

## Arrowhead Core Systems - Golang Implementation

### Deployment Scenario

#### About

- A local deployment scenario has been implemented with a provider and consumer systems based on the *core-java-spring* Github repository skeletons <sup>1</sup>. The core systems implemented in Golang are found on the *eclipse-arrowhead* Github repository <sup>2</sup>. The code, scripts and output of this scenario can be found on the *go-arrowhead-deployment* Github repository <sup>3</sup>.

#### Results

- The Service Registry implementation has been tested and works correctly on registering, unregistering and querying services.
- The Orchestrator implementation has been tested on the orchestration service but it does not work correctly. The code is not complete and currently does not use the Service Registry, it simply responds an empty *OrchestrationResponseDTO*.
- The provider and consumer systems use HTTP protocol and JSON encoding, both communication and information exchange are correctly performed on both Service Registry and Orchestrator.

#### Implementation Issues

- **/common/util/util.go : 66, 91**  
A static address is being used, not obtained from the *application.toml* file.
- **/Orchestrator/routes.go : 25**  
Typo in the route of the orchestration service (*/orchstrator/orchestration*) resulting in 404 error code.
- **/Orchestrator/handlers.go : 91**  
An empty response is being generated and responded to the requesting system.
- **/common/datamodels/datamodels.go : 114**  
In the *OrchestrationFormRequestDTO* the information about the service (*ServiceQueryFormDTO*) is not reflected. The comparison of the JSON requests of the legacy and current DTO for the orchestration service can be found on next page.

#### Other Information

- The OS type and version used for this scenario was Ubuntu 22.04.1 LTS.
- The compilation of the *support-java-spring/* systems found in the *go-arrowhead-deployment* repository has to follow that of the *core-java-spring/core-common/* code and latter execution of the script */core-java-spring/core-common/essentials-deploy/deploy.sh* for its installation in the local Maven repository.

---

<sup>1</sup><https://github.com/eclipse-arrowhead/core-java-spring>

<sup>2</sup><https://github.com/thingwave/eclipse-arrowhead>

<sup>3</sup><https://github.com/fernand0labra/go-arrowhead-deployment>

## JSON representation of the OrchestrationFormRequestDTO

### Legacy OrchestrationFormRequestDTO (core-java-spring)

```
1 {"requesterSystem":{"
2   "systemName":"temperature-controller-1",
3   "address":"localhost",
4   "port":8080,
5   "authenticationInfo":null,
6   "metadata":null
7 },
8 "requesterCloud":null,
9 "requestedService":{"
10  "serviceDefinitionRequirement":"get-temperature",
11  "interfaceRequirements":["HTTP-INSECURE-JSON"],
12  "securityRequirements":null,
13  "metadataRequirements":null,
14  "versionRequirement":null,
15  "minVersionRequirement":null,
16  "maxVersionRequirement":null,
17  "pingProviders":false
18 },
19 "orchestrationFlags":{"...}}
```

### Current OrchestrationFormRequestDTO (eclipse-arrowhead)

```
1 {RequesterSystem:{
2   SystemName:temperature-controller-1,
3   Address: localhost,
4   Port:8080,
5   AuthenticationInfo: Metadata:<nil>
6 }
7 RequesterCloud:{
8   Operator: ,
9   Name: ,
10  Secure:false,
11  Neighbor:false,
12  AuthenticationInfo: ,
13  GatekeeperRelayIds:[],
14  GatewayRelayIds:[]
15 }}
```