

OSLC FMI Adapter

Version

1.1.0

Date

24 July 2016

Authors

Vorachet Jaroensawas, Koneksys (vorachet.jaroensawas@koneksys.com)

Axel Reichwein, Koneksys (axel.reichwein@koneksys.com)

Table of Contents

1. Summary	3
2. Understanding OSLC FMI Adapter Components	3
3. Building OSLC FMI Adapter	4
3.1 Building and Testing C component (optional)	4
3.1.1 To build FMIClient from a terminal command line in Linux or Mac	4
To build FMI Library source	4
To run FMIClient build script for Mac	5
To run FMIClient build script for Linux	5
To install the executable files of the FMIClient into OSLC FMI adapter web project	5
3.1.2 To build FMIClient from the Developer Command Prompt for VS2013 in Windows	6
To run FMIClient build script for Windows	6
To install the executable files of FMIClient into OSLC FMI adapter java project	6
3.1.3 Run a quick test for the FMIClient from a terminal command line	7
3.2 Building and Testing Java components	8
3.2.1 Installing OSLC4J	8
3.2.2 Importing projects into the Eclipse workspace	8
3.2.3 Building the oslc4j-fmi using oslc4j-fmi-build project	8
3.2.4 Running the OSLC FMI Adapter with sample FMUs	9
Option1: Running the OSLC FMI Adapter using Embedded Tomcat	9
Option2: Deploying the OSLC FMI Adapter on a Tomcat server	9
Configuring repository of FMU files	10
The settings of conf.properties	10
3.2.5 Testing the OSLC FMI Adapter Manually	11
Testing the retrieval of OSLC resources in HTML	11
Testing the retrieval of OSLC resources in RDF	11
3.2.6 Testing the OSLC FMI Adapter using JUnit	11
Running JUnit Test Suite of OSLC FMI Adapter	11
4. URI Specifications	12
5. OSLC FMI Adapter Entity Classes	15
6. Third party libraries	16

1. Summary

OSLC4j-FMI is intended as a OSLC adapter for publishing FMUs (Functional Mockup Units) that follow OSLC Standard. This version of the library supports FMI 1.0 and FMI 2.0. See <http://www.fmi-standard.org>

2. Understanding OSLC FMI Adapter Components

OSLC4J-FMI consists of two parts.

C Component

The OSLC FMI adapter C component uses VISUAL STUDIO COMMUNITY 2013 (see <https://www.visualstudio.com/downloads/download-visual-studio-vs>) to compile a C program used to interface FMUs for Windows and use GCC for Linux or Mac.

List of C project

oslc4j-fmi-cinterface

Contains FMIClient source and build scripts, FMI Library source and Windows binaries.

Java Components

The OSLC FMI adapter java components use Maven (see <http://maven.apache.org>) for a build automation tool.

List of Java projects

oslc4j-fmi-build

Top-level maven build script

oslc4j-fmi-ecore

ECore definition for the OSLC FMI adapter entity classes

oslc4j-fmi-resources

Java entity classes for the OSLC FMI adapter which are generated by *oslc4j-fmi-ecore*

oslc4j-fmi-clients

JUnit tests and example test clients for the OSLC FMI adapter

oslc4j-fmi

Java web component for the OSLC FMI adapter

3. Building OSLC FMI Adapter

3.1 Building and Testing C component (optional)

This step is optional because all executable files of the FMIClient on each platform are installed in the following folder. The following directory structure and FMIClient must be available in the `com.modelon.oslc.adapter.fmi` java project before running the OSLC FMI adapter. By default, the latest version of the FMIClient on each platform is provided.

```
com.modelon.oslc.adapter.fmi
/conf
/fmi
/bin
  /darwin64
    FMIClient
    fmlib_shared.dylib
  /linux64
    FMIClient
    libfmlib_shared.so
  /win32
    FMIClient.exe
    msvcr120d.dll
    fmlib_shared.dll
  /win64
    FMIClient.exe
    msvcr120d.dll
    fmlib_shared.dll
```

If you want to build FMIClient from source, use instructions in the next section.

3.1.1 To build FMIClient from a terminal command line in Linux or Mac

CMake (see <http://www.cmake.org>) is used to generate the native build scripts for the FMI Library. CMake 3.3.0 is tested in development of this project on Linux and Mac. For Windows, we use FMI Library binary directly from <http://www.fmi-standard.org>. The recommended version of the FMI Library binaries for Windows and source is 2.0.2b3 which are already prepared in `com.modelon.oslc.adapter.fmi.cinterface/fmi` directory of project.

