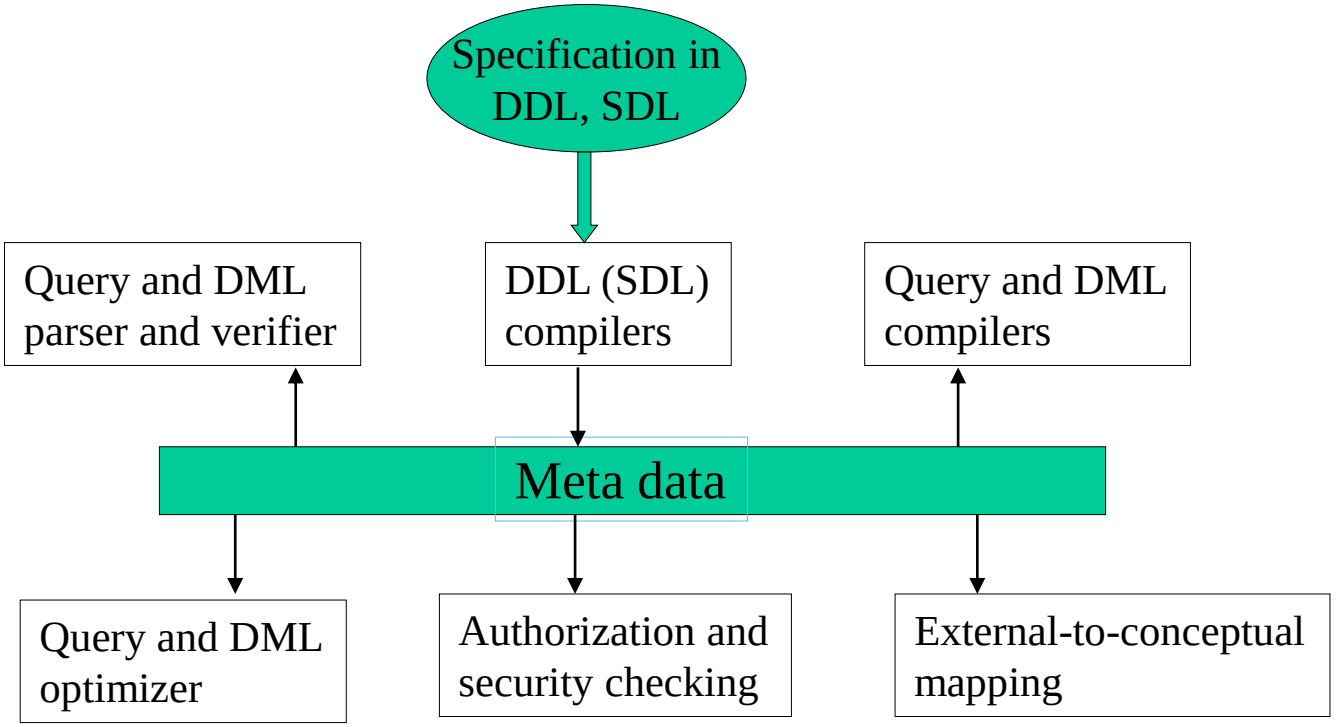
Answers to Midterm - Exam. Oct. 22, 2025

1. (20) (i)What is the ACID principle? (10)
(ii) Briefly describe how the meta data is created and used by the main software modules in a DB system. (10)

Answer:

(i) Atomic – a transaction is done in its entirety, or not at all
Consistent – a transaction leaves the database in a correct state. This is generally up to the programmer to guarantee.
Isolation – a transaction is isolated from other transactions so that there is not adverse inter-transaction interference
Durable – once completed (committed) the result of the transaction is not lost.



The meta date is mainly created by DDL compiler and used by the following DBMS software modules.

- Query and DMLparser and verifier
- Query and DMLcompilers
- Query and DMLoptimizer
- Authorization and security checking
- External-to-conceptual mapping

2.(20) Given the following schedules, show which is recoverable, non-recoverable, cascadeless or strict.

