

---

# Διαδικασίες παραγωγής λογισμικού

---

# Περιεχόμενα

---

- Παρουσίαση μοντέλων διεργασίας ανάπτυξης λογισμικού
- Περιγραφή τριών γενικών μοντέλων διεργασίας ανάπτυξης λογισμικού
- Γενική περιγραφή των διαδικασιών που περιλαμβάνονται στην καταγραφή των απαιτήσεων του λογισμικού, στην ανάπτυξη του λογισμικού, στη δοκιμή του, και στην εξέλιξή του
- Μοντέλο Ενοποιημένης Διαδικασίας
- Γνωριμία με τα εργαλεία CASE

# Η διαδικασία παραγωγής λογισμικού

---

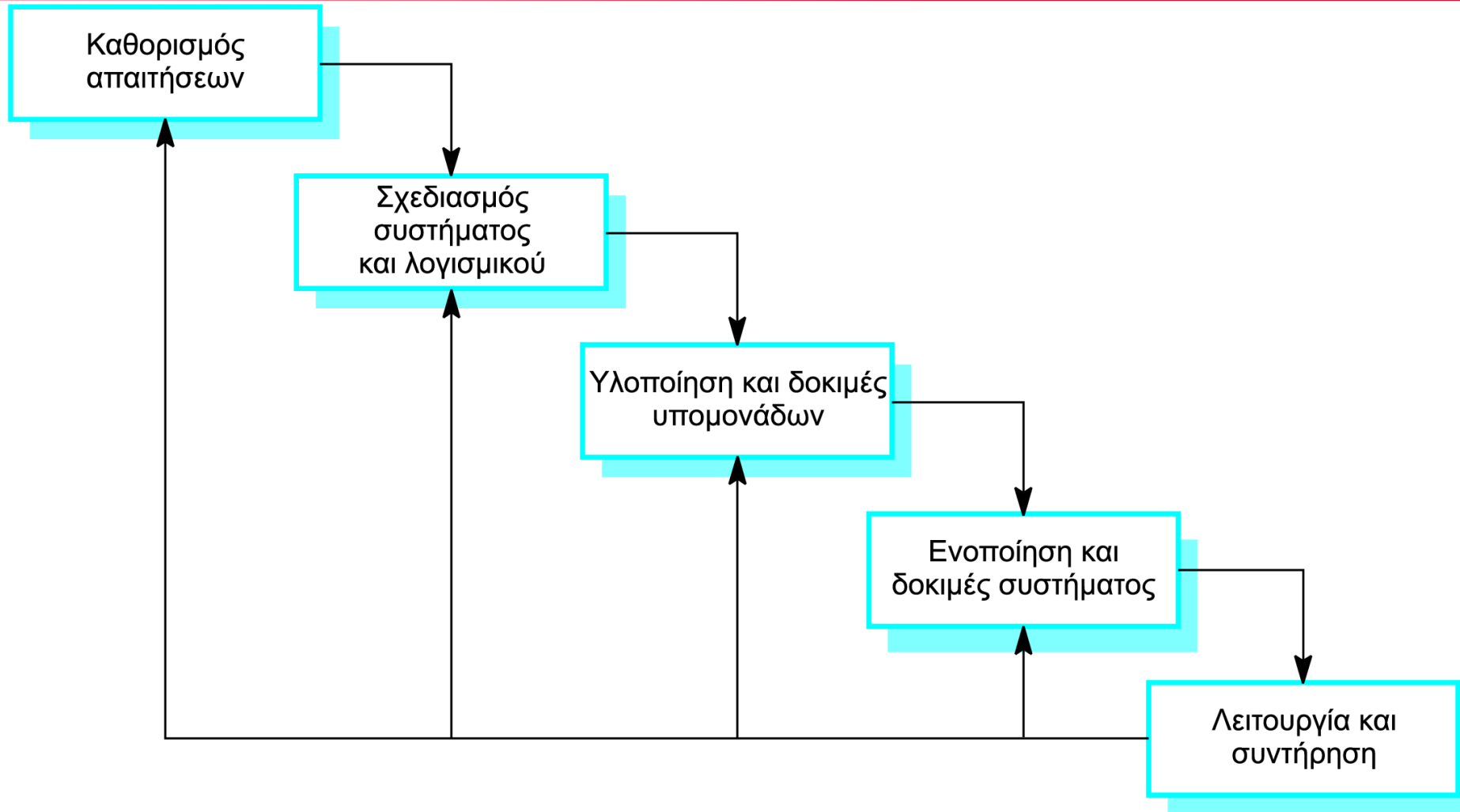
- Ένα δομημένο σύνολο δραστηριοτήτων που απαιτούνται για την ανάπτυξη ενός συστήματος λογισμικού
  - Εξαγωγή προδιαγραφών λογισμικού
  - Σχεδιασμός
  - Επικύρωση
  - Εξέλιξη
- Τα μοντέλα διαδικασιών παραγωγής λογισμικού είναι αφηρημένες αναπαραστάσεις κάποιας διαδικασίας παραγωγής λογισμικού. Επομένως κάθε μοντέλο παρέχει πληροφορίες για ορισμένες μόνο πλευρές της διαδικασίας.

# Γενικά μοντέλα (υποδείγματα) διαδικασιών ανάπτυξης λογισμικού

---

- Το μοντέλο καταρράκτη
  - Ξεχωριστές φάσεις καθορισμού απαιτήσεων και ανάπτυξης.
- Εξελικτική ανάπτυξη
  - Η προσέγγιση αυτή αναμιγνύει τις δραστηριότητες της εξαγωγής προδιαγραφών, της ανάπτυξης, και της επικύρωσης.
- Τεχνολογία λογισμικού βάσει συστατικών στοιχείων
  - Το σύστημα συγκροτείται από υπάρχοντα συστατικά στοιχεία.

# Μοντέλο καταρράκτη



# Φάσεις μοντέλου καταρράκτη

---

- Ανάλυση και καθορισμός απαιτήσεων
- Σχεδιασμός συστήματος και λογισμικού
- Υλοποίηση και δοκιμές υπομονάδων
- Ενοποίηση και δοκιμές συστήματος
- Λειτουργία και συντήρηση
- Το βασικό μειονέκτημα του μοντέλου καταρράκτη είναι η δυσκολία στην πραγματοποίηση τροποποιήσεων αφού η διαδικασία έχει ξεκινήσει. Πρέπει να ολοκληρωθεί μία φάση για να προχωρήσει η διαδικασία στην επόμενη φάση.

# Μειονεκτήματα- Πλεονεκτήματα μοντέλου καταρράκτη

---

- **Πλεονεκτήματα**

- Καλός διαχωρισμός του έργου σε απλούστερες φάσεις.
- Κάθε φάση παράγει ένα σαφώς καθορισμένο παραδοτέο.

- **Μειονεκτήματα**

- Στην πράξη οι φάσεις αλληλεπικαλύπτονται.
- Στην πράξη το μοντέλο δεν είναι γραμμικό: συχνά επιστρέφουμε στην προηγούμενη φάση.
- Συχνά, αλλαγές σε κάποιο στάδιο επιβάλλουν την οπισθοχώρηση και πραγματοποίηση αλλαγών σε πολλά από τα προηγούμενα στάδια.
- Ο πελάτης βλέπει τι τελικά αγοράζει πολύ αργά !!!
- Έλλειψη καθοδήγησης για το χειρισμό δραστηριοτήτων (μετασχηματισμού λογ/κού)

