

Integration of I 4.0 standards

Alina Arunova, Maxim Maltsev, Pylyp Matyash, Sattar Rahimbeyli

Tutor: Irlán Grangel

Outline

1. Motivation
2. Definition of problem
3. Goals
4. Requirements
5. Architecture
6. Implementation
7. Results
8. Future work

Industry 4.0

I 4.0 or Industry 4.0 is a combination of production methods with state-of-the-art information and communication technology. In the world of Industry 4.0, people, machines, equipment, logistics systems and products communicate and cooperate with each other directly.



```
<CAEXFile FileName="..." SchemaVersion="..." xmlns:xsi="..."
xsi:noNamespaceSchemaLocation="...">
  <InterfaceClassLib Name="MyInterfaces">
    <Version>1.0</Version>
    <InterfaceClass Name="Energy"
RefBaseClassPath="BaseInterfaceClassLib@AutomationML...">
      <InterfaceClass Name="Heating"
RefBaseClassPath="BaseInterfaceClassLib@AutomationML...">
        </InterfaceClassLib>
      </InterfaceClassLib>
    </InterfaceClassLib>
  </CAEXFile>
```



```
<UANodeSet xmlns:xsi="..." xmlns:xsd="..." xmlns="...">
  <UAObject NodeId="ns=2;i=1" BrowseName="ManufacturingSystem">
    <DisplayName>ManufacturingSystem</DisplayName>
    <Description></Description>
    <References>
      <Reference ReferenceType="HasTypeDefinition">i=61</Reference>
      <Reference ReferenceType="HasComponent">ns=2;i=8</Reference>
    </References>
  </UAObject>
</UANodeSet>
```

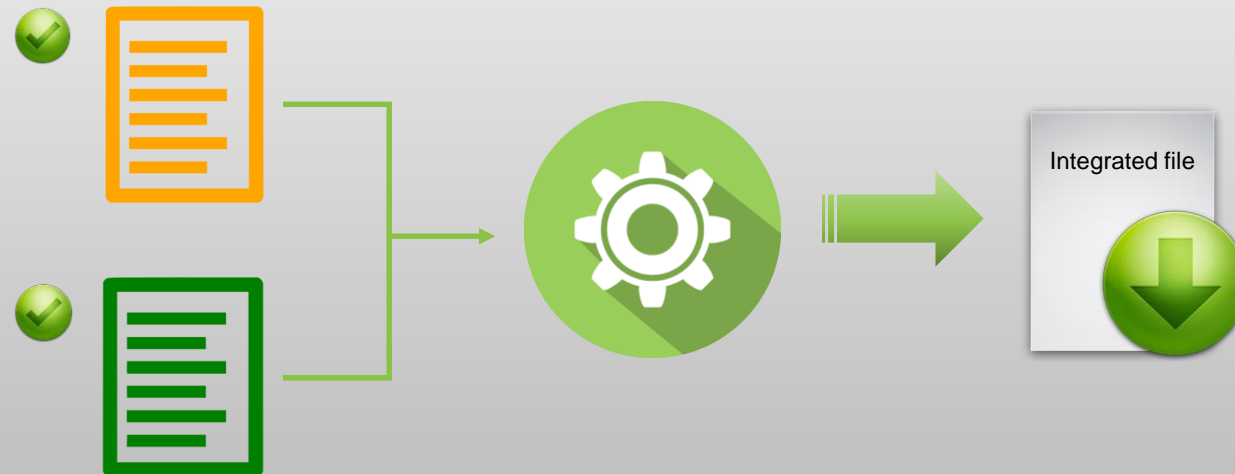
Problems

- Globalization forces large companies to go **international**;
- Multinational companies tend to **consolidate the standards**;
- Different industrial **standards need to be matched**;
- Lots of time wasted onto **manual object notation transcription**.



Goal

- **Validation** of each document against standard schema;
- **Matching** of each entity of the document to a respectful entity of another document;
- Consolidation of the knowledge into the new **unified standard**.



