Database Systems Concepts and Design

CSC201S2/G2

Chapter 4: Normalisation

Introduction

- Normalization is the process of organizing data in a database.
- Includes creating tables and establishing relationships between those tables
- Redundancy and inconsistent dependency.

Unnormalized form(UNF)

- A table that contains one or more repeating groups.
- An attribute or group of attributes within a table that occurs with **multiple values for a single occurrence of the nominated key attributes of that table**.
- A UNF model will suffer problems like **data redundancy** thus it lacks the efficiency of database normalisation.

Example: Repeating Groups

Repeating groups

CustNo	Cname	PropNo	PAddr	RntSt	RntFnsh	Rent	OwnerNo	OName
CR76	John Kay	PG4	6 Lawrence St, Elmont	7/1/10	8/31/06	700	CO40	Tina Murphy
		PG16	5 Nova Dr, East Meadow	9/1/06	9/1/08	900	CO93	Tony Shaw
CR56	Aline Stewart	PG4	6 Lawrence St, Elmont	9/1/02	6/10/04	700	CO40	Tina Murphy
		PG36	2 Manor Rd Scarsdale	8/1/04	12/1/05	750	CO93	Tony Shaw
		PG16	5 Nova Dr, East Meadow	8/1/06	9/1/10	900	CO93	Tony Shaw

Redundant Information

- Data redundancy occurs when the same piece of data is stored in two or more separate places.
- Aim of relational database design is to group attribute into relations to minimize data redundancy and thereby reduce the file storage space required by the implemented base relations.

Example: Data Redundancy

Staff_no	Sname	Position	Salary	Branch_no	Baddress
S1	John	Manager	10000	B005	22 deccan, Pune.
S2	Ann	Assistant	5000	B003	10 bandra, Mumbai.
S3	Suhas	Supervisor	7000	B007	32 Main st, Nasik.
S4	Julie	Assistant	5000	B007	32 Main st, Nasik.
S5	Mary	Assistant	5000	B005	22 deccan, Pune.

- In **Staffbranch** relation there is redundant data. Branch address is repeated for every member of staff located at that branch.

Update Anomalies

- Relations that have redundant data may have problems called **update anomalies**.
- Type of update anomalies are:
  - Insertion
  - Deletion
  - Modification

Example: Update Anomalies

EMP_PROJ (*Emp*#, Proj#, Ename, Pname, No_hours)

Modification Anomaly

- Changing the project name of project number P1 from "Billing" to "Customer-Accounting" may cause this update to be made for all 100 employees working on project P1

Example: Insert Anomalies

EMP_PROJ (*Emp*#, Proj#, Ename, Pname, No_hours)

- Cannot insert a project unless an employee is assigned to .
- Inversely- Cannot insert an employee unless he/she is assigned to a project.

Example: Delete Anomaly

EMP_PROJ (*Emp#*, Proj#, Ename, Pname, No_hours)

When a project is deleted, it will result in deleting all the employees who work on that project. Alternately, if an employee is the sole employee on a project, deleting that employee would result in deleting the corresponding project.

Example: Update Anomalies

- Insert a new staff into the StaffBranch relation;
- Delete a tuple that represents the last member of staff located at a branch B007;
- Change the address of branch B003.

staffNo	sName	position	salary	branchNo	bAddress
SL21	John White	Manager	30000	B005	22 Deer Rd, London
SG37	Ann Beech	Assistant	12000	B003	163 Main St,Glasgow
SG14	David Ford	Supervisor	18000	B003	163 Main St,Glasgow
SA9	Mary Howe	Assistant	9000	B007	16 Argyll St,Aberdeen
SG5	Susan Brand	Manager	24000	B003	163 Main St,Glasgow
SL41	Julie Lee	Assistant	9000	B005	22 Deer Rd, London

Types of Dependencies

Dependencies in DBMS is a relation between two or more attributes:

- Functional Dependency
- Fully-Functional Dependency
- Partial Dependency
- Transitive Dependency

Functional Dependencies

If the information stored in a table can uniquely determine another information in the same table, then it is called **Functional Dependency**.

