EBU5503 Database Systems Revision

Topics covered – 1/2

- Teaching Week 1
  - Introduction
  - Relational model
  - Relational Algebra
  - Entity-relationship(ER) modelling
- Teaching Week 2
  - EER model
  - ER to Relational Model mapping
  - Database design
  - SQL

Topics covered – 2/2

- Teaching Week 3
  - Normalization
  - Advanced Normalization
  - Transaction Management

- Teaching Week 4
  - Distributed DBMS
  - Database Security and Data Ethics
  - XML
  - NoSQL

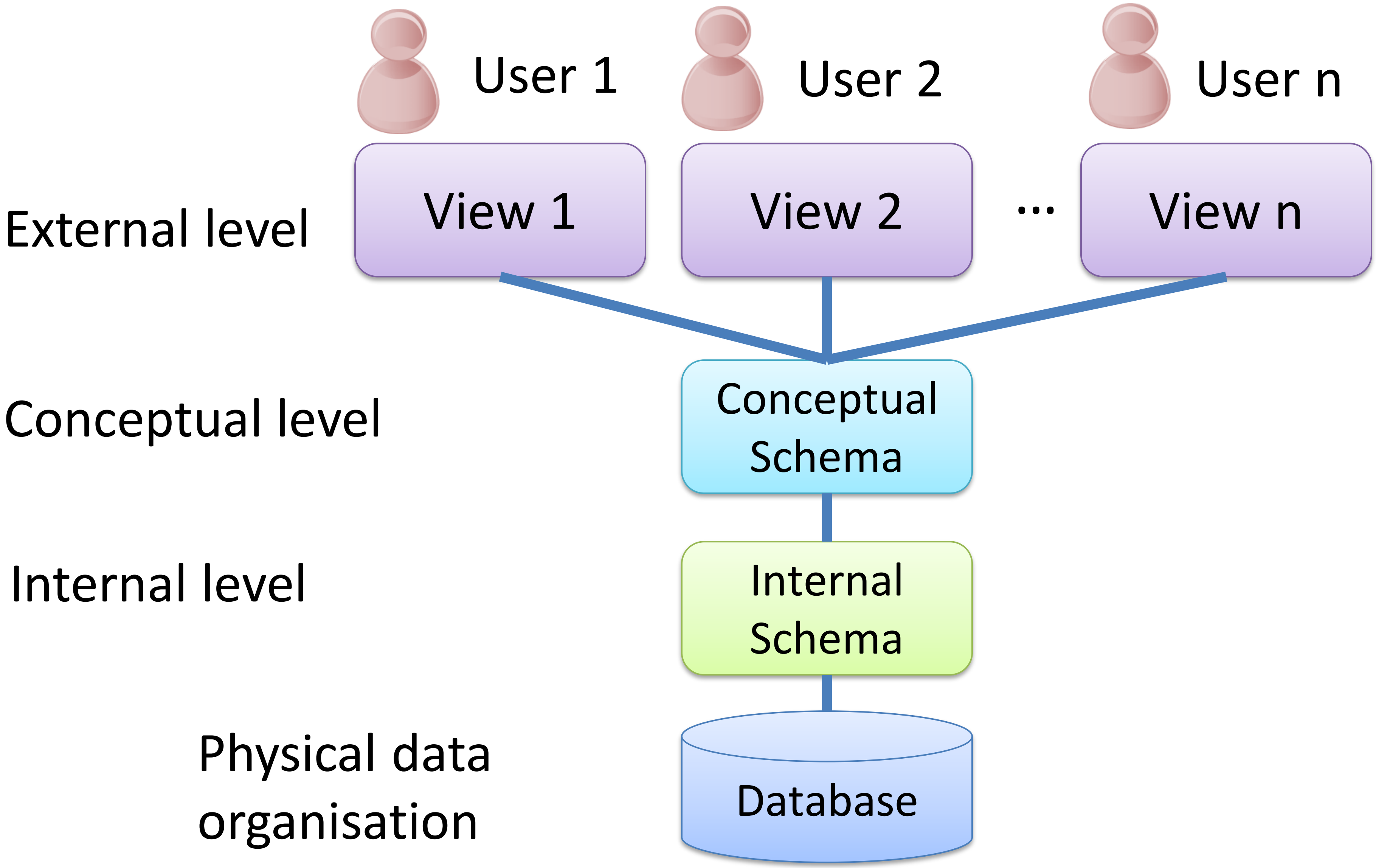
Database Introduction

Concepts

- Data
- Database
- Database management system (DBMS)
