

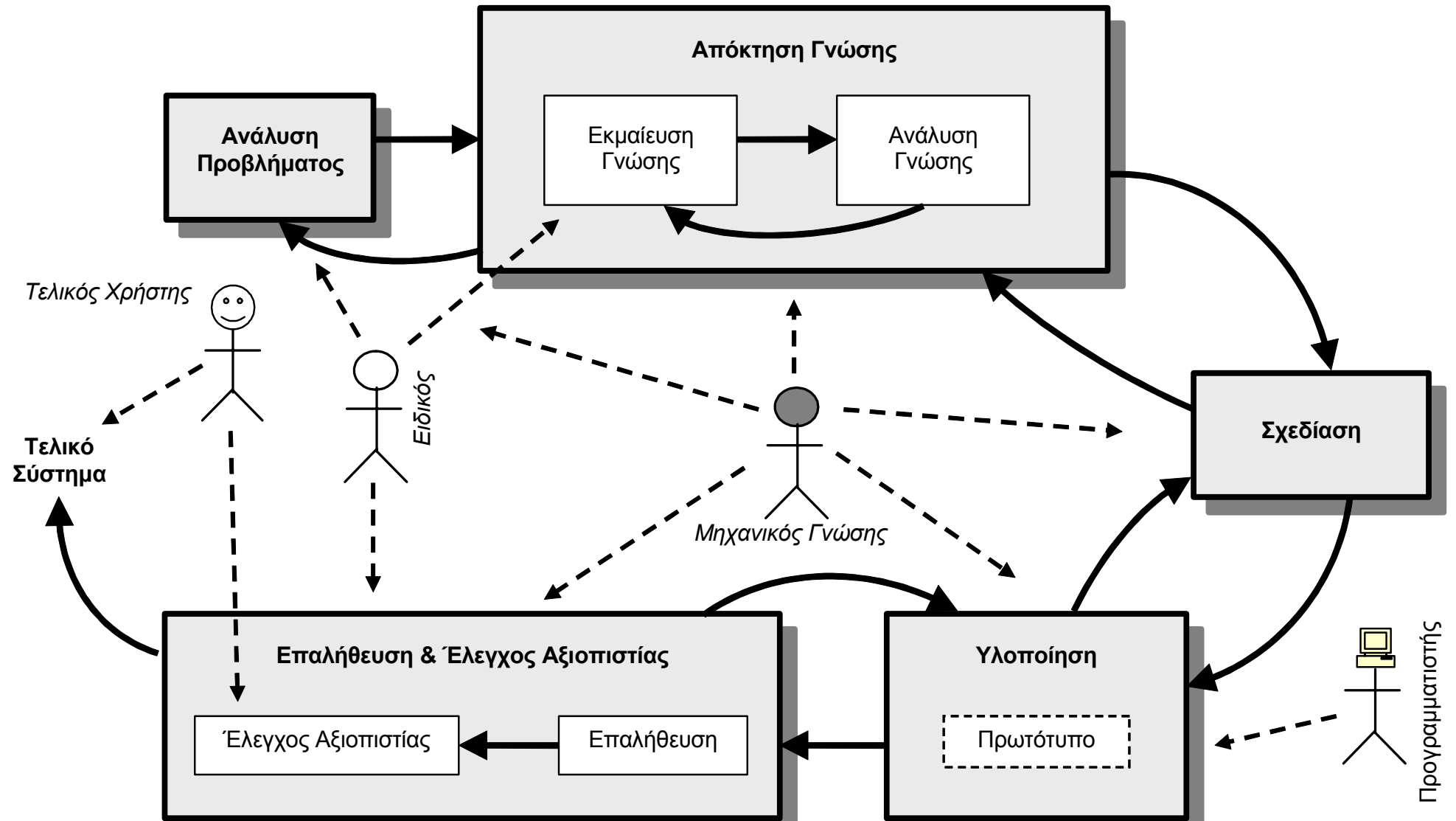
Κεφάλαιο 23

Τεχνολογία Γνώσης

Τεχνητή Νοημοσύνη - Β' Έκδοση

Ι. Βλαχάβας, Π. Κεφαλάς, Ν. Βασιλειάδης, Φ. Κόκκορας, Η. Σακελλαρίου

Διαδικασία Ανάπτυξης Συστημάτων Γνώσης





Ανάλυση Προβλήματος

- ❖ Προσδιορίζεται η επιθυμητή λύση του προβλήματος
- ❖ Κυριότερα ζητήματα:
 - ❑ Είναι το πρόβλημα **κατάλληλο** για επίλυση από σύστημα γνώσης ή συμβατικό πρόγραμμα;
 - ❑ Υπάρχουν έτοιμες μελέτες περιπτώσεων επίλυσης του προβλήματος (**case-studies**);
 - ❑ Ποια είναι τα **οφέλη** από την κατασκευή του συστήματος γνώσης;

Απόκτηση της Γνώσης (*knowledge acquisition*)

- ❖ Ο μηχανικός της γνώσης
 - ❑ Εκμαιεύει από τον ειδικό τη γνώση του πάνω στο πρόβλημα (*knowledge elicitation*)
 - ❑ Μοντελοποιεί (*knowledge analysis & modeling*) τη γνώση σε κάποια ενδιάμεση μορφή αναπαράστασης
- ❖ Εκμαίευση γνώσης (**knowledge elicitation**)
 - ❑ Απαιτεί συνεχή επικοινωνία ανάμεσα στο μηχανικό γνώσης και τον ειδικό.
 - ❑ Εκτός από τις κλασικές μεθόδους, υπάρχουν:
 - **Ημι-αυτόματες μέθοδοι:** π.χ. TEIRESIAS, OPAL, κλπ
 - **Αυτόματες μέθοδοι:** τεχνικές μηχανικής μάθησης
- ❖ Η απόκτηση γνώσης και η παρουσία ειδικού είναι απαραίτητη ακόμα και όταν η γνώση δεν είναι εμπειρική.
 - ❑ *Συλλογιστική των μοντέλων:* Ο μηχανικός γνώσης δεν είναι πάντα δυνατό να μπορεί να ερμηνεύσει τα εγχειρίδια επιστημονικής γνώσης.
 - ❑ *Συλλογιστική των περιπτώσεων:* Ο ειδικός καθορίζει τη σπουδαιότητα των χαρακτηριστικών, τη μέθοδο δεικτοδότησης, τη μέθοδο προσαρμογής των λύσεων, κλπ.



Μοντελοποίηση Γνώσης

Knowledge Analysis & Modeling

- ❖ **Ανάλυση** της γνώσης από το μηχανικό με σκοπό τη δημιουργία ενός **μοντέλου** της γνώσης.
 - ❑ Η αναπαράσταση της γνώσης γίνεται με διάφορες ημιδομημένες μορφές αναπαράστασης.
- ❖ Υπάρχουν μεθοδολογίες που τυποποιούν τη μοντελοποίηση της γνώσης (KADS).
- ❖ Το μοντέλο βοηθά στο να αποκαλυφθούν ατέλειες, ασάφειες και ελλείψεις στη γνώση

Σχεδίαση

- ❖ Προσδιορίζονται:
 - ❑ Η μορφή της αναπαράστασης της γνώσης και η συλλογιστική
 - ❑ Το εργαλείο για την ανάπτυξη του συστήματος γνώσης.
- ❖ Παράγεται η αρχιτεκτονική του συστήματος
 - ❑ Λειτουργικές υπομονάδες (modules) του συστήματος
 - ❑ Λειτουργικότητα της κάθε υπομονάδας
 - ❑ Αλληλεξαρτήσεις υπομονάδων



Υλοποίηση

- ❖ Κωδικοποιείται το μοντέλο της γνώσης χρησιμοποιώντας εργαλεία ανάπτυξης
- ❖ Αρχικά αναπτύσσεται ένα πρωτότυπο σύστημα.
 - ❑ Επιδεικνύεται στον ειδικό και σε μία μικρή ομάδα χρηστών
 - ❑ Καθοδηγεί στη συνέχεια την ανάπτυξη, ή
 - ❑ Οδηγεί σε επανασχεδιασμό όταν δεν ικανοποιεί τις απαιτήσεις που τέθηκαν στην αρχή.
 - ❑ Επαληθεύει τη γνώση που αποκτήθηκε από τον ειδικό και μοντελοποιήθηκε από το μηχανικό γνώσης.
- ❖ Το βάθος της γνώσης πρέπει να είναι μεγάλο.
 - ❑ Πρέπει να μπορεί επιλύσει πλήρως μερικά από τα προβλήματα για τα οποία προορίζεται το σύστημα γνώσης.
- ❖ Το εύρος της γνώσης δε χρειάζεται να είναι μεγάλο.
 - ❑ Δεν είναι αναγκαίο να αντιμετωπίζει πολλές περιπτώσεις.



Επαλήθευση και Έλεγχος Αξιοπιστίας

❖ Επαλήθευση (verification):

- ❑ Έλεγχος της συμβατότητας του συστήματος με τις αρχικές προδιαγραφές.
- ❑ Επιβεβαίωση της συνέπειας και πληρότητας της κωδικοποίησης της γνώσης που περιέχεται στο σύστημα

❖ Έλεγχος αξιοπιστίας (validation):

- ❑ Επιβεβαίωση της ορθότητας και γενικότητας της γνώσης που περιέχει το σύστημα
- ❑ Το σύστημα επιλύει ένα σύνολο από υποδειγματικές περιπτώσεις (test cases).
 - Διαφορετικές από αυτές που χρησιμοποιήθηκαν στην ανάπτυξης του συστήματος
 - Εξασφαλίζεται η **ευρωστία (robustness)** σε μη-προσδοκώμενα δεδομένα
- ❑ Οι λύσεις συγκρίνονται με λύσεις που δόθηκαν από διάφορους ειδικούς του τομέα.



