

## Επεξεργασία Δοσοληψιών (συνέχεια)

#### Πρόβλημα

«Σωστή» εκτέλεση προγραμμάτων όταν επιτρέπουμε *ταυτοχρονισμό* και ακόμα και αν υπάρχουν *αποτυχίες*

#### Δοσοληψία (transaction)

Η εκτέλεση ενός προγράμματος σε μια βάση δεδομένων. Μια ακολουθία από πράξεις R, W που τελειώνει ή με Commit ή με Abort

- **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

#### 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

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

#### Ισοδύναμα Χρονοπρογράμματα βάσει Συγκρούσεων:

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

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

## Επανάληψη: Χρονοπρογράμματα και Δυνατότητα Ανάκαμψης

### • Χρονοπρογράμματα με δυνατότητα ανάκαμψης (recoverable)

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

### • Χρονοπρογράμματα χωρίς διάδοση ανακλήσεων (avoids cascading aborts)

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

### • Αυστηρά Χρονοπρογράμματα (strict)

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

## Τεχνικές Ελέγχου Συνδρομικότητας

- Ο χρήστης δεν ασχολείται με τη συνδρομικότητα
- Το ΣΔΒΔ εξασφαλίζει «αυστή συνδρομικότητα»: *δρομολογεί* τις πράξεις των δοσοληψιών ώστε να προκύπτουν χρονοπρογράμματα σειριοποιημένα βάσει συγκρούσεων
- Μέσω τεχνικών ελέγχου συνδρομικότητας

## Τεχνικές Ελέγχου Συνδρομικότητας

### Τεχνικές

1. **Κλειδώματος (locking)** για να αποτρέψουν τη συνδρομική (ταυτόχρονη) προσπέλαση των δεδομένων από πολλές δοσοληψίες
2. **Διάταξης χρονοσημάτων (timestamps)**
3. **Πιστοποίησης (validation)** μιας δοσοληψίας (αισιόδοξα πρωτόκολλα)

## Τεχνικές Κλειδώματος

Δύο ειδών κλειδιά:

- **διαμοιραζόμενο (shared)** κλειδί ή κλειδί ανάγνωσης
- **αποκλειστικό (exclusive)** κλειδί ή κλειδί εγγραφής

S-Lock(X)      S-Lock(X)      X-Lock(X)  
X-Lock(X)      ✓

### Πρωτόκολλο κλειδώματος δυο φάσεων (Two-Phase Locking 2PL)

Όλες οι πράξεις (αιτήσεις) κλειδώματος μιας δοσοληψίας προηγούνται της πρώτης πράξης (αίτησης) άρσης κλειδώματος της διαδικασίας

- Δυνατή η «αναβάθμιση» κλειδιών

## Τεχνικές Κλειδώματος

- Κάθε δοσοληψία δυο φάσεις
  - μια φάση επέκτασης ή εξάπλωσης
  - μια φάση συρρικνώσεως

• Αποδεικνύεται ότι είναι ασύστο: παράγει χρονοπρογράμματα σειριοποιημένα με συγκρούσεις. Υπάρχουν σειριοποιημένα με συγκρούσεις χρονοπρογράμματα που δεν παράγονται με κλειδωμά δυο φάσεων;

• Παράγεται αυστηρό χρονοπρόγραμμα: Όχι.

Παραλλαγές:

**Αυστηρό Κλειδωμά Δύο Φάσεων:** μια δοσοληψία δεν απελευθερώνει κανένα αποκλειστικό κλειδί πριν επικυρωθεί ή ακυρωθεί

**Υπερ-αυστηρό Κλειδωμά Δύο Φάσεων:** μια δοσοληψία δεν απελευθερώνει κανένα (αποκλειστικό και διαμοιραζόμενο) κλειδί πριν επικυρωθεί ή ακυρωθεί

