# CA420 Databases II

Isolation and Locking in Practice



## Isolation in Practice

In practice:

- implementing full serialisability:
  - limits concurrency, and
  - requires substantial overhead
- many DBMSs also support weakened "*degrees of isolation*"



## Isolation – Four Problem Cases

Lost updates due to `undo`:

- $w_1[x] \rightarrow w_2[x] \rightarrow a_1$
- if before values are used for `undo`, $w_2[x]$ might be a lost update

Dirty reads:

- $w_1[x] \rightarrow r_2[x] \rightarrow c_2 \rightarrow c_1$
- $r_2[x]$ reads dirty – as yet uncommitted – data

Unrepeatable reads:

- $r_1[x] \rightarrow w_2[x] \rightarrow c_2 \rightarrow r_1[x]$
- reads on same item return different values

Lost updates due to interleaving:

- $r_1[x] \rightarrow r_2[x] \rightarrow w_2[x] \rightarrow c_2 \rightarrow w_1[x] \rightarrow c_1$



## Degrees of Isolation – Degree 0, "Chaos"

Degree 0 isolation:

- known as *chaos*
- locking is *well-formed* wrt. writes
- lets others run at higher levels of isolation

The following are valid degree 0 executions:

- $w_1[x] \rightarrow w_2[x] \rightarrow w_2[y] \rightarrow w_1[y] \rightarrow c_1 \rightarrow c_2$ (not CSR)
- $w_1[x] \rightarrow w_2[x] \rightarrow w_2[y] \rightarrow w_1[y] \rightarrow c_1 \rightarrow a_2$ (final value of $x$?)

Locking:

- locking is used only to protect individual write operations
- degree 0 transactions will wait only if a transaction operating at a higher degree of isolation holds a conflicting lock until commit

## Degrees of Isolation – Degree 1, "Browse"

Degree 1 isolation:

- known as *browse*
- locking is well-formed and *strict two-phase* wrt. writes
- no overwriting of dirty data
- no lost updates due to undo of degree 1 $T_i$:
  - $w_i[x] \rightarrow w_j[x] \rightarrow a_i$ (cannot happen)

Locking:

- for degree 1 $T_i$ and some item:
  - $w_i[x] \rightarrow \ldots \rightarrow c_i/a_i \rightarrow w_j[x]$ (for degree 0 $T_j$ or higher)
  - write → write



## Degrees of Isolation – Degree 2, "Cursor Stability"

Degree 2 isolation:

- known as *cursor stability*
- locking is well-formed wrt. reads and writes,
  and strict two-phase wrt. writes
- no lost updates due to undo
  no dirty reads from degree 1 or higher transactions

Locking:

- for degree 2 $T_i$ and some item:
  - as before, plus:
  - $w_j[x] \rightarrow \ldots \rightarrow c_j/a_j \rightarrow r_i[x]$ (for degree 1 $T_j$ or higher)
  - write → write and write → read



## Degrees of Isolation – Degree 3, "Repeatable Reads"

Degree 3 isolation:

- known as *repeatable reads*
- locking is strict two-phase wrt. both reads and writes
- no lost updates due to undo
  no dirty reads
  repeatable reads
  no lost updates due to interleaving with other degree 3 transactions

Locking:

- for degree 3 $T_i$ and some item:
  - as before, plus:
  - $r_i[x] \rightarrow \ldots \rightarrow c_i/a_i \rightarrow w_j[x]$ (for degree 0 $T_j$ or higher)
  - write → write, write → read and read → write
- i.e. strict two-phase locking



## Another Problem Case – Phantoms

| name | address | age |
|---|---|---|
| John | 1 The Mews | 27 |
| Bill | 8 Poplar Avenue | 22 |
| Mary | 14 The Ashes | 33 |

$T_1$

```
select * from people
where age > 25;
```

$T_2$

```
update people set age = 29
where name = 'Bill';
commit;
```

$T_1$

```
select * from people
where age > 25;
```

Phantoms:

- the first `select` returns `John` and `Mary`
- should the second `select` return `John`, `Mary` *and* `Bill`?

## Dealing with Phantoms

With cursor stability, phantoms are ok as transactions do not expect to read a transaction-consistent state

For repeatable reads:

- predicate locking
- next-key, next item locking:
  always obtain corresponding lock also on *next* item in table or index when inserting data or scanning a range of values



## More Information

More information:

- *A theoretical formulation of degrees of isolation in databases*, Vijayalakshmi Atluri, Elisa Bertino and Sushil Jajodia.

  `http://citeseer.ist.psu.edu/49154.html`

- *Transaction Processing: Concepts and Technologies*
  Jim Gray and Andreas Reuter
  Morgan Kaufmann, ISBN: 1-55860-190-2