To build FMI Library source

Download FMIL 2.0.2b3 source file from <http://www.jmodelica.org/downloads/FMIL/FMILibrary-2.0.2b3-src.zip> and extract to `com.modelon.oslc.adapter.fmi.cinterface/fmi/FMILibrary-2.0.2b3-src`

```
git clone https://bitbucket.org/koneksys/oslc-adapter-fmi
cd com.modelon.oslc.adapter.fmi.cinterface
cd fmi
cd FMILibrary-2.0.2b3-src
mkdir build
cd build
cmake -DFMILIB_INSTALL_PREFIX=./install ../
make install test
```

When you successfully build the FMI library, the following folder will be created.

```
com.modelon.oslc.adapter.fmi.cinterface
  fmi\
    FMILibrary-2.0.2b3-src\
      install\
        doc/*
        include/*
        lib/*
```

To run FMIClient build script for Mac

```
cd com.modelon.oslc.adapter.fmi.cinterface/src/FMIClient
./build.mac.sh
```

The output of the build script build.mac.sh will be

```
com.modelon.oslc.adapter.fmi.cinterface
/src
  /FMIClient
    FMIClient
    fmlib_shared.dylib
```

To run FMIClient build script for Linux

```
cd com.modelon.oslc.adapter.fmi.cinterface/src/FMIClient
./build.linux.sh
```

The output of the build script build.linux.sh will be

```
com.modelon.oslc.adapter.fmi.cinterface
/src
  /FMIClient
    FMIClient
    libfmlib_shared.so
```

To install the executable files of the FMIClient into OSLC FMI adapter web project

For Mac

```
com.modelon.oslc.adapter.fmi
/conf
  /fmi
    /bin
      /darwin64
        FMIClient
        fmlib_shared.dylib
```

For Linux 64 bit

```
com.modelon.oslc.adapter.fmi
/conf
  /fmi
    /bin
      /linux64
        FMIClient
        libfmlib_shared.so
```

3.1.2 To build FMIClient from the Developer Command Prompt for VS2013 in Windows

VISUAL STUDIO COMMUNITY 2013 is available to download at <https://www.visualstudio.com/downloads/download-visual-studio-vs> . To prevent common errors in calling cl.exe of the Visual Studio, to use default installation path at "C:\Program Files\Microsoft Visual Studio 12.0" is recommended and make sure Visual C++ and its build tool is available in "C:\Program Files\Microsoft Visual Studio 12.0\VC". The Developer Command Prompt for VS2013 is available in Program Menu > Visual Studio 20130 > Visual Studio Tools

To run FMIClient build script for Windows

```
cd com.modelon.oslc.adapter.fmi.cinterface\src\FMIClient
build.bat
```

The output of the build script build.bat will be

```
com.modelon.oslc.adapter.fmi.cinterface
/src
/FMIClient
  FMIClient.exe
  FMIClient.obj
  msucr120d.dll
  fmlib_shared.dll
```

To install the executable files of FMIClient into OSLC FMI adapter java project

For Windows 32bit

```
com.modelon.oslc.adapter.fmi
/conf
/fmi
/bin
/win32
  FMIClient.exe
  msucr120d.dll
  fmlib_shared.dll
```

For Windows 64bit

```
com.modelon.oslc.adapter.fmi
/conf
/fmi
/bin
/win64
  FMIClient.exe
  msucr120d.dll
  fmlib_shared.dll
```

3.1.3 Run a quick test for the FMIClient from a terminal command line

FMIClient test command line:

```
FMIClient read <FMUFilePath> <TempFolder>
```

Example of FMIClient test command line:

```
FMIClient read FMI1.0_CS_CoupledClutches.fmu c:\temp\  
FMIClient read FMI2.0_CS_PID_Controller.fmu c:\temp\
```

Notes:

- <FMUFilePath> is relative or absolute path of a FMU
- <TempFolder> must be created before running the command line

3.2 Building and Testing Java components

Eclipse IDE for Java EE Developers is recommended for building the OSLC FMI adapter Java components and running development mode of the adapter. The Eclipse IDE is available to download at <http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/mars2>

