ID2209 HT23 Distributed Artificial Intelligence and Intelligent Agents

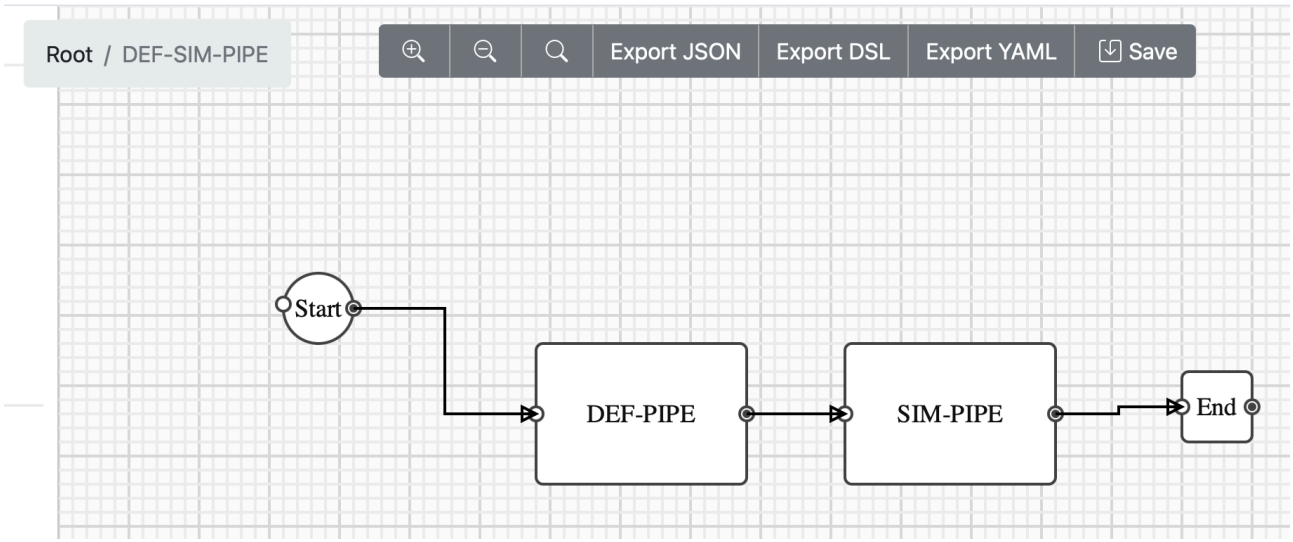
DataCloud Evaluation

by Eugen Lucchiari Hartz

December 01, 2023

Task 1

Screenshot with DEF-SIM-PIPE pipeline graphical view



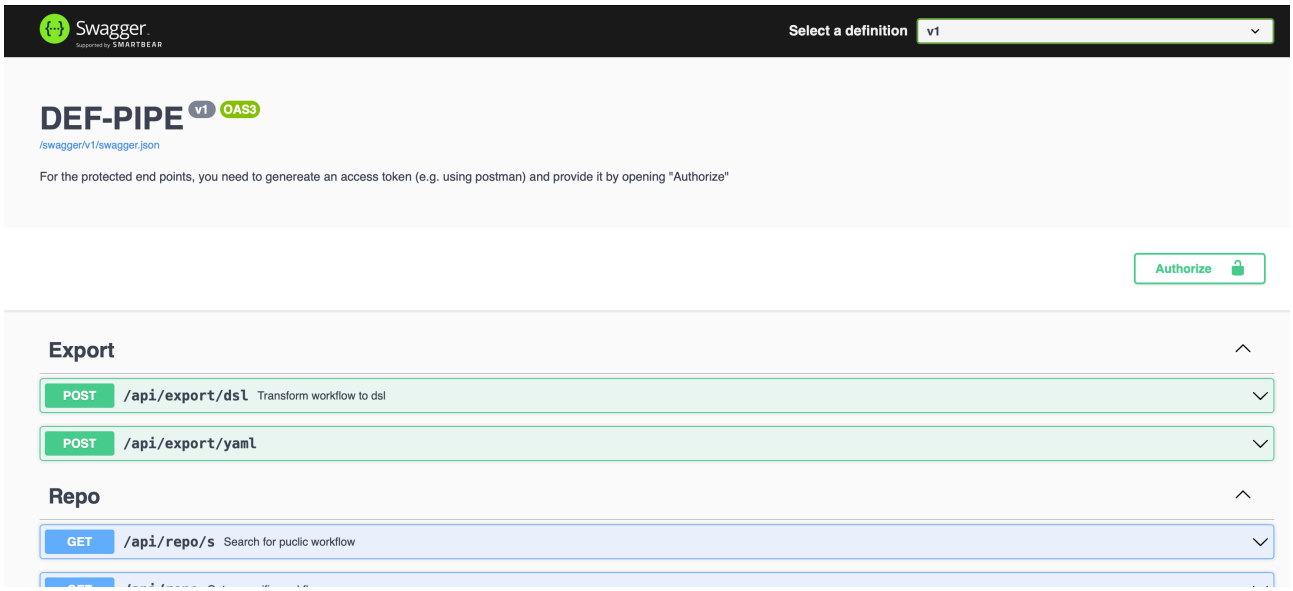
Generated YAML text from the graphical view of the pipeline

