University of Stuttgart
Institute of Industrial Automation
and Software Engineering

Semantic modelling of machine skills and automated matching between user requests and executable skills by applying neural language models

Master Thesis presentation

Manthan Venkataramana Shenoy

Study Program: M.Sc. INFOTECH

GPT-controlled automation system

„GPT4Automation“

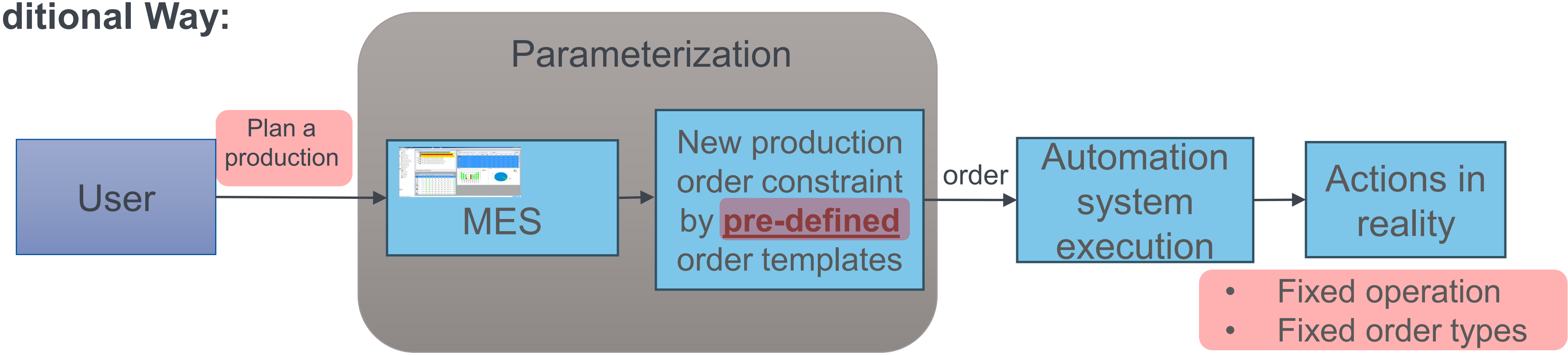
Agenda

- Motivation
- Result-Preview
- Challenges
- State-of-art research
  - Agent-System
  - Skill Modelling
  - Semantic Modelling
- Designed System
- Implementation Result

Motivation: Flexible manufacturing

Why we want to use GPT to control the automation system?

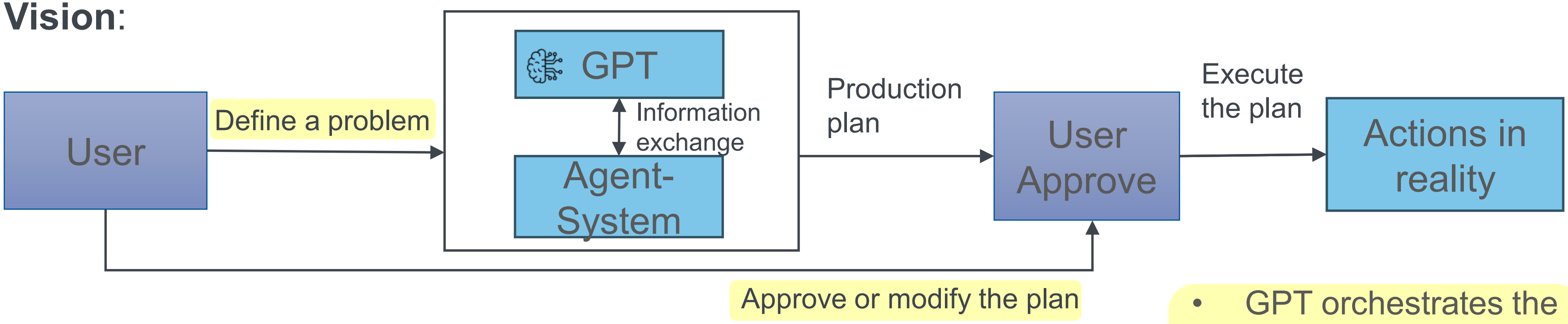
Traditional Way:



User
Plan a production
Parameterization
MES
New production order constraint by **pre-defined** order templates
order
Automation system execution
Actions in reality

- Fixed operation
- Fixed order types

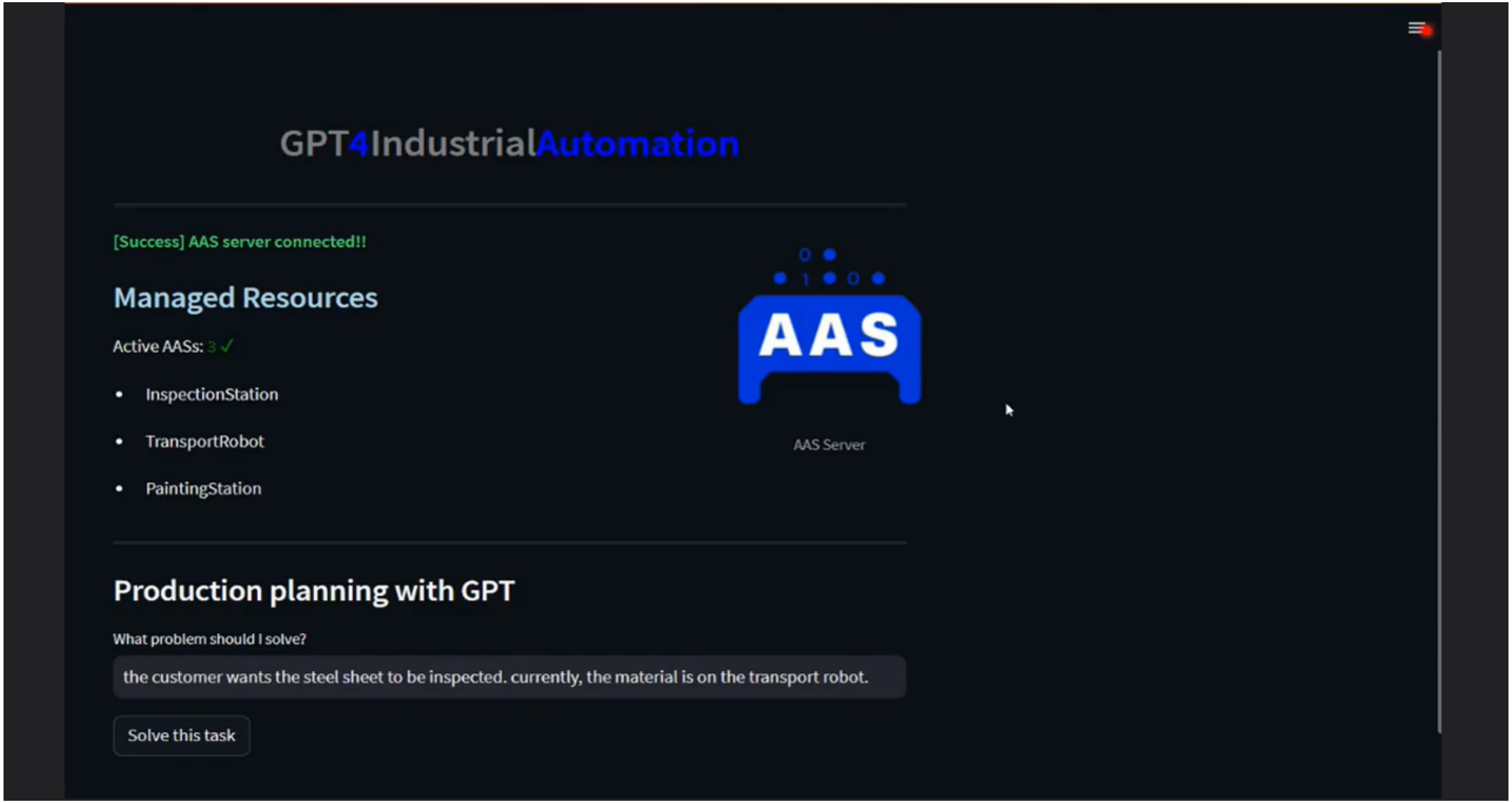
Our Vision:



- GPT orchestrates the production
- Automated problem solving

Result - Preview

GPT4Automation



GPT4IndustrialAutomation

[Success] AAS server connected!!

Managed Resources

Active AASs: 3 ✓

- InspectionStation
- TransportRobot
- PaintingStation

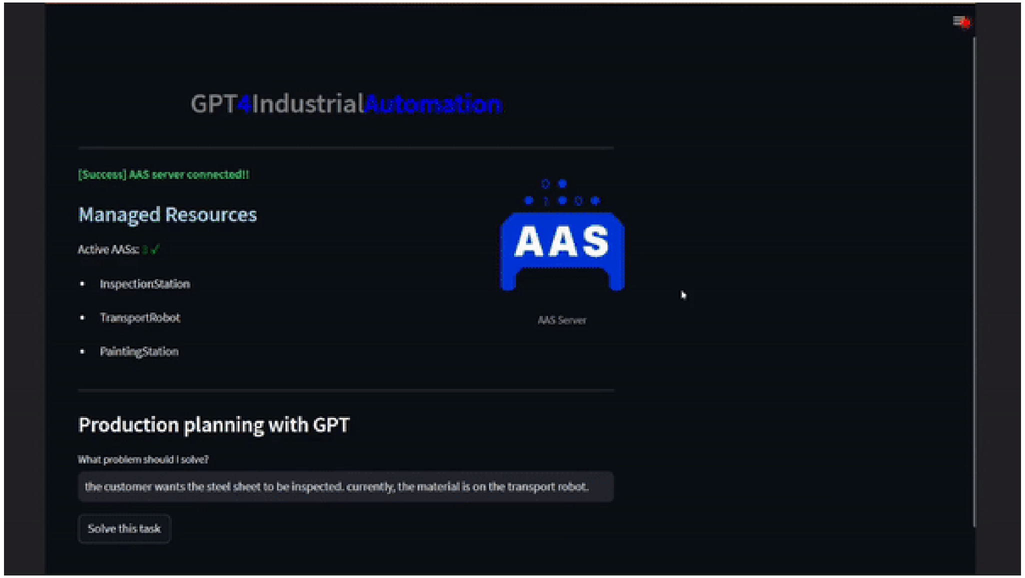
Production planning with GPT

What problem should I solve?

the customer wants the steel sheet to be inspected. currently, the material is on the transport robot.

