Variant with Genetic Algorithm (GA)

Manuscript for Windows 10 64-bits

Authors:

- **Habibi Husain Arifin**
- **Ho Kit Robert Ong**

Requirements:

- **Software and Plugin:**
    - **Windows 10 64-bit (for this moment)**
    - **Internet Browser such as Mozilla Firefox or Google Chrome**
    - **Cameo Systems Modeler 19.0**
        - **Link: www.nomagic.com**
    - **Anaconda with Python 3.7**
        - **Link: https://www.anaconda.com/download/**
    - **Jupyter Notebook 5.6.0**
        - **Link: https://jupyter.org/install**
- **Python Library:**
    - **DEAP (Distributed Evolutionary Algorithms in Python) 1.2.2. DEAP is the Genetic Algorithm python library which is used in this project.**
        - **Link: https://anaconda.org/conda-forge/deap**
        - **Manual: https://deap.readthedocs.io/en/master/**
    - **Graphviz 3.6. It is requisite because Pydot is dependent with this library.**
        - **Installation steps: https://stackoverflow.com/questions/45093811/installing-pygraphviz-on-windows-10-64-bit-python-3-6**
        - **Link Graphiz for Windows 2.38: https://graphviz.gitlab.io/_pages/Download/Download_windows.html**
        - **Link Graphviz Wheel 0.10.1: https://pypi.org/project/graphviz/#files**
    - **Pydot 1.3.0. Pydot is the python library to illustrate the root-node-leaf tree.**
        - **Link: https://anaconda.org/anaconda/pydot**
    - **The Element Tree XML API for Python 3.7.2. (No need to install it with pip, it is in the stdlib).**
        - **Manual: https://docs.python.org/3.7/library/xml.etree.elementtree.html**
    - **StringGenerator 0.3.1**
        - **Link: https://pypi.org/project/StringGenerator/**
    - **RegularExpression RE for Python 3.7.2 (No need to install it with pip, it is in the stdlib).**
        - **Manual: https://docs.python.org/3/library/re.html**
    - **Matplotlib 3.0.2.**
        - **Link: https://anaconda.org/conda-forge/matplotlib**
        - **Manual: https://matplotlib.org/contents.html**
    - **Networkx 2.2**

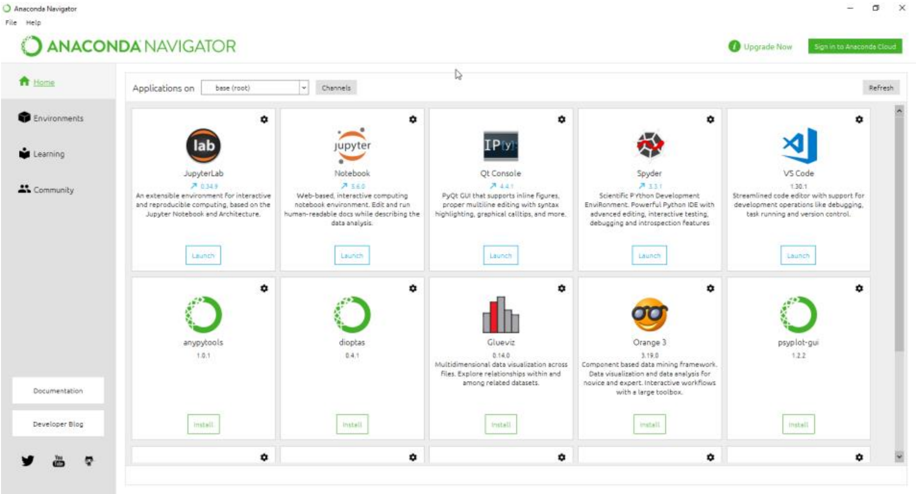
        - **Link: https://anaconda.org/anaconda/networkx**
        - **Manual: http://networkx.github.io/**
    - **Js2Py 0.60**
        - **Link: https://pypi.org/project/Js2Py/**
        - **Manual**

Step 1: Installation Steps (Anaconda and Jupyter Notebook):

