IT420: Database Management and Organization

Normalization
(Chapter 3)

Announcements

- Department coin design contest deadline - February 6
- 6-week exam – Monday, February 12
- Lab 4 – SQL
  - SQL Server:
    ALTER TABLE *tname* ADD ~~COLUMN~~ *newCol* int NOT NULL

SQL is hot!

http://money.cnn.com/galleries/2007/news/0702/gallery.jobs_in_demand/6.html

10 jobs: Big demand, good pay

Employee recruiting firms MRINetwork and Spherion helped us pull togeth demand for good candidates outstrips the supply.

By Jeanne Sahadi, CNNMoney.com senior writer

< BACK NEXT > 6 of 7

Information technology

7. SQL database administrators

8 . .NET and Java developers

Thanks to an increased concentration of web applications and online services that personalize and store users' information, structured query language (SQL) database administrators (DBAs) are in demand. They can command $100,000 and up in major cities like New York , and between $75,000 and $85,000 in smaller ones such as Des Moines.

And as in 2006, developers who are expert users of Java or Microsoft's software programming language .NET are still in demand and can command even higher pay now, on par with the salaries of SQL database administrators.

Facebook

- Relational databases are accessed in much the same way across the board: SQL. Learning how SQL works is crucial to getting anything done in databases, and any GUI is largely a wrapper around the SQL statements one uses to make those actions happen.
- Knowing a little about database design (layout, B-trees, file storage, normalization) is good, mostly for helping you understand good queries.
- We run the LAMP stack here, so we primarily use MySQL databases across the site. I hope this helps a little.
- Another good motivation may be found in the requirements for most engineering positions here on http://www.facebook.com/jobs.php#Opportunities ;)

Database Design Process

- Requirements analysis
- Conceptual design: Entity-Relationship Model
- Logical design: transform ER model into relational schema
- Schema refinement: Normalization
- Physical tuning

Goal

- Understand normal forms

Premise

- We have one or more tables with data
- The data is to be stored in a new database
- QUESTION: keep or change tables structure?

Data Redundancy

Number	LastName	FirstName	Email	Company	Wing
190	Smith	John	jsmith@usna.edu	12	2
673	Doe	Jane	jdoe@usna.edu	7	4
312	Doe	Bob	bred@usna.edu	6	6
152	Johnson	Matt	mat@usna.edu	7	4

Rule:

All mids with same Company have the same Wing
(Company→ Wing)

Problems due to data redundancy?

Modification Anomalies

- Deletion Anomaly: What if we delete all mids in Company 5?
- Insertion Anomaly: What if we want to record the fact the Wing for Company 12 is 6?
- Update Anomaly: What if we change the Wing for Company 7 to be 3?

Update Anomalies

- The MID table before and after an incorrect update operation on Wing for Company = 7

Number	LastName	FirstName	Email	Company	Wing
190	Smith	John	jsmith@usna.edu	12	2
673	Doe	Jane	jdoe@usna.edu	7	4
312	Doe	Bob	bred@usna.edu	6	6
152	Johnson	Matt	mat@usna.edu	7	4

Number	LastName	FirstName	Email	Company	Wing
190	Smith	John	jsmith@usna.edu	12	2
673	Doe	Jane	jdoe@usna.edu	7	5
312	Doe	Bob	bred@usna.edu	6	6
152	Johnson	Matt	mat@usna.edu	7	4

Table decomposition

Number	LastName	FirstName	Email	Company	Wing
190	Smith	John	jsmith@usna.edu	12	2
673	Doe	Jane	jdoe@usna.edu	7	4
312	Doe	Bob	bred@usna.edu	6	6
152	Johnson	Matt	mat@usna.edu	7	4

Number	LastName	FirstName	Email	Company
190	Smith	John	jsmith@usna.edu	12
673	Doe	Jane	jdoe@usna.edu	7
312	Doe	Bob	bred@usna.edu	6
152	Johnson	Matt	mat@usna.edu	7

Company	Wing
6	6
7	4
12	2

Disadvantage?

