# ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΙΙ





Ύλη

- Έννοιες Επεξεργασίας Δοσοληψιών
- Τεχνικές Ελέγχου Συνδρομικότητας
- Τεχνικές Ανάκαμψεις από Σφάλματα
- Κατανεμημένες και Παράλληλες Βάσεις Δεδομένων
- Βάσεις Δεδομένων και Διαδίκτυο
- Βάσεις Δεδομένων και Συστήματα Ομοτίμων (peer-to-peer systems)
- Διαχείριση Χωρικών Δεδομένων

Κοπάξτε τη web σελίδα





Διαδικαστικά

- Θα υπάρχει τελική εξέταση
- Κάποιες ενδιάμεσες ασκήσεις (2 σύνολα) για κατανόηση της ύλης
- Ίσως κάποια γενική άσκηση στα «νέα» θέματα (web, p2p) που θα μετρά μόνο θετικά στο βαθμό (+1 ή 2 μονάδες)



# Επεξεργασία Δοσοληψιών





- Η ταυτόχρονη εκτέλεση προγραμμάτων χρηστών είναι απαραίτητη για την καλή απόδοση ενός ΣΔΒΔ
  - Επειδή οι προσπελάσεις στο δίσκο είναι συχνές και σχετικά αργές, είναι σημαντικό να κρατείται η cpu απασχολημένη με πολλά προγράμματα χρηστών

⇒ Πολυχρηστικά ΣΔΒΔ

Διαπλεγμένο μοντέλο ταυτόχρονης εκτέλεσης

## Δοσοληψία (transaction)

**εκτέλεση ενός προγράμματος** που προσπελαύνει ή τροποποιεί το περιεχόμενο της βάσης δεδομένων

*το πώς βλέπει το ΣΔΒΔ τα προγράμματα των χρηστών*

## Δοσοληψία (transaction)

Ένα πρόγραμμα χρήστη μπορεί να εκτελεί πολλές λειτουργίες στα δεδομένα που ανακτά από τη ΒΔ, αλλά το ΣΔΒΔ ενδιαφέρεται μόνο για τα δεδομένα που *διαβάζονται/γράφονται* στη ΒΔ

- Ανάγνωση(X) - R(X)
- Εγγραφή(X) - W(X)





- Οι χρήστες υποβάλουν δοσοληψίες και πρέπει να μπορούν να θεωρούν ότι κάθε δοσοληψία *εκτελείται μόνη της.*
  - Η **συνδρομικότητα** (concurrency) *επιτυγχάνεται από το ΣΔΒΔ* που διαπλέκει τις πράξεις (αναγνώσεις/εγγραφές) των διαφόρων συναλλαγών





- Όταν μια δοσοληψία υποβάλλεται στο ΣΔΒΔ το σύστημα πρέπει να εξασφαλίσει ότι (α) είτε όλες οι πράξεις της θα ολοκληρωθούν είτε (β) καμία δε θα εκτελεστεί - δηλαδή δε θα έχει καμία επίδραση στη ΒΔ -- ακόμα και αν συμβούν αποτυχίες
- Αυτή είναι μια σημαντική ιδιότητα που πρέπει να εξασφαλίσει το ΣΔΒΔ - Ο χρήστης πρέπει να μπορεί να θεωρεί ότι όλο το πρόγραμμα (πράξεις) εκτελούνται σε ένα βήμα είτε καμία πράξη δεν εκτελείται (ατομικότητα των δοσοληψιών)





Είδη Αποτυχιών

Δυο κατηγορίες: καταστροφή ή όχι της μόνιμης αποθήκευσης (δίσκου)

Παραδείγματα αποτυχιών ...





Πιο αναλυτικά ...





