

# Ejercicios de Evaluación Bloques 1 y 2

Jose Antonio Lorencio Abril

Noviembre 2021

## Contents

<b>1 Cuestión 1.</b>	<b>2</b>
<b>2 Cuestión 2</b>	<b>3</b>
<b>3 Cuestión 3</b>	<b>4</b>
<b>4 Cuestión 4</b>	<b>7</b>
<b>5 Cuestión 5</b>	<b>10</b>
<b>6 Cuestión 6</b>	<b>14</b>
<b>7 Cuestión 7</b>	<b>16</b>
<b>References</b>	<b>18</b>

## 1 Cuestión 1.

En función de su accesibilidad, el conocimiento puede ser dividido entre **Conocimiento Tácito** y **Conocimiento Explícito**. El alumno deberá definir cada uno de los tipos anteriores, elegir un dominio de conocimiento cualquiera y describir algún ejemplo de cada uno de los tipos anteriores.

El **conocimiento tácito** ([3]) es aquel que:

- No es articulable: no puede describirse en términos proposicionales (enunciados verbales); o
- Es implícito: es articulable, pero su descripción entraña dificultades

Normalmente es adquirido a través de la experiencia. Además, dado que no puede observarse desde el exterior, e incluso es posible que ni siquiera el poseedor del conocimiento tácito sea capaz de identificarlo, se ve como un conocimiento de difícil identificación y medida.

Como ejemplo, podemos pensar en algún momento de nuestras vidas en el que hayamos participado en juegos en equipo, sobre todo durante un largo período de tiempo. Centrándome en fútbol, que practiqué mucho en la adolescencia, un efecto que se produce cuando juegas mucho con la misma gente es que, en función de la situación del partido y de la jugada, sabes aproximadamente donde está cada compañero, si va a desmarcarse e incluso si debes intentar pasarle el balón o no. Este conocimiento ha sido adquirido por la sucesión de muchas jugadas similares. Muchas veces es posible dar un buen pase sin mirar, y es muy difícil describir qué te hizo llegar a la conclusión de que era correcto. De igual forma, este conocimiento implícito, en ocasiones proporciona malos resultados, cuando lo que crees que va a hacer tu compañero difiere de lo que realmente hace. Esta situación errónea, de alguna forma, contribuye a tu 'base de datos' personal y será menos probable que vuelva a ocurrir el mismo malentendido. Todo esto sucede de forma subconsciente, realmente no estás pensando activamente en qué hace cada compañero, simplemente lo 'sabes'.

Por otro lado, el **conocimiento explícito** ([2]) es aquel cuyo poseedor puede explicitar por medio de enunciados verbales. Es de esta forma que el conocimiento explícito puede en ocasiones definirse como aquel que no es explícito. No obstante, estas definiciones tienen el problema de la subjetividad, dado que la facilidad e incluso la posibilidad de verbalización del conocimiento, depende de la persona que lo posea.

Completando el ejemplo anterior, podríamos hacer que ese proceso explicado de inferencia que se hace de forma implícita, como es el caso de una jugada ensayada, que ha sido planeada y practicada, cada uno sabe y comprende su función en la jugada. En esa situación, sabes por qué sabes dónde están tu compañeros.

## 2 Cuestión 2

Responde brevemente a las siguientes preguntas:

1. Supón que hemos definido una ontología para representar todos los conceptos relacionados con un mundial de baloncesto y que dicha ontología ha sido desarrollada en Protégé. ¿Cómo podemos evitar que se pueda declarar que un jugador pueda ser incluido en la plantilla de dos equipos diferentes?

Para responder tanto esta como las siguientes preguntas voy a usar la información de [8].

Podemos introducir la siguiente restricción:

```
(forall ?jugador
  (<(number-of-slot-values juega_en ?equipo) 2)
)
```

Otra forma sería simplemente establecer en el slot juega\_en una cardinalidad máxima de 1.

2. Supongamos que queremos realizar una ontología sobre coches. En dicha ontología se representas los distintos tipos de coches, que en nuestro caso serán; sedán, monovolumen, SUV y deportivo, como subclases de coche. ¿Es correcto definir una relación en dicha ontología que tenga como dominio las clases sedán, monovolumen, SUV y coche? Justifica tu respuesta.

No es correcto, puesto que si coche está en el dominio, entonces lo heredarán todas sus subclases. Entonces, dependiendo de nuestra intención, hay dos formas de hacer esto correctamente:

- Si queremos que todas las subclases de coche estén en el dominio de la relación, entonces únicamente pondremos a coche en el dominio
  - Si, por el contrario, solo queremos que algunas subclases estén en el dominio de la relación, entonces no pondremos a coche en el dominio, en el que únicamente estarán las subclases que deseamos que estén en el dominio de la relación
3. Supón que hemos definido una ontología para representar todos los conceptos relacionados con un mundial de baloncesto y que dicha ontología ha sido desarrollada en Protégé. Si se definen clases que indican en qué posición debe jugar cada jugador (base, escolta, alero o pivot), ¿cómo se podría permitir desde la ontología declarar jugadores que puedan jugar en varias posiciones?

Es como el apartado 1, puede hacerse estableciendo una restricción o simplemente indicando una cardinalidad 1 a n.

### 3 Cuestión 3

En clase se han mencionado distintas librerías de ontologías de libre acceso. El alumno deberá escoger una ontología y poner un ejemplo de cada uno de los elementos constitutivos de las ontologías (clases, propiedades, relaciones, ...). Se deberá escoger una ontología lo más completa posible en función de los elementos constitutivos que utiliza.

He elegido una ontología de la Protégé Ontology Library ([4]), concretamente una ontología sobre una biblioteca ([1]). Observamos los siguientes elementos:

- Clases: son los elementos fundamentales de una ontología, y son una descripción formal de una entidad del dominio que se quiere modelar. Las clases pueden tener subclases, que concreten el concepto. En nuestro caso, podemos ver en la siguiente figura 1 la clase LibraryMember, junto con las subclases de la misma.

