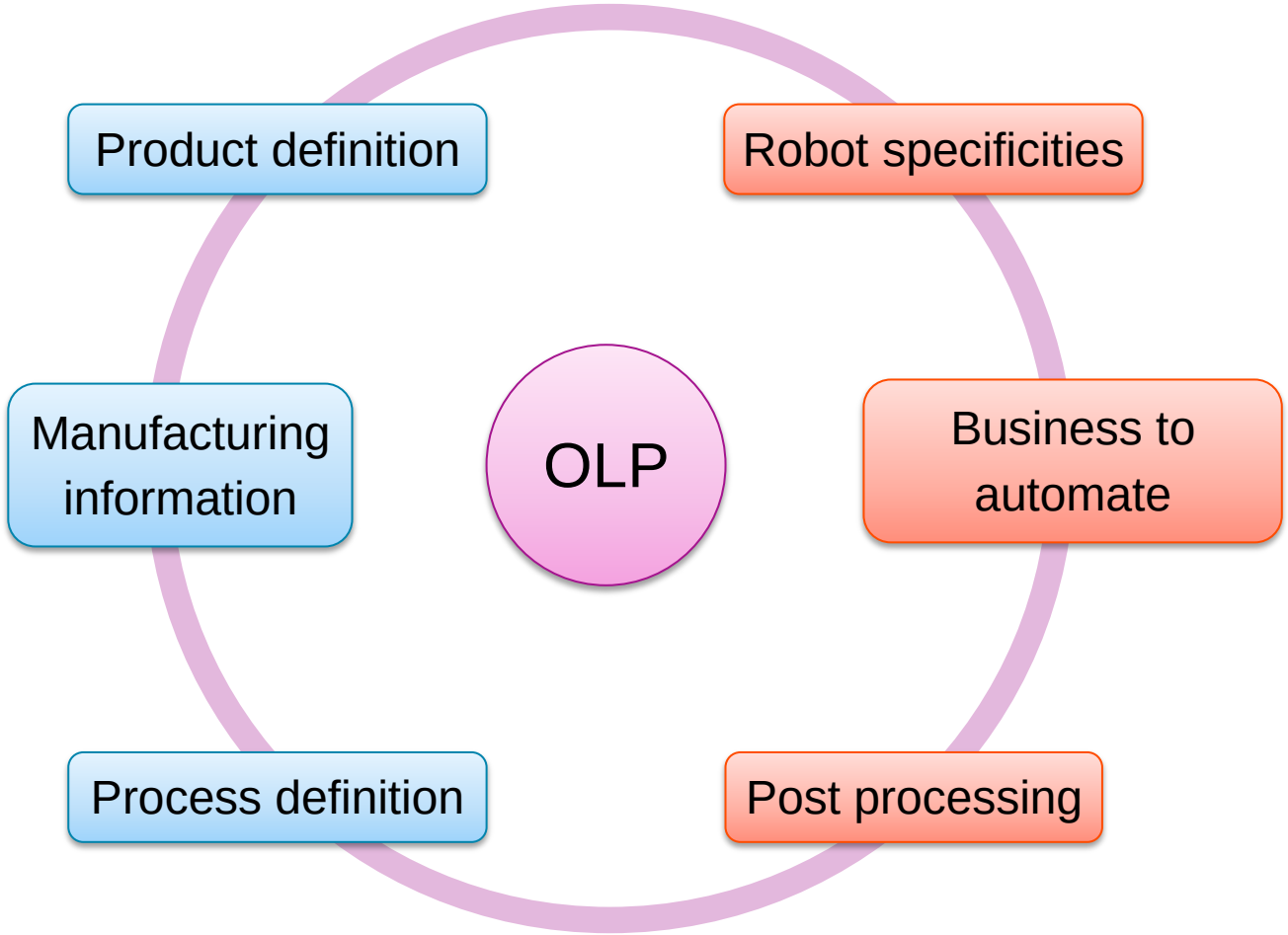
Business oriented robot Off-Line Programming solution using AutomationML

Perig Le-Henaff, Airbus SAS; Sylvain Blanvillain, Capgemini
Nikolai D'Agostino, CENIT AG
Daniele Massa, Loccioni
24th October 2018

AIRBUS

Introduction

- Off-Line Programming (OLP) of automated processes in Aerospace
- Off-Line Programming tools are specific to:
  - Robot types
  - Application to automate
- OLP boost robot integration costs (time & money)
- Vision:
  - to be agnostic to
    - The robot type
    - The process specificities
  - to master the E2E data flow



