

# Engineering und Betrieb Smarter Komponenten in IoT-Netzwerken für die Automatisierung der Produktion

Prof. Dr.-Ing. Michael Weyrich; Andreas Zeller, M. Sc.; Jan-Philipp Schmidt, M. Sc.; Dipl.-Ing. Alexander Faul; Philipp Marks, M. Sc.

Institut für Automatisierungstechnik und Softwaresysteme, Universität Stuttgart, Deutschland, [ias@ias.uni-stuttgart.de](mailto:ias@ias.uni-stuttgart.de)

*Die Verknüpfung Smarter Komponenten zu dezentral koordinierten IoT-Netzwerken bietet neue Chancen für die Produktionsautomatisierung. Allerdings ergeben sich neue Herausforderungen und ein Bedarf an Konzepten für das durchgängige Engineering, die Integration bestehender Komponenten und derer Absicherung für einen verlässlichen Betrieb.*

## 1 Motivation

Die Vernetzung Smarter Komponenten eröffnet zahlreiche Möglichkeiten der Informationsgewinnung, -auswertung und Prozessorganisation. Große Potenziale der Vernetzung von Dingen und Diensten sehen Experten im Bereich der industriellen Automatisierungstechnik, um eine neue Form der Industrialisierung zu erschließen [Aca16]. Dabei bietet die Vernetzung Smarter Komponenten zahlreiche Chancen.

Die durchgängige IT-Vernetzung von Smarten Komponenten erlaubt eine flexible Koordination der Produktionsabläufe. So können durch automatisierte Koordination von Produktionsaufträgen Kleinaufträge effizient produziert, Spitzenlasten durch gezielte Verteilung der Aufträge auf die Ressourcen abgefedert, Produktionsanlagen optimal ausgelastet und Produkte ohne lange Vorlaufzeit in den Produktionsprozess integriert werden.

Dadurch ergeben sich für den Produzenten zahlreiche Marktvorteile. Kleine Stückzahlen können aufgrund eines hohen Automatisierungsgrads kostengünstig gefertigt werden, Maschinenleerlauf wird vermieden und Produkte können ohne lange Vorlaufzeit am Markt platziert werden, bzw. die Produktionsmenge der Nachfrage entsprechend angepasst werden [Aca16].

Dabei besteht noch großer Handlungsbedarf, um die zukünftige Wettbewerbsfähigkeit des Standorts Deutschland zu gewährleisten [Aca16].

Die Herausforderungen sind vielschichtig: Aufgrund des Zusammenspiels unterschiedlicher Gewerke und Bereiche, wie der Logistik, der Automatisierung einzelner Prozesse, der Organisation von betrieblichen Abläufen sowie des vorgelagerten Engineering-Bereichs, ist von verschiedensten Anforderungen der Teilnehmer an die Produktionsautomatisierung auszugehen.

Diese Heterogenität, welche sich in zahlreichen Kommunikationsprotokollen und Semantiken ausdrückt, wird durch ein großes Produktangebot von Automatisierungskomponenten verstärkt. Daher werden Referenzarchitekturen benötigt, die eine möglichst einfache Vernetzung unterschiedlichster Komponenten erlauben. Für die bereichsübergreifende Interoperabilität der Produktionsautomatisierung sind weltweite Standardisierungsbestrebungen im Gang, die eine Referenzarchitektur für das Internet-of-Things (IoT) anstreben. [P40IIC].

Dabei stellt die industrielle Automatisierungstechnik hohe Anforderungen an Verfügbarkeit, Zuverlässigkeit sowie die Sicherheit der vernetzten Systeme.

In diesem Beitrag werden Verfahren zum durchgängigen Engineering Smarter Komponenten sowie zur Integration bestehender Komponenten in das Internet der Dinge und Dienste dargestellt. Die Integration bestehender Komponenten ins Internet der Dinge stellt aufgrund der langen Betriebsdauer von Produktionsanlagen eine Schlüsselrolle zur Akzeptanz von Industrie 4.0 dar. Zur zuverlässigen und handhabbaren Realisierung solcher Smarter Komponenten bedarf es neuer Konzepte für das Engineering und den Betrieb. Zukünftige IoT-Architekturen erlauben im Betrieb die Ad-hoc-Vernetzung von Smarten Komponenten, um das Produktionssystem zu rekonfigurieren. Die so gewonnene Flexibilität stellt allerdings hohe Anforderungen an das Testmanagement zur Absicherung der Produktqualität und Verfügbarkeit der Gesamtanlage.

Die Konzepte, welche in Bild 1 eingeordnet sind, werden dazu genutzt, industrielle Automatisierungssysteme möglichst modular aus Smarten Komponenten aufzubauen, um diese dann zu Produktionsautomatisierungssystemen zusammenzustellen.

## 2 Smarte Komponenten in der Anlagenautomatisierung

In einer Fabrik der Zukunft werden Smarte Komponenten den Produktionsablauf selbstständig untereinander koordinieren können. Eine lose Kopplung über eine serviceorientierte Architektur verspricht zukünftig ein hohes Maß an Flexibilität bei der Konfiguration der IT-Systeme der Produktion.

