

Re: Relational-to-OOP Tax

Source: <http://coding.derkeiler.com/Archive/General/comp.object/2007-02/msg00551.html>

- *From:* "frebe" <frebe73@xxxxxxxxxx>
 - *Date:* 26 Feb 2007 02:56:07 -0800
-

Just for example, nearly nobody doing text mining work uses relational databases

Many SQL vendors has text mining features. Oracle Text for example.

(or SQL databases --- which aren't the same thing).

Indeed true.

Nearly everybody who works in this area attempts to use a SQL database to start with, and switches to something else because they have little real choice.

Most SQL database are designed for typical "enterprise scenarios". Other areas like GIS and text mining are supported by many vendors, but not all. The fact that one might chose a non-SQL database for such areas is not because the relational model is flawed, but because the available DBMS implementations suffers.

SQL supports a fairly general version of basic, structured data, but the minute you get into less-structured data, it tends to fall apart.

Can you prove this? Or do you have some examples?

As for the assertion that SQL isn't really relational, I'll defer to C. J. Date: "As I've written elsewhere many times, the biggest problem with SQL is precisely that it doesn't support the relational model."

Indeed correct. But still SQL is the best available language. Date's

Re: Relational-to-OOP Tax

Tutorial D isn't available in production databases.

Getting back to the original claim that relationalism is all that's needed,

The original claim is that relations are the only data structure needed. Classes is useful for other things like data types.

Date and Darwen addressed this quite thoroughly in: Foundation for Object/Relational Databases: The Third Manifesto. Date summarizes the relationship quite succinctly: "As Hugh Darwen and I have shown in The Third Manifesto,2 object functionality and the relational model are completely orthogonal to one another." He goes on to quote from the third manifesto: "The relational model needs no extension, no correction, no subsumption, and above all no perversion, in order [to support object functionality]." All that's needed is to support relational domains properly (which SQL never did)..."

If you would have continued to read you would have found that they also states that classes should never be mapped to tables, only to domains (datatypes). (See <http://www.dcs.warwick.ac.uk/~hugh/TTM/TTM-TheAskewWall-printable.pdf>, p 52-61, if you need a shorter version).

"We can't accept the equation object class = relation" and "object class = domain"

Date states that relations and objects are orthogonal, that means that they should not be used for the same thing. If you already have the data in a relational way, there are no need to transform it into a network graph.

The relational model needs classes, but not as data structures. But for defining dates, social security number, etc, they are very useful.

My recommendation for both of them would be to read Database in Depth: Relational Theory for Practitioners (by C.J. Date, 2005, published by O'Reilly). Of course, they should also read the Third Manifesto, but based on their writing, they need to do some other reading first.

Although I have only read parts of the book, I would recommend you to read it too. When you would understand why O/R mapping is such stupid thing to do.

/Fredrik

.