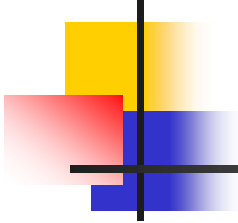


Πληροφοριακά Συστήματα Διοίκησης

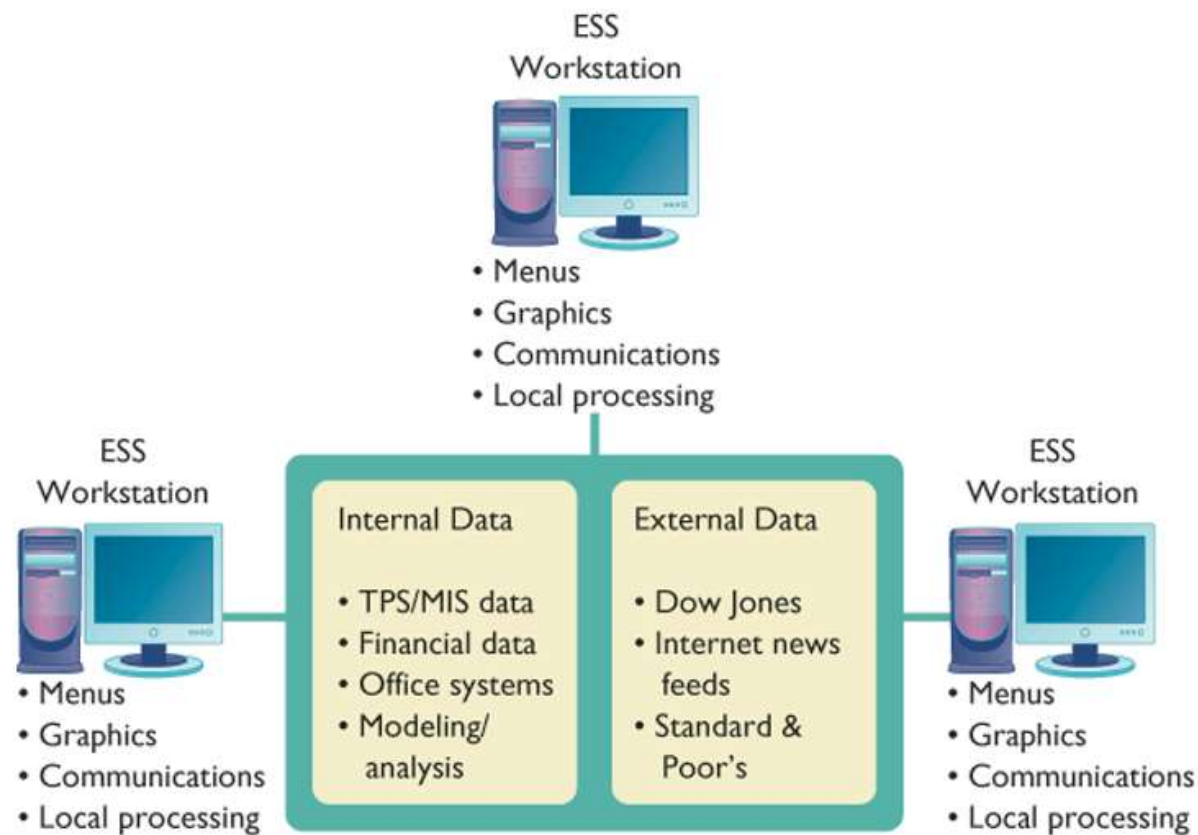
Διδάσκων: Δημόκας Νίκος, Ph.D.



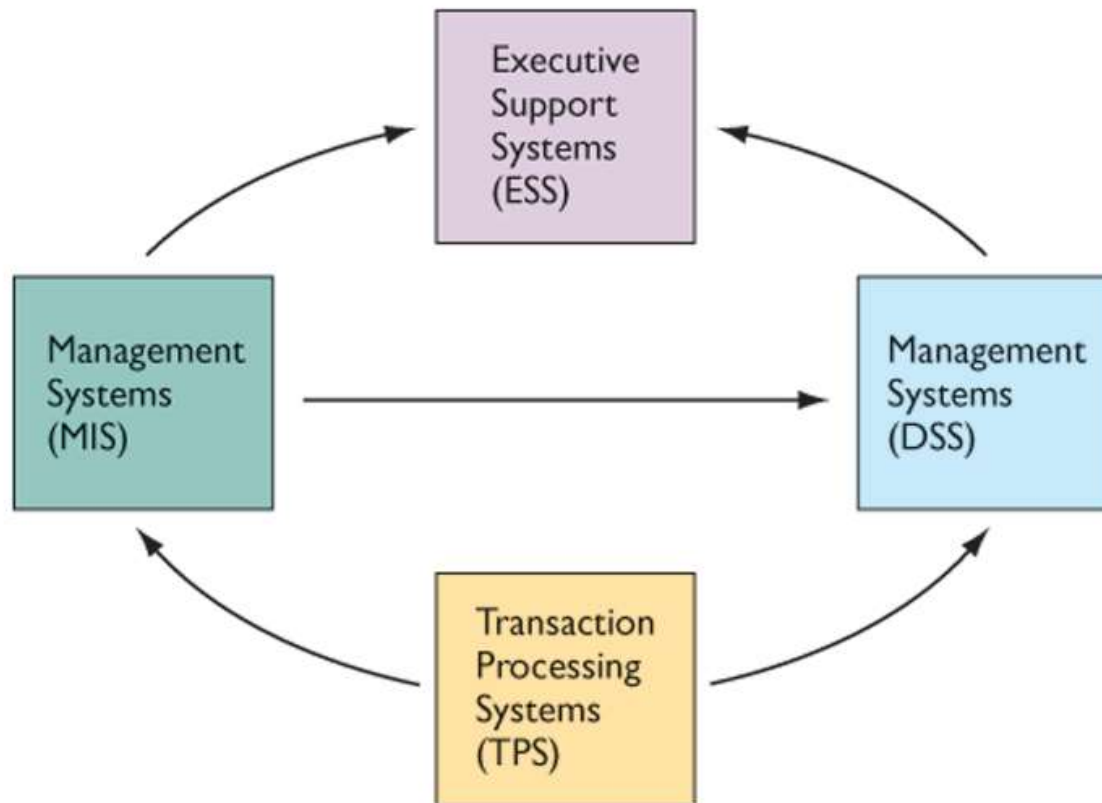
Συστήματα Υποστήριξης Διοίκησης

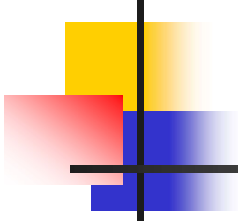
- Παρέχουν τις επικοινωνίες και το προγραμματιστικό περιβάλλον που εξυπηρετεί το στρατηγικό επίπεδο του οργανισμού
- Είσοδος: Εξωτερικά και εσωτερικά συγκεντρωτικά δεδομένα
- Επεξεργασία: Χρήση γραφικών, προσομοιώσεις, αλληλεπίδραση
- Έξοδος: Προβολές, απαντήσεις σε ερωτήσεις
- Χρήστες: Ανώτεροι Διοικητές

