

Eclipse MicroProfile for the Busy Developer

Víctor Orozco

6 de Dezembro de 2018

@tuxtor







**Oracle
Groundbreakers**



ORACLE
Certified Professional
Java SE 8 Programmer

ORACLE
Certified Associate
Java SE 8 Programmer

- 10 anos em Java EE
- CTO@Nabenik
- @tuxtor
- <http://vorozco.com>
- <http://tuxtor.shekalug.org>

Microservicios?



NABENIK

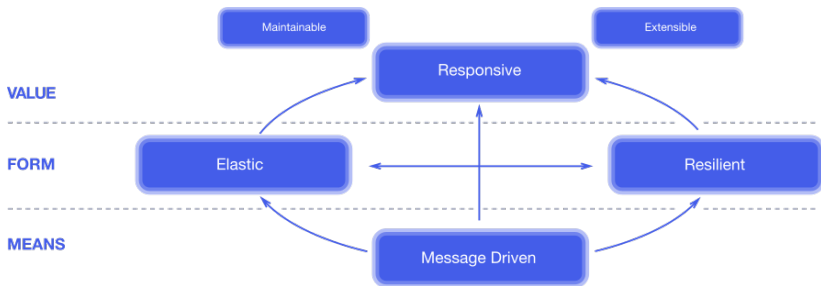
Microservicios?

Preciso migrar tudo para microservicios? O que acontece se não consigo utilizar microservicios?, estou ficando velho?



Reactive applications

Aplicativos reativos



Microserviços são uma ferramenta para um estilo novo de arquitetura.

