



Γνώση

Η γνώση είναι διαφορετική από τα δεδομένα

Γνώση (*knowledge*) είναι ο ‘κοινός παράγοντας’ (π.χ. κανόνες) που **περιγράφει συνοπτικά** τις συσχετίσεις μεταξύ των δεδομένων ενός προβλήματος.

Η γνώση για κάποιο πρόβλημα είτε προσφέρεται από τους ειδικούς (experts) στο αντικείμενο του προβλήματος είτε εξάγεται αυτόματα από τα δεδομένα.

Στοιχεία της Γνώσης

- ❖ Οντότητες – αντικείμενα
- ❖ Ιδιότητες οντοτήτων - Συσχετίσεις μεταξύ οντοτήτων
- ❖ Κανόνες (rules)

Αναπαράσταση Γνώσης και Συλλογιστικές

Αναπαράσταση γνώσης είναι ένα σύνολο συντακτικών και σημασιολογικών παραδοχών, οι οποίες καθιστούν δυνατή την περιγραφή ενός κόσμου χρησιμοποιώντας ένα μονοσήμαντο και τυποποιημένο συμβολισμό.

- ❖ Μία μέθοδος αναπαράστασης γνώσης έχει:
 - ☐ Συντακτικό (syntax)
 - ☐ Σημασιολογία (semantics).
- ❖ **Συνεπαγωγή:** Εστω μια βάση γνώσης (KB) και μια πρόταση a . Εάν η πρόταση a μπορεί να εξαχθεί από τη βάση KB, τότε συμβολίζουμε $KB \models a$
- ❖ **Κάθε μέθοδος αναπαράστασης της γνώσης έχει μια διαφορετική διαδικασία εξαγωγής συμπερασμάτων (Δ.Ε.Σ).**



- ❑ **Δ.Ε.Σ:** Μηχανισμός που χρησιμοποιείται για εξαγωγή συμπερασμάτων από υπάρχουσα γνώση.
- ❑ Είτε δοθείσης της KB η Δ.Ε.Σ. δημιουργεί νέες προτάσεις α τέτοιες ώστε $KB \models \alpha$
- ❑ Είτε δοθείσης KB και μιας πρότασης α η Δ.Ε.Σ αποφασίζει εάν $KB \models \alpha$

- ❑ **Απόδειξη (proof):** Τα βήματα μιας Δ.Ε.Σ.

- ❑
- ❑ Μια Δ.Ε.Σ είναι **ορθή** όταν εξάγει μόνο προτάσεις α για τις οποίες $KB \models \alpha$ (δεν δημιουργεί ‘ψεύτικες’ συνεπαγωγές)
- ❑ Μια Δ.Ε.Σ είναι **πλήρης** όταν μπορεί να βρει απόδειξη για κάθε συνεπαγόμενη πρόταση.
- ❑ Μια Δ.Ε.Σ θέλουμε να είναι ορθή, πλήρης, γρήγορη και εύκολα υλοποιήσιμη



- ❑ Όσο αυξάνεται η ευελιξία της αναπαράστασης τόσο αυξάνεται η πολυπλοκότητα της διαδικασίας εξαγωγής συμπερασμάτων.
- ❑ Η φυσική γλώσσα έχει μέγιστη εκφραστικότητα αλλά εμφανίζει δυσκολίες στην εξαγωγή συμπερασμάτων λόγω της αμφισημίας (ambiguity) και της ερμηνείας με βάση τα συμφραζόμενα (context).
- ❑
- ❑ Η προτασιακή λογική έχει απλή ΔΕΣ αλλά πολύ μικρή εκφραστικότητα.
- ❑ Για να είναι χρήσιμο ένα σύστημα αναπαράστασης γνώσης θα πρέπει να συνδυάζει ικανοποιητική εκφραστικότητα και αποδοτική Δ.Ε.Σ. (π.χ. κατηγορηματική λογική)



Μηχανισμός Εξαγωγής Συμπερασμάτων

Inference Mechanism

❖ Δύο μεγάλες κατηγορίες συστημάτων εξαγωγής συμπερασμάτων

- ❑ **Αιτιολόγηση προς τα πίσω (backward chaining):** Ξεκινάμε από πιθανά συμπεράσματα, τα οποία διατυπώνουμε ως ερωτήσεις και φτάνουμε στις αιτίες που τα υποστηρίζουν (απόδειξη θεωρημάτων, απάντηση ερωτήσεων) (π.χ. Prolog)
- ❑ **Αιτιολόγηση προς τα εμπρός (forward chaining):** Ξεκινάμε από τα δεδομένα του προβλήματος και φτάνουμε σε συμπεράσματα, αποφάσεις ή ενέργειες (π.χ. CLIPS, JESS).



Μέθοδοι Αναπαράστασης Γνώσης

❖ Λογική

- ❑ Προτασιακή λογική (propositional logic)
- ❑ Κατηγορηματική λογική (predicate logic) ή λογική πρώτης τάξης (Prolog)
- ❑ Λογική υψηλότερης τάξης

❖ Δομημένες αναπαραστάσεις γνώσης

- ❑ Σημασιολογικά Δίκτυα (semantic networks)
- ❑ Πλαίσια (frames)
- ❑ Σενάρια (scripts)
- ❑ Γλώσσες περιγραφής οντολογιών (ontologies)

❖ Συστήματα παραγωγής γνώσης με αιτιολόγηση προς τα εμπρός (CLIPS)

❖ Συστήματα αναπαράστασης αβεβαιότητας (π.χ. δίκτυα πεποιθήσεων, ασαφής λογική).

Λογική

Η μαθηματική λογική (*mathematical logic*) είναι η συστηματική μελέτη των έγκυρων ισχυρισμών (*valid arguments*) με χρήση εννοιών από τα μαθηματικά.

