



Πληροφοριακά Συστήματα Διοίκησης

Ανάπτυξη και προμήθεια συστημάτων

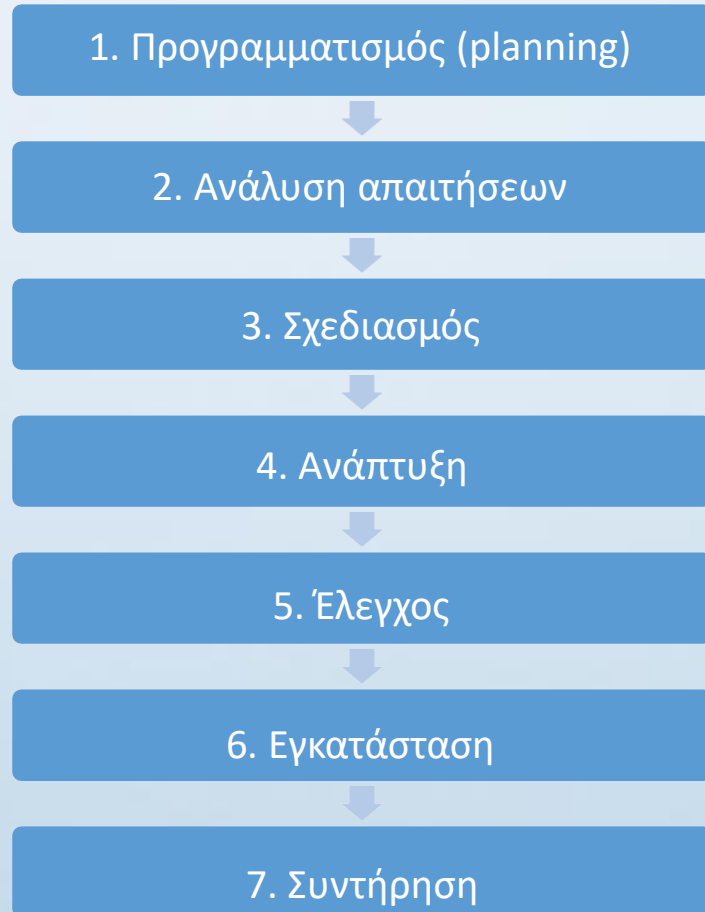
ΚΥΡΙΟΙ ΕΙΔΗ ΠΣ ΣΕ ΕΠΙΧΕΙΡΗΣΕΙΣ



Τρόποι υλοποίησης

- Εσωτερική Ανάπτυξη Συστήματος & outsourcing
- Αγορά έτοιμου λογισμικού
- Χρήση Cloud

Κύκλος ζωής ανάπτυξης συστημάτων



Διαδικασία των 7 φάσεων της ζωής ενός ΠΣ. Η διαδικασία μπορεί να είναι ανατροφοδοτούμενη

1. Προγραμματισμός (planning)

- Αποτελείται από 2 επί μέρους φάσεις και ανατίθεται στην **συντονιστική επιτροπή** (steering committee):
 - Τον καθορισμό της επιχειρησιακής ανάγκης (Ποια προβλήματα θα ξεπερνιούνται με την χρήση των συστημάτων;)
 - Την μελέτη σκοπιμότητας (το σύστημα είναι απαραίτητο ή όχι;)
 - Την αξιολόγηση της σημασίας που έχει για τον οργανισμό (ποια προβλήματα λύνει; Ποια η επιπλέον αξία που προσδίδει στον οργανισμό;)
- Κατά την παρούσα φάση θα πρέπει να αξιολογούνται:
 - η απόδοση της επένδυσης (ROI=Return Of Investment)
 - το ανταγωνιστικό πλεονέκτημα που ενδεχόμενα θα δημιουργήσει
 - θέματα διαχείρισης κινδύνου που προκύπτουν (π.χ. ζητήματα ασφάλειας, απορρήτου, συμμόρφωσης με κανονισμούς)

2. Ανάλυση λειτουργικών και μη λειτουργικών απαιτήσεων

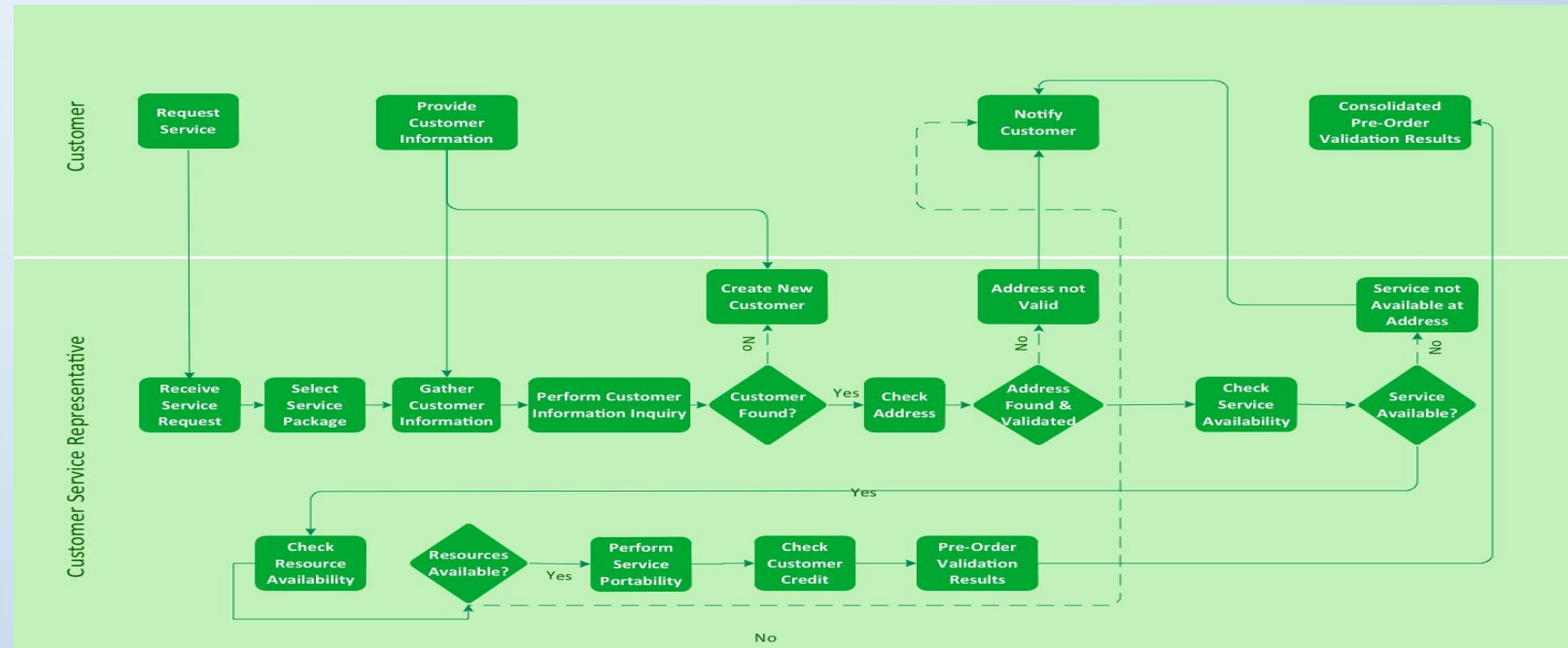
- Στη φάση αυτή δημιουργείται μια λίστα λειτουργικών και μη αποφάσεων που το σύστημα θα πρέπει να πληρεί (έγγραφο καθορισμού απαιτήσεων).
- Η λίστα καταρτίζεται μέσω συνεντεύξεων, συσκέψεων, ερωτηματολογίων, αξιολογήσεων υφιστάμενων διαδικασιών
- Η λίστα των απαιτήσεων ταξινομείται σε:
 - Υποχρεωτικά χαρακτηριστικά
 - Προτιμώμενα χαρακτηριστικά
 - Μη ουσιώδη χαρακτηριστικά

