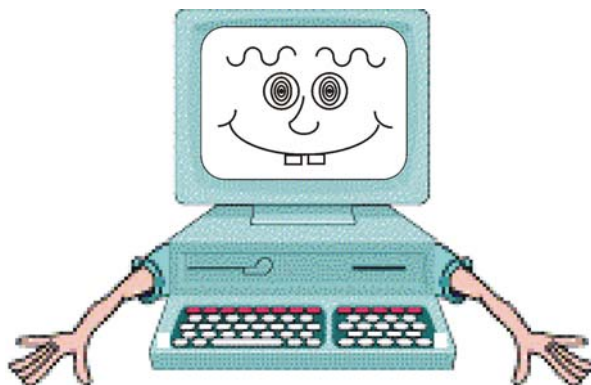




ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ κ' ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

# ΣΗΜΕΙΩΣΕΙΣ ΕΠΙΚΟΙΝΩΝΙΑΣ ΑΝΘΡΩΠΟΥ-ΜΗΧΑΝΗΣ

*Καθηγητής Γιάννης Ιωαννίδης*  
*Δρ. Γιώργος Λέπουρας*



2004-2005

**ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ**

<b>ΜΕΡΟΣ Ι</b>	<b>ΣΥΝΤΕΛΕΣΤΕΣ</b>	<b>3</b>
Κεφάλαιο 1	Εισαγωγή	4
Κεφάλαιο 2	Ο Άνθρωπος	13
Κεφάλαιο 3	Ο Υπολογιστής (Η Μηχανή)	25
Κεφάλαιο 4	Η επικοινωνία	29
Κεφάλαιο 5	Χρηστικότητα	35
<b>ΜΕΡΟΣ ΙΙ</b>	<b>ΑΝΑΠΤΥΞΗ ΣΥΣΤΗΜΑΤΩΝ ΕΑΜ</b>	<b>39</b>
Κεφάλαιο 6	Κύκλος Ζωής Λογισμικού ΕΑΜ	40
Κεφάλαιο 7	Καταγραφή Προδιαγραφών Συστημάτων ΕΑΜ	47
Κεφάλαιο 8	Σχεδίαση Συστημάτων ΕΑΜ	55
Κεφάλαιο 9	Αξιολόγηση Συστημάτων ΕΑΜ	71
<b>ΜΕΡΟΣ ΙΙΙ</b>	<b>ΣΧΕΔΙΑΣΗ ΔΙΕΠΑΦΩΝ</b>	<b>80</b>
Κεφάλαιο 10	Γενικές Αρχές	81
Κεφάλαιο 11	Μορφές Διαλόγου	82
Κεφάλαιο 12	Σχεδιασμός σε περιβάλλον ΠΕΠΕ	87
Κεφάλαιο 13	Σχεδιασμός σε περιβάλλον ΠΗΠ	97
Κεφάλαιο 14	Οπτικοποίηση Πληροφοριών	101
<b>ΜΕΡΟΣ ΙV</b>	<b>ΑΛΛΑ ΘΕΜΑΤΑ</b>	<b>110</b>
Κεφάλαιο 15	Τεκμηρίωση - Εγχειρίδια – Βοήθεια	111
Κεφάλαιο 16	Εργαλεία Ανάπτυξης Διεπαφών	117
<b>ΜΕΡΟΣ V</b>	<b>ΠΑΡΑΡΤΗΜΑΤΑ</b>	<b>121</b>
Κεφάλαιο 17	Παράρτημα Α: Στοιχεία HTML και Javascript	123
Κεφάλαιο 18	Παράρτημα Β: Ορολογία	141
Κεφάλαιο 19	Παράρτημα Γ: Βιβλιογραφία	142

# ΜΕΡΟΣ Ι ΣΥΝΤΕΛΕΣΤΕΣ

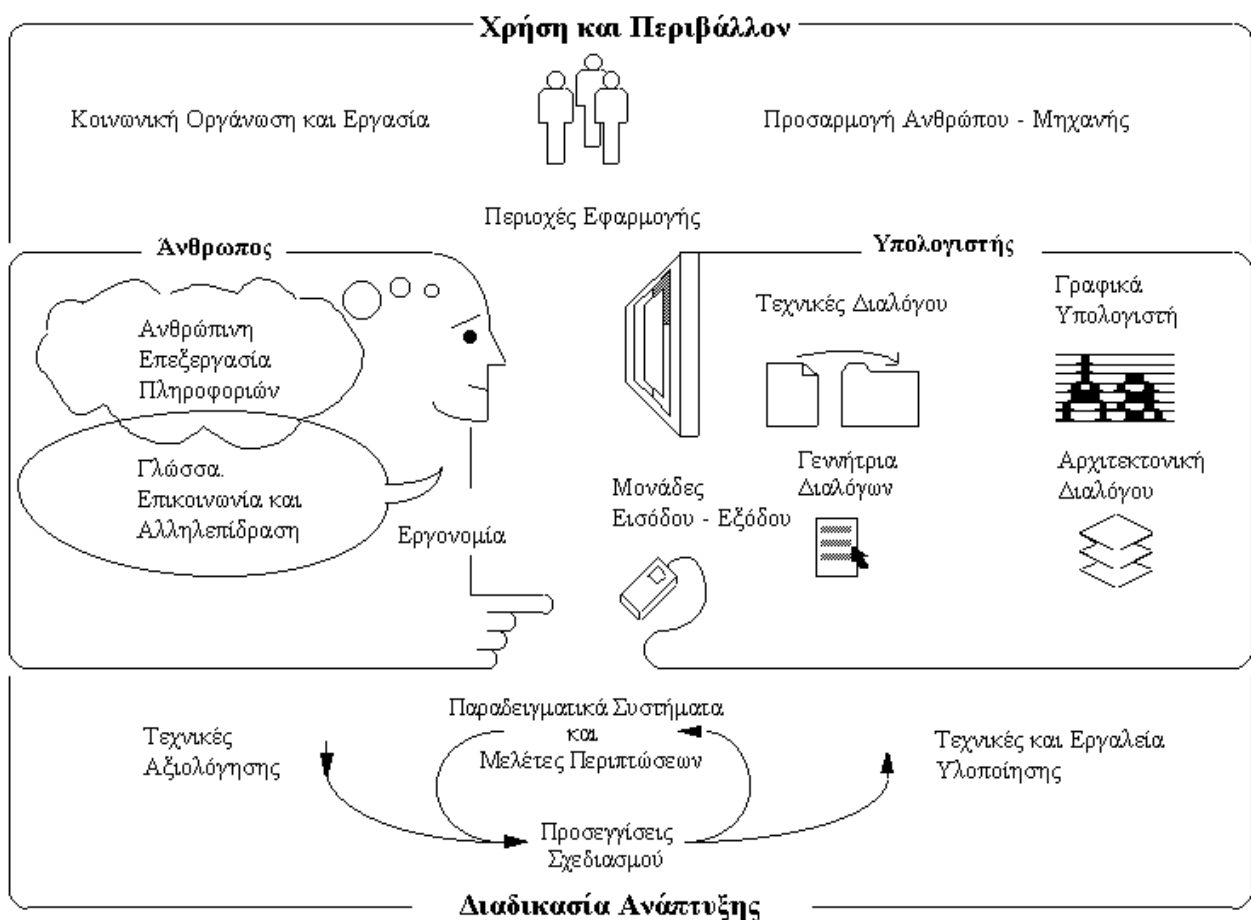
## Κεφάλαιο 1 Εισαγωγή

### 1.1 Γενικοί Ορισμοί της Περιοχής

Ορισμός [σύμφωνα με την ACM]: Ο τομέας της "Επικοινωνίας Ανθρώπου-Μηχανής (EAM)" (*Human-Computer Interaction*) ασχολείται με τον σχεδιασμό, την υλοποίηση, και την αξιολόγηση διαδραστικών υπολογιστικών συστημάτων προορισμένων για ανθρώπινη χρήση και την μελέτη σημαντικών φαινομένων γύρω από αυτά.

Ορισμός [σύμφωνα με τον Moran]: Ο Τομέας της "Διεπαφής του Χρήστη" (*User Interface*) ασχολείται με τις γλώσσες και όργανα εισόδου των χρηστών, τις γλώσσες και όργανα εξόδου των μηχανών, και τα πρωτόκολλα της διάδρασης μεταξύ τους.

Η EAM είναι πολύ πλατύτερος τομέας από την Διεπαφή Χρήστη. Η Διεπαφή Χρήστη καλύπτει μόνο τα χειροπιαστά θέματα ενός υπολογιστή στα οποία ο χρήστης έχει άμεση εμπειρία, όπως δείχνει και η παρακάτω εικόνα [ACM]:



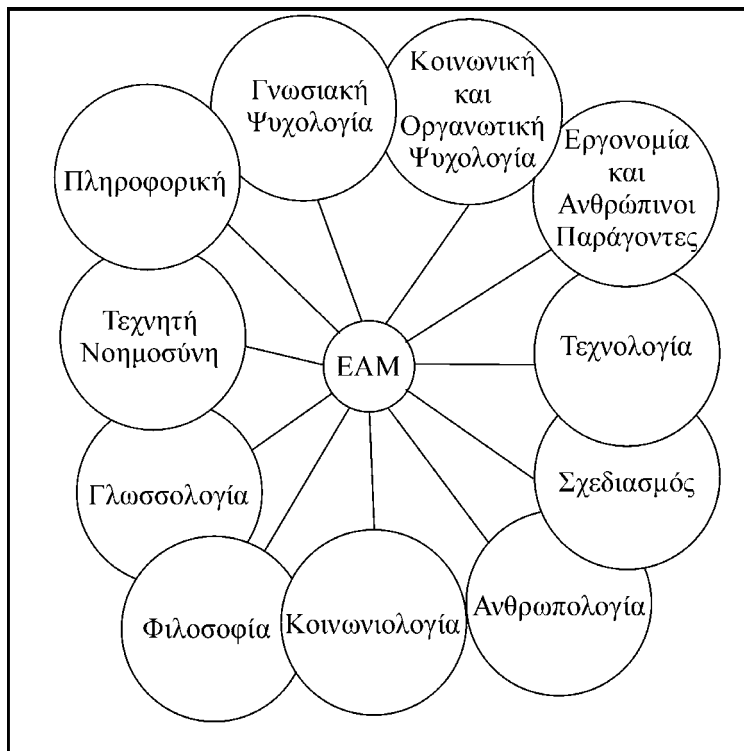
Ο βασικός στόχος της EAM είναι η αυξημένη *Χρηστικότητα* (*Usability*), η οποία επιτυγχάνεται όταν η επικοινωνία ενός χρήστη με ένα σύστημα χαρακτηρίζεται από:

- Μικρό χρόνο εκμάθησης του συστήματος

- Μικρό απαιτούμενο χρόνο για να κάνει ο χρήστης τη δουλειά του
- Μικρό και ελαφριάς μορφής αριθμό λαθών από πλευράς χρήστη
- Μακρά συγκράτηση των χαρακτηριστικών της διεπαφής
- Υποκειμενική ικανοποίηση του χρήστη

Πολλά από τα παραπάνω συγκρούονται μεταξύ τους και δεν είναι δυνατόν να επιτευχθούν ταυτόχρονα.

Υπάρχουν πάρα πολλοί επιστημονικοί τομείς που συνεισφέρουν στην EAM. Σχηματικά αναφέρονται στην παρακάτω εικόνα [Preece, σελ. 38] και αναλύονται ακολούθως.



- Πληροφορική: Σχετικά θέματα είναι η τεχνολογία λογισμικού για την ανάπτυξη εργαλείων, αρχιτεκτονικές και κατάλληλες νοητικές αφαιρέσεις, γραφικά, οπτική αναπαράσταση και εικονική πραγματικότητα, ανάλυση συστημάτων, γλώσσες προγραμματισμού υψηλού επιπέδου.
- Γνωσιακή ψυχολογία: Προσπαθεί να κατανοήσει την ανθρώπινη συμπεριφορά και της νοητικές διεργασίες που την διέπουν. Σχετικά θέματα προκύπτουν από το ότι όλες οι αισθήσεις χρησιμεύουν για επεξεργασία πληροφοριών, και ενδιαφέρει η επίδραση σ' αυτήν την επεξεργασία από την αντίληψη, διάρκεια προσοχής, μνήμη, δυνατότητα εκμάθησης σκέψη, και ικανότητα επίλυσης προβλημάτων χρήστη.
- Κοινωνική και οργανωσιακή ψυχολογία: Προσπαθεί να κατανοήσει την ανθρώπινη συμπεριφορά σε κοινωνικό περιβάλλον. Σχετικά θέματα είναι η εισαγωγή υπολογιστών σε εργασιακό περιβάλλον, η επίδραση σε ανθρώπινες δομές ιεραρχίας και ισχύος, αναγνώριση ρόλων πληροφοριών, κτλ. καθώς και συνεργασίας προς έναν κοινό σκοπό μέσω υπολογιστή.

- **Εργονομία:** Ορίζει και σχεδιάζει εργαλεία και όργανα που ταιριάζουν σε διαφορετικές δυνατότητες χρηστών σε διάφορα περιβάλλοντα εργασίας, διασκέδασης, και σπιτιού. Ενδιαφέρεται για την ασφάλεια, άνεση, παραγωγικότητα, και αξιοπιστία του χρήστη (από χρήστη υπολογιστή μέχρι χρήστη ... καρέκλας αυτοκινήτου), με σχετικά θέματα την αναγνωσιμότητα πληροφορίας, μείωση της ακτινοβολίας οθόνης, μείωση του τεντώματος διαφόρων μυών, σωστή στάση, κτλ.
- **Γλωσσολογία:** Σχετικά θέματα είναι η κατανόηση φυσικής γλώσσας, κατανόηση συνομιλιών (σε περιβάλλοντα συνεργασίας), διεθνοποίηση λογισμικού (ημερομηνίες και απλό κείμενο), αλλά και αντίστροφα τοπικιστική προσαρμογή λογισμικού (π.χ., αφαίρεση μούντζας σαν ένδειξης κέρσορα).
- **Τεχνητή νοημοσύνη:** Ασχολείται με την προσομοίωση έξυπνης ανθρώπινης συμπεριφοράς. Σχετικά θέματα είναι κυρίως η αναγνώριση φυσικής γλώσσας και αναγνώριση φωνής, και πίο πρόσφατα η τεχνολογία πρακτόρων.
- **Φιλοσοφία, Κοινωνιολογία, Ανθρωπολογία:** Σχετικά θέματα είναι η χρήση μεθόδων για τον σχεδιασμό και την αξιολόγηση συστημάτων, π.χ., η χρήση εθνομεθοδολογίας όπου δεν προϋπάρχουν μοντέλα για να επαληθευθούν, οπότε απαιτείται η κατ' ευθείαν μελέτη φαινομένων στο φυσικό τους περιβάλλον.
- **Τεχνολογία (γενικά):** Σχετικά θέματα είναι το χτίσιμο μοντέλων και ο εμπειρικός έλεγχος.
- **Τέχνες (γραφικές και μη):** Σχετικά θέματα είναι η σωστή τοποθέτηση αντικειμένων στην οθόνη, σωστή τοποθέτηση κειμένων, χρωματική αρμονία, ευχάριστη παρουσίαση.

Η έμφαση μας θα είναι κυρίως σε θέματα που αφορούν την πληροφορική, την γνωσιακή ψυχολογία, και τις τέχνες.

## 1.2 Συνοπτική Ιστορία της EAM

Η EAM εμφανίστηκε ως κλάδος με ρίζες στα γραφικά υπολογιστών, τα λειτουργικά συστήματα, την εργονομία, το βιομηχανικό σχεδιασμό/ τεχνολογία (industrial engineering) και τη γνωστική ψυχολογία. Τα γραφικά υπολογιστών γεννήθηκαν από τη χρήση των οθονών καθοδικής λυχνίας (CRT) και των συσκευών κατάδειξης (pen devices) πολύ νωρίς στην ιστορία των υπολογιστών. Αυτό οδήγησε στην ανάπτυξη διαφόρων τεχνικών διάδρασης ανθρώπου-υπολογιστή. Πολλές τεχνικές χρονολογούνται από την εποχή της διδακτορικής διατριβής του Sutherland που αναφερόταν στο Sketchpad [1] η οποία και έθεσε τις βάσεις για την ανάπτυξη του τομέα των γραφικών υπολογιστών. Η συνεχιζόμενη έρευνα στο τομέα των γραφικών επέτρεψε την ανάπτυξη αλγορίθμων και γραφικών που επέτρεπαν την εμφάνιση και διαχείριση ακόμα πιο ρεαλιστικών αντικειμένων (για παράδειγμα τα μέρη μιας μηχανής ή οι ιατρικές εικόνες τμημάτων του σώματος). Τα γραφικά υπολογιστών σχετίζονται άμεσα με την Αλληλεπίδραση Ανθρώπου Υπολογιστή (A.A.Y) ως "αλληλεπιδραστικά γραφικά" (π.χ. διαχείριση μοντέλων σε σχεδιαστικές εφαρμογές).

Άμεσα σχετιζόμενες ήταν και ένα σύνολο προσπαθειών για την επίτευξη της "συμβίωσης ανθρώπου-μηχανής" (man-machine symbiosis, Licklider, 1960) [2], "η επαύξηση της ανθρώπινης διάνοιας" (Engelbart, 1963) και το "Dynabook" (Kay και Goldberg, 1977) [3]. Μέσα από αυτή τη

σειρά των εξελίξεων προήλθε ένα σημαντικό τμήμα της ΑΑΥ. Μερικές από αυτές τις εξελίξεις συμπεριλαμβάνουν το ποντίκι, τις ψηφιογραφικές οθόνες (bitmapped displays), τους προσωπικούς υπολογιστές, τα παράθυρα, τη μεταφορά του γραφείου/ χώρου εργασίας (desktop metaphor) και τους κειμενογράφους με δυνατότητα χρήσης συσκευής κατάδειξης (Baecker & Buxton, 1987, Κεφάλαιο 1) [4].

Το ίδιο χρονικό διάστημα η έρευνα στο χώρο των λειτουργικών συστημάτων, είχε ως αποτέλεσμα την ανάπτυξη τεχνικών για τη διασύνδεση μονάδων εισόδου/ εξόδου, για την ρύθμιση του χρόνου απόκρισης των συστημάτων σε αντιστοιχία με τον ανθρώπινο χρόνο, για πολυεπεξεργασία και για την υποστήριξη παραθυρικών περιβαλλόντων και κινούμενων εικόνων. Αυτή η κίνηση έδωσε ώθηση σε συστήματα διαχείρισης διεπαφών (user interface management systems) και σε εργαλειοθήκες διεπαφών (user interface toolkits).

Ο τομέας των Ανθρώπινων Παραγόντων (Human Factors), εμφανίζεται με την έρευνα του σχεδιασμού εξοπλισμού και της χρήσης του από ανθρώπους κατά τη διάρκεια του δευτέρου Παγκοσμίου πολέμου (Sanders & McCormick, 1987). Πολλά από τα προβλήματα που αντιμετωπίστηκαν τότε είχαν να κάνουν με κινητικά – μηχανικά χαρακτηριστικά (όπως το σχεδιασμό των χειριστηρίων και των οργάνων των αεροπλάνων). Το πρόβλημα της λειτουργικότητας των υπολογιστών ήταν μια φυσική επέκταση του τομέα, εκτός από το γεγονός ότι τα νέα προβλήματα που έπρεπε να αντιμετωπισθούν είχαν σοβαρές γνωστικές, επικοινωνιακές και αλληλεπιδραστικές πτυχές που δεν είχαν ανακύψει παλαιότερα, ωθώντας μια επέκταση προς αυτές τις κατευθύνσεις. Η εργονομία είναι συναφής με τον τομέα των Ανθρώπινων Παραγόντων, προήλθε όμως από μελέτες στο χώρο της εργασίας. Και η εργονομία ενδιαφέρεται για κινητικά-μηχανικά χαρακτηριστικά, με περισσότερο όμως βάρος στην ανατομική πλευρά και στο άγχος/ κούραση. Η αλληλεπίδραση ανθρώπου-υπολογιστή ήταν επίσης άμεσου ενδιαφέροντος για την εργονομία, όμως η επέκταση προς γνωστικά θέματα ήταν αναγκαία δημιουργώντας τον τομέα της γνωστικής εργονομίας (cognitive ergonomics) και της γνωστικής τεχνολογίας (cognitive engineering). Λόγω της ιστορίας τους η εργονομική εξέταση των υπολογιστών δίνει έμφαση στη σχέση με το περιβάλλον εργασίας και τα αποτελέσματα των παραγόντων πίεσης/ άγχους, όπως η επαναληπτική εργασία, η θέση εργασίας ή ο σχεδιασμός των οθονών.

Η βιομηχανική τεχνολογία (industrial engineering) ξεπήδησε από τις προσπάθειες αύξησης της βιομηχανικής παραγωγής, ξεκινώντας τα πρώτα χρόνια του τρέχοντος αιώνα. Η αρχική έμφαση ήταν στο σχεδιασμό αποτελεσματικών μεθόδων εργασίας με τα χέρια (π.χ. η μέθοδος τοποθέτησης τούβλων με τα δύο χέρια), ο σχεδιασμός εξειδικευμένων εργαλείων για την αύξηση της παραγωγικότητας και τη μείωση της κόπωσης και σε μικρότερο βαθμό στο σχεδιασμό του κοινωνικού περιβάλλοντος (π.χ. η εφεύρεση του κουτιού παραπόνων). Η διάδραση με τους υπολογιστές έχει άμεση σχέση με το στόχο της βιομηχανικής μηχανολογίας σε σχέση με το πως χρησιμοποιούνται οι υπολογιστές στο ευρύτερο πλαίσιο των μεθόδων εργασίας.

Η γνωσιακή ψυχολογία εμφανίζεται ως τομέας στο τέλος του 18ου αιώνα, όταν γίνονται οι πρώτες πειραματικές προσπάθειες μελέτης των αισθήσεων. Στα 1950, μια εισροή από ιδέες από τον τομέα της μηχανολογίας επικοινωνίας, της γλωσσολογίας και της μηχανολογίας υπολογιστών οδήγησαν σε μια επιστήμη που βασιζόταν σε πειραματικές αποδείξεις και με κύριο πεδίο έρευνας την επεξεργασία των πληροφοριών από τον άνθρωπο και τις επιδόσεις. Οι ερευνητές της γνωσιακής ψυχολογίας έχουν επικεντρωθεί στην εκμάθηση συστημάτων, στη μεταφορά της γνώσης, τη νοητική αναπαράσταση των συστημάτων από τον άνθρωπο και τις επιδόσεις του ανθρώπου με τέτοια συστήματα.

Τέλος η ανάπτυξη της αγοράς των προσωπικών υπολογιστών συνέδεσε σε μεγάλο βαθμό τις πωλήσεις των υπολογιστών με την ποιότητα των διεπαφών τους. Το αποτέλεσμα ήταν η σταδιακή εξέλιξη μιας τυποποιημένης αρχιτεκτονικής διεπαφών από το επίπεδο της υποστήριξης ποντικών, από το υλικό μέχρι τα παραθυρικά περιβάλλοντα και επίπεδα διαχείρισης εφαρμογών (application management layers). Μαζί με αυτές τις αλλαγές οι ερευνητές και οι σχεδιαστές έχουν ξεκινήσει την ανάπτυξη τεχνικών παραγωγής προδιαγραφών για διεπαφές και τεχνικών αξιολόγησης για διεπαφές.

Η ανάπτυξη του τομέα της Αλληλεπίδρασης Ανθρώπου-Υπολογιστή στα τελευταία 20 χρόνια είναι θεαματική. Αποτελεί ένας από τους πιο γρήγορα αναπτυσσόμενους τομείς στην Πληροφορική σε παγκόσμιο επίπεδο. Ο βασικός λόγος για την ταχεία αυτή ανάπτυξη είναι αύξηση της υπολογιστικής δύναμης σε συνάρτηση με την παράλληλη μείωση του αντίστοιχου κόστους. Οι υπολογιστές ήταν το εργαλείο για τους λίγους, ενώ τώρα έχουν γίνει κτήμα πολλών. Η πλειονότητα του πληθυσμού στον ανεπτυγμένο κόσμο τώρα θεωρεί τις υπολογιστικές εφαρμογές ως μέρος της καθημερινής ρουτίνας. Ως αποτέλεσμα, το πορτρέτο ενός συνηθισμένου χρήστη υπολογιστών έχει αλλάξει. Ενώ παλαιότερα ο μέσος χρήστης υπολογιστών ήταν ταυτόχρονα γνώστης της τεχνολογίας και πιθανότατα ερευνητής του συγκεκριμένου χώρου, σήμερα ο μέσος χρήστης είναι λιγότερο πιθανό να κατανοήσει την τεχνολογία των υπολογιστικών συστημάτων και ακόμη λιγότερο πιθανό να ξοδέψει πολύτιμο χρόνο για την εκμάθηση ενός εξειδικευμένου συστήματος. Για αυτό το λόγο οι υπεύθυνοι ανάπτυξης προγραμμάτων θεωρούν αναγκαίο χαρακτηριστικό των προγραμμάτων τους τη φιλικότητα προς το χρήστη. Αρκετοί από τους χρήστες δεν έχουν την πολυτέλεια της επιπλέον εκπαίδευσης και ζητούν συστήματα που θα μπορούν να χρησιμοποιήσουν από την πρώτη μέρα.

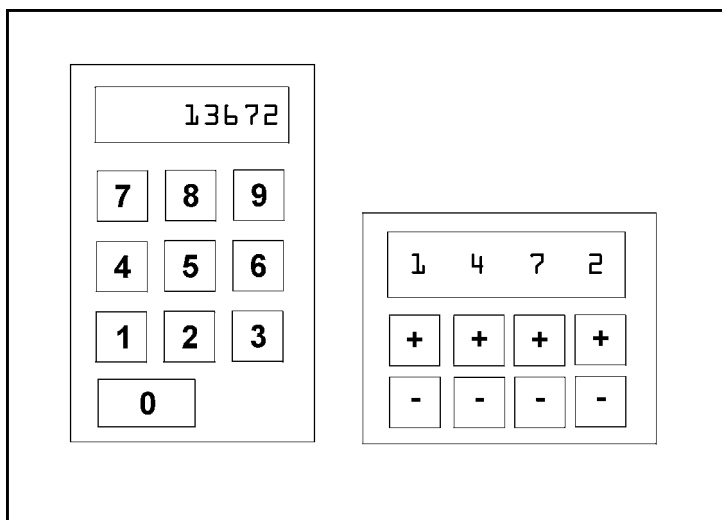
### 1.3 Σημασία της EAM

Τα ακόλουθα παραδείγματα δείχνουν τη σημασία που έχει αποκτήσει ο τομέας της επικοινωνίας ανθρώπου – μηχανής στην καθημερινή μας ζωή. Ακόμη και μικρές, φαινομενικά ασήμαντες λεπτομέρειες στο σχεδιασμό αλληλεπιδραστικών συστημάτων μπορούν να προκαλέσουν μεγάλες καταστροφές.

- Όταν τα (εικονικά) πλήκτρα ΑΠΟΘΗΚΕΥΣΗ (SAVE) και ΔΙΑΓΡΑΦΗ (DELETE) είναι πολύ κοντά, τότε πολύ εύκολα μπορεί να πάει κανείς με το ποντίκι στην λάθος επιλογή. Ακόμη και όταν υπάρχει κουτί διαλόγου που ζητάει επιβεβαίωση της πράξης υπάρχει πρόβλημα, γιατί αυτό το ίδιο κουτί εμφανίζεται μετά και από τις δύο.
- Όλοι ξέρουν πώς να παίζουν βιντεοκασέτες στο βίντεο αλλά ελάχιστοι ξέρουν να το προγραμματίζουν για μαγνητοσκόπηση μια συγκεκριμένη ώρα. Παρόμοια οι περισσότεροι δεν ξέρουν πώς να διορθώνουν την ώρα στο βίντεο, η οποία συχνά αναβοσβήνει. Το 90% των δυνατοτήτων του βίντεο δεν χρησιμοποιείται σχεδόν ποτέ. Αντίθετα, αν και το αυτοκίνητο είναι πιο πολύπλοκο είναι πολύ πιο εύχρηστο. Γιατί; Ένας βασικός λόγος είναι ότι δίνει άμεση ανάδραση στις πράξεις του χρήστη/ οδηγού ενώ στο βίντεο καταλαβαίνεις το λάθος όταν είναι πολύ αργά.
- Χρειάζεται ιδιαίτερη προσοχή στις διεπαφές των οργάνων υγείας, διότι το παραμικρό λάθος μπορεί να αποβεί μοιραίο. Για παράδειγμα, για τον καθορισμό δοσολογίας ακτινοβολίας ή φαρμάκου, είναι πολύ προτιμότερη η χρήση δύο πλήκτρων +/- για κάθε ψηφίο παρά ένα κλασσικό πληκτρολόγιο αριθμομηχανής (κομπιουτεράκι), διότι η διαφορά της επιθυμητής



δόσης από την πληκτρολογούμενη όταν γίνει ένα λάθος μπορεί να είναι τεράστια με το δεύτερο. Σχηματικά αυτό φαίνεται στην παρακάτω εικόνα [Dix, σελ. 7].



- Η αποστολή ενός από τους δύο δορυφόρους (VENUS I ή II) που στάλθηκαν στην Αφροδίτη απέτυχε διότι ο δορυφόρος χτύπησε στο έδαφος της Αφροδίτης χωρίς να δώσει αποτελέσματα. Αυτό οφείλεται σε κάποιο λάθος που είναι ουσιαστικά θέμα EAM. Το πρόγραμμα πλοήγησης του δορυφόρου ήταν γραμμένο σε FORTRAN όπου η στήλη 6 καθορίζει εάν η γραμμή είναι συνέχεια της προηγούμενης. Κάποια εντολή ξεκίνησε από την 6 αντί την 7 και για κακή τύχη το πρόγραμμα συνέχιζε να είναι συντακτικά σωστό (αλλά βέβαια σημασιολογικά λάθος). Αν το πρόγραμμα ήταν γραμμένο μέσω μιας διεπαφής που η πληκτρολόγηση στην στήλη 6 προειδοποιούσε τον χρήστη, τότε μάλλον η αποστολή θα είχε στεφθεί από επιτυχία.
- Το ατύχημα στο πυρηνικό εργοστάσιο του 3-Mile Island των ΗΠΑ οφείλεται μερικά σε άσχημα σχεδιασμένη διεπαφή. Κάποια πινακίδα έκρυβε ένα λαμπάκι, το λαμπάκι έδειχνε λάθος, και όταν πολλά λαμπάκια άναβαν μαζί δεν ήταν εύκολο να παρακολουθήσει κανείς τι γινόταν.
- Απόλυτα αυτοματοποιημένα αεροπλάνα είναι επικίνδυνα γιατί σ' αυτά οι πιλότοι χαλαρώνουν (ουσιαστικά βαριόνται) και χάνουν την ετοιμότητά τους, όπως όταν οδηγούμε αυτοκίνητο σε μεγάλη ευθεία. Οι πιλότοι πρέπει κάθε στιγμή να έχουν τον έλεγχο του αεροπλάνου.
- Η κατάρριψη της Iran Air 655 το 1992 από Αμερικάνικη ρουκέτα θα είχε αποφευχθεί αν και η θέση και το ύψος του αεροσκάφους φαίνονταν στην ίδια οθόνη, γιατί ο συνδυασμός τους θα πιστοποιούσε ότι επρόκειτο για αεροπλάνο της πολιτικής αεροπορίας .
- Έχει παρατηρηθεί ότι άσχημα σχεδιασμένες διεπαφές μπορούν να προκαλέσουν σκολίωση, ερεθισμό ματιών, βαρεμάρα, μειωμένη αποδοτικότητα, και άλλα προβλήματα.

#### 1.4 Μερικά θέματα και αντίστοιχα πειράματα και παραδείγματα

Τα παρακάτω θέματα έχουν μεγάλη σημασία για την χρηστικότητα και αναφέρονται εδώ ενδεικτικά για να δώσουν την αίσθηση του τομέα της EAM. Το καθένα συνοδεύεται από κάποιο

μικρό παράδειγμα ή πείραμα που μπορεί να κάνει κανείς για να αντιληφθεί την ουσία του προβλήματος.

- Η μνήμη αντιδρά πολύ καλύτερα σε κάτι οπτικά γνώσιμο απ' ό,τι σε κάτι οπτικά άγνωστο. Σαν πείραμα, δείξτε σε κάποιον για 1 με 2 δευτερόλεπτα τις παρακάτω λέξεις:
  - ♦ σκουληκομυρμηγκότρυπα
  - ♦ ερβαλιστρεκομαρέθικτοΕίναι σχεδόν βέβαιο ότι θα θυμάται την πρώτη αλλά όχι την δεύτερη.

- Η μνήμη αντιδρά πολύ καλύτερα σε οργανωμένο κείμενο απ' ό,τι σε μη οργανωμένο κείμενο. Σαν πείραμα, δείξτε σε κάποιον για 2 με 4 δευτερόλεπτα τους παρακάτω αριθμούς:
  - ♦ 2653976208
  - ♦ 071 241 6378Είναι σχεδόν βέβαιο ότι θα θυμάται τον δεύτερο αλλά όχι τον πρώτο. Η σημαντική διαφορά είναι ότι στον δεύτερο αριθμό είναι ομαδοποιημένοι οι αριθμοί. Τέτοια δημιουργία ομάδων λέγεται *εγκλεισμός*. Εν γένει ο άνθρωπος μπορεί να θυμηθεί  $7 \pm 2$  πράγματα (άτομα ή ομάδες).

- Ο άνθρωπος αντιδρά καλύτερα στα απαλά χρώματα σε σχέση με τα έντονα, όσον αφορά την διάθεσή του και την παραγωγικότητά του. Δοκιμάστε αντιδράσεις σε μπλε-γκρι ή σκούρο πράσινο σε σχέση με έντονο κίτρινο ή ... λαχανί με βούλες! Αντίστοιχα η αντίληψή του είναι πολύ καλύτερη όταν χρησιμοποιούνται οι σωστοί συνδυασμοί χρωμάτων, όπως σε σχέση με το χρώμα του *υπόβαθρου*. Για παράδειγμα, συγκρίνετε μαύρο σε άσπρο σε σχέση με σκούρο κίτρινο σε ανοιχτό κίτρινο.

- Η παραγωγικότητα του ανθρώπου και ως προς την ποιότητα και ως προς την ποσότητα/ταχύτητα του έργου του επηρεάζεται άμεσα από μέγεθος των εργαλείων που χρησιμοποιεί (αναφορικά με το δικό του μέγεθος) και την αμεσότητα της ανάδρασης που παίρνει σαν αποτέλεσμα των ενεργειών του. Σαν πείραμα, δοκιμάστε να γράψετε μία φράση με ένα κανονικό μολύβι, με σκέτη τη μύτη ενός μηχανικού μολυβιού, με ένα τεράστιο μολύβι μήκους μισού μέτρου και διαμέτρου 5 εκατοστών, με μολύβι ίδιου χρώματος με του χαρτιού, κτλ. Τα αποτελέσματα είναι αρκετά διασκεδαστικά και οι χρόνοι που θα απαιτηθούν στην κάθε περίπτωση πιστοποιεί την ορθότητα της επιλογής που έχει γίνει για το μέγεθος των μολυβιών και των χρωμάτων χαρτιών/ μελάνης!

- Η χρηστικότητα μιας διεπαφής σχετίζεται άμεσα με την πληροφορία που προσφέρει στον χρήστη συνειδητά ή και υποσυνείδητα. Σαν πείραμα, όταν βρεθείτε σε χώρο με αρκετούς ανθρώπους, κοιτάξτε γύρω σας πόσοι φοράνε ψηφιακό ρολόι και πόσοι αναλογικό. Κατά πάσα πιθανότητα οι πρώτοι είναι πολύ λιγότεροι από τους δεύτερους. Τα πειράματα έχουν δείξει ότι ένας βασικός λόγος είναι ότι τα ψηφιακά ρολόγια δεν προσφέρουν καμία διαίσθηση για το πόσος χρόνος πέρασε από κάποια στιγμή ή απομένει για κάποια στιγμή. Αντίθετα, τα αναλογικά ρολόγια δίνουν υποσυνείδητα αυτήν την πληροφορία καθότι η ώρα παρουσιάζεται σε ένα υπόβαθρο ενός πλήρους 12ώρου (για τον ωροδείκτη) και ενός πλήρους 60λέπτου (για τον λεπτοδείκτη). Γι' αυτό και μετά από μία κάμψη στις πωλήσεις των κλασσικών ρολογιών όταν βγήκαν τα ψηφιακά (σε σημείο που κάποιοι προέβλεψαν ότι σύντομα θα εξαφανιστούν), αυτά επανήλθαν και έχουν την μεγάλη πλειοψηφία της αγοράς!

- Η αντίληψη κάποιας πληροφορίας από την μεριά του χρήστη επηρεάζεται σημαντικά από την *μεταφορά* που χρησιμοποιείται για την οπτική της αναπαράσταση, δηλαδή την συγκεκριμένη απεικόνιση της (συνήθως άγνωστης) πληροφορίας σε κάποιο οπτικό αντικείμενο (συνήθως γνώριμο ώστε ο χρήστης να μπορεί εύκολα να καταλάβει το νόημα της πληροφορίας). Ως παράδειγμα γνώριμης μεταφοράς σκεφθείτε την χρήση του κόκκινου και του μπλε για ζεστό και κρύο στις βρύσες, αντίστοιχα, ή των διαγραμμάτων Venn για υπερ/ υπο-σύνολα. Ως παράδειγμα μη γνώριμης μεταφοράς, σκεφθείτε το σύμβολο που χρησιμοποιείτε για την ένδειξη ραδιενεργών υλικών ή χώρων:





## Κεφάλαιο 2 Ο Άνθρωπος

### 2.1 Κανάλια Εισόδου-Εξόδου

Τα κανάλια εισόδου-εξόδου του ανθρώπου όσον αφορά τους υπολογιστές και άλλες μηχανές είναι τα αισθητήριά του, δηλ. τα όργανα των πέντε αισθήσεων του:

- Όραση, Ακοή, και Αφή (κυρίως)
- Όσφρηση και Γεύση (σπανίως)

Παρακάτω θα αναλύσουμε τα χαρακτηριστικά των δύο πρώτων, που είναι και οι σημαντικότερες σε θέματα επικοινωνίας ανθρώπου-υπολογιστή, καθώς και ένα βασικό θέμα γύρω από την κίνηση του χεριού, που είναι συχνά απαραίτητη για την επικοινωνία μεταξύ χρηστών και υπολογιστών.

#### Όραση

Η πληροφορία που δέχεται το μάτι περνά από πολλά στάδια επεξεργασίας για να ξεπεράσει ατέλειες του αρχικού οπτικού σήματος ή του ματιού και να καταλάβει την πραγματική φύση του ορατού αντικειμένου. Παρακάτω συζητούμε τέσσερα από τα πλέον σημαντικά σχετικά θέματα που προκύπτουν γύρω από αυτό.

##### *Μέγεθος και βάθος*

Υπάρχουν διάφοροι τρόποι με τους οποίους το μάτι αντιλαμβάνεται το βάθος σε μία 2-διάστατη εικόνα:

- Επικάλυψη αντικειμένων συνεπάγεται μεγαλύτερο βάθος του επικαλυπτόμενου από το επικάλυπτον αντικείμενο
- Μεγάλο μέγεθος συνεπάγεται μικρό βάθος
- Αντίθεση, καθαρότητα, και λαμπρότητα συνεπάγονται μικρό βάθος
- Κοκκώδης υφή συνεπάγεται μικρό βάθος

##### *Λαμπρότητα-φωτεινότητα (brightness)*

- Η λαμπρότητα είναι υποκειμενική
- Η αντίθεση (contrast) δημιουργείται από διαφορές στην φωτεινότητα
- Η αυξημένη φωτεινότητα δημιουργεί τρέμουλο και σε >50Hz αναβόσβημα

##### *Χρώμα*

- Το χρώμα έχει τρία χαρακτηριστικά
  - ♦ Μήκος κύματος, όπου τα τρία βασικά χρώματα έχουν την σχέση Μπλέ (B) < Πράσινο (G) < Κόκκινο (R)
  - ♦ Ένταση = φωτεινότητα
  - ♦ Κορεσμός = ποσότητα λευκού φωτός
- Το ανθρώπινο μάτι έχει σχετικά μικρή ευαισθησία στο μπλε
- Το 8% των ανδρών και το 1% των γυναικών πάσχει από αχρωματοψία (συνήθως αδυνατούν να ξεχωρίσουν το κόκκινο από το πράσινο)

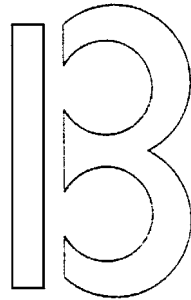
#### Κίνηση

Η κίνηση προκαλεί ταχύτατη αντίδραση ακόμη και όταν υπάρχει μόνο περιφεριακά του ματιού. Απαιτείται μόνο προσοχή ώστε να μην γίνει ενοχλητική όταν δεν χρειάζεται.

Λόγω των παραπάνω και άλλων χαρακτηριστικών της όρασης, ο τρόπος που αντιλαμβανόμαστε αυτό που βλέπουμε οπτικά παίζει διάφορα παιχνίδια:

**Περιβάλλον (context)**

Ο εγκέφαλος χρησιμοποιεί πολλές φορές προηγούμενη γνώση καθώς και το γενικότερο οπτικό περιβάλλον για να κατανοήσει αυτό που βλέπει. Για παράδειγμα, τι απεικονίζει το παρακάτω;



Ο εγκέφαλος θα το αντιληφθεί διαφορετικά, ανάλογα από το σε ποιο από τα δύο παρακάτω περιβάλλοντα εμφανίζεται:

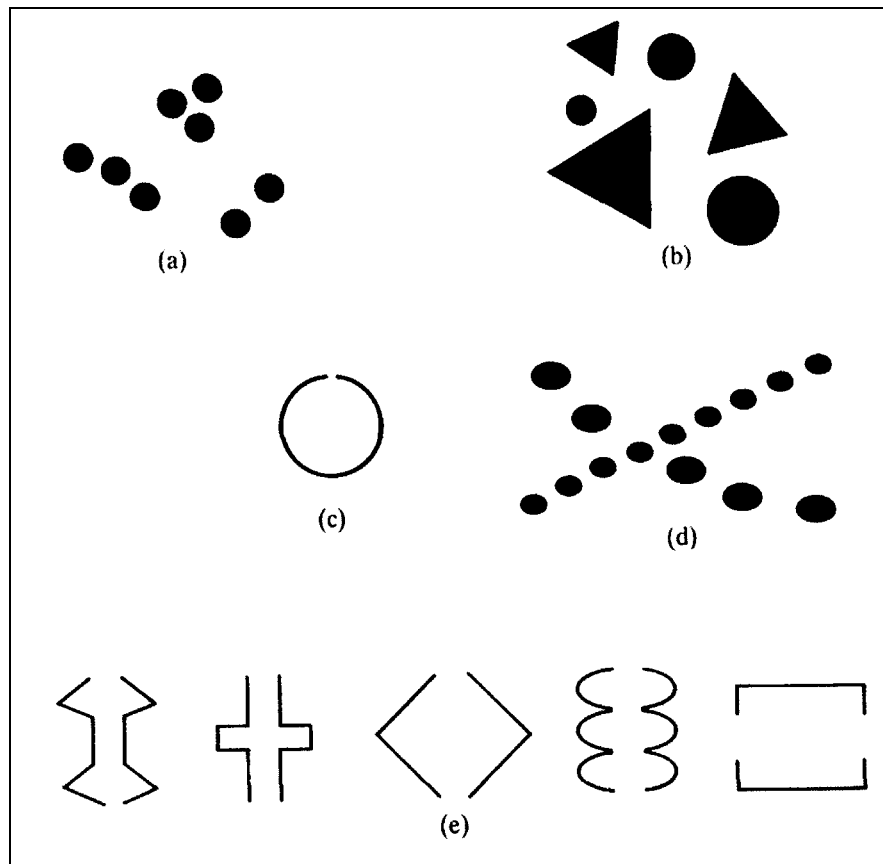


Ανάλογα, προσέξτε την διαφορετική ερμηνεία που δίνεται στο ίδιο σχήμα που εμφανίζεται σαν το μεσαίο γράμμα (Η ή Α) κάθε μιάς από τις παρακάτω λέξεις:

**ΜΑΝ ΠΑΣ**

**Ομαδοποίηση**

Ο εγκέφαλος έχει μία τάση προς ομαδοποίηση των αντικειμένων που βλέπει. Αυτά λέγονται *Κανόνες Gestalt* (από την ομώνυμη ψυχολογική σχολή). Χαρακτηριστικά παραδείγματα ομαδοποίησης ακολουθούν στην παρακάτω εικόνα:





Τα παραπάνω αντιπροσωπεύουν τους πέντε Κανόνες Gestalt, οι οποίοι επιτρέπουν την αντίληψη διαφόρων ατομικών αντικειμένων σαν μεγαλύτερες ολότητες:

- a. μικρή απόσταση
- b. ομοιότητα
- c. ολοκλήρωση (τμημάτων που λείπουν)
- d. συνέχεια (διακεκριμένων αντικειμένων)
- e. συμμετρία


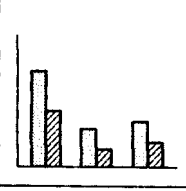
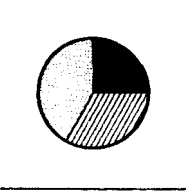
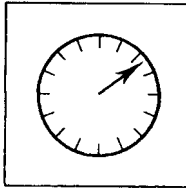
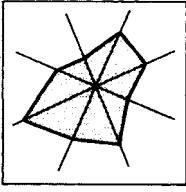
Από όλα τα παραπάνω συμπεραίνουμε ότι απαιτείται ιδιαίτερη προσοχή στις διάφορες παραμέτρους μιας οπτικής απεικόνισης (π.χ., το οπτικό περιβάλλον) ώστε να μεταδίδεται η σωστή εντύπωση.

Ένα άλλο θέμα που παίζει σημαντικό ρόλο στην οπτική αντίληψη είναι η οπτική/γραφική κωδικοποίηση που χρησιμοποιείται για να αναπαραστήσει αντικείμενα και χαρακτηριστικά τους με αντίστοιχα οπτικά αντικείμενα, π.χ., το χρώμα κόκκινο να απεικονίζει κίνδυνο ή ζέστη, το μέγεθος ενός οπτικού αντικείμενου να εξαρτάται από το πλήθος των μελών κάποιου συνόλου που απεικονίζει το οπτικό αντικείμενο, κτλ. Κάθε οπτική κωδικοποίηση έχει τα δικά της πλεονεκτήματα και μειονεκτήματα σε σχέση με το τί απεικονίζει και επομένως διαφορετική καταλληλότητα για κάθε περίπτωση. Οι παρακάτω πίνακες δίνουν ενδεικτικά τις ιδιότητες διαφόρων κωδικοποιήσεων και διαφόρων γραφικών αναπαραστάσεων δεδομένων, αντίστοιχα.

Μέθοδος Κωδικοποίησης	Μέγιστος αριθμός κωδικών	Σχόλια
Αλφαριθμητικά	Απεριόριστος	Πολύ ευέλικτο. Το νόημα μπορεί να είναι αυταπόδεικτο. Ο χρόνος εντοπισμού μπορεί να είναι μεγαλύτερος από της γραφικής κωδικοποίησης.
Σχήματα	10-20	Πολύ αποτελεσματικό εάν ο κωδικός ταιριάζει στο αντικείμενο ή στη λειτουργία που περιγράφεται.
Χρώμα	4-11	Συμπαθητικό και αποτελεσματικό. Υπερβολική χρήση προκαλεί σύγχυση. Περιορισμένο όφελος για άτομα με δυσχρωματοψία.
Γωνία γραμμής	8-11	Καλό σε ειδικές περιπτώσεις, για παράδειγμα, κατεύθυνση ανέμου.
Μήκος γραμμής	3-4	Καλό. Μπορεί να γεμίσει την οθόνη εάν παρουσιαστούν πολλοί κωδικοί.
Πλάτος γραμμής	2-3	Καλό.
Τύπος γραμμής	5-9	Καλό.
Μέγεθος αντικειμένου	3-5	Ικανοποιητικό. Μπορεί να χρειαστεί ιδιαίτερο χώρο. Ο χρόνος εντοπισμού είναι μεγαλύτερος από το σχήμα και το χρώμα.
Φωτεινότητα	2-4	Μπορεί να είναι κουραστικό, ειδικά εάν η αντίθεση της οθόνης είναι κακή.
Αναβόσβημα	2-4	Καλό για να τραβά την προσοχή αλλά θα πρέπει να είναι δυνατή η απόκρυψή του στη συνέχεια. Εκνευριστικό εάν χρησιμοποιηθεί πολύ. Περιορισμός σε μικρά πεδία.
Αντιστροφή εικόνας	Καθόλου δεδομένα	Αποτελεσματικό για να ξεχωρίσουν δεδομένα. Εάν είναι μεγάλη η περιοχή, το τρεμπάιγμα είναι πιο εύκολα αντιληπτό.
Υπογράμμιση	Καθόλου δεδομένα	Χρήσιμο αλλά μπορεί να μειώσει τη ευκρίνεια του κειμένου.
Συνδυασμός κωδικών	Απεριόριστος	Μπορεί να ισχυροποιήσει την κωδικοποίηση αλλά πολύπλοκοι συνδυασμοί μπορεί να προκαλέσουν σύγχυση.

Τεχνική γραφικών	Παράδειγμα	Σημειώσεις χρησιμοποίησης
Διασποράς		Εμφάνιση της συνδιακύμανσης δύο συνεχών μεταβλητών, ή της κατανομής των σημείων σε δι-διάστατο χώρο. Γραμμές ή καμπύλες μπορούν να χρησιμοποιηθούν ώστε να παρουσιαστούν τάσεις.
Γραμμών καμπύλων ή		Εμφάνιση συσχέτισης δύο συνεχών μεταβλητών, ιδιαίτερα σε μεταβολές μιας μεταβλητής σε σχέση με το χρόνο. Εάν χρησιμοποιηθεί ο χρόνος συνήθως τοποθετείται στον οριζόντιο άξονα. Μια τρίτη, διακριτή, μεταβλητή μπορεί να εισαχθεί με τη χρήση κωδικοποίησης χρώματος ή τύπου γραμμής. Κάποιοι σχεδιαστές προτείνουν τη χρήση το πολύ τεσσάρων γραμμών (ή καμπυλών) σε κάθε γράφημα. Όταν χρησιμοποιούνται πολλαπλές γραμμές, κάθε γραμμή μπορεί να έχει μια ετικέτα.



Περιοχής δέσμης ή		<p>Ειδικός τύπος γραφήματος που μπορεί να χρησιμοποιηθεί όταν αρκετά γραφήματα γραμμών ή καμπύλων αναπαριστούν τμήμα ενός συνόλου. Οι σκιασμένες γραμμές τοποθετούνται η μια πάνω στην άλλη και αναπαριστούν τη συμμετοχή κάθε κατηγορίας στο σύνολο. Οι μικρότερες καμπύλες τοποθετούνται στο κάτω μέρος του γραφήματος ώστε να αποτρέπουν ανωμαλίες στις καμπύλες. Οι κατηγορίες ονομάζονται μέσα στις σκιασμένες περιοχές.</p>
Ραβδογράμματα, γραφήματα στηλών ή ιστογράμματα		<p>Εμφανίζει τιμές μιας συνεχούς μεταβλητής για πολλαπλές διαφορετικές οντότητες ή μια μεταβλητή της οποίας παίρνονται δείγματα σε διακριτά χρονικά διαστήματα. Θα πρέπει να έχουν μια συνεπή διεύθυνση (οριζόντια ή κατακόρυφα) για σχετιζόμενα γραφήματα. Το διάστημα μεταξύ διπλανών ράβδων πρέπει να είναι μικρότερο από το πλάτος των ράβδων ώστε να διευκολύνονται οι συγκρίσεις μεταξύ ράβδων.</p>
Πίτα		<p>Παρουσιάζει τη σχετική κατανομή των δεδομένων μεταξύ τμημάτων που σχηματίζουν ένα σύνολο. Παρόλα αυτά, ένα ραβδόγραμμα ή γράφημα στηλών θα επιτρέψει συνήθως πιο ακριβή επεξήγηση. Εάν χρησιμοποιηθούν γραφήματα πίτας, μερικοί σχεδιαστές προτείνουν το πολύ πέντε τμήματα. Τα τμήματα θα πρέπει να ονομάζονται απευθείας και να ενσωματώνονται οι αριθμητικές τιμές που σχετίζονται με τα τμήματα.</p>
Μετρητές προσομοίωσης		<p>Εμφανίζει μια τιμή μιας συνεχούς μεταβλητής. Όταν παρουσιάζονται πολλαπλές τιμές (οπότε και πολλαπλοί μετρητές) που πρέπει να συγκρίνονται μεταξύ τους, είναι συνήθως πιο αποτελεσματική η χρήση κάποια άλλης τεχνικής όπως το ραβδόγραμμα ή το γράφημα στηλών για να δείξει την τιμή κάθε ξεχωριστής οντότητας ή το γράφημα γραμμής για να δείξει τη μεταβολή της τιμής σε συνάρτηση με το χρόνο.</p>
Αραχνοειδές		<p>Παρουσιάζει τις τιμές μιας συνεχούς μεταβλητής για πολλαπλές σχετιζόμενες οντότητες. Οι τιμές παρουσιάζονται σε spokes που φεύγουν από την αρχή. Διαφορετικές συνεχείς μεταβλητές μπορούν να παρουσιαστούν εάν έχουν εισαχθεί σε ευρετήριο ώστε οι κανονικές τιμές κάθε μεταβλητής να συνδεθούν για να σχηματίσουν ένα εύκολα αναγνωρίσιμο πολύγωνα. Χρήσιμο για τη πρόβλεψη σχημάτων, αλλά όχι για τον καθορισμό ακριβών τιμών ή τη διεξαγωγή ακριβών συγκρίσεων ανάμεσα σε τιμές.</p>

Σχετικά με τις κωδικοποιήσεις, θα θέλαμε να συζητήσουμε λίγο περισσότερο την χρήση του χρώματος λόγω κάποιων ιδιοτήτων του αλλά και λόγω της ελκυστικότητας που διαθέτει, με αποτέλεσμα να χρησιμοποιείται αρκετά έντονα από σχεδιαστές συστημάτων. Η χρήση του χρώματος για κωδικοποίηση ή άλλο λόγο θέλει προσοχή. Υπάρχει πάντα ο κίνδυνος για "χρωματική μόλυνση/ρύπανση", δηλαδή για υπερβολική χρήση χρώματος. Αυτό είναι ιδιαίτερα

επικίνδυνο όταν χρησιμοποιούνται πολύ κορεσμένα χρώματα (π.χ., γεμάτο κόκκινο, βαθύ μπλε, κτλ.). Εν γένει πρέπει να γίνεται συντηρητική χρήση της ποσότητας χρώματος στις διεπαφές. Επίσης πρέπει να λαμβάνονται υπ' όψη και οι παρακάτω πειραματικές διαπιστώσεις, ώστε η χρήση του χρώματος να αποφέρει μόνο θετικά αποτελέσματα:

- Το χρώμα είναι ιδιαίτερα χρήσιμο για διαχωρισμό περιοχών ή ομαδοποίηση περιοχών (π.χ., στην οθόνη) ώστε το ψάξιμο να είναι εύκολο.
- Αντίθετα, το χρώμα δεν είναι τόσο χρήσιμο σε κατηγοριοποιήσεις αντικειμένων, και κυρίως όταν υπάρχουν πολλές κατηγορίες, π.χ., περισσότερες από 5-10.
- Το χρώμα είναι πολύ πιο χρήσιμο σε άπειρους χρήστες παρά σε έμπειρους.

Η μέχρι τώρα συζήτησή μας για την όραση κατά βάση επικεντρώθηκε σε θέματα που αφορούν εικόνες, γραφικά, και εν γένει αντικείμενα μη αλφαριθμητικά. Τελειώνουμε με αυτό ακριβώς το θέμα, αναφερόμενοι σε τρία σημεία που σχετίζονται με ιδιότητες της όρασης όταν διαβάζουμε, τα οποία πρέπει να έχουν υπ' όψη οι σχεδιαστές συστημάτων.

- Το μάτι δεν είναι ποτέ ακίνητο. Ανά πάσα στιγμή, τρεμοπαίζει προς όλες τις κατευθύνσεις, πετάγεται δηλαδή πάνω-κάτω, δεξιά-αριστερά. Αυτό καθυστερεί την κοινή ανάγνωση καθώς ο αναγνώστης προσπαθεί να εστιάσει την προσοχή του στις διάφορες λέξεις. Το πρόβλημα μπορεί να μειωθεί σημαντικά όταν το μάτι έχει έναν οδηγό για το πού να εστιαστεί, π.χ., έχοντας το χέρι του αναγνώστη να σαρώνει το κείμενο ακριβώς κάτω από τις λέξεις που διαβάζονται (ή και με άλλες πιο προηγμένες κινήσεις). Όπως αναφέραμε και παραπάνω για το θέμα της κίνησης γενικά, το μάτι από τη φύση του παρακολουθεί την κίνηση του χεριού κάτω από τις λέξεις κι αποφεύγει κατά ένα μεγάλο μέρος το τρεμούλιασμα. Σε τέτοιες τεχνικές είναι που βασίζονται οι μέθοδοι γρήγορου διαβάσματος, οι οποίες, για το ίδιο επίπεδο κατανόησης του κειμένου, εν γένει αυξάνουν την ταχύτητα ανάγνωσης από 150/250 λέξεις το λεπτό σε πάνω από 1000 σε πολλές περιπτώσεις.
- Οι γνωστές λέξεις ή φράσεις αναγνωρίζονται μορφολογικά και δεν διαβάζονται γράμμα-γράμμα ή συλλαβή-συλλαβή. Αυτό το είδαμε και στο Κεφάλαιο 1, με το πείραμα της ανάγνωσης των λέξεων
  - ♦ σκουληκομυρμηγκότρυπα
  - ♦ ερβαλιστρεκομαρέθικτο
 Καθότι η μορφολογική ανάγνωση είναι πιο γρήγορη, όποτε η ταχύτητα είναι το ζητούμενο, η χρήση γνωστών λέξεων είναι πολύ βοηθητική.

Οι σκούροι χαρακτήρες σε ανοιχτό υπόβαθρο διαβάζονται πολύ καλύτερα απ' ό,τι οι ανοιχτοί χαρακτήρες σε σκούρο υπόβαθρο. Οπότε οι παρουσιάσεις κειμένων πρέπει εν γένει να είναι ανάλογες.

## Ακοή

Όπως είναι γνωστό, ο ήχος έχει τρία χαρακτηριστικά, την *συχνότητα* (όπου ο άνθρωπος μπορεί να ακούσει συχνότητες μεταξύ 20Hz και 15KHz), την *ένταση*, και την *χροιά*. Ένα σημαντικό φαινόμενο που επηρεάζει το πώς μπορεί να χρησιμοποιηθεί ο ήχος σε συστήματα διεπαφής είναι το λεγόμενο *φαινόμενο του πάρτυ* (*coctail party phenomenon*), μέσα σε τρομερή φασαρία μπορεί να ξεχωρίσει κανείς κάτι πολύ γνώριμο, όπως για παράδειγμα το όνομά του. Στην επόμενη ευκαιρία, δοκιμάστε το!

Με βάση τα συγκεκριμένα χαρακτηριστικά της ακοής αλλά κυρίως διάφορες πειραματικές παρατηρήσεις, τα παρακάτω αποτελούν κάποιες βασικές σωστές χρήσεις του ήχου σε διεπαφές:

- προσέλκυση της προσοχής του χρήστη σε κρίσιμες στιγμές, π.χ., όταν προκύπτει κάτι επικίνδυνο, ή σε στιγμές που κάποια διαδικασία τελειώνει, π.χ., όταν κάτι είναι μακροχρόνιο και η προσοχή του χρήστη μπορεί να βρίσκεται κάπου αλλού εν τω μεταξύ.
- πληροφόρηση της κατάστασης μίας διαδικασίας επί συνεχούς βάσης, π.χ., όπως το καμπανάκι που χτυπά σε αρκετούς ανελκυστήρες καθώς περνάμε από κάθε όροφο.
- επιβεβαίωση της πραγματοποίησης κάποιας ενέργειας του χρήστη, π.χ. εκτέλεση εντολής, σβήσιμο αρχείου, λάθος εντολή, κτλ.
- καθοδήγηση στη διάρκεια εξερεύνησης

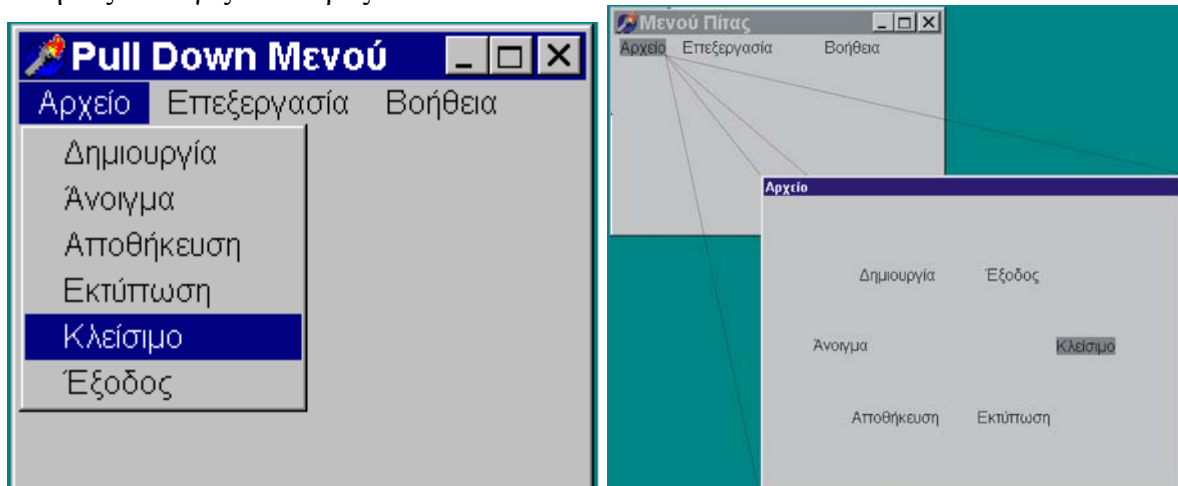
## Κίνηση

Στην επικοινωνία του ανθρώπου με τον υπολογιστή, πολλές φορές απαιτείται κίνηση του χεριού προς κάποια κατεύθυνση, π.χ., κίνηση ποντικιού ώστε ο κέρσορας να πάει σε συγκεκριμένο σημείο. Ο χρόνος κίνησης επηρεάζεται από πολλά πράγματα και δεν μπορεί να προβλεφθεί απόλυτα. Εμπειρικά υπολογίζεται από τον *Κανόνα του Fitts (Fitts' Law)*, που έχει πολλές σχετικά παραπλήσιες μορφές, εκ των οποίων η πιο συνηθισμένη είναι

$$\alpha + \beta \log_2 (1 + \text{απόσταση/μέγεθος})$$

όπου 'απόσταση' είναι το μήκος της κίνησης από το αρχικό σημείο στο τελικό, 'μέγεθος' είναι το μέγεθος του αντικειμένου το οποίο αποτελεί τον στόχο της κίνησης (π.χ., το μέγεθος του παραθύρου στην οθόνη που θέλουμε να πάει ο κέρσορας), και 'α' και 'β' είναι εμπειρικές παράμετροι.

Η παραπάνω εμπειρική μαθηματική έκφραση επηρεάζει το πώς πρέπει να σχεδιάζονται οι στόχοι προς τους οποίους υπάρχει ανάγκη να υπάρχει κίνηση. Για παράδειγμα, έχει δειχθεί ότι οι επιλογείς σε μορφή πίτας, όπως η παρακάτω εικόνα δεξιά, είναι καλύτεροι από τους κατακόρυφους επιλογείς διότι όλες οι επιλογές βρίσκονται στην ίδια απόσταση από το αρχικό σημείο εκκίνησης. Παράλληλα, στους κατακόρυφους επιλογείς, όπως ο επιλογέας στην παρακάτω εικόνα αριστερά, οι πιο συνηθεις επιλογές είναι προς τα πάνω.



