

ScriptAction in Cocoon (2.1 legacy document)

Table of contents

1 Comments.....5

Table of contents

1 ScriptAction.....	3
2 Description.....	3
3 Usage.....	3
3.1 Sitemap pipeline examples.....	3
3.2 Sitemap component configuration example.....	3
3.3 Configuration.....	3
3.4 Setup.....	3
3.5 Effect on Object Model and Sitemap Parameters.....	4
4 Bugs/Caveats.....	5
5 History.....	5
6 Copyright.....	5
7 See also.....	5

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. ScriptAction

NAME	script
WHAT	The ScriptAction component is used to executes any script that can be run by the BSF.
TYPE	Action
BLOCK	bsf
CLASS	org.apache.cocoon.action.ScriptAction
SINCE	Cocoon 2.1
CACHEABLE	not applicable

2. Description

A simple action that executes any script that can be run by the BSF system. BSF supports script languages like javascript, python, etc.

3. Usage

This Action is used for quick prototyping of Action, realizing an Action in a script language instead of java language.

3.1. Sitemap pipeline examples

The following sample executes the script action/my-action.js for each URI matching pattern *.vm. If the ScriptAction succeeds, the pipeline continues, invoking the VelocityGenerator, and serializing the XML stream.

```
<map:match pattern="*.vm"> <map:action type="script" src="action/my-action.js">
<map:generate type="velocity" src="{1}.vm"/> <map:serialize/> </map:action> </map:match>
```

3.2. Sitemap component configuration example

The following sample configures an ScriptAction, naming it script.

```
<map:actions... <map:action name="script" src="org.apache.cocoon.acting.ScriptAction">
logger="sitemap.action.script"/> <!-- optional action configuration --> ... </map:actions>
```

3.3. Configuration

ScriptAction has no configuration options.

3.4. Setup

ScriptAction determines the name of the script getting executed from its src attribute.

The language of the script file is auto-detected by the BSF system from the extension of the script.

The next table lists the default mapping of BSF:

Script language	Extension
javascript	js
jacl	jacl
netrexx	nrx
java	java
javaclass	class
bml	bml
vbscript	vbs
jscript	jss
perlscript	pls
perl	pl
jpython	py
lotusscript	lss
xslt	xslt
pnuts	pnut
beanbasic	bb

ScriptAction registers following objects before invoking the script:

name	Object Typ	Comment
resolver	SourceResolver	Cocoon's source resolver
objectModel	Map	Cocoon's object model
parameters	Parameters	Cocoon's action parameter
actionMap	Map	Used for passing objects from the script back to the ScriptAction, and to Cocoon's sitemap
logger	Logger	Cocoon's logger of this ScriptAction
request	Request	Request provided by the object model
scriptaction	Action	The instance of this ScriptAction
manager	Manager	Cocoon's manager of this ScriptAction

These objects are accessible from within the script.

3.5. Effect on Object Model and Sitemap Parameters

ScriptAction checks the existence of the key scriptaction-continue in the actionMap. If this key exists ScriptAction returns actionMap, otherwise null is returned.

Objects available in the actionMap are available in the sitemap.

4. Bugs/Caveats

Using ScriptAction relies heavily on the setting of the key scriptaction-continue in the mapAction.

Be aware to provide script language implementation, beside BSF implementation.

5. History

07-24-03: initial creation

04-02-04: Updated to Jakarta BSF

6. Copyright

Copyright (C) 1999-2004 The Apache Software Foundation. All rights reserved.

7. See also

A general documentation about actions is available at [Actions](#).

Further Documentation visit [Jakarta BSF Project](#).

1. Comments

add your comments