

Fortschritt-Berichte pak

Band 37
Lean Automation

Dennis Kolberg, M. Sc.

**Entwicklung einer Referenzarchitektur
zur Realisierung von Methoden der
Lean Production mittels digitaler
Technologien**

Entwicklung einer Referenzarchitektur zur Realisierung von Methoden der Lean Production mittels digitaler Technologien

Vom Fachbereich Maschinenbau und Verfahrenstechnik
der Technischen Universität Kaiserslautern
zur Verleihung des akademischen Grades
Doktor-Ingenieur (Dr.-Ing.)
genehmigte Dissertation

von
Dennis Kolberg, M. Sc.
aus Bückeberg

Tag der mündlichen Prüfung: 07. Juni 2018

Dekan: Prof. Dr.-Ing. Jörg Seewig

Promotionskommission:

Vorsitzender: Prof. Dr.-Ing. Roman Teutsch
1. Berichterstatter: Prof. Dr.-Ing. Dr. h.c. Detlef Zühlke
2. Berichterstatter: Prof. Dr.-Ing. Martin Ruskowski
3. Berichterstatter: Prof. Dr.-Ing. Klaus-Dieter Thoben

Vorwort des Verfassers

Schon während meines Studiums faszinierte mich nicht nur die Einfachheit und Effizienz der Lean Production, sondern auch die Vision und das Potential des Internets der Dinge bzw. später Industrie 4.0. Die vorliegende Dissertation ist die Vertiefung und Umsetzung einer von mir jahrelang verfolgten Idee, beide Themen miteinander sinnvoll zu verbinden. Sie entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter im Forschungsbereich Innovative Fabrikssysteme (IFS) am Deutschen Forschungszentrum für Künstliche Intelligenz (DFKI) sowie bei der Technologie-Initiative SmartFactory KL e.V.

Für die Betreuung der Arbeit möchte ich mich besonders bei Herrn Professor Zühlke bedanken, der früh das Potential erkannte, welches sich durch die Verbindung der im ersten Moment nicht offensichtlich zusammengehörenden Bereiche ergibt. Er und Herr Professor Ruskowski, bei welchem ich mich ebenfalls besonders bedanken möchte, prägten durch die Übertragung von Projekt- und Führungsverantwortung maßgeblich meine fachliche und persönliche Entwicklung während der Zeit.

Bei Herrn Professor Thoben möchte ich mich für die Anfertigung des Drittgutachtens und für die Betreuung meiner Masterarbeit seinerzeit bedanken, durch welche ich die Grundidee für diese Dissertation bekam. Selbstverständlich gilt mein Dank auch Herrn Professor Teutsch für die Übernahme des Vorsitzes der Prüfungskommission.

Bedanken möchte ich mich bei allen Freunden und Kollegen am IFS und in der SmartFactory-KL für die vorbildliche Zusammenarbeit und das positive Arbeitsklima auch in arbeitsintensiven Phasen. Erwähnen möchte ich dabei insbesondere Max Birtel, Ellina Marseu, Dr. Marius Orfgen, Dr. Mathias Schmitt, Michael Wendling, Dr. Stephan Weyer und Dr. Fabian Quint, die mit Humor, Hilfsbereitschaft und Teamgeist den Arbeitsalltag und die Arbeit an der Promotion erleichterten. Auch gilt mein Dank den Studenten Joshua Knobloch und Eric Jedermann, die mich bei der Realisierung der Demonstration tatkräftig unterstützten.

Zuletzt möchte ich meiner Familie und insbesondere meinen Eltern danken, die mich nicht nur in frühen Jahren prägten, sondern in den letzten Jahren für die Dissertation oft auf mich verzichten mussten. Mein allergrößter Dank jedoch gilt meiner Ehefrau Ekaterina Kolberg, die für meine Dissertation nicht nur persönliche Interessen zurückstellte, sondern mir täglich den Rücken freihielt und mich in schwierigen Zeiten motivierte. Ihre Unterstützung trug maßgeblich zum Gelingen der Arbeit bei.

Hamburg, im Juni 2018

Kurzfassung

Spätestens seit den 1990ern bildet die Lean Production den Status quo der Produktionssysteme in der diskreten Klein- und Großserienfertigung. Mittels der Umstellung auf eine nachfrageorientierte Produktion und Vermeidung jeglicher nicht-wertschöpfender Aktivitäten konnte die Lean Production bei geringen Investitionen hohe Effizienzsteigerungen erreichen. Heute stößt die Lean Production allerdings an ihre Grenzen, da sie nicht das Potenzial moderner Informations- und Kommunikationstechnologie berücksichtigt. Der Lean Production mangelt es u.a. an der notwendigen Wandelbarkeit der Fertigungslinien, um zukünftige Anforderungen wie z.B. eine kundenindividuelle Fertigung in Losgröße 1 zu ermöglichen.

Die stattfindende Digitalisierung der Produktion will diesen zukünftigen Produktionsanforderungen gerecht werden. Neue digitale Technologien sowie die gestiegene Leistung der Komponenten ermöglichen eine intelligente Produktion, die Mitarbeiter in komplexen Produktionsprozessen unterstützt. Der gleichzeitig stattfindende Preisverfall und die Miniaturisierung in der Informations- und Kommunikationstechnologie führen auch in der Produktion zu allgegenwärtigen, vernetzten Computern. Das Potenzial dieser technologischen Entwicklung ist unumstritten, dennoch mangelt es aktuell an einer übergreifenden und ganzheitlichen Betrachtung, die eine Integration der Digitalisierung und der neuen Anwendungen in bestehende Produktionsumgebungen miteinbezieht.

Ziel der vorliegenden Arbeit ist die Erarbeitung einheitlicher informationstechnischer Schnittstellen für die Lean Production. Die erarbeitete Referenzarchitektur definiert technologieunabhängige Schnittstellen und schafft den notwendigen Rahmen für die Nutzung digitaler Technologien in bestehenden Lean-Methoden. Bestandteil der Referenzarchitektur ist eine Systemarchitektur, die Rollen und deren Verhältnisse beschreibt. Ergänzt wird sie durch ein Informationsmodell, das die je Rolle benötigten Funktionalitäten und die ausgetauschten Nachrichten spezifiziert, sowie durch eine Softwarearchitektur für die Umsetzung der Schnittstellen. Die Referenzarchitektur wurde auf einen bestehenden Forschungsdemonstrator der Technologie-Initiative SmartFactory KL e.V. exemplarisch übertragen.

Abstract

Since the 1990s there is no doubt that Lean Production is the state of the art production system in discrete parts manufacturing. With its pull-driven production and elimination of non-value-adding tasks, Lean Production is able to improve efficiency with little invest. Nevertheless, Lean Production is reaching its limits nowadays. It does not take into account the potential of modern information and communication technology and is not able to realize a lot-size one production of highly customized products. Furthermore, there is a lack of changeability in production lines to meet future production requirements.

Digitization in production tries to cope with these requirements. New digital technologies and the increased computing power are paving the way to intelligent machines and assistant systems for complex manufacturing processes. In addition, decreasing prices and miniaturization of components lead to ubiquitous and connected computers. There is no doubt about the potential of this technological development in production. Nevertheless, a comprehensive and holistic approach which describes how digitization and digitized solutions can be implemented in existing production systems is missing.

The objective of this thesis is the development of uniform communication interfaces for the lean production. The developed reference-architecture defines in a technology-independent way how to realize such interfaces. Besides, it enables the usage of digital technologies in existing lean-methods. It consists of a system architecture which defines roles and their connections. Furthermore, necessary functionalities for each role as well as exchanged messages are specified in an information model. In order to support the realization, the reference-architecture proposes a software-architecture to realize the common and uniform communication interfaces. The reference-architecture has been implemented in an existing research demonstrator at the Technologie-Initiative SmartFactory KL e.V.

Inhalt

1. Einleitung	1
2. Stand der Technik	5
2.1 Digitalisierung in der Produktion.....	5
2.1.1 Begriffsabgrenzung und Paradigmen.....	5
2.1.2 Dezentrale Architekturen für die Produktionssteuerung	7
2.1.3 Informationstechnische Schnittstellen und Kommunikation.....	19
2.1.4 Informationsmodelle und -modellierung	25
2.1.5 Bewertung des Status quo.....	36
2.2 Lean Production.....	38
2.2.1 Begriffsabgrenzung und historische Entwicklung	38
2.2.2 Gestaltungsprinzipien und Methoden.....	41
2.2.3 Lean Automation	50
2.2.4 Bewertung des Status quo.....	55
3. Problemstellung, Zielsetzung und Vorgehensweise.....	57
3.1 Problemstellung	57
3.2 Zielsetzung	58
3.3 Vorgehensweise	60
4. Referenzarchitektur zur Digitalisierung der Lean Production	63
4.1 Methodik zur Entwicklung der Referenzarchitektur.....	63
4.1.1 Prinzipien der Methodik	63
4.1.2 Schritt 1: Lean-Methoden sammeln und bewerten.....	64
4.1.3 Schritt 2: Anforderungen analysieren.....	64
4.1.4 Schritt 3: Systemarchitektur und Informationsmodell entwickeln.....	65
4.1.5 Schritt 4: Schnittstellenarchitektur entwickeln	66
4.2 Bewertung, Modellierung und Anforderungskatalog der Lean-Methoden	67
4.2.1 Bewertung und Auswahl von Lean-Methoden.....	67
4.2.2 Modellierung der ausgewählten Lean-Methoden	70
4.2.3 Funktionale Anforderungen des Anforderungskataloges.....	79
4.2.4 Nicht-funktionale Anforderungen des Anforderungskataloges	82
4.3 Bestandteile der Referenzarchitektur	85
4.3.1 Systemarchitektur.....	85
4.3.2 Informationsmodell der Schnittstelle	91
4.3.3 Schnittstellenarchitektur.....	109

5. Prototypische Realisierung der Referenzarchitektur	113
5.1 Anwendungsszenarien und Rahmenbedingungen	113
5.2 Technische Realisierung.....	119
6. Bewertung und Ausblick	129
6.1 Zusammenfassende Bewertung.....	129
6.2 Ausblick auf weiteren Forschungs- und Entwicklungsbedarf	132
7. Zusammenfassung	135
8. Literaturverzeichnis	137
Anhang A – Anwendungsfälle der Lean-Methoden.....	157
Anhang B – Aktivitätsdiagramme der Lean-Methoden	158
Anhang C – Kommunikationsdiagramme der Lean-Methoden.....	163
Anhang D – Liste der funktionalen Anforderungen	166
Anhang E – Spezifikation der je Rolle anzubietenden Funktionalitäten.....	169
Anhang F – Spezifikation der Nachrichten.....	173
Liste der betreuten studentischen Arbeiten.....	187
Lebenslauf.....	189

1. Einleitung

Mit ihrer 1992 veröffentlichten Studie über Toyotas Produktionssystem prägten [Wom+92] nicht nur den Begriff „Lean Production“, sondern sorgten insbesondere in der westlichen Industrie für einen tief greifenden Wandel in der Produktion. Die Umstellung auf eine nachfrageorientierte Produktion, durchgängig standardisierte und transparente Prozesse sowie die Vermeidung jeglicher nicht-wertschöpfender Aktivitäten waren im Vergleich zur klassischen, von Henry Ford geprägten Massenproduktion, Erfolgskonzepte, die den Automobilhersteller Toyota zum Weltmarktführer machten. 30% kürzere Durchlaufzeiten, 20% geringere Lagerbestände und eine Effizienzsteigerung von 30% pro Mitarbeiter bei verhältnismäßig geringen Investitionen sind Gründe, weshalb heute fast alle diskreten Klein- und Großserienfertigungen die Prinzipien und Methoden der Lean Production einsetzen. [Ger11; Kor+04; Neu07; Ger11; Dom+15] Der Begriff „Lean Production System“ (LPS) fasst heute die unterschiedlichen, individuellen Anpassungen der Lean Production zusammen, die alle auf dem in den 1950ern entstandenen Ansatz von Toyota basieren [Dom+15].

Auch wenn die Vorteile von LPS unumstritten sind, stoßen die Prinzipien und Methoden der Lean Production heute an ihre Grenzen. Die steigende Volatilität der Märkte, immer kürzere Produktlebenszyklen sowie die kundenindividuelle Einzelfertigung sind Entwicklungen, die zur Zeit der Lean Production noch nicht abzusehen waren und somit keine Berücksichtigung fanden [Nyh08]. Konzepte wie eine gleichmäßige Auslastung und einheitliche Taktzeiten von Fertigungslinien werden diesen neuen Anforderungen nicht gerecht. Methoden wie das Kanban-System sind ferner nicht für sich stetig ändernde Fertigungslinien konzipiert. [Erl07; Dic07c] Die Ziele der Lean Production, die Verbesserung von Kosten, Qualität und Durchlaufzeit, müssen heute um das Ziel Wandelbarkeit ergänzt werden [Mou+12]. Wandelbarkeit ist die Fähigkeit, sich zukünftig schnell auf nicht bekannte Umstände einzustellen. Darüber hinaus bleiben die Potenziale moderner Informations- und Kommunikationstechnologie (IKT) wie z.B. verkürzte Informationsflüsse und intelligente, sich selbst steuernde Systeme in der Lean Production ungenutzt.

Der technische Fortschritt, kleinere Baugrößen und ein gleichzeitiger Verfall der Beschaffungskosten für IKT führten in den letzten Jahren zu einer Verbreitung von Computern. Begriffe wie „Ubiquitous Computing“ oder „Internet of Things“ beschreiben allgegenwärtige, nicht sichtbare IT-Systeme, die mit dem Internet vernetzt sind und uns im Alltag unterstützen [Wei91]. Nach der Einführung von Produktionsanlagen, der Elektrifizierung und dem Einsatz von Automatisierungstechnologie wird unter dem Begriff „Industrie 4.0“ die nächste Revolution in der Produktion prognostiziert. Industrie 4.0 beschreibt den vermehrten Einsatz von IKT, um den zukünftigen Marktanforderungen und dem dadurch entstehenden Bedarf nach Wandelbarkeit in der Produktion gerecht zu werden. Als Kerntechnologie wurden sogenannte Cyber-Physische Systeme (CPSe) identifiziert. [Bun; Kag+13] Aufgrund ihrer Leistungsfähigkeit und den Interaktionsmöglichkeiten mit der Umgebung ermöglichen CPSe in der Produktion autonom agierende, intelligente Systeme [Lee08; Bro10a]. Aus der Fabrik wird eine Smart Factory [Züh10]. Neben CPSen finden auch andere, neuartige digitale Technologien wie mobile Endgeräte, Datenbrillen mit Augmented Reality und künstliche Intelligenz Einzug in die Produktion. Das, was wie eine Fiktion klingt, wurde bereits in Forschungsprojekten demonstriert und ist teilweise am Markt als Produkt erhältlich (siehe z.B. [Rei+15]).

Diese technologiegetriebene Entwicklung vernachlässigt allerdings die Integration in und die Migration von bestehenden Produktionsumgebungen. Existierende Referenzarchitekturen beschränken sich auf die technologische Sicht und berücksichtigen die organisatorischen Prozesse und die Rolle des Menschen nicht ausreichend. Das ist insofern überraschend, da die Notwendigkeit einer ganzheitlichen Betrachtung spätestens seit der Ära der Computer Integrated Manufacturing (CIM) bekannt ist. [Deu+15; Züh10; Deu+15] Darüber hinaus fällt heutzutage ein Großteil der Kosten für die Automatisierung im Engineering und für die spätere Anpassung an neue Anforderungen an [Col+05; Lüd+]. Ein Ansatz, dem entgegenzuwirken, ist die sinnvolle Modularisierung und Etablierung definierter Kommunikationsschnittstellen. Modularisierung kann die Interoperabilität insbesondere mit bestehenden Systemen herstellen, die Komplexität reduzieren und die Wandelbarkeit von Produktionsumgebungen verbessern [Deu+15; Col+05].

Aus den heutigen Defiziten der weitverbreiteten Lean Production sowie dem Potenzial der vermehrten Integration von IKT in der Produktion ergibt sich die Frage, wie digitale Technologien die Wandelbarkeit der Lean Production unterstützen können. Ähnlich wie die von [Sch13] bereits 1997 im Kontext der Architektur Integrierter Systeme (ARIS) veröffentlichten Referenzmodelle für administrative Prozesse bedarf es eines formalisierten Modells, das die Prozesse und Interaktionen der Lean Production für die Umsetzung mittels digitaler Technologien beschreibt. Die Wirkung eines solchen Referenzmodells mit seinen beschriebenen Prozessen und Akteuren hat sich in der Vergangenheit im Enterprise-Resource-Planning (ERP)-System SAP gezeigt, in das die Arbeiten von Scheer einfließen [Sch+15] (siehe Abbildung 1).

Die vorliegende Arbeit thematisiert einen ähnlichen Ansatz für die Lean Production. Die erarbeitete, technologieunabhängige Referenzarchitektur legt in einer übergeordneten Systemarchitektur die notwendigen, mit informationstechnischen Schnittstellen ausgestatteten Rollen und Ressourcen fest. In einem Informationsmodell spezifiziert sie die von jeder Rolle anzubietenden Dienste und vereinheitlicht die ausgetauschten Nachrichtentypen. Ergänzt wird beides durch eine Softwarearchitektur für die informationstechnischen Schnittstellen, die bestehende Systeme und die Verwendung von CPSen berücksichtigt. Die in der Referenzarchitektur definierten einheitlichen Schnittstellen vereinfachen die Integration digitaler Techno-

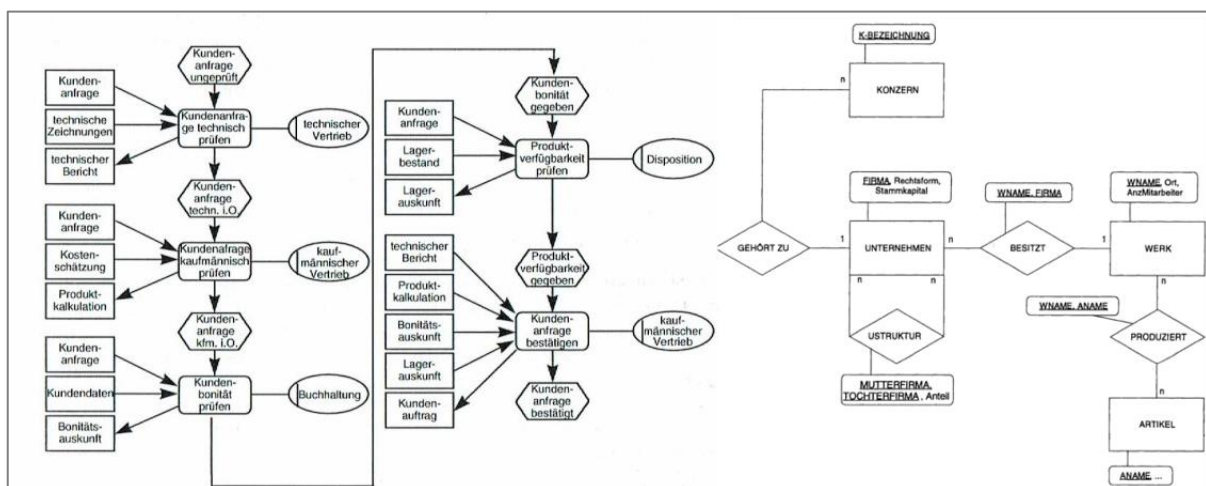


Abbildung 1 Beispiel für ein Referenzmodell zur Abwicklung einer Kundenanfrage (links) und für ein Informationsmodell (rechts) aus ARIS (Quelle: [Sch02], [Sch13a])

logien und unterstützen die Modularisierung in der Lean Production. Hierdurch erreicht sie das angestrebte Ziel, die Wandelbarkeit der Lean Production zu verbessern.

Im Folgenden wird in Kapitel 2.1 der Status quo relevanter Themen der Produktionsdigitalisierung aufgearbeitet und für diese Arbeit relevante Wissensgrundlagen beschrieben. Kapitel 2.2 widmet sich der Lean Production und aktuell vorhandenen, digitalisierten Anwendungen hierfür. Aufbauend auf diesen Erkenntnissen leitet Kapitel 3 das Ziel und die notwendigen (Teil-)Ergebnisse dieser Arbeit her. Die erarbeitete Referenzarchitektur wird anschließend in Kapitel 4 vorgestellt und in Kapitel 5 in einer prototypischen Realisierung verifiziert. Zuletzt widmet sich Kapitel 6 der kritischen Betrachtung der Ergebnisse und gibt einen Ausblick auf weitere, auf diese Arbeit aufbauende Forschungsthemen.

2. Stand der Technik

Das vorliegende Kapitel beschreibt die für diese Arbeit relevanten Themen und gibt einen Überblick über den aktuellen Stand der Forschung und Entwicklung. Wie in der Einleitung bereits hervorgehoben, verbindet die vorliegende Arbeit die aktuell stattfindende Digitalisierung in der Produktion (Kapitel 2.1) mit der bestehenden Lean Production (Kapitel 2.2).

2.1 Digitalisierung in der Produktion

Im Folgenden wird im Themenfeld Digitalisierung neben den der Digitalisierung zugrundeliegenden Paradigmen und vorhandenen Architekturen als Grundlage für die Referenzarchitektur insbesondere auf die digitale Kommunikation eingegangen. Dies umfasst bestehende Schnittstellen, Kommunikationsmethoden sowie Informationsmodelle zur Standardisierung des Informationsaustausches. Das Kapitel schließt mit einer Bewertung des aktuellen Status Quo und den bestehenden Defiziten ab.

2.1.1 Begriffsabgrenzung und Paradigmen

In den vergangenen Jahren fand der Begriff „Digitalisierung“ im deutschsprachigen Raum große Verbreitung. Er beschrieb ursprünglich die digitale Darstellung von Informationen oder die Umwandlung analoger Signale in digitale Signale [Bib]. Ein Beispiel hierfür ist ein Analog-Digital-Wandler, der analoge Messwerte aus einem Temperatursensor in digitale Werte umwandelt. In den letzten Jahren veränderte sich das Verständnis für den Begriff. Digitalisierung beschreibt heute die zunehmende Einführung von modernen digitalen Technologien in Wirtschaft, Gesellschaft und Produktion [Pla]. Dies umfasst zum einen die vermehrte Nutzung von IKT in bestehenden Anwendungen, zum anderen aber auch die Einführung neuer, auf IKT basierender Geschäftsmodelle. Digitalisierung ist ein kontinuierlicher Prozess der Veränderung, der nicht nur technische, sondern auch organisatorische Aspekte betrifft. [Mer+16; Jun+16; Pau16] In dieser Arbeit wird Digitalisierung wie folgt verstanden:

Digitalisierung beschreibt die bewusste Integration von IKT in Prozesse und Strukturen in einem Betrachtungsbereich (in diesem Fall die innerbetriebliche Produktion), welcher zuvor nicht oder nur untergeordnet diese Technologie nutzte.

Beispiele für Digitalisierung in der Produktion sind die Einführung mobiler Endgeräte wie Tablets als Ersatz für Papier, die kontextbezogene Informationsanreicherung mithilfe von Augmented Reality in Form von z.B. Datenbrillen (siehe Abbildung 2) oder, im Rahmen der digitalen Vernetzung, das anlagenübergreifende Auslesen und Interpretieren von Sensor- und Aktorinformationen zur Statusüberwachung.

Im Zusammenhang mit der Digitalisierung ist Industrie 4.0 zu nennen. Die deutsche Bundesregierung verkündete 2011 im Rahmen der digitalen Agenda den Begriff zusammen mit einem dazugehörigen Förderprogramm. Andere Regionen wie z.B. die USA, China oder Europa adaptierten ihn kurze Zeit später und legten ähnliche Programme fest (siehe z.B. [Xin15; The16; Bun]). Heutzutage ist der Begriff „Industrie 4.0“ neben der Digitalisierung nicht nur im deutschsprachigen Raum allgegenwärtig.



Abbildung 2 Beispiel für den Einsatz von Augmented Reality und Datenbrillen für Wartungsarbeiten in der Produktion (Quelle: Technologie-Initiative SmartFactory KL e.V.)

Industrie 4.0 beschreibt die Vision der durchgängig digital vernetzten, weitestgehend autonom agierenden Produktion, in der Komponenten miteinander Informationen austauschen und sich selbst koordinieren. Die Übertragung der Ideen des Internets-der-Dinge auf die Produktion soll diese befähigen, den zukünftigen Bedarf an hochindividualisierten Produkten zu erfüllen. Möglich ist dies dank des technologischen Fortschrittes, der Miniaturisierung in der IKT und des gleichzeitig stattfindenden Preisverfalles dieser Komponenten. Im Gegensatz zur CIM-Ära in den 1980ern strebt Industrie 4.0 nicht die Substitution des Menschen an, sondern dessen Unterstützung. [aca14; Pla; Kag+13]

Kernparadigmen von Industrie 4.0 sind neben der Digitalisierung der Produktion die Modularisierung zur Verbesserung der Wandelbarkeit von Anlagen und Fertigungslinien. Außerdem gehören die dadurch bedingte Dezentralisierung der Steuerung in der Produktion hin zu autonom agierenden Systemen, die durchgängige digitale Vernetzung und die digitale Kommunikation aller in der Produktion beteiligten Entitäten dazu. [Kag+13]

Modularisierung beschreibt die Unterteilung eines Systems in kleinere Einheiten mit definierten Schnittstellen, die eigenständige Funktionalitäten anbieten. Sie ist ein Ansatz, der den Widerspruch zwischen hoher Variantenvielfalt bzw. kundenindividueller Fertigung und kostengünstiger, effizienter Nutzung der Produktionsressourcen mindert. Sie verbessert die Wandelbarkeit durch wiederverwendbare und einfach austauschbare Module. Außerdem reduziert die Modularisierung die Komplexität technischer Systeme, da mit steigender Abstraktionsebene die Detailfunktionalität von Modulen aufgrund der klaren Schnittstellen und Interaktionen außer Acht gelassen werden kann (vgl. „Black-Box-Ansatz“). Die Objektorientierung und damit einhergehende Modularisierung ermöglichte in der Softwareentwicklung z.B. die Entwicklung komplexer Systeme mit vielen Beteiligten. [Mil+98; Kra+07; Mou+12]

Modularisierung im Kontext der Produktion bezieht sich sowohl auf den physischen Aufbau als auch auf die funktionale Unterteilung eines technischen Systems. Im Gegensatz zu einem Baustein muss ein Modul eine signifikante Aufgabe erfüllen. Die Interaktionen mit der Umgebung, d.h. Ein- und Ausgabewerte, sowie die Beschreibung von Schnittstellen, d.h. Verbindungs- und Austauschpunkte, stehen bei der Modulbeschreibung im Mittelpunkt. Die Umgebung beeinflusst hierbei immer die Modularisierung. Je nach Bezugssystem kann bei-

spielsweise eine Entität entweder ein eigenständiges System oder ein Modul in einem anderen System sein. [Mil+98]

Um die signifikanten Aufgaben selbstständig übernehmen zu können, benötigen die Module eine eigenständige Steuerungskomponente, welche interne Prozesse ausführt und kontrolliert. Dies verlagert zwangsläufig die Steuerungslogik von einer zentralen Komponente hin zu mehreren, verteilten Steuerungen. Diese bereits in der Praxis beobachtbare Entwicklung lässt sich als Dezentralisierung bezeichnen. Dezentrale Architekturen beschreiben hierfür den strukturellen Aufbau. Forschung und Praxis konnten in der Vergangenheit bereits den Vorteil dezentral gesteuerter Produktionen insbesondere hinsichtlich der Flexibilität, der Adaptivität und der Robustheit nachweisen. [Mou+12; Lei09]

Durch die Dezentralisierung und Modularisierung steigt die Relevanz der Kommunikation zwischen den Entitäten. Für den digitalen Informationsaustausch einer Steuerung mit seiner Umgebung muss ein gemeinsames Kommunikationsnetzwerk bestehen, über welches alle involvierten Teilnehmer erreichbar sind. Das beschränkt sich nicht nur auf die Produktion, sondern umfasst auch die relevanten übergeordneten IT-Systeme wie z.B. das Manufacturing Execution System (MES) oder das ERP-System. Die seit Industrie 4.0 verstärkte angestrebte gemeinsame Kommunikationsmöglichkeit zwischen allen Entitäten lässt sich als durchgängige digitale Vernetzung bezeichnen. Hierfür benötigen Steuerungen bzw. Module allerdings einheitliche informationstechnische Schnittstellen.

Im Gegensatz zu starren, sich über die Zeit nicht ändernden Konfigurationen ermöglicht eine dezentral gesteuerte und modulare Produktion zur Fertigung von kundenindividuellen Kleinstlosen eine stetige Anpassung an neue Anforderungen. Die im gemeinsamen Kommunikationsnetzwerk ausgetauschten Daten müssen für alle interpretierbar sein. Neben einer gemeinsamen Syntax, welche die Struktur regelt, muss es eine gemeinsame Semantik geben, welche die Bedeutung der Daten beschreibt. [Den12; Gra+16] In der Informatik sind unterschiedliche Modellarten zur Wissensrepräsentation entstanden, die sich hinsichtlich Komplexität und Ausdrucksstärke unterscheiden. In der einfachsten Form lassen sich Daten in einem Nachschlagewerk in Form eines Glossars darstellen, in der ausdrucksstärksten, komplexesten Version wird eine Ontologie verwendet, die zusätzlich logische Relationen zwischen Entitätstypen beschreiben kann (siehe auch [Den12; Dac+03]).

Zusammenfassend sind die Digitalisierung und die Modularisierung wichtige Paradigmen, um die Wandelbarkeit zu verbessern und den zukünftigen Anforderungen an die Produktion gerecht zu werden. Die durchgängige digitale Vernetzung erfordert eine einheitliche, maschinenlesbare Kommunikation und bedingt durch die Modularisierung steigt die Dezentralität und Relevanz der Schnittstellendefinition. Im Folgenden werden daher diese Themen vertiefend betrachtet.

2.1.2 Dezentrale Architekturen für die Produktionssteuerung

2.1.2.1 Definition und Eigenschaften von Architekturen

Der Architektur-Begriff ist zurückzuführen auf den lateinischen Begriff „architecture“, der Baukunst bedeutet. Aufgrund der vielfachen Anwendung in der Stadtplanung, im Bauwesen und in der Informatik ist der Begriff heute nicht einheitlich definiert. Grundsätzlich beschreibt

eine Architektur in allen Domänen eine Struktur zur Realisierung von Funktionalitäten, d.h. sie vereint zwei Sichten auf ein System: Zum einen beschreibt sie die für das Schaffen eines Mehrwerts, also eine Funktion, notwendigen Komponenten, zum anderen beschreibt sie deren Form, im Besonderen die Zusammenstellung der Komponenten. Die Architektur stellt im Wesentlichen die nicht-funktionale Qualität eines Systems dar. [Wik16a; Gol14; Kra+07; Hab+12]

Als eigener Typ von Architekturen beschreibt eine Referenzarchitektur eine Schablone, die in einer bestimmten Domäne ähnlich geartete Probleme löst und die Komplexität reduziert. Eine Referenzarchitektur beschreibt u.a. Systembestandteile, Beziehungen der Entitäten untereinander, Nomenklaturen sowie typische Entwicklungs- und Lösungsmuster. Die Beschreibung bleibt grundsätzlich abstrakt und technologieunabhängig. Sie sollte allerdings neben technischen Aspekten auch die betrachteten Anwendungsfälle und deren Einbettung in die Domäne adressieren. Sofern sich eine Referenzarchitektur nicht aus existierenden Architekturen und Implementierungen ableiten lässt, kann sie auch durch eine Referenzimplementierung oder einen Prototypen verifiziert werden. Obwohl in der Informatik ein einheitlicheres Verständnis für eine Referenzarchitektur besteht, ist der Begriff ähnlich wie der der Architektur in anderen Domänen nicht einheitlich definiert. [Gol14; Bro+12; Lin+15; Clo+09; Rat01]

(Referenz-)Architekturen lassen sich in zwei Typen unterteilen: Zum einen in Architekturen, die ein System beschreiben, und zum anderen in Architekturen, die eine Methode beschreiben. Erstere entsprechen der in der Praxis verbreiteteren Auffassung von Architekturen im Sinne einer Struktur. Letztere entsprechen eher Vorgehensmodellen zur Realisierung der erstgenannten Architekturen. [Kra+07] Eine gute Architektur ist u.a. dadurch gekennzeichnet, dass sie gegebene Rahmenbedingungen einhält, ein strategisches Ziel oder eine domänenspezifische Aufgabenstellung wirtschaftlich erreicht und mit wenig Aufwand erweiterbar bzw. anpassbar ist. Zusätzlich sollte sie für Dritte schnell erfassbar und in sich stimmig sein. [Hab+12] Im Rahmen dieser Arbeit ist eine Referenzarchitektur zu verstehen als:

Eine Referenzarchitektur ist eine technologieunabhängige Beschreibung, welche ein Problem in einer Domäne zielorientiert und möglichst allgemeingültig löst. Die Bestandteile einer Referenzarchitektur sind hierbei von der Domäne und Problemstellung abhängig und umfassen Strukturen oder Vorgehensweisen.

Auch wenn (Referenz-)Architekturen nur ein theoretisches Konstrukt sind, ist ihr Mehrwert unumstritten. Sie besitzen für die Entwicklung komplexer Systeme und die Digitalisierung der Produktion einen hohen Stellenwert. [Sen13; Rei+13; Pir+13] Architekturen helfen Entwicklern, relevante Aspekte des betrachteten Systems von nicht relevanten Dingen abzugrenzen, wodurch Architekturen die Komplexität reduzieren. Auch schafft eine Architektur ein gemeinsames Verständnis für eine Domäne und reduziert Risiken bei der Entwicklung durch das Vermitteln von Wissen und Erfahrung. Sie ist teils sogar zwingend notwendig, da sie den Kontext eines Systems beschreibt. Ein Beispiel hierfür ist die Definition von Modulen, welche ohne die Kenntnisse des Kontexts nicht möglich ist. [Lin+15; Mil+98]

2.1.2.2 Relevanz von dezentralen Architekturen

Die Automatisierungspyramide ist eine für die Beschreibung von automatisierten Fertigungslinien und Anlagen in der Produktion weitverbreitete Architektur. Sie beschreibt die Arten und Abhängigkeiten unterschiedlicher Komponenten in der Produktion und ordnet sie in eine Hierarchie ein. Auch wenn heute unterschiedliche Darstellungen hinsichtlich Anzahl und Abgrenzung der Ebenen in der Automatisierungspyramide existieren (siehe z.B. [Col+13; Züh+11; Reu+08]), ist der grundsätzliche Aufbau stets ähnlich und in Abbildung 3 dargestellt. Auf der untersten Ebene der Pyramide befindet sich eine hohe Anzahl von Aktoren und Sensoren. In der darüber liegenden Steuerungsebene befinden sich Steuerungskomponenten wie z.B. Speicherprogrammierbare Steuerungen (SPSen), welche mehrere Aktoren steuern oder mehrere Sensoren auslesen. Sie kontrollieren ganze Anlagen oder Anlagenabschnitte. Jedes Feldgerät ist in der Regel fest einer Steuerung zugeordnet. In den Ebenen darüber befindet sich eine geringe Anzahl unterschiedlicher Systeme, die anlagenübergreifend die Produktion oder das ganze Unternehmen verwalten. Beispiele hierfür sind MES zur Detailplanung und -steuerung von vorliegenden Aufträgen oder ERP-Systeme, die neben der Produktion auch Dienste für die Buchhaltung oder das Personalwesen bereitstellen und miteinander vernetzen.

Für die vorliegende Arbeit ist insbesondere die Steuerungsebene der Automatisierungspyramide mit ihren Schnittstellen zu übergeordneten Software-Systemen relevant, weshalb im Folgenden speziell Architekturen hierfür betrachtet werden. Eine Steuerungsarchitektur ist eine Architektur, die für die Domäne der Produktion den Zusammenhang zwischen kontrollierenden und ausführenden Instanzen beschreibt. Sie bezieht sich speziell auf den Zusammenhang zwischen Steuerungen, den Ablauf von Entscheidungen sowie die Verteilung

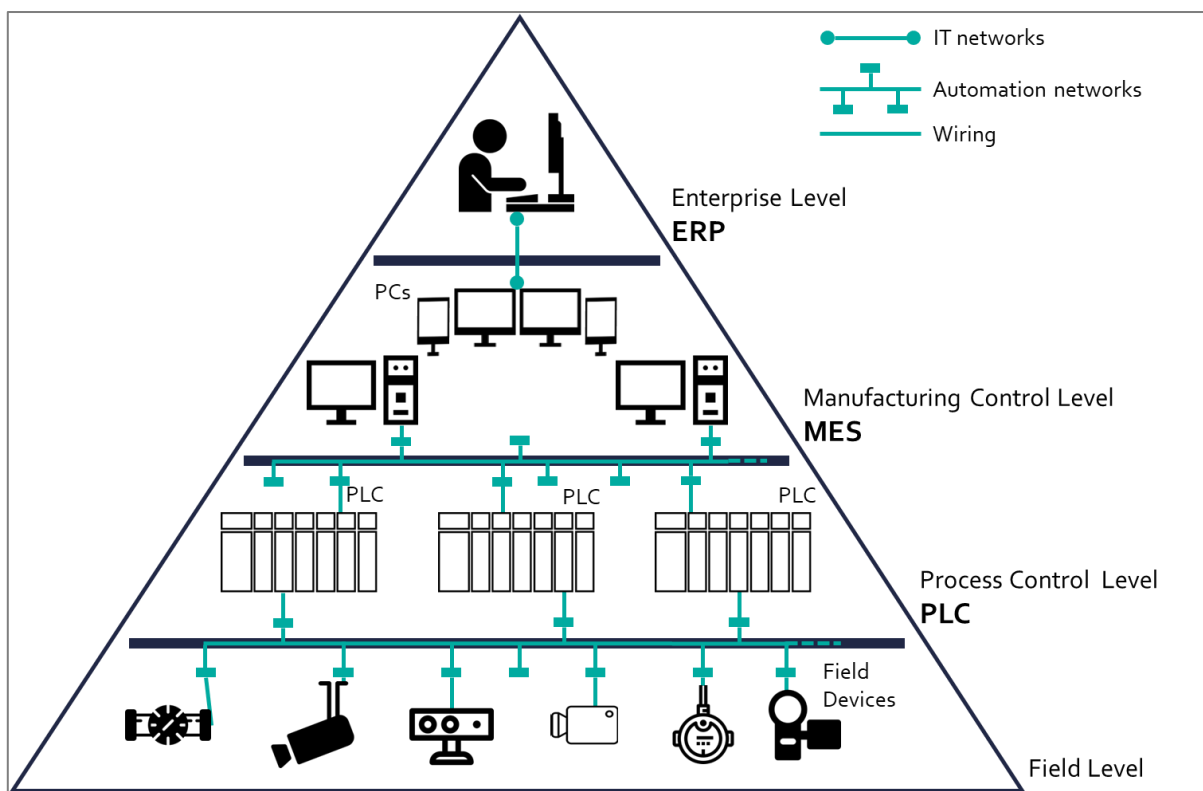


Abbildung 3 Automatisierungspyramide (Quelle: [Züh+11])

von Entscheidungskompetenzen. Steuerungsarchitekturen liefern den grundsätzlichen Rahmen für die spätere Implementierung der Hard- und Software. [Sch13b]

Hinsichtlich der Verteilung der Steuerungslogik in einem Netzwerk mit mehreren Steuerungen lassen sich drei Paradigmen unterscheiden. Je nach Sichtweise, ob aus Sicht der Systemstruktur oder der Steuerungsverteilung, sind unterschiedliche Bezeichnungen für die Architekturen entstanden. In hierarchischen Architekturen übernimmt die übergeordnete Instanz die Kontrolle der ihr untergebenen Einheiten. Durch die vertikale Zusammenfassung von Aufgaben und Verantwortlichkeiten mit steigender Ebene existiert in der Regel eine zentrale Instanz auf höchster Ebene, welche alle anderen ihr untergeordneten Komponenten direkt oder indirekt kontrolliert. Diese Architektur entspricht dem klassischen Ansatz zur Steuerung in der Produktion. Die Pyramidenform der Automatisierungspyramide visualisiert dies. Der Begriff „zentrale Steuerung“ ist ein Synonym zur hierarchischen Architektur. Das Gegenteil zu einer hierarchischen Architektur ist eine heterarchische Architektur. Hier entscheiden Steuerungen selbstständig und sind nicht weisungsgebunden an übergeordnete Systeme. Alle Steuerungen kontrollieren eigene Feldgeräte und bieten Funktionalitäten an. Durch die Kommunikation mit den gleichwertigen anderen Teilnehmern koordinieren sie Prozesse und handeln die Übernahme von Aufgaben aus. Aus Steuerungssicht wird diese Architektur auch als „autonome Steuerung“ bezeichnet. Beispiele für eine solche Architektur finden sich im nächsten Kapitel in den agentenbasierten Ansätzen. Zwischen diesen beiden Extremen finden sich semi-heterarchische Architekturen, welche auch als dezentrale Steuerungen bekannt sind. Sie kombinieren die Ansätze aus beiden Bereichen in unterschiedlicher Ausprägung miteinander. Abbildung 4 visualisiert die vorgestellten Paradigmen und Synonyme. [Sch13b; Sch+07; Sch16; Wel+11]

In Abhängigkeit vom Grad der Heterarchie ist bei dezentralen Steuerungen eine zentrale Fertigungsplanung und -optimierung mangels global verfügbarer Informationen nur bedingt möglich und das Verhalten einzelner Steuerungen sowie die Gesamtleistung der Produktion nur schwer prognostizierbar. Auch sind die Investitionskosten aufgrund der Redundanz von Steuerungen höher. U.a. mangels in der Produktion akzeptierter Standards und geringer

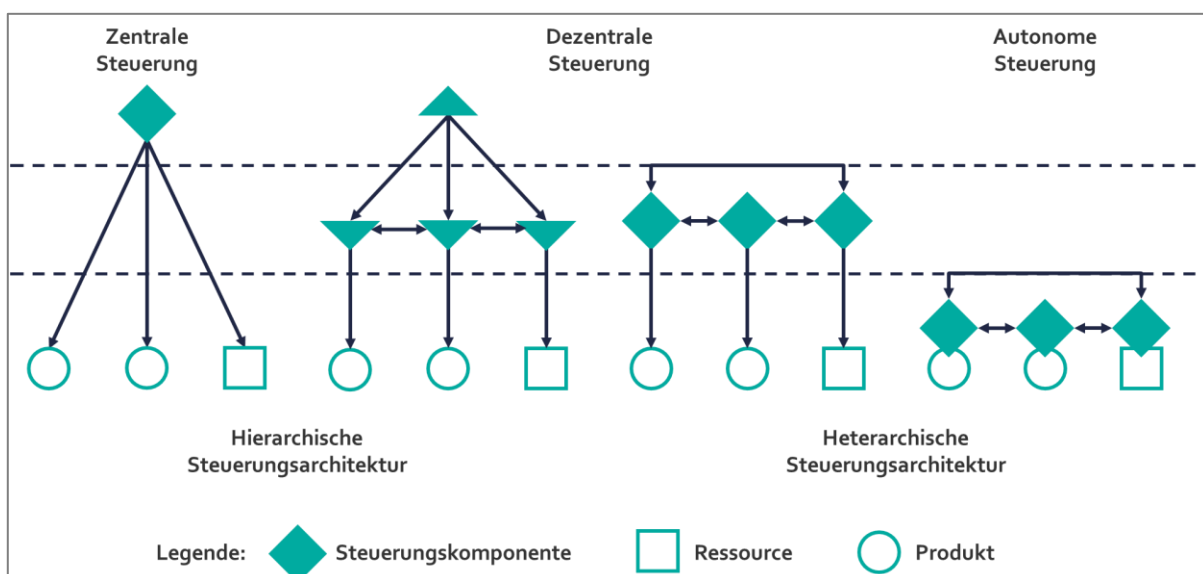


Abbildung 4 Vergleich von Paradigmen bei der Architekturgestaltung (Quelle: [Sch13b])

Interoperabilität zwischen den Herstellern konnten sich autonome Steuerungen bisher nicht etablieren. [Van+98; Col+05; Lei09]

Nichtsdestotrotz erfreuen sich autonome und dezentrale Steuerungen in Forschung und Praxis wachsender Beliebtheit. Gründe dafür sind die sinkenden Kosten, die damit verbundene bessere Rentabilität heterarchischer Architekturen, eine höhere Wandelbarkeit im Vergleich zu starren, zentralen Steuerungen und die lokale Rechenleistung für die Datenvorverarbeitung, um Engpässe in der Datenübertragung zu vermeiden. Sowohl die Wissenschaft als auch die Praxis konnten für autonome Systeme nachweisen, dass sie hinsichtlich Produktivität, Flexibilität, Robustheit und Implementierungsaufwand zentralen Systemen überlegen sind. [Van+98; Mou+12; Lei09; van96; Weg98; Val+94; Lei+08]

Mit der aktuell stattfindenden Industrie 4.0-Entwicklung und der angestrebten Modularisierung der Produktion verlagert sich zwangsläufig die Steuerungslogik von einer zentralen, übergeordneten Instanz auf mehrere Module. Die in den Modulen jeweils vorhandenen Steuerungskomponenten steuern nicht nur die internen Komponenten, sondern kapseln auch die Kommunikation nach extern. Ein lose gekoppeltes Netzwerk mit sich selbst organisierenden Entitäten löst langfristig die klassische Automatisierungspyramide ab. [Her17a; Gor+17; Vog+14] Die im nächsten Kapitel vorgestellten CPSe bieten einen mit Industrie 4.0 aufgekommenen, vielversprechenden Ansatz zur Realisierung dezentraler Steuerungsarchitekturen.

2.1.2.3 Cyber-Physische Systeme zur dezentralen Steuerung der Produktion

CPSe sind kostengünstige und leistungsstarke Systeme, welche mit ihrer physischen Umwelt interagieren und digital mit anderen Systemen kommunizieren. Sie wurden im Rahmen der Industrie 4.0-Entwicklung als Kerntechnologie für die dezentrale Steuerung proklamiert, da sie kostengünstig lokale Rechenleistung in die Produktion bringen. CPSe sollen nicht nur autonom handeln und sich selbst optimieren, sondern durch die Interaktion miteinander den Wandel von einer zentralen zu einer dezentralen Steuerung der Produktion unterstützen. [Bau14; Bun]

Nach [Lee08], welcher den Begriff CPSe prägte, und [Bro10b], welcher sich mit den Potenzialen von CPSe in der Produktion befasste, besteht ein CPS aus einem eingebetteten, leistungsstarken Kleinstrechner zur Datenverarbeitung. Mittels daran angeschlossener Aktoren und Sensoren kann es mit seiner physischen Umwelt interagieren. Über eine integrierte Kommunikationsschnittstelle tauscht es digital Informationen mit anderen Entitäten aus (vgl. Abbildung 5)

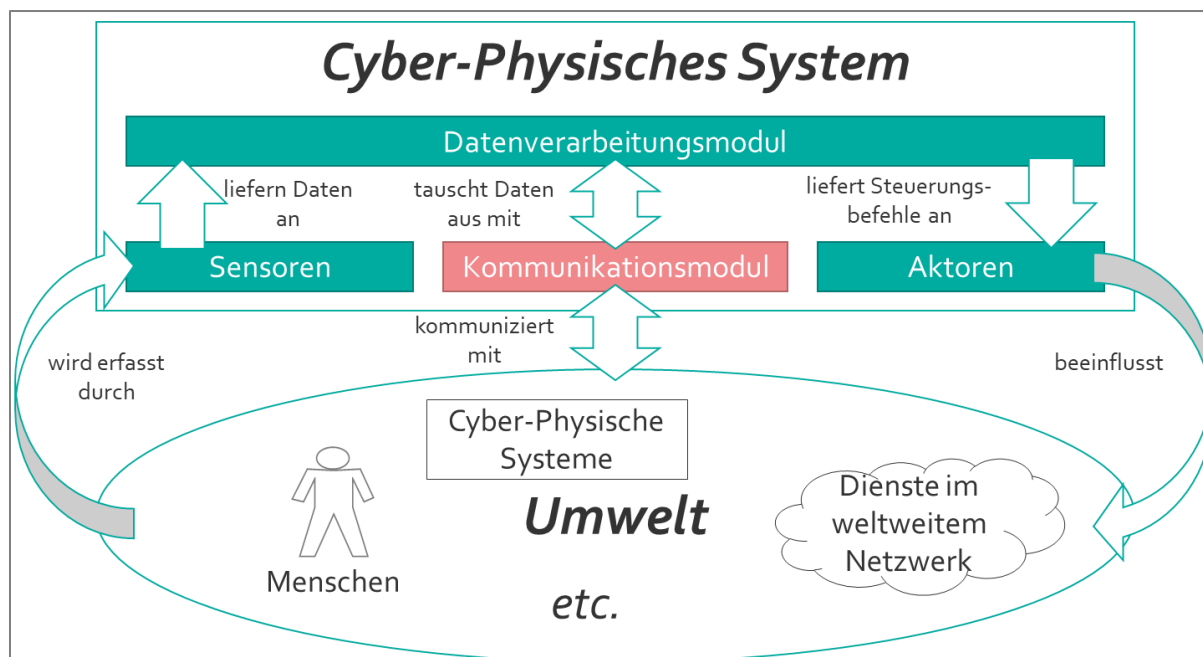


Abbildung 5 Schematische Darstellung eines CPSs (Quelle: [Kol13])

Als Weiterentwicklung von eingebetteten Systemen steht bei einem CPS der Austausch mit seiner physischen und digitalen Umgebung im Fokus. Dieser umfasst sowohl dem CPS bereits bekannte als auch neue Entitäten. Im Kontext der Produktion zählen zu den Entitäten andere CPSs, Anlagen und Menschen. Durch die leistungsstarke Recheneinheit kann es aufgenommene Informationen selbstständig interpretieren und Aktionen auslösen. Ein CPS wird aufgrund dieser Autonomie im allgemeinen Sprachgebrauch auch als „intelligent“ bezeichnet.

Bezogen auf die vorgenannte Automatisierungspyramide ist das primäre Einsatzgebiet von CPS die Steuerungsebene. Mit CPSen ausgestattete Anlagen werden auch als „intelligente Anlagen“ oder „Cyber-Physische Produktionssysteme“ betitelt (siehe z.B. [Pro]). Auch wenn klassische Speicherprogrammierbare Steuerungen (SPS) die Produktion heute noch dominieren, existieren bereits erste kommerzielle Produkte im Sinne eines CPSs. Beispiele sind in Abbildung 6 dargestellt und z.B. in [Rot16] beschrieben. Sie basieren auf offenen Betriebssystemen mit offenen Programmierschnittstellen, auf welchen auch moderne, objektorientierte Hoch-Programmiersprachen wie z.B. Java eingesetzt werden können, um Feldgeräte anzusprechen. Moderne SPSen verfolgen ebenfalls den Ansatz von CPSen, wodurch eine eindeutige Abgrenzung nicht immer möglich ist.

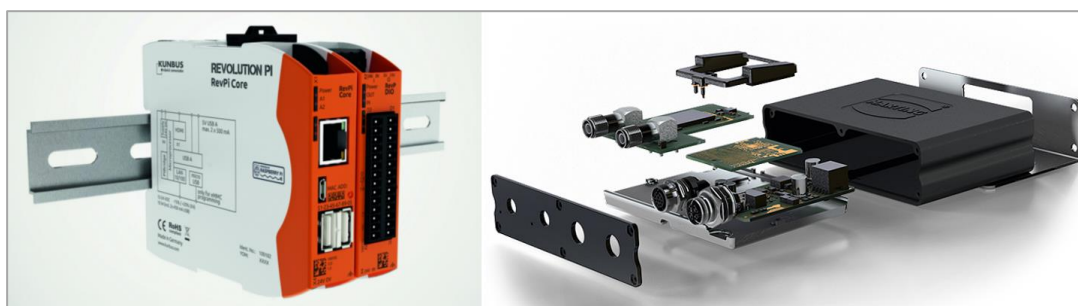


Abbildung 6 Beispiele für kommerzielle Produkte im Sinne von CPSen (links: KUNBUS Revolution Pi; rechts: Harting MICA; Quelle: [Han16; Klo15])

CPSe sind zusammenfassend eine vielversprechende Technologie, um flexibel programmierbare Rechenleistung in der Produktion zu ermöglichen. Aufgrund der Vielzahl der Schnittstellen können sie auch bestehende Betriebsmittel und Anlagen nachträglich an das durch die Digitalisierung angestrebte durchgängige Netzwerk anbinden.

2.1.2.4 Bestehende dezentrale Architekturen zur Produktionssteuerung

Wie zuvor erwähnt, erfreuen sich dezentrale Architekturen in der Produktion u.a. aufgrund ihres Vorteils in komplexen und dynamischen Umgebungen wachsender Beliebtheit. Sie werden nicht erst seit dem Aufkommen von CPSen und Industrie 4.0 erforscht. Erste Ansätze finden sich bereits in den 1980er-Jahren, wobei in der Zeit vor und nach der Jahrtausendwende intensiv an dem Thema geforscht wurde. Eine Auswahl für den Kontext dieser Arbeit relevanter Architekturen und Begrifflichkeiten wird im Folgenden vorgestellt. Der Fokus liegt auf der für diese Arbeit relevanten Rollenverteilung von Steuerungen und Schnittstellen. Weitere Informationen zur historischen Entwicklung dezentraler Architekturen in der Produktion finden sich u.a. in [van96], [Lei09], [Sch+07] und [Sch13b].

Im Bereich der künstlichen Intelligenz entstand der Ansatz der Multiagentensysteme (MAS). Autonome physische und logische Entitäten, sogenannte Agenten, interagieren in MAS miteinander, um parallel ein gemeinsames Ziel zu erreichen, das die Agenten alleine nicht erreichen können. Diese softwaregetriebenen Agenten ermöglichen die Realisierung von heterarchischen Steuerungsarchitekturen in der Produktion. Sie fördern die Komplexitätsreduzierung durch Funktionskapselung und unterstützen eine hohe Wiederverwendbarkeit von Softwarecode. [Göh13; Lei09] Im Fokus von Architekturüberlegungen bei MAS stehen weniger feste Strukturen und Abhängigkeiten, sondern vielmehr die Definition von Agentenrollen, deren Verantwortlichkeiten, Verhaltensarten und -weisen. Analyse- und Entwurfsmethoden wie beispielsweise das Multiagent Systems Engineering (MaSE) leiten aus einem übergeordneten Ziel Anwendungsfälle ab, aus denen sich wiederum Agentenklassen und Nachrichtenaustausche ergeben. Erst im letzten Schritt einer jeden Iteration werden Rahmenbedingungen wie der physische Ort des Agenten und die maximal zulässige Anzahl bestimmter Agenten je Klasse definiert. Agenten werden dabei beispielsweise durch die Wahrnehmungsinhalte, Aktionen, Ziele und die Umwelt (engl. *percepts, action, goals, environments; PAGE*) oder Wissen über die Umwelt, Zielzustände und Absichten (engl. *beliefs, desires, intentions; BDI*) beschrieben. Grundsätzlich lassen sich drei Arten von Verhaltensweisen unterscheiden: Reaktive Agenten reagieren anhand fester Regeln auf Eingangswerte; logikbasierte Agenten treffen auf Basis eines Modells der Umwelt logische Schlussfolgerungen für ihr Handeln zur Zielerreichung; BDI-Agenten verfügen über ein Modell ihrer Umwelt und die Fähigkeit, selbst Ziele festzulegen und zu verfolgen. [Wei+06; Lei09; Los13] Beispiele für bestehende Projekte und Frameworks für MAS in der Produktion sind in [She+06] aufgelistet. Ein weiterer Ansatz zur Dezentralisierung der Produktion sind holonische Fertigungssysteme (engl. *Holonc Manufacturing Systems; HMS*). Der Begriff „Holon“, griechisch für „Teil eines Ganzen“, ist zurückzuführen auf [Koe89] und beschreibt das kleinste Modellierungselement eines Systems mit festgelegten Rollen und Aufgaben. HMS wurden insbesondere im Produktionsumfeld während der CIM-Ära erforscht. [Sch13b] Auch wenn in einem HMS die autonomen Holonen miteinander kooperieren und zur Zielerreichung aufeinander angewiesen sind,

sind HMS im Gegensatz zu MAS nicht vollständig dezentral, sondern hierarchisch strukturiert. Die Holonen können im Gegensatz zu klassischen Hierarchien allerdings unterschiedlichen HMS angehören. [Lei09] Bekannte Referenzarchitekturen wie z.B. PROSA, ADACOR oder GRACE basieren immer auf mindestens drei Grundtypen von Holonen, die in der Produktion miteinander interagieren (siehe z.B. Abbildung 7): Der Produkt-Holon enthält das Prozess- und Produktwissen und beschreibt das zu fertigende Produkt; der Auftrags-Holon repräsentiert die Arbeitsschritte zur Fertigung eines konkreten Produktes; der Ressourcen-Holon hingegen repräsentiert vorhandene Fertigungsressourcen. Teilweise ist ein Unterstützung-Holon vorhanden, der übergreifende Hintergrundaktivitäten übernimmt. Ein Holon kann sowohl eine physische Entität als auch Software sein. Durch die Verwendung von Holon-Typen besteht trotz einer inneren Heterogenität eine Ähnlichkeit nach außen. [Van+98; Göh13]

Beide Ansätze, MAS und HMS, sind nicht gegensätzlich. Bei MAS handelt es sich um ein Konzept und eine Technologie, bei HMS nur um ein Konzept. So können z.B. Agenten ein HMS realisieren. [Lei09] Neben diesen beiden in der Forschung recht bekannten Ansätzen finden sich unter Begriffen wie „Flexible Manufacturing Systems“ (FMS), „Reconfigurable Manufacturing Systems“ (RMS), „Biological Manufacturing Systems“ (BMS), „Fraktale Fabrik“ oder „Agile Manufacturing Systems“ (AMS) weitere Ansätze. Sie übertragen unterschiedliche Paradigmen aus anderen Domänen auf verteilte Steuerungsarchitekturen, um die Wandelbarkeit der Fertigung zu erhöhen. Eine klare Abgrenzung ist hierbei nicht immer möglich. Einen Überblick und Vergleich dieser Ansätze findet sich z.B. in [Sch13b], [Lei09] oder [Sch+07]. MAS, HMS und ähnliche Ansätze beschreiben zusammenfassend aus der Struktursicht einen Sachverhalt. Sie geben keine unmittelbare Aussage über ablaufende Prozesse bzw. wie sich bestehende Produktionsprozesse in Agenten abbilden lassen.

Für den Einsatz von CPSen in der Produktion existieren in der Wissenschaft ebenfalls erste Architekturen. Speziell die Verteilung von Funktionalitäten auf unterschiedliche CPSen, die Ad-hoc-Vernetzung mit anderen Geräten, die Verbindung der physischen mit der Cyber-Welt und sich stetig ändernde Umgebungen sind Herausforderungen, die ein CPS bewältigen muss. Bestehende Referenzarchitekturen aus anderen Bereichen berücksichtigen das bisher nicht ausreichend. [Lee08; Rei+13; Kag+13] Nachstehende Beschreibung stellt eine Auswahl

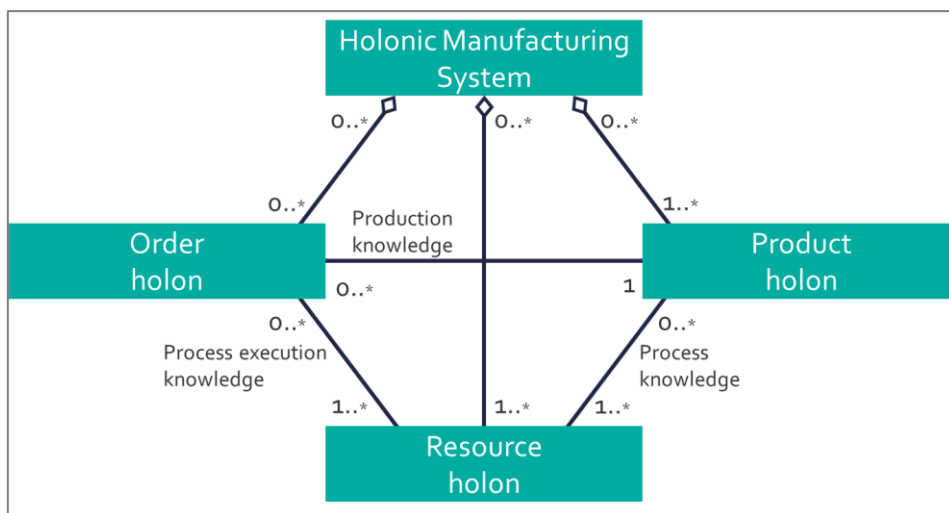


Abbildung 7 Holon-Typen und Zusammenhänge in PROSA (Quelle: [Van+98])

bestehender CPS-Architekturen für die Produktion vor. U.a. [Kol13] oder [Hu+12] beschreiben und vergleichen darüber hinaus weitere Ansätze.

[Lew+13] erarbeiteten eine auf MAS basierende Architektur für CPSe in der Intralogistik, welche sie in einem Demonstrator der Handhabungstechnik evaluierten. Ihr Ansatz unterteilt drei Ebenen in Abhängigkeit von der Rolle des Agenten: Im User-Interaction-Layer befinden sich die Agenten zur Administration des Systems und Interaktion mit dem Menschen via einer Benutzungsoberfläche; Agenten des Transport-Agent-Layer verfolgen das Ziel, Waren zu einem Zielort zu bewegen; und alle Agenten zur Kontrolle von Hardware-Komponenten sind im Hardware-Agent-Layer zusammengefasst. Neben diesen Agentenarten existieren noch zwei übergreifende Agenten, welche Aktivitäten protokollieren und Standorte von Entitäten überwachen. Abbildung 8 gibt einen Überblick über die Aufteilung.

[Vog+15] stellen einen ähnlichen Ansatz für den Aufbau cyber-physischer Produktionssysteme vor. Die auf MAS basierende Architektur unterteilt vier Ebenen: Die Management-Ebene stellt die Vernetzung sicher, die Planning-Ebene verwaltet die Aufträge, die Scheduling-Ebene repräsentiert Agenten mit Produkten und in der Execution-Ebene sind Agenten der Produktionsressourcen gesammelt. Des Weiteren verantworten globale Agenten die Kommunikation.

Im Rahmen des Forschungsprojektes Cyber-Physische Produktionssysteme (CyProS) entstand ein Rahmenkonzept zur Integration von CPSen in bestehende Produktionsumgebungen. Das Rahmenkonzept umfasst ein Vorgehensmodell für das Engineering von CPS-basierten Lösungen, Bausteine zur Beschreibung des Aufbaus von CPSen sowie unterschiedliche CPS-Rollen und deren Eigenschaften. Die CPS-basierte Fabrik besteht aus einem Cyber-Physischen Produktionssystem (CPPS) zur Herstellung von Produkten, einem Cyber-Physischen Transportsystem (CPTS) für deren Transport, einer Cyber-Physischen Infrastruktur (CPI) zur Versorgung und Bereitstellung der Umgebung, einem Cyber-Physischen Produkt (CPP) als Resultat der Produktion sowie dem Produktionsplanungssystem (PPL). Abgesehen vom PPL, welches ausschließlich ein Softwaresystem ist, sind alle Rollen vom Typ eines CPSs. Abbildung 9 gibt einen Überblick über die Beziehungen zwischen den Rollen. Die aufeinander abgestimmten Rollen, Bausteine und das Vorgehensmodell im Rahmenkonzept unterstützen Entwickler bei der Lösung von Problemen mittels CPS-basierter Technologien. [Kol+17]

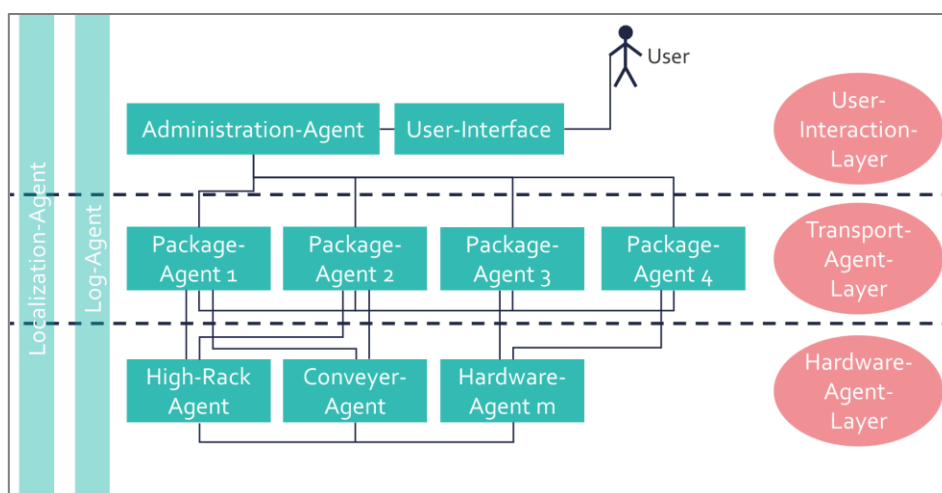


Abbildung 8 MAS-basierte CPS-Architektur für die Intralogistik (Quelle: [Lew+13])

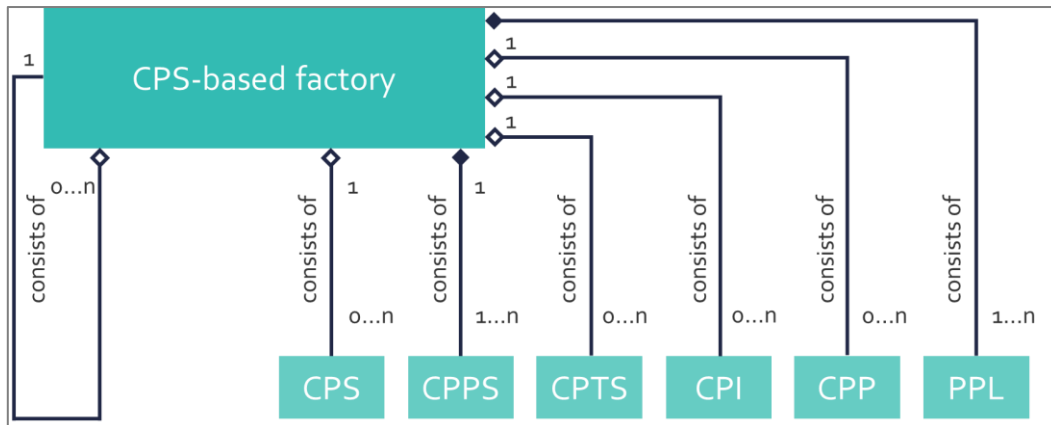


Abbildung 9 Rollen und Beziehungen von CPSen in der Produktion (Quelle: [Kol+17])

Neben diesen zwei Beispielen existieren weitere CPS-Architekturen, welche nicht speziell für die Produktion entworfen wurden. Ähnlich wie der Ansatz von [Lew+13] basiert [San+12] auf MAS. Die Ansätze von [Hu+12], [La+10] und [Lin+10] hingegen verwenden das Paradigma einer serviceorientierten Architektur (SoA). Eine SoA verfolgt das Ziel, durch definierte Schnittstellen und der Kapselung von Funktionalitäten die Austauschbarkeit, Wiederverwendbarkeit und Vernetzbarkeit von Anwendungen herzustellen. Softwareanwendungen, sogenannte Dienste, können zur Realisierung von Geschäftsprozessen lose miteinander gekoppelt werden. Eine SoA besitzt zur Verwaltung und Identifikation vorhandener Dienste einen Verzeichnisdienst, in dem sich alle Dienste mit ihrer Adresse und einer Beschreibung registrieren. Dienste können sich gegenseitig über diesen Verzeichnisdienst finden und gemeinsam Prozesse realisieren. [Mel10] Im Gegensatz zu MAS sind die vorgestellten CPS-Architekturen konkreter, da sie die Integration mittels CPS beschreiben. Allerdings gehen die bestehenden Architekturen ebenfalls nicht auf die Prozesssicht für Produktionsabläufe ein und geben keinen Hinweis, wie bestehende Produktionsumgebungen zu migrieren sind.

Eine neuere Architektur zur Realisierung von dezentralen Fertigungslinien ist die in der Technologie-Initiative SmartFactory KL e.V. entstandene SmartFactoryKL Systemarchitektur. Sie verfolgt das Ziel, die mechatronische Wandelbarkeit zu verbessern, eine individualisierte Massenproduktion zu ermöglichen und eine durchgängige Vernetzung der beteiligten Komponenten herzustellen. Hierzu unterteilt sie die im Kontext einer vollautomatischen Fertigungslinie vorhandenen Komponenten in fünf Ebenen (siehe Abbildung 10). Auf der Produktschicht befindet sich das zu fertigende Produkt, welches mittels eines integrierten Produktgedächtnisses den Prozess steuert und kurzfristige Änderungen an der Produktkonfiguration in den Fertigungsprozess übermitteln kann. Die Produktionsschicht fasst alle Fertigungsmodule zusammen. Für die angestrebte Wandelbarkeit müssen diese kurzfristig austauschbar sein und dedizierte Aufgaben übernehmen. Die darüber liegende Versorgungsschicht versorgt die Module mit allen notwendigen Medien und muss ebenfalls modular sein. Die Integrationsschicht bildet die Schnittstelle zwischen den Fertigungsmodulen und den IT-Systemen. Sie soll einen Echtzeit-Datenaustausch und die durchgängige Verfügbarkeit von Informationen sicherstellen. Hierzu verfügt sie über offene Schnittstellen zur losen Kopplung der Systeme. Die Schnittstellen bieten einheitlich die relevanten Parameter aus den Fertigungsmodulen an und ermöglichen sowohl lesenden als auch schreibenden Zugriff darauf. Die IT-Systemschicht bildet schlussendlich übergeordnete Systeme wie das ERP-System ab.

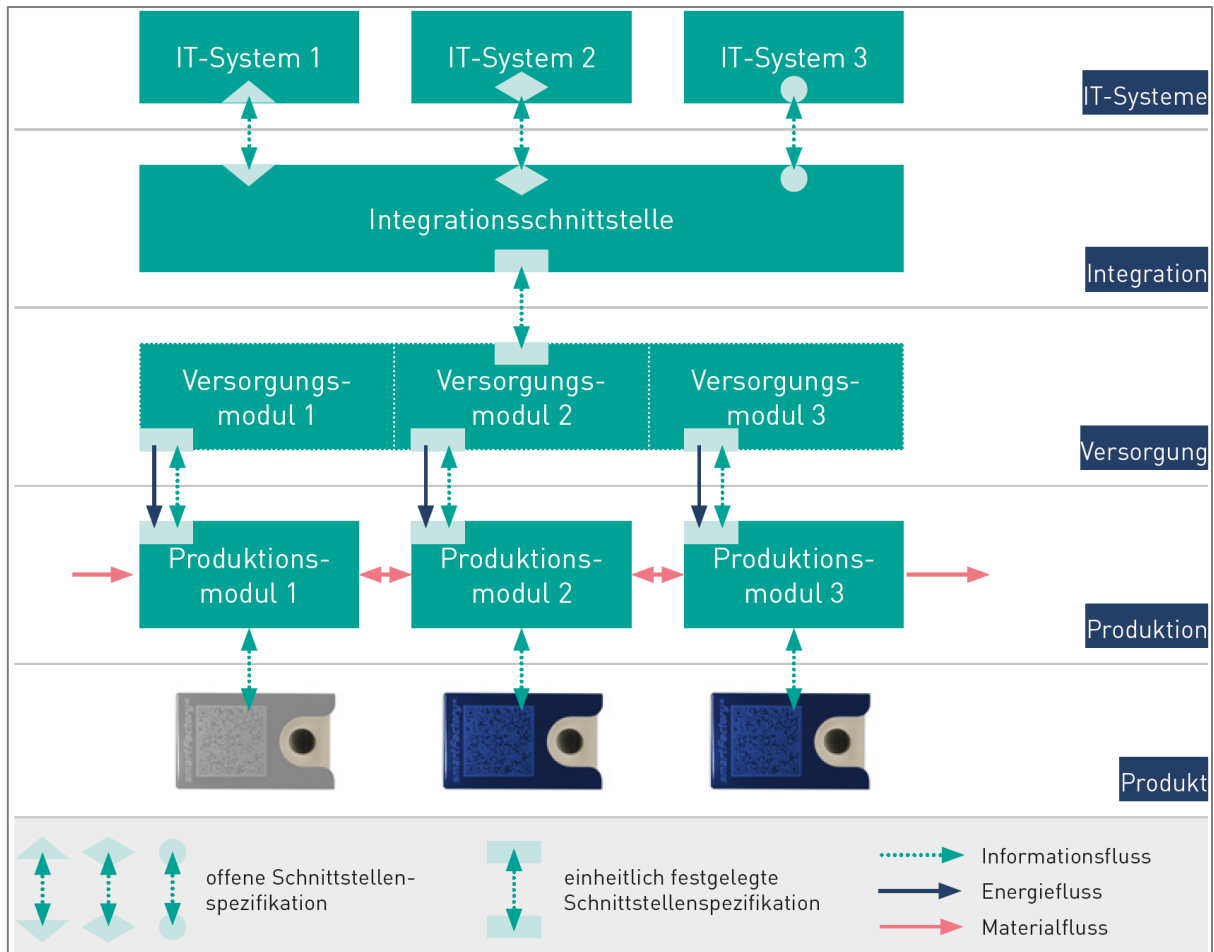


Abbildung 10 Die SmartFactoryKL Systemarchitektur (Quelle: [Gor+16a])

Kennzeichnend für diese Softwaresysteme sind eine schnelle Anbindung mit geringem Aufwand sowie eine Kapselung der Funktionalitäten. Für den Betrieb muss der IT-Systemsicht ein digitales Abbild der physisch stattfindenden Prozesse vorliegen. [Gor+16a; Wey+17; Gor+16b] Die SmartFactoryKL Systemarchitektur beschreibt nicht nur die Struktursicht, sondern spezifiziert auch Schnittstellen für die mechanische, elektrotechnische und informationstechnische Integration. Sie ist dadurch praxisnah für die Implementierung, berücksichtigt aber nicht die Rolle des Menschen und der Logistik. Ferner erfordert sie ein Produktgedächtnis, was nicht immer gegeben ist.

Die Plattform Industrie 4.0 ist eine Vereinigung von u.a. Unternehmen, Verbänden, Wissenschaft und Politik und strebt an, Deutschland als führenden Ausrüster für und Nutzer von neuartigen digitalen Technologien zu positionieren. Eines der Arbeitsergebnisse zur Standardisierung der Technologien ist das Referenzarchitekturmodell für Industrie 4.0 (RAMI 4.0). Das darin beschriebene Klassifizierungssystem bietet die Möglichkeit, eine Technologie bzw. Problemstellung in die Dimensionen Hierarchielevel (in Anlehnung an IEC 61512/ 62264), Lebenszyklus (in Anlehnung an IEC 62890) und Abstraktionsebene einzuordnen. Darüber hinaus beschreibt RAMI 4.0 mit der Industrie 4.0-Komponente und der dazugehörigen Verwaltungsschale eine Architektur zur Einbindung von physischen Entitäten in eine gemeinsame Kommunikationsumgebung (siehe Abbildung 11). Jede Entität wie z.B. ein Produkt, eine Anlage oder ein Feldgerät, besitzt eine virtuelle Beschreibung. Die sogenannte Verwaltungsschale fasst die Beschreibung in einer standardisierten Notation zusammen.

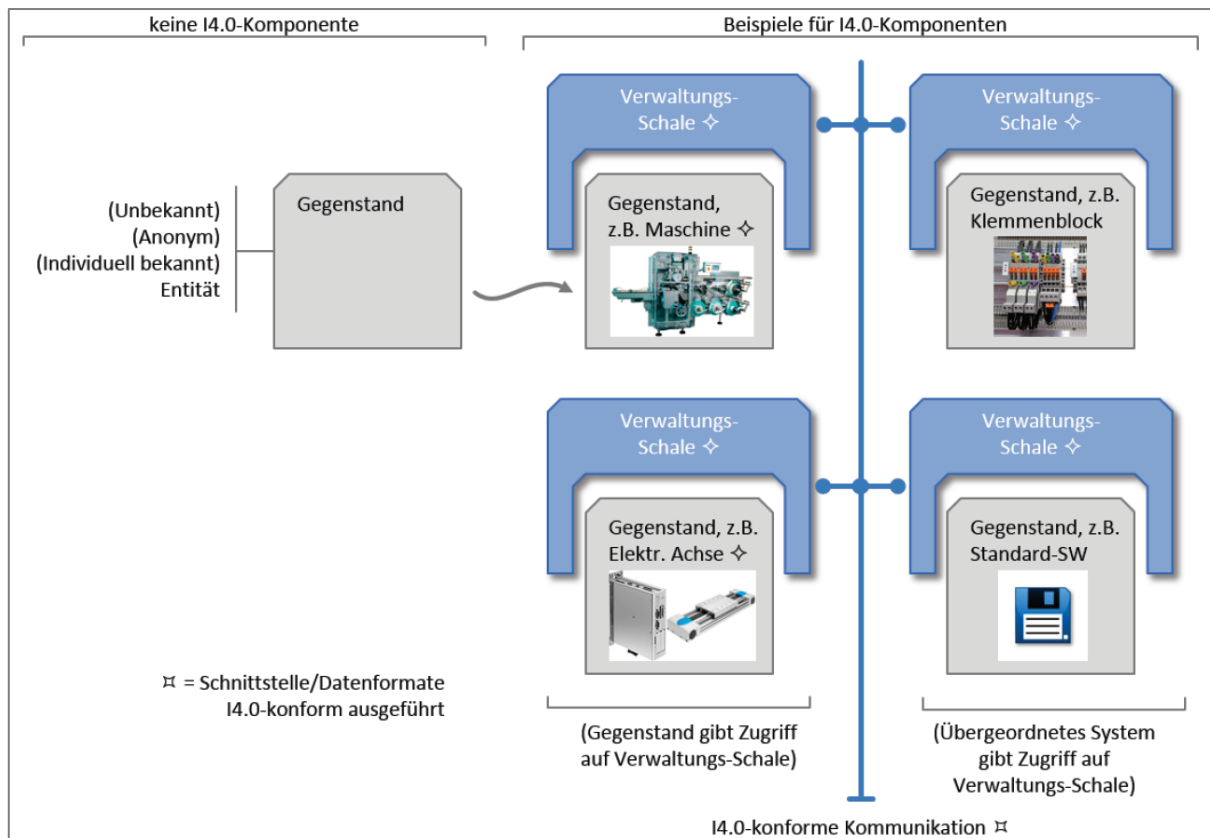


Abbildung 11 RAMI 4.0-Architektur mit I4.0-Komponente und Verwaltungsschale [Ado+15]

men und ist die Schnittstelle zu einem gemeinsamen Kommunikationsnetzwerk. Dritte können die Informationen ohne aufwendige Konfiguration und Programmierung auslesen und interpretieren. Hierfür stellt die Verwaltungsschale sogenannte Sichten zur Verfügung, welche für unterschiedliche Anwendungsfälle die hinterlegten Merkmale bündeln. Die physische Entität und die Verwaltungsschale bilden zusammen die Industrie 4.0-Komponente. [Bun16; Ado+15] Sowohl RAMI 4.0 als auch die Verwaltungsschale befinden sich aktuell noch in der Entwicklung und Evaluierung und lassen keine Aussagen zur Integration in Prozesse zu.

Auch das Industrial Internet Consortium, das amerikanische Pendant zur Plattform Industrie 4.0, erarbeitete eine Referenzarchitektur. Die Industrial Internet Reference Architecture (IIRA) ist ein offener, auf Standards basierender Ansatz, welcher generisch auf hoher Abstraktionsebene branchenübergreifende Gemeinsamkeiten und Anwendungsfälle beschreibt. Sie definiert Sichtweisen für unterschiedliche Anwender und deren Informationsbedürfnisse, gibt darüber hinaus allerdings keine Hinweise zu Rollen, strukturellen Zusammenhängen oder Schnittstellen zwischen Entitäten der Produktion. [Lin+15] Obwohl IIRA disziplinübergreifende Themen adressiert, ist die Prozesssicht nur auf einem abstrakten Niveau berücksichtigt.

Als abschließende Bewertung ist festzuhalten, dass sich die aus der Forschung kommenden agentenbasierten und holonischen Ansätze mangels Reproduzierbarkeit der Prozesse, Echtzeitfähigkeit und für die industrielle Praxis verfügbarer Entwicklungswerkzeuge nicht durchsetzen konnten [Lei09]. Dezentrale Architekturen wie die des Forschungsprojektes CyProS hingegen fokussieren die Struktursicht und lassen die Prozesssicht weitestgehend außen vor. Sie beschreiben Abhängigkeiten zwischen den in der Produktion Beteiligten, geben aber keine Auskunft über z.B. die Abläufe zur Auftragsabwicklung oder Fehlerbehandlung. Trotz

des hohen Stellenwerts von Schnittstellen in modularen und dezentralen Umgebungen definieren sie nicht, welche Informationen zwischen den Entitäten auszutauschen sind. Praxisgetriebene Ansätze wie die SmartFactoryKL Systemarchitektur, RAMI 4.0 oder IIRA befassen sich mit der Schnittstellendefinition. Sie befinden sich allerdings noch in der Entwicklung oder erfordern in der Produktion nicht immer vorhandene Voraussetzungen wie ein Produktgedächtnis. Fast alle Architekturen liefern darüber hinaus keine Beschreibung, wie bestehende Umgebungen zu migrieren sind.

2.1.3 Informationstechnische Schnittstellen und Kommunikation

2.1.3.1 Definition informationstechnischer Schnittstellen und Kommunikation

Für die Realisierung dezentraler Architekturen bedarf es einheitlicher informationstechnischer Schnittstellen zur Kommunikation zwischen den Entitäten. Eine Schnittstelle beschreibt den Berührungspunkt einer Entität mit seiner Umwelt. Schnittstellen lassen sich nach der Art der Interaktion unterscheiden, z.B. kann eine mechanische Schnittstelle die physische Verbindung zwischen zwei Anlagen herstellen, eine funktionale Schnittstelle von einer Software bei einer anderen Software aufrufbare Funktionen beschreiben und eine elektrische Schnittstelle elektrische Signale austauschen. Die Beschreibung von Schnittstellen ist eine wichtige Voraussetzung für die Umsetzung von Modularisierung. [Mil+98; Bib16; Lac+] Im Kontext dieser Arbeit ist eine informationstechnische Schnittstelle definiert als:

Eine informationstechnische Schnittstelle ist ein Interaktionspunkt der Informations- und Datenverarbeitung einer Entität (z.B. einer Arbeitsstation) mit seiner Umwelt (z.B. andere Arbeitsstationen oder Softwaresysteme). Die informationstechnische Schnittstelle ermöglicht den Austausch von Informationen und den Aufruf von Diensten durch Dritte an der anbietenden Entität.

Eine Schnittstelle nimmt die Rolle eines Vermittlers zwischen Abläufen bei Dritten und Abläufen innerhalb der Entität ein. Solange die Schnittstelle unverändert bleibt, können Abläufe innerhalb der Entität ohne Auswirkung auf Dritte beliebig geändert werden. Sie kapselt die internen Funktionalitäten und besteht sowohl aus Hardware als auch aus Software. Bezüglich der internen Architektur einer Entität werden informationstechnische Schnittstellen üblicherweise einer eigenen Ebene bzw. einem eigenen Bereich in der Entität zugeordnet. Auf der untersten Schicht steuern das Betriebssystem und die internen Applikationen die Hardware der Entität an. Eine optionale Middleware verbirgt die Heterogenität der untersten Ebene, bildet Bausteine für die interne Wiederverwendbarkeit und bzw. oder ermöglicht durch beispielsweise intuitivere Programmiermodelle eine einfachere Entwicklung. An diese Ebene setzt die Schnittstellen-Ebene an, die die erwähnte Verbindung interner Funktionalitäten mit der Umwelt herstellt. [Cou+05] Abbildung 12 zeigt ein Beispiel einer Kommunikationsschnittstelle aus einer holonischen Architektur.

Grundlage der Kommunikation zwischen informationstechnischen Schnittstellen und der Umwelt ist ein Kommunikationsprotokoll, das den Austausch von Nachrichten standardisiert und den digitalen Nachrichtenaustausch zwischen Mensch und Maschine sowie insbesondere Maschine und Maschine ermöglicht. Die International Telecommunication Union (ITU),

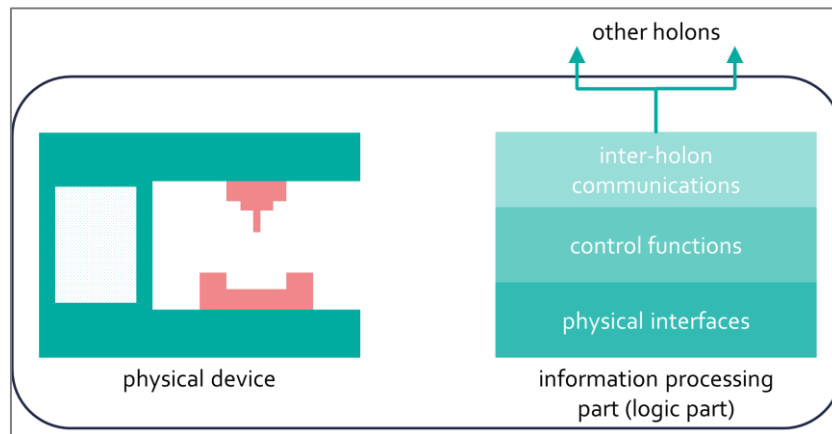


Abbildung 12 Beispiel für die Umsetzung einer informationstechnischen Schnittstelle in agentenbasierten bzw. holonischen Architekturen (Quelle: [Lei09])

welche den internationalen Betrieb von Telekommunikationsnetzen und -diensten koordiniert, definiert ein Kommunikationsprotokoll als eine Sammlung von Regeln und syntaktischen sowie semantischen Formaten, welche das Kommunikationsverhalten von Entitäten untereinander für die Ausführung von Aufgaben regelt [ITU14].

Die syntaktischen Formate beschreiben, wie einzelne Symbole bzw. Zeichen zu Gruppen zusammenzufassen sind. Erst durch die Syntax werden aus übertragenen Signalen Daten. Die semantischen Formate dienen dazu, Daten eine Bedeutung zu geben. Hierdurch ist es den bei der Kommunikation beteiligten Partnern möglich, die empfangenen Signale als Informationen zu interpretieren. [Den12] Im Kontext der zuvor beschriebenen dezentral gesteuerten Produktionen, in denen sich nicht nur die Art und Menge an Teilnehmern ändern kann, sondern auch unbekannte Teilnehmer hinzukommen können, ist eine eindeutige und einheitliche Kommunikation zwingend notwendig.

Die bei der digitalen Interaktion zwischen Entitäten eingesetzten Protokolle lassen sich in unterschiedliche Ebenen einteilen, welche aufeinander aufbauen. Das international standardisierte Open Systems Interconnection Modell (ISO/OSI-Modell) beschreibt auf sieben Ebenen – ausgehend von der Bitübertragung über die Absicherung und Vermittlung bis hin zur Anwendung – die Zusammenhänge und einsetzbaren Protokolle. Eine Übersicht ist in Abbildung 13 dargestellt, wobei für Details auf [Sch06], [Cou+05] oder [Wik16c] verwiesen sei. Im Rahmen dieser Arbeit sind hierbei ausschließlich die Protokolle der sechsten und siebten Schicht relevant, welche sich mit der Syntax und Semantik der übertragenen Daten befassen. Kapitel 2.1.3.3 stellt einen Ausschnitt der Vielzahl an im industriellen Kontext vorhandenen Kommunikationsprotokollen vor.

2.1.3.2 Methoden der Interprozesskommunikation

Die digitale Kommunikation lässt sich hinsichtlich der Netzwerktopologie und der Interprozesskommunikation unterscheiden. Da die Netzwerktopologie für diese Arbeit eine untergeordnete Rolle spielt und die Sterntopologie für die Anbindung von IT-Systemen und Anlagensteuerung der Status quo ist, wird hier nicht weiter auf sie eingegangen. Eine vollständiger Beschreibung von Topologien im Kontext der Produktion findet sich beispielsweise in [Wel+11] oder [Sch06].

