



Informationstechnik

Hochschule Esslingen
University of Applied Sciences

IT-Innovationen

Band 20
Januar 2018

Grußwort des Dekans



Liebe Leserinnen und Leser,

Digitalisierung ist das beherrschende Thema, das nun auch auf breiter Front in der gesellschaftlichen Diskussion angekommen ist. So allgemein der Begriff auch ist und jeder in seiner Welt den Begriff für sich definiert, so besteht doch in einem Konsens:

Die Informationstechnik ist der Enabler dieser Transformation oder auch das Rückgrat dieses Wandels. So verwundert es nicht, dass die Informationstechnik gesuchter Partner in jeder Digitalisierungsstrategie ist und von ihr die Antworten auf alle offenen Fragen erwartet werden. Wie reagieren wir darauf? Wir bringen unsere Expertise in vielfältige, auch ganz neue Anwendungsdomänen, durch neue Kooperationen über Fachbereichsgrenzen hinweg, ein. Wir stellen uns der ethischen Diskussion und veranstalten im Rahmen des „Studium Generale“ Podiumsdiskussionen mit namhaften Persönlichkeiten aus dem öffentlichen Leben. Aber der Wandel macht auch vor uns selbst nicht halt: Der Umgang mit großen Datenmengen oder komplexen Algorithmen zur Unterstützung des autonomen Fahrens sind nur einige der neuen Herausforderungen. Mit unseren beiden in diesem Wintersemester gestarteten Schwerpunkten *Autonome Systeme* und *Cyberphysische Systeme* im Studiengang Technische Informatik tragen wir diesem Rechnung. Unterstützung erhalten wir dazu aus dem Wissenschaftsministerium mit einer auf fünf Jahre angelegten zusätzlichen Professur in diesem Bereich. Dass wir bereits heute junge Menschen für die Lösung der Fragestellungen von morgen ausbilden, zeigt wieder eindrucksvoll der Ihnen vorliegende Band mit einer Zusammenstellung der von unseren Absolventen in Kooperation mit der Industrie bearbeiteten Bachelor-Thesen.

Viel Freude am Lesen wünscht Ihnen Ihr

A handwritten signature in blue ink that reads "Nonnast". The signature is fluid and cursive.

Prof. Jürgen Nonnast

Dekan der Fakultät Informationstechnik

IMPRESSUM

ERSCHEINUNGSORT

73732 Esslingen am Neckar

HERAUSGEBER

Prof. Jürgen Nonnast
Dekan der Fakultät Informationstechnik
der Hochschule Esslingen - University of Applied Sciences

REDAKTIONSANSCHRIFT

Hochschule Esslingen - University of Applied Sciences
Fakultät Informationstechnik
Flandernstraße 101
73732 Esslingen am Neckar

Telefon +49(0)711.397-4211
Telefax +49(0)711.397-4214
E-Mail it@hs-esslingen.de
Website www.hs-esslingen.de/it

REDAKTION, LAYOUT UND DESIGN

Prof. Dr.-Ing. Reinhard Schmidt
Hochschule Esslingen - University of Applied Sciences
Fakultät Informationstechnik
Flandernstraße 101
73732 Esslingen am Neckar

SATZ, ANZEIGEN und VERLAG

Martin Gärtner, B.Eng. & Fabian Müller, B.Eng.
Hochschule Esslingen - University of Applied Sciences
Fakultät Informationstechnik
Flandernstraße 101
73732 Esslingen am Neckar

ERSCHEINUNGSWEISE

Einmal pro Semester, jeweils Januar und Juni

DRUCK

Pixelgurus
Werbung – Werbetechnik – Digitaldruck.
Horbstraße 8
73760 Ostfildern

AUFLAGE

500 Exemplare

ISSN 1869-6457

| | | |
|-----------------------------|--|----|
| Janes Marcel Abou Chleih | Algorithmen zur Fahrbahnerkennung durch Analyse von Farbbereichen mit Hilfe optischer Bildsensoren zur Trajektorienplanung autonomer Modellfahrzeuge | 1 |
| Berat Acikgöz | Integration eines Laserscanners in ein Elektrofahrzeug | 3 |
| Marvin Ahlgrimm | Entwicklung einer Domänenspezifische Programmiersprache zur Regelspezifikation für die Analyse der Semantik von konstanten Daten in C Dateien | 6 |
| Onur Albayrak | Entwicklung einer Cloud- und Smartphone basierten IoT- Lösung | 8 |
| Hilal Altintas | Implementierung von Big Data Use Cases in RB Analytic Platform zur Entwicklung der Web Based Validation | 10 |
| Ali Askar | Entwicklung einer Edge-Cloud basierten Applikationsplattform für zeitkritische Verkehrsüberwachung im Projekt InVerSiV | 12 |
| Carlo Babo | Synchronisation von Daten in verteilten Systemen | 14 |
| Frieder Baumgratz | Konzept für die Integration pneumatischer und elektrischer Regler in eine Softwarearchitektur | 18 |
| Björn Bieniakiewicz | Auf dem Weg zum offenen Ökosystem für intelligente, sichere und vertrauenswürdige Assistenzsysteme im Internet of Things. Implementierung und technologische Verknüpfung von Haushaltsgeräten mit digitalen Assistenten. | 20 |
| Benjamin Braun | Konzipierung einer Softwarearchitektur für das Produktdatenmanagement mit dem Fokus auf Flexibilität hinsichtlich der Integration heterogener Datenquellen | 23 |
| Nagihan Caliskan | Mobile Development – Entwicklung einer webbasierten mobilen App | 25 |
| Zozan Ceyhan | Erstellung eines Kennzahlenreports und Konzeption eines automatisierten Reporting-Systems im Themenbereich mobiler Betriebssysteme – Apple iOS | 27 |
| Akif Cinar | Konzipieren und Implementieren eines Release Management Prozesses mit dem Fokus auf Continuous Integration in einer Microsoft SharePoint Umgebung | 29 |
| Sara Cizmic | Einführung in Microservices: Ein Architekturparadigma zur Entwicklung moderner Softwaresysteme | 31 |
| Onur Demir | Entwurf und Implementierung einer Diagnoseschnittstelle für einen Automotive-Datenlogger | 34 |
| Fabian Flaig | Analyse der Auswirkungen von Bargeldabschaffung am Beispiel von Mobile Payment | 37 |
| Roland Fronczek | Entwurf und Implementierung eines webbasierten Systems zur Identifikation von Fenstern über QR-Codes | 39 |
| Luis Garcia Barth | Möglichkeiten zur Effizienzsteigerung modellbasiert entwickelter Steuergerätesoftware für Kleinmotoren | 41 |
| Leonie Glanzer | Development of a literature-software based on the snowflake method to support the planning, creation and structuring of novel content | 44 |

| | | |
|-------------------|---|----|
| Nadine Haux | Konzept und Implementierung für die Ermittlung und Anpassung von Maschinenstundensätzen in SAP | 46 |
| Fabian Högl | Blockchain und Smart Contracts – Einführung, Analyse und Entwicklung eines Prototyps in der Automobilbranche | 48 |
| Marcel Hubert | Sichere und zuverlässige Aktualisierung linuxbasierter Firmware auf mobilen Telematiksystemen | 50 |
| Rukiye Kaya | Die Blockchain-Technologie auf Basis dezentraler Strukturen. Von Bitcoin zur massentauglichen Technologie für Industrie und Wirtschaft. | 52 |
| Tanja Kempfert | Rule Based Methods for Object Detection | 54 |
| Dinesh Kishore | Anwendbarkeit ausgewählter Lean Six Sigma Tools auf Verwaltungsprozesse | 57 |
| Tobias Klingel | Neuentwicklung eines Web-Frontends für das Benutzer Rollen-Rechte-System einer zentralen Administrationsplattform | 59 |
| Daniel Kratzel | Cloudgestütztes Devicemanagement – Infrastruktur und Wartungssystem für selbstständige Devices ohne Nutzereingriff | 61 |
| Alexandra Kühner | Projekt „DAISI“ – Entwicklungsmethoden und Erfolgsfaktoren zur Etablierung eines neuen Social Intranet bei Daimler | 63 |
| Wolfram Ladurner | Machbarkeitsanalyse zur technischen Realisierbarkeit der Systemerweiterung IO-Link Wireless des unter IEC 61131-9 standardisierten Kommunikationssystems IO-Link | 66 |
| Patrick Leibold | Konsolidierung und Darstellung von Monitoringinformationen auf einem zentralen Web-Dashboard | 68 |
| Julian Maier | Analyse und prototypische Implementierung von Discovery-Funktionen nach OPC UA im Rahmen von Industrie 4.0 für Sensoren der Automatisierungstechnik einschließlich Integration benötigter Funktionen in die Firmware einer Sensorplattform sowie Aufbau einer Discovery-Infrastruktur auf Basis eines OPC UA Global Discovery Servers | 70 |
| Mark Marszal | Evaluation of Time Responses from Video and Audio Transfer Paths in Automotive Systems | 72 |
| Vanessa Orendi | Digitalisierung und Optimierung der Einsatzplanung Vor-Ort Service | 74 |
| Gianluca Panetta | Konzepterstellung zur Treiberentwicklung für den Linux-Kernel | 76 |
| Patrick Pawlowski | Entwicklung einer mobilen, kamerabasierten Zweipunktstrahlerdetektion und -vermessung zur Lagebestimmung | 79 |
| Patrik Reiske | Entwicklung einer Software für den Betrieb eines Radardemonstrators und Entwurf eines digitalen Filters zur Bildstabilisierung | 81 |
| Eugen Rudel | Videobasierte Objekterkennung unter Verwendung von Bildverarbeitungsalgorithmen für autonom fahrende Modellfahrzeuge | 83 |
| Phillip Sailer | Die Entwicklung elektronischer Glücksspiele | 85 |
| Fabian Schilling | Maschinelle Lernen und Edge Computing im Automobilbereich auf dem TX2 | 87 |

