

Post-relational: Object-based Modeling

Physical Data Modeling

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Kevin C.C. Chang, Professor
Computer Science @ Illinois

Learning Objectives

By the end of this video, you will be able to:

- Describe the object-based models after the relational model.
- Identify the motivations behind these models.
- Give examples of such models.

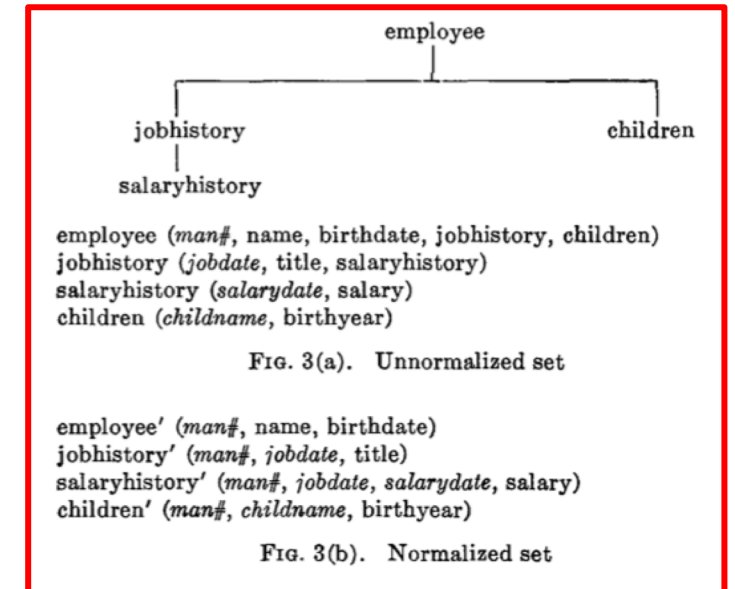
Post-Relational Models: Two Driving Forces

- **Meeting programming paradigms**

- Driven by the "impedance mismatch" with object-oriented programming.
- (1980s) Object-Oriented.
- (1980s) Object-Relational Model.

- **Dealing with data in various new settings**

- Driven by applications beyond enterprise data management.
- (1990s) Document Model.
- (1990s) Key-Value Model.
- (2000s) Graph Model.



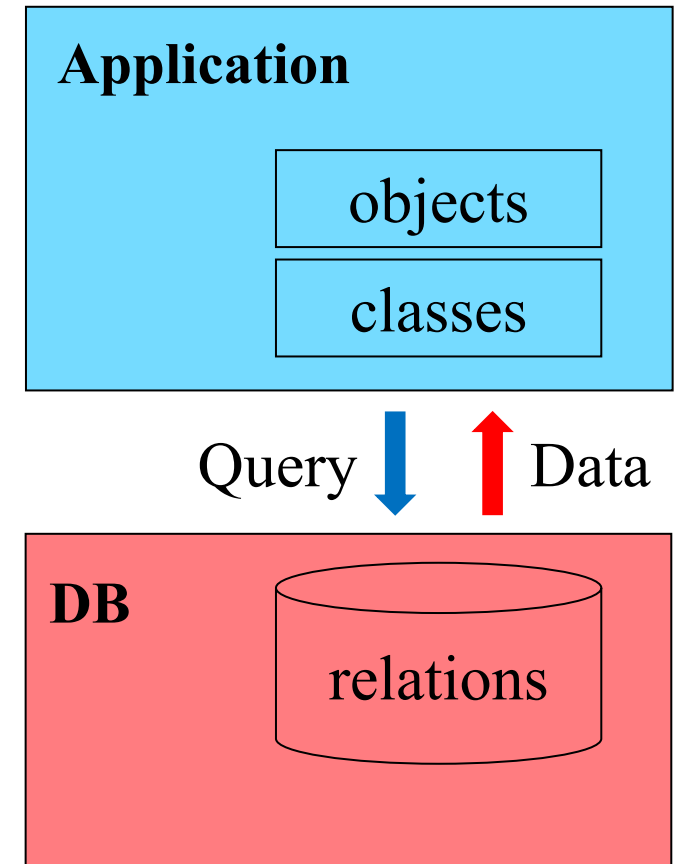
Example enterprise data management scenario from Codd's 1970 relational model paper (Codd 1972)

Impedance Mismatch: Database vs. Application Programming

Object-relational impedance mismatch --

Object (of class) vs **Tuple** (of relation):

- **Object identifiers:** Object referenced by identifiers (pointers) while tuples by values.
- **Nesting:** Object can be nested to contain objects while tuple contains only simple values as attributes.
- **Methods:** Object has methods while tuple contains only attributes.
- **Inheritance:** Classes can form subclass hierarchy while relations cannot.
- **Encapsulation:** Object can hide internal representation.



Database vs. application programming

Meeting Programming Paradigms → Object-based

- Object-Oriented Data Model

- Extending OO programming languages with database functions.
- Current systems:..