(a) R1(X), R3(X), W1(X), R1(Y), W2(Y), W2(X), W3(X), C1, C3, R2(X), C2. (6)
(b) R1(X), R2(X), W1(X), R1(Y), R3(Y), W2(X), C2, W1(Y), C1, W3(Y), C3. (6)
(c) R1(X), W1(X), R2(X), R1(Y), W2(X), C2, A1. (4)
(d) R1(X), W1(X), R2(X), R1(Y), W2(X), W1(Y), A1, A2. (4)

(a) cascadeless
(b) cascadeless
(c) non-recoverable
(d) recoverable

(a) R1(X), R3(X), W1(X), R1(Y), W2(Y), W2(X), W3(X), C1, C3, R2(X), C2.

R1(X) W1(X), R1(Y) C1

W2(Y), W2(X) R2(X), C2

R3(X) W3(X) C3

cacadeless

(b) R1(X), W1(X), R2(X), R1(Y), W2(X), W1(Y), A1, A2.

R1(X) W1(X), R1(Y) W1(Y), C1

R2(X) W2(X), C2

R3(Y) W3(Y), C3

cacadeless

(c) R1(X), W1(X), R2(X), R1(Y), W2(X), C2, A1.

R1(X), W1(X) R1(Y) A1

R2(X) W2(X), C2

non-recoverable

(d) R1(X), W1(X), R2(X), R1(Y), W2(X), W1(Y), A1, A2.

R1(X), W1(X) R1(Y) W1(Y), A1

R2(X) W2(X) A2

recoverable

3.(20) (a) Tell the difference between commit points and checking points. (10)
(b) Assume that at a checking point, all the changes made by transactions will be stored in the database, no matter whether they are committed or not yet committed. When the system crashes as shown in Fig. 1, how do we undo or redo the transactions? Here, we assume that all these transactions are not interferenced. (10)