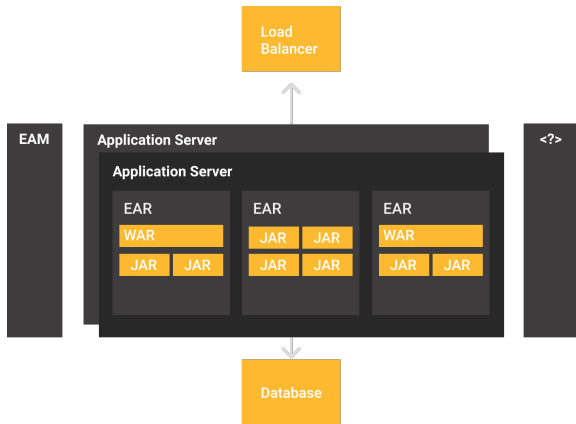


Figura 1: Monólito

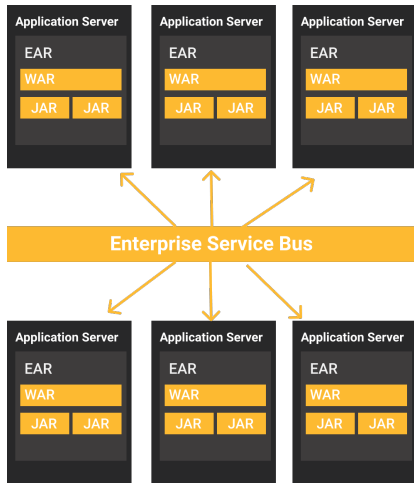


Figura 2: ESB

Microservicios

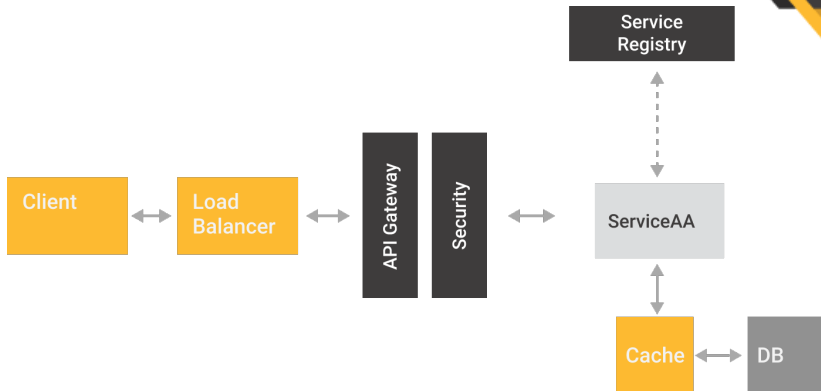


Figura 3: Microservicios

Java EE 8



NABENIK

Java EE 8



Batch	Dependency Injection	JACC	JAXR	JSTL	Management
Bean Validation	Deployment	JASPIC	JMS	JTA	Servlet
CDI	EJB	JAX-RPC	JSF	JPA	Web Services
Common Annotations	EL	JAX-RS	JSON-P	JavaMail	Web Services Metadata
Concurrency EE	Interceptors	JAX-WS	JSP	Managed Beans	WebSocket
Connector	JSP Debugging	JAXB			
JSON-B	Security				

Java EE 8

- Melhor integração JSF-CDI
- Melhor integração JMS-CDI
- HTTP/2
- JSON-B
- Security
- **JAX-RS Reativo**





JAKARTA EE



JAVA EE
GUARDIANS



MICROPROFILE™
OPTIMIZING ENTERPRISE JAVA



NABENIK

Eclipse MicroProfile



NABENIK

Eclipse MicroProfile

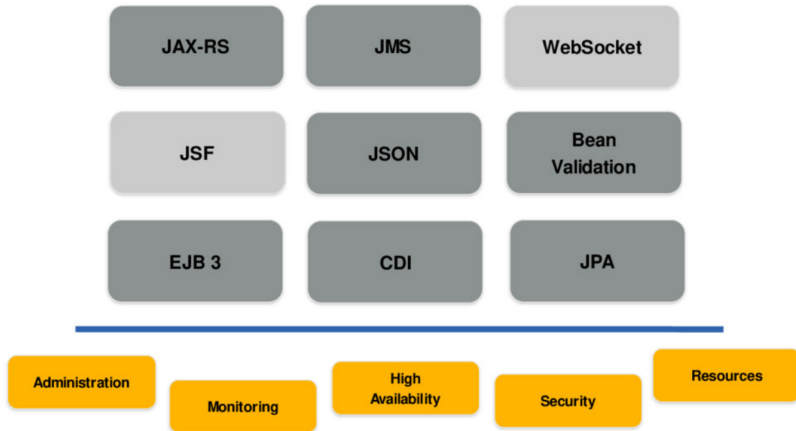
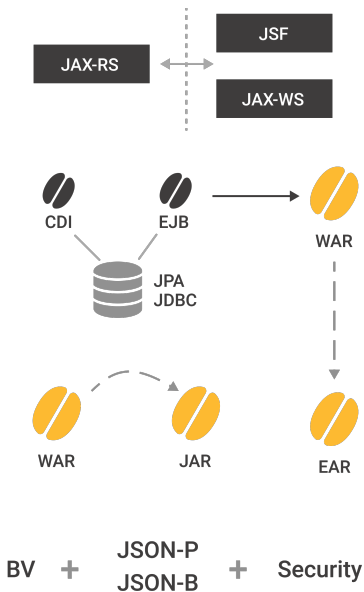


Figura 4: Credito: Reza Rahman



Eclipse MicroProfile



Eclipse MicroProfile

Open Tracing 1.0	Open API 1.0	Rest Client 1.0	JSON-B 1.0
Fault Tolerance 1.0	Metrics 1.0	JWT Propagation 1.0	Health Check 1.0
CDI 2.0	JSON-P 1.1	JAX-RS 2.1	Config 1.1

MicroProfile 2.0

-  = New
-  = No change from last release

Eclipse MicroProfile - Implementações

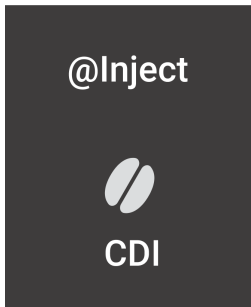
- Thorntail (Red Hat)
- KumuluzEE
- Open Liberty (IBM)
- TomEE
- Helidon (Oracle)
- Hammock
- Apache Meerowave
- **Payara Micro**

