

Masterarbeit

INTELLIGENTE PRODUKTION DURCH DIGITALISIERUNG

ausgeführt am



FACHHOCHSCHULE DER WIRTSCHAFT

Fachhochschul-Masterstudiengang
Automatisierungstechnik-Wirtschaft

von

Marco Anesi, BSc

1510322006

betreut und begutachtet von

FH-Prof. Dipl.-Ing. Dieter Lutzmayr

Graz, im November 2016

.....
Unterschrift

EHRENWÖRTLICHE ERKLÄRUNG

Ich erkläre ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benützt und die benutzten Quellen wörtlich zitiert sowie inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

.....
Unterschrift

DANKSAGUNG

Diese Zeilen möchte ich den Menschen widmen, die mich beim Verfassen dieser Arbeit unterstützten und mir mit Rat und Tat zur Seite gestanden sind. Der erfolgreiche Abschluss dieser wissenschaftlichen Abhandlung konnte nur mit Hilfe von vielen Menschen erreicht werden, denen ich an dieser Stelle allerherzlichst danken möchte!

Weiters möchte ich mich bei Herrn Mario Prem, Herrn Gerald Holzerbauer und Herrn Erich Paierl von Rosendahl Nextrom, sowie bei Herrn Dipl.-Ing. Jürgen Steinbrenner von Siemens bedanken, die diese Arbeit erst möglich gemacht haben. Danke für die Betreuung der Arbeit und für die fachliche Unterstützung.

Danken möchte ich aber auch meinen Eltern, die mich über die gesamte Zeit hinweg wohlwollend durch das Studium und auch durch die Masterarbeit begleitet haben und so manche Verpflichtung von mir fernhielten, damit ich diese Arbeit fertig stellen konnte.

Einen ebenfalls großen Anteil am Erfolg hat meine Lebensgefährtin Sandra. Auch Dir möchte ich danken, dass Du mich tatkräftig unterstützt hast und mir die Zeit eingeräumt hast, die ich für die Fertigstellung der Arbeit benötigte.

Mein Dank gilt Herrn FH-Professor Dipl.-Ing. Dieter Lutzmayr, der meine Masterarbeit betreut und begutachtet hat. Für die hilfreichen Anregungen und die konstruktive Kritik bei der Erstellung dieser Arbeit möchte ich mich herzlich bedanken.

Auch außerhalb des Umfeldes von Fachhochschule und Familie gibt es Menschen, denen ich meinen tiefsten Dank ausdrücken möchte. Hierzu zählen meine Arbeitskolleginnen und Arbeitskollegen, Vorgesetzte, Freunde und Verwandte bei denen ich mich herzlichst bedanken möchte.

Marco Anesi

KURZFASSUNG

Die vorliegende wissenschaftliche Arbeit beschäftigt sich mit der Umsetzbarkeit der derzeitigen Entwicklung von Produktionsanlagen im Kontext von Industrie 4.0 und der einhergehenden Digitalisierung der Maschinen. Es wird ein Bezug zu den Produktionsanlagen der Firma Rosendahl Nextrom GmbH gezogen, um mögliche Use-Cases für die spezifische Branche, in der sich Rosendahl Nextrom bewegt, zu ermitteln.

Der theoretische Teil der Arbeit beschäftigt sich mit der grundlegenden Idee von Industrie 4.0 und warum die wertschöpfende Industrie Bedarf an Veränderung hat. Desweiteren wird auf technologische Weiterentwicklungen in der Automatisierungstechnik eingegangen, welche einen erheblichen Einfluss auf die Produktionsanlagen von morgen haben werden. Dabei werden sowohl Kommunikationsstandards und Sicherheitskonzepte, als auch cloudbasierte Datenerfassung betrachtet.

Aufgrund der Anwendungsgebiete der cloudbasierten Datenerfassung wurde im praktischen Teil der Extruder, eine Anlage zur Verflüssigung von Kunststoffgranulat, so erweitert, dass diese mittels standardisierter Kommunikation Messdaten an einen Cloudspeicher übermittelt. Hierbei wurde sehr großer Wert auf Datensicherheit und einfache Implementierung der Kommunikation gelegt.

Die Erkenntnis durch das theoretische Wissen und die praktische Anwendung eines Industrie 4.0 Projektes mündete in ein Anwendungsszenarium für Rosendahl Nextrom. Die digitale Anbindung und Speicherung der Messdaten ermöglicht viele verschiedene Auswertungsmöglichkeiten. Durch die Analyse mittels *Machine Learning* Algorithmen sollen Verschleißerscheinungen der Maschine frühzeitig erkannt und entsprechende Gegenmaßnahmen eingeleitet werden.

ABSTRACT

This thesis regards the feasibility of current developments of production facilities and the digitalization of machines. A reference to the production facilities of *Rosendahl Nextrom*, an international machinery construction company focuses on cable industry, is drawn to determine possible best practice scenarios.

The theoretical part of the work deals with the basic idea of *Industry 4.0*, and why the producing industry needs a change. Further technological advances in automation technology are dealt with, many of which will have a significant impact on production facilities in the future. For this purpose, communication standards and security concepts as well as cloud-based data collection are considered.

To implement cloud-based data acquisition, the extruder, a machine which processes plastic, is examined in detail in the practical part. It describes the process by which measurement data are uploaded to cloud storage. Naturally, great importance is placed on data security and easy implementation of the communication interface.

The digital connection and storage of the measured data allows many different evaluation possibilities. The realization of the theoretical analysis and the practical application of an Industrial 4.0 project led to an evaluation scenario for Rosendahl Nextrom. The analysis with machine learning algorithms is intended to recognize signs of exhaustion of the machine at an early stage and to react accordingly.

INHALTSVERZEICHNIS

1	Einleitung.....	1
1.1	Unternehmensbeschreibung - Rosendahl Nextrom GmbH.....	2
1.2	Ausgangssituation	3
1.3	Aufgabenstellung	4
1.4	Zielsetzung.....	4
1.5	Aufbau der Arbeit.....	5
2	Computer Integrated Manufacturing (CIM)	6
2.1	Von CIM zu Industrie 4.0	7
2.2	Referenzarchitekturmodell Industrie 4.0 (RAMI 4.0)	8
2.2.1	Layers - Schichten des Modells.....	9
2.2.2	Hierarchy Levels - Hierarchische Ebenen	10
2.2.3	Value Stream - Wertschöpfung und Lebenszyklus.....	10
2.2.4	Zwischenfazit	11
2.3	Cyber-Physikalische-Systeme.....	12
2.3.1	Industrie 4.0-konforme Kommunikation.....	14
2.3.2	Die Verwaltungsschale - Datencontainer eines CPS	15
2.3.2.1	Aufbau der Verwaltungsschale.....	15
2.3.2.2	Identifikationsmöglichkeiten eines CPS in der Verwaltungsschale.....	16
2.3.3	Internet der Dinge	17
2.3.4	Cyber-Physikalische Produktions-Systeme.....	18
3	Digitalisierung der Produktionsanlagen - Standards zur Umsetzung Industrie 4.0	19
3.1	OPC UA - Kommunikationsstack.....	19
3.1.1	Aufbau des Adressraums	20
3.1.2	Zertifikate	21
3.1.3	X.509-Zertifikat.....	21
3.1.4	Secure Channel	22
3.1.5	Session	24
3.1.6	Zusammenfassung	25
3.2	Cloud-Computing.....	26
3.2.1	Speicherung und Analyse von Daten	27
3.2.2	Betrugserkennung bei Kreditkartenzahlungen durch Big Data und Analyseverfahren	28
3.2.3	Einsatz in der Industrie	28
3.2.4	Arten von Cloud-Computing	29
3.3	Datenanalyse – Von Big Data zu Smart Data	30
3.4	Machine Learning	30
3.5	Datensicherheit – Defense in Depth.....	31
3.5.1	Anlagensicherheit	32
3.5.2	Netzwerksicherheit	32
3.5.3	Systemintegrität	33

4	Der Extruder - ROEX.....	34
4.1	Allgemein	34
4.2	Aufbau.....	35
4.3	Funktionsweise	35
5	Datenanbindung	37
5.1	Vom herkömmlichen Extruder zur CPPS-Komponente.....	37
5.2	OPC UA-Implementierung	37
5.2.1	Hardware	41
5.2.2	Software.....	41
5.2.3	Netzwerkstruktur.....	42
5.2.4	Definition von OPC UA-Objekten.....	42
5.2.5	Adressraum des OPC UA-Servers	43
5.2.6	OPC-UA-Server aktivieren.....	43
5.2.6.1	Einstellungen des OPC-UA-Servers	44
5.2.6.2	Erweitern des Programms um die Datenschnittstelle IoT_Data.....	45
5.3	Cloud-Systeme – Auswahlkriterien.....	46
5.4	Microsoft Azure	48
5.4.1	Device Explorer.....	51
5.4.2	OPC UA Publisher-Gateway.....	53
5.4.3	Azure IoT – Streamanalytics Funktion.....	59
6	Predictive Analytics - Anwendungsprojekt	63
6.1	Historische Daten	63
6.2	Der Abnutzungsvorrat.....	66
6.3	Modellbildung in der Microsoft Azure Cloud	67
6.4	Regression.....	68
6.4.1	Numerisches Ergebnis - Lineare Regression	68
6.4.2	Voraussagen einer Wahrscheinlichkeit – logistische Regression	70
6.4.3	Neuronales Regressions-Modell.....	70
6.4.3.1	Aufbau eines neuronalen Netzes	70
6.4.3.2	Training, Lernverhalten und typische Topologien	71
6.4.4	Boosted Decision Tree Regression	72
6.4.5	Decision Forest Regression.....	73
6.5	Evaluierung der Algorithmen	73
7	Resümee und Ausblick.....	77
	Literaturverzeichnis	79
	Abbildungsverzeichnis.....	81
	Tabellenverzeichnis.....	84
	Abkürzungsverzeichnis.....	85

1 EINLEITUNG

Die Industrie mit allen ihren Branchen und Segmenten erfährt derzeit den Hype *Industrie 4.0* der auf Basis der *Hightech-Strategie* der deutschen Bundesregierung und den zugehörigen Forschungsprojekten im Bereich der Industrie entstanden ist. Aufgrund dessen sprechen viele von einer Revolution der Industrie, welche in unserer Historie nunmehr die Vierte in diesem Kontext wäre. Ob man hierbei von einer echten Revolution oder eher von einer Evolution sprechen kann, sei dahingestellt. Auf jeden Fall beruht dieser Hype auf einer Entwicklung der IT-Branche, welche immer mehr in industriellen Fertigungsanlagen Einzug genommen hat. Diese Tatsache entstand aufgrund von Globalisierung, Kosteneffizienzsteigerung sowie Produktionssteigerungen bei gleichzeitiger Erhöhung der Komplexität der Produkte und damit gekoppelt der Komplexität der Produktion, welche computergestützte Systeme notwendig machte. Durch Technologiesprünge in den letzten dreißig Jahren im Bereich der IT wurden diese Technologien zu Basistechnologien in vielen weiteren Segmenten, unter anderem im Bereich der industriellen Produktion und der Automatisierungstechnik. Die stetige Leistungssteigerung der Komponenten und der dadurch sinkende Preis der etablierten Produkte tat ihr Übriges. Man könnte somit schlussfolgern, dass die kundenindividuelle Massenfertigung der heutigen Zeit durch die Informations-Technologien erst ermöglicht wurde.

Bevölkerungswachstum und der steigende Anteil an Menschen welche im globalen Markt konsumieren, werden stetig steigen. Deshalb wird die Industrie nicht mit mangelnder Nachfrage zu kämpfen haben, jedoch mit den konkurrierenden Volkswirtschaften. Weiteres wird die Individualisierung der Produkte ein so starkes Maß annehmen, dass eine Fertigung eines Produkts bis zu einer Losgröße Eins durchaus denkbar wird. Unternehmen werden sich aufgrund des globalen freien Marktes diesem Trend unterwerfen müssen um am Markt Bestand zu haben. Diese Trends machen es erforderlich die Wertschöpfungsprozesse so effektiv und effizient zu gestalten, dass dieser steigende Bedarf an individuellen Produkten gewinnbringend gedeckt werden kann. Hier wird es notwendig sein, vor allem für produzierende Unternehmen in Europa, neue Möglichkeiten zu finden wie Wertschöpfung in Zukunft gestaltet werden kann, um die Herausforderungen am globalen Markt im Bereich der Hochlohnländer zu meistern. Mit dieser Evolution müssen neue Ideen, Konzepte, Technologien und Standards entwickelt werden welche nun unter dem Schlagwort Industrie 4.0 gesammelt und erforscht werden.¹

Diese Masterarbeit beschäftigt sich mit diesen oben beschriebenen Herausforderungen und wurde in Zusammenarbeit mit dem Unternehmen Rosendahl Nextrom GmbH und der Siemens AG Österreich durchgeführt. Es sollen die Themen der Digitalisierung analysiert und ihre Machbarkeit und vor allem ihren Zweck für die Produkte der Firma Rosendahl Nextrom GmbH ermittelt werden.

Das Ziel ist es, durch Implementierung von Technologien Produktionsprozesse zu flexibilisieren, Transparenz in den laufenden Prozessen zu implementieren und wichtige Datenmengen übergeordneten Systemen zur Verfügung zu stellen.

¹ Vgl. Bauernhansl/Hompel/Vogel-Heuser (2014), S.10ff.

Es sollen die Aspekte des Referenzarchitekturmodells Industrie 4.0 (RAMI 4.0) untersucht werden, welches die Grundlage bildet auf der eine Analyse der Produkte erfolgt, um Handlungsfelder für Rosendahl Nextrom GmbH zu erarbeiten. Diese Arbeit hat nicht den Anspruch, ein fertiges Industrie 4.0 Produkt zu liefern. Vielmehr soll das Thema Industrie 4.0 – Digitalisierung der Produktion - bearbeitet werden, um in Absprache mit Rosendahl Nextrom GmbH einen Mehrwert für deren eigenen Prozesse sowie deren Produkte zu generieren. Aufgrund dessen werden Technologien betrachtet, welche zur Umsetzung dieser Mehrwerte notwendig sind sowie eine Implementierung angestrebt.

Fokus dieser Arbeit stellt die vertikale und horizontale Integration der Systeme dar, um den Wertschöpfungsprozess transparenter und offener zu gestalten. Es werden Technologien in ein Automatisierungskonzept implementiert um Daten abzuschöpfen, Mehrwert für die Entwicklung, Wartung und vorbeugende Instandhaltung zu gewinnen.

1.1 Unternehmensbeschreibung - Rosendahl Nextrom GmbH

Die Rosendahl Nextrom GmbH ist ein in Pischelsdorf ansässiges Unternehmen, das Teil der KNILL Gruppe ist, die sich wiederum in „Knill Technology“ (= Rosendahl Nextrom Gruppe) und Knill Energy untergliedert. Rosendahl Nextrom beschäftigt am Standort Pischelsdorf ca. 310 Mitarbeiter und ist als Großunternehmen eingestuft. Pischelsdorf ist der Hauptsitz der Firma Rosendahl Nextrom GmbH, wo sich die Produktions-, Vertriebs- und Forschungszentrale befindet.

1712		Erste Erwähnung von MOSDORFER als Klingenschmiede in Weiz
1949		Eintritt in den Bereich Freileitungsarmaturen
1985		Eintritt in Bereich Schaltanlagenzubehör
1989		Eintritt in Bereich Schranksysteme
1997		Eintritt in Kabel- und Drahtindustrie
1999		Eintritt in die Nieder- und Mittelspannungsindustrie
2002		Ausbau des Bereichs Schranksysteme
2005		Expansion der Kabel- und Drahtindustrie im Glasfaserbereich
2007		Stärkung in der Glasfaserfertigung
2010		Einstieg in die Batteriemaschinensparte
2012		Expansion im Bereich Schranksysteme 300 jähriges Firmenjubiläum der KNILL Gruppe

Abbildung 1: Rosendahl Nextrom – Firmengeschichte, Quelle: Rosendahl Nextrom GmbH.

Die drei Geschäftsfelder werden von einer gemeinsamen Organisation geleitet. Seit mehr als 50 Jahren entwickelt und produziert Rosendahl Nextrom professionelle Lösungen für die speziellen Anforderungen von Kunden weltweit.

Knill Gruppe:

Die Knill Gruppe ist ein privates Familienunternehmen, das derzeit von Christian und Georg Knill geführt wird. Es wurde im Jahre 1712 das erste Mal erwähnt und ist bereits in der 12. Generation in Familienbesitz.

Der Hauptsitz der Knill Gruppe ist in Weiz (Österreich) situiert und beinhaltet international tätige Firmen. Die Knill Gruppe unterteilt sich in zwei Bereiche mit insgesamt 6 Geschäftsfeldern:

- KNILL Energy
- KNILL Technology (= Rosendahl Nextrom Gruppe)

Die Firmengruppe ist weltweit in 15 Ländern mit insgesamt 30 Unternehmen präsent und tätig, und beschäftigt über 2.000 Mitarbeiter und Mitarbeiterinnen, davon 660 in Österreich.

Geschäftsbereiche der Rosendahl Nextrom GmbH:



Die strategische Geschäftseinheit Kabel und Draht bietet Kunden im Segment Kabel- und Drahtherstellung umfassende Lösungen in den Bereichen Strom-, LAN-, Koaxial- und Glasfaserkabel sowie Verdrahtung für Anwendungen in der Automotive-Branche. Die technologischen Kompetenzen umfassen die Bereiche Extrusion, SZ-Verseilung und Glasfaser. Der Umsatz im Geschäftsbereich Kabel und Draht betrug im Wirtschaftsjahr 2015/16 53 Mio €, Mitarbeiter: ca. 50



Das Geschäftsfeld Glasfaser bietet dem Markt hochtechnologische Fertigungsanlagen zur Glasfasererzeugung und der Fertigung von Glasfaserkabel. Die technologischen Kompetenzen umfassen die Herstellung von Preformen, dem Glasfaserziehen, dem Testen, Färben und der Weiterverarbeitung in Kabel bzw. Kabelbündchen. Der Umsatz im Geschäftsbereich Glasfaser wurde im Laufe des Wirtschaftsjahres 14/15 sukzessive durch die Rosendahl Nextrom GmbH übernommen. Der Gesamtumsatz dieser Firma betrug im Wirtschaftsjahr 2015/16 26,5 Millionen €, Mitarbeiter: ca. 87



Der Bereich *Batterie-Maschinen* ist auf Produktionsequipment im Bereich Batteriemontage fokussiert. Dabei werden Lösungen für alle Bleibatterietypen entwickelt, die in Automobilen, Nutzfahrzeugen, Motorrädern und in der Industrie zur Anwendung kommen. Außerdem werden Projekte im Bereich E-Mobility entwickelt und gemäß spezifischen Kundenanforderungen bereitgestellt. Der Umsatz der SBU Battery Machines im Wirtschaftsjahr 2015/16 betrug 13,50 Millionen €, Mitarbeiter: ca. 25

1.2 Ausgangssituation

Die Firma Rosendahl Nextrom GmbH ist im Sondermaschinenbau tätig und produziert unter anderem Anlagen für die Kabelindustrie. Diese Anlagen bestehen aus einzelnen Modulen, welche über das hausintern entwickelte Produktionsleitsystem verknüpft sind. Somit kann die Firma Rosendahl Nextrom komplette Fertigungslinien, aber auch einzelne Module wie zum Beispiel Extruder oder Verseilmaschinen an ihre Kunden liefern. Die einzelnen Module und deren Funktionalitäten sind gekapselte Elemente, die durch eigene Steuerungselemente ausgerüstet werden. Der Datenfluss zwischen den Modulen ist durch proprietäre, steuerungsbasierende Kommunikationsmechanismen realisiert, welche eine vertikale Integration an Fremdsysteme nicht ohne erheblichen Mehraufwand ermöglichen. Die einzelnen Anlagenteile werden durch programmierte Kommunikationsschnittstellen an die SPS (Speicher

programmierbare Steuerung) mittels Feldbus angebunden. Bei Änderungen aufgrund von Softwareupdates oder Erweiterungen müssen diese Schnittstellen händisch angepasst und neu getestet werden. Generell sind die Anlagen hierarchisch vernetzt.

1.3 Aufgabenstellung

Rosendahl Nextrom GmbH will ihre modularen Fertigungsanlagen, welche sie produzieren, durch Einsatz neuer Technologien vernetzen und Daten der Maschinen für übergeordnete Systeme bereitstellen (Stichwort „Vertikale Integration“). In diesem Zusammenhang steht das aktuelle Schlagwort Industrie 4.0, welches die Firma Rosendahl Nextrom GmbH in Kooperation mit der Firma Siemens AG Österreich erforscht. Die Themen der Digitalisierung sollen analysiert und ihre Machbarkeit und ihren Zweck für die eigenen Produkte der Firma Rosendahl Nextrom GmbH ermittelt werden. Durch die einheitliche Vernetzung und Einarbeitung eines Standards soll die Anpassung und Erweiterbarkeit ihrer Anlagen in Zukunft effizienter und kostensparender durchführbar sein. Aufgrund der durchgängigen Datenintegrität und Anbindung sollte auch die Fehlererkennung und Fehlerreduzierung angestrebt werden.

1.4 Zielsetzung

Das Ziel ist es, durch Implementierung von neuen Technologien Produktionsprozesse zu flexibilisieren, Transparenz in den laufenden Prozessen zu implementieren und wichtige Datenmengen übergeordneten Systemen zur Verfügung zu stellen. Damit sollen folgende Themengebiete möglich gemacht werden:

- Selbstoptimierung
- Selbstkonfiguration
- Nachverfolgbarkeit
- Selbstdiagnose der Anlagen
- Einfache Individualisierung der Prozesse auf Kundenwünsche
- Daten der Maschine bereitstellen (systemunabhängig, vertikale Integration) um Analyseprogramme zu entwickeln für z.B.: vorbeugende Wartung, Absatzanalyse
- Webfähigkeit

Der praktische Bezug dieser Arbeit soll durch die Umsetzung des erarbeiteten theoretischen Wissens zum Thema Digitalisierung anhand einer Maschine der Firma Rosendahl Nextrom GmbH erreicht werden. Ein essentieller Anlagenteil der Produktionsanlage, der Extruder, soll durch standardisierte Schnittstellen an ein übergeordnetes System angebunden werden. Diese Schnittstelle soll leicht erweitert werden können und auch auf andere Anlagenteilen implementiert werden. Ziel ist ein standardisiertes Anlagennetz und eine Datenkopplung mittels Plug&Play Mechanismen, welches in Folge für die Umsetzung von weiteren möglichen Szenarien genutzt werden kann.

1.5 Aufbau der Arbeit

Die vorliegende Arbeit setzt sich insgesamt aus drei Teilen zusammen. Der erste Teil widmet sich den theoretischen Ausführungen, die wiederum in mehrere Abschnitte unterteilt wurden. Zum einen gibt der erste Abschnitt des theoretischen Umfangs Einblick in die geschichtliche Entwicklung der Automatisierungs- und Produktionstechnik und nimmt Bezug auf die derzeitigen Entwicklungen im Rahmen von Industrie 4.0. Zum anderen wird auf den Grundgedanken und die Referenzarchitektur von Industrie 4.0 eingegangen. Dabei werden speziell die Cyber-Physikalischen-Systeme und die Verwaltungsschale analysiert, welche die theoretische Grundlage für die weiteren Erkenntnisse und Spezialisierungen der Arbeit geben sollen. Im darauffolgenden Abschnitt werden Technologien aufgearbeitet, welche die Umsetzbarkeit der theoretisch beschriebenen Kernaspekte von Industrie 4.0 gewährleisten sollen. Der praktische Teil der Arbeit befasst sich anfangs mit der Beschreibung des expliziten Anlagenteils, welcher in dem Umsetzungsprojekt bearbeitet wird. Anschließend werden die nötigen Schritte zur Implementierung der vorgestellten und ausgewählten Technologien abgearbeitet und die Umsetzung beschrieben. Der letzte Teil der praktischen Umsetzung beschäftigt sich mit dem Anwendungsszenario, das aufgrund der wissenschaftlichen Arbeit und der Kooperation mit den Verantwortlichen der Firma Rosendahl Nextrom GmbH entstanden ist. Es beschreibt die zukünftige Einsetzbarkeit, die durch das umgesetzte Standardisierungskonzept möglich wird. Das Resümee am Ende der Masterarbeit soll die Ergebnisse zusammenfassen und einen Ausblick für zukünftige Projekte im Bereich Industrie 4.0 geben.

2 COMPUTER INTEGRATED MANUFACTURING (CIM)

Schon in den frühen 80iger Jahren gab es Bestrebungen, mit Computersystemen neue Methoden und Prozesse zu entwickeln, die Unternehmen in ihrem wertschöpfenden Prozess unterstützten und effizienter machen. In dieser Zeit wurde das erste Modell dieser CIM-Architektur entwickelt, das sogenannte „CIM-Wheel“ von CASA. Dieses Modell soll alle Elemente in einem Unternehmen beinhalten und ihre Kooperation untereinander veranschaulichen (Abb. 2).

Schon damals wurde erkannt, dass eine einheitliche und durchgängige Datenbasis eine Schlüsselfunktion einnehmen wird (Common Data). Auch wurde die Verknüpfung und die Abhängigkeiten in den verschiedenen Bereichen eines Unternehmens dargestellt und die Notwendigkeit des Informationsaustausches erkannt (sowohl Produktentwicklung, Produktionsplanung, Produktionssteuerung, als auch die Fabrikautomatisierung).

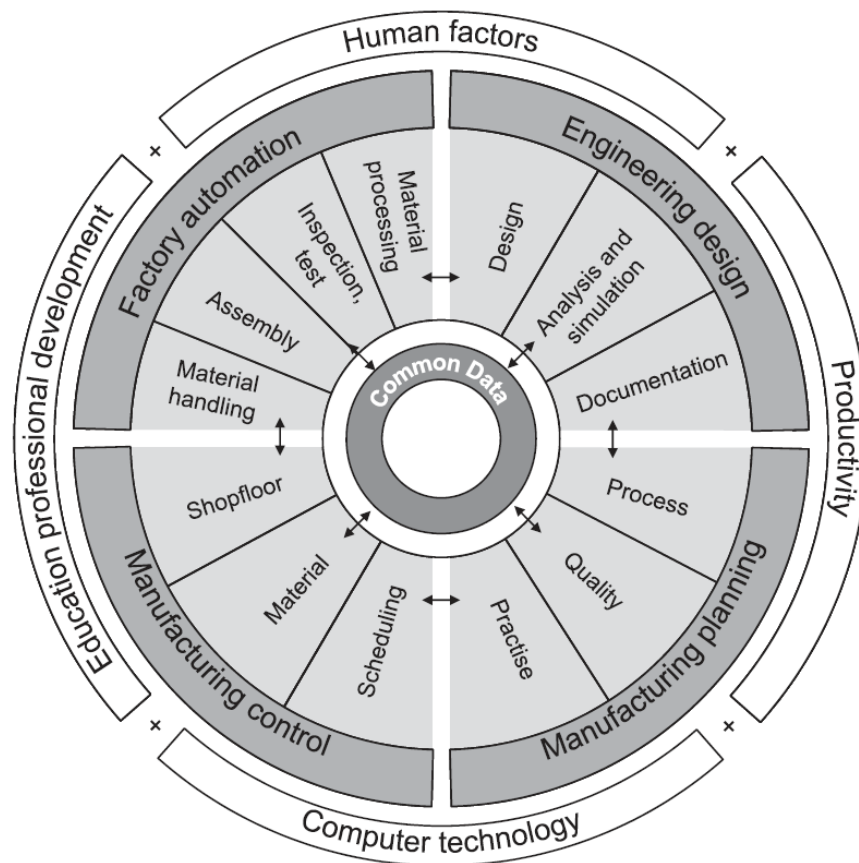


Abbildung 2: CIM-Wheel, Quelle: Hans-Friedrich Jacobi (2013).

Der äußerste Kreis zeigt, dass die Produktivität des Unternehmens von menschlicher Interaktion, der integrierten Computertechnologie, sowie der stetigen Förderung und Weiterbildung abhängig ist. Aus dieser Initiative folgten entsprechende Standards, welche zur Integration von computergestützten Systemen beitragen sollten. Einer der bekanntesten, welcher aus dieser Grundlagenarbeit entstand, ist der ISO-Standard STEP (Standard for the exchange of product model data). Zusammenfassend kann man festhalten, dass die Idee der Datenintegrität und das Zusammenführen von Informationen eines wertschöpfenden Unternehmens zur Steigerung der Effizienz schon in den frühen Zeiten der

Automatisierungstechnik entstanden ist. Es wurde schon damals erkannt, dass durch Einsatz von Rechnern und einer gemeinsamen Datenbasis ein erheblicher Mehrwert in Produktionsbereichen erzielt werden kann.²

2.1 Von CIM zu Industrie 4.0

Die Produktion wurde durch den CIM-Ansatz und folgende zusätzliche Innovationen, wie zum Beispiel der Lean Produktion stetig weiterentwickelt. Aufbauend auf diesen Erkenntnissen und den sich neu ergebenden Möglichkeiten der Digitalisierung wurde im Rahmen von Forschungsinnovationen versucht, industrielle Prozesse durch die Vernetzung von sogenannten cyber-physikalischen Systemen (CPS) autonomer zu gestalten (Abb. 3).

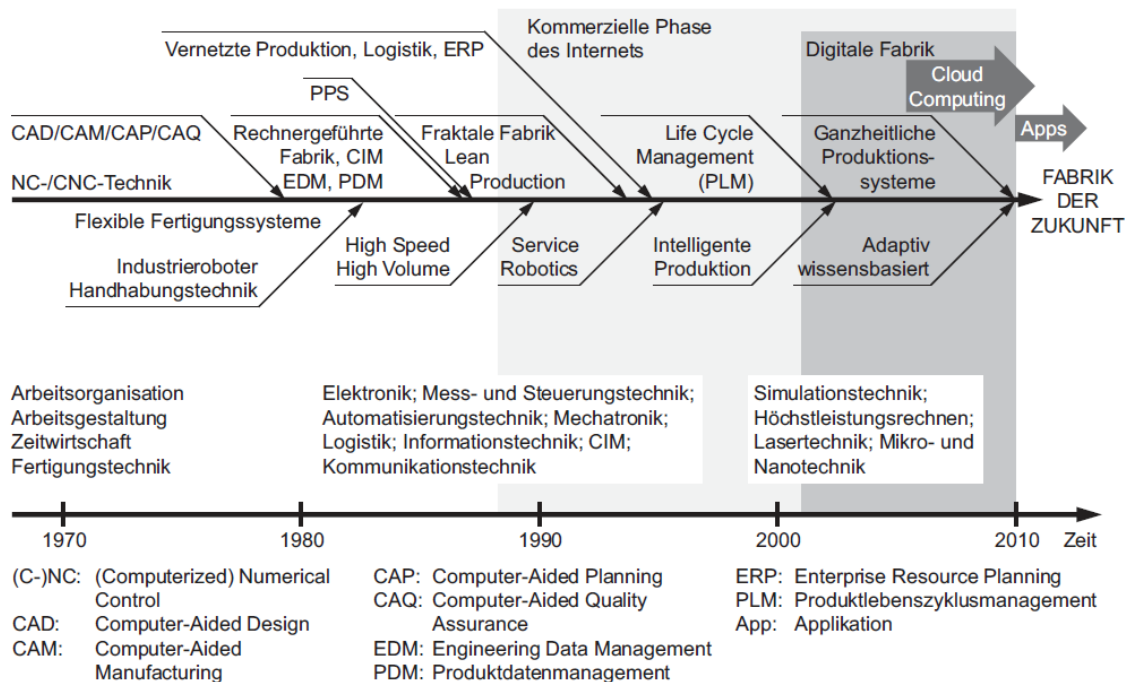


Abbildung 3: Ingenieurrelevante Entwicklungen im Kontext der Vision „Factory of the Future.“, Quelle: Hans-Friedrich Jacobi (2013).

Dabei wurden folgende Kernthemen definiert:³

- „Horizontale Integration über Wertschöpfungsnetzwerke
- Digitale Durchgängigkeit des Engineerings über die gesamte Wertschöpfungskette
- Vertikale Integration und vernetzte Produktionssysteme“

Aus diesen Kernthemen wurden acht Handlungsfelder für Industrie 4.0 identifiziert:⁴

² Vgl. Hans-Friedrich Jacobi (2013), S. 51ff.

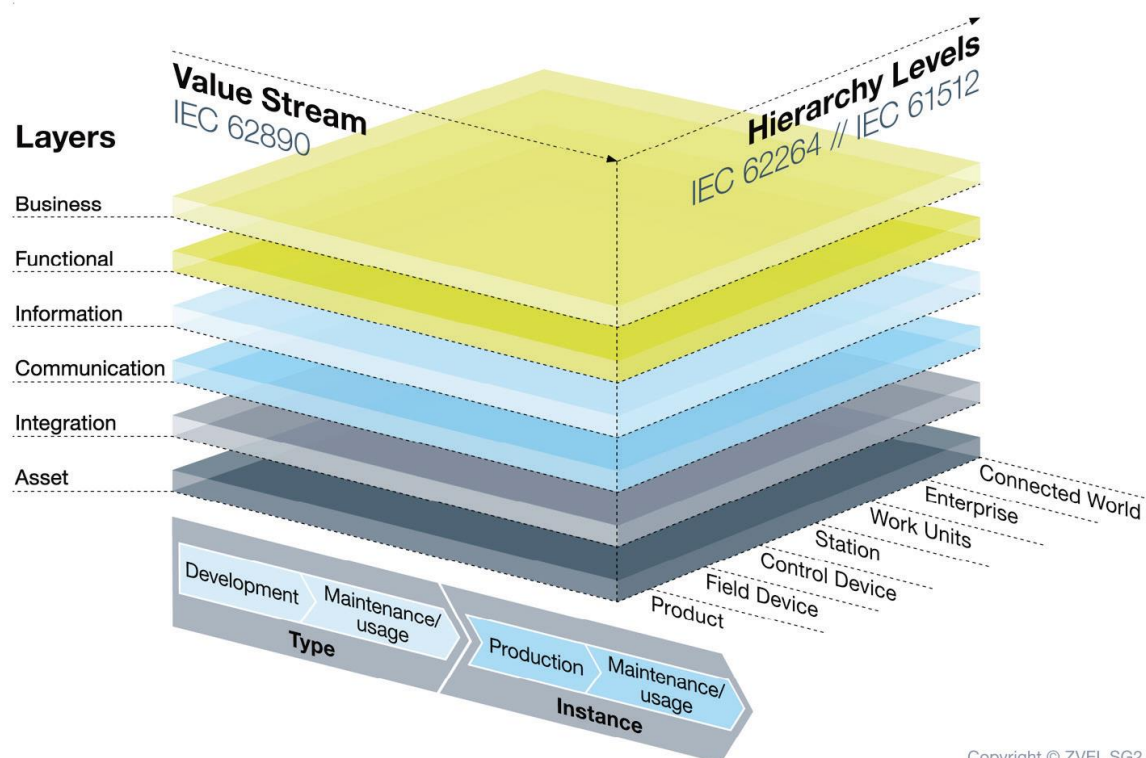
³ Hans-Friedrich Jacobi (2013), S. 88.

⁴ Hans-Friedrich Jacobi (2013), S. 89.

1. „Standardisierung und Referenzarchitektur
2. Beherrschung komplexer Systeme
3. Flächendeckende Breitbandinfrastruktur für die Industrie (ausfallsichere Kommunikationsnetze hoher Qualität, Breitband-Internet-Infrastruktur auch zu den Partnerländern etc.)
4. Sicherheit (beim Produzieren sowie hinsichtlich der Informationsflüsse gegenüber Datenmissbrauch etc.)
5. Arbeitsorganisation und -gestaltung (stärkere Eigenverantwortung und Selbstentfaltung etc.)
6. Aus- und Weiterbildung (lernförderliche Arbeitsorganisation etc.)
7. Rechtliche Rahmenbedingungen (Schutz von Unternehmensdaten, Haftungsfragen, Handelsbeschränkungen, Umgang mit personenbezogenen Daten etc.)
8. Ressourceneffizienz (Umwelt- und Versorgungsrisiken etc.)“

2.2 Referenzarchitekturmodell Industrie 4.0 (RAMI 4.0)

Durch die verschiedenen Industriezweige und -branchen und deren eigene Interessen und Standards, welche diese Industrien hervorbrachten und einsetzten, musste ein Modell entwickelt werden, welches eine einheitliche Sicht und Denkweise für Diskussionen in Bezug auf Industrie 4.0 erlaubt. In der nachfolgenden Abbildung wird das Modell grafisch beschrieben. Es besteht im Prinzip aus drei Dimensionen, welche die Komplexität des Themas anschaulich beschreiben und auf vorhandene Normen in den Teilbereichen referenzieren.⁵



Copyright © ZVEI, SG2

Abbildung 4: Referenzarchitekturmodell Industrie 4.0, Quelle: VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016].

⁵ Vgl. VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016] S. 7ff.

2.2.1 Layers - Schichten des Models

Eine durchgängige und lückenlose Beschreibung von Entwicklungsprozessen, Produktionslinien, Fertigungsmaschinen, Feldgeräten sowie dem Produkt selbst ist von essentieller Bedeutung für eine flexible Wertschöpfung. Mittels der Schichten des Referenzarchitekturmodells kann für jeden Lebenszyklus eines Produktes oder einer Produktionsanlage eine einfache Beschreibung erfolgen. Dabei ist darauf zu achten, dass nicht nur aktive kommunikationsfähige Komponenten betrachtet werden. Auch passive Komponenten müssen durch virtuelle Abbildungen Informationen bereitstellen können (z.B. die mechanische Konstruktion).⁶

- **Asset - Gegenstandsschicht**

Sie repräsentiert alle physisch vorhandenen Elemente in der Wertschöpfungskette (Roboter, Arbeitspläne, Menschen, etc.)