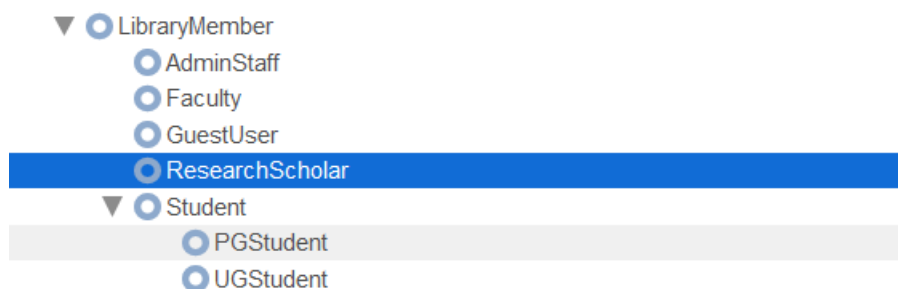


Figure 1: Ejemplos de clases de una ontología

- Clases abstractas: son clases de las que no se pueden crear instancias. Deben ser concretizadas en subclases que no sean abstractas. En el ejemplo anterior, LibraryMember es una clase abstracta, al igual que lo es Student.
- Instancias: representan objetos concretos del dominio pertenecientes a una clase. Por ejemplo, encontramos a Rajini, que es una instancia de un alumno de la clase UGStudent:

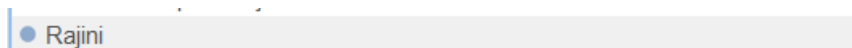


Figure 2: Ejemplo de una instancia

- Propiedades (atributos o slots): representan características utilizadas para incrementar el detalle en la descripción de una clase y sus instancias, pero no tienen suficiente fuerza conceptual par representar una clase por sí mismas. Siguiendo nuestro ejemplo, encontramos la propiedad isBorrowedBy, con Dominio en Book y con Rango en Student:

Property: isBorrowedBy

IRI  
http://www.LibrarySystem.com/ontologies/LMS.owl#isBorrowedBy

**Annotations**

Enter property      Enter value      lang

**Domain**

☐ UGTextBook

☐ PGTextBook

Enter a class name

**Range**

☐ UGStudent

☐ PGStudent

Enter a class name

Figure 3: Ejemplo propiedad

- Relaciones: representan asociaciones entre clases de un dominio. En nuestro ejemplo, la clase ResearchScholar está relacionada con la clase OnlineJournal mediante la relación Browse:

Class Hierarchy

- owl:Thing
  - LibraryMember
    - AdminStaff
    - Faculty
    - GuestUser
    - ResearchScholar**
    - Student
      - PGStudent
      - UGStudent
    - LibraryPersonnel
      - AsstLibrarian
      - JrAssistant
      - Librarian
      - Technician
    - LibraryResource
      - Book
      - CD
      - Journal
      - NewsPaper
      - OnlineJournal**
      - Thesis
    - LibraryService

Class: ResearchScholar

IRI  
http://www.LibrarySystem.com/ontologies/LMS.owl#ResearchScholar

**Annotations**

Enter property      Enter value      lang

**Parents**

☐ LibraryMember

Enter a class name

**Relationships**

☒ Browse ☐ OnlineJournal

☐ Refer ☐ Thesis

Enter property      Enter value      lang

Figure 4: Ejemplo relación

- Funciones: son relaciones en las que uno de los elementos es único, dados los demás elementos. En el caso de esta ontología, vemos, por ejemplo, la relación Use, que identifica a un LibraryMember que usa un InternetAndWebService:

OWL Entity Description Editor: Use	
1	<b>ObjectProperty:</b> Use
2	
3	<b>Domain:</b> [in root-ontology]
4	LibraryMember
5	
6	<b>Range:</b> [in root-ontology]
7	InternetAndWiFiService
8	
9	<b>InverseOf:</b> [in root-ontology]
10	isUsedBy
11	
12	
13	

Figure 5: Ejemplo función

- Restricción: definen propiedades de las relaciones y los conceptos. Por ejemplo, en la figura anterior, encontramos la restricción InverseOf. Esto quiere decir que si X Use Y, entonces Y isUsedBy X. Es una restricción sobre ambas relaciones. Esta restricción además es la que nos asegura que Use es un ejemplo de función.
- Axiomas: permiten modelar sentencias que siempre son ciertas y que no pueden ser modeladas con todos los elementos anteriores. Para necesitar axiomas es probable que la ontología deba tener cierto grado de complejidad, y la ontología de la librería no tiene axiomas, hasta donde yo he visto.

## 4 Cuestión 4

En la figura 6 se muestra un ejemplo de una ontología sobre coches para la que se han definido los siguientes axiomas:

$$\exists x, (coche(x) \vee \neg coche(x)) \quad (1)$$

$$\forall x, (monovolumen(x) \rightarrow \exists y, (float(y) \wedge coche(x) \wedge longitud(x, y) \wedge y > 4,5)) \quad (2)$$

$$\exists x, (monovolumen(x) \wedge longitud(x, 5)) \quad (3)$$

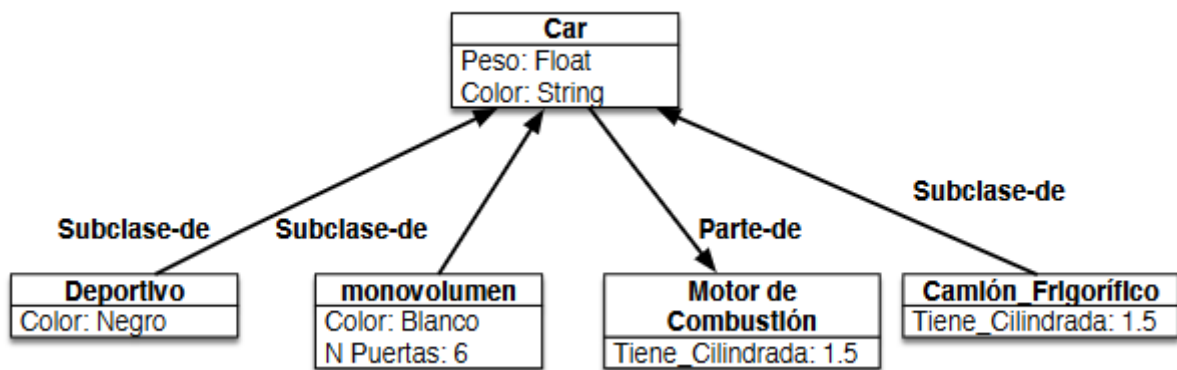


Figure 6: Ejemplo de la ontología sobre coches

Responder:

1. Enumera los posibles errores en el diseño de la ontología e indica cómo los corregirías

- Tal y como está el diagrama, un Coche es una parte de un Motor de Combustión, tiene mucho más sentido que sea al revés. Para arreglarlo basta cambiar el sentido de la flecha. Ahora todas las subclases de coche comparten esta relación por herencia.
- Camión\_Frigorífico tiene la propiedad *Tiene\_Cilindrada*, que está en Motor de Combustión, y que tras el cambio anterior todas comparten. Esta propiedad la eliminaríamos de Camión\_Frigorífico. Además, Motor de Combustión le asigna un valor fijo a esta variable, sería más correcto indicar que es un float, y luego se asigne el valor correspondiente en las subclases.
- La clase Monovolumen tiene la propiedad *N Puertas*, que no está definida en ninguna de las demás subclases, ni en la superclase. Esta propiedad, por tanto, no es muy útil tal y como se está usando. Probablemente sería mejor pasarla a la clase Coche, y que cada subclase le asignase un valor concreto correspondiente.
- Presenta varios errores de notación, pues se hace uso de distintos idiomas y no se nombran los elementos de forma uniforme en cuanto al manejo de minúsculas y mayúsculas, o los espacios. Arreglarlo es sencillo, basta unificar la notación, por ejemplo escribiendo todo en español, empezando por mayúscula y escribiendo los espacios como ' \_ '.
- El tipo de peso es probable que no tenga mucho sentido, ya que los coches pesan bastante y no se va a dar uso de los decimales. Es mejor ponerlo como entero.