Eclipse MicroProfile on Payara 5

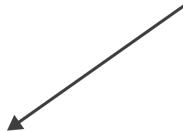
```
<dependency>  
  <groupId>org.eclipse.microprofile</groupId>  
  <artifactId>microprofile</artifactId>  
  <type>pom</type>  
  <version>2.0.1</version>  
  <scope>provided</scope>  
</dependency>
```



Config



Props file



#!/bin/bash
Env var



Value



```
@Inject  
@ConfigProperty(name = "omdbservice.url")  
String omdbDaemonServiceUrl;
```

Externalização da configuração (VM, Docker, Kubernetes)

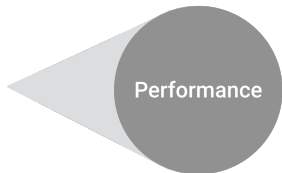
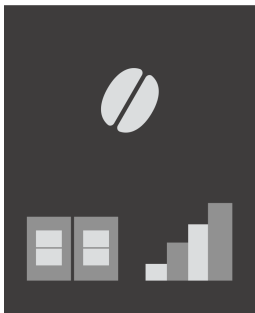


Fault Tolerance

~~a () X~~
aFallback()

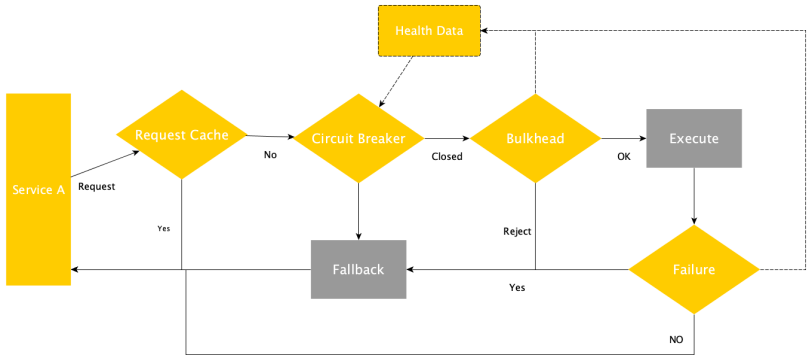


Metrics



Fault Tolerance + Metrics

- *Fault Tolerance* depende da existência de métricas, que podem ser utilizadas para diagnose do nosso sistema expondo os números mediante *Metrics*



Regras de avaliação e alternativas

- Circuit Breaker
- Bulkhead
- Retry
- Timeout
- Fallback



Fault tolerance - Fallback, Timeout

```
@GET
@Path("/{id:[a-z]*[0-9][0-9]*}")
@Fallback(fallbackMethod = "findByIdFallback")
@Timeout(TIMEOUT)
public Response findById(@PathParam("id")
final String imdbId) {
    ...
}

public Response findByIdFallback(@PathParam("id")
final String imdbId) {
    ...
}
```



- JSON or OpenMetrics (Prometheus)
- Vendor
- Base
- Application

¿Quais?

- Counted
- Gauge
- Metered
- Timed
- Histogram



Metrics - Counted

```
@Inject
@Metric
Counter failedQueries;

@GET
@Path("/{id:[a-z]*[0-9][0-9]*}")
@Fallback(fallbackMethod = "findByIdFallBack")
@Timeout(TIMEOUT)
public Response findById(@PathParam("id")
final String imdbId) {
    ...
}

public Response findByIdFallBack(@PathParam("id")
final String imdbId) {
    ...
    failedQueries.inc();
}
```



Inc-dec em tempo real

```
@Gauge(unit = "ExternalDatabases", name = "movieDatabases",
public long getDatabases() {
    int number = (int)(Math.random() * 100);
    int criteria = number % 2;

    if(criteria == 0) {
        return 100;
    }else {
        return 50;
    }
}
```

```
/metrics/application/movieDatabases
```



