

---

# Εισαγωγή στην Τεχνολογία Λογισμικού

# περιεχόμενα παρουσίασης

---

- Αντικείμενο της Τεχνολογίας Λογισμικού
- Η ανάπτυξη λογισμικού
- Μοντέλα διαδικασίας λογισμικού

# τεχνολογία λογισμικού

---

- Κλάδος της πληροφορικής που ασχολείται με τη μελέτη και την εφαρμογή συστηματικών, μεθοδικών και ποσοτικοποιημένων προσεγγίσεων για την ανάπτυξη, λειτουργία και συντήρηση του λογισμικού [IEEE Standard 610.12]
- Στοχεύει στην ανάπτυξη αξιόπιστου λογισμικού με μεγάλο κύκλο ζωής που ικανοποιεί τις απαιτήσεις των χρηστών και των πελατών.
- Εστιάζει τη προσοχή της στην ανάπτυξη και εφαρμογή συστηματικών μεθόδων, τεχνικών και εργαλείων που αφορούν ολόκληρο το κύκλο ζωής του Λογισμικού και υποστηρίζουν την επιτυχία των παραπάνω στόχων.

# «κρίση λογισμικού» 1968

---

- Αποκλίσεις του λογισμικού από τη λειτουργικότητα που επιθυμούσαν οι πελάτες που το παρήγγειλαν,
- Αποκλίσεις από το χρονοδιάγραμμα ανάπτυξης,
- Αποκλίσεις από τον οικονομικό προγραμματισμό ανάπτυξης του λογισμικού,
- Αποκλίσεις από το προβλεπόμενο κόστος του περιβάλλοντος λειτουργίας του λογισμικού,
- Αδυναμία εξέλιξης του λογισμικού με σκοπό τα προσαρμοσθεί στις νέες ανάγκες των πελατών

# Standish 2001

---

- το 2000 το 23% των έργων ανάπτυξης λογισμικού απέτυχαν και ματαιώθηκε η ολοκλήρωσή τους.
- Τον ίδιο χρόνο 28% των έργων ολοκληρώθηκε σύμφωνα με τον χρονικό και οικονομικό τους προγραμματισμό,
- ενώ 49% των έργων ολοκληρώθηκε με αποκλίσεις είτε στον προϋπολογισμό τους είτε στο χρονοπρογραμματισμό τους είτε στην λειτουργικότητα του προϊόντος λογισμικού

# Standish 2001

---

Παράγοντες επιτυχίας:

- Επιχειρησιακή υποστήριξη
- Εμπλοκή χρηστών
- Ικανός διοικητής έργου
- Σαφείς επιχειρησιακοί στόχοι
- Εστιασμένο πεδίο εφαρμογής του προϊόντος
- Πρότυπες υποδομές ανάπτυξης

# μια χρήσιμη αναλογία

---



# μια χρήσιμη αναλογία

---

- **Εργαλείο:** όργανο ή αυτόματο σύστημα για πραγματοποίηση μιας εργασίας με «καλύτερο τρόπο»
- **Μέθοδος – τεχνική :** μια τυπική βηματική περιγραφή δραστηριοτήτων για την παραγωγή ενός αποτελέσματος
- **Διαδικασία :** μέθοδος – τεχνική σε συνδυασμό με εργαλεία που παράγει συγκεκριμένο προϊόν
- Εκτός από τα εργαλεία, τις τεχνικές και την αρχιτεκτονική προσέγγιση υπάρχει και ο άνθρωπος (παραγωγικότητα 1 έως 10)