Μοντέλο ενός Τυπικού Συστήματος Υποστήριξης Διοίκησης



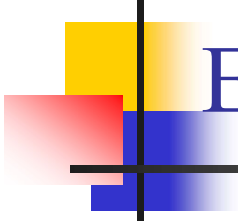
Σχέσεις Μεταξύ των Συστημάτων





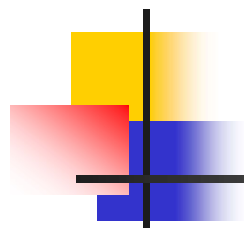
Υποσυστήματα Πληροφοριακών Συστημάτων Διοίκησης

- Υπάρχουν δύο προσεγγίσεις στο διαχωρισμό ενός ΠΣΔ σε υποσυστήματα
 - Σύμφωνα με τις λειτουργικές διαδικασίες
 - Σύμφωνα με τις διοικητικές διαδικασίες



Κύρια Υποσυστήματα Λειτουργιών Επιχειρησιακού Παραγωγικού Οργανισμού

Κύρια Υποσυστήματα	Μερικές Τυπικές Χρήσεις Λειτουργιών
Marketing	Προεκτίμηση και στρατηγική πωλήσεων, ανάλυση δεδομένων πελατών και πωλήσεων
Παραγωγή	Σχεδιασμός και προγραμματισμός της παραγωγής, ανάλυση ελέγχου κόστους
Διοικητική Μέριμνα	Προγραμματισμός και έλεγχος προμηθειών, διαχείριση αποθεμάτων, διανομή
Προσωπικό	Προγραμματισμός απαιτήσεων σε προσωπικό, ανάλυση αποδοτικότητας, διαχείριση μισθών
Οικονομικά/Λογιστικά	Οικονομική ανάλυση, κοστολόγηση, προσδιορισμός απαιτήσεων σε κεφάλαιο, έσοδα
Επεξεργασία Πληροφοριών	Σχεδιασμός Πληροφοριακών Συστημάτων, ανάλυση κόστους-ωφέλειας
Διοίκηση	Στρατηγικός Σχεδιασμός, καταμερισμός επενδύσεων

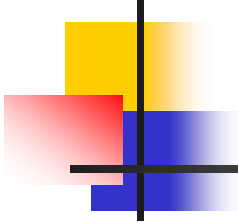


Υποσυστήματα Διαδικασιών

Υποσύστημα Διαδικασιών	Μερικές Τυπικές Χρήσεις
<i>Επεξεργασία Συναλλαγών</i>	Επεξεργασία παραγγελιών, αποστολών και παραλαβών
<i>Λειτουργικός Έλεγχος</i>	Προγραμματισμός διαδικασιών και εκθέσεις αποδοτικότητας
<i>Διοικητικός Έλεγχος</i>	Τυποποίηση προϋπολογισμών και κατακερματισμού επενδύσεων
<i>Στρατηγικός Σχεδιασμός</i>	Τυποποίηση στόχων και στρατηγικών σχεδίων

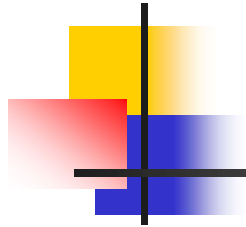
Φυσική Δομή ενός Πληροφοριακού Συστήματος Διοίκησης

Τμήμα Φυσικής	Περιγραφή Δομής ΠΣΔ
<i>Εξοπλισμός</i>	Αναφέρεται στον εξοπλισμό πληροφορικής και στις υπόλοιπες διασυν-δεδεμένες συσκευές. Σκοπός του εξοπλισμού είναι να εξυπηρετήσει τις εξής λειτουργίες: 1. Εισαγωγή στοιχείων 2. Εξαγωγή πληροφοριών 3. Δευτερεύουσα αποθήκευση για δεδομένα και προγράμματα 4. Κεντρική επεξεργασία (υπολογισμούς, ελέγχους και πρωτεύουσα μνήμη) 5. Επικοινωνίες
<i>Λογισμικό</i>	Το λογισμικό αποτελεί ευρεία έννοια που χρησιμοποιείται να δηλώσει τις εντολές που κατευθύνουν τη λειτουργία του εξοπλισμού. Οι δύο βασικοί τύποι λογισμικού είναι το Λογισμικό Συστήματος (System Software) και το Λογισμικό Εφαρμογών (Application Software).
<i>Βάση Δεδομένων</i>	Η Βάση Δεδομένων περιλαμβάνει όλα τα δεδομένα με την απαραίτητη διασύνδεση που διαχειρίζεται το Λογισμικό Εφαρμογών. Ένα αυτόνομο τμήμα αποθηκευμένων δεδομένων συνήθως αποκαλείται αρχείο. Όλα τα δεδομένα αποθηκεύονται σε ταινίες, δίσκους κ.λπ. που είναι μονάδες βοηθητικής ή περιφερειακής μνήμης.
<i>Διαδικασίες</i>	Οι διαδικασίες αποτελούν φυσικό τμήμα της δομής ενός ΠΣΔ καθώς υφίστανται καταγεγραμμένες και χρησιμοποιούνται για την υποστήριξη της λειτουργίας του Συστήματος. Υπάρχουν οι παρακάτω βασικές κατή-γορίες διαδικασιών: 1. Χρήσης και Λειτουργίας 2. Επεξεργασίας 3. Ασφάλειας 4. Αναβάθμισης και Ανανέωσης
<i>Προσωπικό Λειτουργίας</i>	Διαχειριστές Συστημάτων, Αναλυτές Συστημάτων, Προγραμματιστές, Προσωπικό Εισαγωγής Στοιχείων κ.λπ.

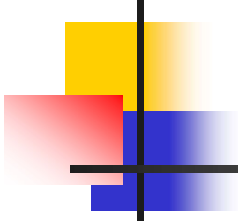


Χρήστες Ηλεκτρονικού Πληροφοριακού Συστήματος

Χρήστης	Χρήση
<i>Υπαλληλικό Προσωπικό</i>	Συναλλαγές, επεξεργασία εισερχομένων στοιχείων και διερευνήσεις
<i>Προϊστάμενοι Τμημάτων</i>	Λειτουργικά δεδομένα. Υποστήριξη σχεδιασμού, προγραμματισμού, αναγνώρισης μη ομαλών λειτουργιών και λήψη αποφάσεων.
<i>Ειδικευμένο Προσωπικό</i>	Πληροφορίες προς ανάλυση. Υποστήριξη στην ανάλυση, το σχεδιασμό και τη σύνταξη εκθέσεων.
<i>Διευθυντές</i>	• Τακτικές εκθέσεις • Ειδικές αναζητήσεις και διερευνήσεις σκοπιμότητας • Ειδικές αναλύσεις και αναλύσεις σκοπιμότητας • Ειδικές εκθέσεις • Υποστήριξη στην αναζήτηση προβλημάτων και αναγνώρισης ευκαιριών • Υποστήριξη στην ανάλυση και λήψη αποφάσεων

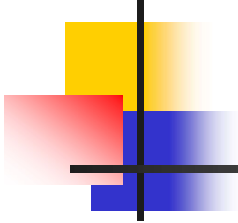


*Μια αναλυτική μελέτη των
Συστημάτων Επεξεργασίας Συναλλαγών
Transaction Processing Systems*