Events rate

```
@Metered(name = "moviesRetrieved",
         unit = MetricUnits.MINUTES,
         description = "Metrics_to_monitor_movies",
         absolute = true)
public Response findExpandedById(@PathParam("id") final Long
                                 /metrics/application/movieDatabases
```



Desempenho e delay

```
@Timed(name = "moviesDelay",  
        description = "Metrics to monitor the times of movie",  
        unit = MetricUnits.MINUTES,  
        absolute = true)  
public Response findExpandedById(@PathParam("id") final Long  
  
/metrics/application/moviesDelay
```



Distribuições

```
@Inject
```

```
MetricRegistry registry;
```

```
@POST
```

```
@Path("/add/{attendees}")
```

```
public Response addAttendees(@PathParam("attendees") Long a
```

```
    Metadata metadata =
```

```
        new Metadata("matrix_attendees",  
                    MetricType.HISTOGRAM);
```

```
    Histogram histogram =
```

```
        registry.histogram(metadata);
```

```
    histogram.update(attendees);
```

```
    return Response.ok().build();
```

```
}
```



Health check



Docker

Kubernetes

- OK?
- How much ok?



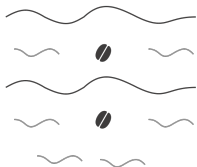
Ta vivo ainda?

```
@Override
```

```
public HealthCheckResponse call() {  
    return HealthCheckResponse.named("TaVivoAinda")  
        .withData("key1", "val1")  
        .withData("key2", "val2")  
        .up()  
        .build();  
}
```



JWT



`@Inject` _____
Principal

`@Inject` _____
Realm



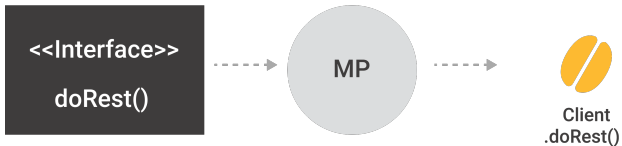
```
@LoginConfig(authMethod = "MP-JWT")
public class ApplicationConfig extends Application {
}

@Inject
private JsonWebToken jwtPrincipal;

@Inject
@Claim("email")
private String email;
```



Type Safe



```
@Path("/playlist")
@Consumes("application/json")
public interface MusicPlaylistService {

    @GET
    List<String> getPlaylistNames();

    @PUT
    @Path("/{playlistName}")
    long updatePlayList(@PathParam("playlistName")
        String name,
        List<Song> playlist)
        throws UnknownPlaylistException;
}
```