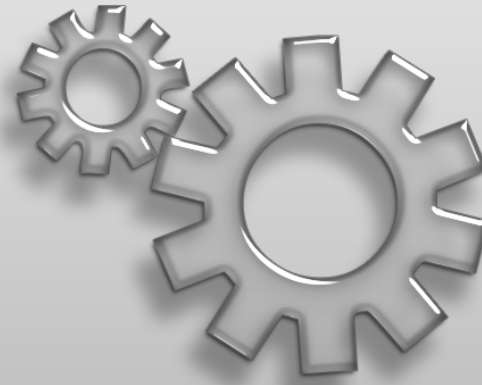
Functional requirements

High Priority:

1. Provide automatic document validation
2. Visualizing input (tree mode)
3. Direct topology mapping
4. Producing integration file

Low Priority:

1. Demo testing of the output against Gold Standard
2. SPARQL query for the output file



Non-functional requirements

High Priority:

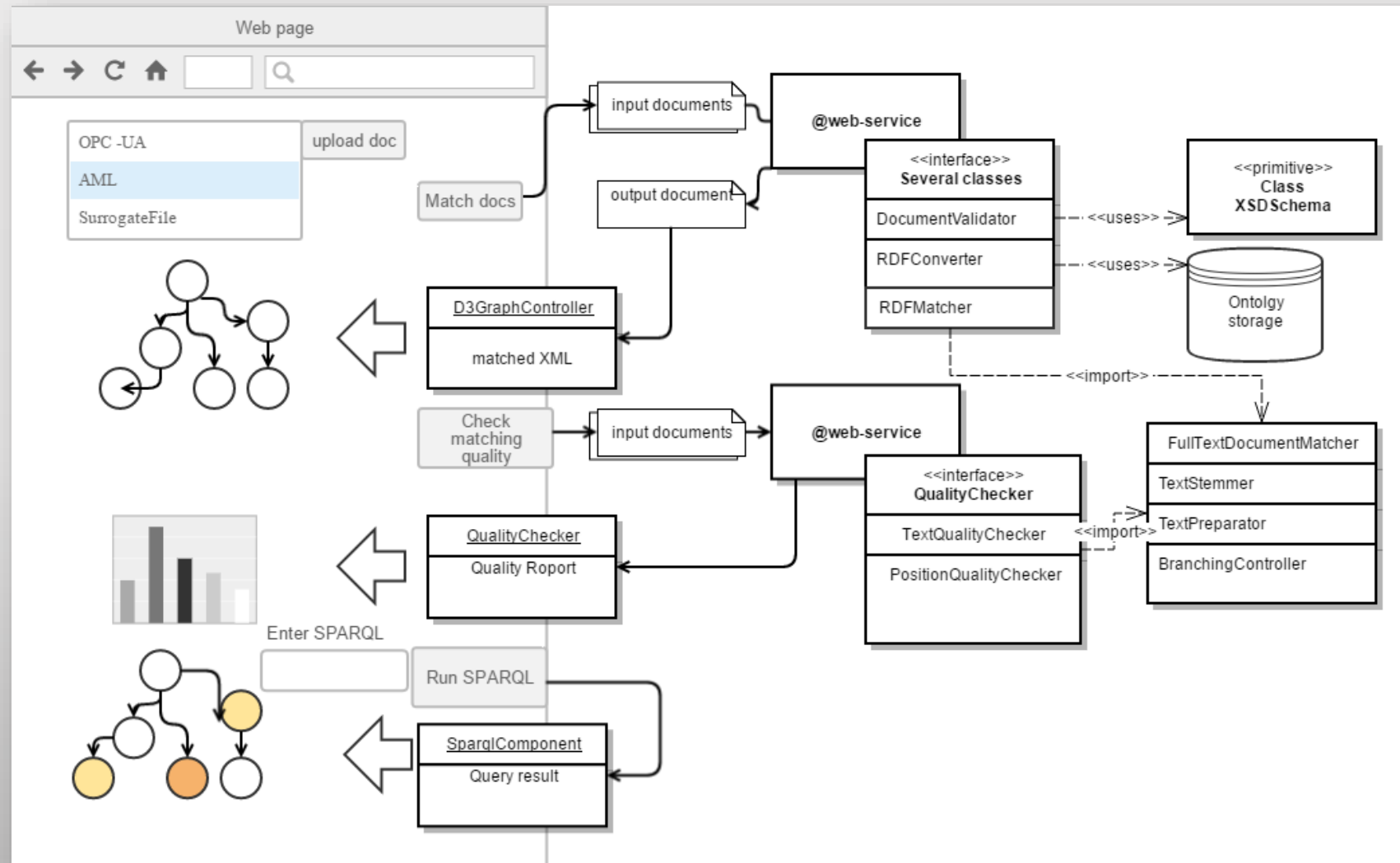
1. Security and fault tolerance

Low Priority:

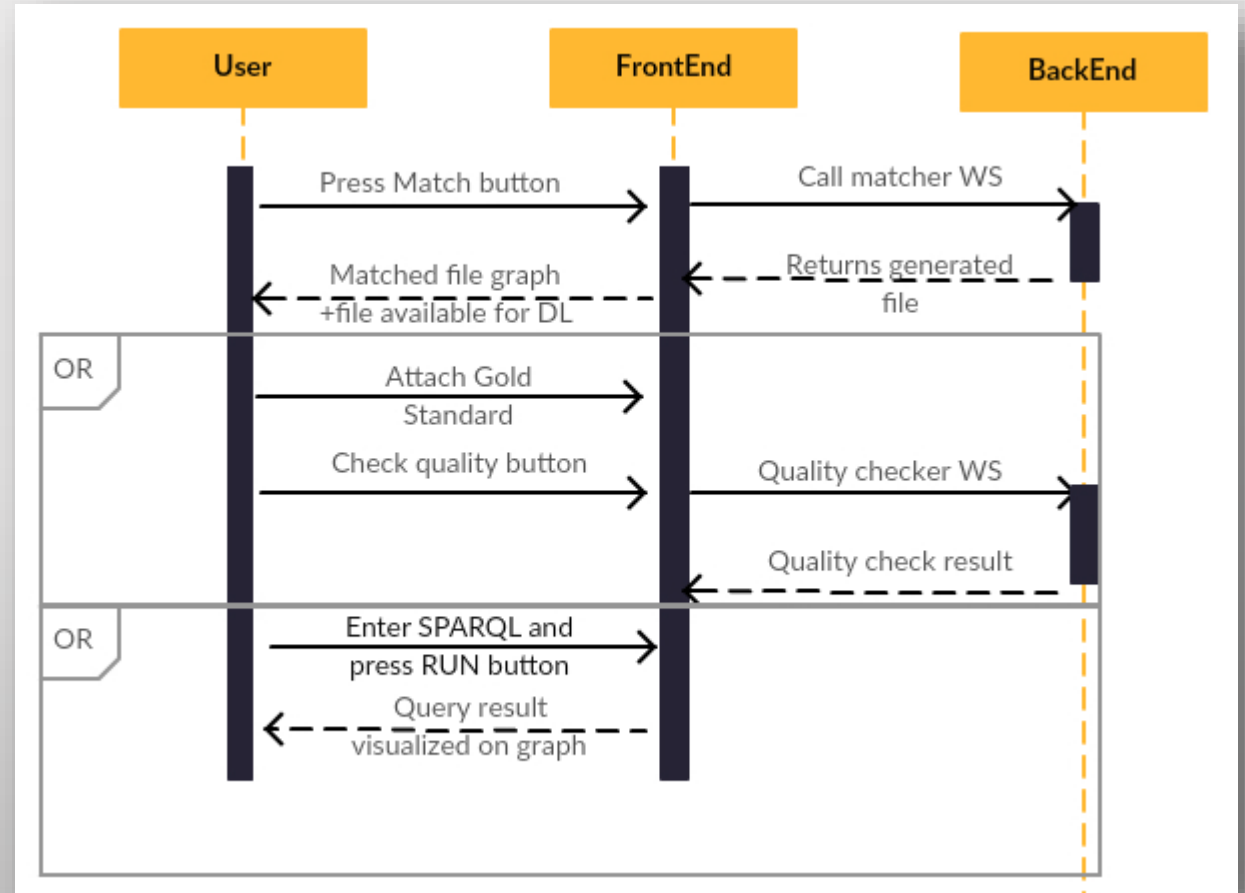
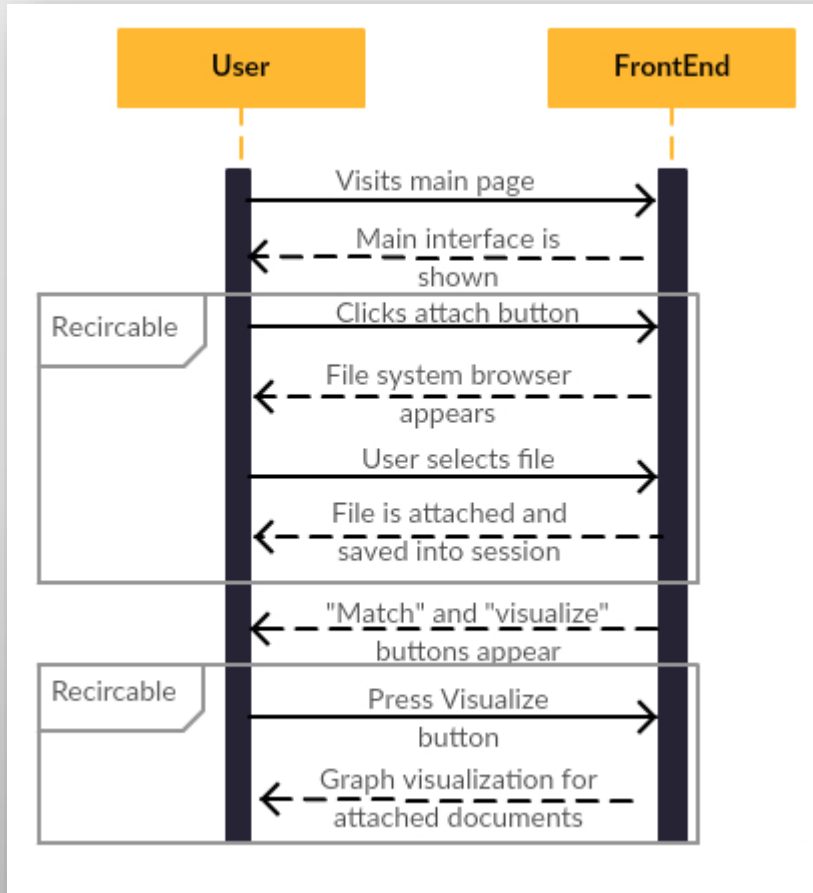
1. Time tolerant processing
2. Scalability
3. Semantic fuzzy matching
4. User-friendly interface