Ερώτηση: Σύμφωνα με τον κανόνα του Fitts ο χρόνος κίνησης του δείκτη του ποντικιού εξαρτάται από το μέγεθος του στόχου και την απόσταση στην οποία βρίσκεται. Αυτό σημαίνει ότι ο μέσος χρόνος επιλογής μιας εντολής ενός επιλογέα (μενού) αυξάνει όσο αυξάνει και ο αριθμός των επιλογών. Αντίστοιχα ο χρόνος επιλογής ενός πλήκτρου που βρίσκεται σε ένα αναδυόμενο

παράθυρο αυξάνει όσο μικραίνει το μέγεθός του και αυξάνει η απόσταση του από το δείκτη του ποντικιού. Μπορείτε να προτείνετε λύσεις που να μειώνουν το χρόνο επιλογής στις παραπάνω περιπτώσεις; Τι προβλήματα πιστεύετε ότι είναι δυνατόν να προκύψουν;

## 2.2 Ανθρώπινη Μνήμη

Ο άνθρωπος έχει μνήμη τριών επιπέδων: μνήμη αισθήσεων, μνήμη εργασίας ή ενδιάμεση μνήμη, και μακροπρόθεσμη μνήμη. Εξηγούμε περιληπτικά την κάθε μία παρακάτω, και μετά εξετάζουμε λεπτομερώς κάποιες ιδιότητες της μακροπρόθεσμης μνήμης που είναι και η πιο σημαντική για την επικοινωνία ανθρώπου-μηχανής.

### Μνήμη Αισθήσεων

Η μνήμη αισθήσεων περιέχει το τελευταίο ερέθισμα του αντίστοιχου αισθητηρίου, π.χ., ματιού, αυτιού, κτλ. Η καταχωρήσεις σ' αυτού του είδους τη μνήμη διατηρούνται για πολύ μικρά χρονικά διαστήματα. Για παράδειγμα, η αντίστοιχη μνήμη του ματιού διατηρεί το όποιο οπτικό ερέθισμα για λιγότερο από 0.5 δευτερόλεπτο. Η ύπαρξη αυτής της μνήμης μπορεί εύκολα να διαπιστωθεί με μικρά πειράματα. Για την μνήμη ματιού, αν κινήσουμε το δάχτυλό μας μπροστά στο μάτι μας πολύ γρήγορα, τότε ανά πάσα στιγμή "βλέπουμε" πολλά είδωλά του, τα οποία εκτός του πλέον πρόσφατου αντιπροσωπεύουν προηγούμενες καταχωρήσεις του στην μνήμη του ματιού. Αντίστοιχα για την μνήμη αυτιού, πολλές φορές ρωτάμε κάποιον να μας επαναλάβει κάτι που ανέφερε, και διαπιστώνουμε τελικά ότι όντως το είχαμε ακούσει (δηλαδή το βρίσκουμε στην μνήμη του αυτιού μας) αν και δεν του είχαμε δώσει σημασία.

### Μνήμη Εργασίας/Ενδιάμεση Μνήμη

Η μνήμη εργασίας περιέχει ένα μικρό αριθμό πραγμάτων που τα θέλουμε στο έργο που επιτελούμε εκείνη την ώρα, κάτι σαν το "πρόχειρο" του διαγωνίσματος. Η χωρητικότητα της μνήμης αυτής είναι περίπου 7 +/- 2 αντικείμενα. Όπως αναφέραμε και στο 1ο Κεφάλαιο, όταν ομαδοποιηθούν τα αντικείμενα, τότε αυξάνεται η χωρητικότητα σε περίπου 7 +/- 2 ομάδες αντικειμένων (το φαινόμενο του *εγκλεισμού*). Αυτό εύκολα μπορεί να διαπιστωθεί αν προσπαθήσετε για παράδειγμα να θυμηθείτε μία απλή σειρά από 11 ψηφία, π.χ.,

16082637764

ή έναν αριθμό τηλεφώνου που επίσης περιέχει τα ίδια 11 ψηφία δοσμένα στην συνήθη μορφή

+1 (608) 263-7764

αλλά τα οποία είναι ομαδοποιημένα σε κωδικό χώρας (1), κωδικό πόλης (608), κωδικό τηλεφωνικού κέντρου (263), και αριθμό μέσα στο τηλεφωνικό κέντρο (7764).

### Μακροπρόθεσμη Μνήμη

Η μακροπρόθεσμη μνήμη περιέχει όλη τη γνώση μας. Έχει ουσιαστικά "άπειρη" χωρητικότητα και διατηρείται ουσιαστικά "επ' άπειρον". Ο χρόνος προσπέλασής της είναι περίπου 0.1 δευτερόλεπτα. Λόγω της σημασίας της μακροπρόθεσμης μνήμης, θα ασχοληθούμε κάπως αναλυτικότερα με τις διεργασίες της, και συγκεκριμένα θα αναλύσουμε τις διαδικασίες εισαγωγής πληροφοριών σ' αυτήν, διαγραφή πληροφοριών, και προσπέλαση πληροφοριών.

*Εισαγωγή:* Η επανάληψη εκμάθησης μίας πληροφορίας, την μεταφέρει από την μνήμη εργασίας στην μακροπρόθεσμη μνήμη. Η ποσότητα που αποθηκεύεται είναι ανάλογη του χρόνου που καταναλώνεται στην εκμάθηση. Αυτό επηρεάζεται επίσης και από τον χρονοκαταμερισμό της εκμάθησης, καθότι κατανεμημένος χρόνος εκμάθησης φέρνει καλύτερα αποτελέσματα σε σχέση με την συγκεντρωμένη χρονικά εκμάθηση.

Ένα άλλο στοιχείο που επηρεάζει την εισαγωγή πληροφοριών στην μνήμη είναι και η φύση των πληροφοριών αυτών. Η οπτικοποιήσιμη πληροφορία είναι πολύ πιο εύκολη να την μάθουμε σε σχέση με πιο αφηρημένη πληροφορία. Σαν πείραμα δοκιμάστε σε μικρό χρονικό διάστημα, π.χ., 5 δευτερόλεπτα, να απομνημονεύσετε τις λέξεις στη λίστα

(σκέψη, ηλικία, κρύο, πονηριά, ησυχία, λογική, ιδέα, αξία, παρελθόν, μεγάλο)

και αυτές στη λίστα

(βάρκα, δένδρο, γάτα, παιδί, χαλί, πιάτο, εκκλησία, όπλο, φλόγα, κεφάλι).

Είναι σχεδόν βέβαιο ότι η πρώτη λίστα θα φέρει μεγαλύτερες δυσκολίες από την δεύτερη. Ακόμη ευκολότερη είναι η απομνημόνευση πληροφοριών που αποτελούν προτάσεις, και ακόμη περισσότερο ολόκληρες ιστορίες ή ποιήματα. Αυτά δημιουργούν έντονη εικονική πληροφορία (vivid imagery) που απομνημονεύεται εύκολα. Γι' αυτό, ακόμη και όταν η πληροφορία αυτή καθ' εαυτή δεν έχει αυτήν την μορφή, πολλές φορές είναι καλό να φτιάχνει κανείς μία ιστορία για παράδειγμα που περιτυλίσσει την πληροφορία με τρόπο που την κάνει πιο εύκολα αποδεκτή. Δοκιμάστε να απομνημονεύσετε τις λέξεις στην λίστα

(αίμα, κρύα, μεγάλο, θυμωμένο, παιδί, δένδρο, αεροπλάνο, σκύλος, φίλος)

έτσι όπως έχει δοθεί και επίσης αφού φτιάξετε μία ιστορία που να τις περιέχει. Η διαφορά θα είναι εμφανής.

*Διαγραφή:* Εν γένει είναι άγνωστο το αν ποτέ σβήνεται η πληροφορία από την μνήμη ή απλά αυξάνει ο χρόνος προσπέλασής της. Αν υπάρχει όντως διαγραφή, τότε υπάρχουν διάφορες θεωρίες για την πλήρη απόσβεση ή την αντικατάσταση (π.χ., μαθαίνοντας το καινούργιο τηλέφωνο κάποιου φίλου, ξεχνάμε το παλιό), καθώς και από το πώς επηρεάζονται αυτά από τα συναισθήματά μας (π.χ., υπάρχει η τάση να θυμόμαστε μόνο τα καλά γεγονότα).

*Προσπέλαση:* Υπάρχουν δύο μορφές προσπέλασης στην ανθρώπινη μνήμη, η *επανάκτηση (recall)*, όπου η πληροφορία προσπελάσσεται στην μνήμη εξ αρχής και χωρίς καμία βοήθεια, και η *αναγνώριση (recognition)*, όπου η πληροφορία μας δίνεται με κάποιο τρόπο ίσως ανάμεσα σε πολλές άλλες και εμείς απλά συνειδητοποιούμε τι είναι αυτό που ψάχνουμε. Και οι δύο μορφές προσπέλασης, και κυρίως η επανάκτηση, βοηθιούνται από αυτά που αναφέρθηκαν και σχετικά με την εισαγωγή πληροφοριών στην μνήμη, δηλαδή για τη δημιουργία ιστορίας που να περιέχει την πληροφορία, κτλ.

Η αναγνώριση είναι πολύ ευκολότερη και αποδοτικότερη από την επανάκτηση. Για παράδειγμα, δοκιμάστε να θυμηθείτε τηλέφωνα φίλων σας με τους οποίους έχετε καιρό να επικοινωνήσετε (επανάκτηση) και μετά δείτε πόσα από αυτά τα τηλέφωνα θα αναγνωρίσετε σε μία ανώνυμη λίστα τηλεφώνων που θα σας δώσει κάποιος (αναγνώριση). Η δεύτερη μέθοδος θα είναι πολύ πιο επιτυχής. Γι' αυτόν τον λόγο, και οι διεπαφές τείνουν προς τα εκεί, π.χ., δεν χρειάζεται πλέον να θυμάται κανείς ονόματα εντολών αλλά τις αναγνωρίζει καθώς εμφανίζονται σε κάποιο επιλογέα. Αντίστοιχα, καθότι η αναγνώριση είναι ευκολότερη, οπτικές και μη κωδικοποιήσεις πρέπει να είναι τέτοιες που να θυμίζουν το κωδικοποιημένο αντικείμενο. Δυστυχώς πολλά αντιπαραδείγματα μπορούν να βρεθούν ανάμεσα στις εντολές αρκετών συστημάτων. Τέτοιες είναι οι εντολές του Unix `cat` (συντομογραφία της φράσης `catenate and print`), `grep` (ακρώνυμο της φράσης `global regular expression and print`), `lpr` (ακρώνυμο της φράσης `line printer`, που μάλιστα αναφέρεται και συγκεκριμένη τεχνολογία ενώ χρησιμοποιείται πολύ ευρύτερα), `biff` (το όνομα του σκύλου της προγραμματίστριας που υλοποίησε την εντολή στο Unix), και άλλες, των οποίων τα ονόματα βέβαια καμία σχέση δεν έχουν με τις διαδικασίες που αντιπροσωπεύουν. Η σωστή επιλογή εντολών και κυρίως εικονιδίων παίρνει υπόψη της το περιβάλλον στο οποίο θα χρησιμοποιηθεί, την γενικότερη δραστηριότητα/εργασία του χρήστη όταν αντιμετωπίζει την εντολή/εικονίδιο, την παράσταση αυτή καθ' εαυτή (στην περίπτωση εικονιδίων), και την αναπαριστάμενη έννοια/ενέργεια, όπως αναλύουμε παρακάτω.

- Περιβάλλον - συμφραζόμενα: Σαν παράδειγμα της σημασίας του περιβάλλοντος, σκεφθείτε τους διαφορετικούς συνειρμούς που δημιουργούν εικονίδια με γυναικεία και ανδρικά παπούτσια όταν βρίσκονται σε ένα εστιατόριο (όπου μάλλον υπονοούν τουαλέτες) ή σε ένα πολυκατάστημα (όπου μάλλον υπονοούν καταστήματα υποδημάτων).
- Εργασία: Τα εικονίδια είναι κατάλληλα για μη αλφαβητική πληροφορία, κυρίως όταν απαιτείται αναγνώριση.
- Παράσταση: Η παράσταση πρέπει να είναι "μνημονική". Συνήθως, οι παραστάσεις που χρησιμοποιούνται συνδέονται με την αναπαριστάμενη έννοια/ενέργεια είτε με άμεση ομοιότητα, είτε αποτελώντας παραδείγματά της, είτε καθαρά συμβολικά, ενώ σε κάποιες περιπτώσεις μπορεί και να είναι εντελώς τυχαίες και ασύνδετες με αυτήν. Παραδείγματα των τεσσάρων αυτών κατηγοριών βρίσκονται στον παρακάτω πίνακα.



Ομοιότητα: Βράχοι που πέφτουν (κοντά σε γκρεμό)



Παράδειγμα: Μαχαιροπήρουνα (σαν ένδειξη εστιατορίου)



Συμβολισμός: Ποτήρια (σαν ένδειξη "εύθραυστο")



Τυχαία: Κύκλος ραδιενέργειας

- Αναπαριστώμενη έννοια: Είναι ευκολότερα τα αντικείμενα και δυσκολότερες οι αφηρημένες έννοιες και οι πράξεις (όπου επιστρατεύονται πολλές φορές έμμεσες αναλογίες, π.χ. ένας ελέφαντας για την ένδειξη "βαρύ", οι οποίες απαιτούν αρχική εκμάθηση). Καλός συνδυασμός είναι τα εικονίδια κίνησης (animated icons).

### 2.3 Σκέψη και Επίλυση Προβλημάτων

Ο άνθρωπος χρησιμοποιεί πολλές μεθόδους όταν σκέφτεται και όταν επιλύει προβλήματα. Η γνώση αυτών είναι βοηθητική στο να καταλάβουμε μερικές φορές πώς πρέπει να φτιάξουμε μία διεπαφή ώστε να καθοδηγούμε τον χρήστη σε έναν συγκεκριμένο τρόπο σκέψης ή/και ώστε να αποφεύγουμε τα προβλήματα που πολλές φορές δημιουργούν αυτοί οι τρόποι. Παρακάτω αναφέρονται τρεις σημαντικοί τρόποι σκέψης και τρεις βασικοί τρόποι επίλυσης προβλημάτων.

*Συμπερασματική σκέψη (deductive reasoning)*: σ' αυτήν χρησιμοποιούνται κανόνες και δεδομένα (αίτια) προς το εμπρός. Για παράδειγμα:

Εάν είναι Παρασκευή θα πάει στη δουλειά	$A \Rightarrow B$
Είναι Παρασκευή	A
Άρα, θα πάει στη δουλειά	B

Παρότι πολύ λογικός και χρήσιμος αυτός ο τρόπος, μπορεί να μπερδέψει:

Κάποιοι άνθρωποι είναι μωρά

Κάποια μωρά κλαίνε

Άρα, κάποιοι άνθρωποι κλαίνε.

Αν και στην πραγματικότητα το συμπέρασμα είναι σωστό, προϋποθέτει ότι όλα τα μωρά είναι άνθρωποι, το οποίο δεν είναι δεδομένο παραπάνω. Σε κάποιες άλλες περιπτώσεις αυτό το πρόβλημα επιφέρει λάθος συμπεράσματα.

*Επαγωγική σκέψη (inductive reasoning)*: σ' αυτήν δημιουργούνται γενικεύσεις από παραδείγματα. Για παράδειγμα:

Εάν όλοι οι ελέφαντες που είδαμε έχουν προβοσκίδα, τότε όλοι οι ελέφαντες έχουν.

*Απαγωγική σκέψη (abductive reasoning)*: σ' αυτήν χρησιμοποιούνται κανόνες και αποτελέσματα (αιτιατά) προς τα πίσω. Για παράδειγμα, μπορεί να χρησιμοποιηθεί ανάποδα το παράδειγμα της Παρασκευής. Μπορεί όμως να οδηγήσει σε λάθη όπως ακριβώς στο παράδειγμα, όπου το ότι θα πάει στην δουλειά δεν σημαίνει απαραίτητα ότι είναι Παρασκευή.

*Παραγωγική και αναπαραγωγική επίλυση προβλημάτων (productive and reproductive problem solving)*: αυτή είναι η θεωρία της σχολής Gestalt. Στην αναπαραγωγική επίλυση, ο άνθρωπος χρησιμοποιεί παλιές εμπειρίες για να βρει την λύση, ενώ στην παραγωγική χρησιμοποιεί την διαίσθησή του και ανακατασκευάζει το πρόβλημα. Χαρακτηριστικό παράδειγμα είναι το λεγόμενο πρόβλημα του εκκρεμούς του Maier (Maier's pendulum problem), όπου δύο σκοινιά κρέμονται από την οροφή, δίνονται πένσες, σφυριά και διάφορα άλλα εργαλεία, και ο στόχος είναι να δεθούν τα δύο σκοινιά. Το πρόβλημα όμως είναι ότι τα δύο σκοινιά είναι αρκετά μακριά το ένα από το άλλο για να μπορέσει κάποιος να πιάσει τις δύο άκρες ταυτόχρονα. Η λύση είναι να κρεμάσει κανείς στο κάθε σχοινί από μία πένσα και μετά να τις σπρώξει έτσι ώστε με το βάρος τους να κινήσουν τα σχοινιά έτσι ώστε να μπλεχτούν μεταξύ τους. Σε διάφορα πειράματα όμως, σχεδόν κανείς δεν το σκέφτεται αυτό. Όταν όμως κινήσει κάποιος το σχοινί, τότε αρκετοί καταφέρνουν να δουν αυτήν τη λύση και την εφαρμόζουν. Αυτό είναι ένα παράδειγμα παραγωγικής επίλυσης, όπου η τυχαία κίνηση του σχοινιού δημιουργεί την κατάλληλη διαίσθηση ώστε να χρησιμοποιηθούν οι πένσες με τελείως διαφορετικό τρόπο από αυτόν για τον οποίον είναι κατά βάση φτιαγμένες.

*Επίλυση με την θεωρία του χώρου προβλήματος (problem space theory)*: σ' αυτήν την θεωρία, δημιουργείται νοητικά ένας γράφος καταστάσεων και μεταβάσεων μεταξύ τους, τον οποίον παρακολουθεί ο άνθρωπος για να λύσει το πρόβλημά του. Για παράδειγμα, αν από την οροφή κρέμεται μία μπανάνα, υπάρχει ένα γραφείο στο δωμάτιο, και ένας άνθρωπος πεινάει αλλά δεν φτάνει την μπανάνα, μπορεί κανείς να κατασκευάσει έναν απλό γράφο με διάφορες καταστάσεις που απεικονίζουν η κάθε μία διαφορετικές τοποθετήσεις της μπανάνας, του γραφείου, και του ανθρώπου, και να βρει την λύση του να μετακινήσει το γραφείο κάτω από την μπανάνα, να ανεβεί σ' αυτό, και να την φτάσει.

*Επίλυση με αναλογία*: σ' αυτόν τον τρόπο επίλυσης προβλημάτων, διαφαίνεται μία αντιστοιχία μεταξύ δύο προβλημάτων, οπότε η γνωστή λύση του ενός ανασκευάζεται ώστε να χρησιμοποιηθεί σαν λύση στο άλλο. Για παράδειγμα, ας υποθέσουμε ότι το παρακάτω [Dix, σελ. 43] είναι γνωστό:

Ένας στρατηγός επιτίθεται σε ένα κάστρο. Δεν μπορεί να στείλει όλους τους ανθρώπους του μαζί στην επίθεση γιατί οι δρόμοι έχουν νάρκες που θα εκραγούν αν περάσουν πολλοί άνθρωποι ταυτόχρονα από πάνω τους. Γι' αυτό, δημιουργεί μικρές

ομάδες οι οποίες ακολουθώντας ξεχωριστούς δρόμους φτάνουν πάλι στον στόχο τους όλες μαζί χωρίς όμως να διατρέχουν κίνδυνο.

Ας θεωρήσουμε τώρα ότι αντιμετωπίζουμε το παρακάτω πρόβλημα:

Ένας γιατρός προσπαθεί να καταστρέψει έναν κακοήθη όγκο με ισχυρή ακτινοβολία. Υπάρχει όμως το πρόβλημα ότι η πλήρης απαιτούμενη δοσολογία θα καταστρέψει και όλα τα υγιή κύτταρα που θα βρει στον δρόμο της. Από την άλλη, μειωμένη δοσολογία θα είναι μεν ακίνδυνη για τα υγιή κύτταρα αλλά και αναποτελεσματική για τον όγκο.

Χρησιμοποιώντας την εμφανή αναλογία με το πρώτο πρόβλημα, εύκολα κανείς μπορεί να φτάσει στην λύση που υπαγορεύει ότι ο όγκος πρέπει να χτυπηθεί από πολλές μεριές ταυτόχρονα με ακτινοβολίες μικρής ισχύος που δεν επηρεάζουν τα υγιή κύτταρα, αλλά που συνολικά φτάνοντας στον όγκο αθροίζονται στην σωστή δοσολογία. Σε διάφορα πειράματα που έχουν γίνει, η άγνοια του πρώτου προβλήματος και της λύσης του κάνουν την επίλυση του δεύτερου αρκετά σπάνιο φαινόμενο.

## 2.4 Μεταφορές

Η απεικόνιση καταστάσεων, αντικειμένων, ή ενεργειών σε αντίστοιχα "γνώριμα" για την διευκόλυνση της μάθησης και της θύμησης λέγεται *μεταφορά* (*metaphor*). Πειραματικά αποτελέσματα δείχνουν ότι το να έχουν την μεταφορά υπ' όψη τους οι χρήστες αυξάνει την απόδοσή τους. Μπορούμε να διακρίνουμε δύο ειδών μεταφορές:

- Λεκτικές μεταφορές, στις οποίες πραγματοποιείται μία νοητική απεικόνιση. Παραδείγματα αποτελούν η γραφομηχανή σαν απεικόνιση του σύγχρονου υπολογιστή, ή η φράση "θα σου κόψω τα πόδια" σαν απεικόνιση διάθεσης τιμωρίας και εκδίκησης.
- Εικονικές μεταφορές διεπαφής, στις οποίες γίνεται ηλεκτρονική αναπαράσταση αντικειμένων και ενεργειών στον υπολογιστή.

Οι μεταφορές είναι εν γένει αποτελεσματικές έστω και με μικρές αναντιστοιχίες:

- Το πλήκτρο Backspace/Delete στη γραφομηχανή απλά τοποθετεί το κέντρο του κινητού σώματος της γραφομηχανής πριν από το τελευταίο στοιχείο που τυπώθηκε, ενώ αντίθετα το αντίστοιχο πλήκτρο στον υπολογιστή σβήνει και το τελευταίο στοιχείο.
- Η μεταφορά του εικονιδίου μιας δισκέτας στο εικονίδιο του καλαθιού των αχρήστων θα έπρεπε απλά να σημαίνει σβήσιμο του περιεχομένου της δισκέτας, αλλά σε αρκετά συστήματα σημαίνει την απελευθέρωση της δισκέτας από την σχισμή της (disk eject).

Σύνθετες μεταφορές όπου χρησιμοποιούνται πολλές απεικονίσεις ταυτόχρονα είναι χρήσιμες για την πραγματικότητα, π.χ., η χρήση της επιφάνειας του γραφείου (desktop) με ράβδο κύλισης (scrollbar) σαν απεικόνιση του χώρου εργασίας ενός παραθύρου στην οθόνη. Ο παρακάτω πίνακας δίνει διάφορα παραδείγματα μερικών χαρακτηριστικών μεταφορών [Preece, σελ. 149].

Μεταφορά	Γνώριμη Γνώση	Περιοχή Εφαρμογής
Επιφάνεια γραφείου	Λειτουργίες γραφείου, αρχείων	Περιβάλλον εργασίας
Βιβλίο εσόδων/εξόδων	Πίνακες με γραμμές και στήλες	Φύλλα υπολογισμού (Spreadsheets)
Φυσικός κόσμος	Κοινή συμπεριφορά	Αντικειμενοστρεφή περιβάλλοντα
Καρτέλες (notecards)	Δομημένο κείμενο	Υπερκείμενο
Ταξίδια	Ταξιδιωτική εμπειρία	Περιβάλλοντα εκμάθησης
Στοιβές	Προτεραιότητες	Αποθήκευση αρχείων
Δωμάτια	Χωροταξική δομή κτηρίων	Περιβάλλοντα πολυμέσων
Πράκτορες	Υπηρεσίες εξυπηρέτησης	Συνεργασία με βοήθεια υπολογιστή



## Κεφάλαιο 3 Ο Υπολογιστής (Η Μηχανή)

Τα περισσότερα χαρακτηριστικά του υπολογιστή μας είναι εν γένει οικεία, οπότε και δεν θα μπορούμε σε πολλές λεπτομέρειες. Για πληρότητα και μόνο, θα αναφερθούμε στα πλέον σημαντικά για την επικοινωνία ανθρώπου-μηχανής από αυτά.

### 3.1 Μονάδες εισόδου

Οι μονάδες εισόδου έχουν σημασία όχι μόνο για εισαγωγή πληροφοριών και εντολών στον υπολογιστή αλλά και για ανάδραση (feedback) που δίνεται στον χρήστη με βάση την ενέργειά του, η οποία μπορεί να είναι οπτική ή ακουστική. Είναι πολύ σημαντικό η ανάδραση αυτή να γίνεται σε χρόνο λιγότερο από 2 δευτερόλεπτα, που αποτελεί το χρονικό διάστημα μετά από το οποίο (όπως έδειξαν διάφορα πειράματα) ο χρήστης γίνεται ανυπόμονος και αρχίζει να αναρωτιέται αν πέτυχε η είσοδος την οποία επιχείρησε ή όχι. Είδ' αλλιώς είναι πιθανό να υπάρξει επανάληψη της ενέργειας εισόδου πληροφοριών ή εντολών από πλευράς χρήστη, που σε συνδυασμό με πιθανή αποθήκευσή τους σε ενδιάμεση μνήμη από πλευράς υπολογιστή (buffering), μπορεί να φέρει προβλήματα από την πολλαπλή εκτέλεση των ίδιων πραγμάτων. Υπάρχουν πολλών διαφορετικών τύπων μονάδες εισόδου, όπως αναφέρονται συνοπτικά παρακάτω.

#### Πληκτρολόγια

- QWERTY: Είναι τα κλασσικά πληκτρολόγια, όπου η τοποθέτηση των πλήκτρων έγινε με βάση την αντίστοιχη της γραφομηχανής. Το όνομά του προέρχεται από τα έξι πρώτα γράμματα από αριστερά της πρώτης από πάνω σειράς πλήκτρων του πληκτρολογίου που αντιστοιχούν σε γράμματα. Η τοποθέτηση των πλήκτρων σ' αυτό έγινε με βάση τις ανάγκες των παλιών τεχνολογίας γραφομηχανών. Συγκεκριμένα ο στόχος ήταν να ελαχιστοποιηθεί η πιθανότητα της εναλλαγής πληκτρολόγησης από την αριστερή και δεξιά μεριά των πλήκτρων, της οποίας η επαυξημένη ταχύτητα (λόγω της χρήσης αριστερού και δεξιού χεριού εναλλάξ) έμπλεκε πολλές φορές τα σιδεράκια που είχαν τους χαρακτήρες και πήγαιναν με δύναμη προς την μελανοταινία και το χαρτί για να τυπώσουν. Παρά την ανυπαρξία τέτοιων προβλημάτων στους υπολογιστές (και ακόμη και στις σύγχρονες γραφομηχανές), και παρά την ύπαρξη αρκετών άλλων, πιο εργονομικών και αποδοτικών πληκτρολογίων, τα πληκτρολόγια QWERTY παραμένουν τα πλέον διαδεδομένα με συντριπτική διαφορά.
- Αλφαβητικά: Είναι πληκτρολόγια όπου τα γράμματα είναι διατεταγμένα αλφαβητικά. Χρησιμοποιούνται κυρίως σε υπολογιστές τσέπης, όπου γνωρίζουν αρκετή επιτυχία για άγνωστους προς το παρόν λόγους!
- DVORAK: Είναι πληκτρολόγια με διαφορετική τοποθέτηση των πλήκτρων για αύξηση ταχύτητας. Συγκεκριμένα, είναι σχεδιασμένα ώστε το 70% των αγγλικών λέξεων να μπορούν να πληκτρολογηθούν από την βασική γραμμή και μόνο, όπου βρίσκονται μονίμως τα δάχτυλα του χρήστη, χωρίς να υπάρχει ανάγκη κίνησης προς τις άλλες γραμμές. Έχει παρατηρηθεί περίπου 10-15% αύξηση ταχύτητας.
- Χορδής (Chord): Είναι πληκτρολόγια με πολύ διαφορετικό σχεδιασμό από τα προηγούμενα. Απαιτούν ταυτόχρονη πίεση πολλών πλήκτρων μαζί για τον καθορισμό ενός χαρακτήρα, όπου πολλές φορές τα συγκεκριμένα πλήκτρα σχηματίζουν λόγω της τοποθέτησής τους στο πληκτρολόγιο τον χαρακτήρα στον οποίον αντιστοιχούν. Τα πληκτρολόγια αυτά είναι χρήσιμα όπου απαιτείται μονόχειρη χρήση πληκτρολογίου ή υπάρχει περιορισμένος χώρος.

### Άλλες Μονάδες Εισαγωγής Κειμένου

Εκτός από τα πληκτρολόγια, υπάρχουν και μερικοί άλλοι τρόποι για εισαγωγή κειμένου στον υπολογιστή.

- Αναγνώριση φωνής, όπου παρ' όλη την πρόοδο που έχει επιτευχθεί, τα υπάρχοντα συστήματα έχουν περίπου 97% επιτυχία στην αναγνώριση φωνημάτων. Δεδομένου ότι μία λέξη αποτελείται κατά μέσο όρο από πέντε φωνήματα, συνεπάγεται ότι υπάρχει σφάλμα σε μία για κάθε έξι λέξεις. Αυτό συνήθως δεν θεωρείται αρκετό, οπότε η αναγνώριση φωνής χρησιμοποιείται μόνο σε περιορισμένα περιβάλλοντα.
- Αναγνώριση χειρόγραφου, όπου η απλή σάρωση και αναγνώριση των χαρακτήρων δεν είναι πολλές φορές αρκετή αλλά απαιτείται η αναγνώριση και άλλων χαρακτηριστικών, π.χ., η αναγνώριση υπογραφών απαιτεί αναγνώριση και της συγκεκριμένης κίνησης του υπογράφοντα, την πίεση που εξασκεί στο χαρτί σε κάθε σημείο, κτλ.
- Αναγνώριση κειμένου, όπου εδώ απλά απαιτείται η απλή σάρωση και αναγνώριση των τυπωμένων χαρακτήρων.

### Δείκτες (Pointing Devices)

Για μετάδοση πληροφορίας προς τον υπολογιστή άλλης μορφής εκτός από κείμενο υπάρχουν διάφορες άλλες μονάδες.

- Ποντίκια (ενός, δύο, ή τριών πλήκτρων)
- Ευαίσθητες οθόνες (touch screen, οπτικές οθόνες - για περιορισμένη χρήση)
- Trackball
- Joystick
- Στυλό φωτός
- Διάφοροι 3-διάστατοι δείκτες (γάντια δεδομένων, 3-διάστατα ποντίκια, κράνη, ρούχα και δωμάτια για εικονική πραγματικότητα, κτλ.)
- Απλή ματιά ή κίνηση κάποιου μέλους του χρήστη (για χρήση σε μαχητικά συνήθως αεροπλάνα, ή σε περιβάλλοντα με ειδικές ανάγκες, π.χ., όπως στο μηχάνημα Equalizer του Steve Hawking)

## 3.2 Μονάδες Εξόδου

Είναι κυρίως διαφόρων μορφών οθόνες (CRT, LCD, και 3-διάστατες) καθώς και μερικών άλλων τύπων μονάδες.

### CRT (Cathode Ray Tube)

Η τεχνολογία βασίζεται σε σάρωση της οθόνης γραμμή-γραμμή περίπου στα 30Hz. Η σάρωση γίνεται είτε πάντα από αριστερά προς τα δεξιά, είτε εναλλάξ το προηγούμενο με από δεξιά προς τα αριστερά, είτε δαντελωτά (interlaced), δηλαδή, πρώτα οι μονές γραμμές και μετά οι ζυγές.

Ένα σημαντικό πρόβλημα που έχουν αυτές οι οθόνες είναι ότι οι διακριτές μεταπτώσεις που απαιτούνται στην ένταση των διαφόρων εικονοστοιχείων (pixels) για την απόδοση πλάγιων ακμών είναι ιδιαίτερα εμφανείς και ενοχλητικές. Γι' αυτό και χρησιμοποιείται η τεχνική του *antialiasing*, η οποία μαλακώνει/αμβλύνει το πρόβλημα. Στις επόμενες εικόνες εμφανίζεται ο χαρακτήρας A σχεδιασμένος με ή χωρίς χρήση της τεχνικής *anti-aliasing*, σε κανονικό μέγεθος και μεγέθυνση.



Τυπικά προβλήματα υγείας που δημιουργούνται από αυτές τις οθόνες είναι οι ακτίνες X (που ακτινοβολούνται κυρίως από το πίσω μέρος της οθόνης), οι υπέρυθρες/ υπεριώδεις ακτίνες (που είναι όμως αμελητέας έντασης), οι ακουστικές και υπερηχητικές συχνότητες, το ηλεκτροστατικό πεδίο (που δημιουργεί δερματικούς ερεθισμούς), και το γενικότερο ηλεκτρομαγνητικό πεδίο (που μπορεί να δημιουργήσει καταρράκτη, καθώς και αποβολές σε εγκύους).

Για την αποφυγή όλων αυτών, υπάρχουν μερικοί κανόνες χρήσεις αυτών των οθονών που καλό είναι όλοι μας να ακολουθούμε: να καθόμαστε πάντα μπροστά από την οθόνη και τότε κοντά στο πίσω μέρος κάποιας άλλης, να έχουμε κάποια απόσταση από την οθόνη, να χρησιμοποιούμε μεγάλο μεγέθους γραμματοσειρές, να κάνουμε διαλείμματα στην χρήση του υπολογιστή, να είμαστε σε καλοφωτισμένο χώρο, και να μην είμαστε μπροστά σε φωτεινό παράθυρο.

### LCD (Liquid Crystal Display)

Χρησιμοποιούνται κυρίως στους φορητούς υπολογιστές και στις ηλεκτρονικές αριθμομηχανές (κομπιουτεράκια), αλλά πρόσφατα έχουν κυκλοφορήσει και για κανονικούς υπολογιστές. Οι οθόνες αυτές είναι μικρές και ελαφριές, καταναλώνουν λιγότερη ενέργεια, και δεν παρουσιάζουν κανένα από τα προβλήματα που αναφέραμε παραπάνω για τις οθόνες CRT. Οι οθόνες τύπου LCD χρησιμοποιούν δύο κρύσταλλα που περιέχουν υγρό ανάμεσά τους. Οι οθόνες αυτού του τύπου χρειάζονται επιπλέον φωτισμό. Τελευταία έχουν αρχίσει να χρησιμοποιούνται οθόνες τεχνολογίας TFT στις οποίες κάθε pixel αντιστοιχεί σε ένα transistor. Οι οθόνες αυτές δεν έχουν τα προβλήματα φωτεινότητας έχουν όμως σχετικά μεγάλο κόκκο οθόνης και είναι ακόμη ακριβή τεχνολογία.

### 3-Διάστατες Οθόνες

Αυτές χρησιμοποιούνται κυρίως σε εφαρμογές εικονικής πραγματικότητας. Περιλαμβάνουν συνήθως κάποιο είδος γυαλιών (όπως γυαλιά πόλωσης ή διαφράγματος) που δίνουν στο χρήστη στερεοσκοπική αίσθηση.

### Άλλου Τύπου Μονάδες Εξόδου

Εκτός από οθόνες υπάρχουν και άλλου τύπου μονάδες εξόδου, όπως οι απαραίτητοι εκτυπωτές, ή για πιο εξειδικευμένες ανάγκες, οι μονάδες σύνθεσης ομιλίας (π.χ., στο τηλέφωνο, όπου κάθε φώνημα είναι αντίστοιχο με μία συλλαβή) ή οι ακουστικές οθόνες για τυφλούς χρήστες.

## 3.3 Μνήμη Υπολογιστών

Όπως είναι γνωστό, οι υπολογιστές διαθέτουν μία ιεραρχία μνήμης:

- καταχωρητές (registers)
- κρυφή/γρήγορη μνήμη (cache)
- κύρια μνήμη
- δευτερεύουσα μνήμη (δίσκοι)
- τριτεύουσα μνήμη (ταινίες, CD, κτλ.)

Στην ιεραρχία αυτή, από πάνω προς τα κάτω το κόστος και η ταχύτητα μειώνεται ενώ η χωρητικότητα αυξάνει. Η επίδραση των διαφόρων τύπων μνήμης στις διεπαφές είναι πολύ σημαντική και πρέπει να λαμβάνεται σοβαρά υπ' όψη κατά των σχεδιασμό τους. Κλασικό παράδειγμα αποτελεί αυτό που αναφέραμε προηγουμένως σχετικά με την περίπτωση καθυστέρησης αντίδρασης του υπολογιστή σε ενέργεια του χρήστη. Για παράδειγμα, σε προσπάθεια να κινηθεί ο κέρσορας προς τα αριστερά κάτω από κάποιο επιθυμούμενο σημείο, αν το σύστημα δεν αντιδράσει, ο χρήστης μπορεί να επανέλθει με επαναλαμβανόμενες πληκτρολογήσεις, οι οποίες αν αποθηκευθούν στην μνήμη και μετά ξαφνικά εκτελεσθούν όλες μαζί δημιουργείται πρόβλημα. Γι' αυτό και οι διεπαφές θα πρέπει να προσφέρουν στο χρήστη μέσω ανάδρασης (feedback) την αίσθηση ότι έχουν αντιληφθεί τις ενέργειές του. Καλό παράδειγμα αποτελεί το τηλέφωνο, όπου ο χρήστης ξέρει πάντα πού βρίσκεται.

---

Ερώτηση: Μπορείτε να αναφέρετε κάποιες συσκευές εισόδου - εξόδου εκτός του πληκτρολογίου; Γνωρίζετε πού χρησιμοποιούνται και πώς λειτουργούν;

---

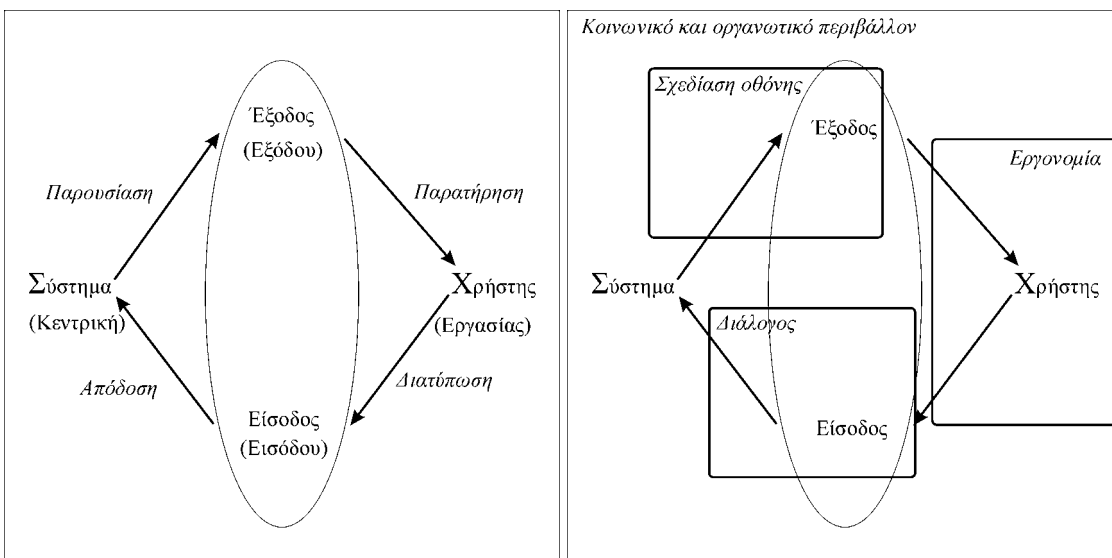
## Κεφάλαιο 4 Η επικοινωνία

### 4.1 Μοντελοποίηση

Υπάρχουν πολλοί τρόποι μοντελοποίησης της επικοινωνίας μεταξύ ανθρώπου και υπολογιστή, οι οποίοι είναι χρήσιμοι για τον κατάλληλο σχεδιασμό των διεπαφών. Ένα τέτοιο γενικό μοντέλο που περιγράφει τα διάφορα στάδια ενός κύκλου αυτής της επικοινωνίας από πλευράς ανθρώπου είναι και το *Μοντέλο Διάδρασης του Norman*, το οποίο καθορίζει τα εξής στάδια:

- Δράση
  - καθορισμός στόχου/σκοπού
  - διαμόρφωση απαιτούμενης ενέργειας
  - καθορισμός ακολουθίας πράξεων
  - εκτέλεση πράξεων
- Αντίδραση
  - αντίληψη κατάστασης συστήματος
  - ερμηνεία κατάστασης συστήματος
  - αξιολόγηση κατάστασης συστήματος σε σχέση με στόχο και ενέργεια

Ένα άλλο μοντέλο αυτού του κύκλου, το οποίο αποτυπώνει και τις διάφορες γλώσσες που χρησιμοποιούνται σε διάφορα σημεία επικοινωνίας καθώς και τις επιμέρους επιστημονικές ενότητες που ασχολούνται με διάφορα τμήματα του όλου κύκλου είναι ο λεγόμενος *Σκελετός Διάδρασης*, ο οποίος δίνεται σχηματικά στις δύο παρακάτω εικόνες. Στην πρώτη αναφέρονται τα τέσσερα βασικά στοιχεία της επικοινωνίας σαν κύριοι κόμβοι, οι συγκεκριμένες πράξεις που διαδραματίζονται μεταξύ τους σαν ακμές, και οι γλώσσες που χρησιμοποιούνται σε κάθε ένα από αυτά μέσα σε παρένθεση. Στην δεύτερη παρουσιάζεται πάλι η βασική εικόνα των κόμβων και των ακμών, και αποτυπώνονται τα στοιχεία (κόμβοι) που εμπλέκονται στους τρεις βασικούς γνωσιακούς χώρους της επικοινωνίας, δηλαδή την εργονομία, την σχεδίαση διαλόγου, και την σχεδίαση οθόνης:



Με βάση τον Σκελετό Διάδρασης, αναλύουμε παρακάτω σε ξεχωριστές υποενότητες, τους τρεις γνωσιακούς χώρους που φαίνονται στην εικόνα, δηλαδή την εργονομία, την σχεδίαση διαλόγου, και την σχεδίαση οθόνης.

## 4.2 Εργονομία

Ένα βασικό θέμα με το οποίο ασχολείται η εργονομία είναι η σωστή τοποθέτηση οργάνων ελέγχου και αναπαραστάσεων δεδομένων, η οποία πρέπει να έχει λογική που να επιτρέπει άμεση προσπέλαση και αντίληψη. Πιθανές οργανώσεις είναι οι εξής:

- *λειτουργική*: παρόμοιες λειτουργίες είναι συγκεντρωμένες στον ίδιο χώρο, εκτός από ακριβώς αντίθετες που πρέπει να έχουν απόσταση, π.χ., το παράδειγμα του 1ου κεφαλαίου με τις λειτουργίες ΑΠΟΘΗΚΕΥΣΗ/ΔΙΑΓΡΑΦΗ.
- *σειριακή*: οι λειτουργίες οργανώνονται ακολουθιακά με βάση τη συνήθη σειρά εκτέλεσης (η οργάνωση αυτή δεν είναι και πολύ συνηθισμένη).
- *συχνοτική*: οι λειτουργίες οργανώνονται ακολουθιακά με βάση τη συχνότητα χρήσης.
- *αλφαβητική*: οι λειτουργίες οργανώνονται αλφαβητικά (επίσης ασυνήθιστη οργάνωση).

Οι οργάνωση οργάνων και αναπαραστάσεων δεδομένων πρέπει να δημιουργούν ευκολίες για τον χρήστη. Συγκεκριμένα, τα πιο σημαντικά δεδομένα πρέπει να είναι στο ύψος του ματιού του και ο φωτισμός να είναι τέτοιος που να αποφεύγεται η αντανάκλαση φωτός.

Ένα άλλο θέμα της εργονομίας είναι ο καθορισμός του φυσικού περιβάλλοντος εργασίας. Απαιτείται να υπάρχει πάντα ο απαραίτητος χώρος για την ομαλή λειτουργία του συστήματος, δίνοντας πάντα μεγάλη σημασία στο μέγεθος του συγκεκριμένου χρήστη, όπου αυτός είναι γνωστός, είδ' αλλιώς του μέσου χρήστη.

Η εργονομία ασχολείται επίσης με διάφορα θέματα υγείας που μπορεί να προκύψουν από την χρήση υπολογιστών. Κύρια σημεία προσοχής είναι η στάση του σώματος του χρήστη, η θερμοκρασία και ο φωτισμός του χώρου, ο θόρυβος του γενικότερου περιβάλλοντος, καθώς και ο χρόνος συνεχούς εργασίας, ο οποίος πρέπει να είναι περιορισμένος.

Τέλος, ένα άλλο σημαντικό στοιχείο μελέτης είναι το χρώμα. Αρκετά συμπεράσματα της εργονομίας για την χρήση του χρώματος τα έχουμε ήδη αναφέρει, π.χ., πρέπει να χρησιμοποιούνται λίγα χρώματα, το μπλε δεν πρέπει να χρησιμοποιείται για κρίσιμες πληροφορίες, λόγω των φαναριών κυκλοφορίας τα κόκκινο-κίτρινο-πράσινο πρέπει να αντιστοιχούν σε κατάσταση κρίσης-ετοιμότητας-κανονικότητας, κτλ.

## 4.3 Σχεδίαση Διαλόγου

Υπάρχουν πολλές μορφές διάδρασης μέσω των οποίων υλοποιείται ο διάλογος ανθρώπου μηχανής:

- Γραμμή εντολών
- Επιλογείς (Μενού εντολών)
- Φυσική γλώσσα
- Ερωταπαντήσεις και επερωτήσεις
- Φόρμες και φύλλα υπολογισμού
- 3-διάστατοι διάλογοι
- Απ' ευθείας διαχείριση
- ΠΕΠΕ (Παράθυρα-Εικονίδια-Ποντίκια-Επιλογείς)

Κάθε μια από τις παραπάνω μορφές διάδρασης εντοπίζεται σε έναν ή περισσότερους τύπους διεπαφής, και αντίστοιχα κάθε ένας από τους διαφορετικούς τύπους διεπαφής μπορεί να εκμεταλλεύεται μία ή και περισσότερες από τις μορφές διάδρασης. Αναλυτικότερη περιγραφή κάθε τέτοιας μορφής υπάρχει στο ΜΕΡΟΣ ΙΙΙΚεφάλαιο 11

#### 4.4 Σχεδίαση Οθόνης

Η σχεδίαση οθόνης έχει σημασία και για την εισαγωγή πληροφοριών στον υπολογιστή και για την παρουσίαση πληροφοριών από τον υπολογιστή προς τον χρήστη. Και στις δύο περιπτώσεις εμφανίζονται τα ίδια θέματα τα οποία πρέπει να έχει υπ' όψη του ο σχεδιαστής. Κατ' αρχάς, ο τρόπος εισαγωγής ή παρουσίασης της πληροφορίας επηρεάζει άμεσα την αντίληψη και ετοιμότητα του χρήστη. Για παράδειγμα, μία λίστα αρχείων διατεταγμένη ως προς το όνομα του αρχείου ή την ημερομηνία φέρνει αμεσότητα στην αντίληψη διαφορετικών χαρακτηριστικών των αρχείων της λίστας που μπορεί να είναι σημαντική σε διαφορετικές εφαρμογές. Εν γένει, έχει μεγάλη σημασία να υπάρχει αντιστοιχία του σκοπού του χρήστη με τον τρόπο που είναι σχεδιασμένη η οθόνη. Συνήθως πολλές επιλογές είναι επιθυμητές, όπως και γίνεται στα περισσότερα συστήματα. Υπάρχουν βέβαια και κάποιες γενικές αρχές που πρέπει πάντα να ακολουθούνται. Αρκετές τις έχουμε αναφέρει ήδη σε προηγούμενες συζητήσεις, π.χ., ότι απαιτείται προσοχή στη χρήση χρώματος, το οποίο δεν είναι συνήθως χρήσιμο για την μετάδοση πληροφορίας αλλά κυρίως για να δίνει έμφαση ή να διαφοροποιεί περιοχές της οθόνης. Άλλες τέτοιες αρχές είναι εύκολα κατανοητές από την προσωπική μας εμπειρία, π.χ., ότι απαιτείται προσοχή στην στοίχιση κάθετων λιστών – λίστες ονομάτων να έχουν στοίχιση στο πρώτο γράμμα του επιθέτου του ονόματος, λίστες δεκαδικών αριθμών να έχουν στοίχιση στο κόμμα που χωρίζει το ακέραιο από το δεκαδικό τμήμα, κτλ.

Πέρα από την χρηστικότητα μεγάλη σημασία στην σχεδίαση της οθόνης παίζει βέβαια και η αισθητική. Δυστυχώς, αυτά τα δύο δεν συμπλέουν πάντοτε, και πολλές φορές συγκρούονται οι ανάγκες της μιας με τις επιθυμίες της άλλης. Για παράδειγμα, η αισθητική υπαγορεύει όλα τα πλήκτρα ελέγχου ενός πολύπλοκου συστήματος να έχουν την ίδια εμφάνιση. Από την άλλη, η χρηστικότητα πιθανώς να απαιτεί ασυνέπεια, δηλαδή να υπάρχουν τέτοια πλήκτρα αρκετών διαφορετικών εμφανίσεων ώστε ο χρήστης να διαχωρίζει άμεσα και εύκολα σε ποιο συγκεκριμένο υποσύστημα ανήκει το καθένα. Υπάρχουν όμως και φορές που η αισθητική μπορεί να βοηθήσει την χρηστικότητα. Χαρακτηριστικό παράδειγμα είναι η μελέτη του χώρου ανάμεσα στα πρωτεύοντα στοιχεία που παρουσιάζονται στην οθόνη, δηλαδή το "αρνητικό" της εικόνας κατά κάποιον τρόπο, όπως η λέξη "Ο ΜΕΤΡΗΤΗΣ" που εμφανίζεται σαν το "αρνητικό" της παρακάτω εικόνας [Dix, σελ 134]:



Η αισθητική μελέτη του "αρνητικού" μπορεί να δώσει αρκετά καλές πληροφορίες για την χρηστικότητα της οθόνης. Για παράδειγμα, αν στοιχεία του αρνητικού που υποτίθεται σχετίζονται μεταξύ τους είναι μακριά το ένα από το άλλο, τότε η σχεδίαση της οθόνης χρειάζεται ξανά μελέτη.

#### 4.5 Σχεδίαση λογισμικού για διεθνείς χρήστες

##### Διεθνοποίηση

Η διεθνοποίηση είναι η διαδικασία ανάπτυξης εφαρμογών λογισμικού που επιτρέπει την εύκολη μετατροπή τους ώστε να λειτουργούν σε διαφορετικά πολιτισμικά περιβάλλοντα [5]. Ο όρος πολιτισμικό περιβάλλον εκτός από τη γλώσσα περιλαμβάνει και θέματα όπως ήθη και έθιμα αλλά και νομοθεσία και άλλους κανονισμούς που μπορεί να επηρεάζουν τον τρόπο λειτουργίας της εφαρμογής. Σύμφωνα με τη μεθοδολογία της διεθνοποίησης όποια πολιτισμικά χαρακτηριστικά

διαθέτει το λογισμικό εκφράζονται παραμετρικά ή απομονώνονται ώστε να διευκολύνεται η μετατροπή του.

Η διαδικασία της διεθνοποίησης απαιτεί μεθοδική προετοιμασία. Καταρχήν θα πρέπει να συλλεχθούν όλες οι ιδιαιτερότητες που αφορούν τα πολιτισμικά περιβάλλοντα στα οποία απευθύνεται ή είναι πιθανό να απευθυνθεί το λογισμικό προϊόν και είναι δυνατό να επηρεάσουν τον τρόπο λειτουργίας του. Για παράδειγμα η ανάπτυξη ενός λογισμικού προϊόντος που περιλαμβάνει μια διαδικασία ταξινόμησης και που απευθύνεται μόνο στην Αμερικάνικη αγορά και υποστηρίζει μόνο αμερικάνικα Αγγλικά (και όχι Ισπανικά ή Γαλλικά που είναι επίσης αρκετά διαδομένα σε μερικές Πολιτείες) είναι απλή. Όταν όμως το ίδιο προϊόν απευθύνεται και σε χρήστες Ευρωπαϊκών χωρών τότε η διαδικασία ανάπτυξης καθίσταται περισσότερο περίπλοκη, επειδή η διαδικασία ταξινόμησης θα πρέπει να μπορεί να χειριστεί ορθά τους χαρακτήρες των άλλων γλωσσών. Ακόμη και για τη Βρετανία θα χρειαστούν αλλαγές διότι η αμερικανική διαδικασία ταξινόμησης θέτει πρώτα τους κεφαλαίους χαρακτήρες και μετά τους πεζούς. Σε ένα τέτοιο ενδεχόμενο η διαδικασία ταξινόμησης θα πρέπει να υλοποιεί όλες τις πιθανές περιπτώσεις, παρότι κάθε μια από αυτές χρησιμοποιείται μόνο στο αντίστοιχο περιβάλλον. Δηλαδή η δυνατότητα της ταξινόμησης Γαλλικών λέξεων υπάρχει έστω και εάν απαιτείται μόνο όταν είναι ενεργό το Γαλλικό περιβάλλον.

Αντίστοιχα οι γλωσσικοί πόροι του λογισμικού προϊόντος απομονώνονται από το κώδικα του προγράμματος ώστε να είναι εύκολος ο εντοπισμός τους και η μετάφρασή τους σε κάθε μια από τις γλώσσες που υποστηρίζονται. Πρακτικά η μετάφραση και η ενσωμάτωση των γλωσσικών πόρων στα Γαλλικά δε θα χρειαστεί παρά μόνο όταν το προϊόν θα πρέπει να προσαρμοστεί στις απαιτήσεις του συγκεκριμένου πολιτισμικού περιβάλλοντος.

Με την παραμετροποίηση ή την απομόνωση των πολιτισμικών χαρακτηριστικών του λογισμικού διευκολύνεται η διαδικασία της τοπικής προσαρμογής. Παρότι αυξάνεται το αρχικό κόστος ανάπτυξης, η αύξηση αυτή είναι πολύ μικρή σε σχέση με το κόστος επανασχεδιασμού και υλοποίησης του ίδιου προϊόντος για κάθε ένα πολιτισμικό περιβάλλον χωριστά όπως είναι αναγκαίο εάν δεν έχει προηγηθεί η διαδικασία διεθνοποίησης. Όταν έχει ολοκληρωθεί η διαδικασία διεθνοποίησης το προϊόν μπορεί εύκολα να προσαρμοστεί στις απαιτήσεις των τοπικών πολιτισμικών περιβαλλόντων.

### **Τοπική Προσαρμογή**

Η διαδικασία της μετατροπής του λογισμικού ώστε να ικανοποιεί τις απαιτήσεις ενός πολιτισμικού περιβάλλοντος ονομάζεται τοπική προσαρμογή. Εάν κατά τη διάρκεια της ανάπτυξης του προϊόντος έχει προηγηθεί η διαδικασία της διεθνοποίησης, τότε ο χρόνος προσαρμογής στις τοπικές πολιτισμικές ιδιαιτερότητες μπορεί να ελαχιστοποιηθεί.

Η διαδικασία της τοπικής προσαρμογής αποτελείται από δυο κύριες φάσεις:

- **Μετάφραση των γλωσσικών πόρων στην τοπική γλώσσα**

Κατά τη διάρκεια αυτής της φάσης όλοι οι γλωσσικοί πόροι μεταφράζονται στην τοπική γλώσσα. Αυτό συνήθως συμπεριλαμβάνει εκτός από τα μενού και τα μηνύματα του λογισμικού, τα κείμενα βοήθειας και την τεκμηρίωση (ηλεκτρονική ή έντυπη).



- Προσαρμογή του λογισμικού προϊόντος σε άλλες τοπικές απαιτήσεις

Κατά τη διάρκεια αυτής της φάσης το λογισμικό προϊόν τροποποιείται ώστε να ανταποκρίνεται σε τοπικές συνήθειες. Εάν έχει προηγηθεί η διαδικασία της διεθνοποίησης τότε είναι πολύ πιθανό να μη χρειαστούν επιπλέον τροποποιήσεις, εφόσον οποιαδήποτε ανάγκη για αλλαγές όπως προσθήκη ειδικών αλγόριθμων ταξινόμησης ή μεθόδων μορφοποίησης ημερομηνίας, ώρας κλπ., μπορεί να έχει ήδη ικανοποιηθεί ή το λογισμικό μπορεί να συμμορφώνεται στις ρυθμίσεις του λειτουργικού συστήματος.

Συνολικά η χρήση της μεθόδου της διεθνοποίησης κατά τη διάρκεια σχεδιασμού του λογισμικού προϊόντος έχει θετικές επιπτώσεις στην ανάπτυξη προϊόντων που απευθύνονται σε διεθνές κοινό. Ειδικά όταν το προϊόν θα προσαρμοστεί σε περισσότερα από ένα τοπικά περιβάλλοντα η διεθνοποίηση μπορεί να μειώσει δραματικά το κόστος ανάπτυξης τοπικά προσαρμοσμένων εκδόσεων για κάθε πολιτισμική κοινότητα, ενώ ταυτόχρονα διευκολύνει και απλοποιεί τη διαδικασία υποστήριξης και συντήρησης των διαφορετικών εκδόσεων. Αποτέλεσμα της διαδικασίας της τοπικής προσαρμογής είναι η παραγωγή διαφορετικών, τοπικά προσαρμοσμένων εκδόσεων του λογισμικού προϊόντος, για κάθε ομάδα χρηστών.



## Κεφάλαιο 5 Χρησιμότητα

Έχοντας πλέον υπ' όψη μας τα κύρια χαρακτηριστικά του ανθρώπου και του υπολογιστή καθώς και τις βασικές μορφές επικοινωνίας μεταξύ των δύο, προχωρούμε στην περιγραφή της διαδικασίας ανάπτυξης συστημάτων EAM υιοθετώντας και χρησιμοποιώντας όλη αυτήν την προηγούμενη γνώση. Το κεφάλαιο αυτό διαιρείται σε τέσσερα τμήματα, όπου το πρώτο αναφέρει τις βασικές αρχές χρησιμότητας πάνω στις οποίες πρέπει να βασίζεται η ανάπτυξη τέτοιων συστημάτων, το δεύτερο περιγράφει τον κύκλο ζωής ενός συστήματος λογισμικού (και πιο συγκεκριμένα ενός συστήματος EAM) με όλα τα στάδιά του, το τρίτο περιγράφει τις διάφορες προσεγγίσεις που υπάρχουν για την χρήση πρωτοτύπων κατά την διάρκεια ανάπτυξης συστημάτων, και το τέταρτο συζητά γενικές τεχνικές σχεδίασης συστημάτων από ομάδες ανθρώπων (αλλά και από άτομα). Λεπτομερής ανάλυση των μεθόδων και τεχνικών που εφαρμόζονται σε κάθε ένα από τα στάδια του κύκλου ζωής ενός λογισμικού EAM γίνεται ξεχωριστά για κάθε στάδιο στα επόμενα κεφάλαια.

### 5.1 Βασικές Αρχές Χρησιμότητας

Τα διαλογικά συστήματα εν γένει αυξάνουν την χρησιμότητα των υπολογιστικών συστημάτων και καθιστούν την επικοινωνία ανθρώπου-υπολογιστή πιο αποδοτική και πιο ευχάριστη. Στόχος μας είναι η ανάπτυξη συστημάτων EAM με τη μέγιστη χρησιμότητα. Αναφορικά με αυτό, δημιουργούνται δύο ερωτήματα: πώς εξασφαλίζεται και πώς μετριέται η χρησιμότητα; Υπάρχουν δύο τρόποι προσέγγισης στις απαντήσεις αυτών των ερωτήσεων, η πρακτική και η θεωρητική. Και οι δύο πηγάζουν από την μακροχρόνια εμπειρία εφαρμογών. Η μεν πρακτική υποδεικνύει παραδείγματα επιτυχημένων συστημάτων των οποίων η σχεδίαση και υλοποίηση μπορούν να εξυπηρετήσουν σαν βάσεις για την ανάπτυξη μελλοντικών προϊόντων. Η δε θεωρητική καθορίζει μια σειρά από βασικές αρχές σχεδιασμού και μια συλλογή από θεωρητικά μοντέλα που περιγράφουν την δυναμική της επικοινωνίας χρήστη και υπολογιστή. Και οι δύο προσεγγίσεις έχουν πλεονεκτήματα και μειονεκτήματα, οπότε το μέλλον των διαλογικών συστημάτων βασίζεται στην υιοθέτηση και των δύο προσεγγίσεων και την αλληλοσυμπλήρωσή τους.

Σχετικά με την προσέγγιση που βασίζεται σε παλιότερα παραδείγματα, πρέπει να τονίσουμε ότι η τεχνολογία των υπολογιστών είναι σε συνεχή εξέλιξη με συνεχή τάση για πτώση του κόστους τους, αύξηση των δυνατοτήτων τους, και αύξηση του αριθμού των εφαρμογών και της κοινότητας των χρηστών τους. Οι τάσεις αυτές επιτρέπουν από την μια και απαιτούν από την άλλη την βελτίωση της επικοινωνίας με τον χρήστη. Αυτή η βελτίωση γίνεται φανερή αν αναλογιστεί κανείς κάποιους βασικούς σταθμούς στην ιστορική πορεία των υπολογιστών, όπως παρακάτω:

- 1950 - Συστήματα χωρίς διάδραση (batch systems), διατρητικές κάρτες
- 1960 - Συστήματα χρονομερισμού (time-sharing systems), οθόνες
- 1970 - Εργαλειοθήκες προγραμματισμού
- 1980 - Προσωπικοί σταθμοί εργασίας, γραφικά
- 1990 - Προσωπικοί υπολογιστές μεγάλης ισχύος
- 2000 - Διαδίκτυο, παγκόσμιο πλέγμα πληροφοριών

Όλες αυτές οι εξελίξεις έχουν δώσει τη δυνατότητα για ανάλογη πρόοδο στις τεχνικές και τις μεθόδους της επικοινωνίας ανθρώπου-μηχανής, με κατάληξη το σύνολο αυτών που έχουμε αναφέρει στα προηγούμενα κεφάλαια και άλλων (π.χ., παραθυρικά περιβάλλοντα, μεταφορές, απευθείας χειρισμός, υπερκείμενο, συνεργασίες με την βοήθεια υπολογιστή, και άλλα) τα οποία έχουν οριοθετήσει τον χώρο της EAM.

Όσον αφορά την θεωρητική προσέγγιση, αυτή προκρίνει μια πλειάδα από βασικές αρχές χρηστικότητας όπως αυτές προκύπτουν από την έρευνα στους χώρους της ψυχολογίας, της κοινωνιολογίας, και της πληροφορικής. Αν και δεν είναι ποτέ δυνατόν να έχουμε μια πλήρη λίστα από αυτές τις αρχές, εν τούτοις η έρευνα έχει προχωρήσει αρκετά ώστε να υπάρχει ήδη ένα σύνολο αρχών αρκετά εμπεριστατωμένο, το οποίο και παρουσιάζεται αμέσως παρακάτω. Τις αρχές αυτές τις διαιρούμε αρχικά σε τρεις κατηγορίες, που αφορούν την δυνατότητα εκμάθησης (πόσο εύκολο είναι για έναν άπειρο χρήστη να χρησιμοποιήσει το σύστημα), την ευελιξία (πόσο ποικίλοι είναι οι τρόποι ανταλλαγής πληροφοριών μεταξύ συστήματος και χρήστη), και την ευρωστία (πόση υποστήριξη δίνεται στον χρήστη για να καθορίσει τους στόχους του και να αντιληφθεί την επίτευξή τους). Για κάθε κατηγορία παρουσιάζουμε έναν πίνακα με τις αρχές που εμπίπτουν στην κατηγορία και κάποια επεξηγηματικά σχόλια.