⌘ Θεωρείστε δύο συναλλαγές (*Xacts*):

```
T1:  BEGIN  R(X), X=X-N,  W(X), R(Y), Y=Y+N, W(Y),  END
T2:  BEGIN  R(X) X=X+M,  W(X)  END
```

- Διαισθητικά, **η T1 μεταφέρει N κρατήσεις θέσεων από μια πτήση (X) και τις μεταφέρει σε μία άλλη (Y). Η T2 απλώς κρατά M θέσεις στην πρώτη πτήση (τη X)**
- Δεν υπάρχει καμία εγγύηση ότι η T1 θα εκτελεστεί πριν την T2 ή το ανάποδο, αν και η δύο υποβληθούν ταυτόχρονα. Ωστόσο, το συνολικό αποτέλεσμα πρέπει να είναι *ισοδύναμο* με τη μία ή την άλλη περίπτωση (δηλαδή, με κάποια ***σειριακή εκτέλεση*** των δύο δοσοληψιών)

X = 100 κρατήσεις
Y = 90 κρατήσεις
Μεταφορά N = 30 κρατήσεων
Νέα κράτηση M = 5 θέσεις

## Προβλήματα Λόγω Συνδρομικότητας

T1 μεταφέρει N κρατήσεις θέσεων από την πτήση X στη Y.
T2 κρατά M θέσεις στη X. Σωστό X = 75, Y = 120

| T1 | T2 |
|---|---|
| BEGIN | |
| R(X) read 100 | |
| X=X-N X=70 | |
| W(X) write X = 70 | |
| R(Y) read 90 | |
| Y=Y+N Y = 120 | |
| W(Y) write Y = 120 | |
| END | |
| | BEGIN |
| | R(X) read 70 |
| | X=X+M X = 75 |
| | W(X) write X = 75 |
| | END |

σειριακή εκτέλεση
T1 → T2

| T1 | T2 |
|---|---|
| | BEGIN |
| | R(X) read 100 |
| | X=X+M X = 105 |
| | W(X) write 105 |
| | END |
| BEGIN | |
| R(X) read 105 | |
| X=X-N X = 75 | |
| W(X) write X = 75 | |
| R(Y) read 90 | |
| Y=Y+N Y = 120 | |
| W(Y) write Y = 120 | |
| END | |

σειριακή εκτέλεση
T2 → T1



X = 100 κρατήσεις
Y = 90 κρατήσεις
Μεταφορά N = 30 κρατήσεων
Νέα κράτηση M = 5 θέσεις

## Προβλήματα Λόγω Συνδρομικότητας

| T1 | T2 |
|---|---|
| BEGIN | |
| R(X) read 100 | |
| X=X-N X = 70 | |
| | BEGIN |
| | R(X) read 100 |
| | X=X+M X = 105 |
| W(X) write X =70 | |
| R(Y) Y = 90 | |
| | W(X) write X = 105 |
| | END |
| Y=Y+N Y = 110 | |
| W(Y) write Y = 110 | |
| END | |

Η τιμή του X είναι λανθασμένη

Απώλεια Ενημερώσεων



X = 100 κρατήσεις
Y = 90 κρατήσεις
Μεταφορά N = 30 κρατήσεων
Νέα κράτηση M = 5 θέσεις

## Προβλήματα Λόγω Συνδρομικότητας

| T1 | T2 |
|---|---|
| BEGIN | |
| R(X) read 100 | |
| X=X-N X = 70 | |
| W(X) write X = 70 | |
| | BEGIN |
| | R(X) read 70 |
| | X=X+M X = 75 |
| | W(X) write X =75 |
| | END |
| Y=Y+N | |

Dirty Read

Προσωρινή Ενημέρωση

Τι θα συμβεί αν η T1 αποτύχει – η T2 θα έχει διαβάσει «ανύπαρκτη τιμή»



X = 100 κρατήσεις
Y = 90 κρατήσεις
Μεταφορά N = 30 κρατήσεων
Νέα κράτηση M = 5 θέσεις

## Προβλήματα Λόγω Συνδρομικότητας

| T1 | T2 |
|---|---|
| BEGIN | |
| R(X) read 100 | |
| | BEGIN |
| | R(X) read 100 |
| | X=X+M X = 105 |
| | W(X) write X = 105 |
| | END |
| R(X) | |

Μη Επαναλήψιμη Ανάγνωση

Η τιμή του X που διαβάζει η T1 είναι διαφορετική!!



