GreenFaaS: Description Language

Celeste Guimapi

Donald Onana

Seb's Workflow model

Figure 1 – Seb's workflow model is based on top of workflow nets with data (WFD-nets)

WFD-nets is a **Petri Net extension** define by the tuple $\langle P, T, F, D, r, w, d, grd \rangle$ where :

- $\langle P, T, F \rangle$ is a Petri Net with P : places (conditions), T : transitions (actions), F : arcs connecting places and transitions
- **D** is a set of data manipulated by the workflow.
- $\langle r, w, d \rangle$ set of operations that can be applied to D, respectively, read, write, and destroy.
- *grd* is a logical condition that must be met to activate a transition

Workflow syntax definition

- To enable developers to define a workflows in Seb's-Flow conforming model, the authors proposed a **uniform JSON syntax** wich can be transcribed to specific platform syntax.
- Each platform have **specific generators (parser)** that transcribe workflows to the respective proprietary definition
- The solution currently support three major cloud providers : AWS, Google Cloud, and Azure.
- Any platform service can be adding by implementing a single interface transcribing Seb's model to the cloud-specific interface.

Workflow syntax definition : exemple

Figure 2 – suppose we have to deploy this workflow.

```
{
    "root": "t1_phase",
    "states": {
        "t1_phase": {
            "type": "task",
            "func_name": "t1",
            "next": "t23_phase",
        },
```

Figure 3 – Seb's syntax definition.

```
"t23_phase":{
    "type": "parallel",
    "parallel_functions": [
        {
            "root": "t2_phase",
            "states": {
            "t2_phase": {
                "type": "task",
                "func_name": "t2"
            }
            }
        },
        {
            "root": "t3_phase",
            "states": {
            "t3_phase": {
                "type": "task",
                "func_name": "t3"
            }
            }
        }
    ],
    "next": "t4_phase"
    }
```

Extended workflow syntax

- Create a new state type **(`alternative`)** to describe alternatives.
- Provide more fields to describe more informations like influence factor, preference ... etc.

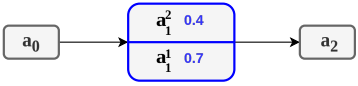

Figure 4 – suppose we have to deploy this little workflow, where after having executed the function a_0, we can execute either a_1^1 or a_1^2

```
"root": "a0_phase",
"states": {
    "a0_phase": {
        "type": "task",
        "func_name": "a0",
        "next": "a1_phase",
    },
```

```
"a1_phase":{
    "type": "alternative",
    "alternative_functions": [
        {
            "pref": "0.7",
            "states": {
            "a11_phase": {
                "type": "task",
                "func_name": "a11"
            }
            }
        },
        {
            "pref": "0.4",
            "states": {
            "a12_phase": {
                "type": "task",
                "func_name": "a12"
            }
            }
        },
    ],
    "next": "a2_phase"
    }
```

Figure 5 – Seb's extended syntax for GreenFaaS.

Extended workflow syntax

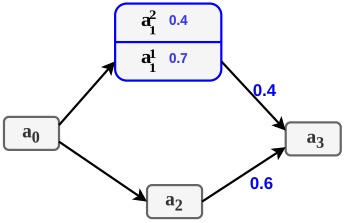

Figure 6 – suppose we need to deploy this workflow.

```
"root": "a0_phase",
"states": {
    "a0_phase": {
        "type": "task",
        "func_name": "a0",
        "next": "ai_phase",
    },
```

Figure 7 – Seb's extended syntax for GreenFaaS.

```
"ai_phase":{
    "type": "parallel",
    "parallel_functions": [
        {
            "root": "a1_phase",
            "inf" : 0.4,
            "states": {
                "a1_phase":{
                    "type": "alternative",
                    "alternative_functions": [
                        {
                            "pref": "0.7",
                            "states": {
                            "a11_phase": {
                                "type": "task",
                                "func_name": "a11"
                            }
                            }
                        },
                        .
                        .
                    ],
                }
            }
        },
        .
        .
    ],
    "next": "a3_phase"
}
```