

# Anforderungen an eine Engineering-Plattform für die KI-basierte Automation

René Heesch\*, Julian Putzke\*, Simon Althoff<sup>‡</sup>, Philip Topalis\*, Marvin Schieseck\*,  
Alexander Fay\*, Oliver Niggemann\*

\*Institut für Automatisierungstechnik  
Helmut Schmidt Universität  
Hamburg, Deutschland

Email: rene.heesch@hsu-hh.de

<sup>‡</sup>Weidmüller Interface GmbH & Co. KG  
Detmold, Deutschland

Email: simon.althoff@weidmueller.com

**Zusammenfassung**—Das Engineering von Applikationen der künstlichen Intelligenz (KI) für Automatisierungssysteme ist eine domänenübergreifende Tätigkeit, welche in besonderer Weise das Zusammenbringen der Welt der Softwareentwicklung und der Welt der Automatisierungstechnik erfordert. Erschwerend kommt hinzu, dass die Automatisierungshardware zumeist nur mit einer herstellerabhängigen Software konfiguriert werden kann. Eine herstellerunabhängige und quelloffene Lösung, welche einerseits das Engineering von Automatisierungslösungen und korrespondierende KI-Applikationen ermöglicht und andererseits die Automation durch software-integrierte Assistenzfunktionen für die KI-Services unterstützt, existiert bisher nicht. Im dtec.bw-Projekt EKI wird dieses Problem adressiert und eine Engineering-Plattform für die einfache Integration von KI-Applikationen und mit KI-basierten Assistenzfunktionen entwickelt. In diesem Beitrag werden die Anforderungen an eine solche herstellerunabhängige und quelloffene Engineering-Plattform aufgezeigt. In diesem Zuge wird ebenfalls analysiert, welche bestehenden Lösungen diese Anforderungen adressieren und vorgestellt, wie die Anforderungen bei der Konzeptionierung dieses neuen Engineeringansatzes umgesetzt werden sollen.

**Index Terms**—CPPS, Engineering, KI-Integration, Automatisierung, Engineering-Plattform

## I. EINLEITUNG

Im heutigen Produktionsumfeld sehen sich Unternehmen zunehmend mit immer häufiger variierenden Produkthanforderungen und damit immer kleiner werdenden Losgrößen konfrontiert [1]. Dies hat zu einem Bedarf an neuen adaptiven und veränderbaren Produktionsanlagen sowie den dazugehörigen Automatisierungslösungen geführt. Gerade das Engineering, also das Anpassen der Automatisierungslösung durch Experten, hat sich in diesem Zusammenhang als Engpass erwiesen. Als Cyber-physische Produktionssysteme (CPPS) bezeichnet man eine Klasse von Produktionssystemen, die für schnelle Anpassungen an veränderte Anforderungen und Bedingungen besonders geeignet sind. Sie bestehen in der Regel aus unterschiedlichen Modulen, die wiederum an variierende Rahmenbedingungen angepasst und miteinander kombiniert werden können. Mit ihrem autonomen und kooperativen Verhalten auf allen Produktionsebenen unterscheiden sie sich signifikant von traditionellen, hierarchischen Automationsstrategien [2].

Um die Interoperabilität innerhalb der CPPS und mit anderen Systemen zu gewährleisten, müssen Informationen über die Systeme maschinenlesbar dargestellt werden. In diesem Zusammenhang wird die Nutzung von semantischen Beschreibungsmitteln, wie beispielsweise Ontologien, untersucht [3]. So können die Funktionen von einzelnen Ressourcen innerhalb der CPPS als abstrakte Beschreibung in Form von *Fähigkeiten* dargestellt werden. Die Implementierung der entsprechenden Fähigkeiten, mit Hilfe derer die Fähigkeit von dem Modul beschrieben und aufgerufen werden kann, wird hingegen als *Skill* bezeichnet [4]. Ein solches Produktionssystem ermöglicht die Nutzung von neuen Automationsansätzen, die auf symbolischen und subsymbolischen Methoden der Künstlichen Intelligenz (KI) basieren [5]. Des Weiteren spielen Methoden der KI in der Produktion im Allgemeinen eine immer bedeutendere Rolle und gehören in vielen Bereichen der Produktion zum Stand der Technik. So können Methoden der KI beispielsweise sowohl für das Überwachen des Maschinenzustandes und das Optimieren von Wartungsintervallen in Form von *Predictive Maintenance* oder für die optische Überwachung der Produktqualität in Form von *automatischer optischer Inspektion* genutzt werden. Für das Auswählen und das Konfigurieren des passenden Ansatzes wird jedoch in der Regel Expertenwissen benötigt. Des Weiteren erfolgt die Integration einer KI-Komponente im Produktionssystem zurzeit nachgelagert und nicht direkt im Prozess des Engineerings, was die Kommunikation mit dem Automationssystem erschwert.