## Πράξεις μιας Δοσοληψίας

- Ας προσπαθήσουμε να ορίσουμε το πρόβλημα στη γενική του μορφή



## Πράξεις μιας Δοσοληψίας

### Πράξεις Δοσοληψιών

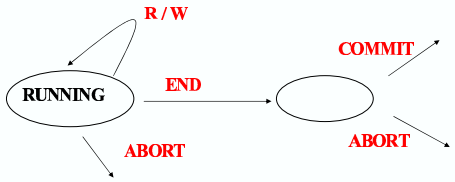
- BEGIN
- R(X)
- W(X)
- END
- COMMIT (επικύρωση) – επιτυχία – όλες οι τροποποιήσεις επικυρώνονται και δεν μπορούν να αναιρεθούν
- ABORT (ακύρωση ή ανάκληση) – αποτυχία – όλες οι τροποποιήσεις πρέπει να αναιρεθούν

## Πράξεις μιας Δοσοληψίας

- Μια δοσοληψία μπορεί να να επικυρωθεί (commit) αφού ολοκληρώσει *όλες* τις πράξεις της ενώ μπορεί να ακυρωθεί (abort) αφού εκτελέσει *κάποιες* από τις πράξεις της
  - Το ΣΔΒΔ *logs* όλες τις πράξεις έτσι ώστε να μπορεί να αναιρέσει (undo) τις πράξεις μιας ακυρωμένης (aborted) δοσοληψίας.



## Καταστάσεις μιας Δοσοληψίας






## Συνδρομικότητα σε ΣΔΒΔ

ΥΠΟΘΕΣΕΙΣ

Κάθε δοσοληψία πρέπει να αφήνει τη ΒΔ σε μια συνεπή κατάσταση αν η ΒΔ ήταν σε συνεπή κατάσταση όταν άρχισε η δοσοληψία (συνέπεια)

- Το ΣΔΒΔ επιβάλει κάποιους ΠΑ (Περιορισμούς Ακεραιότητας) με βάση τους ΠΑ που έχουν δηλωθεί στις εντολές CREATE TABLE
- Πέρα από αυτό, το ΣΔΒΔ δεν καταλαβαίνει τη σημασιολογία των δεδομένων (π.χ., δεν καταλαβαίνει πώς να υπολογίσει το επιτόκιο)

*Αποτελέσματα της διαπλεγμένης εκτέλεσης δοσοληψιών (έλεγχος συνδρομικότητας) και των αποτυχιών (ανάκαμψη)*



## Επιθυμητές Ιδιότητες μιας Δοσοληψίας

Ιδιότητες Δοσοληψιών

- **Atomicity** (ατομικότητα) - είτε όλες οι πράξεις είτε καμία
- **Consistency** (συνέπεια) - διατήρηση συνέπειας της ΒΔ
- **Isolation** (απομόνωση) - δεν αποκαλύπτει ενδιάμεσα αποτελέσματα
- **Durability** (μονιμότητα ή διάρκεια) - μετά την επικύρωση μιας δοσοληψίας οι αλλαγές δεν είναι δυνατόν να χαθούν



## Επιθυμητές Ιδιότητες μιας Δοσοληψίας

- Atomicity (ατομικότητα) → ΤΕΧΝΙΚΕΣ ΑΝΑΚΑΜΨΕΙΣ
- Consistency (συνέπεια) → ΥΠΕΥΘΥΝΟΤΗΤΑ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΤΗ
- Isolation (απομόνωση) → ΕΛΕΓΧΟΣ ΣΥΝΔΡΟΜΙΚΟΤΗΤΑΣ
- Durability (μονιμότητα ή διάρκεια) → ΤΕΧΝΙΚΕΣ ΑΝΑΚΑΜΨΕΙΣ



