



WEITER WISSEN →

Ziel

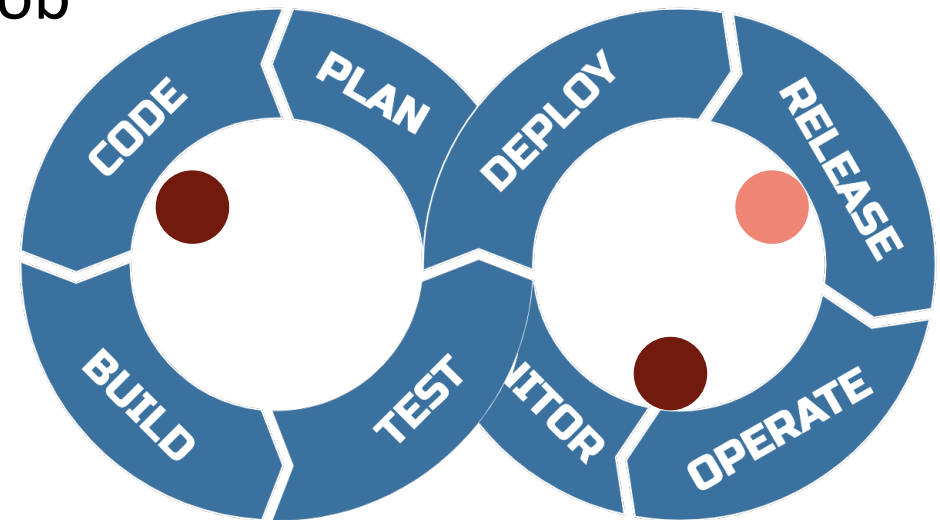
Nach der Lektion haben die Studierenden das bestehende spring-starter Projekt mit einem Logging-Framework kombiniert und können dessen Einsatz planen und vertreten.

Logs

- Spring Framework
- Abhängigkeiten
- Factory Pattern
- Deployment geplant
- Applikation verpackt und bereit

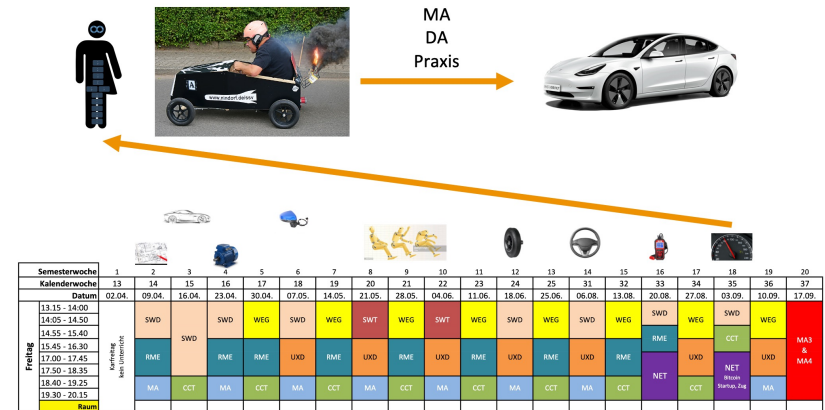
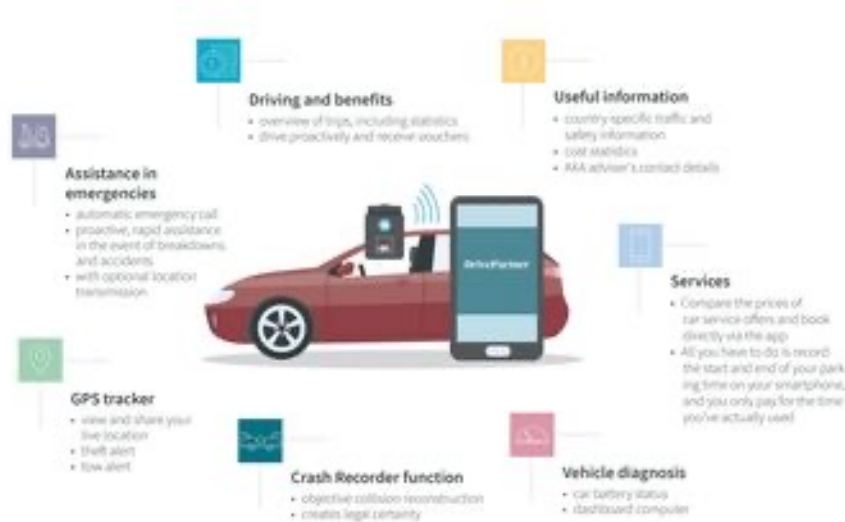
Einmal ausgeliefert, wie wissen wir ob und was schiefgeht?

- Was sind Logs?
- Wie kommen wir da ran?
- In Echtzeit?
- Wohin gehen die?



Agenda

- Ziel
- Übersicht Logging
- SLF4J Logger in Spring
- Bite



GPS tracker

- view and share your live location
- theft alert
- tow alert



Crash Recorder function

- objective collision reconstruction
- creates legal certainty



Vehicle diagnosis

- car battery status
- dashboard computer

dire
• All y
the
ing
and
you

Warum?