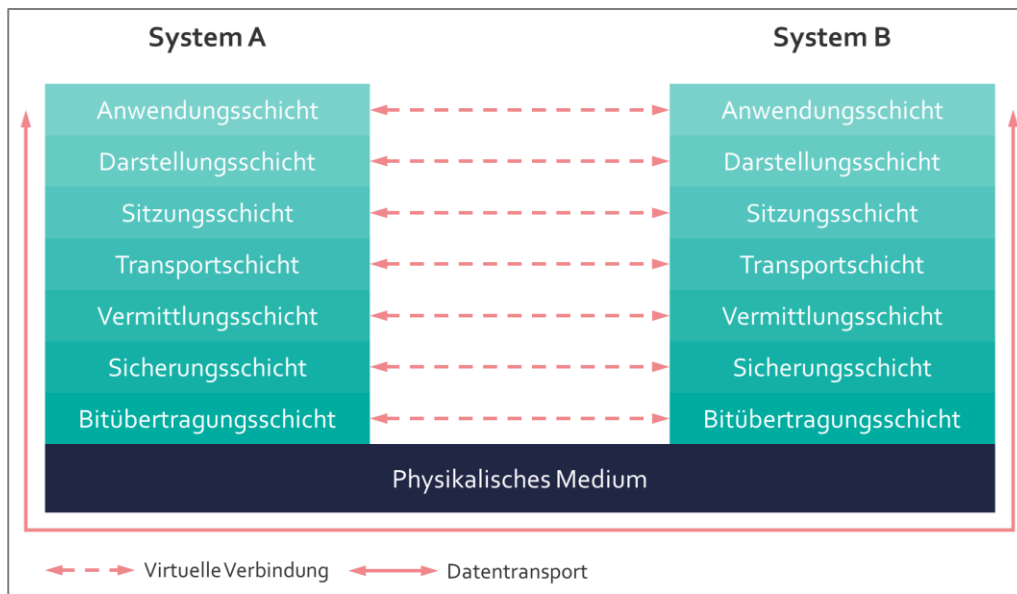


Abbildung 13 Die sieben Schichten des ISO/OSI-Modells (Quelle: [Sel10])

Neben der Topologie und den Protokollen ist bei der dezentralen Steuerung insbesondere die Interprozesskommunikation relevant. Sie beschreibt auf den höheren Schichten des ISO/OSI-Modells den Ablauf des Informationsaustausches zwischen Prozessen und Systemen. Beim Informationsaustausch zwischen zwei Systemen lassen sich die drei nachstehend beschriebenen Methoden unterscheiden [Jed16].

Bei der Request-Reply-Methode geht die Initiative für den Informationsaustausch vom Nachrichtempfänger aus. Dieser greift auf einen Dienst des Nachrichtensenders zu und führt ihn aus. Hierzu muss der Empfänger vom Sender wissen, unter welcher Adresse er erreichbar ist und welche Informationen er dem Dienst übergeben kann bzw. von ihm zurückbekommt. Änderungen nur an dem Sender oder nur am Empfänger sowie kurzfristiges Hinzufügen neuer Empfänger sind nicht möglich bzw. erfordern größere Anpassungen. Es handelt sich somit um eine starke Kopplung zwischen Sender und Empfänger. Beispiele für diese Art der Kommunikation sind das zyklische Abrufen von Informationen eines MES von SPSEN einer Fertigungslinie oder die Remote Method Invocation (RMI) in der Programmiersprache Java, die den Aufruf von Java-Methoden auf anderen Rechnern ermöglicht. Auch Webservices, die unter einem festen Identifikator (engl. Unified Resource Identifier; URI) allen Teilnehmern eine Information bereitstellen, funktionieren nach diesem Prinzip. Moderne Internetplattformen wie z.B. Facebook oder Twitter verwenden sogenannte Application Programming Interfaces (dt. Programmierschnittstellen; API), die Drittanwendungen einen Zugang zu internen Daten bereitstellen (siehe [Fac; Twi17]).

Die Callback-Methode ist auch als Publish-Subscribe-Methode bekannt. Der Nachrichtempfänger registriert sich beim Nachrichtensender und abonniert dort ausgewählte Nachrichten. Will der Sender eine Nachricht versenden, sendet er diese nur an die ihm bekannte Liste von Abonnenten. Die Initiative geht vom Sender aus. Zum Abonnieren von Nachrichten muss dem Empfänger sowohl der Sender als auch die zu abonnierende Nachricht bekannt sein, weshalb ebenfalls eine enge Kopplung vorliegt. Beispiele für diese Form der Kommunikation sind Enterprise Service Busse. Ein Enterprise Service Bus empfängt als zentraler Vermittler-Dienst im Netzwerk alle Nachrichten und verteilt sie anhand eines vorliegenden

Regelwerks an die relevanten Clients. Die Verwendung von Event Handlern in Java, mit welchen Klassen die Nachrichten anderer Klassen abonnieren können, ist ein weiteres Beispiel. Bei der ereignisbasierten Methode sendet der Nachrichtensender beim Eintreten eines Ereignisses eine Nachricht an das Netzwerk. Der Sender hat keine Information, wer diese Nachricht empfängt bzw. ob sie von einem Teilnehmer verarbeitet wird. Die Nachrichtempfänger entscheiden selbstständig, ob sie diese Nachricht verwerten oder ignorieren. Die Initiative geht vom Sender aus, und es herrscht eine lose Kopplung zwischen den Teilnehmern, da diese sich nicht gegenseitig kennen müssen. Die Kommunikation zwischen Sender und dem zuvor erwähnten Enterprise Service Bus ist ein Beispiel hierfür. Der Sender sendet seine Nachricht an den Vermittler ohne Kenntnis darüber, an wen sie weitergeleitet wird. Die Java-Swing-Bibliothek zur Gestaltung von Benutzungsoberflächen nutzt ebenfalls einen eventgetriebenen Ansatz, mit dem sie Benutzereingaben weiterleitet.

Die Request-Reply-Methode hat sich in der Vergangenheit insbesondere durch das Aufkommen von SoA etabliert. Der Ansatz ist für die Realisierung von Geschäftsprozessen mittels orchesterter Softwaredienste geeignet. Die Callback- und ereignisbasierte Methode sind jüngere Ansätze, welche mit der Internet-of-Things-Entwicklung Bekanntheit erlangten. Die Methoden sind insbesondere für komplexe Prozesse und bei sich ändernden Netzwerkteilnehmern geeignet. Beide sind Weiterentwicklungen der SoA und wurden zwischenzeitlich in der Praxis als SoA 2.0 proklamiert (siehe z.B. [Bru+10; Com06; The+17]).

2.1.3.3 Bestehende Protokolle und Standards

Im Kontext der Industrie-4.0-Entwicklung sind eine Vielzahl an Kommunikationsprotokollen und Lösungen aufgekommen, die die vorgenannten Methoden der Interprozesskommunikation verwenden. Im Folgenden wird eine Auswahl für die Produktion relevanter Protokolle und Standards vorgestellt. Für diese Arbeit ist dabei ausschließlich die Kommunikation zwischen Anlagensteuerungen wie z.B. SPS und übergeordneten IT-Systemen wie z.B. MES und ERP-Systemen relevant. Betrachtet werden deshalb und u.a. aufgrund des vergangenen Erfolges und De-facto-Standards für die Netzwerkkommunikation im Internet der Dinge Protokolle, welche auf TCP/IP und Ethernet basieren. Ein Vergleich sowie weitere Protokolle finden sich beispielweise in [Her17b].

Die Open Platform Communications Unified Architecture (OPC UA), auch bekannt als IEC 62541, existiert seit 2008 und ist der Nachfolger von OPC Classic, zu welchem es abwärtskompatibel ist. Wie in Abbildung 15 dargestellt, beschreibt OPC UA ein Informationsmodell, das in Kapitel 2.1.4.3 vorgestellt wird, und ein Protokoll für den maschinenlesbaren Informationsaustausch zwischen unterschiedlichen Geräten.

Dritte können mit OPC UA nicht nur Informationen aus Anlagensteuerungen auslesen, sondern durch Methodenaufrufe (vgl. Callback-Methode) auch Aktivitäten in der Steuerung auslösen. Es lässt sich eine ereignisorientierte Kommunikation umsetzen, welche Netzwerkverkehr durch das zyklische Abfragen von Geräten (engl. Polling) vermeidet. Bei der Verwendung von OPC-UA kann für die Datenübertragung zwischen zwei verschiedenen Protokollen gewählt werden. Der Webservice-Ansatz basiert auf HTTP und dem Simple Object Access Protocol (SOAP) und bietet eine hohe Flexibilität. Das Binärprotokoll hingegen ist auf Durchsatz und Übertragungsgeschwindigkeit optimiert.

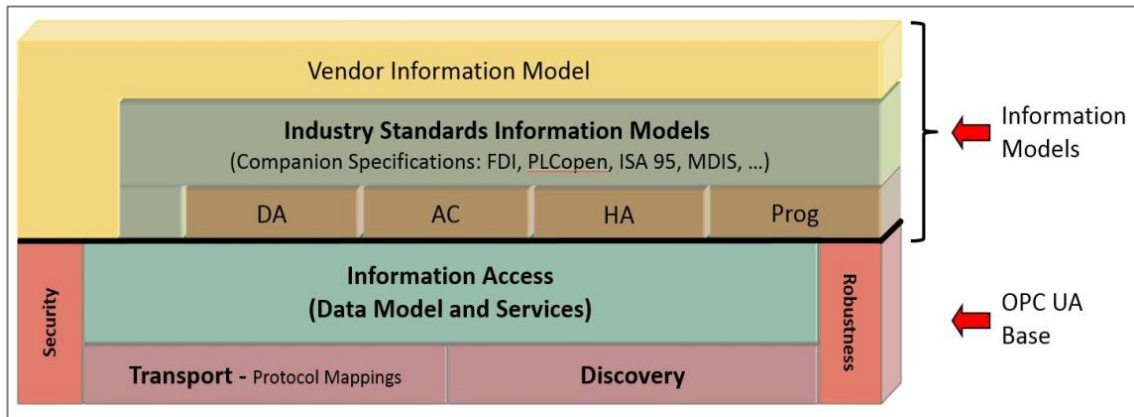


Abbildung 15 Die OPC Unified Architecture (Quelle: [OPC15])

Üblicherweise wird bei der Realisierung einer OPC UA-basierten Kommunikation zwischen einem Client als lesende Anwendung und einem Server als informations anbietende Anwendung unterschieden. Ein Gerät, welches sowohl Informationen anbietet als auch liest, implementiert somit zwei unabhängig voneinander laufende Anwendungen. Abbildung 14 zeigt die Kommunikationsmöglichkeiten zwischen Client, Server und Client und Server unter Verwendung der Callback- und der ereignisbasierten Methode.

Weitere Eigenschaften von OPC UA sind u.a. die Plattformunabhängigkeit, die Serviceorientierung, Skalierbarkeit, Sicherheit und Interoperabilität. [OPC15; Dam+09] Aufgrund dieser Vorteile, der Abwärtskompatibilität und dem Ursprung in der industriellen Produktion wird OPC UA als De-facto-Standard für die digitale Vernetzung der Produktion angesehen. [Spi16; Rot16]

Das Message Queue Telemetry Transport (MQTT) ist ein offenes, von der Organization for the Advancement of Structure Information Standards (OASIS) seit 2013 standardisiertes Protokoll, das eine schlanke und domänenunabhängige Maschine-zur-Maschine (M2M)-Kommunikation ermöglicht. MQTT wurde von IBM mitentwickelt und war in der Vergangenheit u.a. unter dem Namen „SCADA Protokoll“ zur Anbindung von Überwachungssystemen in der Verfahrenstechnik bekannt. [MQT]

MQTT verwendet für die Kommunikation die Publish-Subscribe-Methode. Clients veröffentlichen oder abonnieren Nachrichten von einem gemeinsamen Bus, dem Broker. Eine MQTT-Nachricht besteht aus einer Überschrift (engl. Topic), einem Hinweis zur notwendigen Übertragungsqualität (Qualität des Dienstes; engl. Quality of Service) und einem frei beschreibbaren Inhalt (engl. Payload). Das Abonnieren von Nachrichten beschränkt sich nicht auf feste Topics. Platzhalter und Hierarchien gruppieren Topics und ermöglichen den Empfang vorab unbekannter Nachrichten. [OAS14]

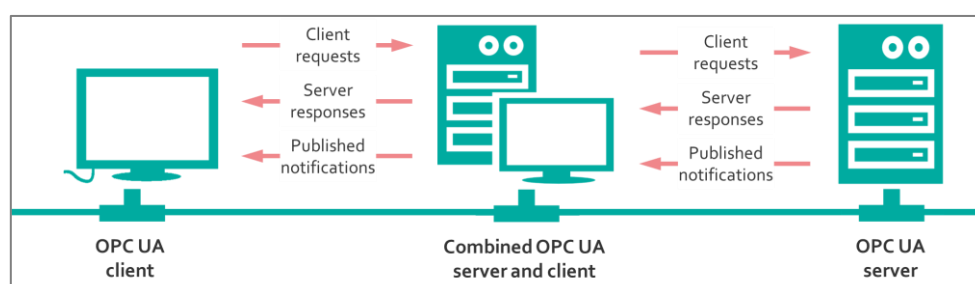


Abbildung 14 Möglichkeiten der Kommunikation in OPC UA (Quelle: [OPC12c])

TCP/IP ist das Standard-Protokoll von MQTT für den Nachrichtenaustausch. Allerdings sind unter Einhaltung bestimmter Bedingungen auch andere Protokolle wie z.B. Zigbee geeignet. Für die Datensicherheit bietet MQTT eine verschlüsselte Verbindung sowie Authentifizierung von Clients am MQTT-Broker. Die Verwendung von Websockets ermöglicht es, lokal installierte Browser ohne zusätzliche Software als MQTT-Clients zu nutzen. [OAS14; dc-17]

MQTT-basierte Anwendungen finden sich nicht nur im Bereich des Auslesens von Sensoren, sondern auch bekannte Internetplattformen wie Facebook verwenden es. Neben der Einfachheit und Offenheit ist ein weiterer Vorteil von MQTT die Vielzahl an Werkzeugen zur Realisierung von MQTT-Lösungen. Bekannte Beispiele sind der Enterprise Integration Bus von IBM oder die graphische Oberfläche Node-RED zur Realisierung von ereignisbasierten Abläufen zwischen Internet-der-Dinge-Geräten (siehe [IBM; JS 17] und Abbildung 16). Durch die Anwendungsvielfalt und den geringen Umsetzungsaufwand für Prototypen erfreut sich MQTT einer hohen Beliebtheit. [Spi16]

Das Data Distributed Services (DDS) ist ein von der Object Management Group (OMG) entwickeltes und standardisiertes Datenaustauschformat für die M2M-Kommunikation. Ziel von DDS ist es, eine effiziente, stabile und rechtzeitige Kommunikation in verteilten Systemen herzustellen. Ursprünglich wurde DDS 2004 für Domänen wie der Finanzwirtschaft, der Flugverkehrskontrolle oder für Smart Grids geschaffen. [Obj15; Cor+]

DDS verwendet wie MQTT die Publish-Subscribe-Methode. Als Middleware zwischen dem Betriebssystem und der Anwendung ermöglicht es die technologieunabhängige Kommunikation mit Drittsystemen und ist unabhängig von bestimmten Programmiersprachen. DDS-Geräte können andere Netzwerkteilnehmer erkennen, sodass keine zentrale Registrierung oder ein gemeinsamer Bus notwendig ist. Die Kommunikation erfolgt direkt zwischen den betroffenen Geräten. Durch ein komplexes Regelwerk für die Qualität der übertragenen Daten lassen sich Bedingungen wie die Lebensdauer, gültige Latenzen und Transportprioritäten definieren. Die Beschreibung der Daten kann in gängigen Beschreibungssprachen wie XML erfolgen. [Cor+; Obj15]

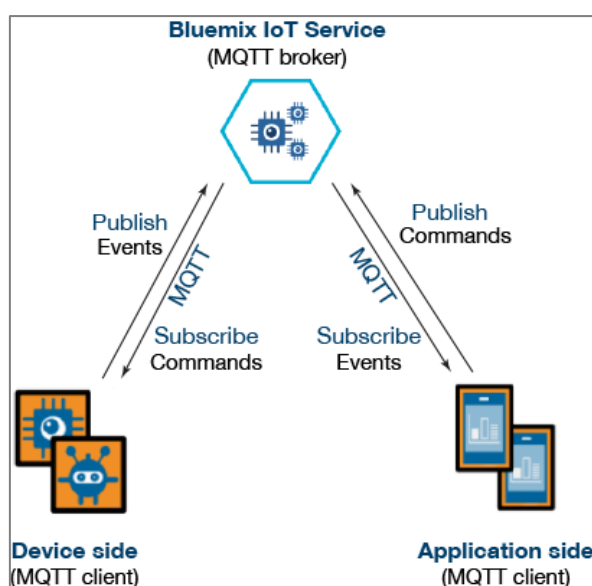


Abbildung 16 Beispiel für die Kommunikation mittels MQTT unter Verwendung von IBM Bluemix als MQTT-Broker (Quelle: [Bin15])

DDS wurde ursprünglich nicht für die Produktion entwickelt, verfügt über keine Sicherheitsmechanismen und ist nicht für die Realisierung von Webservices geeignet. Dennoch wird es teilweise aufgrund von Eigenschaften wie der Interoperabilität, der losen Kopplung von Geräten, der Erweiterbarkeit und der Skalierbarkeit im amerikanischen Raum als das Kommunikationsprotokoll der zukünftigen Produktion angesehen. [Cor+; Spi16]

Weitere Ansätze für die Implementierung informationstechnischer Schnittstellen sind Webservices, die den Nachrichtenaustausch zwischen Diensten auf Basis der Request-Reply-Methode und der Extensible Markup Language (XML) realisieren. Bekannte Standards zur Datenbeschreibung sind SOAP und Web Service Description Language (WSDL) [Aus+04; Mel10]. Aus dem Bereich der agentenbasierten Systeme gibt es ferner noch die Agent Communication Language (FIPA-ACL), welche die Kommunikation und die Schnittstellen zwischen Agenten spezifiziert [Pos07].

2.1.4 Informationsmodelle und -modellierung

2.1.4.1 Eigenschaften, Arten und Abstraktionsklassen von Modellen

Die vorgenannten Standards und Protokolle beschreiben, wie Informationen beschrieben werden. Welche Inhalte zu beschreiben sind, definieren Informationsmodelle. Informationsmodelle gehören zur Gattung der Modelle und besitzen somit weitestgehend die gleichen Eigenschaften. Ziel eines Modells ist es, die in der Realität gegebene Komplexität zu reduzieren, indem es nur die für einen Sachverhalt relevanten Dinge abbildet. Modelle unterliegen dabei dem bewussten und unbewussten Wissen und den Präferenzen des Modellierers. Außerdem sind Modelle ein Mittel zum Zweck und somit pragmatisch. Aufgrund dessen sind Modelle niemals vollständig. [Kra+07; Sta73]

Modelle können graphisch oder semantisch die Realität abbilden. Graphische Modelle dienen primär zur Visualisierung von Zusammenhängen und sind oftmals formalisiert. Beispiele hierfür sind CAD-Modelle aus der Konstruktion, Flussdiagramme aus der Informatik oder Organigramme aus der Organisationslehre. Semantische Modelle wie beispielsweise Ontologien sind weniger zur Repräsentation materieller Gegenstände geeignet, sondern eher zur Darstellung des Denkens und der Wahrnehmung. Darüber hinaus bilden dynamische Modelle wie z.B. Vorgehensmodelle oder Prozessmodelle Abläufe ab. Statische Modelle hingegen repräsentieren Strukturen und Zusammenhänge. Organigramme und UML Klassendiagramme sind Beispiele hierfür. Ferner lassen sich stark formalisierte Darstellungsweisen wie z.B. die Modellierungssprachen UML oder BPMN, und wenig formalisierte Modelle wie z.B. Skizzen unterscheiden. Eine weitere Dimension zur Unterscheidung von Modellen ist der Zweck. Beschreibungsmodelle dokumentieren einen vorliegenden Sachverhalt. Simulationsmodelle wie z.B. Wettersimulationen werden für die Vorhersage verwendet, und Erklärungsmodelle versuchen, Zustände durch Hypothesen zu erklären. [Kra+07; Wil13]

Neben der Art unterscheiden sich Modelle auch im Abstraktionsgrad. Sie lassen sich nach [Sch02] in vier Abstraktionsklassen einteilen, welche in ähnlicher Form bereits in [Str96] identifiziert wurden. Die erste Klasse ist, ähnlich wie ein Foto, ein Abbild eines realen und zu einem bestimmten Betrachtungszeitpunkt vorliegenden Sachverhaltes, d.h. sie bildet Entitäten ab und stellt sie in einen Zusammenhang. Sie ist eine der wichtigsten Abstraktionsklas-

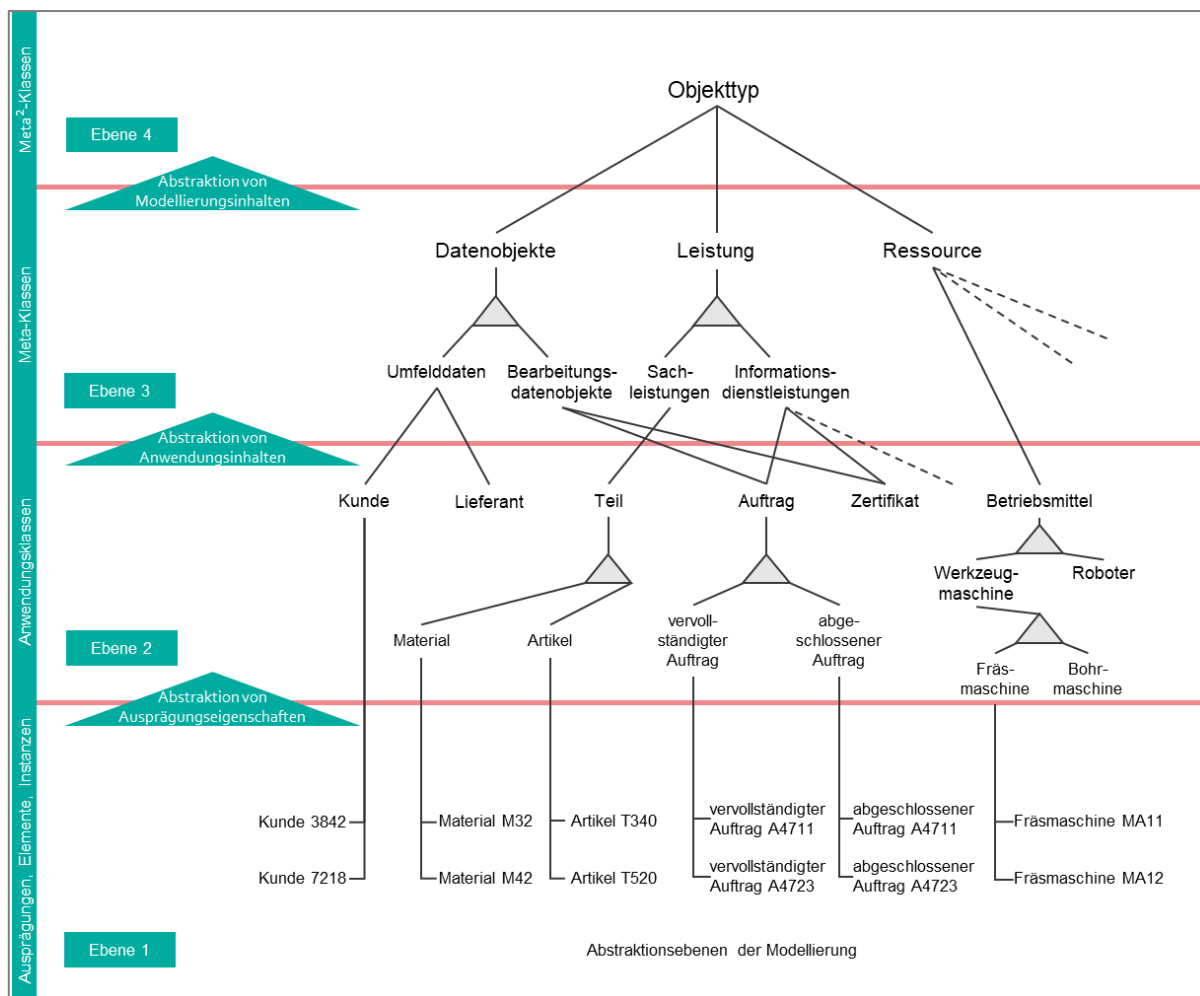


Abbildung 17 Abstraktionsebenen der Modellierung (Quelle: [Sch02])

sen bei der Ist-Analyse, da sie die für die Betrachtung relevanten Entitäten identifiziert. Die zweite Abstraktionsklasse abstrahiert die Entitäten und ordnet sie Entitätstypen mit Attributen zu. Diese Klasse beschreibt somit für einen Anwendungsfall bzw. eine Domäne allgegenwärtig vorkommende Objekte. Die dritte Abstraktionsklasse abstrahiert die Entitätstypen, indem sie diese den für die Anwendung relevanten Gruppen zuordnet. Diese Abstraktionsklasse wird auch als Meta-Modell bezeichnet. Die vierte Klasse, die Meta²-Klasse, definiert für die Gruppen übergeordnete, elementare Grundtypen. Der Übergang zwischen den Klassen ist fließend und teilweise nicht eindeutig möglich. Der Begriff „Objekt“ wird je nach Domäne und Darstellung als Synonym für Entität verwendet. Abbildung 17 zeigt den Zusammenhang der Abstraktionsklassen anhand eines Beispiels.

2.1.4.2 Definition und Relevanz von Informationsmodellen

Der Begriff „Informationsmodell“ ist in der Wirtschaftsinformatik weit verbreitet und wurde mit der Einführung von IT-Systemen für den administrativen Bereich in Unternehmen vermehrt verwendet. Auch wenn sich die Bedeutung des Begriffs über die Zeit veränderte und weiter gefasst wurde, besteht bis zu einem gewissen Detaillierungsgrad eine Gemeinsamkeit bei den Definitionen. [Wil13]

Eine der ältesten Definitionen für ein Informationsmodell stammt von [Pic+94]. Sie definieren Informationsmodelle als eine strukturierte Beschreibung informationsbezogener Aspekte eines betrieblichen Sachverhalts auf konzeptioneller Ebene. Im gleichen Jahr betonte [Sch93] in ihrer Definition, dass Informationsmodelle nie die gesamte Umgebung abbilden, sondern nur die für den Sachverhalt relevanten Dinge. [Teu99] beschränkt sich bei einem Informationsmodell nicht nur auf die informationsbezogenen Aspekte, sondern beschreibt Informationsmodelle auch als ein Abbild des Unternehmens zur Unterstützung der Softwareintegration. Seine Definition ist abstrakter und deutlich weiter gefasst als die im gleichen Jahr publizierte Definition von [Lee99]. Eine oft zitierte Definition ist die von [Wes+01]. Ähnlich wie [Lee99] ist ihrer Ansicht nach ein Informationsmodell eine technologieunabhängige Abstraktion von Entitätstypen, Eigenschaften, Attributen und Operationen in einer organisierten Umgebung. [Pra+03] befassen sich speziell mit der Abgrenzung eines Informationsmodells von einem Datenmodell. Im Gegensatz zu einem Datenmodell, das z.B. die konkrete Datenstruktur einer Datenbank beschreibt, ist ein Informationsmodell immer technologieunabhängig und sein Detaillierungsgrad abhängig von den Bedürfnissen des Modellierers. [Bec+04] sehen Informationsmodelle nicht nur als Werkzeug für die Software-, sondern auch für die Organisationsentwicklung. In Anlehnung an die Eigenschaften von Modellen allgemein betonen sie, dass ein Informationsmodell immer durch die subjektive Sicht des Modellierers geprägt ist. Basierend auf den vorgestellten Definitionen wird im Rahmen dieser Arbeit unter einem Informationsmodell folgendes verstanden:

Ein Informationsmodell ist eine technologieunabhängige Beschreibung von Funktionen, Beziehungen und Attributen von Entitätstypen, um Anwendungsfälle zu realisieren. Zweck des Informationsmodells ist die Unterstützung bei der Integration von IKT in vorgegebene Strukturen und Prozesse.

Abbildung 18 stellt den Zusammenhang zwischen einem Informationsmodell, der Realität und einem Datenmodell im Kontext eines Softwareentwicklungsprojektes exemplarisch dar. Das Informationsmodell bildet die Realität in einer eigenen Sprache ab, was die Bildung von Modellen für die Entwicklung und Integration erleichtert.

Die Relevanz von Informationsmodellen ist in der Praxis unumstritten [Loo+07]. Insbesondere bei der Kommunikation zwischen Fachabteilungen, die den Anwendungsfall kennen, und IT-Abteilungen, die das technische Fachwissen für die Implementierung besitzen, helfen Informationsmodelle, den Sachverhalt für die Entwickler zu verdeutlichen und ihn zielgerichteter durch IT-Systeme abzubilden bzw. zu unterstützen. [Sch93; Pic+94] Informationsmodelle finden insbesondere im Rahmen der Anforderungsdefinition und Konzeption Anwendung (siehe Abbildung 19 und Abbildung 20).

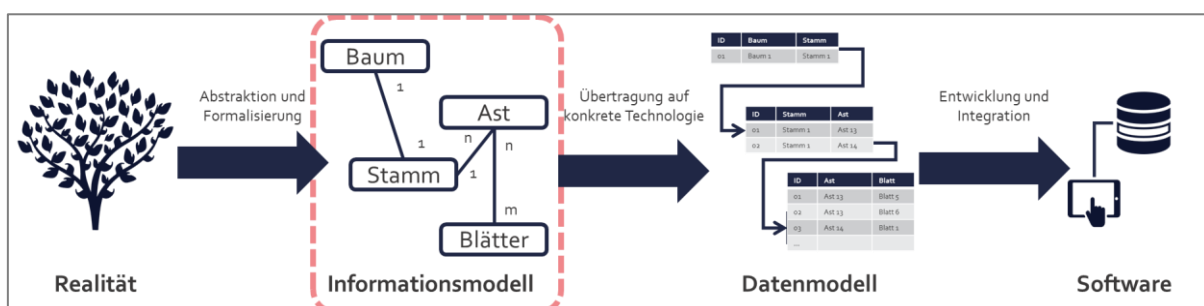


Abbildung 18 Informationsmodelle im Kontext des Softwareentwicklungsprozesses

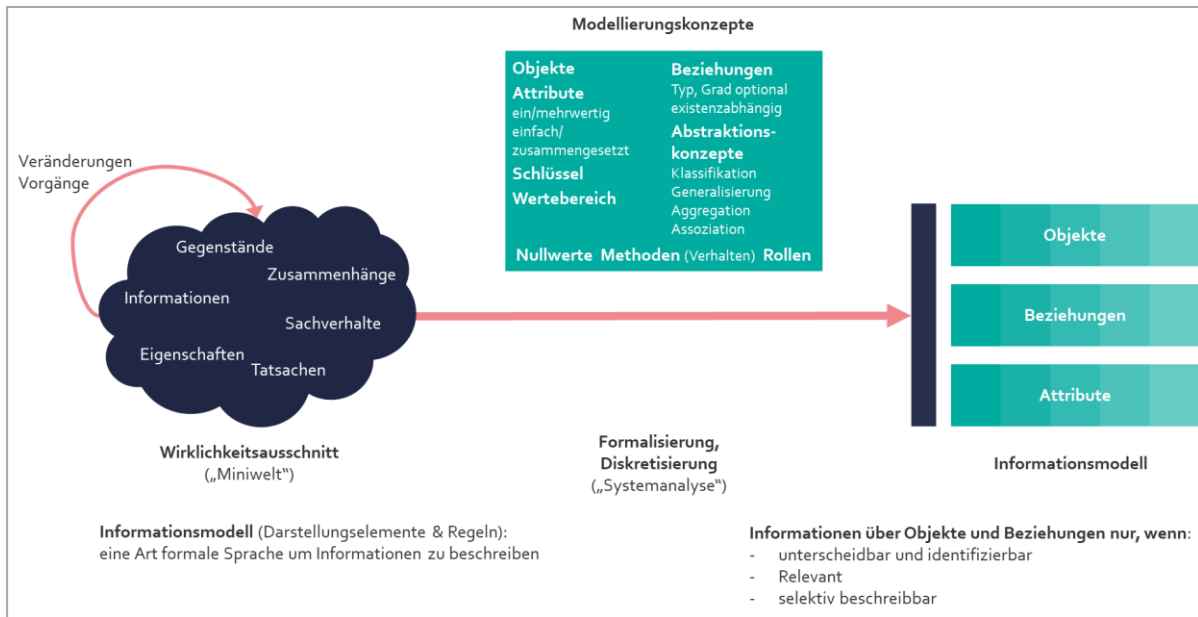


Abbildung 19 Konzepte zur Abstraktion der Wirklichkeit durch ein Informationsmodell (Quelle: [Deß13])

Bestehende Studien stellten dabei fest, dass die Informationsmodellierung in der Praxis bei der Software- und Datenbankentwicklung sowie der Optimierung und Dokumentation von Geschäftsprozessen besondere Aufmerksamkeit bekommt. Insbesondere bei der Konfiguration und Integration in bestehende Umgebungen sind Informationsmodelle nützlich. Grenzen sehen die Befragten u.a. bei der Handhabung komplexer Systeme. [Loo+07]

Durch die technologieunabhängige Betrachtung schränken Informationsmodelle die technische Realisierung in einem Entwicklungsprojekt nicht ein. Darüber hinaus ermöglichen sie in einem gewissen Rahmen die spätere Übertragung auf ähnliche Problemstellungen oder die Realisierung des Projektes mittels anderer Technologien. In der Produktion sind deshalb mit der Zeit diverse, teilweise standardisierte Informationsmodelle entstanden.

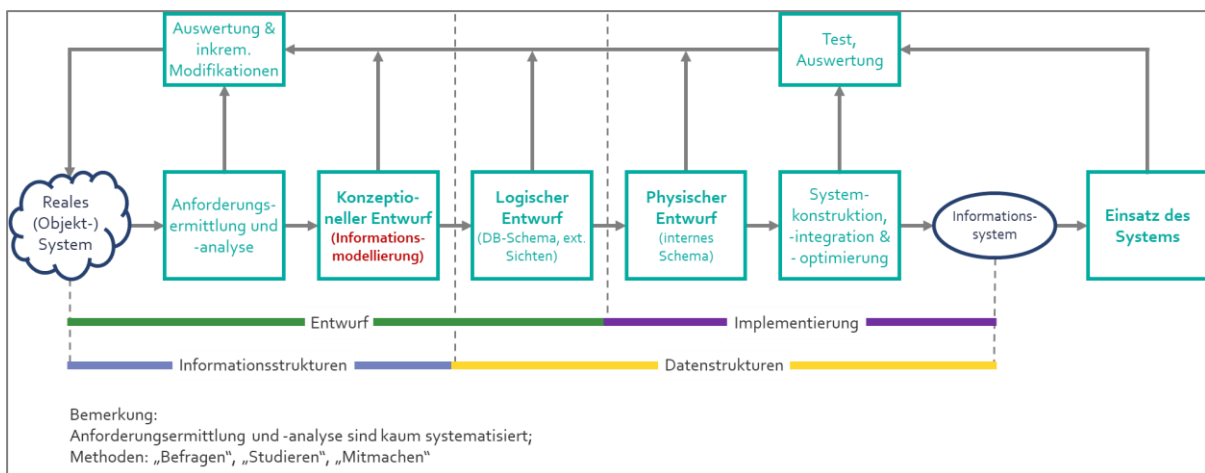


Abbildung 20 Rolle des Informationsmodells im Vorgehen zur Entwicklung von Informationssystemen(Quelle: [Deß13])

2.1.4.3 Bestehende Standards und Informationsmodelle

Das folgende Unterkapitel stellt eine Auswahl bestehender Informationsmodelle vor. Hierbei wird nicht nur auf die für diese Arbeit relevanten Modelle eingegangen, sondern auch auf die Vielfalt der modellierbaren Sachverhalte und Darstellungsformen. Wie zuvor in Kapitel 2.1.3.3 beschrieben, enthält OPC UA ein Kommunikationsprotokoll für den einheitlichen Datenaustausch zwischen unterschiedlichen Geräten. Daneben definiert es im Address Space Model zusätzlich ein Informationsmodell. Bei dem Address Space Model handelt es sich um ein Meta-Modell mit vordefinierten Typen zur Definition eigener, semantisch beschriebener Informationsmodelle. Zu diesen Grundtypen zählen Nodes (dt. Knoten) und References (dt. Beziehungen). Attribute dienen dazu, Nodes näher zu beschreiben. Nodes unterscheiden sich in Nodes, welche Instanzen repräsentieren, und Nodes, welche Typen beschreiben, die dann wiederum instanziiert werden können. Die wichtigsten Grundtypen von Nodes sind Objects, Variables und Methods. Objects dienen der Ordnung und Bildung von Gruppen. Variables repräsentieren Werte, die z.B. Dritte auslesen und ändern können. Methods sind Funktionen, die von außen angesprochen intern eine Aktion auslösen. References beschreiben Beziehungen zwischen Nodes und verweisen immer von einem Quellknoten auf einen Zielknoten. Durch unterschiedliche Typen von References lassen sich Abhängigkeiten zwischen Nodes wie z.B. „X ist Abstraktion von Y“ oder „X organisiert Y“ beschreiben. Auf Basis dieses Grundgerüsts leitet [OPC12], der fünfte Teil der OPC UA Spezifikation, die unterschiedlichen Subtypen von Nodes und References ab und definiert zwingend notwendige sowie optionale Attribute. Hierdurch können Entwickler domänenspezifische Einschränkungen vornehmen und das zu instanziiierende Informationsmodell definieren. Abbildung 21 zeigt, wie sich die einzelnen Typen des Meta-Modells zur Informationsmodellierung nutzen lassen. [Dam+09; OPC12a]

Neben dem in OPC UA beschriebenen Informationsmodell existieren ferner sogenannte Companion Specifications, welche beschreiben, wie existierende Informationsmodelle mithilfe des in OPC UA beschriebenen Metamodells zu konvertieren sind. Beispiele hierfür sind Companion Specifications für Standards und Spezifikationen wie FDI, PLCopen, ISA95 und MTConnect (siehe z.B. [OPC+10; OPC13; OPC+12b]).

MTConnect ist ein ebenfalls seit 2008 existierender, offener, herstellerübergreifender und erweiterbarer Standard zur XML-basierten Beschreibung von Werkzeugmaschinen von der Unternehmensebene bis zur Feldgeräteebene. Ziel von MTConnect ist es, die Datenakquisition von Geräten und Anwendungen zu verbessern und eine Plug'n'Play-Fähigkeit zu ermöglichen, d.h. MTConnect ersetzt oder erweitert nicht bestehende Funktionen eines Gerätes, sondern ermöglicht Dritten einen lesenden Zugriff darauf. [MTC14a; OPC+12b]

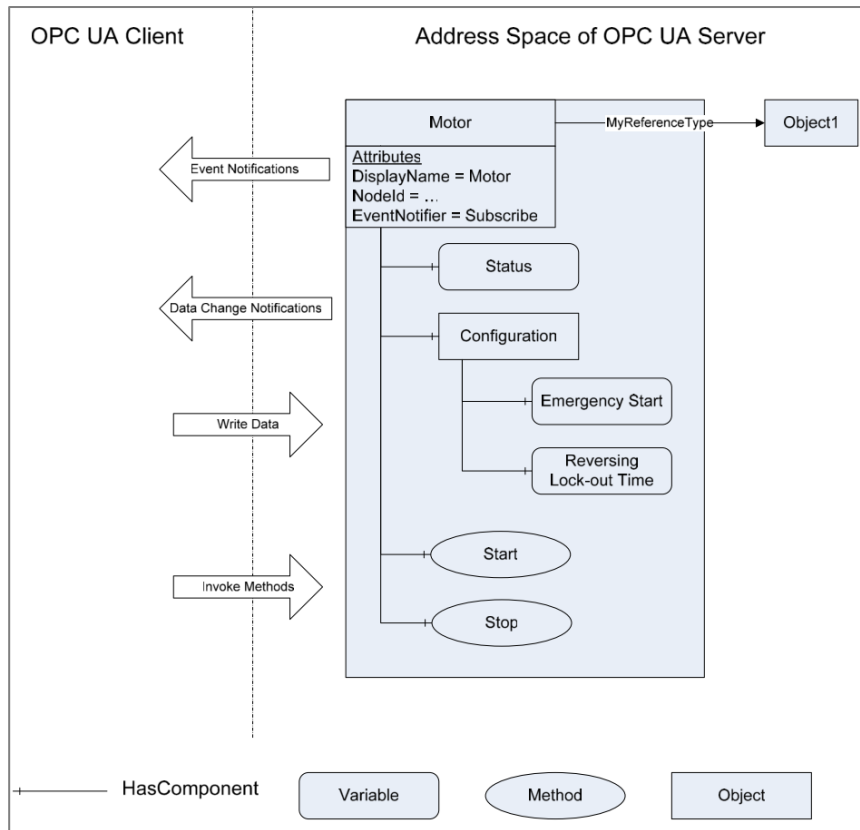


Abbildung 21 Beispiel für das (Meta-)Informationsmodell in OPC UA (Quelle: [Dam+09])

Das Informationsmodell von MTConnect definiert in einem Nachschlagewerk mögliche Fertigungsdaten. Die dem Standard zugrundeliegende Struktur beschreibt vier Komponenten, die miteinander interagieren. Wie in Abbildung 22 dargestellt, repräsentiert Device ein Betriebsmittel oder eine andere Datenquelle. Ein optionaler Adapter bildet die Schnittstelle am Device und konvertiert die Quelldaten in das MTConnect-Informationsmodell. Agents sind Softwarekomponenten, die diese Daten verarbeiten und mittels des gemeinsamen Kommunikationsnetzwerks mit anderen Datenquellen kombinieren können. Application ist schlussendlich der Nutzer der Daten. Er greift über einen Client darauf zu. [OPC+12b]

Das Informationsmodell aus MTConnect ist im Teil 2 „Device Information Model“ beschrieben (siehe [MTC14b]). Es beschreibt Regeln, Terminologie sowie zulässige Operationen und verfügbare Daten für Devices. Devices sind hierbei Container, die die zuvor erwähnten De-

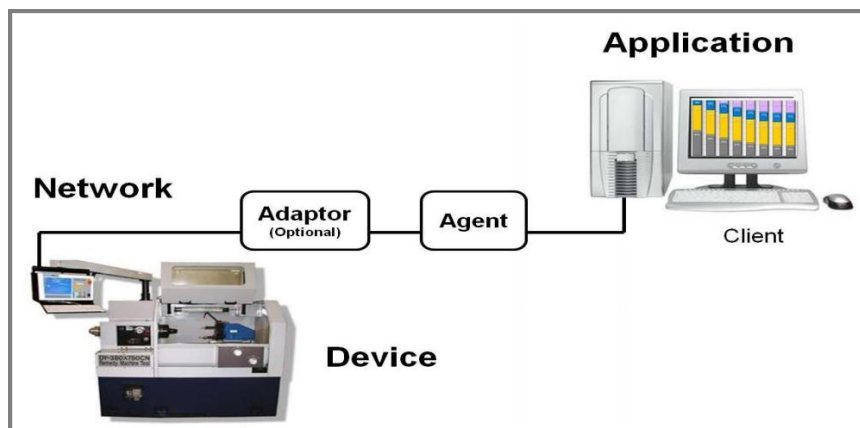


Abbildung 22 MTConnect zugrundeliegende Architektur (Quelle: [OPC+12b])

vice-Entitätstypen gruppieren. Jedes Device unterteilt sich wiederum in Components. Components sind ebenfalls Container für Component- und Subcomponent-Entitätstypen, welche physische oder logische Teile einer Anlage repräsentieren. Unterschiedliche Typen von Components ermöglichen die Klassifizierung und einheitliche Beschreibungen von Eigenschaften je Typ. Eine Component vom Typ Axes repräsentiert beispielsweise ein Feldgerät in einer Anlage, das rotierende oder lineare Bewegungen ausführt, und der Typ Sensor ist spezialisiert auf das Beschreiben von Messwerten eines Sensors. Abbildung 23 zeigt ein Beispiel für eine mögliche Hierarchie zur Repräsentation einer Anlage. Neben diesen Entitätstypen zur Beschreibung der Struktur, der Zusammenhänge und der Art von physischen oder logischen Komponenten existiert das Objekt Dataltem, das die für Dritte lesbaren Informationen vereint. Neben zwingend erforderlichen Attributen können optionale Attribute im Dataltem definiert werden. Ein Dataltem enthält zum einen administrative Angaben wie die Identifikationsnummer, den Namen, die Kategorie, den Typ der gemessenen Daten, die Wiederholungsrate und die Art des angebotenen Wertes (z.B. aktueller Wert, Durchschnittswert). Zum anderen enthält es technische Parameter wie z.B. die Position einer Linearachse oder die vorliegende Temperatur. Die Informationen lassen sich durch Angabe von Eigenschaften eines Wertes semantisch beschreiben. Eine Zuordnung von angebotenen Informationen zu Unternehmensprozessen ist allerdings nicht gegeben. Abbildung 24 zeigt die Struktur eines Dataltems und Beispiele für diese Attribute. [MTC14b]

Der US-Standard ISA 88 für Steuerungssysteme von Batch-Produktionsprozessen sowie das IEC-äquivalent 61512-01 beschreiben die mit am ältesten und bekanntesten Informationsmodelle im Produktionsumfeld. Sie definieren eine Terminologie, Daten und Strukturen zur Beschreibung von Komponenten und Prozessen für die Anbindung von Prozesssteuerungssystemen an Batch-Produktionsprozesse, wie sie z.B. die Chemie- und Pharmabranche nutzt. Ziel ist es, durch die Modularisierung von Funktionalitäten und Strukturen mit definier-

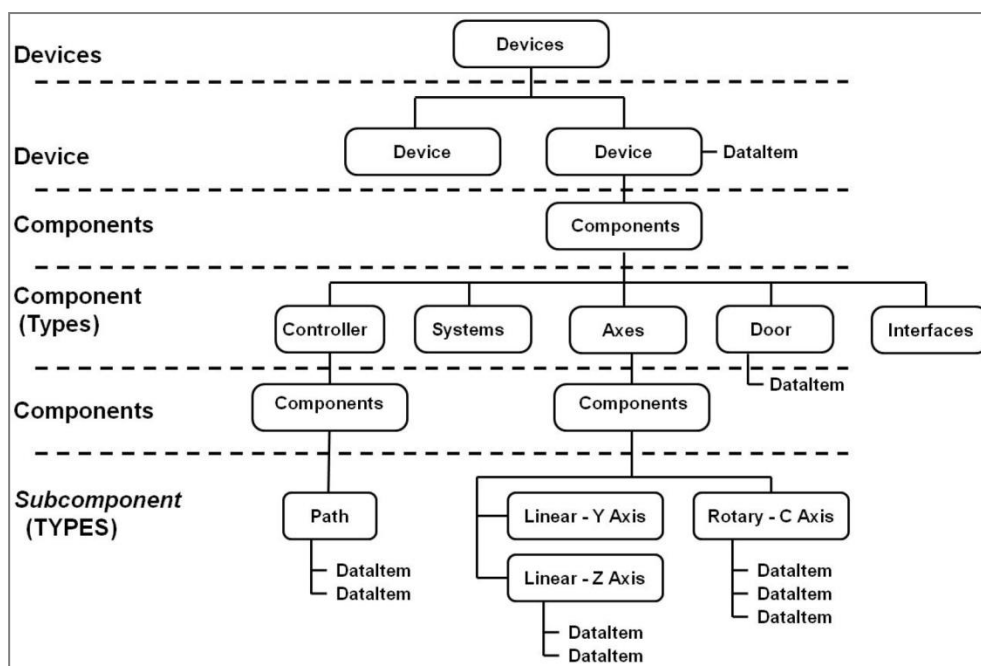


Abbildung 23 Beispiel für ein Informationsmodell zur Repräsentation einer Anlage
(Quelle: [MTC14b])

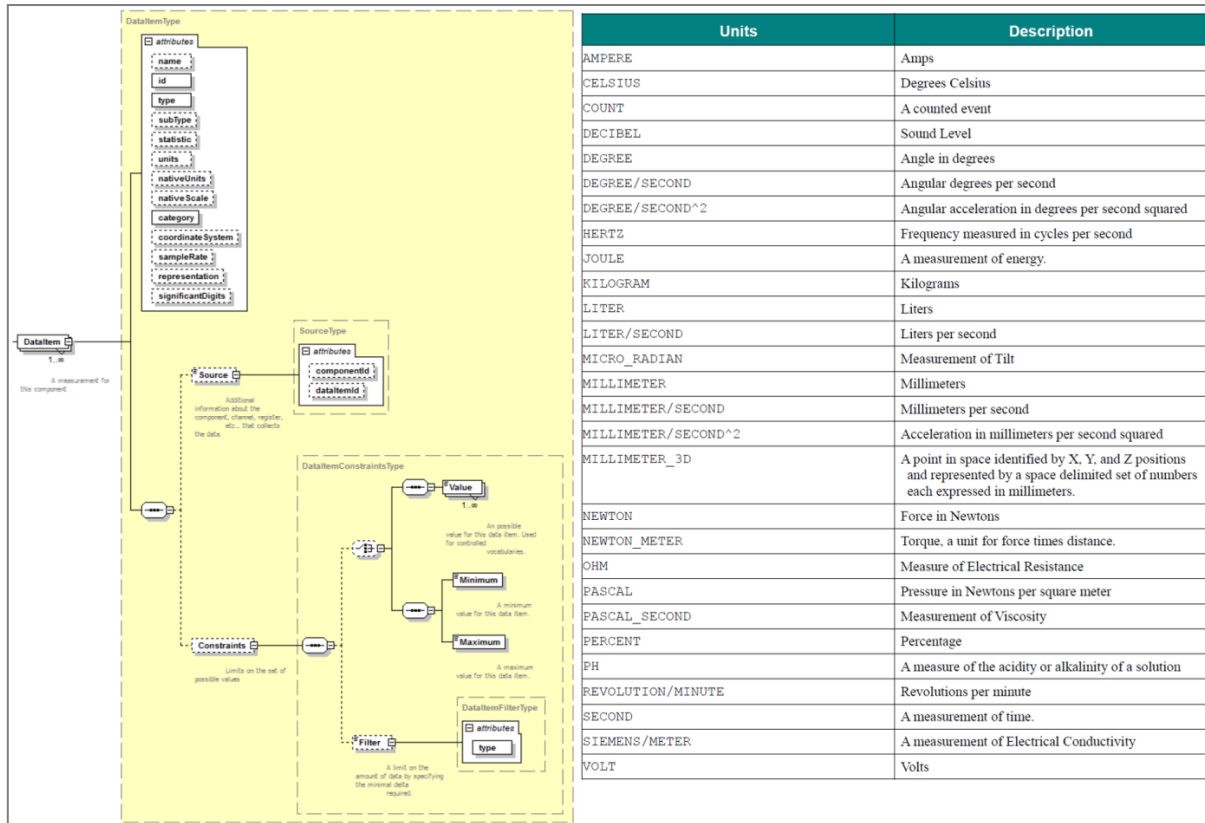


Abbildung 24 Aufbau eines DataItem (links) und Beispiel für Attribute eines DataItems (rechts) (Quelle: [MTC14b])

ten Schnittstellen die Interaktionspunkte zu standardisieren und zu minimieren. [the15]

Der ISA 88 untergliedert Prozesse, Rezepte und Anlagen bzw. IT-Systeme in Typen, aus denen sie bestehen können. Beispielsweise unterteilt das Physical Model der ISA 88 ein Unternehmen in unterschiedliche Standorte. Diese bestehen wiederum aus Bereichen, die in Process Cells alle Elemente zur Herstellung eines Batches zusammenfassen. Process Cells bestehen dabei immer aus Einheiten, welche Betriebsmittel und Steuerungen zur Ausführung einer Aktivität umfassen. Die Module des Physical Models werden in Beziehung zum Procedural Model, das die Ablaufsteuerung beschreibt, und zum Prozessmodell, das Fertigungsaufträge bzw. Rezepte repräsentiert, gesetzt. Abbildung 25 visualisiert den Zusammenhang. [Ame95]

Aufgrund des Erfolges der ISA 88 wurde das Konzept auf diskrete und kontinuierliche Fertigungsprozesse erweitert und in der ISA 95 bzw. IEC 62264 standardisiert. ISA 95 ist dabei nicht nur ein Pendant, sondern auch eine Ergänzung der ISA 88 und detailliert im Physical Model das Unternehmen. Der ISA 95 enthält ein Modell und eine einheitliche Terminologie zur Beschreibung von auszutauschenden Informationen zwischen Verkaufs-, Finanz-, Logistik-, Produktions-, Wartungs- und Qualitätsmanagementsystemen.

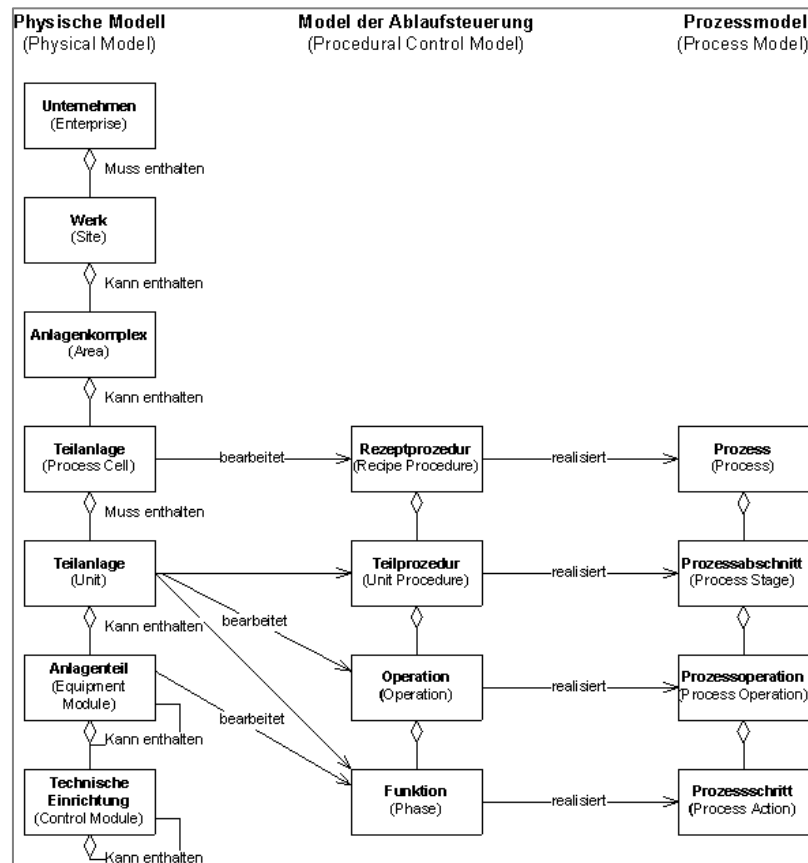


Abbildung 25 Zusammenhang der Teilmodelle und deren Untergliederung in ISA 88
(Quelle: [Wik16b])

Hierzu werden fünf Ebenen, welche auch in Abbildung 26 dargestellt sind, definiert. Level 0 beschreibt den Produktionsprozess. Level 1 beschreibt die für die Steuerung und Kontrolle notwendigen Sensoren und Aktoren. Level 2 enthält Komponenten zur Überwachung und automatisierten Steuerung des Produktionsprozesses. Level 3 beschreibt die Ebene der Produktionsplanung und -steuerung. Level 4 ist die höchste Ebene und beschreibt die übergeordnete Geschäftsmodellebene. Die Unterteilung von Komponenten in den Ebenen erfolgt nach funktionalen Gesichtspunkten in die Teile Business Planning & Logistics, Manufacturing Operations Management, Batch Control, Continuous Control und Discrete Control. Aktivitätsdiagramme, welche Zusammenhänge zwischen Prozessen der einzelnen Ebenen definieren und exemplarisch in Abbildung 27 dargestellt sind, ergänzen das Modell. [Kol15a; DIN08]

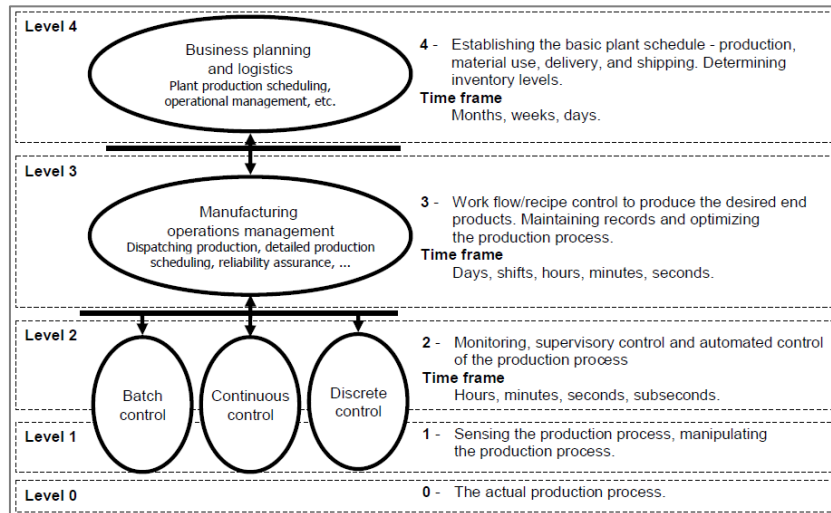


Abbildung 26 Das 5-Ebenen-Modell der ISA 95 (Quelle: [DIN08])

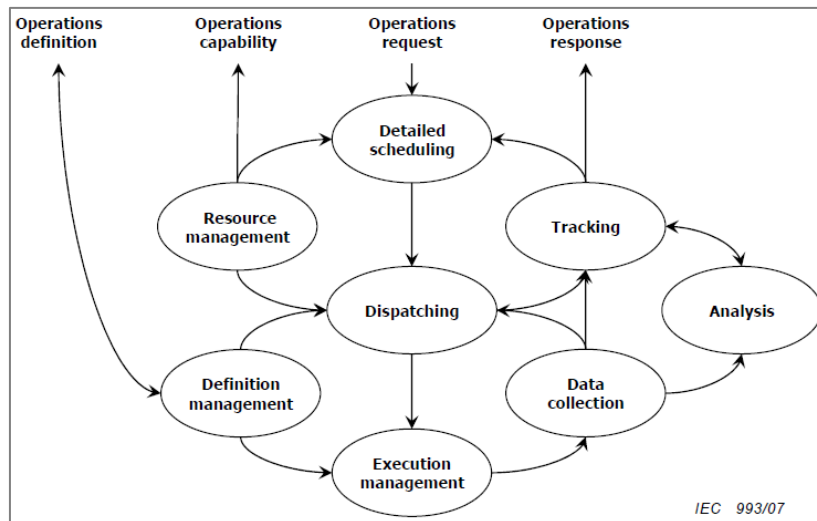


Abbildung 27 Beispiel für ein Aktivitätsmodell der ISA 95 (Quelle: [DIN08])

Bei eCI@ss handelt es sich um einen seit 2000 etablierten Standard zur Klassifizierung und eindeutigen Beschreibung von Produkten und Dienstleistungen. Mit 41.000 Produktklassen und fast 17.000 Merkmalen ist eCI@ss in Bereichen wie der Beschaffung, dem Controlling und dem Vertrieb verbreitet, um Produkte eindeutig zu beschreiben. Darüber hinaus findet es seit jüngster Zeit Anwendung im unternehmensübergreifenden Prozessdatenmanagement sowie dem Engineering. [eCI15; eCI14]

eCI@ss definiert einen vierstufigen Hierarchiebaum mit Klassen von Entitätstypen, welche sich durch ähnliche Eigenschaften auszeichnen und mit jeder Ebene konkreter werden. Merkmale beschreiben die Entitätstypen. Zu diesen Merkmalen gehört z.B. die achtstellige Identifikationsnummer, mit welcher sich auf die zugehörige Klasse schließen lässt. Weitere Merkmale sind abhängig vom Entitätstyp. Wie in Abbildung 28 dargestellt, beschreibt die Nummer 27-02-31-01 beispielsweise Frequenzumrichter mit weniger als 1 kV, die über Merkmale wie die Maße, zulässige Einsatzbereiche oder die maximale Spannungsangabe eindeutig beschrieben werden können. Beim Instanzieren wird diesen Merkmalen dann der Wert des konkreten Objektes zugewiesen. Da es sich um ein Klassifizierungssystem handelt gibt es keine Hinweise, in welchen Prozessen und Anwendungen es wie zu nutzen ist. [eCI15; eCI14]

Version: 9.1 (BASIC) Sprache: Zur Suche in der eCI@ss ADVANCED bitte hier klicken

Suche in: nach:

Klassifikation: 27-02-31-01 Frequenzumrichter =< 1 kV [AKE177013]

Bevorzugte Benennung: Frequenzumrichter =< 1 kV

Definition: -

Schlagworte: Drehzahlsteller Frequenzumformer Frequenzregler Motorabgang Frequenzumsetzer Motorregelung Motorregler Motorsteller

Merkmaleiste:

[0173-1#02-AA0676#002](#) - Herstellerartikelnummer

[0173-1#02-AAU734#001](#) - Herstellerproduktbeschreibung

[0173-1#02-AAU733#001](#) - Herstellerproduktbestellzusatz

[0173-1#02-AAU732#001](#) - Herstellerproduktstamm

[0173-1#02-AAU731#001](#) - Herstellerproduktfamilie

[0173-1#02-AAQ326#001](#) - zusätzlicher Online-Verweis

[0173-1#02-AAP805#002](#) - Artikelbezeichnung

[0173-1#02-AAM551#001](#) - Lieferantenproduktbezeichnung

[0173-1#02-AAU730#001](#) - Lieferantenproduktbeschreibung

[0173-1#02-AAU729#001](#) - Lieferantenproduktstamm

[0173-1#02-AAU728#001](#) - Lieferantenproduktfamilie

[0173-1#02-AA0847#003](#) - Produkttypbezeichnung

[0173-1#02-AA0735#002](#) - Lieferantename

[0173-1#02-AA0736#003](#) - Lieferanteanzahl

[0173-1#02-AA0742#002](#) - Marke

[0173-1#02-AAD931#004](#) - Zolltarifnummer (TARIC)

[0173-1#02-AA0663#002](#) - GTIN

[0173-1#02-AA0677#001](#) - Hersteller-Name

[0173-1#02-AAS453#001](#) - min. Netzspannung

[0173-1#02-AA0494#002](#) - Anzahl der HW-Schnittstellen (Industrial Ethernet)

[0173-1#02-AA0495#002](#) - Anzahl der HW-Schnittstellen (parallel)

[0173-1#02-AA0496#002](#) - Anzahl der HW-Schnittstellen (PROFINET)

[0173-1#02-AA0500#002](#) - Anzahl der HW-Schnittstellen (seriell, RS232)

[0173-1#02-BAB577#006](#) - Tiefe

Beispielhafte Darstellung




Abbildung 28 Beispiel für die Klassifikation mittels eCI@ss (links Hierarchiebaum; rechts Merkmale; Quelle: [eCI12])

Das Informationsmodell der VDI-Richtlinie 2860 „Montage- und Handhabungstechnik; Handhabungsfunktionen, Handhabungseinrichtungen; Begriffe, Definitionen, Symbole“ beschreibt Funktionen im Kontext von Logistikprozessen. Sie vertieft die in der VDI-Richtlinie 2411 beschriebenen Prozesse der Förder- und Lagertechnik, indem sie Elementarfunktionen des Handhabens definiert und in Beziehung zueinander stellt.

Im Gegensatz zu eCI@ss beschreibt die 1990 veröffentlichte Richtlinie keine Strukturen, sondern Tätigkeiten. Sogenannte Elementarfunktionen definieren grundlegende Aktivitäten der Montage- und Handhabungstechnik, welche durch die Kombination miteinander Abläufe in der Produktion abbilden können. Wie in Abbildung 29 dargestellt, sind die Elementarfunktionen in die fünf Bereiche Speichern, Mengen verändern, Bewegen, Sichern und Kontrollieren aufgeteilt. [Ver90] Nicht Bestandteil der Richtlinie ist eine Beschreibung, wie sich diese Elementarfunktionen umsetzen lassen bzw. welche Informationen und Betriebsmittel notwendig sind.

Die vorgestellten Standards bzw. Spezifikationen sind eine Auswahl an in der Produktion verbreiteten Standards zur einheitlichen Beschreibung von Informationen. Weitere Beispiele sind Feldgeräteprofile, welche für ganze Arten von Feldgeräten anwendungsbezogene Funktionen und Eigenschaften beschreiben. Auch für Antriebe und Steuerungen existieren ähnli-

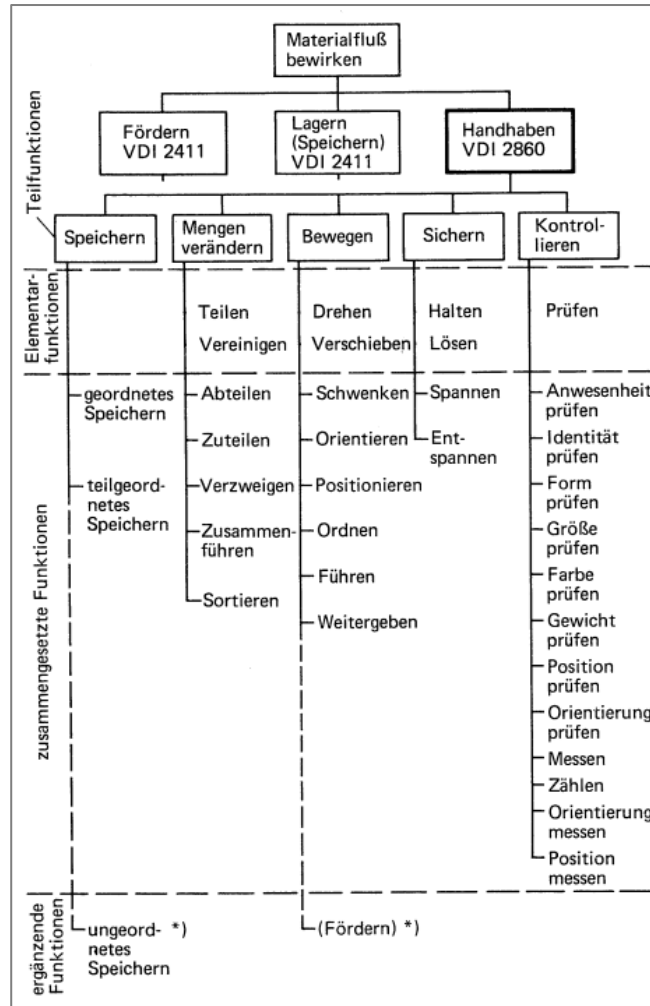


Abbildung 29 Untergliederung von Funktionen in der VDI-Richtlinie 2860 (Quelle: [Ver90])

che Standards wie DIN EN 61800-7-1/-20x/-30x, DriveCom Profile oder PLCopen Motion Control Function Blocks. Zur Beschreibung von Roboterzellen existiert zusätzlich die VDMA 66430-1. [Hod13] Im Bereich der Verpackungs- und Nahrungsmittelindustrie etablierten sich Informationsmodelle wie der Weihenstephan-Standard oder OMAC PackML. Sie standardisieren die Zustände von Anlagen und die Maschinendatenerfassung für übergeordnete IT-Systeme (siehe z.B. [Nok; Tec17a]). Ferner stellten [Beu+15] mit der Machinery Information Base Data Structure einen Ansatz zur Standardisierung von Anlagenstatus vor.

2.1.5 Bewertung des Status quo

Industrie 4.0 und seine internationalen Synonyme versuchen, die sich aus kürzeren Produktlebenszyklen und individuelleren Produkten ergebenden zukünftigen Anforderungen an die Produktion zu bewältigen, indem sie den technischen Fortschritt und Preisverfall in der IKT nutzen und diese Technologien in die Produktion integrieren. Die Digitalisierung und die Modularisierung sowie die dadurch erforderliche Dezentralisierung der Steuerung und die durchgängige Vernetzung zur gemeinsamen Kommunikation sollen der Industrie-4.0-Vision folgend helfen, die neuen Marktforderungen zu bewältigen.

Mit CPSen steht heutzutage eine kostengünstige Technologie zur Verfügung, welche die Steuerungszintelligenz dezentralisiert und flexibel programmierbare Rechenleistung in die

Produktion bringt. Die vielfältigen Schnittstellen von CPSen ermöglichen eine nachträgliche Integration bestehender Komponenten in ein gemeinsames Kommunikationsnetzwerk.

Für eine einheitliche Kommunikation im gemeinsamen Netzwerk etablierten sich Protokolle wie OPC UA, MQTT oder DDS. Sie setzen moderne, für die flexible Verbindung der Entitäten notwendige Paradigmen wie die ereignisbasierte Kommunikation oder die semantische Beschreibung von Informationen um und basieren auf aus dem Internet bekannten, etablierten Standards. Für die maschinenlesbare Beschreibung von Informationen existiert darüber hinaus eine Vielzahl an standardisierten Informationsmodellen. Sie beschreiben für unterschiedliche Domänen und Entitätstypen Informationen, die diese Dritten anbieten sollen. Durch die in RAMI 4.0 beschriebene Verwaltungsschale besteht ferner ein Architektur-Ansatz, welcher die von unterschiedlichen Interessenten benötigten Informationen zentral an einer Stelle bündelt.

Nichtsdestoweniger lässt sich feststellen, dass die aktuellen technischen Entwicklungen hinsichtlich der Integrationsfähigkeit in bestehende Produktionsumgebungen und der fokussierten Anwendungsfälle, welche dem Anwender Nutzen bringen, noch Defizite aufweisen. Die vorgestellten dezentralen Architekturen sind abstrakt, ausschließlich technologieorientiert oder betrachten eine Problemstellung nicht ganzheitlich. Sie geben keine Auskunft darüber, wie der Anwender sie implementieren muss und wie sich bestehende Produktionsprozesse mit diesen Architekturen vereinen lassen (siehe auch [Clo+09]).

Die vorgestellten Kommunikationsprotokolle beschreiben nicht, wofür sie sich in bestehenden Produktionsumgebungen nutzen oder wie sich Altanlagen migrieren lassen. Auch geben sie keinen Hinweis darauf, welche Entitätstypen der Produktion überhaupt zu integrieren sind.

Bestehende Informationsmodelle sind anwendungsorientierter und beschreiben für Domänen die z.B. von Anlagen ausgetauschten Informationen und wie sich diese maschinenlesbar beschreiben lassen. Sie sind für die Einführung eines dezentralen Ansatzes und eines gemeinsamen Kommunikationsprotokolls ein wichtiger Bestandteil [Lei09]. Dennoch sind sie limitiert auf ausgewählte Branchen oder Anwendungsfälle und geben keine Auskunft darüber, wie sie in Produktionsprozessen und -strukturen zu verwenden sind.

Anwender der im Kontext von Industrie 4.0 entstehenden Technologien stehen somit vor der Herausforderung, dass sie zum einen Anwendungsfälle und Prozesse identifizieren müssen, die sie abdecken wollen und in denen die Digitalisierung ihnen einen Mehrwert liefert. Zum anderen müssen sie ihre bestehenden Produktionsstrukturen und -abläufe auf eine von der Technologie vorgegebenen, idealtypischen Zustand migrieren.

Für die Anwendung der vorgestellten Technologien in Produktionsumgebungen, die z.B. nach den verbreiteten Prinzipien und Methoden der Lean Production organisiert sind, bestehen bisher keine Ansätze für den automatisierten, maschinenlesbaren Austausch. Die Forschung identifizierte diese Notwendigkeit bereits. [Kol+15b; Sch+15; Bru+14] [Sch13] zeigte dabei bereits, dass eine solche standardisierte und ganzheitliche Beschreibung von Informationsflüssen zwischen IT-Systemen im administrativen Bereich einen Mehrwert schafft.

2.2 Lean Production

Neben den Grundlagen und dem Status quo zur Digitalisierung in der Produktion wird im Folgenden das Themenfeld Lean Production vorgestellt. Im Fokus stehen hierbei neben den bestehenden Prinzipien und Methoden auch bestehende digitale Lösungen für Lean-Methoden. Das Kapitel 2.2 schließt ebenfalls mit einer Bewertung des aktuellen Stands der Technik sowie einer Betrachtung der Defizite ab.

2.2.1 Begriffsabgrenzung und historische Entwicklung

Der Begriff „Produktionssystem“ wird im Kontext von Industriebetrieben oft fälschlicherweise auf technische Produktionsmittel begrenzt (siehe z.B. [Dom+15; Fes11]). Für diese Arbeit relevant ist die wissenschaftliche Definition aus dem Kontext von Industriebetrieben, welche sich durch Zerlegung des Begriffes erklären lässt.

Für den Begriff „Produktion“ existieren engere und weitere Definitionen. Während engere Definitionen die Produktion als Oberbegriff der Teilefertigung und Montage verstehen [War93], dehnen weitere Definitionen den Begriff auf indirekte Bereiche wie Disposition und Logistik, Planung und Steuerung, Wartung und Instandhaltung sowie die operative Qualitätssicherung aus [Spa03; Ver12]. Die weiter gefassten Definitionen beziehen alle Betroffenen und Tätigkeiten des Leistungserstellungsprozesses mit ein (siehe [Blo+08]). Ein System ist in der Systemtheorie gekennzeichnet durch eine Systemgrenze, Entitäten innerhalb des Systems sowie Beziehungen zwischen diesen Entitäten [Hab+12; Wie14a; Kra+07]. Ein Produktionssystem im Industriebetrieb ist demzufolge eine abgegrenzte Menge von nicht näher spezifizierten Entitäten und deren Beziehungen untereinander, welche dem Leistungserstellungsprozess dienen.

Bestehende Definitionen knüpfen hieran an und beschreiben Produktionssysteme als soziotechnische Systeme, die die für die Herstellung eines Endprodukts notwendigen Strukturen und Prozesse umfasst [Eve96; zit. n. Nyh08]. Konkretere Definitionen fügen dem hinzu, dass alle Elemente auf ein gemeinsames Ziel abgestimmt sind. Darüber hinaus betonen sie den Einsatz von standardisierten Methoden in der Produktion. [Ank+08; Neu07; Cis+02; Ver12]

Als erstes bekannt gewordenen Produktionssystem gilt die wissenschaftliche Betriebsführung (engl. Scientific Management) von Taylor (siehe [Tay12]). Zu Beginn des 20. Jh. stellte Taylor auf Basis wissenschaftlicher Analysen Gesetze und Prinzipien zur standardisierten und optimierten Produktion auf, womit er der bis dato vorherrschenden Vielfalt an Werkzeugen und Abläufen für gleiche Tätigkeiten und der hohen Abhängigkeit von den Fähigkeiten eines Werkers entgegenwirkte. [Tay12] Wichtige Elemente waren u.a. das Vorgeben eines Lösungsweges zur Bewältigung einer Aufgabe, das Ersetzen von „Daumenregeln“ durch fundierte Methoden sowie das Definieren von Einzelschritten mit Bearbeitungszeiten für Aufgaben. Auf dem Scientific Management basieren u.a. die Methoden des Verbands für Arbeitsgestaltung, Betriebsorganisation und Unternehmensentwicklung (REFA). [Dom+15]

In diesem Zusammenhang ist ebenfalls zu Beginn des 20. Jh. die Massenproduktion von Ford entstanden. Sie ist weniger ein eigenes Produktionssystem, sondern vielmehr eine Weiterentwicklung der wissenschaftlichen Betriebsführung. Ford gelang es, Taylors Produktionssystem mit neuen Produktionstechnologien zusammenzuführen. [Cla05; Dom+15] Taylors

standardisierte Arbeitsinhalte ermöglichten es Ford, das Fließband in der Fertigung einzuführen. Außerdem führte er Schablonen zur Qualitätsprüfung ein, entwickelte die Aufgabenfragmentierung weiter und reduzierte die Variantenvielfalt in der Produktion auf ein Modell. Ford konnte so auch ungelernete Arbeitskräfte einsetzen. Die Weiterentwicklung der Methoden oblag dabei einer neuen Berufsgruppe, den Industrial Engineers. [Cla05; Dom+15; Ble14] Der Erfolg der Massenproduktion führte zu einer schnellen Verbreitung des Produktionssystems von Taylor und Ford insbesondere in nordamerikanischen und europäischen Unternehmen. Kritiker bemängelten schon damals, dass die Fragmentierung zu einer Monotonie der Arbeit führe, welche Mitarbeiter demotiviere und gesundheitsschädlich sei. Die Fragmentierung führe ferner zu einer hohen Bürokratie und somit langsamen Anpassungsgeschwindigkeit bei Änderungen. Mit dem Wandel von Verkäufer- zu Käufermärkten reiche es nicht mehr aus, nur eine Produktvariante zu fertigen. [Dan97; Cla05]

Der Marktwandel und der verhältnismäßig kleine japanische Markt konfrontierten den Automobilhersteller Toyota in den 1950ern mit neuen Herausforderungen. Die Fertigung kleiner Losgrößen und eine hohe Variantenvielfalt stellten neue Anforderungen an Toyotas Produktion. Hinzu kam eine bis zu neunmal geringere Produktivität der japanischen Mitarbeiter im Vergleich zu den amerikanischen. [Ohn93] Basierend auf Vorarbeiten der japanischen Familie Toyoda und Ford erarbeitete der Produktionsleiter Ohno das Toyota Produktionssystem (TPS). In seiner Analyse der amerikanischen Automobilhersteller übernahm Ohno u.a. Ideen des umfassenden Qualitätsmanagements (engl. „Total Quality Management“) sowie der nachfrageorientierten Bereitstellung von Waren. Die sich in Japan nicht einstellenden Skaleneffekte der Massenproduktion, die hohen Fixkosten der Automatisierung und ein hoher Anteil an Verschwendung bei den Arbeitsabläufen waren Herausforderungen, die Ohno im TPS adressierte. [Ohn93]

Ergebnis ist eine auf ein gemeinsames Ziel abgestimmte Sammlung von Prinzipien und Methoden zur Organisation der Produktion. Hauptziel des TPS ist die kontinuierliche Verbesserung der Effizienz durch vollständige Vermeidung von Verschwendung und die konsequente Ausrichtung des Unternehmens auf die Bedürfnisse des Kunden. [Ohn93; Gor+11] Die Ergebnisse lassen sich im sogenannten TPS-Haus zusammenfassen, welches in Abbildung 30 dargestellt ist. Basis des TPS bildet neben der Verschwendungsvermeidung die Standardisierung von Abläufen und Werkzeugen. Ferner bildet die Beteiligung des Werkers und die kontinuierliche Suche nach Verbesserungen eine Grundlage. Unter dem Begriff „Produktionsnivellierung“ strebt Ohno eine gleichbleibende, möglichst schwankungsfreie Auslastung der Fertigung an. Aufbauend auf diesem „Fundament“ unterteilen sich die Prinzipien und Methoden in zwei Bereiche: Just-in-Time fasst alle Maßnahmen zur Umsetzung einer nachfrageorientierten Einzelfertigung zusammen und Autonomation, auch unter dem japanischen Begriff Jidoka bekannt, beschreibt alle Maßnahmen für eine automatisierte Fertigung, bei welcher der Werker nur im Falle von Ausnahmen und Fehlern eingreifen muss. Alle Prinzipien und Methoden sind abgestimmt auf die Forderung des Kunden nach einer höchstmöglichen Qualität bei geringstmöglichen Kosten und kürzester Durchlaufzeit. [Dic07c; Tak12; Ohn93]

Durch die verschwendungsarme und somit als schlank empfundene Produktion prägte bereits 1988 Krafcik den Begriff „Lean Production“ als Synonym für das TPS [Kra88], welchen auch Womack, Jones und Roos verwendeten. Womack et al. verglichen in den 1990ern die Produktion Toyotas mit der anderer Automobilhersteller wie z.B. GM, um die Gründe für den plötzlichen Erfolg Toyotas festzustellen (siehe [Wom+07]). In ihrer vom Massachusetts Institute of Technology (MIT) geförderten Studie stellten sie u.a. eine um die Hälfte kürzere Durchlaufzeit und eine ca. 25% höhere Produktivität der Lean Production im Vergleich zur Massenproduktion fest [Wom+07; Bru11]. Neuere Studien in unterschiedlichen Branchen kommen zu ähnlichen Erkenntnissen (siehe [Döp15; Dom+15; Kor+04; Mäh+95]). Ausgehend von der Automobilindustrie übernahmen andere Hersteller die Lean Production und passten das Produktionssystem an ihre eigenen Bedürfnisse an. Namhafte Beispiele sind das Opel-Produktionssystem (OPS), Mercedes Production System (MPS), Bosch Production System (BPS) oder das von Takeda entwickelte Synchronische Produktionssystem, welches die Lean Production auf den Einsatz in Hochlohnländern überträgt (siehe [Dom+15; Her11; Tak12]). Ein Großteil der Unternehmen, insbesondere in der diskreten Klein- und Großserienfertigung und Montage, setzen heute auf die Lean Production basierende Produktionssysteme bzw. deren Prinzipien und Methoden ein [Dom+15; Ger11; Gla+16]. Die Lean Production bildet in einer Vielzahl von Unternehmen den Status quo der Produktionsorganisation. Bedingt durch die Strukturierung und Abstimmung aller Aspekte der Produktion auf ein gemeinsames Ziel hat sich seit den letzten Jahren der Begriff des ganzheitlichen Produktionssystems (engl. Lean Production System; LPS) etabliert, der die Vielzahl an von der Lean Production abstammenden Produktionssystemen zusammenfasst. Seit 2012 existiert die vom Verband Deutscher Ingenieure (VDI) veröffentlichte Richtlinie 2870, die die Struktur, Inhalte sowie Prinzipien und Werkzeuge von ganzheitlichen Produktionssystemen vereinheitlicht [Ver12; Ver13]. Basierend auf dieser Richtlinie ist ein ganzheitliches Produktionssystem im Rahmen dieser Arbeit wie folgt definiert:

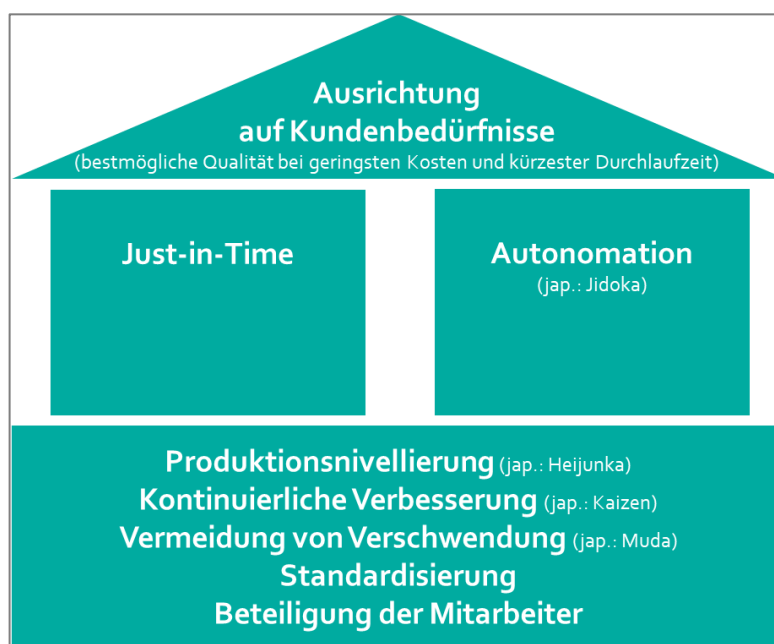


Abbildung 30 Das TPS-Haus und seine Bestandteile (Quelle: [Her11; Lik07])

Ein ganzheitliches Produktionssystem ist ein unternehmensspezifisches Regelwerk (vgl. Betriebsanleitung). Es basiert auf der Lean Production und beschreibt umfassend und durchgängig die Gestaltung der Produktion. Es ist eine auf ein übergeordnetes Ziel abgestimmte Sammlung von Gestaltungsprinzipien und Methoden und gibt eine übergeordnete Struktur vor.

Auch wenn jedes ganzheitliche Produktionssystem unternehmensspezifisch ist, basieren sie alle auf dem TPS und wurden um moderne Konzepte ergänzt [Paw14; Kor+04; Ver12]. Diese Arbeit verwendet im Folgenden als Synonym die englische Übersetzung Lean Production System oder kurz „Lean Production“, um den engen Bezug zu verdeutlichen.

2.2.2 Gestaltungsprinzipien und Methoden

Die zuvor erwähnte VDI-Richtlinie 2870 unterstützt branchenübergreifend Unternehmen unterschiedlicher Größe bei der Einführung von LPS ([Ver12]). Das erste Blatt ist der allgemeinen Struktur und Zusammenhängen im LPS gewidmet. Außerdem befasst sich dieser Teil mit der Einführung und Bewertung von LPS. Das zweite Blatt ist eine Sammlung von Gestaltungsprinzipien und Methoden, die innerhalb der Lean Production zum Einsatz kommen. Neben der bebilderten Beschreibung der Methoden enthält es auch Hinweise zu Vor- und Nachteilen, von der Methode betroffene Rollen und den Einfluss auf die übergeordneten Ziele.

Die VDI-Richtlinie unterteilt die Lean Production in vier Elemente, die über Wirkzusammenhänge miteinander verbunden sind (siehe Abbildung 31). Die Ziele beschreiben einen übergeordneten, angestrebten zukünftigen Zustand. Ein LPS unterscheidet die aus dem TPS bereits bekannten Dimensionen Qualität, Kosten und Zeit. Andere Zieldimensionen sind denkbar, aber im Kontext der Auftragsabwicklung in der Produktion in der Regel den drei zuvor genannten Größen untergeordnet. Ziele wirken auf Unternehmensprozesse, welche u.a. Einzelschritte zur Auftragsabwicklung zusammenfassen. Sie lassen sich unterteilen in Kernprozesse, Unterstützungsprozesse und Führungsprozesse. Gestaltungsprinzipien geben vor, wie die Einzelschritte eines Unternehmensprozesses umzusetzen sind, damit sich ein abgestimmtes Gesamtsystem ergibt. Ein Gestaltungsprinzip ist eine Zusammenfassung ähnlicher Methoden und Werkzeuge, die das vierte Element der Lean Production bilden. Methoden beschreiben eine standardisierte Vorgehensweise zur Zielerreichung, wohingegen Werkzeuge standardisierte Mittel zur Unterstützung dieser Methoden sind. [Ver12]

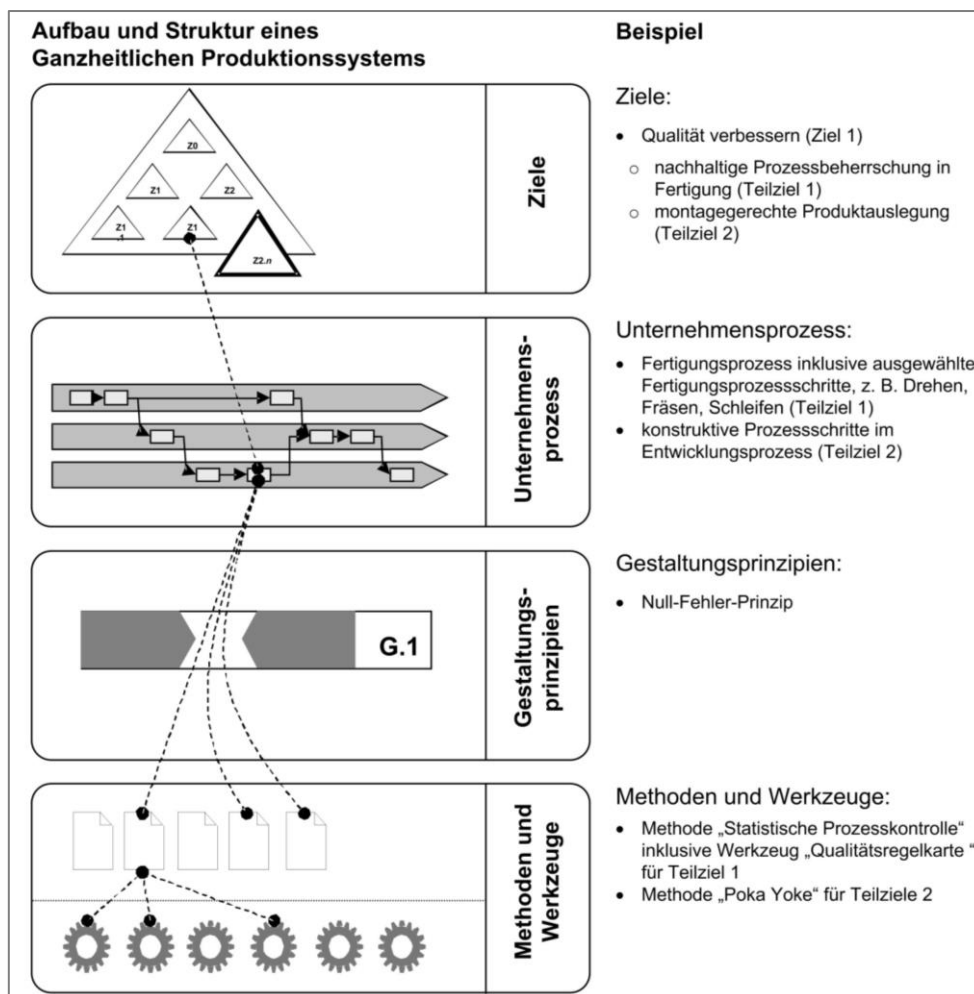


Abbildung 31 Elemente und Zusammenhänge in der Lean Production (Quelle: [Ver12])

Insgesamt beschreibt die VDI-Richtlinie acht Gestaltungsprinzipien mit 35 Methoden und Werkzeugen. Tabelle 1 gibt einen Überblick über diese Methoden sowie im Rahmen dieser Arbeit in anderen Quellen identifizierte weitere Methoden. Im Folgenden wird eine Auswahl relevanter Methoden und Werkzeuge vorgestellt, wobei der Informationsaustausch mit Arbeitsstationen im Fokus steht. Weitergehende Details zu den Methoden und Gestaltungsprinzipien finden sich u.a. in [Ver13], [Ohn93], [Tak12], [Gor+11], [Dic07] oder [Bru11]. Darüber hinaus finden sich weitere Recherchen, Zusammenfassungen und Klassifikationsansätze zu bestehenden Lean-Methoden in [Lag+10], [Jas+14], [Kov+12] oder [Sha+07].

