

# Offline Page Generation with the Command Line Interface (2.1 legacy document)

## Table of contents

1 Comments.....	4
-----------------	---

## Table of contents

1 Overview.....	3
2 Invoking the CLI.....	3
3 Configuring the CLI.....	3
3.1 Using an Xconf file.....	3
3.2 Command Line Parameters.....	3
3.2.1 Specifying Targets.....	4
3.2.2 URI Files.....	4
3.2.3 Broken Link Handling.....	4

**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. Overview

The Command Line Interface provides access to Cocoon's offline generation capabilities.

This page gives details of how to configure and use the CLI. Details of the concepts behind offline page generation are given on the offline generation [overview](#) page.

## 2. Invoking the CLI

The CLI can be invoked from the command line. Change to the root directory of your Cocoon distribution, and then, on Unix use: `./cocoon.sh cli <parameters>` and on Windows use `cocoon.bat cli <parameters>`

The relevant parameters are detailed in the following sections.

## 3. Configuring the CLI

The CLI has two methods of configuration, with an xconf file, and using command line parameters.

The xconf method is the newer, and gives access to a wider range of features, and is thus explained first.

Whilst the xconf method provides access to more features, the command line parameter method is more stable, as there are currently plans to improve the xconf format to allow greater flexibility. If you require a stable and consistent method for accessing the CLI, it is recommended that you use the command line parameter method.

Copy `$COCOON/lib/optional/servlet-2_3.jar` to your lib directory (for example, `WEB-INF/lib`).

### 3.1. Using an Xconf file

To start the CLI using an xconf file, on Unix do `./cocoon.sh cli -x <xconf file>` or on Windows: `cocoon cli -x <xconf file>`.

For a sample xconf file, with comments describing each option, see the [configuration](#) page.

### 3.2. Command Line Parameters

You can get a listing of the available parameters on unix with `./cocoon.sh cli -h` or on Windows with `cocoon cli -h`. This should give a listing something like:

```
----- Executing ----- Main Class: org.apache.cocoon.Main usage: cocoon
cli [options] [targets] ----- cocoon 2.1
Copyright (c) 1999-2003 Apache Software Foundation. All rights reserved.
----- -a,--userAgent use given string for
user-agent header -e,--confirmExtensions confirm that file extensions match mime-type of
pages and amend filename accordingly (default is true) -C,--configFile specify alternate
location of the configuration file (default is ${contextDir}/cocoon.xconf) -D,--defaultFilename
specify a filename to be appended to a URI when the URI refers to a directory -L,--loadClass
```

specify a class to be loaded at startup (specifically for use with JDBC). Can be used multiple times -P,--precompileOnly generate java code for xsp and xmap files -V,--verbose enable verbose messages to System.out -b,--brokenLinkFile send a list of broken links to a file (one URI per line) -c,--contextDir use given dir as context -d,--destDir use given dir as destination -f,--uriFile use a text file with uris to process (one URI per line) -h,--help print this message and exit -k,--logKitconfig use given file for LogKit Management configuration -l,--Logger use given logger category as default logger for the Cocoon engine -p,--accept use given string for accept header -r,--followLinks process pages linked from starting page or not (boolean argument is expected, default is true) -u,--logLevel choose the minimum log level for logging (DEBUG, INFO, WARN, ERROR, FATAL\_ERROR) for startup logging -v,--version print the version information and exit -w,--workDir use given dir as working directory -x,--xconf specify a file containing XML configuration details for the command line interface Note: the context directory defaults to './webapp'

For details of the meaning of each specific parameter, see the [overview](#) page.

### 3.2.1. Specifying Targets

The command line parameter method does not have access to all of Cocoon's URI handling features. However, it is possible to specify multiple URIs to be crawled, all of which will be written to the same destination, and that destination (specified by the -d or --destDir option, may be a file URI or any other protocol for which a ModifiableSource exists (e.g. FTP).

### 3.2.2. URI Files

A URI file offers a simple way to specify multiple URIs. The file is treated as one URI per line.

### 3.2.3. Broken Link Handling

If a broken link file is specified, all broken links will be written to this file, in text format, one URI per line.

## 1. Comments

add your comments