- **Integration - Integrationsschicht**

Ist die Schnittstelle zwischen den Assets und der virtuellen Welt. In der Integrationsschicht werden Ereignisse aus den Assets erzeugt. Zum Beispiel die Interaktionen eines Menschen mit einer Maschine mittels Human-Maschine-Interface. Festzuhalten ist, dass jedes wichtige Ereignis in der physischen Welt (Asset) ein Ereignis in der Virtualität erzeugen muss.

- **Communication - Kommunikationsschicht**

Ziel muss es sein die Informationen (Ereignisse), welche in der Integrationsschicht generiert werden, durch eine vereinheitlichte Kommunikation und Datenformat der überliegenden Informationsschicht bereitzustellen, sowie die Aussteuerung der Integrationsschicht zu gewährleisten

- **Information - Informationsschicht**

In dieser Ebene werden Ereignisse die ermittelt wurden durch Regeln miteinander verknüpft und dadurch höherwertige Daten erzeugt. Man kann hier von einer Ereignisvorverarbeitung sprechen. Es werden formale Regeln beschrieben wie Ereignisse und Daten weiterverarbeitet werden. Konsistente Daten und die Integration dieser Daten, sowie die Datenintegrität sind wichtige Themen die in dieser Ebene sichergestellt werden müssen. Aus diesem Kontext heraus kann man bereits erkennen, dass nicht nur der Gewinn vieler Daten notwendig ist (Big Data), sondern dass diese Daten selektiert und verknüpft werden müssen, um Wissen zu generieren (Smart Data).

- **Functional - Funktionsschicht**

Innerhalb der Funktionsschicht werden in Bezug auf die vorgearbeiteten Ereignisse und Daten aus der Informationsschicht Entscheidungslogiken erzeugt. Somit werden Anwendungen und fachliche Funktionalitäten in einer Laufzeitumgebung umgesetzt. Daraus ergibt sich zwangsläufig die Notwendigkeit, die Interaktion in der horizontalen Integration auf dieser Schicht auszuführen.

- **Business - Geschäftsschicht**

Zeigt den Gesamtprozess in Bezug auf das Geschäftsmodell des Unternehmens, sowie die Funktionen in Zusammenhang mit der Wertschöpfungskette. Daten und Ereignisse die für den

⁶ Vgl. VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016] S. 8ff.

Geschäftsprozess wichtig sind werden in dieser Ebene behandelt und Regeln dazu aufgestellt. Auch rechtliche und regulative Aspekte müssen hier berücksichtigt werden.⁷

2.2.2 Hierarchy Levels - Hierarchische Ebenen

Die Hierarchie-Ebenen sind für die funktionale Zuordnung einer Sachlage im Unternehmen vorgesehen. Aufgrund der Tiefe von Industrie 4.0 ist nicht nur die Steuerungsebene von Interesse. Es beginnt bereits beim Produkt selbst und zieht sich durchgehend von der einfachen Feldebene (intelligente Sensoren/Aktoren) bis in die Unternehmensplanung eines Produktionsstandorts und darüber hinaus (Connected World) in die Vernetzung von Standorten sowie Lieferanten und Kunden. Diese dritte Achse ist somit für die funktionale Einordnung von Abläufen im Unternehmen zuständig. Bei der Entwicklung dieser Kategorisierung wurden die Normen IEC 62264 und IEC 61512 herangezogen und sinnvoll erweitert.⁸

2.2.3 Value Stream - Wertschöpfung und Lebenszyklus

Die dritte Dimension der Darstellung beschreibt den Lebenszyklus des Typs bzw. einer Instanz eines Produktes, Auftrags, einer Fabrik oder einer Maschine. Dieser zeigt die Phasen von der Entwicklung über den Prototypen, der Vorserie, bis zum Serienprodukt und deren Wartung bis zum Ausscheiden aus dem Sortiment.

- **Type**

Ist ein Gegenstand welcher aus der Entwicklung als individueller Typ (Informationen über Bestandteile, Material usw.) bekannt ist.

- **Instanz**

Eine Instanz als Gegenstand wird als mehrfach vorhandener Typ definiert, welcher die immer gleichen Eigenschaften einer Type besitzt. Diese werden mit Seriennummern versehen um eine eindeutige Zuordnung zu gewährleisten.

Der Lebenszyklus einer Maschine oder Anlage ist ein sehr komplexes Zusammenspiel der verbauten Komponenten und deren Typen und Instanzen. Sehr wichtig für den Planungsprozess ist das Zusammenwirken von Geschäftspartnern und den Lebenszyklen der Zulieferteile, da diese für die eigene Planung und nachfolgend die Planung des nächsten Geschäftspartners von Vorteil ist (Abb. 5).⁹ Um dieses Prinzip des Datenflusses zwischen Geschäftspartnern zu realisieren, bedarf es sowohl rechtlichen Rahmenbedingungen als auch technologischen Erweiterungen, da es sich möglicherweise um personenspezifische Daten handeln kann und diese Daten auch gespeichert werden müssen. Bei Maschinenteilen oder ganzen Anlagen muss eine Zuordnung, wie bereits vorhin beschrieben, erfolgen. Dies muss mittels geeigneter Technologien (z.B. Uniform Resource Identifier URI) erfolgen, um den

⁷ Vgl. VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016] S. 9ff.

⁸ Vgl. VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016] S. 9ff.

⁹ Vgl. VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016] S.10ff.

Lebenszyklus eines Bauteils einfacher verfolgen zu können. In weiterer Folge können die Bauteile rasch lokalisiert und eventuell nachfolgende Geschäftspartner informiert werden.

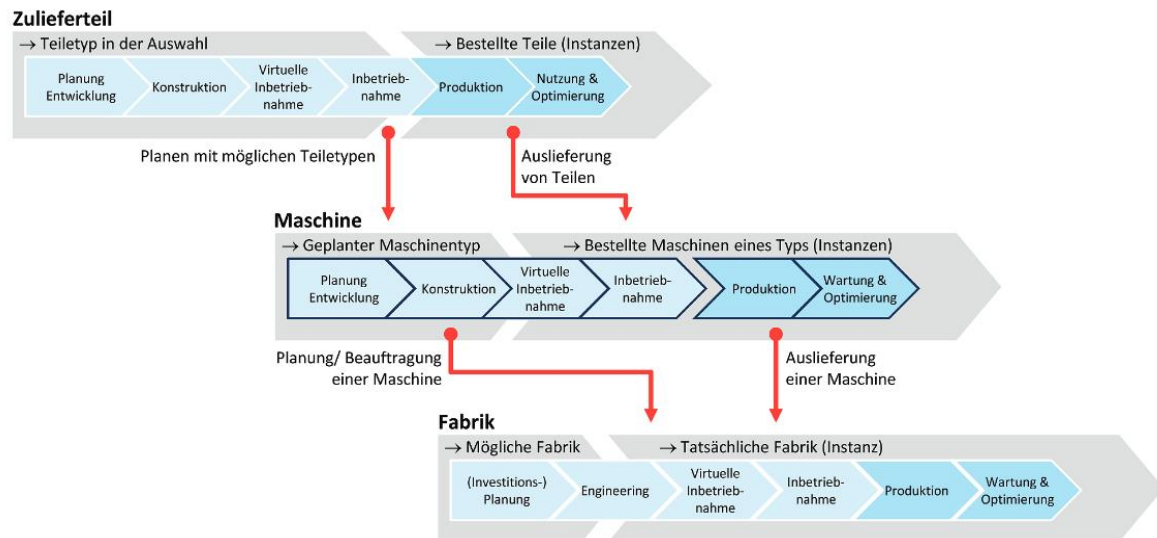


Abbildung 5: Typen und Instanzen im Lebenszyklus, Quelle: VDI/DE-Gesellschaft (2015), Online-Quelle [17.08.2016].

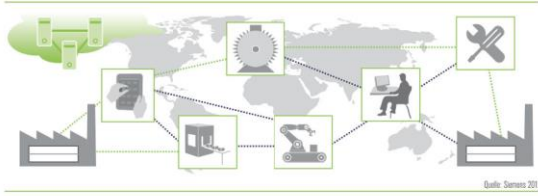
2.2.4 Zwischenfazit

Die weitere Effizienz und Effektivitätssteigerung in wertschöpfenden Prozessen ist durch Prozessoptimierungsverfahren nahezu ausgereizt (z.B. Lean Production). Einer der letzten sinnvollen Hebel ist die Datenverarbeitung und Analyse, um Flaschenhälse, welche für den Menschen nicht erfassbar sind, aufzudecken. Das Referenzarchitekturmodell soll als Anhaltspunkt dienen, um den Sprung zur Produktionsanlage von morgen vollziehen zu können. Dabei sind in erster Linie die Erfassung wichtiger Ebenen für das eigene Unternehmen von Interesse, um einen Entry Point für Industrie 4.0 Projekte identifizieren zu können. Hierbei ist die Standardisierung der Kommunikation ein Kernaspekt zur Umsetzung. Die Digitalisierung der vorhandenen Daten und vor allem die sinnvolle Zuordnung sind die Wegbereiter für Industrie 4.0. Die wichtigsten Aspekte zur Umsetzung sind:

1. Die horizontale Integration von Produktionsanlagen und Standorten zur effizienten Produktionssteuerung bzw. in weiterer Folge die selbstständige Produktionssteuerung
2. Die vertikale Integration, Verknüpfen von Anlagenteilen und Benutzerschnittstellen zur Effizienzsteigerung des Produktionsprozesses
3. Einheitliches Engineering, Informations- und Formatstandards von der Entwicklung bis zum Service. Einheitliche Datenablage und Archivierung
4. Der Mensch als Entscheider und als kreatives Element im Wertschöpfungsprozess¹⁰

¹⁰ Vgl. Bauernhansl/Hompel/Vogel-Heuser (2014), S. 78.

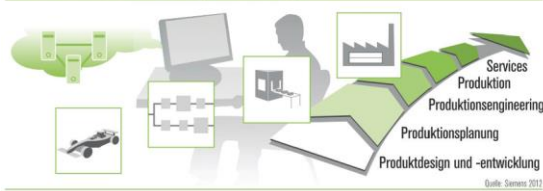
Horizontale Integration



Vertikale Integration



Durchgängiges Engineering



Mensch als Dirigent der Wertschöpfung



Abbildung 6: Die vier wichtigsten Aspekte von Industrie 4.0, Quelle: VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016].

2.3 Cyber-Physikalische-Systeme

Der ganzheitliche Ansatz von Industrie 4.0 birgt zahlreiche Veränderungen in vielen Bereichen der industriellen Fertigung. Aus Sicht der Automatisierungstechnik war bis dato eine strikte hierarchische Struktur die gängige Herangehensweise um Produktionsprozesse abzubilden. Die Automatisierungspyramide mit ihren Ebenen definiert Übergänge, in denen jedoch Informationen, oft sogar bewusst, nicht in die darüberliegenden Systeme transportiert wurden. Dieser hierarchische Ansatz muss im Kontext der vierten Industriellen Revolution überdacht werden.¹¹

Automatisierungsebene

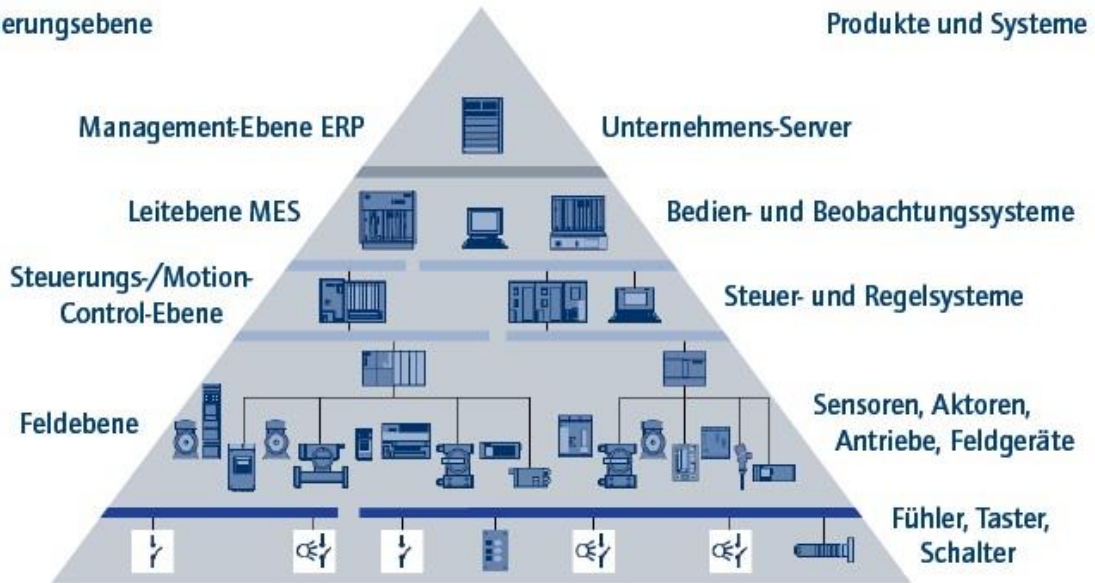


Abbildung 7: Automatisierungspyramide, Quelle: Friedl (2010).

¹¹ Vgl. Bauernhansl/Hompel/Vogel-Heuser (2014), S. 38ff.

In diesem Zusammenhang kommen die Cyber-Physikalischen-Systeme in den Fokus. Hierbei handelt es sich um Systeme, welche Dienste in einer Produktion zur Verfügung stellen. Diese Systeme gibt es natürlich auch in den derzeitigen Produktionsanlagen ebenfalls. Der Unterschied ist, dass zu dieser physikalisch vorhandenen Maschine, ein virtuelles Ebenbild existiert, die Daten der Maschine in aktuellster Form bereitstellt. Dieses virtuelle Abbild kennt die Eigenschaften der Maschine und die Dienste, welche sie ausführen kann. Außerdem werden aktuelle Prozessdaten über die virtuelle Schnittstelle zur Verfügung gestellt.

Im Grunde kann man folgendes zusammenfassen. Ein Cyber-Physisches-System ist ein System, das:¹²

- Mittels Sensoren physikalische Daten erfasst und mittels Aktoren einwirkt
- Ein auf Grundlage der gewonnenen Daten aktiv und reaktiv mit der physikalischen und der digitalen Welt interagiert
- Mit anderen CPS über digitale Netze miteinander verbunden ist, um deren Dienste miteinander zu teilen und Informationen auszutauschen
- Die Bereitstellung der Daten und Services über das Firmennetzwerk hinaus gewährleistet
- Schnittstellen zur Mensch-Maschinen Interaktion bietet

Der größte Hemmschuh, der zur Umsetzung von Industrie 4.0 überwunden werden muss, sind die sogenannten Medienbrüche in den Systemen.¹³ In den meisten Unternehmen ist die grundlegende Infrastruktur für Kommunikationen zwischen den Ebenen bereits gegeben. Hier wird die Aufgabe sein, einheitliche Datenschnittstellen zwischen den CPS zu schaffen, um eine durchgängige, barrierefreie Datenkommunikation zu gewährleisten. Für den Unternehmenserfolg wird entscheidend sein, dass durch Plug&Play Mechanismen einfache Anpassungen und Erweiterungen ihrer Produktionen ermöglicht werden. Im Referenzarchitekturmodell Industrie 4.0 wird dazu der Begriff „Verwaltungsschale“ verwendet.¹⁴ Die Verwaltungsschale soll ein elektronischer Container für Gegenstände in der Produktion sein. In ihr sollen möglichst alle relevanten Daten gespeichert werden um dieses virtuelle Abbild realisieren zu können.¹⁵

¹² Vgl. Bauernhansl/Hompel/Vogel-Heuser (2014), S. 37.

¹³ Vgl. Bauernhansl/Hompel/Vogel-Heuser (2014), S. 81ff.

¹⁴ Vgl. VDI/VE-Gesellschaft (2015), Online-Quelle [17.08.2016].

¹⁵ Vgl. Hoffmeister (2015), Online-Quelle [22.08.2016], S. 1.

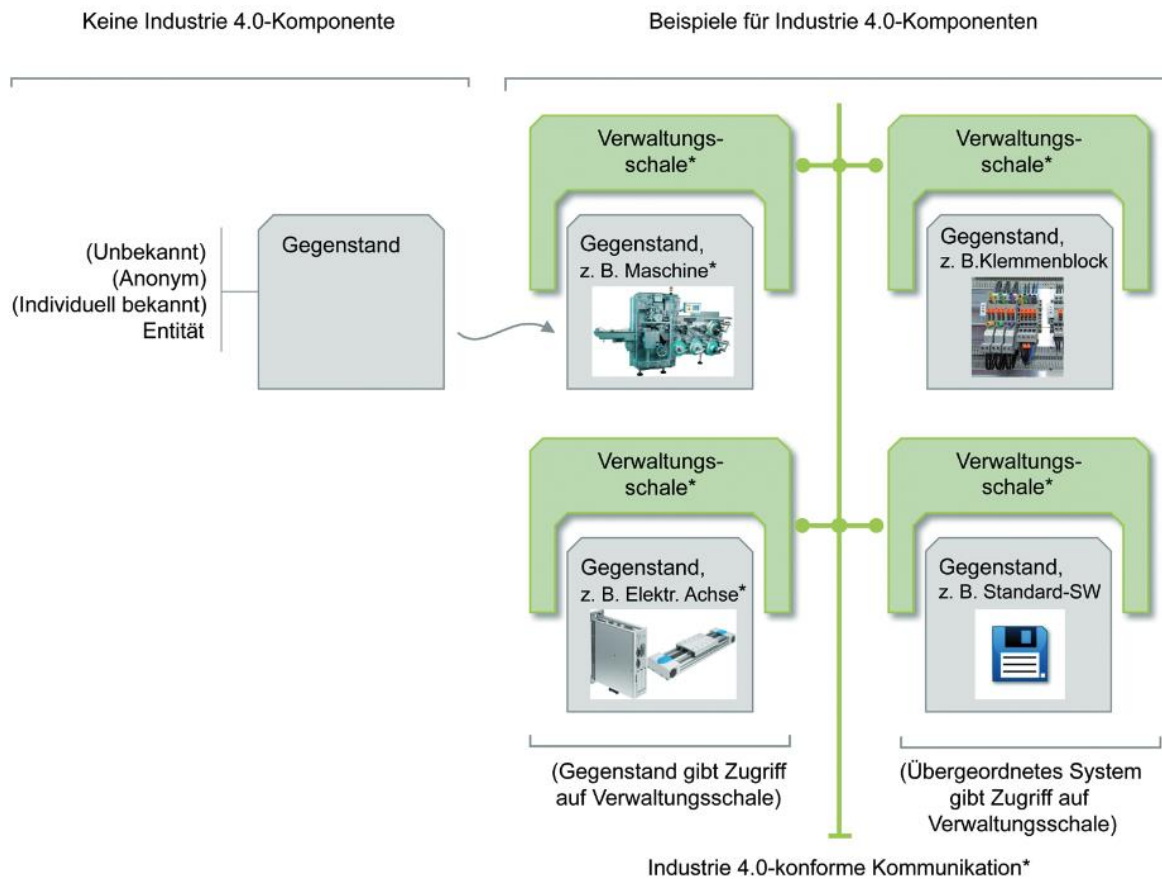


Abbildung 8: Vom Gegenstand zur Industrie 4.0-Komponente, Quelle: Hoffmeister (2015), Online-Quelle [22.08.2016].

2.3.1 Industrie 4.0-konforme Kommunikation

Der Begriff *Industrie 4.0-konforme Kommunikation* wird im Referenzarchitekturmodell sehr allgemein beschrieben. Diese Kommunikation muss per Definition, die Daten der virtuellen Repräsentation eines Gegenstandes bereitstellen oder von einem übergeordneten IT-System gehalten werden. ¹⁶

¹⁶ Vgl. VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016].

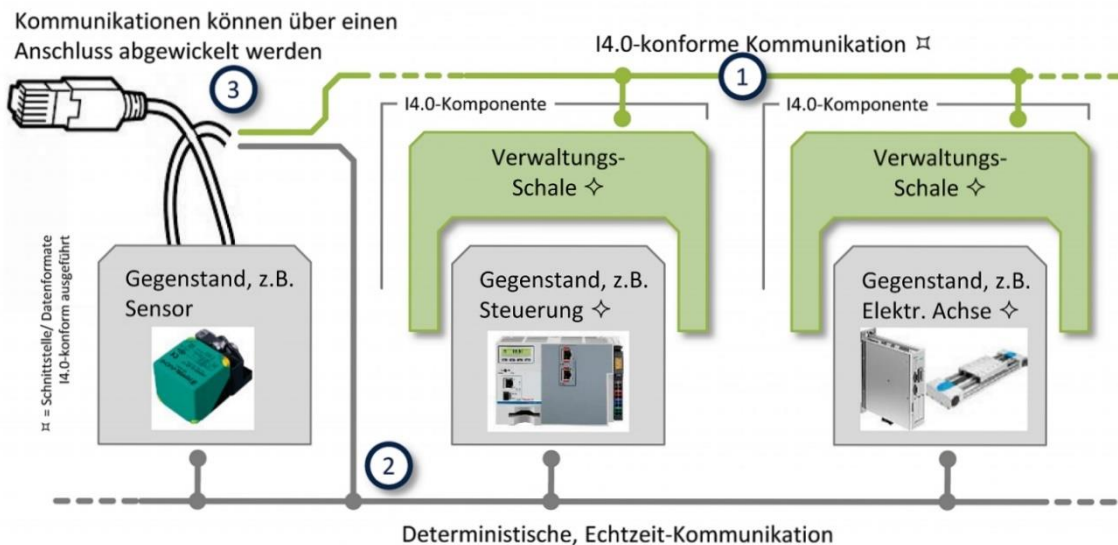


Abbildung 9: Vernetzungsmöglichkeiten, Quelle: Hoffmeister (2015), Online-Quelle [22.08.2016].

In der Abbildung 9 wird der Unterschied zwischen bereits vorhandenen deterministischen, Echtzeit-Kommunikation und der I4.0-konformen Kommunikation ersichtlich. In der I4.0-konformen Kommunikation ist es nicht das Ziel, den Gegenstand selbst anzubinden, sondern den Repräsentanten des Gegenstandes mit den notwendigen Meta-Informationen und Laufzeitdaten dem I4.0-Netzwerk zur Verfügung zu stellen (Abb.9, Nr.1). Ein Kernaspekt der Umsetzung einer I4.0-konformen Kommunikation ist, dass die bestehende Kommunikation keine Beeinflussung durch die zusätzliche Anbindung erfährt (Abb.9, Nr.2). Aufgrund der üblichen Echtzeit-Ethernet-Protokolle die auf der Feldebene der Anlagen eingesetzt werden, ist eine gemeinsame Nutzung der Infrastruktur natürlich möglich. Jedoch ist darauf zu Achten, dass beide Kommunikationsarten logisch voneinander getrennt werden.¹⁷

2.3.2 Die Verwaltungsschale - Datencontainer eines CPS

Grundsätzlich stellt sich die Frage welche Daten und vor allem wie diese Daten in der „Verwaltungsschale“ gespeichert werden sollen. Hier hat der Zentralverband der Deutschen Elektrotechnik- und Elektronikindustrie e. V. und die Plattform Industrie 4.0 einen Ergebnisbericht vorgelegt, welcher grundsätzliche Informationen dazu liefern kann. Es wurde bei der Entwicklung der Struktur und der Inhalte Bezug auf die Norm IEC 62832 „Digitale Fabrik“ genommen. Diese gibt bereits eine Art Regelwerk zur Definition von physischen bzw. logischen Objekten in einer digitalen Fabrik vor.¹⁸

2.3.2.1 Aufbau der Verwaltungsschale

Die Verwaltungsschale ist in Anlehnung an die *Digitale Fabrik* (DF) nach IEC 62832 in einen „Header“ und „Body“ aufgeteilt. Die *Digitale Fabrik* entspricht einem ähnlichen Konzept, das versucht Systeme in einer Fabrik als eine Anordnung von DF Komponenten darzustellen. Der Header trägt jegliche Information zur Identifikation des physikalischen Objekts und dessen Bezeichnung sowie die Identifikation der

¹⁷ Vgl. VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016].

¹⁸ Vgl. Plattform Industrie 4.0 (2016), Online-Quelle [17.08.2016], S. 13ff.

Verwaltungsschale selbst. Im Body finden sich die einzelnen Komponenten des Objekts mit ihren Merkmalen und Eigenschaften. Diese Merkmale referenzieren auf Daten und Funktionen während der Laufzeit. Alle Teilkomponenten bilden somit ein „Manifest“ welches das Objekt ganzheitlich beschreibt. Außerdem muss es möglich sein, *Sichten* zu definieren, welche wichtigen Daten und Eigenschaften filtern und somit überflüssige Daten ausblenden. Mit Sichten sind hier zum Beispiel die verschiedenen Lebenszyklen oder auch Aufgaben (Service, Inbetriebnahme, Produktivphase) gemeint. In der nachfolgenden Darstellung soll dieses Konzept nochmals anschaulich dargestellt werden.¹⁹

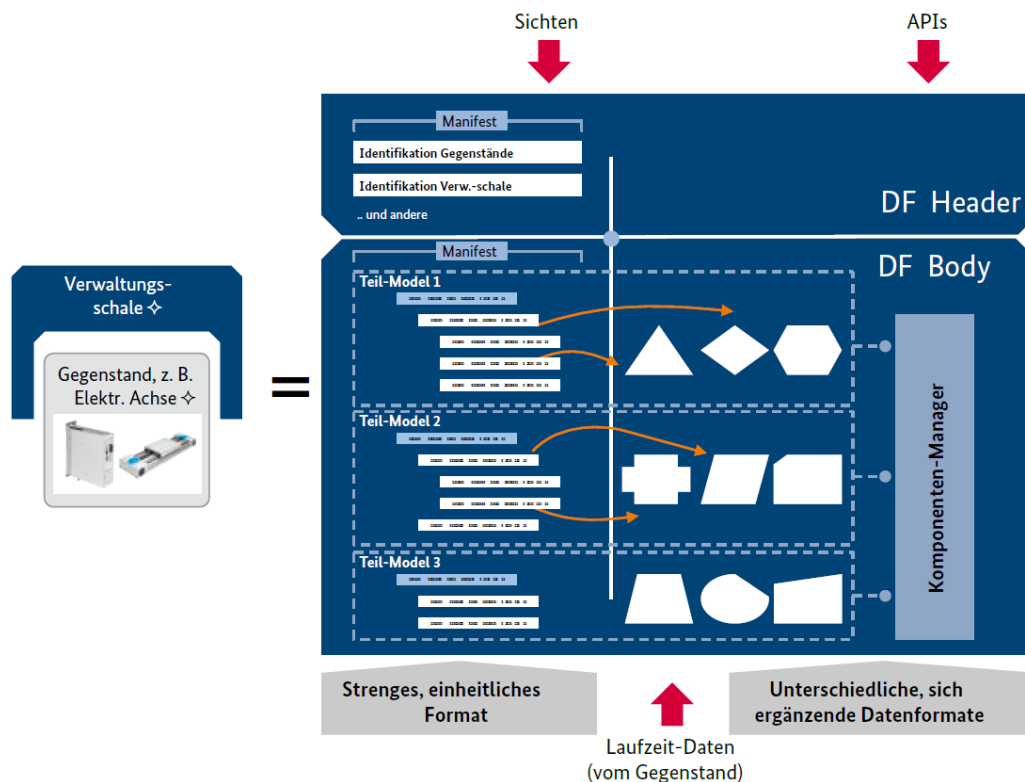


Abbildung 10: Grobstruktur der Verwaltungsschale, Quelle: Plattform Industrie 4.0 (2016), Online-Quelle [17.08.2016].

2.3.2.2 Identifikationsmöglichkeiten eines CPS in der Verwaltungsschale

Wie bereits oben erwähnt, ist die Identifikation eines Objekts durch die Verwaltungsschale maßgeblich. Um dies zu gewährleisten werden zwei Varianten der Identifikation vorgeschlagen:

1. Identifikation nach ISO 29002-5

Die Norm definiert einen Standard, wie eine eindeutige ID festgelegt werden kann. Genauere Details dazu werden in dieser Arbeit nicht behandelt.

2. Identifikation mittels Uniform Resource Identifier (URI)

Durch die auf den weltweiten IT Standard (W3C) beruhende URI wird mittels Domäne und Pfad eine eindeutige Identifikation ermöglicht. Diese Art der ID (Identifikation) wird zum Beispiel für Apps im Google Playstore, aber auch im Kommunikationsstandard OPC UA verwendet.²⁰

¹⁹ Vgl. Plattform Industrie 4.0 (2016), Online-Quelle [17.08.2016], S. 28ff.

²⁰ Vgl. Plattform Industrie 4.0 (2016), Online-Quelle [17.08.2016], S. 31ff.

2.3.3 Internet der Dinge

Wenn Cyber-Physische-Systeme über digitale Netze im Intranet und Internet zur Verfügung stehen, kommt man zwangsläufig zum Begriff *Internet der Dinge* (IoT). Der Begriff Internet der Dinge wurde 1991 von Mark Weiser das erste Mal aufgegriffen und beschreibt ein Netz, welches sich nicht nur auf virtuelle Objekte bezieht, sondern auch physische Objekte beinhaltet. Die CPS sind vergleichbar mit der damaligen Definition physischer Objekte.²¹ Dinge die existieren auch im Internet zu repräsentieren ist eines der Kernziele von Industrie 4.0. Idealerweise soll ein Netzwerk entstehen, in dem die CPS untereinander kommunizieren und deren Information und „Dienstleistungen“ anbieten können. In diesem Zusammenhang wird auch gerne der Begriff „Internet der Dienste“ verwendet. Um dieses globale Netzwerk an Dingen zu realisieren, ist das Verringern der sogenannten Medienbrüche essentiell. Um diesen Gedanken vollständig umsetzen zu können, müssen Daten uneingeschränkt zur Verfügung stehen und müssen sich frei bewegen können. Daten sollten nicht am Weg von der Quelle zu den verschiedensten Zielen manipuliert und geändert werden.²²

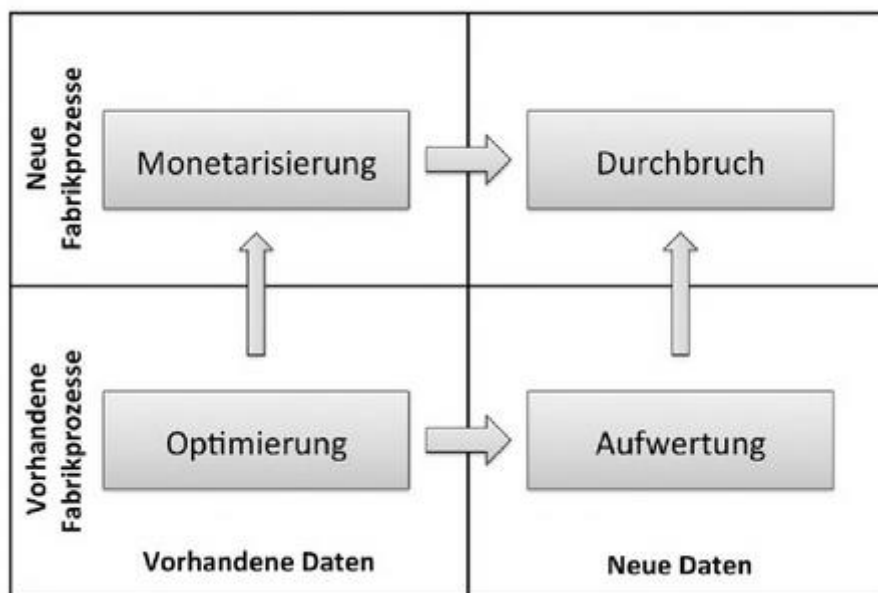


Abbildung 11: Stufen der Optimierung durch Verringerung von Medienbrüchen, Quelle: Bauernhansl/Hompel/Vogel-Heuser (2014).

Im Grunde geht es darum, dass Objekte mittels Netzwerkzugriff adressierbar werden. Dadurch können Sie eindeutig identifiziert werden, Informationen über sich und Ihre Umgebung sammeln und bereitstellen, bzw. Reaktionen auf Ereignisse von außen selbstständig durchführen.

Zwei Blickwinkel werden dazu bei Industrie 4.0 beleuchtet.

1. Produkte können netzwerkfähig sein oder mittels Speichertags (RFID) Informationen über sich bereitstellen. Diese Information soll dafür genutzt werden, dass das Produkt selbst den Produktionsprozess steuern kann. Außerdem ist es möglich, durch einen digitalen Fingerprint

²¹ Vgl. Weiser (1991), Online-Quelle [26.08.2016].

²² Vgl. Bauernhansl/Hompel/Vogel-Heuser (2014) S.81ff.

jegliche Information im Produktionsprozess direkt zu speichern. Jeder Schritt im Lebenszyklus eines Produktes kann somit aufgezeichnet und abgerufen werden.

- Produktionsanlagen bzw. einzelne Fertigungsmaschinen können durch standardisierte Netzwerkanbindung an das Internet der Dinge angebunden werden. Die Datenquellen können dann genutzt werden um Produktionsleistung, Produktionsqualität und Betriebszustände zu steigern und nachzuverfolgen.²³

2.3.4 Cyber-Physikalische Produktions-Systeme

Durch die stetige Vernetzung von Sensorik/Aktorik und Steuerungen in Industrieanlagen welche im Internet der Dinge als cyber-physische Systeme repräsentiert werden, entsteht ein zusammenhängendes Produktionsnetz, welches cyber-physisches Produktions-System genannt wird. Diese CPPS sollen die komplette Verfahrenskette einer Produktion sowie deren Lebenszyklus abdecken.²⁴ Bei der Durchführung von Industrie-4.0-Projekten kann ein CPPS herangezogen werden um eine schlanke und leichte Integration in bereits bestehende Anlagen umzusetzen. In diesem Zusammenhang wird auch von einem agentenbasierten Ansatz gesprochen. Um diesen einfachen Ansatz zu verfolgen werden Anlagenteile bzw. auch ganze Anlagen gekapselt und als CPPS repräsentiert. Wichtig ist, dass der Vernetzungsgrad in diesem CPPS sehr hoch ist. Das CPPS selbst wird dann durch einen Anlagenagenten im Internet der Dinge repräsentiert.²⁵

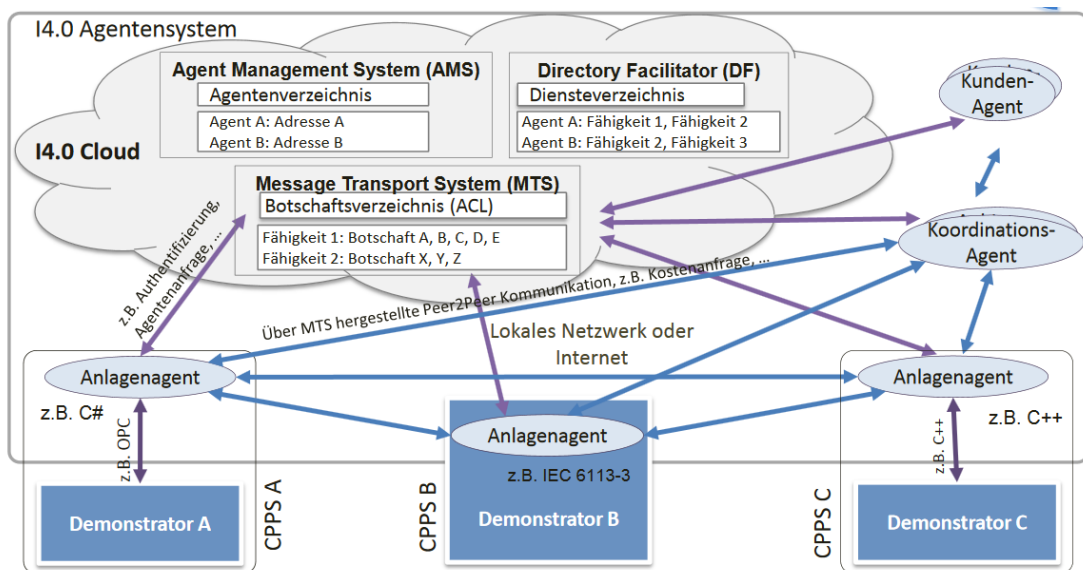


Abbildung 12: Kommunikation innerhalb CPPS mittels Agenten, Quelle: Bauernhans/Hompel/Vogel-Heuser (2014).

²³ Vgl. Bauernhans/Hompel/Vogel-Heuser (2014), S. 60ff.

²⁴ Vgl. Kagermann/Wahlster/Helbig (2013), Online-Quelle [10.07.2016].

²⁵ Vgl. Bauernhans/Hompel/Vogel-Heuser (2014), S. 146.

3 DIGITALISIERUNG DER PRODUKTIONSANLAGEN - STANDARDS ZUR UMSETZUNG INDUSTRIE 4.0

Die Grundlage zur Erreichung der Vision Industrie 4.0 ist die Digitalisierung der Produktion. Das Zukunftsbild, das durch die Forschungsarbeit im Umfeld Industrie 4.0 gezeichnet wurde, muss nun Schritt für Schritt durch Implementierung von geeigneten Technologien umgesetzt werden. Hier ist es notwendig unternehmensspezifisch Ansatzpunkte zu definieren, an denen es sinnvoll ist, eine Umsetzung durchzuführen. Der Weg zur *Intelligenten Produktion* mittels CPPS kann nur in vielen kleinen Schritten erfolgen. Ein möglicher Ansatz zur Umsetzung ist es, bereits bestehende Anlageninfrastrukturen zu analysieren und Erweiterungen in Richtung standardisierter Schnittstellen (Kommunikation) zu planen. Hier muss in erster Linie differenziert werden, wie granular eine CPS definiert werden kann, um in weiterer Folge einen Vorteil für den eigenen Wertschöpfungsprozess zu erreichen. Diese Fragen können nur individuell und unternehmensbezogen beantwortet werden. Basis dieser Umsetzung sind vorhandene Technologien, die interoperabel und systemunabhängig eingesetzt werden können.

