



Συστήματα Γνώσης

Πρακτικό Κομμάτι Μαθήματος
Το Σύστημα Κανόνων CLIPS

Νίκος Βασιλειάδης, Αναπλ. Καθηγητής
Τμήμα Πληροφορικής



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.





ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ
ΑΚΑΔΗΜΑΙΚΑ
ΜΑΘΗΜΑΤΑ



Το Σύστημα Κανόνων CLIPS

Ιστορικά Στοιχεία...

- **C Language Integrated Production System**
- Περιβάλλον προγραμματισμού με κανόνες, αντικείμενα και συναρτήσεις
- Αναπτύχθηκε από τη NASA το 1985
 - Υλοποιήθηκε με τη γλώσσα C
 - Σύνταξη θυμίζει OPS5
 - Λειτουργικότητα όμοια με ART
- Τρέχει σε DOS, Windows, UNIX, VMS
- Υποστηρίζει τον αντικειμενοστραφή προγραμματισμό (COOL)



Δομή του CLIPS

- Λίστα Γεγονότων (facts list)
(name george)
- Βάση κανόνων (rule/knowledge base)
(defrule rain "in case of rain"
 (weather rain) =>
 (assert (action "take umbrella")))
- Μηχανισμός Εξαγωγής Συμπερασμάτων (Inference Engine)
 - Στρατηγικές Επίλυσης Ανταγωνισμού (Conflict Resolution Strategies)



Εκτέλεση Προγράμματος

- Πρόγραμμα
 - Ένα σύνολο από κανόνες και γεγονότα
- Εκτέλεση
 - Ακολουθία από πυροδοτήσεις κανόνων των οποίων οι συνθήκες ικανοποιούνται
- Ικανοποίηση συνθηκών
 - Ταυτοποίηση με γεγονότα
- Η εκτέλεση τερματίζεται όταν:
 - Δεν υπάρχουν άλλοι κανόνες προς πυροδότηση
 - Κληθεί συγκεκριμένη εντολή τερματισμού (**halt**)



Κύκλος Λειτουργίας CLIPS

1. Εύρεση όλων των κανόνων των οποίων οι συνθήκες ικανοποιούνται και προσθήκη τους στην ατζέντα (*agenda - conflict set*).
2. Αν η ατζέντα είναι κενή τότε η εκτέλεση τερματίζεται.
3. Επιλογή ενός κανόνα με βάση τη στρατηγική επίλυσης ανταγωνισμού (*conflict resolution*) και εκτέλεσή του.
4. Επιστροφή στο βήμα 1, εκτός αν υπάρχει εντολή τερματισμού (*halt*).



Σύνταξη του CLIPS

- Θυμίζει LISP
- Είναι Case-Sensitive
- Δομικά Στοιχεία:
 - Σύμβολα, π.χ. `you`, `why_this`, `good-morning`
 - Αλφαριθμητικά, π.χ. `"This is a String"`
 - Αριθμοί, π.χ. `24`, `-45.6`, `8e11`
 - Σχόλια: ότι ακολουθεί τον χαρακτήρα `;`
- Μεταβλητές
 - Μονότιμες π.χ. `?var`, `?x`
 - Πολλαπλών Τιμών π.χ. `$?fruits`, `$?shopping`

