

Tema 3: Hechos

Representación de la Información

Representación de la información

- **Hechos.** Ordenados y no Ordenados.
- **Variables globales.** Constructor defglobal.
- **Objetos.** POO. Instancias de objetos.

Hechos: Órdenes de uso

Órdenes de utilización de Hechos:

- assert
- facts
- retract
- modify
- duplicate
- deftemplate
- deffacts
- reset
- clear

Hechos: Tipos y ejemplos

(nombre-relación campo1 campo2 ...)

- Ordenados

(casa calle-nueva 32)

(animales perro gato vaca)

(asignatura IA 2 Pedro)

- No ordenados (hechos plantillas)

(coche (marca Ford) (modelo focus) (color gris))

(cliente (nombre "Juan Pérez") (tlf 957123456))

Hechos: Ejemplo Ordenados

CLIPS> (assert (libro "El Quijote"))

<Fact-0>

CLIPS> (assert (arbol encina))

<Fact-1>

CLIPS> (facts)

f-0 (libro "El Quijote")

f-1 (arbol encina)

For a total of 2 facts.

CLIPS>

Hechos No Ordenados

- El orden en los campos no es importante.

(clase (estudiantes 30) (profesor "Marta Ramírez"))

(clase (profesor "Marta Ramírez") (estudiantes 30))

- Se pueden modificar utilizando las órdenes (modify) y (duplicate).

Hechos: comandos

(assert <hecho>+)

(facts [<inicio> [<final> [máximo]]])

(retract <índice>+ | *)

(modify <índice> <nueva-casilla>+)

(duplicate <índice> <nueva-casilla>+)

<nueva-casilla>::= (<nombre> <valor>)

Afirmando Hechos

- El comando `assert` introduce datos en la base de hechos.
- La orden `(facts)` sirve para ver la base de hechos. La orden `(clear)` limpia la base de hechos.
- La orden `(reset)` borra hechos e inserta el hecho especial `(initial-fact)` y aquellos que formen parte de un `deffacts`.

Afirmando Hechos

```
CLIPS> (assert (a) (b) (c))
```

```
<Fact-2>
```

```
CLIPS> (facts)
```

```
f-0    (a)
```

```
f-1    (b)
```

```
f-2    (c)
```

For a total of 3 facts.

```
CLIPS> (facts 0)
```

```
f-0    (a)
```

```
f-1    (b)
```

```
f-2    (c)
```

For a total of 3 facts.

Afirmando Hechos

CLIPS> (facts 1)

f-1 (b)

f-2 (c)

For a total of 2 facts.

CLIPS> (facts 2)

f-2 (c)

For a total of 1 fact.

CLIPS> (facts 0 1)

f-0 (a)

f-1 (b)

For a total of 2 facts.

CLIPS> (facts 0 2 2)

f-0 (a)

f-1 (b)

For a total of 2 facts.

CLIPS>

Afirmando Hechos

- Orden (assert)
 - No se puede insertar un hecho que ya existe, excepto utilizando la orden (override-set-fact-duplication).
 - CLIPS> (assert (libro “El Quijote”))
 - <Fact-0>
 - CLIPS> (assert (libro “El Quijote”))
 - FALSE
 - CLIPS>
 - Los índices de los hechos no cambian aunque se borren hechos.

Iniciando la Base de Hechos

- La orden (clear).
 - Elimina todos los hechos de la base de hechos.
 - Reinicializa el índice de hechos a cero.
 - Reestablece CLIPS a su estado original.
 - También elimina la base de conocimiento.

Iniciando la base de Hechos

```
CLIPS> (assert (a) (b) (c))
```

```
<Fact-2>
```

```
CLIPS> (facts)
```

```
f-0    (a)
```

```
f-1    (b)
```

```
f-2    (c)
```

For a total of 3 facts.

```
CLIPS> (clear)
```

```
CLIPS> (facts)
```

```
CLIPS>
```

Hechos: Legibilidad

- Utilizar el retorno de carro (intro) después de cada campo.
- Clips reemplaza los retornos de carro y los tabuladores con espacios simples.