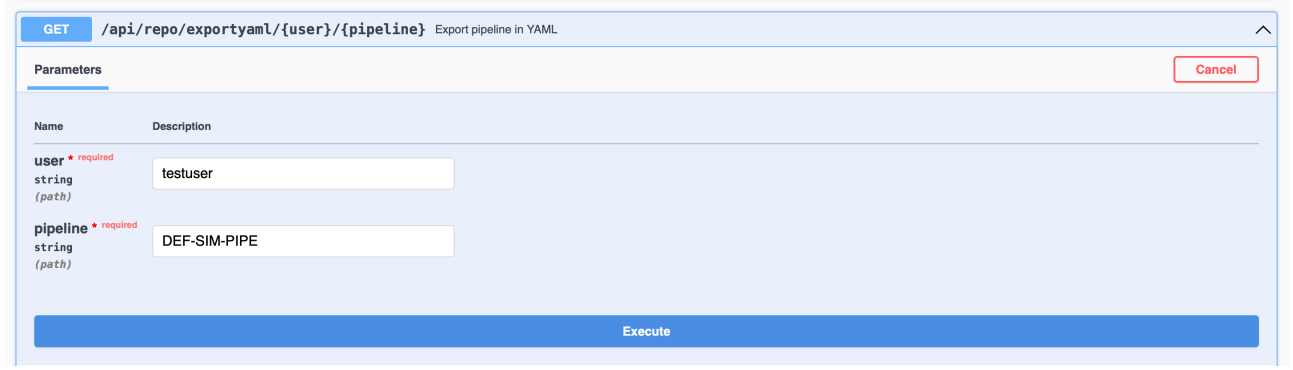
```
apiVersion: argoproj.io/v1alpha1
kind: Workflow
metadata:
  generateName: def-sim-pipe
spec:
  entrypoint: main-workflow
  templates:
    - name: main-workflow
      dag:
        tasks:
```

