# Scheduler

The Cocoon Scheduler is a recently-added component, which uses the Avalon Cornerstone scheduler. It allows you do to anything at a specific interval or at a certain time every day, week, month, etc, much like the cron daemon. It currently lives in the 2.1-dev scratchpad.

## Starting Out

First, you must compile Cocoon with the scratchpad enabled. Make sure that `exclude.webapp.scratchpad` and `exclude.scratchpad` are either commented out or set to **false**.

Second, compile and deploy Cocoon as you would normally.

Third, you must make some changes to your `WEB-INF/cocoon.xconf` file. Add the following lines to the bottom of the file, inside the closing `</cocoon>` tag:

```
<!-- This is the cornerstone scheduler -->
<component role="org.apache.avalon.cornerstone.services.scheduler.TimeScheduler"
           class="org.apache.cocoon.components.scheduler.DefaultTimeScheduler"/>

<!-- This is the cornerstone thread manager -->
<component role="org.apache.avalon.cornerstone.services.threads.ThreadManager"
           class="org.apache.cocoon.components.scheduler.DefaultThreadManager">
  <thread-group>
    <name>default</name>
  </thread-group>
</component>

<!-- This is an extended cocoon scheduler -->
<component role="org.apache.cocoon.components.scheduler.Scheduler"
           class="org.apache.cocoon.components.scheduler.DefaultScheduler">
    <triggers>
        <!-- This is a sample trigger -->
        <trigger name="aggregatorTrigger"
                 target="org.apache.avalon.cornerstone.services.scheduler.Target/aggy">
            <timed type="periodic">
                <period>10000</period> <!-- ms, e.g. 10 seconds -->
            </timed>
        </trigger>
    </triggers>
</component>

<!-- This is a sample target, that is called as configured in the sample above -->

<component role="org.apache.avalon.cornerstone.services.scheduler.Target/aggy"
           class="org.apache.cocoon.components.scheduler.TestTarget"/>
```

Now, restart your servlet container of choice, and check it's STDOUT. In Tomcat you probably want `$TOMCAT_HOME/logs/stdout.log`. Once everything has started, you should now notice a message popping up every ten seconds:

```
Target aggregatorTrigger triggered at Mon Aug 25 01:21:16 CDT 2003
Target aggregatorTrigger triggered at Mon Aug 25 01:21:26 CDT 2003
Target aggregatorTrigger triggered at Mon Aug 25 01:21:36 CDT 2003
Target aggregatorTrigger triggered at Mon Aug 25 01:21:46 CDT 2003
Target aggregatorTrigger triggered at Mon Aug 25 01:21:56 CDT 2003
Target aggregatorTrigger triggered at Mon Aug 25 01:22:06 CDT 2003
```

Obviously, this is a very basic example, and a more complicated example could involve periodic file downloading, user management, database operations, etc.

There are currently two different types of triggers available. One is a periodic trigger, which is activated every N milliseconds, and the other is a "cron" type trigger where you have a little more flexibility with how triggers are activated.

## Periodic Triggers

### Configuration Parameters

**offset**:Initial delay in milliseconds, -1 means fire immediately

**period**: After initial delay in milliseconds, -1 means fire only once after initial delay

The following trigger will wait 2 seconds before triggering, and then trigger every 10 seconds afterwards:

```
<trigger name="test"
         target="org.apache.avalon.cornerstone.services.scheduler.Target/test">
  <timed type="periodic">
    <offset>2000</offset>
    <period>10000</period> <!-- ms, e.g. 10 seconds -->
  </timed>
</trigger>
```

## Cron Triggers

### Configuration Parameters

*(Untested, someone wanna futz with this to make sure it works?) – TonyCollen*

**minute**: the minute at which job is scheduled. (0-59)

**hour**: hour at which job is scheduled. (0-23 or -1 for every hour)

**month**: the month at which job is scheduled. (0-11 or -1 for every month)

**year**: the year when job is scheduled (-1 implies every year)

**day**: either day of week or day of month depending on value of isDayOfWeek.

**isDayOfWeek**: set to `true` if you want **day** to represent the day of the week (1-7), otherwise, set to `false` to represent the day of the month (1-31).

```
<trigger name="test"
         target="org.apache.avalon.cornerstone.services.scheduler.Target/test">
  <timed type="cron">
    <minute>34</minute>
    <hour>11</hour>
    <month>-1</month>
    <year>-1</year>
    <day>-1</day>
    <isDayOfWeek>true</isDayOfWeek>
  </timed>
</trigger>
```

### Coming Soon: Creating your own trigger target.