Μεθοδολογία Ανάλυσης Προβλήματος KADS

Knowledge Acquisition and Domain Structuring

- ❖ Σε πολύπλοκα προβλήματα η ανάλυση χρειάζεται να ακολουθήσει "πεπατημένες οδούς" γιατί αλλιώς ο μηχανικός γνώσης κινδυνεύει να "χαθεί" μέσα στην πολυπλοκότητα
- ❖ Η μεθοδολογία KADS προσφέρει καθοδήγηση:
 - ❑ στην απόκτηση (κυρίως εμπειρικής) γνώσης
 - ❑ στην ανάλυση της γνώσης (κυριότερος στόχος του KADS)
 - ❑ στην μετατροπή της γνώσης σε λεπτομερές σχέδιο-μοντέλο για την υλοποίηση ενός έμπειρου συστήματος



Οι Βασικές Αρχές του KADS

- ❖ Η γνώση πρέπει να αναλύεται πριν από τη σχεδίαση και την υλοποίηση
- ❖ Η ανάλυση της γνώσης πρέπει να οδηγεί από την αρχή σε ανάπτυξη "μοντέλων της γνώσης"
 - ❑ Δομημένες περιγραφές του πεδίου του προβλήματος
 - ❑ Τα μοντέλα βοηθούν στην ευκολότερη κατανόηση των επόμενων σταδίων της εκμείευσης
- ❖ Το μοντέλο πρέπει να είναι εκφρασμένο σε επιστημολογικό επίπεδο
 - ❑ Η γνώση πρέπει να αναπαρασταθεί με μία κατάλληλη ενδιάμεση μορφή αναπαράστασης
 - ❑ Όχι απευθείας στην τελική μορφή αναπαράστασης γνώσης, γιατί ακόμα δεν έχει αποφασιστεί
- ❖ Η ανάλυση θα πρέπει να καλύπτει την λειτουργικότητα του τελικού συστήματος
 - ❑ Ποιος θα χρησιμοποιεί το σύστημα και υπό ποιες συνθήκες
- ❖ Η ανάλυση θα πρέπει να προχωράει σταδιακά
- ❖ Τα δεδομένα και η γνώση που αποκτούνται πρέπει να αναλύονται και μετά να προχωράει η διαδικασία στην απόκτηση νέων

Επίπεδα Αφαίρεσης του KADS

- ❖ Το KADS χρησιμοποιεί γενικά (ή εξιδανικευμένα) μοντέλα της γνώσης για να καθοδηγήσει την ανάπτυξη ενός συστήματος
- ❖ Υπάρχουν 4 επίπεδα αφαίρεσης των μοντέλων
- ❖ *Επίπεδο πεδίου γνώσης (domain layer)*
 - Αντικείμενα του προβλήματος
 - Έννοιες, στοιχεία και σχέσεις μεταξύ τους
- ❖ *Επίπεδο εξαγωγής συμπερασμάτων (inference layer)*
 - Οργάνωση της γνώσης σε επίπεδο αντικειμένων
 - Τρόπος χρήσης αντικειμένων για την επίλυση προβλημάτων.
 - Δηλωτική φύση
- ❖ *Επίπεδο διεργασιών (task layer)*
 - Τρόποι εκτέλεσης στόχων-υποστόχων του προβλήματος ώστε αυτό να λυθεί
 - Διαδικαστική φύση
- ❖ *Επίπεδο στρατηγικής (strategy layer)*
 - Έλεγχος της εκτέλεσης των διεργασιών επίλυσης προβλημάτων

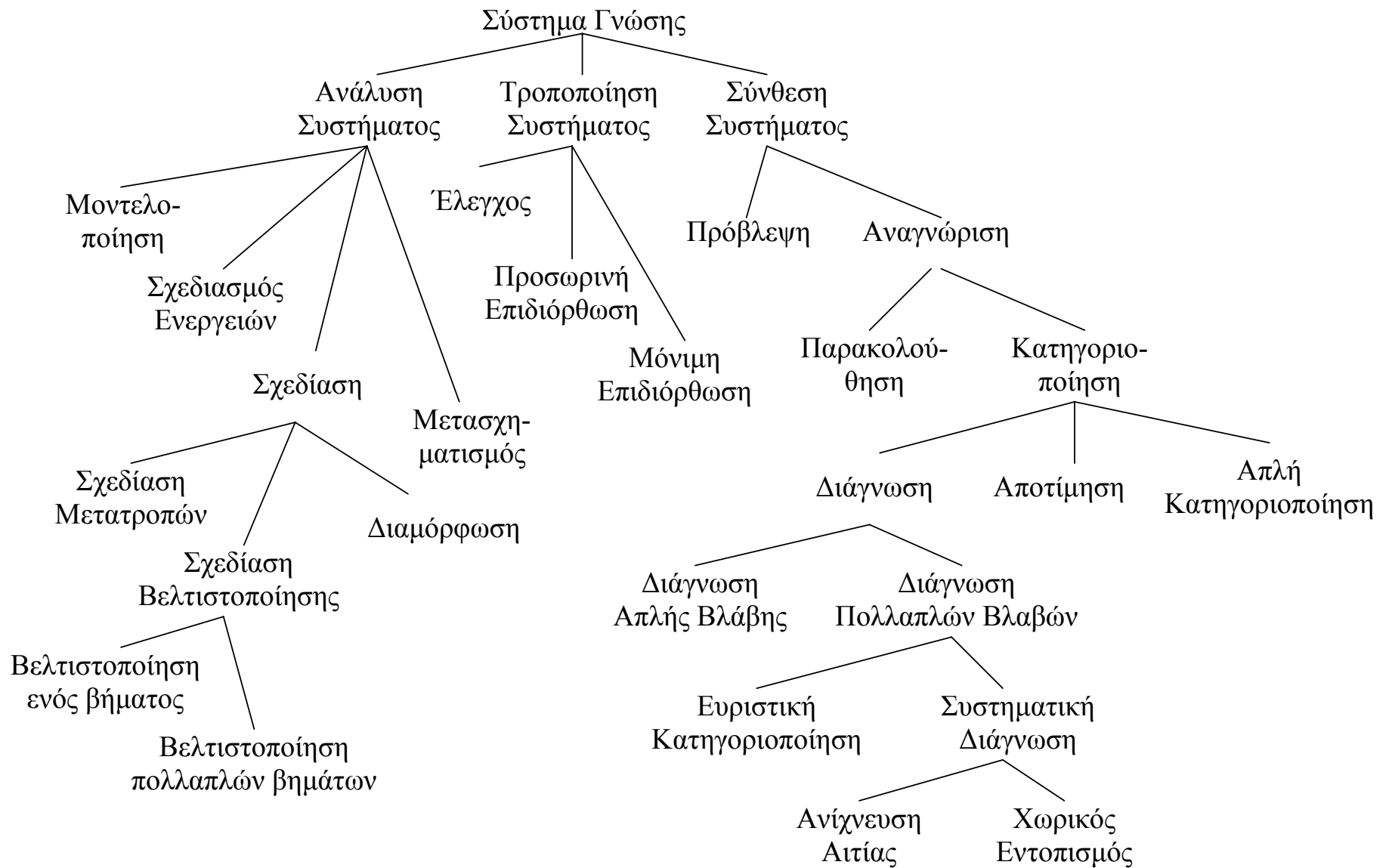


Ερμηνευτικά Μοντέλα

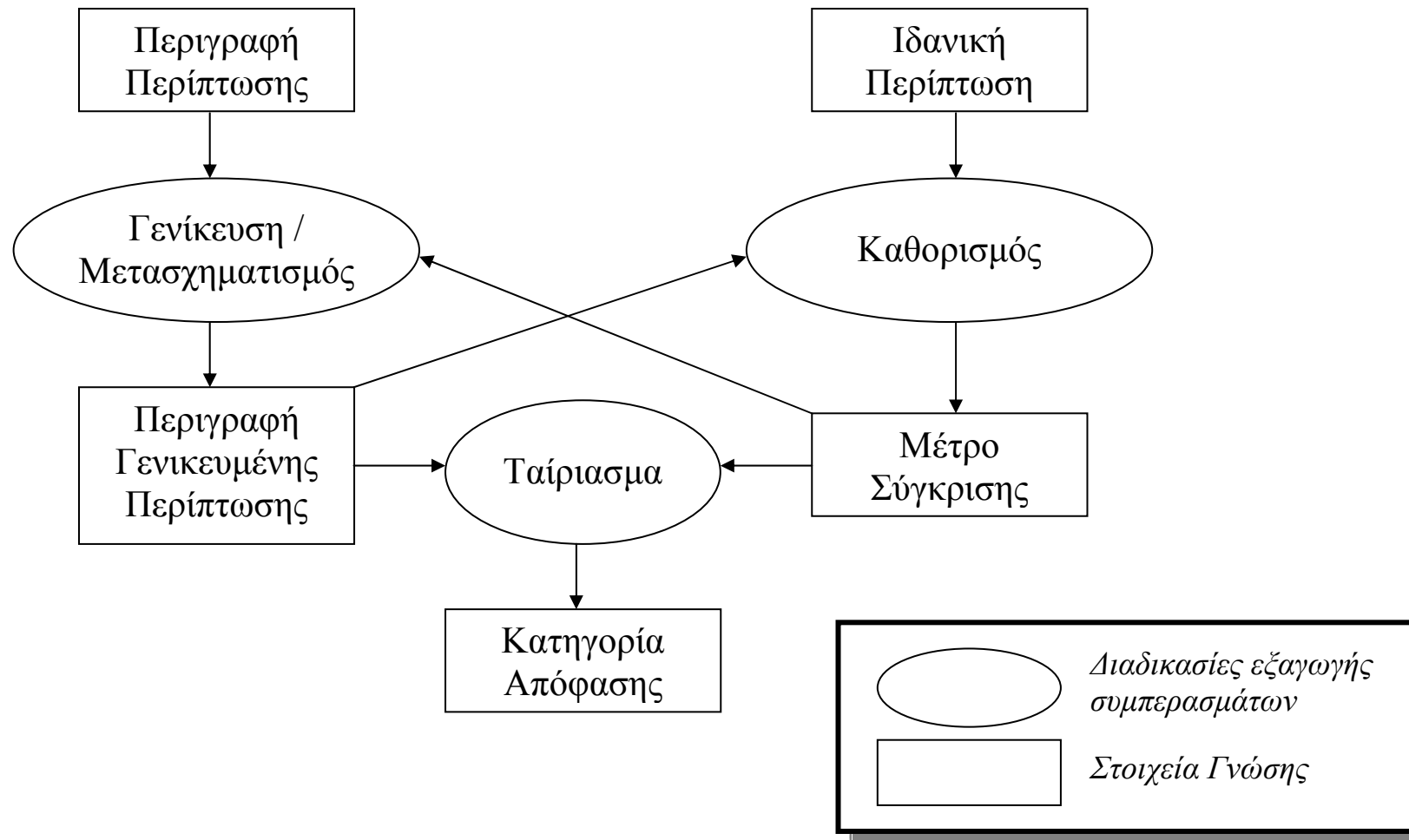
Interpretation Models

- ❖ Βιβλιοθήκη από δομημένες περιγραφές διαφόρων τύπων διεργασιών που μπορεί να επιτελεί ένα σύστημα
 - ❑ Π.χ. κατηγοριοποίηση, διαμόρφωση, διάγνωση, κλπ.
 - ❑ Μοντέλα γενικής χρήσης και επαναχρησιμοποιήσιμα
 - ❑ Για κάθε είδος διεργασίας, περιγράφουν ποια στοιχεία γνώσης και ποιες συμπερασματικές διαδικασίες είναι απαραίτητες
- ❖ Η μεθοδολογία KADS συστήνει την επιλογή και χρήση κάποιου τέτοιου μοντέλου, ανάλογα με το είδος του πεδίου της γνώσης
- ❖ Το KADS έχει μια ιεραρχία διεργασιών
 - ❑ Τα γενικά ερμηνευτικά μοντέλα προσαρμόζονται σταδιακά και γίνονται όλο και πιο συγκεκριμένα
 - ❑ Διάφορα ερμηνευτικά μοντέλα μπορούν να συνδυαστούν μεταξύ τους

