

# A Generalized Swagger (OpenAPI) Centered Web Services Testing Framework

Mohsen BANAN

Email: <http://mohsen.1.banan.byname.net/contact>

<http://www.by-star.net/PLPC/180057>

October 23, 2018

Verbatim Copying Permitted

# Outline of Part I – Overview Of mmwslcm Web Services Testing Framework

- 1 Invokers Development Model
  - Invoker ICMs Development Model
  - Invoker Framework: Ingredients

# Invoker ICMs Development Model

Given a Service Definition (a swagger file) and a Performer Server, you should be able to conveniently Invoke any of the offered Operations through:

- 1 swagger.ui interface – usually offered by the Performer Server
- 2 unisos.mmwslcm :: rinvoker.py – command line and batch oriented equivalent of swagger.ui
- 3 unisos.mmwslcm :: ro.py – invoke-specification – invoke-verification – invoke-reporting
- 4 unisos.mmwslcm – rich library for building invoker Apps

# Remote Invoker ICMs Development Model

Main software packages that implement the framework include:

- 1 Python Bravado – Equivalent of Invoker Codegenartor But Better
- 2 unisos.icm – Interactive Command Module
  - Makes icm.Cmnd classes invocable at command-line
  - do-icm :: Direct Operation ICMs (Used by performers)
- 3 unisos.mmwslcm
  - unisos.mmwslcm.wsInvoker.py – Maps invocations to http requests
  - unisos.mmwslcm.ro.py – Abstracts invoke-specifications
  - unisos.mmwslcm.rinvoker.py – Maps command-line args to invocations

Try It: `pip install unisos.mmwslcm`

# Outline of Part II – rinvoker

- 2 rinvoker.py
  - rinvoker.py Overview

# rinvoker.py Overview

Allows you to list all possible invocations based on a service specification (swagger file).

```
rinvoker.py --svcSpec="http://petstore.swagger.io/v2/swagger.json"
```

Allows you to fully specify an invocation on command line. Example:

```
rinvoker.py --svcSpec="http://petstore.swagger.io/v2/swagger.json"  
            --resource="user" --opName="createUser" -i rinvoker  
            bodyStr="{...}"
```

# Outline of Part III – Operation Scenarios

- 3 Invoke-Specifications, Invoke-Verification, Invoke-Reporting
  - Model Of Invoke – Specification, Verification And Reporting – Scenarios
  - Scenario Specification For Sequences Of Invocations
  
- 4 Invoker-Apps Development Model
  - Invoker-Apps Can Easily Build On unisos.mmwslcm Capabilities
  
- 5 Benefits And Advantages Of The Generalized Swagger Centered Invocation Model
  - Taking Full Advantage Of Service Specification For Testing And Development

# Model Of Invoke – Specification, Verification And Reporting – Scenarios

- Invoke Scenarios Are pure python specification of sequence of invocations.
- Invoke-Expect Scenarios Are pure python specification of sequence of invocations subject to preparations and post-invoke verification and reporting.
- opInvoke class allows for complete invoke specification and complete results to be fully captured.



# Scenario Specification For Sequences Of Invocations

In pure python specify invocation of each operation, for example:

```
rosList.opAdd(  
    svcSpec=akSkToken_svcSpec,    # e.g. "../SvcSpec/akSkToken.json"  
    perfSap=akSkToken_perfSap,    # e.g. "http://localhost:8080/access  
    resource="token",  
    opName="obtain",  
    roParams=ro.Ro_Params(  
        headerParams=None,  
        urlParams=None,  
        bodyParams={  
            'accessKey': "someOtherId",  
            'secretKey': "someOtherCredentials",},),  
    roResults=None,  
)
```

# Validation And Reporting Of Invocations

Building on the previously mentioned Operation Specification, in pure python you can the specify Operation Expectations, for example:

```
roExpectationsList.opExpectationAdd(  
    # Above mentioned operation specification  
    # ...  
    preInvokeCallables=[],  
    postInvokeCallables=[ verify_akSkTokenObtainRo, ],  
)
```

For example, `preInvokeCallables(ro.Ro_OpExpectation)` can include a function that initializes the DB.

For example, `postInvokeCallables(ro.Ro_OpExpectation)` can include a function that verifies the operation's result was as expected and then reports success or failure.

# Invoker-Apps Can Easily Build On unisos.mmwslcm Capabilities

- Bravado does invoker code-generation on the fly.
- `unisos.mmwslcm.oplInvoke` – Abstracts invoke-specifications
- `unisos.mmwslcm.wslInvoker` – Allows for invokation and verification of `oplInvoke`

With these in place, building Invoke-Apps becomes very simple.

# Benefits And Advantages Of The Generalized Swagger Centered Invocation Model

- The Generalized Model and Capabilities Presented Here Apply To Any Service That Exposes Its Swagger Specifications
- A great deal of automation capabilities have become possible based on swagger specifications.
- The Testing Framework Of (invoke-specification, invoke-verification and invoke-reporting) permits for disciplined complete external testing of web-services based on swagger specifications.

Very Often, These Best Current Practices Are Not Being Followed.