Tabelle 1 Übersicht der identifizierten Gestaltungsprinzipien und Methoden sowie deren Bewertung (Quelle: eigene Darstellung basierend auf [Ver13])

Gestaltungsprinzip	Nr.	Methode	Qualität	Kosten	Zeit
Standardisierung	1.1	5S	##	#	##
	1.2	Prozessstandardisierung	###	##	###
Null-Fehler-Prinzip	2.1	5x Warum	###	#	##
	2.2	8D-Report	##	##	##
	2.3	A3-Methode	##	##	##
	2.4	Autonomation (Jidoka)	###	##	##
	2.5	Ishikawa-Diagramm (Root-Cause Analysis)	##	##	##
	2.6	Kurze Regelkreise	##	#	##
	2.7	Poka Yoke	###	##	##
	2.8	Six Sigma	###	##	##
	2.9	Statistische Prozessregelung	###	#	#
	2.10	Werkerselbstkontrolle	###	##	##
Visuelles Management	3.1	Andon	###	#	##
	3.2	Shopfloor Management	###	###	###
	3.3	Gemba			
Kaizen (KVP)	4.1	Audit	##	##	##
	4.2	Benchmarking	##	##	##
	4.3	Cardboard Engineering	#	###	##
	4.4	Ideenmanagement	###	###	###
	4.5	PDCA	##	##	##
	4.6	Bottleneck-Analyse			
	4.7	Prozess-/ Performancemanagement			
Führung	5.1	Hancho	###	##	##
	5.2	Zielmanagement (Hoshin Kanri)	##	##	##
	5.3	Kaikaku			
Fließprinzip	6.1	First In First Out	##	#	#
	6.2	One Piece Flow	###	###	###
	6.3	Schnellrüsten (SMED)	#	##	###
	6.4	Wertstromplanung	#	##	###
	6.5	U-Layout	#	#	###
	6.6	Taktung			
Pull-Prinzip	7.1	Just in Time/ Just in Sequence	#	##	##
	7.2	Kanban	#	###	###
	7.3	Milkrun	#	##	##
	7.4	Heijunka (Produktionsnivellierung)	#	#	##
	7.5	Supermarkt	#	##	##
Verschwendungsvermeidung	8.1	Chaku-Chaku	#	##	##
	8.2	Low Cost Automation	#	###	##
	8.3	Total Productive Maintenance	#	###	#
	8.4	Verschwendungsbewertung	#	###	##
Anmerkung:					
<ul style="list-style-type: none"> Bewertung gemäß VDI-Richtlinie 2870 Blatt 2 (# = wenig unterstützend; ## = unterstützend; ### = sehr gut unterstützend) Nicht bewertete Methoden sind Methoden, welche im Rahmen dieser Arbeit zusätzlich identifiziert wurden und nicht Bestandteil der VDI-Richtlinie 2870 Blatt 2 sind 					

2.2.2.1 Das Gestaltungsprinzip „Standardisierung“ und seine Methoden

Das Gestaltungsprinzip „Standardisierung“ fasst alle Methoden und Werkzeuge zusammen, die für Produktionsprozesse den Ablauf und die Verantwortlichkeiten festlegen. Standardisierung ist eine der Voraussetzungen zur Einführung von LPS. Sie schafft die notwendige Transparenz und ermöglicht eine kontinuierliche Verbesserung. [Tak12; Ver12]

5S ist neben der Methode der Prozessstandardisierung eine weitere Methode dieses Gestaltungsprinzips. Sie ist vermutlich auch die bekannteste Lean-Methode und wird teilweise als 5A oder 6S bezeichnet. 5S steht für die fünf Prinzipien Seiri (dt. Aussortieren), Seiton (dt. Aufräumen), Seiso (dt. Sauberhalten), Saiketsu (dt. Standardisieren) und Shitsuke (dt. Selbstdisziplin), welche von den Mitarbeitern dauerhaft gelebt werden müssen. Sie führen zu einer Umgebung, die Verschwendung durch unnütze Gegenstände, überflüssige Tätigkeiten und störende Dinge vermeidet. 5S ist eine der ersten einzuführenden Methoden für eine

Lean-Transformation, da sie die Grundlage für andere Aktivitäten bildet und mit wenig Aufwand schnell sichtbare Erfolge erzielt. [Ver13; Tak06]

2.2.2.2 Das Gestaltungsprinzip „Null-Fehler“ und seine Methoden

Das Gestaltungsprinzip „Null-Fehler“ gruppiert alle Methoden, die die Weitergabe von fehlerhaften Teilen an nachfolgende Arbeitsstationen vermeiden und eine hohe Produkt- und Prozessqualität anstreben. Durch eine Fehlerquote von 0% sollen nachgelagerte Kosten für die Fehlerbehebung und Verzögerungen eliminiert werden. [Ver12]

Zu diesem Gestaltungsprinzip gehört u.a. die Autonomation-Methode. Sie strebt an, Maschinen mit Sensoren und Aktoren nachzurüsten, damit Maschinen den Fertigungsprozess überwachen und Korrekturen automatisch vornehmen. Erst im Falle einer Störung soll die Maschine anhalten und ein menschliches Eingreifen anfordern. Um diesen Zustand zu erreichen, muss der Verantwortliche beim Auftreten von Fehlern den Fehler auf seine Ursache hin untersuchen und für eine dauerhafte Abhilfe sorgen. [Ver13]

Poka Yoke ist eine Methode, die Fehler in der Produktion vermeidet. Sobald ein Fehler auftritt, ist im ersten Schritt abzugrenzen, in welchem Prozessabschnitt sich der Fehler befindet. Anschließend ist dieser Abschnitt durch Einsatz anderer Lean-Methoden wie dem Ishikawa-Diagramm oder der FMEA auf die Ursache der Abweichung hin zu untersuchen. Ist die Ursache analysiert, müssen Ansätze zur zukünftigen Fehlervermeidung entwickelt werden. Hierzu enthält Poka Yoke eine Sammlung an Empfehlungen. Beispiele hierfür sind die Vermeidung von Ähnlichkeiten und Symmetrien, um Verwechslungsgefahren zu minimieren (siehe Abbildung 32). [Ver13]

Der 8D-Report ist eine Methode, die im Falle eines aufgetretenen Problems den betroffenen Lieferanten mittels eines standardisierten Formulars alle notwendigen Informationen zur kurz- und langfristigen Fehlerbehebung mitteilt (siehe Abbildung 33). Lieferanten sind hierbei nicht nur im klassischen Sinne Unternehmen außerhalb der eigenen Unternehmensgrenze, sondern auch vorgelagerte Arbeitsstationen in der Produktion, welche Vorprodukte bereitstellen. Zu den durchzuführenden Schritten gehört das Festlegen verantwortlicher Personen für das aufgetretene Problem, die Beschreibung des Problems und der Umstände, das Fest-

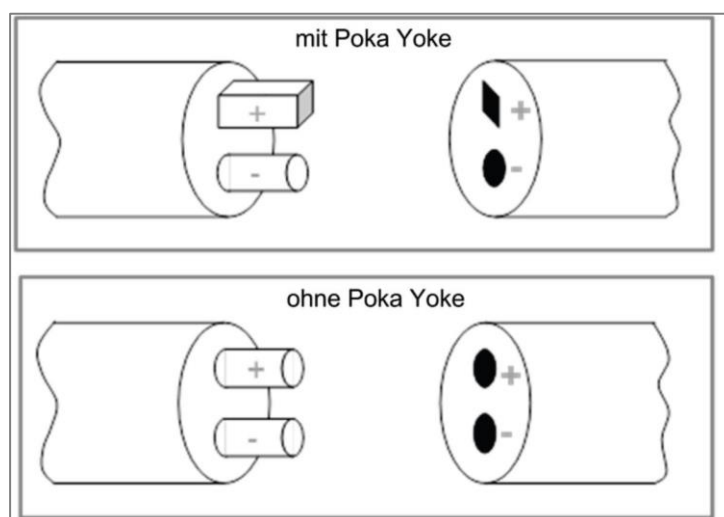


Abbildung 32 Beispiel einer Poka-Yoke-Lösung für eine Steckverbindung (Quelle: [Ver13])

Lieferant (Supplier) <i>Anschrift (Adress/Location)</i>			
8 D – REPORT			
Beanstandung (Concern Title)		Beanstand.-Nr. (Ref. No.)	Eröffnet am: (Start Date)
Berichtsdatum (Status Date)	Teilebezeichnung: (Part Name) Zeichnungsnummer/Index: (Part Number/Index)		
1 Team Name, Abt. (Depmt) Teamleit. (Champ.)	2 Problembeschreibung (Problem Description) Fehlercharakter (Problem Profile Data)		
3 Sofortmaßnahme(n) (Containment Action(s))		% Wirkung (Effect)	Einführungs- datum (Implem. date)
4 Fehlerursache(n) (Root Cause(s))		%Beteiligung (Contribution)	
5 Geplante Abstellmaßnahme(n) (Chosen Permanent Corrective Action(s))		Wirksamkeitsprüfung (Verification)	
6 Eingeführte Abstellmaßnahme(n) (Implemented Permanent Corrective Action(s))		Ergebniskon- trolle (Controls)	Einsatztermin (Implement. date)
7 Fehlerwiederholung verhindern (Action(s) to Prevent Recurrence) Implementation in:		verantwortlich (responsible)	Einführ.termin (Implem.date)
<input type="checkbox"/> Product FMEA <input type="checkbox"/> Process FMEA <input type="checkbox"/> Control Plan <input type="checkbox"/> Pro- cedure			
8 Teamerfolg gewürdigt (Congratulate your Team)		Abschluß- datum (Close Date)	Ersteller (Rep.by) Tel., Fax-Nr.

Abbildung 33 Beispiel für ein standardisiertes 8D-Report-Formular (Quelle: [Ver13])

legen von Sofortmaßnahmen, die Fehlerdiagnose, das anschließende Erarbeiten von Abstellmaßnahmen und die Kontrolle und Würdigung des Erfolges. Hierfür verwendet das verantwortliche Team ein einheitliches Formular, auf welchem es die in jedem Schritt erfassten Informationen dokumentiert. [Ver13; Gor+11]

2.2.2.3 Das Gestaltungsprinzip „Visuelles Management“ und seine Methoden

Das Gestaltungsprinzip „Visuelles Management“ fasst alle Methoden zusammen, welche mithilfe bildlicher Darstellung von Informationen, Prozessen und Prozessergebnissen die Transparenz erhöhen, Motivation verbessern und Fehler sichtbar machen. [Ver12]

„Andon“ bezeichnet im Japanischen ursprünglich eine Laterne. Im Kontext der Lean Production steht diese Methode für die intuitive Darstellung von Soll- und Istwerten bzw. Fehlerzuständen. Dies erfolgt an Arbeitsstationen entweder optisch in Form von Singalleuchten oder

akustisch durch Lautsprecher. Sogenannte Andon-Tafeln sind in der Fertigung gut sichtbar angebrachte Tafeln, die die Informationen aus allen Arbeitsstationen gebündelt anzeigen (siehe Abbildung 34). Die Andon-Reißleine ergänzt oftmals diese Darstellung. Sie ist von allen Arbeitsstationen aus erreichbar, und wenn der Mitarbeiter sie zieht, müssen alle Arbeitsstationen einer Fertigungslinie umgehend anhalten. [Ver13; Ohn93; Tak12]

Shopfloor-Management verlagert Treffen zwischen operativen Mitarbeitern und Führungskräften hin zum Ort des Geschehens (jap. Gemba) in die Produktion. Die Führungskraft soll sich dadurch ein besseres Bild vom tatsächlichen Problem verschaffen, ihre Rolle weg vom Lösungsgeber hin zum Moderator bzw. Coach ändern und Hilfe zur Selbsthilfe liefern. In kurzzyklischen Treffen besprechen die Beteiligten relevante Kennzahlen wie die Ausbringungsmenge, die Fehlerquoten, den Stand von Verbesserungsmaßnahmen und neu aufgetretene Probleme. Hierzu verwendet jedes Team eine Shopfloor-Tafel, welche jederzeit für alle gut sichtbar in der Fertigungslinie platziert ist (siehe Abbildung 35). Für das regelmäßige Treffen aktualisieren die Mitarbeiter die auf der Tafel angezeigten Informationen, indem sie diese z.B. von Anlagen abrufen, aus IT-Systemen entnehmen oder selbstständig bestimmen. [Ver13; Kün16]

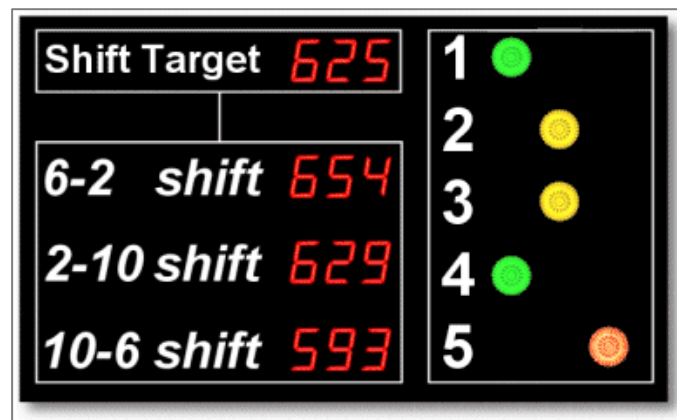


Abbildung 34 Beispiel für eine Andon-Tafel mit Soll-/Ist-Übersicht und dem Status der Arbeitsstationen (Quelle: [Lon93])



Abbildung 35 Beispiel für eine Shopfloor-Tafel in der Produktion (Quelle: [WEI])

2.2.2.4 Das Gestaltungsprinzip „Kaizen“ und seine Methoden

Das Gestaltungsprinzip „Kaizen“, auch „Kontinuierlicher Verbesserungsprozess“ (KVP) genannt, basiert auf dem stetigen Streben nach Perfektion und fasst Methoden zur Identifikation, Planung und Realisierung von Verbesserungsideen zusammen. [Ohn93; Ver12]

Zu den Methoden von Kaizen gehört u.a. der Deming-Kreis, der auch unter dem Namen PDCA bekannt ist. Die Methode beschreibt einen iterativ zu durchlaufenden Verbesserungsprozess, in welchem zuerst ein Plan (P) für eine Maßnahme erstellt wird. Auslöser dieser Planerstellung können gemeldete Probleme oder Verbesserungsvorschläge sein, die z.B. der Reihe nach abgearbeitet werden. Anschließend wird dieser Plan durchgeführt (D) und das Ergebnis geprüft (check; C). Die Prüfung erfolgt anhand zuvor festgelegter Kriterien wie z.B. Kennzahlen. Auf Basis dieser Prüfung wird anschließend agiert (A), indem Nachbesserungen, z.B. in Form von neuen Verbesserungsvorschlägen, geplant werden und sich die Schritte wiederholen. PDCA schafft Transparenz und sichert ein objektives Handeln auf Basis von Fakten, anstatt wilde, sprunghafte Aktivitäten. [Ver13]

2.2.2.5 Das Gestaltungsprinzip „Mitarbeiter- und zielorientierte Führung“ und seine Methoden

Das Gestaltungsprinzip „Mitarbeiter- und zielorientierte Führung“ basiert auf dem Grundgedanken, dass der Mensch in der Produktion die wichtigste Ressource ist und er am ehesten mögliche Verbesserungen kennt. Dementsprechend fasst das Gestaltungsprinzip alle Methoden zusammen, welche eine Kultur der Verschwendungsvermeidung unterstützt. [Ver12]

Hierzu gehören die Methoden Hancho (dt. Teamleiter) und Zielmanagement. Hancho soll die Problemlösungskompetenz des Teams erhöhen, indem der bestqualifizierteste Mitarbeiter die Teamleitung übernimmt und die Qualitätssicherung verantwortet. Zielmanagement entspricht dem „Management by Objective“, wonach die Führungskraft mit jedem Mitarbeiter u.a. klare, spezifische, messbare und realistische Ziele sowie einen Plan zur Erreichung dieser vereinbart. [Ver13]

2.2.2.6 Das Gestaltungsprinzip „Fließprinzip“ und seine Methoden

Unter dem „Fließprinzip“ sind Methoden für einen schnellen, durchgängigen und schwankungsarmen Material- und Informationsfluss zusammengefasst. Die Methoden zielen darauf ab, eine direkte Weitergabe des Bearbeitungsstücks ohne Wartezeit zu ermöglichen. [Ver12]

Neben der bekannten Wertstromplanung gehört auch First-In-First-Out (FIFO) zu den Methoden des Fließprinzips. Es beschreibt, dass Aufträge in der Reihenfolge des Eintreffens und nicht nach Dringlichkeit oder anderen Kriterien zu bearbeiten sind. Hierdurch lassen sich angehäuften Aufträge bei Vollauslastung vermeiden, und ein kontinuierlicher Werkstückfluss entsteht. One Piece Flow beschreibt zusätzlich die Verkleinerung von Losgrößen auf die Stückzahl eins. Die Lean-Methode unterstützt nicht nur die bedarfsorientierte Produktion, sondern minimiert auch Liegezeiten und zeigt Qualitätsfehler auf. [Ver13; Tak12; Ohn93; Tak12]

Die Methode Single-Minute-Exchange-of-Die (SMED; dt. Schnellrüsten) minimiert Rüstzeiten und gewährleistet einen kontinuierlichen Warenfluss. Hierzu protokolliert der KVP-

Verantwortliche die externen und internen Rüstzeiten für eine Anlage. Externe Rüstzeiten sind die Rüstvorgänge, die parallel zur Produktion durchführbar sind wie z.B. das Bereitstellen von Wechselwerkzeugen. Interne Rüstzeiten sind Zeiten, für die ein Anlagenstillstand notwendig ist wie z.B. für das Wechseln des Werkzeugs. Anschließend werden Maßnahmen abgeleitet, die die internen Rüstzeiten externalisieren (siehe Abbildung 36). Außerdem wird geprüft, inwiefern Standardisierung und andere Paradigmen die internen und externen Rüstzeiten noch weiter reduzieren können. Beispielsweise kann nach dem Poka-Yoke-Prinzip mittels Nuten das einzusetzende Werkzeug exakt und ohne Korrektur in die richtige Position geführt werden. Auch kann die Verwendung von Schnellspannsystemen anstatt von Schrauben die Rüstzeiten reduzieren. [Ver13; Tak12; Gor+11]

2.2.2.7 Das Gestaltungsprinzip „Pull-Prinzip“ und seine Methoden

Das Pull-Prinzip fasst alle Methoden zusammen, die die Materialversorgung am Bedarf des Kunden ausrichtet. Arbeiten sind erst dann auszuführen, wenn die nachgelagerte Arbeitsstation diese anfordert. Aufträge werden somit durch die Produktion „gezogen“ und nicht „gedrückt“. Das Pull-Prinzip strebt einen geringstmöglichen Steuerungsaufwand und geringstmögliche Bestände an. [Ver12]

Der Begriff „Produktionsnivellierung“ (jap. Heijunka) beschreibt die Glättung des Produktionsprogramms, indem Kundenaufträge in kleinere, wiederkehrende Fertigungsaufträge aufgeteilt werden. Sie unterstützt die einheitliche Taktung von Fertigungslinien. Die zuvor beschriebene SMED-Methode muss hierfür bereits realisiert sein, da die verringerten Losgrößen eine erhöhte Anzahl an Umrüstvorgängen erfordern. Oft wird die Produktionsnivellierung mithilfe sogenannter Heijunka-Tafeln in der Produktion umgesetzt. Heijunka-Tafeln haben für jede Arbeitsstation eine Spalte, in der die Auftragsreihenfolge ersichtlich ist und der Werker Aufträge nur nach dem First-In-First-Out-Prinzip entnehmen kann bzw. darf. [Ver13]

Kanban (dt. Karte) ist eine weitere, seit der Lean Production bekannte Methode. Sie realisiert eine nachfrageorientierte, dezentrale Produktionssteuerung. Vor- und nachgelagerte Arbeitsstationen bilden paarweise immer einen Regelkreis, in dem sie Bedarfe und Materialien austauschen. Sobald an einer Arbeitsstation X der Bestand an Vormaterial unter den Meldebe-

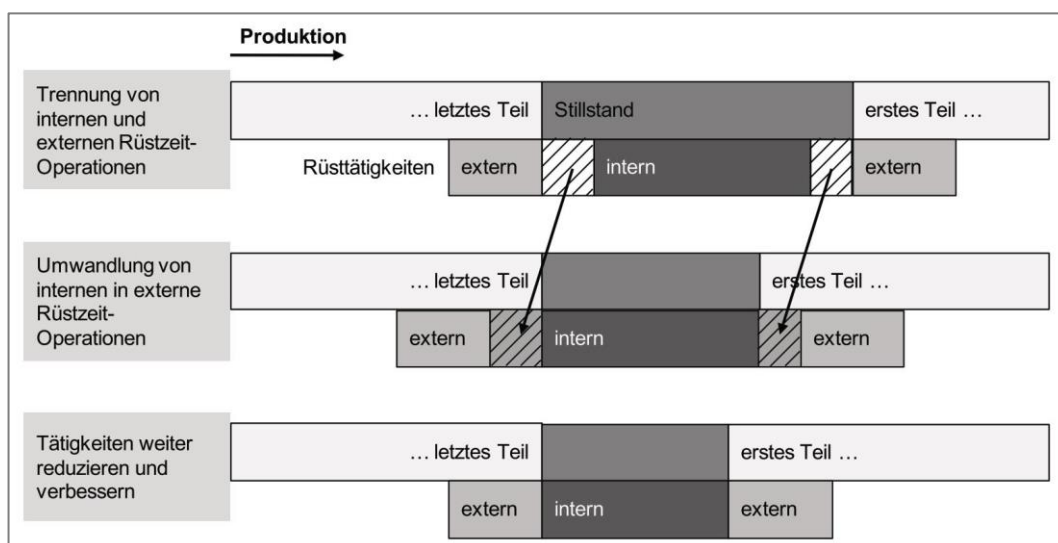


Abbildung 36 Vorgehen zur Rüstzeitenreduzierung mittels SMED (Quelle: [Ver13])

tionen ab, sammelt Kanban und produzierte Waren ein und liefert diese an die Ziel-Arbeitsstationen aus. [Tak12; Ver13]

2.2.2.8 Das Gestaltungsprinzip „Vermeidung von Verschwendung“ und seine Methoden

Das letzte vorgestellte Gestaltungsprinzip „Vermeidung von Verschwendung“ beschreibt Methoden, die die Unternehmensprozesse am Kundenbedarf ausrichten und alle nicht-wertschöpfenden Tätigkeiten eliminieren. Ohno teilte diese nicht-wertschöpfenden Tätigkeiten in die drei Gruppen Muda (Verschwendung), Muri (Überlastung) und Mura (Unausgeglichenheit) ein. Muda wiederum unterscheidet sieben Arten von Verschwendung: Neben der Produktion von nicht benötigten Teilen (Überproduktion), Lagerbeständen, Wartezeiten bei der Bearbeitung und Transporte gehören unnötige Bearbeitungsschritte, unnötige Bewegungen zur Bearbeitung sowie Ausschuss und Nacharbeit dazu. [Ohn93; Ver12]

Unter dieses Prinzip fallen auch Methoden wie Chaku-Chaku oder Low Cost Automation. Chaku-Chaku befasst sich mit der u-förmigen Anordnung von Fertigungslinien mit manuellen Arbeitsstationen. Die erreichte Mehrfachbedienung von Anlagen und Minimierung von Wegen kann die Produktivität um bis zu 25% im Vergleich zur Anordnung als Linie steigern [Bil+12]. Abbildung 39 zeigt hierzu ein Beispiel.

Low Cost Automation, auch unter Low Cost Intelligent Automation (LCIA) bekannt, bezeichnet die einfache, eigenorganisierte Automatisierung. Diese Methode soll Mitarbeiter befähigen, einfache und mit wenig Investition zu realisierende technische Hilfen eigenständig umzusetzen. Insbesondere [Tak04] prägte diese Methode und sammelte Beispiele hierfür.

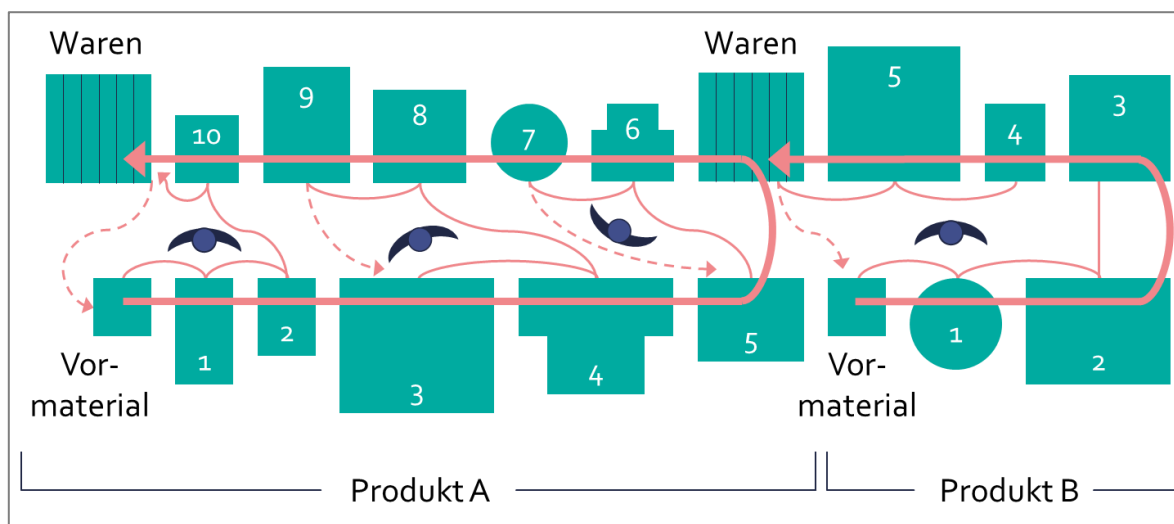


Abbildung 39 Aufbau einer Chaku-Chaku-Linie (Quelle: [Tak12])

2.2.3 Lean Automation

LPS fokussieren Aspekte der Produktionsorganisation und streben eine höchstmögliche Einfachheit an. Entgegen der in der Praxis weitverbreiteten Meinung ist der Einsatz von Automatisierungs- und Informationstechnologie nicht ausgeschlossen. Schon [Ohn93] beschrieb unter dem Begriff „Autonation“ eine Automatisierung aller standardisierten Prozesse. Ba-

sierend auf dem Prinzip der Automation beschreiben [Tak04] und [Tak06] Handlungsoptionen, um die Lean Production in Hochlohnländern einzuführen. Kernaspekte waren u.a. Designprinzipien für günstige, standardisierte und selbsterstellte technische Lösungen zur Automatisierung sowie die Integration von Anlagen in einen gemeinsamen, synchronisierten Werkstückfluss.

Seit Mitte der 1990er existieren darüber hinaus Auseinandersetzungen zu Digitalisierung in der Lean Production (siehe z.B. [Fra93; Gro93; Sch94]). Unter dem Begriff „Lean Automation“ lassen sich seither Aktivitäten zusammenfassen, die die Methoden und Philosophie der Lean Production mit Automatisierungs- und Informationstechnologie kombinieren. Mit dem Aufkommen von Industrie 4.0 sind in der neueren Zeit Anwendungen im Sinne von Lean Automation in der Praxis verfügbar. Eine Auswahl bestehender Lösungen mit Fokus auf ihrer informationstechnischen Integration wird im Folgenden vorgestellt. Weitere Beispiele und Forschungsaktivitäten finden sich auch in [Kol+16] und [Kol+15].

Für die Kanban-Methode existiert eine Vielzahl von Lean-Automation-Anwendungen. Für den digitalen Kanban-Austausch als Alternative zu physischen Karten wurde der Begriff „e-Kanban“ geprägt. [Lag+10] e-Kanban vermeidet durch Digitalisierung verlorene, redundante oder beschädigte Kanban. Ferner verbessert sich die Wandelbarkeit, da für die Anpassung von Losgrößen, Arbeitsplänen und Taktzeiten die Kanban nicht mehr einzeln einzusammeln und neu auszustellen sind. [Dic07a] Aktuell existieren unterschiedliche Lösungen für die Realisierung eines e-Kanban-Systems bzw. Teile dessen. So kann die Identifikation von Materialbedarfen an Arbeitsstationen z.B. mittels mit RFID ausgestatteten, physischen Kanban-Karten erfolgen, welche für die Materialanforderung digital ausgelesen werden und an der Arbeitsstation verbleiben (siehe z.B. [Hab13; Bod+09; Möh+10]). 2009 stellte die Firma Würth Industrie Services GmbH & Co. KG ein solches e-Kanban-System vor. Beim sogenannten iBin handelt es sich um einen mit einem batteriebetriebenen Kamera- und Kommunikationsmodul nachgerüsteten Kanban-Behälter, der den Lagerbestand im Behälter überwacht (siehe Abbildung 40). Fällt der Lagerbestand unter den Meldebestand, versendet iBin automatisch und drahtlos eine Bestellung. Diese schickt iBin entweder an das beim Lieferanten vorhandene ERP-System oder es exportiert sie als CSV-Datei für den Austausch mit anderen Anwendungen. Durch das im Kanban-Behälter nachgerüstete Kameramodul lassen sich Pufferbestände an den Arbeitsstationen reduzieren. Ferner können C-Teile, die sich aufgrund ihres geringen Teilwerts wirtschaftlich nicht für eine Pull-Steuerung eignen, nun eher nachfrageorientiert disponiert werden. Ein Beispiel hierfür sind Schrauben, die der Kanban-Behälter bei Bedarf direkt beim Lieferanten nachbestellt. [Wür13]



Abbildung 40 Das iBin-Behälter- und Bestellsystem
(Quelle: Würth Industrie Services GmbH & Co. KG)

Bei dem von der manufactus GmbH angebotenen e-Kanban-System handelt es sich ausschließlich um eine Softwarelösung. Sie ermöglicht den digitalen Austausch von Kanban zwischen Arbeitsstationen, die Visualisierung der Auslastung und anstehender Aufträge und der Optimierung von Taktzeiten und Pufferbeständen. Das System ist webbasiert und ein Datenaustausch mit Dritten ist via Email- oder Fax-Versand möglich. [man12]

Eher indirekt setzte sich das Forschungsprojekt CyProS mit dem Thema Lean Automation auseinander. Das 2015 beendete Projekt befasste sich mit der Erforschung und Demonstration von CPSen und neuartiger digitaler Technologien in der Produktion. Eines der Ergebnisse war ein bedarfsorientierter Milkrun. Beim klassischen Milkrun versorgt die Logistik in regelmäßigen Intervallen anhand vorliegender Kanban die Arbeitsstationen mit Nachschub bzw. holt Produkte ab. Bei der nachfrageorientierten Variante berechnet eine Software anhand digital gemeldeter Bedarfe echtzeitnah die Fahrzyklen. Die Meldung erfolgt hierbei mittels bestehender Terminals zur Betriebsdatenerfassung (BDE) und dem MES direkt auf das am Logistiksystem vorhandene Tablet. Der Logistiker wiederum quittiert abgeholte und angelieferte Waren, indem er QR-Codes scannt. Für den Datenaustausch innerhalb des Unternehmens zwischen MES, Systemen der Produktionsplanung und -steuerung (PPS) und BDE wurden Webservices eingesetzt. Die Tourenplanung erfolgte bei einem externen Dienstleister, der die Daten durch das Hypertext Transfer Protocol Secure (HTTPS) gesichert per Webservices erhält und die Ergebnisse an die proprietäre Tablet-Anwendung des Logistikers übermittelt. Die nachfrageorientierte Versorgung konnte bei gleicher Liefertreue über 50% der gefahrenen Zyklen und ca. 25% der Wegstrecken einsparen. [Lap+14; Ste+15; Zim+15]

Ein weiteres Ergebnis des Projektes CyProS ist eine digitalisierte Heijunka-Tafel, welche die Wittenstein AG einsetzt. Anstatt täglich neue Auftragskarten den Anlagen zuzuordnen, werden diese digital aus dem MES ausgelesen und auf Monitoren angezeigt. Die Visualisierung erfolgt wie bei der papierbasierten Version mittels Spalten je Anlage und dem FIFO-Prinzip. Neben einem geringeren administrativen Aufwand kann die Arbeitsvorbereitung kurzfristige Eilaufträge in der digitalen Version echtzeitnah einplanen. Abbildung 41 zeigt die ursprüngliche Heijunka-Tafel und die digitale Version. [Ste+15]



Abbildung 41 Herkömmliche (links) und digitalisierte (rechts) Heijunka-Tafel (Quelle: Wittenstein AG)

Die Softwarelösung der Salt Solutions GmbH ist eine Erweiterung für das ERP-System SAP zur Produktionsoptimierung. Sie ermöglicht die Nivellierung der Produktion, Bestimmung von Losgrößen und Analyse von Wertströmen. [Sal16] Ähnliche Funktionalitäten bietet das sphinx open Digital Teamboard der in-integrierte informationssysteme GmbH. Diese Lean-Automation Anwendung digitalisiert das Shopfloor-Management. Auf der digitalen Shopfloor-Tafel lassen sich Produktionskennzahlen einsehen, Wertströme analysieren und Verbesserungen mittels der PDCA-Methode koordinieren. Das System lässt den Datenaustausch mit anderen Produkten des Herstellers zu, die Möglichkeiten zum Datenaustausch mit Drittsystemen sind nicht bekannt. [in-17]

Das 2016 gegründete Startup everlean entwickelte eine digitalisierte Lösung für den KVP. Wie in Abbildung 42 dargestellt, verwenden sie in ihrer Anwendung eine webbasierte Oberfläche, die Mitarbeiter mittels mobiler Endgeräte aufrufen können. Werker können so ortsunabhängig und intuitiv Fehlermeldungen inklusive zugehöriger Dokumente melden. Über eine weitere Oberfläche kann der KVP-verantwortliche Mitarbeiter anschließend die gemeldeten Probleme verwalten und deren Umsetzung gemeinsam im Team koordinieren. Ein Datenaustausch mit Drittsystemen wurde nicht realisiert. [Eur16]

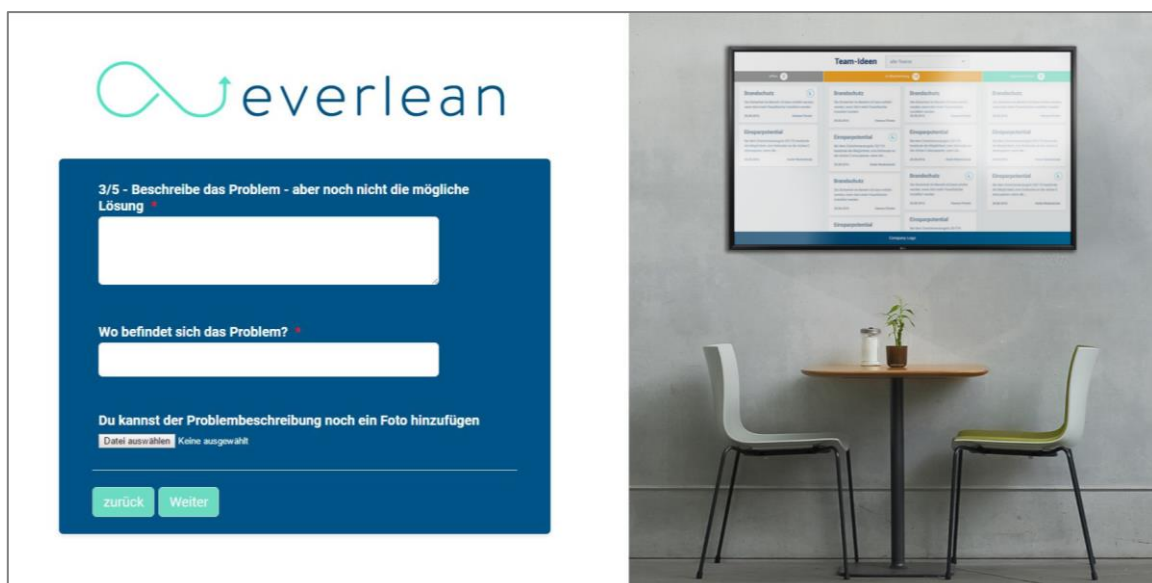


Abbildung 42 Beispiel für die Weboberfläche zur Erfassung (links) und die Verwaltung (rechts) von KVP-Vorschlägen (Quelle: everlean UG)

Bei der Anwendung Andon-Droid handelt es sich um eine Applikation für Android-basierte mobile Endgeräte, um die Andon-Methode zu realisieren. Mitarbeiter einer Arbeitsstation können mittels eines vor Ort installierten mobilen Endgerätes dem Hancho Fehler melden. Dieser wiederum kann mit einem mobilen Endgerät den Empfang bestätigen und eine Rückmeldung geben, wann er an der Arbeitsstation eintrifft. Abbildung 43 stellt die unterschiedlichen Benutzungsoberflächen dar. Durch diese Lean-Automation-Anwendung wird der Informationsfluss verkürzt und Störungen sind ohne Sichtkontakt zur Arbeitsstation erkennbar. Die Kommunikation zwischen den mobilen Endgeräten erfolgt mit einem proprietären Protokoll, sodass auf allen eingebundenen Endgeräten die Anwendung installiert sein muss. [Mic16]

Weitere Forschungsaktivitäten im Kontext von Lean Automation befassten sich mit der Integration von IT- und Automatisierungstechnik in Chaku-Chaku-Linien. Das Projektergebnis war u.a. ein Auftragsverwaltungssystem an der Chaku-Chaku-Linie, das üblicherweise vom ERP-System übernommene Aufgaben im Sinne der Selbstorganisation dem Werker überträgt. Des Weiteren stellte das Konsortium fest, dass Automatisierungstechnik in wertschöpfenden, sich wiederholenden Aufgaben einen Mehrwert bringt. In diesen Tätigkeiten ist die Amortisationszeit kürzer und die Wiederholgenauigkeit und Präzision von Maschinen besser als die von Menschen. Logistiktätigkeiten, unregelmäßige, komplexe Aufgaben und die Fehlerbehandlung sind für die Automatisierung weniger geeignet. [Bil+12] Die Übertragung von Lean-Methoden und -Paradigmen auf den Engineering-Prozess für automatisierte Anlagen [Yam+17], die Roboterunterstützung in Chaku-Chaku-Linien zur Verbesserung manueller Arbeitsstationen bei großen Losgrößen [Fra] und die betriebswirtschaftliche Bewertung von Industrie-4.0-Anwendungen in der Lean Production [Leh16] sind ebenfalls Forschungsaktivitäten im Kontext von Lean Automation. Auch sind eine Richtlinie für die Integration von Automatisierungstechnik in Fertigungs- und Montagebereiche der Automobilindustrie [Zaf13],

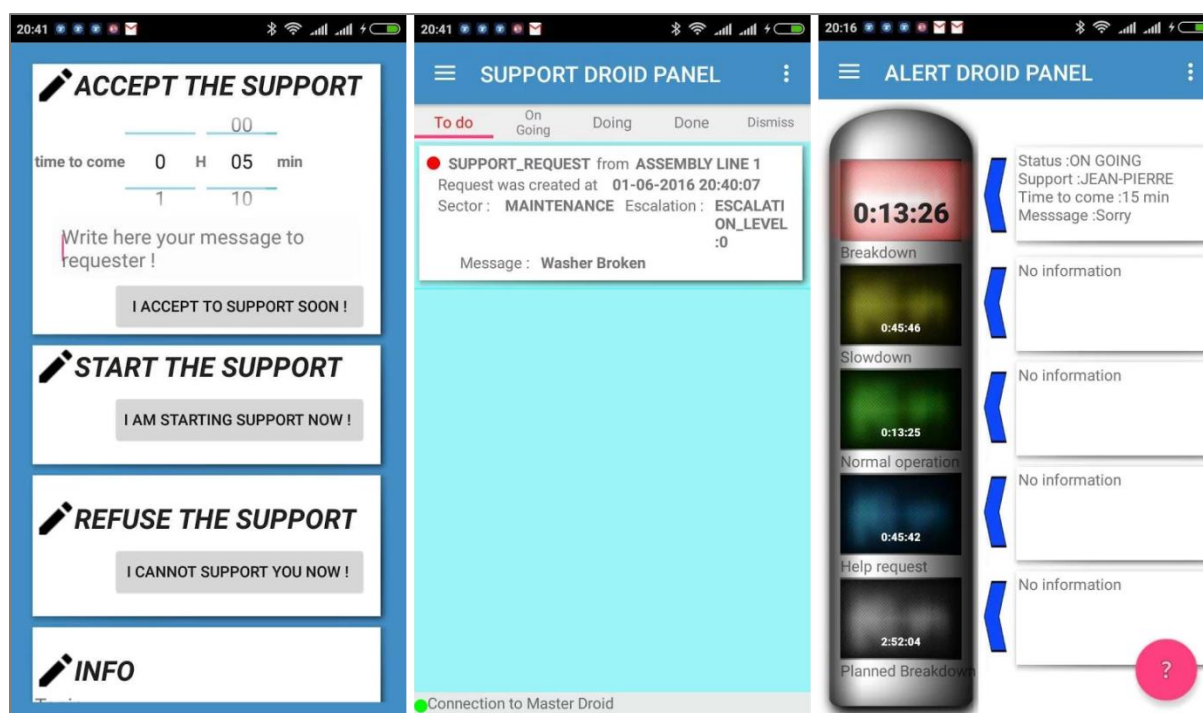


Abbildung 43 Oberfläche der Andon Droid-Anwendung für den Empfang von Störungen (links), deren Bestätigung (mittig) und dem Status einer Arbeitsstation (rechts)

die Konzeption eines CPS-basierten Werkstückträgers für ein Kanban-System [Kol13], eine digitale, cloud-basierte Shopfloor-Tafel [Sch17] und die Realisierung des Andon-Prinzips als Machbarkeitsnachweis für eine ereignisbasierte Kommunikationsarchitektur in der Produktion [The+17] zu nennen.

Die vorgestellten Lean-Automation-Anwendungen bringen durch einen verkürzten Informationsaustausch und automatisierte Prozesse einen Mehrwert in die Lean Production. Wie der Vergleich der Beispiele in Tabelle 2 zeigt, verfügen die bestehenden Lösungen nur über wenige Schnittstellen zu Drittsystemen und die Kommunikation erfolgt primär über proprietäre Protokolle. Dies bedeutet für den Nutzer, dass er für die Umsetzung der Lean-Methoden zwischen den IT-Systemen und Arbeitsstationen mehrere Kommunikationsschnittstellen und evtl. redundante Hardware vorsehen muss. Änderungen an Produktionsprozessen oder neue Technologien erfordern eine aufwendige Anpassung der Lean-Automation-Lösungen. Ferner decken die existierenden Lean-Automation-Anwendungen nicht alle 40 in Kapitel 2.2.2 vorgestellten Lean-Methoden ab. Sie setzen die Digitalisierung um, adressieren aber nicht ausreichend die mangelnde Wandelbarkeit der Lean Production und das Industrie-4.0-Paradigma Modularisierung.

2.2.4 Bewertung des Status quo

Die Verbreitung in der Praxis und wissenschaftliche Studien belegen, dass LPS die Wirtschaftlichkeit der Produktion erheblich steigern. Trotz des Erfolges weist die Lean Production in Hinblick auf zukünftige Marktanforderungen Schwachstellen auf.

Ursprünglich in den 1950ern entstanden, sind die Möglichkeiten moderner digitaler Technologien nicht in den Gestaltungsprinzipien und Methoden von LPS berücksichtigt. Hierzu zählen beispielsweise verkürzte Informationsflüsse, die automatische Datenverarbeitung durch Anbindung an Drittsysteme und die bessere Qualität bei der Erfassung von Prozessinformationen und Fehlern. Das klassische Kanban arbeitet beispielsweise mit physischen Karten, welche aufwendig in der Anpassung sind und abhandenkommen können. Der notwendige physische Transport führt außerdem zu einem verzögerten Produktionsbeginn an der vorge-

Tabelle 2 Vergleich der vorgestellten Lean-Automation-Lösungen

Lean-Automation-Anwendung	Adressierte Lean-Methode	Zielsetzung	Informationstechnische Integration
Würth: iBin	Kanban	Verkürzung und Automatisierung des Informationsflusses für die Materialbestellungen	Proprietäre Anbindung an das Würth-eigene System zur Materialdisposition oder Export als CSV-Datei
manufactus: e-Kanban	Kanban	Verkürzung des Informationsflusses, Visualisierung von Aufträgen und Prozessoptimierung	E-Mail- und Fax-Versand möglich
CyProS: bedarfsorientierter Milkrun	Milkrun; Kanban	Steigerung der Effizienz des Milkruns	Proprietäre Anbindung an das MES der Wittenstein AG
CyProS: digitale Heijunka-Tafeln	Heijunka; Nivellierung	Verringerung der administrativen Aufwände für die Produktionsplanung	Proprietäre Anbindung an das MES der Wittenstein AG
SALT Solutions: Lean-Add On	Nivellierung; Wertstromplanung	Optimierung von Wertströmen, Losgrößen und Nivellierung der Produktion	Anbindung an das ERP-System SAP
In-Informationssysteme: sphinx open Digital Teamboard	Shopfloor Management; Andon; PDCA; Wertstromplanung	Visualisierung von Informationen für Mitarbeiter zur Produktionsüberwachung und -optimierung	Proprietäre Anbindung an die eigene Produktpalette
Everlean: digitales KVP	Ideenmanagement	Digitale Erfassung und Verwaltung von Vorschlägen für den KVP	Keine Anbindung an Drittsysteme verfügbar
Andon Droid	Android	Verkürzung des Informationsflusses für die Störungsbehandlung durch mobile Endgeräte	Proprietäre Anbindung an Instanzen der Anwendung

lagerten Arbeitsstation. Auch müssen Mitarbeiter Shopfloor- und Heijunka-Tafeln täglich aufwendig pflegen.

Standardisierung von Abläufen und Strukturen und daraus resultierend auch von Interaktionspunkten ist eine der Voraussetzungen für die Einführung von LPS. Dennoch spielt die Modularisierung in der Lean Production eine untergeordnete Rolle. Fertigungslinien werden als starre Verkettungen von Arbeitsstationen geplant. Zukünftige, vom Markt geforderte individuelle Produkte und die damit einhergehende stetige Anpassung von Fertigungslinien bzw. -prozessen kann nur durch aufwendiges Umrüsten und manuelles Anpassen von Losgrößen und Taktzeiten erreicht werden. [Rot16]

Auch enthalten LPS einen Zielkonflikt. Die Lean Production strebt durch Methoden wie das Pull-Prinzip eine direkte Kopplung der Produktion an die Auftragseingänge an. Aufgrund der zukünftigen schwankenden Nachfragen und einer steigenden Produktindividualisierung liegt die für z.B. das Just-in-Time-Prinzip notwendige konstante Auslastung nicht mehr vor. Aktuelle Ansätze wie ein Kanban mit belastungsorientierter Auftragsfreigabe (BOA) oder das Material Requirements Planning (MRP; dt. Materialbedarfsplanung) entkoppeln die Marktnachfrage und schaffen eine konstante Fertigung, verbessern allerdings nicht die Wandelbarkeit bei neuen Produkten oder Varianten. [Erl07; Ohn93; Dic07b]

Praktiker kritisieren, dass Automatisierungstechnik und IKT aufgrund ihrer technischen Komplexität nicht den Paradigmen der Lean Production entsprechen und beide Bereiche somit nicht kombinierbar seien. Dies ist insofern falsch, als die Lean Production mit dem Begriff „Automation“ die Automatisierung standardisierter Prozesse explizit anstrebt und die Vorteilhaftigkeit dessen bereits nachgewiesen wurde [Ohn93; Bil+12]. Es lassen sich sogar Übereinstimmungen in den Paradigmen der Lean Production und Industrie 4.0 feststellen: Beide bevorzugen dezentral gesteuerte Strukturen vor komplexen, monolithischen Systemen und streben eine schnelle Anpassung der Produktion an [Tak12; Dic07c; Züh10].

Die vorgestellten Lean-Automation-Beispiele zeigten, dass nicht nur die Forschung die Ergänzung von LPS durch IKT betrachtet. Es existieren für unterschiedliche Lean-Methoden bereits am Markt erhältliche Produkte, welche mittels des digitalen Datenaustausches einen Mehrwert in bestehenden Produktionen liefern. Allerdings handelt es sich bei diesen Lösungen um herstellerspezifische Implementierungen von Software und Hardware, die nicht alle Lean-Methoden abdecken. Zukünftige Änderungen an z.B. Arbeitsstationen oder der Produktions-IT können hier ebenfalls wieder aufwendig Anpassungen der jeweiligen Lösung erfordern. Die mit Industrie 4.0 angestrebte herstellerunabhängige Modularität und Ad-hoc-Integration neuer Technologien ist kaum gegeben. Die Lean-Automation-Lösungen betrachten auch nicht die mangelnde Wandelbarkeit der Lean Production.

Zusammenfassend ist festzuhalten, dass die Lean Production aktuell der Status quo in der Produktion ist, aber nur bedingt für die zukünftigen Marktanforderungen gerecht wird. Das Zielsystem der Lean Production, die Verbesserung von Kosten, Qualität und Durchlaufzeit, muss um das Ziel Wandelbarkeit ergänzt werden [Mou+12]. Digitalisierung und Industrie 4.0 streben eine verbesserte Wandelbarkeit an und liefern hierfür Ansätze, die allerdings noch nicht ausreichend in LPS berücksichtigt wurden. Im Bereich Lean Automation besteht somit noch Forschungs- und Entwicklungsbedarf.

3. Problemstellung, Zielsetzung und Vorgehensweise

Das vorherige Kapitel erörterte den Stand der Technik, relevante Begriffe und bestehende Defizite in den Themenfeldern „Produktionsdigitalisierung“ und „Lean Production“. Aufbauend hierauf leitet dieses Kapitel die für die Arbeit zugrunde liegende Problemstellung (Kapitel 3.1) und das sich daraus ergebende Ziel und Ergebnis (Kapitel 3.2) her. Ferner beschreibt das Kapitel die Vorgehensweise (Kapitel 3.3) zur Erreichung der Teilergebnisse und Beantwortung der wissenschaftlichen Fragestellungen.

3.1 Problemstellung

Megatrends wie kürzere Produktlebenszyklen und der Trend hin zu kundenindividuellen Einzelprodukten stellen an die Produktion der Zukunft neue Anforderungen. In Hinblick auf die Art und die produzierte Menge der Produkte wird die Nachfrage dynamischer und komplexer. [Abe+11; Nyh08; BMB13] Für die Produktion bedeutet dies, dass sie schneller auf sich ändernde Marktanforderungen reagieren und wandelbarer sein muss. Nicht nur der Aufwand für die Rekonfiguration einer Fertigungslinie ist zu minimieren, auch die Integration neuer oder die Veränderung bestehender Arbeitsstationen und Betriebsmittel sollte schnell und mit geringem Aufwand vonstattengehen. [Nyh08]

Aus organisatorischer Sicht ist die Organisation nach Prinzipien der Lean Production der Status quo in der diskreten Klein- und Großserienfertigung [Dom+15; Ger11]. Grund für diese weite Verbreitung sind u.a. die Abstimmung der Gestaltungsprinzipien und Methoden auf ein gemeinsames Ziel, die einfache und kostengünstige Möglichkeit der Umsetzung sowie die signifikante Verbesserung relevanter Kennzahlen wie z.B. Durchlaufzeit, Fehlerquote und Kapitalbindung. Zusätzlich etabliert die Lean Production in einer Produktion standardisierte Prozesse und schafft Transparenz. [Lik+11; Wom+07]

In Hinblick auf die zuvor genannten zukünftigen Anforderungen an die Produktion stößt die Lean Production allerdings an ihre Grenzen. Auch wenn sie Flexibilität zur Fertigung unterschiedlicher Varianten ermöglicht, ist die Wandelbarkeit der Produktion kein adressiertes Ziel [Mou+12]. Das betrifft sowohl die Struktur- als auch die Ablaufsicht. Eine Rekonfiguration von Fertigungslinien bei neuen Produktionsprozessen durch z.B. das Hinzufügen oder die Substitution von Arbeitsstationen ist nicht vorgesehen. Auch sind eine Anpassung an sich ändernde Nachfragemengen und daraus resultierende, geänderte Taktzeiten und Losgrößen nur aufwendig möglich. Des Weiteren ist ein Austausch von Betriebsmitteln und IT-Systemen z.B. bei der Einführung neuer Technologien in der Lean Production nicht vorgesehen. In den 1950ern entstanden, nutzt die Lean Production nicht das Potenzial digitaler Technologien. Hierzu zählen z.B. die Vorteile eines durchgängigen, schnellen Datenaustausches, die Unterstützung des Werkers durch intelligente Werkzeuge oder das selbstständige Agieren von Anlagen. Auch wenn die Lean Production heute der Status quo der Produktionsorganisation ist, wird sie nur bedingt den zukünftigen Anforderungen gerecht.

Die unter dem Begriff „Industrie 4.0“ stattfindenden Entwicklungen befassen sich mit der verstärkten Integration neuartiger digitaler Technologien in die Produktion. Angestrebt wird u.a. eine durchgängige Digitalisierung, Modularisierung und Vernetzung zur Verbesserung der Wandelbarkeit. Technologische Grundlage für die vermehrte Integration von IKT in der Produktion bilden CPSe, welche digital vernetzt mit allen anderen Entitäten in ihrer Umgebung

kommunizieren, autonom in der physischen Welt agieren und den Werker durch eine echtzeitnahe Optimierung unterstützen. [Kag+13; aca14] Praxis und Forschung erkannten das Potenzial von Industrie 4.0 bereits und realisierten einzelne Anwendungen.

Dennoch mangelt es an einem übergreifenden und ganzheitlichen Ansatz, der sich mit der Digitalisierung von wertschöpfenden Anwendungsfällen in bestehenden Produktionsumgebungen befasst. [Clo+09] Anwender stehen heute vor der Herausforderung, dass sie Anwendungsfälle und Prozesse, in denen Digitalisierung einen Mehrwert liefert, selbst identifizieren müssen. Für diese Anwendungsfälle müssen sie anschließend auf Basis bestehender Informationsmodelle und Kommunikationsprotokolle ihre eigene Schnittstelle spezifizieren, um ihre bereits gegebene Produktion durchgängig zu vernetzen. Diese Schritte sind nicht nur aufwendig, sondern benötigen auch eine hohe Fachexpertise. Die Digitalisierung bestehender Produktionsumgebungen wird hierdurch gehemmt.

Digitale Technologien besitzen das Potenzial, die Schwachstellen der verbreiteten Lean Production zu mindern und die in der Praxis bestehenden Produktionsumgebungen zur Erfüllung der zukünftigen Marktanforderungen zu befähigen. Erste Anwendungen, die in der Lean Production digitale Technologien nutzen, existieren bereits. Die am Markt existierenden Anwendungen decken allerdings nicht alle Lean-Methoden ab. Ferner wurden Sie in der Regel unabhängig voneinander entwickelt und basieren auf herstellereinspezifischen Implementierungen mit proprietären Schnittstellen. Die bestehenden Lean-Automation-Anwendungen lassen sich somit nur aufwendig an neue Gegebenheiten anpassen und sind für einen Datenaustausch mit Drittsystemen nur eingeschränkt geeignet (siehe z.B. [Kol+15b]). Aktuell existiert keine Empfehlung, welche Entwickler und Anwender ganzheitlich bei der Digitalisierung ihrer Lean Production unterstützt. Dies betrifft insbesondere die für die Modularisierung und Integration von IKT notwendigen einheitlichen Kommunikationsschnittstellen. [Sch+15]

Die ganzheitliche Betrachtung von Anwendungsfällen, bestehenden Prozessen und gegebener Strukturen zur Definition einheitlicher Kommunikationsschnittstellen für die Lean Production hätte somit das Potenzial, die Schwachstellen der Lean Production zu mindern. Ferner können sie zukünftig Entwicklungs- und Integrationsaufwände für digitale Technologien verringern.

3.2 Zielsetzung

Das Ziel der vorliegenden Arbeit ist eine verbesserte Wandelbarkeit der Lean Production durch einheitliche informationstechnische Schnittstellen. Die hierdurch umgesetzte Modularisierung lässt den Austausch von Arbeitsstationen in Fertigungslinien und den Austausch eingesetzter digitaler Technologien zu. Einheitliche Schnittstellen für die Kommunikation ermöglichen ferner einen digitalen Datenaustausch und reduzieren im Gegensatz zu den aktuell noch proprietären Lösungen die Entwicklungs- und Integrationsaufwände.

Für eine breite Übertragbarkeit stellt diese Arbeit eine technologieunabhängige Referenzarchitektur vor, die informationstechnische Schnittstellen für die Lean Production definiert. Hierfür beschreibt sie die für die digitale Umsetzung von Lean-Methoden notwendigen Dienste und wie Dritte diese flexibel aufrufen können. Die Referenzarchitektur besteht aus drei Teilen. Eine übergeordnete Systemarchitektur beschreibt die Grenze des betrachteten Systems und dessen Struktur. Sie grenzt die mit informationstechnischer Schnittstelle ausgestat-

teten, relevanten Entitätstypen von den nicht relevanten Entitätstypen ab und definiert Aufgaben und Beziehungen zwischen ihnen. Im Informationsmodell für die informationstechnischen Schnittstellen definiert die Referenzarchitektur die für Dritte anzubietenden Dienste, die ausgetauschten Nachrichtentypen sowie deren Inhalte. Im Gegensatz zur Systemarchitektur handelt es sich beim Informationsmodell nicht um die Beschreibung aus Struktursicht, sondern aus Kommunikationssicht. Für den praktischen Einsatz bei Anwendern und als Unterstützung für Entwickler bei der Implementierung sind die Systemarchitektur sowie das Informationsmodell nicht ausreichend. Die Referenzarchitektur enthält daher eine Schnittstellenarchitektur, die eine modulare Softwarearchitektur für die informationstechnischen Schnittstellen definiert. Die Schnittstellenarchitektur kann flexibel auf unterschiedliche Arten von bestehender und zukünftiger Hardware adaptiert werden. Abbildung 44 zeigt die Zusammenhänge zwischen dem Ziel und Ergebnis der Arbeit sowie den zuvor vorgestellten Teilergebnissen.

Um diese Teilergebnisse zu erreichen, ist die Beantwortung folgender wissenschaftlicher Fragestellungen vorgesehen:

1. Welche Entitätstypen sind in den Lean-Methoden involviert und müssen daher eine informationstechnische Schnittstelle anbieten, wie lassen sie sich einheitlich beschreiben und in welchem Verhältnis stehen sie zueinander?
2. Welche Dienste müssen die Entitätstypen zur informationstechnischen Integration der Lean-Methoden anbieten und welche Informationen tauschen die Entitätstypen aus?
3. Wie ist eine informationstechnische Schnittstelle an einem Entitätstyp in der Produktion aufgebaut, um die zuvor identifizierten Dienste einheitlich und auch in bestehenden Systemen nachrüstbar anzubieten?

Neben diesen gestaltungsorientierten Fragestellungen besteht außerdem die untergeordnete, allerdings vorab zu klärende Frage:

4. Wie lassen sich methodisch die benötigten Informationen aus der Lean Production herleiten und in eine größtmöglich allgemeingültige Referenzarchitektur überführen?

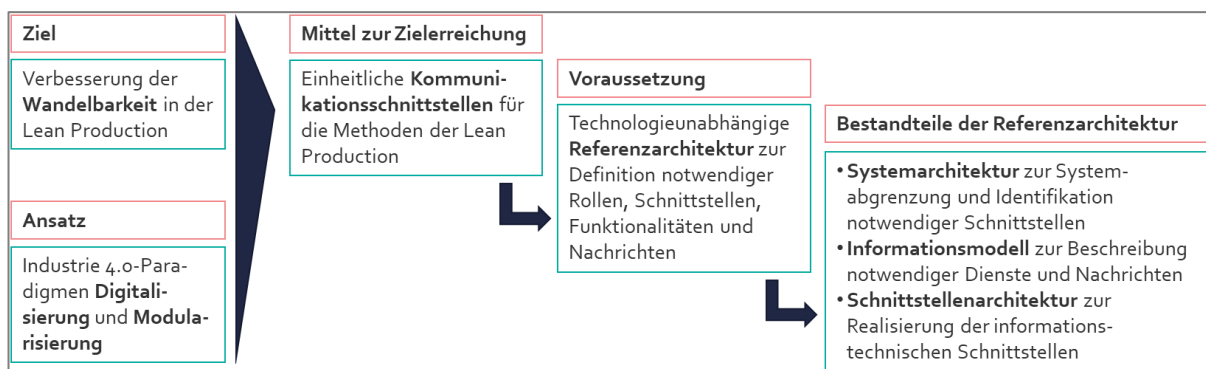


Abbildung 44 Zusammenhang zwischen Zielsetzung und dem Ergebnis der Arbeit

Nicht Betrachtungsgegenstand der Arbeit ist die Entwicklung einer konkreten Schnittstelle für die Umsetzung. Die Referenzarchitektur ist technologieunabhängig beschrieben, um die Übertragbarkeit auf unterschiedliche Produktionsumgebungen zu ermöglichen. Aufgrund der Vielfalt an existierenden Komponenten wie z.B. Feldgeräte und Steuerungen ist die technische Umsetzung nicht allgemeingültig beschreibbar. Kapitel 5 stellt allerdings exemplarisch für eine vorliegende Produktionsumgebung die technische Realisierung vor.

Ferner strebt die Arbeit an, die bestehenden Methoden der Lean Production durch informationstechnische Schnittstellen zu digitalisieren. Die sich dadurch ggf. ergebende Möglichkeit für neue oder optimierte Methoden in der Lean Production ist ebenfalls nicht Betrachtungsgegenstand.

Das Haupteinsatzgebiet der Lean Production ist die diskrete Klein- bzw. Großserienfertigung, in welcher sie auch entstanden ist. Diese Annahme wird für die Abstraktion der Methoden getroffen, um Mehrdeutigkeiten zu vermeiden und ein breites Spektrum an relevanten Methoden realisieren zu können.

Mangels einer allgemeingültigen Spezifikation und Modellierung der Lean-Methoden als Grundlage für diese Arbeit kann diese nicht den Anspruch erheben, allgemeingültig zu sein. Sie zeigt einen möglichen Weg zur Zielerreichung auf Basis einer Vielzahl an berücksichtigter Literatur zum Thema „Lean Production“ auf. Ein flexibel erweiterbar gestaltetes Informationsmodell ermöglicht Unternehmen ferner eine individuelle Anpassung und nachträgliche Erweiterung. Zuletzt ist die betriebswirtschaftliche Evaluation der Referenzarchitektur ebenfalls nicht Bestandteil der Arbeit.

3.3 Vorgehensweise

Wie in Abbildung 45 dargestellt, gliedert sich das Vorgehen zur Beantwortung der Forschungsfragen und Erreichung der Teilergebnisse in drei Schritte. Der erste Teil der Arbeit fasste die bestehenden Vorarbeiten zur Digitalisierung in der Produktion zusammen (Kapitel 2.1). Dies umfasste die Paradigmen von Industrie 4.0 sowie bestehende Konzepte und

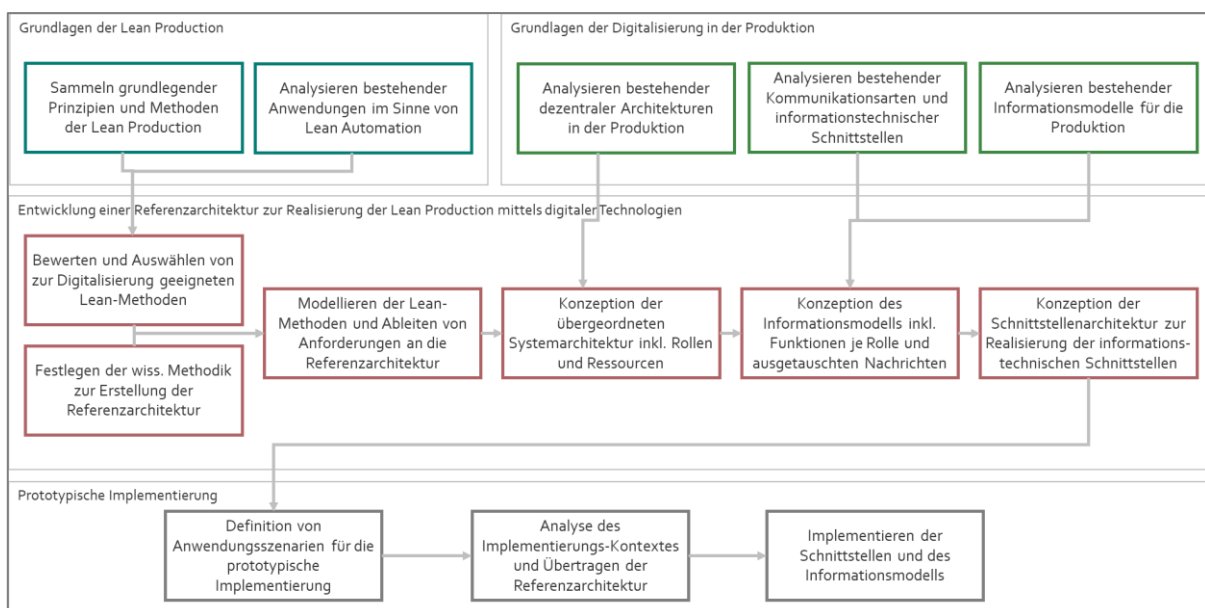


Abbildung 45 Vorgehensweise zur Entwicklung der Referenzarchitektur

Architekturen zur Produktionsdigitalisierung. Des Weiteren wurde der Status quo für informationstechnische Schnittstellen in der Produktion und bestehende Informationsmodelle für die Produktion aufgearbeitet. Die Ergebnisse aus diesem Arbeitsschritt bilden die Wissensgrundlage für die Ausarbeitung der nachstehend beschriebenen Systemarchitektur und des Informationsmodells. Sie fließen außerdem in die Anforderungen an die Referenzarchitektur ein.

Parallel hierzu wurden die Prinzipien und Methoden der Lean Production gesammelt und gesichtet (Kapitel 2.2). Hierbei stand die Identifikation der beteiligten Entitätstypen, deren benötigten Funktionalitäten sowie vorhandenen Informationsflüssen im Vordergrund. Ferner wurden existierende Lean-Automation-Anwendungen untersucht. Die Ergebnisse sind der Hauptbestandteil für die im Folgenden erarbeiteten Anforderungen an die Referenzarchitektur und beschreiben aus organisatorischer Sicht die Produktionsumgebung.

Der folgende Teil der Arbeit befasst sich mit der Entwicklung der Referenzarchitektur, für welche vorab die Methodik für die einzelnen Schritte festzulegen ist (Kapitel 4.1). Im Anschluss werden die Methoden der Lean Production hinsichtlich ihres Potenzials zur Digitalisierung bewertet und die relevanten Lean-Methoden in einem Modell abstrahiert (Kapitel 4.2). Aus den Modellen wird abschließend der Anforderungskatalog für die Referenzarchitektur abgeleitet.

Danach wird die Systemarchitektur konzipiert (Kapitel 4.3.1). Sie beschreibt, wo informationstechnische Schnittstellen in der Lean Production notwendig sind, d.h. sie liefert als Resultat eine Beschreibung der involvierten Entitätstypen und deren Beziehungen untereinander. Aus der Systemarchitektur und den Modellen der Lean-Methoden ergibt sich das Informationsmodell, welches die je Entitätstyp anzubietenden Dienste spezifiziert. Wie diese Dienste von Entitäten realisiert werden, ist abschließend in der Architektur für die informationstechnischen Schnittstellen beschrieben.

Im letzten Teil der Arbeit erfolgt die exemplarische Anwendung der Referenzarchitektur in der Modellfabrik *SmartFactory*^{KL}. Hierzu werden zunächst Anwendungsszenarien definiert, die die Referenzarchitektur verifizieren. Ferner wird der Kontext der Implementierung analysiert (Kapitel 5.1). Die hierauf angepassten informationstechnischen Schnittstellen werden danach in eine bestehende Anlage implementiert und das Informationsmodell darauf instanziiert. Ausgewählte Methoden der Lean Production sind anschließend durch digitalisierte Anwendungen unter Verwendung der informationstechnischen Schnittstellen nutzbar (Kapitel 5.2).

4. Referenzarchitektur zur Digitalisierung der Lean Production

Dieses Kapitel beschreibt die Referenzarchitektur mit ihren Bestandteilen Systemarchitektur, Informationsmodell der Schnittstellen und Schnittstellenarchitektur. Aus der Sammlung, Bewertung und Modellierung existierender Lean-Methoden leitet Kapitel 4.2 den Anforderungskatalog für die Referenzarchitektur ab. Nach dieser problemorientierten Sichtweise nimmt Kapitel 0 anschließend eine lösungsorientierte Sichtweise ein und leitet die Systemarchitektur, das Informationsmodell und die Schnittstellenarchitektur her (siehe Abbildung 46). Das Kapitel 4.1 beschreibt zunächst die hierfür ausgewählten wissenschaftlichen Methoden und Werkzeuge.

4.1 Methodik zur Entwicklung der Referenzarchitektur

Ein stringentes und transparentes Vorgehen unterstützt die Entwicklung der Referenzarchitektur, indem es die Komplexität reduziert und bei den abstrakten Aufgaben unterstützt. Für die jeweiligen Teilschritte zur Erstellung der Referenzarchitektur werden in diesem Kapitel geeignete wissenschaftliche Methoden und Werkzeuge diskutiert und für die einzelnen Bearbeitungsschritte ausgewählt.

4.1.1 Prinzipien der Methodik

Im Gegensatz zu klassischen Produktentwicklungsvorhaben, in denen weitestgehend isoliert ein Produkt entwickelt wird, muss diese Arbeit die Umgebung, auf die die Referenzarchitektur angewandt wird, mit berücksichtigen. Hierbei muss sie ganzheitlich sowohl die technologischen Rahmenbedingungen als auch die durch die Lean Production gegebenen organisatorischen Aspekte betrachten. Darüber hinaus darf die Arbeit die Rolle des Menschen nicht vernachlässigen. Nach [Sen13] handelt es sich somit nicht um ein klassisches Produktentwicklungsvorhaben, sondern um ein Systementwicklungsvorhaben.

Das Systems Engineering ist eine Disziplin, die eine übergreifende Sichtweise einnimmt und sich mit der zielgerichteten Umsetzung von Anforderungen in einer Architektur sowie der Einbettung eines Systems in einen Kontext befasst. [Wei08; Kra+07] Es sieht einen Sachverhalt, unabhängig davon, ob er physisch oder digital ist, als System an. Dieser Sachverhalt grenzt sich mittels einer zu definierenden Systemgrenze von anderen Sachverhalten ab. Entitäten und deren Beziehungen untereinander beschreiben ihn. Falls notwendig, lassen sich die Entitäten wiederum als Subsysteme vertiefen. Für die Beschreibung eines Systems und Subsystems ist nicht relevant, wie seine Entitäten im Detail funktionieren. Lediglich die Eingangs- und Ausgangsobjekte der Entität müssen im Sinne des Black-Box Ansatzes bekannt sein. Durch diesen Ansatz kann ein komplexer Sachverhalt in einem einfachen Modell abge-

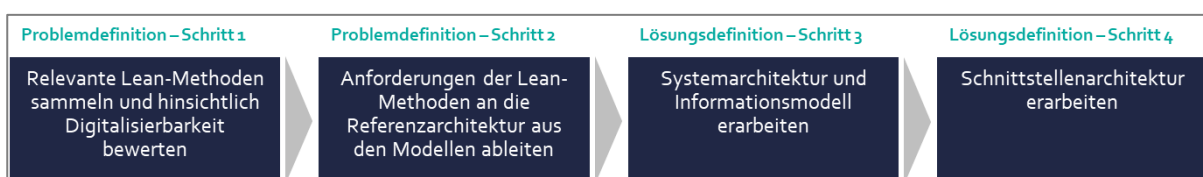


Abbildung 46 Zusammenfassung der Schritte in Kapitel 4

bildet werden. Wichtige Prinzipien und Ansätze des Systems Engineering sind u.a. das Fokussieren auf Zusammenhänge und Wirkungen zwischen den Elementen und ihrer Umgebung, ein hierarchisches Denken ausgehend vom groben, übergeordneten System hin zum Detail sowie eine ausgeprägte Konzeptionsphase. Die Konzeptionsphase beinhaltet die Zielformulierung, die Analyse des Ist-Zustandes sowie die Bewertung und Auswahl einer Lösungsalternative auf Basis von Modellen und schnellen Prototypen. Im Anschluss daran erfolgt erst die Umsetzung. [Hab+12; Wei08]

Das Vorgehen zur Erstellung der Referenzarchitektur orientiert sich am Systems Engineering und an damit verbundenen Disziplinen wie dem Anforderungsmanagement und der modellgetriebenen Entwicklung und verfolgt die zuvor genannten Prinzipien und Ansätze.

4.1.2 Schritt 1: Lean-Methoden sammeln und bewerten

Im ersten Schritt zur Entwicklung der Referenzarchitektur sind existierende Prinzipien und Methoden der Lean Production zu sammeln. Als wissenschaftliches Vorgehen wurde hierbei eine umfassende Literaturrecherche mit der Identifikation grundlegender Arbeiten gewählt. Der Stand der Technik in Kapitel 2.2 und Tabelle 1 fasst die identifizierten Lean-Methoden und herangezogenen Quellen zusammen.

Anschließend sind die identifizierten Prinzipien und Methoden der Lean Production hinsichtlich ihrer Eignung zur Digitalisierung zu bewerten. Ziel ist es, die für die weitere Bearbeitung relevanten Methoden der Lean Production zu sondieren. Geeignete und bedingt geeignete Methoden werden im weiteren Verlauf in die Referenzarchitektur integriert. Die nicht geeigneten Methoden werden nicht weiter betrachtet und explizit dokumentiert, um Nutzer der Referenzarchitektur auf die Grenzen hinzuweisen. Methodisch wird hierbei im Folgenden ein Kriterienkatalog erarbeitet und jede Methode bewertet.

Danach werden die ausgewählten Lean-Methoden detailliert. Hierfür setzt die vorliegende Arbeit konzeptuelle Modelle ein. Modelle unterstützen die Komplexitätsreduzierung, indem sie einen Sachverhalt abstrahieren und nur für die Problemstellung relevante Aspekte visualisieren [Sta73]. Beispiele sind die aus UML bekannten Anwendungsfall- und Aktivitätsdiagramme (siehe [Rup+07]).

Ergebnis des ersten Schrittes ist eine Liste relevanter Methoden der Lean Production sowie deren vertiefende Beschreibung in Form von Modellen. Beide bilden die Grundlage für die Anforderungen an die Referenzarchitektur und definieren die Systemgrenze des betrachteten Sachverhaltes.

4.1.3 Schritt 2: Anforderungen analysieren

Auf Basis der Bewertungsliste und der Modelle wird im nächsten Schritt ein Anforderungskatalog erstellt. Insbesondere in der Informatik ist eine Unterteilung in funktionale und nicht-funktionale Anforderungen verbreitet. Funktionale Anforderungen beschreiben konkrete Fähigkeiten des Systems, die zu erfüllen sind. Nicht-funktionale Anforderungen gehen darüber hinaus und können allgemeine Qualitätseigenschaften und Gestaltungsrichtlinien des Systems beschreiben. [Poh+11]

Die systematische Ermittlung, Spezifikation und Verwaltung von Anforderungen ist Bestandteil des Anforderungsmanagements. [Poh+11] Ein umfassender Überblick über Elemente und Methoden des Anforderungsmanagements, auf welche sich auch das International Requirements Engineering Boards (IREB) bezieht und bestehende Standards wie z.B. die IEEE 830-1998 berücksichtigt, findet sich in [Poh+11] sowie [Bal11]. Nachstehende, für diese Arbeit relevante Inhalte des Anforderungsmanagements sind hieraus entnommen.

Zur Ermittlung der Anforderungen kann eine Vielzahl an Methoden wie z.B. Kreativitätstechniken, Befragungstechniken oder dokumentenzentrierte Techniken eingesetzt werden. Da diese Arbeit auf die zuvor erstellte Liste und die Modelle der Lean-Methoden sowie die identifizierte Literatur zurückgreift, handelt es sich um eine dokumentenzentrierte Technik.

Für eine unmissverständliche Anforderungsdokumentation sollten das Systemumfeld, die Beschreibung von bestehenden Schnittstellen zur Umwelt aus fachlicher Sicht, die geforderten Funktionalitäten sowie die Nutzergruppe dokumentiert werden. Ferner sind getroffene Annahmen und Randbedingungen aufzunehmen. Satzschablonen, welche dem Autor zu verwendende Begriffe vorgeben und Unklarheiten sowie fehlende Informationen vermeiden, unterstützen die Dokumentation der funktionalen Anforderungen. Die Dokumentation der nicht-funktionalen Anforderungen erfolgt in der Regel mittels unspezifischer, aber flexibler Freitexte.

Ergebnis dieses Schrittes ist ein aus den Ergebnissen von Schritt 1 abgeleiteter Anforderungskatalog. Die darin enthaltenen funktionalen Anforderungen werden anschließend in die übergeordnete Systemarchitektur und das Informationsmodell der Referenzarchitektur überführt. Des Weiteren dokumentieren die nicht-funktionalen Anforderungen im Anforderungskatalog u.a. Gestaltungsrichtlinien, die insbesondere in die Systemarchitektur und später in die Schnittstellenarchitektur einfließen.

4.1.4 Schritt 3: Systemarchitektur und Informationsmodell entwickeln

Anschließend wird im Sinne des Top-Down-Ansatzes zuerst die übergeordnete Systemarchitektur abgeleitet. Wie im Kapitel 2.1.2 bereits beschrieben, modelliert eine Architektur Funktionen in einer gegebenen Struktur und lässt sich nach [Kra+07] unterscheiden in Architekturen zur Beschreibung von Strukturen und Architekturen zur Unterstützung der methodischen Erstellung.

Die Systemarchitektur dieser Arbeit beschreibt die Rollen der Lean Production sowie mögliche Rolleninhaber bzw. Komponenten einer Rolle. Neben den Beziehungen untereinander beschreibt die Systemarchitektur ferner, welche Rollen über eine informationstechnische Schnittstelle verfügen. Die erarbeitete Architektur ist somit eine Strukturbeschreibung.

Zum wissenschaftlichen Vorgehen finden sich im Systems Engineering und in der Wirtschaftsinformatik zur Architekturentwicklung zwei mögliche Startpunkte. Sie beginnen entweder mit einer funktionsorientierten Sichtweise, welche zuerst die Abläufe und Funktionalitäten eines Systems beschreibt, oder mit der strukturorientierten Sichtweise, welche statische Zusammenhänge und die involvierten Entitätstypen beschreibt [Hab+12]. Mit welcher Sichtweise zu beginnen ist, ist nicht einheitlich definiert. Die ARIS empfiehlt beispielsweise, zuerst die Abläufe zu definieren, um daraus die Struktur und die Datenaustausche abzuleiten. Sie verwendet ferner einen Top-Down-Ansatz und detailliert die übergeordnete Architek-

tur immer weiter. ARIS überträgt ein technologieunabhängiges Konzept schrittweise auf konkrete, zu implementierende Technologien [Sch02]. Das in der Systems Modeling Toolbox (SYSMOD) beschriebene Vorgehen empfiehlt, auf Basis der Anwendungsfälle und Anforderungen sowie der bestehenden Umgebung zuerst die Interaktionen und Schnittstellen zwischen Entitätstypen zu definieren, bevor anschließend die übergeordnete Struktur modelliert wird [Wei08].

Die Referenzarchitektur verwendet Ansätze aus beiden Vorgehen, um zur Systemarchitektur und dem Informationsmodell zu gelangen. Zu Beginn wurde der Ablauf der Anwendungsfälle einer jeden Lean-Methode modelliert und anschließend daraus Anforderungen abgeleitet. Aus den Ablaufmodellen und den Anforderungen können die involvierten Rollen der Systemarchitektur und die ausgetauschten Informationen für das Informationsmodell inklusive ihrer Attribute extrahiert werden.

Ergebnis dieses Schrittes ist zum einen die Systemarchitektur mit den relevanten Rollen und den notwendigen informationstechnischen Schnittstellen. Zum anderen entsteht in diesem Schritt das Informationsmodell für die informationstechnischen Schnittstellen, das die Dienste einer jeden Rolle sowie ausgetauschte Nachrichten spezifiziert.

4.1.5 Schritt 4: Schnittstellenarchitektur entwickeln

Als Nächstes wird aus der erarbeiteten, technologieunabhängigen Systemarchitektur und dem technologieunabhängigen Informationsmodell die Schnittstellenarchitektur abgeleitet. Die Schnittstellenarchitektur beschreibt, wie sich die informationstechnischen Schnittstellen in den in der Systemarchitektur identifizierten Rollen realisieren lassen. Auch beschreibt sie, wie zwischen den an den informationstechnischen Schnittstellen angebotenen Diensten und den physischen Ressourcen der Entität interagiert werden kann. Die Schnittstellenarchitektur ist ebenfalls unabhängig von konkreten Techniken, schränkt aber den Lösungsraum für Technologien ein. Die Schnittstellenarchitektur bildet das Bindeglied zwischen dem abstrakten Konzept und der konkreten Implementierung. Im Fokus dieser Arbeit steht dabei u.a. die Realisierung der informationstechnischen Schnittstellen mittels der Referenzarchitektur in bestehenden Produktionsumgebungen.

Die in Kapitel 2.1 vorgestellten Technologien und Paradigmen aus dem Kontext der aktuell stattfindenden Industrie-4.0-Entwicklung sind der Ansatz zur Verbesserung der Wandelbarkeit in der Lean Production und somit die Grundlage zur Realisierung der Referenzarchitektur. Aus Hardware-Sicht verwendet diese Arbeit für das wissenschaftliche Vorgehen daher die Paradigmen und Technologien von Industrie 4.0 und CPSen. Retrofitting beschreibt hierbei das nachträgliche Erweitern von Funktionalitäten an Industrieanlagen [Ehr+15]. Die Schnittstellenarchitektur setzt die Idee des Retrofittings um, um die in der Produktion bereits bestehenden Komponenten mit den informationstechnischen Schnittstellen nachzurüsten.

Aus Software-Sicht handelt es sich bei der Schnittstellenarchitektur um eine Softwarearchitektur für verteilte Systeme. Die Softwaretechnik liefert für gängige Fragestellungen und Probleme beim Entwurf von Softwarearchitekturen Empfehlungen. Sie teilen sich u.a. in die Gruppen Architekturstile und Entwurfsmuster auf. Architekturstile beschreiben die Gesamtstruktur und spezifizieren, wie die Subsysteme zusammenarbeiten. Entwurfsmuster hingegen empfehlen für gängige Entwurfsprobleme generische Lösungen und legen die Struk-

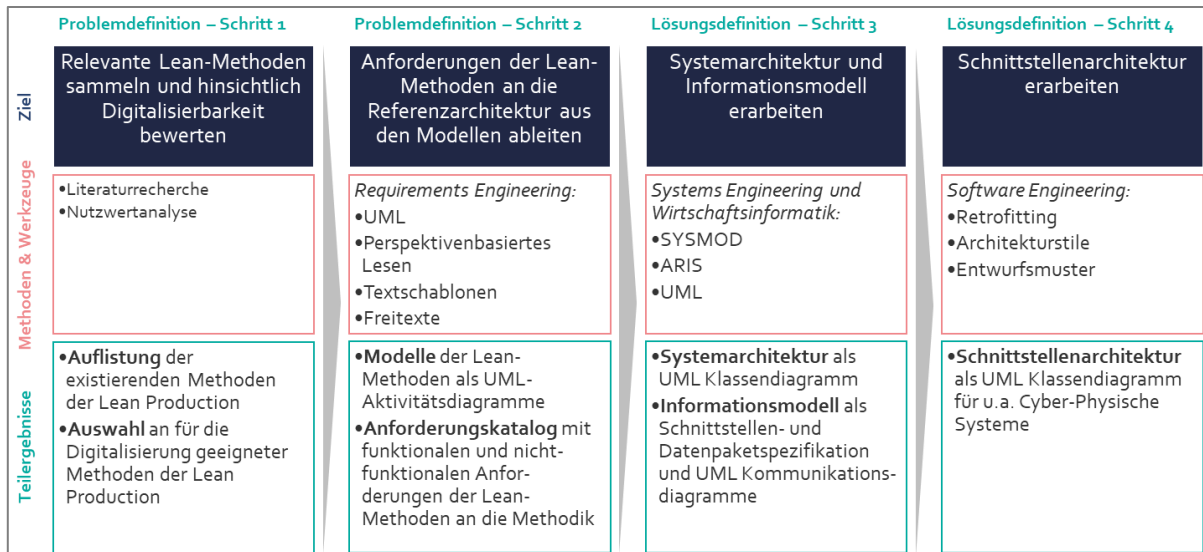


Abbildung 47 Wissenschaftliches Vorgehen und Teilergebnisse zur Erstellung der Referenzarchitektur

tur von Subsystemen fest. [Bal11] Die Schnittstellenarchitektur berücksichtigt diese bestehenden Empfehlungen.

Ergebnis dieses Schrittes ist die Architektur für die informationstechnischen Schnittstellen, welche beschreibt, wie technologieunabhängig die informationstechnischen Schnittstellen umgesetzt werden können. Sie berücksichtigt dabei die Tatsache, dass sich auch bestehende Produktionsumgebungen integrieren lassen. Abbildung 47 fasst die vorgestellten Schritte zur Referenzarchitekturerstellung zusammen.

4.2 Bewertung, Modellierung und Anforderungskatalog der Lean-Methoden

Mit Hilfe der vorgestellten Methoden und Werkzeuge werden im Folgenden die betrachtenden Lean-Methoden hinsichtlich ihrer Eignung zur Digitalisierung und des Mehrwerts durch einheitliche, digitale informationstechnische Schnittstellen bewertet und modelliert. Aufbauend darauf stellt dieses Kapitel anschließend die abgeleiteten funktionalen und nicht-funktionalen Anforderungen vor.

4.2.1 Bewertung und Auswahl von Lean-Methoden

Wie in Kapitel 2.2 beschrieben, liefern u.a. die VDI-Richtlinie 2870 sowie die Arbeiten von Ohno, Takeda, Brunner und Dickmann eine umfassende Sammlung und Beschreibung der Lean-Methoden und -Prinzipien. Für einen möglichst vollständigen Überblick wurden diese Sammlungen mit anderen, bereits in Kapitel 2.2 genannten Quellen verglichen und ggf. ergänzt. Als Ergebnis wurden 40 Methoden zur Unterstützung von acht Gestaltungsprinzipien der Lean Production identifiziert.

Die Bewertungskriterien leiten sich aus dem Ziel der Arbeit ab, d.h. die Bewertung erfolgt hinsichtlich der Eignung der Lean-Production-Methoden zur Digitalisierung, der Verbesserung der Wandelbarkeit durch einheitliche informationstechnische Schnittstellen und des Potenzials für einen verkürzten Informationsaustausch in der Lean-Methode. Eine Lean-

Methode ist zur Digitalisierung für diese Arbeit umso mehr geeignet, je eindeutiger sie die Verantwortlichkeiten und Prozesse (Kriterium 1) beschreibt. Außerdem ist eine Methode, welche ein konkretes Werkzeug beschreibt, besser geeignet als jene, welche eine Verhaltens- oder Gestaltungsrichtlinie definiert (Kriterium 2). Die Lean-Methoden sind zur Zielerreichung geeignet, wenn einheitliche informationstechnische Schnittstellen eine Verbesserung der Wandelbarkeit und eine Verkürzung der Informationsaustauschdauer erwarten lassen (Kriterium 3 und 4). Ferner ist die Digitalisierung der Kommunikation für die Lean-Methoden geeignet, wenn sie diese nicht negativ beeinflusst und Entitätstypen der Produktion wie z.B. Anlagen oder Werkerassistenzsysteme involviert sind (Kriterium 5 und 6).

