Smallrye Stork

Service discovery and selection like a pro with Stork

@auritamh

Aurea Muñoz 🐦 [auritamh](https://twitter.com/auritamh)

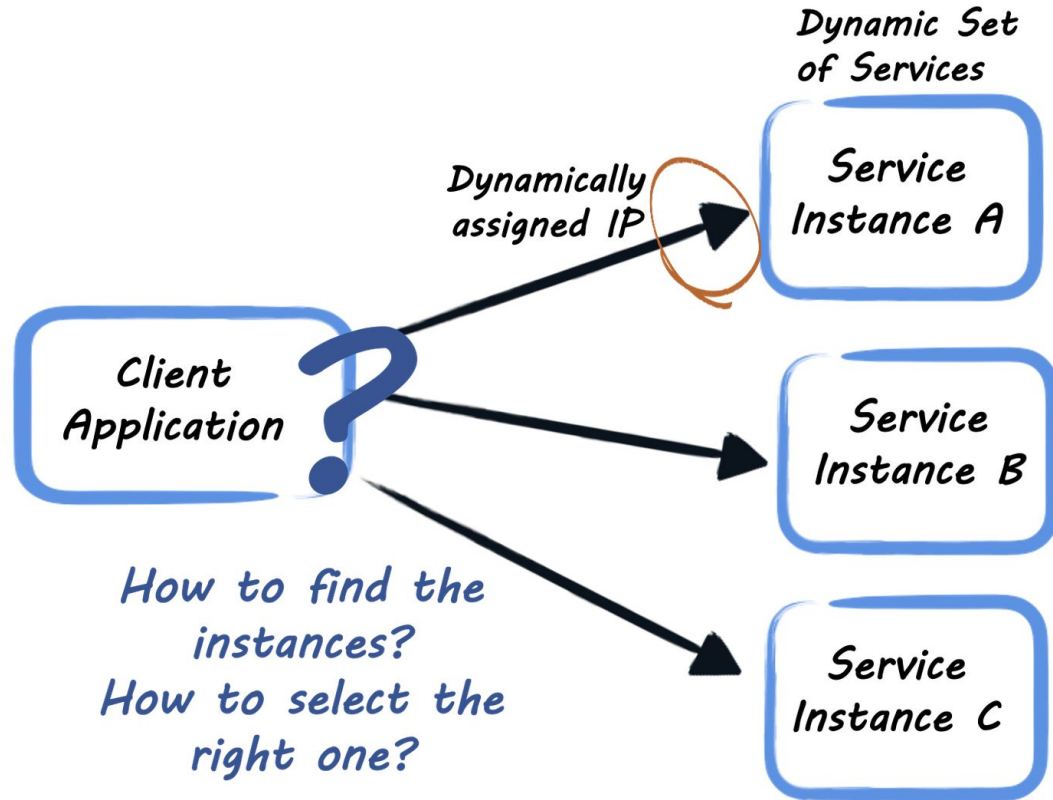Senior Software Engineer @ Red Hat

Madrid JUG leader

Passionate about open source and clean code,
so tune in to learn how to make