## Επιθυμητές Ιδιότητες μιας Δοσοληψίας

- Isolation (απομόνωση) → ΕΛΕΓΧΟΣ ΣΥΝΔΡΟΜΙΚΟΤΗΤΑΣ

Βαθμός απομόνωσης 0 :: δεν επικαλύπτει ασταθείς αναγνώσεις (dirty reads) δοσοληψιών με μεγαλύτερο βαθμό

Βαθμός απομόνωσης 1 :: δεν παρουσιάζει απώλειες ενημέρωσης

Βαθμός απομόνωσης 2 :: ούτε απώλειες ενημέρωσης, ούτε ασταθείς αναγνώσεις

Βαθμός απομόνωσης 3 :: επίπεδο 2 + επαναλήψιμες αναγνώσεις

Τώρα που καταλάβαμε το πρόβλημα::

μπορούμε να βρούμε ένα θεωρητικό μοντέλο που να το εκφράζει;





Ορισμοί

1. δοσοληψία
2. διαπεπλεγμένη εκτέλεση δοσοληψιών (χρονοπρόγραμμα)
3. σωστό -- αποδεκτό χρονοπρόγραμμα

Στη συνέχεια ... (στο επόμενο μάθημα)

Τεχνικές για να πετύχουμε σωστά χρονοπρογράμματα (έλεγχος συνδρομικότητας)





Μια δοσοληψία είναι μια ακολουθία από πράξεις εγγραφής και ανάγνωσης που τελειώνει με μια πράξη επικύρωσης (commit) ή με μια πράξη ακύρωσης (abort)





⌘ Θεωρείστε τις δύο συναλλαγές (*Xacts*) του παραδείγματος:

| | |
|---|---|
| T1: | BEGIN R(X), X=X-N, W(X), R(Y), Y=Y+N, W(Y), END |
| T2: | BEGIN R(X) X=X+M, W(X) END |

- T1: R(X) W(X) R(Y) W(Y) *C*
- T2: R(X) W(X) *C*





- *Εκφράζει μια συγκεκριμένη εκτέλεση ενός συνόλου δοσοληψιών*
- Οι πράξεις των δοσοληψιών εμφανίζονται στο χρονοπρόγραμμα με τη σειρά που εκτελούνται

Συγκεκριμένα

Ένα χρονοπρόγραμμα (schedule) S των δοσοληψιών $T_1$, $T_2$, .., $T_n$ είναι μια διάταξη των πράξεων τους με τον περιορισμό ότι για κάθε δοσοληψία $T_i$ που συμμετέχει στο S οι πράξεις της $T_i$ στο S πρέπει να εμφανίζονται με την ίδια σειρά που εμφανίζονται στην $T_i$





Θα χρησιμοποιούμε δείκτη στις πράξεις που να δείχνει σε ποια δοσοληψία αναφέρονται

| T1 | T2 |
|---|---|
| $R_1(X)$ | |
| $W_1(X)$ | |
| $R_1(Y)$ | |
| $W_1(Y)$ | |
| $C_1$ | |
| | $R_2(X)$ |
| | $W_2(X)$ |
| | $C_2$ |

S: $R_1(X)$ $W_1(X)$ $R_1(Y)$ $W_1(Y)$ $C_1$ $R_2(X)$ $W_2(X)$ $C_2$

| T1 | T2 |
|---|---|
| $R_1(X)$ | |
| | $R_2(X)$ |
| $W_1(X)$ | |
| $R_1(Y)$ | |
| | $W_2(X)$ |
| | $C_2$ |
| $W_1(Y)$ | |
| $C_1$ | |