| | | |
|---------------------|--|-----|
| Tobias Schmadl | Konzipierung und Realisierung einer Erweiterung des Engineering-Tools TX-Open der Siemens AG für das Erstellen einer XML-basierten Konfiguration zur Anbindung von Komponenten über den Feldbus Modbus an die Gebäudeleittechnik | 89 |
| Daniel Fabian Seitz | Die Prinzipien von Scrum | 91 |
| Armin Tausch | Jump'n'run Spiel für Touch-Tables in Unity | 93 |
| Kevin Thomas | Synchronisation von Daten in verteilten Systemen | 14 |
| Teodora Todorova | Entwicklung einer Vorgehensweise zur Testautomatisierung der Mercedes me-App mit Hilfe von HP Mobile Center und HP LeanFT | 95 |
| Manuel Tutsch | Algorithmen zur Fahrspurerkennung bei autonom fahrenden Modellfahrzeugen mit Hilfe von maschinellem Sehen und OpenCV | 97 |
| Thuy Vi Uong | Vergleich von Anomalie Erkennungssoftware | 99 |
| Fabian Weber | Evaluation und Optimierung eines Continuous Delivery Prozesses | 101 |
| Paris Wegener | Evaluierung der Integration des PPM-Protokolls in den Engineering-Prozess der OPCON Automatisierungslösungen von Bosch für Anwendungen mit Bezug auf Industrie 4.0 | 103 |
| Marc Wendel | Projektvorgehensmodell für Cloud Business Applications im Rahmen des RB IT-PEP Prozesses | 105 |
| Marvin Wiegand | Untersuchung einer Blockchain im Kontext der IT-Sicherheit bei Transaktionen mit IoT-Daten anhand einer Beispielanwendung | 107 |
| Reinhard Wolfmaier | Kamerabasierte Detektion von Umwelteinflüssen mit Machine Learning Algorithmen | 110 |

Algorithmen zur Fahrbahnerkennung durch Analyse von Farbbereichen mit Hilfe optischer Bildsensoren zur Trajektorienplanung autonomer Modellfahrzeuge

Janes Marcel Abou Chleih*, Jürgen Koch, Reiner Marchthaler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Autonomes Fahren ist spätestens seit dem letzten Jahr ein weitläufig bekannter Begriff und ein Thema von immenser wirtschaftlicher und gesellschaftlicher Bedeutung. Komplexe Assistenz- und Informationssysteme innerhalb des Fahrzeugs werden miteinander verknüpft, um ein gesamtheitliches Informations- und Unterhaltungsangebot bereitzustellen, welches den Fahrer beim Führen des Fahrzeugs unterstützt.

Zukünftig sollen diese Systeme nicht mehr allein als Unterstützungsoption angesehen werden, sondern in der Lage sein, komplexe Verkehrsmuster zu analysieren und das Fahrzeug entsprechend sicher durch den Verkehr zu navigieren. Die Informationen, welche diese Systeme für den Betrieb benötigen, werden dabei über Sensoren eingespeist. Diese analysieren die Verkehrssituation, werten die Umgebungsinformationen wie Fahrspuren und Hindernisse aus. Vor einigen Jahren wurde hierbei vorwiegend auf Ultraschall (2) und Lasertechnologien wie Radar (3) und Lidar (4) gesetzt. In den letzten Jahren werden jedoch zunehmend auch Mono- und Stereokamerasysteme (1) verwendet (siehe Abb. 1). Dies liegt unter anderem daran, dass Kamerasysteme im Vergleich zu Radar und Lidar eine viel höhere Bandbreite von Informationen über ihre Bilddaten bereitstellen [1].

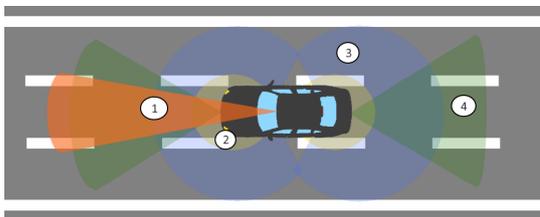


Abbildung 1: Sensoren autonomer Fahrzeuge

Kameras nutzen die physikalischen Eigenschaften von Licht, allem voran den sogenannten photoelektrischen Effekt. Grundlegend ist

über Licht zu wissen, dass es Energie besitzt, die proportional zur Frequenz ist. Die Frequenz ist umgekehrt proportional zur Wellenlänge des Lichtes, welche die Farbe, wie der Mensch sie wahrnimmt, bestimmt. Negativ geladene Platten (Kathoden) verlieren ihre Ladung (Elektronen), sobald diese mit UV-Licht bestrahlt werden [3].

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 |
| 3 | 4 | 3 | 4 | 3 | 4 | 3 | 4 |
| 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 |
| 3 | 4 | 3 | 4 | 3 | 4 | 3 | 4 |
| 1 | 2 | 1 | 2 | 1 | 2 | 1 | 2 |
| 3 | 4 | 3 | 4 | 3 | 4 | 3 | 4 |

Abbildung 2: Beispielhafte Darstellung einer Bayer-Matrix

Kamerasysteme bestehen aus Halbleitern, die eine Art dieses Effektes nutzen. Statt die Ladung zu verlieren, steigert Licht jedoch die Leitfähigkeit des Halbleiters. Anhand dieser Ladung kann wiederum die Farbe des Lichtes gemessen werden. Da das weiße Licht jedoch aus vielen Lichtwellen besteht, nutzt man bei Kameras Filter, die den Lichtstrahl in die Grundfarben Rot, Grün und Blau spaltet. Der Aufbau dieser Filter unterscheidet sich hierbei, was zu einer Klassifizierung in 1-Chip- und 3-Chip-Kameras führt.

Bei **3-Chip-Kameras** hat jeder Pixel drei Sensoren. Vor diesen Sensoren wird ein Prisma angebracht, welches das Licht in rote, grüne und blaue Teile aufspaltet. Jeder Sensor ist für einen dieser Farbanteile zuständig. Schlussendlich wird durch eine Software ein Farbwert aus diesen drei Anteilen errechnet.

Bei **1-Chip-Kameras** steht für den Pixel nur ein Sensor zur Verfügung. Vor diesem wird ein Farbdrehband angebracht, der drei Farbfilter beinhaltet. Dieses Drehband wird bei der Aufnahme eines Bildes schnell gedreht und die resultierenden drei Farbanteile wie bei 3-Chip-Kameras umgerechnet. Eine günstigere Vari-

*Diese Arbeit wurde durchgeführt bei der Firma IT-Designers GmbH, Esslingen

ante für 1-Chip-Kameras ist das Mosaiksystem. Bei diesem werden vier Sensoren zu einem Pixel zusammengefasst, wobei jeder Sensor ein definiertes Farbfilter besitzt. In der Praxis hat sich hierbei die sogenannte Bayer-Matrix durchgesetzt [2].

Über eine Schnittstelle zur Kamera werden die Bilder oder Frames, die meist als **RGB** (Red, Green, Blue) Bilddaten vorliegen, in das System eingespeist und stehen nun für den anstehenden Analyseprozess bereit.

Anschließend sollte eine Vorverarbeitung des Bildes erfolgen. Hierbei werden störende Faktoren, wie Rauschen oder Pixelfehler aus dem Bild entfernt, um eine einfachere Analyse zu ermöglichen. Eine gängige Methode ist es, ein Filter zu nutzen. Hierbei ist der Typ des Rauschens allerdings ausschlaggebend. Während Impulsrauschen ein nicht-lineares Filter, wie den Median-Filter benötigt, ist ein kontinuierliches Rauschen mit einem linearen Filter, wie dem Gaußschen zu beseitigen [4].

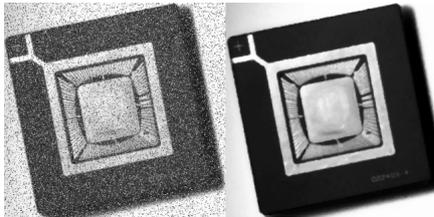


Abbildung 3: Anwendung eines Median-Filters auf ein Bild mit Impulsrauschen

Nach der Vorverarbeitung sollen die Mittelspurmarkierungen mit Hilfe ihrer Farbsignaturen erkannt werden. Hierbei wird im Bereich der Computer Vision häufig der **HSV** (Hue, Saturation, Value) Farbraum genutzt. Dieser Farbraum trennt Farbhelligkeit (Value) von der eigentlichen Farbinformation (Hue) und ermöglicht somit eine effizientere Analyse und Klassifizierung der Farbbereiche [2].