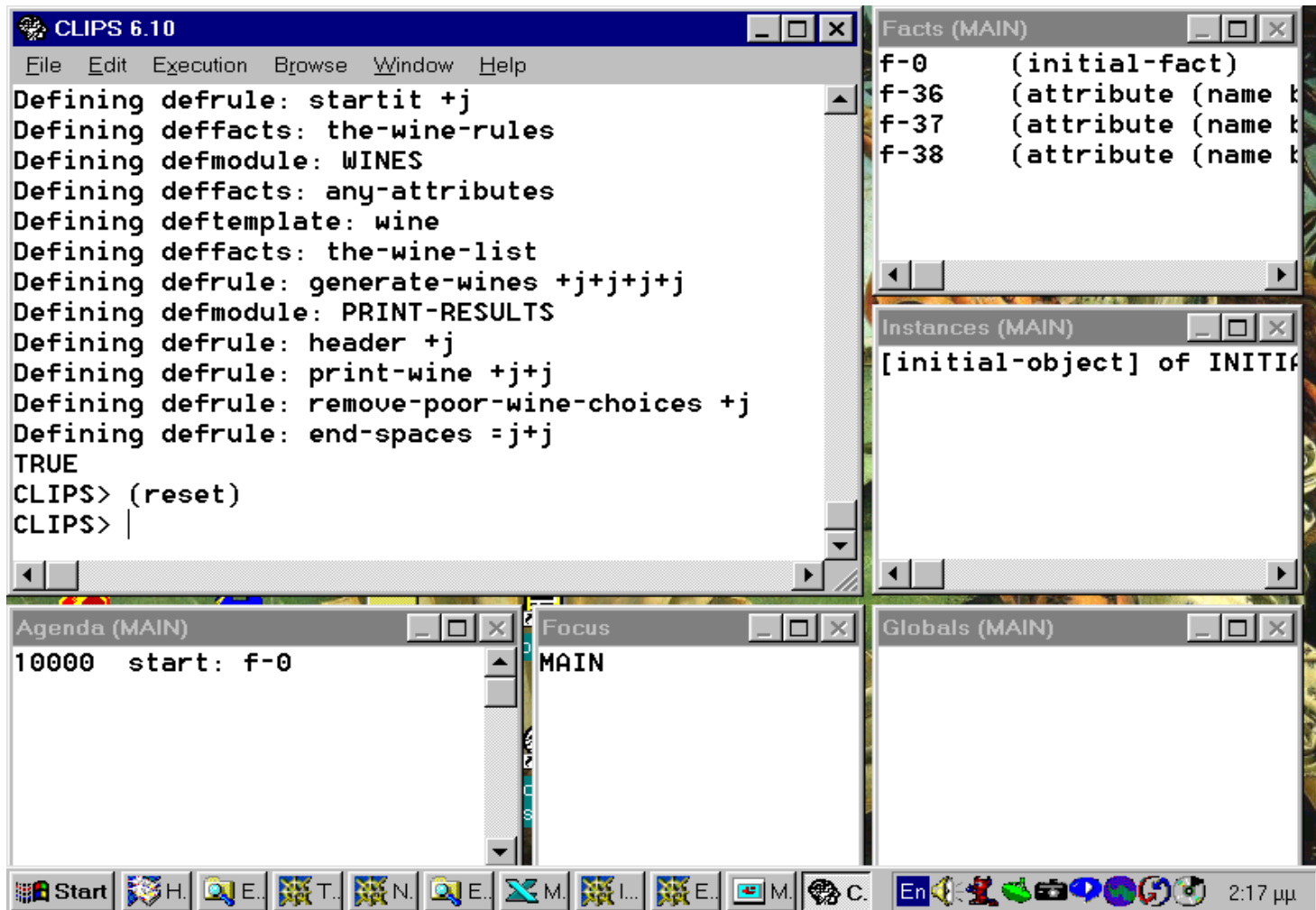


Μεταβλητές

- Εμφανίζονται
 - Στις συνθήκες ενός κανόνα
 - Στις ενέργειες ενός κανόνα
- Παίρνουν τιμές
 - Κυρίως στις συνθήκες των κανόνων μέσω της διαδικασίας ταυτοποίησης
 - Η ανάθεση τιμής σε μεταβλητή στις ενέργειες ενός κανόνα είναι δυνατή με τη χρήση κατάλληλης συνάρτησης, αλλά καλό είναι να αποφεύγεται.
- Η εμβέλεια των μεταβλητών περιορίζεται στον κανόνα που αυτές εμφανίζονται.



Περιβάλλον CLIPS



Γεγονότα

- Λίστες από σύμβολα που περικλείονται σε παρενθέσεις
π.χ.