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

Example: Functional Dependencies

STUDENT

STUD_NO	STUD_NAME	STUD_PHONE	STUD_STATE	STUD_COUNTRY	STUD_AGE
1	RAM	9716271721	Haryana	India	20
2	RAM	9898291281	Punjab	India	19
3	SUJIT	7898291981	Rajsthan	India	18
4	SURESH		Punjab	India	21

STUD_NO -> STUD_NAME and

STUD_NO -> STUD_PHONE hold

*A **STUD_NO** uniquely identifies a **STUD_NAME** and **STUD_PHONE**

STUD_NAME->STUD_STATE does not hold

***Two students can have same name (Like RAM in the below table) and hence same state**

Full-functional Dependencies

Full functional dependency indicates that if A and B are attributes of a relation, B is fully functionally dependent on A if B is functionally dependent on A, **but not on any proper subset of** A.

A non-key attribute depends on the entire primary key, rather than just a portion of it.

Example

supplier_id	item_id	price
1	1	540
2	1	545
1	2	200
2	2	201
1	1	540
2	2	201
3	1	542

{supplier_id, item_id} -> price

*supplier_id nor item_id can uniquely determine the price

*Both supplier_id and item_id together can do so

***Price full-functional depend on** supplier_id and item_id

Partial Dependencies

A functional dependency P→Q is partially dependent if there is some attributes that can be removed from P and the dependency still holds.

A non-key attribute is functionally dependent on only part of the composite primary key, not the entire key.

Example

name	roll_no	course
Ravi	2	DBMS
Tim	3	OS
John	5	Java

{name} -> course

{roll_no} -> course

{name, roll_no} -> course

*Both the attributes name and roll_no alone are able to uniquely identify a course

*The relationship is partially dependent

Transitive Dependencies

When an **indirect relationship causes functional dependency** it is called **Transitive Dependency**.

Attribute B depends on attribute A (A -> B) and C depends on B (B -> C), indirectly establishing a dependency between A and C (A -> C).

Non-key attribute depends on another non-key attribute, which in turn depends on the primary key

Example

roll_no	name	city	zip-code
1	abc	pune	411044
2	jkl	mumbai	400001
3	uvw	pune	411044
4	xyz	delhi	110001

{roll_no} -> city

{city} -> zip-code

{roll_no} -> zip_code

*roll-no = 1 has city=pune and city=pune will have zip-code=411044. So wherever roll-no is 1 , zip-code will be 411044

*The relationship is transitive dependency

Exercise

A	B	C
7	1	8
7	2	5
7	3	5
5	8	8

Which functional dependencies holds relation

1. AB →C && C→ B
2. BC → A && B→C
3. BC →A && A→ C
4. AC→ B && B→C

v	w	x	y	z
7	8	c	9	4
8	7	c	9	4
7	8	c	2	4
7	8	c	2	2

Which functional dependencies holds relation

R(v, w, x, y, z)

1. v → wx
2. yz → x
3. x→ yz

Exercise

A	B	C
a_1	b_1	c_1
a_1	b_1	c_2
a_2	b_1	c_1
a_2	b_1	c_3

List all functional dependencies satisfied by the relation

R (A, B, C)

α	β	γ
6	1	7
4	3	7
4	8	1
6	7	1

Which functional dependencies are holds the relation

1. $\propto \rightarrow \beta$ && $\beta\gamma \rightarrow \propto$
2. $\gamma \rightarrow \beta$ && $\propto \rightarrow \beta$
3. $\beta \rightarrow \gamma$ && $\propto \beta \rightarrow \gamma$
4. $\propto \rightarrow \gamma$ && $\beta\gamma \rightarrow \propto$


Q
&
A

Normalisation

- Database design technique that reduces data redundancy and eliminates undesirable characteristics like Insertion, Update and Deletion Anomalies.
- There are three main reasons to normalize a database.
    - minimize duplicate data,
    - minimize or avoid update anomalies
    - simplify queries.

Normalisation

A process of organizing the data in database to avoid data redundancy, insertion anomaly, update anomaly & deletion anomaly.

Steps to normalize the database:

- First Normal Form (1NF)
- Second Normal Form (2NF)
- Third Normal Form (3NF)
- Boyce-Code Normal Form (BCNF)

First Normal Form (INF)

- All data values are atomic.
- INF is a relation in which the intersection of each row and column contains **one and only one value**.

Approach for removing repeating groups:

- Entering appropriate data in the empty columns of rows containing the repeating data.
- Placing the repeating data, along with a copy of the original key attribute(s), in a separate relation.
- A primary key is identified for the new relation.