Diese Umwandlung sowie die Analyse der Farbbereiche erfolgen mit Hilfe der **OpenCV** (Open Source Computer Vision) Bibliothek, welche implementierte und optimierte Bildverarbeitungsalgorithmen bereitstellt.

Ist die Vorbereitung des Bildes abgeschlos-

sen, können nun die Fahrbahnmarkierungen aus dem Bild extrahiert werden. Hierbei bietet sich alternativ zu Farberkennung auch eine Konturerkennung über Kanten an, da Fahrbahnmarkierungen mit ihrer Rechteckform effizient gesucht werden können.

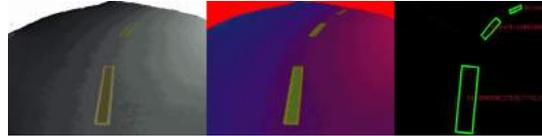


Abbildung 4: Eingabebild im RGB Farbraum, Konvertierung in HSV und finale Analyse

Anhand der gefundenen Fahrbahnmarkierungen kann nun eine Trajektorie entlang der Punktdaten aufgebaut werden.

Dazu können **Bézier-Kurven** oder Parabeln genutzt werden. Bei Bézier-Kurven handelt es sich um ein mathematisches Konstrukt, welches eine Kurve entlang mehrerer Punkte beschreibt. Eine Kurve n -ten Grades wird hierbei durch $n + 1$ Punkte bestimmt (P_0, \dots, P_n).

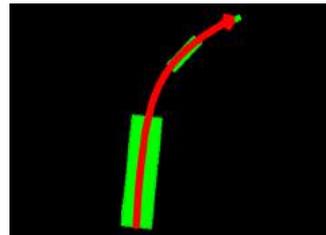


Abbildung 5: Beispielhafte Trajektorie entlang mehrerer Fahrbahnmarkierungen

Eine Voraussetzung für die Funktionsfähigkeit dieses Systems ist allerdings die Verarbeitungsgeschwindigkeit. Die Analyse der Bilddaten muss in Echtzeit erfolgen, damit auf Gefahren und Ereignisse spontan und adäquat reagiert werden kann. Ist dies gegeben, so können die Informationen aus der Trajektorienplanung an das Steuergerät des Fahrzeugs gesendet und somit die Quer- und Längssteuerung durch das autonome System übernommen werden.

[1] Verband der Automobilindustrie: Automatisierung: Von Fahrerassistenzsystemen zum automatisierten Fahren. <https://www.vda.de/dam/vda/publications/2015/automatisierung.pdf>. Version: 2015, Abruf: 17. Oktober 2017

[2] Priese, Lutz: Computer Vision: Einführung in die Verarbeitung und Analyse digitaler Bilder. 1. Heidelberg : Springer Vieweg, 2015. - ISBN 978-3-662-45129-8

[3] Rüchardt, Eduard: Verständliche Wissenschaft. Bd. 35: Sichtbares und Unsichtbares Licht. 2. Berlin and Heidelberg and New York : Springer Verlag, 1938. - ISBN 978-3-642-88181-7

[4] Dang, Thao: Vorlesung Machine Vision, WS2017/2018

Bildquellen:

- Abbildung 1: Nach Verband der Automobilindustrie, 2015
- Abbildung 2: Nach Priese, 2015
- Abbildung 3: Thao Dang, Vorlesung Machine Vision
- Abbildung 4, 5: Eigene Darstellung

Integration eines Laserscanners in ein Elektrofahrzeug

Berat Acikgöz*, Jürgen Koch, Reiner Marchthaler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Motivation

In den 90er-Jahren waren das Antiblockiersystem ABS und die Fahrdynamikregelung ESP die einzigen Systeme, die den Fahrer unterstützten. Während diese Systeme eine unterstützende Funktion haben, so spricht man bei heutigen Systemen bereits von bedingt automatisiertem Fahren nach der SAE-Stufe 3. Ein Beispiel für die Stufe 3 ist der Stauassistent im neuen Audi A8, der bis zu einer Geschwindigkeit von $60 \frac{km}{h}$ alle fahrdynamischen Aufgaben übernimmt. [1]



Abbildung 1: NEV – Neighborhood Electrical Vehicle

In der Forschung werden immer mehr Anstrengungen zur Weiterentwicklung der Fahrerassistenzsysteme erbracht. So sollen bereits im Jahre 2030 die ersten vollautonom fahrenden Autos verfügbar sein [2]. Doch nicht nur dem Fahrer selbst, sondern auch für die Mitfahrer bietet das autonome Fahren einen Mehrwert. Verschiedene Feldtests, wie z.B. in Niederbayern, ermöglichen schon heute den Einsatz von autonom fahrenden Bussen als öffentliches Verkehrsmittel [3]. Möglich wird die Entwicklung intelligenter Fahrerassistenzsysteme jedoch erst durch den Einsatz moderner Umfellsensoren. Neben Ultraschall-, Radar-

und Kamerasensoren werden nun auch Laserscanner in Serienfahrzeugen eingesetzt.

Aufgabenstellung

Diese Abschlussarbeit behandelt die Integration eines Laserscanners bzw. LiDARs in ein Neighborhood Electrical Vehicle, kurz NEV (siehe Abbildung 1). Das NEV dient als Entwicklungsplattform für den Einsatz von Sensoren und zur Entwicklung von Fahrerassistenzsystemen. Bereits integrierte Systeme wie eine ToF-Kamera, by-wire-Systeme (Lenkung, Gaspedal, Bremse) und Kommunikation über den CAN-Bus sind die ersten Schritte zum autonomen Fahren des NEVs.

Die Integration beginnt mit der Auswahl und Beschaffung eines geeigneten Laserscanners und Rechners. Die Aufgabe des Rechners ist die Sensordatenverarbeitung und Objekterkennung mithilfe des „Robot Operating Systems“, kurz ROS, und der „Point Cloud Library“.

Um die vom Scanner detektierten Objekte plausibilisieren zu können, soll eine Sensordatenfusion mit der vorhandenen ToF-Kamera vollzogen werden. Die dadurch erlangten Informationen über das Umfeld sollen einen vorhandenen Notbremsassistenten, durch Verhinderung einer fehlerhaften Notbremsung und Nichterkennen von Objekten, noch sicher machen.

Hardwareauswahl

Der Laserscanner Hokuyo UTM-30LX-EW ist ein 2D-Laserscanner mit einer Reichweite von 30m und einer Winkelöffnung von 270°. Diesem stand ein Laserscanner des Modells SICK LMS-153 mit ähnlichen Eigenschaften gegenüber. Der entscheidende Vorteil des Hokuyos ist die größere Reichweite, welche beim SICK nur 18m beträgt.

Die Winkelauflösung des Hokuyo beträgt $0,25^\circ$. Somit kann er die Umgebung im Vergleich zu einem Radar viel präziser erkennen. Neben der Entfernung der gemessenen Punkte kann auch die Intensität des Echos ausgewertet werden.

*Diese Arbeit wurde durchgeführt bei der Firma GIGATRONIK Stuttgart GmbH, Stuttgart

Das schnelle Scanintervall von 25ms ermöglicht zudem eine genauere Aussage über die Geschwindigkeit. Diese wird durch das Ableiten zweier Distanzmessungen errechnet. Des Weiteren steigert das Multi-Echo-Verfahren die Zuverlässigkeit der Abstandswerte, da pro Messung bis zu drei Echos ausgewertet werden.



Abbildung 2: Laserscanner Hokuyo UTM-30LX-EW

Als Rechner kommt ein Industrie-Tablet Motion CL920 mit einem leistungsstarken Pentium Vierkern-Prozessor, 4GB Arbeitsspeicher und einer 120GB großen SSD zum Einsatz. Dieser stand bereits zur Verfügung und bietet genügend Rechenleistung für die Objekterkennung unter ROS.

Clustering/Segmentierung der Punktwolke

Bei der Segmentierung einer Punktwolke werden benachbarte Punkte unter bestimmten Voraussetzungen zu einem Cluster zusammengefasst. Das Hauptmerkmal für die Segmentierung ist der euklidische Abstand der

Punkte. Für dessen Berechnung ist eine Transformation der Punkte vom Polarkoordinatensystem in das kartesische Koordinatensystem erforderlich.

Um kostbare Rechenzeit einzusparen, sollten zunächst alle nicht relevanten Punkte gefiltert werden. Dabei können entweder einfache Filter wie Range-Filter oder stochastische Filter zur Filterung von verrauschten Messdaten eingesetzt werden.

Die Abstandstoleranz bestimmt anschließend, welche Punkte demselben Cluster zugewiesen werden. Um eine realistische Erkennung zu ermöglichen, ist die Begrenzung der minimalen und maximalen Anzahl der Punkte in einem Cluster notwendig.