Die Bewertung der Lean-Methoden hinsichtlich der jeweiligen Kriterien erfolgt auf einer Skala mit der Ausprägung „1 – eher nicht erfüllt“ über „2 – teilweise erfüllt“ bis „3 – eher erfüllt“ nach Sichtung der jeweiligen Methode in den zuvor genannten Quellen. Da sich die Bewertung mit der Sondierung nicht relevanter Lean-Methoden und einer ersten Abschätzung des Potenzials jeder Methode befasst, sind die bei der Bewertung einfließende Subjektivität des Bewertenden und die Oberflächlichkeit der Skalierung von untergeordneter Relevanz. Die Kriterien 4 (kein Verhaltens- oder Gestaltungs-kriterium), 5 (kein negativer Einfluss durch die Digitalisierung) und 6 (Entitätstyp der Produktion involviert) sind hierbei Ausschlusskriterien. Ist eine Lean-Methode zu abstrakt, würden informationstechnische Schnittstellen ihren Erfolg mindern, oder bezieht sich eine Lean-Methode nicht auf die Produktion, wird sie aussondiert und für die Referenzarchitektur nicht weiter betrachtet. Die Bewertung erfolgte unter Berücksichtigung der in Kapitel 3.2 getroffenen Annahme, dass eine diskrete Klein- bzw. Großserienfertigung vorliegt.

Tabelle 3 zeigt das Ergebnis der Bewertung. Die acht Lean-Methoden „Andon“, „Kanban“, „8D-Report“, „Milkrun“, „One Piece Flow“, „Schnellrüsten (SMED)“, „Supermarkt“ und „Taktung“ erreichten die besten Bewertungsergebnisse. Sie sind konkrete Werkzeuge mit eindeutiger Beschreibung von Verantwortlichkeiten und Prozessen, verfügen über das Potenzial, die Wandelbarkeit zu verbessern und die Dauer des Informationsaustausches zu verkürzen, beziehen sich auf die Produktion und werden nicht negativ durch die Digitalisierung beeinflusst. Ferner lässt sich erkennen, dass die Lean-Methoden im Mittelfeld tendenziell deshalb weniger Punkte erhalten, weil es sich um Gestaltungsprinzipien handelt oder keine konkrete, einheitliche Beschreibung von Prozessen vorhanden ist. Bei den Lean-Methoden im letzten Drittel der Bewertung besitzt eine digitale Kommunikation voraussichtlich nicht das Potenzial, sie zu unterstützen. Sie sind sehr abstrakt beschrieben, eher Gestaltungsprinzipien als konkrete Werkzeuge oder informationstechnische Schnittstellen lassen sich in diesen Lean-Methoden nicht sinnvoll umsetzen.

Insgesamt wurden zehn Lean-Methoden von der weiteren Betrachtung ausgeschlossen, da sie die Kriterien nicht erfüllen. „Low Cost Automation“ beschreibt ein Paradigma für die Gestaltung von Automatisierungslösungen. [Tak04] hat hierzu eine Vielzahl an Lösungen beschrieben, die dieses Prinzip realisieren. Nichtsdestoweniger gibt es kein einheitliches Vorgehen, und bei der Methode handelt sich um eine allgemeine Denkweise zum Aufbau von Automatisierungstechnik. „Total Productive Maintenance“ ist ein Ansatz, um Anlagen kontinuierlich vorbeugend instand zu halten. Die beschriebenen Schritte zur Einführung beziehen sich auf die einmalige Einführung innerhalb einer Arbeitsstation und sehen darüber hinaus keinen Informationsaustausch vor. Auch wenn das „Zielmanagement“ klare, aufeinander ab-

Tabelle 3 Ergebnis der Bewertung der Lean-Methoden
(rot markiert = aussortierte Methoden)

Nr.	Lean-Methode	(1) Wandelbarkeit	(2) Informationsfluss	(3) Beschreibung	(4) Werkzeug	(5) Einfluss	(6) Produktionsbezug	Summe	Rang
3.1	Andon	3	3	3	3	3	3	18	1
7.2	Kanban	3	3	3	3	3	3	18	1
2.2	8D-Report	3	3	3	3	2	3	17	3
7.3	Milkrun	2	3	3	3	3	3	17	3
6.2	One Piece Flow	3	2	3	3	3	3	17	3
6.3	Schnellrüsten (SMED)	3	3	3	2	3	3	17	3
7.5	Supermarkt	3	2	3	3	3	3	17	3
6.6	Taktung	3	3	2	3	3	3	17	3
2.4	Autonomation (Jidoka)	3	3	2	2	3	3	16	9
4.6	Bottleneck-Analyse	2	3	2	3	3	3	16	9
6.1	First In First Out	2	2	3	3	3	3	16	9
4.4	Ideenmanagement	2	3	3	3	3	2	16	9
7.4	Heijunka (Produktionsnivellierung)	2	3	2	3	3	3	16	9
4.7	Prozess-/Performancemanagement	3	3	2	2	3	3	16	9
1.2	Prozessstandardisierung	3	3	2	2	3	3	16	9
6.4	Wertstromplanung	2	3	2	3	3	3	16	9
7.1	Just in Time/Just in Sequence	3	2	1	3	3	3	15	17
2.7	Poka Yoke	2	3	2	2	3	3	15	17
3.2	Shopfloor Management	2	2	3	2	3	3	15	17
2.8	Six Sigma	3	3	1	2	3	3	15	17
2.10	Werker selbstkontrolle	2	3	2	2	3	3	15	17
8.2	Low Cost Automation	3	3	1	1	3	3	14	22
2.9	Statistische Prozessregelung	2	2	1	3	3	3	14	22
3.3	Gemba	1	2	3	2	2	3	13	24
2.5	Ishikawa-Diagramm	1	2	2	3	2	3	13	24
2.6	Kurze Regelkreise	2	3	1	2	3	2	13	24
8.4	Verschwendungsbewertung	1	2	1	3	3	3	13	24
4.1	Audit	1	2	2	2	3	2	12	28
8.3	Total Productive Maintenance	2	2	1	1	3	3	12	28
5.2	Zielmanagement (Hoshin Kanri)	2	2	2	1	3	2	12	28
1.1	5S	1	1	1	3	3	2	11	31
4.2	Benchmarking	1	1	2	2	3	2	11	31
4.5	PDCA	1	1	1	3	3	2	11	31
5.1	Hancho	1	1	3	1	1	3	10	34
2.3	A3-Methode	1	1	2	3	1	2	10	34
6.5	U-Layout	1	1	2	2	1	3	10	34
8.1	Chaku-Chaku	1	1	1	1	2	3	9	37
2.1	5x Warum	1	1	1	3	1	2	9	37
4.3	Cardboard Engineering	1	1	1	3	1	2	9	37
5.3	Kaikaku	1	1	1	1	2	2	8	40

gestimmte Ziele anstrebt, lässt die Methode die konkrete Umsetzung offen und ist somit nicht ausreichend konkret. Die Lean-Methode „Hancho“ bezieht sich auf die Organisation von Teams und fordert, dass das bestqualifizierte Teammitglied als Vorarbeiter die Qualität im Team sichert. Diese Methode lässt sich ebenfalls nicht in informationstechnischen Schnittstellen abbilden und wurde aussortiert. Die „A3-Methode“, „5x Warum“ und „Cardboard Engineering“ sind Lean-Methoden, die auf der Interaktion von Mitarbeitern bzw. auf physischen Gegebenheiten basieren. Digitale Kommunikation entspricht in diesen Fällen nicht der Philosophie des visuellen Managements. „Chaku-Chaku“ und das „U-Layout“ beziehen sich auf die physische, strukturelle Gestaltung einer Fertigungslinie und sind somit nicht in der Referenzarchitektur realisierbar. Auch die nachhaltige Implementierung der Lean-Philosophie, „Kaikaku“ genannt, ist kein konkretes Werkzeug, welches sich mit informationstechnischen Schnittstellen unterstützen lässt. Zusammenfassend ist festzuhalten, dass von den 40 Lean-Methoden 30 für die weitere Betrachtung relevant sind. Sie werden im nächsten Schritt durch die Modellierung vertieft.

4.2.2 Modellierung der ausgewählten Lean-Methoden

4.2.2.1 Vorgehens- und Darstellungsweise

Nach der Identifikation und Auswahl der Lean-Methoden können diese anschließend modelliert werden, um im Anschluss daran Anforderungen an die Referenzarchitektur abzuleiten. Die Modellierung verfolgt das Ziel, die Lean-Methoden hinsichtlich der involvierten Entitätstypen, der ausgetauschten Informationen und des einzuhaltenden Prozesses zu untersuchen. Außerdem beschreiben die in diesem Schritt erarbeiteten Ergebnisse die Referenzarchitektur aus einer funktionsorientierten Sicht. Die Prozesse der Lean-Methoden werden nur an den Stellen angepasst, an denen es für eine digitale Kommunikation notwendig ist.

Wie in Kapitel 4.1 bereits beschrieben, abstrahieren die konzeptuellen Modelle die einzelnen Lean-Methoden. Da die Lean-Methoden teilweise unterschiedliche Szenarien umfassen, werden sie im ersten Schritt in Anwendungsfälle unterteilt. Zur Darstellung bieten sich hier die UML Anwendungsfalldiagramme an.

Mithilfe von UML Aktivitätsdiagrammen lassen sich im nächsten Schritt die Abläufe inklusive paralleler Vorgänge und Entscheidungen detaillieren. Neben dem Fokus auf Prozesse umfassen UML Aktivitätsdiagramme Notationselemente, die Objektflüsse und involvierte Akteure darstellen. Sie sind ein gängiges Instrument, das den Ablauf zur Realisierung eines Anwendungsfalls beschreibt und die Grundlage für die Identifikation von Nachrichtenflüssen zwischen involvierten Entitätstypen schafft. [Rup+07] Aus Vereinfachungsgründen stellt diese Arbeit abweichend von der Spezifikation die ausführende Rolle im UML Aktivitätsdiagramm nicht mittels sogenannter Swimlanes dar, sondern nennt sie direkt im Notationselement für Aktivitäten. Auf welche Art und Weise jede Rolle die Aktivität umsetzt, ist gemäß dem Black-Box-Ansatz irrelevant und wird daher als weitere Vereinfachung nicht modelliert.

Abbildung 48 zeigt den Zusammenhang der verwendeten Modelle und wie sich Anforderungen aus einer Lean-Methode ableiten. Die folgenden Unterkapitel erläutern das Vorgehen der Modellerstellung exemplarisch anhand der Lean-Methoden „Kanban“, „Andon“ und „Schnellrüsten (SMED)“. Anhang A listet alle aus den Lean-Methoden abgeleiteten Anwendungsfälle auf. Die vollständigen Modelle aller 30 betrachteten Lean-Methoden sind im Anhang B aufgeführt.

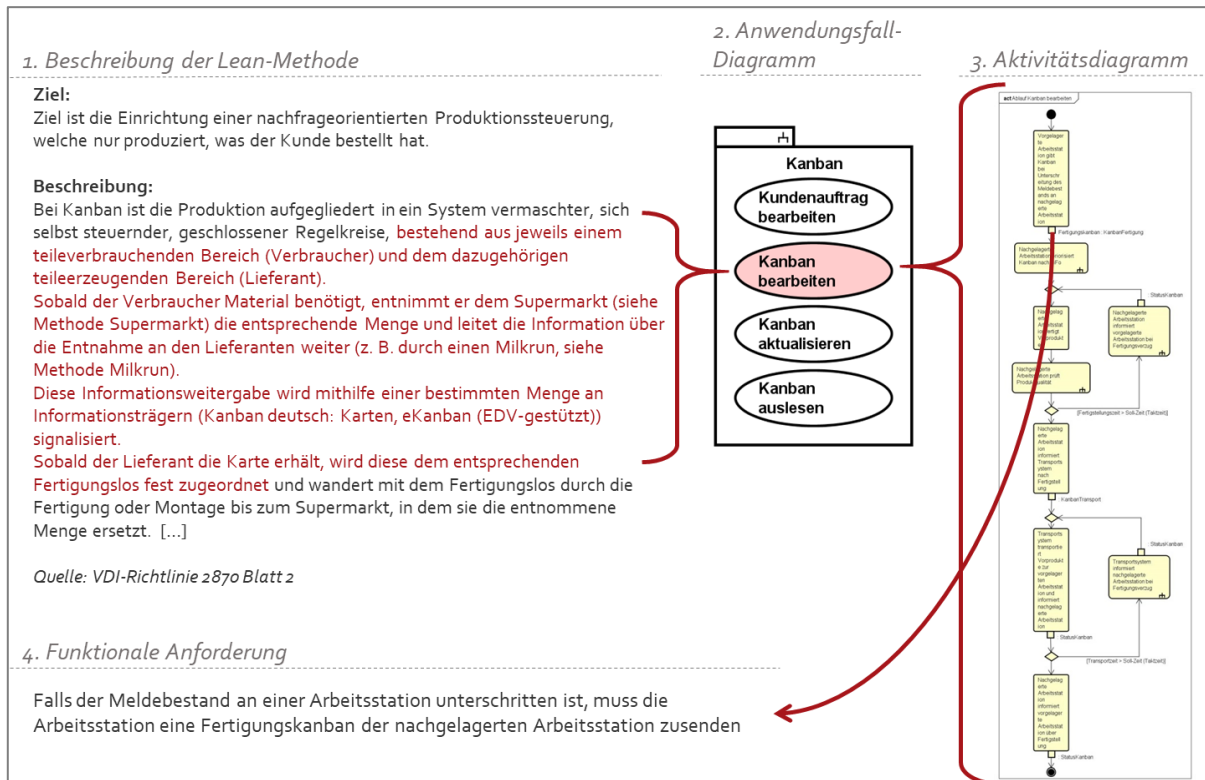


Abbildung 48 Zusammenhang der Lean-Methode, der Modelle und der daraus abgeleiteten Anforderungen

4.2.2.2 Modell der Kanban-Methode

Wie bereits in Kapitel 2.2.2.7 beschrieben, ist Kanban eine Methode zur Produktionssteuerung mittels starrer, dezentral gesteuerter Regelkreise zwischen Arbeitsstationen. Sie ist nicht nur eine der bekanntesten Lean-Methoden, sondern auch ein zentrales Element der Lean Production. Aufgrund dessen existiert für die Kanban-Methode eine Vielzahl an Quellen, welche den Prozess und Informationsaustausch beschreiben.

Die Kanban-Methode umfasst vier Anwendungsfälle, deren Abläufe hinsichtlich des Datenaustausches zu betrachten sind (siehe Abbildung 49). Neben dem Bearbeiten von Kanban und, im Falle der letzten Arbeitsstation einer Fertigungslinie, Kundenaufträgen müssen Arbeitsstationen ferner Kanban auslesen und aktualisieren können.

Wie auch in Abbildung 50 oben dargestellt, gibt die Auftragsverwaltung, in der Regel ein MES oder ERP-System, einen Auftrag an die letzte Arbeitsstation der Fertigungslinie frei und löst damit die Produktion aus. Gemäß dem FiFo-Prinzip bearbeitet die Arbeitsstation anschließend den Kundenauftrag und informiert nach Fertigstellung das Logistiksystem, dass Teile zur Abholung bereitstehen. Dies erfolgt in der Regel mittels einer Kanban oder optisch

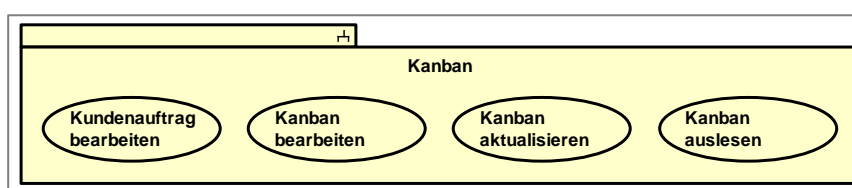


Abbildung 49 Anwendungsfälle der Kanban-Methode

durch belegte Abholflächen. Falls die Arbeitsstation länger als die geplante Taktzeit benötigt, signalisiert sie im Sinne der Andon-Methode den Verzug. Ähnlich verhält sich das Logistiksystem, wenn es den Auftrag nicht binnen der Taktzeit von der Arbeitsstation zum Zielort liefern kann. Für das Auslösen der Produktion zwischen Arbeitsstationen sieht der Ablauf ähnlich aus. Es wird allerdings anstatt des Kundenauftrags eine Kanban ausgetauscht. Auslöser ist hierbei das Unterschreiten des Meldebestandes für Vorprodukte an einer Arbeitsstation.

Ändern sich die Produktvarianten oder Produktionsprozesse, müssen die Kanban an einer Arbeitsstation durch eine aktualisierte Version ausgetauscht werden (siehe Abbildung 50, links unten). Insbesondere für die kontinuierliche Verbesserung und bei Veränderungen der Produktionslinie informiert sich die KVP-verantwortliche Person bei den Arbeitsstationen, welche Kanban diese versenden und empfangen können (siehe Abbildung 50, rechts unten).

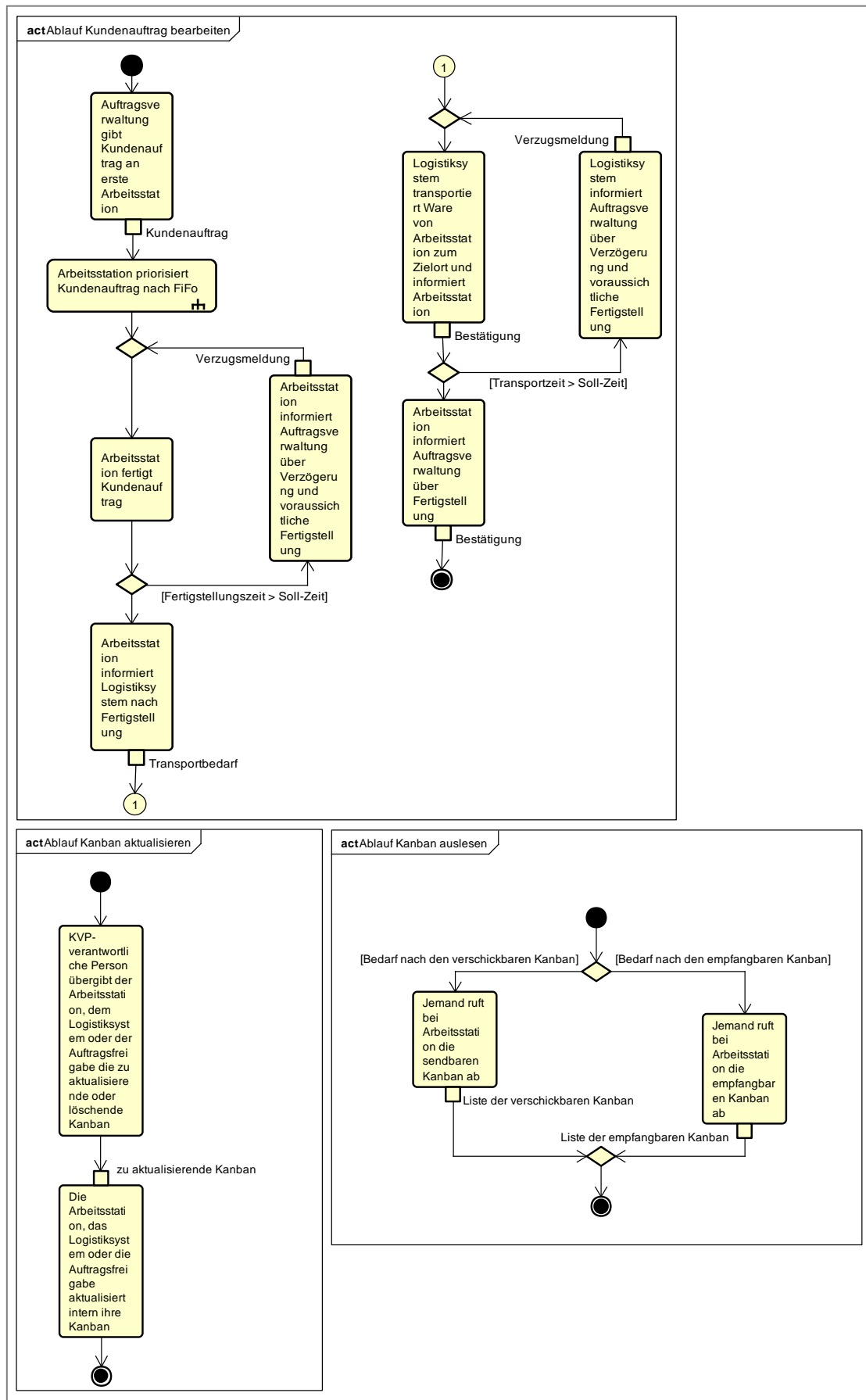


Abbildung 50 Aktivitätsdiagramme für die Anwendungsfälle der Kanban-Methode

4.2.2.3 Modell der Andon-Methode

Andon beschreibt die Darstellung von Soll- und Istwerten bzw. Fehlerzuständen von Arbeitsstationen (vgl. Kapitel 2.2.2.3). Hierbei werden oft auf für alle in der Produktion gut einsehbare Übersichtstafeln zurückgegriffen. Die Andon-Reißleine ist darüber hinaus ein Instrument, das im Falle eines nicht kurzfristig behebbaren Problems die Fertigungslinie umgehend stoppt bzw. Unterstützung durch den Teamleiter anfordert.

Die Andon-Methode setzt sich aus fünf Anwendungsfällen zusammen (siehe Abbildung 51). Es umfasst das Auslesen des Status der Entität, der Bearbeitung einer Störung an einer Arbeitsstation, das Abrufen der Andon-Kennzahlen für die Andon-Tafel sowie für den KVP-Verantwortlichen das Auslesen aller aufgetretenen Störungen und die Aktualisierung der Soll-Kennzahlen.

Zum Auslesen des Status einer Arbeitsstation kann auf das in [Ver13] beschriebene Modell mit fünf Stufen zurückgegriffen werden. Auf Anfrage von Dritten liefert die Arbeitsstation den entsprechenden eigenen Status zurück. Bei der Bearbeitung einer Störung muss eine Arbeitsstation zuerst umgehend den Hancho informieren und die Störung durch einen Statuswechsel visualisieren. Kann der Hancho die Störung nicht beheben, fordert er Verstärkung an. Sobald der Fehler behoben ist, meldet der Hancho dies der Arbeitsstation, und diese wiederum wechselt ihren Status zurück in den ursprünglichen, betriebsbereiten Zustand. Parallel zur Störungsbehebung misst die Arbeitsstation die Zeit, um im Falle einer Taktzeitüberschreitung mittels der Andon-Reißleine das Signal für einen Stopp der Fertigungslinie zu senden. Für das Auslesen der auf der Übersichtstafel darzustellenden Kennzahlen von einer Arbeitsstation muss die Arbeitsstation eine Nachricht versenden können, welche die Ist-Werte zu Lean-relevanten Kennzahlen enthält, d.h. die Nachricht muss mindestens Aussagen zur Qualität, zu Kosten bzw. Aufwänden und zur Durchlaufzeit beinhalten. Gleichzeitig muss die Arbeitsstation aktualisierte Soll-Werte zu diesen Kennzahlen empfangen können. Darüber hinaus muss eine Arbeitsstation dem KVP-Verantwortlichen eine Liste aller aufgetretenen Fehler übermitteln können. Abbildung 52 und Abbildung 53 stellen die Abläufe für die fünf beschriebenen Anwendungsfälle und die zugehörigen Nachrichtenaustausche zwischen Arbeitsstation, Hancho und Fertigungslinie dar.

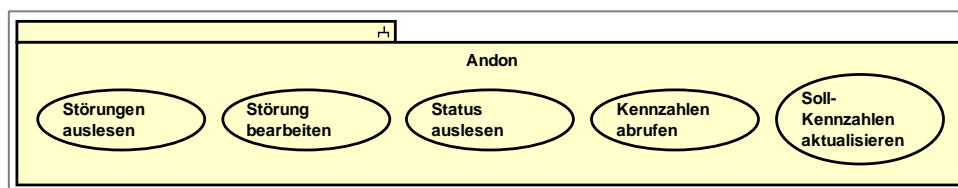


Abbildung 51 Relevante Anwendungsfälle der Andon-Methode

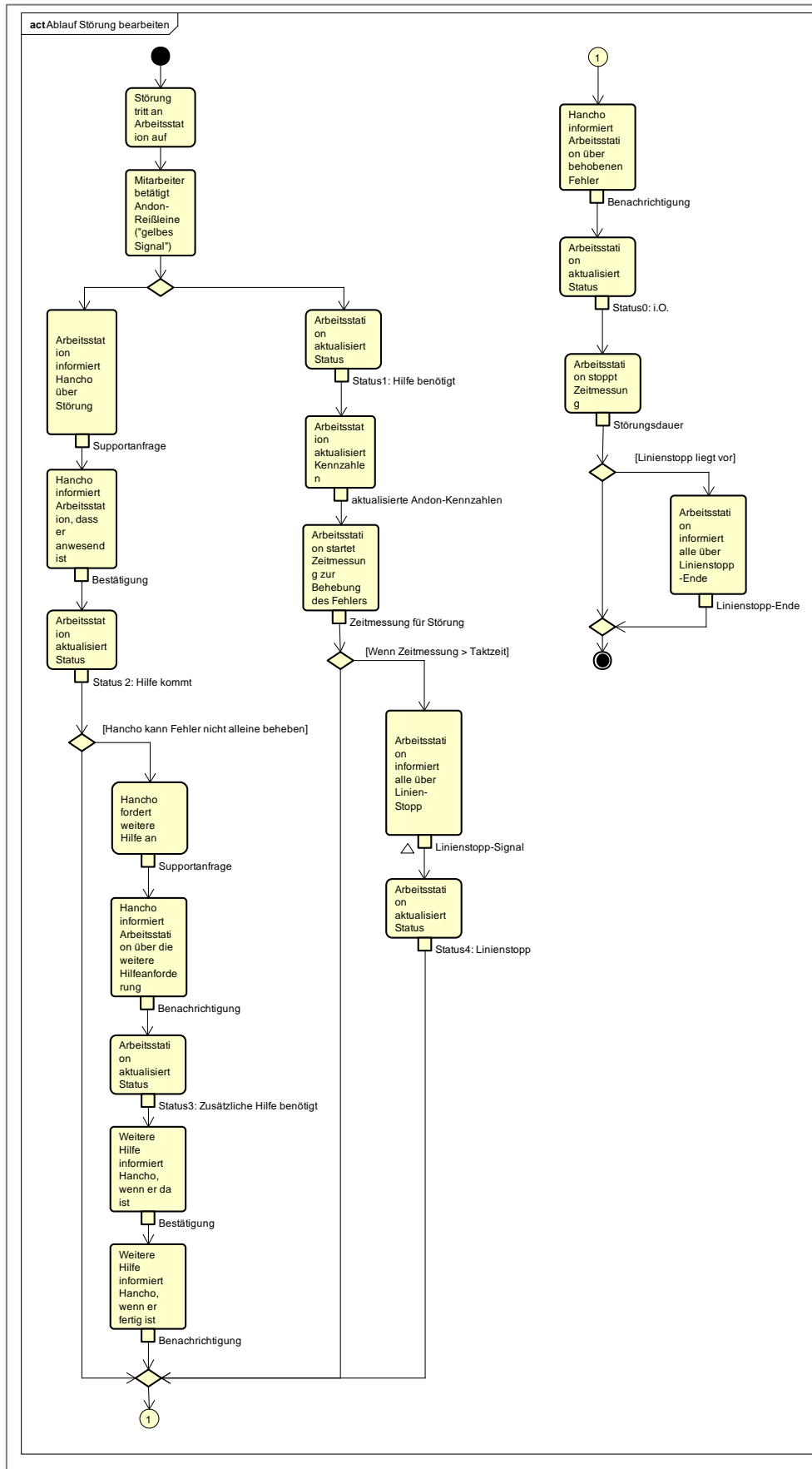


Abbildung 52 Aktivitätsdiagramme für den Anwendungsfall „Störung bearbeiten“ der Andon-Methode

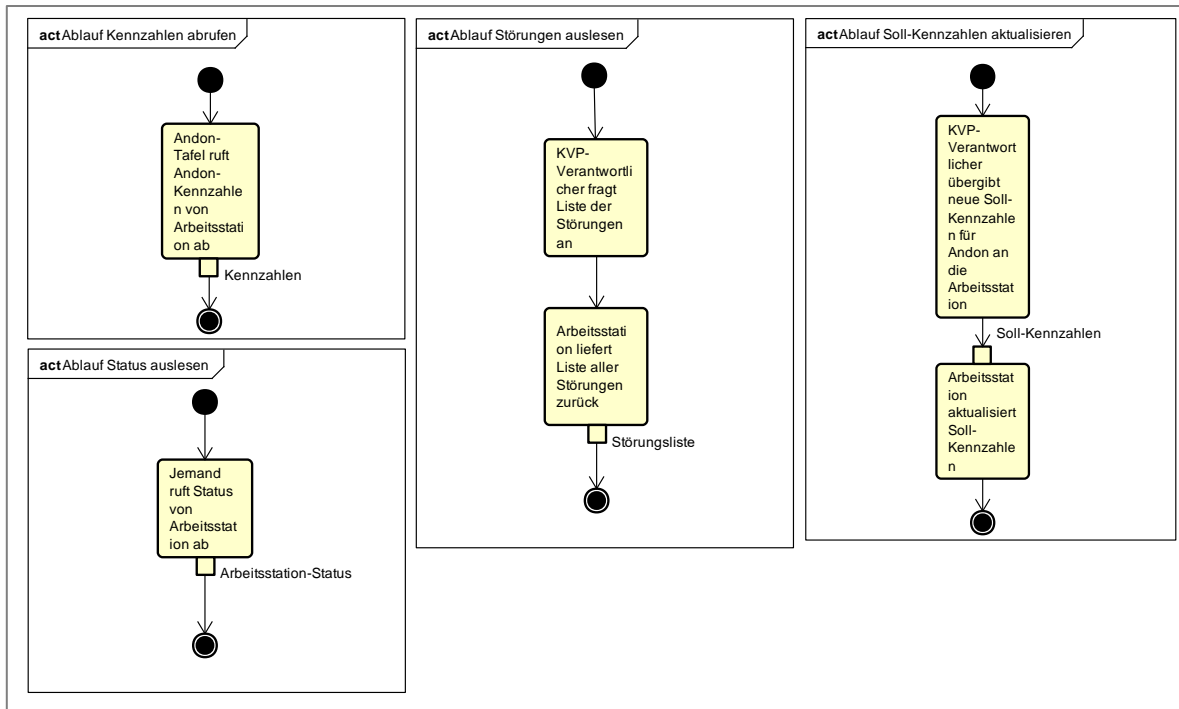


Abbildung 53 Aktivitätsdiagramme der übrigen Anwendungsfälle der Andon-Methode

4.2.2.4 Modell der SMED-Methode

SMED strebt die Verkürzung von Umrüstvorgängen an, um Stillstände an Arbeitsstationen auf das Minimum zu reduzieren (vgl. Kapitel 2.2.2.6). Für die Referenzarchitektur sind nicht die konkreten Tätigkeiten zur Rüstzeitreduzierung relevant, sondern die zu erfassenden Informationen zur Analyse. Für die Modellierung wurde die SMED-Methode in die Anwendungsfälle „Rüstvorgang durchführen“ und „Rüstzeiten auslesen“ unterteilt (siehe Abbildung 54).

SMED unterteilt im Anwendungsfall „Rüstvorgang durchführen“ in externe Rüstoperationen, die parallel zur Laufzeit einer Arbeitsstation stattfinden können, und interne Rüstoperationen, für die ein Anlagenstillstand notwendig ist. Bei einem Rüstvorgang muss eine Arbeitsstation somit mindestens vier Informationen erfassen: Die Dauer des externen Rüstvorgangs vor dem Stillstand, die Dauer des Stillstands selbst, die externe Dauer nach der Wiederinbetriebnahme und das zugehörige Produkt bzw. die Kanban, welche den Rüstvorgang auslöste. Beim Anwendungsfall „Rüstzeiten auslesen“ liefert die Arbeitsstation dem KVP-Verantwortlichen eine Liste mit gesammelten Rüstvorgängen, die er anschließend analysieren kann. Zu dieser Analyse kann z.B. die Auswahl der Produkte mit der längsten Rüstzeit gehören, für die anschließend der KVP-Verantwortliche Verbesserungsmaßnahmen mithilfe der PDCA-Methode festlegt. Abbildung 55 beschreibt die je Anwendungsfall ablaufenden Aktivitäten.

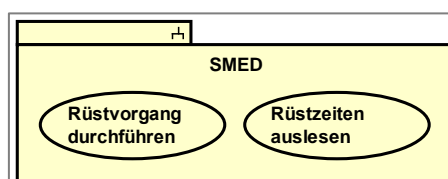


Abbildung 54 Relevante Anwendungsfälle der SMED-Methode

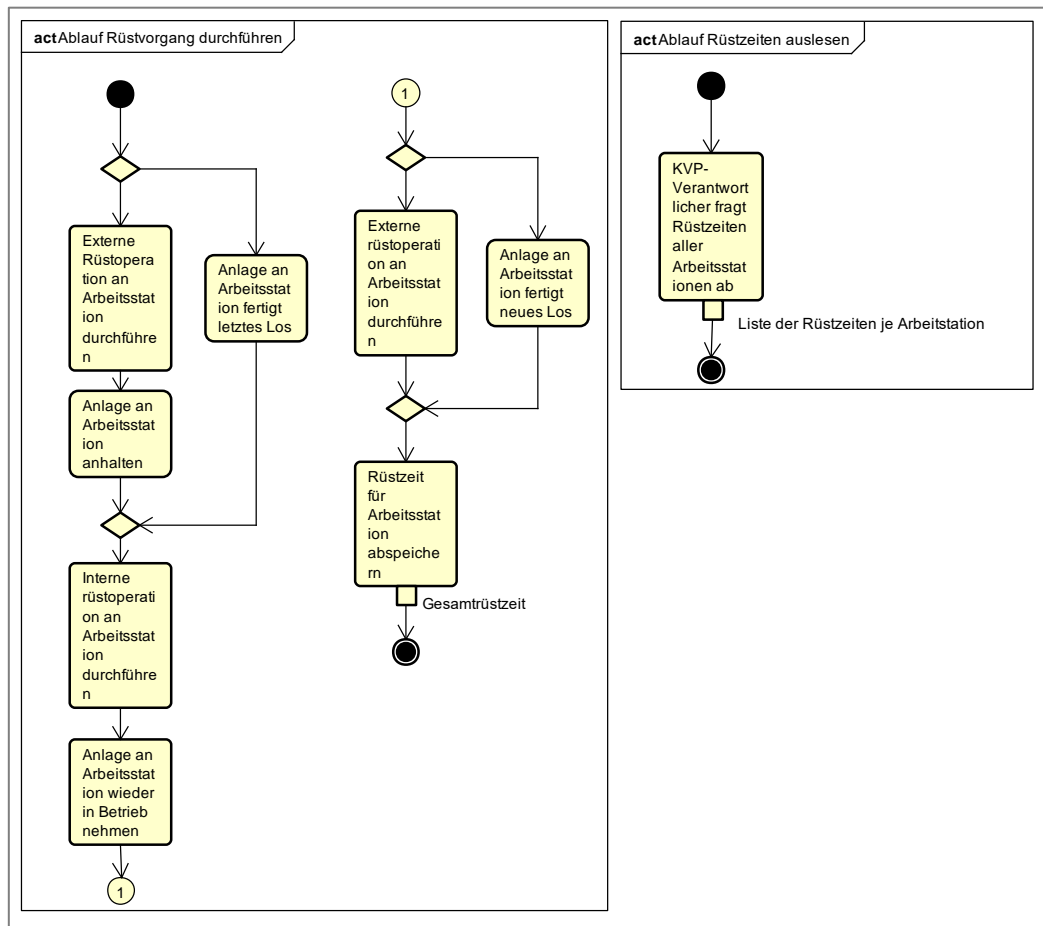


Abbildung 55 Aktivitätsdiagramme für die Anwendungsfälle der SMED-Methode

4.2.2.5 Zusammenfassung

Zusammenfassend wurden für die 30 als für die Referenzarchitektur relevant identifizierten Lean-Methoden insgesamt 60 Anwendungsfälle identifiziert (siehe Abbildung 56). Aktivitätsdiagramme detaillierten anschließend jeweils die Anwendungsfälle. Bei der Modellierung der stattfindenden Prozesse standen der Informationsfluss und die beteiligten Akteure im Fokus. Im nächsten Schritt werden aus diesen Modellen die funktionalen Anforderungen abgeleitet.

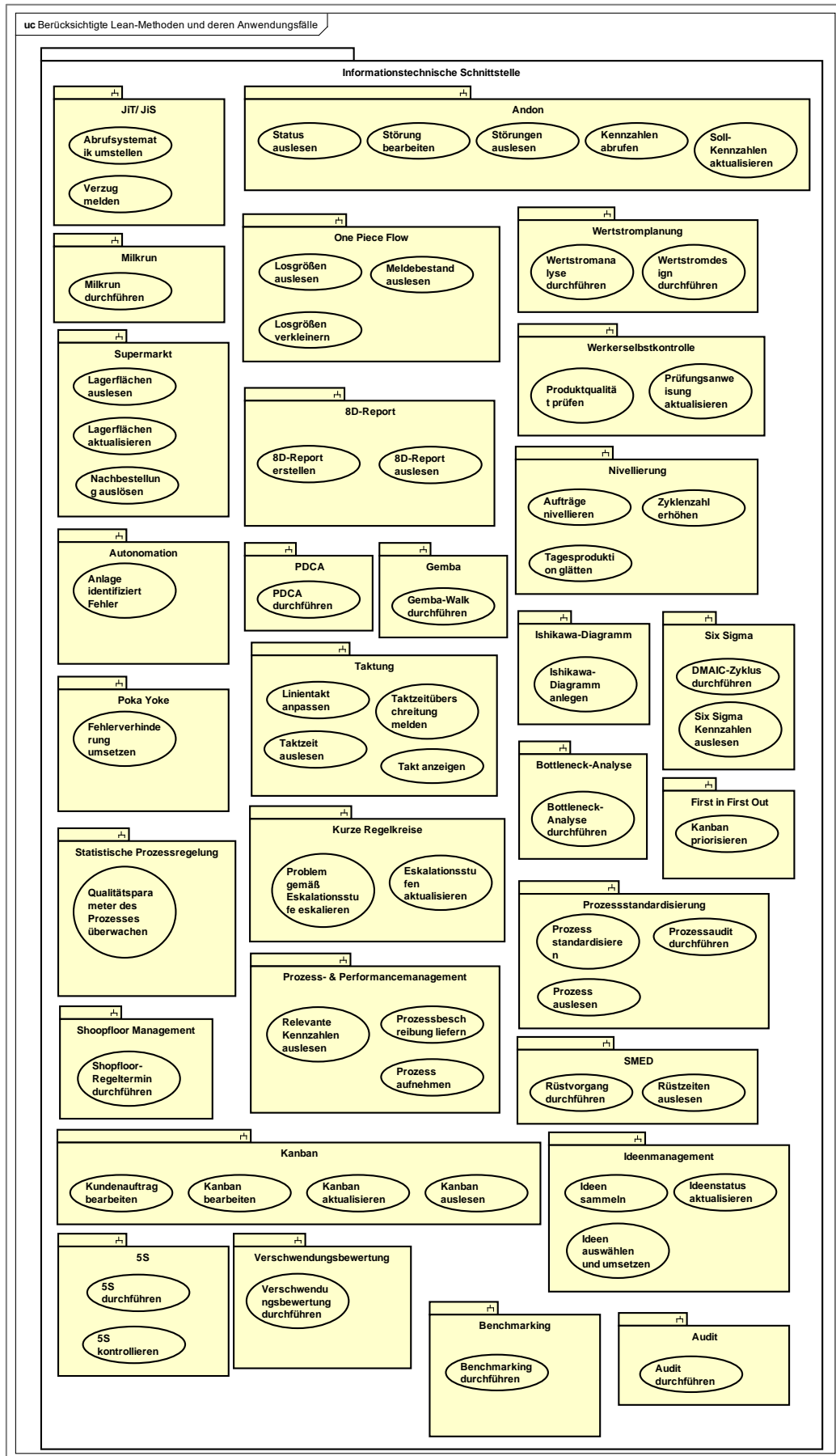


Abbildung 56 Die 30 in der Referenzarchitektur zu berücksichtigenden Lean-Methoden mit ihren 60 Anwendungsfällen

4.2.3 Funktionale Anforderungen des Anforderungskataloges

Die zuvor modellierten Prozesse beschreiben die Abläufe je Lean-Methode. Jeder Prozess ist hinsichtlich stattfindender Informationsflüsse zwischen den Entitäten zu untersuchen, um hieraus das Informationsmodell abzuleiten. Falls ein Informationsaustausch stattfindet, sind der Inhalt dieser Nachricht sowie die involvierten Entitäten zu spezifizieren. Durch die Dokumentation in Form von funktionalen Anforderungen kann die Vielzahl an Informationsaustauschen strukturiert gesammelt und in die Referenzarchitektur überführt werden.

Die Beschreibung funktionaler Anforderungen erfolgt anhand der zuvor bereits erwähnten Satzschablonen aus dem Anforderungsmanagement. Die im Kontext dieser Arbeit verwendete Textschablone enthält Angaben zum Bedarfszeitpunkt einer Information (zu welchem Zeitpunkt bzw. unter welcher Bedingung), zur Verbindlichkeit (muss oder sollte), zur aktiv involvierten Entität (Subjekt), zur Art der Interaktion (selbstständig aktiv werden oder jemandem die Möglichkeit bieten), zur betroffenen Entität (was, wo und/oder womit) sowie zur Tätigkeit (als Prozesswort bzw. Verb). Aus dieser Satzschablone ergeben sich somit menschenlesbare Sätze der Form:

<WANN (sobald.../falls...)>
 <VERBINDLICHKEIT (muss/sollte)>
 <WER>
 <BENUTZERINTERAKTION ((leer)/fähig sein, .../... die Möglichkeit bieten)>
 <OBJEKT>
 <PROZESSWORT>

4.2.3.1 Funktionale Anforderungen der Kanban-Methode

Für die Kanban-Methode ergeben sich aus den zuvor vorgestellten Modellen die in Tabelle 4 abgebildeten Anforderungen an die Referenzarchitektur. Demnach muss eine Arbeitsstation der vorgelagerten Arbeitsstation eine Fertigungs-Kanban zusenden können, falls der Meldebestand unterschritten wird. Diese Kanban muss u.a. die Adresse der Quell- und Senke-Arbeitsstation, die Produktbezeichnung und -nummer, die Anzahl der Teile und den Werkstückträgertyp beschreiben. Des Weiteren muss eine Arbeitsstation die Möglichkeit anbieten, den geänderten Status (z.B. „Lieferverzug“ oder „Erledigt“) zu einer selbst verschickten Kanban zu empfangen. Um die Abholung des fertiggestellten Teils zu veranlassen, muss eine Arbeitsstation dem Logistiksystem einen Transportauftrag übermitteln können. Dieser Transportauftrag muss den Abhol- und Lieferort, die Art und Menge der zu transportierenden Werkstückträger und den Lieferzeitpunkt enthalten. Ferner muss sie auf Anfrage eine Liste der versendbaren und empfangbaren Kanban zur Verfügung stellen und vom KVP-Verantwortlichen vorgenommene Kanban-Änderungen umsetzen.

Tabelle 4 Funktionale Anforderungen der Kanban-Methoden

FR-ID	Ausgefüllte Satzschablone
81	Falls der Meldebestand unterschritten ist, muss die Arbeitsstation die Kanban der vorgelagerten Arbeitsstation zusenden .
82	Falls ein Kundenauftrag freigegeben wurde, muss die Auftragsverwaltung den Auftrag der letzten Arbeitsstation zusenden .
83	Falls die Konfiguration abgefragt wird, muss die Arbeitsstation fähig sein, die von der Arbeitsstation verarbeitbaren Kanban bereitzustellen .
84	Falls der KVP-Verantwortliche die Konfiguration aktualisiert, muss die Arbeitsstation fähig sein, die Daten der betroffenen Kanban zu aktualisieren .
85	Falls die Kanban fertig bearbeitet wurde, muss die Arbeitsstation dem Logistiksystem einen Transportauftrag zusenden .
86	Falls die Fertigstellungszeit einer Kanban überschritten ist, muss [die Arbeitsstation; das Logistiksystem] die Verzögerung dem Auftraggeber der Kanban zusenden .
87	Falls die Bearbeitung einer Kanban fertig ist, muss [die Arbeitsstation; das Logistiksystem] die Fertigstellung dem Auftraggeber der Kanban zusenden .
88	Falls die Konfiguration abgefragt wird, muss die Arbeitsstation fähig sein, die von der Arbeitsstation verschickbaren Kanban bereitzustellen .

4.2.3.2 Funktionale Anforderungen der Andon-Methode

Aus dem zuvor vorgestellten Aktivitätsdiagramm für die Andon-Methode leiten sich die in Tabelle 5 beschriebenen Anforderungen an die Kommunikation zwischen den beteiligten Entitätstypen ab. Die ausgetauschten Nachrichten beziehen sich auf die Andon-Tafeln, welche Soll- und Ist-Kennzahlen der Arbeitsstationen visualisieren. Die Arbeitsstationen müssen in der Lage sein, diese Werte zu aktualisieren. Für die Störungsbehebung müssen Nachrichten zur Unterstützungsanforderung zwischen dem Hancho, der Arbeitsstation und Dritten austauschbar sein. Zur Visualisierung von Statusänderungen an einer Arbeitsstation werden ferner Statusnachrichten ausgetauscht. Falls die Mitarbeiter einer Arbeitsstation eine Störung

Tabelle 5 Anforderungskatalog für die Andon-Methode

FR-ID	Ausgefüllte Satzschablone
28	Falls an der Arbeitsstation eine Störung auftritt, muss die Arbeitsstation dem Werker die Möglichkeit bieten, dem Hancho die Störung zu senden .
29	Falls die Störung nicht innerhalb des Taktes bzw. neben der Fertigung behebbar ist, muss die Arbeitsstation dem Hancho die Möglichkeit bieten, von der Arbeitsstation das rote Stoppsignal an alle Arbeitsstationen der Linie zu senden .
30	Falls der Hancho fertig ist, muss die Arbeitsstation dem Hancho die Möglichkeit bieten, die Fertigstellung mitzuteilen .
31	Falls eine Arbeitsstation das Linienstoppsignal empfängt , muss die Arbeitsstation die Arbeit zum Ende des Taktes anhalten.
32	Falls der Hancho das Eintreffen quittiert , muss die Arbeitsstation den eigenen Status aktualisieren .
33	Falls die Störung behoben ist und der Status wieder "0: i.O." ist, muss die Arbeitsstation allen Arbeitsstationen der Linie das Ende des Linienstopps mitteilen .
34	Falls die Arbeitsstation einen Linienstopp auslöst, muss die Arbeitsstation die Andon-Kennzahlen aktualisieren .
35	Falls der KVP-Verantwortliche die Störungsmeldungen an der Arbeitsstation abfragt, muss die Arbeitsstation fähig sein, die Liste der Störungsmeldungen zurückzugeben .
36	Falls der Status abgefragt wird, muss die Arbeitsstation fähig sein, den Status zurückzugeben .
37	Falls der KVP-Verantwortliche die Soll-Kennzahlen für die Andon-Tafel aktualisiert, muss die Arbeitsstation die Soll-Kennzahlen der Andon-Tafel aktualisieren .
38	Falls die Andon-Kennzahlen einer Arbeitsstation angefragt werden, muss die Arbeitsstation die Andon-Kennzahlen zurückgeben .
39	Falls der Hancho eintrifft, muss die Arbeitsstation dem Hancho die Möglichkeit bieten, das Eintreffen zu quittieren .
40	Falls der Hancho die Störung nicht alleine beheben kann, muss die Arbeitsstation dem Hancho die Möglichkeit bieten, weitere Unterstützung anzufordern .
41	Falls der Hancho weitere Unterstützung anfordert, muss die Arbeitsstation den eigenen Status aktualisieren .
42	Falls der Fehler behoben wurde, muss die Arbeitsstation den eigenen Status aktualisieren .
43	Falls der Werker an der Arbeitsstation eine Störung meldet, muss die Arbeitsstation den eigenen Status aktualisieren .
44	Falls weitere Unterstützung eintrifft, muss die Arbeitsstation der Unterstützung die Möglichkeit bieten, das Eintreffen zu quittieren .
45	Falls die weitere Unterstützung fertig ist, muss der Hancho der Unterstützung die Möglichkeit bieten, die Fertigstellung mitzuteilen .

innerhalb eines Taktes nicht beheben können, muss eine Arbeitsstation mittels der Andon-Reißleine das Linienstopp-Signal an alle anderen Arbeitsstationen senden können.

4.2.3.3 Funktionale Anforderungen der SMED-Methode

Die Anforderungen der SMED-Methode werden ähnlich wie zuvor aus der Modellierung abgeleitet und sind in Tabelle 6 dargestellt. Hieraus ergibt sich, dass eine Arbeitsstation in der Lage sein muss, bei Rüstvorgängen für ein Produkt bzw. eine Kanban die Dauer für die externe Vorbereitung, die interne Durchführung und die externe Nachbereitung zu protokollieren. Auf Anfrage muss die Arbeitsstation dem KVP-Verantwortlichen die gesammelten Rüstvorgangsdauern zur Verfügung stellen können.

4.2.3.1 Zusammenfassung

Die funktionalen Anforderungen beschreiben auf Basis der zuvor erstellten Modelle Anforderungen hinsichtlich auszutauschender Informationen an die Entitätstypen der Lean Production. Die in diesem Kapitel beschriebenen Methoden zeigen den unterschiedlichen Bedarf an Informationen, welcher nicht zwingend mit der Komplexität der Lean-Methode korreliert. So ist der für die verhältnismäßig komplexe Kanban-Methode notwendige Datenaustausch zwischen den Beteiligten mit weniger Anforderungen beschreibbar als der der im ersten Moment als einfach erachteten Andon-Methode.

Ferner lässt sich der Umfang der Referenzarchitektur erkennen. Die 30 Lean-Methoden unterteilen sich in 60 Anwendungsfälle. Aus den Anwendungsfällen wiederum wurden 97 unterschiedliche Anforderungen an den Datenaustausch abgeleitet. Die vollständige Liste der Anforderungen ist in Anhang D aufgeführt. Die funktionalen Anforderungen bilden zusammen mit den im nächsten Kapitel vorgestellten nicht-funktionalen Anforderungen den Anforderungskatalog für die Referenzarchitektur. Aus den verwendeten Textschablonen lässt sich

Tabelle 6 Anforderungskatalog für die SMED-Methode

FR-ID	Ausgefüllte Satzschablone
65	Falls der KVP-Verantwortliche die Rüstzeiten abfragt, muss die Arbeitsstation fähig sein, die Ist-Rüstzeiten an der Arbeitsstation bereitzustellen .
66	Falls die externe Rüstvorbereitung ansteht, muss die Arbeitsstation dem Werker die Möglichkeit bieten, die Dauer der externen Rüstvorbereitung aufzunehmen .
67	Falls die interne Rüstoperation ansteht, muss die Arbeitsstation dem Werker die Möglichkeit bieten, die Dauer der internen Rüstoperation aufzunehmen .
68	Falls die externe Rüstnachbereitung ansteht, muss die Arbeitsstation dem Werker die Möglichkeit bieten, die Dauer der externen Rüstnachbereitung aufzunehmen .

die in Kapitel 0 beschriebene Systemarchitektur und das Informationsmodell ableiten.

4.2.4 Nicht-funktionale Anforderungen des Anforderungskataloges

Allgemein beschreiben nicht-funktionale Anforderungen diejenigen Anforderungen, die über die funktionalen Anforderungen hinausgehen. Sie stellen u.a. Randbedingungen dar. Sie beziehen sich oftmals auf die Gestalt des zu realisierenden Systems und sind nicht immer vollständig quantifizierbar. Durch die Vorgabe von übergreifenden Verhaltensanforderungen wie z.B. Aussagen zur Zuverlässigkeit und Änderbarkeit sowie zu berücksichtigenden Paradigmen haben nicht-funktionale Anforderungen einen hohen Einfluss auf die Systemarchitektur. [Bal11; Poh+11]

In der Praxis hat sich die natürlichsprachige Dokumentation nicht-funktionaler Anforderungen durchgesetzt, welche auch diese Arbeit verwendet [Poh+11]. Quellen für die Erhebung der Anforderungen an die Referenzarchitektur sind die Lean Production, da sich die Referenzarchitektur in diese einbetten muss. Durch den Fokus auf den digitalen Datenaustausch werden Anforderungen aus der Softwareentwicklung berücksichtigt. Speziell der Bereich der lose gekoppelten Systeme bzw. der Objektorientierung wird beachtet, um den Zweck der Referenzarchitektur, die Verbesserung der Wandelbarkeit, zu erreichen. Des Weiteren werden Anforderungen von verteilten Systemen herangezogen, da in der Lean Production mehrere Entitäten miteinander kommunizieren. Zuletzt sind auch nicht-funktionale Anforderungen aus dem Themenfeld „Industrie 4.0“ zu berücksichtigen, da Industrie 4.0 die Technologien zur Realisierung der Referenzarchitektur liefert.

Die Referenzarchitektur muss „lean“ sein, damit sie die Lean-Methoden optimal unterstützt und sich nahtlos in die Lean Production einfügt. Im Sinne der Standardisierung sollte die Referenzarchitektur für alle Rollen möglichst identische informationstechnische Schnittstellen definieren, die für Dritte nachvollziehbar dokumentiert ist. Im Sinne von Poka Yoke und Kaizen dürfen die informationstechnische Schnittstellen ferner fehlerhafte Daten nicht weitergeben und müssen es ermöglichen, aus den Fehlern zu lernen. Im Sinne des Fließ- und Pull-Prinzips und der Verschwendungsvermeidung dürfen die informationstechnische Schnittstellen erst auf Nachfrage aktiv werden, müssen dann aber in der Lage sein, die richtige Information in der richtigen Güte zur richtigen Zeit am richtigen Ort zur Verfügung zu stellen. Im Sinne von 5S bzw. der Verschwendungsvermeidung dürfen die informationstechnischen Schnittstellen nicht mehr Informationen speichern, anbieten und kommunizieren als benötigt. Auch sollte die Referenzarchitektur trotz des technischen Fokus den Menschen berücksichtigen. Im Rahmen von Hoshin Kanri sind die Ziele der einzelnen beteiligten Entitäten auf ein gemeinsames übergeordnetes Ziel abzustimmen. Die Referenzarchitektur muss im Sinne von Kaikaku die Lean-Philosophie verwirklichen. Zuletzt müssen die informationstechnischen Schnittstellen gemäß der TPM-Methode und der Low-Cost-Automation-Methode mit maßvollen, einfachen und möglichst selbst organisierbaren und wartbaren Technologien umgesetzt werden.

Wie in Kapitel 2.1.1 vorgestellt, beschreibt Industrie 4.0 keine konkrete Technologie, sondern neue Paradigmen in der Produktion. Die Referenzarchitektur muss im Sinne der Ad-hoc-Vernetzung die Selbstkonfiguration und eine schnelle Rekonfiguration unterstützen. Im Sinne der durchgängigen digitalen Vernetzung muss die Referenzarchitektur unterschiedlichste Teilnehmer der Produktion, egal ob physisch oder digital bzw. lebendig oder nicht, digital miteinander vernetzen können. Aus der Vision der intelligenten Fabrik der Zukunft leitet sich

die Anforderung ab, dass die Referenzarchitektur die selbstständige Koordination und Entscheidung der involvierten Entitäten unterstützt. Ferner muss die Referenzarchitektur im Sinne der Wandelbarkeit so weit wie möglich erweiterbar und auf neue, aktuell noch nicht bekannte oder berücksichtigte Anforderungen anpassbar sein. Der vermehrte Einsatz von IKT in der Produktion ist ebenfalls ein Paradigma von Industrie 4.0, welches allerdings als Anforderung nicht weiter verfolgt wird, da es der für diese Arbeit gewählte Ansatz ist und somit gegeben ist. Aus dem Bereich der CPSe in der Produktion kommt die Anforderung, dass die Referenzarchitektur in heterogenen Umgebungen mit sich ändernden Systemgrenzen anwendbar sein muss [Wie+14b; Kol+16].

In der Softwareentwicklung ist die Erhebung und Dokumentation von nicht-funktionalen Anforderungen bereits sehr ausgeprägt. Die [Int11] definiert u.a. Kategorien wie Zuverlässigkeit, Benutzbarkeit, Änderbarkeit und Übertragbarkeit für Software-Systeme. Im Sinne der Zuverlässigkeit müssen die informationstechnischen Schnittstellen immer auf eine Anfrage antworten und dürfen nur zuvor definierte, für Dritte einsehbare und somit erwartete Ausgabewerte zurückgeben. Hinsichtlich der Benutzerbarkeit müssen die informationstechnischen Schnittstellen im Fehlerfall eine schnelle Diagnose ermöglichen. Die Anforderungen bezüglich der Änderbarkeit und Übertragbarkeit entsprechen den zuvor bereits zum Thema „Wandelbarkeit und Standardisierung“ beschriebenen Anforderungen. An lose gekoppelte Systeme besteht die Anforderung, dass die Verbindung zwischen einzelnen Systemen nicht fest, sondern flexibel und änderbar ist. Im Sinne der Kapselung bieten Systeme nach außen nur die Dienste an, die Dritte benötigen. Unter dem Paradigma der Komponentenorientierung wird die Zerlegung eines Systems in wiederverwendbare, standardisierbare Elemente verstanden. Im Sinne der Standardisierung soll hierzu auf offene, frei verfügbare Standards zurückgegriffen werden. [Rem+11; Mel10] An verteilte Systeme besteht die nicht-funktionale Anforderung, dass der Schnittstelle ein plattformunabhängiges Informationsmodell zugrunde liegen muss. Dieses Informationsmodell muss ferner für die Implementierung in heterogenen Systemen geeignet sein. Im Sinne der Dienstgütequalität müssen die informationstechnischen Schnittstellen Garantien hinsichtlich Qualität und Zeit abgeben. Anforderungen wie Erweiterbarkeit, Kompositionsaspekte und Verwendung von Standards wurden zuvor genannt. [Maj10]

Zusammenfassend ist für die nicht-funktionalen Anforderungen an die Referenzarchitektur festzuhalten, dass sich diese aus den Bereichen Lean Production, Industrie 4.0 und Softwareentwicklung ableiten. Sie sind weniger konkret wie die funktionalen Anforderungen und stellen eher Gestaltungsrichtlinien für die Referenzarchitektur dar. Tabelle 7 fasst die beschriebenen 30 nicht-funktionalen Anforderungen zusammen.

Tabelle 7 Nicht-funktionale Anforderungen an die Referenzarchitektur

NFR-ID	Beschreibung
NFR01	Im Sinne der Standardisierung sollten die informationstechnischen Schnittstellen für jeden möglichst identisch und transparent sein.
NFR02	Im Sinne des Null-Fehler-Prinzips und Kaizen dürfen falsche Informationen nicht weitergegeben werden und es muss möglich sein, aus ihnen zu lernen.
NFR03	Im Sinne des Fließ- und Pull-Prinzips darf erst auf Nachfrage gehandelt werden; dann müssen aber die richtigen Informationen in der richtigen Güte am richtigen Ort zur Verfügung stehen.
NFR04	Im Sinne der Verschwendungsvermeidung (Muda) dürfen nicht mehr Informationen als nötig gespeichert, angeboten und kommuniziert werden.
NFR05	Im Sinne der Mitarbeiterbeteiligung muss der Mensch berücksichtigt werden.
NFR06	Im Sinne des PDCA-Prozesses sollte die Umsetzung von Handlungen den Schritten Planung, Durchführung, Prüfung und ggf. Korrektur folgen.
NFR07	Im Sinne der Hoshin-Kanri-Methode sollte die Referenzarchitektur durch die Anwendung des Kaskadenprinzips horizontal und vertikal skalierbar sein.
NFR08	Im Sinne der Low-Cost-Automation-Methode sollten die informationstechnischen Schnittstellen durch maßvolle, einfache und selbstorganisierbare Mittel realisiert werden.
NFR09	Im Sinne von Chaku-Chaku sollte ein Werker mehreren Arbeitsstationen zugeordnet werden können.
NFR10	Im Sinne von Chaku-Chaku sollte eine Anlage ohne Quittierung des Werkers nicht weiter arbeiten und eine Schnittstelle für das Quittieren vorsehen.
NFR11	Im Sinne der TPM-Methode sollten die informationstechnischen Schnittstellen durch jeden wartbar sein.
NFR12	Im Sinne der Kaikaku-Methode muss die Referenzarchitektur die Lean-Philosophie unterstützen.
NFR13	Im Sinne der Ad-hoc-Vernetzung muss die schnelle Re- und Selbstkonfiguration unterstützt werden.
NFR14	Im Sinne der vertikalen und horizontalen digitalen Integration müssen unterschiedlichste Teilnehmer miteinander vernetzt werden.
NFR15	Im Sinne der Smart Factory muss die Referenzarchitektur die selbstständige Koordination und Entscheidungsfindung von Entitäten unterstützen.
NFR16	Im Sinne der Wandelbarkeit muss die Referenzarchitektur an zukünftige Anforderungen anpassbar sein.
NFR17	Im Sinne von CPSen in der Produktion muss die Referenzarchitektur sich ändernde Systemgrenzen und Teilnehmer berücksichtigen.
NFR18	Im Sinne von Migrierbarkeit muss die Referenzarchitektur sich in bestehende Umgebungen einbetten können.
NFR19	Im Sinne der Zuverlässigkeit müssen die informationstechnischen Schnittstellen immer und nur erwartete Antworten auf Anfragen liefern.
NFR20	Im Sinne der Benutzbarkeit müssen die informationstechnischen Schnittstellen im Fehlerfall eine schnelle Diagnose ermöglichen.
NFR21	Im Sinne der Änderbarkeit muss die Referenzarchitektur hinsichtlich zukünftiger Anforderungen anpassbar sein.
NFR22	Im Sinne der Übertragbarkeit müssen die informationstechnischen Schnittstellen für jede Rolle möglichst identisch gestaltet sein.
NFR23	Im Sinne lose gekoppelter Systeme müssen die Verbindungen zwischen den informationstechnischen Schnittstellen änderbar und flexibel sein.
NFR24	Im Sinne der Kapselung dürfen die informationstechnischen Schnittstellen nicht mehr als die benötigten Funktionen und Informationen nach außen anbieten.
NFR25	Im Sinne der Komponentenorientierung sollen die informationstechnischen Schnittstellen in wiederverwendbare, standardisierte Elemente zerlegbar sein.
NFR26	Im Sinne der Standardisierung sollte auf offene, frei verfügbare Standards zurückgegriffen werden.
NFR27	Im Sinne dezentraler Systeme sollte der informationstechnischen Schnittstellen ein plattformunabhängiges Informationsmodell zugrunde liegen.
NFR28	Im Sinne dezentraler Systeme sollte die Referenzarchitektur in heterogene Systeme implementierbar sein.
NFR29	Im Sinne der Dienstgütequalität müssen die informationstechnischen Schnittstellen Garantien hinsichtlich Datenqualität und Zeitverhalten abgeben.
NFR30	Im Sinne der Kompositionsaspekte müssen die informationstechnischen Schnittstellen sich aus Komponenten zusammensetzen lassen.

4.3 Bestandteile der Referenzarchitektur

Aus dem vorgestellten Anforderungskatalog kann im nächsten Schritt die Referenzarchitektur abgeleitet werden. Das Kapitel unterteilt sich in die Bestandteile der Referenzarchitektur „Systemarchitektur“ (Kapitel 4.3.1), „Informationsmodell der Schnittstelle“ (Kapitel 4.3.2) und „Schnittstellenarchitektur“ (Kapitel 4.3.3).

4.3.1 Systemarchitektur

Die übergeordnete Systemarchitektur definiert die Grenze des betrachteten Systems und beschreibt dessen Struktur, d.h. sie grenzt die zur Realisierung des zuvor erarbeiteten Anforderungskataloges relevanten Entitätstypen von den für die Referenzarchitektur nicht relevanten Entitätstypen ab. Des Weiteren beschreibt die Systemarchitektur die Aufgaben und Beziehungen zwischen den Entitätstypen. Die Systemarchitektur ist ein Modell der in Kapitel 2.1.4.1 beschriebenen zweiten Abstraktionsebene.

Die Entitätstypen, die in der Systemarchitektur als Rollen bezeichnet werden, sind ein abstraktes Konstrukt. Kennzeichnend für eine Rolle ist, dass sie eine informationstechnische Schnittstelle anbieten muss. Die Rollen ergeben sich aus dem zuvor aufgestellten Anforderungskatalog. Die funktionalen Anforderungen definieren durch die Felder <WER> und <OBJEKT> der verwendeten Satzschablone die jeweils involvierten Rollen (vgl. 4.2.3.1). Für die Instanziierung der Rollen, d.h. deren Umsetzung in der Praxis, definiert die Systemarchitektur ferner relevante Ressourcen. Sie sind vergleichbar mit Bausteinen, aus denen sich die Rollen bilden lassen. Abbildung 57 zeigt die aus dem Anforderungskatalog abgeleitete Systemarchitektur. Insgesamt wurden sechs Rollen identifiziert, die sich durch acht für die Referenzarchitektur relevante Ressourcen realisieren lassen. Im Folgenden werden diese Rollen und Ressourcen vorgestellt.

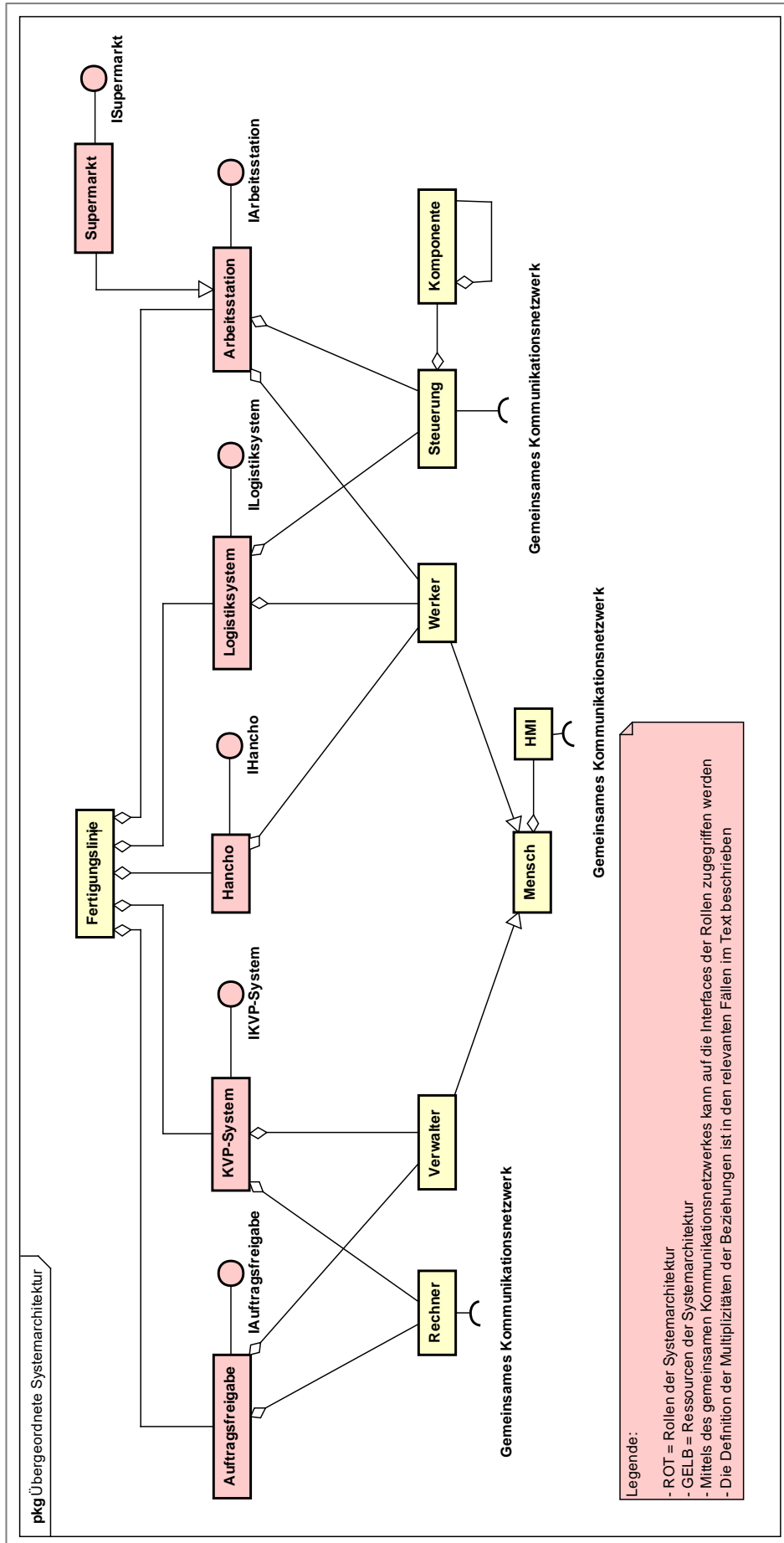


Abbildung 57 Struktur der Systemarchitektur

Ähnlich wie in der Organisationslehre beschreibt eine Rolle ein Erwartungsprofil an den Inhaber der Rolle, welche Aufgaben und Tätigkeiten er ausführen soll. Wie zuvor erwähnt, verfügt jede Rolle über eine informationstechnische Schnittstelle, welche vom gemeinsamen Kommunikationsnetzwerk erreichbar ist. Eine Rolle lässt sich mehr als einmal instanzieren. Für den Betrieb einer Lean-Production-Fertigungslinie ist nachstehende Anzahl an Instanzierungen notwendig.

Die Rolle **Auftragsfreigabe** ist verantwortlich für die Umwandlung von Kundenaufträgen in Fertigungsaufträge. Sie löst somit die Bestellung auf in die richtige Art und Anzahl an Kanban und verteilt diese an die entsprechenden Fertigungslinien. Sie steuert die Fertigungslinie, indem sie im Sinne der Pull-Steuerung der letzten Arbeitsstation eine Kanban zukommen lässt. Die Auftragsfreigabe besteht immer aus einem Rechner und optional aus einem anthropomorphischen Verwalter. Die Auftragsfreigabe hat eine informationstechnische Schnittstelle zur Kommunikation mit Dritten. An einer Fertigungslinie gibt es immer eine Auftragsfreigabe, wobei eine Auftragsfreigabe für mehrere Fertigungslinien zuständig sein kann.

Die Rolle **Arbeitsstation** übernimmt mindestens einen Arbeitsschritt der Fertigung und trägt somit unmittelbar zur Wertschöpfung bei. Eine Arbeitsstation ist eine logische Zusammenfassung eines abgeschlossenen Fertigungsschrittes bzw. -bereiches, der sich in der Praxis in der Regel durch eine eigene Kostenstelle oder Identifikationsnummer auszeichnet. Kanban-Regelkreise verbinden die Arbeitsstationen einer Fertigungslinie miteinander. Die Arbeitsstation am Ende der Fertigungslinie, die das Produkt finalisiert, kommuniziert als einzige mit der zuvor beschriebenen Auftragsfreigabe-Rolle. Die Arbeitsstation besteht immer aus mindestens einer via Steuerung kontrollierten Komponente und verfügt ebenfalls über eine informationstechnische Schnittstelle. Optional können ein oder mehrere Werker zugeordnet sein. Eine besondere Form der Arbeitsstation ist der **Supermarkt**. Der Supermarkt ist nicht wertschöpfend, sondern ein Pufferlager zwischen Fertigungslinien und kritischen Produktionsabschnitten. Pufferlager entkoppeln einzelne Produktionsbereiche voneinander und gleichen unterschiedliche Taktzeiten an. Eine Fertigungslinie verfügt immer über mindestens eine Arbeitsstation und optional über einen oder mehrere Supermärkte.

Die Rolle **Logistiksystem** ist verantwortlich für den Transport von Waren zwischen den Arbeitsstationen. Hierbei kann es sich um das zu fertigende Produkt handeln, welches zwischen den Arbeitsstationen der Fertigungslinie transportiert wird, um Vorprodukte, um andere Rohmaterialien zur Versorgung der Arbeitsstationen, um leere Werkstückträger, die an ihre ursprüngliche Position zurück sollen, oder Kanban. Ähnlich wie die Arbeitsstation besteht das Logistiksystem immer aus einer oder mehreren mittels Steuerung steuerbaren Komponenten und ggf. aus einem oder mehreren Werkern. Diese Steuerung bietet für das gemeinsame Kommunikationsnetzwerk die informationstechnische Schnittstelle an. Das Logistiksystem ist mobil. In der Regel befindet sich in einer Fertigungslinie ein Logistiksystem für den zyklisch stattfindenden Milkrun. Es sind aber auch mehrere Logistiksysteme möglich.

Die Rolle **Hancho** ist im Gegensatz zur Arbeitsstation und zum Logistiksystem keine fest im Wertschöpfungsprozess involvierte Rolle, sondern übernimmt alle Tätigkeiten der Planung und Überwachung eines definierten Bereiches wie z.B. einer Fertigungslinie. Sie ist vergleichbar mit einem Teamleiter für einen bestimmten Fertigungsabschnitt. Der Hancho eilt als Springer außerdem bei Problemen an Arbeitsstationen zu Hilfe. Einer oder mehrere Werker nehmen diese Rolle ein. Zur Kommunikation verwendet die Rolle ebenfalls die informati-

onstechnische Schnittstelle. In einer Fertigungslinie gibt es je nach Anzahl der Mitarbeiter einen oder mehrere Hanchos. Da die Aufgaben des Hanchos in der Lean Production nicht einheitlich definiert sind, kann ein Hancho entweder Teamleiter oder Springer oder beides sein.

Die sechste Rolle, das **KVP-System**, ist verantwortlich für alle Aktivitäten zur kontinuierlichen Verbesserung. Sie analysiert die Fertigung, nimmt Fehler auf, leitet Verbesserungspotenziale ab, meldet diese dem Verantwortlichen zur Realisierung der Potenziale und überwacht die Umsetzung. Ein KVP-System wird immer mithilfe eines Rechners umgesetzt und von einem oder mehreren Verwaltern benutzt. Ferner bietet auch diese Rolle eine informationstechnische Schnittstelle an. Eine Fertigungslinie hat üblicherweise ein KVP-System, ein KVP-System kann umgekehrt aber für mehrere Fertigungslinien zuständig sein.

Die nachstehend beschriebenen Ressourcen repräsentieren in der Realität notwendige bzw. vorhandene Komponenten, um die abstrakten Rollen umzusetzen. Die Ressourcen sind somit die für die Realisierung der informationstechnischen Schnittstellen notwendigen Bausteine.

Zu den Ressourcen gehört die Ressource **Rechner**, welche automatisiert Berechnungen durchführt und Daten speichert. Sie ist über eine Netzwerkschnittstelle an ein gemeinsames Kommunikationsnetzwerk angebunden. Rechner können z.B. PCs oder Server sein.

Die Ressource **Steuerung** führt ebenfalls automatisiert Berechnungen durch und verfügt über eine gemeinsame Netzwerkschnittstelle zur Kommunikation. Im Gegensatz zum Rechner ist sie in der Lage, Aktoren, Sensoren und andere Komponenten zu steuern bzw. zu regeln. Steuerungen können z.B. klassische SPSen oder CPSe sein.

Die Ressource **Komponente** ist ein Sammelbegriff für physische Systeme, die für die Referenzarchitektur eine untergeordnete Rolle spielen. Komponenten sind nicht zwingend an das gemeinsame Kommunikationsnetzwerk angebunden. Sie interagieren mit Steuerungen, die wiederum eine informationstechnische Schnittstelle für die Lean-Methoden anbieten. Konkret können dies alle physischen Gegenstände der Produktion sein wie z.B. Feldgeräte, andere Steuerungen oder Werkzeuge.

Die Mensch-Maschine-Schnittstelle (engl. Human Machine Interface; **HMI**) repräsentiert als Ressource alle Schnittstellen zwischen dem Menschen und der digitalen Welt. Sie bietet dem Menschen die Möglichkeit, das gemeinsame Kommunikationsnetzwerk zu nutzen. Hierzu gehören z.B. Tablets oder Datenbrillen.

Die Ressource **Mensch** unterteilt sich in zwei Gruppen und verfügt im Kontext dieser Arbeit immer über ein HMI. Zum einen kann dies ein **Werker** sein, der operativ in der Produktion an der Wertschöpfung beteiligt ist. Hierzu zählen z.B. Mitarbeiter an einer Arbeitsstation, welche ein Produkt fertigen. Zum anderen kann dies ein **Verwalter** sein, welcher planende und verwaltende Aufgaben primär im administrativen Bereich wahrnimmt.

Die **Fertigungslinie** fasst alle Ressourcen und Rollen zusammen und ist die sechste Ressource. Eine Fertigungslinie ist dadurch gekennzeichnet, dass sie eine Aneinanderreihung von Arbeitsstationen sowie notwendige Betriebsmittel zur Fertigung eines Produktes umfasst. Eine Produktion kann dabei aus mehreren Fertigungslinien bestehen. Die Rollen und Ressourcen können mehreren Fertigungslinien zugeordnet sein.

Die aus dem Anforderungskatalog abgeleitete Systemarchitektur mit ihren Rollen und Ressourcen zeigt, dass nicht alle in der Produktion vorhandenen Objekte für die Referenzarchitektur relevant sind. U.a. sind die üblicherweise von ERP-Systemen wahrgenommenen Aufgaben wie die Produktionsprogrammplanung und Personaleinsatzplanung außerhalb des Betrachtungsbereiches. Auch das Produkt selbst spielt für die Referenzarchitektur eine untergeordnete Rolle und bietet keine informationstechnische Schnittstelle an, da es in der Lean Production nicht mit anderen Systemen kommuniziert. Offensichtlich ist auch, dass alle Objekte und Prozesse außerhalb der Produktion wie z.B. die IT-Systeme der Buchhaltung oder die der Lieferanten und Kunden nicht für die weitere Betrachtung benötigt werden.

Zusammenfassend konnten aus den modellierten Lean-Methoden und dem aufgestellten Anforderungskatalog sechs Rollen identifiziert werden. Die Auftragsfreigabe, das KVP-System, der Hancho, das Logistiksystem, die Arbeitsstation und der Supermarkt sind für die Digitalisierung der Lean-Methoden notwendig und müssen eine informationstechnische Schnittstelle anbieten. Die von ihnen angebotene Schnittstelle wird im Folgenden durch das Informationsmodell vertieft. Jede Rolle und ihre Schnittstelle lässt sich mittels der beschriebenen Ressourcen in der Praxis realisieren.

Abbildung 58 beschreibt exemplarisch, wie sich eine Fertigungslinie mithilfe der Ressourcen und Rollen abstrahieren lässt, um die notwendigen informationstechnischen Schnittstellen zu identifizieren. In der Fertigungslinie vorhandene Werkzeuge und Maschinen werden als Komponenten abstrahiert. Ein dort vorhandenes IT-System wie ein BDE-Terminal oder eine Anlagensteuerung realisiert die informationstechnische Schnittstelle und ist an das gemeinsame Kommunikationsnetzwerk angebunden. Andere ggf. vorhandene Steuerungen etc. sind dem untergeordnet und müssen nicht außerhalb der Arbeitsstation kommunizieren. Falls keine Geräte für die Realisierung der informationstechnischen Schnittstelle vorhanden sind, wie beispielsweise in einem Regal, können z.B. nachgerüstete CPSe diese Funktion übernehmen. Der Hancho verfügt in dem Beispiel über eine Smart Watch mit einer informationstechnischen Schnittstelle zur Kommunikation mit allen anderen. Das Flurförderfahrzeug des Milkruns tauscht drahtlos Daten aus. Wie zuvor beschrieben, sind Fertigungslinien durch Supermärkte als Pufferlager voneinander entkoppelt, und nur die letzte Arbeitsstation (Nr. 13) der links in der Abbildung senkrecht dargestellten Haupt-Fertigungslinie empfängt Aufträge des ERP-Systems.

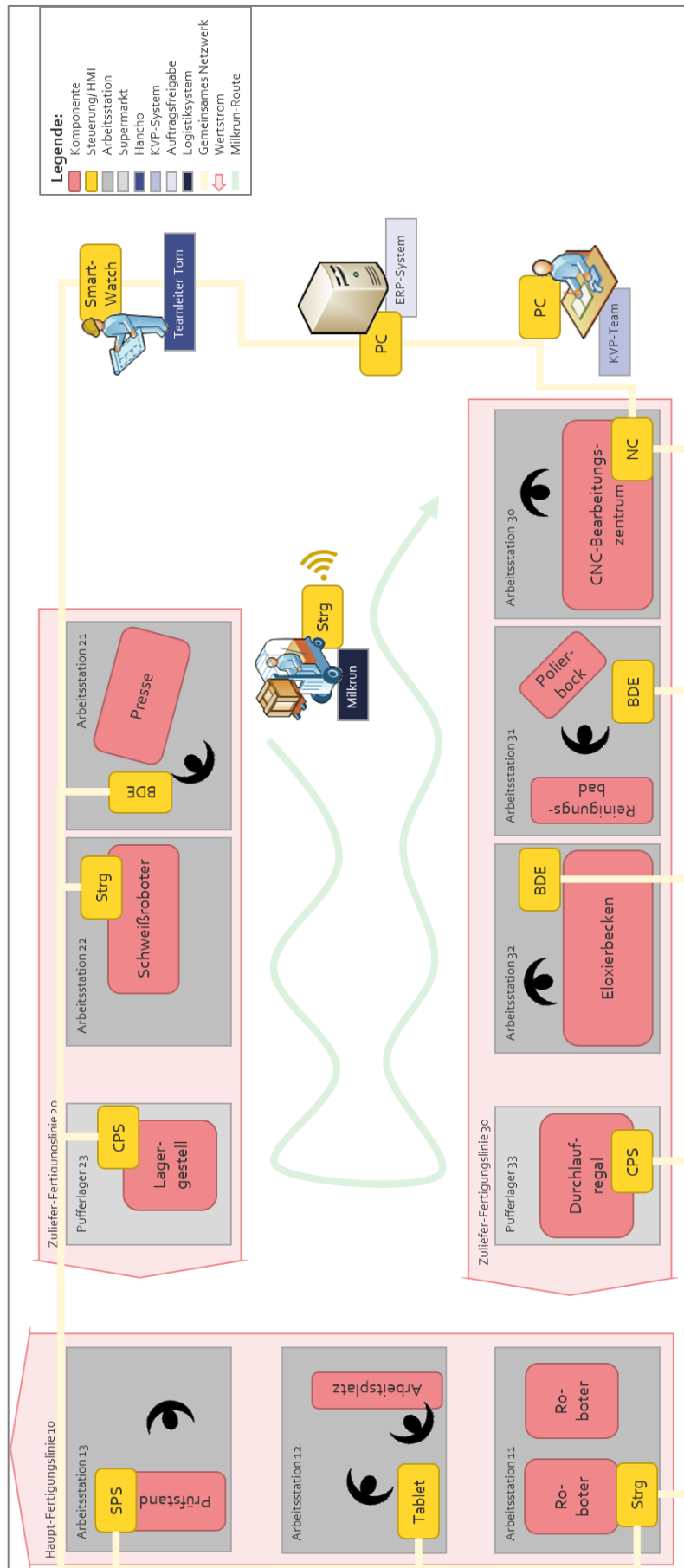


Abbildung 58 Beispiel für die Abstraktion einer Produktion mit mehreren Fertigungslinien mittels der Systemarchitektur (Draufsicht)

4.3.2 Informationsmodell der Schnittstelle

4.3.2.1 Vorgehens- und Darstellungsweise des Informationsmodells

Für die zuvor identifizierten Rollen der Lean Production kann zusammen mit den funktionalen Anforderungen das Informationsmodell der informationstechnischen Schnittstellen erstellt werden. Dies umfasst die Definition der von jeder Rolle anzubietenden Dienste, die Definition von ausgetauschten Nachrichtentypen und deren Inhalte. Im Gegensatz zur Systemarchitektur handelt es sich nicht um die Beschreibung aus Struktursicht, sondern aus Kommunikationssicht.

Zur Formalisierung der identifizierten Nachrichtenflüsse und der involvierten Akteure werden UML Kommunikationsdiagramme eingesetzt. Sie reduzieren die einzelnen Anwendungsfälle auf den Nachrichtenaustausch und Dienstaufwurf zwischen Sendern und Empfängern. Im Gegensatz zu den weitverbreiteten UML Sequenzdiagrammen verzichten UML Kommunikationsdiagramme auf die Darstellung des zeitlichen Ablaufs des Nachrichtenflusses. [Poh+11] Ein Teilergebnis der Arbeit ist die Beschreibung der notwendigen Dienste, die die jeweilige informationstechnische Schnittstelle anbieten muss. Der zeitliche Ablauf ist daher hier irrelevant, weshalb auch die in UML Kommunikationsdiagrammen verwendeten Sequenznummern keine Bedeutung haben.

Die mindestens zur Definition der je Rolle anzubietenden Dienste ergeben sich aus den unterschiedlichen Kommunikationsmethoden (vgl. Kapitel 2.1.3.2). Bei der Request-Reply-Methode wird eine Funktion an einer Entität aufgerufen. Hierfür muss bekannt sein, welche Entität die Funktionalität anbietet, welche Parameter sie übergeben bekommt und welcher Rückgabewert zu erwarten ist. Für die Implementierung der Callback-Methode müssen der Sender einer Nachricht, die Nachricht selbst sowie der Abonnent bekannt sein. Die ereignisbasierte Methode hat den geringsten Spezifikationsbedarf. Für sie muss lediglich der Sender eines Ereignisses sowie die gesendete Nachricht definiert werden. Ferner ist es für den Integrator, der eine Methode der Lean Production digital umsetzt, hilfreich, die zugehörige Lean-Methode und optional eine Beschreibung des Dienstes und der Nutzung zu kennen. Für eine von der Art der Implementierung unabhängige Definition der Dienste muss somit mindestens bekannt sein:

- Die Rolle, die den Dienst anbietet
- Der Name des Dienstes
- Die zu übergebenden Parameter
- Der erwartete Rückgabewert (falls vorhanden)
- Die zugehörige Lean-Methode, die dadurch unterstützt wird
- Optional eine Beschreibung des Dienstes bzw. der Nutzung

Die Dokumentation erfolgt anhand einer Tabelle, welche die zuvor genannten Attribute je Dienst auflistet. Alternative, in der Praxis verbreitete Ansätze wie die in der Wirtschaftsinformatik verbreiteten Service Level Agreements (SLAs; siehe z.B. [Hil02]), die menschenlesbar von einem Dienstleister zugesicherte Leistungen dokumentieren, sind an dieser Stelle zu detailliert. Auch die zur maschinenlesbaren Interpretation von Webservices bekannten XML-basierten WSDL (siehe z.B. [Chi+07]) würden hier die Freiheit für die technische Implementierung einschränken.

Die Beschreibung der Nachrichtentypen hingegen umfasst andere Informationen. Eine Nachricht ist ein Datenpaket, das die bei einem Dienstaufwurf übergebenen oder die von dieser zurückgegebenen Informationen zusammenfasst. Nachrichten sowie deren Attribute lassen sich mithilfe von UML Klassendiagrammen beschreiben, mit welchen auch Abhängigkeiten wie Generalisierungen beschrieben werden können. Neben einer Bezeichnung des Nachrichtentypes sind zur Beschreibung die darin enthaltenen Attribute und deren möglichen Ausprägungen relevant. Die Vorgabe von Datentypen für die Attribute ist für eine implementierungsunabhängige Beschreibung nur bedingt möglich, da unterschiedliche Protokolle und Programmiersprachen unterschiedliche Datentypen verwenden können. Sie lassen sich somit nur im Rahmen eines Beispiels angeben. Für die Dokumentation der Nachrichtentypen ergeben sich folgende Pflichtinhalte:

- Der Name des Nachrichtentypes
- Die darin enthaltenen Pflichtattribute
- Die Inhalte bzw. die mögliche Ausprägung der Attribute
- Optional eine Beschreibung, ein Beispiel für die Nutzung und mögliche Datentypen

4.3.2.2 Anzubietende Dienste der Rollen

Die anzubietenden Dienste der Rollen lassen sich in drei Typen unterteilen. Der erste Typ an Diensten liefert auf Anfrage Werte zurück (Präfix „liefere...“), ohne Veränderungen bei der Rolle durchzuführen. Der zweite Typ empfängt Daten von Dritten zur Weiterverarbeitung (Präfix „empfangen...“) und der dritte Typ aktualisiert bestehende Daten wie z.B. Stammdaten in einer Rolle (Präfix „aktualisiere...“). Nachstehend ist eine Beschreibung der bereits in Kapitel 4.2.2 modellierten Lean-Methoden inklusive ihrer Kommunikationsdiagramme aufgeführt. Eine Auswahl weiterer Kommunikationsdiagramme befindet sich in Anhang C.