Current situation in Off-Line Programming

Configuration

Product definition

Business to implement

Post Pro

Robot type

Many & different input types

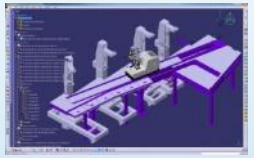
Specific programming solutions

Specific commissioning & execution tools

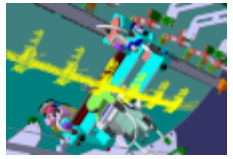
Programming solution

Programming solution

Programming solution



Commissioning Execution

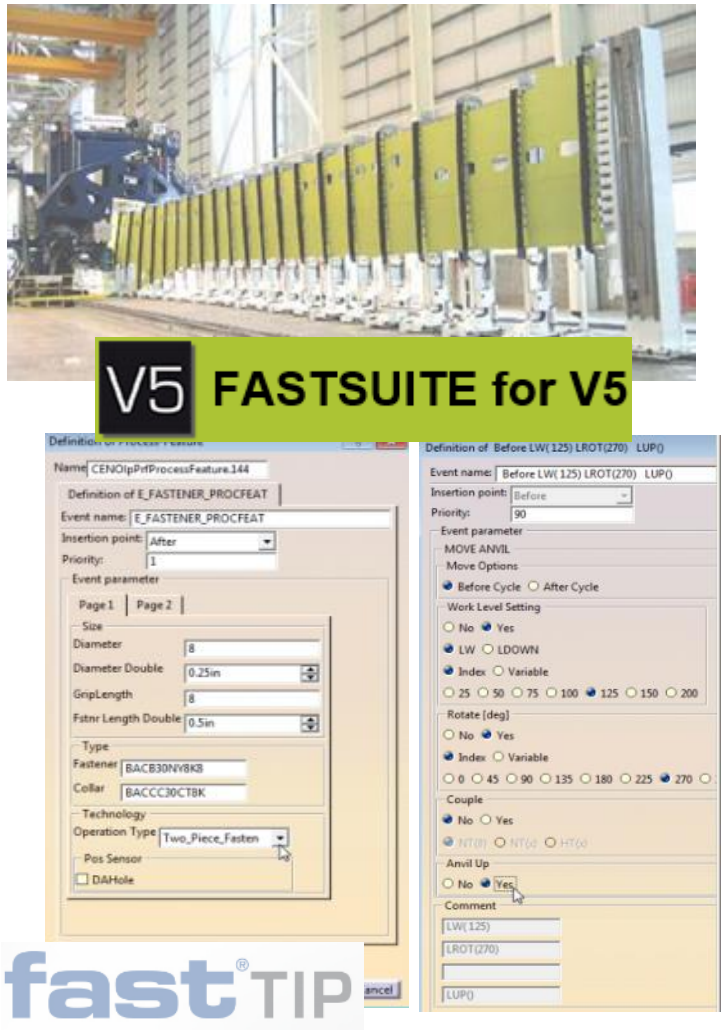


Commissioning Execution

Commissioning Execution

Automation System Specific Offline-Programming

- **Currently there are just a few highly specialised and complex automation systems for automated airframe assemby**
  - Example: wing assembly with Low Voltage Electric Riveter (LVER) for A380 wing assembly
    - About 40 controlled NC-axes
    - Many cycles with dozens of parameters each
    - up to 10.000 rivets per panel at A380
  - **OLP with generic OLP-Tools not economic (Just virtual teach-in)**
    - Tedious single position selection
    - Manual cycle definition is error prone
    - No detailed collision check
  - **Current solution CENIT FASTIP with machine specific customisation**
    - Process-oriented programming approach
    - dedicated UI for each assembly system (tool and cycle selection)
    - dedicated automated programming strategies
    - dedicated consistency checks
    - dedicated collision check