Διαγράμματα ροής διαδικασιών

Βασικός στόχος της λειτουργίας των ΠΣ:

Ανασχεδιασμός επιχειρησιακών διαδικασιών (BPR=Business Process Reengineering): Σχεδιασμός και ανάλυση της ροής εργασιών ενός οργανισμού με στόχο την εξάλειψη διαδικασιών που δεν προσθέτουν κανενός είδους αξία

- Η αποτύπωση των διαδικασιών που θα υποστηρίζει το σύστημα γίνεται μέσω των διαγραμμάτων ροής διαδικασιών (process diagrams)
- Τα διαγράμματα ροής διαδικασιών είναι γραφικές παραστάσεις που περιγράφουν κατανοητά κάθε διαδικασία την οποία θα υποστηρίζει το νέο Πληροφοριακό σύστημα



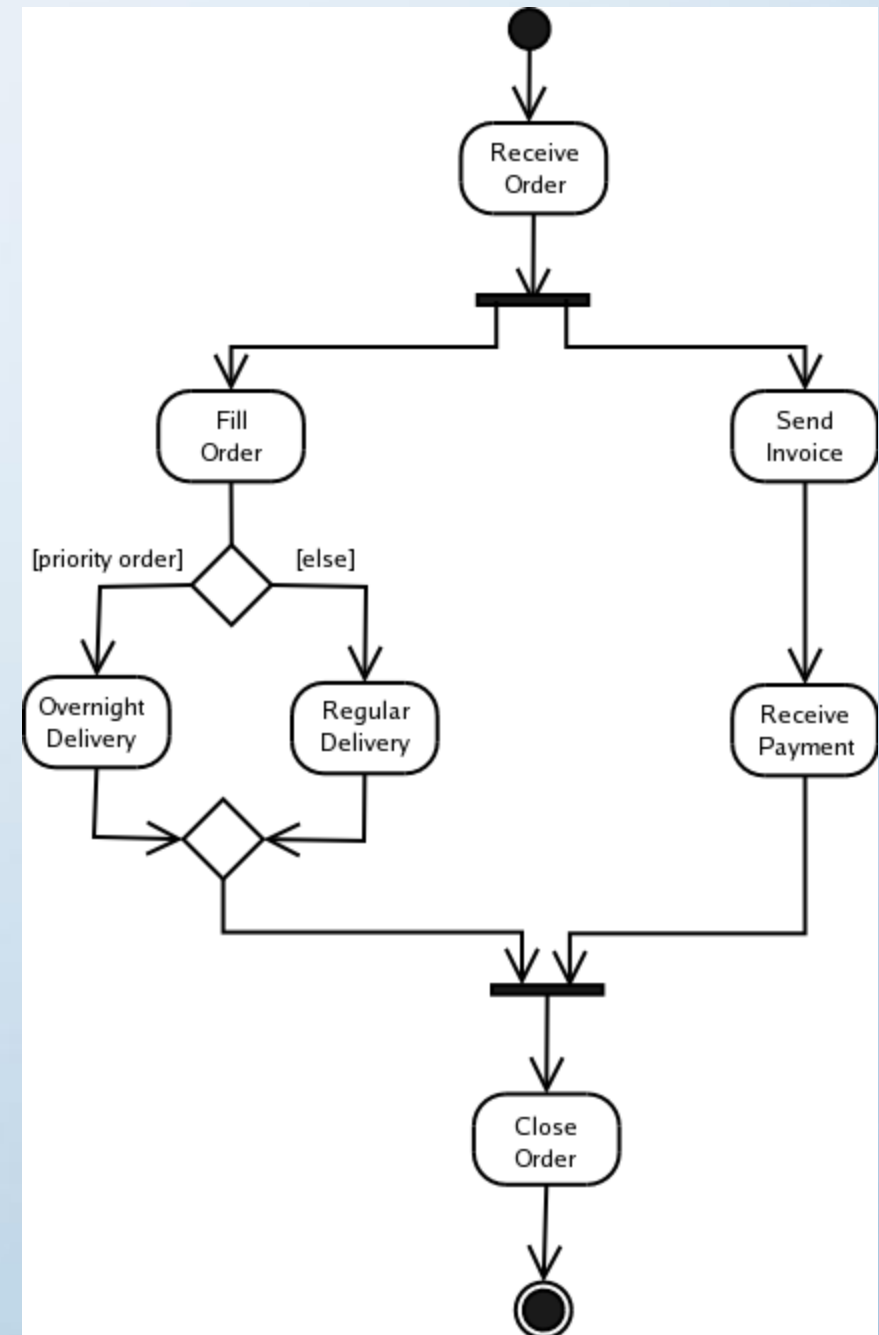
<http://www.conceptdraw.com/solution-park/business-process-cross-functional-flowcharts>

Έγγραφο ορισμού απαιτήσεων (RDD=Requirements Definition Document)

- Επίσημο έγγραφο καθορισμού απαιτήσεων. **Προσδιορίζει με κάθε λεπτομέρεια τα λειτουργικά χαρακτηριστικά** θα πρέπει να έχει το σύστημα
- Τα ενδιαφερόμενα μέρη **συνυπογράφουν** το κείμενο, επιβεβαιώνοντας ότι πράγματι αυτό είναι το σύστημα που χρειάζονται (περίπτωση outsourcing)
- Το συγκεκριμένο έγγραφο χρησιμοποιείται στην αναζήτηση εμπορικών διαθέσιμων ΠΣ τα οποία πιθανώς καλύπτουν τις απαιτήσεις
- Αν μεγάλο ποσοστό των απαιτήσεων καλύπτονται (>80%) και η τιμή είναι λογική τότε προτιμάται η αγορά του έτοιμου λογισμικού

3. Σχεδιασμός (design)

- Στην τρίτη φάση ζωής ενός ΠΣ πραγματοποιείται η μετατροπή του εγγράφου ορισμού απαιτήσεων σε ένα **εκτελέσιμο τεχνικό σχέδιο**
- Καθορίζει την αρχιτεκτονική του συστήματος
- Η διόρθωση λαθών στη φάση της σχεδίασης βοηθά στην αποφυγή κοστοβόρων διορθωτικών κινήσεων στο μέλλον
- Πραγματοποιείται, συνήθως, μέσω της γλώσσας UML (Unified Markup Language) που αποτελεί μια τυποποιημένη προσέγγιση στη μοντελοποίηση συστημάτων