- Je komplexer die Applikation, je schwieriger die «Observation»
- Was läuft?
- Was läuft falsch/nicht?
- Enthalten
 - Fehler
 - Informationen zu Events
 - Warnungen
- Jede anständige Anwendung schreibt einigermaßen anständige Logs
- Ersetzen den Entwickelnden **nicht** den Debugger

Wohin?

- Oft: Konsole
 - Lokal
 - Deployment – Shipped
- Zentrale Log-Server
 - SaaS
 - Indexieren
- Auch: Dateien
 - tail -f
 - Rollend wechseln und vllt. archivieren
- Wichtig ist
 - Zugang haben
 - Handhabung lernen

Warum nicht immer Konsole?

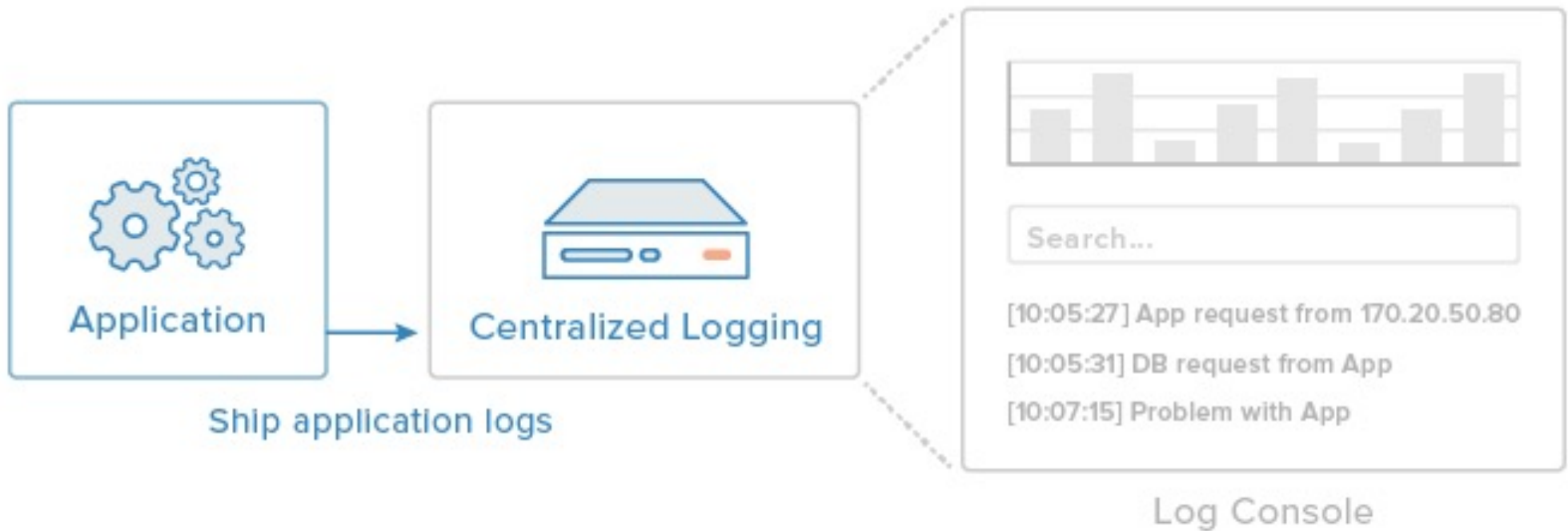
- **Neustart löscht alles!**
- Kein Format, nicht parseable
- In Produktion z.T./u.U. deaktiviert
- kein «Log Shipping»
- Performance Probleme weil oft synchron
- Wenig bis nicht konfigurierbar
- Keine Log Levels, kein Filter

Wir brauchen als ein erwachsenes Logframework.

Wohin?

- Oft: Konsole
 - Lokal
 - Deployment – Shipped
- Zentrale Log-Server
 - SaaS
 - Indexieren
- Auch: Dateien
 - tail -f
 - Rollend wechseln und vllt. archivieren
- Wichtig ist
 - Zugang haben
 - Handhabung lernen

Centralized Logging



```
14 @PostConstruct
15 public void init() {
16     LOG.trace("Starting procedure to add a new expedition.");
17     LOG.debug("Adding expedition with name = {}", "ExPediTiOnName");
18     LOG.trace("Creating the expedition using the {} factory", "ArcticExpedition.class");
19     LOG.info("Expedition {} created", 12345);
20     LOG.debug("Adding {} scientists to the expedition {}", Arrays.asList("fred", "george"), 12345);
21     LOG.warn("{} is not qualified to take part in {}", "fred", 12345);
22     LOG.error("Adding expedition failed", new RuntimeException("UNQUALIFIED_PARTICIPANT"));
23 }
```