Let GPT control the production

Motivation and the benefits



GPT4IndustrialAutomation

[Success] AAS server connected!!

Managed Resources

Active AASs: 3 ✓

- InspectionStation
- TransportRobot
- PaintingStation

Production planning with GPT

What problem should I solve?

the customer wants the steel sheet to be inspected. currently, the material is on the transport robot.

New system:

- **Customized** production plan generated by GPT
- GPT provides the **knowledge** required to solve a production planning problem

How was it made?

Technical challenges that have been tackled

Challenges

State-of-the-art research topics

Challenge 1:
Control interfaces

SOTA research topic 1:
Skill modelling

Challenge 2:
GPT must understand the system

SOTA research topic 2:
Semantic modelling

Challenge 3:
Execution of the actions

SOTA research topic 3:
Agent-system

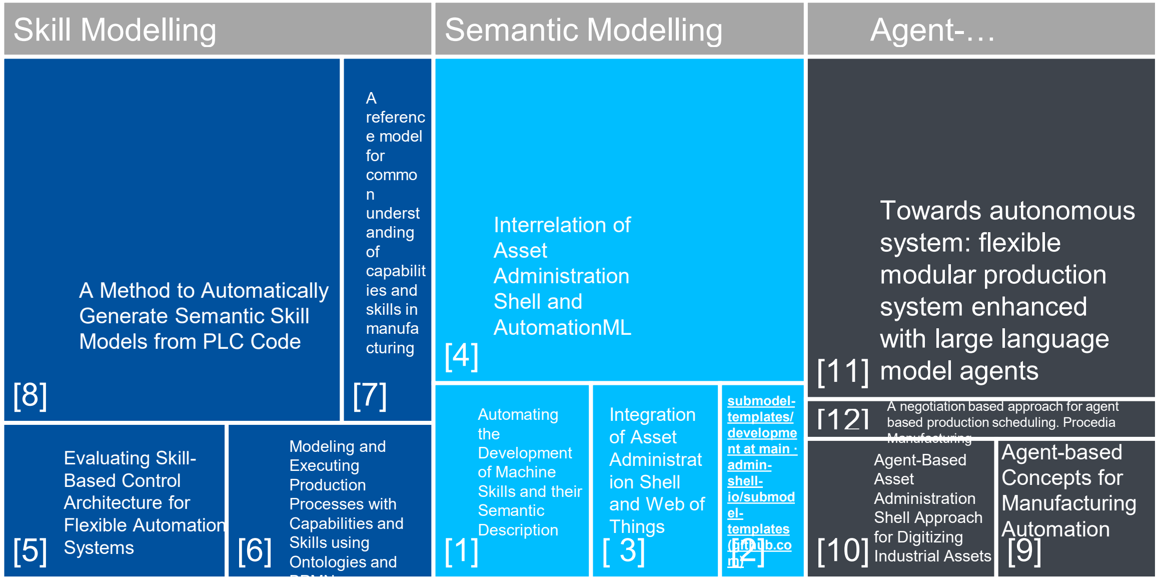
Challenge 4:
Overall: Flexible production

SOTA research topic 4:
Industy 4.0
Digital twins (AAS)

Literature Review

- Semantic modelling
  - [1], [2], [3], [4]
- Skill modelling
  - [5],[6],[7],[8]
- Agent-system modelling
  - [9],[10],[11],[12]
- Industy 4.0 Digital twins (AAS)
  - [13],[14]

Research Areas



Skill Modelling

[8] A Method to Automatically Generate Semantic Skill Models from PLC Code

[7] A reference model for common understanding of capabilities and skills in manufacturing

[5] Evaluating Skill-Based Control Architecture for Flexible Automation Systems

[6] Modeling and Executing Production Processes with Capabilities and Skills using Ontologies and BPMN

Semantic Modelling

[4] Interrelation of Asset Administration Shell and AutomationML

[1] Automating the Development of Machine Skills and their Semantic Description

[3] Integration of Asset Administration Shell and Web of Things

[2] submodel-templates/development at main · admin-shell-io/submodel-templates (github.com)

Agent-…

[11] Towards autonomous system: flexible modular production system enhanced with large language model agents

[12] A negotiation based approach for agent based production scheduling. Procedia Manufacturing

[10] Agent-Based Asset Administration Shell Approach for Digitizing Industrial Assets

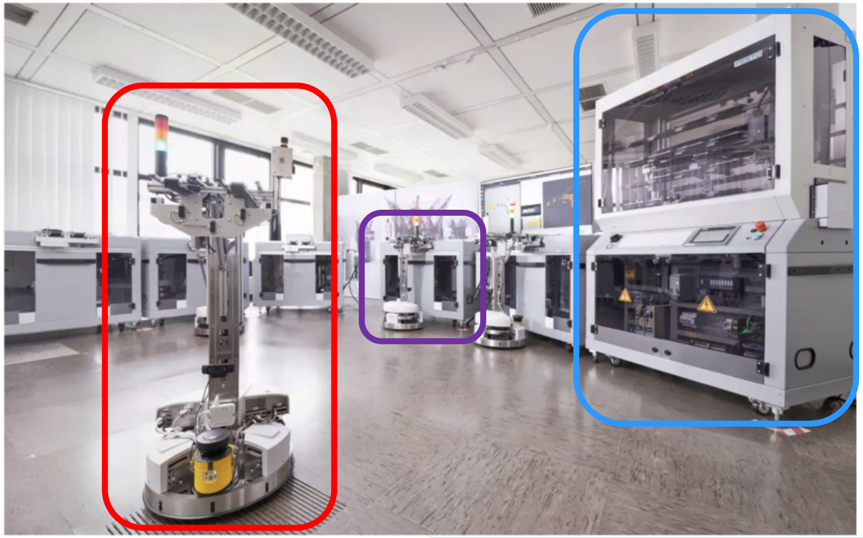
[9] Agent-based Concepts for Manufacturing Automation

- Active in these research fields:
  - Team of Univ.-Prof. Dr.-Ing. Alexander Fay
  - Team of Prof. Dr.-Ing. Christian Diedrich
  - Team of Prof. Dr.-Ing. Dr. h. c. Michael Weyrich

Based on this foundation

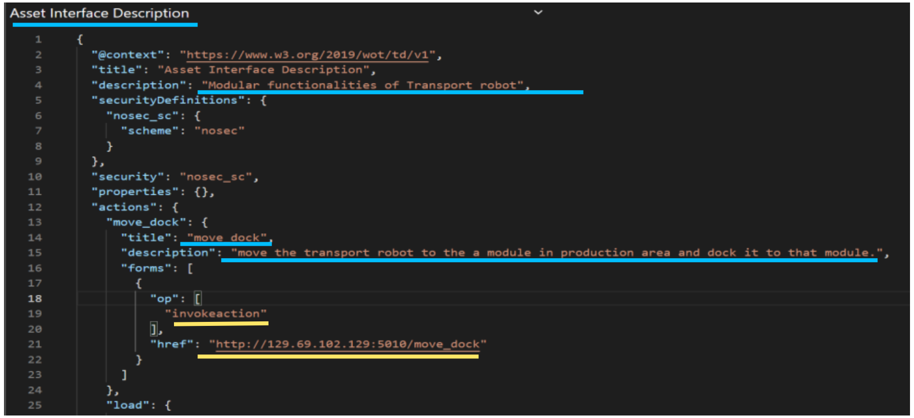
GPT for industrial automation system

- Agent-system modelling
  - Autonomous decision making
  - Expose the control interfaces
- Skills modelling
  - Control interfaces as skills
- Semantic modelling
  - Interfaces is semantically annotated
  - Let GPT understand the production
- Flexible production in context of I4.0
  - New research direction



How skills looks like

How semantics in skill model looks like



Asset Interface Description

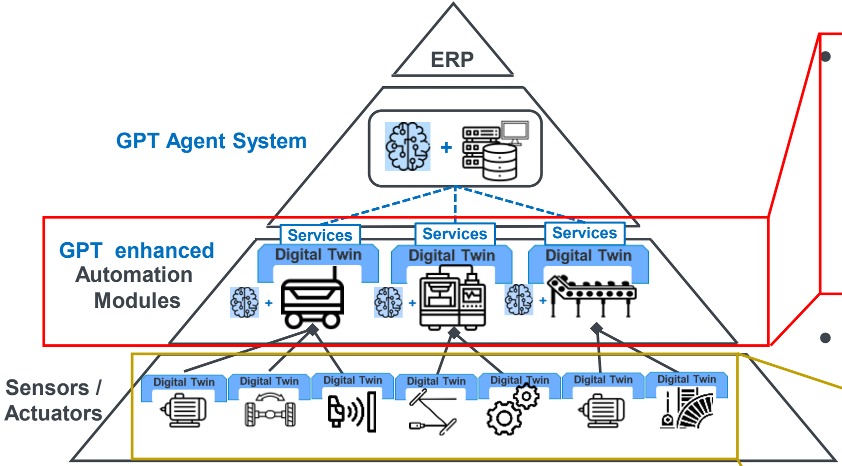
```
{
  "@context": "https://www.w3.org/2019/wot/td/v1",
  "title": "Asset Interface Description",
  "description": "Modular functionalities of Transport robot",
  "securityDefinitions": {
    "nosec_sc": {
      "scheme": "nosec"
    }
  },
  "security": "nosec_sc",
  "properties": {},
  "actions": {
    "move_dock": {
      "title": "move_dock",
      "description": "move the transport robot to the a module in production area and dock it to that module.",
      "forms": [
        {
          "op": [
            "invokeaction"
          ],
          "href": "http://129.69.102.129:5010/move_dock"
        }
      ]
    },
    "load": {
```