Im Zuge der Forschungstätigkeit an Industrie 4.0 und ihrer Umsetzungsempfehlung wurden auch vorhandene Standards analysiert. Um Digitalisierung voranzutreiben ist es nicht unbedingt notwendig im ersten Schritt ganz neue Kommunikationsmechanismen und Netzwerkschnittstellen zu erfinden. Es wurde durchaus erkannt, dass mit vorhandenen Protokollen und einer Infrastruktur die Rahmenbedingungen geschaffen werden können. In diesem Teil der Arbeit werden Technologien beschrieben welche sich hauptsächlich mit der horizontalen und vertikalen Kommunikation befassen. Aus dieser Recherche haben sich einige Technologien herauskristallisiert, welche in den nachfolgenden Kapiteln erläutert werden.

3.1 OPC UA - Kommunikationsstack

Es gibt derzeit viele Standards welche in industriellen Produktionsanlagen eingesetzt werden. Diese wurden in den letzten Jahren sehr stark durch den Ethernet Standard geprägt. Dieser definiert einheitliche Übertragungsmedien wie zum Beispiel Twisted Pair oder Wireless LAN (WLAN) sowie Übertragungsprotokolle wie zum Beispiel Transmission Control Protocol (TCP) oder Hypertext Transfer Protocol (HTTP). Die vorherig beschriebene Situation macht es notwendig Kommunikationsstandards bis in den Applikation Layer variabel zu implementieren.

Folgende Eigenschaften werden zum Beispiel notwendig sein:²⁶

- Eine semantische Beschreibung von realen Objekten, welche Ihren Zweck und Nutzen maschinenlesbar beschreibt, um automatisierte Dienste überhaupt möglich zu machen
- Eine Beschreibung ihrer Funktionen, welche unabhängig vom Gerät oder Übertragungsstandard ist

OPC Unified Architecture wurde im Zuge seiner Neuauflage in Richtung Interoperabilität und Systemunabhängigkeit entwickelt. OPC UA beschreibt einen Standard, der die Protokolle TCP (Transmission Control Protocol) und SOAP (Simple Object Access Protocol) in der Transportschicht des

²⁶ Vgl. Bauernhansl/Hompel/Vogel-Heuser (2014), S. 326ff.

ISO/OSI Schichtenmodells nutzt und somit in bereits vorhandenen Ethernet-Infrastrukturen eingebunden werden kann. Es wurden existierende Standards verwendet (Sicherheit, Transport, Plug&Play) und in einem offenen, auch für eigene Zwecke erweiterbaren Standard zusammengesetzt. Die Industrie hat diesen OPC-UA-Standard bereits als mögliches Kommunikationstack zur Koppelung von verschiedenen Systemen erkannt und bereits als IEC Norm 62541 verabschiedet.²⁷

3.1.1 Aufbau des Adressraums

OPC UA kann nicht nur als Kommunikationsprotokoll bezeichnet werden. Es definiert einen Adressraum welcher es ermöglicht nicht nur reine Rohdaten abzulegen. Der Adressraum definiert durch einen objektorientierten Ansatz auch Verknüpfungen (References) zwischen Datenpaketen (Nodes) und Eigenschaften. Somit können Metadaten abgespeichert und Eigenschaften abgelegt werden. Durch den einheitlichen Zugriff auf diese Daten, welche durch ein Informationsmodell, das im OPC UA Netz jedem Teilnehmer (Server, Client) bekannt ist, werden Daten in einem virtuellen Client/Server-Verbund objektorientiert verwaltet.

OPC UA geht hier jedoch noch einen Schritt weiter. Es wurde auch eine Serviceschnittstelle definiert, wodurch nicht nur Daten bzw. Objekte abgerufen werden können, sondern auch Dienstprogramme auf Servern oder dezentral auf Clients hinterlegt werden. Durch diese Funktion können Dienste und Services (zum Beispiel kleine Apps, welche einfache Datenmanipulation durchführen) im internen Netzwerk oder über das Wide Area Network (WAN) zur Verfügung gestellt werden.

Um den Zugriff durch Externe zu verhindern, welche keine Berechtigung auf OPC UA Objekte haben, wurde im Zuge der UA Entwicklung auch sehr großer Wert auf *Security* gelegt. Es wurde ein eigener Teil in der OPC UA Spezifikation nur zum Thema *Security* implementiert. Hier hat man bereits bewährte Mechanismen aus der IT-Welt herangezogen.²⁸ Diese werden nun näher beschrieben.²⁹

- **Application Authentication**

Alle Programme welche auf Clients oder Servern implementiert sind müssen eine eindeutige Zertifizierung aufweisen -> Application Instance Certificate

Mit dem Uniform Resource Identifier (URI) werden Instanzen sowie deren Anbieter und Produkte identifiziert

- **User Authentication**

Die Benutzer-Authentisierung kann durch drei Varianten erfolgen die in den nächsten Unterkapiteln näher erläutert werden:

- Username / Password
- WS-Security Token

²⁷ Vgl. (Mar) Seite 3ff.

²⁸ Vgl. Lange/Iwanitz/Burke (2013), S. 240ff.

²⁹ Vgl. International Electrotechnical Commission (Hrsg.) (2010).

- X.509 Certificate

- **User Authorization**

OPC UA kann durch Benutzerverwaltung den Adressraum reglementieren und so Zugriffe steuern (read, write, browse oder execute)

Um die sicheren Zugriffe zu gewährleisten wurden Services implementiert (Part 4 – Services), welche den Zugriff in Richtung Vertraulichkeit, Integrität und Authentisierung gewährleisten.

Generell basiert OPC UA Security auf einer Zertifikat-Verwaltung mit der Basis *x.509 Certificates*. Dabei überlässt OPC UA dem Projektanten wie stark der Zugriff reglementiert werden soll. Grundsätzlich können Clients nur Sitzungen mit vertrauenswürdigen Servern erstellen. Umgekehrt akzeptieren Server nur Sitzungen von vertrauenswürdigen Clients. Diese grundlegende Sicherheitsanforderung ist wesentlicher Bestandteil des Protokolls, auf welcher Basis zusätzliche Sicherheitskriterien ausgewählt und aktiviert werden können. Nachfolgend werden kurz die Begriffe x.509-Zertifikat und die zugrundeliegende Infrastruktur einer vertrauenswürdigen Datenübertragung erklärt.

3.1.2 Zertifikate

OPC UA-Anwendungen, welche auf Server oder Client basieren, benötigen ein einzigartiges x.509 Zertifikat welches aus einem öffentlichen und privaten Schlüssel zusammengesetzt wurde (asymmetrische Kryptographie). Die Zertifikate müssen von einer vertrauenswürdigen Stelle ausgestellt und verwaltet werden. Diese Zertifizierungsstelle wird als Certificate Authorities (CA) bezeichnet. Die CA können von lokalen selbstsignierenden CA bis hin zu zentral verwalteten CA aufgebaut werden.

3.1.3 X.509-Zertifikat

Ein X.509-Zertifikat ist ein digitales Zertifikat, welches den „public key infrastructure“ (PKI) Standard verwendet, um öffentliche Schlüssel und den dazugehörigen Benutzer, Computer oder Services zu identifizieren. Das Zertifikat wird zur Überprüfung der Identität herangezogen.

Ein X.509-Zertifikat enthält Informationen über die Identität, an die ein Zertifikat ausgestellt wird und die Identität, die sie ausgestellt hat. Die Standardinformationen in einem X.509-Zertifikat enthalten:³⁰

- Version des x.509 Zertifikats
- Seriennummer der Entität (zum Beispiel einer Applikation)
- Informationen zum Algorithmus
- Aussteller des Zertifikates (meist die CA)
- Gültigkeit des Zertifikates (von... bis...)
- Zertifikatinhaber
- Schlüsselinformationen des Inhabers
 - Public Key Algorithmus
 - Public Key

³⁰ Vgl. Network Working Group (Hrsg.) (2008).

3.1.4 Secure Channel

Zum Aufbau einer Verbindung zwischen OPC-UA-Teilnehmern (Server oder Client) wird ein sicherer Kanal zwischen den beiden Kommunikationspartnern eröffnet. Dieser Ablauf wurde der SSL/TSL (Secure Sockets Layer/Transport Layer Security) Spezifikation nachempfunden, wie sie zum Beispiel bei einer HTTPS Verbindung zum Einsatz kommt. Die Verbindung selbst kann signiert, signiert und verschlüsselt sein oder keine Sicherungsverfahren verwenden. Der Server muss seine jeweiligen Sicherheitserfordernisse (signed, encrypted, none) dem Client mitteilen und der Client muss diese Sicherheitserfordernisse beherrschen um einen Secure Channel zu eröffnen. Bei aktivierten Sicherungsverfahren kommen asymmetrische und symmetrische Verschlüsselungsverfahren zum Einsatz.³¹ Diese werden in der nachfolgenden Tabelle genauer beschrieben.³²

Verschlüsselungsoptionen	Beschreibung
Signiert und verschlüsselt	Die Nachricht wird mit dem privaten Schlüssel des Versenders signiert. Der Nachrichteninhalte wird mit dem öffentlichen Schlüssel des Empfängers verschlüsselt. Ein Abhören oder Verändern der Nachricht ist somit nicht mehr möglich.
Signiert und unverschlüsselt	Die Nachricht wird nur signiert, jedoch nicht verschlüsselt. Ein Abhören der Nachricht ist somit möglich, jedoch kann diese nicht verändert werden.
Keine Sicherungsverfahren	Die Authentizität und Integrität der Nachricht kann nicht festgestellt werden

Tabelle 1: Beschreibung von möglichen Verschlüsselungsoptionen, Quelle: Eigene Darstellung.

³¹ Vgl. Lange/Iwanitz/Burke (2013), S. 243ff.

³² Vgl. International Electrotechnical Commission (Hrsg.) (2010).

- **Symmetrische Verschlüsselung**

Die symmetrische Verschlüsselung ist die ältere der beiden Verschlüsselungsphilosophien. Sie beruht darauf, dass es nur einen Schlüssel gibt, den beide Kommunikationspartner kennen müssen um Nachrichten auszutauschen. Visuell kann man es mit einem Tresor vergleichen, in dem die Nachricht eingeschlossen ist und nur jemand mit dem Schlüssel Zugriff darauf hat.³³

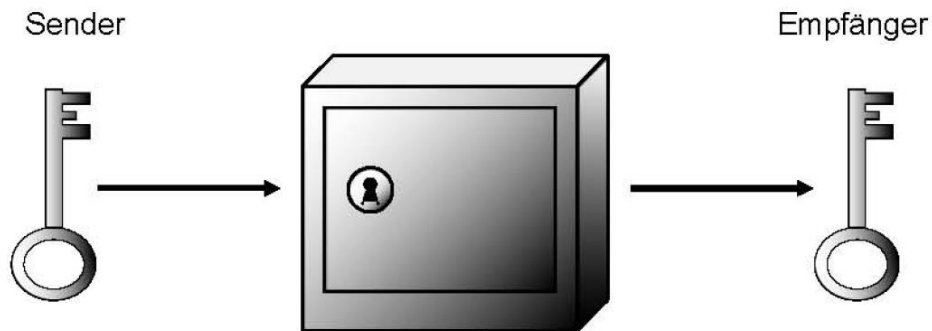


Abbildung 13: Visualisierung der symmetrischen Verschlüsselung, Quelle: Jörg (2010).

- **Asymmetrische Verschlüsselung**

Die asymmetrische Verschlüsselung ist immer durch zwei Schlüsselpaare gekennzeichnet. Den sogenannten öffentlichen (public) und den privaten (private) Schlüssel. Wenn eine Nachricht von A zu B sicher übermittelt werden soll wird die Nachricht mit dem Public Key des B verschlüsselt. Nur B kann die Nachricht mit dem nur ihm bekannten Private Key wieder entschlüsseln und lesen. In Abbildung 14 wird dieses Verfahren nochmal visuell dargestellt. Der Schlitz des Briefkastens entspricht dem Public Key, den jeder kennt. Nur der Briefkastenbesitzer hat den Schlüssel zum Öffnen und Entnehmen der Nachricht, sozusagen den privaten Schlüssel.³⁴

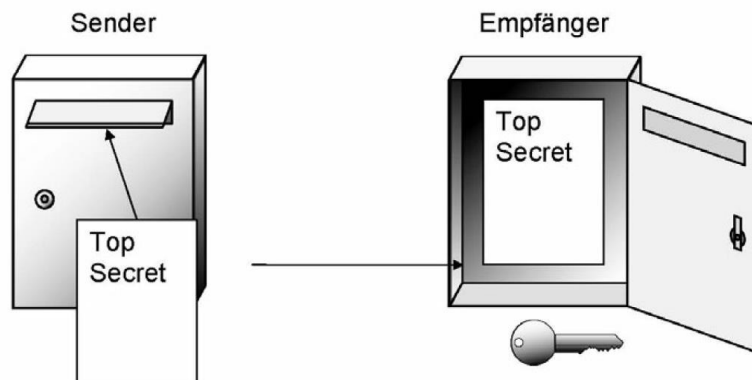


Abbildung 14: Visualisierung einer asymmetrischen Verschlüsselung, Quelle: Jörg (2010).

OPC UA nutzt eine Kombination aus diesen beiden Verschlüsselungsarten um einen sicheren Kanal zwischen Client und Server zu eröffnen. Grundsätzlich müssen sich die beiden Teilnehmer

³³ Vgl. Jörg (2010), S. 7ff.

³⁴ Vgl. Quelle: Jörg (2010), S.13ff.

identifizieren und verifizieren. Durch den sicheren Kanal wird dann eine Übertragungs-*Session* eröffnet, welche durch einen Zeitstempel und eine eindeutige ID festgelegt wird.

Das Verfahren des Handshakes zwischen Client und Server bis zum Aufbau eines Security Channels welches oben beschrieben wurde, wird im nachfolgenden Bild anschaulich erklärt.

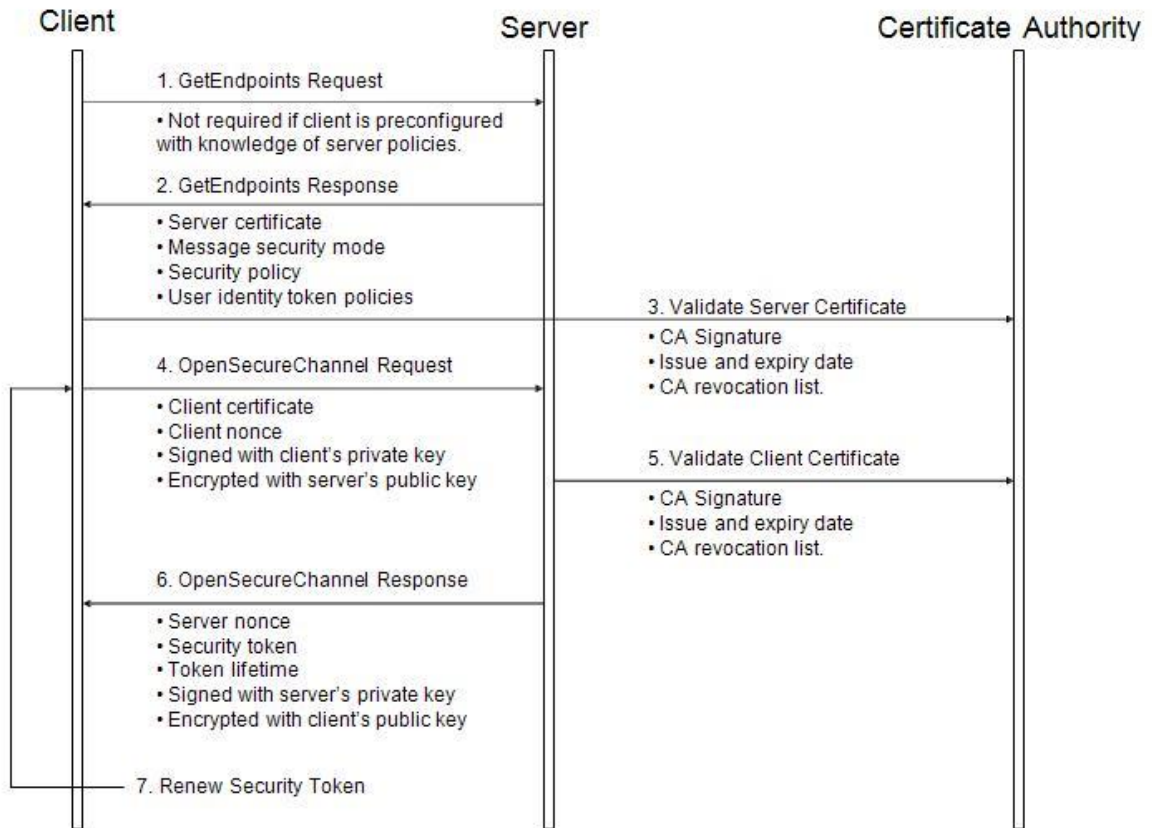


Abbildung 15: Aufbau eines Secure Channel, Quelle: International Electrotechnical Commission (Hrsg.) (2010).

3.1.5 Session

Nach dem Aufbau eines „sicheren Kanals“, in dem Client und Server sich gegenseitig authentifiziert haben und die Integrität der Übertragung sichergestellt ist, kommt es zum Aufbau einer Übertragungs-Sitzung (Session). Da diese *Session* zweckgebunden ist, wie zum Beispiel ein Client, der auf einen Wert am Server zugreift und diesen manipulieren möchte, muss der Zugriff aufgrund der vergebenen Berechtigung des Clients limitiert werden. Generell muss der Client und dessen Applikation seine Identität bekannt geben und überprüft werden. Genau dieser Abgleich passiert bei der Erzeugung einer *Session*.

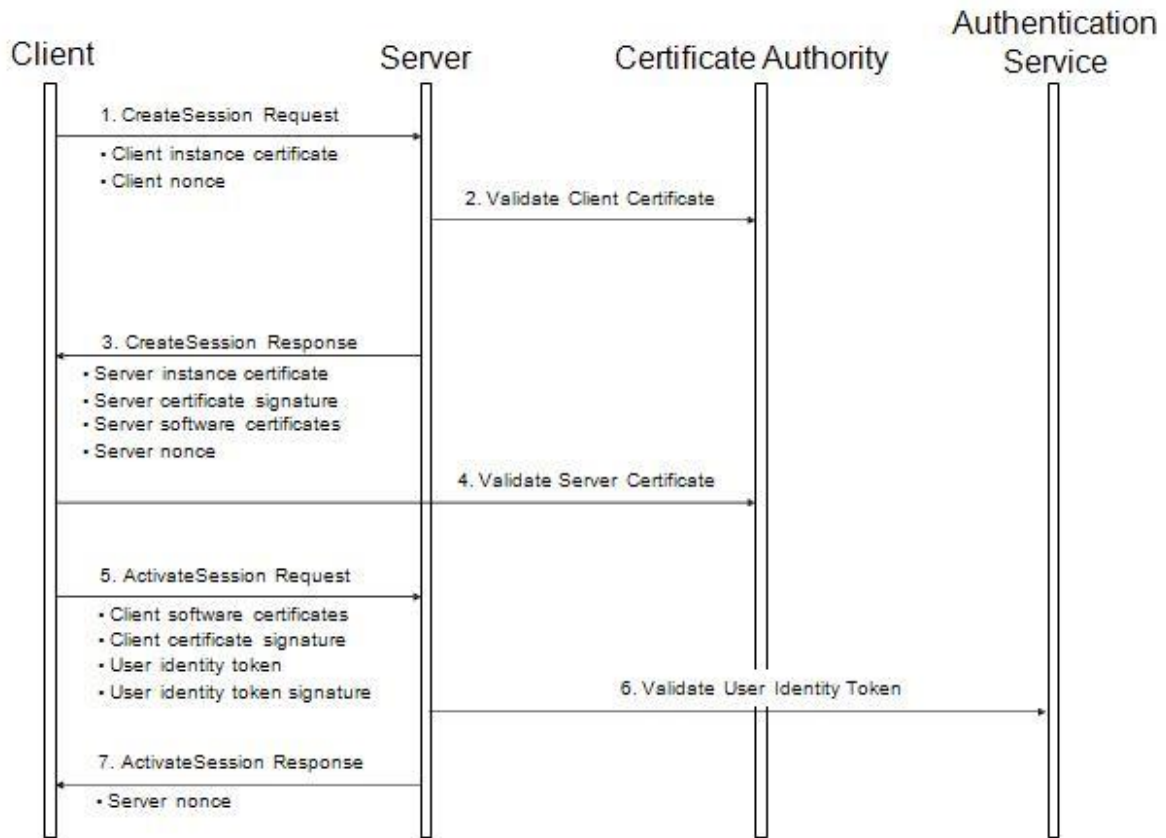


Abbildung 16: Aufbau einer Session, Quelle: International Electrotechnical Commission (Hrsg.) (2010).

Unter dem Begriff "Zugangskontrolle" wird ein anwendungsspezifischer Zugriff auf Daten und Funktionen am Server, entsprechend der vergebenen Rolle, die diese Applikation innehat, verstanden. Entsprechend der Abbildung 16, wird nach überprüfter Identität des Clients und Servers mit der CA eine aktive Session angefragt. Die Authentifizierung des Benutzers wird hier mittels *Brokered-Trust-Model* durchgeführt. Der Client erhält von einer vertrauenswürdigen dritten Stelle einen Token als Nachweis seiner Identität. Dieser wiederum sendet der Client an den Server, welcher den Token an dieser dritten Stelle erneut auf Gültigkeit überprüft. Mittels der Benutzer-Authentifizierung kann der Server nun aufgrund der Berechtigungen den Zugriff verweigern oder gewähren.³⁵

3.1.6 Zusammenfassung

OPC UA ist aufgrund der beschriebenen Funktionen und Sicherheitsmechanismen ein wichtiger Baustein zur Umsetzung von Industrie 4.0 Projekten. Vor allem die Interoperabilität und Unabhängigkeit der Plattform sind der Schlüssel zur CPS-Kommunikation in einer zukünftigen Produktionsanlage. Aufgrund der Offenheit können Datenankopplungen durch Erweiterungen an viele verschiedene Systeme realisiert werden. Offenheit und Plattformunabhängigkeit machen dieses System natürlich sehr komplex, welches eine genaue Planung der Struktur erfordert. Auch die implementierten Security Funktionen machen den Einsatz von OPC UA als Industrie-4.0-Kommunikation interessant. Generell kann durch den offenen Adressraum,

³⁵ Vgl. Lange/Iwanitz/Burke (2013), S. 242ff.

den objektorientierten Zugriff und der Vernetzungsmöglichkeit eine „Verwaltungsschale“, wie sie von der *Plattform Industrie 4.0* beschrieben wurde durchaus umgesetzt werden. Festzuhalten ist jedoch, dass eine echtzeitfähige Datenübertragung mittels OPC UA derzeit nicht möglich ist. Somit kann OPC UA derzeit keinen klassischen Feldbus in Anlagen ersetzen. Das ist bei der Ankoppelung von CPS in ein IoT (Internet of Things) aber derzeit noch nicht notwendig. Die OPC Foundation arbeitet momentan an einer Umsetzung zur Integration der Echtzeitfähigkeit mittels TSN (Time Sensitive Network), welche noch nicht abgeschlossen und veröffentlicht wurde.³⁶

3.2 Cloud-Computing

Anlagen und Produktionsprozesse bzw. ganze Fertigungsstraßen werden in Zukunft durch Einführung von CPS Technologien barrierefrei miteinander kommunizieren und Daten bereitstellen. In der Theorie sollen diese CPS im Internet der Dinge selbstständig koordiniert und gesteuert werden, welches eine selbstkonfigurierbare und optimierbare Produktion ermöglicht. Diese Daten und Kommunikationswege müssen durch Dienste gesteuert und verwaltet werden. Cloud-Systeme können eine Antwort auf die Frage sein, wie solche Datenmengen effizient und vor allem dauerhaft gespeichert und zur Verfügung gestellt werden können. Ein Cloud-System ist ein verteiltes System, das an verschiedenen Orten auf unabhängige physikalische Server zugreift und Daten und Dienste bereitstellt. Dieses verteilte Netzwerk kann eine hohe Verfügbarkeit der Daten gewährleisten. Für den Nutzer solcher Cloud-Systeme erscheint es jedoch wie ein zentraler Dienst, der zur Verfügung steht. Somit stellt ein Cloud-System eine perfekte Schnittstelle da, um CPS in einem übergeordneten Netzwerk abzubilden.³⁷

Aufgrund der Tatsache, dass die Kommunikation zwischen der Cloud und den Produktionsanlagen über ein öffentliches Netzwerk, dem Internet, geführt wird, muss hier besonders auf die Datenübertragungssicherheit geachtet werden. Es ergeben sich durch die Offenheit zwangsläufig Sicherheitsrisiken, die bei der Planung beachtet werden sollen:³⁸

- Anlagensicherheit vor Manipulation durch Dritte (Sabotage)
- Kommunikationssicherheit, Abhören bzw. Fälschen von Daten
- Informationssicherheit, Know-How Schutz

Wie bereits im Abschnitt 3.1.1 „OPC Unified Architecture“ müssen auch zwischen Cloud und Anlage Sicherheitsaspekte berücksichtigt werden. Somit müssen Cloud-Anbindungen neben dem Bereitstellen der Daten und Dienste, die Unversehrtheit und die Vertraulichkeit der Daten gewährleisten. Dies stellt eine nicht unerhebliche Anforderung an Cloud-System-Anbieter, welche bei der Auswahl einen hohen Stellenwert haben sollten.

Security spielt bei Industrie 4.0 eine tragende Rolle, da viele Themengebiete über diese zentrale Datenhaltung abgewickelt werden sollen:

³⁶ Vgl. Foundation (2016), Online-Quelle [04.08.2016].

³⁷ Vgl. Bauernhansl/Hompel/Vogel-Heuser (2014), S. 238ff.

³⁸ Vgl. Bauernhansl/Hompel/Vogel-Heuser (2014), S. 409ff.

- Wartung
- Verwaltung von Produktionsressourcen
- Aussteuerung von Produktionsanlagen bzw. CPPS
- Reporting aktueller Produktionsdaten
- Usw.

In der vorliegenden Arbeit soll der Fokus auf die Speicherung der Daten, welche von den CPS bereitgestellt werden, gelegt werden. In weiterer Folge kann durch die Analyse der Daten ein Mehrwert für den Anlagenbetreiber bzw. Anlagenhersteller generiert werden.

3.2.1 Speicherung und Analyse von Daten

Derzeitige Produktionsanlagen zeichnen sich durch strikte hierarchische Informationsstrukturen aus, welche in ihrer Flexibilität stark eingeschränkt sind. Rohdaten, welche für die Prozesssteuerung eines Anlagenteils notwendig sind (zum Beispiel der aktuelle Istwert einer Temperaturregelung), werden ausschließlich gekapselt in einzelnen Funktionen (zum Beispiel ein Regler) verwendet. Die Aufzeichnung dieser Daten kann jedoch aufgrund von zusammenhängenden Ereignissen oder Abhängigkeiten durchaus Relevanz besitzen, welche aufgrund derzeitiger Überlegungen noch nicht erkannt wurde.

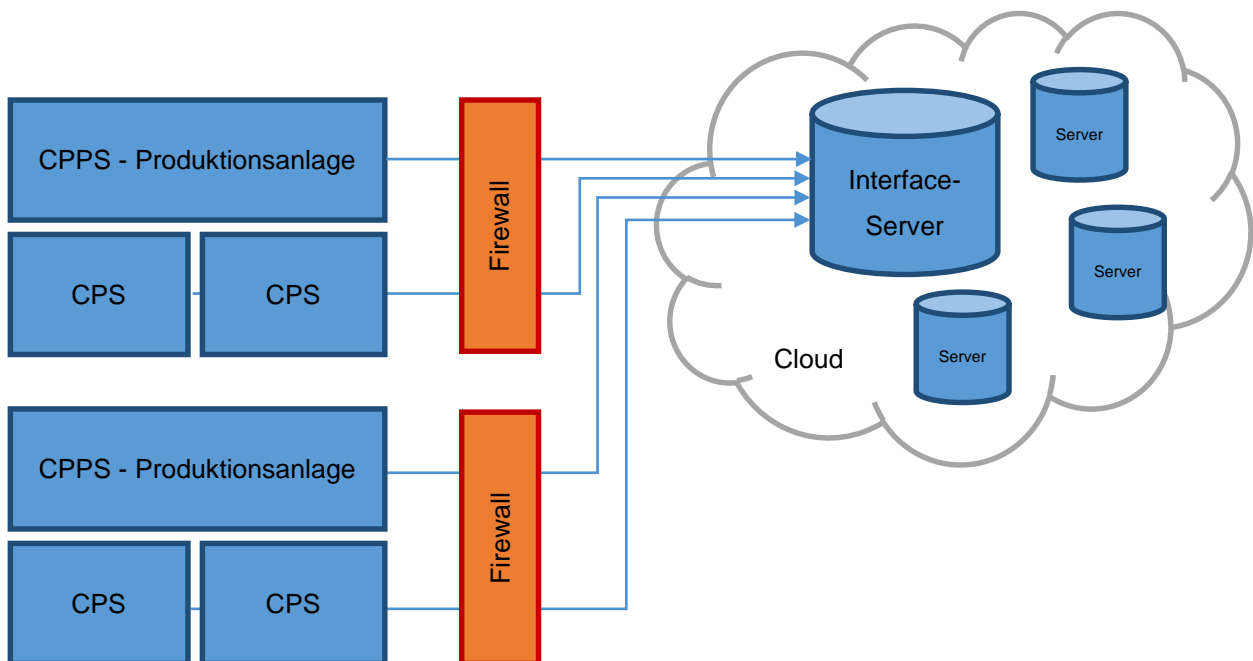


Abbildung 17: Datenspeicherung aus der Anlage in die Cloud, Quelle: Eigene Darstellung.

Die vorhin erwähnte hierarchische und vor allem lokale am Ort des Geschehens vorhandene Infrastruktur macht diese Art der Automation eines Produktionsprozesses notwendig. Die Datenflut, welche durch das

Übertragen und Speichern jeglicher erfasster Prozessgrößen entsteht, wäre durch lokale Infrastruktur wie sie derzeit in Produktionsanlagen eingesetzt wird, nicht in den Griff zu bekommen.

Cloud-Systeme können durch ihre dezentralen und dynamischen agierenden Strukturen diesem Problem Herr werden. Nun stellt sich die Frage, warum sollte man alle diese Prozesswerte überhaupt erfassen und speichern. Hier kann ein Beispiel aus der Finanzwelt mehr Einblick geben.

3.2.2 Betrugserkennung bei Kreditkartenzahlungen durch Big Data und Analyseverfahren

Im Bereich der Finanzwelt sind Verfahren zur Erfassung und Speicherung von Transaktionen bereits jetzt State of the Art. Vor allem im Bereich der Kreditkartenunternehmen werden diese Daten verwendet um Betrugsfälle zu erkennen bzw. Betrug präventiv entgegenzuwirken. Hierzu werden Transaktionen aufgrund von bekannten Handlungsmustern auf Plausibilität geprüft. Um Handlungsmuster zu erkennen werden sogenannte „Regeln“ eingesetzt, welche nach dem „wenn-dann“ Prinzip funktionieren. Somit kann bekanntes Wissen verknüpft und neues Wissen generiert werden. Wichtig dabei ist, dass die vorhandenen Daten in ihrer Menge und Qualität einen gewissen Standard einhalten. Festzuhalten ist, dass „schlechte“ Daten auch ein „schlechtes“ Handlungsmuster zur Folge haben.

Um nun mit diesem generierten Wissen auch Ereignisse zu erkennen (in diesem Fall einen möglichen Betrug), müssen analytische Verfahren eingesetzt werden. Generell kann festgehalten werden, dass solche Regelwerke nur Ereignisse aufgrund von Erfahrungen und Expertenwissen bewerten können. Die Kombination aus bereits vorhandenem Wissen, Erfahrung und dem Einsatz von mathematisch-statistischen Verfahren kann die notwendige Flexibilität und Dynamik garantieren, um auch zukünftige noch nicht bekannte Situationen zu erkennen (Data Mining). Ein Beispiel wäre die Erkennung einer möglichen unrechtmäßigen Abbuchung aufgrund der geografischen Lage, zum Beispiel eine Abbuchung in Peking, obwohl die letzte Buchung vor drei Stunden in Wien getätigt wurde.³⁹

3.2.3 Einsatz in der Industrie

Durch dieses Beispiel wurde ersichtlich, dass es durchaus in anderen Bereichen wichtig und notwendig ist Daten zu speichern und zu analysieren. Auch in der Industrie können neue Anwendungsszenarien entwickelt werden, welche mit Data Mining und Analyse eine Verbesserung bzw. Weiterentwicklung der Produktionsanlagen bringt. Nachfolgend werden Szenarien aufgezeigt, welche der Firma Rosendahl in Bezug auf Ihre Produktionsanlagen Mehrwert durch *Cloud-Computing* und *Analytics* generieren können. Diese wurden gemeinsam mit Rosendahl Nextrom identifiziert und nachfolgend aufgelistet:

1. Erkennung von Flaschenhälsen in der laufenden Produktion
2. Fehlererkennung und Fehleranalyse aufgrund der Verknüpfung von Prozessdaten/Umweltbedingungen/Nutzerverhalten
3. Informationen für Entwicklung und Produktdesign generieren

³⁹ Vgl. SAS Institute GmbH (2013), Online-Quelle [08.08.2016].

4. Fehlervermeidung durch Analyse bisheriger Ausfälle und Erkennung von kritischen Betriebszuständen
5. Erkennen des Optimierungsbedarfs von Ressourcen wie zum Beispiel Energie⁴⁰

3.2.4 Arten von Cloud-Computing

Das Portfolio von Cloud-Anbietern wird häufig in die anzutreffenden Formen kategorisiert. Hierbei handelt es sich um eine Einteilung aufgrund der Erreichbarkeit innerhalb oder außerhalb des Unternehmens. Nachfolgend werden die Varianten beschrieben:⁴¹

- **Public Cloud**

Eine der im privaten Bereich am weit verbreitetsten Formen ist die Public Cloud. Deshalb wird der Begriff Cloud meist mit der Public Cloud assoziiert. Es handelt sich dabei um eine Cloud, welche durch einen dritten, externen Service Provider angeboten wird und somit vielen verschiedenen Kunden zur Verfügung gestellt werden kann.

- **Private Cloud**

Viele Unternehmen setzen auf diese Form der Cloud. Die Infrastruktur und der Bezug der Leistung erfolgt vom eigens geführten Cloud System, welches nur bei expliziter Berechtigung Externen zur Verfügung gestellt werden kann. Diese Form bedarf sehr viel Aufwand und Eigenleistung.

- **Hybrid Cloud**

Ein solches System ist eine Kombination, und setzt sich aus den Formen Public und Private Cloud zusammen. Meist geht es darum, eigene Kapazitätsgrenzen zu umgehen. Das bedeutet, dass ein Unternehmen auf Leistungen einer Public Cloud zugreift, um die eigene Private Cloud zu entlasten oder zu ergänzen.

- **Community Cloud**

Diese Form besteht dadurch, dass ein Konglomerat an Firmen dieselbe Cloud-Infrastruktur nutzt und als gemeinsame Plattform versteht. Meist handelt es sich hierbei um Organisationen, welche ein gemeinsames Ziel verfolgen und somit eine Interessensgemeinschaft bilden.

Neben der Form der Cloud werden noch die verschiedenen Services bzw. Leistungen unterschieden, die durch ein Cloud System übernommen werden können.⁴²

- **Infrastructure as a Service (IaaS)**

Hierbei handelt es sich um das zur Verfügungstellen von Rechen-, Speicher, und Netzwerkleistung.

⁴⁰ Vgl. Niggemann (2014).

⁴¹ Vgl. Filler (2015) S.22ff.

⁴² Vgl. Filler (2015) S.26.

- **Platform as a Service (PaaS)**

Die Cloud bietet dem Nutzer Entwicklungstools, die über standardisierte Schnittstellen verfügen. Dadurch kann der Kunde Softwareanwendungen in der Cloud entwickeln und muss selbst nicht über das nötige Equipment verfügen.

- **Software as a Service (SaaS)**

Selbst erstellte Software kann über die Cloud bezogen oder angebunden werden. Es gibt auch Varianten in denen fertig standardisierte Softwareanwendungen über Bibliotheken zur Verfügung gestellt werden.

3.3 Datenanalyse – Von Big Data zu Smart Data

Die vorhin beschriebene Tatsache macht es notwendig, Wege und Mittel zu finden, um Datenstrukturen sinnvoll zu analysieren und auszuwerten. Abweichungs- und Fehlererkennung sind aufgrund der steigenden Komplexität der Produktionsanlagen ein wichtiger Forschungsbereich. Durch die digitale Anbindung und Verknüpfung von Produkten und Produktionsanlagen über das Internet der Dinge (IoT) werden Unmengen von Daten produziert und zugänglich gemacht. Diese Daten werden aufgrund Ihrer Datenmenge und Frequenz als Big Data bezeichnet. Derzeit müssen Fehler durch Implementierung in der Steuerungsebene erfasst und ausgewertet werden. Um dies durchführen zu können ist es notwendig mögliche Fehler bereits im Vorhinein zu kennen und die Erfassung dieser Fehler im Steuerungsprogramm zu definieren. Ziel sollte es jedoch sein, diese Fehlererkennung aus der Steuerungsebene heraus in ein übergeordnetes System zu heben. Des Weiteren muss die Fehler- und Abweichungserkennung nicht statisch, sondern lernend und durch sich selbst erweiternd ausgeführt werden. Aus der Analyse und Weiterverarbeitung von Big Data können somit Tendenzen oder Ereignisse erkannt werden. Diese Ergebnisse werden als Smart Data bezeichnet. In der nachfolgenden Abbildung kann der gesamte Prozess einer Datenanalyse betrachtet werden.