5. Έλεγχος (testing)


- Πραγματοποιούνται καθ' όλη τη διάρκεια της φάσης ανάπτυξης, κάθε φορά που ολοκληρώνονται τα υποσυστήματα
- Όταν το σύστημα ολοκληρωθεί τότε υποβάλλεται σε ενδεδειγμένο έλεγχο
- Καθένας έλεγχος μιμείται γεγονότα που θα κληθεί να χειριστεί το σύστημα όταν τεθεί σε πλήρη λειτουργία

Τύπος ελέγχου	Περιγραφή
Έλεγχος μονάδας (unit testing)	Έλεγχος υπομονάδων που γίνεται στη φάση της ανάπτυξης
Έλεγχος συστήματος (system testing)	Έλεγχος συστήματος από προγραμματιστές και χρήστες
Έλεγχος ακραίων καταστάσεων (stress testing)	Έλεγχος απόκρισης του συστήματος σε καταστάσεις φόρτου
Έλεγχος παράλληλης λειτουργίας (parallel testing)	Σύγκριση της εξόδου του συστήματος με το σύστημα που αντικαθιστά
Έλεγχος ολοκλήρωσης (integration testing)	Έλεγχος ορθής λειτουργίας συστήματος με άλλα συστήματα με τα οποία αλληλοεπιδρά
Έλεγχος αποδοχής (acceptance testing)	Τελικός έλεγχος συμφωνίας με τις προδιαγραφές

6. Εγκατάσταση (deployment)

- Γίνεται εγκατάσταση του λογισμικού που μπαίνει σε κανονική λειτουργία όπου και συνεχίζεται ο έλεγχος του
- Τρόποι εγκατάσταση:
 - **Παράλληλη εγκατάσταση** (το παλιό σύστημα βρίσκεται σε λειτουργία ενώ το νέο σύστημα ξεκινά να λειτουργεί και διενεργούνται έλεγχοι ορθής λειτουργίας)
 - **Σταδιακή εγκατάσταση** (τα επιμέρους υποσυστήματα τίθενται σε λειτουργία σε φάσεις και όχι ταυτόχρονα, π.χ. ERP)
 - **Άμεση εγκατάσταση** (το παλιό σύστημα τίθεται εκτός λειτουργίας και αντικαθίσταται από το νέο σε μια συγκεκριμένη ημερομηνία)

7. Συντήρηση

- Φάση διαρκείας μέχρι την απεγκατάσταση του συστήματος.
 - Περιλαμβάνει τις απαραίτητες εργασίες έτσι ώστε να υποστηριχθούν οι συνεχώς μεταβαλλόμενες επιχειρησιακές απαιτήσεις (π.χ. αλλαγή νομοθετικού πλαισίου)
 - Με το πέρασμα του χρόνου, το βάρος της συντήρησης μπορεί να γίνει πολύ μεγάλο
- 
- Συστήματα παλαιού τύπου (legacy systems): Συστήματα τα οποία στηρίζονται σε απαρχαιωμένες αρχιτεκτονικές τα οποία συνεχίζουν να χρησιμοποιούνται λόγω του ότι είτε λειτουργούν ικανοποιητικά είτε η αντικατάστασή τους είναι εξαιρετικά ακριβή

