

# Sistemas Basados en Conocimiento

Introducción

Arquitectura - Motor de Inferencias

LCC - Introducción a la Inteligencia  
Artificial

Dra. Ana Casali

# Inicio de los Sistemas Expertos

**60's**



**Se buscaban soluciones generales**



**70's**



**80's**



**Los sistemas son más eficientes en dominios acotados**



**La calidad y cantidad de conocimiento son esenciales para una buena performance**



**Es importante independizar el conocimiento del dominio, del mecanismo de inferencia.**

# Sistemas basados en conocimiento (SBC-KBS)

## Definiciones.

➤ Sistemas que resuelven problemas aplicando una representación simbólica de la experiencia humana. Jackson.

➤ Sistemas que representan el conocimiento sobre el dominio en forma explícita y separada del resto del sistema.

Waterman.

# Sistemas basados en conocimiento (SBC-KBS)

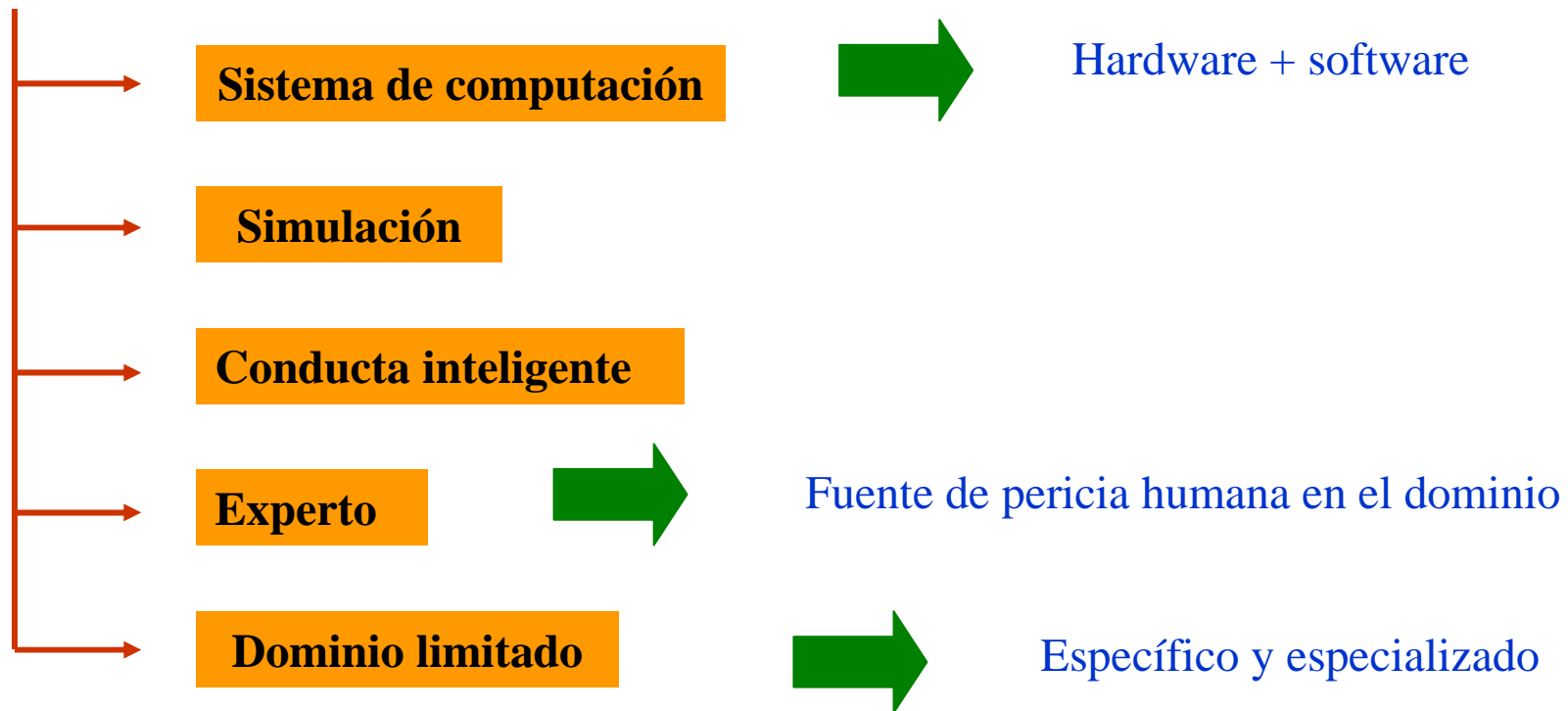
## Definiciones.

➤ Sistemas de software capaz de soportar la **representación explícita del conocimiento** de un **dominio específico** y de **explotarlo a través de mecanismos apropiados de razonamiento** para proporcionar un comportamiento de alto nivel en la resolución de problemas.

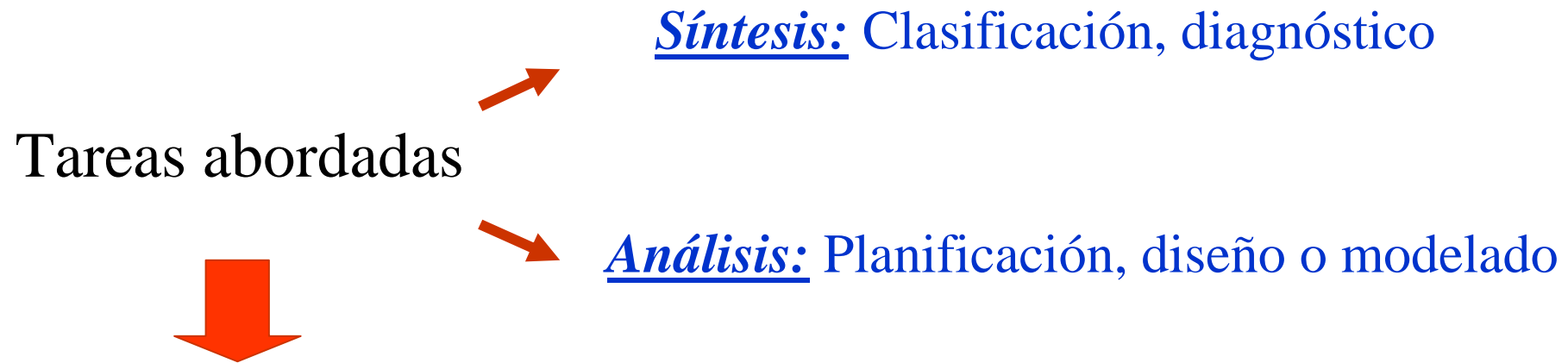
Guida y Taso

# Sistemas Expertos: Definición.

Es un sistema de cómputo capaz de simular la conducta inteligente de un experto humano en un dominio específico y especializado, con el objeto de resolver problemas.



# SE/SBC: Tareas



*Las áreas de aplicación son muy variadas !!!*

- Los SBC abordan problemas complejos en dominios específicos en los que el peso de las heurísticas para acotar el espacio de búsqueda es importante.

