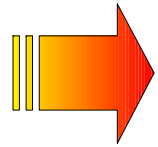
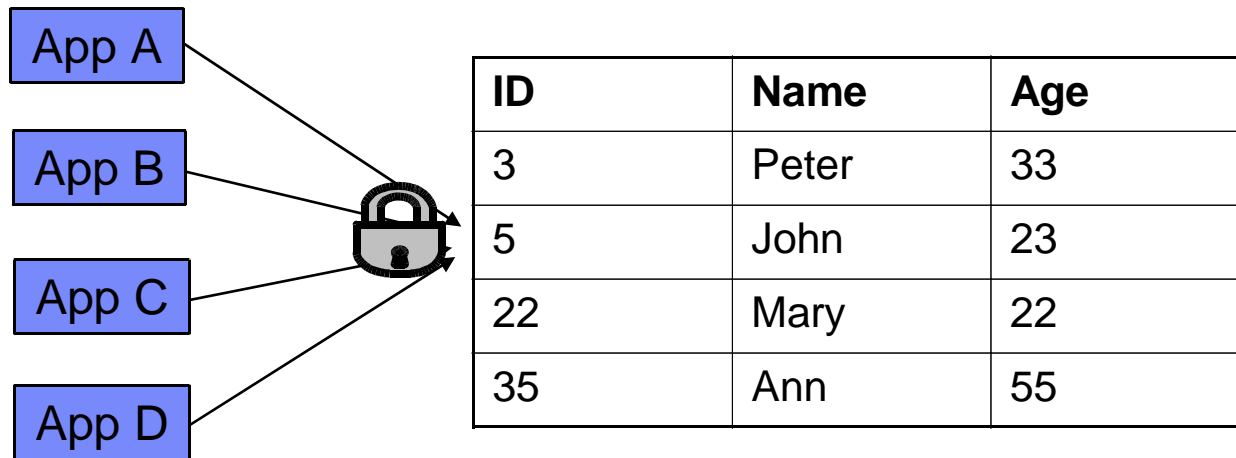


Agenda



- Transactions
- Concurrency & Locking
- Lock Wait
- Deadlocks

Concurrency and Locking



- Concurrency:
 - Multiple users accessing the same resources at the same time
- Locking:
 - Mechanism to ensure data integrity and consistency

Locking

- Locks are acquired automatically as needed to support a transaction based on “isolation levels”
- COMMIT and ROLLBACK statements release all locks
- Two basic types of locks:
 - Share locks (S locks) – acquired when an application wants to read and prevent others from updating the same row
 - Exclusive locks (X locks) – acquired when an application updates, inserts, or deletes a row

Problems if there is no concurrency control

- Lost update
- Uncommitted read
- Non-repeatable read
- Phantom read

Lost update

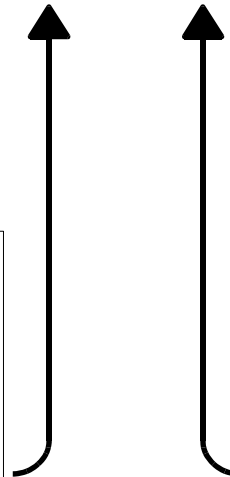
reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A



App B



Lost update

reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

**update reservations
set name = 'John'
where seat = '7C'**

App B

The diagram illustrates a lost update scenario. Two applications, App A and App B, are shown below a 'reservations' table. App A has executed an update query to set the name to 'John' for seat '7C'. App B is also shown, but its update is lost because it overwrites App A's change. Two arrows point from the update boxes of App A and App B to the 'name' column of the '7C' row in the table, indicating the update operation.

Lost update

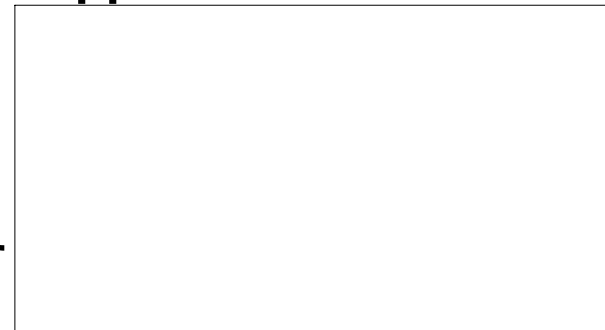
reservations

seat	name	...
7C	John	
7B	_____	
...		

App A

update reservations
set name = 'John'
where seat = '7C'

App B



Lost update

reservations

seat	name	...
7C	John	
7B	_____	
...		