- Las subclases no deberían definir el color, ya que un mismo modelo de coche puede estar disponible en varios colores, esto debe hacerse en las instancias.
- Según los axiomas, deberíamos definir el atributo longitud, como float.

Una vez identificados estos fallos, los corregimos y la ontología queda como en la figura 7:

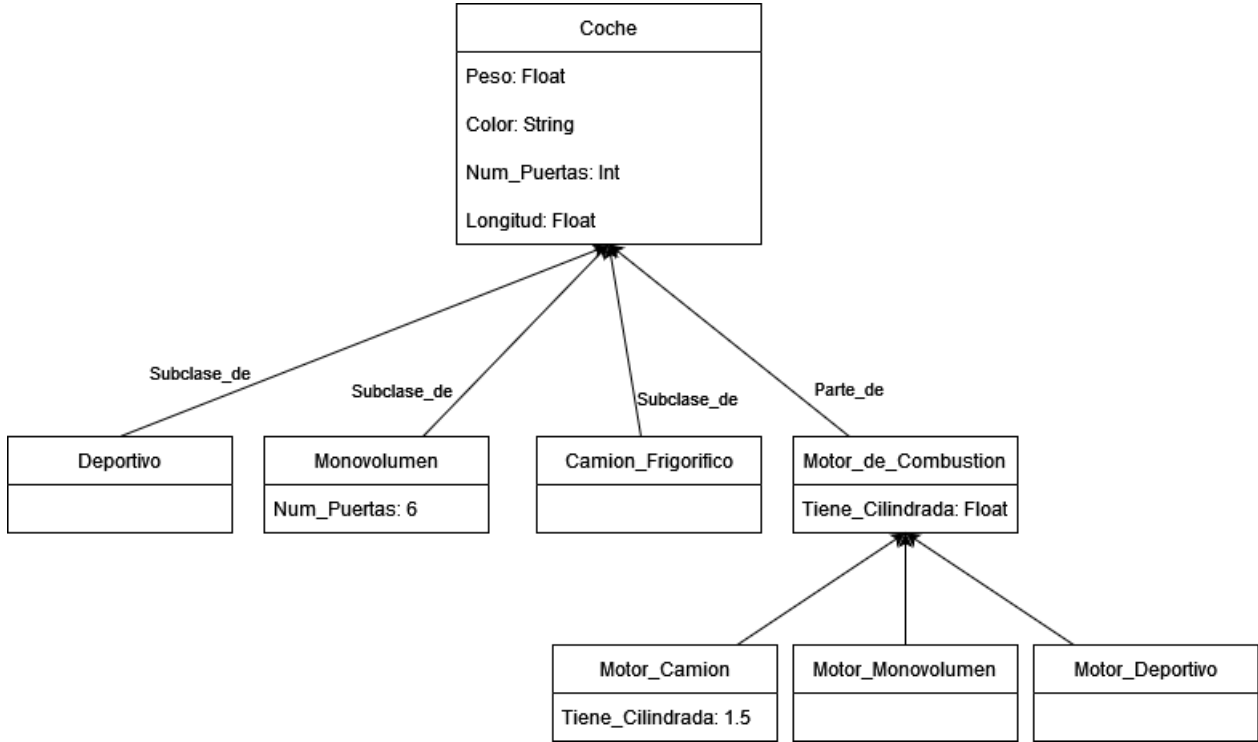


Figure 7: Ontología sobre coches corregida

En los siguientes apartados trabajaremos sobre esta ontología.

- ¿Qué dirías sobre la corrección de los axiomas de la antología? Añade dos axiomas más que sean consistentes con la ontología.
  - $\exists x, (coche(x) \vee \neg coche(x))$ : esto es una tautología, por lo que su inclusión como axioma es innecesario.
  - $\forall x, (monovolumen(x) \rightarrow \exists y, (float(y) \wedge coche(x) \wedge longitud(x,y) \wedge y > 4,5))$ : este axioma es casi correcto, establece una restricción sobre la longitud de los monovolúmenes. Sobra comprobar  $coche(x)$  ya que si  $x$  es un monovolumen, entonces  $x$  es coche por la herencia.
  - $\exists x, (monovolumen(x) \wedge longitud(x,5))$ : este axioma es correcto, aunque es un poco extraño pedir que existe un monovolumen con una longitud concreta dentro de la ontología.

Y podemos añadir los siguientes axiomas adicionales:

- $\forall x, (coche(x) \rightarrow \exists y, (Motor\_de\_Combustion(y) \wedge Parte\_de(x,y)))$ : este axioma indica que para todo  $x$  coche, existe un motor de combustión  $y$  que es parte de  $x$ .
- $\forall x, (coche(x) \rightarrow \exists y, (integer(y) \wedge Num\_Puertas(x,y) \wedge y < 6))$ : los deportivos tienen menos de 6 puertas.



3. ¿Cómo se podría corregir y extender la ontología?

Parece un poco extraño identificar un camión como un tipo de coche, por lo que parece adecuado añadir una clase Vehículo, de la que hereden Coche, Camión y los demás vehículos que deseemos contemplar en la ontología. El resultado final puede verse en la figura 8.