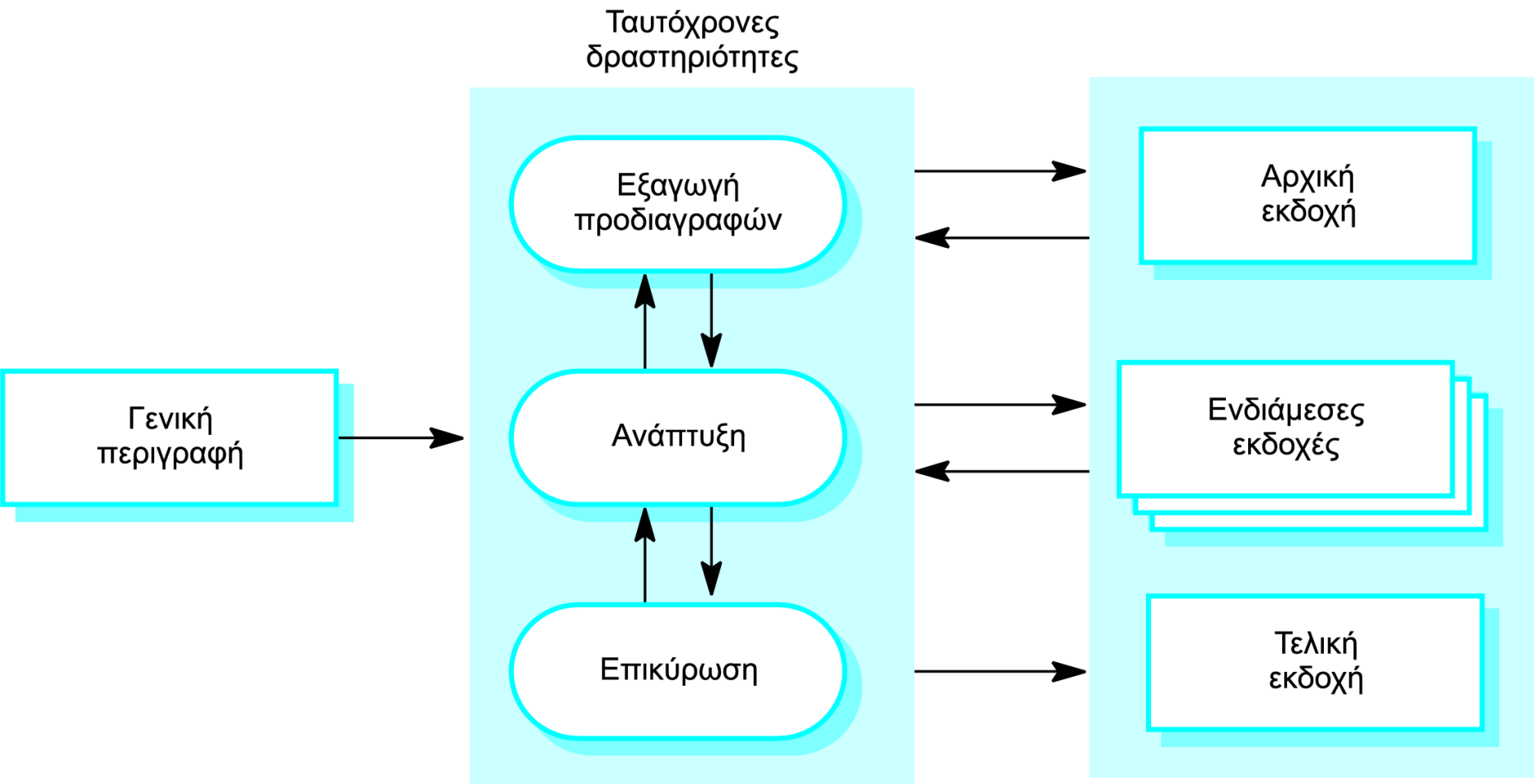
# Εξελικτική ανάπτυξη

---

- **Διερευνητική ανάπτυξη**
  - Η διαδικασία έχει στόχο τη συνεργασία με τον πελάτη προκειμένου να αναπτυχθεί το τελικό σύστημα από το αρχικό περίγραμμα προδιαγραφών. Ξεκινά από τις απαιτήσεις που είναι καλά κατανοητές και εξελίσσεται με την προσθήκη νέων δυνατοτήτων που προτείνονται από τον πελάτη.
- **Δημιουργία αναλώσιμων πρωτοτύπων**
  - Ο στόχος είναι η κατανόηση των απαιτήσεων του συστήματος. Ξεκινά από τις απαιτήσεις που δεν είναι καλά κατανοητές για να διασαφηνιστεί τι πραγματικά χρειάζεται.



# Εξελικτική ανάπτυξη



# Εξελικτική ανάπτυξη

---

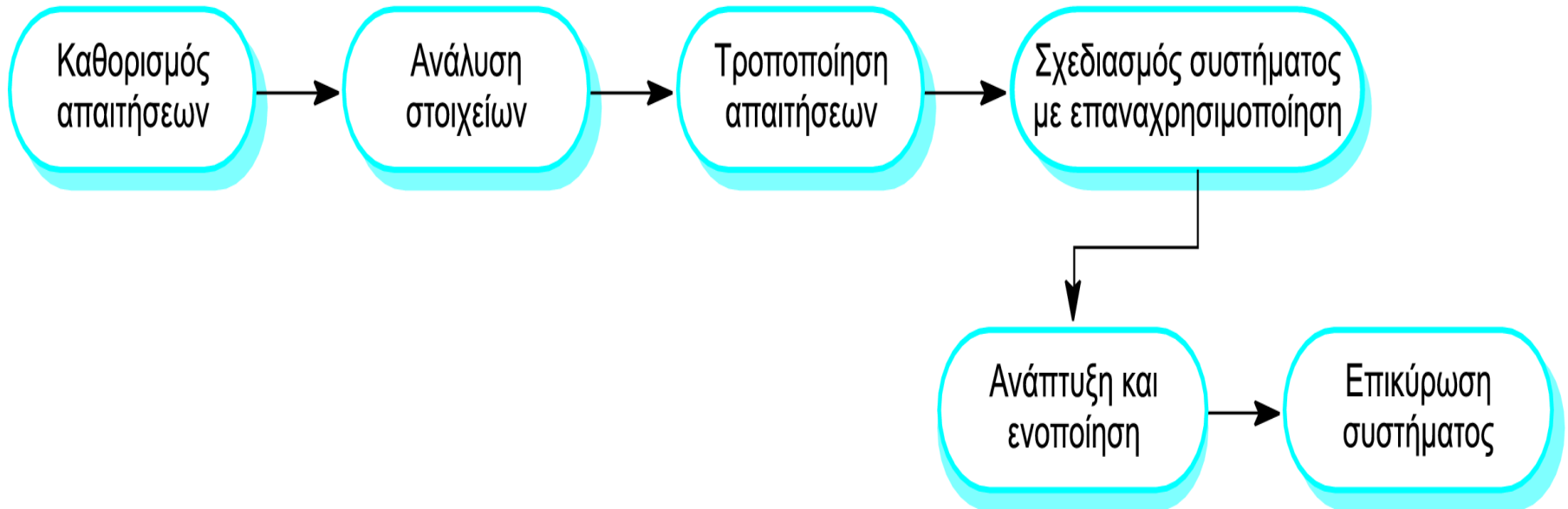
- Προβλήματα
  - Έλλειψη διαφάνειας της διαδικασίας
  - Τα συστήματα συχνά έχουν κακή δομή
  - Μπορεί να απαιτούνται ειδικές ικανότητες (για παράδειγμα, ικανότητες σε γλώσσες προγραμματισμού για τη γρήγορη παραγωγή πρωτοτύπων).
- Δυνατότητα εφαρμογής
  - Σε διαδραστικά συστήματα μικρού ή μεσαίου μεγέθους
  - Σε τμήματα μεγάλων συστημάτων (για παράδειγμα, για τη διασύνδεση χρήστη)
  - Για συστήματα μικρής διάρκειας ζωής.

# Τεχνολογία λογισμικού βάσει συστατικών στοιχείων

---

- Βασίζεται στην αρχή της επαναχρησιμοποίησης, η οποία προβλέπει την ενσωμάτωση συστημάτων από υπάρχοντα συστατικά στοιχεία ή από εμπορικά συστήματα.
- Στάδια της διαδικασίας
  - Ανάλυση συστατικών στοιχείων
  - Τροποποίηση απαιτήσεων
  - Σχεδιασμός συστήματος με επαναχρησιμοποίηση
  - Ανάπτυξη και ενοποίηση
- Η συγκεκριμένη προσέγγιση χρησιμοποιείται ολοένα και περισσότερο λόγω της εμφάνισης προτύπων συστατικών στοιχείων.