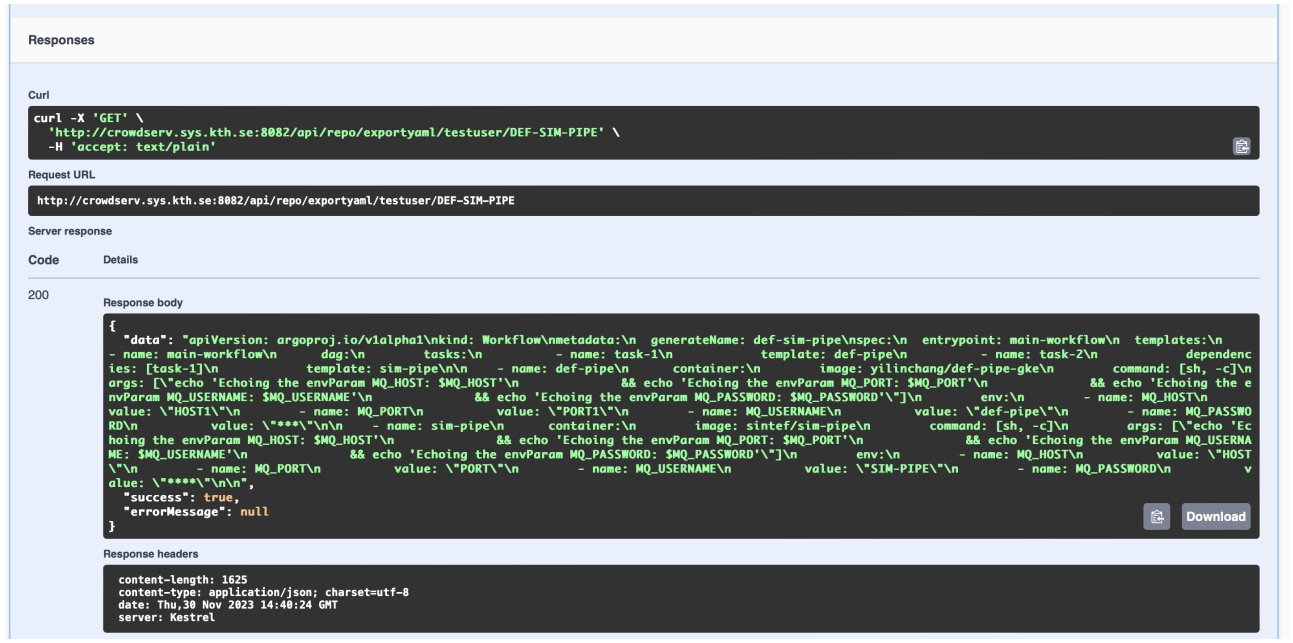
```
      - name: task-1
        template: def-pipe
      - name: task-2
        dependencies: [task-1]
        template: sim-pipe

- name: def-pipe
  container:
    image: yilinchang/def-pipe-gke
    command: [sh, -c]
    args: ["echo 'Echoing the envParam MQ_HOST: $MQ_HOST'
          && echo 'Echoing the envParam MQ_PORT: $MQ_PORT'
          && echo 'Echoing the envParam MQ_USERNAME: $MQ_USERNAME'
          && echo 'Echoing the envParam MQ_PASSWORD: $MQ_PASSWORD'"]
    env:
    - name: MQ_HOST
      value: "HOST1"
    - name: MQ_PORT
      value: "PORT1"
    - name: MQ_USERNAME
      value: "def-pipe"
    - name: MQ_PASSWORD
      value: "***"

- name: sim-pipe
  container:
    image: sintef/sim-pipe
    command: [sh, -c]
    args: ["echo 'Echoing the envParam MQ_HOST: $MQ_HOST'
          && echo 'Echoing the envParam MQ_PORT: $MQ_PORT'
          && echo 'Echoing the envParam MQ_USERNAME: $MQ_USERNAME'
          && echo 'Echoing the envParam MQ_PASSWORD: $MQ_PASSWORD'"]
    env:
    - name: MQ_HOST
      value: "HOST"
    - name: MQ_PORT
      value: "PORT"
    - name: MQ_USERNAME
      value: "SIM-PIPE"
    - name: MQ_PASSWORD
      value: "****"
```

