Rapid Development Tools for Java EE 8 & MicroProfile

Gaurav Gupta, Payara Developer
About me

• Payara Principal Developer
• NetBeans Dream Team member
• Jeddict Creator (jeddict.github.io)
• @ImJeddict

Gaurav Gupta
@jGauravGupta
Agenda

• What’s new in Java EE 8 & MicroProfile: Quick overview
• Application generation in microservice style w/ Jeddict
• Q&A
What’s new with Java EE 8?

- JAX-RS 2.1
- CDI 2.0
- Security API 1.0
- JSON-B 1.0
- JPA 2.2
- Bean Validation 2.0
- Servlet 4.0
- JSF 2.3

Checkout [https://github.com/javaee-samples/javaee8-samples](https://github.com/javaee-samples/javaee8-samples) for details on the examples
JSON-B

- Standard solution like JAXB
- Default mapping between classes and JSON
- Customizable
  - Compile time
    - Property naming @JsonbProperty
    - Property ignoring @JsonbTransient
    - Null handling @JsonbNillable
    - Property ordering @JsonbPropertyOrder
    - Date Format @JsonbDateFormat
    - Number Format @JsonbNumberFormat
    - Adapter @JsonbTypeAdapter
  - Runtime configuration
    - Configuration builder JsonbConfig
class Employee {

    private String name;

    private String pin;

    private String email;

    {
        "name": "Gaurav",
        "pin": "J1-Secret",
        "email": "gaurav.gupta@payara.fish",
    }
}
@JsonbPropertyOrder({"email", "name"})
class Employee {

    @JsonbProperty("empName")
    private String name;

    @JsonbTransient
    private String pin;

    private String email;

    {
        "email": "gaurav.gupta@payara.fish",
        "empName": "Gaurav"
    }
}
Bean Validation 2

- Java 8 Date and Time API Support

```java
@Past(message = "must be a past date")
private java.time.Year yearOfBirth;
```

- Type Annotations

```java
private List<@NotNull @Email String> emails;
private String @NotNull @Email[] emails;
private Map<@Valid Employee, @Valid Address> addressMap = new HashMap<>();
```
Bean Validation 2

- java.util.Optional Support

```java
private Optional<@Past LocalDate> marriageAnniversary;
private Optional<@Size(max = 20) String> name;
```

- Repeating Annotations

```java
@Max(value = 2000, groups = Default.class)
@Max(value = 5000, groups = GoldCustomer.class)
private long withdrawalAmount;
```
Bean Validation 2

- Introduces new constraints
  - @Email
  - @NotBlank
  - @NotEmpty
  - @PastOrPresent
  - @FutureOrPresent
  - @Negative
  - @NegativeOrZero
  - @Positive
  - @PositiveOrZero
JPA 2.2

- @Repeatable annotations
- Support Java 8 Date and Time API
- Ability to return stream of query result
- CDI Injection in AttributeConverters
@Entity
@NamedQueries({
    @NamedQuery(name = "Employee.findAll",
                query = "SELECT e FROM Employee e"),
    @NamedQuery(name = "Employee.findByName",
                query = "SELECT e FROM Employee e WHERE e.name = :name")
})
class Employee {

    @Convert(converter=LocalDateConverter.class)
    private LocalDate dateOfBirth;
}

AttributeConverter implementation
JPA 2.2

Container annotation not required

```java
@Entity
@NamedQueries({
    @NamedQuery(name = "Employee.findAll",
                query = "SELECT e FROM Employee e"),
    @NamedQuery(name = "Employee.findByName",
                query = "SELECT e FROM Employee e WHERE e.name = :name")
})

class Employee {

    @Convert(converter=LocalDateConverter.class)
    private LocalDate dateOfBirth;
}
```

AttributeConverter not required
```java
@JPA 2.2

@Entity
@NamedQuery(name = "Employee.findAll",
            query = "SELECT e FROM Employee e")
@NamedQuery(name = "Employee.findByName",
            query = "SELECT e FROM Employee e WHERE e.name = :name")
class Employee {

  private LocalDate dateOfBirth;

}
```
JPA 2.2

- Stream query results

```java
Stream<Employee> employees = em.createQuery("SELECT e FROM Employee", Employee.class)
                      .getResultStream();
```
Java EE Profiles

