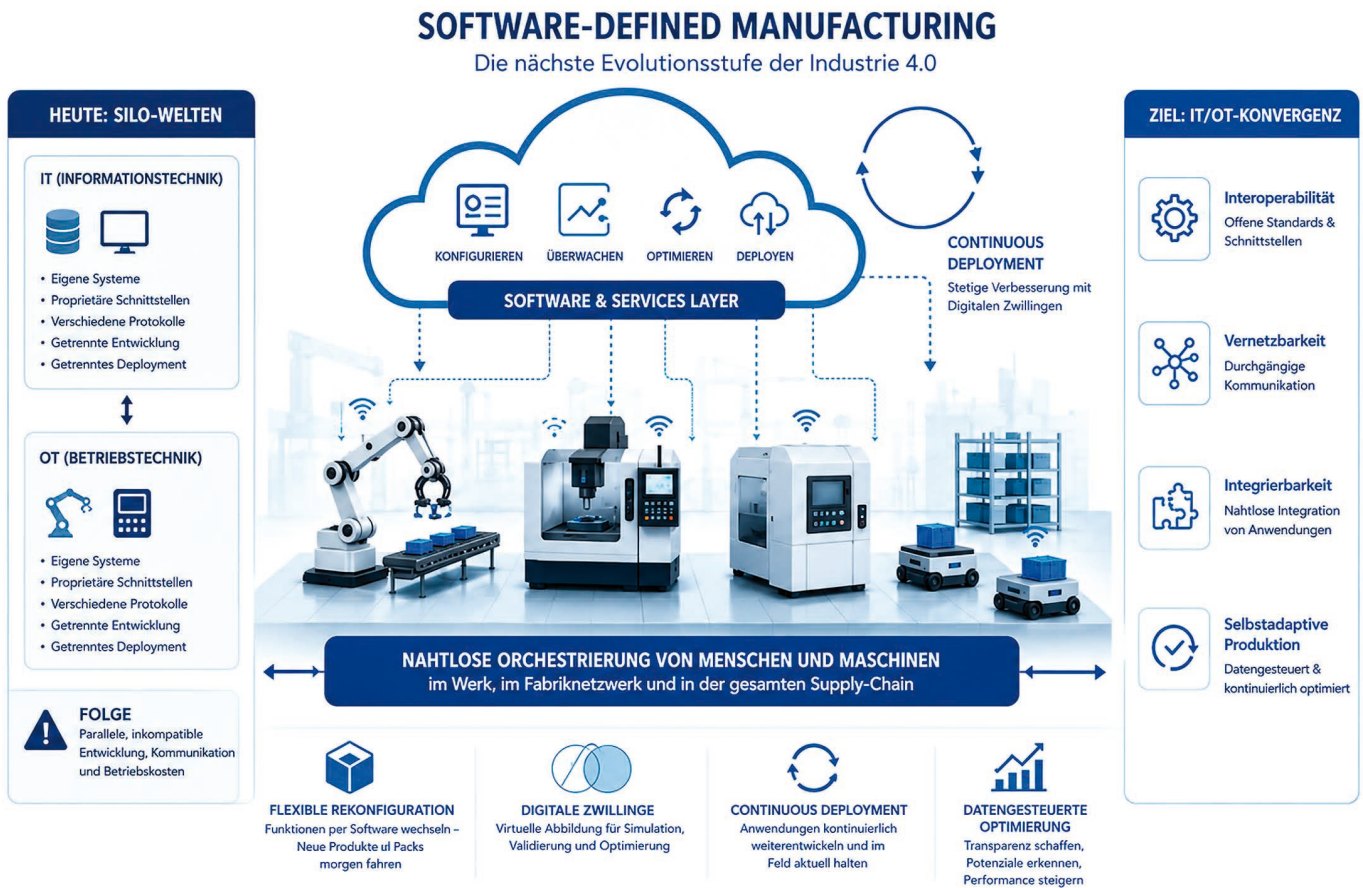


Flexible Steuerungsarchitekturen in der Praxis

Architekturansätze für IT-OT-konvergentes Softwareengineering in der Automatisierungstechnik



