

Log Einstellungen

Mit diesem Formular kann die **Funktionsweise der Protokollierung** punktgenau festgelegt werden. Es ist möglich festzulegen, bei welchen Usern und bei welchen Bildschirm-Aktionen gelogged werden soll. Des Weiteren ist es möglich auch festzulegen, welche Programmteile protokollieren sollen.

Wohin die Protokolle geschrieben werden sollen ist über die so genannten Appender einstellbar. Im Webdesk EWP stehen standardmäßig 4 Appender zur Verfügung:

Appender	Beschreibung
Database	Dieser Appender stellt das komfortabelste Log-Medium dar, hier wird jedes Event in einen Log pro Aktion und User geschrieben. Dieses Log gilt für die gesamte Verweildauer innerhalb einer Aktion (das bedeutet, dass auch das Blättern innerhalb einer Aktion dazuzählt!) Der Nachteil ist, dass diese Protokollierungsart auch die aufwendigste darstellt, da ständig schreibende Datenbankzugriffe erfolgen.
Console	Dieser Appender schreibt auf die Konsole des Servlet Containers (z.B. Tomcat) in welchem der Webdesk EWP Server abläuft. Es handelt sich hierbei um ein Text-File, das chronologisch aufgebaut ist. Es ist hier naturgemäß schwierig, Logevents von einzelnen Usern herauszufiltern.
Error-File	Dieser Appender ist ein spezieller Appender, welcher nur Events vom Typ ERROR oder FATAL entgegennimmt. Das Ziel ist ein File im Webdesk Verzeichnis (webdesk3/WEB-INF/logs/error.log)
Log-File	Dieser Appender schreibt auf ein Logfile innerhalb der Webdesk Applikation. Im Unterschied zum Console-Log werden in diesem Fall nur Logevents vom Webdesk in dieses Log geschrieben. Während beim Console Appender auch andere Webapplikationen im selben Container auf diesen Appender loggen könnten. (webdesk3/WEB-INF/logs/log4j.log)

- **Alle Aktionen loggen**
Durch anklicken der Check-Box werden alle Aktionen (Menüpunkte) geloggt (protokolliert)
- **Aktionen, welche geloggt werden sollen**
Auswahl einer bestimmten Aktion, die protokolliert werden soll
- **Alle User loggen (Log All Users?)**
Durch anklicken der Check-Box werden alle User geloggt (protokolliert)
- **Folgende User loggen**
Auswahl der User aus der Übersicht "Available Users"
 - **Available Users** - zeigt die verfügbaren User an
 - **Your Selection** - zeigt die ausgewählten User an
 - Einschränkung auf eine bestimmte Anzahl von Usern (anklicken eines Benutzers aus der Auswahlliste, dann anklicken des einfachen Pfeils - so werden die markierten Benutzer übernommen. Klickt man auf den Doppelpfeil, so werden alle Benutzer übernommen)
- **Loglevel of Jobs**
Debug, Info, Warn, Error, Fatal, All
Priorisierung von Fehlermeldungen (wobei *Debug* die unterste Ebene darstellt und die meisten Meldungen ausgibt - Bsp. "lade Datensatz mit Nummer...". *Info* - es werden nur Meldungen ausgegeben, die eine höhere Priorität haben als Info. *Error / Fatal* - es werden nur Fehlermeldungen ausgegeben)
- **Loglevel of Connectors**
Analog dem Loglevel der Jobs
- **Category**
Hier kann der Logger auf Java- Modulebene festgelegt werden (z.B.: at.workflow.webdesk > loggt das angegebene Modul)
- **Additivity**
 - **true / false**
Definiert, ob die Logevents auch an alle übergeordneten Log-Appender weitergegeben werden sollen. Beispiel: Es gibt einen Root-Logger, der auf die Konsole loggt, wenn nun ein untergeordneter Logger auf die DB logged, und dieses Flag auf true steht, werden alle Events auch an den Console-Appender weitergeleitet!
- **Loglevel**

Debug, Info, Warn, Error, Fatal, All

Priorisierung von Fehlermeldungen (wobei **Debug** die unterste Ebene darstellt und die meisten Meldungen ausgibt - Bsp. "lade Datensatz mit Nummer...". **Info** - es werden nur Meldungen ausgegeben, die eine höhere Priorität haben als Info. **Error / Fatal** - es werden nur Fehlermeldungen ausgegeben)

- **Appender**

Der Appender definiert, wohin eine Kategorie geloggt (protokolliert) wird (siehe add category) > gibt an, wohin die Ausgaben umgeleitet werden:

- **Datenbank** - Ausgaben werden in die Datenbank umgeleitet (und können mit der Ansicht Logeinträge später eingesehen werden!)
- **Console** - die Ausgaben werden auf die Console ausgegeben (das ganze zu finden im tomcat5/logs/stdout.log)
- **Log File** - die Ausgabe wird auf ein spezielles log- File ausgegeben
- **Error File** - in dieser Datei sollten nur die wirklichen Fehlermeldungen protokolliert werden (File: webdesk3/WEB-INF/logs/error.log)

- **add Appender**

Hinzufügen eines neuen Appenders

- **delete Appender**

Die Definition wird gelöscht

- **add Category**

Anhängen eines Logs an ein Package (Name muss mit Package übereinstimmen)

- **delete Category**

Die Kategorie wird gelöscht

NOTIZ

Die Datenbankprotokollieren (Appender = DATABASE) ist ausschließlich aktiviert, wenn der **angemeldete User und die aufgerufene Aktion** zur Protokollierung aktiviert wurden. Hier ist die korrekte Parametrierung des oberen Teils des Formulars von wesentlicher Bedeutung!

Wichtige Logger:

Name des Loggers	Beschreibung
shark	Logger für die Workflow Engine
at.workflow.webdesk	Logger über alle Webdesk Klassen und Services
org.hibernate.SQL	Logger über alle SQL generierende Klassen im Hibernate
DatabaseManager	Logger über DODS (O/R Mapping Tool, welches von Shark verwendet wird)
Persistence	Logger über DAOs von Workflow Engine

NOTIZ

Umgekehrt ist es nicht notwendig, Aktionen oder Benutzer für die Protokollierung zu aktivieren, wenn in andere Medien als die Datenbank gelogged werden soll. (Beispiel: CONSOLE, FILE, ERROR-FILE)

Bei Fragen zur detaillierten Funktionsweise der Parameter kontaktieren Sie bitte die Workflow EDV GmbH.

Verweise

Log4j Manual

<http://logging.apache.org/log4j/docs/manual.html>