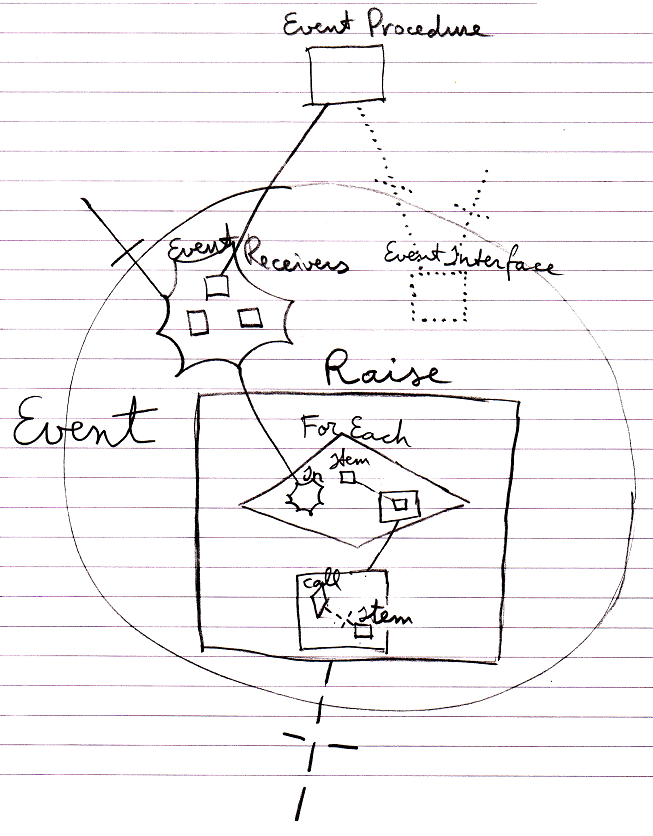
|  |
| --- |
| Circle Language Spec: Events |

## Explicit Implementation of Event Object

The concept of events is nothing more than a combination of constructs: an array of command references, a command interface and supporting the command interface, registering the command inside the array of command references and then calling all the commands in the array at specific times.

Explicitly drawing out the event concept using constructs mentioned above, the picture would look like this:



That is the way Event objects are internally implemented.

What you see is the registered event implementation (at the top), the collection of event recievers, the event interface, the Raise command and its implementation that loops through the event receiver commands and calls each of them.

Note that even though the concept is that parents can pick up messages from children, it is not the *parents* that are registered, but the *event procedures* that are registered.

Even though the event concept is just a combination of other constructs, it is such an important concept, that the new computer language explicitly defines it as a sepate construct with its own notation.