Τόσα διαφορετικά χρονοπρογράμματα όσες και πιθανές εκτελέσεις

S: $R_1(X)$ $R_2(X)$ $W_1(X)$ $R_1(Y)$ $W_2(X)$ $C_2$ $W_1(Y)$ $C_1$





| T1 | T2 |
|---|---|
| $R_1(X)$ | |
| $W_1(X)$ | |
| | $R_2(X)$ |
| | $W_2(X)$ |
| | $C_2$ |
| $A_1$ | |

S: $R_1(X)$ $W_1(X)$ $R_2(X)$ $W_2(X)$ $C_2$ $A_1$





**Διάταξη πράξεων**

S1: $R_1(X)$ $R_2(X)$ $W_1(X)$ $R_1(Y)$ $W_2(X)$ $C_2$ $W_1(Y)$ $C_1$

S2: $R_2(X)$ $R_1(X)$ $W_1(X)$ $R_1(Y)$ $W_2(X)$ $C_2$ $W_1(Y)$ $C_1$

Ποια είναι η σχέση των χρονοπρογραμμάτων S1 και S2;

Διαισθητικά «δεν διαφέρουν»





**Σύγκρουση πράξεων σε χρονοπρόγραμμα**

Δύο πράξεις σε ένα χρονοπρόγραμμα συγκρούονται αν (α) ανήκουν σε διαφορετικές δοσοληψίες, (β) προσπελαύνουν το ίδιο στοιχείο, και (γ) μια από αυτές είναι πράξη εγγραφής (W)





σημασία έχει η σχετική θέση (διάταξη) των πράξεων που συγκρούονται

S1: $R_1(X)$ $R_2(X)$ $W_1(X)$ $R_1(Y)$ $W_2(X)$ $C_2$ $W_1(Y)$ $C_1$

S2: $R_2(X)$ $R_1(X)$ $W_1(X)$ $R_1(Y)$ $W_2(X)$ $C_2$ $W_1(Y)$ $C_1$

Τα S1 και S2 ισοδύναμα (διαφέρουν μόνο στη διάταξη πράξεων που δε συγκρούονται) ⇒ μερική διάταξη





Επίσης, ζητάμε να μην περιέχει ενεργές δοσοληψίες (πλήρες)

Ένα πλήρες χρονοπρόγραμμα (schedule) S των δοσοληψιών $T_1$, $T_2$, .., $T_n$ είναι ένα σύνολο από πράξεις και μια *μερική* διάταξη των πράξεων αυτών με τους ακόλουθους περιορισμούς:

(i) οι πράξεις του S είναι ακριβώς οι πράξεις των $T_1$, $T_2$, .., $T_n$ συμπεριλαμβανομένης μιας πράξης ακύρωσης ή επικύρωσης ως τελευταίας πράξης σε κάθε δοσοληψία στο χρονοπρόγραμμα

(ii) για κάθε δοσοληψία $T_i$ που συμμετέχει στο S οι πράξεις της $T_i$ στο S πρέπει να εμφανίζονται με την ίδια σειρά που εμφανίζονται στην $T_i$

(iii) Για κάθε ζεύγος *συγκρουόμενων* πράξεων, μια από τις δύο πρέπει να προηγείται της άλλης στο χρονοπρόγραμμα

## Ορισμός Χρονοπρογράμματος

Επικυρωμένη προβολή C(S) ενός χρονοπρογράμματος S η οποία περιλαμβάνει μόνο τις πράξεις του S που ανήκουν σε επικυρωμένες δοσοληψίες



## Επεξεργασία Δοσοληψιών

Ορισμοί

√ 1. δοσοληψία

√ 2. διαπεπλεγμένη εκτέλεση δοσοληψιών (χρονοπρόγραμμα)

⇒ 3. σωστό -- αποδεκτό χρονοπρόγραμμα

Ισοδυναμία με σειριακό χρονοπρόγραμμα



## Σειριοποιησιμότητα

Σειριακά Χρονοπρογράμματα: χρονοπρογράμματα που δεν διαπλέκουν πράξεις διαφορετικών δοσοληψιών (οι πράξεις κάθε δοσοληψίας εκτελούνται διαδοχικά, χωρίς παρεμβολή πράξεων από άλλη δοσοληψία)

Παρατήρηση: Αν κάθε δοσοληψία διατηρεί τη συνέπεια, τότε κάθε σειριακό χρονοπρόγραμμα διατηρεί τη συνέπεια