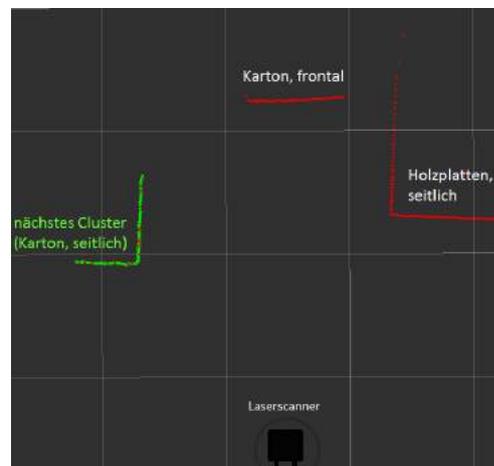


Abbildung 3: Punktwolke mit Clustering

In Abbildung 3 sind in grün alle Punkte visualisiert, die zu dem am nächsten gelegenen Cluster gehören. Die zwei roten Cluster haben eine größere Entfernung zum Laserscanner und sind somit für den Einsatz im Notbremsassistenten weniger von Bedeutung.

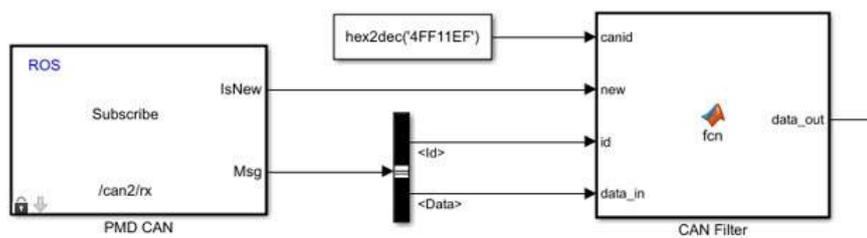


Abbildung 4: Simulink CAN-Eingangsblock nach Portierung

Notbremsassistent

Der sich im Einsatz befindende und auf der ToF-Kamera basierende Notbremsassistent wurde in Matlab/Simulink entwickelt und mithilfe der Codegenerierung auf ein Embedded-Steuergerät, der GIGABOX XL, geflasht.

Bedingt durch den Einsatz von ROS und Linux kann die GIGABOX XL nicht weiterverwendet werden. Des Weiteren fehlt eine Ethernet-Schnittstelle, welche zur Kommunikation mit dem Laserscanner unabdingbar ist. Aus diesen Gründen muss der Notbremsassistent auf eine neue Umgebung portiert werden. Die Portierung umfasst anfangs die Einrichtung einer neuen Simulink-Umgebung, welche die Codegenerierung für ein x86-Linux-System (64 Bit) unterstützt.

Anschließend sind alle hardwaregebundenen Teile des Modells anhand eines neuen Konzeptes auf die neue Umgebung anzupassen. Die Kommunikation über CAN (Einlesen von Sensordaten und Steuerung des Brake-

by-Wire-Systems) wird dabei durch den Einsatz des ROS-Nodes „socketcan_bridge_node“ übernommen. Dieser Node greift auf den Linux-eigenen SocketCAN zu und ermöglicht so die Kommunikation über die angeschlossenen USB-CAN-Interfaces.

Ausblick

Die anfangs erwähnte ToF-Kamera besitzt eine integrierte Objekterkennung und wird momentan für den Notbremsassistenten verwendet. Da dieser jedoch Schwierigkeiten bei der Erkennung von Objekten unter bestimmten Voraussetzungen hat, soll die Sensordatenfusion eine genauere Aussage über die Umgebung ermöglichen.

Der Notbremsassistent ist in seiner jetzigen Form zwar portiert und lauffähig, jedoch ist die Zuverlässigkeit noch ausbaufähig. Durch eine genauere Analyse des Systems sollen die Schwachstellen identifiziert und ein Konzept zur Verbesserung erstellt werden.

-
- [1] CHIP DIGITAL GMBH, 2017. Audi A8 Staupilot im Test: Autonom auf der Autobahn fahren. München: CHIP Digital GmbH [Zugriff am 16.11.2017]. Verfügbar unter http://www.chip.de/news/Audi-A8-Stau-Pilot-im-Autobahn-Test_123142033.html
- [2] WINTZENBURG, Jan Boris, 2014. Ich werde sehr relaxt sein, wenn das Auto alleine fährt. In: stern online, 10.12.2014. Zugriff am 04.11.2017. Verfügbar unter <https://www.stern.de/auto/stern-gespraech-mit-daimler-chef-dieter-zetsche--autonomes-fahren-kommt-bis-2030-3231852.html>
- [3] ZEIT ONLINE GMBH, 2017. Autonomer Bus pendelt in Niederbayern. Hamburg: ZEIT ONLINE GmbH [Zugriff am 16.11.2017]. Verfügbar unter <http://www.zeit.de/mobilitaet/2017-10/deutsche-bahn-autonomes-fahren-bus-oe-pnv-bad-birnbach>

Bildquellen:

- Abbildung 1, 3, 4: Eigene Abbildung
- Abbildung 2: <http://www.robotshop.com/media/catalog/product/cache/1/image/900x900/9df78eab33525d08d6e5fb8d27136e95/h/o/hokuyo-utm-30lx-ew-scanning-laser-rangefinder.jpg>

Entwicklung einer domänenspezifische Programmiersprache zur Regelspezifikation für die Analyse der Semantik von konstanten Daten in C Dateien

Marvin Ahlgrimm*, Jürgen Koch, Timo Vanoni

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Motivation

Bei der Vector Informatik GmbH wird Software für viele verschiedene Steuergeräte mit unterschiedlichen Prozessorarchitekturen und Einstellungen entwickelt. Damit die Konfigurationen für die einzelnen Steuergeräte automatisch erstellt werden, hat Vector Informatik GmbH ein eigenes Tool entwickelt, um C Code für die Steuergeräte zu generieren. Der C Code wird dabei anhand von Compilermakros zusammengestellt und an den Kunden ausgeliefert.

Da es sich bei den Steuergeräten um sicherheitsrelevante Komponenten handelt, muss die Vector Informatik GmbH garantieren können, dass der generierte Code fehlerfrei funktioniert. Bisher wird hierfür ein Sicherheitshandbuch zusammen mit dem Code ausgeliefert. Der Kunde ist dann verpflichtet, die Anmerkungen im Sicherheitshandbuch zu kontrollieren. Das kann je nach Komplexität des ausgelieferten Moduls mühsam und zeitaufwendig sein.

Ein erster Versuch diesen Prozess zu vereinfachen ist es, mit dem Pluginsystem MICROSAR Safe Silence Verifier (MSSV) den Code zu validieren. Mit MSSV lassen sich Regeln spezifizieren, die den generierten Code verifizieren.

Ein MSSV Plugin zu entwickeln ist sehr zeitaufwendig. Daher soll eine domänenspezifische Sprache (DSL) dem Komponentenentwickler dabei helfen, schnell Elemente aus dem C Code zu ermitteln und den semantischen Wert zu verifizieren.

Domäne

Die C Sprache besteht aus verschiedenen Datentypen wie u.a. int, char, array und struct. Das Ziel ist es, dem Entwickler ein Werkzeug an die Hand zu geben, mit dem er die Werte dieser Datentypen aus dem Quellcode ermitteln und deren Werte auf Korrektheit überprüfen kann. Dies beschränkt sich nur auf statische Werte, d.h. Werte die zur Compilezeit bereits bekannt sind.

Dabei soll der Entwickler im Anschluss die

Möglichkeit haben einen Bericht zu generieren. Dieser soll die ermittelten Werte und Bemerkungen in einer lesbaren Form, wie etwa ein HTML/XML Dokument oder eine Textdatei, darstellen.

Programmiersprache

Eine Programmiersprache bietet den Entwicklern ein Werkzeug um IT spezifische Probleme zu lösen. Dabei entwickelt der Entwickler ein Programm, welches auf einer Zielarchitektur ausgeführt wird und somit auch die Befehle des Programmierers.

Im Universum der Programmiersprachen gibt es grob unterteilt den Bereich der Universalprachen (z.B. Java) und der domänenspezifischen Sprachen (z.B. SQL)[1]. Die zweite Gruppe adressiert Probleme einer gewissen Domäne direkter als die Universalprache und kosten den Programmierer weniger Zeit, um ein bestimmtes Problem zu lösen.

Um eine Programmiersprache auszuführen, muss diese zunächst geparkt werden. Dabei werden einzelne Zeichen oder Zeichenketten zu Tokens zusammengefasst. Dieser Vorgang wird lexikalische Analyse genannt[1]. Im nächsten Schritt folgt die syntaktische Analyse, bei der die generierten Tokens auf die richtige Reihenfolge überprüft werden und daraus ein abstrakter Syntaxbaum (AST) generiert wird. Danach kann dieser optional auf semantische Korrektheit überprüft werden. Diese Phase nennt man Analysephase und wird von dem Frontend eines Compilers/Interpreters durchgeführt.

Nach der Analysephase kann der AST entweder von einem Interpreter ausgeführt oder weiter zu Maschinencode umgewandelt werden, welcher auf der Zielplattform ausgeführt werden kann.

Die Zielplattform der DSL, die in dieser Abschlussthesis erstellt wird, ist ein Interpreter. Durch die Ausführung der DSL in einem Interpreter können zur Laufzeit weitere bereits existierende Bibliotheken dazugeschaltet werden, welche notwendig sind um C Elemente aus dem Quellcode zu ermitteln.

