

# Sistema Basado en Reglas: Ejemplo

**SRP**

**Carlos Carrascosa Casamayor**  
**Vicente J. Julián Inglada**



# Enunciados en Lenguaje Natural

1

**Se sanciona a todo aquel socio que no haya devuelto un libro en menos de 7 días desde que se lo llevó.**

2

**No se puede prestar un libro que ya esté prestado.**

3

**Sólo se pueden prestar libros a socios.**

4

**No se puede prestar un libro a un socio sancionado.**

5

**Mario y Pepe son socios.**

6

**“Hyperion” y “Endymion” son libros.**

7

**Mario se llevó prestado “Hyperion” el día 1.**

8

**Pepe se llevó prestado “Endymion” el día 5.**

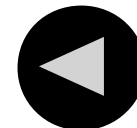
9

**Pepe ha devuelto “Endymion” hoy.**

10

**Mario ha devuelto “Hyperion” hoy.**

- ¿Se puede llevar Mario “Endymion” hoy?**



# Base de Hechos:

- 5.1 (socio nombre "Mario" sancionado no)
- 5.2 (socio nombre "Pepe" sancionado no)
- 6.1 (libro titulo "Hyperion" )
- 6.2 (libro titulo "Endymion" )
- 7 (prestamo libro "Hyperion" socio "Mario" dia 1)
- 8 (prestamo libro " Endymion" socio "Pepe" dia 5)
- 9 (devolucion libro "Endymion" socio "Pepe" dia 10 )
- 10 (devolucion libro "Hyperion" socio "Mario" dia 10 )
- 11 (hoy dia 10)
- 12 (prestamo libro "Endymion" socio "Mario" dia 10 )

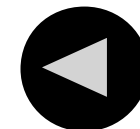


# Base de Reglas (I):

1

```
(defrule sancionar
  ?hecho <- ( socio nombre ?x sancionado no )
            ( libro titulo ?y )
            ( prestamo libro ?y socio ?x dia ?z )
            ( not ( and (devolucion libro ?y socio ?x dia ?w )
                       ( test (> ?w ?z) )
                       ( test (> (+ ?z 7) ?w ) ) ) ) )

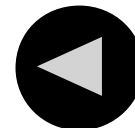
=>
(retract ?hecho)
(assert (socio nombre ?x sancionado si)) )
```



# Base de Reglas (II):

2

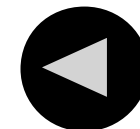
```
(defrule prestamo-unico
  ( libro titulo ?x )
  ( socio nombre ?y $? )
  ( socio nombre ?z $? )
  ?hecho <- ( prestamo libro ?x socio ?y dia ?d1 )
             ( prestamo libro ?x socio ?z dia ?d2 )
             ( test ( > ?d1 ?d2 ) )
             ( not ( and ( devolucion libro ?x socio ?z dia ?d3 )
                        ( test ( > ?d3 ?d2 ) )
                        ( test ( < ?d3 ?d1 ) ) ) ) )
  =>
  (retract ?hecho)
  (printout t "No se puede prestar dos veces" crlf))
```



# Base de Reglas (III):

3

```
(defrule prestamo-socio
  ( libro titulo ?x )
  ?hecho <- ( prestamo libro ?x socio ?y $? )
             (not (socio nombre ?y $? ) )
  =>
  (retract ?hecho)
  (printout t "No se puede prestar a
              una persona que no sea socia" crlf))
```



# Base de Reglas (IV):

4

```
(defrule prestamo-sancionado
  ( libro titulo ?x )
  ( socio nombre ?y sancionado si )
  ( hoy dia ?z )
  ?hecho <- ( prestamo libro ?x socio ?y dia ?z )
  =>
  (retract ?hecho)
  (printout t "No se puede prestar
    a una persona sancionada" crlf))
```

