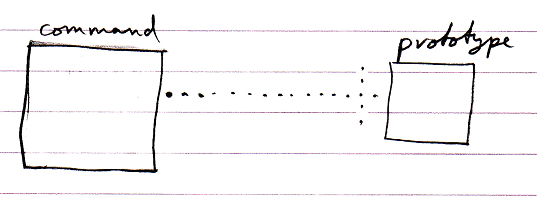
|  |
| --- |
| Circle Language Spec: Interfaces |

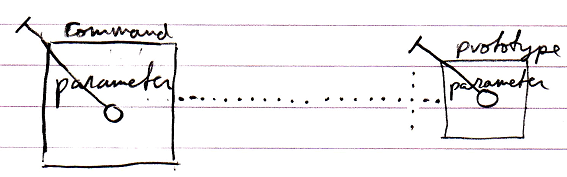
## Command Interfaces

This article is here to point out that commands can have interfaces just like objects. A command having an interface approximates the same functionality as C++ function prototypes or delegates in .NET. Only in the new computer language much more is possible. Because a command is an object, any interface concept is applicable to commands as well.

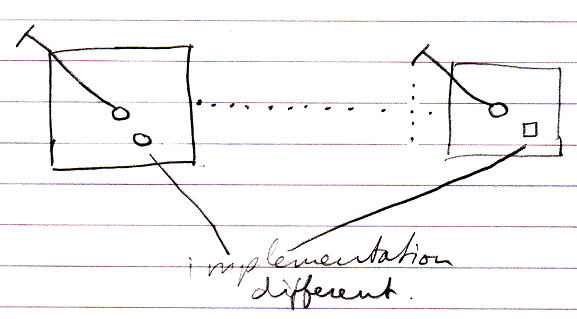
Here is your basic function prototype concept expressed in a new computer language diagram:



Here is the example when the command has a paramater.

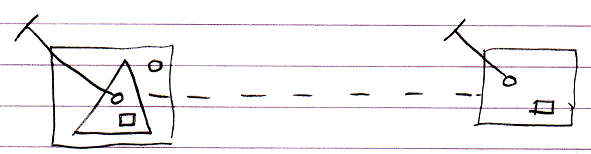


The implementation of the command can be different from its function prototype:

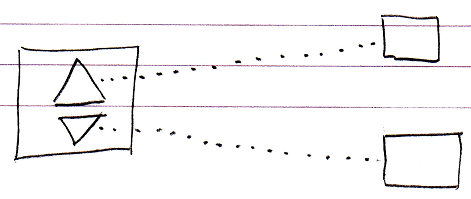


But this was classic C++ function prototype. The new computer language has more possibilities when it comes to command interfaces, some of which are displayed below.

Command inheritance:



Multiple command interfaces:



So basically anything regarding interfaces that can be done with objects, can also be done with commands.