*Diese Arbeit wurde durchgeführt bei der Firma Vector Informatik GmbH, Weilimdorf

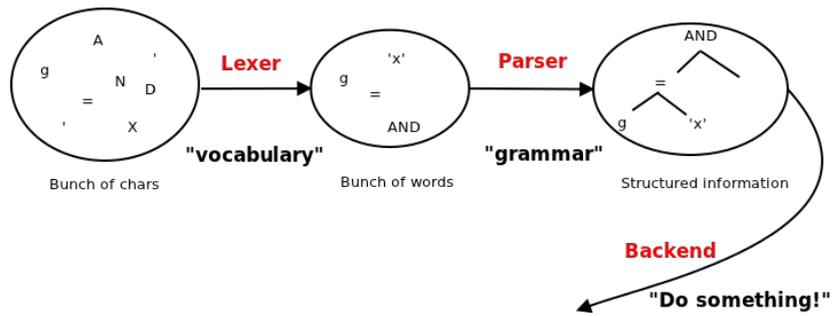


Abbildung 1: Compiler Frontend

Umsetzung

Die zu entwickelnde Sprache soll als Frontend für das Projekt Vector Code Analyzer verwendet werden. Der Code Analyzer ist ein Framework, um statische Codeanalysen durchzuführen. Dabei soll die DSL in das Framework integriert und konfiguriert werden. Die DSL soll einem Entwickler die Möglichkeit geben unkompliziert zu beschreiben, welche C Elemente analysiert werden sollen.

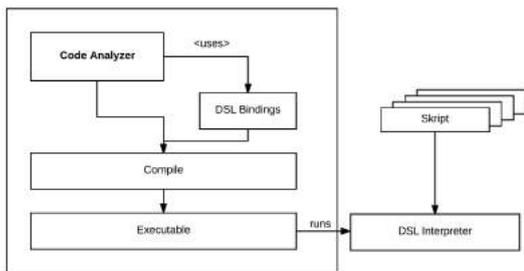


Abbildung 2: Struktur des Code Analyzer

Dabei wird ein DSL Skript über einen Lexer geparsed, der aus den einzelnen Zeichen oder Zeichenketten Tokens erstellt. Die Tokens werden dann von einem Tokenparser geparsed und auf syntaktische Korrektheit überprüft. Der Tokenparser generiert einen abstrakten Syntaxbaum (AST)[2].

Für das AST Objekt wird ein Visitor Pattern implementiert, durch welches es möglich ist, die einzelnen Syntaxelemente mehrfach zu traversieren[3]. Dies ermöglicht es die semantische Analyse durchzuführen. Dabei wird jedes Syntaxelement besucht und die Typen und Operationen des Scripts können ausgewertet werden.

Zum Schluss wird der AST ein letztes mal traversiert und die einzelnen Operationen werden interpretiert.

Der Interpreter besitzt eine Schnittstelle zum Code Analyzer, wodurch man mit Selektionsanweisungen aus der DSL Zugriff auf Elemente in C Dateien erhält. Die gefundenen C Elemente können in der DSL Sprache mit Operatoren und Funktionen auf gewünschte Werte überprüft werden, um zu verifizieren, dass der analysierte Code richtig konfiguriert wurde.

```
// C Code
int a[] = {5, 6, 8};
int x_1 = 1;
int x_2 = 2;
int x_3 = 3;
//-----

# DSL Code
a_array = $a[0..1] # [5,6]

a_array.assert_each(() => {
  e1 >= 5 # e1 ist impliziert definiert
})

x_big = $x_*.filter(() => {
  e1 > 1
})

x_big.each(() => {
  writeTable(varName(e1), val(e1))
})

# Schreibt:
# x_2 | 2
# x_3 | 3
```

Das obige Skript selektiert zuerst das Array aus dem C Code und stellt sicher, dass alle Elemente ≥ 5 sind. Im Anschluss werden über eine Wildcard alle C Elemente selektiert, die mit „x_“ beginnen. Aus der Selektionen werden alle Elemente gefiltert die >1 sind. Das Ergebnis wird dann in dem Bericht als Tabelle dargestellt.

[1] Domain Specific Languages – Martin Fowler

[2] <https://ruslanspivak.com/> – Ruslan Spivak

[3] Design Patterns – Erich Gamma, John Vlissides, Ralph Johnson und Richard Helm

Bildquellen:

- Abbildung 1: <https://www.matthieuamiguet.ch/>
- Abbildung 2: Eigene Darstellung

Entwicklung einer Cloud- und Smartphone basierten IoT-Lösung

Onur Albayrak*, Kai Warendorf, Dominik Schoop

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

„I dont need a hard disk in my computer If I can get to server faster...carrying around these non-connected computers is byzantine by comparison.“ [1] Steve Jobs, Late Chairman and Co-Founder of Apple

Ein Trend, der heutzutage unumgänglich ist und sowohl im privaten als auch im geschäftlichen Bereich immer mehr gefördert wird: Die Auslagerung der Daten in die Cloud und somit das Vermeiden jeglicher Festplatten, wenn man schneller zum Server kommen kann.

Unter **Cloud-Computing** sind die allgemeinen Bereitstellungen sowie Nutzung von IT-Infrastruktur wie z.B. die Speicherung der Daten, Rechenleistung oder der Speicherplatz zu verstehen. Die Idee dahinter soll sein, dass man künftig die IT-Infrastrukturen über ein Rechnernetz zur Verfügung stellt, ohne das diese lokal abgelegt werden müssen. Erforderlich sind hierfür sind sogenannte Schnittstellen und Protokolle um die Bereitstellung und Nutzung der jeweiligen IT-Leistungen zu verwirklichen. [2]

Aktuelle Technik

Anlegen, aufschieben, Knopf drücken, fertig- die Möglichkeiten mit dem Laser-Entfernungsmessgeräte macht das Messen verschiedener Längen und Entfernungen heutzutage besonders einfach. Mithilfe intuitiver Bedienung, lesefreundlicher Hintergrundbeleuchtung, ist das Gerät außerdem in der Lage, Entfernungen oder Flächen auszumessen, indem es einen Laserstrahl bis zur Referenzkante aussendet. Stellt man das Messgerät auf den Boden und zielt die Decke an, muss man den roten Knopf tätigen, wo sofort die Höhe angezeigt wird. Drückt man den roten Knopf erneut, so beginnt man mit einer neuen Messung.

Die neuesten Messgeräte besitzen sogar einen Speicher, wo die Messwerte ebenfalls mit einem Knopfdruck gespeichert werden. [3]



Abbildung 1: Aktuellstes Zamo Entfernungsmessgerät

Einordnung der Bachelorarbeit

Die Entwicklung einer neuen Oberfläche für die Lasermessgeräte von Power Tools Robert Bosch GmbH soll mithilfe einer dynamischen Webseite in der Lage sein, Zeit, Beschreibung, Messwerte, Messwertmethoden, Ausrichtung und die dazugehörigen Sprachaufnahmen in Echtzeit darzustellen ggf. zu verwalten.

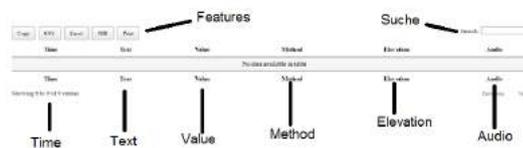


Abbildung 2: Auszug aus der dynamischen Weboberfläche

*Diese Arbeit wurde durchgeführt bei der Firma Power Tools Robert Bosch GmbH, Leinfelden-Echterdingen

Der Grundgedanke hierbei ist, dass man die Messgeräte für eine Baustellendokumentation dem zuständigen Projekt/Bauleiter aushändigt und dieser nach jedem Messwert über die Spracheingabe den dazugehörigen Text für die Beschreibung des Messwertes hineinspricht. Dies hat den Vorteil, dass die bereits gespeicherten Werte auch im Nachhinein den Messungen zugeordnet werden können. Jeder Messwert wird automatisch zeitlich eingeordnet (das aktuellste zuerst) mit dem Server kommunizierend auf die Weboberfläche verschickt. Der Anwender verfügt ebenso über die Rechte, die Textbeschreibung auf der Weboberfläche manuell umzuändern. Zusätzliche Funktionen wie das Kopieren der Messwerte, CSV-Auszug, Excel-Export, konvertieren als PDF-Datei oder das Ausdrucken der Messwerte unterstützen den Bauleiter bei seinen Aufgaben, indem er viel flexibler und zeiteinsparender handeln kann.

Usability Befragung anhand der Minimal-lösung

Die Usability Befragung ist eine quantitative Methode, bei der die Usability mit Hilfe eines Fragebogens ermittelt wird. Dazu werden den Benutzern verschiedene Fragen über die Interaktion und einzelne Funktionen der Buttons bzw. Elemente gestellt. Um dieses Designkriterium umzusetzen, ist es notwendig zu wissen, wie die Weboberfläche auf die Echtzeit der Eingabe vom User reagiert. Damit die Weboberfläche möglichst aktiv genutzt werden soll, wird diese möglichst benutzerfreundlich dargestellt.