`(name John Papas)`

`(shopping_list cheese wine bread book)`

`(days Monday Friday Sunday)`

- Το πρώτο στοιχείο της λίστας είναι ο **τύπος** του γεγονότος και τα υπόλοιπα είναι οι **παράμετροι**
 - Κάθε γεγονός έχει μοναδικό αριθμό-ταυτότητα (**fact index**) που καθορίζεται αυτόματα
- Εμφάνιση γεγονότων

`CLIPS> (facts)`

`f-0 (name John Papas)`

`for a total of 1 fact.`



Εισαγωγή Γεγονότων

- Με τη χρήση της εντολής `assert`
(`assert <fact>`)

```
CLIPS> (assert (gift book))
```

```
<Fact-0>
```

```
CLIPS> (facts)
```

```
f-0          (gift book)
```

```
for a total of 1 fact.
```

```
CLIPS> (assert (day Monday) (phone  
891363))
```

```
CLIPS> (facts)
```

```
f-0          (gift book)
```

```
f-1          (day Monday)
```

```
f-2          (phone 891363)
```

```
for a total of 3 facts.
```



Εισαγωγή Γεγονότων

- Με τη χρήση της εντολής **deffacts** (μαζί με **reset**)

```
(deffacts <name> "comments"  
  (<fact1>  
   (<fact2>  
    ...  
    (<fact n>  
     )
```



Εντολή deffacts

```
CLIPS> (deffacts colours "this is  
to insert some colours" (colour  
red) (colour blue) (colour green))
```

```
CLIPS> (reset)
```

```
CLIPS> (facts)
```

```
f-0      (initial-fact)
```

```
f-1      (colour red)
```

```
f-2      (colour blue)
```

```
f-3      (colour green)
```

```
for a total of 4 facts.
```

```
CLIPS> (facts 2)
```

```
f-2      (colour blue)
```

```
f-3      (colour green)
```

```
for a total of 2 facts.
```



Επιτρέπονται διπλά γεγονότα?

```
CLIPS> (assert (name nick))
```

```
<Fact-0>
```

```
CLIPS> (facts)
```

```
f-0 (name nick)
```

```
For a total of 1 facts.
```

```
CLIPS> (assert (name nick))
```

```
FALSE
```

```
CLIPS> (facts)
```

```
f-0 (name nick)
```

```
For a total of 1 facts.
```



Διαγραφή Γεγονότων

- Χρήση της εντολής (**retract**)
(**retract <fact-index>**)

```
CLIPS> (retract 1 3)
CLIPS> (facts)
f-0      (initial-fact)
f-2      (colour blue)
for a total of 2 facts.
CLIPS> (retract 1)
[PRNUTIL1] Unable to find fact-1
```

- Χρήση της εντολής (**clear**)



Παρακολούθηση Προσθήκης και Διαγραφής Γεγονότων

- Χρήση της εντολής (`watch facts`)

```
CLIPS> (watch facts)
CLIPS> (assert (age 23))
==> f-0      (age 23)
<Fact-0>
CLIPS> (reset)
<== f-0      (age 23)
==> f-0      (initial-fact)
CLIPS> (assert (mobile 0977233445))
==> f-1      (mobile 0977233445)
<Fact-1>
CLIPS> (retract 1)
<== f-1      (mobile 0977233445)
CLIPS> (facts)
f-0 (initial-fact)
for a total of 1 fact.
```



Κανόνες

- Μορφή:
 - **if** (Συνθήκες) **then** (Ενέργειες)
- Συνθήκες:
 - Γεγονότα & Μεταβλητές
- Ενέργειες:
 - Πράξεις (εντολές) μετά την ενεργοποίηση του κανόνα
- Σημασία:
 - **Εάν** ικανοποιούνται οι συνθήκες (δηλαδή ταυτοποιούνται με τη λίστα γεγονότων),
 - **Τότε** εκτέλεσε τις ενέργειες