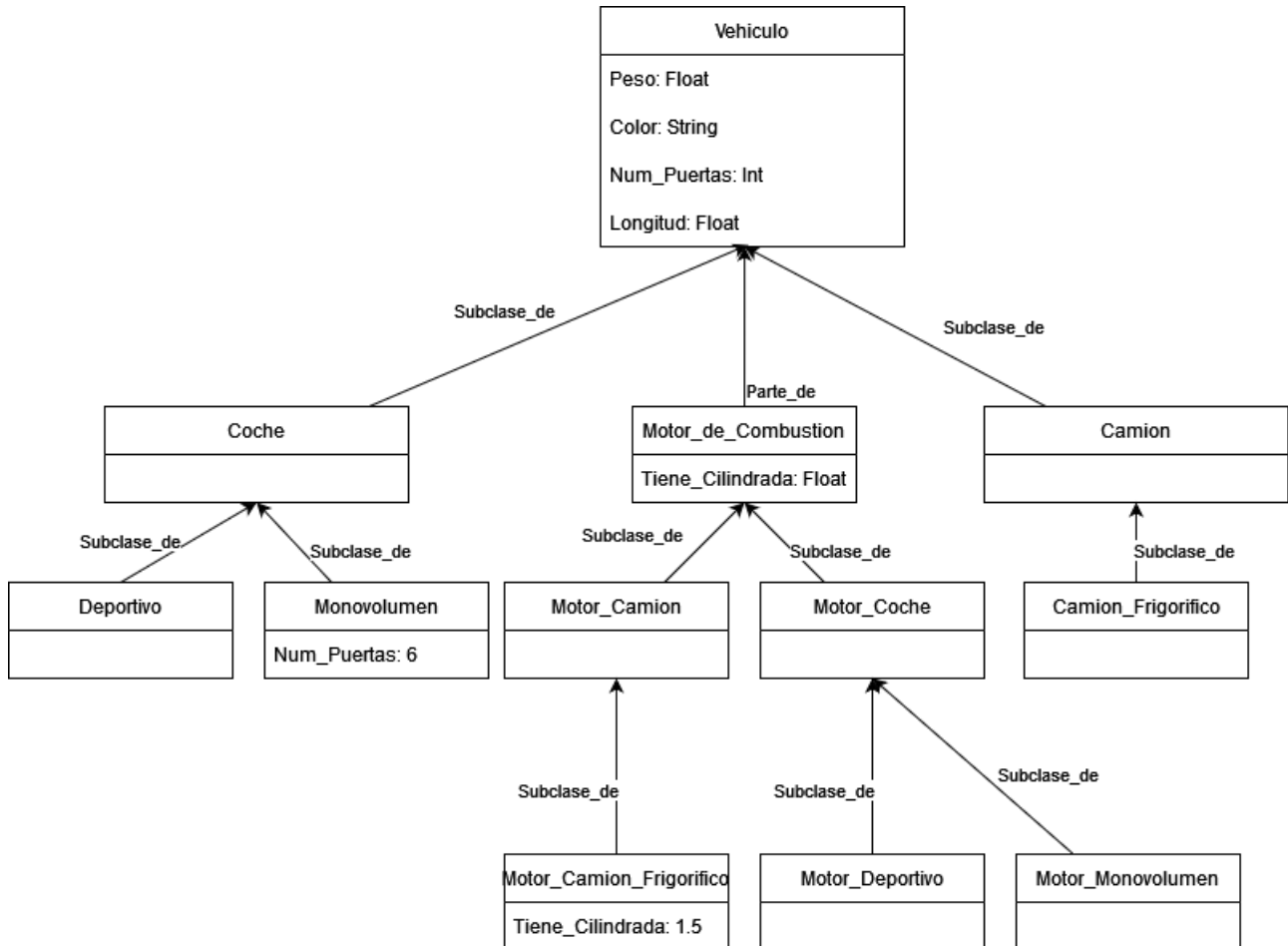


Figure 8: Ontología sobre vehículos resultante tras corregir y extender la ontología inicial

## 5 Cuestión 5

Desarrollar los ciclos de funcionamiento de un SBR, utilizando su red RETE, que tiene el siguiente conjunto de reglas:

- R1:  $(A \ B) \wedge (C \ ?x) \rightarrow \text{Añadir}((I \ ?x)) \wedge \text{Añadir}((D \ ?x))$
- R2:  $(C \ ?x) \wedge (E \ B) \rightarrow \text{Suprimir}((A \ B)) \wedge \text{Añadir}((I \ ?x))$
- R3:  $(A \ B) \wedge (C \ ?x) \wedge (D \ ?y) \rightarrow \text{Suprimir}((D \ ?y)) \wedge \text{Suprimir}((E \ B))$
- R4:  $(G \ K) \wedge (H \ J) \wedge (L \ M) \rightarrow \text{Suprimir}((H \ J))$
- R5:  $(H \ J) \wedge (D \ ?x) \wedge (I \ ?z) \rightarrow \text{Suprimir}((H \ J)) \wedge \text{Suprimir}((C \ ?x))$

1. Construir la red RETE que lo representa

Basándome en lo explicado en teoría ([6]), he construido la siguiente red RETE:

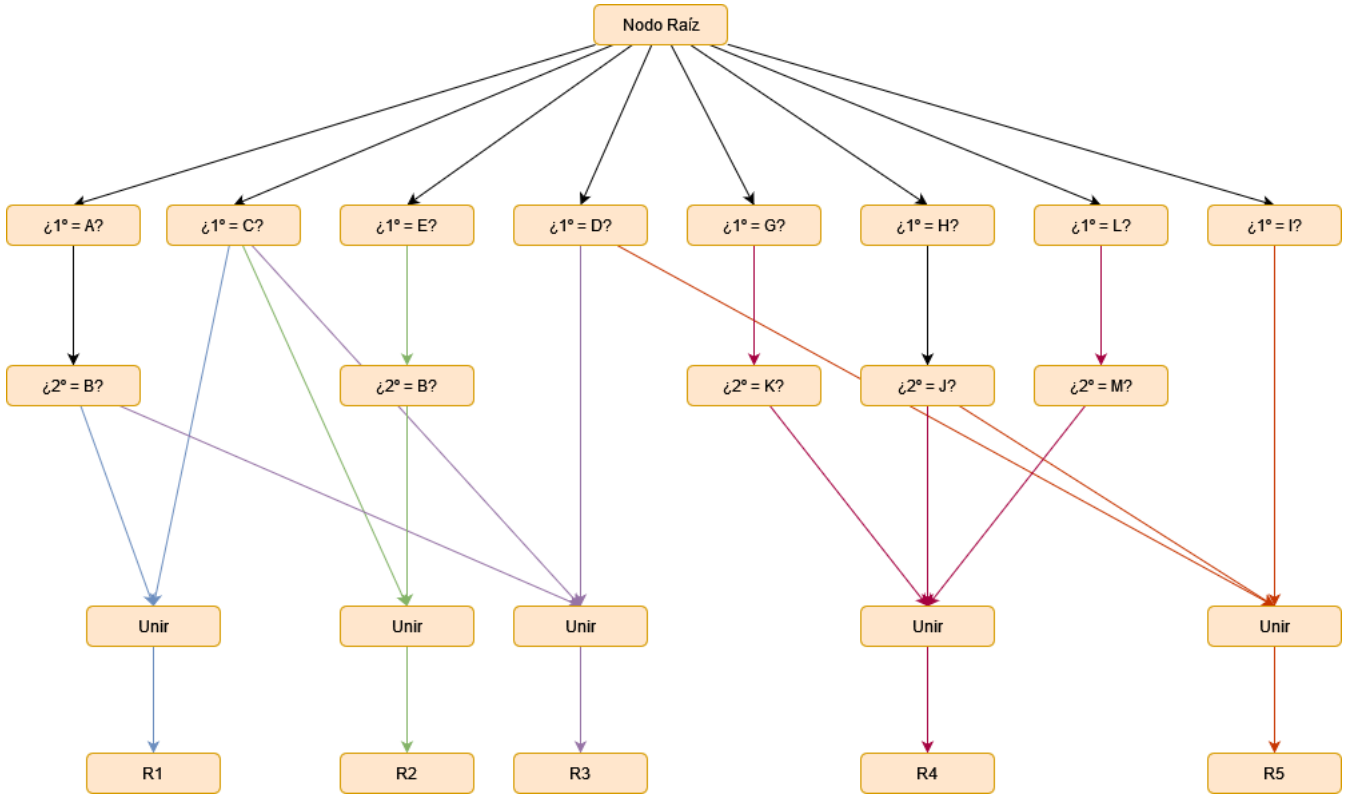


Figure 9: Red RETE para el ejercicio 5

2. ¿Cuál será el conjunto de conflicto (CC) resultante para la siguiente base de hechos inicial?

$$\begin{aligned}
 h_1 &= (10 \ (A \ B)) & h_2 &= (11 \ (C \ D)) & h_3 &= (14 \ (C \ E)) \\
 h_4 &= (8 \ (E \ B)) & h_5 &= (15 \ (D \ F)) & h_6 &= (5 \ (G \ K)) \\
 h_7 &= (7 \ (H \ J)) & h_8 &= (9 \ (L \ M)) & h_9 &= (13 \ (I \ E))
 \end{aligned}$$

Al añadir los hechos, la red RETE anterior se transforma en la siguiente:

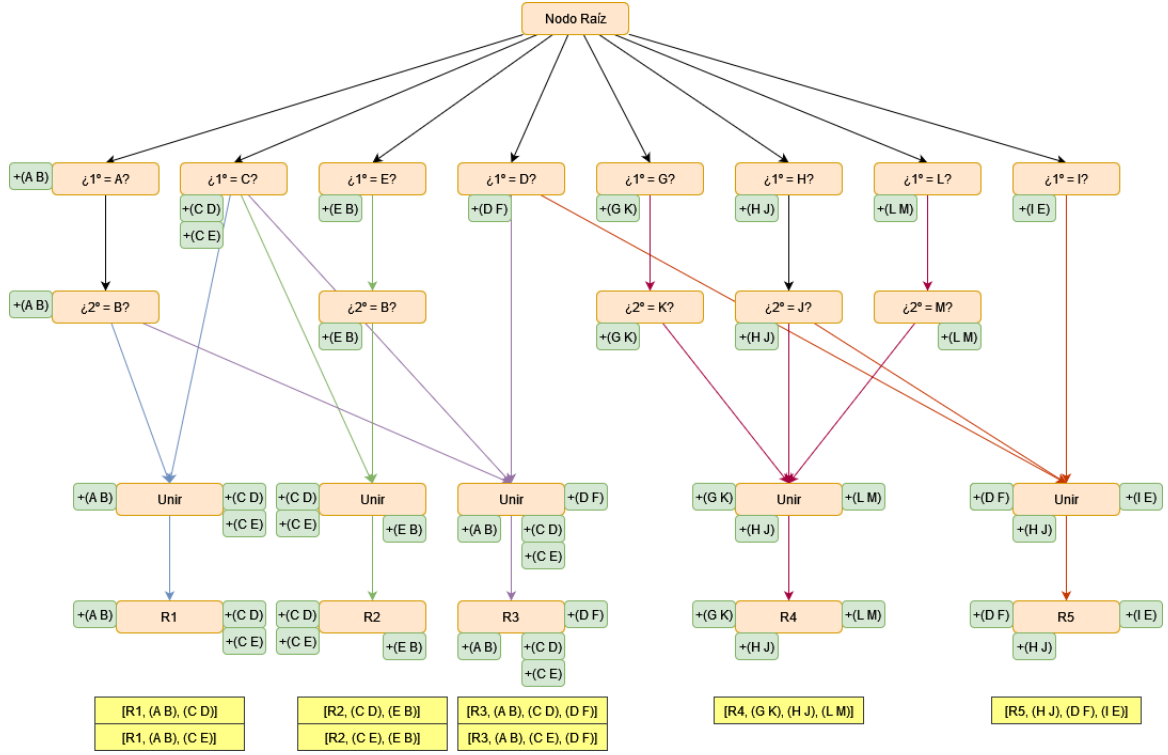


Figure 10: Red RETE tras añadir los hechos

3. A partir de dicho CC y considerando que estamos en el ciclo 16, desarrollar los siguientes ciclos de ejecución, partiendo de la hipótesis de que no se ha ejecutado ninguna instanciación del conjunto de conflictos y teniendo en cuenta la siguiente regla para la resolución de conflictos:

$$(PR \bullet E) \rightarrow (P \bullet A) \rightarrow AB$$

Entendiendo la antigüedad como el elemento más antiguo, y definiéndose la prioridad entre las reglas como:

- $R2, R3 > R4$
- $R4, R5 > R1$

Partimos el conjunto conflicto inicial

$$CC_0 = (R1, R2, R3, R4, R5)$$

Según el enunciado del ejercicio, no se ha instanciado ninguna regla aún, por tanto, el principio de refracción no elimina ninguna regla del conjunto de conflicto, por tanto

$$CC_{PR} = CC_0$$

Al aplicar la especificidad, obtenemos

$$CC_E = (R2, R3, R4, R5)$$

Por tanto

$$CC_1 = CC_{PR} \cap CC_E = CC_E = (R2, R3, R4, R5)$$

Aplicamos la prioridad dada en el enunciado, y queda

$$CC_P = (R_{21}, R_{22}, R_{31}, R_{32}, R_5)$$

y por la antigüedad, dado que es  $A(R_{21}) = 16 - 8 = 8$ ,  $A(R_{22}) = 8$ ,  $A(R_{31}) = 6$ ,  $A(R_{32}) = 6$ ,  $A(R_4) = 11$ ,  $A(R_5) = 9$ , por tanto

$$CC_A = (R_{31}, R_{32})$$

y entonces

$$CC_2 = (R_{31}, R_{32})$$

Por último, aplicamos el criterio de arbitrariedad y elegimos, por ejemplo  $R_{31}$ . De modo que

$$CC = (R_{31})$$

como  $|CC| = 1$ , se detiene el algoritmo de equiparación y se dispara esta única regla, R3:  $(A \ B) \wedge (C \ ?x) \wedge (D \ ?y) \rightarrow \text{Suprimir}((D \ ?y)) \wedge \text{Suprimir}((E \ B))$ . Entonces, se introducen los nuevos hechos  $(- (17 \ (D \ F)))$  y  $(- (17 \ (E \ B)))$ . Así, el nuevo árbol RETE queda:

