



# ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ

---

## Βάσεις Δεδομένων II

### Τεχνικές Ανάνηψης - ARIES

Μανώλης Μαραγκουδάκης

Τμήμα Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων

---



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



## Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



## Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Αιγαίου**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Πανεπιστήμιο Αιγαίου-  
Τμήμα Μηχανικών  
Πληροφοριακών και  
Επικοινωνιακών Συστημάτων



## Βάσεις Δεδομένων II

**Ενότητα 3β-Τεχνικές Ανάνηψης-ARIES**  
Μανώλης Μαραγκουδάκης

[www.icsd.aegean.gr/mmarag](http://www.icsd.aegean.gr/mmarag)

# ARIES

- Algorithm for Recovery and Isolation Exploiting Semantics
- Δουλεύει με
  - Steal
  - No-force
- 3 φάσεις
  - Ανάλυση
  - REDO
  - UNDO
- Βασικές αρχές
  - Write-ahead logging
  - Επανάληψη ιστορικού στη φάση REDO
  - Logging όλων των αλλαγών στη φάση UNDO

## Ημερολόγιο Συστήματος (Log)

Σειριακή εγγραφή στο log

Για κάθε εγγραφή στοιχείου, στο log μια καταχώρηση:

**<XID, pageID, offset, length, old data, new data>**

ID  
δοσοληψίας

αριθμός  
σελίδας

θέση  
στη  
σελίδα

μήκος  
παλιά  
τιμή  
(undo)

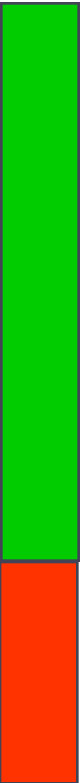
νέα τιμή  
(redo)

- Κάθε καταχώρηση στο log έχει ένα μοναδικό αριθμό **Log Sequence Number (LSN)**.
  - LSNs πάντα αυξάνουν.
- Το log γράφεται στο δίσκο. Το σύστημα κρατά τη μεταβλητή **flushedLSN**: το μέγιστο LSN που γράφτηκε στο δίσκο μέχρι τώρα

flushedLSN

"Log tail"  
στην RAM

Καταχωρήσεις  
του log ήδη  
flushed στο δίσκο

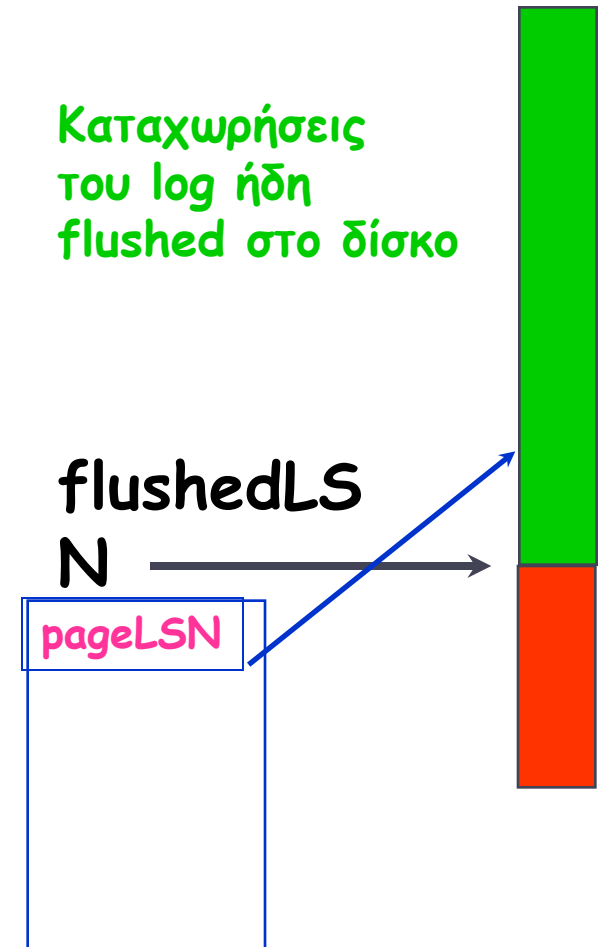


- Κάθε σελίδα δεδομένων περιέχει έναν **pageLSN**: το πιο πρόσφατο LSN για εγγραφή σε αυτή τη σελίδα