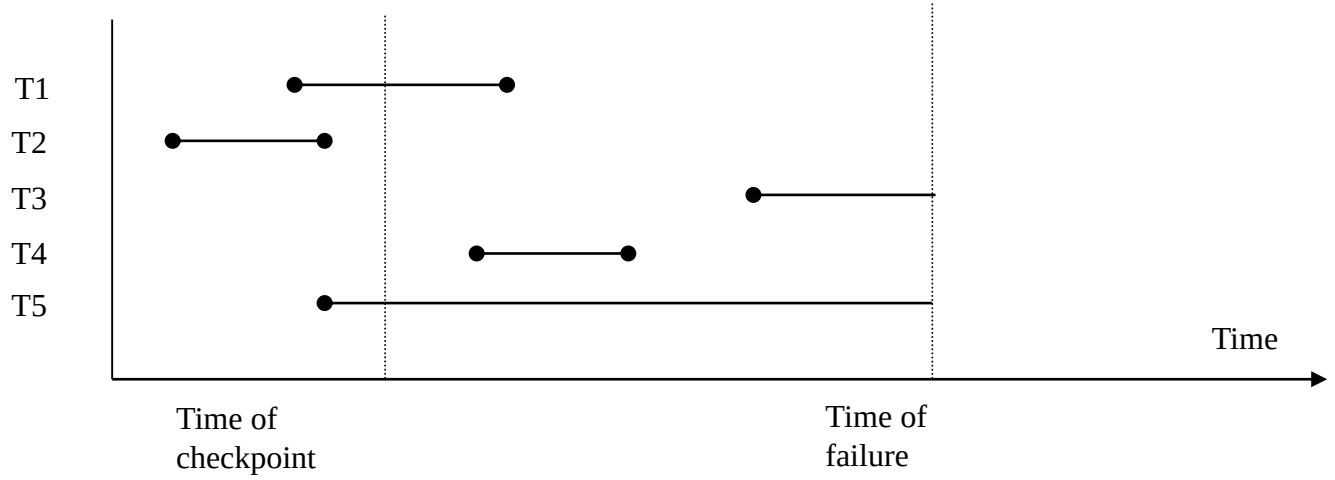


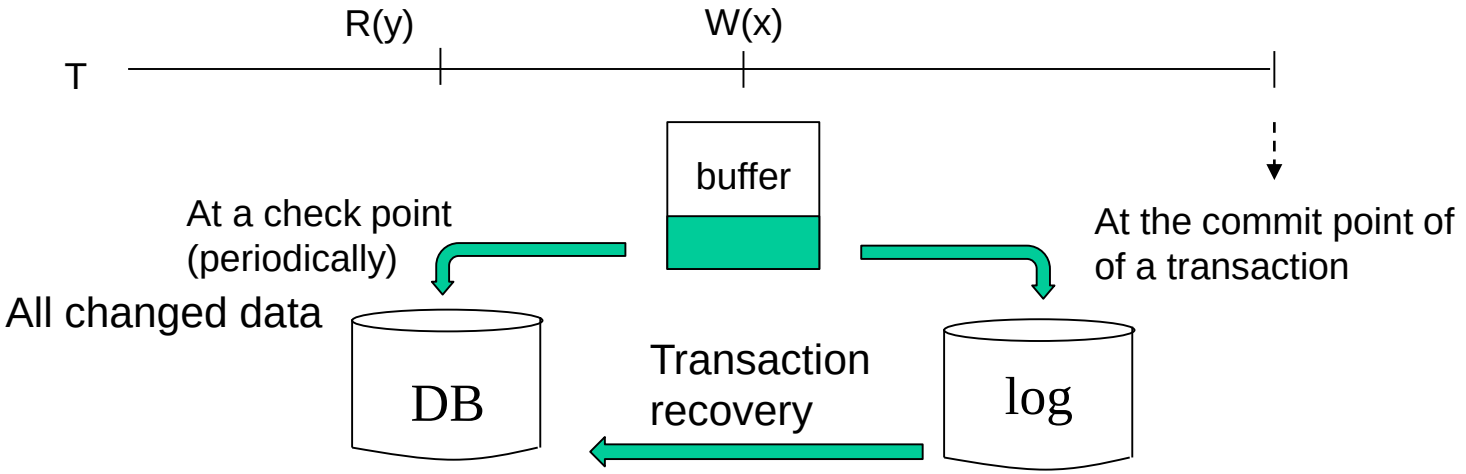
Fig. 1

3. (a) A transaction has committed when it has finished all this work, and explicitly conducted the commit command. At this point:

- The DBMS force-writes all changes/updates made by a transaction to the log. A commit record for the Transaction is also established in the log.

A checkpoint occurs periodically. At this time point:

- The DBMS force-writes all the changes made by transactions into the database. In addition, a checkpoint record will be stored in the log.



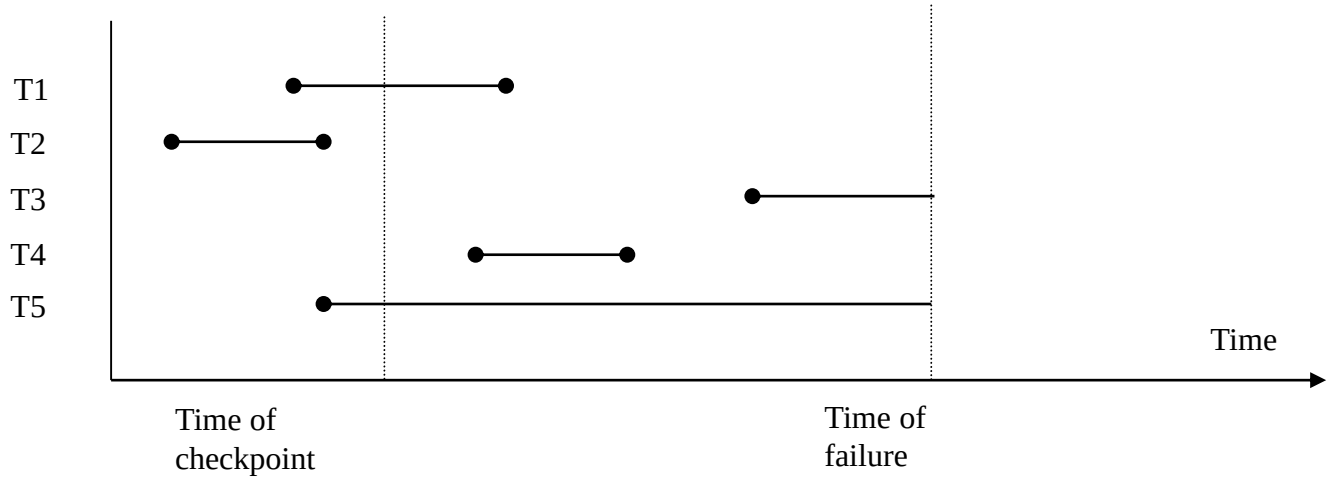


Fig. 1

3. (b) T1 – redone
T2 – no redo, no undo
T3 – no redo, no undo
T4 – redone
T5 – undone