# Ανάπτυξη προσανατολισμένη στην επαναχρησιμοποίηση



# Επαναληπτικές διαδικασίες

---

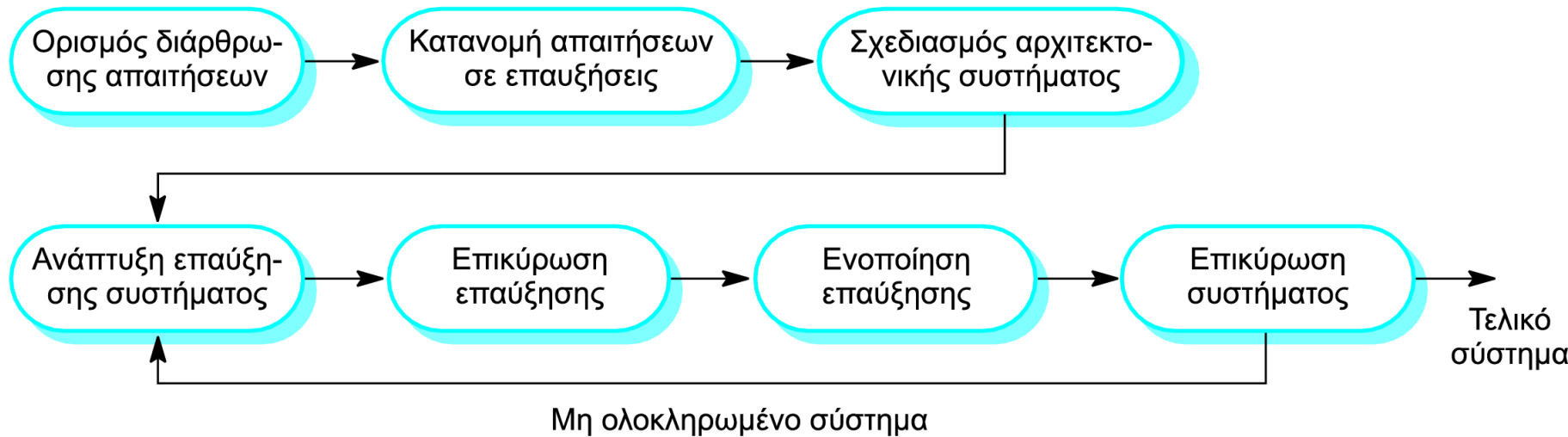
- Οι απαιτήσεις ενός συστήματος ΠΑΝΤΑ εξελίσσονται κατά την πορεία ενός έργου, συνεπώς στη διαδικασία ανάπτυξης μεγάλων συστημάτων πάντα υπάρχουν επαναληπτικές διαδικασίες που περιλαμβάνουν την αναθεώρηση προγενέστερων σταδίων.
- Η επαναληπτική μεθοδολογία μπορεί να εφαρμοστεί σε οποιοδήποτε από τα γενικά μοντέλα διαδικασιών.
- Δύο (σχετικές) προσεγγίσεις
  - Βαθμιαία παράδοση
  - Σπειροειδής ανάπτυξη

# Βαθμιαία παράδοση

---

- Το σύστημα δεν παραδίδεται σε ένα και μόνο στάδιο, αλλά η ανάπτυξη και η παράδοση χωρίζονται σε παραδοτέες επαυξήσεις, καθεμιά από τις οποίες αποδίδει μέρος των απαιτούμενων λειτουργιών.
- Ορίζεται η προτεραιότητα των απαιτήσεων των χρηστών και οι απαιτήσεις με την πιο υψηλή προτεραιότητα περιλαμβάνονται στις πρώτες επαυξήσεις.
- Αφού ξεκινήσει η ανάπτυξη μιας επαύξησης, οι απαιτήσεις της συγκεκριμένης επαύξησης παγιώνονται, αλλά οι απαιτήσεις μεταγενέστερων επαυξήσεων μπορούν να συνεχίσουν να εξελίσσονται.

# Βαθμιαία ανάπτυξη



# Πλεονεκτήματα βαθμιαίας ανάπτυξης

---

- Κάθε επαύξηση ικανοποιεί κάποιες απαιτήσεις των πελατών, επομένως μέρος των λειτουργιών του συστήματος διατίθεται προς χρήση πριν από την ολοκλήρωση του έργου.
- Οι πρώτες επαυξήσεις αποτελούν πρωτότυπα που συμβάλλουν στην κατάστρωση των απαιτήσεων για τις επαυξήσεις που ακολουθούν.
- Μικρότερος κίνδυνος ολικής αποτυχίας του έργου.
- Οι υπηρεσίες υψηλότερης προτεραιότητας συνήθως δοκιμάζονται περισσότερο.



# Ακραίος προγραμματισμός

---

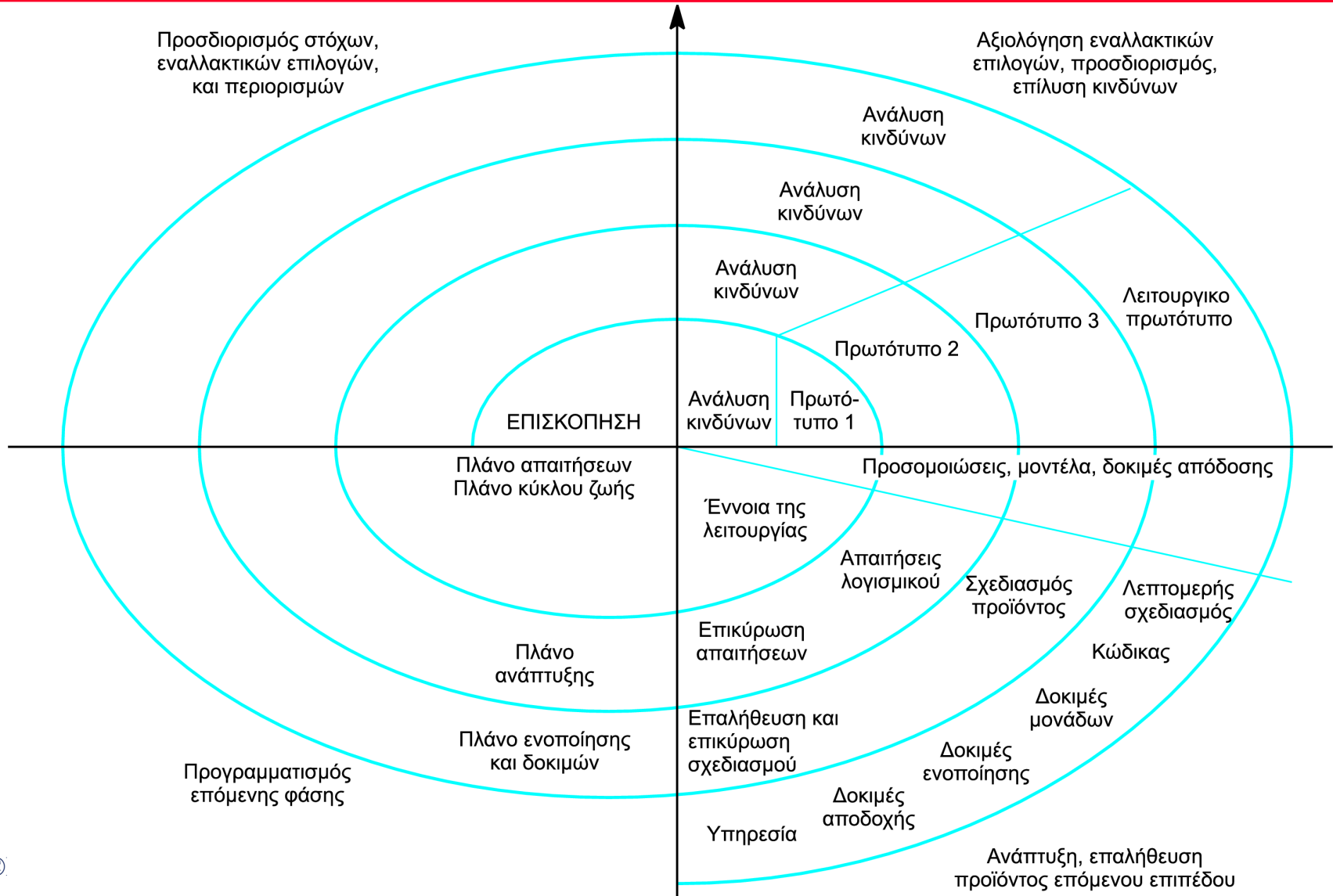
- Μια προσέγγιση που στηρίζεται στην ανάπτυξη και παράδοση πολύ μικρών επαυξήσεων λειτουργικών δυνατοτήτων.
- Βασίζεται στη σταθερή βελτίωση του κώδικα, τη συμμετοχή του πελάτη στην ομάδα ανάπτυξης και τον προγραμματισμό σε ζεύγη.
- Περισσότερα σε μεταγενέστερο μάθημα

# Σπειροειδής ανάπτυξη

---

- Η διαδικασία αναπαρίσταται με τη μορφή ελικοειδούς γραμμής και όχι ως ακολουθία δραστηριοτήτων με κάποιες επανόδους από τη μία δραστηριότητα στην άλλη.
- Κάθε βρόχος της σπείρας αντιπροσωπεύει και μια φάση της διαδικασίας.
- Δεν υπάρχουν σταθερές φάσεις όπως η φάση της εξαγωγής προδιαγραφών ή του σχεδιασμού λογισμικού - οι βρόχοι της σπείρας επιλέγονται ανάλογα με τις απαιτήσεις.
- Καθ' όλη τη διαδικασία γίνεται ρητή εκτίμηση (αξιολόγηση) και επίλυση των κινδύνων.