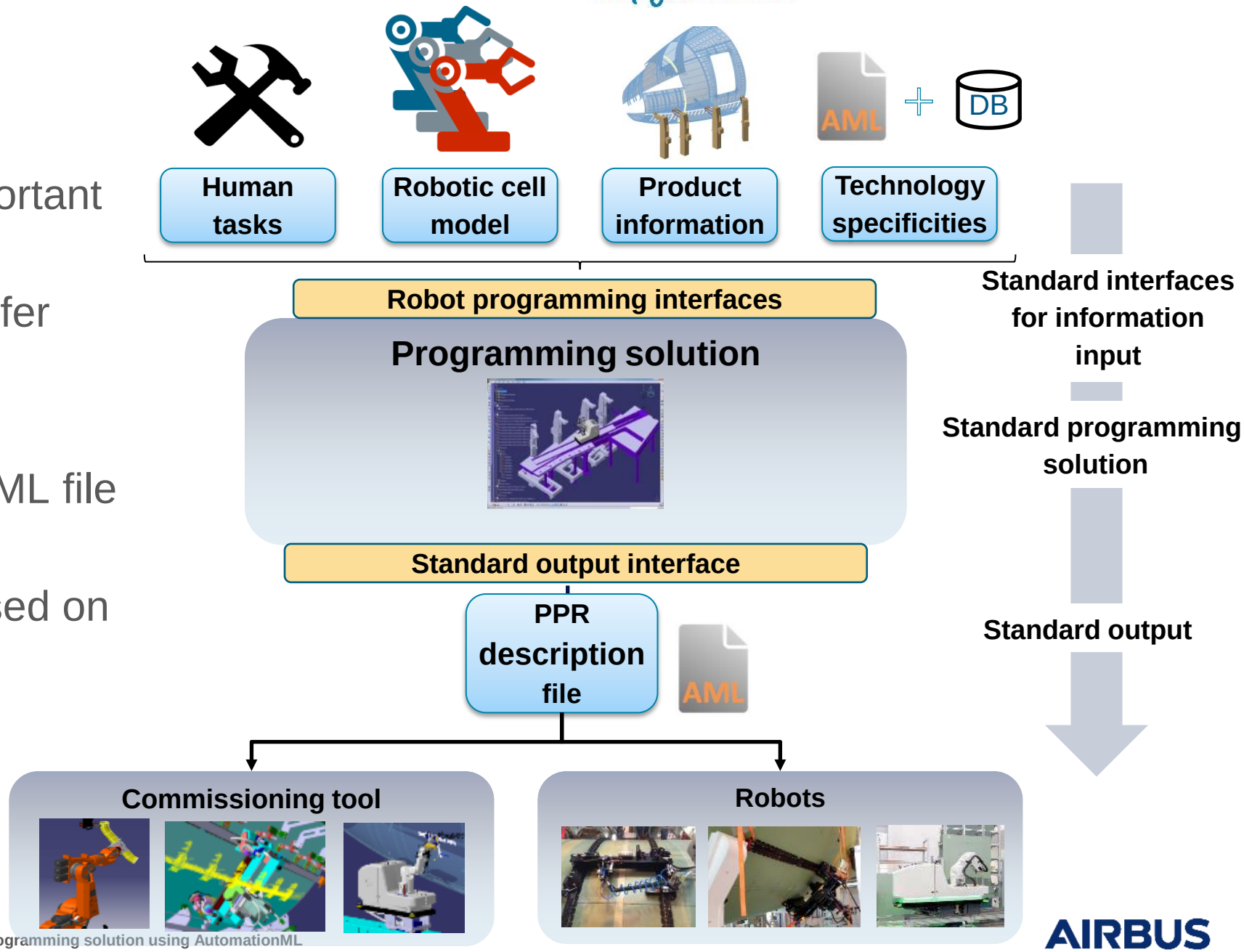


New trend in automation: Lightweight/collaborative robots

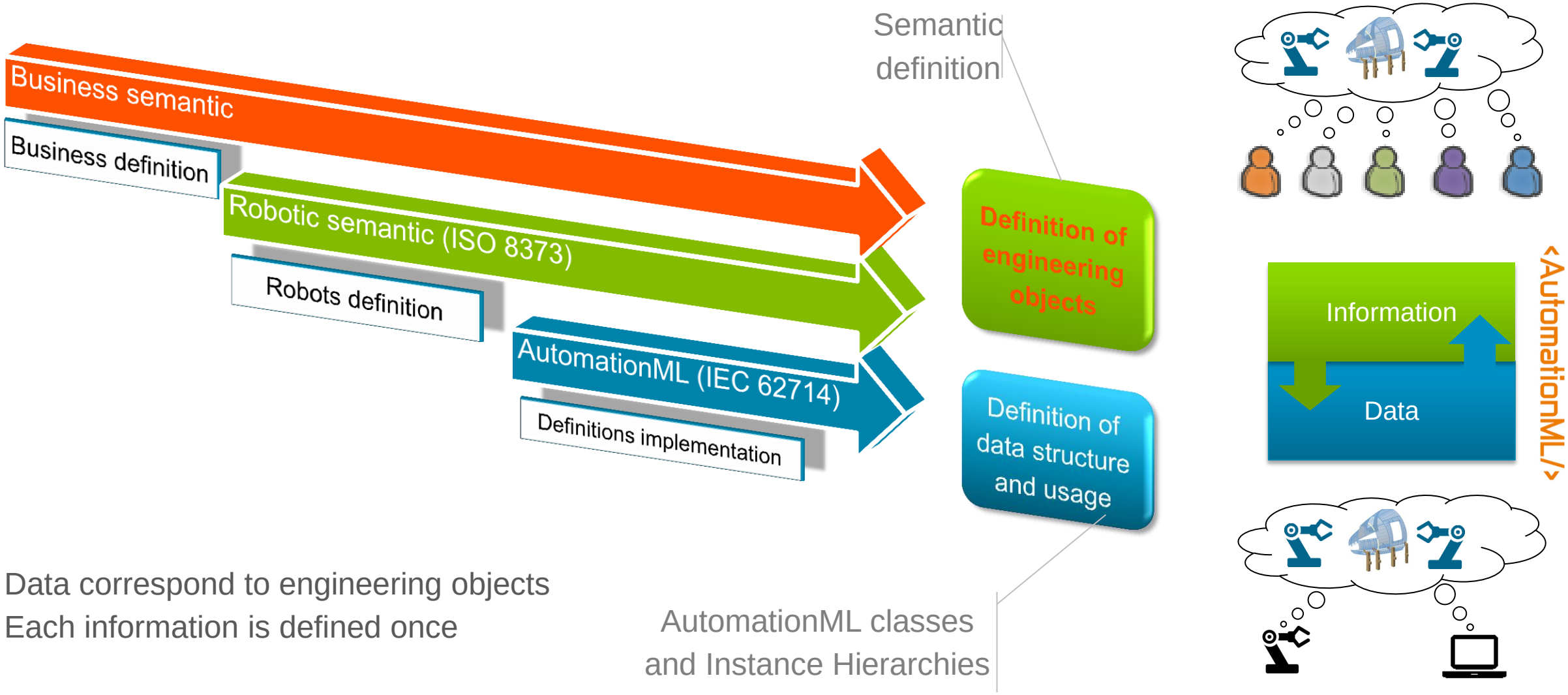
- **Increasing number of systems with less complexity**
  - Individual customisation of OLP for each system becomes un-economic
  - Hence, process specific and device agnostic approach is necessary
  - However, customised setup for each individual system shall be kept
- **Solution:**
  - Automated customisation based on machine readable business process description
  - **Business process description for automation system necessary**
  - **Usage of PPR Model approach:**
    - **Product**
      - Product requirements to be fulfilled by manufacturing process
        - e.g. assembly parts by fastening
    - **Process**
      - Mapping of manufacturing process (install fastener) to product requirements (assembly join)
    - **Resource**
      - Robot, End effector, working cycles, etc.
      - Humans – robots interaction

Requirements

- Software interfaces are important
- File based information transfer
- Technology specificities are described in an AutomationML file
- Programming solution focused on business objects
- Standard output using AutomationML