3.2.1 Installing OSLC4J

1. Follow the instruction to install Eclipse Lyo at <https://github.com/ld4mbse/oslc4j>

3.2.2 Importing projects into the Eclipse workspace

1. In Eclipse, import projects from Git. (File → Import → Git → Projects from Git)
2. Click on the Clone Repository icon
3. In the URI field, paste the following URL: <https://bitbucket.org/koneksys/oslc-adapter-fmi>
4. The Host and Repository fields will autofill.
5. Click Next, only select the master branch
6. Click Next until Finish.
7. Select a wizard to use for importing projects, only select Import existing Eclipse projects
8. There are 6 projects including oslc4j-fmi, oslc4j-fmi-build, oslc4j-fmi-cinterface, oslc4j-fmi-ecore, oslc4j-fmi-resources and oslc4j-fmi-clients. They are in the list to be imported.
9. Select all 6 projects and click Finish
10. The 5 projects are in the Eclipse Project Explorer

3.2.3 Building the oslc4j-fmi using oslc4j-fmi-build project

1. In Eclipse, open the Project Explorer view. (Window → Show View → Project Explorer)
2. Expand the oslc4j-fmi-build project
3. Right click pom.xml -> Run As -> Maven clean
4. Right click pom.xml -> Run As -> Maven install

3.2.4 Running the OSLC FMI Adapter with sample FMUs

The OSLC FMI Adapter provides sample FMU files for each platform which are prepared in the Eclipse project.

Option1: Running the OSLC FMI Adapter using Embedded Tomcat

The Development Mode allows you developing and testing the web application easily through development lifecycle using Eclipse and Maven.

You can also consider this mode as an easiest way to run the OSLC FMI Adapter with the sample FMU files is to run the adapter with the Apache Tomcat Maven Plugin and Embedded Tomcat which are pre-configured in the oslc4j-fmi project.

1. In Eclipse, open the Project Explorer view. (Window → Show View → Project Explorer)
2. Expand the oslc4j-fmi project
3. Right click pom.xml -> Run As -> Run Configurations
4. In Run Configurations Dialog, Right click Maven Build -> New
5. In the right hand side, "Name" text input, enter "Run OSLC FMI Adapter"
6. In the right hand side, "Base directory" text input, click "Browse Workspace" and click workspace "oslc4j-fmi". After the workspace is clicked, text "\${workspace_loc:/oslc4j-fmi}" should be appeared in the "Base directory" text input
7. In the right hand side, "Goals" text input, enter "tomcat7:run"
8. Click run (Once you create this Maven goal in Eclipse you can simply run it again in Eclipse Run menu icon or Menu Run → Run History)
9. After Tomcat is successfully started, open URL <http://localhost:8686/oslc4jfmi/services/catalog>

Option2: Deploying the OSLC FMI Adapter on a Tomcat server

This section describes steps to deploy the OSLC FMI Adapter with sample FMU files on a Tomcat server.

The differentiation between Development Mode Tomcat Port and Production Mode Tomcat Port: With Development Mode Tomcat, the OSLC FMI adapter service will run on your local machine using port 8686. for development purpose. When you want to deploy the adapter on Production Mode Tomcat, the port of OSLC FMI adapter service will be managed by the Tomcat server configuration. For example, if you running Tomcat on port 8080, so the OSLC FMI adapter service will run on port 8080.

1. In Eclipse, open the Project Explorer view. (Window → Show View → Project Explorer)
2. Expand the oslc4j-fmi project
3. Copy the folder named "fmi" under the "conf" folder (See section 5 for more detail of the configuration directory)
4. Paste the folder in the Tomcat installation directory (Example: C:\Program Files\apache-tomcat-8.0.24-windows-x64\apache-tomcat-8.0.24\fmi)
5. In Eclipse, expand the oslc4j-fmi project
6. Open the /target folder
7. Rename the oslc4j-fmi-1.1.0.war file into oslc4jfmi.war
8. Copy oslc4jfmi.war into the /webapps folder of the of the Tomcat installation directory
9. start Tomcat
10. Open URL <http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/catalog>
11. If you running Tomcat on localhost using port 8080, so the URL is <http://localhost:8080/oslc4jfmi/services/catalog>

Configuring repository of FMU files

The OSLC FMI Adapter allows you to configure the location of FMU files.