- Data model
- Schema versus data
- Database languages:
  - Data Definition Language (DDL)
  - Data Manipulation Language (DML)

Data abstraction

The three-level ANSI-SPARC architecture



User 1

User 2

User n

External level

View 1

View 2

…

View n

Conceptual level

Conceptual Schema

Internal level

Internal Schema

Physical data organisation

Database

Relational Model

Concepts

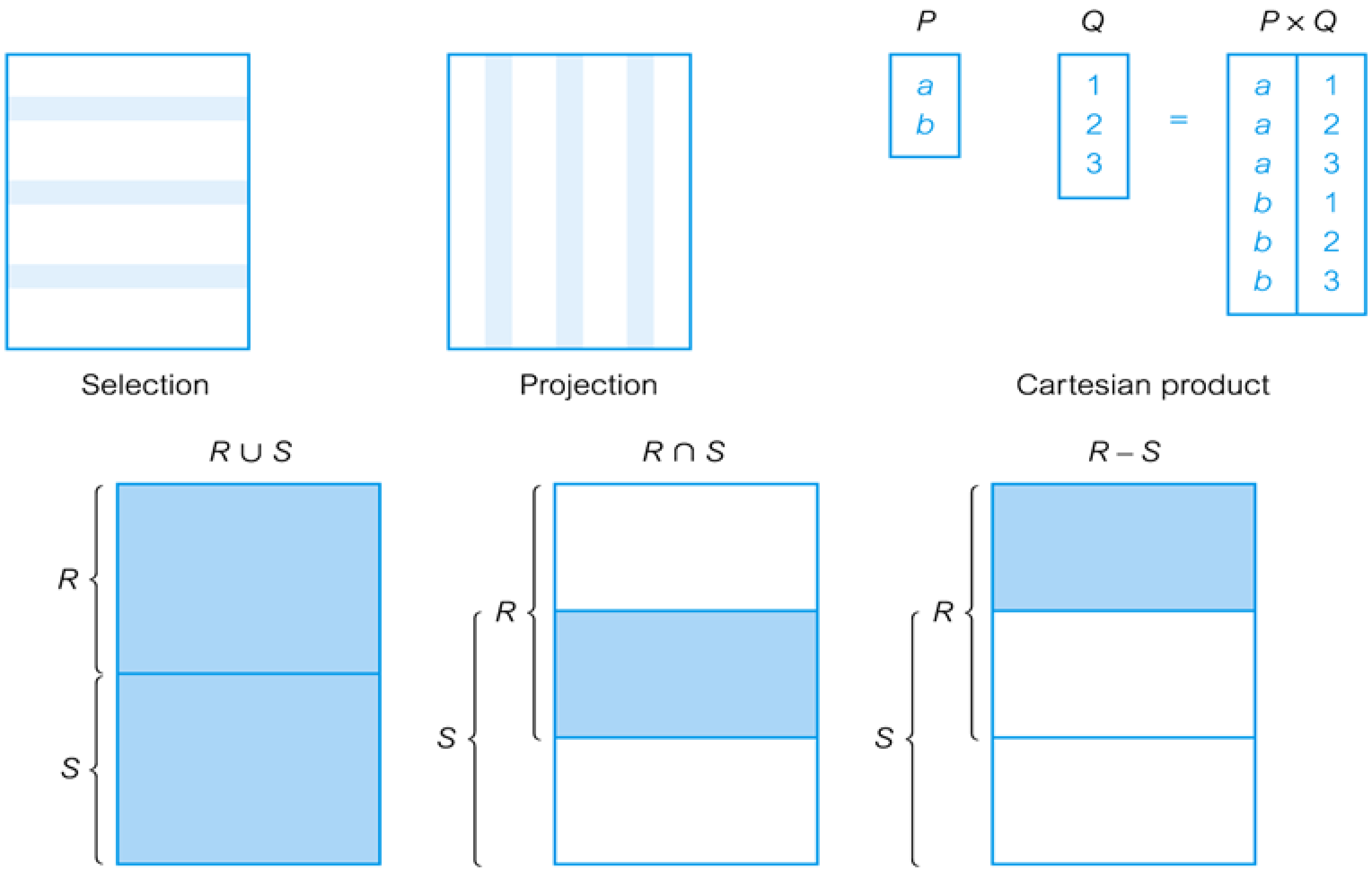
- Relational model:
  - Relation (table)
  - Attribute (column)
  - Tuple (row)
  - Cardinality
  - Degree
  - Domain
- Keys:
  - Candidate key
  - Primary key
  - Foreign key