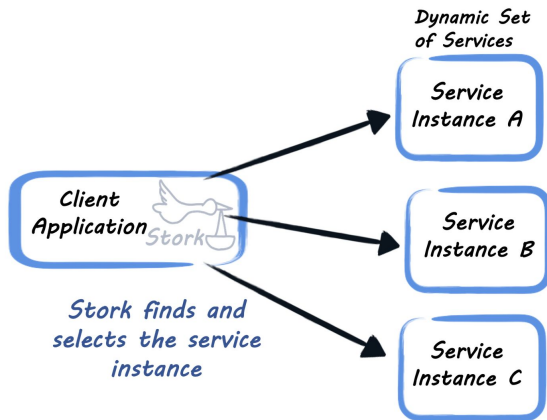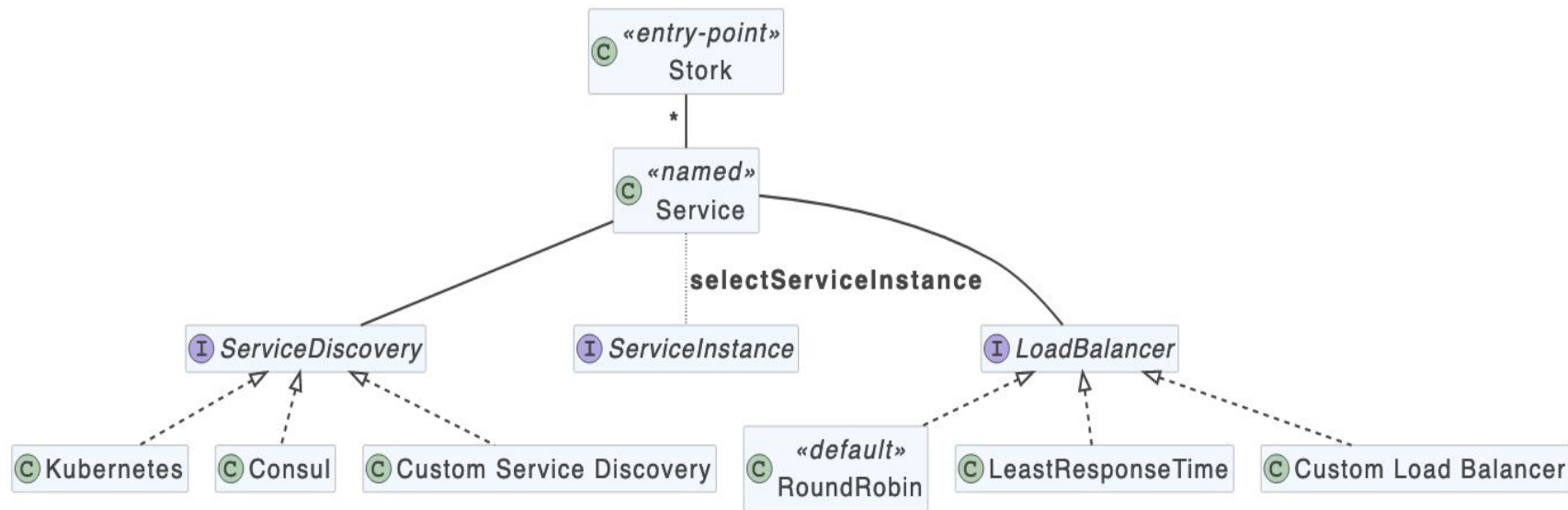the developer experience awesome.

The problem

What is Stork?

SmallRye Stork is a *service discovery and client-side load-balancing framework*.

- A Quarkus integration, but Stork can be used also standalone

- Built-in support for Consul, Kubernetes, Eureka and more.

