|  |
| --- |
| Circle Language Spec: Black Boxes |

## Public & Private Assignment

This section demonstrates how accessing system aspects looks in assignment notations.

#### Object Assignment

|  |  |
| --- | --- |
| Public Object Get  in Assignment | Friend Object Get in Assignment |
|  |  |
|  |  |
| Public Object Set in Assignment | Friend Object Set in Assignment |
|  |  |

#### Class Assignment

|  |  |
| --- | --- |
| Public Use As Class in Assignment | Friend Use As Class in Assignment |
|  |  |
|  |  |
| Public Class Set  in Assignment | Friend Class Set  in Assignment |
|  |  |

#### Class Get Connection

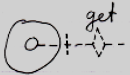
At first glance you would guess, that Class Get in assignment looks something like this:



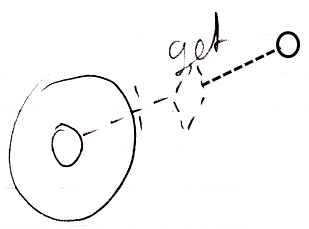
But that is Use As Class. The smaller circle on the left becomes the class of the circle on the right, requiring an Object Get, followed by a Class Set. Class Get is far less common: it retrieves a reference to the class of an object, and then you can assign the class to another object.

Using Class Get always requires the system command notation, because the more fundamental notation above is used by Use As Class.

The system command notation is not yet final, though. The Class Get access connector may look something like this:



Yielding over the class from one object to another may look something like this:



Even though the eventual notation does not look like assignment, it still involves an assigment to the object on the right, whose aspect is changed.

Yielding over Class aspect:

|  |  |
| --- | --- |
| Public Reference Class Get Connection | Friend Reference Class Get Connection |
|  |  |
|  |  |
| Public Object Class Get Connection | Friend Object Class Get Connection |
|  |  |
|  |  |

An alternative: assigning Class object to Object aspect of other symbol:

< Change notation. >

|  |  |
| --- | --- |
| Public Reference Class Get Connection | Friend Reference Class Get Connection |
| *(source is inward)* | *(source is inward)* |
|  |  |
| Public Object Class Get Connection | Friend Object Class Get Connection |
| *(source is outward)* | *(source is outward)* |

#### Remarks

##### Only Object and Class Assignments

Only Object and Class assignments are covered here.

Value and Clone assignments were covered in ***Connections*** *to System Aspects*.

The Execute aspect does not involve assignment.

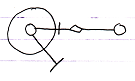
The Data aspect does not work with assignment either.

The Existance aspect may involve assignment but that is not covered here.

So it comes down to Object and Class assignments.

##### Connectors & Assignments

The following diagram shows Object Set in an assignment, and the Object Set access connector:



The access mark in the assignment and the access mark in the connector have opposite directions.

This is because for the assignment notation it was decided, that an the access mark expresses the direction of the assignment, and the connector expresses the direction of the *potential connection*, so the direction of the result. It was decided, that it would be clearer to express the direction of the *assignment* inside the diagram notation, instead of letting the assignment call express the direction of its *result*.