



Μηχανική Λογισμικού

Μάθημα 3^ο

Ανάλυση απαιτήσεων

Intellectual Property Rights

- Οι παρούσες διαφάνειες περιέχουν διαμορφωμένο ή και αυτούσιο υλικό από τα ακόλουθα συγγράμματα:
 - Διαφάνειες για το μάθημα «Τεχνολογία Λογισμικού» της ΣΗΜΜΥ του ΕΜΠ των Νίκου Παπασπύρου, Γιάννη Μαϊστρου, Βασίλη Βεσκούκη
 - Διαφάνειες για το μάθημα «Τεχνολογία Λογισμικού» του τμήματος Ψηφιακών Συστημάτων του Παν. Πειραιώς της Ανδριάνας Πρέντζα
 - Διαφάνειες για το βιβλίο «Software Engineering» του Ian Sommerville

Περιεχόμενα

- Ανάλυση απαιτήσεων



Ανάλυση Απαιτήσεων

Πως περιγράφουμε το τι περιμένουμε από ένα σύστημα λογισμικού

Περιεχόμενα

- Οι έννοιες των απαιτήσεων χρήστη και των απαιτήσεων συστήματος
- Οι έννοιες των λειτουργικών και μη λειτουργικών απαιτήσεων
- Τεχνικές και κείμενα για την περιγραφή απαιτήσεων

Ζαχαρίας ΔενΤοΦτιάχνω

- **ΕΠΕΙΣΟΔΙΟ 10:** Ο πελάτης έχει πάντα δίκιο
- **INTERLUDE:** Οδηγός επιβίωσης κατά την εκμείευση προδιαγραφών
- Οι προδιαγραφές άργησαν μια μέρα
- **INTERLUDE:** Μικρός οδηγός για τις συναντήσεις developers με πελάτες

Μηχανική απαιτήσεων (requirements engineering)

- Η διαδικασία καθορισμού
 - των υπηρεσιών που ο χρήστης επιθυμεί να του παρέχει το σύστημα
 - Των περιορισμών υπό τους οποίους γίνεται η ανάπτυξη και η λειτουργία του

Τι προσπαθούμε να πετύχουμε;

- Να κατανοήσουμε τα προβλήματα που θα κληθεί να επιλύσει το σύστημα που φτιάχνουμε
- Να κάνουμε ερωτήσεις και να δώσουμε αντίστοιχες απαντήσεις τόσο για το πρόβλημα όσο και για το σύστημα
- Να αποφασίσουμε τι πρέπει να κάνει το σύστημα
- Να αποφασίσουμε τι ΔΕΝ πρέπει να κάνει το σύστημα
- Να εξασφαλίσουμε ότι το σύστημα θα ικανοποιήσει τις απαιτήσεις των διαφόρων εμπλεκομένων με αυτό
- Να δημιουργήσουμε μια αρχική κατάσταση για την σχεδίαση και ανάπτυξη του συστήματος
- Να μπορέσουμε να οδηγηθούμε σε σενάρια ελέγχου του συστήματος

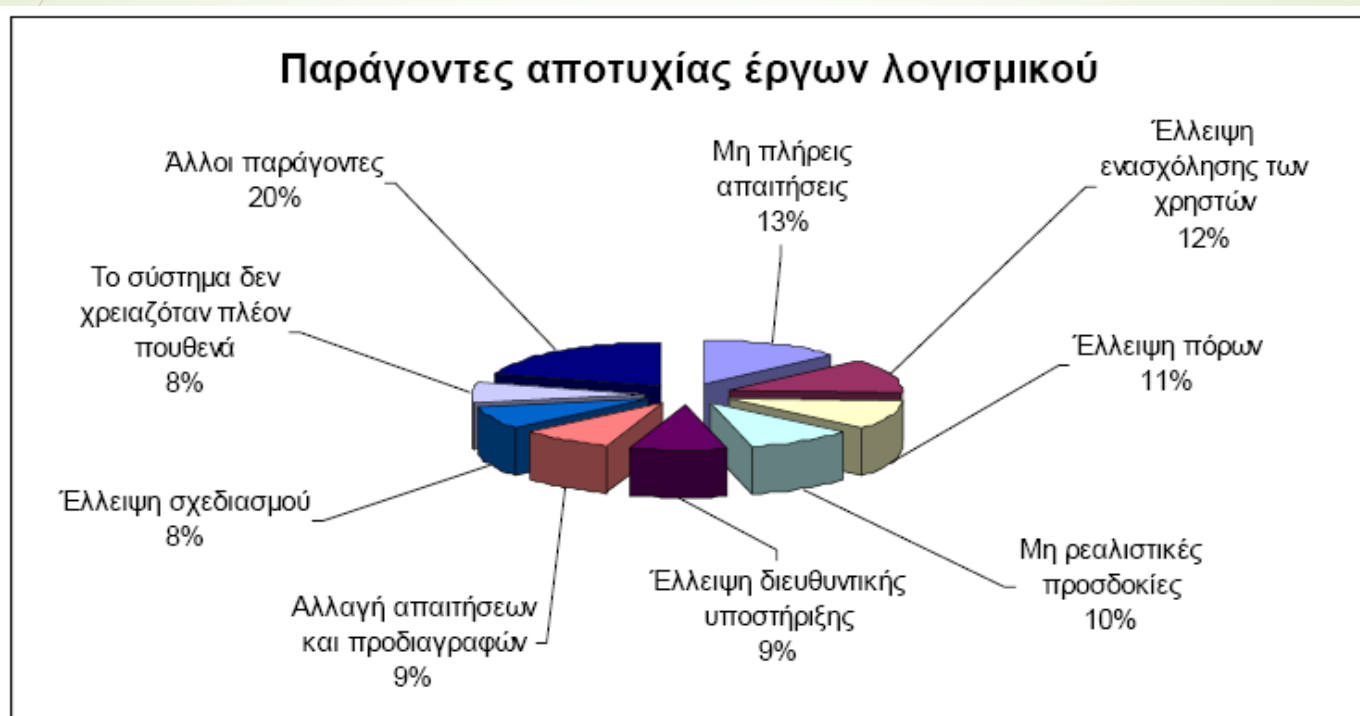
Τι ορίζεται ως απαίτηση;

- Μια απαίτηση μπορεί να κυμαίνεται
 - Από μια γενική δήλωση σχετικά με μια υπηρεσία ή έναν περιορισμό του συστήματος
 - Έως μια πολύ λεπτομερή προδιαγραφή (π.χ. με χρήση μαθηματικών ή άλλων μέσων)

Τι ορίζεται ως απαίτηση;

- Αυτό είναι αναπόφευκτο αφού οι απαιτήσεις έχουν διπλό ρόλο
 - Μπορεί να περιλαμβάνεται σε μια προκήρυξη για κάποιο έργο
 - Οπότε πρέπει να αφήνει βαθμούς ελευθερίας
 - Μπορεί να περιλαμβάνεται στη σύμβαση ενός έργου
 - Οπότε πρέπει να περιέχει ικανό βαθμό λεπτομέρειας

Σημασία απαιτήσεων



Έρευνα της Standish Group σε 350 εταιρίες και 8000 έργα (1995)