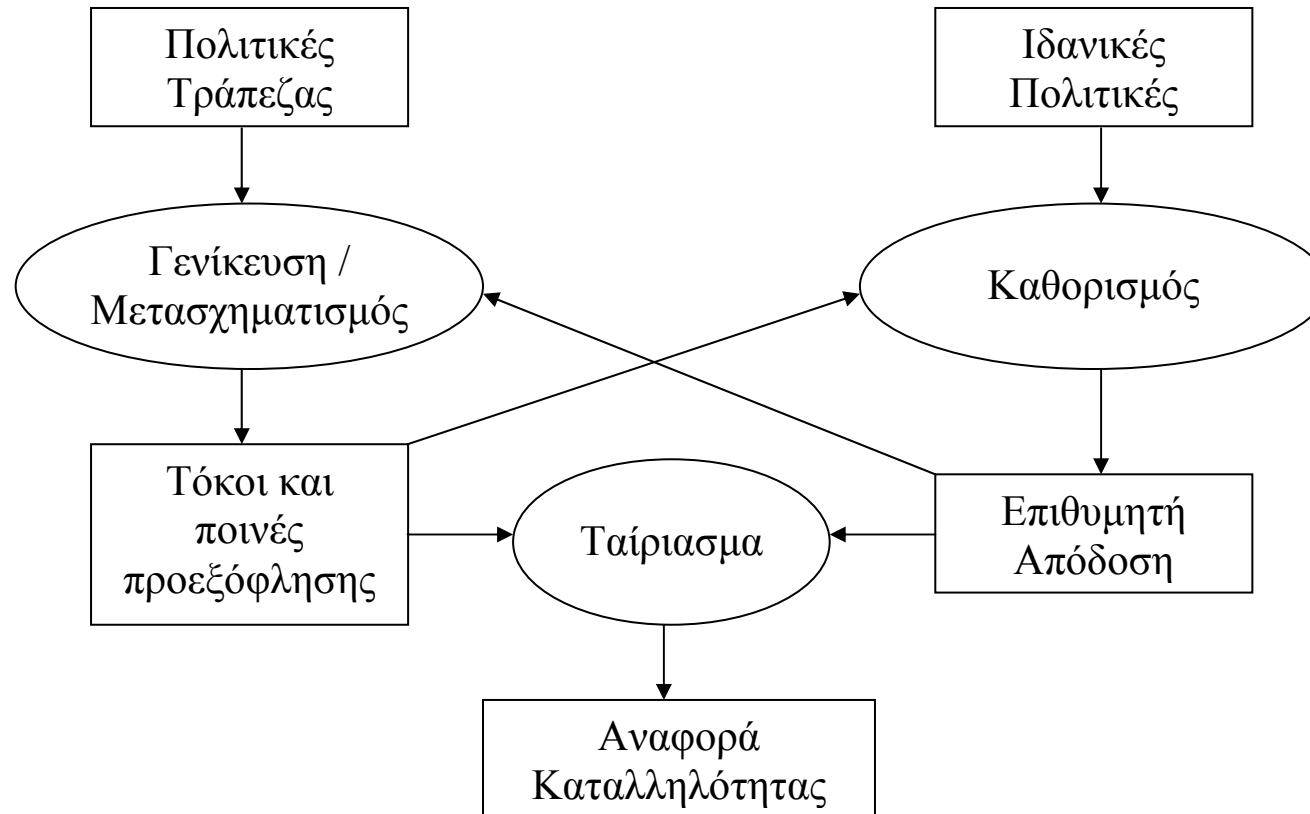
Η Ιεραρχία Διεργασιών του KADS



Ερμηνευτικό Μοντέλο Αποτίμησης



Ερμηνευτικό Μοντέλο για Αποτίμηση Επένδυσης





Εκμαίευση Γνώσης

Knowledge Elicitation

- ❖ Διαδικασία απόκτησης (εκμαίευσης) της γνώσης από άτομα που θεωρούνται "ειδικοί" στο συγκεκριμένο τομέα γνώσης (*domain experts*).
 - ❑ Π.χ. επιστήμονες, τεχνικοί ή εμπειρογνώμονες.
 - ❑ Θεωρείται το πιο δύσκολο-αμφίβολο βήμα στην ανάπτυξη ενός συστήματος γνώσης.
- ❖ **Ειδικός** είναι το άτομο που έχει ειδική γνώση ή ικανότητα πάνω σε ένα θέμα.
- ❖ **Γνώση:** Κατανόηση του κόσμου η οποία αποκτάται μέσω εμπειρίας ή μελέτης.
 - ❑ Πληροφορίες, εμπειρίες, ικανότητες, δεξιότητες
 - ❑ Είδη γνώσης:
 - Αντικείμενα, γεγονότα, διαδικασίες, κτλ.,
 - Βαθιά-επιφανειακή γνώση (deep-shallow knowledge).
 - Ρητή και άρρητη γνώση (explicit-tacit knowledge)



Προβλήματα στην Εκμαίευση της Γνώσης

- ❖ **Παράδοξο της ειδίκευσης**
 - ❑ Όσο πιο πολύ ισχυρίζεται κάποιος ότι είναι ειδικός σε κάποιο θέμα, τόσο πιο δύσκολη είναι η ανταλλαγή πληροφοριών μαζί του.
- ❖ **Ευσεβής πόθος (wishful thinking)**
 - ❑ Ο ειδικός εκφράζει το **τι θα έπρεπε να γίνεται** και όχι το **τι πραγματικά γίνεται**.
- ❖ **Κατάλληλο υπόβαθρο γνώσης του μηχανικού γνώσης.**
- ❖ **Έλλειψη χρόνου**
- ❖ Ο μηχανικός γνώσης πρέπει να διασφαλίσει ότι ικανοποιούνται οι **στόχοι** της **συνέντευξης**.
- ❖ **Αμεροληψία** του μηχανικού γνώσης
- ❖ **Ανεπιτήδειος έμπειρος (inexpert expert)**
- ❖ **Απροθυμία** του ειδικού να μεταδώσει γνώση
- ❖ Ο ειδικός μπορεί να μην είναι εξοικειωμένος στη διαδικασία της εκμαίευσης γνώσης
- ❖ Αδυναμία του ειδικού να **θυμηθεί** ακριβώς τη **ροή** των γεγονότων σε παλιές περιπτώσεις
- ❖ **Επικοινωνιακά προβλήματα** του ειδικού ή/και του μηχανικού γνώσης
- ❖ **Υποκειμενικότητα** του ειδικού

Χρήση ενός μόνο Ειδικού

Πλεονεκτήματα	Μειονεκτήματα
<ul style="list-style-type: none">❖ Προτιμητέο όταν αναπτύσσεται ένα απλό σύστημα ΔΓ<ul style="list-style-type: none">❑ Το πεδίο της γνώσης του προβλήματος είναι περιορισμένο	<ul style="list-style-type: none">❖ Πολλές φορές τα προβλήματα απαιτούν πολλαπλούς τομείς γνώσης για να επιλυθούν<ul style="list-style-type: none">❑ Ένας ειδικός δεν μπορεί να έχει εμπειρία σε όλα
<ul style="list-style-type: none">❖ Διευκολύνεται ο χρονοπρογραμματισμός των συναντήσεων	<ul style="list-style-type: none">❖ Όταν υπάρχει ένας μόνο ειδικός είναι πιο πιθανό να αναβάλλει τις συναντήσεις, παρά όταν είναι πολλοί
<ul style="list-style-type: none">❖ Οι ασυνέπειες και οι αντιφάσεις στη γνώση είναι ευκολότερο να αντιμετωπιστούν	<ul style="list-style-type: none">❖ Ένας μόνο ειδικός → μία μοναδική συλλογιστική οδός<ul style="list-style-type: none">❑ Αυθεντία: Η γνώμη του ειδικού ακολουθείται «τυφλά»❑ Το πεδίο της γνώσης δεν μπορεί να εξεταστεί σε βάθος
<ul style="list-style-type: none">❖ Με λίγα άτομα παρόντα, ο ειδικός «ανοίγεται» πιο εύκολα για να δώσει τη γνώση του	<ul style="list-style-type: none">❖ Αν ο ειδικός δεν έχει επικοινωνιακές ικανότητες τότε η απόκτηση της γνώσης δυσχεραίνει

