# Circle Language Spec Plan Automatic Containment

*Author: JJ van Zon*

*Location: Oosterhout, The Netherlands*

*Date: May 26, 2008 – June 21, 2008*

## Goal

This would update the article Automatic Containment. It could include ideas, found during the project *Diagram Expression for Classes & Relations*.

## Automatic Containment for Relations

The desire to have most relationships become bidirectional, seemed to lead to a bit of a problem. The notation for a bidirectional relationship could be a *line merge* and a *symbol merge*.

In a strict approach, this may make the diagrams look much different. To get a clearer view of it, an example diagram might be drawn out with line and symbol merges and alongside it, one *without* those line and symbol merges. Then it might be better visible how that works out in these diagrams.

One thing suspected, is that more things will end up next to each other, instead of inside each other. That may anull the effect of having a containment structure / like a map of your code.

So the 'problem' here is the friction between wanting bidirectional relationships, a simple, clean notation for it and on the other hand how nice containment structures might look in the diagram notation.

Some of the possible solutions:

* Drop the idea that all relationships should be bidirectional.
* Drop the idea that referential structure should be automatically converted to a containment structure / automatically determining composite aggregation derived from association aggregation. It may be a nice idea for the future, but it seems to conflicts with other nice ideas.
* Make a relationship have a primary direction, like the arrow in Parent 🡪 Children. You tend to display the Children as actual children in the containment notation, not as sibblings, because of the bidirectionalyt.
* Lists may tend to be displayed as children, which may automatically define what is a child, and what is a sibling.

## Steps

[x] Look at Computer Language Coding Principles.doc

[ ] ->.. Process as cross out lists:

[ ] ->.. Symbol Language.doc

[ ] ->.. Relational Structure.doc

## Project Elements

### Project Steps

- Maybe go through all ideas you can find.

- Update article Automatic Containment

<These ideas do not seem to have much to do with 'Relations'>

- Features to possibly add:

- Implicit connection through parent

- Implicit contents through reference target

- Imaginary reference not created if single real reference already there.

- ~ Alleen een imaginary reference op een container die meer referenties toevoegt.

- Imaginary target, logical target, physical target

- Update the whole article

- Adapt the Classes & Relations articles accordingly

Among other things:

- You have to do a line merge between the reference and the referrers.

### Products

- Updated Automatic Containment article