Engineering-Plattformen, welche sich alleine mit der Programmierung von Automatisierungslösungen (z.B. SPS) beschäftigen, fehlt die Funktionalität für die Integration und Anwendung von KI-basierten Assistenzfunktionen sowie die Integration von KI-Anwendungen. Stattdessen gibt es eine Vielzahl von Plattformen der unterschiedlichen Hersteller von Automatisierungshardware. Diese Plattformen können häufig lediglich für das Engineering der zugehörigen Hardware genutzt werden [6].

Aus diesem Grund wird im Rahmen des DTEC-Projektes 'EKI - Engineering für die KI-basierte Automation in virtuellen und realen Produktionsumgebungen' in Zusammenarbeit

mit dem Projektpartner Weidmüller Interface GmbH & Co. KG eine offene und erweiterbare Engineering-Plattform entwickelt. Diese Engineering-Plattform soll sowohl Echtzeit- wie Nicht-Echtzeit-KI-Applikationen ausführen können. Zudem soll die Engineering-Plattform über KI-basierte Assistenzfunktionen verfügen, die den Anwender bei der Auswahl und Integration von KI-Algorithmen sowie bei der Ablaufplanung und Parametrisierung unterstützen. So soll beispielsweise eine Assistenzfunktion entwickelt werden, die automatisiert eine Sequenz von parametrisierten Skill-Aufrufen erstellen kann. Diese greift dabei auf KI-basierte Planungsansätze zurück und nutzt als Input die Fähigkeitsbeschreibungen der zur Verfügung stehenden Module und stellt fest, welche Module eines CPPS mit welchen Parametersätzen benötigt werden, um ein gewünschtes Produkt herzustellen. Die Sequenz von Skill-Aufrufen kann anschließend von einer übergeordneten Steuerung ausgeführt werden, welche von den betreffenden Modulen den entsprechenden Skill aufruft. Anlässlich der Erweiterbarkeit der Plattform soll es zudem möglich sein, weitere Assistenzfunktionen in die Plattform einzubringen.

Im Rahmen dieses Beitrages werden in einem ersten Schritt die Anforderungen an eine Engineering-Plattform definiert, welche für die Integration von KI-Komponenten im Engineering geeignet ist und eine Erweiterung von KI-basierten Assistenzfunktionen zulässt. Im Anschluss erfolgt eine Betrachtung des Standes der Technik hinsichtlich der Anforderungen. Abschließend wird ein Konzept für die zu entwickelnde Plattform sowie die geplante Validierung anhand einer Realanlage vorgestellt.

## II. ANFORDERUNGEN

Im Rahmen des Projektes wurden verschiedene Anforderungen an eine offene und KI-integrierbare Engineering-Plattform identifiziert. Speicherprogrammierbare Steuerungen (SPS) sind ein elementarer Bestandteil heutiger Automatisierungssysteme [7]. Es existiert eine Vielzahl unterschiedlicher SPS-Hersteller. Obwohl es den IEC61131 Standard für das Programmieren von Steuerungen gibt, nutzt der Großteil der Hersteller eine eigene Ausprägung der Sprache. Dies hat zur Folge, dass für das Programmieren der Steuerungen von unterschiedlichen Herstellern in der Regel auch jeweils eine herstellerabhängige integrierte Entwicklungsumgebung (IDE) benötigt wird, für die ein Nutzer oft zusätzliche Lizenzgebühren entrichten muss [6]. Zudem erhöht sich der Druck auf produzierende Unternehmen, flexibler zu werden und ihre Produktionssysteme, beispielsweise aufgrund sich wandelnder Kundenanforderungen, schneller anpassen zu können, um wettbewerbsfähig zu bleiben [1]. Die beschriebene Vielzahl von unterschiedlicher Engineering-Software, die auf dem Gerät des Automatisierungsingenieurs installiert werden muss, und jeweils nur für ausgewählte Steuerungen des Maschinenparks verwendet werden kann, steht der schnellen Anpassung der Automatisierung jedoch im Wege. Daraus leiten sich folgende Anforderungen an eine offene und KI-fähige Engineering-Plattform ab:

*Anforderung 1: Mithilfe der Plattform muss das Engineering unabhängig vom Hersteller einer Steuerung möglich sein.*

*Anforderung 2: Die Engineering-Plattform muss innerhalb kurzer Zeit und ohne komplexe Lizenzmodelle auf dem Gerät des Automatisierungsingenieurs einsatzbereit sein.*

Aufgrund der Komplexität der zu automatisierenden Prozesse verfügen moderne Produktionsanlagen in der Regel über mehr als eine SPS. Neben den konventionellen Kommunikationsschnittstellen für die Kommunikation zwischen einzelnen Hardwarekomponenten der Automatisierung existieren ebenfalls Standards für den Datenaustausch zwischen Geräten unterschiedlicher Hersteller, wie die OPC Unified Architecture (OPC UA).

*Anforderung 3: Für das Engineering von komplexen Automatisierungssystemen ist es notwendig, dass die Engineering-Plattform das Konfigurieren gängiger Kommunikationsschnittstellen unterstützt.*

Ein weiterer Trend, mit dem ein Automatisierungsingenieur im heutigen Produktionsumfeld konfrontiert wird, ist der zunehmende Einsatz von Methoden der KI in der Produktion. So stellte das Bundesministerium für Wirtschaft und Energie bereits 2020 fest, dass KI in der Produktion eine intelligente Automatisierung von Prozessen, beispielsweise mithilfe von Predictive Maintenance (PM), erlaubt [8]. PM bezeichnet dabei die vorausschauende Wartung von Anlagen. Anders als bei einer Wartung in festen Intervallen, wird hier mit Hilfe Methoden der KI, basierend auf Maschinen- und Prozessdaten, der optimale Zeitpunkt für eine Wartung bestimmt [9]. PM ist dabei nur eine mögliche Anwendung von Methoden der KI in der Produktion. So ist auch die Nutzung von Bildverarbeitungsmethoden für die Qualitätsüberwachung und Identifikation in der Fertigung beispielsweise ein gängiges Mittel in der Automobilindustrie [10].

*Anforderung 4: Mithilfe der Engineering-Plattform muss es möglich sein, KI-Anwendungen zu konfigurieren und diese im Engineering automatisierter Anlagen zu integrieren.*

Insbesondere, aber nicht ausschließlich, subsymbolische Methoden aus dem Bereich des Maschinellen Lernens (ML) bergen dabei besondere Herausforderungen für die Integration. Das Ziel von ML-Algorithmen ist das Erkennen von Mustern und Gesetzmäßigkeiten in Datensätzen. Zu diesem Zweck müssen die entsprechenden Datensätze in ausreichender Qualität und Quantität vorliegen. In Abhängigkeit des Algorithmus wird anschließend auf Basis der vorliegenden Daten ein ML-Modell trainiert. Die Anwendung des trainierten Modells auf neue Daten wird als Inferenz bezeichnet [9]. Da eine SPS nicht für das Ablegen größerer Datenmengen geeignet ist, führt dies zu einer weiteren Anforderung:

*Anforderung 5: Innerhalb der Engineering-Plattform muss eine Anbindung der SPS an eine Cloud für die Datenspeicherung ermöglicht werden. In diesem Zuge muss es ebenfalls möglich sein, die Datenspeicherung zu konfigurieren, um neben den zu speichernden Daten ebenfalls die Abstrakte jedes Parameters festzulegen.*

Für das Training des ML-Modells muss außerdem ausreichend Rechenleistung zur Verfügung stehen. Der Rechen-

aufwand ist dabei von dem geplanten Anwendungsfall und dem ausgewählten Algorithmus abhängig. Allerdings reicht die Leistungsfähigkeit der Steuerungen für die Berechnung von aufwendigen Algorithmen nicht aus [11]. Ebenfalls in Abhängigkeit des Anwendungsfalls und des Algorithmus muss die Inferenz entweder vor Ort auf der Steuerung oder in der Cloud ausgeführt werden. Dies führt zu zwei weiteren Anforderungen:

*Anforderung 6: Es muss aus der Engineering-Plattform möglich sein, das Training und die Anwendung von KI-Algorithmen in der Cloud zu konfigurieren, damit jederzeit genug Rechenressourcen für die Umsetzung von KI-Algorithmen zur Verfügung stehen.*

