Otto-Von-Guericke-Universität Magdeburg Fakultät Für Maschinenbau

Project: Automation of Plastic Plate Manufacturing

Submitted for the fulfillment of

Establishing Digital Engineering Chains

to

Prof. Dr.-Ing. habil. Arndt Lüder

Faculty of Mechanical Engineering

Institute for Ergonomics, Factory Automation and Factory Operations By

Team F

Paras Savaliya (234630)

Tran Phoi Tren (230540)

Kiran Sanekal (235189)

Nikhil Francis (235126)

Abstract

Industrial revolution has brought major changes in the field of manufacturing and have massively influenced the economic growth of the countries. The industrial revolution, over the century has witnessed three separate phases of revolution & currently moving towards the fourth phase. The fourth industrial revolution (Industry 4.0) is the ongoing automation of traditional manufacturing and industrial practices, using modern smart technology. Large-scale machine-to-machine communication and the internet of things (IoT) are integrated for increased automation, improved communication, self-monitoring and production of smart machines that can analyse and diagnose issues without the need for human intervention. The use of Industry 4.0 requires the components to be interoperable so that they seamlessly communicate across companies and industries. Asset Administration Shell (AAS) provides the base for development and use of such Industry 4.0 components. In this project, we intend to fully automate the existing plastic plates manufacturing process made by recycled plastic containers and model an Asset Administration Shell (AAS) structure for the involved production system components. In addition, we have also integrated the logistics aspects of the production unit provided to us by the respective team.

Declaration

We hereby, Paras Savaliya, Kiran Sanekal, Nikhil Paul Francis and Tran Phoi (Solomon) Tien, declare that we have done this project independently and without unauthorized help, that we have not used other sources and aids than those indicated, and that we have indicated as such all content taken literally or alternated from the sources used.

The work has not yet been submitted in the same or similar form to any other examination authority and has not been published

Table of Contents:

List of Figures

List of Tables

1.0. Introduction

1.1. Problem Statement

Why do we need to recycle plastics?

Less than 10% of all plastic generated globally is recycled. This plastic ends up in landfills, oceans, in animals, and even in our food and water. According to one survey done by EPA the total amount of plastics generated in 2018 were 36,680 thousand tons out of which only 3090 thousand tons of plastics were recycled, and 36,970 thousand tons of plastics went into landfills. While the overall recycling rate is quite small – 8.7% in 2018, the recycling rate of PET plastics was 29.1% and that of HDPE was 29.3% in 2018 [1]. Polypropylene (PP) is one of the largest consumed plastics and is often the least recycled plastic [2]. Thus, the plastics are harming the ecosystem and if not taken seriously, may lead to huge amounts of solid waste which might take centuries to decompose in landfills and oceans. Therefore, recycling of plastic is crucial to conserve the environment.

What is being done currently?

Not all types of plastics are recycled. Plastics like PET, HDPE and PP are among the most recyclable plastics. The recycling process involves collection of plastics and sorting them according to their grades. After sorting, the plastic is cleaned thoroughly to remove any impurities or foreign debris. Then it goes through a granulation process where the shredder shreds the plastics into small granules. These granules are then melted and used to make new products. [3]. Recycling can only be economically viable for recycles if they produce high volumes of product with minimal environmental impact.

Why do we need automation?

To improve the current recycling process, the recycling businesses need to address the current challenges of the industry and also develop new innovative approaches. With the help of technology and the new approach bought by Industry 4.0, it is possible to generate a circular economy for plastics. [4]. The current challenges can be solved using automation technology.

With automation it is possible to attain higher productivity and lower costs. Automation can help in reducing the labour cost, reduction in wastage of raw material and overall reduction in production costs. Robotic automation can help in improving safety standards of the plant. Automation also offers more flexibility in the production system with use of technologies such as robotic arms which can execute multiple tasks at a time. [5] In essence automation can make handling of larger volumes easier while lowering the environmental impact of recycling processes without increasing the plant footprints. [4]

1.2. Asset Administration Shell

The Asset Administration Shell is the interface connecting Industry 4.0 standards to the physical assets and stores all the data and information regarding the asset. The Administration shell is addressable in the network and identifies the asset unambiguously and provides a standardised and secure communication interface. It is a corner stone of interoperability between the applications managing the manufacturing systems. The Asset Administration Shell

- Establishes cross-company interoperability.
- Is available for both non-intelligent (passive) and intelligent assets.
- Covers the complete life cycle of the assets and services.
- Enables integrated value chains.
- Is the digital basis for autonomous systems and AI.

2.0. Step 1: Development of PPR model for the production system

Any production system covers three main parts: products, processes and resources. They are interdependent with each other. Products are produced by processes, which are executed by resources.

2.1. Products

The products are the intended output of a production system. Customers look at whether the product has its desired properties and capabilities. The products are represented by oval

The goal of this project is to develop a production system that is capable of converting plastic waste into usable sheets. Currently, there are many efforts done locally by small groups to convert plastic waste into useful sheets which can further be used to create various end plastic products. The plastics that are currently used for this project are High-Density Polyethylene (HDPE) and Polypropylene (PP).

The intermediate products during production processes are mixed granules, hot pressed sheets, cold pressed sheets, quality-controlled sheet and finished sheets.

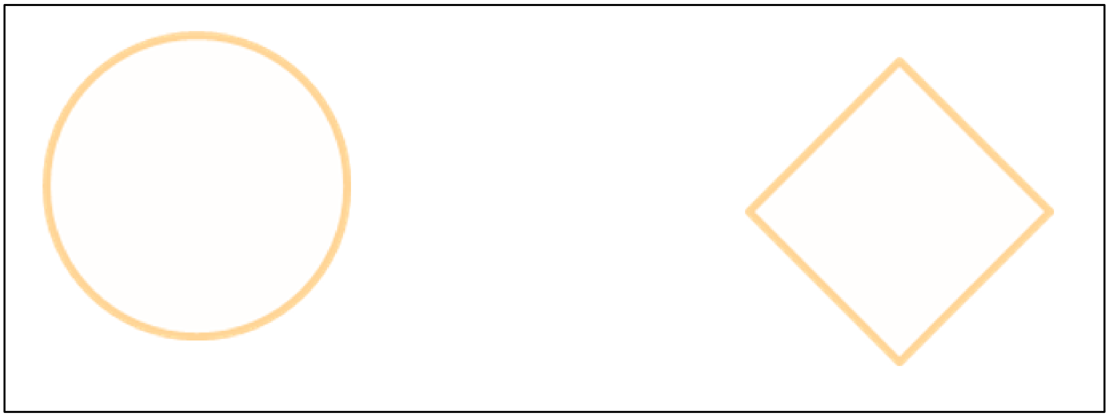

Figure 1: Representation of products in PPR diagram

2.2. Process

Transforms input which can be materials, energy, information to outputs of desired type. It requires resources for execution.

The various processes within our production system are

1. The newly arrived plastic granules are unloaded by KUKA 1(unloading).
2. It is then transported to the mixer by conveyor (mixing).
3. Silicon oil is sprayed in the mould with the help of a nozzle (spraying).
4. KUKA 2 places the mould for filling.
5. KUKA 1 fills the mould with mixed granules.
6. Filled mould is conveyed to vibrating table with the help of accumulator (vibrating)
7. Evenly spread mould is covered with KUKA 2 (covering).
8. The mould is then transported to hot press (Hot pressing).
9. The excess is removed from hot pressed sheet by KUKA 3. (Excess removing).
10. The hot-pressed sheet is then conveyed to cold press. (Cold pressing)
11. The form is removed by KUKA 4.
12. The sheet is transported to optical scanning system.
13. It is the transported to CDP station where drilling centring and polishing is done by KUKA 5.
14. Final inspection is done by optical scanning system.
15. Products that do not meet desired quality are rejected.
16. The finished sheet is then sent to Labeller. (Labelling)

Figure 2: Representation of processes

2.3. Resources

Represents the technology which is used for converting raw materials to finished good.

The following resources are used in our production system:

- Mixer
- Vibrating table
- Spraying Unit
- Hot press machine
- Cold press machine
- Optical scanning System
- Measuring instruments
- Cutting/Drilling/Polishing tools
- Labelling machine
- Accumulators
- Turntable's
- Conveyors

Figure 3: Representation of resources

The machinery listed below are required for setting up a basic manufacturing plant:

2.3.1. Robotic Arm:

Requirement: Lifting the weight of the raw material and the finished product after the processing.

Lifting Weight: If we consider the maximum size of the plate to be 1000mm*1000mm*30mm then the weight of the plate will be approx. 29 kg.

Range: The range of the arm should be at least 1200mm so that it can handle the plates.

The following robot arm can be used: KR IONTEC

Figure 4: KUKA Robot KR IONTEC []

Table 1: Robotic Arm Requirement and Cost

	Requirements	KUKA Robot KR 50 R2100
Degree of freedom	>5	6
Lifting weight [kg]	>29	50

Reach [mm]	>1000	2101
Programmability	Yes	Yes
Cost [€]		25000 Euro

2.3.2. Heat press

· Requirement:

To melt the plastic inside the mould and make it suitable to process it according to the desired shape.

· The plastic will be heated up to a temperature of 160°C by maintaining an equal pressure and temperature on the mould plate. Considering the dimensions of the plate, the hot-pressing operation will be performed for an estimated time of 10 min. The mode of operation can be manual or automatic.

Table 2: Heat Press Requirement and Cost

Attributes	Requirements	Heat Press Machine
Maximum Temperature [°C]	150-200	500
Work surface dimensions [mm]	1620 X 1620 X 1780	2000*2000*1000
Maximum force/load [tons]	50	70
Cost [€]		2600 Euro

2.3.3. Cold press

· Requirement:

To solidify the plastic inside the mould to get the required shape. Considering the dimensions of the plate, the cold -pressing operation will be performed for an estimated time of 10 min. The press will be operated automatically.

Table 3: Cold Press Requirement and Cost

	Requirements	Cold Press Machine
Operating type		Pneumatic
Work surface dimensions	1620 X 1620 X 1780	2000 X 2000X 2000
Maximum force/load(in tons)	50	70
Weight	450	
Cost[€]		2000

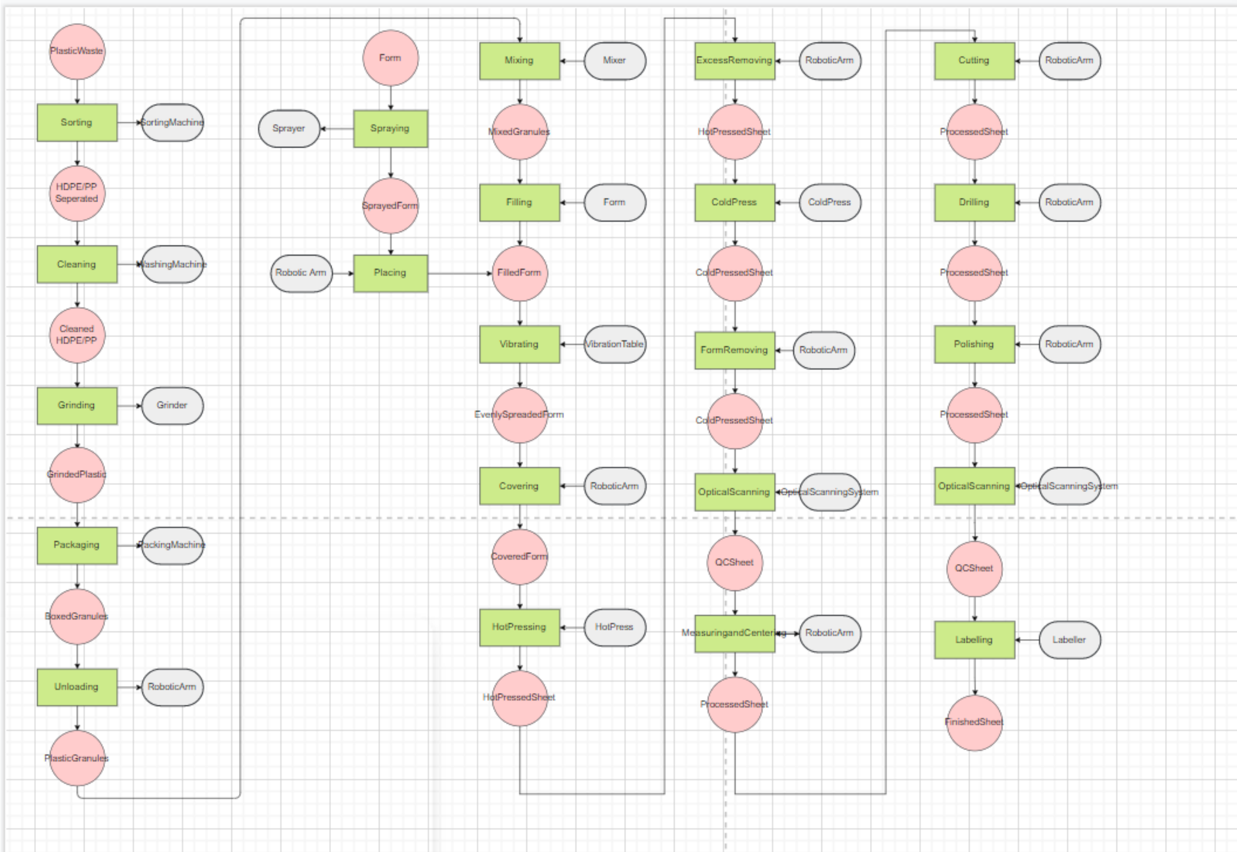

Figure 5: PPR

2.4. Bill of Operations

1. Unloading
2. Mixing
3. Form Placing
4. Spraying
5. Hot pressing
6. Removing Excess
7. Cold pressing
8. Removing form
9. Optical scanning 1
10. Measuring and Centring
11. Cutting, Drilling & Polishing
12. Optical scanning 2
13. Labelling

2.5. Bill of Materials

- Plastic Bin (for rejected panels)
- Clamps for post processing processes
- Screws for mountings
- Form
- HDPE/PP
- Plastic granules
- Plastic Sheet (Panel)

3.0. Step 2: Basic Engineering

In this step, we are detailing the resources of the production system. Hierarchy of resources are necessary to understand whether there is possibility of automation and the resources are capable of producing products of required properties.

Entity Relationship diagrams are used for this purpose.

The main elements of ERD diagram are

3.1 Entity: Describes an identifiable instance of an object i.e., instance model or class of similar objects i.e., type model. It can be detailed by attributes.