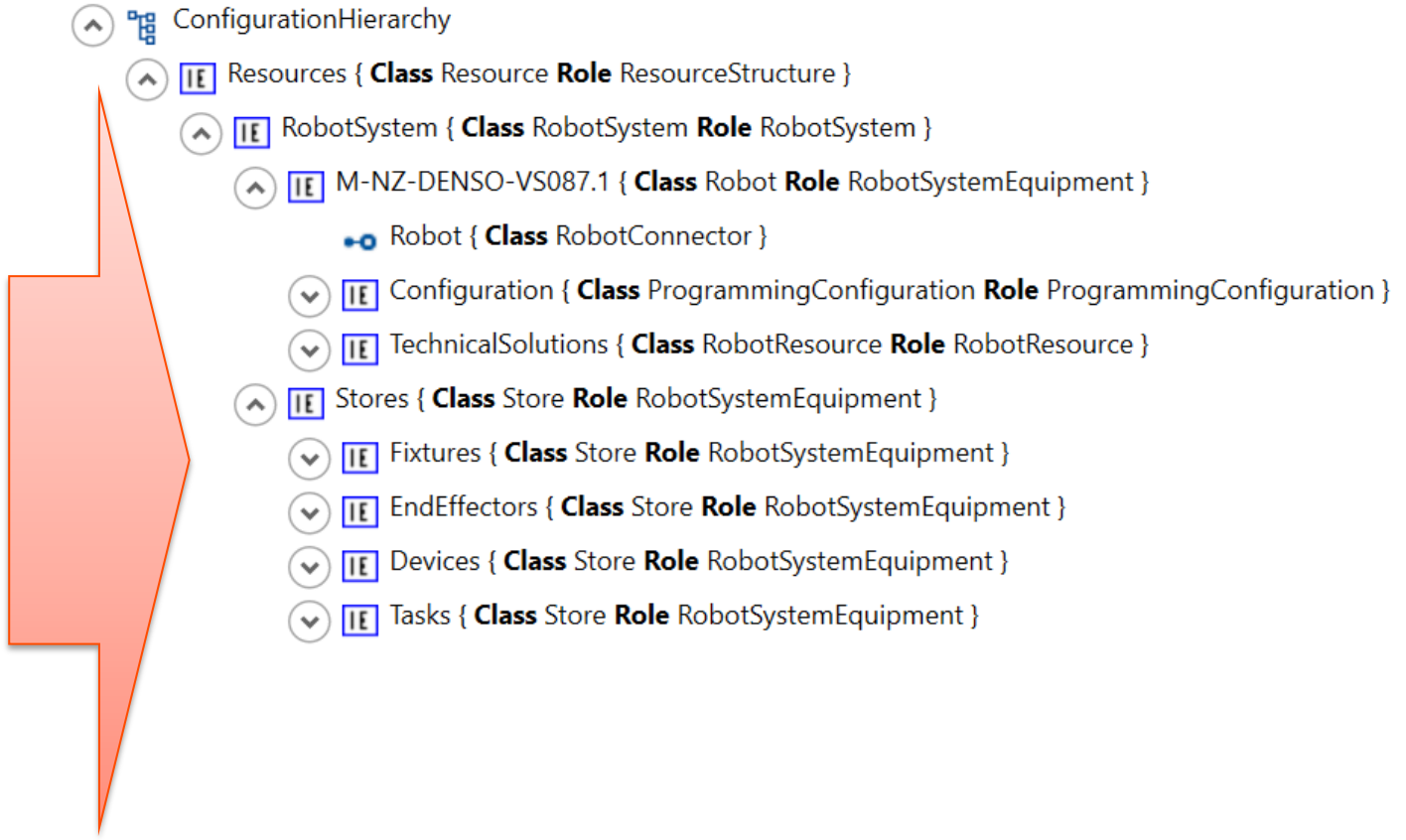
Semantic description of automated processes



- Data correspond to engineering objects
- Each information is defined once

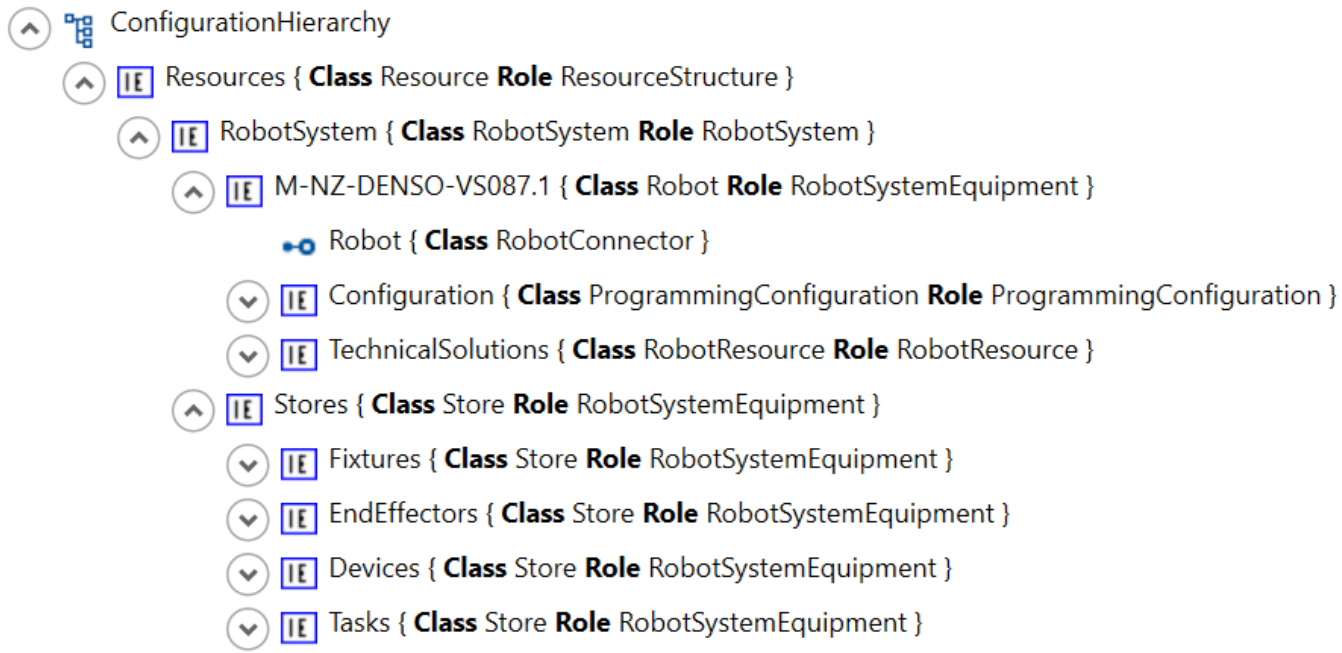
Automated customisation of CENIT OLP System with AutomationML