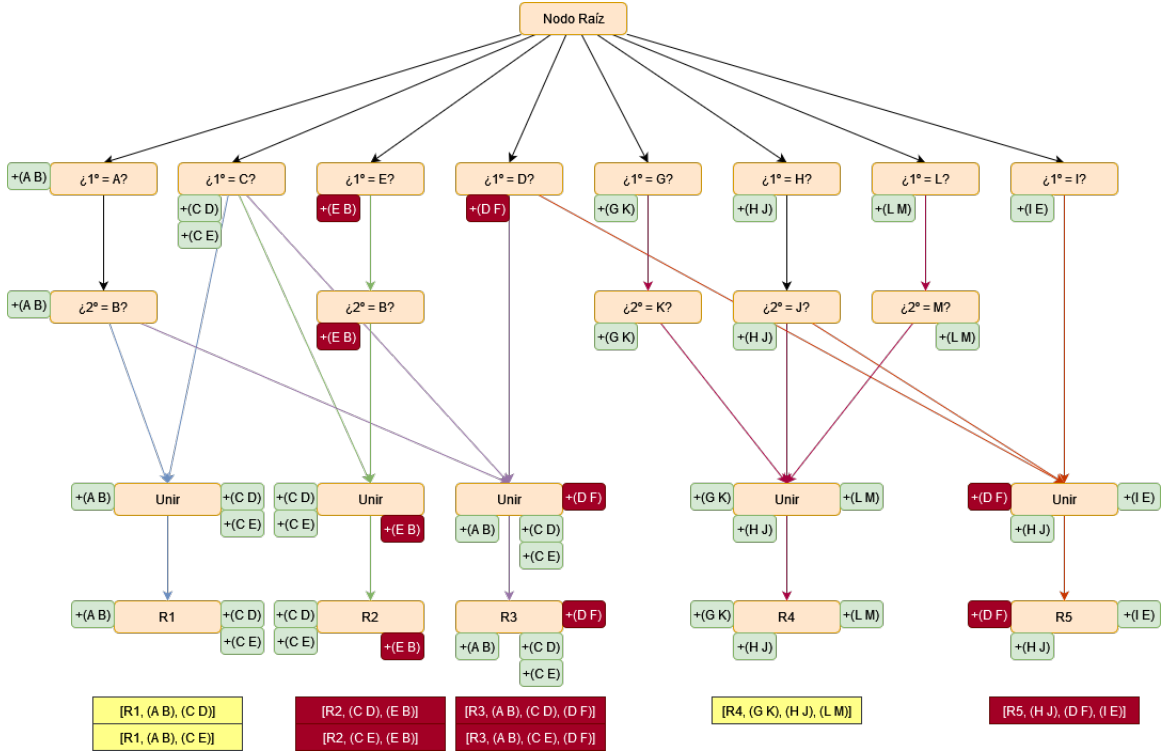


Figure 11: Red RETE al aplicar  $R_{31}$ . Como solo se ejecutan supresiones, lo único que se hace es eliminar los hechos marcados por la regla, y todo lo que arrastren. Marco en granate todo lo que quedaría eliminado

Al eliminarlos definitivamente vemos la siguiente imagen:

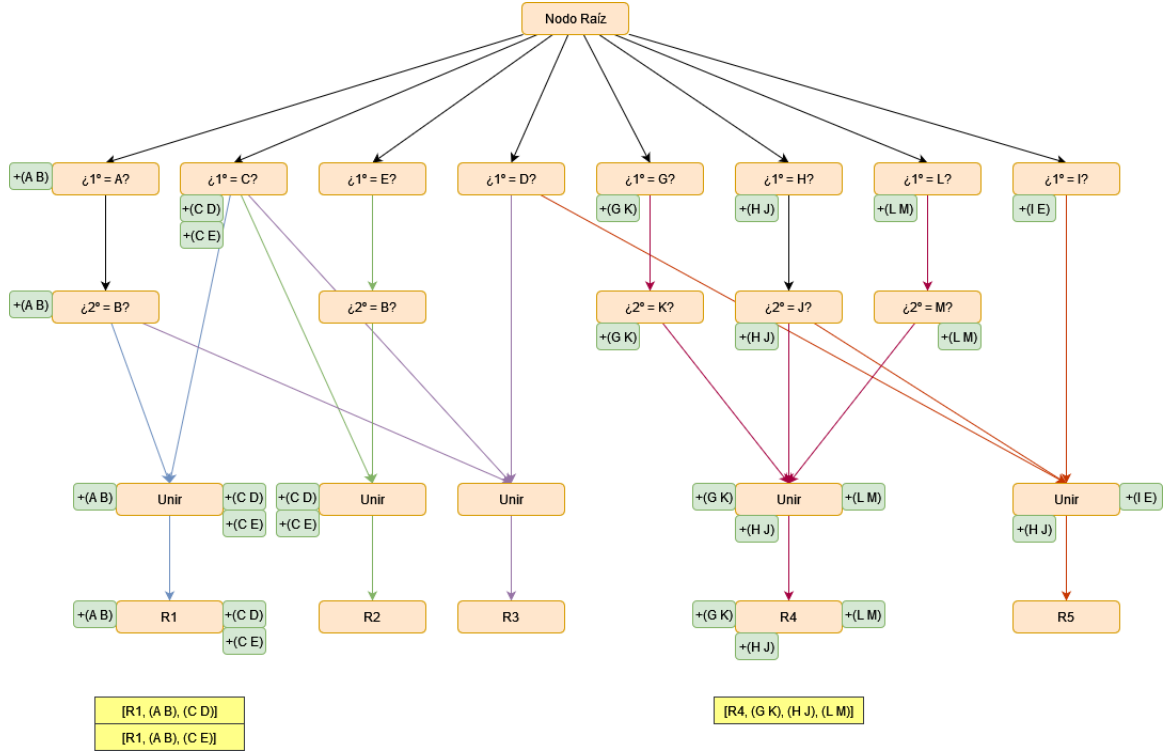


Figure 12: Red RETE al eliminar los elementos marcados en granate en 11

Y tenemos

$$CC_0 = (R1_1, R1_2, R4)$$

Por el principio de refracción, como ninguna de estas reglas ha sido ejecutada antes, no se cambia nada:

$$CC_{PR} = (R1_1, R1_2, R4)$$

y por especificidad:

$$CC_E = (R1_1, R1_2, R4)$$

por lo que

$$CC_1 = (R1_1, R1_2, R4)$$

Por la prioridad es

$$CC_P = (R4)$$

y por la antigüedad, como  $A(R1_1) = 18 - 10 = 8$ ,  $A(R1_2) = 8$ ,  $A(R4) = 11$  es

$$CC_A = (R1_1, R1_2)$$

y así

$$CC_2 = \emptyset$$

y como  $|CC| = 0$ , se detiene el algoritmo.