By default, the following configuration directory is prepared in the Eclipse project named `oslc4j-fmi`, so you can run the development mode Tomcat with do nothing. However, you can edit the configuration in `oslc4j-fmi/conf/fmi/conf/conf.properties` if you want to change the location of FMU files.

```
com.modelon.oslc.adapter.fmi
/conf
/fmi
/bin/*           (Binary of FMIClient on each platform)
/conf/conf.properties (Configuration file)
/fmu/*.fmu       (FMU files)
/temp/*          (FMU unzipped files)
```

In configuring the adapter on production using a standalone Tomcat server, the installation is similar to the Development Mode Tomcat but the configuration directory is located in the Tomcat Installation Directory.

```
<Tomcat Installation Directory>
/bin           (Tomcat bin folder)
/lib           (Tomcat lib folder)
/logs          (Tomcat logs folder)
/temp          (Tomcat temp folder)
/work          (Tomcat work folder)
/webapps       (Tomcat webapps folder)
/oslc4jfmi.war (The deployment package of OSLC FMI adapter)
/conf          (The configuration of OSLC FMI adapter)
/fmi
/bin/*         (Binary of FMIClient on each platform)
/conf/conf.properties (Configuration file)
/fmu/*.fmu     (FMU files)
/temp/*        (FMU unzipped files)
```

The settings of `conf.properties`

The OSLC FMI adapter will lookup the location of FMU files in a config file named `conf.properties`. The configuration contains only one variable.

```
repository = {Location of FMU repository directory}
```

Default value: `conf/fmi/fmu/`

Post-conditions:

- the value will be used to look up the location of FMU files

3.2.5 Testing the OSLC FMI Adapter Manually

Testing the retrieval of OSLC resources in HTML

1. Launch your web browser
2. In the URL field, If you are running the adapter with Development Mode Tomcat, type <http://localhost:8686/oslc4jfmi/services/catalog> . If you are running the adapter with Production Mode Tomcat using port 8080, type <http://localhost:8080/oslc4jfmi/services/catalog> . This will send a HTTP GET request to retrieve the HTML representation of the FMI Service Provider Catalog.
3. You will then see an HTML page showing you the list of Service Providers. You can browse from the Service Providers to the Services and ultimately to the OSLC FMU resources.

Testing the retrieval of OSLC resources in RDF

This sections requires a Rest client software to get OSLC resources in RDF. The recommended software is Postman that available to download at <https://www.getpostman.com>

The following testing guide uses Postman in Google Chrome browser.

1. Click on the Postman icon at the top right of the Chrome browser . A new tab will open.
2. In the URL field, If you are running the adapter with Development Mode Tomcat, type <http://localhost:8686/oslc4jfmi/services/catalog> . If you are running the adapter with Production Mode Tomcat using port 8080, type <http://localhost:8080/oslc4jfmi/services/catalog> .
3. Click on the Headers field to the right of the URL field
4. Enter Accept in the Header field and *application/rdf+xml* in the value field
5. Click Send
6. This will send a HTTP GET request to retrieve the RDF/XML representation of the sample FMU files as predefined in the configuration directory.

3.2.6 Testing the OSLC FMI Adapter using JUnit

Running JUnit Test Suite of OSLC FMI Adapter

1. In Eclipse, open the Project Explorer view. (Window → Show View → Project Explorer)
2. Make sure the adapter is running before running JUnit test Suite. (See section 3.2.4 option 1)
3. Expand the oslc4j-fmi-clients project
4. Expand “src/test/java” folder
5. Expand the package “com.modelon.oslc.adapter.fmi.clients.tests”
6. Right click AllTests.java -> Run As -> JUnit Test
7. This will run JUnit Test Suite in Eclipse and JUnit View will be opened and show the result after the test is completed.

4. URI Specifications

OSLC Service Catalog - Accept (text/html, application/rdf+xml)

GET <http://localhost:8686/oslc4jfmi/services/catalog>
GET <http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/catalog>

OSLC Service Providers - Accept (text/html, application/rdf+xml)

GET <http://localhost:8686/oslc4jfmi/services/serviceProviders>
GET <http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/serviceProviders>

OSLC Service Provider - Accept (text/html, application/rdf+xml)

GET http://localhost:8686/oslc4jfmi/services/serviceProviders/{FMIFMU_Identifier}
GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/serviceProviders/{FMIFMU_Identifier}

OSLC FMI Resource Shapes - Accept (text/html, application/rdf+xml)

GET <http://localhost:8686/oslc4jfmi/services/fmi/resourceShapes>
GET <http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/fmi/resourceShapes>

FMI RDF vocabulary - Accept (text/html, application/rdf+xml)

GET <http://localhost:8686/oslc4jfmi/services/rdfvocabulary>
GET <http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/fmi/rdfvocabulary>

FMI Concept: FMU

OSLC Query Capability - Accept (text/html, application/rdf+xml)

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/FMU
GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/FMU/{FMU_Identifier}

FMI Concept: ModelDescription

OSLC Query Capability - Accept (text/html, application/rdf+xml)

