1.204 Lecture 2

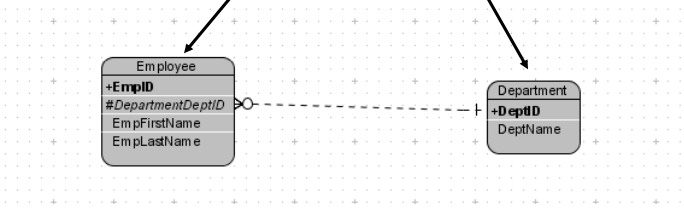
Data models, concluded
Normalization

Keys

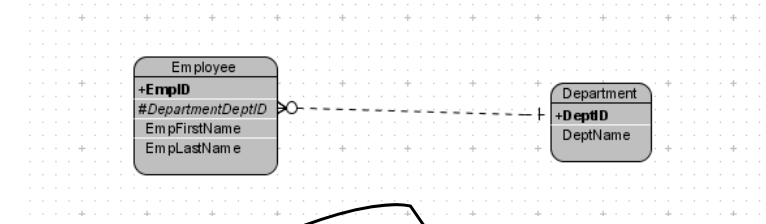
- **Primary key: one or more attributes that uniquely identify a record**
  - **Name or identifying number, often system generated**
  - **Composite keys are made up of two fields**
    - **E.g., aircraft manufacturer and model number**

Foreign keys

- **Primary key of the independent or parent entity type is maintained as a non-key attribute in the dependent or child entity type**



Foreign keys

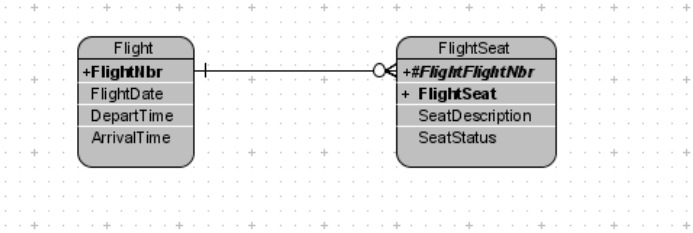


EmpID	DeptID	EmpFirstName	EmpLastName
4436	483	Brown	John
4574	483	Jones	Helen
5678	372	Smith	Jane
5674	372	Crane	Sally
9987	923	Black	Joe
5123	923	Green	Bill
5325	483	Clinton	Bob

DeptID	DeptName
930	Receiving
378	Assembly
372	Finance
923	Planning
483	Construction

Database requires a valid department number when employee is added Employee ID is the unique identifier of employees; department number is not needed as part of the employee primary key

Composite foreign keys

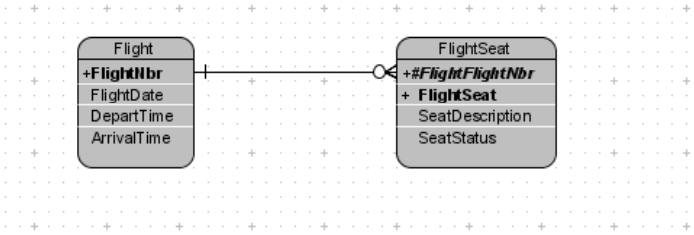


Independent/parent

Dependent/child (must contain, as a foreign key, the primary key of the independent entity)

Assume a charter airline: every flight has a different number
What has to change if this is a scheduled carrier?

Composite foreign keys



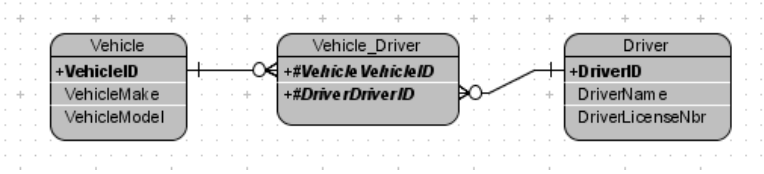
Flight			
FlightNbr	**FlightDate**	**DepartTime**	**ArrivalTime**
243	9/24/00	9:00am	11:00am
253	9/24/00	10:00am	12:30pm
52	9/24/00	11:00am	2:00pm

FlightSeat			
FlightNbr	**SeatNbr**	**SeatStatus**	**SeatDescription**
243	8A	Confirmed	Window
243	7D	Reserved	Aisle
243	14E	Open	Center
253	1F	Open	Window
253	43A	Confirmed	Window

Flight number must be part of the flight seat primary key; this is different than employee and department, where department is not required.