Architecture



Workflow



Responsibilities



Alina



Phil



Maxim



Sattar

Converting <i>AML</i> and <i>OPC UA</i> to <i>RDF</i>	
Matching of 2 files	
Database and WebController development	
Documentation	Deployment
Final presentation	

Front end tool for uploading , downloading and extracting the data	
Documentation	
Tree visualization of the data	Design development
Gold Standard	Testing

Tools



Challenges

Frontend

- Browser session file storage
- Flexible tree-view visualization

Backend

- OPCUA XSLT transformation



Results

Choosing a file:

browse a file from your computer

Select .aml/.opcua file to upload:

Choose File

no file selected

Select .aml/.opcua file to upload:

Choose File

no file selected

Visualize tree-view

Download the file

Visualization

Download the file

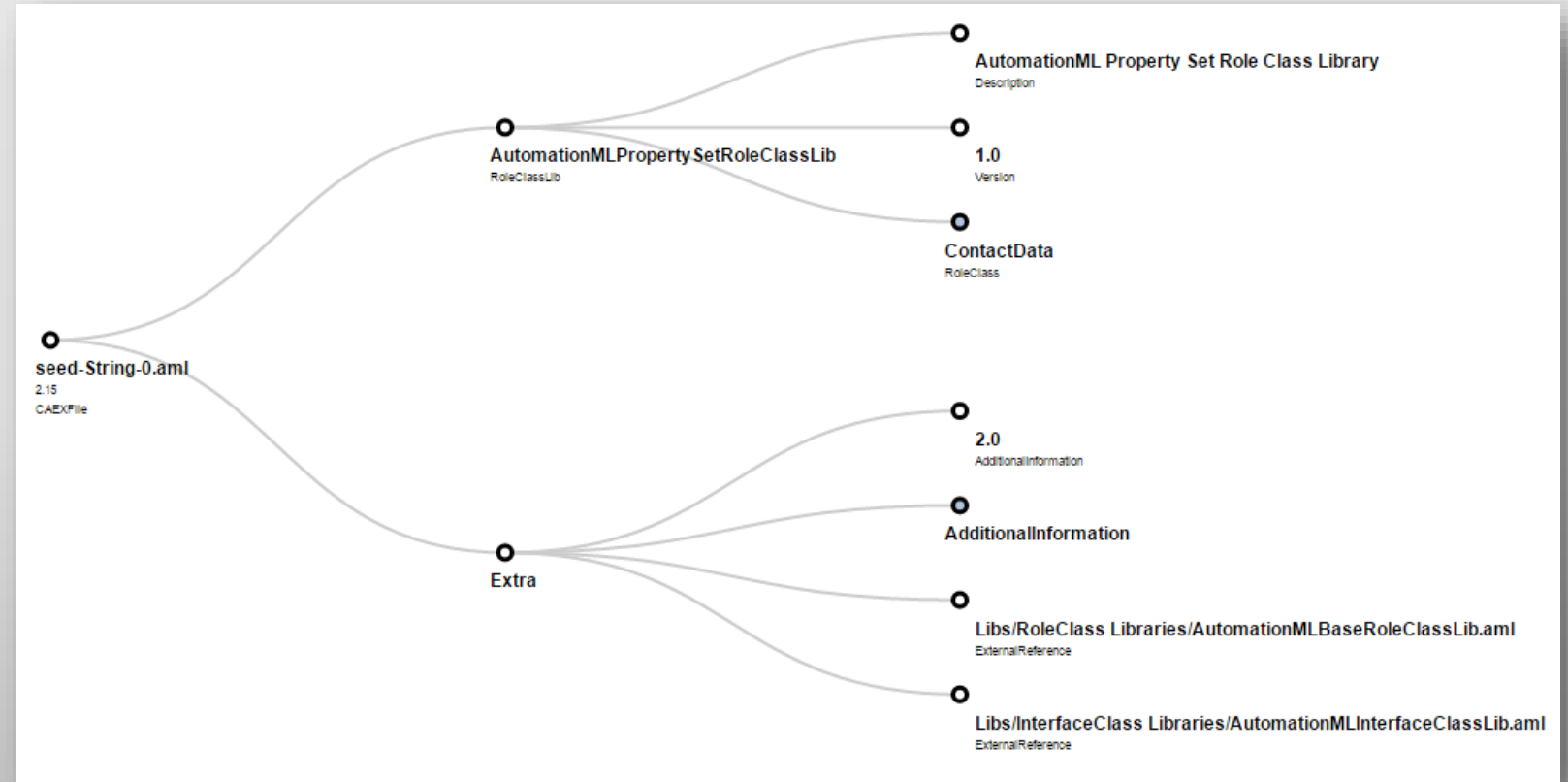
Select .aml/.opcua file to upload:

Choose File

no file selected

Results

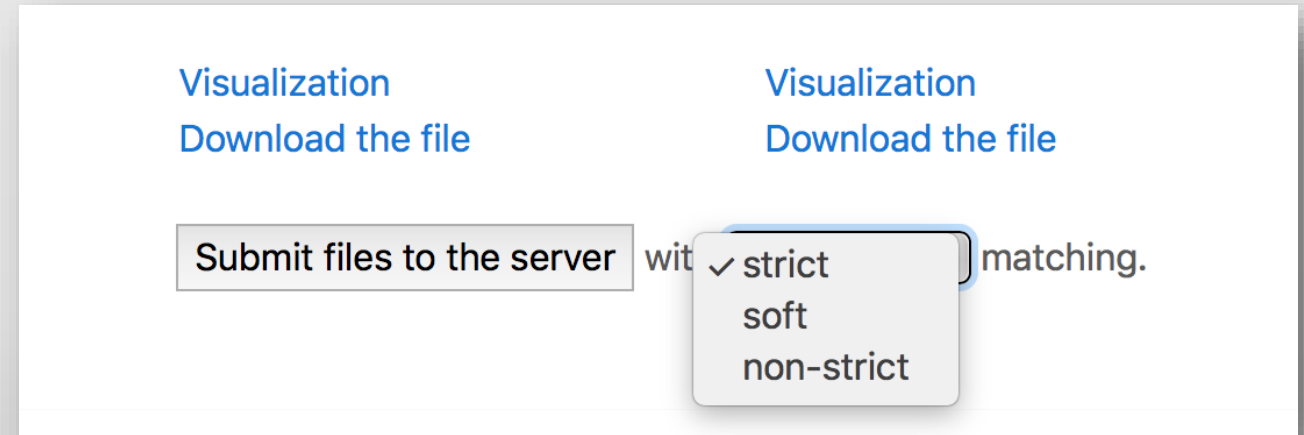
Visualization: Tree view



Results

Matching:

- Strict (match by statement);
- Soft (match by predicate with the common subject);
- Non-strict (match by subject).