3 important aspects

- Agent-System
- Skill Modelling
- Semantic modelling

Agent-system

Automation modules is modeled as agents



[12]

- Control service interfaces:
  - Skills: (for general planning)

Skill 1: Transport workpiece
Skill 2: Leave production area
…

Skill 1: Print a pattern
Skill 2: Coat a workpiece
…

  - Functionalities: (for detailed planning)

F1: Move_dock()
F2: load()
F3: unload()
F4: dock()
F5: Undock()
…

F1: controlConveyor(id=1, start,...)
F2: controlStopper(id=2 , start, print..)
F3: controlConveyor(id=1, stop,...)
....
F7: controlJunction(close)
…

3 important aspects

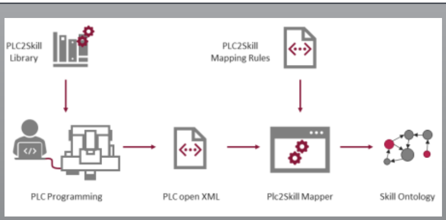
- Agent-System
- **Skill Modelling**
- Semantic modelling

Related works

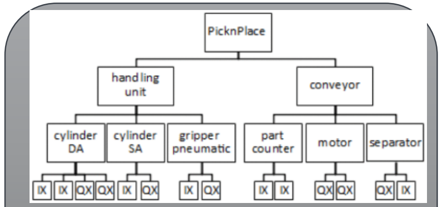
On Skill Modeling

- PLC2Skill [8]
  - Model the skills based on the existing PLC-Code (IEC 61131-10)
- IAS Skill [5]
  - Function orchestration
  - Modeled using state-machine with IEC 61499
- Hierarchical control Architecture [5]
  - Multi-layered control, may use IEC 61499
- Service-oriented Architecture (SOA) [7][5]
  - Skills can be modeled with OPC UA

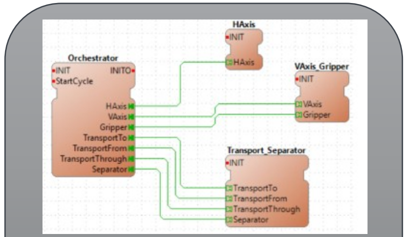
    or REST API as service



PLC2SKill



Hierarchical control

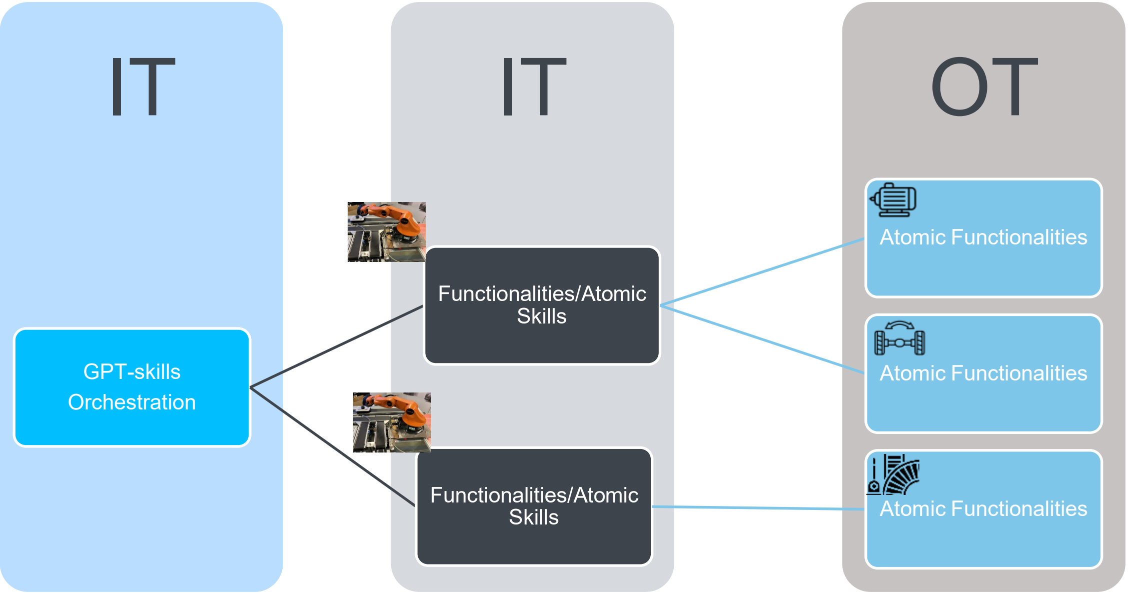


Integrated Assembly Solutions (IAS) Skill

Designed Skill Modelling Architecture

From the aspect of IT&OT

- Hierarchical control Approach using SOA



IT

GPT-skills Orchestration

IT

Functionalities/Atomic Skills

Functionalities/Atomic Skills

OT

Atomic Functionalities

Atomic Functionalities

Atomic Functionalities

The task: Inspect and paint the workpiece

controlStopper (id=1,state= “down” , param = “fault”)

Opcua-client.get_node('ns=3;s="xW1_MB50“')
Opcua-client.get_node('ns=3;s="xW1_BG21“')
.....

{REST:API}

OPC UA

3 important aspects

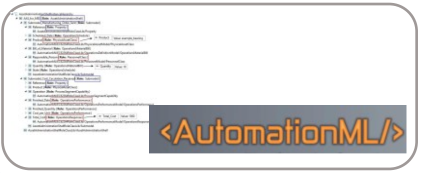
- Agent-System
- Skill modelling
- Semantic modelling

Related Works

On Semantic Modelling

- Semantic modelling with [4] AutomationML
  - Semantics comes from AuotmationML
- Ontology Design Pattern [1][6][7]
  - Designed for PLC2skill, also can be used with other skill frameworks
- AAS IEC 63278-1 [2][3]
  - Metamodelling techniques
  - Asset Interface Description

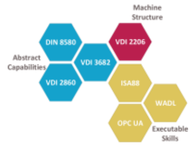
AAS submodel template



- PLC Code information
- Function Block information
- Asset Structure Information

PLCOpen XML

- Function Block information
- Concept definition

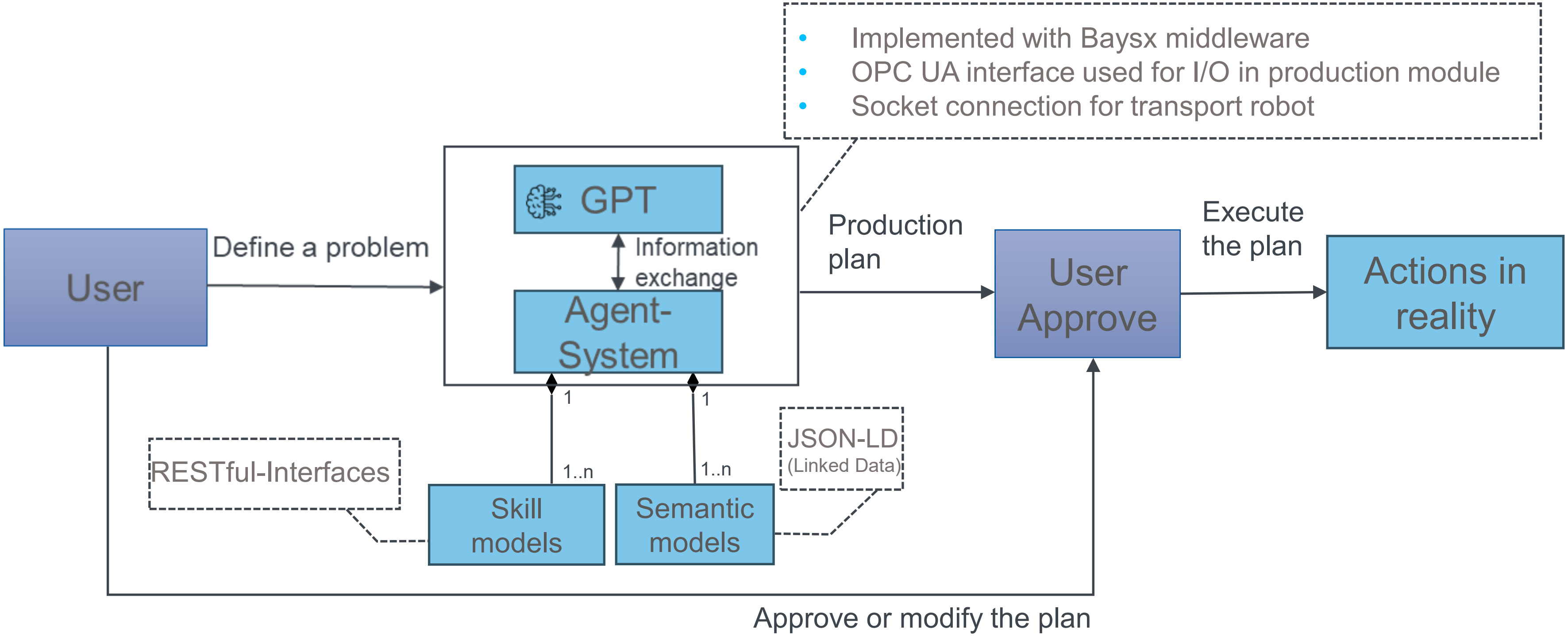


Ontology Descriptions

e.g.: PLC2Skill (uses IEC 61131-10)