MicroProfile

Java EE Web Profile

Java EE Full Profile
MicroProfile 1.0

CDI 1.2  JSON-P 1.0  JAX-RS 2.0
MicroProfile 1.1

- Config 1.0
- CDI 1.2
- JSON-P 1.0
- JAX-RS 2.0

= New
= No change from last release
MicroProfile 1.2

- Health Check 1.0
- Metrics 1.0
- Config 1.1
- Fault Tolerance 1.0
- JWT Propagation 1.0
- CDI 1.2
- JSON-P 1.0
- JAX-RS 2.0

= New
= Updated
= No change from last release
MicroProfile 1.3

- Open Tracing 1.0
- Open API 1.0
- Rest Client 1.0
- Fault Tolerance 1.0
- Metrics 1.1
- JWT Propagation 1.0
- CDI 1.2
- JSON-P 1.0
- JAX-RS 2.0
- Health Check 1.0
- Config 1.2

- = New
- = Updated
- = No change from last release
MicroProfile Config

• Default Config Sources
  – System Properties
    • System.getProperties()
  – Environment Variables
    • System.getenv()
  – Application Configuration
    • META-INF/microprofile-config.properties

• Custom Config Sources
  – e.g Shared Database table
MicroProfile Config

- CDI Injection

```java
@Inject
@Config(name = "service.enabled", defaultValue = "false")
private Boolean enabled;

@Inject
@Config(name = "color", defaultValue = "white")
private String color;
```

- Programmatic Lookup

```java
ConfigProvider.getConfig()
    .getValue(String propertyName, Class<T> propertyType);
```
MicroProfile Metrics

• Annotations
  – @Counted
  – @Timed
  – @Metered
  – @Gauge
  – @Metrics

• Scope
  – Base
  – Application
  – Vendor

• Format
  – Json
  – Prometheus
MicroProfile Metrics - Endpoint

https://localhost:8080/metrics

```
# TYPE application:requests_total counter
application:requests_total 29382
# TYPE application:requests_rate_per_second gauge
application:requests_rate_per_second 12.223
# TYPE application:requests_one_min_rate_per_second gauge
application:requests_one_min_rate_per_second 12.563
# TYPE application:requests_five_min_rate_per_second gauge
application:requests_five_min_rate_per_second 12.364
# TYPE application:requests_fifteen_min_rate_per_second gauge
application:requests_fifteen_min_rate_per_second 12.126
```
MicroProfile Health Check

```java
@Health
public class CheckDiskSpace implements HealthCheck {

@Override
public HealthCheckResponse call() {
    return HealthCheckResponse.named("disk-space")
        .withData("free", "105300")
        .withData("total", "10246000")
        .up()
        .build();
}
}
```
MicroProfile Health Check - Endpoint

https://localhost:8080/health

```json
{
    "outcome": "UP",
    "checks": [
        {
            "name": "disk-space",
            "state": "UP",
            "data": {
                "free": "105300",
                "total": "10246000"
            }
        }
    ]
}
```
MicroServices Architecture

1. CLIENT
2. Service Discovery / Registry (Register)
3. Query
4. Routing

Gateway

Microservices
1. MICROSERVICE 1
2. MICROSERVICE 2
..n MICROSERVICE

Database
1. db 1
2. db 2
..n db
MicroServices Architecture

1. **CLIENT**

2. **Gateway**
   - Consul Service Discovery & Registry
   - Query

3. **Gateway**
   - Register

4. **Gateway**
   - Routing

   - Employee Service
   - Product Service
   - MICROSERVICE

   - db 1
   - db 2
   - db...n
Java EE 8 & MicroProfile App generation w/ jeddict.github.io
Java EE to Jakarta EE
Jakarta EE

The New Home of Cloud Native Java

Powered by participation, Jakarta EE is focused on enabling community-driven collaboration and open innovation for the cloud.

Jakarta EE Working Group  Stay Connected

Strategic Members

Fujitsu  IBM  Oracle  Payara  Red Hat  Tomitribe
Thank you
GREAT INDIAN DEVELOPER SUMMIT 2019

Conference: April 23-26, Bangalore

Register early and get the best discounts!

www.developersummit.com  @greatindiandev  bit.ly/gidslinkedin