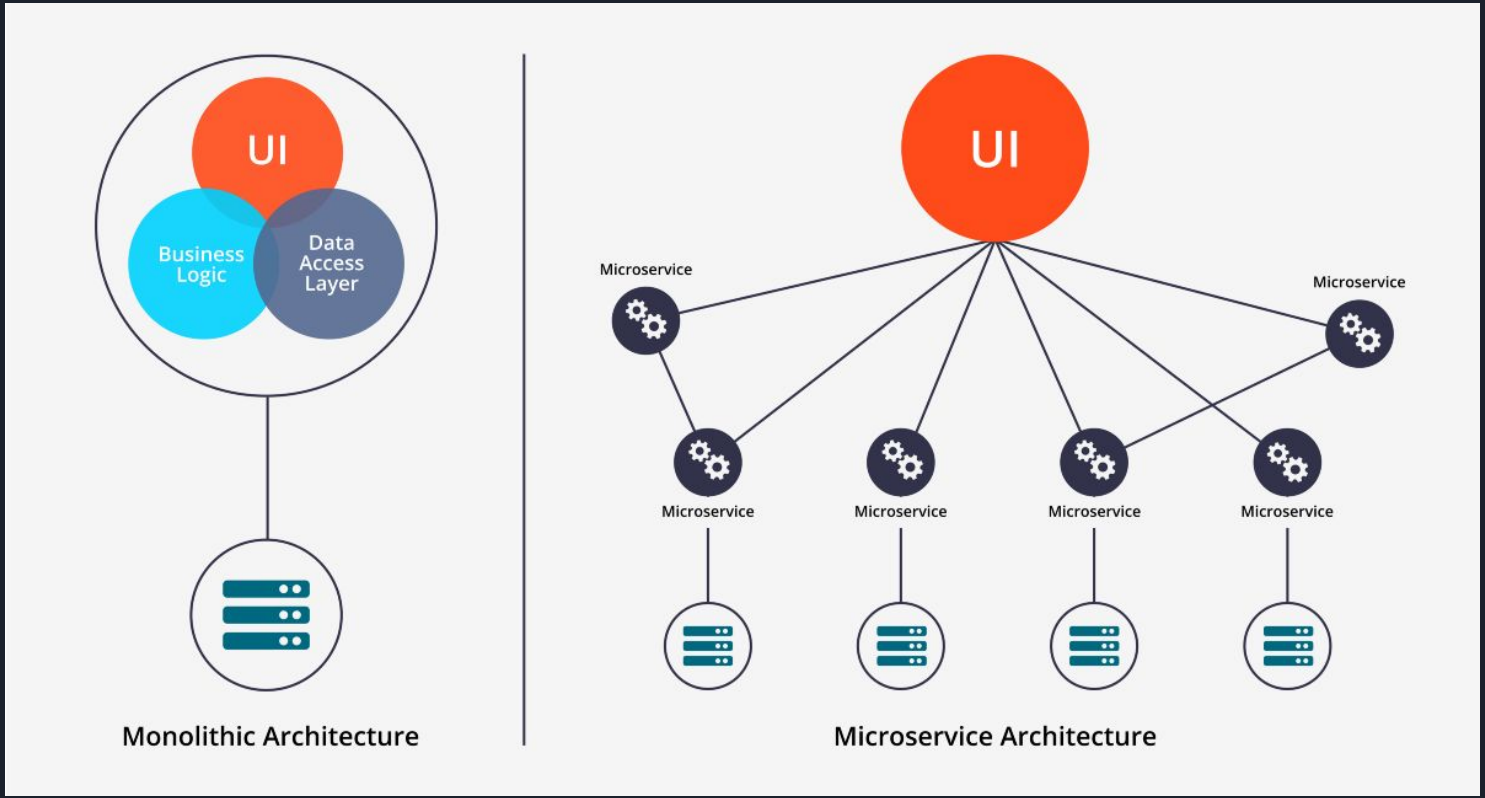
Helidon

Collection of Java libraries for writing microservices.