System Design

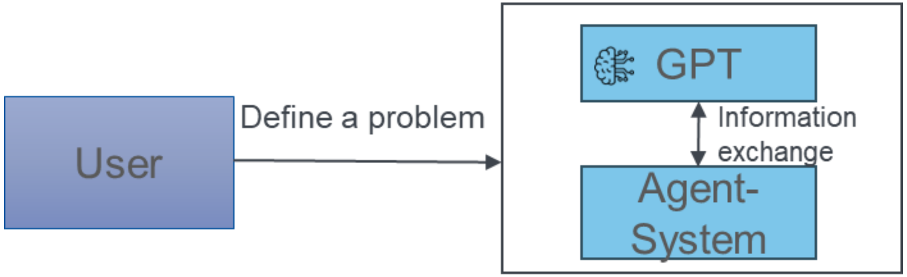
GPT4Automation control workflow

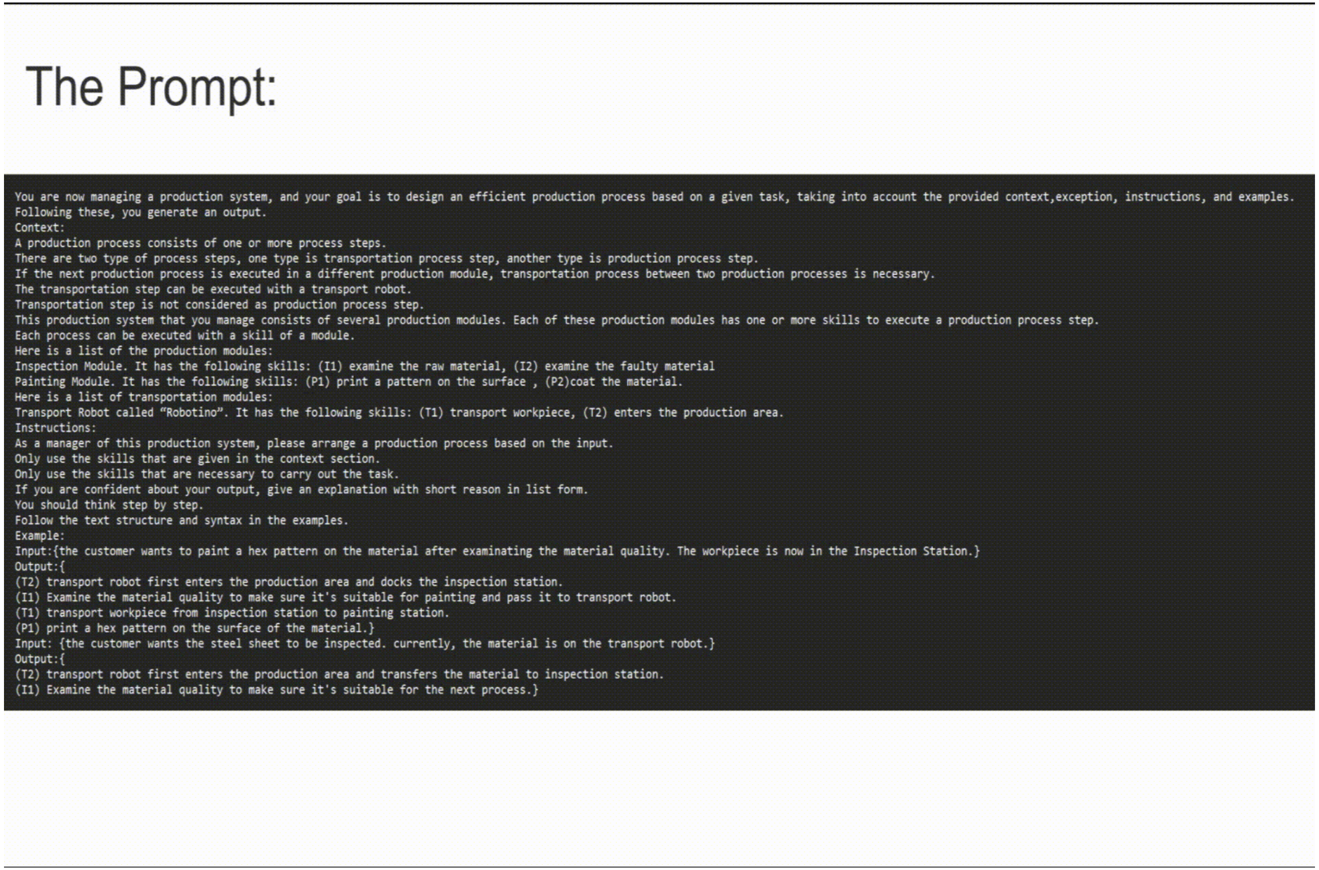


- Implemented with Baysx middleware
- OPC UA interface used for I/O in production module
- Socket connection for transport robot

How does the prompting works?

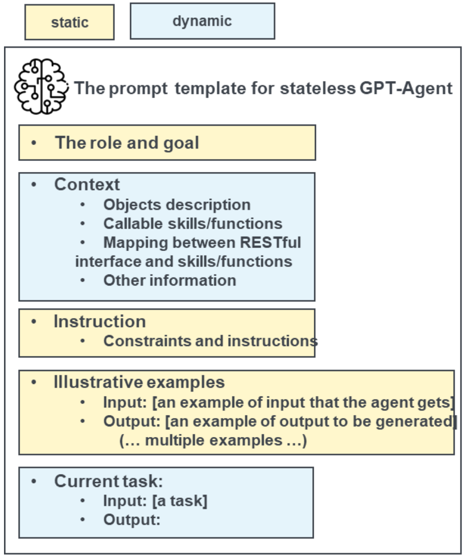
Prompting & interaction with GPT





The Prompt:

```
You are now managing a production system, and your goal is to design an efficient production process based on a given task, taking into account the provided context,exception, instructions, and examples.
Following these, you generate an output.
Context:
A production process consists of one or more process steps.
There are two type of process steps, one type is transportation process step, another type is production process step.
If the next production process is executed in a different production module, transportation process between two production processes is necessary.
The transportation step can be executed with a transport robot.
Transportation step is not considered as production process step.
This production system that you manage consists of several production modules. Each of these production modules has one or more skills to execute a production process step.
Each process can be executed with a skill of a module.
Here is a list of the production modules:
Inspection Module. It has the following skills: (I1) examine the raw material, (I2) examine the faulty material
Painting Module. It has the following skills: (P1) print a pattern on the surface , (P2)coat the material.
Here is a list of transportation modules:
Transport Robot called “Robotino”. It has the following skills: (T1) transport workpiece, (T2) enters the production area.
Instructions:
As a manager of this production system, please arrange a production process based on the input.
Only use the skills that are given in the context section.
Only use the skills that are necessary to carry out the task.
If you are confident about your output, give an explanation with short reason in list form.
You should think step by step.
Follow the text structure and syntax in the examples.
Example:
Input:{the customer wants to paint a hex pattern on the material after examinating the material quality. The workpiece is now in the Inspection Station.}
Output:{
(T2) transport robot first enters the production area and docks the inspection station.
(I1) Examine the material quality to make sure it's suitable for painting and pass it to transport robot.
(T1) transport workpiece from inspection station to painting station.
(P1) print a hex pattern on the surface of the material.}
Input: {the customer wants the steel sheet to be inspected. currently, the material is on the transport robot.}
Output:{
(T2) transport robot first enters the production area and transfers the material to inspection station.
(I1) Examine the material quality to make sure it's suitable for the next process.}
```



static

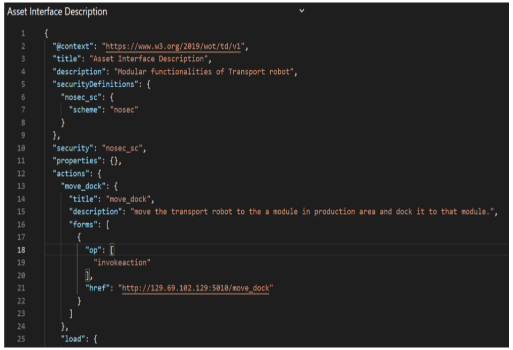
dynamic

The prompt template for stateless GPT-Agent

- The role and goal
- Context
  - Objects description
  - Callable skills/functions
  - Mapping between RESTful interface and skills/functions
  - Other information
- Instruction
  - Constraints and instructions
- Illustrative examples
  - Input: [an example of input that the agent gets]
  - Output: [an example of output to be generated]
    (… multiple examples …)
- Current task:
  - Input: [a task]
  - Output:

Information integration into the prompt

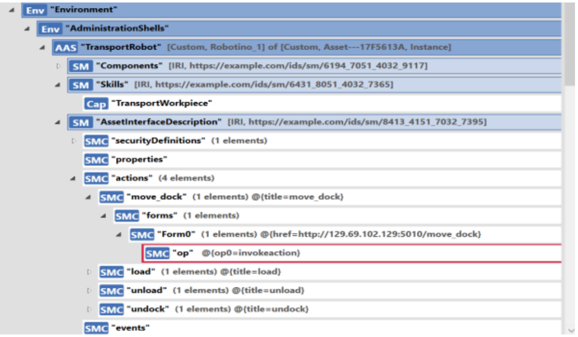
Asset interface Description in JSON-LD



Asset Interface Description

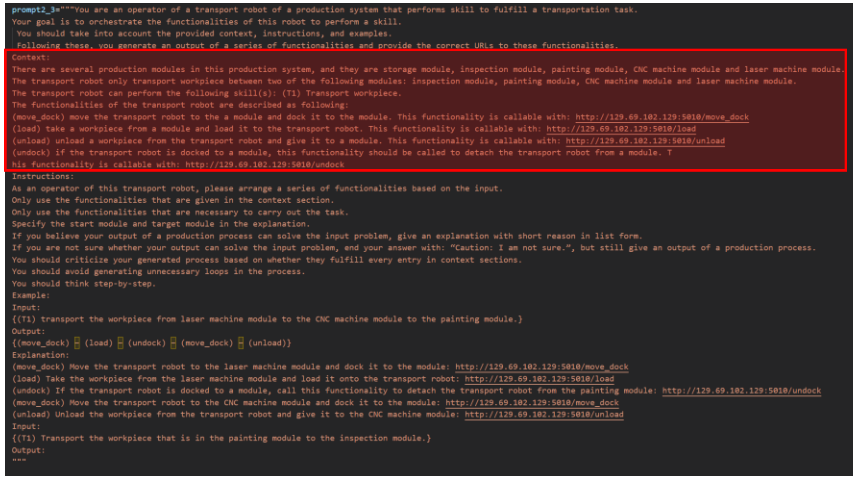
```
{
  "@context": "https://www.w3.org/2019/wot/td/v1",
  "title": "Asset Interface Description",
  "description": "Modular functionalities of Transport robot",
  "securityDefinitions": {
    "nosec_sc": {
      "scheme": "nosec"
    }
  },
  "security": "nosec_sc",
  "properties": {},
  "actions": {
    "move_dock": {
      "title": "move_dock",
      "description": "move the transport robot to the a module in production area and dock it to that module.",
      "forms": [
        {
          "op": [
            "invokeaction"
          ],
          "href": "http://129.69.102.129:5010/move_dock"
        }
      ]
    },
    "load": {
```

AAS Submodel



```
Env "Environment"
  Env "AdministrationShells"
    AAS "TransportRobot" [Custom, Robotino_1] of [Custom, Asset---17F5613A, Instance]
      SM "Components" [IRI, https://example.com/ids/sm/6194_7051_4032_9117]
      SM "Skills" [IRI, https://example.com/ids/sm/6431_8051_4032_7365]
        Cap "TransportWorkpiece"
      SM "AssetInterfaceDescription" [IRI, https://example.com/ids/sm/8413_4151_7032_7395]
        SMC "securityDefinitions" (1 elements)
        SMC "properties"
        SMC "actions" (4 elements)
          SMC "move_dock" (1 elements) @{title=move_dock}
            SMC "forms" (1 elements)
              SMC "Form0" (1 elements) @{href=http://129.69.102.129:5010/move_dock}
                SMC "op" @{op0=invokeaction}
          SMC "load" (1 elements) @{title=load}
          SMC "unload" (1 elements) @{title=unload}
          SMC "undock" (1 elements) @{title=undock}
        SMC "events"
```

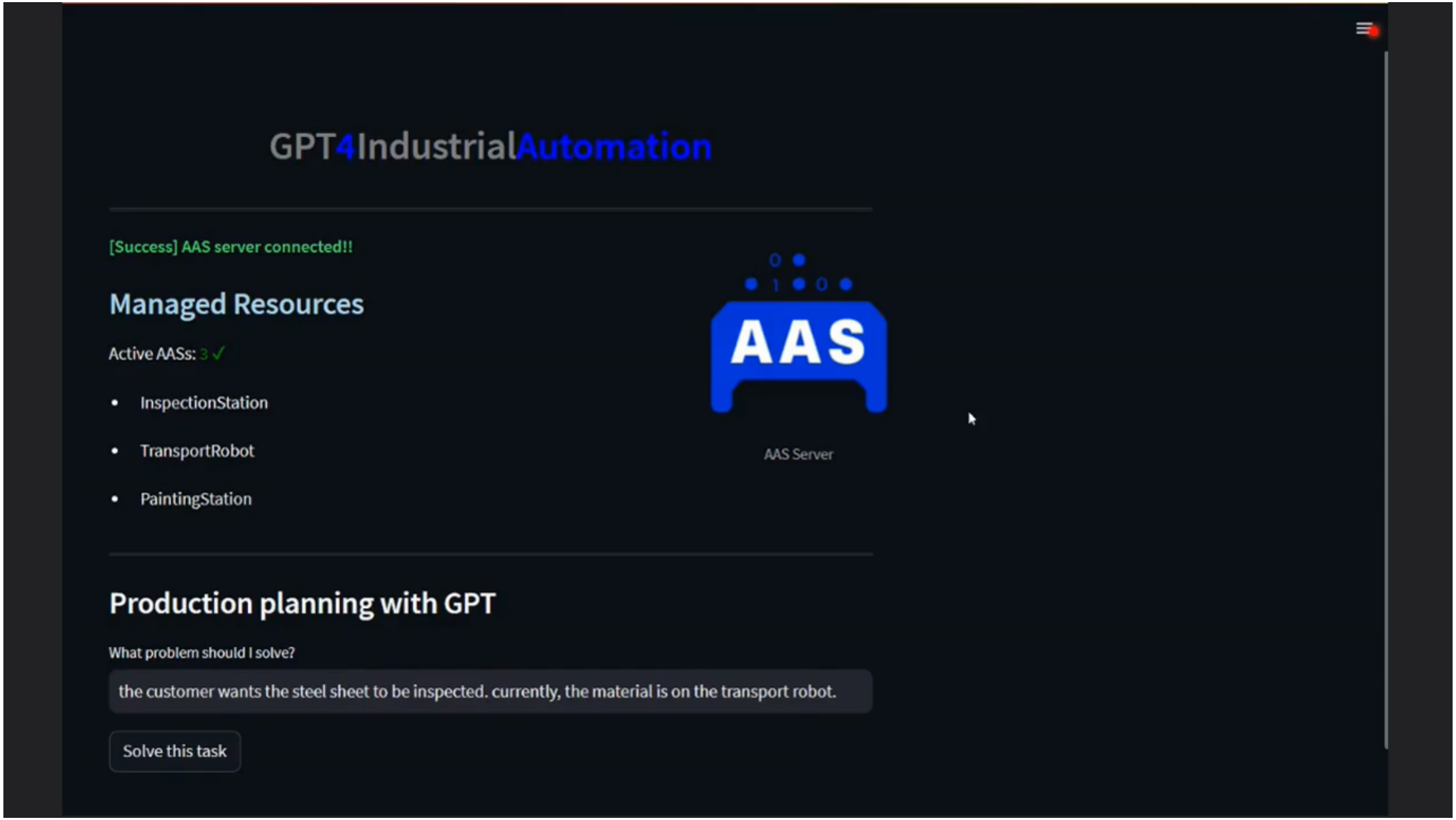
GPT-Prompt



```
prompt2_3="""You are an operator of a transport robot of a production system that performs skill to fulfill a transportation task.
Your goal is to orchestrate the functionalities of this robot to perform a skill.
 You should take into account the provided context, instructions, and examples.
 Following these, you generate an output of a series of functionalities and provide the correct URLs to these functionalities.
Context:
There are several production modules in this production system, and they are storage module, inspection module, painting module, CNC machine module and laser machine module.
The transport robot only transport workpiece between two of the following modules: inspection module, painting module, CNC machine module and laser machine module.
The transport robot can perform the following skill(s): (T1) Transport workpiece.
The functionalities of the transport robot are described as following:
(move_dock) move the transport robot to the a module and dock it to the module. This functionality is callable with: http://129.69.102.129:5010/move_dock
(load) take a workpiece from a module and load it to the transport robot. This functionality is callable with: http://129.69.102.129:5010/load
(unload) unload a workpiece from the transport robot and give it to a module. This functionality is callable with: http://129.69.102.129:5010/unload
(undock) if the transport robot is docked to a module, this functionality should be called to detach the transport robot from a module. T
his functionality is callable with: http://129.69.102.129:5010/undock
Instructions:
As an operator of this transport robot, please arrange a series of functionalities based on the input.
Only use the functionalities that are given in the context section.
Only use the functionalities that are necessary to carry out the task.
Specify the start module and target module in the explanation.
If you believe your output of a production process can solve the input problem, give an explanation with short reason in list form.
If you are not sure whether your output can solve the input problem, end your answer with: “Caution: I am not sure.”, but still give an output of a production process.
You should criticize your generated process based on whether they fulfill every entry in context sections.
You should avoid generating unnecessary loops in the process.
You should think step-by-step.
Example:
Input:
{(T1) transport the workpiece from laser machine module to the CNC machine module to the painting module.}
Output:
{(move_dock) □ (load) □ (undock) □ (move_dock) □ (unload)}
Explanation:
(move_dock) Move the transport robot to the laser machine module and dock it to the module: http://129.69.102.129:5010/move_dock
(load) Take the workpiece from the laser machine module and load it onto the transport robot: http://129.69.102.129:5010/load
(undock) If the transport robot is docked to a module, call this functionality to detach the transport robot from the painting module: http://129.69.102.129:5010/undock
(move_dock) Move the transport robot to the CNC machine module and dock it to the module: http://129.69.102.129:5010/move_dock
(unload) Unload the workpiece from the transport robot and give it to the CNC machine module: http://129.69.102.129:5010/unload
Input:
{(T1) Transport the workpiece that is in the painting module to the inspection module.}
Output:
"""
```

Result

GPT4Automation



GPT4IndustrialAutomation

[Success] AAS server connected!!