<b>Αρχές που επηρεάζουν την δυνατότητα εκμάθησης (learnability)</b>		
<b>Αρχή</b>	<b>Ορισμός</b>	<b>Σχετιζόμενες αρχές</b>
Προβλεψιμότητα	Η υποστήριξη του χρήστη στην πρόγνωση του αποτελέσματος μελλοντικών ενεργειών κρίνοντας από το αποτέλεσμα περασμένων.	Ορατότητα λειτουργιών
Δυνατότητα σύνθεσης	Η υποστήριξη του χρήστη στη διαπίστωση του αποτελέσματος προηγούμενων ενεργειών στην τρέχουσα κατάσταση.	Άμεση και αργοπορημένη ειλικρίνεια
Οικειότητα	Το επίπεδο στο οποίο η προηγούμενη γνώση του χρήστη από άλλες πραγματικές καταστάσεις ή υπολογιστικά συστήματα μπορεί να χρησιμοποιηθεί κατά τη διάδρασή του με ένα νέο σύστημα.	Ικανότητα να μαντεύεις σωστά άγνωστα σύμβολα και λειτουργίες
Δυνατότητα γενίκευσης	Η υποστήριξη του χρήστη στην επέκταση γνώσης από συγκεκριμένες περιπτώσεις διάδρασης στην ίδια και σε άλλες εφαρμογές σε ανάλογες καταστάσεις.	
Συνέπεια	Ομοιότητα στην συμπεριφορά κατά την εισαγωγή και εμφάνιση δεδομένων, που προκύπτει από παρόμοιες καταστάσεις ή στόχους.	

<b>Αρχές που επηρεάζουν την δυνατότητα εκμάθησης (learnability)</b>	
<b>Αρχή</b>	<b>Παράδειγμα</b>
Προβλεψιμότητα	
Δυνατότητα σύνθεσης	Η ράβδος διαδρομής που εμφανίζεται σε ορισμένους δικτυακούς τόπους. Η ανάδραση κατά την μεταφορά ενός αρχείου από έναν κατάλογο σε άλλον.
Οικειότητα	Η χρήση της μεταφοράς της γραφομηχανής για χρήστες (ειδικά για χρήστες του Δυτικού κόσμου).
Δυνατότητα γενίκευσης	Η χρήση του εντολών επεξεργασίας (αποκοπή, αντιγραφή, επικόλληση) σε εφαρμογές που εκτελούνται σε παραθυρικά περιβάλλοντα.

Συνέπεια	Η συνέπεια στην εμφάνιση αλλά και στην εκτέλεση λειτουργιών εφαρμογών σε παραθυρικά περιβάλλοντα.
----------	---

<b>Αρχές που επηρεάζουν την ευελιξία (flexibility)</b>		
<b>Αρχή</b>	<b>Ορισμός</b>	<b>Σχετιζόμενες αρχές</b>
Πρωτοβουλία διαλόγου	Η παραχώρηση στο χρήστη της ελευθερίας από τεχνητούς περιορισμούς που τίθενται από το σύστημα στους διαλόγους εισαγωγής δεδομένων.	Πρωτοβουλία συστήματος και χρήστη
Πολυ-νηματισμός	Η δυνατότητα του συστήματος να υποστηρίζει τη διάδραση του χρήστη σε περισσότερες από μια εργασίες την ίδια χρονική στιγμή.	Σύγχρονος ή εναλλασσόμενος πολυτροπισμός (πολλαπλή χρήση)
Δυνατότητα μετανάστευσης ενεργειών	Η δυνατότητα της μεταφοράς του ελέγχου της εκτέλεσης μιας συγκεκριμένης ενέργειας, ώστε να την αναλαμβάνει ο χρήστης, το σύστημα ή και οι δυο.	
Δυνατότητα αντικατάστασης	Η δυνατότητα του συστήματος να επιτρέπει ισοδύναμες αναπαραστάσεις δεδομένων εισόδου ή εξόδου να ανταλλάσσονται με άλλες.	Ποικιλία αναπαραστάσεων, ίσες ευκαιρίες
Δυνατότητα τροποποιήσεων	Η δυνατότητα τροποποίησης της διεπαφής από το χρήστη ή το σύστημα.	Προσαρμοστικότητα

<b>Αρχές που επηρεάζουν την ευελιξία (flexibility)</b>	
<b>Αρχή</b>	<b>Παράδειγμα</b>
Πρωτοβουλία διαλόγου	Ο χρήστης να είναι ελεύθερος να εκκινήσει οποιοδήποτε διάλογο (πχ. διάλογος ανεύρεσης και αντικατάστασης).
Πολυ-νηματισμός	Εμφάνιση μηνύματος λάθους με ταυτόχρονο ηχητικό σήμα.
Δυνατότητα μετανάστευσης ενεργειών	Η λειτουργία ορθογραφικού ελέγχου ή αυτή της μηχανικής μετάφρασης (απολύτως αυτόματη, ημι-αυτόματη ή εκτελούμενη μόνο από το χρήστη).
Δυνατότητα αντικατάστασης	Η εισαγωγή δεδομένων σε διαφορετικές μονάδες μέτρησης ή σε διαφορετική μορφή ή ακόμη και με κάποια έκφραση.
Δυνατότητα τροποποιήσεων	Η προσαρμογή της ράβδου εργαλείων στις απαιτήσεις του χρήστη.

<b>Αρχές που επηρεάζουν την ευρωστία (robustness)</b>		
<b>Αρχή</b>	<b>Ορισμός</b>	<b>Σχετιζόμενες αρχές</b>
Δυνατότητα παρατήρησης	Η δυνατότητα του χρήστη να αξιολογεί την εσωτερική κατάσταση του συστήματος από την εξωτερική του εικόνα.	Δυνατότητα φυλλομέτρησης, στατικές και δυναμικές αρχικές τιμές, συνοχή καταστάσεων, μονιμότητα, ορατότητα λειτουργιών
Δυνατότητα ανάκαμψης	Η δυνατότητα του χρήστη να πραγματοποιήσει διορθωτικές κινήσεις	Άμεση και έμμεση αναίρεση, αντιστοιχία δυσκολίας

	μόλις αναγνωρίσει κάποιο σφάλμα.	εσφαλμένης προσπάθειας και προσπάθειας αναίρεσης
Ανταπόκριση	Ο τρόπος με τον οποίο ο χρήστης αντιλαμβάνεται το ρυθμό επικοινωνίας με το σύστημα.	Σταθερότητα
Συμμόρφωση ενεργειών	Ο βαθμός στον οποίο οι υπηρεσίες του συστήματος υποστηρίζουν όλες τις ενέργειες που ο χρήστης επιθυμεί να εκτελέσει και με τον τρόπο που εκείνος τις κατανοεί.	Ικανές και αναγκαίες συνθήκες εργασίας

<b>Αρχές που επηρεάζουν την ευρωστία (robustness)</b>	
<b>Αρχή</b>	<b>Παράδειγμα</b>
Δυνατότητα παρατήρησης	Η δυνατότητα παρατήρησης της συνολικής δομής ενός εγγράφου ενώ ο χρήστης εισάγει κείμενο.
Δυνατότητα ανάκαμψης	Από πλευράς χρήστη η αναίρεση. Από πλευράς συστήματος η δυνατότητα αποκατάστασης λειτουργίας σε περίπτωση κατάρρευσης.
Ανταπόκριση	Ο χρόνος που χρειάζεται για την εμφάνιση ενός επιλογέα.
Συμμόρφωση ενεργειών	Η δυνατότητα που προσφέρει ένα πρόγραμμα επεξεργασίας κειμένου για την εισαγωγή μιας σειράς αριθμών σε ένα πίνακα.

Όλες οι παραπάνω αρχές πρέπει να λαμβάνονται υπ' όψη και όταν σχεδιάζεται το σύστημα αλλά και όταν αργότερα αξιολογείται.

Ερώτηση: Αναζητήστε πώς διατηρούνται οι παραπάνω αρχές στα Windows και σε άλλα λογισμικά προϊόντα που χρησιμοποιείτε. Υπάρχουν περιπτώσεις στις οποίες παραβιάζονται;

## **ΜΕΡΟΣ ΙΙ ΑΝΑΠΤΥΞΗ ΣΥΣΤΗΜΑΤΩΝ ΕΑΜ**

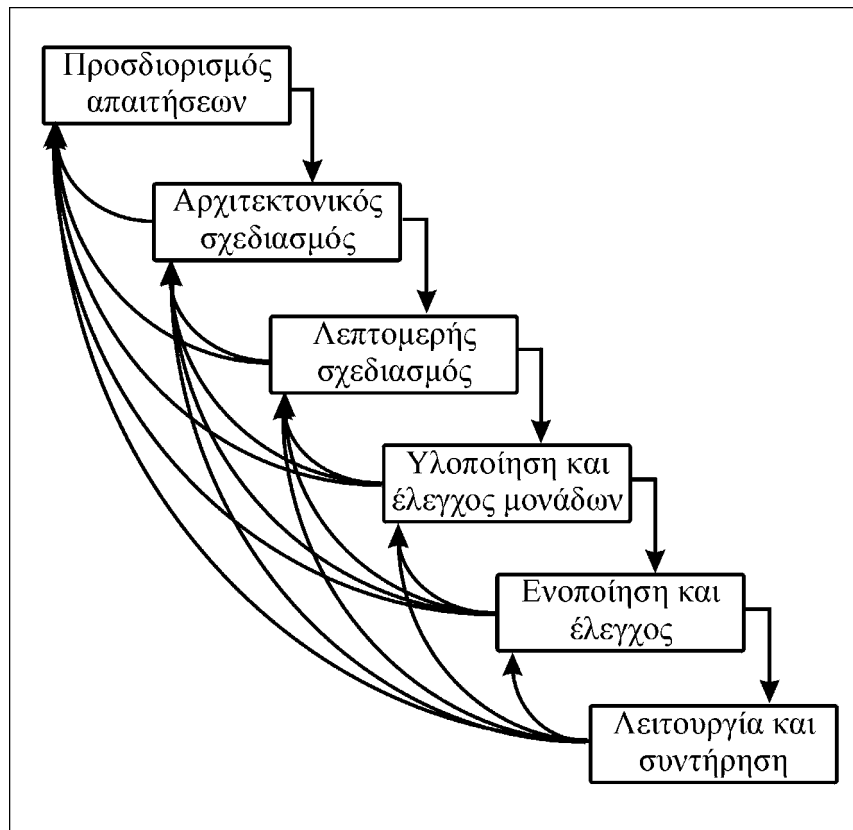
## Κεφάλαιο 6 Κύκλος Ζωής Λογισμικού EAM

Ο βασικός κύκλος ζωής ενός συστήματος λογισμικού επικοινωνίας ανθρώπου-μηχανής δεν διαφέρει σε τίποτε από τον αντίστοιχο κύκλο ενός οποιουδήποτε άλλου συστήματος λογισμικού. Η μόνη ιδιαιτερότητα είναι ότι η όλη διαδικασία γίνεται έχοντας σαν επίκεντρο τον χρήστη (user-centered design), πράγμα το οποίο δεν συμβαίνει συνήθως σε άλλα συστήματα. Ως εκ τούτου, η βασική συζήτηση σε αυτήν την ενότητα είναι παρόμοια με αντίστοιχες γύρω από γενικά θέματα τεχνολογίας λογισμικού, εκτός από κάποια σημεία όπου ο παράγων άνθρωπος απαιτεί κάποια διαφοροποίηση.

Το βασικό μοντέλο ανάπτυξης λογισμικού είναι το λεγόμενο *Μοντέλο Καταρράκτη* (*Cascade Model*), λόγω της μορφής του διαγράμματός του, όπως φαίνεται παρακάτω [Dix, σελ. 180, 183, και 188]. Στο διάγραμμα αυτό, διακρίνουμε έξι ξεχωριστά στάδια, τα οποία ορίζονται ως εξής:

1. *Καθορισμός προδιαγραφών*: Στο στάδιο αυτό καθορίζεται το τι αναμένεται να προσφέρει το σύστημα και όχι το πώς. Για συστήματα EAM, η προσέγγιση είναι κυρίως από την πλευρά του χρήστη. Οι προδιαγραφές μπορούν να καθοριστούν σε διάφορα επίπεδα λεπτομέρειας και χρησιμοποιώντας διάφορων τύπων συμβολισμούς, αλλά στην τελική τους έκδοση συγκεκριμενοποιούνται σε μία γλώσσα προδιαγραφών που είναι ακριβής ώστε να είναι επεξεργάσιμες από ένα υπολογιστή.
2. *Αρχιτεκτονική σχεδίαση*: Στο στάδιο αυτό καθορίζεται το πώς θα προσφέρει το σύστημα αυτά που υπαγορεύουν οι προδιαγραφές στο επίπεδο της βασικής δομής του συστήματος. Δηλαδή γίνεται ενός υψηλού επιπέδου καθορισμός των μονάδων του συστήματος και της λειτουργικότητας κάθε μίας καθώς και οι αλληλεπιδράσεις μεταξύ τους.
3. *Λεπτομερής σχεδίαση*: Στο στάδιο αυτό καθορίζεται και πάλι το πώς θα προσφέρει το σύστημα αυτά που υπαγορεύουν οι προδιαγραφές αλλά με σαφώς μεγαλύτερη λεπτομέρεια και βάθος. Συγκεκριμένα, καθορίζονται όλα εκείνα τα στοιχεία που απαιτούνται ώστε να υλοποιηθούν κατάλληλα οι μονάδες του συστήματος. Επιπλέον, λαμβάνονται υπ' όψη και όλα τα χαρακτηριστικά των μονάδων που δεν έχουν άμεση σχέση με την λειτουργικότητά τους αυτή καθ' εαυτή, π.χ., υψηλή απόδοση, αξιοπιστία, κτλ.
4. *Υλοποίηση και έλεγχος μονάδων*: Στο στάδιο αυτό, υλοποιείται και ελέγχεται κάθε μία μονάδα του συστήματος ξεχωριστά.
5. *Ολοκλήρωση και έλεγχος του όλου συστήματος*: Στο στάδιο αυτό, όλες οι επιμέρους μονάδες, ελεγμένες ως προς την ατομική τους ορθότητα και πληρότητα, συναρμολογούνται και δημιουργούν το συνολικό σύστημα το οποίο περνά και αυτό από διεξοδικό έλεγχο.
6. *Λειτουργία και συντήρηση*: Στο στάδιο αυτό, το σύστημα μπαίνει σε παραγωγική λειτουργία και συντηρείται με βάση ό,τι προβλήματα παρουσιάσει. Στο στάδιο αυτό δεν υπάρχει ουσιαστικά ανάπτυξη του συστήματος.



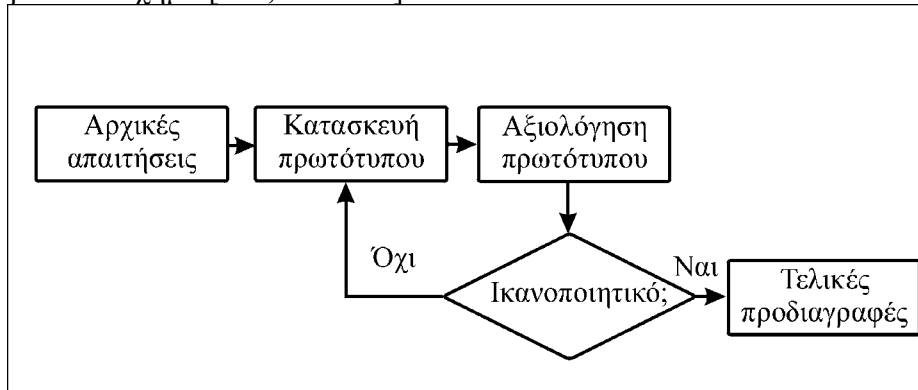


Παρατηρείστε ότι εν γένει υπάρχει ανάδραση από οποιοδήποτε στάδιο του μοντέλου σε οποιοδήποτε άλλο προηγούμενο για διόρθωση πιθανών λανθασμένων αποφάσεων, το οποίο είναι πολύ σημαντικό και για γενικά συστήματα λογισμικού αλλά και ειδικότερα για διαδραστικά συστήματα όπου δεν είναι δυνατόν να ξέρουμε όλες τις προδιαγραφές εκ των προτέρων. Επίσης θα πρέπει να τονιστεί ότι κατά μέσο όρο, το 50% του σχεδιαστικού χρόνου ενός οποιουδήποτε συστήματος λογισμικού καταναλώνεται στα διαδραστικά στοιχεία του, οπότε έχει πολύ μεγάλη σημασία να ακολουθείται μία σαφής και αυστηρή μεθοδολογία για αυτά ώστε να αποφεύγονται τα σφάλματα.

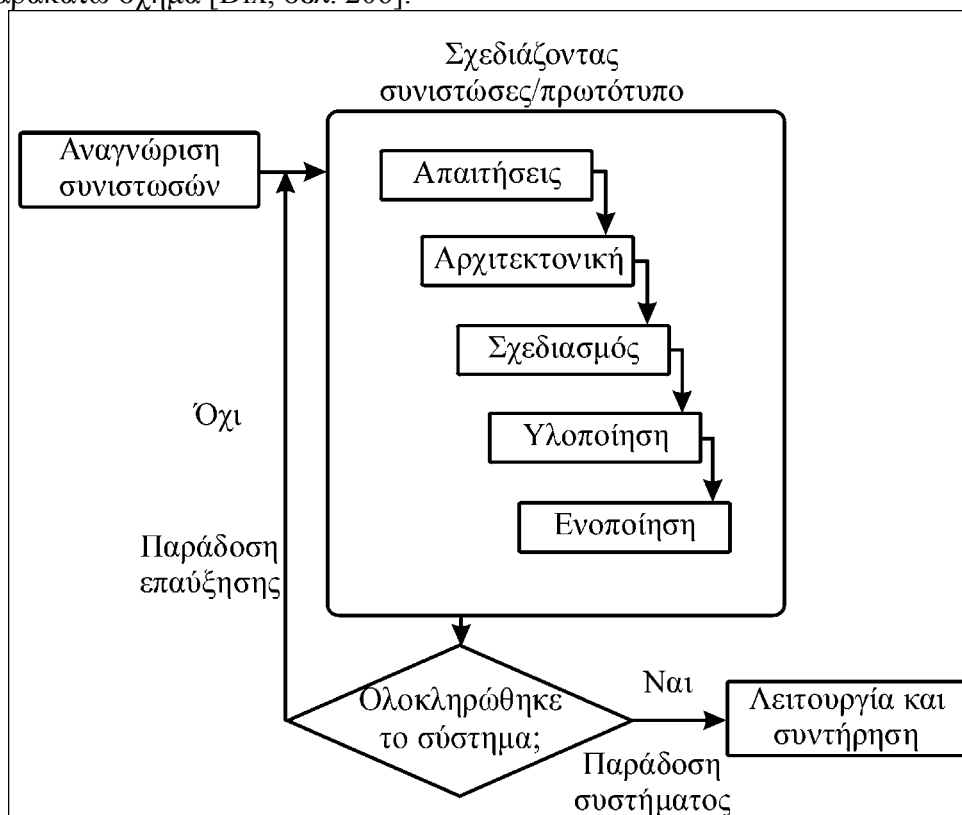
## 6.1 Επαναληπτικός Σχεδιασμός και Πρωτοτυποποίηση

Όπως αναφέραμε στην αμέσως προηγούμενη υποενότητα, ένα βασικό χαρακτηριστικό του μοντέλου ανάπτυξης λογισμικού EAM είναι η δυνατότητα μετάβασης σε προηγούμενα στάδια με βάση τα αποτελέσματα μεταγενέστερων σταδίων. Όταν το σύστημα που αναπτύσσεται είναι ιδιαίτερα πολύπλοκο, η ανάδραση αυτή δεν είναι δυνατόν να εφαρμοστεί με επιτυχία δουλεύοντας με το πλήρες σύστημα από την αρχή. Οι λόγοι είναι ότι η πολυπλοκότητα δημιουργεί πολλές ευκαιρίες για σφάλματα, η ανεύρεση αυτών των σφαλμάτων είναι δύσκολη και ακριβή, και το κόστος κάθε σταδίου αυτού καθ' εαυτού είναι πολύ υψηλό από την αρχή οπότε κάθε σφάλμα και διόρθωση συνεπάγεται την αχρήστευση του αποτελέσματος σημαντικής προσπάθειας. Για τον λόγο αυτό οι πρώτες διασχίσεις του μοντέλου καταρράκτη προς το κάτω/εμπρός γίνονται με υλοποίηση πρωτοτύπων τα οποία είναι πολύ απλούστερα και τα οποία γίνονται ολοένα και πιο πολύπλοκα μέχρι να φτάσουν να αντιπροσωπεύουν το συνολικό επιθυμητό σύστημα. Η αρχική απλότητα των πρωτοτύπων συνεπάγεται και το χαμηλό κόστος όλων των διαδικασιών, οπότε τα προβλήματα που αναφέρθηκαν προηγουμένως μειώνονται σημαντικά. Υπάρχουν τρεις βασικές προσεγγίσεις στην χρήση πρωτοτύπων μέσα στον κύκλο ανάπτυξης λογισμικού EAM:

1. *Για πέταμα (throw-away)*: Στην προσέγγιση αυτή, το πρωτότυπο υλοποιείται, ελέγχεται, τα στοιχεία που συλλέγονται από αυτό χρησιμοποιούνται για την υλοποίηση του τελικού συστήματος, αλλά το πρωτότυπο το ίδιο πετάγεται στα σκουπίδια. Η τεχνική αυτή ουσιαστικά χρησιμοποιείται με στόχο την ανάδραση στο στάδιο του προσδιορισμού των (τελικών) προδιαγραφών. Σχηματικά η πρωτοτυποποίηση 'για πέταμα ' αναπαριστάνεται στο παρακάτω σχήμα [Dix, σελ. 206].

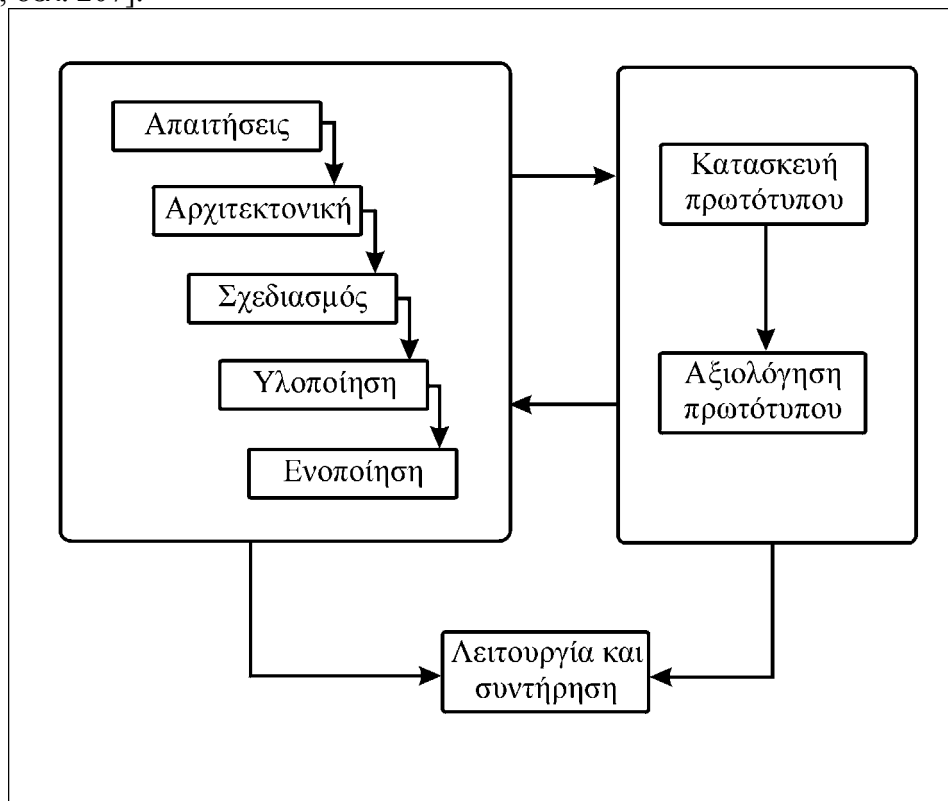


2. *Κλιμακωτή (incremental)*: Στην προσέγγιση αυτή, υπάρχει ένα αρχικό στάδιο πριν από αυτά του καταρράκτη, στο οποίο διαμερίζεται το όλο υπό ανάπτυξη σύστημα σε κάποιες συνιστώσες οι οποίες αναπτύσσονται μία-μία. Κάθε διάσχιση του καταρράκτη, δηλαδή, ασχολείται με μία μόνο από αυτές τις συνιστώσες, και η διαδικασία δεν ξεκινάει για κάποια συνιστώσα αν η προσπάθεια στην προηγούμενη δεν έχει ολοκληρωθεί ώστε να είναι αυτή έτοιμη για να λειτουργήσει. Σχηματικά η κλιμακωτή πρωτοτυποποίηση αναπαριστάνεται στο παρακάτω σχήμα [Dix, σελ. 206].



3. *Εξελικτική (evolutionary)*: Στην προσέγγιση αυτή, το πρωτότυπο είναι μια μικρογραφία του όλου συστήματος που δεν πετιέται στο τέλος αλλά εξελίσσεται σιγά-σιγά με κάθε επανάληψη των σταδίων του καταρράκτη μέχρι να φτάσει να είναι το σύστημα αυτό καθ'

εαυτό. Σχηματικά η εξελικτική πρωτοτυποποίηση αναπαριστάται στο παρακάτω σχήμα [Dix, σελ. 207].



Ανεξάρτητα από τον τρόπο με τον οποίον ένα πρωτότυπο χρησιμοποιείται στην διαδικασία ανάπτυξης ενός συστήματος, για την ίδια την πρωτοτυποποίηση υπάρχουν αρκετές τεχνικές που μπορούν να χρησιμοποιηθούν:

- *Στον πίνακα (storyboards)*: Είναι η απλούστερη μορφή πρωτοτυποποίησης και επικεντρώνεται στην εξωτερική εμφάνιση του υπό ανάπτυξη συστήματος χωρίς καθόλου συζήτηση της λειτουργικότητάς του. Είναι ουσιαστικά η τεχνική που χρησιμοποιείται στον κινηματογράφο όταν σκισάρονται οι σκηνές πριν κινηματογραφηθούν. Σήμερα υπάρχει η δυνατότητα χρήσης υπολογιστή αντί για πίνακα.
- *Προσομοιώσεις*: Όπως είναι κατανοητό, οι προσομοιώσεις δίνουν περισσότερη λειτουργικότητα και επιτρέπουν την εξαγωγή περισσότερων συμπερασμάτων.
- *Εξειδικευμένες γλώσσες προγραμματισμού*: Χρησιμοποιώντας γλώσσες προγραμματισμού που έχουν φτιαχτεί ειδικά για πρωτοτυποποίηση πετυχαίνουμε την εύκολη και γρήγορη κατασκευή του πρωτότυπου (rapid prototyping). Το μόνο μειονέκτημα είναι ότι η απόδοση αυτών των πρωτοτύπων είναι χαμηλή, οπότε δεν δίνουν καθόλου πληροφορία και για την απόδοση του τελικού συστήματος. Η ταχύτητα πρωτοτυποποίησης όμως που προσφέρουν είναι ένα πολύ σημαντικό χαρακτηριστικό, οπότε και πολλές φορές προτιμούνται από τις άλλες τεχνικές.

Τελειώνοντας με τον επαναληπτικό σχεδιασμό και την πρωτοτυποποίηση, πρέπει να επιστήσουμε την προσοχή στις αδυναμίες που έχει και αυτή η μεθοδολογία. Πολλές φορές αρχικές λανθασμένες αποφάσεις δεν αλλάζουν όσο και σωστή να είναι η πρωτοτυποποίηση. Επίσης έστω και σε ένα απλουστευμένο πρωτότυπο, είναι δύσκολο μερικές φορές να δει κανείς την πηγή του προβλήματος απλά από το σύμπτωμα. Πρέπει λοιπόν να χρησιμοποιούνται πρωτότυπα, αλλά ας έχουμε υπ' όψη μας ότι δεν είναι η πανάκεια.

---

Ερώτηση: Αναλαμβάνετε να αναπτύξετε:

- Ένα σύστημα διαχείρισης χαρτοφυλακίου μετοχών το οποίο θα εκτελείται μέσω Internet. Το νομικό πλαίσιο εκσυγχρονίζεται σε συχνά χρονικά διαστήματα, προσθέτοντας νέες δυνατότητες και τροποποιώντας παλαιότερες.
- Ένα Infokiosk για παροχή τουριστικών πληροφοριών για ένα νησί. Οι πελάτες δεν έχουν εμπειρία από τέτοια συστήματα και ήθελαν να δουν κάποια δοκιμαστικά πριν καταλήξουν σε τελικές προδιαγραφές.
- Ένα σύστημα διαχείρισης βάσης δεδομένων βιβλιοθήκης. Το σύστημα βασίζεται σε κάποιες βασικές λειτουργίες από τις οποίες υλοποιούνται και οι υπόλοιπες.
- Ένα σύστημα διαχείρισης ιστορικών ασθενών. Το σύστημα θα υλοποιηθεί παράλληλα από μια ομάδα συνεργαζόμενων φορέων.

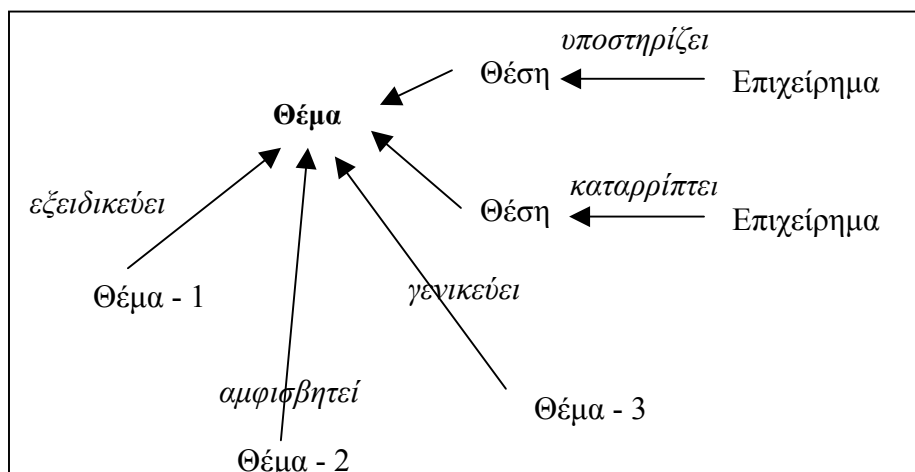
Ποια μεθοδολογία ανάπτυξης θα επιλέγατε σε κάθε περίπτωση και γιατί

## 6.2 Μεθοδολογίες Σχεδίασης

Ανεξάρτητα από το αντικείμενο μιας σχεδιαστικής προσπάθειας, ο τρόπος σκέψης και επεξήγησης που ακολουθείται στην διαδικασία παίζει σημαντικό ρόλο και μπορεί να οδηγήσει σε πολύ διαφορετικά αποτελέσματα διαφορετικής ποιότητας. Κυρίως όταν πρόκειται για σχεδίαση πολύπλοκων συστημάτων, αντικειμένων, ή διαδικασιών, χρειάζεται μια σχετικά αυστηρή μεθοδολογία για την σύγκριση και την εξέταση διαφόρων εναλλακτικών σχεδιαστικών λύσεων και την επιλογή της επικρατέστερης με βάση συγκρουόμενα κριτήρια. Η ανάγκη αυτή είναι ακόμη πιο επιτακτική όταν πρόκειται για ομαδική και όχι ατομική σχεδίαση, όπου υπεισέρχονται και συγκρούσεις μεταξύ πιθανών διαφορετικών απόψεων των ατόμων πάνω στα υπάρχοντα κριτήρια.

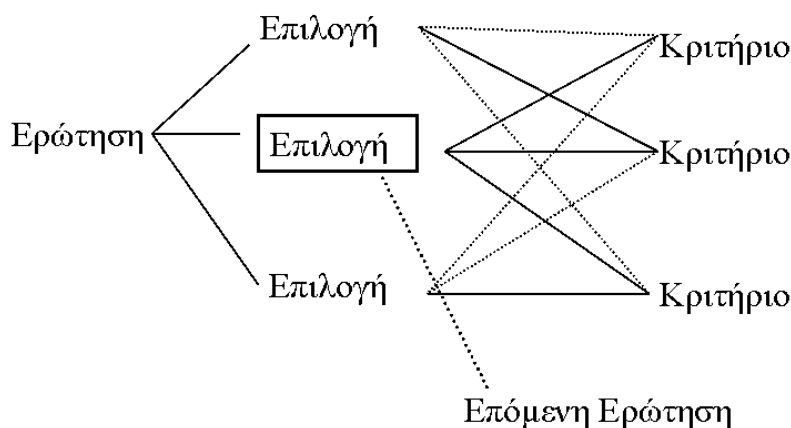
Υπάρχουν αρκετές μεθοδολογίες σχεδιασμού που εφαρμόζονται στην πράξη. Όλες μπορούν να εφαρμοστούν χρησιμοποιώντας απλά χαρτί και μολύβι αλλά και για τις περισσότερες υπάρχουν και συστήματα λογισμικού που διευκολύνουν, αυτοματοποιούν, αλλά και καθοδηγούν τις διαδικασίες. Αναφερόμαστε στις δύο πιο σημαντικές ίσως μεθοδολογίες:

- *Πληροφορικά Συστήματα Βασισμένα σε Θέματα (Issue-Based Information Systems - IBIS).* Κατά την μεθοδολογία αυτή, δημιουργείται ένας γράφος με τρεις διαφορετικούς τύπους κόμβων. Ο πρώτος τύπος αντιπροσωπεύει σχεδιαστικά θέματα, όπου τα κύρια θέματα μπαίνουν σαν αρχικοί κόμβοι, και άλλα σχετικά θέματα (ειδικότερα θέματα, γενικότερα θέματα, συγκρουόμενα θέματα, κτλ.) μπαίνουν σαν επιπρόσθετοι κόμβοι. Ο δεύτερος τύπος αντιπροσωπεύει διάφορες θέσεις που μπορεί να πάρει κάποιος απέναντι σε κάποιο θέμα, ενώ ο τρίτος τύπος αντιπροσωπεύει επιχειρήματα που υποστηρίζουν ή καταρρίπτουν κάποιες θέσεις. Η βασική μορφή ενός τέτοιου γράφου φαίνεται παρακάτω [Dix, σελ. 215].



Ο γράφος αυτός μεταλλάσσεται συνεχώς κατά την διάρκεια του σχεδιασμού με προσθαφαιρέσεις κόμβων και ακμών. Η διαδικασία αυτή σταματά όταν πλέον δεν μπορούν να γίνουν άλλες αλλαγές, οπότε και (ελπίζει κανείς ότι) ο γράφος θα αναδεικνύει μία οι περισσότερες σωστές σχεδιαστικές λύσεις. Η μεθοδολογία αυτή είναι πολύ χρήσιμη κατά την διαδικασία της σχεδίασης αυτής καθ' εαυτής, οπότε οι διάφορες μορφές από τις οποίες περνά ο γράφος χρησιμεύουν και σαν πρακτικό της διαδικασίας, ενώ η τελική μορφή του μπορεί να αποτελέσει και την περιγραφή του τελικού αποτελέσματος.

- *Ανάλυση Χώρου Σχεδιασμού.* Η μεθοδολογία αυτή είναι παρόμοια με την προηγούμενη όσον αφορά το γεγονός ότι το αποτέλεσμα είναι πάλι ένας γράφος με κόμβους τα βασικά σχεδιαστικά θέματα, τις επιλογές αντιμετώπισής τους, και τα κριτήρια με βάση τα οποία μία επιλογή κρίνεται ως θετική ή αρνητική, και ακμές τις κατάλληλες συνδέσεις μεταξύ τους. Η βασική μορφή ενός τέτοιου γράφου φαίνεται παρακάτω, όπου οι διακεκομμένες ακμές μεταξύ κριτηρίων και επιλογών συμβολίζουν την αρνητική αποτίμηση των επιλογών σε σχέση με τα συνδεδεμένα κριτήρια, ενώ οι πλήρεις ακμές συμβολίζουν την θετική αποτίμηση [Dix, σελ. 216].



Η βασική διαφορά αυτής της μεθοδολογίας από την προηγούμενη είναι ότι αυτή δεν προορίζεται για χρήση κατά την διάρκεια μιας σχεδιαστικής προσπάθειας. Αντίθετα χρησιμοποιείται μετά την κατάληξη της προσπάθειας, και προσπαθεί να αποτυπώσει την γενική εικόνα του σχεδιαστικού χώρου έτσι όπως έχει γίνει τελικά κατανοητός από την σχεδιαστική ομάδα. Ένα ενδιαφέρον ερώτημα είναι πόσο μοιάζει ο γράφος αυτής της μεθόδου που βγαίνει εκ των υστέρων με τον τελικό γράφο που προκύπτει από την δυναμική διαδικασία του IBIS.

Η χρήση μιας οποιασδήποτε από τις παραπάνω διαδικασίες θεωρείται απαραίτητη για την ικανοποίηση κάποιων απαιτήσεων σχεδιαστικής ποιότητας σε πολύπλοκα σχεδιαστικά προβλήματα.



## **Κεφάλαιο 7 Καταγραφή Προδιαγραφών Συστημάτων EAM**

### **7.1 Προδιαγραφές Βασικών Συντελεστών EAM**

Η καταγραφή προδιαγραφών ξεκινά με την ανάλυση των απαιτήσεων των χρηστών. Η καταγραφή προδιαγραφών, περιλαμβάνει τους τρεις βασικούς συντελεστές της αλληλεπίδρασης ανθρώπου-μηχανής: τον άνθρωπο-χρήστη, τον υπολογιστή και τη διεργασία.

Περισσότερες λεπτομέρειες για τη διαδικασία της ανάλυσης απαιτήσεων των χρηστών και της καταγραφής των προδιαγραφών δίνονται αμέσως παρακάτω σε ξεχωριστές υποενότητες, ενώ ακολουθούν δύο ενότητες που πραγματεύονται επιπρόσθετες μεθόδους που ακολουθούνται ειδικά στον καθορισμό προδιαγραφών διαλογικών συστημάτων επικεντρωμένων στον άνθρωπο: την χρήση οδηγιών και προτύπων, και την χρήση μηχανικής χρηστικότητας.

#### **Ανάλυση απαιτήσεων χρηστών**

Με δεδομένο ότι το σύστημα απευθύνεται σε χρήστες το πρώτο βήμα ενός σχεδιαστή λογισμικού είναι η γνωριμία με τους χρήστες του συστήματος.

Ο σχεδιαστής παρακολουθεί τους χρήστες στην εργασία τους και σημειώνει τις εργασίες που εκτελούνται, τα έγγραφα που διακινούνται ή τηρούνται, αλλά και στοιχεία όπως πόσο συχνά κτυπά το τηλέφωνο, γίνεται διακοπή ρεύματος, κλπ. Ειδικά αυτά τα στοιχεία έχουν αρκετές φορές μεγάλη σημασία στο σχεδιασμό ενός εύχρηστου συστήματος.

Εάν για παράδειγμα ο σχεδιαστής παρατηρήσει ότι η προσοχή του χρήστη αποσπάται συχνά από την εργασία του λόγω των τηλεφώνων ή των ερωτήσεων του κοινού, το σύστημα μπορεί να εμφανίζει τις εργασίες που έχει εκτελέσει μέχρι εκείνη τη στιγμή ο χρήστης και αυτές που είναι ανεκτέλεστες. Εάν πάλι γίνεται συχνά διακοπή ρεύματος το σύστημα μπορεί να κρατά προσωρινά αντίγραφα ασφαλείας σε τακτά χρονικά διαστήματα ή να τεθεί ως προδιαγραφή η ύπαρξη συστήματος αδιαλείπτου παροχής ρεύματος.

Η καταγραφή των εργασιών που εκτελούνται είναι αναγκαία επειδή προσφέρει μια πρώτη εικόνα για τη λειτουργικότητα που πρέπει να παρέχει το σύστημα. Αντίστοιχα, η καταγραφή των εγγράφων που τηρούνται θα προσφέρει μια εικόνα των δεδομένων που πρέπει να διατηρεί η βάση δεδομένων και η καταγραφή των εγγράφων που διακινούνται τις παραγόμενες και τις εισερχόμενες καταστάσεις. Κατά τη διάρκεια της ανάλυσης απαιτήσεων ο σχεδιαστής μπορεί να χρησιμοποιήσει εκτός από την απλή παρατήρηση, συνεντεύξεις ή ερωτηματολόγια προς τους χρήστες.

Κατά την ανάλυση απαιτήσεων των χρηστών καταγράφεται και η παρούσα κατάσταση. Η παρούσα κατάσταση περιγράφει το υπάρχον σύστημα που μπορεί να είναι μηχανογραφικό ή χειρογραφικό. Όμως η ανάλυση απαιτήσεων δεν σταματά εκεί. Καταγράφει πιθανές αλλαγές οι οποίες θα βελτιώσουν την απόδοση του συστήματος ή και απαιτούνται για την ορθή λειτουργία του νέου συστήματος. Σημαντικό βήμα κατά την ανάλυση απαιτήσεων είναι και η οριοθέτηση της ομάδας των χρηστών (target group). Κατά την καταγραφή των απαιτήσεων της συγκεκριμένης ομάδας θα πρέπει να ληφθούν υπόψη παράμετροι όπως οι εξής:

- Πολιτισμικά χαρακτηριστικά της ομάδας των χρηστών. Στοιχεία όπως η μητρική γλώσσα των χρηστών, οι πολιτισμικές ιδιαιτερότητες παίζουν ιδιαίτερο ρόλο. Είναι πιθανό πως θα πρέπει να υποστηριχθούν διαφορετικά αλφάβητα, να αλλαχθούν εικόνες που έχουν διαφορετικό νόημα σε διαφορετικές ομάδες, κτλ.
- Εμπειρία της ομάδας των χρηστών είτε γενικά με υπολογιστές είτε με ανάλογα συστήματα.
- Απαιτήσεις ατόμων με ειδικές ανάγκες

Αφού έχει ολοκληρωθεί η ανάλυση απαιτήσεων των χρηστών, ο σχεδιαστής προχωρά στον καθορισμό των προδιαγραφών που θα αποτελέσουν τη βάση για το σχεδιασμό του συστήματος.

### Καταγραφή προδιαγραφών

Πριν από τις επιμέρους συζητήσεις, πρέπει να τονίσουμε ότι η κατάρτιση προδιαγραφών είναι η συνισταμένη πολλών, μερικές φορές αντίρροπων δυνάμεων. Ως παράδειγμα μπορούμε να αναφέρουμε την περίπτωση του σχεδιαστή που επιθυμεί να υλοποιήσει το 'καταπληκτικό' ηλεκτρονικό παιχνίδι με διεπαφή εικονικής πραγματικότητας (και ελάχιστες προδιαγραφές γάντι και κράνος εικονικής πραγματικότητας). Παρότι ένα τέτοιο παιχνίδι θα είναι πιθανότατα καταπληκτικό, ελάχιστοι χρήστες θα έχουν την οικονομική δυνατότητα να το αποκτήσουν. Τελικά οι προδιαγραφές θα είναι αποτέλεσμα συμβιβασμού.

### Διεργασία

Η καταγραφή των προδιαγραφών του λογισμικού θα βασιστεί στην ανάλυση απαιτήσεων των χρηστών. Σε αυτήν την περίπτωση ο ρόλος του σχεδιαστή του λογισμικού είναι η "μετάφραση" της ανάλυσης απαιτήσεων σε προδιαγραφές οι οποίες μπορούν να χρησιμοποιηθούν άμεσα στη φάση του σχεδιασμού. Εάν για παράδειγμα οι χρήστες έχουν ζητήσει τη δυνατότητα να δουν στατιστικά αποτελέσματα για περασμένα έτη, ο σχεδιαστής θα πρέπει να θέσει ως προδιαγραφή την τήρηση δεδομένων παλαιότερων ετών. Αντίστοιχα, εάν οι χρήστες θέλουν να μπορούν να μεταφέρουν τα δεδομένα τους σε κάποια εφαρμογή λογιστικών φύλλων όπως το Excel, θα πρέπει να τεθεί ως προδιαγραφή η δυνατότητα αποθήκευσης των δεδομένων σε μια μορφή αναγνώσιμη από το συγκεκριμένο πρόγραμμα.

Κατά τη διάρκεια της καταγραφής των προδιαγραφών θα πρέπει να ληφθούν υπόψη το νομικό πλαίσιο καθώς και πιθανοί κανονισμοί και οδηγίες που διέπουν και επηρεάζουν τη λειτουργία του συστήματος. Σημαντικές είναι και οι παγιωμένες ή καθιερωμένες αντιλήψεις που αφορούν το ευρύτερο χώρο της εφαρμογής (π.χ., αντιστοιχία χρωμάτων - κομμάτων για την εφαρμογή απεικόνισης των εκλογικών αποτελεσμάτων των εκλογών). Μέσα από την καταγραφή των προδιαγραφών της διεργασίας θα πρέπει να αναδεικνύεται η λειτουργικότητά της. Οι λειτουργίες που θα προσφέρονται στο χρήστη θα πρέπει να ομαδοποιηθούν σε κατηγορίες (γενικές, ειδικές, ανάλογα, με την εμπειρία των χρηστών, τα δικαιώματα, κτλ.).

Έστω ότι κατά τη διάρκεια της ανάλυσης απαιτήσεων ενός συστήματος μηχανογράφησης μιας γραμματείας, ο σχεδιαστής έχει παρατηρήσει ότι για τη διαχείριση των εγγραφών των φοιτητών οι χειριστές του συστήματος επιθυμούν να μπορούν να:

- Εισάγουν νέους φοιτητές (μεμονωμένα ή μαζικά)
- Διαγράφουν λανθασμένες εγγραφές ή διπλοεγγραφές
- Εκτυπώνουν την κατάσταση ενός φοιτητή είτε για όλα τα μαθήματα που έχει δηλώσει είτε για τα μαθήματα στα οποία έχει εξετασθεί επιτυχώς
- Εκτυπώνουν την κατάσταση των φοιτητών που έχουν δηλώσει ένα μάθημα
- Διατηρούν λίστα με τις τροποποιήσεις που έγιναν στα αρχεία των φοιτητών κάθε μέρα, μαζί με το όνομα αυτού που έκανε την τροποποίηση
- Τυπώνουν πιστοποιητικά για την εφορία και το στρατό
- Εισάγουν αποτελέσματα εξετάσεων

Τότε οι αντίστοιχες προδιαγραφές μπορούν να αποτυπωθούν ως εξής<sup>1</sup>:

Διαχείριση φοιτητών

Εισαγωγή

Μεμονωμένη

<sup>1</sup> Σε πραγματικές συνθήκες η ανάλυση απαιτήσεων - καταγραφή προδιαγραφών περιλαμβάνει επεξήγηση αυτών των προδιαγραφών, αποτύπωση των αλλαγών οι οποίες θα καταστούν δυνατές με την υιοθέτηση του νέου συστήματος, καθώς και τα στοιχεία (έντυπα, φόρμες, κλπ.) που έχουν συλλεχθεί στην ανάλυση απαιτήσεων.



- Μαζική
- Επεξεργασία
  - Τροποποίηση στοιχείων εγγραφής
  - Τροποποίηση στοιχείων μαθημάτων (δηλώσεις)
- Εκτύπωση
  - Λίστα μαθημάτων
    - Δηλωθέντων σε χρονική περίοδο
    - Εξετασθέντων επιτυχώς
  - Πιστοποιητικών
    - Εφορία
    - Στρατό
- Διαχείριση μαθημάτων
  - Εισαγωγή φοιτητών που έχουν δηλώσει μάθημα
  - Εκτύπωση λίστας φοιτητών που έχουν δηλώσει μάθημα
  - Εισαγωγή αποτελεσμάτων εξετάσεων
  - Εισαγωγή νέων μαθημάτων
  - Αλλαγή κωδικού μαθήματος

Όπως φαίνεται και στην παραπάνω λίστα οι προδιαγραφές απαντούν στο τι θέλουμε να υλοποιήσουμε και όχι στο πώς (που είναι θέμα σχεδιασμού).

### **Υπολογιστής**

Ανάλογα με τις προδιαγραφές της διεπαφής (και του λογισμικού γενικότερα), η καταγραφή των προδιαγραφών του υπολογιστή περιλαμβάνει χαρακτηριστικά που αφορούν τα μηχανήματα στα οποία θα εκτελείται το λογισμικό. Θέματα όπως η οθόνη (χρώμα, ανάλυση, ...), η χρήση κάποιας συγκεκριμένης συσκευής κατάδειξης (ποντίκι, στυλό φωτός, ...), ο τύπος του επεξεργαστή και η μνήμη που διαθέτει ο υπολογιστής, το λειτουργικό σύστημα, η ταχύτητα του δικτύου καθώς και η χρήση ειδικών περιφερειακών (π.χ., μονάδα ανάγνωσης έξυπνων καρτών ή πιστωτικών καρτών) θα πρέπει να εξετασθούν. Παράλληλα, σημαντικό ρόλο παίζει και το περιβάλλον στο οποίο θα λειτουργεί ο υπολογιστής (π.χ., Info kiosk ή ο υπολογιστής του αεροπλάνου).

## **7.2 Παραδειγματική Άσκηση**

Ως παραδειγματική άσκηση θα εξετάσουμε την καταγραφή προδιαγραφών για ένα σύστημα διαχείρισης χαρτοφυλακίου μετοχών. Από την ανάλυση απαιτήσεων της χρηματιστηριακής εταιρείας η οποία ενδιαφέρεται να αναπτύξει το συγκεκριμένο σύστημα γνωρίζουμε τα ακόλουθα: Οι βασικοί χρήστες του συστήματος θα είναι οι πελάτες της εν λόγω εταιρείας καθώς και οι χρηματιστές της. Επιπλέον η εταιρεία επιθυμεί το σύστημα να είναι προσβάσιμο και σε πιθανούς νέους πελάτες αλλά με μειωμένο αριθμό ενεργοποιημένων λειτουργιών. Η εταιρεία διαθέτει ήδη κάποια άτομα υπεύθυνα για τη μηχανοργάνωση τα οποία θα είναι οι αποκλειστικά αρμόδιοι για τη διαχείριση του συστήματος. Το σύστημα θα παρέχει τη δυνατότητα στους εγγεγραμμένους χρήστες να εκτελούν αγοραπωλησίες μετοχών. Όλοι οι χρήστες θα έχουν τη δυνατότητα να βλέπουν στατιστικά (γραφήματα ή/και πίνακες) στοιχεία για το χρηματιστήριο, συγκεκριμένες μετοχές ή κατηγορίες μετοχών, δείκτες είτε σε άμεση σύνδεση με το χρηματιστήριο για την τρέχουσα συνεδρίαση είτε για παλαιότερες χρονικές περιόδους που μπορούν να καθορίσουν οι ίδιοι. Το σύστημα θα πρέπει να είναι προσβάσιμο μέσω του Internet και θα πρέπει να μπορεί να εκτελεστεί και σε υπολογιστές με μειωμένη επεξεργαστική ισχύ.

### Καταγραφή προδιαγραφών παραδειγματικής άσκησης

Από την παραπάνω (περιληπτική) ανάλυση απαιτήσεων είναι δυνατή η καταγραφή των προδιαγραφών του συστήματος διαχείρισης μετοχών. Προς αυτήν την κατεύθυνση ομαδοποιούμε ιεραρχικά τις λειτουργίες<sup>2</sup>. Στα πλαίσια της παραδειγματικής άσκησης δεν θα αναπτύξουμε το σύνολο των πιθανών λειτουργιών, αλλά θα περιοριστούμε σε ένα υποσύνολο αυτών, προσπαθώντας να αναδείξουμε τη μορφή που θα έχουν οι προδιαγραφές. Για κάθε λειτουργία αναφέρουμε ποιος έχει δικαίωμα να την εκτελέσει, τι δεδομένα χρειάζεται για να εκτελεστεί και τι αποτέλεσμα έχει. Εκτός από τις λειτουργίες αναφέρονται και κάποιες γενικές προδιαγραφές που θα πρέπει να πληροί το σύστημα και που επηρεάζουν τη διεπαφή.

#### Σύστημα

Σύνδεση (εγγεγραμμένος χρήστης, χρηματιστής, διαχειριστής)

[δεδομένα: όνομα χρήστη και συνθηματικό αποτελέσματα: είσοδος στο σύστημα ή μήνυμα λάθους]

Επιλογή χρήστη (χρηματιστής)

Η λειτουργία εκτελείται μόνο από το χρηματιστή και του επιτρέπει να επιλέξει κάποιο χρήστη και να διαχειριστεί το χαρτοφυλάκιό του.

[δεδομένα: όνομα χρήστη (username) ή ονοματεπώνυμο χρήστη]

Αποσύνδεση (εγγεγραμμένος χρήστης, χρηματιστής, διαχειριστής)

Η λειτουργία αποτελεί την έξοδο από το σύστημα.

[δεδομένα: - ]

#### Χαρτοφυλάκιο

Δημιουργία (εγγεγραμμένος χρήστης, χρηματιστής)

Επιτρέπει τη δημιουργία ενός νέου χαρτοφυλακίου μετοχών.

[δεδομένα: όνομα χρήστη]

Τροποποίηση (εγγεγραμμένος χρήστης, χρηματιστής)

Προσθήκη μετοχής

[δεδομένα: όνομα χρήστη, όνομα ή διακριτικό μετοχής, προαιρετικά ημερομηνία αγοράς, τρόπος απόκτησης, τιμή κτήσης, τεμάχια]

Αφαίρεση μετοχής

[δεδομένα: όνομα χρήστη, όνομα ή διακριτικό μετοχής, τεμάχια]

Διαγραφή (εγγεγραμμένος χρήστης, χρηματιστής)

[δεδομένα: όνομα χρήστη, όνομα χαρτοφυλακίου]

Εντολή αγοράς (εγγεγραμμένος χρήστης, χρηματιστής)

Πρόκειται για διαβίβαση εντολής αγοράς μιας μετοχής από μέρος του χρήστη.

[δεδομένα: όνομα χρήστη, όνομα ή διακριτικό μετοχής, τεμάχια, τιμή, αποτέλεσμα: επιβεβαίωση αγοράς ή απόρριψη αγοράς, χρέωση στο χαρτοφυλάκιο]

Εντολή πώλησης (εγγεγραμμένος χρήστης, χρηματιστής)

Πρόκειται για διαβίβαση εντολής πώλησης μιας μετοχής από μέρος του χρήστη.

[δεδομένα: όνομα χρήστη, όνομα ή διακριτικό μετοχής, τεμάχια, τιμή, αποτέλεσμα: επιβεβαίωση πώλησης ή απόρριψη πώλησης, χρέωση χαρτοφυλακίου]

#### Αποτίμηση

Τρέχουσας αξίας (όλοι)

Εμφανίζει την αξία του χαρτοφυλακίου με τις τρέχουσες τιμές των μετοχών.

[δεδομένα: όνομα χρήστη, όνομα ή διακριτικό μετοχής, αποτέλεσμα: πίνακας τιμών]

Αξία κτήσης (εγγεγραμμένος χρήστης, χρηματιστής)

Εμφανίζει την αξία του χαρτοφυλακίου με βάση την αξία των μετοχών όταν αγοράστηκαν.

<sup>2</sup> Σε μια πραγματική κατάσταση οι προδιαγραφές θα κάλυπταν περισσότερα θέματα όπως π.χ. τη βάση δεδομένων. Στην παρούσα ανάλυση επικεντρώναστε στη διεπαφή.

[δεδομένα: όνομα χρήστη, όνομα ή διακριτικό μετοχής ,αποτέλεσμα: πίνακας τιμών]

Στατιστικά (όλοι – εγγεγραμμένοι χρήστες ή μη)

Διακύμανση μετοχών

Εμφανίζει τη διακύμανση μιας ή περισσότερων μετοχών σε κάποια χρονική περίοδο

[δεδομένα: όνομα χρήστη, όνομα ή διακριτικό μετοχών, χρονικό διάστημα, αποτέλεσμα: πίνακας τιμών ή γράφημα]

Διακύμανση δεικτών

Εμφανίζει τη διακύμανση ενός ή περισσότερων δεικτών σε κάποια χρονική περίοδο

[δεδομένα: όνομα χρήστη, όνομα ή διακριτικό μετοχών, χρονικό διάστημα, αποτέλεσμα: πίνακας τιμών ή γράφημα]

Διαχείριση χρηστών (διαχειριστής)

Δημιουργία

[δεδομένα: στοιχεία νέου χρήστη, ομάδα χρήστη (απλός ή χρηματιστής)]

Τροποποίηση

[δεδομένα: στοιχεία χρήστη]

Διαγραφή

[δεδομένα: όνομα χρήστη]

Γενικές προδιαγραφές

Χρόνος ανάδρασης

Για όλες τις λειτουργίες (εκτός αγοράς και πώλησης) μέγιστος χρόνος αναμονής δέκα δευτερόλεπτα, επιθυμητός χρόνος ένα δευτερόλεπτο.

Ασφάλεια

Επιπλέον ασφάλεια για τους χρηματιστές με την ταυτόχρονη χρήση έξυπνης κάρτας για τη σύνδεση στο σύστημα.

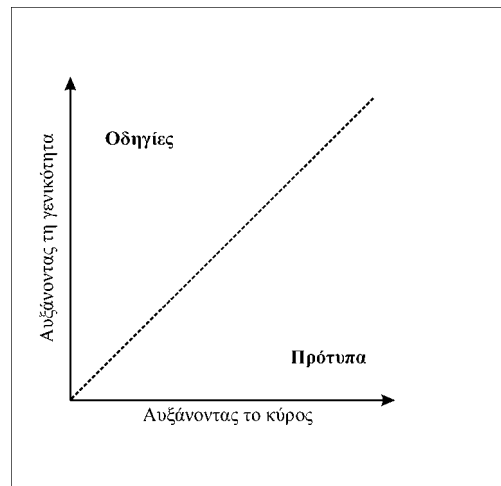
Προδιαγραφές υπολογιστή

Ελάχιστες: Οθόνη 15", ανάλυση 640x480, επεξεργαστής Intel Pentium 200MHz ή συμβατός, μνήμη 16 MB, modem 28.8 kbps.

Επιθυμητές: Οθόνη 17", ανάλυση 1024x768, επεξεργαστής Intel Pentium II 400MHz ή συμβατός, μνήμη 64 MB, modem 56 kbps.

### 7.3 Οδηγίες (Guidelines) και Πρότυπα (Standards)

Κατά τη διάρκεια της καταγραφής των προδιαγραφών πρέπει να ληφθούν υπόψη σχετικά πρότυπα και οδηγίες. Αυτά είναι δυνατό να αφορούν τόσο τον άνθρωπο (π.χ., εργονομικά πρότυπα), το υπολογιστικό σύστημα (π.χ., πρότυπα ακτινοβολίας οθόνης), όσο και τη διεργασία (π.χ., πρότυπα για ηλεκτρονική ανταλλαγή δεδομένων (electronic data interchange - EDI). Είτε επιλέξουμε οδηγίες είτε πρότυπα στην διαδικασία αυτή των προδιαγραφών πετυχαίνουμε διάφορους επιθυμητούς στόχους: αυξάνουμε την χρηστικότητα, αποφεύγουμε υπερβολικές επιλογές, και αποφεύγουμε και την επανάληψη διαδικασιών και ανακαλύψεων που έχουν ήδη κάνει άλλοι (δηλαδή αποφεύγουμε την λεγόμενη "επανάληψη της ανακάλυψης του τροχού").



Οι οδηγίες και τα πρότυπα έχουν συμπληρωματικά χαρακτηριστικά, όπως διαφαίνεται στην παραπάνω εικόνα. Οι κύριες διαφορές μεταξύ των δύο έχουν ως εξής:

- Τα πρότυπα συντάσσονται από διεθνείς και εθνικούς οργανισμούς ή άλλους επίσημους φορείς οπότε έχουν μεγαλύτερο κύρος. Βασίζονται σε λανθάνουσα θεωρία και παρακολουθούν τις εξελίξεις της τεχνολογίας. Συνήθως είναι αρκετά εξειδικευμένα και περιορίζουν πολύ τον συντάκτη των προδιαγραφών αφού οι υποδείξεις τους είναι ιδιαίτερα αυστηρές. Ένα άλλο πρόβλημα με τα πρότυπα είναι και αυτό που έχει πει ο Tannenbaum: "Το ωραίο με τα πρότυπα είναι ότι υπάρχουν τόσο πολλά που μπορείς να διαλέξεις όποιο σε βολεύει!".
- Οι οδηγίες συντάσσονται από ανεπίσημες ομάδες επιστημόνων, και καλύπτουν κενά που υπάρχουν λόγω έλλειψης πλήρους θεωρίας, όπως συμβαίνει για παράδειγμα στα διαλογικά συστήματα. Δεν έχουν βέβαια το κύρος των προτύπων, αλλά έχουν το πλεονέκτημα ότι είναι αρκετά γενικές οπότε αφήνουν αρκετά περιθώρια ελευθερίας και φαντασίας στον συντάκτη των προδιαγραφών.

Η επιλογή μεταξύ οδηγιών και προτύπων εξαρτάται κυρίως από τον συντάκτη των προδιαγραφών, και βασίζεται στις ανάγκες του συγκεκριμένου συστήματος όπως αυτές αντιπαραβάλλονται με τα χαρακτηριστικά των οδηγιών και των προτύπων που αναφέραμε παραπάνω. Να τονίσουμε επίσης ότι η χρήση των οδηγιών και των προτύπων δεν περιορίζεται μόνο στην καταγραφή προδιαγραφών αλλά μπορεί να επεκταθεί και στην σχεδίαση συστημάτων.

#### 7.4 Μηχανική Χρηστικότητα (Usability Engineering)

Μία άλλη μέθοδος που εφαρμόζεται στον καθορισμό προδιαγραφών συστημάτων EAM είναι και η λεγόμενη μηχανική χρηστικότητας. Χρησιμοποιείται κυρίως στον καθορισμό αρκετά συγκεκριμένων και υψηλής σημασίας πραγμάτων. Μπορεί να εφαρμοσθεί σε πολλά χαρακτηριστικά γνώρισμα ενός διαλογικού συστήματος. Η συζήτησή μας θα περιοριστεί σε ένα παράδειγμα μέσα από το οποίο θα κατανοήσουμε τις βασικές αρχές της μηχανικής χρηστικότητας αλλά και συγκεκριμένες επιλογές που υπάρχουν για τα διάφορα ερωτήματα που δημιουργούνται κατά την διάρκεια της εφαρμογής της. Το παράδειγμα σχετίζεται με την αρχή χρηστικότητας της δυνατότητας αντιστροφής ενεργειών που αναφέραμε στο Κεφάλαιο 5 και αφορά τον καθορισμό προδιαγραφών για την λειτουργία της αναίρεσης ενός προγράμματος μαγνητοσκόπησης σε ένα βίντεο. Ο παρακάτω πίνακας περιέχει ένα δείγμα προδιαγραφών για την περίπτωση [Dix, σελ. 200]. Η αριστερή του στήλη είναι προκαθορισμένη από την μεθοδολογία της μηχανικής χρηστικότητας αυτής καθ' εαυτής.

Αρχή χρηστικότητας:	Δυνατότητα αντιστροφής ενεργειών
Μετρήσιμη λειτουργία:	Αναίρεση (undo) μιας εσφαλμένης προγραμματιστικής ακολουθίας
Μέθοδος μέτρησης:	Ο αριθμός των διακεκριμένων ενεργειών του χρήστη για την αναίρεση του τρέχοντος προγράμματος.
Τρέχον επίπεδο:	Κανένα προϊόν δεν επιτρέπει τέτοιου τύπου αναίρεση.
Χειρότερη περίπτωση:	Όσες ενέργειες χρειάζεται για τον προγραμματισμό να είναι εσφαλμένες
Σχεδιαζόμενο/ προβλεπόμενο επίπεδο:	Ένας μέγιστος αριθμός δύο διακεκριμένων ενεργειών του χρήστη.
Βέλτιστη περίπτωση:	Μια διακεκριμένη εντολή ακύρωσης.