Der Wunsch nach Flexibilität treibt die Entwicklung von Produktionsautomatisierungssystemen seit Jahrzehnten voran. Zunächst ersetzen Speicherprogrammierbare Steuerungen festverdrahtete, verbindungsprogrammierte Steuerungen und erlauben dadurch eine Anpassung des Produktionsprozesses durch eine Änderung des Steuerungsprogrammes [Rei09], [Str98]. Durch den Trend hin zu kundenindividuellen Produktvarianten in kleiner Losgröße, steigen die Anforderungen an die Flexibilität von Produktionsautomatisierungssystemen weiter. Flexible Produktionssysteme, wie beispielsweise CNC-Maschinen, genügen im Ansatz dieser Forderung, leiden aber darunter, dass der Funktionsumfang des Systems bereits durch das Engineering vollumfänglich bestimmt ist [BuMa08, Elm05]. Diese Einschränkung ist für alle zentral gesteuerten Systeme gültig.

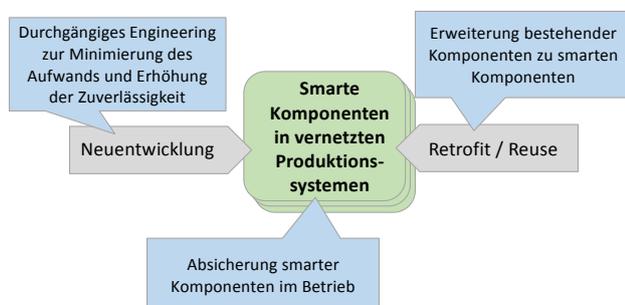
Der Ansatz der Holonic Manufacturing Systems (HMS) adressiert diese Limitation durch die Dezentralisierung der

Steuerung. HMS wurden entwickelt, um eine hohe und vorhersagbare Produktionsleistung zu erbringen und dabei robust auf Störungen und flexibel auf Änderungen reagieren zu können [BaCh06, BrNo03]. Ein HMS besteht aus autonomen Einheiten, die in einer verteilten, doch hierarchisch strukturierten Gruppe kooperieren, um das Gesamtziel des Produktionssystems zu erreichen.

Durch aktuelle Fortschritte in der Informations- und Kommunikationstechnik, insbesondere der Cloud- und Internet-technologie und der sich daraus ergebenden Vernetzungsmöglichkeiten, kann der Grundgedanke der verteilten Steuerung mittlerweile sinnvoll umgesetzt werden. Um zentrale und dezentrale Steuerungsansätze miteinander zu vergleichen, können Kriterien wie Rekonfigurierbarkeit, Robustheit, Durchsatz und Vorhersagbarkeit etc. herangezogen werden [TPBS+13]. Dabei stehen diese Faktoren teilweise auch in negativer Korrelation zueinander. Smarte Komponenten folgen dem dezentralen Steuerungsansatz und ermöglichen dadurch das Maß an Flexibilität, das im Rahmen der Industrie 4.0-Diskussion häufig gefordert wird [DWHB15].

Durch den Wandel hin zu verteilten Steuerungen ändert sich die Entwurfsmethodik von Steuerungssoftware von einer hierarchischen Kompositionsstruktur mit zentralen Steuerungssystemen, wie der SPS, hin zur Kopplung von verteilten Steuerungen aus einzelnen Modulen. Dies führt dazu, dass Abhängigkeiten zwischen den Modulen dann von den Modulen selbst verwaltet werden müssen statt von einer übergeordneten Instanz. Damit die Module sich selbst verwalten und in ein Netzwerk integrieren können müssen die Komponenten „smarter“ werden, d.h. Funktionen wie Selbstanpassung oder Autoadaptation realisieren [WES16]. Dies führt zu einer deutlichen Komplexitätserhöhung innerhalb der Komponenten, die diese Komplexität allerdings kapseln, was letztlich zu einer Vereinfachung führt bei der Integration führt.

Um die Komplexität verteilter IoT-Systeme auch zukünftig zu beherrschen beschäftigt sich das Institut für Automatisierungstechnik und Softwaresysteme der Universität Stuttgart im Feld der Produktionsautomatisierung mit dem Engineering verteilter Automatisierungssysteme, der Integration und Interoperabilität heterogener Systemwelten, der Wissensverarbeitung dezentraler Systeme und der Verifikation und Validierung von automatisierten -Systemen im Bereich der Anlagenautomatisierung.

