JEngine REST API Specification v2.0

BP2014W1 - Team

July 9, 2015

One goal of the JEngine was to provide a flexible API. Developers can use this API to build an UI or to control/use the engine from within another application. There are methods to configure, control and monitor processes and their instances. This document will provide a specification/documentation of the Rest-Interface, by providing the information about the allowed methods and examples .

1 Structure

`http://domain:port/jengine/api/`	`interface/{version}`
	`config/{version}`
	`jcomparser/{version}`
	`history/{version}`
	`analytics/{version}`

Table 1: API Core Path

The API is divided into four parts (Table 1):

1. **Interface.** The interface provides methods for creating and controlling instances of scenarios. More information can be found in Chapter 2.1.
2. **Config.** This API can be used to configure a process model. Currently the configuration is restricted to email tasks. More information and examples can be found in Chapter 2.2.
3. **Comparser.** The JEngine can be used to execute Production Case Management (PCM) models created with the JEDItor[1]. The JComparser is an interface between the core of the engine and the model repository provided by the JEDItor. The JComparser provides and interface to view and load models from the repository. More information and examples can be found in chapter 2.3.

[1] `BP2014W1processeditor.html`

4. **History.** Detailed information about state changes of data object and activities as well as data attribute changes of running and terminated scenario instances can be accessed via the history interface. Chapter 2.4 provides examples and detailed information.

5. **Analytics.** The JAnalytics Framework can be used to run analytic algorithms on the datalogs of the JHistory. Chapter 2.5 gives further detail.

2 Functions

This chapter provides information about all possible REST calls. For more detailed information you may take a look at the *Java doc* provides for each class.

2.1 Interface

The interface provides methods mainly used by the UI. It offers functions for viewing and controlling scenarios and their instances.

2.1.1 Scenarios

Get all Scenarios
http GET `interface/{version}/scenario`

```
{
  ”ids”:[140,141,142,143],
  ”links”:{
    ”140”:”http://domain:port/jengine/api/interface/v2/scenario/140”,
    ”141”:”http://domain:port/jengine/api/interface/v2/scenario/141”,
    ”142”:”http://domain:port/jengine/api/interface/v2/scenario/142”,
    ”143”:”http://domain:port/jengine/api/interface/v2/scenario/143”
  },
  ”labels”:{
    ”140”:”TestScenario”,
    ”141”:”CoffeeScenario”,
    ”142”:”BookingScenario”,
    ”143”:”BoschScenario”
  }
}
```

Example 1: Example output of get all Scenarios request

Get all Scenarios with filter condition
http GET `interface/{version}/scenario?filter=Coffee`

```
{
  ”ids”:[141],
  ”links”:{
    ”141”:”http://domain:port/jengine/api/interface/v2/scenario/141”
  },
  "labels":{
    "141":"CoffeeScenario"
  }
}
```

Example 2: Example output of get all Scenarios request with filter condition

Get detailed information about a Scenario
http GET `interface/{version}/scenario/1`

```
{
  "modelid":0,
  "instances":"http://domain:port/jengine/api/interface/v2/scenario/1/instance",
  "name":"HELLOWORLD",
  "id":1,
  "modelversion":0
}
```

Example 3: Example output of get a specific scenario

2.1.2 Scenario Instances

Get all scenario instances Scenario instances, that are in state terminated will not be listed.
http GET `interface/{version}/scenario/1/instance`

```
{
  "ids":[47,48,49],
  "links":{
    "47":"http://domain:port/jengine/api/interface/v2/scenario/1/instance/47",
    "48":"http://domain:port/jengine/api/interface/v2/scenario/1/instance/48",
    "49":"http://domain:port/jengine/api/interface/v2/scenario/1/instance/49"
  },
  "labels":{
    "47":"Berlin-Istanbul",
    "48":"Herr Meyer Reise",
    "49":"Sommerurlaub"
  }
}
```

Example 4: Example output of get all scenario instances

Get all scenario instances with filter condition
http GET `interface/{version}/scenario/1/instance?filter=Herr`

```
{
  "ids":[47,48,49],
  "links":{
    "48":"http://domain:port/jengine/api/interface/v2/scenario/1/instance/48"
  },
  "labels":{
    "48":"Herr Meyer Reise"
  }
}
```

Example 5: Example output of get all scenario instances with condition

Start a new instance

A new instance can be started by performing a PUT or POST. The PUT provides the possibility to name the new instance.

http PUT `interface/{version}/scenario/1/instance`

http POST `interface/{version}/scenario/1/instance`

If you want to name the scenario instance you have to provide a body (Figure 6):

```
{
  "name":"This is the new instance"
}
```

Example 6: Body of the PUT request to create a new named instance

Get detailed information about a scenario instance

http GET `interface/{version}/scenario/1/instance/48`

```
{
    "name":"Herr Meyer Reise",
    "id":48,
    "terminated":false,
    "scenario_id":1,
    "activities":"http://domain:port/jengine/api/interface/v2/scenario/1/instance/
                                                                      48/activity"
}
```