App A

update reservations
set name = 'John'
where seat = '7C'

App B

update reservations
set name = 'Mary'
where seat = '7C'

Lost update

reservations

seat	name	...
7C	Mary	
7B	_____	
...		

App A

update reservations
set name = 'John'
where seat = '7C'

App B

update reservations
set name = 'Mary'
where seat = '7C'

Lost update

reservations

seat	name	...
7C	Mary?	
7B	_____	
...		

App A

update reservations
set name = 'John'
where seat = '7C'

App B

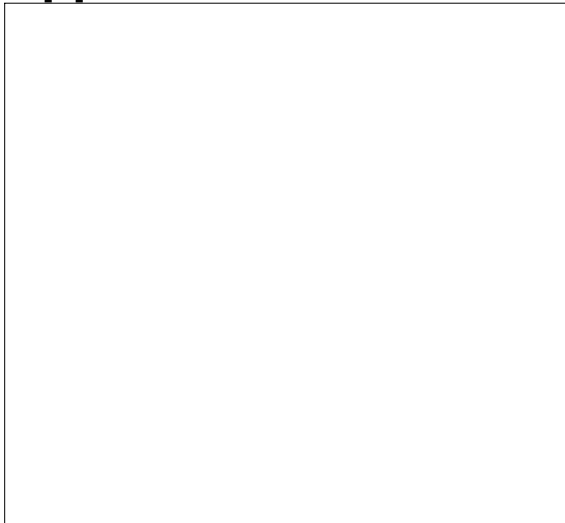
update reservations
set name = 'Mary'
where seat = '7C'

Uncommitted read (also known as “dirty read”)

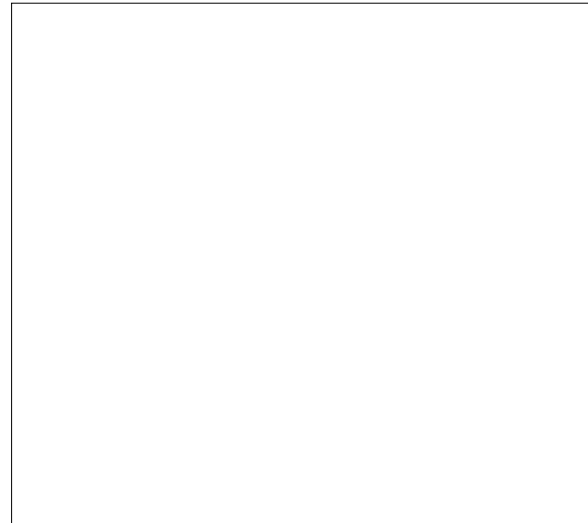
reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

A large, empty rectangular box representing the workspace for App A.

App B

A large, empty rectangular box representing the workspace for App B.

Uncommitted read (also known as “dirty read”)

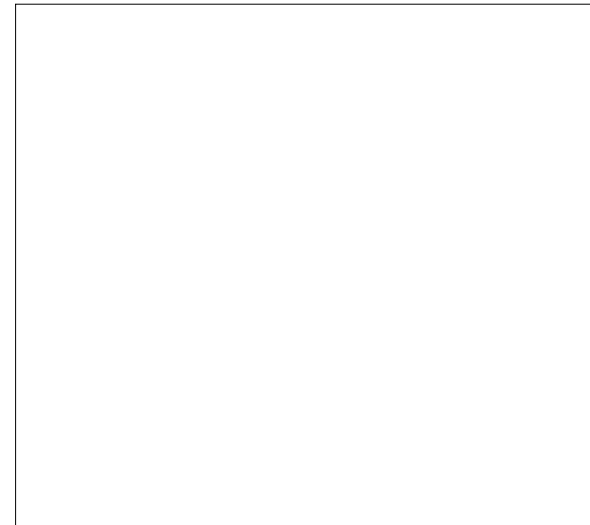
reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

**update reservations
set name = 'John'
where seat = '7C'**

App B



Uncommitted read (also known as “dirty read”)

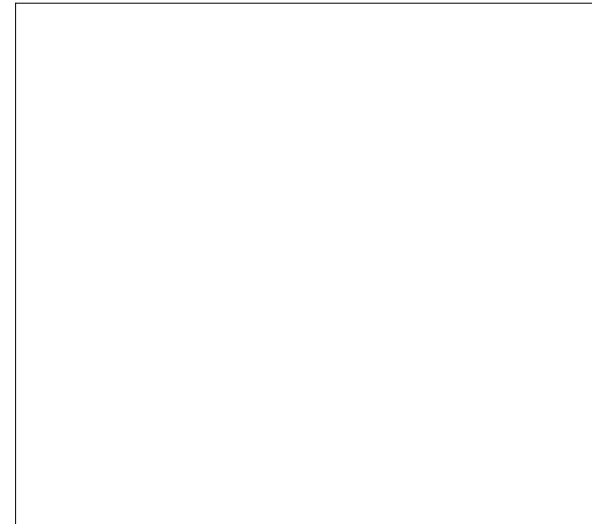
reservations

seat	name	...
7C	John	
7B	_____	
...		