Στρατηγικές ανάπτυξης και σχεδιασμού συστημάτων

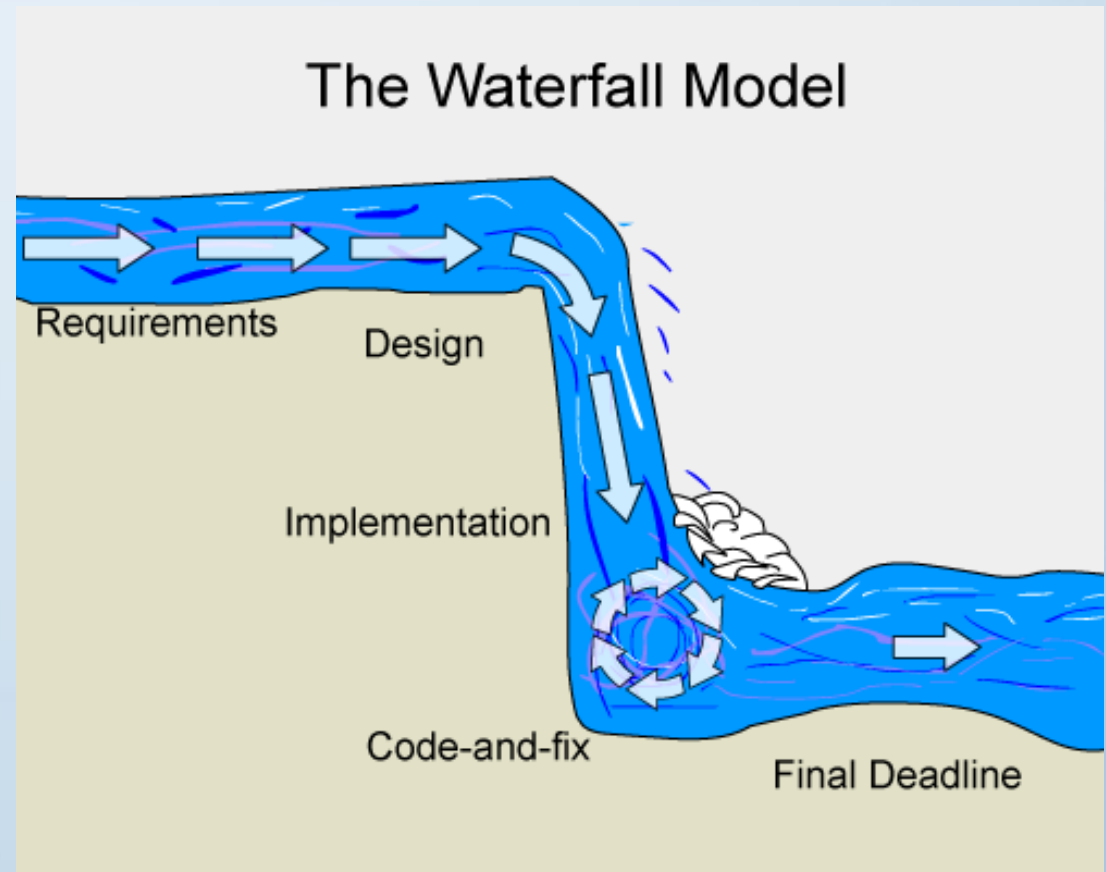
Η μέθοδος του
καταρράκτη

Επαναληπτικές
μέθοδοι

Ευέλικτες
μέθοδοι

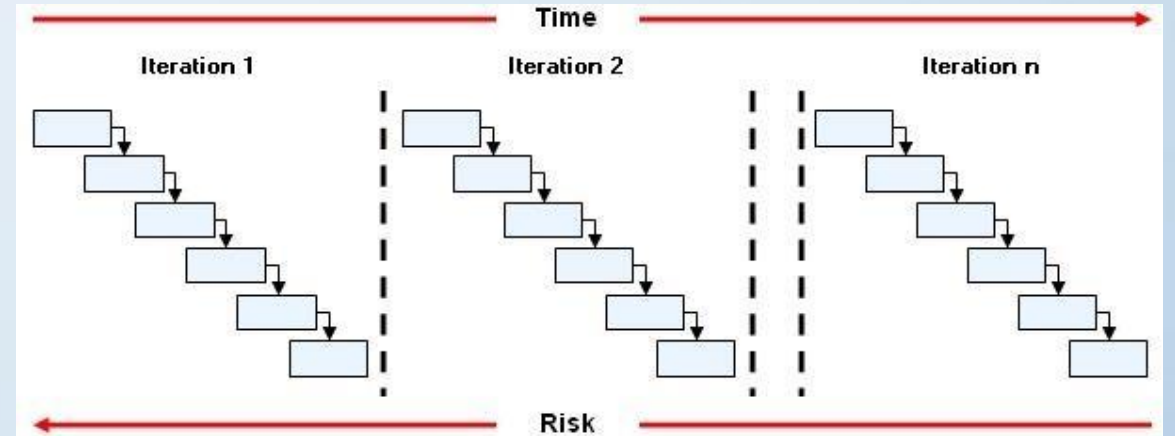
Η μέθοδος του καταρράκτη (waterfall method)

- Στη μέθοδο του καταρράκτη οι φάσεις του κύκλου ζωής ανάπτυξης συστημάτων λαμβάνουν χώρα διαδοχικά, με την κάθε εργασία να ξεκινά μόνο εφόσον έχει ολοκληρωθεί η προηγούμενη
- Στη φάση ανάλυσης καθορίζονται οι απαιτήσεις και υπολογίζεται ο χρόνος και οι πόροι που θα απαιτηθούν από κάθε επόμενη φάση
- Οι προγραμματιστές δεν ξεκινούν να γράφουν κώδικα παρά μόνο όταν όλες οι προηγούμενες φάσεις της ανάπτυξης έχουν ολοκληρωθεί
- Η μέθοδος του καταρράκτη συχνά αποτυγχάνει καθώς οι απαιτήσεις μπορεί να μεταβληθούν πριν ολοκληρωθεί το έργο



Επαναληπτικές μέθοδοι (iterative methods)

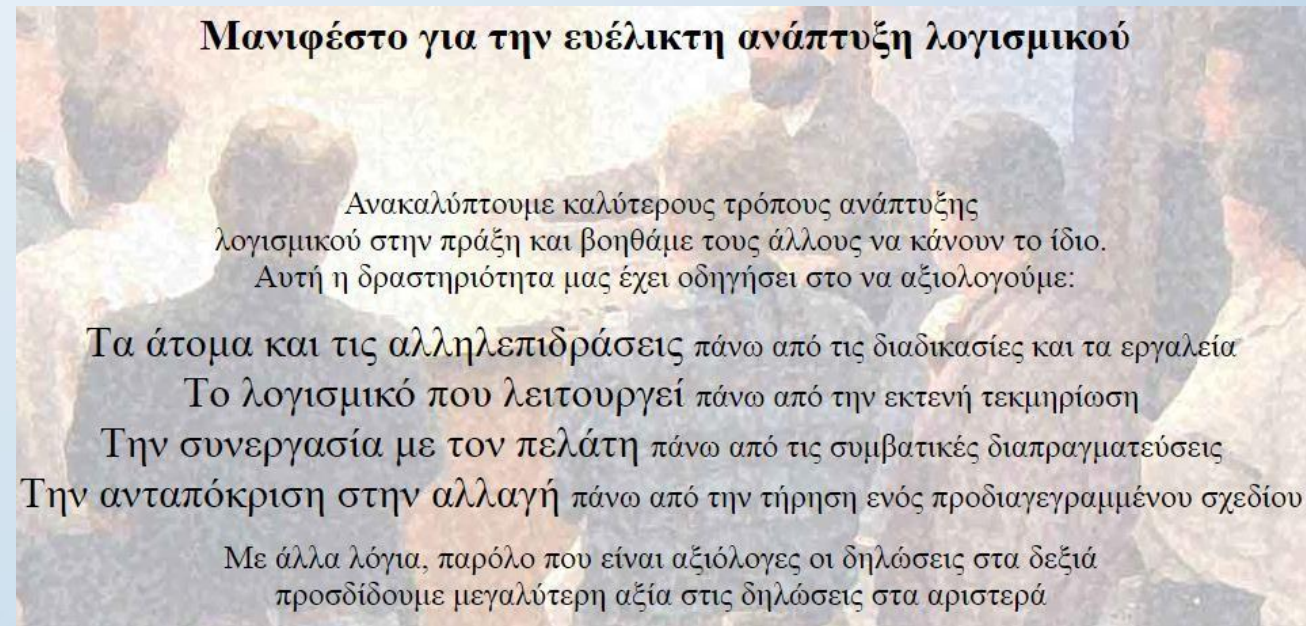
- Οι επαναληπτικές μέθοδοι είναι στρατηγικές που συμπιέζουν τον χρονικό ορίζοντα ανάπτυξης του λογισμικού έτσι ώστε να μειωθεί η επίδραση των μεταβαλλόμενων επιχειρησιακών αλλαγών
- Εστιάζουν στο διαθέσιμο χρόνο μέχρι την επόμενη έκδοση
- Η ομάδα ανάπτυξης καθορίζει ποιες από τις απαιτήσεις θα μπορούν να παραδοθούν στο διαθέσιμο χρόνο



Ταχεία ανάπτυξη εφαρμογής (RAD=Rapid Application Development): Οι προγραμματιστές δημιουργούν ένα πρωτότυπο λογισμικό το οποίο μοιράζονται με τους χρήστες έτσι ώστε να λάβουν διορθώσεις και ανατροφοδότηση

Ευέλικτες μέθοδοι (agile software development)