Zur Realisierung des zuvor vorgestellten Modells der Kanban-Methode muss die informationstechnische Schnittstelle der **Auftragsfreigabe** den Dienst „empfangenStatusAuftragsKanban“ anbieten. Wie im Kommunikationsdiagramm in Abbildung 59 dargestellt, kann sie mit

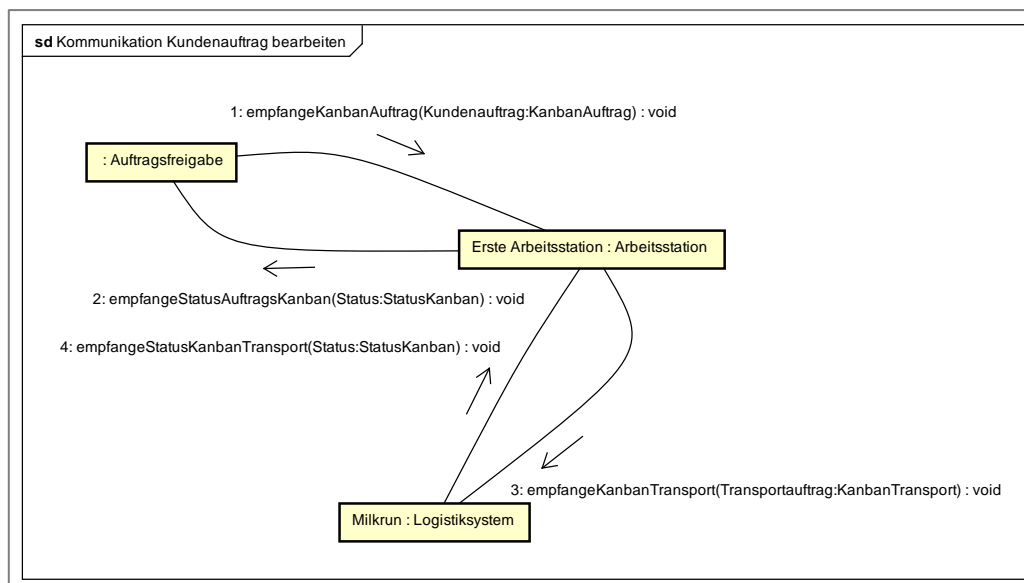


Abbildung 59 Kommunikationsdiagramm für den Anwendungsfall „Kundenauftrag bearbeiten“ der Kanban-Methode

diesem Dienst Statusaktualisierungen zu einem zuvor an die letzte Arbeitsstation einer Fertigungslinie gesendeten Kundenauftrag erhalten und diesen beispielsweise nach der Bearbeitung als erledigt markieren. Mit dem Dienst „aktualisiereKanban“ kann die Auftragsfreigabe geänderte Auftrags-Kanban empfangen. Aus den Modellen der zuvor vorgestellten Andon- und SMED-Methode ergeben sich keine notwendigen Dienste, welche die Auftragsfreigabe anbieten muss. Zur Realisierung von JIT/JIS und Produktionsnivellierung bietet die Auftragsfreigabe den Dienst „liefereAuftragsliste“ an, der für einen vorgegebenen Zeitraum eine Liste der Aufträge liefert. Dieser Dienst wird beispielsweise vom KVP-System verwendet, um im Rahmen der Produktionsnivellierung die Taktung und die Losgrößen der Fertigungslinie auf die Kundenbedarfe auszurichten. Für die Wertstromplanung bietet die Auftragsfreigabe den Dienst „liefereKundenbedarfe“ und „liefereAuftragswarteschlange“, welche für eine vorgegebene Produktgruppe die benötigten Informationen zurückgibt. Tabelle 8 liefert einen Überblick über die für Dritte im gemeinsamen Kommunikationsnetzwerk anzubietenden Dienste.

Abbildung 60 und Tabelle 9 zeigen die von der informationstechnischen Schnittstelle des **KVP-Systems** anzubietenden Dienste. Aus der zuvor beschriebenen Kanban-, SMED- und Andon-Methode ergeben sich keine Dienste, die das KVP-System realisieren muss. Für die 8D-Report-Methode muss das KVP-System fähig sein, einen 8D-Report zu empfangen, um Aktivitäten wie die Produkt- und Prozess-FMEA und die Erstellung eines Kontrollplans zu initiieren. Poka Yoke benötigt vom KVP-System den Dienst „empfangenFehleranalyse“ und „empfangenPokaYokeUmsetzung“, um Fehleranalysen zu interpretieren und die Bearbeitung zuvor ausgelöster Poka-Yoke-Aktivitäten zu dokumentieren. Mit dem Dienst „liefereLinienSixSigmaKennzahlen“ gibt das KVP-System die für eine vorgegebene Fertigungslinie

Tabelle 8 Von der informationstechnischen Schnittstelle der Auftragsfreigabe anzubietende Dienste

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
Kanban	»empfangenStatusAuftragsKanban«	Status : StatusKanban	-	Empfängt eine übergebene Statusaktualisierung zu einer von der Auftragsfreigabe zuvor versendeten Auftragskanban. Hierdurch kann die Auftragsfreigabe über Verzögerungen und die Fertigstellung des Auftrags informiert werden.
Heijunka; JIT/ JIS	»liefereAuftragsliste«	Startdatum : Date, Enddatum : Date, Antwortadresse : iSID	ListeAufträge	Liefert eine Liste aller Kundenaufträge in einem vorgegebenen Zeitraum. Hierdurch kann das KVP-System die monatlichen Kundenbedarfe bestimmen, um eine Nivellierung durchzuführen und die Auftragsystematik für JiT/ JiS umzustellen.
Wertstromplanung	»liefereAuftragswarteschlange«	Produktgruppe : WertstromanalyseAuftragswarteschlange, Antwortadresse : iSID	WertstromanalyseAuftragswarteschlange	Liefert die für eine übergebene Produktgruppe aktuelle Auftragswarteschlange (Queue), d.h. die Kundenaufträge, die bereits bestellt sind, sich aber noch nicht in der Fertigung befinden.
Wertstromplanung	»liefereKundenbedarfe«	Produktgruppe : WertstromanalyseProduktgruppe, Antwortadresse : iSID	WertstromanalyseKundenbedarfe	Liefert die Kundenbedarfe für eine übergebene Produktgruppe. Auf Basis dieser Information kann z.B. das KVP eine Wertstromplanung durchführen.
Kanban	»aktualisiereKanban«	geänderte-Kanban: KanbanAuftrag	-	Aktualisiert oder löscht eine bestehende Auftrags-Kanban bei der Auftragsfreigabe.

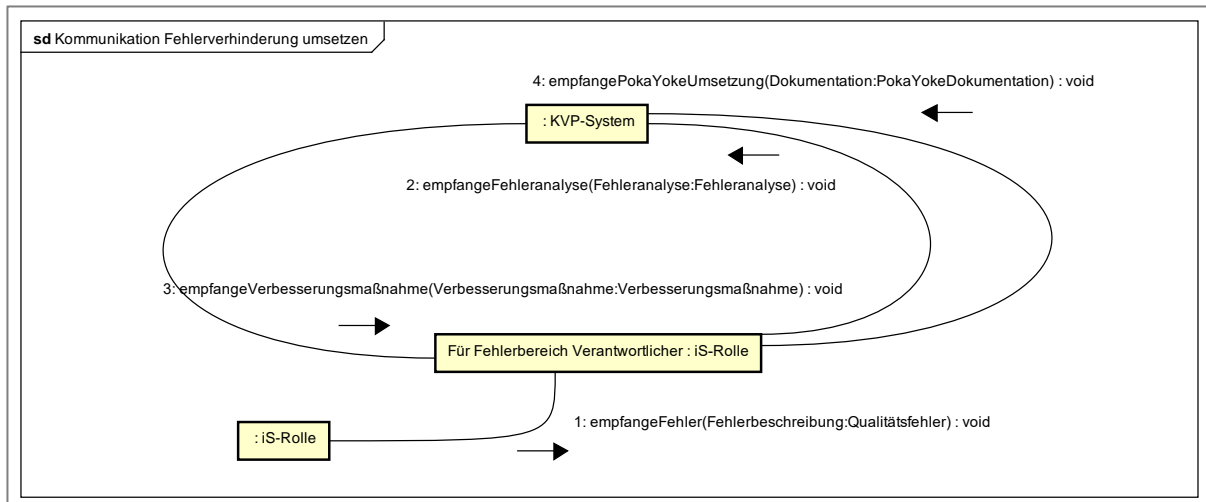


Abbildung 60 Kommunikationsdiagramm für den Anwendungsfall „Fehlerverhinderung umsetzen“ der Poka-Yoke-Methode

definierten Soll-Kennzahlen zurück. Für das Ideenmanagement bietet es den Dienst „empfangenVerbesserungsvorschlag“, mit der es Verbesserungsvorschläge entgegennimmt. „liefereAuditbericht“ gibt zu einer vorgegebenen Audit-Nummer den Bericht zurück. Da das KVP-System 5S-Maßnahmen initiiert, muss es mittels des „empfangenStatusKanban5S“-Dienstes fähig sein, den Status einer 5S-Maßnahme zu empfangen. Bis auf den Dienst für den 8D-Report, der nur von der Arbeitsstation genutzt wird, werden alle Dienste des KVP-Systems von allen Rollen in der Lean Production genutzt.

Das **Logistiksystem** bietet für die Kanban-Methode den Dienst „empfangenKanbanTransport“ an. Dritte können durch diesen Dienst dem Logistiksystem eine notwendige Materialbewegung zur Disposition mitteilen. Wie zuvor erwähnt, kann für die Referenzarchitektur ein

Tabelle 9 Von der informationstechnischen Schnittstelle des KVP-Systems anzubietende Dienste

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
8D-Report	»empfangenFehlerwiederholungsaktivität«	aktueller8D-Report : 8D-Report	-	Initiiert am KVP-System den Prozess, um eine Produkt- und Prozess-FMEA durchzuführen sowie einen Kontrollplan und -prozess zu definieren (Schritt 7 des 8D-Reports in VDI 2870-2).
Poka Yoke; Ishikawa-Diagramm	»empfangenFehleranalyse«	Fehleranalyse: Fehleranalyse	-	Empfängt die Ergebnisse einer FMEA oder eines Ishikawa-Diagramms zur Analyse und zum Ableiten von Maßnahmen.
Poka Yoke	»empfangenPokaYokeUmsetzung«	Dokumentation: PokaYoke-Dokumentation	-	Dokumentiert die erfolgreich eingeführte Poka-Yoke-Maßnahme.
Six Sigma	»liefereLinienSixSigmaKennzahlen«	Fertigungslinie: String; Antwortadresse: iSID	KennzahlenSixSigma	Liefert die für Six Sigma relevanten Kennzahlen einer Fertigungslinie.
Ideenmanagement	»empfangenVerbesserungsvorschlag«	Idee: Verbesserungsvorschlag	-	Empfängt einen Verbesserungsvorschlag zur weiteren Nachverfolgung (Bewertung und ggf. Umsetzung).
Audit	»liefereAuditbericht«	AuditID: int; Antwortadresse: iSID	Auditbericht	Liefert einen spezifischen Auditbericht zu einer übergebenen Audit-ID.
5S	»empfangenStatusKanban5S«	Status: StatusKanban	-	Empfängt eine Statusaktualisierung zu einer zuvor versendeten 5S-Kanban.

Black-Box-Ansatz verwendet werden, d.h. wie das Logistiksystem diesen Transportauftrag verarbeitet, ist für das Informationsmodell irrelevant. Gemäß der Kanban-Methode wird das Logistiksystem im nächsten Milkrun den Transportauftrag erfüllen. Denkbar wäre auch, dass es früher fährt oder den Transportauftrag wiederum an einen Dritten zur Erfüllung weitervermittelt. Für die Wertstromplanung ermöglicht das Logistiksystem dem KVP-System, die hierfür relevanten Materialflussdaten mit dem Dienst „liefereMaterialflussdaten“ für eine vorgegebene Produktgruppe abzurufen. Im Rahmen der Taktung-Methode kann das Logistiksystem mittels „aktualisiereTaktzeit“ neue Soll-Taktzeiten empfangen oder mittels „liefereTaktzeitListe“ die Liste der aktuellen Taktzeiten je Produkt zurückgeben. Eine Übersicht der Dienste ist in Tabelle 10 dargestellt.

Mit 27 Diensten bietet die **Arbeitsstation** die meisten Dienste an, was u.a. am Betrachtungsfokus durch die Lean-Methoden liegt. Für die Kanban-Methode muss eine Arbeitsstation in der Lage sein, Kundenaufträge von der Auftragsfreigabe als auch von der Fertigungskanban anderer Arbeitsstationen zu empfangen. Ferner muss es zu verschickten Fertigungs- und Transport-Kanban Statusaktualisierungen entgegennehmen können, um z.B. bei Verzug oder Fertigstellung zu reagieren. Dem KVP-System kann eine Arbeitsstation mit dem Dienst „liefereEmpfangbareKanbanListe“ und „liefereVerschickbareKanbanListe“ eine Liste sowohl der empfangbaren als auch der verschickbaren Kanban zur Verfügung stellen. Wenn sich im Rahmen der Lean-Methoden Produktionsnivellierung oder Taktung die Informationen auf Kanban ändern, neue Kanban erstellt werden oder alte obsolet sind, kann das KVP-System mit dem Dienst „aktualisiereKanban“ die geänderte Kanban der Arbeitsstation übermitteln. Für die Andon-Methode bietet die Arbeitsstation den Dienst „aktualisiereSollKennzahlenAndon“ und „liefereAndonKennzahlen“ zum Auslesen und Aktualisieren der Ist- und Soll-Kennzahlen. Außerdem können Dritte mittels „liefereStörungsliste“ eine Liste der protokollierten Fehler auslesen, um Maßnahmen zur kontinuierlichen Verbesserung zu planen. Der Dienst „liefereStatus“ gibt den Betriebsmodus einer Arbeitsstation gemäß des in der [Ver13] beschriebenen 5-Stufen-Modells für die Andon-Methode zurück. Neben diesen

Tabelle 10 Von der informationstechnischen Schnittstelle des Logistiksystems anzubietende Dienste

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
Kanban; Milkrun	»empfangenKanbanTransport«	Transportauftrag: Kanban-Transport	-	Empfängt und verarbeitet eine Kanban mit einem Transportauftrag. Hierdurch wird dem Logistiksystem mitgeteilt, dass eine Materialbewegung z.B. im nächsten Milkrun notwendig ist.
Wertstromplanung	»liefereMaterialflussdaten«	Produktgruppe: WertstromanalyseProduktgruppe; Antwortadresse: iSID	WertstromanalyseMaterialflussdaten	Liefert Materialflussinformationen für eine übergebene Produktgruppe. Auf Basis dieser Information kann z.B. das KVP-System eine Wertstromplanung durchführen.
Taktung	»aktualisiereTaktzeit«	AktualisierteTaktzeit:Taktzeit	-	Aktualisiert die allgemeine bzw. produktbezogene Taktzeit eines Logistiksystems.
Taktung	»liefereTaktzeitListe«	Antwortadresse: iSID	ListeTaktzeiten	Liefert eine Liste der vergangenen, bearbeiteten Produkte und deren Soll- und Ist-Zeiten des Logistiksystems.
Kanban	»aktualisiereKanban«	geänderteKanban: Kanban-Transport	-	Aktualisiert oder löscht eine bestehende Transport-Kanban.

Diensten bietet die Arbeitsstation weitere Dienste u.a. für die Erstellung und Bearbeitung eines 8D-Reports, die Erstellung einer Bottleneck- und Wertstromanalyse und die Anzeige relevanter Informationen für das Shopfloor-Management an. Nachstehende Tabelle 11 zeigt einen Ausschnitt der Dienste für die Methoden Kanban, Andon und SMED. Die vollständige Liste ist im Anhang E zu finden.

Tabelle 11 Von der informationstechnischen Schnittstelle der Arbeitsstation anzubietende Dienste

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
Kanban	»aktualisiereKanban«	geänderte-Kanban: KanbanFertigung	-	Aktualisiert eine bestehende Fertigungs-Kanban anhand der empfangenen Fertigungs-Kanban. Hierdurch kann u.a. das KVP-System nach einer Änderung der Taktzeiten und Losgrößen neue Kanban verteilen.
Andon	»aktualisiereSollKennzahlenAndon«	geänderte-SollKennzahlen: KennzahlenAndonSoll	-	Aktualisiert die Soll-Kennzahlen von Andon anhand der übergebenen, neuen Kennzahlen. Hierdurch kann das KVP-System nach u.a. einer Nivellierung der Produktion die Vorgabewerte in einer Arbeitsstation aktualisieren.
Kanban	»empfangenKanbanAuftrag«	Kundenauftrag: KanbanAuftrag	-	Ermöglicht das Übermitteln eines Kundenauftrags an eine Arbeitsstation, welche anschließend die Produktion aufnimmt. Wird von der ersten Arbeitsstation einer Fertigungslinie verwendet, um von der Auftragsfreigabe Aufträge zu erhalten.
Kanban; Supermarkt	»empfangenKanbanFertigung«	Fertigungs-Kanban: KanbanFertigung	-	Empfängt eine Kanban mit einem Fertigungsauftrag von einer anderen (vorgelagerten) Arbeitsstation. Hierdurch wird der Arbeitsstation mitgeteilt, dass sie mit der Produktion des auf der Kanban angegebenen Produktes beginnen soll.
Kanban	»empfangenStatusKanbanFertigung«	Status: StatusKanban	-	Empfängt eine übergebene Statusaktualisierung zu einer von der Arbeitsstation zuvor versendeten Fertigungskanban. Hierdurch kann die Arbeitsstation über Verzögerungen und die Fertigstellung der Fertigungs-Kanban informiert werden.
Kanban; Milkrun	»empfangenStatusKanbanTransport«	Status: StatusKanban	-	Empfängt eine übergebene Statusaktualisierung zu einem von der Arbeitsstation zuvor versendeten Transportauftrag. Hierdurch kann die Arbeitsstation über Verzögerungen und die Fertigstellung des Transports informiert werden.
Andon	»liefereAndonKennzahlen«	Antwortadresse: iSID	KennzahlenAndon	Liefert die Soll- und Ist-Kennzahlen für Andon. Wird u.a. für die Anzeige auf der Andon-Tafel verwendet.
Kanban	»liefereEmpfangbareKanbanListe«	Antwortadresse: iSID	ListeKanban	Liefert dem Anfragenden eine Liste aller Kanban, die die Arbeitsstation empfangen und verarbeiten kann. Verarbeiten bedeutet hierbei, dass sie die auf der Kanban angegebenen Produkte selbst herstellen kann. Wird u.a. von anderen Teilnehmern und dem KVP verwendet, um herauszufinden, was eine Arbeitsstation leisten kann.
SMED	»liefereRüstvorgängeListe«	Antwortadresse: iSID	ListeRüstvorgänge	Liefert eine Liste aller Rüstvorgänge, die an einer Arbeitsstation in der Vergangenheit durchgeführt wurden (inkl. deren Dauer). Wird vom KVP-System für Verbesserungen verwendet.
Andon	»liefereStatus«	Antwortadresse: iSID	StatusArbeitsstation	Liefert den Status einer Arbeitsstation gemäß der Andon-Abstufung (z.B. 5-Stufen-Modell in VDI 2870-2). Die Information wird für die Andon-Tafel sowie die Visualisierung des Arbeitsstations-Zustands verwendet.
Andon	»liefereStörungsliste«	Antwortadresse: iSID	ListeStörungen	Liefert eine Liste aller aufgetretenen Störungen an einer Arbeitsstation. Diese kann vom KVP-System anschließend hinsichtlich KVP-Maßnahmen ausgewertet werden oder der Anzeige auf einer Shopfloor-Tafel dienen.
Kanban	»liefereVerschickbareKanbanListe«	Antwortadresse: iSID	ListeKanban	Liefert dem Anfragenden eine Liste aller Kanban, die die Arbeitsstation versenden und deren Produkte sie als Vorprodukte verarbeiten kann. Wird u.a. von anderen Teilnehmern und dem KVP verwendet, um herauszufinden, was eine Arbeitsstation leisten kann.

Die informationstechnische Schnittstelle der **Supermarkt-Rolle** erbt als Spezialisierung der Arbeitsstation alle Eigenschaften dieser und hat darüber hinaus zwei weitere Dienste. Da der Supermarkt im Gegensatz zu einer Arbeitsstation keine wertschöpfende Tätigkeit durchführt, sondern ein Lager ist, bietet der Supermarkt mit dem Dienst „liefereLagerdaten“ eine Übersicht der verfügbaren Lagerorte, ihrer Kapazitäten und zugeordneten Produkte. Darüber hinaus bietet er auch einen Dienst, der die Stammdaten des Lagers aktualisiert. Tabelle 12 stellt die angebotenen Dienste dar.

Neben den rollenspezifischen Diensten existieren auch Dienste, die jede Rolle anbieten muss. Diese an alle vererbten Eigenschaften sind in der abstrakten **iS-Rolle** definiert. Hierzu gehört beispielsweise aus der Andon-Methode die Möglichkeit, Unterstützung anzufordern und eine Statusaktualisierung auf eine Unterstützungsanfrage zu empfangen. Auch können alle Rollen einen Linienstopp aus- oder auflösen. Neben den zuvor vorgestellten Methoden besitzen alle Rollen der Systemarchitektur die Fähigkeit, KVP-Maßnahmen umzusetzen. Tabelle 13 gibt einen Überblick über die Dienste, die von jeder Rolle anzubieten sind.

Tabelle 12 Von der informationstechnischen Schnittstelle des Supermarktes anzubietende Dienste

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
Supermarkt	»liefereLagerdaten«	Antwortadresse: iSID	Lagerdaten	Liefert die Lagerdaten (Orte, Kapazitäten, zugeordnete Produkte etc.) eines Supermarktes.
Supermarkt	»aktualisiereLagerdaten«	neueLagerdaten: Lagerdaten	-	Aktualisiert die Lagerdaten (Orte, Kapazitäten, zugeordnete Produkte etc.) eines Supermarktes.

Tabelle 13 Von der informationstechnischen Schnittstelle jeder iS-Rolle anzubietende Dienste

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
Kurze Regelkreise	»aktualisiereEskalationsstufen«	Aktualisierte Eskalationsstufen: Eskalationsstufen	-	Empfängt eine Beschreibung der geänderten Eskalationsstufen.
8D-Report	»empfange8D-ReportAktualisierung«	8D-Report: 8D-Report	-	Empfängt eine Aktualisierung zu einem zuvor zur Bearbeitung an den Lieferanten geschickten 8D-Report. Die Aktualisierung kann beispielsweise von der vorherigen Arbeitsstation eingeleitete Sofortmaßnahmen nach einem aufgetretenen Produktfehler sein.
Poka Yoke	»empfangeFehler«	Fehlerbeschreibung: Qualitätsfehler	-	Empfängt eine Fehlerbenachrichtigung, die den eigenen Verantwortungsbereich betrifft.
5S	»empfangeKanban5S«	5S-Auftrag: Kanban5S	-	Löst eine 5S-Aktivität aus.
Andon	»empfangeKanbanSupport«	aktuelleStörung: KanbanSupport	-	Empfängt und verarbeitet eine Kanban mit einer Supportanfrage. Hierdurch kann ein Hancho bei einer Störung bei jeder Unterstützung anfordern.
Andon	»empfangeLinienstopp«	-	-	Gibt das Signal, dass die Fertigungslinie gestoppt werden muss. Entspricht der Andon-Reißleine.
Andon	»empfangeLinienstoppaktualisierung«	Status: StatusLinienstopp	-	Übermittelt eine Statusaktualisierung für einen vorausgehenden Linienstopp. Wird u.a. verwendet, um nach einer behobenen Störung die Linie wieder zu starten.
Prozess- & Performancemanagement; Prozessstandardisierung	»empfangeProzessbeschreibung«	Prozess: Prozessbeschreibung	boolean	Empfängt einen beschriebenen Prozess, welcher übernommen werden soll und dessen Akzeptanz zu quittieren ist (true = angenommen).
(mehrere Lean-Methoden)	»empfangeVerbesserungsmaßnahme«	Verbesserungsmaßnahme: Verbesserungsmaßnahme	-	Empfängt eine Verbesserungsmaßnahme, die die iS-Rolle umsetzen soll.
(mehrere Lean-Methoden)	»empfangeVerbesserungsmaßnahmeAktualisierung«	Status: StatusVerbesserungsmaßnahme	-	Gibt dem Ersteller eine Aktualisierung zu einer zuvor übergebenen Verbesserungsmaßnahme.
Andon	»empfangeStatusKanbanSupport«	Status: StatusKanban	-	Empfängt eine übergebene Statusaktualisierung zu einer von der iS-Rolle zuvor versendeten Support-Kanban. Hierdurch kann z.B. die Arbeitsstation über Verzögerungen und die Fertigstellung der Unterstützung informiert werden.
Ideenmanagement	»empfangeVerbesserungsvorschlagsAktualisierung«	Status: StatusVerbesserungsvorschlag	-	Empfängt eine Aktualisierung zu einem zuvor eingereichten Verbesserungsvorschlag.
8D-Report	»liefere8D-Reportliste«	Antwortadresse: iSID	Liste8D-Reports	Liefert eine Liste aller 8D-Reports, die bei der iS-Rolle aufgetreten sind. Hierdurch kann das KVP-System eine Auswertung für KVP-Maßnahmen erstellen.
Prozessstandardisierung; Prozess- & Performancemanagement	»liefereProzessbeschreibungen«	Antwortadresse: iSID	ListeProzessbeschreibungen	Liefert eine Liste mit durchführbaren Prozessen.
-	»liefereRollentyp«	-	String	Liefert den Typen der iS-Rolle, um es Dritten zu ermöglichen, die Funktionalitäten einer angesprochenen Instanz in Erfahrung zu bringen.

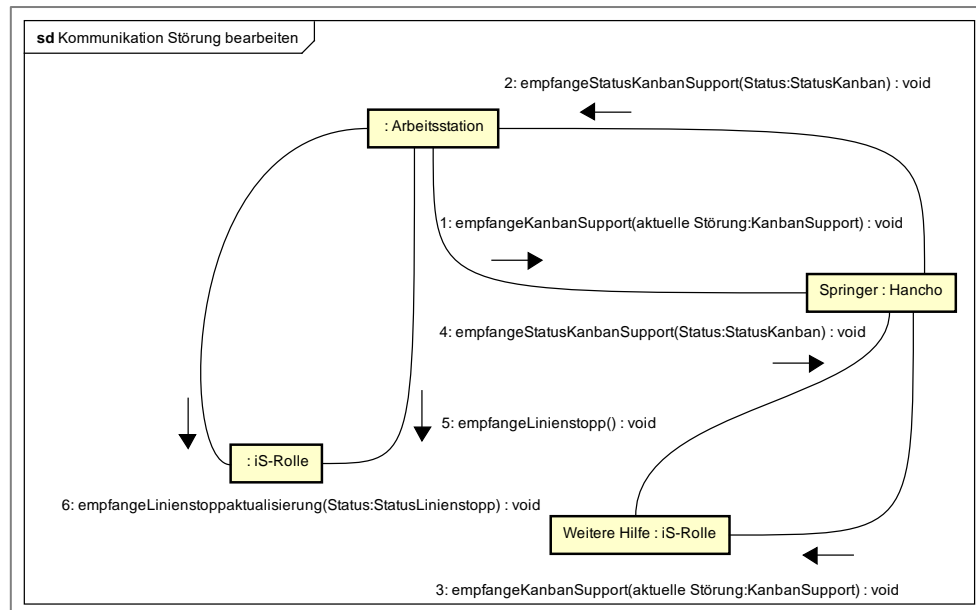


Abbildung 61 Kommunikationsdiagramm für den Anwendungsfall „Störung bearbeiten“ der Andon-Methode

Der **Hancho** nimmt im Rahmen der Referenzarchitektur eine besondere Rolle ein. Diese Rolle bietet keine speziellen Dienste für Dritte an, greift aber auf die Dienste der anderen Rollen zu. Dienste, die beim Hancho aufgerufen werden, sind über die abstrakte iS-Rolle bereits abgedeckt, welche der Hancho erbt. Abbildung 61 zeigt den Aufruf der geerbten Dienste am Beispiel der Andon-Methode.

4.3.2.3 Ausgetauschte Nachrichten und deren Inhalte

Neben den sechs Rollen wurden aus den Satzschablonen der funktionalen Anforderungen 43 Nachrichtentypen abgeleitet, die die Rollen zur Realisierung der Lean-Methoden miteinander austauschen. Die vollständige Spezifikation mit den identifizierten Pflichtattributen ist im Anhang F aufgeführt. Aufgrund der Vielzahl erläutert dieses Kapitel nur einen Ausschnitt der Nachrichten aus den zuvor beschriebenen Lean-Methoden.

Beim Eintreffen eines Kundenauftrags übergibt die Auftragsfreigabe unter Verwendung des Dienstes „empfangenKanbanAuftrag“ eine Nachricht vom Typ **KanbanAuftrag**, die eine Spezialisierung vom Typ Kanban ist, an die Arbeitsstation. KanbanAuftrag enthält alle von der Arbeitsstation für die Fertigung eines Auftrags benötigten Informationen. Neben einer Identifikationsnummer der Kanban sind das die Identifikationsnummer und Bezeichnung des zu fertigenden Produktes, dessen Losgröße, der für den Transport zu verwendende Werkstückträger sowie die Adresse des Ablieferortes der Produkte. Das Attribut „Lieferzeit“ gibt ferner den Zeitpunkt der Fertigstellung an. Das Attribut „Gültigkeit“ ist ein Ablaufdatum für eine Kanban und wird nur dann benötigt, wenn alte Produkte nicht mehr gefertigt werden und die Kanban somit ihre Gültigkeit verliert. Tabelle 14 gibt einen Überblick über alle Attribute sowie Beispiele hierfür.

Tabelle 14 Attribute des Nachrichtentypes KanbanAuftrag

Attribut	Inhalt/Ausprägung	Beispiel
»AuftragsID«	Identifikationsnummer dieser Kanban	Fortlaufende Nummer oder Zeitstempel
»AdresseWarenausgang«	Ablieferort der gefertigten Produkte	Kostenstelle oder Abstellflächen-Nr.
»ProduktID«	Individuelle Bezeichnung des Werkstücks	Artikelnummer
»ProduktName«	Menschenlesbarer Name des Werkstücks	Artikelname
»Losgröße«	Bedarfsmenge bzw. mit dieser Kanban zu fertigende Menge	Zahlenwert > 0
»WerkstückträgerTyp«	Ladehilfsmittel für den Transport	Typ- oder Größenangabe eines Behälters
»AdresseSenke«	Empfänger der Kanban	Kostenstelle oder Abstellflächen-Nr.
»Lieferzeit«	Optional; Bedarfszeitpunkt (vgl. Taktzeit-Ende)	Zeitstempel im Format YYYY-MM-dd HH:MM
»Gültigkeit«	Optional; Datum, bis wann diese Kanban gültig war	Zeitstempel im Format YYYY-MM-dd HH:MM, wenn ein Ablaufdatum bekannt, ansonsten leer

Der Nachrichtentyp **KanbanFertigung** ist ebenfalls vom Typ Kanban und enthält die zwischen den Arbeitsstationen ausgetauschten Informationen zur Fertigung eines Teiles (siehe Tabelle 15). Es enthält neben dem Namen der Kanban und der Identifizierungsnummer auch den Namen, die Nummer und die Menge des zu fertigenden Produktes sowie den Werkstückträger-Typ, mit dem es transportiert wird. Für den Transport ist außerdem angegeben, welche Arbeitsstation das Produkt anfordert, welche Arbeitsstation der Empfänger und somit der Ausfühler ist und welcher Werkstückträger zu verwenden ist. Neben diesen Pflichtangaben enthält es ebenfalls optional die Attribute „Lieferzeit“, um zu beschreiben, bis wann das Produkt einzutreffen hat, und „Gültigkeit“ zur Beschreibung des Ablaufdatums. KanbanFertigung enthält zusätzlich noch das Attribut „Taktzeit“. In klassischen Lean-Fertigungslinien, die nur ein Produkt mit einem festen Takt fertigen, ist dieses Attribut nicht notwendig. Sofern in einer Arbeitsstation mehrere Produkte gefertigt werden und in Hinblick auf die angestrebte Verbesserung der Wandelbarkeit, wird dieses Attribut optional mit aufgenommen, um unterschiedliche Taktzeiten zu ermöglichen.

Tabelle 15 Attribute des Nachrichtentypes KanbanFertigung

Attribut	Inhalt/Ausprägung	Beispiel
»KanbanID«	Identifikationsnummer dieser Kanban	Fortlaufende Nummer oder Zeitstempel
»Losgröße«	Losgröße des zu fertigenden Produktes	Zahlenwert > 0
»AdresseQuelle«	Versender der Kanban	Arbeitsstation-Nr. oder Kostenstelle
»AdresseSenke«	Empfänger der Kanban	Arbeitsstation-Nr. oder Kostenstelle
»ProduktID«	Individuelle Bezeichnung des Werkstücks	Artikelnummer
»ProduktName«	Menschenlesbarer Name des Werkstücks	Artikelname
»WerkstückträgerTyp«	Ladehilfsmittel für den Transport	Typ- oder Größenangabe eines Behälters
»Lieferzeit«	Optional; Lieferzeitpunkt der Produkte	Zeitstempel im Format YYYY-MM-dd HH:MM
»Taktzeit«	Optional; Taktzeit für das zu fertigende Produkt	Zahlenwert >0 in Minuten
»Gültigkeit«	Optional; Datum, bis wann diese Kanban gültig war	Zeitstempel im Format YYYY-MM-dd HH:MM, wenn ein Ablaufdatum bekannt; ansonsten leer

Tabelle 16 Attribute des Nachrichtentypes KanbanTransport

Attribut	Inhalt/Ausprägung	Beispiel
»TransportID«	Identifikationsnummer dieser Kanban	Fortlaufende Nummer oder Zeitstempel
»AdresseQuelle«	Abholort der Ware	Lagerplatz-Nr. oder Kostenstelle
»AdresseSenke«	Ablieferort der Ware	Lagerplatz-Nr. oder Kostenstelle
»WerkstückträgerTyp«	Typ des Werkstückträgers für den Transport	Typ- oder Größenangabe eines Behälters
»Werkstückträger Anzahl«	Anzahl der zu transportierenden Werkstückträger	Zahlenwert > 0
»Lieferzeit«	Ablieferzeitpunkt (vgl. Taktzeit-Ende)	Zeitstempel im Format YYYY-MM-dd HH:MM
»Taktzeit«	Optional; Taktzeit für das zu transportierende Produkt	Zahlenwert >0 in Minuten
»Gültigkeit«	Optional; Datum, bis wann diese Kanban gültig war	Zeitstempel im Format YYYY-MM-dd HH:MM, wenn ein Ablaufdatum bekannt; ansonsten leer

Die Nachricht **KanbanTransport**, welche ebenfalls vom Typ Kanban ist, beschreibt Transportbedarfe für das Logistiksystem. Die KanbanTransport beschreibt den Quell- und Zielort von zu transportierenden Waren. Darüber hinaus enthält sie Informationen über die Art des Werkstückträgers sowie deren Menge und den Zeitpunkt, wann die Ware am Zielort eintreffen muss. Die Ware bzw. das Produkt selbst muss hierbei nicht dem Logistiksystem bekannt sein, da für den Transport ausschließlich die Information über den Werkstückträger relevant ist und nicht dessen Inhalt. Tabelle 16 gibt einen Überblick über die Attribute sowie Beispiele hierfür.

Neben den zuvor genannten Spezialisierungen des Nachrichtentyps Kanban existiert noch der Nachrichtentyp **Kanban5S**, welcher die Informationen für eine 5S-Aktivität beim Empfänger enthält und diese auslöst. Auch die bei den Nachrichten der Andon-Methode beschriebene **KanbanSupport** ist eine Spezialisierung der Nachricht Kanban. Abbildung 62 stellt einen Überblick über die aus dem Typ Kanban abgeleiteten Nachrichtentypen dar.

Die Kanban-Methode verwendet den Nachrichtentyp **ListeKanban**, der eine Sammlung von Kanban beinhaltet. Die Dienste „liefereEmpfangbareKanbanListe“ und „liefereVerschickbareKanbanListe“ der Arbeitsstation liefern mit dem Nachrichtentyp die verschickbaren oder empfangbaren Kanban zurück. Es besteht somit ausschließlich aus einer beliebigen Anzahl an Kanban und hat darüber hinaus keine weiteren Attribute. Moderne Programmiersprachen wie Java bieten hierfür einen eigenen Datentypen, der für die Liste verwendet werden kann.

Mit dem Nachrichtentyp **StatusKanban** kann eine Arbeitsstation die Statusaktualisierung für jegliche Arten von Kanban übermitteln. Die StatusKanban enthält hierzu die betroffene Kan-

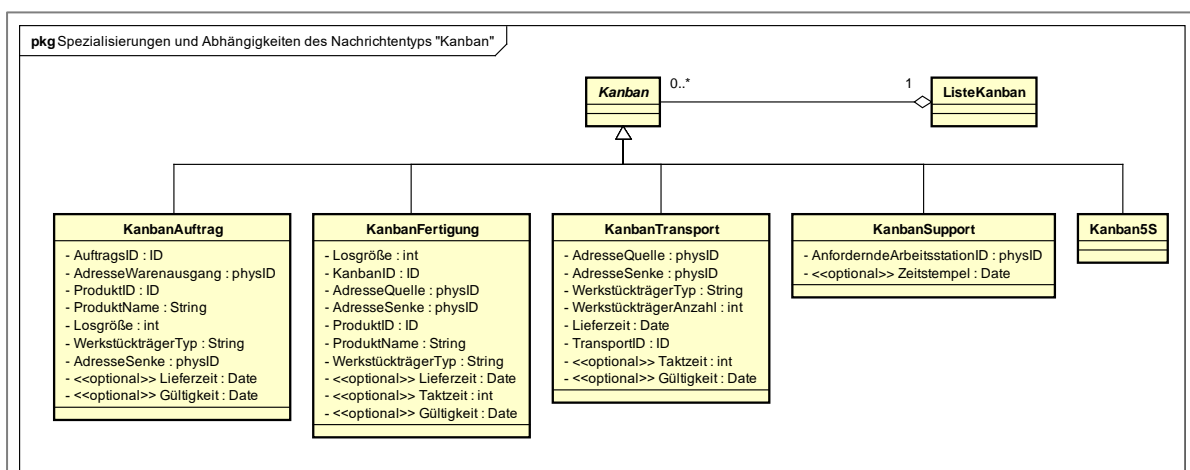


Abbildung 62 Spezialisierungen und Abhängigkeiten des Nachrichtentyps „Kanban“

ban-Nr. und einen Zeitstempel, wann die Aktualisierung erstellt wurde. Außerdem enthält sie den Status, wobei hier die Ausprägungen „Fertig“ und „Verzögerung“ vorgesehen sind. Für die später vorgestellte KanbanSupport gibt es zusätzlich die Ausprägung „Anwesend“. Sowohl die Empfangsquittierung einer Kanban als auch eine Angabe eines neuen Erfüllungsdatums bei Verzögerungen sind für die klassische Lean Production nicht vorgesehen. Sie können, falls für die Implementierung gefordert, optional hinzugefügt werden. Der Nachrichtentyp StatusKanban ist in Tabelle 17 dargestellt.

Zur Realisierung des Dienstes „liefereAndonKennzahlen“ der Andon-Methode wird der Nachrichtentyp **KennzahlenAndon** benötigt, der die relevanten Kennzahlen einer Arbeitsstation enthält. Abgeleitet aus dem übergeordneten Ziel der Lean Production ergeben sich für eine Arbeitsstation die mindestens erforderlichen Attribute „Aktuelle Taktzeit“, „Aktuelle Ausbringungsmenge“ und „Anzahl Störungen“ sowie das Zeitintervall, auf welches sich diese Zahlen beziehen (siehe Tabelle 18). Der Nachrichtentyp **KennzahlenAndonSoll** (siehe Tabelle 19) ist ein eigenständiger Nachrichtentyp und im Nachrichtentyp KennzahlenAndon enthalten. Bis auf die Anzahl der Störungen besitzt er die gleichen Attribute. Er bezieht sich allerdings auf die Soll-Werte, nicht die Ist-Werte.

Die Lean Production fordert neben den Kennzahlen für die Andon-Methode weitere **Kennzahlen**. In dem Informationsmodell hat jede Lean-Methode ihren eigenen Nachrichtentyp, mit dem Dritte die für diese Methode relevanten Kennzahlen abrufen können. Hierdurch kann es zu Redundanzen bei Kennzahlen kommen, da unterschiedliche Nachrichtentypen gleiche Werte beinhalten können. Allerdings bietet es die Möglichkeit, Individualisierungen und spätere Ergänzungen mit wenig Aufwand umzusetzen, da dem zur Lean-Methode zugehörigen Nachrichtentyp nur ein weiteres Attribut hinzuzufügen ist. Ferner bezieht sich diese Redundanz auf die Nachrichtentypen. Für die Erzeugung der Nachricht kann eine iS-Rolle durchaus den Wert an derselben Stelle speichern und auslesen. Abbildung 63 zeigt die weiteren Nachrichtentypen zum Austausch von Kennzahlen in Abhängigkeit von ihrer Generalisierung „Kennzahlen“.

Tabelle 17 Attribute des Nachrichtentypes StatusKanban

Name:	Inhalt/Ausprägung	Beispiel
»BetroffeneKanban«	Kanban	Gesamte Kanban oder deren ID als Referenz
»Status«	Kanban-Status gemäß Vorgabetypen	Vorgabe-Typen: „Fertig“, „Verzögerung“, „Anwesend“
»Zeitstempel«	Zeitpunkt der Erstellung	Zeitstempel im Format YYYY-MM-dd HH:MM:ss

Tabelle 18 Attribute des Nachrichtentypes KennzahlenAndon

Attribut	Inhalt/Ausprägung	Beispiel
»KennzahlenAndonSoll«	Enthält die Soll-Kennzahlen einer Arbeitsstation	Soll-Taktzeit, -Ausbringungsmenge, -Zeitintervall
»IstTaktzeit«	Tatsächliche Taktzeit der Arbeitsstation	Zeitangabe in Minuten pro Stück
»ZeitIntervallMinuten«	Zeitintervall, auf das sich die Ist-Kennzahlen beziehen	Angabe in Minuten
»AnzahlStörungen«	Anzahl der Störungen im Zeitintervall	Zahlenwert > 0
»IstAusbringungsmenge«	Anzahl der produzierten Produkte in einem Zeitintervall	Zahlenwert > 0

Tabelle 19 Attribute des Nachrichtentypes KennzahlenAndonSoll

Attribut	Inhalt/Ausprägung	Beispiel
»SollTaktzeit«	Vorgabe-Taktzeit für die Arbeitsstation	Zeitangabe in Minuten pro Stück
»SollAusbringungsmenge«	Vorgabe-Ausbringungsmenge für die Arbeitsstation	Zahlenwert > 0 in Stück
»ZeitIntervallMinuten«	Zeitintervall, auf das sich die Ist-Kennzahlen beziehen	Angabe in Minuten

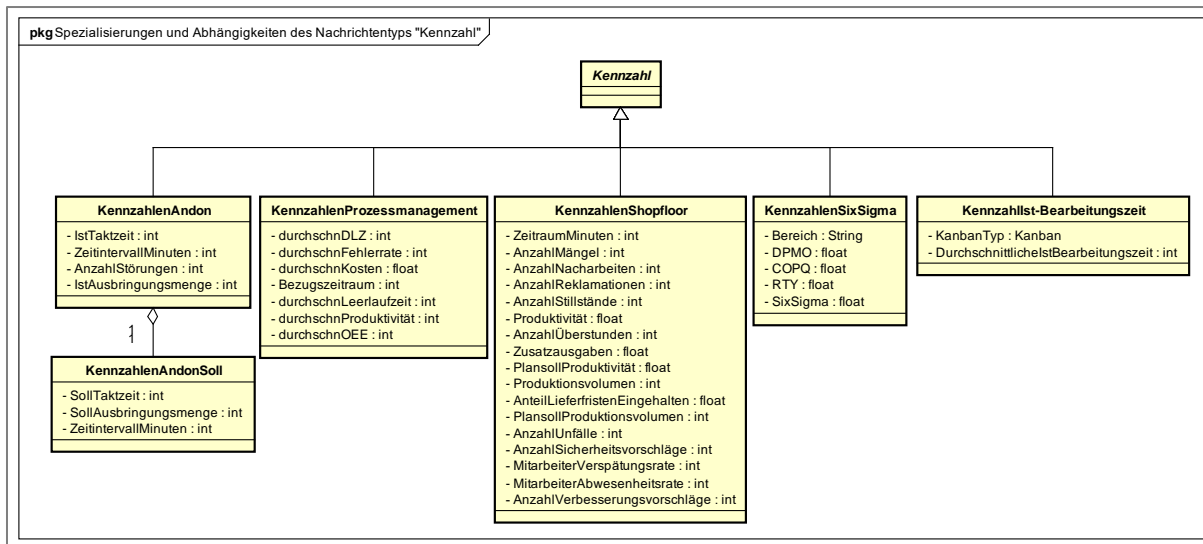


Abbildung 63 Spezialisierungen und Abhängigkeiten des Nachrichtentyps „Kennzahlen“

Der Nachrichtentyp **StatusArbeitsstation**, der beim Aufruf von „liefereStatus“ zurückgegeben wird, enthält die Identifikationsnummer der abgefragten Arbeitsstation und dessen Status (siehe Tabelle 20). Beim Status sind gemäß [Ver13] fünf verschiedene Ausprägungen möglich:

- Status 0: Betriebsbereit
- Status 1: Unterstützung benötigt
- Status 2: Unterstützung kommt
- Status 3: Zusätzliche Unterstützung benötigt
- Status 4: Arbeitsstation-Stopp

Der Nachrichtentyp **Linienstopp** löst den Stopp einer Fertigungslinie aus. Eine Arbeitsstation sendet diesen Nachrichtentyp im Falle eines größeren Problems an alle Arbeitsstationen. Es enthält den Zeitpunkt, wann der Stopp ausgelöst wurde, sowie die Identifikationsnummer der auslösenden Arbeitsstation. Die Arbeitsstation kann mit einer **StatusLinienstopp**-Nachricht den Linienstopp aufheben bzw. bei längeren Verzögerungen, beispielsweise über einen Takt hinaus, auf den nach wie vor vorliegenden Stopp-Status hinweisen. Tabelle 21 und Tabelle 22 geben einen Überblick über die Attribute.

Tabelle 20 Attribute des Nachrichtentypes StatusArbeitsstation

Name:	Inhalt/Ausprägung	Beispiel
»ArbeitsstationID«	Identifikationsnummer der Arbeitsstation	Arbeitsstation-Nr. oder Kostenstelle
»Status«	Status gemäß Andon-Einteilung	5-stufiges Model der VDI-Richtlinie 2870

Tabelle 21 Attribute des Nachrichtentypes Linienstopp

Name:	Inhalt/Ausprägung	Beispiel
»ArbeitsstationID«	Identifikationsnummer der Arbeitsstation	Arbeitsstation-Nr. oder Kostenstelle
»Zeitstempel«	Zeitpunkt des Stopps	Zeitstempel im Format YYYY-MM-dd HH:MM:ss

Tabelle 22 Attribute des Nachrichtentypes StatusLinienstopp

Name:	Inhalt/Ausprägung	Beispiel
»AuflösendeArbeitsstationID«	Identifikationsnummer der Arbeitsstation	Arbeitsstation-Nr. oder Kostenstelle
»Status«	Status des Linienstopps	Ausprägung „Stopp“ oder „Beendet“

Im Rahmen der Andon-Methode kann das KVP-System zur Realisierung des Anwendungsfalls „Störung auslesen“ den Dienst „liefereStörungsliste“ bei der Arbeitsstation aufrufen, wodurch es eine Nachricht vom Typ **ListeStörungen** erhält. Der Nachrichtentyp enthält eine Liste aller an einer Arbeitsstation aufgetretenen Störungen (siehe Tabelle 23). Ähnlich wie bei der ListeKanban handelt es sich hierbei um eine Sammlung von Elementen. In diesem Fall sind es Störungen, welche durch einen Zeitstempel, den Störungstyp, der zugeordneten Arbeitsstation und einer optionalen Beschreibung beschrieben sind (siehe Tabelle 24). Bei den Störungstypen unterscheidet die Lean Production gemäß [Tak12] zwischen den Arten:

- Aktive Sicherheitsbeeinträchtigung
- Passive Sicherheitsbeeinträchtigung
- Qualitätsmängel
- Maschinendefekt
- Materialmangel
- Rückstand
- Störung beim Umrüsten

Auch wenn für die Durchführung der SMED-Methode innerhalb einer Arbeitsstation regelmäßig Daten erfasst werden, ist für den Austausch mit Dritten nur ein Nachrichtentyp relevant. Der Dienst „liefereRüstvorgängeListe“ liefert **ListeRüstvorgänge** zurück, mit denen das KVP-System von einer Arbeitsstation alle protokollierten Rüstzeiten abrufen kann. Die Rüstzeiten umfassen neben der internen Rüstzeit und der externen Vor- und Nachbereitung auch die Nummer der Arbeitsstation und das Produkt (siehe Tabelle 25 und Tabelle 26).

Tabelle 23 Attribute des Nachrichtentypes ListeStörungen

Name:	Inhalt/Ausprägung	Beispiel
»Störung[]«	Liste (Array) mit Störungen an einer Arbeitsstation	Liste mit Störungen einer Arbeitsstation

Tabelle 24 Attribute des Nachrichtentyps StörungenArbeitsstation

Attribut	Inhalt/Ausprägung	Beispiel
»Zeitstempel«	Zeitpunkt, wann die Störung aufgetreten ist	Zeitstempel im Format YYYY-MM-dd HH:MM
»Störungstyp«	Art der Störung gemäß [Tak12]	Ausprägungen „Sicherheitsbeeinträchtigung aktiv“, „Sicherheitsbeeinträchtigung passiv“, „Qualitätsmängel“, „Maschinendefekt“, „Materialmangel“, „Rückstand“, „Umrüsten“
»ArbeitsstationID«	Identifikationsnummer der Arbeitsstation	Arbeitsstation-Nr. oder Kostenstelle
»Störungsbeschreibung«	Optional; Beschreibung der Störung	Freitext

Tabelle 25 Attribute des Nachrichtentypes ListeRüstvorgänge

Name:	Inhalt/Ausprägung	Beispiel
»Rüstzeit[]«	Liste (Array) mit protokollierten Rüstzeiten	Liste mit Rüstzeiten für die jeweiligen Produkte

Tabelle 26 Attribute des Nachrichtentypes Rüstzeit

Attribut	Inhalt/Ausprägung	Beispiel
»ProduktID«	Individuelle Bezeichnung des Werkstücks	Artikelnummer
»ArbeitsstationID«	Identifikationsnummer der Arbeitsstation	Arbeitsstation-Nr. oder Kostenstelle
»DauerExterneVorbereitung«	Dauer für die Vorbereitung des Rüstens während der Laufzeit der Anlage	Zahlenwert >= 0 in Minuten
»DauerInterneDurchführung«	Dauer für das Rüsten während des Stillstands der Anlage	Zahlenwert >= 0 in Minuten
»DauerExterneNachbereitung«	Dauer für die Nachbereitung des Rüstens während der Laufzeit der Anlage	Zahlenwert >= 0 in Minuten

4.3.2.4 Zusammenfassung

Das Kapitel stellte das Informationsmodell der Referenzarchitektur vor. Bei der Definition wurden die unterschiedlichen Ansätze zur digitalen Kommunikation aus Kapitel 2.1.3 berücksichtigt. Für die sechs in der übergeordneten Systemarchitektur relevanten Rollen wurde das Informationsmodell definiert, welches die an den informationstechnischen Schnittstellen anbietenden Dienste beschreibt (siehe Abbildung 64). Zur Beschreibung der insgesamt 61 Dienste gehören neben der Aktivität, die Dritte an den informationstechnischen Schnittstellen aufrufen können, auch die Eingabe- und Ausgabewerte dieser Dienste. Abbildung 65 zeigt die für den Datenaustausch zwischen Entitätstypen definierten 71 Klassen inklusive deren Attribute. Von diesen Klassen werden 43 im Rahmen der Dienstaufrufe als Nachrichten ausgetauscht.

Das Informationsmodell ist hierbei mittels unterschiedlichster Programmiersprachen und Kommunikationsprotokolle realisierbar. Bei der Spezifikation der Attribute in den Nachrichten verzichtet sie, bis auf die gängigen Grundtypen wie Strings oder Integer, auf programmiersprachenspezifische Eigenschaften. Für die Kommunikation berücksichtigt sie unterschiedliche Arten des Dienstaufrufes in verteilten Systemen. Da die Digitalisierung der Lean-Methoden im Fokus steht, verzichtet sie auch auf implementierungsspezifische, zusätzliche Dienste z.B. zur Erkennung von Netzwerkteilnehmern, An- und Abmeldeverwaltung, Fehlerbehandlung bei verlorenen Datenpaketen etc. Mögliche Ansätze hierfür sowie eine exemplarische Umsetzung der informationstechnischen Schnittstellen sind in Kapitel 5 beschrieben.

Als letzter Teil der Referenzarchitektur wird die Schnittstellenarchitektur vorgestellt. Diese fokussiert sich auf die Aufteilung der benötigten Softwarekomponenten, um die informationstechnischen Schnittstellen auf unterschiedlichen Geräten umzusetzen und bestehende Produktionsumgebungen nachzurüsten.

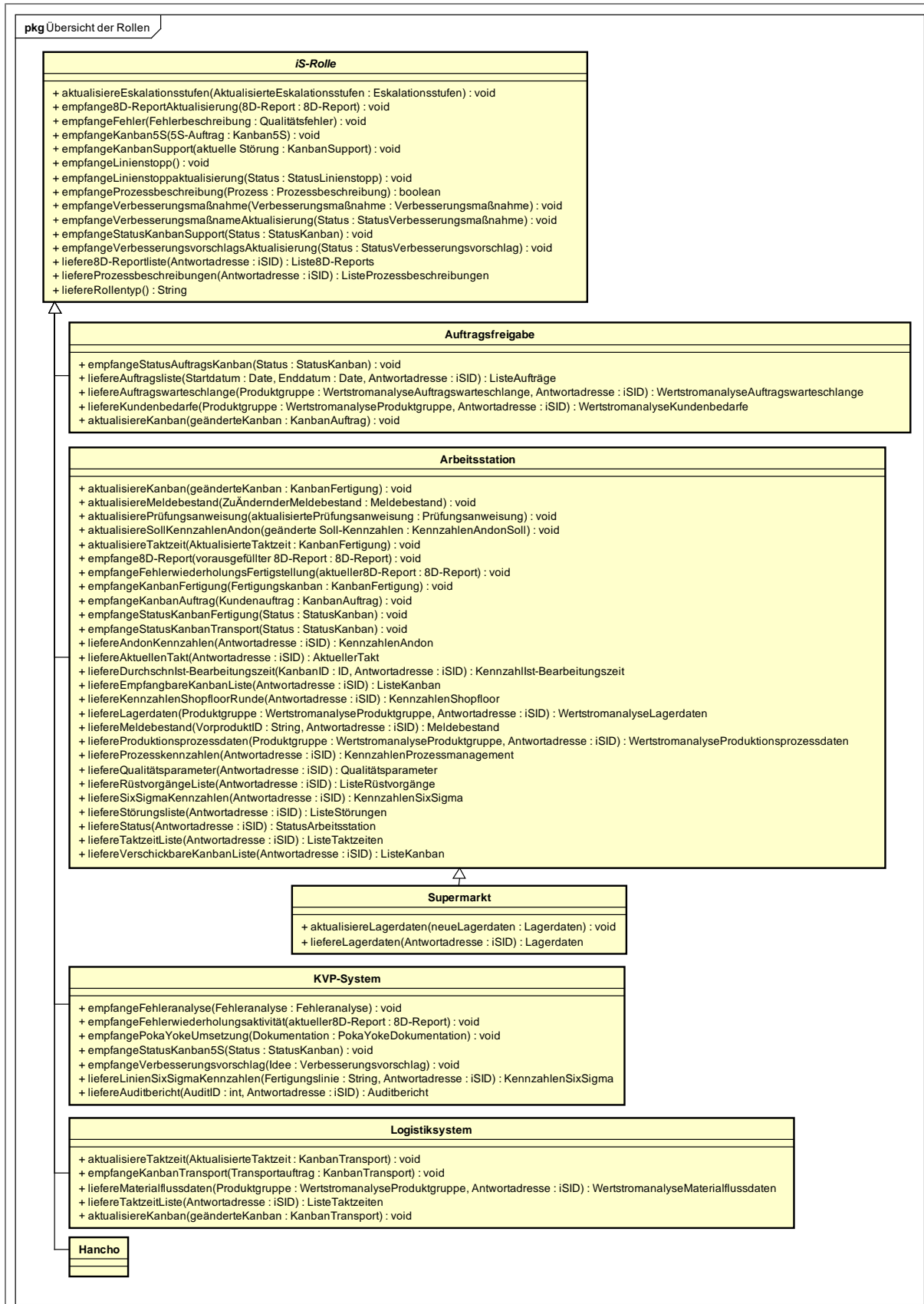


Abbildung 64 Klassendiagramm der Rollen und deren anzubietenden Dienste

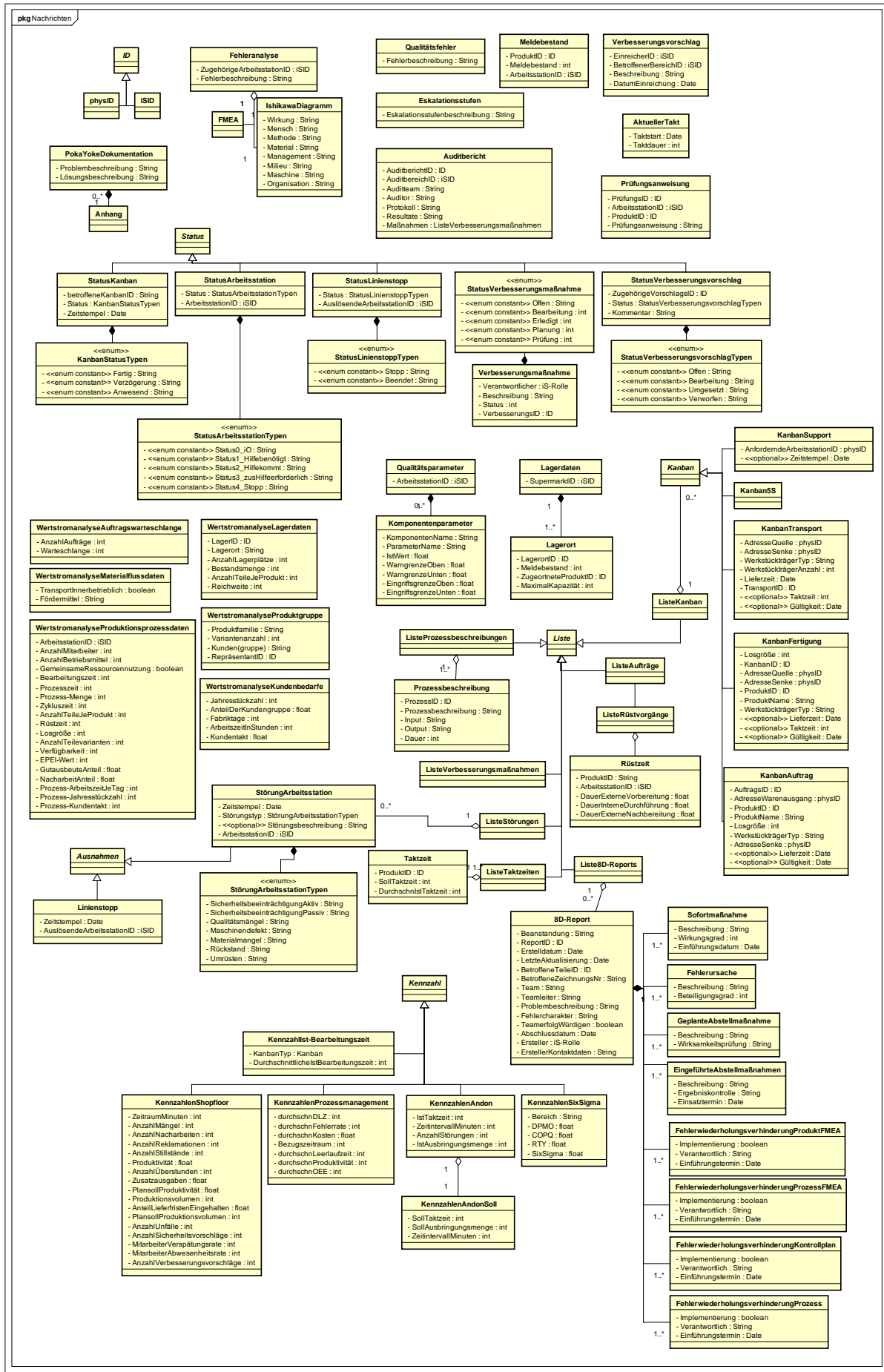


Abbildung 65 Klassendiagramm der Nachrichtentypen

4.3.3 Schnittstellenarchitektur

In der Praxis liegt nicht immer eine durchgängig an ein gemeinsames Kommunikationsnetzwerk angebundene Produktionsumgebung vor. Die alleinige Definition von Rollen mit informationstechnischen Schnittstellen und eines Informationsmodells reicht nicht aus, da sie offenlässt, wie sich bestehende Anlagen anbinden lassen. Dieses Unterkapitel stellt im Folgenden eine Schnittstellenarchitektur vor, die skalierbare und wandelbare informationstechnische Schnittstellen in bestehenden, heterogenen Umgebungen ermöglicht.

Ausgehend von einer noch nicht digital angebundenen Rolle mit einem Rechner oder einer Steuerung ergeben sich drei zu berücksichtigende Migrationsmöglichkeiten. Die erste Möglichkeit ist der direkte Austausch der Steuerung bzw. des Rechners. Hierbei wird die bestehende Ressource durch ein Gerät ausgetauscht, das die informationstechnische Schnittstelle anbietet. Die zweite Möglichkeit ist die Umgruppierung von Steuerungen bzw. Rechnern. Bestehende Arbeitsstationen werden in einer Rolle logisch neu gruppiert, und eine nachträglich installierte Steuerung bzw. ein nachträglich installierter Rechner bietet die informationstechnische Schnittstelle der Rolle an. Andersherum lässt sich auch eine größere Arbeitsstation in zwei separate Rollen aufteilen, die wiederum ihre eigenen Schnittstellen anbieten. Die dritte zu berücksichtigende Migrationsmöglichkeit ist das Nachrüsten eines Adapters. Die logische Zugehörigkeit der Ressourcen zu Rollen bleibt dabei unverändert und es wird eine zusätzliche Ressource integriert, die die informationstechnische Schnittstelle anbietet.

Abbildung 66 visualisiert die Möglichkeiten des Nachrüstens. Die dritte Alternative ist hierbei eine effiziente Migrationsmöglichkeit, da Adapter in der Regel kostengünstiger als Steuerungen oder Rechner sind und bestehende Systeme weitestgehend unverändert bleiben. Auch ist in dieser Migrationsmöglichkeit die Kommunikation im gemeinsamen Netzwerk von der Prozesssteuerung entkoppelt, sodass beide unabhängig voneinander austauschbar sind.

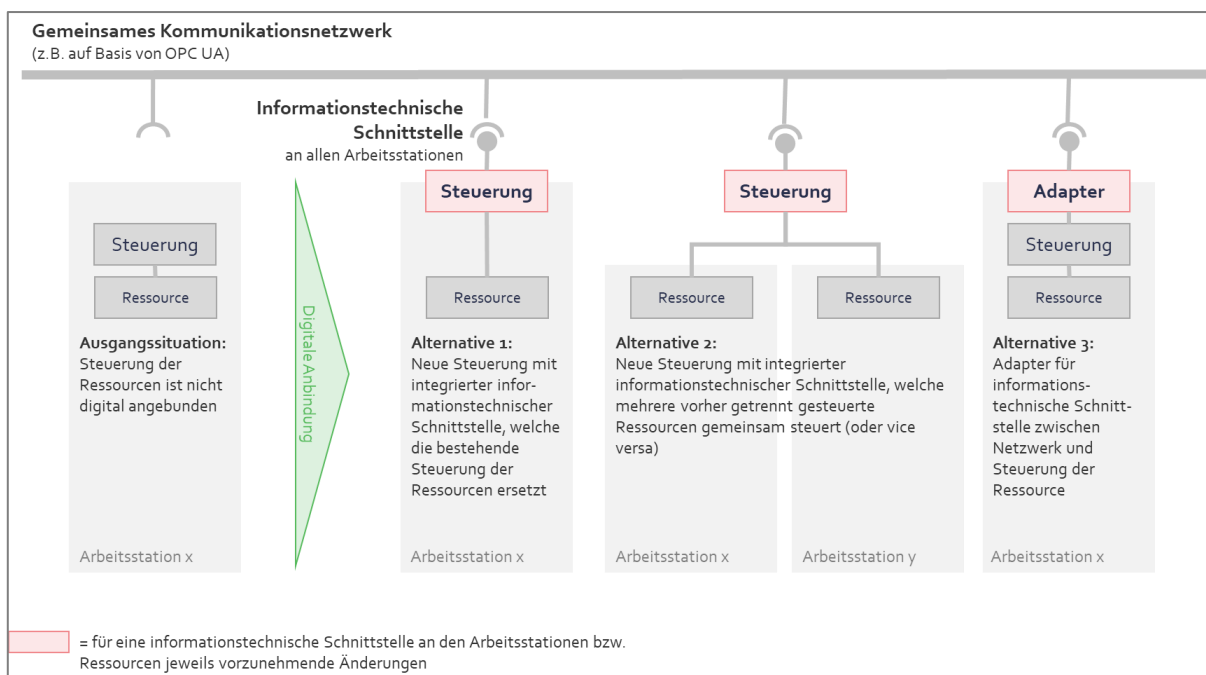


Abbildung 66 Handlungsalternativen zum Nachrüsten einer informationstechnischen Schnittstelle

Aus Hardware-Sicht muss die Schnittstellenarchitektur auf die identifizierten Ressourcen „Rechner“, „Steuerung“ und „HMI“ möglichst einheitlich übertragbar sein. Eine Hardware muss immer über eine Netzwerkschnittstelle für die Kommunikation und ausreichend Rechenleistung für die Datenverarbeitung verfügen. Je nach Rolle und Gegebenheiten muss sie eine HMI-Schnittstelle, die Anschlussmöglichkeit von Feldgeräten oder eine Schnittstelle für angeschlossene Steuerungen besitzen. Bei den Rollen KVP-System und Auftragsfreigabe liegt es nahe, die informationstechnische Schnittstelle direkt auf dem jeweiligen Rechner zu installieren. Auch z.B. vom Hancho genutzte mobile Geräte wie Tablets können die jeweilige informationstechnische Schnittstelle anbieten. Beim Logistiksystem und der Arbeitsstation wäre die zusätzliche Installation auf einem Industrie-PC, der auch die SPS virtualisiert, oder einer SPS mit entsprechender Flexibilität denkbar. Alternativ muss für die Anbindung ein Adapter verwendet werden, welcher z.B. auch bei Regalen der Supermarkt-Rolle notwendig ist. Die im Stand der Technik vorgestellten CPSe sind aufgrund der genannten Eigenschaften besonders geeignet.

Aus Software-Sicht übernimmt eine informationstechnische Schnittstelle drei Aufgaben. Sie interagiert mit den anderen Rollen im gemeinsamen Netzwerk, steuert die an sie angebotenen Ressourcen und verarbeitet Daten, um sie zwischen den Rollen und Ressourcen zu vermitteln (siehe [Lei09]). Für die softwareseitige Realisierung der angestrebten Skalierbarkeit und Wandelbarkeit fließen die aus der Softwaretechnik bekannten Konzepte und Prinzipien des objektorientierten Designs ein (siehe z.B. [Gol14; Bal11]). Hierzu gehört u.a. das Konzept der Kapselung, wonach nur benötigte Informationen nach außen angeboten werden. Die Schnittstellenarchitektur definiert zum gemeinsamen Netzwerk hin eine für jede Instanz einer Rolle technologieunabhängige, identische Schnittstelle. Für die Steuerung der Ressourcen muss die Schnittstellenarchitektur bestehende, heterogene Komponenten u.a. mit proprietären Treibern kapseln und einheitlich integrieren. Für die dritte Aufgabe, die Steuerung des Prozesses und Vermittlung von Daten, muss die Schnittstellenarchitektur in sich skalierbar sein. Je nach gewählter Migrationsmöglichkeit muss die Schnittstelle mehr oder weniger Aufgaben der Prozesssteuerung übernehmen. Neben den Grundfunktionalitäten für die informationstechnische Schnittstelle sieht die Schnittstellenarchitektur somit Komponenten für die Steuerung und Überwachung vor. Aus diesen Vorüberlegungen leitet sich zusammen mit dem Anforderungskatalog die Schnittstellenarchitektur ab. Sie ist in Abbildung 67 dargestellt und wird im Folgenden erläutert.

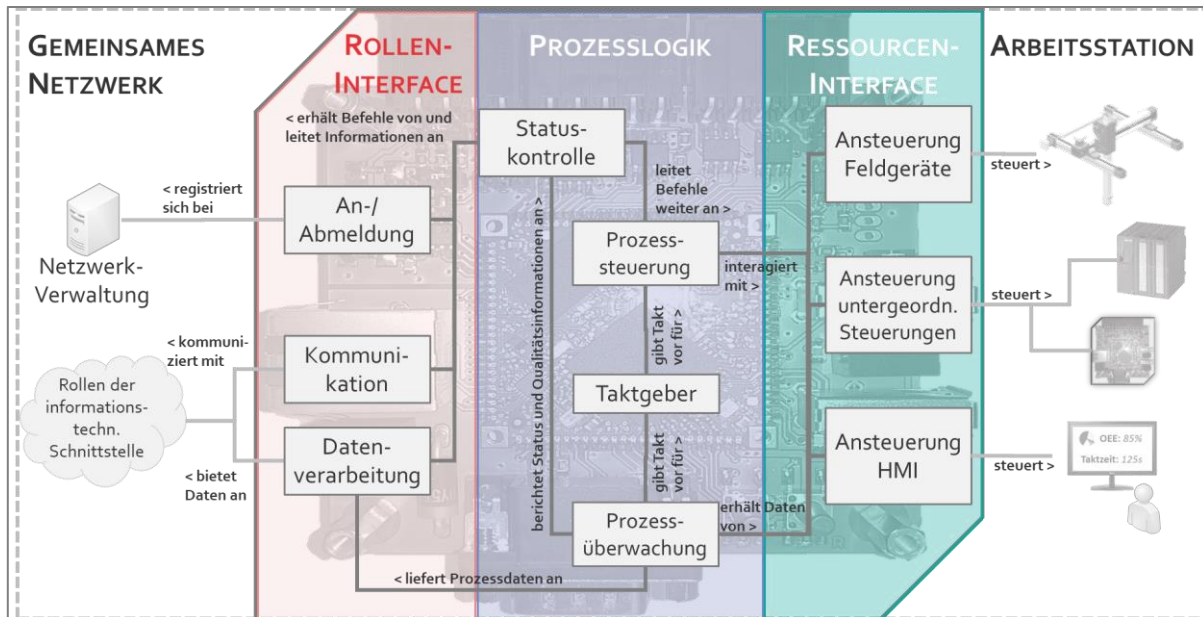


Abbildung 67 Schnittstellenarchitektur (Quelle: [Kol+16])

Eine informationstechnische Schnittstelle ist in ihrer umfangreichsten Form als Adapter, z.B. als CPS, zwischen den Ressourcen und deren Steuerung an der Arbeitsstation und dem gemeinsamen Kommunikationsnetzwerk installiert. Die **Rollen-Interface-Gruppe** repräsentiert die Instanz einer Rolle innerhalb des gemeinsamen Netzwerks. Sie beinhaltet alle Softwarekomponenten, welche die Kommunikation mit anderen Entitätstypen im Netzwerk bereitstellt und koordiniert. Die Softwarekomponente zur **An-/Abmeldung** im gemeinsamen Kommunikationsnetzwerk ist abhängig von der verwendeten Kommunikationsmethode. Eine **Kommunikationskomponente** vermittelt empfangene Dienstaufträge an die Prozesslogik-Gruppe. Die **Datenverarbeitungskomponente** ist ein Zwischenspeicher und speichert die von Dritten abrufbaren Daten wie z.B. historische Daten. Diese Daten sind für den eigentlichen Betrieb, d.h. die Prozessausführung, nicht notwendig oder können in der bestehenden Steuerung nicht gespeichert werden. Die Datenverarbeitungskomponente übernimmt die Aufgabe, Rohdaten entsprechend den definierten Nachrichtentypen vorzuverarbeiten.

Alle drei Bereiche kommunizieren direkt mit der **Statuskontrolle** in der **Prozesslogik-Gruppe**, welche den logischen Ablauf innerhalb einer informationstechnischen Schnittstelle koordiniert. Die Statuskontrolle aktualisiert die Softwarekomponenten der Rollen-Interface-Gruppe und löst Aktionen dort aus. Darüber hinaus koordiniert sie die Softwarekomponenten der Prozesslogik- und der Ressourcen-Interface-Gruppe. Hierzu gehört die **Prozesssteuerung**, welche über die Softwarekomponenten der Ressourcen-Interface-Gruppe den Fertigungsablauf steuert. Die **Prozessüberwachung** hingegen kontrolliert die ausgeführten Schritte innerhalb der Arbeitsstation. Das **Taktmodul** hat die Aufgabe, den gemeinsamen Takt der Fertigungslinie abzustimmen.

Die Prozesssteuerung und Prozessüberwachung sind die einzigen Softwarekomponenten, die auf die Ressourcen-Interface-Gruppe zugreifen. Die **Ressourcen-Interface-Gruppe** fasst die Adapter zusammen, die die direkte Interaktion mit den Ressourcen der Arbeitsstation vornehmen. Hierzu gehören die **Ansteuerung von Feldgeräten**, die **Ansteuerung untergeordneter Steuerungen** wie SPSen, CPSen und Robotersteuerungen oder die **An-**

steuerung für HMI. Diese Softwarekomponenten stellen die gerätespezifischen Treiber etc. bereit.

Durch die Unterteilung in Gruppen und Softwarekomponenten besteht eine hohe Wiederverwendbarkeit des Softwarecodes, wenn diese auf die unterschiedlichen Rollen mit ihren unterschiedlichen Hardware-Ausprägungen übertragen werden. Der Softwarecode der Rollen-Interface-Gruppe bleibt bei der Implementierung weitestgehend unverändert. Die Prozesslogik-Gruppe benötigt zur Rollen-Interface-Gruppe hin keine Anpassungen, muss aber die rollenspezifischen Softwarekomponenten der Ressourcen-Interface-Gruppe integrieren. Die Anbindung der Ressourcen-Interface-Gruppe hingegen muss an die Gegebenheiten der Rolle angepasst werden. Je nach verwendeter Hardware kann auch nur eine Auswahl der Gruppen realisiert werden.

Zusammenfassend beschreibt die Schnittstellenarchitektur abschließend, wie sich die informationstechnischen Schnittstellen realisieren lassen. Sie ergänzt die Systemarchitektur, welche die Struktur und Tätigkeiten beschreibt, und das Informationsmodell, welches Dienste und Nachrichten definiert. Zusammen bilden diese drei Teilergebnisse die Referenzarchitektur. Das nächste Kapitel widmet sich der prototypischen Realisierung der einheitlichen informationstechnischen Schnittstellen gemäß der Referenzarchitektur.

5. Prototypische Realisierung der Referenzarchitektur

Die im vorherigen Kapitel vorgestellte Referenzarchitektur beschreibt technologieunabhängig, welche Dienste die einzelnen Teilnehmer in der Produktion anbieten müssen, um die Methoden der Lean Production digital umzusetzen. Dieses Kapitel beschreibt die Verifikation der Referenzarchitektur, die nach [Clo+09] durch die prototypische Realisierung an einem industriellen Demonstrator in der *SmartFactory*^{KL} [Tec17b] vorgenommen wird. Ferner zeigt sie einen möglichen Weg, wie Nutzer die Referenzarchitektur in ihrer bestehenden Produktionsumgebung anwenden können. Kapitel 5.1 beschreibt das der Verifikation zugrunde liegende Anwendungsszenario und die Testumgebung, in welcher die Referenzarchitektur umgesetzt wurde. Kapitel 5.2 beschreibt anschließend die Realisierung.

5.1 Anwendungsszenarien und Rahmenbedingungen

Zur Demonstration des in Kapitel 3.2 beschriebenen Ziels wurden zwei Anwendungsszenarien definiert:

- 1) Technologische Wandelbarkeit für Lean-Automation-Anwendungen
- 2) Schnelle Wandelbarkeit von Fertigungslinien

Anwendungsszenario 1 demonstriert den mittels der einheitlichen informationstechnischen Schnittstellen ermöglichten und im Vergleich zu proprietären Realisierungen aufwandsarmen Austausch von Lean-Automation-Anwendungen. Abbildung 68 zeigt exemplarisch, wie unterschiedliche Technologien dieselbe Lean-Methode realisieren können, ohne Veränderungen an der informationstechnischen Schnittstelle vornehmen zu müssen. Die Auftragsfreigabe kann entweder als klassische, PC-basierte Anwendung ausgeführt werden oder durch eine moderne Version als Anwendung auf einem mobilen Tablet.

Anwendungsszenario 2 demonstriert die von der steigenden Produktindividualisierung und sinkenden Losgrößen geforderte stetige Anpassung von Fertigungslinien. Mithilfe der in der Referenzarchitektur definierten einheitlichen informationstechnischen Schnittstellen kann in der Praxis eine Arbeitsstation durch anders geartete Arbeitsstationen ausgetauscht werden,

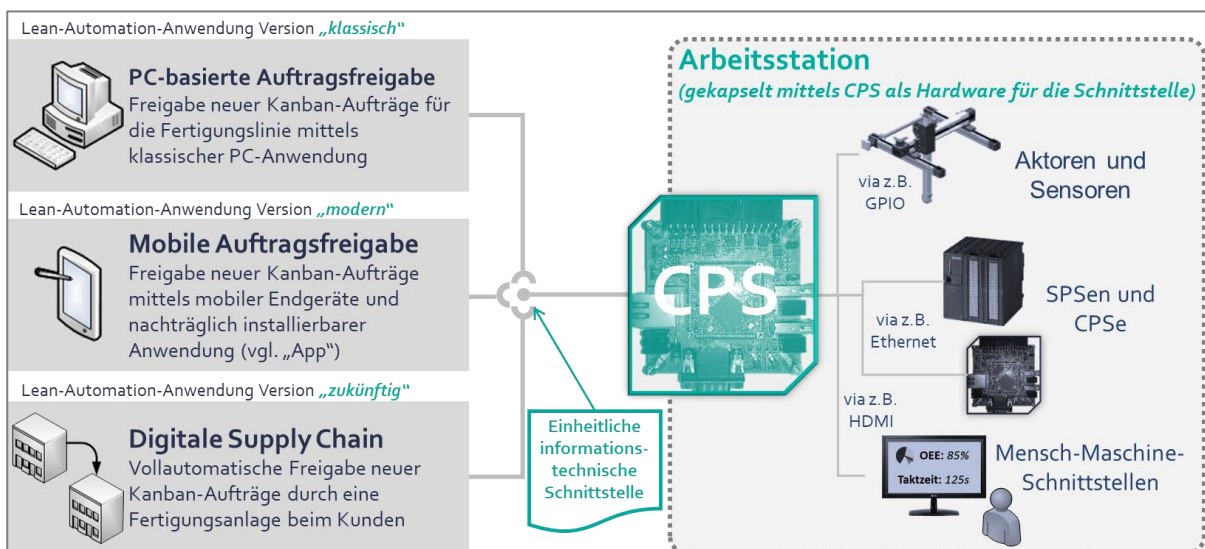


Abbildung 68 Anwendungsszenario 1: Beispiel für die technologische Unabhängigkeit der Lean-Automation-Anwendung

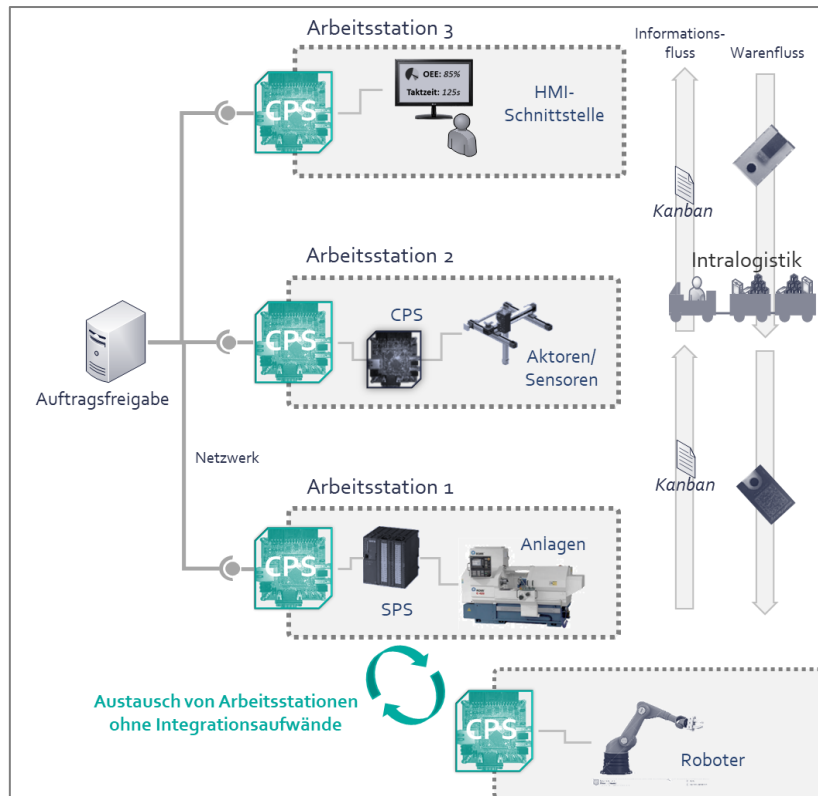


Abbildung 69 Beispiel für den Austausch von Arbeitsstationen ohne notwendige Anpassungen der Umwelt

ohne dass sich die Umgebung außerhalb der Arbeitsstation anpassen muss (siehe Abbildung 69). In der prototypischen Realisierung ersetzt ein Robotermodul einen bestehenden manuellen Arbeitsschritt im Fertigungsprozess. Dank der einheitlichen informationstechnischen Schnittstelle ist das Robotermodul sofort lauffähig.

Neben diesen zwei aus der übergeordneten Zielsetzung der Arbeit abgeleiteten Anwendungsszenarien demonstriert die prototypische Realisierung, wie die Referenzarchitektur auf eine bestehende Fertigungslinie zu übertragen ist. Hierzu gehören die Modellierung mittels der in der Systemarchitektur beschriebenen Rollen und deren Ressourcen, die Realisierung des Informationsmodells und die Entwicklung der informationstechnischen Schnittstellen mithilfe der Schnittstellenarchitektur.

Die Implementierung der informationstechnischen Schnittstellen ist für die einzelnen Lean-Methoden grundsätzlich ähnlich. Aufgrund der Vielzahl an Lean-Methoden und des damit verbunden hohen Aufwands für die Implementierung umfasst die prototypische Realisierung nur einen Ausschnitt. Hierzu gehören die Lean-Methoden Kanban, Just in Time, Bottleneck Analyse, One Piece Flow, FiFo und Nivellierung.

Die Umgebung für die prototypische Realisierung ist der bestehende Demonstrator „SkalA“ der Technologie-Initiative SmartFactory KL e.V. Er eignet sich für die Realisierung aufgrund des einer Fertigungslinie ähnlichen Prozesses mit entkoppelter Logistik, dem Einsatz von CPSen zur Kapselung von Diensten sowie der Einbindung des Werkers und von Automatisierungstechnik. Tabelle 27 fasst die Rahmenbedingungen und das Ziel der prototypischen Realisierung zusammen.

Tabelle 27 Steckbrief der prototypischen Realisierung

Ziel der Verifikation	<ul style="list-style-type: none"> • Demonstration der Anwendbarkeit der Referenzarchitektur an einer bestehenden Fertigungslinie • Demonstration der verbesserten Wandelbarkeit einer Lean-Production-Fertigungslinie
Anwendungsszenarien	<ul style="list-style-type: none"> • Technologischer Wandelbarkeit von Lean-Automation-Anwendungen • Rekonfiguration einer Fertigungslinie durch Austausch einer Arbeitsstation
Umgesetzte Elemente der Referenzarchitektur	<ul style="list-style-type: none"> • Systemarchitektur • Informationsmodell • Schnittstellenarchitektur
Umgesetzte Lean-Methoden	<ul style="list-style-type: none"> • Kanban • Just in Time • Bottleneck Analyse • One Piece Flow • FiFo • Nivellierung
Demonstrationsobjekt	<ul style="list-style-type: none"> • Bestehender Demonstrator „SkalA“ der <i>SmartFactory</i>^{KL} zur Herstellung eines Visitenkartenhalters mittels dezentral organisierter CPSe

Der Demonstrator „SkalA“ montiert in insgesamt sechs Schritten einen dreiteiligen Visitenkartenhalter. Nach dem Beschreiben des RFID-Tags auf der Grundplatte des Visitenkartenhalters und dem Bestücken des Werkstückträgers durch wahlweise den Werker oder ein separates Robotermodul wird die Grundplatte auf dem Werkstückträger zentriert und anschließend eine Halteklammer eingesetzt. Ein aufgeschobener Deckel wird dann mit der Grundplatte verpresst. Der Werker prüft abschließend manuell die Produktqualität. Abbildung 70 zeigt die Arbeitsschritte und die dazwischen stattfindenden Transporte des Produktes.

In dem Demonstrator sind mehrere CPSe installiert. Es handelt sich hierbei um einen auf dem Aria G25 basierenden Einplatinenrechner mit 400 Mhz ARM-Prozessor, 256 MB Arbeitsspeicher und Linux als Betriebssystem (siehe [Acm]). Die angeschlossenen Sensoren und Aktoren wie beispielsweise Lichtschranken, Endschalter, Ventilinseln für Pneumatikzylinder oder MODBUS-basierte Motorcontroller für die Linearachsen werden von den CPSen über die digitalen Ein- und Ausgänge angesteuert. Die CPSe verfügen darüber hinaus über eine Ethernet-Schnittstelle und sind alle mit einem gemeinsamen TCP/IP-basierten Netzwerk

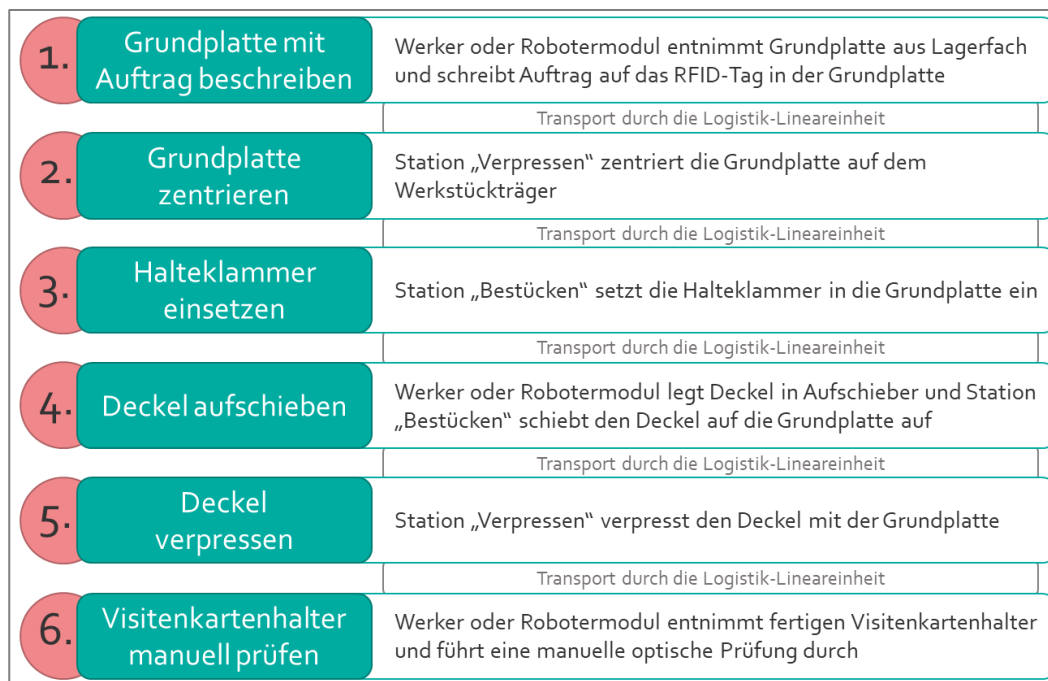


Abbildung 70 Arbeitsschritte für den Visitenkartenhalter am Demonstrator

verbunden. Ein CPS kapselt zusätzlich das Robotermodul. Der Werker kann über ein Tablet und einen Projektor als HMI mit den CPSen interagieren. Zusätzlich sind in dem Demonstrator zwei PCs installiert, welche den Projektor ansteuern und zur Visualisierung von Abläufen und Informationen eingesetzt werden.