- ❖ Ένας ισχυρισμός (*argument*) αποτελείται από συγκεκριμένες προτάσεις, τις υποθέσεις (*premises*), από τις οποίες παράγονται άλλες προτάσεις που ονομάζονται συμπεράσματα (*conclusions*)

Όλοι οι άνθρωποι είναι θνητοί,	(υπόθεση)
Ο Σωκράτης είναι άνθρωπος,	(υπόθεση)
επομένως, ο Σωκράτης είναι θνητός	(Συμπέρασμα)



Προτασιακή Λογική

- ❖ Στην *προτασιακή λογική (propositional logic)* κάθε γεγονός αναπαριστάται με μια λογική πρόταση, η οποία χαρακτηρίζεται είτε ως *αληθής (true)* ή ως *ψευδής (false)*.
- ❖ Οι λογικές προτάσεις μπορούν να συνδυαστούν με τη χρήση *λογικών συμβόλων* ή *συνδετικών (connectives)*.

Συνδετικά και Σημασία

Σύμβολο	Ονομασία / Επεξήγηση
\wedge	σύζευξη (λογικό "ΚΑΙ")
\vee	διάζευξη (λογικό "Η")
\neg	άρνηση
\rightarrow	συνεπαγωγή ("ΕΑΝ ΤΟΤΕ")
\leftrightarrow	διπλή συνεπαγωγή ή ισοδυναμία ("ΑΝ ΚΑΙ ΜΟΝΟ ΑΝ").

Παράδειγμα

P : "Ο Νίκος είναι προγραμματιστής"

Q : "Ο Νίκος έχει Υπολογιστή"

$P \rightarrow Q$: *Εάν "Ο Νίκος είναι προγραμματιστής", τότε "Ο Νίκος έχει Υπολογιστή"*

R : "Το τρίγωνο ΑΒΓ είναι ισόπλευρο"

V : "Το τρίγωνο ΑΒΓ έχει όλες τις πλευρές του ίσες"

$R \leftrightarrow V$: "Το τρίγωνο ΑΒΓ είναι ισόπλευρο" *αν και μόνο αν* "Το τρίγωνο ΑΒΓ έχει όλες τις πλευρές του ίσες"

Μηχανισμοί Εξαγωγής Συμπερασμάτων

❖ Πίνακες αλήθειας (Truth Tables)

❖ "Τρόπος του θέτειν" (modus ponens):

$$P \wedge (P \rightarrow Q) \models Q \quad (\text{modus ponens})$$

❖ "Αρχή της ανάλυσης" ή αναγωγής (resolution principle).

$$(P \vee Q) \wedge (R \vee \neg Q) \models P \vee R \quad (\text{αρχή της ανάλυσης})$$

Μειονέκτημα: Μικρή εκφραστικότητα

Πλεονέκτημα: Εύκολη εξαγωγή συμπερασμάτων

$P \rightarrow Q \leftrightarrow \neg P \vee Q$: Χρησιμοποιείται για μετασχηματισμό των κανόνων σε Conjunctive Normal Form (σύζευξη από όρους που καθένας τους είναι διάζευξη δύο ορισμάτων)

Παράδειγμα

"Ο Νίκος είναι προγραμματιστής"

Εάν "Ο Νίκος είναι προγραμματιστής" τότε "Ο Νίκος έχει Υπολογιστή"

Από τα ανωτέρω εξάγεται (modus ponens) το "Ο Νίκος έχει Υπολογιστή"

Παράδειγμα

"έχει ομίχλη" τότε "υπάρχει κίνδυνος" (1)

"υπάρχει κίνδυνος" τότε "απαιτείται χαμηλή ταχύτητα" (2)

"έχει ομίχλη" (3)

Από τα ανωτέρω εξάγεται (π.χ. με την αρχή της ανάλυσης) ότι

"απαιτείται χαμηλή ταχύτητα"

Απόδειξη:

(1) και (3) \rightarrow "υπάρχει κίνδυνος" (4)

(2) και (4) \rightarrow "απαιτείται χαμηλή ταχύτητα" (5)

Κατηγορηματική Λογική

- ❖ Η κατηγορηματική λογική (*predicate logic*) επεκτείνει την προτασιακή λογική εισάγοντας
 - ❑ όρους (*terms*),
 - ❑ κατηγορήματα (*predicates*) (σημαντικό: ένα κατηγορήμα είναι TRUE ή FALSE)
 - ❑ ποσοδείκτες (*quantifiers*).

Σύμβολο	Ονομασία / Επεξήγηση
\wedge	Σύζευξη (λογικό "ΚΑΙ")
\vee	Διάζευξη (λογικό "Η")
\neg	Άρνηση
\rightarrow	συνεπαγωγή ("ΕΑΝ ΤΟΤΕ")
\leftrightarrow	ισοδυναμία ("ΕΑΝ ΚΑΙ ΜΟΝΟ ΕΑΝ")
\forall	καθολικός ποσοδείκτης ($\forall x$ σημαίνει για κάθε x)
\exists	υπαρξιακός ποσοδείκτης ($\exists x$ σημαίνει υπάρχει x)

Κατηγορήματα και Ορίσματα

- ❖ Ένα γεγονός αναπαριστάται με έναν *ατομικό τύπο* της μορφής:

$$P(A_1, A_2, \dots, A_n)$$

όπου το P ονομάζεται *κατηγόρημα* (*predicate*) και τα A_1, A_2, \dots, A_n *ορίσματα* (*arguments*).