Είδη απαιτήσεων - Απαιτήσεις χρήστη

- Κείμενο σε φυσική γλώσσα (πιθανώς μαζί με σχεδιαγράμματα)
- Περιγράφει τις υπηρεσίες που παρέχονται από ένα σύστημα προς τον τελικό χρήστη
- Συνήθως συγγράφεται από τον αναθέτοντα και απευθύνεται στον προσφέροντα

Είδη απαιτήσεων - Απαιτήσεις συστήματος

- ▶ Ένα δομημένο κείμενο με λεπτομερείς περιγραφές των υπηρεσιών ενός συστήματος.
- ▶ Αποτελεί μέρος του συμβολαίου μεταξύ πελάτη και Αναδόχου

Είδη απαιτήσεων - Απαιτήσεις (Προδιαγραφές) λογισμικού

- Λειτουργίες ενός συστήματος οι οποίες εκτελούνται από το αντίστοιχο λογισμικό
- Λεπτομερής περιγραφή του λογισμικού η οποία χρησιμεύει σαν βάση για την ανάπτυξή του.
- Απευθύνεται στους προγραμματιστές

Παράδειγμα

- Ζητείται η κατασκευή ενός συστήματος παρακολούθησης βιοσημάτων κατ'οίκον σε ασθενείς με χρόνιες παθήσεις, το οποίο με χρήση αισθητήρων συλλέγει ΗΚΓ, κορεσμό οξυγόνου, θερμοκρασία και αρτηριακή πίεση (συστολική και διαστολική). Το σύστημα αποθηκεύει τα στοιχεία αυτά και εξάγει αποτελέσματα, όπως μέση τιμή αρτηριακής πίεσης (συστολικής και διαστολικής) ανά ημέρα κτλ. Το σύστημα αποτελείται από ηλεκτρόδια ΗΚΓ, αισθητήρες θερμοκρασίας και πίεσης, παλμικό οξύμετρο, ηλεκτρονικούς υπολογιστές και ανθρώπους.

Παράδειγμα

- Η περιγραφή των απαιτήσεων από το σύστημα συλλογής και επεξεργασίας βιοσημάτων περιγράφεται ως εξής:
 - Το σύστημα εκτελεί την εργασία συλλογής δεδομένων βιοσημάτων με τη βοήθεια αισθητήρων, την ηλεκτρονική μετάδοση των δεδομένων αυτών στο κέντρο υπηρεσιών, την αποθήκευση αυτών σε ηλεκτρονική μορφή, την επεξεργασία με την εξαγωγή χαρακτηριστικών μεγεθών καθώς και την αξιολόγηση των μετρήσεων και την αποστολή προειδοποιήσεων σε περίπτωση μεγεθών εκτός ορίων.
- Ποιες είναι οι απαιτήσεις από το σύστημα και ποιες οι απαιτήσεις από το λογισμικό;

Παράδειγμα

- Απαιτήσεις από το σύστημα
 - Λήψη και αποστολή μέτρησης ΗΚΓ
 - Λήψη και αποστολή μέτρησης κορεσμού οξυγόνου
 - Λήψη και αποστολή μέτρησης θερμοκρασίας
 - Λήψη και αποστολή μέτρησης αρτηριακής πίεσης (συστολική και διαστολική)

Παράδειγμα

- Απαιτήσεις από το λογισμικό
 - Αποθήκευση μετρήσεων ΗΚΓ
 - Αποθήκευση μετρήσεων κορεσμού οξυγόνου
 - Αποθήκευση μετρήσεων θερμοκρασίας
 - Αποθήκευση μετρήσεων αρτηριακής πίεσης (συστολική και διαστολική)
 - Επεξεργασία μετρήσεων (ΗΚΓ, Κορεσμού οξυγόνου, θερμοκρασίας, αρτηριακής πίεσης)
 - Υπολογισμός μέσης τιμής αρτηριακής πίεσης (συστολικής και διαστολικής) ανά ημέρα

Παράδειγμα

- Συνιστώσες συστήματος

- **Μηχανές**

- Λήψη και αποστολή μετρήσεων

- **Λογισμικό**

- Αποθήκευση και επεξεργασία μετρήσεων

- **Άνθρωποι**

- Επίβλεψη λειτουργίας και αξιολόγηση μετρήσεων

Είδη απαιτήσεων

- Πολλές φορές τα όρια είναι δυσδιάκριτα
 - Δεν είναι δλδ εύκολο να οριστεί τι αφορά στον χρήστη, τι στο σύστημα και τι στο λογισμικό
- Σε μια τέτοια περίπτωση εξετάζουμε τις απαιτήσεις συνολικά είτε ως απαιτήσεις συστήματος είτε ως απαιτήσεις από το λογισμικό
 - Παραδίδεται ένα έγγραφο για όλα

Definitions and specifications

Requirements definition

1. The software must provide a means of representing and accessing external files created by other tools.

Requirements specification

- 1.1 The user should be provided with facilities to define the type of external files.
- 1.2 Each external file type may have an associated tool which may be applied to the file.
- 1.3 Each external file type may be represented as a specific icon on the user's display.
- 1.4 Facilities should be provided for the icon representing an external file type to be defined by the user.
- 1.5 When a user selects an icon representing an external file, the effect of that selection is to apply the tool associated with the type of the external file to the file represented by the selected icon.



Λειτουργικές απαιτήσεις

Λειτουργικές απαιτήσεις - Ορισμός

- Φράσεις που περιγράφουν
 - Ποιες υπηρεσίες παρέχονται από ένα σύστημα
 - Πως ένα σύστημα αντιδρά σε συγκεκριμένες εισόδους
 - Πως ένα σύστημα συμπεριφέρεται σε συγκεκριμένες συνθήκες

Λειτουργικές απαιτήσεις

- Εξαρτώνται από
 - Το είδος του λογισμικού
 - Το είδος των χρηστών
 - Τον τύπο του συστήματος στο οποίο θα εγκατασταθεί το σύστημα
- Οι λειτουργικές απαιτήσεις χρήστη μπορούν να είναι γενικό κείμενο
- Οι λειτουργικές απαιτήσεις συστήματος θα πρέπει να είναι λεπτομερείς

Παραδείγματα λειτουργικών απαιτήσεων

- Ο χρήστης θα πρέπει να μπορεί να ψάξει σε όλες τις διαθέσιμες Β.Δ. ή να μπορεί να επιλέξει ένα υποσύνολό τους
- Το σύστημα θα διαθέτει κατάλληλους viewers για να μπορεί ο χρήστης να διαβάζει κείμενα που βρίσκονται στην αποθήκη κειμένων
- Κάθε εισερχόμενη παραγγελία θα παίρνει ένα μοναδικό αναγνωριστικό (ORDER_ID) με το οποίο θα καταχωρείται στην περιοχή αποθήκευσης που διατίθεται για τον συγκεκριμένο πελάτη

Πληρότητα και συνέπεια

- Οι απαιτήσεις πρέπει να χαρακτηρίζονται και από πληρότητα και από συνέπεια
- Πληρότητα
 - Θα πρέπει να περιέχουν περιγραφές όλων των λειτουργιών που απαιτούνται
- Συνέπεια
 - Η μία απαίτηση δεν θα πρέπει να συγκρούεται με άλλη
- Πολύ δύσκολο στην πράξη

