

Nachdenkzettel Logging

Cazim Ukela 40064 | Daniel Hiller 40181 | Niklas Mäckle 40060

Vorname, Name, Matrikelnummer

1. Kennzeichnen Sie in der Config die Stellen wo über das

- was geloggt wird
- wieviel geloggt wird
- wo geloggt wird
- wie geloggt wird entschieden wird

```
<Configuration>
  <Appenders>
    <File name="A1" fileName="A1.log" append="false">
      <PatternLayout pattern="%t %-5p %c{2} - %m%n"/>
    </File>
    <Console name="STDOUT" target="SYSTEM_OUT">
      <PatternLayout pattern="%d %-5p [%t] %C{2} (%F:%L) - %m%n"/>
    </Console>
  </Appenders>
  <Loggers>

    <!-- You may want to define class or package level per-logger rules -->
    <Logger name="se2examples.core.businessLogic.VehicleManager" level="debug">
      <AppenderRef ref="A1"/>
    </Logger>
    <Root level="debug">
      <AppenderRef ref="STDOUT"/>
    </Root>
  </Loggers>
</Configuration>
```

2. Geben Sie je ein Beispiel wann Sie den loglevel

- error Wenn eine Exception geworfen wird.
 - info Wenn erfolgreich ein externes File geladen wurde.
 - debug Kann anstelle von Sys.out verwendet werden, um seinen Code zu debuggen.
- verwenden

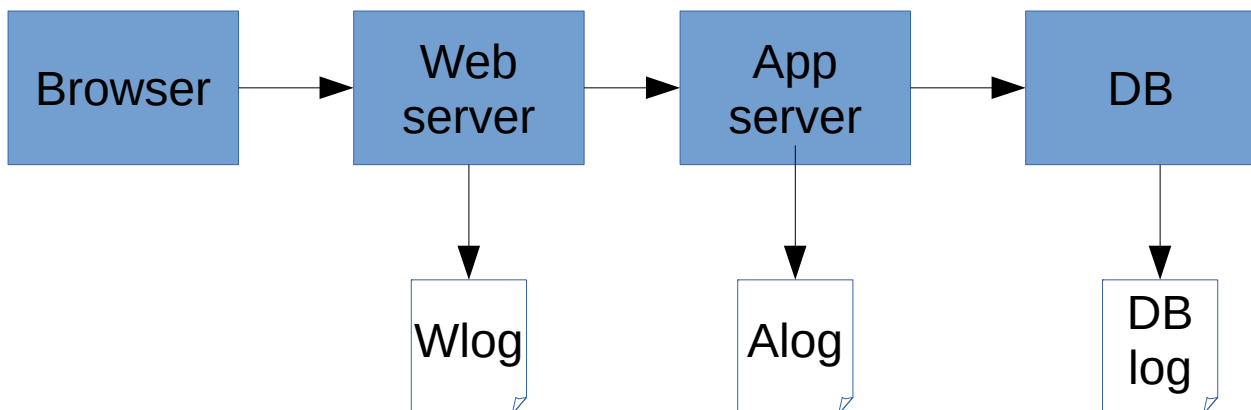
3. Sie verwenden einen FileAppender für das Logging. Jetzt soll Ihre Application im Datacenter laufen. Was machen Sie mit dem FileAppender?

Entweder den Rolling Appender bis zu einer gewissen Menge oder bis zu einem gewissen zeitlichen Intervall limitieren, sprich in von vorne beginnen zu lassen.

4. Macht logging Ihre Application langsamer? Was passiert wenn Sie `log.debug(„foobar“);` aufrufen? Wie sollte sich das Logging Subsystem verhalten?

Ja, aber nur sehr marginal.

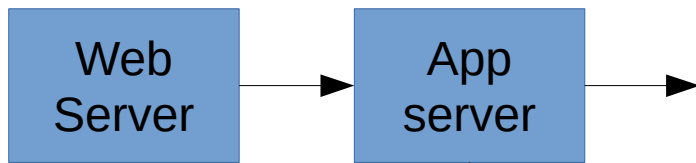
5. Ein Request an Ihre Application durchläuft einen Proxy Server, dann einen Web Server, dann einen Application Server und dann die Datenbank. Auf jedem Server loggen Sie die Requests. Welches Problem tritt auf?



Man muss idR. mehrere Logfiles anschauen um das Problem zu finden, ebenfalls koennte es passieren dass bei uebergreifenden Problem diese nicht richtig zugeordnet werden koennen.

6. Was sollten Sie pro Komponente/Tier loggen?

Web-Server: Auf dem Web-Server sollten beispielsweise Dinge wie der HTTP Request und Statuscode gespeichert werden



App-Server: Auf dem App-Server sollten Dinge wie Exceptions, Zeiten, Laden von Dateien etc. geloggt werden

Was steht in Ihrem
Logfile für den App
Server?

7. Aus Geschwindigkeitsgründen halten Sie teure DB-Connections auf Vorrat in einem Pool. Jeder Request vom Client braucht dann eine Connection. Der Pool hat die Methoden:

```
DB Connection con = ConnectionPool.getConnection();  
ConnectionPool.freeConnection( DBConnection dbCon);
```

Was loggen Sie in Ihrem App Server? Oder anders gefragt: Was wollen Sie beim Umgang mit dem Pool als Software-Architektin wissen?

Zum einen möchte man wissen, welche Verbindungen aktuell frei sind, bzw. welche zur Verwendung zur Verfügung stehen. Zum anderen wäre es auch sinnvoll zu loggen, WER, WANN, auf WELCHE Connection zugegriffen hat.