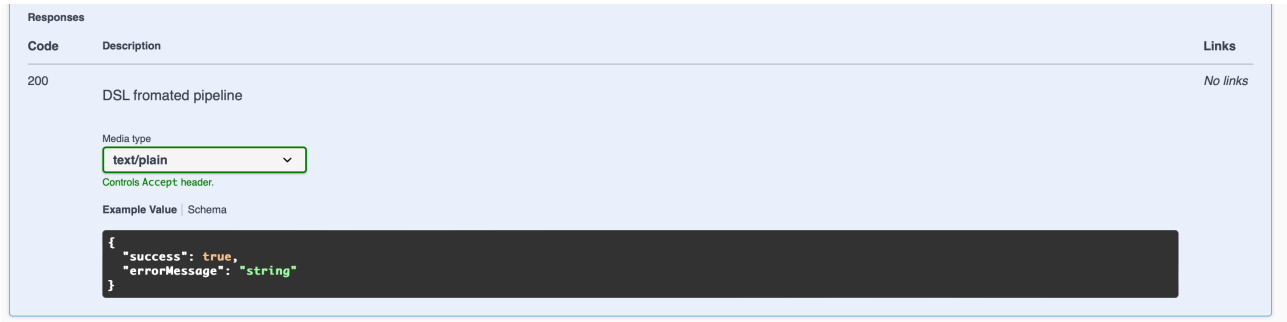
Task 2

Protocol of invocation and output of invocation of the API from http://crowdserv.sys.kth.se:8082/docs









curl -X 'GET' \
'http://crowdserv.sys.kth.se:8082/api/repo/exportyaml/testuser/DEF-SIM-PIPE' \
-H 'accept: text/plain'

http://crowdserv.sys.kth.se:8082/api/repo/exportyaml/testuser/DEF-SIM-PIPE

{
"data": "apiVersion: argoproj.io/v1alpha1\nkind: Workflow\nmetadata:\n generateName: def-sim-pipe\nspec:\n entrypoint: main-workflow\n templates:\n - name: main-workflow\n dag:\n tasks:\n - name: task-1\n template: def-pipe\n - name: task-2\n dependencies: [task-1]\n template: sim-pipe\n\n - name: def-pipe\n container:\n image: yilinchang/def-pipe-gke\n command: [sh, -c]\n args: [\"echo 'Echoing the envParam MQ_HOST: $MQ_HOST'\n && echo 'Echoing the envParam MQ_PORT: $MQ_PORT'\n && echo 'Echoing the envParam MQ_USERNAME: $MQ_USERNAME'\n && echo 'Echoing the envParam MQ_PASSWORD: $MQ_PASSWORD'\"]\n env:\n - name: MQ_HOST\n value: \"HOST1\"\n - name: MQ_PORT\n value: \"PORT1\"\n - name: MQ_USERNAME\n value: \"def-pipe\"\n - name: MQ_PASSWORD\n value: \"***\"\n\n - name: sim-pipe\n container:\n image: sintef/sim-pipe\n command: [sh, -c]\n args: [\"echo 'Echoing the envParam MQ_HOST: $MQ_HOST'\n && echo 'Echoing the envParam MQ_PORT: $MQ_PORT'\n && echo 'Echoing the envParam MQ_USERNAME: $MQ_USERNAME'\n && echo 'Echoing the envParam MQ_PASSWORD: $MQ_PASSWORD'\"]\n env:\n - name: MQ_HOST\n value: \"HOST\"\n - name: MQ_PORT\n value: \"PORT\"\n - name: MQ_USERNAME\n value: \"SIM-PIPE\"\n - name: MQ_PASSWORD\n value: \"****\"\n\n",
"success": true,
"errorMessage": null
}