Example OO database systems

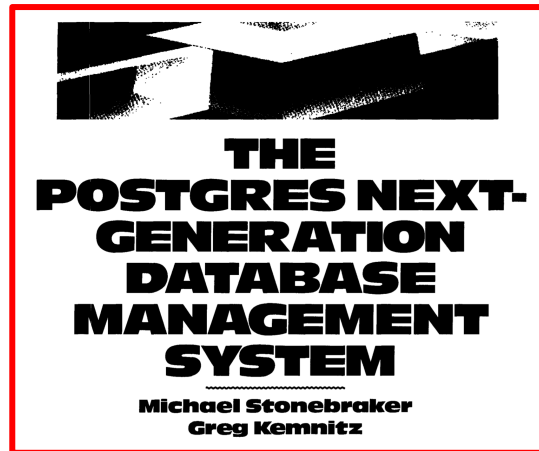
- Object-Relational Data Model

- Extending RDBMS with object features.
- Current systems: PostgreSQL
 - Most RDBMS supports object features to various extents.



Example object-relational database system

And, this is ANOTHER Turing Award



1991



1996



From Postgres to PostgreSQL

The image is a screenshot of the A.M. Turing Award website. At the top, there is a search bar and a grid of award winners' portraits. Below this, a navigation bar allows filtering by "ALPHABETICAL LISTING", "YEAR OF THE AWARD", or "RESEARCH SUBJECT". The main content area displays the profile of Michael Stonebraker, including his photo, a "DL" (Distinguished Lecturer) badge, and his citation for the year 2014. The citation reads: "For fundamental contributions to the concepts and practices underlying modern database systems." Below the citation, there are links to "SHORT ANNOTATED BIBLIOGRAPHY", "ACM TURING AWARD LECTURE VIDEO", "RESEARCH SUBJECTS", "ADDITIONAL MATERIALS", and "VIDEO INTERVIEW". A detailed paragraph follows, describing his contributions to database management technology and his work on the INGRES system.

A.M. Turing Award 2014 citation. Retrieved from
http://amturing.acm.org/award_winners/stonebraker_1172121.cfm

“Object-Relational” Example: Table Inheritance

Table inheritance in PostgreSQL

Beers-Ales scenario, in Object-Relational way:

- Superclass

```
CREATE TABLE Beers (  
    name text,  
    brewer text,  
    alcohol float)
```

- Subclass

```
CREATE TABLE Ales (  
    color char(10) ) INHERITS (Beers);
```

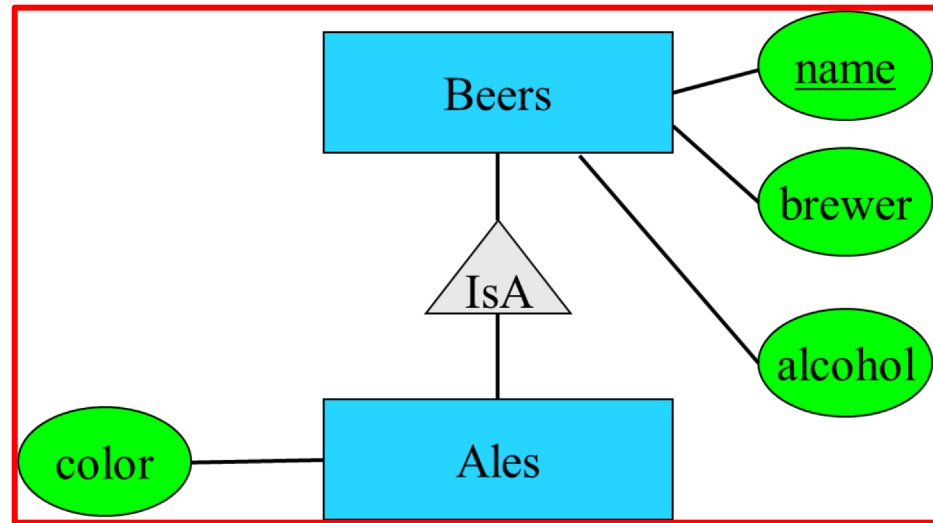
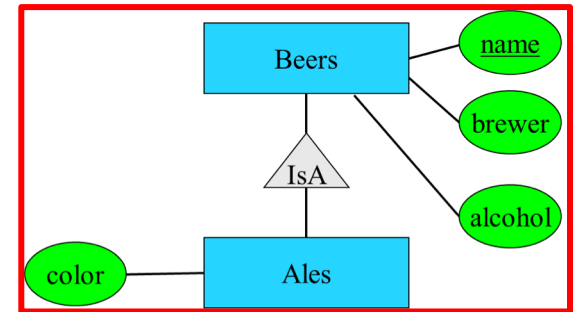


Table Inheritance: Interesting Complications

- When querying about “Beers”, what tables should match?
 - Default is to match not only Beers but also Ales.
- When inserting into “Beers”, what tables should match?
 - Seems natural to only insert into Beers.
- How do I limit matching to only tuples in Beers?
 - You say “ONLY Beers”. E.g.
SELECT name, alcohol FROM **ONLY Beers**
- How to expand matching to not only Beers but also Ales?
 - You say “Beers*”.
- See [Inheritance in PostgreSQL Manual](#).



On StackOverflow, some suggested against using inheritance in PostgreSQL (and recommended the Null-Value approach). Can you imagine why?

To someone who has experience using inheritance in PostgreSQL: Is it worth using it, or better not to? In which situation you would use it?

To be honest, I do not fully understand the difference between the relational and OO models...

postgresql inheritance

share improve this question

edited Jun 22 '11 at 2:59



Dr. Person Person II

1,778 ● 4 ● 22 ● 29

asked May 13 '10 at 20:45



Anton Prokofiev

413 ● 4 ● 16

Question on table inheritance in PostgreSQL, 2011.
Retrieved from <https://stackoverflow.com>.