The ER Model

Overview of the ER Model

- The Entity-Relationship model is a high level way to look at data
- The ER model is made up of three parts
  - Entities
  - Relationships
  - Attributes
- These things tend to translate well to relational databases

Attributes

- Attributes are the smallest division of data
- They should not be able to be split
- Good Examples
  - First name
  - Social Security Number
  - Quantity
- Bad Examples
  - Address
  - Car
  - Toppings on a pizza

Attributes continued

- If you can break up an attribute you should
- Name:
  - Jane Smith BAD
- First name:
  - Jane
- Last name: GOOD
  - Smith

Attributes continued 2

Good

- Street address
  - 123 Lani Kai St
- City
  - Kahului
- St
  - HI
- Zip
  - 96790

Bad

- Address
  - 123 Lani Kai St, Kahului, HI 96790

Entities

- Entities are the things in your data
- Examples:
  - People
  - Space ships
  - Locations
  - Inventory
- Composed of attributes
- Stored together in tables

Entities Continued

- A table of characters from Battlestar Galactica

bsg_people				
Id	Fname	Lname	Homeworld	Age

- Id - Unique identifier (more on this later)
- Fname – First name
- Lname – Last name
- Homeworld – Planet the character is from
- Age – Characters age

Entities continued 2

bsg_people				
Id	Fname	Lname	Homeworld	Age
0	William	Adama	Caprica	61
1	Lee	Adama	Caprica	30
2	Laura	Roslin	Caprica	NULL
3	Kara	Thrace	Caprica	NULL

- Here is an example instance of the bsg_people entity table

Entities continued 3

- Some more example table schemas

addresses				
Id	Street	City	State	Zip

students				
Id	Fname	Lname	Dept	GPA

parts			
Id	Pname	Dept	Qty

Relationships

- Relationships in the ER model are relations between entities
- They connect entities to other entities
- Can have restrictions on number of entities allowed
- Can be built into entity tables or can be its own table
  - Depends on number of entities involved

Relationship examples

- Employees **can work in many** departments
- Managers **can manage one** department
- Students **can be in many** classes
- A class **must be in at least one** department
- A transaction **must be associated to one** customer

Relationships examples 2

- We had characters from a homeworld

bsg_people				
Id	Fname	Lname	Homeworld	Age

- Previously 'Homeworld' was an attribute
- What if 'Homeworld' was an entity?

bsg_planets				
Id	Name	Population	Language	Capitol

- We might now say "A character **must be from one** planet."

A One to Many Relationship

- In this case we have a few conditions
  - A character **must** be from a planet
  - A character is only from **one** planet
  - A planet can be the homeworld for **many** characters
- Depending on how you phrase it this is a many-to-one or a one-to-many relationship
- We can just add a column for this (or change our current column)

Example table

bsg_people

Id	Fname	Lname	Homeworld	Age
0	William	Adama	2	61
1	Lee	Adama	2	30
2	Laura	Roslin	2	NULL
3	Kara	Thrace	2	NULL

bsg_planets

Id	Name	Population	Language	Capitol
1	Gemenon	2800000000	Old Gemenese	Oranu
2	Caprica	4900000000	Caprican	Caprica City

- Homeworld now references bsg_planets(id)

Many-to-Many

- Suppose we had a list of certifications

bsg_cert

Id	title
0	Raptor
1	Viper
2	Mechanic
3	Command

- A character can be certified in many things

Many-to-Many Continued

- So what if…

bsg_cert				
Id	Title	Character1	Character2	…

Many-to-Many Continued

- So what if…

bsg_cert				
Id	Title	Character1	Character2	…

- No, bad, wrong, don't do this, ever, no!
- Instead:

bsg_cert_people	
cid	pid

Many-to-Many Example

- Lee is certified in Command and Viper
- Laura is certified in Command

bsg_cert_people	
cid	pid
3	1
1	1
3	2

bsg_cert	
Id	title
0	Raptor
1	Viper
2	Mechanic
3	Command

bsg_people				
Id	Fname	Lname	Homeworld	Age
0	William	Adama	2	61
1	Lee	Adama	2	30
2	Laura	Roslin	2	NULL
3	Kara	Thrace	2	NULL

In Review

- Attributes describe Entities and Relationships
- Entities are things
- Relationships say how things are related
  - Can be required or optional
  - Can allow only one entity or can allow many
  - Sometimes requires its own table, sometimes not
  - Can have its own attributes