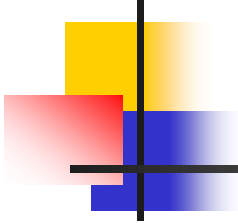
ACID ιδιότητες

- Atomicity
- Consistency
- Isolation
- Durability



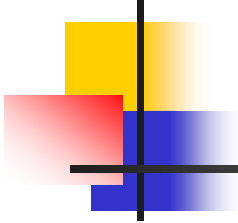
Atomicity

- Atomicity: οι μεταβολές στη ΒΔ πρέπει να ακολουθούν έναν *όλα ή τίποτα* κανόνα. Κάθε συναλλαγή ορίζεται ως *ατομική* εάν όταν ένα μέρος της συναλλαγής αποτύχει τότε αποτυγχάνει ολόκληρη η συναλλαγή και η ΒΔ παραμένει αμετάβλητη.
- Οι συναλλαγές μπορούν να αποτύχουν για διάφορους λόγους:
 1. Αποτυχία υλικού: Ένας οδηγός δίσκου αποτυγχάνει, εμποδίζοντας κάποιες μεταβολές στη ΒΔ να ολοκληρωθούν
 2. Αποτυχία συστήματος: Ο χρήστης χάνει τη σύνδεση εν μέσω συναλλαγής
 3. Αποτυχία ΒΔ: π.χ., Η ΒΔ δεν έχει επιπλέον αποθηκευτικό χώρο
 4. Αποτυχία εφαρμογής: Η εφαρμογή προσπαθεί να θέσει δεδομένα τα οποία παραβιάζουν έναν κανόνα της ΒΔ



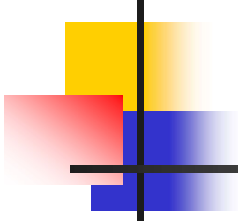
Consistency

- Η ιδιότητα της *συνέπειας* βεβαιώνει ότι η ΒΔ παραμένει σε συνεπή – σταθερή κατάσταση.
- Ο κανόνας συνέπειας εφαρμόζεται μόνο σε κανόνες ακεραιότητας. Έτσι, εάν μια ΒΔ επιτρέπει τα πεδία μιας πλειάδας να λειτουργούν ως αναφορές σε άλλες πλειάδες, τότε η συνέπεια συνεπάγεται ότι η ΒΔ πρέπει να επιβάλει ακεραιότητα αναφορών



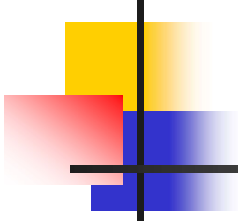
Isolation

- Η ιδιότητα της *απομόνωσης* αναφέρεται στην απαίτηση ότι άλλες λειτουργίες – διαδικασίες δεν μπορούν να προσπελάσουν δεδομένα τα οποία έχουν μεταβληθεί από μια συναλλαγή η οποία δεν έχει ακόμη ολοκληρωθεί.
- Μια συναλλαγή πρέπει να περιμένει την ολοκλήρωση μιας άλλης εφόσον χρειάζεται να προσπελάσει τα δεδομένα που μεταβάλλονται από την πρώτη.



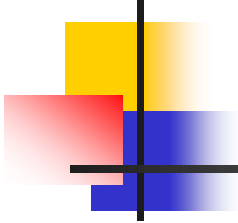
Durability

- Αντοχή είναι η εγγύηση της ΒΔ ότι εάν ο χρήστης δηλώσει την επιτυχία μιας συναλλαγής τότε η συναλλαγή δεν χάνεται. Οι αλλαγές στα δεδομένα που επιφέρει η συναλλαγή θα επιβιώσουν από τυχόν αποτυχία του συστήματος, και όλοι οι περιορισμοί ακεραιότητας θα έχουν ικανοποιηθεί. Πολλά συστήματα ΒΔ υλοποιούν την αντοχή με το να γράφουν τις συναλλαγές σε transaction log αρχεία έτσι ώστε να επαναδημιουργήσουν το σύστημα ύστερα από μια μελλοντική αποτυχία.



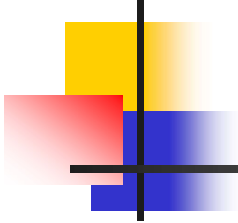
Παράδειγμα (1/3)

- Η ΒΔ έχει δύο πεδία, Α και Β, σε δύο εγγραφές. Ένας περιορισμός ακεραιότητας απαιτεί ότι η τιμή στο Α και η τιμή στο Β πρέπει αθροιστικά να είναι 100.
- Αποτυχία ατομικότητας
 - Η συναλλαγή αφαιρεί 10 από το Α και προσθέτει 10 στο Β. Εάν επιτύχει τότε είναι έγκυρη. Διαφορετικά, εάν η συναλλαγή δεν μπορεί να μεταβάλλει το Β τότε ο περιορισμό ακεραιότητας παραβιάζεται.



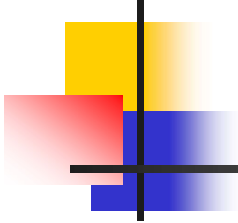
Παράδειγμα (2/3)

- Αποτυχία συνέπειας
 - Παράδειγμα κανόνων που μπορούν να επιβληθούν από ένα σύστημα ΒΔ είναι ότι
 - οι τιμές των πρωτεύον κλειδιών μιας εγγραφής χαρακτηρίζουν μοναδικά την εγγραφή,
 - οι τιμές που αποθηκεύονται στα πεδία είναι σωστού τύπου και στο σωστό εύρος,
 - τα εξωτερικά κλειδιά είναι έγκυρα



Παράδειγμα (3/4)

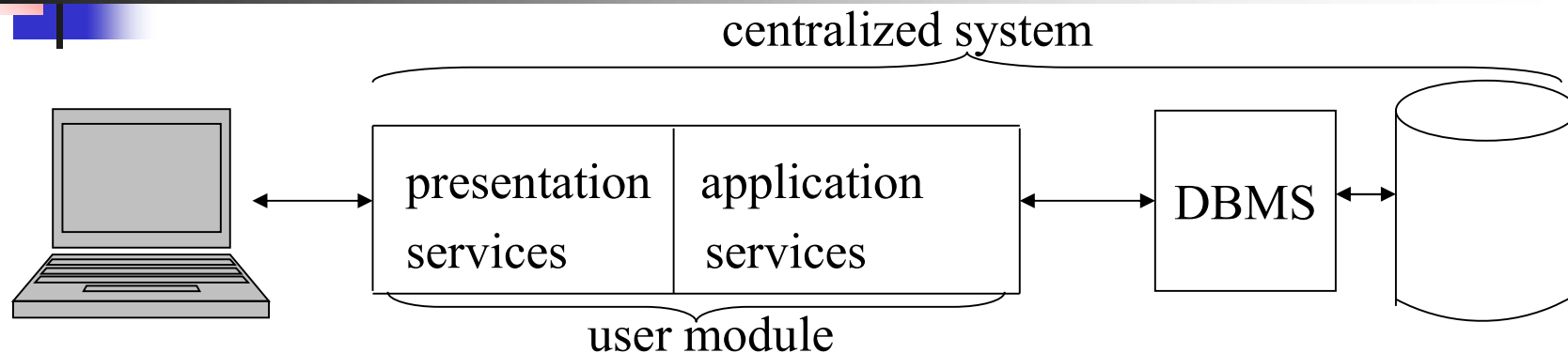
- Θεωρούμε 2 συναλλαγές που εκτελούνται ταυτόχρονα. Κάθε μια προσπαθεί να μεταβάλλει τα ίδια δεδομένα. T1 μεταφέρει 10 από το A στο B. T2 μεταφέρει 10 από B στο A.
 - Αφαιρέστε 10 από το A
 - Προσθέστε 10 στο B
 - Αφαιρέστε 10 από το B
 - Προσθέστε 10 στο A
- Εάν οι ενέργειες πραγματοποιηθούν στη σειρά η απομόνωση διατηρείται αν και το T2 πρέπει να περιμένει. Τι συμβαίνει αν το T1 αποτύχει στη μέση. Η ΒΔ αποβάλλει τα αποτελέσματα του T1, και το T2 βλέπει μόνο έγκυρα δεδομένα.