# ιδιαιτερότητα λογισμικού

---

- Είναι πολύπλοκο
- Απαιτεί προσαρμοστικότητα
- Είναι άυλο
- Αλλάζει διαρκώς

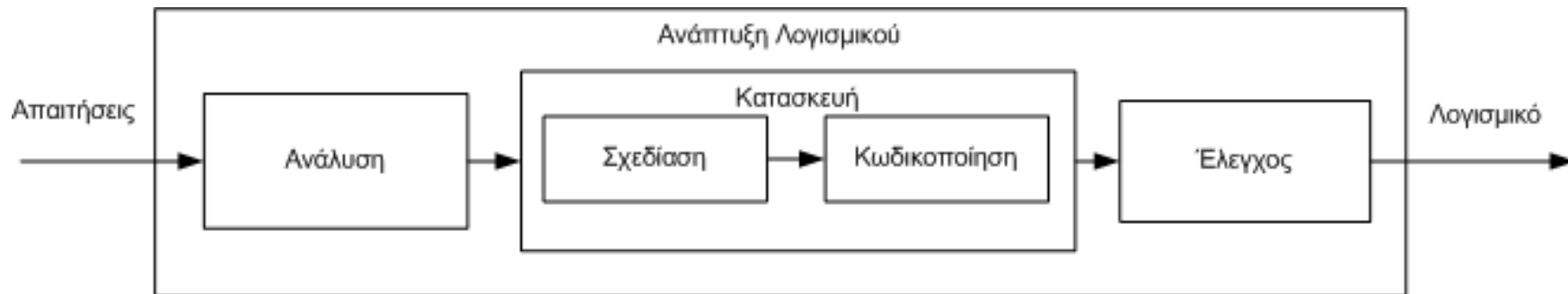
# ποιοτικό λογισμικό

---

- Διαθεσιμότητα
- Απόδοση
- Ευελιξία
- Ακεραιότητα
- Διαλειτουργικότητα
- Συντηρησιμότητα
- Μεταφερσιμότητα
- Αξιοπιστία
- Επαναχρησιμότητα
- Ευρωστία
- Ελεγχξιμότητα
- Ευχρηστία