1. **Install Anaconda with Python 3.7**
   Use the installer that you have downloaded from the official website.
2. **Install Jupyter Notebook 5.6.0**
   To run the notebook, run the following command at the Command Prompt (Windows):

```
jupyter notebook
```

3. **You can start Jupyter Notebook with two ways:**
    - **Start from Anaconda menu (click Launch):**



JupyterLab
0.34.9
An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook
5.6.0
Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Qt Console
4.4.1
PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Spyder
3.3.1
Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

VS Code
1.30.1
Streamlined code editor with support for development operations like debugging, task running and version control.

anypytools
1.0.1

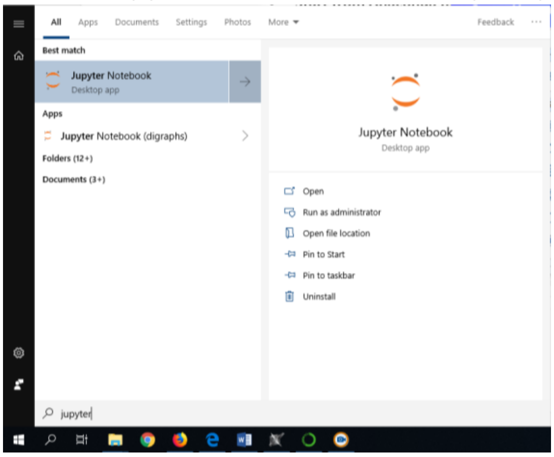
dioptas
0.4.1

Glueviz
0.14.0
Multidimensional data visualization across files. Explore relationships within and among related datasets.

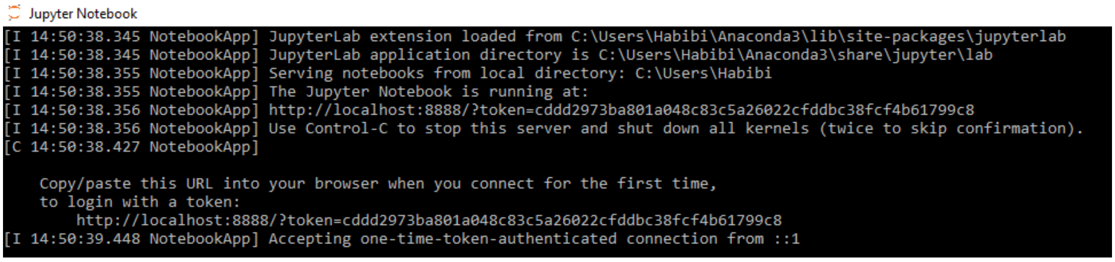
Orange 3
3.19.0
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