Χρήση πολλών Ειδικών

Πλεονεκτήματα	Μειονεκτήματα
<ul style="list-style-type: none">❖ Τα πολύπλοκα προβλήματα απαιτούν πολλές ειδικότητες για να λυθούν	<ul style="list-style-type: none">❖ Δυσκολίες συντονισμού❖ Πολλές φορές απαιτείται να υπάρχουν περισσότεροι μηχανικοί γνώσης
<ul style="list-style-type: none">❖ Όσο πιο πολλές και διαφορετικές γνώμες ακούει ο μηχανικός της γνώσης, τόσο πιο πολύ μπορεί να εμβαθύνει στο πεδίο της γνώσης γιατί μπορεί να το «δει» από πολλές σκοπιές	<ul style="list-style-type: none">❖ Ο μηχανικός γνώσης μπορεί να αποπροσανατολιστεί από τις διαφορετικές απόψεις
<ul style="list-style-type: none">❖ Η συνεργασία και ο διάλογος πολλών ειδικών μπορεί να δημιουργήσει γνώση	<ul style="list-style-type: none">❖ Συχνά υπάρχουν διαφωνίες
<ul style="list-style-type: none">❖ Οι επίσημες συναντήσεις είναι πολλές φορές πρόκληση για προσφορά και δημιουργικότητα εκ μέρους των ειδικών (ανταγωνισμός)	<ul style="list-style-type: none">❖ Θέματα εμπιστευτικότητας



Μεθοδολογίες Εκμαίευσης Γνώσης

Συνέντευξη

- ❖ Ο πιο διαδομένος και αποδοτικός τρόπος εκμαίευσης γνώσης
 - ❑ Χρησιμοποιούνται συνήθως στα αρχικά στάδια απόκτησης της γνώσης
 - ❑ Απαιτεί εκτεταμένη προετοιμασία και εξάσκηση από την πλευρά του μηχανικού
 - ❑ Είναι χρονοβόρα διαδικασία
- ❖ **Πλεονέκτημα:** άμεση επαφή με τον ειδικό και παρατήρηση της συμπεριφοράς του

Είδη Συνεντεύξεων

- ❖ **Μη-δομημένες** συνεντεύξεις.
 - ❑ Γενικές ερωτήσεις που υποβάλλονται με την ελπίδα της καταγραφής όσο περισσότερων πληροφοριών
- ❖ **Ημιδομημένες** συνεντεύξεις.
 - ❑ Σειρά ανοιχτών ερωτήσεων και θεμάτων που πρέπει να καλυφθούν.
- ❖ **Δομημένες** συνεντεύξεις.
 - ❑ Ερωτηματολόγιο με αυστηρά καθορισμένη δομή που περιλαμβάνει συγκεκριμένες ερωτήσεις σχετικές με τα χαρακτηριστικά του προβλήματος



Τεχνικές Συνέντευξης (1/4)

❖ Άμεση Παρατήρηση

- Ο μηχανικός γνώσης πρέπει να παρατηρεί, να ερμηνεύει και να καταγράφει τις διαδικασίες επίλυσης προβλημάτων στο φυσικό τους χώρο
- Η παρατήρηση μπορεί να εκτρέψει την προσοχή των υπολοίπων υπαλλήλων
- Η καταγραφή των γεγονότων απέχει χρονικά από τα ίδια τα γεγονότα, συνεπώς μπορεί να υπάρξουν λάθη
- Είναι απαραίτητο οι επισκέψεις να είναι σύντομες και επαναλαμβανόμενες

❖ Ανάλυση Πρωτοκόλλου

- Ο ειδικός καλείται να λύσει ένα πρόβλημα σκεπτόμενος μεγαλόφωνα
- Αποτελεσματικός τρόπος καταγραφής της διαδικασίας σκέψης
- Βοηθάει τον ειδικό να συνειδητοποιήσει τις διαδικασίες τις οποίες περιγράφει
- Η μέθοδος βοηθάει αργότερα στην αναπαράσταση της γνώσης

Τεχνικές Συνέντευξης (2/4)

❖ Επαναδιδασκαλία (teach-back):

- ❑ Ο μηχανικός γνώσης προσπαθεί να επαναδημιουργήσει και να συνοψίσει ότι έχει ειπωθεί από τον ειδικό και να το διδάξει σε αυτόν.
- ❑ *Μειονέκτημα:* Ο ειδικός μπορεί να κάνει το λάθος να εγκρίνει την επαναδιδασκαλία, χωρίς να εμβαθύνει σε έλεγχο για πληρότητα και η συνέπεια

❖ Διδακτική συνέντευξη (tutorial interview):

- ❑ Ο ειδικός δίνει μια διάλεξη πάνω στην περιοχή του θέματος.
- ❑ *Μειονέκτημα:* Δεν επιτρέπει στο μηχανικό γνώσης να επιβάλλει τη δομή της διαδικασίας

❖ Ταξινόμηση καρτών (card sorting)

- ❑ Ανακάλυψη κατάλληλων ιδιοτήτων των στοιχείων που απαρτίζουν την περιοχή του πεδίου, για την ταξινόμηση των εννοιών.
- ❑ Κάθε στοιχείο της περιοχής γράφεται σε μία κάρτα.
- ❑ Ο μηχανικός γνώσης ζητάει από τον ειδικό να ταξινομήσει τις κάρτες σε σωρούς.
- ❑ Ρωτάει τη βάση στην οποία στηρίχτηκε η ταξινόμηση
- ❑ Ρωτάει παραπλήσιες ερωτήσεις για κάθε στοιχείο μέλος του σωρού.
- ❑ *Ιεραρχική ταξινόμηση:* Μετά την πρώτη ταξινόμηση, κάθε σωρός θεωρείται σαν ένας χώρος (domain) και μπορεί να ταξινομηθεί ξεχωριστά σε υποσωρούς

Τεχνικές Συνέντευξης (3/4)

Βαθμωτά πλέγματα (laddered grids)

- ❖ Οι έννοιες θεωρείται ότι ανήκουν σε έναν άξονα γενικού-ειδικού
- ❖ Γίνονται ερωτήσεις:
 - ❑ Στον άξονα γενικού-ειδικού, ή αντίθετα
 - ❑ Κάθετα στον άξονα
- ❖ Εκκίνηση από έννοια-"σπόρο"
 - ❑ Ανάπτυξη δικτύου περιγραφής των σχέσεων των στοιχείων του θέματος (domain items)
- ❖ Κατευθυντικές ερωτήσεις:
 - ❑ Για τη μετακίνηση από το γενικό προς το ειδικό:
 - "Μπορείς να δώσεις παράδειγμα του ..."
 - "Πώς μπορείς να εξηγήσεις ότι ..."
 - ❑ Για τη μετακίνηση από το ειδικό προς το γενικό:
 - "Τι κοινό υπάρχει ..."
 - "Τι παραδείγματα υπάρχουν από ..."
 - "Τι διαφορές υπάρχουν από ..."
 - ❑ Για την κάθετη μετακίνηση:
 - "Τι εναλλακτικά παραδείγματα του ... υπάρχουν ..."

Τεχνικές Συνέντευξης (4/4)

Πλέγματα Ρεπερτορίων (repertory grid)

- ❖ Ο ειδικός καλείται να κατηγοριοποιήσει το πεδίο του προβλήματος χρησιμοποιώντας το προσωπικό του μοντέλο
 - ❑ Το πλέγμα χρησιμεύει στη σύλληψη και στην αξιολόγηση του μοντέλου του ειδικού
- ❖ Το πλέγμα είναι ένας πίνακας διπλής εισόδου
 - ❑ Τα στοιχεία τοποθετούνται με διαβάθμιση
- ❖ Κάθε **στοιχείο της περιοχής** κατηγοριοποιείται σύμφωνα με ένα σύνολο από **έννοιες ή χαρακτηρισμούς**
 - ❑ Εφαρμόζονται σε όλα τα στοιχεία σε κάποιο βαθμό
- ❖ Κάθε έννοια εκφράζεται σε μια γραμμική, αριθμητική κλίμακα.
 - ❑ Η κλίμακα είναι ίδια κάθε φορά.
 - ❑ Τυπικά οι τιμές κυμαίνονται 1-5 ή 1-10
 - ❑ Υπάρχουν δύο ακραίες τιμές, π.χ. βαρύς/ελαφρύς, φτηνός/ακριβός, κ.α.
 - ❑ Η μέση τιμή (π.χ. 3 στα 5) αντιπροσωπεύει μια ενδιάμεση τιμή της έννοιας.
- ❖ Ζητείται από τον ειδικό να αποδώσει μια τιμή σε κάθε έννοια για όλα τα στοιχεία της περιοχής, στο **πλέγμα** που δημιουργείται.

Παράδειγμα Πλέγματος Ρεπερτορίων

	<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="text-align: center;"> <p>μικροκλοπή</p> <p>διάρρηξη</p> <p>ναρκωτικά</p> <p>δολοφονία</p> <p>ληστεία</p> <p>βιασμός</p> </div> </div>						
οποιονδήποτε	2	1	1	1	1	5	μόνο γυναίκες
μεγάλη καταδίκη	2	1	1	2	3	5	μικρή καταδίκη
ειδική τοποθεσία	2	5	1	1	4	5	οποιαδήποτε τοποθεσία
προσχεδιασμένο	5	3	1	2	5	4	αφθόρμητα
μη-απειλητικός	3	2	2	5	5	5	απειλητικός
απρόσωπο	2	2	1	5	4	5	προσωπικό
ασήμαντο	1	3	1	5	4	5	σημαντικό
μη-βίαιος	1	1	2	5	5	5	βίαιος

μικροκλοπή

διάρρηξη

ναρκωτικά

δολοφονία

ληστεία

βιασμός

	<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="text-align: center;"> <p>μικροκλοπή</p> <p>διάρρηξη</p> <p>ναρκωτικά</p> <p>δολοφονία</p> <p>ληστεία</p> <p>βιασμός</p> </div> </div>						
-	-	-	-	-	-	-	
10	-	-	-	-	-	-	
10	10	-	-	-	-	-	
18	18	16	-	-	-	-	
15	15	21	9	-	-	-	
23	21	29	13	10	-	-	



Επεξεργασία Πλεγμάτων Ρεπερτορίων

- ❖ Εξετάζεται αν κάποιο ζευγάρι είναι παρόμοιο κατά τη σύγκριση των οριζοντίων γραμμών του πλέγματος
 - Παραλείπονται κάποιες παραπλήσιες έννοιες
- ❖ Υπολογίζεται, σε ένα νέο πλέγμα, πόσο όμοια ή ανόμοια είναι τα στοιχεία της περιοχής μεταξύ τους.
- ❖ *Πλεονέκτημα:* Αναγκάζει τον ειδικό να σκεφτεί το πρόβλημα πιο σοβαρά
- ❖ *Μειονέκτημα:* Είναι δύσκολη η διαχείριση των μεγάλων πλεγμάτων
- ❖ Η μέθοδος χρησιμοποιείται συνήθως στα αρχικά στάδια της απόκτησης της γνώσης

Επαλήθευση (verification)

Η ορθή ανάπτυξη του συστήματος (O'Keefe, 1987)

Building the system right.

