Unit 7: Database Design

General Information

- Initial simplified definitions:
  - data = recorded facts and numbers.
  - database = the structure used to hold or store that data.
  - Data is processed to provide information
- The purpose of a database is to help people keep track of things, and the most commonly used type of database is the relational database.
- A relational database stores data in tables. A table has rows and columns, like those in a spreadsheet. A database usually has multiple tables, and each table contains data about a different type of thing
- Each row in a table is uniquely identified by a primary key, and the values of these keys are used to create the relationships between the tables.
- A database is a self-describing collection of integrated tables. An Integrated table is a table that stores both data and the relationships among the data.
- A database is self-describing because it contains a description of itself. Thus, databases contain not only tables of user data, but also tables of data that describe that user data. Such descriptive data is called metadata because it is data about data.
- Databases arise from three sources: from existing data, from the development of new information systems, and from the redesign of existing databases.
- Source (Kroenke et al, 2018: Chapters 1 & 3)

General Info

- A database = a collection of related data / A shared collection of logically related data and its description, designed to meet the information needs of an organization.
- A database management system (DBMS) = the software that manages and controls access to the database.
- A database application = a program that interacts with the database at some point in its execution. (Connolly & Begg, 2015: Chapters 1, 4, 14)

TIMEFRAME	DEVELOPMENT	COMMENTS
1960s (onwards)	File-based systems	Precursor to the database system. Decentralized approach: each department stored and controlled its own data.
Mid-1960s	Hierarchical and network data models	Represents first-generation DBMSs. Main hierarchical system is IMS from IBM and the main network system is IDMS/R from Computer Associates. Lacked data independence and required complex programs to be developed to process the data.
1970	Relational model proposed	Publication of E. F. Codd's seminal paper "A relational model of data for large shared data banks," which addresses the weaknesses of first-generation systems.
1970s	Prototype RDBMSs developed	During this period, two main prototypes emerged: the Ingres project at the University of California at Berkeley (started in 1970) and the System R project at IBM's San José Research Laboratory in California (started in 1974), which led to the development of SQL.
1976	ER model proposed	Publication of Chen's paper "The Entity-Relationship model—Toward a unified view of data." ER modeling becomes a significant component in methodologies for database design.
1979	Commercial RDBMSs appear	Commercial RDBMSs like Oracle, Ingres, and DB2 appear. These represent the second generation of DBMSs.
1987	ISO SQL standard	SQL is standardized by the ISO (International Standards Organization). There are subsequent releases of the standard in 1989, 1992 (SQL2), 1999 (SQL:1999), 2003 (SQL:2003), 2008 (SQL:2008), and 2011 (SQL:2011).
1990s	OODBMS and ORDBMSs appear	This period initially sees the emergence of OODBMSs and later ORDBMSs (Oracle 8, with object features released in 1997).
1990s	Data warehousing systems appear	This period also see releases from the major DBMS vendors of data warehousing systems and thereafter data mining products.
Mid-1990s	Web–database integration	The first Internet database applications appear. DBMS vendors and third-party vendors recognize the significance of the Internet and support web–database integration.
1998	XML	XML 1.0 ratified by the W3C. XML becomes integrated with DBMS products and native XML databases are developed.

Figure 1.10 Historical development of database systems.

(Connolly & Begg, 2015: Chapters 1, 4, 14)

Relational Database Management Systems (DBMS)

- This methodology consists of three main phases: conceptual, logical, and physical database design.
  - The first phase starts with the production of a conceptual data model that is independent of all physical considerations.
  - This model is then refined in the second phase into a logical data model by removing constructs that cannot be represented in relational systems.
  - In the third phase, the logical data model is translated into a physical design for the target DBMS. The physical design phase considers the storage structures and access methods required for efficient and secure access to the database on secondary storage.
- (Connolly & Begg, 2015: Chapters 1, 4, 14)

TABLE 1.2 Advantages of DBMSs.

Control of data redundancy	Economy of scale
Data consistency	Balance of conflicting requirements
More information from the same amount of data	Improved data accessibility and responsiveness
Sharing of data	Increased productivity
Improved data integrity	Improved maintenance through data independence
Improved security	Increased concurrency
Enforcement of standards	Improved backup and recovery services

(Connolly & Begg, 2015: Chapters 1, 4, 14)

TABLE 1.3 Disadvantages of DBMSs.