Integrity constraints

- Entity Integrity
    - In a base relation, no attribute of a primary key can be null.
- Referential Integrity
    - If foreign key exists in a relation, either foreign key value must match a candidate key value of some tuple in its home relation or foreign key value must be null.
- General Constraints

Relational Algebra

Relational Algebra Operations



Selection

Projection

Cartesian product

Union

Intersection

Set difference

Relational Algebra Operations

T

A	B
a	1
b	2

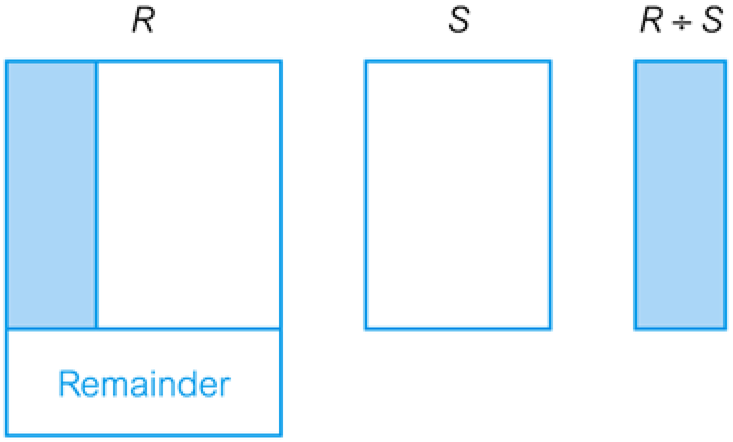
U

B	C
1	x
1	y
3	z

$T \bowtie U$

A	B	C
a	1	x
a	1	y

Natural join



Divis on (shaded area)

V

A	B
a	1
a	2
b	1
b	2
c	1

W

B
1
2

$V \div W$

A
a
b

Example of division

ER & EER

Entity Relationship (ER) Model

- Entity, Relationship, Attributes
- Multiplicity of relationship types:
  - one-to-one (1:1)
  - one-to-many (1:*)
  - many-to-many (*:*)
- Ternary relationship
- Making Assumptions

Enhanced Entity Relationship (EER) Model

- Specialization/Generalization
    - participation constraints (*mandatory* or *optional)*
    - disjoint constraints (*disjoint* (or) or *nondisjoint (and)*)

Logical Database Design (ER to Relational Model mapping)

Mapping ER model concepts to relations

- Entity to relations
- Binary 1-1, 1-N, N-M relationships
- Complex relationships
- Multi-valued attributes

