WildFly Swarm

Michał Szynkiewicz,
Senior Software Engineer @ Red Hat

Agenda

Microservices

WildFly Swarm and Microservices

Microservices

Communication is terrible, Jeff Bezos

WildFly SWARM

Microservices

„Two Pizza Rule”

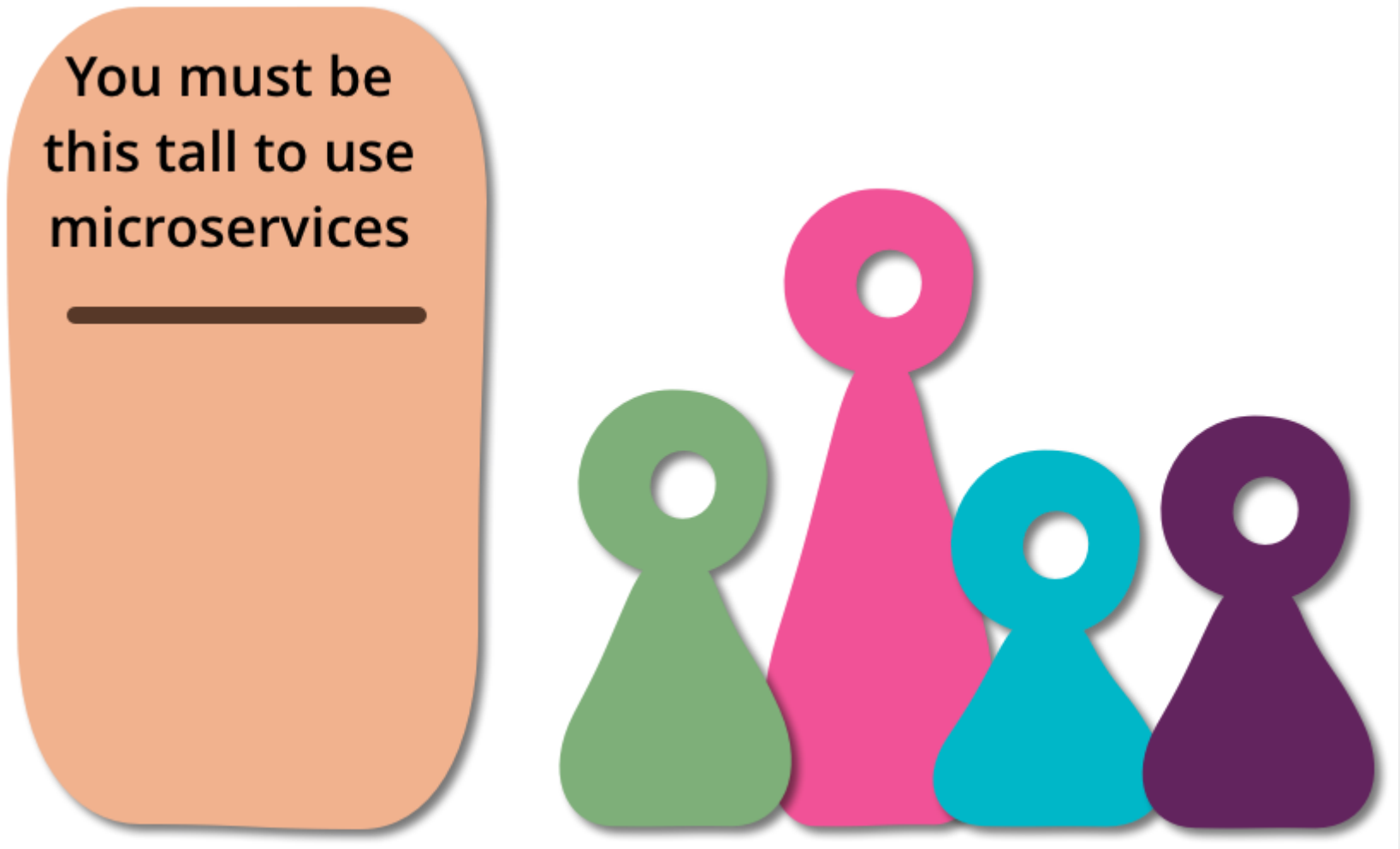
Well defined responsibility

Technology adjusted to the needs

Independent scaling

It’s easier to replace a service

Faster changes



Martin Fowler
https://martinfowler.com/bliki/MicroservicePrerequisites.html

WildFly SWARM

Splitting the system, aka defining service boundaries.