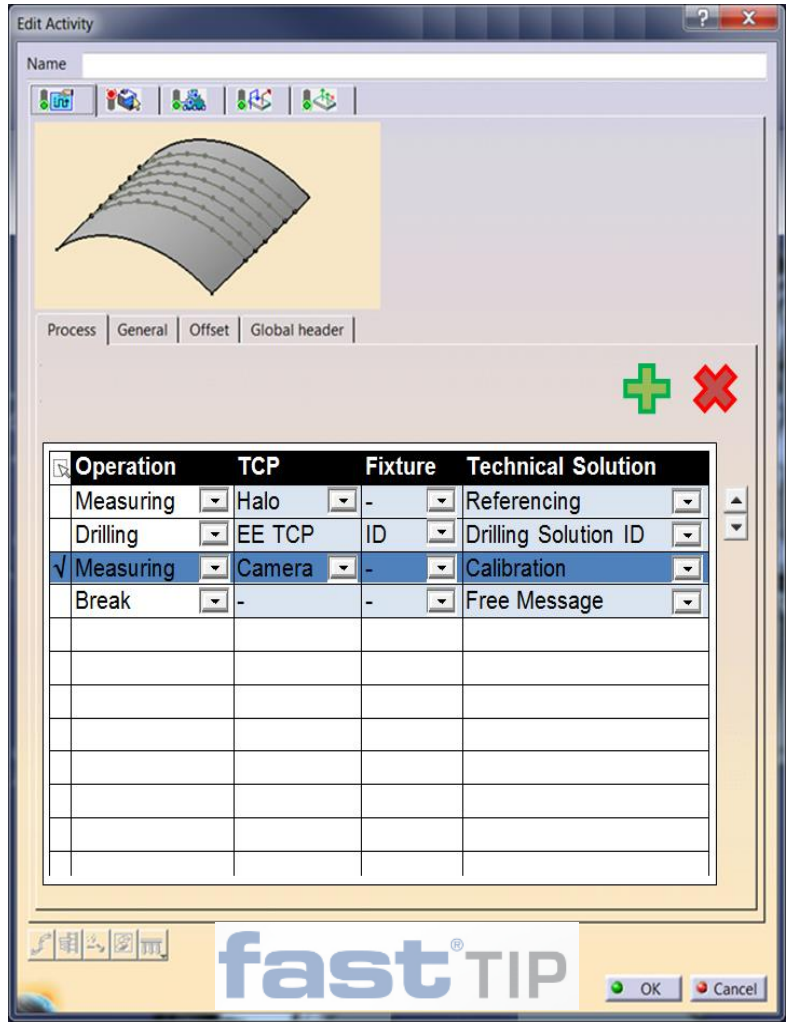
- Definition of device agnostic generic UI structure in OLP System CENIT FASTIP
- Initialise UI with resource and process description based on AutomationML
- Objects to be customised:
  - Manufacturing operations
    - e.g. drilling, fastener installation, measure, etc.
  - Tool-center points
  - End-effectors
  - Cycles (Technical solutions)
  - Auxiliary functions (break, operator message, etc.)
  - Dependencies of valid combinations
- Download of programming result with AutomationML as well



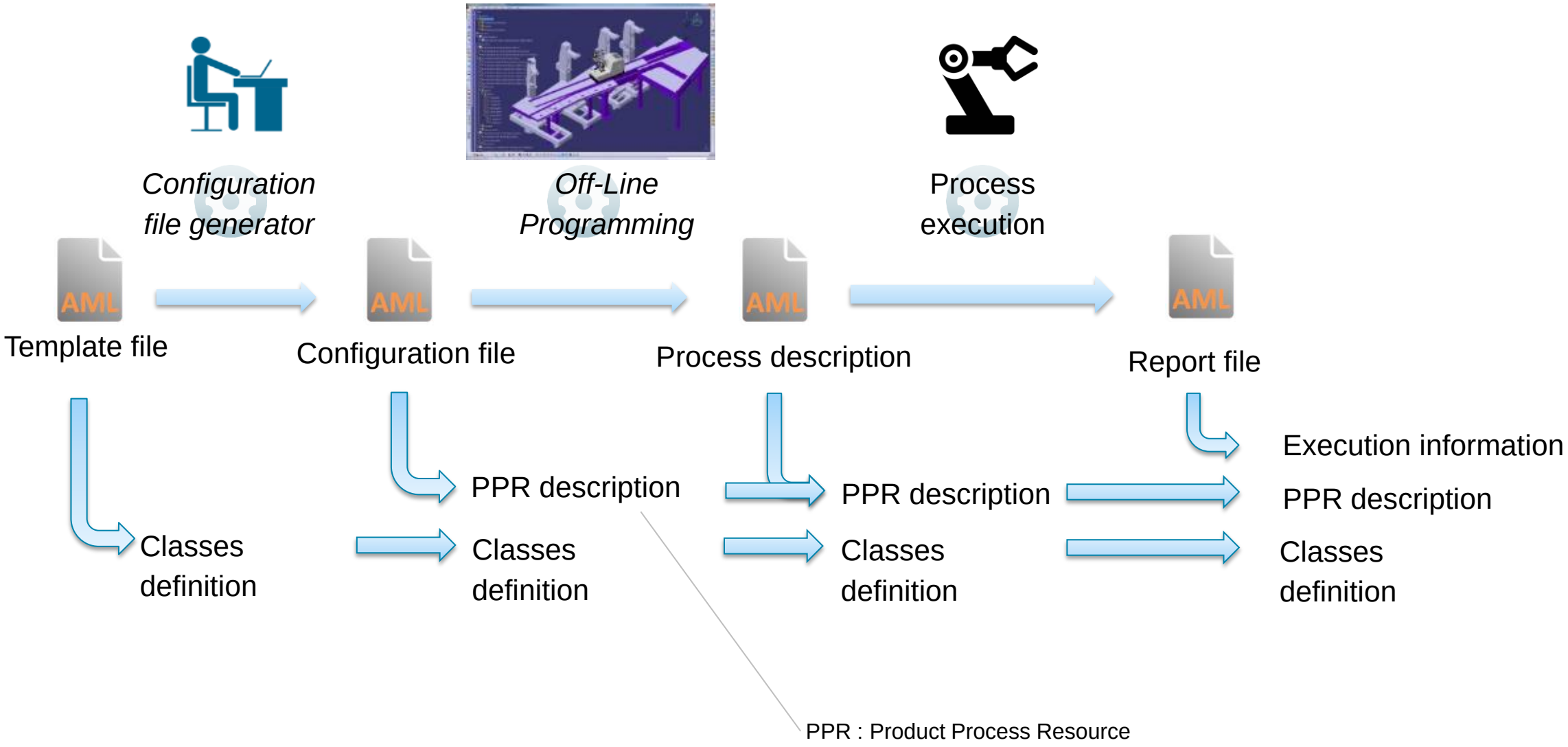
Automated customisation of CENIT OLP System with AutomationML