# Σπειροειδές μοντέλο διαδικασίας παραγωγής λογισμικού



# Τομείς σπειροειδούς μοντέλου

---

- Ορισμός αντικειμενικών στόχων
  - Προσδιορίζονται συγκεκριμένοι στόχοι σε κάθε φάση.
- Αξιολόγηση και περιορισμός κινδύνων
  - Γίνεται αξιολόγηση κινδύνων και δημιουργούνται διαδικασίες περιορισμού τους.
- Ανάπτυξη και επικύρωση
  - Επιλέγεται ένα μοντέλο για την ανάπτυξη του συστήματος - μπορεί να είναι οποιοδήποτε από τα γενικά μοντέλα.
- Προγραμματισμός
  - Γίνεται μια επισκόπηση του έργου και προγραμματισμός της επόμενης φάσης της σπειροειδούς ανάπτυξης.

# Δραστηριότητες διαδικασιών

---

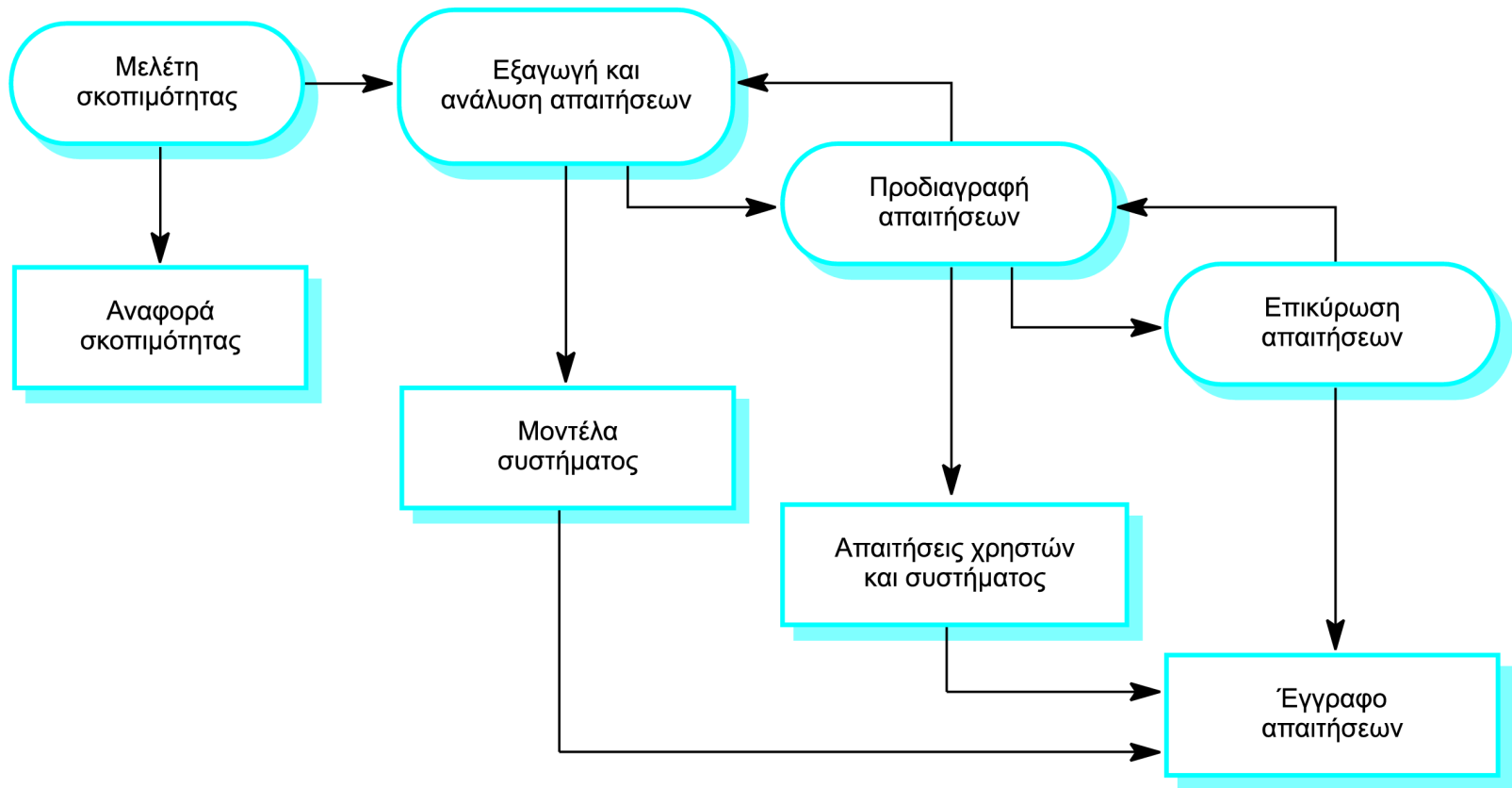
- Εξαγωγή προδιαγραφών λογισμικού
- Σχεδιασμός και υλοποίηση λογισμικού
- Επικύρωση λογισμικού
- Εξέλιξη λογισμικού

# Εξαγωγή προδιαγραφών λογισμικού

---

- Η διαδικασία καθορισμού των απαιτούμενων υπηρεσιών των περιορισμών λειτουργίας και ανάπτυξης του συστήματος.
- Διαδικασία καθορισμού απαιτήσεων
  - Μελέτη σκοπιμότητας
  - Εξαγωγή και ανάλυση απαιτήσεων
  - Προδιαγραφή απαιτήσεων
  - Επικύρωση απαιτήσεων

# Διαδικασία καθορισμού απαιτήσεων



# Σχεδιασμός και υλοποίηση λογισμικού

---

- Η διαδικασία της μετατροπής ενός συνόλου προδιαγραφών σε ένα σύστημα λογισμικού που λειτουργεί.
- Σχεδιασμός λογισμικού
  - Σχεδιασμός μιας δομής λογισμικού που υλοποιεί το σύνολο προδιαγραφών
- Υλοποίηση
  - Μετάφραση της δομής αυτής σε εκτελέσιμο πρόγραμμα
- Οι δραστηριότητες του σχεδιασμού και της υλοποίησης σχετίζονται στενά και μπορεί να παρεμβάλλονται.

