

A banner image for AutomationML. It features a blue-toned background with industrial automation elements: a robotic arm on the left, a car chassis in the center, and a control panel with a screen on the right. The text '<AutomationML/>' is prominently displayed in orange with a white outline.

<AutomationML/>

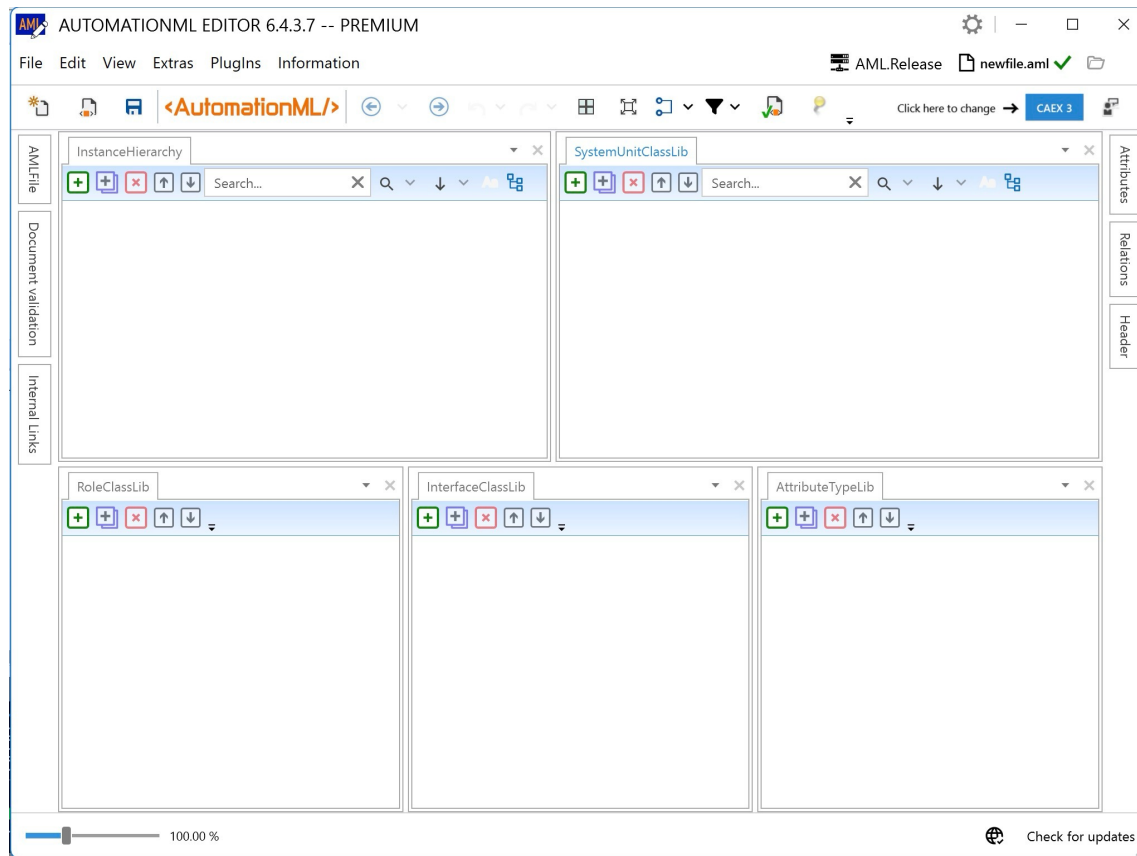
The Glue for Seamless
Automation Engineering

AI based plugin development for the AutomationML Editor

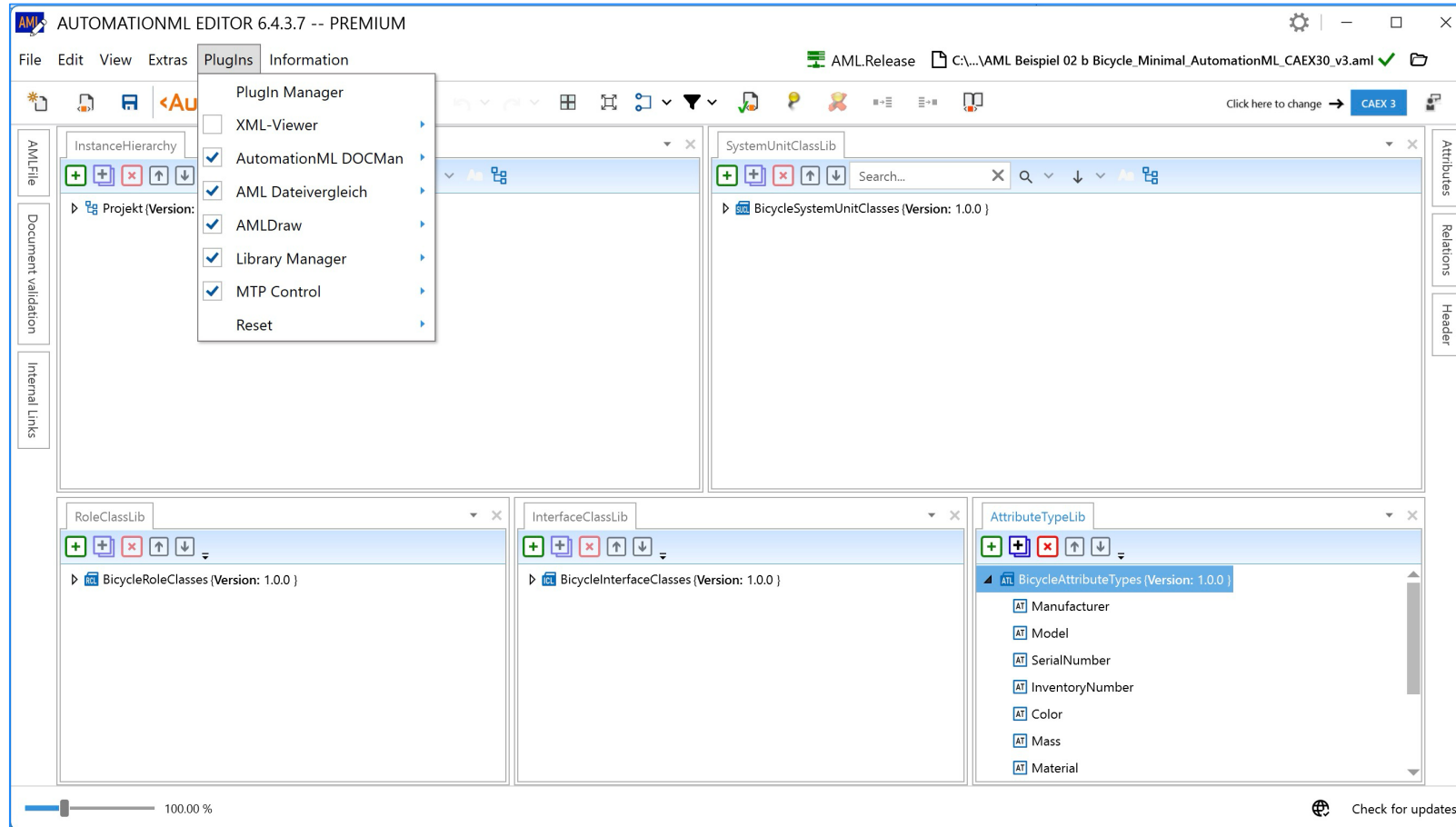
Rainer Drath

Pforzheim University of Applied Sciences

AML Editor plugin architecture



AML Editor plugin architecture



Example: XML Viewer

The screenshot displays the AUTOMATIONML EDITOR 6.4.3.7 -- PREMIUM interface. The main workspace is divided into several panes:

- InstanceHierarchy**: Shows a tree structure with "Projekt (Version: 1.0.0)".
- SystemUnitClassLib**: Shows a tree structure with "BicycleSystemUnitClasses (Version: 1.0.0)".
- RoleClassLib**: Shows a tree structure with "BicycleRoleClasses (Version: 1.0.0)".
- InterfaceClassLib**: Shows a tree structure with "BicycleInterfaceClasses (Version: 1.0.0)".
- AttributeTypeLib**: Shows a tree structure with "ATL BicycleAttributeTypes (Version: 1.0.0)".

