



**adaptTo()**

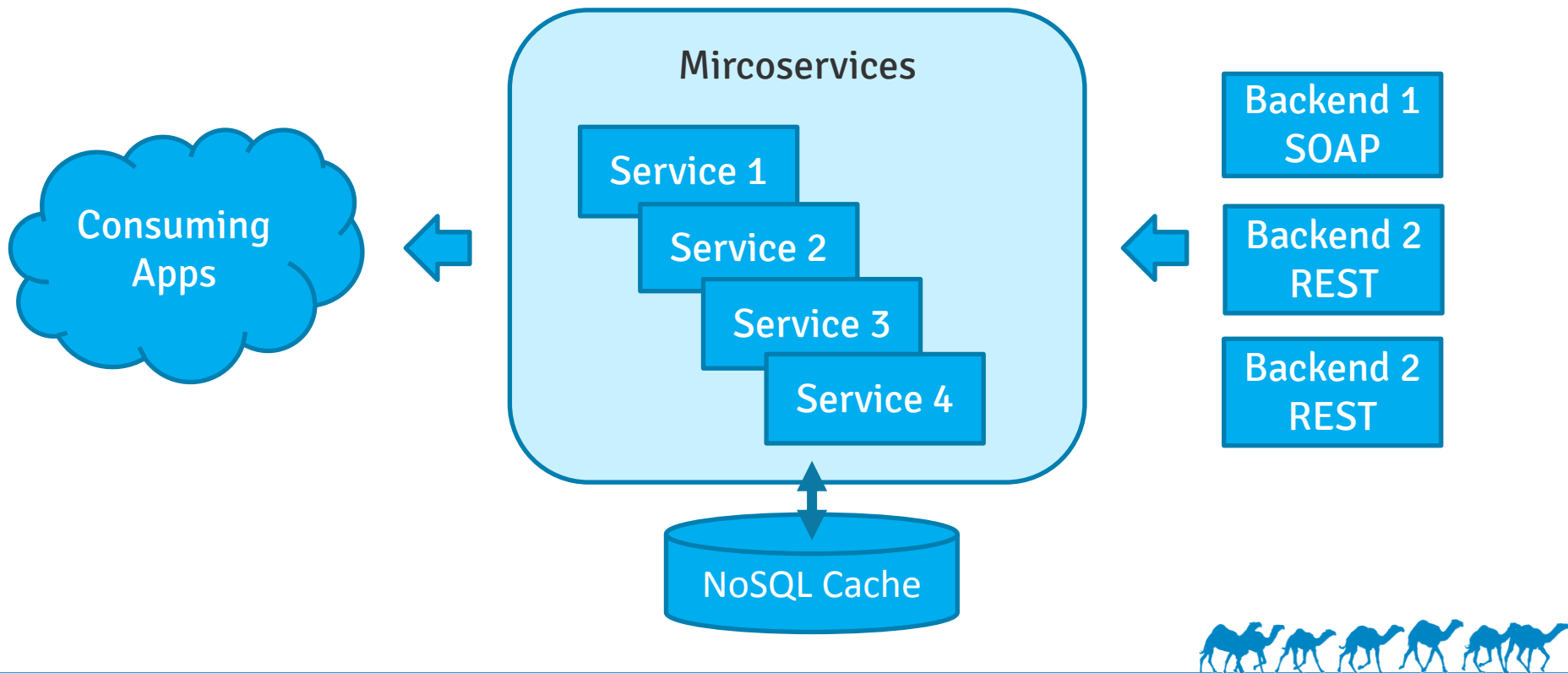
APACHE SLING & FRIENDS TECH MEETUP  
BERLIN, 28-30 SEPTEMBER 2015