# Primeros Sistemas Expertos

MYCIN

Stanford (Buchanan - Shortliffe 1976): Diagnóstico y tratamiento de enfermedades infecciosas (aprox. 500 reglas). Introduce factores de certeza.

PROSPECTOR

Stanford-US Geological Survey (Duda-Gashning-Hart, 1979): Evaluación de yacimientos minerales y petrolíferos. (aprox 1600 reglas). Probabilidades condicionales y Teorema de Bayes.

Conocimiento  
inexacto

INTERNIST

Pittsburg (Pople-Myers, 1977): Diagnóstico en Medicina interna. CADUCEUS: Analiza relaciones causales.

CASNET

Rutgens (Kulikowsky-Weiss-Safir, 1977): Oftalmología. Aplica un modelo causal para el diagnóstico de glaucoma.

Razonamiento experto  
Modelos causales

R1

Carnegie Mellon (Mc Dermott): Configuración de ordenadores VAX (aprox 500 reglas). Descompone el diseño en distintos niveles de abstracción.

Reducción espacio  
de búsqueda

MOLGEN

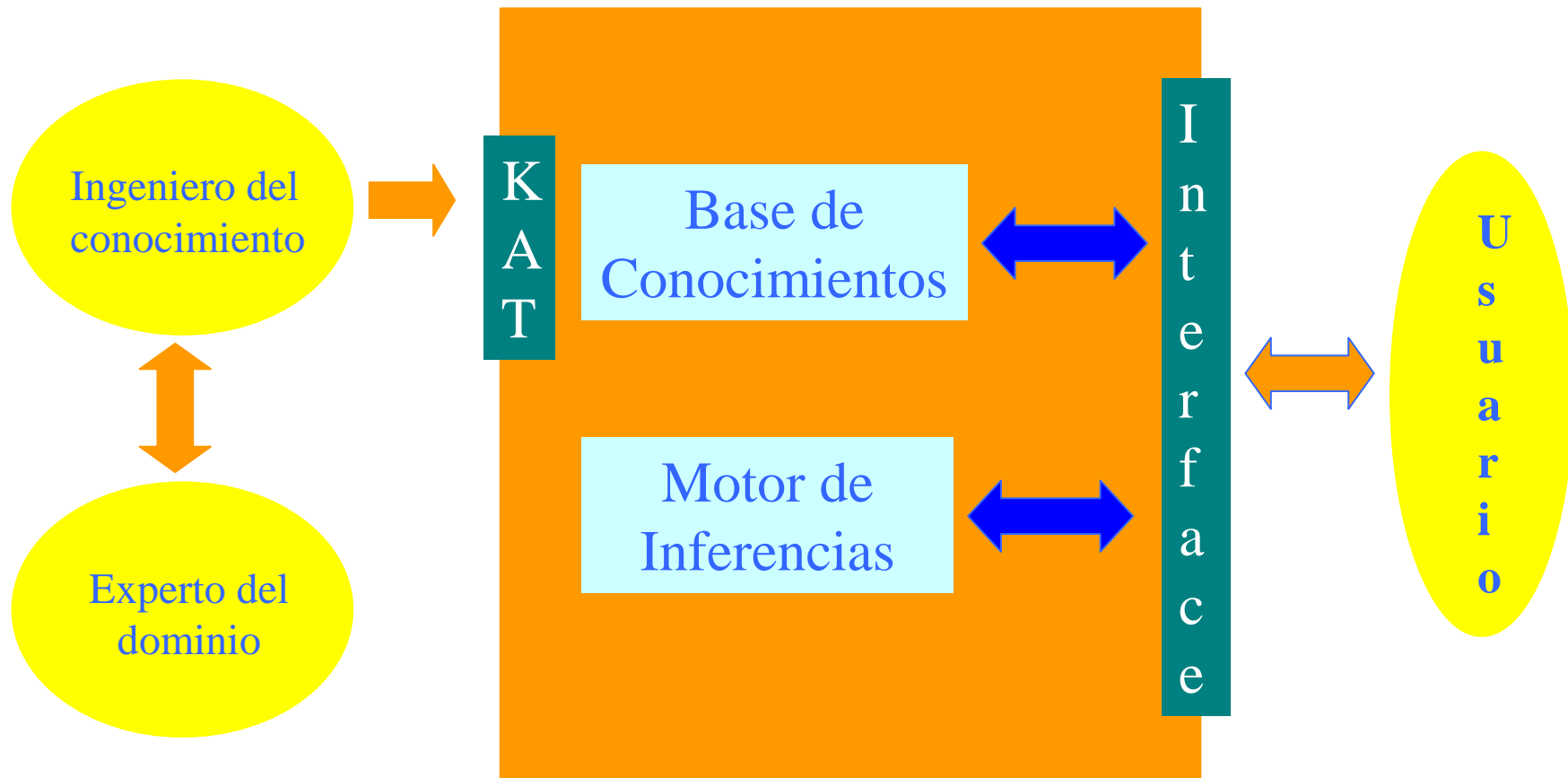
Stanford (Stefik): Diseño de experimentos genéticos. Utiliza abstracción.

# Habilidades que se esperan de un SE

- ✓ Manipular con fluidez descripciones simbólicas.
- ✓ Buena performance, comparable al menos a la de los especialistas.
- ✓ Interfaz amigable, adecuada a las necesidades de la aplicación.
- ✓ Habilidad para manipular conocimiento incompleto e impreciso.
- ✓ Capacidad de explicar sus decisiones a los usuarios.
- ✓ Posibilidad de justificar sus conclusiones.

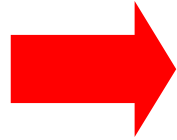


# Estructura básica de un SE.



# Estructura básica de un Sistema Experto

Base de  
Conocimientos

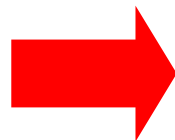


Su estructura de datos queda definida en términos del esquema de representación elegido para incorporar el conocimiento del dominio de trabajo.



Separados entre sí

Motor de  
Inferencias



Es la estructura de control de un SE, contiene el programa que gestiona la BC y otros mecanismos necesarios para administrar un sistema de naturaleza interactiva.

# Periodo industrial de la IA

Década de  
los 80

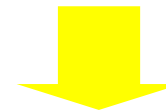


Importante apoyo económico a los esfuerzos de I+D.  
Gran cantidad de SE en distintos dominios.



*Problemas con la metodología de desarrollo*

*Limitaciones propias del tipo de sistema.*



\* Necesidad de IC

\* Buenas soluciones para  
cierto tipo de problemas

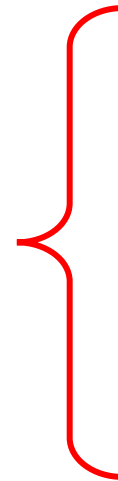
# INGENIERIA DEL CONOCIMIENTO

- Es la disciplina tecnológica que se centra en el *desarrollo, funcionamiento y mantenimiento* de Sistemas Basados en Conocimiento (SBC – KBS) de una forma sistemática, disciplinada y cuantificable (utilizando Ingeniería de Software)

# INGENIERIA DEL CONOCIMIENTO

Conjunto de Conocimientos y Técnicas que permiten aplicar el saber científico a la utilización del conocimiento.

