# Schema Wizard Server-Side Logging Standards and Strategy

The Schema Wizard server utilizes logging in Java and Python. Logs are managed by configuration files and use project-level appenders.

## Problem

During external test deployments, it was reported that the Schema Wizard server’s log files generated too much content, which quickly depleted free disk space.

## Strategy

The Schema Wizard server now logs up to the *INFO* level to Standard Output, and logs up to the *DEBUG* level to a file located on the local filesystem. This generates useful and non-superfluous logging in the console and limits the storage space taken up by log files.

## Constraints

* Log files will be rolling – only one log file will be created per logger.
* The file size for individual logs will be limited to 10MB.

## Versions

The packages and versions of logging are listed below:

Java: log4j 1.2.17

Python: logging (built-in) PEP 282

## Considerations

* Docker and Docker Compose logs echo standard output, so they will only display up to the *INFO* level when using commands such as:

docker logs <container>

or

docker-compose logs <container>

or parsing the docker logs files.

* Docker log files are limited by a run-time configuration option specified on the command-line or the docker-compose file. That command is:

--log-opt max-size=10m

## Development

Developers who would like to use the *DEBUG* logs may either refer to the generated log files or modify the console output log level in the log properties files. These files are located at:

digitaledge\_datamodeltoolkit \interpretation-engine\src\main\python\logging.conf

digitaledge\_datamodeltoolkit\schema-wizard\src\main\resources\log4j.properties

## Appendix

Log levels (lowest level first):

1. WARN
2. TRACE
3. OFF
4. INFO
5. FATAL
6. ERROR
7. DEBUG
8. ALL