- ❖ Οι ατομικοί τύποι με τη χρήση των **ποσοδεικτών και μεταβλητών** συνδυάζονται για τη διατύπωση **κανόνων** της μορφής (**συνθήκες \rightarrow συμπεράσματα**)

Παραδείγματα

Οι γονείς αγαπούν τα παιδιά τους $\forall x \forall y (\text{parent}(x, y) \rightarrow \text{loves}(x, y))$

Όλοι οι παίκτες του μπάσκετ είναι ψηλοί $\forall x (\text{basket_player}(x) \rightarrow \text{tall}(x))$

Υπάρχουν φτωχοί άνθρωποι $\exists x (\text{man}(x) \wedge \text{poor}(x))$

Κάθε άνθρωπος έχει μητέρα μεγαλύτερη από αυτόν $\forall x (\text{man}(x) \rightarrow \exists y (\text{mother}(y, x) \wedge \text{elder}(y, x)))$

Ατυπος κανόνας: όταν έχουμε \forall βάζουμε \rightarrow , όταν έχουμε \exists βάζουμε \wedge

Παράδειγμα Αναπαράστασης σε Κατηγορηματική Λογική

- ❖ Κάθε ζώο που έχει φτερά και γεννάει αυγά είναι πουλί.
- ❖ Κάθε θηλαστικό που τρέφεται με κρέας ή έχει κοφτερά δόντια είναι σαρκοβόρο.
- ❖ Κάθε σαρκοβόρο με χρώμα καφέ-πορτοκαλί που έχει ρίγες είναι τίγρης.
- ❖ Κάθε σαρκοβόρο με χρώμα καφέ-πορτοκαλί που έχει μαύρες βούλες είναι τσιτάχ.
- ❖ Κάθε πουλί το οποίο δεν πετά και κολυμπά είναι πιγκουΐνος.

$\forall x \text{ (} \text{ΕΧΕΙ}(x, \text{ΦΤΕΡΑ}) \wedge \text{ΓΕΝΝΑΕΙ}(x, \text{ΑΥΓΑ}) \text{) } \rightarrow \text{ΕΙΝΑΙ}(x, \text{ΠΟΥΛΙ}) .$

$\forall x \text{ (} \text{ΕΙΔΟΣ}(x, \text{ΘΗΛΑΣΤΙΚΟ}) \wedge (\text{ΤΡΕΦΕΤΑΙ}(x, \text{ΚΡΕΑΣ}) \vee \text{ΕΧΕΙ}(x, \text{ΔΟΝΤΙΑ}(\text{ΚΟΦΤΕΡΑ}))) \text{) } \rightarrow \text{ΕΙΝΑΙ}(x, \text{ΣΑΡΚΟΒΟΡΟ}) .$

$\forall x \text{ (} \text{ΕΙΝΑΙ}(x, \text{ΣΑΡΚΟΒΟΡΟ}) \wedge \text{ΧΡΩΜΑ}(x, \text{ΚΑΦΕ-ΠΟΡΤΟΚΑΛΙ}) \wedge \text{ΕΧΕΙ}(x, \text{ΡΙΓΕΣ}(\text{ΜΑΥΡΕΣ}) \text{) } \rightarrow \text{ΕΙΝΑΙ}(x, \text{ΤΙΓΡΗΣ}) .$

$\forall x \text{ (} \text{ΕΙΝΑΙ}(x, \text{ΣΑΡΚΟΒΟΡΟ}) \wedge \text{ΧΡΩΜΑ}(x, \text{ΚΑΦΕ-ΠΟΡΤΟΚΑΛΙ}) \wedge \text{ΕΧΕΙ}(x, \text{ΒΟΥΛΕΣ}(\text{ΜΑΥΡΕΣ}) \text{) } \rightarrow \text{ΕΙΝΑΙ}(x, \text{ΤΣΙΤΑΧ}) .$

$\forall x \text{ (} \text{ΕΙΝΑΙ}(x, \text{ΠΟΥΛΙ}) \wedge (\neg \text{ΠΕΤΑ}(x)) \wedge \text{ΚΟΛΥΜΠΑ}(x) \text{) } \rightarrow \text{ΕΙΝΑΙ}(x, \text{ΠΙΓΚΟΥΪΝΟΣ}) .$

Πλεονεκτήματα – Μειονεκτήματα Κατηγορηματικής Λογικής

❖ Πλεονεκτήματα

- ❑ αντιστοιχία με τη φυσική γλώσσα,
- ❑ ικανοποιητική εκφραστικότητα λόγω της χρήσης ποσοδεικτών (περιγραφή κανόνων)

Διαδικασία εξαγωγής συμπερασμάτων

❖ Βασιζόμαστε στην "*Αρχή της ανάλυσης*" (ή *αναγωγής*) (*resolution principle*).

$$(P \vee Q) \wedge (R \vee \neg Q) \models P \vee R \quad (\text{αρχή της ανάλυσης})$$

χρησιμοποιώντας τη μέθοδο της "*εις άτοπο απαγωγής*" (*refutation*).

❖ Για να αποδείξουμε το A εισάγουμε στη βάση γνώσης το $\neg A$ και, εφαρμόζοντας διαδοχικά την αρχή της ανάλυσης, προσπαθούμε να καταλήξουμε σε αντίφαση (άτοπο) της μορφής ότι καταλήγουμε να υπάρχει στη βάση γνώσης τόσο το γεγονός B όσο και το $\neg B$).

- ❖ Η παραπάνω Δ.Ε.Σ δεν μπορεί να παράγει συνεπαγόμενες προτάσεις, παρά μόνο να ελέγξει εάν μια πρόταση είναι συνεπαγόμενη ή όχι (απόδειξη θεωρημάτων, απάντηση ερωτήσεων).
- ❖ Όταν εφαρμόζουμε την αρχή της ανάλυσης, κατά την ενοποίηση (ταίριασμα) δύο προτάσεων μπορεί να γίνει και αντικατάσταση μεταβλητών με τιμές.
 - π.χ οι προτάσεις $\text{ΕΙΝΑΙ}(x, \text{ΘΗΛΑΣΤΙΚΟ})$ και $\text{ΕΙΝΑΙ}(\text{ΦΑΛΛΑΙΝΑ}, \text{ΘΗΛΑΣΤΙΚΟ})$ ενοποιούνται με την αντικατάσταση $x = \text{ΦΑΛΛΑΙΝΑ}$
- ❖ Για την εφαρμογή της ανωτέρω μεθοδολογίας η γνώση αναπαρίσταται σαν σύζευξη διαζεύξεων (conjunction of disjunctions): **Συζευκτική Κανονική Μορφή (ΣΚΜ)**.
- ❖ Στην ΣΚΜ όλες οι μεταβλητές θεωρούνται ότι είναι καθολικά ποσοτικοποιημένες στην αρχή της πρότασης, γι' αυτό και η ποσοτικοποίηση παραλείπεται.
- ❖ Βασικές ταυτότητες που χρησιμοποιούνται για την μετατροπή σε ΣΚΜ:

$\neg(\neg p) \equiv p$	$\neg(p \wedge q) \equiv \neg p \vee \neg q$	$\neg(p \vee q) \equiv \neg p \wedge \neg q$
$p \Rightarrow q \equiv \neg p \vee q$	$\neg \exists x [p(x)] \equiv \forall x [\neg p(x)]$	$\neg \forall x [p(x)] \equiv \exists x [\neg p(x)]$
$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$		$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$



- ❖ Χρησιμοποιούμε διαφορετικά σύμβολα μεταβλητών σε κάθε πρόταση, δηλ το σύμβολο x εμφανίζεται ως όνομα μεταβλητής σε μια μόνο πρόταση. Εάν εμφανιστεί σε δεύτερη πρόταση του δίνουμε άλλο όνομα (π.χ. το γράφουμε ως y).
- ❖ Πολλές φορές αντί για \forall χρησιμοποιείται κόμμα (,): ($\neg \text{banana}(x), \text{yellow}(x)$)
- ❖ Παραδείγματα: Εστω η βάση γνώσης:

R1: $\forall x \text{ ΠΑΡΑΓΕΙ}(x, \text{ΓΑΛΛΑ}) \rightarrow \text{ΕΙΝΑΙ}(x, \text{ΘΗΛΑΣΤΙΚΟ})$

R2: $\forall x \text{ ΕΧΕΙ}(x, \text{ΦΤΕΡΑ}) \rightarrow \text{ΕΙΝΑΙ}(x, \text{ΠΟΥΛΙ})$

Σε ΣΚΜ η βάση γνώσης γράφεται:

R1: $\neg \text{ΠΑΡΑΓΕΙ}(x, \text{ΓΑΛΛΑ}) \vee \text{ΕΙΝΑΙ}(x, \text{ΘΗΛΑΣΤΙΚΟ})$

R2: $\neg \text{ΕΧΕΙ}(y, \text{ΦΤΕΡΑ}) \vee \text{ΕΙΝΑΙ}(y, \text{ΠΟΥΛΙ})$

- ❖ Να δείξετε ότι η πρόταση $\forall x \forall y [(\text{mother}(y, x) \wedge \text{rich}(x)) \rightarrow \text{happy}(y)]$ γράφεται σε ΣΚΜ ως: $\neg \text{mother}(y, x) \vee \neg \text{rich}(x) \vee \text{happy}(y)$
 - 1) $\neg (\text{mother}(y, x) \wedge \text{rich}(x)) \vee \text{happy}(y)$ ($P \rightarrow Q \leftrightarrow \neg P \vee Q$)
 - 2) $\neg \text{mother}(y, x) \vee \neg \text{rich}(x) \vee \text{happy}(y)$



Παράδειγμα. Εστω η βάση γνώσης:

R1: $\forall x \text{ ΕΧΕΙ}(x, \text{ΦΤΕΡΑ}) \rightarrow \text{ΕΙΝΑΙ}(x, \text{ΠΟΥΛΙ})$

R2: $\text{ΕΧΕΙ}(\text{ΑΕΤΟΣ}, \text{ΦΤΕΡΑ})$

Ερώτηση: $\text{ΕΙΝΑΙ}(\text{ΑΕΤΟΣ}, \text{ΠΟΥΛΙ})$?

Σε ΣΚΜ η βάση γνώσης γράφεται:

R1: $\neg \text{ΕΧΕΙ}(x, \text{ΦΤΕΡΑ}) \vee \text{ΕΙΝΑΙ}(x, \text{ΠΟΥΛΙ})$

R2: $\text{ΕΧΕΙ}(\text{ΑΕΤΟΣ}, \text{ΦΤΕΡΑ})$

Εισάγουμε στη βάση γνώσης την άρνηση της ερώτησης

R3: $\neg \text{ΕΙΝΑΙ}(\text{ΑΕΤΟΣ}, \text{ΠΟΥΛΙ})$

Από τις R1 και R3 με ενοποίηση και αντικατάσταση ($\text{ΑΕΤΟΣ}/x$) προκύπτει ότι

R4: $\neg \text{ΕΧΕΙ}(\text{ΑΕΤΟΣ}, \text{ΦΤΕΡΑ})$

Η R4 είναι αντιφατική με την πρόταση R2.

Αρα η πρόταση $\text{ΕΙΝΑΙ}(\text{ΑΕΤΟΣ}, \text{ΠΟΥΛΙ})$ είναι TRUE.



Παραδείγματα δέντρων αναγωγής (resolution tree)