Graphically represented by rectangles.

e.g.: objects in a company like products, resources or engineering documents.

3.2 Relation: represents dependencies existing between two or more entities.

Graphically represented by diamonds.

e.g.: Assigning sensors to check the accuracy of the process.

3.3 Attributes: represents the properties of relation or entities of similar types have the same properties.

Graphically represented by circles.

e.g.: properties like maximum speed, supply voltage etc.

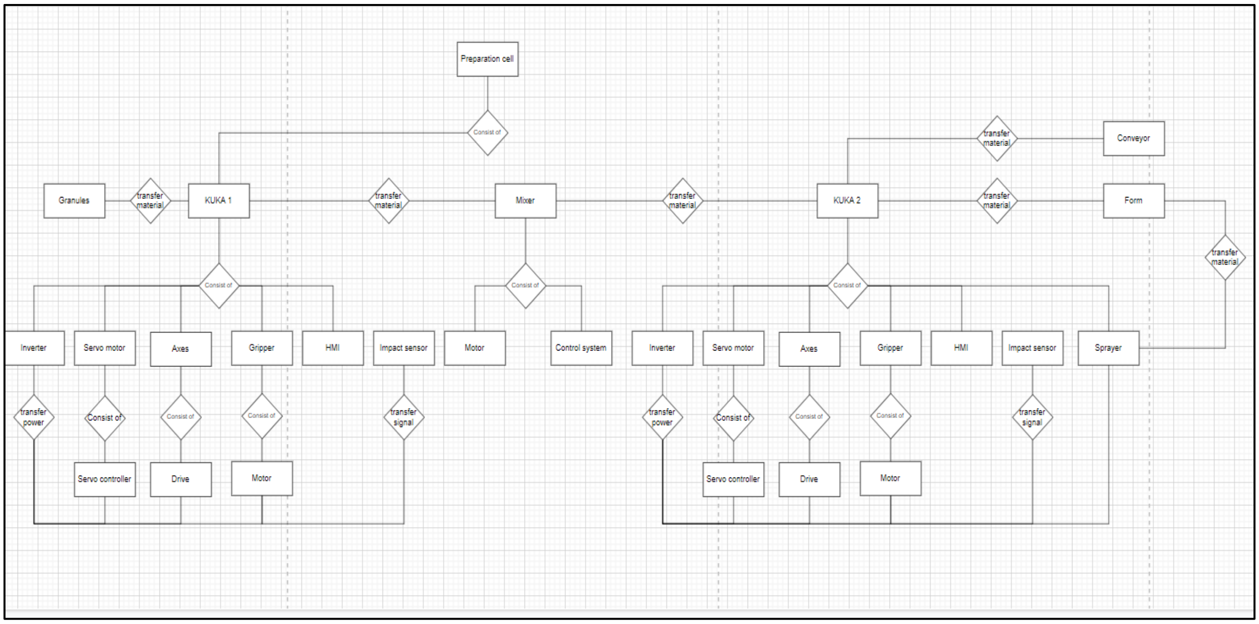

Figure 6: ERD diagram for preparation cell

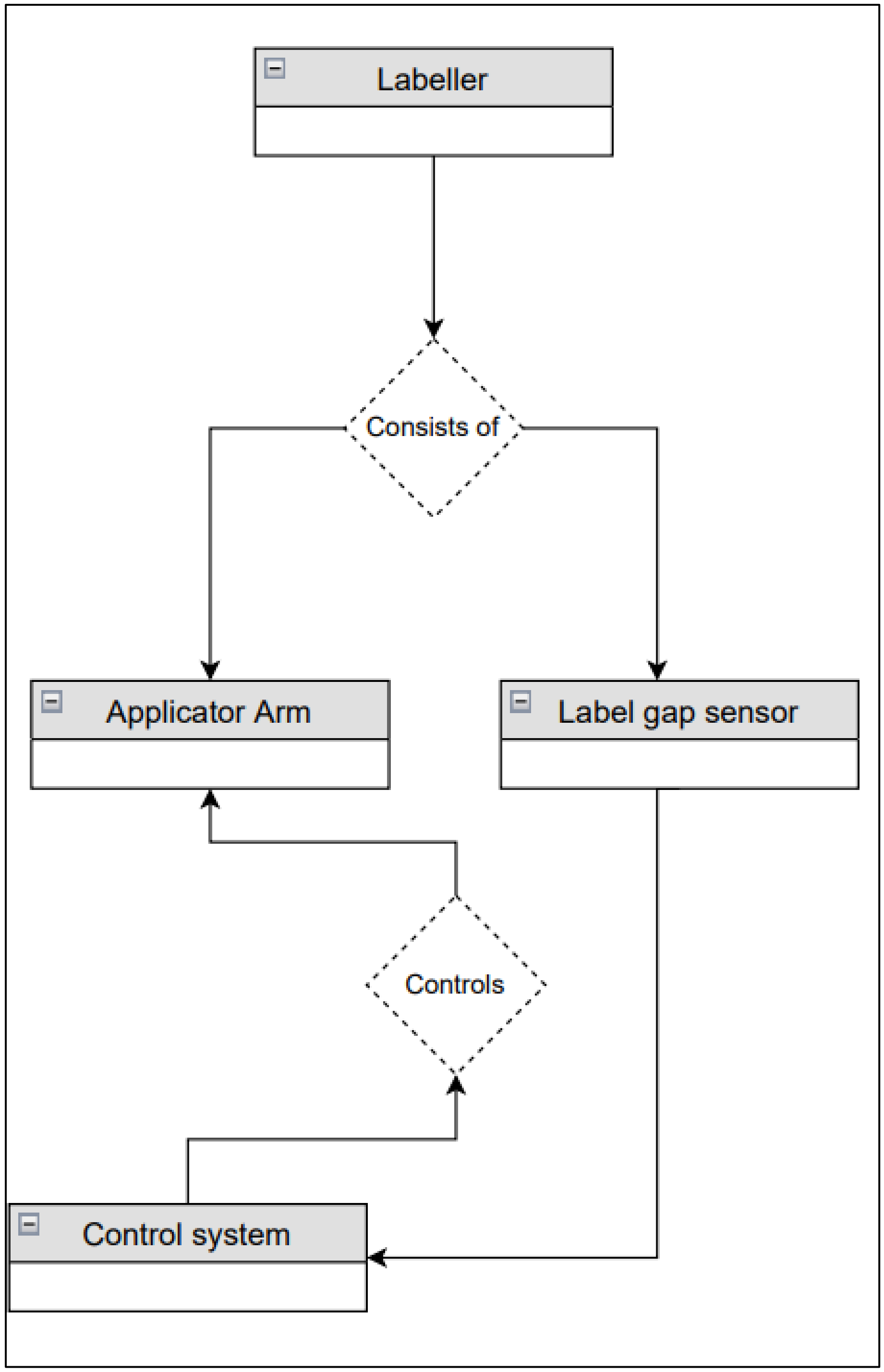

Figure 7: : ERD diagram for Labelling cell

4.0. Step 3: Common Concept Identification

In this step various viewpoints of the data logistics system were defined. Any product, resource, or process which are called concepts consists of functional, mechanical, PPR, electrical and automation related views. The interaction between these concepts was also defined. The attributes relevant to these engineering disciplines were defined as domain-specific languages. Each domain-specific language consists of a set of concepts with related properties and relationships. By this step we will understand the requirements of all the components used.

The following steps were done:

4.1. Identify activities and derive views

- Based on the PPR model we define product design
- Based on the functional hierarchy we define function view
- Electrical view based on requirements of electrical components
- Mechanical view based on mechanical properties of components.
- Automation view

4.2. Identify concepts

- Concepts can be products, processes and resources.
- Relationships between these concepts are also defined

Table 4: Concepts with related views and relationships

	Product Design	Identification	Function Design
Unloading	ThroughputTime CycleTime Capacity	ProcessID SerialNumber	
Mixing	ThroughputTime CycleTime BatchSize	ProcessID SerialNumber	
Placing	ThroughputTime SequencingMode PositionAccuracy	ProcessID SerialNumber	
Spraying	ThroughputTime Material	ProcessID SerialNumber	
Filling	ThroughputTime	ProcessID SerialNumber	
Vibrating	ThroughputTime VibrationAmplitude VibrationFrequency	ProcessID SerialNumber	
Covering	ThroughputTime PositionAccuracy	ProcessID SerialNumber	
HotPressing	ThroughputTime PressingForce Temperature	ProcessID SerialNumber	

	Product Design	Identification	Function Design	Mechanic	Electrical	Automation
RoboticArm	ID Type Resource ManufacturingCellName ManufacturingCellNumber MeanTimeBetweenFailure MeanTimeToRepair PurchaseCost RepairCost MaintenanceCost	Asset ID Manufacturer name Serial number Hardware version	Energy cost Reach Max speed Working speed Max acceleration Working acceleration	Material Weight Dimensions		WorkingVelocityAxes WorkingAccelerationAxes Smoothing
ServoMotor		Asset ID Manufacturer name Serial number Hardware version	Max Speed Energy Cost	Size Weight Speed Torque	Max Rotation Speed Nominal Rotation Speed Output Power Rated Voltage Rated torque	Type of encoder
ServoController		Asset ID Manufacturer name Serial number Hardware version Software version	Energy Cost	Size Maximum Weight Operating temperature range Operating Humidity	SupplyVoltage SupplyCurrent NumberOfControlledAxis	MaxMotionSpeed MaxMotionAcceleration I/O configuration ON/OFF delay Control Method Communications function
ImpactSensor			ImpactCapacity	HousingMaterial Weight Mechanical conditions Operating temperature range	Rated voltage Supply frequency Power Consumption	

	Identification	Function Design	Mechanic	Electrical	Automation	Logistics
MixingCell		Contained Resource Robotic Arm Mixer, Form, VibratingTale, Sprayer				
PressingCell		Contained resource Hot press, cold Press RoboticArm				
ScanningCell		Contained resource: ScanningSystem				
CDPCell		Contained resource: RoboticArm				
LabellingCell		Contained resource: Labeller				
RoboticArm		Contained in Mixing Cell, Pressing Cell, CDP cell				
ServoMotor		Contained in Robotic arm	Mounted in robotic arm			
ServoController		Contained in Robotic arm	Mounted in robotic arm			
ImpactSensor		Contained in Robotic arm	Mounted in robotic arm			
Inverter		contained in Robotic arm	Mounted in robotic arm	Input energy wire Output energy wire to robotic arm		
HMI		Contained in Robotic arm Contained in desktop	Mounted in robotic arm	Input information from robotic arm Output information wire to computer		
EndEffector		Contained in Robotic Arm	Mounted in robotic arm	Input energy from Inverter		
Mixer						
Motor		Contained in mixer	Mounted in mixer	Input energy wire		
VibratingTable						
Sprayer						
Nozzle		Contained in sprayer	Mounted in sprayer			
PhotoElectricSensor			Mounted in Conveyor	Input electric wire Output information wire to desktop		
HotPress						
ColdPress						
Pneumatic cylinder		Contained in hotpress	Mounted in hotpress			

5.0. Step 4: Convert Common Concepts into data model

In this step, the common concepts identified in our previous engineering phase will be used to create data models in AutomationML dialects. Global data models will be created for the development of the data logistics and local data models fullfill the requirements of engineering tools. This global data models refer to one common concept representing the data of all the individual disciplines. AML generator is used for the generation of these data models and a file structure in YML format will be given as the input for the generator.

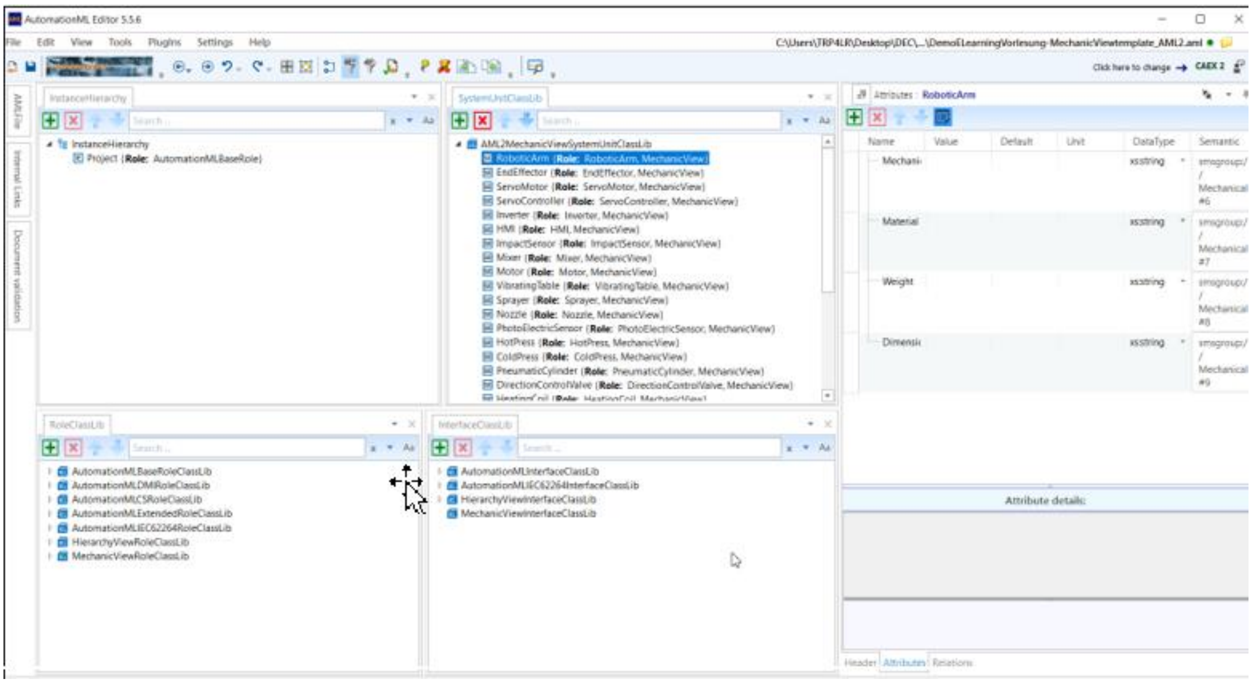

Figure 8: Common Concept AML

The data models generated in the AutomationML dialects will be in the form of Role class (RC), Interface class (IC) and System unit class (SUC). These help in structuring the engineering data in the object-oriented methods that can be reused for further development. Role classes define the object semantics in an abstract manner without exposing the underlying technical implementation while the interface classes describe an abstract relation between elements. System unit classes are reusable components that can be used during system modelling depending on different viewpoints of engineering disciplines. In addition to these classes, we will have Internal Elements (IE) which represent the actual physical components. In this project step, the AML generator uses a Java based environment to create the data models in four steps. In first step, role classes will be created for individual viewpoints and interface classes will be created representing the hierarchy of individual viewpoints. In second step, role classes will be created for concepts within individual role class libraries belonging to a specific viewpoint and in next step, system unit classes will be created within individual system unit class libraries belonging to specific viewpoints. The final step is the creation of overall system unit class for all concepts by combining the information from all the views. The word "View" here refers to the individual engineering disciplines that was listed during the common concept identification phase. This overall SUC represent all concepts which becomes Global domain specific language for this project.