Στον πίνακα αυτό, η μετρήσιμη λειτουργία είναι μία συγκεκριμένη περίπτωση στην οποία ενσαρκώνεται η γενική αρχή χρηστικότητας που αναφέρεται πιο πάνω. Για την μέθοδο μέτρησης (την μετρήσιμη παράμετρο, δηλαδή) υπάρχουν πολλές επιλογές, όπως αυτές που ακολουθούν [Dix, σελ. 201].

1. Ο χρόνος ολοκλήρωσης μιας ενέργειας
2. Το ποσοστό ολοκλήρωσης της ενέργειας
3. Το ποσοστό ολοκλήρωσης της ενέργειας στη μονάδα του χρόνου
4. Η αναλογία επιτυχιών προς αποτυχίες
5. Ο χρόνος που αναλώθηκε σε σφάλματα
6. Το ποσοστό ή αριθμός σφαλμάτων
7. Το ποσοστό ή ο αριθμός των ανταγωνιστών που ήταν καλύτεροι
8. Ο αριθμός των εντολών που χρησιμοποιήθηκαν
9. Η συχνότητα χρήσης της βοήθειας ή της τεκμηρίωσης
10. Το ποσοστό των θετικών/ αρνητικών σχολίων των χρηστών
11. Ο αριθμός επανάληψης των ανεπιτυχών εντολών
12. Ο αριθμός συνεχόμενων επιτυχιών ή αποτυχιών
13. Ο αριθμός των περιπτώσεων που η διεπαφή παραπλάνησε το χρήστη
14. Ο αριθμός των θετικών και αρνητικών χαρακτηριστικών που θυμόνται οι χρήστες
15. Ο αριθμός των εντολών που δε χρησιμοποιήθηκαν
16. Ο αριθμός των regressive behaviors
17. Ο αριθμός των χρηστών που προτιμούν το σύστημά σας
18. Ο αριθμός των περιπτώσεων στις οποίες οι χρήστες αναγκάζονται να βρουν λύση σε κάποιο πρόβλημα
19. Ο αριθμός των περιπτώσεων στις οποίες ο χρήστης διακόπηκε από κάποια ενέργεια
20. Ο αριθμός των περιπτώσεων στις οποίες ο χρήστης χάνει τον έλεγχο του συστήματος
21. Ο αριθμός των περιπτώσεων στις οποίες ο χρήστης εκφράζει δυσαρέσκεια ή ευχαρίστηση

Με βάση την μηχανική χρηστικότητας, πρέπει να καθοριστούν τέσσερις τιμές για αυτή την παράμετρο, που εμφανίζονται σαν οι τέσσερις τελευταίες γραμμές του πίνακα στον παραπάνω πίνακα της προδιαγραφής του συγκεκριμένου παραδείγματος:

- η τιμή της στα τρέχοντα συστήματα (ή στην τρέχουσα έκδοση του ίδιου συστήματος όταν πρόκειται για επανασχεδιασμό)
- η χειρότερη τιμή που θα θεωρηθεί εντός προδιαγραφών
- η επιθυμητή τιμή
- η βέλτιστη τιμή που είναι βέβαια ακόμη πιο επιθυμητή αλλά για την επίτευξη της οποίας δεν θα πρέπει να ανέλθει το κόστος.

Οι τιμές που μπορούν να δοθούν στην μετρήσιμη παράμετρο για οποιοδήποτε από τα παραπάνω επίπεδα είναι πάντα σχετικά με κάποιο από τα παρακάτω [Dix, σελ. 202]:

1. ένα υπάρχον σύστημα ή προηγούμενη έκδοση
2. ένα ανταγωνιστικό σύστημα
3. την εκτέλεση της ενέργειας χωρίς τη χρήση ενός υπολογιστικού συστήματος
4. μια απόλυτη κλίμακα
5. το δικό σας πρωτότυπο
6. τις προηγούμενες επιδόσεις του χρήστη
7. καθένα από τα συστατικά του συστήματος χωριστά
8. ένα διαδοχικό μοίρασμα της διαφοράς μεταξύ της καλύτερης και της χειρότερης τιμής που παρατηρήθηκε στις δοκιμές με χρήστες.

Τελειώνοντας να αναφέρουμε ότι ένα πρόβλημα με την μηχανική χρηστικότητα είναι ότι προϋποθέτει γνώση των ενεργειών του χρήστη καθώς και γνώση "του καλού και του κακού". Για ο παράδειγμα της αναίρεσης προγραμματισμού του βίντεο, οι παραπάνω προδιαγραφές προϋποθέτουν ότι ξέρουμε ότι ο χρήστης θα κάνει προγραμματιστικά σφάλματα που θα θέλουν αναίρεση, καθώς και ότι είναι καλύτερο η διαδικασία αυτή να έχει όσο το δυνατόν λιγότερα βήματα. Αν και στην συγκεκριμένη περίπτωση αυτά είναι αρκετά προφανή και σύμφωνα με τις γενικές αρχές χρηστικότητας, τα πράγματα δεν είναι πάντοτε τόσο ξεκάθαρα, οπότε χρειάζεται προσοχή.

---

Ερωτήσεις:

1. Αναζητήστε στο Internet αναφορές/ παραδοτέα ανάλυσης απαιτήσεων - καταγραφής προδιαγραφών ερευνητικών και αναπτυξιακών προγραμμάτων.
  2. Αναζητήστε πρότυπα για την Επικοινωνία - Ανθρώπου Μηχανής. Αναζητήστε σε κατασκευαστές λειτουργικών συστημάτων (Microsoft, Apple, Sun) οδηγίες για την ανάπτυξη διεπαφών για τα περιβάλλοντά τους.
  3. Μπορείτε να εφαρμόσετε τη μηχανική χρηστικότητας για τον προγραμματισμό ενός φούρνου με χρονοδιακόπτη;
-

## Κεφάλαιο 8 Σχεδίαση Συστημάτων EAM

Η σχεδίαση συστημάτων EAM ουσιαστικά περιλαμβάνει τρεις ξεχωριστές διαδικασίες: μοντελοποίηση χρηστών, ανάλυση εργασίας, και σχεδίαση διαλόγου. Στόχος της πρώτης είναι η αναπαράσταση της νοητικής και γνωσιακής κατάστασης του χρήστη καθώς επιτελεί τις διάφορες εργασίες που απαιτούνται στο υπό σχεδίαση σύστημα. Στόχος της δεύτερης είναι η αναπαράσταση αυτών καθ' εαυτών των ενεργειών του χρήστη στην διάρκεια αυτών των εργασιών, δηλαδή της παρατηρήσιμης συμπεριφοράς του. Τέλος, στόχος της τρίτης είναι η συγκεκριμενοποίηση της διεπαφής όπως αυτή θα φαίνεται στην οθόνη με τα στοιχεία επικοινωνίας που θα δίνει στον χρήστη. Παρά την σαφή διαφορά που υπονοούν οι ορισμοί των δύο πρώτων διαδικασιών (νοητικά/γνωσιακά βήματα σε αντίθεση με βήματα-ενέργειες), τα μοντέλα, οι τεχνικές, και οι μέθοδοι που χρησιμοποιούνται για αυτά είναι παρόμοια και υπάρχει εν γένει μία μεγάλη ασάφεια όσον αφορά το πού σταματά η μία διαδικασία και πού αρχίζει η άλλη. Για τον λόγο αυτό επικεντρωνόμαστε μόνο σε δύο μοντέλα που κατά βάση προτείνονται για μοντελοποίηση χρηστών. Από την φύση τους, τα μοντέλα αυτά μπορούν κάλλιστα να χρησιμοποιηθούν και για ανάλυση εργασίας με κάποιες διαφοροποιήσεις στην έμφαση [Dix, σελ 262].

### 8.1 Μοντελοποίηση Χρηστών – Το Μοντέλο GOMS

Το μοντέλο GOMS προτάθηκε από τους Card, Moran και Newell και είναι ακρόνυμο για το Goals, Operators, Methods, Selection Rules (Στόχοι, Τελεστές, Μέθοδοι, Κανόνες Επιλογής) [Card et al., 1983]. Καταρχήν θα ορίσουμε τα στοιχεία που αποτελούν το GOMS:

*Στόχοι:* Οι στόχοι περιγράφουν αυτό που θέλει να επιτύχει ο χρήστης. Εάν, για παράδειγμα, ο χρήστης θέλει να ελαχιστοποιήσει ένα παράθυρο σε κάποιο παραθυρικό περιβάλλον, ο στόχος που θέτει είναι ΕΛΑΧΙΣΤΟΠΟΙΗΣΗ\_ΠΑΡΑΘΥΡΟΥ. Οι στόχοι συχνά αναλύονται σε υπο-στόχους. Για παράδειγμα, εάν ο χρήστης επιθυμεί να φωτοτυπήσει ένα άρθρο, ο στόχος θα είναι ΦΩΤΟΤΥΠΗΣΗ\_ΑΡΘΡΟΥ και μπορεί να αναλυθεί σε υπο-στόχους ΑΝΕΥΡΕΣΗ\_ΑΡΘΡΟΥ, ΦΩΤΟΤΥΠΗΣΗ\_ΣΕΛΙΔΑΣ, ΣΥΛΛΟΓΗ\_ΑΝΤΙΓΡΑΦΟΥ, ΣΥΛΛΟΓΗ\_ΠΡΩΤΟΤΥΠΟΥ. Συχνά οι στόχοι και οι υπο-στόχοι παρουσιάζονται ιεραρχικά, μια αυστηρά ιεραρχική δομή δεν είναι όμως αναγκαία. Κάποιες εκδόσεις του GOMS επιτρέπουν την ταυτόχρονη ενεργοποίηση/ εκτέλεση περισσότερων τους ενός στόχων, ενώ άλλες εκδόσεις χρησιμοποιούν μια επίπεδη δομή που δεν εμφανίζει κάποια ιεραρχία.

*Τελεστές:* Ο τελεστής είναι μια ενέργεια που εκτελείται προς επίτευξη κάποιου στόχου. Οι τελεστές μπορεί να είναι ενέργειες που έχουν σχέση με την αντίληψη, τη σκέψη (γνωσιακές ενέργειες), απλές κινήσεις (πιθανώς και μηχανικές) ή και συνδυασμός των παραπάνω. Οι τελεστές μπορούν να τροποποιήσουν τη νοητική κατάσταση του χρήστη ή την κατάσταση του εξωτερικού περιβάλλοντος. Ο χρόνος που απαιτείται για την εκτέλεση μιας ενέργειας θεωρείται ανεξάρτητος του ιστορικού, δηλαδή των ενεργειών που έχουν προηγουμένως εκτελεστεί. Ο χρόνος εκτέλεσης μπορεί να προσδιορισθεί κατά προσέγγιση από κάποια σταθερά (π.χ. ο μέσος χρόνος που χρειάζεται ένα έμπειρος χρήστης για να πιάσει ένα πλήκτρο), μια στατιστική κατανομή, ή από μια συνάρτηση (π.χ. κάποια συνάρτηση που συσχετίζει τη θέση του πλήκτρου με το χρόνο που χρειάζεται ο χρήστης για να το επιλέξει). Από τα παραπάνω συνεπάγεται ότι η ακρίβεια πρόβλεψης του χρόνου εκτέλεσης του μοντέλου GOMS, σχετίζεται άμεσα με την ακρίβεια με την οποία προσεγγίζεται ο χρόνο εκτέλεσης ενός τελεστή.

*Μέθοδοι:* Οι μέθοδοι είναι ακολουθίες τελεστών και κλήσεων σε υπο-στόχους, οι οποίες επιτυγχάνουν ένα στόχο. Εάν οι στόχοι έχουν ιεραρχική μορφή τότε υπάρχει μια αντίστοιχη ιεραρχία μεθόδων. Το περιεχόμενο των μεθόδων εξαρτάται από το σύνολο των πιθανών τελεστών καθώς και από τη φύση των εργασιών που αναπαριστώνται. Για παράδειγμα, αν σε κάποιο παραθυρικό σύστημα μπορεί κανείς να "κλείσει" ένα παράθυρο υπό μορφή εικονιδίου είτε μέσω

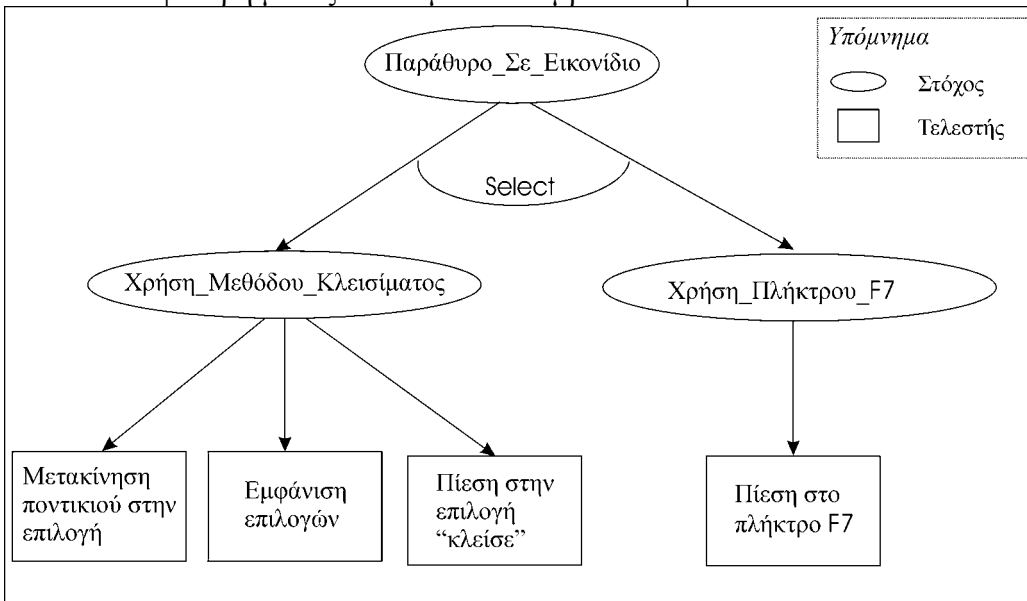
κάποιας εντολής σε κάποιον επιλογή είτε με απλή πίεση του πλήκτρου F7, τότε σε μία αναπαράσταση GOMS, για τον κοινό στόχο Παράθυρο-σε-Εικονίδιο θα έχουμε δύο μεθόδους:

```

GOAL: Παράθυρο_Σε_Εικονίδιο
.   [select GOAL: Χρήση_Μεθόδου_Κλεισίματος
.       . Μετακίνηση ποντικιού στην επιλογή
.       . Εμφάνιση επιλογών
.       . Πίεση στην επιλογή "κλείσε"
GOAL: Χρήση_Πλήκτρου_F7
.       . Πίεση στο πλήκτρο F7]

```

Οι κουκκίδες χρησιμεύουν σαν ενδείξεις των ιεραρχικών επιπέδων των στόχων. Η αντίστοιχη περιγραφή σε δενδρική μορφή φαίνεται παρακάτω, όπου η διαφορά μεταξύ στόχων και τελεστών είναι πλέον φανερή μεταξύ ενδιάμεσων κόμβων και φύλλων.



**Κανόνες επιλογής:** Συχνά υπάρχουν περισσότερες από μια μέθοδοι που μπορούν να χρησιμοποιηθούν για την επίτευξη ενός στόχου (όπως φαίνεται και στο παραπάνω παράδειγμα). Σε μια τέτοια περίπτωση χρειάζονται κανόνες επιλογής για την αναπαράσταση της γνώσης του χρήστη σχετικά με την προτιμότερη μέθοδο. Συνήθως τέτοιοι κανόνες βασίζονται σε συγκεκριμένες ιδιότητες της εργασίας και δημιουργούνται από την εμπειρία ή την εκπαίδευση του χρήστη. Στο προηγούμενο παράδειγμα ο χρήστης Χάρης μπορεί να έχει τους παρακάτω κανόνες:

Χρήστης = Χάρης

Κανόνας 1: ΧΡΗΣΗ\_ΜΕΘΟΔΟΥ\_ΚΛΕΙΣΙΜΑΤΟΣ, όταν δεν ισχύει άλλος κανόνας

Κανόνας 2: ΧΡΗΣΗ\_ΠΛΗΚΤΡΟΥ\_F7, όταν η εφαρμογή είναι το DOOM.

**Σχέση Στόχων και Τελεστών:** Επίπεδο λεπτομέρειας. Η διαφορά μεταξύ των στόχων και των τελεστών έγκειται στο επίπεδο της λεπτομέρειας στο οποίο θέλει να φτάσει όποιος εκτελεί την ανάλυση. Όπως φαίνεται και στο προηγούμενο παράδειγμα, σε ένα στόχο αντιστοιχεί τουλάχιστον μια μέθοδος που χρησιμοποιεί τελεστές για να δηλώσει τον τρόπο επίτευξής του. Αντίθετα οι τελεστές δε χρειάζεται (σύμφωνα με αυτόν που εκτελεί την ανάλυση) να αναλυθούν περισσότερο. Το επίπεδο στο οποίο θα σταματήσει η ανάλυση (οπότε και έχουμε φτάσει σε τελεστές) είναι αυτό που οι χρόνοι εκτέλεσης είναι ανεξάρτητων των συμφραζομένων (context-free) και μπορούν να προσδιορισθούν άμεσα για κάθε τελεστή. Γι' αυτό το λόγο τα μοντέλα GOMS σταματούν συνήθως στο επίπεδο εκτέλεσης εντολής ή πίεσης πλήκτρου. Παρ' όλα αυτά δεν είναι αναγκαίο να φτάσει όλη η ανάλυση στο ίδιο επίπεδο λεπτομέρειας. Ο αναλυτής μπορεί να δώσει έμφαση σε συγκεκριμένα σημεία που τον ενδιαφέρουν και σε άλλα σημεία να σταματήσει πολύ υψηλότερα.



## Τεχνικές ανάλυσης βασισμένες στο μοντέλο GOMS

Μετά τη δημοσίευση του βιβλίου των Card, Moran και Newell "The Psychology of Human Computer Interaction (Η Ψυχολογία της Αλληλεπίδρασης Ανθρώπου Υπολογιστή) [Card et al., 1983], το μοντέλο GOMS έγινε μια από τις λίγες θεωρητικές έννοιες στην ΕΑΜ που είναι ευρύτατα γνωστές. Σήμερα υπάρχουν αρκετές διαφορετικές τεχνικές ανάλυσης που βασίζονται στο μοντέλο και αρκετές εφαρμογές τους σε πραγματικές συνθήκες σχεδιασμού [John & Kierras, 1996a]. Στη συνέχεια θα παρουσιάσουμε δύο από αυτές: την αρχική έκδοση του μοντέλου GOMS όπως παρουσιάστηκε από τους Card, Moran και Newell και την απλούστερη από τις τεχνικές που βασίζονται στο μοντέλο GOMS, το Keystroke Level Model - Μοντέλο Επιπέδου Πληκτρολόγησης (KLM). Άλλες τεχνικές που χρησιμοποιούνται, αλλά δε θα παρουσιαστούν στα πλαίσια αυτού του βιβλίου, είναι μια πιο αναλυτική έκδοση που ονομάζεται NGOMSL [Kierras, 1988, 1996] και μια έκδοση που μπορεί να μοντελοποιήσει επικαλυπτόμενες ανθρώπινες δραστηριότητες η CPM-GOMS η οποία χρησιμοποιεί διαγράμματα PERT για την αναπαράσταση τελεστών και των μεταξύ τους συσχετίσεων.

### Μορφή ενός μοντέλου GOMS

Ανάλογα με την τεχνική που θα χρησιμοποιηθεί διαφέρει και ο τρόπος αναπαράστασης. Δύο είναι οι βασικές μορφές: αυτή του *προγράμματος* και η *σειριακή μορφή*. Στη μορφή προγράμματος οι μέθοδοι δέχονται ένα σύνολο από αποδεκτές τιμές και εκτελούν το αντίστοιχο στιγμιότυπο (instance) της εργασίας. Το πλεονέκτημα αυτής της μορφής είναι η διαδικαστική αναπαράσταση που μπορεί να βοηθήσει στη γρήγορη υλοποίηση του αντίστοιχου προγράμματος. Τα μειονεκτήματα αυτής της αναπαράστασης είναι ότι απαιτείται η εκτέλεση του μοντέλου για να αντιληφθεί κανείς τη σειρά εκτέλεσης των τελεστών, καθώς και το γεγονός ότι μια πλήρης αναπαράσταση μπορεί να απαιτήσει ιδιαίτερο κόπο, ειδικά εάν πρόκειται για μορφή άμεσα εκτελέσιμη από υπολογιστή. Σε αντίθεση με τη μορφή προγράμματος, οι μέθοδοι στη σειριακή μορφή του μοντέλου GOMS περιέχουν μια σταθερή ακολουθία τελεστών για την επίτευξη ενός στόχου. Τα πλεονεκτήματα και τα μειονεκτήματα της σειριακής μορφής είναι τα αντίστροφα από αυτά της μορφής προγράμματος. Η ακολουθία των τελεστών για την επίτευξη ενός στόχου είναι άμεσα εμφανής, όμως μπορεί να υπάρχει πληροφορία για τη δομή των μεθόδων η οποία δεν είναι δυνατό να αποτυπωθεί. Και πάλι εάν και η αποτύπωση είναι σε σειριακή μορφή είναι αρκετά απλή, η δημιουργία και η πρόβλεψη για μεγάλο αριθμό εργασιών μπορεί να αποδειχτεί χρονοβόρα και κοπιαστική εργασία.

### Μοντέλο GOMS των Card, Moran και Newell

[Card et al., 1983, Ch.5, Card et al., 1980b]

Η μορφή του μοντέλου GOMS που προτάθηκε από τους Card, Moran και Newell (το ονομάζουμε CMN-GOMS για να το διαχωρίσουμε από τις άλλες εκδόσεις του μοντέλου), έχει μια αυστηρή ιεραρχία στόχων. Οι μέθοδοι παρουσιάζονται σε μορφή προγράμματος και μπορούν να έχουν υπο-μεθόδους και συνθήκες. Με αυτό τον τρόπο το μοντέλο μπορεί να προβλέψει και την ακολουθία τελεστών και το χρόνο εκτέλεσης.

Το μοντέλο στηρίζεται στο Model Human Processor (MHP). Συγκεκριμένα, το GOMS όπως αναφέρεται στο John and Kieras GOMS Analysis of UI Techniques, βασίζεται σε δύο από τις Αρχές Λειτουργίας (Principles of Operation) του MHP την Αρχή του Χώρου Προβλήματος (Problem Space Principle) και την Αρχή της Λογικότητας (Rationality Principle). Η Αρχή του Χώρου Προβλήματος θεωρεί ότι η δραστηριότητα του χρήστη μπορεί να χαρακτηριστεί ως η εφαρμογή μιας σειράς από ενέργειες που ονομάζονται *τελεστές*, για να μετασχηματίσει μια αρχική κατάσταση σε μια κατάσταση *στόχο*. Με την απόκτηση εμπειρίας, η σειρά των τελεστών παγιώνεται ως *μέθοδος* και μπορεί να ανακληθεί χωρίς να απαιτείται η δημιουργία της. Η Αρχή της Λογικότητας θεωρεί ότι οι χρήστες θα αναπτύξουν μεθόδους που είναι αποτελεσματικές, ανάλογα πάντα με τη δομή του περιβάλλοντος και τις ικανότητες του χρήστη. Με αυτό τον τρόπο η

επικοινωνία του ανθρώπου με τον υπολογιστή μπορεί να θεωρηθεί ότι είναι η εκτέλεση μεθόδων για την επίτευξη στόχων και επειδή οι άνθρωποι προσπαθούν να είναι αποτελεσματικοί, η αποτελεσματικότητα των μεθόδων θα εξαρτάται κατά κύριο λόγο από την ευχρηστία του σχεδιασμού. Δηλαδή, κρίνοντας το σχεδιασμό μπορούμε να προβλέψουμε κρίσιμες πτυχές της επικοινωνίας του χρήστη με τον υπολογιστή.

#### Παράδειγμα ανάλυσης GOMS

Έχοντας ορίσει τα βασικά στοιχεία του μοντέλου GOMS και το πώς αυτά συσχετίζονται, θα δώσουμε ένα μικρό παράδειγμα μοντελοποίησης χρήστη. Γι' αυτό το σκοπό θα χρησιμοποιήσουμε το αρχικό μοντέλο GOMS όπως προτάθηκε από τους Card, Moran και Newell. Το παράδειγμα πραγματεύεται την χρήση φωτοτυπικού μηχανήματος για την φωτοτύπηση ενός άρθρου από κάποιο περιοδικό [Δείτε και Dix, σελ 235]:

#### GOAL: Φωτοτύπηση\_Άρθρου

```

.      GOAL:  Ανεύρεση_Άρθρου
.      GOAL:  Φωτοτύπηση_Σελίδας επανάληψη μέχρι να τελειώσουν οι σελίδες
.      .      GOAL:  Προσανατολισμός_Σελίδας
.      .      .      Άνοιγμα_Καλύματος_Κρυστάλλου
.      .      .      Επιλογή_Σελίδας
.      .      .      Τοποθέτηση_Σελίδας
.      .      .      Κλείσιμο_Καλύματος
.      .      GOAL:  Επιβεβαίωση_Αντιγράφου
.      .      .      Εύρεση_Θέση_Δημιουργούμενων_Αντιγράφων
.      .      .      Εξέταση_Αντιγράφου
.      GOAL:  Συλλογή_Αντιγράφου
.      .      Εύρεση_Θέσης_Δημιουργούμενων_Αντιγράφων
.      .      Παραλαβή_Αντιγράφου (ο βασικός στόχος ικανοποιήθηκε!)
.      GOAL:  Συλλογή_Περιοδικού
.      .      Άνοιγμα_Καλύματος_Κρυστάλλου
.      .      Παραλαβή_Περιοδικού
.      .      Κλείσιμο_Καλύματος_Κρυστάλλου

```

Ένα σημαντικό χαρακτηριστικό του μοντέλου GOMS είναι ότι η ανάλυση της δομής των στόχων του δίνει κάποια μέτρα απόδοσης του υπό σχεδίαση συστήματος

- Το βάθος της στοίβας μπορεί να χρησιμοποιηθεί για να εκτιμήσει τις απαιτήσεις σε βραχυπρόθεσμη μνήμη.
- Μετά την υλοποίηση του συστήματος, οι κανόνες επιλογής του μοντέλου μπορούν να αντιπαραβληθούν με πραγματικές ακολουθίες ενεργειών χρηστών για διαφοροποιήσεις.

Σε αρχικά πειράματα έχει επιτευχθεί 90% ακρίβεια στην πρόβλεψη των εντολών των χρηστών.

Ένα άλλο σημαντικό όφελος που προσφέρει το GOMS είναι ότι μπορεί να χρησιμοποιηθεί ώστε να βρεθούν σημεία πιθανού σφάλματος του χρήστη, και να επιβληθούν συγκεκριμένοι τρόποι υλοποίησης του συστήματος ώστε αυτά να αποφευχθούν. Κλασσικό παράδειγμα αποτελεί το θέμα της πρόωρης "ολοκλήρωσης του στόχου" (closure). Σε μια τέτοια περίπτωση ο στόχος του χρήστη επιτυγχάνεται πριν την ολοκλήρωση όλων των επιμέρους ενεργειών, το οποίο εγκυμονεί κινδύνους κάποιες φορές. Κάτι τέτοιο συμβαίνει και στο παραπάνω παράδειγμα της φωτοτύπησης άρθρου. Ο στόχος είναι η δημιουργία και παραλαβή ενός φωτοαντίγραφου του πρωτότυπου. Στο παραπάνω μοντέλο ο στόχος αυτό επιτυγχάνεται όταν παίρνουμε το αντίγραφο και πριν πάρουμε το πρωτότυπο από το φωτοτυπικό, με κίνδυνο το τελευταίο να ξεχαστεί. Η λύση είναι το μοντέλο να επιβάλει την αντίστροφη διάταξη των στόχων, δηλαδή ο στόχος Συλλογή\_Αντιγράφου να έρχεται μετά από τον στόχο Συλλογή\_Περιοδικού, αλλά πώς μπορεί να γίνει αυτό σε ένα φωτοτυπικό;

Παρόμοιο πρόβλημα υπήρχε και με τα αρχικά μηχανήματα ανάληψης χρημάτων της τράπεζας. Σε μερικές τράπεζες το μηχάνημα έδινε πρώτα τα χρήματα και μετά επέστρεφε την κάρτα, με αποτέλεσμα πολλοί χρήστες του συστήματος να ξεχνούν τις πιστωτικές τους κάρτες στο μηχάνημα. Το αντίστοιχο μοντέλο GOMS σε απλουστευμένη μορφή ήταν όπως παρακάτω:

GOAL: Ανάλυση\_Χρημάτων  
. GOAL: Χρήση\_Αυτόματου\_Μηχανήματος\_Συναλλαγής  
. . Τοποθέτηση\_Κάρτας  
. . Εισαγωγή\_Προσωπικού\_Κωδικού  
. . Εισαγωγή\_Ποσού\_Ανάληψης  
. . Ανάλυση\_Χρημάτων  
(ο βασικός στόχος ικανοποιήθηκε – κίνδυνος να ξεχαστεί η κάρτα!)  
. . Παραλαβή\_Κάρτας

Το πρόβλημα μπορεί να επιλυθεί (και όντως έχει επιλυθεί στις πιο σύγχρονες εκδόσεις αυτών των μηχανημάτων) με αντιστροφή των δύο τελευταίων βημάτων, δηλαδή να πρέπει ο χρήστης να πάρει πρώτα την κάρτα και μετά τα χρήματα. Κάτι τέτοιο είναι δυνατό στα μηχανήματα αυτά, διότι η κάρτα είναι στον έλεγχο του συστήματος όσο διάστημα γίνεται η συνδιαλλαγή, οπότε υπάρχει έλεγχος στο πότε απελευθερώνεται. Αλλά κάτι τέτοιο δεν ισχύει για το πρωτότυπο κάποιας φωτοτύπησης σε ένα φωτοτυπικό μηχάνημα!

Παράδειγμα 2ο - Διαχείριση αρχείων

Παρουσιάζουμε ένα παράδειγμα ανάλυσης με χρήση του μοντέλου GOMS. Οι εργασίες που θα αναλυθούν είναι η διαγραφή, η μεταφορά και η αντιγραφή ενός αρχείου, η διαγραφή, η μεταφορά και η αντιγραφή ενός καταλόγου. Ως περιβάλλοντα εργασίας θα θεωρηθούν το MS-DOS και τα MS Windows με το Windows Explorer.

### Μοντέλο GOMS για το MS-DOS (έκδοση 5.0)

GOAL: διαγραφή\_αρχείου

Ανάκτηση ονόματος εντολής (del)  
Ανάκτηση ονόματος αρχείου  
Ανάκτηση διαδρομής καταλόγου και δημιουργία πρώτης παραμέτρου εντολής  
Πληκτρολόγηση εντολής  
Επιβεβαίωση διαγραφής

GOAL: μεταφορά αρχείου

GOAL: αντιγραφή αρχείου σε νέα θέση  
Ανάκτηση ονόματος εντολής (copy)  
Ανάκτηση ονόματος αρχείου  
Ανάκτηση διαδρομής αρχικού καταλόγου και δημιουργία πρώτης παραμέτρου εντολής  
Σκέψη για το όνομα του αρχείου προορισμού  
Ανάκτηση διαδρομής καταλόγου προορισμού και δημιουργία δεύτερης παραμέτρου εντολής  
Πληκτρολόγηση εντολής  
Επιβεβαίωση αντιγραφής  
GOAL: διαγραφή αρχικού αρχείου  
Ανάκτηση ονόματος εντολής (del)  
Ανάκτηση ονόματος αρχείου

Ανάκτηση διαδρομής καταλόγου και δημιουργία πρώτης παραμέτρου εντολής  
 Πληκτρολόγηση εντολής  
 Επιβεβαίωση διαγραφής

GOAL: αντιγραφή αρχείου

Ανάκτηση ονόματος εντολής (copy)

Ανάκτηση ονόματος αρχείου

Ανάκτηση διαδρομής αρχικού καταλόγου και δημιουργία πρώτης παραμέτρου εντολής

Σκέψη για το όνομα του αρχείου προορισμού

Ανάκτηση διαδρομής καταλόγου προορισμού και δημιουργία δεύτερης παραμέτρου εντολής

Πληκτρολόγηση εντολής

Επιβεβαίωση αντιγραφής

GOAL: διαγραφή καταλόγου *επανάλαβε μέχρι να μην υπάρχει υποκατάλογος*

Ανάκτηση ονόματος εντολής (rmdir)

Ανάκτηση ονόματος καταλόγου και δημιουργία πρώτης παραμέτρου εντολής

GOAL: διαγραφή αρχείων καταλόγου

Ανάκτηση ονόματος εντολής (del)

Ανάκτηση ονόματος αρχείου

Ανάκτηση διαδρομής καταλόγου και δημιουργία πρώτης παραμέτρου εντολής

εντολής

Πληκτρολόγηση εντολής

Επιβεβαίωση διαγραφής

Πληκτρολόγηση εντολής

GOAL: μεταφορά καταλόγου

GOAL: αντιγραφή καταλόγου σε νέα θέση

GOAL: διαγραφή αρχικού καταλόγου

GOAL: αντιγραφή καταλόγου

GOAL: δημιουργία νέου καταλόγου

GOAL: αντιγραφή αρχείων στο νέο κατάλογο

GOAL: αντιγραφή (υπο)καταλόγων αρχικού καταλόγου

GOAL: δημιουργία καταλόγου

Ανάκτηση ονόματος εντολής (mkdir)

Ανάκτηση ονόματος καταλόγου και δημιουργία πρώτης παραμέτρου εντολής

Πληκτρολόγηση εντολής

### Μοντέλο GOMS για τα MS Windows με Windows Explorer

Υπάρχει ένα σύνολο από διαφορετικές μεθόδους για την επίτευξη των στόχων που έχουμε θέσει. Θα περιγράψουμε δύο εναλλακτικές προσεγγίσεις για κάθε περίπτωση. Η πρώτη προσέγγιση χρησιμοποιεί τεχνικές απευθείας διαχείρισης (direct manipulation techniques), ενώ η δεύτερη μέθοδος χρησιμοποιεί τους αναδυόμενους επιλογείς (pop up menus). Με δεδομένο ότι οι μέθοδοι παραμένουν ίδιες είτε πρόκειται για αρχείο είτε πρόκειται για κατάλογο, μπορούμε να περιγράψουμε γενικότερες μεθόδους οι οποίες εφαρμόζονται σε κάθε αντικείμενο του περιβάλλοντος εργασίας του χρήστη.

## Πρώτη προσέγγιση: τεχνικές απευθείας διαχείρισης

GOAL: διαγραφή αντικειμένου

GOAL: μεταφορά αρχείου στο καλάθι αχρήστων (waste basket)

Επιβεβαίωση διαγραφής

GOAL: μεταφορά αντικειμένου

GOAL: μεταφορά αντικειμένου στον ίδιο τομέα (partition)

Εύρεση εικονιδίου αντικειμένου στην οθόνη

Εύρεση εικονιδίου προορισμού στην οθόνη

Μετακίνηση δείκτη ποντικιού πάνω από το εικονίδιο

Πίεση και διατήρηση πιεσμένου αριστερού πλήκτρου ποντικιού

Μετακίνηση δείκτη ποντικιού στο εικονίδιο προορισμού

Επιβεβαίωση αντιστροφής χρωμάτων εικονιδίου προορισμού

Απελευθέρωση αριστερού πλήκτρου ποντικιού

Επιβεβαίωση μεταφοράς

GOAL: μεταφορά αντικειμένου σε διαφορετικό τομέα (partition)

Εύρεση εικονιδίου αντικειμένου στην οθόνη

Εύρεση εικονιδίου προορισμού στην οθόνη

Μετακίνηση δείκτη ποντικιού πάνω από το εικονίδιο

Πίεση και διατήρηση πιεσμένου αριστερού πλήκτρου ποντικιού

Πίεση πλήκτρου SHIFT του πληκτρολογίου

Μετακίνηση δείκτη ποντικιού στο εικονίδιο προορισμού

Επιβεβαίωση αντιστροφής χρωμάτων εικονιδίου προορισμού

Απελευθέρωση αριστερού πλήκτρου ποντικιού

Επιβεβαίωση μεταφοράς

GOAL: αντιγραφή αντικειμένου

GOAL: αντιγραφή αντικειμένου στον ίδιο τομέα (partition)

Εύρεση εικονιδίου αντικειμένου στην οθόνη

Εύρεση εικονιδίου προορισμού στην οθόνη

Μετακίνηση δείκτη ποντικιού πάνω από το εικονίδιο

Πίεση και διατήρηση πιεσμένου αριστερού πλήκτρου ποντικιού

Πίεση πλήκτρου CTRL του πληκτρολογίου

Μετακίνηση δείκτη ποντικιού στο εικονίδιο προορισμού

Επιβεβαίωση αντιστροφής χρωμάτων εικονιδίου προορισμού

Απελευθέρωση αριστερού πλήκτρου ποντικιού

Επιβεβαίωση αντιγραφής

GOAL: αντιγραφή αντικειμένου σε διαφορετικό τομέα (partition)

Εύρεση εικονιδίου αντικειμένου στην οθόνη

Εύρεση εικονιδίου προορισμού στην οθόνη

Μετακίνηση δείκτη ποντικιού πάνω από το εικονίδιο

Πίεση και διατήρηση πιεσμένου αριστερού πλήκτρου ποντικιού

Μετακίνηση δείκτη ποντικιού στο εικονίδιο προορισμού

Επιβεβαίωση αντιστροφής χρωμάτων εικονιδίου προορισμού

Απελευθέρωση αριστερού πλήκτρου ποντικιού

## Επιβεβαίωση αντιγραφής

**Δεύτερη προσέγγιση: χρήση αναδυόμενων επιλογέων**

GOAL: διαγραφή αντικειμένου

- Εύρεση εικονιδίου αντικειμένου στην οθόνη
- Εύρεση εικονιδίου προορισμού στην οθόνη
- Μετακίνηση δείκτη ποντικού πάνω από το εικονίδιο
- Πίεση δεξιού πλήκτρου ποντικού
- Επιλογή διαγραφής από τον αναδυόμενο επιλογέα
- Επιβεβαίωση διαγραφής

GOAL: μεταφορά αντικειμένου

- Εύρεση εικονιδίου αντικειμένου στην οθόνη
- Εύρεση εικονιδίου προορισμού στην οθόνη
- Μετακίνηση δείκτη ποντικού πάνω από το εικονίδιο
- Πίεση δεξιού πλήκτρου ποντικού
- Επιλογή αποκοπής από τον αναδυόμενο επιλογέα
- Μετακίνηση δείκτη ποντικού στο εικονίδιο προορισμού
- Επιβεβαίωση αντιστροφής χρωμάτων εικονιδίου προορισμού
- Πίεση δεξιού πλήκτρου ποντικού
- Επιλογή επικόλλησης από τον αναδυόμενο επιλογέα
- Επιβεβαίωση μεταφοράς

GOAL: αντιγραφή αντικειμένου

- Εύρεση εικονιδίου αντικειμένου στην οθόνη
- Εύρεση εικονιδίου προορισμού στην οθόνη
- Μετακίνηση δείκτη ποντικού πάνω από το εικονίδιο
- Πίεση δεξιού πλήκτρου ποντικού
- Επιλογή αντιγραφής από τον αναδυόμενο επιλογέα
- Μετακίνηση δείκτη ποντικού στο εικονίδιο προορισμού
- Επιβεβαίωση αντιστροφής χρωμάτων εικονιδίου προορισμού
- Πίεση δεξιού πλήκτρου ποντικού
- Επιλογή επικόλλησης από τον αναδυόμενο επιλογέα
- Επιβεβαίωση αντιγραφής

**Σχόλια**

Όπως φαίνεται και από την ανάλυση με τη βοήθεια του μοντέλου GOMS η εργασίες απλοποιούνται κατά πολύ στο γραφικό περιβάλλον σε σχέση με το περιβάλλον γραμμής εντολής. Η ασυνέπεια που παρατηρείται στις μεθόδους απευθείας διαχείρισης (διαφορετικές μέθοδοι αν το αντικείμενο προορισμού βρίσκεται στον ίδιο ή σε διαφορετικό τομέα) αυξάνει την πολυπλοκότητα. Γι' αυτό το λόγο απλούστερη (αριθμός μεθόδων – βημάτων) αναδεικνύεται η χρήση αναδυόμενων επιλογέων.

**Άσκηση**

Παρατηρήστε πώς υλοποιούνται στο Macintosh οι αντίστοιχες εργασίες.

## Μοντέλο KLM

Το μοντέλο KLM είναι ακρόνυμο του Keystroke Level Model (Μοντέλο Επιπέδου Πληκτρολόγησης) και αποτελεί την απλούστερη τεχνική ανάλυσης που βασίζεται στο μοντέλο GOMS [Card et al. 1980a]. Ο αναλυτής χρησιμοποιεί το παραθέτοντας την ακολουθία των τελεστών και στο τέλος αθροίζει τους χρόνους εκτέλεσης κάθε τελεστή. Ο αναλυτής επιλέγει τη μέθοδο που θα ακολουθηθεί για την επίτευξη ενός στόχου. Το μοντέλο παρουσιάζεται σε σειριακή μορφή και περιλαμβάνει μόνο βασικούς τελεστές σε επίπεδο πληκτρολόγησης.

Υπάρχουν διαφορετικές εκδόσεις του μοντέλου KLM που επεκτείνουν τη λειτουργικότητά του προσθέτοντας νέους τελεστές και κανόνες για την περιγραφή των εργασιών. Η έκδοση που περιγράφεται περιλαμβάνει επτά τύπους τελεστών που παρουσιάζονται στον επόμενο πίνακα.

Τελεστής	Ενέργεια χρήστη
K	Πίεση πλήκτρου πληκτρολογίου
B	Πίεση πλήκτρου ποντικιού
P	Μετακίνηση δείκτη ποντικιού
H	Εναλλαγή του χεριού μεταξύ πληκτρολογίου και του ποντικιού
D	Σχεδιασμός γραμμών με το ποντίκι
M	Νοητική προετοιμασία (σκέψη) πριν τη εκτέλεση μιας φυσικής κίνησης
R	Αντίδραση του συστήματος, η οποία μπορεί να αγνοηθεί εάν ο χρήστης δεν χρειάζεται να περιμένει, όπως κατά τη διάρκεια της πληκτρολόγησης

Σε καθένα από αυτούς τους τελεστές αντιστοιχεί ένας χρόνος εκτέλεσης που μπορεί να είναι μια συγκεκριμένη τιμή ή μια παραμετροποιημένη συνάρτηση (για παράδειγμα η τιμή του K εξαρτάται από την ταχύτητα πληκτρολόγησης, το εάν πρόκειται για πλήκτρο πληκτρολογίου ή ποντικιού, αν έχουμε απλή πίεση, απελευθέρωση, κλπ.). Το μοντέλο περιλαμβάνει και κάποιους βασικούς κανόνες που έχουν κατά καιρούς εμπλουτισθεί, όπως η τοποθέτηση του τελεστή M πριν από κάθε τελεστή K (εκτός αν πρόκειται για ακολουθία τελεστών K) ή πριν από P που εκτελείται για την επιλογή μιας εντολής.

Για κάθε έναν τελεστή υπάρχουν τυπικοί χρόνοι (όπως έχουν πειραματικά μετρηθεί), οι οποίοι συμβολίζονται με  $T_K$ ,  $T_B$ ,  $T_P$ ,  $T_H$ ,  $T_D$ ,  $T_M$ , και  $T_R$ , αντίστοιχα. Ο ακόλουθος πίνακας αναφέρει τέτοιους χρόνους.

Τελεστής	Περιγραφή/Παρατηρήσεις	Χρόνος (δ/λεπτα)
K	πίεση πλήκτρου	
	πολύ καλός δακτυλογράφος (90 λέξεις το λεπτό)	0,12
	μέτριος δακτυλογράφος (40 λέξεις το λεπτό)	0,28
	μη δακτυλογράφος	1,20
B	πίεση πλήκτρου ποντικιού	
	πάτημα ή απελευθέρωση του πλήκτρου	0,10
	πάτημα και απελευθέρωση	0,20
P	μετακίνηση δείκτη ποντικιού (νόμος του Fitts = $0,1 \log_2(D/S + 0,5)$ ) συνήθης κίνηση	1,10
H	εναλλαγή του χεριού μεταξύ πληκτρολογίου και του ποντικιού	0,40
D	σχεδιασμός γραμμών με το ποντίκι – ανά συγκεκριμένη περίπτωση	-

M	νοητική προετοιμασία πριν την εκτέλεση μιας φυσικής κίνησης	1,35
R	η αντίδραση του συστήματος – ανά συγκεκριμένη περίπτωση	-

Σαν παράδειγμα εφαρμογής του μοντέλου KLM, σκεφτείτε μια εργασία με τα ακόλουθα στάδια:

1. ο χρήστης επιλέγει ένα εικονίδιο (από μια ομάδα έξι εικονιδίων που εμφανίζεται στην οθόνη) και εφαρμόζει διπλή πίεση του αριστερού πλήκτρου του ποντικιού πάνω του
2. αυτή η ενέργεια έχει ως αποτέλεσμα την εμφάνιση ενός πλαισίου διαλόγου που περιέχει ένα πεδίο εισαγωγής κειμένου προς το οποίο κινεί ο χρήστης τον κέρσορα του ποντικιού
3. ο χρήστης εισάγει έναν 3-ψήφιο ακέραιο σ' αυτό το πεδίο (τον οποίο έχει προηγούμενα σημειώσει)
4. ο χρήστης πιέζει με το ποντίκι στο πλήκτρο OK ώστε να ολοκληρωθεί η εργασία

Ο χρόνος εκτέλεσης  $T$  είναι το άθροισμα των επιμέρους ενεργειών (θεωρούμε ότι ο χρήστης έχει ήδη το χέρι στο ποντίκι, η διπλή πίεση αντιστοιχεί σε δύο απλές πιέσεις και ότι πρέπει να πιέσει με το δείκτη του ποντικιού στο πεδίο εισαγωγής κειμένου για να γράψει):

$$\begin{aligned}
 T &= T_M + T_P + 2 T_B[\text{Αριστερό Πλήκτρο}] + && \text{στάδιο 1} \\
 &T_P + && \text{στάδιο 2} \\
 &T_B + T_H + T_M + T_K + T_K + T_K + && \text{στάδιο 3} \\
 &T_H + T_M + T_P + T_B[\text{Αριστερό Πλήκτρο}] && \text{στάδιο 4} \\
 &= 3T_M + 3T_P + 4T_B + 2T_H + 3T_K
 \end{aligned}$$

Θεωρώντας ότι ο χρήστης είναι πολύ καλός στην πληκτρολόγηση, έχουμε

$$T = 3 * 1,35 + 3 * 1,10 + 4 * 0,20 + 2 * 0,40 + 3 * 0,12 = 9,31 \text{ δ/λεπτα.}$$

### Σύγκριση CMN-GOMS και KLM

Μια βασική διαφορά είναι η μορφή αναπαράστασης. Το CMN-GOMS χρησιμοποιεί αυτή του προγράμματος οπότε και η παραγόμενη ανάλυση είναι γενική και εκτελέσιμη, ενώ το KLM χρησιμοποιεί τη σειριακή αναπαράσταση. Η κλήση υποστόχων και η επιλογή μεθόδων στο CMN-GOMS προβλέπονται από το μοντέλο, ενώ στο KLM πρέπει να καθορισθούν από τον αναλυτή. Τέλος η ιεραρχία στόχων είναι εμφανής στο CMN-GOMS ενώ υπονοείται στο KLM. Οι τελεστές επιβεβαίωσης του CMN-GOMS αντιστοιχούν στον τελεστή M του KLM. Βέβαια στο μοντέλο KLM οι τελεστές M τοποθετούνται στην αρχή των μεθόδων, ενώ στο CMN-GOMS ο χρόνος για σκέψη τοποθετείται ως τελεστής "Επιβεβαίωση" στο τέλος των μεθόδων.

### Μειονεκτήματα του μοντέλου GOMS

Υπάρχουν βέβαια και κάποια μειονεκτήματα στο μοντέλο GOMS. Ένα βασικό είναι ότι το μοντέλο περιγράφει μόνο το πώς ένας άψογος χρήστης πετυχαίνει κάποιο στόχο. Δεν μπορεί δηλαδή να περιγράψει πιθανά σφάλματα του χρήστη, μια και δεν είναι δυνατό να εκφράσει την μη επίτευξη στόχων. Ένα άλλο είναι ότι εάν χρησιμοποιηθεί μετά την υλοποίηση τότε περιγράφει το συγκεκριμένο μοντέλο χρήσης που ενστερνίστηκε η υλοποίηση (και δεν υπάρχει ελευθερία επιλογής ανάμεσα σε όλα τα πιθανά μοντέλα χρήσης, όπως θα μπορούσε να γίνει κατά τη διάρκεια της σχεδίασης).

Διαβάστε επίσης:

Bonnie E. John and David E. Kierras, Using a GOMS for User Interface Design and Evaluation: Which Technique? ACM Transactions on Computer Human Interaction, Vol. 3, No. 4, Dec. 1996, 287-319



## 8.2 Σχεδίαση Διαλόγου

### Μεθοδολογίες

Οι γλώσσες προγραμματισμού έχουν κατά βάση τρία επίπεδα:

1. λεκτικό
2. συντακτικό
3. σημασιολογικό

Στην EAM, "διάλογος" σημαίνει "δομή" της συνδιαλλαγής μεταξύ του χρήστη και του υπολογιστή. Οπότε "διάλογος" αναφέρεται κυρίως στο συντακτικό επίπεδο (και ίσως λίγο και στο λεκτικό επίπεδο) της γλώσσας συνδιαλλαγής. Η συμβολική έκφραση του διαλόγου είναι σημαντικό κομμάτι του σχεδιασμού ενός συστήματος EAM και γίνεται με διάφορους τρόπους. Πολλοί παίρνουν την μορφή διαγραμμάτων ενώ άλλοι παίρνουν την μορφή δομημένου κειμένου.

Από τους βασισμένους σε διαγράμματα τρόπους σχεδίασης αναφέρουμε τους πιο σημαντικούς:

- *Δίκτυα μετάβασης κατάστασης*: Αυτά είναι τα κλασσικά δίκτυα μετάβασης κατάστασης εφαρμοσμένα στην περιοχή της EAM. Η εφαρμογή τους απαιτεί την αναγνώριση των βασικών καταστάσεων του συστήματος καθώς συνδιαλλάσσεται με τον χρήστη και το πως περνά από την μία κατάσταση στην άλλη. Ένα απλό παράδειγμα δίνεται παρακάτω.
- *Ιεραρχικά δίκτυα μετάβασης κατάστασης*: Αποτελούν γενίκευση των παραπάνω έτσι ώστε συγκεκριμένα υποδίκτυα μετάβασης καταστάσεων διαφόρων επιπέδων λεπτομέρειας να μπορούν να εξετασθούν μεμονωμένα. Φτιάχνεται λοιπόν μία ιεραρχία από δίκτυα, κάποιιοι κόμβοι των οποίων μπορεί να είναι αναδρομικά άλλα δίκτυα. Τέτοιες ιεραρχίες βοηθούν την σχεδίαση, διότι επιτρέπουν την συγκέντρωση της προσοχής του σχεδιαστή σε συγκεκριμένα τμήματα κάθε φορά.

Και οι δύο παραπάνω μεθοδολογίες χωλαίνουν όταν έχουν πολλά παράλληλα τμήματα στον διάλογο (όπως πολλές ανεξάρτητες επιλογές, π.χ., επιλογές για bold, italic, και υπογραμμισμένα γράμματα), καθότι η ακριβής περιγραφή τους κάνει τον αριθμό των απαραίτητων καταστάσεων να αυξάνει εκθετικά. Παρόμοια προβλήματα δημιουργούνται και με το πλήκτρο ESC και την επιλογή HELP, όπου μπορεί κανείς να πάει από οποιαδήποτε σημείο του διαλόγου. Κάποιες λύσεις στο πρόβλημα αυτό δίνονται από τις παρακάτω μεθόδους, και κυρίως την επόμενη.

- *Δίκτυα Petri (Petri Nets)*: Γράφοι ειδικά φτιαγμένοι για να περιγράφουν πολύπλοκα συστήματα που μπορούν να βρίσκονται σε πολλαπλές καταστάσεις ταυτόχρονα, όπως συστήματα hardware, αλλά και συστήματα EAM.
- *Σχέδια καταστάσεων*: Και αυτά είναι ουσιαστικά δίκτυα μετάβασης καταστάσεων, προσανατολισμένα όμως προς την οπτική αναπαράσταση πολύπλοκων δομών. Χαρακτηριστικό πετυχημένο παράδειγμα είναι τα διαγράμματα καταστάσεων Harel.
- *Διαγράμματα ροής*: Αυτά είναι τα κλασσικά διαγράμματα που χρησιμοποιούμε και στον προγραμματισμό. Διαφέρουν από τα δίκτυα μετάβασης καταστάσεων στο ότι οι κόμβοι τους είναι διαδικασίες ή αποφάσεις και όχι καταστάσεις. Ένα πρόβλημα με τα διαγράμματα ροής είναι ότι συνήθως αντικατοπτρίζουν μία προσέγγιση επικεντρωμένη στον προγραμματιστή και όχι στον χρήστη. Για παράδειγμα, χρησιμοποιούν τους ίδιους συμβολισμούς για επικοινωνία του συστήματος είτε με τον χρήστη είτε με ένα αρχείο. Στην EAM είναι καλό στα διαγράμματα ροής να διαφοροποιούνται οι κόμβοι με βάση τις ενέργειες του χρήστη. Ένα απλό παράδειγμα (χωρίς διαφοροποιήσεις) δίνεται παρακάτω.

Από τους βασισμένους σε δομημένο κείμενο τρόπους σχεδίασης διαλόγου αναφέρουμε επιγραμματικά μόνο τους πιο σημαντικούς:

- Γραμματικές διαλόγων: Είναι παρόμοιες με τις γραμματικές γλωσσών προγραμματισμού.
- Κανόνες παραγωγής: Περιγράφουν τον διάλογο με κανόνες της μορφής "if συνθήκη then ενέργεια".
- Άλγεβρες γεγονότων: Χρησιμοποιούν συγκεκριμένα σύνολα τελεστών με συγκεκριμένες ιδιότητες για να περιγράφουν τον διάλογο.

Ανεξαρτήτως της συγκεκριμένης μεθόδου που χρησιμοποιείται για την καταγραφή του σχεδίου διαλόγου, η βιομηχανία εργάζεται πολλές φορές με βάση κάποια *σενάρια χρήσης*. Τα σενάρια χρήσης αποτελούν ουσιαστικά διάφορες ακολουθίες από πράξεις που ο πιθανός χρήστης να ακολουθήσει για να επιτελέσει συνήθη τμήματα της όλης διεργασίας που υλοποιείται κατά την διάρκεια της διάδρασης. Η εστίαση της προσοχής μας πάνω σε αυτά τα σενάρια αποφεύγει την λεπτομερή και μακροσκελή καταγραφή όλων των πιθανών πράξεων που θα μπορούσε ο χρήστης να εκτελέσει, χωρίς να ξεφεύγουμε από τον βασικό σκοπό της διεπαφής, που θα πρέπει να εκφράζεται στα σενάρια χρήσης.

Τέλος ο σχεδιασμός διαλόγου πρέπει να περιλαμβάνει και περιγραφή της οπτικής εμφάνισης της οθόνης όπως αυτή θα παρουσιάζεται στον χρήστη (κατά κάποιο τρόπο, το λεκτικό επίπεδο του διαλόγου). Αυτό στις απλούστερες των περιπτώσεων μπορεί να γίνει με περιγραφή υπό μορφή κειμένου, αλλά στην γενική περίπτωση απαιτεί κάποιο σχέδιο ή κάποια εικόνα που να δείχνει αρκετά παραστατικά τις σχετικές θέσεις των διαφόρων αντικειμένων και την ειδικότερη εμφάνισή τους.

### **Παραδείγματα σχεδιασμού διαλόγων**

#### *Πρόγραμμα πλοήγησης στο διαδίκτυο*

Στην υποενότητα αυτή παρουσιάζουμε σαν παράδειγμα σχεδίασης ένα απλό πρόγραμμα πλοήγησης στο διαδίκτυο. Ακολουθούμε την μεθοδολογία των σεναρίων χρήσης και χρησιμοποιούμε δύο διαγράμματα ροής για την καταγραφή των σεναρίων, και ένα δίκτυο μετάβασης καταστάσεων για την περιγραφή της βασικής δομής του διαλόγου. Στα σενάρια επικεντρωνόμαστε στο χρήστη και περιγράφουμε και καταστάσεις στις οποίες το σύστημα δεν παρουσιάζει μεταβολή.

Το πρώτο σενάριο (Σενάριο Σ1) αντιπροσωπεύει την επιτυχή ανεύρεση μιας σελίδας και συνίσταται στην παρακάτω ακολουθία βημάτων:

Σ1.Βήμα1: Ο χρήστης ξεκινά το πρόγραμμα πλοήγησης

Σ1.Βήμα2: Ο χρήστης αναμένει

Σ1.Βήμα3: Εμφανίζεται η αρχική οθόνη του προγράμματος πλοήγησης

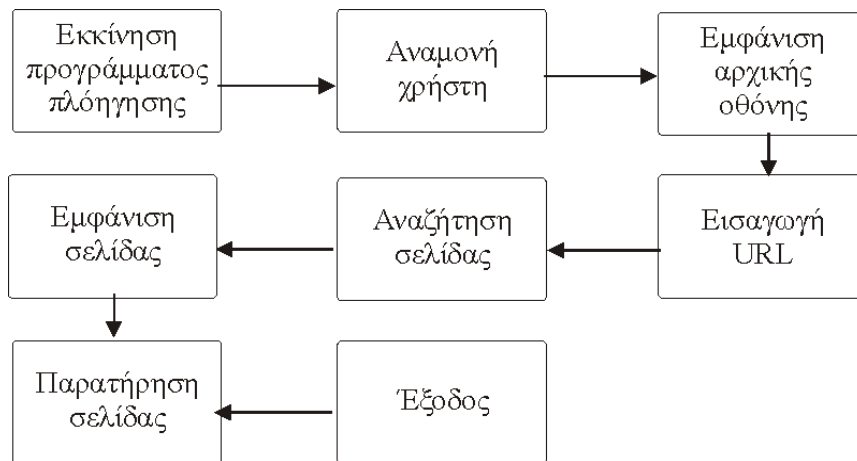
Σ1.Βήμα4: Ο χρήστης εισάγει την διεύθυνση μιας σελίδας (URL)

Σ1.Βήμα5: Το πρόγραμμα πλοήγησης αναζητά την σελίδα από το διαδίκτυο

Σ1.Βήμα6: Το πρόγραμμα πλοήγησης εντοπίζει τη σελίδα και την εμφανίζει

Σ1.Βήμα7: Ο χρήστης την παρατηρεί

Σ1.Βήμα8: Ο χρήστης εξέρχεται του συστήματος



Το δεύτερο σενάριο (Σενάριο Σ2) περιγράφει μια περίπτωση ανεπιτυχούς προσπάθειας αναζήτησης. Το Σενάριο Σ2 συνίσταται στην παρακάτω ακολουθία βημάτων:

Σ2.Βήμα1: Ο χρήστης ξεκινά το πρόγραμμα πλοήγησης

Σ2.Βήμα2: Ο χρήστης αναμένει

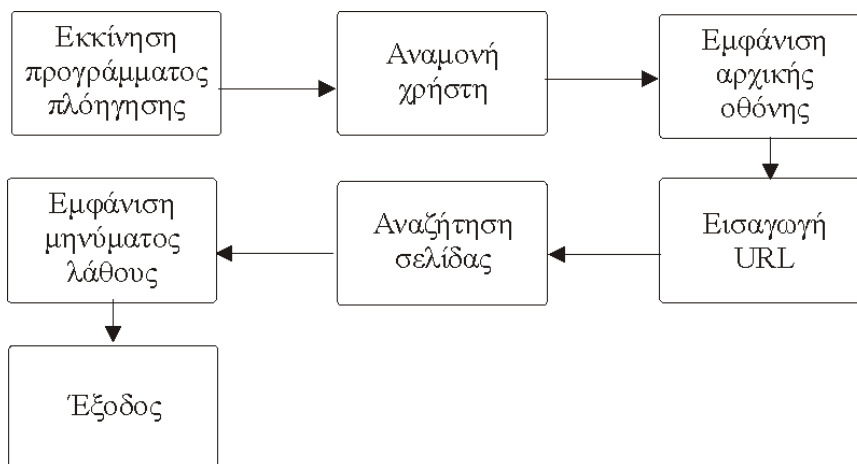
Σ2.Βήμα3: Εμφανίζεται η αρχική οθόνη του προγράμματος πλοήγησης

Σ2.Βήμα4: Ο χρήστης εισάγει την διεύθυνση μιας σελίδας (URL)

Σ2.Βήμα5: Το πρόγραμμα πλοήγησης αναζητά την σελίδα από το διαδίκτυο

Σ2.Βήμα6: Το πρόγραμμα πλοήγησης δεν εντοπίζει τη σελίδα και εμφανίζει μήνυμα λάθους

Σ2.Βήμα7: Ο χρήστης εξέρχεται του συστήματος



Η ανάλυση των σεναρίων μπορεί να οδηγήσει σε μια καταγραφή των καταστάσεων στις οποίες μπορεί να περιέλθει το σύστημα. Ανάλογα με την οπτική γωνία που επιλέγουμε μπορούμε να εξετάσουμε τις καταστάσεις από την πλευρά του χρήστη ή τις καταστάσεις από την πλευρά του συστήματος. Στην πρώτη περίπτωση οι καταστάσεις είναι:

K1: Εκκίνηση του συστήματος

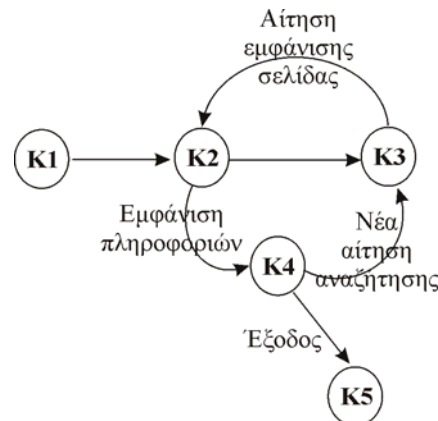
K2: Αναμονή χρήστη

K3: Αίτηση αναζήτησης

K4: Παρατήρηση αποτελέσματος (μήνυμα λάθους ή σελίδα)

K5: Έξοδος από το σύστημα

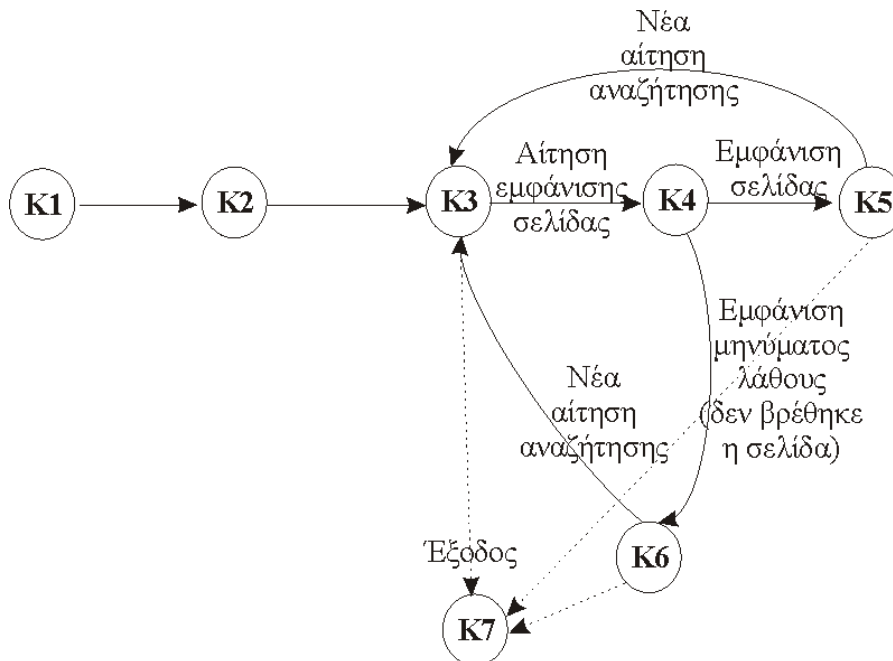
Γνωρίζοντας και τη διαδοχή των καταστάσεων δημιουργούμε το ακόλουθο Δίκτυο Μετάβασης Κατάστασης.