Bei den vier eingesetzten CPSen handelt es sich um das CPS-Logistiksystem, das ausschließlich den Transport des Werkstücks mittels einer Linearachse kontrolliert. Die CPS-Station „Bestücken“ steuert das pneumatische Einsetzen der Halteklammer aus einem an einer Linearachse montiertem Magazin und das pneumatische Aufschieben des Deckels. Die CPS-Station „Verpressen“ zentriert die Grundplatte und verpresst den Deckel mit zwei von ihr angesteuerten Linearachsen. Das vierte CPS ist an einem separaten Robotermodul installiert und ruft durch die im Netzwerk kommunizierten Befehle entsprechende Funktionen am KUKA-Roboter aus. Das Robotermodul ist redundant zu den manuellen Schritten des Werkers und kann optional zugeschaltet werden. Abbildung 71 zeigt den Aufbau des Grundmoduls ohne Robotermodul mit der Verteilung der CPSen sowie die Bestandteile des gefertigten Visitenkartenhalters. Abbildung 72 zeigt das Grundmodul und das separate Robotermodul.

Obwohl der Demonstrator mit seiner geringen Baugröße und den in den Arbeitsschritten mehrfach verwendeten Arbeitsstationen nicht wie eine klassische Fertigungslinie der Lean Production aussieht, lässt sich der Prozess hierauf übertragen. Die Montage ist ein sequenziell ablaufender Prozess, bei welchem ein Logistiksystem ähnlich wie beim Milkrun den Transport zwischen den Arbeitsstationen übernimmt.

Die CPS-Stationen und je nach Konfiguration die manuelle Station oder das Robotermodul führen zwei Schritte des zuvor beschriebenen Montageprozesses aus. Jede Station und das Robotermodul bieten daher zwei informationstechnische Schnittstellen an und repräsentieren so zwei separate Arbeitsstationen einer Fertigungslinie. Beispielsweise ist die physische CPS-Station „Verpressen“ eine logische Arbeitsstation zum Zentrieren der Platte (Schritt 1)

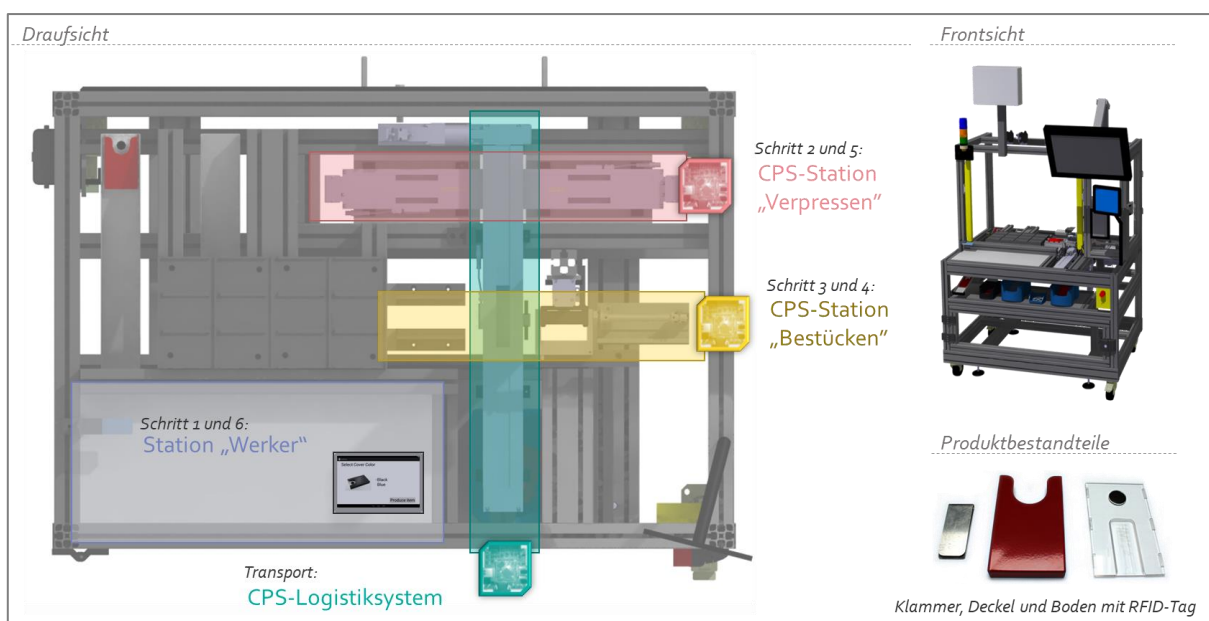


Abbildung 71 Aufbau des Demonstrators und des Produktes (Quelle: eigene Darstellung basierend auf Technologie-Initiative SmartFactory KL e. V.)

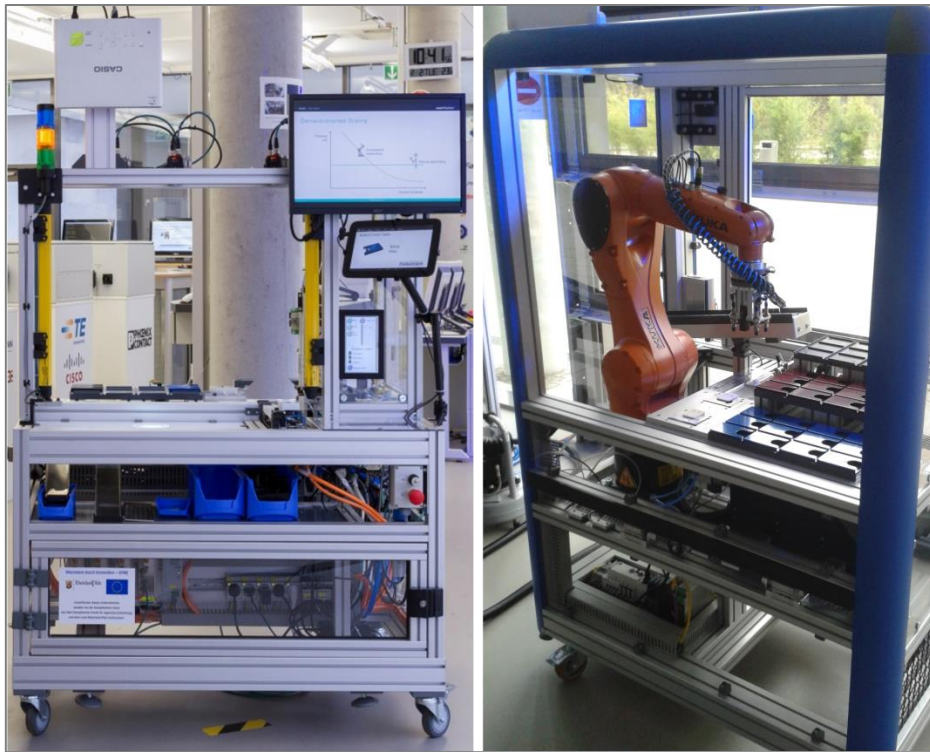


Abbildung 72 Demonstrator „SkalA“ (links) und das optionale Robotermodul als Alternative zum manuellen Arbeitsschritt (rechts) (Quelle: Technologie-Initiative SmartFactory KL e.V.)

und eine logische Arbeitsstation zum Verpressen der Platte (Schritt 4). Die Linearachse am CPS-Logistiksystem übernimmt die Funktion des Milkruns. Auf einem zusätzlich am Demonstrator verbauten PC sind die Auftragsfreigabe und das KVP-System installiert. Das bereits vorhandene Tablet wird als HMI für den Werker an der ersten Arbeitsstation verwendet. In dieser Konstellation sind fast alle Rollen der Systemarchitektur in einem sequenziell laufenden Fertigungsprozess, ähnlich wie in einer Fertigungslinie, vorhanden und kommunizieren in einem gemeinsamen Kommunikationsnetzwerk miteinander (siehe Abbildung 73). Die Rolle Supermarkt ist mangels einer zuliefernden Fertigungslinie in dieser Konstellation nicht vorgesehen. Die Rolle des Hancho bzw. ein HMI für ihn wurde ebenfalls nicht realisiert, allerdings die von den Arbeitsstationen für ihn ausgeworfenen Nachrichten. Diese Topologie ist die Grundlage für die im nächsten Kapitel vorgestellte Implementierung der informationstechnischen Schnittstellen.

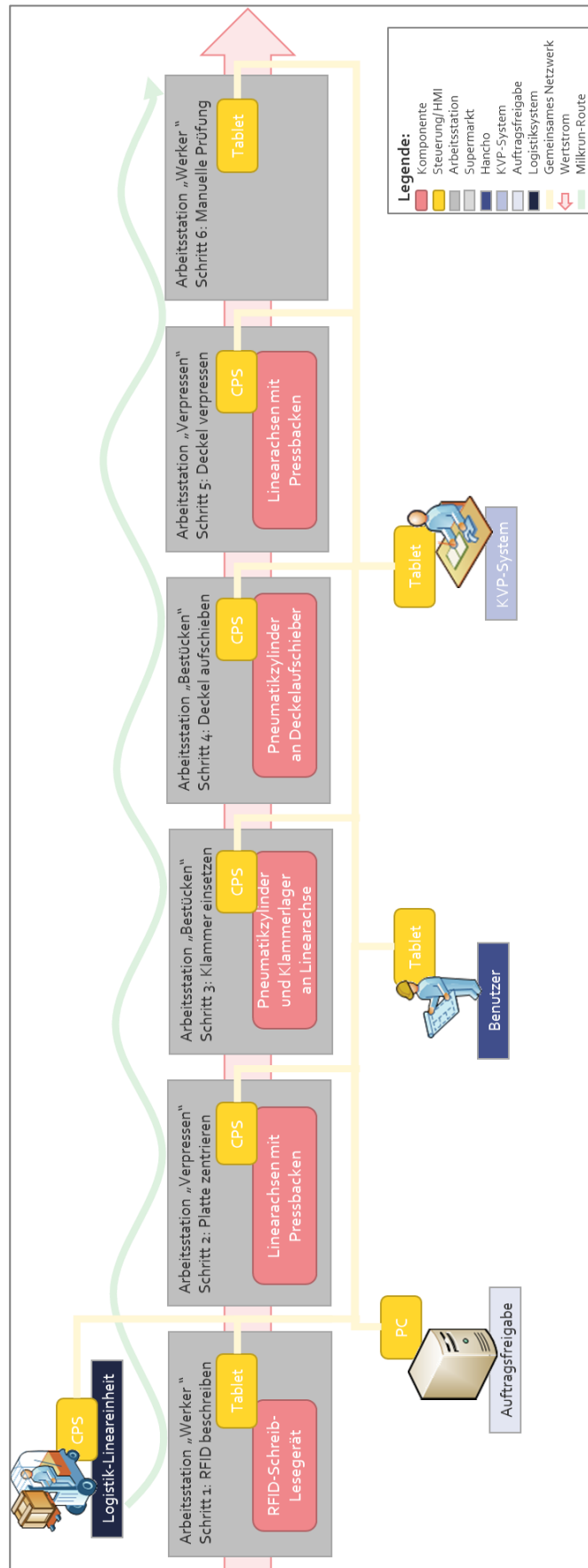


Abbildung 73 Übertragung des Montageprozesses am Demonstrator auf eine Lean-Production-Fertigungslinie

5.2 Technische Realisierung

Mithilfe der übergeordneten Systemarchitektur wurde die Fertigungslinie des Demonstrators „SkalA“ modelliert. Die modellierte Systemarchitektur legt die Verteilung der informationstechnischen Schnittstellen und die durch die Schnittstelle anzusteuern Ressourcen für die Implementierung fest.

Zwei unterschiedlich realisierte Auftragsfreigaben demonstrieren den technologischen Austausch von Lean-Automation-Lösungen (Anwendungsszenario 1). Die eine Version ist eine moderne Android-Applikation auf einem Tablet, und die andere Version ist eine herkömmliche Java-basierte Anwendung für den PC. Die jeweils bereitgestellte informationstechnische Schnittstelle ist von der Rolle „Auftragsfreigabe“ abgeleitet. Das KVP-System ist auf einem PC installiert, welcher eine informationstechnische Schnittstelle der Rolle „KVP-System“ bereitstellt.

Die Station „Werker“ besteht aus einem Tablet und einem PC mit Projektor. Da die Station zwei Schritte übernimmt und somit zwei Arbeitsstationen repräsentiert, bietet sie die zwei informationstechnischen Schnittstellen „Boden einlegen“ und „Produkt entnehmen“ an. Zur Demonstration der flexiblen Anpassung einer Fertigungslinie (Anwendungsszenario 2) bietet das CPS am Robotermodul die gleichen Schnittstellen an.

Das CPS an der Station „Verpressen“ stellt die informationstechnischen Schnittstellen „Boden zentrieren“ und „Deckel verpressen“ bereit. Es steuert direkt die Motorcontroller der zugehörigen Linearachsen. Die Schnittstellen „Klammer einschieben“ und „Deckel einschieben“ stellt die CPS-Station „Bestücken“ bereit. Es steuert ebenfalls direkt mehrere Aktoren und Sensoren sowie einen Motorcontroller. Das CPS der Logistik-Lineareinheit stellt eine informationstechnische Schnittstelle der Rolle Logistiksystem bereit. Ferner steuert es direkt den Motorcontroller der dazugehörigen Linear-achse, die dort angebrachten Endschalter, eine Lichtschranke, eine Ventilinsel und einen Vakuumerzeuger.

Abbildung 74 zeigt das mithilfe der Systemarchitektur erstellte Modell des Demonstrators. Insgesamt sind zwölf informationstechnische Schnittstellen zu realisieren. Zwei Schnittstellen steuern direkt Feldgeräte an, zwei Schnittstellen kapseln bestehende Steuerungen von Feldgeräten und vier Schnittstellen dienen der Interaktion mit dem Menschen und steuern keine Feldgeräte oder untergeordneten Steuerungen an.

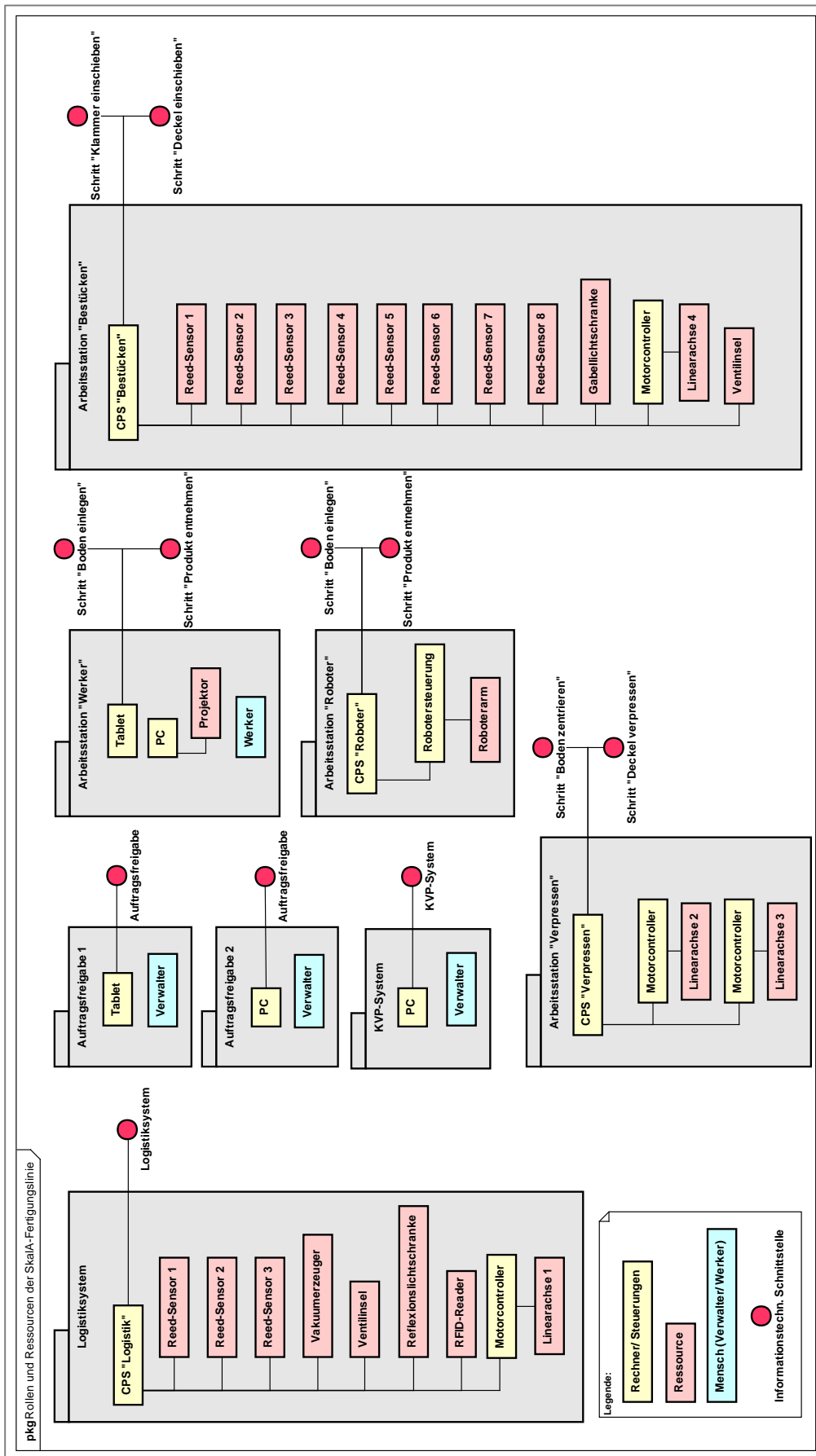


Abbildung 74 Informationstechnische Schnittstellen, Rollen und Ressourcen der Fertigungsline am Demonstrator „Skala“

Für die Kommunikation zwischen den Entitäten wurde das in Kapitel 2.1.3.3 beschriebene Kommunikationsprotokoll MQTT verwendet. Aufgrund der schlanken Lösung, guten Dokumentation und freien verfügbaren Software ist es ein pragmatischer Ansatz für die prototypische Realisierung. Ferner kann mit MQTT die Publish-Subscribe-Methode umgesetzt werden, welche als Weiterentwicklung von SoA in heterogenen, sich ändernden Umgebungen zukünftig ein vielversprechender Ansatz ist. Auch OPC UA verfolgt ähnliche Bestrebungen, die sich zum Zeitpunkt der Realisierung allerdings noch in der Spezifikation befanden.

Der MQTT-Broker, der im Netzwerk als zentrale Vermittlungsstelle die von Clients abonnierten Nachrichten verwaltet und Nachrichten zwischen diesen vermittelt, wurde mithilfe des frei verfügbaren Eclipse Mosquitto (siehe [The]) auf einem separaten PC im Demonstrator bereitgestellt. MQTT bietet bei der Publish-Subscribe-Methode die Möglichkeit, auch nicht explizit abonnierte Nachrichten den richtigen Clients zuzustellen. Hierzu verwendet MQTT bei den in den Datenpaketen enthaltenen Topics Hierarchien, welche ähnlich wie die auf PCs bekannten Ordnerhierarchien durch einen Schrägstrich („/“) gebildet werden können und der Gruppierung von Themen dienen. Mittels Platzhaltern („#“ und „+“) können Clients des Weiteren Topics abonnieren, deren Name ihnen vorab nicht bekannt ist.

Für die Implementierung der Topics wurde eine Hierarchie verwendet, welche sich in Fertigungslinien, Rollen und konkrete Entitäten untergliedert. Die im Informationsmodell spezifizierten Dienste sind diesen Entitäten untergeordnet, d.h. zu jedem von einer Entität anzubietenden Dienst existiert ein eigenes Topic, das der Dienstanbieter beim MQTT-Broker abonniert. Die Topic-Hierarchie folgt dem Schema:

[Produktionsstandort] / [Fertigungslinie] / [Rolle] / [Entität] / [Dienstname]

Darüber hinaus existiert ein Topic, mit dem Arbeitsstationen gemeinsam angesprochen werden können (sog. Broadcasten). Dieser Dienst wird u.a. von der Rolle KVP-System verwendet, um gleichzeitig Informationen von allen Arbeitsstationen abzurufen. Das Topic ist nicht zwingend notwendig, sondern dient lediglich der Vereinfachung, damit nicht jede Arbeitsstation einzeln angesprochen werden muss. Abweichend von dem vorherigen Schema ist der für die Andon-Reißleine bei allen Rollen notwendige Dienst „empfangelinienstopp“ zum Anhalten und Fortsetzen der gesamten Fertigungslinie in der Hierarchie nicht den Entitäten untergeordnet, sondern der Fertigungslinie. Abbildung 75 zeigt exemplarisch den Ablauf eines Dienstaufwurfes, Tabelle 28 gibt einen Überblick über die von den Rollen abonnierten Topics.

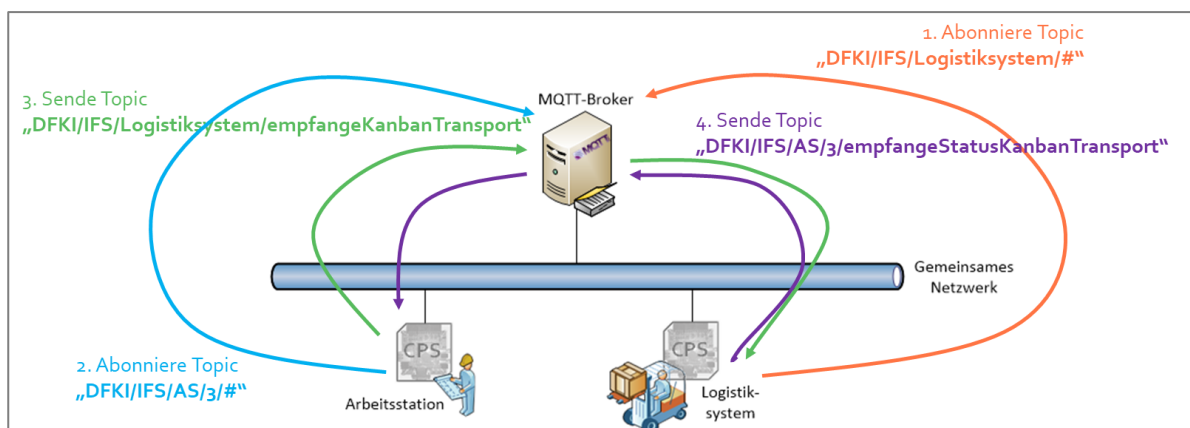


Abbildung 75 Beispiel für die Kommunikation der Methode Kanban mittels MQTT

Tabelle 28 Übersicht der in der prototypischen Realisierung je Rolle abonnierten Topics („#“ = MQTT-Platzhalter für alle untergeordneten Topics) (Quelle: [Kno17])

Rolle der informationstechnischen Schnittstelle	Abonnierte Topic
Auftragsfreigabe	DFKI/IFS/Auftragsfreigabe/# DFKI/IFS/Produktionsstopp/empfangelinienstopp DFKI/IFS/Produktionsstopp/empfangelinienstoppaktualisierung
Arbeitsstation	DFKI/IFS/AS/[Arbeitsstation-ID]/# DFKI/IFS/AS/Broadcast/# DFKI/IFS/Produktionsstopp/empfangelinienstopp DFKI/IFS/Produktionsstopp/empfangelinienstoppaktualisierung
Logistiksystem	DFKI/IFS/Logistiksystem/# DFKI/IFS/Produktionsstopp/empfangelinienstopp DFKI/IFS/Produktionsstopp/empfangelinienstoppaktualisierung
Hancho	DFKI/IFS/Hancho/# DFKI/IFS/Produktionsstopp/empfangelinienstopp DFKI/IFS/Produktionsstopp/empfangelinienstoppaktualisierung
KVP-System	DFKI/IFS/KVPSystem/# DFKI/IFS/Produktionsstopp/empfangelinienstopp DFKI/IFS/Produktionsstopp/empfangelinienstoppaktualisierung
Supermarkt	DFKI/IFS/Supermarkt/# DFKI/IFS/Produktionsstopp/empfangelinienstopp DFKI/IFS/Produktionsstopp/empfangelinienstoppaktualisierung