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ModelDescription
GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ModelDescription/{ModelDescription_Identifier}

FMI Concept: DefaultExperiment

OSLC Query Capability - Accept (text/html, application/rdf+xml)

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/DefaultExperiment
GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/DefaultExperiment/{DefaultExperiment_Identifier}

FMI Concept: ModelExchange

OSLC Query Capability - Accept (text/html, application/rdf+xml)

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ModelExchange
GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ModelExchange/{ModelExchange_Identifier}

FMI Concept: CoSimulation

OSLC Query Capability - Accept (text/html, application/rdf+xml)

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/CoSimulation
GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/CoSimulation/{CoSimulation_Identifier}

FMI Concept: SourceFile

OSLC Query Capability - Accept (text/html, application/rdf+xml)

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/SourceFile
GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/SourceFile/{SourceFile_Identifier}

FMI Concept: UnitDefinition

OSLC Query Capability - Accept (text/html, application/rdf+xml)

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/UnitDefinition
GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/UnitDefinition/{UnitDefinition_Identifier}

FMI Concept: BaseUnit**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/BaseUnit

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/BaseUnit/{BaseUnit_Identifier}

FMI Concept: BaseUnitDefinition**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/BaseUnitDefinition

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/BaseUnitDefinition/{BaseUnitDefinition_Identifier}

FMI Concept: VendorAnnotation**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/VendorAnnotation

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/VendorAnnotation/{VendorAnnotation_Identifier}

FMI Concept: VendorTool**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/VendorTool

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/VendorTool/{VendorTool_Identifier}

FMI Concept: Annotation**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/Annotation

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/Annotation/{Annotation_Identifier}

FMI Concept: TypeDefinition**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/TypeDefinition

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/TypeDefinition/{TypeDefinition_Identifier}

FMI Concept: TypeDef**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/TypeDef

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/TypeDef/{TypeDef_Identifier}

FMI Concept: Enumeration**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/Enumeration

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/Enumeration/{Enumeration_Identifier}

FMI Concept: ModelVariables**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ModelVariables

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ModelVariables/{ModelVariables_Identifier}

FMI Concept: ScalarVariable**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ScalarVariable

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ScalarVariable/{ScalarVariable_Identifier}

FMI Concept: ScalarVariableType**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ScalarVariableType

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ScalarVariableType/{ScalarVariableType_Identifier}

FMI Concept: ListingAlias**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ListingAlias

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ListingAlias/{ListingAlias_Identifier}

FMI Concept: ListingDirectDependency**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ListingDirectDependency

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ListingDirectDependency/{ListingDirectDependency_Identifier}

FMI Concept: ModelStructure**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ModelStructure

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/ModelStructure/{ModelStructure_Identifier}

FMI Concept: Derivative**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/Derivative

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/Derivative/{Derivative_Identifier}

FMI Concept: InitialUnknown**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/InitialUnknown