QUE SE OCUPA DE LA



ADQUISICION

REPRESENTACION

INFERENCIA

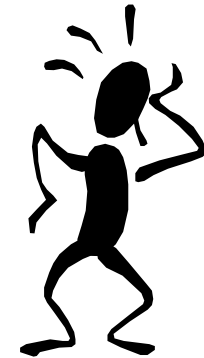
# SBC: Ventajas



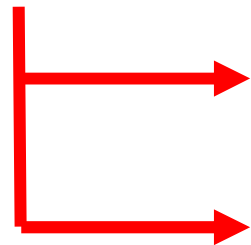
- **El conocimiento no se pierde.**
  - disponible en ambientes hostiles o con carencia de especialistas.
- **Reducción del espacio de búsqueda con heurísticas para que el problema sea tratable en un tiempo razonable.**
- **Manejo de conocimiento incierto e incompleto.**
- **Posibilidad de justificar el razonamiento seguido.**
- **Aumento de fiabilidad, evitando que prevalezcan las últimas experiencias.**
- **Modificación sencilla de la BC por su característica modular.**

# SBC: Inconvenientes

- La adquisición del conocimiento es difícil y cara.
- La reutilización del conocimiento en contextos diferentes no es simple.
- Falta de creatividad y sentido común.
- Obstáculos para el aprendizaje y la adaptación.



Quedan inmersos en el campo de los Sistemas Inteligentes.



Se trabaja sobre metodología de desarrollo

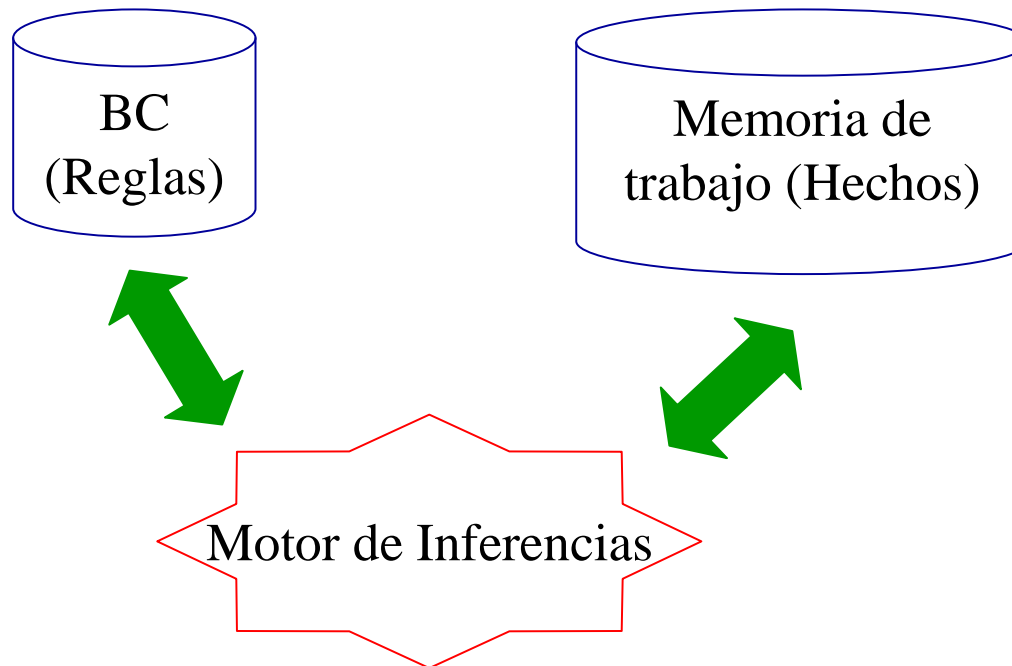
Se los combina con otras tecnologías

# SE basados en reglas de producción

Sistemas de  
producción



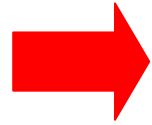
Newell y Simon (1972): Al resolver problemas, las personas utilizan su *memoria a largo plazo* (permanente) que aplican a situaciones actuales contenidas en su *memoria a corto plazo* (transitoria). Esto puede generar modificaciones en la última.





# SE basados en reglas de producción

Reglas de  
producción



< CONDICION >

< ACCION >



⇒ Son “gránulos” de conocimiento.

⇒ Reúnen información relativa a las condiciones de disparo y a los efectos resultantes del disparo.

⇒ Son estructuras bidireccionales.

SI X ES MAMIFERO Y X COME CARNE

ENTONCES X ES CARNIVORO.


## SE basados en reglas de producción

- ✓ Se utilizan Reglas de Producción para representar el conocimiento  
IF <premisa> THEN <conclusión y/o acción>
- ✓ Son los elementos de deducción básicos
- ✓ El proceso de inferencia se basa fundamentalmente en la Regla de Inferencia de la lógica denominada MP  
 $A \rightarrow B, A / \therefore B$

# Reglas de producción

Modelo formal para representar un elemento mínimo de conocimiento

IF <premisa> THEN <conclusión y/o acción>

<conclusión y/o acción> 

- Conclusión
- Acción
- Estrategia

✓ La premisa puede tener conectivos lógicos

<premisa> = <cláusula<sub>1</sub> AND/OR...AND/OR  
cláusula<sub>k</sub>>

# SE basados en reglas de producción

↳ *Naturaleza modular:* El conocimiento es fácil de encapsular y expandir.

↳ *Explicaciones sencillas:* El registro de reglas disparadas permite presentar la cadena de razonamiento asociada.

↳ *Semejanza con el proceso cognitivo humano:* modelo natural del razonamiento humano.

↳ Completitud y consistencia como responsabilidad del desarrollador.

↳ Se necesitan otras estructuras para lograr una visión global del conocimiento del dominio.

➤ Se las combina con otros formalismos de representación.