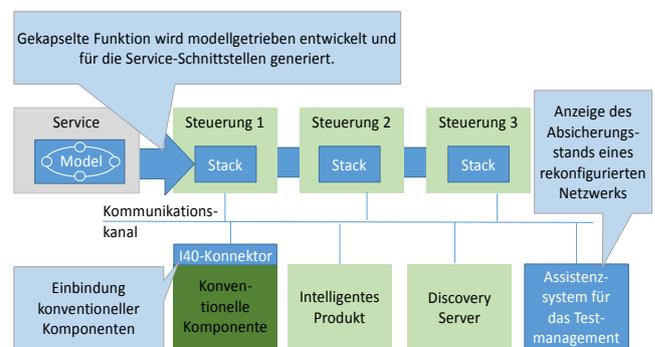


**Bild 1:** Forschungsaspekte bei der Neuentwicklung und Wiederverwendung Smarter Komponenten

### 3 Konzepte

Serviceorientierte Architekturen bieten ein hohes Maß an Flexibilität, weshalb Sie auch als das Kommunikationsparadigma der Industrie 4.0 gehandelt werden [DIN91345]. Dabei werden Funktionalitäten als Services angeboten, die über semantisch definierte Schnittstellen abgerufen werden können. Diese sogenannte „lose Kopplung“ zwischen den Komponenten ermöglicht diese Flexibilität. Serviceorientierte Architekturen schaffen aufgrund der verteilten Architekturstruktur aber auch eine Reihe neuer Herausforderungen und Rahmenbedingungen, welchen Konzepte um das Engineering und dem Betrieb Rechnung tragen müssen.

Im Folgenden werden die Forschungen zur Systemarchitektur, der Verbindung unterschiedlicher Systemwelten und dem Test von IoT-Systemen in der Automatisierungstechnik vorgestellt und eingeordnet. Eine Einordnung der Konzepte ist in Bild 2 dargestellt. Die Konzepte sind dabei durch blaue Sprechblasen gekennzeichnet.



**Bild 2:** Einordnung der vorgestellten Konzepte

### Durchgängige Engineering durch modellgetriebene Entwicklung

Als Vorbild für die Entwicklung von verteilten Kompositionsstrukturen bietet sich die Automobilindustrie als Vorreiter an. Dort ist die Referenzarchitektur AUTOSAR etabliert und ermöglicht eine hohe Zuverlässigkeit und Sicherheit durch den Einsatz von Wiederverwendungskonzepten [Hein]. Gleichzeitig wird durch die standardisierten Schnittstellen ein modellgetriebener Entwicklungsprozess ermöglicht, der eine graphische Modellierung und Codegenerierung erlaubt. So können mit geringem Aufwand verteilte Steuerungen entwickelt werden.



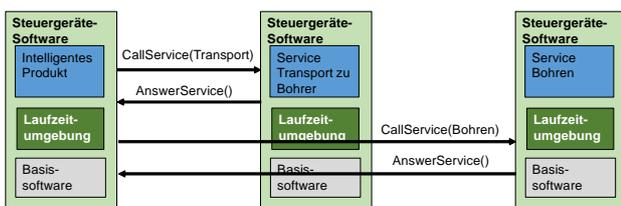
**Bild 3:** Modellgetriebener Entwicklungsprozess

AUTOSAR stellt als Architektur für die Automobilindustrie allerdings keine Konzepte für Flexibilität und Wandelbarkeit zur Verfügung, da diese Anforderungen im Auto-

mobil nicht gegeben sind. Steuergerätenetze in Fahrzeugen werden nach der Systementwicklung im Feld nicht mehr wesentlich verändert.

Im Gegensatz dazu fordert Industrie 4.0 eine flexible Koordinationsstruktur für Steuerungssysteme, um die oben genannten Anforderungen zu erfüllen. Hierzu kommen Service-orientierte Architekturen (SOA) zum Einsatz. Um die Vorteile von AUTOSAR in der Automatisierungstechnik nutzbar zu machen, wurde eine Softwarearchitektur entworfen [ScWe15]. Die Grundstruktur der Softwarearchitektur entspricht dabei AUTOSAR. Sie enthält die Basissoftware mit grundlegenden Funktionen, eine Laufzeitumgebung mit standardisierten Schnittstellen und die Applikationsschicht mit der Steuerungslogik. Die so entworfene Architektur unterstützt durch die definierten Schnittstellen eine modellgetriebene Entwicklung. Dabei wurden die Schnittstellen der Laufzeitumgebung an das Koordinationskonzept der SOA angepasst. Eine generierte Applikation, kann also in einen Server generiert werden, der die automatisierte Funktion als Service zur Verfügung stellt. Neben den automatisierten Teilfunktionen, die als Services im Netzwerk zur Verfügung gestellt werden sind weitere Komponenten notwendig. Zunächst müssen die Services zielgerichtet abgerufen werden. Dem Paradigma „das Produkt findet seinen Weg durch die Fertigung“ folgend, kommt es zu einem Abruf dieser Services durch das Produkt bzw. seine Vorstufen. Dabei kennt dieses Intelligente Produkt seinen Bauplan und damit die Schritte und Abhängigkeiten die in der Produktion notwendig sind. Mit diesem Wissen können die Services, die den Ablauf eines Produktionsschrittes anbieten, durch eine dem entstehenden Produkt zugeordnete Softwarekomponente, welche das Intelligente Produkt verwaltet, aufgerufen werden.

Eine weitere notwendige Softwarekomponente ist der Discovery Server. Dieser stellt ein Verzeichnis dar, in dem die Services registriert werden. Er ist selbst nicht am Produktionsprozess beteiligt, stellt aber dem intelligenten Produkt die notwendigen Informationen zur Verfügung, um die benötigten Services zu finden.



