Serializable

Conflict Serializability
(Διένεξη σειριοποιησιμότητας)

(Αν μια από τις εντολές των δυο συναλλαγών που εκτελούνται παράλληλα, είναι write τότε έχω conflict-διένεξη).

- If a schedule ***S*** **can be transformed into a schedule** ***S´*** by a series of swaps of non-conflicting instructions, we say that ***S*** **and** ***S´*** are **conflict equivalent (ισοδύναμα ως προς τις διενέξεις)**.
- We say that a **schedule** ***S*** is **conflict serializable (σειριοποιήσιμο ως προς τις διενέξεις)** if it is conflict equivalent to a **serial schedule**
- A schedule is conflict serializable **(σειριοποιήσιμο ως προς τις διενέξεις)** if and only if its precedence graph (γράφημα προτεραιότητας) is acyclic.

Recoverable – Cascadeless

Χρονοδιαγράμματα με δυνατότητα Αποκατάστασης / χωρίς διαδοχικές αναιρέσεις

- **Recoverable schedule** — if a transaction T_j reads a data item previously written by a transaction T_i, then the **commit operation of** T_i **(δηλ αυτής που έκανε το write) must appear before the commit operation of** T_j.
- Ένα χρονοδιάγραμμα δεν έχει δυνατότητα αποκατάστασης αν μια από τις συναλλαγές του έχει πάρει τιμές για κάποια δεδομένα που μια άλλη συναλλαγή τα έχει αναιρέσει.
- **Cascadeless schedules** — for each pair of transactions T_i and T_j such that T_j **reads a data item previously written by** T_i, the **commit operation of** T_i **(δηλ αυτής που έκανε το write) appears before the read operation of** T_j.
- Επομένως τα Cascadeless schedules είναι υποσύνολα των Recoverable schedules
- A database must provide a mechanism that will ensure that all possible schedules are both:
  - Conflict serializable (σειριοποιήσιμα ως προς τις διενέξεις).
  - Recoverable and preferably cascadeless (με δυνατότητα Αποκατάστασης και προτιμώνται χωρίς διαδοχικές αναιρέσεις)

The Two-Phase Locking Protocol
Πρωτόκολλο κλειδώματος Δυο Φάσεων

- This protocol **ensures conflict-serializable schedules** (χρονοδιαγράμματα σειριοποιήσιμα ως προς τις διενέξεις).

- Phase 1: Growing Phase (Φάση ανάπτυξης)
  - Transaction may obtain locks
  - Transaction **may not** release locks
- Phase 2: Shrinking Phase (φάση σύμπτυξης)
  - Transaction may release locks
  - Transaction **may not** obtain locks

It can be proved that the transactions can be serialized in the order of their lock points (σημεία κλειδώματος) (i.e., the point where a transaction acquired its final lock = this is the end of the growing phase of the transaction).

Two-phase locking ***does not*** **ensure freedom from deadlocks**.

Για την αποφυγή αδιεξόδου μπορεί να χρειαστεί να γίνει διαδοχική αναίρεση όταν ακολουθούμε το πρωτόκολλο κλειδώματος δυο φάσεων. Οι διαδοχικές αναιρέσεις μπορούν να αποφευχθούν με το αυστηρό (strict) πρωτόκολλο κλειδώματος δυο φάσεων (δηλ τα αποκλειστικά κλειδώματα να κρατούνται έως να εκτελεστεί η συναλλαγή δηλ να γίνει commit). (σελ 667).

Το ενδελεχές (rigorous) πρωτόκολλο κλειδώματος δυο φάσεων είναι ακόμα πιο αυστηρό καθώς επιβάλει όλα τα κλειδώματα να κρατούνται έως να εκτελεστεί η συναλλαγή δηλ να γίνει commit