

**rawkey**

**COLLABORATORS**

	<i>TITLE :</i> rawkey		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 7, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

---

# Contents

<b>1</b>	<b>rawkey</b>	<b>1</b>
1.1	rawkey_plugin: Introduction . . . . .	1
1.2	rawkey_plugin: Usage . . . . .	2
1.3	rawkey_plugin: History . . . . .	3

---

# Chapter 1

## rawkey

### 1.1 rawkey\_plugin: Introduction

rawkey plugin

by Fabio Rotondo <fsoft@intercom.it>

rawkey plugin is a PLUGIN that offers a simple way to get keyboard events from the user even if he is not accessing a GUI gadget (the window *\*MUST\** be active anyway).

This plugin is a `_passive_` plugin, ie. it not renders anything on the window. It can handle events from these keys:

- Cursors Keys,
- Ctrl key,
- Shift keys (it *\*do\** distinguish between the two keys)
- Alt keys (it *\*do\** distinguish between the two keys)
- Amiga keys (it *\*do\** distinguish between the two keys)
- Help key
- Function keys

---

Usage

History

---

## 1.2 rawkey\_plugin: Usage

rawkey plugin

This plugin acts as a "filter" between the window and the user events. You can react to user actions by setting the right answer event to user inputs.

As I mentioned above, you can monitor some keys:

- Cursors Keys,
- Ctrl key,
- Shift keys (it \*do\* distinguish between the two keys)
- Alt keys (it \*do\* distinguish between the two keys)
- Amiga keys (it \*do\* distinguish between the two keys)
- Help key
- Function keys

To set a relationship between a key and an action you just have to define a PROC that will handle that specific event, using the `setattrs()` method.

For example, to set an event for the cursor-up key, just do:

```
[PLA_KEY_UP, {proc_up}]
```

Please, note that the taglist should be NIL terminated.

The PROC `proc_up` will do something for the cursor-up key, for example, I use this to move up/down the current selection on a ListView gadget.

The PROC `proc_up` can will receive a parameter (that defaults to NIL) that can be set by the programmer using the tag: `PLA_KEY_USERDATA`.

rawkey plugin has these publically accessible methods:

PROC `init()` OF rawkey

This method is most likely the one that you would call when NEW'ing the object; but it is safe to call at other times to reset all procs to NIL.

PROC `setattrs(tags)` OF rawkey

This method sets the relationships between a key and a proc.

`tags` is a standard taglist, that should be NIL terminated. Here are all valid tags:

<code>PLA_KEY_UP</code>	- Cursor-up key
<code>PLA_KEY_DOWN</code>	- Cursor-down key
<code>PLA_KEY_RIGHT</code>	- Cursor-right key
<code>PLA_KEY_LEFT</code>	- Cursor-left key

```
PLA_KEY_CONTROL - CTRL key

PLA_KEY_LSHIFT - Left shift key
PLA_KEY_RSHIFT - Right shift key

PLA_KEY_LALT - Left alt key
PLA_KEY_RALT - Right alt key

PLA_KEY_LAMIGA - Left Amiga key
PLA_KEY_RAMIGA - Right Amiga key

PLA_KEY_HELP - Help key

PLA_KEY_F1 - F1 key
PLA_KEY_F2 - F2 key
PLA_KEY_F3 - F3 key
PLA_KEY_F4 - F4 key
PLA_KEY_F5 - F5 key
PLA_KEY_F6 - F6 key
PLA_KEY_F7 - F7 key
PLA_KEY_F8 - F8 key
PLA_KEY_F9 - F9 key
PLA_KEY_F10 - F10 key

PLA_KEY_USERDATA - Optional argument passed to function
```

See the example source code, `rawkey_demo.e`, for some actual examples of this plugin in use.

### 1.3 rawkey\_plugin: History

- v1.0 (05.10.97)
- o Initial release.