

Re: Biz Tree Challenge

Source: <http://coding.derkeiler.com/Archive/General/comp.object/2007-01/msg00433.html>

- *From:* "topmind" <topmind@xxxxxxxxxxxxxxxxxx>
 - *Date:* 25 Jan 2007 21:36:51 -0800
-

On Jan 25, 5:05 pm, Robert Martin <uncle...@xxxxxxxxxxxxxxxxxx> wrote:

On 2007-01-24 16:03:30 -0600, "topmind" <topm...@xxxxxxxxxxxxxxxxxx> said:

Buchman's trash lives on in OO. New name, same crap.

...OO is not a network database. OO is not a database.

Okay, "network structures".

OO
is a programming technique that involves tables of pointers to
functions. You like tables, don't you?

Actually, those decisions tables that you like so much are very similar
to the vtables used in OO languages to select behaviors.

If such were query-able and dynamic and better normalized and didn't
require repetitive set-gets and came built-in with relational operators
(or at least a standard set of collection handling behaviors), perhaps
I would warm up to them a bit. OO code is like a static database of
sorts.

Also, relational requires certain rules. OO does not. This means that
every OO developer will produce something quite different. The
relational rules make it easier to follow and dissect structures. It is
similar to Goto's versus nested blocks. Goto's allow more freedom, but
also create vast inconsistencies of style. It is building-codes versus
shanty towns.

At least you are moving in the right direction. It is just that first

Re: Biz Tree Challenge

you'll have to climb a crate full of Everests.

Technically, OO is generally maps of pointers to maps. Maps are the fundamental structure of OO, not tables.

--

Robert C. Martin (Uncle Bob) | email: uncle...@xxxxxxxxxxxxxxxxxxxx

-T-

.