

# Zeitsteuerung von Abläufen in Java auf Basis von Terracotta Quartz

Ralph Löwe

# Gliederung

1. Einleitung
2. Klassische Java-Hilfsmittel
3. Terracotta Quartz
4. Anwendungsobjekte
5. Anwendungsbeispiel
6. Fazit und Ausblick
7. Literatur



# 1. Einleitung

- Zeitsteuerung
  - Abläufe
  - Ereignisse
- Beispiele für Abläufe:
  - Sicherung
  - Archivieren
  - Löschen
  - Export



# 1. Einleitung

- Typische Werkzeuge
  - cron
  - Torque
  - UC4
  - ...
- Problem: Schnittstellen für Java
- Alternative: Terracotta Quartz

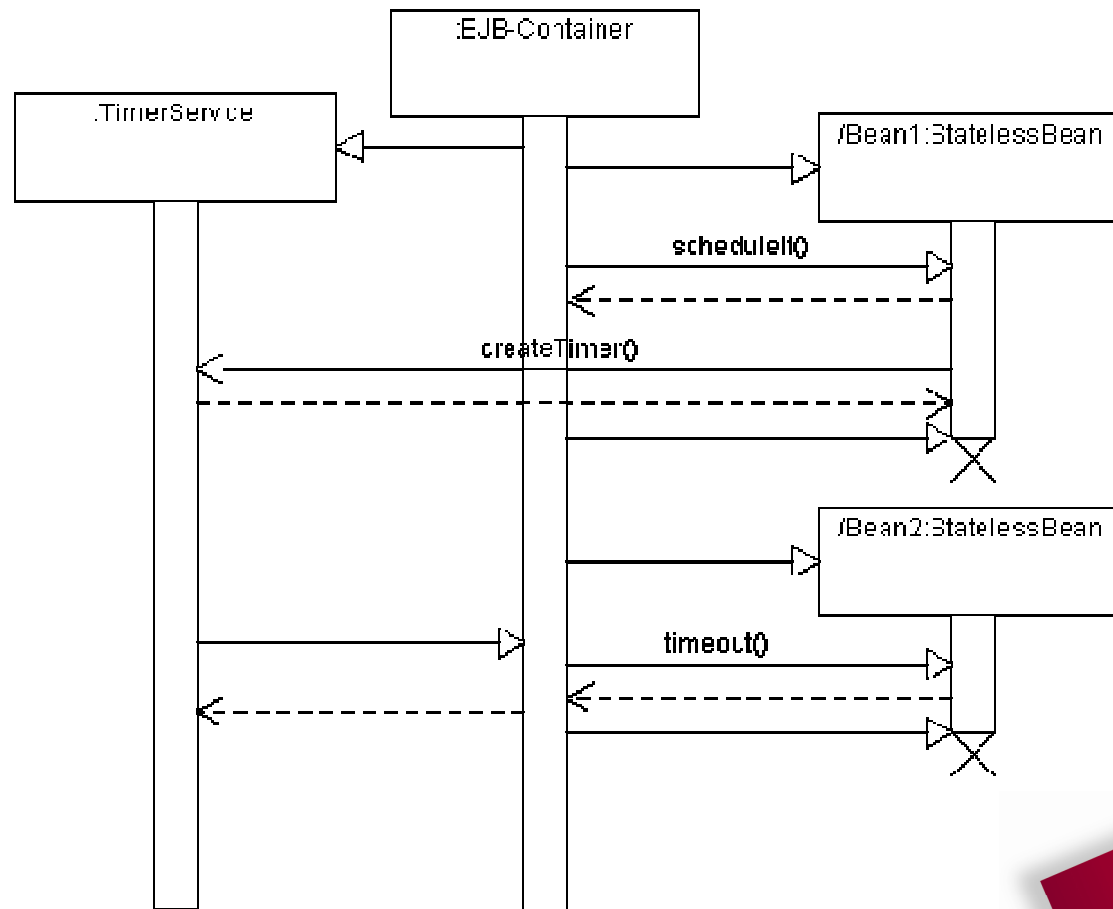


## 2. Klassische Java-Hilfsmittel

- Java SE: Individuallösung
- Java EE:
  - TimerService in EJB 2.1 und EJB 3.0
  - Erweitert in EJB 3.1
  - Probleme:
    - Keine starke Trennung von Ereignis und Ablauf
    - Verhalten im Cluster
    - Kein zentrales Sperren von Abläufen



## 2. Beispiel in EJB 3.0



EJB 3.0 mit TimerService (vereinfacht)

## 2. Beispiel in EJB 3.0

```
@Stateless
public class StatelessBean implements StatelessRemote
{
    @Resource
    private SessionContext context;

    public String scheduleIt(String message) {
        TimerService service = context.getTimerService();
        service.createTimer(1000, message);
        return message;
    }

    // ...
}
```



## 2. Beispiel in EJB 3.0

```
// ...  
  
@Timeout  
public void timeout(Timer timer) {  
    String message = (String)timer.getInfo();  
    System.out.println("Timer executed: " + message);  
}  
}
```