Τα πιο δημοφιλή πρωτόκολλα - ευρεία χρήση σε εμπορικά συστήματα

## Τεχνικές Κλειδώματος

- Πρόβλημα με τη δημιουργία αδιεξόδων
- Κόστος διαχείρισης κλειδιών

Δυο τεχνικές:

- **Πρωτόκολλα Πρόληψης Αδιεξόδων (Deadlock Prevention):** Αποφυγή δημιουργίας αδιεξόδου

- όλα τα κλειδιά δίνονται μαζί
- διάταξη των δοσοληψιών (με χρήση χρονοσημών)

- **Πρωτόκολλα Ανίχνευσης Αδιεξόδου (Deadlock Detection):** Ελέγχουμε περιοδικά αν το σύστημα βρίσκεται σε κατάσταση αδιεξόδου (π.χ με χρήση γράφου αναμονής)

## Διάταξη Χρονοσημάτων

Το χρονοσημα δημιουργείται από το ΣΔΒΔ και προσδιορίζει μοναδικά μια δοσοληψία

**Ιδέα: διάταξη των δοσοληψιών με βάση το χρονοσημα τους** (δηλαδή, χρονοπρόγραμμα ισοδύναμο με σειριακό στο οποίο οι δοσοληψίες εμφανίζονται διατεταγμένες με βάση τις τιμές των χρονοσημάτων)

⇒ άρα η σειρά προπέλασης στα δεδομένα πρέπει να μη παραβιάζει τη σειριοποιησιμότητα

Δηλαδή: αν μια πράξη  $a_i$  μιας δοσοληψίας  $T_i$  συγκρούεται με μια πράξη  $a_j$  μιας δοσοληψίας  $T_j$  και  $TS(T_i) < TS(T_j)$ , τότε η  $a_i$  πρέπει να προηγείται της  $a_j$ . Αλλιώς, restart τη δοσοληψία.

## Διάταξη Χρονοσημάτων

Κάθε δεδομένο  $X$  έχει δύο τιμές χρονοσημάτων:

$XS(X)$  (χρονοσημα ανάγνωσης) το μεγαλύτερο μεταξύ όλων των χρονοσημάτων των δοσοληψιών που διάβασαν το  $X$  (δισκθητικά, η πιο πρόσφατη που το διάβασε)

$XSE(X)$  (χρονοσημα εγγραφής) το μεγαλύτερο μεταξύ όλων των χρονοσημάτων των δοσοληψιών που έγραψαν το  $X$

Κάθε πράξη ανάγνωσης και εγγραφής μιας δοσοληψίας συνοδεύεται από κατάλληλους ελέγχους του χρονοσημάτων της και των χρονοσημάτων του αντίστοιχου δεδομένου.

Αν ο έλεγχος αποτύχει, η δοσοληψία απορρίπτεται και αρχίζει πάλι

## Διάταξη Χρονοσημάτων

- Παράγει ασιατά χρονοπρογράμματα
- Δε δημιουργούνται αδιέξοδα (αφού καμία δοσοληψία δεν περιμένει)
- Μπορεί να παραχθούν χρονοπρογράμματα που δεν έχουν δυνατότητα ανάκαμψης (not recoverable)

Παραλλαγή για χρονοπρόγραμμα χωρίς διάδοση ανακλήσεων (ιδέα: commit bit με κάθε δεδομένο - καθυστέρηση αναγνώσεων)

Παραλλαγή για ασιατρά (αμανοή αναγνώσεων και εγγραφών)

Άλλη ιδέα ( $R(X)$  ή  $W(X)$  από  $T$  με  $XS(T) > XSE(X)$ ) (αν μικρότερο ακυρώνεται) περιμένει μέχρι να επικυρωθεί ή να ακυρωθεί η δοσοληψία που έγραψε το  $X$  (όχι αδιέξοδα)

Επιβάρυνση διαχείρισης χρονοσημάτων

Απόρριψη και επανεκκίνηση δοσοληψιών

## Αισιόδοξες Τεχνικές

Οι τεχνικές κλειδώματος είναι συντηρητικές (αποφεύγονται οι συγκρούσεις)

