** *NZ 0198*  
**Date:** *23NOV*  
**Seat:** *27K*  
**From:** *MELBOURNE*  
**To:** *CHRISTCHURCH*  
**Boarding:** *1815*  
**Gate:** *4*

# 5. Frames

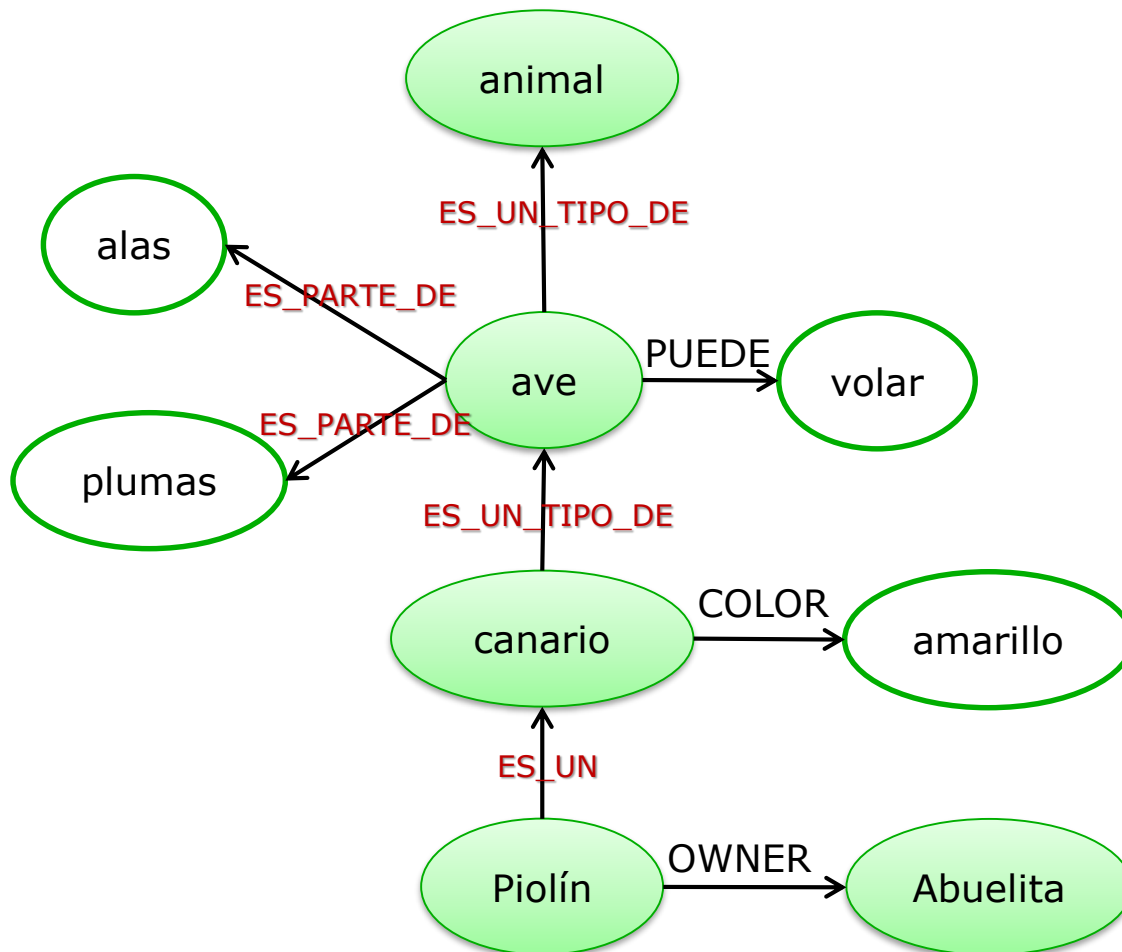


- Organización jerárquica que presenta los mismos sistemas de herencia que las redes IS-A.
  - Una estructura de marcos es una ampliación de una estructura de redes IS-A básica donde los nodos son reemplazados por grupos estructurados de información (marcos)
  
- Una jerarquía de marcos es un árbol en el que cada uno de los nodos es un marco
  - Para cada nodo hijo, el enlace con el nodo padre es un enlace de herencia
  - El nodo hijo hereda todos los *slots* de su padre a menos que se especifique lo contrario (*sobrecontrol*)
  - Los nodos pueden ser de dos tipos:
    - Nodos de clase (todos los nodos no terminales han de ser de clase)
    - Nodos de instancia

# 5. Frames



Relación Red Semántica <> Frame



## Ave

ES\_UN\_TIPO\_DE: animal

ES\_PARTE\_DE: alas, plumas

Puede: volar

## Canario

ES\_UN\_TIPO\_DE: ave

Color: amarillo

## Piolín

ES\_UN: canario

Owner: Abuelita

# 5. Frames



- Se describe mediante su nombre y la lista de atributos

**Frame** <nombre>

slot <nombre-slot>

...

slot <nombre-slot>

métodos

**acción** <nombre-método> (parámetros)

**función** <nombre-método> (parámetros) devuelve <tipo>

- Una primera línea con el nombre del marco
- Una sucesión de líneas, llamadas slots que describen atributos particulares de ese frame
- Slots especiales (métodos) dan la posibilidad de mezclar conocimiento procedimental con declarativo



## Slots

**Slot** <nombre>

++ dominio (lista de frames)

++ rango <tipo-simple>

++ cardinalidad (1 o N)

valor (valor o lista de valores)

demons <tipo-demon>

**accion** <nombre-accion> / **función**<nombre-funcion> devuelve <tipo>\*

herencia

- Describen las características del frame en los *facets* (++) *son obligatorios*)
- Se puede definir procedimientos (*demons*, sin argumentos) para realizar cálculos bajo ciertos eventos
  - If-needed (al consultar el slot)
  - if-added (al asignar valor al slot),
  - if-removed (al borrar el valor)
  - if-modified (al modificar el valor)

# 5. Frames



## Métodos

- Acciones o funciones (con parámetros) que permiten obtener información sobre el frame
- Estos métodos pueden invocarse desde frames abstractos (clases) o frames concretos (instancias)
- Pueden ser
  - heredables (permitimos invocarlos en los descendientes)
  - no heredables (exclusivos del frame)

## Relaciones

- Permiten conectar los frames entre si para expresar su relación





1. Introducción
2. Lógica de Predicados de Primer Orden
3. Reglas
  1. Reglas de Producción
  2. Proceso de Inferencia
4. Redes Semánticas
5. Frames
6. Resumen

## 6. Resumen



Técnica	Descripción	Formalismo	Representación
<b>Lógica</b>	<ul style="list-style-type: none"><li>• Representación: Lenguaje de la lógica</li><li>• Razonamiento: Procedimientos de resolución sintáctica (inferencia)</li></ul>	Declarativo	Basada en relaciones
<b>Reglas</b>	<ul style="list-style-type: none"><li>• Representación: Cláusulas de Horn</li><li>• Razonamiento: MPG encadenado hacia delante o hacia atrás</li></ul>	Declarativo	Basada en acciones
<b>Redes Semánticas</b>	<ul style="list-style-type: none"><li>• Representación: Grafos dirigidos que representan conceptos y relaciones</li><li>• Razonamiento: Encadenamiento y Herencia</li></ul>	Estructurado	Basada en relaciones
<b>Marcos</b>	<ul style="list-style-type: none"><li>• Representación: Patrones (clases) de conceptos, situaciones...</li><li>• Razonamiento: Herencia/Herencia múltiple</li></ul>	Estructurado	Basada en objetos