Design Theory
Normalization and Normal Forms

Hui Chen [a]

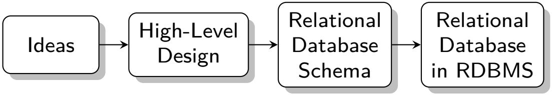
[a]CUNY Brooklyn College, Brooklyn, NY, USA

March 29, 2022

Outline

Outline

Overview


Ideas
High-Level Design
Relational Database Schema
Relational Database in RDBMS

A Design Challenge

There are a variety of ways that we can design relational schema – there is a space for improvement.

- Problem. we are trying to combine too much into one relation $\rightarrow$ maintenance problems called *anomaly*.
- Problem. we are trying to create to many relations $\rightarrow$ *difficult to answer queries or retrieve the data*

How do we identify such design problem and make improvements? – a design trade-off must be made.

- A well developed theory – dependencies and normalization
- *Normalization* – the process of converting a relation into a normal form.
  - The process usually consists of decomposing a table into two or more tables with fewer attributes
  - When normalizing relations, we are generally sacrificing retrieval speed to prevent data maintenance problems – a trade-off

Outline

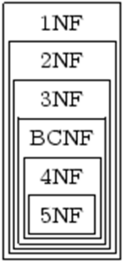
Normalization

- Redundancy in the database may lead to anomalies.
- The normalization is a technique to reduce redundancy.
  - It is a decomposition process to split tables up, so that the relation is in a *normal form*.
  - The splitting is performed carefully so that no information is lost
  - There different level of normal forms, the higher the normal form is, the lower the redundancy.

Normal Forms

Normal forms build on each other

- First Normal Form (1NF)
- Second Normal Form (2NF)
- Third Normal Form (3NF)
- Boyce-Codd Normal Form (BCNF)
- Fourth Norm Form (4NF) – not to discuss, on your own
- Fifth Norm Form (5NF/PJNF) – not to discuss, on your own
- N. Domain-Key Normal Form (DKNF) – not to discuss, on your own



Note that a relation in a higher normal form is always in a lower normal (observe the Venn diagram).

We limit the discussion to 1NF – BCNF.

First Normal Form (1NF)

1NF: A relation R is in first normal form (1NF) if and only if all underlying domains contain *atomic values only*

What does this mean?

- No duplicate rows – Each table has a key: minimal set of attributes which can uniquely identify a record
- No multi-value attributes allowed – The values in each column of a table are atomic, i.e., no table of tables.
- There are no repeating groups – two columns do not store similar information in the same table.

Example 1: 1NF or not?

EMPLID	Name	Course	Grades
1112223333	Sasha	CISC3810	A
1112223333	Sasha	CISC3810	A

Example 1: 1NF or not?

EMPLID	Name	Course	Grades
1112223333	Sasha	CISC3810	A
1112223333	Sasha	CISC3810	A

- Not in 1NF! Because it violates
  - No duplicate rows – Each table has a key: minimal set of attributes which can uniquely identify a record

Example 2: 1NF or not?

EMPLID	Name	Grades	Courses
1112223333	Sasha	A,B	CISC3115,CISC3810
1112224444	John	B,A	CISC3171,CISC3810

Example 2: 1NF or not?

EMPLID	Name	Grades	Courses
1112223333	Sasha	A,B	CISC3115,CISC3810
1112224444	John	B,A	CISC3171,CISC3810

- Not in 1NF! It violates,
  - No multi-value attributes allowed – The values in each column of a table are atomic, i.e., no table of tables.

Example 3: 1NF or Not

EMPLID	Name	Course1	Course2	Grade1	Grade2
1112223333	Sasha	CISC3115	CISC3810	A	B
1112224444	John	CISC3171	CISC3810	B	A

Example 3: 1NF or Not

EMPLID	Name	Course1	Course2	Grade1	Grade2
1112223333	Sasha	CISC3115	CISC3810	A	B
1112224444	John	CISC3171	CISC3810	B	A

- Not in 1NF! It violates,
  - There are no repeating groups – two columns do not store similar information in the same table.

Issues with Relations *not* in 1NF

What issues could there be with regard to the relations not in 1NF?

Can these happen and under what scenario?

- Insertion anomaly
- Deletion anomaly
- Update anomaly

Issues with Relations in 1NF

Is the following relation in 1NF?

