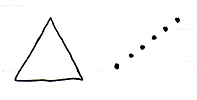
|  |
| --- |
| Circle Language Spec: Interfaces |

## Interface Aspect

While systems seem to be composed of a lot of different types of objects, all objects and types are in reality no more than a very limited set of types, that can be configured to look like other types. Those basic objects are called *system objects*. The behavior of those system objects is controlled by controlling their *system aspects*. System objects is covered in a separate chapter called *System Objects*, but there the *interface* aspect was completely left out of the story. In this part of the documentation everything about the interface aspect is laid out to make the concept of interfaces catch up with the *System Objects* story.

The Interface aspect allows you to control how objects look on the outside, while the insides of the objects can differ completely. Another concept which employs the word *interfaces* is an object (an interface) melting together with its container. From a systematic point of view they seem to be completely unrelated concepts, however in practice they are conjoinedly used. To distinguish them we call them the Interface aspect and the Interface Merging aspect. The main diagram symbolization of the Interface aspect is as follows:



The triangle is the symbolization of the Interface Merging aspect, and the dotted line is the symbolization of Interface aspect.

Interface Merging in the future will probably become an aspect that extends beyond the concept of interfaces, because it could for instance apply to the concept of default members too.