Managed Resources

Active AASs: 3 ✓

- InspectionStation
- TransportRobot
- PaintingStation

Production planning with GPT

What problem should I solve?

the customer wants the steel sheet to be inspected. currently, the material is on the transport robot.

Conclusion and Outlook

Summary:

- Successfully use GPT to generate workplan for production service
- Address the flexible manufacturing problem in autonomous system

Outlook:

- More evaluation on **GPT** and the **System Reliability**
- Add more automation components and more complicated skills
- More complicated coordination of the operations

University of Stuttgart
Institut of Industrial Automation
and Software Engineering

Thank you!

Manthan Venkatarmana Shneoy

e-mail st175289@stud.uni-stuttgart.de
phone +49 (0) 711 685-
fax +49 (0) 711 685-

University of Stuttgart
Pfaffenwaldring 47, 70550

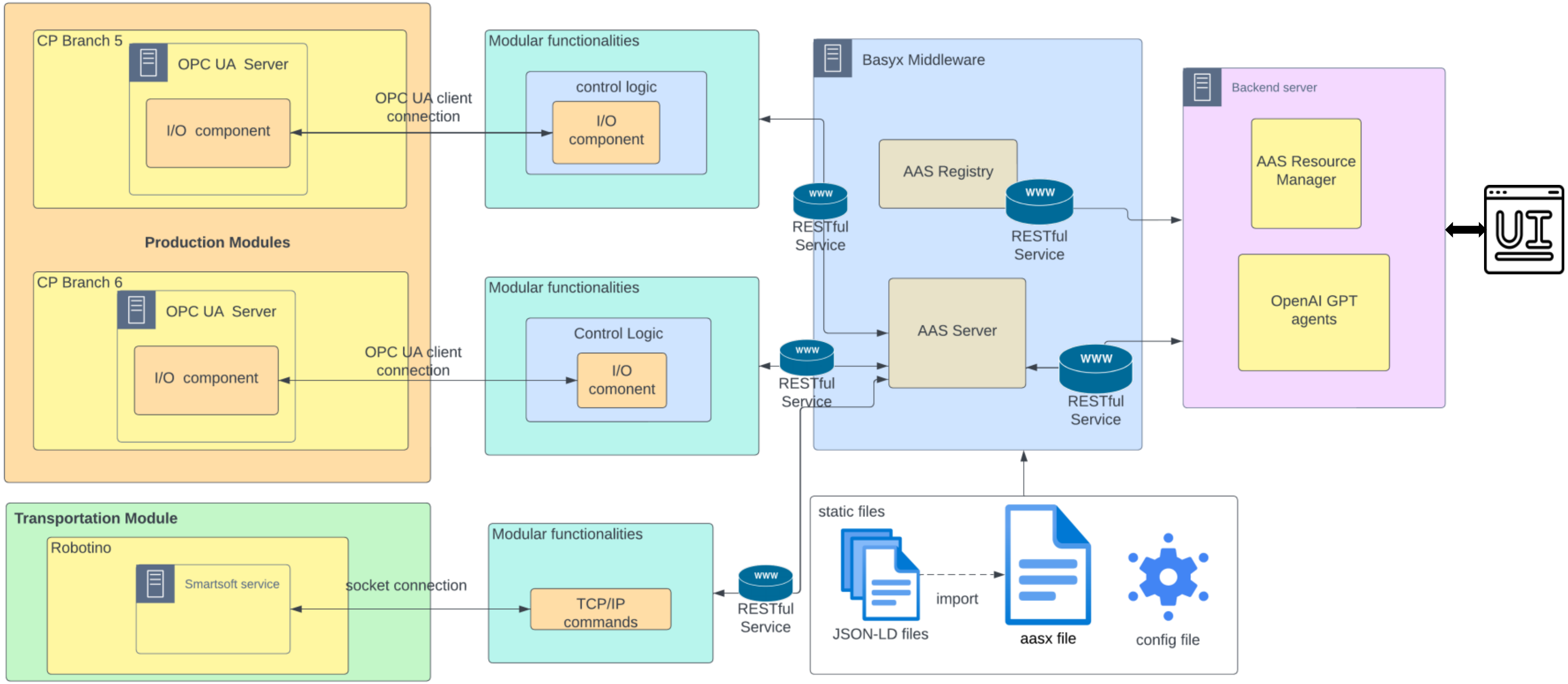
References[1/2]

- **[1] Köcher, A., Hildebrandt, C., Caesar, B., Bakakeu, J., Peschke, J., Scholz, A., & Fay, A. (2020). Automating the Development of Machine Skills and their Semantic Description. *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*, *2020-September*, 1013–1018. https://doi.org/10.1109/ETFA46521.2020.9211933**
- [2] submodel-templates/development at main · admin-shell-io/submodel-templates (github.com)
- **[3] Pakala, H. K., Oladipupo, K. O., Käbisch, S., & Diedrich, C. (n.d.). *Integration of Asset Administration Shell and Web of Things*. https://doi.org/10.25673/39570**
- **[4] *Interrelation of Asset Administration Shell and AutomationML Position paper State: May 2021 Interrelation of AAS and AutomationML 2*. (n.d.).**
- [5] K. Dorofeev and M. Wenger, "Evaluating Skill-Based Control Architecture for Flexible Automation Systems," 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Zaragoza, Spain, 2019, pp. 1077-1084, doi: 10.1109/ETFA.2019.8869050.
- [6] Köcher, A., Miguel Vieira da Silva, L., & Fay, A. (n.d.). *Modeling and Executing Production Processes with Capabilities and Skills using Ontologies and BPMN*. Retrieved June 30, 2023, from https://github.com/bpmn-io/bpmn-js
- [7] Köcher, A., Belyaev, A., Hermann, J., Bock, J., Meixner, K., Volkmann, M., Winter, M., Zimmermann, P., Grimm, S., & Diedrich, C. (2023). A reference model for common understanding of capabilities and skills in manufacturing. *At-Automatisierungstechnik*, *71*(2), 94–104. https://doi.org/10.1515/AUTO-2022-0117/MACHINEREADABLECITATION/RIS
- [8] Kocher, A., Jeleniewski, T., & Fay, A. (2021). A Method to Automatically Generate Semantic Skill Models from PLC Code. *IECON Proceedings (Industrial Electronics Conference)*, *2021-October*. https://doi.org/10.1109/IECON48115.2021.9589674

References [2/2]

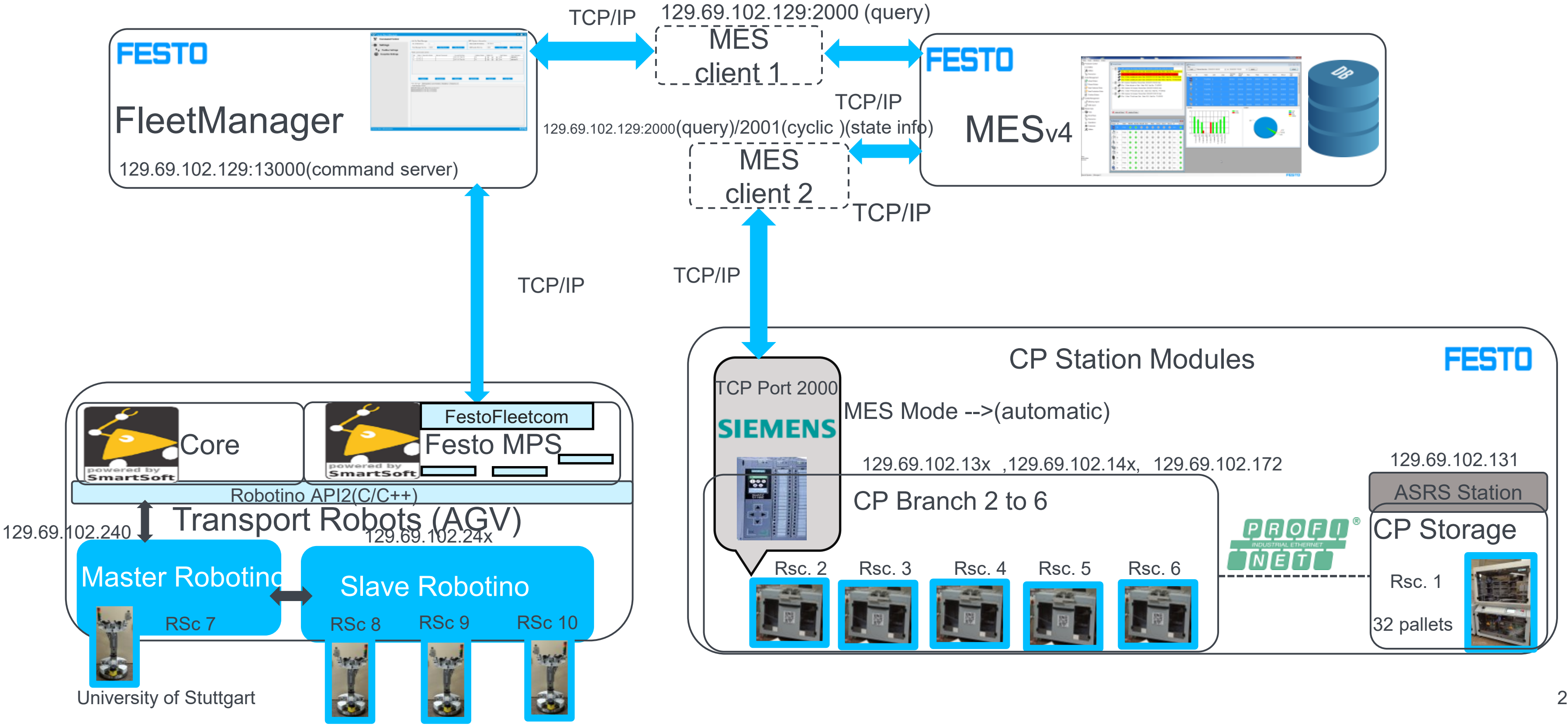
- **[9] Göhner, P., & Weyrich, M. (n.d.). *Agent-based Concepts for Manufacturing Automation*. Retrieved July 1, 2023, from http://www.ias.uni-stuttgart.de**
- **[10] Sakurada, L., Leitao, P., & la Prieta, F. de. (2022). Agent-Based Asset Administration Shell Approach for Digitizing Industrial Assets. *IFAC-PapersOnLine*, *55*(2), 193–198. https://doi.org/10.1016/J.IFACOL.2022.04.192**
- **[11] Klein, M., Löcklin, A., Jazdi, N., & Weyrich, M. (2018). A negotiation based approach for agent based production scheduling. *Procedia Manufacturing*, *17*, 334–341. https://doi.org/10.1016/J.PROMFG.2018.10.054**
- **[12] Xia, Y., Shenoy, M., Jazdi, N., & Weyrich, M. (2023). *Towards autonomous system: flexible modular production system enhanced with large language model agents*. https://arxiv.org/abs/2304.14721v3**
- [13] Belyaev, Alexander & Diedrich, Christian. (2019). Specification "Demonstrator I4.0-Language" v3.0.
- [14] Grunau, S., Redeker, M., Göllner, D., Wisniewski, L., Grunau, S., Wisniewski, L., Redeker, M., & Göllner, D. (2022). *The Implementation of Proactive Asset Administration Shells: Evaluation of Possibilities and Realization in an Order Driven Production*. 131–144. https://doi.org/10.1007/978-3-662-64283-2_10
- [15] BaSyx - Eclipsepedia