Decisions

- Do we have to decompose / merge?
- How do we identify problems caused by redundancy?
  - Functional dependencies

Functional Dependency (FD)

- X → Y (X *determines* Y)
  - If same value for X then same value for Y
- Examples:
  - Any primary key
    - Alpha → (Name, Class, DateOfBirth)
  - EmployeeRating → Wage
  - (NbHours, HourlyPrice)→Charge

Functional Dependency (FD) Rules

- If A → (B, C), then A → B and A→ C
- If (A,B) → C, then
  - neither A nor B determines C by itself
  - A and B determine C

FD Facts

- A functional dependency is a statement about *all allowable* instances of a table
- You cannot find the functional dependencies simply by looking at some data:
  - Data set limitations
  - Must be logically a determinant
- Given some data in a table R, we can check if it *violates* some FD, but we cannot tell if the FD *holds* over R!

Functional Dependencies in the MIDSHIPMAN Table

Assuming data is representative, determine the FD

Alpha	LastName	FirstName	Major	Advisor
111342	Thomas	Sarah	IEA	Lewis
112368	Smith	John	IFP	Jones
116644	Mikalson	Michael	IFA	Skapanski
117862	Doe	Jane	IFA	Skapanski
123116	Doe	Bob	IFP	Lefferton
120908	Johnson	John	IFP	Jones
121198	Thomas	Thomas	IEA	Lewis
129722	Jefferson	Janet	IFP	Lefferton
129832	Thomas	Sarah	IFP	Lefferton

Functional Dependencies in the MIDSHIPMAN Table

Alpha → (LastName, FirstName, Major, Advisor)

Advisor → Major

What Makes Determinant Values Unique?

- A determinant is unique in a relation if, and only if, it determines every other column in the relation
- Unique determinants = superkey

Key

- A set of columns is a key for a relation if :
  1. a) No two distinct rows can have same values in all key columns

     or equivalently

     b) determines all of the other columns in a relation

  2. This is not true for any subset of the key
- Candidate key = key
- Primary key, Alternate key

Normal Forms

- Relations are categorized as a **normal form** based on which modification anomalies or other problems that they are subject to:

Source of Anomaly	Normal Forms	Design Principles
Functional dependencies	1NF, 2NF, 3NF, BCNF	BCNF: Design tables so that every determinant is a candidate key
Multivalued dependencies	4NF	4NF: Move each multivalued dependency to a table of its own
Data constraints and oddities	5NF, DK/NF	DK/NF: Make every constraint a logical consequence of candidate keys and domains

Normal Forms

Number	Last Name	First Name	Email	Company	Wing
190	Smith	John	jsmith@ucmp.edu	12	2
673	Doe	Jane	jdoe@ucmp.edu	7	4
312	Doe	Bob	jbob@ucmp.edu	6	6
152	Johnson	Matt	mjoh@ucmp.edu	7	4

- **1NF** – A table that qualifies as a relation is in 1NF
- **Boyce-Codd Normal Form (BCNF)** – A relation is in BCNF if every determinant is a (candidate) key

"I swear to construct my tables so that all nonkey columns are dependent on the key, the whole key and nothing but the key, so help me Codd."

Eliminating Modification Anomalies from Functional Dependencies in Relations

- Put all relations into Boyce-Codd Normal Form (BCNF):

1. Identify every functional dependency
2. Identify every candidate key
3. If there is a functional dependency that has a determinant that is not a candidate key:
   - A. Move the columns of that functional dependency to a new relation
   - B. Make the determinant of that functional dependency the primary key of the new relation
   - C. Leave a copy of the determinant as a foreign key in the original relation
   - D. Create a referential integrity constraint between the original relation and the new relation
4. Repeat step 3 until every determinant of every relation is a candidate key

(Note: In step 3, if there is more than one such functional dependency, start with the one with the most columns.)