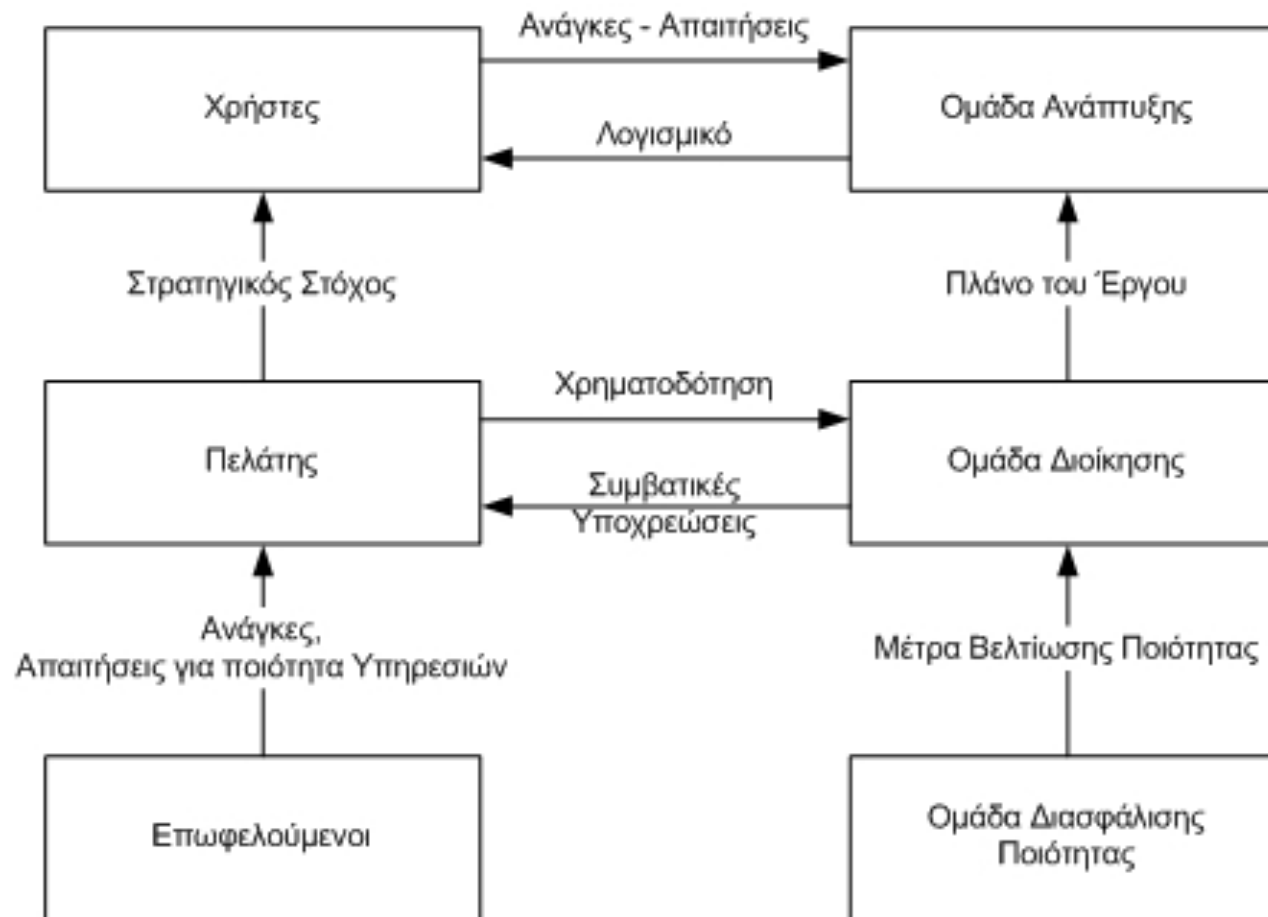
# ανάπτυξη λογισμικού

---



# εργασίες και αρμοδιότητες

---



# δραστηριότητες ανάπτυξης

---

- Προσδιορισμός απαιτήσεων
- Σχεδίαση
- Κατασκευή
- Έλεγχος
- Συντήρηση

# σχετικό κόστος διόρθωσης σφαλμάτων

---

	Δραστηριότητα Εντοπισμού και Διόρθωσης Σφάλματος				
<b>Δραστηριότητα Δημιουργίας Σφάλματος</b>	Απαιτήσεις	Αρχιτεκτονική	Κατασκευή	Έλεγχος Συστήματος	Συντήρηση
Απαιτήσεις	1	3	5-10	10	10-100
Αρχιτεκτονική	-	1	10	15	25-100
Κατασκευή	-	-	1	10	10-25

# διοίκηση έργου

---

- Ένα έργο ανάπτυξης δεν περιορίζεται σε καθαρά τεχνικά θέματα, αλλά και σε άλλα όπως
- ο χρονοπρογραμματισμός του έργου,
- η κατάρτιση του προϋπολογισμού του,
- η στελέχωση της ομάδας ανάπτυξης,
- η οργάνωση της ομάδας ανάπτυξης,
- η παρακολούθηση της πορείας του έργου,
- η διαχείριση των κινδύνων
- Όλες αυτές η δραστηριότητες αφορούν τη διοίκηση του έργου που είναι εξ ίσου σημαντικές όπως και οι βασικές.
- Ιδιαίτερα ο ρόλος του διοικητή έργου είναι κρίσιμος για την επιτυχή ολοκλήρωση των έργων.

# διοικητής έργου

---

Προσόντα του διοικητή έργου

- Επιχειρηματική αντίληψη
- Τεχνικές γνώσεις
- Γνώση διοίκησης έργων
- Ικανότητες λήψης αποφάσεων
- Ικανότητες παρακολούθησης διαδικασιών
- Ικανότητα ανάλυσης λεπτομερειών
- Ικανότητες οργάνωσης
- Ικανότητες επικοινωνίας
- Ηγετικές ικανότητες