**Bild 4:** Architektur der Steuergeräte-Software

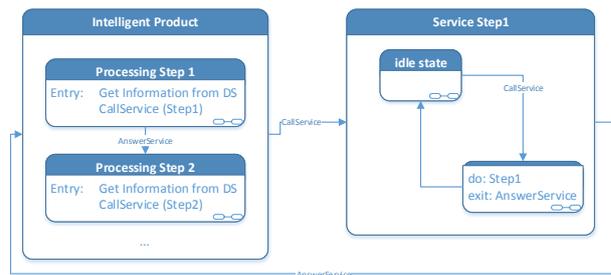
Grundlegende Softwaremodule, wie IO, Kommunikation und Betriebssystem werden wiederverwendbar in der Basissoftware realisiert. Die erstellte Laufzeitumgebung setzt eine Client-Server-Kommunikation um, welche die Grundlage für eine service-orientierte Architektur (SOA) darstellt [ScZe16].

Um eine bessere Durchgängigkeit im Entwicklungsprozess zu erreichen, können die Services mit geeigneten Modellierungstools als Zustandsautomaten modelliert werden, wozu Matlab Simulink zur Implementierung genutzt wird. In der Regel werden funktionale Modelle im Rahmen des Feinentwurfs vor der Implementierungsphase erzeugt. Die

standardisierten Schnittstellen ermöglichen eine automatische Generierung des Codes für die Services. Eingangsgrößen für die Zustandsautomaten sind die Sensoren, die ein Prozessabbild ermöglichen, und eingehende Nachrichten. Startbedingung für den Ablauf des Zustandsautomaten ist der Aufruf des Services. Dieser wird im Zustandsautomaten als Transition mit der Bedingung der eingehenden Nachricht *CallService* modelliert. (siehe Bild 4). Entsprechend wird nach Ablauf des Zustandsautomaten die Antwort der erfolgreichen Abarbeitung zurück geschickt.

Neben den Steuerungsfunktionen müssen die zuvor vorgestellten Softwarekomponenten als Applikationen entwickelt werden und in das Netzwerk integriert werden. Die Abhängigkeiten zwischen Produkt und Produktionsprozess konnten ebenfalls als Zustandsautomat modelliert werden.

Im Zustandsautomat der dem Produkt zugeordneten Softwarekomponente repräsentiert jeder Zustand einen Produktionsschritt. In einem solchen Zustand werden zunächst die notwendigen Informationen vom Discovery-Server eingeholt, und anschließend über den Serviceaufruf (*CallService*) der Prozessschritt gestartet, der vom entsprechenden Modul ausgeführt wird. Nach Abschluss der Bearbeitung sendet das jeweilige Modul eine Nachricht, dass der Prozessschritt durchgeführt wurde. (*AnswerService*). Dies triggert innerhalb des Intelligenten Produkts eine Transition, sodass der nächste Prozessschritt bearbeitet wird.



**Bild 5:** Modellierung von einem intelligentem Produkt und einem Service

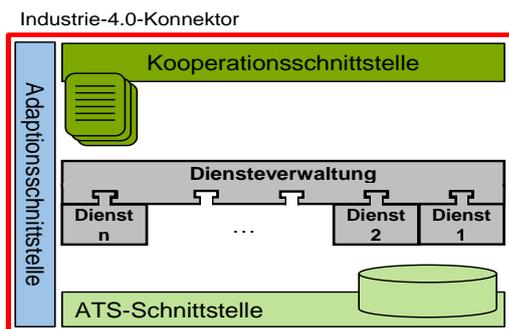
Durch diese Art der Prozesssteuerung wird im Gegensatz zu zentralen SPS-Steuerungen oder fest vernetzten Systemen, eine Flexibilisierung durch eine Zentralisierung der Kommunikationslogik ermöglicht. Komplexe Produkte können eine Modellierung von Teilprodukten voraussetzen. In diesem Fall kann für jedes Teilprodukt ein Zustandsautomat modelliert und die einzelnen Zustandsautomaten zu einem Petri-Netz zusammengeführt werden.

Der Discovery-Server stellt eine Liste dar, die die notwendigen Informationen enthält, damit die dem Produkt zugeordnete Softwarekomponente die Services sowohl informationstechnisch, als auch physikalisch erreichen kann. Diese Informationen werden beim Anmelden eines neuen Steuerungsmoduls in das Netzwerk vom Neuen Modul in einer Register-Message an den Discovery-Server gesendet.

## Einbindung konventioneller Komponenten in IoT-Netzwerke der Automatisierung

Investitionen in Produktionseinrichtungen erfordern einen hohen Kapitaleinsatz. Dies führt zu langen Laufzeiten von Maschinen. Es ist abzusehen, dass es unwirtschaftlich sein wird bestehende Anlagen und deren Komponenten durch neuartige IoT-Komponenten zu ersetzen. Daher ist von einer langen Übergangszeit auszugehen, in der sowohl klassische Komponenten als auch Smarte Komponenten gemeinsam betrieben werden. Somit reicht es nicht aus, ausschließlich das Engineering neuer Smarter Komponenten, zu betrachten.