# Reglas de producción- Ejemplos

## Sintaxis Reglas en JESS

```
Jess> (defrule bienvenido-menor
```

```
"Saludo para niños"
```

```
(persona {edad < 3})
```

```
=>
```

```
(printout t "Hola niño!"
```

```
crlf)
```

# Sistemas de producción.

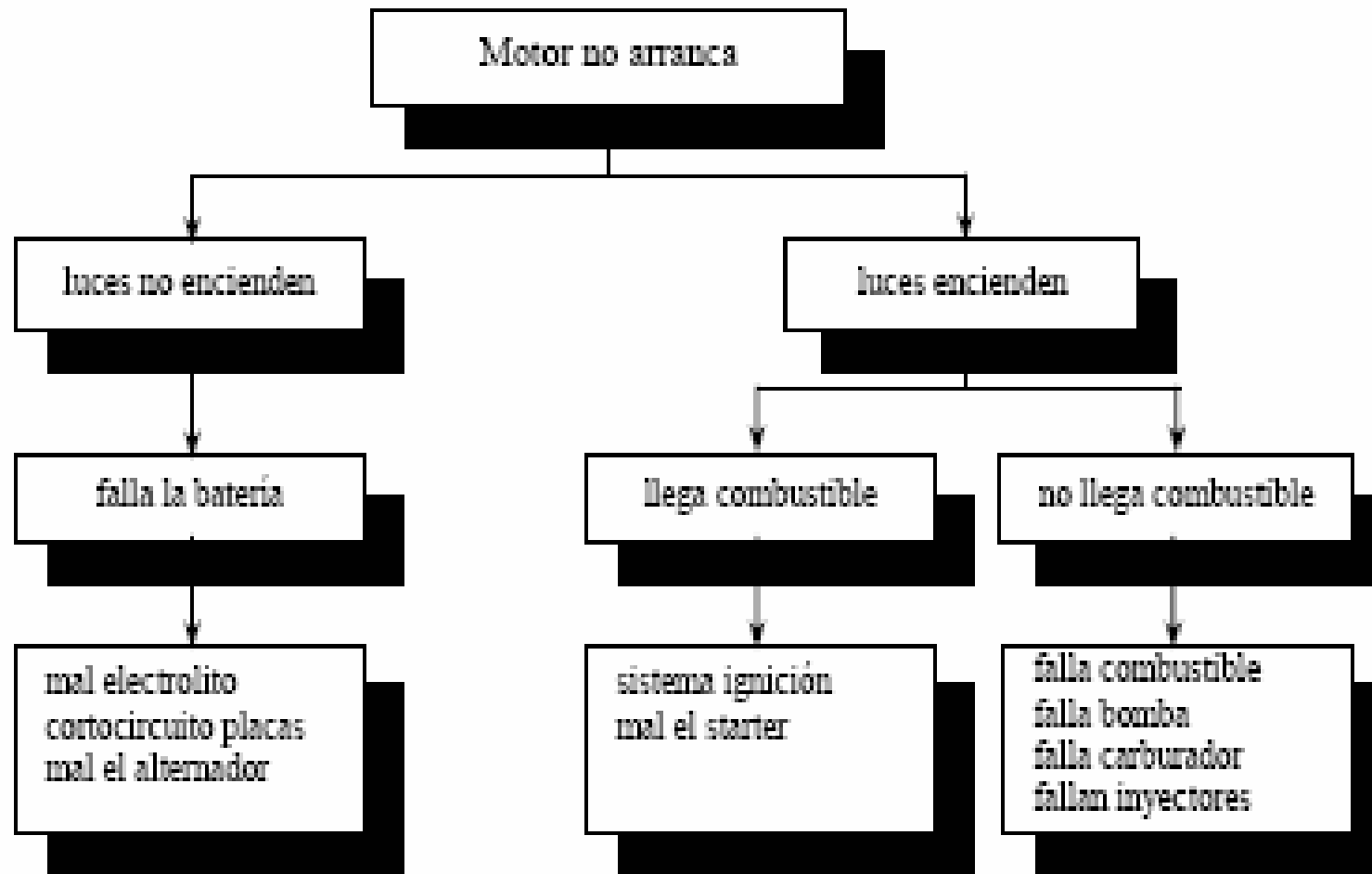
## Un ejemplo de regla

En el desarrollo de un sistema basado en conocimiento relativo al diagnóstico de una falla en un auto, se podrá expresar:

*“ si el motor no arranca y las luces no encienden, entonces la falla está en la batería ”.*

# Sistemas de producción.

## Un ejemplo



# El desarrollo de las reglas de producción para construir la KB.

Cada una de estas reglas aparece como un gránulo de conocimiento que es independiente



La ampliación de la KB puede realizarse sin mayor problema.

If el motor no arranca  
And las luces no encienden  
Then falla la batería.

If el motor no arranca  
And las luces encienden  
And llega combustible  
Then falla el sistema de ignición.

If el motor no arranca  
And las luces encienden  
And no llega combustible  
Then falla el sistema de combustible.



# El desarrollo de las reglas de producción para construir la KB.

estas reglas deben estar relacionadas por los hechos que mencionan para que puedan ser interpretadas en el proceso de inferencia.

respetar la sintaxis e los hechos representados

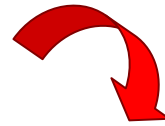


```
If el motor no arranca
And las luces no encienden
Then falla la batería.

If el motor no arranca
And las luces encienden
And llega combustible
Then falla el sistema de ignición.

If el motor no arranca
And las luces encienden
And no llega combustible
Then falla el sistema de combustible.
```

Sistemas de producción  
Como razonamos???



**MECANISMO DE INFERENCIA**  
**(MOTOR DE INFERENCIA)**



El cual determina de que forma utilizar las  
reglas para alcanzar el objetivo planteado

# La ejecución del proceso de inferencia en los KBS

¿El motor arranca?	
SI	
NO	X

¿Las luces encienden?	
SI	
NO	X

¿Llega combustible al motor?	
SI	X
NO	

Las respuestas seleccionadas originan los hechos: **“el motor no arranca”**, **“las luces no encienden”** y **“llega combustible”** que ingresarán a la memoria de trabajo del KBS.

# La ejecución del proceso de inferencia en los KBS

If el motor no arranca  
And las luces no encienden  
Then falla la batería.

If el motor no arranca  
And las luces encienden  
And llega combustible  
Then falla el sistema de ignición.

If el motor no arranca  
And las luces encienden  
And no llega combustible  
Then falla el sistema de combustible.

El motor de inferencia, explora la KB buscando reglas que puedan ser disparadas (por ej. coincidencias en las premisas).

La primera regla se desencadena, y produce entonces la conclusión: **“falla la batería”**, y el problema queda resuelto.

# Inferencia en los KBS

Las herramientas actuales soportan procesos de matching más complejos, que satisfacen algún predicado:

- que el valor de un slot de un objeto cumpla alguna condición, por ej.
  - que su valor sea desconocido
  - que la lista asociada a un slot tenga 3 elementos

```
Jess> (defrule match-tres-items  
        (lista-almacen ? ? ?)
```


=>

```
(printout t "Hay una lista de tres items."  
          crlf))
```

# Sistemas de producción

## Motor de Inferencia

### Direcciones de búsqueda:

- 
- ✓ Hacia delante, *Forward Chaining* o guiada por los hechos. **JESS**
  - ✓ Hacia atrás, *Backward Chaining* o guiada por los objetivos.

Para seleccionar las reglas candidatas en cada estado utiliza el FILTRADO o MATCHING.