### Μειονεκτήματα

- επιβάρυνση (overhead) χειρισμού κλειδώματος
- αποφυγή/ανίχνευση αδιεξόδων
- lock contention για τα δεδομένα που χρησιμοποιούνται συχνά

Αν οι συγκρούσεις είναι σπάνιες, μεγαλύτερη συγχρονικότητα, αν αντί για κλειδωμα, έλεγχος για συγκρούσεις όταν μια δοσοληψία επικυρώνεται (commits)

## Αισιόδοξες Τεχνικές

Κάθε δοσοληψία έχει τρεις φάσεις

- **ΑΝΑΓΝΩΣΗ**: η δοσοληψία διαβάζει από τη βδ, αλλά τροποποιεί προσωπικά αντίγραφα των δεδομένων
- **ΠΙΣΤΟΠΟΙΗΣΗ**: έλεγχος για συγκρούσεις
- **ΕΓΓΡΑΦΗ**: γράφει τα τοπικά αντίγραφα στη βδ

Κατά την πιστοποίηση μιας δοσοληψίας ελέγχουμε τα σύνολα

**ReadSet(T)**: το σύνολο των δεδομένων που διάβασε η  $T$

**WriteSet(T)**: το σύνολο των δεδομένων που διάβασε η  $T$

με τα αντίστοιχα σύνολα όλων των ταυτόχρονα εκτελουμένων δοσοληψιών του συστήματος

## Πιστοποίηση

Έλεγχος συνθηκών που είναι ικανές για να εξασφαλίσουν ότι δεν υπήρχαν συγκρούσεις

- Κάθε δοσοληψία  $T$  έχει ένα μοναδικό αριθμό TID (χρονοσημα)
- Το TID ανατίθεται στο τέλος της φάσης ΑΝΑΓΝΩΣΗΣ (ακριβώς πριν αρχίσει η πιστοποίηση)
- Με κάθε δοσοληψία

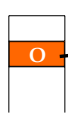
**ReadSet(T)**: το σύνολο των δεδομένων που διάβασε η  $T$

**WriteSet(T)**: το σύνολο των δεδομένων που διάβασε η  $T$

## Πολλαπλές Εκδόσεις

Κάθε εγγραφή παράγει ένα καινούργιο αντίγραφο ενώ οι αναγνώσεις διαβάζουν ένα κατάλληλο παλιό

ΚΥΡΙΟ ΤΜΗΜΑ  
(Τρέχουσες εκδόσεις των αντικειμένων της ΒΔ)



ΣΥΛΛΟΓΗ ΕΚΔΟΣΕΩΝ  
VERSION POOL  
(Παλιές εκδόσεις που μπορεί να είναι χρήσιμες σε κάποιους ενεργούς αναγνώστες)



Οι αναγνώστες μπορούν πάντα να προχωρήσουν -- αλλά μπορεί να χρειαστεί να περιμένουν μέχρι την επικύρωση των εγγραφών

Διατήρηση παλαιότερων εκδοχών, χρήσιμη ως ιστορικό εξέλιξης των τιμών (χρονικές (temporal) βάσεις δεδομένων)

## Πολλαπλές Εκδόσεις

Βασική ιδέα:

οι αναγνώστες διαβάζουν κατάλληλες εκδόσεις ώστε να διατηρήσουν τη σειριοποιησιμότητα

Παραλλαγές των τεχνικών ελέγχου συνδρομικότητας

Θα δούμε με χρονοσχήματα

## Πολλαπλές Εκδόσεις και Χρονοσχήματα

- Κάθε έκδοση ενός αντικειμένου έχει ως  $XΣE$  το  $XΣ$  της  $T$  που την έγραψε και ως  $XΣA$  το  $XΣ$  της  $T$  που διάβασε αυτήν την έκδοση πιο πρόσφατα

Μια πράξη εγγραφής μιας δοσοληψίας  $T$  δημιουργεί μια καινούργια έκδοση του δεδομένου, με  $XΣA$  και  $XΣE$  ίσο με  $XΣ(T)$

