

14.–17. 09. 2009
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Meld' dich mal wieder

Logging aber richtig

Jens Schauder

LINEAS Informationstechnik GmbH



Fragen:

Wer nutzt ...

Logging?

ein **Framework?**

Mehrere Logging
Hierarchien?

MDC?

Antworten!

Warum dieser
Vortrag?



Photo: <http://blog.schauderhaft.de>

Debug-Logs

```
System.out.println(„In getRows #4“)
```

```
log.debug(„In getRows #4“)
```

Frameworks

Log4J

JUL

java.util.logging

"Whatever JUL can do, Log4j can also do
- and more."

<http://java.sys-con.com/node/48541>

Commons Logging

SIf4J

Eigenbau



Rock Solid

Photo: <http://www.sxc.hu/profile/timobalk>

Exception Logs

```
Exception ex = new SomethingsWrongException();  
log.error(ex);  
throw ex;
```

```
} catch (SomethingsWrong ex){  
    log.error(ex);  
    throw new SomethingsWrongBelow(ex);  
}
```



```
} catch (SomethingsWrong ex){  
    if (retry < 10){  
        logger.warn(„retrying after Exception“, ex);  
        retry();  
    } else {  
        logger.error(ex);  
        exit();  
    }  
}
```

```
} catch (SomethingsWrong ex){}
```

→ **Design**

sauberes Exceptionhandling

Was loggen?

Root Cause

die erste Exception

Crown Effect

Anwendersicht

Identifikation



Schwierig!



Einfach!

MDC

Mapped Diagnostic Context

```
MDC.put(„orderid“, orderId);  
MDC.put(„process“, „cancel order“);
```

```
%X{process} %X{orderid} %
```

→ **Design**
saubere Schichten

Log Level

TRACE

Entry/Exit
gar nicht

DEBUG

Kritische Stellen

Wichtige Entscheidungen

INFO

Wichtige Infos

keine Aktion notwendig

Version

Zustand

WARN

verdächtig
nicht kritisch
Retries

ERROR

Aktion notwendig
kein Totalverlust
($<1/\text{Tag}$)

FATAL

Totalverlust
Vollständig unerwartet
Anwendung beenden
($<1/\text{Tag}$)

80%

Performance Logs

Remote Calls

RMI

JDBC

Webservices

Messaging

Lange Prozesse

Suchen

Sortieren

Dateizugriff

Schichtwechsel

Presentation
Domain Logic
Persistence
Interfaces

Trennung von
Exceptions?

Logger
Hierarchien

Klassennamen

de.schauderhaft.domain.Order

Prefixe

TIME.de.schauderhaft.domain

RATE.de.schauderhaft.domain

AUDIT.de.schauderhaft.domain

Level vs Prefix

TIME

DEBUG = Schichtwechsel

INFO = Lange Prozesse

WARN = Remote Calls

Eskalation

(TIME.*, DEBUG) -> (*, WARN)

Wahnsinn!?

Packages

de.schauderhaft.

<module>.

<layer>.

*

AOP

→ **Design**
saubere Struktur

RATE vs TIME

AUDIT

empfangen

gesendet

bestätigt

geantwoordet

Details

Konstanten

I18N?

Timestamp

YYYYMMDD HH24:MI:SS.NNN

Parsebar

Achtung!

Performance


```
if(logger.isDebugEnabled()) {  
    logger.debug("Entry number: " + i + " is " +  
        String.valueOf(entry[i]));  
}
```

Datenschutz



Photo: <http://www.sxc.hu/profile/porah>

Meta Logga

<http://code.google.com/p/metalogga/>

Antworten?



Photo: <http://www.sxc.hu/profile/CraigPJ>

14.–17. 09. 2009
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Vielen Dank!

Jens Schauder

LINEAS Informationstechnik GmbH