Ένα σειριακό χρονοπρόγραμμα είναι σωστό

S: $R_1(X)$ $W_1(X)$ $R_1(Y)$ $W_1(Y)$ $C_1$ $R_2(X)$ $W_2(X)$ $C_2$



## Σειριοποιησιμότητα

Ισοδύναμα Χρονοπρογράμματα :

Για κάθε κατάσταση της ΒΔ, το αποτέλεσμα της εκτέλεσης του πρώτου χρονοπρογράμματος είναι το ίδιο με το αποτέλεσμα του δεύτερου χρονοπρογράμματος

Ένα χρονοπρόγραμμα ισοδύναμο με ένα σειριακό είναι σωστό



## Σειριοποιησιμότητα

Ισοδύναμα Χρονοπρογράμματα : Για κάθε κατάσταση της ΒΔ, το αποτέλεσμα της εκτέλεσης του πρώτου χρονοπρογράμματος είναι *το ίδιο* με το αποτέλεσμα του δεύτερου χρονοπρογράμματος

Αρκεί;

Είναι δυνατόν να ελεγχθεί;



## Σειριοποιησιμότητα

Σειριοποιήσιμο Χρονοπρόγραμμα :

Ένα χρονοπρόγραμμα που είναι *ισοδύναμο* με κάποιο σειριακό

Τι σημαίνει ισοδύναμο;

Τι σημαίνει ισοδύναμο;

- Ισοδυναμία βάσει συγκρούσεων
- Ισοδυναμία όψεων





• Ισοδύναμα Χρονοπρογράμματα βάσει Συγκρούσεων:

Δυο χρονοπρογράμματα είναι ισοδύναμα βάσει συγκρούσεων αν η διάταξη κάθε ζεύγους συγκρουόμενων πράξεων είναι ίδια και στα δυο χρονοπρογράμματα.





S1: $R_1(X)$ $R_2(X)$ $W_1(X)$ $R_1(Y)$ $W_2(X)$ $C_2$ $W_1(Y)$ $C_1$

*Παραδείγματα*

S2: $R_2(X)$ $R_1(X)$ $W_1(X)$ $R_1(Y)$ $W_2(X)$ $C_2$ $W_1(Y)$ $C_1$

S3: $R_2(X)$ $W_2(X)$ $C_2$ $R_1(X)$ $W_1(X)$ $R_1(Y)$ $W_1(Y)$ $C_1$

S4: $R_2(X)$ $R_1(X)$ $W_1(X)$ $R_1(Y)$ $W_1(Y)$ $C_1$ $W_2(X)$ $C_2$





Σειριοποιησιμότητα βάσει Συγκρούσεων:

Ένα χρονοπρόγραμμα S είναι σειριοποιήσιμο βάσει συγκρούσεων αν είναι ισοδύναμο βάσει συγκρούσεων με κάποιο σειριακό χρονοπρόγραμμα S'.

•Σε αυτήν την περίπτωση μπορούμε να αναδιατάξουμε τις μη συγκρουόμενες πράξεις στο S μέχρι να σχηματίσουμε ένα ισοδύναμο σειριακό χρονοπρόγραμμα.





S1: $R_1(X)$ $R_2(X)$ $W_1(X)$ $R_1(Y)$ $W_2(X)$ $C_2$ $W_1(Y)$ $C_1$

S2: $R_2(X)$ $R_1(X)$ $W_1(X)$ $R_1(Y)$ $W_2(X)$ $C_2$ $W_1(Y)$ $C_1$

Σειριοποιήσιμα;

Sα: $R_1(X)$ $W_1(X)$ $R_1(Y)$ $W_1(Y)$ $C_1$ $R_2(X)$ $W_2(X)$ $C_2$

Sβ: $R_2(X)$ $W_2(X)$ $C_2$ $R_1(X)$ $W_1(X)$ $R_1(Y)$ $W_1(Y)$ $C_1$





⌘ Θεωρείστε δύο συναλλαγές (*Xacts*):

| | |
|---|---|
| T1: | BEGIN A=A+100, B=B-100 END |
| T2: | BEGIN A=1.06*A, B=1.06*B END |

❖ Διαισθητικά, η πρώτη μεταφέρει $100 από το λογαριασμό B στο λογαριασμό A. Η δεύτερη καταθέτει και στους δύο τόκο 6%.

