

**adaptTo()**

APACHE SLING & FRIENDS TECH MEETUP  
10-12 SEPTEMBER 2018

# System Ready framework

Christian Schneider - Adobe, Apache  
Andrei Dulvac - Adobe, Apache

# The speakers



- Christian Schneider
  - Adobe Switzerland
  - Apache
  - @schneider\_chris 



- Andrei Dulvac
  - Adobe Switzerland
  - Apache
  - @adulvac 



# When is an *OSGi application* ready? (1)

- Why do you need to know?
  - When can automated tests start accessing your system (initial startup)
  - When is your application ready to receive traffic? (zero downtime deployments)



# When is an *OSGi application* ready? (2)

- What could *ready* mean?
  - Particular OSGi services are active
  - Oak initial indexing is done/ search is available
  - Critical async startup jobs are finished (e.g. LDAP users migration)
  - For complex applications – need **app/deployment -specific checks**



# Use cases

- Kubernetes
  - Provides HTTP readiness and liveness checks
- PaxExam
  - Waiting for a system in integration tests
  - Root-cause analysis – why did a system not start?
- End-to-end automated tests
  - Waiting for the application to be ready before starting the tests
- Standalone deployments
  - Initial startup
  - Runtime readiness



# Felix systemready

- 49KB OSGi bundle in Felix, framework-agnostic, very little dependencies
- Ready checks
  - Framework started
  - List of services present
  - List of DS components present
  - Extensible by application-specific checks (SPI)
- Aggregated system status for liveness and readiness - **GREEN**/ **YELLOW**/ **RED**
- Built-in HTTP servlets – 200 for **GREEN**, 503 otherwise
- Root-cause analysis for missing components

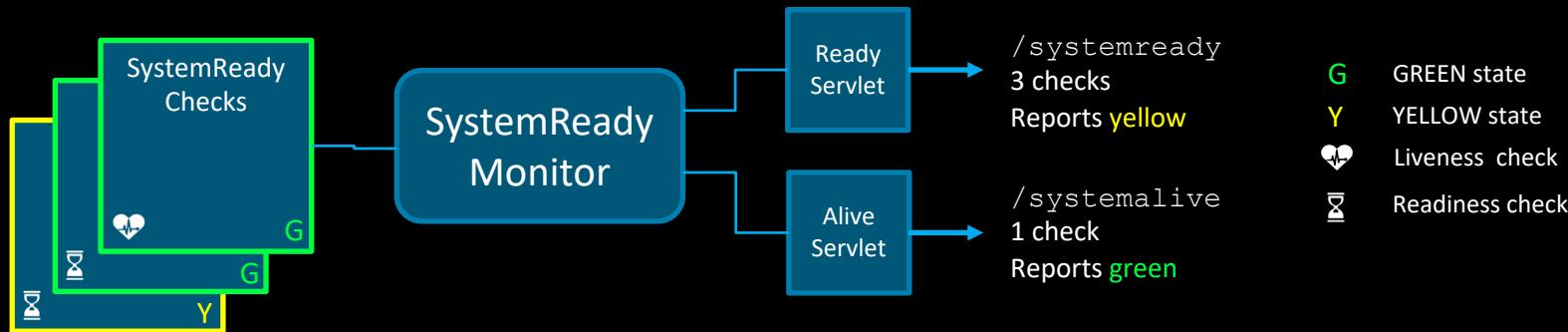


# How the default servlets output looks like

HTTP RESPONSE CODE: 503

```
{"systemStatus": "YELLOW",
"checks": [
  {
    "check": "Framework Start Ready Check",
    "status": "GREEN",
    "details": "Framework started. Start level: 30"
  },
  {
    "check": "My custom check",
    "status": "YELLOW",
    "details": "I still have 2/5 startup whatchamacallits to do"
  }
}
```

# Felix systemready architecture





# How to use systemready

- Add it to your sling application (not included in the launchpad)
- Shipped OOTB with AEM 6.5 (currently in development)
- Automated integration tests (e.g. PaxExam)
- But also any OSGi deployment



Now let's blow up some kubernetes pods

# Demo layout

- Kubernetes – OSGi web microservice example
- AEM jar deployment



# Q&A + Resources



- Resources
  - <https://github.com/apache/felix/tree/trunk/systemready>
  - <https://github.com/cschneider/osgi-example-systemready>