Die verschiedenen Befragungskriterien beziehen sich auf die Interaktion des Benutzers, welches basierend auf folgende Kriterien sind: [4]

- Aufgabenbemessenheit
- Erwartungskonformität
- Lernförderlichkeit
- Steuerbarkeit
- Fehlertoleranz
- Individualisierbarkeit

Vor- und Nachteile der Cloud Services

Aus der Sicht von Privatanwendern liegen die Vorteile definitiv darin, dass für die Speicherung von Daten keine eigenen Hardware-Ressourcen benötigt werden. In kurzen Worten: Datenträger wie externe Festplatten oder USB-Sticks werden obsolet und man läuft nicht die Gefahr die Daten zu verlieren oder diese zu beschädigen. Ein weiterer Vorteil ist, dass man jederzeit mit unterschiedlichen Geräten auf die Cloud-Dienste zugreifen kann, egal ob mobil oder stationär. Voraussetzung hierfür wäre natürlich eine bestehende Internet-Verbindung. Aus Unternehmenssicht betrachtet, unterstützt die Cloud Nutzung vor allem bei der Einsparung der teilweise erheblichen Investitionen für Dienste, die in der Cloud bei externen Dienstleistern angemietet werden können. Dabei spart man sowohl von den Personalkosten, als auch von den Hardware-Ressourcen. Die Nachteile eines Unternehmens dagegen sind beispielweise die Abhängigkeit zum Anbieter, der sich vielleicht unzureichend um Kunden kümmert oder zu wenig Kapazitäten aufbringt. Die zweite Frage wäre, wie der Anbieter den Umgang mit empfindlichen und unternehmensbezogenen Daten umgeht. Weiterer Nachteil kann sein, dass man die unterschiedlichen Arbeitsprozesse, die vom Cloud-Anbieter fest vorgeschrieben sind, den zur Verfügung gestellten Softwarelösungen anpassen muss. [5]

[1] AZ Quotes. <http://www.azquotes.com/quote/1250846>. Abrufdatum: 15.11.2017

[2] Microsoft Azure. <https://azure.microsoft.com/de-de/overview/what-is-cloud-computing/> Abgerufen am: 15.11.2017

[3] Bosch Presse. <http://www.bosch-presse.de/pressportal/de/de/laser-entfernungsmesser-zamo-von-bosch-43106.html>. Abrufdatum: 15.11.2017

[4] usability.de GmbH & Co. KG. <https://www.usability.de/leistungen/methoden/onsite-befragung.html>. Abrufdatum: 16.11.2017

[5] Jeanine Wein(2016): Vor- und Nachteile von Clouds, Projektreferentin der Stiftung MedienKompetenz Forum Südwest, 2016.

Bildquellen:

- Abbildung 1: cdn.idealo.com/folder/Product/5095/2/5095223/s1_produktbild_gross/bosch-zamo-ii.png
- Abbildung 2: www.datatables.net

Implementierung von Big Data Use Cases in RB Analytic Platform zur Entwicklung der Web Based Validation

Hilal Altintas*, Gabriele Gühring, Reiner Marchthaler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Big Data Analytics Infrastructur

Messungen aus Feldtestfahrzeugen werden zur weiter Entwicklung von bestehenden Systemen durchgeführt. Diese Messdaten können zur Analyse der Lebensdauer oder auch weitere Kundenanforderungen genutzt werden. Im Laufe der Digitalisierung und die hohe Messdatenverfügbarkeit von Feldtestfahrzeugen werden neue Validierungsstrategien und -methoden in den Fachbereichen von Diesel Systemen entwickelt. Domain Experten stellen Use Cases aus den verschiedenen Fachbereichen wie zum Beispiel der Denoxtronic- und Einspritzsystementwicklung zur Verfügung. Ziel der Bachelorarbeit ist es diese gelieferten Use Cases in die neu entstehende Big Data Landschaft von Bosch zu implementieren. Der standardisierte Datenimport und die Aufbereitung dieser wöchentlich ankommenden Daten erfolgt über die EADM in einen Hadoop Cluster auf der RB Analytics Plattform. Ein Hadoop-Cluster wird genutzt um die Speicherung und Analyse von großen Mengen an unstrukturierter Daten in einer verteilten Rechnungsumgebung entwickelt wurde. Die Daten werden an ihrem Ablageort verarbeitet. Ein Hadoop Cluster arbeitet nach dem bekannten Master-Slave-Prinzip, in dem die Rolle des Masters von der „Namenode“ übernommen wird. Es sorgt für die Verteilung der Blöcke und die mehrfache Replizierung dieser Blöcke. In welche Blöcken die einzelnen Daten zerlegt sind und wo diese Blöcke abgelegt wurden weiß der Namenode. Dieser „Name-Node“ Knoten ermöglicht es Metadaten zu File-System, Verzeichnisstrukturen und Dateien zu verwalten. Die Daten werden auf den Daten-Nodes geschoben, das hier übernimmt es die Rolle des Slaves.

Big Data

Die Evaluierung, Auswertung von Daten kann in vielen Softwares durchgeführt werden. Bei der Verarbeitung von Big Data, jedoch können Software Tools wie zum Beispiel von Matlab an ihre Grenzen kommen. Big Data kann jenes Dataset sein was auf Grund der Größe

der Daten nicht mehr funktioniert, d.h. große Datenmengen zu erfassen, verarbeiten und zu analysieren würden standard-Tools nicht mehr schaffen. Die Daten werden innerhalb der Arbeit in die verfügbaren Data Analytics Tools. Diese Tools werden nach Evaluierungskriterien wie Performance, Flexibilität, Bedienbarkeit bei ihrer Nutzung für Data Mining, und Visualisierung bewertet werden. Die Robustness Toolbox ist eine Entwicklung der ETAS GmbH, dass innerhalb der Arbeit eine Daten Visualisierung und für schnelle Report Erstellung ermöglicht, genutzt. Der Vorteil liegt darin das ein Zugriff auf die Daten auf der EADM möglich.

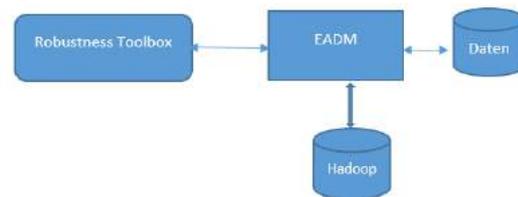


Abbildung 1: Datenimport auf der EADM

Use Case

Ein Use Case ist ein Anwendungsfall, das das Verhalten eines Systems beim Versuch das Ziel zu erreichen, beschreibt. Ein Use Case aus der Denoxtronic- und Einspritzsystementwicklung wird innerhalb der Arbeit behandelt. Hier beschreibt das Use Case den Fall, dass eine Kristallisation an der Spritzlochscheibe entsteht und somit ein Unterdruck erzeugt wird. Die SCR Technologie von Bosch besitzt mehrere Generationen an diesen Denoxtronic-Systemen, die für die Reduktion von Nox-Emissionen in Kraftfahrzeugen eingesetzt wird. Mit dieser Technologie gelingt es die durch den Verbrennungsmotor entstandenen Emissionen in Wasser und Stickstoff umzuwandeln. Mit dieser Technologie gelingt es vielen Automobilherstellern die Emissionsgrenzwerte in den verschiedenen Ländern einzuhalten. Jedoch bringt diese Technologie auch eine Schwierigkeit in ihrer

*Diese Arbeit wurde durchgeführt bei der Firma Bosch GmbH, Stuttgart-Feuerbach

Eigenschaft mit ab einer Temperatur von minus 11 Grad zu kristallisieren. Durch diese Kristallisation kann ein Unterdruck entstehen, so dass das Dosierventil blockiert wird. Um diesen Vorgang zu analysieren und zu beobachten in welchem Messdaten dieser Unterdruck auftaucht oder bei welchen Fahrzeugen kann Data Mining die richtige Strategie sein.

Data Mining

Durch Data Mining ermöglicht in vielen Fällen ein großes Datenvolumen zu bewältigen und „das Graben nach Informationen und nutzbaren Wissen“ aus einem Datenberg. Die Interessen des Data Mining sind Techniken zur Beschreibung und Suche von Mustern zu finden um daraus Vorhersagen zumachen. Hierzu kommen aus verschiedenen Bereichen wie zum Beispiel die Statistik und des Maschinen Lernens. Ein Projekt in der Data Mining stattfindet durchläuft verschiedene Prozesse, wie das Vorbereiten der Daten. Die Vorbereitung enthält die Datensammlung, Datenauswahl und die Planung. Im Weiteren wird eine Vorverarbeitung der Daten geleistet bevor es mit der Analyse beginnen kann. IN der Vorverarbeitung können Transformationen für die zu Unterstützenden Formatierungen gemacht werden. Innerhalb der Datenanalyse können Korrelationsanalysen, Regressionsanalysen und Methoden wie Clustering durchgeführt werden. Der letzte Prozess ist die Nachbereitung dieser analysierten und verarbeiteten Daten. Ob eine Methode sich für das Use Case geeignet hat könnte aus den Ergebnisse interpretiert werden und dokumentiert werden. Die Prognose, Visualisierung, Klassifikation, das Clustering und die Klassifikation sind die wichtigsten Methoden des Data Minings.