Example

CustNo	Cname	PropNo	PAddr	RntSt	RntFnsh	Rent	OwnerNo	OName
CR76	John Kay	PG4	6 Lawrence St, Elmont	7/1/10	8/31/06	700	CO40	Tina Murphy
		PG16	5 Nova Dr, East Meadow	9/1/06	9/1/08	900	CO93	Tony Shaw
CR56	Aline Stewart	PG4	6 Lawrence St, Elmont	9/1/02	6/10/04	700	CO40	Tina Murphy
		PG36	2 Manor Rd Scarsdale	8/1/04	12/1/05	750	CO93	Tony Shaw
		PG16	5 Nova Dr, East Meadow	8/1/06	9/1/10	900	CO93	Tony Shaw

Normalise into 1NF using First approach

ClientNo	propertyNo	cName	pAddress	rentStart	rentFinish	rent	ownerNo	oName
CR76	PG4	John Kay	6 lawrence St,Glasgow	1-Jul-00	31-Aug-01	350	CO40	Tina Murphy
CR76	PG16	John Kay	5 Novar Dr, Glasgow	1-Sep-02	1-Sep-02	450	CO93	Tony Shaw
CR56	PG4	Aline Stewart	6 lawrence St,Glasgow	1-Sep-99	10-Jun-00	350	CO40	Tina Murphy
CR56	PG36	Aline Stewart	2 Manor Rd, Glasgow	10-Oct-00	1-Dec-01	370	CO93	Tony Shaw
CR56	PG16	Aline Stewart	5 Novar Dr, Glasgow	1-Nov-02	1-Aug-03	450	CO93	Tony Shaw

1NF ClientRental relation with the first approach

1NF ClientRental relation with the second approach

ClientNo	cName
CR76	John Kay
CR56	Aline Stewart

ClientNo	propertyNo	pAddress	rentStart	rentFinish	rent	ownerNo	oName
CR76	PG4	6 Lawrence St,Glasgow	1-Jul-00	31-Aug-01	350	CO40	Tina Murphy
CR76	PG16	5 Novar Dr, Glasgow	1-Sep-02	1-Sep-02	450	CO93	Tony Shaw
CR56	PG4	6 Lawrence St,Glasgow	1-Sep-99	10-Jun-00	350	CO40	Tina Murphy
CR56	PG36	2 Manor Rd, Glasgow	10-Oct-00	1-Dec-01	370	CO93	Tony Shaw
CR56	PG16	5 Novar Dr, Glasgow	1-Nov-02	1-Aug-03	450	CO93	Tony Shaw

1NF ClientRental relation with the second approach

Example

Module	Dept	Lecturer	Texts
M1	D1	L1	T1, T2
M2	D1	L1	T1. T3
M3	D1	L2	T4
M4	D2	L3	T1, T5
M5	D2	L4	T6

Unnormalised

Module	Dept	Lecturer	Texts
M1	D1	L1	T1
M1	D1	L1	T2
M2	D1	L1	T1
M2	D1	L1	T3
M3	D1	L2	T4
M4	D2	L3	T1
M4	D2	L3	T5
M5	D2	L4	T6

INF

Exercise

Convert the following relation in 1NF

Product Id	Colour	Price
1	Black, red	Rs. 210
2	Green	Rs. 150
3	Red	Rs. 110
4	Green, blue	Rs. 260
5	Black	Rs. 100

Problems in INF

Module	Dept	Lecturer	Texts
M1	D1	L1	T1
M1	D1	L1	T2
M2	D1	L1	T1
M2	D1	L1	T3
M3	D1	L2	T4
M4	D2	L3	T1
M4	D2	L3	T5
M5	D2	L4	T6

INSERT anomalies

Can't add a module with no texts

UPDATE anomalies

To change lecturer for M1, we have to change two rows

DELETE anomalies

If we remove M3, we remove L2 as well

Second Normal Form (2NF)

- A relation is in second normal form (2NF) if it is in 1NF and no non-key attribute is partially dependent on a candidate key

OR

- Second normal form (2NF) is a relation that is in first normal form and every non-primary-key attribute is fully functionally dependent on the primary key.
- The normalization of 1NF relations to 2NF involves the removal of partial dependencies.
- If a partial dependency exists, remove the function dependent attributes from the relation by placing them in a new relation along with a copy of their determinant.