Autor:
Moritz Walker, M.Sc.
Wissenschaftlicher Mitarbeiter
„Echtzeitkommunikation
und Steuerungshardware“
Universität Stuttgart
Institut für Steuerungstechnik
der Werkzeugmaschinen
und Fertigungseinrichtungen (ISW)
<http://www.isw.uni-stuttgart.de>

Software-defined Manufacturing (siehe Aufmacherbild) gilt als die nächste Evolutionsstufe von Industrie 4.0. Dabei wird das Prinzip des Software-defined Everything (SDX) auf die Produktionstechnik übertragen. SDX ist ein branchenübergreifender Ansatz, bei dem die Steuerung physischer Geräte und Systeme von starrer Hardware auf flexibel und zentral programmierbare Software verlagert wird. Anstatt manuelle Anpassungen an Geräten vorzunehmen, werden Funktionen per Software konfiguriert, automatisiert und aus der Ferne aktualisiert. Wird dieses Konzept auf die Produktionstechnik übertragen, werden Herstellungsprozesse und Produktionssysteme vollständig durch Softwareanwendungen und Services konfiguriert, überwacht und optimiert. In seiner vollen Ausbaustufe ermöglicht das SDM somit die nahtlose Orchestrierung von Menschen und Maschinen innerhalb der Fabrik, im Fabriknetzwerk und entlang der gesamten Supply Chain.

Rekonfiguration des Produktionssystems

Vor dem Hintergrund der Realisierung dieser Vision ist dem klassischen Ziel der Industrie 4.0, also der vernetzten, flexiblen und datengetriebenen, selbstadaptiven Produktion, noch heute hohe Bedeutung zuzumessen. Das heißt, dass durch die Entkopplung von Hardware und Software des Produktionssystems eine Rekonfiguration des Produktionssystems möglich wird, indem die Software anforderungsgetrieben rekonfiguriert wird. So kann bspw. derselbe Roboter heute mit einer hochdynamischen Robotersteuerung ausgestattet Pick-and-Place-Aufgaben erfüllen und morgen Fräsen, indem die Robotersteuerung durch eine CNC-Steuerung mit besserer Bahngenauigkeit ersetzt wird.

Continuous Deployment

Weiterhin soll mittels Continuous Deployment in Verbindung mit Digitalen Zwillingen