Demo



NABENIK

Java 8, JAX-RS, CDI, EJB, MicroProfile

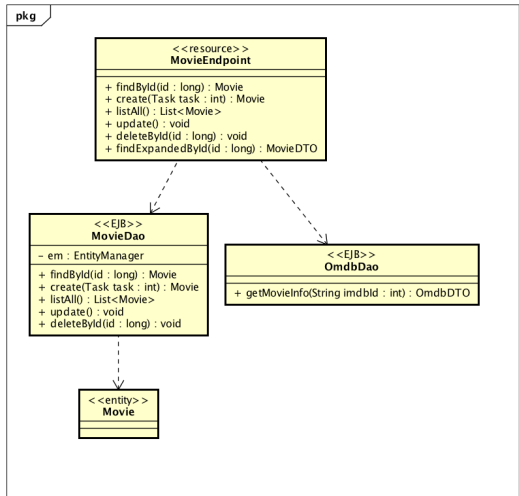
<https://github.com/tuxtor/payara-demo>

<https://github.com/tuxtor/omdb-demo>



Stacks tradicionais

- EJB
- **JTA**
- JAX-RS
- CDI

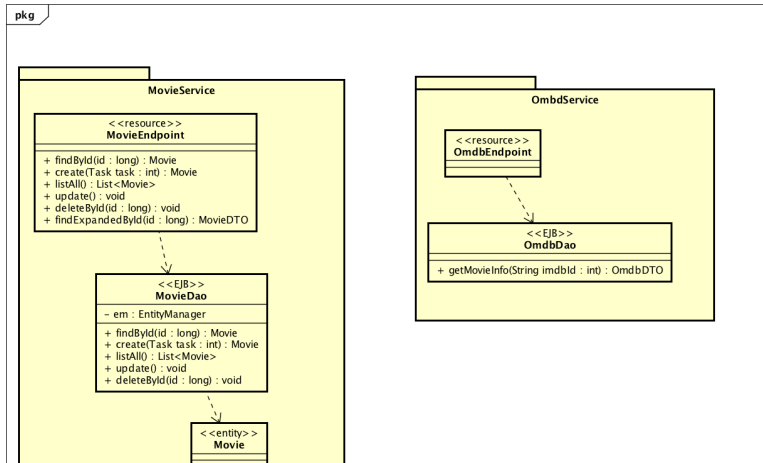


EE + MicroProfile - Demo

MicroProfile: JAX-RS, CDI, Config, Fault Tolerance, Metrics

Payara Micro: EJB, JTA

Fatores externos: Location, Deployment, Orchestration, Balancing, Consistency, Patterns



Microprofile

- Config
- Backing service
- Disposability

Cloud

- Codebase (Git-Flow)
- Dependencies (Maven)
- Build, Release, Run
- Processes (Pipelines)
- Port binding
- Concurrency (Docker - k8s)
- Dev / Prod parity
- Logs
- Admin process



MENU ORACLE
Cloud Infrastructure

Containers

Clusters
Registry

Registry

Create Repository

tuxor

- microprofile (Public)
- microprofile/omdb-demo
- microprofile/payara-demo (Public)

1

microprofile/payara-demo

User: vorozco@nabenik.com

Created: an hour ago

Access: Public

Readme

No readme has been created yet for this repository.



RUNNING

instance-20181206-0243

Create Custom Image

Start

Stop

Reboot

Terminate

Apply Tag(s)

Instance Information

Tags

Instance Information

Availability Domain: hgWe:US-ASHBURN-AD-1

Fault Domain: FAULT-DOMAIN-2

Region: iad

Shape: VM.Standard2.1

Virtual Cloud Network: [vcn20181206044411](#)

Maintenance Reboot: -

Primary VNIC Information



```
tuxtor@millenium-falcon-2:~$ docker tag omdb-demo iad.ocir.io/tuxtor/microprofile/omdb-demo:latest
```

```
tuxtor@millenium-falcon-2:~$ docker push iad.ocir.io/tuxtor/microprofile/omdb-demo
The push refers to repository [iad.ocir.io/tuxtor/microprofile/omdb-demo]
31d661ce120c: Pushing [=====>] 952.3kB/1.46MB
5dd1fd455c13: Layer already exists
2aa3decaee68: Layer already exists
685fdd7e6770: Layer already exists
c9b26f41504c: Layer already exists
cd7100a72410: Layer already exists
```



Stateful Rules

Source: 0.0.0.0/0	IP Protocol: TCP	Source Port Range: All	Destination Port Range: 22	Allows: TCP traffic for ports: 22 SSH Remote Login Protocol
Source: 0.0.0.0/0	IP Protocol: ICMP	Type and Code: 3, 4		Allows: ICMP traffic for: 3, 4 Destination Unreachable: Fragmentation Needed and Don't Fragment was Set
Source: 10.0.0.0/16	IP Protocol: ICMP	Type and Code: 3		Allows: ICMP traffic for: 3 Destination Unreachable
Source: 0.0.0.0/0	IP Protocol: TCP	Source Port Range: All	Destination Port Range: 8080-8085	Allows: TCP traffic for ports: 8080-8085



Obrigado

- me@vorozco.com
- @tuxtor
- <http://vorozco.com>
- <http://github.com/tuxtor/slides>



This work is licensed under a Creative Commons Attribution-ShareAlike 3.0.