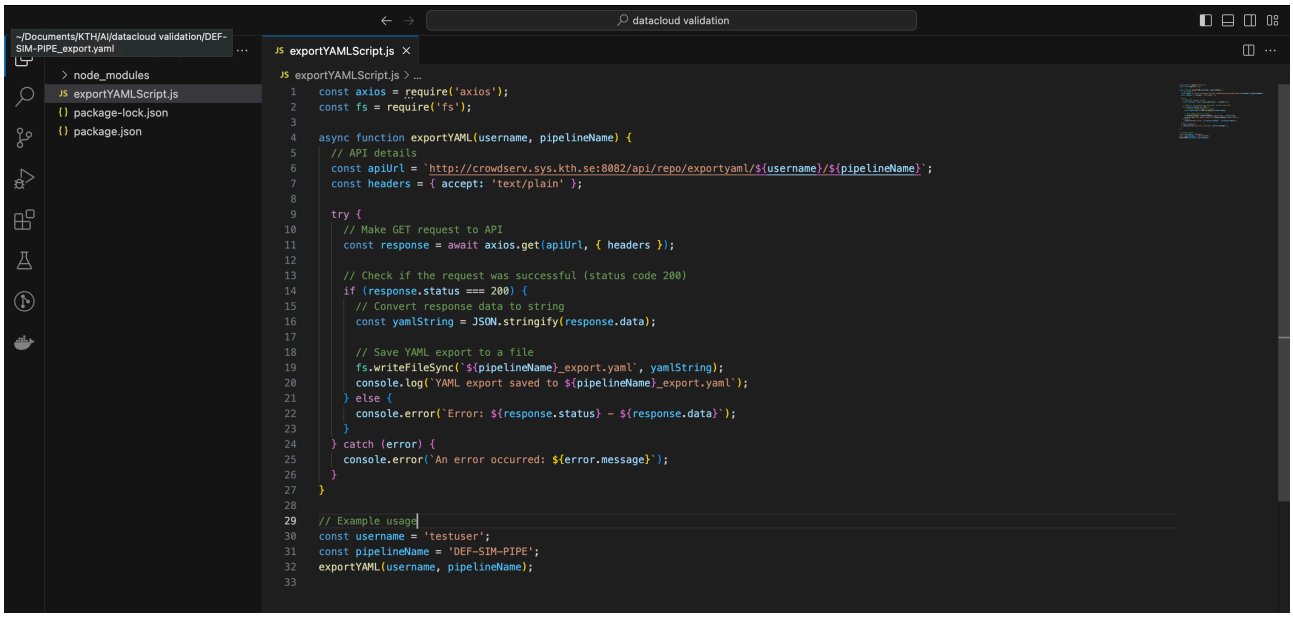
content-length: 1625
content-type: application/json; charset=utf-8
date: Thu,30 Nov 2023 14:40:24 GMT
server: Kestrel

{
"success": true,
"errorMessage": "string"

}

Task 3

Code for invocation of API from your program and invocation result



exportYAMLScript.js

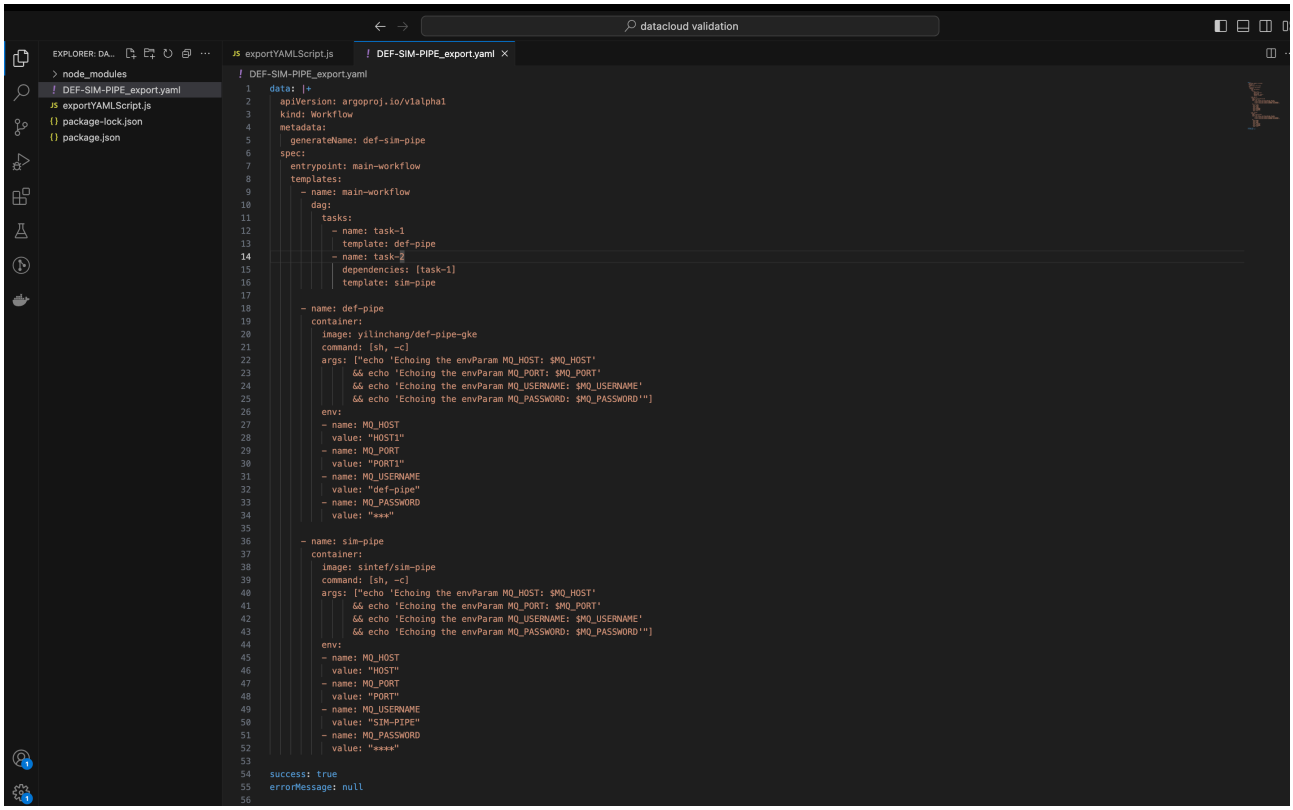
```
const axios = require('axios');
const fs = require('fs');

async function exportYAML(username, pipelineName) {
  // API details
  const apiUrl = `http://crowdserv.sys.kth.se:8082/api/repo/
exportyaml/${username}/${pipelineName}`;
  const headers = { accept: 'text/plain' };

  try {
    // Make GET request to API
    const response = await axios.get(apiUrl, { headers });

    // Check if the request was successful (status code 200)
    if (response.status === 200) {
      // Convert response data to string
      const yamlString = JSON.stringify(response.data);

      // Save YAML export to a file
      fs.writeFileSync(`${pipelineName}_export.yaml`, yamlString);
      console.log(`YAML export saved to ${pipelineName}
_export.yaml`);
    } else {
```

```
      console.error(`Error: ${response.status} - ${response.data}
`);
    }
  } catch (error) {
    console.error(`An error occurred: ${error.message}`);
  }
}

// Example usage
const username = 'testuser';
const pipelineName = 'DEF-SIM-PIPE';
exportYAML(username, pipelineName);
```