*Anforderung 7: Innerhalb der Engineering-Plattform muss es möglich sein, trainierte Modelle aus der Cloud zu exportieren und auf der SPS zu deployen, damit die Inferenz auf der SPS erfolgen kann.*

Neben der Integration von KI-Algorithmen im Engineering von Produktionsanlagen existieren ebenfalls Ansätze, mit Hilfe von symbolischen oder subsymbolischen Methoden der KI das Engineering automatisierter Anlagen an sich zu erleichtern. Diese Assistenzfunktion betrifft das Programmieren der Automatisierungslösung, z.B. unter Nutzung der IEC61131. Hier soll die Möglichkeit geschaffen werden, dass eine in die Engineering-Lösung integrierte KI-basierte Assistenzfunktion dem Automatisierungsingenieur Vorschläge für die Programmierung von Anlagen anbietet. Diese sind zum Großteil jedoch noch Stand der Forschung und können noch nicht in der Breite angewendet werden. Dennoch sollten grundsätzlich neue Ansätze, die den Automatisierungsingenieur beim Engineering unterstützen, in der Engineering-Plattform integriert und somit der Plattform neue Funktionalitäten hinzugefügt werden können.

*Anforderung 8: Es muss möglich sein, die Engineering-Plattform zu erweitern und beispielsweise mit KI-basierten Assistenzfunktionen auszustatten, welche ebenfalls auf alle, zur Verfügung stehenden Schnittstellen zugreifen können.*

### III. STAND DER TECHNIK

Im Folgenden werden aktuelle Entwicklungen und Lösungen im Zusammenhang mit den definierten Anforderungen an Engineering-Plattformen vorgestellt.

Mit der Einführung der SPS wurden von verschiedenen Herstellern unterschiedliche Laufzeitumgebungen und Programmiersprachen für ihre Hardware entwickelt. Um die dadurch entstandene Komplexität für die Nutzer zu reduzieren, wurde von der International Electrotechnical Commission der IEC61131 Standard ausgearbeitet, mithilfe derer die Wiederverwendung von Code erleichtert werden soll. Für die Wiederverwendung von Code ist es jedoch zusätzlich notwendig, dass die Informationen, die auf der einen SPS programmiert wurden, auf andere SPS übertragen werden können und das Programm dieselben Funktionen aufweist wie zuvor. Um dieses Problem zu adressieren, wurde vom PLCopen Technical Committee 6 ein erweiterbares XML-Interface definiert. PLCopen ist eine

Organisation, die Standards im Bereich der industriellen Steuerungstechnik definiert, durch deren Anwendung die Effizienz bei der Entwicklung und Wartung von industrieller Softwareapplikationen gesteigert werden soll. Das Ziel des XML-Interfaces ist das Übertragen von Programmcode, vollständigen Projekten oder benutzerdefinierten Funktionsblöcken von einer Entwicklungsumgebung in eine andere [12].

Mit der Weiterentwicklung der Hardware und der zunehmenden Komplexität der Programme, die durch die Hardware ausgeführt werden kann, wurden auch die integrierte Entwicklungsumgebungen (IDEs) zum Entwickeln der Programme deutlich komplexer. Dies führt wiederum zu einem erhöhten Aufwand für die Installation der entsprechenden Software. Eine Möglichkeit, dies Problem zu umgehen, bieten web-basierte IDEs. Im Vergleich zu einer IDE, welche lokal auf dem Gerät des Anwenders installiert ist und dann auch nur dort verwendet werden kann, bietet eine web-basierte IDE die Flexibilität, auf nahezu jedem Gerät ohne Installation verwendbar zu sein. Der Benutzer muss lediglich einen Browser öffnen und sich über den entsprechenden Server mit der web-basierten IDE verbinden und kann anschließend in dieser arbeiten [13].