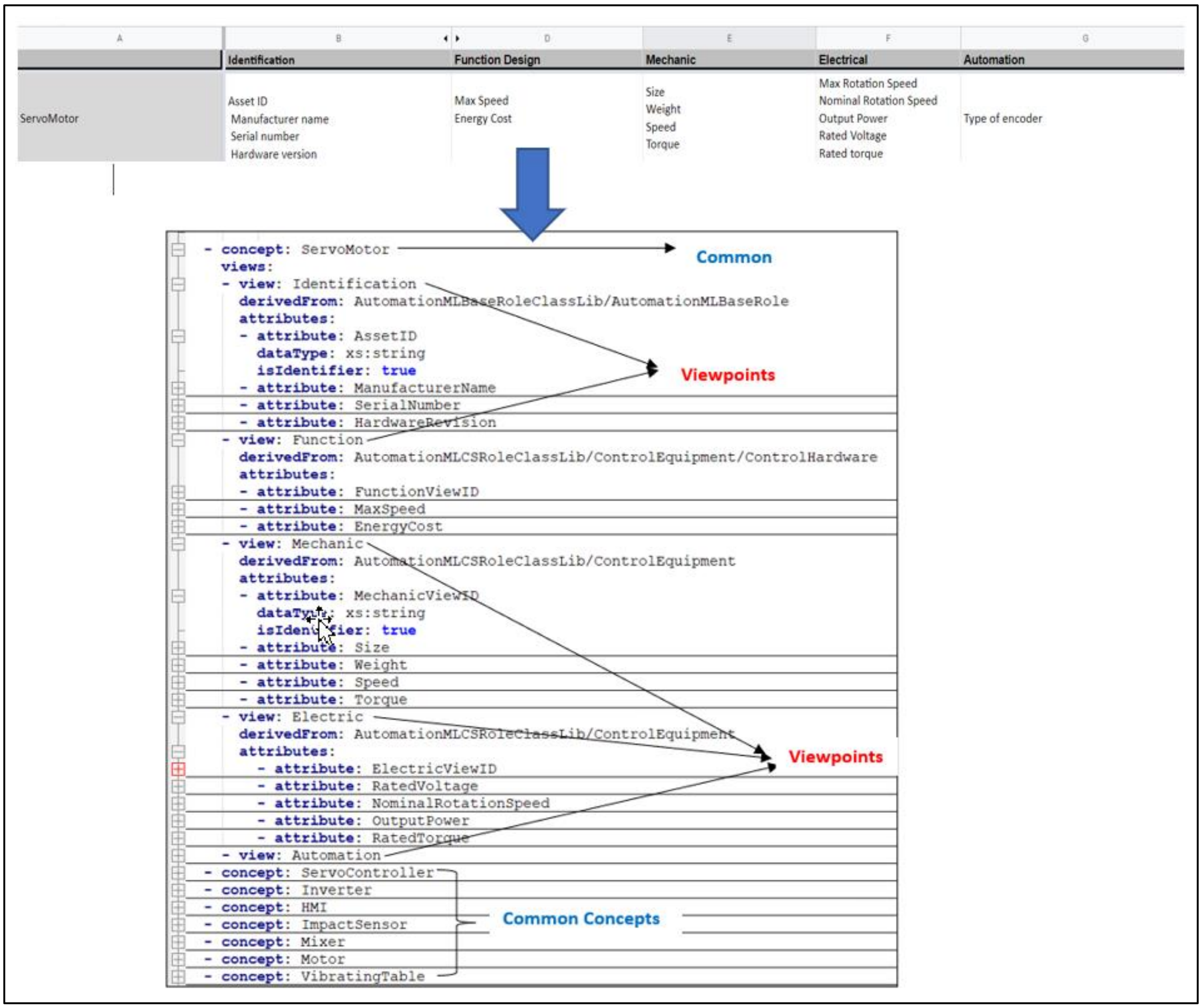

Figure 9: Example of Servomotor from common concept identification

File semantics:

The objects and its related information are derived from the previous phase of concept identification. The rows containing the common concepts sheet is represented under the structure **concept** in

this phase. The columns from previous phase becomes the **view** and the related properties becomes the **attributes** in the YML structure. Thus, all the concepts with its attributes from the individual engineering domains are captured within this single YML file and the AML generator transforms this data into AML data models

6.0. Step 5: Tool configuration for the local data models

In this step we design configurations for the local data models which were created in the previous phases using Automation tool. To achieve this translation from Automation local domain specific languages to tool configurations, we make use of an engineering tool called MDRE. This tool was developed as a prototype concepts from the students at Jena university, Germany. In this project, we use a text based JSON file format to create our model configurations for individual engineering disciplines. Thus, each individual view will have one configuration file which are later used within an environment to setup domain specific models in the following stages

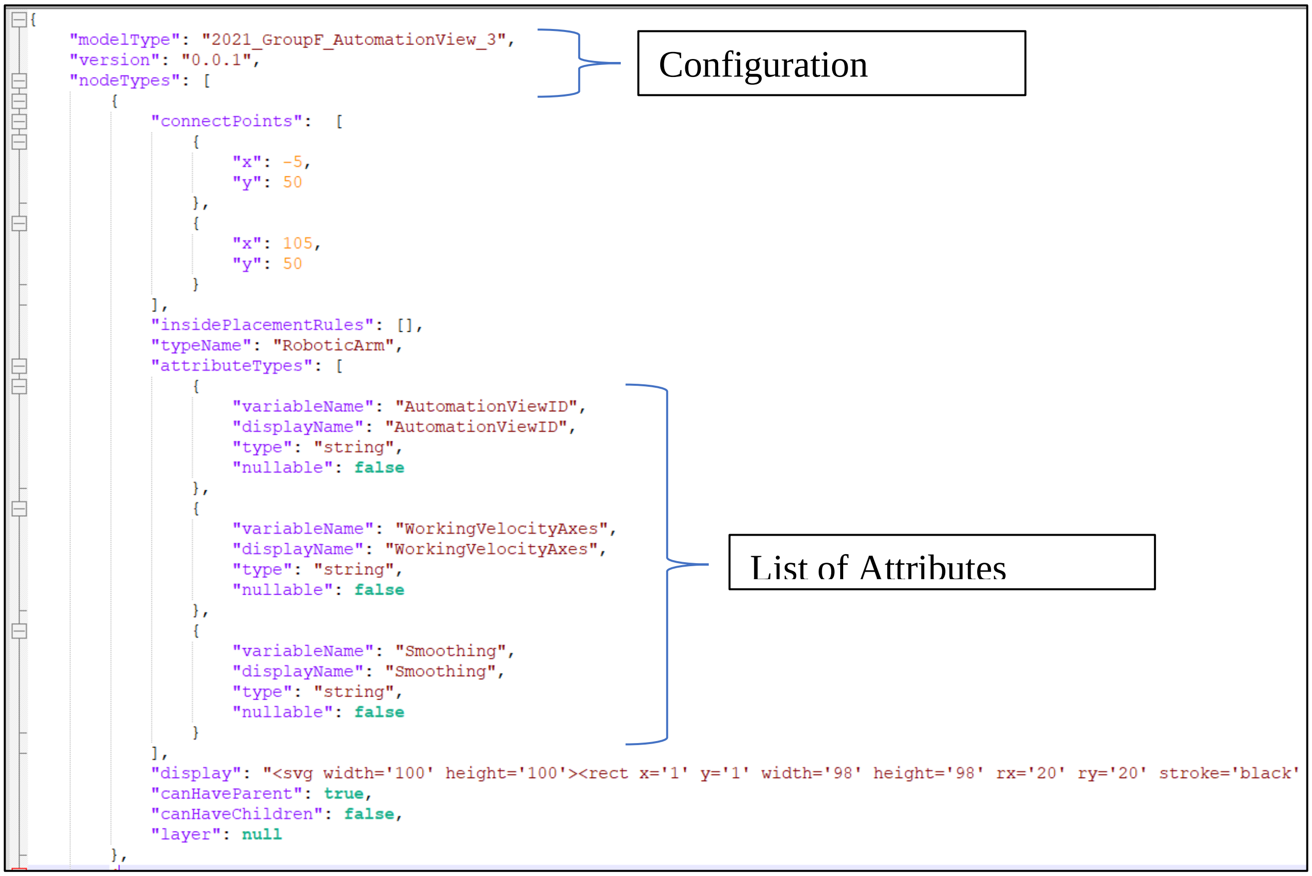

Figure 10: Structure of Node type definition within JSON configuration file

6.1.1. File semantics:

The structure of the JSON file consists of Node types and edge types(fig6). In addition, the file structure also contains basic information for the identification of configurations. Each node represents a common concept with respect to an individual engineering domain as listed during the previous design phases. The nodes contain list of attributes for describing the properties of the respective concepts. The node types will also have the information regarding the connecting points that can be exploited in the modelling stage.

```
"edgeTypes": [
    {
        "connects": [
            {
                "from": "RoboticArm",
                "fromAllowedConnectionPoints": [],
                "to": "Mixer",
                "toAllowedConnectionPoints": []
            },
            {
                "from": "Mixer",
                "fromAllowedConnectionPoints": [],
                "to": "Form",
                "toAllowedConnectionPoints": []
            },
            {
                "from": "Sprayer",
                "fromAllowedConnectionPoints": [],
                "to": "RoboticArm",
                "toAllowedConnectionPoints": []
            },
            {
                "from": "RoboticArm",
                "fromAllowedConnectionPoints": [],
                "to": "Form",
                "toAllowedConnectionPoints": []
            },

        ],
        "bidirectional": false,
        "typeName": "Unidirectional Arrow",
        "attributeTypes": [],
        "display": "<g><defs><marker id='head' orient='auto' markerWidth='8' markerHeight='8' refX='0.1'
        "canHaveParent": true,
        "canHaveChildren": false,
        "layer": null
    },
```

Edge types to define connection between concepts

Figure 11: Structure of Node type definition within JSON configuration file

The edge types represent the connection between nodes and based on the application, the edge types can be set to unidirectional or bidirectional type. The appearance of the nodes and edges can also be customized within the node descriptions. The fig.8 gives the example of a configuration file of mechanical view. This output file holds the all the concepts and their related attributes within the mechanical viewpoint.

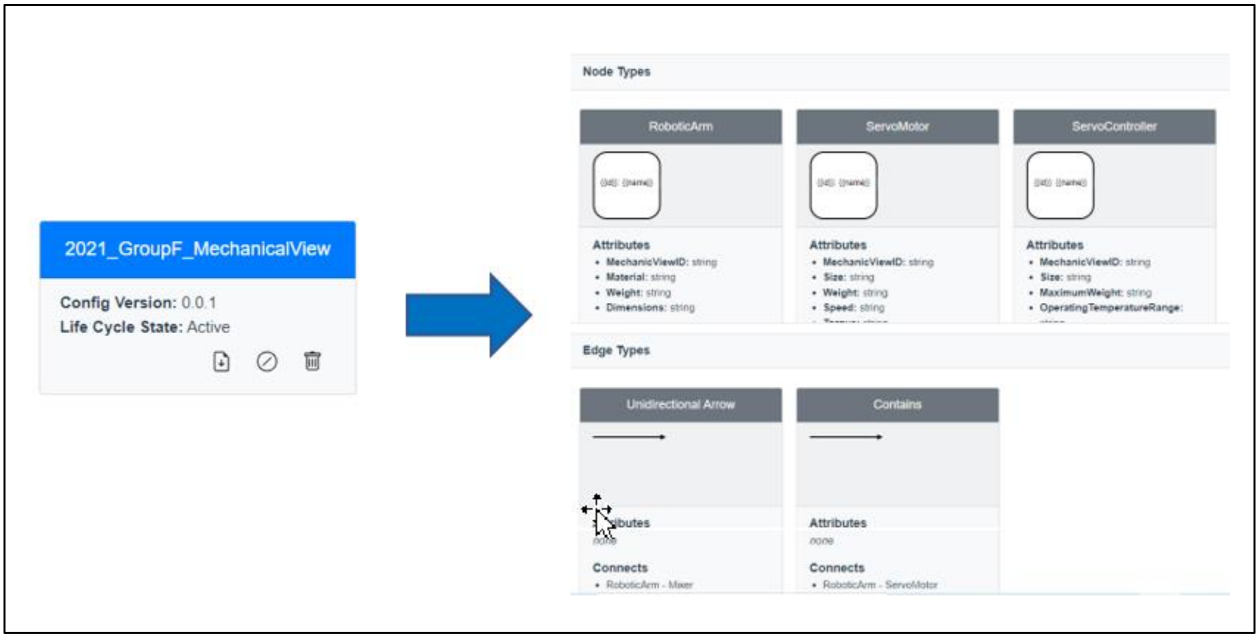

Figure 12: Output configuration file for Mechanical view

7.0. Step 6: MDRE tool data

After generating the tool configuration files in step 5, the configuration files are now used to create the models for each discipline. The design of the intended production system is created with necessary components and the relations between them.

For each view one model is generated by adding nodes which can be interconnected with edges. All the necessary attributes are filled in for each node. The output of this step is a comma separated file which contains all the engineering tool data required for the further steps.

For this step the models are created for PPR view, Function view, Mechanical view, Electrical view, and Automation View.