# SE basados en reglas de producción

MOTOR DE  
INFERENCIAS



*Dos formas de funcionamiento.*



**Inductivo:** A partir de un objetivo intenta verificar los hechos que los sostienen



BACKWARD  
CHAINING



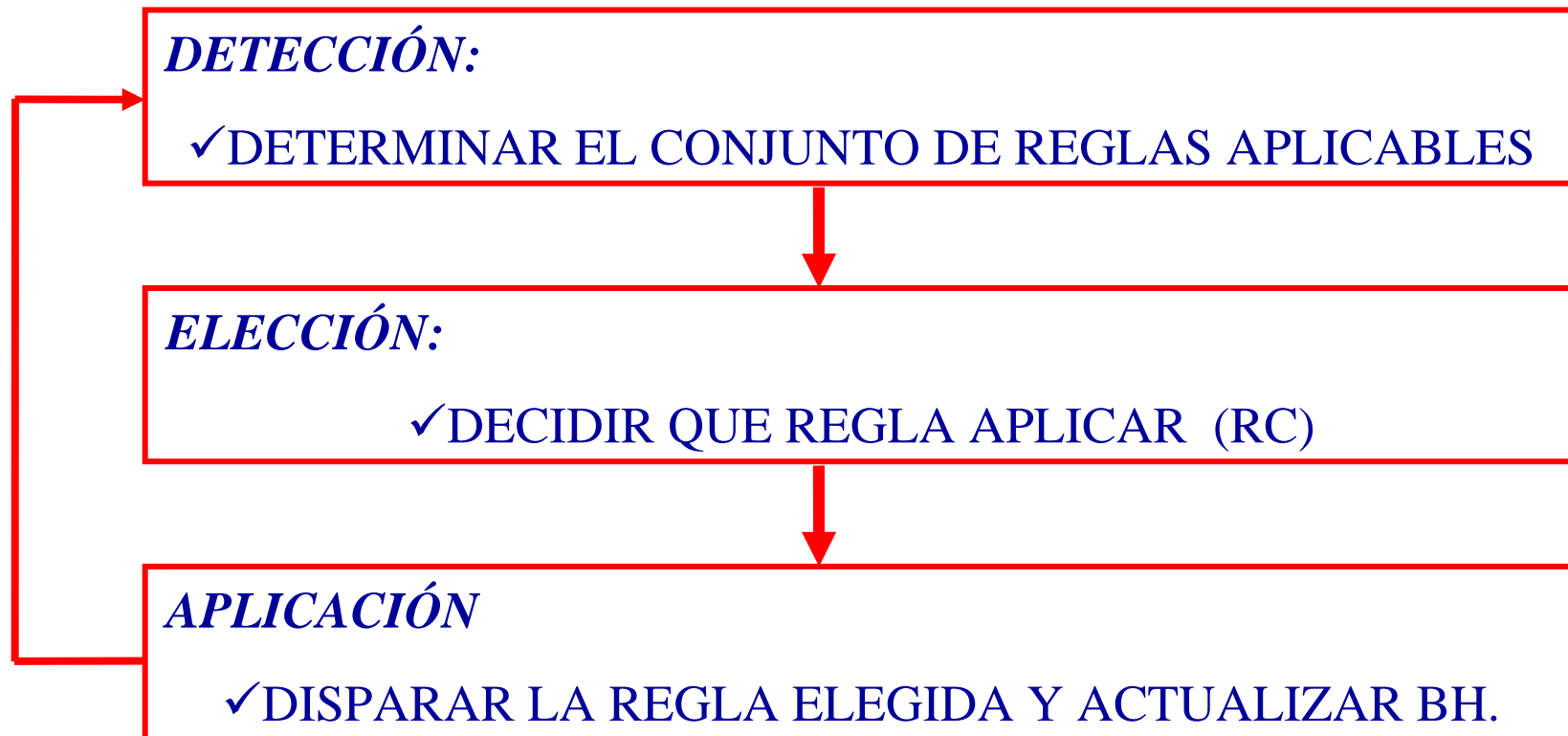
**Deductivo:** A partir de los hechos disponibles infiere todas las conclusiones posibles



FORWARD  
CHAINING

# SE basados en reglas de producción

Forward Chaining: Ciclo base de un motor deductivo.





# SE basados en reglas de producción

Encadenamiento hacia adelante - Forward Chaining.

BH := CONOCIMIENTO INICIAL (HECHOS).

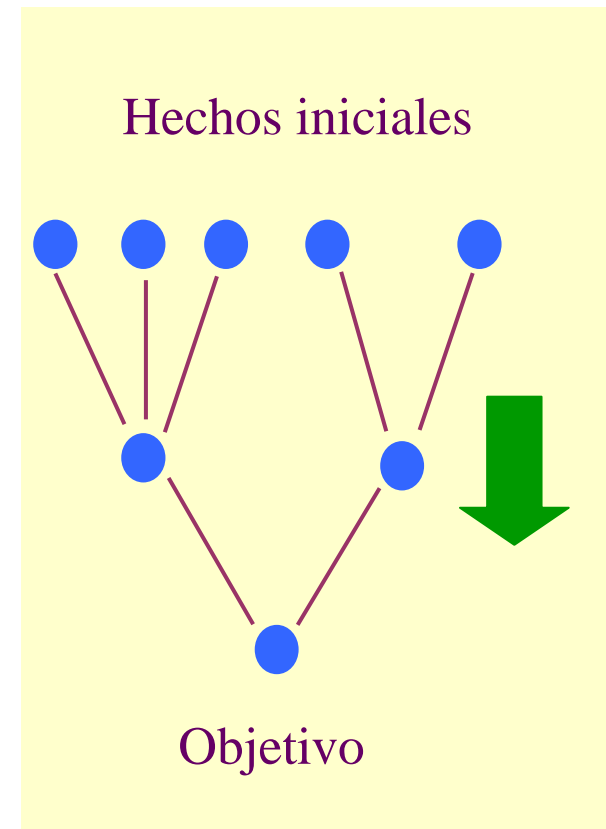
HASTA OBJETIVO O SIN REGLAS PARA DISPARAR.

**REPITA:**

(1) ENCONTRAR K CONJUNTO DE REGLAS CUYAS PREMISAS CUMPLEN CON BH (CONJUNTO DE CONFLICTO-CC).

(2) ELEGIR R DE K SEGÚN ESTRATEGIA DE SOLUCIÓN DE CONFLICTOS (RC).

(3) DISPARAR R Y ACTUALIZAR BH. (RECORDAR R).



