

# Re: Design problem

---

*Source:* <http://coding.derkeiler.com/Archive/General/comp.object/2006-02/msg00086.html>

---

- *From:* Robert C. Martin <[unclebob@xxxxxxxxxxxxxxxxxxx](mailto:unclebob@xxxxxxxxxxxxxxxxxxx)>
  - *Date:* Sun, 05 Feb 2006 23:41:37 -0600
- 

On 1 Feb 2006 02:29:31 -0800, "Mark Nicholls"  
<[Nicholls.Mark@xxxxxxxxxx](mailto:Nicholls.Mark@xxxxxxxxxx)> wrote:

Robert C. Martin wrote:

On 6 Jan 2006 13:38:11 -0800, [roberts.noah@xxxxxxxxxx](mailto:roberts.noah@xxxxxxxxxx) wrote:

I'm having trouble with what seems like it should be a relatively simple problem and am wondering what ideas people here might have.

I have a shape class and inside this shape are "elements" (vector drawn based system). Certain things can be done to shapes, namely you can rotate, skew, flip, zoom..ummm...that's about it but you never know what could be added. Elements at this point can comprise of one of the following three constructs: A line; a polygon/polyline; or a circle. Lines are two point elements that contain a line from one to the other. Polygons are point coordinate arrays with a start and an end; empty polygons are actually lines from point to point otherwise it is a filled element. A circle is a center and a radius.

Now the problem. Sometimes these operations are done one after the other and I can increase speed (and yes we want it to be fast) if I can do them all at once. I want a raw "perform" function on either the

## Re: Design problem

element or the operation so that I can pass in individual operations or constructs of operations. The easy way to do this is to build matrices and pass them in, but how to apply these to a circle?? I can of course have enums representing types of operations and have some ifs in the circle construct but this fails the LSP (and I happen to agree with that principle). I do have a couple of ideas but there are failings in all of them (including rethinking how a circle is represented).

What ideas do you guys have?

Think of a circle as being defined by the radius line. It's not a point and a length, it's two points.

Thats fine and dandy, but the space of circles is not closed under the operations described...

i.e. compressing the Y axis makes a circle into a ellipse, not a little circle.

Compressing the Y axis wasn't on the list above. I presume, then, that you can compress or expand either axis. Can you define the behavior of the system when you expand or compress an axis? For example, does the aspect ratio of a square change if you expand or compress the Y axis? If not, then how does the square behave? What about a rectangle, does the aspect ratio change? Why would it change for a rectangle and not for a square? What about equilateral triangles, or any regular polyhedron? Do they become irregular? Or do they maintain their regularity? What should the behavior of a circle be if the Y axis is compressed?

---

Robert C. Martin (Uncle Bob) | email: [unclebob@xxxxxxxxxxxxxxxxxxxx](mailto:unclebob@xxxxxxxxxxxxxxxxxxxx)  
Object Mentor Inc. | blog: [www.butunclebob.com](http://www.butunclebob.com)  
The Agile Transition Experts | web: [www.objectmentor.com](http://www.objectmentor.com)  
800-338-6716

"The aim of science is not to open the door to infinite wisdom,  
but to set a limit to infinite error."  
— Bertolt Brecht, Life of Galileo

.

Re: Design problem