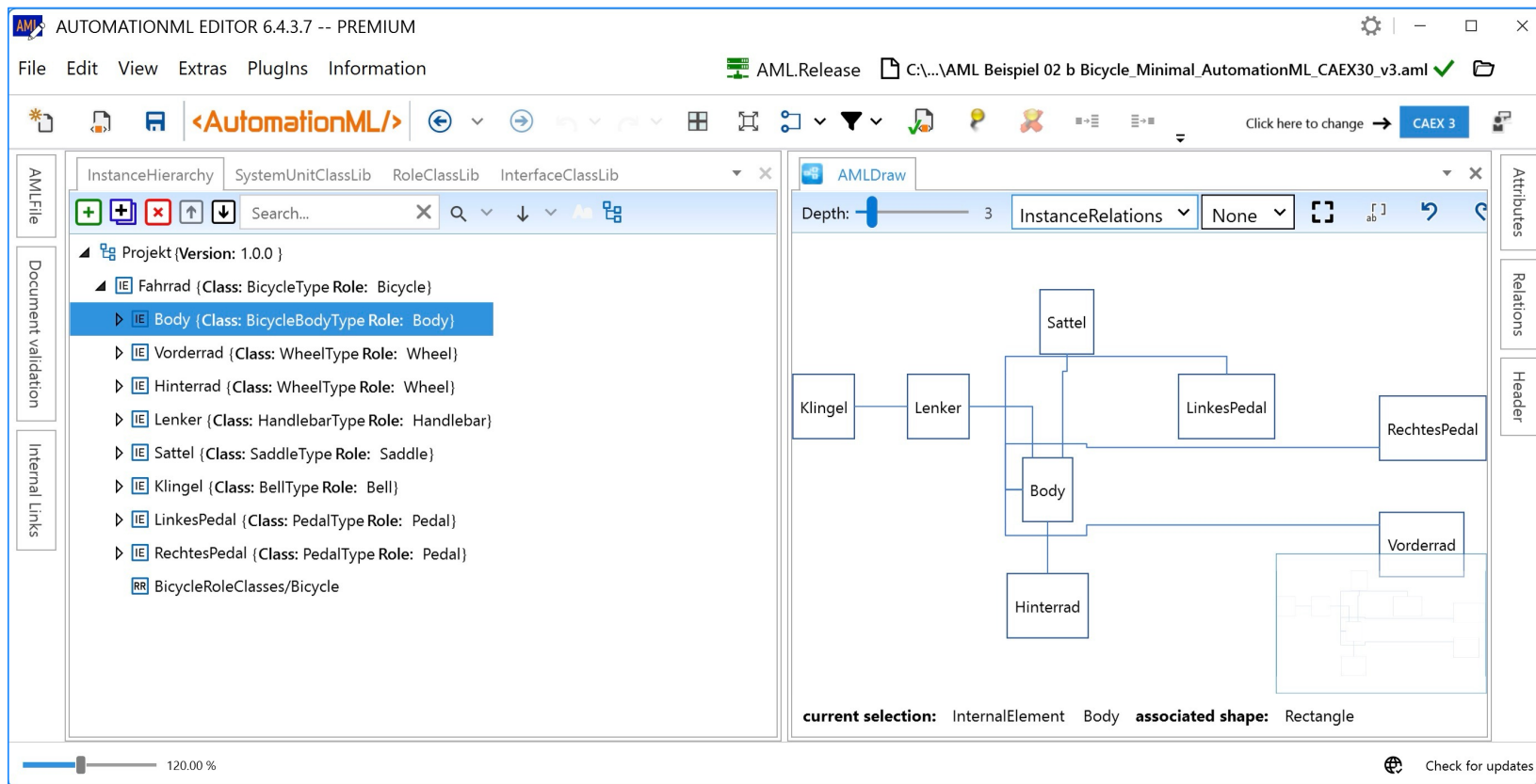
The **XML-Viewer** pane on the right displays the XML content of the selected element, "ATL BicycleAttributeTypes (Version: 1.0.0)". The XML is as follows:

```
<AttributeTypeLib
  Name="BicycleAttributeTypes"
  ID="a726fddc-003e-5692-b7d5-573cb7fa1ec0"
  xmlns="http://www.dke.de/CAEX"
  <Version>1.0.0</Version>
  <AttributeType
    Name="Manufacturer"
    ID="3775912f-a316-56ff-b39f-af3c97e6e146"
    AttributeDataType="xs:string">
    <Description>Manufacturer of an asset.</Description>
  </AttributeType>
  <AttributeType
    Name="Model"
    ID="474fbf62-3c94-5354-8792-0f5e664f2c61"
    AttributeDataType="xs:string">
    <Description>Commercial model designation.</Description>
  </AttributeType>
  <AttributeType
    Name="SerialNumber"
    ID="f36583d0-24dc-506b-9d63-fad7c78cddfc"
    AttributeDataType="xs:string">
    <Description>Unique serial number of a concrete asset.</De
```

The interface also includes a menu bar (File, Edit, View, Extras, PlugIns, Information), a toolbar, and a status bar at the bottom showing "120.00 %".

<AutomationML/>

Example: AML Draw



Example: AML Document Manager

AUTOMATIONML EDITOR 6.4.3.7 -- PREMIUM

File Edit View Extras Plugins Information

AML.Release C:\...\AML Beispiel 02 b Bicycle_Minimal_AutomationML_CAEX30_v3.aml

Click here to change → CADX 3

InstanceHierarchy SystemUnitClassLib RoleClassLib InterfaceClassLib AttributeTypeLib

Search...

BicycleAttributeTypes (Version: 1.0.0)

- Manufacturer
- Model
- SerialNumber
- InventoryNumber
- Color
- Mass
- Material
- FrameSize
- WheelDiameter
- TireWidth
- Width
- Diameter

AutomationML DOCMan

BicycleAttributeTypes

Attribute Manufacturer

Class name	Manufacturer
Description	Manufacturer of an asset.
Parent class	
Path for element reference	BicycleAttributeTypes/Manufacturer
Attributes	None

Attribute Model

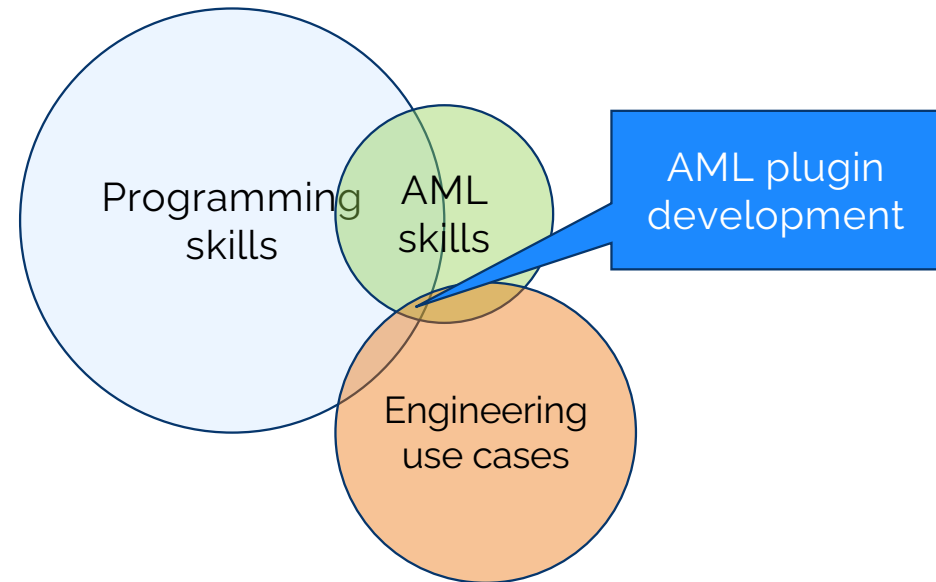
Class name	Model
Description	Commercial model designation.
Parent class	
Path for element reference	BicycleAttributeTypes/Model
Attributes	None