System Architecture



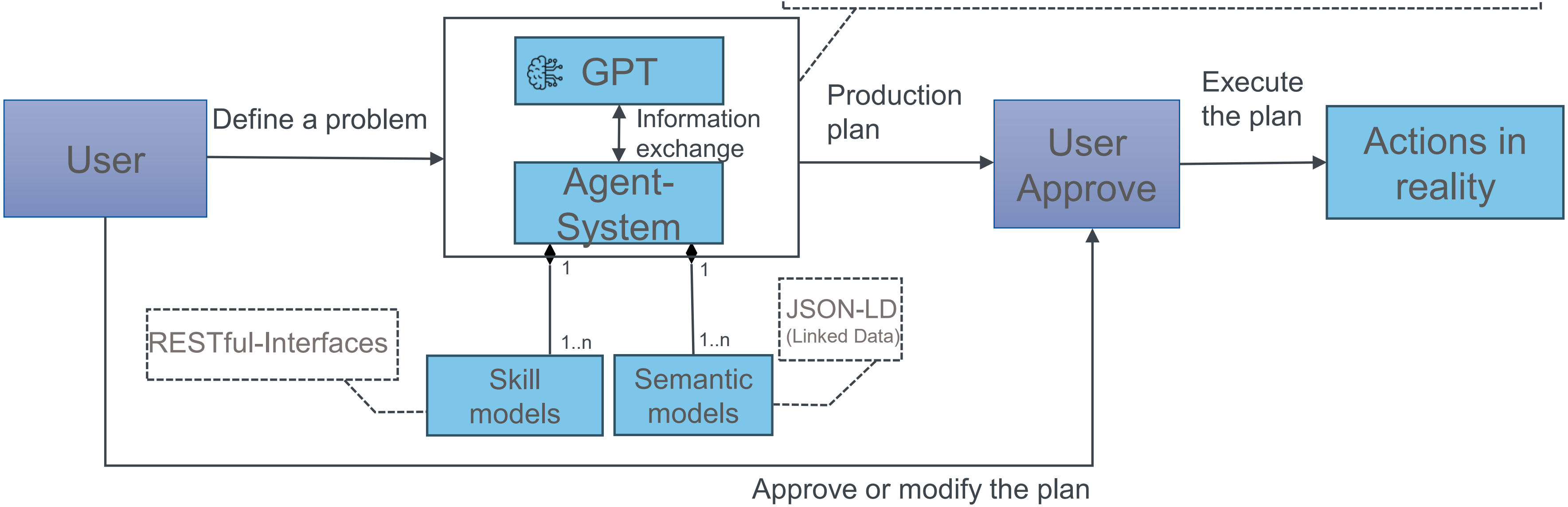
Current System Overview

Communication & information exchange point of view

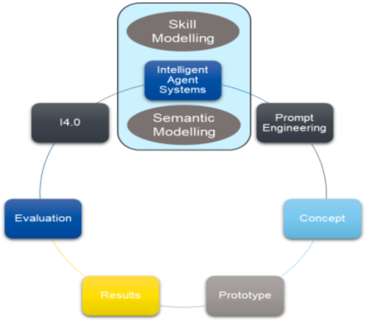


GPT4Automation control workflow

- Implemented with Baysx middleware
- OPC UA interface used for I/O in production module
- Socket connection for transport robot



Industry 4.0 (I4.0)

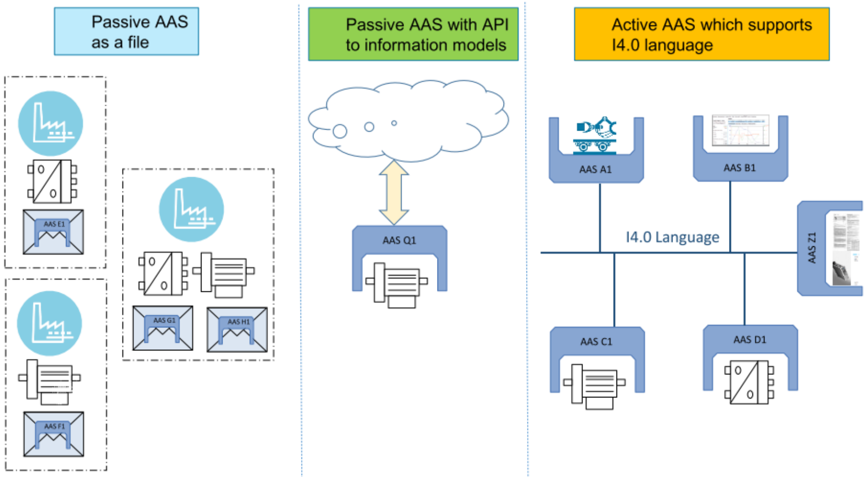


- Decentralised control and decision making
- Interoperability

- CPS to **modularized** I4.0 component using AAS and communicate by I4.0 standards

- I4.0 component & I4.0 language (proactive AAS)