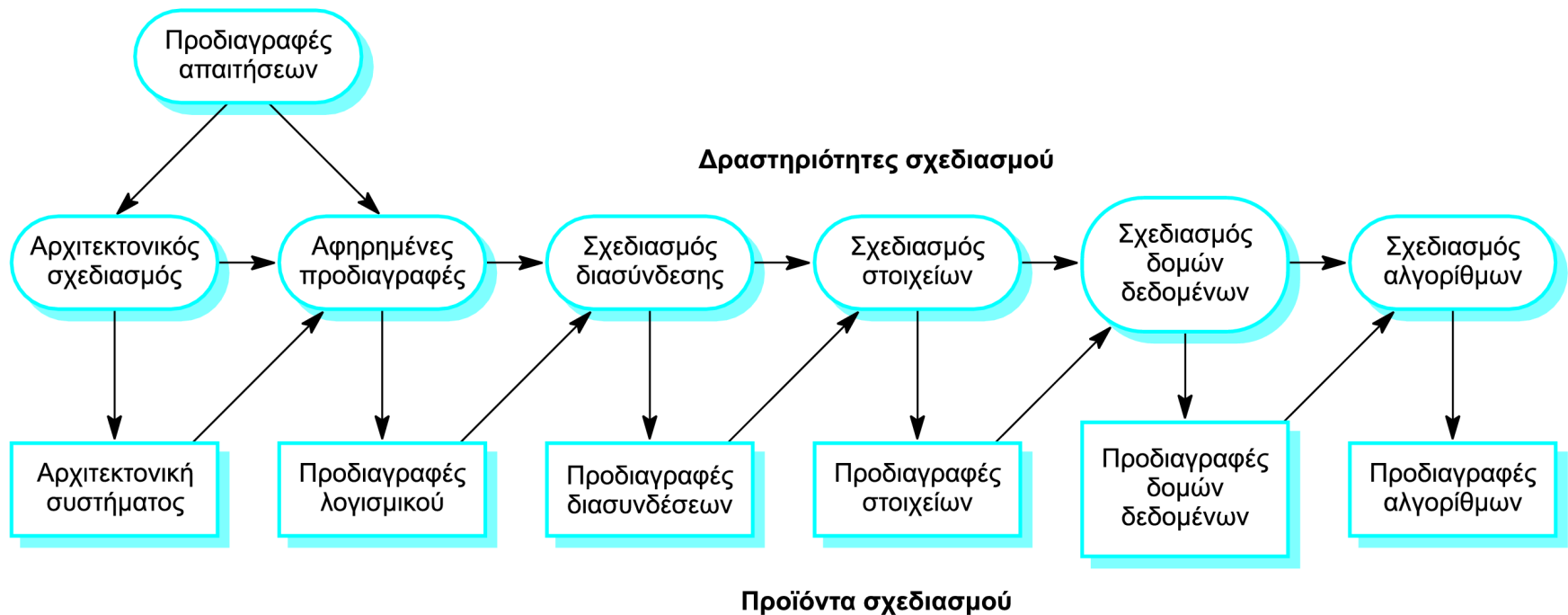


# Δραστηριότητες της διαδικασίας σχεδιασμού

---

- Αρχιτεκτονικός σχεδιασμός
- Αφηρημένες προδιαγραφές
- Σχεδιασμός διασυνδέσεων
- Σχεδιασμός συστατικών στοιχείων
- Σχεδιασμός δομών δεδομένων
- Σχεδιασμός αλγορίθμων

# Η διαδικασία παραγωγής λογισμικού



# Δομημένες μέθοδοι

---

- Συστηματικές προσεγγίσεις για την ανάπτυξη του σχεδιασμού λογισμικού.
- Ο σχεδιασμός συνήθως τεκμηριώνεται ως σύνολο μοντέλων με αναπαραστάσεις.
- Πιθανά μοντέλα
  - Μοντέλο αντικειμένων
  - Μοντέλο ακολουθίας
  - Μοντέλο μετάβασης κατάστασης
  - Μοντέλο δομής
  - Μοντέλο ροής δεδομένων

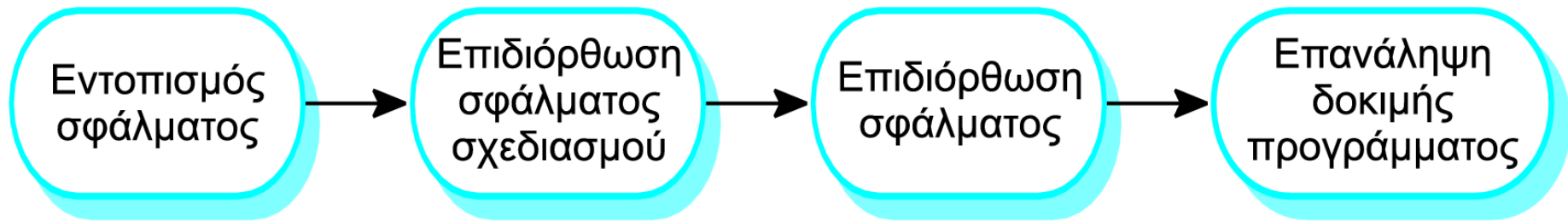
# Προγραμματισμός και αποσφαλμάτωση

---

- Μετάφραση του σχεδιασμού λογισμικού σε πρόγραμμα και απαλοιφή των σφαλμάτων από αυτό.
- Ο προγραμματισμός είναι μια προσωπική δραστηριότητα και δεν υπάρχει κάποια γενική διαδικασία που συνήθως ακολουθείται.
- Οι προγραμματιστές κάνουν κάποιες δοκιμές στον κώδικα του προγράμματος για να αποκαλύψουν ελαττώματά του και να τα απαλείψουν κατά τη διαδικασία αποσφαλμάτωσης.

# Η διαδικασία αποσφαλμάτωσης

---

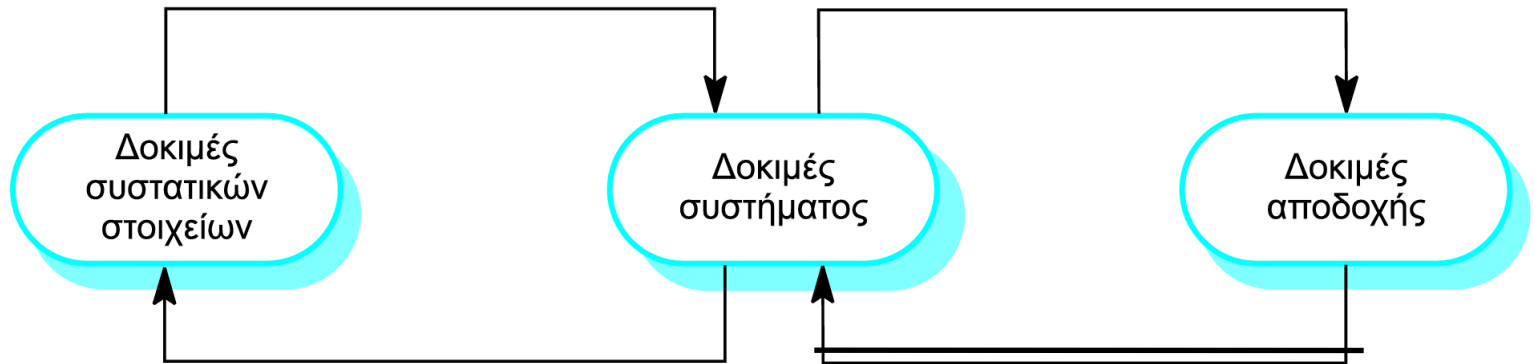


# Επικύρωση λογισμικού

---

- Οι διαδικασίες επαλήθευσης και επικύρωσης έχουν στόχο να δείξουν ότι το σύστημα είναι σύμφωνο με τις προδιαγραφές του και ότι ικανοποιεί τις προσδοκίες του πελάτη που το αγοράζει.
- Περιλαμβάνουν τον έλεγχο και την επισκόπηση διαδικασιών, καθώς και δοκιμές του συστήματος.
- Η δοκιμή ενός συστήματος συνεπάγεται την εκτέλεσή του με στιγμιότυπα δεδομένων ελέγχου τα οποία εξάγονται από τις προδιαγραφές των πραγματικών δεδομένων που πρόκειται να επεξεργαστεί το σύστημα.