Clustering

Clustering Methoden eignen sich für großen Datenmengen, die auf Korrelation analysiert werden soll. Korrelation kann zwischen zwei Messgrößen wie zum Beispiel die Temperatur und dem Druck wie in dem Kristallisation Use Case beschrieben. Eine Korrelation ob linear oder nicht kann durch eine Clusteranalyse ermittelt werden. Bei einer Clustering werden Datensätze in Gruppen eingeteilt, und eine Berechnung dieser Gruppierung wird als Clusteranalyse bezeichnet. In solch einer Darstellung können außerhalb der in den gebildeten Clustern liegende Punkte als Ausreißer gesehen werden und somit diesen Ausreißer analysiert werden.

K-Means-Algorithmus: Dieses Algorithmus ist in vielen Tools verfügbar und wird zur Clusteranalyse verwendet. Dieses Algorithmus findet die Zentren der Cluster und kann dann die Objekte gruppieren.

Input:

- n:Anzahl der Cluster
- D:Data set das N Objekten angehört

Output:

- k Cluster

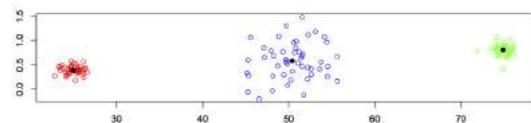


Abbildung 2: K-Means Clustering

[1] Thomas A. Runkler Data Mining – Modelle und Algorithmen intelligenter Datenanalyse

Bildquellen:

- Abbildung 1: Selbst erstelltes Diagramm für die Bachelorarbeit
- Abbildung 2: cowlet.org/2013/12/23/understanding-data-science-clustering-with-k-means-in-r.html

Entwicklung einer Edge-Cloud basierten Applikationsplattform für zeitkritische Verkehrsüberwachung im Projekt InVerSiV

Ali Askar*, Jürgen Koch, Reiner Marchthaler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Das Vorantreiben von neuen Technologien im Hinblick auf Fahrassistenzsysteme und automatisiertes Fahren spielt heutzutage bei den Automobilherstellern eine bedeutende Rolle. Hierzu werden Fahrzeuge, zum Erkennen von kritischen Situationen, mit weitreichender Sensorik ausgestattet. Jedoch reicht der fahrzeugseitige Einsatz dieser Sensoren für ein vollständiges Erkennen von Gefahrensituationen nicht aus.

Projekt InVerSiV

Im Forschungsvorhaben des Projektes InVerSiV soll die Komplexität einer belebten Megastadt in Zukunft für automatisiertes Fahren beherrschbar gemacht werden. Dabei sollen die Gefahren für unterschiedliche Verkehrsteilnehmer ohne Sensorik rechtzeitig erkannt und gezielt behandelt werden. Hierzu wird die Sensorik in Fahrzeugen mit den straßenseitigen Sensorplattformen, die zur Umfelderkennung ausgestattet sind, verknüpft [1]. Das Projekt InVerSiV wird aus Mitteln des Europäischen Fonds für regionale Entwicklung (EFRE) gefördert (s. Abbildung 1).



Abbildung 1: Logos InVerSiV & Projektförderer

Ziel dieser Arbeit

Das Ziel dieser Bachelorarbeit ist es, durch das Entwickeln einer Edge-Cloud basierten Applikationsplattform, eine dezentrale Datenverarbeitung im Projekt InVerSiV nach dem Early Demonstrator Konzept zu ermöglichen. Auf Basis der Messdaten, sollen Verkehrsteilnehmer bei einer Gefahrensituation, z.B. bei einer möglichen Kollision an einer Kreuzung, vorzeitig gewarnt werden.

Early Demonstrator

Der Early Demonstrator zeigt die erste Integration des Gesamtkonzeptes im Projekt InVerSiV. Dabei werden zwei Sensorikplattformen in einem abgeschlossenen Testfeld (Universitätsstraße Dortmund) an einem Traversensystem aufgehängt (vgl. Abbildung 2). Zur Erprobung werden zwei Fahrzeuge in Einsatz genommen, die mit unterschiedlichen Geschwindigkeiten an einer Kreuzung vorbeifahren.



Abbildung 2: Aufbau Early Demonstrator

Architektur

Die Applikationsplattform verwendet ein Raspberry Pi, der als Internet of Things (IoT) Gateway dienen soll. Das in der Java Virtual Machine (JVM) laufende Framework Vert.x ist der Kern der Applikationsplattform und soll eine asynchrone Datenverarbeitung ermöglichen. Der auf Vert.x basierte User Datagram Protocol (UDP) Server empfängt Nachrichten aus den Sensorikplattformen, die im DataHandler für eine Kartendarstellung verarbeitet werden. Der Vert.x Eventbus sendet nur die nötigen Informationen asynchron an die Client-Anwendung, die von der adesso AG zur Verfügung gestellt wurde. Die Architektur der Edge-Cloud basierten Applikationsplattform ist in der Abbildung 3 dargestellt.

*Diese Arbeit wurde durchgeführt bei der Firma adesso AG, Stuttgart

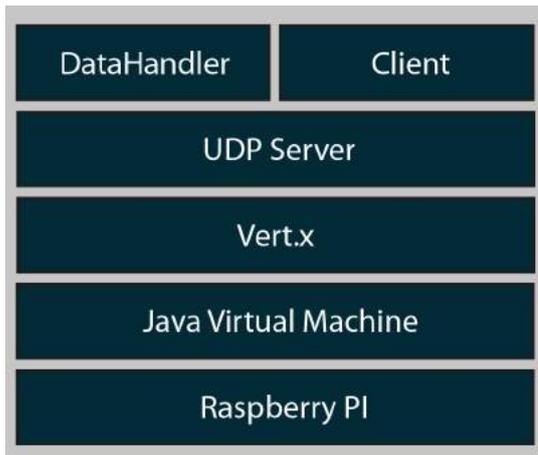


Abbildung 3: Architektur der Applikationsplattform

Simulation

Erprobungen auf dem Testfeld während der Entwicklung wären aufgrund der Ortslage von großem Zeitaufwand. Daher werden die Umfeldsensoren und deren Messdaten mit MATLAB simuliert. Die simulierten Daten werden per UDP-Verbindung zur Applikationsplattform gesendet.

Positionstransformation

Jede Sensorikplattform, die an dem Traversensystem aufgehängt ist, bildet ein eigenes kartesisches Koordinatensystem. Die Position eines im Umfeld erkannten Objektes bezieht sich auf den euklidischen Abstand zwischen der Plattform und dem Objekt. Um die Koordinaten auf ein gemeinsames Weltkoordinatensystem zu bringen, werden dazu Rotationsmatrizen berechnet.

Kartendarstellung

Die Geokoordinaten der ortsfesten Sensorikplattformen sind im Header einer UDP-Nachricht enthalten und werden in Earth Centered Earth Fixed (ECEF) Koordinaten transformiert, woraus dann die x-, y- und z-Koordinate des Fahrzeugs ermittelt wird. Nach Berechnung der Orientierung des Traversensystems bezüglich Norden, kann die berechnete Position dementsprechend rotiert wer-

den. Nach der Rotierung wird die Position in geographischen Koordinaten umgerechnet. Anschließend sendet die Applikationsplattform eine asynchrone Nachricht an die Client-Anwendung, die die geographische Lage eines Fahrzeugs auf einer Karte punktförmig darstellt (vgl. Abbildung 4).

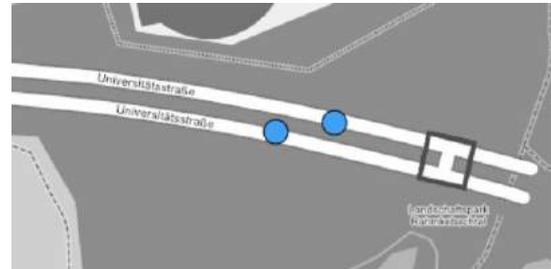


Abbildung 4: Kartendarstellung in der Client-Anwendung

Kollisionsberechnung

Fahrer, die mit ihren Fahrzeugen an der Kreuzung vorbeifahren und in eine Kollision involviert werden könnten, müssen vorzeitig gewarnt werden. Da sich die geplante Fahrtrichtung ohne fahrzeugseitigen Einsatz von Sensoren schwer ermitteln lässt, wird die Kollisionssituation in mehrere Warnungsstufen unterteilt. Zum Festlegen der Warnungsstufen wird z. B. nach Berechnen der Time-to-Collision (TTC) berücksichtigt, ob die Geschwindigkeit des Fahrzeugs absinkt. Ein größer werdender Gierwinkel würde aussagen, dass das Fahrzeug an der Kreuzung abbiegt.

Ausblick

Simulierte Daten können stark von der Realität abweichen. Daher werden im weiteren Verlauf der Bachelorarbeit die in der Erprobung aufgezeichneten Daten ausgewertet. Darauf folgend werden die Berechnungen auf echten Daten getestet und ggf. angepasst. Zudem ist noch geplant, für fortlaufende Verbesserung der Messdaten eine Kalman-Filter Variante zu verwenden. Es ist sehr wichtig, dass bei der Übertragung von Daten keine Pakete verloren gehen. Daher wird das Systems auf Verzögerungszeiten überprüft.

