

Re: OCP and Java

Source: <http://coding.derkeiler.com/Archive/General/comp.object/2005-02/0161.html>

stevenwurster_at_lycos.com

Date: 02/08/05

Date: 8 Feb 2005 06:35:04 -0800

Thomas Gagne wrote:

- >
- > *Until Dionne Warwick and her psychic friends network start programming, I have*
- > *a hard time imagining any programmer with the clairvoyance required to make*
- > *anything /final/. It is mistaken to believe anything we write is so perfect*
- > *that it can't be improved upon or that future situations won't require it to*
- > *do something different -- however subtle. Better to comment the code and*
- > *recommend that anyone tinkering with it be extremely cautious.*

The only place that I can think of where frozen is used in the Eiffel libraries is in the class ANY (analagous to class Object in Java). There are a few routines used in a general set of contracts whose behavior is expected to be consistent throughout an entire system.

An example is a routine that compares to objects for equality. I believe it's called 'standard_is_equal'. It calls the non-frozen routine 'is_equal', which any descendant can redefine. The idea there is that a general set of contracts use 'standard_is_equal' and related routines, which rely on the behaviors of 'is_equal' and related. It's an interesting solution. Honestly, however, I cannot think of any time I would want to freeze a routine.

- > > *I believe the second issue (redefining an attribute into a routine) is*
 - > > *being looked into by the Eiffel committee so that it would be possible*
 - > > *in the future.*
 - >
 - > *Variables and functions don't have different namespaces in Eiffel?*
- If this
- > *kind of flexibiliy is valuable there are already programming languages without*
 - > *those namespace issues.*

It's not a namespace issue, if you want to call it that. The idea is that an attribute cannot be redefined into a routine because somewhere in the ancestor hierarchy someone may have assigned a value to that attribute. If it can become a routine, the assignment does not hold polymorphically. The ability to assign to attributes via routines is being researched in the Eiffel community, and that may help to solve this problem.

- > *OO languages are interesting in that they included a genetic taxonomy of their*
- > *own from which animals are created. Given the need to manipulate a collection*
- > *of stuff programmers can choose either a specialized trait (a string) or a*
- > *more generalized one (ordered collection). If they don't need ordering an*
- > *even less specialized "bag" trait may be sufficient.*

That's a library issue, not really a language one. Your point is correct, none the less.

Steve