Attribute SerialNumber

Class name	SerialNumber
Description	Unique serial number of a concrete asset.
Parent class	
Path for element reference	BicycleAttributeTypes/SerialNumber
Attributes	None

82.82 % Check for updates

Preconditions for AML Editor Plugin Development



So far: Workflow for AML Editor Plugin Development



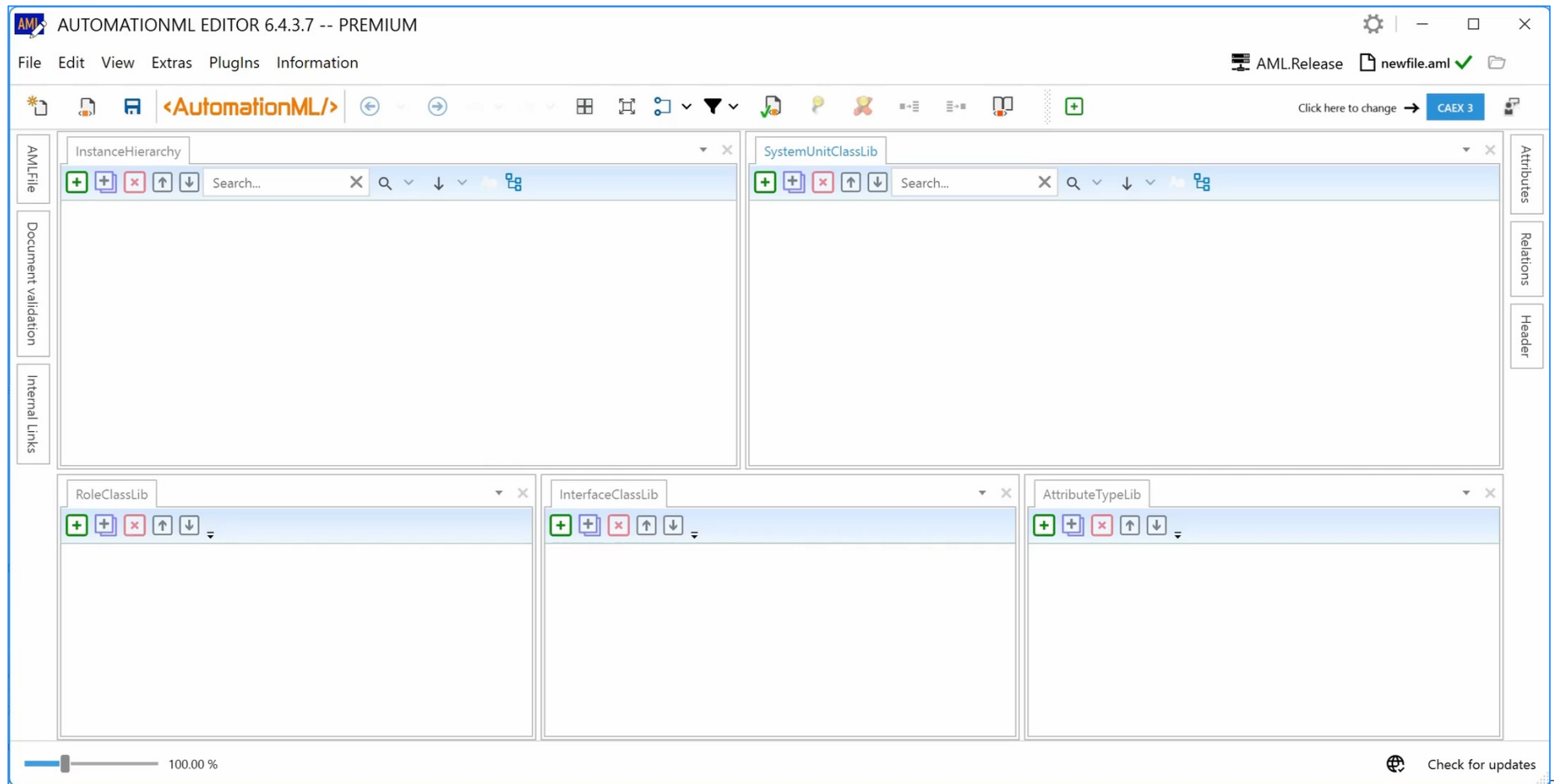
1. Create a new project directory:
`Aml.Editor.Plugin.<Name>\`
2. Create a .csproj file (there is a template)
3. Create the main class and XAML
4. Implement functionality (using the CAEX API)
5. dotnet build → generates .nupkg in build\Plugins\
6. publish globally, or install locally

Student work

I asked my students to find out how to simplify plugin development.

Phase 1: Simplify the AML Plugin template application

Idea 1: An AutomationML Plugin Generator



<AutomationML/>

Phase 2: Simplify the programming itself

With agentic AI

Prominent Systems for Agent-Based AI

The market is broad, and the pace of innovation is rapid. Today, these prominent examples are particularly helpful for putting things into

OpenAI Codex

Coding Agent for Software Development Using Terminal, DIE, APP, and Cloud Delegation

Claude Code

Antropic's terminal-based coding agent for understanding the codebase and making changes within the local workflow

GitHub Copilot coding agent

Asynchron Coding-Agent in GitHub for Issues, Pull Requests and Agent-Sessions

Devin

Cognition's autonomous software agent with its own working environment and review loops

Qwen 3.8 27B

Local AI System with High-Quality Coding

Perplexity Comet / Labs

Agent-based browser and project agent for research, web tasks, and complete artifacts

How to program with Agentic AI

- » Programming with Claude Code is a controlled working relationship.
- » We developed a *claude code skill* that explains the AML Editor plugin template and its application to claude code.

New Workflow

- 1. Formulate the Task: Clear request or specific improvement**
2. The agent reads the context and identifies assumptions.
3. Planning and execution: The agent breaks down the work and carries it out.
4. Testing and review: Changes are verified, not just made.
- 5. Handoff: You decide what is good enough**

From July–August 2026, we developed:

Students work:

1. AML to C# Code Generator
2. AI chatbot for AML Editor
3. Media-Attacher
4. MTP Control
5. MTP Checker
6. Text based Project Comparer
7. Bulk Editor