4.(20)Given the relation schemas shown below, generate an optimal query tree for the following query:

```
SELECT    fname, lname
FROM      EMPLOYEE, DEPARTMENT, PROJECT,
          WORKS_ON
WHERE     salary > 50000 and Pname = 'Web Database'
          and Pnumber = PNO and hours > 30
          and dno = dnumber
          and Dname = 'CS'
          and ssn = ESSN
```

DEPARTMENT

Dname, dnumber, mgrssn, mgrstartdate

EMPLOYEE

fname, minit, lname, ssn, bdate, address, sex, salary, superssn, dno

PROJECT

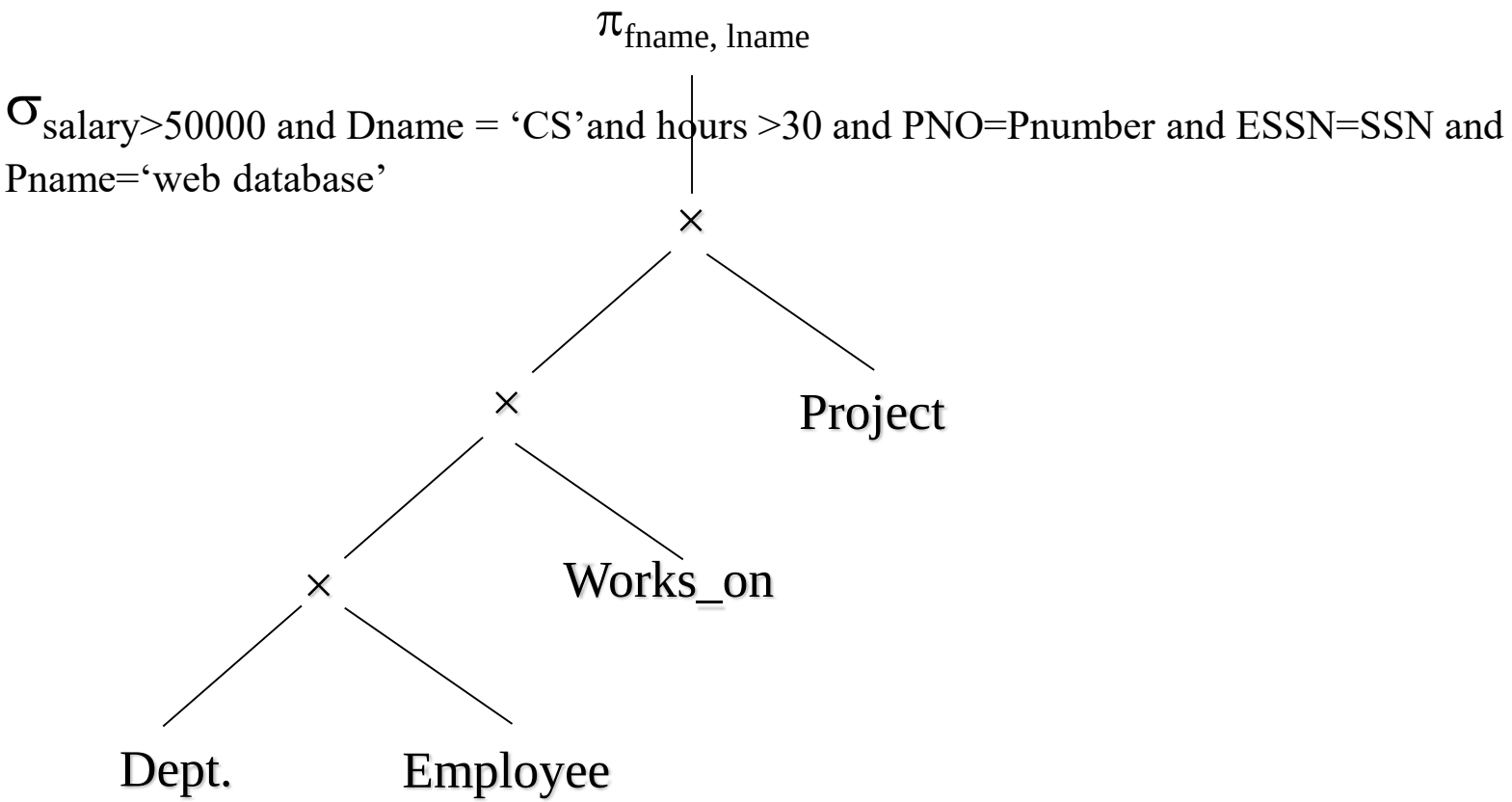
Pname, Pnumber, Plocation, Dnum

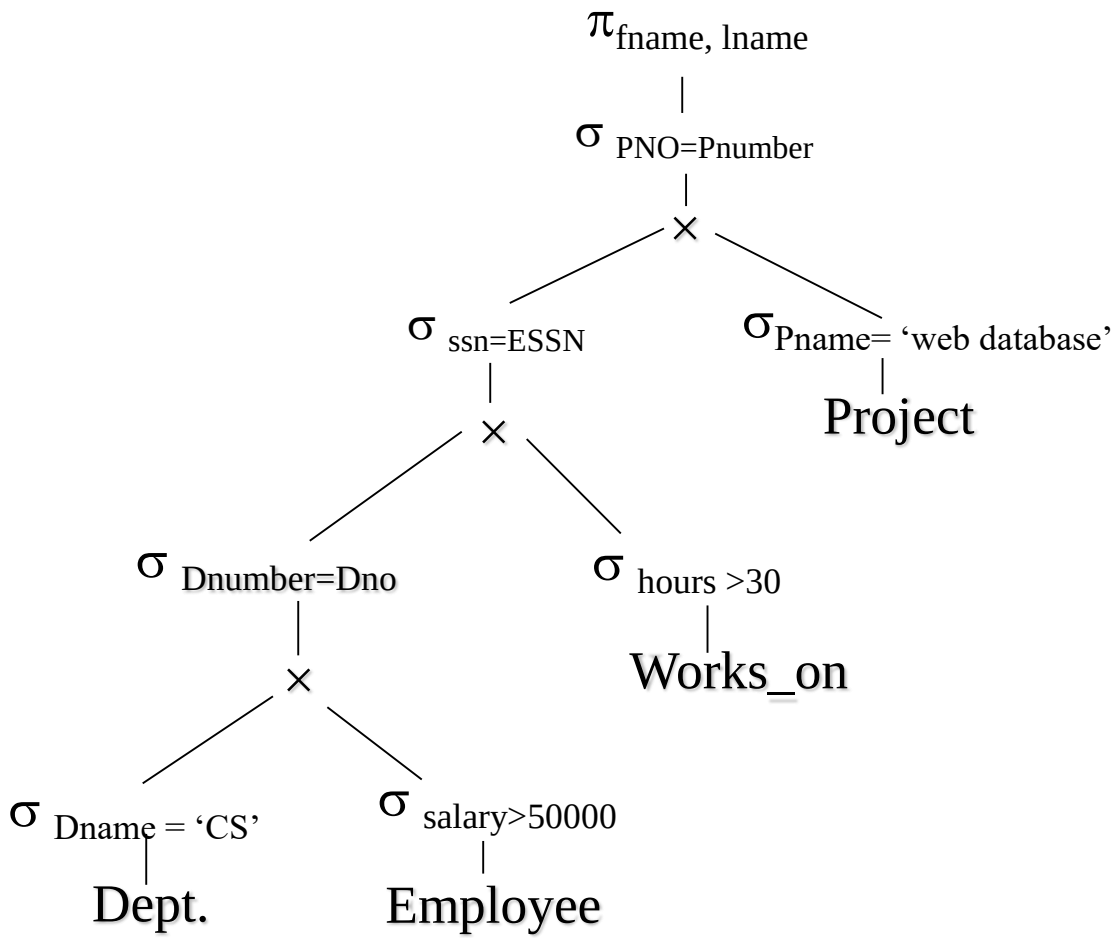
WORKS_ON