StarterApplication

```
2021-07-30 15:34:30.150 INFO 12189 --- [main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2021-07-30 15:34:30.151 INFO 12189 --- [main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.48]
2021-07-30 15:34:30.261 INFO 12189 --- [main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2021-07-30 15:34:30.262 INFO 12189 --- [main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1139 ms
2021-07-30 15:34:30.398 INFO 12189 --- [main] c.a.n.s.swdt.starter.logging.LogWarmer : Expedition 12345 created
2021-07-30 15:34:30.399 WARN 12189 --- [main] c.a.n.s.swdt.starter.logging.LogWarmer : fred is not qualified to take part in 12345
2021-07-30 15:34:30.403 ERROR 12189 --- [main] c.a.n.s.swdt.starter.logging.LogWarmer : Adding expedition failed

java.lang.RuntimeException: UNQUALIFIED_PARTICIPANT
    at ch.abtts.nds.swe.swdt.starter.logging.LogWarmer.init(LogWarmer.java:22) ~[main/:na] <4 internal lines>
    at org.springframework.beans.factory.annotation.InitDestroyAnnotationBeanPostProcessor$LifecycleElement.invoke(InitDestroyAnnotationBeanPostProcessor.java:389)
    at org.springframework.beans.factory.annotation.InitDestroyAnnotationBeanPostProcessor$LifecycleMetadata.invokeInitMethods(InitDestroyAnnotationBeanPostProcessor.java:361)
    at org.springframework.beans.factory.annotation.InitDestroyAnnotationBeanPostProcessor.postProcessBeforeInitialization(InitDestroyAnnotationBeanPostProcessor.java:136)
    at org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.applyBeanPostProcessorsBeforeInitialization(AbstractAutowireCapableBeanFactory.java:1454)
```

```
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;
```

```
private static final Logger LOG = LoggerFactory.getLogger(LogWarmer.class);
```

Lektion 25

Factory Pattern

Doch wo ist der Rest?

```
LOG.trace("Starting procedure to add a new expedition.");  
LOG.debug("Adding expedition with name = {}. ", "ExPediTiOnName");
```

- Log Level ist standardmässig INFO!
- Reicht für Produktion längstens
- Level
 - Trace: Spuren, fast nie benutzt, minimalste Veränderungen
 - Debug: Wie gesagt, im Falle einer Fehlersuche hilfreich
 - In Produktion Debug Logs schreiben kostet unglaublich viel Ressourcen
 - Info: Ab hier können wir das Verhalten beobachten
 - Warn: Es läuft zwar (noch) aber es ging/geht was ein bisschen schief
 - Error: Fehler, ein Mensch sollte sich das anschauen
 - (Fatal): ± Applikation wird wahrscheinlich zwangsbeendet oder stürzt ab, Daten gehen oder gingen verloren

Spring oder Framework

```
# application.properties values
logging.level.root=WARN
logging.level.org.springframework=ERROR
logging.level.com.quickprogrammingtips=DEBUG
```



```
logging.level.root=TRACE
```

```
2021-07-30 15:52:48.655 TRACE 12552 --- [main] .s.c.a.CommonAnnotationBeanPostProcessor : Invoking init method on bean 'logWarmer': public void ch.abtts.nds.swe.swdt.starter.logging
2021-07-30 15:52:48.655 TRACE 12552 --- [main] c.a.n.s.swdt.starter.logging.LogWarmer : Starting procedure to add a new expedition.
2021-07-30 15:52:48.655 DEBUG 12552 --- [main] c.a.n.s.swdt.starter.logging.LogWarmer : Adding expedition with name = ExPediTiOnName.
2021-07-30 15:52:48.657 TRACE 12552 --- [main] c.a.n.s.swdt.starter.logging.LogWarmer : Creating the expedition using the ArcticExpedition.class factory
2021-07-30 15:52:48.657 INFO 12552 --- [main] c.a.n.s.swdt.starter.logging.LogWarmer : Expedition 12345 created
2021-07-30 15:52:48.657 DEBUG 12552 --- [main] c.a.n.s.swdt.starter.logging.LogWarmer : Adding [fred, george] scientists to the expedition 12345
2021-07-30 15:52:48.657 WARN 12552 --- [main] c.a.n.s.swdt.starter.logging.LogWarmer : fred is not qualified to take part in 12345
2021-07-30 15:52:48.657 ERROR 12552 --- [main] c.a.n.s.swdt.starter.logging.LogWarmer : Adding expedition failed
```


Ressourcen

- Log schreiben ist nicht billig
- CPU
- Memory
- Disk-Space

- Sollte asynchron erfolgen
- Darf nicht «blocking» sein, sonst 🍷 -neck

Weil das alles nicht einfach ist - richtig! - lassen wir das wen anderen tun!

Frameworks

- Java
 - SLF4J ← In Spring integriert, für uns mehr (viel mehr) als ausreichend
 - Logback
 -  insights
- PHP
 - link
- Javascript (node)
 - link link
 - winston
 - bunyan
- golang
 - log package
- Gibt es auch für FrontEnds aber ist nicht so normal
 - ihr kommt nicht an die Logs meine iPhones...

Logs

Bite

<https://nds-swe.github.io/swdt/docs/tasks/bites/logging>



WEITER WISSEN.

Wir begleiten Sie!