# Η διαδικασία δοκιμών



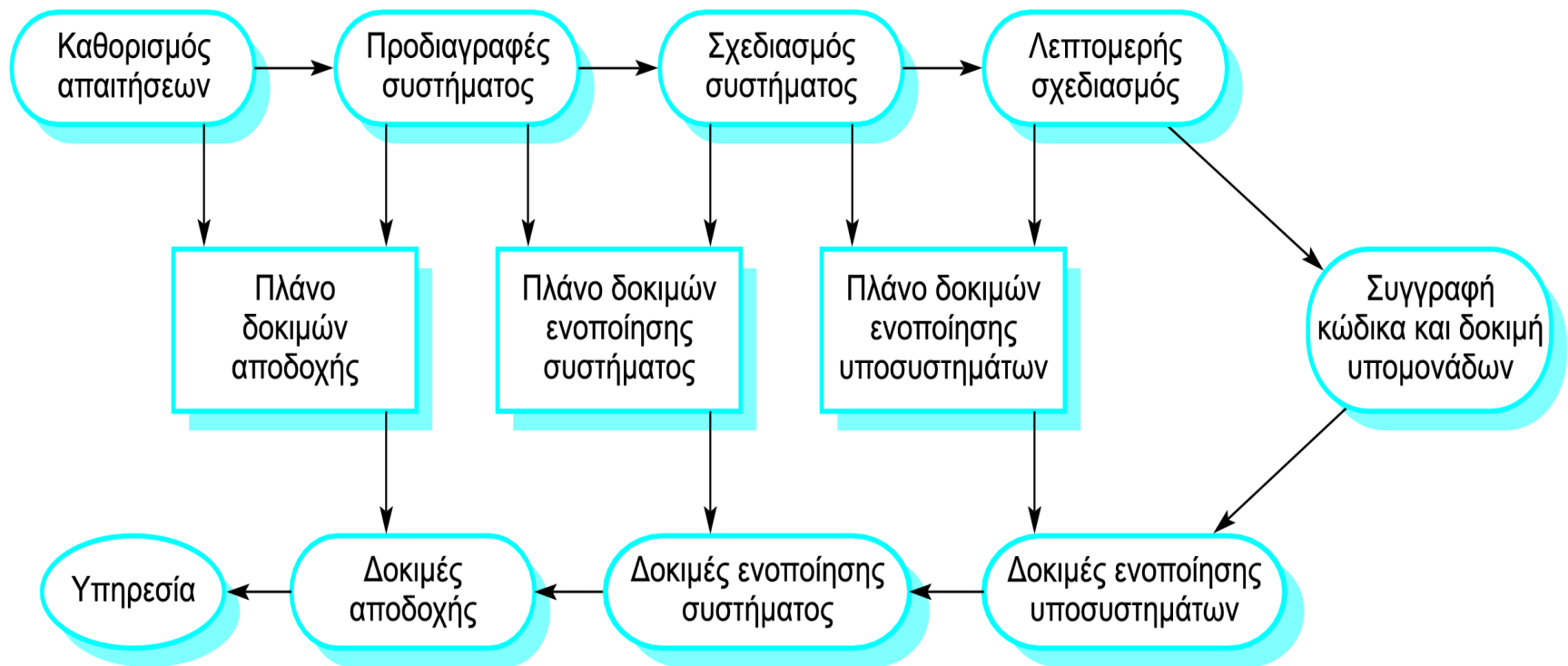
# Στάδια δοκιμών

---

- Δοκιμές συστατικών στοιχείων (ή υπομονάδων)
  - Κάθε συστατικό στοιχείο δοκιμάζεται ανεξάρτητα
  - Τα στοιχεία μπορεί να είναι συναρτήσεις, αντικείμενα ή συναφείς ομαδοποιήσεις τέτοιων οντοτήτων.
- Δοκιμές συστήματος
  - Δοκιμή του συστήματος στο σύνολό του. Είναι ιδιαίτερα σημαντική η δοκιμή των ανακυπτουσών ιδιοτήτων.
- Δοκιμές αποδοχής
  - Δοκιμή με δεδομένα που παρέχονται από τον πελάτη για να ελεγχθεί ότι το σύστημα πληροί τις ανάγκες του.



# Φάσεις δοκιμών

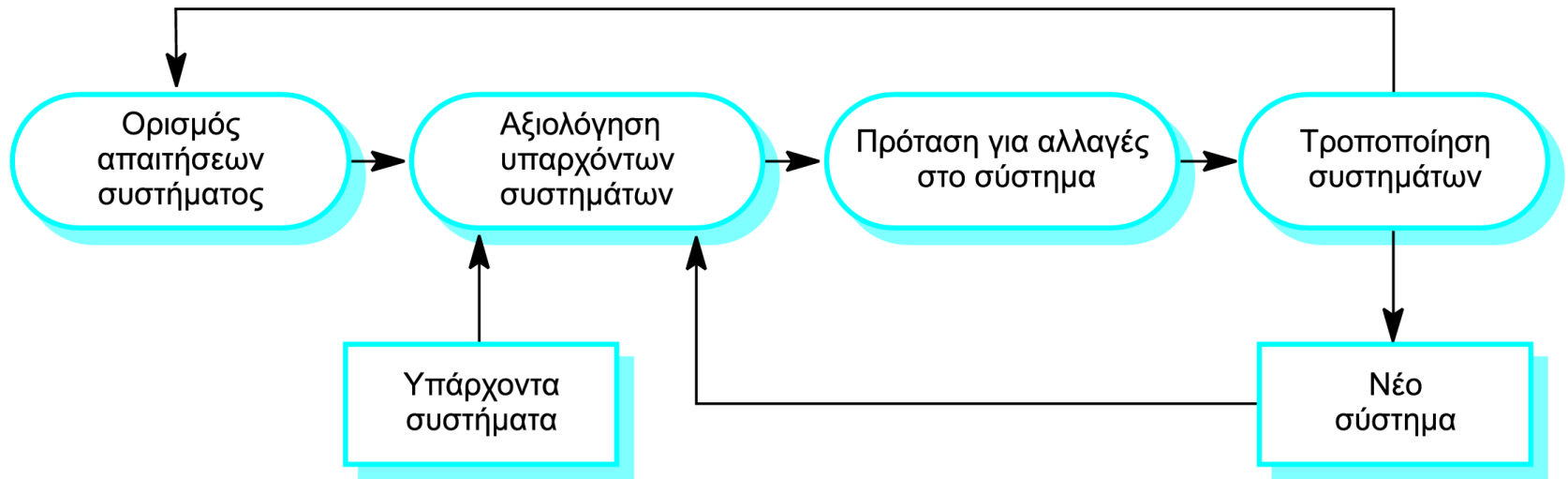


# Εξέλιξη λογισμικού

---

- Το λογισμικό είναι ευέλικτο από τη φύση του και μπορεί να τροποποιείται.
- Καθώς αλλάζουν οι απαιτήσεις λόγω των μεταβαλλόμενων επιχειρηματικών συνθηκών, το λογισμικό που υποστηρίζει την επιχείρηση πρέπει επίσης να εξελίσσεται και να μεταβάλλεται.
- Αν και υπήρχε μια διαχωριστική γραμμή μεταξύ της διαδικασίας ανάπτυξης λογισμικού και της διαδικασίας εξέλιξης (συντήρησης), η διάκριση αυτή ολοένα και χάνει το νόημά της αφού όλο και λιγότερα συστήματα λογισμικού είναι σήμερα εντελώς καινούργια.

# Εξέλιξη συστήματος



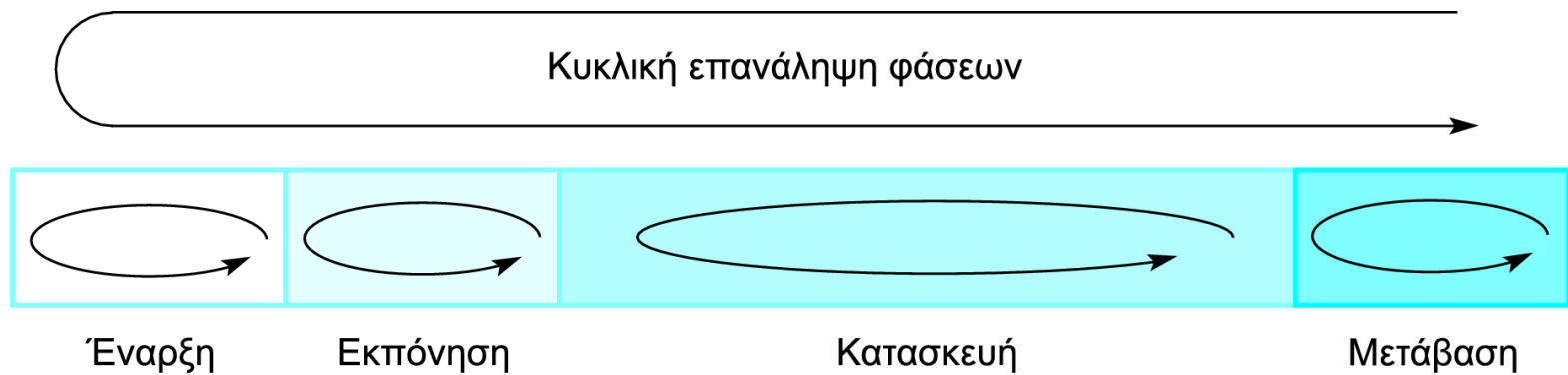
# Rational Unified process

## Ενοποιημένη Διαδικασία (ΕΔ)

---

- Ένα σύγχρονο παράδειγμα μοντέλου διαδικασίας που προέρχεται από την προσπάθεια που έχει επενδυθεί στη γλώσσα UML και τη σχετική διαδικασία.
- Συνήθως περιγράφεται από 3 προοπτικές
  - Μια δυναμική προοπτική, που δείχνει τις φάσεις του μοντέλου μέσα στο χρόνο
  - Μια στατική προοπτική, η οποία παρουσιάζει τις δραστηριότητες της διαδικασίας
  - Μια πρακτική προοπτική, που προτείνει τη χρήση καλών πρακτικών κατά τη διαδικασία

# Μοντέλο φάσεων RUP



# Φάσεις RUP

---

- Έναρξη
  - Σύσταση της επιχειρηματικής χρήσης του συστήματος.
- Εκπόνηση/ Επεξεργασία
  - Κατανόηση του πεδίου του προβλήματος και διατύπωση της αρχιτεκτονικής για το σύστημα.
- Κατασκευή
  - Σχεδιασμός, προγραμματισμός και δοκιμές του συστήματος.
- Μετάβαση
  - Μεταφορά του συστήματος στο περιβάλλον λειτουργίας του.