Πληρότητα και συνέπεια

- Ασαφείς απαιτήσεις είναι «ανοικτές» σε παρερμηνεία ανάλογα με το ποιος τις διαβάσει
- Π.χ. τι σημαίνει ‘κατάλληλοι viewers’
 - Χρήστης – Εξειδικευμένος viewer για κάθε τύπο αρχείου
 - Προγραμματιστής – Viewer που εμφανίζει τα περιεχόμενα του αρχείου ως απλό κείμενο



Μη λειτουργικές απαιτήσεις

Ορισμός

- Αφορούν σε περιορισμούς των λειτουργιών ενός συστήματος όπως π.χ.
 - Χρονισμός
 - Συμμόρφωση με πρότυπα
 - Χρήση συγκεκριμένων εργαλείων ανάπτυξης κλπ
 - Συμμόρφωση με συγκεκριμένη νομοθεσία

Κατηγορίες ΜΛ απαιτήσεων

- Απαιτήσεις χρήσης
 - Καθορίζουν χαρακτηριστικά χρήσης συστήματος, user interface και υλικό τεκμηρίωσης που θα έχει στη διάθεσή του ο τελικός χρήστης
- Απαιτήσεις αξιοπιστίας
 - Καθορίζουν τη συμπεριφορά του λογισμικού σε καταστάσεις ενδογενών ή εξωγενών σφαλμάτων, τη διαδικασία αποκατάστασης, την πρόβλεψη τέτοιων καταστάσεων, και την επιθυμητή διαθεσιμότητα του λογισμικού
- Απαιτήσεις επιδόσεων
 - Εισάγουν περιορισμούς σε λειτουργικές απαιτήσεις σχετικά με το χρόνο εκτέλεσής τους και με τη χρήση πόρων όπως η μνήμη και οι μονάδες επεξεργασίας

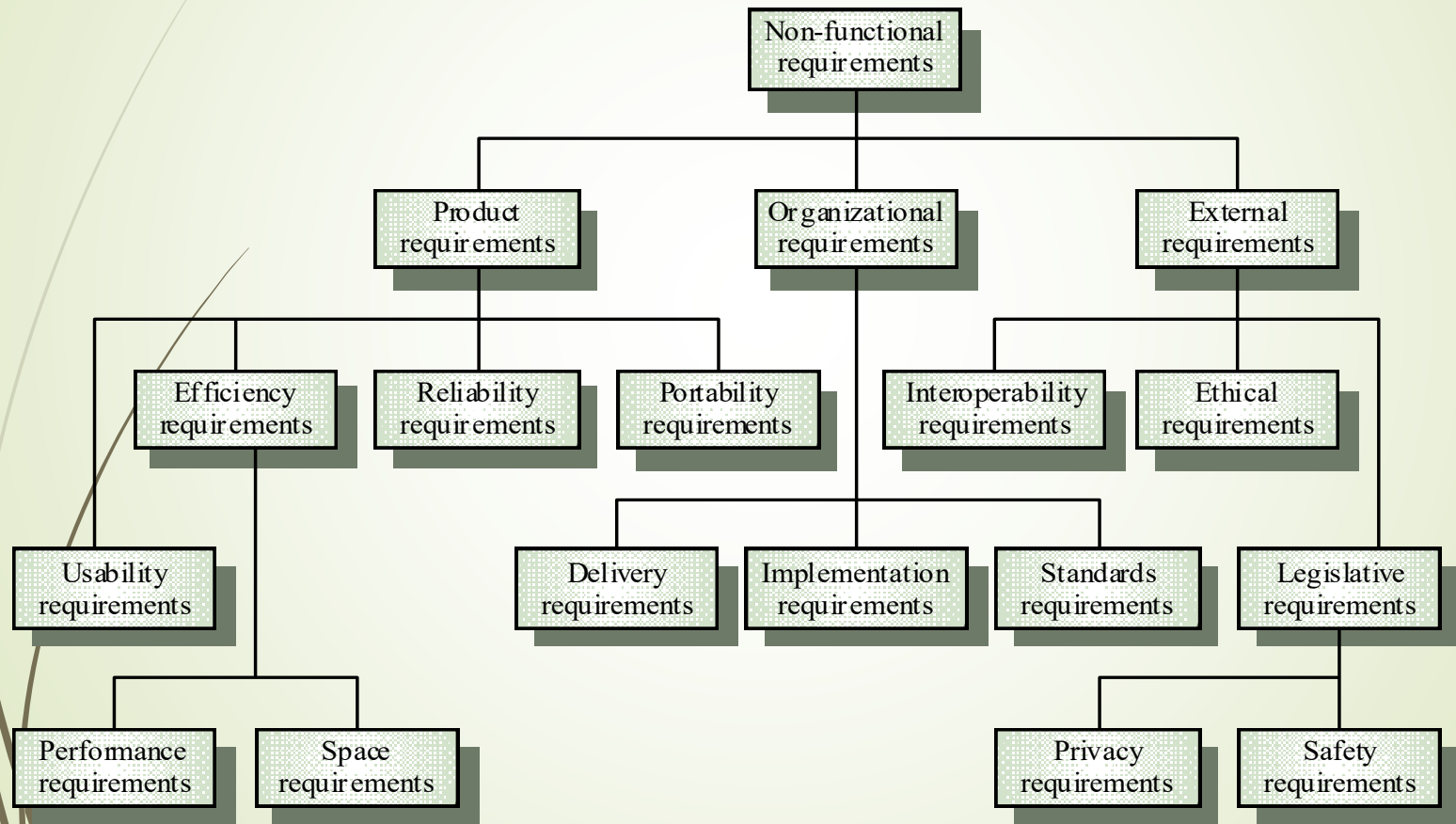
Κατηγορίες ΜΛ απαιτήσεων

- Απαιτήσεις υποστήριξης
 - Καθορίζουν τα επιθυμητά χαρακτηριστικά για τον έλεγχο και τη συντήρηση του λογισμικού
- Απαιτήσεις σχεδίασης
 - Καθορίζουν τον τρόπο με τον οποίο θα πρέπει να γίνει η σχεδίαση λογισμικού, π.χ χρήση προτύπων, συγκεκριμένων μεθοδολογιών κτλ.
- Απαιτήσεις υλοποίησης
 - Καθορίζουν τον τρόπο με τον οποίο θα πρέπει να γίνει η συγγραφή του πηγαίου κώδικα του λογισμικού

Κατηγορίες ΜΛ απαιτήσεων

- Απαιτήσεις επικοινωνίας με άλλα συστήματα
 - Καθορίζουν τα εξωτερικά συστήματα, λογισμικού ή άλλα, με τα οποία το λογισμικό θα επικοινωνεί, καθώς και τον τρόπο (π.χ. πρότυπα, φυσική σύνδεση) πραγματοποίησης της επικοινωνίας αυτής
- Απαιτήσεις Βάσεων Δεδομένων
 - Καθορίζουν τις οντότητες για τη διαχείριση των οποίων είναι υπεύθυνο το σύστημα λογισμικού, καθώς και τα ιδιώματα καθεμίας από αυτές
- Φυσικές απαιτήσεις
 - Καθορίζουν τα επιθυμητά φυσικά χαρακτηριστικά του λογισμικού και του συστήματος (λειτουργικό σύστημα, προδιαγραφές υπολογιστή, απαιτούμενες δικτυακές συνδέσεις)