psyplot-gui
1.2.2

    - **Search “Jupyter Notebook” command prompt in Windows Start Menu:**

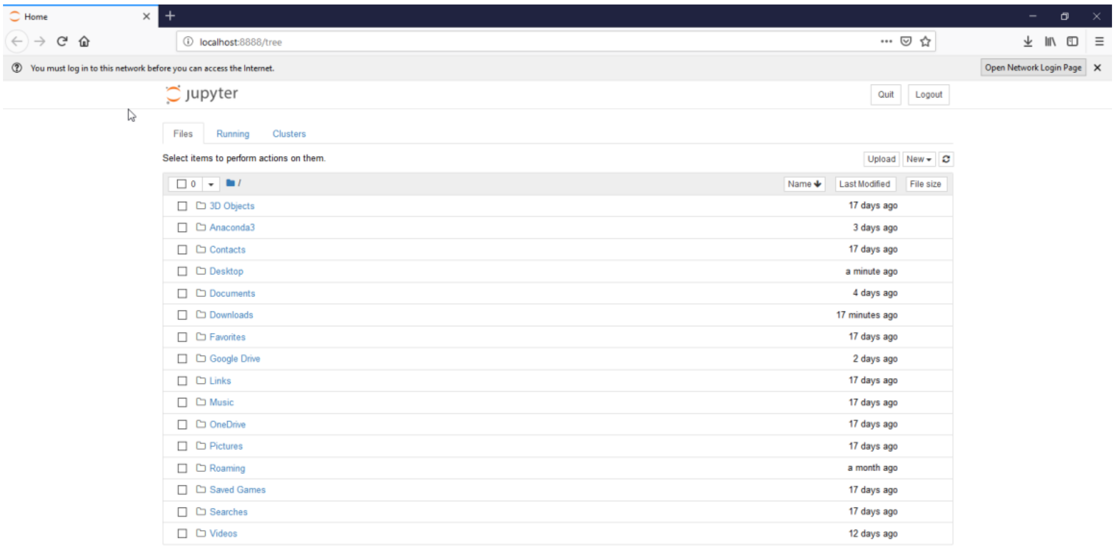


4. **As soon as you start Jupyter Notebook, you will be redirect to your default browser (in my case, it is mozilla firefox)**



```
[I 14:50:38.345 NotebookApp] JupyterLab extension loaded from C:\Users\Habibi\Anaconda3\lib\site-packages\jupyterlab
[I 14:50:38.345 NotebookApp] JupyterLab application directory is C:\Users\Habibi\Anaconda3\share\jupyter\lab
[I 14:50:38.355 NotebookApp] Serving notebooks from local directory: C:\Users\Habibi
[I 14:50:38.355 NotebookApp] The Jupyter Notebook is running at:
[I 14:50:38.356 NotebookApp] http://localhost:8888/?token=cddd2973ba801a048c83c5a26022cfddbc38fcf4b61799c8
[I 14:50:38.356 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 14:50:38.427 NotebookApp]

    Copy/paste this URL into your browser when you connect for the first time,
    to login with a token:
        http://localhost:8888/?token=cddd2973ba801a048c83c5a26022cfddbc38fcf4b61799c8
[I 14:50:39.448 NotebookApp] Accepting one-time-token-authenticated connection from ::1
```

5. **Jupyter Notebook is ready to use.**

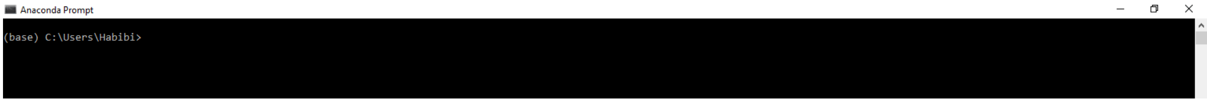


Select items to perform actions on them.

Name	Last Modified	File size
□ 3D Objects	17 days ago	
□ Anaconda3	3 days ago	
□ Contacts	17 days ago	
□ Desktop	a minute ago	
□ Documents	4 days ago	
□ Downloads	17 minutes ago	
□ Favorites	17 days ago	
□ Google Drive	2 days ago	
□ Links	17 days ago	
□ Music	17 days ago	
□ OneDrive	17 days ago	
□ Pictures	17 days ago	
□ Roaming	a month ago	
□ Saved Games	17 days ago	
□ Searches	17 days ago	
□ Videos	12 days ago	

Step 2: Installation Steps (Python Library):

1. **Open Anaconda Prompt from Windows Start Menu. You will be redirect to Anaconda terminal.**



```
(base) C:\Users\Habibi>
```

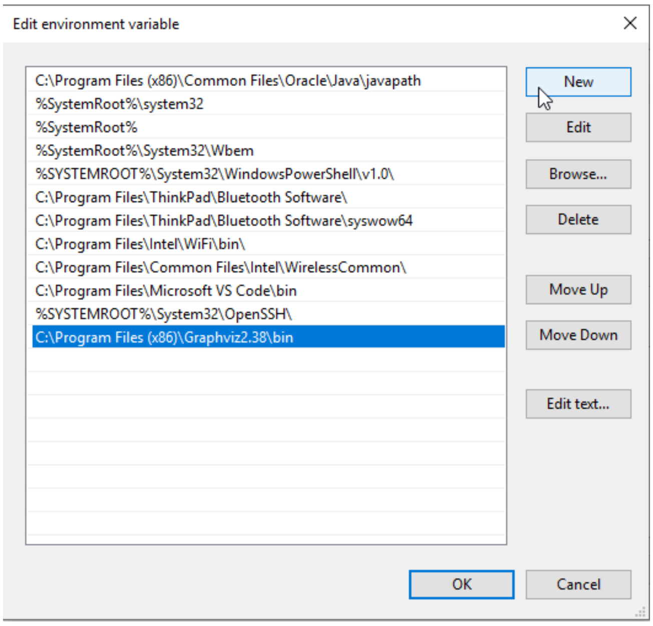
2. **Add conda-forge channel into the system root using Anaconda Prompt.**

```
conda config --append channels conda-forge
```