Aufgrund der zunehmenden Bedeutung von KI-Methoden im industriellen Umfeld gewinnt auch die Art und Weise, wie diese im Engineering integriert werden können, zunehmend an Bedeutung. Ein integrierter Ansatz, für die direkte Erstellung und Anbindung von KI-Algorithmen im Engineering einer vollautomatisierten Anlage, existiert zurzeit nicht. So sind Lösungen vorhanden, welche in erster Linie das Ausführen von KI-Anwendungen fokussieren. Hierbei wird gängigerweise die KI-Anwendung auf ein Maschinen- oder Prozessdatenbasierten Maschine-Learning-Ansatz (ML-Ansatz) reduziert. Das typische Vorgehen ist die Einbindung von Datensammlern in die Fertigung, um Datensätze für das Trainieren und Validieren der zu erzeugenden Modelle zu generieren. Diese Datensätze werden dann in gängigen Frameworks wie PyTorch, TensorFlow, SciKit-Learn<sup>®</sup> genutzt, um Modelle zu erzeugen. Hierdurch ergeben sich kaum Einschränkungen, wo die Daten gespeichert oder die Modelle trainiert werden. Allerdings erfolgt die Datensammlung durch manuelle Konfiguration und muss für die Inferenz auf einer SPS erneut durchgeführt werden. Durch die Anwendung außerhalb einer Engineering-Plattform fallen allerdings Synergien weg, da Schnittstellen mehrfach angebunden oder Datenflüsse wiederkehrend manuell angelegt werden müssen [14].

Eine Möglichkeit zum Erstellen der benötigten Datenflüsse bieten sogenannte Datenfluss-orientierte (dataflow) Programme. Sie wurden entwickelt, da viele Industrianwendungen die Integration von Online-Services und nahezu Echtzeitanwendung benötigen. Obwohl dies auch mit herkömmlichen Tools möglich ist, kann die Umsetzung hinsichtlich der Erstellung und Verknüpfung von Datenverarbeitungsschritten, APIs, zeitaufwendig sein [15]. Dataflow Programme, wie NodeRed oder WoTKit Processor, bieten eine Lösung für dieses Problem. Sie verfügen über grafische Oberflächen, in denen gekapselte Softwareobjekte miteinander verbunden und ausgeführt werden können. Nodes stellen dabei Dateninputs, Datenoutputs

oder Funktionen dar, während Pfeile die Nodes verbinden und den Datenfluss dazwischen bestimmen. NodeRed ist ein web-basiertes Tool, das den Fokus auf die Konnektivität zwischen der Hardware und den APIs setzt. Zudem ist es ein offenes Tool, was von der Erweiterung der Community profitiert.

Bezüglich der Ausführung von trainierten KI-Modellen bestehen heute schon Lösungen, die das Ausführen der Inferenz sowohl in Echtzeit als auch Nicht-Echtzeit ermöglichen. Die nicht-Echtzeit-fähigen Applikationen dominieren zurzeit deutlich, da ML-Algorithmen typischerweise nicht für Latenz-kritische Anwendungen und damit als Eingangsgröße in Automatisierungsprogrammen genutzt werden. KI-Modelle, welche tiefe neuronale Netze nutzen, z.B. für die optische Inspektion nach einem Fertigungsschritt, benötigen deutlich länger als die Zykluszeit der Steuerung. Die trainierten KI-Modelle können auf der Steuerung ausgeführt werden und nach der Berechnung des Modells können beispielsweise fehlerhafte Werkstücke aus der Fertigung ausgeschleust oder nachbearbeitet werden. Hierfür stehen typischerweise aber mehrere Sekunden zur Verfügung [16]. Um Einschränkungen von KI- bzw. ML-Frameworks zu umgehen, setzt sich die Nutzung von ONNX als Beschreibungsstandard für KI-Anwendungen durch. ONNX beschreibt die mathematischen Operationen zwischen Eingangs- und Ausgangsdaten der ML-Modelle und entkoppelt somit die KI-Engineeringumgebung von der Ausführung des KI-Modells.

In gängigen Lösungen von Automatisierungsanbietern wird das Erlernen von KI-Anwendungen allein dem Können und der Kreativität des Automatisierungsexperten oder des Data Scientists überlassen. Typischerweise fehlt dem SPS-Programmierer das nötige Wissen in Bezug auf Maschinelle Lernverfahren und die Erfahrung in Bezug auf die Nutzung von KI-Engineering-Umgebungen. Der Data Scientist erstellt typischerweise ohne Domänenwissen in Bezug auf Automatisierungstechnik Modelle rein anhand der aufgezeichneten Daten [17].

Dies macht die Notwendigkeit deutlich, dass neben der reinen Integration von KI-/ML-Anwendungen in die Automatisierung eine Unterstützung notwendig ist, um effizient ML-Modelle zu erstellen, auf eine Steuerung zu integrieren und anzuwenden. Daher sind Assistenzfunktionen ein essenzieller Teil einer KI-basierten Automation, da diese auf unterschiedlichste Weise den Automatisierungsingenieur über den KI-Anwendungs-Lebenszyklus unterstützen können.

