

Logging (J2EE only)

The Natural Web I/O Interface client and Natural for Ajax use the Java Logging API. In case of problems with the Natural Web I/O Interface client or Natural for Ajax, you can enable logging and thus write the logging information to an output file. This should only be done when requested by Software AG support.

You configure logging using the configuration tool.

Note:

Some logging information is also written to the console, regardless of the settings in the configuration file. The console shows the information which is normally provided by the logging levels SEVERE, WARNING and INFO.

This chapter covers following topics:

- Name and Location of the Configuration File
 - Logging on Sun Java System Application Server
 - Logging on JBoss Application Server
 - Invoking the Logging Configuration Page
 - Overview of Options for the Output File
-

Name and Location of the Configuration File

The name of the configuration file is *natlogger.xml*. The path to this file depends on the application server and type of client that you are using.

- **JBoss Application Server**

Natural Web I/O Interface client:

```
<application-server-install-dir>server/default/deploy/naturalunicode.rar/log
```

Natural for Ajax:

```
<application-server-install-dir>/server/default/deploy/njx<nn>ra.rar/log
```

- **Sun Java System Application Server**

Natural Web I/O Interface client:

```
<application-server-install-dir>/domains/domain1/applications/j2ee-modules/naturalunicode/log
```

Natural for Ajax:

```
<application-server-install-dir>/domains/domain1/applications/j2ee-modules/njx<nn>ra/log
```

Logging on Sun Java System Application Server

On Sun Java System Application Server, the logging information is written to the normal server log. That is because Sun Java System Application Server uses the same Java Logging API as the Natural Web I/O Interface client and Natural for Ajax. You can thus use a powerful Sun Java System Application Server tool, the Log Viewer, for analyzing the log. The Log Viewer is started from the web-based Admin Console; for further information, see the documentation of the Sun Java System Application Server.

We recommend that you disable the file handler in the configuration file *natlogger.xml*. Thus, you avoid that the logging information is written to two different log files (that is, the normal server log and the output file defined in *natlogger.xml*).

Logging on JBoss Application Server

JBoss Application Server uses a different logging API (log4j). In this case, we recommend that you enable the file handler in the configuration file *natlogger.xml*.

Invoking the Logging Configuration Page

The content of the configuration file *natlogger.xml* is managed using the **Logging Configuration** page of the configuration tool.

To invoke the Logging Configuration page

1. In the frame on the left, choose the **Logging Configuration** link.

The **Logging Configuration** page appears in the right frame. Example for the Natural Web I/O Interface client:

Logging Configuration

Specify the output log file characteristics.

- "/" : The local pathname separator
- "%t": The system temporary directory
- "%h": The value of the "user.home" system property
- "%g": The generation number to distinguish rotated logs
- "%u": A unique number to resolve conflicts
- "%%": Translates to a single percent sign "%"

File pattern name:

File type:

File size (0=unlimited):

Number of files:

File enabled: Yes No

Append mode: Yes No

Specify log levels for individual modules. The available settings are:

- SEVERE: Events that interfere with normal program execution
- WARNING: Warnings, including exceptions
- INFO: Messages related to server configuration or server status, excluding errors
- CONFIG: Messages related to server configuration
- FINE: Minimal verbosity
- FINER: Moderate verbosity
- FINEST: Maximum verbosity

Communication:

Resource adapter:

Session beans:

Message beans:

Configuration file:

Logging:

Utilities:

Natural Web I/O Interface pages:

With Natural for Ajax, additional modules are provided at the bottom of the **Logging Configuration** page.

2. Specify the characteristics of the output file as described below in the section *Overview of Options for the Output File*.
3. Specify the log levels for individual modules by selecting the log level from the corresponding drop-down list box.

A brief description for each log level is provided on the **Logging Configuration** page.

4. Choose the **Save Configuration** button to write the modifications to the configuration file.

Caution:

When you do not choose the **Save Configuration** button but logout instead or leave the configuration tool by entering another URL, your modifications are not written to the configuration file.

Overview of Options for the Output File

The following options are provided for specifying the characteristics of the output file:

Option	Description
File pattern name	<p>The pattern for generating the output file name. Default: "%h/nwolog%g.log".</p> <p>The default value means that an output file with the name <i>nwolog<number>.log</i> will be created in the home directory of the user who has started the application server.</p> <p>For detailed information on how to specify the pattern, see the Java API documentation at http://java.sun.com/j2se/1.5.0/docs/api/java/util/logging/FileHandler.html.</p>
File type	<p>The format of the output file. Select one of the following entries from the drop-down list box:</p> <ul style="list-style-type: none"> ● Text format Output in simple text format (default). ● XML format Output in XML format. <p>The corresponding formatter class is then used.</p>
File size	The maximum number of bytes that is to be written to an output file. Zero (0) means that there is no limit. Default: "0".
Number of files	The number of output files to be used. This value must be at least "1". Default: "10".
File enabled	If set to Yes (default), the file handler is enabled. If set to No , the file handler is disabled.
Append mode	If set to Yes , the logging information is appended to the existing output file. If set to No (default), the logging information is written to a new output file.