Acquire (systemMutex)

System wide lock!

Lock based discipline for BOA services.

but; can cause problems - deadlock.

doesn't mean...

Locks give flexibility - c: transactional view

Two-phase commit

High level transactions

Rolling back Databases.

Block atomicity using Locks

Agenda

CSE 53 (Aml OS)
Look at the looks.

We'll remove the looks.

Look at the looks.

But looks, locked by the transactor.

Look at the looks.

Look at the looks.

Is it correct?

A common transactional example.

Each transaction must acquire a lock for every shared object if it intends to make a change.
Of course more concurrency, lock can be needed to be protected. This is after an object is not needed. The destruction can replace a lock only until it passes the lock point. The destruction may not replace the locks.

Cons: 4-tier thread.

Search phase:

Two-phase locking:

Can the performance be improved further?

Giving more concurrency, lock can be

- Consecutions with mutually exclusive
- User finer granularity locks

Simple locking:
Deadlock handling techniques

Deadlock avoidance

Deadlock detection

Deadlock prevention

Exit from statement

Deadlocks can occur

Serializable

Complete (serializable)

Not serializable

Deadlocks are necessary

Deadlocks are not necessary

Pessimistic

Optimistic

Preemptive

Non-preemptive

Cycle is necessary

Cycle is not necessary

Acquire lock

Avoid lock

Lock order

Deadlock avoidance
symptoms + breast cancer.
- else in background:
  - periodicity, can cycle detection
  - cycle detection
  -

- try

- upon time expiration, check
- if time for expiration + finish)
- else if it is initialized (expected
each item is associated with a time

- Time expiration
  Optimizations (cont)

- 1 2 3 4
- 0.9

- if it is accessed and
  - enough to snoop any
  - high-enough bucket
  - must have at least
  - by itself, the decryption
  -
procedure PAY-INTEREST (ty, acc)

if acc > 0 then

   \[ \text{procedure for biannual transactions} \]

   \[ \text{Adjustment for biannual transactions} \]

   \[ \text{condensation} \]

   \[ \text{exponential back calculation} \]

   \[ \text{break algorithm} \]

   \[ \text{transactions have expired and are in progress} \]

   \[ \text{lock} \]