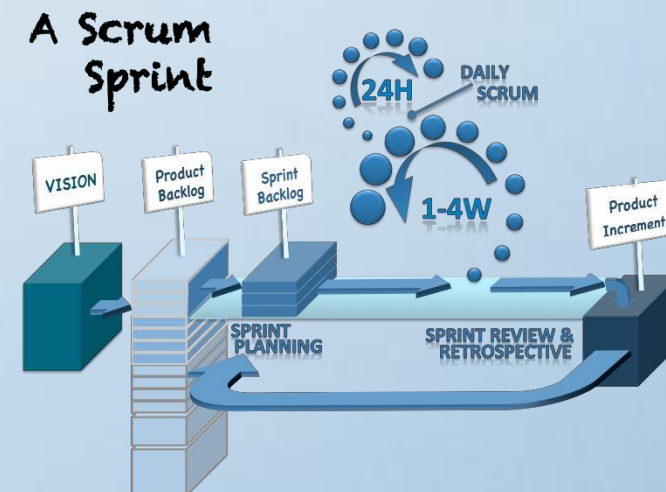
- Η ομάδα ανάπτυξης είναι κατά κανόνα πολύ συνεκτική και τα μέλη της συσσεγάζονται
- Στην ομάδα ανάπτυξης συμμετέχουν με αποκλειστική δέσμευση χρόνου ένας ή περισσότεροι χρήστες
- Πολλές εργασίες εκτελούνται ταυτόχρονα έτσι ώστε να επιταχυνθεί η παράδοση ενός πλήρως λειτουργικού λογισμικού
- Η διάρκεια ανάπτυξης (time box) είναι πολύ μικρή (2-6 εβδομάδες)
- Οι συχνότερα χρησιμοποιούμενες ευέλικτες μέθοδοι είναι η Scrum και ο ακραίος προγραμματισμός (XP)



<http://agilemanifesto.org/iso/el/>

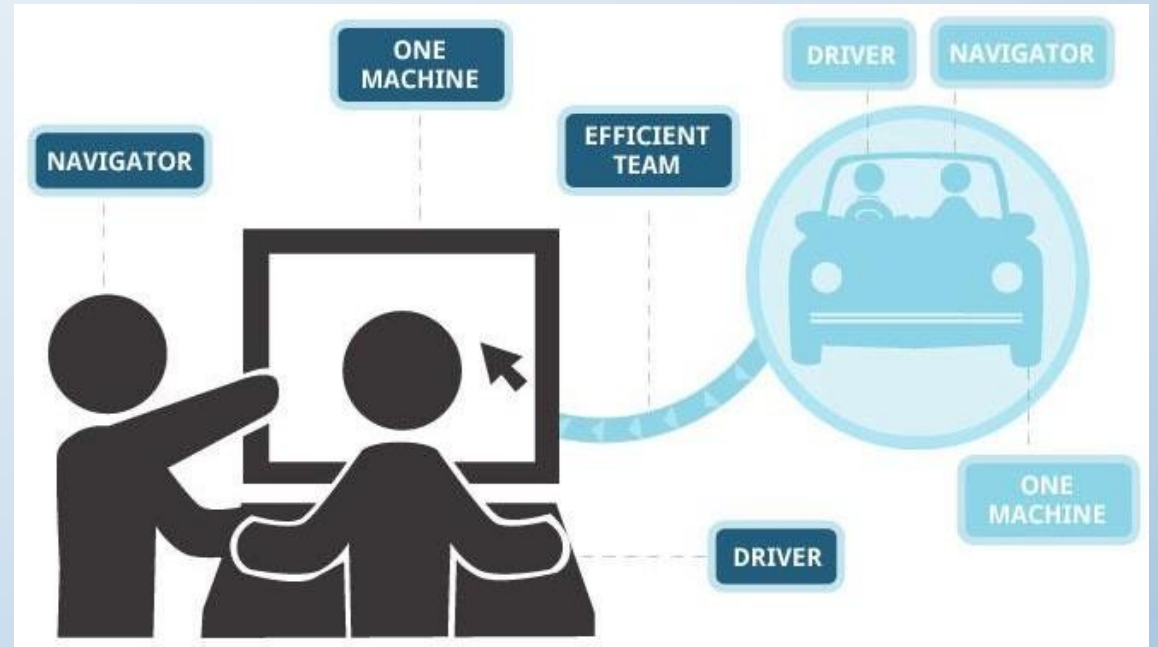
Η μέθοδος Scrum

- Ισχυρά συνεκτικές ομάδες με συχνές συσκέψεις μικρής διάρκειας
- Ρόλοι στην ομάδα:
 - **Ιδιοκτήτης προϊόντος:** χρήστης που συμμετέχει στην ομάδα και διασφαλίζει ότι το έργο προσδίδει αξία στην επιχείρηση
 - **Αρχηγός (master) του Scrum:** διαχειριστής του έργου
- Κάθε ομάδα εκτελεί ένα επαναληπτικό κύκλο ανάπτυξης διάρκειας 1-4 εβδομάδων (sprint)
- Στην αρχή κάθε επαναληπτικού κύκλου ο ιδιοκτήτης περιγράφει και ιεραρχεί τις ανεκτέλεστες απαιτήσεις (backlog)
- Η ομάδα ξεκινά άμεσα εργασία με στόχο στο τέλος κάθε επαναληπτικού κύκλου να παραδοθεί λογισμικό με νέα λειτουργικά χαρακτηριστικά



Ακραίος προγραμματισμός (eXtreme Programming-XP)

- Ο XP βασίζεται σε ομάδες και χαρακτηρίζεται από τη συχνή έκδοση εφαρμόσιμου λογισμικού, τα στενά χρονικά πλαίσια ανάπτυξης, τους προγραμματιστές που εργάζονται σε ζεύγη και την έμφαση στη διενέργεια ελέγχων
- Το έργο ξεκινά με ιστορίες χρηστών (user stories) που συχνά γράφονται σε κάρτες 3x5 εκ.)
- Οι προγραμματιστές εργάζονται σε ζεύγη (pair programming) αξιολογώντας ο ένας την εργασία του άλλου, είναι του ίδιου επιπέδου και μοιράζονται τον ίδιο υπολογιστή



pair programming

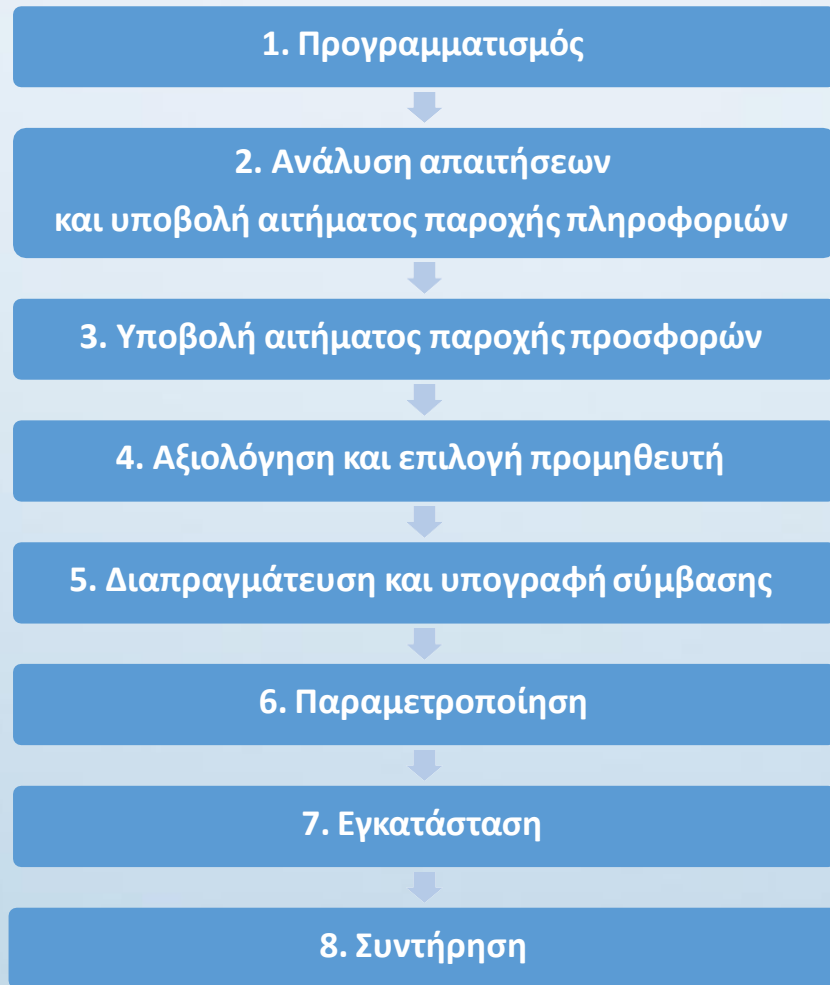
Σύγκριση προσεγγίσεων ανάπτυξης λογισμικού