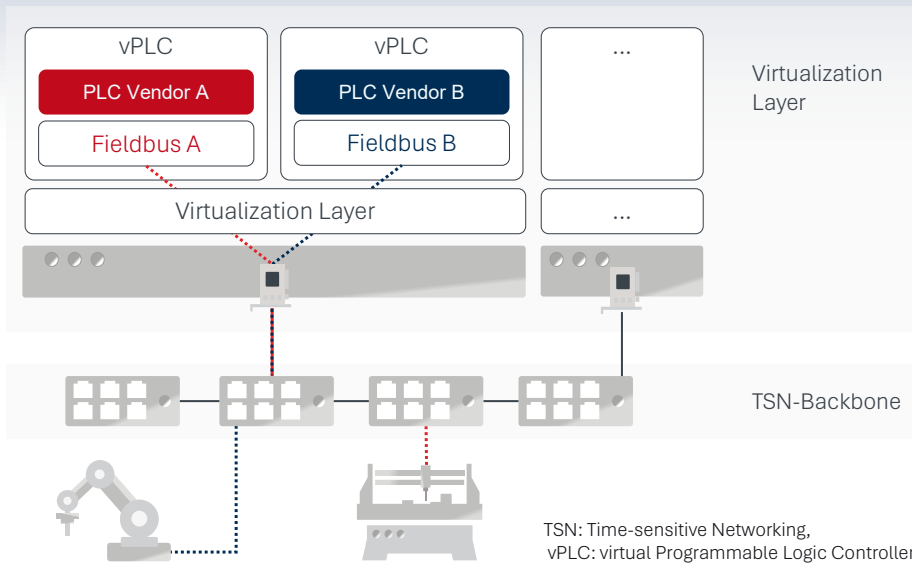


Bild 1: IT/OT-konvergentes Software-Engineering

eine stetige Verbesserung herbeigeführt werden, indem Anwendungen durchgehend weiterentwickelt und im Feld aktuell gehalten werden. Da sich das Produktionssystem in dieser Weise kontinuierlich weiterentwickelt, ist eine Interoperabilität, Vernetzbarkeit und Integrierbarkeit der unterschiedlichen Anwendungen unabdingbar. Silo-Technologien und proprietäre Schnittstellen erschweren die Umsetzung der vernetzten und selbstadaptiven Produktion.

Silo-Technologien

Die Problematik der Silo-Technologien manifestiert sich insbesondere im Bruch zwischen der OT (Betriebstechnik) und IT (Informationstechnik). Heute sind beide Welten strikt getrennt, werden unterschiedlich programmiert, mit verschiedenen Protokollen vernetzt und mit unterschiedlichen Methoden in Betrieb genommen (deployt). IT- und OT-Systeme sind unter Umständen nicht miteinander kompatibel und nicht integriert. Die Folge sind parallele und inkompatible Entwicklungs-, Kommunikations- und Betriebsarten. Ansätze für SDM aus Forschung und Praxis setzen auf IT/OT-Konvergenz, um die beschriebenen Probleme zu lösen und so eine kontinuierliche Weiterentwicklung des Produktionssystems zu ermöglichen.

IT-OT-Konvergenz

IT/OT-Konvergenz wurde im Artikel „Kern-technologien flexibler Steuerungsarchitekturen“ Untertitel: Offene Steuerungsarchitekturen als Enabler für IT/OT-Konvergenz in der Vision von Industrie 4.0 (Autor: Rebekka Neumann) in Ausgabe 7/2026 [1] beleuchtet und wird daher im Folgenden nur so weit eingeführt, wie es zum Verständnis des Big-Pictures notwendig ist.

Die IT/OT-Konvergenz umfasst das Ziel, mittels Offenheit und Standardisierung Interoperabilität zwischen den Welten und den Entitäten innerhalb der Welten zu schaffen. Dies beruht insbesondere auf zwei Säulen, nämlich die Bereitstellung der Software auf einer einheitlichen Infrastruktur sowie eine einheitliche Infrastruktur zur Kommunikation. Erstere wird durch Virtualisierung realisiert. Während in der Vergangenheit

speicherprogrammierbare Steuerungen (englisch: Programmable Logic Controller, PLC) als Hardware-Box im Schaltschrank installiert wurden, sind diese heute zunehmend als virtuelle PLC (vPLC) verfügbar.

Mittels Methoden der Virtualisierung wie Containern und Virtuellen Maschinen (VMs) – eigentlich Entwicklungen aus der IT – wird es möglich, Steuerungen verschiedener Hersteller auf einer einheitlichen Infrastruktur zu betreiben. Wo früher also eine feste Zuordnung von PLC-Runtime zu Hardware vorgegeben war, ist heute flexibles Deployment möglich. Dies öffnet die Tür hin zu Ansätzen, bei denen die IT-Software und OT-Software virtualisiert auf dedizierten Compute-Clustern läuft, kontinuierlich up-to-date gehalten wird und bedarfsgerecht „per Knopfdruck“ einfach eine neue Steuerung eines beliebigen Herstellers in Betrieb genommen wird. Dieses Konzept konnte am ISW erfolgreich gezeigt werden, indem mehrere industrielle CNC-Kerne zeitgleich und von einem Edge-Node ausgehend mehrere Maschinen in der Fertigungshalle des ISW mit deterministischer Echtzeit steuern.