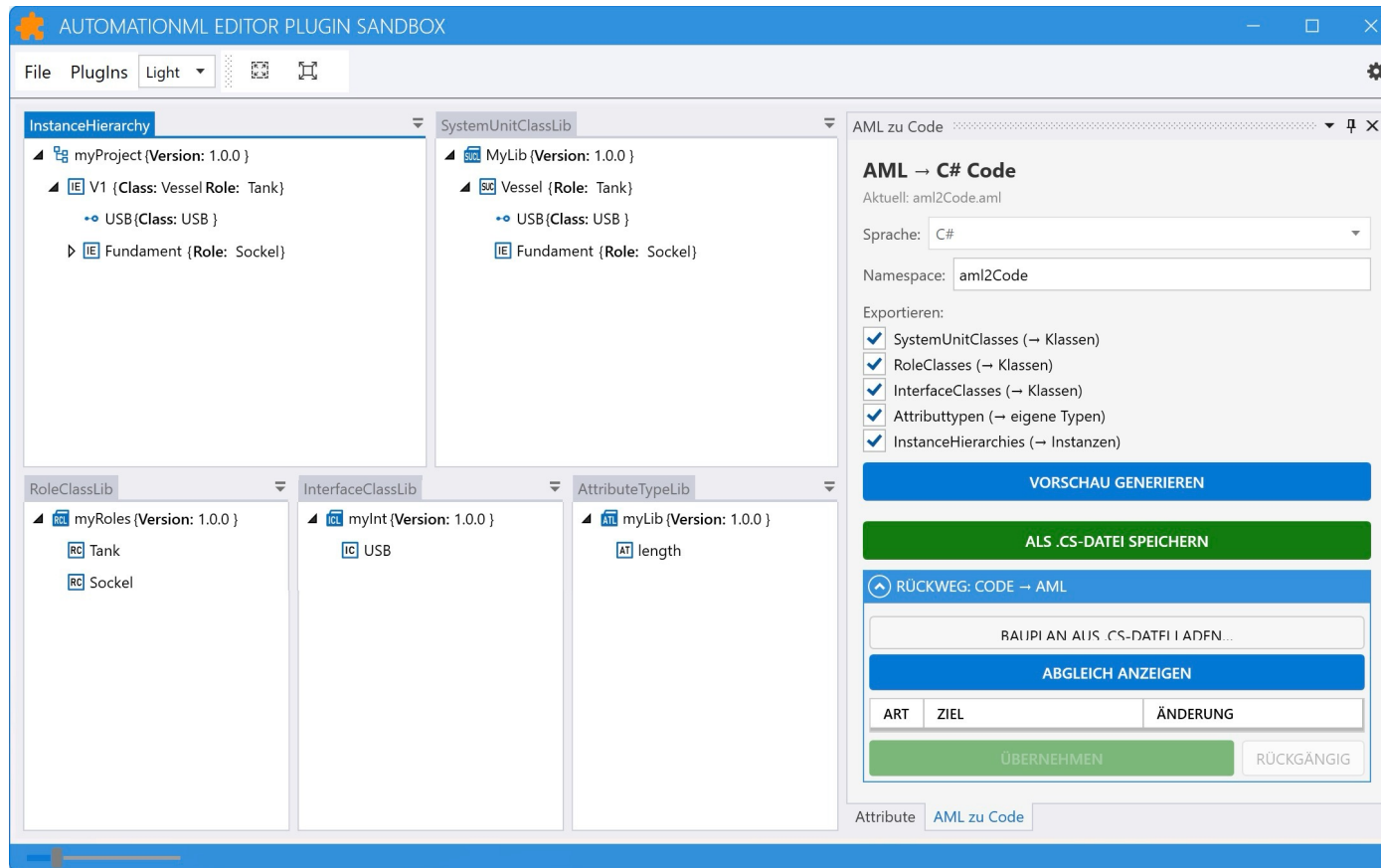
Rainer Draths work

1. Excel-Interface
2. Library Manager
3. ChatGPT Skill “General AutomationML Example Generator”

Student work

With agentic AI

1. AML2Code



1. AML2Code

```
using AutomationML.Model;

namespace aml2Code
{
    // --- Verwendungshilfen -----
    //
    // Je Klasse des Dokuments eine Zeile. Der Konstruktor setzt NUR den Bezug -
    // Einheit, Datentyp und alles Weitere gehoeren zur Definition in der
    // Bibliothek, nicht zu jeder Verwendung.

    public class length : AmlAttribute { public length() { RefAttributeType = "myLib/length"; } }

    public class Vessel : InternalElement { public Vessel() { RefBaseSystemUnitPath = "MyLib/Vessel"; } }

    public class USB : ExternalInterface { public USB() { RefBaseClassPath = "myInt/USB"; } }

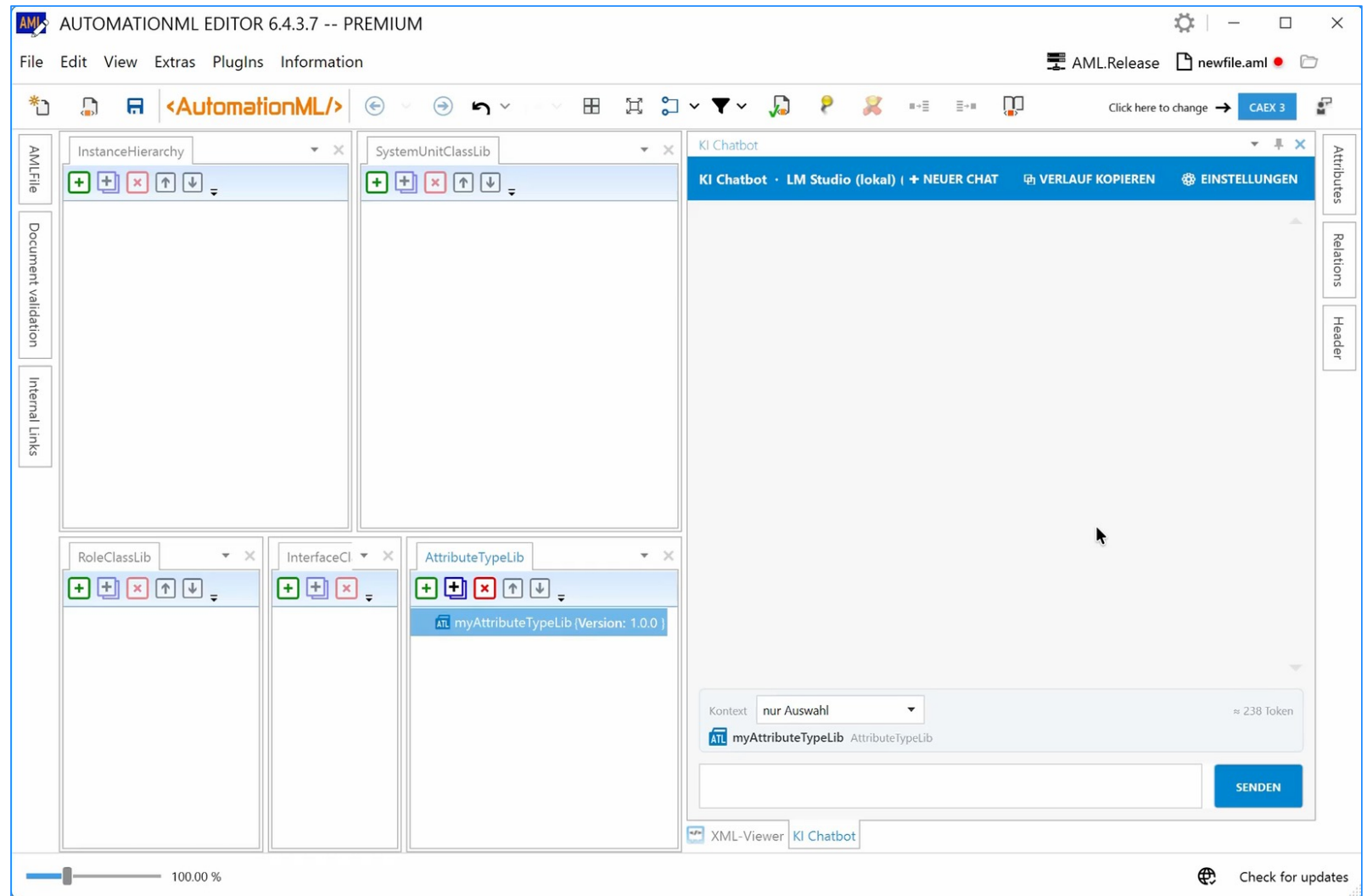
    public static class Rollen
    {
        public const string Sockel = "myRoles/Sockel";
        public const string Tank = "myRoles/Tank";
    }

    public static class aml2CodeDokument
    {
        public static AmlDocument Build()
        {
            var doc = new AmlDocument
            {
                SchemaVersion = "3.0",
                FileName = "aml2Code.aml",
                SuperiorStandardVersion = "AutomationML 2.1",
                SourceDocumentInformation = "<SourceDocumentInformation OriginName=\"AutomationML Editor\" Ori

            BaueAttributtypen_myLib(doc);
            BaueKlassen_MyLib(doc);
            BaueRollen_myRoles(doc);
            BaueSchnittstellen_myInt(doc);
            BaueInstanzen_myProject(doc);

            return doc;
        }
    }
}
```

2. AML Chatbot



3. Media Attacher

AUTOMATIONML EDITOR 6.4.3.7 -- PREMIUM

File Edit View Extras PlugIns Information

AML.Release C:\Temp\PluginTests\03 Media Attacher\MediaAttacher.aml

<AutomationML/>

Click here to change → CAEX 3

InstanceHierarchy

- Projekt (Version: 1.0.0)