SQL

- SELECT statement

```
SELECT [DISTINCT | ALL]
   {* | [columnExpression [AS newName]] [,...] }
FROM TableName [alias] [, ...]
[WHERE              condition]
[GROUP BY   columnList]
[HAVING             group condition]
[ORDER BY   columnList]
```

- Aggregates

  COUNT, SUM, AVG, MIN, MAX

- Subqueries
- JOIN

  NATURAL JOIN, JOIN… USING…, JOIN…ON…, LEFT (RIGHT) JOIN

This is **a summary only**, i.e. there are other SQL commands you learnt which are not listed here. They too are examinable …

Normalization

Normalization – 1/5

- Purpose of normalization
- Data redundancy
- Update Anomalies
  - Insertion
  - Deletion
  - Modification
- Normalization is done via decomposition
  - Lossless-join
  - Dependency preservation

Normalization – 2/5

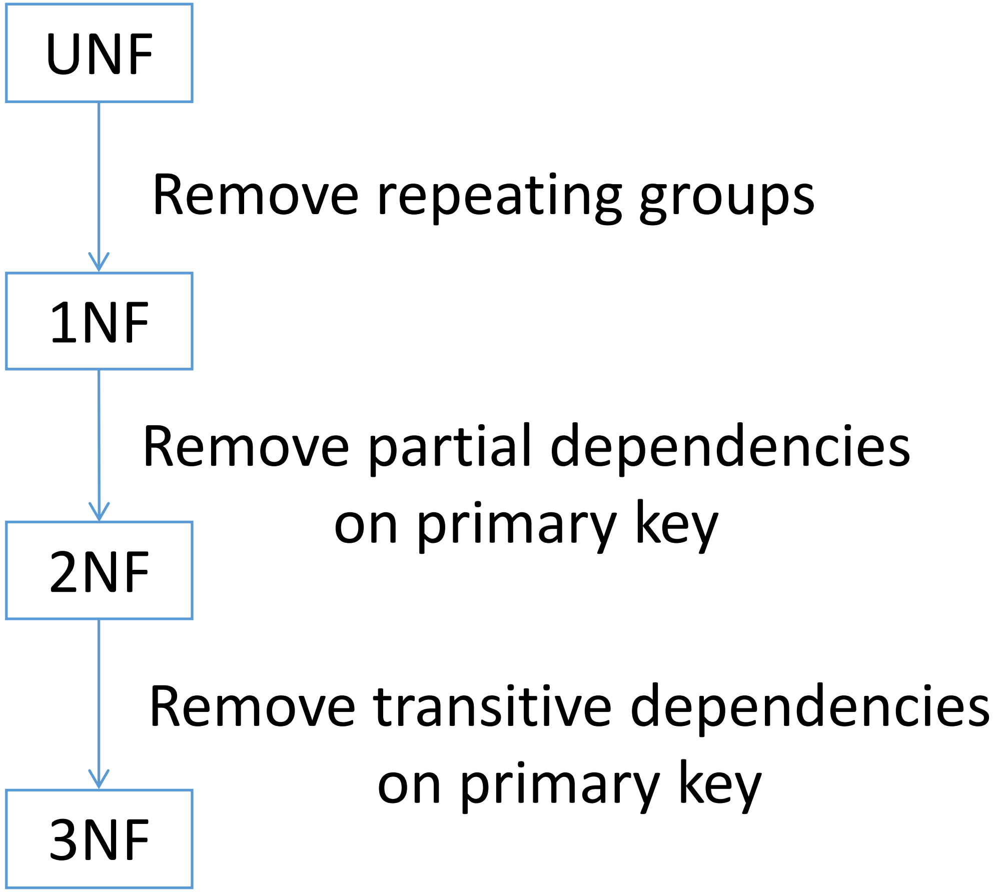
- Functional dependencies:

  A and B are attributes of relation R, B is functionally dependent on A (denoted A → B), if each value of A in R is associated with exactly one value of B in R.