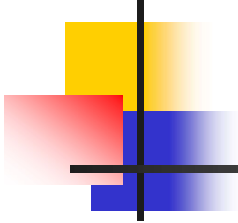
Παράδειγμα (4/4)

- Μια συναλλαγή μεταφέρει 10 από το A στο B. Αφαιρεί 10 από το A και μετά προσθέτει 10 στο B. Σε αυτό το σημείο, ένα μήνυμα «επιτυχίας» στέλνεται στο χρήστη. Όμως, οι αλλαγές βρίσκονται ακόμα στην ουρά στο buffer του δίσκου περιμένοντας την αποστολή στο δίσκο. Η ισχύς μειώνεται και οι αλλαγές χάνονται. Ο χρήστης θεωρεί ότι έγιναν οι αλλαγές.
- Για να ικανοποιηθεί ο περιορισμός *durability*, η ΒΔ επιβεβαιώνει την επιτυχία μιας συναλλαγής όταν τερματιστεί.

Απλό Σύστημα Χρήστη



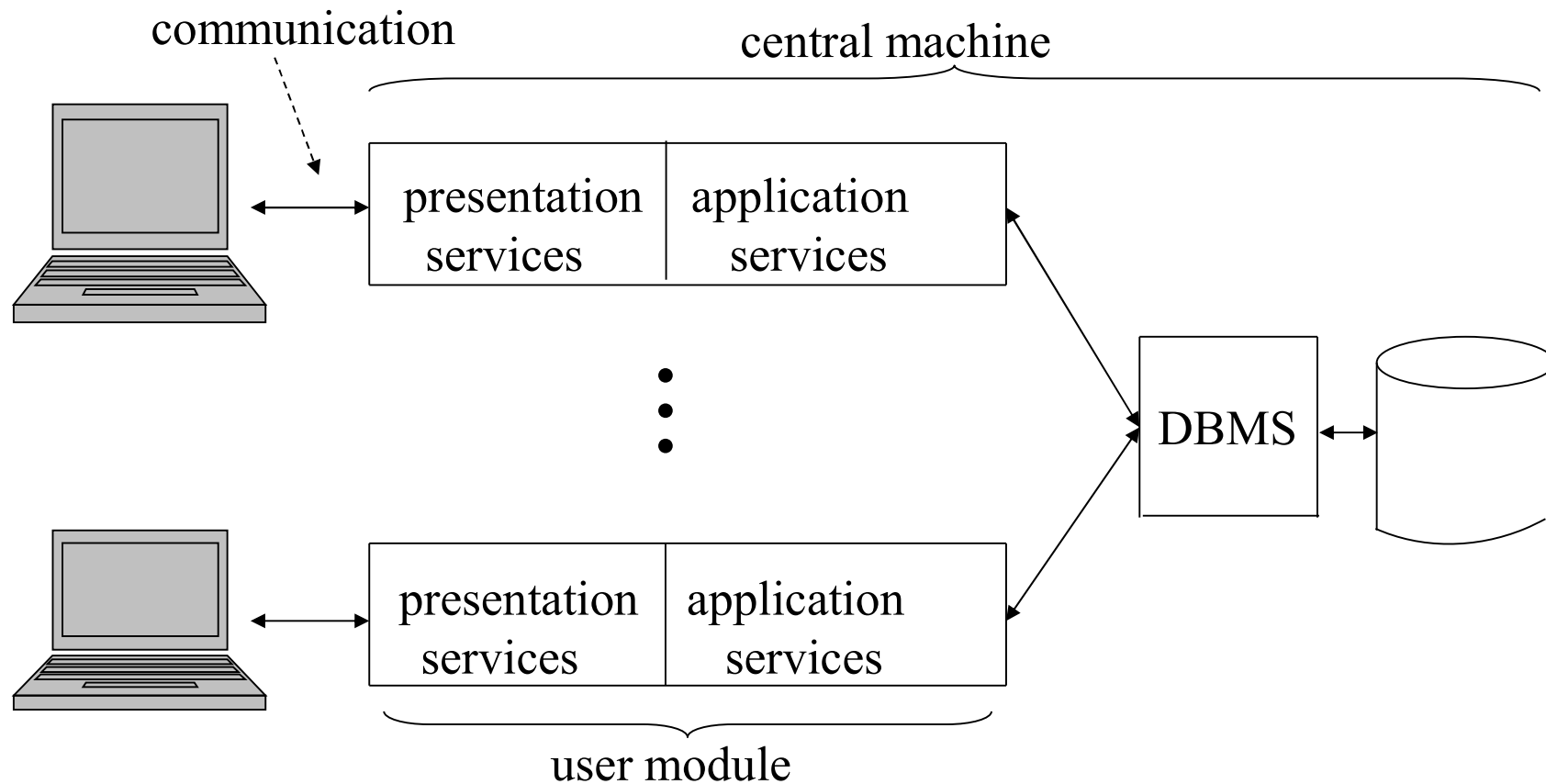
- **Presentation Services** – επιδεικνύει φόρμες, διαχειρίζεται ροές πληροφορίας από και προς την οθόνη
- **Application Services** – υλοποιεί τις αιτήσεις του χρήστη και αλληλεπιδρά με το DBMS
- **ACID properties** automatic (isolation is trivial) ή δεν απαιτείται

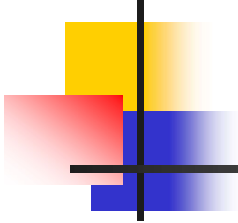


Κεντριοποιημένο Multi-User Σύστημα

- Dumb terminals συνδέονται σε υπολογιστή μεγάλης ισχύος (mainframe)
 - Οι υπηρεσίες εφαρμογής και προβολής (Application, presentation) πραγματοποιούνται στο mainframe
- Απαιτούνται ACID ιδιότητες
 - Απομόνωση: DBMS βλέπει ένα πρόγραμμα διαστρωμάτωσης
 - Ατομικότητα και ανοχή: Το σύστημα υποστηρίζει ένα ολοκληρωμένο εγχείρημα