- ❖ Έλεγχος της συμβατότητας του συστήματος με τις αρχικές προδιαγραφές.
- ❖ Επιβεβαίωση της συνέπειας και πληρότητας της κωδικοποίησης της γνώσης που περιέχεται στο έμπειρο σύστημα.
 - ❑ Έλεγχος λαθών που οφείλονται στους κατασκευαστές του συστήματος.
 - ❑ Δεν ελέγχεται η ίδια η γνώση που εκμαιεύτηκε αλλά ο τρόπος με τον οποίο υλοποιήθηκε.
 - ❑ Ο έλεγχος πραγματοποιείται από το μηχανικό της γνώσης με τη βοήθεια εργαλείων (π.χ. CHECK, TEIRESIAS)
- ❖ Υπάρχουν λάθη
 - ❑ Σε συστήματα Κανόνων
 - ❑ Στις δομημένες αναπαραστάσεις γνώσης.



Λάθη στις Δομημένες Αναπαραστάσεις Γνώσης

- ❖ Οφείλονται συνήθως σε εννοιολογικές παρανοήσεις
- ❖ *Λάθη ταξινόμησης*
 - ❑ Κάποια κλάση τοποθετείται σε άλλο σημείο της ιεραρχίας από εκείνο που θα έπρεπε.
- ❖ *Λάθη ιδιοτήτων*
 - ❑ Κάποιες ιδιότητες (slots) τοποθετούνται σε λάθος σημεία της ιεραρχίας.
 - ❑ Π.χ. αν κάποια ιδιότητα s αποδίδεται και στις δύο υποκλάσεις B και C της κλάσης A , τότε το ορθότερο είναι η ιδιότητα αυτή να ορισθεί στην κλάση A αντί στις υποκλάσεις της.
- ❖ *Λάθη τιμών*
 - ❑ Οι ιδιότητες κάποιων αντικειμένων είτε παίρνουν τιμές αντίθετες με τους περιορισμούς της ιδιότητας, είτε παίρνουν την εξ' ορισμού τιμή, ενώ δεν είναι απαραίτητο.



Λάθη στα Συστήματα Κανόνων

- ❖ Πιθανά συντακτικά και σημασιολογικά λάθη σε συστήματα κανόνων.
- ❖ Επηρεάζουν την ορθότητα του συστήματος
 - ❑ Πλεονάζοντες κανόνες (redundant rules)
 - ❑ Αντικρουόμενοι κανόνες (conflicting rules)
 - ❑ Υπονοούμενοι κανόνες (subsumed rules)
 - ❑ Κυκλικοί κανόνες (circular rules)
 - ❑ Περιττές συνθήκες (unnecessary conditions)
- ❖ Επηρεάζουν την πληρότητα του συστήματος
 - ❑ Αδιέξοδοι κανόνες (dead-end rules)
 - ❑ Απόντες κανόνες (missing rules)
 - ❑ Απρόσιτοι κανόνες (unreachable rules)

Πλεονάζοντες Κανόνες (redundant rules)

- ❖ *Συντακτικός πλεονασμός*: Ίδιες συνθήκες και συμπέρασμα.

```
(defrule rule1
```

```
  (humidity high)
```

```
  (temperature hot)
```

```
=>
```

```
  (assert (weather thunderstorms)))
```

```
(defrule rule2
```

```
  (temperature hot)
```

```
  (humidity high)
```

```
=>
```

```
  (assert (weather thunderstorms)))
```

- ❑ Μπορεί να προκαλέσουν προβλήματα όταν συνοδεύονται από συντελεστές βεβαιότητας, γιατί αυξάνουν τεχνητά τη βεβαιότητα του συμπεράσματος.

- ❖ *Σημασιολογικός πλεονασμός*: Συνθήκες ή/και συμπεράσματα μπορεί να είναι διαφορετικά στη σύνταξη αλλά ίδια στη σημασία.

```
(defrule rule3
```

```
  (humidity high)
```

```
  (temperature hot)
```

```
=>
```

```
  (assert (weather thunderstorms)))
```

```
(defrule rule4
```

```
  (temperature hot)
```

```
  (humidity high)
```

```
=>
```

```
  (assert (weather electrical-storms)))
```

- ❑ Είναι πιο σπάνιο φαινόμενο
- ❑ Αντιμετωπίζεται δυσκολότερα λόγω αδυναμίας αυτόματου ελέγχου από το σύστημα της ομοιότητας των εννοιών
- ❑ Οφείλεται στη μη σωστή δόμηση του συστήματος εννοιών (πλαίσια)

Αντικρουόμενοι Κανόνες (conflicting rules)

❖ Ίδιες συνθήκες, διαφορετικά συμπεράσματα.

```
(defrule rule5
```

```
  (temperature hot)
```

```
  (humidity high)
```

```
=>
```

```
  (assert (weather thunderstorms)))
```

```
(defrule rule6
```

```
  (temperature hot)
```

```
  (humidity high)
```

```
=>
```

```
  (assert (weather sunshine)))
```

Υπονοούμενοι Κανόνες (subsumed rules)

- ❖ Αν ένας κανόνας έχει περισσότερους περιορισμούς στη συνθήκη του από έναν άλλο, ενώ και οι δύο έχουν το ίδιο συμπέρασμα.

```
(defrule rule7
```

```
  (temperature hot)
```

```
  (humidity high)
```

```
  (pressure low)
```

```
=>
```

```
  (assert (weather thunderstorms)))
```

```
(defrule rule8
```

```
  (temperature hot)
```

```
  (humidity high)
```

```
=>
```

```
  (assert (weather thunderstorms)))
```

- ❖ Μόνο ένας κανόνας είναι απαραίτητος.

- Είτε ο **rule7** είναι πολύ εξειδικευμένος ενώ δε χρειάζεται, ή ο **rule8** είναι πολύ γενικός.

- ❖ Αν υπάρχουν συντελεστές βεβαιότητας, τότε μπορεί οι υπονοούμενοι κανόνες να χρησιμεύουν στην αύξηση της βεβαιότητας του συμπεράσματος.

```
(defrule rule7
```

```
  CF=0,5
```

```
  ...)
```

```
(defrule rule8
```

```
  CF=0,7
```

```
  ...)
```

- Αν ισχύει μόνο **temperature hot** και **humidity high** η βεβαιότητα καταιγίδας είναι 0,7 (λόγω **rule8**).

- Αν επιπλέον ισχύει ότι **pressure low**, τότε η βεβαιότητα αυξάνει σε 0,85 (λόγω **rule8** και **rule7**).

Κυκλικοί Κανόνες (circular rules)

- ❖ Δημιουργούν προβλήματα τερματισμού (αέναος βρόχος - infinite loop)
- ❖ Εμφανίζονται με 2 μορφές:
 - *Συντακτικά κυκλικοί κανόνες:* Η συνθήκη κάποιου κανόνα αποτελεί συμπέρασμα κάποιου άλλου και αντίστροφα.

```
(defrule rule9
```

```
  (brothers ?x ?y)
```

```
=>
```

```
  (assert (same-parents ?x ?y)))
```

```
(defrule rule10
```

```
  (same-parents ?x ?y)
```

```
=>
```

```
  (assert (brothers ?x ?y)))
```

- Δημιουργείται πρόβλημα μόνο στα συστήματα παραγωγής που δεν ελέγχουν αν κάποιο συμπέρασμα έχει εισαχθεί ξανά.
- *Κυκλικά δεδομένα:* Η συνθήκη και το συμπέρασμα κάποιου κανόνα αναφέρονται σε δεδομένα που συνδέονται μεταξύ τους κυκλικά.

```
(defrule rule11
```

```
  (important-city ?x)
```

```
  (connected-with-road ?x ?y)
```

```
=>
```

```
  (assert (important-city ?y)))
```

- Δημιουργείται πρόβλημα τερματισμού μόνο αν τα δεδομένα έχουν κυκλική αλληλεξάρτηση.

Περισσότερες Συνθήκες (unnecessary conditions)

- ❖ Δύο κανόνες με ίδια συμπεράσματα αλλά περίπου ίδιες συνθήκες

```
(defrule rule12
```

```
  (patient ?x)
```

```
  (has-pink-spots ?x)
```

```
  (has-fever ?x)
```

```
=>
```

```
  (assert (has-measles ?x)))
```

```
(defrule rule12
```

```
  (patient ?x)
```

```
  (has-pink-spots ?x)
```

```
  (not (has-fever ?x))
```

```
=>
```

```
  (assert (has-measles ?x)))
```

- ❖ Η συνθήκη **has-fever** φαίνεται μη αναγκαία για να έχει κάποιος ιλαρά.

- Οι 2 κανόνες πρέπει να συμπυκνωθούν σε 1:

```
(defrule rule14
```

```
  (patient ?x)
```

```
  (has-pink-spots ?x)
```

```
=>
```

```
  (assert (has-measles ?x)))
```

- ❖ Πολλές φορές τέτοιες καταστάσεις προκαλούνται όχι λόγω άχρηστων συνθηκών αλλά λόγω:

- Ελλιπών συνθηκών
- Λάθος κωδικοποίηση της γνώσης
- Λάθος στην εκμειευμένη γνώση

Αδιέξοδοι Κανόνες (dead-end rules)

❖ Κανόνες με συμπεράσματα τα οποία:

- ❑ Δεν ανήκουν στα τελικά συμπεράσματα του συστήματος.
- ❑ Δεν εμφανίζονται στις συνθήκες άλλων κανόνων.

❖ Παράδειγμα

```
(defrule rule15
```

```
  (gas-gauge empty)
```

```
=>
```

```
  (assert (gas-tank empty)))
```

- ❑ Είναι αδιέξοδος κανόνας, αν το συμπέρασμα **gas-tank empty**:
 - Δεν ανήκει στους τελικούς στόχους του συστήματος.
 - Δεν εμφανίζεται στη συνθήκη κάποιου άλλου κανόνα.