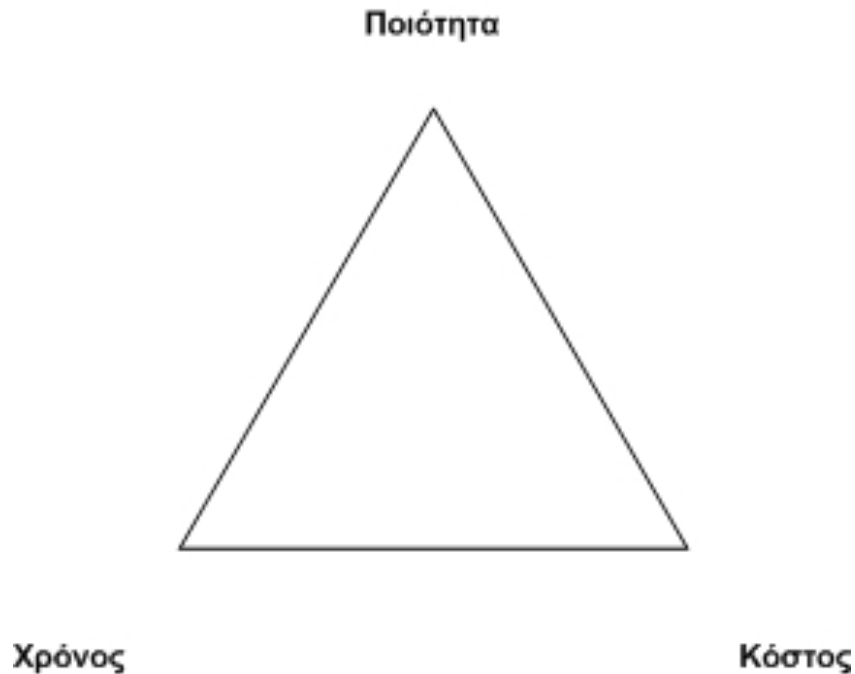
# διασφάλιση ποιότητας

---

- Σκοπός της διασφάλισης ποιότητας είναι να επιβεβαιώνει διαρκώς ότι κάθε παραδοτέο της ομάδας ανάπτυξης διαθέτει το επίπεδο ποιότητας που έχει ορίσει η διοίκηση του έργου και να εισηγείται στη Διοίκηση του έργου τυχόν αναγκαίες ενέργειες για την βελτίωση της ποιότητας των παραδοτέων της ομάδας ανάπτυξης.

# διασφάλιση ποιότητας

---



- Η επίτευξη υψηλής ποιότητας στο λογισμικό συνοδεύεται από μείωση του χρόνου ανάπτυξης.
- Ο χρόνος ανάπτυξης αυξάνεται όταν επιτυγχάνεται εξαιρετική ποιότητα σε ειδικές κατηγορίες του λογισμικού.

# διαχείριση διάταξης

---

Η διαχείριση διάταξης (configuration management) αφορά όλες τις δραστηριότητες που έχουν ως αποτέλεσμα, την αξιοποίηση όλων των ενδιάμεσων προϊόντων της διαδικασίας ανάπτυξης με σκοπό την παραγωγή εκδόσεων (version) του λογισμικού που ανταποκρίνονται στις απαιτήσεις κάποιας ομάδας χρηστών, τη διαχείριση διαφορετικών εκδόσεων του λογισμικού και την υποστήριξη της πολιτικής εκδόσεων. Η δραστηριότητα της διαχείρισης διάταξης αφορά τόσο την ομάδα Διοίκησης έργου όσο και την ομάδα ανάπτυξης

# μοντέλα διαδικασίας λογισμικού

---

- Κύκλος ζωής: ο χρόνος από την σύλληψη της ιδέας ανάπτυξης ενός προϊόντος μέχρι την απόσυρσή του
- Για να μπορέσουμε να επέμβουμε με σωστό τρόπο στις μεθοδολογίες και στις τεχνικές ανάπτυξης του λογισμικού πρέπει να έχουμε γνώση του τρόπου ανάπτυξης, λειτουργίας και απόσυρσης του λογισμικού, των χρονικών φάσεων που συνθέτουν τη ζωή του λογισμικού σ' ολόκληρο τον κύκλο ζωής του.
- Τα μοντέλα διαδικασίας λογισμικού μας παρέχουν μία γενική προσέγγιση για τον τρόπο ανάπτυξης του λογισμικού.

# μοντέλο διαδικασίας λογισμικού

---

- Μοντέλο διαδικασίας λογισμικού: μια απλοποιημένη περιγραφή μιας διαδικασίας ανάπτυξης λογισμικού που παρουσιάζεται από ορισμένη οπτική γωνία. Το μοντέλο διαδικασίας λογισμικού περιγράφει τον τρόπο οργάνωσης των δραστηριοτήτων ανάπτυξης (απαιτήσεις, σχεδίαση, κωδικοποίηση, έλεγχος) του λογισμικού.
- Διαδικασία ανάπτυξης (software process) : υιοθετεί ένα μοντέλο διαδικασίας λογισμικού και το εξειδικεύει περιγράφοντας μια ακολουθία βημάτων που έχουν ως αποτέλεσμα την ανάπτυξη λογισμικού υψηλής ποιότητας στο προβλεπόμενο χρόνο και στο προϋπολογισμένο κόστος.