Κεντριοποιημένο Multi-User Σύστημα

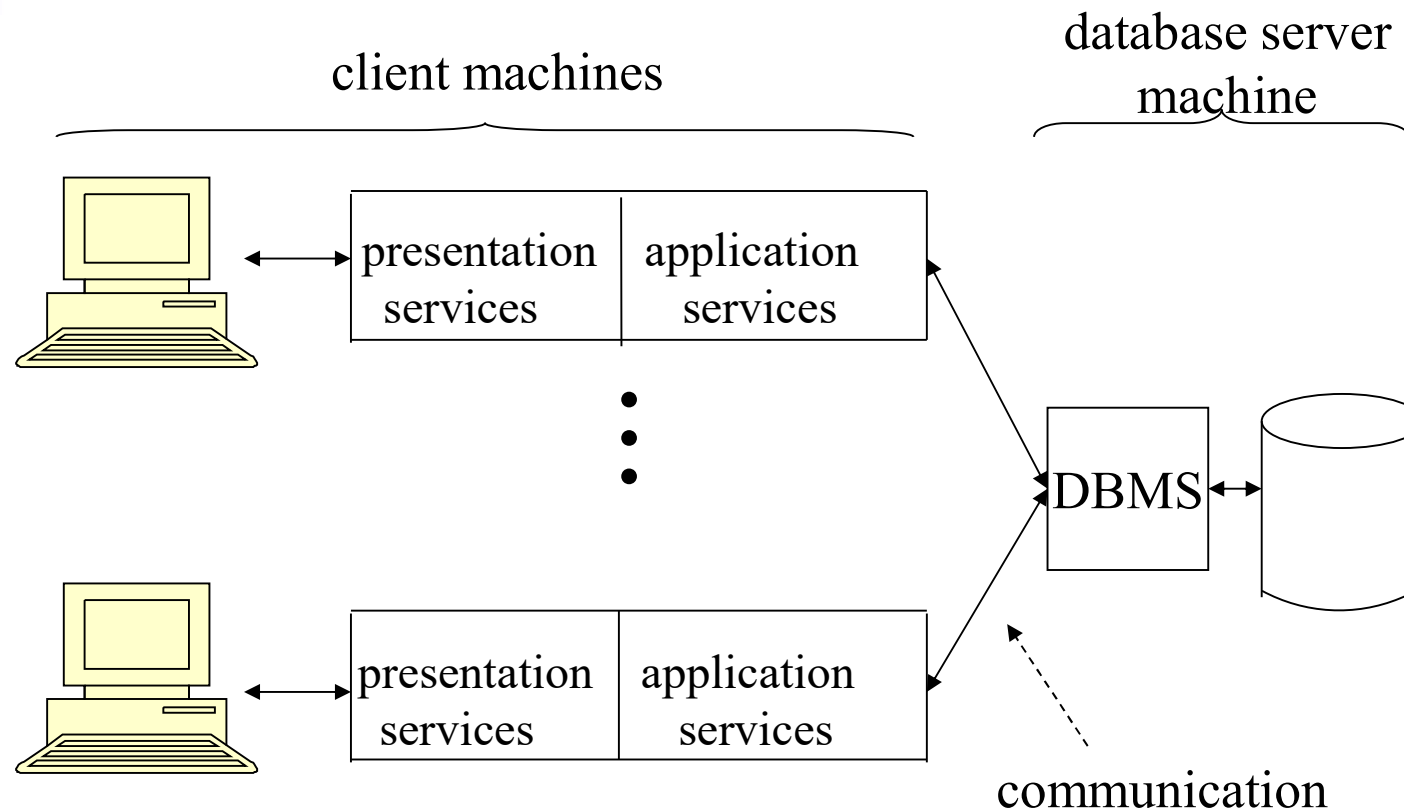


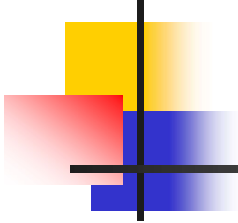


Transaction Processing σε Κατανεμημένα Συστήματα

- Το μειωμένο κόστος του υλικού και της επικοινωνίας κάνει δυνατή την κατανομή components του συστήματος επεξεργασίας συναλλαγών
 - Dumb terminals αντικαθίστανται από υπολογιστές
- Χρησιμοποιείται γενικά η Client/server οργάνωση