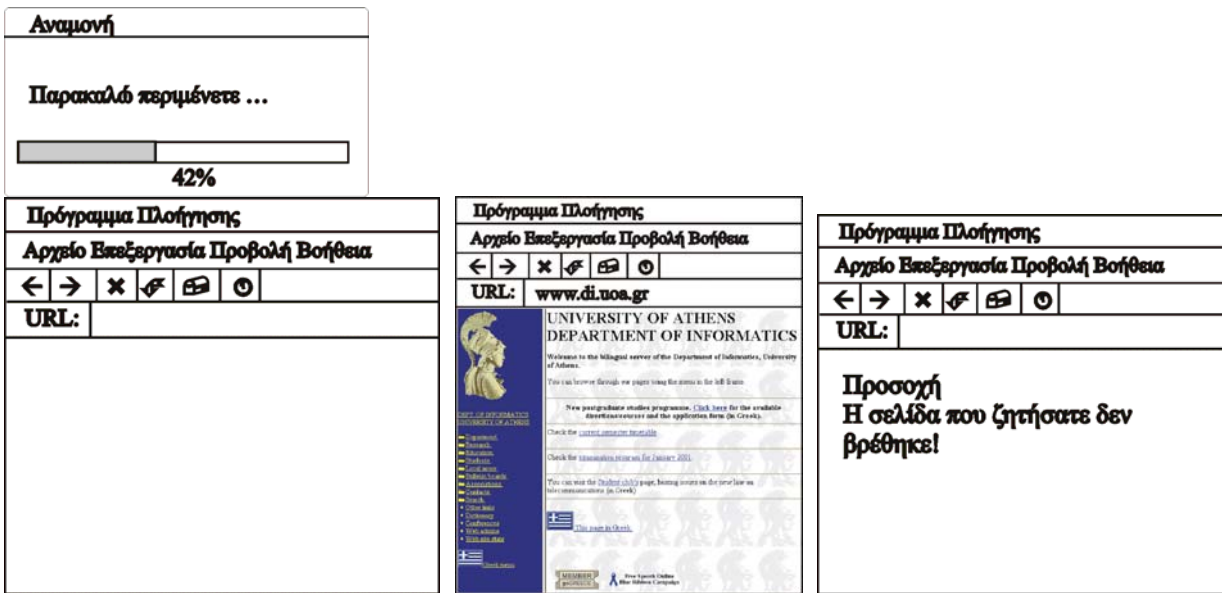
Στην δεύτερη περίπτωση οι καταστάσεις είναι οι ακόλουθες:

- K1: εκκίνηση προγράμματος πλοήγησης
- K2: αρχικοποίηση μεταβλητών
- K3: εισαγωγή διεύθυνσης από τον χρήστη
- K4: αναζήτηση σελίδας
- K5: εμφάνιση σελίδας
- K6: εμφάνιση μηνύματος λάθους
- K7: έξοδος από το πρόγραμμα

Παίρνοντας υπ' όψη και το πώς το σύστημα μεταβαίνει από την μία κατάσταση στην άλλη, καταλήγουμε στο παρακάτω δίκτυο μετάβασης καταστάσεων:

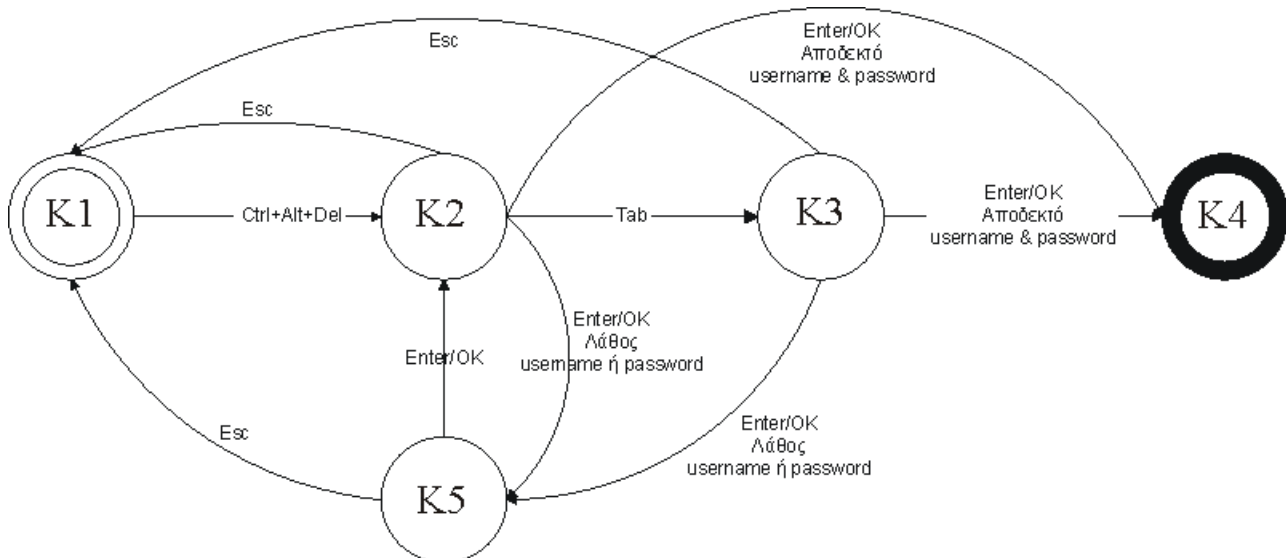


Έχοντας μια αρκετά καθαρή εικόνα για το πώς συμπεριφέρεται ο χρήστης και το σύστημα, ολοκληρώνουμε την σχεδίαση με την συγκεκριμενοποίηση της εμφάνισης των διαφόρων οθονών που είναι οι ακόλουθες:



*Διαδικασία σύνδεσης (login)*

Στο δεύτερο παράδειγμα αναφέρουμε τη διαδικασία σύνδεσης σε κάποιο λειτουργικό σύστημα. Το Δίκτυο Μετάβασης Κατάστασης που περιγράφει τη σχετική διαδικασία εμφανίζεται στην επόμενη εικόνα. Σε αυτό το παράθυρο επικεντρωθήκατε στα βήματα που πρέπει να εκτελέσει ο χρήστης και στις αντίστοιχες οθόνες.



- K1. Εμφάνιση αρχικής εικόνας
- K2. Εμφάνιση οθόνης εισαγωγής
- K3. Εισαγωγή username
- K4. Εισαγωγή password
- K5. Είσοδος στο σύστημα
- K6. Μήνυμα λάθους

Στη συνέχεια παρουσιάζουμε τις οθόνες.

Σύνδεση στο σύστημα  Πατήστε Ctrl+Alt+Del για να συνδεθείτε	Σύνδεση στο σύστημα Username: <input type="text"/> Password: <input type="text"/> <input type="button" value="Σύνδεση"/> <input type="button" value="Ακύρωση"/>	Σύνδεση στο σύστημα  Λανθασμένο username ή password <input type="button" value="Εντάξει"/>
--	--	--

Ερωτήσεις:

Χρησιμοποιείτε το μοντέλο GOMS για να αναλύσετε τις ακόλουθες διαδικασίες στο MS-DOS, Unix και Windows:

- Διαγραφή ενός αρχείου
- Μεταφορά ενός αρχείου
- Διαγραφή ενός καταλόγου
- Μεταφορά ενός καταλόγου

Σχεδιάστε το δίκτυο μετάβασης κατάστασης για τις παρακάτω περιπτώσεις:

- Διαδικασία ανάκτησης ενός αρχείου
- Διαδικασία σύνδεσης στο Unix, Windows NT σε Domain
- Διαδικασία ανάληψης χρημάτων από Αυτόματη Ταμειακή Μηχανή
- Διαδικασία προγραμματισμού εγγραφής σε συσκευή βίντεο

## Κεφάλαιο 9 Αξιολόγηση Συστημάτων EAM

Η αξιολόγηση αφορά την συλλογή και την επεξεργασία δεδομένων σχετικών με την ευχρηστία και τη λειτουργικότητα του σχεδιασμού ή της υλοποίησης ενός συγκεκριμένου προϊόντος. Ειδικότερα, στην περίπτωση της Επικοινωνίας Ανθρώπου-Μηχανής η αξιολόγηση εξετάζει τη λειτουργικότητα και την ευχρηστία των διεπαφών, στοχεύοντας στον προσδιορισμό της επίδρασής τους στον άνθρωπο-χρήστη καθώς και στην αναγνώριση προβλημάτων τόσο κατά τη διάρκεια του σχεδιασμού όσο και κατά τη διάρκεια της υλοποίησης.

Η αξιολόγηση είναι μια συνεχής διαδικασία που λαμβάνει χώρα καθ' όλη τη διάρκεια της ανάπτυξης ενός συστήματος. Ανάλογα με τη φάση στην οποία βρίσκεται η ανάπτυξη ενός συστήματος μπορούμε να διακρίνουμε την αξιολόγηση στη φάση της σχεδίασης και την αξιολόγηση στη φάση της υλοποίησης. Θα πρέπει να γίνει κατανοητό ότι η αξιολόγηση στη μια από τις δύο φάσεις, δεν αναιρεί την ανάγκη αξιολόγησης κατά τη διάρκεια της δεύτερης. Η βασική διαφορά ανάμεσα στις μεθόδους αξιολόγησης που εκτελούνται κατά τη διάρκεια του σχεδιασμού και αυτών που εκτελούνται κατά τη διάρκεια της υλοποίησης είναι ότι οι μεν πρώτες εστιάζονται στην εργασία που εκτελεί ο σχεδιαστής του συστήματος και δεν περιλαμβάνουν συνήθως χρήστες, ενώ στη δεύτερη περίπτωση μελετάται η λειτουργία του συστήματος. Κατά τη διάρκεια του σχεδιασμού μπορούν να χρησιμοποιηθούν μέθοδοι όπως η ευρεστική αξιολόγηση [6], οι γνωσιακοί περίπατοι [7, 8], αλλά και άλλες μέθοδοι με διαφορετικά πλεονεκτήματα και μειονεκτήματα [9, 10]. Κατά τη διάρκεια της υλοποίησης μπορούν να χρησιμοποιηθούν μέθοδοι όπως η πειραματική αξιολόγηση του λογισμικού προϊόντος με τη βοήθεια ομάδων χρηστών, ή η παρακολούθηση των χρηστών και η καταγραφή με διάφορα μέσα των ενεργειών τους ή και συνεντεύξεις των χρηστών ώστε να αποτυπωθεί η γνώμη τους [11, 12].

πριν εξετάσουμε κάπως λεπτομερειακά τις μεθόδους αξιολόγησης σχεδιασμού και υλοποίησης, σημειώνουμε ότι μπορούμε να διακρίνουμε τις μεθόδους αξιολόγησης και ανάλογα με το χώρο στον οποίο εκτελούνται και τους ανθρώπους που εμπλέκονται (π.χ. σχεδιαστές, χρήστες, κτλ.). Αναφορικά με αυτό, κατά βάση διακρίνουμε την αξιολόγηση σε περιβάλλον εργαστηρίου και την αξιολόγηση στο εργασιακό περιβάλλον των χρηστών. Κάθε μια από τις επιλογές έχει αντίστοιχα πλεονεκτήματα και μειονεκτήματα.

- *Αξιολόγηση σε περιβάλλον εργαστηρίου:* Στην περίπτωση αυτή η αξιολόγηση λαμβάνει χώρα σε ένα ειδικά διαμορφωμένο χώρο. Σε μερικές περιπτώσεις, ιδιαίτερα όταν η αξιολόγηση αφορά το σχεδιασμό, σε μια τέτοια διαδικασία αξιολόγησης εμπλέκεται μόνο ο σχεδιαστής. Στην περίπτωση όμως που κριθεί αναγκαίο (συνήθως σε αξιολόγηση υλοποίησης), χρήστες μπορούν να μεταφερθούν στο εργαστήριο. Το περιβάλλον εργαστηρίου προσφέρει πλεονεκτήματα όπως η δυνατότητα χρήσης εξελιγμένων οργάνων μετρήσεων και παρακολούθησης (κάμερες, καθρέπτες διπλής όψεως, ειδικά διαμορφωμένους υπολογιστές με λογισμικό που επιτρέπει την καταγραφή των κινήσεων των χρηστών ή/ και τον ταυτόχρονο σχολιασμό από τους σχεδιαστές, κ.α.). Ειδικά για περιπτώσεις που το περιβάλλον χρήσης του υπολογιστικού συστήματος βρίσκεται σε απομακρυσμένη ή επικίνδυνη τοποθεσία (όπως ο υπολογιστής ενός αεροπλάνου) ο χώρος του εργαστηρίου μπορεί να είναι η μόνη λύση. Από τη άλλη όμως το απομονωμένο περιβάλλον του εργαστηρίου δεν ευνοεί τη δημιουργία συνθηκών εργασίας που να πλησιάζουν τις πραγματικές. Αυτό σε μερικές περιπτώσεις μπορεί να οδηγήσει την αξιολόγηση σε λανθασμένα συμπεράσματα.

- *Αξιολόγηση σε περιβάλλον εργασίας*: Στην περίπτωση αυτή η αξιολόγηση λαμβάνει χώρα στο περιβάλλον εργασίας του χρήστη, στον οποίο μεταφέρεται ο σχεδιαστής ή ο υπεύθυνος αξιολογητής. Σε μια τέτοια περίπτωση η αξιολόγηση μπορεί να διαρκέσει μεγάλο χρονικό διάστημα και η εργασία του αξιολογητή δυσχεραίνεται από τους εγγενείς θορύβους και διακοπές του εργασιακού χώρου. Από την άλλη όμως ο αξιολογητής έχει τη δυνατότητα να παρατηρήσει γεγονότα που δεν θα συνέβαιναν στο χώρο του εργαστηρίου (αν και η παρουσία του αξιολογητή στο χώρο εργασίας των χρηστών μπορεί να προκαλέσει αλλαγή στη συνήθη συμπεριφορά των τελευταίων).

Συνολικά παρότι η αξιολόγηση στο περιβάλλον εργασίας των χρηστών μπορεί να κοστίζει πολύ περισσότερο, είναι εν γένει προτιμότερη μια και μπορεί να δώσει ακριβή εικόνα της διάδρασης ανθρώπου – υπολογιστή σε πραγματικές συνθήκες. Έχοντας υπ' όψη μας τα παραπάνω, προχωράμε τώρα στις δύο υποενότητες που ακολουθούν στην ανάλυση μεθόδων αξιολόγησης σχεδιασμού και υλοποίησης.

### 9.1 Αξιολόγηση σχεδιασμού

Όπως ήδη αναφέραμε μπορούμε να διαχωρίσουμε τις μεθόδους αξιολόγησης ανάλογα με τη φάση ανάπτυξης κατά την οποία εκτελούνται. Ειδικότερα η αξιολόγηση του σχεδιασμού έχει ως στόχο την αξιολόγηση του σχεδιαστικής προσέγγισης που ακολουθήθηκε και η έγκαιρη χρήση του (κατά προτίμηση πριν την έναρξη της φάσης υλοποίησης) μπορεί να βοηθήσει στον εντοπισμό σφαλμάτων, των οποίων η καθυστερημένη απαλοιφή είναι δυνατό να κοστίζει ιδιαίτερα τόσο σε πόρους όσο και σε χρόνο.

Κατά τη διάρκεια του σχεδιασμού υπάρχουν διάφορες μέθοδοι που μπορούν να αξιοποιηθούν τόσο αυτόνομα όσο και σε συνδυασμό. Στη συνέχεια αναφέρουμε τις κυριότερες:

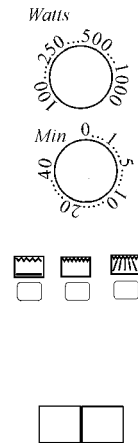
#### Γνωσιακός Περίπατος

Ο γνωσιακός περίπατος εκτελείται από έναν σχεδιαστή ή κάποιον έμπειρο στη γνωσιακή ψυχολογία με στόχο να ανακαλύψει σε ποιο βαθμό το σχεδιαζόμενο σύστημα υποστηρίζει το χρήστη στην εκτέλεση των εργασιών του. Ο αξιολογητής "περιδιαβαίνει" βήμα προς βήμα τα σημεία που απαιτεί η ολοκλήρωση κάθε ενέργειας και τα συγκρίνει με ψυχολογικά κριτήρια. Για κάθε εργασία που αξιολογείται, ο αξιολογητής εξετάζει θέματα όπως ποια είναι η επίδραση της διάδρασης στο χρήστη, ποιες γνωσιακές διεργασίες απαιτούνται, ή ποια προβλήματα εκμάθησης μπορεί να ανακύψουν. Για να διευκολυνθεί το έργο του αξιολογητή, ο τελευταίος έχει στη διάθεσή του μια φόρμα με ερωτήσεις σχετικές με τις ενέργειες και τους στόχους του χρήστη. Συνολικά, για να εκτελεστεί η διαδικασία του γνωσιακού περιπάτου τέσσερα είναι τα απαραίτητα συστατικά:

- περιγραφή (πρωτότυπου) συστήματος, έστω και ημιτελής
- περιγραφή των στόχων του χρήστη
- πλήρης ακολουθία ενεργειών του χρήστη που απαιτούνται για την επίτευξη κάθε στόχου
- ενδείξεις για τον χρήστη (εμπειρία, ηλικία, κλπ.)

Σαν παράδειγμα αξιολόγησης με χρήση γνωσιακού περιπάτου, ας θεωρήσουμε ότι σχεδιάζουμε τη διεπαφή ενός χρήστη με έναν φούρνο μικροκυμάτων με γκριλ. Ο αρχικός σχεδιασμός είναι ο ακόλουθος:





Θέλουμε να γνωρίζουμε εάν ο σχεδιασμός βοηθά το χρήστη. Ξεκινούμε αναγνωρίζοντας την εργασία: *προγραμματισμός ψησίματος σε γκριλ και μικροκύματα*. Θεωρώντας ότι ο χρήστης έχει προηγούμενη γνώση με φούρνους αλλά όχι με το συγκεκριμένο μοντέλο, καθορίζουμε τα βήματα (σε υψηλό, αφηρημένο επίπεδο) που θέτει.

- Θέσε χρόνο ψησίματος
- Θέσε επίπεδο ισχύος μικροκυμάτων
- Επέλεξε αντίστοιχο πρόγραμμα
- Πάτησε το πλήκτρο εκκίνησης

Ο στόχος εμφανίζεται όπως ο χρήστης τον θεωρεί, λαμβάνοντας υπόψη την εμπειρία του. Αναλύοντας τα παραπάνω βήματα σε κινήσεις:

1. Θέσε χρόνο ψησίματος  
επιλογή του αντίστοιχου χρόνου από το χρονόμετρο
2. Θέσε επίπεδο ισχύος μικροκυμάτων  
επιλογή της αντίστοιχης ισχύος από τον επιλογέα
3. Επέλεξε αντίστοιχο πρόγραμμα  
απο-επιλογή προηγούμενου προγράμματος  
επιλογή επιθυμητού προγράμματος
4. Πάτησε το πλήκτρο εκκίνησης  
πάτημα του πλήκτρου εκκίνησης

Στη συνέχεια προσπαθούμε να απαντήσουμε στα παρακάτω, για κάθε ενέργεια του χρήστη:

- θα μπορέσει ο χρήστης να προσθέσει όλα τα απαραίτητα βήματα;
- όντως ο χρήστης προσπαθεί να πετύχει αυτό που κάνει η ενέργεια;
- φαίνονται τα εργαλεία με τα οποία η επόμενη ενέργεια μπορεί να πραγματοποιηθεί;
- αφού βρει το κατάλληλο εργαλείο, ξέρει ότι είναι το σωστό;
- καταλαβαίνει οι χρήστες ότι έκαναν το σωστό πράγμα (ανάδραση);

Στο συγκεκριμένο παράδειγμα στη διεπαφή δεν υπάρχει κάποια ένδειξη για το ότι πρέπει να απο-επιλεγεί το προηγούμενο πρόγραμμα, ούτε είναι άμεσα κατανοητό τι θα συμβεί εάν πατήσουμε και το πλήκτρο "Γκριλ" και το πλήκτρο "Γκριλ και μικροκύματα". Σ' αυτό συμβάλλει και το γεγονός ότι το σύστημα δε μας προσφέρει κάποια ανάδραση (π.χ. ενδεικτική λυχνία) για το πρόγραμμα που πραγματικά εκτελείται.

### Ευρεστική Αξιολόγηση

Η ευρεστική αξιολόγηση είναι παρόμοια μέθοδος με το γνωσιακό περίπατο μια και στην εκτέλεσή της συμμετέχουν αρκετοί αξιολογητές (5 αρκούν για να βρουν το 75% των προβλημάτων) που εξετάζουν το σχεδιασμό σε σχέση με γνωστά κριτήρια χρηστικότητας, είναι όμως λιγότερο δομημένη από το γνωσιακό περίπατο. Τα κριτήρια χρηστικότητας που χρησιμοποιούνται έχουν να κάνουν με οδηγίες (guidelines) και πρότυπα (standards). Μερικοί βασικοί ευρεστικοί κανόνες είναι οι ακόλουθοι:

- απλός και φυσικός διάλογος
- χρήση της γλώσσας του χρήστη (όσο περισσότερο πλησιάζει ο διάλογος τη γλώσσα του χρήστη τόσο μικραίνει ο χρόνος εκμάθησης, μια και ο χρήστης δεν χρειάζεται να μάθει επιπλέον και τη γλώσσα του συστήματος)
- ελαχιστοποίηση του φόρτου μνήμης
- συνέπεια
- ανάδραση στις ενέργειες του χρήστη
- ξεκάθαρες έξοδοι από τμήματα του συστήματος, για να δίνεται όσο το δυνατό μεγαλύτερη ευελιξία στη χρήση του
- πλήκτρα συντόμευσης (shortcuts) για τους έμπειρους χρήστες
- περιγραφικά, ακριβή μηνύματα σφαλμάτων
- δυνατότητα στο χρήστη να διορθώνει σφάλματα

Στους παραπάνω κανόνες μπορούμε να προσθέσουμε ένα σύνολο δέκα νεώτερων και πιο ολοκληρωμένων κανόνων:

- ορατότητα της κατάστασης του συστήματος
- συμφωνία μεταξύ του συστήματος και του έξω κόσμου (φυσικότητα, όχι τεχνικοί όροι,...)
- ελευθερία χρήστη και έλεγχος στα χέρια του (σημεία εξόδου -> υποστήριξη undo/redo)
- συνέπεια και πρότυπα
- αποφυγή σφαλμάτων από τη φύση του συστήματος
- αναγνώριση αντί για ανάκληση (υπάρχουν δυο τρόποι προσπέλασης της μνήμης: αναγνώριση και ανάκληση, με την αναγνώριση να είναι πιο αποτελεσματική από την ανάκληση)
- ευελιξία και ταχύτητα χρήσης, διαφοροποίηση αρχαρίων και προχωρημένων χρηστών, δυνατότητα προσαρμογής του συστήματος στις συνήθειες του χρήστη
- αισθητικός και μινιμαλιστικός σχεδιασμός
- άμεση αναγνώριση, διάγνωση και διόρθωση σφαλμάτων
- βοήθεια και τεκμηρίωση (documentation)

### Αξιολόγηση Βασισμένη στη Μελέτη Άλλων Αξιολογήσεων

Ο τύπος αξιολόγησης που βασίζεται στη μελέτη άλλων αξιολογήσεων αξιοποιεί τον πλούτο γνώσης που παρέχουν προηγούμενες μελέτες αξιολόγησης. Αν και η αξιολόγηση έχει άμεση σχέση με το σύστημα που αξιολογείται, πολλά από τα συμπεράσματα είναι αρκετά γενικά και άμεσα εφαρμόσιμα και σε άλλες περιπτώσεις (π.χ. μέγεθος ή αναγνωρισιμότητα εικονιδίων, σειρά μενού ή εντολών μέσα στα μενού, κλπ). Βέβαια αυτό δε σημαίνει ότι όλα τα συμπεράσματα που έχουν εξαχθεί από προηγούμενες μελέτες αξιολόγησης μπορούν άμεσα να εφαρμοσθούν σε οποιαδήποτε περίπτωση. Μπορούν όμως να βοηθήσουν στην εξοικονόμηση χρόνου και πόρων.

## **Αξιολόγηση Βασισμένη σε Μοντέλα**

Αρκετά γνωσιακά και σχεδιαστικά μοντέλα προσφέρουν ενδείξεις για τις επιδόσεις των χρηστών. Για παράδειγμα το GOMS ή το KLM μπορούν να βοηθήσουν στην πρόβλεψη των επιδόσεων των χρηστών και να βοηθήσουν στην επιλογή συγκεκριμένων σχεδιαστικών προσεγγίσεων.

### **9.2 Αξιολόγηση Υλοποίησης**

Οι τεχνικές αξιολόγησης που αναφέρθηκαν μέχρι τώρα αφορούν την ανακάλυψη σφαλμάτων και προβλημάτων κατά το στάδιο του σχεδιασμού. Παρότι κάτι τέτοιο είναι αναγκαίο, δεν είναι από μόνο του αρκετό ώστε να μη χρειαστεί αξιολόγηση κατά το στάδιο της υλοποίησης. Η αξιολόγηση σ' αυτό το στάδιο περιλαμβάνει τη δοκιμασία του προϊόντος με πραγματικούς χρήστες (αντί των αξιολογητών) που είναι και οι πιο κατάλληλοι για την ανάδειξη προβλημάτων μια και είναι τα άτομα στα οποία απευθύνεται το σύστημα. Συνολικά μπορούμε να διακρίνουμε τρεις τρόπους αξιολόγησης της υλοποίησης: πειραματική αξιολόγηση, τεχνικές ερωτήσεων και τεχνικές παρατήρησης.

#### **Πειραματική αξιολόγηση**

Η πειραματική αξιολόγηση αφορά τη χρήση ενός ελεγχόμενου πειράματος έτσι ώστε να εξεταστούν βασικές πτυχές ολόκληρου ή μέρους του συστήματος. Η πειραματική αξιολόγηση προσφέρει εμπειρικές αποδείξεις για την υποστήριξη της ορθότητας των υποθέσεων που ελήφθησαν κατά το σχεδιασμό. Οποιοδήποτε πείραμα περιλαμβάνει τα ίδια κύρια βήματα:

1. Καταρχήν ο υπεύθυνος για την αξιολόγηση διατυπώνει μια υπόθεση της οποίας την ορθότητα θα πρέπει να δοκιμάσει μετρώντας κάποιες παραμέτρους και χαρακτηριστικά του συστήματος.
2. Ο αξιολογητής επιλέγει τις ανεξάρτητες και τις εξαρτημένες μεταβλητές. Οι ανεξάρτητες μεταβλητές είναι οι μεταβλητές που ελέγχει (και μεταβάλλει) ο αξιολογητής, ενώ οι εξαρτημένες είναι αυτές που μετρώνται. Για παράδειγμα, εάν θέλει να μετρήσει κάποιος την επίπτωση που έχει (εάν έχει) η αλλαγή του μεγέθους των εικονιδίων στο χρόνο που χρειάζεται ο χρήστης για να τα επιλέξει, η ανεξάρτητη μεταβλητή θα είναι το μέγεθος του εικονιδίου και η εξαρτημένη ο χρόνος επιλογής.
3. Προετοιμάζεται ένας αριθμός σεναρίων που διαφέρουν μόνο στις τιμές των ανεξάρτητων μεταβλητών. Ο αξιολογητής επιλέγει (ανάλογα και με την υπόθεση και το δείγμα που έχει στη διάθεσή του) εάν θα εκτελέσει το πείραμα *μεταξύ-ομάδων* (between groups) ή *εντός-ομάδων* (within groups). Στην περίπτωση που το πείραμα εκτελείται μεταξύ ομάδων κάθε ομάδα χρηστών δοκιμάζεται σε ένα μόνο σενάριο. Στην περίπτωση που το πείραμα εκτελείται εντός ομάδων κάθε χρήστης δοκιμάζεται σε όλα τα σενάρια. Η μέθοδος μεταξύ-ομάδων πλεονεκτεί μια και οι χρήστες δεν αποκτούν εμπειρία από κάθε σενάριο που εκτελούν ώστε να επηρεάζονται. Το μειονέκτημα αυτής της μεθόδου είναι ο αυξημένος αριθμός υποκειμένων που απαιτούνται. Επίσης απαιτεί και ιδιαίτερη προσοχή στην κατανομή των υποκειμένων σε ομάδες έτσι ώστε οι ομάδες μεταξύ τους να μην έχουν διαφορές. Σε ορισμένες περιπτώσεις μπορεί να επιλεγεί μικτή μέθοδος (π.χ., στην περίπτωση δύο ανεξάρτητων μεταβλητών, η μια να μετρηθεί μεταξύ-ομάδων και η άλλη εντός-ομάδων).
4. Εκτελείται το πείραμα και μετριοούνται οι εξαρτημένες μεταβλητές. Οποιαδήποτε μεταβολή στις μετρήσεις αποδίδεται στην αντίστοιχη μεταβολή των ανεξάρτητων μεταβλητών.
5. Τα δεδομένα που συλλέχθηκαν περνούν από στατιστική ανάλυση. Η απλή παρατήρηση μπορεί να δώσει το μεγαλύτερο μέρος του ζητούμενου αποτελέσματος, όμως με τη στατιστική ανάλυση μπορεί να διαφανούν ιδιαίτερα χαρακτηριστικά και να εξαχθούν πιο ασφαλή

συμπεράσματα. Στην περίπτωση που τα δεδομένα περνούν από στατιστική ανάλυση είναι χρήσιμο να φυλάσσονται και στην αρχική τους μορφή ώστε να είναι δυνατό να επανέλθει κάποιος για επιπλέον επεξεργασία εάν χρειαστεί.

Δεδομένου ότι η πειραματική αξιολόγηση στηρίζεται στην επεξεργασία δεδομένων που συλλέγονται κατά την διάδραση χρηστών με το πληροφορικό σύστημα είναι απαραίτητο να δοθεί ιδιαίτερη προσοχή στα ακόλουθα:

- Στα χαρακτηριστικά των ατόμων του δείγματος τα οποία θα πρέπει να συμπίπτουν με τα χαρακτηριστικά των χρηστών (ή των προβλεπόμενων χρηστών) του προϊόντος. Παράμετροι όπως η ηλικία, η εμπειρία, το μορφωτικό επίπεδο αλλά και φυσικά χαρακτηριστικά στις περιπτώσεις χρηστών με ειδικές ανάγκες, θα πρέπει να εξετασθούν κατά την επιλογή του δείγματος.
- Στις εργασίες που θα κληθούν να εκτελέσουν οι χρήστες που θα λάβουν μέρος στο πείραμα.
- Στο χώρο στον οποίο θα λάβει χώρα η αξιολόγηση. Το περιβάλλον του πειράματος μπορεί να είναι ο χώρος εργασίας των χρηστών, μπορεί όμως και να είναι κάποιος ειδικά διαμορφωμένος χώρος όπως ο χώρος ενός εργαστηρίου.

Κάθε πείραμα πρέπει να σχεδιασθεί προσεκτικά ώστε να συλλεχθούν αξιόπιστα και γενικεύσιμα αποτελέσματα. Εκτός από τις παραμέτρους που αναφέρουμε πιο πάνω, υπάρχουν και μερικά ακόμη στοιχεία στα οποία πρέπει να δοθεί προσοχή. Το πρώτο βήμα είναι η επιλογή της υπόθεσης. Η υπόθεση περιγράφει καθαρά το στόχο που θα πρέπει να αποδειχθεί. Με τη διατύπωση της υπόθεσης είναι δυνατή και η καταγραφή των εξαρτημένων μεταβλητών (δηλαδή των μεταβλητών που μετρώνται) και των ανεξάρτητων μεταβλητών (δηλαδή των μεταβλητών που ελέγχει και χειρίζεται ο αξιολογητής).

### Τεχνικές παρατήρησης

Οι μέθοδοι επερωτήσεων είναι λιγότερο 'επίσημες' από τις πειραματικές, μπορούν όμως να αναδείξουν μέσα από την οπτική γωνία του χρήστη, πτυχές του συστήματος που δεν είναι δυνατό να αποτυπωθούν με άλλες μεθόδους. Στις τεχνικές αυτές διακρίνουμε τις εξής:

- *Φωναχτή σκέψη.* Στην τεχνική αυτή οι χρήστες καλούνται να εκτελέσουν ορισμένες προκαθορισμένες ενέργειες και καθώς τις εκτελούν εκφράζουν δυνατά τις σκέψεις τους, να πουν την άποψή τους για τυχόν προβλήματα που αντιμετωπίζουν, πώς θα προσπαθήσουν να τα λύσουν, κλπ.
- *Ανάλυση πρωτοκόλλου.* Υπάρχει ένα σύνολο διαφορετικών τεχνικών για την καταγραφή των κινήσεων των χρηστών:
  - Χαρτί και μολύβι. Είναι απλή μέθοδος που περιορίζεται από την ταχύτητα αυτού που κρατά σημειώσεις.
  - Μαγνητοφώνηση. Είναι χρήσιμη εάν ο χρήστης 'σκέπτεται φωναχτά', όμως η μετατροπή σε κείμενο μπορεί να είναι πρόβλημα.
  - Μαγνητοσκόπηση. Έχει το επιπλέον πλεονέκτημα ότι επιτρέπει να δει κανείς τι κάνει ο χρήστης όσο εργάζεται (και όσο ο χρήστης παραμένει στο οπτικό πεδίο της κάμερας).
  - Παρακολούθηση μέσω υπολογιστή. Το σύστημα καταγράφει τις κινήσεις του χρήστη, δηλαδή τη χρήση του πληκτρολογίου και του ποντικιού. Η τεχνική αυτή είναι σχετικά απλή δημιουργεί όμως μεγάλο όγκο δεδομένων που πρέπει να τύχουν επεξεργασίας ώστε να εξαχθούν συμπεράσματα.
  - Σημειώσεις χρήστη. Στην τεχνική αυτή οι ίδιοι οι χρήστες κρατούν σημειώσεις σχετικά με τις εργασίες που εκτελούν και τα προβλήματα που αντιμετωπίζουν.
- *Εργαλεία αυτόματης ανάλυσης πρωτοκόλλων.* Μια και η ανάλυση πρωτοκόλλου είναι επίπονη και κουραστική εργασία, έχουν αναπτυχθεί εργαλεία τα οποία αυτοματοποιούν την ανάλυση. Κάποια από αυτά τα εργαλεία επιτρέπουν την άμεση εισαγωγή σχολίων από την μεριά του

αξιολογητή καθώς μαγνητοσκοπείται ο χρήστης, το συγχρονισμό μεταξύ της καταγραφής των κινήσεων του χρήστη από τον υπολογιστή και της μαγνητοσκόπησης ή της μαγνητοφώνησης, καθώς και της επεξεργασίας των δεδομένων που συλλέγονται από τα εργαλεία παρακολούθησης μέσω υπολογιστή.

- *Εκ των υστέρων περίπατοι.* Σε αρκετές περιπτώσεις τα δεδομένα που συλλέγονται δεν είναι αρκετά λεπτομερή (π.χ. μπορεί ο χρήστης να λέει ότι επιλέγει μια εντολή χωρίς να εξηγήσει τι τον οδήγησε σ' αυτήν την απόφαση). Σε τέτοιες περιπτώσεις μετά το τέλος της εργασίας είναι δυνατό ο χρήστης να επανέλθει (πιθανότατα βλέποντας την ταινία που κατέγραφε τις κινήσεις του ή τις σημειώσεις του) και να επεξηγήσει κάποιες ενέργειές του.

### Τεχνικές (επ)ερωτήσεων

Οι τεχνικές ερωτήσεων δίνουν τη δυνατότητα στους χρήστες να εκφράσουν άμεσα την άποψή τους.

- *Συνεντεύξεις.* Οι συνεντεύξεις έχουν το πλεονέκτημα ότι ο αξιολογητής μπορεί να μεταβάλλει τη ροή της συνέντευξης δίνοντας περισσότερο βάρος σε κάποια απαντήσεις του χρήστη που θεωρεί ότι έχουν μεγαλύτερη σημασία. Τα αποτελέσματα που εξάγονται από τις συνεντεύξεις είναι συνήθως μη μετρήσιμα αλλά μπορούν να βοηθήσουν στην αναγνώριση σημαντικών προβλημάτων.
- *Ερωτηματολόγια.* Τα ερωτηματολόγια είναι λιγότερο ευέλικτα μπορούν όμως να χρησιμοποιηθούν για τη συλλογή και μετρήσιμων δεδομένων. Τις ερωτήσεις μπορούμε να τις διακρίνουμε στους εξής τύπους:
  - Γενικές (ή ταυτότητας). Είναι ερωτήσεις που αφορούν το γενικό προφίλ του χρήστη.
  - Ελευθέρου κειμένου. Είναι ανοικτές ερωτήσεις του τύπου: "Μπορείτε να υποδείξετε κάποιες βελτιώσεις;"
  - Γραμμικές ή κλίμακας. Είναι ερωτήσεις της μορφής:

Χρησιμοποιείτε το ποντίκι για να επιλέξετε κάποια από τις εντολές των μενού;

1. Πάντα	2. Συχνά	3. Μερικές φορές	4. Σπάνια	5. Ποτέ	6. Δεν γνωρίζω/ Δεν απαντώ
----------	----------	------------------	-----------	---------	----------------------------

- Πολλαπλών επιλογών (Multiple choice)
- Διατεταγμένες (1ος, 2ος,.....). Είναι ερωτήσεις της μορφής:  
 Διατάξτε ως προς την ευχρηστία τις ακόλουθες μεθόδους επιλογής εντολών από μενού (όπου 3 είναι η πιο εύχρηστη και 1 η πιο δύσχρηστη):  
 Με πληκτρολόγιο .....  
 Με ποντίκι .....  
 Με πλήκτρο συντόμευσης.....

### 9.3 Επιλογή Μεθόδου Αξιολόγησης

Η επιλογή ανάμεσα στις διάφορες επιλογές μεθόδων αξιολόγησης που μπορεί να είναι στην διάθεση του αξιολογητή μπορεί να βασισθεί σε οκτώ τουλάχιστον κριτήρια:

1. Το σύστημα βρίσκεται σε στάδιο σχεδιασμού ή υλοποίησης;
2. Θα είναι εργαστηριακή ή πραγματικού περιβάλλοντος;
3. Θα είναι υποκειμενική ή αντικειμενική (όσον αφορά τον αξιολογητή);
4. Θα είναι ποιοτική ή ποσοτική;
5. Ποια είναι η ποσότητα της πληροφορίας που δίνεται στον αξιολογητή;

6. Ποιο είναι το χρονικό περιθώριο για την εξαγωγή του αποτελέσματος της αξιολόγησης;
7. Σε ποιο βαθμό μπορεί η αξιολόγηση να παρέμβει στην εργασία των χρηστών;
8. Ποιοι είναι οι διαθέσιμοι πόροι;

#### 9.4 Πίνακας παρουσίασης μεθόδων αξιολόγησης διεπαφής

Μέθοδος Αξιολόγησης	Επικεντρώνεται σε	Φάση Ανάπτυξης	Υποστήριξη	Πλεονεκτήματα
<b>Ευρεστική αξιολόγηση</b>  (Εμπειρος σχεδιαστής)	Πρωτότυπο, υπάρχον προϊόν, σχεδιασμός σε χαρτί	Καταγραφή προδιαγραφών-σχεδιασμός	Γρήγορη μέθοδος και με μικρό κόστος, δεν απαιτούνται χρήστες	Εξάγει τη γνώμη 4, 5 εμπειρων  Βρίσκει τα πιο συνηθισμένα προβλήματα ευχρηστίας  Δοκιμασμένη μέθοδος που αποτρέπει την ανάγκη για εκτεταμένες δοκιμές χρηστικότητας
<b>Γνωσιακός Περίπατος</b>	Πρωτότυπο, υπάρχον προϊόν, σχεδιασμός σε χαρτί	Καταγραφή προδιαγραφών-σχεδιασμός, ακόμη και υλοποίηση	Γρήγορη μέθοδος και με μικρό κόστος, δεν απαιτούνται χρήστες	Βρίσκει τα πιο συνηθισμένα προβλήματα ευχρηστίας  Αξιόπιστη μέθοδος για τον έλεγχο της διεπαφής
<b>Αξιολόγηση Βασισμένη στη Μελέτη Άλλων Αξιολογήσεων</b>	Προδιαγραφές, Σχεδιασμός ή και Πρωτότυπο	Όλες οι φάσεις, συνήθως καταγραφή προδιαγραφών ή σχεδιασμός	Απαιτείται γνώση της περιοχής	Μπορεί να εξοικονομήσει χρόνο
<b>Αξιολόγηση Βασισμένη σε Μοντέλα</b>	Μοντέλο (GOMS, UAN, ...)	Σχεδιασμός	Απαιτεί δημιουργία μοντέλου  Το κόστος ανάλογο της πολυπλοκότητας της διεπαφής	Βρίσκει αρκετά προβλήματα (π.χ. πρόωρης ολοκλήρωσης στόχου, πολύπλοκων λειτουργιών)  Προσφέρει πρόβλεψη χρόνου εκτέλεσης
<b>Πειραματική αξιολόγηση στο εργαστήριο</b>	Πρωτότυπο ή πρώτη έκδοση λογισμικού	Σχεδιασμού, Υλοποίησης	Χρειάζεται ιδιαίτερη προετοιμασία, απαιτούνται 10 με 30 χρήστες και κατάλληλος εξοπλισμός	Επιτρέπει τον έλεγχο των εργασιών και του περιβάλλοντος  Επιτρέπει στην ομάδα ανάπτυξης να παρακολουθεί  Στα μειονεκτήματα περιλαμβάνονται οι απομόνωση των χρηστών και η απώλεια πληροφοριών που σχετίζονται με το πραγματικό περιβάλλον  Χρήσιμο για την συλλογή αντικειμενικών δεδομένων

Μέθοδος Αξιολόγησης	Επικεντρώνεται σε	Φάση Ανάπτυξης	Υποστήριξη	Πλεονεκτήματα
<b>Τεχνικές παρατήρησης στο χώρο του χρήστη</b>	Πρωτότυπο ή πρώτη έκδοση λογισμικού	Σχεδιασμού, Υλοποίησης	Χρειάζεται κάποια προετοιμασία, 1 με 2 μέρες στο χώρο του χρήστη, 10 με 30 χρήστες και καταγραφή σε βίντεο ή ταινία ήχου	Υψηλή αξιοπιστία, συνδυάζει την παρακολούθηση με τον έλεγχο  Οι χρήστες αποκαλύπτουν νέες ιδέες με την παρουσία τους στη δουλειά  Πολύ καλή κατανόηση της περιοχής, οι ομάδες σχεδιασμού αποκτούν ιδιαίτερη γνώση για την εργασία
<b>Τεχνικές (επ)ερωτήσεων</b>  (Παρακολούθηση και συνεντεύξεις στο περιβάλλον εργασίας, διανομή ερωτηματολογίων)	Τις εργασίες των χρηστών, την πραγματική ροή της δουλειάς, την κατανόηση του περιβάλλοντος και της χρήσης του προϊόντος ή την καταγραφή της άποψης των χρηστών για το λογισμικό  Πρωτότυπο ή πρώτη έκδοση λογισμικού	Καταγραφή προδιαγραφών, σχεδιασμός διεπαφής, Υλοποίησης	Χρειάζεται αρκετή προετοιμασία, απαιτεί χρόνο και πρόσβαση σε 4 με 10 χρήστες για συνεντεύξεις  Για ερωτηματολόγια χρειάζονται περισσότερα άτομα (> 30), το κόστος όμως σε σχέση με τις συνεντεύξεις είναι μικρότερο	Πραγματική κατανόηση της εργασίας  Αντίληψη των αναγκών των χρηστών και καλή μέθοδος για την ανακάλυψη νέων χαρακτηριστικών  Χρήσιμο για την αξιολόγηση πολύπλοκων εφαρμογών σε φάση πρωτοτυποποίησης  Χρήσιμο για την συλλογή υποκειμενικών δεδομένων

Ερώτηση: Σας ζητούν να αξιολογήσετε ένα σύστημα το οποίο χρησιμοποιείται ήδη, έτσι ώστε οι όποιες προτάσεις σας να χρησιμοποιηθούν στην επόμενη έκδοσή του. Τα χρήματα που σας διαθέτουν για αυτόν το σκοπό είναι σχετικά περιορισμένα. Ποια ή ποιες μεθόδους αξιολόγησης θα χρησιμοποιούσατε και γιατί;

## **ΜΕΡΟΣ ΙΙΙ ΣΧΕΔΙΑΣΗ ΔΙΕΠΑΦΩΝ**



## **Κεφάλαιο 10            Γενικές Αρχές**

Ο σχεδιασμός της διεπαφής αποτελεί το ενδιάμεσο βήμα μεταξύ της καταγραφής προδιαγραφών και της υλοποίησης της αντίστοιχης διεπαφής. Η διαδικασία του σχεδιασμού είναι συνήθως επαναληπτική και περιλαμβάνει δημιουργία πρωτοτύπων, την αξιολόγησή τους από τους χρήστες ή/ και έμπειρους σχεδιαστές και την αποκρυστάλλωση του σχεδιασμού της διεπαφής [Δείτε και στο αντίστοιχο κεφάλαιο].

Κατά τη διάρκεια του σχεδιασμού της διεπαφής ο σχεδιαστής θα πρέπει να μη ξεχνά ότι φτιάχνει ένα σύστημα το οποίο απευθύνεται κατ' αρχήν σε ανθρώπους και κατά δεύτερο λόγο σε χρήστες. Κάτω από αυτό το πρίσμα η σχεδιαζόμενη διεπαφή θα πρέπει να λαμβάνει υπόψη τις ανθρώπινες ιδιαιτερότητες όπως έχουν αναλυθεί στο κεφάλαιο της περιγραφής του ανθρώπου. Η επιλογή των χρωμάτων, των ετικετών, της τοποθέτησης των πλήκτρων και των λειτουργιών δεν θα πρέπει να αντιβαίνουν στα χαρακτηριστικά του ανθρώπου. Οι επόμενες εικόνες παρουσιάζουν λανθασμένη και ορθή χρήση κινούμενης εικόνας σε μια διεπαφή. Στην πρώτη περίπτωση η κινούμενη εικόνα αποσπά την προσοχή του χρήστη και πιθανότατα δεν του επιτρέπει να ολοκληρώσει τον στόχο του. Στη δεύτερη περίπτωση η κινούμενη εικόνα βοηθά το χρήστη να εντοπίσει τις νέες πληροφορίες μέσα στον όγκο των πληροφοριών που του παρουσιάζονται.

Στη συνέχεια του κεφαλαίου θα επικεντρωθούμε σε θέματα που αφορούν άμεσα στον άνθρωπο-χρήστη και ειδικότερα στο σχεδιασμό διεπαφών για περιβάλλοντα ΠΕΠΕ και ΠΠΠ. Θα προσπαθήσουμε μέσα από παραδείγματα να τονίσουμε ορθές και λανθασμένες επιλογές. Συχνά τα παραδείγματα που παρουσιάζονται δε θα έχουν εφαρμογή μόνο στο περιβάλλον στο οποίο παρουσιάζονται αλλά και σε άλλα περιβάλλοντα. Πρέπει να επισημανθεί ότι σε αρκετές περιπτώσεις δεν υπάρχει μόνο μια αλλά περισσότερες εναλλακτικές λύσεις. Στα πλαίσια του περιορισμένου χώρου του κεφαλαίου θα παρουσιάζονται αυτές που θεωρούνται βέλτιστες, χωρίς όμως να σημαίνει ότι δεν υπάρχουν και άλλες σχεδιαστικές προσεγγίσεις με εξίσου καλά αποτελέσματα.

## Κεφάλαιο 11      Μορφές Διαλόγου

### 11.1 Γραμμή εντολών

Στη μορφή διάδρασης της γραμμής εντολών ο χρήστης εισάγει εντολές μέσω πληκτρολογίου στον υπολογιστή. Ως τέτοιο περιβάλλον διάδρασης μπορούμε να αναφέρουμε το Unix και το MS DOS. Παρά τα μειονεκτήματα της γραμμής εντολών, δεν προβλέπεται να εκλείψει σύντομα σαν μορφή διάδρασης γιατί έχει και πολλά πλεονεκτήματα.

Πλεονεκτήματα	Μειονεκτήματα
Ευελιξία - κάθε εντολή έχει παραμέτρους	Δυσκολία εκμάθησης
Αμεσότητα προσπέλασης στις λειτουργίες του συστήματος - όλες οι εντολές σε επίπεδη δομή	Απαίτηση επανάκτησης από τη μνήμη
Ευκολία επικοινωνίας από απομακρυσμένο περιβάλλον	Επαύξηση των παραπάνω λόγω της μη σωστής ονομασίας εντολών, συνήθως με βάση τον προγραμματιστή και όχι τον χρήστη, π.χ., biff
Ευκολία χρήσης για επαναλαμβανόμενες εργασίες καθώς και για έμπειρους χρήστες	Απουσία ανάδρασης

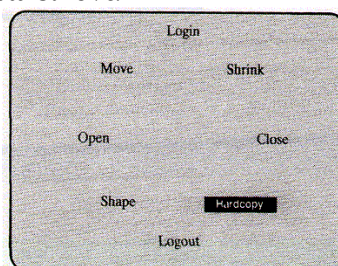
### 11.2 Επιλογείς (μενού εντολών)

Στη μορφή διάδρασης με επιλογείς, ο χρήστης επιλέγει την εντολή που θέλει ανάμεσα σε μία δοσμένη λίστα εντολών. Παράδειγμα τέτοιου περιβάλλοντος αποτελούν εφαρμογές όπως το PC-Tools (όπου ο χρήστης μπορούσε να εκτελέσει εντολές του DOS επιλέγοντας μέσα από μενού εντολών) ή το Edit (ο κειμενογράφος) του DOS.

Πλεονεκτήματα	Μειονεκτήματα
Απαίτηση αναγνώρισης και όχι επανάκτησης	Έλλειψη ευελιξίας της γραμμής εντολών
Ευκολία χρήσης όταν είναι απεικονισμένες, ονοματισμένες, και οργανωμένες σωστά οι επιλογές	Προβληματική η δημιουργία "προγραμμάτων" για επανάληψη ενεργειών

Υπάρχουν αρκετοί τύποι επιλογέων:

- pull-down (πτυσσόμενος): τραβιέται κάτω από μία "μόνιμη" εντολή στην κορυφή, ενώ σε πολλά συστήματα εξαφανίζεται αυτόματα
- pop-up (αναδυόμενος): εμφανίζεται στο σημείο που έχει υποδειχθεί, εκεί που ήταν κάποια εικόνα συνήθως, ενώ εξαφανίζεται με συγκεκριμένη εντολή ΚΛΕΙΣΙΜΟ του χρήστη
- pie (κυκλικός): εμφανίζεται στο σημείο που έχει υποδειχθεί, αλλά με τις επιλογές κυκλικά, όπως φαίνεται στην παρακάτω εικόνα



Οι επιλογές ενός επιλογέα μπορούν να οργανωθούν με διάφορους τρόπους: αλφαβητικά, με βάση κατηγορίες (νοηματικά), σειριακά (δεν είναι πολύ συνηθισμένο), με βάση συχνότητα χρήσης. Εντολές με αντίθετο αποτέλεσμα τοποθετούνται μακριά ή μία από την άλλη (π.χ., ΑΠΟΘΗΚΕΥΣΗ / ΔΙΑΓΡΑΦΗ). Τα παρακάτω παραδείγματα από το περιβάλλον ενός ιατρείου είναι ενδεικτικά:

- οι πελάτες ενός ιατρείου μάλλον απαιτούν αλφαβητική οργάνωση,
- το πρόγραμμα ραντεβού του γιατρού μάλλον απαιτεί σειριακή (χρονολογική) οργάνωση,
- τα εργαλεία του γιατρού μάλλον απαιτούν συχνοτική οργάνωση.

Οι επιλογείς είναι συνήθως ιεραρχικοί και αυτό δυσκολεύει την εύρεση κάποιων εντολών. Κάθε επιλογή έχει συνήθως μια συντομογραφία πληκτρολόγησης, αλλά συνήθως δεν μπορεί να χρησιμοποιηθεί εάν ο επιλογέας είναι εμφανισμένος στην οθόνη, στοιχείο ανειλικρινούς διεπαφής.

### 11.3 Φυσική γλώσσα

Σε αυτή τη μορφή διάδρασης ο χρήστης επικοινωνεί με τον υπολογιστή χρησιμοποιώντας φυσική γλώσσα, την οποία μπορεί να εισάγει στον υπολογιστή είτε μέσω του πληκτρολογίου είτε με κάποιο σύστημα αναγνώρισης φωνής.

Πλεονεκτήματα	Μειονεκτήματα
Μεγάλη ευελιξία και ευκολία για τον χρήστη	Δυσκολία υλοποίησης, περιορισμένο λεξιλόγιο και αντίστοιχα περιορισμένες χρήσεις

Η χρήση φυσικής γλώσσας βρίσκεται μακριά από υψηλά σημεία απόδοσης προς το παρόν, και δεν έχει φτάσει εκεί που απαιτείται για γενικές εφαρμογές. Και να φτάσει πάντως, η φυσική γλώσσα αυτή καθ' εαυτή είναι αμφίσημη και επομένως δεν μπορεί να εξυπηρετήσει απόλυτα το συνήθως αυστηρό περιβάλλον ενός υπολογιστή. Σαν παραδείγματα αμφισημίας, σκεφθείτε τα εξής δύο:

- *κτύπησε το παιδί με το ξύλο*: δύο ερμηνείες
  - το παιδί που χτυπήθηκε κράταγε ένα ξύλο
  - το όργανο του χτυπήματος του παιδιού ήταν ένα ξύλο
- *time flies like an arrow*: έχουν μετρηθεί περίπου 16 ερμηνείες λόγω των πολλών ερμηνειών που μπορούν να δοθούν στις διάφορες λέξεις της φράσης, όπως
  - ο χρόνος πετάει/τρέχει/φεύγει σαν ένα βέλος
  - στις χρονικές μύγες (οι χρονόμυγες!) αρέσει ένα βέλος
  - να χρονομετράς μύγες, να σου αρέσει ένα βέλος
  - χρονομέτρα μύγες όπως ένα βέλος

Με βάση την σημερινή τεχνολογία αλλά και λόγω των παραπάνω δυσκολιών, απέχουμε πολύ ακόμη από τον υπολογιστή των τηλεοπτικών σειρών επιστημονικής φαντασίας, όπως το «Ταξίδι στ' Αστέρια» («Star Trek»).

### 11.4 Ερωταπαντήσεις (A - Questions and Answers) και επερωτήσεις (B - Queries)

Σε αυτές τις μορφές διάδρασης, ο χρήστης επικοινωνεί με τον υπολογιστή θέτοντας ερωτήσεις. Οι μορφές αυτές διάδρασης εφαρμόζονται σε περιορισμένα περιβάλλοντα. Οι ερωταπαντήσεις χρησιμοποιούνται συνήθως όπου απαιτείται ανταλλαγή πληροφορίας μικρού μεγέθους, π.χ., ένα Ναι ή ένα Όχι, κάποιος κωδικός αριθμός κάποιου προϊόντος, κτλ. Οι επερωτήσεις είναι παρόμοιες με τις ερωταπαντήσεις αλλά έχουν την δυνατότητα να είναι πιο πολύπλοκες. Χαρακτηριστικό παράδειγμα χρήσης επερωτήσεων είναι οι βάσεις δεδομένων όπου χρησιμοποιείται η γλώσσα επερωτήσεων SQL.

<b>Πλεονεκτήματα</b>	<b>Μειονεκτήματα</b>
(Α) Ευκολία εκμάθησης, οπότε χρησιμότητα για άπειρους χρήστες	(Α+Β) Περιορισμένη εφαρμοστικότητα σε ειδικά περιβάλλοντα
(Β) Υψηλή εκφραστικότητα ερωτημάτων	(Β) Απαίτηση εμπειρίας στη σύνταξή τους
	(Β) Απουσία ανάδρασης (συνήθως)

### 11.5 Φόρμες και φύλλα υπολογισμού

Σε αυτή τη μορφή διάδρασης, ο χρήστης εισάγει στοιχεία σε φόρμες ή σε κελιά φύλλων υπολογισμού (ή λογιστικών φύλλων - spreadsheets). Η χρήση και των δύο είναι κυρίως για εισαγωγή δεδομένων αλλά και για την εμφάνιση των αποτελεσμάτων των ερωτήσεων. Και οι δύο μορφές διάδρασης βασίζονται στην παρουσίαση πληροφοριών, ερωτήσεων, και κενών προς συμπλήρωση με τρόπο παρόμοιο με τα αντίστοιχα χαρτικά έγγραφα. Παραδείγματα τέτοιων τύπων διάδρασης είναι οι εφαρμογές τύπου Lotus 1-2-3 ή Excel, καθώς και εφαρμογές όπου ο χρήστης συμπληρώνει στοιχεία σε προδιατεταγμένα σημεία της οθόνης (σελίδες εγγραφής σε υπηρεσίες στο διαδίκτυο, φόρμες της εφορίας, της τράπεζας, κτλ.).

<b>Πλεονεκτήματα</b>	<b>Μειονεκτήματα</b>
Δυνατότητα αυστηρού ελέγχου των εισαγόμενων δεδομένων	Περιορισμένη εφαρμοστικότητα σε ειδικά περιβάλλοντα
Αυτόματη τοποθέτηση κέρσορα στο σωστό σημείο	
Ευκολία εκμάθησης για τον άπειρο χρήστη λόγω της ομοιότητας με τα αντίστοιχα χάρτινα έγγραφα	
Καταλληλότητα και για έμπειρους χρήστες, με τον κατάλληλο σχεδιασμό	

### 11.6 3-διάστατες διεπαφές

Σε αυτή τη μορφή διάδρασης περιλαμβάνονται από την απλή σκίαση αντικειμένων για την δημιουργία εντυπώσεων μέχρι την προοπτική παρουσίαση αντικειμένων και τελικά μέχρι την εικονική πραγματικότητα. Όπου υπεισέρχεται θέμα σκιαγράφησης, υπάρχει πάντα η σιωπηρή συνθήκη ότι το φως είναι επάνω δεξιά.

<b>Πλεονεκτήματα</b>	<b>Μειονεκτήματα</b>
Οικονομία χώρου όταν γίνεται σωστή εκμετάλλευση των τριών διαστάσεων	Δυσκολία χρήσης (αποπροσανατολισμός) όταν υπάρχει υπερβολική ελευθερία στις τρεις διαστάσεις

### 11.7 Απευθείας διαχείριση

Η απευθείας διαχείριση (direct manipulation) επιτρέπει στον χρήστη να επικοινωνεί με τον υπολογιστή μέσω μιας διεπαφής που επιτρέπει τον άμεσο χειρισμό εικονιδίων που αναπαριστούν αντικείμενα οικεία στο χρήστη από τον πραγματικό κόσμο. Με αυτόν τον τρόπο η διεπαφή προσφέρει στο χρήστη ένα οικείο περιβάλλον για την αλληλεπίδρασή του με τον υπολογιστή. Οι ενέργειες του χρήστη είναι γρήγορες, αναστρέψιμες, και πιθανόν κλιμακωτές. Ενέργειες που υλοποιούν απευθείας διαχείριση είναι η "δείχνω και πιέζω" ("point and click"), η "διπλοπιέζω", η "σύρω", κτλ. Η απευθείας διαχείριση είναι στενά συνδεδεμένη με την μορφή διάδρασης ΠΕΠΕ

(βλέπε παρακάτω) αλλά όχι απαραίτητα. Μπορεί να εμφανίζεται σε συνδυασμό με πολλών ειδών μονάδων εισόδου, όπως οι συνηθισμένες οθόνες, οι ευαίσθητες οθόνες, και άλλες.. Το μόνο πιθανό πρόβλημα της απευθείας διαχείρισης είναι οι διαφορές που είναι δυνατό να υπάρξουν ανάμεσα στο μοντέλο του χρήστη στο οποίο βασίστηκε η υλοποίηση της διεπαφής και στους πραγματικούς χρήστες. Οι διαφορές αυτές εάν είναι πολλές μπορούν να προκαλέσουν την αδυναμία κατανόησης της διεπαφής από το χρήστη και τη δημιουργία ενός χάσματος μεταξύ χρήστη και συστήματος (χάσμα εκτέλεσης και χάσμα αξιολόγησης).

## 11.8 ΠΕΠΕ (Παράθυρα, Εικονίδια, Ποντίκια, Επιλογείς)

Το αγγλικό ακρώνυμο για το ΠΕΠΕ είναι WIMP το οποίο επιδέχεται δύο ισοδύναμες ερμηνείες: "Windows, Icons, Mice, Pull-down menus" (που αντιστοιχεί στην Ελληνική μετάφραση) αλλά και "Windows, Icons, Menus, Pointers". Η ΠΕΠΕ είναι η βασική μορφή διάδρασης των σύγχρονων υπολογιστών και αποτελεί υποσύνολο των λεγομένων "γραφικών διεπαφών χρήστη" ("graphical user interfaces"). Σε αυτή τη μορφή διάδρασης, ο χρήστης επικοινωνεί με τον υπολογιστή χρησιμοποιώντας (κατά την ονομασία της διάδρασης) παράθυρα, εικονίδια, ποντίκια, επιλογείς, αλλά και επιπρόσθετα εργαλεία, όπως (εικονικά) πλήκτρα, ράβδοι εργαλείων, παλέτες, πλαίσια διαλόγων, και άλλα. Αν και όλα τα χαρακτηριστικά όλων παραπάνω είναι λίγο-πολύ γνωστά, αναφέρουμε παρακάτω επιγραμματικά τα πιο σημαντικά από αυτά.

- *Παράθυρα*: Υπάρχουν πολλών ειδών παράθυρα, επικαλυπτόμενα, μη επικαλυπτόμενα (tiled), στοιβαγμένα (cascading, όπου κάθε νέο επίπεδο είναι λίγο κάτω και δεξιά από το προηγούμενο), ελαστικά (όπου το περίγραμμα του ενός σπρώχνει το περίγραμμα του άλλου καθώς αυξομειώνεται το μέγεθός του), και άλλα. Συχνά συνοδεύονται από ράβδους ολίσθησης (scrollbar).

Όταν απαιτούνται πολλά παράθυρα για την εκτέλεση μιας εργασίας συχνά δημιουργείται οργανωτικό πρόβλημα (overhead) για την οθόνη. Για την μελέτη τέτοιων περιπτώσεων είναι χρήσιμη η έννοια του "συνόλου παραθύρων εργασίας" (window working set), που ορίζεται σαν το σύνολο των παραθύρων που απαιτούνται για την διεκπεραίωση μιας εργασίας. Αν το σύνολο παραθύρων εργασίας δεν χωρά στην οθόνη χωρίς επικαλύψεις, τότε υπάρχει πρόβλημα στον σχεδιασμό. Για την αντιμετώπιση αυτού του προβλήματος, αρκετά συστήματα ανοίγουν και κλείνουν παράθυρα αυτόματα με βάση την ροή της εργασίας του χρήστη.

Για την απεικόνιση ακμών και άλλων σχημάτων σε παράθυρα άλλοτε γίνεται χρήση bitmaps και άλλοτε μαθηματικών εκφράσεων (σπλήνες – splines). Οι ψηφιογραφικές εικόνες (bitmaps) απεικονίζονται πολύ γρήγορα αλλά δύσκολα μεγεθύνονται, στρέφονται, κτλ. Το ακριβώς αντίθετο συμβαίνει με τις μαθηματικές εκφράσεις (π.χ., Postscript).