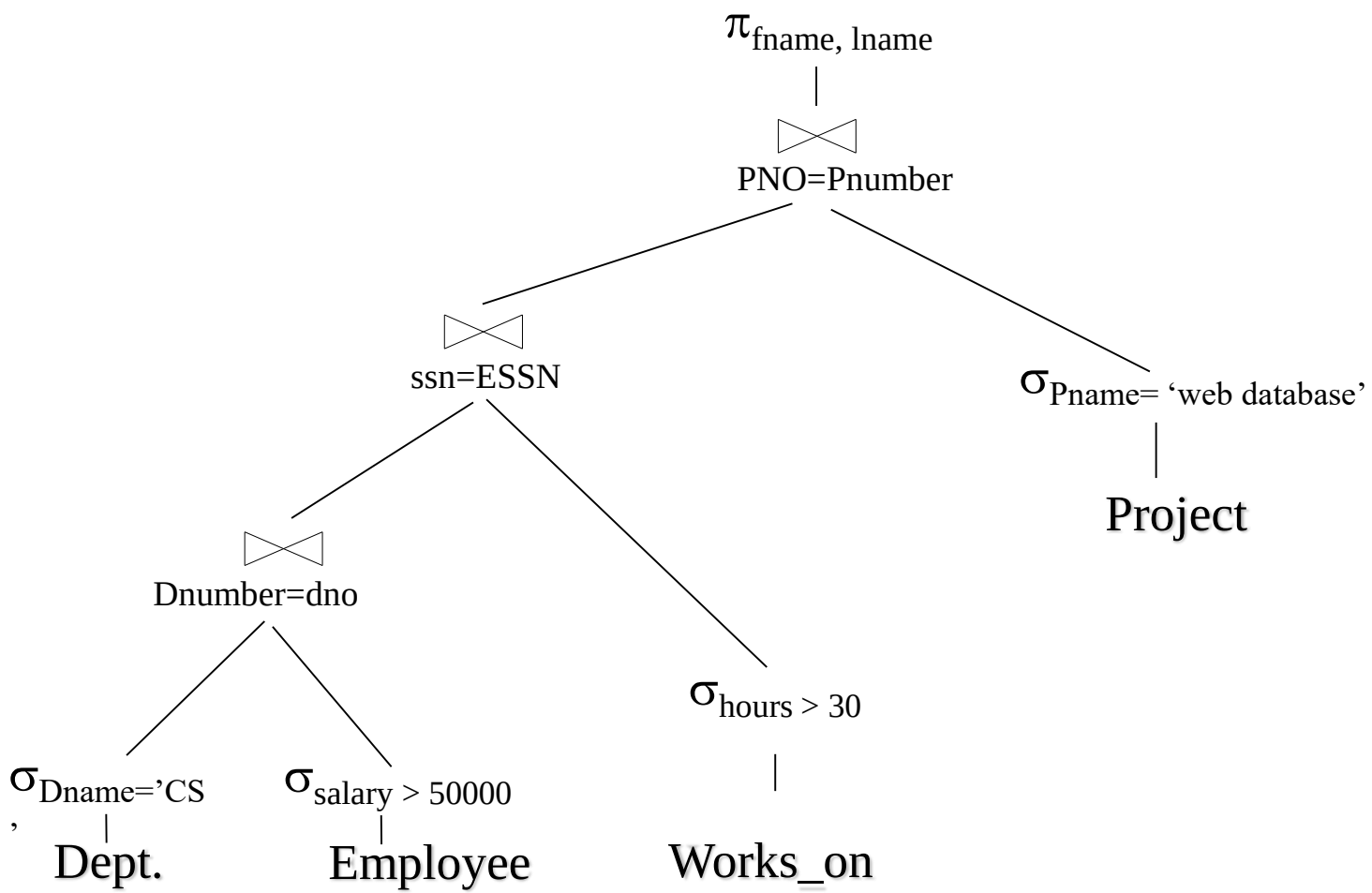
ESSN, PNO, HOURS

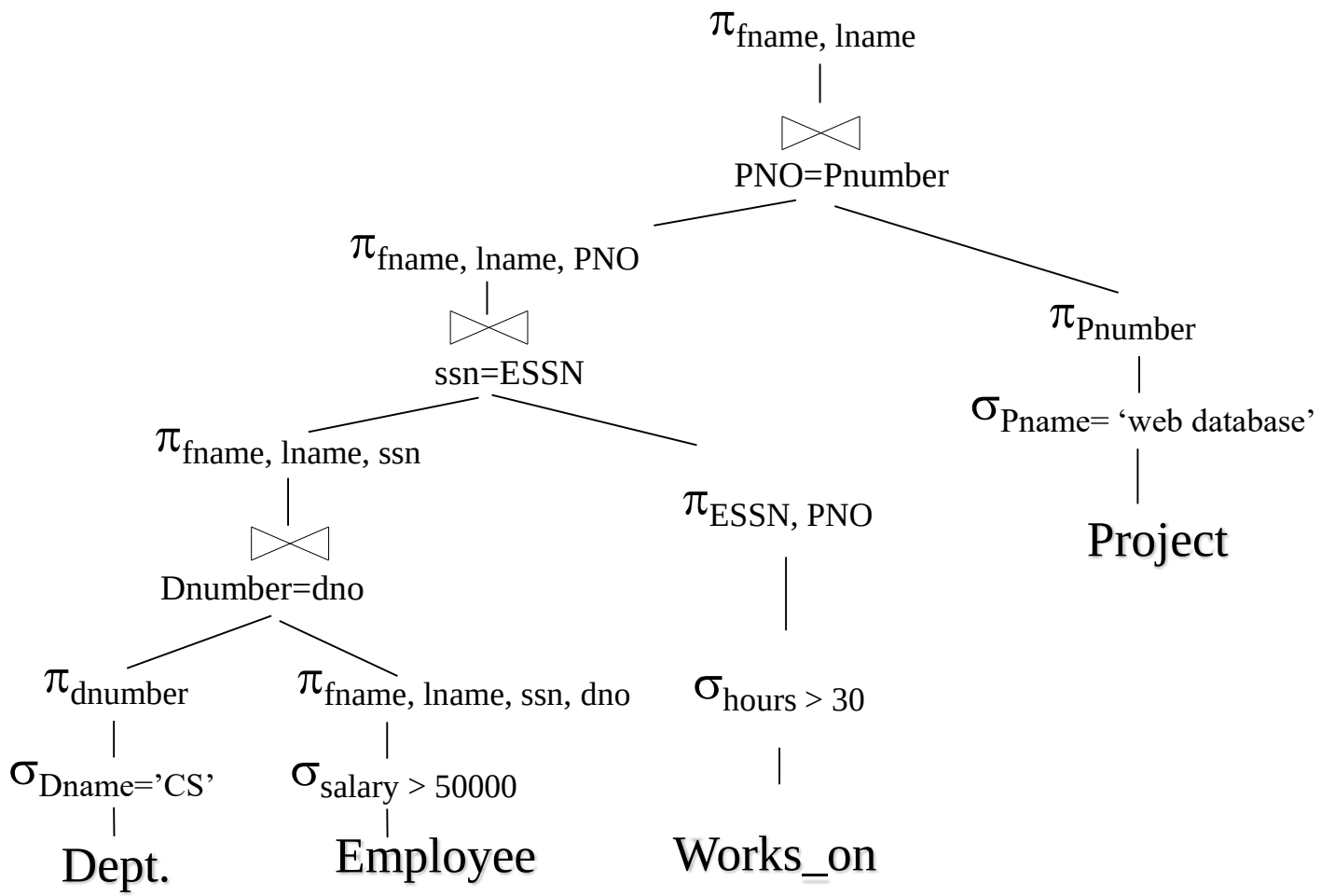
$$\pi_{\text{fname, lname}}((\sigma_{\text{salary} > 50000}(\text{Employee}) \underset{\text{dno=Dnumber}}{\bowtie} \sigma_{\text{Dname='CS'}}(\text{Department}) \underset{\text{ssn=ESSN}}{\bowtie} ((\sigma_{\text{hours} > 30}(\text{Works\_on}) \underset{\text{PNO=Pnumber}}{\bowtie} \sigma_{\text{Pname='web database'}}(\text{Project}))$$

- Question 4


$\pi_{fname, lname}$
$\sigma_{salary>50000 \text{ and } Dname = 'CS' \text{and hours } >30 \text{ and } PNO=Pnumber \text{ and } ESSN=SSN \text{ and } Pname='web\ database'}$
×
×
Project
×
Works_on
Dept.
Employee


