CCT396, Fall 2011

Database Design and Implementation

Yuri Takhteyev

University of Toronto

Week 6

Normalization

Two Approaches

1. Start with something, then fix it

2. Start by modeling
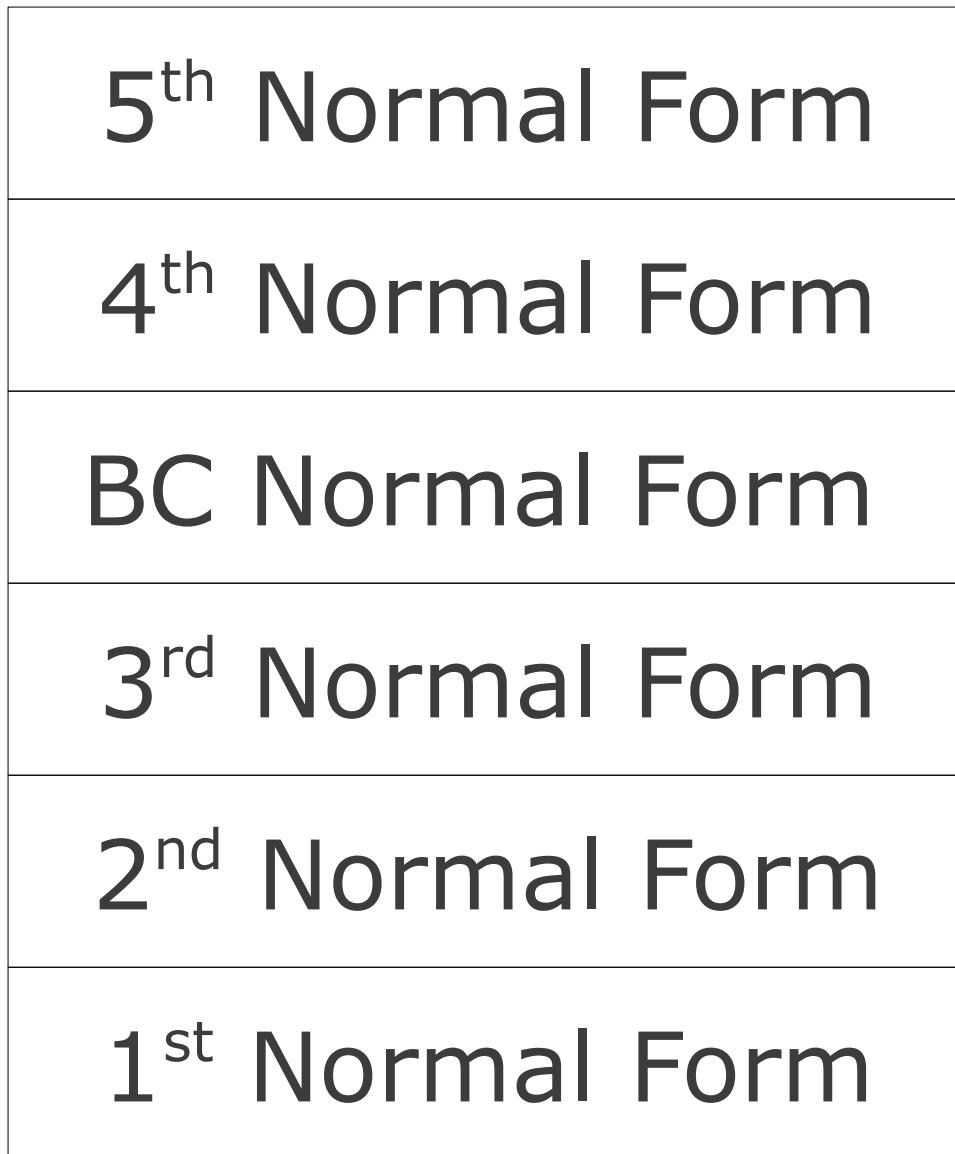
Normalization

What Problems?

problem type	cause	solution
update	redundancy	decomposition
insertion	redundancy	decomposition
deletion	redundancy	decomposition

Normal Forms

5th Normal Form
4th Normal Form
BC Normal Form
3rd Normal Form
2nd Normal Form
1st Normal Form

fixing weird issues

you can do it!

trivial!