3. **Install DEAP using Anaconda Prompt.**

```
conda install -c conda-forge deap
```

4. **Install Graphviz**
   - **Install Graphviz for Windows**
   - **Add the Graphviz bin path `C:\Program Files (x86)\Graphviz2.38\bin` in your Windows path (environment variables).**



Edit environment variable

C:\Program Files (x86)\Common Files\Oracle\Java\javapath
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
C:\Program Files\ThinkPad\Bluetooth Software\
C:\Program Files\ThinkPad\Bluetooth Software\syswow64
C:\Program Files\Intel\WiFi\bin\
C:\Program Files\Common Files\Intel\WirelessCommon\
C:\Program Files\Microsoft VS Code\bin
%SYSTEMROOT%\System32\OpenSSH\
C:\Program Files (x86)\Graphviz2.38\bin

- **Close and reopen your terminals so the path changes recognized.**
- **Install the graphviz wheel using Anaconda Prompt.**

```
pip install graphviz-0.8.3-py2.py3-none-any.whl
```

5. **Install Pydot using Anaconda Prompt**

```
conda install -c anaconda pydot
```

6. **Install StringGenerator using Anaconda Prompt**

```
pip install StringGenerator
```

7. **Install Matplotlib using Anaconda Prompt**

```
conda install -c conda-forge matplotlib
```

8. **Install Networkx using Anaconda Prompt**

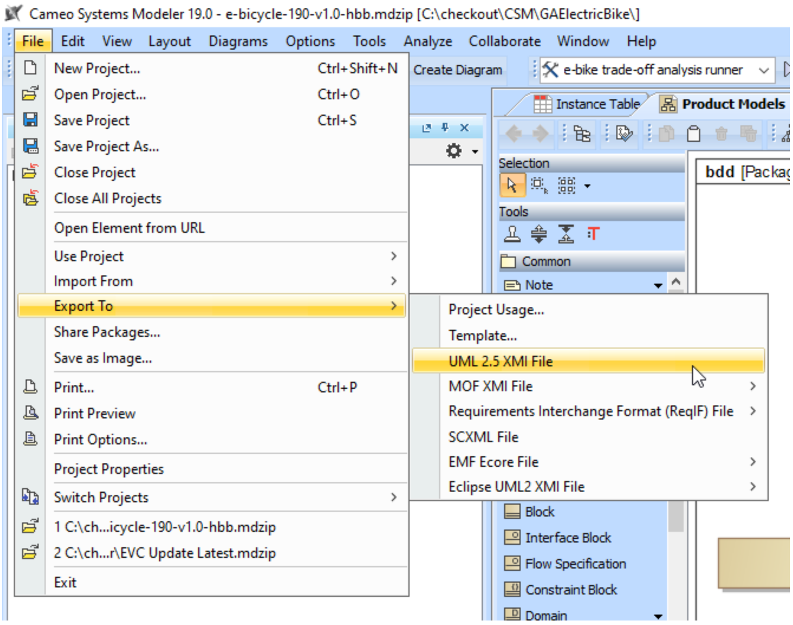
```
conda install -c anaconda networkx
```

