

# Logging (2.6)

Once you have Sakai installed, configured and started, you can monitor Sakai by watching the logs. The log level for the standard Sakai source code and the demo is set to show info and warnings only. Watch for the `WARN:` messages. There are going to be some "normal" ones at startup, and some will likely slip by at runtime, but any warning is potentially something you might want to check out.

Logging levels can be specified in `sakai.properties`. This augments and overrides the levels set in the default config file. Example:

```
log.config.count=3
log.config.1 = ALL.org.sakaiproject.log.impl
log.config.2 = OFF.org.sakaiproject
log.config.3 = DEBUG.org.sakaiproject.db.impl
```

This uses the established (if awkward) method of having a `name.count` followed by `name.1`, `name.2` etc. to form an array of strings for the value "name". In this case, the name is "log.config". The values are of the form `LEVEL.logger`, and the possible levels are: `OFF TRACE DEBUG INFO WARN ERROR FATAL ALL`.

Sakai uses [log4j](#) for logging. See the official log4j documentation for more information about how to configure it if you have questions, but a few notes are collected here below.

To change the logging for Sakai in the source modify `kernel/kernel-common/src/main/config/log4j.properties` the following property:

```
log4j.logger.org.sakaiproject=INFO
```

To turn on debug logging for all of Sakai, change the value from `INFO` to `DEBUG`. In order to enable debug logging for a single Sakai components, add a line such as in the following example, which will leave most of Sakai at `INFO`, but generate `DEBUG` level messages for the SQL service:

```
log4j.logger.org.sakaiproject=INFO
log4j.logger.org.sakaiproject.component.framework.sql.BasicSqlService=DEBUG
```

The logging controls are part of the `LogConfigurationManager`, implemented as a component in the Kernel. It can be disabled, if that's desired, with an entry in `sakai.properties`:

```
enabled@org.sakaiproject.log.api.LogConfigurationManager = false
```

For Mac and \*nix systems, the most important log is found in Tomcat's `logs/catalina.out`. It can be instructive to watch this log as Tomcat is starting up, by using a startup command like the following:

```
bin/startup.sh; tail -f logs/catalina.out
```

Tomcat on Windows tends to be a little more puzzling about its logs, and it includes more of them, but its default behavior is to open `catalina.out` in a new window as soon as you start Tomcat. If you need more information about the logs in Windows, we'll refer you to the official Tomcat documentation.

 The SMTP server logs from Sakai will be written to the `$(CATALINA_HOME)/sakai/logs` directory.