# Το μοντέλο του καταρράκτη

---



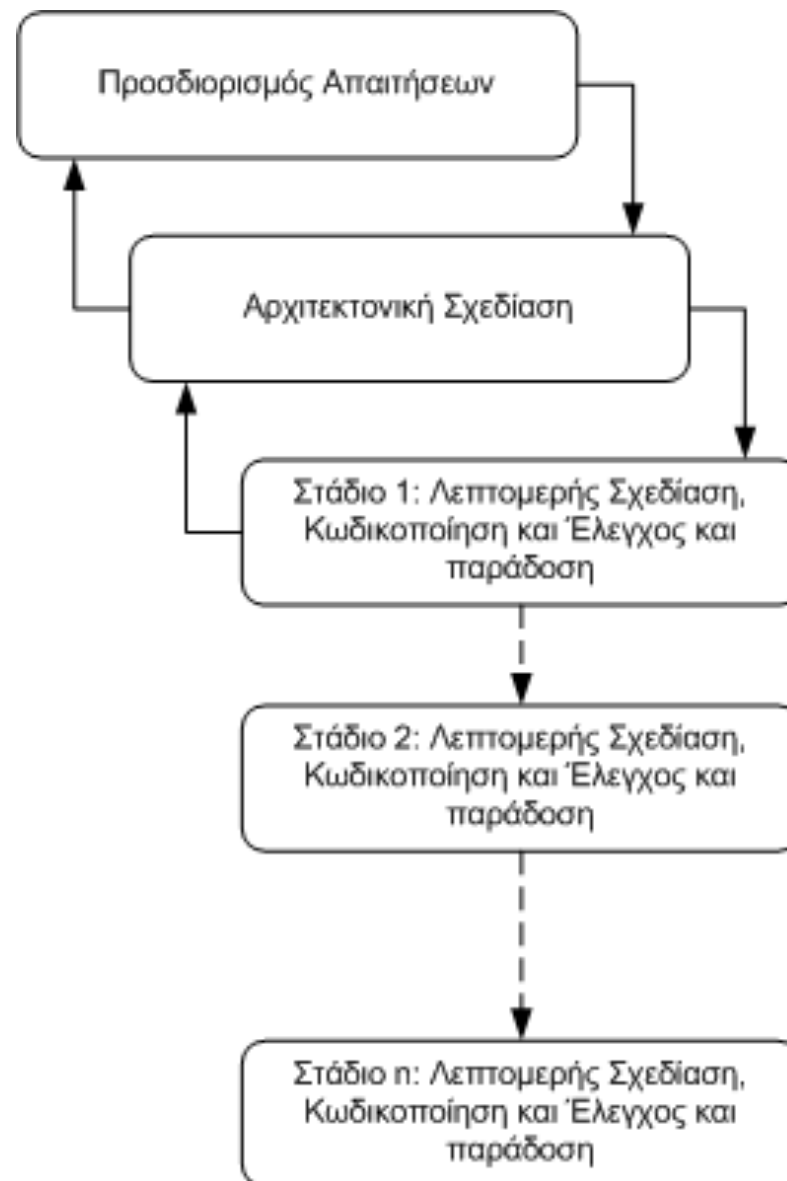
# Το μοντέλο του καταρράκτη

---

- Ακολουθιακή οργάνωση των δραστηριοτήτων ανάπτυξης.
- Αξιολόγηση
  - + «Εύκολος» χρονικός προγραμματισμός και «απλή» διοίκηση έργου.
  - Προϋποθέτει σταθερότητα στις απαιτήσεις και την αρχιτεκτονική.
  - Μικρή συμμετοχή τελικών χρηστών.
  - Παραπλανητική εικόνα της πορείας του έργου.
  - Δεν ευνοεί την ενεργητική διαχείριση κινδύνων.