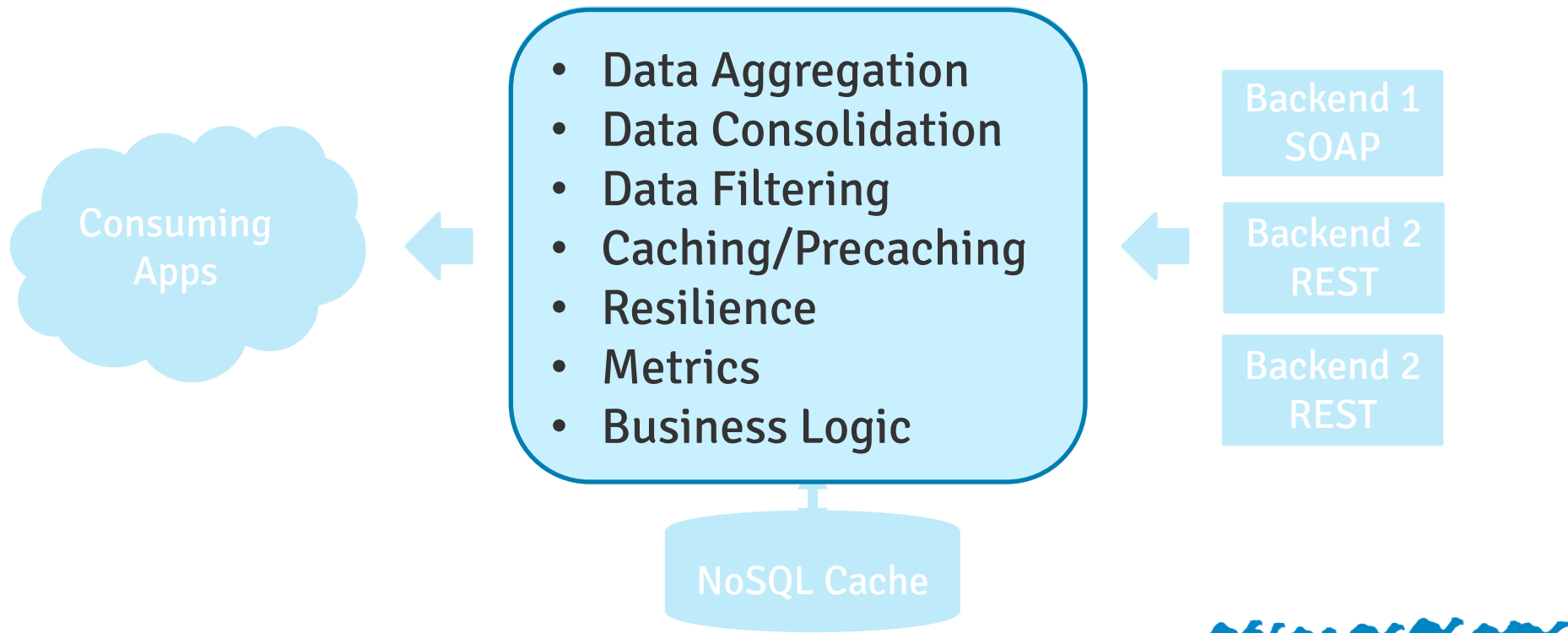
**wcm.io Caravan – OSGi Microservices**

Stefan Seifert, pro!vision GmbH

# Our Microservice Scenario



# Our Microservice Scenario



- Open Source Project (ASL 2.0)
- For RESTful Microservices
- Collection of modular libraries
  - Use only what you need