Dieses Szenario leitet zum zweiten Aspekt der IT/OT-Konvergenz über (Bild 1). Jede Maschinensteuerung muss mittels eines Feldbusses mit den Feldgeräten angebunden werden. Würde nun jede vPLC auf einem Rechnerknoten eine dedizierte Netzwerkkarte erhalten, würde dies unnötige Hardware- und Verkabelungsaufwände bedeuten. Abhilfe schafft hier Time-Sensitive Networking (TSN), welches es erlaubt, mehrere Echtzeit- und Nicht-Echtzeit-Datenströme über „dasselbe Kabel“ zeitgleich abzuwickeln. So erfolgt die Anbindung der Feldgeräte in der Fertigungshalle des ISW an die vPLCs über die Tunnelung verschiedener Protokolle wie UDP, EtherCAT, Profinet RT oder CC-Link über TSN: Als Teil des Ethernet-Standards ist TSN kein proprietäres Protokoll, sondern schließt perfekt die Lücke zwischen OT und IT, indem die Protokolle beider Welten über eine einheitliche und gemeinsame Netzwerkinfrastruktur umgesetzt werden können. Gemeinsam mit offenen, standardisierten Protokollen wie OPC UA, welche für IT und OT geeignet sind, kann so IT/OT-Konvergenz in der Produktion gezeigt werden. Während die Aspekte der Konvergenz der Bereitstellung durch Virtualisierung und der Kommunikation durch TSN bereits viel Beachtung geschenkt wurde, ist die Konvergenz des Software-Engineerings von nicht minder großer Bedeutung.

IT/OT-konvergentes Software-Engineering

Wie bei der Bereitstellung und bei der Kommunikation erfolgt das Engineering von Steuerungsanwendungen heute für IT- und OT-Software grundlegend unterschiedlich. Historisch bedingt waren Entwicklungsumgebungen für PLCs geschlossene Systeme, die eine Integration mit anderen Herstellern nur bedingt zulassen.

Im Gegensatz dazu wird IT-Software häufig mit offenen, (quasi-)standardisierten

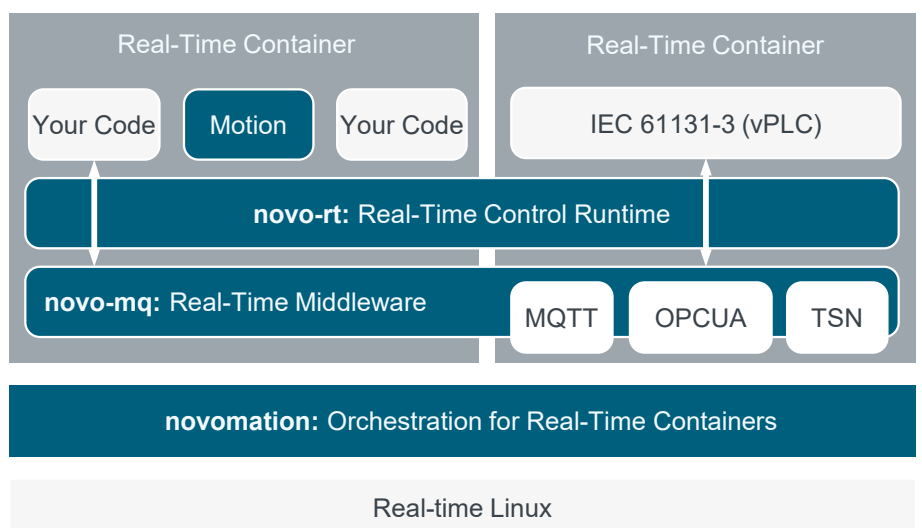


Bild 2: Stack des ISW für konvergentes Software-Engineering für die Automatisierungstechnik.