Κατηγορίες ΜΛ απαιτήσεων



Διαπίστωση μη λειτουργικών απαιτήσεων

- Τα διάφορα συστήματα λογισμικού πρέπει να πετυχαίνουν κάποιους στόχους
 - Οι στόχοι είναι συνήθως διατυπωμένοι μέσω γενικών εκφράσεων
 - Π.χ. το σύστημα πρέπει να είναι «εύκολο στη χρήση»
- Οι μη λειτουργικές απαιτήσεις βοηθούν στην επίτευξη τέτοιων στόχων
 - Πρέπει όμως να μπορεί να διαπιστωθεί η λειτουργία τους
 - Πρέπει λοιπόν να συνδεθούν με ένα ή περισσότερα μετρήσιμα μεγέθη

Παραδείγματα

➤ Στόχος συστήματος

- Το σύστημα θα πρέπει να είναι εύκολο στη χρήση του στο επίπεδο των εμπείρων χρηστών. Επιπλέον θα πρέπει να είναι οργανωμένο με τέτοιο τρόπο ώστε τα λάθη να ελαχιστοποιούνται

Παραδείγματα

➤ Η αντίστοιχη μετρήσιμη μη λειτουργική απαίτηση

- Οι έμπειροι χρήστες θα πρέπει να μπορούν να χρησιμοποιούν όλες τις λειτουργίες του συστήματος μετά από δίωρη εκπαίδευση.
- Μετά την εκπαίδευση ο μέσος αριθμός λαθών από έμπειρους χρήστες δεν θα πρέπει να ξεπερνά τα δύο / ημέρα.

Requirements measures

Property	Measure
Speed	Processed transactions/second User/Event response time Screen refresh time
Size	K Bytes Number of RAM chips
Ease of use	Training time Number of help frames
Reliability	Mean time to failure Probability of unavailability Rate of failure occurrence Availability
Robustness	Time to restart after failure Percentage of events causing failure Probability of data corruption on failure
Portability	Percentage of target dependent statements Number of target systems

Παράδειγμα

38



- Να ορίσετε λειτουργικές και μη λειτουργικές απαιτήσεις μιας εφαρμογής λογισμικού που να εκτελεί τις 4 πράξεις της αριθμητικής

Παράδειγμα

- Λειτουργικές απαιτήσεις («τι να κάνει το λογισμικό»)
 - 1. Εκτέλεση της πράξης «πρόσθεση»
 - 2. Εκτέλεση της πράξης «αφαίρεση»
 - 3. Εκτέλεση της πράξης «πολλαπλασιασμός»
 - 4. Εκτέλεση της πράξης «διαίρεση»
 - 5. Εκτέλεση υπολογισμού επί τοις εκατό
 - 6. Εκτέλεση υπολογισμού τετραγωνικής ρίζας
 - 7. Μηδενισμός καταχωρητή

Παράδειγμα

- Μη λειτουργικές απαιτήσεις («ποια χαρακτηριστικά να έχει»)
 - 1. Η μορφή του να θυμίζει τη γνώριμη μορφή υπολογιστή τσέπης
 - 2. Να χειρίζεται πραγματικούς αριθμούς 10 ψηφίων
 - 3. Να δέχεται εντολές από ποντίκι και πληκτρολόγιο

Παράδειγμα – Ταξινόμηση απαιτήσεων

- Η εφαρμογή θα πρέπει να τρέχει σε περιβάλλον Linux
 - **Μη λειτουργική απαίτηση – Περιβάλλον Λειτουργίας**
- Η εφαρμογή θα πρέπει να υπολογίζει τη μέση τιμή και τυπική απόκλιση της παραμέτρου A
 - **Λειτουργική απαίτηση**
- Η εφαρμογή θα πρέπει να μπορεί να χρησιμοποιείται ταυτόχρονα από περισσότερους από έναν χρήστες
 - **Μη λειτουργική απαίτηση - Χρήσης**
- Η σχεδίαση θα πρέπει να γίνεται με τη χρήση του προτύπου UML
 - **Μη λειτουργική απαίτηση - Σχεδίασης**
- Η εφαρμογή θα πρέπει να διαχειρίζεται αρχείο φοιτητών και βαθμολογίας στα μαθήματα που παρακολουθούν
 - **Λειτουργική απαίτηση**

Παράδειγμα – Ταξινόμηση απαιτήσεων

- Η καταχώρηση της βαθμολογίας θα πρέπει να γίνεται μόνο από εξουσιοδοτημένους χρήστες
 - **Μη λειτουργική απαίτηση - Χρήσης**
- Η εφαρμογή θα υλοποιηθεί σε γλώσσα C++
 - **Μη λειτουργική απαίτηση - Υλοποίησης**
- Η επικοινωνία της εφαρμογής με τη βάση δεδομένων θα γίνεται μέσω δικτύου TCP/IP
 - **Μη λειτουργική απαίτηση - Επικοινωνίας**
- Σε περίπτωση μη διαθεσιμότητας του δικτύου, η προσπάθεια σύνδεσης θα επαναλαμβάνεται ανά 5 λεπτά και επί 1 ώρα
 - **Μη λειτουργική απαίτηση - Αξιοπιστίας**
- Η εφαρμογή θα εκτυπώνει λίστα με τους εγγεγραμμένους φοιτητές ανά μάθημα
 - **Λειτουργική απαίτηση**



Συλλογή και ανάλυση απαιτήσεων

Επιμέρους βήματα κατά τη φάση ανάλυσης απαιτήσεων

- Κατανόηση προβλήματος
- Συλλογή απαιτήσεων
- Επίλυση συγκρούσεων
- Ιεράρχηση
- Επαλήθευση απαιτήσεων

Επιμέρους βήματα κατά τη φάση ανάλυσης απαιτήσεων

- Κατανόηση προβλήματος
 - Εξοικείωση του κατασκευαστή με την ουσία του προβλήματος
 - Δημιουργική αλλά δύσκολη εργασία!

Επιμέρους βήματα κατά τη φάση ανάλυσης απαιτήσεων

- Συλλογή απαιτήσεων
 - Αρχική καταγραφή απαιτήσεων σε λίστα μέσω
 - Συνεντεύξεων
 - Ερωτηματολογίων
 - Συζητήσεων με ειδικούς
 - Ταξινόμηση απαιτήσεων σε ομάδες ανάλογα με το υποσύνολο προβλήματος που αφορούν ή άλλο τρόπο.
 - Εντοπισμός ασυνεπειών – δύο ή περισσότερες απαιτήσεις η ικανοποίηση των οποίων δεν μπορεί να γίνει ταυτόχρονα. ΑΡΑ
- Επίλυση συγκρούσεων

Επιμέρους βήματα κατά τη φάση ανάλυσης απαιτήσεων

- Ιεράρχηση
 - Ως προς τη σειρά ικανοποίησής τους
 - Καθορίζει τη χρονική αλληλουχία ενσωμάτωσης στο λογισμικό λειτουργιών που ικανοποιούν τις απαιτήσεις
 - Καθορίζει ποιες απαιτήσεις δε θα ικανοποιηθούν καθόλου, αν επιβληθεί από εξωτερικούς παράγοντες (π.χ. κόστος)