- *Εικονίδια*: Τα βασικά σημεία των εικονιδίων έχουν ήδη συζητηθεί στο 2ο Κεφάλαιο, στη συζήτηση για την ανθρώπινη μνήμη. Την μοναδική προσθήκη που θα θέλαμε να κάνουμε σε αυτό το σημείο είναι ότι τα εικονίδια μπορούν να χρησιμοποιηθούν και για σμίκρυνση παραθύρων.
- *Δείκτες (Ποντίκια)*: Η λειτουργία των ποντικιών είναι γνωστή και δεν χρειάζεται ιδιαίτερες επεξηγήσεις. Για πληρότητα και μόνο να αναφέρουμε ότι υπάρχουν πέντε βασικές πράξεις ποντικών: δείχνω, πιέζω, διπλοπιέζω, πιέζω συνεχώς, και σύρω. Το σύνολο αυτό μπορεί να επαυξηθεί όταν οι αντίστοιχες πράξεις του ποντικιού συνοδεύονται με ταυτόχρονη πίεση κάποιου πλήκτρου ελέγχου. Τέλος, να αναφέρουμε ότι υπάρχουν πολλές διαφορετικές μορφές δεικτών που υποδεικνύουν την θέση των ποντικιού, π.χ., βέλη για επιλογές, κάθετες γραμμές για κείμενο, ρολόγια για ένδειξη χρήσης, κλεψύδρες για ένδειξη προόδου, και άλλες.

- *Επιλογείς*: Οι επιλογείς αναλύθηκαν παραπάνω και σαν αυτούσια μορφή διάδρασης ανεξάρτητα από την προσέγγιση ΠΕΠΕ, οπότε δεν θα συζητηθούν ξανά εδώ.
- *Πλήκτρα*: Υπάρχουν αρκετά διαφορετικά είδη πλήκτρων με διαφορετική χρησιμότητα το καθένα, π.χ., άλλα για εναλλαγή καταστάσεων (toggle), άλλα για εκτέλεση εντολής, κτλ. Σε περιπτώσεις χρήσης πλήκτρων για τον καθορισμό μίας ή περισσότερων επιλογών/καταστάσεων από ένα σύνολο, όταν οι επιλογές/ καταστάσεις είναι αμοιβαία ξένες μεταξύ τους, τότε τα *ραδιοπλήκτρα* αποτελούν ένα σωστό τύπο πλήκτρου, ενώ όταν δεν είναι ξένες μεταξύ τους, ο σωστός τύπος είναι τα *πλήκτρα ελέγχου*.
- *Ράβδοι εργαλείων (toolbars)*: Χρησιμοποιούνται για να δώσουν γρήγορους τρόπους επίκλησης εντολών/επιλογών που χρησιμοποιούνται συχνά. Κάθε εργαλείο αναπαριστάται στην ράβδο από κάποιο εικονίδιο. Εν γένει, τα εικονίδια αυτά δεν δημιουργούν πάντα άμεσα τους σωστούς συνειρμούς για να αναγνωρίζει ο χρήστης τα αντίστοιχα εργαλεία. Κατ' αυτόν τον τρόπο, στις πρώτες φορές χρήσης του επιλογέα βλέπει ο χρήστης ταυτόχρονα και το εικονίδιο και την εντολή, κι έτσι μαθαίνει την σημασία του εικονιδίου. Το αποτέλεσμα είναι ότι μετά από κάποιο μικρό χρονικό διάστημα, ο χρήστης μπορεί να κάνει χρήση της ράβδου εργαλείων κατ' ευθείαν και η χρήση του επιλογέα δεν είναι αναγκαία (για τα εργαλεία που εμφανίζονται στην ράβδο). Αυτό που συμβαίνει είναι ότι εμφανίζεται για μερικά δευτερόλεπτα ένα πλαίσιο συμβουλών με την αντίστοιχη εντολή όταν μεταφέρει κανείς το δείκτη του ποντικιού πάνω στο εικονίδιο. Εναλλακτικά μπορεί να χρησιμοποιηθούν εικονίδια μαζί με κείμενο όπως στο Netscape.
- *Παλέτες*: Μία παλέτα είναι ένα σύνολο εικονιδίων που επιτρέπουν στον χρήστη να καθορίσει (και μετά να γνωρίζει συνεχώς) την κατάσταση διάδρασης (interaction mode) στην οποία βρίσκεται το σύστημα, π.χ., μία από τις δύο καταστάσεις του editor vi στο Unix με βάση την οποία ερμηνεύονται οι επόμενες πληκτρολογήσεις, ένα από πολλά διαθέσιμα χρώματα ενός πακέτου γραφημάτων με το οποίο θα χρωματίζονται τα αμέσως επόμενα σχήματα, και άλλα.
- *Πλαίσια διαλόγων*: Είναι χρήσιμα για εισαγωγή αλφαβητικών (ή και αριθμητικών) δεδομένων όταν δεν υπάρχουν περιορισμοί στο σύνολο των πιθανών τιμών που μπορούν να εισαχθούν (οπότε η χρήση επιλογέων είναι αδύνατη). Σημαντικό πλεονέκτημα των πλαισίων είναι ότι η εμφάνισή τους προσελκύει την προσοχή του χρήστη και έτσι αποφεύγονται προβλήματα και καθυστερήσεις. Και εδώ υπάρχουν πολλές μορφές τέτοιων πλαισίων:
  - Πλαίσια υποχρεωτικού διαλόγου (modal)
  - Πλαίσια προαιρετικά διαλόγου (modeless)
  - Πλαίσια επερωτήσεων - τα εμφανίζει το σύστημα
  - Πλαίσια μηνυμάτων - συνήθως ένδειξη σφαλμάτων

---

Ερώτηση: Κατά την ανάπτυξη μιας διεπαφής είναι δυνατή η συνύπαρξη περισσότερων από ένα από τους τύπους διαλόγου που παρουσιάστηκαν. Μπορείτε να αναγνωρίσετε ποιοι τύποι διαλόγων χρησιμοποιούνται στα Windows, Unix, Linux, MS Excel, Oracle, MS Access;

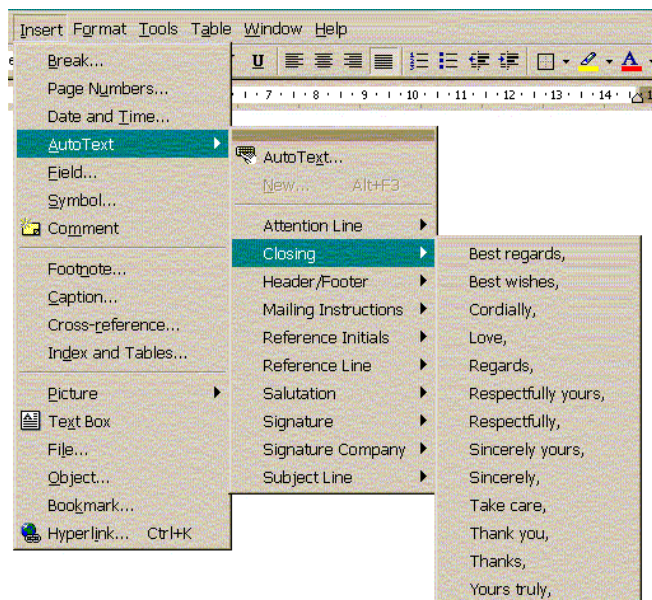
---

## Κεφάλαιο 12 Σχεδιασμός σε περιβάλλον ΠΕΠΕ

Σε ένα περιβάλλον ΠΕΠΕ τα βασικά 'συστατικά' της διεπαφής είναι τα παράθυρα, τα εικονίδια, το ποντίκι και οι επιλογείς. Για τα συστατικά αυτά θα παραθέσουμε μια περιορισμένη παρουσίαση των χαρακτηριστικών τους και των σημείων που θα πρέπει να προσεχθούν κατά το σχεδιασμό. Ειδικά για τα παράθυρα θα δώσουμε έμφαση στα πλαίσια διαλόγου, δηλαδή σε παράθυρα που υλοποιούν διαλόγους χρήστη – υπολογιστή.

### 12.1 Επιλογείς

Η χρήση των επιλογέων είναι ένα από τα χαρακτηριστικά της διεπαφής τύπου ΠΕΠΕ. Ο επιλογέας εμφανίζεται και σε άλλους τύπους διεπαφών και αποτελεί μια ομάδα ή λίστα εντολών ή λειτουργιών τις οποίες μπορεί να επιλέξει και να εκτελέσει ο χρήστης. Επειδή η παράθεση όλων των εντολών σε μια μακροσκελή λίστα θα έκανε έναν επιλογέα αναποτελεσματικό δυσκολεύοντας το έργο της ανεύρεσης μιας εντολής, ένα επιλογέας μπορεί να έχει περισσότερα από ένα επίπεδα. Με αυτόν τον τρόπο οι εντολές ομαδοποιούνται και ο επιλογέας αποκτά υπο-επιλογείς, όπως φαίνεται στην εικόνα.



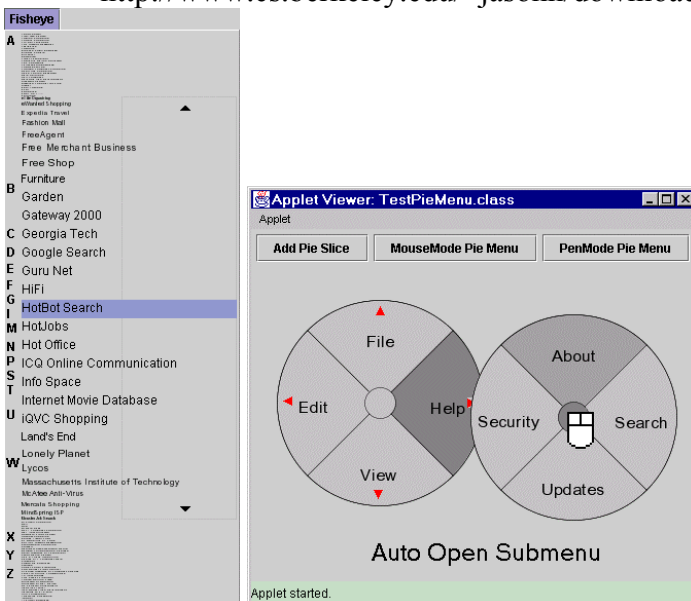
Εικόνα 1: Παράδειγμα επιλογέα πολλών επιπέδων

Ανάλογα με τον τρόπο που εμφανίζονται οι επιλογείς μπορούν να χωριστούν σε πτυσσόμενους (pull-down) και αναδυόμενους (pop-up). Στην περίπτωση των πτυσσόμενων επιλογέων η ράβδος των επιλογέων είναι εμφανής και όταν ο χρήστης επιλέξει έναν από τους επιλογείς αυτός ανοίγει προς τα κάτω. Οι αναδυόμενοι επιλογείς εμφανίζονται συνήθως με το πάτημα του δεξιού πλήκτρου του ποντικιού και σχετίζονται πάντα με το αντικείμενο στο οποίο έδειχνε ο δείκτης του ποντικιού (ενώ οι πτυσσόμενοι σχετίζονται με τη λειτουργία). Η εμφάνισή τους κατ' απαίτηση του χρήστη δίνει στον τελευταίο την αίσθηση του ελέγχου της επικοινωνίας και με δεδομένο ότι είναι σχετικοί με τα συμφραζόμενα απαλλάσσουν το χρήστη από περιττές πληροφορίες.

Το βάθος, δηλαδή ο αριθμός των επιπέδων ενός επιλογέα και το πλάτος, δηλαδή ο αριθμός των επιλογών σε κάθε επίπεδο, αποτελούν βασικά χαρακτηριστικά ενός επιλογέα. Ένας μεγάλος αριθμός ερευνητικών εργασιών ([Kiger 1984], [Lanfauer and Nachbar 1985], [Norman and Chin 1988], [Wallace et al. 1987]) έχουν εξετάσει τη σχέση μεταξύ βάθους και πλάτους ενός επιλογέα. Οι ερευνητικές αυτές εργασίες επισημαίνουν ότι και η σημασιολογική δομή ενός επιλογέα δεν πρέπει να παραβλέπεται, μικρότερος αριθμός επιπέδων βελτιώνει το χρόνο εύρεσης μιας επιλογής.

Οι ακόλουθες σχεδιαστικές οδηγίες αφορούν στη μορφή διαλόγων με χρήση Επιλογέων και ισχύουν γενικότερα χωρίς να προϋποθέτουν την ύπαρξη διεπαφής μορφής ΠΕΠΕ.

- Τα ονόματα των εντολών θα πρέπει να είναι εύκολα αναγνωρίσιμα.
- Η ονοματολογία θα πρέπει να είναι συνεπής (γραμματικά, από πλευράς όρων και εμφάνισης).
- Η δομή των επιλογέων θα πρέπει να αντιστοιχεί σε αυτήν των ενεργειών που πρέπει να εκτελέσει ο χρήστης (π.χ. Εισαγωγή -> Εικόνας -> Από αρχείο).
- Ο τρόπος επιλογής θα πρέπει να είναι εύκολος, να διατηρείται συνεπής και να απαιτεί μικρό χρόνο εκμάθησης από τον χρήστη.
- Θα πρέπει να παρέχεται μια συνεπής μέθοδος επιστροφής σε επιλογείς προηγούμενου επιπέδου ή αναίρεσης λανθασμένης επιλογής.
- Σε περιβάλλοντα τύπου ΠΕΠΕ, και για επιλογές που εμφανίζονται και στην ράβδο εικονιδίων, είναι χρήσιμο να εμφανίζεται το αντίστοιχο εικονίδιο πριν το όνομα της επιλογής.
- Περιορίστε κατά το δυνατό το πλήθος των επιλογών και το βάθος των υπο-επιλογέων. Προτιμήστε να αναπτυχθείτε σε πλάτος παρά σε βάθος.
- Προσφέρετε πλήκτρα συντόμευσης για τις πιο συχνά χρησιμοποιούμενες επιλογές.
- Μη ξεχνάτε τον κανόνα του Fitt. Υπάρχουν εναλλακτικοί τύποι επιλογέων που βασίζονται σε αυτόν τον κανόνα (κυκλικοί επιλογείς ή επιλογείς με μεταβλητό μέγεθος επιλογών). Παραδείγματα αποτελούν οι επόμενες εικόνες. Δείτε και στο <http://www.cs.umd.edu/hcil/fisheymenu/>, <http://www.cs.berkeley.edu/~jasonh/download/software/piemenu/>.



## 12.2 Εικονίδια

Σε ένα WIMP τα εικονίδια χρησιμοποιούνται για να αναπαραστήσουν αντικείμενα του περιβάλλοντος εργασίας του χρήστη. Με αυτόν τον τρόπο ένα εικονίδιο μπορεί να αναπαραστήσει ένα λογισμικό προϊόν που βρίσκεται εγκατεστημένο στο περιβάλλον εργασίας, ένα αρχείο ή ένα παράθυρο που έχει ελαχιστοποιηθεί για να μην καταλαμβάνει χώρο. Τα εικονίδια που χρησιμοποιούνται μπορεί να αποτελούν μεταφορές συνηθισμένων στο χρήστη αντικειμένων όπως το καλάθι αχρήστων ή το γραμματοκιβώτιο. Έτσι ο χρήστης μπορεί να αντιληφθεί πιο εύκολα τη χρήση τους. Εάν για παράδειγμα θέλει να απορρίψει κάποιο αρχείο θα το βάλει στο καλάθι αχρήστων, ενώ εάν θέλει να δει το ηλεκτρονικό ταχυδρομείο θα κοιτάξει στο γραμματοκιβώτιο.



Η πληθώρα εικονιδίων μπορεί να δημιουργήσει προβλήματα στους χρήστες, μια και θα είναι δύσκολο να τα απομνημονεύσουν και να κατανοήσουν τη σημασία τους. Πρόσθετα προβλήματα μπορούν να δημιουργηθούν και λόγω πολιτισμικών διαφορών. Ως παράδειγμα μπορούμε να αναφέρουμε την περίπτωση του Βρετανικού γραμματοκιβωτίου που σε διαφορετικό πολιτισμικό περιβάλλον μπορεί να παρερμηνευτεί ως απορριμματοδέκτης. Αντίστοιχα η επόμενη εικόνα που χρησιμοποιείται συχνά ως απαγορευτικό σύμβολο σε μηνύματα λάθους, είναι πολύ πιθανό πως θα θεωρηθεί προσβλητική από τους Έλληνες χρήστες.



**Εικόνα 2:** Απαγορευτικό σύμβολο

Επιπλέον πρόβλημα δημιουργείται με τη χρήση εικονιδίων που έχουν μεταφορική σημασία. Για παράδειγμα, τα εικονίδια στην επόμενη εικόνα φαίνεται πως έχουν συγγενική σημασία, όμως στην πράξη αντιστοιχούν σε αρκετά διαφορετικές λειτουργίες. Τα κιάλια στην εικόνα (α) εμφανίζονται στο MS Word σε εικονίδιο που αντιστοιχεί στη λειτουργία ανεύρεσης ενός συγκεκριμένου κειμένου ή μορφοποίησης. Τα γυαλιά στην εικόνα (β) εμφανίζονται στο ίδιο λογισμικό προϊόν σε εικονίδιο που αντιστοιχεί στη λειτουργία εμφάνισης λίστας με τις μεταβλητές της ενεργής μακροεντολής.



**Εικόνα 3:** Εικονίδια από τη ράβδο εργαλείων του MS Word

Δεν υπάρχει κάποιος εμφανής λόγος για τον οποίο τα κιάλια θα πρέπει να συσχετιστούν με τη διαδικασία 'έρευνας' και όχι με τη διαδικασία 'αποκάλυψης' ή 'εμφάνισης' που έχουν τα γυαλιά (και το αντίστροφο). Ανάλογα, υπάρχουν πολλές λειτουργίες που δεν είναι δυνατό να αναπαρασταθούν με εικονίδια παρά μόνο με τη χρήση μεταφορών. Τέτοια όμως εικονίδια αφήνουν περιθώρια για παρερμηνεία από διαφορετικούς χρήστες. Γι' αυτόν το λόγο και αρκετές εφαρμογές 'προσφέρουν' μαζί με το εικονίδιο και κείμενο που χαρακτηρίζει τη λειτουργία που απεικονίζει το εικονίδιο, όπως φαίνεται στην επόμενη εικόνα.



**Εικόνα 4:** Εικονίδια με κείμενο

Σε αυτή τη λύση συνηγορεί και το γεγονός ότι οι χρήστες που είχαν στη διάθεσή τους κατά την αλληλεπίδρασή τους με εφαρμογές μόνο εικονίδια, εργάζονταν πιο αργά από χρήστες που είχαν είτε μόνο κείμενο είτε κείμενο και εικόνα [13].

Εικονίδια

- Οι ενέργειες που απεικονίζουν τα εικονίδια πρέπει να είναι άμεσα αναγνωρίσιμες.
- Προσφέρετε στους χρήστες τη δυνατότητα επιλογής εικονιδίων.
- Προσφέρετε επεξηγηματικό κείμενο για τα εικονίδια.
- Δοκιμάζετε τα εικονίδια σε διαφορετικές αναλύσεις.
- Τα επιλεγμένα εικονίδια πρέπει να ξεχωρίζουν εύκολα από τα μη επιλεγμένα.

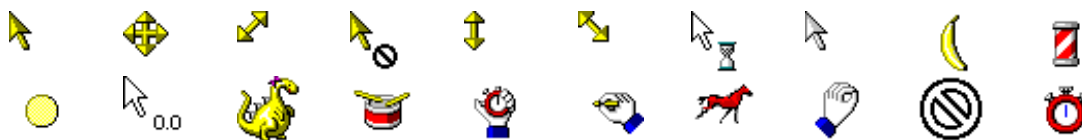
- Τα εικονίδια που αντιστοιχούν σε απενεργοποιημένες εντολές πρέπει να ξεχωρίζουν από τα υπόλοιπα.

### 12.3 Ποντίκι

Ο προσφορότερος τρόπος για να εκτελέσει ο χρήστης κάποια ενέργεια είναι η συσκευή κατάδειξης, η οποία είναι συνήθως -αλλά όχι κατ' ανάγκη- το ποντίκι. Η συσκευή κατάδειξης μετακινεί κάποιο δείκτη που εμφανίζεται στην οθόνη του υπολογιστή του χρήστη. Μετακινώντας το δείκτη και χρησιμοποιώντας τα πλήκτρα της συσκευής κατάδειξης ο χρήστης είναι σε θέση να εκτελέσει διάφορες ενέργειες όπως να επιλέξει αντικείμενα, να τα σύρει στην οθόνη, να τροποποιήσει το σχήμα τους, κλπ. Οι ενέργειες που μπορεί να εκτελέσει ο χρήστης εξαρτώνται από το αντικείμενο.

Παρότι ο χρήστης έχει συνήθως τη δυνατότητα να επιλέξει και να εκτελέσει εντολές από ένα μενού χρησιμοποιώντας το πληκτρολόγιο, η συσκευή κατάδειξης αποτελεί τον κύριο τρόπο διαχείρισης των εντολών των μενού.

Ο δείκτης του ποντικιού μπορεί να χρησιμοποιηθεί για να μεταφέρει πληροφορίες στο χρήστη. Εάν για παράδειγμα έχει σχήμα κλεψύδρας, ο χρήστης καταλαβαίνει ότι η εφαρμογή είναι απασχολημένη με κάποια λειτουργία. Μερικοί δείκτες εμφανίζονται στην επόμενη εικόνα:



Εικόνα 5: Εικονίδια δεικτών στα MS Windows

### 12.4 Παράθυρα – Πλαίσια Διαλόγου

Έχουμε ήδη αναφέρει οδηγίες που αφορούν στους επιλογείς και τα εικονίδια. Στη συνέχεια θα αναφέρουμε οδηγίες σχετικές με τα παράθυρα.

#### Παράθυρα

- Τίτλος παραθύρου που να έχει σχέση με τα περιεχόμενά του. Ειδικά για πλαίσια διαλόγου ο τίτλος του παραθύρου μπορεί να σχετίζεται με τη λειτουργία που αυτό υλοποιεί.
- Η ορολογία, οι γραμματοσειρές, τα χρώματα και η γενικότερη εμφάνιση των παραθύρων θα πρέπει να είναι συνεπής τόσο με τα υπόλοιπα παράθυρα της εφαρμογής όσο και με το περιβάλλον στο οποίο εκτελείται η εφαρμογή (MS Windows, Macintosh, X Windows). Συμβουλευτείτε τις οδηγίες των κατασκευαστών του περιβάλλοντος στο οποίο προγραμματίζετε.
- Χρησιμοποιείτε πλαίσια υποχρεωτικού διαλόγου μόνο όταν είναι απαραίτητο.
- Προσφέρετε στους χρήστες εύκολο τρόπο πλοήγησης και διαχείρισης των παραθύρων.
- Φροντίστε ώστε τα παράθυρα να εμφανίζονται σωστά σε διαφορετικές αναλύσεις οθόνης. Προσφέρετε στο χρήστη τη δυνατότητα τροποποίησης του μεγέθους των παραθύρων.
- Εξασφαλίστε ότι στα πλαίσια διαλόγου δεν υπάρχουν αντικείμενα που είναι απαραίτητα στο χρήστη τα οποία αποκρύπτονται ή επικαλύπτονται.

#### Χειριστήρια

Σε ένα παραθυρικό περιβάλλον υπάρχουν διάφορα αντικείμενα που μπορεί να χρησιμοποιήσει κάποιος σχεδιαστής λογισμικού για να προσφέρει τη λειτουργικότητα της εφαρμογής του στους χρήστες. Τα αντικείμενα αυτά επιτρέπουν στους χρήστες το χειρισμό της λειτουργικότητας μιας εφαρμογής ονομάζονται *χειριστήρια* (controls). Τα πιο συνηθισμένα χειριστήρια είναι τα ακόλουθα:

- Ετικέτα (label)
- Πλαίσιο κειμένου (text box)

- Πλαίσιο επιλογής (check box)
- Ραδιοπλήκτρο ή πλήκτρο επιλογής (radio button ή option button)
- Πλαίσιο πτυσσόμενης λίστας (combo box)
- Ράβδος ολίσθησης (scroll bar)
- Πλήκτρο εντολής (command button)
- Πλαίσιο λίστας (list box)
- Πλαίσιο διαλόγου (dialog box)

#### Παράδειγμα 1: Φόρμες εισαγωγής στοιχείων

Σημεία που πρέπει να προσεχθούν: εμπειρία και συνήθειες χρηστών, συμπληρωματικότητα μεθόδων, ημερομηνίες, δυνατότητα επιλογής διαφορετικών μονάδων

Οι φόρμες αποτελούν ίσως το πιο συνηθισμένο τρόπο εισαγωγής δεδομένων. Εάν το μηχανογραφικό σύστημα μεταφέρει ένα υπάρχον χειρογραφικό σύστημα σε ηλεκτρονική μορφή τότε είναι χρήσιμο να διατηρείται η ομοιότητα ώστε οι χρήστες να είναι εξοικειωμένοι (εικόνα από την ηλεκτρονική υποβολή φορολογίας εισοδήματος). Φυσικά η διατήρηση της ομοιότητας δεν συνεπάγεται και την αναπαραγωγή "δυσκαμνιών" του χειρογραφικού συστήματος. Εάν για παράδειγμα στο χειρογραφικό σύστημα υπάρχει ένα έντυπο στο οποίο οι χρήστες συμπληρώνουν, μεταξύ άλλων, την ημερομηνία υποβολής, τα αθροίσματα των ποσών σε δραχμές και τα αντίστοιχα ποσά σε ευρώ, τότε στην ηλεκτρονική μορφή του εντύπου αυτά τα πεδία θα πρέπει να συμπληρώνονται αυτόματα.

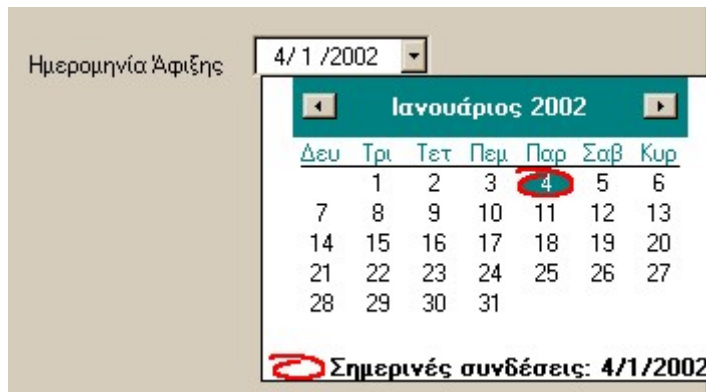
Ειδικά τα πεδία εισαγωγής ημερομηνιών αποτελούν ένα ενδιαφέρον παράδειγμα, καθώς υπάρχει ένα σύνολο από αντικείμενα με τα οποία μπορούν να υλοποιηθούν. Γενικά είναι επιθυμητή η ύπαρξη συμπληρωματικών (όπως εισαγωγή της ημερομηνίας είτε μέσω πλαισίου κειμένου, είτε με επιλογή από ένα ημερολόγιο) όχι όμως και επικαλυπτόμενων μεθόδων εισαγωγής δεδομένων (όπως εισαγωγή ημερομηνίας είτε μέσω πλαισίου κειμένου είτε μέσω τριών πλαισίων ένα για κάθε τμήμα της ημερομηνίας). Όπως θα μπορούσε να προβλεφθεί και με χρήση του μοντέλου GOMS η αύξηση του αριθμού των εναλλακτικών μεθόδων αυξάνει την πολυπλοκότητα της διεπαφής, των επιλογών του χρήστη, άρα και τόσο το χρόνο που θα χρειαστεί ένας χρήστης για να εκτελέσει μια εργασία (έναν επιπλέον τελεστής Μ κάθε φορά που πρέπει να επιλέξει μια μέθοδο) όσο και το χρόνο που εκμάθησης της διεπαφής. Κάτω από αυτό το πρίσμα είναι προτιμότερο να προσφέρουμε εμείς στο χρήστη μια μέθοδο που είναι αντικειμενικά η βέλτιστη, παρά να προσφέρουμε πολλαπλές μεθόδους για κάθε πιθανή ομάδα χρηστών. Στις επόμενες εικόνες δίνουμε μερικά παραδείγματα από μεθόδους εισαγωγής ημερομηνίας.

Ημερομηνία Άφιξης   
(HH/MM/YYYY)

Πλαίσιο εισαγωγής ημερομηνίας

Ημερομηνία Άφιξης

Πτυσσόμενο πλαίσιο εισαγωγής ημερομηνίας. Ο χρήστης μπορεί είτε να εισάγει την ημερομηνία όπως σε ένα απλό πλαίσιο κειμένου (όπου όμως οι πλάγιες μπάρες / δεν είναι δυνατό να σβηστούν), είτε να πιάσει στο κάτω βέλος και να εμφανιστεί ένα ημερολόγιο (επόμενη εικόνα).



Σημαντική είναι η σωστή χρήση των λεκτικών (ετικετών, μηνυμάτων, κλπ.), η παράθεση επεξηγήσεων όπου χρειάζεται και η εμφάνιση μηνυμάτων λανθασμένων δεδομένων με ταυτόχρονη μεταφορά της εστίασης στο πεδίο το οποίο προκάλεσε το μήνυμα λάθους. Επίσης προσοχή θα πρέπει να δίνεται και στο πολιτισμικό υπόβαθρο των χρηστών, ειδικά σε εφαρμογές που αναπτύσσονται στο διαδίκτυο. Για παράδειγμα, η ημερομηνία 3/1/2002 μπορεί να σημαίνει 3 Ιανουαρίου 2002 ή 1 Μαρτίου 2002, γι' αυτό και η περιφραστική αναγραφή της ημερομηνίας, όταν αυτό είναι δυνατό, είναι προτιμότερη.

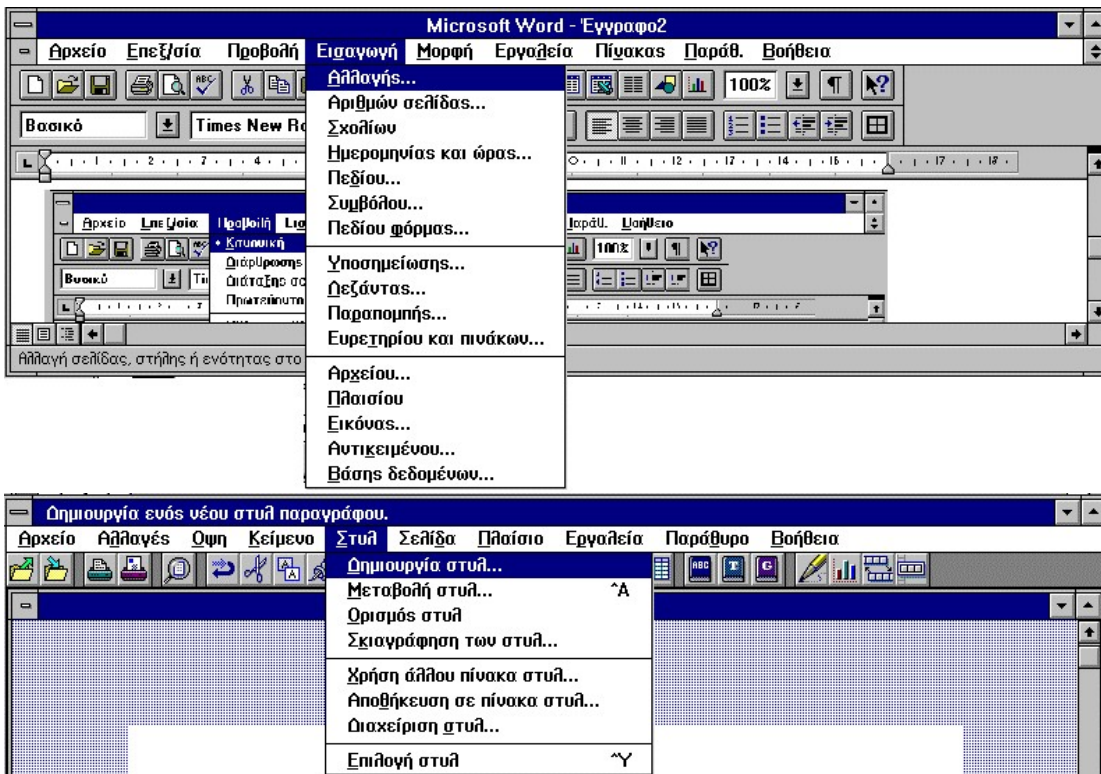
Συχνά ο χρήστης καλείται να εισάγει δεδομένα όπως χρηματικά ποσά ή μετρήσεις. Όπου είναι δυνατό θα πρέπει να παρέχεται η δυνατότητα χρήσης διαφορετικών μονάδων μέτρησης. Μια λύση είναι η χρήση ενός ραδιοπλήκτρου και ενός πλαισίου κειμένου όπου στο πρώτο επιλέγει τη μονάδα μέτρησης και στο δεύτερο εισάγει την αριθμητική τιμή, όπως φαίνεται στην επόμενη εικόνα.

Εναλλακτικά θα μπορούσε να χρησιμοποιηθεί ένα πλαίσιο κειμένου όπου ο χρήστης θα εισήγαγε την αριθμητική τιμή ακολουθούμενη από το σύμβολο της μονάδας μέτρησης (π.χ. cm για εκατοστά, " για ίντσες, E για Ευρώ, κλπ). Η δεύτερη λύση είναι υπερτερει όταν πρόκειται για έναν περιορισμένο (1 ή 2) αριθμό από πλαίσια στα οποία πρέπει να εισαχθούν αριθμητικές τιμές και όταν ο χρήστης είναι σχετικά έμπειρος, γιατί δε χρειάζεται να εναλλάσσει μεταξύ ποντικιού (για την επιλογή της μονάδας μέτρησης) και πληκτρολογίου (για την εισαγωγή της αριθμητικής τιμής).

#### Παράδειγμα 2: Σχεδιασμός επιλογών

Αριθμός επιπέδων και αριθμός επιλογών ανά επίπεδο. Εναλλακτικές μέθοδοι εκτέλεσης λειτουργιών

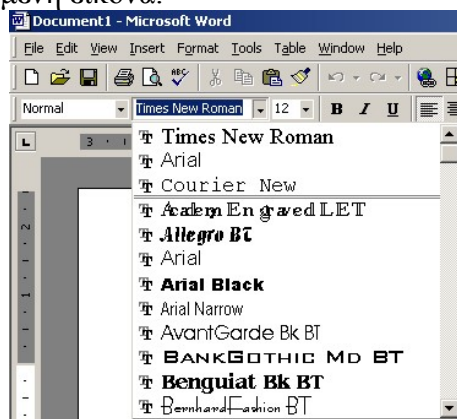
Όπως αναφέρθηκε και στο γενικό σχεδιασμό των επιλογών, οι επιλογές θα πρέπει να ομαδοποιούνται σε επιλογές με τρόπο οικείο ή τουλάχιστον λογικό για τους χρήστες. Για παράδειγμα, στο Microsoft Word, οι λειτουργίες αφορούν στη επεξεργασία κειμένου και ομαδοποιούνται ανάλογα τη γενικότερη εργασία που μπορεί να επιθυμεί να εκτελέσει ο χρήστης, όπως προβολή του εγγράφου, εισαγωγή ειδικών δεδομένων στο έγγραφο, μορφοποίηση του εγγράφου, κλπ. Η ομαδοποίηση αυτή δεν είναι μοναδική. Στις επόμενες δύο εικόνες παρουσιάζονται οι επιλογές από δύο προγράμματα επεξεργασίας κειμένου, του MS Word 6 και του Lotus Ami Pro.



Όπως φαίνεται και στις εικόνες η ομαδοποίηση των επιλογών δεν είναι ίδια.

Η ομαδοποίηση συνήθως ακολουθεί κάποιους απλούς αλλά αποτελεσματικούς κανόνες:

- Οι επιλογές είναι λογικά σχετιζόμενες. Στο παράδειγμα του επεξεργαστή κειμένου, ένας χρήστης αναμένει ότι στον επιλογέα *Πίνακας* θα βρει επιλογές που σχετίζονται με την διαχείριση (εισαγωγή, διαγραφή, τροποποίηση, ...) πινάκων μέσα στο έγγραφο. Επιλογές στον ίδιο επιλογέα μπορεί να διαχωρίζονται με μια γραμμή, ένδειξη ότι ανήκουν σε διαφορετική υπο-ομάδα.
- Οι επιλογές καλύπτουν όλες τις περιπτώσεις χωρίς να είναι επικαλυπτόμενες.
- Οι επιλογές μπορεί να παρουσιάζονται αλφαβητική, κατά σειρά σπουδαιότητας ή συχνότητα χρήσης. Η παρουσίαση κατά αλφαβητική σειρά αποτελεί έναν απλό και εύχρηστο τρόπο παρουσίασης καθώς είναι εύκολο για τους χρήστες να εντοπίσουν μια επιλογή. Επιλογείς που τροποποιούν την εμφάνισή τους δημιουργούν προβλήματα στους χρήστες (Mitchell and Shneiderman, 1988). Μια εναλλακτική προσέγγιση είναι η τοποθέτηση των τριών πιο συχνά χρησιμοποιούμενων επιλογών στην αρχή διαχωρισμένες από το σύνολο των επιλογών που παραμένουν στατικές (Sears and Shneiderman, 1993), όπως φαίνεται στην επόμενη εικόνα.



Παράδειγμα 3: Απόκρυψη/ εμφάνιση όγκου πληροφορίας και ομαδοποίηση της πληροφορίας  
 Σημεία που πρέπει να προσεχθούν: Δυνατότητες εκτέλεσης, ελέγχου του όγκου πληροφορίας και του τρόπου παρουσίασης

Κατά τη διάρκεια της διάδρασης ανθρώπου – υπολογιστή, εμφανίζεται συχνά η ανάγκη παρουσίασης ενός μεγάλου όγκου πληροφοριών στο χρήστη. Όπως έχει αναφερθεί και στο κεφάλαιο που περιγράφεται ο άνθρωπος-χρήστης, η δυνατότητα ταυτόχρονης επεξεργασίας πολλών πληροφοριών από τον άνθρωπο είναι περιορισμένη και θα πρέπει να δίνεται ιδιαίτερη προσοχή τόσο στον όγκο της πληροφορίας όσο και στον τρόπο με τον οποίο αυτή παρουσιάζεται.

Υπάρχουν αρκετά αντικείμενα που μπορούν να χρησιμοποιηθούν για να βοηθήσουν στην ομαδοποίηση της πληροφορίας καθώς και στη δυναμική εμφάνιση ή απόκρυψη πληροφοριών. Για παράδειγμα, εάν το ζητούμενο ήταν ο σχεδιασμός μιας οθόνης που θα επιτρέπει στο χρήστη ενός γραφείου ενοικίασεως αυτοκινήτων να δει άμεσα μια λίστα με τα αυτοκίνητα που διαθέτει το γραφείο ενοικίασεως, κατασκευαστή, μοντέλο, ημ/νια αγοράς, πινακίδες κυκλοφορίας και αν είναι νοικιασμένο ή όχι και για συγκεκριμένα αυτοκίνητα να δει επιπλέον στοιχεία όπως λίστα ατόμων που το έχουν ενοικιάσει (ονοματεπώνυμο, αριθμ. ταυτότητας, ημερομηνία ενοικίασης, ημερομηνία επιστροφής) ή λίστα προβλημάτων (ημερομηνία, περιγραφή προβλήματος, ενέργειες αποκατάστασης), μια πιθανή λύση θα ήταν η χρήση ενός πλαισίου όπως στην επόμενη εικόνα.



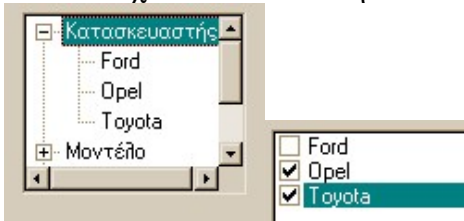
Στο παραπάνω πλαίσιο ο χρήστης μπορεί να εμφανίσει ή να αποκρύψει πληροφορίες ανάλογα με τις απαιτήσεις του. Το πλαίσιο δε προσφέρει άμεσα τη δυνατότητα αναζήτησης ή ταξινόμησης. Αυτό θα μπορούσε να λυθεί με την προσθήκη ενός αναδυόμενου επιλογέα ο οποίος θα πρόσφερε στο χρήστη τη δυνατότητα να εκτελέσει κάποιες λειτουργίες όπως αναζήτηση αυτοκινήτου με συγκεκριμένα χαρακτηριστικά. Όταν ο αριθμός των αυτοκινήτων είναι μεγάλος ένα τέτοιο πλαίσιο πιθανώς να μην προσφέρει την απαραίτητη ευχρηστία. Εναλλακτικά θα μπορούσε να χρησιμοποιηθεί μια δομή ενός πίνακα με στήλες τις πληροφορίες που επιθυμεί ο χρήστης, όπως φαίνεται στην επόμενη εικόνα.

Ο χρήστης μπορεί να ταξινομήσει κατ' αύξουσα ή φθίνουσα σειρά ή/και να ομαδοποιήσει τις πληροφορίες πατώντας στην αντίστοιχη ετικέτα. Για παράδειγμα πατώντας στην ετικέτα *Κατασκευαστής* τα αυτοκίνητα ομαδοποιούνται ανά κατασκευαστή και ταξινομούνται κατ' αλφαβητική σειρά. Στην περίπτωση που ο χρήστης επιθυμεί τη χρήση φίλτρων αυτό θα μπορούσε να υλοποιηθεί είτε με την υλοποίηση μιας φόρμας αναζήτησης ή με έναν πίνακα δεδομένων με επιπλέον ιδιότητες.

Κατασκευαστής	Μοντέλο	Έτος Κυκλ.	Αρ. Κυκλ.	Διαθεσιμότητα
Όλοι	Όλα	Όλα	Όλοι	Όλα
Toyota	Yaris	2001	ZHK4444	Διαθέσιμο
Ford	Focus	2000	ZHA3333	Διαθέσιμο

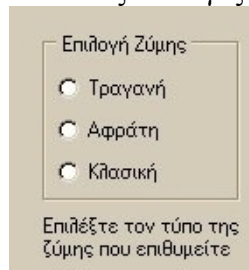
Πίνακας με τα δεδομένα των αυτοκινήτων που διαθέτει το γραφείο ενοικίασεως. Ο χρήστης μπορεί να εισάγει διάφορα φίλτρα στα πτυσσόμενα πλαίσια ώστε να περιορίσει τον όγκο της πληροφορίας που εμφανίζεται. Εναλλακτικά ο σχεδιαστής της οθόνης θα μπορούσε να χρησιμοποιήσει ένα

πλαίσιο όπως φαίνεται στις επόμενες εικόνες. Ο χρήστης μπορεί να επιλέξει τους κατασκευαστές ή άλλα στοιχεία των αυτοκινήτων που επιθυμεί να δει.

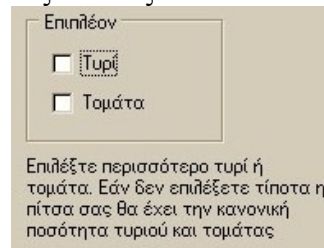


#### Παράδειγμα 4: Ανυπαρξία αντικειμένου / νέα αντικείμενα

Υπάρχουν περιπτώσεις στις οποίες τα αντικείμενα που έχει στη διάθεσή του ο σχεδιαστής δεν επαρκούν για την άμεση υλοποίηση των λειτουργιών που επιθυμεί. Για παράδειγμα, εάν ο σχεδιαστής θέλει να υλοποιήσει μια φόρμα για την παραγγελία πίτσας, όπου ο χρήστης θα επιλέγει τον τύπο της ζύμης μεταξύ τραγανής, αφράτης και κλασικής, να προσδιορίσει εάν θέλει επιπλέον τυρί και τομάτα και να επιλέξει τουλάχιστον ένα από τα υλικά: ζαμπόν, μπέικον, σαλάμι, μανιτάρια, ανανά και αντσούγιες. Ο σχεδιασμός των δύο πρώτων είναι απλός καθώς το ραδιοπλήκτρο μας δίνει τη δυνατότητα επιλογής μιας και μόνο μιας επιλογής από πολλές, ενώ τα πλαίσια επιλογής μας επιτρέπει να διαλέξουμε καμία, μία ή και όλες τις επιλογές. Ο σχεδιασμός αυτών των για τις δύο αυτές επιλογές φαίνεται στις επόμενες εικόνες.



Επιλογή τύπου ζύμης: Ένα από πολλά

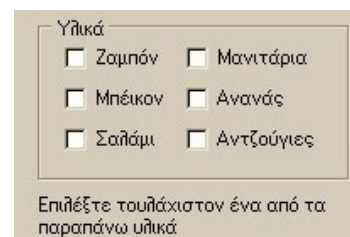


Επιλογή επιπλέον υλικών: Πολλά από πολλά

Δεν υπάρχει όμως αντικείμενο που να υλοποιεί το 'τουλάχιστον ένα'. Ο σχεδιαστής μπορεί να επιλέξει να χρησιμοποιήσει ένα από τα υπάρχοντα αντικείμενα όπως πλαίσια επιλογής ή ραδιοπλήκτρα, πλαίσιο λίστας ή λίστα πλαισίων επιλογής και να αλλάξει τη συμπεριφορά τους, ή να δημιουργήσει κάποιο εντελώς νέο αντικείμενο. Από αυτές τις εναλλακτικές λύσεις η τροποποίηση της συμπεριφοράς του ραδιοπλήκτρου είναι η χειρότερη επειδή θα προκαλέσει τη μεγαλύτερη σύγχυση στο χρήστη. Ο σχεδιαστής μπορεί να επιλέξει κάποια από τις υπόλοιπες λύσεις και να αποτρέψει το χρήστη να συνεχίσει (με απενεργοποίηση της αντίστοιχης επιλογής) εάν δεν επιλέξει τουλάχιστον ένα συστατικό (επόμενες εικόνες).



Πλαίσιο λίστας για επιλογή "τουλάχιστον ένα"



Πλαίσια επιλογής για επιλογή "τουλάχιστον ένα"



Λίστα πλαισίων επιλογής



Δύο πλαίσια λίστας

Η λύση του σχεδιασμού ενός νέου αντικειμένου (παραδείγματα παρουσιάζονται στις επόμενες εικόνες) απαιτεί καταρχήν σχεδιασμό από εξειδικευμένο γραφίστα και εκτεταμένους ελέγχους αξιολόγησης ώστε να διαπιστωθεί κατά πόσο είναι κατανοητά από τους χρήστες. Οι χρήστες θα χρειαστούν χρόνο για την εκμάθηση της σημασιολογίας του νέου αντικειμένου. Οπωσδήποτε η δημιουργία ενός νέου αντικειμένου αποτελεί λύση που θα πρέπει να επιλέγεται με ιδιαίτερη προσοχή και θα πρέπει να αποφεύγεται όταν τα νέα αντικείμενα συμπεριφέρονται περίπου ή ακριβώς όπως και κάποιο υπάρχον αντικείμενο.



Παραδείγματα εικονιδίων



## **Κεφάλαιο 13 Σχεδιασμός σε περιβάλλον ΠΠΠ**

Οι εφαρμογές του Παγκόσμιου Πλέγματος Πληροφοριών (WWW) παρουσίαζαν μέχρι πρόσφατα περιορισμένη λειτουργικότητα. Το πρόβλημα ήταν ότι η HTML που χρησιμοποιείται ως γλώσσα περιγραφής των σελίδων στο διαδίκτυο παρέχει μόνο ένα υποσύνολο των αντικειμένων που υπάρχουν στις συνηθισμένες διεπαφές. Προς αυτήν την κατεύθυνση η Java, η JavaScript, η XML μπορούν να χρησιμοποιηθούν για να ξεπεραστεί αυτό το μειονέκτημα.

### **13.1 Γενικές σκέψεις για το σχεδιασμό διεπαφών στο ΠΠΠ**

Το ΠΠΠ (WWW) -όπως προκύπτει και από το όνομά του- είναι ένα παγκόσμιο δίκτυο πληροφοριών. Το γεγονός ότι οι χρήστες που προσπελαίνουν τις πληροφορίες στο ΠΠΠ μπορεί να βρίσκονται χιλιάδες χιλιόμετρα μακριά από τον εξυπηρέτη ΠΠΠ διαφοροποιεί σε σημαντικό βαθμό τις προτεραιότητες που τίθενται κατά το σχεδιασμό μιας διεπαφής σε σχέση με αυτές των συνηθισμένων παραθυρικών διεπαφών.

#### **Ταχύτητα σύνδεσης**

Όπως έχει αναφερθεί σε προηγούμενο κεφάλαιο η ταχύτητα ανάδρασης είναι σημαντικό χαρακτηριστικό μιας διεπαφής. Για να μπορεί να εργαστεί αποδοτικά ένας χρήστης η διεπαφή, σε οποιοδήποτε περιβάλλον, θα πρέπει να ανταποκρίνεται άμεσα (σε λιγότερο από ένα δευτερόλεπτο). Μετά τα 10 δευτερόλεπτα ο χρήστης σταματά να ασχολείται με την εργασία και ασχολείται με άλλες εργασίες μέχρι να υπάρξει ανταπόκριση. Ειδικά στο ΠΠΠ, ακόμη και εάν ο εξυπηρέτης βρίσκεται σε κάποιο πανίσχυρο μηχάνημα με σύνδεση μεγάλου εύρους με το διαδίκτυο, η ταχύτητα με την οποία ο οποιοσδήποτε χρήστης θα προσπελάσει το δικτυακό τόπο θα περιοριστεί από το εύρος της δικής του σύνδεσης με το ΠΠΠ. Αυτό το χαρακτηριστικό επηρεάζει και τις σχεδιαστικές προσεγγίσεις ενός δικτυακού τόπου.

Οι σελίδες θα πρέπει να είναι μικρές σε μέγεθος και χωρίς μεγάλες εικόνες ή θα πρέπει να είναι δυνατή η τμηματική παρουσίασή τους. Για αυτούς τους λόγους δεν συνιστάται η χρήση πινάκων, ειδικά αυτών που καλύπτουν ολόκληρη τη σελίδα, επειδή το πρόγραμμα πλοήγησης θα πρέπει να περιμένει μέχρι να ολοκληρωθεί η λήψη του κώδικα HTML που περιγράφει τον πίνακα, πριν να τον εμφανίσει. Αντίστοιχα θα πρέπει να χρησιμοποιούνται οι παράμετροι WIDTH και HEIGHT στο διακριτικό <IMG SRC> ώστε το πρόγραμμα πλοήγησης να μπορεί να υπολογίσει το χώρο που χρειάζεται για την εμφάνιση της εικόνας. Επίσης είναι θα πρέπει να χρησιμοποιείται η παράμετρος ALT ώστε χρήστες που δεν βλέπουν εικόνες (είτε επειδή το έχουν απενεργοποιήσει, είτε επειδή το πρόγραμμα πλοήγησης που χρησιμοποιούν δεν το υποστηρίζει) να αντιλαμβάνονται το περιεχόμενο της σελίδας. Βέβαια, το κείμενο που θα χρησιμοποιείται θα πρέπει να είναι περιεκτικό. Η χρήση του ALT="εικόνα 1" δεν προσφέρει καμιά ιδιαίτερη πληροφορία στο χρήστη. Αντίθετα το ALT="Φωτογραφία σε μπαρ της Πάρου - διακοπές, Ιούλιος 1995" βοηθά το χρήστη να αντιληφθεί το περιεχόμενο της φωτογραφίας!

#### **Πρόγραμμα πλοήγησης**

Ένα ακόμη ζήτημα που θα πρέπει να λαμβάνεται υπόψη κατά το σχεδιασμό δικτυακών τόπων είναι ότι ο χρήστης μπορεί να χρησιμοποιεί κάποιο από μια πληθώρα διαφορετικών εκδόσεων διαφορετικών προγραμμάτων πλοήγησης. Για αυτό το λόγο θα πρέπει να αποφεύγεται η χρήση κώδικα (HTML, JavaScript, κλπ) που δεν υποστηρίζεται παρά μόνο από τις τελευταίες εκδόσεις συγκεκριμένων προγραμμάτων πλοήγησης (π.χ. η JavaScript 1.2 υποστηρίζεται πλήρως μόνο από την έκδοση 4.\* του Netscape και την έκδοση 5.5 του Internet Explorer - η τρέχουσα έκδοση της JavaScript είναι η 1.4).

### Χώρος στην οθόνη

Ο χώρος της οθόνης του χρήστη είναι σημαντικός. Ακόμη και σε μια οθόνη 19" -η οποία δεν μπορεί να θεωρηθεί ως η οθόνη που διαθέτει ο μέσος χρήστης- συνεχίζει να είναι σχετικά περιορισμένος και όταν προστεθεί ο χώρος που καταλαμβάνει το πρόγραμμα πλοήγησης (επιλογείς, εικονίδια, ράβδος τοποθεσίας, ...) περιορίζεται ακόμη περισσότερο. Τελικά ο χώρος που απομένει μέσα στο πρόγραμμα πλοήγησης για την εμφάνιση πληροφοριών θα πρέπει να χρησιμοποιηθεί με ιδιαίτερη προσοχή από το σχεδιαστή γιατί είναι πολύτιμος.

### Πλοήγηση στο δικτυακό τόπο

Ένας καλά σχεδιασμένος δικτυακός τόπος (όπως συμπεραίνουμε και από τις αρχές χρηστικότητας) θα πρέπει να δίνει στο χρήστη τη δυνατότητα να απαντάει κάθε στιγμή στις εξής ερωτήσεις:

- Πού βρίσκομαι;
- Πώς έφτασα εδώ;
- Πού μπορώ να πάω;

Η απάντηση στην πρώτη ερώτηση δίνεται με την εμφάνιση σε κάθε σελίδα του ονόματος ή του σήματος (logo) του ιδρύματος (εκπαιδευτικό ίδρυμα, εταιρεία, κλπ) που παρουσιάζεται στο δικτυακό τόπο, καθώς και με την παροχή ενός ιεραρχικού διαγράμματος στο οποίο θα φαίνεται η τρέχουσα σελίδα στην οποία βρίσκεται ο χρήστης. Μια τέτοια ράβδος (η οποία δημιουργείται δυναμικά) παρουσιάζεται στην επόμενη εικόνα.



Η δεύτερη ερώτηση μπορεί να απαντηθεί, από το πλήκτρο BACK του προγράμματος πλοήγησης όπου διατηρείται συνήθως το ιστορικό των επισκέψεων σε προηγούμενες σελίδες. Επίσης μπορεί να απαντηθεί από τη ράβδο της προηγούμενης εικόνας, αν και σε αυτή την περίπτωση δεν παρουσιάζεται κατ' ανάγκη το πλήρες ιστορικό της διαδρομής του χρήστη.

Τέλος, η τρίτη ερώτηση μπορεί να απαντηθεί με τη χρήση συνδέσμων. Η χρήση τυποποιημένων συνδέσμων (όπου υπάρχει η συνήθης μορφή των συνδέσμων, το χρώμα του συνδέσμου αλλάζει σε μοβ όταν το έχει επισκεφθεί ο χρήστης, κλπ) υπερτερεί άλλων μεθόδων (εικονιδίων, πτυσσόμενων μενού, ...) επειδή ο χρήστης μπορεί να δει άμεσα όλες τις επιλογές που του προσφέρονται και ποιες από αυτές έχει ήδη επισκεφθεί.

Με δεδομένο ότι η "καλή μέρα από το πρωί φαίνεται" ιδιαίτερη προσοχή θα πρέπει να δοθεί στην αρχική σελίδα, ώστε να έλκει το χρήστη να διατρέξει το δικτυακό τόπο. Προς αυτήν την κατεύθυνση, μια λειτουργική και εύχρηστη αρχική σελίδα θα πρέπει να περιέχει κάποιο ή όλα από τα ακόλουθα:

- κατάλογο πλοήγησης
- λειτουργία αναζήτησης
- νέα
- άμεση πρόσβαση στην πιο συχνά χρησιμοποιούμενη λειτουργία

Παράδειγμα τέτοιων σελίδων αποτελούν οι αρχικές σελίδες της AMD και της Intel.



Παρότι η HTML δεν παρέχει άμεση δυνατότητα δημιουργίας μενού, ο προγραμματιστής μπορεί να δημιουργήσει είτε στατικά μενού με HTML είτε δυναμικά μενού με προγράμματα που εκτελούνται στην πλευρά του εξυπηρετή. Για τον ίδιο σκοπό μπορεί να χρησιμοποιήσει και γλώσσες όπως η Java και η Javascript. Σε αυτές τις περιπτώσεις ισχύει ότι και για τις εικόνες.

Ολοκληρώνοντας τη σύντομη αναφορά σε θέματα σχεδιασμού διεπαφών για το ΠΠΠ υπενθυμίζουμε την ακόλουθη αναφορά από τον οδηγό σχεδιασμού σελίδων στο ΠΠΠ:

*Ο ορθός σχεδιασμός δικτυακών τόπων είναι σε μεγάλο βαθμό θέμα ισορροπίας μεταξύ της δομής και της σχέσης των μενού ή των αρχικών σελίδων και των υπόλοιπων συνδεδεμένων γραφικών και εγγράφων. Στόχος είναι η κατασκευή μιας ιεραρχίας από μενού και σελίδες, τις οποίες ο χρήστης θα αισθάνεται φυσιολογικές και καλά δομημένες και οι οποίες δε θα παρεμποδίζουν τη χρήση του δικτυακού τόπου και δε θα παραπλανούν το χρήστη [14].*

### 13.2 Πολυγλωσσικοί δικτυακοί τόποι

Ένας πολυγλωσσικός δικτυακός τόπος περιλαμβάνει πληροφορίες σε περισσότερες από μια γλώσσες. Συνήθως περιλαμβάνει πληροφορίες σε μια κύρια γλώσσα και μετάφραση κάποιων ή όλων των σελίδων σε άλλες γλώσσες. Ο χρήστης επιλέγει την γλώσσα της προτίμησης του και στις περισσότερες περιπτώσεις, σε οποιοδήποτε σημείο της πλοήγησης του μπορεί να δει την ίδια σελίδα σε κάποια από τις διαθέσιμες γλώσσες.



Ο σχεδιασμός και η υλοποίηση ενός πολυγλωσσικού δικτυακού τόπου έχει να αντιμετωπίσει μερικά επιπλέον θέματα που σχετίζονται με την συνύπαρξη των ίδων πληροφοριών σε περισσότερες από μια γλώσσες. Στην περίπτωση που δεν χρησιμοποιηθεί κάποια βάση δεδομένων αλλά επιλεγεί κάποια στατική υλοποίηση υπάρχουν δύο διαφορετικές προσεγγίσεις στην ανάπτυξη

του δικτυακού τόπου. Σύμφωνα με την πρώτη μπορεί να χρησιμοποιηθεί η προσθήκη κάποιου αναγνωριστικού γλώσσας σε αρχεία που διαφέρουν ανάμεσα σε γλώσσες. Για παράδειγμα το αρχείο *index.html* θα ονομασθεί *index-eng.html* για την Αγγλική έκδοση και *index-grk.html* για την Ελληνική. Αυτή η προσέγγιση δεν περιορίζεται μόνο στα HTML αρχεία αλλά καλύπτει κάθε αρχείο που θα πρέπει να τροποποιηθεί ανάλογα με την γλώσσα. Με δεδομένο ότι οι διαφορετικές εκδόσεις του ίδιου αρχείου τοποθετούνται όλες στον ίδιο κατάλογο, η προσέγγιση αυτή θα λειτουργήσει ικανοποιητικά για μικρό αριθμό γλωσσών.

Εναλλακτικά ο δικτυακός τόπος μπορεί να αναπτυχθεί έτσι ώστε τα κοινά μεταξύ γλωσσών αρχεία να τοποθετηθούν σε έναν κατάλογο και στη συνέχεια να δημιουργηθούν κατάλογοι για κάθε γλώσσα όπου κάθε κατάλογος είναι αντίγραφο του άλλου. Η προσέγγιση αυτή αποτυπώνεται στην επόμενη εικόνα.

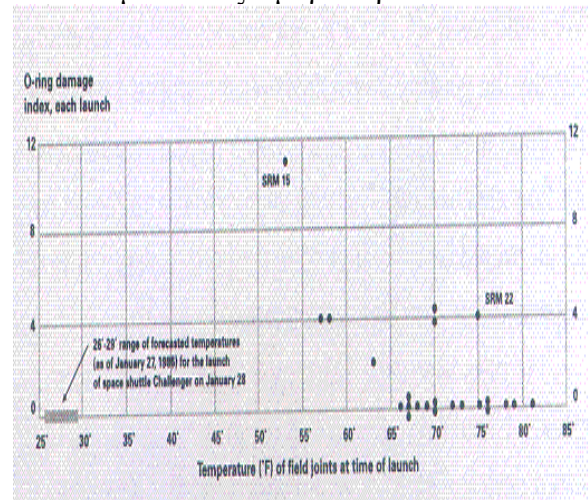
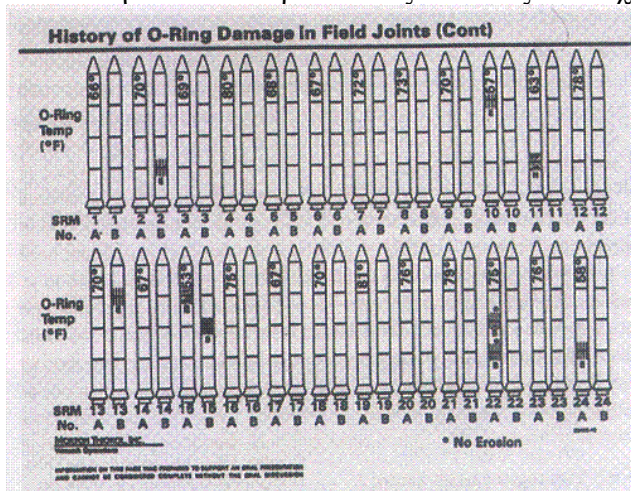
## Κεφάλαιο 14 Οπτικοποίηση Πληροφοριών

### 14.1 Εισαγωγή

Υπάρχουν δύο ερμηνείες της έννοιας "Οπτικοποίηση Πληροφοριών"\*:

1. Μία στην οποία αντιστοιχεί η έκφραση "Μια εικόνα αξίζει όσο 1000 λέξεις", όπου η οπτικοποίηση χρησιμεύει για την επικοινωνία μιας προϋπάρχουσας ιδέας.
2. Μία στην οποία γίνεται χρήση της όρασης για σκέψη, όπου η οπτικοποίηση χρησιμεύει για την δημιουργία ή την ανακάλυψη της ίδιας της ιδέας.

Στο κεφάλαιο αυτό κυρίως θα ασχοληθούμε με την δεύτερη ερμηνεία. Η οπτικοποίηση βασίζεται στην διαίσθηση που δημιουργείται από την εξωτερική αναγνώριση (external cognition). Για παράδειγμα, προσπαθήστε να κάνετε έναν πολλαπλασιασμό διψήφιων αριθμών (π.χ., 34 x 72) στο μυαλό σας και μετά γράφοντας του αριθμούς. Θα δείτε ότι η πιθανότητα σφάλματος στην πρώτη περίπτωση είναι πολύ μεγαλύτερη καθώς επίσης και ο απαιτούμενος χρόνος (κατά μέσο όρο 5 φορές περισσότερος). Ως άλλο παράδειγμα, τα διαγράμματα που μπορεί κάποιος να χρησιμοποιήσει για κάποια πληροφορία έχουν μεγάλη επίπτωση. Στην περίπτωση του δυστυχήματος του διαστημικού λεωφορείου Challenger, την προηγούμενη ημέρα οι μηχανικοί είχαν την παρακάτω αριστερή εικόνα μπροστά τους όπου φαίνονταν οι θέσεις των προβλημάτων σε διάφορες πτήσεις του παρελθόντος καθώς και η θερμοκρασία την ώρα της εκτόξευσης. Δυστυχώς η συσχέτιση της θερμοκρασίας με το μέγεθος του προβλήματος δεν ήταν καθόλου ξεκάθαρη. Ενώ αν είχαν μπροστά τους το διάγραμμα παρακάτω δεξιά, η συσχέτιση είναι ξεκάθαρη: με μεγάλη πιθανότητα, χαμηλότερες θερμοκρασίες δημιουργούν μεγαλύτερα προβλήματα και η θερμοκρασία της επικείμενης εκτόξευσης είναι κατά πολύ χαμηλότερη οποιασδήποτε του παρελθόντος. Το πώς θα δειχθούν τα δεδομένα παίζει μεγάλο ρόλο!



Ας δώσουμε τώρα έναν επίσημο ορισμό της Οπτικοποίησης Πληροφοριών: Είναι η μέσω υπολογιστών χρήση διαδραστικών, οπτικών αναπαραστάσεων αφηρημένων δεδομένων για την αύξηση της αναγνώρισης.