Es besteht eine Notwendigkeit, Komponenten, welche



**Bild 6:** Aufbau des Industrie-4.0-Konnektors

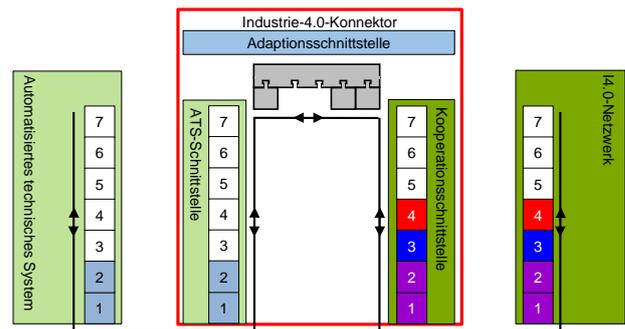
nicht die Eigenschaften einer Smarten Komponente besitzen, zu befähigen mit Smarten Komponenten zu interagieren. Hierzu müssen vor allem die bestehenden Kommunikationsfähigkeiten der Komponente erweitert werden, um sich mit Smarten Komponenten austauschen zu können.

Architekturwürfe der Plattform Industrie 4.0 [StI40] als auch die Internet Reference Architecture (IIRA) [LiMi15] sehen spezielle Strukturen hierzu vor. Im Bereich der Plattform Industrie 4.0 wird derzeit die sog. Verwaltungsschale konzipiert.

Eine Möglichkeit eine entsprechende Struktur für existierende automatisierte Systeme zu realisieren ist das Konzept des Industrie-4.0-Konnektors [FaJa16]. Dieser besteht aus drei Hauptteilen, der Schnittstelle zum automatisierten technischen System (ATS-Schnittstelle), der Schnittstelle zum Netzwerk der Smarten Komponenten (Kooperationsschnittstelle) und der Dienstverwaltung mit den Diensten, welche die Funktionen des automatisierten Systems als Service bereitstellen. Dabei handelt es sich hierbei beim Dienst um eine Softwarekomponente des Konnektors und wird nicht synonym zum Begriff „service“ verwendet. Die Dienstverwaltung koordiniert diese Dienste innerhalb des Konnektors. Sie entspricht dabei auch nicht dem oben erwähnten Konzept eines Discovery Servers. Im Bild 6 sind diese, ergänzt um die Adaptionsschnittstelle, dargestellt. Der Industrie-4.0-Konnektor bildet mit seinen zwei Schnittstellen ein Verbindungselement zwischen dem automatisierten System und dem Netzwerk der Smarten Komponenten.

Das Abrufen der Services, zum Beispiel durch ein intelligentes Produkt, geschieht durch Nachrichten, welche von der Kooperationsschnittstelle mittels der Dienstverwaltung

an die Dienste weitergeleitet werden. Die Dienstverwaltung ist hierbei für das Routing der Nachrichten verant-



**Bild 7:** Datenfluss durch den Industrie-4.0-Konnektor

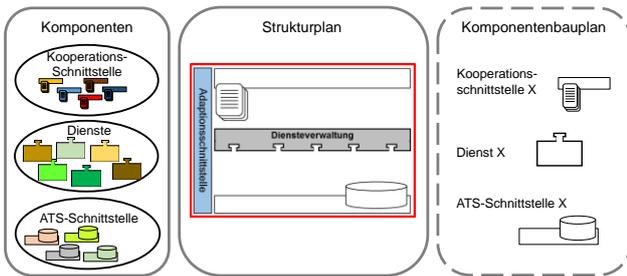
wortlich und ermöglicht damit ein flexibles Hinzufügen und Entfernen verschiedener Dienste.

Zur Bereitstellung der Services, kann ein Dienst mittels der ATS-Schnittstelle auf Informationen des automatisierten Systems zugreifen, bzw. an dieses übermitteln und damit definierte Aktionen durchführen. Ebenso kann der Dienst mittels der Kooperationsschnittstelle Nachrichten an andere automatisierte Systeme senden um weiter Informationen zu erhalten. Da für die Kommunikation zwischen automatisierten Systemen kein standardisiertes Kommunikationsprotokoll existiert, sondern eine Vielzahl an möglichen Protokollen ist die Kooperationsschnittstelle so gestaltet, dass diese austauschbar ist. Damit dies möglich ist, überträgt die Kooperationsschnittstelle, bevor die Nachrichten an die Dienstverwaltung weitergeleitet werden, die Informationen der empfangenen Nachrichten in ein XML basiertes Protokoll. Bei zu sendenden Nachrichten erfolgt dieser Vorgang in umgekehrter Richtung. Mit diesem Ansatz ist auch die Vielzahl an Protokollen, welche von den automatisierten Systemen verwendet werden keine Schwierigkeit. Da allerdings selbst bei Verwendung desselben Protokolls unterschiedliche Syntaxen verwendet werden, ist eine zusätzliche Syntaxübersetzungsschicht notwendig [FaWe16]. Diese Schicht basiert auf einem Modell der ATS-Schnittstelle, welches in einer Ontologie gespeichert ist und von einem Interpreter ausgewertet wird. Mittels der Adaptionsschnittstelle ist es möglich den Industrie-4.0-Konnektor anzupassen, indem z. B. hinterlegte Adressen für Auskunftsdienste oder der verwendete Syntax der ATS-Schnittstelle geändert werden.