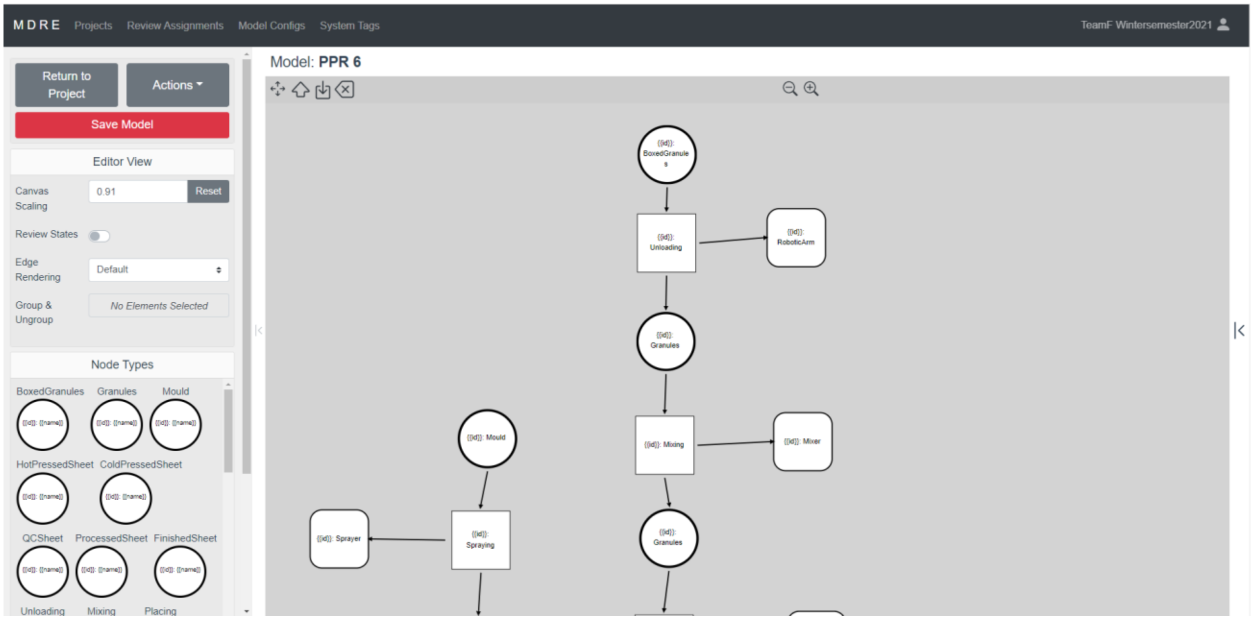

Figure 13: PPR model of production system in MDRE tool

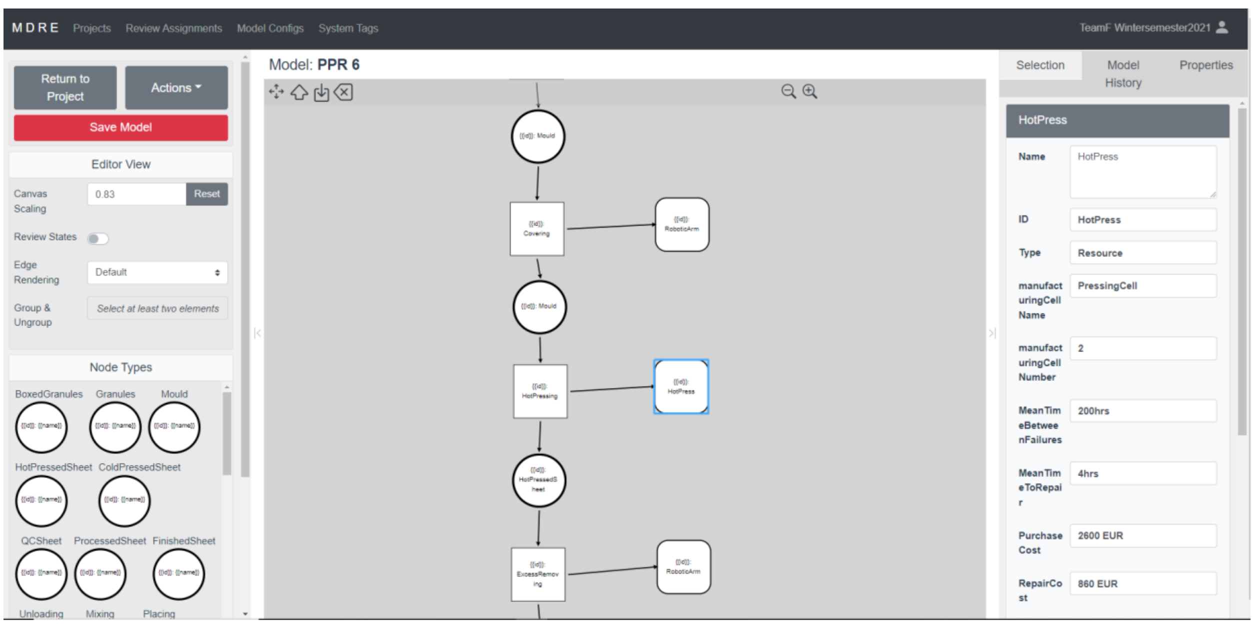

Figure 14: A node (Hot Press) with attributes in PPR model

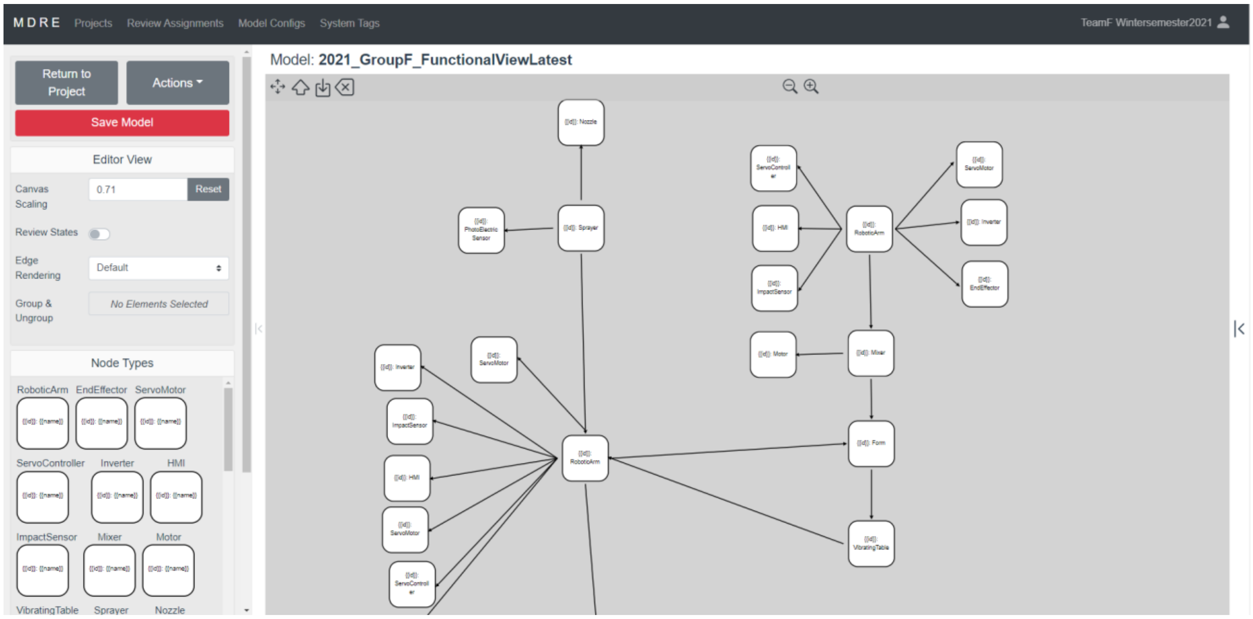

Figure 15: Function view model in MDRE tool

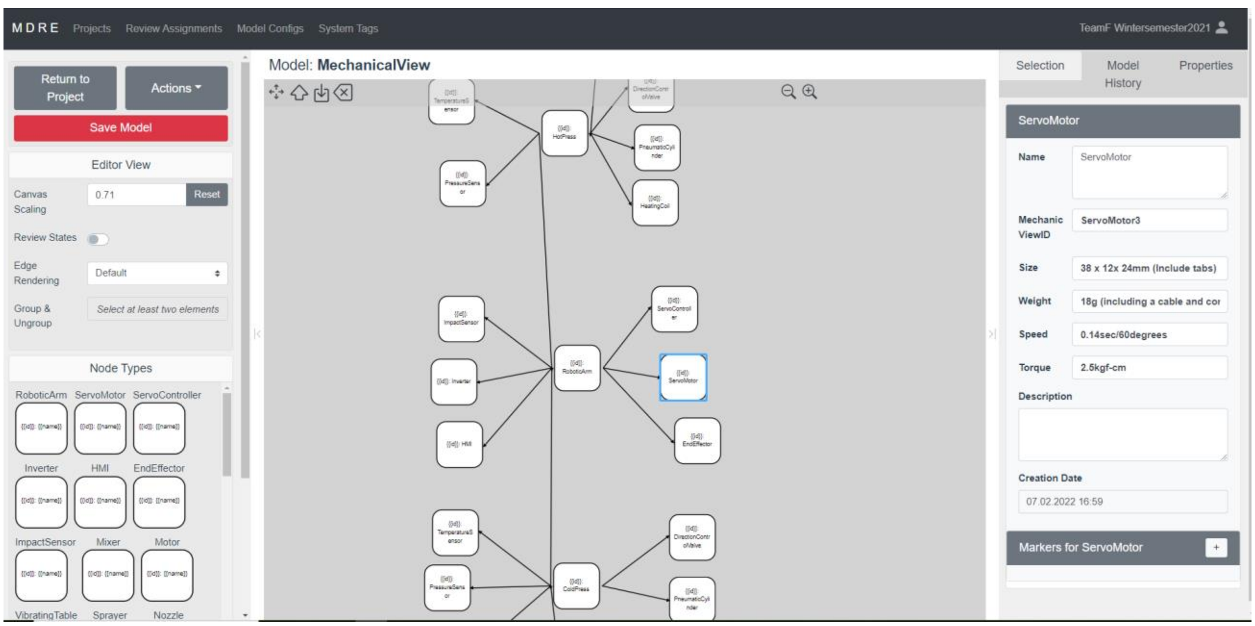

Figure 16: Mechanical view model in MDRE tool

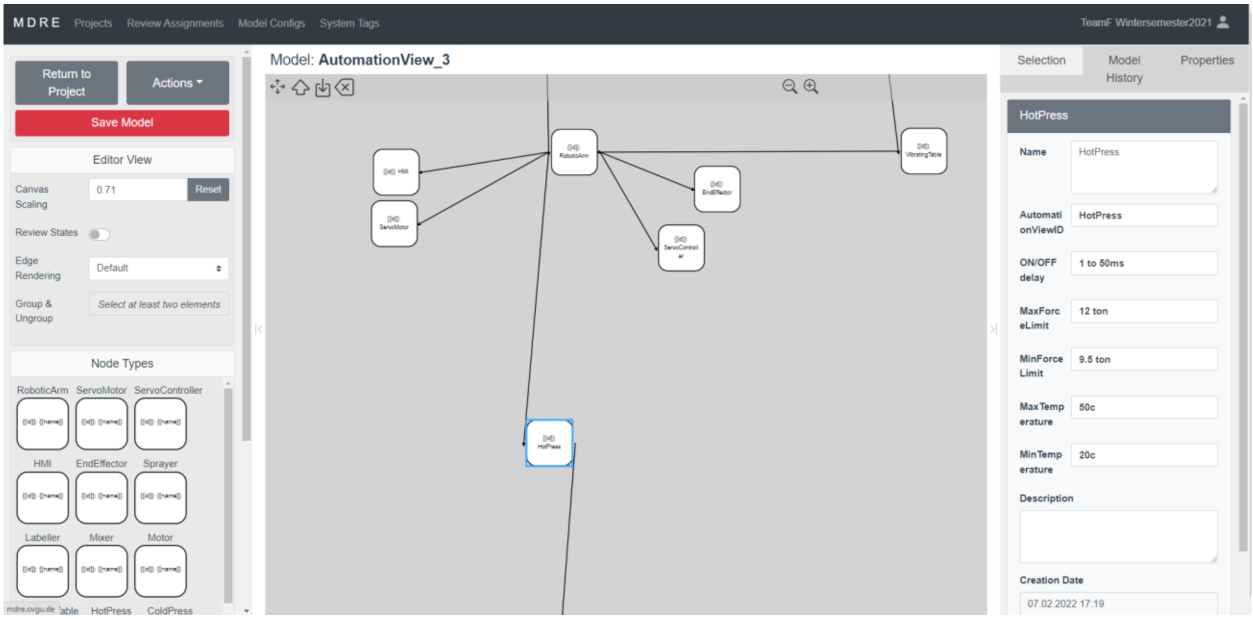

Figure 17: Automation view model in MDRE tool

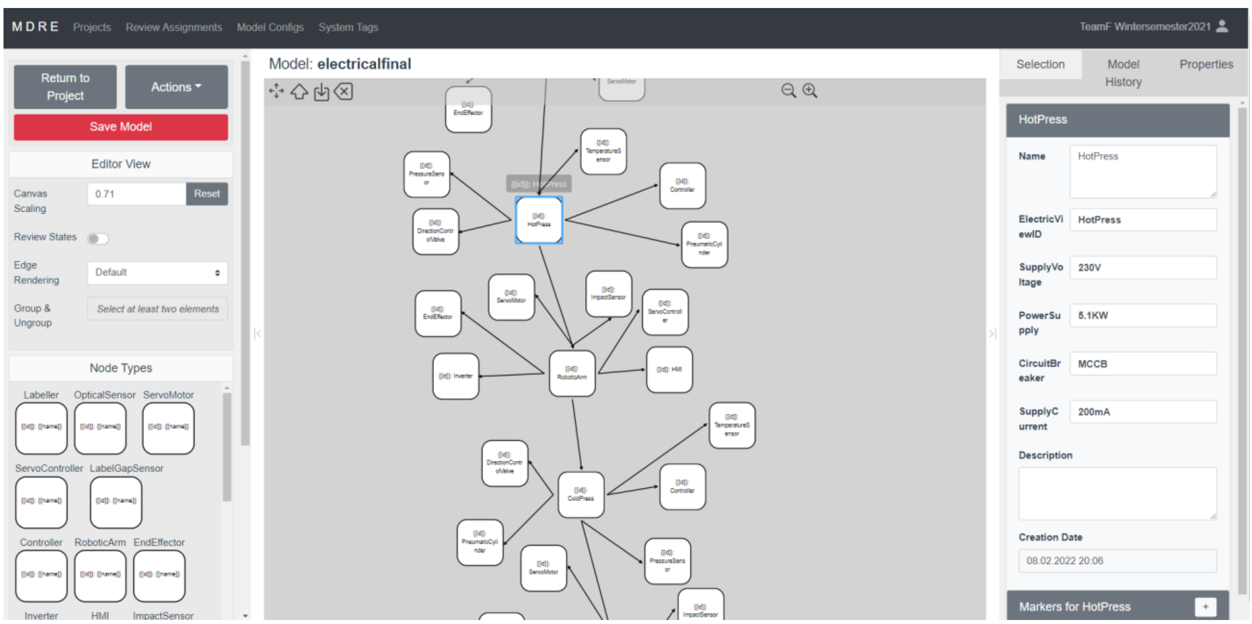

Figure 18: Electrical view model in MDRE tool

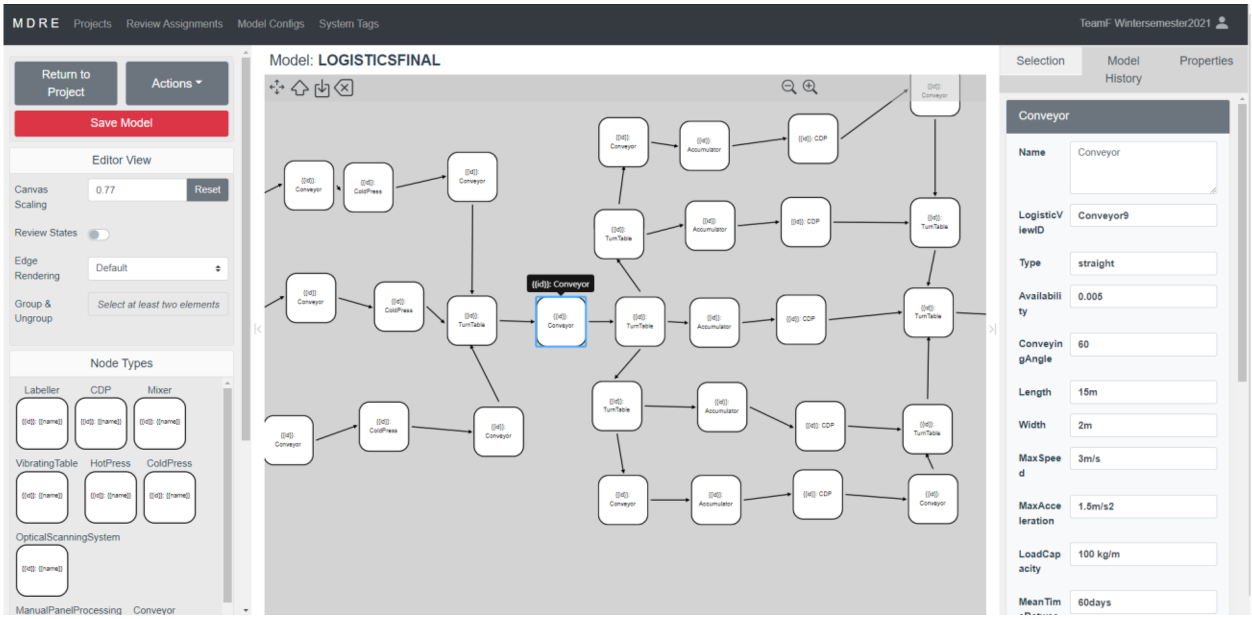

Figure 19: Logistics view model in MDRE tool


```
Model Metadata:
ModelName,ModelUUID,Created,LastModified,LinkedTo
"PPR 6","62023ed35d15c208960997dd","2022-02-08T09:58:43.398Z","2022-02-08T10:15:33.773Z",""


Node Data:

"NodeType","UUID","Name","Description","parentId","ID","Type","manufacturingCellName","manufacturingCellNumber","EstimatedCost","BoxMaterial","Length","W
"BoxedGranules","6ccd5eb9-b27b-4d0a-ad91-ecfdaad315e0","BoxedGranules","","null","ID","Product","MixingCell","1","35EUR","Cardboard","50cm","75cm","50cm"

"NodeType","UUID","Name","Description","parentId","ID","Type","manufacturingCellName","manufacturingCellNumber","Material","EstimatedCost","Density","Gra
"Granules","5ac7688a-b9b9-4ea8-a866-d47e21e9c9ff","Granules","","null","ID","Product","MixingCell","1","HDPE/PP","2 EUR/kg","0.9-0.95 gm/cm3","10 mm","1:
"Granules","9f22a8fd-a901-4098-8b52-9e8d2f352f18","Granules","","null","ID","Product","MixingCell","1","HDPE/PP","2 EUR/kg","0.9-0.95 gm/cm3","10 mm","1:

"NodeType","UUID","Name","Description","parentId","ID","Type","manufacturingCellName","manufacturingCellNumber","FormMaterial","FillMaterial","Weight","S
"Mould","a591aaad-ee07-4edd-9a67-7763f359d3a4","Mould","","null","ID","MixingCell","MixingCell","1","Aluminium","HDPE/PP","Weight","1200*1200*100 cm","10
"Mould","e94b265c-1abe-4e9d-b90a-9869777c73c6","Mould","","null","ID","MixingCell","MixingCell","1","Aluminium","HDPE/PP","Weight","1200*1200*100 cm","10
"Mould","4ab6777f-9277-46d5-8d9e-b58e2290b8be","Mould","","null","ID","MixingCell","MixingCell","1","Aluminium","HDPE/PP","Weight","1200*1200*100 cm","10
"Mould","d13e4fbf-0e12-4a01-9769-eb311f1a2341","Mould","","null","ID","MixingCell","MixingCell","1","Aluminium","HDPE/PP","Weight","1200*1200*100 cm","10
"Mould","fcd30512-5472-43cf-b7c1-ba6ac2fccb71","Mould","","null","ID","MixingCell","MixingCell","1","Aluminium","HDPE/PP","Weight","1200*1200*100 cm","10

"NodeType","UUID","Name","Description","parentId","ID","Type","manufacturingCellName","manufacturingCellNumber","Material","Length","Width","Thickness","
"HotPressedSheet","90a0abdf-fa50-4d1e-a0e6-2f0de689977f","HotPressedSheet","","null","ID","Product","PressingCell","2","HDPE/PP","100 cm","100 cm","30 mm
"HotPressedSheet","e39cb585-7b21-4627-8d2d-36199f7cfbe0","HotPressedSheet","","null","ID","Product","PressingCell","2","HDPE/PP","100 cm","100 cm","30 mm

"NodeType","UUID","Name","Description","parentId","ID","Type","manufacturingCellName","manufacturingCellNumber","Material","Length","Width","Thickness","