❖ Δεν υπάρχει καμία εγγύηση ότι η T1 θα εκτελεστεί πριν την T2 ή το ανάποδο, αν και η δύο υποβληθούν ταυτόχρονα. Ωστόσο, το συνολικό αποτέλεσμα πρέπει να είναι ισοδύναμο με τη μία ή την άλλη περίπτωση (δηλαδή, με κάποια *σειριακή εκτέλεση* των δύο δοσοληψιών)

T1 μεταφέρει $100 από το λογαριασμό Β στον Α.
T2 καταθέτει και στους δύο τόκο 6%.

## Παραδείγματα

- Θεωρείστε ένα πιθανό χρονοπρόγραμμα:

| | | |
|---|---|---|
| T1: | A=A+100, | B=B-100 |
| T2: | A=1.06*A, | B=1.06*B |

- Αυτό είναι OK. Αλλά:

| | | |
|---|---|---|
| T1: | A=A+100, | B=B-100 |
| T2: | A=1.06*A, B=1.06*B | |

- Το δεύτερο χρονοπρόγραμμα:

| | | |
|---|---|---|
| T1: | $R_1(A)$ $W_1(A)$, | $R_1(B)$, $W_2(B)$ |
| T2: | $R_2(A)$, $W_2(A)$ $R_2(B)$ | $W_2(B)$ |



## Παραδείγματα

- Ανάγνωση Uncommitted δεδομένων (WR Συγκρούσεις "dirty reads", προσωρινή ενημέρωση):

| | | |
|---|---|---|
| T1: | $R_1(A)$ $W_1(A)$ | $R_1(B)W_1(B)$, Abort |
| T2: | $R_2(A)$ $W_2(A)$, C | |

- Μη επαναλήψιμες αναγνώσεις (RW Συγκρούσεις):

| | | |
|---|---|---|
| T1: | $R_1(A)$ | $R_1(A)$ $W_1(A)$ C |
| T2: | $R_2(A)$ $W_2(A)$ C | |



## Παραδείγματα

- Απώλειες Ενημερώσεων (WW Συγκρούσεις):

| | | |
|---|---|---|
| T1: | $W_1(A)$ | $W_1(B)$ C |
| T2: | $W_2(A)$ $W_2(B)$ C | |



## Έλεγχος Σειριοποιησιμότητας

Υπάρχει τρόπος να ελέγξουμε *αποδοτικά* αν ένα χρονοπρόγραμμα είναι σωστό, δηλαδή σειριοποιήσιμο βάσει συγκρούσεων;



## Έλεγχος Σειριοποιησιμότητας

Γράφος προήγησης (precedence graph) ή γράφος σειριοποιησιμότητας (serialization graph)

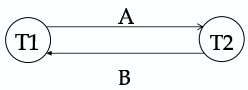
Κόμβος :: Δοσοληψία

Ακμή $T_i \rightarrow T_j$ αν μια πράξη της $T_i$ προηγείται μιας συγκρουόμενης πράξης της $T_j$



