

# Cocoon Forms: row-action widget (2.1 legacy document)

## Table of contents

1 Comments.....	3
-----------------	---

**Table of contents**

1 Concept.....3

2 Configuration.....3

**Warning:**

This document was copied as is from the Cocoon 2.1 documentation, but has not yet been fully reviewed or moved to its new home.

## 1. Concept

This is a specific type of [action widget](#) that handles frequent actions occurring on a repeater row, such as adding/removing a row and moving it up and down. These widgets should be placed inside a repeater and act on the current row.

## 2. Configuration

```
<fd:row-action id="..." action-command="add-after|delete|move-up|move-down">
<fd:label>...</fd:label> <fd:help>...</fd:help> <fd:hint>...</fd:hint> <fd:on-action> ...
</fd:on-action> </fd:row-action>
```

The **action-command** attribute should have either the value add-after, delete, move-up or move-down.

**fd:on-action** allows additional event handlers to be defined, see also [Event Handling](#). The interface to be implemented for Java event listeners is `org.apache.cocoon.forms.event.ActionListener`. The `WidgetEvent` subclass is `org.apache.cocoon.forms.event.ActionEvent`. The event handlers are called *after* the action is performed except for the delete row action where event handlers are called *before* the row is deleted.

Where all you want to do is submit a specific row on a repeater, simply add a `fd:submit` element to the widgets for the repeater.

Then, you can access the submitted row either using an event handler with `event.getSourceWidget().getParent()`, or from the flow using `form.getWidget().getSubmitWidget().getParent()`. The row itself has a `lookupWidget(widgetName)` method that can be used to access specific widgets for the row.

## 1. Comments

add your comments