Example 7: Example output of get a single scenario instance

2.1.3 Activity Instances

List all activity instances

http GET `interface/{version}/scenario/1/instance/48/activity`

```
{
    "activities":{
        "189":{
            "link":"http://domain:port/jengine/interface/v2/scenario/1/instance/
                                                              48/activity/189",
            "id":189,
            "label":"Book journey",
            "state":"ready"},
        "6686":{
            "link":"http://domain:port/jengine/interface/v2/scenario/1/instance/
                                                                48/activity/6686",
            "id":6686,
            "label":"Cancel journey",
            "state":"ready"
        },
        "ids":[189,6686]
}
```

Example 8: Example of get all activities

Get Activities with State
http GET `interface/{version}/scenario/1/instance/48/ activity?state=ready`

```
{
    "activities":{
        "189":{
            "link":"http://domain:port/jengine/interface/v2/scenario/1/instance/
                                                                  48/activity/189",
            "id":189,
            "label":"Book journey",
            "state":"ready"},
        "6686":{
            "link":"http://domain:port/jengine/interface/v2/scenario/1/instance/
                                                                 48/activity/6686",
            "id":6686,
            "label":"Cancel journey",
            "state":"ready"
        },
        "ids":[189,6686]
}
```

Example 9: Example of get all activities with an specific state

Get Activities with Filter
http GET `interface/{version}/scenario/1/instance/48/activity?filter=Cancel`

```
{
    "activities":{
        "6686":{
            "link":"http://domain:port/jengine/interface/v2/scenario/1/instance/
                                                                 48/activity/6686",
            "id":6686,
            "label":"Cancel journey",
            "state":"ready"
        },
        "ids":[6686]
}
```

Example 10: Example of get all activities with an specific filter condition

Parameter	Options
state	ready, running, termianted
filter	a String

Table 2: Query Parameters and their options

Change the state of an activity instance

http PUT `interface/{version}/scenario/1/instance/48/activity?state=begin`

Parameter	Options
state	begin, terminate

Table 3: Query Parameters and their options

```
{
    "message":"activity state changed."
}
```

Example 11: Example output of begin activity

2.1.4 Data objects

This path provides methods to get information about Data Object instance used inside an scenario instance.

Get all Data Objects of an instance

http GET `interface/{version}/scenario/1/instance/48/ dataobject`

```
{
  "ids":[1,2],
  "results":{
      "1":{
        "link":"http://server:port/interface/v2/scenario/1/instance/72/dataobject/1",
        "id":1,
        "label":"Flight",
        "state":"payed"
      },
      "2":{
          "link":"http://server:port/interface/v2/scenario/1/instance/72/dataobject/2",
          "id":2,
          "label":"Reservation",
          "state":"init"
      }
  }
}
```

Example 12: Example output of get all Dataobjects

Get detailed information about a Data Objects instance

http GET `interface/{version}/scenario/1/instance/48/ dataobject/1`

```
{
    "id":1,
    "label":"object1",
    "state":"Flight"
}
```

Example 13: Example output of get all Dataobjects

Terminations Bedingungen

http PUT `interface/{version}/scenario/1/instance/48/ activity?state=begin`

```
{
  "conditions":{
  "1":[{"data_object":"Flight","set_id":1,"state":"payed"},
      {"data_object":"Reservation","set_id":1,"state":"payed"}]
  "2":[{"data_object":"Flight","set_id":2,"state":"canceled"},
      {"data_object":"Reservation","set_id":2,"state":"canceled"}]
  },
  "setIDs":[1,2]
}
```

Example 14: Example output of get all Dataobjects

Get Output

http GET `interface/{version}/scenario/1/instance/2/activity/8587/output`

```
[
    {
        "id":209,
        "linkDataObject":"http://172.16.64.113:8080/JEngine/api/interface/v2/scenario/160/in
        /1406/outputset/209"
    }
]
```

Example 15: Example output of get output for activities

Get References

http GET `interface/{version}/scenario/1/instance/2/activity/8587/references`

```
{
    "ids":[],
    "activities":[]
}
```

Example 16: Example output of get references for activities

Get Outputset

http GET `interface/{version}/scenario/1/instance/2/activity/outputset/209`

```
[
    {
        "label":"Bestellung",
        "id":1184,
        "state":"init",
        "attributeConfiguration":
            [
                {
                    "id":298,
                    "name":"Preis",
                    "type":"",
                    "value":""
                },
                {
                    "id":299,
                    "name":"Artikel",
                    "type":"",
                    "value":""
                }
            ]
    }
]
```

Example 17: Example output of get outputset

2.2 JConfiguration alias Config

2.2.1 Service eMail Tasks

Get all available email tasks

http GET `config/{version}/scenario/148/emailtask/`

```
{
    "ids":
        [
            448
        ]
}
```

Example 18: Example output of get all available email tasks

Get details for a specific email task

http GET `config/{version}/scenario/1/emailtask/448/?`

```
{
    "receiver":"bp2014w1@byom.de",
    "subject":"Test",
    "message":"Test Message"
}
```

Example 19: Example output of get all available email tasks

PUT changes for an email task

http PUT `config/{version}/emailtask/448/?`

```
{
    "receiver":"bp2014w1@byom.de",
    "subject":"Test",
    "message":"Test Message"
}
```