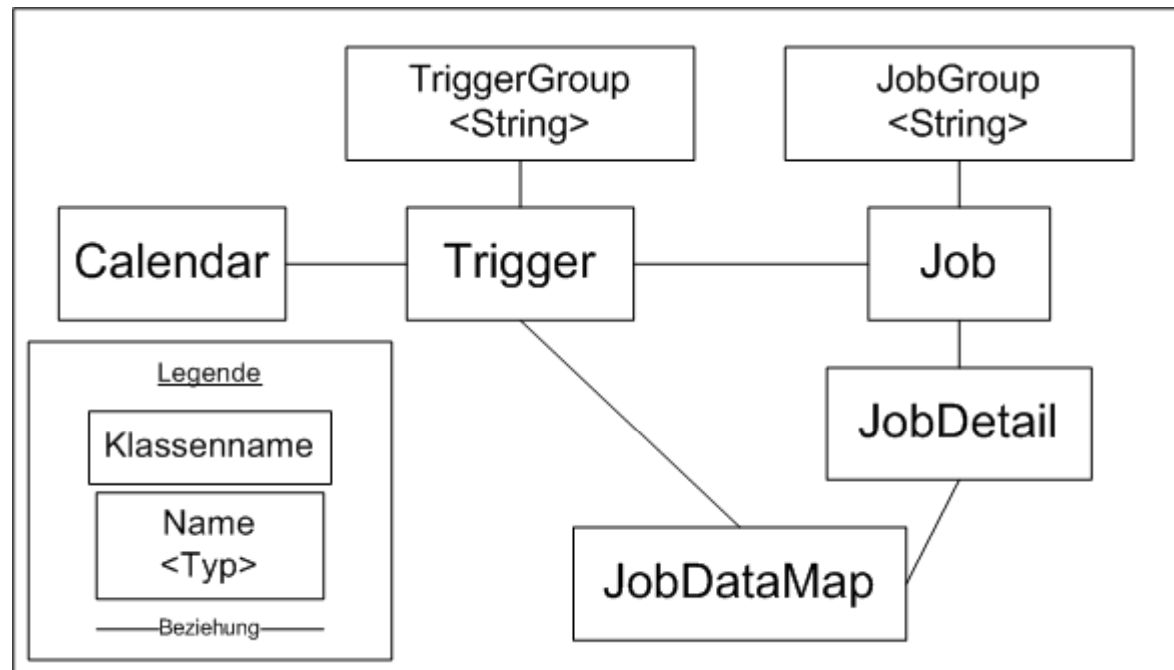


# 3. Terracotta Quartz

- Quelloffenes Projekt
- Ursprünglich: Opensymphony
- Ab Ende 2009: Terracotta
- Funktionen
  - Wartungsfenster
  - Trennung von Ereignis und Ablauf
  - Clusterfähigkeit
- Java EE: EJB TimerService



# 4. Anwendungsobjekte



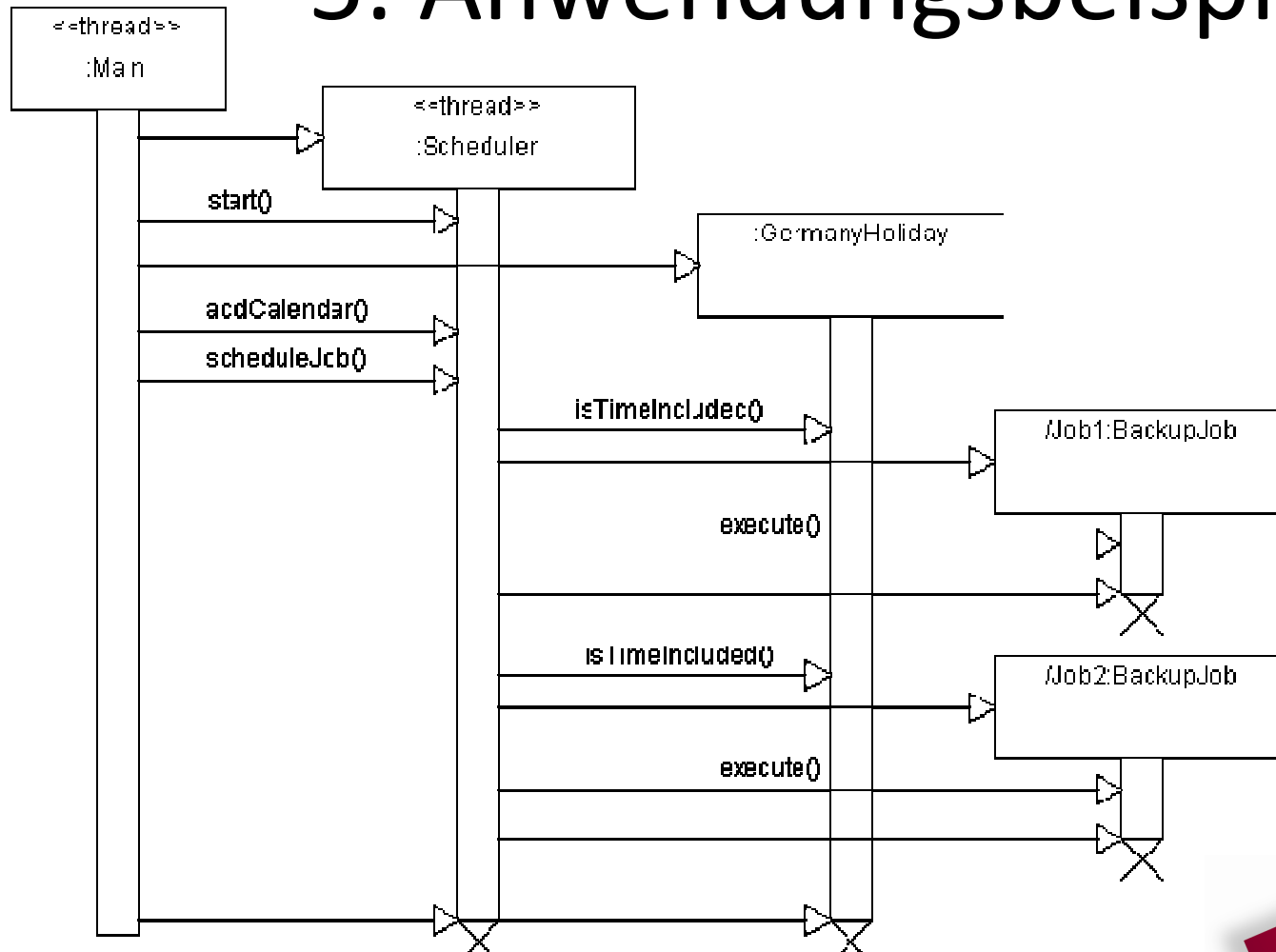
- InterruptableJob, StatefulJob
- SimpleTrigger, CronTrigger