❖ Προβλήματα που προκαλούν οι αδιέξοδοι κανόνες:

- ❑ Άχρηστα συμπεράσματα (πρόβλημα απόδοσης του συστήματος)
- ❑ Ένδειξη για κανόνες που λείπουν (πρόβλημα ορθότητας-πληρότητας του συστήματος)



Απόντες Κανόνες (missing rules)

- ❖ Η απουσία κανόνων μπορεί να γίνει αντιληπτή από:
 - ❑ Παρουσία γεγονότων που δεν εμφανίζονται στη συνθήκη κανενός κανόνα.
 - ❑ Παρουσία τελικών συμπερασμάτων που δεν εμφανίζονται στο συμπέρασμα κανενός κανόνα.
 - ❑ Ύπαρξη αδιέξοδων κανόνων.

Απρόσιτοι Κανόνες (unreachable rules)

- ❖ Κανόνες που δεν ενεργοποιούνται ποτέ γιατί οι συνθήκες στις οποίες στηρίζονται δεν αποτελούν το συμπέρασμα κανενός κανόνα, ούτε ανήκουν στα αρχικά δεδομένα.
- ❖ Είναι ακριβώς το αντίθετο των αδιέξοδων κανόνων.

Έλεγχος Αξιοπιστίας (validation)

Η ανάπτυξη του σωστού συστήματος (O'Keefe, 1987)

Building the right system.

- ❖ Διαπίστωση της ορθότητας του τελικού συστήματος σε σχέση με τις ανάγκες και απαιτήσεις του τελικού χρήστη.
 - ❑ Επιβεβαίωση της ορθότητας των αποτελεσμάτων του συστήματος
 - ❑ Επιβεβαίωση ότι το σύστημα ανταποκρίνεται στις απαιτήσεις των χρηστών
- ❖ Τελικός ποιοτικός έλεγχος στην ανάπτυξη συστημάτων γνώσης.
 - ❑ Ελέγχεται η ορθότητα και γενικότητα της ίδιας της γνώσης που εκμαιεύτηκε και περιέχεται στο έμπειρο σύστημα
 - ❑ Ελέγχεται αν το σύστημα επιλύει τα προβλήματα με ορθό και επακριβή τρόπο

Μεθοδολογίες Ελέγχου Αξιοπιστίας

- ❖ *Άτυπος έλεγχος*: Συναντήσεις μηχανικού γνώσης με ειδικούς κατά τη διάρκεια ανάπτυξης του συστήματος για τον έλεγχο της εγκυρότητας κάποιων αποτελεσμάτων
- ❖ *Δοκιμασία σε υποδειγματικές περιπτώσεις (test cases)*
 - ❑ Οι λύσεις που δίνει το έμπειρο σύστημα συγκρίνονται με λύσεις που δόθηκαν από διάφορους ειδικούς του τομέα στις ίδιες περιπτώσεις.
 - ❑ Οι ειδικοί συμφωνούν ή διαφωνούν (ίσως διαβαθμισμένα) με τις υποδείξεις του έμπειρου συστήματος.
 - ❑ *Δοκιμασία Turing*: Οι λύσεις ειδικών και συστήματος παρουσιάζονται με την ίδια μορφή σε άλλους ειδικούς, οι οποίοι τις αξιολογούν αντικειμενικά.
 - ❑ Οι υποδειγματικές περιπτώσεις πρέπει να είναι διαφορετικές από αυτές που χρησιμοποιήθηκαν στις προηγούμενες φάσεις ανάπτυξης του συστήματος.
 - Εξασφαλίζεται η **ευρωστία (robustness)** σε μη-προσδοκώμενα δεδομένα.
 - ❑ Όταν αυξάνεται η πολυπλοκότητα του συστήματος, ο αριθμός των δοκιμασιών που θα έπρεπε να πραγματοποιηθούν αυξάνεται εκθετικά.
- ❖ *Δοκιμασία σε πραγματικές συνθήκες*
 - ❑ Κίνδυνος απώλειας εμπιστοσύνης από τους τελικούς χρήστες
 - ❑ Πρέπει να γίνεται κοντά στο τελικό στάδιο ανάπτυξης

Μεθοδολογίες Ελέγχου Αξιοπιστίας

- ❖ *Έλεγχος αξιοπιστίας των υποσυστημάτων:* Το σύστημα γνώσης χωρίζεται σε ανεξάρτητα υποσυστήματα τα οποία ελέγχονται ξεχωριστά
 - ❑ Ευκολότερη η επίλυση προβλημάτων σε μικρότερα συστήματα
 - ❑ Η αξιοπιστία του καθενός υποσυστήματος ξεχωριστά δεν εγγυάται πάντα την αξιοπιστία του συνολικού συστήματος
- ❖ *Ανάλυση ευαισθησίας:* Δοκιμασία με σύνολο παραμέτρων που διαφέρουν λίγο σε μία από τις παραμέτρους κάθε φορά
 - ❑ Ιδιαίτερα χρήσιμος έλεγχος σε συστήματα με αβεβαιότητα.

Κριτήρια Αξιοπιστίας

- ❖ Σύγκριση με γνωστά αποτελέσματα
- ❖ Σύγκριση με την απόδοση ειδικών
 - ❑ Μεγαλύτερη ανεκτικότητα σε λάθη, αφού και ο ειδικός μπορεί να κάνει λάθη
- ❖ Σύγκριση με αποτελέσματα που προβλέπονται θεωρητικά
 - ❑ Συνήθως γίνεται όταν το σύστημα γνώσης μοντελοποιεί κάποια φυσική διαδικασία
 - ❑ Όταν δεν υπάρχει ακριβές θεωρητικό μοντέλο, γιατί το φυσικό σύστημα είναι πολύπλοκο, τότε δεν είναι δυνατή αυτή η σύγκριση

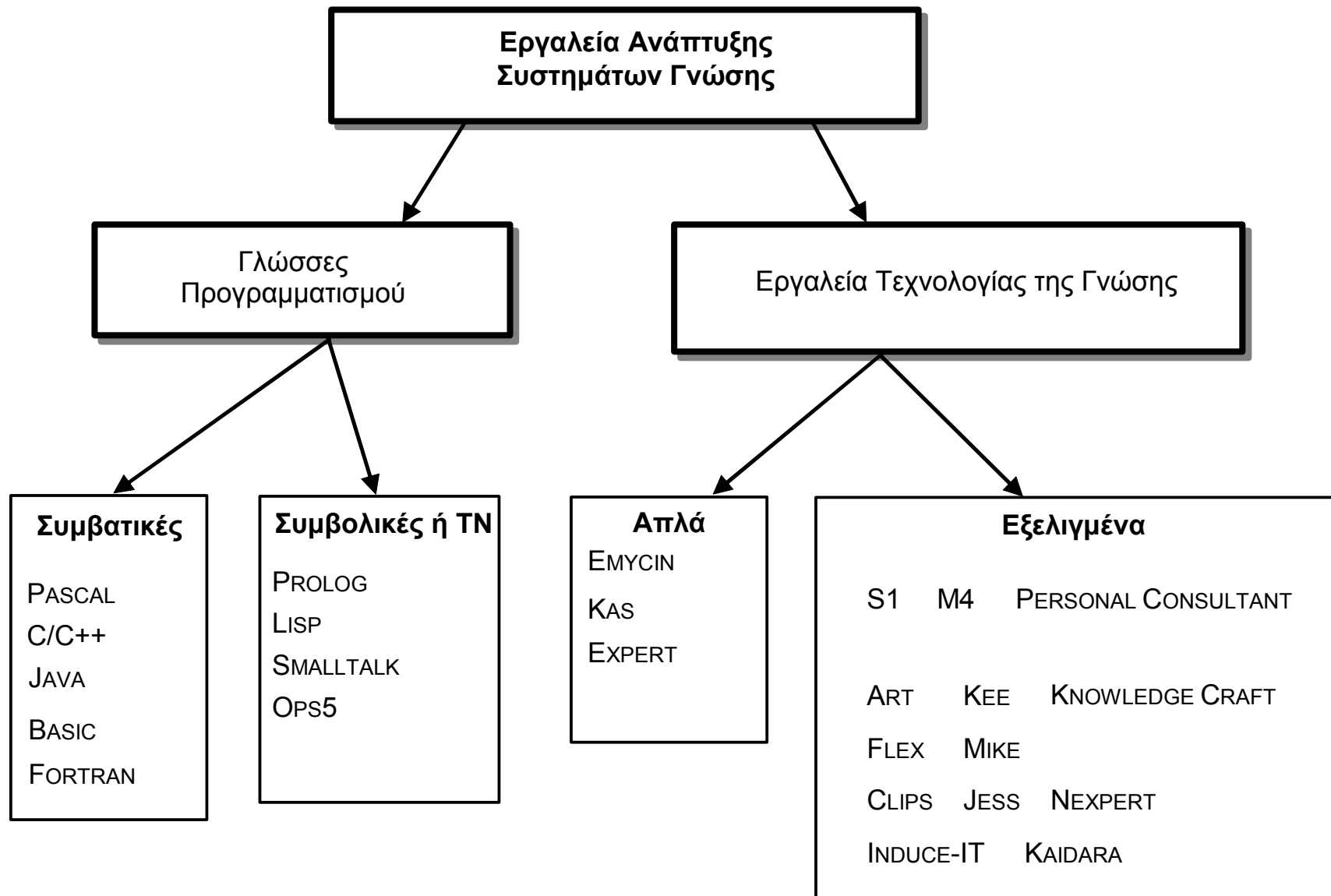
Μέτρηση Αξιοπιστίας

- ❖ **Ακρίβεια (accuracy):** Ποσοστό των αποδεκτών απαντήσεων του συστήματος
 - ❑ Αποδεκτές απαντήσεις είναι αυτές που συμπίπτουν σε αυτές ενός ειδικού
- ❖ **Επάρκεια (adequacy):** Ποσοστό κάλυψης (coverage) του πεδίου γνώσης του προβλήματος
 - ❑ Π.χ. ένα σύστημα κατηγοριοποίησης αναγνωρίζει σωστά το 83% των ειδών
 - ❑ Θα μπορούσε το ποσοστό να περιέχει και βάρη, δίνοντας μεγαλύτερη έμφαση στα σημαντικότερα στοιχεία του πεδίου της γνώσης