πfname, lname
σ PNO=Pnumber
×
σ ssn=ESSN
σPname= ‘web database’
Project
×
σ Dnumber=Dno
σ hours >30
Works_on
×
σ Dname = ‘CS’
σ salary>50000
Dept.
Employee


πfname, lname
PNO=Pnumber
ssn=ESSN
σPname= ‘web database’
Project
Dnumber=dno
σhours > 30
σDname=’CS’
σsalary > 50000
Dept.
Employee
Works_on


πfname, lname
PNO=Pnumber
πfname, lname, PNO
ssn=ESSN
πPnumber
σPname= ‘web database’
Project
πfname, lname, ssn
πESSN, PNO
Dnumber=dno
πdnumber
πfname, lname, ssn, dno
σhours > 30
σDname=’CS’
σsalary > 50000
Dept.
Employee
Works_on

5(i) Consider the following transactions T1 and T2, which execute concurrently. Is the result of this execution correct? Explain your answer and try this for X = 10, Y = 20, N = 5 and M = 8. (5)

T1	**T2**
Read(X)	
X := X - N	
	Read(X)
	X := X + M
Write(X)	
Read(Y)	
	Write(X)
Y := Y + N	
Write(Y)	
Commit	
	Commit

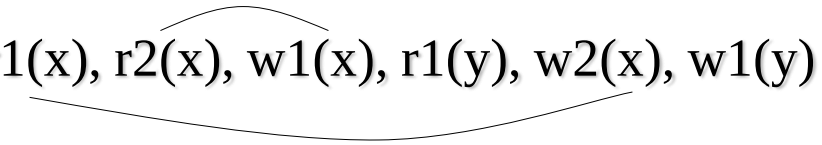
5(ii) If the two transactions are executed interleavingly as follows and the initial values for X and Y are 20 and 30, respectively, what are the values of X and Y after the execution. Explain whether it is correct by using the concept of conflict equivalence. (5)

T1	**T2**
read_item(Y)	
	read_item(X)
	read_item(Y)
	Y:=X+Y
	write_item(Y)
	Commit
read_item(X)	
X :=X+Y	
write_item(X)	
Commit	

Answer:

5(i) The result produced by the interleaved transaction is not correct.
It suffers the so-called lost-update problem.
After this interleaved execution, X becomes 5 and Y becomes 25.
The result of the serial execution is X = 13 and Y = 25.

The corresponding schedule is not serializable due to the two conflict pairs:
r1(x), w2(x); r2(x), w1(x).



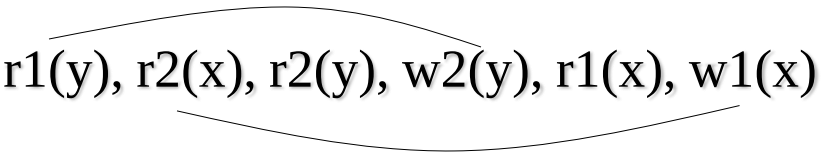
Answer:

5(ii) The result produced by the interleaved transaction is not correct.
It suffers the so-called incorrect summary problem.
After this interleaved execution, both X and Y are equal to 50.
The result of the serial execution is:

if T1 is before T2, then X = 50 and Y = 80;
if T2 is before T1, then X = 50 and Y = 70.

The corresponding schedule is not serializable due to the two conflict pairs:

r1(y), w2(y); r2(x), w1(x).



6. (10) (i) Please explain whether the following identity holds. (5)

$\pi_L(R \cap S) = \pi_L(R) \cap \pi_L(S)$,

where R and S are two tables, and L is a list of attributes.

(ii) Show the following identity always holds. (5)

Not (c1 OR c2) ≡ (Not c1) AND (Not c2),

where c1 and c2 are two boolean variables.

Answer:

(i) The identity does not hold.

See the following counter-example:

R =

A	B
2	3
1	3

S =

A	B
1	3
2	1

$\pi_A(R \cap S)$ =

A
1

$\pi_A(R) \cap \pi_A(S)$ =

A
1
2

They are not equal.

Answer:

(i) The identity holds as shown below.

c1	c2	Not (c1 OR c2)	(Not c1) AND (Not c2)
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0