"ColdPressedSheet","f997dbdd-ef7d-4dff-9fd9-3d5b91062874","ColdPressedSheet","","null","ID","Product","PressingCell","2","HDPE/PP","100 cm","100 cm","30
"ColdPressedSheet","70f69c77-96e8-44bd-b3b9-e5424021b025","ColdPressedSheet","","null","ID","Product","PressingCell","2","HDPE/PP","100 cm","100 cm","30

"NodeType","UUID","Name","Description","parentId","ID","Type","manufacturingCellName","manufacturingCellNumber","PartNumber","Material","Length","Width",
"QCSheet","82f5ad16-ec15-475a-9809-c24768199935","QCSheet","","null","ID","Product","ScanningCell","manufacturingCellNumber","PN01","HDPE/PP","100 cm","1
"QCSheet","9c933a33-2d6e-43c8-8900-fb804387a742","QCSheet","","null","ID","Product","ScanningCell","manufacturingCellNumber","PN01","HDPE/PP","100 cm","1

"NodeType","UUID","Name","Description","parentId","ID","Type","manufacturingCellName","manufacturingCellNumber","Material","Length","Width","Thickness","
"ProcessedSheet","aad89b62-2fd2-43e1-8d18-384c3a153f64","ProcessedSheet","","null","ID","Product","CDPCell","4","HDPE/PP","100 cm","100 cm","30 mm","30 k
"ProcessedSheet","60bfee77-7f22-4352-a97e-564e5f96301f","ProcessedSheet","","null","ID","Product","CDPCell","4","HDPE/PP","100 cm","100 cm","30 mm","30 k
"ProcessedSheet","64191bae-f02b-47cc-bf22-412a263517b8","ProcessedSheet","","null","ID","Product","CDPCell","4","HDPE/PP","100 cm","100 cm","30 mm","30 k
"ProcessedSheet","47d4a2c9-8d25-44a3-882c-8db120aabc43","ProcessedSheet","","null","ID","Product","CDPCell","4","HDPE/PP","100 cm","100 cm","30 mm","30 k
```


Figure 20: Step 6 output - MDRE export CSV file for PPR view

8.0. Step 7: Data Pipeline configuration

The engineering data generated from the MDRE tool export is used here to transform into local Data Specific Languages (DSLs). The transformers are used to translate the tool data from the csv file to the respective AML2 generated in step 4. The aim is to instantiate the System unit classes generated in step 4 to create an instance hierarchy.

Input:

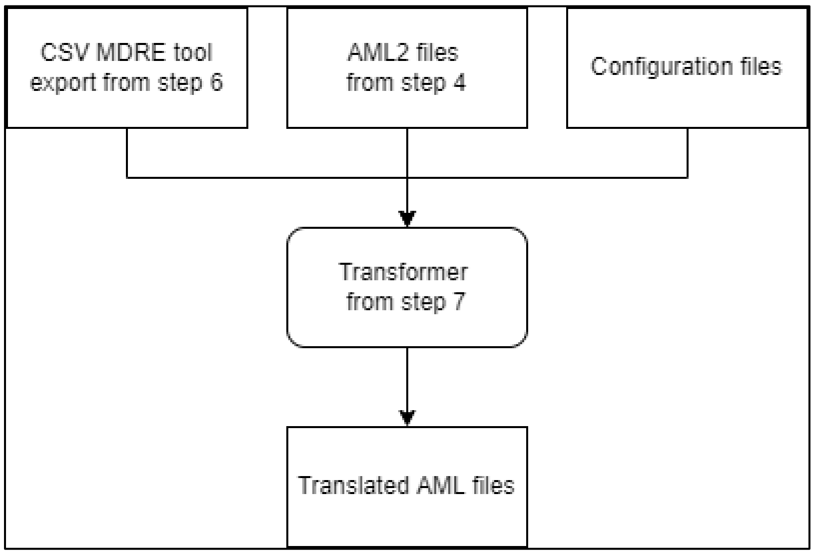

Figure 21: Step 7 inputs and outputs

Output: The transformer instantiates the objects using the SUCL. As seen in figure below the robotic arm gets instantiated into different robotic arms with different attributes. The attributes now have values like ID: KUKA5 and so on.

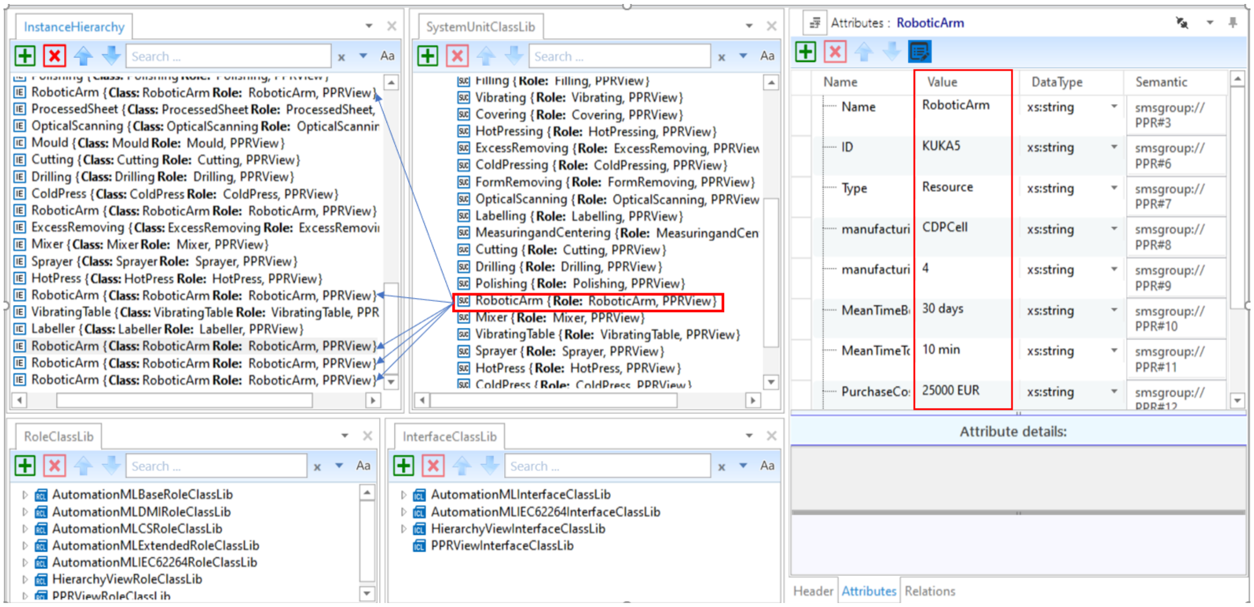

Figure 22: Step 7 translated AML file

9.0. Step 8: Integration

After generating system hierarchy for each view in step 7, the main purpose of this step is to integrate all 6 different views (function, mechanical, PPR, electrical, automation and logistic) into one common, global AML file which contains not only all the views but also their inter-relations. There are 5 sub-steps in total which will be described in detail in the followings:

9.1. Step 8: sub-step 1 -2: Pre-process the AML

This is a technical step which translates the local tool artifacts into engineering artifacts that are suitable for the local DSL. This step is to ensure that the local AML file for each view is suitable to be transformed into global AML file for the next sub-step.

