Project: Developing, Deploying, Running, and Evaluating a Complex Serverless Workflow Application (FC)

November 2020

1 General project description

The aim of the project is to develop a serverless application comprising of several functions that use cloud services. The project will have six phases:

1. **Develop the FC** (Function Choreography) with our FC Editor[1], which will generate an AFCL file. This phase covers mainly the control flow of the FC;
2. **Develop the functions** of the FC and deploy them on AWS Lambda. Define the inputs and outputs of each function;
3. **Run your FC** with our *xAFCL* engine. Build CFCL from your AFCL (prepare input JSON, specify resource properties in the AFCL, data flow, as well as source in `dataIns` of the FC);
4. **Evaluate your FC**;
5. **Write a technical report**;
6. **Prepare a presentation and present your project**;

You will work on phases 1, 2, and 3 of the project as a part of homeworks 06, 07, and 08, respectively. Details for each of these homeworks will be described in a separate task given in OLAT and explained by the corresponding lecturer during the proseminar.

Several projects are defined and accessible on the following link[2]. For each project, you can see the possible programming languages in which you can develop the FC functions. One team can select a single programming language from the given options. Note that there is a single programming language for

[1] http://fceditor.dps.uibk.ac.at:8180/

[2] https://github.com/sashkoristov/PSDS2020W/tree/main/project

some projects. Please read descriptions of all projects and discuss within the team which pair (project and programming language) is most suitable for you.

Each team can send an email with your preferences (sashko@dps.uibk.ac.at) starting from the most preferable project until the least preferable one.

We are open to your proposals for similar use cases from the real life that can be composed similarly as given FCs. The minimum requirements are:

- complex FC with `parallelFor` and `parallel`, so that it makes sense to run it on FaaS system.
- Data flow (distribution or element index)
- at least one serverless function per student that uses some cloud service (not necessarily of AWS).

The main goal is to distribute the work across multiple resources (serverless functions in this case), which assumes to achieve some speedup and increase the throughput.

2 Homework 06 - Develop your FC in AFCL with our FCEditor

Analyse and build the assigned FC in AFCL using the FCEditor. Explain which compound functions you will use and the level of parallelism (parallel sections and parallel loops).

Think of your functions and their dataIns and dataOuts.

How will you specify the data-flow and data distribution?

Apply the visual representation of your draft FC.

3 Homework 07 - Develop the FC functions

Develop, deploy, and test the FC functions on AWS. Define all inputs and outputs of functions. Define data flow.

4 Homework 08 - Integration

Run the FC with our *xAFCL* enactment engine. Prepare input JSON, convert AFCL into CFCL.

5 Important notes

- You can use smaller inputs than the required ones for the evaluation part to test your project while developing.
- You can test your FC from Homework 06 with "dummy" or no-op functions and run it with *xAFCL*.

6 Options for Bachelor theses

Interested students can contact us for an extension of the project towards a bachelor thesis (one student or a group of two students). The topics will be defined and published on our web site[3].

[3] http://dps.uibk.ac.at/bachelor-available-theses/