Complexity
Size
Cost of DBMSs
Additional hardware costs
Cost of conversion
Performance
Greater impact of a failure

Characteristics of Relations
Rows contain data about an entity.
Columns contain data about attributes of the entities.
All entries in a column are of the same kind.
Each column has a unique name.
Cells of the table hold a single value.
The order of the columns is unimportant.
The order of the rows is unimportant.
No two rows may be identical.

(Kroenke et al, 2018: Chapters 1 & 3)

FIGURE 3-9
Three Sets of Equivalent Terms

Table	Column	Row
Relation	Attribute	Tuple
File	Field	Record

Relation Model

- In the relational model, all data is logically structured within relations (tables). Each relation has a name and is made up of named attributes (columns) of data. Each tuple (row) contains one value per attribute.
- Relation A relation is a table with columns and rows.
- Attribute An attribute is a named column of a relation.
- Domain A domain is the set of allowable values for one or more attributes. Every attribute in a relation is defined on a domain. Domains may be distinct for each attribute, or two or more attributes may be defined on the same domain.
- A tuple is a row of a relation.
- The degree of a relation is the number of attributes it contains
- The cardinality of a relation is the number of tuples it contains.
- Relational database A collection of normalized relations with distinct relation names
- These normal forms are defined so that a relation in BCNF is in 3NF, a relation in 3NF is in 2NF, and a relation in 2NF is in 1NF. Thus, if you put a relation into BCNF, it is automatically in the lesser normal forms
- Null Represents a value for an attribute that is currently unknown or is not applicable for this tuple.

(Connolly & Begg, 2015: Chapters 1, 4, 14)

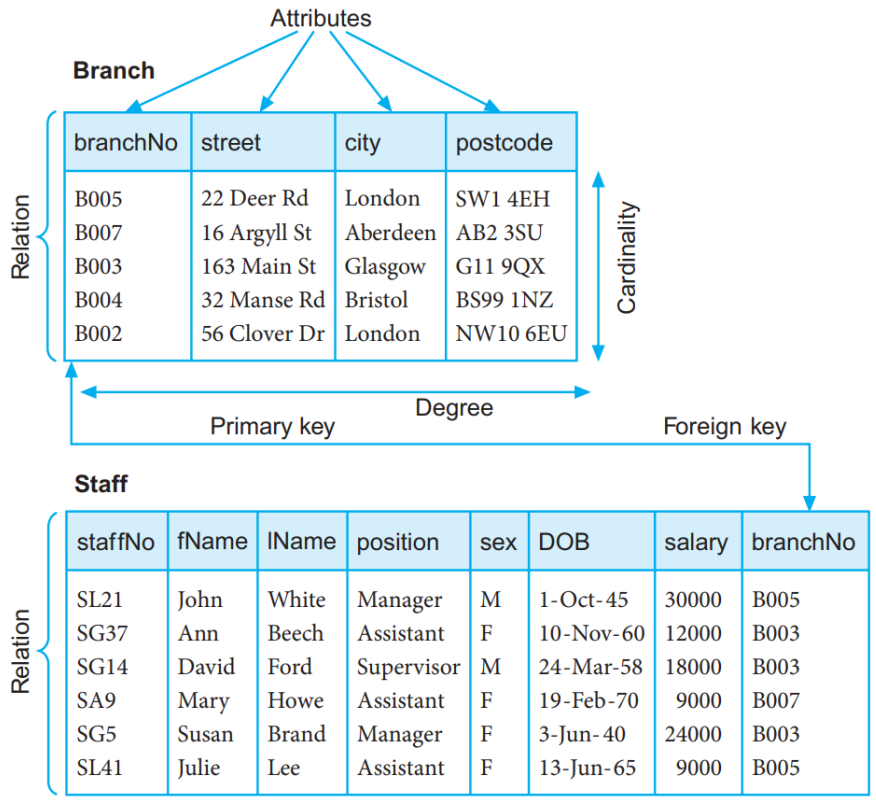

Figure 4.1
Instances of the Branch and Staff relations.

Attribute	Domain Name	Meaning	Domain Definition
branchNo	BranchNumbers	The set of all possible branch numbers	character: size 4, range B001–B999
street	StreetNames	The set of all street names in Britain	character: size 25
city	CityNames	The set of all city names in Britain	character: size 15
postcode	Postcodes	The set of all postcodes in Britain	character: size 8
sex	Sex	The sex of a person	character: size 1, value M or F
DOB	DatesOfBirth	Possible values of staff birth dates	date, range from 1-Jan-20, format dd-mmm-yy
salary	Salaries	Possible values of staff salaries	monetary: 7 digits, range 6000.00–40000.00

Figure 4.2
Domains for some attributes of the Branch and Staff relations.