# το επαυξητικό μοντέλο

---



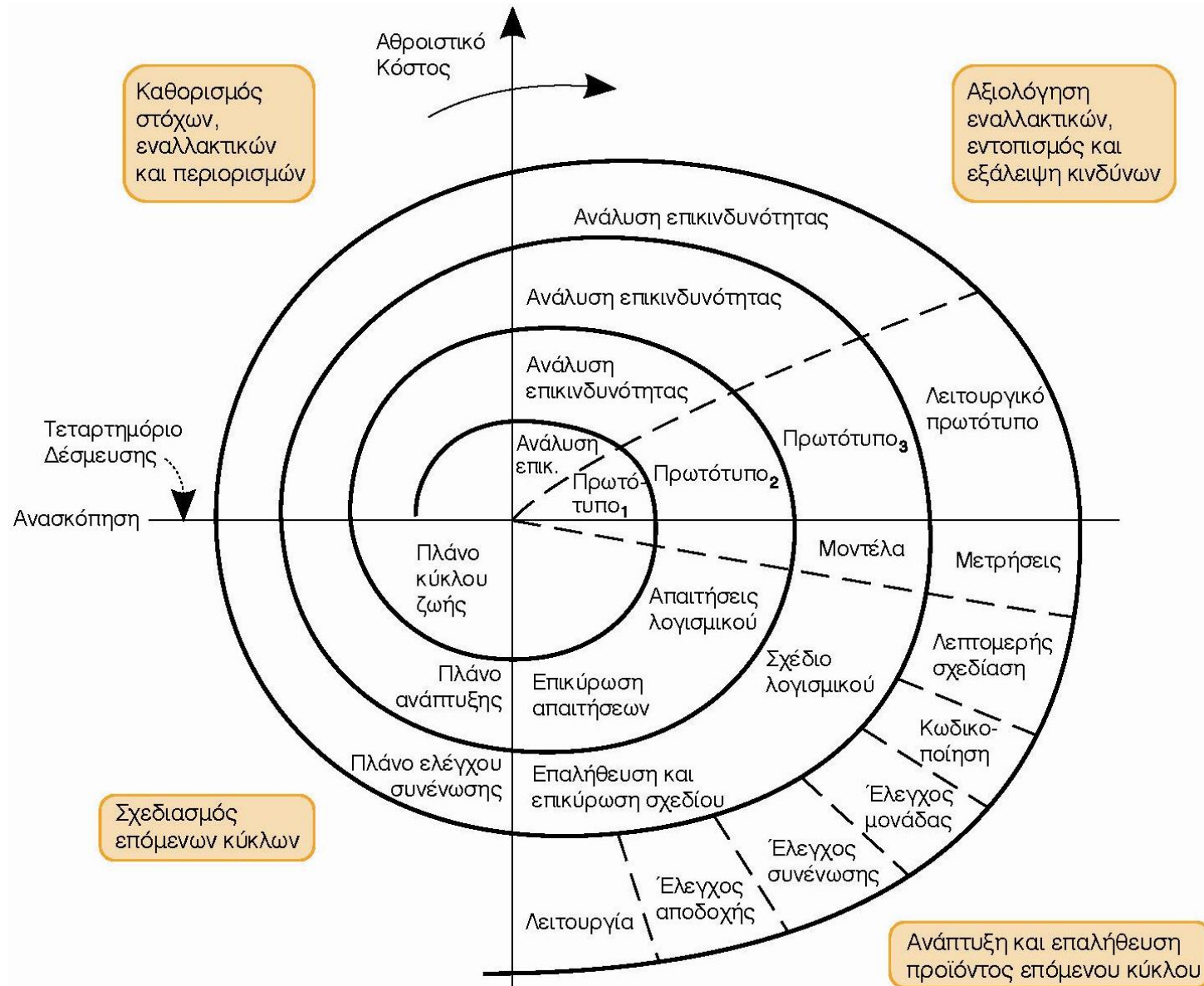


# το επαυξητικό μοντέλο

---

- Η λεπτομερής σχεδίαση και η κωδικοποίηση γίνεται σταδιακά με παραγωγή νέων εκδόσεων του λογισμικού
- Αξιολόγηση
  - + Μεγαλύτερη ευελιξία σε σχέση με το μοντέλο του καταρράκτη
  - + Βελτιωμένη διαχείριση κινδύνων και καλύτερη αξιολόγηση της πορείας του έργου
  - Προϋποθέτει σταθερότητα των απαιτήσεων και της αρχιτεκτονικής του λογισμικού

# το σπειροειδές Μοντέλο



# το σπειροειδές Μοντέλο

---

- Ανάπτυξη σε κύκλους.
- Σε κάθε κύκλο πραγματοποιούνται πολλές δραστηριότητες ανάπτυξης (απαιτήσεις, σχεδίαση, κωδικοποίηση).
- Οι αποφάσεις για τον προγραμματισμό κάθε κύκλου καθορίζονται από τους κινδύνους του έργου.
- Δημιουργία πρωτοτύπων (prototypes) ως μέσο μείωσης των κινδύνων του έργου.