Putting a Relation into BCNF: ASSIGNMENT_GRADES

Alpha	Assignment	Points	PointsTotal
129722	QUIZ1	10	10
129722	QUIZ2	2.5	10
129722	QUIZ3	2	20
122422	QUIZ1	6	10
122422	QUIZ2	7	10
122422	QUIZ3	18	20
129936	QUIZ1	6	10
129936	QUIZ2	8	10
129936	QUIZ3	20	20

Putting a Relation into BCNF: ASSIGNMENT_GRADES

ASSIGNMENT_GRADES (Alpha, Assignment, Points, PointsTotal)

(Alpha, Assignment) → (Points, PointsTotal)
Assignment → (PointsTotal)

ASSIGNMENT (Assignment, PointsTotal)
GRADES (Alpha, *Assignment*, Points)

Where GRADES.Assignment must exist in ASSIGNMENT.Assignment

Putting a Relation into BCNF: New Relations

Alpha	Assignment	Points
129722	QUIZ1	10
129722	QUIZ2	2.5
129722	QUIZ3	2
122422	QUIZ1	6
122422	QUIZ2	7
122422	QUIZ3	18
129936	QUIZ1	6
129936	QUIZ2	8
129936	QUIZ3	20

Assignment	PointsTotal
QUIZ1	10
QUIZ2	10
QUIZ3	20

Redundancy Example

- PartKit →→ Part, PartKit→ Price

PartKit	Part	Price
Bike Repair	Wrench	14.95
Bike Repair	Screwdriver	14.95
Bike Repair	Tube Fix	14.95
Vice	Vice Jaw	125.00
Vice	Handle	125.00
Vice	Extension Screw	125.00
First Aid	Bandaids	24.95
First Aid	Aspirin	24.95
First Aid	Elastic Band	24.95
First Aid	Ibprofin	24.95

Multivalued Dependencies

Employee →→ Degree

Employee	Degree
Jones	BS
Jones	AA
Greene	PhD
Greene	MS
Greene	BS
Chau	BS

PartKit →→ Part

PartKit	Part
Bike Repair	Wrench
Bike Repair	Screwdriver
Bike Repair	Tube Fix
Vice	Vice Jaw
Vice	Handle
Vice	Extension Screw
First Aid	Bandaids
First Aid	Aspirin
First Aid	Elastic Band
First Aid	Ibprofin

Eliminating Anomalies from Multivalued Dependencies

- Multivalued dependencies are not a problem if they are in a separate relation, so:
  - Always put multivalued dependencies into their own relation
  - This is known as **Fourth Normal Form (4NF)**

Normalize or Not?

Customer(<u>CustID</u>, Name, City, State, Zip)

- Assuming that city and state determine zip code, is Customers table in BCNF?
- If Customers table is not in BCNF, would you or would you not normalize it to BCNF? Give one reason for the choice you make

Class Exercise

- R(A, B, C, D, E, F)
  A→(B,C,D,E,F)
  B→C
  (D,E)→F
- Is A a key? Why?
- Is R in BCNF? Why?
- If R not in BCNF, decompose to BCNF

Class Exercise

ID	Name	University	MainCampus
1	John Smith	Cornell	Ithaca
2	John Smith	MIT	Boston
3	Matt Johnson	Ithaca College	Ithaca
4	Chris Brown	USNA	Annapolis
5	Jane Doe	Cornell	Ithaca
6	Ric Crabbe	USNA	Annapolis

•Example of deletion anomaly?

•Do these FDs hold? Why?
- •ID→University
- •Name→ID
- •University→MainCampus
- •MainCampus→Name

Summary

- Modification anomalies
- Functional dependency
  - X → Y (X *determines* Y)
  - Unique determinant ⇔(candidate) key
- 1NF – A table that qualifies as a relation is in 1NF
- Boyce-Codd Normal Form (BCNF) – A relation is in BCNF if every determinant is a (candidate) key
- 4NF – Multivalued dependencies are in a relation by themselves