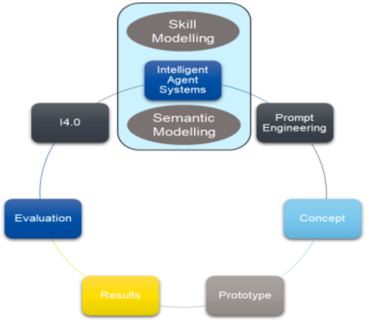
- AAS -DIN EN IEC 63278-1

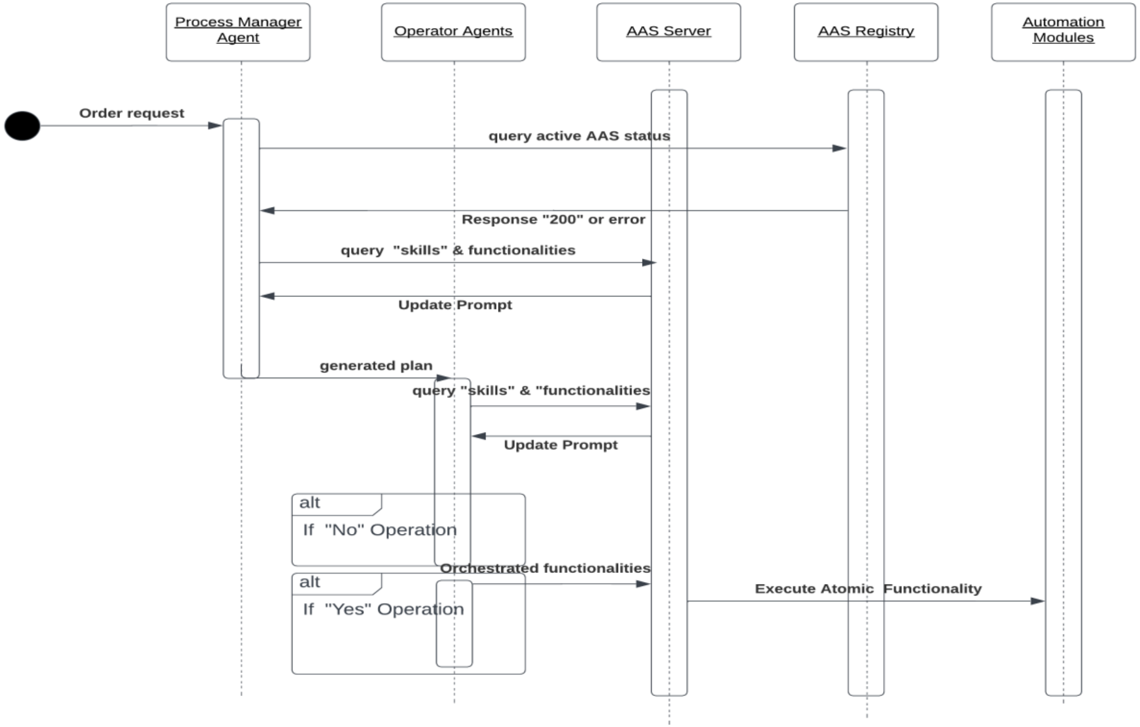


Source: Plattform Industrie4.0

Sequence Diagram

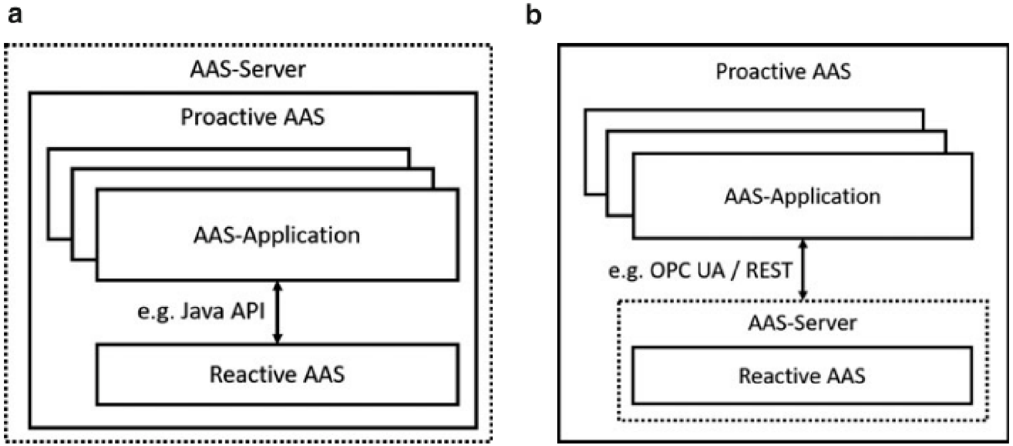
Communication PoV

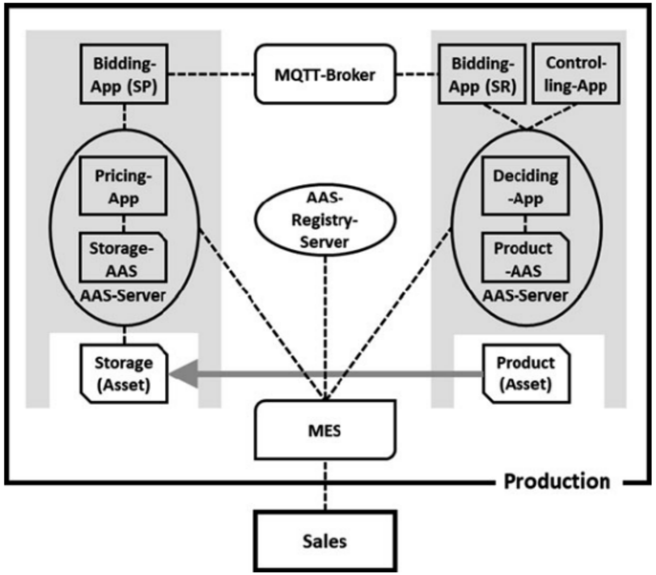




Industry 4.0

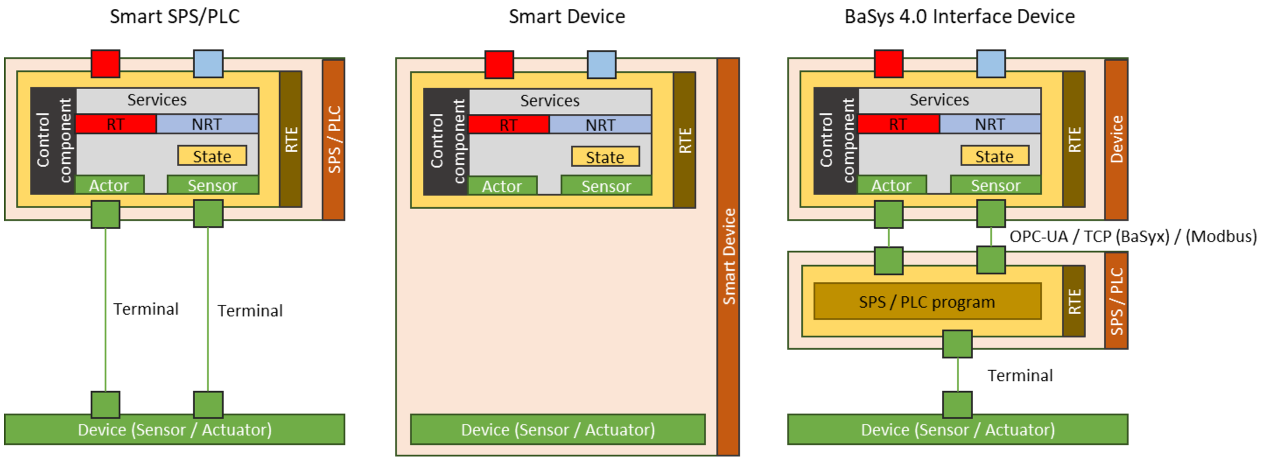
Proactive AAS





[14]

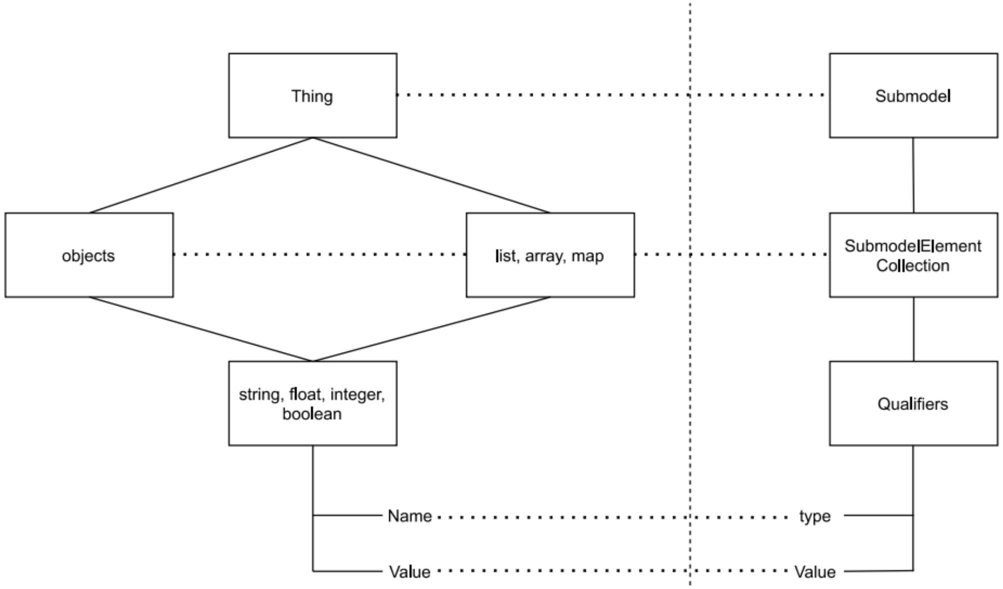
Skill Modelling



[BaSyx Device integration - Eclipsepedia]

Semantic Modelling

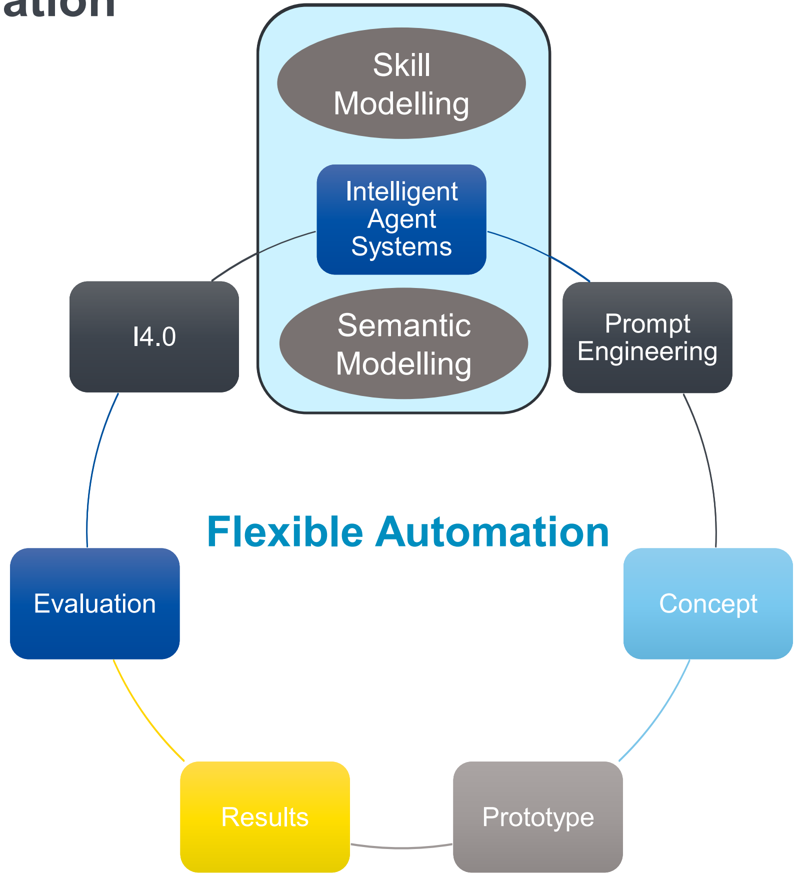
Asset Interface Description



[3]

Goal: Flexible automation

Agenda



Skill Modelling
Intelligent Agent Systems
Semantic Modelling
I4.0
Prompt Engineering
Flexible Automation
Evaluation
Concept
Results
Prototype