- WAL: Πριν γραφτεί μια σελίδα,  $\text{pageLSN} \leq \text{flushedLSN}$

*WAL(1) Η καταχώρηση στο log για μια εγγραφή γράφεται στο δίσκο πριν οι αντίστοιχες σελίδες να γραφούν στο δίσκο*

Καταχωρήσεις του log ήδη flushed στο δίσκο



### Πεδία LogRecord:

Προηγούμενο  
LSN για τη  
δοσοληψία

→ prevLSN  
XID  
type

Μόνο για  
εγγραφές

pageID  
length  
offset  
before-image  
after-image

Τύποι καταχωρήσεων

- Update
- Commit
- Abort
- End
- Compensation Log Records (CLRs)
  - για πράξεις UNDO



- Πίνακας Δοσοληψιών:
  - Μια εγγραφή για κάθε ενεργή δοσοληψία
  - Πεδία εγγραφής:  
XID,  
status, (running/committed/aborted),  
lastLSN (τελευταίο LSN που αφορά τη δοσοληψία).

XID	status	lastLSN
-----	--------	---------

- Πίνακας Τροποποιημένων σελίδων:

- Μια εγγραφή για κάθε τροποποιημένη σελίδα στη μνήμη.
- Πεδία εγγραφής
  - `recLSN` -- LSN της καταχώρησης του log της πρώτης αλλαγής στη σελίδα