Hechos Legibilidad

CLIPS> (clear)

CLIPS> (assert (lista-comidas
 helado
 fresas
 nata
 pescado))

<Fact-0>

CLIPS> (facts)

f-0 (lista-comidas helado fresas nata pescado)

For a total of 1 fact.

CLIPS>

Hechos Legibilidad

- Los espacios en blanco se usan para separar múltiples campos.
- Cuidado con las mayúsculas y minúsculas que pueden producir hechos distintos.
- Dentro de una “cadena de caracteres” los espacios en blanco sí afectan.

Hechos Legibilidad

CLIPS> (clear)

CLIPS> (assert (es-animal caballo))

<Fact-0>

CLIPS> (assert (es-animal caballo))

FALSE

CLIPS> (assert (es-animal Caballo))

<Fact-1>

CLIPS> (facts)

f-0 (es-animal caballo)

f-1 (es-animal Caballo)

For a total of 2 fact.

CLIPS>

Hechos Legibilidad

CLIPS> (clear)

CLIPS> (assert (es-animal "perro"))

<Fact-0>

CLIPS> (assert (es-animal " perro "))

<Fact-1>

CLIPS> (facts)

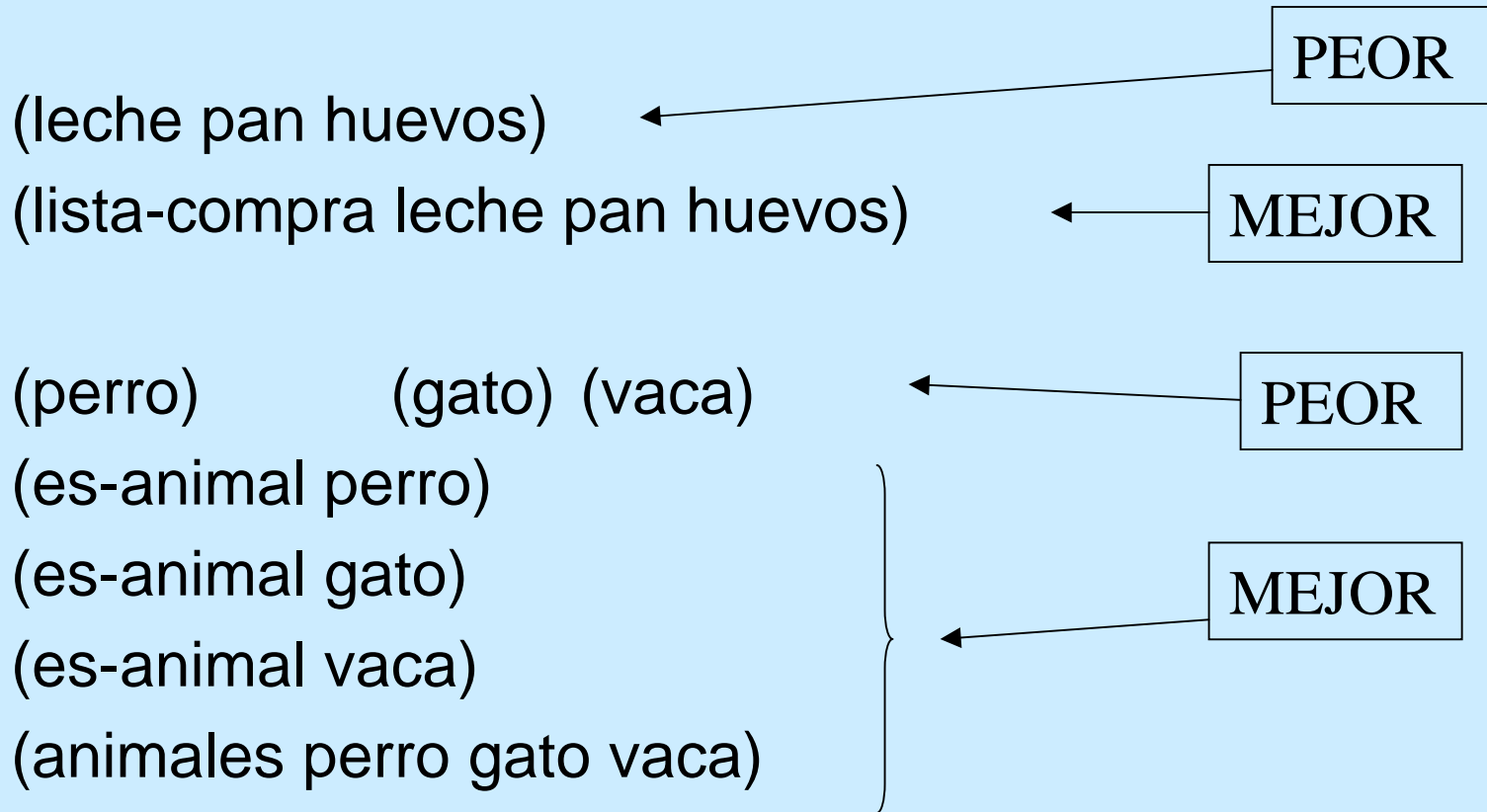
f-0 (es-animal "perro")