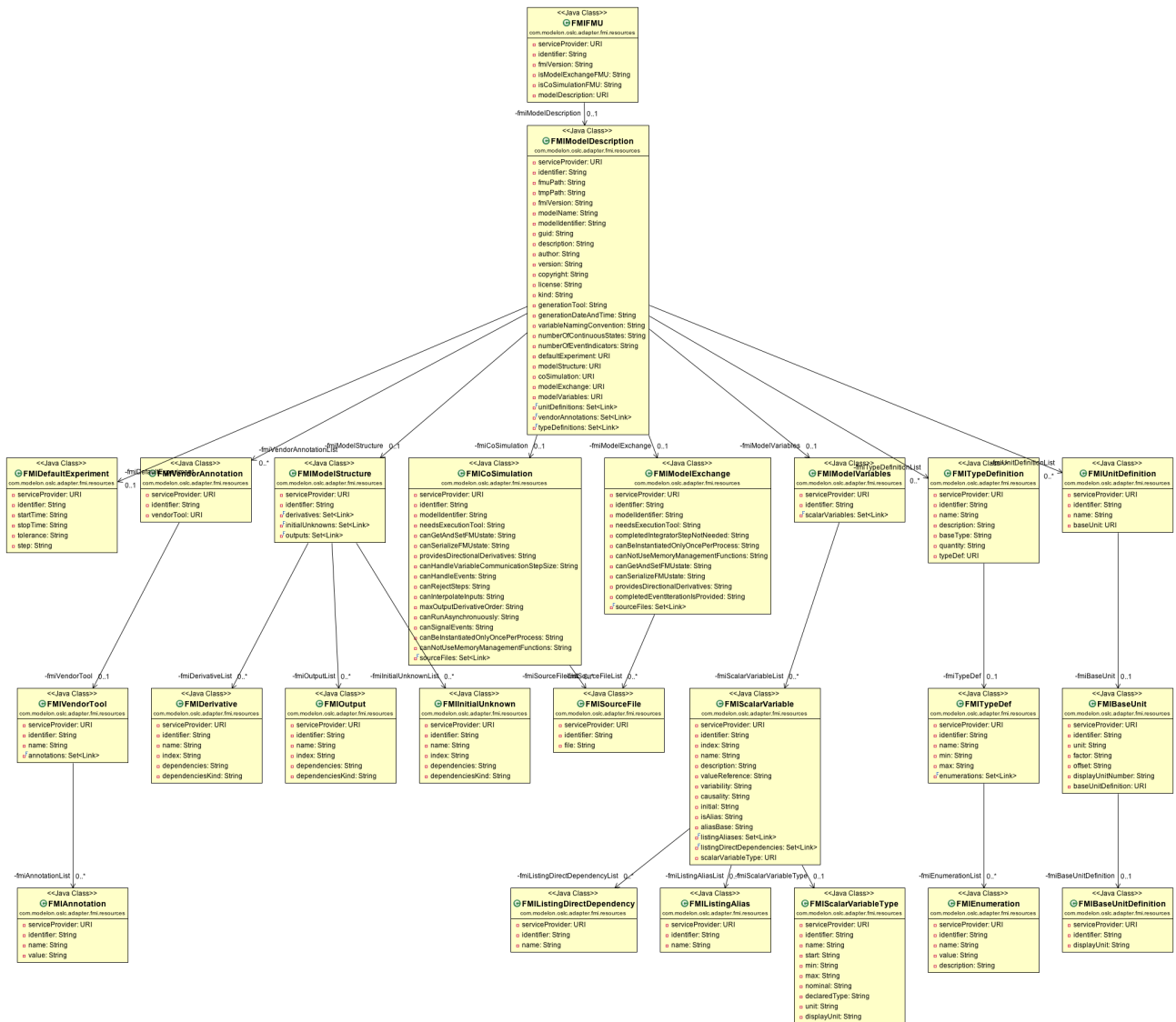
GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/InitialUnknown/{InitialUnknown_Identifier}

FMI Concept: Output**OSLC Query Capability - Accept (text/html, application/rdf+xml)**

GET http://localhost:8686/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/Output

GET http://<TomcatHost>:<TomcatPort>/oslc4jfmi/services/Resource/{FMIFMU_Identifier}/Output/{Output_Identifier}

5. OSLC FMI Adapter Entity Classes



6. Third party libraries

The OSLC FMI adapter utilises code from third party libraries. The respective copyright information and licenses are listed below.

The third party libraries for OSLC FMI adapter runtime

- **Eclipse Public License and the Eclipse Distribution License :**
 - OSLC4J Core 3.0.0-SNAPSHOT (<https://github.com/eclipse/lyo.core>)
 - OSLC4J JSON4J Provider 3.0.0-SNAPSHOT (<https://github.com/eclipse/lyo.core/tree/master/org.eclipse.lyo.oslc4j.provider.json4j>)
 - OSLC4J Jena Provider 3.0.0-SNAPSHOT (<https://github.com/eclipse/lyo.core/tree/master/org.eclipse.lyo.oslc4j.provider.jena>)
 - OSLC4J Query 3.0.0-SNAPSHOT (<https://github.com/eclipse/lyo.core/tree/master/org.eclipse.lyo.core.query>)
 - OSLC4J Wink Application 3.0.0-SNAPSHOT (<https://github.com/eclipse/lyo.core/blob/master/org.eclipse.lyo.oslc4j.wink/pom.xml>)
- **CDDL 1.0:**
 - [javax.ws.rs](https://www.java.ws.rs) 1.1.1 (<https://jsr311.java.net>)
- **The Apache Software License, Version 2.0:**
 - Jackson-annotations 2.7.0 (<https://github.com/FasterXML/jackson>)
 - Jackson-core 2.7.0 (<https://github.com/FasterXML/jackson-core>)
 - Jackson-databind 2.7.0 (<https://github.com/FasterXML/jackson-databind>)

The library for OSLC FMI adapter C interface runtime

- **BSD License:**
 - FMILibrary 2.0.2b3 (<http://www.jmodelica.org/FMILibrary>)