PID	recLSN
-----	--------

## ARIES: Η Γενική Εικόνα



### LogRecords

prevLSN  
XID  
type  
pageID  
length  
offset  
before-image  
after-image



### Σελίδες Δεδομένων

σε κάθε  
σελίδα  
pageLSN



### Πίνακας Δοσοληψιών

lastLSN  
status

### Πίνακας Τροποποιημένων Σελίδων

recLSN  
flushedLSN

## ARIES: Παράδειγμα



Πίνακας Δοσοληψιών

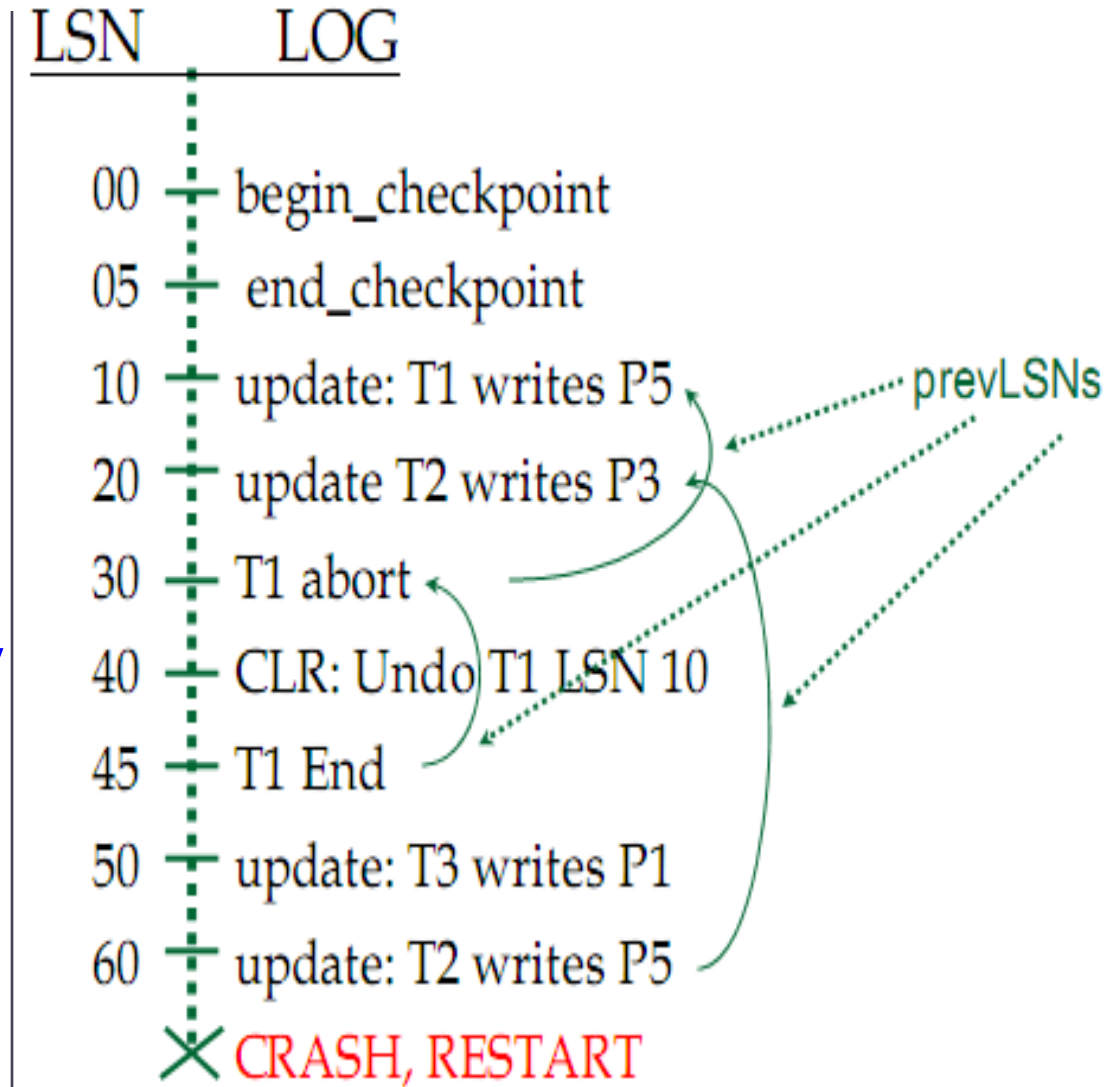
lastLSN

status

Πίνακας Τροποποιημένων  
Σελίδων

recLSN

flushedLSN



## ARIES: Ομαλή Εκτέλεση Δοσοληψίας

- Μια ακολουθία από *reads* & *writes*, που τελειώνουν με *commit* ή *abort*.
  - Υποθέτουμε ότι οι εγγραφές στο δίσκο είναι ατομικές
- Αυστηρό 2PL  $\Rightarrow$  αυστηρά χρονοπρογράμματα
- STEAL, NO-FORCE, με προεγγραφή log (Write-Ahead Logging)

## ARIES: Επικύρωση Δοσοληψίας

1. Γράψε μια καταχώρηση τύπου **commit** στο log
2. Γράψε στο δίσκο όλες τις καταχωρήσεις του log έως το `lastLSN` της δοσοληψίας

$$\text{flushedLSN} \geq \text{lastLSN (WAL)}$$

3. Τέλος του Commit
4. Γράψε μια καταχώρηση τύπου **end** στο log

## ARIES: Ακύρωση Δοσοληψίας

- Αρχικά, ας θεωρήσουμε ότι δε συμβαίνουν αποτυχίες

ΣΤΟΧΟΣ: UNDO (αναίρεση) των εγγραφών (updates)

## ARIES: Ακύρωση Δοσοληψίας

*«Αναπαράγουμε» το log της δοσοληψίας ανάποδα*

1. Γράφουμε στο log μια καταχώρηση τύπου **abort**
2. Παίρνουμε το **lastLSN** της δοσοληψίας από τον Πίνακα Δοσοληψιών
3. Ακολουθούμε την αλυσίδα των καταχωρήσεων στο log βάσει του πεδίου **prevLSN**



## ARIES: Ακύρωση Δοσοληψίας

- Για να εκτελέσουμε την αναίρεση πρέπει να έχουμε *κλειδί* στα δεδομένα
  - Πριν να γράψουμε την προηγούμενη τιμή σε μια σελίδα, γράφουμε στο log μια καταχώρηση τύπου **CLR**
- Η καταχώρηση **CLR** έχει ένα πεδίο **undonextLSN**: επόμενο LSN προς αναίρεση
- Μια καταχώρηση τύπου **CLR** δε χρειάζεται να αναιρεθεί (αλλά μπορεί να χρειαστεί να επαναληφθεί)
- Στο τέλος, γράψε μια καταχώρηση τύπου **end**

## ARIES: Σημεία Ελέγχου

Περιοδικά, το ΣΔΒΔ δημιουργεί ένα **σημείο ελέγχου (checkpoint)**, με σκοπό τη μείωση του χρόνου που χρειάζεται για ανάκαμψη στην περίπτωση αποτυχίας

- Γράφει τον Πίνακα Δοσοληψιών και τον Πίνακα Τροποποιημένων Σελίδων στο log
- Γράφει στο log:
  - μια εγγραφή **begin\_checkpoint**: που δηλώνει πότε άρχισε το chkrpt.
  - μια εγγραφή **end\_checkpoint**: που δηλώνει πότε τέλειωσε το chkrpt