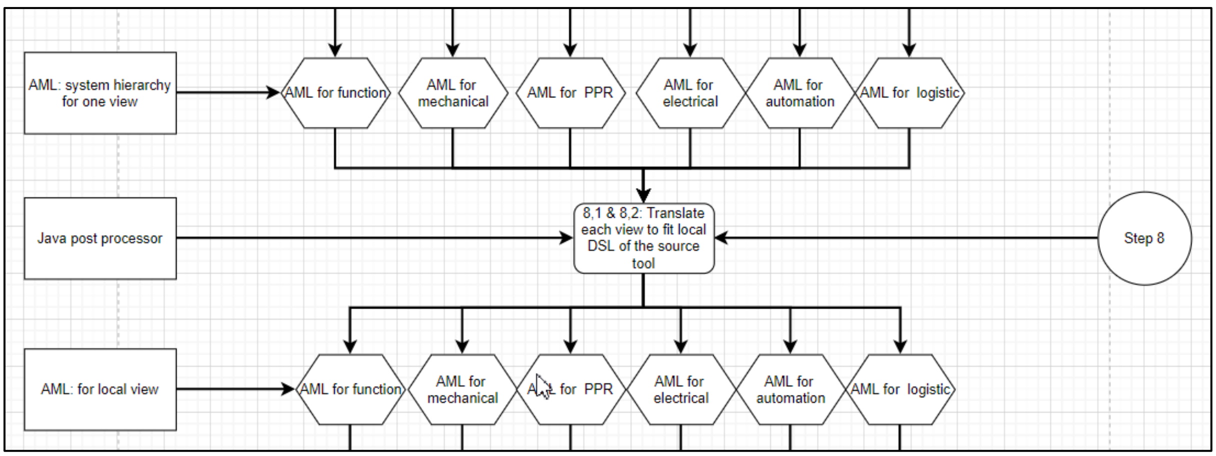

Figure 23: Step 8 sub-step 1-2

9.2. Step 8: sub-step 3: Transforming local to global DSL

The next sub-step is to map the artefacts from the local DSL to global DSL (AML file).

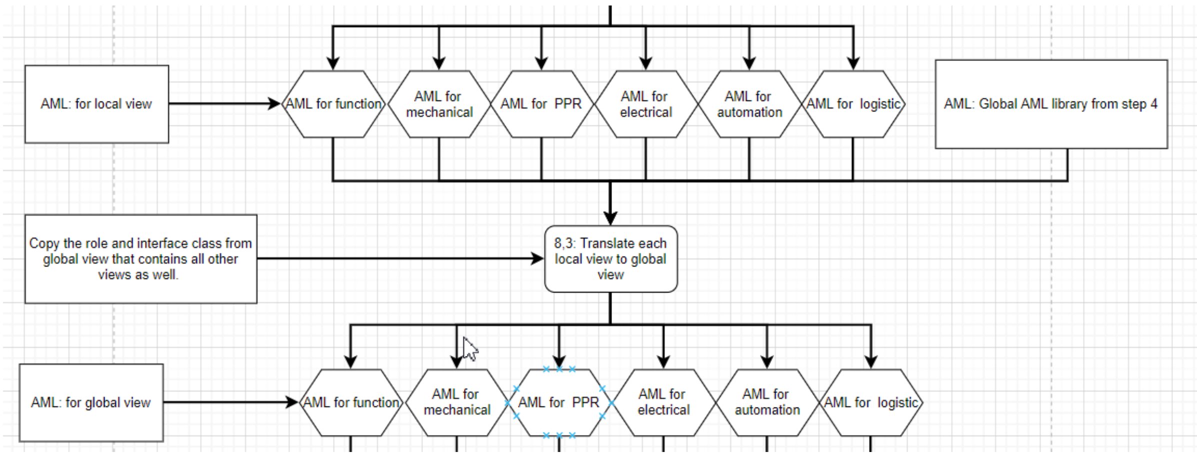

Figure 24: Step 8 sub-step 3

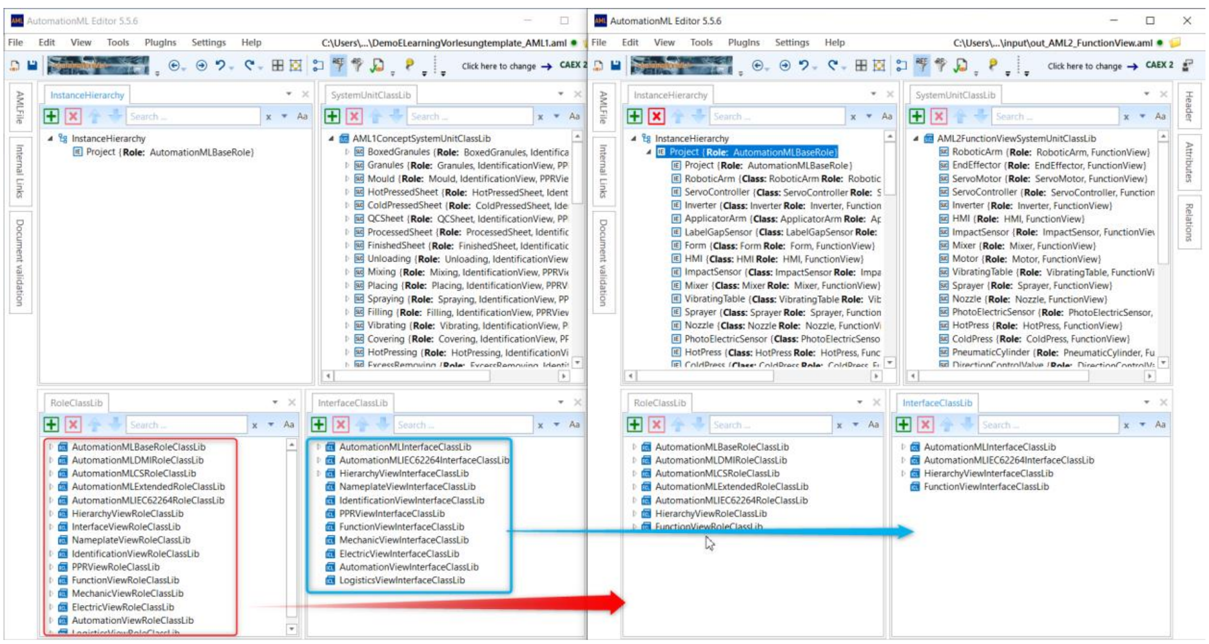

Figure 25: Mapping global class into each local view

By mapping the role and interface class from a global AML template (left side) to the local AML view (right side), an AML based structure for the global DSL was generated as shown below:

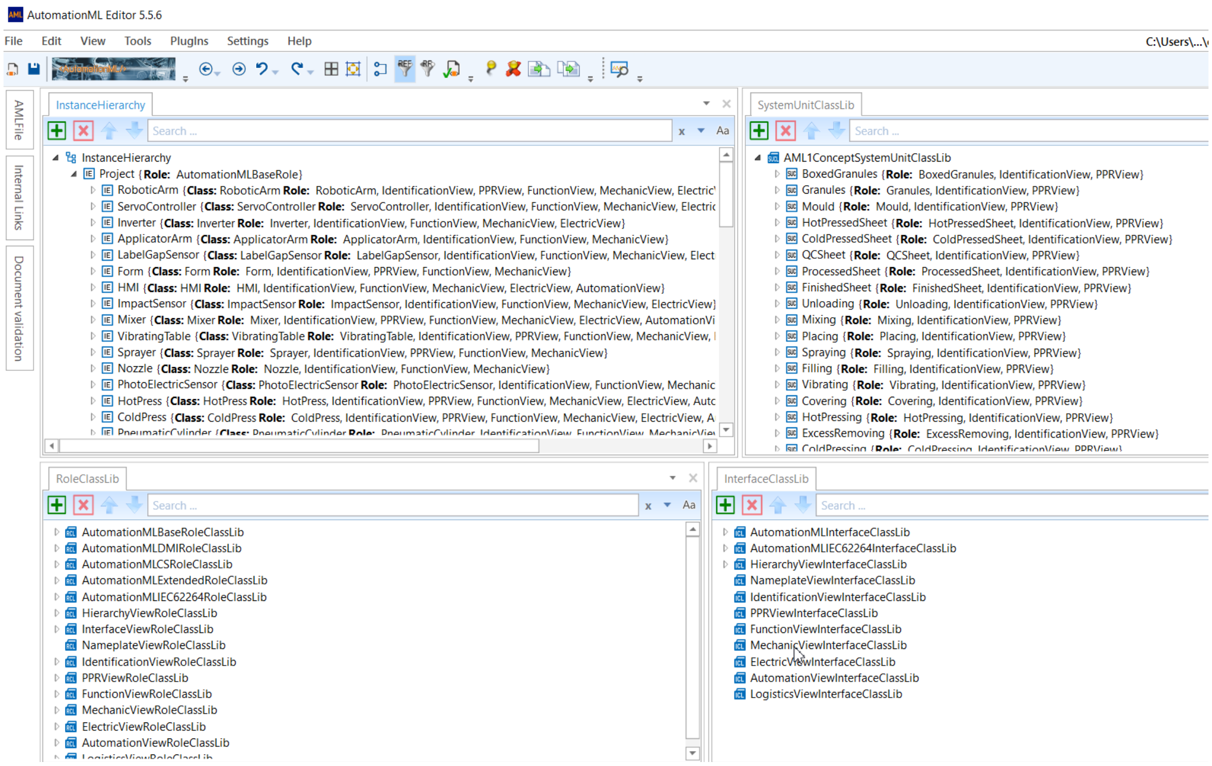

Figure 26: AML based structure for global DSL

As shown in the figure above, the AML file is now containing global roles, interfaces, and system unit class.

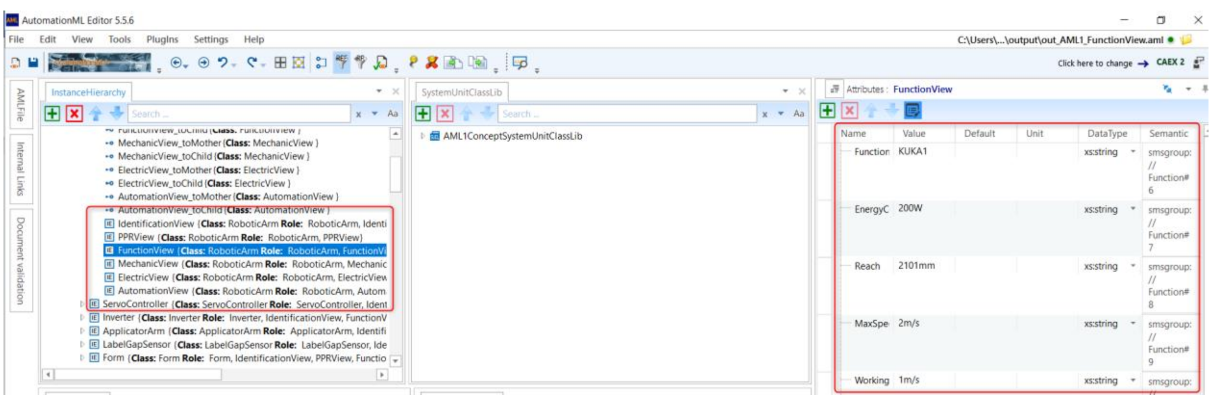

Figure 27: Views with attributes

After the transformation, each view contains its own attributes as well.

9.3. Step 8 sub-step 4: Integration

After that, all views were integrated using a Java based processor to generate one local DSL, AML file which contains all the attribute values from all views.

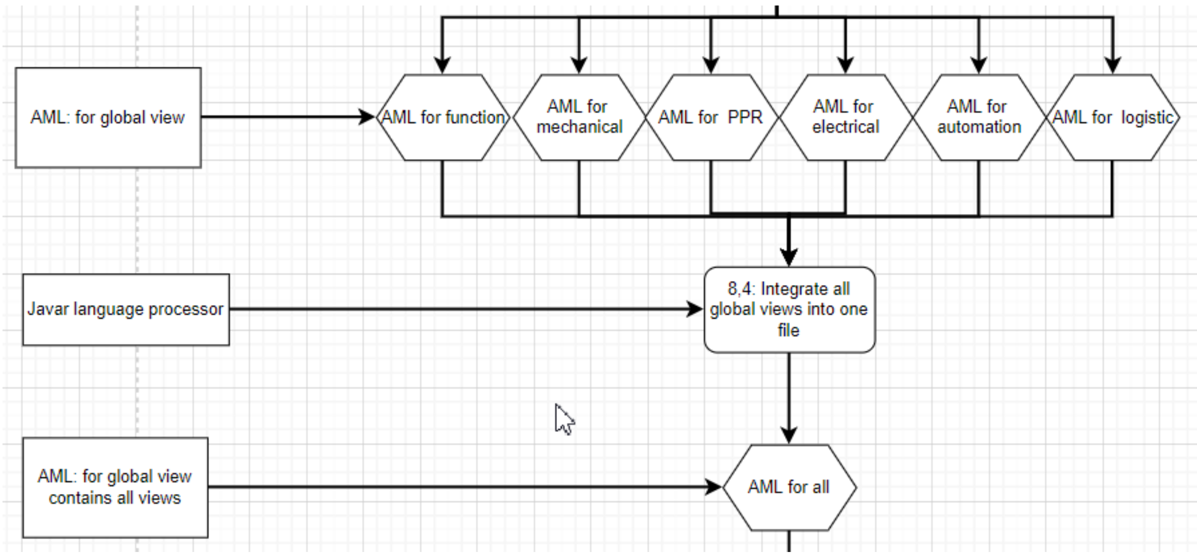

Figure 28: Step 8 sub-step 4

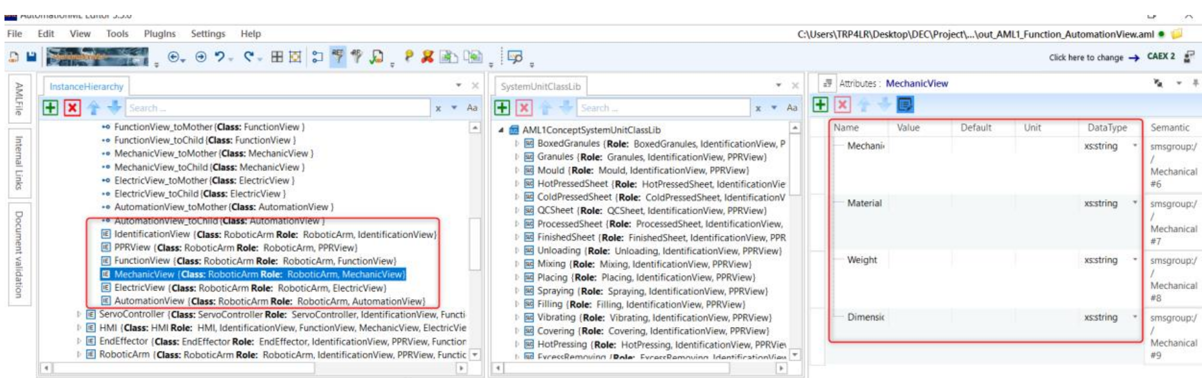

Figure 29: All attributes for all views

9.4. Step 8 sub-step 5: Tracing

The final sub-step is to use a Java based processor to also define the inter-relations between different views. Now that there is one complete AML file that contains all views, their attributes and relations, the next step is to link such engineering artifacts with asset administration tools.