9. **Install Js2Py using Anaconda Prompt**

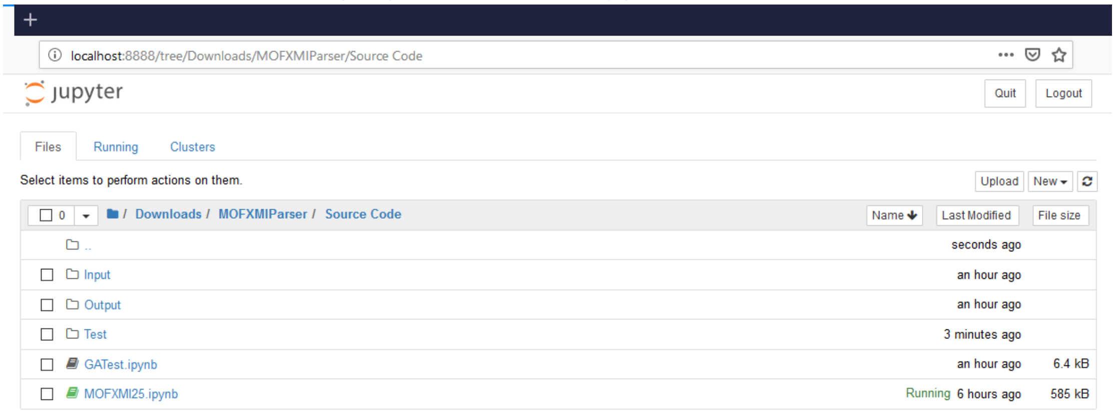
```
pip install Js2Py
```

Step 3: Run the XMI25Element code to get the metamodel for Genetic Algorithm

1. **Export your SysML project from CSM 19.0 to UML 2.5 XMI File. You will get an .xml file (such as “e-bicycle-190-v1.0-hbb.xml”).**



2. **Open the Jupyter Notebook and find your project directory (such as "MOFXMIParser") and find "Source Code" directory. Open "MOFXMI25.ipynb" file.**

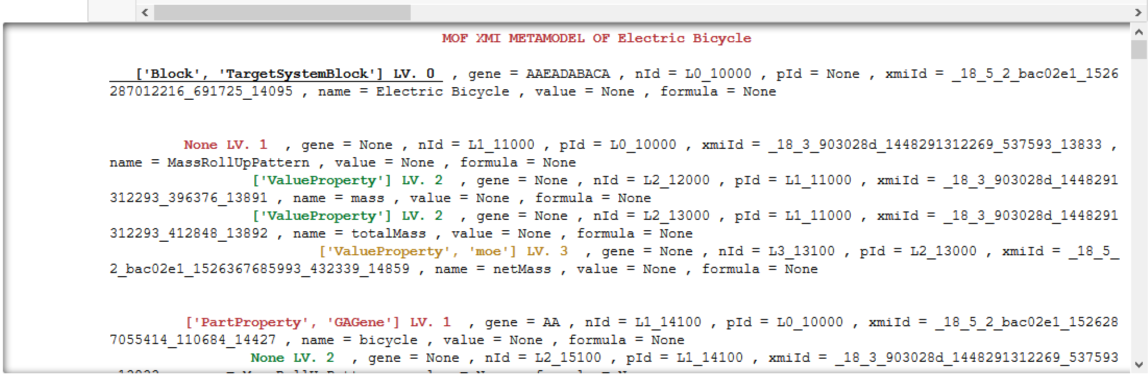


Select items to perform actions on them.

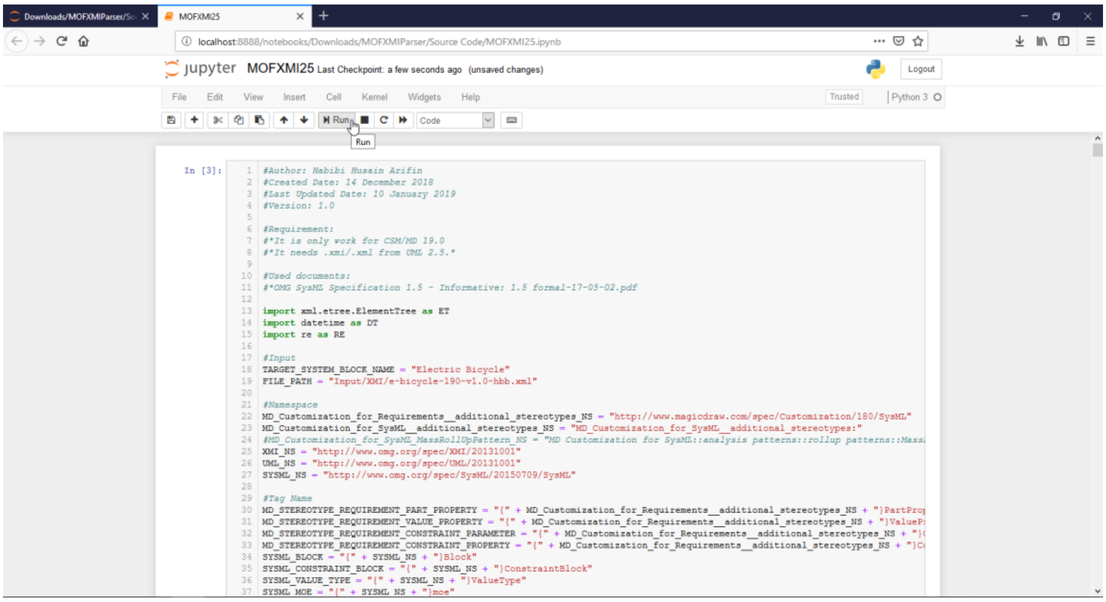
☐ 0 / Downloads / MOFXMIParser / Source Code