Example 20: Example output of PUT request to change an email task

2.2.2 Scenarios

Delete an scenario

http DELETE `config/{version}/scenario/156/?`

2.2.3 Webservice Tasks

GET all webservice tasks

http GET `config/{version}/scenario/1/webservice?`

```
{
    "ids":[527]
}
```

Example 21: Example output of get all webservice tasks for a specific scenario

GET specific webservice task

http GET `config/{version}/scenario/1/webservice/253?`

```
{
    "body":"",
    "link":"http://localhost:9998/interface/v2/scenario/156/instance",
    "method":"POST",
    "attributes":
        [
            {
                "order":1,
                "controlnode_id":527,
                "key":"id",
                "dataattribute_id":14
            }
        ]
}
```

Example 22: Example output of get specific webservice tasks for a specific scenario

PUT specific webservice task

http PUT `config/{version}/webservice/253?`

```
{
"method":"POST",
"link":"http://localhost:9998/interface/v2/scenario/156/instance",
"attributes":
    [
        {
            "order":1,
            "controlnode_id":527,
            "key":"id",
            "dataattribute_id":14
        },
        {
            "order":2,
            "controlnode_id":527,
            "key":"link",
            "dataattribute_id":15
        }
    ]
}
```

Example 23: Example output of get specific webservice tasks for a specific scenario

2.3 JComparser

2.3.1 Scenarios

Get all available scenarios for import
http GET `jcomparser/scenarios`

```
{
    "ids":
        {
            "800682779972007735":"Coffee process",
            "635432698":"Coffee process",
            "932221086":"Kaffeeprozess"
        }
}
```

Example 24: Example output of get all available scenarios from jcomparser

Get image of scenario
http GET `jcomparser/scenarios/932221086/image/`

```
\missingfigure{example scenario image}
```

Example 25: Example output of get activies log

POST launch import of scenario
http POST `jcomparser/scenarios/932221086/`

2.4 JHistory alias History

2.4.1 Activities

Get activities log

http GET `history/{version}/scenario/149/instance/1085/activities/`

```
{
    ”881”:
        {
            ”activityinstance_id”:7871,
            ”timestamp”:”2015−03−17 16:26:29.0”,
            ”scenarioinstance_id”:1085,
            ”label”:”Kaffeestand pruefen”,
            ”newstate”:”init”
        },
    [..]
}
```

Example 26: Example output of get activies log

2.4.2 Data Objects

Get dataobjects log

http GET `history/{version}/scenario/149/instance/1085/dataobjects/`

```
{
    ”68”:
        {
            ”timestamp”:”2015−03−17 16:26:56.0”,
            ”scenarioinstance_id”:1085,
            ”dataobjectinstance_id”:879,
            ”new_state_name”:”fertig”,
            ”name”:”Kaffee”,
            ”old_state_id”:120,
            ”old_state_name”:”gekocht”,
            ”new_state_id”:119
        },
    [..]
}
```

Example 27: Example output of get dataobjects log

2.4.3 Attributes

Get attribute log

http GET `history/{version}/scenario/149/instance/1085/attributes/`

```
{
    ”1”:
        {
            ”h.scenarioinstance_id”:1,
            ”da.name”:”Preis”,
            ”h.timestamp”:”2015-05-07 09:43:33.0”,
            ”h.id”:1,
            ”h.dataattributeinstance_id”:1,
            ”h.newvalue”:””,
            ”do.name”:”Bestellung”
        },
}
```

Example 28: Example output of get activies log

2.5 JAnalytics

2.5.1 Analytic Algorithms

The names of all registered algorithms executable can be obtained via

```
http GET analytics/{version}/services
```

POST analysis

The Algorithm can be started via the REST-Interface using

```
http POST analytics/{version}/services/
          de.uni_potsdam.hpi.bpt.bp2014.janalytics.{classNameOfAlgorithm}
```

If the Algorithm expects any parameters you have to put them in the request body as JSON Array with your POST. You have to keep the order in mind, the request body is the later input for the calculateResult(String[] args) method.

After the POST-Request a http 303 See Other with a new URI is returned. The URI (/jengine/api/analytics/v2/services/de.uni_potsdam.hpi.bpt.bp2014.janalytics.{class NameOfAlgorithm}/resultId/{resultId}) can be used for a GET-Request to obtain the result of the calculation, the resultId is used to identify the result of the different POSTs.

Get analysis results

After the Algorithm has terminated, the results can be obtained via

```
http GET analytics/{version}/services/
         de.uni_potsdam.hpi.bpt.bp2014.janalytics.{classNameOfAlgorithm}/
         resultId/{resultId}
```

to the same URL as in the 303 See Other and returns a JSON with the output of the Algorithm.

It should be considered, that the Angular.js frontend will automatically send a GET Request to the URI of the http 303 See Other and no additional GET Request is needed when the output of an algorithm is integrated into the frontend.

```
{
    "scenarioId":"163","meanScenarioInstanceRuntime":"0d:0h:3m:17s"
}
```

Example 29: Example output of get request for Example-Algorithm (meanScenarioInstanceRuntime)