Λάθη στην Αναπαράσταση της Γνώσης

- ❖ **Λάθη απόφασης:** Συμβαίνουν όταν το σύστημα καταλήγει σε λάθος αποτέλεσμα
 - ❑ Επηρεάζουν την ακρίβεια του συστήματος
 - ❑ Διαπιστώνονται εύκολα, αλλά εντοπίζονται και διορθώνονται δύσκολα
- ❖ **Λάθη παράλειψης:** Συμβαίνουν όταν το σύστημα δεν μπορεί να καταλήξει σε αποτέλεσμα
 - ❑ Η απαραίτητη γνώση για να λυθεί κάποιο πρόβλημα παραλήφθηκε
 - ❑ Επηρεάζουν την επάρκεια του συστήματος
 - ❑ Διαπιστώνονται δύσκολα γιατί η δοκιμαστική περίπτωση (test case) που θα αποκαλύψει την έλλειψη δεν είναι προφανής στο μηχανικό γνώσης

Εργαλεία Ανάπτυξης Συστημάτων Γνώσης



Γλώσσες Προγραμματισμού TN

- ❖ Αποτελούν εργαλείο για γρήγορη κατασκευή πρωτοτύπου του συστήματος γνώσης.
 - ❑ Ο κώδικας μπορεί να εκτελεστεί και να ελεγχθεί την ώρα που δημιουργείται.
- ❖ Η διασύνδεση αυτών των γλωσσών με το χρήστη δεν είναι αρκετά εξελιγμένη.
- ❖ Οι γλώσσες προγραμματισμού διαθέτουν συνήθως έναν απλό μηχανισμό ελέγχου.
- ❖ Δίνουν στον προγραμματιστή τη δυνατότητα να "δημιουργήσει":
 - ❑ Μηχανισμό ελέγχου για το σύστημα γνώσης (συλλογιστική).
 - ❑ Τρόπο αναπαράστασης της γνώσης, με τις δομές δεδομένων.
- ❖ Κατηγορίες γλωσσών προγραμματισμού TN:
 - ❑ Συναρτησιακός προγραμματισμός (π.χ. LISP)
 - ❑ Λογικός προγραμματισμός (π.χ. PROLOG)
 - ❑ Αντικειμενοστραφής προγραμματισμός (π.χ. SMALLTALK)
 - ❑ Προγραμματισμός με κανόνες παραγωγής (π.χ. OPS5)

Συναρτησιακός Προγραμματισμός (LISP)

- ❖ Είναι προσανατολισμένη στο χειρισμό συμβόλων και λιστών.
- ❖ **Διαδικαστική θεώρηση:** Η γνώση του προβλήματος αναμιγνύεται με τη γνώση του τρόπου επίλυσης.
- ❖ Έχει μόνο 2 τύπους δεδομένων (άτομο-σύμβολο και **λίστα**).
- ❖ Το λεξιλόγιο αποτελείται από 6 θεμελιώδεις συναρτήσεις.
 - ❑ Με βάση αυτές τις στοιχειώδεις συναρτήσεις ο χρήστης ορίζει πιο σύνθετες.
- ❖ **Πλεονεκτήματα:**
 - ❑ Δυναμικότητα, αυτόματη διαχείριση της μνήμης, εύκολη διόρθωση λαθών.
 - ❑ Δυνατότητα για ορισμό αναδρομικών συναρτήσεων.
 - ❑ **Τμηματοποιημένη (modular)** ανάπτυξη προγραμμάτων με τη χρήση πολλών συναρτήσεων.
- ❖ **Μειονέκτημα:**
 - ❑ Δεν αποδίδει κάποια ιδιαίτερη σημασία στα σύμβολα που χειρίζεται.
 - **Μέσο** "κατασκευής" εργαλείων ανάπτυξης συστημάτων γνώσης (π.χ. OPS5).
 - **Όχι** εργαλείο άμεσης κωδικοποίησης και εκτέλεσης της γνώσης.

LISP

Παράδειγμα

- ❖ Η συνάρτηση **consecutive** ελέγχει αν τα ορίσματα **a** και **b** εμφανίζονται συνεχόμενα στη λίστα **list**.

```
(defun consecutive (a b list)
  (cond ((null list) NIL)
        ((and (equal a (car list)) (equal b (car (cdr list)))) T)
        (T (consecutive a b (cdr list)))))
```

- ❖ Κλήση της συνάρτησης για να ελέγξει αν δύο γεγονότα (**temperature_inc**, **pressure_inc**) έχουν συμβεί το ένα αμέσως μετά το άλλο κατά την παρακολούθηση της λειτουργίας ενός κινητήρα:

```
(consecutive
 'temperature_inc 'pressure_inc
 '(open_valve temperature_inc piston_compression pressure_inc))
```

Λογικός Προγραμματισμός (PROLOG) (1/2)

- ❖ Συμβολική γλώσσα που βασίζεται στην κατηγορηματική λογική πρώτης τάξης.
- ❖ Ένα πρόβλημα:
 - ❑ Περιγράφεται με τη μορφή γεγονότων (αξιώματα) και κανόνων (θεωρήματα)
 - ❑ Δεν περιέχει τον ακριβή αλγόριθμο επίλυσης του προβλήματος.
- ❖ Υπάρχει σαφής διαχωρισμός της γνώσης από το μηχανισμό ελέγχου.
 - ❑ Η εξαγωγή συμπερασμάτων γίνεται με τη διαδικασία της **εις άτοπο απαγωγής**.
 - ❑ Οι εναλλακτικές λύσεις ενός προβλήματος ερευνώνται πρώτα σε βάθος (DFS).
- ❖ **Υπόθεση κλειστού κόσμου (closed-world assumption):**
 - ❑ Όσες πληροφορίες δεν αναφέρονται ρητά μέσα στη βάση γνώσης θεωρείται ότι δεν αληθεύουν (ψευδείς).
- ❖ Κατάλληλη για την ανάπτυξη συστημάτων γνώσης:
 - ❑ Οι κανόνες περιγράφουν έναν κανόνα συστήματος γνώσης.
 - ❑ **Μηχανισμός ελέγχου:** Ανάστροφη ακολουθίας εκτέλεσης κανόνων (**backward chaining**).
 - ❑ **Μηχανισμός επίλυσης συγκρούσεων:** Επιλέγεται πάντα ο πρώτος κανόνας ή γεγονός.
 - ❑ Στις μοντέρνες εκδόσεις της PROLOG υπάρχουν αρκετές επεκτάσεις γραφικής διασύνδεσης με το χρήστη.

Λογικός Προγραμματισμός (PROLOG) (2/2)

- ❖ Η PROLOG δίνει αρκετές δυνατότητες μετα-προγραμματισμού
 - ❑ Μπορεί να υλοποιηθούν πλαίσια και *ορθή ακολουθίας εκτέλεσης κανόνων (forward chaining)*.
 - ❑ Οι μηχανισμοί που αναπτύσσονται πάνω από την PROLOG είναι αρκετά αργοί
 - ❑ Οι επεκτάσεις αποτελούν "ξένο" σώμα ως προς τη σύνταξη και ως προς την εκτέλεση.
- ❖ Η PROLOG έχει χρησιμοποιηθεί κυρίως για την ανάπτυξη πρωτοτύπου
- ❖ Σήμερα εξακολουθεί να χρησιμοποιείται για ανάπτυξη συστημάτων γνώσης
 - ❑ Με προσθήκες κάποιων εξειδικευμένων εργαλείων (π.χ. FLEX στην LPA PROLOG).

PROLOG

Παράδειγμα

- ❖ Τα γεγονότα αποτελούν αξιώματα, δηλαδή εκφράζουν γνώση που ισχύει ρητά.
 - ❑ Το παρακάτω γεγονός δηλώνει ότι ο σωλήνας `pipe1` είναι χαλασμένος ή δυσλειτουργεί.
`works_bad(pipe1)`.
- ❖ Οι κανόνες αποτελούν θεωρήματα, δηλαδή εκφράζουν γνώση που ισχύει υπό συνθήκη.
 - ❑ Ο παρακάτω κανόνας δηλώνει πως αν κάποιο εξάρτημα δυσλειτουργεί, τότε και όλα τα εξαρτήματα που το περιέχουν εμφανίζονται ότι δυσλειτουργούν.
`works_bad(X) :- is_part_of(Y,X), works_bad(Y)`.
- ❖ Το κατηγορήμα `works_bad` μπορεί να κληθεί με δύο τρόπους, ανάλογα με αν το όρισμα έχει συγκεκριμένη τιμή, δηλαδή είναι δεσμευμένο με τιμή (*bound*) ή είναι μεταβλητή.
 - ❑ Η παρακάτω κλήση ελέγχει αν η βαλβίδα `valve1` δυσλειτουργεί:
`?- works_bad(valve1)`.
 - ❑ Η παρακάτω κλήση επιστρέφει (μέσω οπισθοδρόμησης) όλα τα εξαρτήματα που μπορεί να αποδειχθεί ότι δυσλειτουργούν:
`?- works_bad(X)`.