1NF

No multi-valued attributes

pet_id	owner	names
1	Bob	"Slim", "the Serpent", "Ribbon"
2	Gwen	"Fluffy", "Big Dog"

The Wrong Solution

pet_id	owner	name1	name2	name3
1	Bob	"Slim"	"the Serpent"	"Ribbon"
2	Gwen	"Fluffy"	"Big Dog"	

The Right Solution

pet_id	owner
1	Bob
2	Gwen

pet_id	names
1	Slim
1	the Serpent
1	Ribbon
2	Fluffy
2	Big Dog

2NF and 3NF

Getting rid of
"functional dependencies"

pet_id	pet_name	owner_email	owner_name
...	...	...	...
...	...	...	...

pet_id → pet_name, owner_email, owner_name

owner_email → owner_name

Functional Dependency

pet_id → owner_name

"owner_name is functionally
   depend on pet_id"

"pet_id functionally determines
   owner_id"

"owner_name describes pet_id"

Examples

(movie_id, title, year, rating)

movie → title, year, rating

Examples

(building_code, room_no, session_code, time_slot, course_code)

building_code, room_no, session_code, time_slot → course_code

Examples

(building_code, room_no, session_code, time_slot, course_code, course_title)

building_code, room_no, session_code, time_slot →
course_code, course_title

course_code → course_title

Examples

(building_code, room_no, session_code, time_slot, course_code, course_title, year, instructor_utorid, instructor_name)

Examples

(building_code, room_no, session_code, time_slot, course_code, course_title, year, instructor_utorid, instructor_name, leap_year, instructor_employee_no, instructor_rank, num_students, room_capacity)

Full Dependency

a full key → attribute

    (This is what we want to see.)

"Each attribute must describe the key, the whole key, and nothing but the key. So help me Codd."

Partial Dependency

part of a key → non-key attribute

(building_code, room_number,
  building_name, capacity)

Keys: (building_code, room_number)

A partial dependency:
  building_code → building_name

Examples

(pet_id, pet_name, owner_name)

(gadget_id, agent_id)

(city_name, province_code,
province_name, population)
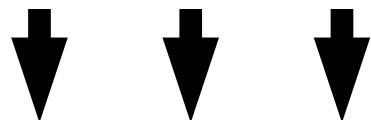
2nd Normal Form

No partial dependencies

Getting to 2NF

Decomposition:

(building_code, room_number,
  building_name, capacity)

⬇ ⬇ ⬇

(building_code, room_number,
  capacity)
(building_code, building_name)

Transitive Dependency

non-key attribute → non-key attribute (a simplified definition)

(pet_id, pet_name, species_id, species_name)

Keys: pet_id

A transitive dependency:
 species_id → species_name
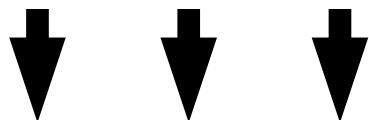
3rd Normal Form

No transitive dependencies

Getting to 3NF

Decomposition:

(pet_id, pet_name, species_id,
 species_name)

↓ ↓ ↓

(pet_id, pet_name, species_id)
(species_id, species_name)

Examples

(building_code, room_no, session_code, time_slot, course_code, course_title, year, instructor_utorid, instructor_name, leap_year, instructor_employee_no, instructor_rank, num_students, room_capacity)

Examples

(building_code, room_no, session_code,
 time_slot, course_code, instructor,
 num_students)

(building_code, room_no, room_capacity)

(session_code, year)

(year, leap_year)

(course_code, course_title)

(instructor_utorid, instructor_name,
 instructor_employee_no, instructor_rank)

Better Yet

(room_id, building_code, room_no, room_capacity)

(room_id, session_code, time_slot, course_code, instructor, num_students)

(session_code, year)

(year, leap_year)

(course_code, course_title)

(instructor_utorid, instructor_name, instructor_employee_no, instructor_rank)

Boyce-Codd NF

"All determinants must be candidate keys."

(Closes a "loophole" in 3NF.)

BCNF

(instructor_id, instructor_utorid,
 time_slot)

Keys:
    (instructor_id, time_slot)
    (instructor_utoroid, time_slot)

instructor_id → instructor_utorid
instructor_utorid → instructor_id

(3NF, but not BCNF)

3NF vs. BCNF

It is always possible to decompose a table into 3NF, but not always into BCNF.

4NF

Similar to BCNF but based on multivalued dependencies ($\twoheadrightarrow$).

Usually implies mistaking a binary relationships for ternary.

Examples tend to look contrived.

Not going to be on the exam.

5NF

We could break this table up and then re-create it with a join. (A simplified definition.)

Again, examples tend to look contrived.

Not going to be on the exam.

Questions?

Alter Table: Add

```
alter table «table»
add column «column» «type»;
```

```
alter table pet
add column pet_id integer;
```

Alter Table: Drop

```
alter table «table»
drop column «column»;
```

```
alter table pet
drop column pet_id;
```

Alter Table: Modify

```
alter table «table»
modify column «column» «type»;
```

```
alter table pet
modify column pet_id char(5);
```

Alter Table: Rename

```
alter table «old_name»
rename «new_name»;
```

```
alter table pet
rename animal;
```

Adding Constraints

```
alter table «table»
add constraint foreign key
«foreign key constraint»;
```

```
alter table pet
add constraint foreign key
(owner_id) references
owner(owner_id);
```

Alternatively...

Drop the table.
Create a new one.

The Final Project

Get started!

Week 3 Exercises

Madame Z's Database Revisited