- Οι εκδόσεις συνδέονται (chained backward) μπορούμε να σρήσουμε εκδόσεις που είναι πολύ «παλιές για να έχουν ενδιαφέρον»

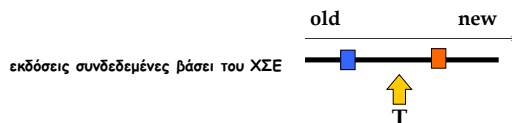
- Κάθε δοσοληψία: **Reader** ή **Writer**.
  - Writer *μπορεί* να γράφει κάποιο αντικείμενο; Reader *μόνο* διαβάσει.
  - Κάθε δοσοληψία δηλώνει αν είναι Reader όταν ξεκινά.

Στόχος είναι η Readers να μην αποτυγχάνουν

## Πολλαπλές Εκδόσεις και Χρονοσχήματα

Reader  $T$

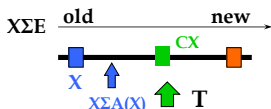
- Για την ανάγνωση ενός αντικειμένου:
  - Βρες την **πιο νεώτερη έκδοση με  $XΣE < XΣ(T)$**  (Η αναζήτηση ξεκινά από την πιο πρόσφατη - κοιτά σχήμα)
  - Αν υποθέσουμε ότι όλες οι εκδόσεις υπάρχουν, μια Reader  $T$  δε χρειάζεται να επανηχηθεί.



## Πολλαπλές Εκδόσεις και Χρονοσχήματα

Writer  $T$

- Για την ανάγνωση, ακολουθήσε το πρωτόκολλο του Reader
- Για να γράψεις ένα αντικείμενο  $X$ :
  - Βρες τη νεώτερη έκδοση με  $XΣE(X) < XΣE(T)$
  - Αν  $XΣA(X) < XΣ(T)$ , η  $T$  δημιουργεί ένα αντίγραφο  $CX$  του  $X$ , με ένα δείκτη στο  $X$ , και  $XΣE(CX) = XΣ(T)$ ,  $XΣA(CX) = XΣ(T)$ . (Οι εγγραφές γίνονται buffered μέχρι την επικύρωση της  $T$ , οι άλλες δοσοληψίες μπορούν να δουν τις τιμές των  $XΣ$  αλλά δεν μπορούν να διαβάσουν την έκδοση  $CX$  μέχρι η  $T$  να επικυρωθεί.)
- Αλλιώς, απόρριψη.



## Δυναμικές Βάσεις Δεδομένων

Αν επιτρέψουμε εγγραφές και διαγραφές στοιχείων, τα πρωτόκολλα δε δουλεύουν σωστά (ούτε το αστρικό 2PL)

Sailors(sid, sname, rating, age)

Boats(bid, bname, color)

Reserves(sid, bid, day)

- $T1$  κλειδώνει όλες τις σελίδες που περιέχουν εγγραφές sailor με  $rating = 1$ , και βρίσκει τον πιο ηλικιωμένο (έστω,  $age = 71$ ).
- Μετά, η  $T2$  εισάγει ένα νέο sailor:  $rating = 1$ ,  $age = 96$ .
- $T2$  επίσης διαγράφει τον πιο ηλικιωμένο ναυτικό με  $rating = 2$  (έστω,  $age = 80$ ), και επικυρώνεται.
- $T1$  τώρα κλειδώνει όλες τις σελίδες που περιέχουν εγγραφές sailor με  $rating = 2$ , βρίσκει τον πιο ηλικιωμένο (έστω,  $age = 63$ ).