Second Normal Form (2NF)

ClientNo	propertyNo	cName	pAddress	rentStart	rentFinish	rent	ownerNo	oName
CR76	PG4	John Kay	6 lawrence St,Glasgow	1-Jul-00	31-Aug-01	350	CO40	Tina Murphy
CR76	PG16	John Kay	5 Novar Dr, Glasgow	1-Sep-02	1-Sep-02	450	CO93	Tony Shaw
CR56	PG4	Aline Stewart	6 lawrence St,Glasgow	1-Sep-99	10-Jun-00	350	CO40	Tina Murphy
CR56	PG36	Aline Stewart	2 Manor Rd, Glasgow	10-Oct-00	1-Dec-01	370	CO935	Tina Shaw
CR56	PG16	Aline Stewart	5 Novar Dr, Glasgow	1-Nov-02	1-Aug-03	450	CO93	Tony Shaw

Second Normal Form (2NF)

- Client(**clientNo**, cName)
- Property(**propertyNo**, pAddress, rent, ownerNo, oName)
- Client-Property(**clientNo, propoertyNo, rentStart, rentFinish**)

Third Normal Form (3NF)

A table design is said to be in 3NF if both the following conditions hold:

- Table must be in 2NF
- Transitive functional dependency of non-prime attribute on any super key should be removed.

OR

A relation is in third normal form (3NF) if it is in 2NF and no non-key attribute is transitively dependent on a candidate key

The normalization of 2NF relations to 3NF involves the removal of transitive dependencies by placing the attribute(s) in a new relation along with a copy of the determinant.

Third Normal Form (3NF)

ClientNo	propertyNo	cName	pAddress	rentStart	rentFinish	rent	ownerNo	oName

- propertyNo → ownerNo
- ownerNo → oName
- So **oName** transitively depend on **propertyNo**

Third Normal Form (3NF)

- Client (clientNo, cName)
- Owner (ownerNo, oName)
- Property(propertyNo, pAddress, rent, ownerNo)
- Client-Property (clientNo, propertyNo, rentStart, rentFinish)

Exercise

Consider the Students table:

1. Identify any repeating groups and functional dependences
2. Show all the intermediate steps to derive the **third normal** form for STUDENT.

Alpha	**Name**	**Email**	**Courses**	**Marks**
100111	John Doe	doe@usna.edu	NN204, SI204, IT221	2,3,3
092244	Matt Smith	smith@usna.edu	SM223, EE301	4,4
113221	Melinda Black	black@usna.edu	SI204	3
090112	Tom Johnson	Johnson@usna.edu	NN204, SI204, IT221	4,2,3

Exercise

Alpha	Name	Email	Courses	Marks
100111	John Doe	doe@usna.edu	NN204	2
100111	John Doe	doe@usna.edu	SI204	3
100111	John Doe	doe@usna.edu	IT221	3
092244	Matt Smith	smith@usna.edu	SM223	4
092244	Matt Smith	smith@usna.edu	EE301	4
113221	Melinda Black	black@usna.edu	SI204	3
090112	Tom Johnson	Johnson@usna.edu	NN204	4
090112	Tom Johnson	Johnson@usna.edu	SI204	2
090112	Tom Johnson	Johnson@usna.edu	IT221	3

In 2NF

- Student (Alpha, Name)
- StudentCourse (Alpha,Courses,Marks)

In 3NF

- Student (Alpha, Name, Email)
- StudentCourse (Alpha,Courses, Marks)

Exercise

Consider the Patients table:

1. Identify any repeating groups and functional dependences
2. Show all the intermediate steps to derive the **third normal** form for **PATIENT**.

Patient no	Patient name	Doctor no	Doctor name	Appointment date	Consultant Name	Consultant address	Sample
01027	Grist	919	Robinson	3/9/2014	Farnes	Acadia Rd	blood
				20/12/2014	Farnes	Acadia Rd	none
				10/10/2014	Edwards	Beech Ave	urine
08023	Daniels	818	Seymour	3/9/2014	Farnes	Acadia Rd	none
				3/9/2014	Russ	Fir St	sputum
191146	Falken	717	Ibbotson	4/10/2014	Russ	Fir St	blood
001239	Burgess	818	Seymour	5/6/2014	Russ	Fir St	sputum
007249	Lynch	717	Ibbotson	9/11/2014	Edwards	Beach Ave	none

Exercise

PATIENT table is in INF

Patient no	Patient name	Doctor no	Doctor name	Appointment date	Consultant Name	Consultant address	Sample
01027	Grist	919	Robinson	3/9/2014	Farnes	Acadia Rd	blood
01027	Grist	919	Robinson	20/12/2014	Farnes	Acadia Rd	none
01027	Grist	919	Robinson	10/10/2014	Edwards	Beech Ave	urine
08023	Daniels	818	Seymour	3/9/2014	Farnes	Acadia Rd	none
08023	Daniels	818	Seymour	3/9/2014	Russ	Fir St	sputum
191146	Falken	717	Ibbotson	4/10/2014	Russ	Fir St	blood
001239	Burgess	818	Seymour	5/6/2014	Russ	Fir St	sputum
007249	Lynch	717	Ibbotson	9/11/2014	Edwards	Beach Ave	none

Exercise

Consider the Pets table:

1. Identify any repeating groups and functional dependences
2. Show all the intermediate steps to derive the **third normal** form for **PETS**.