Επιμέρους βήματα κατά τη φάση ανάλυσης απαιτήσεων

- Επαλήθευση απαιτήσεων
 - Νέα επαφή με τον πελάτη με τη μορφή σύσκεψης ή ανταλλαγής εγγράφων
 - Αν οι απαιτήσεις ικανοποιούν, τότε έναρξη κατασκευής μοντέλων περιγραφής λογισμικού
 - Αν οι απαιτήσεις δεν ικανοποιούν, τότε πισωγυρίσματα



Έγγραφα απαιτήσεων

Software Requirements Specification (SRS)

Έγγραφα απαιτήσεων

- Περιγράφουν λειτουργικές και μη λειτουργικές απαιτήσεις με τρόπο τέτοιο έτσι ώστε να είναι κατανοητές από τον μέσο χρήστη χωρίς να χρειάζονται ειδικές τεχνικές γνώσεις
- Ορίζονται με χρήση
 - Φυσικής γλώσσας
 - Πινάκων
 - Διαγραμμάτων

Editor grid requirement

2.6 Grid facilities To assist in the positioning of entities on a diagram, the user may turn on a grid in either centimetres or inches, via an option on the control panel. Initially, the grid is off. The grid may be turned on and off at any time during an editing session and can be toggled between inches and centimetres at any time. A grid option will be provided on the reduce-to-fit view but the number of grid lines shown will be reduced to avoid filling the smaller diagram with grid lines.

Φυσική Γλώσσα - Προβλήματα

- Έλλειψη ακρίβειας
 - Ένα εξαιρετικά ακριβές κείμενο συνήθως είναι δύσκολο να κατανοηθεί
- Σύγχυση απαιτήσεων
 - Μέσα στο ίδιο κείμενο συνυπάρχουν λειτουργικές και μη λειτουργικές απαιτήσεις
- Ενοποίηση απαιτήσεων
 - Πολλές και διαφορετικές απαιτήσεις «ενώνονται» σε μία έκφραση ή πρόταση

Δομημένη παρουσίαση

2.6 Grid facilities

2.6.1 The editor shall provide a grid facility where a matrix of horizontal and vertical lines provide a background to the editor window. This grid shall be a passive grid where the alignment of entities is the user's responsibility.

Rationale: A grid helps the user to create a tidy diagram with well-spaced entities. Although an active grid, where entities 'snap-to' grid lines can be useful, the positioning is imprecise. The user is the best person to decide where entities should be positioned.

Specification: ECL IPSE/WS/Tools/DE/FS Section 5.6

Βηματική περιγραφή απαιτήσεων

3.5.1 Adding nodes to a design

3.5.1.1 **The editor shall provide a facility for users to add nodes of a specified type to their design.**

3.5.1.2 The sequence of actions to add a node should be as follows:

1. The user should select the type of node to be added.
2. The user should move the cursor to the approximate node position in the diagram and indicate that the node symbol should be added at that point.
3. The user should then drag the node symbol to its final position.

Rationale: The user is the best person to decide where to position a node on the diagram. This approach gives the user direct control over node type selection and positioning.

Specification: ECL IPSE/WS/Tools/DE/FS. Section 3.5.1

Κανόνες για την συγγραφή απαιτήσεων

- Χρήση κοινής δομής κειμένου για όλες τις απαιτήσεις
- Χρήση της γλώσσας με συνεπή τρόπο π.χ.
 - «πρέπει» για απαραίστες απαιτήσεις
 - «είναι επιθυμητό» για προαιρετικές
- Η πολύ τεχνική ορολογία θα πρέπει να αποφεύγεται
- Μια καλή ιδέα είναι η χρήση πρότυπων φορμών

Δομή πρότυπης φόρμας

ECLIPSE/Workstation/Tools/DE/FS/3.5.1

Function Add node

Description Adds a node to an existing design. The user selects the type of node, and its position. When added to the design, the node becomes the current selection. The user chooses the node position by moving the cursor to the area where the node is added.

Inputs Node type, Node position, Design identifier.

Source Node type and Node position are input by the user, Design identifier from the database.

Outputs Design identifier.

Destination The design database. The design is committed to the database on completion of the operation.

Requires Design graph rooted at input design identifier.

Pre-condition The design is open and displayed on the user's screen.

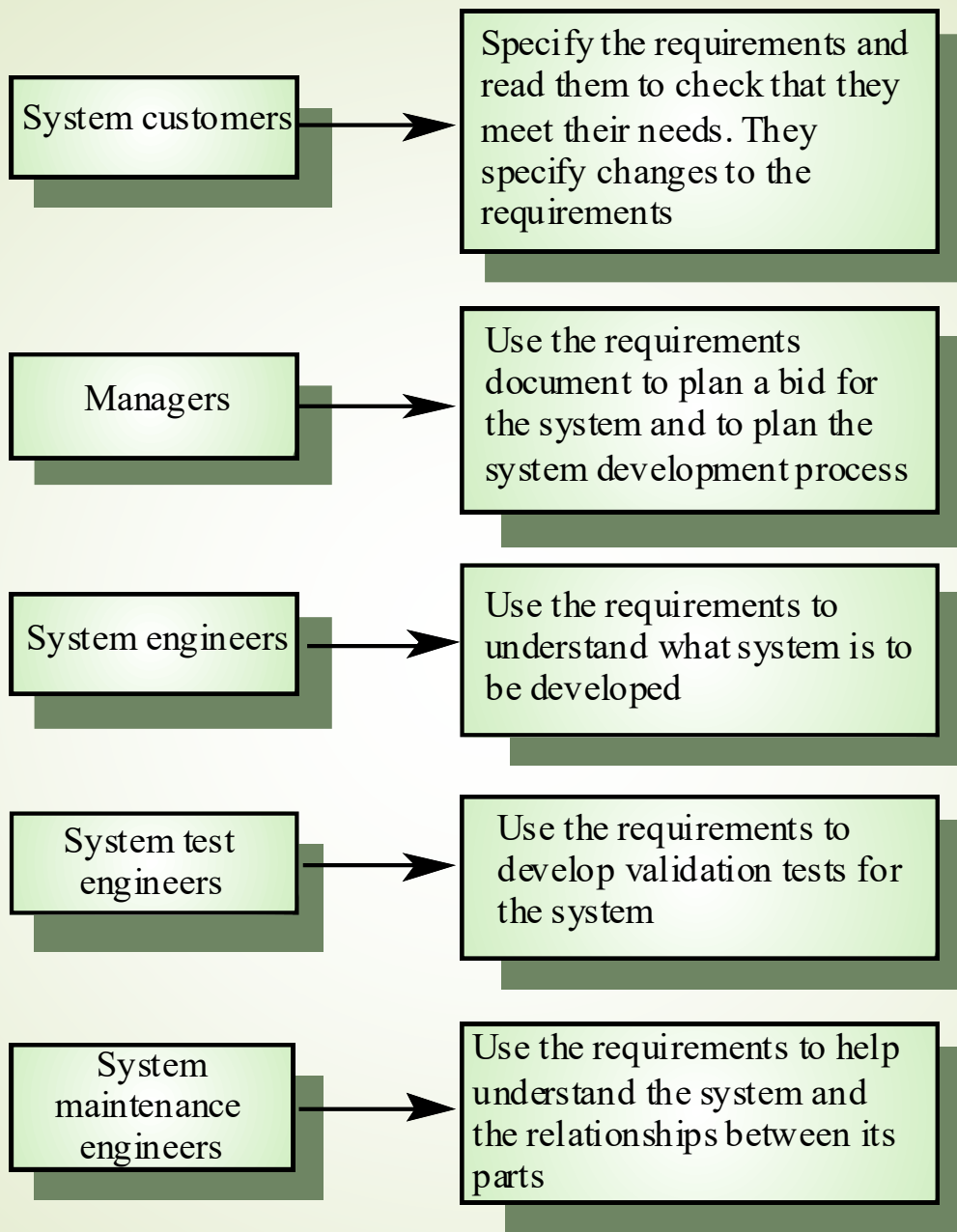
Post-condition The design is unchanged apart from the addition of a node of the specified type at the given position.