- Ασαφές (fuzzy)
  - Οι δοσοληψίες εξακολουθούν να εκτελούνται. Οι πίνακες είναι ακριβείς ως προς το `begin_checkpoint`
  - Δεν γράφονται τροποποιημένες σελίδες στο δίσκο

Το LSN της εγγραφής του chkpt γράφεται σε ασφαλή χώρο (**master record**)

## Τρεις Φάσεις

**ΑΝΑΛΥΣΗ** -- ανακατασκευή των πινάκων του συστήματος: ποιες σελίδες είναι τροποποιημένες, ποια είναι η κατάσταση των δοσοληψιών

**ΕΠΑΝΑΛΗΨΗ (REDO)** -- επανάληψη όλων των δοσοληψιών

**ΑΚΥΡΩΣΗ (UNDO)** -- αναίρεση των δοσοληψιών που δεν ήταν επικυρωμένες

**ΑΝΑΛΥΣΗ** -- ανακατασκευή των πινάκων του συστήματος

(ΠΙΝΑΚΑΣ ΔΟΣΟΛΗΨΙΩΝ & ΠΙΝΑΚΑΣ ΤΡΟΠΟΠΟΙΗΜΕΝΩΝ ΣΕΛΙΔΩΝ)

χρησιμοποιώντας το τελευταίο *ολοκληρωμένο checkpoint*:

- ποια είναι η κατάσταση των δόσοληψιών από τότε;
- ποιες σελίδες είναι τροποποιημένες από τότε ;

## ARIES: Η Φάση της Ανάλυσης

- Αρχή από το τελευταίο checkpoint
- Βρίσκουμε το αντίστοιχο begin\_checkpoint

## ARIES: Η Φάση της Ανάλυσης

Διαβάζουμε το log *forward* ξεκινώντας από το **begin\_checkpoint**.

Αν βρούμε καταχώρηση τύπου:

**end**: σβήνουμε τη δοσοληψία από τον Πίνακα Δοσοληψιών

**άλλο τύπο**: εισαγωγή δοσοληψίας με **lastLSN = LSN**, αλλαγή κατάστασης αν **commit**

**update** (εγγραφή):

Αν η *P* δεν είναι στον Πίνακα Τροποποιημένων Σελίδων

εισαγωγή του *P* στον πίνακα με  
**recLSN=LSN**.

## ARIES: Η Φάση της Ανάλυσης

Πίνακας Δοσοληψιών: όλες οι δοσοληψίες που ήταν ενεργές τη στιγμή της αποτυχίας

Πίνακας Τροποποιημένων Σελίδων: όλες οι σελίδες που έχουν τροποποιηθεί (κάποιες μπορεί να έχουν γραφτεί ήδη στο δίσκο, μα δεν έχουμε αυτή τη πληροφορία στο log)

00 begin_checkpoint	T1 abort 70	P5 20
10 end_checkpoint	T3 running 60	P3 60
20 update: T1 writes P5	Πίνακας Δοσοληψιών	Πίνακας Τροποποιημένων Σελίδων
40 T2 Commit		
50 T2 end		
60 update: T3 writes P3		
70 T1 abort		



## Τρεις Φάσεις

ΑΝΑΛΥΣΗ --

ΕΠΑΝΑΛΗΨΗ (REDO) -- επανάληψη όλων των  
δοσοληψιών

ΑΚΥΡΩΣΗ (UNDO) -- αναίρεση των δοσοληψιών που  
δεν ήταν επικυρωμένες

## ARIES: Η Φάση της Επανάληψης

- Επαναλαμβάνουμε το log ώστε να φτάσουμε στην κατάσταση τη στιγμή της αποτυχίας
- Επαναλαμβάνουμε τις εγγραφές ακόμα και των δοσοληψιών που ακυρώθηκαν (aborted)

Προχωράμε στο log forward ξεκινώντας από το **μικρότερο recLSN** στον Πίνακα Τροποποιημένων Σελίδων

## ARIES: Η Φάση της Επανάληψης

Για κάθε καταχώρηση τύπου **update** ή **CLR**, επανέλαβε την πράξη εκτός αν:

- η σελίδα δεν είναι στον Πίνακα Τροποποιημένων Σελίδων
- η σελίδα είναι στον Πίνακα Τροποποιημένων Σελίδων αλλά  $recLSN > LSN$
- $pageLSN$  (στο δίσκο)  $\geq LSN$
- Σημείωση: δε γνωρίζουμε ποιες σελίδες έχουν γραφτεί στο δίσκο