- Όλες οι μέθοδοι μπορούν να επιφέρουν καλά ή κακά αποτελέσματα ανάλογα με την περίπτωση και τους ανθρώπους που καλούνται να τις εφαρμόσουν
- Η επιλογή μεθόδου εξαρτάται από τη σαφήνεια των απαιτήσεων
- Μεγάλο ποσοστό των έργων λογισμικού χρησιμοποιούν παρά τα μειονεκτήματα που έχει τη μέθοδο καταρράκτη
- Στη μέθοδο καταρράκτη:
 - Οι προγραμματιστές μπορεί να θεωρούν την εργασία τους επιτυχημένη ακόμα και αν το έργο αποτύχει
 - Οι προγραμματιστές τείνουν να αντιστέκονται σε αλλαγές ή προσθήκες στις απαιτήσεις μετά τη φάση ανάλυσης
- Στις ευέλικτες ομάδες:
 - Οι προγραμματιστές πρέπει να δέχονται ευχάριστα τις αλλαγές διότι κατανοούν ότι ο τελικός σκοπός είναι να δημιουργηθεί λογισμικό που θα θέλει πραγματικά ο πελάτης
 - Η ευθύνη για την παράδοση ενός έργου είναι συλλογική

Τρόποι υλοποίησης

- Εσωτερική Ανάπτυξη Συστήματος & outsourcing
- Αγορά έτοιμου λογισμικού
- Χρήση Cloud

Διαδικασία προμήθειας Πληροφοριακού Συστήματος



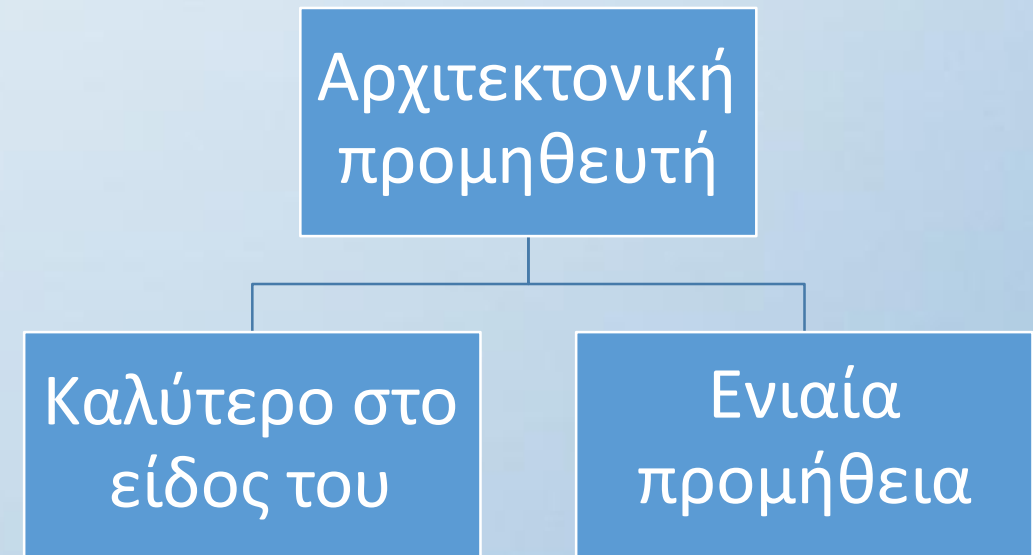
Αίτημα παροχής πληροφοριών (RFI=Request For Information): Σύνταξη αρχικού καταλόγου απαιτήσεων και αποστολή σε μεγάλο αριθμό υποψήφιων προμηθευτών

Αίτημα παροχής προσφορών (RFP=Request For Proposal): Πρόσκληση στην οποία περιλαμβάνονται οι λεπτομερείς απαιτήσεις προς επιλεγμένες εταιρείες λογισμικού έτσι ώστε να υποβάλλουν επίσημη πρόταση με περιγραφή υπηρεσιών και κόστους

Αξιολόγηση διαθέσιμων προσφορών

- Η συντονιστική επιτροπή
 1. μελετά επιδείξεις λογισμικού, επισκέπτεται εταιρείες στις οποίες χρησιμοποιείται και προσπαθεί να περιορίσει τις επιλογές στις πλέον αξιόλογες
 2. καταστρώνει στρατηγική αξιολόγησης με ιεράρχηση κριτηρίων

Βαρύτητα (1=πολύ κακή έως 5=πολύ καλή)	30% Συνάφεια με απαιτήσεις	20% Εμπειρία προμηθευτή	10% Συστάσεις προμηθευτή	20% Αρχιτεκτονική προμηθευτή	20% Κόστος	100% Σύνολο
Προμηθευτής Α'	5	3	2	1	1	2,7
...	2,5