Side-effects None

Definition: ECLIPSE/Workstation/Tools/DE/RD/3.5.1

Το έγγραφο απαιτήσεων

- Το έγγραφο απαιτήσεων αποτελεί μια επίσημη δήλωση του τι χρειάζεται να κάνουν οι αυτοί που θα αναπτύξουν το σύστημα
- Περιέχει τόσο ορισμούς όσο και προδιαγραφές των απαιτήσεων
- Δεν είναι κείμενο σχεδίασης.
 - Όσο είναι δυνατό περιγράφει Τι πρέπει να κάνει το σύστημα και όχι ΠΩΣ θα το κάνει



Οι χρήστες του
εγγράφου
απαιτήσεων

Απαιτήσεις από το έγγραφο απαιτήσεων

- Πρέπει να περιγράφει την εξωτερική συμπεριφορά του συστήματος
- Πρέπει να καθορίζει τους περιορισμούς υπό τους οποίους θα πραγματοποιηθεί η ανάπτυξη
- Πρέπει να αλλάζει εύκολα
- Πρέπει να αποτελεί εργαλείο αναφοράς κατά την συντήρηση του συστήματος

Το πρότυπο της ΙΕΕΕ

- Προτείνει μια γενική δομή η οποία πρέπει να προσαρμοστεί στις ανάγκες του εκάστοτε συστήματος
 - Εισαγωγή
 - Γενική περιγραφή
 - Συγκεκριμένες απαιτήσεις
 - Παραρτήματα
 - Ευρετήριο

Το πρότυπο της ΙΕΕΕ - Εισαγωγή

- Εισαγωγή
 - Σκοπός
 - Πλαίσιο έργου
 - Λεξικό όρων
 - Αναφορές

Το πρότυπο της ΙΕΕΕ – Εισαγωγή – Πλαίσιο έργου

- Επωνυμία συστήματος
- Επεξήγηση του τι κάνει / δεν κάνει το σύστημα
- Αναμενόμενα οφέλη

Το πρότυπο της ΙΕΕΕ – Γενική περιγραφή

- Γενική περιγραφή
 - Σχέση με άλλα προϊόντα / συστήματα
 - Λειτουργίες συστήματος
 - Χαρακτηριστικά χρηστών
 - Περιορισμοί
 - Υποθέσεις και εξαρτήσεις

Το πρότυπο της ΙΕΕΕ – Γενική περιγραφή - Σχέση με άλλα προϊόντα / συστήματα

- Διεπαφές συστήματος
- Διεπαφές χρήστη
- Διεπαφές υλικού
- Διεπαφές λογισμικού
- Διεπαφές επικοινωνιών
- Μνήμη
- Απαιτήσεις από το περιβάλλον λειτουργίας

Το πρότυπο της ΙΕΕΕ – Γενική περιγραφή – περιορισμοί

- Νομικά / ρυθμιστικά θέματα
- Περιορισμοί υλικού
- Θέματα αξιοπιστίας
- Θέματα ασφάλειας κλπ

Traceability of requirements

- Κάθε απαίτηση καλό είναι να μπορεί να αντιστοιχηθεί στην πηγή της π.χ.
 - Περίπτωση χρήσης
 - Νόμος ή κανονισμός
 - Διεθνές πρότυπο
- Έτσι εξασφαλίζεται ότι η απαίτηση πράγματι χρειάζεται από το έργο

Συμπεράσματα

- Οι απαιτήσεις καθορίζουν τι πρέπει να κάνει το σύστημα και υπό ποιους περιορισμούς γίνεται η ανάπτυξη και η λειτουργία του
- Οι λειτουργικές απαιτήσεις καθορίζουν τις υπηρεσίες που παρέχονται από το σύστημα
- Οι μη λειτουργικές απαιτήσεις θέτουν περιορισμούς είτε στο ίδιο το σύστημα είτε στη διαδικασία ανάπτυξης
- Οι απαιτήσεις χρήστη είναι δηλώσεις υψηλού επιπέδου του τι πρέπει να κάνει το σύστημα, από τη σκοπιά του χρήστη

Case Study: Γραμματεία (1/1)

➤ Πελάτης

- Υπεύθυνος λειτουργίας της γραμματείας

➤ Σκοπιμότητα:

- Το μεγάλο πλήθος φοιτητών, καθηγητών και μαθημάτων καθώς και η πολυπλοκότητα των εργασιών υποστήριξης επιβάλλουν τη χρήση αυτοματοποιημένης εφαρμογής

Case Study: Γραμματεία (1/2)

► Περιγραφή:

- Η εφαρμογή θα πρέπει να τηρεί αρχεία φοιτητών, καθηγητών, μαθημάτων, εγγραφής σε μαθήματα, καθώς και αποτελέσματα βαθμολογίας.
- Η εφαρμογή θα πρέπει να τυπώνει καταστάσεις φοιτητών, καθηγητών, μαθημάτων και βαθμολογίας, με κριτήρια που θα δίνει ο χρήστης.
- Η εφαρμογή δε θα πρέπει να επιτρέπει τη διαγραφή ενός φοιτητή ή καθηγητή από το αρχείο αν αυτός έχει εγγραφεί σε ή του έχει ανατεθεί μάθημα, αντίστοιχα.
- Το περιβάλλον λειτουργίας θα είναι ένας αυτόνομος ηλεκτρονικός υπολογιστής με Windows XP

Έγγραφο προδιαγραφών λογισμικού γραμματείας

- **Λειτουργική απαίτηση A1:** Τήρηση αρχείου σπουδαστών.
 - Η εφαρμογή εμφανίζει φόρμα, μέσω της οποίας ο χρήστης ενημερώνει το αρχείο σπουδαστών.
- **Είσοδος:** στοιχεία σπουδαστών από το πληκτρολόγιο.
- **Επεξεργασία:** ετοιμασία και επαλήθευση εγγραφής.
- **Έξοδοι:** ενημερωμένο αρχείο σπουδαστών ή μήνυμα λάθους
- **Σχετιζόμενη Περίπτωση χρήσης:**
 - Εισαγωγή Σπουδαστή
 - Διαγραφή Σπουδαστή
 - Αλλαγή στοιχείων Σπουδαστή