(Connolly & Begg, 2015: Chapters 1, 4, 14)

4.2.5 Relational Keys

- As stated earlier, there are no duplicate tuples within a relation. Therefore, we need to be able to identify one or more attributes (called relational keys) that uniquely identifies each tuple in a relation. In this section, we explain the terminology used for relational keys.
- Superkey An attribute, or set of attributes, that uniquely identifies a tuple within a relation
- Candidate key A superkey such that no proper subset is a superkey within the relation
- A candidate key K for a relation R has two properties: • Uniqueness. In each tuple of R, the values of K uniquely identify that tuple. • Irreducibility. No proper subset of K has the uniqueness property.
- There may be several candidate keys for a relation. When a key consists of more than one attribute, we call it a composite key.
- Primary key The candidate key that is selected to identify tuples uniquely within the relation.
- Foreign key An attribute, or set of attributes, within one relation that matches the candidate key of some (possibly the same) relation.

(Connolly & Begg, 2015: Chapters 1, 4, 14)

Normalisation: A technique for producing a set of relations with desirable properties, given the data requirements of an enterprise

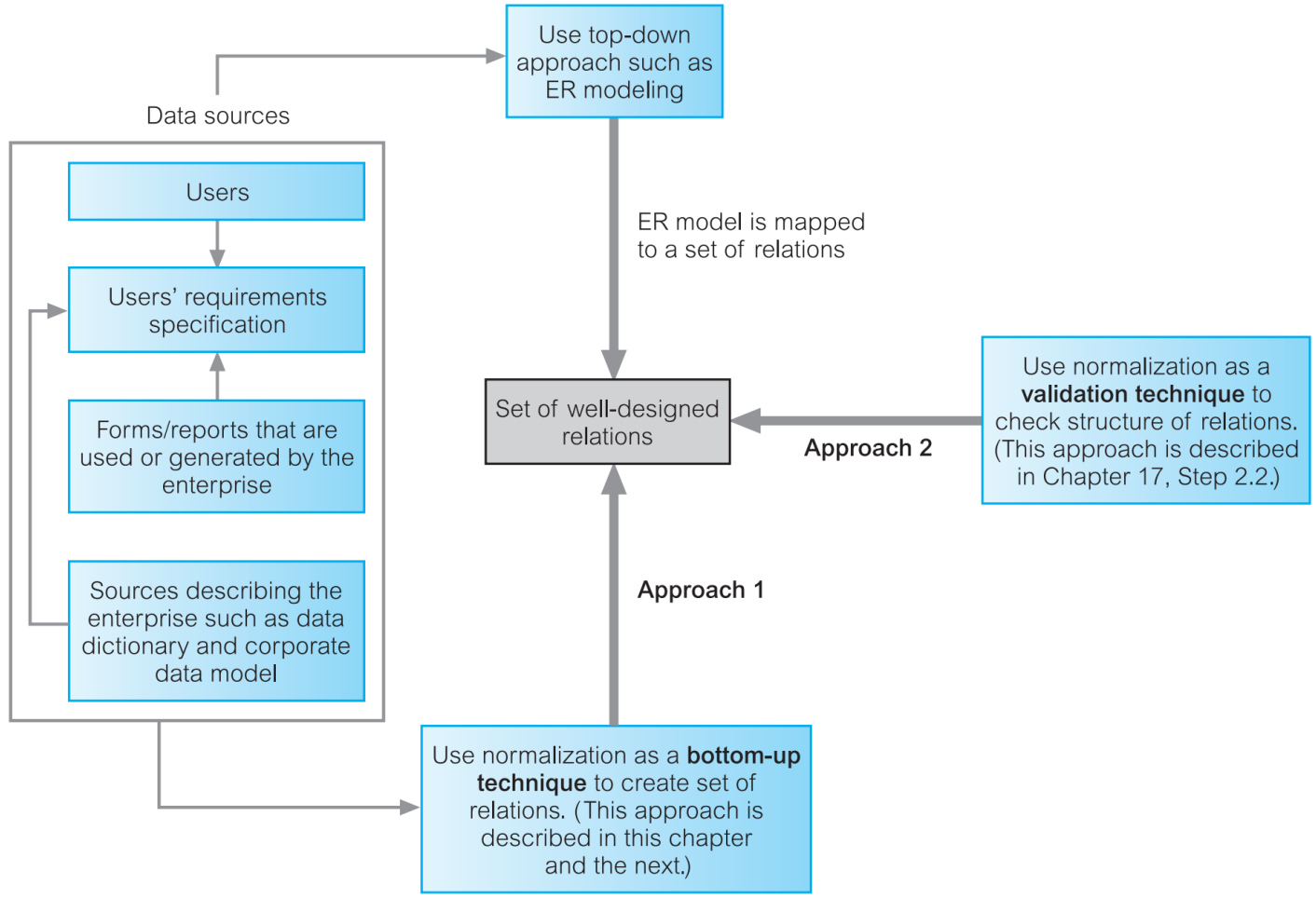

Figure 14.1 How normalization can be used to support database design.

Normalization is a formal technique for analyzing relations based on their primary key (or candidate keys) and functional dependencies (Codd, 1972b). The technique involves a series of rules that can be used to test individual relations so that a database can be normalized to any degree. When a requirement is not met, the relation violating the requirement must be decomposed into relations that individually meet the requirements of normalization.

(Connolly & Begg, 2015: Chapters 1, 4, 14)

first normal form (1NF)	second normal form (2NF).	Third normal form (3NF)
all rows must be unique a relation in which the intersection of each row and column contains one and only one value	A relation that is in first normal form and every noncandidate-key attribute is fully functionally dependent on any candidate key	A relation that is in first and second normal form and in which no non-primary-key attribute is transitively dependent on the primary key.