Μοντέλο δύο επιπέδων (Two-Tier) ενός TPS

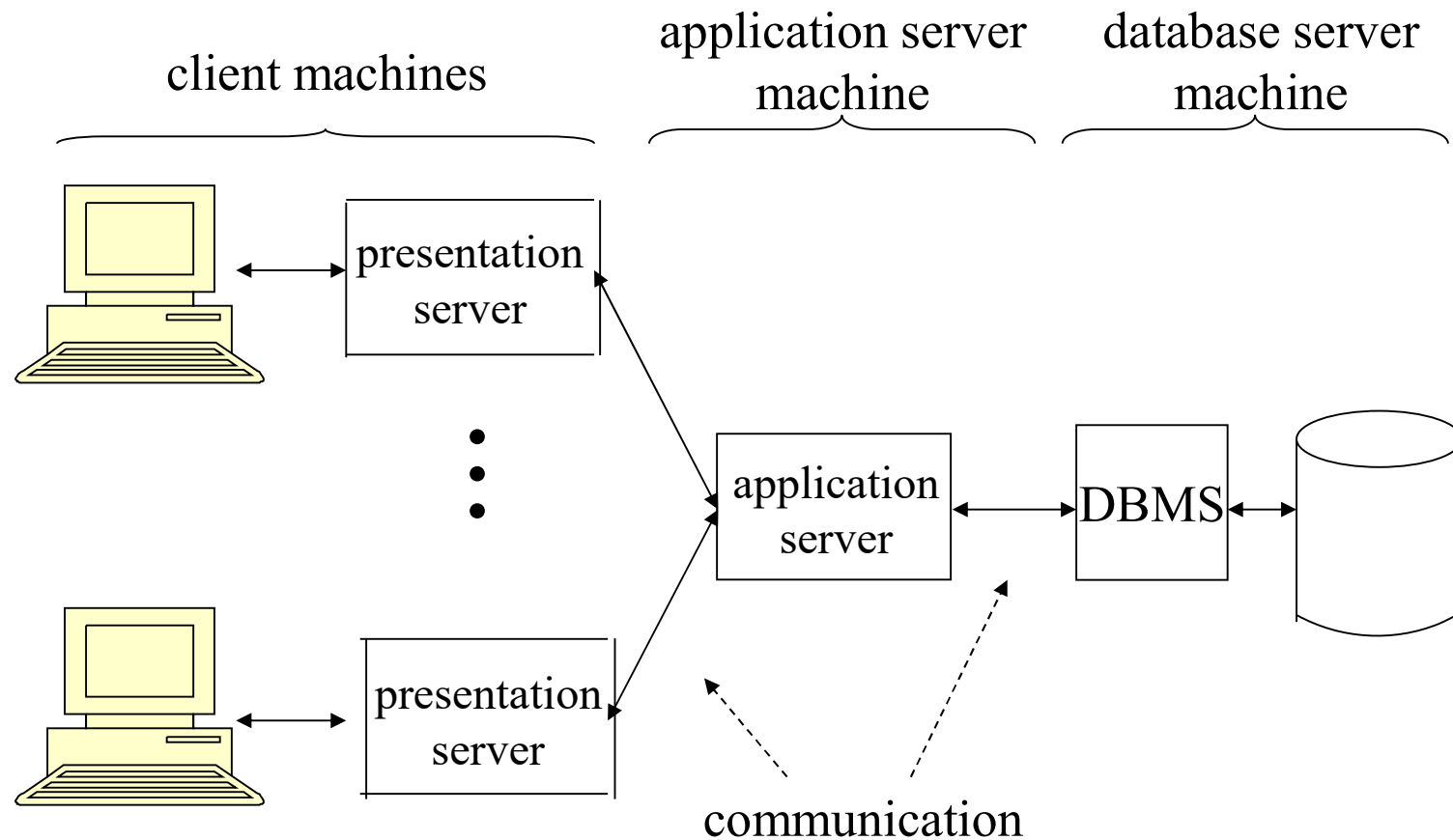


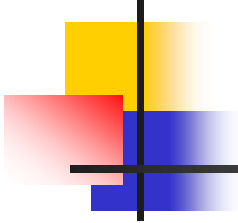


Χρήση των Stored Procedures

- Πλεονεκτήματα που προκύπτουν όταν ο database server εξάγει ένα stored procedure interface αντί για ένα SQL interface
 - Ασφάλεια: Οι διαδικασίες είναι προστατευμένες επειδή διατηρούνται στον εξυπηρετητή της ΒΔ
 - Μείωση της κίνησης στο δίκτυο, επειδή μειώνεται ο χρόνος απόκρισης
 - Εύκολη διατήρηση καθώς οι νέες εκδόσεις δεν χρειάζεται να διανεμηθούν στους πελάτες (client sites)
 - Η αυθεντικοποίηση υλοποιείται στο επίπεδο διαδικασίας (και όχι στο statement)
 - Οι διαδικασίες αναπτύσσονται εκ των προτέρων
 - Οι εφαρμογές δεν αλληλεπιδρούν απευθείας με τη ΒΔ

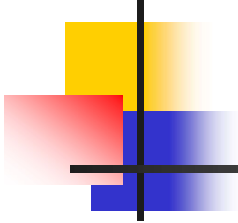
Μοντέλο τριών επιπέδων (Three-Tier) ενός TPS





Application Server

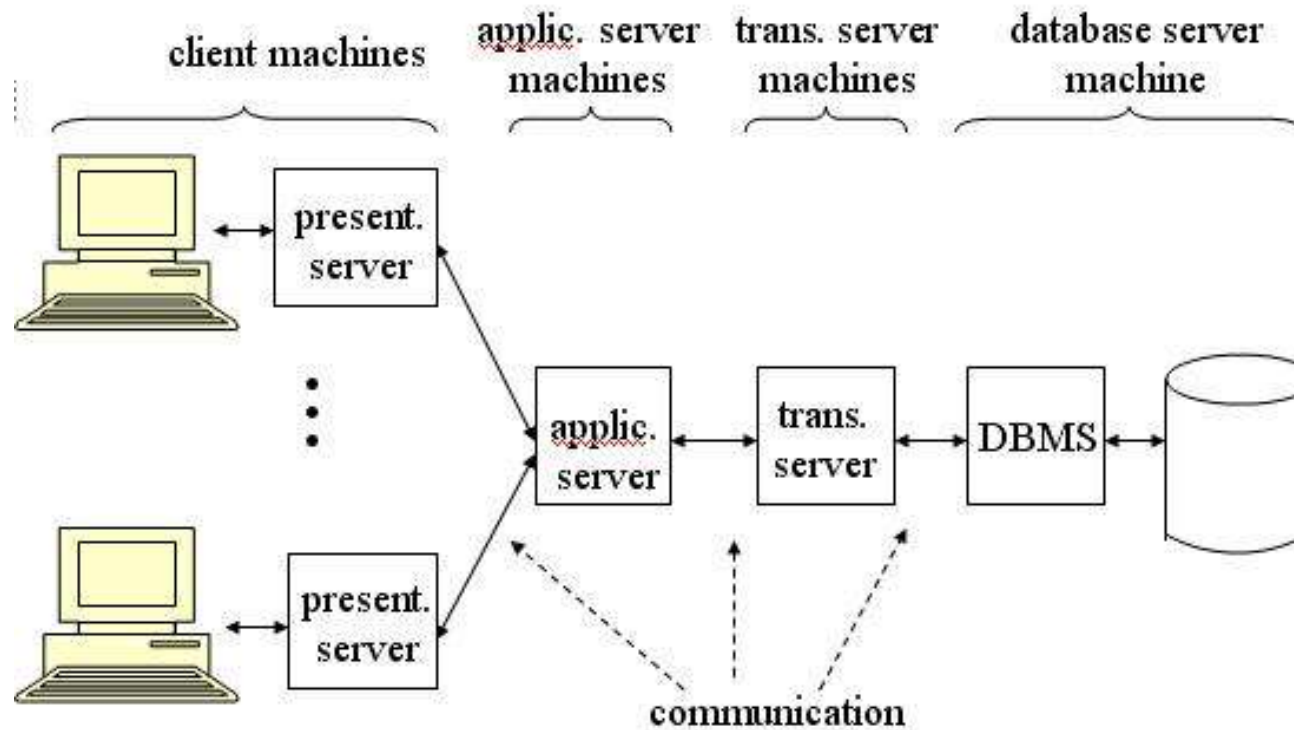
- Θέτει όρια συναλλαγών
- Λειτουργεί ως ελεγκτής ροών εργασίας: υλοποιεί τις αιτήσεις του χρήστη ως μια ακολουθία εργασιών
 - π.χ., registration = (check prerequisites, add student to course, bill student)
- Λειτουργεί ως δρομολογητής
 - Οι κατανεμημένες συναλλαγές περιλαμβάνουν πολλούς εξυπηρετητές
- Επειδή οι ροές εργασίας είναι χρονοβόρες και ο application server εξυπηρετεί πολλούς πελάτες, application server είναι συχνά multi-threaded

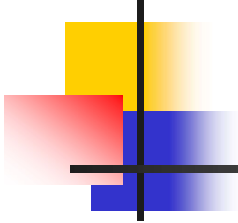


Transaction Server

- Ένας Transaction server βρίσκεται κοντά σε ένα DBMS
 - Ο Application server βρίσκεται κοντά στους πελάτες
- Transaction server πραγματοποιεί μαζική επεξεργασία δεδομένων
 - Ο Application server χρησιμοποιεί κάθε διαθέσιμο transaction server για να εκτελέσει μια stored procedure. Συνεισφέρει στην ισορροπία φόρτου εργασίας (load balancing)