1. $\text{is_man}(\text{socrates})$

2. $\neg \text{is_man}(X) \vee \text{is_mortal}(X)$

3. $\neg \text{is_mortal}(\text{socrates})$

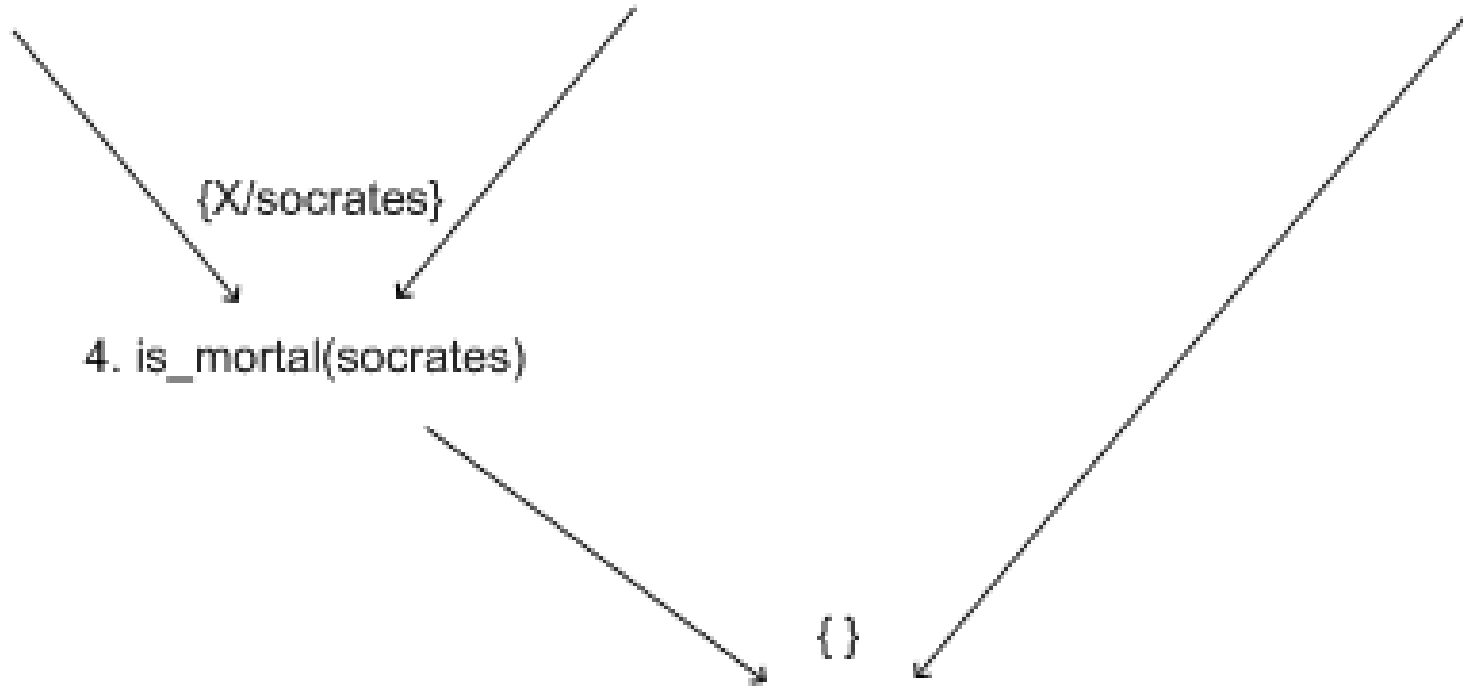
$\{X/\text{socrates}\}$

4. $\text{is_mortal}(\text{socrates})$

$\{\}$

5. False

X: μεταβλητή





Παράδειγμα αναγωγής

Βάση γνώσης ($\Delta(x)$, $E(x)$, $\Phi(x)$, $N(x)$ κατηγορήματα, A σταθερά)

- 1) $\forall x \Delta(x) \rightarrow E(x)$
- 2) $\forall x \Phi(x) \rightarrow \sim E(x)$
- 3) $\forall x N(x) \rightarrow \Delta(x)$
- 4) $\Phi(A)$

Να αποδείξετε ότι $\sim N(A)$.

Απάντηση: Μετατροπή σε ΣΚΜ (προσέξτε ότι χρησιμοποιούνται διαφορετικά ονόματα μεταβλητών σε κάθε πρόταση)

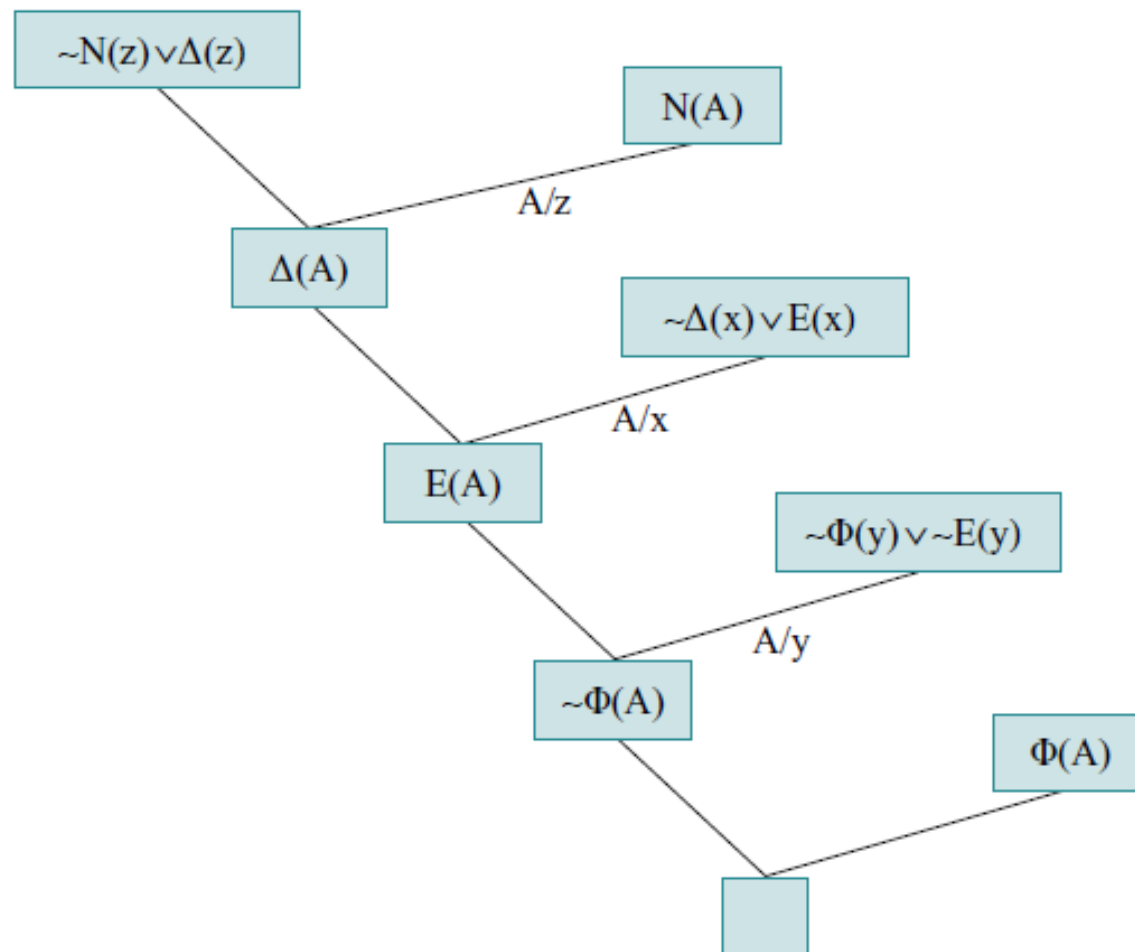
- 1) $\sim \Delta(x) \vee E(x)$
- 2) $\sim \Phi(y) \vee \sim E(y)$
- 3) $\sim N(z) \vee \Delta(z)$
- 4) $\Phi(A)$

Εισάγουμε στη βάση γνώσης το $N(A)$ (άρνηση της ερώτησης $\sim N(A)$).