Κανόνες

- Σύνταξη:

```
(defrule <όνομα κανόνα>  
  "<σχόλια>"  
  (<συνθήκη 1>  
   ...  
   (<συνθήκη n>  
=>  
  (<εντολή 1>  
   ...  
   (<εντολή m>  
)
```



Κανόνες

- Παράδειγμα

```
(defrule soccer-time
```

```
  "Warns for your soccer time"
```

```
  (day sunday)
```

```
  (time afternoon)
```

```
=>
```

```
  (assert (go for soccer)))
```



Ταυτοποίηση

Συνθήκη	Γεγονός	Αναθέσεις τιμών
<code>(day ?d ?t)</code>	<code>(day fri 12)</code>	<code>?d=fri</code> <code>?t=12</code>
<code>(list \$?lst)</code>	<code>(list a b c d)</code>	<code> \$?lst=(a b c d)</code>
<code>(car ?c model \$?m license ?l)</code>	<code>(car 1 model BMW FIAT license wqw45)</code>	<code>?c=1</code> <code> \$?m=(BMW FIAT)</code> <code>?l=wqw45</code>



Ταυτοποίηση

Συνθήκη	Γεγονός
<code>(day ?d ?t)</code>	<code>(days fri 12)</code>
<code>(list a b \$?lst)</code>	<code>(list 1 2 c d e f)</code>
<code>(car ?c type \$?m license ?l)</code>	<code>(car 1 2 type BMW FIAT license wqw45)</code>



Παράδειγμα Επιλογής Δώρου

Γεγονότα – Κανόνες Επιλογής Δώρου

```
;;; FACTS
(deffacts ini "initial facts"
  (age 19)
  (like music)
  (educated yes))
```

```
;;; RULES FOR CHOOSING GIFTS
(defrule cd "cd gift rule"
  (or (agegroup middle)
      (agegroup old))
  (like music)
  =>
  (assert (gift CD)))
```



Παράδειγμα Επιλογής Δώρου

Κανόνες Επιλογής Δώρου

```
(defrule toy "toy gift rule"  
  (agegroup child)  
  (price expensive)  
=>  
  (assert (gift toy)))
```

```
(defrule clothes "clothes gift  
rule"  
  (agegroup middle)  
  (price expensive)  
=>  
  (assert (gift clothes)))
```



Παράδειγμα Επιλογής Δώρου

Κανόνες Επιλογής Δώρου

```
(defrule flowers "flowers gift  
rule"  
  (agegroup old)  
  (price expensive)  
=>  
  (assert (gift flowers)))
```

```
(defrule book "book gift rule"  
  (or (agegroup middle)  
      (agegroup child))  
  (educated yes)  
=>  
  (assert (gift book)))
```



Παράδειγμα Επιλογής Δώρου

Κανόνες Ηλικιακών Ομάδων

```
(defrule child "child age rule"  
  (age ?x)  
  =>  
  (if (< ?x 15) then (assert (agegroup  
    child))))
```

```
(defrule middle "middle age rule"  
  (age ?x)  
  =>  
  (if (and (> ?x 14) (< ?x 35)) then  
    (assert (agegroup middle))))
```

```
(defrule old "old age rule"  
  (age ?x)  
  =>  
  (if (> ?x 34) then (assert (agegroup  
    old))))
```



Παράδειγμα Επιλογής Δώρου

```
CLIPS> (load
  "gift.clp")
TRUE
CLIPS> (reset)
CLIPS> (facts)
f-0      (initial-
  fact)
f-1      (age 19)
f-2      (like
  music)
f-3      (educated
  yes)
For a total of 4
facts.
```

```
CLIPS> (run)
CLIPS> (facts)
f-0      (initial-
  fact)
f-1      (age 19)
f-2      (like
  music)
f-3      (educated
  yes)
f-4      (agegroup
  middle)
f-5      (gift CD)
f-6      (gift book)
For a total of 7
facts.
```





Τέλος Ενότητας

Επεξεργασία: Εμμανουήλ Ρήγας

Θεσσαλονίκη, 17/3/2014



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