Couchbase



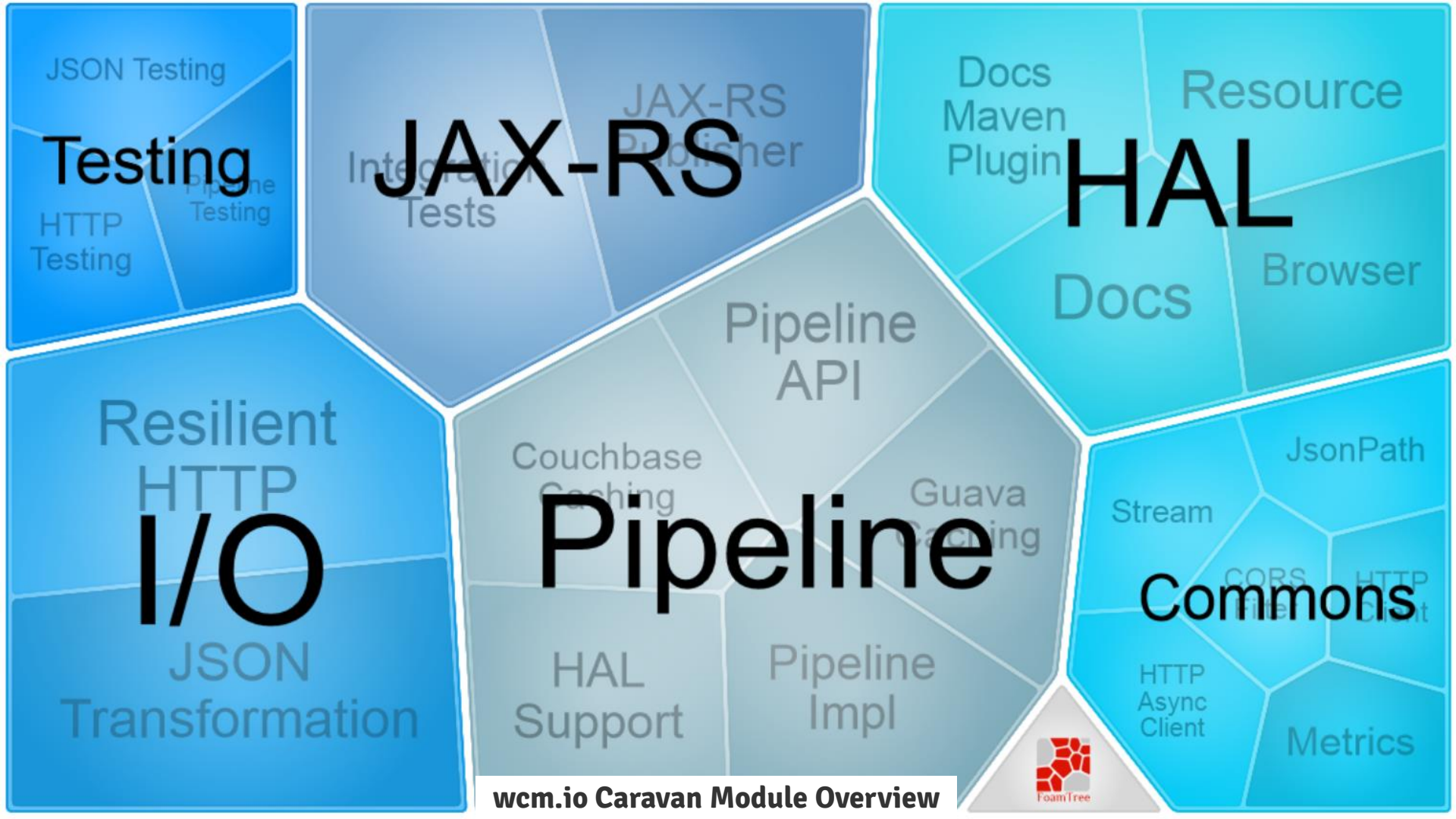
<http://caravan.wcm.io/>



# Microservice Stack based on OSGi

- Based on OSGi and Apache Felix
- Uses some Sling Tooling
- Not using Sling Resource API
- No Repository involved
- Uses only REST/HTTP, no OSGi Remoting






# Resilient HTTP Client

- Based on the “Netflix Stack”
  - Hystrix for Resilience, Circuit Breaker etc.
  - Ribbon for Software Load Balancing
- Asynchronous processing via RxJava
- RFC 6570 URI Templating
- Apache HTTP Client



<http://caravan.wcm.io/io/http/>



- Based on Jersey 
- Publish OSGi components as JAX-RS services
- Separate JAX-RS applications per bundle



