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

## System Events

Without even defining any objects or class libraries, there are already a lot of standard events you can use. In most situations these will suffice.

They are events mostly related to changes to the data. Any type of change to the data can be picked up as an event.

More specifically: each system command has an event going off before it is executed and an event going off after it is executed: an Execting event (before) and an Executed event (after).

Derived from this comes the concept of being able to pick up a Value Set command going off. This event is called the Value Changed event. Picking up an event before the value is set also makes you able to cancel that change before it even happened. In case of the Value aspect this is the Value Changing event. So you have the opportunity to cancel a system call altogether.

Any other event of a call to a system command, such as Add, Remove and Execute can be picked up, before or after it happened, and could be cancelled.

Below is an overview of all events that derive from this concept. This results in a big list of standard events, varying in aspect, read/write direction and the before and after events. Something to denote is that Changed events are the ones most commonly used.

### Changed Events

Value Changed

Object Changed

Class Changed

Name Changed

Data Changed

Interface Changed

Interface Merged Changed

Clone Written

### Changing Events

Value Changing

Object Changing

Class Changing

Name Changing

Data Changing

Interface Changing

Interface Merged Changing

Writing Clone

### Get Events

Getting Reference

Reference Gotten

Getting Object

Object Gotten

Getting Reference-Class

Reference-Class Gotten

Getting Object-Class

Object-Class Gotten

Getting Value

Value Gotten

Cloning

Cloned

Getting Name

Name Gotten

Getting Data

Data Gotten

Getting Reference-Interface

Reference-Interface Gotten

Getting Object-Interface

Object-Interface Gotten

Getting Interface Merged

Interface Merged Gotten

### Use Events

Reference Used As Object

Using Reference As Object

Using As Class

Used As Class

Using Reference As Class

Reference Used As Class

Using As Interface

Used As Interface

Using Reference As Interface

Reference Used As Interface

### List Events

Added (used more)

Adding (used less)

Removing (used more)

Removed (used less)

Storage Count Changed

Storage Count Changing

### Execution & Existence Events

Executed

Executing

Created

Creating

Annulled

Annulling

### Pointer to Pointer Events

Perhaps the different overloads for pointer-to-pointer situations should have separate events as well, but this depends on design choices and this issue will be left open for now.