Bild 7 zeigt die Funktionsweise des Industrie-4.0-Konnektors, in dem der Datenfluss vom Netzwerk zum ATS und zurück dargestellt ist. Hierbei ist zu beachten, dass zwischen dem Industrie-4.0-Netzwerk und der Kooperationsschnittstelle andere Protokolle eingesetzt werden, als zwischen dem automatisiertem System und der ATS-Schnittstelle. Dies wird durch die unterschiedlichen Farben innerhalb der ISO-OSI-Stacks dargestellt. Verbunden werden diese unterschiedlichen Kombinationen durch die Dienste, welche die Daten jeweils auf der siebten Ebene der ISO-OSI-Stacks abgreifen, verarbeiten und weiterleiten.

Für die Realisierung dieses Konzeptes benötigt der Entwickler sowohl Wissen über das jeweilige automatisierte System als auch über die Details der Programmierung des

Industrie-4.0-Konnektors. Um diese Situation zu vereinfachen wird das Wiederverwendungskonzept Framework verwendet, welches in Bild 8 dargestellt ist [FaGö15]. Dieses reduziert das für die Implementierung des Industrie-



**Bild 8:** Framework zur Erstellung des Industrie-4.0-Konnektors

4.0-Konnektors notwendige Wissen, indem es einen Bauplan des Konnektors in Form einer Programmstruktur mit expliziten Schnittstellen vorgibt. Innerhalb dieses Planes kann der Entwickler bereits implementierte Komponenten an den Schnittstellen integrieren. Existiert keine Komponente mit der gewünschten Funktionalität, bietet das Framework die Möglichkeit eigene Komponenten mit Hilfe des Komponentenbauplans zu realisieren und in das Framework zu integrieren.

Die Evaluation des gezeigten Konzeptes erfolgte sowohl an einem automatisierten System zur Heißwasser-Bereitung, als auch am Modellprozess einer modularen Produktionsanlage. Das Heißwassersystem verfügt über einen CAN-Bus, der zur internen und externen Kommunikation verwendet wird. Zur Realisierung des Konnektors kam hierbei die Programmiersprache Java zum Einsatz. Die Ontologie für die ATS-Schnittstelle wurde mit dem Tool Protege realisiert und mittels der OWL-API Version 3.0 in den Java-Code integriert. Die modulare Produktionsanlage besteht aus einzelnen Produktionsstationen, welche jeweils von einem Raspberry Pi gesteuert werden. Der Industrie-4.0-Konnektor wurde hierbei ebenfalls mit der Programmiersprache Java realisiert, wobei im Unterschied zum Heißwassersystem kein CAN-Bus zur Kommunikation mit dem automatisierten System zum Einsatz kommt, sondern verschiedene Skripte innerhalb des Raspberry Pi.

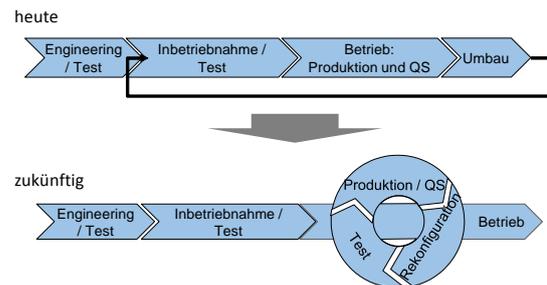
Beide Realisierungen haben gezeigt, dass es mit dem Industrie-4.0-Konnektor möglich ist bestehende automatisierte Systeme so zu erweitern, dass diese gemeinsam mit smarten Komponenten in ein Produktionssystem integriert werden können. Hierbei ist von Vorteil, dass für diese Erweiterung die bestehenden Systeme nicht verändert werden müssen.

### Absicherung von Smarten Komponenten in Produktionssystemen

Diese hohe Flexibilität eines serviceorientierten Netzwerks Smarter Komponenten schafft aber auch hohe Anforderungen an dessen Absicherung. Heutzutage gilt bei Systemen der Produktionsautomatisierung größtenteils der Leitsatz „never change a running system“. So werden Änderungen an Automatisierungssystemen von Produktionsanlagen nach der Inbetriebnahme nur selten durchgeführt. Sind grö-

ßere Umbaumaßnahmen notwendig, werden diese Änderungen bei Wiederinbetriebnahme durch Inbetriebnahmetests abgesichert. So finden heutzutage Testmaßnahmen hauptsächlich während des Engineerings und der Inbetriebnahme statt (siehe Bild 9 oben) [ZeWe15].

Kapitel 2 beschreibt die Bedeutung von Rekonfigurationen für zukünftige Systeme der Produktionsautomatisierung. Aufgrund der Flexibilität der einzelnen smarten Komponenten werden Umbaumaßnahmen zunehmend durch Rekonfigurationen abgelöst werden. Diese Rekonfigurationen, welche auch Änderungen des Produktionsablaufs darstellen, müssen während der Betriebsphase durch Tests abgesichert werden. So ergibt sich ein Kreislauf von Produktion, Rekonfiguration und Test während der Betriebsphase (siehe Bild 9 unten).