SystemUnitClassLib

- myLib (Version: 1.0.0)
 - Robot (Version 1.0.0)
 - Hammer (Version 1.0.0)

RoleClassLib

InterfaceClassLib

AttributeTypeLib

- myATL (Version: 1.0.0)

Header: SystemUnitClass

Information	
Description	
Copyright	
Additional information	Additional information collection
Source	
Object source	Source object information collection
Versioning	
Change mode	Undefined
Version	1.0.0
Revision	Revision collection
Identification	
ID	cc21998d-4695-4698-97a1-512e040f4b36
Name	Hammer

Attributes Relations Header

100.00 %

Check for updates

3. Media Attacher

AUTOMATIONML EDITOR 6.4.3.7 -- PREMIUM

File Edit View Extras PlugIns Information

AML.Release C:\Temp\PluginTests\03 Media Attacher\MediaAttacher.aml

<AutomationML/>

Click here to change → CAEX 3

InstanceHierarchy

- Projekt (Version: 1.0.0)

SystemUnitClassLib

- myLib (Version: 1.0.0)
 - Robot (Version 1.0.0)
 - Hammer (Version 1.0.0)

RoleClassLib

InterfaceClassLib

AttributeTypeLib

- myATL (Version: 1.0.0)

Header: SystemUnitClass

Information	
Description	
Copyright	
Additional information	Additional information collection
Source	
Object source	Source object information collection
Versioning	
Change mode	Undefined
Version	1.0.0
Revision	Revision collection
Identification	
ID	cc21998d-4695-4698-97a1-512e040f4b36
Name	Hammer