Μοντέλο τριών επιπέδων (Three-Tier) ενός TPS

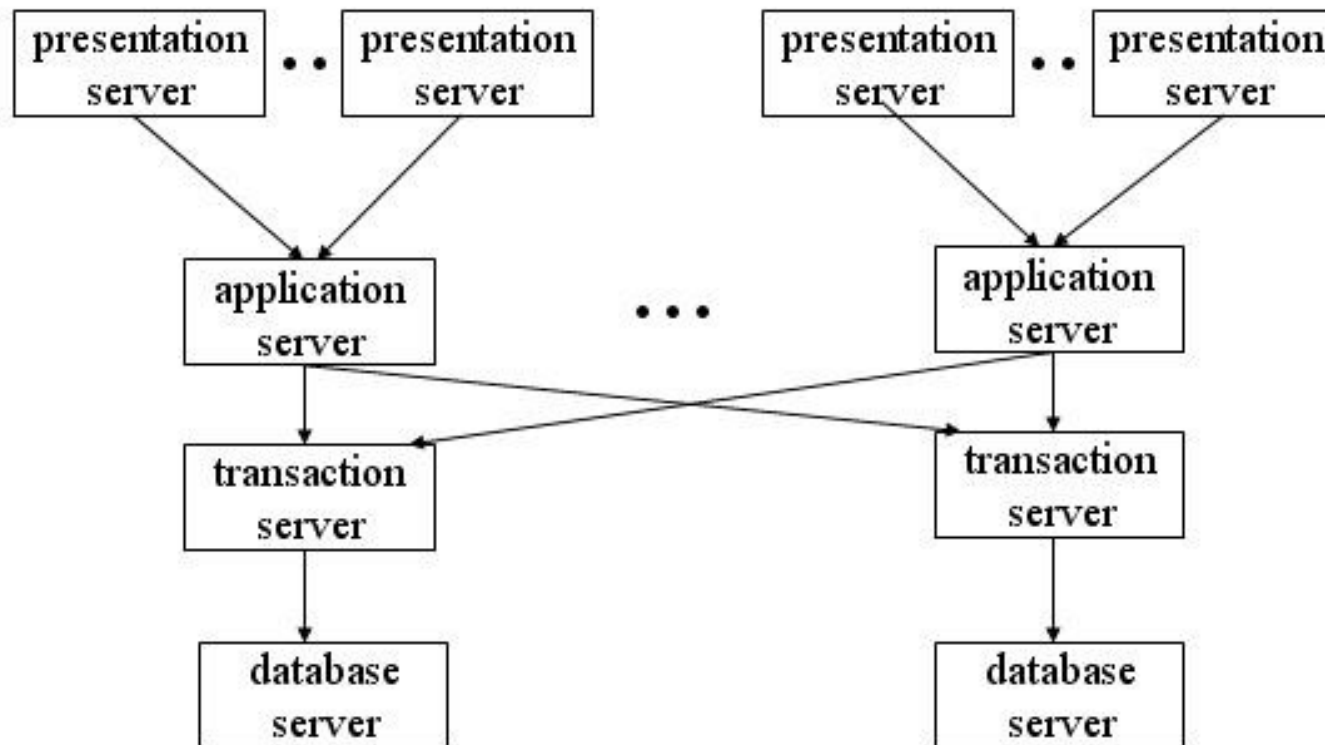


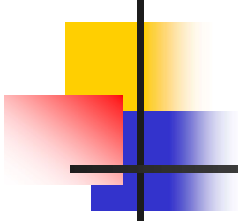


Επίπεδα Αφαίρεσης

- Ο εξυπηρετητής παρουσίασης (presentation server) υλοποιεί τη διασύνδεση με το χρήστη
- Ο εξυπηρετητής εφαρμογής (application server) υλοποιεί την εξυπηρέτηση των αιτήσεων των χρηστών
- Οι Stored procedures (ή transaction server) υλοποιεί τη διασύνδεση με τη ΒΔ
- Ο εξυπηρετητής ΒΔ (database server) υλοποιεί το σχεσιακό μοντέλο

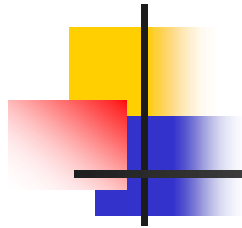
Διασύνδεση των servers σε Μοντέλο τριών επιπέδων





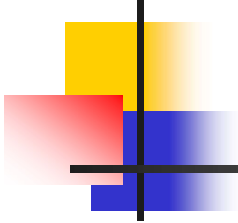
Sessions και πλαίσιο (context)

- Ένα session υφίσταται ανάμεσα σε δύο οντότητες εάν ανταλλάσσουν μηνύματα ενώ συνεργάζονται για να πραγματοποιήσουν κάποιο στόχο
- Κάθε οντότητα διατηρεί κάποια πληροφορία που περιγράφει την κατάσταση σχετικά με τη συμμετοχή στο κοινό σκοπό
- Η πληροφορία κατάστασης αναφέρεται ως πλαίσιο (context)



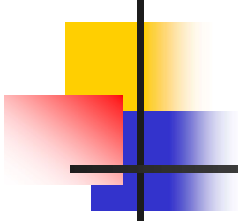
Sessions

- Εγκαθιδρύονται σε διαφορετικά επίπεδα αφαίρεσης:
 - Communication sessions (χαμηλό επίπεδο αφαίρεσης)
 - Το πλαίσιο περιγράφει την κατάσταση του καναλιού επικοινωνίας
- Client/server sessions (Υψηλό επίπεδο αφαίρεσης)
 - Το πλαίσιο χρησιμοποιείται από τον εξυπηρετητή και περιγράφει τον πελάτη



Επικοινωνία σε TPSs

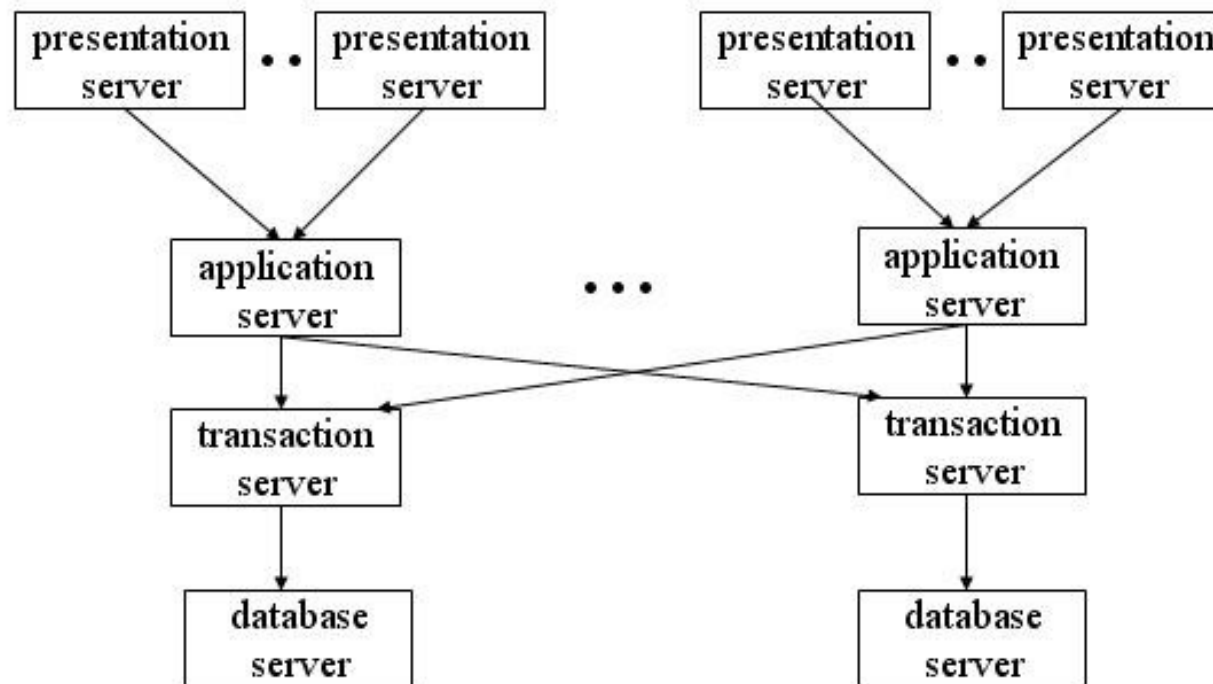
- Μοντέλο δύο επιπέδων (Two-tiered model):
 - Ο Presentation/application server επικοινωνεί με τον database server
- Μοντέλο τριών επιπέδων (Three-tiered model):
 - Ο Presentation server επικοινωνεί με τον application server
 - Ο Application server επικοινωνεί με τον transaction/database server
- Στην περίπτωση αποστολής πολλαπλών μηνυμάτων
 - Αποδοτική και αξιόπιστη επικοινωνία είναι απαραίτητη
- Τα Sessions χρησιμοποιούνται για να πετύχουν αυτούς τους στόχους
 - Session set-up/take-down costly => session is long-term