- Definition of device agnostic generic UI structure in OLP System CENIT FASTIP driven by AutomationML
- Process and machine specific information is directly accessible by the user without additional customisation
- Consistency rules covered by logic relations in AutomationML





Clear end-to-end information flow



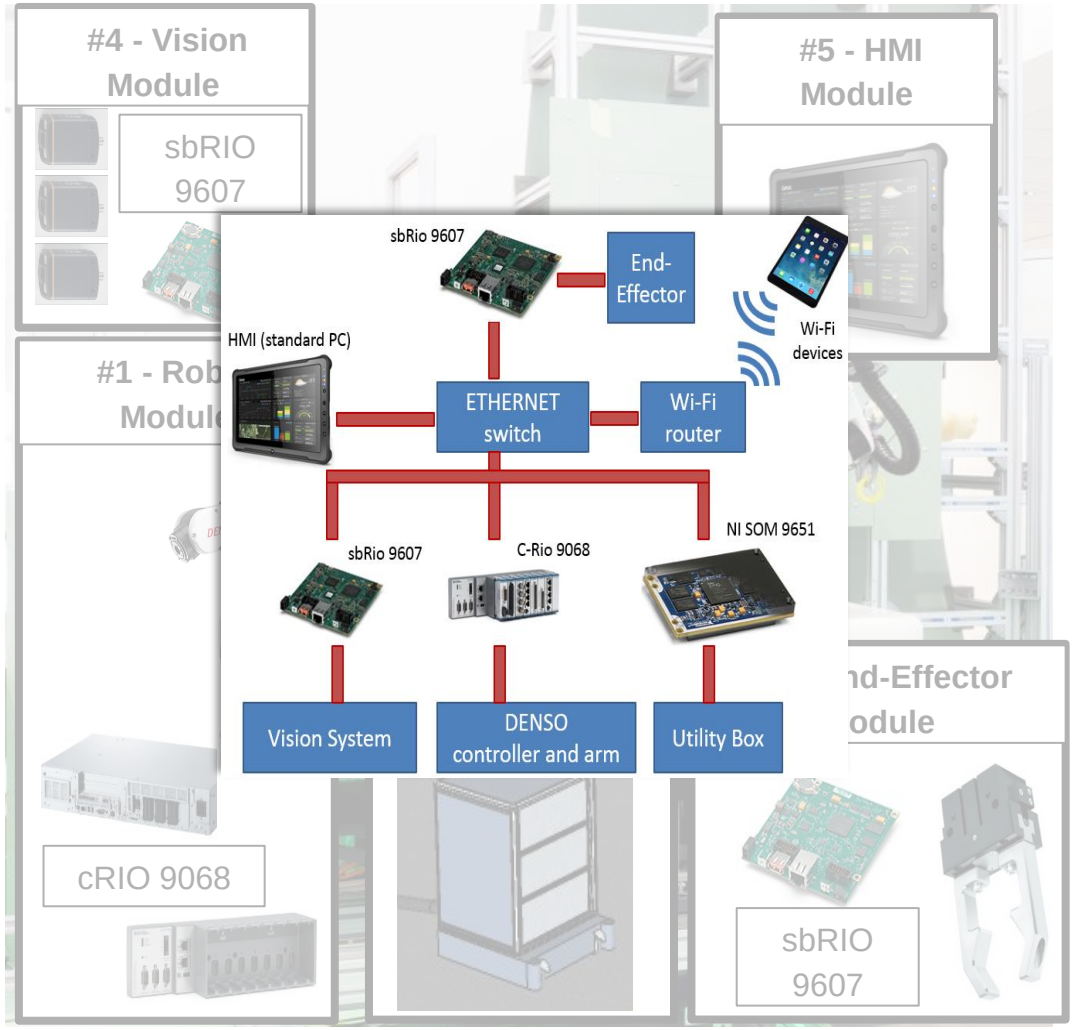
Use case: MEL-O (Modular Extensible Lightweight Robot - Outside)

Smart and cognitive robotic cell made of standard, “off-the-shelf”, re-usable technology blocks and components

Each hardware component of the cell can be considered an independent module with its own embedded intelligence:

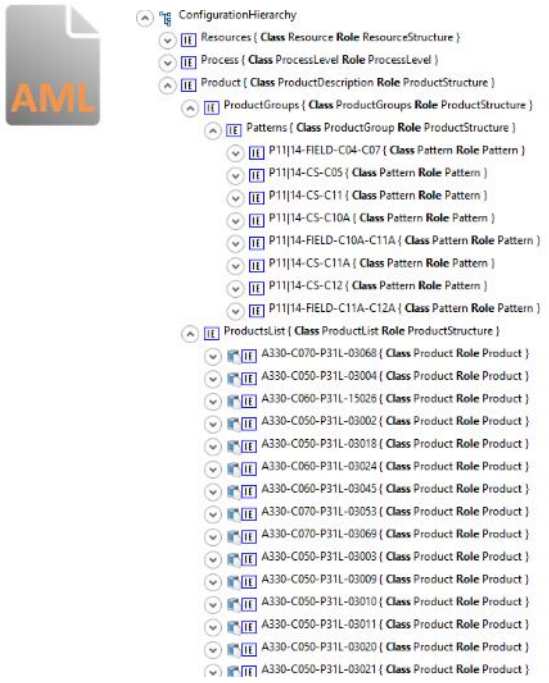
- Robotic arm
- End-Effector
- Vision System
- HMI
- Electrical Cabinet with digital I/Os and pneumatic valves

Hardware architecture is fully open and scalable



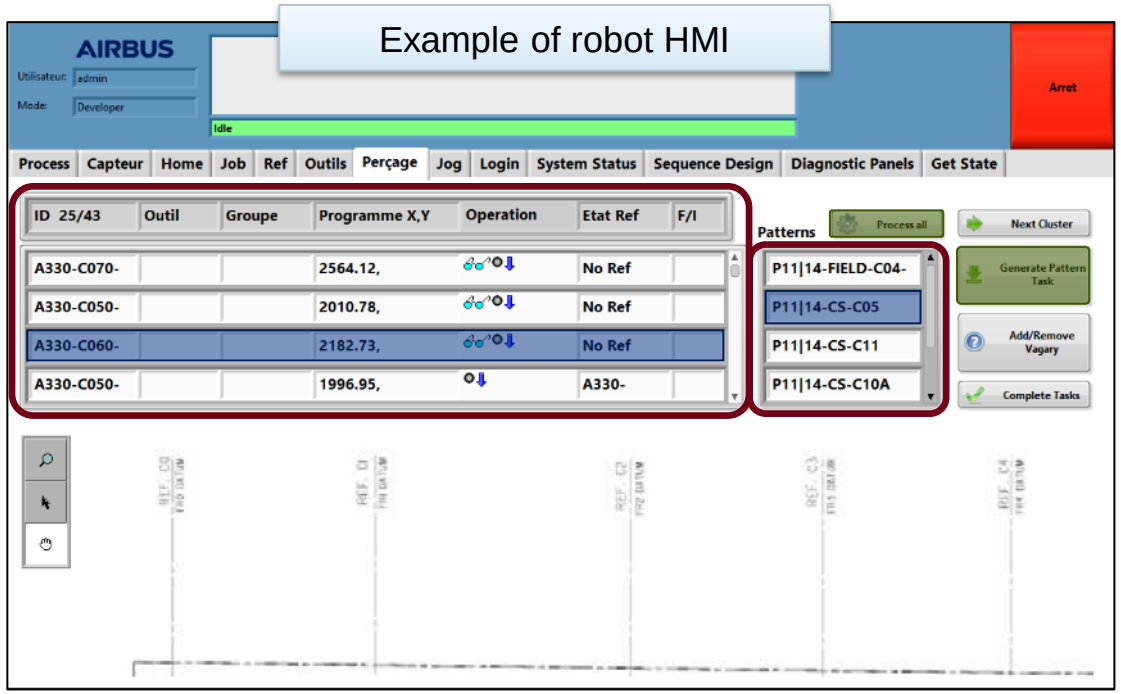
Retrieving process information from AML

- The AML file is loaded at run-time; **AML Engine** has been used to write a **wrapper** for the robotic cell.
- Starting from the **Process and Product description**, information about **groups of operations** and **product elements** are displayed on the **HMI**.
- The user can **manage process uncertainties** to exclude part of the task. These **information** is **written** back to **AML file**.
- The user can **select** the **sub-part of process to start**, or start the whole process.



AML Wrapper

Example of robot HMI



Parsing information to robot and process execution

Process information is converted to the robotic cell specific language, and a sequence of tasks for the different modules is generated, depending on the selected process part to be executed.