## Έλεγχος Σειριοποιησιμότητας

| | | |
|---|---|---|
| T1: | $R_1(A)$ $W_1(A)$, | $R_1(B)$ $W_1(B)$ |
| T2: | $R_2(A)$ $W_2(A)$ $R_2(B)$ | $W_2(B)$ |




*Η ετικέτα στην ακμή δείχνει σε πιο δεδομένο συγκρούονται*

S1: $R_1(X)$ $R_2(X)$ $W_1(X)$ $R_1(Y)$ $W_2(X)$ $C_2$ $W_1(Y)$ $C_1$

S2: $R_1(X)$ $R_2(X)$ $W_2(X)$ $C_2$ $W_1(X)$ $R_1(Y)$ $W_1(Y)$ $C_1$

*Γράφοι:*





Θεώρημα

**Ένα χρονοπρόγραμμα είναι σειριοποιήσιμο (βάσει συγκρούσεων) αν και μόνο αν ο γράφος προήγησής του είναι ακυκλικός.**





**χρονοπρογράμματα ισοδύναμα με σειριακά αφήνουν τη βάση δεδομένων σε συνεπή κατάσταση**

Δυο είδη σειριοποιησιμότητας

- Με βάση όψεις (πιο γενική, αλλά δεν υπάρχει λογαριθμικός αλγόριθμος)
- Με βάση συγκρούσεις (υπάρχει αλγόριθμος (γράφος συνδρομικότητας)





χρονοπρογράμματα ισοδύναμα με σειριακά αφήνουν τη βάση δεδομένων σε συνεπή κατάσταση

Αρκεί;

Όχι, στην περίπτωση αποτυχιών

Θα δούμε κάποιες ιδιότητες που το εξασφαλίζουν





- Δυο χρονοπρογράμματα S1 και S2 είναι **ισοδύναμα όψεων** αν:
  - Αν στο S1, η Ti διαβάζει την αρχική τιμή του A, τότε η Ti επίσης διαβάζει την αρχική τιμή του A στο S2
  - Αν στο S1, η Ti διαβάζει την τιμή του A που έγραψε η Tj, τότε η Ti διαβάζει την τιμή του A που έγραψε η Tj και στο S2
  - Αν στο S1, η Ti γράφει την τελική τιμή του A, τότε η Ti γράφει την τελική τιμή του A και στο S2

| | | |
|---|---|---|
| T1: R(A) | W(A) | |
| T2: | W(A) | |
| T3: | | W(A) |

| | | |
|---|---|---|
| T1: R(A),W(A) | | |
| T2: | W(A) | |
| T3: | | W(A) |





ζητούμενο: όταν μια δοσοληψία επικυρωθεί δεν θα χρειαστεί ποτέ να ανακληθεί

## Χρονοπρογράμματα με δυνατότητα ανάκαμψης

αν καμιά δοσοληψία T στο S δεν επικυρώνεται έως ότου επικυρωθούν όλες οι δοσοληψίες οι οποίες τροποποίησαν ένα δεδομένο που διαβάζει η T

Χρονοπρογράμματα με δυνατότητα ανάκαμψης: καμιά δοσοληψία T στο S δεν επικυρώνεται έως ότου επικυρωθούν όλες οι δοσοληψίες οι οποίες τροποποίησαν ένα δεδομένο που διαβάζει η T

$R_1(X)\ W_1(X)\ R_2(X)\ R_1(Y)\ W_2(X)\ C_2\ A_1$

$R_1(X)\ W_1(X)\ R_2(X)\ R_1(Y)\ W_2(X)\ W_1(Y)\ C_1\ C2$





• πρόβλημα: διαδιδόμενη ανάκληση (όταν μια δοσοληψία πρέπει να ανακληθεί γιατί διάβασε κάποιο στοιχείο από μια δοσοληψία που απέτυχε)

Χρονοπρογράμματα που αποφεύγουν τη διάδοση ανακλήσεων

αν κάθε δοσοληψία T στο S διαβάζει μόνο στοιχεία που έχουν γραφεί από επικυρωμένες δοσοληψίες





Αυστηρά Χρονοπρογράμματα

οι δοσοληψίες δεν μπορούν ούτε να διαβάσουν ούτε να γράψουν ένα στοιχείο X έως ότου επικυρωθεί η δοσοληψία που έγραψε το X

Δυνατότητα επιστροφής στην before image

$W_1(X, 5)\ W_2(X, 9)\ A_1$





Ορίσαμε το πρόβλημα

στο επόμενο μάθημα::

Τεχνικές Ελέγχου Συνδρομικότητας

στο μεθεπόμενο μάθημα::

Τεχνικές Ανάκαμψης από Σφάλματα