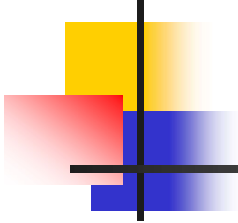


Επικοινωνία Sessions

- Πλαίσιο: ακολουθιακός αριθμός, διεύθυνση, κλειδιά κρυπτογράφησης, ...
- Σημαντικό είναι το επιπλέον κόστος (overhead) διατήρησης των sessions
 - Έτσι ο αριθμός των sessions πρέπει να είναι περιορισμένος
- Μοντέλο δύο επιπέδων:
 - Ένας πελάτης έχει ένα session με κάθε ένα database server που προσπελαύνει
- Μοντέλο τριών επιπέδων:
 - Κάθε πελάτης έχει ένα session με τον application server
 - Κάθε application server πολυπλέκει τη σύνδεση του με το database server ως προς όλες τις συναλλαγές που υποστηρίζει

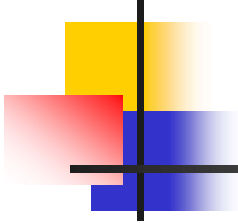
Sessions σε τριών επιπέδων μοντέλο





Ετερογενή εναντίον Ομογενών TPS

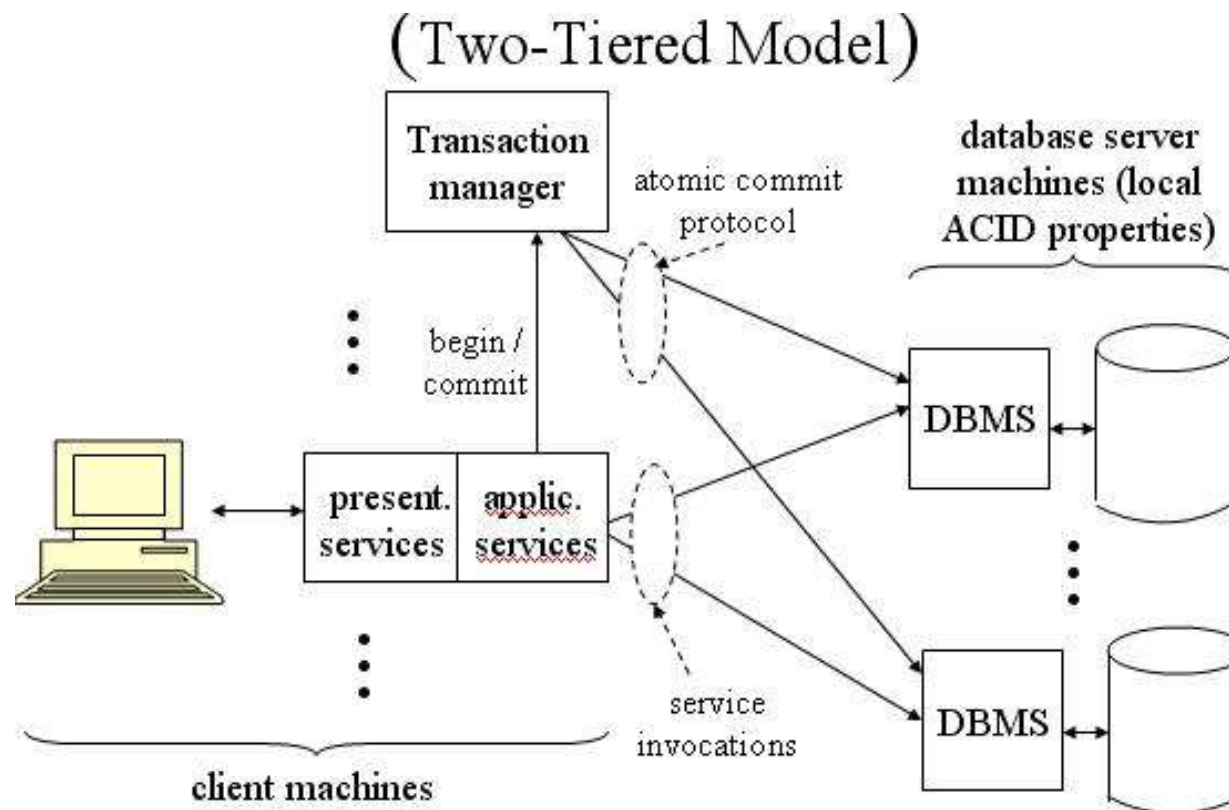
- Τα ομογενή συστήματα συντίθεται από modules υλικού και λογισμικού του ίδιου κατασκευαστή
 - Τα Modules επικοινωνούν διαμέσου κατάλληλων διασυνδέσεων
 - Έτσι, τα προϊόντα άλλων προμηθευτών δεν μπορούν να συμπεριληφθούν
 - Αναφέρονται ως TP-Lite συστήματα
- Τα ετερογενή συστήματα συντίθεται από modules υλικού και λογισμικού διαφορετικών κατασκευαστών
 - Τα Modules επικοινωνούν διαμέσου προτύπων
 - Αναφέρονται ως TP-Heavy συστήματα

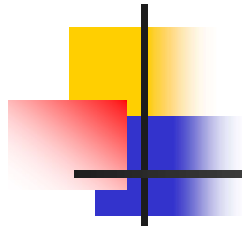


Διαχειριστής Συναλλαγών (1/2)

- Το ενδιάμεσο λογισμικό υποστηρίζει την συνολική ατομικότητα των κατανεμημένων συναλλαγών
 - Μια εφαρμογή καλεί ένα διαχειριστή όταν αρχικοποιείται μια συναλλαγή
 - Ο διαχειριστής πληροφορείται κάθε φορά που ένας νέος εξυπηρετητής λαμβάνει μέρος στο σύστημα συναλλαγών
 - Η εφαρμογή καλεί το διαχειριστή όταν ολοκληρωθεί η συναλλαγή
 - Ο διαχειριστής συντονίζει το πρωτόκολλο δέσμευσης ατομικότητας μεταξύ των εξυπηρετητών για να εγγυηθεί μια συνολική ατομικότητα

Διαχειριστής Συναλλαγών (2/2)





TP Monitor

- Ένα TP Monitor είναι μια συλλογή από συστατικά ενδιάμεσου λογισμικού (middleware components) τα οποία είναι χρήσιμα για το κτήσιμο ενός ετερογενούς συστήματος επεξεργασίας συναλλαγών
 - Περιλαμβάνει το διαχειριστή συναλλαγών
 - Ανεξάρτητες υπηρεσίες εφαρμογής
- Επίπεδο λογισμικού που βρίσκεται μεταξύ του λειτουργικού συστήματος και της εφαρμογής
- Παράγει την αφαίρεση μιας (συνολικής) συναλλαγής