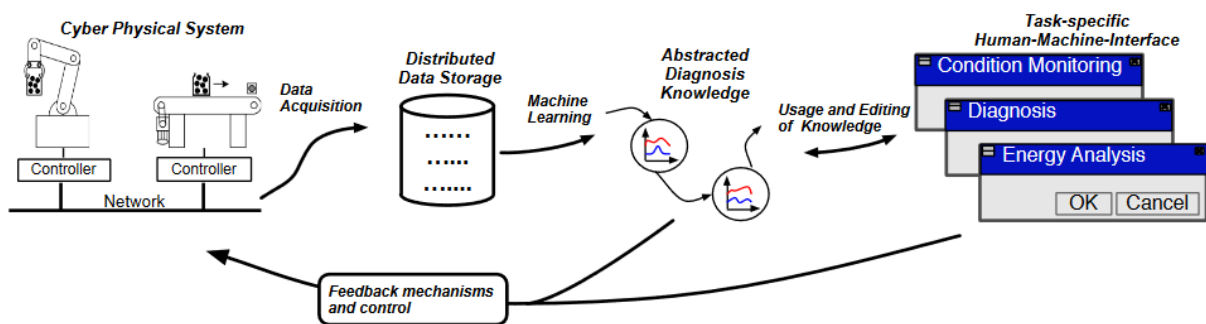


Abbildung 18: Prinzip der Datenanalyse, Quelle: Niggemann/Biswas/Kinnebrew/Khorasgani/Volgmann/Bunte (2015).

3.4 Machine Learning

Um diese gewünschten Erkenntnisse aus der Vielzahl an Daten zu erlangen, bedarf es automatischen Mechanismen. Hierzu wird der Begriff *Machine Learning* (Maschinelles Lernen) bzw. *Data Science* verwendet. Machine Learning basiert auf der Idee, dass aufgrund von Erfahrung eine Aufgabe besser

gelöst werden kann. Tom M. Mitchell hat das in seinem Buch „Machine Learning“ aus dem Jahre 1997 wie folgt formuliert:

„A computer program is said to learn from **experience E** with respect to some class of **tasks T** and **performance measure P**, if its performance at tasks in T, as measured by P improves with experience E.“⁴³

Man kann sofort erkennen, dass diese Art des automatischen Lernens immer Daten aus der Vergangenheit erfordert. Je mehr Informationen über einen Sachverhalt vorliegen, umso genauer kann eine Vorhersage für die Zukunft erfolgen. Diese spezielle Art der Analyse wird in der Fachliteratur als *Predictive Analysis* bezeichnet.⁴⁴

Es wird der Ansatz der modellbasierten Anomalie-Erkennung erforscht.⁴⁵ Hierbei wird durch ein Anlernen des Normalzustandes einer Anlage, ein Modell generiert, welches danach als Referenz herangezogen wird. Dabei ist es nicht notwendig das genaue mathematische Modell und deren Eigenschaften (Stabilität, Konvergenz, usw.) zu kennen. Das Modell beruht auf I/O Daten der Anlage, welche den Prozess beschreiben. Die Abweichungserkennung wird dann durch den Vergleich des Ist Zustandes mit dem Referenzmodell realisiert.⁴⁶

Durch dieses Szenario kann eine vorrausschauende Wartung (Predictive Maintenance) realisiert werden. Bei der Umsetzung solcher Projekte ist es wichtig vorab zu klären was das Ziel bzw. Problem ist um die richtigen Lösungswege ermitteln zu können. Im Falle vorrausschauender Wartung werden nachfolgend ein paar Beispiele vorgestellt:

- Voraussage ob ein Gerät oder eine Komponente ausfallen wird noch bevor es eintritt,
- Diagnose von Ausfällen und deren Gründen, um in weiterer Folge Qualitätsprobleme zu beseitigen,
- Voraussage wann ein Gerät ausfallen wird

3.5 Datensicherheit – Defense in Depth

Aufgrund der durchgehenden Vernetzung der CPS und CPPS in einem produzierenden Betrieb gibt es nicht nur positive Aspekte. Durch die Anbindung kritischer Anlagen und Prozesse und der höheren Zahl an Entry Points in das interne Netzwerk wird das Risiko von Cyber-Angriffen drastisch erhöht. Damit das eigene Netzwerk und die darin befindlichen CPS davor geschützt werden, müssen auch zum Thema Netzwerk-Security Lösungen gefunden werden. Eine mögliche Strategie welche die IEC 62443 als Orientierung vorschlägt, ist die sogenannte „Defense in Depth“ - Strategie. Diese Strategie basiert auf einer mehrschichtigen Verteidigungsmaßnahme, welche zum Beispiel durch Segmentierung des Netzwerks, VPN, Firewalls und Benutzerauthentifizierung realisiert wird.⁴⁷ Einen vollständigen Schutz kann *Defense in*

⁴³ Mitchell (1997) S.2.

⁴⁴ Vgl. Roger Barga (2015), S. 5ff.

⁴⁵ Vgl. Niggemann/Biswas/Kinnebrew/Khorasgani/Volgmann/Bunte (2015).

⁴⁶ Vgl. (Zho13), S. 6.

⁴⁷ Vgl. Lass/Kotarski (2014), S. 405ff.

Depth nicht bieten. Jedoch werden die Eintrittsbarrieren durch die vielschichtige Abwehrstrategie sehr hoch gehalten. Angriffe auf Basis der Standardmethodiken wie zum Beispiel Portscans, Bruteforce, usw. können mittels Defense in Depth sehr gut aufgedeckt und unterbunden werden, womit auch das Interesse der meisten Angreifer verloren gehen wird. Generell kann die Strategie beliebig granular eingesetzt werden, wobei durch zunehmende Komplexität auch die Kosten erheblich steigen. Wie bereits erwähnt wird die Sicherheit des IT-Netzes durch mehrere Barrieren und Schichten erhöht. In der Abbildung 19 wird das Defense-in-Depth Konzept grafisch dargestellt und nachfolgend die einzelnen Schichten näher beschrieben.⁴⁸

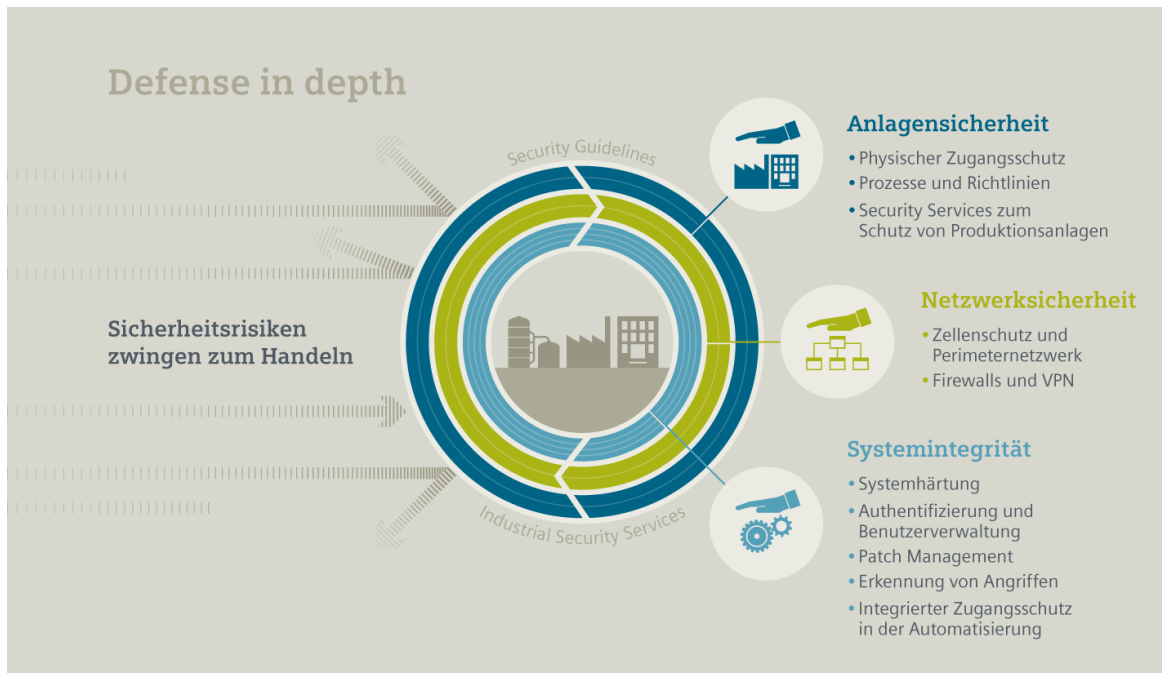


Abbildung 19: Defense in Depth Überblick, Quelle: Siemens AG

3.5.1 Anlagensicherheit

Die erste Eintrittsbarriere wird durch ein durchgängiges Sicherheitskonzept der Anlage bzw. des Standortes gewährleistet. Der Zugriff auf Daten muss nicht zwangsläufig von Extern erfolgen. Bei der Anlagensicherheit wird der physische Zugangsschutz betrachtet, welcher durch Prozesse und Richtlinien gelebt werden muss. Nicht berechnigte Personen können so von Zugängen innerhalb des Unternehmens ferngehalten werden. Oft werden diese Vorschriften und Richtlinien in Unternehmen durch *Security Guidelines* den Mitarbeitern zur Verfügung gestellt. Regelmäßige Auditierung und Überprüfung der Maßnahmen sollen dabei helfen die Anlagensicherheit dauerhaft zu erhöhen.

3.5.2 Netzwerksicherheit

Die Netzwerksicherheit ist die Ebene, die am häufigsten mit Security assoziiert wird. Dabei handelt es sich um die Absicherung des internen Netzes vor externen Zugriffen. Es gibt mehrere Technologien, die jedoch nur in Kombination miteinander einen effektiven Schutz bieten können. Die Segmentierung des Netzwerks

⁴⁸ Vgl. Lass/Kotarski (2014) S.404ff.

und Abtrennung mittels Firewalls wäre eine dieser Kombinationen. Durch die Segmentierung erhält man einen effektiven Zellschutz, welche zusätzlich durch Firewalls geschützt werden können. Somit kann verhindert werden, dass bei erfolgtem Zugriff auf einen Bereich automatisch alle anderen Netzwerksegmente frei zugänglich sind. Ein weiterer Punkt der Netzwerksicherheit ist die Kommunikation zwischen den einzelnen Netzwerksegmenten. Diese Kommunikation muss so reglementiert sein, dass jeglicher nicht für die Anwendung notwendiger Datenverkehr blockiert wird.⁴⁹

3.5.3 Systemintegrität

Die Systemintegrität beschäftigt sich mit dem Thema Authentifizierung und Verschlüsselung, sowie mit der Benutzerverwaltung. Es ist darauf zu achten, dass Kommunikation generell gewissen Verschlüsselungsstandards entspricht. Zugangspunkte zu anlagenspezifischen Komponenten, wie zum Beispiel *Speicher-programmierbare Steuerungen* oder *Human-Maschine-Interfaces*, müssen mittels Benutzer und Passwort geschützt sein. Diese Maßnahmen steigern die Systemhärte und den Know-How-Schutz des Unternehmens.

⁴⁹ Vgl. Lass/Kotarski (2014) S.405ff.

4 DER EXTRUDER - ROEX

Der Extruder ist eine wesentliche Komponente von Fertigungsanlagen in der Kabelindustrie. Seine Aufgabe ist es, auf die vorbeigeführte Phase eines Leiters den Isoliermantel aufzubringen. Die interne Firmenbezeichnung für den Extruder lautet *Rosendahl Extruder* (ROEX).

4.1 Allgemein

Im Fertigungsprozess eines Kabels werden mehrere Schritte durchlaufen. Der Prozess beginnt bei der Bündelung und dem Verseilen der Phasen. Die Anordnung der einzelnen Produktionsmaschinen ist eine sogenannte Fertigungsstraße oder -linie, da das Produkt aufgrund seiner Physik und der Fertigungseinheit nur mittels eines kontinuierlichen Fertigungsprozesses effizient gefertigt werden kann. Ein Schritt im Fertigungsprozess ist das Aufbringen des Isolationsmaterials bzw. des Kunststoffmantels. Der Extruder übernimmt hierbei das Schmelzen des Kunststoff-Granulates und dessen Beförderung zum nachfolgenden Gerät bzw. zum Werkzeug in dem die Phase ummantelt wird. Diese Maschinenkomponente ist immer Teil einer Fertigungsanlage im Umfeld der Kabelindustrie.

4.2 Aufbau

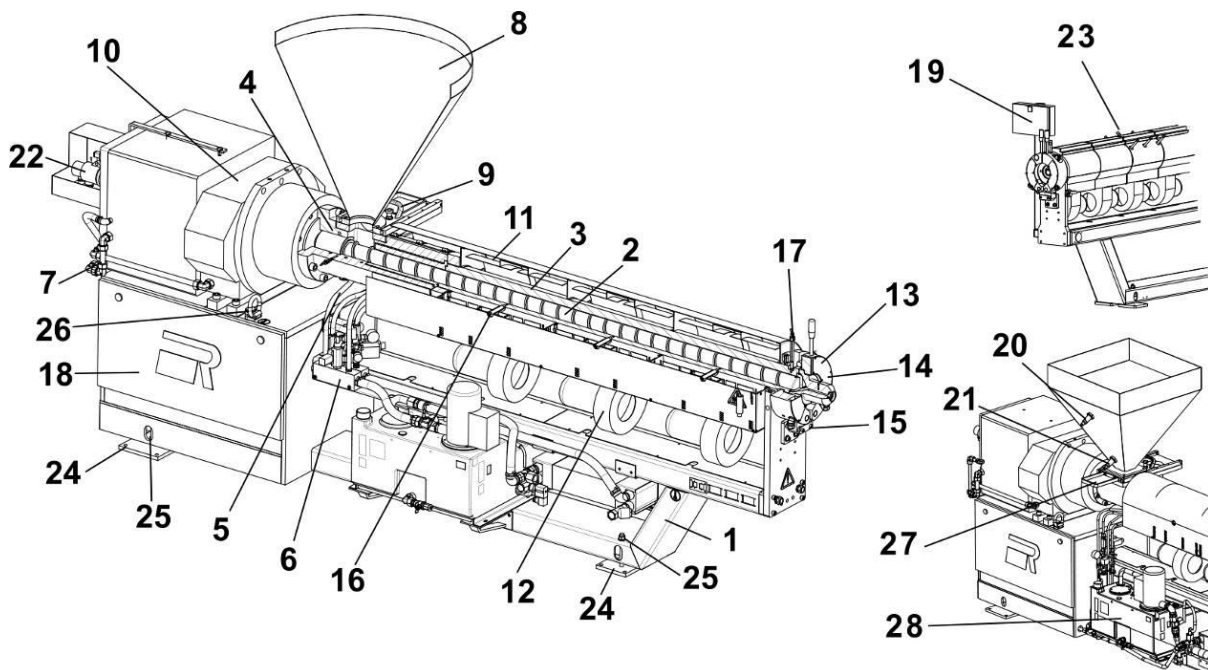


Abbildung 20: Technische Übersicht – Extruder fix am Boden befestigt, Quelle: Betriebs- und Wartungsanleitung ROEX S.14.

- | | |
|-------------------------------------|--|
| 1. Rahmen | 16. Temperaturfühler (Option) |
| 2. Schnecke | 17. Druck-Fühler |
| 3. Zylinder | 18. Schaltschrank |
| 4. Büchse | 19. Steck-Anschlüsse |
| 5. Büchsen-Temperierung | 20. Füllstands-Überwachung - Nachfüllen |
| 6. Kühlverteiler | 21. Füllstands-Überwachung – Stop |
| 7. Motor-Kühlung | 22. Drehbare Zweiweg-Zuführung mit Sifonrohr - für Schneckenkühlung (Option) |
| 8. Trichter (Option) | 23. Gaseinspritzung (Option) |
| 9. Trichter-Verschiebung | 24. Fixierschrauben |
| 10. Getriebe mit integrierte Motore | 25. Stellschrauben (zur Höheneinstellung) |
| 11. Heizbänder | 26. Hebeösen |
| 12. Radialventilator | 27. Einzug-Schutz |
| 13. Spannsegment | 28. Kühl-Baugruppe (Option) |
| 14. Heizpatronen (nicht sichtbar) | |
| 15. Segment-Halterung | |

4.3 Funktionsweise

Der Extruder wird über die Granulat-Zuführung (evtl. mit Trichter) mit dem Kunststoff-Rohmaterial versorgt. Über die Schnecke (siehe Abbildung 20, Punkt 2) wird das Rohmaterial mit Druck an den Heizzonen vorbei gefördert. Das Kunststoffgranulat wird verflüssigt und in den Spritzkopf eingebracht. Im Spritzgusswerkzeug wird der flüssige Kunststoff mit Druck auf die Phase aufgebracht. Der Produktionsprozess ist ein kontinuierlicher Prozess, wobei das Kabel mit kontinuierlicher Geschwindigkeit am Extruder vorbeigeführt wird. Damit der Kunststoff immer konstant aufgebracht werden kann, ist eine exakte Regelung der Fördergeschwindigkeit der Schnecke und der Temperatur der Heizzonen notwendig.

Um eine konstante Materialzuführung zu gewährleisten wird dieser Bereich der Büchse (4) durch eine sogenannte *Büchsen-Temperierung* geregelt (5). Diese Maßnahme verhindert das Verkleben der Eintrittsöffnung aus dem Trichter bzw. der Materialzuführung.

Das Granulat wird mittels Förderschnecke zum Spritzkopf befördert und dabei über Heizbänder erwärmt und geschmolzen. Damit ein konstantes Temperaturprofil über den gesamten Förderweg erreicht werden kann, sind alle Heizzonen mit Radialventilatoren ausgestattet, welche ein Kühlen der Heizzonen ermöglichen. Dies ist notwendig, da bei einem möglichen Stillstand des Extruders der Kunststoff überhitzen könnte. Die Zylindertemperatur der Zonen wird über Temperaturfühler geregelt.

Der Extruder ist variabel einsetzbar, da das Spritzgusswerkzeug bzw. der Adapter zum Austritt des Kunststoffes getauscht werden kann. Dazu wird mittels Spannsegmenten (13) der jeweilige Adapter auf den Zylinder (3) gespannt. Auch das Spannsegment wird zusätzlich beheizt (14) um bis zum Adapter eine gleichbleibende Temperatur halten zu können.

Damit der Extruder immer mit Granulat versorgt wird, wird der Füllstand des Trichters (8) mit zwei Sensoren überwacht (20, 21). Der obere Sensor meldet dem Bediener, dass Granulat nachgefüllt werden muss. Wird der Mindestfüllstand unterschritten, meldet der zweite Sensor diese Unterschreitung und es wird ein Stopp der Anlage ausgelöst.

5 DATENANBINDUNG

Der Extruder wird mittels einer *Speicher-programmierbaren Steuerung* (SPS) geregelt und angesteuert. Dabei ist zu beachten, dass die Regelung der Temperatur und des Vorschubs der Spindel im Verbund mit einem Frequenzumrichter, den Schwerpunkt der Maschinensteuerung ausmacht. Die Bedienerschnittstelle wird über ein von der Firma Rosendahl Nextrom eigens entwickeltes PC-System realisiert (RIO). Somit kann ein Bediener der Anlage über einen zentralen Rechner Einstellungen am System vornehmen, sowie Fehlermeldungen auslesen. Die Anbindung an die einzelnen Feldgeräte (Heizungsregler, Frequenzumrichter) sind mit dem Ethernet Standard „PROFINET IO“ realisiert. PROFINET IO ist ein echtzeitfähiger Anlagenbus, welcher es ermöglicht Sollwerte und Istwerte zwischen den Teilnehmern auszutauschen. Der zentrale Leitrechner wird über TCP/IP angebunden und über eine offene Verbindung mit Daten versorgt, welcher über Datenbausteinzugriffe realisiert wird.

5.1 Vom herkömmlichen Extruder zur CPPS-Komponente

Der erste Schritt zum Wandel zur CPPS-Komponente, ist der Einsatz geeigneter Hardware-Komponenten, die die gewünschten Kommunikationsschnittstellen beinhalten. In diesem Fall muss besonders auf die OPC-UA-Integration geachtet werden. Die Firma Siemens bietet mit der aktuellsten Hardwareplattform S7-1500 Firmware Stand 2.0 die Möglichkeit einen OPC-UA-Server direkt auf der lokalen Steuerung zu realisieren.

5.2 OPC UA-Implementierung

Der Extruder ist per Definition ein CPPS. Er beinhaltet verschiedene CPS und verknüpft deren Funktionen zu einer Produktionseinheit. Die CPPS sollte anschließend im IoT (Internet of Things) erreichbar sein.

Um die Anbindung an das IoT zu erreichen, muss als Erstes die derzeitige Zielhardware und -software, welche den Extruder automatisiert, erfasst werden. Da diese Hardware jedoch noch nicht den OPC-UA-Stack implementiert hat, stellt sich die Frage, welche Möglichkeiten zur Integration von OPC UA zu Verfügung stehen. Nachfolgend werden verschiedene Ansätze zur Integration aufgelistet:

1. Die Implementierung eines externen OPC-UA-Servers und proprietäre Anbindung der Daten aus dem Steuerungsprogramm. Der Cloud-Dienst kann direkt mittels OPC UA oder mit Softwareschnittstellen (herstellerabhängig) angebunden werden. Vorteil dieses Lösungsansatzes ist, dass an der Anlage keine Umbauten aufgrund von Hardwareänderungen (SPS und Peripherie) durchgeführt werden müssen. Die Software muss um die Datenverbindung erweitert werden, jedoch hält sich dieser Aufwand in Grenzen. Der Nachteil ist der Medienbruch zwischen Steuerung und OPC-UA-Server, der durch die proprietäre Anbindung eingeschleust wird. Bei Änderungen an der Anlage aufgrund von Weiterentwicklungen, Störungsbeseitigung oder Kundenwünschen muss die Datenschnittstelle und die OPC-UA-Implementierung bearbeitet, geprüft und validiert werden. Wie schon in vorigen Kapiteln beschrieben sind solche Medienbrüche für Plug&Play Mechanismen schwer berücksichtigbar.

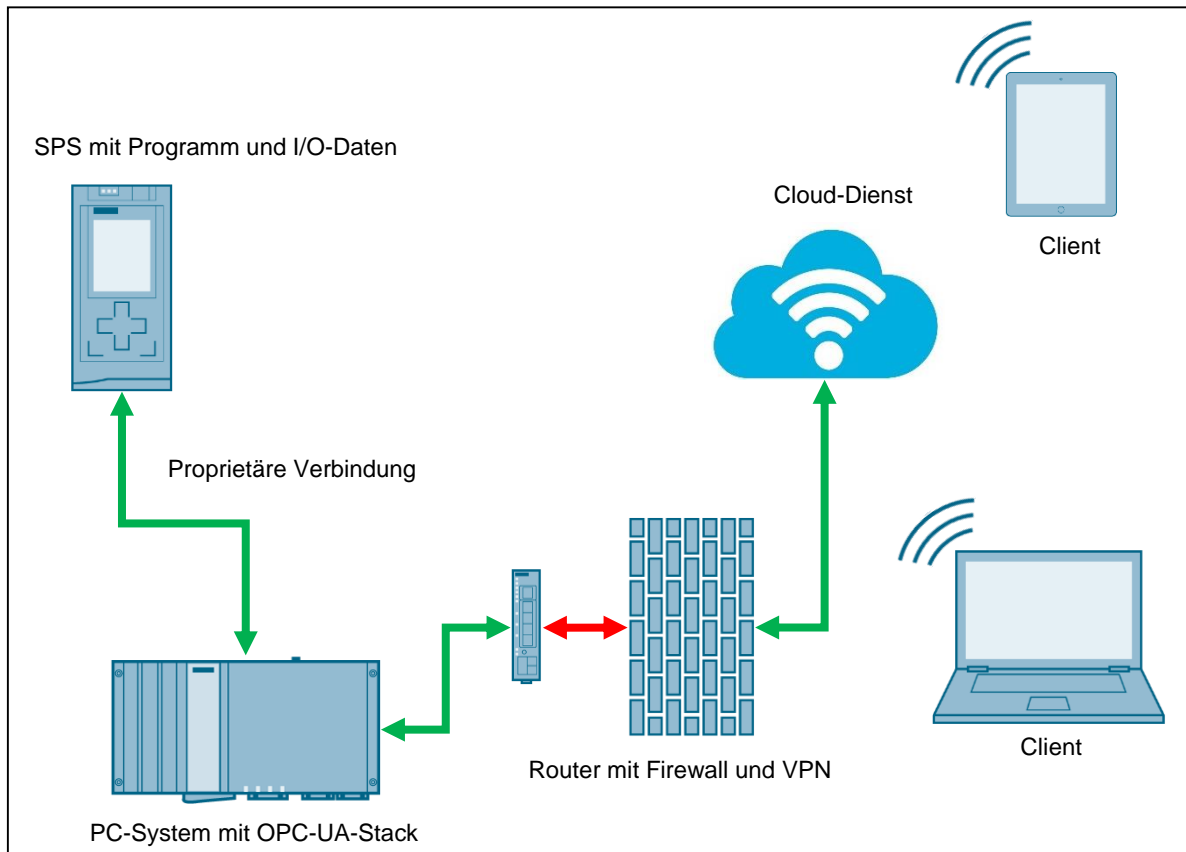


Abbildung 21: Variante 1 – Visuelle Darstellung, Quelle: Eigene Darstellung.

2. Die Anpassung der Hard- und Software, so dass der OPC-UA-Stack direkt im SPS-Code implementiert ist. Dies wird in der Regel dadurch erreicht, dass Hersteller den OPC-UA-Stack in der Firmware der SPS bereits integrieren. Ein externer OPC-UA-Server, welcher über proprietäre Verbindungen angebunden werden muss, entfällt dadurch gänzlich. Der Vorteil ist, dass bei nachträglichen Änderungen im Programm auch die OPC-UA-Objekte automatisch aktualisiert werden. Ein händisches Eingreifen ist somit hinfällig. Der Nachteil ist die Hardware- und Software-Anpassung, da im schlimmsten Fall ein Herstellerwechsel und ein Know-How-Aufbau notwendig werden.

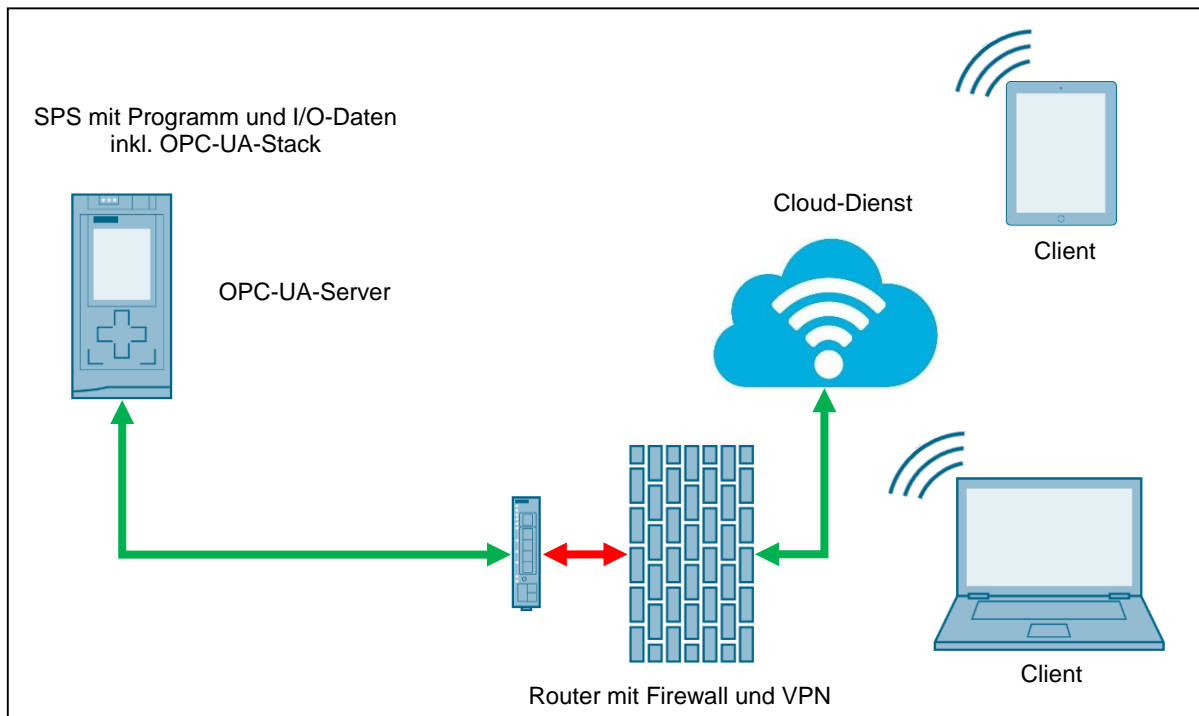


Abbildung 22: Variante 2 – Visuelle Darstellung, Quelle: Eigene Darstellung.

3. Eine weitere Möglichkeit wäre eine Kombination aus den ersten beiden Varianten. Der Unterschied besteht darin, dass die Kommunikation zwischen der SPS und dem PC vor Ort mittels OPC-UA-Browser durchgeführt wird. Die SPS stellt mittels implementierten OPC-UA-Server die notwendigen Daten zur Verfügung. Der PC wird mit einer OPC-UA-Client-Applikation ausgestattet um die Daten entweder mittels OPC UA oder anderen Kommunikationsschnittstellen an die Cloud weiterzuleiten. Diese Variante ist interessant, da die Daten lokal auch für andere Zwecke verwendet werden können.

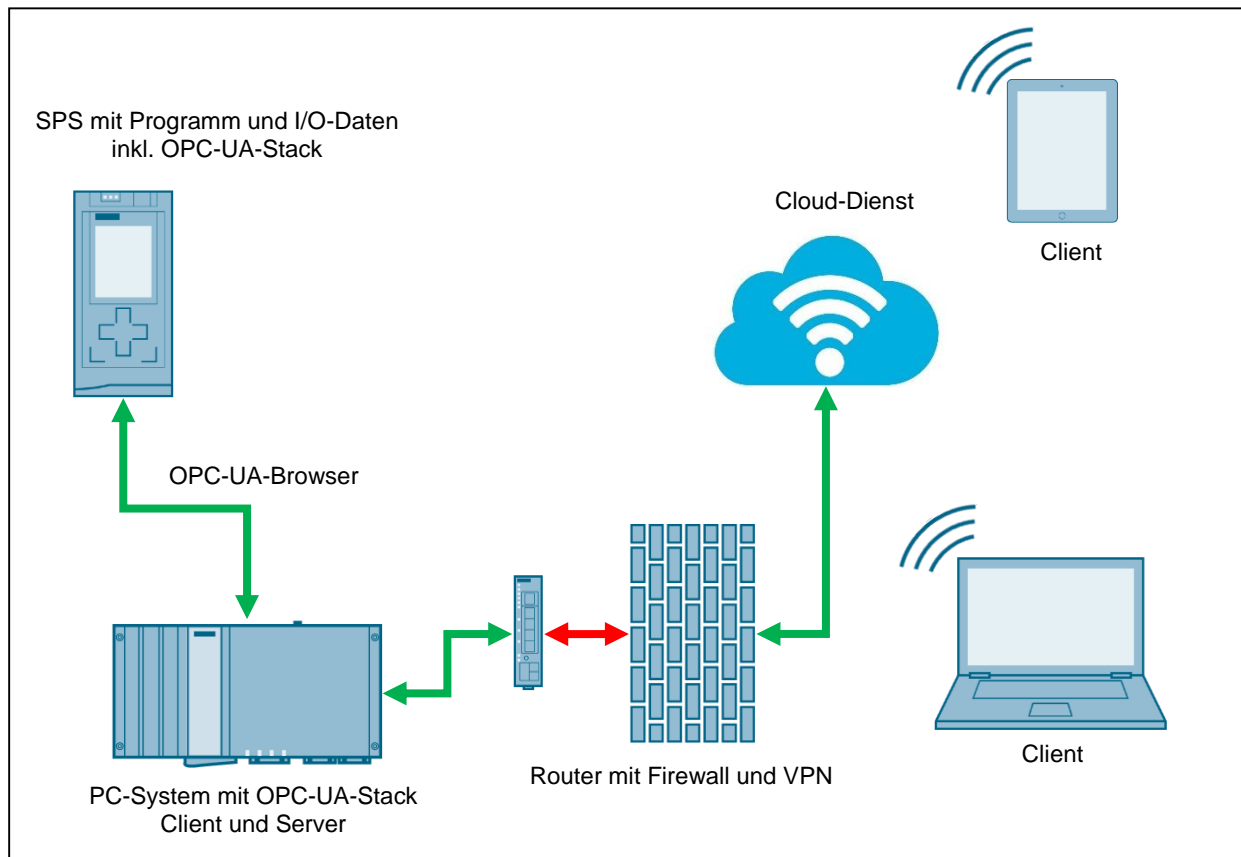


Abbildung 23: Variante 3 – Visuelle Darstellung, Quelle: Eigene Darstellung.

In Gesprächen mit der Firma Rosendahl Nextrom GmbH stellte sich die Variante 3 als die sinnvollste und praktikabelste Variante heraus. Hierbei wurden vor allem auf die Sicherheitsaspekte und Erweiterbarkeit der OPC-UA-Serververbindungen Wert gelegt. Zur Umsetzung dieser Variante sind nun mehrere Schritte notwendig, die nachfolgend aufgelistet werden.

- **Steuerungstechnik**
 1. Auswahl der Steuerungskomponenten (SPS) mit integrierter OPC-UA-Funktionalität
 2. Erstellung eines Strukturplans zur Erstellung eines OP UA Adressraums für das *ROEX*-Modul
 3. OPC-UA-Objekt Struktur implementieren
- **PC-System lokal**
 4. OPC-UA-Client Applikation in .NET umsetzen
 5. Kommunikationsschnittstelle für die Anbindung eines Cloud-Services implementieren
- **Cloud**
 6. Auswahl eines CloudProviders
 7. Konfiguration der Plattform
 8. Erstellen einer Datenbankapplikation bzw. Speicherapplikation
- **Maschinelles Lernen – Predictive Maintenance**
 9. Aufbereitung von historischen Daten
 10. Erstellen eines Lernalgorithmus zur Datenanalyse

5.2.1 Hardware

Aufgrund der verbauten Steuerung und der eingesetzten I/O Module wurde eine Umschlüsselung auf eine S7-1500 Steuerung durchgeführt. Dabei wurden die Leistungsdaten der bestehenden Hardware als Referenz für die neue Hardwareplattform verwendet. Nachfolgend wird eine tabellarische Gegenüberstellung der Baugruppen dargestellt. Die Datenblätter der Baugruppen können dem Anhang entnommen werden.

	Bisher eingesetzt		Neue Plattform	
	Bezeichnung	Artikelnummer	Bezeichnung	Artikelnummer
CPU	IM151-8 2PN/DP	6ES7151-8AB01-0AB0	1512SP-1PN	6ES7512-1DK01-0AB0
Digitaleingang	8DI DC24V	6ES7131-4BF00-0AA0	DI 8x24VDC ST	6ES7131-6BF00-0BA0
Analogeingang	2AI U ST	6ES7134-4FB01-0AB0	AI 4xU/I 2-wire ST	6ES7134-6HD00-0BA1
Analogausgang	2AO U ST	6ES7135-4FB01-0AB0	AQ 4xU/I ST	6ES7135-6HD00-0BA1
Digitalausgang	8DO DC24V/0,5A	6ES7132-4BF00-0AA0	DQ 8x24VDC/0.5A	6ES7132-6BF00-0BA0

Tabelle 2: Aufstellung der Hardware Umschlüsselung, Quelle: Eigene Darstellung.

5.2.2 Software

Durch den Tausch der Hardwareplattform von der Baureihe ET200S auf S7-1500 musste auch die Steuerungssoftware auf die neue Entwicklungsumgebung TIA Portal V14 migriert werden.



Abbildung 24: Migration der Software, Quelle: Eigene Darstellung.

Siemens bietet in der Entwicklungsumgebung *TIA Portal V14* eine Möglichkeit, Projekte die mit der älteren Entwicklungsumgebung *STEP7 Manager* umgesetzt wurden, zu migrieren. Diese Funktion wurde in diesem Fall auch verwendet. Jedoch muss im Vorfeld das alte Projekt aufbereitet werden, damit die Migration ohne Fehler abgeschlossen werden kann. Hierbei ist es wichtig erstmals die Programmstruktur zu analysieren, um zum Beispiel eine Löschung der Zugriffspunkte vorzunehmen, welche gar nicht mehr existieren. Auch wurden jegliche Netzwerkteilnehmer aus dem Projekt gelöscht. Diese Netzwerkteilnehmer (Profinet-I/O-Devices) können jederzeit wieder hinzugefügt werden.

Wie bereits erwähnt ist ein wichtiger Schritt zur Migration des Steuerungsprogramms die Analyse des Codes. Dies soll jedoch in dieser Arbeit nicht näher erläutert werden. Der Quellcode und die detaillierte Programmanalyse sind zwar Teil der Umsetzungsarbeit, jedoch für den Leser irrelevant.

Festzuhalten ist, dass das Steuerungsprogramm auf die aktuelle Siemens *Simatic S7-1500* Steuerungsfamilie migriert wird, um die Vorzüge des integrierten OPC-UA-Servers zu nutzen.