InvNo	InvDate	CustNo	CustName	ItemNo	ItemName	ItemPrice	Qty
1001	04/04/22	212	Will	1	Screw	199	5
1001	04/04/22	212	Will	3	Bolt	399	5
1001	04/04/22	212	Will	5	Washer	99	9
1002	04/11/22	225	Chris	1	Screw	199	10
1002	04/11/22	225	Chris	2	Nut	499	6
1003	04/11/22	240	Lee	1	Screw	199	4
1003	04/11/22	240	Lee	2	Nut	499	3
1004	04/12/22	218	Latasha	4	Hammer	999	8

- Does it have redundant data?
  - What FDs hold?
  - What are super keys and keys?
- What anomalies may occur?

Issues with Relations in 1NF

- A table in 1NF may have redundant data.
- A table in 1NF does not show data consistency and integrity in the long run due to the anomalies.

Second Normal Form (2NF)

2NF: A relation R is in second normal form (2NF) if and only if it is in 1NF and every non-key attribute is *fully* dependent on the key

What does this mean?

- All requirements for 1NF must be met.
- FD holds: key $\rightarrow$ non-key attributes
- FD should not holds: part of key $\rightarrow$ part of non-key attributes

Example: 1NF, 2NF, or Neither?

EMPLID	Name	Course#	CourseName	Credit	Grade
1112223333	Amy	CISC 1115	Java I	5	A
1112223334	Latasha	CISC 3115	Java II	4	A

Example: 1NF, 2NF, or Neither?

EMPLID	Name	Course#	CourseName	Credit	Grade
1112223333	Amy	CISC 1115	Java I	5	A
1112223334	Latasha	CISC 1115	Java I	5	B
1112223334	Latasha	CISC 3115	Java II	4	A

- 1NF but not 2NF. Why?

Example: 1NF, 2NF, or Neither?

EMPLID	Name	Course#	CourseName	Credit	Grade
1112223333	Amy	CISC 1115	Java I	5	A
1112223334	Latasha	CISC 1115	Java I	5	B
1112223334	Latasha	CISC 3115	Java II	4	A

- 1NF but not 2NF. Why?
- Key: $\{EMPLID, Course\#\}$, which means, $\{EMPLID, Course\#\} \to \{Name, CourseName, Credit\}$
- FD holds, but it should not $Course\# \to \{CourseName, Credit\}$ since $\{Course\#\} \subset \{EMPLID, Course\#\}$ and $\{CourseName, Credit\} \subset \{Name, CourseName, Credit, Grade\}$:

Normalizing 1NF to 2NF

Convert 1NF to 2NF

- Redundant data across multiple rows of a table must be moved to a separate table.
- The resulting tables must be related to each other by use of foreign key.

Example: Normalizing 1NF to 2NF

EMPLID	Name	Course#	CourseName	Credit	Grade
1112223333	Amy	CISC 1115	Java I	5	A
1112223334	Latasha	CISC 1115	Java I	5	B
1112223334	Latasha	CISC 3115	Java II	4	A

$\downarrow$

EMPLID	Name	Course#	Grade
1112223333	Amy	CISC 1115	A
1112223334	Latasha	CISC 1115	B
1112223334	Latasha	CISC 3115	A

Course#	CourseName	Credit
CISC 1115	Java I	5
CISC 3115	Java II	4

Example: Normalizing 1NF to 2NF

Is there any additional way to normalize the following relation in 1NF to those in 2NF?

EMPLID	Name	Course#	CourseName	Credit	Grade
1112223333	Amy	CISC 1115	Java I	5	A
1112223334	Latasha	CISC 1115	Java I	5	B
1112223334	Latasha	CISC 3115	Java II	4	A

Summary

Normal Form	Characteristics
1NF	simple table, no repeating groups, and PK identified
2NF	1NF and no partial dependencies

Issues with Relations in 2NF

Is the following relation in 2NF? The relation is about invoices and customers of a business. The business assigns invoice number uniquely and each customer gets a unique customer number.

InvNo	InvDate	CustNo	CustName
1001	04/02/22	212	Will
1002	04/03/22	233	Amy
1003	04/03/22	244	Lee
1004	04/04/22	285	Emma

- Does it have redundant data?
  - What FDs hold?
  - What are super keys and keys?
- What anomalies may occur?

Issues with Relations in 2NF

Is the following relation in 2NF? The relation is about invoices and customers of a business. The business assigns invoice number uniquely and each customer gets a unique customer number.