Attributes Relations Header

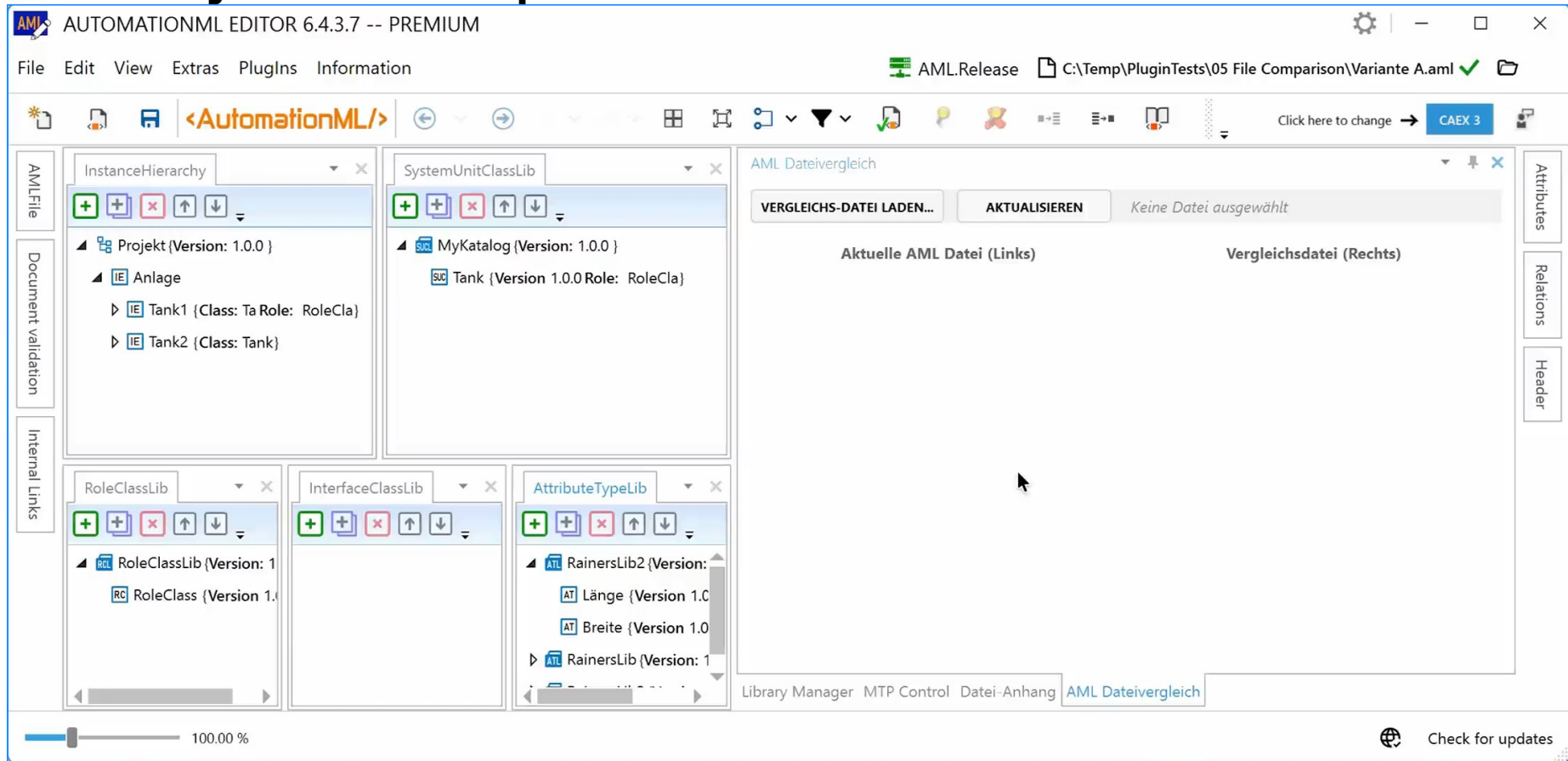
100.00 %

Check for updates

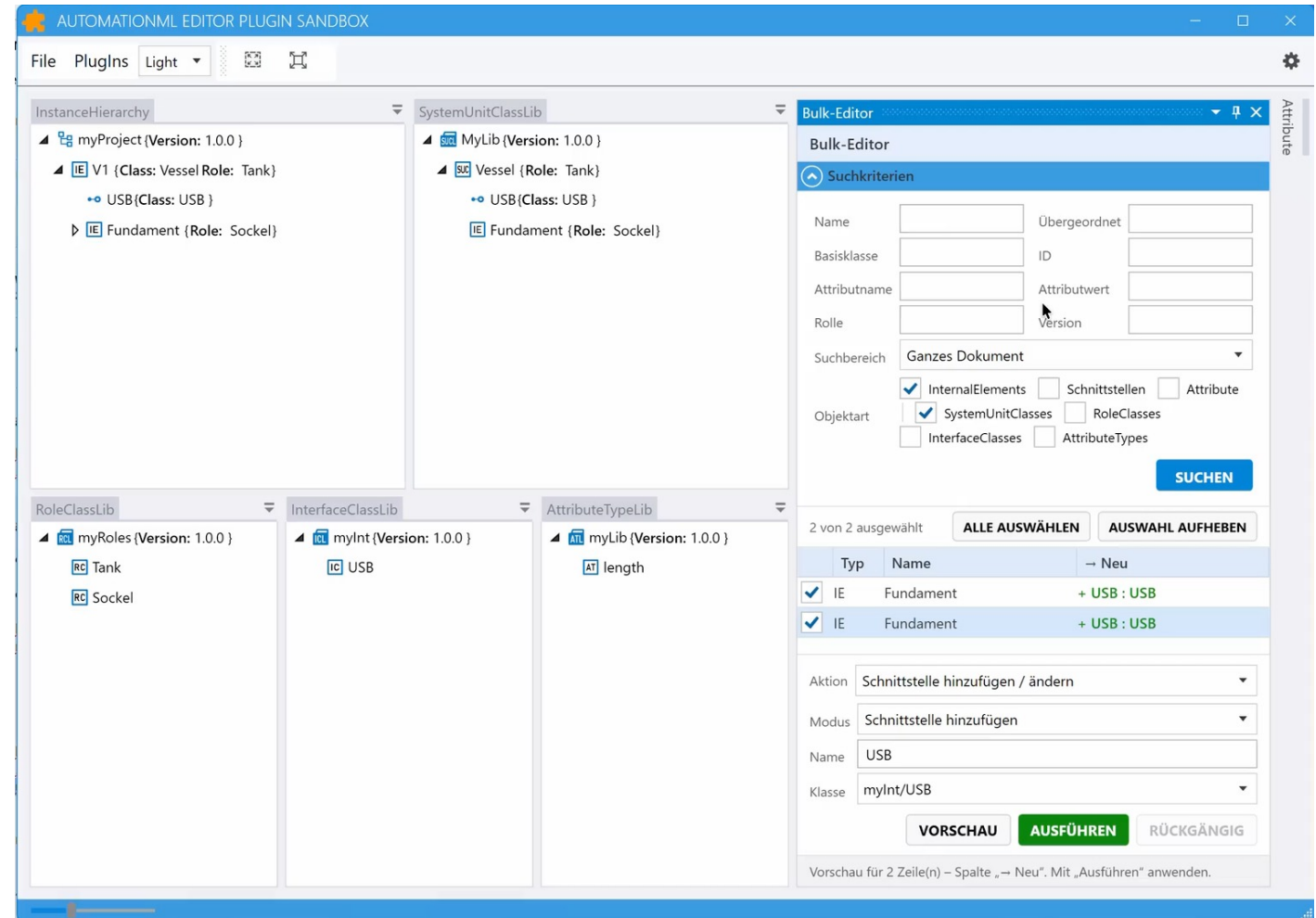
4.+5. MTP plugins

» See the other speech...

6. Project Comparison



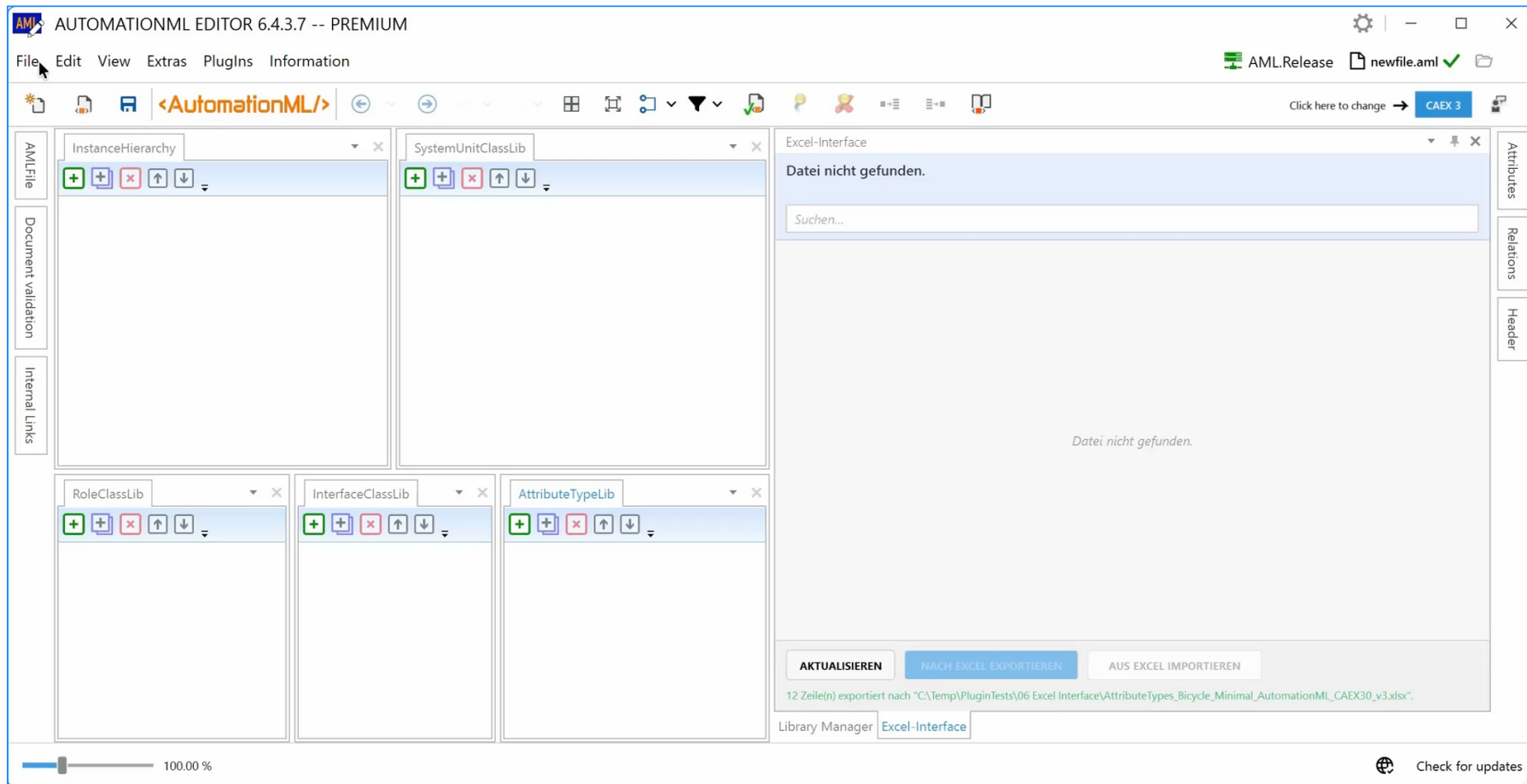
7. Bulk Editor



Rainer Draths work

With agentic AI

1. Excel-Interface



2. Library Manager



- » Inheritance tree visualization
- » Alphabetic sorting of classes in libraries
- » Cleanup of AML files (remove waste)
- » Search, find, navigate
- » Split Libraries into different files
- » Library move and switch assistant
- » File comparison on object level
- » Live Quality Checker with a rule engine, checking all rules of IEC 62424, IEC 62714 and Library Development Guideline

2. Library Manager

AUTOMATIONML EDITOR 6.4.3.7 -- PREMIUM

File Edit View Extras Plugins Information

AML.Release C:\...\AML Beispiel 02 b Bicycle_Minimal_AutomationML_CAEX30_v3.aml

Click here to change → CAEX 3

InstanceHierarchy SystemUnitClassLib

Projekt (Version: 1.0.0) BicycleSystemUnitClasses (Version: 1.0.0)

RoleClassLib InterfaceClassLib AttributeTypeLib

BicycleRoleClasses (Version: 1.0.0) BicycleInterfaceClasses (Version: 1.0.0) BicycleAttributeTypes (Version: 1.0.0)

Library Manager

AML Beispiel 02 b Bicycle_Minimal_AutomationML_CAEX30_v3.aml

1.136.0

Bewertet den Zustand des Dokuments in Prozent - getrennt nach Verletzungen der AML-Norm, Pflichtvorgaben und Empfehlungen der AML-Library-Guideline.

ANZEIGE

Gesundheitszustand: 79,5 %

Fehler: 88,6 % (110 Beanstandung(en) aus 61 Regel(n)) Empfehlung: 52,5 % (145 Beanstandung(en) aus 15 Regel(n))

Anzeigen: ☐ Befunde ☒ Regeln 76 Regel(n) insgesamt Regelwerk: Edition 2 ☐ Positivbefunde

Regeln durchsuchen (z. B. „Attribut“ oder „Mirror“)

Fehler — 88,6 % (61 Regel(n), davon 9 mit Beanstandungen)