## 6 Cuestión 6

Considérese las siguientes reglas de MYCIN:

R1: SI:

- (a) el cultivo se localiza en la sangre, y
- (b) la identidad del organismo no se conoce con total certeza (función NOTDEFINITE), y
- (c) la tinción del organismo es gramnegativa, y
- (d) la morfología del organismo es de tipo bastoncillo, y
- (e) el paciente presenta quemaduras serias,

ENTONCES: Existe una débil evidencia (.5) de que el organismo es un pseudomonas.

R2: SI:

- (a) El cultivo se localiza en la sangre, y
- (b) el paciente presenta eczemas gangrenosos en la piel,

ENTONCES: Existen pruebas que sugieren (.4) que el organismo es un pseudomonas

R4: SI:

- (a) Si la identidad del organismo no es conocida con total certeza (función NOTDEFIS), y
- (b) si la tinción del cultivo es gramnegativa, y
- (c) la morfología del organismo es de tipo bastoncillo, y
- (d) el organismo es aerobio

ENTONCES: Existe una fuerte evidencia (.7) que el organismo es una enterobacteria.

R3: SI:

- (a) Si la tinción del cultivo es gramnegativa, y
- (b) la morfología del organismo es de tipo bastoncillo, y
- (c) el paciente ha estado expuesto a contagio

ENTONCES: Existen pruebas que sugieren (.7) que el organismo es un pseudomonas

y que la base de hechos contiene las siguientes evidencias:

- $E_1$  = El paciente presenta quemaduras serias,  $CF = 1$
- $E_2$  = La morfología es de tipo bastoncillo con  $CF = 0.4$
- $E_3$  = El cultivo se localiza en la sangre,  $CF = 0.3$
- $E_4$  = El paciente presenta eczemas gangrenosos en la piel con  $CF = 1$
- $E_5$  = La tinción es gramnegativa,  $CF = 0.5$
- $E_6$  = El paciente ha estado expuesto a contagio,  $CF = 0.4$
- $E_7$  = El paciente ha estado expuesto a contagio,  $CF = 0.8$

¿Con qué factor de certeza podemos afirmar que el organismo es un pseudomonas o una enterobacteria? Tal y como hemos visto en teoría ([5]), vamos a calcular los factores de certeza, a partir de los hechos suministrados.

Primero, vemos que  $E_6$  y  $E_7$  proporcionan un factor de certeza sobre la misma conclusión, por tanto, podemos combinarlos en  $E_8$  = El paciente ha estado expuesto a contagio,  $CF = CF_6 + CF_7 - CF_6 CF_7 = 0.88$ .

Además, como la identidad del organismo es totalmente desconocida ( $FC = 0$ ), entonces  $NOTDEFINITE = TRUE$ . Así:

Respecto a la regla R1 tenemos

$$R1 : (E_2 \ 0.3) \wedge NOTDEFINITE \wedge (E_5 \ 0.5) \wedge (E_2 \ 0.4) \wedge (E_1 \ 1) \quad \text{ENTONCES } (Pseudomonas \ 0.5)$$

como  $NOTDEFINITE = TRUE$ , entonces es

$$R1 : (E_2 \ 0.3) \wedge (E_5 \ 0.5) \wedge (E_2 \ 0.4) \wedge (E_1 \ 1) \quad \text{ENTONCES } (Pseudomonas \ 0.5)$$

y entonces

$$CF_{pseudomonas, R1} = \min \{0.3, 0.5, 0.4, 1\} \cdot 0.5 = 0.3 \cdot 0.5 = 0.15$$

La regla R2 se dispara con los hechos:

$$R2 : (E_2 \ 0.3) \wedge (E_4 \ 1) \quad \text{ENTONCES } (Pseudomonas \ 0.4)$$

Para la conjunción es  $CF_{\wedge} = \min \{0.3, 1\} = 0.3$  y el de la conclusión

$$CF_{pseudomonas, R2} = CF_{\wedge} \cdot CF_2 = 0.3 \cdot 0.4 = 0.12$$

La regla R3 se dispara con los hechos:

$$R3 : (E_5 \ 0.5) \wedge (E_2 \ 0.4) \wedge (E_8 \ 0.88) \quad \text{ENTONCES } (Pseudomonas \ 0.7)$$

Para la conjunción es  $CF_{\wedge} = \min \{0.5, 0.4, 0.88\} = 0.4$ , por lo que el de la conclusión es

$$CF_{pseudomonas, R3} = CF_{\wedge} \cdot CF_3 = 0.4 \cdot 0.7 = 0.28$$

Ya tenemos toda la evidencia acerca de que el organismo es pseudomonas, falta combinarla. El organismo es **un pseudomonas con un factor de certeza** de

$$\begin{aligned} FC_{pseudomonas} &= FC_{pseudomonas, R1} + FC_{pseudomonas, R2} + FC_{pseudomonas, R3} \\ &- FC_{pseudomonas, R1} \cdot FC_{pseudomonas, R2} - FC_{pseudomonas, R1} \cdot FC_{pseudomonas, R3} - FC_{pseudomonas, R2} \cdot FC_{pseudomonas, R3} \\ &+ FC_{pseudomonas, R1} \cdot FC_{pseudomonas, R2} \cdot FC_{pseudomonas, R3} = \\ &= 0.15 + 0.12 + 0.28 - 0.15 \cdot 0.12 - 0.15 \cdot 0.28 - 0.28 \cdot 0.12 + 0.15 \cdot 0.12 \cdot 0.28 = 0.46144 \end{aligned}$$

Por último, para la regla R4, dado que no tenemos información sobre la identidad del organismo, es  $NOTDEFIS = NIL$ , por lo que la regla no se evalúa y es  $FC_{enterobacteria} = FC_{enterobacteria, R4} = 0$ . Por tanto, el factor de certeza que pide el enunciado es:

$$CF_{pseudomonas \vee enterobacteria} = 0.46144 + 0 = 0.46144$$

## 7 Cuestión 7

Una de las herramientas más conocidas para el desarrollo de sistemas expertos es CLIPS (C Language Integrated Production Systems).