The screenshot shows a web interface with two identical sections. Each section has a blue link 'Visualization' and a blue button 'Download the file'. Below these is a grey button labeled 'Submit files to the server'. To the right of this button is a dropdown menu with three options: 'strict' (selected with a checkmark), 'soft', and 'non-strict'. The text 'with' appears to the right of the dropdown, and 'matching.' appears to the right of the 'strict' option.

Visualization
Download the file

Submit files to the server with ☒ strict ☐ soft ☐ non-strict matching.

Results

SPARQL query
Download result file
Visualize result file

Golden Standard:
Matching quality

Match percentage: **100%**

Golden Standard	Output
------------------------	---------------

[Visualization](#)
[Download the file](#)

[Visualization](#)
[Download the file](#)

Integrated file has been created.

Choose the file format for downloading: JSON ▾ Download

You can see the visualization of the integrated file [here](#).

In order to retrieve any specific information, input your SPARQL query below:

select * from \$table\$ where {?s ?p ?o}

Run Query

*use \$table\$ as a table name

[Visualization](#)
[Download the result in JSON](#)

Results: Strict matching example

AutomationML

```
<CAEXFile FileName="..." SchemaVersion="..." xmlns:xsi="..."
xsi:noNamespaceSchemaLocation="...">
  <InterfaceClassLib Name="MyInterfaces">
    <Version>1.0</Version>
    <InterfaceClass Name="Energy"
RefBaseClassPath="BaseInterfaceClassLib@AutomationML..."/>
    <InterfaceClass Name="Heating"
RefBaseClassPath="BaseInterfaceClassLib@AutomationML..."/>
  </InterfaceClassLib>
</CAEXFile>
```

```
<CAEXFile FileName="..." SchemaVersion="..." xmlns:xsi="..."
xsi:noNamespaceSchemaLocation="...">
  <InterfaceClassLib Name="MyInterfaces">
    <Version>1.0</Version>
    <InterfaceClass Name="Energy"
RefBaseClassPath="BaseInterfaceClassLib@AutomationML..."/>
    <InterfaceClass Name="Water"
RefBaseClassPath="BaseInterfaceClassLib@AutomationML..."/>
  </InterfaceClassLib>
</CAEXFile>
```

Results: Strict matching example

AutomationML

```
<https://w3id.org/i40/aml/InterfaceClass2>
  a    <https://w3id.org/i40/instance/InterfaceClass> ;
  <https://w3id.org/i40/instance/refBaseClassPath>

"BaseInterfaceClassLib@AutomationMLInterfaceClassLib/AutomationMLBaseInterface" .

<https://w3id.org/i40/aml/CAEXFile1>
  a    <https://w3id.org/i40/instance/CAEXFile> ;
  <https://w3id.org/i40/instance/hasFileName>
    "Topology.aml" ;
  <https://w3id.org/i40/instance/hasInterfaceClassLib>
    <https://w3id.org/i40/aml/InterfaceClassLib1> ;
  <https://w3id.org/i40/instance/hasSchemaVersion>
    "2.15" .
```

```
<https://w3id.org/i40/aml/InterfaceClass1>
  a    <https://w3id.org/i40/instance/InterfaceClass> ;
  <https://w3id.org/i40/instance/hasAttributeName>
    "Energy" ;
  <https://w3id.org/i40/instance/refBaseClassPath>

"BaseInterfaceClassLib@AutomationMLInterfaceClassLib/AutomationMLBaseInterface" .

<https://w3id.org/i40/aml/InterfaceClassLib1>
  a    <https://w3id.org/i40/instance/InterfaceClassLib> ;
  <https://w3id.org/i40/instance/hasAttributeName>
    "MyInterfaces" ;
  <https://w3id.org/i40/instance/hasInterfaceClass>
    <https://w3id.org/i40/aml/InterfaceClass1> ,
  <https://w3id.org/i40/aml/InterfaceClass2> .
```

Future work

- OPC UA and AML integration
- Fuzzy matching
- Developing ontology based mapping



Thank you for
attention!