<u>InvNo</u>	InvDate	CustNo	CustName
1001	04/02/22	212	Will
1002	04/03/22	233	Amy
1003	04/03/22	244	Lee
1004	04/04/22	285	Emma

2NF: key is $InvNo$; although $CustNo \to CustName$, $\{CustNo\} \not\subset \{InvNo\}$

- The following FDs holds among the others: $InvNo \to CustNo$ and $CustNo \to CustName$
- Update anomaly: updating CustNo but forgetting to update CustName will cause inconsistency

Issues with Relations in 2NF

A relation in 2NF may satisfy the following property,

- transitive dependency: C is transitively dependent on A if there exists B such that: $A \rightarrow B$ and $B \rightarrow C$.

As a result, update/delete anomaly may occur when some attribute is transitively depends on the key.

Third Normal Form (3NF)

A relation R is in third normal form (3NF) if and only if it is in 2NF and every non-key attribute is non-transitively dependent on the key.

What does this mean?

- All requirements for 2nd NF must be met.
- Given key K, there does not exist A and B where $K \neq A$, $K \neq B$, and $A \neq B$ such that $K \rightarrow A$ and $A \rightarrow B$

Normalizing 2NF to 3NF

Convert 2NF to 3NF

- Eliminate fields that transitively depend on the key;
- that is, any field that is dependent not only on the key but also on another non-key field must be moved to another table.
- The resulting tables must be related to each other by use of foreign key.

Example: Normalizing 2NF to 3NF

InvNo	InvDate	CustNo	CustName
1001	04/02/22	212	Will
1002	04/03/22	233	Amy
1003	04/03/22	244	Lee
1004	04/04/22	285	Emma

↓

InvNo	InvDate	CustNo
1001	04/02/22	212
1002	04/03/22	233
1003	04/03/22	244
1004	04/04/22	285

CustNo	CustName
212	Will
233	Amy
244	Lee
285	Emma

Summary

Normal Form	Characteristics
1NF	simple table, no repeating groups, and PK identified
2NF	1NF and no partial dependencies
3NF	2NF and no transitive dependencies

Issues with Relations in 3NF

Consider a database for scheduling college classes and we have a relation as follows,

Building	Room	StartTime	EndTime	Instructor
IH	1121	11:00	12:15	Amy
IH	1121	09:30	10:45	Will
IA	325	09:30	10:45	John
IA	325	11:00	12:15	Will

Is this relation in 3NF?

Issues with Relations in 3NF

Consider a database for scheduling college classes and we have a relation as follows,

Building	Room	StartTime	EndTime	Instructor
IH	1121	11:00	12:15	Amy
IH	1121	09:30	10:45	Will
IA	325	09:30	10:45	John
IA	325	11:00	12:15	Will

Is this relation in 3NF?

- Is this relation in 1NF
- Is this relation in 2NF
- Is this relation in 3NF

Issues with Relations in 3NF

Consider a database for scheduling college classes and we have a relation as follows,

Building	Room	StartTime	EndTime	Instructor
IH	1121	11:00	12:15	Amy
IH	1121	09:30	10:45	Will
IA	325	09:30	10:45	John
IA	325	11:00	12:15	Will

Is this relation in 1NF? ✓

- No duplicate rows – keys?
  - Building, Room, StartTime (why?)
  - Building, Room, EndTime (why?)
  - StartTime, Instructor (why?)
  - EndTime, Instructor (why?)
- Similar columns (no multi-valued attributes)? ✓
- No repeating groups? ✓

Issues with Relations in 3NF

Consider a database for scheduling college classes and we have a relation as follows,

Building	Room	StartTime	EndTime	Instructor
IH	1121	11:00	12:15	Amy
IH	1121	09:30	10:45	Will
IA	325	09:30	10:45	John
IA	325	11:00	12:15	Will

Is this relation in 2NF? ✓

- No partial dependencies? The following FD's are not partial FD's because the determinants (left-hand-sides) are keys and there does not exist a non-trivial FD whose determinant is a proper subset of the determinants and the determinant functionally determines a non-key attribute.
  - Building, Room, StartTime $\rightarrow$ Building, Room, StartTime, EndTime, Instructor
  - Building, Room, EndTime $\rightarrow$ Building, Room, StartTime, EndTime, Instructor
  - StartTime, Instructor $\rightarrow$ Building, Room, StartTime, EndTime, Instructor
  - EndTime, Instructor $\rightarrow$ Building, Room, StartTime, EndTime, Instructor

Issues with Relations in 3NF

Consider a database for scheduling college classes and we have a relation as follows,

Building	Room	StartTime	EndTime	Instructor
IH	1121	11:00	12:15	Amy
IH	1121	09:30	10:45	Will
IA	325	09:30	10:45	John
IA	325	11:00	12:15	Will

Is this relation in 3NF? ✓

- transitive dependency? – Is there is a non-key attribute that depends on something other than a key?