- Extensible service discovery mechanisms and client load-balancing strategies

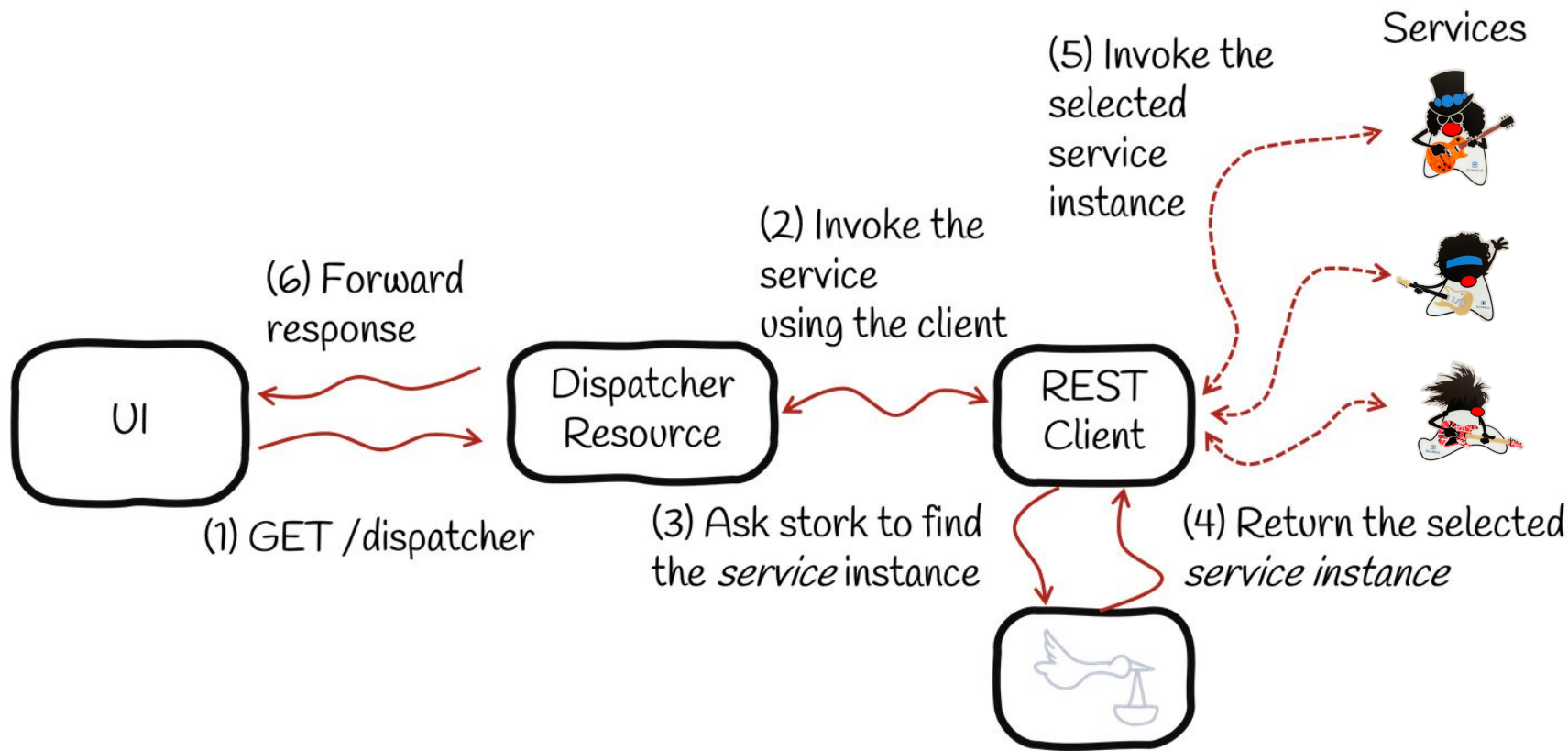- A programmatic API and a *managed* approach

Concepts

Managed approach: Stork and Quarkus

Demo time

The Guitar Hero app

Bonus: more advanced features

- Caching the service instances for service discovery (customizable also)

- Customize service discovery strategy based on your environnement.

```xml
                                                        pom.xml
<dependency>
    <groupId>io.smallrye.stork</groupId>
    <artifactId>stork-service-discovery-static-list</artifactId>
</dependency>
<dependency>
    <groupId>io.smallrye.stork</groupId>
    <artifactId>stork-service-discovery-kubernetes</artifactId>
</dependency>
```

```properties
                                                        application.properties
quarkus.stork.guitar-hero-service.service-discovery.type=static
quarkus.stork.guitar-hero-service.service-discovery.address-list=localhost:${slash.http-port}

%kube.quarkus.stork.guitar-hero-service.service-discovery.type=kubernetes

quarkus.kubernetes.env.vars.QUARKUS_PROFILE=kube
```

https://github.com/aureamunoz/stork-guitar-hero-demo

http://smallrye.io/smallrye-stork/

Thank you