# Καλές πρακτικές RUP

---

- Επαναληπτική ανάπτυξη του λογισμικού
- Διαχείριση απαιτήσεων
- Χρήση αρχιτεκτονικών βάσει συστατικών στοιχείων
- Οπτική μοντελοποίηση του λογισμικού.
- Επαλήθευση ποιότητας του λογισμικού
- Έλεγχος μεταβολών του λογισμικού

# Ροές εργασιών

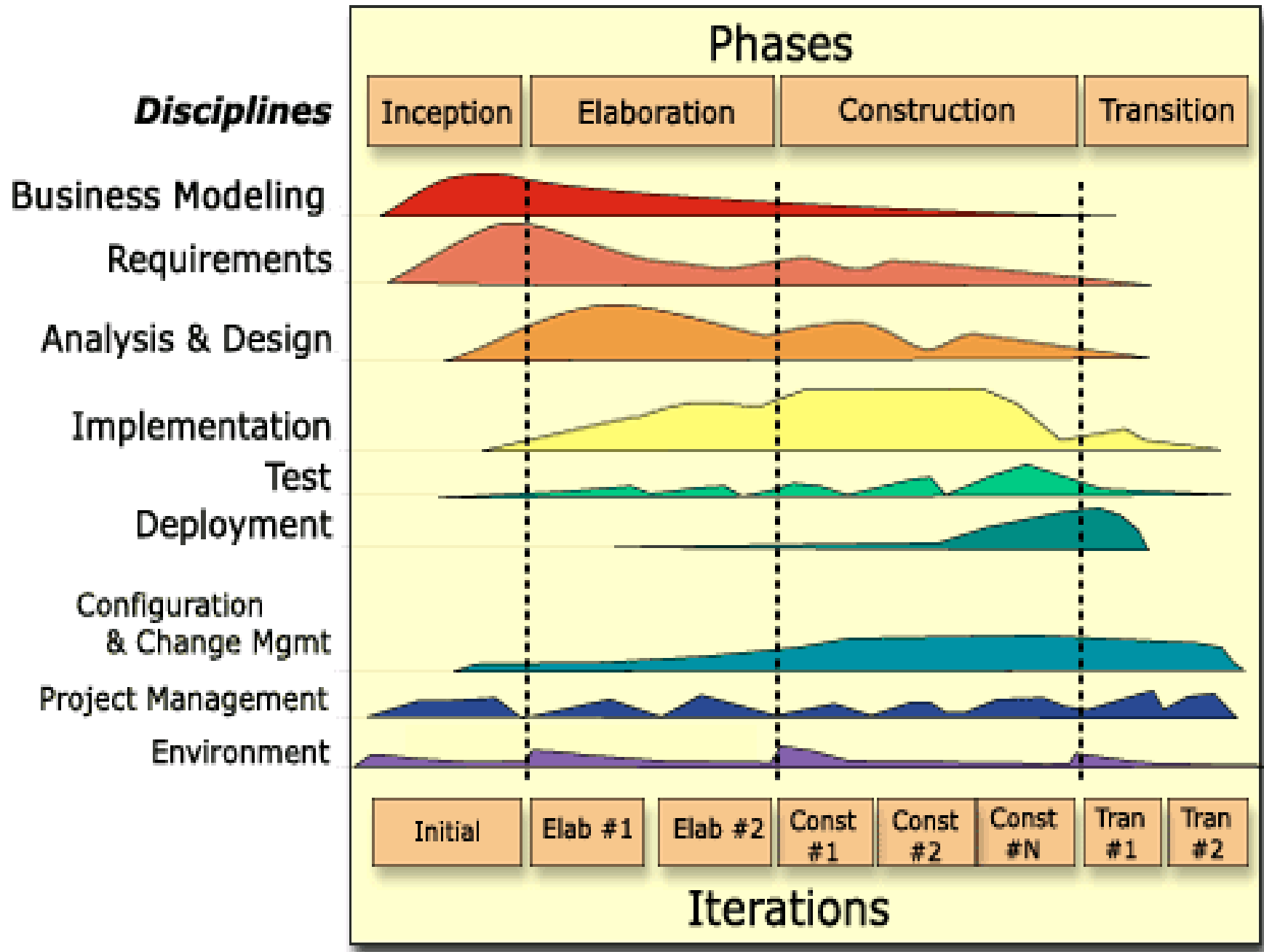
---

- Υπάρχουν 9 διαφορετικές ροές
  - Επιχειρηματικά μοντέλα
  - Καταγραφή απαιτήσεων
  - Ανάλυση και ανάπτυξη
  - Υλοποίηση
  - Έλεγχος
  - Παράδοση
  - Διαχείριση έργου
  - Διαχείριση κώδικα
  - Περιβάλλον



# Στατικές ροές εργασιών

---



# Η εναρκτήρια φάση

---

- Σκοπός της είναι να εξετάσει κατά πόσο το έργο μπορεί να γίνει και να καθορίσει κάποιο κοινό στόχο από όλα τα ενδιαφερόμενα μέλη
- Αν έχει προ-αποφασιστεί ότι το έργο θα γίνει τότε η εναρκτήρια φάση είναι πολύ σύντομη, το πολύ μία εβδομάδα ή λιγότερο
- Συνήθως η εναρκτήρια φάση για τα περισσότερα έργα δεν κρατά παρά λίγες εβδομάδες (2-3)
- Τι γίνεται στην εναρκτήρια φάση
  - Ορίζονται οι στόχοι του έργου (Vision Document). Αυτό είναι ένα έγγραφο που μπορεί να είναι από μια παράγραφο ως μερικές σελίδες και περιέχει μια γενική σύνοψη του έργου
  - Παράγεται ένα μοντέλο περιπτώσεων χρήσης
  - Γίνεται ανάλυση κόστους/κέρδους και ρίσκου
  - Ορίζεται ένα 10%-20% των περιπτώσεων χρήσης
  - Μπορεί να κατασκευαστούν και κάποια δείγματα

# Η φάση επεξεργασίας

---

- Ορίζεται η πλειοψηφία των απαιτήσεων του έργου
- Αντιμετωπίζονται οι πιθανοί κίνδυνοι
- Υλοποιούνται τα βασικά κομμάτια της αρχιτεκτονικής του έργου
- Η φάση επεξεργασίας ολοκληρώνεται συνήθως σε 2 ως 4 επαναλήψεις.
- Κάθε επανάληψη έχει προ-καθορισμένη διάρκεια (συνήθως από 2 ως 4 εβδομάδες)
- Περίπου το 80% των περιπτώσεων χρήσης πρέπει να έχει καταγραφεί σε αυτήν την φάση
- Πρέπει να έχει οριστεί το μοντέλο περιπτώσεων χρήσης
- Πρέπει να έχει οριστεί το μοντέλο σχεδίασης
- Πρέπει να έχει οριστεί το τι θα ελεγχθεί στο τέλος
- Πρέπει να έχει οριστεί η δομή της βάσης δεδομένων
- Πρέπει στο τέλος να έχει παράγει εκτελέσιμο κώδικα με υλοποιημένη τη βασική αρχιτεκτονική

# Η κατασκευαστική φάση

---

- Αφορά την υλοποίηση του έργου
- Έχει διάρκεια συνήθως  $\geq 4$  επαναλήψεις
- Κάθε επανάληψη έχει προ-καθορισμένη διάρκεια (συνήθως από 2 ως 4 εβδομάδες)
- Εστιάζει στην υλοποίηση
- Υλοποιεί τις περιπτώσεις χρήσης της προηγούμενης φάσης και αυξάνει τη λειτουργικότητα
- Υλοποιούνται και λεπτομέρειες και όχι μόνο τα βασικά του αρχιτεκτονικού σχεδίου
- Η ανάλυση συνεχίζεται αλλά η σχεδίαση και η υλοποίηση είναι τα βασικά