- 5) $N(A)$



Διαδικασία αναγωγής



Καταλήγουμε σε άτοπο: $\Phi(A) \wedge \sim \Phi(A)$. Άρα η ερώτηση $\sim N(A)$ είναι αληθής.

Προτάσεις Horn

- ❖ Οι προτάσεις Horn είναι υποσύνολο της λογικής πρώτης τάξης και έχουν την εξής μορφή:

$$(\text{σώμα}) \quad Q_1 \wedge Q_2 \wedge \dots \wedge Q_n \rightarrow R \text{ (κεφαλή)}$$

- ❖ Μοναδικό άτομο R στην κεφαλή, απαγορεύονται οι αρνήσεις στο σώμα.
- ❖ Ευκολότερη διαδικασία απάντησης ερωτήσεων (SLD resolution)
- ❖ Η γλώσσα προγραμματισμού **Prolog** χρησιμοποιεί τις προτάσεις Horn σαν μέθοδο αναπαράστασης.

Χαρακτηριστικά της Prolog

- ❖ Η πιο διαδεδομένη γλώσσα λογικού προγραμματισμού
- ❖ Χρησιμοποιεί τις **προτάσεις Horn** σαν μέθοδο αναπαράστασης.
- ❖ Ένα πρόγραμμα αποτελείται από ακολουθία προτάσεων
- ❖ Κάθε πρόταση είναι i) είτε ατομική ii) είτε κανόνας τύπου Horn.
- ❖ Όλες οι μεταβλητές έχουν **καθολική ποσοτικοποίηση** και μεταβλητές με το ίδιο όνομα σε διαφορετικές προτάσεις θεωρούνται ξεχωριστές.
- ❖ Οι ερωτήσεις μπορούν να περιλαμβάνουν μεταβλητές
- ❖ Υλοποίηση του μηχανισμού απάντησης ερωτήσεων (SLD resolution):
 - Ξεκινώντας από την ερώτηση εφαρμόζουμε αιτιολόγηση προς τα πίσω (ενοποίηση και αντικατάσταση) ακολουθώντας αναζήτηση σε βάθος
 - Οι προτάσεις εξετάζονται για ενοποίηση με τη σειρά εμφάνισης στο πρόγραμμα (γι' αυτό βάζουμε πρώτα τα γεγονότα).
 - Οι συνθήκες στο σώμα ενός κανόνα εξετάζονται για ενοποίηση από αριστερά προς τα δεξιά.



Παράδειγμα: Έστω το παρακάτω πρόγραμμα Prolog (τα q, r, p, s κατηγορήματα, τα a, b, c, d σταθερές και τα X, Y, Z, W μεταβλητές):

q(a).

q(b).

q(c).

r(b).

r(c).

r(d).

p(X,Y) :- q(X), r(Y).

s(Z) :- p(Z,Z).

και ότι θέτουμε το **ερώτημα: ?- s(W)**. Ποιες απαντήσεις θα δώσει η Prolog στο ερώτημα και με ποια σειρά; Να αιτιολογήσετε την απάντησή σας.

Απάντηση:

Στόχος S(W) -> νέος στόχος p(W,W) -> νέος στόχοι q(W), r(W).

- q(W) αληθές για W=a, εξετάζεται ο στόχος r(a) που ΔΕΝ είναι αληθής.
- q(W) αληθές για W=b, εξετάζεται ο στόχος r(b) που είναι αληθής -> **W=b**
- q(W) αληθές για W=c, εξετάζεται ο στόχος r(c) που είναι αληθής -> **W=c**

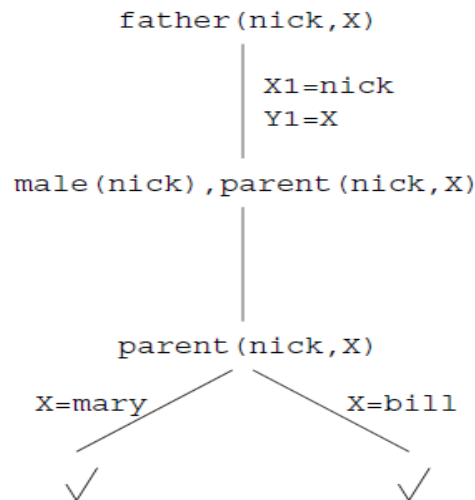
Αρα η Prolog θα απαντήσει **W=b ; W=c με τη συγκεκριμένη σειρά.**