## ARIES: Η Φάση της Επανάληψης

Για να επαναλάβεις μια πράξη:

- επανεκτέλεση
- θέσε  $pageLSN = LSN$
- δε χρειάζεται log

00 begin_checkpoint	T1 abort 70	P5 20
10 end_checkpoint	T3 running 60	P3 60
20 update: T1 writes P5	Πίνακας Δοσοληψιών	Πίνακας Τροποποιημένων Σελίδων
40 T2 Commit	Φάση της Επανάληψης (REDO)	
50 T2 end	Ξεκινάμε από το 20, γιατί;	
60 update: T3 writes P3		
70 T1 abort		

## ARIES: Ανάκαμψη από Αποτυχία

### Τρεις Φάσεις

ΑΝΑΛΥΣΗ --

ΕΠΑΝΑΛΗΨΗ (REDO) -- επανάληψη όλων των  
δοσοληψιών

**ΑΝΑΙΡΕΣΗ (UNDO)** -- αναίρεση των δοσοληψιών που  
δεν ήταν επικυρωμένες

## ARIES: Η Φάση της Αναίρεσης

Κατασκευή λίστας

$ToUndo = \{ lastLSN \text{ της } T \mid T \text{ δοσοληψία προς αναίρεση} \}$

δοσοληψίες προς αναίρεση: δοσοληψίες ενεργές κατά την αποτυχία

## ARIES: Η Φάση της Αναίρεσης

### Repeat:

Επέλεξε (και αφαίρεσε) το μεγαλύτερο LSN στην λίστα ToUndo.

If το LSN είναι τύπου **CLR** και `undonextLSN == NULL`

γράψε μια καταχώρηση τύπου `end` για τη δοσοληψία.

If το LSN είναι τύπου **CLR** και `undonextLSN != NULL`

Πρόσθεσε το `undonextLSN` στην λίστα ToUndo

Else if το LSN είναι τύπου **update**.

Undo το στάδιο του `update`,

Γράψε μια καταχώρηση τύπου **CLR**,

Πρόσθεσε το `prevLSN` στην λίστα ToUndo.

### Until ToUndo is empty.

```
00 begin_checkpoint
10 end_checkpoint
20 update: T1 writes P5
40 T2 Commit
50 T2 end
60 update: T3 writes P3
70 T1 abort
```

```
T1 abort 70 P5 20
```

```
T3 running 60 P3 60
```

Πίνακας Δοσοληψιών      Πίνακας Τροποποιημένων Σελίδων

Φάση της Επανάληψης (REDO)

Ξεκινάμε από το 20, γιατί;

Φάση της Αναίρεσης (UNDO)

ToUndo = {70, 60}

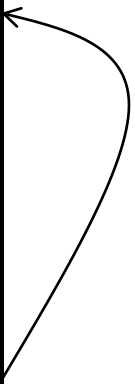
LSN	Type	Tid	PrevLSN	Data
5	BOT	3		
6	UP	1	3	C
7	BOT	2		
8	UP	2	7	A
9	EOT	1	6	
10	UP	3	5	B
11	UP	2	8	D
12	EOT	2	11	
13	UP	3	10	E

Losers:  $T_3$

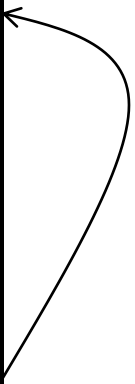


LSN	Type	Tid	PrevLSN	Data
5	BOT	3		
6	UP	1	3	C
7	BOT	2		
8	UP	2	7	A
9	EOT	1	6	
10	UP	3	5	B
11	UP	2	8	D
12	EOT	2	11	
13	UP	3	10	E

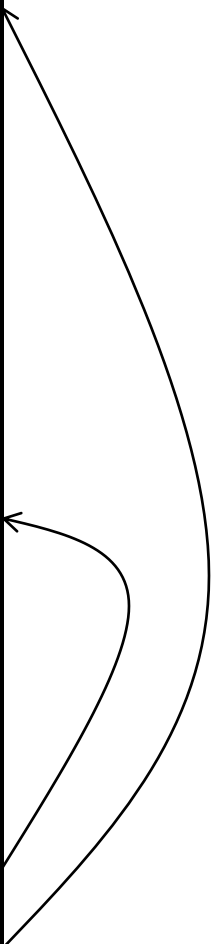
LSN	Type	Tid	PrevLSN	Data
5	BOT	3		
6	UP	1	3	C
7	BOT	2		
8	UP	2	7	A
9	EOT	1	6	
10	UP	3	5	B
11	UP	2	8	D
12	EOT	2	11	
13	UP	3	10	E
14	CLR	3	13	10