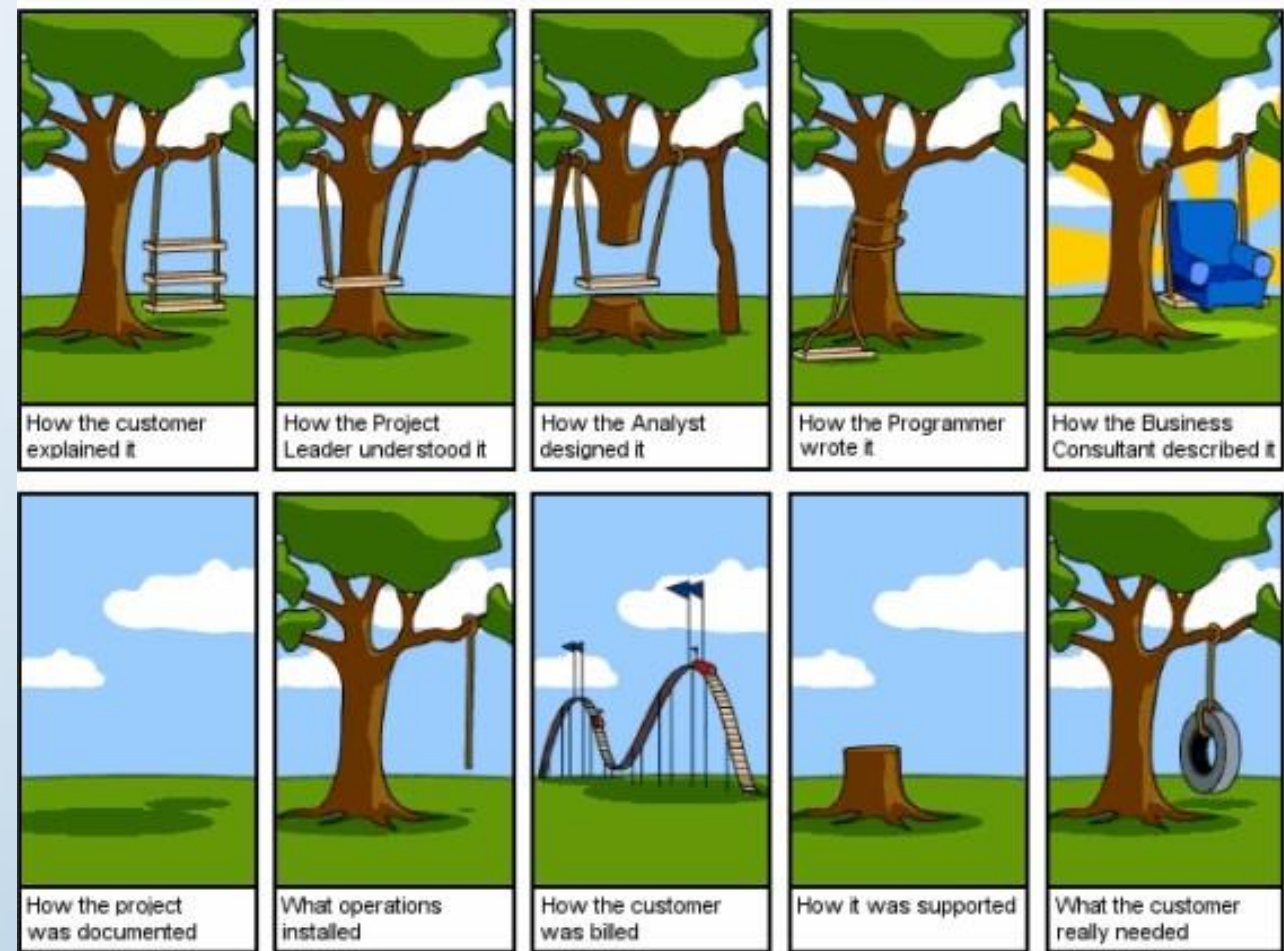


Προσαρμογή και παραμετροποίηση

- Αν το έτοιμο λογισμικό δεν καλύπτει όλες τις απαιτήσεις η επιχείρηση μπορεί είτε να προσαρμόσει τις διαδικασίες της έτσι ώστε να ταιριάζουν με όσα υποστηρίζει το λογισμικό είτε να ζητήσουν από τον προμηθευτή παραμετροποίηση του λογισμικού
- Ο παραμετροποιημένος κώδικας μπορεί να δίνει ανταγωνιστικό πλεονέκτημα αλλά μπορεί να δημιουργήσει προβλήματα αυξημένου κόστους, σφαλμάτων, προβλημάτων υποστήριξης κ.α.
- Οι περισσότερες επιχειρήσεις προσπαθούν να κρατούν την παραμετροποίηση στο ελάχιστο δυνατό επίπεδο



Ο ανθρώπινος παράγοντας στην ανάπτυξη και προμήθεια του λογισμικού

- Μπορεί να προκύψουν χάσματα στην επικοινωνία στις ομάδες που δημιουργούνται για την ανάπτυξη ή την προμήθεια λογισμικού λόγω του διαφορετικού υποβάθρου των μελών τους
- Τα χάσματα επικοινωνίας θα πρέπει να γεφυρωθούν έτσι ώστε να υπάρξει επιτυχία στο έργο
- Η καλή γνώση ενός μέρους της διαδικασίας δεν αρκεί για τη συνολική σχεδίαση της διαδικασίας από την αρχή μέχρι το τέλος
- Ο αναλυτής που ηγείται της προσπάθειας θα πρέπει να έχει τη συνολική εικόνα και να συνθέτει τις εμπειρίες και τις γνώσεις των μελών της ομάδας ενθαρρύνοντας την επικοινωνία μεταξύ τους





Προμήθεια λογισμικού (η στρατηγική αγοράς έναντι της ανάπτυξης κατά παραγγελία)

- Η στρατηγική αξία, το συνολικό κόστος, ο χρόνος που απαιτείται για ανάπτυξη, η ανάγκη παραμετροποίησης και η διαθεσιμότητα πληροφοριακών πόρων επηρεάζουν την απόφαση αγοράς ή ανάπτυξης λογισμικού

 Ανάπτυξη κατά παραγγελία 

Πλεονεκτήματα	Μειονεκτήματα
<ul style="list-style-type: none">• μπορεί να προσδώσει ανταγωνιστικό πλεονέκτημα• Καλύτερη προσαρμογή στις ανάγκες του οργανισμού	<ul style="list-style-type: none">• υψηλότερο κόστος• απαιτεί περισσότερο χρόνο

 Αγορά έτοιμου λογισμικού 

Πλεονεκτήματα	Μειονεκτήματα
<ul style="list-style-type: none">• ακολουθεί βέλτιστες πρακτικές κάθε κλάδου• μικρότερο κόστος• μικρότερος χρόνος υλοποίησης	<ul style="list-style-type: none">• δεν καλύπτει όλες τις απαιτήσεις• μπορεί να μην ταιριάζει με την υπάρχουσα αρχιτεκτονική

Παράγοντες αποτυχίας των έργων

- Έλλειψη υποστήριξης από την ανώτατη διοίκηση
- Απουσία εμπλοκής των ενδιαφερόμενων μερών
- Ασαφείς απαιτήσεις
- Τεχνολογικά προβλήματα
- Ανεξέλεγκτη αύξηση αντικειμένου
- Ασαφείς ρόλοι και αρμοδιότητες
- Μη ρεαλιστικά χρονοδιαγράμματα
- Κακή επικοινωνία
- Κακοδιαχείριση αλλαγής



Download from
Dreamstime.com

This watermarked comp image is for previewing purposes only.