App A

update reservations
set name = 'John'
where seat = '7C'

App B



Uncommitted read (also known as “dirty read”)

reservations

seat	name	...
7C	John	
7B	_____	
...		

App A

**update reservations
set name = 'John'
where seat = '7C'**

App B

**Select name
from reservations
where seat is '7C'**

Uncommitted read (also known as “dirty read”)

reservations

seat	name	...
7C	John	
7B	_____	
...		

App A

update reservations
set name = 'John'
where seat = '7C'

App B

Select name
from reservations
where seat is '7C'

John

Uncommitted read (also known as “dirty read”)

reservations

seat	name	...
7C	John	
7B	_____	
...		

App A

update reservations
set name = 'John'
where seat = '7C'

Roll back

App B

Select name
from reservations
where seat is '7C'

John

Uncommitted read (also known as “dirty read”)

reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

**update reservations
set name = 'John'
where seat = '7C'**

Roll back

App B

**Select name
from reservations
where seat is '7C'**

John

Uncommitted read (also known as “dirty read”)

reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

update reservations
set name = 'John'
where seat = '7C'

Roll back

App B

Select name
from reservations
where seat is '7C'

John

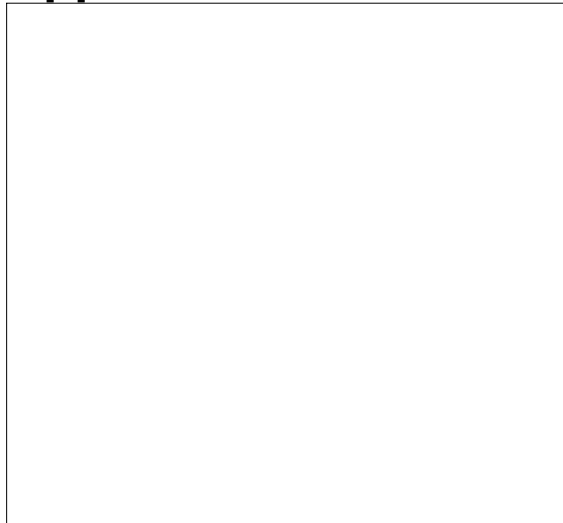
Further processing in App
B uses incorrect /
uncommitted value of
“John”

Non-repeatable read

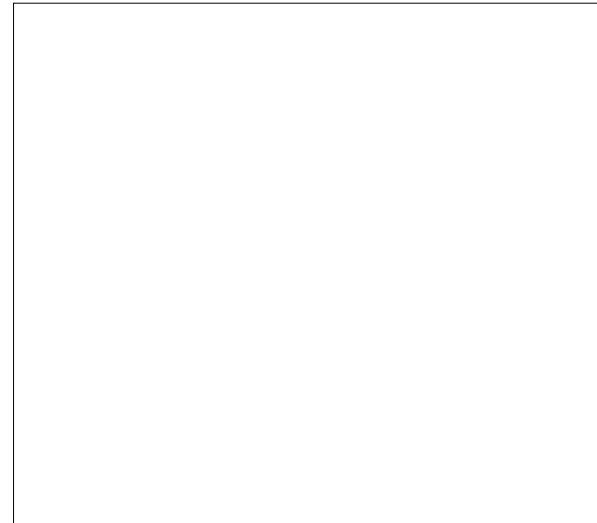
reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

A large, empty rectangular box representing the state of App A.

App B

A large, empty rectangular box representing the state of App B.

Non-repeatable read

reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

```
select seat
from reservations
where name is NULL
```

App B

Non-repeatable read

reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

**select seat
from reservations
where name is NULL**

App B

7C

7B

Non-repeatable read

reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

**select seat
from reservations
where name is NULL**

App B

**update reservations
set name = 'John'
where seat = '7C'**

7C

7B

Non-repeatable read

reservations

seat	name	...
7C	John	
7B	_____	
...		

App A

select seat
from reservations
where name is NULL

App B

update reservations
set name = 'John'
where seat = '7C'

7C

7B

Non-repeatable read

reservations

seat	name	...
7C	John	
7B	_____	
...		