# 5. Anwendungsbeispiel

- Backup
- Täglich um 22 Uhr
- Nicht an Sonn- und **Feiertage**



# 5. Anwendungsbeispiel



Quartz Scheduler, Anwendungsbeispiel (vereinfacht)

# 5. Anwendungsbeispiel

```
public class BackupJob implements Job {  
    // Der JobExecutionContext beinhaltet JobDataMap  
    // und JobDetail.  
    public void execute(JobExecutionContext context) {  
        Calendar fireTime = Calendar.getInstance();  
        fireTime.setTime(context.getFireTime());  
        System.out.println("executing backup: " +  
            fireTime.get(Calendar.SECOND));  
        // Hier kann die Logik der Sicherung  
        // implementiert werden.  
    }  
}
```



# 5. Anwendungsbeispiel

```
public class GermanyHoliday extends BaseCalendar
    implements org.quartz.Calendar {
    public boolean isTimeIncluded(long l) {
        Calendar cal = Calendar.getInstance();
        cal.setTimeInMillis(l);
        // Heute ist Tag der Deutschen Einheit.
        if (cal.get(Calendar.DAY_OF_MONTH) == 3 &&
            cal.get(Calendar.MONTH) == Calendar.OCTOBER)
            return false;
        // ... // Ansonsten
        else
            return true;
    }
}
```



# 5. Anwendungsbeispiel

```
Scheduler s = StdSchedulerFactory.getDefaultScheduler();  
s.start();
```

```
GermanyHoliday holiday = new GermanyHoliday();  
JobDetail detail =  
    new JobDetail("BackupJob", null, BackupJob.class);  
Trigger trigger = TriggerUtils.makeDailyTrigger(22, 0);
```

```
s.addCalendar("GermanyHoliday", holiday, true, true);  
trigger.setCalendarName("GermanyHoliday");  
s.scheduleJob(detail, trigger);
```

```
s.shutdown();
```



# 6. Fazit und Ausblick

- Vorteile von Terracotta Quartz
  - Mehr Zeitsteuerung als bisher in Java
  - Bessere Trennung von Ereignis und Ablauf
  - Anwendungsübergreifende Kalender
  - Wartungsfenster
- Probleme:
  - Verknüpfung der Objekte mittels Zeichenketten
  - Fehlerdiagnose
- Enterprise Java Beans 3.1 ff.





# 7. Literatur / Diskussion

- Cavaness, C., 2006. *Quartz Job Scheduling Framework: Building Open Source Enterprise Applications*, Prentice Hall. Verfügbar unter: <http://www.amazon.com/Quartz-Job-Scheduling-Framework-Applications/dp/0131886703>
- Gamma, E. et al., 2004. *Entwurfsmuster*, Übersetzer: D. Riehle, Addison-Wesley
- Java Community Process (JCP), <http://jcp.org>
- Opensymphony, <http://www.opensymphony.com>
- Quartz, <http://www.quartz-scheduler.org>
- Terracotta Inc., <http://www.terracotta.org>