#### IV. KONZEPT

Nachdem der Stand der Technik mit Blick auf die definierten Anforderungen an eine KI-integrierte Engineering-Plattform vorgestellt wurde, wird in diesem Abschnitt konzeptionell die geplante Umsetzung der Plattform im Projekt EKI vorgestellt.

Um den Zugriff auf die Steuerung gemäß der ersten Anforderung zu gewährleisten, wird im Rahmen des Projektes PLC Open und dessen Erweiterungen genutzt, da es sich um eine hersteller- und produktneutrale Schnittstelle handelt. Für das EKI Projekt sind beispielsweise PLCopen Communication

und PLCopen XML Exchange relevant. PLC Communications standardisiert die Nutzung von OPC UA in Kombination mit PLCopen und PLCopen XML Exchange erlaubt es, Programme, Bibliotheken und Projekte zwischen Entwicklungsumgebungen auszutauschen.

Anforderung zwei fordert eine kurzfristige Einsatzbereitschaft der Plattform auf dem Gerät des Automatisierungsingenieurs. Der Großteil der marktrelevanten Anbieter macht es notwendig, dass der Automatisierungsingenieur eine proprietäre Software auf dem Endgerät installiert, was insbesondere bei der Versionierung zu Schwierigkeiten führen kann. Eine Alternative hierzu ist eine web-basierte Engineering-Plattform, bei der der Zugriff über den Browser erfolgt. Die Engineering-Plattform kann dabei sowohl auf der Steuerung als auch in der Cloud gehostet werden. Dadurch muss der Bediener lediglich die IP-Adresse oder die Cloud URL im Browser angeben und kann so ohne Installationsaufwand oder speziellen Lizenzanforderungen darauf zugreifen.

Als Kommunikationsschnittstelle wurde sich für verschiedene Standards entschieden und somit auch Anforderung 3 erfüllt. Dies betrifft einerseits http(s), Modbus TCP und OPC UA. Allerdings wird in dem Projekt OPC UA priorisiert verwendet, da hier neben der Möglichkeit von verschlüsselter Kommunikation auch ein semantisch beschriebenes Informationsmodell zur Verfügung gestellt wird, was für das Erstellen und Anwenden von KI-Anwendungen ein entscheidender Vorteil ist. Verbreitete Kommunikationsprotokolle wie Modbus TCP werden vor allem für die Datenakquise von Energiemessgeräten verwendet. Energiemessgeräte in unterschiedlicher Messgenauigkeit sind eine häufig genutzte Datenquelle für KI-Anwendungen, da sich Anlagenzustände häufig aus diesen Daten herleiten lassen.

Um KI Anwendung über die Engineering-Plattform zu konfigurieren und diese in die automatisierte Anlage zu integrieren, muss es möglich sein, containerisierte Software zu erstellen und anzuwenden (deployen). Industriell weit verbreitet ist hierbei Docker. Bei dieser Vorgehensweise ist es allerdings wünschenswert, dass hardwareseitig eine Linux Anwendung zur Verfügung steht. Dies bezieht sich insbesondere auf das Deployment von bereits trainierten und containerisierten KI-Anwendungen, welche auf die Rechenleistung der vorgesehene Steuerung/Hardware abgestimmt sind.

Handelt es sich um Modelle, die komplex, beziehungsweise rechenintensiv sind, werden die Modelle mittels einer virtuellen Maschine in einer frei wählbaren Cloud trainiert. In der Engineering-Plattform soll eine Anbindung an Online Rechenressourcen über NodeRed möglich sein. Es werden hierbei die zur Verfügung gestellten Nodes der jeweiligen Anbieter genutzt. Mithilfe dieses Werkzeugs lassen sich auch Datenflüsse zu den unterschiedlichen Cloudanbietern realisieren. Die dort vorhandenen KI-Engineering-Umgebungen bieten mächtige Werkzeuge, welche die Anforderung fünf bis sieben abdecken können, also die Datenspeicherung, das Training und den Export des Modells.