Έγγραφο προδιαγραφών λογισμικού γραμματείας

- **Λειτουργική απαίτηση A4:** Διαγραφή σπουδαστή.
 - Η εφαρμογή εμφανίζει φόρμα στην οποία ο χρήστης δίνει τα στοιχεία του σπουδαστή που επιθυμεί να διαγράψει.
- **Είσοδος:** στοιχεία σπουδαστή προς διαγραφή.
- **Επεξεργασία:** έλεγχος ύπαρξης σπουδαστή, έλεγχος εγγραφής του σε μάθημα, έλεγχος συμμετοχής του σε εξέταση.
- **Έξοδοι:** αρχείο με διαγραφμένη εγγραφή σπουδαστή ή μήνυμα λάθους.
- **Σχετιζόμενη Περίπτωση χρήσης:**
 - Διαγραφή Σπουδαστή

Τυπικές μέθοδοι περιγραφής απαιτήσεων



Τι εννοούμε;

- **Τυπικές** = Βασισμένες στα Μαθηματικά
- **Μέθοδοι** = Δομημένες προσεγγίσεις, στρατηγικές
- Χρήση των μαθηματικών με δομημένο τρόπο με σκοπό την ανάλυση και την επίλυση προβλημάτων

Χρήσεις τυπικών μεθόδων

- Υλισμικό
 - Σε όλα τα σημαντικά ολοκληρωμένα κυκλώματα
- Λογισμικό
 - Επαλήθευση λογισμικού και έλεγχος μοντέλων
- Σχεδίαση
 - OCL, Z, B
- Έλεγχος
 - Αυτόματη δημιουργία σεναρίων ελέγχου
 - Παράλληλες προσομοιώσεις

Ποιοι κλάδοι των μαθηματικών;

- Θεωρία συνόλων
- Συναρτήσεις - Συχετίσεις
- Κατηγορική λογική 1^{ης} τάξης
- Κατηγορήματα «πριν-μετά»

Θεωρία συνόλων

- «Όλοι οι άνθρωποι είναι είτε άνδρες είτε γυναίκες»
 - Άνθρωποι = Άνδρες \cup Γυναίκες
- «Κανείς δεν είναι συγχρόνως και άνδρας και γυναίκα»
 - Άνδρας \cap Γυναίκα = \emptyset

Συναρτήσεις - Συσχετίσεις

- “Κάθε πελάτης έχει τον προσωπικό του βοηθό”
- βοηθός : Πελάτες \rightarrow Υπάλληλοι
- “Κάθε πελάτης διαθέτει ένα σύνολο από λογαριασμούς”
- ΛογαριασμοίΤου: Πελάτες \rightarrow P(Λογαριασμοί)

Κατηγορική λογική 1^{ης} τάξης

79

- ➡ Όποιος δουλεύει Κυριακή, χρειάζεται να έχει ειδική άδεια

$\forall p \in \text{Employee} : \text{workOnSunday}(p) \Rightarrow \text{hasPermit}(p)$

- Κάθε πελάτης πρέπει να έχει τουλάχιστον έναν λογαριασμό

$\forall c \in \text{Customers} : \exists a \in \text{Accounts} : a \in \text{AccountsOf}(c)$

Κατηγορήματα πριν - μετά

“Οι διάφοροι άνθρωποι μπορούν να εισέρχονται στο κτήριο εφόσον έχουν μαζί τους ταυτότητα. Μόλις εισέλθουν, θα πρέπει να αφήσουν την ταυτότητα στο γραφείο υποδοχής»

EnterBuilding(p) =

PRE

hasAuthorization(p)

carriesPassport(p)

THEN

peopleInBuilding' = peopleInBuilding \cup { p }

passportsAtDesk' = passportsAtDesk \cup {passportOf(p)}

not carriesPassport(p)

Πλεονεκτήματα των τυπικών μεθόδων

- Συμπυκνωμένη γραφή
- Ακρίβεια
- Εντοπισμός **διφορούμενων** εννοιών
- Καθαρή μορφή **επικοινωνίας**
- Σε κάνει να ρωτάς τις **σωστές ερωτήσεις**

Συμπυκνωμένη γραφή

- Συγκρίνεται τα ακόλουθα:
 - “Για κάθε εισιτήριο που εκδίδεται, θα πρέπει να υφίσταται αντίστοιχο άτομο που να του επιτρέπεται η είσοδος στην συναυλία με αυτό το εισιτήριο. Το άτομο αυτό καλείται κάτοχος του εισιτηρίου”
 - **TicketOwner: IssuedTickets → Person**

Ακρίβεια

- “Οι άνθρωποι **συνήθως** σταματούν τα αυτοκίνητα τους στα κόκκινα φανάρια”
- Τι σημαίνει **συνήθως**;
- Πως θα χτίσουμε ένα σύστημα που θα ενσωματώνει αυτήν την προδιαγραφή;
- Ποιες είναι οι συνέπειες;
- Τι θα πρέπει να γίνει όταν ο X άνθρωπος δεν σταματήσει;
- **Αν δεν αποκατασταθεί η ακρίβεια τότε δεν μπορούμε να έχουμε τυπική περιγραφή**

Εντοπισμός διαφορούμενων εννοιών

- “Όταν η θερμοκρασία είναι πολύ υψηλή τότε θα πρέπει να κλείσει ο εξαερισμός ή θα πρέπει να ενημερωθεί το προσωπικό συντήρησης”
- Μπορούμε να κάνουμε και τα δύο; Ή μήπως όχι;
 - `temperature_is_high => (notify_staff or ventilation_on)`
 - H
 - `temperature_is_high => (notify_staff xor ventilation_on)`

Καθαρή Μορφή Επικοινωνίας

- Κάθε μαθηματικός συμβολισμός έχει ένα **πολύ ακριβές νόημα**.
- Μπορούν να δομηθούν **νέες έννοιες** με βάση τις παλιές
- Τα μαθηματικά δεν απαιτούν **ευχέρεια λόγου** (γραπτού / προφορικού) και είναι αντιληπτά ανεξαρτήτως εθνικότητας

Ρωτώντας τις σωστές ερωτήσεις...

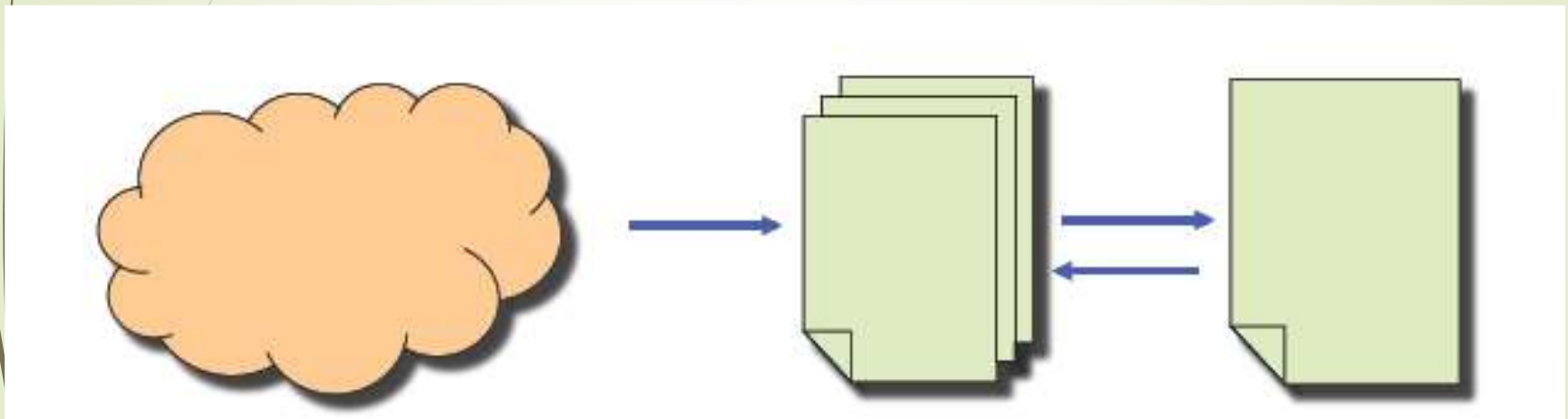
- “Κάθε πελάτης είναι είτε **έμπιστος** είτε **μη έμπιστος**”