# το σπειροειδές Μοντέλο

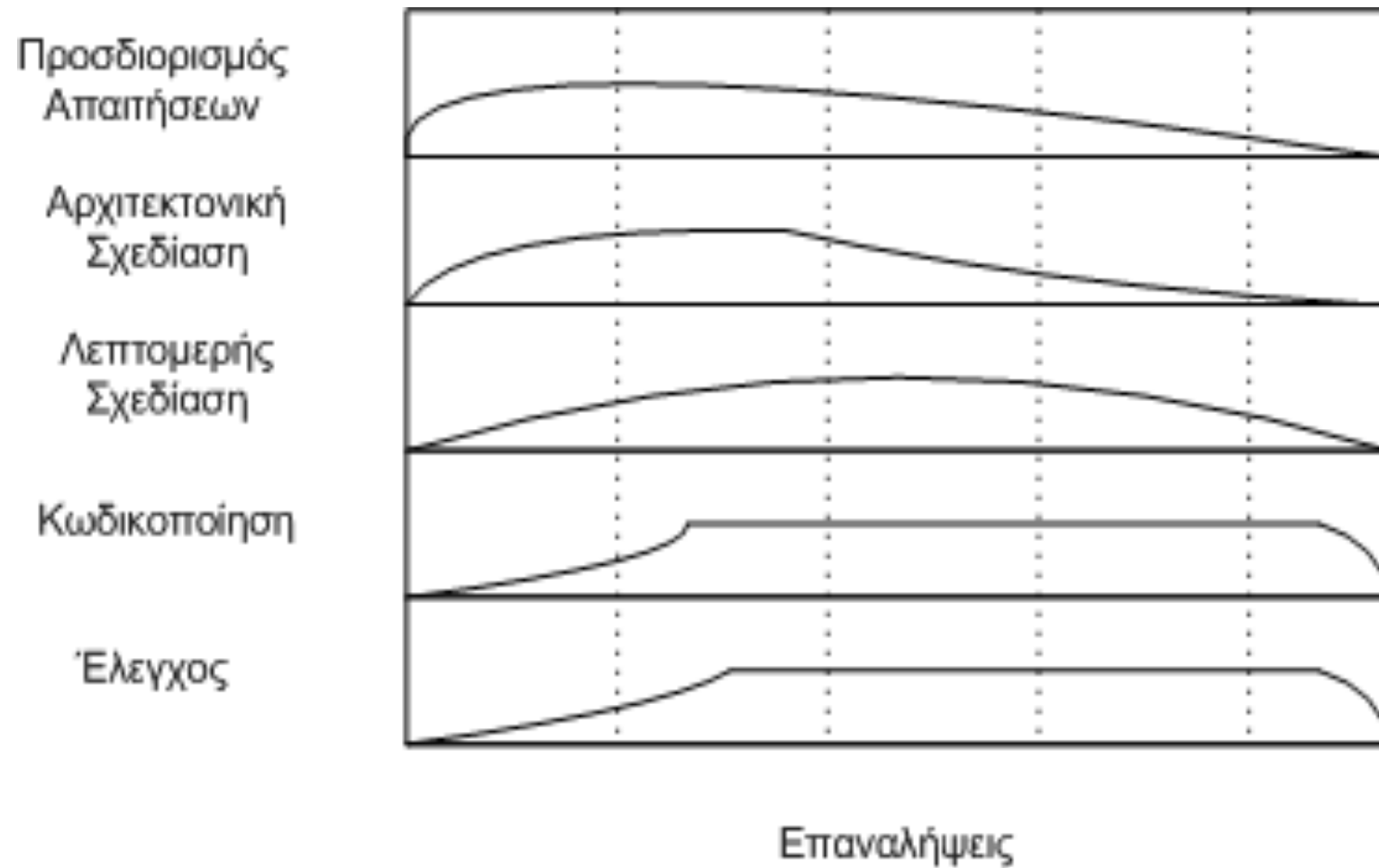
---

## Αξιολόγηση

- + Ενεργητική αντιμετώπιση κινδύνων.
- + Προσαρμογή σε μεταβαλλόμενες απαιτήσεις.
- + Καλύτερη συμμετοχή των χρηστών.
- + Βελτιωμένη αξιολόγηση της πορείας του έργου.
- Απαιτητική διοίκηση έργου.