Neben den MQTT-Topics enthalten Datenpakete einen Payload. Dieser enthält in der technischen Realisierung die Attribute der ausgetauschten Nachrichtentypen und deren Ausprägung. Als Beschreibungssprache wurde XML verwendet, da XML eine semantische Beschreibung der Informationen ermöglicht. Die semantische Beschreibung von Dateninhalten ist im Gegensatz zu proprietären Ansätzen nicht nur maschinenlesbar, sondern auch von Menschen lesbar, was die Wandelbarkeit unterstützt. Die in XML vorgesehene Vergabe von Namensräumen für Schemata ermöglicht es ferner, unterschiedliche Versionen von Nachrichtentypen umzusetzen, sodass spätere Aktualisierungen oder individuelle Anpassungen durch den Nutzer nicht zu Konflikten bei der Interpretation führen. Abbildung 76 zeigt am Beispiel der Nachricht „KanbanFertigung“ die Realisierung der Attribute im Payload. Die optionalen Attribute „Lieferzeit“ und „Taktzeit“ zeigen hierbei die Möglichkeit, das Informationsmodell der Referenzarchitektur nachträglich nach eigenen Bedürfnissen zu erweitern.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  <KanbanFertigung xmlns="http://www.DFKI.de/LeanSchema_v1.0">
    <Losgröße>1</Losgröße>
    <KanbanID>1</KanbanID>
    <AdresseQuelle>Arbeitsstation 1</AdresseQuelle>
    <AdresseSenke>Arbeitsstation 3</AdresseSenke>
    <ProduktID>2</ProduktID>
    <ProduktName>M8 Schraube</ProduktName>
    <WerkstückträgerTyp>Träger 7</WerkstückträgerTyp>
    <Lieferzeit>
      <Wert>2017-7-4-14-48-44</Wert>
      <Einheit>y-m-d-h-min-sec</Einheit>
    </Lieferzeit>
    <Taktzeit>
      <Wert>90</Wert>
      <Einheit>sec</Einheit>
    </Taktzeit>
    <Gültigkeit>true</Gültigkeit>
  </KanbanFertigung>
```

Abbildung 76 Beispiel für die XML-basierte Beschreibung des Dateninhalts der Nachricht „KanbanFertigung“ (Quelle: [Kno17])

Wie in Abbildung 77 und Abbildung 78 dargestellt, wurde die informationstechnischen Schnittstellen unter Verwendung der in Kapitel 4.3.3 beschriebenen Architektur realisiert. Als Programmiersprache kam Java zum Einsatz. Die Schnittstelle kapselt die bereits bestehende Ansteuerung der Feldgeräte und die bestehende proprietäre Kommunikation zur Ansteuerung der Motorcontroller. Besonderheiten bei der Implementierung ergeben sich bei der Umsetzung des KVP-Systems und der Auftragsfreigabe, welche als reine Software-Systeme keine Feldgeräte ansteuern und somit aus der Schnittstellenarchitektur nur die Softwarekomponenten der Rollen-Interface- und Prozesslogik-Gruppe realisieren.

Als Bestandteil der Softwarekomponente „Ansteuerung HMI“ der Ressourcen-Interface-Gruppe verfügt jede Arbeitsstation über eine eigene Webseite, über welche sich u.a. die verschickbaren und empfangbaren Kanban sowie der aktuell vorliegende Auftrag für den Mitarbeiter der Arbeitsstation einsehen lässt (siehe Abbildung 79). Die Benutzungsoberflächen der Auftragsfreigabe und des KVP-Systems wurden ebenfalls als Java-Applikation umgesetzt.

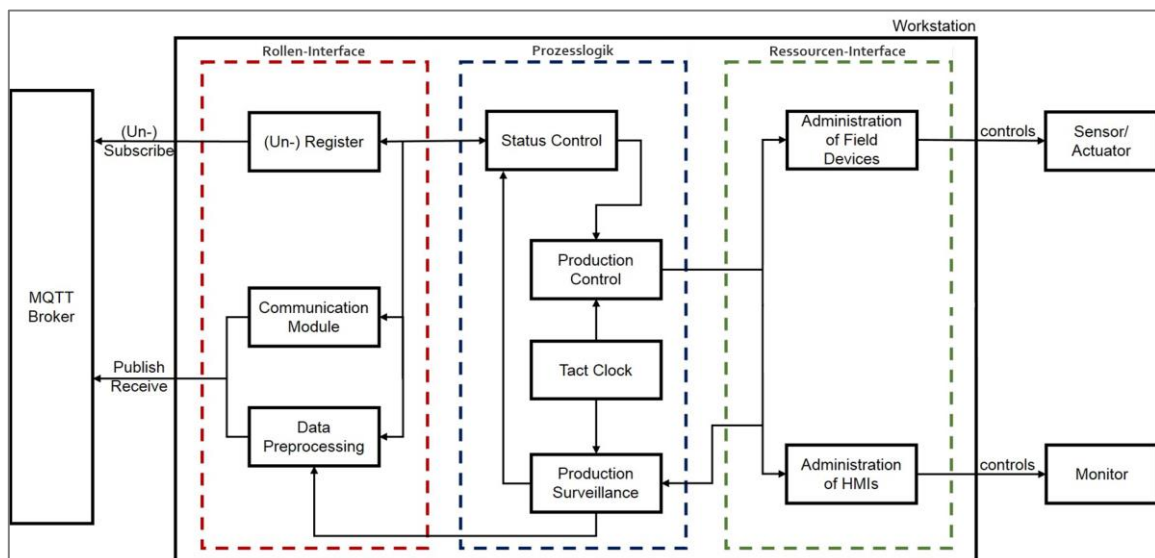


Abbildung 77 Realisierte Architektur der informationstechnischen Schnittstelle für die Arbeitsstationen „Bestücken“ und „Verpressen“ (Quelle: [Kno17])

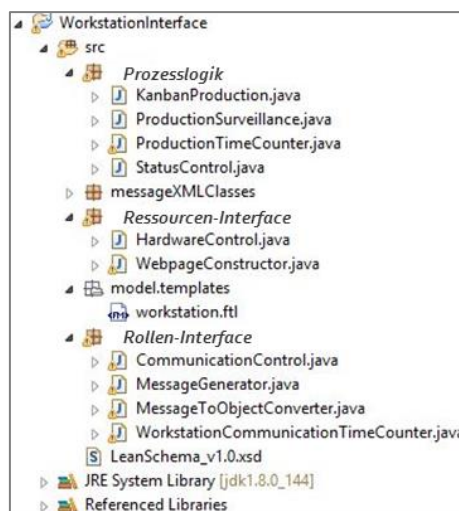


Abbildung 78 Struktur des Softwarecodes für die Schnittstelle an den Arbeitsstationen (Quelle: [Kno17])

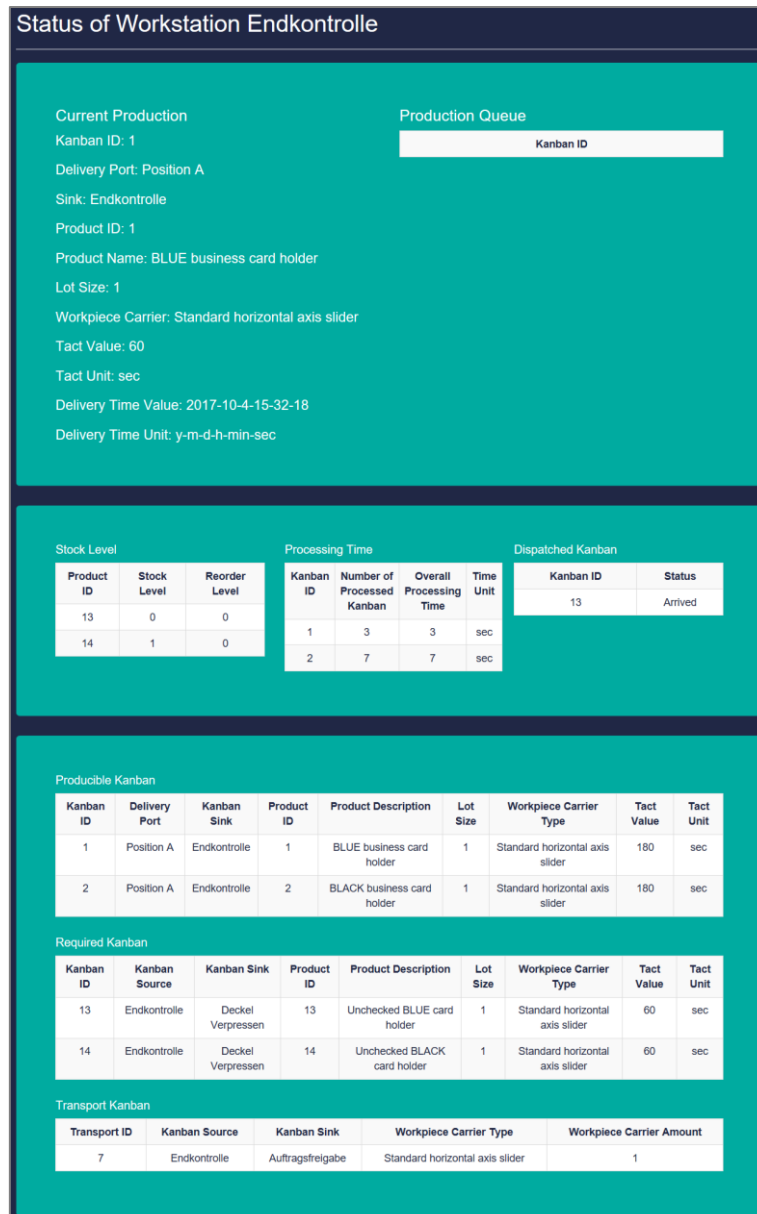


Abbildung 79 Beispiel einer Webseite von einer Arbeitsstation (Quelle: [Kno17])

Auf Basis dieser Implementierung können die eingangs beschriebenen Anwendungsszenarien demonstriert werden. Abbildung 80 zeigt die für Anwendungsszenario 1 realisierte Java-Applikation für den PC (links) und die Android-Anwendung am Tablet (rechts). Bei der Implementierung musste lediglich die Spezifikation der Nachrichten sowie das verwendete MQTT-Protokoll berücksichtigt werden. Ein Eingriff in die Arbeitsstationen selbst oder ein Reengineering bestehender Implementierungen war nicht notwendig. Der Aufwand für die Realisierung der neuen Oberfläche reduzierte sich somit auf die Entwicklung der Oberfläche. Obwohl andere Software- als auch Hardware-Technologien (PC und Tablet) zum Einsatz kommen, können beide Lösungen auch während der Laufzeit ohne Rekonfiguration substituiert werden.

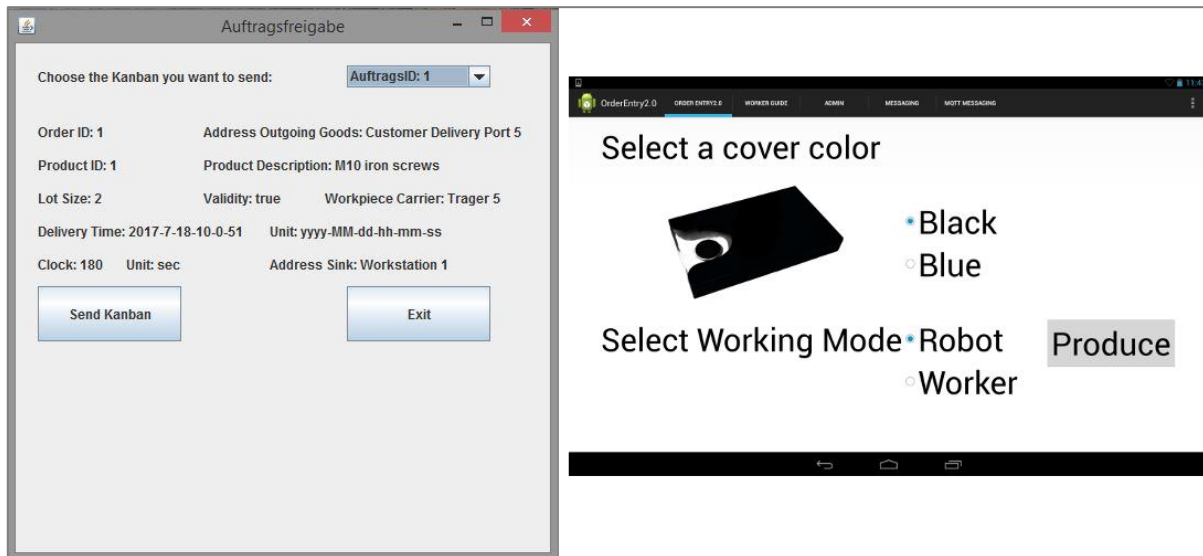


Abbildung 80 Auftragsfreigabe als PC-Anwendung (links) und Tablet-Anwendung (rechts)
(Quelle: [Kno17])

Um das Anwendungsszenario 2 zu demonstrieren, wurde die informationstechnische Schnittstelle auch auf dem CPS des Robotermoduls implementiert. Das Robotermodul substituiert die Schritte „RFID beschreiben“ und „Qualitätsprüfung“, die zuvor vom Werker durchgeführt wurden. Nach einem Tausch einer Arbeitsstation ist eine Aktualisierung der Quell- und Zieladressen der Kanban notwendig. Im Anwendungsszenario, in dem das Robotermodul zwei Arbeitsstationen substituiert, sind fünf Nachrichten auszutauschen:

1. Topic „DFKI/IFS/Auftragsfreigabe/aktualisiereKanban“:

Aktualisiert bei der Auftragsfreigabe die Zieladresse für neue Aufträge, da sich die Arbeitsstation für den letzten Schritt „Qualitätsprüfung“ der Fertigungslinie geändert hat.

2. Topic „DFKI/IFS/AS/5/aktualisiereKanban“:

Aktualisiert bei der Arbeitsstation „Verpressen“ die Zieladresse für neue Vorprodukte, da sich die Arbeitsstation für den Schritt „RFID beschreiben“ der Fertigungslinie geändert hat.

3. Topic „DFKI/IFS/AS/99/aktualisiereKanban“:

Teilt dem Robotermodul mit, welche Kanban sie zukünftig empfangen kann und an wen sie Kanban schicken muss, wenn sie Nachschub benötigt. Dies umfasst sowohl den Schritt „Qualitätsprüfung“ als auch den Schritt „RFID beschreiben“, da das Robotermodul mehrfachverwendet wird.

4. Topic „DFKI/IFS/AS/2/aktualisiereKanban“:

Teilt der Arbeitsstation „Verpressen“ mit, dass sie zukünftig Kanban von einer anderen, nachgelagerten Arbeitsstation erhält.

5. Topic „DFKI/IFS/Logistiksystem/aktualisiereKanban“:

Teilt dem Logistiksystem die aktualisierten Quell- und Zieladressen der Transport-Kanban mit, die sich durch den Austausch geändert haben.

Die Nachrichten können mit jedem beliebigen System verschickt werden, welches mittels MQTT und dem XML-Schema konforme Nachrichten verschicken kann. Obwohl unterschiedliche Technologien an den Arbeitsstationen zum Einsatz kommen, reduziert die informations-

technische Schnittstelle den Aufwand für die Rekonfiguration auf wenige, standardisierte Nachrichten. Die Anmeldung und Abmeldung am MQTT-Broker erfolgt bei der prototypischen Realisierung automatisch. Abbildung 81 zeigt einen Auszug der für die Auftragsabwicklung mit dem Robotermodul über den MQTT-Broker ausgetauschten Nachrichten.

Neben den Anwendungsszenarien wurden auch die eingangs genannten Lean-Methoden realisiert. Im Gegensatz zur zuvor im Demonstrator realisierten proprietären, push-orientierten Kommunikation erfolgt die Kommunikation nun nachfrageorientiert gemäß der Kanban-Methode (siehe Abbildung 82). Die Auftragsfreigabe am PC oder am Tablet gibt einen neuen Auftrag für die letzte Arbeitsstation „Manuelle Prüfung“ frei. Da diese kein fertiges Produkt bereithält, sendet sie unter Verwendung des Dienstes „empfangenKanbanFertigung“ eine Kanban an den vorgelagerten Schritt „Deckel verpressen“. Auch diese Arbeitsstation hat kein passendes Produkt, sodass sie mittels desselben Dienstes eine Kanban an den wiederum vorgelagerten Schritt „Deckel aufschieben“ versendet. Da im verwendeten Demonstrator „Skala“ keine Pufferbestände an den Arbeitsstationen existieren, sendet jede Arbeitsstation dem Vorgänger eine Kanban zu. Die Kommunikation erfolgt im Demonstrator nach einem rekursiven Ansatz. Erreicht die Kanban den ersten Schritt „RFID beschreiben“, stellt der Werker einen Boden bereit und quittiert den Arbeitsschritt. Die dazugehörige informationstechnische Schnittstelle informiert mittels Transportkanban das Logistiksystem, und das Produkt wird zur nächsten Arbeitsstation befördert. Ferner meldet es dem Kanban-Versender, dass die Kanban bearbeitet wurde. Durch das verwendete MQTT-Protokoll werden nicht nur geänderte Status einer Kanban kommuniziert, sondern auch der Empfang einer Kanban. Sofern eine Arbeitsstation eine Kanban nicht binnen der Soll-Zeit bearbeitet, sendet sie einen Fehler an den Hancho.

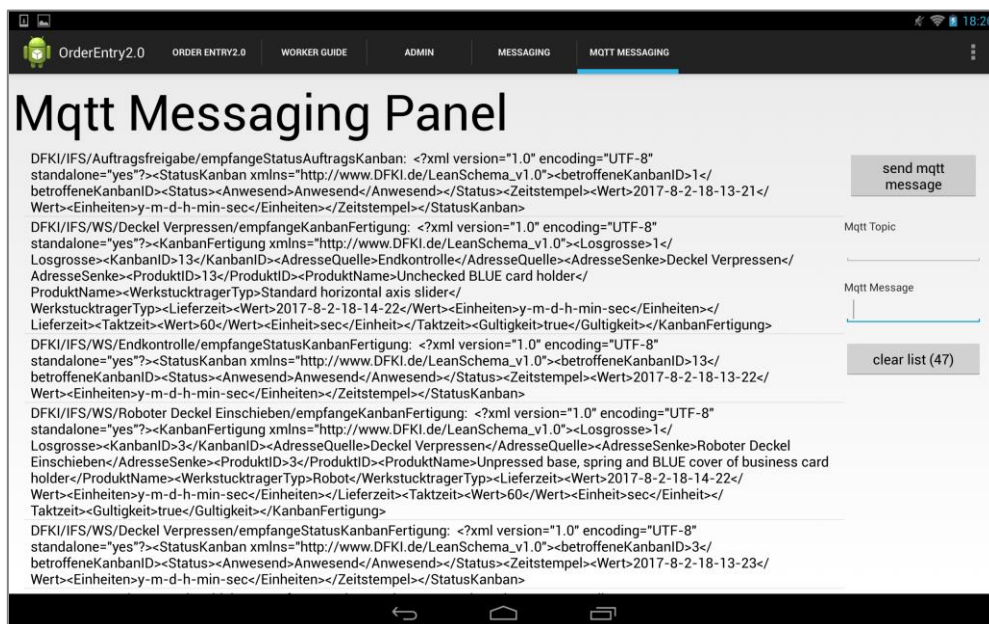


Abbildung 81 Auszug der am MQTT-Broker zur Auftragsbearbeitung mit dem Robotermodul stattfindenden Kommunikation

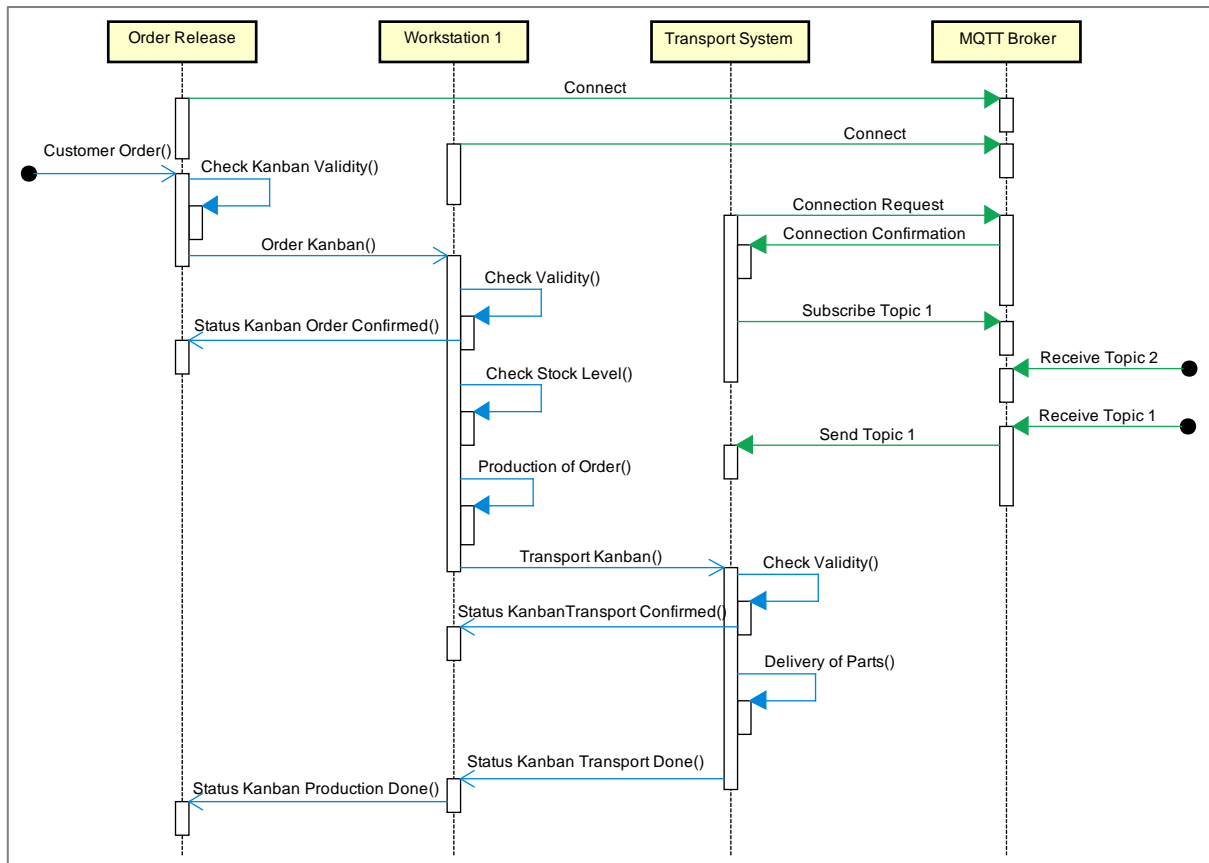


Abbildung 82 Nachfrageorientierte Kommunikation mittels Kanban im Demonstrator
(Quelle: [Kno17])

Für eine Bottleneck-Analyse kann das KVP-System alle bisher an den CPS-Arbeitsstationen gefertigten Kanban mit deren Ist-Bearbeitungszeit abrufen. Im Anschluss ermittelt das KVP-System die Arbeitsstation mit der längsten Durchlaufzeit. Diese Arbeitsstation stellt den Bottleneck dar und sollte optimiert werden, um die Durchlaufzeit weiter zu reduzieren. Die Auswertung erfolgt mit der in Abbildung 83 gezeigten Benutzungsoberfläche des KVP-Systems. Der Anwender kann hierrüber auch auf Basis der bei der Auftragsfreigabe vorliegenden Aufträge die Produktion nivellieren, d.h. für einen definierten Zeitraum werden die tatsächlichen Aufträge ermittelt und auf eine Jahresproduktionsmenge hochgerechnet. Das KVP-System berechnet mit diesem Wert die tägliche Soll-Ausbringungsmenge und -Taktzeit, die z.B. die Andon-Methode nutzt.

Die FiFo-Methode wurde am Demonstrator in der Auftragsfreigabe realisiert, die immer nur einen Auftrag für die Fertigung anhand des Eingangsdatums freigibt. Durch die fehlenden Pufferbestände und die bereits vorhandene Einzelstückfertigung konnte auch der One Piece Flow problemlos realisiert werden.

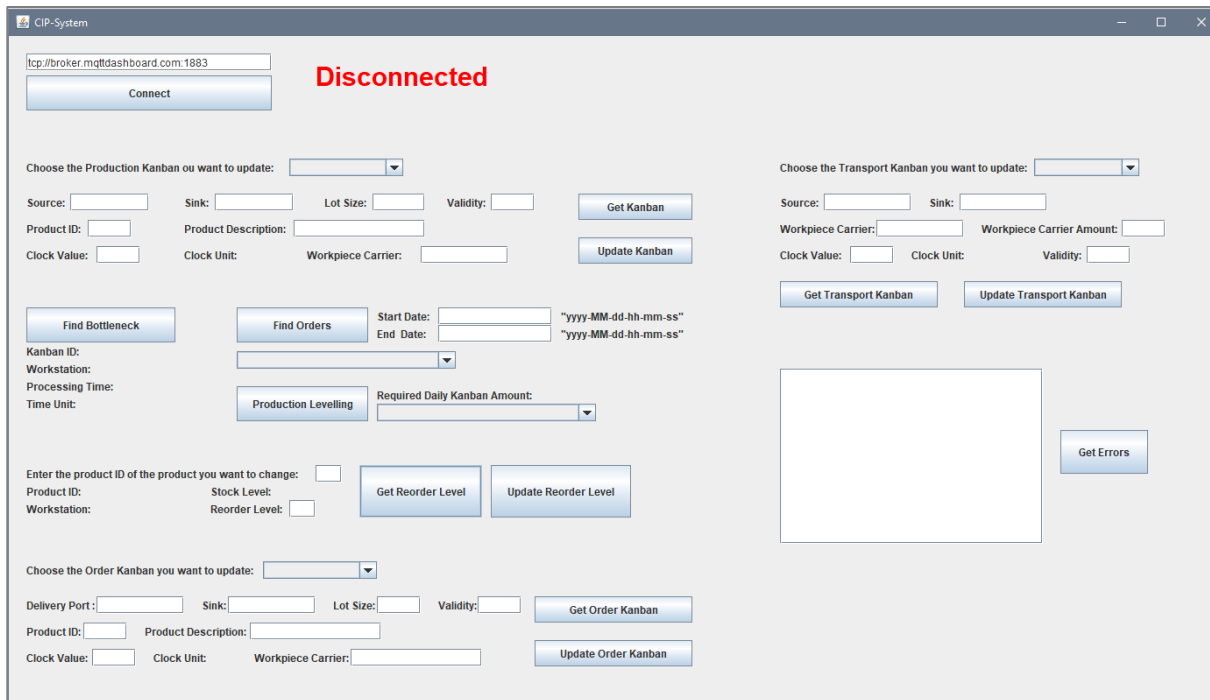


Abbildung 83 Benutzeroberfläche des KVP-Systems (Quelle: [Kno17])

Die prototypische Realisierung zeigte, wie sich die Referenzarchitektur auf eine bestehende Fertigungslinie übertragen lässt. Die in der Realität vorhandenen und relevanten Betriebsmittel konnten mit der Systemarchitektur modelliert werden, was die Identifikation von Rollen und den notwendigen, zu implementierenden informationstechnischen Schnittstellen ermöglichte. Unter Verwendung des leichtgewichtigen MQTT-Kommunikationsprotokolls und XML lassen sich die im Informationsmodell definierten Dienste und ausgetauschten Nachrichten realisieren. Die Schnittstellenarchitektur unterstützte dabei die Softwareentwicklung auf den CPSen.

Die Anwendungsszenarien zeigten, dass sich mithilfe der informationstechnischen Schnittstellen neue Lean-Automation-Anwendungen ohne Anpassungen der Umgebung integrieren lassen. Auch lassen sich Fertigungslinien der Lean Production mit wenig Aufwand rekonfigurieren. Die prototypische Realisierung bestätigte das von der Referenzarchitektur angestrebte Ziel, durch einheitliche informationstechnische Schnittstellen die Wandelbarkeit in einer nach Lean Production organisierten Fertigungslinie zu verbessern.

6. Bewertung und Ausblick

Ausgehend von dem Stand der Technik in der Produktionsdigitalisierung und der Lean Production wurde in den vorherigen Kapiteln die Problemstellung hergeleitet. Die vorgestellte Referenzarchitektur zeigte hierfür einen Lösungsansatz, der in der prototypischen Realisierung verifiziert wurde. Dieses Kapitel reflektiert die Arbeit, indem es die Teilergebnisse und die Vorgehensweise bewertet (Kapitel 6.1) und einen Ausblick auf weitere Forschungsbedarfe gibt (Kapitel 6.2).

6.1 Zusammenfassende Bewertung

Übergeordnetes Ziel der vorliegenden Arbeit ist es, die Wandelbarkeit der Lean Production mittels einheitlicher informationstechnischer Schnittstellen zu verbessern und die erfolgreiche Lean Production in der vermehrt IT-geprägten Produktion zu verankern. Die vorliegende Arbeit erarbeitete hierfür eine Referenzarchitektur, die diese Schnittstellen für die Entitätstypen der Lean Production definiert. Die zu klärenden Forschungsfragen waren, welche Entitäten in der Lean Production welche Nachrichtentypen mittels welcher Dienste miteinander austauschen und wie eine Schnittstelle aufgebaut sein muss, die sich bei bestehenden und zukünftigen Entitäten einer Lean-Umgebung umsetzen lässt.

Die Systemarchitektur der Referenzarchitektur beschreibt die für die Lean Production benötigten Entitätstypen. Im Gegensatz zu anderen Referenzarchitekturen verwendet diese Arbeit anstatt eines Ebenenmodells ein Rollenkonzept für die Beschreibung. Die Systemarchitektur trennt abstrakte Rollen mit den von ihnen angebotenen informationstechnischen Schnittstellen als logische Gruppierung und physischen Ressourcen zur Realisierung der Rollen. Dies ermöglicht es, bereits bestehende Produktionsumgebungen mithilfe der Referenzarchitektur zu modellieren, notwendige Schnittstellen zu identifizieren und die Komplexität zu reduzieren. Bedingt durch den Fokus auf die Lean Production lässt die Systemarchitektur offen, wie die Auftragsfreigabe in bestehenden MES und ERP-Systemen umzusetzen ist und welche Wechselwirkungen zu nicht berücksichtigten, weiteren IT-Systemen bestehen. Ebenfalls bedingt durch die Lean Production lässt sich die Systemarchitektur nur auf Stückgutprozesse der Klein- und Großserie übertragen und nicht auf verfahrenstechnische Prozesse oder Werkstattfertigungen.

Das Informationsmodell der Referenzarchitektur beschreibt, welche Dienste die zuvor definierten Rollen für jede Lean-Methode anbieten müssen und welche Nachrichten sie austauschen. Es berücksichtigt nur die notwendigen Informationen und Dienste; die definierten Nachrichtentypen und deren Attribute lassen sich nachträglich ergänzen. Durch die technologieunabhängige Beschreibung definiert das Informationsmodell keine Dienste, die für die Interprozesskommunikation notwendig sind. Hierzu gehören z.B. Dienste für den Kommunikationsaufbau und die Handhabung von verlorenen Nachrichten. Die mangelnde Eindeutigkeit und Beschreibung mancher Lean-Methoden führte dazu, dass die zugehörigen Dienste und Nachrichten im Extremfall nur als Platzhalter aufgenommen wurden. Die ausbleibende Evaluation in der Praxis hätte dies ergänzen können. Die Dokumentation des Informationsmodells erfolgte mittels Klassen- und Kommunikationsdiagramme sowie Tabellen. Wie die prototypische Realisierung zeigte, reicht dies für eine maschinenlesbare, semantische Umsetzung aus. Die Verwendung einer Ontologie wurde für diese Arbeit verworfen, da Ontolo-

gien der Beschreibung von Wissen und nicht von Funktionalitäten dienen. Eine Überführung des Informationsmodells in eine Ontologie ist aber bei Bedarf mit wenig Aufwand möglich.

Die Schnittstellenarchitektur beschreibt als dritter Bestandteil der Referenzarchitektur notwendige Softwarekomponenten für die Umsetzung der informationstechnischen Schnittstellen. Sie berücksichtigt unterschiedliche Hardware und Migrationsmöglichkeiten und ist modular aufgebaut, um ebenfalls die Wandelbarkeit zu unterstützen. Die Berücksichtigung von Konzepten und Prinzipien des objektorientierten Designs ermöglicht eine hohe Wiederverwendbarkeit und Skalierbarkeit des Softwarecodes. Die Schnittstellenarchitektur unterstützt den Entwickler bei der Strukturierung der Softwarekomponenten, gibt allerdings keine konkrete Hilfestellung für die Anbindung unterschiedlicher Feldgeräte. Die prototypische Realisierung zeigte, wie sich die Schnittstellenarchitektur auf CPSen unter Verwendung von Hochprogrammiersprachen umsetzen lässt. Die Übertragung auf bestehende SPSen, welche ohne zusätzlichen Adapter die informationstechnische Schnittstelle anbieten, wurde nicht realisiert. Dies sollte allerdings bei neueren SPSen einfach möglich sein, da sie vermehrt auf Industrie-PCs als Hardware und Schnittstellen zur Erweiterung der Funktionalität setzen.

Der in Kapitel 4.2.3 aufgestellte Anforderungskatalog konnte weitestgehend berücksichtigt werden. Die funktionalen Anforderungen fanden Berücksichtigung in der Referenzarchitektur, wo sie durch Dienste und ausgetauschte Nachrichten realisiert wurden. Auf welche Art und Weise jede Rolle die Anforderungen intern realisiert, war im Sinne des Black-Box-Ansatzes allerdings nicht Bestandteil der Betrachtung. Bei der Umsetzung der funktionalen Anforderungen ist aufgefallen, dass Dienste von Lean-Methoden mehrfach verwendet werden. Ein Beispiel hierfür ist die von allen Rollen angebotene Funktion „empfangVerbesserungsmaßnahme“, welche die Lean-Methoden Gemba, Six Sigma, Prozessstandardisierung, Audit, Benchmarking, Ideenmanagement, Poka Yoke und Verschwendungsbewertung verwenden.

Bei den nicht-funktionalen Anforderungen konnte die Forderung nach einer definierten Dienstgütequalität nicht eingehalten werden, da sie vom verwendeten Kommunikationsprotokoll abhängig ist. Gleiches gilt für die Umsetzung des Pull-Prinzips. Auch lässt sich eine informationstechnische Schnittstelle im Sinne der günstigen und selbst erstellten Automatisierung nicht realisieren. Für die Implementierung sind Kenntnisse in der Automatisierungstechnik und bzw. oder Softwareentwicklung notwendig, welche vermutlich nicht in allen Unternehmen in diesem Umfang vorhanden sind. Alle anderen Anforderungen flossen allerdings in die Gestaltung der Referenzarchitektur ein. Sie ist transparent, standardisiert und so schlank wie möglich gestaltet. Sie ermöglicht sowohl die Erweiterung der Funktionalitäten als auch eine Realisierung auf unterschiedlichen Technologien. Durch die Einführung der übergeordneten, abstrakten Rolle „iS-Rolle“ existiert eine höchstmögliche Ähnlichkeit zwischen den Rollen. Die Schnittstellenarchitektur unterstützt eine aufwandsarme Migration bestehender Umgebungen.

Zusammenfassend lässt sich für die erarbeiteten Ergebnisse festhalten, dass sie die konzeptionelle, technologieunabhängige Grundlage für eine standardisierte, digitale Umsetzung der Lean-Methoden legen. Obwohl die Referenzarchitektur auf der Richtlinie [Ver13] und gängiger Literatur zur Lean Production aufbaut, sind nicht alle Lean-Methoden hinreichend standardisiert. Das Problem der geringen Formalisierung und Konsistenz von Lean-Methoden wurde in der Wissenschaft bereits erkannt und kritisiert (siehe z.B. [Sch+15]). Auch bedarf es bei der Realisierung unternehmensindividueller Anpassungen. Insofern ist eine allgemeingül-

tige Spezifikation nur schwer möglich. Die vorliegende Referenzarchitektur versucht, dem durch Erweiterbarkeit und technologische Unabhängigkeit zu begegnen.

Die prototypische Realisierung zeigte, dass die Referenzarchitektur hinreichend beschrieben ist, um bestehende Lean-Methoden digital unterstützt umzusetzen. Die umgesetzten Anwendungsszenarien demonstrierten die schnelle Rekonfiguration von Fertigungslinien und den verbesserten Austausch von Systemen in der Lean Production. Dies beruht zum einen auf den einheitlichen Schnittstellen, die die Modularität verbessern, und zum anderen auf den definierten digitalen Diensten und Nachrichten für die Lean-Methoden, durch die die Standardisierung und ein kurzer Informationsfluss ermöglicht werden. Das übergeordnete Ziel der Arbeit, die Verbesserung der Wandelbarkeit, ist somit durch die Referenzarchitektur erreicht. Offen ist allerdings die Quantifizierung der Verbesserung im Vergleich zum Status quo.

Hinsichtlich des Vorgehens zur Erstellung der Referenzarchitektur lässt sich zusammenfassend feststellen, dass die Kombination aus Ansätzen des Systems Engineerings, des Anforderungsmanagements und der modellgetriebenen Entwicklung zielführend war. Der Top-Down-Ansatz und das mehrstufige Vorgehen zur Überführung der Lean-Methoden in einer Referenzarchitektur konnten die von der Vielzahl an Lean-Methoden verursachte Komplexität reduzieren und die Nachvollziehbarkeit ermöglichen.

Die anfängliche Bewertung und Auswahl der zur Digitalisierung geeigneten Lean-Methoden ist für die eigentliche Erstellung der Referenzarchitektur nicht notwendig. Dennoch half sie im Rahmen dieser Arbeit dabei, ein besseres Verständnis der Lean-Methoden und deren Potenzial zur Digitalisierung zu schaffen. Die Bewertung reduzierte ferner den Umfang der Arbeit, indem frühzeitig nicht relevante Lean-Methoden aussortiert wurden.

Die Modellierung der Lean-Methoden als UML Aktivitätsdiagramme war ein wichtiger Schritt, um die je Lean-Methode vorhandenen Quellen in einem Modell einheitlich zu beschreiben. Experteninterviews oder Ist-Aufnahmen in bestehenden Produktionsumgebungen hätten neben der Literaturrecherche die Qualität und Allgemeingültigkeit der Arbeit verbessern können, konnten allerdings Mangels Verfügbarkeit nicht herangezogen werden. Die Definition von Anwendungsfällen je Lean-Methode half dabei, die komplexen Abläufe in kleinere, isoliert betrachtete Abschnitte zu zerlegen. Im Gegensatz zur Prozessoptimierung war es bei der Modellierung wichtig, den Fokus auf den Austausch von Nachrichten zwischen Prozessschritten zu legen. Ohne diese Sichtweise und ohne den verwendeten Black-Box-Ansatz würden die Aktivitätsdiagramme für die Arbeit irrelevante Prozessschritte abbilden.

Der erstellte Anforderungskatalog half anschließend, die umfangreichen Anforderungen an die Referenzarchitektur zu strukturieren. Die Satzschablonen waren ein einfaches, aber wirkungsvolles Werkzeug, um aus den Aktivitätsdiagrammen die involvierten Rollen und ausgetauschten Nachrichten abzuleiten. Dieser Arbeitsschritt ist nicht zwingend notwendig, half allerdings dabei, die Komplexität zu reduzieren und Nachvollziehbarkeit herzustellen.

Bei der Erstellung der Referenzarchitektur wurde, im Gegensatz zu klassischen Prozessmanagement-Ansätzen, nicht mit der Prozesssicht, sondern mit der Struktursicht begonnen. Dies war insofern notwendig, um die verfügbaren Akteure einzugrenzen und um festzulegen, welche informationstechnischen Schnittstellen notwendig sind. Bei der Spezifikation der Nachrichtentypen zeigte sich, dass nur wenige Lean-Methoden detailliert genug beschrieben sind. Teilweise konnten die Nachrichtenattribute nur durch tiefer gehende Recherchen oder

aus der übergeordneten Zielsetzung der Lean Production hergeleitet werden. Dies war beispielsweise bei den Six-Sigma-Kennzahlen der Fall. Bei der Definition der Nachrichtenattribute war auch eine klare Abgrenzung von Lean-Methoden zu anderen Konzepten wie Total Productive Maintenance oder Six Sigma notwendig, was aufgrund der hohen Ähnlichkeit nicht immer einfach war. Die definierten Nachrichtentypen waren somit ein Kompromiss, der die für die informationstechnischen Schnittstellen notwendige Spezifikation mit der teilweise nicht detailliert genug vorliegenden Definition der Lean-Methoden und für die individuelle Umsetzung in der Praxis notwendigen Freiheiten vereint.

Für das Vorgehen lässt sich zusammenfassend festhalten, dass die Verwendung insbesondere von Ansätzen aus dem Systems Engineering zusammen mit einem schrittweisen Vorgehen für die initiale Erstellung der Referenzarchitektur zielführend war. Spätere Erweiterungen der Referenzarchitektur können auf die Bewertung der Lean-Methode und ggf. auf die Anforderungsdokumentation verzichten. Problematisch war die Modellierung, da es für die Modellierung von Architekturen und Informationsmodellen für verteilte Systeme kaum etablierte Vorgehen gibt. Die Systemarchitektur und die Kommunikationsdiagramme waren ein schlanker Ansatz, der mit vertretbarem Aufwand den notwendigen Informationsbedarf befriedigte.

6.2 Ausblick auf weiteren Forschungs- und Entwicklungsbedarf

Der weitere Forschungs- und Entwicklungsbedarf unterteilt sich in die inhaltliche Weiterentwicklung der Referenzarchitektur, deren technologische Realisierung in der Praxis und auf der Referenzarchitektur aufbauende, weitere Arbeiten.

Bezogen auf die inhaltliche Weiterentwicklung besteht weiterer Evaluations- und Standardisierungsbedarf, um die Allgemeingültigkeit der Referenzarchitektur zu verbessern. In der Praxis ist wahrscheinlich immer eine individuelle Anpassung notwendig. Die Referenzarchitektur ist daher aktuell eher als Empfehlung oder Best Practice für die digitale Umsetzung von Lean-Methoden zu sehen. Eine Umsetzung in mehreren, realen Produktionsumgebungen kann die Qualität des Informationsmodells verbessern und neben den bereits identifizierten, notwendigen auch sinnvolle, optionale Dienste und Nachrichten definieren. Hierdurch wird auch die Wandelbarkeit hinsichtlich für die Lean-Methoden eingesetzter Technologien verbessert.

Darüber hinaus bedarf es für den praxisrelevanten Einsatz auch einer Übertragung der Referenzarchitektur auf andere, zukunftsrelevante Ansätze. Hierzu gehört die vorgestellte Verwaltungsschale der RAMI 4.0. Zu klärende Fragen sind, wie sich die Rollen, deren Dienste und die ausgetauschten Nachrichten auf das Konzept der Sichten, APIs und Teilmodelle der Verwaltungsschale übertragen lassen. Eine Companion Specification kann ergänzend beschreiben, wie sich das Informationsmodell aus der Referenzarchitektur mittels des Meta-Informationsmodells aus OPC UA umsetzen lässt. Hierdurch wäre für Entwickler ein eindeutiger Bezug zwischen der hier erarbeiteten technologieunabhängigen Referenzarchitektur und der konkreten Implementierung gegeben. Der zukünftig in OPC UA verfügbare Publish-Subscribe-Mechanismus könnte eine ereignisbasierte Kommunikation realisieren.

Für die technologische Realisierung bedarf es weiterer Arbeiten bei der Schnittstellenarchitektur. Insbesondere für die Ressourcen-Interface-Gruppe, die die Kommunikation zwischen

den Schnittstellen und die Anbindung der Steuerungen und Feldgeräte übernimmt, besteht Potenzial, durch einheitliche Bibliotheken die Integration bis hin zu einem Plug'n'Produce-Ansatz zu vereinfachen. [Hod13] beschreibt einen Ansatz zur flexiblen Integration von Feldgeräten mittels einheitlicher Bibliotheken, der auf die Schnittstellenarchitektur ggf. übertragen werden kann. Denkbar wäre auch die Anbindung der CPS-basierten informationstechnischen Schnittstellen an eine gemeinsame digitale Plattform, aus der die notwendigen Treiber und Softwarekomponenten bei Bedarf nachgeladen werden. Inspiriert von den sogenannten App-Stores von Smartphones stellte [Sch16] einen Ansatz vor, wie industrielle Komponenten mittels nachträglich installierter Software neue Dienste zur Verfügung stellen können.

Die vorliegende Arbeit hat nicht nur einen Ansatz gezeigt, wie sich die Wandelbarkeit der Lean Production mittels digitaler Technologien verbessern lässt, sie schaffte auch die Grundlage für die Umsetzung der Lean Production in einer von IT geprägten Produktionsumgebung. Aufbauend hierauf lassen sich neue Anwendungen zur Weiterentwicklung der Lean Production erarbeiten. Aus technologischer Sicht besteht Forschungsbedarf bei der Umsetzung weiterer Lean-Methoden mittels digitaler Technologien. Die in Kapitel 2.2.3 vorgestellten Lean-Automation-Anwendungen decken bei weitem nicht alle Lean-Methoden ab. Es ist somit zu erforschen, inwiefern neue Technologien wie Datenbrillen, künstliche Intelligenz oder Wearables in den Lean-Methoden genutzt werden können, um neben Zeit, Qualität und Kosten die Wandelbarkeit weiter zu verbessern. Die in der Referenzarchitektur erarbeiteten informationstechnischen Schnittstellen liefern hierfür die notwendigen Informationen und reduzieren Integrationsaufwände. Abbildung 84 zeigt Beispiele für zukünftige Lean Automation-Anwendungen. Ferner ordnet sie die Verwaltungsschale aus RAMI 4.0 sowie die OPC UA Companion Specification in den Kontext dieser Arbeit ein.

Die Arbeit befasste sich damit, wie digitale Technologien die Lean Production positiv beeinflussen können. Nicht betrachtet wurde dabei umgekehrt, wie Lean-Philosophien die Entwicklung von Industrie-4.0-Anwendungen beeinflussen können. Durch den vermehrten Einzug von IKT in die Produktion steigt die Komplexität der Systeme. Neue, digitalisierte Lösungen müssen für den Werker allerdings genauso intuitiv und einfach bedienbar sein. Sie müs-

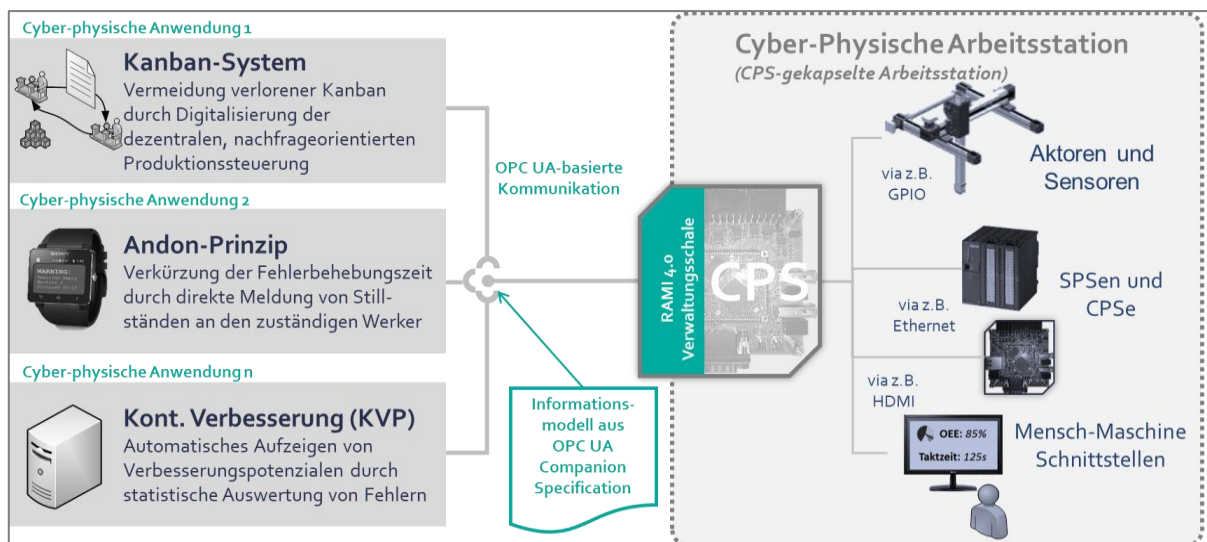


Abbildung 84 Beispiel und Ausblick für weitere Forschungsaktivitäten im Kontext der informationstechnischen Schnittstellen für Lean-Methoden

sen so robust in anspruchsvollen Produktionsumgebungen funktionieren wie die altbekannten, physischen Werkzeuge. Auch die Entwicklung und vor allem die Wartung und Erweiterung der eingeführten Systeme muss für Automatisierungstechniker und Informatiker einfach sein, um der wachsenden Dynamik gerecht zu werden. Die Lean Production hingegen erlangte Bekanntheit aufgrund ihrer Einfachheit, Wirksamkeit und Integration des Menschen. Es gilt somit zu erforschen, wie Industrie-4.0-Technologien diese Aspekte in der Entwicklung, in der Integration und im Betrieb umsetzen können, damit sie ebenfalls „lean“ bleiben.

Zusammenfassend besteht Forschungsbedarf bei der Evaluation der vorgestellten Referenzarchitektur, um die Wandelbarkeit und Übertragbarkeit weiter zu verbessern. Ferner bedarf es konkreter Ansätze, um das technologieunabhängige Konzept in die Praxis umzusetzen. Zuletzt muss im Kontext der Referenzarchitektur erforscht werden, wie neue Technologien weitere Lean Methoden verbessern können. Unabhängig davon ist noch ungeklärt, welchen Einfluss die Lean-Philosophie auf Industrie-4.0-Technologien haben kann. Lean Automation beschreibt mehr als die in dieser Arbeit betrachtete Umsetzung von Lean-Methoden unter Verwendung digitaler Technologien. Es beschreibt auch die Anwendung der Lean-Philosophie auf deren Entwicklung. In der Produktion muss IKT zukünftig so schlank und selbstverständlich bedienbar sein wie heutzutage Hammer, Schraubendreher und Bohrmaschine.

7. Zusammenfassung

Aufgrund des vergangenen Erfolges und der Einfachheit ist die Lean Production heutzutage der Status quo in der diskreten Klein- und Großserienfertigung. Die mangelnde Berücksichtigung von Wandelbarkeit lässt die Lean Production hinsichtlich zukünftiger Marktanforderungen allerdings an ihre Grenzen stoßen. Auch ist das Potenzial moderner digitaler Technologien in dem Produktionssystem nicht ausreichend berücksichtigt. Industrie 4.0 hingegen fasst eine Entwicklung zusammen, die die durchgängige Vernetzung und Integration von CPSen in der Produktion beschreibt, um den zukünftigen Anforderungen gerecht zu werden. Dieses technologiegetriebene Vorgehen berücksichtigt aktuell nicht die Integration in und die Migration von bestehenden Prozessen und Organisationen.

Ziel der vorliegenden Arbeit war die Verbesserung der Wandelbarkeit in der Lean Production durch einheitliche informationstechnische Schnittstellen. Die erarbeitete Referenzarchitektur ermöglicht die Integration digitaler Technologien in die Lean-Methoden, um die vorgenannten Schwachstellen zu mindern. Im Fokus stand hierbei die digitale Kommunikation zwischen Beteiligten in der Lean Production. Kern der Arbeit war daher die Spezifikation informationstechnischer Schnittstellen, welche für eine Anbindung in ein gemeinsames Kommunikationsnetzwerk die von jedem anzubietenden Dienste und ausgetauschten Informationen beschrieb.

Für die Entwicklung der Referenzarchitektur wurde auf etablierte Ansätze aus dem Systems- und Requirements-Engineering zurückgegriffen. Die anfängliche Bewertung stellte eine schlanke, zielgerichtete Digitalisierung der Lean-Methoden sicher und sondierte Methoden aus, die mangels Mehrwert durch Digitalisierung oder unklarer Beschreibungen nicht digital umsetzbar sind. Beispiele hierfür sind das Cardboard-Engineering und die A3-Methode. Die Unterteilung in Anwendungsfälle und die anschließende Modellierung als Aktivitätsdiagramme ermöglichte es, Teilprozesse isoliert zu betrachten und die Komplexität zu reduzieren. Ausgehend von den Modellen wurde ein Anforderungskatalog mit funktionalen und nicht-funktionalen Anforderungen erstellt, die als Eingangswerte in die Referenzarchitektur einfließen. Die funktionalen Anforderungen beschrieben aus den Modellen abgeleitete Informationsbedarfe und benötigte Funktionen der Produktionsbeteiligten. Die nicht-funktionalen Anforderungen leiteten sich aus den Paradigmen von Industrie 4.0, der Lean Production sowie der modernen Softwareentwicklung ab. Wichtige Aspekte waren die zukünftige Erweiterbarkeit und Integration in heterogene Umgebungen.

Einer von drei Bestandteilen der Referenzarchitektur ist die Systemarchitektur, welche die in der Lean Production relevanten Akteure als logische Rollen repräsentiert. Ferner legt sie fest, aus welchen Ressourcen sich diese Rollen in der Realität zusammensetzen und wo informationstechnische Schnittstellen notwendig sind. Zu den identifizierten Rollen gehören die Auftragsfreigabe, die Arbeitsstation, das Logistiksystem, der Supermarkt, der Hancho und das KVP-System.

Jede der identifizierten Rollen verfügt über eine informationstechnische Schnittstelle, welche die Rollen mittels eines gemeinsamen Kommunikationsnetzwerkes miteinander verbindet. Durch die Beschreibung der anzubietenden Dienste je Rolle können die Lean-Methoden digital umgesetzt werden. Auch wenn die Beschreibung der Dienste technologieunabhängig erfolgte, berücksichtigt sie unterschiedliche Kommunikationsansätze wie beispielsweise die

Request-Reply- oder die ereignisbasierte Methode. Neben den Diensten je Rolle umfasst der zweite Teil der Referenzarchitektur, das erarbeitete Informationsmodell, auch eine Definition der ausgetauschten Nachrichtentypen. Die Nachrichtentypen enthalten minimal anzubietende Attribute für die einzelnen Lean-Methoden. Das Informationsmodell der Referenzarchitektur ist dabei so gestaltet, dass Nutzer es später erweitern und individualisieren können. Auch ist eine durchgängige Nachverfolgbarkeit zwischen Lean-Methode und Realisierung in der Schnittstelle möglich.

Als dritter Bestandteil umfasst die Referenzarchitektur eine Softwarearchitektur für die informationstechnischen Schnittstellen. Sie beschreibt, wie z.B. mithilfe der Industrie-4.0-Kerntechnologie CPS die Schnittstelle umgesetzt werden kann. Hierbei stand eine höchstmögliche Standardisierung von Softwarekomponenten im Fokus, die die Migration bestehender Produktionsumgebungen ermöglicht.

Die prototypische Realisierung der Referenzarchitektur anhand eines bestehenden industriellen Forschungsdemonstrators in der *SmartFactory*^{KL} diente abschließend der Verifikation der Ergebnisse. Hierbei zeigte sich, dass es die informationstechnischen Schnittstellen ermöglichen, auf den CPS-Stationen unterschiedlichste Lean-Methoden bereitzustellen, auf welche Dritte mittels unterschiedlichster Endgeräte zugreifen konnten. Ferner konnte der Softwarecode mehrfachverwendet werden, was die Entwicklungszeit reduzierte. Nichtsdestoweniger waren durch das verwendete Kommunikationsprotokoll MQTT Erweiterungen der Dienste notwendig, da die Referenzarchitektur aufgrund der Technologieunabhängigkeit diese nicht berücksichtigt.

Die vorliegende Arbeit hat die Grundlagen für die digitale Umsetzung von Lean-Methoden gelegt, um die Wandelbarkeit der Lean Production zu verbessern. Weiterer Forschungs- und Entwicklungsbedarf besteht in der Evaluation der Referenzarchitektur und der technischen Umsetzung der informationstechnischen Schnittstellen. Des Weiteren muss erforscht werden, wie sich aufbauend auf der Referenzarchitektur weitere Lean-Methoden durch IKT realisieren lassen. Unabhängig von dieser Arbeit gilt es zu erforschen, wie die Prinzipien der Lean Production die Entwicklung, die Integration und die Nutzung von Industrie-4.0-Technologien unterstützen können. Im Sinne der Lean Philosophie müssen zukünftige, digitale Anwendungen so schlank und einfach für den Anwender handhabbar sein wie sein alltägliches Werkzeug aus dem Werkzeugwagen. Erst hierdurch wird der Begriff „Lean Automation“ zusammen mit den Ergebnissen dieser Arbeit seiner Bedeutung gerecht.

8. Literaturverzeichnis

- [Abe+11] Abele, E. ; Reinhart, G.: Zukunft der Produktion : Herausforderungen, Forschungsfelder, Chancen. München : Hanser Verlag, 2011
- [aca14] acatech - Deutsche Akademie der Technikwissenschaften: Industrie 4.0 - Whitepaper FuE Themen. URL http://www.acatech.de/fileadmin/user_upload/Baumstruktur_nach_Website/Acatech/root/de/Aktuelles___Presse/ Presseinfos___News/ab_2014/Whitepaper_Industrie_4.0.pdf – Überprüfungsdatum 2017-11-12
- [Acm] Acme Systems srl: Aria G25 - Low cost Linux Embedded SMD module. URL <https://www.acmesystems.it/aria> – Überprüfungsdatum 2017-11-19
- [Ado+15] Adolphs, P. ; Bedenbender, H. ; Dirzus, D. ; Ehlich, M. ; Epple, U. ; Hankel, M. ; Heidel, R. ; Hoffmeister, M. ; Huhle, H. ; Kärcher, B. ; Koziolk, H. ; Pichler, R. ; Pollmeier, S. ; Schewe, F. ; Walter, A. ; Waser, B. ; Wollschlaeger, M.: Referenzarchitekturmodell Industrie 4.0 (RAMI4.0) : Statusreport. URL https://www.vdi.de/fileadmin/user_upload/VDI-GMA_Statusreport_Referenzarchitekturmodell-Industrie40.pdf – Überprüfungsdatum 2017-11-12
- [Ame95] American National Standard: ANSI/ISA-8.01-1995. Batch control - Part 1: Models and Terminology. Research Triangle Park, N.C. : ISA, 1995
- [Ank+08] Ankele, A. ; Staiger, T. J. ; Koch, T.: Chefsache Produktionssystem : Pfade zum Erfolg. Stuttgart : LOG_X Verlag, 2008
- [Aus+04] Austin, D. ; Barbir, A. ; Ferris, C. ; Garg, S.: Web Service Architecture Requirements. URL <https://www.w3.org/TR/wsa-reqs/> – Überprüfungsdatum 2017-03-14
- [Bal11] Balzert, H.: Lehrbuch der Softwaretechnik : Entwurf, Implementierung, Installation und Betrieb. 3. Aufl. Heidelberg : Spektrum Akademischer Verlag, 2011
- [Bau14] Bauernhansl, T.: Die Vierte Industrielle Revolution – Der Weg in ein wertschaffendes Produktionsparadigma. In: Vogel-Heuser; Bauernhansl; ten Hompel (Hrsg.): Industrie 4.0 in Produktion, Automatisierung und Logistik : Anwendung, Technologien und Migration. Wiesbaden : Springer Vieweg, 2014, S. 5–35
- [Bec+04] Becker, J. ; Schütte, R.: Handelsinformationssysteme : Domänenorientierte Einführung in die Wirtschaftsinformatik. 1. Aufl. München : Mi-Wirtschaftsbuch, 2004
- [Beu+15] Beudert, R. ; Leurs, L. ; Zuponcic, S.: Machinery Information Base Data Structure. In: ODVA 2015 Industry Conference & 17th Annual Meeting, 2015

- [Bib] Bibliographisches Institut GmbH: digitalisieren. URL <http://www.duden.de/node/716435/revisions/1375044/view> – Überprüfungsdatum 2017-11-19
- [Bib16] Bibliographisches Institut GmbH: Schnittstelle, die. URL <http://www.duden.de/node/712790/revisions/1151882/view> – Überprüfungsdatum 2017-11-19
- [Bil+12] Bilberg, A. ; Hadar, R.: Adaptable and Reconfigurable LEAN Automation - a competitive solution in the western industry. In: 22th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM), 2012, S. 389–396
- [Bin15] Bin Tang, C.: Explore MQTT and the Internet of Things service on IBM Bluemix : Use a sample Java app or the Node-RED editor to learn how it works. URL <https://www.ibm.com/developerworks/cloud/library/cl-mqtt-bluemix-iot-node-red-app/> – Überprüfungsdatum 2017-03-05
- [Ble14] Bleher, N.: Produktionssysteme erfolgreich einführen. Wiesbaden : Springer Gabler, 2014
- [Blo+08] Bloech, J. ; Bogaschewsky, R. ; Buscher, U. ; Daub, A. ; Götze, U. ; Roland, F.: Einführung in die Produktion. 6., überarb. Aufl. Berlin : Springer, 2008
- [BMB13] BMBF: Zukunftsbild „Industrie 4.0“. URL https://www.bmbf.de/pub/Zukunftsbild_Industrie_4.0.pdf – Überprüfungsdatum 2017-12-13
- [Bod+09] Bode, W. ; Rose, D.: Zur rechten Zeit am rechten Ort. In: VDI-Ingenieur-Forum, Monat 01. Münster : VDI, 2009, S. 8–9
- [Bro+12] Brown, P. ; Estefan, J. A. ; Laskey, K. ; Thornton, D.: Reference Architecture Foundation for Service Oriented Architecture Version 1.0. URL http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/cs01/soa-ra-v1.0-cs01.html#_Toc343761541 – Überprüfungsdatum 2017-11-19
- [Bro10a] Broy, M. (Hrsg.): Cyber-Physical Systems : Innovation durch softwareintensive eingebettete Systeme. Berlin, Heidelberg : Springer Verlag, 2010
- [Bro10b] Broy, M.: Cyber-Physical Systems - Wissenschaftliche Herausforderungen bei der Entwicklung. In: Broy (Hrsg.): Cyber-Physical Systems : Innovation durch softwareintensive eingebettete Systeme. Berlin, Heidelberg : Springer Verlag, 2010, S. 17–31
- [Bru11] Brunner, F. J.: Japanische Erfolgskonzepte. 2. Aufl. München, Wien : Carl Hanser Fachbuchverlag, 2011
- [Bru+10] Bruns, R. ; Dunkel, J.: Event-Driven Architecture : Softwarearchitektur für ereignisgesteuerte Geschäftsprozesse. Berlin : Springer, 2010

- [Bru+14] Brusaferrì, A. ; Ballarino, A. ; Cavadini, F. A. ; Manzocchi, D. ; Mazzolini, M.: CPS-based hierarchical and self-similar automation architecture for the control and verification of reconfigurable manufacturing systems. In: IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 2014, S. 1–8
- [Bun] Bundesministerium für Bildung und Forschung, Referat IT-Systeme: Zukunftsbild "Industrie 4.0". URL http://www.bmbf.de/pubRD/Zukunftsbild_Industrie_40.pdf – Überprüfungsdatum 2017-11-12
- [Bun16] Bundesministerium für Wirtschaft und Energie (BMWi): Struktur der Verwaltungsschale. URL http://www.plattform-i40.de/I40/Redaktion/DE/Downloads/Publikation/struktur-der-verwaltungsschale.pdf?__blob=publicationFile&v=6 – Überprüfungsdatum 2017-11-19
- [Chi+07] Chinnici, R. ; Moreau, J.-J. ; Ryman, A. ; Weerawarana, S.: Web Service Description Language (WSDL) Version 2.0 : Part 1: Core Language. URL <http://www.w3.org/TR/wsdl20/wsdl20.pdf> – Überprüfungsdatum 2017-11-19
- [Cis+02] Cisek, R. ; Habicht, C. ; Neise, P.: Gestaltung wandlungsfähiger Produktionssysteme. In: Zeitschrift für wirtschaftlichen Fabrikbetrieb (ZWF), Bd. 97, Nr. 09, 2002, S. 441–445
- [Cla05] Clarke, C.: Automotive Production Systems and Standardisation : From Ford to the Case of Mercedes-Benz. Heidelberg : Physica-Verlag, 2005
- [Clo+09] Cloutier, R. ; Muller, G. ; Verma, D. ; Nilchiani, R. ; Hole, E. ; Bone, M.: The Concept of Reference Architectures. In: Systems Engineering, Bd. 13, Nr. 1, 2009, S. 14-27
- [Col+05] Colombo, A. W. ; Jammes, F. ; Smit, H. ; Harrison, R. ; Lastra, J.L.M. ; Delamer, I. M.: Service-oriented architectures for collaborative automation. In: 31st Annual Conference of IEEE Industrial Electronics Society (IECON), 2005, S. 2649-2654
- [Col+13] Colombo, A. W. ; Karnouskos, S. ; Bangemann, T.: A system of systems view on collaborative industrial automation. In: IEEE International Conference on Industrial Technology (ICIT), 2013, S. 1968–1975
- [Com06] Computerwoche: Gartner propagiert SOA 2.0. URL <http://www.computerwoche.de/a/gartner-propagiert-soa-2-0,577497> – Überprüfungsdatum 2017-11-19
- [Cor+] Corsaro, A. ; Schmidt, D. C.: The Data Distribution Service : The Communication Middleware Fabric for Scalable and Extensible Systems-of-Systems. URL

- <http://www.dre.vanderbilt.edu/~schmidt/PDF/dds-sos.pdf> – Überprüfungsdatum 2017-11-12
- [Cou+05] Coulouris, G. ; Dollimore, J. ; Kindberg, T.: Verteilte Systeme : Konzepte und Design. 3., überarb. Aufl. München : Pearson Studium, 2005
- [Dac+03] Daconta, M. C. ; Obrst, L. J. ; Smith, K. T.: The semantic web : A guide to the future of XML, web services, and knowledge management. Indianapolis, USA : Wiley Publishing, 2003
- [Dam+09] Damm, M. ; Leitner, S.-H. ; Mahnke, W.: OPC Unified Architecture. Berlin, Heidelberg : Springer-Verlag Berlin Heidelberg, 2009
- [Dan97] Danuser, R.: Business process reengineering : Der Wandel zu prozessorientierten Strukturen. Bern : Haupt, 1997
- [dc-17] dc-square GmbH: MQTT over Websockets with HiveMQ. URL <http://www.hivemq.com/blog/mqtt-over-websockets-with-hivemq> – Überprüfungsdatum 2017-03-06
- [Den12] Dengel, A.: Semantische Technologien : Grundlagen - Konzepte - Anwendungen. Heidelberg : Springer Berlin Heidelberg, 2012
- [Deß13] Deßloch, S.: Informationsmodelle : Vorlesungsunterlagen der Vorlesung Informationssysteme SS 2013. URL <http://www.lgis.informatik.uni-kl.de/cms/fileadmin/courses/SS2013/IS/lecture/k03-Informationsmodelle.pdf> – Überprüfungsdatum 2017-03-27
- [Deu+15] Deuse, J. ; Weisner, K. ; Hengstebeck, A. ; Busch, F.: Gestaltung von Produktionssystemen im Kontext von Industrie 4.0. In: Botthof; Hartmann (Hrsg.): Zukunft der Arbeit in Industrie 4.0. Berlin : Springer Vieweg, 2015, S. 99–109
- [Dic07a] Dickmann, E.: Elektronische Kanban-Systeme (eKanban). In: Dickmann (Hrsg.): Schlanker Materialfluss : mit Lean Production, Kanban und Innovationen. Berlin, Heidelberg, New York : Springer Vieweg, 2007, S. 340–347
- [Dic07b] Dickmann, P.: Hybride Steuerungskonzepte. In: Dickmann (Hrsg.): Schlanker Materialfluss : mit Lean Production, Kanban und Innovationen. Berlin, Heidelberg, New York : Springer Vieweg, 2007, S. 143–148
- [Dic07c] Dickmann, P. (Hrsg.): Schlanker Materialfluss : mit Lean Production, Kanban und Innovationen. Berlin, Heidelberg, New York : Springer Vieweg, 2007
- [DIN08] DIN Deutsches Institut für Normung e.V.: DIN EN 62264-3. Integration von Unternehmens-EDV und Leitsystemen - Teil 3: Aktivitätsmodelle für das operative Produktionsmanagement. Berlin : Beuth Verlag, 2008-01
- [Dom+15] Dombrowski, U. (Hrsg.); Mielke, T. (Hrsg.): Ganzheitliche Produktionssysteme : Aktueller Stand und zukünftige Entwicklungen. Berlin : Springer, 2015

- [Döp15] Döppler.Team: Das "U" macht den Unterschied. URL <http://www.leanmagazin.de/lean-praxis/lean-production/1080-das-u-macht-den-unterschied.html> – Überprüfungsdatum 2017-11-19
- [eCl12] eCl@ss e.V.: Suche in eCl@ss nach Klassen, Merkmalen und Werten : 27-02-31-01 Frequenzumrichter =< 1 kV [AKE177013]. URL http://www.eclasscontent.com/index.php?id=27023101&version=9_1&language=de&action=det – Überprüfungsdatum 2017-11-09
- [eCl14] eCl@ss e.V.: Classification Class. URL http://wiki.eclass.eu/wiki/Classification_Class – Überprüfungsdatum 2017-11-19
- [eCl15] eCl@ss e.V.: Der eCl@ss-Standard. URL <http://www.eclass.de/eclasscontent/standard/overview.html.de> – Überprüfungsdatum 2017-11-19
- [Ehr+15] Ehrlich, M. ; Wisniewski, L. ; Jasperneite, J.: Usage of Retrofitting for Migration of Industrial Production Lines to Industry 4.0. In: Jahreskolloquium Kommunikation in der Automation (KommA), 2015
- [Erl07] Erlach, K.: Wertstromdesign : Der Weg zur schlanken Fabrik. Berlin, Heidelberg : Springer Verlag, 2007
- [Eur16] European Innovation Hub GmbH: everlean | Die digitale App für Lean Management (KVP). URL <http://www.everlean.de/> – Überprüfungsdatum 2016-10-05
- [Eve96] Eversheim, W.: Produktionstechnik und -verfahren. In: Kern (Hrsg.): Handwörterbuch der Produktionswirtschaft (Enzyklopädie der Betriebswirtschaftslehre, Bd. 7). 2. Aufl. Stuttgart : Schäffer-Poeschel, 1996
- [Fac] Facebook Inc.: Graph API - Übersicht. URL <https://developers.facebook.com/docs/graph-api/overview> – Überprüfungsdatum 2017-02-27
- [Fes11] Festo Didactic: Das Modulare Produktionssystem MPS schreibt Geschichte. URL http://www.festo-didactic.com/download.php?name=20%20Jahre%20MPS-Stationen.pdf&c_id=1100&file=20_jahre_mps_stationen_1.pdf – Überprüfungsdatum 2017-11-12
- [Fra93] Franke, R.: Automatisierung und Lean Production - ein Wertepaar? In: Logistik - Lösungen für die Praxis, Bd. 1, 1993, S. 389–409
- [Fra] Fraunhofer Institute for Manufacturing Engineering and Automation: LIAA Project Homepage. URL http://www.project-leanautomation.eu/index.php?id=liaa_home – Überprüfungsdatum 2017-12-12

- [Ger11] Gerberich, T.: Lean oder MES in der Automobilzulieferindustrie : Ein Vorgehensmodell zur fallspezifischen Auswahl. 1. Aufl. Wiesbaden : Gabler Verlag, 2011
- [Gla+16] Glass, R. ; Seifermann, S. ; Metternich, J.: The Spread of Lean Production in the Assembly, Process and Machining Industry. In: Alexopoulos (Hrsg.): 5th CIRP Global Web Conference Research and Innovation for Future Production (CIRPe), Bd. 55, 2016, S. 278–283
- [Göh13] Göhner, P.: Agentensysteme in der Automatisierungstechnik. Berlin, Heidelberg : Springer Vieweg, 2013
- [Gol14] Goll, J.: Architektur- und Entwurfsmuster der Softwaretechnik : Mit lauffähigen Beispielen in Java. 2., aktual. Aufl. Wiesbaden : Springer Vieweg, 2014
- [Gor+11] Gorécki, P. ; Pautsch, P.: Lean Management : Auf den Spuren des Erfolges der Managementphilosophie von Toyota und Co. 2. Aufl. München : Hanser, 2011
- [Gor+17] Gorecky, D. ; Hennecke, A. ; Schmitt, M. ; Weyer, S. ; Zühlke, D.: Wandelbare modulare Automatisierungssysteme. In: Reinhart (Hrsg.): Handbuch Industrie 4.0 : Geschäftsmodelle, Prozesse, Technik. München : Hanser Verlag, 2017, S. 555–583
- [Gor+16a] Gorecky, D. ; Weyer, S.: SmartFactoryKL Systemarchitektur für Industrie 4.0-Produktionsanlagen : SF-1.1: 04/2016. URL http://smartfactory.de/wp-content/uploads/2017/08/SF_WhitePaper_1-1_DE.pdf – Überprüfungsdatum 2017-12-13
- [Gor+16b] Gorecky, D. ; Weyer, S. ; Hennecke, A. ; Zühlke, D.: Design and Instantiation of a Modular System Architecture for Smart Factories. In: 12th IFAC Intelligent Manufacturing Systems (IMS), 2016, S. 79–84
- [Gra+16] Grangel-González, I. ; Halilaj, L. ; Coskun, G. ; Auer, S. ; Collarana, D. ; Hoffmeister, M.: Towards a Semantic Administrative Shell for Industry 4.0 Components. URL <http://arxiv.org/pdf/1601.01556.pdf> – Überprüfungsdatum 2017-11-19
- [Gro93] Groebel, K.-P.: Flexible Systeme zur Bauteilmontage : der Lean Production folgt die Lean Automation. In: Stahlmarkt, Bd. 1993, Nr. 43, 1993, S. 56–60
- [Hab+12] Haberfellner, R. ; Fricke, E. ; de Weck, O. ; Vössner, S.: Systems Engineering : Grundlagen und Anwendung. 12. Aufl. Zürich : Orell Füssli Verlag, 2012
- [Hab13] Habermüller, M.: Erfassungsgeräte an SAP-Systeme anbinden : Schlanke Produktionsabläufe mit Kanban und RFID-Technik. URL http://www.it-production.com/index.php?seite=einzel_artikel_ansicht&id=57721&page=2 – Überprüfungsdatum 2017-11-19

- [Han16] Hanser Verlag: Neuer Industrie-PC auf Basis des Raspberry Pi : Kunbus entwickelt offenen ›Revolution Pi‹ mit Raspbian. URL <https://www.hanser-konstruktion.de/news/uebersicht/artikel/neuer-industrie-pc-auf-basis-des-raspberry-pi-2185080.html> – Überprüfungsdatum 2017-11-12
- [Her17a] Herkommer, G.: Studie: Die Zukunft der Steuerungstechnik. URL <http://www.computer-automation.de/steuerungsebene/steuerregeln/artikel/143519/>
- [Her11] Herlyn, W. J.: PPS in der Automobilindustrie : Produktionsprogrammplanung von Fahrzeugen und Aggregaten. München : Carl Hanser, 2011
- [Her17b] Herrmann, L.: Vergleich bestehender Kommunikationsprotokolle und Informationsmodelle. Kaiserslautern, TU Kaiserslautern, Lehrstuhl für Produktionsautomatisierung. Studienarbeit. 2017
- [Hil02] Hiles, A.: The complete guide to IT service levels agreements : Aligning IT services to business needs. 3. Aufl. Brookfield, USA : Rothstein, 2002
- [Hod13] Hodek, S.: Methode zur vollautomatischen Integration von Feldgeräten in industrielle Steuerungssysteme. Kaiserslautern, TU Kaiserslautern, Lehrstuhl für Produktionsautomatisierung. Dissertation. 2013
- [Hu+12] Hu, L. ; Xie, N. ; Kuang, Z. ; Zhao, K.: Review of Cyber-Physical System Architecture. In: 2012 IEEE 15th International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops (ISORCW), 2012, S. 25–30
- [IBM] IBM Deutschland GmbH: IBM Integration Bus. URL <http://www-03.ibm.com/software/products/de/ibm-integration-bus> – Überprüfungsdatum 2017-11-19
- [in-17] in-integrierte informationssysteme GmbH: Die Kombination aus Lean und Industrie 4.0: in-GmbH stellt sphinx open Digital Teamboard vor. URL <https://sphinx-open.de/blog/2017/03/20/die-kombination-aus-lean-und-industrie-4-0-in-gmbh-stellt-sphinx-open-digital-teamboard-vor/>
- [Int11] International Organization for Standardization (ISO): 25000:2011-03. Software engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE. Berlin : Beuth Verlag, 2011-03
- [ITU14] ITU-T Focus Group on M2M Service Layer: M2M service layer: APIs and protocols overview : Focus Group Technical Report. URL http://www.itu.int/dms_pub/itu-t/opb/fg/T-FG-M2M-2014-D3.1-PDF-E.pdf – Überprüfungsdatum 2017-11-19
- [Jas+14] Jasti, N. V. K. ; Kodali, R.: Lean production : Literature review and trends. In: International Journal of Production Research, Bd. 53, Nr. 3, 2014, S. 867–885

- [Jed16] Jedermann, E.: Modulare Erweiterung ereignisbasierter Fertigungssteuerung in heterogener Umgebung am Beispiel eines Demonstrators der SmartFactoryKL. Kaiserslautern, TU Kaiserslautern, Arbeitsgruppe Robotersysteme, Fachbereich Informatik. Bachelorarbeit. 2016
- [JS 17] JS Foundation: Node-RED : A visual tool for wiring the Internet of Things. URL <https://nodered.org/> – Überprüfungsdatum 2017-03-06
- [Jun+16] Jung, R. ; Nüesch, R. ; Plazibat, A. ; Barón, J. F.: Digitalization Strategy : Drivers, Potentials and the Road to Implementation - A Case from the Media Company Ringier. In: Controlling, Bd. 28, Nr. 4-5, 2016, S. 248–253
- [Kag+13] Kagermann, H. ; Wahlster, W. ; Helbig, J.: Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0 : Abschlussbericht des Arbeitskreises Industrie 4.0. URL http://www.acatech.de/fileadmin/user_upload/Baumstruktur_nach_Website/Ac-atech/root/de/Material_fuer_Sonderseiten/Industrie_4.0/Abschlussbericht_Industrie4.0_barrierefrei.pdf – Überprüfungsdatum 2017-11-19
- [Klo15] Klose, M.: HARTING stellt MICA als Serienmodell auf der SPS vor. URL <https://www.pressebox.de/pressemitteilung/harting-kгаа/HARTING-stellt-MICA-als-Serienmodell-auf-der-SPS-vor/boxid/767481> – Überprüfungsdatum 2017-10-29
- [Kno17] Knobloch, J. B.: Design and Implementation of a modular CPS-Interface for a flexible Integration of Workstations in the Lean Production. Kaiserslautern, TU Kaiserslautern, Arbeitsgruppe Robotersysteme, Fachbereich Informatik. Masterarbeit. 2017
- [Koe89] Koestler, A.: The ghost in the machine. London : Arkana, 1989
- [Kol13] Kolberg, D.: Konzept eines Cyber-Physischen Logistiksystems in einer schlanken Produktion am Beispiel einer Reihenfertigung mit Kanban. Bremen, Universität Bremen, BIBA - Bremer Institut für Produktion und Logistik. Masterarbeit. 2013
- [Kol15a] Kolberg, D.: Informationsmodell für Cyber-Physische Systeme zur Beschreibung von Objekten von der Unternehmensebene bis zur Feldgeräteebene. In: Automation 2015 : Benefits of Change - the Future of Automation, 2015, S. 227–239
- [Kol+17] Kolberg, D. ; Berger, C. ; Pirvu, B.-C. ; Franke, M. ; Michniewicz, J.: CyProF - Insights from a Framework for Designing Cyber-Physical Systems in Production Environments. In: Factories of the Future in the digital environment - Proceedings of the 49th CIRP Conference on Manufacturing Systems, 2017, S. 32–37

- [Kol+16] Kolberg, D. ; Knobloch, J. ; Zühlke, D.: Towards a lean automation interface for workstations. In: International Journal of Production Research, Bd. 55, Nr. 10, 2016, S. 1–12
- [Kol+15b] Kolberg, D. ; Zühlke, D.: Lean Automation enabled by Industry 4.0 Technologies. In: Dolgui; Sasiadek; Zaremba (Hrsg.): 15th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2015, 2015, S. 1870–1875
- [Kor+04] Korge, A. ; Scholtz, O.: Ganzheitliche Produktionssysteme : Produzierende Unternehmen innovativ organisieren und führen. In: wt Werkstattstechnik online, Bd. 94, Nr. 1/2, 2004, S. 2–6
- [Kov+12] Kováč, M. ; Kováčová, L.'u.: Analysis of tools for lean manufacturing. In: Transfer inovácií, Nr. 24, 2012, S. 17–20
- [Kra88] Krafcik, J. F.: Triumph of the Lean Production System. In: SLOAN Management Review, Bd. 30, Nr. 1, 1988, S. 41–52
- [Kra+07] Krallmann, H. (Hrsg.); Schönherr, M. (Hrsg.); Trier, M. (Hrsg.): Systemanalyse im Unternehmen : Prozessorientierte Methoden der Wirtschaftsinformatik. 5., vollst. überarb. Aufl. München, Wien : Oldenbourg, 2007
- [Kün16] Künzel, H. (Hrsg.): Erfolgsfaktor Lean Management 2.0 : Wettbewerbsfähige Verschlinkung auf nachhaltige und kundenorientierte Weise. Berlin, Heidelberg : Springer Berlin Heidelberg, 2016
- [La+10] La, H. J. ; Kim, S. D.: A Service-Based Approach to Designing Cyber Physical Systems. In: 2010 IEEE/ACIS 9th International Conference on Computer and Information Science (ICIS), 2010, S. 895–900
- [Lac+] Lackes, R. ; Siepermann, M.: Stichwort: Schnittstelle : Gabler Wirtschaftslexikon. URL <http://wirtschaftslexikon.gabler.de/Archiv/74916/schnittstelle-v10.html> – Überprüfungsdatum 2017-11-19
- [Lag+10] Lage Junior, M. ; Filho, G. M.: Variations of the kanban system: Literature review and classification. In: Int. J. Production Economics, Bd. 125, Nr. 1, 2010, S. 13–21
- [Lap+14] Lappe, D. ; Veigt, M. ; Franke, M. ; Kolberg, D. ; Schlick, J. ; Stephan, P. ; Guth, P. ; Zimmerling, R.: Vernetzte Steuerung einer schlanken Intralogistik : Simulationsbasierte Potentialanalyse einer bedarfsorientierten Materialversorgung in der Fertigung. In: wt Werkstattstechnik online, Bd. 104, Nr. 3, 2014, S. 112–117
- [Lee08] Lee, E. A.: Cyber Physical Systems: Design Challenges. In: Proceedings of the 11th IEEE Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing, 2008, S. 363–369

- [Lee99] Lee, T. Y.: Information Modeling: From design to implementation. In: Naha-vandi (Hrsg.): Proceedings of the 2nd World Manufacturing Congress (WMC). Millet : ICSC Academic Press, 1999, S. 315–321
- [Leh16] Lehr, G.: Betriebswirtschaftliche Bewertung von Industrie 4.0 Anwendungen in der Lean Production. Kaiserslautern, TU Kaiserslautern, Lehrstuhl für Strategie, Innovation und Kooperation. Bachelorarbeit. 2016
- [Lei09] Leitão, P.: Agent-based distributed manufacturing control : A state-of-the-art survey. In: Engineering Applications of Artificial Intelligence, Bd. 22, Nr. 7, 2009, S. 979–991
- [Lei+08] Leitão, P. ; Restivo, F. J.: Implementation of a Holonic Control System in a Flexible Manufacturing System. In: IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), Bd. 38, Nr. 5, 2008, S. 699–709
- [Lew+13] Lewandowski, M. ; Werthmann, D. ; Gath, M. ; Lawo, M.: Agent-based Control for Material Handling Systems in In-House Logistics : Towards Cyber-Physical Systems in In-House-Logistics Utilizing Real Size Evaluation of Agent-based Material Handling Technology. In: European Conference on Smart Objects, Systems and Technologies (Smart SysTech). 243, 2013, S. 5–9
- [Lik07] Liker, J. K.: Der Toyota-Weg : 14 Managementprinzipien des weltweit erfolgreichsten Automobilkonzerns. 2., unveränd. Aufl. München : FinanzBuch-Verl., 2007
- [Lik+11] Liker, J. K. ; Meier, D. ; Braun, A.: Der Toyota-Weg : Für jedes Unternehmen. 5., unveränd. Aufl. München : FinanzBuch-Verl., 2011
- [Lin+10] Lin, K.-J. ; Panahi, M.: A real-time service-oriented framework to support sustainable cyber-physical systems. In: 2010 8th IEEE International Conference on Industrial Informatics (INDIN), 2010, S. 15–21
- [Lin+15] Lin, S.-W. ; Miller, B. ; Durand, J. ; Joshi, R. ; Didier, P. ; chigani, A. ; Torenbeek, R. ; Duggal, D. ; Martin, R. ; Bleakley, G. ; Kind, A. ; Molina, J. ; Schrecker, S. ; Lembree, R. ; Soroush, H. ; Garbis, J. ; Crawford, M. ; Harper, E. ; Raman, K. ; Witten, B.: Industrial Internet Reference Architecture. URL <http://www.iiconsortium.org/IIRA.htm> – Überprüfungsdatum 2017-11-19
- [Lon93] London Electronics Ltd: Networked Andon Systems with history analysis. URL <https://www.london-electronics.com/andon-displays.php> – Überprüfungsdatum 2017-11-19
- [Loo+07] Loos, P. (Hrsg.); Krcmar, H. (Hrsg.): Architekturen und Prozesse : Strukturen und Dynamik in Forschung und Unternehmen. Berlin, Heidelberg : Springer Verlag, 2007

- [Los13] Loskyll, M.: Entwicklung einer Methodik zur dynamischen kontextbasierten Orchestrierung semantischer Feldgerätefunktionalitäten. Kaiserslautern, TU Kaiserslautern, Lehrstuhl für Produktionsautomatisierung. Dissertation. 2013
- [Lüd+] Lüder, A. ; Schmidt, N.: AutomationML - Fachexperten erklären das Format : Whitepaper. URL <http://www.sps-magazin.de/downloads/WhitepaperAutomationML.pdf> – Überprüfungsdatum 2017-12-13
- [Mäh+95] Mählick, H. (Hrsg.); Pankus, M. (Hrsg.): Herausforderung Lean Production: Wettbewerbsvorsprung der Unternehmenserneuerung. Berlin, Heidelberg : Springer Verlag, 1995
- [Maj10] Majer, F.: Semantisches Informationsmodell für die Betriebsunterstützung dienstorientierter Systeme. Karlsruhe, Karlsruher Institut für Technologie, Fakultät für Informatik. Dissertation. 2010. URL <http://digbib.ubka.uni-karlsruhe.de/volltexte/documents/1281043> – Überprüfungsdatum 2017-11-19
- [man12] manufactus GmbH: e-Kanban Whitepaper. URL <http://www.manufactus.com/e-kanban-white-paper/> – Überprüfungsdatum 2017-12-13
- [Mel10] Melzer, I.: Service-orientierte Architekturen mit Web Services : Konzepte - Standards - Praxis. 4. Aufl. Heidelberg : Spektrum Akademischer Verlag, 2010
- [Mer+16] Mertens, P. ; Barbian, D.: Digitalisierung und Industrie 4.0 – Trend mit modischer Überhöhung? In: Informatik-Spektrum, Bd. 39, Nr. 4, 2016, S. 301–309
- [Mic16] Michaud Cadcz, O.: Andon Droid-Lean Manufacturing. URL <https://play.google.com/store/apps/details?id=com.andon.leanmanufacturing&hl=de&rdid=com.andon.leanmanufacturing> – Überprüfungsdatum 2017-12-07
- [Mil+98] Miller, T. D. ; Elgård, P. E.: Defining Modules, Modularity and Modularization : Evolution of the Concept in a Historical Perspective. In: Design for integration in manufacturing : Proceedings of the thirteenth IPS Seminar. Aalborg : Dept. of Production, Aalborg University, 1998
- [Möh+10] Möhrmann, U. ; Itasse, S.: Elektronisches Kanban sichert Versorgung in Montagelinien : E-Kanban. URL <http://www.maschinenmarkt.vogel.de/themenkanale/materialflusslogistik/logistiksoftware/articles/296342/> – Überprüfungsdatum 2017-12-17
- [Mou+12] Mourtzis, D. ; Doukas, M.: Decentralized manufacturing systems review : Challenges and outlook. In: Logistics research, Bd. 5, Nr. 3/4, 2012, S. 113–121
- [MQT] MQTT Community: Frequently Asked Questions : MQTT. URL <http://mqtt.org/faq> – Überprüfungsdatum 2017-03-03

- [MTC14a] MTConnect Institute: MTConnect Standard Part 1 - Overview and Protocol. 2014-09-30. URL http://static1.squarespace.com/static/54011775e4b0bc1fe0fb8494/t/557f2897e4b04b2acdba80b5/1434396823825/mtc_part_1_overview_v1.3.pdf – Überprüfungsdatum 2017-12-17
- [MTC14b] MTConnect Institute: MTConnect Standard Part 2 - Device Information Model. 2014-09-30. URL http://static1.squarespace.com/static/54011775e4b0bc1fe0fb8494/t/557f28c6e4b022d40310f21a/1434396870272/mtc_part_2_components_v1.3.pdf – Überprüfungsdatum 2017-12-17
- [Neu07] Neuhaus, R.: Produktionssysteme : Entstehung - Aufbau - Implementierung. Bergisch Gladbach : Heider, 2007
- [Nok] Nokleby, C.: PackML Unit/Machine Implementation Guide : Part 1: PackML Interface State Manager. URL http://omac.org/wp-content/uploads/2016/11/PackML_Unit_Machine_Implementation_Guide-V1-00.pdf – Überprüfungsdatum 2017-11-12
- [Nyh08] Nyhuis, P. (Hrsg.): Wandlungsfähige Produktionssysteme : Heute die Industrie von morgen gestalten. Garbsen : PZH, Produktionstechn. Zentrum, 2008
- [OAS14] OASIS: MQTT Version 3.1.1. 2014-10-29. URL <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html> – Überprüfungsdatum 2017-03-05
- [Obj15] Object Modelling Group (OMG): Data Distribution Service (DDS). Object Modelling Group (OMG), 2015-04-10. URL <http://www.omg.org/spec/DDS/1.4> – Überprüfungsdatum 2017-12-13
- [Ohn93] Ohno, T.: Das Toyota-Produktionssystem. Frankfurt am Main, New York : Campus Verlag, 1993
- [OPC12a] OPC Foundation: OPC Unified Architecture Specification Part 3: Address Space Model. 2012-07-10. URL <https://opcfoundation.org/developer-tools/specifications-unified-architecture/part-3-address-space-model/> – Überprüfungsdatum 2016-09-28
- [OPC13] OPC Foundation: ISA-95 Common Information Model For OPC UA. 2013-11-06. URL <https://opcfoundation.org/developer-tools/specifications-unified-architecture/isa-95-common-object-model/> – Überprüfungsdatum 2016-09-29
- [OPC+12b] OPC Foundation; MTConnect Institute: MTConnect-OPC UA Companion Specification. 2012-11-08. URL <http://static1.squarespace.com/static/54011775e4b0bc1fe0fb8494/t/5581c564e4b0f8af8806da42/1434568036828/MTConnect->

- OPC-UA-Companion-Specification_v1.0.pdf – Überprüfungsdatum 2017-11-12
- [OPC12c] OPC Foundation: OPC Unified Architecture Specification Part 1: Overview and Concepts. 2012-07-10. URL <https://opcfoundation.org/developer-tools/specifications-unified-architecture/part-1-overview-and-concepts/> – Überprüfungsdatum 2017-11-12
- [OPC12d] OPC Foundation: OPC Unified Architecture Specification Part 5: Information Model. 16.07.2012. URL <https://opcfoundation.org/developer-tools/specifications-unified-architecture/part-1-overview-and-concepts/> – Überprüfungsdatum 2017-11-12
- [OPC15] OPC Foundation: Unified Architecture. URL <https://opcfoundation.org/about/opc-technologies/opc-ua/> – Überprüfungsdatum 2017-11-19
- [OPC+10] OPC Foundation; PLCopen: OPC UA Information Model for IEC 61131-3. 2010-03-24. URL http://www.plcopen.org/pages/tc4_communication/downloads/plcopen_opcua_information_model_%20v100.pdf – Überprüfungsdatum 2016-09-29
- [Pau16] Paul, H.: Industrie 4.0 : Annäherung an ein Konzept. In: Forschung aktuell, Bd. 2016, Nr. 05, 2016
- [Paw14] Pawellek, G.: Ganzheitliche Fabrikplanung : Grundlagen, Vorgehensweise, EDV-Unterstützung. 2. Aufl. Berlin, Heidelberg : Springer Vieweg, 2014
- [Pic+94] Picot, A. ; Maier, M.: Ansätze der Informationsmodellierung und ihre betriebswirtschaftliche Bedeutung. In: Schmalenbachs Zeitschrift für betriebswirtschaftliche Forschung, Bd. 46, 1994, S. 107–126
- [Pir+13] Pirvu, B.-C. ; Schlick, J. ; Hodek, S. ; Zühlke, D.: Conceptual Overview of a Smart-Factory Architecture. In: Proceedings of the 6th International Conference on Manufacturing Science and Education (MSE), 2013, S. 179–182
- [Pla] Plattform Industrie 4.0: Was ist Industrie 4.0. URL <http://www.plattform-i40.de/I40/Navigation/DE/Industrie40/WasIndustrie40/was-ist-industrie-40.html;jsessionid=D7ED4118DED525F176F382A8F939369D> – Überprüfungsdatum 2017-12-13
- [Poh+11] Pohl, K. ; Rupp, C.: Basiswissen Requirements Engineering : Aus- und Weiterbildung zum "Certified Professional for Requirements Engineering". 3., korrig. Aufl. Heidelberg : dpunkt-Verl., 2011
- [Pos07] Poslad, S.: Specifying protocols for multi-agent systems interaction. In: ACM Transactions on Autonomous and Adaptive Systems, Bd. 2, Nr. 4, 2007, S. 15–24

- [Pra+03] Pras, A. ; Schoenwaelder, J.: On the Difference between Information Models and Data Models : Main Result of the 8th workshop of the Network Management Research Group (NMRG) of the Internet Research Task Force (IRTF) hosted by the University of Texas at Austin. URL <http://www.rfc-editor.org/rfc/pdf/rfc3444.txt.pdf> – Überprüfungsdatum 2017-11-12
- [Pro] Projektkonsortium Cyber-Physische Produktionssysteme (CyProS): Cyber-Physische Produktionssysteme (CyProS) : Produktivitäts- und Flexibilitätssteigerung durch die Vernetzung intelligenter Systeme in der Fabrik. URL <http://www.projekt-cypros.de/> – Überprüfungsdatum 2017-11-19
- [Rat01] Rational Software Corporation: Artifact: Reference Architecture. URL http://sce.uhcl.edu/helm/RationalUnifiedProcess/process/artifact/ar_refarch.htm – Überprüfungsdatum 2017-11-19
- [Rei+13] Reinhart, G. ; Engelhardt, P. ; Geiger, F. ; Philipp, T. R. ; Wahlster, W. ; Zühlke, D. ; Schlick, J. ; Becker, T. ; Lückelt, M. ; Pirvu, B.-C. ; Stephan, P. ; Hodek, S. ; Scholz-Reiter, B. ; Thoben, K.-D. ; Gorltd, C. ; Hribernik, K. A. ; Lappe, D. ; Veigt, M.: Cyber-Physische Produktionssysteme : Produktivitäts- und Flexibilitätssteigerung durch die Vernetzung intelligenter Systeme in der Fabrik. In: wt Werkstattstechnik online, Bd. 103, Nr. 2, 2013, S. 84–89
- [Rei+15] Reinhart, G. (Hrsg.); Scholz-Reiter, B. (Hrsg.); Wahlster, W. (Hrsg.); Zühlke, D. (Hrsg.): Intelligente Vernetzung in der Fabrik : Industrie 4.0 Umsetzungsbeispiele für die Praxis. Stuttgart : Fraunhofer Verlag, 2015
- [Rem+11] Rempp, G. ; Akermann, M. ; Löffler, M. ; Lehmann, J.: Model Driven SOA : Anwendungsorientierte Methodik und Vorgehen in der Praxis. Berlin, Heidelberg : Springer-Verlag Berlin Heidelberg, 2011
- [Reu+08] Reuter, M. ; Zacher, S.: Regelungstechnik für Ingenieure : Analyse, Simulation und Entwurf von Regelkreisen. 12., korrig. und erw. Aufl. Wiesbaden : Vieweg+Teubner / GWV Fachverlage, 2008
- [Rot16] Roth, A. (Hrsg.): Einführung und Umsetzung von Industrie 4.0 : Grundlagen, Vorgehensmodell und Use Cases aus der Praxis. Berlin : Springer Gabler, 2016
- [Rup+07] Rupp, C. ; Queins, S. ; Zengler, B.: UML 2 glasklar : Praxiswissen für die UML-Modellierung. 3., aktual. Aufl. München : Hanser, 2007
- [Sal16] Salt Solutions AG: SALT Add-Ons für die Produktion.
- [San+12] Sanislav, T. ; Miclea, L.: An agent-oriented approach for cyber-physical system with dependability features. In: 2012 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR 2012), 2012, S. 356–361
- [Sch02] Scheer, A.-W.: ARIS - Vom Geschäftsprozess zum Anwendungssystem. Berlin, Heidelberg : Springer Verlag, 2002

- [Sch13a] Scheer, A.-W.: Wirtschaftsinformatik : Referenzmodelle für industrielle Geschäftsprozesse. 7., Aufl. Berlin : Springer Berlin, 2013
- [Sch93] Scheller, A.: Informationsmodellierung für verteilte Anwendungen auf Basis standardisierter Datenmodelle. München : Oldenbourg, 1993
- [Sch94] Schling, F.: Montage zwischen Lean und Automation - der Vorwerk-Weg. In: Milberg; Reinhart; Milberg (Hrsg.): Münchner Kolloquium '94 : Der Weg zu Wettbewerbsfähigkeit und Standortsicherung. Landsberg/Lech : Verl. Moderne Industrie, 1994, S. 289–301
- [Sch16] Schmitt, M.: Entwicklung einer Referenzarchitektur zur dynamischen Adaption von Feldgerätefunktionalitäten mittels modularer Softwareanwendungen. Kaiserslautern, TU Kaiserslautern, Lehrstuhl für Produktionsautomatisierung. Dissertation. 2016
- [Sch17] Schmitz, P.: Konzeption und Realisierung einer digitalen Shopfloor-Tafel für die SmartFactory-KL. Kaiserslautern, TU Kaiserslautern, Lehrstuhl für Werkzeugmaschinen und Steuerungen. Forschungsprojekt. 2017
- [Sch06] Schnell, G. (Hrsg.): Bussysteme in der Automatisierungs- und Prozesstechnik : Grundlagen, Systeme und Trends der industriellen Kommunikation. 6., überarb. und aktual. Aufl. Wiesbaden : Vieweg, 2006
- [Sch+07] Scholz-Reiter, B. ; Freitag, M.: Autonomous Processes in Assembly Systems. In: CIRP Annals - Manufacturing Technology, Bd. 56, Nr. 2, 2007, S. 712–729
- [Sch13b] Schreiber, S.: Entwicklung einer Vergleichs- und Bewertungsmöglichkeit von dezentralen Steuerungsarchitekturen für Produktionssysteme. Hamburg, Universität der Bundeswehr Hamburg, Fakultät für Maschinenbau der Helmut-Schmidt-Universität. Dissertation. 2013
- [Sch+15] Schubel, A. ; Seel, C. ; Schneider, M.: Informationsmodelle für die Produktions- und Logistikplanung : Eine Literaturanalyse des aktuellen Referenzmodellbestands. In: Thomas (Hrsg.): Wirtschaftsinformatik Proceedings, 2015
- [Sel10] Selig, A.: Informationsmodell zur funktionalen Typisierung von Automatisierungsgeräten. Stuttgart, Universität Stuttgart, Institut für Steuerungstechnik der Werkzeugmaschinen und Fertigungseinrichtungen, Fraunhofer IPA. Dissertation. 2010
- [Sen13] Sandler, U. (Hrsg.): Industrie 4.0 : Beherrschung der industriellen Komplexität mit SysLM. Berlin, Heidelberg : Springer Berlin Heidelberg, 2013
- [Sha+07] Shah, R. ; Ward, P. T.: Defining and developing measures of lean production. In: Journal of Operations Management, Bd. 25, Nr. 4, 2007, S. 785–805

- [She+06] Shen, W. ; Hao, Q. ; Yoon, H. J. ; Norrie, D. H.: Applications of agent-based systems in intelligent manufacturing : An updated review. In: Advanced Engineering Informatics, Bd. 20, Nr. 4, 2006, S. 415–431
- [Spa03] Spath, D. (Hrsg.): Ganzheitlich produzieren : Innovative Organisation und Führung. Stuttgart : LOG_X Verl., 2003
- [Spi16] Spinnarke, S.: OPC UA wird (neben anderen) Industrie 4.0-Standard. In: Produktion.de, 2016
- [Sta73] Stachowiak, H.: Allgemeine Modelltheorie. Vienna : Springer Vienna, 1973
- [Ste+15] Stephan, P. ; Schlick, J.: Reduzierung organisatorischer Verluste durch den Einsatz von Cyber-Physischen-Systemen in der Zahnradfertigung der WITTENSTEIN bastian GmbH. In: Reinhart; Scholz-Reiter; Wahlster; Zühlke (Hrsg.): Intelligente Vernetzung in der Fabrik : Industrie 4.0 Umsetzungsbeispiele für die Praxis. Stuttgart : Fraunhofer Verlag, 2015, S. 363–385
- [Str96] Strahinger, S.: Metamodellierung als Instrument des Methodenvergleichs : Eine Evaluierung am Beispiel objektorientierter Analysemethoden. Aachen, TH Darmstadt. Dissertation. 1996
- [Tak04] Takeda, H.: LCIA - Low Cost Intelligent Automation : Produktivitätsvorteile durch Einfachautomatisierung. 1.Aufl. München : mi-Fachverlag/ Redline, 2004
- [Tak06] Takeda, H.: The synchronized Production System : Going beyond Just-in-Time through Kaizen. London : KoganPage, 2006
- [Tak12] Takeda, H.: Das synchrone Produktionssystem : Just-in-time für das ganze Unternehmen. 7. Aufl. München : Mi-Wirtschaftsbuch, 2012
- [Tay12] Taylor, F. W.: The Principles of Scientific Management. Newburyport : Dover Publications, 2012
- [Tec17a] Technische Universität München: Weihenstephaner Standards. URL <http://weihenstephaner-standards.de/index.php?id=5> – Überprüfungsdatum 2017-03-20
- [Tec17b] Technologie-Initiative SmartFactory KL e.V.: SmartFactoryKL. URL <http://www.smartfactory-kl.de/> – Überprüfungsdatum 2017-06-25
- [Teu99] Teubner, R. A.: Organisations- und Informationssystemgestaltung : Theoretische Grundlagen und integrierte Methoden. Wiesbaden : Deutscher Universitätsverlag, 1999
- [The] The Eclipse Foundation: Mosquitto : An Open Source MQTT v3.1/v3.1.1 Broker. URL <https://mosquitto.org/> – Überprüfungsdatum 2017-08-30

- [The16] The Industrial Internet Consortium: The Industrial Internet Consortium: A global not-for-profit Partnership of Industry, Government and Academia. URL <http://www.iiconsortium.org/about-us.htm> – Überprüfungsdatum 2017-11-19
- [the15] the International Society of Automation: ISA 88, Batch Control. URL <https://www.isa.org/isa88/> – Überprüfungsdatum 2014-10-23
- [The+17] Theorin, A. ; Bengtsson, K. ; Provost, J. ; Lieder, M. ; Johnsson, c. ; Lundholm, T. ; Lennartson, B.: An event-driven manufacturing information system architecture for Industry 4.0. In: International Journal of Production Research, Bd. 55, Nr. 5, 2017, S. 1–15
- [Twi17] Twitter Inc.: Twitter Developer Documentation : API Overview. URL <https://dev.twitter.com/overview/api> – Überprüfungsdatum 2017-02-27
- [Val+94] Valckenaers, P. ; Bonneville, F. ; van Brussel, H. ; Bongaerts, L. ; Wyns, J.: Results of the holonic control system benchmark at KU Leuven. In: 4th International Conference on Computer Integrated Manufacturing and Automation Technology, 1994, S. 128–133
- [Van+98] Van Brussel, H. ; Wyns, J. ; Valckenaers, P. ; Bongaerts, L. ; Peeters, P.: Reference architecture for holonic manufacturing systems : PROSA. In: Computers in Industry, Bd. 37, Nr. 3, 1998, S. 255–274
- [van96] van Dyke Parunak, H.: Applications of distributed artificial intelligence in industry. In: O'Hare; Jennings (Hrsg.): Foundations of Distributed Artificial Intelligence. New York, USA : John Wiley & Sons, 1996, S. 139–164
- [Ver90] Verein Deutscher Ingenieure e.V.: Richtlinie 2860. Montage- und Handhabungstechnik: Handhabungsfunktionen, Handhabungseinrichtungen; Begriffe, Definitionen, Symbole. Berlin : Beuth Verlag, 1990-05
- [Ver12] Verein Deutscher Ingenieure e.V.: Richtlinie 2870 Blatt 1. Ganzheitliche Produktionssysteme - Grundlagen, Einführung und Bewertung. Berlin : Beuth Verlag, 2012-07
- [Ver13] Verein Deutscher Ingenieure e.V.: Richtlinie 2870 Blatt 2. Ganzheitliche Produktionssysteme - Methodenkatalog. Berlin : Beuth Verlag, 2013-02
- [Vog+14] Vogel-Heuser, B. (Hrsg.); Bauernhansl, T. (Hrsg.); ten Hompel, M. (Hrsg.): Industrie 4.0 in Produktion, Automatisierung und Logistik : Anwendung, Technologien und Migration. Wiesbaden : Springer Vieweg, 2014
- [Vog+15] Vogel-Heuser, B. ; Schölder, T. ; Pröll, S. ; Jeschke, S. ; Ewert, D. ; Niggemann, O. ; Windmann, S. ; Berger, U. ; Lehmann, C.: Agentenbasierte cyberphysische Produktionssysteme : Anwendungen für Industrie 4.0, Bd. 57, Nr. 09, 2015, S. 36–45

- [War93] Warnecke, H.-J.: Der Produktionsbetrieb 2 : Produktion, Produktionssicherung. 2., neubearb. Auflage. Berlin, Heidelberg : Springer Berlin Heidelberg, 1993
- [Web06] Weber, R.: Kanban-Einführung : Das effiziente, kundenorientierte Logistik- und Steuerungskonzept für Produktionsbetriebe. 4., überarb. Aufl. Renningen : expert-Verl., 2006
- [Weg98] Wegner, P.: Interactive foundations of computing. In: Theoretical Computer Science, Bd. 192, Nr. 2, 1998, S. 315–351
- [WEI] WEIGANG-Baden-Elsass-GmbH: Visualisieren. URL <http://www.weingang-be.de/visualisierung.html> – Überprüfungsdatum 2017-11-19
- [Wei08] Weikiens, T.: Systems Engineering with SysML/UML : Modeling, Analysis, Design. 1. Aufl. s.l. : Elsevier professional, 2008
- [Wei91] Weiser, M.: The computer for the 21st century. In: Scientific American, Bd. 265, Nr. 3, 1991, S. 94–104
- [Wei+06] Weiß, G. ; Jakob, R.: Agentenorientierte Softwareentwicklung. Dordrecht : Springer, 2006
- [Wel+11] Wellenreuther, G. ; Zastrow, D.: Automatisieren mit SPS - Theorie und Praxis. 5., korrig. und erweiter. Aufl. Wiesbaden : Vieweg + Teubner, 2011
- [Wes+01] Westerinen, A. ; Schnizlein, J. ; Strassner, J. ; Scherling, M. ; Quinn, B. ; Herzog, S. ; Huynh, A. ; Carlson, M. ; Perry, J. ; Waldbusser, S.: Terminology for Policy-Based Management. URL <https://tools.ietf.org/html/rfc3198> – Überprüfungsdatum 2017-03-20
- [Wey+17] Weyer, S. ; Quint, F. ; Fischer, S. ; Gorecky, D. ; Zühlke, D.: Die SmartFactory für individualisierte Kleinserienfertigung. In: Reinhart (Hrsg.): Handbuch Industrie 4.0 : Geschäftsmodelle, Prozesse, Technik. München : Hanser Verlag, 2017, S. 691–708
- [Wie14a] Wiendahl, H.-P.: Betriebsorganisation für Ingenieure : Mit 3 Tabellen. 8., überarb. Aufl. München : Hanser, 2014
- [Wie+14b] Wiesner, S. ; Gorltd, C. ; Soeken, M. ; Thoben, K. D. ; Drechsler, R.: Requirements Engineering for Cyber-Physical Systems : Challenges in the Context of "Industrie 4.0". In: Advances in Production Management Systems (IFIP Advances in Information and Communication Technology), 2014, S. 281–288