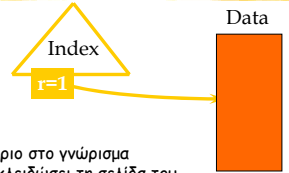
- Δεν υπάρχει ισοδύναμη με την παραπάνω σειριακή εκτέλεση

## Δυναμικές Βάσεις Δεδομένων

- Η T1 έμμεσα υποθέτει ότι έχει κλειδώσει το σύνολο όλων των εγγραφών με  $rating = 1$ .
  - Η υπόθεση ισχύει μόνο αν δεν προστεθούν εγγραφές ενώ εκτελείται η T1
  - Χρειάζεται κάποιος μηχανισμός για να το επιβάλει: κλειδωμά ευρετηρίου (index locking) και κλειδωμά συνθηκών (predicate locking).

Τα παράδειγμα δείχνει ότι η σειριοποιησιμότητα βάσει συγκρούσεων δίνει σειριοποιησιμότητα μόνο όταν τα αντικείμενα στη βδ είναι σταθερά

## Δυναμικές Βάσεις Δεδομένων



### Κλειδωμά Ευρετηρίου

- Αν υπάρχει ένα πυκνό ευρετήριο στο γνώρισμα  $rating$ , η T1 θα μπορούσε να κλειδώσει τη σελίδα του ευρετηρίου που περιέχει τις εγγραφές με  $rating = 1$ .
  - Αν δεν υπάρχει καμία εγγραφή με  $rating = 1$

## Δυναμικές Βάσεις Δεδομένων

### Κλειδωμά Συνθήκης

- Κλειδώσε όλες τις εγγραφές που ικανοποιούν κάποια συνθήκη λογικής, π.χ.,  $age > 2 * salary$ .
- Το κλειδωμά ευρετηρίου είναι μια ειδική μορφή κλειδώματος συνθήκης που το ευρετήριο επιτρέπει αποδοτική υλοποίηση του κλειδώματος συνθήκης

## Κλειδωμά Πολλαπλής Κλιμάκωσης

### Ιεραρχική δομή

Νέα είδη κλειδίων IS (πρόθεση διαμοιραζόμενου) IX (πρόθεση αποκλειστικό)

Για να κλειδώσει με S (X) πρέπει να κλειδώσει όλους τους προγόνους με IS (IX) αρχίζοντας από τη ρίζα

Συχνά IX και S (διάβασμα όλου, τροποποίηση ενός) είδος κλειδιού SIX

	IS	IX	S	SIX	X
IS	T	T	T	T	
IX	T	T			
S	T		T		
SIX	T				
X					

## Κλειδωμά Πολλαπλής Κλιμάκωσης

Δύο φάσεις  
Σειρά αποδέσμευσης κλειδίων;

## Υποστήριξη δοσοληψιών στην SQL-92

Δοσοληψία αυτόματα με προτάσεις που τροποποιούν είτε τη βάση δεδομένων είτε το σχήμα

Μέθοδος πρόσβασης: READ ONLY ή READ WRITE

Επίπεδο απομόνωσης READ UNCOMMITTED, READ COMMITTED, REPEATABLE READ, SERIALIZABLE (αυστηρό 2PL)

SET TRANSACTION ISOLATION LEVEL SERIALIZABLE READ ONLY  
default: SERIALIZABLE, READ WRITE

## Υποστήριξη δσοληψιών στην SQL-92

Isolation Level	Dirty Read	Unrepeatable Read	Phantom Problem
Read Uncommitted	Maybe	Maybe	Maybe
Read Committed	No	Maybe	Maybe
Repeatable Reads	No	No	Maybe
Serializable	No	No	No

Αναλυτικά στο βιβλίο

## Υποστήριξη δσοληψιών

IBM DB2, Informix, Microsoft SQL server, Sybase Αυστηρό 2PL (με παραλλαγές για επίπεδα απομόνωσης λιγότερα αυστηρά του SERIALIZABLE)

Microsoft SQL server: χρονόσημα εγγραφής

Oracle 8: πολλαπλές εκδόσεις, ώστε οι αναγνώσεις δεν περιμένουν ποτέ

Κλειδίωμα πολλαπλής κλιμάκωσης σε επίπεδο πίνακα, σελίδας και εγγραφής