# το επαναληπτικό μοντέλο

---



# το επαναληπτικό μοντέλο

---

- Είναι εξορισμού και επαυξητικό (iterative and incremental).
- Η ανάπτυξη του λογισμικού δεν οργανώνεται σε σειριακές φάσεις αλλά σε επαναλήψεις.
- Σε κάθε επανάληψη πραγματοποιούνται όλες (ή σχεδόν όλες) οι δραστηριότητες ανάπτυξης, με διαφορετική όμως βαρύτητα στην πορεία του χρόνου.
- Κάθε επανάληψη παράγει μία ημιτελή αλλά ελεγμένη έκδοση του λογισμικού.
- Το επαναληπτικό μοντέλο αξιοποιεί και τη χρονική πλαισίωση (timeboxing) των επαναλήψεων. Δεν επιτρέπεται η μετάθεση της ημερομηνίας λήξης μίας επανάληψης.

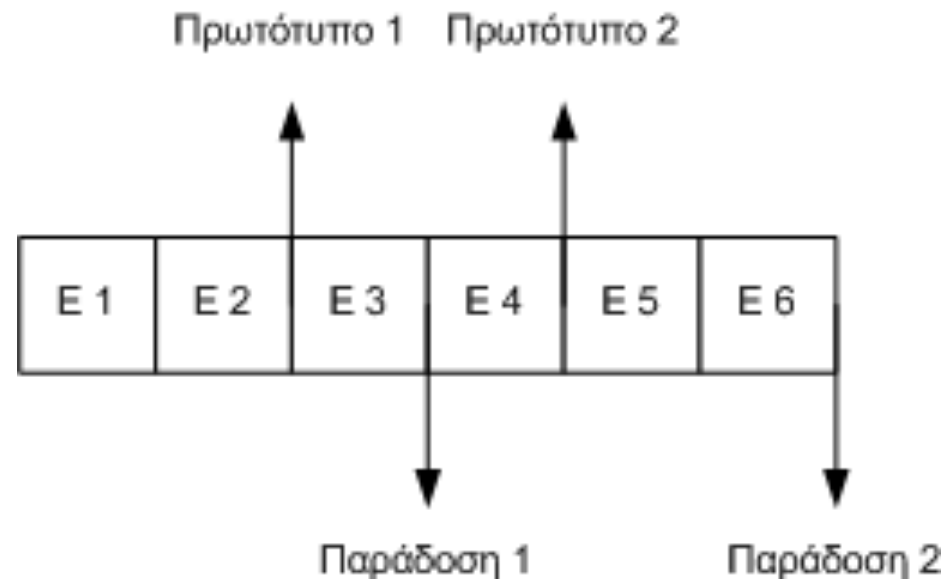
# το επαναληπτικό μοντέλο

---

- Αξιολόγηση
  - + Άριστη προσαρμογή σε μεταβαλλόμενες απαιτήσεις
  - + Δυνατότητες για ενεργητική αντιμετώπιση των κινδύνων του έργου
  - + Ενθάρρυνση της ενεργούς και διαρκούς συμμετοχής των χρηστών
  - + Άριστη παρακολούθηση της πορείας του έργου με τον καθορισμό βραχυχρόνιων στόχων
  - «Παρεξηγήσιμο» μοντέλο. Αυτοί που το ακολουθούν είναι λιγότεροι από αυτούς που το επικαλούνται
- Το επαναληπτικό μοντέλο είναι πλέον το κυρίαρχο μοντέλο του κύκλου ζωής του λογισμικού.
- Παρέχει μεγάλη ευελιξία στον τρόπο που παραδίδεται το λογισμικό

# το επαναληπτικό μοντέλο και η πρωτοτυποποίηση

---



- Εξελικτική πρωτοτυποποίηση (evolutionary prototyping). Με την ολοκλήρωση κάποιων επαναλήψεων παραδίδονται πρωτότυπα στον πελάτη για αξιολόγηση και ανατροφοδότηση
- Εξελικτική παράδοση (evolutionary delivery). Με την ολοκλήρωση κάποιων επαναλήψεων παραδίδονται στον πελάτη εκδόσεις του λογισμικού προς χρήση.