youtube.com/watch?v=jgUeOjlmOOw

YouTube Search

First Normal Form

- Each field of a table may contain only one item
- All of the data items in a column must mean the same thing
- Each row of the table must be unique
- A table must have no repeating columns

Database Normalisation: First Normal Form

Press Esc to exit full screen

First Normal Form

Student Name	Course Titles
Kevin Drumm	Computer Science, Mathematics, Physics
Murvin Drake	Physics, Chemistry
John Jones, 1234	Music
Sally-Jane Jones	Biology, Economics
David , Married	Mathematics, Physics
Murvin Drake	Physics, Chemistry

Database Normalisation: First Norma

49,863 views • Jan 13, 2019

Computer Science
98.7K subscribers

Database Normalisation: First Normal Form

First Normal Form

ID	Student Name	Marital Status	Course Title
1	Kevin Drumm	Single	Computer Science
1	Kevin Drumm	Single	Mathematics
1	Kevin Drumm	Single	Physics
2	Murvin Drake	Single	Physics
2	Murvin Drake	Single	Chemistry
3	John Jones	Single	Music
4	Sally-Jane Jones	Single	Biology
4	Sally-Jane Jones	Single	Economics
5	David	Married	Mathematics
5	David	Married	Physics
6	Murvin Drake	Single	Physics
6	Murvin Drake	Single	Chemistry

Not perfect table, but shows 1NF

6:44 / 7:26

https://www.youtube.com/watch?v=_K7fcFQowy8&list=RDCMUCSX3MR0gnKDxyXAyljWzm0Q&index=3

Database Normalisation: Second Normal Form

Second Normal Form

- The data must be in First Normal Form
- Each non key field must be about the same thing as the primary key
- Each table must contain data about only one type of thing

Database Normalisation: Second Normal Form

Press Esc to exit full screen

Second Normal Form

Students

ID	First Name	Last Name	Marital Status
1	Kevin	Drumm	Single
2	Murvin	Drake	Single
3	John	Jones	Single
4	Sally-Jane	Jones	Single
5	David	Smith	Married

StudentCourse

ID	Course Title	Grade
1	Computer Science	A
1	Mathematics	B
1	Physics	C
2	Physics	B
2	Chemistry	C
3	Music	C
4	Biology	A
4	Economics	B
5	Mathematics	C
5	Physics	D

Courses

Course Title	Fee	Qualification
Computer Science	£2000	Advanced Level
Mathematics	£2500	Advanced Level
Physics	£1800	Advanced Level
Chemistry	£1800	Advanced Level
Music	£1200	Diploma
Biology	£1000	Certificate
Economics	£1500	Diploma