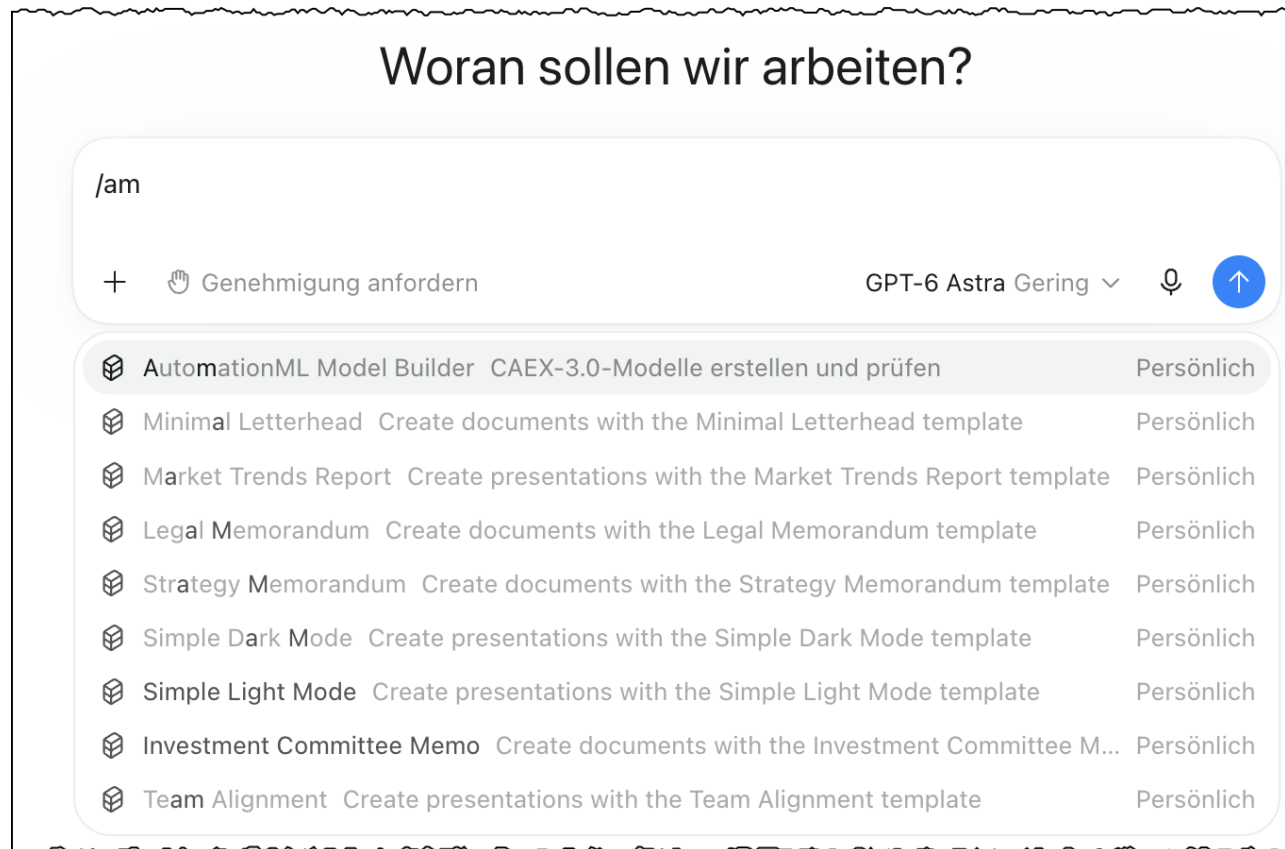
CAEX-001	Keine doppelten IDs im Dokument	Ed. 2+3	0 von 72	100,0 %
CAEX-002	SuperiorStandardVersion ist gesetzt (auch in referenzierten Dokumenten)	Ed. 2+3	0 von 1	100,0 %
CAEX-004	Referenzierte Dokumente folgen SchemaVersion 3.0	Ed. 2+3	0 von 0	nichts zu prüfen
CAEX-005	SourceDocumentInformation ist vollständig	Ed. 2+3	1 von 5	80,0 %
CAEX-006	Gleichrangige Klassen haben unterschiedliche Namen	Ed. 2+3	0 von 32	100,0 %
CAEX-008	Instanzen haben eine nicht leere ID	Ed. 2+3	0 von 35	100,0 %
CAEX-009	Mirror-/Internallink-Referenzen lösen auf	Ed. 2+3	14 von 14	0,0 %
CAEX-010	Bibliotheken haben eine Versionsnummer	Ed. 2+3	0 von 4	100,0 %
CAEX-012	Bibliotheks-/InstanceHierarchy-Namen sind eindeutig	Ed. 2+3	0 von 5	100,0 %
CAEX-013	Revision-Verweise (OldVersion/NewVersion) lösen auf	Ed. 2+3	0 von 0	nichts zu prüfen
CAEX-014	Eltern-Klassen existieren	Ed. 2+3	35 von 52	32,7 %
CAEX-016	Klassen erben nur von derselben Klassenart	Ed. 2+3	0 von 17	100,0 %