Die letzte Anforderung betrifft die Erweiterbarkeit der Engineering-Plattform. Hierzu ist ein AddOn Manager ge-

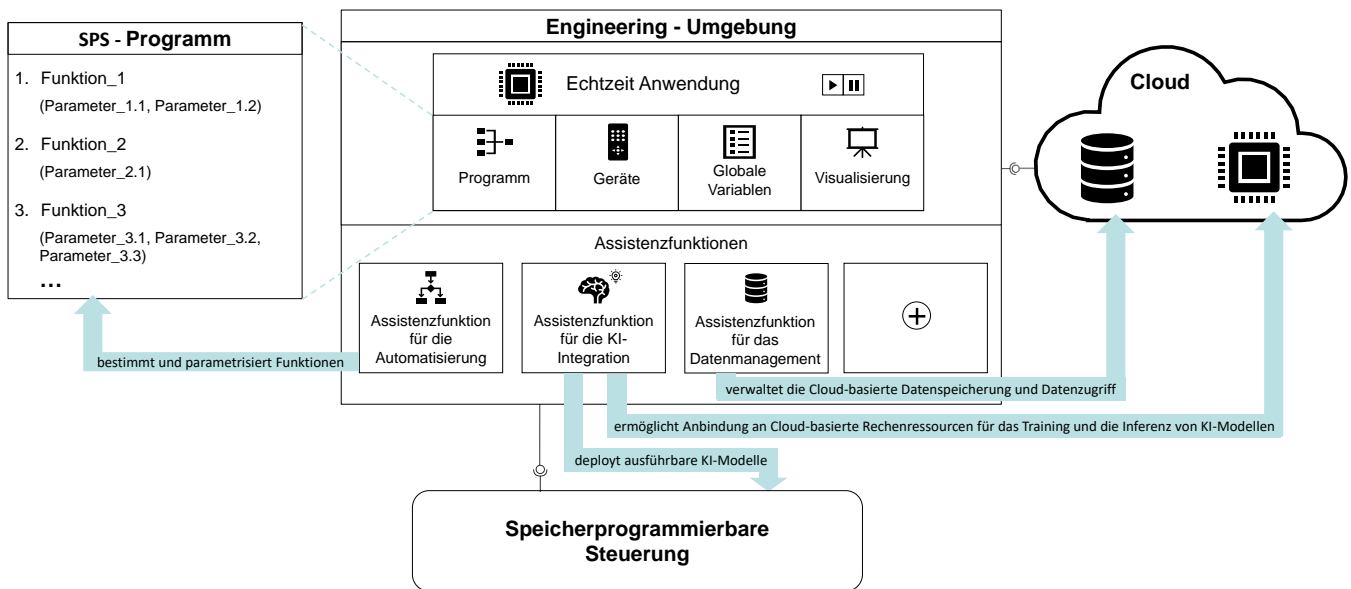


ABBILDUNG 1. DARSTELLUNG DER GEPLANTEN ENGINEERING-PLATTFORM SOWIE DER FUNKTIONSWEISE AUSGEWÄHLTER ASSISTENZFUNKTIONEN

plant, der es dem Anwender ermöglicht, über einen Docker Container seine neue Anwendung oder Assistenzfunktion in der Engineering-Plattform zu integrieren. Die Erweiterung der Plattform durch die Anwender innerhalb einer eigenen Community ist ausdrücklich erwünscht. Daher wird diese auch zum Ende des Projektes unter einer MIT-Lizenz veröffentlicht und so der Community zur Verfügung gestellt.

Eine Darstellung des Konzepts der geplanten Engineering-Plattform sowie der Funktionsweise der geplanten Assistenzfunktionen kann der ABBILDUNG 1 entnommen werden. Im Zentrum steht die Engineering-Plattform von Weidmüller Interface GmbH & Co. KG, die einerseits aus einer Echtzeitumgebung, inklusive SPS-Programm, Geräteverwaltung, Variablen- und Visualisierung, und andererseits aus den Assistenzfunktionen/AddOns besteht. Die Plattform ermöglicht eine Anbindung an cloudbasierte Rechenressourcen, sowie die speicherprogrammierbare Steuerung der Anlage.

## V. AUSBLICK

Angesichts der zunehmenden Bedeutung von KI in der Industrie wurde im Rahmen des Beitrages analysiert, welche besonderen Anforderungen diese Entwicklung an eine Engineering-Plattform stellt, die für die Integration von KI-Applikationen im Engineering geeignet ist. Außerdem wurde ermittelt, welche Voraussetzungen eine Engineering-Plattform erfüllen muss, damit sie ebenfalls um KI-basierte Assistenzfunktionen erweitert werden kann. Basierend auf den definierten Anforderungen und einem Überblick über den Stand der Technik in diesem Bereich, wurde das Konzept für eine offene, erweiterbare und web-basierte Engineering-Umgebung vorgestellt. Gemäß dem Konzept wird im Rahmen des Forschungsprojektes EKI von der Firma Weidmüller Interfaces GmbH & Co. KG eine Engineering-Plattform entwickelt und

anschließend anhand einer Realanlage validiert. Außerdem werden im weiteren Verlauf des Projektes von der Helmut Schmidt Universität KI-basierte Assistenzfunktionen entwickelt, die den Anwender sowohl im Zuge des Engineerings an sich, als auch bei der Auswahl von KI-Applikationen in Abhängigkeit des Anwendungsfalls, unterstützen.