6:11 / 9:20

Database Normalisation: Second Normal Form

Second Normal Form

ID + Course Title=Composite Key

ID	First Name	Last Name	Marital Status	Course Title	Fee	Qualification	Grade
1	Kevin	Drumm	Single	Computer Science	£2000	Advanced Level	A
1	Kevin	Drumm	Single	Mathematics	£2500	Advanced Level	B
1	Kevin	Drumm	Single	Physics	£1800	Advanced Level	C
2	Murvin	Drake	Single	Physics	£1800	Advanced Level	B
2	Murvin	Drake	Single	Chemistry	£1800	Advanced Level	C
3	John	Jones	Single	Music	£1200	Diploma	C
4	Sally-Jane	Jones	Single	Biology	£1000	Certificate	A
4	Sally-Jane	Jones	Single	Economics	£1500	Diploma	B
5	David	Smith	Married	Mathematics	£2500	Advanced Level	C
5	David	Smith	Married	Physics	£1800	Advanced Level	D

Functionally dependent on the id

2:24 / 9:20

https://www.youtube.com/watch?v=_K7fcFQowy8&list=RDCMUCSX3MR0gnKDxyXAyljWzm0Q&index=3

Third Normal Form

- The data must be in Second Normal Form
- There is no other non key attribute that you would need to change in a table if you changed a non key attribute

Third Normal Form

Students

ID	First Name	Last Name	Marital Status
1	Kevin	Drumm	Single
2	Murvin	Drake	Single
3	John	Jones	Single
4	Sally-Jane	Jones	Single
5	David	Smith	Married

StudentCourse

ID	Course Title	Grade
1	Computer Science	A
1	Mathematics	B
1	Physics	C
2	Physics	B
2	Chemistry	C
3	Music	C
4	Biology	A
4	Economics	B
5	Mathematics	C
5	Physics	D

Courses

Course Title	Fee	Qualification	Teacher ID	Teacher Name
Computer Science	£2000	Advanced Level	1	Miss Lovelace
Mathematics	£2500	Advanced Level	2	Mr Pascal
Physics	£1800	Advanced Level	3	Mr Einstein
Chemistry	£1800	Advanced Level	4	Mr Bunsen
Music	£1200	Diploma	5	Miss Holiday
Biology	£1000	Certificate	6	Mr Darwin
Economics	£1500	Diploma	7	Mr Keynes

Course Title → Teacher ID → Teacher Name

Course Title → Teacher Name → Teacher ID

Transitive dependency: they all depend on each other (e.g if teacher is changed or course title is changed)

Third Normal Form

Students

ID	First Name	Last Name	Marital Status
1	Kevin	Drumm	Single
2	Murvin	Drake	Single
3	John	Jones	Single
4	Sally-Jane	Jones	Single
5	David	Smith	Married

StudentCourse

ID	Course Title	Grade
1	Computer Science	A
1	Mathematics	B
1	Physics	C
2	Physics	B
2	Chemistry	C
3	Music	C
4	Biology	A
4	Economics	B
5	Mathematics	C
5	Physics	D

Courses

Course Title	Fee	Qualification	Teacher
Computer Science	£2000	Advanced Level	1
Mathematics	£2500	Advanced Level	2
Physics	£1800	Advanced Level	3
Chemistry	£1800	Advanced Level	4
Music	£1200	Diploma	5
Biology	£1000	Certificate	6
Economics	£1500	Diploma	7

Teachers

Teacher ID	Teacher Name
1	Miss Lovelace
2	Mr Pascal
3	Mr Einstein
4	Mr Bunsen
5	Miss Holiday
6	Mr Darwin
7	Mr Keynes

https://www.youtube.com/watch?v=_K7fcFQowy8&list=RDCMUCSX3MR0gnKDxyXAyljWzm0Q&index=3

To learn

- Install Oracle SQL Developer

Sources

- Connolly, T. & Begg, C. (2015) DATABASE SYSTEMS A Practical Approach to Design, Implementation, and Management. New York: Pearson.

- Kroenke, D. et al (2018) DATABASE PROCESSING FUNDAMENTALS, DESIGN, AND IMPLEMENTATION. New York: Pearson

- Check: https://www.studytonight.com/dbms/database-normalization.php
- https://www.youtube.com/watch?v=_K7fcFQowy8&list=RDCMUCSX3MR0gnKDxyXAyljWzm0Q&index=3