- Full functional dependencies
- Transitive functional dependencies

Normalization – 3/5

- UNF, 1NF, 2NF, 3NF, BCNF, 4NF


UNF
Remove repeating groups
1NF
Remove partial dependencies on primary key
2NF
Remove transitive dependencies on primary key
3NF

Normalization – 4/5

- 1NF: A relation in which the intersection of each row and column contains one and only one value.
- 2NF: A relation that is in 1NF and every non-primary-key attribute is fully functionally dependent on the primary key.
- 3NF: A relation that is in 1NF and 2NF and in which no non-primary-key attribute is transitively dependent on the primary key.
- General definitions for 2NF and 3NF:
  - 2NF: A relation that is in first normal form and every *non-candidate-key* attribute is fully functionally dependent on any *candidate key*.
  - 3NF: A relation that is in first and second normal form and in which no *non-candidate-key* attribute is transitively dependent on any *candidate key*.

Normalization – 5/5

- BCNF: every determinant is a candidate key.
- 4NF: a relation is in 4NF if and only if for every nontrivial multi-valued dependency A->>B, A is a candidate key of the relation.
- Multi-valued Dependency (MVD):
  - for each value of A there is a set of values for B and a set of values for C. However, the set of values for B and C are independent of each other.
  - Trivial MVD: A MVD A −>> B in relation R is defined as being trivial if

    (a) B is a subset of A *or* (b) A $\cup$ B = R.
  - Nontrivial MVD: if neither (a) nor (b) are satisfied

Algorithm for decomposing relations into BCNF

- Relation R with FDs
- Compute keys for R
- Repeat until all relations are in BCNF:
    - Pick any R’ with A->B that violates BCNF
    - Decompose R’ into R1(A, B) and R2(A, rest)
    - Compute FDs for R1 and R2
    - Compute keys for R1 and R2

Transaction Management

Transactions

- Transaction: Action, or series of actions, carried out by user or application, which reads or updates contents of database.
- ACID properties of a transaction:

  Atomicity, Consistency, Isolation, Durability
- Concurrency control purpose
- Concurrency control problems:
  - Lost update problem.
  - Uncommitted dependency problem.
  - Inconsistent analysis problem.

Transaction management

- Serialisability
  - Schedule
  - Serial schedule/nonserial schedule
  - Aim of serialisability
  - Serialisable schedule/Conflict serializable schedule
- Two-Phase Locking (2PL)
- Deadlock
- Database recovery
  - log file
  - checkpointing

Distributed DBMSs

- Client/Server Architecture
- Distributed database
- Distributed processing
- Distributed DBMS
  - Fragmentation
  - Allocation
  - Replication
- Levels of transparency in DDBMSs
- Advantages/disadvantages of distributed DBMS

Database Security and Data Ethics

- Database security
- Database security measures:
  - Authorization, Access controls, Views, Backup and recovery, Integrity, Encryption
- SQL injection attack
- Preventing SQL injection
- Data Ethics: concept and importance
- Principles of Data Ethics
- Relevant legislation for Data Ethics

XML

- XML definition and basic concepts
- Relational model versus XML
- Well-formed XML, Valid XML
- DTD, XSD

NoSQL

- NoSQL: motivation, how it differs from RDBMS, and concepts
- BASE versus ACID properties
- NoSQL systems
- NoSQL application areas

Revision

- Read **all** lecture notes, go through all exercises, tutorial questions, quiz questions etc
- Make your own notes
- Understanding is the key
- Everything that is on the lecture slides is examinable
- Check – have you met the learning outcomes at the beginning of each lecture notes' set?
- Practice **writing down** the description of concepts or explanations, in your own words
- Explain concepts to each other with your friends – helps your own learning

Exam techniques

- Read the exam instructions
- Read all the questions first
  - Start with the questions about topics you feel most knowledgeable about
- Read the questions **carefully**
  - Look out for words like “explain”, “illustrate”, “describe”, “list”, “use examples”…