$\forall c \in \text{customer: } \text{trusted}(c) \text{ xor } \text{untrusted}(c)$

- Μετά την πραγματοποίηση αγοράς μέσω Web, ο χρήστης καταχωρείται αυτόματα ως **νέος πελάτης**

- Ο νέος πελάτης $\text{InternetPurchase}(\text{by}) =$
 $\text{customers}' = \text{customers} \cup \{\text{by}\}$ ή μη

Η γενική προσέγγιση

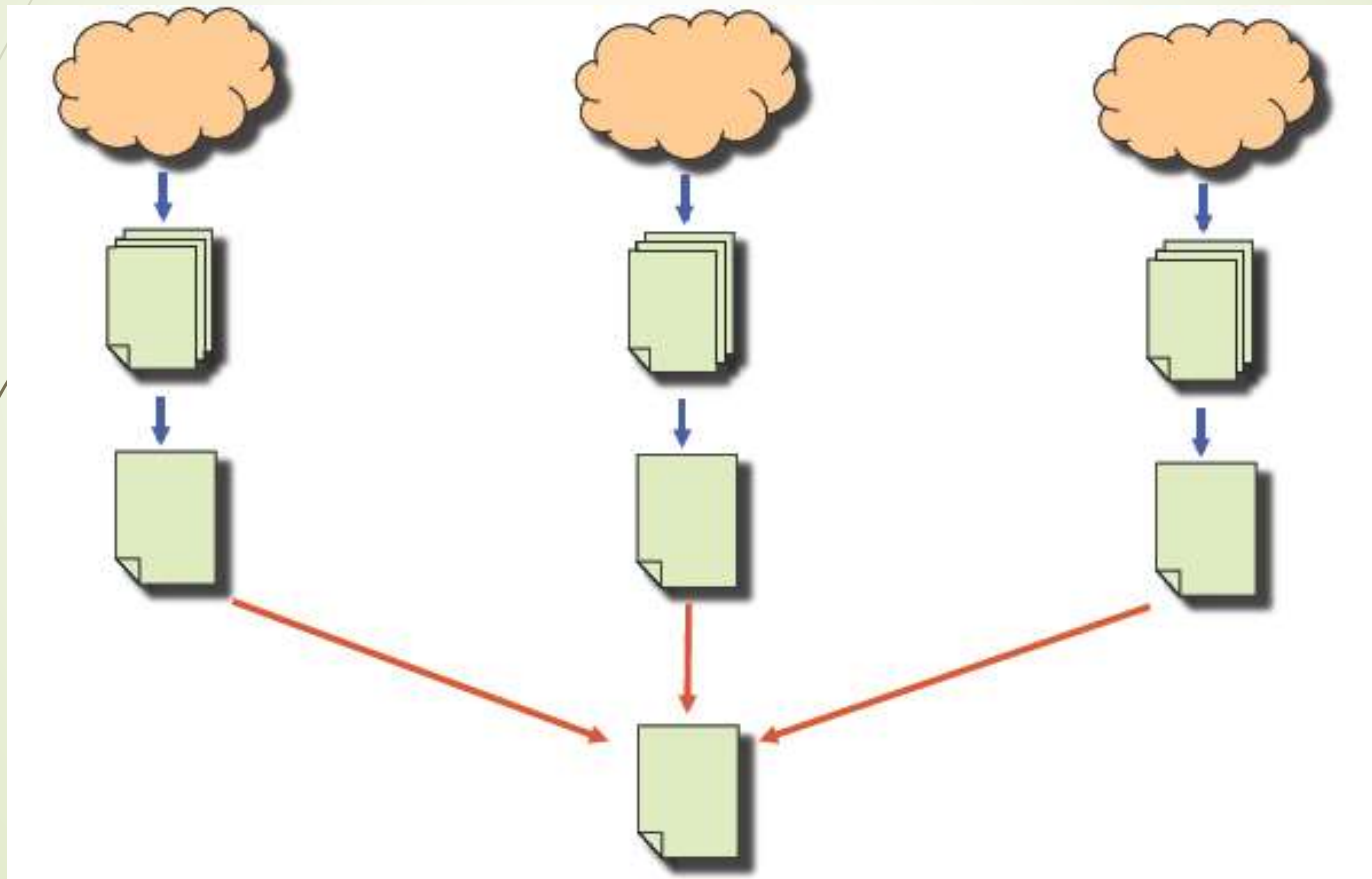


Ιδέες

**Έγγραφο σε
Φυσική Γλώσσα**

**Τυπικό
Έγγραφο**

Συνένωση σε ένα έγγραφο



Παράδειγμα

- «Το λογισμικό θα πρέπει να ελέγχει την θερμοκρασία του δωματίου. Θα διαβάζει την τρέχουσα θερμοκρασία από ένα θερμόμετρο. Εφόσον η θερμοκρασία πέσει κάτω από ένα όριο τότε θα πρέπει να ανάβει η θέρμανση για να ανεβάσει την θερμοκρασία. Εφόσον η θερμοκρασία ανέβει πάνω από ένα όριο, τότε θα πρέπει να ανάβει το ψυκτικό σύστημα για να χαμηλώσει την θερμοκρασία»

[...]

- «**Απαίτηση ασφαλείας:** Η θέρμανση και το ψυκτικό σύστημα δεν θα πρέπει ποτέ να είναι αναμμένα ταυτοχρόνως»

Τυπική περιγραφή

current_temperature : INTEGER

lower_limit: INTEGER

upper_limit: INTEGER

Τυπική περιγραφή

cooling_system : { on, off }

heating_system : { on, off }

$(\text{cooling_system} = \text{on}) \Rightarrow (\text{heating_system} = \text{off})$

$(\text{heating_system} = \text{on}) \Rightarrow (\text{cooling_system} = \text{off})$

Τυπική περιγραφή

Switch on event

```
switch_on_cooling_system =
```

```
SELECT
```

```
cooling_system = off &
```

```
current_temperature > upper_limit
```

```
THEN
```

```
cooling_system := on
```

```
END
```

Τυπική περιγραφή

Switch on event

switch_on_heating_system =

SELECT

heating_system = off &

current_temperature < lower_limit

THEN

heating_system := on

END

Η γλώσσα Z

- Αναπτύχθηκε στα τέλη του '70 στην Οξφόρδη
- Πρότυπο ISO από το 2002 (ISO/IEC 13568:2002)
- Μεγάλη υποστήριξη από
 - Μεγάλη κοινότητα χρηστών
 - Πλήθος σχετικών εργαλείων



Ερωτήσεις;