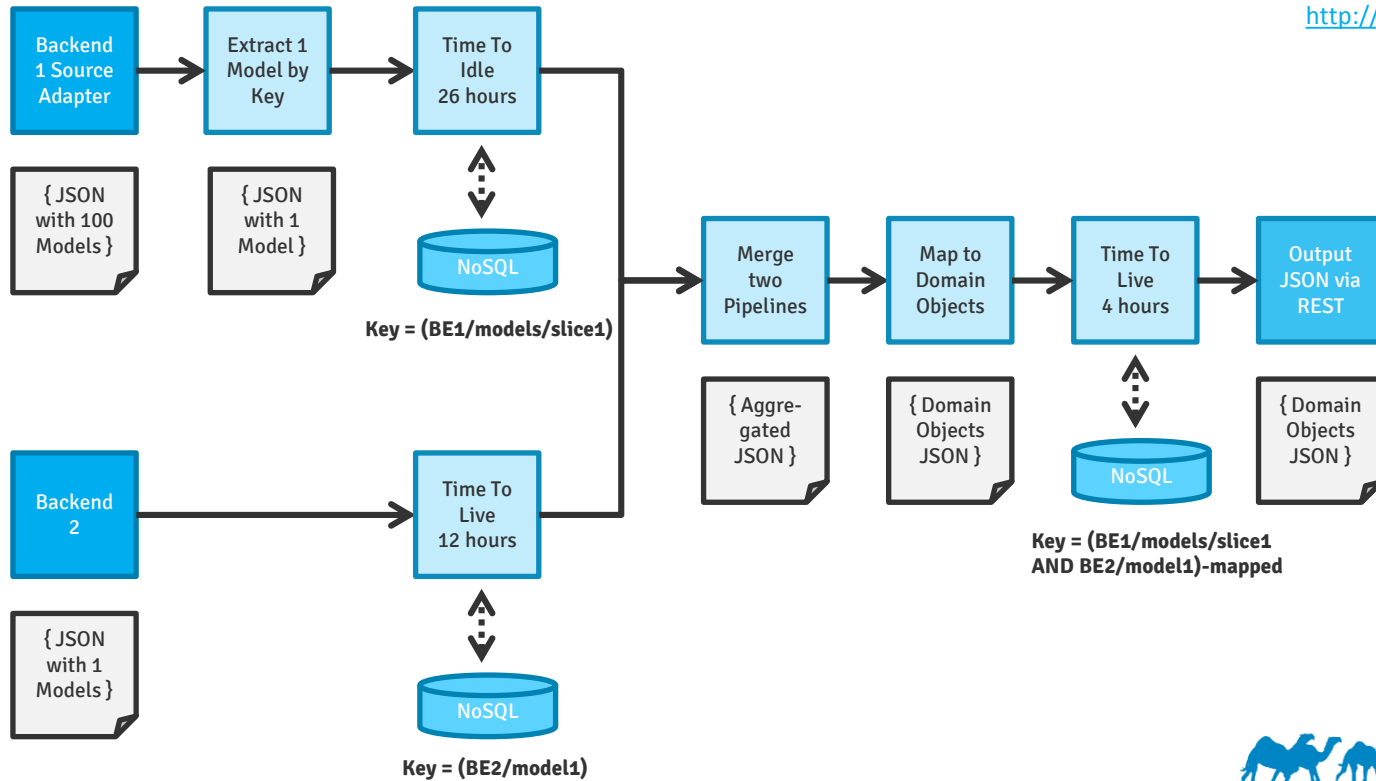


- Hypertext Application Language (HAL)
- HAL Resource Builder
- Integrates HAL Browser
- HAL documentation generated from sources



# JSON Pipelining

<http://caravan.wcm.io/pipeline/>



- HTTP Resilient Client: Stable
- JAX-RS Integration: Stable
- Pipelining, HAL Resource API: Works well, but APIs need to be refactored
- Metrics: First Steps



JSON Testing

Testing

Pipeline Testing

HTTP Testing

JAX-RS  
Integration Tests

JAX-RS  
Publisher

Docs  
Maven  
Plugin

Resource

HAL

Browser

Docs

Resilient

HTTP

I/O

JSON  
Transformation

Pipeline  
API

Couchbase  
Caching

Pipeline

Guava  
Caching

HAL  
Support

Pipeline  
Impl

Stream

JsonPath

Commons

CORS  
Filter

HTTP  
Client

HTTP  
Async  
Client

Metrics

<http://caravan.wcm.io>



FoamTree