

# ChainSaw

## Log Cocoon with Chainsaw v2

Chainsaw is a powerful GUI-based Log viewer. It's a companion application to Log4j and seems to have replaced [Log Factor 5](#)

See also [Chainsaw homepage](#) and [<http://logging.apache.org/log4j/docs/chainsaw.html>]

## Getting Chainsaw v2

### Web start=

- you can launch Chainsaw with Web Start [here](#)
- at the startup popup, select "SocketReceiver" on "Port 4445"

### Installing Chainsaw locally

(not tested yet)

### Note

One thing to note: The Chainsaw v2 is based on the new version of log4j - 1.3 - which is currently alpha. Cocoon probably uses log4j 1.2.8, and [SocketAppender](#) will only work partially between these two versions (SocketAppender sends serialized [LoggingEvent](#) objects over the wire, and [LoggingEvent](#) changed in 1.3).

If you can log to a file, use Chainsaw to tail your log file - use a [LogFilePatternReceiver](#) instead of a [SocketReceiver](#).

There's an example configuration available from the Welcome tab that shows how to configure Chainsaw to process log files (the [LogFilePatternReceiver](#) configuration section).

Copy this configuration file, change the configuration to match the location and format of your log file, and specify the URL to this configuration file in the application-wide preferences - configuration URL box.

When you restart Chainsaw, the log file will get processed (and if you set 'tailing' to 'true' in the config, your file will be tailed from inside Chainsaw).

## Configuring Cocoon (using [SocketAppender](#))

- Set this properties in WEB-INF/web.xml (it's already in there, just uncomment it) :

```
<init-param>
  <param-name>logger-class</param-name>
  <param-value>org.apache.avalon.excalibur.logger.Log4JLoggerManager</param-value>
</init-param>

<init-param>
  <param-name>log4j-config</param-name>
  <param-value>/WEB-INF/log4j.xconf</param-value>
</init-param>
```

- And in WEB-INF/log4j.xconf:

```
<appender name="COCOON_DEFAULT"
class="org.apache.log4j.net.SocketAppender">
  <param name="RemoteHost" value="localhost" />
  <param name="Port" value="4445" />
</appender>

<root>
  <priority value="debug" />
  <appender-ref ref="COCOON_DEFAULT" />
</root>
```

- you might want to add the dtd (`{src}\java\org\apache\log4j\xml\log4j.dtd`) to WEB-INF/

## Configuring Cocoon (using a log File)

- Set this properties in WEB-INF/web.xml (same as above)
- And in WEB-INF/log4j.xconf (should already be there):

```
<appender name="COCOON_DEFAULT" class="org.apache.log4j.FileAppender">
  <param name="File" value="${context-root}/WEB-INF/logs/log4j.log" />
  <param name="Append" value="false" />
  <layout class="org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="%t %-5p %c{2} - %m%n"/>
  </layout>
</appender>

<root>
  <priority value="debug" />
  <appender-ref ref="COCOON_DEFAULT" />
</root>
```

- you might want to add the dtd (\\src\\java\\org\\apache\\log4j\\xml\\log4j.dtd) to WEB-INF/

## Other Ressources:

- <http://www.codeconsult.ch/bertrand/archives/000366.html>
- <http://logging.apache.org/log4j/>