Schedules: SERIAL and SERIALIZABLE

Definition of SCHEDULE

The sequence of the read/write operations of several transactions as they are executed in the Database
Serial Schedule

• Transactions execute fully.
• One at a time.
• No interleaving.
• Different orders of execution may produce different final values

Serializable Schedule

• Interleaved.
• Equivalent to SOME serial schedule.
• Equivalence does NOT mean "ending up with the same values as".
• Equivalence cannot depend on initial values of database items.
• Cannot depend on values written
  DB doesn’t know logic of transaction.
• Depends only on order of operations.
Conflicting Operations

• Used to define how schedules are equivalent
• 2 OPERATIONS CONFLICT if
  – belong to different transactions
  – access same data item
  – at least one is a write
• **IMPORTANT**: they do **NOT** have to **ACTUALLY** come into **CONFLICT**!
  – A better name would be ‘Potentially Conflicting Operations’

CONFLICT EQUIVALENCE

• 2 Schedules are Conflict Equivalent
• If the order of any 2 conflicting operations is the same in both schedules.
• SERIALIZABLE SCHEDULE is CONFLICT EQUIVALENT to some serial schedule
Example of Conflict Equivalence

• Transaction A and Transaction B each Read and Write X.
• Conflicting Operations:
  – Trans A Read X and Trans B Write X
  – Trans A Write X and Trans B Read X
  – Trans A Write X and Trans B Write X
• Trans A Read X and Trans B Read X do not conflict

Conflicting Operations

<table>
<thead>
<tr>
<th>TRANS A</th>
<th>TRANS B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read Y</td>
<td>Read X</td>
</tr>
<tr>
<td>Write Y</td>
<td>Read Z</td>
</tr>
<tr>
<td>Read X</td>
<td>Write Z</td>
</tr>
<tr>
<td>Write X</td>
<td>Write X</td>
</tr>
</tbody>
</table>
They are conflict equivalent!

Schedule 1
- TRANS A
  - Read Y
  - Write Y
  - Read X
  - Write X
- TRANS B
  - Read X
  - Read Z
  - Write Z
  - Write X

For each pair of conflicting ops, order is the same.

Schedule 2
- TRANS A
  - Read Y
  - Write Y
  - Read X
  - Write X
- TRANS B
  - Read X
  - Read Z
  - Write Z
  - Write X

Serial Sched 1
Green & Blue in opposite orders. The two are not conflict equivalent.

Is Schedule 3 Serializable?

Schedule 3
- TRANS A
  - Read Y
  - Read X
  - Write Y
  - Read Z
  - Write X
  - Write Z
- TRANS B
  - Read X
  - Read Z
  - Write Y
  - Write Z
  - Write X
We are not done!

- Schedule 3 is not equivalent to one serial schedule.
  Where Trans B precedes Trans A
- But it might be equivalent to another
  Where Trans A precedes Trans B
- To be serializable, it needs to be equivalent to only one serial schedule.

Is Schedule 3 Serializable? Again.

Schedule 3

<table>
<thead>
<tr>
<th>TRANS A</th>
<th>TRANS B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read Y</td>
<td>Read X</td>
</tr>
<tr>
<td></td>
<td>Read Z</td>
</tr>
<tr>
<td>Write Y</td>
<td></td>
</tr>
<tr>
<td>Read X</td>
<td></td>
</tr>
<tr>
<td>Write X</td>
<td></td>
</tr>
</tbody>
</table>

Serial Sched 2

<table>
<thead>
<tr>
<th>TRANS A</th>
<th>TRANS B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read Y</td>
<td>Read Y</td>
</tr>
<tr>
<td></td>
<td>Read Z</td>
</tr>
<tr>
<td>Write Y</td>
<td></td>
</tr>
<tr>
<td>Read X</td>
<td></td>
</tr>
<tr>
<td>Write X</td>
<td></td>
</tr>
</tbody>
</table>

Green in opposite orders. The two are not conflict equivalent
Conclusion

• Schedule 3 is not equivalent to serial schedule Trans A; Trans B.
• Schedule 3 is not equivalent to serial schedule Trans B; Trans A.
• There are no other Trans A/Trans B serial schedules.
• Schedule 3 is not serializable.

A shorter way

Schedule 3

TRANS A
Read Y
Write Y
Read X
Write X

TRANS B
Read X
Read Z
Write Z
Write X

Trans B Read X comes before Trans A Write X. If there is an equivalent serial schedule, it must be one where Trans B comes before Trans A.

Trans A Read X comes before Trans B Write X. If there is an equivalent serial schedule, it must be one where Trans A comes before Trans B.

So there can be no equivalent serial schedule.
A Graphical Representation

Schedule 3

TRANS A
- Read Y
- Write Y
- Read X
- Write X

TRANS B
- Read X
- Read Z
- Write Z
- Write X

Trans A Read X -- Trans B Write X

Trans B Read X -- Trans A Write X

Precedence Graph