Pet Id	Pet Name	Pet Type	Pet Age	Owner	Visit Date	Procedure id	Procedure name
246	Rover	Dog	12	Sam Cook	JAN 13/2002 MAR 27/2002 APR 02/2002	01 10 05	Rabies Vaccination Examine And Treat Wound Heart Worm Test
298	Spot	Dog	2	Terry Kim	JAN 21/2002 MAR 10/2002	08 05	Tetanus Vaccination Heart Worm Test
341	Morris	Cat	4	Sam Cook	JAN 23/2001 JAN 13/2002	01 01	Rabies Vaccination Rabies Vaccination
519	Tweedy	Bird	2	Terry Kim	APR 30/2002 APR 30/2002	20 12	Annual Check Up Eye Wash

Exercise

Pet Id	Pet Name	Pet Type	Pet Age	Owner	Visit Date	Procedure id	Procedure name
246	Rover	Dog	12	Sam Cook	JAN 13/2002	01	Rabies Vaccination
246	Rover	Dog	12	Sam Cook	MAR 27/2002	10	Examine And Treat Wound
246	Rover	Dog	12	Sam Cook	APR 02/2002	05	Heart Worm Test
298	Spot	Dog	2	Terry Kim	JAN 21/2002	08	Tetanus Vaccination
298	Spot	Dog	2	Terry Kim	MAR 10/2002	05	Heart Worm Test
341	Morris	Cat	4	Sam Cook	JAN 23/2001	01	Rabies Vaccination
341	Morris	Cat	4	Sam Cook	JAN 13/2002	01	Rabies Vaccination
519	Tweedy	Bird	2	Terry Kim	APR 30/2002	20	Annual Check Up
519	Tweedy	Bird	2	Terry Kim	APR 30/2002	12	Eye Wash

Exercise

PET table is in 3NF

Pet (Pet id, Pet name, Pet type, Pet age, Owner)

PetOwner (Pet id, Visited date, Procedure id)

Procedure(Procedure id, Procedure name)

Boyce-Codd Normal Form (BCNF)

Advance version of 3NF referred as 3.5NF.

BCNF is stricter than 3NF.

A table complies with BCNF if it is in 3NF and for every functional dependency X->Y, X should be the super key of the table.

Example

Student	Course	Teacher
Aman	DBMS	AYUSH
Aditya	DBMS	RAJ
Abhinav	E-COMM	RAHUL
Aman	E-COMM	RAHUL
abhinav	DBMS	RAJ

Primary key: {Student, Course}

Functional dependency

{student, course} -> Teacher

Teacher-> Course

teacher is not super key but determines course.

Example

After decomposing it into Boyce-Codd normal form

Student	Course
Aman	DBMS
Aditya	DBMS
Abhinav	E-COMM
Aman	E-COMM
Abhinav	DBMS

Course	Teacher
DBMS	AYUSH
DBMS	RAJ
E-COMM	RAHUL

Exercise

After decomposing it into Boyce-Codd normal form

Stu_ID	Stu_Branch	Stu_Course	Branch_Number	Stu_Course_No
101	Computer Science & Engineering	DBMS	B_001	201
101	Computer Science & Engineering	Computer Networks	B_001	202
102	Electronics & Communication Engineering	VLSI Technology	B_003	401
102	Electronics & Communication Engineering	Mobile Communication	B_003	402

Exercise

The table below shows an extract from a tour operator's data on travel agent bookings. Derive the 3NF of the data, showing all the intermediate steps

Batch no	Agent no	Agent name	holiday code	cost	quantity booked	Airport code	airport name
1	76	Bairns travel	B563	363	10	1	Luton
			B248	248	20	12	Edinburgh
			B428	322	18	11	Glasgow
2	142	Active Holidays	B563	363	15	1	Luton
			C930	568	2	14	Newcastle
			A270	972	1	14	Newcastle
			B728	248	5	12	Edinburgh
3	76	Bairns travel	C930	568	11	1	Luton
			A430	279	15	11	Glasgow


Q
&
A