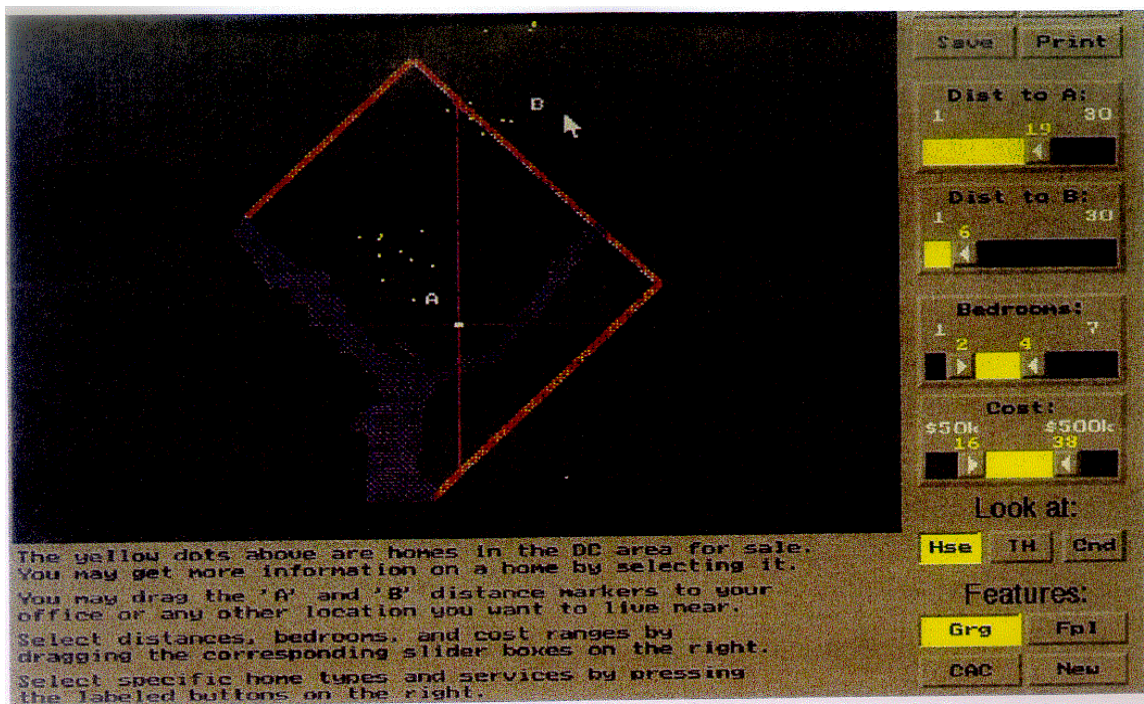
Η χρήση της οπτικοποίησης πληροφοριών είναι ποικίλη στη διαδικασία της αποκρυστάλλωσης γνώσης.

1. Συλλογή (συγκομιδή) πληροφοριών
2. Ψάξιμο για σχήμα (αναπαράσταση)

\* Το κεφάλαιο αυτό βασίζεται στο βιβλίο [16].

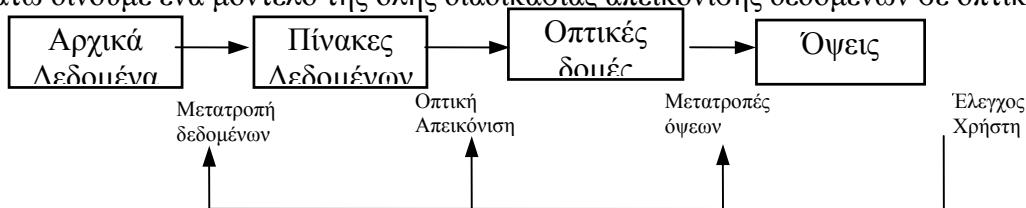
3. Συγκεκριμενοποίηση σχήματος με δεδομένα. Το υπόλειμμα αποτελείται από σημαντικά δεδομένα που δεν εκφράζονται στο σχήμα. Οπότε απαιτείται επιστροφή στο βήμα 2 για μείωση του υπολείμματος
4. Επίλυση προβλήματος με σύγκριση στοιχείων
5. Ψάξιμο νέου σχήματος που επαναδιατυπώνει το πρόβλημα ώστε να απαιτείται απλούστερη σύγκριση
6. Πακετάρισμα των προτύπων που βρέθηκαν προς παρουσίαση

Ως παράδειγμα κάποιων από τα παραπάνω βήματα να δώσουμε το πρόγραμμα Homefinder που σχεδιάστηκε για να βοηθάει ανθρώπους να βρουν σπίτι στην ευρύτερη περιοχή της Ουάσινγκτον. Στην παρακάτω εικόνα φαίνεται (έστω και κάπως δύσκολα λόγω του σκούρου υπόβαθρου) ότι πάνω στον χάρτη της περιοχής (η πόλη περικλείεται από τον ρόμβο) δίνονται με κουκίδες οι θέσεις των σπιτιών που ικανοποιούν τις συνθήκες που εκφράζονται στο δεξί μέρος της οθόνης πάνω σε κάποια ιδιώματα των σπιτιών (τα οποία αποτελούν ένα συγκεκριμένο σχήμα). Στο βήμα 3 ή 5 θα μπορούσαν να επαναδιατυπωθούν τα ίδια δεδομένα με βάση άλλα ιδιώματα, όπως “τύπος γειτονιάς”, “κατηγορία σπιτιού” (για 1 άτομο, για οικογένεια, κτλ), κτλ. Αυτά θα αποτελούσαν καινούργιο σχήμα.



## 14.2 Μοντέλο διαδικασίας απεικόνισης

Παρακάτω δίνουμε ένα μοντέλο της όλης διαδικασίας απεικόνισης δεδομένων σε οπτική μορφή.



Οι πίνακες δεδομένων αποτελούν μία σημαντική ενδιάμεση έννοια για την οργάνωση δεδομένων. Η οπτική απεικόνιση δε των πινάκων είναι το κύριο βήμα της οπτικοποίησης πληροφορίας. Στη

συνέχεια, σε ξεχωριστές υποενότητες θα αναλύσουμε κάθε ένα από τα παραπάνω κεντρικά σημεία της διαδικασίας (και τα στοιχεία και τα βήματα).

## Πίνακες

Οι πίνακες αντιστοιχούν ως ένα βαθμό τις κλασσικές σχέσεις από τις βάσεις δεδομένων, ξεχωρίζοντας τις λεγόμενες μεταβλητές (= τα πεδία των σχέσεων) και τις λεγόμενες περιπτώσεις (= τα στιγμιότυπα των σχέσεων). Πάντοτε έχουμε μία συνάρτηση από τις περιπτώσεις σε κάθε μεταβλητή.

Υπάρχουν πολλών και διαφόρων μορφών πίνακες:

- Με μία μεταβλητή να αντιπροσωπεύει κάθε περίπτωση (σε ορολογία βάσεων, με πρωτεύον κλειδί ενός πεδίου)
- Με δύο ή περισσότερες μεταβλητές να αντιπροσωπεύουν κάθε περίπτωση (αντίστοιχα, με πρωτεύον κλειδί δύο ή περισσότερων πεδίων)
- Με μη ατομική τιμή μεταβλητών, ώστε να μπορεί να εκφράσει δίκτυα και δένδρα περίπτωση (αντίστοιχα, σχέση μη πρώτης κανονικής μορφής)

Οι μεταβλητές των πινάκων διακρίνονται σε τρεις βασικές κατηγορίες:

- *Ονομαστική* (σύμβολο N), όπου δύο τιμές μπορούν μόνο να συγκριθούν με = και ≠), π.χ., τίτλοι ταινιών
- *Διατεταγμένη* (σύμβολο O), όπου υπάρχει ολική διάταξη  $\leq$  μεταξύ των τιμών, π.χ., καταλληλότητα ταινιών με τιμές ΚΑΤΑΛΛΗΛΟ  $\leq$  ΑΚΑΤΑΛΛΗΛΟ  $\leq$  ΑΥΣΤΗΡΩΣ ΑΚΑΤΑΛΛΗΛΟ
- *Ποσοτική* (σύμβολο Q), όπου επιπλέον μπορούν να γίνουν και αριθμητικές πράξεις μεταξύ των τιμών, π.χ., μήκος ταινιών

Μεταξύ των διαφόρων κατηγοριών μεταβλητών επιτρέπονται πολλές μετατροπές:

Q  $\rightarrow$  O διαχωρισμός σε διαστήματα ή άλλη κατηγοριοποίηση,  
π.χ., μήκος  $\rightarrow$  {Μικρό, Μεσαίο, Μεγάλο}

O  $\rightarrow$  N αγνοούμε τη διάταξη (σπάνιο)

N  $\rightarrow$  O επιβάλλουμε διάταξη, π.χ., λεξικογραφική διάταξη τίτλων

Σημαντική επίσης έννοια για τους πίνακες είναι τα *μεταδεδομένα*, τα οποία είναι χαρακτηριστικά των δεδομένων, πέραν του τύπου τους, και προσδιορίζουν επακριβέστερα τι μπορεί να γίνει και τι όχι στα δεδομένα. Για παράδειγμα, ας θεωρήσουμε κάποιες παραμέτρους (μεταβλητές) που είναι συναρτήσεις του γεωγραφικού μήκους και πλάτους. Για την παράμετρο υψόμετρο, μπορούμε να συμπληρώσουμε ενδιάμεσες τιμές για όποια γεωγραφικά μήκη και πλάτη λείπουν, ενώ για την παράμετρο αριθμός ατυχημάτων προφανώς δεν μπορούμε. Αυτή η πληροφορία είναι μεταδεδομένο. Οι μεταβλητές και οι περιπτώσεις ενός πίνακα είναι επίσης μεταδεδομένα, καθώς και ο τύπος των μεταβλητών.

## Μετατροπές Δεδομένων

Κατά την μετατροπή δεδομένων, συνήθως επιτελείται απώλεια αλλά και καμιά φορά κέρδος πληροφορίας. Οι μετατροπές αυτές μπορούν να τοποθετηθούν σε τέσσερις κατηγορίες:

Τιμές  $\rightarrow$  Συναγόμενες τιμές

Τιμές  $\rightarrow$  Συναγόμενη δομή

Δομή  $\rightarrow$  Συναγόμενες τιμές

Δομή → Συναγόμενη δομή

Ο παρακάτω πίνακας δίνει κάποια παραδείγματα κάθε κατηγορίας:

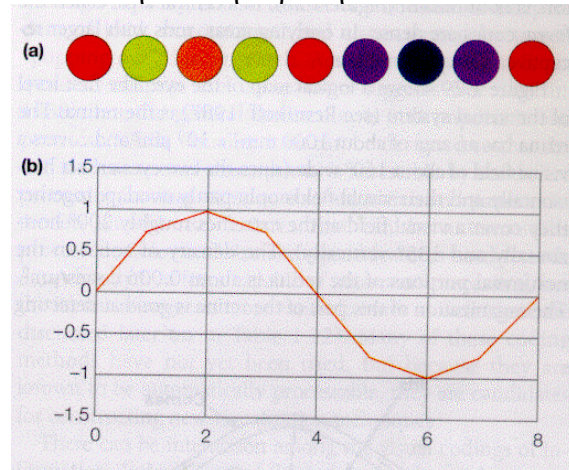
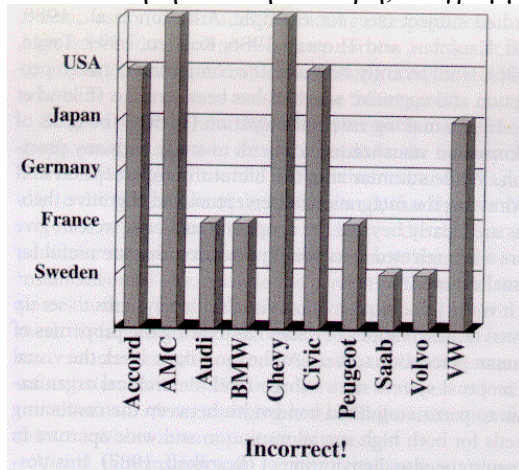
	Συναγόμενη Τιμή	Συναγόμενη Δομή
Τιμή	Μέσος Όρος	Διάταξη Κατηγοριοποίηση Προαγωγή (Μεταβλητή → Περίπτωση)
Δομή	Καθαίρεση (Περίπτωση → Μεταβλητή)	$X, Y, Z \rightarrow P_{x,y,z}$

Ατομικές μετατροπές δεδομένων μπορούν να συντεθούν αλυσιδωτά χωρίς όρια, έτσι ώστε να επιφέρει κάποιος οποιεσδήποτε αλλαγές, οσοδήποτε πολύπλοκες, απαιτούνται από το πρόβλημα.

### Οπτικές δομές

Η οπτική απεικόνιση πρέπει να είναι

- εκφραστική, δηλαδή να αναπαριστά τα δεδομένα στον δεδομένο πίνακα δεδομένων και μόνο αυτά. Ως αντιπαράδειγμα, θεωρείστε το παρακάτω διάγραμμα αριστερά. Περιέχει σφάλματα (π.χ., το Audi δεν βγαίνει στην Γαλλία) και χρησιμοποιεί έναν διατεταγμένο άξονα στην ψ-διάσταση για να αναπαραστήσει μία ονομαστική μεταβλητή, εκφράζοντας έτσι συσχετίσεις που δεν υπάρχουν στα δεδομένα (π.χ., ότι οι ΗΠΑ είναι πιο πάνω από την Ιαπωνία).
- αποδοτική, δηλαδή να δημιουργεί στο χρήστη άμεση κατανόηση και διάκριση διαφόρων, και να προκαλεί ελάχιστα σφάλματα. Ως παράδειγμα/αντιπαράδειγμα, θεωρείστε τις δύο απεικονίσεις παρακάτω δεξιά. Και οι δύο αναπαριστούν ένα ημίτονο, όπως πόσο πιο καθαρή και κατανοητή είναι η δεύτερη, διαγραμματική, απεικόνιση από την πρώτη.



Οι οπτικές δομές αποτελούνται από τρία κυρίως στοιχεία:

1. χωρικό υπόβαθρο
2. οπτικά στοιχεία
3. γραφικές ιδιότητες των οπτικών στοιχείων

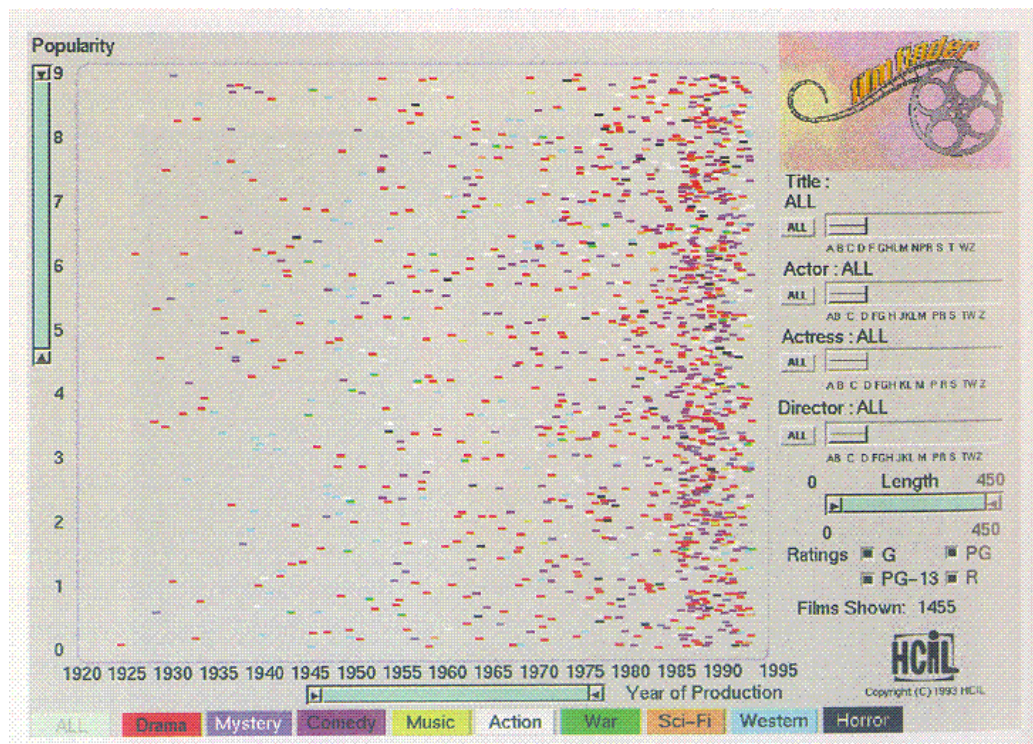
Αυτά αναλύονται παρακάτω διεξοδικότερα.



1) *Χωρικό υπόβαθρο*. Αποτελεί ξεχωριστή οντότητα, επειδή είναι πολύ σημαντική η απεικόνιση δεδομένων στο χώρο. Για παράδειγμα, το χωρικό υπόβαθρο είχε πολύ σημαντικό ρόλο στο θέμα του Challenger που προαναφέραμε. Το χωρικό υπόβαθρο θεωρείται αποτελούμενο από άξονες, δύο που καταλαμβάνουν συνήθως τους συνήθεις ρόλους του χ και ψ, και άλλους που τοποθετούνται υπό διάφορες μορφές στον χώρο. Διακρίνουμε τέσσερις τύπους αξόνων:

- U - αδόμητος (όπου ουσιαστικά δεν υπάρχει, π.χ., συνήθεις απεικονίσεις γράφων)
- N - ονομαστικός
- O - διατεταγμένος
- Q - ποσοτικός

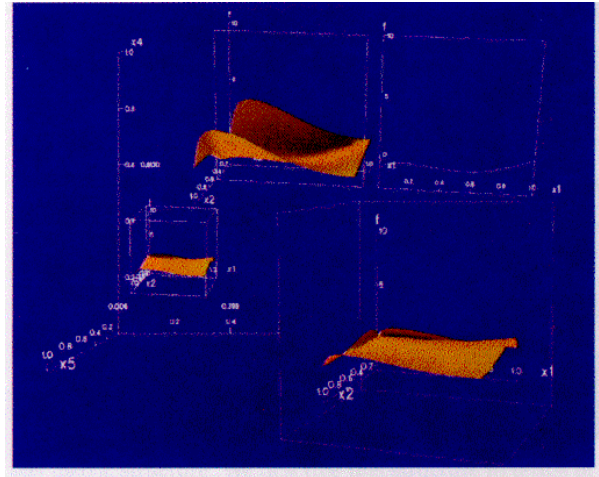
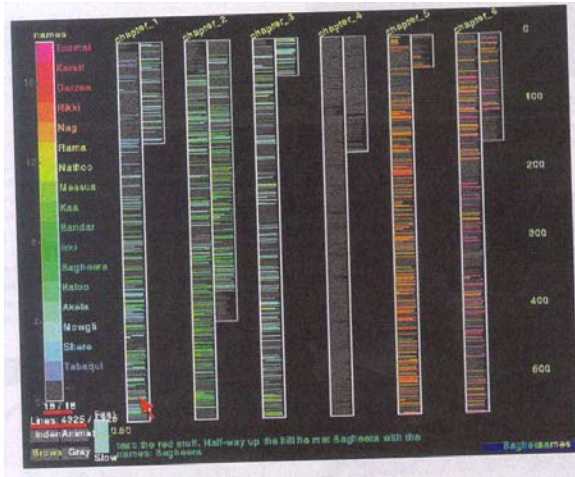
Εκ των πραγμάτων είναι πολύ σημαντικό ποιες μεταβλητές απεικονίζονται στους άξονες χ και ψ, καθώς αυτοί δημιουργούν την βασική εποπτεία του χώρου των δεδομένων. Για παράδειγμα, στο εργαλείο FilmFinder του οποίου ο στόχος είναι να μπορεί κάποιος να μελετήσει εύκολα τα στοιχεία πολλών ταινιών, όπως φαίνεται στην παρακάτω εικόνα, έχουμε την απεικόνιση Χρονιά → Qχ και (Δείκτης προτίμησης) → Qψ. Σ' αυτόν τον συμβολισμό, το Q υποδεικνύει ότι πρόκειται για ποσοτικό άξονα, και το χ και ψ ότι τοποθετείται η μεταβλητή αυτή στον αντίστοιχο χωρικό άξονα. Επίσης παρατηρούμε στην εικόνα και τις ακόλουθες απεικονίσεις μεταβλητών σε άξονες: (Αύξων αριθμός ταινίας) → P, δηλαδή κάθε ταινία αντιπροσωπεύεται από ένα σημείο του χ-ψ χώρου, Καταλληλότητα → Oψ, (Τύπος ταινίας) → Nχ, κτλ, όπου διάφορες μεταβλητές απεικονίζονται σε άξονες διαφόρων κουμπιών.



Η χωρική τοποθέτηση έχει διάφορες τεχνικές που αυξάνουν την πληροφορία της:

















- *Σύνθεση* (composition): αντικατοπτρίζει την ορθογώνια τοποθέτηση αξόνων σε δύο διαστάσεις. Και σε τρεις διαστάσεις γίνεται αλλά δυσκολεύει.
- *Ευθυγράμμιση* (alignment): είναι η επανάληψη άξονα σε διαφορετική θέση στο χώρο.
- *Αναδίπλωση* (folding): είναι η συνέχιση άξονα σε μια ορθόγωνη διάσταση
- *Αναδρομή*: είναι η συνεχής υποδιαίρεση του χώρου.
- *Υπερφόρτωση* (overloading): είναι η επαναχρησιμοποίηση του ίδιου χώρου για τον ίδιο πίνακα.

Οι παρακάτω εικόνες αποτελούν παραδείγματα αναδίπλωσης και υπερφόρτωσης, αντίστοιχα. Η αριστερή αντιστοιχεί διάφορα κεφάλαια βιβλίου σε άξονες χρωματίζοντας διαφορετικά τις γραμμές όπου αναφέρονται οι ήρωες του βιβλίου, και αναδιπλώνει τους άξονες των κεφαλαίων που δεν χωράνε (όλων στην εικόνα). Η δεύτερη υπερφορτώνει τον χώρο για να οπτικοποιήσει πολυδιάστατους πίνακες δεδομένων.



2) *Οπτικά Στοιχεία*. Αυτά είναι τα οπτικά αντικείμενα που εμφανίζονται στο χώρο, στην οπτική αναπαράσταση. Κατά τα γνωστά, αυτά είναι τεσσάρων τύπων: σημεία (0-διάστατα), γραμμές (1-διάστατα), περιοχές (2-διάστατα), και όγκοι (3-διάστατα). Σημειωτέον ότι σε αντίθεση με τις αντίστοιχες μαθηματικές έννοιες, τα σημεία και οι γραμμές καταλαμβάνουν επιφάνεια, δηλαδή χρειάζεται επιφάνεια για να οπτικοποιηθεί κάτι που δεν έχει επιφάνεια, είδ' αλλιώς δεν θα φαινόνταν!

3) *Γραφικές ιδιότητες*. Οι ιδιότητες των οπτικών στοιχείων είναι πάρα πολλές, όμως υπάρχουν έξι που αποτελούν ένα βασικό σύνολο που έχει την μεγαλύτερη εφαρμοστικότητα, στο οποίο και επικεντρωνόμαστε: μέγεθος, προσανατολισμός, σκίαση (gray scale), χρώμα, υφή (texture), και σχήμα. Μία έβδομη βασική είναι η θέση του στοιχείου στο χώρο, την οποία συζητήσαμε ξεχωριστά όμως λόγω της ιδιαίτερα υψηλής σημασίας της. Οι επτά αυτές (με την θέση σε παρένθεση) μπορούν να τοποθετηθούν στον παρακάτω πίνακα με βάση δύο χαρακτηριστικά: το αν είναι χωρικές ή του αντικειμένου αυτού καθ' εαυτού (το οποίο βασίζεται στο τμήμα του εγκεφάλου που εικάζεται ότι τις επεξεργάζεται), και το αν είναι κατάλληλες για την αναπαράσταση της έκτασης μιας κλίμακας ή για την διαφοροποίηση μεταξύ τιμών.

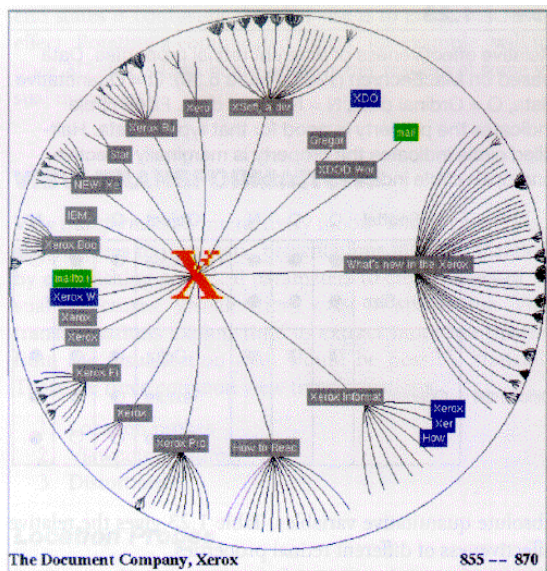
	Χωρικές	Αντικείμενου
Έκταση κλίμακας	(Θέση) $\_ \_ \_ \_$ Μέγεθος $0 \ 0 \ 0 \ 0$	Σκίαση    
Διαφοροποίηση	Προσανατολισμός $- /   \backslash$	Χρώμα     Υφή     Σχήμα    

Κάθε ιδιότητα είναι καλή για διαφορετικά πράγματα. Στον παρακάτω πίνακα υποδεικνύουμε τους τύπους μεταβλητών (Q, O, N) για τους οποίους ενδείκνυται η κάθε ιδιότητα. Το σύμβολο ++

χρησιμοποιείται για να δείξει ιδιαίτερα καλή εκφραστικότητα, το σύμβολο + για να δείξει απλά υποφερτή εκφραστικότητα, ενώ ή έλλειψη συμβόλου αντιπροσωπεύει κακή επιλογή.

	Χωρικές	Q	O	N	Αντικειμένους	Q	O	N
Εκταση	(θέση) Μέγεθος	+	++	++	Σκίαση	+	++	
Διαφοροποίηση	Προσανατολισμός	+	+	++	Χρώμα Υφή Σχήμα	+	+	++

4) *Σύνδεση και Εγκλεισμός*. Πέραν των τριών βασικών συνιστωσών μιας οπτικής δομής, αναφέρουμε και τις μεθόδους της σύνδεσης και του εγκλεισμού, όπου χωρίς χωρικό υπόβαθρο (ουσιαστικά με ανύπαρκτους άξονες) μπορεί να υποδηλωθεί γράφος. Οι συνδέσεις μας δίνουν την συνήθη απεικόνιση (αριστερή εικόνα παρακάτω), ενώ ο εγκλεισμός είναι καλός για δένδρα (δεξιά εικόνα παρακάτω).

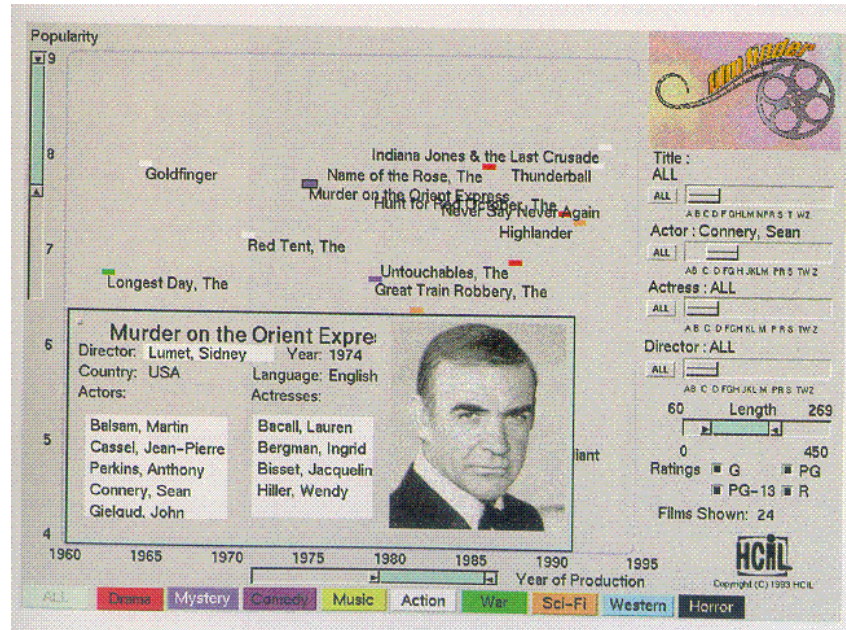


5) *Χρονική απεικόνιση*. Το μάτι πολύ ευαίσθητο σε αλλαγές. Ως εκ τούτου μπορούμε να απεικονίσουμε και οποιαδήποτε μεταβλητή στον χρόνο (αντί στον χώρο ή μία από τις ιδιότητες των οπτικών στοιχείων που αναφέραμε). Απαιτείται προσοχή όμως διότι μία καλή με πρώτη ματιά απεικόνιση φαίνεται να είναι η απεικόνιση της μεταβλητής “Χρόνος” → Χρόνο. Αυτό όμως δεν είναι πάντα επιθυμητό. Απεικονίζοντας τον “χρόνο” στην Qx, για παράδειγμα, είναι πολλές φορές καλύτερο διότι επιτρέπει την σύγκριση μεταξύ χρόνων που αλλιώς δεν είναι δυνατή.

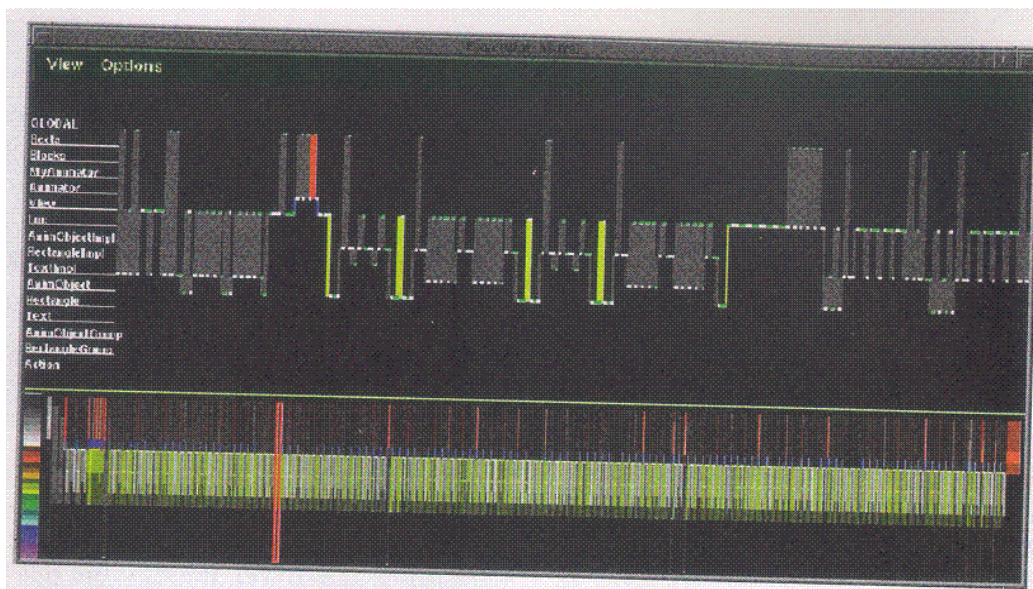
### Μετατροπές όψεων

Με τον παραπάνω όρο εννοούμε ουσιαστικά διαδραστικές μετατροπές οπτικών δομών οι οποίες μετατρέπουν μια οπτική αναπαράσταση από στατική σε δυναμική. Υπάρχουν τρεις βασικοί τύποι:

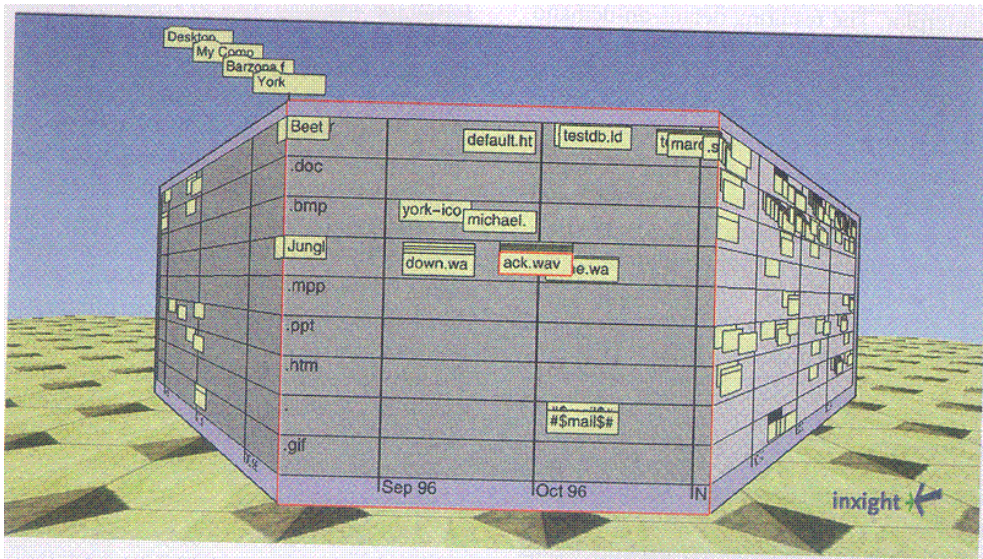
- *Ακίδα θέσης*: Η πίεση του πλήκτρου του ποντικιού σε συγκεκριμένη θέση στο χώρο ή και απλό πέρασμα του δρομέα πάνω από τη θέση δίνει περισσότερες πληροφορίες. Στο εργαλείο FilmFinder, για παράδειγμα, αν τοποθετήσουμε τον δρομέα πάνω μία συγκεκριμένη ταινία προκύπτει περισσότερη πληροφορία.



- *Έλεγχος οπτικής γωνίας:* Οι συνήθεις δυνατότητες της εστίασης, κίνησης δεξιά και αριστερά, κτλ. Όλα αυτά βοηθούν, αλλά χάνει κανείς το υπόλοιπο περιβάλλον των δεδομένων το οποίο μπορεί να είναι σημαντικό. Η λύση ίσως βρίσκεται σε έναν συνδυασμό δύο όψεων, λεπτομερειακής και επικεκριμένης, όπως στο παρακάτω παράδειγμα, όπου στην κάτω όψη συνεχίζει να παρουσιάζεται ολόκληρη η κλίμακα τιμών της παραμέτρου μαζί με το διάστημα το οποίο έχει επιλεγεί για να δώσει την πάνω όψη.



- *Παραμόρφωση:* Είναι ένας άλλος τρόπος για να επικεντρωθεί ο χρήστης σε κάποιο σημείο ενδιαφέροντος χωρίς να χαθεί το περιβάλλον. Ο στόχος είναι η δημιουργία εστιακού σημείου και του περιβάλλοντος στην ίδια εικόνα. Παράδειγμα αποτελεί το δέντρο υπερβολής που είδαμε παραπάνω στις συνδέσεις και τους εγκλεισμούς, καθώς και ο "Τοίχος Εστίασης" όπου δείχνει παρακάτω τα αρχεία σε διάφορα ευρετήρια.



## Έλεγχος χρήστη

Μέσα από τον έλεγχο του χρήστη και τα τρία προηγούμενα στάδια μετατροπών μπορούν να επηρεαστούν. Για τον τρόπο που πρέπει να γίνεται ο έλεγχος από τον χρήστη ισχύουν όλα αυτά που συζητήσαμε στα προηγούμενα κεφάλαια. Ο άμεσος χειρισμός είναι βέβαια πολύ σημαντικός όταν μπορεί να εφαρμοστεί.

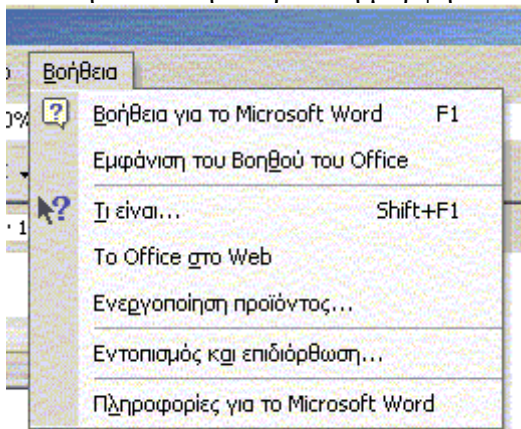
Τελειώνοντας, να υπογραμμίσουμε την εξής γενική αρχή της οπτικοποίησης πληροφορίας, την οποία πρέπει πάντα να έχει ο χρήστης υπ' όψη του, και επομένως και ο δημιουργός των εργαλείων. Η σωστή σειρά εξερεύνησης των δεδομένων είναι:

- Πρώτα γενική αντίληψη
- Μετά εστίαση και επιλογή
- Μετά οι λεπτομέρειες όποτε χρειάζονται

## **ΜΕΡΟΣ IV    ΑΛΛΑ ΘΕΜΑΤΑ**

## Κεφάλαιο 15 Τεκμηρίωση - Εγχειρίδια – Βοήθεια

Για να μπορέσουν οι χρήστες να αξιοποιήσουν τις δυνατότητες που τους προσφέρει ένα λογισμικό προϊόν χρειάζονται εκπαίδευση. Αρκετές φορές αυτή η εκπαίδευση προέρχεται από συναδέλφους που ήδη γνωρίζουν το λογισμικό, όμως στη γενική περίπτωση οι χρήστες προσφεύγουν στην τεκμηρίωση η οποία πρέπει να συνοδεύει το λογισμικό και η οποία μπορεί να βρίσκεται είτε σε έντυπη είτε σε ηλεκτρονική μορφή.



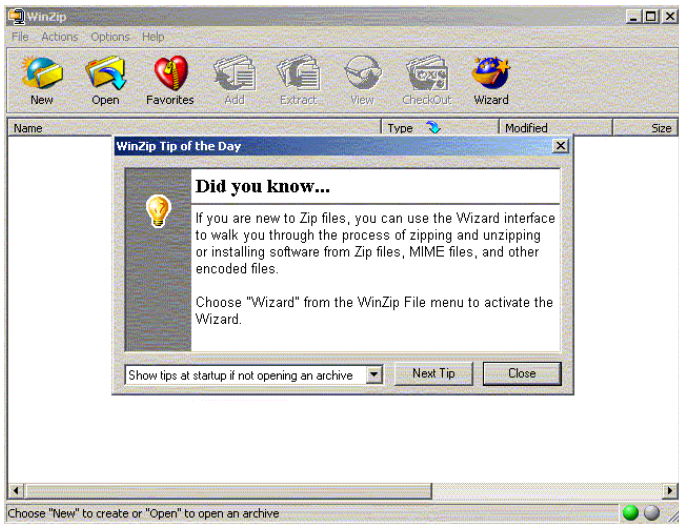
**Εικόνα 6. Επιλογέας βοήθειας του MS Word. Υπάρχουν διαφορετικά είδη υποστήριξης που μπορεί να λάβει ο χρήστης: βοήθεια άμεσης πρόσβασης σχετική με τα συμφραζόμενα (F1), γενική βοήθεια μέσω του βοηθού, βοήθεια για συγκεκριμένα αντικείμενα (Shift+F1), πληροφορίες μέσω του Διαδικτύου, κλπ.**

Ανάλογα με το περιεχόμενό της η τεκμηρίωση μπορεί να κατηγοριοποιηθεί σε διαφορετικές ομάδες έντυπης ή ηλεκτρονικής τεκμηρίωσης όπως συνοπτικούς οδηγούς εκμάθησης, μαθήματα, εγχειρίδια χρήσης, εγχειρίδια μετάβασης, κάρτες λειτουργιών, κλπ. Ο επόμενος πίνακας κατηγοριοποιεί την τεκμηρίωση ανάλογα με το περιεχόμενό της και το μέσο (έντυπη ή ηλεκτρονική) στο οποίο βρίσκεται.

Τύπος Τεκμηρίωσης	Έντυπη	Ηλεκτρονική	Περιγραφή
Συνοπτικοί οδηγοί εκμάθησης	✓	✓	Βοηθούν τους χρήστες να εξοικειωθούν με τα βασικά χαρακτηριστικά του λογισμικού και να δοκιμάσουν σε σύντομο χρονικό διάστημα τις λειτουργίες του.
Μαθήματα	✓	✓	Χρησιμοποιούνται για να εκπαιδεύσουν τους χρήστες. Στην ηλεκτρονική τους μορφή μπορεί να περιλαμβάνουν προσομοιώσεις, κινούμενες εικόνες ή και βίντεο ώστε να καταστήσουν πιο ενδιαφέρον το μάθημα.

Τύπος Τεκμηρίωσης	Έντυπη	Ηλεκτρονική	Περιγραφή
Εγχειρίδια χρήσης	✓	✓	Καλύπτουν όλες τις λειτουργίες και τα χαρακτηριστικά του λογισμικού, περιλαμβάνουν ευρετήριο όρων και περιεχόμενα ενώ στην ηλεκτρονική τους μορφή συχνά διαθέτουν και σύστημα αναζήτησης.
Εγχειρίδια μετάβασης	✓	✓	Χρησιμοποιούνται για να βοηθήσουν χρήστες έμπειρους με αντίστοιχα λογισμικά προϊόντα να εξοικειωθούν με τις λειτουργίες του εν λόγω λογισμικού προϊόντος.
Κάρτες λειτουργιών	✓	✓	Μικρές κάρτες που περιλαμβάνουν τις πιο συχνά χρησιμοποιούμενες εντολές, με σύντομη περιγραφή και τα πλήκτρα συντόμευσης, εάν υπάρχουν.
Συμβουλές εκκίνησης		✓	Εμφανίζονται στην αρχή του προγράμματος ή κατ' απαίτηση του χρήστη και προσφέρουν συμβουλές για επιλογές του λογισμικού.
Άμεση βοήθεια σχετική με τα συμφραζόμενα		✓	Εμφανίζεται κατ' απαίτηση του χρήστη ή σε ορισμένα συστήματα αυτόματα όταν αναγνωρίσει ότι ο χρήστης έχει πρόβλημα και προσφέρει βοήθεια σχετική με την εργασία που εκτελεί ο χρήστης.
Πρόγραμμα παρουσίασης		✓	Προσφέρει μια επισκόπηση του λογισμικού και των δυνατοτήτων του, δίνεται συνήθως σε υποψήφιους αγοραστές του προγράμματος.

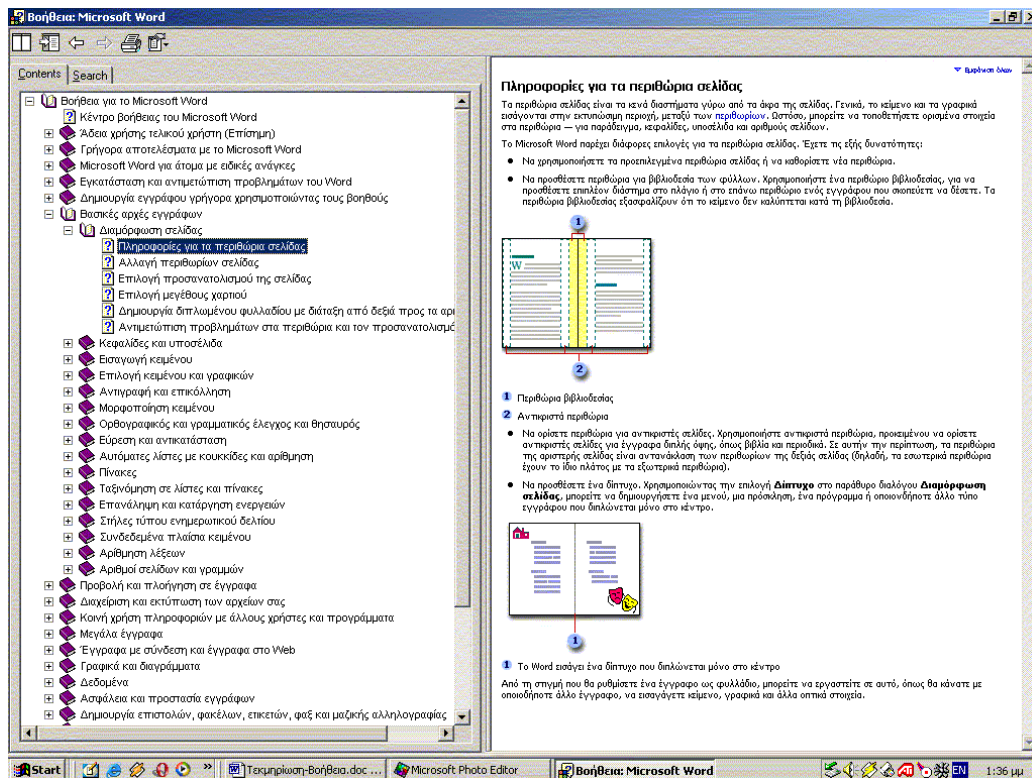




Εικόνα 7. Συμβουλές εκκίνησης του WinZip.

### 15.1 Έντυπη και Ηλεκτρονική μορφή

Αρχικά τα περισσότερα λογισμικά προϊόντα προσφέρονταν με τεκμηρίωση σε έντυπη μορφή. Καθώς όμως το υλικό εξελισσόταν και τα συστήματα γινόντουσαν πιο πολύπλοκα, αρκετά λογισμικά άρχισαν να προσθέτουν ηλεκτρονικές εκδόσεις της έντυπης τεκμηρίωσης σε σημείο που σήμερα υπάρχουν προϊόντα που προσφέρουν την τεκμηρίωση μόνο σε ηλεκτρονική μορφή και προαιρετικά, κατόπιν παραγγελίας και σε έντυπη.



Εικόνα 8. Περιεχόμενα βοήθειας του MS Word. Ο χρήστης μπορεί να βρει τις πληροφορίες που χρειάζεται είτε μέσω των περιεχομένων είτε μέσω της αναζήτησης.

Κάθε μια από τις δύο μορφές, έντυπη και ηλεκτρονική, έχει συγκεκριμένα πλεονεκτήματα και μειονεκτήματα, έτσι ώστε η χρήση τους να είναι σε μεγάλο βαθμό συμπληρωματική. Ο επόμενος πίνακας παρουσιάζει τα πλεονεκτήματα και τα μειονεκτήματα της ηλεκτρονικής τεκμηρίωσης.

<b>Πλεονεκτήματα</b>	<b>Μειονεκτήματα</b>
Η τεκμηρίωση περιλαμβάνεται με το λογισμικό. Ο χρήστης δε χρειάζεται να ψάχνει στη βιβλιοθήκη για βιβλία.	Το διάβασμα από βιβλίο είναι πιο ξεκούραστο από το διάβασμα από οθόνη. Επίσης η ανάλυση των εικόνων στο χαρτί είναι καλύτερη.
Ειδικά στην περίπτωση της βοήθειας άμεσης πρόσβασης που είναι σχετική με τα συμφραζόμενα, ο χρήστης έχει άμεση πρόσβαση σε υποστήριξη που είναι σχετική με την εργασία που εκτελεί.	Ο χώρος στην οθόνη είναι περιορισμένος. Το άνοιγμα παραθύρων βοήθειας περιορίζει το χώρο εργασίας του χρήστη. Επιπλέον, η μετάβαση μεταξύ παραθύρων δυσκολεύει τους χρήστες, καθώς πρέπει να θυμούνται που βρίσκονται σε κάθε παράθυρο.
Ο χρήστης δεν χρειάζεται να καταναλώσει χώρο στο γραφείο του για να ανοίξει το βιβλίο και να βρει το κείμενο που επιθυμεί.	Σε μια σελίδα χαρτί χωρά περισσότερη πληροφορία από ότι σε μια σελίδα οθόνης και η φυλλομέτρηση είναι συνήθως πιο γρήγορη.
Η τεκμηρίωση μπορεί να περιλαμβάνει εκτός από εικόνες και κινούμενες εικόνες, βίντεο, ήχο, οι οποίες κεντρίζουν το ενδιαφέρον του χρήστη και τον βοηθούν να εξοικειωθεί πιο γρήγορα με το λογισμικό.	Νέοι χρήστες μπορεί να μην είναι εξοικειωμένοι με τη διεπαφή της ηλεκτρονικής τεκμηρίωσης. Σε μια τέτοια περίπτωση η ανάγκη για εκμάθηση του συστήματος ηλεκτρονικής τεκμηρίωσης θα καθυστερήσει την εκμάθηση του ίδιου του λογισμικού.
Η τεκμηρίωση μπορεί να περιλαμβάνει σύστημα αναζήτησης που επιταχύνει την ανεύρεση της επιθυμητής πληροφορίας.	
Με το διαδίκτυο η πληροφορία μπορεί να ανανεώνεται αυτόματα.	

## 15.2 Συγγραφή Τεκμηρίωσης

Παλαιότερα το έργο της συγγραφής της τεκμηρίωσης το αναλάμβανε κάποιος από τους προγραμματιστές, συνήθως όταν το προγραμματιστικό μέρος είχε σχεδόν ολοκληρωθεί. Αυτό είχε ως αποτέλεσμα, συχνά η τεκμηρίωση να μην έχει την απαιτούμενη ποιότητα και να μην προσφέρει την αναγκαία υποστήριξη στην εκμάθηση του λογισμικού. Έκτοτε έχει σημειωθεί ιδιαίτερη πρόοδος στον τομέα της τεκμηρίωσης του λογισμικού. Το συγγραφικό ρόλο το αναλαμβάνει εξειδικευμένο προσωπικό που γνωρίζει και την τεχνολογία του προϊόντος και είναι επίσης σε θέση να συγγράψει κείμενο με ομαλή ροή. Κατά τη διάρκεια της συγγραφής της τεκμηρίωσης τα ακόλουθα σημεία πρέπει να προσεχθούν.

### Χρόνος συγγραφής

Η συγγραφή της τεκμηρίωσης θα πρέπει να ξεκινήσει αρκετά νωρίς στη διάρκεια ανάπτυξης του λογισμικού. Στη γενική περίπτωση είναι μια εργασία που εκτελείται παράλληλα με την ανάπτυξη του λογισμικού και μπορεί να ξεκινήσει περίπου κατά τη φάση σχεδιασμού. Αυτό επιτρέπει να υπάρχει μεγαλύτερος χρόνος για τη συγγραφή και ποιοτικότερο αποτέλεσμα. Επιπλέον η τεκμηρίωση μπορεί να δοθεί σε χρήστες του λογισμικού ώστε να υπάρχει και αξιολόγηση τόσο της

τεκμηρίωσης όσο και του ίδιου του λογισμικού μέσω της τεκμηρίωσης. Με αυτόν τον τρόπο είναι δυνατό να χρησιμοποιηθεί η τεκμηρίωση ως μέσο αξιολόγησης του σχεδιασμού του λογισμικού.

### **Μέγεθος κειμένου**

Το μέγεθος ενός εγχειριδίου είναι σημαντικός παράγοντας στην εξασφάλιση της αποδοχής της τεκμηρίωσης από τους χρήστες. Είναι αποδεδειγμένο ότι στη γενική περίπτωση η εικόνα του χρήστη που θα δαπανήσει αρκετές ώρες ή και μέρες διαβάζοντας ένα εγχειρίδιο, πριν ξεκινήσει να χρησιμοποιεί το λογισμικό δεν ισχύει. Οι περισσότεροι χρήστες θέλουν να ξεκινήσουν να χρησιμοποιούν το λογισμικό όσο γίνεται πιο γρήγορα, να δοκιμάσουν λειτουργίες και να δουν αποτελέσματα. Όταν ξεκινά ο χρήστης να μαθαίνει ένα λογισμικό προϊόν πρέπει ταυτόχρονα να ασχοληθεί τόσο με το ίδιο το λογισμικό αναζητώντας τις λειτουργίες που επιθυμεί όσο και να διαβάσει το εγχειρίδιο χρήσης για να βρει τις σχετικές πληροφορίες. Ένα ογκώδες έντυπο εγχειρίδιο χρήσης μπορεί να αποτελέσει τροχοπέδη για ένα νέο χρήστη. Προς αυτήν την κατεύθυνση συνιστάται τα εγχειρίδια να είναι σχετικά περιορισμένα στο μέγεθος ώστε να μην αποθαρρύνουν τους χρήστες και να προσφέρουν πρώτα τις πιο συχνά χρησιμοποιούμενες επιλογές, ώστε να βοηθήσουν τους νέους χρήστες να χρησιμοποιήσουν το λογισμικό όσο γίνεται πιο γρήγορα.

### **Οργάνωση και μορφή γραφής**

Είτε έντυπη είτε ηλεκτρονική η τεκμηρίωση θα πρέπει να χρησιμοποιεί απλή, κατανοητή γλώσσα γραφής χωρίς ιδιαίτερες τεχνικές λεπτομέρειες οι οποίες μπορεί να αποθαρρύνουν νέους, χωρίς ιδιαίτερη εμπειρία χρήστες. Θα πρέπει να αποφεύγεται η χρήση ορολογίας ή όταν αυτό δεν είναι εφικτό να προσφέρεται επεξήγηση (εκτός του παραρτήματος ορολογίας). Όπως αναφέρθηκε καινωρίτερα το κείμενο θα πρέπει να είναι λιτό και να μην πλατειάζει με περιττές πληροφορίες.

Αν και υπάρχουν διαφορετικές εναλλακτικές διαρθρώσεις του κειμένου, μια προσέγγιση όπου παρουσιάζονται οι δυνατότητες του λογισμικού ανά εργασία είναι συνήθως αρκετά αποτελεσματική και βοηθά τους χρήστες να ξεκινήσουν να χρησιμοποιούν το λογισμικό γρήγορα. Για παράδειγμα, στην τεκμηρίωση μπορεί να παρουσιάζεται η εργασία ανάκτησης ενός εγγράφου, τι νόημα έχει και τι σκοπό εξυπηρετεί και στη συνέχεια πώς επιτυγχάνεται με το εν λόγω λογισμικό. Χρήστες που έχουν εμπειρία από αντίστοιχα λογισμικά μπορούν να ξεκινήσουν να διαβάζουν κατευθείαν από το σημείο που εξηγεί πώς επιτυγχάνεται η εργασία με το νέο λογισμικό.

Η τεκμηρίωση πρέπει να περιλαμβάνει περιεχόμενα, ευρετήριο όρων και παραρτήματα με γλωσσάριο ορολογίας και πίνακα λαθών. Ο πίνακας λαθών μπορεί να βοηθήσει τους χρήστες να αναγνωρίσουν λάθη και θα πρέπει να περιλαμβάνει οδηγίες αποκατάστασης των λαθών που περιγράφονται. Επιπλέον, ειδικά για ηλεκτρονική τεκμηρίωση θα πρέπει να περιλαμβάνεται σύστημα αναζήτησης πληροφοριών για την εύκολη και γρήγορη ανεύρεση πληροφοριών. Θα πρέπει να σημειωθεί ότι το διαφορετικό μέσο παρουσίασης της έντυπης και της ηλεκτρονικής τεκμηρίωσης συνεπάγεται και διαφορετικό τρόπο παρουσίασης ώστε να είναι αποτελεσματική. Εν γένει, η ηλεκτρονική τεκμηρίωση δεν θα πρέπει να είναι ίδια με την έντυπη αλλά σε διαφορετική μορφή. Χρειάζεται ιδιαίτερο σχεδιασμό ώστε να αποφευχθούν μεγάλες σελίδες (μια σελίδα σε έντυπο αντιστοιχεί σε περισσότερες στην ηλεκτρονική τεκμηρίωση), να δημιουργηθεί λίστα λέξεων-κλειδιών για την αναζήτηση, να δημιουργηθούν κινούμενες εικόνες και προσομοιώσεις και να εκμεταλλευτεί κανείς τις δυνατότητες που του δίνονται από το διαφορετικό μέσο παρουσίασης.

### **Ανθρωπομορφισμός**

Ο ανθρωπομορφισμός στα πλαίσια της επικοινωνίας ανθρώπου μηχανής είναι η απόδοση ανθρώπινων χαρακτηριστικών και ιδιοτήτων στη μηχανή. Ο άνθρωπος θεωρεί ότι η μηχανή μπορεί να σκέφτεται, έχει αισθήματα, να αποφασίζει μόνη της, κλπ. Όπως αναφέρει ο Shneiderman ([15], σελ. 360):

"Οι λέξεις και τα γραφικά μιας διεπαφής μπορεί να προκαλέσουν σημαντικές διαφορές στην αντίληψη, στη συναισθηματική αντίδραση και στα κίνητρα των ανθρώπων. Χαρακτηριστικά ευφυΐας, αυτονομίας, ελεύθερης βούλησης, ή γνώσης σε υπολογιστές μπορούν να ξεγελάσουν, μπερδέψουν ή να οδηγήσουν σε λάθος συμπεράσματα τους χρήστες. Η υπονόηση ότι οι υπολογιστές μπορούν να σκεφτούν, να γνωρίσουν ή να κατανοήσουν, μπορεί να δώσει στους χρήστες ένα λανθασμένο μοντέλο για το πώς λειτουργούν οι υπολογιστές και ποιες είναι οι δυνατότητες της μηχανής."

Η χρήση βοηθών με ανθρωπομορφικά χαρακτηριστικά σε διάφορα λογισμικά προϊόντα (Εικόνα 9) έχει προκαλέσει ποικίλες αντιδράσεις.



**Εικόνα 9. Βοηθός του Office. Η χρησιμοποίηση αντικειμένων με ανθρωπομορφικά χαρακτηριστικά και γλώσσα επικοινωνίας έχει μεγάλο αριθμό τόσο πολέμιων όσο και οπαδών.**

Η συγγραφή κειμένου μπορεί να ακολουθήσει αντίστοιχη μορφή. Για παράδειγμα, μπορείτε να γράψετε:

*Πατώντας τα πλήκτρα Ctrl+S ο υπολογιστής σας θα αποθηκεύσει το κείμενο που γράφετε. Εάν δεν υπάρχει κάποια τροποποίηση από την τελευταία αποθήκευση ο υπολογιστής σας το γνωρίζει και δε θα αποθηκεύσει το κείμενο. Ο υπολογιστής θα κλείσει το πρόγραμμα εάν πατήσετε τα πλήκτρα Alt+F4. Εάν προσπαθήσετε να κλείσετε το πρόγραμμα χωρίς να έχετε αποθηκεύσει τις αλλαγές ο υπολογιστής σας θα σας προειδοποιήσει.*

Το παραπάνω κείμενο αποδίδει ανθρωπομορφικά χαρακτηριστικά στον υπολογιστή και θα πρέπει να αποφεύγεται. Αντί αυτού μπορείτε να γράψετε:

*Για να αποθηκεύσετε το κείμενο που γράφετε πατήστε τα πλήκτρα Ctrl+S. Εάν δεν υπάρχει κάποια τροποποίηση από την τελευταία αποθήκευση το κείμενο δε θα αποθηκευτεί. Για να κλείσετε το πρόγραμμα πατήστε τα πλήκτρα Alt+F4. Εάν προσπαθήσετε να κλείσετε το πρόγραμμα χωρίς να έχετε αποθηκεύσει τις αλλαγές θα δείτε ένα μήνυμα προειδοποίησης.*

Εναλλακτικά μπορείτε να γράψετε:

*Αποθήκευση κειμένου: Πατήστε Ctrl+S*

*Κλείσιμο προγράμματος: Πατήστε Alt+F4*

Το παραπάνω κείμενο είναι σύντομο και περιεκτικό και μπορεί να χρησιμοποιηθεί ιδιαίτερα όταν έχετε ήδη εξηγήσει λεπτομέρειες της λειτουργίας του προγράμματος.

## **Κεφάλαιο 16            Εργαλεία Ανάπτυξης Διεπαφών**

Πριν την έλευση των διεπαφών που στηρίζονται σε γραφικά (GUIs) η ανάπτυξη μιας διεπαφής ήταν αναπόσπαστο τμήμα της διαδικασίας ανάπτυξης ολόκληρου του λογισμικού προϊόντος. Καθώς όμως η χρήση γραφικών διεπαφών και ειδικότερα παραθυρικών διεπαφών εξαπλωνόταν, άρχισαν να εμφανίζονται εργαλεία που είχαν ως στόχο την υποστήριξη του σχεδιαστή και του προγραμματιστή κατά τη διάρκεια ανάπτυξης της διεπαφής. Τα εργαλεία αυτά επικεντρώνονταν στη διεπαφή του λογισμικού προϊόντος προσφέροντας στον προγραμματιστή έτοιμα αντικείμενα σε βιβλιοθήκες (ή εργαλειοθήκες -toolboxes- όπως αλλιώς ονομάζονταν) τα οποία μπορούσε να χρησιμοποιήσει για την υλοποίηση της διεπαφής, ενώ σε αρκετές περιπτώσεις πρόσφεραν ένα γραφικό τρόπο άμεσης διαχείρισης του περιβάλλοντος ανάπτυξης. Ως παραδείγματα εργαλειοθηκών αντικειμένων μπορούμε να αναφέρουμε αυτά του Motif και του Sense 8 World Toolkit. Τα πρώτο χρησιμοποιείται για την ανάπτυξη παραθυρικών εφαρμογών σε περιβάλλον X-Windows, και το δεύτερο για την ανάπτυξη τρισδιάστατων διεπαφών σε περιβάλλον Unix ή Windows. Και στις δύο περιπτώσεις ο κώδικας γράφεται σε C (αν και αυτό δεν ισχύει γενικά για όλες τις εργαλειοθήκες) και ο προγραμματιστής καλεί μέσα στο κώδικα τα αντίστοιχα αντικείμενα ή συναρτήσεις που χρειάζεται χωρίς να χρειάζεται να προγραμματίσει άμεσα τον τρόπο λειτουργίας των αντικειμένων. Στην επόμενη εικόνα παρουσιάζεται ο κώδικας για την εμφάνιση ενός παραθύρου με ένα πλήκτρο σε Motif. Ο προγραμματιστής καλεί τις συναρτήσεις της εργαλειοθήκης για να δημιουργήσει ένα παράθυρο, ένα πλήκτρο, να συνδέσει μια συνάρτηση με τα συμβάντα του πλήκτρου (να καλείται η συνάρτηση όταν ο χρήστης πιέζει το πλήκτρο) και στη συνέχεια για να εμφανίσει όλη την ιεραρχία αντικειμένων στην οθόνη.

```

#include <Xm/Xm.h>
#include <Xm/PushB.h>
/* Το πρόγραμμα δημιουργεί ένα παράθυρο με ένα πλήκτρο το οποίο όταν πιεσθεί
κλείνει το παράθυρο */

/* Πρωτότυπο συνάρτησης κλήσης */
void pushed_fn(void);

main(int argc, char **argv){

    Widget top_wid, button;
    XtAppContext app;

    top_wid = XtVaAppInitialize(&app, "First", NULL, 0, &argc, argv, NULL, NULL);

    button = XmCreatePushButton(top_wid, "Quit", NULL, 0);

    XtManageChild(button);

    /* σύνδεση συνάρτησης στο πλήκτρο */
    XtAddCallback(button, XmNactivateCallback, pushed_fn, NULL);

    /* δημιουργία ιεραρχίας αντικειμένων */
    XtRealizeWidget(top_wid);

    /* επανάληψη επεξεργασίας μηνυμάτων */
    XtAppMainLoop(app);
}

void pushed_fn(void) {
    exit(0)
}

```

### Εικόνα 10 Πρωτογενής κώδικας για την εμφάνιση ενός παραθύρου με ένα πλήκτρο σε περιβάλλον X Windows

Με την εξάπλωση του διαδραστικού μοντέλου της απευθείας/ άμεσης διαχείρισης, εμφανίστηκαν προγραμματιστικά εργαλεία όπου ο προγραμματιστής επέλεγε τα αντικείμενα που ήθελε να εμφανίζονται στην οθόνη από αντίστοιχες λίστες εικονιδίων και τα τοποθετούσε στο παράθυρο της υπο-κατασκευή εφαρμογής.

Ανάλογα με την εταιρεία που ανέπτυξε το εργαλείο έχουν χρησιμοποιηθεί διαφορετικά ονόματα για να χαρακτηρίσουν αυτήν την κατηγορία εργαλείων όπως κατασκευαστής διεπαφών (user interface builder) και περιβάλλον γρήγορης ανάπτυξης εφαρμογών (rapid application development environment). Παρότι παλαιότερα υπήρχαν περιβάλλοντα που μπορούσαν να χρησιμοποιηθούν μόνο κατά τη διάρκεια του σχεδιασμού και της πρωτοτυποποίησης, σήμερα τα περισσότερα εργαλεία μπορούν να χρησιμοποιηθούν και κατά τη φάση της υλοποίησης για την παραγωγή κώδικα.