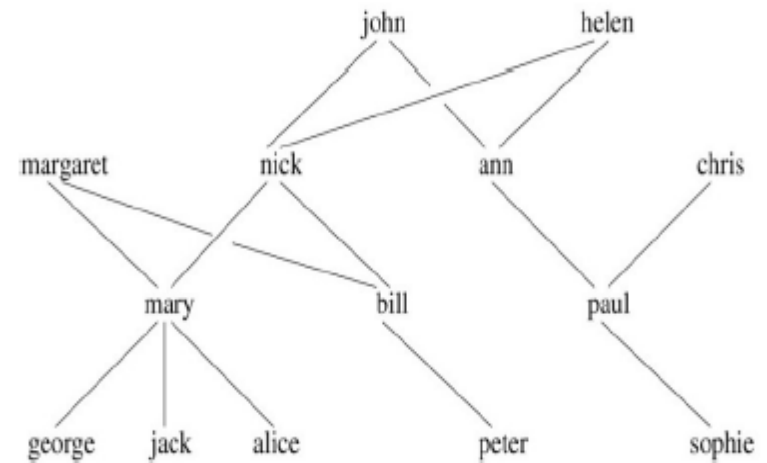
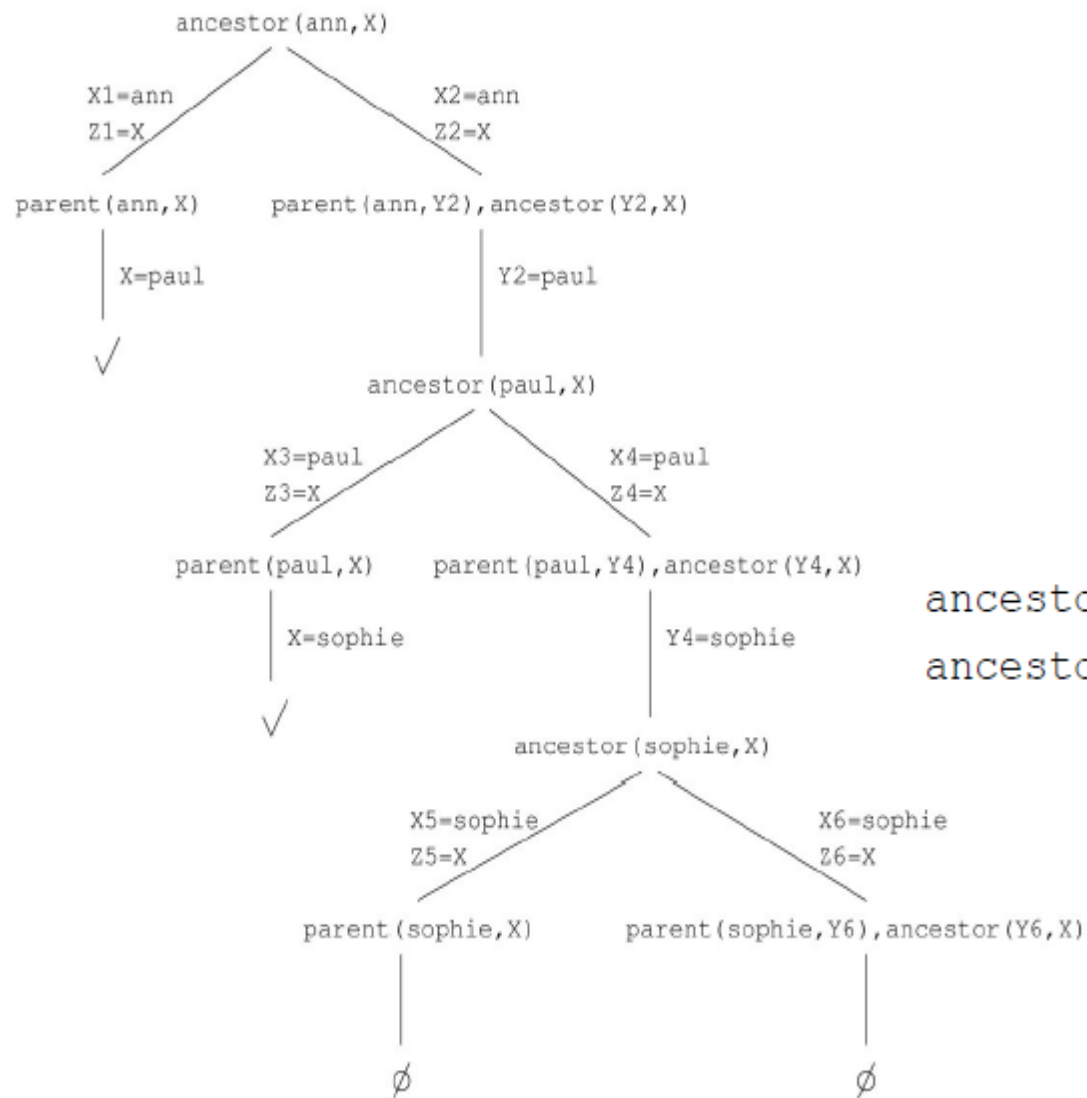
parent(john, nick).	parent(john, ann).	male(john).	male(nick).
parent(helen, nick).	parent(helen, ann).	male(bill).	male(george).
parent(nick, mary).	parent(nick, bill).	male(jack).	male(peter).
parent(margaret, mary).	parent(margaret, bill).	male(chris).	male(paul).
parent(mary, george).	parent(mary, jack).		
parent(mary, alice).	parent(bill, peter).	female(helen).	female(margaret).
parent(chris, paul).	parent(ann, paul).	female(mary).	female(alice).
parent(paul, sophie).		female(ann).	female(sophie).

father(X1, Y1) :- male(X1), parent(X1, Y1).

Ποια είναι τα παιδιά του Νίκου;

?- father(nick, X).





```
ancestor(X,Z) :- parent(X,Z) .  
ancestor(X,Z) :- parent(X,Y) , ancestor(Y,Z) .
```