Αντικειμενοστραφής Προγραμματισμός

- ❖ Αποτελεί τη σύγχρονη τάση στην κατασκευή λογισμικού
 - ❑ Ορθή δόμηση προβλήματος, ευκολία συντήρησης λογισμικού
- ❖ **Αντικείμενο (object):** Συλλογή από συσχετιζόμενα δεδομένα (χαρακτηριστικά) με συγκεκριμένη δομή
 - ❑ Αντιπροσωπεύει συνήθως μία οντότητα του φυσικού κόσμου
 - ❑ Π.χ., ένα αυτοκίνητο μπορεί να αναπαρασταθεί ως αντικείμενο με χαρακτηριστικά όπως ο κατασκευαστής του, ο αριθμός θυρών και θέσεων, η μέγιστη ταχύτητα, το βάρος του, κλπ.
- ❖ **Πλεονέκτημα:** Προσφέρει εκφραστικές δομές για αναπαράσταση:
 - ❑ Των αντικειμένων του φυσικού κόσμου.
 - ❑ Των συσχετίσεων μεταξύ τους,
 - ❑ Των διεργασιών που λαμβάνουν χώρα.
- ❖ **Μειονέκτημα:** Η επιλογή των αντικειμένων και μηνυμάτων γίνεται τεχνητά-αφύσικα για τα περισσότερα προβλήματα.
 - ❑ Τα αντικείμενα αποτελούν απλώς το μέσο υλοποίησης.
- ❖ Πιο γνωστές γλώσσες:
 - ❑ SMALLTALK (συμβολικός προγραμματισμός), C++/Java (συμβατικός προγραμματισμός), LOOPS, FLAVORS και CLOS (αντικειμενοστραφείς επεκτάσεις της γλώσσας LISP), COOL (συνοδεύει τη γλώσσα παραγωγής CLIPS)



Προγραμματισμός με Κανόνες Παραγωγής (OPS5)

- ❖ Αποτελεί φυσικό τρόπο υλοποίησης της γνώσης.
- ❖ Είναι εύκολος στην εκμάθηση, λόγω απλότητας
- ❖ Η απλότητα είναι αρκετές φορές περιοριστική, αφού δεν επιτρέπει τη δημιουργία σύνθετων αναπαραστάσεων της γνώσης ή απλών διαδικαστικών αλγορίθμων.
- ❖ Χρησιμοποιείται ο **αλγόριθμος Rete**
 - ❑ Επιταχύνει την **ταυτοποίηση (pattern matching)** των κανόνων με τα δεδομένα στη μνήμη εργασίας
- ❖ Μηχανισμός ελέγχου:
 - ❑ Βασίζεται στη συγκέντρωση των ενεργοποιημένων κανόνων στο **σύνολο συγκρούσεων**.
 - ❑ Εφαρμόζονται ευριστικοί αλγόριθμοι **επίλυσης συγκρούσεων** για την επιλογή του κανόνα που θα εκτελεστεί.
- ❖ Λύσεις:
 - ❑ Συνδυασμός με άλλες προγραμματιστικές τεχνικές
 - ❑ Χρήση προηγμένων αρχιτεκτονικών δόμησης συστημάτων TN (π.χ. μαυροπίνακας)



Εργαλεία Τεχνολογίας Γνώσης

- ❖ Εξειδικευμένα εργαλεία για την ανάπτυξη συστημάτων γνώσης.
- ❖ Σημαντικά χαρακτηριστικά:
 - ❑ Μορφές αναπαράστασης γνώσης και συλλογιστικής
 - ❑ Ευκολίες διασύνδεσης με το χρήστη και με το περιβάλλον γενικότερα
- ❖ Τα σύγχρονα εργαλεία είναι εξελιγμένα
 - ❑ Τα απλά εργαλεία παρουσιάστηκαν κυρίως στις δεκαετίες του '70 και του '80

Απλά Εργαλεία

- ❖ Είναι γνωστά ως *κελύφη εμπείρων συστημάτων (expert system shells)*.
- ❖ Προήλθαν από υπάρχοντα έμπειρα συστήματα, με αφαίρεση της βάσης γνώσης τους
 - ❑ Περιβάλλουν ως *κελύφη* μια βάση γνώσης.
- ❖ Ο όρος χρησιμοποιείται καταχρηστικά για να περιγράψει όλα τα εργαλεία τεχνολογίας γνώσης

Το Κέλυφος EMYCIN

- ❖ Προήλθε από το έμπειρο σύστημα MYCIN.
 - ❑ Χρησιμοποιήθηκε για την κατασκευή διαφόρων εμπείρων συστημάτων:
LITHIO (γεωλογία) CLOT (ασθένειες πήξης του αίματος) PUFF (πνευμονικές ασθένειες)
TAX-ADVISOR (νομική) SACON (ανάλυση μηχανολογικών σχεδίων) HEAD MED (ψυχιατρικές διαγνώσεις)
- ❖ Τα γεγονότα παριστάνονται σαν τριάδες: "**έννοια-παράμετρος-τιμή**".
 - ❑ Κάθε τριάδα συνοδεύεται και από ένα **συντελεστή βεβαιότητας**.
- ❖ Οι κανόνες εκτελούνται **ανάστροφα**:
 - ❑ Όταν επαληθεύεται η **συνθήκη** τότε προστίθενται στη μνήμη οι τριάδες της **ενέργειας**.
 - ❑ Η ενέργεια συνοδεύεται με συντελεστές βεβαιότητας.
- ❖ Υποστηρίζονται **μετα-κανόνες**:
 - ❑ Εξετάζουν τις συνθήκες των κανόνων που μπορούν να εκτελεστούν.
 - ❑ Καθορίζουν τη σειρά εκτέλεσης, ή αποτρέπουν την εκτέλεση κάποιων κανόνων.
- ❖ **Μειονέκτημα**: Δεν είναι κατάλληλα για την επίλυση όλων των προβλημάτων, αφού δημιουργήθηκαν από έμπειρο σύστημα που επιλύει συγκεκριμένο πρόβλημα.
 - ❑ Π.χ. το EMYCIN είναι κατάλληλο κυρίως για προβλήματα διάγνωσης (όπως το MYCIN).
 - ❑ Υποστήριξη ενός είδους αναπαράστασης γνώσης, συλλογιστικής και αβεβαιότητας

Εξελιγμένα Εργαλεία (1/2)

- ❖ Εργαλεία της τεχνολογίας της γνώσης τα οποία υποστηρίζουν:
 - ❑ Πολλαπλούς τρόπους αναπαράστασης γνώσης και συλλογιστικής.
 - ❑ Εξελιγμένες δυνατότητες διασύνδεσης.
 - ❑ Μεγαλύτερη γενικότητα εφαρμογών.
- ❖ Κάποια προήλθαν αρχικά από έμπειρα συστήματα
 - ❑ Στη συνέχεια μετατράπηκαν και επεκτάθηκαν ώστε να υποστηρίζουν περισσότερα είδη αναπαράστασης γνώσης και συλλογιστικής.
 - ❑ Π.χ. PERSONAL CONSULTANT, S1 και M4, κ.ά.
- ❖ Κάποια αναπτύχθηκαν από την αρχή σε κάποια γλώσσα προγραμματισμού.
 - ❑ ART, KEE, και KNOWLEDGE CRAFT (αναπτύχθηκαν σε LISP)
 - ❑ FLEX και MIKE (αναπτύχθηκαν σε PROLOG)
 - ❑ NEXPERT και CLIPS (αναπτύχθηκαν σε C/C++)
 - ❑ JESS (αναπτύχθηκε σε JAVA)
- ❖ Υπάρχουν εργαλεία που υποστηρίζουν την ανάπτυξη συστημάτων γνώσης με τη χρήση της συλλογιστικής των περιπτώσεων
 - ❑ Π.χ. INDUCE-IT, KAIDARA, κτλ.

Εξελιγμένα Εργαλεία (2/2)

- ❖ Διαφορές με τα απλά εργαλεία:
 - ❑ Δε δεσμεύονται από τη δόμηση και τους περιορισμούς κάποιου συστήματος γνώσης.
 - ❑ Δίνουν πολλές δυνατότητες στην κατασκευή και τη συντήρηση των συστημάτων γνώσης.
 - ❑ Δυσκολότερα στην εκμάθηση
 - Περιέχουν πολλές και ετερογενείς προγραμματιστικές έννοιες
- ❖ Οι κυριότεροι μέθοδοι αναπαράστασης γνώσης:
 - ❑ **Πλαίσια:** Πλεονεκτούν στη δομημένη αναπαράσταση σύνθετων φυσικών αντικειμένων και στο συμπαγή τρόπο χειρισμού τους.
 - ❑ **Κανόνες:** Πλεονεκτούν στο δηλωτικό τρόπο αναπαράστασης εμπειρικών συσχετίσεων μεταξύ παρατηρηθέντων δεδομένων και επαρκούντων δράσεων για την αντιμετώπιση των περιπτώσεων.
- ❖ Ο συνδυασμός των δύο μεθόδων αναπαράστασης γνώσης γίνεται ως εξής:
 - ❑ Οι συνθήκες των κανόνων μπορούν να αναφέρονται σε τιμές των ιδιοτήτων των πλαισίων.
 - ❑ Οι ενέργειες μπορούν να αλλάζουν τις τιμές των ιδιοτήτων ή να δημιουργούν και να διαγράφουν πλαίσια.



PERSONAL CONSULTANT

- ❖ Είναι εξέλιξη του EMYCIN.
 - ❑ Γεγονότα και Κανόνες.
 - ❑ Τα γεγονότα παριστάνονται με τριάδες όπως και στο EMYCIN.
 - ❑ Υποστηρίζει αβεβαιότητα
- ❖ Υποστηρίζει ανάστροφη και ορθή συλλογιστική
- ❖ Εξελιγμένες δυνατότητες:
 - ❑ Επεξεργαστής κειμένου για τη βάση γνώσης
 - ❑ Επεξήγηση πορείας συλλογισμού
 - ❑ Πρόγραμμα επεξεργασίας φυσικής γλώσσας
 - ❑ Έλεγχος της συνέπειας της βάσης γνώσης



ART

Automated Reasoning Tool

- ❖ Προσφέρει διάφορους τρόπους αναπαράστασης της γνώσης.
 - Κανόνες (ορθή - ανάστροφη συλλογιστική)
 - Σενάρια.
 - κλπ.
- ❖ Εξελιγμένοι μηχανισμοί ελέγχου
 - Αρχιτεκτονική **μαυροπίνακα**
 - Μηχανισμός συντήρησης της αλήθειας (truth maintenance)**
- ❖ Εξελιγμένες δυνατότητες:
 - ART Studio: Γραφικό εργαλείο για τη σταδιακή ανάπτυξη της βάσης γνώσης μέσω παραθύρων, μενού, κλπ.
 - Δυνατότητα κλήσης προγραμμάτων σε άλλες γλώσσες προγραμματισμού



ΚΕΕ

Knowledge Engineering Environment

- ❖ Η αναπαράσταση γνώσης γίνεται με **μονάδες (units)** ή **πλαίσια**.
- ❖ Ορθή ή ανάστροφη εκτέλεση κανόνων, με μηχανισμό οπισθοδρόμησης
- ❖ Εξελιγμένες δυνατότητες:
 - ❑ Εργαλείο σύνταξης της βάσης γνώσης.
 - ❑ ΚΕΕ Worlds: Αναπαράσταση και σύγκριση εναλλακτικών σεναρίων
 - ❑ TMS: μηχανισμός **συντήρησης** της αλήθειας
 - ❑ ΚΕΕ Connection: διασύνδεση με σχεσιακές βάσεις δεδομένων