# SE basados en reglas de producción

## Ejemplo Forward Chaining: Primer paso

### BASE DE REGLAS

R1:  $p \wedge q \rightarrow s$

R2:  $r \rightarrow t$

R3:  $s \wedge t \rightarrow u$

R4:  $s \wedge r \rightarrow v$



BH: p q r



➤  $CC = \{ R1, R2 \}$

➤  $R1 \leftarrow RC$

➤ DISPARA R1

➤  $BH \leftarrow s$

➤ R1 APLICADA

# SE basados en reglas de producción

## Ejemplo Forward Chaining: 2º paso

### BASE DE REGLAS

R1:  $p \wedge q \rightarrow s$

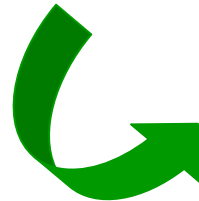
R2:  $r \rightarrow t$

R3:  $s \wedge t \rightarrow u$

R4:  $s \wedge r \rightarrow v$



BH: p q r s



➤  $CC = \{ R2, R4 \}$

➤  $R2 \leftarrow RC$

➤ DISPARA R2

➤  $BH \leftarrow t$

➤ R2 APLICADA

# SE basados en reglas de producción

## Ejemplo Forward Chaining: 3<sup>er</sup> paso

### BASE DE REGLAS

R1:  $(p \wedge q \rightarrow s)$

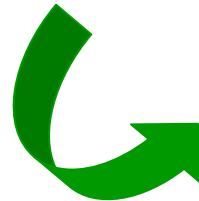
R2:  $(r \rightarrow t)$

R3:  $s \wedge t \rightarrow u$

R4:  $s \wedge r \rightarrow v$



BH: p q r s t



➤  $CC = \{ R3, R4 \}$

➤  $R3 \leftarrow RC$

➤ DISPARA R3

➤  $BH \leftarrow u$

➤ R3 APLICADA

# SE basados en reglas de producción

Ejemplo Forward Chaining: 4º paso

## BASE DE REGLAS

(R1:  $p \wedge q \rightarrow s$ )

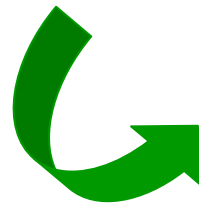
(R2:  $r \rightarrow t$ )

(R3:  $s \wedge t \rightarrow u$ )

R4:  $s \wedge r \rightarrow v$



BH: p q r s t u



➤  $CC = \{R4\}$

➤  $R4 \leftarrow RC$

➤ DISPARA R4

➤  $BH \leftarrow v$

➤ R4 APLICADA

# SE basados en reglas de producción

Ejemplo Forward Chaining: 5º paso

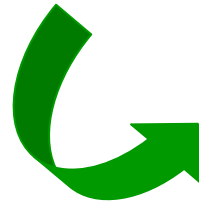
## BASE DE REGLAS

(R1:  $p \wedge q \rightarrow s$ )

(R2:  $r \rightarrow t$ )

(R3:  $s \wedge t \rightarrow u$ )

(R4:  $s \wedge r \rightarrow v$ )



➤  $CC = \{ \}$

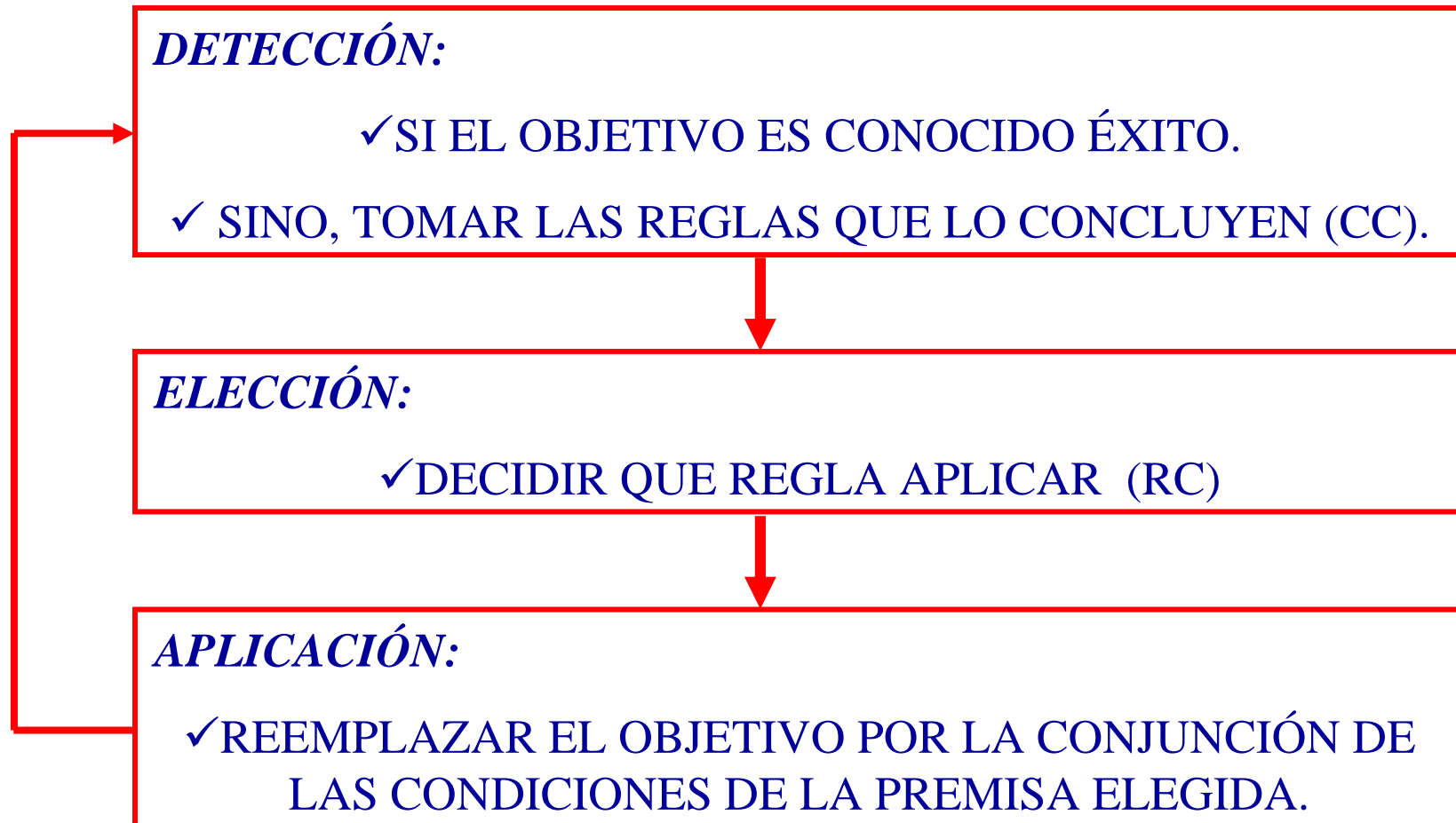
➤ FIN



BH: p q r s t u v

# SE basados en reglas de producción

Backward Chaining: Ciclo base de un motor inductivo.



# SE basados en reglas de producción

## Encadenamiento hacia atrás - Backward Chaining.

BH := CONOCIMIENTO INICIAL (HECHOS).