5.2.3 Netzwerkstruktur

Im nachfolgenden Bild (Abbildung 25) wird die Netzwerkstruktur des Extruders dargestellt. Dabei handelt es sich um das PROFINET-IO-System, welches die Feldebene der Maschine beschreibt. Es folgt eine Aufzählung der Komponenten:

- **ROEX38_00B**: CPU 1512SP-1 PN, Profinet-IO-Controller
IP-Adresse: 192.168.0.40
Subnetmask: 255.255.255.0
- **roex.h54.g120**: Siemens-Frequenzumrichter G120, Profinet-IO-Device
IP-Adresse: 192.168.0.100
Subnetmask: 255.255.255.0
- **roex.h54.etr132**: PSG-Temperaturregler für 132 Regelzonen, Profinet-IO-Device
IP-Adresse: 192.168.0.60
Subnetmask: 255.255.255.0

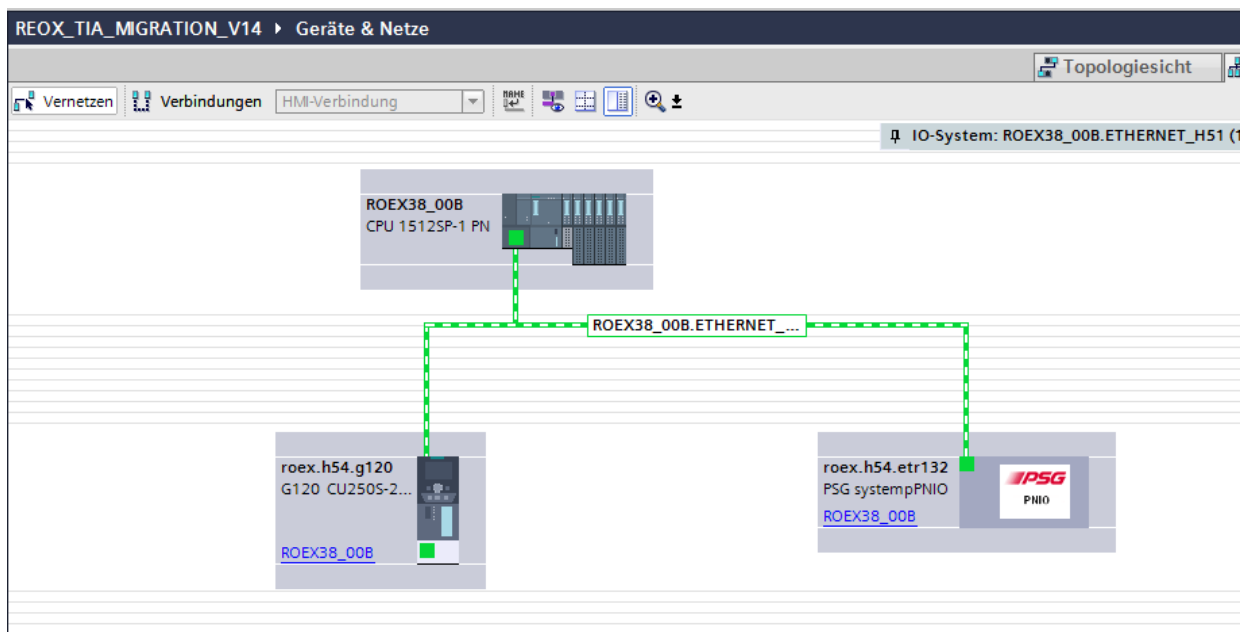


Abbildung 25: Netzwerksicht im TIA Portal V14, Quelle: Eigene Darstellung.

5.2.4 Definition von OPC UA-Objekten

Nach dem der Extruder auf die neue Zielhardware umgestellt und die Netzwerkstruktur festgelegt wird, müssen die Datenpunkte definiert werden, welche in der Cloud gespeichert werden sollen. Eine Aufgabenstellung kann dabei helfen, die notwendigen Daten und Objekte aus der Fertigungsanlage zu definieren.

Hierbei sind nicht nur Kenntnisse der Programmierung der Steuerung notwendig, viel wichtiger in diesem Zusammenhang ist das Wissen über die Fertigung und die Prozessabläufe des Extruders.

Aufgrund der Expertise im Hause Rosendahl Nextrom GmbH wurde schnell klar, dass generell Prozessdaten und Statusdaten zur Analyse herangezogen werden müssen.

- **Statusdaten**
 - Betriebsart – Hand-/Automatikbetrieb
 - Betriebszustand – Produktion, Leerlauf, Wartung
 - Kommunikationsstatus – Watchdog
 - Fehlerstatus – Sammelfehlermeldung

- **Prozessdaten**
 - Temperatur an der Schnecke (mehrere Messpunkte)
 - Druckwerte
 - Drehzahl der Schnecke (Frequenzumrichter)
 - Last der Schnecke (Frequenzumrichter)
 - Querschnitt des Produkts (Kabel)
 - Anlagengeschwindigkeit
 - Spulenzähler

5.2.5 Adressraum des OPC UA-Servers

Der Adressraum des OPC-UA-Servers wird automatisch durch das Entwicklungstool TIA Portal V14 erstellt. Einfluss kann über die Einstellungen in den Datenbausteinen genommen werden. Die Beschreibung dazu folgt im nächsten Unterkapitel.

5.2.6 OPC-UA-Server aktivieren

Nach der Migration, der Aufbereitung der Netzwerkstruktur durch Generieren eines Subsystems und der Topologie, kann nun der OPC-UA-Server der CPU aktiviert werden. In der nachfolgenden Abbildung 26 ist das Einstellungsmenü der Projektierungssoftware ersichtlich. In der Projektnavigation unter *Geräte & Netze* öffnet sich die Geräteansicht der Steuerung. Durch betätigen des Reiters *Eigenschaften*, öffnet sich im unteren Teil des Projektierungstools das Kontextmenü. Unter dem Menüpunkt *OPC UA* kann der Server nun aktiviert werden.

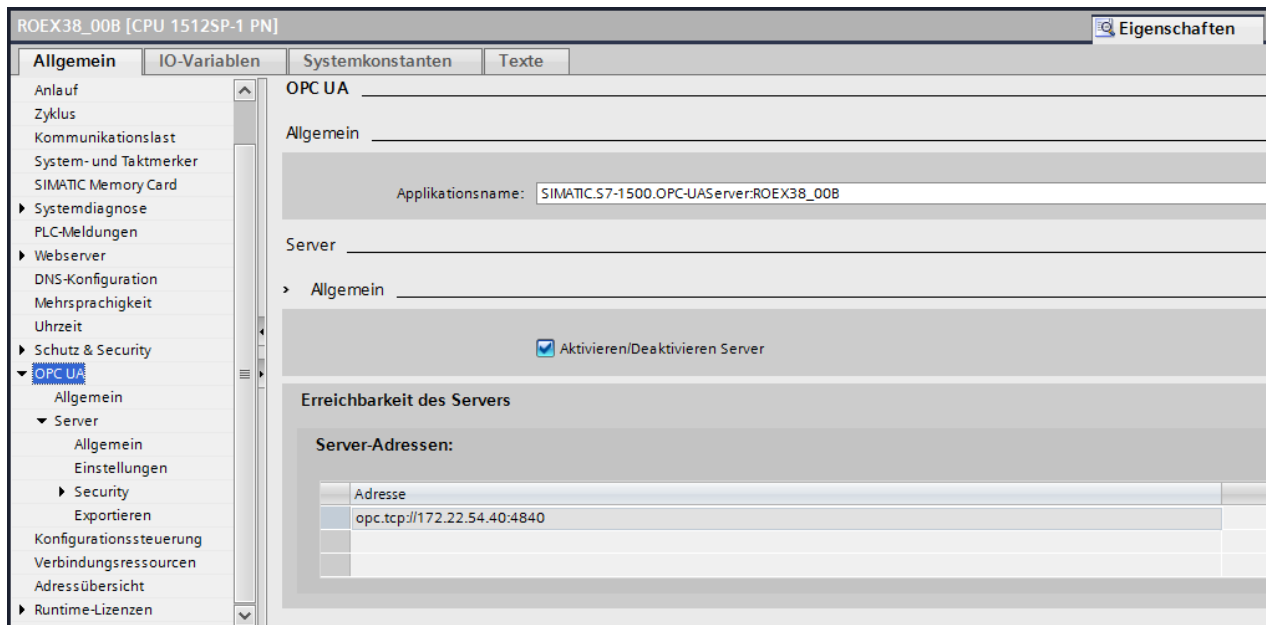


Abbildung 26: Aktivieren des OPC-UA-Servers, Quelle: Eigene Darstellung

In der Abbildung 26 ist auch die Server-URI des OPC-UA-Servers und der Applikationsname ersichtlich. Diese Information wird im weiteren Verlauf, vor allem für die Client Anbindung wichtig sein.

5.2.6.1 Einstellungen des OPC-UA-Servers

Nach der Aktivierung kann der OPC-UA-Server durch Laden des Programmes in die Steuerung bereits verwendet werden. Es sollten jedoch vorher noch einige Einstellungen vorgenommen werden, um den Server auf die Applikation abzustimmen.

Für die Einstellungen des OPC-UA-Servers gibt es die folgenden Parameter:

- TCP-Port
- Kleinstes Sendeintervall
- Kleinstes Abtastintervall

TCP-Port für OPC UA

Der OPC-UA-Standard definiert einen Default-TCP-Port 4840. Dieser wird von allen OPC-UA-Clients welche nach dem Standard zertifiziert sind unterstützt. Das bedeutet, dass bei der Verbindung eines Clients nur die Server-URI mitgeteilt werden muss (der Port kann entfallen). Es kann aber auch ein anderer Port gewählt werden, welcher sich im Nummernband zwischen 1024 und 49151 befindet. Bei der Wahl eines speziellen Ports muss dieser beim Verbindungsaufbau mitangegeben werden (z.B.: `opc.tcp://172.22.54.40:5180`). In diesem Projekt wurde der Standard-Port 4840 beibehalten.

Kleinstes Sendeintervall

Mit diesem Parameter kann der zeitliche Abstand eingestellt werden, mit der der Server eine *Notification* (Benachrichtigung) bei Werteänderung an den Client sendet. Dieser Parameter ist vor allem dann wichtig, wenn keine zyklische Wertabfrage zwischen Client und Server vereinbart ist, sondern eine *Subscription* (Abonnement). Der Server sendet dann nur bei Werteänderung den neuen Wert zu.

Wenn der OPC-UA-Client eine Aktualisierung alle 1000 ms fordert, sendet der Server dementsprechend nur jede Sekunde den neuen Wert. Falls der Server eine Aktualisierungszeit von 100 ms eingestellt hat, das kleinste Sendeintervall jedoch 200ms ist, sendet der Server auch nur alle 200 ms einen neuen Wert an den Client. Diese Einstellung kann somit die Netzwerkauslastung begrenzen.

Kleinstes Abtastintervall

Das kleinste Abtastintervall ist die Abtastrate der CPU, welche die Werte der CPU Variable erfasst und mit dem bisherigen Wert vergleicht. Das bedeutet bei einer Einstellung von 100 ms wird genau dieses Intervall eingehalten um die Aktualität des Wertes zu prüfen.

Subscription statt zyklisches Abfragen

Bei einigen Daten macht es Sinn, diese nicht zyklisch abzufragen (Polling), sondern nur bei einer Wertänderung. Hierfür wurde in der OPC-UA-Umsetzung die sogenannte *Subscription* eingeführt (Abonnement). Eine *Subscription* mit der gewünschten Variable meldet sich am Server an und informiert den Client über eine Wertänderung der Variable. Da ein Server meist sehr viele CPU-Variablen überwacht, informiert der Server mittels sogenannten Notifikationen den Client über die neuen Werte der Variablen.

5.2.6.2 Erweitern des Programms um die Datenschnittstelle IoT_Data

Im Extruder-SPS-Programm werden die Abläufe und Regelungen der Anlage gesteuert und ausgeführt. Eine effiziente Anbindung der notwendigen Daten ohne Eingriff in das bestehende Programm sollte bevorzugt eingesetzt werden. Um dies zu gewährleisten, wird ein Datenbaustein angelegt, welche das gesamt notwendige Datenpaket zusammenfasst. In diesem Datenbaustein wird auch das Leserecht des OPC-UA-Servers gesetzt (erreichbar aus HMI/OPC UA).

REOX_TIA_MIGRATION_V14 > ROEX38_00B [CPU 1512SP-1 PN] > Programmbausteine > IoT_Data [DB200]										
Aktualwerte behalten Momentaufnahme Momentaufnahmen in Startwerte kopieren Startwerte als Aktualwerte laden										
IoT_Data										
Name	Datentyp	Startwert	Remanenz	Erreichbar a..	Schrei...	Sichtbar i...	Einstellwert	Überwach..	Kommentar	
1	Static									
2	ActSpeed_Screw	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Actual Speed from FC
3	ActMeltPressure	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Actual Pressure Melting
4	ActMotorLoad	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Actual Motor Load from FC
5	Screw_ID	String	"	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		Identification Number Screw
6	Screw_RPM	Word	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		RPM from the Screw
7	OperatingHour	Date_And_Time	DT# 1990-01-01-0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Betriebsstunden
8	Temp_Melt	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature Melting
9	Temp_Cyl_1	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature
10	Temp_Cyl_2	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature
11	Temp_Cyl_3	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature
12	Temp_Cyl_4	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature
13	Temp_Cyl_5	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature
14	Temp_Flange	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature
15	Temp_Entry	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature
16	Temp_Head_1	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature
17	Temp_Head_2	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature
18	Temp_Head_3	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature
19	Temp_Head_4	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature
20	Temp_Head_5	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Temperature

Abbildung 27: Datenbaustein im SPS Programm zur Datenabholung durch den OPC UA Server, Quelle: Eigene Darstellung.

Die echten Messwerte werden dann mittels Funktion (FC) auf diesen Datenbaustein geschrieben. Damit ist gewährleistet, dass erstens kein Wert direkt über den OPC-UA-Server verändert werden kann und somit

einen Programmfehler hervorruft, und zweitens, dass die IoT-Schnittstelle getrennt vom eigentlichen SPS Programm beschrieben wird.

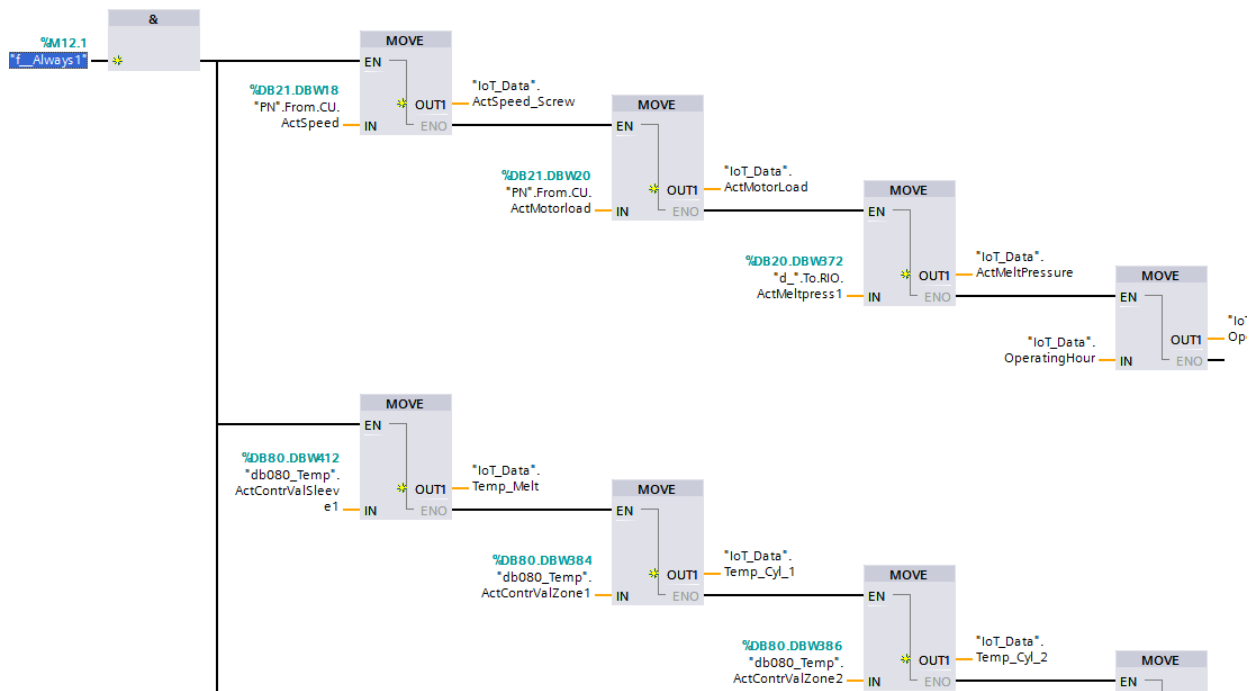


Abbildung 28: Daten in Datenbaustein IoT_Data kopieren, Quelle: Eigene Darstellung.

5.3 Cloud-Systeme – Auswahlkriterien

Im theoretischen Teil der Arbeit wurden die Vorzüge und Grundlagen von Cloud-Computing behandelt und aufgezeigt. Nun muss für die Datenanbindung des Extruders ein System gewählt werden, das die nötigen Voraussetzungen für die Umsetzung des Projektes erfüllt. Die Grundlage der Analyse sind technische und wirtschaftliche Aspekte.

Generell stellt sich die Frage welche Form der Cloud eingesetzt werden kann. Da die Firma *Rosendahl Nextrom* im Bereich *Cloud-Computing* bis dato, wenig Erfahrungen hat und derzeit nicht einsetzt, gibt es intern auch keine Infrastruktur die herangezogen werden könnte. Der Aufwand um eine *Private Cloud* bereitzustellen, übersteigt die zeitlichen Vorgaben die diesem Projekt zu Grunde liegen. Deshalb wird zur Umsetzung des Entwicklungsprojektes eine *Public Cloud* Lösung gewählt, da aufgrund der geringen Anzahl an Datenpunkten einer einzelnen Maschine das Risiko von öffentlich gespeicherten Daten seitens *Rosendahl Nextrom* als gering eingestuft wurde.

Bei der Auswahl des *Public-Cloud*-Anbieters waren mehrere grundsätzliche Voraussetzungen zu erfüllen welche nachfolgend erläutert werden.

- **Funktionen und Verfügbarkeit**

Voraussetzung ist, dass die Cloud Funktionen für alle im theoretischen Teil beschriebenen Leistungsebenen bereitstellt. Hierzu zählen Speicherkonten und eventuell virtuelle Maschinen (Infrastructure as a Service), Entwicklungstools zur Anbindung von Endgeräten auf OPC-UA-Basis (Software as a Service) und Tools zur Entwicklung von Big Data-Analysen (Platform as a Service).

- **Verfügbarkeit**
Die Verfügbarkeit des Dienstes war ausschlaggebend für die Wahl des Anbieters und sollte im obersten Bereich (> 99 %) liegen.
- **Serverstandort**
Die gespeicherten Daten sollen vorzugsweise auf mitteleuropäischen Servern gespeichert werden. Dann gilt allein europäisches Recht und der Schutz vor dem US-amerikanischen Patriot Act ist gewährleistet.
- **Kosten**
Der Cloud Dienstleister sollt ein sogenanntes Pay-per-Use Kostenmodell anbieten. Das bedeutet, dass nur dann Kosten anfallen, wenn ein Dienst verwendet wird (z.B. speichern in ein Speicherkonto).

Aus diesen Kriterien werden drei der namhaftesten Cloud-Dienstleister ausgewählt und zur genauen Analyse herangezogen.

Dazu zählen:

- Amazon Web-Services
- Google Cloud Platform
- Microsoft Azure

	Amazon web services	Google Cloud Platform	Microsoft Azure
Kosten	Pay-per-use	Pay-per-use	Pay-per-use
Serverstandort	Frankfurt/DE	Frankfurt/DE	Frankfurt/DE
Verfügbarkeit (Quelle:CloudHarmony)	99,9 %	99,9 %	99,9 %
OPC UA-Integration	Keine Information	Keine Information	.NET Core (SDK) und Beispiel Projekte (C#) frei verfügbar
Entwicklungstools Big Data/Machine Learning	vorhanden	vorhanden	vorhanden

Tabelle 3: Gegenüberstellung der Cloud-Servicedienstleister, Quelle: Eigene Darstellung.

Die Gegenüberstellung der drei Cloud-Servicedienstleister zeigt, dass nur eine Entscheidung aufgrund von bestehendem Know-How bzw. unterstützenden Softwarepaketen und Beschreibungen von Implementierung von OPC-UA-Kommunikation in die Cloud möglich ist. Die anderen Parameter decken sich fast zu 100 % und können so keine Entscheidungshilfe bieten. Da Microsoft und OPC UA eine Entwicklungspartnerschaft eingegangen sind und somit unterstützendes Material (SDK, Beschreibungen, Artikel, usw.) zum Thema zur Verfügung gestellt wird, wird das Cloud System *Azure* von Microsoft ausgewählt.

5.4 Microsoft Azure

Da die Datenbasis in der SPS nur vorbereitet wurde, muss nun der Cloud-Dienst eingerichtet werden. *Microsoft Azure* bietet mit deren Cloud basierten Diensten die Möglichkeit ein Netzwerk von Geräten zu bilden, die über das Internet an die *Azure Cloud* angebunden werden können. *Microsoft Azure* ermöglicht dies mit dem *IoT Hub* Dienst (Internet of Things Hub). Dieser Dienst muss erstellt und aktiviert werden. Hierfür muss ein bestehendes *Microsoft Azure* Konto vorhanden sein.

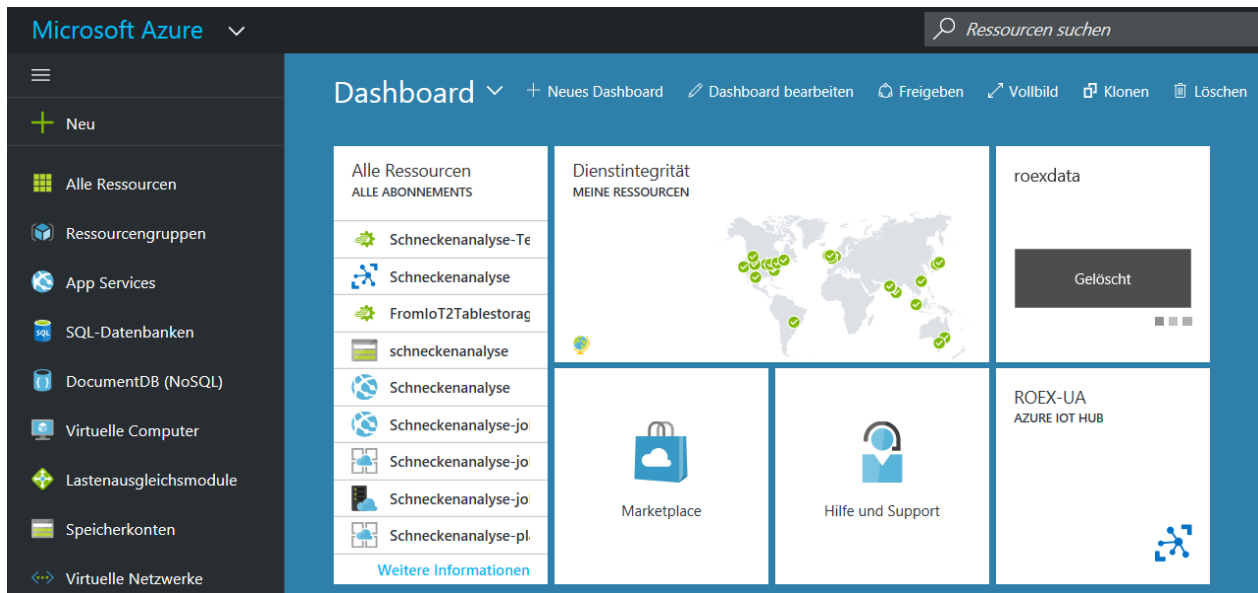


Abbildung 29: Microsoft Azure Dashboard, Quelle: Eigene Darstellung.

Um einen *IoT Hub* zu erstellen, muss dieser neue Dienst hinzugefügt werden. Der *IoT Hub* bildet die Grundlage für die Anbindungen der Geräte an die Cloud und bietet die Datenströme anderen Diensten in der Cloud an.

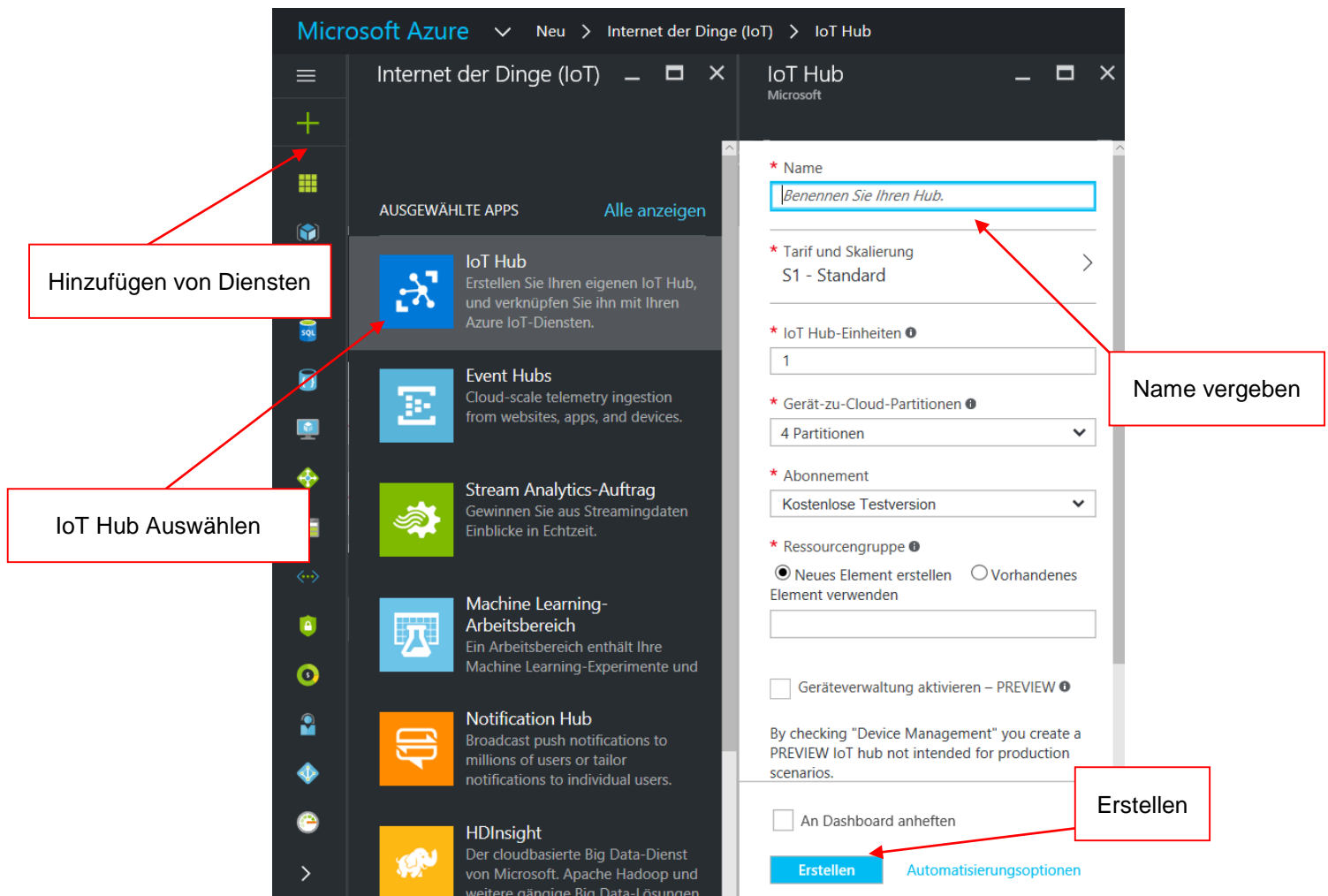


Abbildung 30: IoT Hub hinzufügen, Quelle: Eigene Darstellung.

Nach dem Erstellen des *IoT Hub* erscheint der Dienst im Dashboard des Accounts. Mit Klick auf das Symbol wird die Übersicht geöffnet.

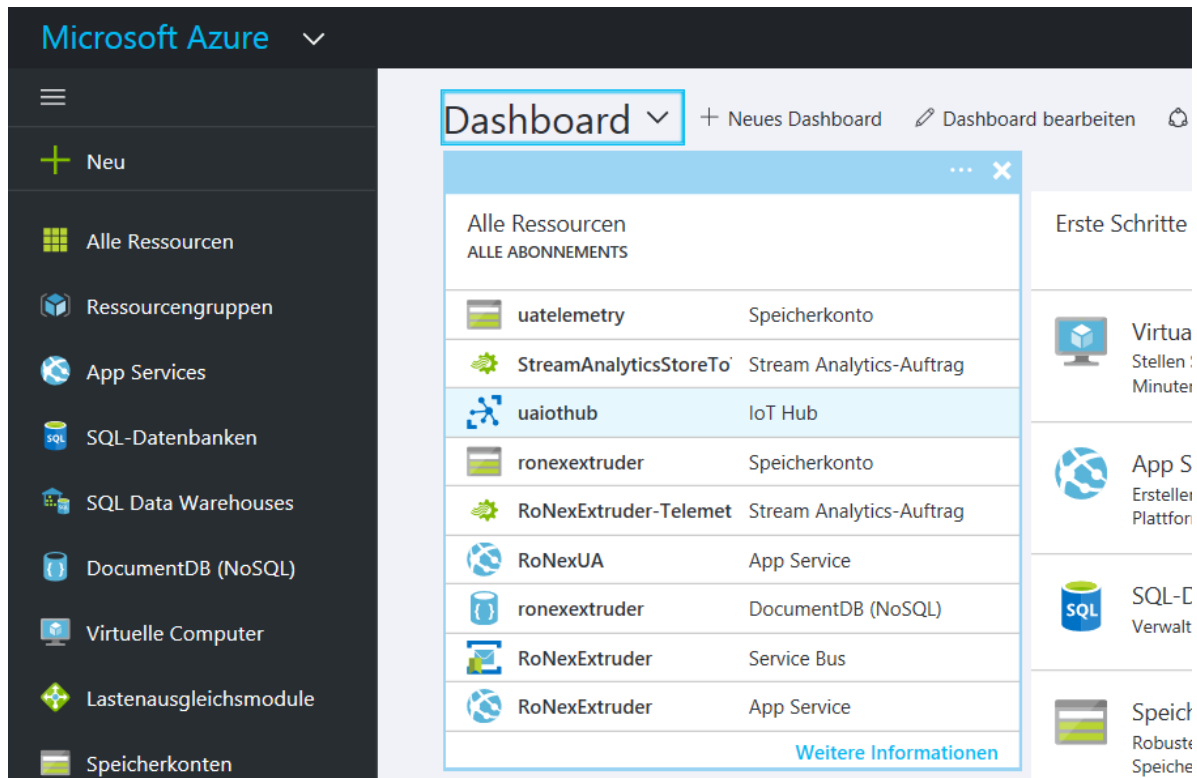


Abbildung 31: IoT Hub im Dashboard, Quelle: Eigene Darstellung.


Über das Dashboard (siehe Abbildung 31) kann in die Übersicht des IoT-Hubs gewechselt werden. In der Übersicht sieht man den Status des *IoT Hub* mit wichtigen Daten wie zum Beispiel dem Hostname, welcher für die Anbindung der Geräte wichtig ist. Auch kann man eine Übersicht der gesendeten Daten an den *IoT Hub* einblenden lassen. Über den Menüpunkt Geräte können die angebotenen Devices eingesehen und eventuell aktiviert/deaktiviert werden. Das bedeutet, dass dieses Gerät keine Daten mehr an die *IoT Hub* schicken kann bzw. die Daten nicht angenommen werden. Um Geräte überhaupt anbinden zu können, muss der Zugriffsschlüssel bekannt sein. Die Verbindung zwischen dem *IoT Hub* und den Geräten wird auf Gerätebasis verschlüsselt. Hierbei werden die Zugriffssteuerung und die Sicherheitsanmeldeinformationen verwendet. Dies wird mittels Security Token gewährleistet, welcher mittels *Shared Access Signature (SAS)* authentifiziert wird. Die SAS ist im *IoT Hub* hinterlegt und muss bekanntgegeben werden. In diesem Projekt wird die SPS, die ein Gerät darstellt, über ein Gateway angebunden. Die Verbindung zwischen der SPS und dem Gateway wird mittels OPC UA realisiert. Das Gateway entspricht im *IoT Hub* einem Gerät und setzt entsprechend die Daten von OPC UA um.


Name der Zugriffsrichtlinie
iothubowner


Berechtigungen


- Lesevorgänge in Registrierung ⓘ
- Schreibvorgänge in Registrierung ⓘ
- Dienstverbindung ⓘ
- Geräteverbindung ⓘ

Schlüssel für gemeinsamen Zugriff

Primärschlüssel ⓘ
KeXoRpt4EZoOss5/tlc3oggT6Nglom0RhE 

Sekundärschlüssel ⓘ
H8i2oyQmDXSzSveA4XP7Mdanpi+NKa9 

Verbindungszeichenfolge – Primärschlüssel ⓘ
HostName=uaiothub.azure-devices.net;S 

Verbindungszeichenfolge – Sekundärschlüssel ⓘ
HostName=uaiothub.azure-devices.net;S 

SAS-Token →

Abbildung 32: Zugriffskontrolle, Quelle: Eigene Darstellung

5.4.1 Device Explorer

Um ein Gerät anzulegen kann der *Device Explorer* verwendet werden. Der *Device Explorer* ist ein Programm, das Microsoft zur Datenanbindung von Geräten kostenfrei anbietet. Hier können eine Verbindung mit dem *IoT Hub* hergestellt werden und Geräte angelegt werden. Auch der Datenstrom kann beobachtet werden und der Device Explorer somit als Diagnose- und Inbetriebnahme-Tool eingesetzt werden.

Um eine Verbindung herzustellen muss der *Connection String* in das Eingabefeld kopiert werden:

```
HostName=uaiothub.azure-devices.net;SharedAccessKeyName=iiothubowner;
```

```
SharedAccessKey=H8i2oyQmDXSzSveA4XP7MdaNpi+NKa9fuljWyZwXRS4=
```

In der nachfolgenden Abbildung 33 wird dieser Vorgang dargestellt.

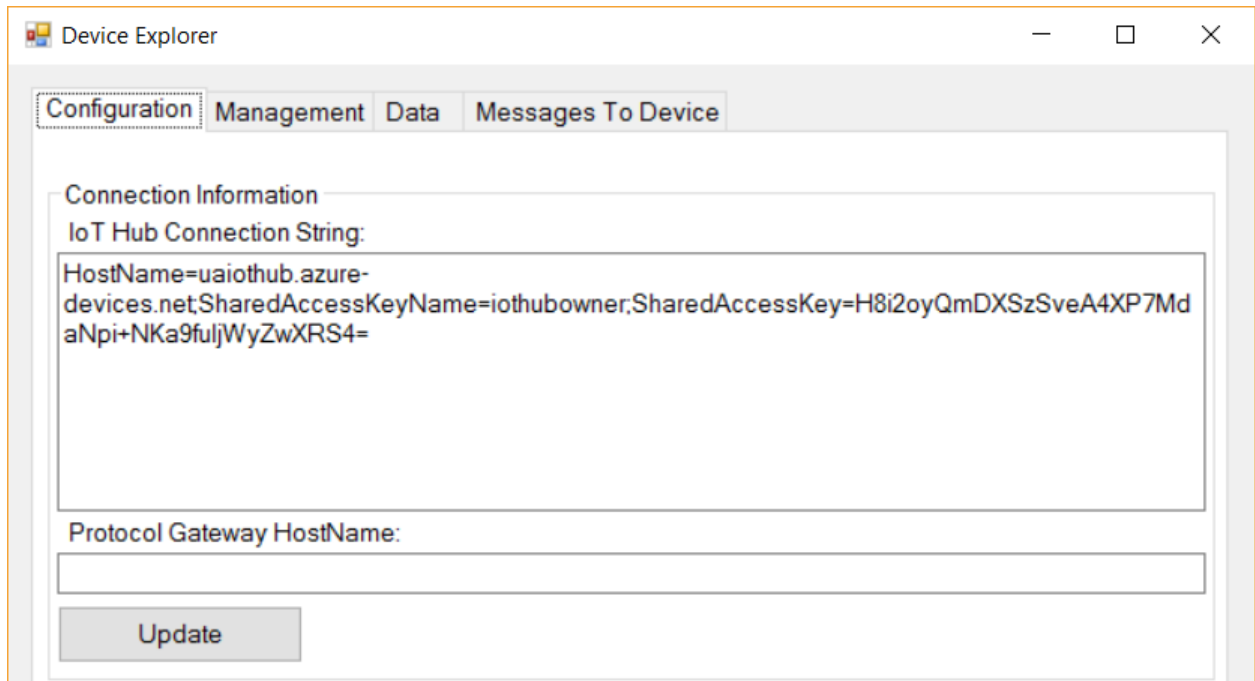


Abbildung 33: Device Explorer, Quelle: Eigene Darstellung.

Durch das Bestätigen über die Schaltfläche *Update* wird der *IoT Hub* angebunden. Über den Reiter *Management* können nun Geräte angelegt werden. Über den Button *Create* öffnet sich ein Kontextmenü, in dem der Name des Gerätes vergeben werden muss. Der Security Key wird standardmäßig automatisch generiert. Dieser Security Key wird in weiterer Folge in der Gateway Konfiguration benötigt. Das Gerät ist im Anschluss sofort im *IoT Hub* auffindbar (siehe Abbildung 35).