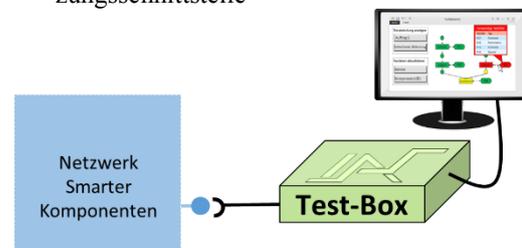


**Bild 9:** Testen im Lebenszyklus einer Produktionsanlage

Die dezentrale Struktur einer SOA verursacht einer Verteilung des Systemwissens auf die einzelnen Komponenten des IT-Systems. Dieses Systemwissen ist dabei nicht statisch, sondern ändert sich beispielsweise durch Ad-Hoc-Vernetzung und Rekonfigurationen der Komponenten. Zur Absicherung des Gesamtsystems ist das Wissen über die Abhängigkeiten zwischen den Smarten Komponenten aber zwingend erforderlich. Dies ist darauf zurückzuführen, dass es nicht genügt Komponenten isoliert zu testen (Modultest), sondern auch die Interaktion der Komponenten validiert werden muss (Integrationstest).

Um diese Abhängigkeiten zur ermitteln und somit die Testfälle auszuwählen, welche zur Absicherung des Gesamtsystems notwendig sind wird ein Assistenzsystem am IAS entwickelt, welches folgende Funktionalitäten bietet:

- Erkennung der Systemstruktur
- Modellierung der Systemabhängigkeiten
- Analysieren der Auswirkungen von Änderungen
- Ableiten notwendiger Testfälle zur Absicherung
- Darstellung der Anweisungen über eine Benutzungsschnittstelle



**Bild 10:** Übersicht der Testmanagement-Box

Zur Erkennung der Systemstruktur des Automatisierungssystems integriert sich die Testmanagementbox als Client in das SOA-Netzwerk (siehe Bild 6) [ZeWe16]. Dabei liest

die Testmanagement-Box Informationen über vorhandene Smarte Komponenten aus und baut eine Systemstruktur auf. Die Informationen über das Verhalten, die Schnittstellen und zur Absicherung notwendigen Testfälle der Komponenten müssen dabei einmalig modelliert werden. Denkbar ist, dass der Hersteller der Komponenten die Informationen mitliefert und dadurch einen Mehrwert seines Produkts erschafft. Mit diesen Informationen ist es möglich die Abhängigkeiten zwischen den Komponenten zu modellieren. Anhand des Abhängigkeitsgraphs ist es möglich die Komponenten zu identifizieren, welche von Änderungen betroffen sind. Diese Informationen werden graphisch aufgearbeitet und über eine Benutzungsschnittstelle dem Test-Ingenieur zu Verfügung gestellt. Dies hat das Ziel, bei geringem Testaufwand den Absicherungsgrad signifikant zu erhöhen.

Der Prototyp, welcher im Rahmen, dieses Projektes aufgebaut wird, besteht einerseits aus einem Automatisierungssystem einer Produktionsanlage und der Testmanagement-Box. Dabei kommunizieren die Smarten Komponenten des Automatisierungssystems mittels OPC-UA. Bei OPC-UA handelt es sich um ein serviceorientiertes Protokoll, welches dem Server-Client-Paradigma folgt. Die Testmanagement-Box kann sich dabei, über einen OPC-UA Client ad-hoc in das Automatisierungssystem integrieren und über einen Discovery-Server Informationen über vorhandene Komponenten beziehen.

## 4 Zusammenfassung

Mithilfe der vorgestellten Konzepte lässt sich die Realisierung in IoT-Netzwerken für die Automatisierung vereinfachen. Für die neuartigen Rahmenbedingungen des IoT in der industriellen Automatisierungstechnik wurden drei Konzepte vorgestellt.

Die Konzepte ergeben bieten dabei folgenden Mehrwert:

- Die modellbasierte Entwicklung der verteilten Steuerungen führt zu einer Reduktion der Komplexität für den Entwicklungsingenieur. Dabei können über die Engineeringphase hinaus die Modelle für Echtzeit-Monitoring, Diagnosesysteme oder für das Testmanagement verwendet werden.
- Für die Anbindung konventioneller Komponenten wurde ein I40-Konnektor vorgestellt. Dieser erlaubt die Integration bestehender, konventioneller Anlagen in einen Industrie 4.0-Verbund. Durch die Nutzung eines Frameworks wird der Entwicklungs- und Adaptionsprozess des Konnektors für spezifische Altanlagen vereinfacht.
- Zur permanenten Absicherung flexibler, sich rekonfigurierender Automatisierungssysteme wurde das Konzept eines Assistenzsystems zum Testmanagement dargelegt. Dies berücksichtigt die sich stetig wandelnde Systemumgebung in der Automatisierung der Zukunft. Dies soll auch bei komplexen, sich verändernden Abhängigkeit eine ausreichende Testabdeckung ermöglichen.