REGELN NACH EXCEL EXPORTIEREN

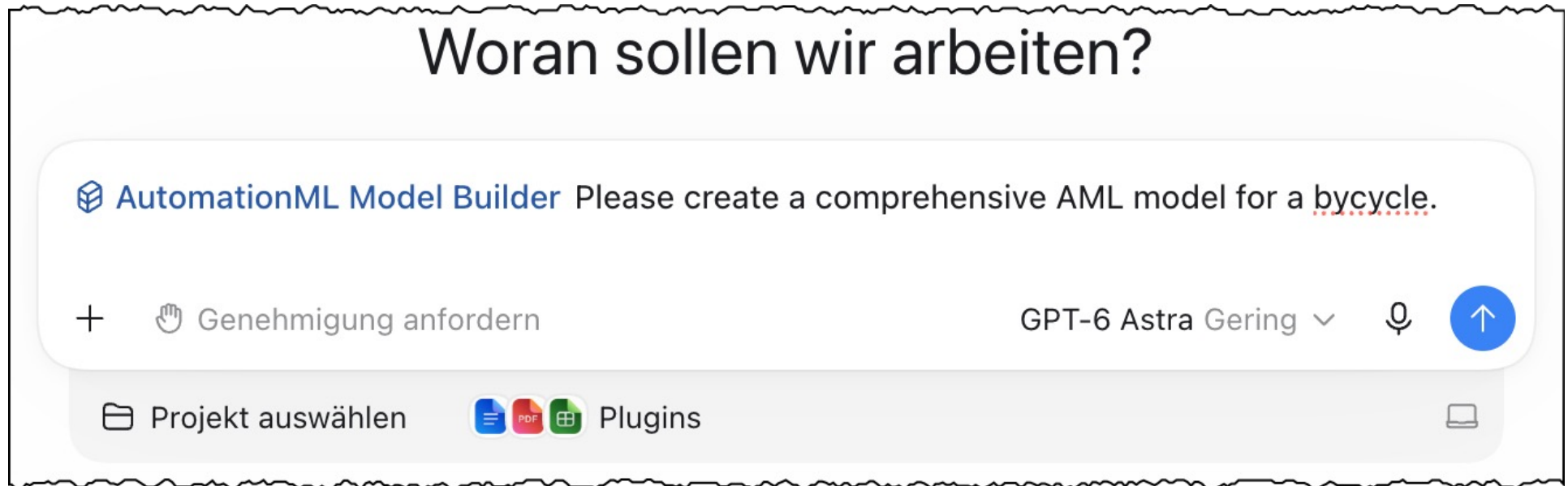
100.00 %

Check for updates

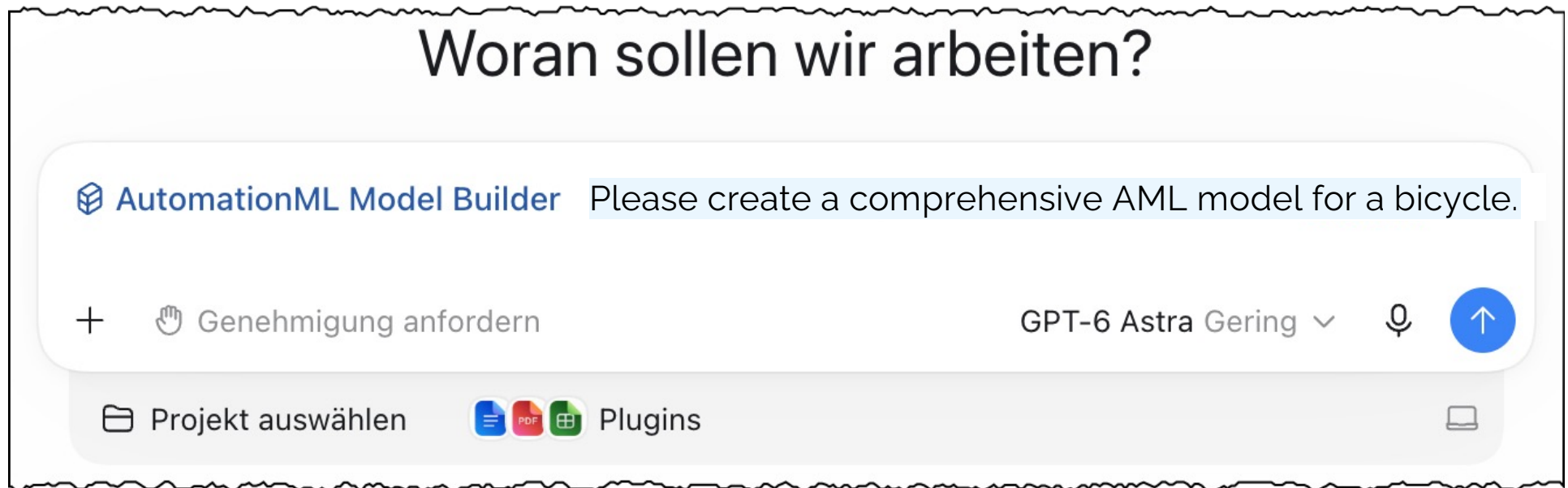
3. ChatGPT Skill „Generic AML Example Generator“



Prompt your example



Prompt your example



AutomationML EDITOR 6.4.3.7 -- PREMIUM

File Edit View Extras PlugIns Information

AML.Release C:\...\AML Beispiel 02 a Bicycle_Groß_AutomationML_CAEX30_v2.aml

<AutomationML/>

Click here to change CAEX 3

InstanceHierarchy

Projekt (Version: 1.0.0)

SystemUnitClassLib

BicycleSystemUnitClasses (Version: 1.0.0)

RoleClassLib

BicycleRoleClasses (Version: 1.0.0)

InterfaceClassLib

BicycleInterfaceClasses (Version: 1.0.0)

AttributeTypeLib

BicycleAttributeTypes (Version: 1.0.0)

AMLDraw

Depth: 3 InstanceRelations

Attributes

Relations

Header

current selection: InternalElement Frame associated shape: Rectangle

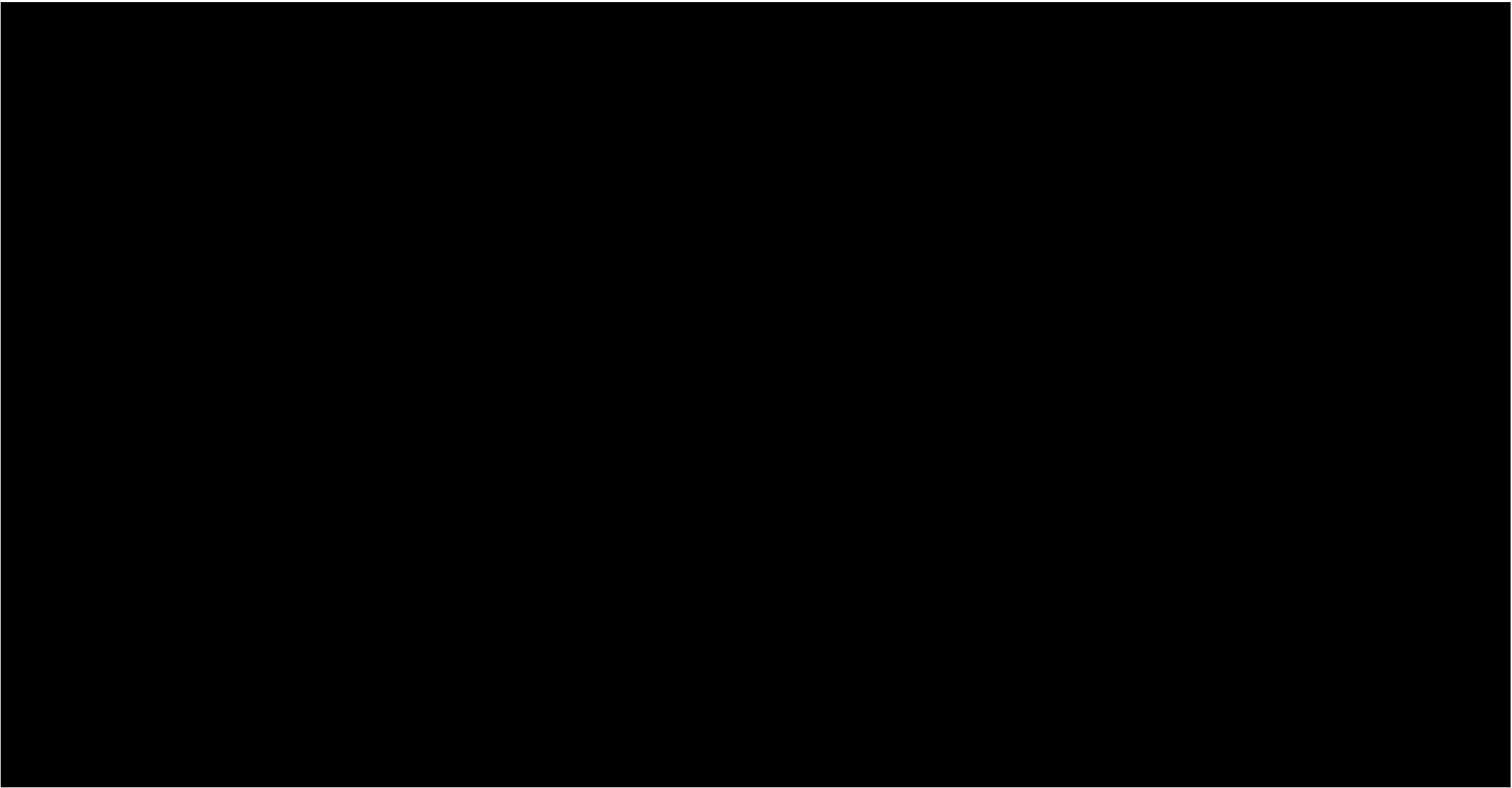
How you can write your own AML Editor plugins

Prompt for a DEMO plugin

"

1. The plugin should iteratively search the AML file currently open in the AML Editor.
2. It searches all instances (InternalElement, Interface, Attribute) that lack a class reference or a description.
3. Display the results in a table. The table should have the columns: "Library," "Name," and "Missing."
4. When an item in the table is selected, the corresponding element in the AML Editor should be selected too. If the AML element is hidden, it should be expanded.
5. Provide a function that exports the resulting table to EXCEL.

"



AutomationML/

Lessons Learned

- » Agentic AI is a game changer.
- » Agentic AI reads context, plans steps, uses tools and delivers tested results.

Recommendations for Agent-Based Programming

The more influence an agent has, the more important leadership, adaptability, and the ability to accept criticism become on the human side.

Rule 1: Goal Before Solution: Define what success looks like, not just how you'd roughly go about building it.

Rule 2: Break It Down into Small Enough Pieces: Delegate work packages, not vague wish lists. Good division of labor creates good momentum

Rule 3: Explicitly state boundaries: Files, risks, quality rules, tests, and non-goals must be clearly defined

Rule 4: Trust only after review

Agent-based systems can accomplish a lot. But judgment, prioritization, and responsibility is on the humans side.

Summary



- » Agentic AI is a game changer in AML Editor plugin development
 - Describe the modelling challenge in your own words.
 - Let AI create the plugins and handle the AML details. Review the result and refine the solution.
- » Less time coding. More focus on engineering know-how and engineering use cases.
- » My experiences: performance factor 60x-90x compared to manual development, testing and development

Take-Away

- » Plugins
- » Claude skill for plugin development
- » ChatGPT skill for AML example generation



AutomationML e.V. c/o IAF
Universitätsplatz 2
39106 Magdeburg
office@automationml.org