App A

**select seat
from reservations
where name is NULL**

...

**select seat
from reservations
where name is NULL**

App B

**update reservations
set name = 'John'
where seat = '7C'**

7C

7B

Non-repeatable read

reservations

seat	name	...
7C	John	
7B	_____	
...		

App A

**select seat
from reservations
where name is NULL**

...

**select seat
from reservations
where name is NULL**

App B

**update reservations
set name = 'John'
where seat = '7C'**

7C

7B

7B

Non-repeatable read

reservations

seat	name	...
7C	John	
7B	_____	
...		

App A

select seat
from reservations
where name is NULL

...

select seat
from reservations
where name is NULL

App B

update reservations
set name = 'John'
where seat = '7C'

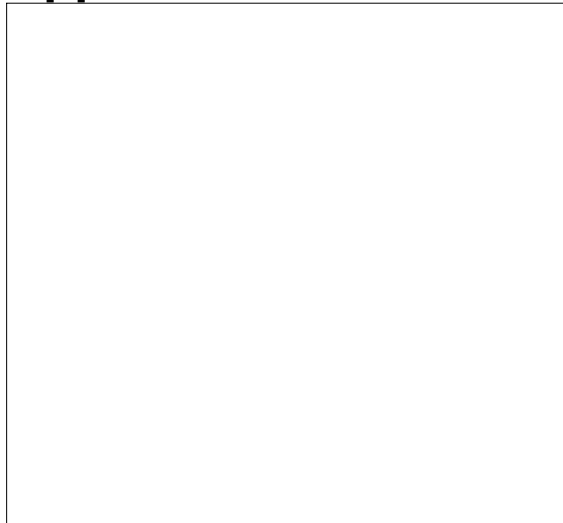
The same SELECT (read) returns a different result: Less rows (in this case '7C' doesn't show anymore). This is a non-repeatable read

Phantom read

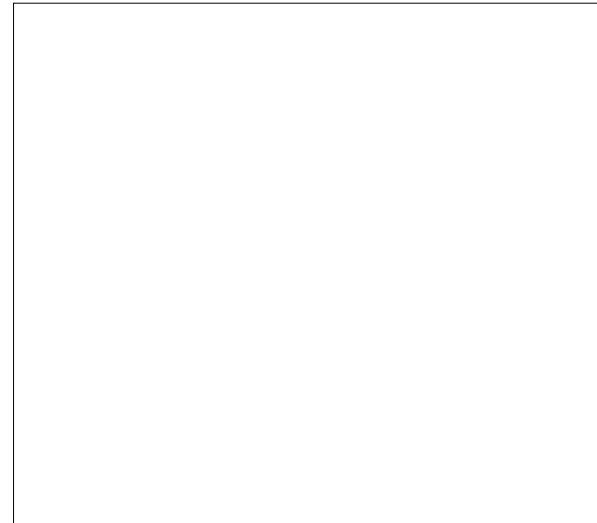
reservations

seat	name	...
7C	Susan	
7B	_____	
...		

App A

A large, empty rectangular box representing the state of App A.

App B

A large, empty rectangular box representing the state of App B.

Phantom read

reservations

seat	name	...
7C	Susan	
7B	_____	
...		

App A

```
select seat
from reservations
where name is NULL
```

App B

Phantom read

reservations

seat	name	...
7C	Susan	
7B	_____	
...		

App A

**select seat
from reservations
where name is NULL**

7B

App B

Phantom read

reservations

seat	name	...
7C	Susan	
7B	_____	
...		

App A

**select seat
from reservations
where name is NULL**

7B

App B

**update reservations
set name = NULL
where seat = '7C'**

Phantom read

reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

**select seat
from reservations
where name is NULL**

App B

**update reservations
set name = NULL
where seat = '7C'**

7B

Phantom read

reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

**select seat
from reservations
where name is NULL**

...

**select seat
from reservations
where name is NULL**

App B

**update reservations
set name = NULL
where seat = '7C'**

7B

Phantom read

reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

**select seat
from reservations
where name is NULL**

...

**select seat
from reservations
where name is NULL**

App B

**update reservations
set name = NULL
where seat = '7C'**

7B

7C

7B

Phantom read

reservations

seat	name	...
7C	_____	
7B	_____	
...		

App A

select seat
from reservations
where name is NULL

...

select seat
from reservations
where name is NULL

App B

update reservations
set name = NULL
where seat = '7C'

The same SELECT (read) returns a different result: More rows (phantom rows, in this case '7C', is shown)
This is a phantom read

7B

7C

7B