Invocation result

you get the result after running node exportYAMLScript.js in the terminal

DEF-SIM-PIPE_export.yaml




```
data: |+
  apiVersion: argoproj.io/v1alpha1
  kind: Workflow
  metadata:
    generateName: def-sim-pipe
```

```
  spec:
    entrypoint: main-workflow
    templates:
      - name: main-workflow
        dag:
          tasks:
            - name: task-1
              template: def-pipe
            - name: task-2
              dependencies: [task-1]
              template: sim-pipe

      - name: def-pipe
        container:
          image: yilinchang/def-pipe-gke
          command: [sh, -c]
          args: ["echo 'Echoing the envParam MQ_HOST: $MQ_HOST'
                && echo 'Echoing the envParam MQ_PORT: $MQ_PORT'
                && echo 'Echoing the envParam MQ_USERNAME:
$MQ_USERNAME'
                && echo 'Echoing the envParam MQ_PASSWORD:
$MQ_PASSWORD'"]
          env:
          - name: MQ_HOST
            value: "HOST1"
          - name: MQ_PORT
            value: "PORT1"
          - name: MQ_USERNAME
            value: "def-pipe"
          - name: MQ_PASSWORD
            value: "***"

      - name: sim-pipe
        container:
          image: sintef/sim-pipe
          command: [sh, -c]
          args: ["echo 'Echoing the envParam MQ_HOST: $MQ_HOST'
                && echo 'Echoing the envParam MQ_PORT: $MQ_PORT'
                && echo 'Echoing the envParam MQ_USERNAME:
$MQ_USERNAME'
                && echo 'Echoing the envParam MQ_PASSWORD:
$MQ_PASSWORD'"]
          env:
          - name: MQ_HOST
            value: "HOST"
          - name: MQ_PORT
            value: "PORT"
          - name: MQ_USERNAME
            value: "SIM-PIPE"
          - name: MQ_PASSWORD
            value: "****"
```

```
success: true
errorMessage: null
```