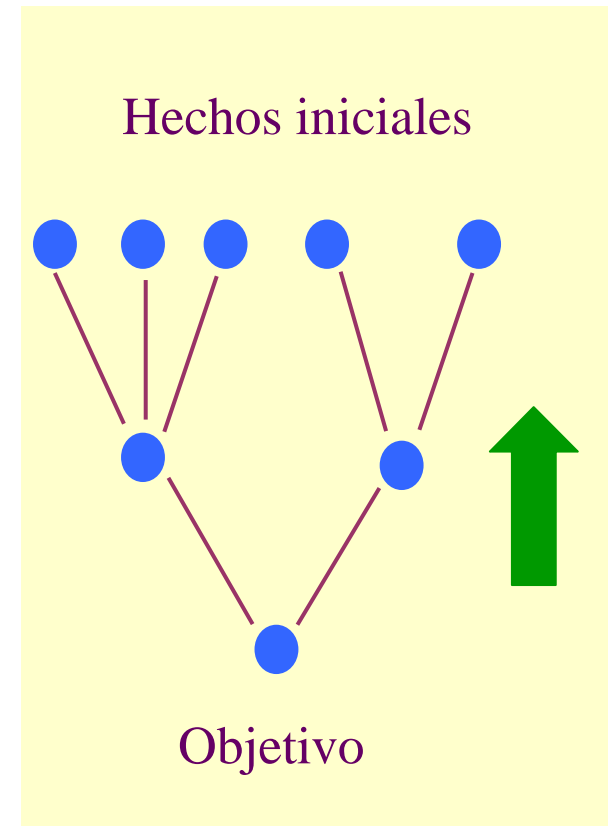
HASTA OBJETIVO O SIN REGLAS PARA  
DISPARAR.

REPITA

(1) ENCONTRAR K CONJUNTO DE REGLAS,  
CUYAS CONCLUSIONES PUEDEN  
UNIFICARSE CON LA HIPÓTESIS (CONJUNTO  
DE CONFLICTO).

(2) ELEGIR R DE K SEGÚN ESTRATEGIA DE  
SOLUCIÓN DE CONFLICTOS (POSIBLE  
BACKTRACKING).

(3) SI LA PREMISA DE R NO ESTÁ EN BH,  
TOMARLA COMO SUBOBJETIVO.





# SE basados en reglas de producción

## Ejemplo Backward Chaining: Primer paso

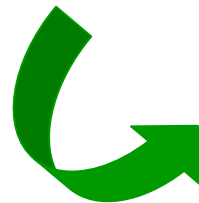
### BASE DE REGLAS

R1:  $p \wedge q \rightarrow s$

R2:  $r \rightarrow t$

R3:  $s \wedge t \rightarrow u$

R4:  $s \wedge r \rightarrow v$



➤ OBJETIVO:  $v$

➤  $v \in BH$  ?

➤  $v \notin BH$

➤ SIGUE



BH:  $p \ q \ r$

# SE basados en reglas de producción

## Ejemplo Backward Chaining: Segundo paso

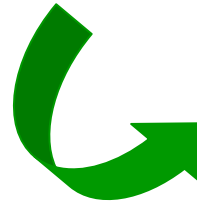
### BASE DE REGLAS

R1:  $p \wedge q \rightarrow s$

R2:  $r \rightarrow t$

R3:  $s \wedge t \rightarrow u$

R4:  $s \wedge r \rightarrow v$



BH:  $p \ q \ r$

➤ MATCHING CON  $v$

➤  $CC = \{ R4 \}$

➤  $s \in BH?$

➤  $s \notin BH$

➤  $s$  SUBOBJETIVO

➤ SIGUE

# SE basados en reglas de producción

## Ejemplo Backward Chaining: Tercer paso

### BASE DE REGLAS

R1:  $p \wedge q \rightarrow s$

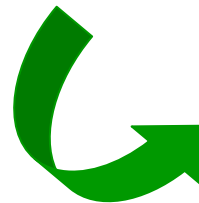
R2:  $r \rightarrow t$

R3:  $s \wedge t \rightarrow u$

R4:  $s \wedge r \rightarrow v$



BH: p q r



- MATCHING CON s
- $CC = \{ R1 \}$
- $p \in BH?$  SI.
- $q \in BH?$  SI.
- DISPARA R1.
- $BH \leftarrow s$

# SE basados en reglas de producción

Ejemplo Backward Chaining: 4º paso

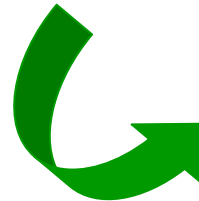
## BASE DE REGLAS

R1:  $p \wedge q \rightarrow s$

R2:  $r \rightarrow t$

R3:  $s \wedge t \rightarrow u$

R4:  $s \wedge r \rightarrow v$



➤  $CC = \{ R4 \}$

➤  $r \in BH?$  SI.

➤ DISPARA R4.

➤  $BH \leftarrow v$



BH: p q r s

# SE basados en reglas de producción

Ejemplo Backward Chaining: 5º paso

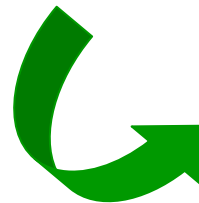
## BASE DE REGLAS

R1:  $p \wedge q \rightarrow s$

R2:  $r \rightarrow t$

R3:  $s \wedge t \rightarrow u$

R4:  $s \wedge r \rightarrow v$



➤ OBJETIVO OK.