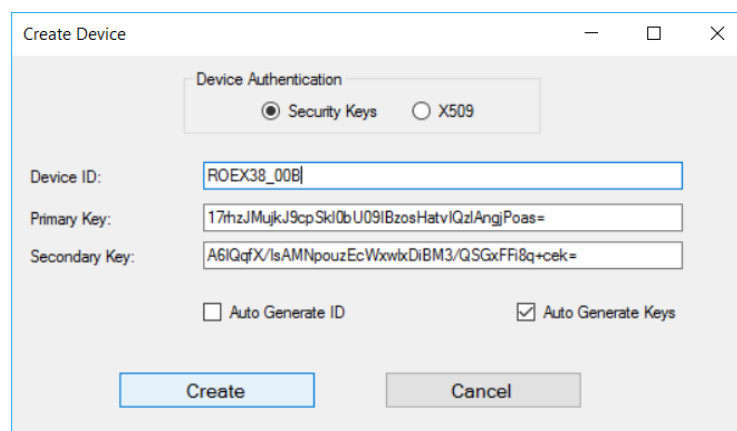
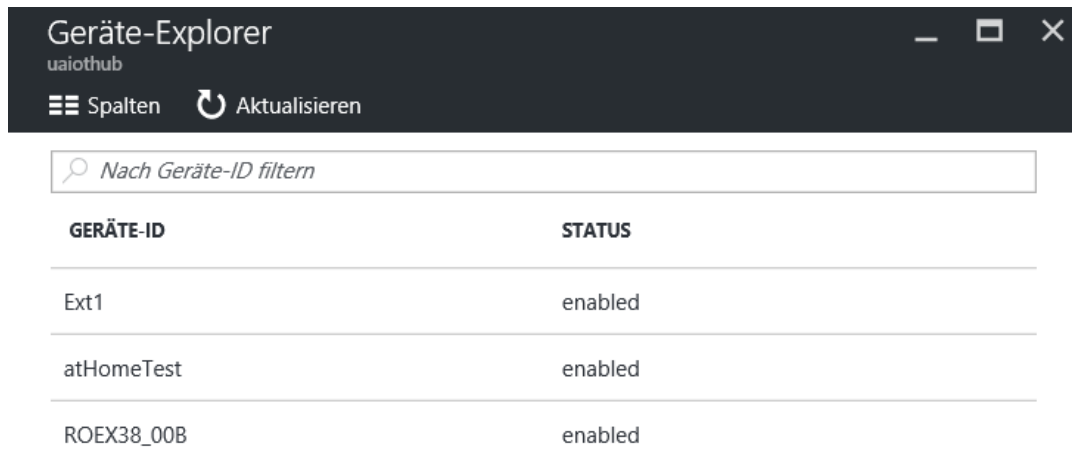


Abbildung 34: Create Device, Quelle: Eigene Darstellung.



GERÄTE-ID	STATUS
Ext1	enabled
atHomeTest	enabled
ROEX38_00B	enabled

Abbildung 35: IoT Hub Übersichtsseite, Quelle: Eigene Darstellung.

5.4.2 OPC UA Publisher-Gateway

Nach dem Anlegen des *IoT Hub*-Endpunktes und des Gerätes müssen nun die Daten von der SPS mittels Gateway angebunden werden. Hierfür muss ein OPC UA-Client implementiert werden, welcher die Daten auf dem Server liest und anschließend in das entsprechende Format (JSON) zur Übertragung in das *IoT Hub* umwandelt. Microsoft und die OPC Foundation haben hierzu in Kooperation ein Software Development Kit (SDK) veröffentlicht (UA NET Core), welcher Bausteine zur Implementierung auf .NET-Basis bietet. Die Verbindung zwischen dem Gateway und dem IoT Hub wird mittels *Advanced Message Queuing Protocol* (AMQP) realisiert. Das Paket wird als Referenz zur Umsetzung dieses Gateways herangezogen. Es wird die Windows-Forms Umsetzung *OPC UA Publisher* auf die Bedürfnisse des Projektes angepasst und entsprechend modifiziert.

Der erste Schritt ist es die Datenverbindung zwischen SPS und *IoT Hub* zu testen. Hierfür mussten einige Einstellungen in der Konfigurationsdatei des *OPC UA Publishers* angepasst werden. Das Programm ermöglicht es einem Benutzer, Abonnements in einer Verbindungs-Sitzung zu erstellen, um ein oder mehrere überwachte Elemente bzw. Variablen am OPC UA-Server (sprich auf der SPS) zu beobachten. Überwachte Elemente werden vom Server gelesen, in *JSON* codiert und an das *IoT Hub*, welches ein *AMQP*-Brokerendpunkt ist, veröffentlicht. Jede Endpunktverbindung wird als Teil der Konfiguration im .NET-Projekt unter *Opc.Ua.Publisher.config* hinterlegt. Nachfolgend werden die wichtigsten Konfigurationseinstellungen mit Ausschnitten aus dem Code erläutert.

Zunächst muss die Verbindung einen eindeutigen Namen erhalten. Deshalb sollte zum besseren Verständnis sowohl die Verbindung als auch der Endpunkt benannt werden. Weiters ist der Hostname des *AMQP*-Brokers einzutragen. In diesem Fall handelt es sich um einen *Azure IoT Hub-Broker*, welcher wie folgt aufgebaut ist: „Name des IoT Hub“.azure-device.net

```

385 <AmqpConnectionConfiguration>
386
387 <!-- A unique name for this connection -->
388 <Name>My OPC Publishing Endpoint</Name>
389
390 <!-- The broker's host name - In case of e.g. an Azure broker, the host is part of the
391 connection string provided by the Azure portal and prefixed with 'Host='. Otherwise
392 it is the server the broker runs on, such as localhost, etc. -->
393 <Host>uaiotHub.azure-devices.net</Host>

```

Abbildung 36: Name und Hostname des AMQP-Brokers (Azure IoT Hub) vergeben, Quelle: Eigene Darstellung.

Nachdem der Name und Host definiert wurden, muss nun der explizite Endpunkt der Verbindung bekannt gemacht werden. Der Endpunkt am *IoT Hub* ist das Gerät, das vorher mittels *Device Explorer* angelegt wurde (*ROEX38_00B* ist die Bezeichnung des Extruders im SPS Programm und wurde zur Benennung des Geräts herangezogen). Somit verbindet sich der *Publisher* auf den Endpunkt *ROEX38_00B* des Host *uaiotHub.azure-device.net* und sendet die Daten welche mittels *Subscription* (Abonnement) vom OPC UA-Server angefordert werden.

```

399 <!-- The publishing endpoint on the broker host -
400 Use the url that follows the host name starting with a '/'
401 In case of an Azure broker, the endpoint is part of the connection string.
402 To configure the connection to use Azure IoT Hub device to cloud endpoint use
403 /devices/{DeviceId}/messages/events
404 and substitute DeviceId with the device id in the connections string (prefix 'DeviceId=')
405 To configure the connection to use Azure Service bus use the Entity value in the
406 connection string (prefix 'Entity=') and prefix with '/'. For example
407 /{Entity}
408 -->
409 <Endpoint>/devices/ROEX38_00B/messages/events</Endpoint>
410

```

Abbildung 37: Endpunkt definieren, Quelle: Eigene Darstellung.

Zur sicheren Verbindung zwischen *IoT Hub* und *Publisher* wird eine authentifizierte Verbindung erstellt. Dazu wird der *Shared Access Key* des angelegten Geräts benötigt, der beim Hinzufügen des Geräts vergeben wurde und nur dem Besitzer bekannt ist. Dieser Key wird im Projekt fix hinterlegt.

```

416 <!-- User name and key/password to secure connection.
417 In case of e.g. an Azure broker, the key and principal name are part of
418 the connection string (prefix 'SharedAccessKeyName=' and 'SharedAccessKey').
419 In the case of iot hub's device connection string, leave the key name blank and
420 only provide the shared access key value from the connection string.
421 For unauthenticated connections leave both blank
422 -->
423 <KeyName></KeyName>
424 <KeyValue>YrPnn12Xkc1cmoRi/V028V1H0vt7Bj40tPeFQ00Lkb4=</KeyValue>
425 <!-- Specifies the encoding of the key. The default encoding is utf-8, however,
426 Azure IoT Hub Connection strings are base 64 encoded, this setting is thus for
427 convinience -->
428 <KeyEncoding>Base64</KeyEncoding>
429

```

Abbildung 38: Shared Access Key des Device, Quelle: Eigene Darstellung.

Zur Absicherung der Verbindung (amqp link) muss sich die Applikation (*OPC UA Publisher*) am *IoT Hub* authentifizieren. Dies wiederum wird mittels claim-based-security Token realisiert. Die Azure Cloud verwaltet Benutzer und Services, welche an zentraler Stelle in der Cloud zusammengefasst werden. Die zentrale Stelle stellt Security Tokens aus. Hierbei handelt es sich um eine *Brokered-Trust* Methode, welche auch *OPC UA* verwendet, um sichere *Sessions* zu erstellen. Dieser Token beinhaltet den Typ und den

Bereich, für den der Token Gültigkeit besitzt. Der *OPC UA Publisher* fragt einen Secure-Token (SAS-Token) für den Zugang zum Gerät über den *servicebus.windows.net:sastoken* an und sendet ihn anschließend zur Authentifizierung dem IoT Hub zu. Der IoT Hub Service prüft den Token und vergibt die Berechtigung an dieses Gerät, Daten zu übermitteln.⁵⁰

```
430 <!-- Set to true to use claims based security (cbs) to secure the amqp link. This
431      requires token type and the scope of the token to be configured as well (see below).
432      This should be set to true for all brokers supporting CBS, including all Azure brokers -->
433 <UseCbs>true</UseCbs>
434 <!-- Token type for Azure brokers is servicebus.windows.net:sastoken -->
435 <TokenType>servicebus.windows.net:sastoken</TokenType>
436 <!-- Token scope for the cbs token.
437      For Azure Servicebus endpoints, use
438      http://{Host}/{Entity}
439      For Azure IoT Hub endpoints, use
440      {Host}/devices/{DeviceId}
441      For all other endpoints, use
442      {Host}/{Endpoint}
443      Host, DeviceId, and Entity name are part of the Azure portal supplied
444      connection string (see publishing endpoint section) -->
445 <TokenScope>uiothub.azure-devices.net/devices/Ext1</TokenScope>
446 <!-- Specifies the token lifetime in milliseconds, defaults to 60 seconds -->
447 <!-- <TokenLifetime>60000</TokenLifetime> -->
448
```

Abbildung 39: Tokenscope für claim based Security einstellen, Quelle: Eigene Darstellung.

Nachdem diese Einstellungen vorgenommen wurden muss das Projekt neu erstellt werden. Durch das Öffnen der Windows-Forms-Anwendung *Opc.Ua.Publisher.exe* erscheint die Publisher-Umgebung. Über die Adresszeile welche schon im Projekt voreingestellt wurde, kann ein entsprechender OPC UA-Server eingetragen werden. Die IP-Adresse des OPC UA-Servers auf der Steuerung wurde aus Testzwecken auf 192.168.0.40 geändert. Der Port des Servers lautet 4840 und ist somit als Standard-Port definiert, der nicht angeführt werden muss. Mit dem Bestätigen der Schaltfläche *Connect* öffnet sich ein neues Fenster, das sogenannte *Server Configuration* Fenster. In diesem Fenster werden generelle Eigenschaften des Servers angezeigt, wie zum Beispiel die *Application URI* oder der *Protocol Typ*. Auf der rechten Seite findet man ein Auswahl-Fenster. In diesem Bereich werden die verschiedenen Sicherheitsmerkmale, die der Server unterstützt, angezeigt. In diesem Beispiel wird die Variante *SignAndEncrypt – Basic128Rsa15* verwendet.

⁵⁰ Vgl. Kyle (2012), Online-Quelle [22.09.2016].

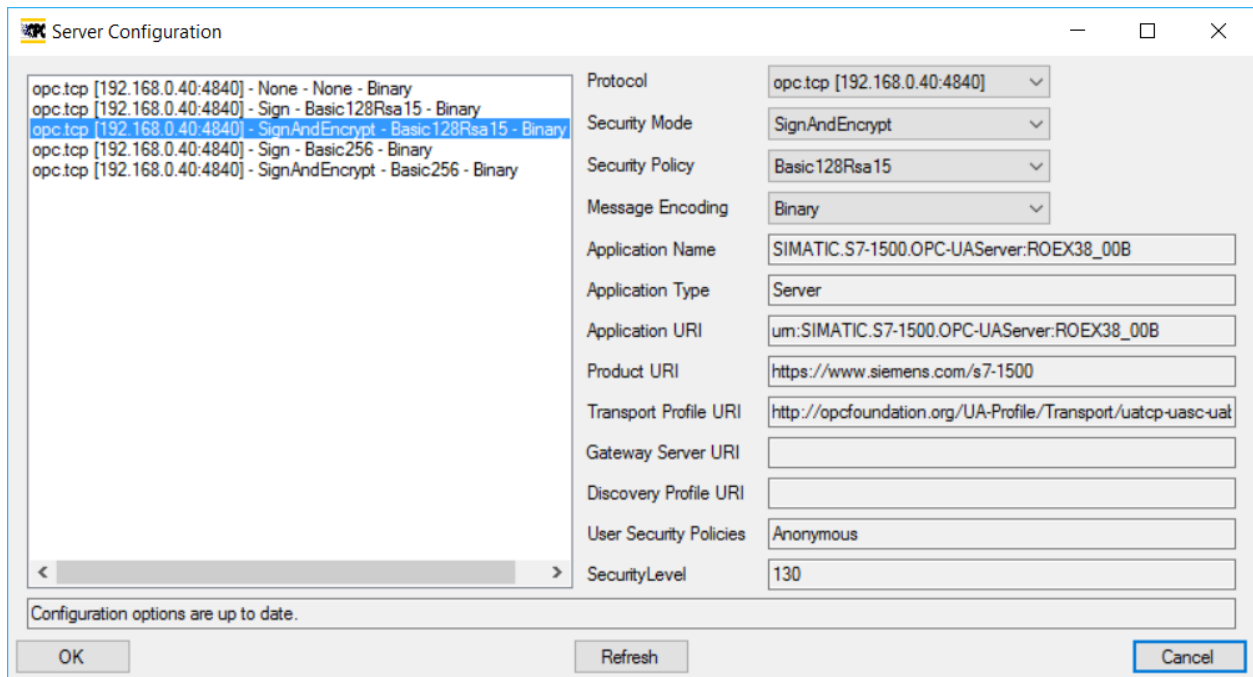


Abbildung 40: Server Configuration, Quelle: Eigene Darstellung.

Auch wenn sich das gesamte System (SPS und Publisher), in einem privaten Netzwerk befindet, sollte eine signierte und verschlüsselte Kommunikation eingerichtet werden. Wenn jemand Zugang über einen anderen Punkt im Netzwerk erhält, kann trotzdem ein Mitlesen bzw. Manipulieren von Nachrichten verhindert bzw. erschwert werden (mehrschichtiges Security-Konzept). Mittels der Schaltfläche **OK** kann eine Verbindung aufgebaut werden. Es öffnet sich ein weiteres Fenster, in dem der Name der Session und die Authentifikationsart ausgewählt werden können.

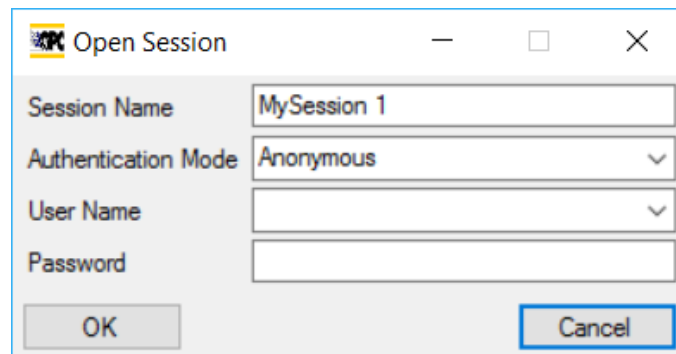


Abbildung 41: Open Session, Quelle: Eigene Darstellung.

Der Publisher repräsentiert einen OPC-UA-Client, welcher auf den OPC-UA-Server auf der Steuerung zugreifen möchte. Es wäre möglich, dass der Server nur Clients akzeptiert welche sich mittels Benutzername und Passwort authentifizieren können. In diesem Beispiel wurde das Schreiben von Variablen generell untersagt, weshalb eine Benutzerauthentifizierung im ersten Schritt nicht notwendig ist. Der Server akzeptiert aufgrund seiner Einstellung somit auch anonyme Clients. Anschließend wird im Hauptfenster die Verbindung und die Objektstruktur (Adressraum) des OPC-UA-Servers angezeigt. Hier wird sofort ersichtlich, dass es sich um die SPS der Anlage handelt (ROEX38_00B). Wie im Bild ersichtlich, kann nur auf den Bereich der globalen Datenbausteine zugegriffen werden (*DataBlocksGlobal*) und auf

den im SPS-Programm vorbereiteten Datenbaustein *IoT_Data*. Der nächste Schritt ist nun die gewünschten Variablen mittels *Subscription* an die Cloud weiterzuleiten.

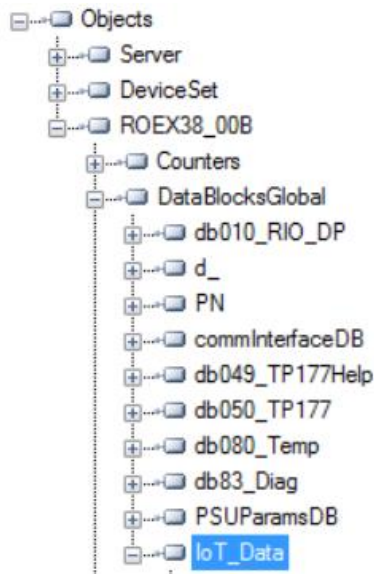


Abbildung 42: OPC UA – Objekte, Quelle: Eigene Darstellung.

Mittels Rechts-Klick auf die gewünschte Variable kann im Kontextmenü über *Subscription/New* ein neues Abonnement eingerichtet werden.

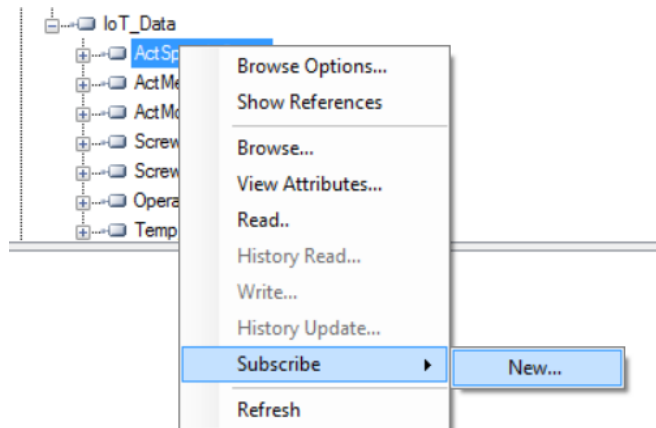


Abbildung 43: Erstellen einer Subscription, Eigene Darstellung.

Im darauffolgenden Fenster können die Eigenschaften des Abonnements eingestellt werden. Diese Einstellungen sollten sehr genau betrachtet werden um vor allem die Netzwerkauslastung im Vorhinein zu begrenzen.

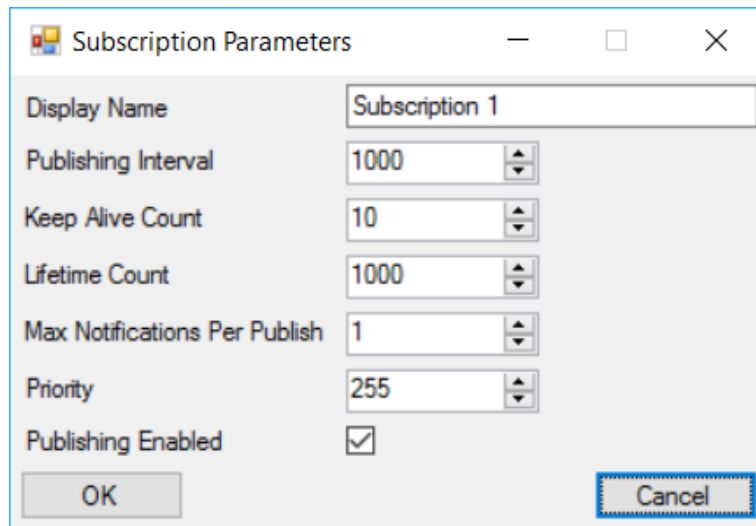


Abbildung 44: Subscription Parameter, Quelle: Eigene Darstellung.

Nachfolgend werden die wichtigsten Einstellungsmöglichkeiten genauer erklärt.

Display Name	Name des <i>Subscription</i>
Publishing Interval	Definiert, wie oft der Server prüft, ob Benachrichtigungspakete (<i>Notifications</i>) für ein Abonnement an den Client gesendet werden. Ein typischer Wert ist 1000 ms. Es sollte nicht -1 oder 0 sein. Der Server kann das <i>Publishing Intervall</i> auf ein Standardminimum erhöhen. In diesem Projekt wurde ein kleinstes Sendeintervall von 200 ms eingestellt.
Keep Alive Count	Wenn es keine neuen Daten zum Senden nach dem nächsten <i>Publishing Intervall</i> gibt, sendet der Server grundsätzlich keine <i>Notification</i> an den Client. Der <i>Keep Alive Counter</i> gibt an, wie oft der Server diese <i>Notification</i> überspringen darf, bis trotzdem eine <i>Notification</i> gesendet wird. Dies dient dem Client als Hinweis, dass die <i>Subscription</i> noch am Leben ist und dass es gerade keine neuen Daten gibt.
Lifetime Count	Der Client ist dafür verantwortlich, <i>Publish Requests</i> an den Server zu senden, damit der Server <i>Publish Responses</i> zurückschicken kann. Die <i>Publish Responses</i> werden verwendet, um die Benachrichtigungen zu übermitteln. Der Server überprüft auch, ob der Client erreichbar ist, indem er erkennt, ob neue <i>Publish Requests</i> empfangen werden. <i>LifeTime Count</i> definiert die Anzahl der <i>Publishing Intervalle</i> , die auf eine neue <i>Publish Request</i> warten, bevor der Client nicht mehr aktiv ist. Das Abonnement wird dann vom Server entfernt.
Max Notifications per Publish	<i>Max Notifications per Publish</i> bestimmt, wie viele Anfragen an den Server gesendet werden.

Tabelle 4: Subscription Parameterbeschreibung, Quelle: Eigene Darstellung.

Nach dem Bestätigen der Einstellungen über die Schaltfläche *OK* werden die gewünschten Variablen bzw. Nodes des OPC UA Servers beobachtet und entsprechend bei Wertänderung oder nach Ablauf des *Keep Alive Count* eine *Notificaton* an das *IoT Hub* gesendet.

The screenshot shows a window titled 'Subscription 1 - Data Changes' with a menu bar containing 'Subscription', 'Window', and 'Conditions'. Below the menu bar is a table with the following columns: ID, Name, Node ID, Mode, Sampling Interval, Revised Sampling Interval, Queue Size, and Revised Queue Size. The table contains five rows of data. Below this table is another table with columns: Item, Variable, Value, Status, Source Time, and Server Time. This second table contains five rows of data corresponding to the variables in the first table.

ID	Name	Node ID	Mode	Sampling Interval	Revised Sampling Interval	Queue Size	Revised Queue Size
1	ActSpeed_Screw	ns=3;s="IoT_Data"."ActSpeed_Screw"	Reporting	0	100	1	1
2	ActMeltPressure	ns=3;s="IoT_Data"."ActMeltPressure"	Reporting	0	100	1	1
3	ActMotorLoad	ns=3;s="IoT_Data"."ActMotorLoad"	Reporting	0	100	1	1
4	Screw_ID	ns=3;s="IoT_Data"."Screw_ID"	Reporting	0	100	1	1
5	Screw_RPM	ns=3;s="IoT_Data"."Screw_RPM"	Reporting	0	100	1	1

Item	Variable	Value	Status	Source Time	Server Time
[4]	ActSpeed_Screw	0	Good	09:07:44.651	09:07:44.650
[5]	ActMeltPressure	0	Good	09:08:01.354	09:08:01.353
[6]	ActMotorLoad	0	Good	09:08:03.731	09:08:03.731
[7]	Screw_ID	0	Good	09:08:06.531	09:08:06.531
[8]	Screw_RPM	0	Good	09:08:16.430	09:08:16.430

Abbildung 45: Übersicht der angebotenen Variablen, Quelle: Eigene Darstellung.

Die Daten werden nun an das *Azure IoT Hub* gesendet. In der Cloud muss der Datenstrom auf ein sogenanntes Speicherkonto umgeleitet werden, damit die Daten nicht verloren gehen. *Microsoft Azure* bietet hier eine umfangreiche Speicherdienstfunktion.

5.4.3 Azure IoT – Streamanalytics Funktion

Der Datenstrom des *IoT Hub* muss in einen Datenspeicher umgeleitet werden. In der *Microsoft Azure Cloud* wird das mittels dem sogenannten *Stream Analytics* Dienst realisiert. Dem Dienst müssen die Quelle und das Ziel der Daten bekannt gemacht werden. Dies wird in einem Einstellungsmenü durchgeführt. Wichtig dabei ist, das richtige Datenformat und die Codierung der Daten einzustellen, damit auch der Dateninhalt übertragen werden kann. Der Publisher sendet die Daten im *JSON* Format in *UTF-8* Formatierung an den *IoT Hub*. In Abbildung 46 wird das Einstellungsmenü dargestellt.

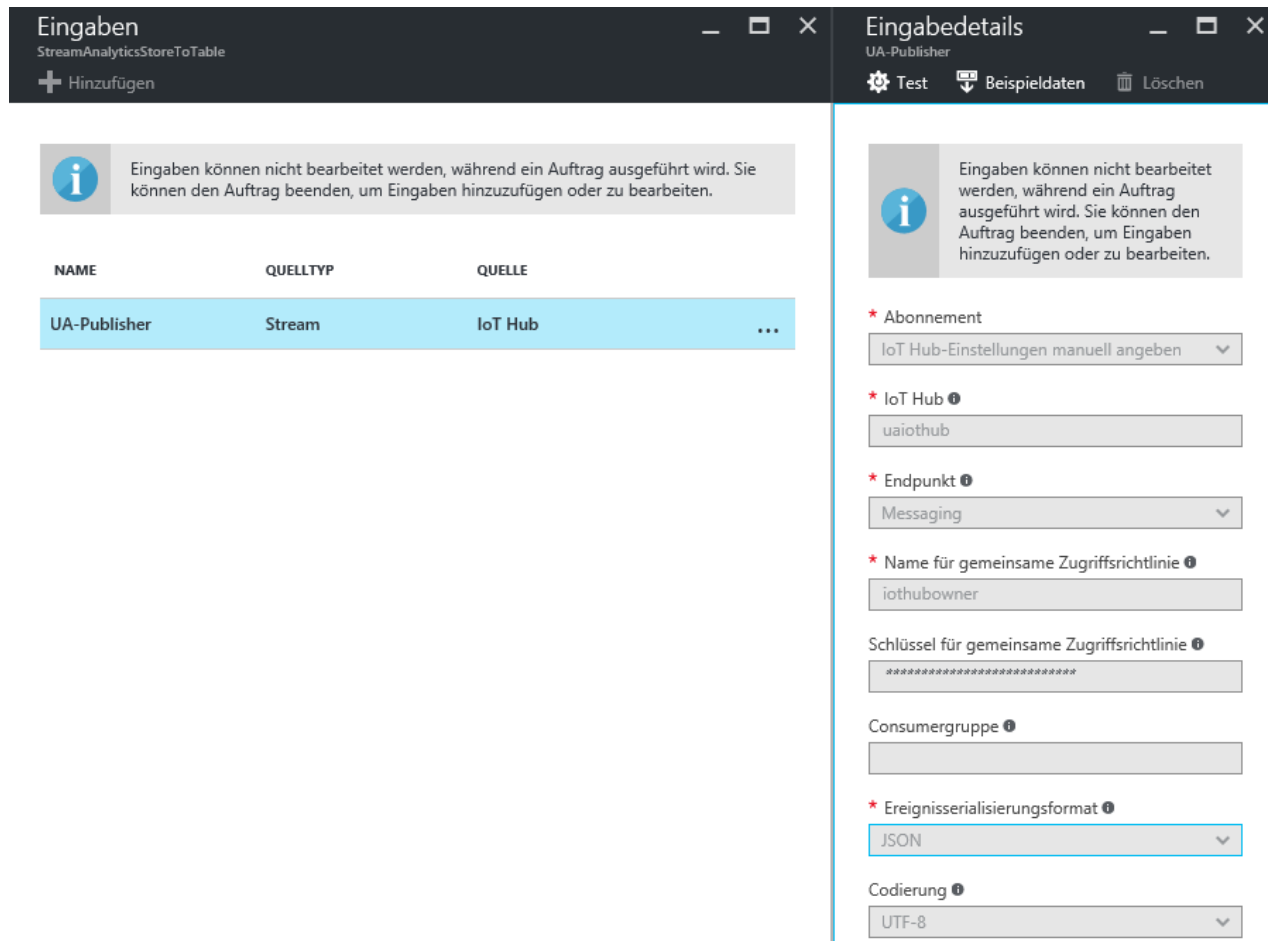


Abbildung 46: Stream Analytics Einstellungen, Quelle: Eigene Darstellung.

- **JSON**

JavaScript Object Notation (JSON) ist ein leichtes, textbasiertes, sprachunabhängiges Datenaustauschformat. JSON hat sich als gängiges Notationsformat zur Übertragung und zum Speichern von strukturierten Daten durchgesetzt. Meist wird es bei Webanwendungen in Verbindung mit JavaScript oder WebSockets zum Datentransfer verwendet. Im Gegensatz zur Extensible Markup Language (XML) ist JSON keine Struktur-beschreibende Sprache, welche sie sehr schlank und effizient macht.⁵¹

- **UTF-8**

UTF-8 ist die am weitesten verbreitete Kodierung für Unicode-Zeichen. Die Internet Engineering Task Force verlangt von allen neuen Internetkommunikationsprotokollen, dass die Zeichenkodierung deklariert wird und dass UTF-8 eine der unterstützten Kodierungen ist. Auch bei der in Webbrowsern angewendeten Auszeichnungssprache HTML setzt sich UTF-8 zur Darstellung sprachspezifischer Zeichen zunehmend durch.⁵²

⁵¹ Vgl. Network Working Group (Hrsg.) (2006).

⁵² Vgl. W3Techs on (2016), Online-Quelle [28.10.2016].

Der *Stream Analytics* Dienst wird nach der Einstellung der Eingangsquelle (*IoT Hub* Datenstrom) und dem Ausgangsziel (BLOB- und Tabellenspeicher) mittels ausführbarem Skript programmiert.

In diesem Skript können Vorauswahlen bzw. auch Berechnungen durchgeführt werden. In diesem Fall werden alle ankommenden Daten in den BLOB-Speicher geschrieben. Der Tabellenspeicher wird nur mit definierten Datenbereichen befüllt.

```
1 -- Table output
2 SELECT
3     MonitoredItem.id, Value.Value, Value.SourceTimestamp, MonitoredItem.Uri, IoTHub.ConnectionDeviceId, IoTHub.ConnectionDeviceGenerationId
4 INTO
5     [OutputToTable]
6 FROM
7     [UA-Publisher]
8
9
10 -- Blob output, debugging
11 SELECT
12     *
13 INTO
14     [BlobOutput]
15 FROM
16     [UA-Publisher]
```

Abbildung 47: Ausführungsskript zur Datenüberführung in den Speicher der Azure Cloud, Quelle: Eigene Darstellung.

Die einzelnen Datenbereiche (z.B. Value) in der Notification des OPC-UA-Publishers können somit einzeln angewählt und abgespeichert werden. Nachfolgend ist eine solche Notification im *JSON* Format ersichtlich:

```
{
  "MonitoredItem":{
    "Id":"s=IoT_Data.ActMeltPressure",
    "Uri":"http://www.siemens.com/simatic-s7-opcua"
  },
  "ClientHandle":5,
  "Value":{
    "Value":30,
    "SourceTimestamp":"2016-10-20T1"
  }
}
```

Abbildung 48: Notification im JSON Format, Quelle: Eigene Darstellung.

- **BLOB-Speicher**

Der BLOB-Speicher wurde implementiert um generell den Datenstrom und den Speichervorgang zu testen. Der BLOB-Speicher ist ein Dienst zur Speicherung großer Mengen von unstrukturierten Objektdaten, beispielsweise Text- oder Binärdaten. BLOB ist die Abkürzung für Binary Large Objects, was bedeutet, dass es sich dabei um binäre Objekte handelt. Der BLOB, der durch das Streamen der Daten vom Publisher entsteht, ist eine .JSON Datei welche die Informationen zeilenbasiert abspeichert.

```
{
  "MonitoredItem": {
    "Id": "s=IoT_Data.ActSpeed_Screw",
    "Uri": "http://www.siemens.com/simatic-s7-opcua",
    "ClientHandle": 14,
    "Value": 900
  },
  "MonitoredItem": {
    "Id": "s=IoT_Data.ActMeltPressure",
    "Uri": "http://www.siemens.com/simatic-s7-opcua",
    "ClientHandle": 15,
    "Value": 100
  },
  "MonitoredItem": {
    "Id": "s=IoT_Data.ActMotorLoad",
    "Uri": "http://www.siemens.com/simatic-s7-opcua",
    "ClientHandle": 16,
    "Value": 50
  },
  "MonitoredItem": {
    "Id": "s=IoT_Data.Screw ID",
    "Uri": "http://www.siemens.com/simatic-s7-opcua",
    "ClientHandle": 17,
    "Value": "IDderSchnecke3"
  },
  "MonitoredItem": {
    "Id": "s=IoT_Data.Screw RPM",
    "Uri": "http://www.siemens.com/simatic-s7-opcua",
    "ClientHandle": 18,
    "Value": 4369
  },
  "MonitoredItem": {
    "Id": "s=IoT_Data.ActSpeed_Screw",
    "Uri": "http://www.siemens.com/simatic-s7-opcua",
    "ClientHandle": 14,
    "Value": 500
  },
  "MonitoredItem": {
    "Id": "s=IoT_Data.ActMeltPressure",
    "Uri": "http://www.siemens.com/simatic-s7-opcua",
    "ClientHandle": 15,
    "Value": 0
  },
  "MonitoredItem": {
    "Id": "s=IoT_Data.ActMotorLoad",
    "Uri": "http://www.siemens.com/simatic-s7-opcua",
    "ClientHandle": 16,
    "Value": 0
  },
  "MonitoredItem": {
    "Id": "s=IoT_Data.Screw ID",
    "Uri": "http://www.siemens.com/simatic-s7-opcua",
    "ClientHandle": 17,
    "Value": "IDderSchnecke2"
  }
}
```

Abbildung 49: BLOB-Speicher Einträge, Quelle: Eigene Darstellung.

In der Abbildung 49 kann man erkennen, dass jeder neue Datensatz, welcher durch eine Notification übertragen wird, einen neuen Zeileneintrag hervorruft. Somit wächst der BLOB-Speicher unstrukturiert mit jeder Notification an.

- **Tabellenspeicher**

Um einen strukturierten Zugriff auf die Daten zu erhalten, wird ein weiteres Speicherverfahren eingesetzt, der sogenannte Tabellenspeicher. Der Tabellenspeicherdienst dient der Speicherung von sogenannten nicht relationalen, strukturierten Daten. Das bedeutet, dass die Daten keine Beziehungen zueinander aufweisen, wie es in Datenbanksystemen, zum Beispiel SQL-Datenbanken, der Fall ist. Aus diesem Grund wird auch von sogenannten *NoSQL*-Datenspeichern gesprochen. In der nachfolgenden Abbildung 50 kann man die entsprechenden Dateneinträge sehen, welche von der SPS über den Publisher an das *IoT Hub* gesendet wurden. Die Daten sind durch die entsprechenden Dateneinträge und den Zeilenindex strukturiert abgespeichert und dienen als Grundlage für analytische Verfahren.

PartitionKey	Timestamp	value	connectiondeviceid	id	sourcetimestamp
s=IoT_Data.Screw_RPM	2016-10-27T10:33:01.907Z	4369	ROEX38_00B	s=IoT_Data.Screw_RPM	2016-10-27T10:32:48.912Z
s=IoT_Data.Screw_RPM	2016-10-27T09:09:11.349Z	4112	ROEX38_00B	s=IoT_Data.Screw_RPM	2016-10-27T09:08:56.402Z
s=IoT_Data.Screw_ID	2016-10-27T10:33:01.913Z	IDderSchnecke3	ROEX38_00B	s=IoT_Data.Screw_ID	2016-10-27T10:32:48.910Z
s=IoT_Data.Screw_ID	2016-10-27T09:09:11.371Z	IDderSchnecke2	ROEX38_00B	s=IoT_Data.Screw_ID	2016-10-27T09:08:56.400Z
s=IoT_Data.ActSpeed_Screw	2016-10-27T09:09:11.332Z	900	ROEX38_00B	s=IoT_Data.ActSpeed_Screw	2016-10-27T09:08:56.395Z
s=IoT_Data.ActSpeed_Screw	2016-10-27T10:33:01.394Z	500	ROEX38_00B	s=IoT_Data.ActSpeed_Screw	2016-10-27T10:32:48.896Z
s=IoT_Data.ActSpeed_Screw	2016-10-28T07:07:59.441Z	0	ROEX38_00B	s=IoT_Data.ActSpeed_Screw	2016-10-28T07:07:44.651Z
s=IoT_Data.ActSpeed_Screw	2016-10-28T07:08:13.992Z	0	ROEX38_00B	s=IoT_Data.ActSpeed_Screw	2016-10-28T07:08:01.354Z
s=IoT_Data.ActSpeed_Screw	2016-10-28T07:08:16.632Z	0	ROEX38_00B	s=IoT_Data.ActSpeed_Screw	2016-10-28T07:08:03.731Z
s=IoT_Data.ActSpeed_Screw	2016-10-28T07:08:20.303Z		ROEX38_00B	s=IoT_Data.ActSpeed_Screw	2016-10-28T07:08:06.531Z
s=IoT_Data.ActSpeed_Screw	2016-10-28T07:08:29.131Z	0	ROEX38_00B	s=IoT_Data.ActSpeed_Screw	2016-10-28T07:08:16.430Z
s=IoT_Data.ActMotorLoad	2016-10-27T10:33:01.908Z	50	ROEX38_00B	s=IoT_Data.ActMotorLoad	2016-10-27T10:32:48.906Z
s=IoT_Data.ActMotorLoad	2016-10-27T09:09:11.358Z	20	ROEX38_00B	s=IoT_Data.ActMotorLoad	2016-10-27T09:08:56.398Z
s=IoT_Data.ActMeltPressure	2016-10-27T10:33:01.446Z	100	ROEX38_00B	s=IoT_Data.ActMeltPressure	2016-10-27T10:32:48.896Z
s=IoT_Data.ActMeltPressure	2016-10-27T09:09:11.286Z	50	ROEX38_00B	s=IoT_Data.ActMeltPressure	2016-10-27T09:08:56.396Z

Abbildung 50: Tabellenspeicher Einträge der Daten, Quelle: Eigene Darstellung.