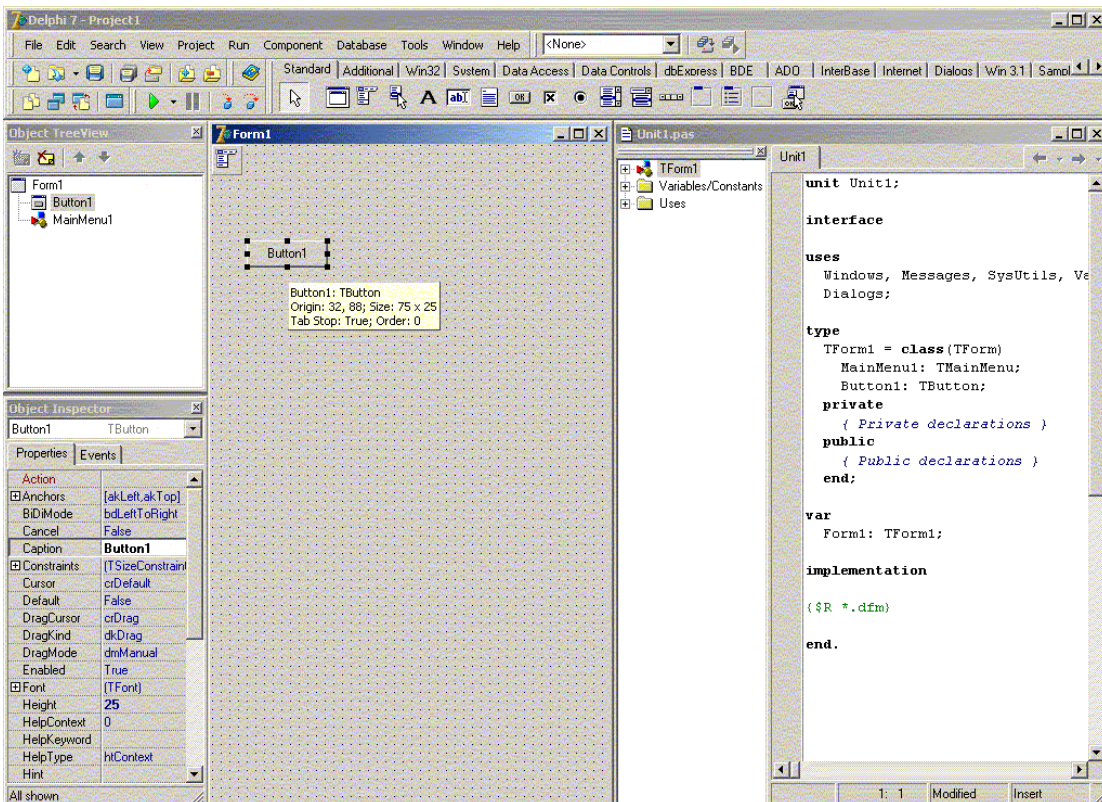
Τα σημαντικότερα πλεονεκτήματα αυτών των εργαλείων είναι:

- Διαχωρισμός διεπαφής από το υπόλοιπο λογισμικό
- Υποστήριξη του σχεδιασμού
- Υποστήριξη στη συνεργασία με μη-τεχνικούς
- Δοκιμή εναλλακτικών σχεδιαστικών προσεγγίσεων πολύ νωρίς
- Επιβολή προτύπων
- Ευκολία στη διατήρηση οδηγιών, ομοιομορφίας και συνέπειας

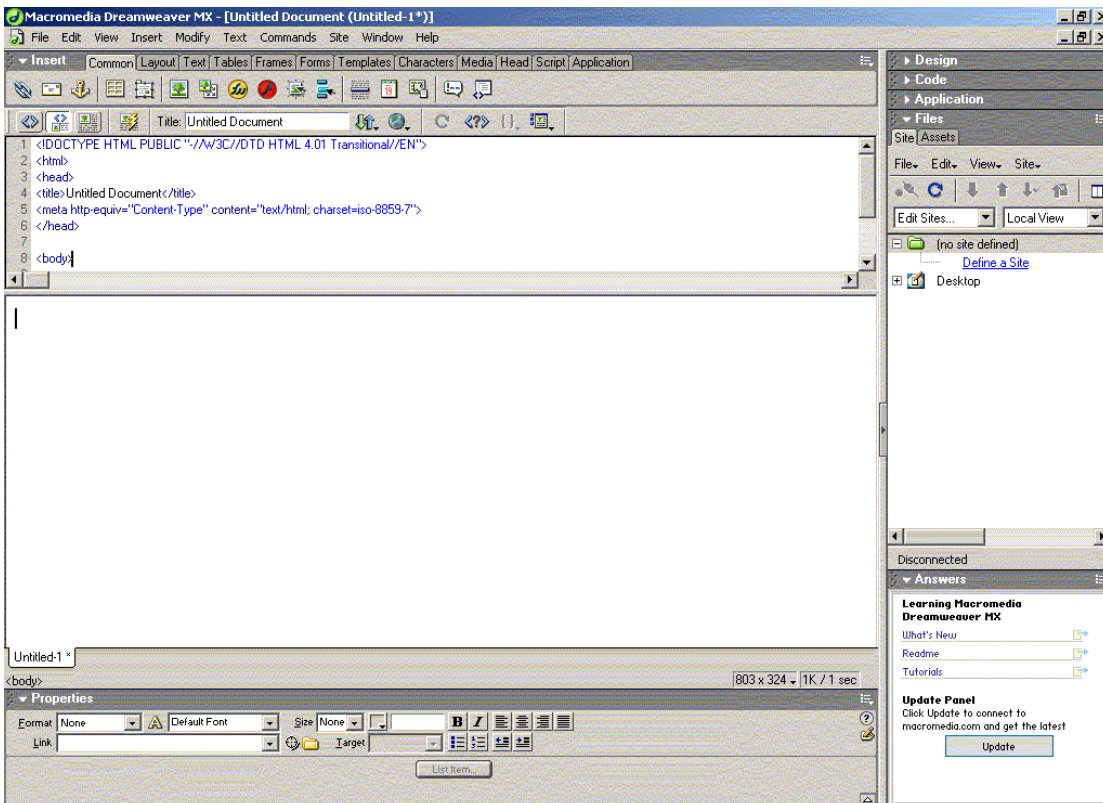
- Υποστήριξη της ομαδικής εργασίας
- Μικρός κύκλος μεταξύ σχεδιασμού-υλοποίησης και αξιολόγησης
- Αυξημένη παραγωγικότητα
- Ευκολία στην τεκμηρίωση και τη συγγραφή βοήθειας
- Ευκολία στη συντήρηση
- Μεταφερσιμότητα κώδικα μεταξύ διαφορετικών συστημάτων

Ένας απλός διαχωρισμός μεταξύ των εργαλείων μπορεί να γίνει ανάλογα με την προγραμματιστική βάση που χρησιμοποιούν για την υλοποίηση της λειτουργικότητας της διεπαφής. Υπάρχουν εργαλεία που χρησιμοποιούν κάποια γλώσσα προγραμματισμού όπως C/C++, Java, Pascal ή Basic και άλλα που στηρίζονται σε κάποια scripting γλώσσα όπως JavaScript. Παραδείγματα της πρώτης κατηγορίας είναι η MS Visual C++, η Borland Java, η Borland Delphi, η MS Basic, ενώ στη δεύτερη κατηγορία ανήκουν εργαλεία όπως το Macromedia Dreamweaver, το Sense 8 World Up και το Platinum VRML Editor.

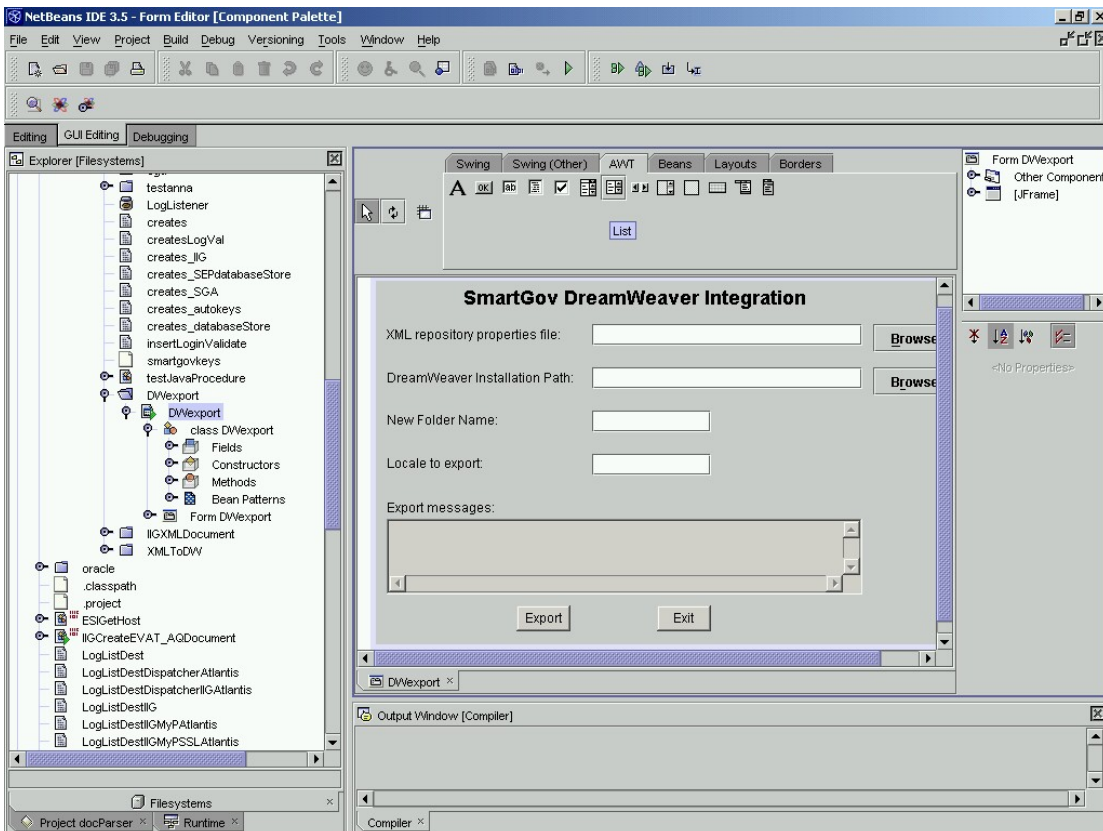
Όλα τα εργαλεία διαθέτουν μερικά κοινά λειτουργικά χαρακτηριστικά όπως φαίνεται και στις επόμενες εικόνες. Ο προγραμματιστής έχει τη δυνατότητα να επιλέξει αντικείμενα από λίστες αντικειμένων, να τα τοποθετήσει στη φόρμα του παραθύρου, να ορίσει χαρακτηριστικά τους ή να γράψει πρωτογενή κώδικα για να ελέγξει τον τρόπο λειτουργίας τους.



**Εικόνα 11 Περιβάλλον εργασίας στη Borland Delphi**



Εικόνα 12 Περιβάλλον εργασίας του Macromedia Dreamweaver



Εικόνα 13 Περιβάλλον εργασίας του NetBeans



# **ΜΕΡΟΣ V ΠΑΡΑΡΤΗΜΑΤΑ**

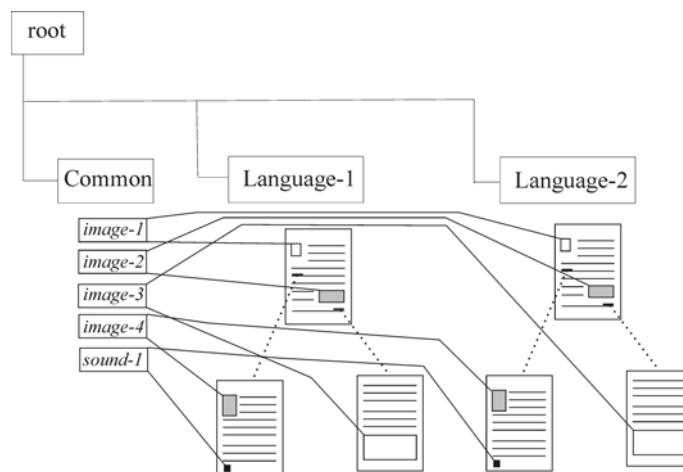


## Κεφάλαιο 17 Παράρτημα Α: Στοιχεία HTML και Javascript

Η υλοποίηση συστημάτων ΕΑΜ μπορεί να γίνει σε πάρα πολλές γλώσσες και με πάρα πολλούς τρόπους ανάλογα με τις ανάγκες του περιβάλλοντος που θα μπει το σύστημα, των χρηστών, και της εφαρμογής. Στο κεφάλαιο αυτό παραθέτουμε μερικά εισαγωγικά στοιχεία για τις γλώσσες HTML και Javascript, οι οποίες χρησιμοποιούνται στην εργασία που συνοδεύει την βασική ύλη των διαλέξεων αυτό το εξάμηνο, και η οποία πραγματεύεται μία εφαρμογή στο διαδίκτυο. Να σημειώσουμε αναφορικά ότι το διαδίκτυο (internet) ή Παγκόσμιο Πλέγμα Πληροφοριών (World Wide Web) ακολουθεί το μοντέλο εξυπηρετούμενου-εξυπηρετή (client-server) και χρησιμοποιεί το πρωτόκολλο HTTP (Hyper Text Transfer Protocol) για την επικοινωνία ανάμεσα στους κόμβους του. Ως παραδείγματα εξυπηρετών (Web servers) μπορούμε να αναφέρουμε τους Internet Information Server, Apache, NCSA, και Netscape, ενώ ως εξυπηρετούμενους ή προγράμματα πλοήγησης (Web clients ή browsers) τους Netscape Navigator, Internet Explorer, Tango, κτλ. Σήμερα τα περισσότερα προγράμματα πλοήγησης υποστηρίζουν διάφορους τύπους δεδομένων (κείμενο, γραφικά, ήχο, βίντεο) και διάφορα πρωτόκολλα επικοινωνίας (FTP, Gopher, HTTP, NNTP).

### 17.1 Υπερκείμενα

Τα υπερκείμενα διαφέρουν από τα απλά κείμενα στη δυνατότητά τους να διασυνδέονται, παρακάμπτοντας το γραμμικό τρόπο παρουσίασης της πληροφορίας.



### 17.2 Τι είναι η HTML;

Η HTML είναι μια γλώσσα με την οποία περιγράφουμε την μορφοποίηση μιας σελίδας. Κάθε εντολή ή διακριτικό (tag) της HTML περικλείεται σε  $\langle \rangle$  (μικρότερο-μεγαλύτερο), ώστε να ξεχωρίζει από το υπόλοιπο κείμενο. Τα περισσότερα διακριτικά ομαδοποιούνται σε ζεύγη. Το πρώτο δείχνει την αρχή και το δεύτερο (το οποίο ξεκινά με  $\langle / \rangle$ ) το τέλος επίδρασης του συγκεκριμένου διακριτικού.

Παράδειγμα

Η δήλωση:

$\langle I \rangle$  Αυτό είναι δείγμα πλάγιας γραφής  $\langle / I \rangle$

θα εμφανιστεί ως:

*Αυτό είναι δείγμα πλάγιας γραφής*

Το `<I>` δείχνει την αρχή του κειμένου που θα εμφανιστεί πλάγιο, ενώ το `</I>` δείχνει το τέλος του κειμένου.

Παράδειγμα (συνέχεια)

Οι δηλώσεις:

`<I>` Αυτό είναι δείγμα πλάγιας γραφής `</I>`

και

`<i>` Αυτό είναι δείγμα πλάγιας γραφής `</i>`

είναι ισοδύναμες.

### 17.3 Δομή εγγράφου HTML

Η γενική δομή ενός εγγράφου HTML είναι η ακόλουθη:

```
<HTML> <HEAD>
<TITLE> Τίτλος κειμένου</TITLE>
</HEAD>
<BODY>
Κυρίως κείμενο
</BODY>
</HTML>
```

Παράδειγμα

```
<HTML><HEAD>
<TITLE>This is a demo </TITLE>
</HEAD>
<BODY>
Αυτό είναι ένα δοκιμαστικό έγγραφο
HTML.
Αυτή η γραμμή και
αυτή η γραμμή θα εμφανιστούν μαζί.
</BODY></HTML>
```

Όπως φαίνεται στο παραπάνω παράδειγμα η HTML αδιαφορεί για χαρακτήρες αλλαγής γραμμής που περιέχονται στο κείμενο. Εάν πρέπει να αλλάξει γραμμή πρέπει να δηλωθεί. Στη συνέχεια παρουσιάζονται ορισμένα συνηθισμένα διακριτικά.

#### Διακριτικά Επικεφαλίδες

#### μορφοποίησης

#### κειμένου

```
<H1>Επικεφαλίδα πρώτου επιπέδου</H1>
<H2>Επικεφαλίδα δεύτερου επιπέδου</H2>
<H3>Επικεφαλίδα τρίτου επιπέδου</H3>
```

#### Πλάγια-Έντονα-Κεντραρισμένα

`<I>` Πλάγια γραφή `</I>`

Θα εμφανίσει: *Πλάγια γραφή*

`<B>` Έντονη γραφή `</B>`

Θα εμφανίσει: **Έντονη γραφή**

`<CENTER>` Κεντραρισμένο `</CENTER>`

Θα εμφανίσει: Κεντραρισμένο

## Παράγραφοι - Αλλαγή γραμμής

<P> Αρχίζει νέα παράγραφο  
Για ομοιομορφία με τα άλλα διακριτικά υπάρχει και το </P>  
<BR> Ορίζει μια αλλαγή γραμμής

## Λίστες

### Αριθμημένη λίστα

```
<OL>
  <LI>Πρώτο θέμα
  <LI>Δεύτερο θέμα
</OL>
```

### Λίστα με κουκίδες

```
<UL>
  <LI> θέμα
  <LI> θέμα
</UL>
```

### Λίστες ορισμού

```
<DL>
  <DT> Πρώτο θέμα
  <DD> Ορισμός 1ου θέματος
  <DT> Δεύτερο θέμα
  <DD> Ορισμός 2ου θέματος
</DL>
```

## Καθολικός Προσδιοριστής Πόρου (URL)

### Σύνταξη:

Μέθοδος πρόσβασης://διεύθυνση/ διαδρομή καταλόγου/αρχείο

Καθορίζει μια διεύθυνση ενός αντικειμένου (σελίδας HTML, εικόνας, κλπ.) στο Internet.

### Παραδείγματα

```
http://www.di.uoa.gr/Welcome.html
ftp://sunsite.doc.ic.ac.uk/computing
telnet://simpson-01.cs.strath.ac.uk
news:alt.hypertext
mailto://http@di.uoa.gr
```

## Σύνδεσμοι

### Η δήλωση:

```
<A HREF="http://www.di.uoa.gr/Welcome.html"> Σελίδα Τμήματος  
Πληροφορικής </A>
```

θα δημιουργήσει ένα σύνδεσμο στο αντίστοιχο έγγραφο.

Πιέζοντας στο Σελίδα Τμήματος Πληροφορικής μεταφερόμαστε στο έγγραφο Welcome.html που βρίσκεται στο δικτυακό τόπο www.di.uoa.gr.

### Εικόνες

Η δήλωση `<IMG SRC="Fig1.gif" ALT="Figure 1">` εισάγει στη σελίδα την εικόνα που βρίσκεται στο αρχείο Fig1.gif . Σε περίπτωση που δεν υπάρχει το αρχείο θα εμφανίσει ένα προκαθορισμένο εικονίδιο.

### Ειδικοί χαρακτήρες

Οι χαρακτήρες `<`, `>`, `&` και `"` είναι ειδικοί χαρακτήρες της HTML. Σε ένα κείμενο αντικαθίστανται ως εξής:

```
<    &lt;
>    &gt;
&    &amp;
"    &quot;
```

### Περισσότερες δυνατότητες

Η δήλωση `<HR>` εμφανίζει μια οριζόντια γραμμή., ενώ με το διακριτικό `<PRE> </PRE>` το κείμενο που περικλείεται μέσα σε αυτό θα εμφανιστεί όπως είναι στοιχισμένο (χρησιμοποιώντας κάποια γραμματοσειρά σταθερού μεγέθους, όπως Courier).

### Πίνακες

Ένας πίνακας ορίζεται ως εξής:

```
<Table>
<tr><td>1ο κελί πρώτης γραμμής</td>
<td>2ο κελί πρώτης γραμμής</td></tr>
<tr><td>1ο κελί δεύτερης γραμμής</td>
<td>2ο κελί δεύτερης γραμμής</td></tr>
</Table>
```

Ο πλήρης ορισμός των διακριτικών είναι ο ακόλουθος:

```
<table border=5 bordercolor=blue align=center width=50% height=123
cellpadding=3 cellspacing=5 bgcolor=yellow background=bg.gif>
<tr align=right valign=bottom bgcolor=blue bordercolor=yellow
background=bg.gif>
<td align=right valign=top bgcolor=green bordercolor=black
height=50 width=100 colspan=2 rowspan=3 background=bg.gif>
```

Ο ακόλουθος πίνακας

A	ΠΕΡΙΕΧΟΜΕΝΑ		
P	Κεφάλαιο 1	Κεφάλαιο 2	Κεφάλαιο 3
X	Κεφάλαιο 4	Κεφάλαιο 5	Κεφάλαιο 6
H	Κεφάλαιο 7	Παράρτημα Α	Παράρτημα Β

ορίζεται ως εξής:

```
<table border=1 cellspacing=0 cellpadding=0>
<tr>
<td width=34 rowspan=4 valign=top>
<center><p>A</p>
```

```

    <p>P</p>
    <p>X</p>
    <p>H</p></center>
</td>
<td colspan=3 valign=middle>
<center><p>ΠΕΡΙΕΧΟΜΕΝΑ</p></center>
</td>
</tr>
<tr>
<td valign=top> <p>Κεφάλαιο 1</p> </td>
<td valign=top> <p>Κεφάλαιο 2</p> </td>
<td valign=top> <p>Κεφάλαιο 3</p> </td>
</tr>
<tr>
<td valign=top> <p>Κεφάλαιο 4</p> </td>
<td valign=top> <p>Κεφάλαιο 5</p> </td>
<td valign=top> <p>Κεφάλαιο 6</p> </td>
</tr>
<tr>
<td valign=top> <p>Κεφάλαιο 7</p> </td>
<td valign=top> <p>Παράρτημα Α</p> </td>
<td valign=top> <p>Παράρτημα Β</p> </td>
</tr>
</table>

```

## Πλαίσια

Τα πλαίσια χρησιμοποιούνται για την εμφάνιση περισσότερων της μιας σελίδας σε ένα παράθυρο. Χρησιμοποιούν δύο διακριτικά, το FRAMESET και το FRAME. Το FRAMESET λέει στο πρόγραμμα πλοήγησης πώς να χωρίσει το παράθυρο σε γραμμές και στήλες και το FRAME τι να τοποθετήσει μέσα στις γραμμές και τις στήλες.

Πλαίσια (παράδειγμα)

```

<frameset cols="30%,70%">
<frame name="left" src="menu.html">
<frame name="right" src="main.html">
</frameset>

```

Τι αποτέλεσμα θα έχει κάθε μια από τις επόμενες δηλώσεις;

```

<frameset rows="20%, *, 2*">
<frameset rows="150, *">
<frameset rows="20, *, 20%">

```

Πρώτα υπολογίζονται οι απόλυτες τιμές (pixels)

Μετά υπολογίζονται τα ποσοστά ως προς το σύνολο του παραθύρου

Οι αναλογικές τιμές υπολογίζονται τελευταίες με βάση ό,τι έχει απομείνει.

Ο επόμενος κώδικας:

```

<HTML>
<HEAD>

```

```

<TITLE>Welcome to the Department of Informatics</TITLE>

```

```

<BASE href="http://www.di.uoa.gr/Welcome.html">

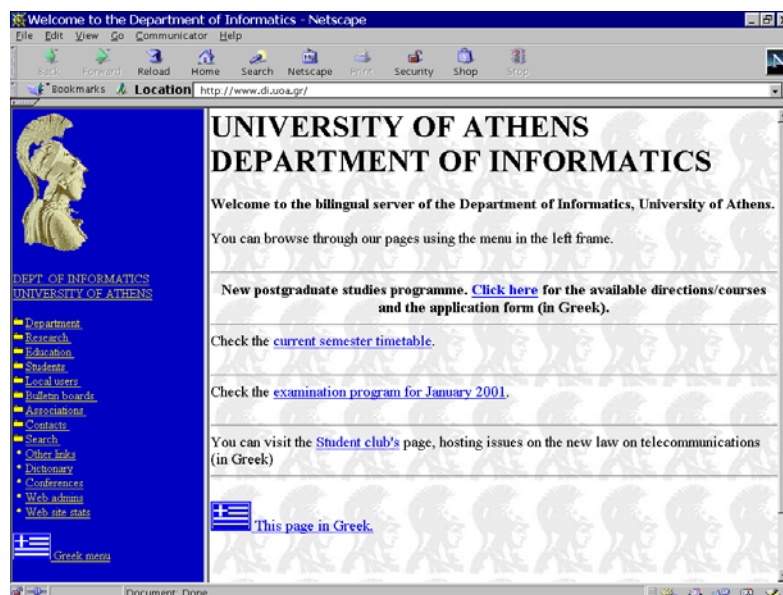
</HEAD>
<FRAMESET ROWS="100%">
<FRAMESET COLS="25%,75%">
<FRAME SRC="/English/menu/menu.shtml" SCROLLING="auto" NAME="left-
frm" MARGINWIDTH="0" MARGINHEIGHT="0" NORESIZE>

<FRAME SRC="/English/menu/Cell2.html" SCROLLING="auto"
NAME="right-frm" MARGINWIDTH="0" MARGINHEIGHT="0">

</FRAMESET>
<NOFRAMES>
It seems that your Web Browser does not support frames!<BR>
Don't panic!<BR>
Press <A HREF="http://www.di.uoa.gr/Welcome-
noframes.html">here</A> to view
the page without frames.
</NOFRAMES>
</HTML>

```

Θα εμφανίσει την ακόλουθη σελίδα:



Η σελίδα που εμφανίζεται στο πλαίσιο με όνομα *left-frm* είναι η <http://www.di.uoa.gr/English/menu/menu.shtml> και στο πλαίσιο με όνομα *right-frm* είναι η <http://www.di.uoa.gr/English/menu/Cell2.html>.

### Φόρμες

Οι φόρμες επιτρέπουν τη δημιουργία σελίδων οι οποίες δίνουν τη δυνατότητα αποστολής δεδομένων από τον χρήστη. Σε μια σελίδα είναι δυνατό να παρουσιασθούν τα ακόλουθα αντικείμενα:

- Πλαίσια εισαγωγής κειμένου



Εισαγωγή email

- Πλαίσια εισαγωγής συνθηματικών

Εισαγωγή συνθηματικού:

- Πλαίσια εισαγωγής κειμένου πολλαπλών γραμμών

- Πλαίσια λίστας

- Πλαίσια πτυσσόμενης λίστας

- Πλαίσια επιλογής

Τι θέλετε;  
 Αναλυτική βαθμολογία  
 Βεβαίωση εφορίας

- Ραδιοπλήκτρα

Έκδοση σε 2 εργασίες;  
 Ναι  
 Όχι

- Πλήκτρα

Μια φόρμα ορίζεται με το διακριτικό <FORM>. Με την αποστολή της φόρμας εκτελείται η συνδεδεμένη διαδικασία:

```
<form action="mailto:user@di.uoa.gr" method=post>
```

Στην παραπάνω δήλωση τα δεδομένα αποστέλλονται μέσω e-mail στο χρήστη user@di.uoa.gr. Αντίστοιχα μπορεί να χρησιμοποιηθεί το http πρωτόκολλο για την αποστολή των δεδομένων σε κάποιο πρόγραμμα επεξεργασίας των δεδομένων.

*Πλαίσια εισαγωγής κειμένου*

```
<br>Εισαγωγή email<input type="text" name="e-mail" size=30  
value="glepourea@di.uoa.gr" maxlength=50>
```

*Πλαίσια εισαγωγής συνθηματικών*

```
<br>Εισαγωγή συνθηματικού: <input type="password" name="passwd"  
size=10 maxlength=10>
```

*Πλαίσια εισαγωγής κειμένου πολλαπλών γραμμών*

```
<textarea name="feedback" rows=4 cols=40 >  
Χώρος για σχόλια </textarea>
```

*Πλαίσια λίστας*

```
<select name= "list" size=2 multiple>
  <option value="1" >Ένα αντίγραφο
  <option value="2" >Δύο αντίγραφα
</select>
```

Η διαφορά του πλαισίου λίστας από αυτό της πτυσσόμενης λίστας είναι ότι στο πλαίσιο λίστας είναι ορατές περισσότερες από μια επιλογές (στο παράδειγμα size=2) και επιτρέπονται περισσότερες από μια ταυτόχρονες επιλογές, ενώ στην πτυσσόμενη λίστα μόνο μια επιλογή (size=1, ή παράλειψη του size για μια και 1 είναι η προκαθορισμένη τιμή).

*Πλαίσια πτυσσόμενης λίστας*

```
<select name= "pulldownlist">
  <option value="1" >Ένα αντίγραφο
  <option value="2" >Δύο αντίγραφα
</select>
```

*Πλαίσια επιλογής*

```
<br>Τι θέλετε;
<br> Αναλυτική βαθμολογία <input type="checkbox" name="grade"
value="yes" checked>
<br> Βεβαίωση εφορίας <input type="checkbox" name="irs"
value="yes" checked>
```

*Ραδιοπλήκτρα*

```
<br>Έκδοση σε 2 εργάσιμες;
<br><input type="radio" name="receipt" value="yes" checked> Ναι
<br> <input type="radio" name="receipt" value="no" > Όχι
```

*Πλήκτρα*

```
<br><input type="reset" value="Διαγραφή">
<input type="submit" value="Αποστολή">
```

**Explorer vs Netscape***Διακριτικά μόνο για Explorer*

```
<marquee behavior=scroll direction=left srolldelay=100
scrollamount=40> Το κείμενο αυτό κινείται από δεξιά στα αριστερά,
με βήμα 40 pixels και καθυστέρηση 100 milliseconds </marquee>
<bgsound src= "tune.wav" loop="3">
```

*Διακριτικά μόνο για Netscape*

```
<multicol cols=3 gutter=30 width=800>Δημιουργία στηλών στο
Netscape</multicol>
<blink>Κείμενο που αναβοσβήνει</blink>
```

**Εισαγωγή video-ήχου**

```
<embed src= "video.avi">
<embed src= "sound.wav">
```

## Επιπλέον οδηγίες

Χρησιμοποιώντας από το πρόγραμμα πλοήγησης την επιλογή View -> Page Source, μπορείτε να εξετάσετε τον κώδικα HTML που υλοποιεί την σελίδα που βλέπετε και να πάρετε χρήσιμες ιδέες. Αναλυτικούς οδηγούς της γλώσσας HTML θα βρείτε στις επόμενες διευθύνσεις:

<http://www.w3.org/>

<http://www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html>

<http://www-leland.stanford.edu/~ttorres/Internet/toc.html>

<http://www.emerson.emory.edu/services/html/html.html>

<http://werbach.com/barebones/>

<http://developer.netscape.com/docs/manuals/htmlguid/index.htm> ή τοπικά στο

<http://www.di.uoa.gr/~rouvas/HTML-tagref/index.htm>

<http://www.eeei.gr/odhgos/htmlfaq/howpage.htm>

## 17.4 Στοιχεία Javascript

### Γενικά περί JavaScript

Η Javascript είναι η γλώσσα που μετατρέπει τα παθητικά από άποψη διάδρασης με τον χρήστη έγγραφα σε ενεργητικά. Μια πληθώρα από αντιδράσεις στις λειτουργίες του χρήστη είναι δυνατόν να προγραμματιστούν, χωρίς την ανάγκη επικοινωνίας με τον εξυπηρετή (web server) από τον οποίο έχει προέλθει το έγγραφο.

Η Javascript είναι γλώσσα διερμηνευόμενη (interpreted), δηλαδή εκτελείται χωρίς να μεταγλωττιστεί (compiled) πρώτα. Για τον λόγο αυτό δεν είναι κατάλληλη για δραστηριότητες που απαιτούν εντατική χρήση του υπολογιστή. Είναι όμως κατάλληλη για την υλοποίηση διεπαφών και την επικοινωνία χρήστη-υπολογιστή.

Η Javascript μπορεί να δημιουργήσει δυναμικά κώδικα HTML ο οποίος τυπικά εμφανίζεται σε ένα νέο παράθυρο, και έτσι δημιουργείται ένας νέος διάλογος ή παρουσιάζονται δεδομένα στο χρήστη. Επίσης η πλοήγηση σε μια σελίδα HTML μπορεί να προκαλέσει κλήση συναρτήσεων Javascript που μπορούν να την παρουσιάσουν διαφορετικά στον χρήστη.

Τέλος η Javascript μοιάζει πολύ με άλλες γλώσσες προγραμματισμού. Η σύνταξη της είναι απλοποιημένη C, ενώ διαθέτει και αντικειμενοστραφείς λειτουργίες όπως η Java ή η C++.

Η Javascript διαθέτει:

- συναρτήσεις (function add(a,b) { return a + b } ),
- ενιαίους τύπους με αυτόματη μετατροπή (a=1; b='2' b=a + b),
- υπερφόρτωση: τύπων δεδομένων (add('1', 3) -> 4), και
- αντικειμενοστραφείς λειτουργίες (var ctr = new counter(); counter.start()).

Επίσης μπορεί να προσπελάσει την ιεραρχία των πλαισίων ενός προγράμματος πλοήγησης (π.χ. self.parent.frame1), καθώς και βασικές λειτουργίες του (π.χ. history.go(-1)).

Χρησιμοποιώντας τη JavaScript είναι δυνατό να γραφτούν προγράμματα που ελέγχουν τα συμβάντα που σχετίζονται με αντικείμενα της σελίδας, ελέγχουν δεδομένα τη στιγμή που εισάγονται στη σελίδα και εκτελούν υπολογισμούς στην πλευρά του προγράμματος πλοήγησης. Για να εκτελεστεί η JavaScript χρειάζεται ένα πρόγραμμα πλοήγησης που να την υποστηρίζει (όπως Netscape 4.0, Internet Explorer 4.0 ή νεώτερες εκδόσεις).

## Εκδόσεις της JavaScript

Υπάρχουν αρκετές εκδόσεις της JavaScript που υποστηρίζονται από διαφορετικά προγράμματα πλοήγησης και εκδόσεις αυτών. Το γεγονός αυτό έχει οδηγήσει σε ασυνέπειες. Μια προσπάθεια προτυποποίησης έγινε από την ECMA. Τελικά αν και ο προγραμματισμός με JavaScript είναι αρκετά απλός, η δημιουργία προγραμμάτων τα οποία να έχουν τη δυνατότητα να εκτελεστούν ορθά σε διαφορετικά προγράμματα πλοήγησης (π.χ. Netscape Navigator 4.\* και MS Internet Explorer 5.\*) μπορεί να αποδειχθεί πολύ δύσκολη.

## Ενσωμάτωση της JavaScript σε HTML

Ο κώδικας της JavaScript ενσωματώνεται στην HTML χρησιμοποιώντας το διακριτικό (tag) *SCRIPT*.

```
<SCRIPT language="JavaScript">
<!-- Χρησιμοποιούμε διακριτικά σχολίων για να μην εμφανίζεται ο
<!-- κώδικας JavaScript σε παλαιότερα προγράμματα πλοήγησης
document.write("Καλημέρα!");
//-->
</SCRIPT>
```

Τον κώδικα JavaScript μπορείτε πρακτικά να το τοποθετήσετε σε οποιοδήποτε σημείο του αρχείου HTML. Οι συναρτήσεις συνηθίζεται να δηλώνονται μέσα στο τμήμα HEAD της HTML σελίδας, επειδή αυτό το τμήμα διαβάζεται πρώτο, οπότε εξασφαλίζεται ότι οι συναρτήσεις θα είναι έτοιμες προς εκτέλεση πριν ο χρήστης μπορεί να κάνει κάτι. Οι κλήσεις JavaScript που εκτελούν κάποια αλληλεπίδραση με τον χρήστη τοποθετούνται στο τμήμα BODY της HTML σελίδας.

Η παράμετρος language είναι προαιρετική, αλλά χρήσιμη επειδή επιτρέπει να καθορίσετε ότι ένα τμήμα κώδικα εκτελείται μόνο από προγράμματα πλοήγησης που υποστηρίζουν μια ιδιαίτερη έκδοση της JavaScript. Μπορείτε επίσης να αποθηκεύσετε τον κώδικα JavaScript σε εξωτερικά αρχεία με κατάληξη .js:

```
<script language="JavaScript" src="somefunction.js"></script>
```

Προσοχή: Η παράμετρος src δεν υποστηρίζεται από κάποια παλαιότερα προγράμματα πλοήγησης.

## Γραμματική της JavaScript

Η JavaScript, όπως και οι άλλες γλώσσες προγραμματισμού, αποτελούνται από δηλώσεις (που μας επιτρέπουν να εκτελούμε καταχωρήσεις, συγκρίσεις, κλπ.). Οι σημειώσεις προϋποθέτουν τη γνώση κάποιας γλώσσας προγραμματισμού, οπότε και η χρήση μεταβλητών, τελεστών και δηλώσεων δεν θα είναι άγνωστη.

### Μεταβλητές και τύποι δεδομένων

Τα ονόματα μεταβλητών στη JavaScript ξεκινούν είτε με γράμμα ("A".."Z", "a".."z") είτε με κάτω παύλα ("\_"). Οι επόμενοι χαρακτήρες μπορεί να είναι αριθμητικοί χαρακτήρες ("0".."9"). Η JavaScript κάνει διάκριση ανάμεσα σε πεζούς και κεφαλαίους χαρακτήρες. Άρα η μεταβλητή X είναι διαφορετική από τη x.

Στη JavaScript τα δεδομένα μπορεί να ανήκουν σε κάποιον από τους παρακάτω τύπους:

Τύπος Δεδομένων	Περιγραφή
Αριθμητικά Ακέραιοι	- Μπορεί να βρίσκονται στο οκταδικό, δεκαδικό ή δεκαεξαδικό σύστημα. Οι οκταδικοί αριθμοί αρχίζουν με 0 (μηδέν) και οι δεκαεξαδικοί με 0x. Οι οκταδικοί περιλαμβάνουν τα ψηφία (0-7) και οι δεκαεξαδικοί 0-9 και A-F.
Αριθμητικοί Κινητής Υποδιαστολής	- Μπορεί να είναι αριθμοί με δεκαδικό μέρος ή αριθμοί της μορφής α.βEε που σημαίνει α,β x 10 <sup>ε</sup> .
Λογικές (Boolean)	True και False. Κάθε έκφραση με τιμή ίση με το 0 αντιστοιχεί σε

	False και με τιμή ίση με 1 σε True.
Συμβολοσειρές	Όπως "Καλημέρα". Ορίζονται με διπλά ή απλά εισαγωγικά.
Null	Σημαίνει ότι η μεταβλητή δεν έχει καμιά τιμή.
Απροσδιόριστη (NaN)	Η τιμή που έχει μια μεταβλητή μετά τη δημιουργία της και πριν της δοθεί κάποια τιμή.

Δε χρειάζεται να έχετε δηλώσει μια μεταβλητή νωρίτερα (όπως π.χ. στη Pascal ή τη C). Μια μεταβλητή ορίζεται όταν της δώσετε τιμή π.χ. `neaMetabliti=9`; Η νέα μεταβλητή `neaMetabliti` θα θεωρηθεί αυτόματα αριθμητικού τύπου. Εναλλακτικά μπορείτε να ορίσετε μια μεταβλητή χρησιμοποιώντας το `var`, π.χ. `var x`; Στη δεύτερη περίπτωση η μεταβλητή `x` έχει ορισθεί και η τιμή της είναι NaN. Η JavaScript δεν εκτελεί ιδιαίτερους ελέγχους στους τύπους των μεταβλητών. Είναι δυνατή η χρήση μεταβλητών χωρίς να έχουν προηγουμένως δηλωθεί και η μετατροπή μεταξύ διαφορετικών τύπων. Όταν αναμειγνύονται μεταβλητές αριθμητικού τύπου με συμβολοσειρές τότε στην περίπτωση που ο τελεστής είναι το συν (+) θεωρείται ότι όλες οι μεταβλητές είναι συμβολοσειρές, ενώ εάν ο τελεστής είναι το πλην (-) οι μεταβλητές θεωρούνται αριθμητικού τύπου. Οπότε η έκφραση `x = 12 + "14"` και η `x = "12" + 14` θα επιστρέψουν αποτέλεσμα "1214", ενώ η `x=12-"7"` θα επιστρέψει 5 και η `x = 12-"a"` θα επιστρέψει NaN.

#### Πίνακες

Ένας πίνακας είναι ένα αντικείμενο που ορίζεται ως εξής:

```
var pinakas=new Array(arrayLength);
```

ή

```
var pinakas=new Array(element0, element1, ..., elementN);
```

#### Εύρος/ Εμβέλεια

Όταν δίνετε μια τιμή σε μια νέα μεταβλητή, πρέπει να προσέξετε το θέμα του εύρους. Μια μεταβλητή μπορεί να είναι καθολική ή τοπική. Μια καθολική μεταβλητή μπορεί να κληθεί από οποιοδήποτε κώδικα JavaScript μέσα στη σελίδα. Μια τοπική μεταβλητή μπορεί να κληθεί μόνο μέσα στη συνάρτηση που έχει ορισθεί. Οι μεταβλητές που ορίζονται μέσα σε μια συνάρτηση έχουν τοπική εμβέλεια.

#### Τελεστές

Στους επόμενους πίνακες παρουσιάζονται οι πιο συχνά χρησιμοποιούμενοι τελεστές.

#### Τελεστές συσχέτισης

Τελεστής	Περιγραφή
Ισότητα (==)	Επιστρέφει true εάν οι τελεστέοι έχουν την ίδια τιμή.
Ανισότητα (!=)	Επιστρέφει true εάν οι τελεστέοι δεν έχουν την ίδια τιμή.
Μεγαλύτερο (>)	Επιστρέφει true εάν ο αριστερός τελεστέος έχει μεγαλύτερη τιμή από τον αριστερό τελεστέο.
Μεγαλύτερο ή ίσο (>=)	Επιστρέφει true εάν ο αριστερός τελεστέος έχει μεγαλύτερη ή ίση τιμή από τον αριστερό τελεστέο.
Μικρότερο (<)	Επιστρέφει true εάν ο αριστερός τελεστέος έχει μικρότερη τιμή από τον αριστερό τελεστέο.
Μικρότερο ή ίσο (<=)	Επιστρέφει true εάν ο αριστερός τελεστέος έχει μικρότερη ή ίση τιμή από τον αριστερό τελεστέο.

#### Αριθμητικοί τελεστές

Τελεστής	Περιγραφή	Παράδειγμα
% (Υπόλοιπο)	Δυναδικός τελεστής. Επιστρέφει το υπόλοιπο της διαίρεσης των δύο τελεστέων.	17 % 4 επιστρέφει 1.

++ (Increment)	Μοναδιαίος τελεστής. Προσθέτει μια μονάδα στον τελεστέο. Εάν τοποθετείται πριν τον τελεστέο (++x), επιστρέφει την τιμή του τελεστέου αφού την αυξήσει κατά ένα, ενώ εάν τοποθετηθεί μετά τον τελεστέο επιστρέφει την τιμή του τελεστέου πριν την αυξήσει κατά ένα.	Εάν το χ έχει τιμή ίση με το 2, τότε η δήλωση ++x θέτει το x ίσο με 3 και επιστρέφει 3, ενώ η δήλωση x++ θέτει το x ίσο με 3 και επιστρέφει 2.
-- (Decrement)	Μοναδιαίος τελεστής. Αφαιρεί μια μονάδα από τον τελεστέο. Εάν τοποθετείται πριν τον τελεστέο (--x), επιστρέφει την τιμή του τελεστέου αφού την μειώσει κατά ένα, ενώ εάν τοποθετηθεί μετά τον τελεστέο επιστρέφει την τιμή του τελεστέου πριν την μειώσει κατά ένα.	Εάν το χ έχει τιμή ίση με το 2, τότε η δήλωση --x θέτει το x ίσο με 1 και επιστρέφει 1, ενώ η δήλωση x-- θέτει το x ίσο με 1 και επιστρέφει 2
- (Αντίθετος)	Μοναδιαίος τελεστής. Επιστρέφει τον αντίθετο του τελεστέου	Εάν το χ είναι ίσο με 1 θα επιστρέψει -1.

**Λογικοί τελεστές**

Τελεστής	Περιγραφή	Παράδειγμα
Σύζευξη (&&)	Έκφραση1 && έκφραση2. Επιστρέφει την έκφραση1 εάν μετατρέπεται σε false αλλιώς επιστρέφει την έκφραση2.	"στυλό" && "μολύβι" επιστρέφει "μολύβι", false && "μολύβι" επιστρέφει false.
Διάζευξη (  )	Έκφραση1    έκφραση2. Επιστρέφει την έκφραση1 εάν μετατρέπεται σε true αλλιώς επιστρέφει την έκφραση2.	"στυλό"    "μολύβι" επιστρέφει "στυλό" false    "στυλό" επιστρέφει "στυλό" false    (1==0) επιστρέφει false.
Άρνηση not (!)	!(έκφραση1). Εάν η έκφραση είναι αληθής επιστρέφει false, εάν η έκφραση είναι ψευδής επιστρέφει true.	!(στυλό) επιστρέφει false. !false επιστρέφει true.

**Συντομεύσεις τελεστών**

Συντόμευση	Ισοδύναμη έκφραση
x += y	x = x + y
x -= y	x = x - y
x *= y	x = x * y
x /= y	x = x / y
x %= y	x = x % y
x <<= y	x = x << y
x >>= y	x = x >> y
x >>>= y	x = x >>> y
x &= y	x = x & y
x ^= y	x = x ^ y
x  = y	x = x   y

**Δηλώσεις**

Οι δηλώσεις καθορίζουν τη ροή εκτέλεσης εντολών. Στη JavaScript οι δηλώσεις διαχωρίζονται με ερωτηματικό (;). Η χρήση ερωτηματικού δεν είναι υποχρεωτική όταν οι δηλώσεις βρίσκονται σε διαφορετικές γραμμές, είναι όμως καλό να χρησιμοποιούνται.

### Διακλαδώσεις

#### if

Διατύπωση	Παράδειγμα
<pre>if (συνθήκη)     {δηλώσεις;} else     {δηλώσεις;}</pre>	<pre>if (i&lt;0)     {document.write("Αρνητικός");} else     {document.write("Θετικός");}</pre>

#### switch (Netscape και MSIE 4)

Διατύπωση	Παράδειγμα
<pre>switch (έκφραση) {     case ετικέτα1:         δήλωση;         break;     case ετικέτα2:         δήλωση;         break;     default: statement; }</pre>	<pre>switch (x) {     case 1:         document.write("1η επιλογή");         break;     case 2:         document.write("2η επιλογή");         break;     default: document.write("Λάθος επιλογή"); }</pre>

### Επανάληψεις

#### for

Διατύπωση	Παράδειγμα
<pre>for (αρχική τιμή; συνθήκη; βήμα)     {δηλώσεις}</pre>	<pre>For (i=1;i&lt;=n;i++)     {x=2*i;}</pre>

#### do..while (Netscape & MSIE 4)

Διατύπωση	Παράδειγμα
<pre>do {     δηλώσεις;} while (συνθήκη)</pre>	<pre>do {    document.write(i);     i++;} while (i&lt;10)</pre>

#### while

Διατύπωση	Παράδειγμα
<pre>while (συνθήκη)     {δηλώσεις;}</pre>	<pre>while (i&lt;10) {    document.write(i);     i++;}</pre>

### break και continue

Οι εντολές χρησιμοποιούνται για τη διακοπή εκτέλεσης μιας επανάληψης. Το break συνεχίζει με την πρώτη εντολή μετά την επανάληψη, ενώ το continue σταματά την εκτέλεση της τρέχουσας επανάληψης, συνεχίζει όμως την επανάληψη εάν η συνθήκη είναι αληθής.

### Διαχείριση αντικειμένων

#### for..in

```
for (μεταβλητή in αντικείμενο)
    {δηλώσεις;} //Διασχίζει τις ιδιότητες του αντικειμένου
```

**with**

with (αντικείμενο)

**Σχόλια**

//μια γραμμή σχολίων

/\*πολλές γραμμές

σχολίων\*/

**Συναρτήσεις**

```
function onomaSynartisis(parametros1, parametros2, ...)  
    {δηλώσεις;}
```

**Παράδειγμα συνάρτησης**

```
<SCRIPT language="javascript">  
function fact (n) {  
    if (n <= 0) return 1  
    return n * fact(n-1)  
}  
</SCRIPT>
```

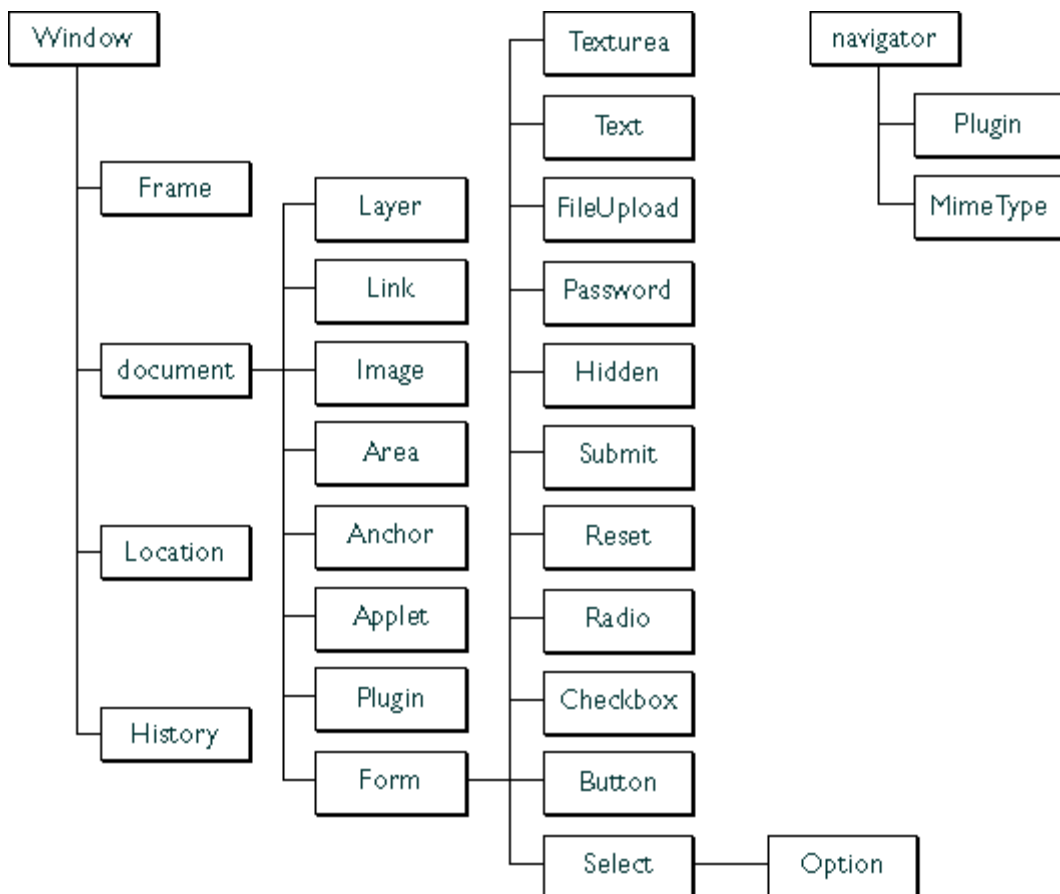
Και η κλήση της συνάρτησης:

```
alert (fact (10))
```

**Αντικείμενα**

Ένα αντικείμενο είναι μια συλλογή από δεδομένα μαζί με τις μεθόδους (συναρτήσεις) που επιδρούν στα δεδομένα του αντικείμενου. Όταν ανοίγεται ένα έγγραφο στο πρόγραμμα πλοήγησης, δημιουργείται ένας αριθμός αντικειμένων JavaScript με τιμές που αντιστοιχούν στην HTML του εγγράφου. Με αυτόν τον τρόπο μπορεί ο προγραμματιστής να γνωρίζει πόσα αντικείμενα βρίσκονται σε μια φόρμα ή πόσα πλαίσια υπάρχουν στο παράθυρο του προγράμματος πλοήγησης. Αυτά τα αντικείμενα ορίζονται με κάποια ιεραρχία. Η ιεραρχία αντικειμένων όπως καθορίζονται στο Netscape παρουσιάζεται στην επόμενη εικόνα:





Στη JavaScript χρησιμοποιείτε την τελεία για να διαχωρίσετε το αντικείμενο από τις ιδιότητές του και τις μεθόδους του. Για παράδειγμα μπορείτε να γράψετε:

```
window.status ή self.status
```

και να αναφερθείτε στη ράβδο κατάστασης (status) του παραθύρου. Μπορείτε να της δώσετε τιμή γράφοντας:

```
window.status = "Καλώς ήλθατε στη σελίδα μας"; ή
```

```
window.status = "Προηγούμενος έγραψε " + windows.defaultStatus;
```

### Δημιουργία αντικειμένων

Υπάρχουν δύο τρόποι για να δημιουργηθεί ένα αντικείμενο. Ο πρώτος είναι η άμεση δημιουργία ενός αντικειμένου ταυτόχρονα με τη δήλωσή του:

```
myCar={brand:"Toyota", model:"Corolla", color:"white"}
```

Η δήλωση αυτή θα εκτελεστεί σε Netscape 4.0 και νεώτερη έκδοση.

Στο δεύτερο τρόπο και πιο συνηθισμένο τρόπο, δηλώνουμε πρώτα ένα πρωτότυπο του αντικειμένου:

```
function car(brand, model, color)
{
  this.brand = brand;
  this.model = model;
  this.color = color;
  this.writeBrand = printBrand;
}

```

όπου έχουμε ήδη δηλώσει τη συνάρτηση printBrand:

```
function printBrand()
{
  document.write("<br>");
}

```

```
document.write(this.brand);
}
```

Η συνάρτηση writeBrand αποτελεί μια μέθοδο του αντικειμένου car. Στη συνέχεια οι δηλώσεις:

```
myCar = new car("Toyota", "Corolla", "white");
mySecondCar = new car("Toyota", "Starlet", "blue");
```

ορίζουν δύο στιγμιότυπα του αντικειμένου car. Μπορούμε να τυπώσουμε τον κατασκευαστή του αυτοκινήτου με την κλήση:

```
myCar.writeBrand();
```

### Συμβάντα

Τα συμβάντα είναι κατά κύριο λόγο αποτελέσματα ενεργειών του χρήστη. Για παράδειγμα το πάτημα ενός πλήκτρου του πληκτρολογίου ή η κίνηση του ποντικιού δημιουργούν συμβάντα. Ο προγραμματιστής μπορεί να χρησιμοποιήσει τα συμβάντα για να παρακολουθήσει τις ενέργειες των χρηστών και προσφέρει κατάλληλη ανάδραση. Για να το επιτύχει αυτό καλεί αντίστοιχους χειριστές συμβάντων όπως τον onChange ή onclick.

Χειριστής συμβάντος	Λόγος δημιουργίας συμβάντος
onAbort	Ο χρήστης διακόπτει τη μεταφορά/ φόρτωση μιας εικόνας.
onBlur	Ένα αντικείμενο μιας φόρμας ή ένα παράθυρο ή πλαίσιο χάνουν την εστίαση.
onChange	Ένα πεδίο μιας φόρμας χάνει την εστίαση και η τιμή του έχει μεταβληθεί.
onClick	Έχει γίνει κλικ σε ένα αντικείμενο.
ondblclick	Έχει γίνει διπλό-κλικ σε ένα αντικείμενο.
ondragdrop	Ο χρήστης αφήνει ένα αντικείμενο στο παράθυρο του προγράμματος πλοήγησης.
onerror	Η μεταφορά ενός εγγράφου ή εικόνας οδηγεί σε σφάλμα.
onfocus	Ένα παράθυρο ή πλαίσιο ή frameset ή αντικείμενο φόρμας λαμβάνει την εστίαση.
onkeydown	Ο χρήστης πιέζει και κρατά πιεσμένο ένα πλήκτρο.
onkeypress	Ο χρήστης πιέζει και απελευθερώνει ένα πλήκτρο.
onkeyup	Ο χρήστης απελευθερώνει ένα πλήκτρο.
onload	Το πρόγραμμα πλοήγησης ολοκληρώνει τη μεταφορά των δεδομένων ενός παραθύρου ή όλων των πλαισίων σε ένα διακριτικό FRAMESET.
onmousedown	Ο χρήστης πιέζει και κρατά πιεσμένο το πλήκτρο του ποντικιού.
onmousemove	Ο χρήστης μετακινεί το ποντίκι.
onmouseout	Ο δείκτης του ποντικιού εξέρχεται από μια περιοχή εικόνας ή συνδέσμου.
onmouseover	Ο δείκτης του ποντικιού περνά πάνω από ένα αντικείμενο.
onmouseup	Ο χρήστης απελευθερώνει το πλήκτρο του ποντικιού.
onmove	Ο χρήστης ή ένα πρόγραμμα JavaScript μετακινεί ένα παράθυρο ή πλαίσιο.
onreset	Ο χρήστης κάνει κλικ σε ένα πλήκτροReset.
onresize	Ο χρήστης μεταβάλλει το μέγεθος ενός παραθύρου ή

	πλαίσιου.
onSelect	Ο χρήστης επιλέγει κείμενο σε ένα πλαίσιο εισαγωγής κειμένου (μιας ή πολλαπλών γραμμών).
onSubmit	Ο χρήστης αποστέλλει τη φόρμα.
onUnload	Ο χρήστης φεύγει από ένα έγγραφο.

### Παράδειγμα

```

<HTML>
<HEAD>
<META http-equiv="Content-Type" content="text/html; charset=iso-8859-7">
<META http-equiv="Content-Language" content="el">
<TITLE>Δοκιμαστική σελίδα με JavaScript</TITLE>

<SCRIPT language="JavaScript">
//Όταν ο δείκτης περνά πάνω από το εικονίδιο αυτό αλλάζει
//Δημιουργούμε ένα αντικείμενο που περιέχει τα εικονίδια
function myButton(x, y, inactiveImg, activeImg){
this.inactive = new Image(x,y)
this.active = new Image(x,y)
this.inactive = inactiveImg
this.active = activeImg
}
</SCRIPT>
</HEAD>

<BODY>
<SCRIPT language="JavaScript">
First= new myButton(42,42,"inactive.gif", "active.gif")
</SCRIPT>
<A HREF="http://www2.mm.di.uoa.gr/jsript-example.html"
onMouseOver = "document.first_img.src=First.active"
onMouseOut = "document.first_img.src=First.inactive">
<IMG SRC="inactive.gif" name="first_img" width="42"
height="42">
</A>
</BODY>
</HTML>

```

Το παράδειγμα είναι διαθέσιμο στη διεύθυνση <http://www2.mm.di.uoa.gr/aay/example/jsript-example.html>.

### Επιπλέον πληροφορίες

Αυτή ήταν μια σύντομη περιγραφή της Javascript. Περισσότερα μπορείτε να δείτε στα αντίστοιχα εγχειρίδια του Netscape:

<http://developer.netscape.com/docs/manuals/communicator/jsguide4/contents.htm>

και της MS Internet Explorer:

<http://msdn.microsoft.com/scripting/default.htm?scripting/jsript/techinfo/jsdocs.htm>

Μερικές ακόμη διευθύνσεις:

<http://www.internet.com>

<http://www.di.uoa.gr/~rouvas>

<http://rummelplatz.uni-mannheim.de/~skoch/js/script.htm>

<b>Application management layer</b>	Επίπεδο διαχείρισης εφαρμογών
<b>Bitmap</b>	Ψηφιογραφικός
<b>Browser</b>	Πλοηγητής, φυλομετρητής
<b>Cathode Ray Tube</b>	Οθόνη καθοδικής λυχνίας
<b>Check box</b>	Πλαίσιο ελέγχου
<b>Cognitive engineering</b>	Γνωσιακή τεχνολογία
<b>Cognitive psychology</b>	Γνωσιακή (γνωστική) ψυχολογία
<b>Direct Manipulation</b>	Άμεση διαχείριση
<b>Graphical User Interface</b>	Γραφική διεπαφή
<b>Hardware</b>	Υλικό
<b>Human-Computer Interaction</b>	Αλληλεπίδραση/Διάδραση Ανθρώπου-Υπολογιστή
<b>Industrial engineering</b>	Βιομηχανική τεχνολογία
<b>Interaction</b>	Διάδραση, αλληλεπίδραση
<b>Interface</b>	Διεπαφή, διαπροσωπεία, διασύνδεση
<b>Interlaced</b>	Δαντελωτό, πεπλεγμένο
<b>Natural language</b>	Φυσική Γλώσσα
<b>Pointing device</b>	Δείκτης, συσκευή κατάδειξης
<b>Software</b>	Λογισμικό
<b>User interface</b>	Διεπαφή ή διασύνδεση χρήστη υπολογιστή
<b>User interface management systems</b>	Συστήματα διαχείρισης διεπαφών
<b>User interface toolkits</b>	Εργαλειοθήκες διεπαφών

**Κεφάλαιο 19      Παράρτημα Γ: Βιβλιογραφία**

- 
- [1] I. Sutherland, Sketchpad: A man-machine graphical communication system. Proceedings of the Spring Joint Computer Conference, 1963, Pages 329-46.
- [2] J. C. R. Licklider, Man-computer symbiosis, IRE Transactions of Human Factors in Electronics, 1960, Volume 1, 4-11.
- [3] A. Kay and A. Goldberg, Personal dynamic media, IEEE Computer, 1977, Volume 10, Number 3, Pages 31-44.
- [4] R. M. Baecker and W. A. S. Buxton, editors, Readings in Human-Computer Interaction: A Multi-disciplinary Approach, Los Altos, CA: Morgan Kaufmann.
- [5] Uren E., Robert H. and Perinotti T. Software Internationalization and localization. An introduction. Van Nostrand Reinhold, New York, 1993.
- [6] Nielsen, J., Finding Usability Problems through Heuristic Evaluation, Proceedings of CHI'92, pages 373-380, 1992.
- [7] Wharton, C., Bradford, J., Franzke, M., Applying Cognitive Walkthroughs to more Complex User Interfaces: Experiences, Issues, and Recommendations, Proceedings of CHI'92, pages 381-388, 1992.
- [8] Rowley E. D., Rhoades G. D., The Cognitive Jogthrough: A Fast-Paced User Interface Evaluation Procedure, Proceedings of CHI'92, pages 389-395, 1992.
- [9] Clare-Marie Karat, Robert Campbell, and Tara Fiegel, Comparison of Empirical Testing and Walkthrough methods in User Interface Evaluation, Proceedings of CHI'92, pages 397-404, 1992.
- [10] Nielsen, J., Philips, L. V., Estimating the Relative Usability of Two Interfaces: Heuristic, Formal, and Empirical Methods Compared, Proceedings of InterCHI'93, pages 214-221, 1993.
- [11] Jenny Preece, Yvonne Rogers, Helen Sharp, David Benyon, Simon Holland, Tom Carey, Human Computer Interaction, Chapter 30, Usage Data: Observations, Monitoring, Users' Opinions, pages 615-628, Addison Wesley, 1994.

---

[12] Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, Human Computer Interaction, pages 374-393, Prentice Hall International (UK), 1993.

[16] Stuard Card, Jock Mackinlay, Ben Shneiderman, “Readings in Information Visualization: Using Vision to Think”, Morgan Kaufmann, 1999.

[13] Egidio, C., Patterson, J., Pictures and category labels as navigational aids for catalog browsing, CHI'88 Conference Proceedings, 1988, pp. 127-132.

[14] LYNCH, P. J., (1995). Yale University C/AIM WWW Style Guide. [<http://info.med.yale.edu/caim/StyleManual>]

[15] Shneiderman B., Designing the User Interface, Third Edition, Addison Wesley Longman, Inc., 1998