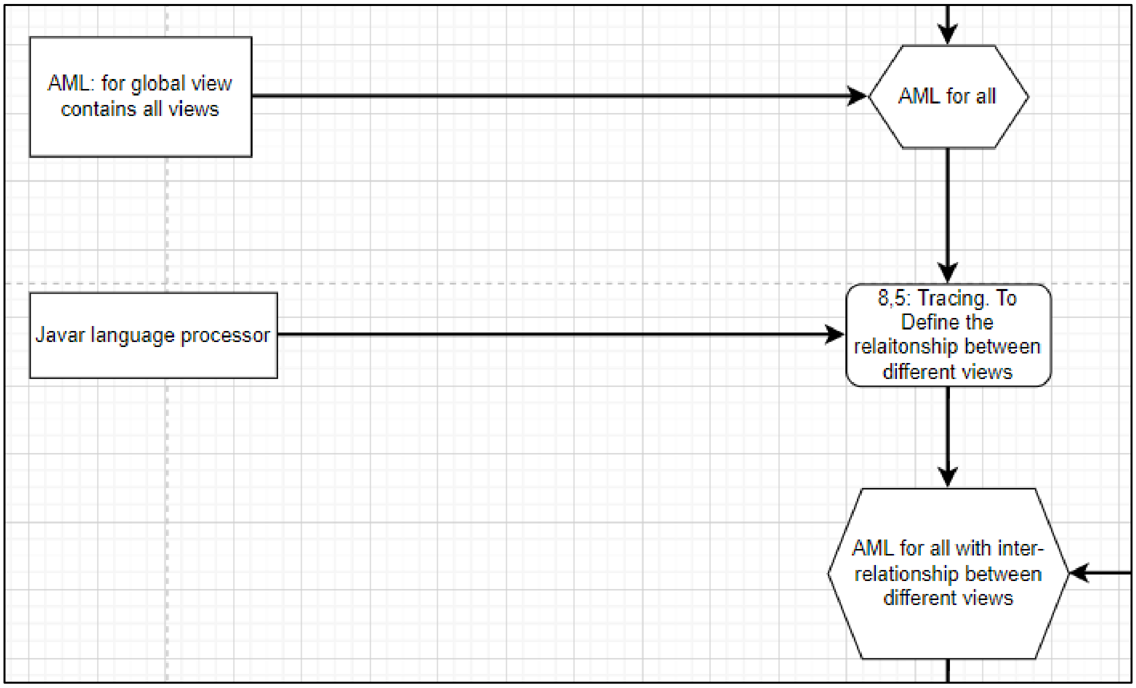

Figure 30: Step 8 sub-step 5: Tracing

10.0. Step 9: AAS

The purpose of this final step is to link the engineering artifacts to AAS as assets. Such assets can be linked to a cyber-physical production system. Since all subsystems and its relationships are defined, making one change can be linked to another. This makes the production system more flexible and creates reusable artifacts.

There are two main sub-steps which are described below:

10.1. Step 9 sub-step 1: AAS mapping

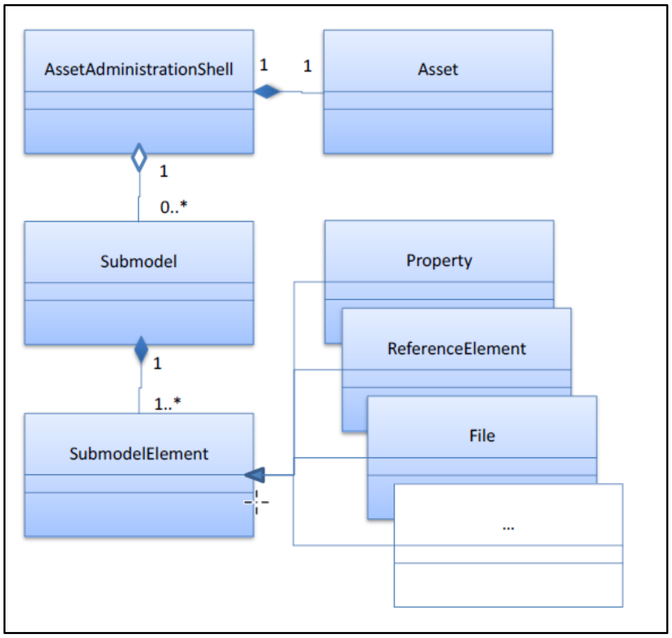

Figure 31: AAS mapping structure

As shown in the figure above, for AAS mapping, each concept must be its own project (right below system hierarchy) and thus all its attributes shall become the sub-model.

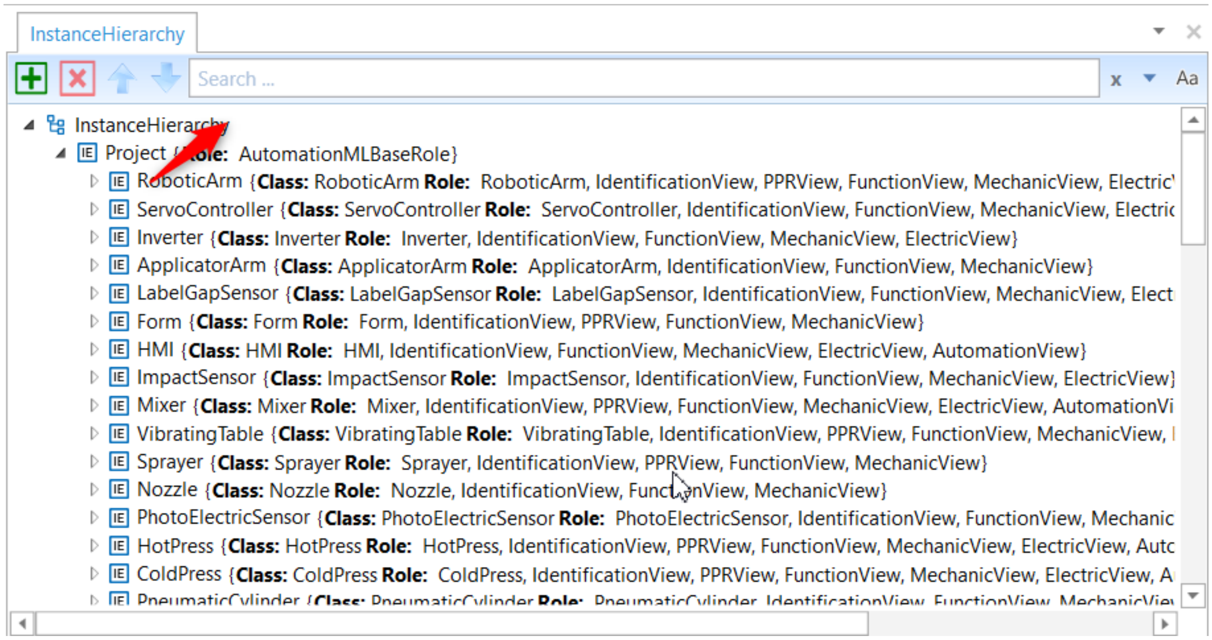

Figure 32: Moving concepts into Instance Hierarchy

Since there are too many concepts to be moved, only a few concepts out of each process, product and resource will be chosen for AAS mapping. In this project, 10 total concepts were chosen including

1. Hot-press
2. Granules
3. Turntable
4. OpticalScanningSystem
5. VibratingTable
6. HMI
7. FinishedSheet
8. Accumulator
9. Cutting
10. RoboticArm

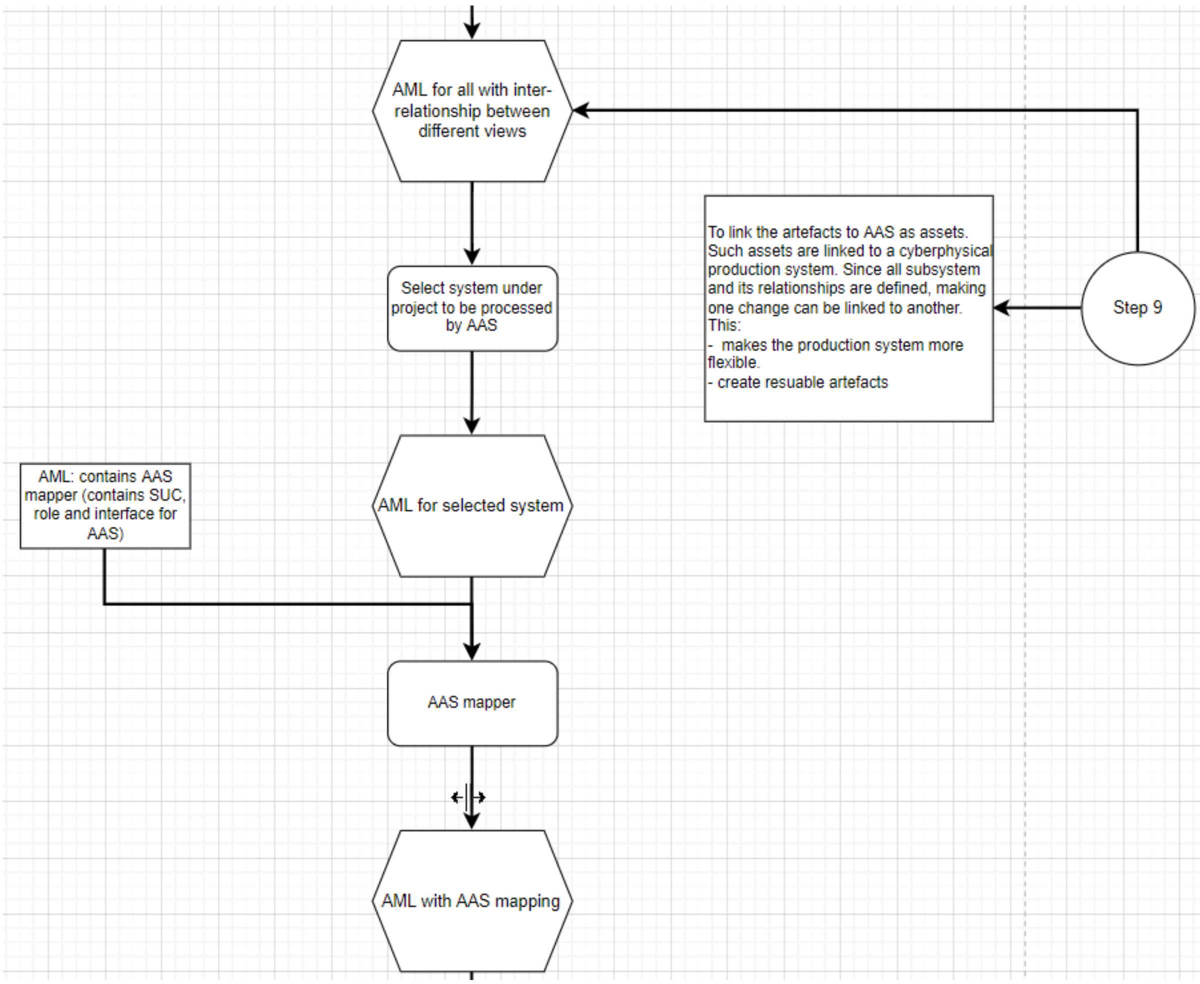

Figure 33: Step 9

After selecting the 10 concepts, an AML which contains AAS system unit, role and interface were imported into the AML file and mapped. After mapping, the AML file contains AAS attributes as shown below:

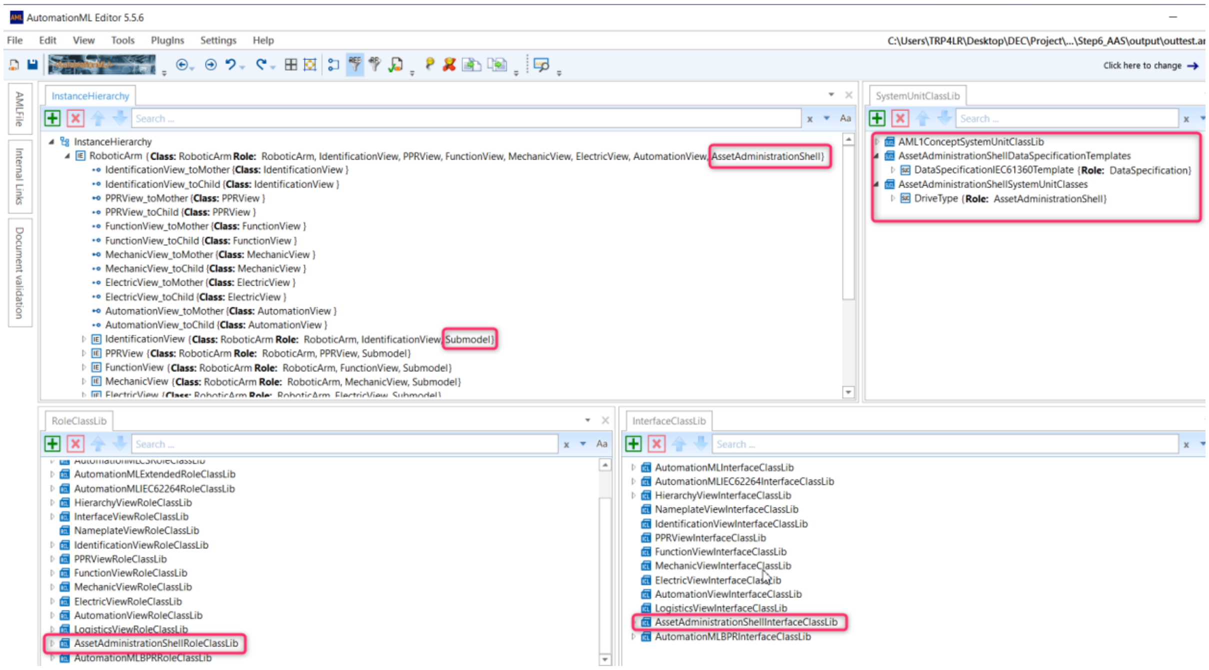

Figure 34: AAS mapping output

10.2. Step 9 sub-step 2: AAS running

Sub-step 2 uses AAS runtime from BaSys 4.0 to run the AML file on the website through its local host. The output is in html format where users can change the value directly.