6 PREDICTIVE ANALYTICS - ANWENDUNGSPROJEKT

Um einen Nutzen aus der erfolgten Datenanbindung des CPPS „Extruder“ an das Cloud System „Azure Cloud“ zu generieren, wurde im Zuge eines neuen Projektes ein Anwendungsszenario definiert, welches im Zuge einer weiteren Anlage umgesetzt werden soll. Es soll durch die Möglichkeit der Datenspeicherung die Daten der Schnecke des Extruders erfasst werden. Die Schnecke wird aufgrund von Verschleißerscheinungen nach einer derzeit definierten Laufzeit im Zuge der geplanten Wartung getauscht. Da die Schnecke eine durchaus kostspielige Komponente ist, kommt es auch vor, dass Kunden der Firma Rosendahl Nextrom GmbH die Schnecke nicht im vorgeschriebenen Wartungsintervall tauschen und so einen Ausfall der Anlage riskieren.

Die Verschleißerscheinungen der Schnecke werden durch die Drehzahl, das Schneckenmaterial, die Kunststoffart des Produkts und das Temperaturprofil beeinflusst. Die Auswirkung einer bereits verschlissenen Schnecke ist der geringere Ausbringungswert aus dem Extruder, welcher in der Regelung durch eine höhere Drehzahl kompensiert werden muss.

Die Schnecke soll daher eindeutig gespeichert und mit Messdaten hinterlegt werden. Um die Schnecke eindeutig zu identifizieren wird im Zuge dieses Projektes ein *Radio Frequency Identification* (RFID) System integriert. Die Schnecke wird mit einem RFID-Tag ausgestattet, welcher mit der Type, der Seriennummer und dem Material der Schnecke beschrieben ist. Im Extruder wird ein RFID-Lesegerät eingebaut um die Schnecke zu identifizieren und aufgenommene Messdaten der Schnecke zuzuweisen. Ein weiterer Schritt ist es die wichtigen Parameter zur Detektion des Zustandes der Schnecke zu ermitteln. Es ist absolut essentiell, dass die definierten Daten schon ab dem ersten Einsatz der Schnecke aufgezeichnet werden, damit diese als Referenzprofil dem Analysealgorithmus zur Verfügung gestellt werden können. Ziel wird es sein, eine genaue Bestimmung des Zustands der Schnecke zu erhalten und eine Vorhersage der zu erwartenden Restlebensdauer aufgrund von historischen Daten zu ermitteln.

Aus diesem Kontext ergeben sich verschiedene neue Geschäftsmodelle für Rosendahl Nextrom. Die Vorhersage der Verschleißgrenze der Schnecke (*Predictive Analytics*) kann zum Beispiel als Optionspaket der Anlage angeboten werden. Es wäre durchaus denkbar, dies mit einem zeitgebundenen Pauschalmodell anzubieten, welches den Tausch der Schnecke und die Servicekosten dafür beinhaltet. So kann Rosendahl Nextrom in Bezug auf das heikle Bauteil ihren Kunden ein Servicepaket zur Absicherung deren Produktionsanlagen anbieten.

6.1 Historische Daten

Um generell eine Aussage über den Verschleiß der Schnecke treffen zu können sind historische Daten notwendig. Diese Daten werden aus bestehenden Anlagen exportiert und dem Projekt zur Verfügung gestellt. Im ersten Schritt wird versucht zu eruieren, wie mit den vorhandenen Daten rückwirkend eine Aussage über den Verschleiß der Schnecke getätigt werden kann. Hierbei ist es jedoch notwendig, die vorhandenen Datenpunkte zu beschreiben.

1	DateTime	ANL_Prod	ANL_ReelCnt	EXT1_iLoad	EXT1_iSpee	MES_DWa	MES_VAbz		Counter	Ausstoß
2	28.04.2015 15:49	AERO-DF 03	548	22.22900391	12.6855469	0	19.48242188		1	0.65112782
3	28.04.2015 15:50	AERO-DF 03	548	18.32885742	10.0683594	0	15.99121094		1	0.62961832
4	28.04.2015 15:51	AERO-DF 03	548	18.74389648	10.0488281	0	15.94848633		1	0.63008037
5	28.04.2015 15:52	AERO-DF 03	548	18.37768555	10.0390625	0	15.98510742		1	0.62802596
6	28.04.2015 15:53	AERO-DF 03	548	18.75610352	10.0878906	0	16.02783203		1	0.62939832
7	28.04.2015 15:54	AERO-DF 03	548	19.0612793	10.078125	0	15.97290039		1	0.63095147
8	28.04.2015 15:55	AERO-DF 03	548	18.23120117	10.0390625	0	15.82641602		1	0.63432318
9	28.04.2015 15:56	AERO-DF 03	548	18.41430664	10.0390625	0	16.015625		1	0.62682927
10	28.04.2015 15:57	AERO-DF 03	548	18.84155273	10.078125	0	15.9362793		1	0.63240138

Abbildung 51: Dataset zur Modellbildung, Quelle: Eigene Darstellung.

- **DateTime**
Zeitstempel des Zeileneintrags
- **ANL_ProductName**
Das Produkt, welches zu diesem Eintrag produziert wurde. Diese Information ist von Bedeutung, da nur gleiche Produkte verglichen werden sollten. Das hat mit der Rezeptur und dem Temperaturprofil des jeweiligen Produkts zu tun.
- **ANL_ReelCnt**
Ein Spulenzähler, welcher ein Indikator für die Betriebsart ist (produzierende Phase). Mit der Spule ist die Kabeltrommel am Ende der Produktionslinie gemeint, auf die das fertige Produkt aufgewickelt wird
- **EXT1_iLoad**
Motorlast des Antriebssystems der Schnecke (Quelle: Frequenzumrichter)
- **EXT1_iSpeed**
Motordrehzahl des Antriebssystems der Schnecke (Quelle: Frequenzumrichter)
- **MES_DWarm**
Der Durchmesser des Produkts, welcher in Kombination mit der Anlagengeschwindigkeit ein Indikator für den Ausstoß ist
- **MES_VAbz**
Anlagengeschwindigkeit, Geschwindigkeit des vorbeigeführten Rohleiters

Aus diesen Daten wird das Produkt mit den meisten Datensätzen exportiert. Der erste Schritt ist die Bereinigung der Daten. Das bedeutet, dass Leerzeilen entfernt werden und auch Felder mit dem Wert „0“ bereinigt werden. Um die Aussagekraft des Modells weiter zu erhöhen, werden nur Datenzeilen welche eine Anlagengeschwindigkeit und Drehzahl von größer 10 aufweisen, herangezogen. Das wird damit begründet, dass unter diesem Wert die Anlage nicht im produktiven Betrieb sein konnte. Nach dieser Bereinigung bleiben rund 43.000 Datenzeilen übrig, welche Messdaten von ca. einem Jahr beinhalten.

Für die visuelle Auswertung wird das Dataset um mehrere Spalten erweitert. Diese werden im Folgenden kurz beschrieben.

- **Counter**
Da der Spulenzähler ANL_ReelCnt über alle Produkte, welche auf der Anlage gelaufen sind, gezählt wurde, ist der Wert für die Auswertung eines Produkts nicht aussagekräftig. Deshalb wird der Startwert auf „1“ gestellt und entsprechend dem ANL_ReelCnt hochgezählt. Im Dataset wurden 1.158 Spulen eines Produkts aufgewickelt.

- **Ausstoß**

Die Spalte „Ausstoß“ ist eine Verhältniszahl zwischen der Drehzahl *EXT1_iSpeed* und der Anlagengeschwindigkeit *MES_VAbz*. Da die Geschwindigkeit alleine kein Parameter für den Verschleiß sein kann, wird diese immer im Verhältnis zur Anlagengeschwindigkeit ausgewertet, was zur Folge hat, dass ein stabiler Verlauf erkennbar wird.

Das Dataset wird anschließend grafisch ausgewertet, um grundsätzlich Tendenzen zu ermitteln. In der nachfolgenden Abbildung 52 wird ein Durchschnitt des Wertes „Ausstoß“ nach der produzierten Spule (Count) aufgetragen. Die gestrichelte Linie ist die Tendenz des Ausstoßes über die Spulen. Man kann erkennen, dass der Ausstoß mit der Zeit (abgewickelten Spulen) abnimmt.

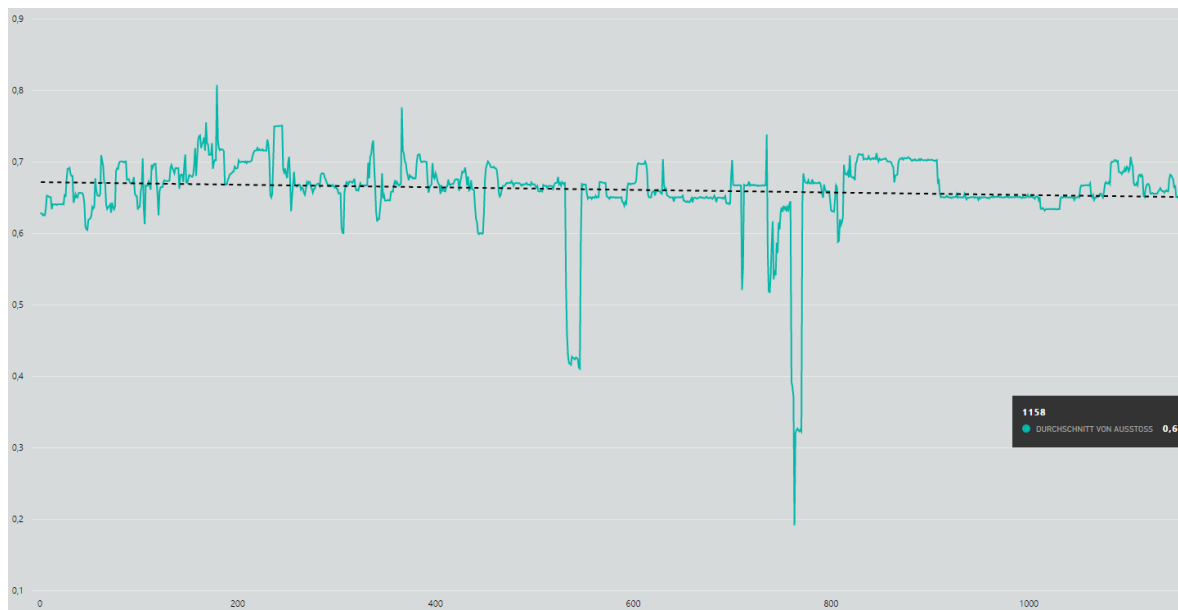


Abbildung 52: Durchschnitt von Ausstoß nach Counter; Quelle: Eigene Darstellung.

In der Abbildung 53 ist die Motorlast nach den produzierten Spulen (Count) aufgetragen. Über den selben Verlauf steigt die Motorlast stetig an, obwohl der Ausstoß grundsätzlich abnimmt.

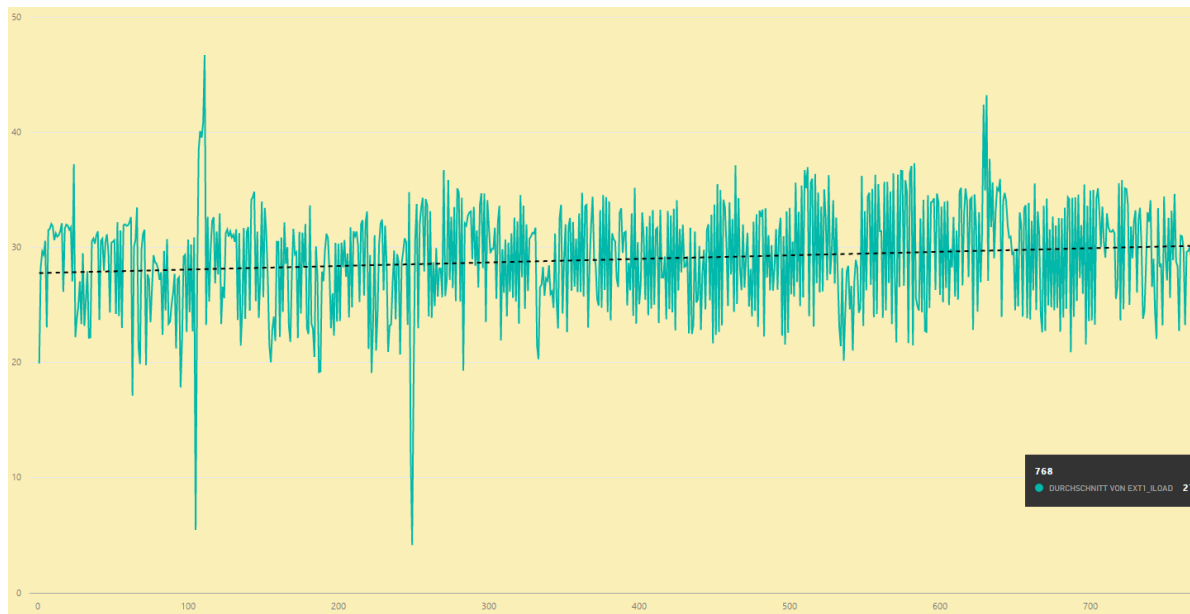


Abbildung 53: Durchschnitt von EXT_iLoad nach Counter, Quelle: Eigene Darstellung.

Dasselbe wie für die Last gilt auch für die Drehzahl des Antriebssystems (EXT_iSpeed). Die Drehzahl steigt über den Betrachtungszeitraum stetig an.



Abbildung 54: Durchschnitt von EXT_iSpeed nach Counter, Quelle: Eigene Darstellung.

6.2 Der Abnutzungsvorrat

Grundsätzlich wäre die Idee zur Ermittlung des Zustandes der Schnecke einen Prozentwert für den Verschleiß bzw. den Abnutzungsvorrat zu ermitteln. Der Abnutzungsvorrat beschreibt den Status der Schnecke, und soll ein Indikator dafür sein, wann die Schnecke die Schadensgrenze erreicht haben wird. Der Begriff des Abnutzungsvorrats kommt aus der Instandhaltung und beschreibt den Verschleiß eines Bauteils. Ein Bauteil startet immer mit 100% seines Abnutzungsvorrates. Durch die Verwendung des Bauteils nützt sich der Bauteil ab und verliert einen Teil seines Abnutzungsvorrates, bis er die Schadensgrenze

erreicht hat und ausfällt. Durch Wartung und Instandsetzung kann der Abnutzungsvorrat eventuell wieder angehoben werden. Wenn das nicht möglich ist muss ein Tausch durchgeführt werden. Der Istzustand ist ein Indikator für den Verschleiß. Da jedoch ein Istzustand von mehreren Faktoren abhängt, müssen die Messdaten in Verhältnisse gestellt werden. Die Ermittlung des Istzustandes ist auch nur mit dem Abgleich eines Metamodells möglich, welches durch die bereits im vorherigen Kapitel beschriebenen Daten erstellt werden kann. Mit diesem Metamodell ist auch eine Vorhersage der Restlebensdauer bis zur Schadensgrenze möglich. Dies kann als Zeitwert, in Laufmeter, oder in abgewickelten Spulen erfolgen.

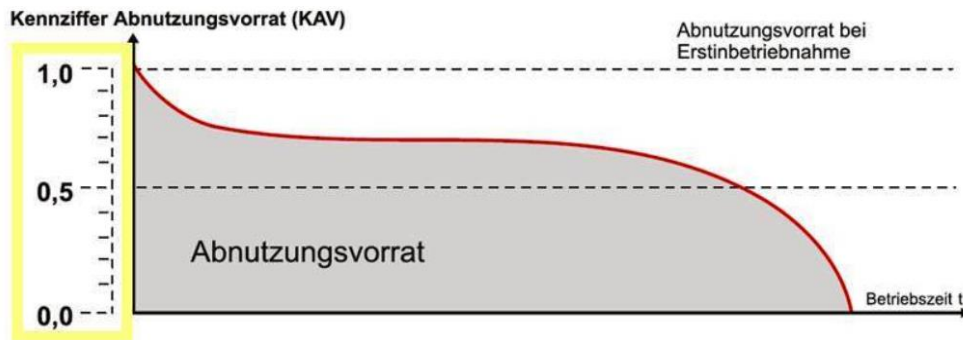


Abbildung 55: Abnutzungsvorrat; Quelle: Wikimedia Foundation (2014), Online-Quelle [11.11.2016].

6.3 Modellbildung in der Microsoft Azure Cloud

Der Bereich *Machine Learning* basiert auf verschiedenen Analyseverfahren sowie Lernalgorithmen. Die Grundlage dazu ist immer eine unbekannte Zielfunktion, die meist sehr komplex ist. Eine Modellbildung im klassischen Sinne ist sehr aufwendig und die Genauigkeit der Vorhersage steht oft nicht in Relation mit den Kosten für deren Erstellung. In vielen Fällen reicht eine Annäherung meist aus. *Machine Learning* Algorithmen übernehmen nicht mehr Aufgaben als vorhin beschrieben. Sie schätzen einen Wert voraus, welcher aber nicht exakt der Wirklichkeit entspricht. Das Modell lernt aus Lerndaten und Beispieldaten und stellt eine Hypothese auf, wie die unbekannte Zielfunktion auf Eingangsdaten reagieren könnte. Meist werden solche *Machine Learning* Algorithmen stetig mit den realen Messdaten verglichen, um das Schätzergebnis zu evaluieren und fortlaufend zu prüfen, ob das Modell auch weiterhin der Realität entspricht.

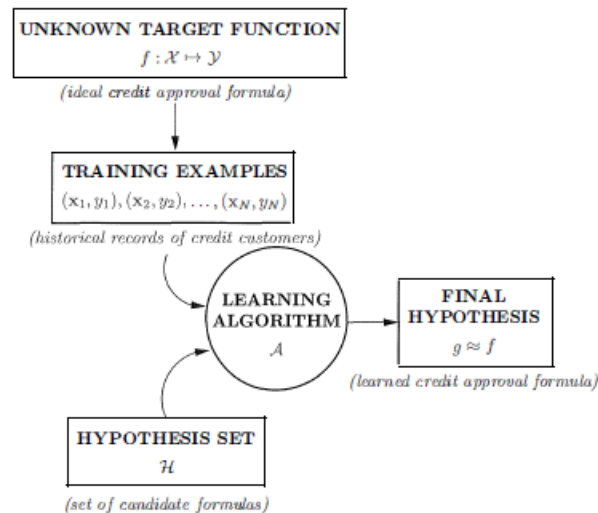


Abbildung 56: Quelle: Abu-Mostafa/Magdon-Ismael/Lin (2012) S.4.

Die *Microsoft Azure Cloud* bietet für die Erstellung von Vorhersagemodellen ein eigenes Softwaretool, welches direkt auf der Cloudapplikation betreibbar ist. Somit wird zur Realisierung der komplexen Modellbildung kein Softwarepaket am Rechner benötigt. Man kann jederzeit und von jedem Rechner aus mittels Browser auf die Software zugreifen. Dieses Softwarepaket nennt sich *Azure Machine Learning Studio* und bietet von der Datenaufbereitung der Lern- und Testdaten, über die Modellbildung und dem Evaluieren der Ergebnisse, bis hin zur Veröffentlichung des Modells mittels Webservices, alle notwendigen Tools zur Realisierung einer *Predictive Analytics* Aufgabe.

6.4 Regression

Regression ist eine Technik welche es möglich macht, numerische Ergebnisse vorherzusagen. Die Eingangsvariablen können numerisch oder kategorisch sein. Jedoch ist das Ergebnis dieser Algorithmen immer numerisch. Regressionen unterscheiden sich in Ihren Ausprägungen vor allem in der Anzahl an abhängigen und unabhängigen Variablen, welche zum Ergebnis beitragen.

6.4.1 Numerisches Ergebnis - Lineare Regression

Lineare Regression wird für einfache, eindimensionale Vorhersagen verwendet. Sie wird hauptsächlich bei Regressionsproblemen mit vielen unabhängigen Variablen verwendet, um ein einzelnes Ergebnis zu erhalten. Zum Testen eines Modells wird das lineare Regressionsmodell verwendet. Dabei werden als Eingangsparameter die bereits im Kapitel 6.1 beschriebenen Variablen verwendet. Allerdings stehen die meisten Variablen in Abhängigkeit zueinander, was das einfache lineare Regressionsmodell an sich ausschließen würde. Die lineare Abhängigkeit der Eingangsvariablen zur Ausgangsvariable, dem Abnutzungsvorrat, ist eine Hypothese. Die Ergebnisse der Vorhersage müssen mit realen Messdaten abgeglichen werden.

Die Formel eines einfachen Regressionsmodells hat folgende Struktur:⁵³

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_n x_n + \dots + \varepsilon$$

- y ist das Ergebnis der Formel. Es stellt den Wert der Vorhersage dar
- x_1, x_2, x_3 , usw. sind die unabhängigen Variablen, welche in diesem Fall die Geschwindigkeit, die Last, usw. sind
- β_0 ist eine Konstante, welche der Schnittpunkt der Regressionsgeraden ist
- b_1, b_2, b_3 , usw. sind die Koeffizienten der unabhängigen Variablen
- ε ist der Fehler oder das Rauschen, das der Antwortvariable zugeordnet ist und kann nicht durch die unabhängigen Variablen x_1, x_2 und x_3 erklärt werden

In der nachfolgenden Abbildung 57 wird ersichtlich, wie ein Datenmodell im *Azure Machine Learning Studio* erstellt wird. Der Datensatz *AERO-DF03* welcher für das Modell vorbereitet wurde, wird in die Cloud geladen und kann für die Modellbildung verwendet werden. Mit dem Modul *Train Model* kann nun auf Basis der Daten und dem verwendeten Algorithmus (in diesem Fall die Lineare Regression) ein Modell erstellt werden. Das Modul *Score Model* nutzt das erstellte Modell und kann eine Datenquelle bzw. einen weiteren Datensatz analysieren und eine Vorhersage ausgeben. Der verwendete Datensatz ist in diesem Fall ein weiterer historischer Datensatz. In weiterer Folge sollen Live-Daten der Anlage einfließen.

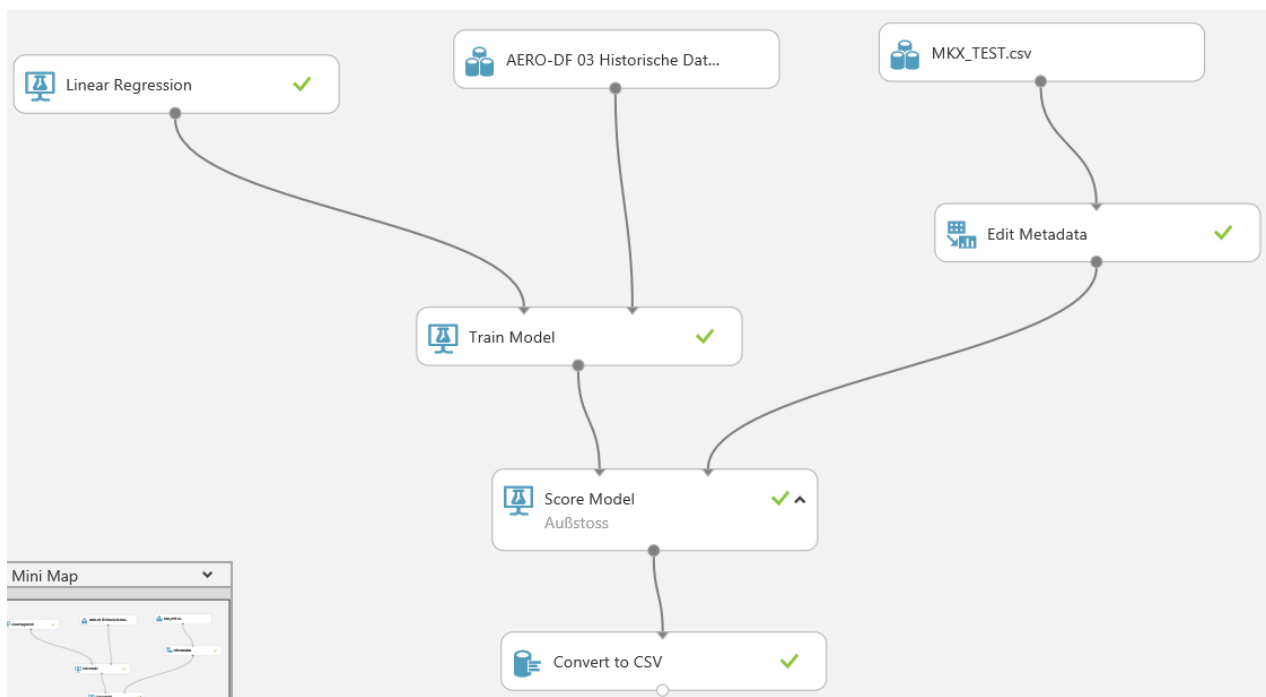


Abbildung 57: Modellbildung in Azure Machine Learning Studio, Quelle: Eigene Darstellung.

⁵³ Vgl. Roger Barga (2015) S. 134.

6.4.2 Voraussagen einer Wahrscheinlichkeit – logistische Regression

Ein weiterer Ansatz ist es die Wahrscheinlichkeit eines Ausfalls der Baugruppe zu bestimmen. In diesem Modell wird etwas benötigt, das ein Ergebnis auf den Wahrscheinlichkeitsbereich 0 bis 1 beschränkt. Anders ausgedrückt, wie hoch die Wahrscheinlichkeit eines Ausfalls in einem Zeitraum in Prozent ist. Somit kann die Ausgabe als Wahrscheinlichkeit interpretiert werden, welche für ein binäres Ereignis steht, welches in einem gewissen Zeitraum auftreten kann (Ausfall). Die lineare Klassifikation beschäftigt sich auch mit einem binären Ereignis. Im Gegensatz dazu ist es jedoch möglich, dass die "Klassifizierung" in der *logistischen Regression* ungewiss sein darf, wobei Zwischenwerte zwischen 0 und 1 diese Unsicherheit ausdrücken. Es wird auch als *Sigmoid* bezeichnet, weil seine Form wie ein abgeflachtes 's' aussieht.⁵⁴

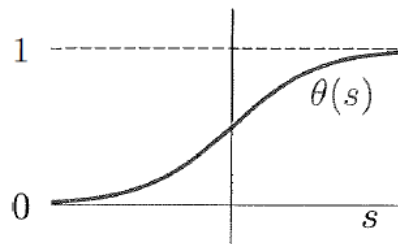


Abbildung 58: Sigmoid, Quelle: Abu-Mostafa/Magdon-Ismail/Lin (2012) S.90.

6.4.3 Neuronales Regressions-Modell

Dieses Vorhersagemodell basiert auf einem künstlichen neuronalen Netz, das ursprünglich entwickelt wurde, um die Funktion des menschlichen Gehirns oder neuronalen Systems zu simulieren bzw. zu imitieren. Das künstliche neuronale Netz besteht aus vielen parallel geschalteten Netzwerken, die aus Elementen bestehen, die als *Neuronen* bezeichnet werden. Diese werden über unidirektionale Signalkanäle miteinander verbunden. Jedes Verarbeitungselement hat eine einzige Ausgangsverbindung, die in beliebig viele Kollateralverbindungen verzweigt. Jedes Neuron trägt das gleiche Signal, das Verarbeitungsausgangssignal. Ein neuronales Netzwerk hat die Fähigkeit, seine strukturellen Bestandteile, also die Neuronen, zu organisieren, um bestimmte Berechnungen um ein Vielfaches schneller durchzuführen.⁵⁵

6.4.3.1 Aufbau eines neuronalen Netzes

Die Grundlage für solche neuronalen Netzwerke ist das Neuron, das auch als Knoten oder Neurode bezeichnet wird. Neuronen werden in der Regel in Schichten gruppiert. Hierzu zählen die Eingabe-, die Ausgabe- und die verborgene -Schicht. Die Anzahl der Neuronen in der Eingangsschicht besteht aus den Parametern (z.B. Ausstoß), die zur Beurteilung des Verschleißes der Schnecke verwendet werden. Die Anzahl der Neuronen in der verborgenen Schicht wird frei vergeben und hängt vom Grad der Komplexität ab. Es gibt keine feste Regel für die Bestimmung der Anzahl der Neuronen in der verborgenen Ebene. Die

⁵⁴ Vgl. Abu-Mostafa/Magdon-Ismail/Lin (2012) S.90.

⁵⁵ Vgl. Palanisamy/Rajendran/Shanmugasundaram (2007) S.37ff.

Anzahl der Ausgangsknoten wird als eins angenommen, um den Wert des Schneckenverschleißes anzuzeigen.⁵⁶

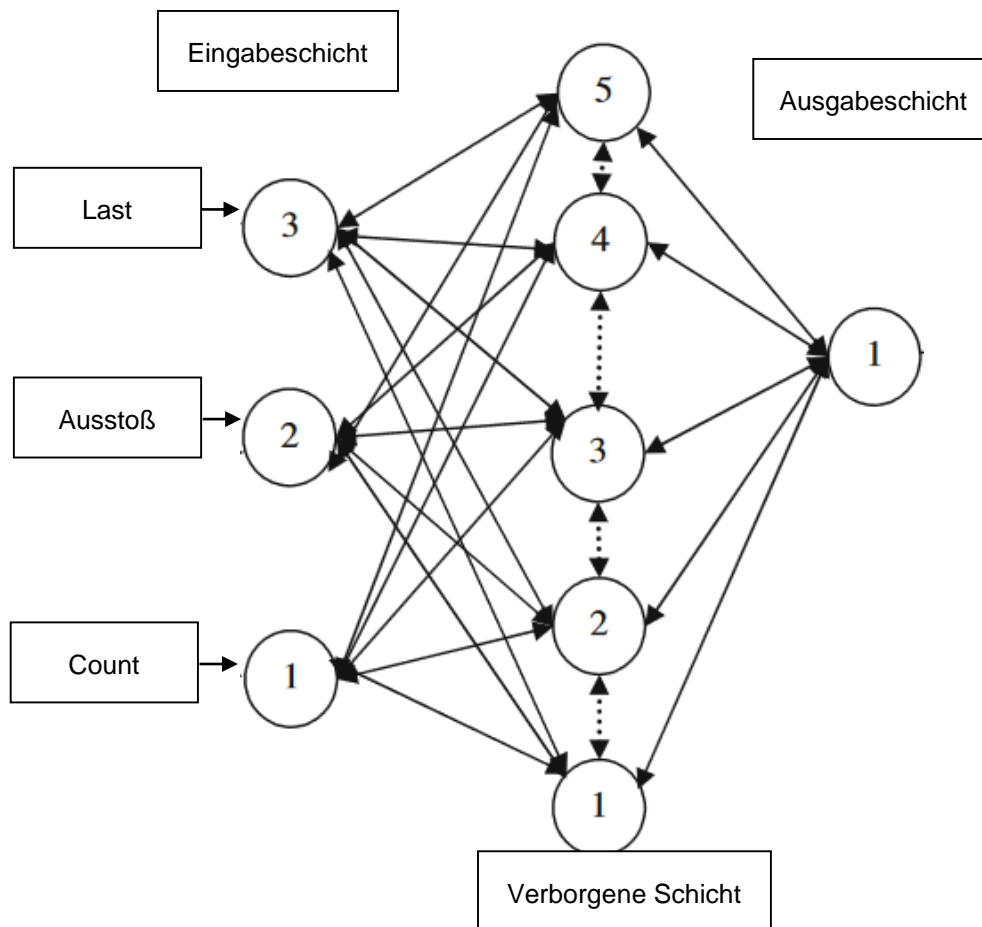


Abbildung 59: Aufbau neuronales Netzwerk, Quelle: Eigene Darstellung.

6.4.3.2 Training, Lernverhalten und typische Topologien

Das Anlernen eines neuronalen Netzes wird, vereinfacht dargestellt, durch die Vergabe der Gewichtungen und Schwellenwerte eines Neurons erreicht. Diese Werte werden für ein spezielles Kriterium optimiert und danach evaluiert, um die Fehlerrate zu ermitteln und gegebenenfalls Parameter nachzubessern.

Es gibt verschiedene Arten von Topologien eines neuronalen Netzwerks, auf die nachfolgend eingegangen wird:⁵⁷

- **Vorwärtsbetriebenes Netz (feed forward network)**

Beschreibt ein Netzwerk, das keine Schleifen oder Rückwirkungen aufweist und somit Informationen nur in eine Richtung weitergibt - von der Eingangsschicht zur Ausgangsschicht

- **Rückgekoppeltes Netz (recurrent network)**

⁵⁶ Vgl. Palanisamy/Rajendran/Shanmugasundaram (2007) S.38ff.

⁵⁷ Vgl. Kruse/Borgelt/Braune/Klawonn/Moewes/Steinbrecher (2015) S.40ff.

Beschreibt ein Netzwerk, das Ausgaben auch auf Eingaben rückkoppeln kann und somit Schleifen und Kreise im Netzwerk aufweist. Der Informationsfluss kann auch von der Ausgangsschicht zur Eingangsschicht erfolgen.

Nach der Konstruktion eines Netzes folgt die Trainingsphase, in der das Netz *lernt*. Theoretisch kann ein Netz durch folgende Methoden lernen:⁵⁸

- Erstellen von neuen Verbindungen oder Löschen bestehender
- Anpassung der Gewichtung eines Neurons
- Veränderung des Schwellenwertes eines Neurons
- Erstellen von Neuronen oder Löschen bestehender

6.4.4 Boosted Decision Tree Regression

Dem *Boosted Decision Tree* Algorithmus liegt der sogenannte *Entscheidungsbaum* zu Grunde. Ein Entscheidungsbaum ist eine verschachtelte Anordnung von *IF*-Abfragen, die eine Verästelung bewirken und eine Entscheidung bzw. ein Ergebnis am Ende herbeiführen. Im nachfolgenden Bild kann man dies sehr gut erkennen. Mit Entscheidungsbäumen werden in erster Linie Klassifizierungs- oder Regressionsaufgaben gelöst. In diesem Fall handelt es sich um die Vorhersage eines Wertes und somit um Regression. Im Baum folgt die Abfrage grundsätzlich immer demselben Schema. Da es sich um Zahlenwerte handelt wird meist auf größer gleich abgefragt und entsprechend dem Ergebnis, der *linke* oder *rechte Ast* weiterverfolgt, bis das *Blatt* des Baumes erreicht wurde und somit auch das Ergebnis. Der Algorithmus wurde mit dem sogenannten *Boost* Verfahren weiterentwickelt. *Boosting* ist eine von mehreren klassischen Methoden für die Erstellung von zusammengeführten (Ensemble) Modellen. Im *Azure Machine Learning Studio* nutzen gehobene Entscheidungsbäume eine effiziente Implementierung des *Multiple Additive Regression Trees* (MART) *Gradientenverstärkungsalgorithmus*. Es baut jeden Regressionsbaum schrittweise auf, wobei eine vordefinierte *Verlustfunktion* verwendet wird, um den Fehler in jedem Schritt zu messen und im nächsten Schritt zu korrigieren. Somit ist das Vorhersagemodell ein Ensemble von schwächeren Vorhersagemodellen. In der Abbildung 60 kann man erkennen, dass das Modell insgesamt 100 Entscheidungsbäume mittels dieser Methode erstellt hat und aus diesen 100 Entscheidungsbäumen ein Ergebnis aus der Summe der Ergebnisse der einzelnen Entscheidungsbäume generiert wird.⁵⁹

⁵⁸ Vgl. Kriesel (2007), Online-Quelle [17.November.2016] S.53.

⁵⁹ Vgl. Microsoft, Online-Quelle [17.November.2016].

Compare Regressors > Train Model > Trained model

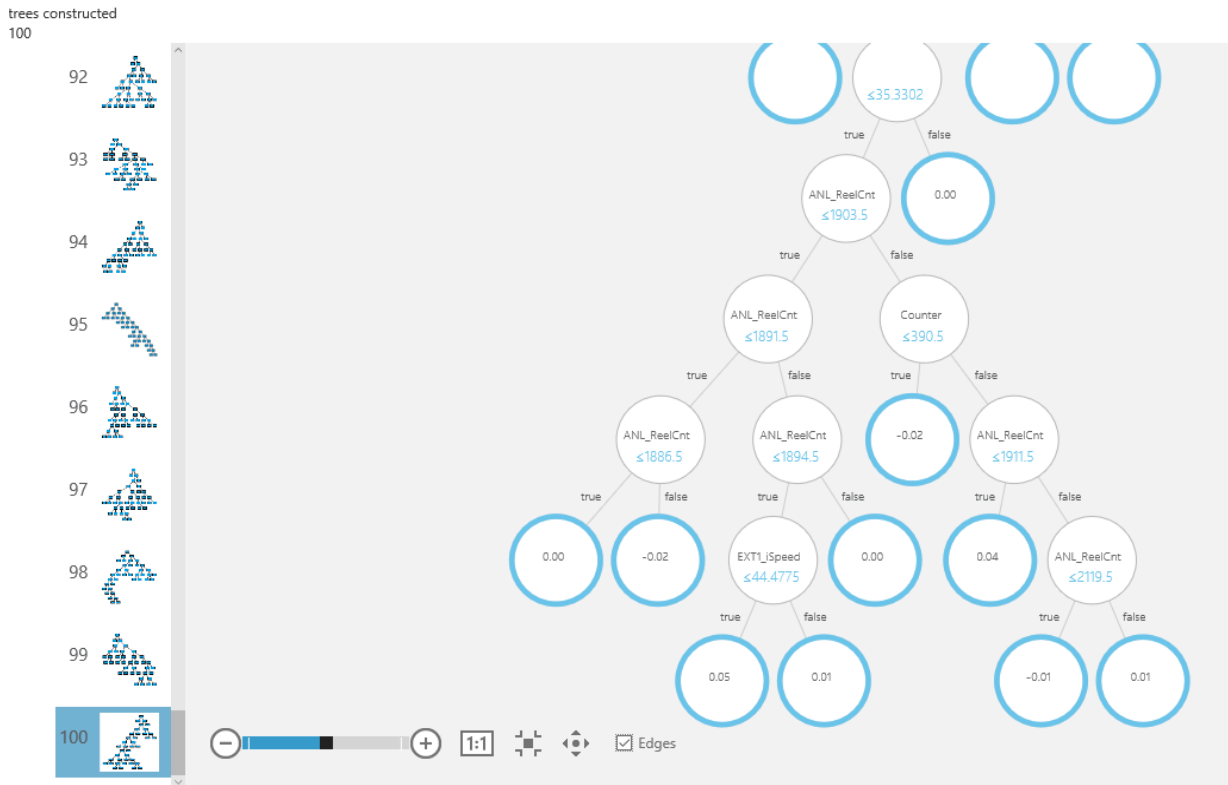


Abbildung 60: Boosted Decision Tree Beispiel aus dem erstellten Modell, Quelle: Eigene Darstellung.