LSN	Type	Tid	PrevLSN	Data
5	BOT	3		
6	UP	1	3	C
7	BOT	2		
8	UP	2	7	A
9	EOT	1	6	
10	UP	3	5	B
11	UP	2	8	D
12	EOT	2	11	
13	UP	3	10	E
14	CLR	3	13	10

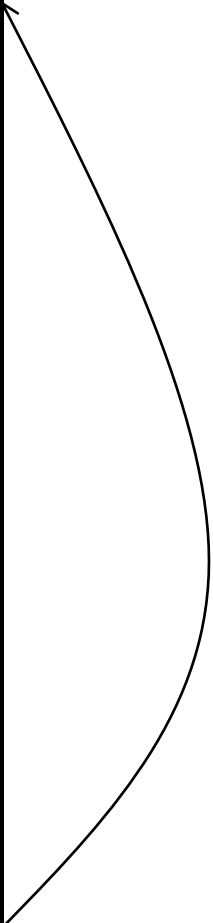


LSN	Type	Tid	PrevLSN	Data
5	BOT	3		
6	UP	1	3	C
7	BOT	2		
8	UP	2	7	A
9	EOT	1	6	
10	UP	3	5	B
11	UP	2	8	D
12	EOT	2	11	
13	UP	3	10	E
14	CLR	3	13	10
15	CLR	3	14	5

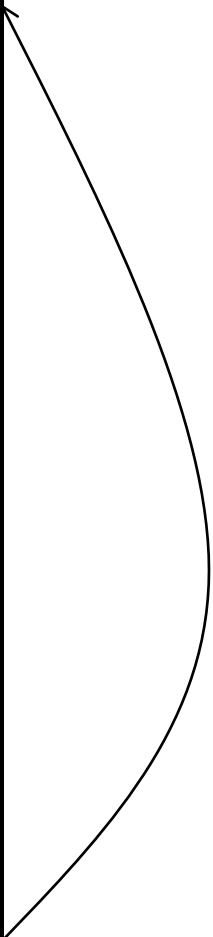


LSN	Type	Tid	PrevLSN	Data
5	BOT	3		
6	UP	1	3	C
7	BOT	2		
8	UP	2	7	A
9	EOT	1	6	
10	UP	3	5	B
11	UP	2	8	D
12	EOT	2	11	
13	UP	3	10	E
14	CLR	3	13	10
15	CLR	3	14	5

LSN	Type	Tid	PrevLSN	Data
5	BOT	3		
6	UP	1	3	C
7	BOT	2		
8	UP	2	7	A
9	EOT	1	6	
10	UP	3	5	B
11	UP	2	8	D
12	EOT	2	11	
13	UP	3	10	E
14	CLR	3	13	10
15	CLR	3	14	5



LSN	Type	Tid	PrevLSN	Data
5	BOT	3		
6	UP	1	3	C
7	BOT	2		
8	UP	2	7	A
9	EOT	1	6	
10	UP	3	5	B
11	UP	2	8	D
12	EOT	2	11	
13	UP	3	10	E
14	CLR	3	13	10
15	CLR	3	14	5
16	EOT	3	15	