Name		Last Modified	File size
..		seconds ago	
☐ Input		an hour ago	
☐ Output		an hour ago	
☐ Test		3 minutes ago	
☐ GATest.ipynb		an hour ago	6.4 kB
☐ MOFXMI25.ipynb	Running	6 hours ago	585 kB

3. **Run the 1st Cell of the program. The result will show you all the necessary element for Genetic Algorithm process.**



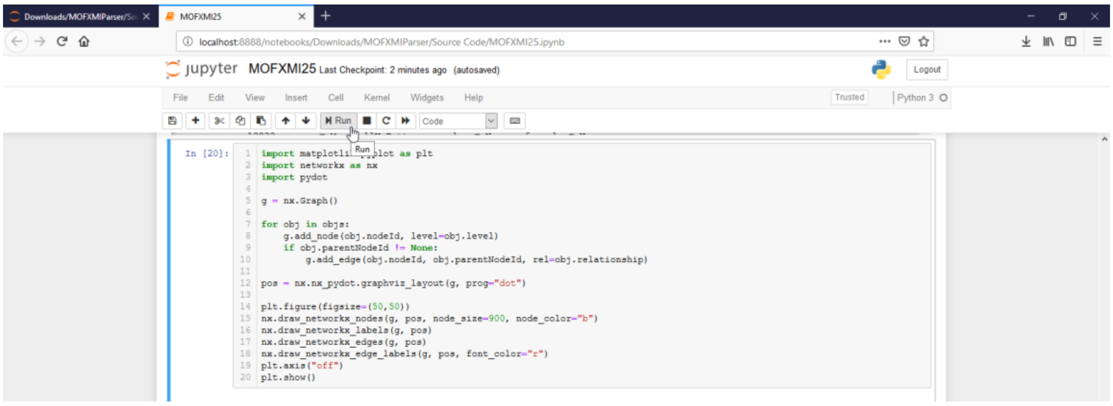
```
                          MOF XMI METAMODEL OF Electric Bicycle

    ['Block', 'TargetSystemBlock'] LV. 0  , gene = AAEADABACA , nId = L0_10000 , pId = None , xmiId = _18_5_2_bac02e1_1526
287012216_691725_14095 , name = Electric Bicycle , value = None , formula = None


          None LV. 1  , gene = None , nId = L1_11000 , pId = L0_10000 , xmiId = _18_3_903028d_1448291312269_537593_13833 ,
name = MassRollUpPattern , value = None , formula = None
                ['ValueProperty'] LV. 2  , gene = None , nId = L2_12000 , pId = L1_11000 , xmiId = _18_3_903028d_1448291
312293_396376_13891 , name = mass , value = None , formula = None
                ['ValueProperty'] LV. 2  , gene = None , nId = L2_13000 , pId = L1_11000 , xmiId = _18_3_903028d_1448291
312293_412848_13892 , name = totalMass , value = None , formula = None
                        ['ValueProperty', 'moe'] LV. 3  , gene = None , nId = L3_13100 , pId = L2_13000 , xmiId = _18_5_
2_bac02e1_1526367685993_432339_14859 , name = netMass , value = None , formula = None


          ['PartProperty', 'GAGene'] LV. 1  , gene = AA , nId = L1_14100 , pId = L0_10000 , xmiId = _18_5_2_bac02e1_152628
7055414_110684_14427 , name = bicycle , value = None , formula = None
                None LV. 2  , gene = None , nId = L2_15100 , pId = L1_14100 , xmiId = _18_3_903028d_1448291312269_537593
```