- [Wik16a] Wikipedia, Die freie Enzyklopädie: Architektur. URL <https://de.wikipedia.org/w/index.php?title=Architektur&oldid=160809292> – Überprüfungsdatum 2017-12-13

- [Wik16b] Wikipedia, Die freie Enzyklopädie: ISA-88. URL <https://de.wikipedia.org/w/index.php?title=ISA-88&oldid=153834168> – Überprüfungsdatum 2017-12-13
- [Wik16c] Wikipedia, Die freie Enzyklopädie: OSI-Modell. URL <https://de.wikipedia.org/w/index.php?title=OSI-Modell&oldid=159528643> – Überprüfungsdatum 2017-02-18
- [Wil13] Wilfling, S.: Management organisationaler Anpassungsprozesse : Transformationsvorhaben durch geeignete Informationsbasis erfolgreich planen und umsetzen. Wiesbaden : Springer, 2013
- [Wom+07] Womack, J. P. ; Jones, D. T. ; Roos, D.: The machine that changed the world : [the story of lean production ; Toyota's secret weapon in the global car wars that is revolutionizing world industry]. 1. Aufl. New York, USA : Free Press, 2007
- [Wom+92] Womack, J. P. ; Roos, D. ; Jones, D. T.: Die zweite Revolution in der Autoindustrie : Konsequenzen aus der weltweiten Studie aus dem Massachusetts Institute of Technology. 5. Aufl. Frankfurt a.M. : Campus-Verl., 1992
- [Wür13] Würth Industrie Service GmbH & Co. KG: iBin - Bestände im Blick : iNFORMATIV. iNNOVATIV. iNTELLIGENT. URL http://www.wuerth-industrie.de/web/de/wuerthindustrie/cteilemanagement_5/kanban/die_revolution_im_c_teile_management__behaelter_ibin_1/ibin_kanbanbehaelter.php – Überprüfungsdatum 2017-12-13
- [Xin15] Xinhua: 'Made in China 2025' plan unveiled. In: China Daily 2015. URL http://www.chinadaily.com.cn/bizchina/2015-05/19/content_20760528.htm – Überprüfungsdatum 2016-09-20
- [Yam+17] Yamazaki, Y. ; Takata, S. ; Onari, H. ; Kojima, F. ; Kato, S.: Lean Automation System Responding to the Changing Market. In: Factories of the Future in the digital environment - Proceedings of the 49th CIRP Conference on Manufacturing Systems, 2017, S. 201–206
- [Zaf13] Zafarzadeh, M.: A Guideline for Efficient Implementeation of Automation in Lean Manufacturing Environment. Västerås, Mälardalens Högskola Eskilstuna Västerås. Masterarbeit. 2013. URL <http://mdh.diva-portal.org/smash/get/diva2:640154/FULLTEXT01.pdf> – Überprüfungsdatum 2017-11-19
- [Zim+15] Zimmerling, R. ; Hölscher, J. H.: Intelligente Intralogistik zur bedarfsgerechten Materialversorgung der Produktion vernetzt mit der Transportlogistik. In: Reinhart; Scholz-Reiter; Wahlster; Zühlke (Hrsg.): Intelligente Vernetzung in

der Fabrik : Industrie 4.0 Umsetzungsbeispiele für die Praxis. Stuttgart : Fraunhofer Verlag, 2015, S. 353–360

[Züh10] Zühlke, D.: SmartFactory - Towards a Factory-of-Things. In: IFAC annual Reviews in control, 2010, S. 129–138

[Züh+11] Zühlke, D. ; Ollinger, L.: Agile Automation Systems Based on Cyber-Physical Systems and Service-Oriented Architectures. In: Lee (Hrsg.): Advances in Automation and Robotics : Selected Papers from the 2011 International Conference on Automation and Robotics (ICAR 2011), 2011, S. 567–574

Anhang A – Anwendungsfälle der Lean-Methoden

Nachstehende Tabelle listet alle 60 definierten Anwendungsfälle der berücksichtigten Lean-Methoden auf:

Tabelle 29 Anwendungsfälle je Lean-Methode

ID	Lean-Methode	Anwendungsfall
1.1	5S	5S durchführen
		5S kontrollieren
1.2	Prozessstandardisierung	Prozess standardisieren
		Prozess auslesen
		Prozessaudit durchführen
2.2	8D-Report	8D-Report erstellen
		8D-Report auslesen
2.4	Autonomation	Anlage identifiziert Fehler
2.5	Ishikawa-Diagramm	Ishikawa-Diagramm anlegen
2.6	Kurze Regelkreise	Problem gemäß Eskalationsstufe eskalieren
		Eskalationsstufen aktualisieren
2.7	Poka Yoke	Fehlerverhinderung umsetzen
2.8	Six Sigma	DMAIC-Zyklus durchführen
		Six Sigma Kennzahlen auslesen
2.9	Statistische Prozessregelung	Qualitätsparameter des Prozesses überwachen
2.10	Werkerselbstkontrolle	Produktqualität prüfen
		Prüfungsanweisung aktualisieren
3.1	Andon	Status auslesen
		Kennzahlen abrufen
		Störung bearbeiten
		Soll-Kennzahlen aktualisieren
		Störungen auslesen
3.2	Shopfloor Management	Shopfloor-Regeltermin durchführen
3.3	Gemba	Gemba-Walk durchführen
4.1	Audit	Audit durchführen
4.2	Benchmarking	Benchmarking durchführen
4.4	Ideenmanagement	Ideen sammeln
		Ideen auswählen und umsetzen
		Ideenstatus aktualisieren
4.5	PDCA	PDCA durchführen
4.6	Bottleneck-Analyse	Bottleneck-Analyse durchführen
4.7	Prozess- & Performancemanagement	Relevante Kennzahlen auslesen
		Prozessbeschreibung liefern
		Prozess aufnehmen
6.1	First in First Out	Kanban priorisieren
6.2	One Piece Flow	Losgrößen verkleinern
		Losgrößen auslesen
		Meldebestand auslesen
6.3	SMED	Rüstvorgang durchführen
		Rüstzeiten auslesen
6.4	Wertstromplanung	Wertstromanalyse durchführen
		Wertstromdesign durchführen
6.6	Taktung	Linientakt anpassen
		Taktzeit auslesen
		Taktzeitüberschreitung melden
		Takt anzeigen
7.1	JiT/ JiS	Abrufsystematik umstellen
		Verzug melden
7.2	Kanban	Kundenauftrag bearbeiten
		Kanban bearbeiten
		Kanban aktualisieren
		Kanban auslesen
7.3	Milkrun	Milkrun durchführen
7.4	Nivellierung	Aufträge nivellieren
		Tagesproduktion glätten
		Zyklenzahl erhöhen
7.5	Supermarkt	Lagerflächen auslesen
		Lagerflächen aktualisieren
		Nachbestellung auslösen
8.4	Verschwendungsbewertung	Verschwendungsbewertung durchführen

Anhang B – Aktivitätsdiagramme der Lean-Methoden

Nachstehend sind für eine Auswahl an Lean-Methoden die erstellten Aktivitätsdiagramme abgebildet.

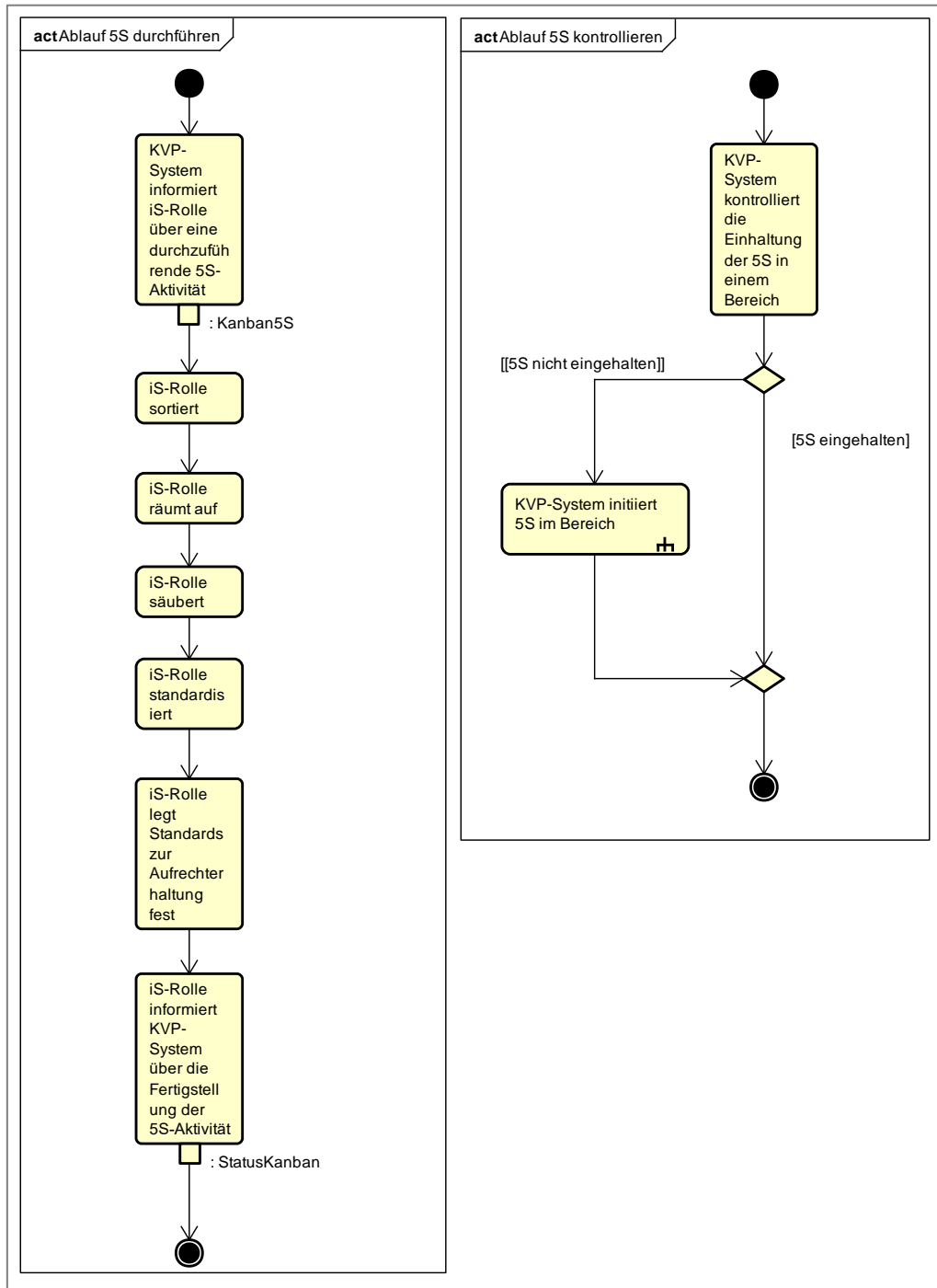


Abbildung 85 Aktivitätsdiagramme der 5S-Methode

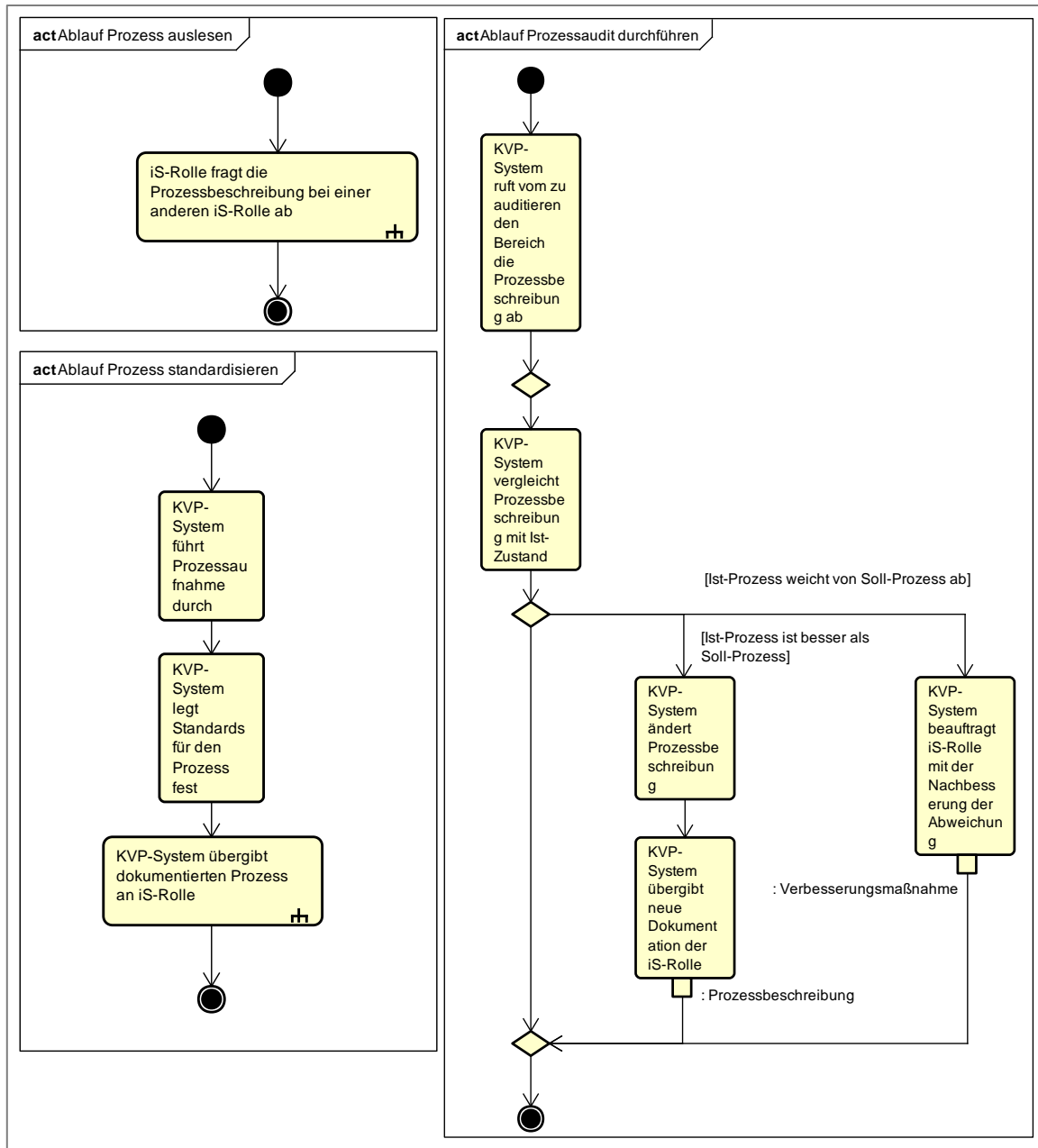


Abbildung 86 Aktivitätsdiagramme der Prozessstandardisierungs-Methode

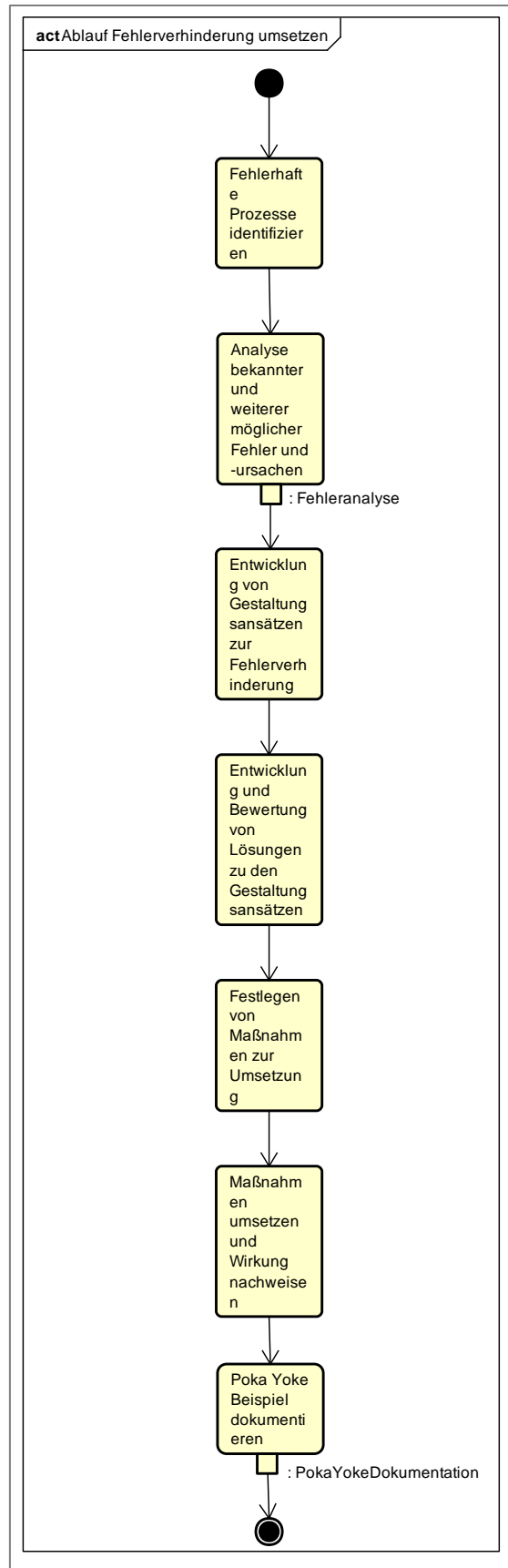


Abbildung 87 Aktivitätsdiagramm der Poka Yoke-Methode

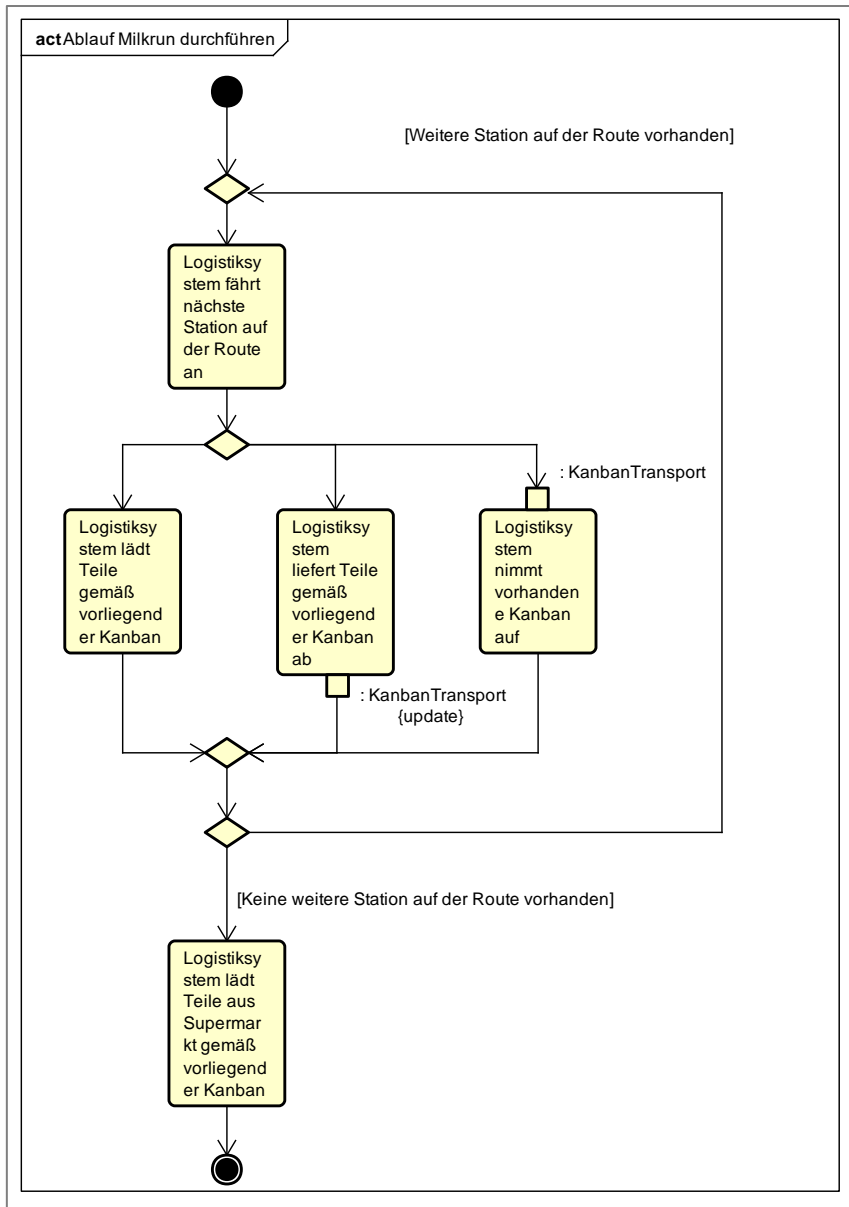


Abbildung 88 Aktivitätsdiagramme der Milkrun-Methode

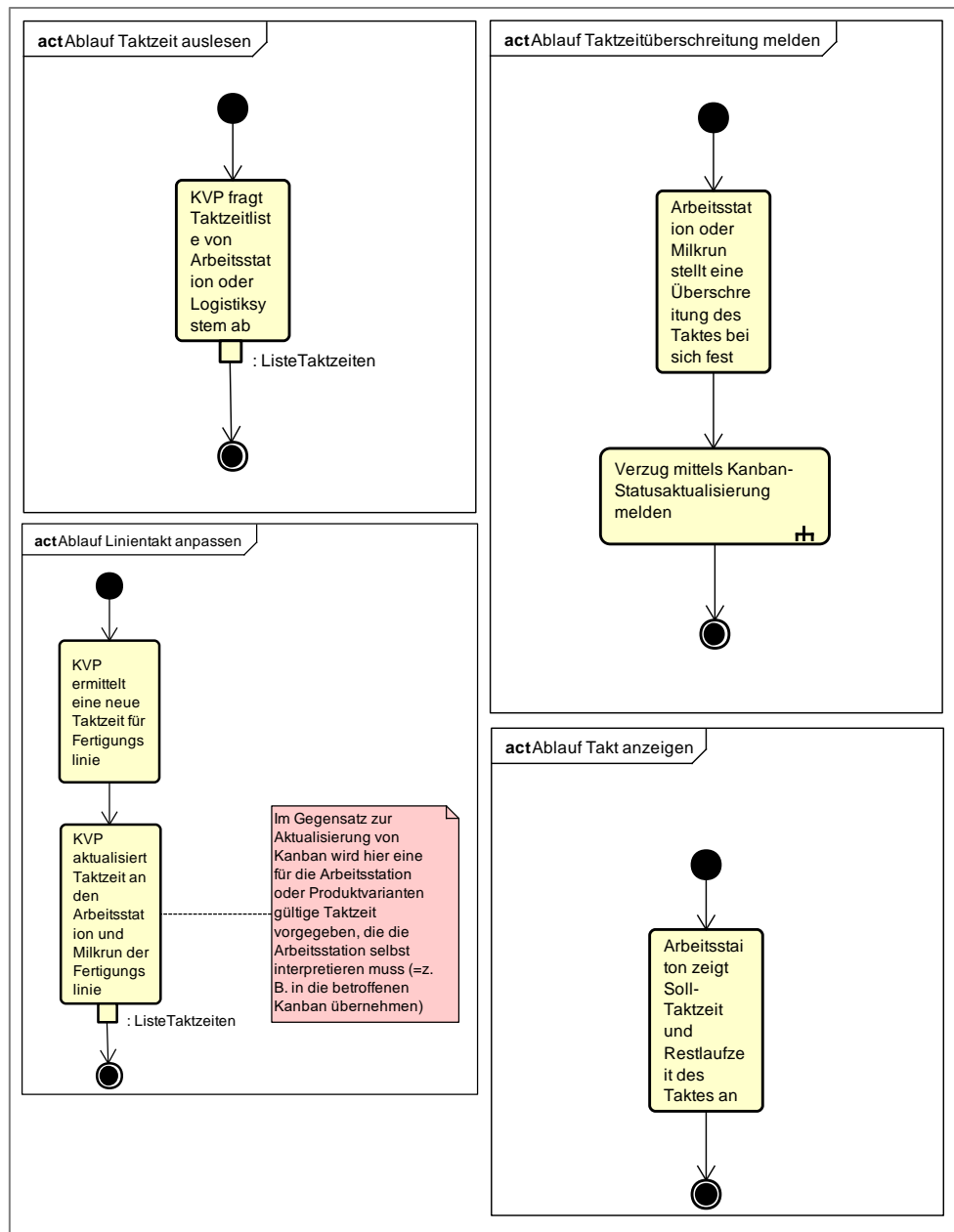


Abbildung 89 Aktivitätsdiagramme der Taktungs-Methode

Anhang C – Kommunikationsdiagramme der Lean-Methoden

Nachstehend sind für eine Auswahl an Lean-Methoden die erstellten Kommunikationsdiagramme abgebildet.

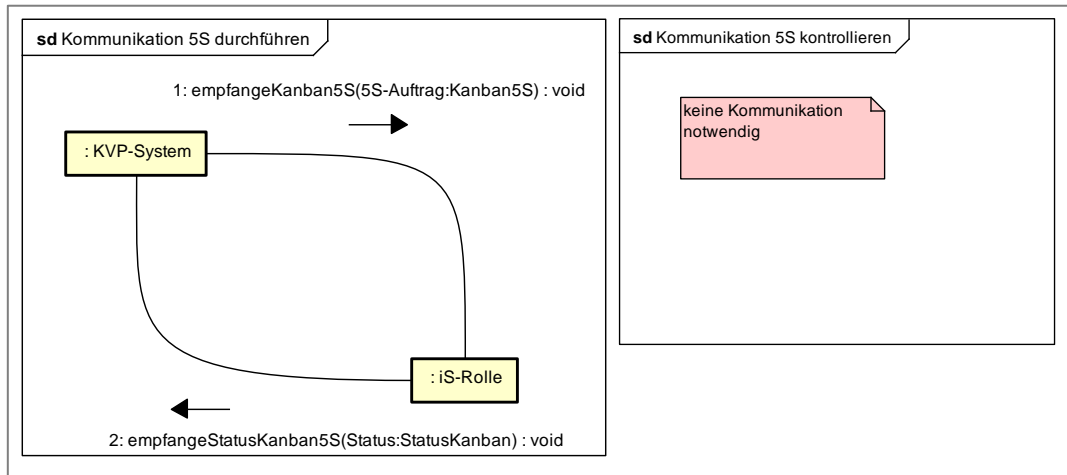


Abbildung 90 Kommunikationsdiagramme der 5S-Methode

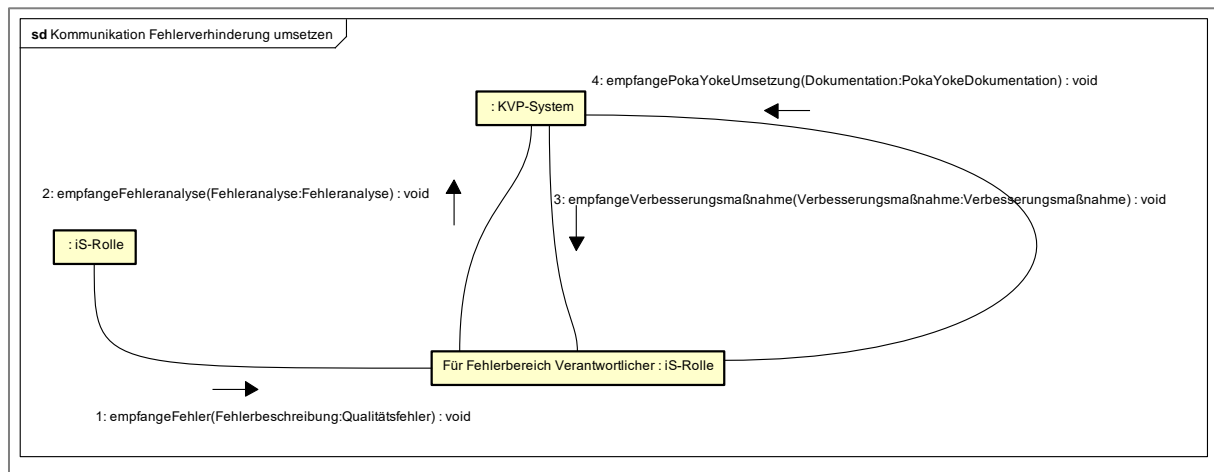


Abbildung 91 Kommunikationsdiagramm der Poka Yoke-Methode

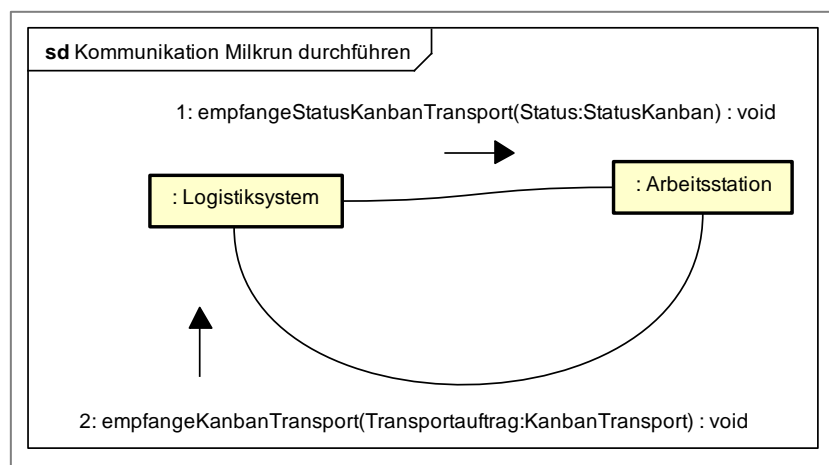


Abbildung 92 Kommunikationsdiagramm der Milkrun-Methode

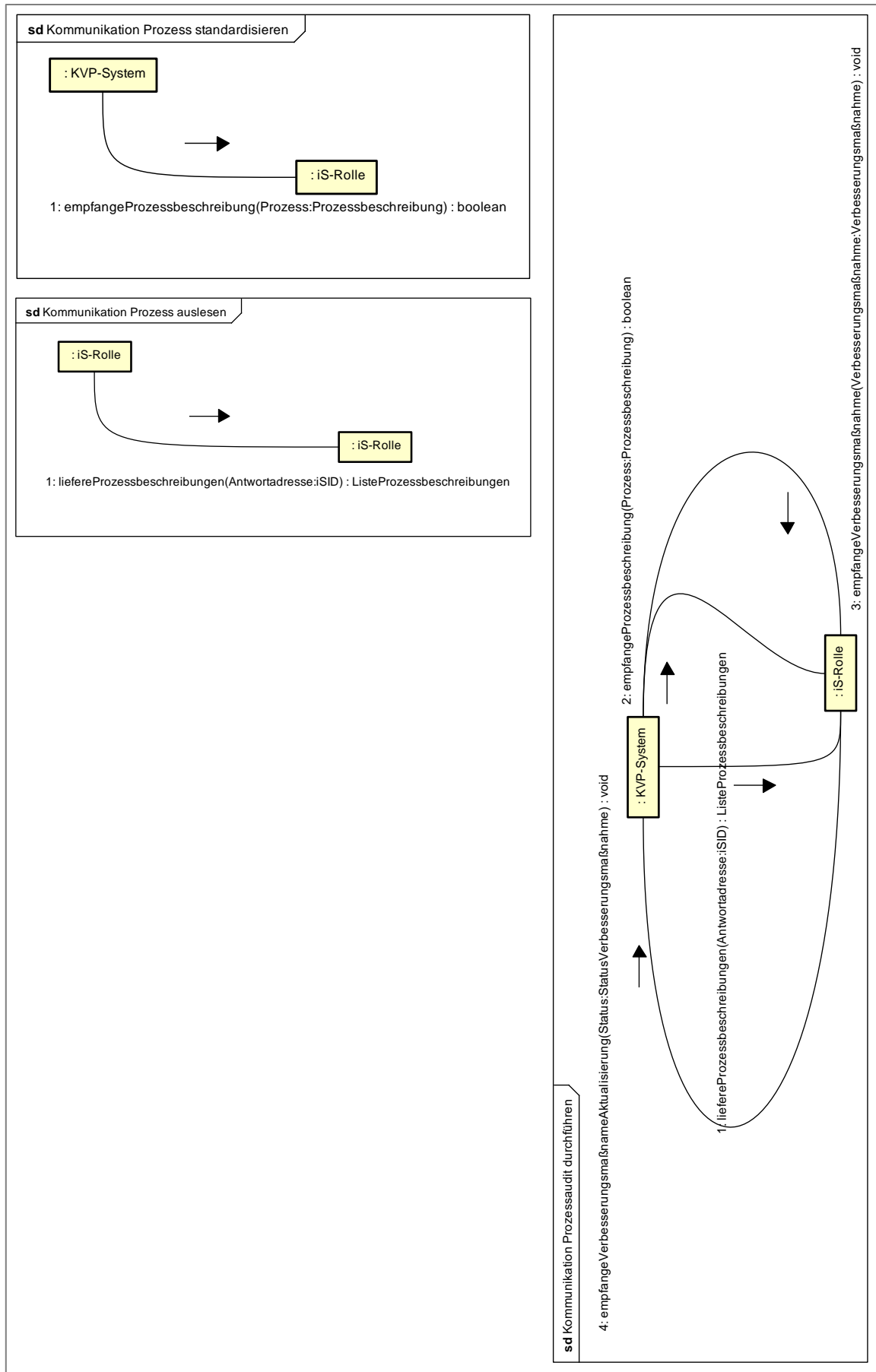


Abbildung 93 Kommunikationsdiagramme der Prozessstandardisierungs-Methode

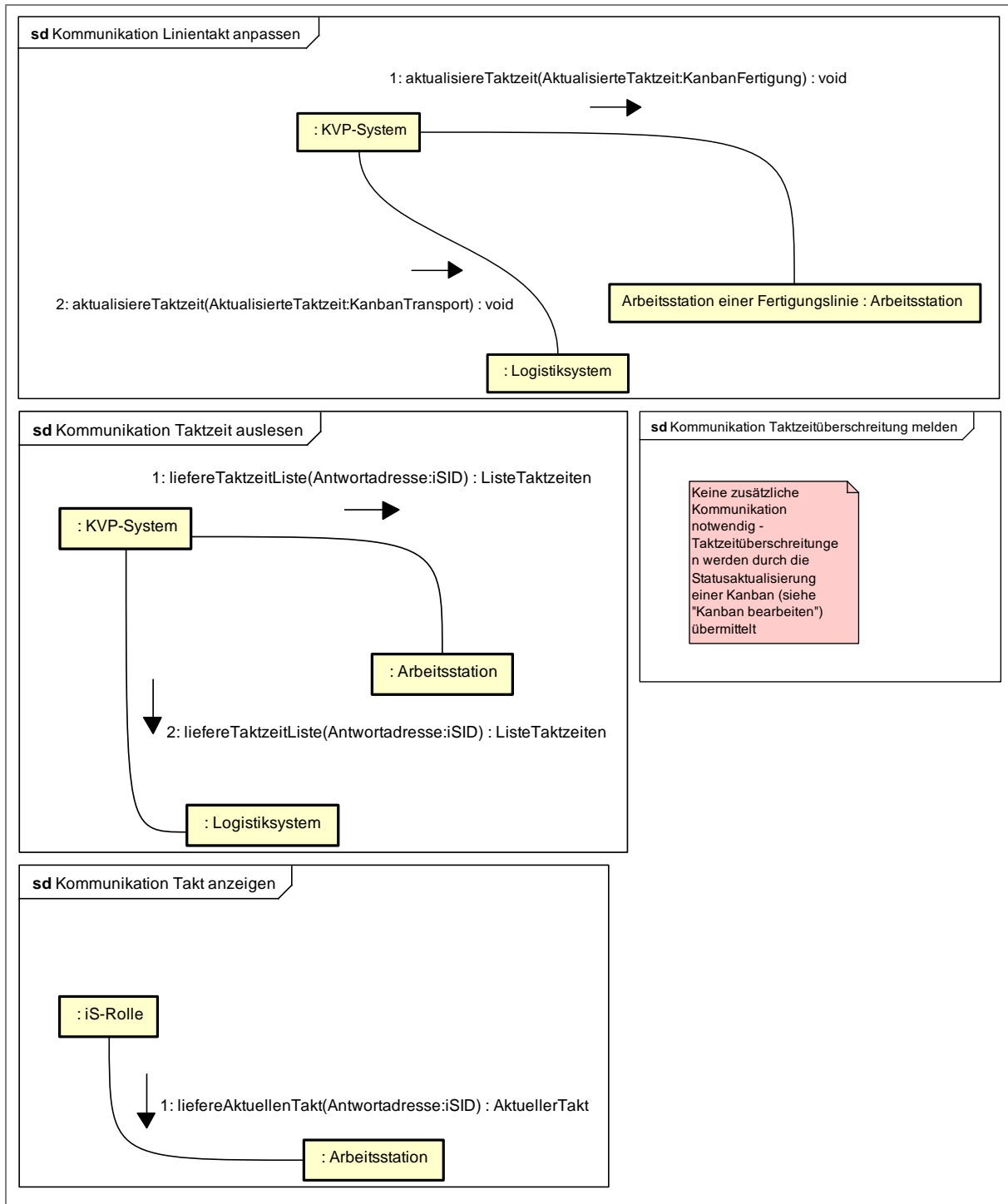


Abbildung 94 Kommunikationsdiagramm der Taktungs-Methode

Anhang D – Liste der funktionalen Anforderungen

Nachstehende Tabelle listet die 97 identifizierten, aus den Modellen der Lean-Methoden abgeleiteten funktionalen Anforderungen auf:

Tabelle 30 Funktionale Anforderungen

FR-ID	Beschreibung
1	Falls der KVP-Verantwortliche einen 5S-Auftrag zusendet, muss jeder die 5S-Aktivität durchführen
2	Falls jemand eine Prozessbeschreibung anfragt, muss jeder fähig sein, eine Liste mit den dokumentierten Prozessen zurückzuliefern
3	Falls der KVP-Verantwortliche einen neuen Prozess definiert hat, muss die Arbeitsstation fähig sein, diesen Prozess umzusetzen
4	Falls der KVP-Verantwortliche eine Prozessbeschreibung anfragt, muss jeder fähig sein, eine Liste mit den dokumentierten Prozessen zurückzuliefern
5	Falls der KVP-Verantwortliche einen Prozess überarbeitet hat, muss jeder fähig sein, diesen Prozess umzusetzen
6	Falls der KVP-Verantwortliche eine Auditabweichung feststellt, muss jeder die Abweichung beheben
7	Falls der Zuständige die Nachbesserung durchgeführt hat, muss der Zuständige die Aktualisierung der Verbesserungsmaßnahme dem KVP zu senden
8	Falls der KVP-Verantwortliche eine Prüfungsanweisung aktualisiert, muss die Arbeitsstation fähig sein, die aktualisierte Prüfungsanweisung zu übernehmen
9	Falls der KVP-Verantwortliche die vorliegenden 8D-Reports anfragt, muss jeder fähig sein, eine Liste der eigenen 8D-Reports zurückzuliefern
10	Falls ein Teileempfänger dem Teilezulieferer einen 8D-Report zusendet, muss die Arbeitsstation fähig sein, den 8D-Report zu verarbeiten
11	Falls ein Teilezulieferer den 8D-Report aktualisiert hat, muss die Arbeitsstation fähig sein, die Aktualisierung des 8D-Report zu empfangen
12	Falls Arbeitsstation Maßnahmen zur Fehlerwiederholungsvermeidung festgelegt hat, muss der KVP-Verantwortliche fähig sein, die Maßnahmen zur Fehlerwiederholungsvermeidung zu verarbeiten
13	Falls die Arbeitsstation die Fehlerwiederholungsmaßnahmen umgesetzt hat, muss die Arbeitsstation fähig sein, die Aktualisierung des 8D-Reports zu empfangen
14	Falls eine Anlage an einer Arbeitsstation eine Abweichung identifiziert, die sie nicht selber beheben kann, muss die Arbeitsstation die Störung gemäß Andon-Methode behandeln
15	Falls jemand ein ausgefülltes Ishikawa-Diagramm zusendet, muss der KVP-Verantwortliche fähig sein, das Ishikawa-Diagramm zu interpretieren
16	Falls der KVP-Verantwortliche aus dem Ishikawa-Diagramm Maßnahmen ableitet, muss der für diesen Bereich verantwortliche fähig sein, die Sofortmaßnahmen umzusetzen
17	Falls die Eskalationsstufen aktualisiert werden, muss jeder fähig sein, die aktualisierten Eskalationsstufen zu übernehmen
18	Falls ein Fehler identifiziert wurde, muss jeder dem für diesen Bereich Verantwortlichen eine Fehlermeldung zu senden
19	Falls ein Fehler empfangen wurde, muss der für diesen Bereich verantwortliche die Fehleranalyse dem KVP-Verantwortlichen zu senden
20	Falls der KVP-Verantwortliche aus der Fehleranalyse Maßnahmen ableitet, muss der für diesen Bereich verantwortliche fähig sein, die Sofortmaßnahmen umzusetzen
21	Falls eine Poka Yoke-Maßnahme umgesetzt wurde, muss der für diesen Bereich verantwortliche dem KVP-Verantwortlichen die Dokumentation der erfolgreichen Umsetzung zu senden
22	Falls der KVP-Verantwortliche Verbesserungsmaßnahmen geplant hat, muss jeder fähig sein, die Verbesserungsmaßnahmen umzusetzen
23	Falls der KVP-Verantwortliche Überwachungsmechanismen für die Six Sigma-Maßnahme festlegt, muss jeder fähig sein, den Überwachungsmechanismus umzusetzen
24	Falls eine Verbesserungsmaßnahme umgesetzt wurde, muss jeder dem KVP-Verantwortlichen den aktualisierten Status zu senden
25	Falls die Six Sigma-Kennzahlen abgefragt werden, muss die Arbeitsstation fähig sein, die Six Sigma Kennzahlen zurückzuliefern
26	Falls die Six Sigma-Kennzahlen für eine Fertigungslinie benötigt werden, muss der KVP-Verantwortliche fähig sein, die Six Sigma Kennzahlen der Linie zurückzuliefern
27	Falls die Qualitätsparameter einer Komponente abgerufen werden, muss die Arbeitsstation fähig sein, die Qualitätsparameter für diese Komponente zurückzuliefern
28	Falls an der Arbeitsstation eine Störung auftritt, muss die Arbeitsstation dem Werker die Möglichkeit bieten dem Hanco die Störung zu senden
29	Falls die Störung nicht innerhalb des Taktes bzw. neben der Fertigung beherrschbar ist, muss die Arbeitsstation dem Hanco die Möglichkeit bieten von der Arbeitsstation das rote Stoppsignal an alle Arbeitsstationen der Linie zu senden
30	Falls der Hanco fertig ist, muss die Arbeitsstation dem Hanco die Möglichkeit bieten die Fertigstellung mitzuteilen
31	Falls eine Arbeitsstation das Linienstoppsignal empfängt, muss die Arbeitsstation die Arbeit zum Ende des Taktes anhalten
32	Falls der Hanco das Eintreffen quittiert, muss die Arbeitsstation den eigenen Status aktualisieren
33	Falls die Störung behoben ist und der Status wieder "0: i.O." ist, muss die Arbeitsstation allen Arbeitsstationen der Linie das Ende des Linienstopps mitteilen
34	Falls Arbeitsstation einen Linienstopp auslöst, muss die Arbeitsstation die Andon-Kennzahlen aktualisieren
35	Falls der KVP-Verantwortliche die Störungsmeldungen an der Arbeitsstation abfragt, muss die Arbeitsstation fähig sein, die Liste der Störungsmeldungen zurückzugeben
36	Falls der Status abgefragt wird, muss die Arbeitsstation fähig sein, den Status zurückzugeben

37	Falls der KVP-Verantwortliche die Soll-Kennzahlen für die Andon-Tafel aktualisiert, muss die Arbeitsstation die Soll-Kennzahlen der Andon-Tafel aktualisieren
38	Falls die Andon-Kennzahlen einer Arbeitsstation angefragt werden, muss die Arbeitsstation die Andon-Kennzahlen zurückgeben
39	Falls der Hancho eintrifft, muss die Arbeitsstation dem Hancho die Möglichkeit bieten das Eintreffen zu quittieren
40	Falls der Hancho die Störung nicht alleine beheben kann, muss die Arbeitsstation dem Hancho die Möglichkeit bieten weitere Unterstützung anzufordern
41	Falls der Hancho weitere Unterstützung anfordert, muss die Arbeitsstation den eigenen Status aktualisieren
42	Falls der Fehler behoben wurde, muss die Arbeitsstation den eigenen Status aktualisieren
43	Falls der Werker an der Arbeitsstation eine Störung meldet, muss die Arbeitsstation den eigenen Status aktualisieren
44	Falls weitere Unterstützung eintrifft, muss die Arbeitsstation der Unterstützung die Möglichkeit bieten das Eintreffen zu quittieren
45	Falls die weitere Unterstützung fertig ist, muss der Hancho der Unterstützung die Möglichkeit bieten die Fertigstellung mitzuteilen
46	Falls der Shopfloor-Termin vorbereitet wird, muss der Hancho die Kennzahlen auf der Shopfloor-Tafel aktualisieren
47	Falls jemand eine Verbesserungsmaßnahme einleitet, muss der Bearbeiter fähig sein, die Verbesserungsmaßnahme umzusetzen
48	Falls sich der Status einer Verbesserungsmaßnahme ändert, muss der Bearbeiter die Aktualisierung der Verbesserungsmaßnahme dem Ersteller zu senden
49	Falls der KVP-Verantwortliche aus dem Audit Verbesserungsmaßnahmen ableitet, muss jeder fähig sein, die Verbesserungsmaßnahme umzusetzen
50	Falls die Leitungsebene einen Auditbericht anfordert, muss der KVP-Verantwortliche fähig sein, der Leitungsebene den spezifischen Audit-Bericht zu senden
51	Falls das Benchmarking Verbesserungsmaßnahmen ergibt, muss jeder fähig sein, die Verbesserungsmaßnahmen zu empfangen
52	Falls jemand einen Verbesserungsvorschlag hat, muss der KVP-Verantwortliche fähig sein, diesen Verbesserungsvorschlag anzunehmen
53	Falls der KVP-Verantwortliche den Status des Verbesserungsvorschlags aktualisiert, muss der Ersteller fähig sein, diese Statusaktualisierung zu empfangen
54	Falls KVP-Verantwortliche einen Verbesserungsvorschlag umsetzen will, muss jeder fähig sein, den Verbesserungsvorschlag als Verbesserungsmaßnahme zu empfangen
55	Falls der Verbesserungsvorschlag umgesetzt wurde, muss der Empfänger des Verbesserungsvorschlags dem KVP-Verantwortlichen eine Statusaktualisierung zu senden
56	Falls der Verbesserungsvorschlag bearbeitet wurde, muss der KVP-Verantwortliche dem Einreicher des Verbesserungsvorschlags den aktualisierten Status zu senden
57	Falls eine Maßnahme umgesetzt wird, muss der Bearbeiter den PDCA-Zyklus befolgen
58	Falls der KVP-Verantwortliche die Ist-Bearbeitungszeiten für einen Kanbantyp anfragt, muss die Arbeitsstation fähig sein, die Ist-Bearbeitungszeit für einen Kanbantyp zurückzuliefern
59	Falls der KVP-Verantwortliche die Taktzeit für einen Kanbantyp ändert, muss die Arbeitsstation fähig sein, die Daten der betroffenen Kanban zu aktualisieren
60	Falls die Prozesskennzahlen angefragt werden, muss die Arbeitsstation fähig sein, die Prozesskennzahlen zurückzuliefern
61	Falls der KVP-Verantwortliche die Prozesse anfragt, muss die Arbeitsstation fähig sein, eine Liste mit den durchführbaren Prozessen zurückzuliefern
62	Falls der KVP-Verantwortliche oder die Arbeitsstation einen neuen Prozess definiert hat, muss die Arbeitsstation fähig sein, diesen Prozess umzusetzen
63	Falls die Losgröße abgefragt wird, muss die Arbeitsstation fähig sein, die von der Arbeitsstation verschickbaren Kanban bereitzustellen
64	Falls der KVP-Verantwortliche die Losgröße verkleinert, muss die Arbeitsstation fähig sein, den Meldebestand verkleinern
65	Falls der KVP-Verantwortliche die Rüstzeiten abfragt, muss die Arbeitsstation fähig sein, die Ist-Rüstzeiten an der Arbeitsstation bereitzustellen
66	Falls die externe Rüstvorbereitung ansteht, muss die Arbeitsstation dem Werker die Möglichkeit bieten die Dauer der externen Rüstvorbereitung aufzunehmen
67	Falls die interne Rüstoperation ansteht, muss die Arbeitsstation dem Werker die Möglichkeit bieten die Dauer der internen Rüstoperation aufzunehmen
68	Falls die externe Rüstnachbereitung ansteht, muss die Arbeitsstation dem Werker die Möglichkeit bieten die Dauer der externen Rüstnachbereitung aufzunehmen
69	Falls der KVP-Verantwortliche die Kundenbedarfe anfragt, muss die Auftragsfreigabe fähig sein, die Kundenbedarfe für die Wertstromanalyse zurückzuliefern
70	Falls der KVP-Verantwortliche die Produktionsprozessdaten anfragt, muss die Arbeitsstation fähig sein, die Produktionsprozessdaten für die Wertstromanalyse zurückzuliefern
71	Falls der KVP-Verantwortliche die Materialflussdaten anfragt, muss die Logistik fähig sein, die Materialflussdaten für die Wertstromanalyse zurückzuliefern
72	Falls der KVP-Verantwortliche die Lagerdaten anfragt, muss die Arbeitsstation fähig sein, die Lagerdaten für die Wertstromanalyse zurückzuliefern
73	Falls der KVP-Verantwortliche die Auftragsdaten abrufen, muss die Auftragsfreigabe fähig sein, die Auftragsdaten für die Wertstromanalyse zurückzuliefern
74	Falls der KVP-Verantwortliche die Daten für die Wertstromanalyse ausgewertet hat, muss der KVP-Verantwortliche Verbesserungsmaßnahmen an den Verantwortlichen kommunizieren und überwachen
75	Falls die Arbeit nicht binnen des Taktes fertiggestellt werden kann, muss die Arbeitsstation bzw. die Logistik der nachgelagerten Station den Verzug mitteilen
76	Falls das KVP neue Taktzeit(en) für die Fertigungslinie ermittelt hat, muss die Arbeitsstation fähig sein, die aktualisierte(n) Taktzeiten zu empfangen

77	Falls KVP die Taktzeiten abfragt, muss die Arbeitsstation fähig sein, eine Liste mit den gefertigten Produkten mit Soll- und Ist-durchschnittlichen Taktzeiten zurückgeben
78	Sobald Arbeitsstation ein Produkt fertigt, sollte die Arbeitsstation fähig sein, die Soll-Zeit und Restlaufzeit des Taktes auszugeben
79	Falls der KVP-Verantwortliche die Aufträge eines vergangenen Zeitintervalls abrufen, muss die Auftragsfreigabe fähig sein, eine Liste der bisher bearbeiteten Aufträge zurückzuliefern
80	Falls eine Kanban nicht binnen der Soll-Zeit bearbeitet werden kann, muss der Bearbeiter dem Kanban-Versender eine Verzugsmitteilung zu senden
81	Falls der Meldebestand unterschritten ist, muss die Arbeitsstation die Kanban der nachgelagerten Arbeitsstation zu senden
82	Falls ein Kundenauftrag freigegeben wurde, muss die Auftragsverwaltung den Auftrag der letzten Arbeitsstation zu senden
83	Falls die Konfiguration abgefragt wird, muss die Arbeitsstation fähig sein, die von der Arbeitsstation verarbeitbaren Kanban bereitzustellen
84	Falls der KVP-Verantwortliche die Konfiguration aktualisiert, muss die Arbeitsstation fähig sein, die Daten der betroffenen Kanban zu aktualisieren
85	Falls die Kanban fertig bearbeitet wurde, muss die Arbeitsstation der Logistik einen Transportauftrag zu senden
86	Falls die Fertigstellungszeit einer Kanban überschritten ist, muss die Arbeitsstation bzw. die Logistik die Verzögerung dem Auftraggeber der Kanban zu senden
87	Falls die Bearbeitung einer Kanban fertig ist, muss die Arbeitsstation bzw. die Logistik die Fertigstellung dem Auftraggeber der Kanban zu senden
88	Falls die Konfiguration abgefragt wird, muss die Arbeitsstation fähig sein, die von der Arbeitsstation verschickbaren Kanban bereitzustellen
89	Falls die Logistik Teile an einer Arbeitsstation abliefern, muss die Logistik die Transportkanban der Arbeitsstation aktualisieren
90	Falls eine Arbeitsstation die Produktion einer Kanban fertigstellt, muss die Arbeitsstation der Logistik einen Transportauftrag zu senden
91	Falls der KVP-Verantwortliche die Aufträge eines vergangenen Zeitintervalls abrufen, muss die Auftragsfreigabe fähig sein, eine Liste der bisher bearbeiteten Aufträge zurückzuliefern
92	Falls der KVP-Verantwortliche die Losgrößen neu berechnet, muss die Arbeitsstation fähig sein, die Daten der betroffenen Kanban zu aktualisieren
93	Falls der KVP-Verantwortliche die Zykluszahl für ein Produkt erhöht, muss die Arbeitsstation fähig sein, die Daten der betroffenen Kanban zu aktualisieren
94	Falls der KVP-Verantwortliche die Eigenschaften der Lagerflächen abfragt, muss der Supermarkt-Verantwortliche die Lagerdaten zurückliefern
95	Falls der KVP-Verantwortliche die Eigenschaften der Lagerfläche aktualisiert, muss der Supermarkt-Verantwortliche fähig sein, die neuen Lagerdaten zu aktualisieren
96	Falls der Meldebestand eines Supermarkt-Lagerorts unterschritten wird, muss der Supermarkt-Verantwortliche eine Fertigungskanban an die vorgelagerte Arbeitsstation zu senden
97	Falls der KVP-Verantwortliche aus der Verschwendungsbewertung eine Verbesserungsmaßnahme ableitet, muss jeder fähig sein, die Verbesserungsmaßnahme umzusetzen

Anhang E – Spezifikation der je Rolle anzubietenden Funktionalitäten

Nachstehende Tabellen listen die je Rolle anzubietenden Funktionalitäten auf:

Tabelle 31 Funktionalitäten der Arbeitsstation

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
Kanban	»aktualisiereKanban«	geänderteKanban: KanbanFertigung	-	Aktualisiert eine bestehende Fertigungs-Kanban anhand der empfangenen Fertigungs-Kanban. Hierdurch kann u.a. das KVP-System nach einer Änderung der Taktzeiten und Losgrößen neue Kanban verteilen.
One Piece Flow	»aktualisiereMeldebestand«	ZuÄndernderMeldebestand: Meldebestand	-	Ändert den Meldebestand, ab welchem eine Arbeitsstation eine Kanban für Nachschub an die vorgelagerte Arbeitsstation verschickt
Werker-selbstkontrolle	»aktualisierePrüfungsanweisung«	aktualisiertePrüfungsanweisung: Prüfungsanweisung	-	Aktualisiert eine Prüfungsanweisung anhand der übergebenen, aktualisierten Version
Andon	»aktualisiereSoll-KennzahlenAndon«	geänderteSoll-Kennzahlen: KennzahlenAndonSoll	-	Aktualisiert die Soll-Kennzahlen von Andon anhand der übergebenen, neuen Kennzahlen. Hierdurch kann das KVP-System nach u.a. einer Nivellierung der Produktion die Vorgabe-Zahlen in einer Arbeitsstation aktualisieren.
Taktung	»aktualisiereTaktzeit«	AktualisierteTaktzeit: KanbanFertigung	-	Aktualisiert die Taktzeit für ein Produkt an einer Arbeitsstation
8D-Report	»empfange8D-Report«	vorausgefüllter8D-Report: 8D-Report	-	Empfängt einen teils vorausgefüllten 8D-Report eines nachgelagerten Teileempfängers
8D-Report	»empfangeFehlerwiederholungsFertigstellung«	aktueller8D-Report : 8D-Report	-	Bestätigt die Bearbeitung der Fehlerwiederholungsschritte (u.a. Prozess- und Produkt-FMEA durchführen)
Kanban; Supermarkt	»empfangeKanbanFertigung«	Fertigungskanban: KanbanFertigung	-	Empfängt eine Kanban mit einem Fertigungsauftrag von einer anderen (vorgelagerten) Arbeitsstation. Hierdurch wird der Arbeitsstation mitgeteilt, dass sie mit der Produktion des auf der Kanban angegebenen Produktes beginnen soll.
Kanban	»empfangeKanbanAuftrag«	Kundenauftrag: KanbanAuftrag	-	Ermöglicht das Übermitteln eines Kundenauftrags an eine Arbeitsstation, welche anschließend die Produktion aufnimmt. Wird von der ersten Arbeitsstation einer Fertigungslinie verwendet, um von der Auftragsfreiegabe Kanban-Aufträge zu erhalten.
Kanban	»empfangeStatusKanbanFertigung«	Status: StatusKanban	-	Empfängt eine übergebene Statusaktualisierung zu einer von der Arbeitsstation zuvor versendeten Fertigungskanban. Hierdurch kann die Arbeitsstation über Verzögerungen und die Fertigstellung der Fertigungskanban informiert werden.
Kanban; Milkrun	»empfangeStatusKanbanTransport«	Status: StatusKanban	-	Empfängt eine übergebene Statusaktualisierung zu einem von der Arbeitsstation zuvor versendeten Transportauftrag. Hierdurch kann die Arbeitsstation über Verzögerungen und die Fertigstellung des Transports informiert werden.
Andon	»liefereAndon-Kennzahlen«	Antwortadresse: iSID	KennzahlenAndon	Liefert die Soll- und Ist-Kennzahlen für Andon. Wird u.a. für die Anzeige auf der Andon-Tafel verwendet
Taktung	»liefereAktuellenTakt«	Antwortadresse: iSID	AktuellerTakt	Liefert Taktdauer und Startzeit zur Berechnung der Restlaufzeit und Anzeige auf z.B. Tafeln zurück
Bottleneck-Analyse	»liefereDurchschnitts-Bearbeitungszeit«	KanbanID: ID; Antwortadresse: iSID	KennzahlenIst-Bearbeitungszeit	Liefert die durchschnittliche Ist-Bearbeitungszeit für einen Kanban-Typ
Kanban	»liefereEmpfangbareKanbanListe«	Antwortadresse: iSID	ListeKanban	Liefert dem Anfragenden eine Liste aller Kanban mit, die die Arbeitsstation empfangen und verarbeiten kann. Verarbeiten bedeutet hierbei, dass sie die auf der Kanban angegebenen Produkte selbst herstellen kann. Wird u.a. von anderen Teilnehmern und dem KVP-System verwendet um herauszufinden, was eine Arbeitsstation leisten kann.
Shopfloor Management	»liefereKennzahlenShopfloorRunde«	Antwortadresse: iSID	KennzahlenShopfloor	Liefert die Kennzahlen für die Shopfloor-Tafel in der Shopfloor-Runde

Werts-trompla-nung	»liefereLagerda-ten«	Produktgruppe: Wertstromanaly-seProduktgruppe; Antwortadresse: iSID	Wertstromanaly-seLagerdaten	Liefert die für die Wertstromanalyse notwendigen Lagerdaten zu einer Produktgruppe
One Piece Flow	»liefereMeldebe-stand«	VorproduktID: String; Antwortadresse: iSID	Meldebestand	Liefert den Meldebestand für ein Produkt, bei dem eine Kanban zur Nachschub-Beschaffung ausgelöst wird.
Werts-trompla-nung	»liefereProdukti-onsprozessdaten«	Produktgruppe: Wertstromanaly-seProduktgruppe; Antwortadresse: iSID	Wertstromanaly-seProduktions-prozessdaten	Liefert die für die Wertstromanalyse notwendigen Produktionsprozessdaten einer Arbeitsstation zu einer Produktgruppe
Prozess-& Perfor-mance-ma-nage-ment	»liefereProzess-kennzahlen«	Antwortadresse: iSID	KennzahlenPro-zessmanagement	Liefert eine Liste mit für das Prozessmanagement relevanten Kennzahlen
Statisti-sche Prozess-regelung	»liefereQualitätspa-rameter«	Antwortadresse: iSID	Qualitätsparame-ter	Liefert die für die Qualität relevanten Parameter der an einer Arbeitsstation verwendeten Komponenten
SMED	»liefereRüstvor-gängeListe«	Antwortadresse: iSID	ListeRüstvorgän-ge	Liefert eine Liste aller Rüstvorgänge mit deren Dauer, die an einer Arbeitsstation in der Vergangenheit durchgeführt wurden. Wird vom KVP-System für Verbesserungen verwendet.
Six Sig-ma	»liefereSixSigma-Kennzahlen«	Antwortadresse: iSID	Kennzah-lenSixSigma	Liefert die für Six Sigma relevanten Kennzahlen einer Arbeitsstation
Andon	»liefereStörungslis-te«	Antwortadresse: iSID	ListeStörungen	Liefert eine Liste aller aufgetretenen Störungen an einer Arbeitsstation. Diese kann vom KVP-System anschließend hinsichtlich KVP-Maßnahmen ausgewertet werden oder der Anzeige auf einer Shopfloor-Tafel dienen.
Andon	»liefereStatus«	Antwortadresse: iSID	StatusArbeitssta-tion	Liefert den Status einer Arbeitsstation gemäß der Andon-Abstufung (z.B. 5-Stufen-Modell in VDI 2870-2). Die Information wird für die Andon-Tafel sowie die Visualisierung des Arbeitsstations-Zustands verwen-det.
Taktung	»liefereTaktzeitLis-te«	Antwortadresse: iSID	ListeTaktzeiten	Liefert eine Liste der vergangenen bearbeiteten Pro-dukte und deren Soll- und Ist-Zeiten der Arbeitsstation
Kanban	»liefereVerschick-bareKanbanListe«	Antwortadresse: iSID	ListeKanban	Liefert dem Anfragenden eine Liste aller Kanban, die die Arbeitsstation versenden und deren Produkte sie als Vorprodukte verarbeiten kann. Wird u.a. von anderen Teilnehmern und dem KVP verwendet um heraus-zufinden, was eine Arbeitsstation leisten kann.

Tabelle 32 Funktionalitäten der Arbeitsstation

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
Kanban	»empfangenStau-tusAuftragsKan-ban«	Status : Sta-tusKanban	-	Empfängt eine übergebene Statusaktualisierung zu einer von der Auftragsfreigabe zuvor versendeten Auftragskanban. Hierdurch kann die Auftragsfreigabe über Verzögerungen und die Fertigstellung des Auf-trags informiert werden.
Heijunka; JIT/ JIS	»liefereAuftragslis-te«	Startdatum : Date, Enddatum : Date, Antwortadresse : iSID	ListeAufträge	Liefert eine Liste aller Kundenaufträge in einem vorge-benen Zeitraum. Hierdurch kann das KVP-System die monatlichen Kundenbedarfe bestimmen, um eine Nivellierung durchzuführen und die Auftragssystematik für JiT/ JiS umzustellen.
Werts-trompla-nung	»liefereAuftrags-warteschlange«	Produktgruppe : Wertstromanaly-seAuftragswarte-schlange, Antwortadresse : iSID	Wertstromanaly-seAuftragswarte-schlange	Liefert die für eine übergebene Produktgruppe aktuelle Auftragswarteschlange (Queue). D.h. die Kundenauf-träge, die bereits bestellt sind, sich aber noch nicht in der Fertigung befinden.
Werts-trompla-nung	»liefereKundenbe-darfe«	Produktgruppe : Wertstromanaly-seProduktgruppe, Antwortadresse : iSID	Wertstromanaly-seKundenbedarfe	Liefert die Kundenbedarfe für eine übergebene Pro-duktgruppe. Auf Basis dieser Information kann z.B. das KVP-System eine Wertstromplanung durchführen.

Kanban	»aktualisiereKanban«	geänderteKanban: KanbanAuftrag	-	Aktualisiert oder Löscht eine bestehende Auftrags-Kanban bei der Auftragsfreigabe
--------	----------------------	--------------------------------	---	---

Tabelle 33 Funktionalitäten der iS-Rolle

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
Kurze Regelkreise	»aktualisiereEskalationsstufen«	AktualisierteEskalationsstufen: Eskalationsstufen	-	Empfängt eine Beschreibung der geänderten Eskalationsstufen
8D-Report	»empfangen8D-ReportAktualisierung«	8D-Report: 8D-Report	-	Empfängt eine Aktualisierung zu einem zuvor zur Bearbeitung an den Lieferanten geschickten 8D-Report. Die Aktualisierung kann beispielsweise von der Vorliefer-Arbeitsstation eingeleitete Sofortmaßnahmen nach einem aufgetretenen Produktfehler sein.
Poka Yoke	»empfangenFehler«	Fehlerbeschreibung: Qualitätsfehler	-	Empfängt eine Fehlerbenachrichtigung, die den eigenen Verantwortungsbereich betrifft
5S	»empfangenKanban5S«	5S-Auftrag: Kanban5S	-	Löst eine 5S-Aktivität aus
Andon	»empfangenKanbanSupport«	aktuelleStörung: KanbanSupport	-	Empfängt und verarbeitet eine Kanban mit einer Supportanfrage. Hierdurch kann eine iS-Rolle im Falle einer Störung beim Hancho Unterstützung anfordern.
Andon	»empfangenLinienstopp«	-	-	Gibt das Signal, dass die Fertigungslinie gestoppt werden muss. Entspricht der Andon-Reißleine.
Andon	»empfangenLinienstoppaktualisierung«	Status: StatusLinienstopp	-	Übermittelt eine Statusaktualisierung für einen vorausgehenden Linienstopp. Wird u.a. verwendet, um nach einer behobenen Störung die Linie wieder zu starten.
Prozess- & Performance-Management; Prozessstandardisierung	»empfangenProzessbeschreibung«	Prozess: Prozessbeschreibung	boolean	Empfängt einen beschriebenen Prozess, welcher übernommen werden soll und dessen Akzeptanz zu quittieren ist (true = angenommen)
(mehrere)	»empfangenVerbesserungsmaßnahme«	Verbesserungsmaßnahme: Verbesserungsmaßnahme	-	Empfängt eine Verbesserungsmaßnahme, die die iS-Rolle umsetzen soll. Z.B. eine temporäre Behebung einer Störung an einer Arbeitsstation, welche der Hancho übermittelt hat.
(mehrere)	»empfangenVerbesserungsmaßnahmeAktualisierung«	Status: StatusVerbesserungsmaßnahme	-	Gibt dem Ersteller eine Aktualisierung zu einer zuvor übergebenen Verbesserungsmaßnahme.
Andon	»empfangenStatusKanbanSupport«	Status: StatusKanban	-	Empfängt eine übergebene Statusaktualisierung zu einer von der iS-Rolle zuvor versendeten Support-Kanban. Hierdurch kann z.B. die Arbeitsstation über Verzögerungen und die Fertigstellung der Unterstützung informiert werden.
Ideenmanagement	»empfangenVerbesserungsvorschlagsAktualisierung«	Status: StatusVerbesserungsvorschlag	-	Empfängt eine Aktualisierung zu einem zuvor eingereichten Verbesserungsvorschlag
8D-Report	»liefere8D-Reportliste«	Antwortadresse: iSID	Liste8D-Reports	Liefert eine Liste aller 8D-Reports, die bei der iS-Rolle aufgetreten sind. Hierdurch kann das KVP-System eine Auswertung für KVP-Maßnahmen erstellen.
Prozessstandardisierung; Prozess- & Performance-Management	»liefereProzessbeschreibungen«	Antwortadresse: iSID	ListeProzessbeschreibungen	Liefert eine Liste mit durchführbaren Prozessen
-	»liefereRollentyp«	-	String	Liefert den Typen der iS-Rolle, um es Dritten zu ermöglichen die Funktionalitäten einer angesprochenen Instanz in Erfahrung zu bringen.

Tabelle 34 Funktionalitäten des KVP-Systems

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
8D-Report	»empfangenFehlerwiederholungsaktivität«	aktueller8D-Report : 8D-Report	-	Initiiert am KVP-System den Prozess, um eine Produkt- und Prozess-FMEA durchzuführen sowie einen Kontrollplan und -prozess zu definieren (Schritt 7 des 8D-Reports in VDI 2870-2)
Poka Yoke; Ishikawa-Diagramm	»empfangenFehleranalyse«	Fehleranalyse: Fehleranalyse	-	Empfängt die Ergebnisse einer FMEA oder Ishikawa zur Analyse und zum Ableiten von Maßnahmen
Poka Yoke	»empfangen-PokaYokeUmsetzung«	Dokumentation: PokaYokeDokumentation	-	Dokumentiert die erfolgreich eingeführte Poka-Yoke-Maßnahme
SixSigma	»liefereLiniennSixSigmaKennzahlen«	Fertigungslinie: String; Antwortadresse: iSID	KennzahlenSixSigma	Liefert die für Six Sigma relevanten Kennzahlen einer Fertigungslinie
Ideemanagement	»empfangenVerbesserungsvorschlag«	Idee: Verbesserungsvorschlag	-	Empfängt einen Verbesserungsvorschlag zur weiteren Nachverfolgung (Bewertung und ggf. Umsetzung)
Audit	»liefereAuditbericht«	AuditID: int; Antwortadresse: iSID	Auditbericht	Liefert einen spezifischen Auditbericht zu einer übergebenen Audit-ID
5S	»empfangenStatusKanban5S«	Status: StatusKanban	-	Empfängt eine Statusaktualisierung zu einer zuvor versendeten 5S-Kanban

Tabelle 35 Funktionalitäten des KVP-Systems

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
Kanban; Milkrun	»empfangenKanbanTransport«	Transportauftrag: KanbanTransport	-	Empfängt und verarbeitet eine Kanban mit einem Transportauftrag. Hierdurch wird eine dem Logistiksystem mitgeteilt, dass eine Materialbewegung z.B. im nächsten Milkrun notwendig ist.
Wertstromplanung	»liefereMaterialflussdaten«	Produktgruppe: WertstromanalyseProduktgruppe; Antwortadresse: iSID	WertstromanalyseMaterialflussdaten	Liefert Materialflussinformationen für eine übergebene Produktgruppe. Auf Basis dieser Information kann z.B. das KVP eine Wertstromplanung durchführen.
Taktung	»aktualisiereTaktzeit«	AktualisierteTaktzeit:Taktzeit	-	Aktualisiert die allg. bzw. produktbezogene Taktzeit einer Logistiksystem
Taktung	»liefereTaktzeitListe«	Antwortadresse: iSID	ListeTaktzeiten	Liefert eine Liste der vergangenen bearbeiteten Produkte und deren Soll- und Ist-Zeiten des Logistiksystems
Kanban	»aktualisiereKanban«	geänderteKanban: KanbanTransport	-	Aktualisiert oder löscht eine bestehende Transport-Kanban.

Tabelle 36 Funktionalitäten des Supermarkts

Lean-Methode	Dienstname	Parameter	Rückgabewert	Beschreibung
Supermarkt	»liefereLagerdaten«	Antwortadresse: iSID	Lagerdaten	Liefert die Lagerdaten (Orte, Kapazitäten, zugeordnete Produkte etc.) eines Supermarktes
Supermarkt	»aktualisiereLagerdaten«	neueLagerdaten : Lagerdaten	-	Aktualisiert die Lagerdaten (Orte, Kapazitäten, zugeordnete Produkte etc.) eines Supermarktes

Anhang F – Spezifikation der Nachrichten

Nachstehend sind die für den Aufruf der Funktionen bei den Rollen benötigten Nachrichten und ihre Attribute näher beschrieben:

8D-Report:

Der 8D-Report besteht aus Attributen und in ihm enthaltenen Objekten, welche wiederum eigene Attribute haben. Der 8D-Report ist eine digitale Version des papierbasierten Berichtes. Der 8D-Report ist ein Formular, welches ein Kunde/ Warenempfänger (z.B. Arbeitsstation) ausfüllt, um die Fehlerbeseitigung sowie Sofortmaßnahmen mit dem Lieferanten (z.B. einer anderen Arbeitsstation) zu koordinieren. Hierbei werden acht Dimensionen/ Schritte betrachtet, woraus sich der Name des Reports ergibt.

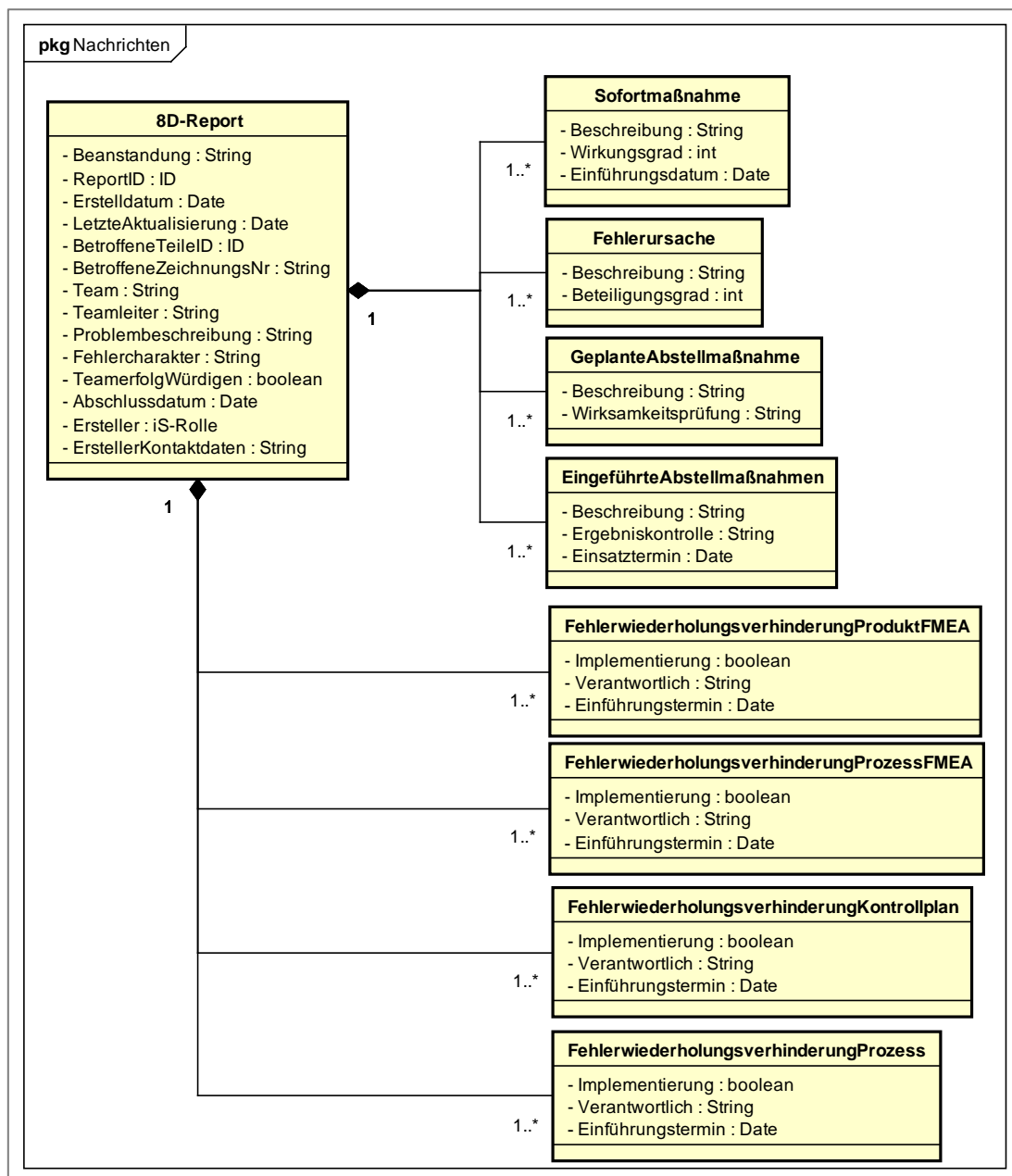


Abbildung 95 Struktur der Nachricht 8D-Report

Tabelle 37 Attribute des 8D-Reports

Name	Summary
»Beanstandung«	Titel der Reklamation; z.B. Text
»ReportID«	Identifikationsnummer des 8D-Reports; z.B. Fortlaufende Nummer oder Zeitstempel
»Erstelldatum«	Datum der Erstellung; z.B. Zeitstempel im Format YYYY-MM-dd
»LetzteAktualisierung«	Datum der letzten Aktualisierung; z.B. Zeitstempel im Format YYYY-MM-dd HH:MM:ss
»BetroffeneTeileID«	Artikelnummer, die von der Reklamation betroffen ist; z.B. Artikelnummer
»BetroffeneZeichnungsNr«	Zeichnungsnummer des betroffenen Artikels/ Produkts; z.B. Zeichnungsnummer
»Team«	Für die Bearbeitung des 8D-Reports involvierte Personen; z.B. Namensliste
»Teamleiter«	Für die Bearbeitung des 8D-Reports verantwortliche Person; z.B. Vor- und Nachname
»Problembeschreibung«	Detaillierte Beschreibung des aufgetretenen Problems/ Fehlers; z.B. Text
»Fehlercharakter«	Charakterisierung des Fehlers; z.B. Text
»TeamerfolgWürdigen«	Name des verantwortlichen Vorgesetzten, welcher den Erfolg würdigte; z.B. Text
»Abschlussdatum«	Abschlussdatum des 8D-Reports; z.B. Zeitstempel im Format YYYY-MM-dd
»Ersteller«	Name des Erstellers des 8D-Reports; z.B. Text
»ErstellerKontaktdaten«	Kontaktdaten des Erstellers; z.B. Text
»Sofortmaßnahmen«	Beschreibung eingeleiteter Sofortmaßnahmen; z.B. Beschreibungstext, Wirkungsgrad in %, Einführungsdatum je Sofortmaßnahme; s.u.
»Fehlerursache«	Beschreibung von Ursachen des Fehlers; z.B. Beschreibungstext und Beteiligungsgrad in % je Ursache; s.u.
»GeplanteAbstellmaßnahmen«	Beschreibung von Abstellmaßnahmen; z.B. Beschreibungstext und Ergebnis der Wirksamkeitsprüfung je Abstellmaßnahme.
»EingeführteAbstellmaßnahmen«	Beschreibung von eingeführten Abstellmaßnahmen; z.B. Beschreibungstext, Ergebniskontrolle als Datum mit Namen des Kontrolleurs, Einsatztermin je Abstellmaßnahme.
»FehlerwiederholungsverhinderungProduktFMEA«	Checkbox zur Kontrolle der langfristigen Fehlervermeidung; z.B. Checkbox, Verantwortlicher Mitarbeiter und Einführungstermin.
»FehlerwiederholungsverhinderungProzessFMEA«	Checkbox zur Kontrolle der langfristigen Fehlervermeidung; z.B. Checkbox, Verantwortlicher Mitarbeiter und Einführungstermin
»FehlerwiederholungsverhinderungKontrollplan«	Checkbox zur Kontrolle der langfristigen Fehlervermeidung; z.B. Checkbox, Verantwortlicher Mitarbeiter und Einführungstermin.
»FehlerwiederholungsverhinderungProzess«	Checkbox zur Kontrolle der langfristigen Fehlervermeidung; z.B. Checkbox, Verantwortlicher Mitarbeiter und Einführungstermin.

Auditbericht:

Der Auditbericht ist ein Bericht des Audits zur Vorstellung bei der Leitung.

Tabelle 38 Attribute des Auditberichtes

Attribute	Ausprägung/ Inhalte
»AuditberichtID«	Identifikationsnummer des Auditberichtes; z.B. Zeitstempel im Format YYYY-MM-dd hh:ss
»AuditbereichID«	Identifikationsnummer des Betrachtungsbereiches; z.B. Arbeitsstation-Nr. oder Kostenstelle
»Auditteam«	Namen der Auditteammitglieder; z.B. Auflistung von Text (Name + Vorname) oder Personal-Nr.
»Auditor«	Verantwortliche Auditor; z.B. Text
»Protokoll«	Protokoll des Audits; z.B. Text
»Resultate«	Ergebnisse (Abweichungen und Erfüllungen) des Audits; z.B. Text
»Maßnahmen«	Abgeleitete Maßnahmen zur Erfüllung des Audits; z.B. Text

Eskalationsstufen:

Beschreibung der Stufen, nach denen Probleme im Unternehmen eskaliert werden.

Tabelle 39 Attribute der Eskalationsstufen

Attribute	Ausprägung/ Inhalte
»Eskalationsstufenbeschreibung«	Beschreibung der Eskalationsstufen; z.B. Text

Fehleranalyse:

Enthält das Ergebnis einer Fehleranalyse.

Tabelle 40 Attribute der Fehleranalyse

Attribute	Ausprägung/ Inhalte
»ZugehörigeArbeitsstationID«	Identifikationsnr. Der für den Bereich verantwortlichen Arbeitsstation; z.B. Arbeitsstation-Nr. oder Kostenstelle
»Fehlerbeschreibung«	Beschreibung des zugehörigen Fehlers; z.B. Text
»IshikawaDiagramm«	Als Text ausgefülltes Ishikawa-Diagramm mit den Dimensionen Wirkung, Mensch, Methode, Material, Management, Milieu, Maschine, Organisation
»FMEA«	Ausgefüllter FMEA-Fragebogen

Kanban5S:

Kanban, welche keine weiteren Attribute enthält und beim Empfänger eine 5S-Aktivität auslöst.

KanbanAuftrag

Die Nachricht KanbanAuftrag ist vom Typ Kanban und beschreibt die durch die Auftragsfreigabe an eine Arbeitsstation ausgelöste Anweisung zur Fertigung eines Endproduktes.

Tabelle 41 Attribute der KanbanAuftrag

Attribute	Ausprägung/ Inhalte
»AuftragsID«	Identifikationsnummer dieser Kanban; z.B. Fortlaufende Nummer oder Zeitstempel
»AdresseWarenausgang«	Ablieferort der gefertigten Produkte; z.B. Kostenstelle oder Abstellflächen-Nr.
»ProduktID«	Individuelle Bezeichnung des Werkstücks; z.B. Artikelnummer
»ProduktName«	Menschenlesbarer Name des Werkstücks; z.B. Artikelname
»Losgröße«	Bedarfsmenge bzw. mit dieser Kanban zu fertigende Menge; z.B. Zahlenwert > 0
»WerkstückträgerTyp«	Ladehilfsmittel für den Transport; z.B. Typ- oder Größenangabe eines Behälters
»AdresseSenke«	Empfänger der Kanban (erste Arbeitsstation einer Linie); z.B. Identifikationsnummer der Schnittstelle/ Arbeitsstation
»Lieferzeit« <<optional>>	Bedarfszeitpunkt (vgl. Taktzeit-Ende); z.B. Zeitstempel im Format YYYY-MM-dd HH:MM
»Gültigkeit« <<optional>>	Optional; Datum, bis wann diese Kanban gültig war; z.B. Zeitstempel im Format YYYY-MM-dd HH:MM wenn ein Ablaufdatum bekannt; ansonsten leer

KanbanFertigung

Kanban, welche eine Arbeitsstation bei Unterschreitung des Meldebestands an eine vorgelagerte Arbeitsstation sendet, um dort die Produktion von Nachschub auszulösen.

Tabelle 42 Attribute der KanbanFertigung

Attribute	Ausprägung/ Inhalte
»Losgröße«	Losgröße des zu fertigenden Produkts; z.B. Zahlenwert > 0
»KanbanID«	Identifikationsnummer dieser Kanban; z.B. Fortlaufende Nummer oder Zeitstempel
»AdresseQuelle«	Sender der Kanban; z.B. Arbeitsstation-Nr. oder Kostenstelle
»AdresseSenke«	Empfänger der Kanban; z.B. Arbeitsstation-Nr. oder Kostenstelle
»ProduktID«	Individuelle Bezeichnung des Werkstücks; z.B. Artikelnummer
»ProduktName«	Menschenlesbarer Name des Werkstücks; z.B. Artikelname
»WerkstückträgerTyp«	Ladehilfsmittel für den Transport; z.B. Typ- oder Größenangabe eines Behälters
»Lieferzeit« <<optional>>	Optional; Lieferzeitpunkt der Produkte; z.B. Zeitstempel im Format YYYY-MM-dd HH:MM
»Taktzeit« <<optional>>	Optional; Taktzeit für das zu fertigende Produkt; z.B. Zahlenwert >0 in Minuten
»Gültigkeit« <<optional>>	Optional; Datum, bis wann diese Kanban gültig war; z.B. Zeitstempel im Format YYYY-MM-dd HH:MM wenn ein Ablaufdatum bekannt; ansonsten leer

KanbanSupport

Fordert Support von Dritten außerhalb der Arbeitsstation an. Ebenfalls vom Typ Kanban.

Tabelle 43 Attribute der KanbanSupport

Attribute	Ausprägung/ Inhalte
»AnforderndeArbeitsstationID«	Adresse der anfordernden Arbeitsstation
»Zeitstempel« <<optional>>	Optional; Zeitstempel, wann die Anforderung erstellt wurde

KanbanTransport

Nachricht vom Typ Kanban, welche beim Logistiksystem einen Transportbedarf anmeldet bzw. anfordert.

Tabelle 44 Attribute der KanbanTransport

Attribute	Ausprägung/ Inhalte
»AdresseQuelle«	Abholort der Ware; z.B. Lagerplatz-Nr. oder Kostenstelle
»AdresseSenke«	Ablieferort der Ware; z.B. Lagerplatz-Nr. oder Kostenstelle
»WerkstückträgerTyp«	Anzahl der zu transportierenden Ladehilfsmittel; z.B. Zahlenwert > 0
»WerkstückträgerAnzahl«	Anzahl der zu transportierenden Ladehilfsmittel
»Lieferzeit«	Ablieferzeitpunkt (vgl. Taktzeit-Ende); z.B. Zeitstempel im Format YYYY-MM-dd HH:MM
»TransportID«	Identifikationsnummer dieser Kanban; z.B. Fortlaufende Nummer oder Zeitstempel
»Taktzeit« <<optional>>	Optional; Taktzeit für das zu transportierende Produkt; z.B. Zahlenwert >0 in Minuten
»Gültigkeit« <<optional>>	Optional; Datum, bis wann diese Kanban gültig war; z.B. Zeitstempel im Format YYYY-MM-dd HH:MM wenn ein Ablaufdatum bekannt; ansonsten leer

KennzahlenAndon

Nachricht vom Typ Kennzahl mit den für die Andon-Methode relevanten Kennzahlen.

Tabelle 45 Attribute der KanbanAndon

Attribute	Ausprägung/ Inhalte
»IstTaktzeit«	Tatsächliche Taktzeit der Arbeitsstation; z.B. Zeitangabe in Minuten pro Stück
»ZeitintervallMinuten«	Zeitintervall, auf das sich die Ist-Kennzahlen beziehen; z.B. Angabe in Minuten
»AnzahlStörungen«	Anzahl der Störungen im Zeitintervall; z.B. Zahlenwert > 0
»IstAusbringungsmenge«	Anzahl der produzierten Produkte in einem Zeitintervall; z.B. Zahlenwert > 0
»KennzahlenAndonSoll«	Enthält die Soll-Kennzahlen einer Arbeitsstation; z.B. Soll-Taktzeit, -Ausbringungsmenge, -Zeitintervall.

KennzahlenAndonSoll

Enthält die Soll-Kennzahlen einer Arbeitsstation; z.B. Soll-Taktzeit, -Ausbringungsmenge, -Zeitintervall.

Tabelle 46 Attribute der KennzahlenAndonSoll

Attribute	Ausprägung/ Inhalte
»SollTaktzeit«	Vorgabe-Taktzeit für die Arbeitsstation; z.B. Zeitangabe in Minuten pro Stück
»SollAusbringungsmenge«	Vorgabe-Ausbringungsmenge für die Arbeitsstation; z.B. Zahlenwert > 0
»ZeitintervallMinuten«	Zeitintervall, auf das sich die Ist-Kennzahlen beziehen; z.B. Angabe in Minuten

KennzahlenProzessmanagement

Nachricht vom Typ Kennzahl. Enthält die für Prozessmanagement notwendigen Kennzahlen, welche sich u.a. aus den übergeordneten Zielen der Lean Production und üblichen Prozesskennzahlen ableiten.

Tabelle 47 Attribute der KennzahlenProzessmanagement

Attribute	Ausprägung/ Inhalte
»durchschnDLZ«	Durchlaufzeit im Bezugszeitraum zur Bewertung der Kundenanforderung "kürzeste Zeit"; z.B. Zahlenwert > 0 in Minuten
»durchschnFehlerrate«	Fehlerrate im Bezugszeitraum zur Bewertung der Kundenanforderung "beste Qualität"; z.B. Zahlenwert >=0 in %
»durchschnKosten«	Entstandene Kosten im Bezugszeitraum zur Bewertung der Kundenanforderung "geringste Kosten"; z.B. Zahlenwert > 0 in Euro
»Bezugszeitraum«	Bezugszeitraum, auf den sich die Kennzahlen beziehen; z.B. Zahlenwert >=0 in Minuten
»durchschnLeerlaufzeit«	Leerlaufzeit im Bezugszeitraum zur Bewertung der Verschwendung; z.B. Zahlenwert > 0 in Minuten
»durchschnProduktivität«	Produktivität im Bezugszeitraum zur Bewertung der Prozessperformance; z.B. Zahlenwert >=0 in %
»durchschnOEE«	Gesamtanlageneffektivität im Bezugszeitraum zur Bewertung der Prozessperformance; z.B. Zahlenwert >=0 in %

KennzahlenShopfloor

Enthält die Kennzahlen für das regelmäßige Shopfloor-Meeting zur Visualisierung auf der Shopfloor-Tafel.

Tabelle 48 Attribute der KennzahlenShopfloor

Attribute	Ausprägung/ Inhalte
»ZeitraumMinuten«	Zeitraum, auf den sich die Kennzahlen beziehen; z.B. Zahlenwert > 0 in Minuten vgl. Brunner, S.105: QKLSA-Kriterien
»AnzahlMängel«	Anzahl der aufgetretenen Mängel im Zeitraum; z.B. Zahlenwert >= 0 vgl. Brunner, S.105: QKLSA-Kriterien
»AnzahlNacharbeiten«	Anzahl der vorgenommenen Nacharbeiten im Zeitraum; z.B. Zahlenwert >= 0 vgl. Brunner, S.105: QKLSA-Kriterien
»AnzahlReklamationen«	Anzahl der Reklamationen im Zeitraum; z.B. Zahlenwert >= 0 vgl. Brunner, S.105: QKLSA-Kriterien
»AnzahlStillstände«	Anzahl der Stillstände im Zeitraum; z.B. Zahlenwert >= 0 vgl. Brunner, S.105: QKLSA-Kriterien
»Produktivität«	Produktivität der betrachteten Arbeitsstation (=Output / Input); z.B. Zahlenwert >= 0 in % vgl. Brunner, S.105: QKLSA-Kriterien
»AnzahlÜberstunden«	Anzahl der aufgetretenen Überstunden im Zeitraum; z.B. Zahlenwert >= 0 in h vgl. Brunner, S.105: QKLSA-Kriterien
»Zusatzausgaben«	Zusatzausgaben der Arbeitsstation; z.B. Zahlenwert >= 0 in Euro vgl. Brunner, S.105: QKLSA-Kriterien
»PlansollProduktivität«	Vorgabe-Produktivität der Arbeitsstation (=Soll-Output / Soll-Input); z.B. Zahlenwert >= 0 in % vgl. Brunner, S.105: QKLSA-Kriterien
»Produktionsvolumen«	Von der Arbeitsstation produzierte Menge; z.B. Zahlenwert >= 0 vgl. Brunner, S.105: QKLSA-Kriterien
»AnteilLieferfristenEingehalten«	Anteil der eingehaltenen Lieferfristen (Kanban); z.B. Zahlenwert >= 0 in % vgl. Brunner, S.105: QKLSA-Kriterien
»PlansollProduktionsvolumen«	Vorgabe-Produktionsvolumen der Arbeitsstation; z.B. Zahlenwert >= 0 vgl. Brunner, S.105: QKLSA-Kriterien
»AnzahlUnfälle«	Anzahl der aufgetretenen Unfälle im Zeitraum; z.B. Zahlenwert >= 0 vgl. Brunner, S.105: QKLSA-Kriterien
»AnzahlSicherheitsvorschläge«	Anzahl der gesammelten Sicherheitsvorschläge im Zeitraum; z.B. Zahlenwert >= 0 vgl. Brunner, S.105: QKLSA-Kriterien
»MitarbeiterVerspätungsrate«	Verspätungsrate der Mitarbeiter an der Arbeitsstation; z.B. Zahlenwert >= 0 in % vgl. Brunner, S.105: QKLSA-Kriterien
»MitarbeiterAbwesenheitsrate«	Abwesenheitsrate der Mitarbeiter an der Arbeitsstation; z.B. Zahlenwert >= 0 in % vgl. Brunner, S.105: QKLSA-Kriterien
»AnzahlVerbesserungsvorschläge«	Anzahl der aufgenommenen Verbesserungsvorschläge; z.B. Zahlenwert >= 0 vgl. Brunner, S.105: QKLSA-Kriterien

KennzahlenSixSigma

Nachricht vom Typ Kennzahl, welche die für Six Sigma notwendigen Kennzahlen enthält.

Tabelle 49 Attribute der KennzahlenSixSigma

Attribute	Ausprägung/ Inhalte
»Bereich«	Betrachtungsbereich, auf den sich die Kennzahlen beziehen; z.B. Arbeitsstation-Nr., Kostenstelle oder Fertigungslinien-Nr.
»DPMO«	Defects per Million Opportunities; z.B. Zahlenwert >=0 in Promille
»COPQ«	Costs of poor Quality; z.B. Zahlenwert >=0 in Euro
»RTY«	Rolled Throughput Yield; z.B. Zahlenwert >=0 in %
»SixSigma«	Six-Sigma-Kennzahl; z.B. Zahlenwert >=0

KennzahlIst-Bearbeitungszeit

Nachricht vom Typ Kanban, welche die protokollierten Ist-Bearbeitungszeiten einer Arbeitsstation enthält.

Tabelle 50 Attribute der KennzahlenIst-Bearbeitungszeit

Attribute	Ausprägung/ Inhalte
»KanbanTyp«	Kanban, für die die durchschnittliche Zeit berechnet wurde; z.B. Kanban
»DurchschnittlicherIstBearbeitungszeit«	Durchschnittliche Ist-Bearbeitungszeit auf Basis vergangener Daten für diese Kanban; z.B. Zahlenwert > 0 in h

Lagerdaten

Beschreibt die Lagerbelegung bei einer Arbeitsstation oder einem Supermarkt.

Tabelle 51 Attribute der Lagerdaten

Attribute	Ausprägung/ Inhalte
»SupermarktID«	Identifikationsnummer des Supermarktes; z.B. Supermarkt-Nr. oder Kostenstelle
»Lagerort«	Objekt, welches den Lagerort durch die LagerortID, Meldebestand, zugeordneten Produkten und der maximalen Kapazität näher spezifiziert

Linienstopp

Nachricht vom Typ Ausnahme. Beschreibt, dass die Fertigungslinie gestoppt werden muss. Alle betroffenen Objekte (Arbeitsstationen) führen ihren Takt noch zu Ende und stoppen danach die Arbeit.

Tabelle 52 Attribute der Linienstopp

Attribute	Ausprägung/ Inhalte
»Zeitstempel«	Zeitpunkt des Stopps; z.B. Zeitstempel im Format YYYY-MM-dd HH:MM:ss
»AuslösendeArbeitsstationID«	Identifikationsnummer der Arbeitsstation; z.B. Arbeitsstation-Nr. oder Kostenstelle

Liste8D-Reports

Liste (Array) der 8D-Reports; z.B. Liste mit 8D-Reports.

Tabelle 53 Attribute der Liste8D-Reports

Name:	Inhalt/ Ausprägung
»8D-Report[]«	Liste (Array) mit 8D-Reports

ListeAufträge

Liste (Array) der Kundenaufträge; z.B. Liste mit Aufträgen (Artikelnr. und Anzahl). Aufgrund der Individualität in Unternehmen sind die Attribute nicht näher spezifiziert.

ListeKanban

Liste (Array) mit Kanban; z.B. Liste mit Kanban.

Tabelle 54 Attribute der ListeKanban

Name:	Inhalt/ Ausprägung
»Kanban[]«	Liste (Array) mit Kanban

ListeProzessbeschreibungen

Nachricht vom Typ Liste. Enthält eine Liste mit Prozessbeschreibungen; z.B. Prozessbeschreibungen beschrieben durch Arbeitsstation ID, Beschreibung, Input, Output, Dauer.

Tabelle 55 Attribute der ListeProzessbeschreibungen

Name:	Inhalt/ Ausprägung
»Prozessbeschreibung[]«	Liste (Array) mit Prozessbeschreibungen, die z.B. durch die Attribute Prozess-Nr., -beschreibung, Input, Output und Dauer definiert sind

ListeRüstvorgänge

Liste (Array) mit protokollierten Rüstzeiten; z.B. Liste mit Rüstzeiten für die jeweiligen Produkte.

Tabelle 56 Attribute der ListeRüstvorgänge

Name:	Inhalt/ Ausprägung
»Rüstzeit[]«	Liste (Array) mit Rüstzeiten, die z.B. durch die zugehörige Produkt-Nr., Arbeitsstation-Nr., Dauer der externen Vorbereitung, Dauer der internen Durchführung und Dauer der externen Nachbereitung definiert sind.

ListeStörungen

Nachricht vom Typ Liste. Enthält eine Liste (Array) mit Störungen an einer Arbeitsstation.

Tabelle 57 Attribute der ListeStörungen

Name:	Inhalt/ Ausprägung
»Störungen[]«	Liste (Array) mit Störungen an einer Arbeitsstation, die z.B. durch einen Zeitstempel des Auftretens, dem vorgegebenen Typ der Störung sowie einer Störungsbeschreibung und der Arbeitsstations-Nr. beschrieben sind.

ListeTaktzeiten

Liste (Array) von hinterlegten Taktzeiten (mit Produkt-Nr., Soll-Taktzeit und durchschnittlicher Ist-Taktzeit); z.B. Liste mit Produktbezogener Taktzeiten.

Tabelle 58 Attribute der ListeTaktzeiten

Name:	Inhalt/ Ausprägung
»Taktzeiten[]«	Liste (Array) mit Taktzeiten. Diese wiederum können die Attribute zugehörige Produkt-Nr., Soll-Taktzeit und durchschnittliche Ist-Taktzeit enthalten.

ListeVerbesserungsmaßnahmen

Nachricht vom Typ Liste. Enthält eine Liste mit Verbesserungsmaßnahmen (z.B. Änderungen aus dem Wertstromdesign), welche nicht näher spezifiziert ist.

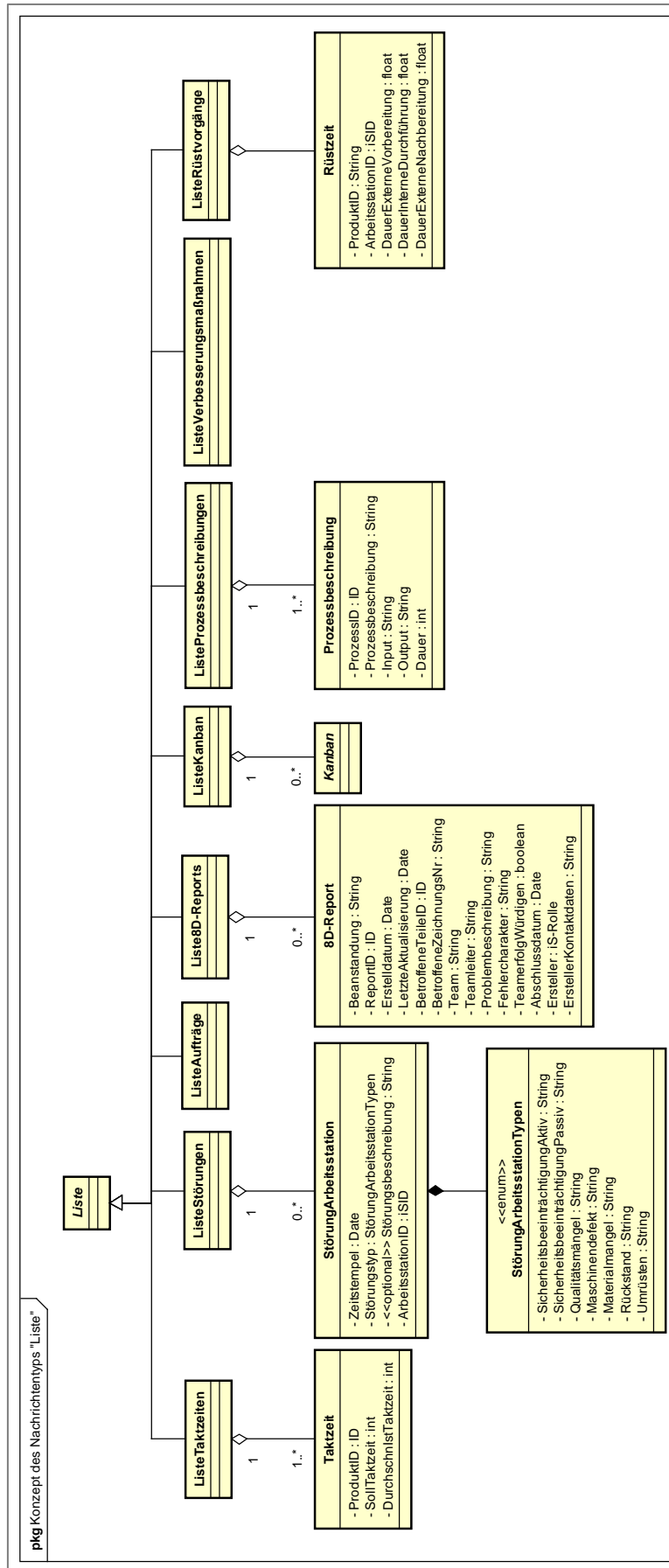


Abbildung 96 Überblick über die unterschiedlichen Listentypen und ihre Inhalte

Meldebestand

Meldebestand an einer Arbeitsstation für ein Vorprodukt, bei welchem der vorgelagerten Arbeitsstation eine Kanban zugeschickt wird.

Tabelle 59 Attribute der Meldebestand

Attribute	Ausprägung/ Inhalte
»ProduktID«	Produktnummer, auf das sich der Meldebestand bezieht; z.B. Artikelnummer
»Meldebestand«	Meldebestand, bei dessen Unterschreitung eine Kanban geschickt wird; z.B. Zahlenwert > 0
»ArbeitsstationID«	Identifikationsnummer der Arbeitsstation; z.B. Arbeitsstation-Nr. oder Kostenstelle

PokaYokeDokumentation

Dokumentiertes Poka Yoke Beispiel, was umgesetzt und dessen Erfolg nachgewiesen ist.

Tabelle 60 Attribute der PokaYokeDokumentation

Attribute	Ausprägung/ Inhalte
»Problembeschreibung«	Beschreibung des gelösten Problems; z.B. Text
»Lösungsbeschreibung«	Beschreibung der zur Problemlösung umgesetzten und evaluierten Poka-Yoke-Lösung; z.B. Text
»Anhang«	Objekt vom Typ Anhang, z.B. Fotos, Video oder andere frei wählbare Objekte

Prozessbeschreibung

Nachricht, welche einen Prozess beschreibt. Den Prozess kann eine Instanz ausführen.

Tabelle 61 Attribute der Prozessbeschreibung

Attribute	Ausprägung/ Inhalte
»ProzessID«	Identifikationsnummer der iS-Rolle; z.B. Arbeitsstation-Nr. oder Kostenstelle
»Prozessbeschreibung«	Beschreibung der durchgeführten Aktivität; z.B. Text
»Input«	Erwartete Eingabe/ Vormaterial für die Durchführung der Aktivität; z.B. Text
»Output«	Gelieferte Ausgabe/ Ergebnis; z.B. Text
»Dauer«	Dauer für den Prozess; z.B. Zahlenwert ≥ 0 in Minuten

Prüfungsanweisung

Prüfungsanweisung für ein Produkt an einer Arbeitsstation zur Kontrolle der Qualität nach Fertigstellung an der Arbeitsstation.

Tabelle 62 Attribute der Prüfungsanweisung

Attribute	Ausprägung/ Inhalte
»PrüfungID«	Identifikationsnummer der Prüfungsanweisung; z.B. Fortlaufende Nummer oder Zeitstempel
»ArbeitsstationID«	Identifikationsnummer der Arbeitsstation, auf die sich die Prüfungsanweisung bezieht; z.B. Arbeitsstation-Nr. oder Kostenstelle
»ProduktID«	Identifikationsnummer des Produktes, auf das sich die Prüfungsanweisung bezieht; z.B. Arbeitsstation-Nr. oder Kostenstelle
»Prüfungsanweisung«	Beschreibung der zu prüfenden Dinge bzw. der Prüfungsschritte; z.B. Text

Qualitätsfehler

Beschreibt einen identifizierten Qualitätsfehler, welcher z.B. mittels Poka Yoke zu beheben ist.

Tabelle 63 Attribute der Qualitätsfehler

Attribute	Ausprägung/ Inhalte
»Fehlerbeschreibung«	Beschreibung des aufgetretenen Qualitätsfehlers; z.B. Text

Qualitätsparameter

Nachricht, welche für eine Komponente an einer Arbeitsstation Qualitätsparameter definiert. Dies sind für regelmäßig aufgenommene Messwerte Warn- und Eingriffsgrenzen, die mitgeteilt werden sollen bzw. ein Eingreifen erfordern.

Tabelle 64 Attribute der Qualitätsparameter

Attribute	Ausprägung/ Inhalte
»ArbeitsstationID«	Identifikationsnummer der Arbeitsstation; z.B. Arbeitsstation-Nr. oder Kostenstelle
»Komponentenparameter[]«	Objekt, welches die Parameter einer Komponente beschreibt. Diese ist durch den Komponentennamen, Parametername, Ist-Wert, obere und untere Warngrenze sowie obere und untere Eingriffsgrenze definiert

StatusArbeitsstation

Liefert den Status einer Arbeitsstation gemäß der Andon-Abstufung (z.B. 5-Stufen-Modell in VDI 2870-2). Die Information wird für die Andon-Tafel sowie die Visualisierung des Arbeitsstations-Zustands verwendet.

Tabelle 65 Attribute der StatusArbeitsstation

Attribute	Ausprägung/ Inhalte
»Status«	Status gemäß Andon-Einteilung; z.B. 5-stufiges Model der VDI-Richtlinie 2870 (Ausprägungen: Status 0 (i.O.), Status 1 (Hilfe benötigt), Status 2 (Hilfe kommt), Status 3 (zusätzliche Hilfe erforderlich) und Status 4 (Stopp))
»ArbeitsstationID«	Identifikationsnummer der Arbeitsstation; z.B. Arbeitsstation-Nr. oder Kostenstelle
»StatusArbeitsstationTypen«	

StatusKanban

Beschreibt den Bearbeitungsstatus einer Kanban.

Tabelle 66 Attribute der StatusKanban

Attribute	Ausprägung/ Inhalte
»betroffeneKanbanID«	Kanban; z.B. Gesamte Kanban oder deren ID als Referenz
»Status«	Kanban-Status gemäß Vorgabetypen; z.B. Vorgabe-Typen: "Fertig", "Verzögerung", "Anwesend"
»Zeitstempel«	Zeitpunkt der Erstellung des Status; z.B. Zeitstempel im Format YYYY-MM-dd HH:MM:ss

StatusLinienstopp

Übermittelt eine Statusaktualisierung für einen vorausgehenden Linienstopp. Wird u.a. verwendet, um nach einer behobenen Störung die Linie wieder zu starten.

Tabelle 67 Attribute der StatusLinienstopp

Attribute	Ausprägung/ Inhalte
»Status«	Identifikationsnummer der Arbeitsstation; z.B. Arbeitsstation-Nr. oder Kostenstelle
»AuslösendeArbeitsstationID«	Status des Linienstopps; z.B. Stopp oder Beendet

StatusVerbesserungsmaßnahme

Nachricht zur Beschreibung des Status einer Verbesserungsmaßnahme mit den festen Ausprägungen „Offen“ (Verbesserungsmaßnahme noch nicht begonnen), „Bearbeitung“ (Verbesserungsmaßnahme in Umsetzung), „Erledigt“ (Verbesserungsmaßnahme abgeschlossen (umgesetzt oder verworfen), „Planung“ (Verbesserungsmaßnahme in Planung) und „Prüfung“ (Erfolg der Verbesserungsmaßnahme in Prüfung).

StatusVerbesserungsvorschlag

Empfängt eine Aktualisierung zu einem zuvor eingereichten Verbesserungsvorschlag.

Tabelle 68 Attribute der StatusVerbesserungsvorschlag

Attribute	Ausprägung/ Inhalte
»ZugehörigeVorschlagsID«	Identifikationsnummer des Verbesserungsvorschlags; z.B. Zeitstempel im Format YYYY-MM-dd hh:ss
»Status«	Status des Verbesserungsvorschlags; z.B. Ausprägungen "Offen", "Bearbeitung", "Umgesetzt", "Verworfen"
»Kommentar«	Ergänzendes Kommentar zum Status; z.B. Text

Verbesserungsmaßnahme

Empfängt eine Verbesserungsmaßnahme, die die iS-Rolle umsetzen soll. Z.B. eine temporäre Behebung einer Störung an einer Arbeitsstation, welche der Hancho übermittelt hat.

Tabelle 69 Attribute der Verbesserungsmaßnahme

Attribute	Ausprägung/ Inhalte
»Verantwortlicher«	Für die Umsetzung verantwortliche Entität; z.B. String
»Beschreibung«	Beschreibung der Verbesserungsmaßnahme; z.B. String
»Status«	Status der Umsetzung; z.B. Vorgabe-Typen (nach PDCA): "Offen", "Planung", "Bearbeitung", "Prüfung", "Erledigt"
»VerbesserungsID«	Identifikationsnummer des Verbesserungsvorschlags; z.B. Zeitstempel im Format YYYY-MM-dd hh:ss

Verbesserungsvorschlag

Vorschlag für einen Verbesserungsvorschlag im Sinne des KVP-Ansatzes.

Tabelle 70 Attribute der Verbesserungsvorschlag

Attribute	Ausprägung/ Inhalte
»EinreicherID«	Identifikationsmerkmal des Einreichers; z.B. Arbeitsstation-Nr. oder Kostenstelle
»BetroffenerBereichID«	Identifikationsmerkmal für den betroffenen Bereich; z.B. Arbeitsstation-Nr. oder Kostenstelle
»Beschreibung«	Beschreibung des Verbesserungsvorschlags; z.B. Text
»DatumEinreichung«	Datum der Einreichung des Verbesserungsvorschlags; z.B. Zeitstempel im Format YYYY-MM-dd

WertstromanalyseAuftragswarteschlange

Nachricht, welche die Auftragswarteschlange der Auftragsfreigabe enthält.

Tabelle 71 Attribute der WertstromanalyseAuftragswarteschlange

Attribute	Ausprägung/ Inhalte
»AnzahlAufträge«	Anzahl der zum Zeitpunkt der Ist-Aufnahme wartenden Aufträge; z.B. Zahlenwert > 0
»Warteschlange«	Formel: = Anzahl wartender Aufträge * Kundentakt; z.B. Zahlenwert > 0 in h

WertstromanalyseKundenbedarfe

Beschreibt die Bedarfe der betrachteten Kundengruppe für die ausgewählte Produktgruppe.

Tabelle 72 Attribute der WertstromanalyseKundenbedarfe

Attribute	Ausprägung/ Inhalte
»Jahresstückzahl«	Jahresabsatzmenge der Produktgruppe; z.B. Zahlenwert > 0
»AnteilDerKundengruppe«	Prozentualer Anteil der betrachteten Kundengruppe am Gesamtabsatz; z.B. Prozentwert
»Fabriktag«	Arbeitstage pro Jahr laut Fabrikkalender; z.B. Zahlenwert > 0
»ArbeitszeitInStunden«	Effektive Arbeitsstunden pro Tag (exkl. Stillstand und Pause; z.B. Zahlenwert > 0
»Kundentakt«	Formel: = (Fabriktag * Arbeitszeit) / Jahresstückzahl; z.B. Zahlenwert > 0 in h pro Stück

WertstromanalyseLagerdaten

Liefert die für die Wertstromanalyse notwendigen Lagerdaten zu einer Produktgruppe

Tabelle 73 Attribute der WertstromanalyseLagerdaten

Attribute	Ausprägung/ Inhalte
»LagerID«	Identifikationsnummer des Lagerplatzes; z.B. Lagerplatz-Nr. oder Kostenstelle
»Lagerort«	Bezeichnung des Lagerplatzes; z.B. Text
»AnzahlLagerplätze«	Kapazität des Lagerplatzes; z.B. Zahlenwert > 0
»Bestandsmenge«	Zum Zeitpunkt der Ist-Aufnahme vorhandener Bestand; z.B. Zahlenwert > 0
»AnzahlTeileJeProdukt«	Anzahl der verwendeten Bauteile für die Fertigung der Produktgruppe; z.B. Zahlenwert > 0
»Reichweite«	Formel: = Bestandsmenge / (Anzahl Gleichteile * Jahresabsatzmenge); z.B. Zahlenwert > 0 in z.B. Tagen

WertstromanalyseMaterialflussdaten

Beschreibt für die Wertstromanalyse alle Daten, die den Materialfluss und Lagerungen betreffen.

Tabelle 74 Attribute der WertstromanalyseMaterialflussdaten

Attribute	Ausprägung/ Inhalte
»TransportInnerbetrieblich«	Kennzeichnet ob es sich um einen innerbetrieblichen oder außerbetrieblichen Transport handelt; z.B. boolean
»Fördermittel«	Beschreibung des Fördermittels; z.B. String

WertstromanalyseProduktgruppe

Beschreibt die betrachtete Produktgruppe für die Wertstromanalyse.

Tabelle 75 Attribute der WertstromanalyseProduktgruppe

Attribute	Ausprägung/ Inhalte
»Produktfamilie«	In der Wertstromplanung betrachtete Produktgruppe bzw. -familie; z.B. String
»Variantenanzahl«	Anzahl der möglichen Produktvarianten zur Beurteilung der Heterogenität; z.B. Zahlenwert > 0
»Kunden(gruppe) «	Zusammenfassung ähnlicher Kunden mit ähnlichen Konditionen; z.B. String
»RepräsentantID«	Die Produktfamilie repräsentierendes Produkt bzw. deren ID; z.B. Fortlaufende Nummer oder Zeitstempel

WertstromanalyseProduktionsprozessdaten

Beschreibt die Daten eines Produktionsprozesses (=Arbeitsstation).

Tabelle 76 Attribute der WertstromanalyseProduktionsprozessdaten

Attribute	Ausprägung/ Inhalte
»ArbeitsstationID«	Identifikationsnummer der Arbeitsstation; z.B. Arbeitsstation-Nr. oder Kostenstelle
»AnzahlMitarbeiter«	Anzahl der eingesetzten Mitarbeiter an der Arbeitsstation; z.B. Zahlenwert >= 0
»AnzahlBetriebsmittel«	Anzahl der eingesetzten Betriebsmittel an der Arbeitsstation; z.B. Zahlenwert >= 0
»GemeinsameRessourcennutzung«	Anzahl der parallel gleichzeitig nutzbaren Ressourcen; z.B. Zahlenwert >= 0
»Bearbeitungszeit«	Dauer der Bearbeitung pro Stück (inkl. Rüstzeiten und Nebenzeiten); z.B. Zahlenwert > 0 in h. Die Bearbeitungszeit (BZ) gibt an, wie lange ein Teil bearbeitet wird (Abb. 2.15, Fall 1a). Diese Zeit beinhaltet sowohl den manuellen Arbeitsinhalt für den Mitarbeiter als auch die Laufzeit des Betriebsmittels. Beim Fräsen wäre das also der Zeitbedarf zum Aufspannen des Teils, zum Programmieren der Maschine, zum Fräsen und schließlich zur Entnahme des Teils. Außerdem sind alle unmittelbar mit dem Bearbeitungsvorgang verbundenen Nebenzeiten (NZ) enthalten. Der Zeitanteil an der Bearbeitungszeit für Aktivitäten, die das Produkt tatsächlich so verändern, wie der Kunde es haben möchte, wird als Wertschöpfungszeit (WZ) bezeichnet (Abb. 2.15, Fall 1b). Die Zuordnung von Tätigkeiten zu diesen Kategorien setzt eine detaillierte Zeitaufnahme voraus, die man beispielsweise im Anschluss an eine Wertstromanalyse zur Prozessoptimierung durchführen kann.
»Prozesszeit«	Dauer von Teilen im Produktionsprozess wenn mehrere Teile parallel gefertigt werden (Chargenfertigung); z.B. Zahlenwert > 0 in h. Die Prozesszeit (PZ) gibt an, wie lange sich Teile im jeweiligen Produktionsprozess befinden, sofern mehrere Teile gleichzeitig bearbeitet werden. Im Durchlaufprozess werden die Teile einzeln zu- und abgeführt (Abb. 2.15, Fall 2a), während im Chargenprozess die Teile in Losen gleichzeitig bearbeitet werden (Abb. 2.15, Fall 2b). Beispiel für den ersten Fall ist ein Durchlaufofen, für den zweiten Fall das Härten oder Waschen von Teilen in Körben.
»Prozess-Menge«	Mit jedem Durchgang gefertigte Anzahl an Produkten; z.B. Zahlenwert > 0
»Zykluszeit«	Formel: = (Bearbeitungszeit * Anzahl Teile je Produkt) / Anzahl Ressourcen; z.B. Zahlenwert > 0. Spezifisch für die Wertstrommethode ist die Angabe der Zykluszeit (ZZ). Diese Zeit gibt an, nach welchem Zeitintervall ein Teil oder ein Produkt in einem Produktionsprozess fertiggestellt wird (Abb. 2.15, Fall 4). Steht nur ein Betriebsmittel zur Verfügung, dann sind Bearbeitungszeit und Zykluszeit gleich. Wenn beispielsweise ein Monteur zehn Minuten für die Montage eines Produktes benötigt, dann wird auch alle zehn Minuten ein Produkt fertiggestellt. Verdoppelt man die Zahl der Ressourcen, dann halbiert sich die Zykluszeit. Im Beispiel liefern zwei Monteure alle fünf Minuten ein Fertigprodukt ab. Bei Durchlauf- und Chargenprozessen ergibt sich die Zykluszeit durch Division der Prozesszeit durch die Anzahl der Teile im Prozess. Bei einem Durchlaufofen mit 100 Teilen und einer Prozessdauer von 100 Minuten beträgt demnach die Zykluszeit eine Minute. Die Zykluszeit gibt die Leistungsfähigkeit des Produktionsprozesses in Zeiteinheiten bei kontinuierlichem Betrieb ohne rüst- oder störungsbedingte Unterbrechungen wieder. Die Zykluszeit beschreibt das Kapazitätsangebot eines Produktionsprozesses in Bezug auf den Kundenbedarf. Er gibt an, welcher als Kundentakt formulierte Kundenbedarf maximal gedeckt werden kann.
»AnzahlTeileJeProdukt«	Anzahl Gleichteile pro Endprodukt in Stück; z.B. Zahlenwert > 0
»Rüstzeit«	Dauer, während der ein Betriebsmittel aufgrund eines Wechsels von Vorrichtungen o.ä. nicht für die Bearbeitung zur Verfügung steht. Die Rüstzeit (RZ) ist die Zeit, während der ein Betriebsmittel aufgrund eines Wechsels der Vorrichtungen, Werkzeuge oder Materialien (Farbe, Coil, Granulat) für eine neue Teilevariante nicht für die Bearbeitung zur Verfügung steht (Abb. 2.15, Fall 3). Sie wird gemessen vom letzten Gutteil der vorhergehenden bis zum ersten Gutteil der folgenden Variante.
»Losgröße«	Anzahl gefertigter Gleichteile in direkter Abfolge im Produktionsprozess; z.B. Zahlenwert > 0
»AnzahlTeilevarianten«	Anzahl der möglichen Teilevarianten an der Arbeitsstation; z.B. Zahlenwert > 0
»Verfügbarkeit«	Anteil der Rüstzeit und Maschinenbelegungszeit an der Gesamtzeit; z.B. Zahlenwert > 0 in %

»EPEI-Wert«	<p>Every Part – Every Intervall; Zeitraum, um alle Varianten einmal zu fertigen. Der EPEI-Wert im Ist-Zustand eines Produktionsprozesses ergibt sich aus der Summe der Bearbeitungszeit für alle Produktvarianten in den jeweils vorgegebenen Losgrößen zuzüglich der notwendigen Rüstzeiten sowie geplanter und ungeplanter Stillstände. Dieser Wert besagt, wie lange es unter den aktuellen Bedingungen dauert, bis alle Varianten einmal produziert worden sind. Im Unterschied zu der bloßen Angabe von Rüstzeiten und Losgrößen kann man aus diesem Wert sehr leicht ablesen, wie flexibel ein Produktionsprozess momentan ist. $EPEI = (Summe\ Bearbeitungszeiten + Summe\ Rüstzeiten) / (Anzahl\ Ressourcen * \text{tägl.}\ Arbeitszeit)$</p>
»GutausbeuteAnteil«	Anteil der Gutteile; z.B. Zahlenwert > 0 in %
»NacharbeitAnteil«	Anteil der Nacharbeitsteile; z.B. Zahlenwert > 0 in %
»Prozess-ArbeitszeitJeTag«	Arbeitszeit je Tag für den betrachteten Prozess; z.B. Zahlenwert > 0 in h
»Prozess-Jahresstückzahl«	Jahresstückzahl für den betrachteten Prozess; z.B. Zahlenwert > 0
»Prozess-Kundentakt«	<p>Korrekturwert für den Kundentakt unter Berücksichtigung von Qualitätsproblemen und Sonderlösungen; z.B. Der prozessspezifische Kundentakt ist ein Korrekturwert zum Kundentakt, der übergreifend für den gesamten Wertstrom gilt. Hier fließen Qualitätsprobleme und Sonderlösungen für die prozessspezifische Arbeitszeit ein. Relevant ist dieser Wert auch bei uneinheitlich gebildeten Produktfamilien, die zu verzweigten, also komplexeren Wertströmen führen. Insgesamt kann man das prozentual ausgedrückte Verhältnis zwischen Kundentakt und prozessspezifischem Kundentakt auch als Qualitätsmerkmal für den betrachteten Wertstrom auffassen. Dieser Wert gibt an, wie weit Kapazitätsangebot und Variantenflexibilität eingeschränkt werden. $Prozess-Kundentakt = Kundentakt * (Prozess-Arbeitszeit / Arbeitszeit\ pro\ Tag) * (Jahresstückzahl / Jahresstückzahl\ Prozessschritt)$</p>

Liste der betreuten studentischen Arbeiten

- [Dil17] Diluckshan Theivendram: Konzeption und Realisierung einer digitalen Schnittstelle für Anlagen zur Realisierung von Lean Methoden. Kaiserslautern, TU Kaiserslautern, Lehrstuhl für Werkzeugmaschinen und Steuerungen. Studienarbeit. 2017
- [Her17] Herrmann, L.: Vergleich bestehender Kommunikationsprotokolle und Informationsmodelle. Kaiserslautern, TU Kaiserslautern, Lehrstuhl für Produktionsautomatisierung. Studienarbeit. 2017
- [Iac15] Iacob, A.: Integration model for a CPS-controlled working station providing Just-In-Time in lean production. Kaiserslautern, TU Kaiserslautern, pak. Studienarbeit. 2015
- [Jed16] Jedermann, E.: Modulare Erweiterung ereignisbasierter Fertigungssteuerung in heterogener Umgebung am Beispiel eines Demonstrators der SmartFactoryKL. Kaiserslautern, TU Kaiserslautern, Arbeitsgruppe Robotersysteme, Fachbereich Informatik. Bachelorarbeit. 2016
- [Kno15] Knobloch, J. B.: Entwurf eines Frameworks für SOA-basierte, Cyber-Physische Arbeitsstationen in der Lean Production. Kaiserslautern, TU Kaiserslautern, Arbeitsgruppe Robotersysteme, Fachbereich Informatik. Bachelorarbeit. 2015
- [Kno17] Knobloch, J. B.: Design and Implementation of a modular CPS-Interface for a flexible Integration of Workstations in the Lean Production. Kaiserslautern, TU Kaiserslautern, Arbeitsgruppe Robotersysteme, Fachbereich Informatik. Masterarbeit. 2017
- [Leh16] Lehr, G.: Betriebswirtschaftliche Bewertung von Industrie 4.0 Anwendungen in der Lean Production. Kaiserslautern, TU Kaiserslautern, Lehrstuhl für Strategie, Innovation und Kooperation. Bachelorarbeit. 2016
- [Mar15] Marseu, E.: Konzeption einer Entwicklungsmethodik für Cyber-Physische Produktionsanlagen und Evaluierung in der Praxis. Kaiserslautern, TU Kaiserslautern, pak. Diplomarbeit. 2015
- [Sch17] Schmitz, P.: Konzeption und Realisierung einer digitalen Shopfloor-Tafel für die SmartFactory-KL. Kaiserslautern, TU Kaiserslautern, Lehrstuhl für Werkzeugmaschinen und Steuerungen. Forschungsprojekt. 2017
- [Vir15] Virendra Kumar Ashiwal: Development of an information model based on OPC UA to support intelligent networking of Cyber Physical Systems. Kaiserslautern, TU Kaiserslautern, Lehrstuhl für Produktionsautomatisierung. Masterthesis. 2015

Lebenslauf

Persönliche Daten

Name: Dennis Kolberg
Email: mail@denniskolberg.de

Ausbildung

2010-10 – 2013-12 Studium Wirtschaftsingenieurwesen (M.Sc.) an der Universität Bremen
2011-08 – 2012-01 Studium Informatik (ERASMUS) an der Lappeenranta University of Technology in Finnland
2017-10 – 2010-10 Studium Wirtschaftsingenieurwesen (B.Sc.) an der Universität Bremen
1994 – 2003-06 Abitur am Besselgymnasium der Stadt Minden

Berufliche Erfahrungen und Praktika

2017-02 – 2018-03 Senior Researcher und Wissenschaftlicher Leiter im Forschungsbereich Innovative Fabriksysteme beim Deutschen Forschungszentrum für Künstliche Intelligenz GmbH in Kaiserslautern
2014-03 – 2017-01 Researcher und u.a. Teamleiter bei der Technologie-Initiative SmartFactory KL e.V. in Kaiserslautern
2008-03 – 2014-02 Selbstständiger Unternehmensberater in Bremen
2013-02 – 2013-06 Interim-Projektleiter bei der FAUN Umwelttechnik GmbH & Co. KG in Osterholz-Scharmbeck
2012-05 – 2012-09 Praktikant bei der Campana & Schott Realisierungsmanagement GmbH in Köln
2007-02 – 2007-09 Disponent in der Materialwirtschaft bei der RK Rose+Krieger GmbH in Minden
2004-08 – 2007-01 Auszubildender zum Industriekaufmann bei der RK Rose+Krieger GmbH in Minden

Übersicht der Fortschritt-Berichte pak

Band 1 (1999)	Romberg, Markus	Konzept und Entwicklung eines Redaktionssystems zur Produktion von Anwenderdokumentationen – Anwendergerechte Online und Papierdokumentation aus einer Datenquelle
Band 2 (1999)	Kliwer, Rüdiger	Reverse Engineering von Steuerungs-Software – Ein Beitrag zur besseren Beherrschung der Software-Wartung für speicherprogrammierbare Steuerungen
Band 3 (1999)	Wahl, Martin	Systematische Entwicklung nutzergerechter Maschinenoberflächen – Ein Software-Werkzeug mit integrierten Ergonomie- und Gestaltungsregeln
Band 4 (2001)	Eissler, Ralf	Useware-Engineering in der Produktionstechnik – Systemtechnische Methodik zur Entwicklung ergonomischer Bediensysteme für komplexe Maschinen
Band 5 (2002)	Röse, Kerstin	Methodik zur Gestaltung interkultureller Mensch-Maschine-Systeme
Band 6 (2002)	Schröder, Christoph	Integration von Sensorik in die visuelle Roboterprogrammierung – Datenkonzept und Realisierung am Beispiel typischer Sensoren der industriellen Robotik
Band 7 (2003)	Krauss, Lutz	Entwicklung und Evaluation einer Methodik zur Untersuchung von Interaktionsgeräten für Maschinen- und Prozessbediensysteme mit grafischen Benutzungsoberflächen
Band 8 (2003)	Reuther, Achim	useML – Systematische Entwicklung von Maschinenbediensystemen mit XML
Band 9 (2004)	Zhao, Binzang	Modellierung und Evaluation von 3D-Eingabegeräten für die Maschinenbedienung
Band 10 (2005)	Schlick, Jochen	Steigerung der Prozesssicherheit automatisierter Mikromontagen durch Diagnose und Kompensation von Störungen
Band 11 (2006)	Long, Liu	Localization of Human-Machine Systems for Production Machines in Mainland-China
Band 12 (2006)	Mukasa, Kizito	Modellbasierte Generierung prototypischer Bedienschnittstellen – Ein Gestaltungswerkzeug zur formalen Beschreibung generischer, konsistenter Bedienschnittstellen mit XML
Band 13 (2007)	Oortmann, Holger	Workspace – Vom integrierten Plattformkonzept für die Produktion zum exemplarischen 3D-Userinterface
Band 14 (2007)	Bödcher, Alexander	Methodische Nutzungskontext-Analyse als Grundlage eines strukturierten USEWARE-Entwicklungs-Prozesses
Band 15 (2007)	Ehrmann, Markus	Beitrag zur Effizienzsteigerung bei der Programmierung flexibler, roboterbasierter Montagezellen – Konzeption und Realisierung eines nutzergerechten Programmiersystems
Band 16 (2007)	Bock, Carsten	Einsatz formaler Spezifikation im Entwicklungsprozess von Mensch-Maschine-Schnittstellen
Band 17 (2008)	Pohlmann, Eric	Methodik zur prozessorientierten Planung serviceorientierter Fabriksteuerungssysteme
Band 18 (2008)	Thiels, Nancy	Personalisierung von technischen Maschinenbenutzungsoberflächen auf der Grundlage einer erweiterten Analysephase

Band 19 (2008)	Seckner, Marc	Unterstützung automatisierter Mikroproduktion durch wandlungsfähige Montageanlagen – Konzeption und Realisierung einer flexiblen und reaktiven Mikromontage
Band 20 (2009)	Görlich, Daniel	Laufzeit-Adaption von Benutzungsschnittstellen für Ambient-Intelligence-Umgebungen mittels Raumbasierter Benutzungsmodelle
Band 21 (2010)	Meixner, Gerrit	Entwicklung einer modellbasierten Architektur für multimodale Benutzungsschnittstellen
Band 22 (2012)	Stephan, Peter M.	Entwicklung einer Referenzarchitektur zur Nutzung semantisch interpretierter Ortsinformationen am Beispiel der Instandhaltung
Band 23 (2012)	Flörchinger, Florian	Entwicklung einer generischen Geräteidentifikation mit industriellen Feldgeräten mittels mobiler Bediengeräte
Band 24 (2013)	Seißler, Marc	Modellbasierte Entwicklung kontextsensitiver Benutzungsschnittstellen zur Unterstützung der mobilen Instandhaltung
Band 25 (2013)	Loskyll, Matthias	Entwicklung einer Methodik zur dynamischen kontextbasierten Orchestrierung semantischer Feldgerätekombinationen
Band 26 (2013)	Hodek, Stefan	Methode zur vollautomatischen Integration von Feldgeräten in industrielle Steuerungssysteme
Band 27 (2014)	Gerber, Tobias	Methodology for a Flexible and Transparent Allocation of Production Process KPI to Business Software Systems
Band 28 (2014)	Kümmerling, Moritz	Methodische Entwicklung einer Referenzarchitektur zur nutzeradaptiven Mensch-Maschine-Interaktion in Verkehrsmitteln mittels plattformunabhängiger Mobilitätsprofile
Band 29 (2014)	Gorecky, Dominic	Entwicklung einer Methodik zur informationstechnischen Integration von virtuellem Training innerhalb der Digitalen Fabrik
Band 30 (2014)	Orfgen, Marius	Methodische Entwicklung eines Spezifikationsformats für automotiv Infotainmentsysteme
Band 31 (2015)	M. Reboredo, J. Pedro	Semantische Feldgeräte-Integrationsplattform Ein Beitrag zur durchgängigen Integration von Feldgeräten in Fabriknetzwerke
Band 32 (2015)	Zhang, Ran	Referenzarchitektur zur Erhöhung der Flexibilität in der GUI-Applikationsanbindung für automotiv Infotainmentsysteme
Band 33 (2016)	Schmitt, Mathias	Methodische Entwicklung einer Referenzarchitektur zur dynamischen Adaption von Feldgerätekombinationen mittels modularer Softwareanwendungen
Band 34 (2017)	Ollinger, Lisa	A Model-driven Engineering Methodology for the Development of Service-oriented Control Procedures for Automated Manufacturing Processes
Band 35 (2018)	Quint, Fabian	Entwicklung einer Referenzarchitektur zur Integritätssicherung von Cyber-Physischen Systemen in dynamischen Produktionsumgebungen
Band 36 (2018)	Weyer, Stephan	Methodik zur informationstechnischen Integration Cyber-Physischer Produktionssysteme innerhalb der Digitalen Fabrik
Band 37 (2018)	Kolberg, Dennis	Entwicklung einer Referenzarchitektur zur Realisierung von Methoden der Lean Production mittels digitaler Technologien