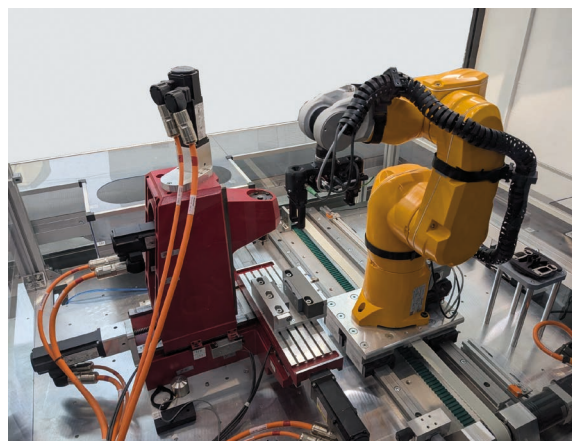
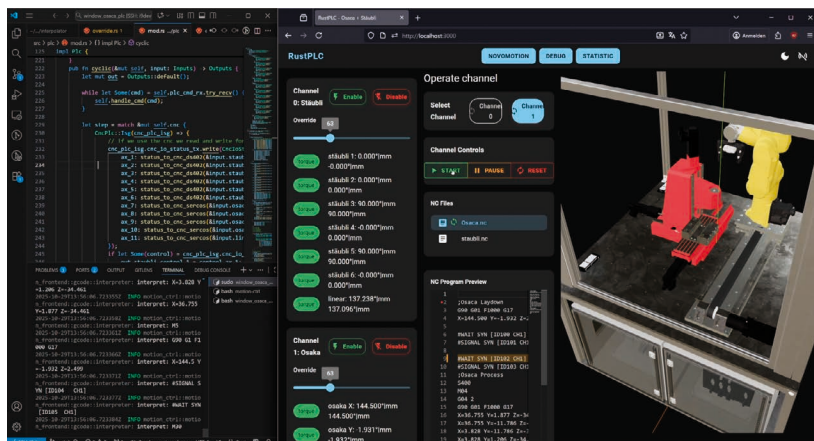


Bild 3: Fräs- und Roboterzelle (rechts) und gänzlich in Rust programmierte Motion, Steuerungslogik, Simulation und Visualisierung (links).

Programmiersprachen wie C, Rust oder C# und FOSS-Compilern und -Runtimes umgesetzt. Dies führt zu einem Bruch in der Entwicklung. Typische Werkzeuge aus der IT, wie CI/CD-Pipelines als Bestandteil der Source-Code-Verwaltung, automatisiertes Testing usw. können oft nur unter Zuhilfenahme von Spezialsoftware auf Steuerungsprojekte angewandt werden. Die jüngsten Entwicklungen im Bereich KI zeigen, dass Offenheit hier ein entscheidender Wettbewerbsvorteil sein kann, indem z. B. der Anwender selbständig neue Technologien wie KI-Agenten zur Quellcodeentwicklung integrieren können.

Offene Steuerungsarchitekturen

Bei technischen Neuerungen war stets eine Integration durch den PLC-Anbieter in die PLC-Runtime notwendig. Zwar unterstützen PLCs oft die Integration von z. B. C-Code, dieser kann allerdings nicht unabhängig von der PLC-Runtime kompiliert und ausgeführt werden. Glücklicherweise wandelt sich die Steuerungstechnik in diesem Zusammenhang hin zu offenen Steuerungsarchitekturen. Diese erlauben eine einfachere Integration von Software in das herstellereigene Ökosystems wie beispielweise den aktuellen Ansätzen etablierter Steuerungsanbieter zeigen. Grundsätzlich wird es so möglich, einzelne Module der Steuerung in individuell verschiedenen Technologien umzusetzen. So kann klassischer PLC-Code der IEC 61131-3 mit Hochsprachen integriert werden.

Use-Cases und Benefits von konvergentem Software-Engineering

Hier eröffnet sich allerdings eine weitere Gelegenheit: IT-Software und OT-Software, also Steuerungscode, kann mit einem einheitlichen Software-Engineering-Ansatz realisiert werden. Dies ist möglich, indem beide Welten in eine einheitliche Programmiersprache mit identischen Toolchains, Compilern und Runtimes umgesetzt werden. So zeigte das ISW auf der SPS 2023 eine vollständig in Rust programmierte Automatisierung einer Roboterzelle (Bild 3). Dabei wurden alle Komponenten, von der HMI bis hin zu Steuerungslogik und Motion-Control, mittels

desselben Rust-Compilers entwickelt. Der Ansatz reduziert Komplexität, da IT- und OT-Software mit denselben Methoden entwickelt werden. Dadurch sinkt der Integrationsaufwand, und Systeme lassen sich schneller testen und ausrollen. Weiterhin kann das System einfach in individuell testbare und bereitstellbare Module zerlegt werden, die nativ als Docker-Container umgesetzt werden können. Auch können Teams enger zusammenarbeiten, weil keine Sprach- oder Plattformgrenzen mehr bestehen. Auch Open-Source-Bibliotheken können nahtlos integriert werden.