Downloads/MOFXMIParser/Sc
MOFXMI25
localhost:8888/notebooks/Downloads/MOFXMIParser/Source Code/MOFXMI25.ipynb
jupyter MOFXMI25 Last Checkpoint: a few seconds ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
Run

In [3]:

```
#Author: Habibi Husain Arifin
#Created Date: 14 December 2018
#Last Updated Date: 10 January 2019
#Version: 1.0

#Requirement:
#*It is only work for CSM/MD 19.0
#*It needs .xmi/.xml from UML 2.5.*

#Used documents:
#*OMG SysML Specification 1.5 - Informative: 1.5 formal-17-05-02.pdf

import xml.etree.ElementTree as ET
import datetime as DT
import re as RE

#Input
TARGET_SYSTEM_BLOCK_NAME = "Electric Bicycle"
FILE_PATH = "Input/XMI/e-bicycle-190-v1.0-hbb.xml"

#Namespace
MD_Customization_for_Requirements__additional_stereotypes_NS = "http://www.magicdraw.com/spec/Customization/180/SysML"
MD_Customization_for_SysML__additional_stereotypes_NS = "MD_Customization_for_SysML__additional_stereotypes:"
#MD_Customization_for_SysML_MassRollUpPattern_NS = "MD Customization for SysML::analysis patterns::rollup patterns::Mass
XMI_NS = "http://www.omg.org/spec/XMI/20131001"
UML_NS = "http://www.omg.org/spec/UML/20131001"
SYSML_NS = "http://www.omg.org/spec/SysML/20150709/SysML"

#Tag Name
MD_STEREOTYPE_REQUIREMENT_PART_PROPERTY = "{" + MD_Customization_for_Requirements__additional_stereotypes_NS + "}PartProp
MD_STEREOTYPE_REQUIREMENT_VALUE_PROPERTY = "{" + MD_Customization_for_Requirements__additional_stereotypes_NS + "}ValueP
MD_STEREOTYPE_REQUIREMENT_CONSTRAINT_PARAMETER = "{" + MD_Customization_for_Requirements__additional_stereotypes_NS + "}C
MD_STEREOTYPE_REQUIREMENT_CONSTRAINT_PROPERTY = "{" + MD_Customization_for_Requirements__additional_stereotypes_NS + "}C
SYSML_BLOCK = "{" + SYSML_NS + "}Block"
SYSML_CONSTRAINT_BLOCK = "{" + SYSML_NS + "}ConstraintBlock"
SYSML_VALUE_TYPE = "{" + SYSML_NS + "}ValueType"
SYSML_MOE = "{" + SYSML_NS + "}moe"
```

4. **Run the 2nd Cell of the program to show the Root-Node-Leaf Tree.**


Downloads/MOFXMIParser/Sc
MOFXMI25
localhost:8888/notebooks/Downloads/MOFXMIParser/Source Code/MOFXMI25.ipynb
jupyter MOFXMI25 Last Checkpoint: 2 minutes ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
Run

In [20]:

```
import matplotlib.pyplot as plt
import networkx as nx
import pydot

g = nx.Graph()

for obj in objs:
    g.add_node(obj.nodeId, level=obj.level)
    if obj.parentNodeId != None:
        g.add_edge(obj.nodeId, obj.parentNodeId, rel=obj.relationship)

pos = nx.nx_pydot.graphviz_layout(g, prog="dot")

plt.figure(figsize=(50,50))
nx.draw_networkx_nodes(g, pos, node_size=900, node_color="b")
nx.draw_networkx_labels(g, pos)
nx.draw_networkx_edges(g, pos)
nx.draw_networkx_edge_labels(g, pos, font_color="r")
plt.axis("off")
plt.show()
```