Foreign keys (many-many relationships)

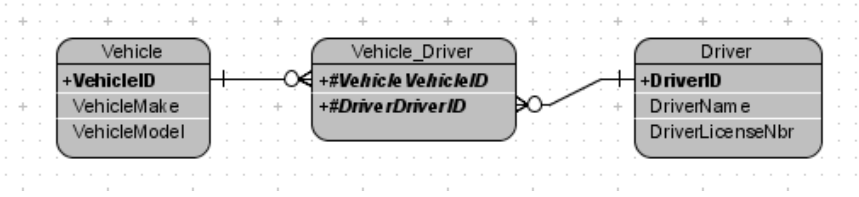
- **Primary key of parent is used in primary key of child**



Independent **Dependent** **Independent**

Vehicle can be driven by many drivers; driver can drive many vehicles

Many-to-many relationships with foreign keys



Vehicle		
VehicleID	VehicleMake	VehicleModel
35	Volvo	Wagon
33	Ford	Sedan
89	GMC	Truck

Vehicle Driver	
VehicleID	DriverID
35	900
35	253
89	900

Driver		
DriverID	DriverName	DriverLicenseNbr
253	Ken	A23423
900	Jen	B89987

Never create an entity with vehicle1, vehicle2,… !

Five normal forms: preventing errors

- **1: All occurrences of an entity must contain the same number of attributes.**
  - **No lists, no repeated attributes.**
- **2: All non- primary key fields must be a function of the primary key.**
- **3: All non- primary key fields must not be a function of other non- primary key fields.**
- **4: A row must not contain two or more independent multi-valued facts about an entity.**
- **5: A record cannot be reconstructed from several smaller record types. (Informal)**

Examples based on William Kent, "A Simple Guide to Five Normal Forms in Relational Database Theory", Communications of the ACM 26(2), Feb. 1983

First normal form

- **All rows must be fixed length**
  - **Does not allow variable length lists.**
  - **Does not allow repeated fields, e.g., vehicle1, vehicle2, vehicle3…**
    - **However many columns you allow, you will always need one more…**
    - **Use a many-many relationship instead, always. See vehicle-driver example**

Second normal form

Part	Warehouse	Quantity	WarehouseAddress
42	Boston	2000	24 Main St
333	Boston	1000	24 Main St
390	New York	3000	99 Broad St

- **All non-key fields must be a function of the full key**
  - **Example that violates second normal form:**
    - **Key is Part + Warehouse**
    - **Someone found it convenient to add Address, to make a report easier**
    - **WarehouseAddress is a fact about Warehouse, not about Part**
  - **Problems:**
    - **Warehouse address is repeated in every row that refers to a part stored in a warehouse**
    - **If warehouse address changes, every row referring to a part stored in that warehouse must be updated**
    - **Data might become inconsistent, with different records showing different addresses for the same warehouse**
    - **If at some time there were no parts stored in the warehouse, there may be no record in which to keep the warehouse's address.**

Second normal form

- **Solution**
  - **Two entity types: Inventory, and Warehouse**
  - **Advantage: solves problems from last slide**
  - **Disadvantage: If application needs address of each warehouse stocking a part, it must access two tables instead of one. This used to be a problem but rarely is now.**

Part	Warehouse	Quantity
42	Boston	2000
333	Boston	1000
390	New York	3000

Warehouse	WarehouseAddress
Boston	24 Main St
New York	99 Broad St

Third normal form

Employee	Department	DepartmentLocation
234	Finance	Boston
223	Finance	Boston
399	Operations	Washington

- **Non-key fields cannot be a function of other non-key fields**
  - **Example that violates third normal form**
    - **Key is employee**
    - **Someone found it convenient to add department location for a report**
    - **Department location is a function of department, which is not a key**
  - **Problems:**
    - **Department location is repeated in every employee record**
    - **If department location changes, every record with it must be changed**
    - **Data might become inconsistent**
    - **If a department has no employees, there may be nowhere to store its location**

Third normal form

- **Solution**
  - **Two entity types: Employee and department**

Employee	Department
234	Finance
223	Finance
399	Operations

Department	DepartmentLocation
Finance	Boston
Operations	Washington

TV: "The truth, the whole truth, and nothing but the truth"
DB: "The key, the whole key, and nothing but the key"

Fourth normal form

Employee	Skill	Language
Brown	cook	English
Smith	type	German

- **A row should not contain two or more independent multi-valued facts about an entity.**
  - **Example that violates fourth normal form:**
    - **An employee may have several skills and languages**
  - **Problems**
    - **Uncertainty in how to maintain the rows. Several approaches are possible and different consultants, analysts or programmers may (will) take different approaches, as shown on next slide**

Fourth normal form problems

Employee	Skill	Language
Brown	cook	
Brown	type	
Brown		French
Brown		German
Brown		Greek

- **Blank fields ambiguous. Blank skill could mean:**
  - **Person has no skill**
  - **Attribute doesn't apply to this employee**
  - **Data is unknown**
  - **Data may be found in another record (as in this case)**
- **Programmers will use all these assumptions over time, as will data entry staff, analysts, consultants and users**
  - **Disjoint format is used on this slide. Effectively same as 2 entity types.**

Fourth normal form problems, cont.