6.4.5 Decision Forest Regression

Der *Decision Forest Algorithmus* basiert gleich wie der *Boosted Decision Tree* Algorithmus auf Entscheidungsbäumen. Außerdem ist dieser Algorithmus auch eine Kombination aus mehreren Vorhersagemodellen (Ensemble). Der große Unterschied besteht darin, dass die verschiedenen Bäume dieses Algorithmus keine Abhängigkeit untereinander aufweisen. Die Entscheidungsbäume werden vielmehr aus unterschiedlichen Teilmengen des Datensatzes ermittelt. Das Ergebnis dieses Algorithmus ist wiederum die Summe der Ergebnisse aller Entscheidungsbäume des Modells.

6.5 Evaluierung der Algorithmen

Um den Grad der Genauigkeit der Vorhersage aus beschriebenen Algorithmen zu bestimmen kann die Fehlerquote mittels Testdaten ermittelt werden. Die folgenden Messwerte werden für die Bewertung von Regressionsmodellen aus dem Evaluierungsprozess ermittelt:⁶⁰

- **Mean absolute error (MAE)** misst, wie nahe die Prognosen des Modells an den tatsächlichen Ergebnissen sind. Umso niedriger dieser Wert ist, umso besser ist das Ergebnis.

⁶⁰ Vgl. Microsoft (2016), Online-Quelle [17.November.2016].

- **Root mean squared error (RMSE)** erstellt einen einzelnen Wert, der den Fehler im Modell zusammenfasst. Durch Quadrieren der Differenz berücksichtigt die Metrik die Differenz zwischen Übervorhersage und Untervorhersage.
- **Relative absolute error (RAE)** ist die relative absolute Differenz zwischen erwarteten und tatsächlichen Werten; Relativ deshalb, weil die mittlere Differenz durch das arithmetische Mittel dividiert wird.
- **Relative squared error (RSE)** normalisiert den Gesamtquadratfehler der vorhergesagten Werte ähnlich, indem er durch den Gesamtquadratfehler der tatsächlichen Werte dividiert wird.
- **Coefficient of detemination** repräsentiert die Vorhersagekraft des Modells als Wert zwischen 0 und 1. Null bedeutet, dass das Modell zufällig ist (erklärt nichts); 1 bedeutet, dass es eine perfekte Vorhersage gibt.

Diese Auswertung kann mit jeglichem Regressionsmodell durchgeführt werden und dient dem Vergleich der verschiedenen Algorithmen auf Ihre Vorhersagequalität. Damit kann eine nachvollziehbare Evaluierung und dementsprechende Auswahl des Verfahrens durchgeführt werden. Für diesen Zweck wird ein neues *Experiment* im *Azure Machine Learning Studio* erstellt. In diesem Experiment wird der bereits für die *Lineare Regression* herangezogenen Datensatz verwendet und in zwei Datensätze aufgeteilt (*Split Data*). Um ein Modell zu trainieren und danach zu testen bzw. evaluieren, benötigt man Lerndaten (*Training Data*) und Testdaten (*Test Data*). Wie in der Abbildung 62 ersichtlich, ist der linke Zweig nach dem Modul *Split Data* an das *Train Model* angebunden und stellt die Lerndaten bereit, wobei der rechte Zweig die Testdaten für das *Score Model* bereitstellt. Im *Score Model* wird auf die Testdaten das gelernte Modell angewandt und eine Vorhersage durchgeführt. Im Ergebnisdatsatz findet man dieses Ergebnis unter einer neuen Spalte *Scored Label* wieder.

DateTime	ANL_ProductName	ANL_ReelCnt	EXT1_iLoad	EXT1_iSpeed	MES_DWarm	MES_VAbz	Counter	Aussto	Verschlei	Scored Labels
06.05.2015 22:18	AERO-DF 03	609	31.45752	31.835938	0	50.238037	55	0.633702	90.22	90.420631
13.10.2015 21:03	AERO-DF 03	1868	29.492188	32.275391	0	49.890137	353	0.646929	50.12	49.594772
30.04.2015 18:02	AERO-DF 03	569	32.366943	31.982422	0	50.048828	22	0.639024	94.26	95.15316
14.10.2015 08:36	AERO-DF 03	1875	35.455322	43.398438	0	65.29541	360	0.664648	48.9	48.792397
14.08.2015 06:44	AERO-DF 03	1340	31.506348	40	0	55.151367	231	0.725277	64.78	64.606941
25.07.2015 00:32	AERO-DF 03	1161	30.938721	39.013672	0	54.730225	159	0.712836	74.02	73.911171

Abbildung 61: Ergebnisauszug der Testdaten des Neuronal Regressions Modells, Quelle: Eigene Darstellung.

Um diese Ergebnisse zu prüfen kann nun mittels *Evaluate Model* das *Scored Label* auf die bereits vorhin beschriebenen Parameter (Fehlerquote) geprüft werden. Die Ergebnisse der Evaluierung des *Neuronalen Regression Modells* sind wie folgt im Detail dargestellt.:

- **Mean Absolute Error:** 0,093373
- **Root Mean Squared Error:** 0,116546

- **Relative Absolute Error:** 0,003931
- **Coefficient of Determination:** 0,999982

Diese Evaluierung wird für alle Regressionsmodelle mit denselben Lern- und Testdaten durchgeführt und in eine Auswertung zusammengeführt. In der nachfolgenden Abbildung wird der gesamten Ablauf der Evaluierung dargestellt.

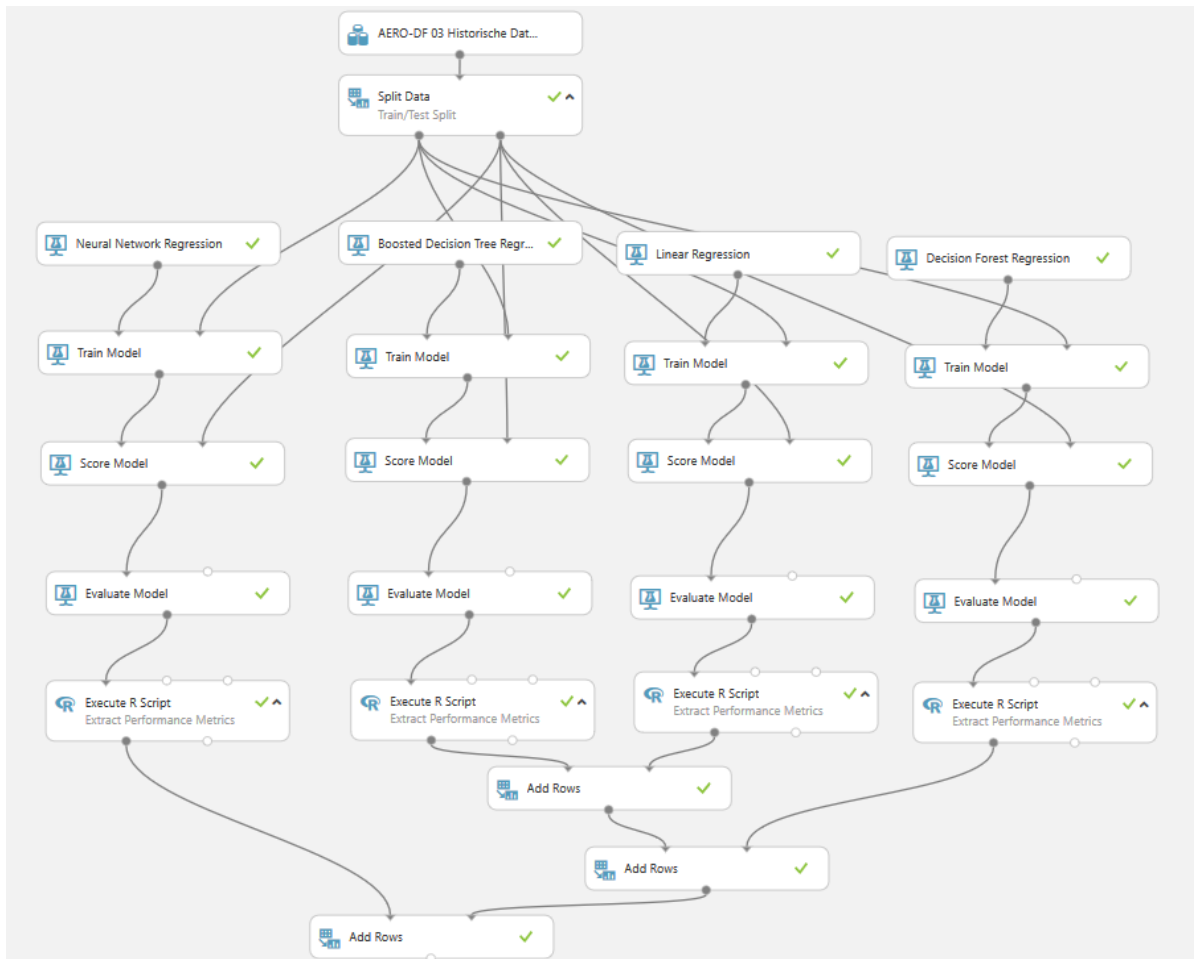


Abbildung 62: Vergleichsbaum zur Erhebung des besten Regressionsmodells anhand von Evaluierungsdaten, Quelle: Eigene Darstellung.

Das Ergebnis dieser Evaluierung auf Basis des gewählten Datensatzes ergibt klare Vorteile für den *Boosted Decision Tree Algorithmus*. Jedoch stellt sich auch heraus, dass die beiden Algorithmen *Neural Network Regression* und *Lineare Regression* auch einsetzbar wären. Die Fehlerquote bei diesen beiden Algorithmen ist auch im unteren Bereich angesiedelt. Die *Decision Forest Regression* jedoch zeigt eindeutige Schwächen für die Aufgabenstellung, welche sich bei einer mittleren absoluten Fehlerabweichung von fast 19,4 zeigt. Da die Linearisierung des Verschleißes von 0 bis 100 % durchgeführt wurde, kann dieser Wert 1 zu 1 in Prozent angegeben werden, was eine Abweichung von fast 20 % ergeben würde. Nachfolgend werden die genauen Berechnungsergebnisse in tabellarischer und grafischer Form dargestellt.

Algorithm	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Neural Network Regression	0.398725	0.529073	0.016788	0.000372	0.999628
Boosted Decision Tree Regression	0.093373	0.116546	0.003931	0.000018	0.999982
Linear Regression	0.958289	1.325482	0.040349	0.002335	0.997665
Decision Forest Regression	19.426449	22.411553	0.817953	0.667415	0.332585

Tabelle 5: Numerische Darstellung des Evaluierungsergebnisses; Quelle: Eigene Darstellung.

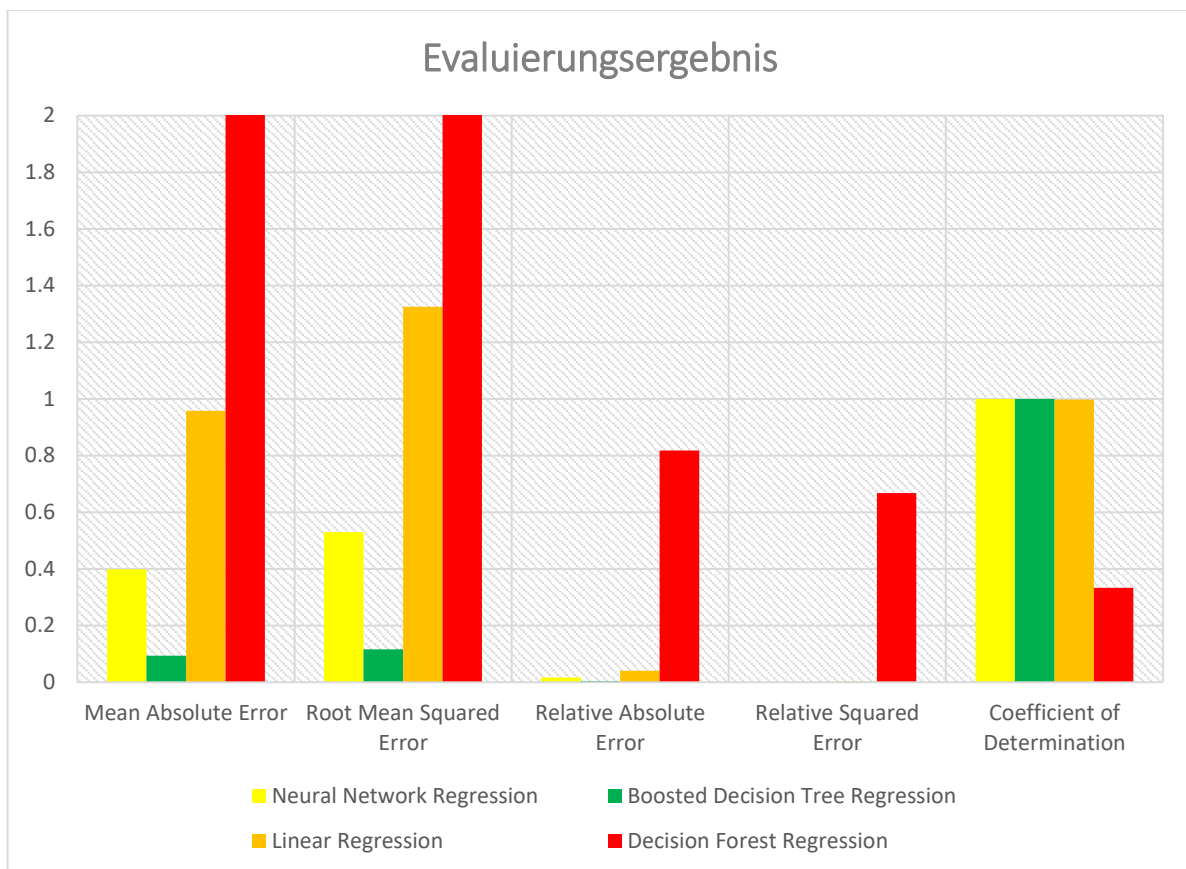


Abbildung 63: Grafische Darstellung des Evaluierungsergebnisses, Quelle: Eigene Darstellung.

Mit den drei Regressionsarten *Lineare Regression*, *Neuronales Regression Modell* und *Boosted Decision Tree Regression* kann durchaus eine Aussage über den Abnutzungsvorrat oder Verschleiß der Schnecke getroffen werden. Die Genauigkeit der Vorhersage muss als nächster Schritt im Entwicklungsprozess mit weiteren Daten von Anlagen evaluiert und geprüft werden.

7 RESÜMEE UND AUSBLICK

Im Zuge der vorliegenden Masterarbeit wurde eine Analyse der Umsetzbarkeit von Industrie 4.0 für die Firma *Rosendahl Nextrom GmbH* durchgeführt. Dabei wurden Teile des *Referenzarchitekturmodells Industrie 4.0* herangezogen und wissenschaftlich betrachtet. Aus diesem wissenschaftlichen Blickwinkel heraus, war es notwendig, die praktischen Anknüpfungspunkte für die Firma *Rosendahl Nextrom GmbH* zu erkennen und darzulegen. Aus der Gegenüberstellung von wissenschaftlichen und praktischen Gesichtspunkten zeigte sich, dass Industrie 4.0 nicht für alle Unternehmen gleich umgesetzt werden kann. Es gibt in jeder Branche und jedem wertschöpfenden Prozess Eigenheiten, die einkalkuliert werden müssen und es notwendig machen, nur die Punkte des Referenzarchitekturmodells heranzuziehen, die auch wirklich Nutzen und Mehrwert bieten. Hier stellte sich heraus, dass die Vernetzung von Produktionsanlagen einer der größten Hebel für Rosendahl Nextrom sein kann.

Ein Ziel des Projektes war es, durch einheitliche Vernetzung und Nutzen von standardisierten Kommunikationsprotokollen zukünftig effizienter bei der Erweiterung bzw. Weiterentwicklung von Produkten zu sein. Dies spiegelt sich im Referenzarchitekturmodell in allen Schichten und hierarchischen Ebenen wieder. Im Zuge der Recherche stellte sich heraus, dass *OPC Unified Architecture*, einer der bevorzugten Protokolle der *Machine2Machine* Kommunikation sein wird. Zu diesem Zweck wurde der *OPC Unified Architecture* Standard in einer Anlage, dem *Extruder*, integriert und die Anbindung designt.

Ein weiteres Ziel war es, die Produktionsanlagen virtuell abzubilden und an ein übergeordnetes System anzubinden, um dem Grundgedanken von *Industrie 4.0* und den *Cyber-Physikalischen-Systemen* gerecht zu werden. Dazu wurde das Thema *Cloud-Computing* aufgegriffen, das im Zuge der Recherche in der behandelten Literatur als zukunftsweisende Technologie erkannt wurde. Dabei ergab sich, dass bei Einsatz von *Cloud-Computing* viele der geforderten Voraussetzungen, wie zum Beispiel Verfügbarkeit und Plattformunabhängigkeit, erfüllt werden können. Andererseits musste festgestellt werden, dass die Datenintegrität und der Schutz von Wissen einen großen Stellenwert einnehmen müssen, um diese Systeme im großen Stil einzusetzen. Es kann anhand der beschriebenen Cloud-Systeme darauf geschlossen werden, dass die Datensicherheit aufgrund der implementierten Security-Funktionen sehr hoch ist. Im Rahmen dieses Projektes werden nur sehr kleine, unkritische Datenmengen angebunden. Erst bei der weiteren Anbindung von *CPS* an die Cloud wird sich zeigen, ob die Security-Funktionen dafür ausreichend sind. Im Zuge der Datenkopplung zwischen dem *Cyber Physikalischen System* und der Cloud ergab sich auch eine Diskrepanz zwischen Theorie und praktischer Umsetzung. In der theoretischen Abhandlung wurde festgestellt, dass die Verringerung von *Medienbrüchen* in der Kommunikation ausschlaggebend für den Erfolg von *Industrie 4.0* Projekten sein wird. Praktisch konnte das leider nicht erreicht werden, da für die Anbindung der *Speicher Programmierbaren Steuerung* (SPS) und der Cloud ein *Gateway* implementiert werden musste. Hierbei wird sehr deutlich, dass für die propagierte *Plug&Play* Kommunikation in der Industrie 4.0 noch einiges an Standardisierungs- und Entwicklungsarbeit zu leisten ist.

Durch Recherchen zur Nutzung der Vorteile der Vernetzung von Produktionsanlagen wurde das Gebiet des *Maschinellen Lernens* fokussiert. Durch den Einsatz von Lernalgorithmen ergeben sich viele Möglichkeiten, wie zum Beispiel die Nutzung von Daten um Informationen für die Entwicklung, das Nutzerverhalten, aber auch vorbeugende Wartung zu generieren. Dabei zeigte sich die Möglichkeit ein

Datenanalyseprojekt mitzugestalten. Die Schnecke des Extruders sollte zukünftig mittels Prozessdaten überwacht werden, um den Zustand der Schnecke vorherzusagen und Ausfälle zu minimieren. Zu diesem Zweck wurden mehrere Datenmodelle in der Cloud mit historischen Daten erstellt. Es wurde ein linearer Verlauf der Abnutzung der Schnecke angenommen, welcher mit verschiedensten Algorithmen analysiert wurde. Im Zuge der Evaluierung der Vorhersageergebnisse konnte der genaueste Algorithmus festgestellt werden. Da jedoch keine genauen Informationen zum Verschleißverhalten jemals aufgenommen oder gemessen wurden, ist der lineare Verlauf des Verschleißes eine reine Annahme und muss durch Abgleichen der Daten an echten Extrudern validiert werden. Hierbei spielt auch die Qualität der historischen Daten eine große Rolle. Die Vorarbeit, die in dieser Masterarbeit zum Thema *Predictive Analytics* geleistet wurde, sollte mit weiteren historischen Daten erweitert und verbessert werden. Die Validierung der Daten konnte im Zuge dieser Arbeit nicht durchgeführt werden, da dies den zeitlichen Rahmen gesprengt hätte.

Im nächsten Schritt muss die Datenspeicherung zur Validierung der Modelle vorangetrieben werden. Nur durch die Aufzeichnung, der in der Arbeit bereits definierten Datenpunkte an einem im Betrieb befindlichen Extruder, kann das Modell weiterentwickelt werden. Hierzu muss die entwickelte Schnittstelle in einem in Produktion befindlichen Extruder eingebunden werden. Nur dann ist es möglich, Daten einer Schnecke von der Inbetriebnahme bis zum Tausch der Schnecke aufzuzeichnen und eine echte Überprüfung des Modells durchzuführen.

Das Know-How, welches in der Firma Rosendahl Nextrom GmbH aufgrund der voranschreitenden Datenanalyseverfahren aufgebaut wird, sollte für weitere Anwendungen in ihren Produktionsanlagen eingesetzt werden. Durch die Implementierung der beschriebenen Kommunikationsschnittstelle in diesen Anlagen, kann die Datenanalyse leicht umgesetzt werden.

Das fachliche Feld, das sich durch *Industrie 4.0* auftut, ist enorm. Ein gekapseltes Tätigkeitsfeld in Unternehmen wird in Zukunft nicht mehr gefragt sein. IT- und Automatisierungsinfrastruktur verschmelzen zusehends und machen es notwendig eine durchgängige Ausbildung in vielen Bereichen zu genießen. Das Gebiet *Maschinelles Lernen* wird zukünftig ein zusätzliches Fachgebiet des Automatisierungs- und IT-Fachmanns sein (evtl. auch Verfahrenstechnikers). Im Bereich der Datenanalyse steckt das Potential für Effizienz- und Effektivitätssteigerung, welche durch Industrie 4.0 ausgeschöpft werden möchte. Viele Branchen nutzen bereits sehr lange diese Technologie, wie zum Beispiel Kreditkartenunternehmen. In der Industrie wurde das Potential von Datenanalyse erst seit kurzem erkannt. Fachleute im Bereich der Datenanalyse werden gefragter sein denn je. Demzufolge werden Mitarbeiter und Mitarbeiterinnen, die ihre Kernkompetenz in der Automatisierungstechnik und der Datenanalyse haben, noch gefragter sein. Aus diesem Grund wird es notwendig sein, diese Kompetenz in das Studium der Automatisierungstechnik mitaufzunehmen.

LITERATURVERZEICHNIS

Gedruckte Werke (16)

International Electrotechnical Commission, International (Hrsg.) (2010): *IEC/TR 62541-2 OPC Unified Architecture - Teil 2: Modell für die IT-Sicherheit*

Network Working Group (Hrsg.) (2006): *RFC4627: The application/json Media Type for JavaScript Object Notation (JSON)*

Abu-Mostafa, Yaser; Magdon-Ismael, Malik; Lin, Hsuan-Tien (2012): *Learning from Data*, 1.Edition Auflage

Bauernhansl, Thomas; Hompel, Michael; Vogel-Heuser, Birgit (2014): *Industrie 4.0 in Produktion, Automatisierung und Logistik*, Springer Vieweg

E., Westkämper (2012): *Forschung für die Fabriken der Zukunft*, Fraunhofer IPA, Stuttgart

Filler, Ines (2015): *Cloud Computing als Baustein von Industrie 4.0*, Diplomica Verlag GmbH, Hamburg

Hans-Friedrich Jacobi (2013): *Digitale Produktion*, Springer Verlag, Berlin Heidelberg

Jörg, Schwenk (2010): *Sicherheit und Kryptographie im Internet*, 3.Auflage Auflage, Vieweg+Teubner Verlag, Bochum

Kruse, Rudolf; Borgelt, Christian; Braune, Christian; Klawonn, Frank; Moewes, Christian; Steinbrecher, Matthias (2015): *Computational Intelligence. Eine methodische Einführung in Künstliche Neuronale Netze, Evolutionäre Algorithmen, Fuzzy-Systeme und Bayes-Netze*, 2. Auflage, Springer Verlag, Magdeburg

Lange, Jürgen; Iwanitz, Franz; Burke, Thomas (2013): *Von Data Access bis Unified Architecture*, 5.Auflage Auflage, VDE Verlag GmbH

Lass, Sander; Kotarski, David (2014): *IT-Sicherheit als besondere Herausforderung von Industrie 4.0*

Mitchell, Tom (1997): *Machine Learning*

Niggemann, Oliver; Biswas, Gautam; Kinnebrew, John; Khorasgani, Hamed; Volgmann, Sören; Bunte, Andreas (2015): *Data-Driven Monitoring of Cyber-Physical Systems*, Springer Verlag

Palanisamy, P.; Rajendran, I.; Shanmugasundaram, S. (2007): *Prediction of tool wear using regression and ANN models in end-milling operation*, Springer-Verlag, India

Roger Barga, Valentine (2015): *Predictive Analytics with Azure Microsoft Azure Machine Learning*, 2nd Edition Auflage, Apress

Wissenschaftliche Artikel (1)

Niggemann, Prof. (2014): *BigData in der Produktion*, in: visIT - Industrielle Datenanalyse, Fraunhofer IOSB, S. 9

Online-Quellen (15)

VDI/VDE-Gesellschaft (2015): *Referenzarchitekturmodell Industrie 4.0 (RAMI4.0)*

www.vdi.de [Stand: 17.08.2016]

Plattform Industrie 4.0 (2016): *Struktur der Verwaltungsschale - Fortentwicklung des Referenzmodells für die Industrie 4.0-Komponente*

www.plattform-i40.de [Stand: 17.08.2016]

SAS Institute GmbH (2013): *Betrugserkennung mit analytischen Verfahren*

www.sas.de [Stand: 08.08.2016]

W3Techs on (2016): *Usage of character encodings for websites*

https://w3techs.com/technologies/overview/character_encoding/all/ [Stand: 28.10.2016]

Wikimedia Foundation (2014): *Academic dictionaries and encyclopedias*

http://de.academic.ru/pictures/dewiki/75/Kennziffer_Abnutzungsvorrat_KAV.jpg [Stand: 11.11.2016]

Microsoft (2016): *Microsoft Azure - Evaluate Model*

<https://msdn.microsoft.com/library/en-us/Dn905915.aspx> [Stand: 17.November.2016]

Microsoft (2016): *Boosted Decision Tree Regression*

<https://msdn.microsoft.com/library/en-us/Dn905801.aspx> [Stand: 17.November.2016]

Network Working Group (2003): *RFC3629: UTF-8, a transformation format of ISO 10646*

<https://tools.ietf.org/html/rfc3629> [Stand: 13.Oktober.2016]

Network Working Group (2008): *RFC 5280 - Internet X.509 Public Key Infrastructure Certificate*

<https://tools.ietf.org/html/rfc5280> [Stand: 12.Oktober.2016]

Foundation, OPC (2016): *Toolkit [Case Study] and Pub/Sub White Paper*

<https://opcfoundation.org/opc-connect/2016/06/toolkit-case-study-and-pubsub-white-paper/> [Stand: 04.08.2016]

Hoffmeister, Dr. (2015): *Die Industrie 4.0- Komponente*

www.zvei.org [Stand: 22.08.2016]

Kagermann, H; Wahlster, W; Helbig, J. (2013): *Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0 – Abschlussbericht des Arbeitskreises Industrie 4.0*

www.bmbf.de [Stand: 10.07.2016]

Kriesel, David (2007): *Ein kleiner Überblick über Neuronale Netze*

http://www.dkriesel.com/science/neural_network [Stand: 17.November.2016]

Kyle, Bruce (2012): *Windows Azure Security Best Practices – Part 5: Claims-Based Identity, Single Sign On*

<https://blogs.msdn.microsoft.com/usisvde/2012/03/13/windows-azure-security-best-practices-part-5-claims-based-identity-single-sign-on/> [Stand: 22.09.2016]

Weiser, Mark (1991): *UCI Donald Bren, School of Information & Computer Sciences*

<https://www.ics.uci.edu/~corps/phaseii/Weiser-Computer21stCentury-SciAm.pdf> [Stand: 26.08.2016]

ABBILDUNGSVERZEICHNIS

Abbildung 1: Rosendahl Nextrom – Firmengeschichte, Quelle: Rosendahl Nextrom GmbH.	2
Abbildung 2: CIM-Wheel, Quelle: Hans-Friedrich Jacobi (2013).	6
Abbildung 3: Ingenieurrelevante Entwicklungen im Kontext der Vision „Factory of the Future.“, Quelle: Hans-Friedrich Jacobi (2013).	7
Abbildung 4: Referenzarchitekturmodell Industrie 4.0, Quelle: VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016].	8
Abbildung 5: Typen und Instanzen im Lebenszyklus, Quelle: VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016].	11
Abbildung 6: Die vier wichtigsten Aspekte von Industrie 4.0, Quelle: VDI/VDE-Gesellschaft (2015), Online-Quelle [17.08.2016].	12
Abbildung 7: Automatisierungspyramide, Quelle: Friedl (2010).	12
Abbildung 8: Vom Gegenstand zur Industrie 4.0-Komponente, Quelle: Hoffmeister (2015), Online-Quelle [22.08.2016].	14
Abbildung 9: Vernetzungsmöglichkeiten, Quelle: Hoffmeister (2015), Online-Quelle [22.08.2016].	15
Abbildung 10: Grobstruktur der Verwaltungsschale, Quelle: Plattform Industrie 4.0 (2016), Online-Quelle [17.08.2016].	16
Abbildung 11: Stufen der Optimierung durch Verringerung von Medienbrüchen, Quelle: Bauernhansl/Hompel/Vogel-Heuser (2014).	17
Abbildung 12: Kommunikation innerhalb CPPS mittels Agenten, Quelle: Bauernhansl/Hompel/Vogel-Heuser (2014).	18
Abbildung 13: Visualisierung der symmetrischen Verschlüsselung, Quelle: Jörg (2010).	23
Abbildung 14: Visualisierung einer asymmetrischen Verschlüsselung, Quelle: Jörg (2010).	23
Abbildung 15: Aufbau eines Secure Channel, Quelle: International Electrotechnical Commission (Hrsg.) (2010).	24
Abbildung 16: Aufbau einer Session, Quelle: International Electrotechnical Commission (Hrsg.) (2010).	25
Abbildung 17: Datenspeicherung aus der Anlage in die Cloud, Quelle: Eigene Darstellung.	27
Abbildung 18: Prinzip der Datenanalyse, Quelle: Niggemann/Biswas/Kinnebrew/Khorasgani/Volgmann/Bunte (2015).	30
Abbildung 19: Defense in Depth Überblick, Quelle: Siemens AG.	32
Abbildung 20: Technische Übersicht – Extruder fix am Boden befestigt, Quelle: Betriebs- und Wartungsanleitung ROEX S.14.	35
Abbildung 21: Variante 1 – Visuelle Darstellung, Quelle: Eigene Darstellung.	38

Abbildung 22: Variante 2 – Visuelle Darstellung, Quelle: Eigene Darstellung	39
Abbildung 23: Variante 3 – Visuelle Darstellung, Quelle: Eigene Darstellung	40
Abbildung 24: Migration der Software, Quelle: Eigene Darstellung.	41
Abbildung 25: Netzwerksicht im TIA Portal V14, Quelle: Eigene Darstellung.....	42
Abbildung 26: Aktivieren des OPC-UA-Servers, Quelle: Eigene Darstellung	44
Abbildung 27: Datenbaustein im SPS Programm zur Datenabholung durch den OPC UA Server, Quelle: Eigene Darstellung.	45
Abbildung 28: Daten in Datenbaustein IoT_Data kopieren, Quelle: Eigene Darstellung.....	46
Abbildung 29: Microsoft Azure Dashboard, Quelle: Eigene Darstellung.....	48
Abbildung 30: IoT Hub hinzufügen, Quelle: Eigene Darstellung.	49
Abbildung 31: IoT Hub im Dashboard, Quelle: Eigene Darstellung.	50
Abbildung 32: Zugriffskontrolle, Quelle: Eigene Darstellung.....	51
Abbildung 33: Device Explorer, Quelle: Eigene Darstellung.	52
Abbildung 34: Create Device, Quelle: Eigene Darstellung.....	52
Abbildung 35: IoT Hub Übersichtsseite, Quelle: Eigene Darstellung.	53
Abbildung 36: Name und Hostname des AMQP-Brokers (Azure IoT Hub) vergeben, Quelle: Eigene Darstellung.	54
Abbildung 37: Endpunkt definieren, Quelle: Eigene Darstellung.	54
Abbildung 38: Shared Access Key des Device, Quelle: Eigene Darstellung.	54
Abbildung 39: Tokenscope für claim based Security einstellen, Quelle: Eigene Darstellung.....	55
Abbildung 40: Server Configuration, Quelle: Eigene Darstellung.	56
Abbildung 41: Open Session, Quelle: Eigene Darstellung.....	56
Abbildung 42: OPC UA – Objekte, Quelle: Eigene Darstellung.	57
Abbildung 43: Erstellen einer Subscription, Eigene Darstellung.....	57
Abbildung 44: Subscription Parameter, Quelle: Eigene Darstellung.....	58
Abbildung 45: Übersicht der angebotenen Variablen, Quelle: Eigene Darstellung.	59
Abbildung 46: Stream Analytics Einstellungen, Quelle: Eigene Darstellung.....	60
Abbildung 47: Ausführungsskript zur Datenüberführung in den Speicher der Azure Cloud, Quelle: Eigene Darstellung.	61
Abbildung 48: Notification im JSON Format, Quelle: Eigene Darstellung.....	61
Abbildung 49: BLOB-Speicher Einträge, Quelle: Eigene Darstellung.....	62
Abbildung 50: Tabellenspeicher Einträge der Daten, Quelle: Eigene Darstellung.....	62

Abbildung 51: Dataset zur Modellbildung, Quelle: Eigene Darstellung.....	64
Abbildung 52: Durchschnitt von Ausstoß nach Counter; Quelle: Eigene Darstellung.	65
Abbildung 53: Durchschnitt von EXT_iLoad nach Counter, Quelle: Eigene Darstellung.	66
Abbildung 54: Durchschnitt von EXT_iSpeed nach Counter, Quelle: Eigene Darstellung.	66
Abbildung 55: Abnutzungsvorrat; Quelle: Wikimedia Foundation (2014), Online-Quelle [11.11.2016].	67
Abbildung 56: Quelle: Abu-Mostafa/Magdon-Ismael/Lin (2012) S.4.	68
Abbildung 57: Modellbildung in Azure Machine Learning Studio, Quelle: Eigene Darstellung.....	69
Abbildung 58: Sigmoid, Quelle: Abu-Mostafa/Magdon-Ismael/Lin (2012) S.90.	70
Abbildung 59: Aufbau neuronales Netzwerk, Quelle: Eigene Darstellung.	71
Abbildung 60: Boosted Decision Tree Beispiel aus dem erstellten Modell, Quelle: Eigene Darstellung... ..	73
Abbildung 61: Ergebnisauszug der Testdaten des Neuronal Regressions Modells, Quelle: Eigene Darstellung.	74
Abbildung 62: Vergleichsbaum zur Erhebung des besten Regressionsmodells anhand von Evaluierungsdaten, Quelle: Eigene Darstellung.....	75
Abbildung 63: Grafische Darstellung des Evaluierungsergebnisses, Quelle: Eigene Darstellung.	76

TABELLENVERZEICHNIS

Tabelle 1: Beschreibung von möglichen Verschlüsselungsoptionen, Quelle: Eigene Darstellung.	22
Tabelle 2: Aufstellung der Hardware Umschlüsselung, Quelle: Eigene Darstellung.	41
Tabelle 3: Gegenüberstellung der Cloud Servicedienstleister, Quelle: Eigene Darstellung.	47
Tabelle 4: Subscription Parameterbeschreibung, Quelle: Eigene Darstellung.	58
Tabelle 5: Numerische Darstellung des Evaluierungsergebnisses; Quelle: Eigene Darstellung.	76

ABKÜRZUNGSVERZEICHNIS

AMQP	Advanced Message Queuing Protocol
BLOB	Binary Large Objects
CA	Certificate Authorities
CBS	Claim Based Security
CIM	Computer Integrated Manufacturing
CPPS	Cyber-Physikalische Produktions-Systeme
CPS	Cyber-Physikalische Systeme
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
I/O	Input/Output
ID	Identification
IoT	Internet of Things
IT	Informationstechnologie
JSON	JavaScript Object Notation
NoSQL	Not only SQL
OPC UA	OLE for Process Control Unified Architecture
PKI	Public-Key-Infrastruktur
RFID	Radio Frequency Identification
RAMI 4.0	Referenzarchitekturmodell Industrie 4.0
SAS	Shared Access Signatures
SOAP	Simple Object Access Protocol
SPS	Speicher programmierbare Steuerung
SSL/TSL	Secure Sockets Layer/Transport Layer Security
STEP	Standard for the exchange of product model data
TCP	Transmission Control Protocol
TSN	Time Sensitive Network
URI	Uniform Resource Identifier
UTF-8	8-Bit UCS Transformation Format
XML	Extensible Markup Language