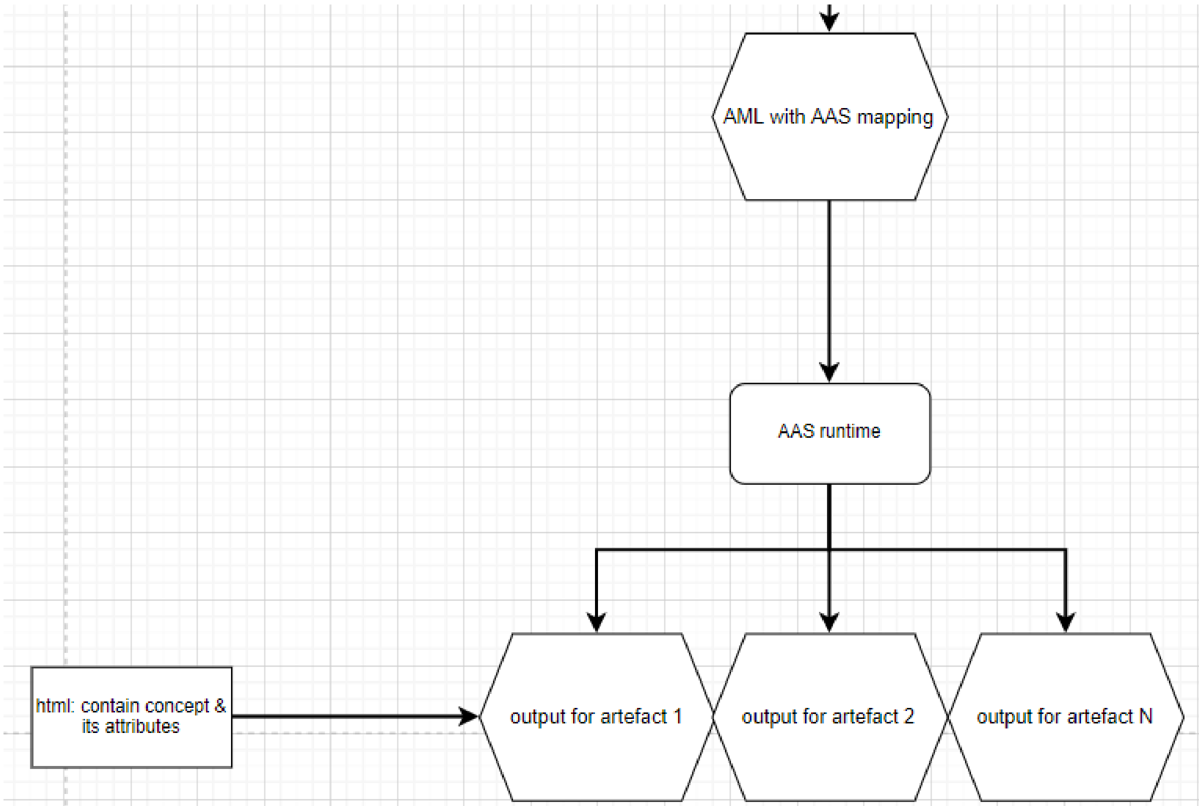

Figure 35: Step 9 sub-step 2

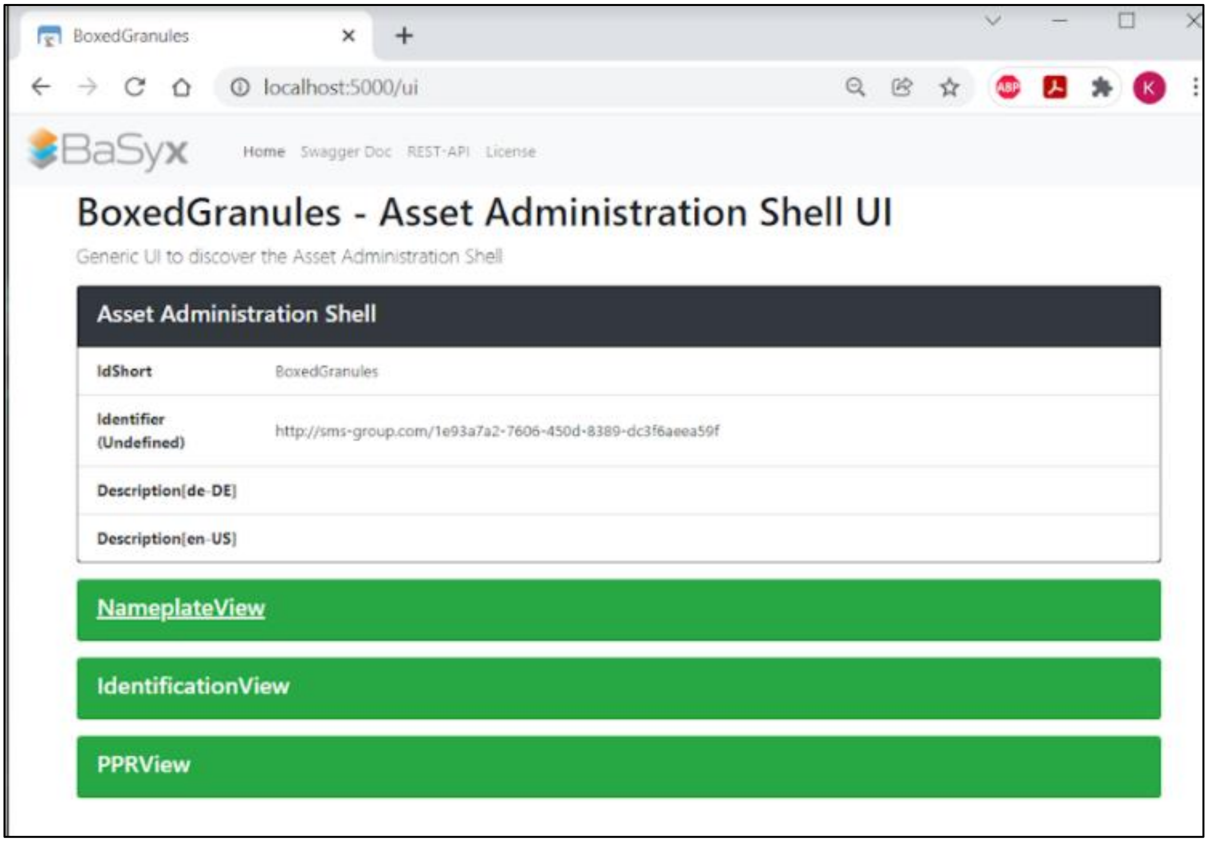

Figure 36: Output after AAS runtime

11.0. Step 10: Adding logistic view and improvements

What need to be added to integrate logistics elements

What need to be improved to update the results to increase pipeline quality

Step 1: PPR, BOO, BOM

- PPR: grouped the processes into different cells. The product handling between cells is now handled by logistic elements (conveyor, accumulator, turntable)

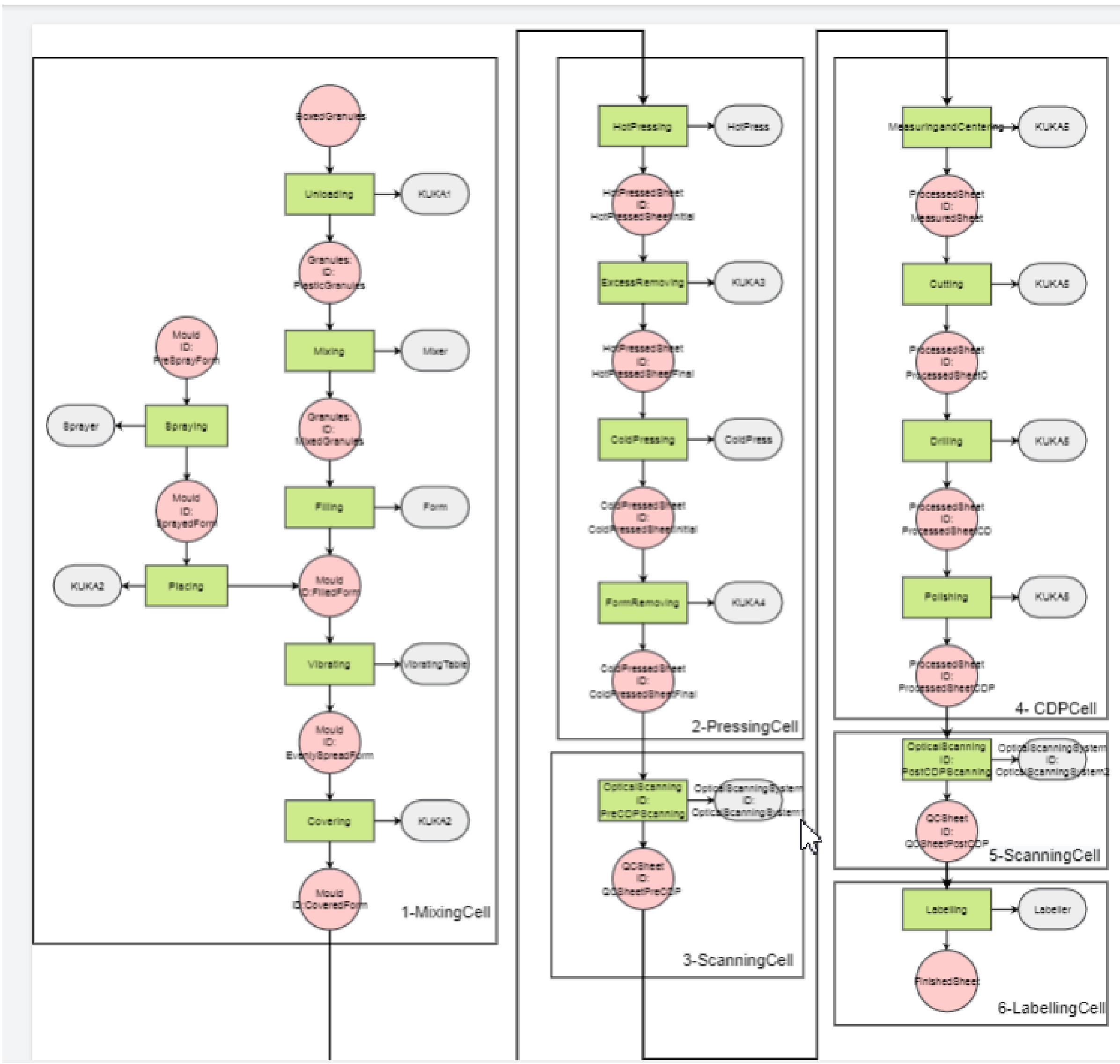

Figure 37: Manufacturing cells and Logistic view

- BOO & BOM: added logistic resources

Step 3: Concept identification:

- Added logistic view for resources that are connected to the logistic element. For instance, for HotPress, ColdPress, VibratingTable and Mixer, besides the other 5 views, logistic view is added to accommodate the logistic layout provided by the logistic team.
- Added 3 new logistic concepts: Conveyor, Accumulator, TurnTable. The attributes were provided by the logistic team.
- **Improvements**: to add higher level cells that contain resources (Mixing cell which contain robotic arm and mixer). Therefore, changes could be done within the mixing cell, for example, instead of robotic arm, another actuator is used. This change, however, will not affect the overall plant design; mixing cell remained in the overall plant concept. Furthermore, each cell could provide a throughput. This will improve logistic planning.

Step 4 & 5: Convert common concepts to data models

- Logistic views and concept were added to the common YAML file and configuration JSON file.

Step 6: MDRE tool data

- Layout given by the logistic team was duplicated on MDRE tool
- Values to logistic attributes were given
- The data provided by logistics team was updated in each objects of logistics view
- **Improvements:** apply the higher-level abstraction from improvements in step 3 to define the relationship between cells and its contents. Abstracting the concepts even more could help in reducing the number of concepts. As a result, creating new objects or modifying existing concepts will be easier.

Step 7: Data pipeline configuration

- **Improvements**: unique names (that have different length) should be given to the concepts so that less iterations needed to be done.

Step 8 & 9: Integration & AAS

- Added logistic view

12.0. Conclusion

In the end, our objectives have been met; the engineering data pipeline was successfully transferred as inputs for Assets Administration Shell creating reusable artifacts as well as a flexible cyber-physical production system. Moreover, data for different product, process and resources and their various views (logistic, function, automation, etc.) were managed; the inter-relations between different engineering disciplines were defined as well using MDRE and AML transformers. Such data logistic and management are of the essence for future industry 4.0 where cyber-physical systems are getting more complex.

13.0. References

[1]"Plastics: Material-Specific Data | US EPA", US EPA, 2022. [Online]. Available: https://www.epa.gov/facts-and-figures-about-materials-waste-and-recycling/plastics-material-specific-data.

[2]"Polypropylene Recycling - An Introduction", The Balance Small Business, 2022. [Online]. Available: https://www.thebalancesmb.com/an-overview-of-polypropylene-recycling-2877863.

[3]S. Companies, "HDPE Recycling: How Is HDPE Recycled? | Scranton Products", Scranton Products, 2022. [Online]. Available: https://www.scrantonproducts.com/how-do-they-recycle-hdpe/.

[4]"Automation-driven recycling of plastic: when technology meets sustainability | synTQ", synTQ, 2022. [Online]. Available: https://www.syntq.com/automation-driven-recycling-of-plastic-when-technology-meets-sustainability/.

[5]"3 Reasons Why Automation Is a Boon to The Plastic Industry", Plastivision, 2022. [Online]. Available: http://www.plastivision.org/blog/3-reasons-why-automation-is-a-boon-to-the-plastic-industry/#:~:text=Robotic%20automation%20offers%20flexibility%20to,needs%20and%20changing%20market%20trends.&text=Apart%20from%20reducing%20labour%20costs,your%20overall%20cost%20of%20production.

[6]"Plattform-i40.de. 2022. Asset Administration Shell - Reading Guide. [online] Available at: <https://www.plattform-i40.de/IP/Redaktion/EN/Downloads/Publikation/Asset_Administration_Shell_Reading_Guide.html> [Accessed 13 February 2022].

February 2022].

[7]"Plattform-i40.de. 2022. Asset Administration Shell - Reading Guide. [online] Available at: <https://www.plattform-i40.de/IP/Redaktion/EN/Downloads/Publikation/Asset_Administration_Shell_Reading_Guide.html> [Accessed 13 February 2022].

[8]"KR IONTEC | KUKA AG", KUKA AG, 2022. [Online]. Available: https://www.kuka.com/en-de/products/robot-systems/industrial-robots/kr-iontec.