

# Esquemas Híbridos

Eduardo Morales, Enrique Sucar

INAOE

# Contenido

- 1 Introducción
- 2 Híbridos Externos
- 3 Híbridos Internos  
Centaur: Marcos y Reglas.  
Nexpert: Reglas y Objetos
- 4 Implementación

# Esquemas Híbridos

## Introducción

### Híbridos Externos

### Híbridos Internos

Centaur: Marcos y  
Reglas.

Nexpert: Reglas y  
Objetos

## Implementación

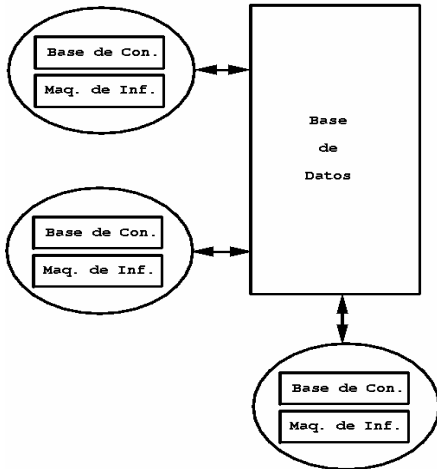
- Las formas de representar conocimiento que vimos no son mutuamente exclusivas.
- Un esquema híbrido es una combinación de diversas formas de representación de conocimiento para resolver un problema.
- Hay dos formas básicas de combinar diversas representaciones: Externa e Interna.

# Híbridos Externos

- En este esquema dos o mas módulos con diferentes formas de representación interactúan entre si.
- Cada módulo tiene una sola forma de representación y se combina con los otros módulos mediante variables de entrada/salida o mediante una estructura de datos común (Base de Datos).
- En principio cada subsistema tiene la forma de representación más adecuada para resolver una parte del problema, y se combina con las demás para solucionar un problema mayor.
- Este esquema da origen al sistema de pizarrón, y al hacerse en forma distribuida, a los sistemas multi-agentes.

# Híbridos Externos

## Híbridos "Externos"



# Híbridos Internos

- En este tipo de sistemas se combinan varias formas de representación que interactúan para resolver cierto problema.
- Con esto se aprovechan diversas propiedades de las formas de representación que complementan sus capacidades.
- Por ejemplo, se combinan las reglas con prototipos aprovechando las abstracciones de marcos dentro de reglas, o marcos y redes semánticas formando redes de prototipo, etc.
- Dos ejemplos de este tipo de esquemas son: *Centaur*, que combina marcos y reglas para diagnóstico médico; y *Nexpert*, una herramienta que involucra reglas y objetos para el desarrollo de sistemas expertos.

# Centaur: Marcos y Reglas

- Originalmente diseñado para el diagnóstico de enfermedades pulmonares, ha sido extendido para otro tipo de aplicaciones.
- La idea básica es la de asociar reglas a marcos. Es decir, que una estructura tipo marco provee un contexto explícito en el que actúan ciertas reglas.
- Las reglas se ligan a un atributo de un marco, y se ven simplemente como un “slot” adicional del prototipo correspondiente.
- De esta forma, el marco indica la “situación” o contexto en que aplica la regla, evitando los “trucos” que se tienen que hacer en sistemas de producción “puros” para tener un efecto similar.

# Reglas dentro de Prototipos

- Centaur tiene una clasificación de enfermedades pulmonares que se estructuran en una jerarquía de prototipos.
- Cada prototipo contiene un número de marcos (subprototipos) que incluyen el conocimiento e información referente ese tipo de enfermedad; y asociado a c/u de estos hay una serie de reglas que indican como obtener dicha información.
- También los marcos pueden tener meta-reglas que le indican como razonar con dichas estructuras.



Esquemas  
HíbridosEduardo  
Morales,  
Enrique Sucar

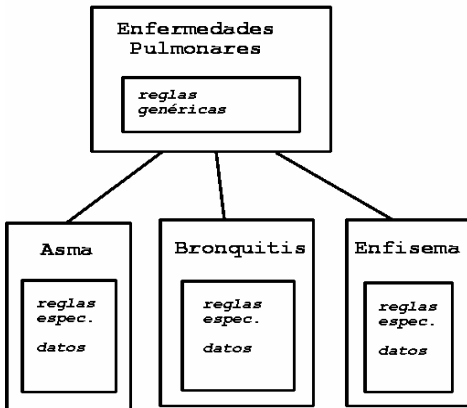
Introducción

Híbridos  
ExternosHíbridos  
InternosCentaur: Marcos y  
Reglas.Nexpert: Reglas y  
Objetos