Almost all the successful microservice stories have started with a monolith that got too big and was broken up
Almost all the cases where I've heard of a system that was built as a microservice system from scratch, it has ended up in serious trouble.
Martin Fowler

Configuring service dependencies

More deployments

Logging

Monitoring

More configuration for CI, etc

Microservices baseline (M. Fowler):

Rapid provisioning

Basic Monitoring

Rapid application deployment

Swarm

Java EE API

WildFly constituents split into *fractions*

Pick what you need and package in an uber-jar with your app

Configure in ~~Java~~ or yaml file

Demo, part 1

Beyond Java EE

Logstash

Hystrix

Zipkin

Consul

Swagger

Ribbon

Jolokia

Flyway

Keycloak

Openshift

Teiid

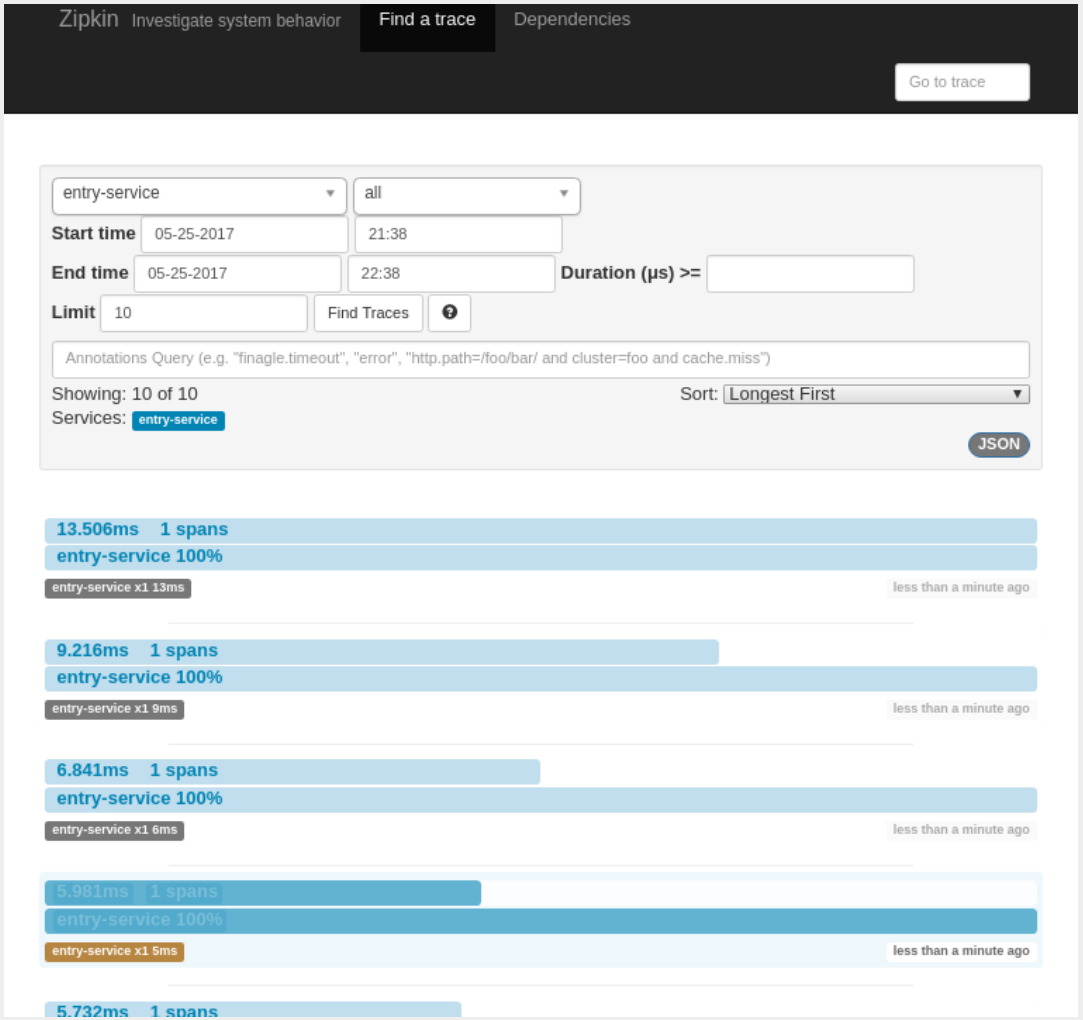
Infinispan

Keycloak/sso

- Authentication server
- Oauth and more
- Single Sign On

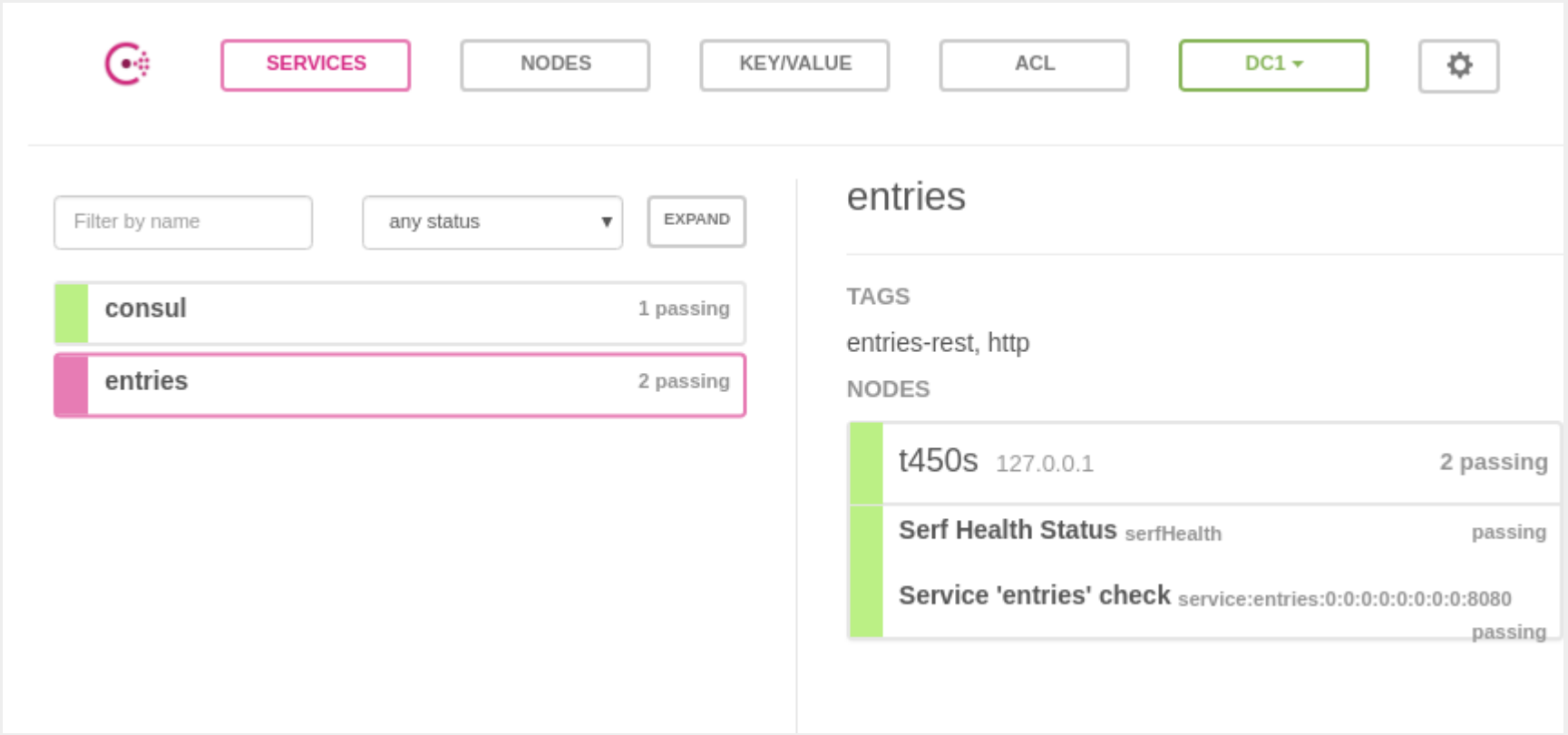
Zipkin

- Distributed tracing
- Optional analysis (requires a Spark job)



Consul

- Service discovery & configuration



OpenShift

- Container platform based on Kubernetes (docker)
- Supported and developed by Red Hat
- For local development, use *minishift* https://www.openshift.org/minishift/

Demo, part 2

Transition enablers

Move your code outside Java EE container right away

Reuse your standalone.xml!

Required fractions can be autodetected, no need to know them before you start

RHOAR: Red Hat OpenShift Application Runtimes

- A number of lightweight application „runtimes”:
  - WildFly Swarm
  - Vert.x
  - Spring Boot
  - Node.js
- Catalogue of OpenShift-enabled *boosters*

Reading, etc

- https://wildfly-swarm.gitbooks.io/wildfly-swarm-users-guide/
- https://github.com/wildfly-swarm/wildfly-swarm-examples
- https://github.com/wildfly-swarm-openshiftio-boosters/
- http://wildfly-swarm.io/generator/
- „Building Microservices”, Sam Newman
- „Enterprise Java Microservices” [MEAP], Ken Finnigan

Questions?

#wildfly-swarm on Freenode