Otba ZERAMDINI

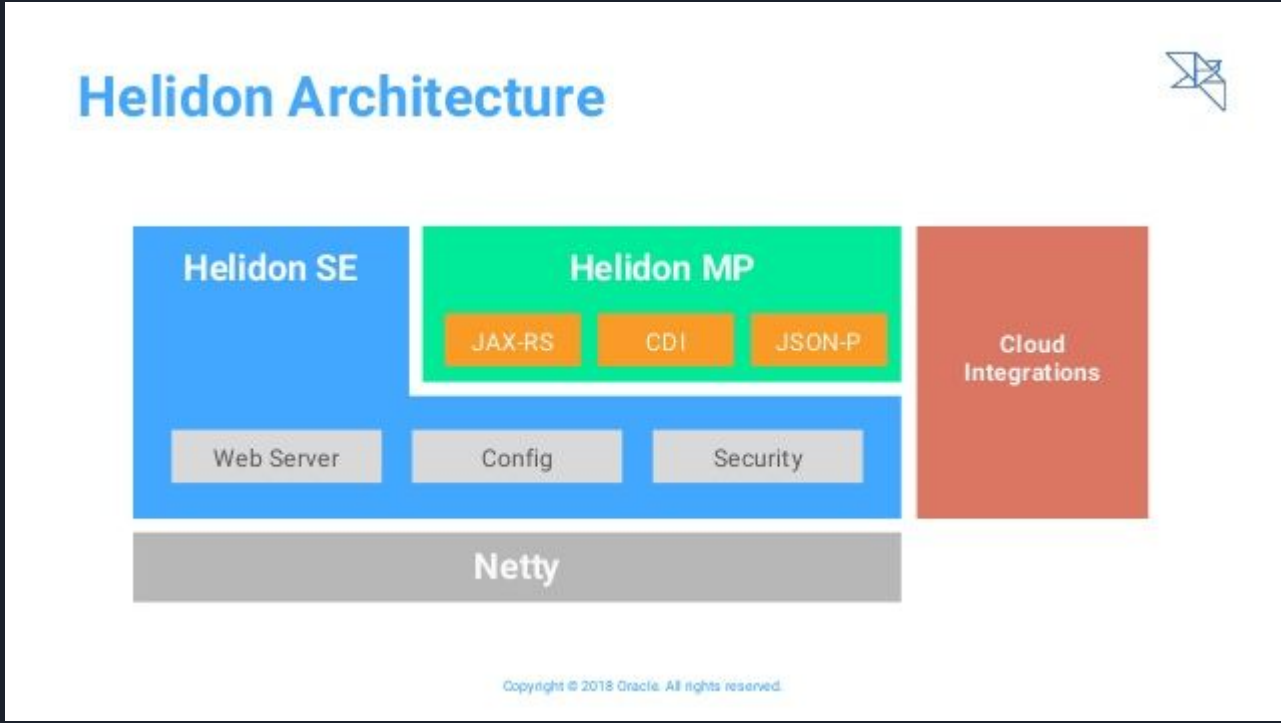
Monolithique vs Microservice



Qu’est ce que Helidon ?

- Helidon est framework microservices open source développé par Oracle.
- Une collection de librairies Java conçue pour créer des applications basées sur les microservices.
- Introduit par Oracle en septembre 2018
- La version actuelle est la version 2.1.0 (Java 8 passage de 1.0 vers 2.0)

Architecture de Helidon



Pourquoi helidon ?

- Conçu spécifiquement pour les microservices.

- Minimaliste et Simpliste (microframework)

- Rapide(Netty web server et support de GraalVM).

- Comporte deux versions :
  - Helidon SE
  - Helidon MP

Helidon SE vs Helidon MP

helidon SE

- Microframework
- Tiny Footprint
- Functional style
- Reactive
- Transparent

helidon MP

- MicroProfile 2.2
- Small Footprint
- Declarative style
- Dependency Injection
- CDI, JAX-RS, JSON-P/B

helidon SE

```
Routing routing =
Routing.builder()
    .get("/hello", (req, res) ->
        res.send("Hello World"))
    .build();

WebServer.create(routing)
    .start();
```

helidon MP

```
@Path("hello")
public class HelloWorld {
    @GET
    public String hello() {
        return "Hello World";
    }
}
```

Evolution de Helidon

Helidon SE 2.0

Build & Tools

- Executable hollow jar
- GraalVM native-image
- Jlink image
- Helidon CLI

Components

- Reactive Web Server
- Health Check
- gRPC Server and Client
- Reactive Streams
- Config
- Metrics
- DB Client
- Reactive Messaging
- Security
- Tracing
- WebClient
- WebSocket

Added in Helidon 2.0

Helidon MP 2.0

Build & Tools

- Executable hollow jar
- GraalVM native-image
- Jlink image
- Helidon CLI
- Jakarta Websocket

Components

- Jakarta RESTful Web Services
- Jakarta JSON Binding
- MicroProfile Metrics
- MicroProfile JWT Auth
- MicroProfile Tracing
- Jakarta CDI
- Jakaarta Persistence & Transactions
- MicroProfile Health Checks
- MicroProfile REST Client
- MicroProfile Reactive Streams Operators
- Jakarta JSON Processing
- MicroProfile Config
- MicroProfile Fault Tolerance
- MicroProfile OpenAPI
- MicroProfile Reactive Messaging

Added in Helidon 2.0

Déploiement d’une application développée avec helidon

- Jar
  - Executable jar
  - Image Docker

- Optimisation Jlink image
  - Stockage uniquement des codes des modules utilisés
  - un temps de démarrage plus rapide et une taille d'image plus petite

- Optimisation GraalVM image native
  - Un temps de démarrage plus rapide et une plus petite consommation de mémoire

Comparaison de frameworks microservices

Criterion	Spring Boot	Vert.x	Helidon	Quarkus
Startzeit	2.226s	0.200s	0.619s	0.562s
First Response	0.190s	0.350s	0.540s	0.523s
RPS Small	8712	5372	7082	9269
RPS Large	79	99	98	98
Average Response Time Small Data	1.120ms	1.320ms	1.390ms	1.050ms
Median Response Time Small Data	1.030ms	1.700ms	1.130ms	0.914ms
Average Response Time Slow Data	126ms	101ms	102ms	101ms
Median Response Time Slow Data	118ms	101ms	101ms	101ms

Conclusion

- Avantages
  - Un micro framework léger et rapide
  - facile à prendre en main
  - 2 versions helidon SE et helidon MP
  - Idéal pour les architectures microservices

- Inconvénients
  - Nouveau framework (février 2019)
  - Peu de documentation et de tutoriels

Démonstration