Log settings

The log settings can be used to define precisely how **logging** works. It is possible to specify for which users and for which screen actions logging is to take place. Furthermore, it is also possible to specify which program parts are to log.
The **log levels** (Debug, Info, Warn, Error, Fatal, All) determine the prioritization of error messages, where ***Debug*** is the lowest level and outputs the most messages (e.g.: "loading record with number..."), ***Info*** only outputs messages that have a higher priority than Info, and ***Error / Fatal*** only outputs error messages.

Where the logs are to be written can be set via so-called appenders. In Webdesk EWP, 4 appenders are available by default:

Appender	Description
Database	This appender is the most convenient log medium; here every event is written to a log per action and user. This log applies for the entire time spent within an action (which means that paging within an action counts as well!). The disadvantage is that this type of logging is also the most expensive, since write accesses to the database occur constantly.
Console	This appender writes to the console of the servlet container (e.g. Tomcat) in which the Webdesk EWP server runs. It is a text file with a chronological structure. It is naturally difficult here to filter out log events of individual users.
Error file	This appender is a special appender that only accepts events of type ERROR or FATAL. The target is a file in the Webdesk directory (webdesk3/WEB-INF/logs/error.log)
Log file	This appender writes to a log file within the Webdesk application. In contrast to the console log, in this case only log events from Webdesk are written to this log, whereas with the console appender other web applications in the same container could also log to this appender. (webdesk3/WEB-INF/logs/log4j.log)

Database logging (appender = DATABASE) is only activated if the **logged-in user and the called action** have been activated for logging. Here the correct parameterization of the upper part of the form is of essential importance!
Conversely, it is not necessary to activate actions or users for logging if logging is to go to media other than the database (console, file, error file).

Important loggers:

Name of the logger	Description
shark	Logger for the workflow engine
at.workflow.webdesk	Logger for all Webdesk classes and services
org.hibernate.SQL	Logger for all SQL-generating classes in Hibernate
DatabaseManager	Logger for DODS (O/R mapping tool used by Shark)

Persistence	Logger for DAOs of the workflow engine

Links

- Log4j manual[1]

Felder

Name	Wert
Modul	Portal & Organisation (po)
Webdesk Actionname	po_editLogSettings
Artefakt-Typ	Action

1. http://logging.apache.org/log4j/docs/manual.html