Die Konzepte stellen zurzeit noch monolithische Säulen dar und greifen eine spezifische Herausforderung zukünftiger Automatisierungssysteme in der Produktion auf. Um

einen Mehrwert zwischen den Konzepten zu schaffen, ist der Aufbau zukünftiger übergreifender Prototypen geplant.

## 5 Literatur

- [Aca16] Acatech: Umsetzungsempfehlung für das Zukunftsprojekt Industrie 4.0, April 2016.
- [Hein] Heinecke: AUTomotive Open System ARchitecture – An Industry-Wide Initiative to Manage the Complexity of Emerging Automotive E/E-Architectures.
- [WES16] Westkämper E.; Löffler C.: Strategien der Produktion: Technologien, Konzepte und Wege in die Praxis, Springer Vieweg, 2016.
- [P40IIC] <https://www.plattform-i40.de/I40/Redaktion/DE/Pressemitteilungen/2016/2016-03-02-kooperation-iic.html>, abgerufen am 16.08.2016.
- [ScWe15] Schmidt, J.; Weyrich M.: Entwurf einer embedded Software-Architektur für dezentrale microcontrollerbasierte Steuerungssysteme, Automation 2015, VDI-Verlag, Baden-Baden, 2015.
- [ScZe16] Schmidt, J.; Zeller, A.; Weyrich M.: Ansatz zur modellgetriebenen Entwicklung flexibler Automatisierungssysteme durch Serviceorientierung, EKA 2016.
- [StI40] Statusreport Industrie 4.0 – Technical Assets, [https://www.vdi.de/fileadmin/vdi\\_de/redakteur\\_dateien/gma\\_dateien/Statusreport\\_I40\\_TechnicalAssets\\_Begriffe\\_WEB.pdf](https://www.vdi.de/fileadmin/vdi_de/redakteur_dateien/gma_dateien/Statusreport_I40_TechnicalAssets_Begriffe_WEB.pdf); Nov. 2015.
- [LiMi15] Lin S.; Miller B. et al: Industrial Internet Reference Architecture - IIRA-1-7-ajs.pdf. Available: <http://www.iiconsortium.org/IIRA-1-7-ajs.pdf> (2015, Oct. 15).
- [FaJa16] Faul, A.; Jazdi, N.; Weyrich M.: Approach to Interconnect Existing Industrial Automation Systems with the Industrial Internet, 20<sup>th</sup> IEEE ETFA, Berlin, 2016, im Druck.
- [FaWe16] A. Faul and M. Weyrich, "Flexible Gatewayschnittstelle zur Anbindung unterschiedlicher Netzwerke," in Entwurf komplexer Automatisierungssysteme: EKA 2016 : Beschreibungsmittel, Methoden, Werkzeuge und Anwendungen, 2016.
- [FaGö15] Faul, A.; Göhner P.: Framework für die dynamische Kooperation von automatisierten Systemen, Automation 2015, VDI-Verlag, Baden-Baden, 2015.
- [DIN91345] DIN SPEC 91345: Referenzarchitekturmodell Industrie 4.0 (RAMI 4.0)
- [ZeWe15] Zeller, A.; Weyrich M.: Test Case Selection for Networked Production Systems, 20<sup>th</sup> IEEE ETFA, Luxemburg, 2015.
- [ZeWe16] Zeller, A.; Weyrich M.: Absicherung der Rekonfigurationen von Produktionssystemen während des Betriebs, Automation 2016, Baden Baden, 2016.
- [TPBS+13] Trentesaux, D., Pach, C., Bekrar, A., Sallez, Y., Berger, T., Bonte, T., ... & Barbosa, J. (2013). Benchmarking flexible job-shop scheduling and control systems. *Control Engineering Practice*, 21(9), 1204-1225.
- [Rei09] Reißener, B.; Feldbusssysteme zur industriellen Kommunikation, München: Oldenbourg-Industrieverlag, 2009.
- [Str98] Strohmarm, G.; Automatisierungstechnik 1, Grundlagen, analoge und digitale Prozeßleittechnik, München: Oldenbourg, 1998.
- [BuMa08] Buzacott, J.; Mandelbaum, M.: Flexibility in manufacturing and services: achievements, insights and challenges, *Flexible Services and Manufacturing Journal* 20, Nr. 1, S. 13–58, 2008.
- [Elm05] ElMaraghy, H.: Flexible and reconfigurable manufacturing systems paradigms, *Intern. Journal of Flexible Manufacturing Systems* 17, Nr. 4, S. 261–276, 2005.
- [BaCh06] Babiceanu, R.F.; Chen, F.: Development and Applications of Holonic Manufacturing Systems: A Survey, *Journal of Intelligent Manufacturing* 17, Nr. 1, S. 111–131, 2006.
- [BrNo03] Brennan, R.W.; Norrie, D.H.: From FMs to HMs, *Advanced Information Processing, Agent-based manufacturing: Advances in the holonic approach*, S. M. Deen, Ed, Berlin: Springer, S. 31–49, 2003.
- [DWHB15] Deuse, Jochen, et al. "Gestaltung von Produktionssystemen im Kontext von Industrie 4.0." *Zukunft der Arbeit in Industrie 4.0*. Springer Berlin Heidelberg, 2015. 99-109.