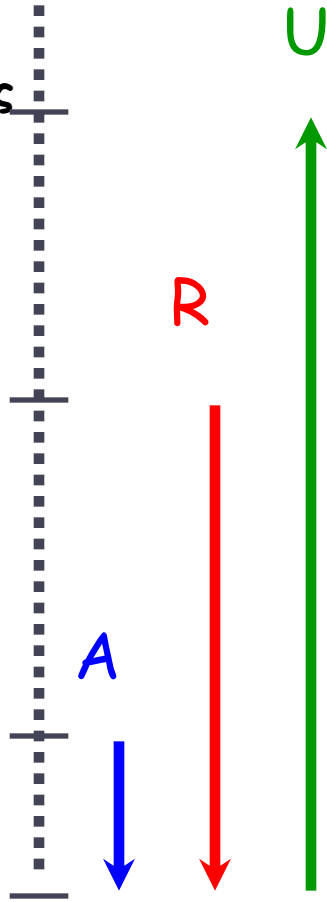
## ARIES: Ανάκαμψη από Αποτυχία

Παλιότερη  
καταχώρηση log  
δοσοληψίας ενεργής  
κατά την αποτυχία

Μικρότερο recLSN  
στον Πίνακα  
Τροποποιημένων  
Σελίδων μετά την  
Ανάλυση

Τελευταίο  
chkpt

ΑΠΟΤΥΧΙΑ



- ❖ Άρχισε από ένα σημείο ελέγχου (που βρίσκεται μέσω του **master record**).
- ❖ Τρεις Φάσεις.
  - Εύρεση ποιες δοσοληψίες μετά το σημείο ελέγχου επικυρώθηκαν, ποιες ακυρώθηκαν (**ΑΝΑΛΥΣΗ**).
  - **REDO** όλες τις δοσοληψίες.
    - ◆ (repeat history)
  - **UNDO** το αποτέλεσμα των αποτυχημένων δοσοληψιών



# Παράδειγμα

LSN		LOG
00	—	begin_checkpoint
10	—	end_checkpoint
20	—	update: T1 writes P1
30	—	update: T2 writes P2
40	—	update: T3 writes P3
50	—	T2 commit
60	—	update: T3 writes P2
70	—	T2 end
80	—	update: T1 writes P5
90	—	T3 abort
	×	CRASH, RESTART

# Παράδειγμα

LSN	LOG
00	begin_checkpoint
10	end_checkpoint
20	update: T1 writes P1
30	update: T2 writes P2
40	update: T3 writes P3
50	T2 commit
60	update: T3 writes P2
70	T2 end
80	update: T1 writes P5
90	T3 abort
	✗ CRASH, RESTART

LSN 20	Add (T1,20) to TT and (P1,20) to DPT
LSN 30	Add (T2,30) to TT and (P2,30) to DPT
LSN 40	Add (T3,40) to TT and (P3,40) to DPT
LSN 50	Change status of T2 to C
LSN 60	Change (T3,40) to (T3,60)
LSN 70	Remove T2 from TT
LSN 80	Change (T1,20) to (T1,70) and add (P5,70) to DPT
LSN 90	No action

LSN 20 | Check whether P1 has pageLSN more than 10 or not. Since it is a committed transaction, we probably need not redo this update.

LSN 30 | Redo the change in P2

LSN 40 | Redo the change in P3

LSN 50 | No action

LSN 60 | Redo the changes on P2

LSN 70 | No action

LSN 80 | Redo the changes on P5

LSN 90 | No action

LSN 80 | Undo the changes in P5. Append a CLR: Undo T1 LSN 80, set undonextLSN = 20. Add 20 to ToUndo.

ToUndo consists of (60, 20).

LSN 60 | Undo the changes on P2. Append a CLR: Undo T3 LSN 60, set undonextLSN = 40. Add 40 to ToUndo.

ToUndo consists of (40, 20).

LSN 40 | Undo the changes on P3. Append a CLR: Undo T3 LSN 40, T3 end

ToUndo consists of (20).

LSN 20 | Undo the changes on P1. Append a CLR: Undo T1 LSN 20, T1 end

# Τελικό Log μετά το Restart

LSN	LOG	LSN 00	begin_checkpoint	
00	begin_checkpoint	LSN 10	end_checkpoint	
10	end_checkpoint	LSN 20	update: T1 writes P1	
20	update: T1 writes P1	LSN 30	update: T2 writes P2	
30	update: T2 writes P2	LSN 40	update: T3 writes P3	
40	update: T3 writes P3	LSN 50	T2 commit	prevLSN = 30
50	T2 commit	LSN 60	update: T3 writes P2	prevLSN = 40
60	update: T3 writes P2	LSN 70	T2 end	prevLSN = 50
70	T2 end	LSN 80	update: T1 writes P5	prevLSN = 20
80	update: T1 writes P5	LSN 90	T3 abort	prevLSN = 60
90	T3 abort	LSN 100	CLR: Undo T1 LSN 80	undonextLSN= 20
×	CRASH, RESTART	LSN 110	CLR: Undo T3 LSN 60	undonextLSN= 40
		LSN 120,125	CLR: Undo T3 LSN 40	T3 end.
		LSN 130,135	CLR: Undo T1 LSN 20	T1 end.