# Η φάση μετάβασης

---

- Αρχίζει όταν ολοκληρωθεί η υλοποίηση και το λογισμικό παραδοθεί στον πελάτη
- Έχει διάρκεια συνήθως μέχρι 2 επαναλήψεις
- Κάθε επανάληψη έχει προ-καθορισμένη διάρκεια (συνήθως από 2 ως 4 εβδομάδες)
- Σηματοδοτεί την έναρξη χρήσης του συστήματος
- Ο έλεγχος μεταφέρεται περισσότερο στην ομάδα συντήρησης και ελέγχου
- Αφορά κυρίως την ενοποίηση με τα υπάρχοντα συστήματα

# Οι επαναλήψεις και η UP

---

- Η UP μια επαναληπτική διαδικασία ανάπτυξης.
- Η ανάπτυξη του προϊόντος οργανώνεται γύρω από επαναλήψεις συγκεκριμένης διάρκειας (π.χ. 4 εβδομάδων).
- Κάθε επανάληψη είναι ένα mini-project που μοιάζει με τον καταράκτη.
- Κάθε επανάληψη καταλήγει σε ένα εκτελέσιμο, δοκιμασμένο σύστημα που απλά μπορεί να μην υλοποιεί όλες τις απαιτήσεις

# Η σχέση ανάμεσα σε ροές και φάσεις

---

- Οι ροές εκτελούνται επαναληπτικά κάθε φορά που μπαίνουμε σε μια καινούργια φάση
- Το πόσος χρόνος δαπανάται σε κάθε ροή εξαρτάται από τη φάση που βρισκόμαστε
- Δεν είναι απαραίτητο να εκτελεστούν όλες οι ροές σε κάθε φάση



# Τεχνολογία λογισμικού με τη βοήθεια υπολογιστή

---

- Η τεχνολογία λογισμικού με τη βοήθεια υπολογιστή (CASE) αναφέρεται σε εφαρμογές που χρησιμοποιούνται για την υποστήριξη των διαδικασιών ανάπτυξης και εξέλιξης λογισμικού.
- Αυτοματοποίηση δραστηριοτήτων
  - Προγράμματα επεξεργασίας με γραφικό περιβάλλον για την ανάπτυξη μοντέλων συστημάτων
  - Λεξικό δεδομένων για τη διαχείριση των οντοτήτων του σχεδιασμού
  - Πρόγραμμα κατασκευής διασύνδεσης χρηστών με γραφικά
  - Προγράμματα αποσφαλμάτωσης για την εύρεση ελαττωμάτων στον κώδικα
  - Αυτόματοι μεταφραστές (μεταξύ γλωσσών προγραμματισμού) για την παραγωγή νέων εκδόσεων προγραμμάτων.

# Τεχνολογία CASE

---

- Η τεχνολογία CASE έχει οδηγήσει σε σημαντικές βελτιώσεις της διαδικασίας παραγωγής λογισμικού. Όμως οι βελτιώσεις αυτές είναι μικρότερης τάξης μεγέθους από τις προβλέψεις
  - Η τεχνολογία λογισμικού απαιτεί δημιουργική σκέψη - αυτό δεν αυτοματοποιείται εύκολα
  - Η τεχνολογία λογισμικού είναι ομαδική δραστηριότητα και, για μεγάλα έργα, πολύς χρόνος αφιερώνεται στην επικοινωνία μεταξύ των ομάδων. Η τεχνολογία CASE δεν παρέχει μεγάλη υποστήριξη σε αυτό.

# Κατηγοριοποίηση εργαλείων CASE

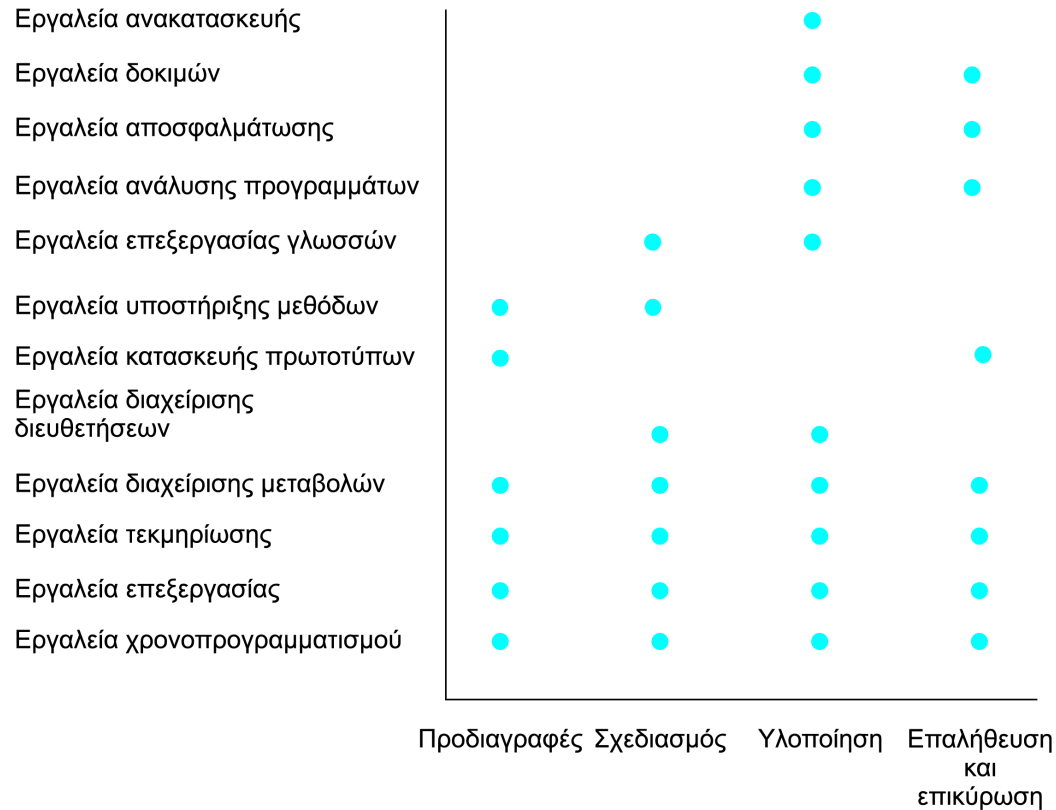
---

- Η κατηγοριοποίηση των εργαλείων CASE μάς βοηθά να κατανοήσουμε τους τύπους τους και το ρόλο τους στην υποστήριξη των δραστηριοτήτων μιας διαδικασίας παραγωγής λογισμικού.
- Λειτουργική προοπτική
  - Τα εργαλεία κατατάσσονται σύμφωνα με τη συγκεκριμένη λειτουργία που παρέχουν.
- Διαδικασιακή προοπτική
  - Τα εργαλεία κατατάσσονται σύμφωνα με τις δραστηριότητες διαδικασιών που υποστηρίζουν.
- Προοπτική ενοποίησης
  - Τα εργαλεία κατατάσσονται με βάση την οργάνωσή τους σε ενοποιημένες μονάδες.

# Λειτουργική κατηγοριοποίηση εργαλείων

---

# Κατηγοριοποίηση εργαλείων με βάση τις δραστηριότητες



# Ενσωμάτωση τεχνολογίας CASE

---

- **Εργαλεία**
  - Υποστηρίζουν μεμονωμένες εργασίες της διαδικασίας, όπως είναι ο έλεγχος της συνέπειας ενός σχεδιασμού, η επεξεργασία κειμένου, κ.λπ.
- **Εργαστήρια**
  - Υποστηρίζουν φάσεις της διαδικασίας όπως είναι οι προδιαγραφές, ο σχεδιασμός κ.λπ. Συνήθως αποτελούνται από ένα σύνολο ενοποιημένων εργαλείων.
- **Περιβάλλοντα**
  - Υποστηρίζουν ολόκληρη τη διαδικασία παραγωγής λογισμικού ή ένα σημαντικό τμήμα της. Συνήθως περιλαμβάνουν πολλά ενοποιημένα εργαστήρια.

# Εργαλεία, εργαστήρια, περιβάλλοντα