## DANKSAGUNG

Die Autoren bedanken sich für die Förderung bei dtcc.bw – Zentrum für Digitalisierungs- und Technologieforschung der Bundeswehr [Projekt EKI].

## LITERATUR

- [1] H. Kagermann, J. Helbig, A. Hellinger, and W. Wahlster, *Recommendations for implementing the strategic initiative INDUSTRIE 4.0: Securing the future of German manufacturing industry; final report of the Industrie 4.0 Working Group*. Forschungsunion, 2013.
- [2] L. Monostori, "Cyber-physical Production Systems: Roots, Expectations and R&D Challenges," *Procedia CIRP*, vol. 17, pp. 9–13, 2014.
- [3] G. Fenza, M. Gallo, V. Loia, D. Marino, F. Orciuoli, and A. Volpe, "Semantic CPPS in Industry 4.0," in *Advanced Information Networking and Applications*, ser. Springer eBook Collection, L. Barolli, F. Amato, F. Moscato, T. Enokido, and M. Takizawa, Eds. Cham: Springer International Publishing and Imprint Springer, 2020, vol. 1151, pp. 1057–1068.
- [4] A. Köcher, C. Hildebrandt, L. M. Vieira da Silva, and A. Fay, "A Formal Capability and Skill Model for Use in Plug and Produce Scenarios," in *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, vol. 1, 2020, pp. 1663–1670, ISSN: 1946-0759.
- [5] A. Köcher, R. Heesch, N. Widulle, A. Nordhausen, J. Putzke, A. Windmann, and O. Niggemann, "A Research Agenda for AI Planning in the Field of Flexible Production Systems," in *IEEE International Conference on Industrial Cyber-Physical Systems (ICPS)*, 2022.
- [6] V. Vyatkin, "Software engineering in industrial automation: State-of-the-art review," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 3, pp. 1234–1249, 2013.
- [7] W. Bolton, *Programmable logic controllers*. Newnes, 2015.
- [8] BMWi, "Auf Künstliche Intelligenz kommt es an: Beitrag von KI zur Innovationsleistung und Performance der deutschen Wirtschaft," 2020.

- [9] A. Mockenhaupt, "Digitalisierung und KI in der Produktion," in *Digitalisierung und Künstliche Intelligenz in der Produktion*. Springer, 2021, pp. 227–269.
- [10] M. Kropik, "Produktion," in *Produktionsleitsysteme für die Automobilindustrie*. Springer, 2021, pp. 129–235.
- [11] T. Bauernhansl, M. Ten Hompel, and B. Vogel-Heuser, *Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung-Technologien-Migration*. Springer, 2014.
- [12] M. Marcos, E. Estevez, F. Perez, and E. Van Der Wal, "XML exchange of control programs," *IEEE Industrial Electronics Magazine*, vol. 3, no. 4, pp. 32–35, 2009.
- [13] M. Dutta, K. K. Sethi, and A. Khatri, "Web based integrated development environment," *International Journal of Innovative Technology and Exploring Engineering*, vol. 3, no. 10, pp. 56–60, 2014.
- [14] T. Żabiński, T. Maczka, and J. Kluska, "Industrial platform for rapid prototyping of intelligent diagnostic systems," in *Polish Control Conference*. Springer, 2017, pp. 712–721.
- [15] M. Blackstock and R. Lea, "Toward a distributed data flow platform for the web of things (distributed node-red)," in *Proceedings of the 5th International Workshop on Web of Things*, 2014, pp. 34–39.
- [16] Beckhoff Automation GmbH & Co. KG, Hülshorstweg 20 33415 Verl, "Machine Learning," 24.08.2022. [Online]. Available: <https://www.beckhoff.com/de-de/produkte/automation/twincat-3-machine-learning/>
- [17] M. Lukas, D. Stock, and A. Csiszar, "FabOS: Towards an open, distributed, real-time-capable, and secure operating system for production," *Procedia CIRP*, vol. 104, pp. 962–967, 2021.