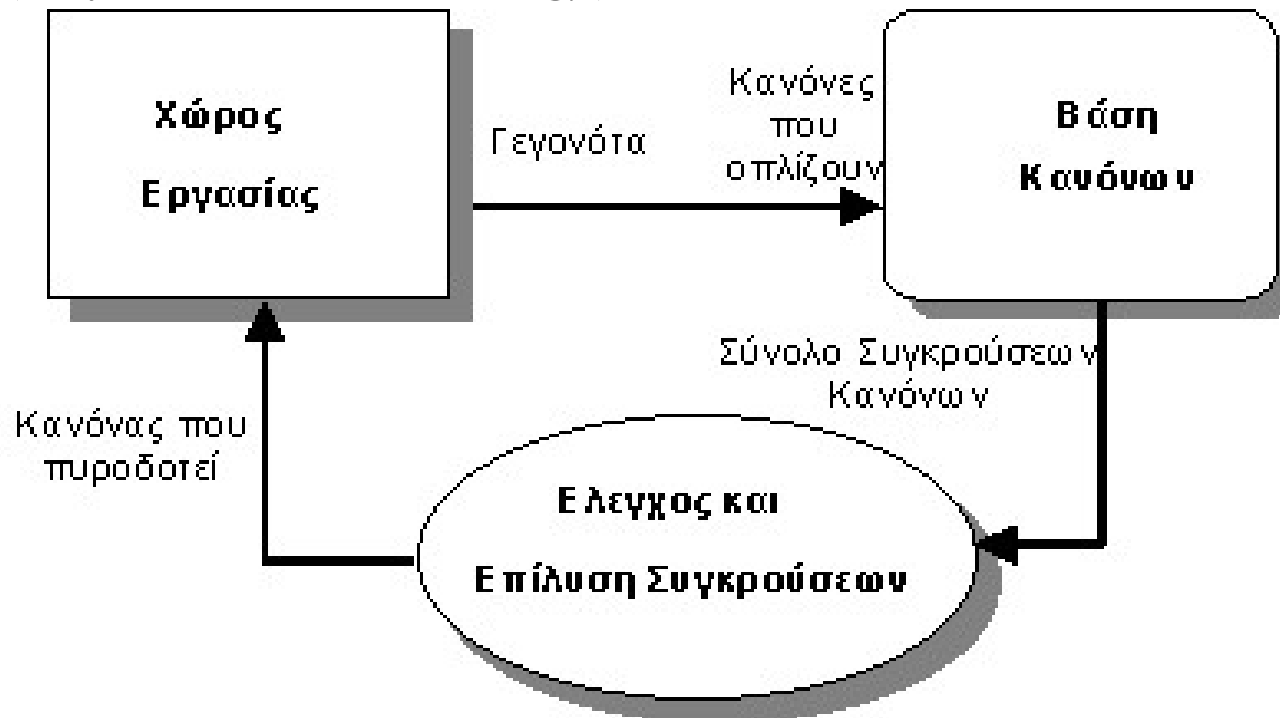
Συστήματα Παραγωγής (γνώσης) με αιτιολόγηση προς τα εμπρός (forward chaining)

Μορφές Κανόνων	Εκφράζει	Επεξήγηση
IF <i>συνθήκες</i> THEN <i>ενέργειες</i>	Διαδικαστική γνώση	Αν οι <i>συνθήκες</i> αληθεύουν τότε εκτέλεσε τις <i>ενέργειες</i>
IF <i>συνθήκες</i> THEN <i>συμπέρασμα</i>	Δηλωτική γνώση	Αν οι <i>συνθήκες</i> αληθεύουν τότε αληθεύει και το <i>συμπέρασμα</i>

- ❖ Αν ικανοποιούνται οι συνθήκες ενός κανόνα τότε εξάγεται κάποιο συμπέρασμα (νέα γνώση) ή εκτελείται κάποια ενέργεια
- ❖ Δεν υπάρχουν ερωτήσεις
- ❖ Σε κάθε βήμα εκτελούνται κανόνες που προσθέτουν (ή αφαιρούν) προτάσεις και η διαδικασία συνεχίζεται μέχρι να ικανοποιηθεί κάποιο κριτήριο τερματισμού.

Αρχιτεκτονική Συστημάτων Παραγωγής

- ❖ Ένα σύστημα παραγωγής (production system) αποτελείται από τρία μέρη:
 - ❑ Τη βάση (μνήμη) κανόνων.
 - ❑ Τη μνήμη γεγονότων (ή μνήμη εργασίας)
 - ❑ Το μηχανισμό ελέγχου (*control* ή *scheduler*), ο οποίος υλοποιεί μία στρατηγική επίλυσης συγκρούσεων (*conflict resolution strategy*).





Κύκλος λειτουργίας ενός Συστήματος Παραγωγής

Έως ότου δε μπορεί να εκτελεστεί κανένας κανόνας επανέλαβε:

1. Βρες όλους του κανόνες που οπλίζουν (ενεργοί κανόνες), δηλαδή οι συνθήκες τους ικανοποιούνται από τα γεγονότα στη μνήμη γεγονότων (φάση ταιριάσματος)
2. Σύμφωνα με το μηχανισμό επίλυσης συγκρούσεων, διάλεξε έναν (συνήθως) ενεργό κανόνα (φάση επίλυσης συγκρούσεων).
3. Πυροδότησε (εκτέλεσε) τον κανόνα που διάλεξες στο βήμα 2 (φάση ενεργειών).

Επίλυση Συγκρούσεων

- ❖ Μερικές από τις πιο γνωστές στρατηγικές επίλυσης συγκρούσεων είναι οι εξής:
 - ☐ Τυχαία (random).
 - ☐ Διάταξης (ordering) με βάση την εμφάνιση στη βάση γνώση ή με βάση προγραμματιζόμενες προτεραιότητες (rule saliency)
 - ☐ Επιλογή του κανόνα που ενεργοποιήθηκε πιο πρόσφατα (recency).
 - ☐ Επιλογή του πιο παλιού ενεργοποιημένου κανόνα
 - ☐ Αποφυγή επανάληψης (refractoriness).



Εφαρμογές Συστημάτων Παραγωγής (expert systems)

- ❖ Πολλές εφαρμογές σε έλεγχο, σχεδίαση, συστήματα υποστήριξης αποφάσεων (π.χ. υποβοήθηση ιατρικής διάγνωσης, αυτόματος πιλότος, έλεγχος αντιδραστήρων, διαλογικά συστήματα, επιχειρηματικές αποφάσεις, χρηματοοικονομικά εργαλεία κλπ).
- ❖ Ομοιότητα με τον τρόπο που λειτουργούμε στις καθημερινές μας δραστηριότητες
- ❖ Ρομποτικά συστήματα: θέση-κατάσταση (από σένσορες) → απόφαση ενέργειας (π.χ. κίνηση) → εκτέλεση ενέργειας → νέα θέση-κατάσταση κλπ.