f-1 (es-animal " perro ")

For a total of 4 facts.

CLIPS>

Hechos: estilo



(nombre-relación campo1 campo2 ...)

Retractar Hechos

- La orden (retract) elimina hechos de la base de hechos.
 - Se puede especificar un índice o el propio hecho. (retract <índice>)
 - No se puede eliminar un hecho ya eliminado.
 - Se pueden eliminar varios hechos.
 - Se pueden eliminar todos los hechos con *. (retract *)

Retractor Hechos

CLIPS> (clear)

CLIPS> (assert (animal-es pato))

<Fact-0>

CLIPS> (assert (animal-sonido cua))

<Fact-1>

CLIPS> (assert (El pato dice "Cua"))

<Fact-2>

CLIPS> (facts)

f-0 (animal-es pato)

f-1 (animal-sonido cua)

f-2 (El pato dice "Cua")

For a total of 3 facts.

CLIPS>

Retractar Hechos

CLIPS> (retract 2)

CLIPS> (facts)

f-0 (animal-es pato)

f-1 (animal-sonido cua)

For a total of 2 facts.

CLIPS>

CLIPS> (retract 2)

[PRNTUTIL1] Unable to find fact f-2.

CLIPS>

CLIPS> (retract 1)

CLIPS> (facts)

f-0 (animal-es pato)

For a total of 1 fact.

CLIPS> (retract *)

CLIPS> (facts)

CLIPS>

Órdenes de Depuración

- La orden (watch facts).
 - Muestra los hechos que están siendo insertados o eliminados.
 - = > Hecho entrando en lista de hechos
 - < = Hecho saliendo de lista de hechos

CLIPS> (clear)

CLIPS> (watch facts)

CLIPS> (assert (animal-es pato))

=> f-0 (animal-es pato)

<Fact-0>

CLIPS>

Órdenes de Depuración

```
CLIPS> (reset)
<== f-0    (animal-es pato)
==> f-0    (initial-fact)
CLIPS> (assert (animal-es pato))
==> f-1    (animal-es pato)
<Fact-1>
CLIPS> (retract 1)
<== f-1    (animal-es pato)
CLIPS> (facts)
f-0    (initial-fact)
For a total of 1 fact.
CLIPS>
```


Órdenes de Depuración

- La orden (unwatch facts).
 - Desactiva la presentación de hechos.
CLIPS> (unwatch facts)
- Otros elementos a visualizar:
 - (watch slots).
 - (watch rules).
 - (watch methods).
 - (watch deffunctions).
 - (watch compilations).
 - (watch globals).
 - (watch all).
 - ...