1. ¿Cómo es la estrategia para la resolución de conflictos de CLIPS? Describe el proceso y cada una de las técnicas utilizadas.

A la hora de hacer funcionar un SBR, existe la posibilidad de que se cumplan los antecedentes de distintas reglas. El conjunto de reglas que pueden ser ejecutadas, dados los hechos actuales, se denomina **conjunto conflictivo**, y es relevante diseñar una buena forma de elección de la siguiente regla a ejecutar.

Según el Manual de Referencia de CLIPS ([7]), CLIPS denomina **agenda** al conjunto conflictivo, que contiene las reglas activadas (aquellas que pueden ser ejecutadas), ordenadas por prioridad. La primera regla en la agenda será la siguiente en ser ejecutada, y el orden y la prioridad se determinan en base a las siguientes normas:

- (a) Reglas activadas recientemente se colocan por encima de todas las reglas con menor prioridad, y por debajo de aquellas con prioridad mayor
- (b) Dada una regla, la estrategia de resolución actual es usada para determinar su posición entre las demás reglas con la misma prioridad
- (c) Si una regla se activa por la misma inserción o eliminación de un hecho, y los pasos (a) y (b) no son capaces de determinar la posición de esta, entonces la regla se coloca arbitrariamente en relación con las otras reglas con las que fue activada.

Para el punto (b), CLIPS proporciona siete estrategias de resolución de conflictos:

- **Estrategia en profundidad:** una regla activada se posiciona por encima de todas las reglas de la misma prioridad.
- **Estrategia en anchura:** las reglas activadas se posicionan debajo de todas las reglas de igual prioridad.
- **Estrategia de simplicidad:** se define la **especificidad de una regla** está determinada por la cantidad de comparaciones que deben ser comprobadas en el LHS de una regla.  
La estrategia de simplicidad posiciona las reglas activadas por encima de todas aquellas reglas con la misma prioridad y con mayor o igual especificidad.
- **Estrategia de complejidad:** de las reglas con la misma prioridad, las reglas activadas se posicionan por encima de todas las reglas con menor o igual especificidad.
- **Estrategia LEX:** de entre las reglas con la misma prioridad, las que se activan se sitúan utilizando la estrategia LEX de la herramienta OPS5. Esta consta de las siguientes normas:
  - i) Se usa cómo de recientes son las instancias de patrones que activaron la regla. Cada hecho e instancia se marca con un 'time tag' para indicar su novedad con respecto a los demás hechos e instancias.
  - ii) Las instancias de patrones asociadas a cada regla se ordenan en orden decreciente.
  - iii) Una activación con una instancia de patrón más reciente se coloca antes que las activaciones con instancias de patrones menos recientes.
  - iv) Si una activación tiene más instancias de patrones que otra y los time tags son todos iguales, entonces la activación con más time tags se coloca antes en la agenda.
  - v) Si dos activaciones tienen la misma novedad, la activación con mayor especificidad se posiciona por encima.



- **Estrategia MEA:** entre las reglas con la misma prioridad, las que se activan se colocan usando la estrategia MEA de la herramienta OPS5. Hace lo siguiente:
  - i) El time tag de la entidad patrón asociado al primer patrón se usa para determinar dónde colocar la activación.
  - ii) Una activación que tiene un primer patrón con un time tag mayor que el time tag del primer patrón de otra activación se coloca antes en la agenda.
  - iii) Si ambas activaciones tienen el mismo time tag asociado al primer patrón, entonces se utiliza la estrategia LEX.
- **Estrategia aleatoria:** a cada activación de una regla se le asigna un número aleatorio que se usa para determinar su posición entre las activaciones de igual prioridad.

Además, es importante destacar que CLIPS mantiene el **principio de refracción**, es decir, impide que una regla ya ejecutada se ejecute con el mismo conjunto de antecedentes.

2. ¿A qué estrategias de resolución de conflictos de las vistas en clase correspondería cada una de ellas?

La estrategia general de selección de la regla con prioridad más alta se corresponde con la estrategia vista en teoría **basada también en prioridad**.

Donde podemos observar las demás técnicas vistas en clase ([6]) es en aquellas situaciones en las que la prioridad de dos reglas coincide. Observamos:

- (a) La **selección arbitraria** es la estrategia aleatoria.
- (b) La **selección de la primera regla** es la estrategia en anchura.
- (c) La **selección de la regla más específica** es la estrategia de complejidad.
- (d) La **selección de la regla con el elemento añadido más recientemente** se corresponde con la estrategia MEA.
- (e) **Seleccionar una nueva regla**, esto lo hace para mantener el principio de refracción, pero no se corresponde con una estrategia explícita de CLIPS (no podemos elegir que esto deje de hacerse, y tampoco tendría mucho sentido en la gran mayoría de casos).
- (f) **Seleccionar todas las reglas:** CLIPS no utiliza esta estrategia de resolución de conflictos.

Las demás estrategias que usa CLIPS no las hemos estudiado en teoría, la estrategia en profundidad, de simplicidad y LEX, pero no son más que variantes simples de las que ya hemos mencionado antes.

## References

- [1] ayesha banu79. Library ontology.
- [2] M. Davies. Knowledge (explicit and implicit): Philosophical aspects. In Neil J. Smelser and Paul B. Baltes, editors, *International Encyclopedia of the Social and Behavioral Sciences*, pages 8126–8132. Pergamon, Oxford, 2001. <https://www.sciencedirect.com/science/article/pii/B0080430767009955>.
- [3] Chris Kimble. Knowledge management, codification and tacit knowledge. *Information Research*, 18(2), June 2013. <https://web.archive.org/web/20131024060920/http://informationr.net/ir/18-2/paper577.html>.
- [4] Protege Ontology Library. Protege ontology library.
- [5] Jose T Palma. Sistemas basados en reglas: Mycin. Apuntes de teorÃa, 2021.
- [6] Jose T. Palma. Sistemas basados en reglas: Tecnicas de equiparacion. Diapositivas de teoria, 2021.
- [7] Gary Riley. *CLIPS Reference Manual*. CLIPS, February 2021. <http://clipsrules.sourceforge.net/documentation/v640/bpg.pdf>.
- [8] Stanford. Pal documentation. <https://protege.stanford.edu/plugins/paltabs/pal-documentation/index.htm>.