ID 15628100

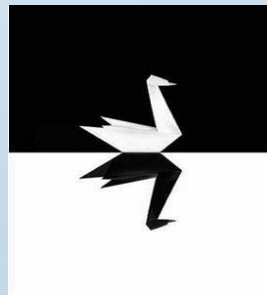
© Pzaxe | Dreamstime.com

Επίδραση του ανθρώπινου παράγοντα στη διαχείριση έργου

- Οι άνθρωποι συχνά υποπίπτουν σε σφάλματα
- Αναγνώριση καταστάσεων που οδηγούν συστηματικά σε λάθος εκτιμήσεις μπορούν να μειώσουν τη συχνότητα εμφάνισής τους
 - **Στρέβλωση επιβεβαίωσης:** Η τάση εστίασης σε πληροφορίες που υποστηρίζουν την άποψη που έχει ήδη σχηματιστεί, αγνοώντας δεδομένα τα οποία πιθανό την καταρρίπτουν
 - **Πλάνη προγραμματισμού:** Η τάση υποεκτίμησης του χρόνου που χρειάζεται για την ολοκλήρωση μιας εργασίας
 - **Στρέβλωση διαθεσιμότητας:** Η τάση υπολογισμού της πιθανότητας να συμβεί ένα γεγονός με βάση την ευκολία ανάκλησης σχετικών περιστατικών από προσωπική εμπειρία

Μαύροι κύκνοι

- Μαύρος κύκνος είναι ένας όρος που περιγράφει εξαιρετικά σπάνια γεγονότα που είναι σχεδόν αδύνατο να προβλεφθούν αλλά μπορεί να έχουν τεράστιες επιπτώσεις
- Οι μαύροι κύκνοι δημιουργούν τεράστιες προκλήσεις στο στρατηγικό σχεδιασμό
- Παραδείγματα:
 - Σπάνια φυσικά φαινόμενα (σεισμοί, πυρκαγιές, πλημμύρες κ.α.)
 - Τρομοκρατία
 - Οικονομικές καταστροφές
 - Καινοτόμα προϊόντα (προσωπικοί υπολογιστές, διαδίκτυο, έξυπνα τηλέφωνα κ.α.)

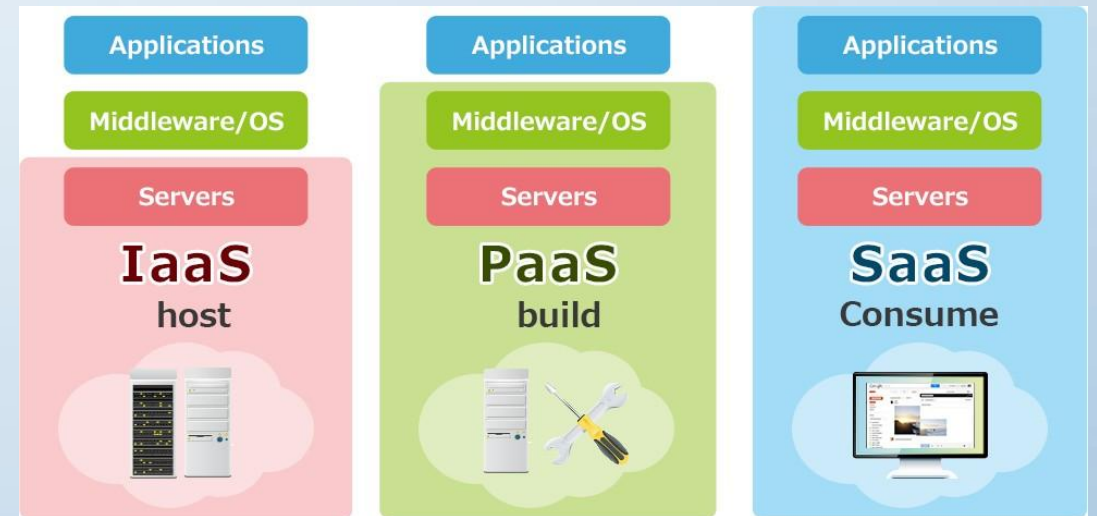


Τρόποι υλοποίησης

- Εσωτερική Ανάπτυξη Συστήματος & outsourcing
- Αγορά έτοιμου λογισμικού
- Χρήση Cloud

Αρχιτεκτονική προσανατολισμένη σε υπηρεσίες (SOA=Service Oriented Architecture)

- Η SOA είναι μια σύγχρονη τάση η οποία κερδίζει έδαφος στο σχεδιασμό λογισμικού
- Τα συστήματα «συναρμολογούνται» από σχετικά ανεξάρτητα επιμέρους στοιχεία λογισμικού καθένα εκ των οποίων χειρίζεται μια συγκεκριμένη επιχειρησιακή υπηρεσία



Software as a Service (SaaS)

- Το SaaS είναι η πλέον συνηθισμένη μορφή υπηρεσίας νέφους (cloud service), χρησιμοποιεί φυλλομετρητές (browsers) για να έχει πρόσβαση σε λογισμικό το οποίο εκτελείται σε εξυπηρετητές (servers)
- Η διαχείριση του λογισμικού και η εγκατάστασή του δεν είναι ευθύνη της επιχείρησης
- Δημοφιλείς SaaS εφαρμογές για επιχειρήσεις είναι το λογισμικό διαχείρισης πελατειακών σχέσεων Salesforce, Microsoft Office 365, η εφαρμογή διαμοιρασμού και συγχρονισμού αρχείων Dropbox, το λογισμικό τηλεδιασκέψεων GoToMeeting κ.α.
- Η χρήση εφαρμογών SaaS τείνει να μειώνει τα κόστη που σχετίζονται με το λογισμικό καθώς το κόστος άδειας χρήσης είναι μικρότερο και δεν υπάρχει η ανάγκη τεχνικού προσωπικού που θα πραγματοποιεί εγκαταστάσεις και ενημερώσεις



Platform as a Service (PaaS)

- Το PaaS λειτουργεί σε χαμηλότερο επίπεδο σε σχέση με το SaaS και παρέχει την υποδομή στην οποία μπορεί να αναπτυχθεί ή να εγκατασταθεί λογισμικό
- Οι πάροχοι PaaS φροντίζουν έτσι ώστε να προσφέρουν στους ενδιαφερόμενους υποδομές εξυπηρετητών και δικτυακής υποδομής σε συνδυασμό με λογισμικό λειτουργικών συστημάτων και άλλου λογισμικού εξυπηρετητών
- Οι επιχειρήσεις μπορούν να ζητούν επιπλέον πόρους καθώς τους χρειάζονται αντί να επενδύουν σε υλικό που μπορεί στην πλειονότητα των περιπτώσεων να υποχρησιμοποιείται
- Πάροχοι PaaS είναι οι: Engine Yard, Heroku, Google App Engine, Red Hat OpenShift, Windows Azure Cloud Services, Amazon Web Services, AppFog κ.α.

Infrastructure as a Service (IaaS)

- Το IaaS αφορά την αυτοματοποιημένη διάθεση υπολογιστικής ισχύος σε συνδυασμό με αποθήκευση στο νέφος καθώς και δυνατότητες δικτύωσης
- Οι πάροχοι IaaS παρέχουν στους πελάτες τους τη δυνατότητα να δημιουργήσουν ένα εικονικό κέντρο δεδομένων στο νέφος και να έχουν πρόσβαση στους ίδιους πόρους με αυτούς που θα είχαν αν διέθεταν ένα παραδοσιακό κέντρο δεδομένων στις δικές τους εγκαταστάσεις
- Πάροχοι IaaS είναι οι: Amazon Web Services, Windows Azure, Google Compute Space, Rackspace Open Cloud κ.α.