Issues with Relations in 3NF

Consider a database for scheduling college classes and we have a reliation as follows,

Building	Room	StartTime	EndTime	Instructor
IH	1121	11:00	12:15	Amy
IH	1121	09:30	10:45	Will
IA	325	09:30	10:45	John
IA	325	11:00	12:15	Will

Is this relation in 3NF? ✓

- Although the following FD's exist, Instructor is not a key and the right-hand-sides are part of keys – there does not exist a transitive FD.
  - Instructor $\rightarrow$ {Building, Room, StartTime}
  - Instructor $\rightarrow$ {Building, Room, EndTime}

Issues with Relations in 3NF

Consider a database for scheduling college classes and we have a relation as follows,

Building	Room	StartTime	EndTime	Instructor
IH	1121	11:00	12:15	Amy
IH	1121	09:30	10:45	Will
IA	325	09:30	10:45	John
IA	325	11:00	12:15	Will

Is this relation in 3NF? ✓

- But due to the existence of these FD's
  - Instructor $\rightarrow$ {Building, Room, StartTime}
  - Instructor $\rightarrow$ {Building, Room, EndTime}
- If we change an instructor's name without checking on meeting location and time for the rows for the instructor, there is a chance we put the instructor at two locations at the same time!
- That is an update anomaly!

Boyce-Codd Normal Form (BCNF)

A relation R is in Boyce-Codd normal form (BCNF) if for every nontrivial functional dependency $X \to A$ where X is a key of R.

What does this mean?

- Anything but the key – no attribute depends on anything other than a key (excluding trivial dependencies)

Normalizing 3NF to BCNF

Convert 3NF to BCNF

- To put the relation in BCNF, create a separate table based on the functional dependency $X \to$ that violates BCNF.
- For this example, remove (Instructor, Building, Room, StartTime) to a separate relation.
- *Or* remove (Instructor, Building, Room, EndTime) to a separate relation.
- Use the foreign key constraint to Link the two relations

Example: Normalizing 3NF to BCNF

Building	Room	StartTime	EndTime	Instructor
IH	1121	11:00	12:15	Amy
IH	1121	09:30	10:45	Will
IA	325	09:30	10:45	John
IA	325	11:00	12:15	Will

↓

Building	Room	StartTime	EndTime
IH	1121	11:00	12:15
IH	1121	09:30	10:45
IA	325	09:30	10:45
IA	325	11:00	12:15

Building	Room	StartTime	Instructor
IH	1121	11:00	Amy
IH	1121	09:30	Will
IA	325	09:30	John
IA	325	11:00	Will

Outline

Properties of Decomposition

Discussed a property of decomposition

- Elimination of Anomalies
  - Decompose a relation to normal forms to reduce redundancies; which reduces chances of anomalies.

Not discussed what other properties we should have – You should continue to explore these on your own.

- Recoverability of information – can we recover the original relation from the tuples in its decomposition?
- Preservation of dependencies – can we satisfy the original functional dependencies when we reconstruct the original relation from the decomposition by joining?

Comparison of Normal Forms

Also explore more on your own

Property	3NF	BCNF	4NF
Eliminate redundancy due to FD's	No[2]	Yes	Yes
Eliminates redundancy due to MVD's[1]	No	No	Yes
Preserves FD's	Yes	No[3]	No[3]
Preserves MVD's[4]	No	No	No

[1]: MVD – multivalued dependencies
[2]: Although "No", 3NF is often enough to eliminate this redundancy.
[3]: BCNF does not guarantee preservation of FD's, but in typical cases (or often) the dependencies are preserved.
[4]: None of the normal forms guarantee preservation of MVD's, although in typical cases (often), the dependencies are preserved.

Outline

Summary

Normal Form	Characteristics
1NF	simple table, no repeating groups, and PK identified
2NF	1NF and no partial dependencies
3NF	2NF and no transitive dependencies
BCNF	Every determinant is a key (nothing but the key)

On your own

- Properties of decomposition

Outline

Assignment

Let's work on an assignment using paper and pencil/pen ...