➤ FIN

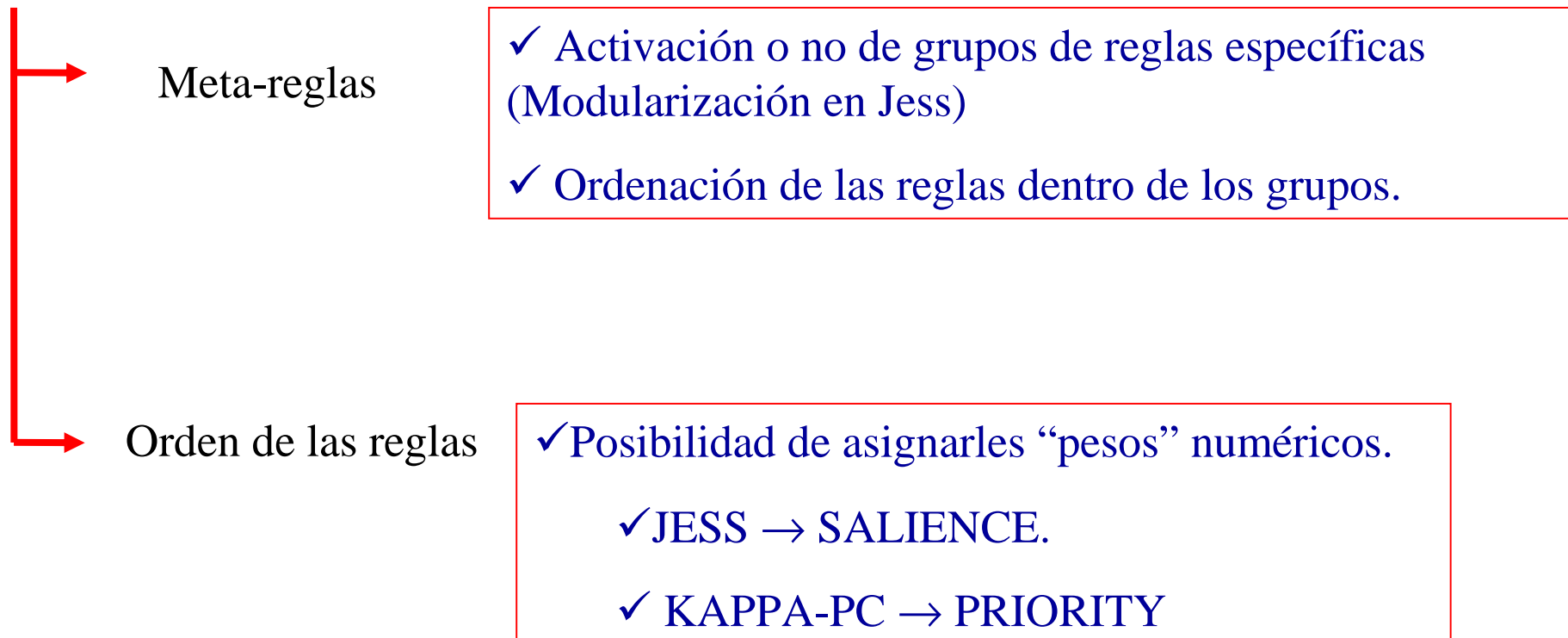


BH: p q r s v

# Sistemas expertos: Metaconocimiento

- ✓ Es el conocimiento estratégico vinculado a la utilización del conocimiento del dominio del Sistema Experto.
- ✓ Es conocimiento que predica sobre el conocimiento involucrado en el sistema.

## Métodos de implementación:



# Sistemas de producción

## *VENTAJAS*

- ⇒ Flexibles.
- ⇒ Sencillos de modificar y extender.
- ⇒ A los expertos les resulta simple “pensar en reglas”.

## *PROBLEMAS*

- ⇒ Completitud y consistencia.
- ⇒ El conocimiento se separa en pequeños “gránulos”.

Es común que se los combine con otros formalismos.

# Sistemas estructurados

✓ REDES SEMANTICAS (70s)

✓ OBJETOS (80s)

✓ ONTOLOGÍAS (90s)



## SE: Herramientas

Una *herramienta para la construcción de sistemas expertos* (expert system building tools-shell) es un software para el desarrollo de sistemas expertos,:

- un motor de inferencias,
  - forma/s de representar el conocimiento (BC vacía) y
  - una interfaz para el usuario final (opcional)
- permite construir sistemas expertos agregando el conocimiento específico sobre el dominio particular que se quiere resolver.

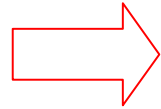
## SE: Herramientas

Permiten construir sistemas expertos agregando el conocimiento específico sobre el dominio particular que se quiere resolver.

- JESS (Java Expert System Shell- Lic. académica)
- CLIPS (C – NASA, software libre)
- DROOLS (gestión de reglas de negocios-algoritmo Rete)
- SOAR (KBS – MAS)
- KAPPA-PC (C - Intellicorp, software liberado)
- ...

# Herramientas

**JESS**



Java Expert System Shell



Formalismos de representación



Hechos y Reglas (Jess)  
Clases (Java) y reglas  
Ontologías y reglas



Funcionamiento del  
Motor de Inferencias



El funcionamiento sistemático  
está definido (Forward).  
Se puede forzar Backward

# ¿Qué es Jess ?

- JESS (Java Expert System Shell).

*<http://herzberg.ca.sandia.gov/jess/>  
([ejfried@ca.sandia.gov](mailto:ejfried@ca.sandia.gov))*

- Diseñado para integrar fácilmente Clips y Java.
- Originalmente (1995) era un clon de Clips, programado en Java.
- Hoy es mucho más. (versión actual: 7.1)
- Objetivo: Aplicaciones Java + razonamiento de un sistema experto (reglas+motor de inferencias). Integración Ontología + inferencia

# Características

- Utiliza el algoritmo de Rete para procesar las reglas.
- Puede obtener mayor rapidez que CLIPS.
- Incluye encadenamiento hacia atrás (5.0)
- Permite manipulación directa de objetos Java (también de objetos definidos en Protege)
- Incluye lógica difusa en la 5.0 (FuzzyJess):  
*[http://ai.iit.nrc.ca/IR\\_public/fuzzyJToolkit.html](http://ai.iit.nrc.ca/IR_public/fuzzyJToolkit.html)*

# Características

- Utiliza el algoritmo de Rete para procesar las reglas.
- Puede obtener mayor rapidez que CLIPS.
- Incluye encadenamiento hacia atrás (5.0)
- Permite manipulación directa de objetos Java (también de objetos definidos en Protege)
- Incluye lógica difusa en la 5.0 (FuzzyJess):  
*[http://ai.iit.nrc.ca/IR\\_public/fuzzyJToolkit.html](http://ai.iit.nrc.ca/IR_public/fuzzyJToolkit.html)*

# Bibliografía

- 1- Sistemas Expertos - Principios y Programación - Giarratano & Riley. 3ª Ed. Int. Thomson Ed, 2000.
- 2- Jess in Action Java Rule-based Systems- Ernest Friedman-Hill - Manning Publications Co. 2003.

# Problema

Se solicita desarrollar un sistema capaz de hacer un pronóstico del estado del tiempo, solicitando al usuario la menor cantidad de información necesaria. Para construir la Base de conocimientos se cuenta con la siguiente información:

Los estratos son nubes de baja altura con aspecto de capas. Los nimboestratos son nubes de altura media. Los cirroestratos son nubes de gran altura. Los cúmulos y cumulonimbos pueden estar a baja o gran altura y su aspecto es como pilas de algodón, pero los cumulonimbos son grises, mientras que los cúmulos son blancos.

Si no hay nubes puede inferirse que el tiempo será soleado. La presencia de cúmulos indica buen tiempo. Las nubes con estratos indican lluvia ligera pero si el viento sopla de noreste a sur la lluvia puede ser prolongada. Las nubes con nimboestratos indican lluvia breve si la dirección del viento es sur a norte, pero se indica lluvia persistente si la dirección del viento es de noreste a sur. Los cumulonimbos indican chubascos. Los cirroestratos indican lluvia en las 24 hs. si el viento es norte a sur.



# Problema

Se solicita:

- Representar el dominio del problema y estructurarlo, utilizando hechos estructurados (clases de objetos). De forma que se puedan representar todos los elementos involucrados y sus características.
- Obtener del texto un grupo de reglas implícitas y formalizarlas de acuerdo a la representación del dominio.
- Implementar un mini sistema experto, que a partir del estado actual del cielo, vientos, etc. pueda dar un pronóstico.