Process Information

AML

Process description

AML Wrapper

Robot Task List

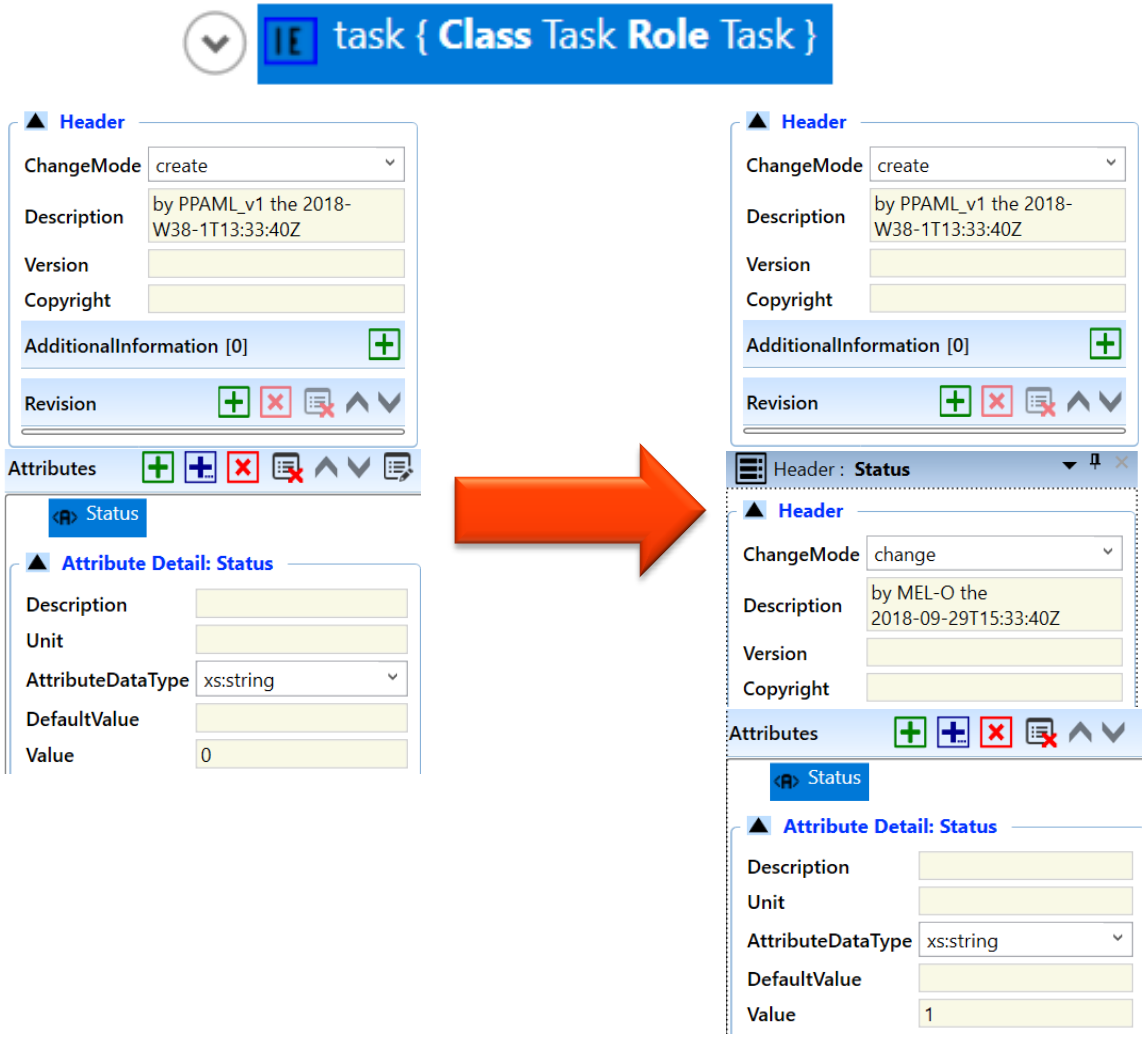
Module Name	Task Name	Task Parameters
Robot Arm	Move	Position, speed, …
Vision System	Measure	Exp. time, ROI, …
Tool	Start Cycle	Tool Parameters

Process Execution

Log process execution information to AML file

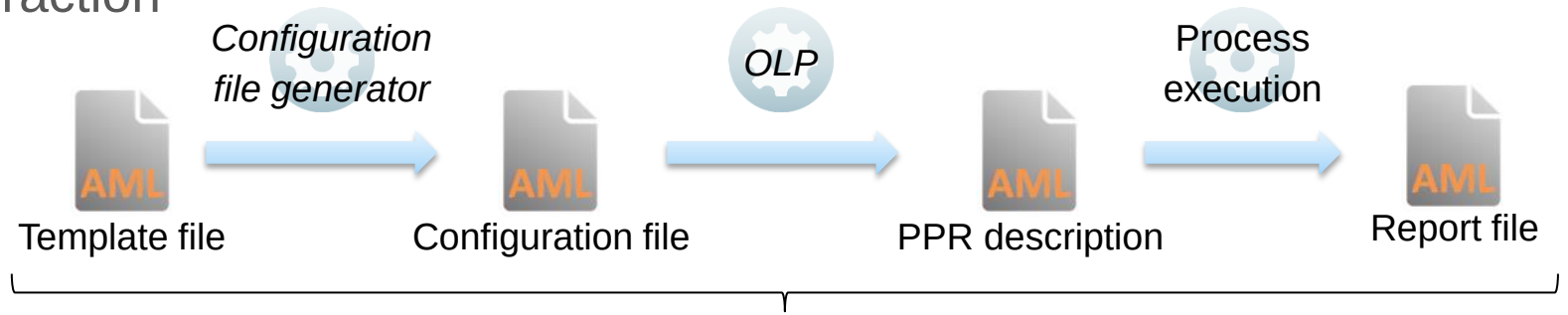
Information about process execution are written to AML report file:

- Execution time info
- Faults
- Sensors measures
- Task executed properly
- Other Critical Parameters
- …

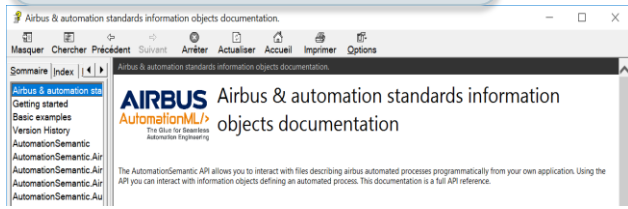


Integrated tool chains and interfaces

- File based information exchange
- AutomationML based Process-Product-Resource model for automated processes description
- AML based Airbus API for data interaction
- File generation & visualization



File content interaction with Airbus API



Perspectives and current interests

- Implementation of robotic ontology (ISO 8373) in AML
- Robot & machines motions modelling
- Automated processes modelling (PPR structure)
- Import of automation components information into OLP tools
- Digital semantic representation of products, process and resources for future automated process planning by automated resource skill to product requirement mapping

Thank you

AIRBUS