Implementación

# Híbridos Externos

## CENTAUR



# Jerarquía de prototipos en Centaur

- En operación, primero se dan ciertos datos iniciales de la enfermedad.
- Estos activan ciertas reglas que llevan a la activación de algunos prototipos.
- Se tiene una forma de darles prioridad a los prototipos, y se escoge para su evaluación el de mayor prioridad.
- Se obtiene la información referente a ese marco, y se continua el ciclo hasta llegar a cierto nivel de confianza en los resultados.
- De esta forma se usan marcos para estructurar reglas en forma modular, y controlar la interacción entre ellas.

# Nexpert: Reglas y Objetos

- *Nexpert* es una herramienta (coraza o *shell*) de propósito general para el desarrollo de sistemas expertos.
- Se basa en la combinación de reglas y objetos.
- En cierta forma es similar a Centaur, ya que los objetos de Nexpert se pueden considerar como un sistema de prototipos.
- Sin embargo, la forma en que interactúan estas 2 representaciones es diferente, ya que en vez de agrupar reglas dentro de objetos, estas 2 representaciones se ven como dos dimensiones del conocimiento que interactúan (se intersectan) entre sí.

# Interacción entre objetos y reglas

- Las reglas operan sobre atributos de objetos.
- Las reglas pueden ser genéricas operando sobre clases o partes de objetos (*pattern-matching*).
- Al evaluar reglas se pueden heredar atributos de la jerarquía de clases/objetos y se pueden disparar los métodos para obtener valores (*demons*).

Esquemas  
HíbridosEduardo  
Morales,  
Enrique Sucar

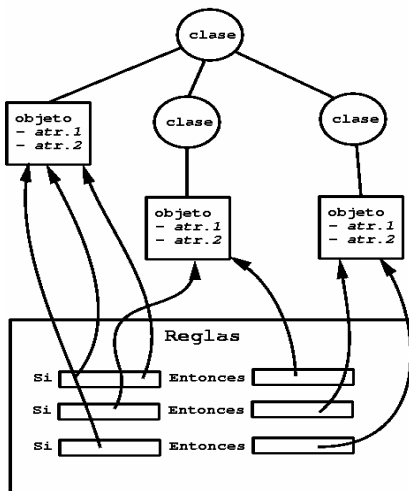
Introducción

Híbridos  
ExternosHíbridos  
InternosCentaur: Marcos y  
Reglas.Nexpert: Reglas y  
Objetos

Implementación

## Híbridos Externos

## Nexpert Object



# Implementación

Una regla puede ser representada por un frame:

```
(frame reglaN  
  (if (valor: ...))  
  (then (valor: ...))  
  (notas (valor: ...)))
```

# Implementación

Una regla puede apoyarse en la estructura de los frames:

(Regla N

(IF (frame<sub>1</sub> propiedad<sub>1,i</sub> valor<sub>1,i</sub>)

(frame<sub>2</sub> propiedad<sub>2,j</sub> valor<sub>2,j</sub>)

...)

(THEN (frame<sub>n</sub> propiedad<sub>n,k</sub> valor<sub>n,k</sub>)

(frame<sub>m</sub> propiedad<sub>m,l</sub> valor<sub>m,l</sub>)

...))

# Implementación

- La regla toma valores de *frames* en sus condiciones y modifica/genera frames en sus acciones.
- También se puede combinar con mecanismos de herencia para obtener valores de *frames* a partir de herencia de sus antecesores.
- Por ejemplo:

(regla 37

(If (clase ?X camion)

(valor peso ?X ?P)

(> ?P 10000)

(min-cardinalidad llantas ?X 10))

(THEN (clase ?X camion-grande)))



# Implementación

- Al disparar la regla añadimos a una instancia de camión la clase de camión-grande y por lo tanto hereda todos los valores del frame camión-grande.
- También se pueden organizar módulos de reglas al asociar éstos con los *frames*.
- Por ejemplo, un frame camión asociado a un método que invoque reglas de diagnóstico (heredables a sus especializaciones).
- Puede servir para guiar las reglas por especificidad.

# Algunos sistemas comerciales

- KEE: frames, reglas, Lisp
- ART: OPS5, TMS
- Knowledge Craft: OPS5, Prolog, CRL