

Normal Forms

CISC437/637, Lecture #9

Ben Carterette

Copyright © Ben Carterette

1

Normal Forms

- **Normal forms** help determine whether a relation is subject to logical inconsistencies and anomalies
- There is a series of normal forms of increasing “strictness” and lower chance of inconsistency
 - First normal form (1NF)
 - Third normal form (3NF)
 - Boyce-Codd normal form (BCNF)
 - Fourth, fifth, sixth normal forms (4NF, 5NF, 6NF)

Copyright © Ben Carterette

2

First Normal Form

- 1NF is the simplest normal form
- A relation is in 1NF if all of its fields contain only atomic values
 - No lists or sets
- Our definition of a relation implies 1NF holds by default:
 - By enforcing domain constraints, we guarantee every relation is in 1NF

Copyright © Ben Carter@Co

3

Redundancy Revisited

- We touched on redundancy in the context of database design decisions
- Redundancy is an extremely important consideration in those decisions
 - Redundant storage can be a problem
 - The real problem is maintaining the data during inserts/updates/deletes
 - Anything that modifies redundant data has to be sure to modify every “copy” of that data, or inconsistencies can be introduced
- Redundancy can be quite subtle, hard to detect
 - It doesn't strictly mean duplicate information

Copyright © Ben Carter@Co

4

Does This Relation Instance Contain Redundancy?

<i>ID</i>	<i>name</i>	<i>salary</i>	<i>dept_name</i>	<i>building</i>	<i>budget</i>
22222	Einstein	95000	Physics	Watson	70000
12121	Wu	90000	Finance	Painter	120000
32343	El Said	60000	History	Painter	50000
45565	Katz	75000	Comp. Sci.	Taylor	100000
98345	Kim	80000	Elec. Eng.	Taylor	85000
76766	Crick	72000	Biology	Watson	90000
10101	Srinivasan	65000	Comp. Sci.	Taylor	100000
58583	Califieri	62000	History	Painter	50000
83821	Brandt	92000	Comp. Sci.	Taylor	100000
15151	Mozart	40000	Music	Packard	80000
33456	Gold	87000	Physics	Watson	70000
76543	Singh	80000	Finance	Painter	120000

Copyright © Ben Carter@De

5

Functional Dependencies

- A **functional dependency** is a type of integrity constraint related to relation keys
 - Field(s) *X* *functionally determine* field(s) *Y* if
 - for every possible instance *r* of relation *R*,
 - for any two tuples *t1*, *t2* in *r*,
 - * if $\pi_X(t1) = \pi_X(t2)$
 - * then $\pi_Y(t1) = \pi_Y(t2)$
 - We write $X \rightarrow Y$ to indicate functional dependency
- Functional dependencies are statements about all possible relation instances
- If *K* is a candidate key, $K \rightarrow R$

Copyright © Ben Carter@De

6