Logging

SPIN logging features were designed to withstand "semi-trusted" investigators who could potentially attempt to conceal unwarranted investigations (publications).

1. Logging Query Activity
2. Application Logging with Log4j

Logging to Local Database

By default SPIN logs to a local memory resident database provided by the JDK called Derby.
You can override Derby and log to your own database (such as mysql) by providing a copy for your database provider:

- edit SqlMapConfig.properties
- edit LogEntrySqlMap.xml
- edit spin-ibatis-logentry-create.sql

Application Logging with Log4j

SPIN uses log4j to set the logging level (DEBUG, INFO, WARN, ERROR, FATAL) and set the output targets (FILE, CONSOLE).

Log4J is described in detail here: http://logging.apache.org/log4j/docs/ , though most users will only need to set the logging level and/or the location of the log file.

Log4j logging levels are hierarchal, meaning that an ERROR message will also be logged if the level is set to DEBUG. The relationship between the levels is as follows:

DEBUG > INFO > WARN > ERROR >FATAL

Log4J also allows multiple *appender* types such as CONSOLE or FILE or even SMTP.

Defaults

By default, SPIN will use the log4j configuration file located in /opt/spin/main/conf

This lo4j configuration will log all WARN, ERROR, or FATAL level messages to Tomcat STDOUT.
Any SPIN messaging related classes will log at the INFO level, also to Tomcat STDOUT.