Employee	Skill	Language
Brown	cook	French
Brown	cook	German
Brown	cook	Greek
Brown	type	French
Brown	type	German
Brown	type	Greek

- **Cross product format. Problems:**
  - **Repetitions: updates must be done to multiple records and there can be inconsistencies**
  - **Insertion of a new skill may involve looking for a record with a blank skill, inserting a new record with possibly a blank language or skill, or inserting a new record pairing the skill with some or all of the languages.**
  - **Deletion is worse: It means blanking a skill in one or more records, and then checking you don't have 2 records with the same language and no skill, or it may mean deleting one or more records, making sure you don't delete the last mention of a language that should not be deleted**

Fourth normal form solution

- **Solution: Two entity types**
  - **Employee-skill and employee-language**

Employee	Skill
Brown	cook
Brown	type

Employee	Language
Smith	French
Smith	German
Smith	Greek

- **Note that skills and languages may be related, in which case the starting example was ok:**
  - **If Smith can only cook French food, and can type in French and Greek, then skill and language are not multiple independent facts about the employee, and we have not violated fourth normal form.**
- **Examples you're likely to see:**
  - **Person on 2 projects, in 2 departments**
  - **Part from 2 vendors, used in 4 assemblies**

Fifth normal form

- **A record cannot be reconstructed from several smaller record types.**
- **Example:**
  - **Agents represent companies**
  - **Companies make products**
  - **Agents sell products**
- **Most general case (allows any combination):**

Agent	Company	Product
Smith	Ford	car
Smith	GM	truck

  - **Smith does not sell Ford trucks nor GM cars**
  - **If these are the business rules, a single entity is fine**
  - **But…**

Fifth normal form

- **In most real cases a problem occurs**
  - **If an agent sells a certain product and she represents the company, then she sells that product for that company.**

Agent	Company	Product
Smith	Ford	car
Smith	Ford	truck
Smith	GM	car
Smith	GM	truck
Jones	Ford	car

(Repetition of facts)

  - **We can reconstruct all true facts from 3 tables instead of the single table:**

Agent	Company
Smith	Ford
Smith	GM
Jones	Ford

Agent	Product
Smith	car
Smith	truck
Jones	car

Company	Product
Ford	car
Ford	truck
GM	car
GM	truck

(No repetition of facts)

Fifth normal form

- **Problems with the single table form**
  - **Facts are recorded multiple times. E.g., the fact that Smith sells cars is recorded twice. If Smith stops selling cars, there are 2 rows to update and one will be missed.**
  - **Size of this table increases multiplicatively, while the normalized tables increase additively. With big operations, this is a big difference.**
    - **100,000 x 100,000 is a lot bigger than 100,000 + 100,000**
- **It's much easier to write the business rules from 5th normal**
  - **Rules are more explicit**
  - **Supply chains usually have all sorts of 5th normal issues**

Fifth normal form, concluded

- **An example with a subtle set of conditions**

Non-normal

Agent	Company	Product
Smith	Ford	car
Smith	Ford	truck
Smith	GM	car
Smith	GM	truck
Jones	Ford	car
Jones	Ford	truck
Brown	Ford	car
Brown	GM	car
Brown	Toyota	car
Brown	Toyota	bus

Can you quickly deduce the business rules from this table?

Fifth normal

Agent	Company
Smith	Ford
Smith	GM
Jones	Ford
Brown	Ford
Brown	GM
Brown	Toyota

Company	Product
Ford	car
Ford	truck
GM	car
GM	truck
Toyota	car
Toyota	bus

Agent	Product
Smith	car
Smith	truck
Jones	car
Jones	truck
Brown	car
Brown	bus

- **Jones sells cars and GM makes cars, but Jones does not represent GM**
- **Brown represents Ford and Ford makes trucks, but Brown does not sell trucks**
- **Brown represents Ford and Brown sells buses, but Ford does not make buses**

Summary

- **Real systems have subtle system rules**
  - **Care in data modeling and system rules is needed to achieve good data quality**
    - **This is an interactive, conversational process, done with lots of people**
  - **Care in data normalization is needed to preserve data quality**
    - **Normalization ensures that each fact is stored in one and only one place (with rare exceptions). If a fact is stored in two or more places, they can and will become inconsistent, and then you won't know the fact at all.**
    - **This is a technical process, done in the back room with some technical people**

1.204 Computer Algorithms in Systems Engineering

Spring 2010