Selbstverständlich bringt das konvergente Software-Engineering auch Herausforderungen mit sich. Da Steuerungslogik in Echtzeit ausgeführt wird und in Echtzeit kommunizieren muss, bedarf es eines Frameworks, welches diese Funktionen bereitstellt. Bild 3 zeigt die notwendigen Lösungsbausteine, wie sie am ISW entwickelt und verwendet werden.

Echtzeitfähiges Linux-System

Die Basis bildet ein echtzeitfähiges Linux-System. Alle Komponenten, sowohl vPLCs als auch eigene Echtzeit-Module in Hochsprachen, werden als Docker-Container mittels eines Orchestrators bereitgestellt. Dieser sorgt für die Allokation der für die Einhaltung der Echtzeit notwendigen Ressourcen, wie CPU und RAM. Die IT/OT-konvergente Kommunikation und somit der Datenaustausch zwischen Modulen und zwischen Modulen und Feldgeräten erfolgt durch eine eigens entwickelte, echtzeitfähige Middleware, welche TSN über Gerätegrenzen hinweg unterstützt. Die echtzeitfähige Einplanung und Ausführung der Module wird durch die Real Time Control Runtime umgesetzt. So wird es möglich, alle Komponenten, ob Steuerungslogik

oder HMI, einheitlich mit derselben Toolchain in einer frei wählbaren Entwicklungsumgebung zu programmieren. Dabei können alle gängigen IT-Tools, von Unit-Tests bis hin zu CI/CD-Pipelines, für automatische Build- und Deployment-Pipelines genutzt werden.

Zusammenfassung

IT/OT-konvergentes Software-Engineering ist ein zentraler Enabler für Software-Defined Manufacturing und damit für hochflexible, rekonfigurierbare Produktionssysteme. Durch Virtualisierung, standardisierte Kommunikation auf Basis von TSN und einheitliche Software-Toolchains kann die klassische Trennung von IT und OT überwunden werden. Dadurch entstehen integrierte, schneller entwickel- und deploybare Systeme mit geringerer Komplexität und höherer Flexibilität. Gleichzeitig bleiben Echtzeitfähigkeit, Determinismus und Systemintegration zentrale technische Herausforderungen, welche durch eine geeignete Architektur aber gelöst werden können. Insgesamt bildet die Konvergenz die Grundlage für softwaregetriebene, adaptive Produktionssysteme der nächsten Generation.

Wer schreibt:

Moritz Walker forscht im Bereich Echtzeitkommunikation, Steuerungshardware und -software am Institut für Steuerungstechnik (ISW). Dort hat er seine Doktorarbeit über die Orchestrierung virtualisierter industrieller Steuerungen (vPLCs) in Edge-Computing-Umgebungen in Verbindung mit einem Time-Sensitive-Networking (TSN)-Backbone verfasst. Mit der Ausgründung novomation unterstützt er Unternehmen wie Maschinen- und Anlagenbauer bei der Konzeption und Umsetzung von zukunfts-sicheren Steuerungsarchitekturen. ◀

Link

[1] „Kerntechnologien flexibler Steuerungsarchitekturen“
Untertitel: Offene Steuerungsarchitekturen als Enabler für IT/OT- Konvergenz in der Vision von Industrie 4.0 (Autor: Rebekka Neumann)“ in Ausgabe 7/2026
Link: <https://webkiosk.epaper-kiosk.beam-verlag.de/7-2026/71230667> ab Seite 78.