[1] <http://www.inversiv.de>

Bildquellen:

- Abbildung 1: <https://www.kn.e-technik.tu-dortmund.de/cms/de/Forschung/Projekte/Laufend/Nationale-Projekte/InVerSiV.html>
- Abbildung 2, 3: Eigene Abbildung
- Abbildung 4: Eigene Abbildung (Karte von Openstreetmap)

Synchronisation von Daten in verteilten Systemen

Carlo Babo und Kevin Thomas*, Reinhard Schmidt, Andreas Rößler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Motivation

Der reibungslose Datenaustausch zwischen Auftraggeber und Lieferant gewinnt in der heutigen Zeit immer mehr an Bedeutung. Sowohl die stetig wachsende Menge an Daten (verursacht durch eine zunehmende Digitalisierung), als auch die immer häufiger auftretende Vergabe von Aufträgen an externe Unternehmen, machen einen schnellen und flüssigen Datentransfer zwischen unterschiedlichen Systemen unverzichtbar. Im Bereich der Fahrzeugherstellung stehen Fahrzeughersteller und Lieferant von Fahrzeugkomponenten im ständigen Datenaustausch. Hierfür nutzen Sie unterschiedliche Lifecycle Management Systeme. Diese werden in der Softwareentwicklung von der Planung bis zur Fertigstellung eingesetzt und finden im Anforderungsmanagement, Änderungsmanagement, Qualitätsmanagement, Releasemanagement, sowie in der Anwendungsimplementierung und der Softwarepflege ihre Verwendung [1]. Über gibt ein Auftraggeber seinem Lieferanten eine Anforderungsliste oder äußert einen Änderungswunsch über einen Änderungsauftrag, erwartet dieser den Status der Bearbeitung im eigenen System nachzuverfolgen. Um diesen Austausch zu erfüllen, müssen die Systeme der Auftraggeber und die der Lieferanten kontinuierlich synchronisiert werden [2]. Für diesen Zweck soll im Rahmen dieser Abschlussarbeit ein Prototyp für eine eigene Softwarelösung entwickelt und im Anschluss mit der käuflichen Lösung von Tasktop verglichen werden, um die Frage nach dem „Make or Buy“ zu klären.

Verwendete Technologien

Der Prototyp für die betriebseigene Softwarelösung wird unter der Verwendung folgender Technologien entwickelt:

Eclipse IDE: Als Entwicklungsumgebung wird die Eclipse IDE Oxygen für Java EE Entwickler eingesetzt.

Java: Die API-Schnittstellen, der Workflow-Manager, der Message Broker und der Datenkonvertierer werden in der Programmiersprache Java entwickelt.

Maven: Das Build and Dependency Management Tool der Apache Software Foundation vereinfacht die Entwicklung von Java-Anwendungen im Team. Die libraries werden einmalig als dependency in die pom.xml eingebunden. Die Software lädt diese selbstständig herunter und bindet Sie automatisch in das Projekt ein. Hierdurch wird aufwendiges und mehrmaliges Einbinden von Bibliotheken vermieden [3].

Spring AMQP: Die Abkürzung AMQP steht für Advanced Messaging Queue Protocol. Das AMQ-Protokoll ist ein open standard Protokoll. Dies bedeutet sie gewährleistet eine verlässliche Kommunikation zwischen zwei oder mehreren Organisationen. Dabei steht der schnellstmögliche Transport von Nachrichten mit wertvollem Inhalt im Fokus. Außerdem bietet das AMQ-Protokoll Sicherheit, Zuverlässigkeit und Interoperabilität [4].

MongoDB: MongoDB ist eine NoSQL-Datenbank. Sie basiert auf einem Dokumentdatenmodell mit dynamischen Schemata und kann JSON-ähnliche Dokumente verwalten [5]. Im Prototyp agiert sie als Zwischenspeicher und Puffer für große Mengen an Daten.

RabbitMQ: RabbitMQ ist ein Message Broker von Rabbit Technologies Ltd. und implementiert das oben beschriebene AMQP. Seine Aufgabe besteht darin, Nachrichten von einem oder mehreren Endpunkten zu empfangen und schnell, sowie zuverlässig an weitere Endpunkte zu verteilen. Im Prinzip kann man RabbitMQ als Poststelle verstehen. Nachrichten werden von einem Endpunkt in den Briefkasten geworfen und ein Postmann überbringt diese auf schnellstem Wege zu einem oder mehreren anderen Endpunkten [6].

Apache Camel: Basierend auf Enterprise Integration Patterns (EIP) ist Apache Camel eine Routing- und Konvertierungsengine der Apache Software Foundation. Die Software erlaubt dem Programmierer über vordefinierte Regeln zu bestimmen, welche Nachrichten angenommen und wie diese transformiert und konvertiert werden. Somit kann die Engine dazu verwendet werden, Daten in System A von einem beliebigen Format (z.b. XML), passend in ein anderes Format (z.b. JSON) für ein System B zu konvertieren [7].

*Diese Arbeit wurde durchgeführt bei der Firma Robert Bosch GmbH, Schwieberdingen

Workflow-Manager

Der Kopf des Systems ist der Workflow-Manager. Dieser beinhaltet die Logik und bestimmt welche Komponente welche Aufgaben erledigt. Wie alle anderen Komponenten, ist der Workflow-Manager mit dem RabbitMQ Broker verbunden und kommuniziert über Message Queues mit den Consumern. Veränderungen von Daten auf Seiten eines Systems müssen erkannt und auf das verbundene Partnersystem übertragen werden. Hierdurch erreicht man Synchronizität. Dafür stehen dem Workflow-Manager mehrere Befehle zur Verfügung. Bei dem Aufruf eines Befehls, wird ein Request in Form einer RabbitMQ Nachricht an den Exchange geschickt. Dieser ordnet die Nachricht der richtigen Queue zu und wartet auf die Verarbeitung durch die Konnektoren oder der Konvertierungskomponente. Bei der Ausführung des Befehls erwartet der Workflow-Manager eine Response. Falls kein Fehler auftritt, erhält dieser einen positiven Response. Tritt ein Fehler auf, wird ein negativer Response mit Fehlercode an den Workflow-Manager geroutet. Von dort aus wird entschieden, ob der Befehl wiederholt verschickt wird oder eine andere Lösung zu berücksichtigen ist. Um präventiv gegen Datenverlust vorzugehen, speichert man vor der Bearbeitung alle relevanten Daten in die MongoDB. Zur Identifizierung erhält der Manager den Unique Key als ID über den Response. Auf diese Weise bleiben dem Manager die Orte der gespeicherten Nachrichten erhalten und dieser kann zur Weiterverarbeitung leichter auf die Elemente zugreifen.

Die Folgenden Befehle werden in den Workflowmanager implementiert:

Read: Bei einem Read-Befehl werden alle Daten von einem System aufgelistet.

ReadLastChanges: Dieser Befehl ist eine Abwandlung des Read-Befehls. Über ihn werden die zuletzt geänderten Daten aufgelistet. Über Parameter ist es zudem möglich weitere Filter zu setzen.

Merge: Bei diesem Befehl wird eine abgeänderte Datei auf das Partnersystem hochgeladen. Diese Befehle sind für die API-Schnittstellen von Bedeutung.

Convert: Da beide Systeme mit Daten in verschiedenen Formaten arbeiten, müssen diese passend umgewandelt werden. Mithilfe des Convert-Befehls wird dies ermöglicht. Die Datei wird in das ausgewählte Format konvertiert und abgespeichert. Diesen Befehl erhält die Konvertierungskomponente des Prototyps.

API-Schnittstellen

Die Verschiedenen Tools, die miteinander synchronisiert werden, sind über API-Schnittstellen mit dem Prototyp verbunden. Zu den zentralen Aufgaben der Konnektoren gehören das Auslesen von Daten aus den Tools, das Schreiben der Daten in die MongoDB und das Hochladen von neuen oder konvertierten Daten in die Tools. Hierfür erhalten die Schnittstellen Befehle vom Workflow-Manager, so genannte Requests. Während dieser im Gesamtsystem als Kopf betrachtet wird, agieren die Konnektoren als Hände, die die Vorgaben des Kopfes ausführen. Erhält eine Schnittstelle einen passenden Request, antwortet dieser dem Workflow-Manager mit einem positiven Response. Andernfalls wird ein negativer Response gesendet. Hierdurch können Irrläufer rechtzeitig erkannt und behandelt werden. Außerdem besteht die Möglichkeit über Konnektoren Daten nach bestimmten Kriterien aus den Tools auszulesen. Dazu fügt der Workflow-Manager einem Befehl einen Parameter hinzu und man liest über die Schnittstellen nur die zuletzt veränderten Daten oder Daten aus einem bestimmten Zeitraum aus. Können die Konnektoren Daten erfolgreich in die MongoDB zwischenspeichern, wird eine weitere response an den Workflow-Manager gesendet, die zusätzlich zu der positiven Antwort auch die IDs der MongoDB-Einträge enthält.

Datenkonvertierung

Zur Konvertierung der Daten wird in der betriebseigenen Softwarelösung eine Komponente auf der Basis von Apache Camel implementiert. Diese hat ausschließlich die Aufgabe, Daten über ein Mapping von einem Format in ein anderes zu konvertieren und somit für das Hochladen in beliebige Systeme passend zu machen.



Abbildung 1: Die Formatkonvertierung in der Konvertierungskomponente wird mit Apache Camel realisiert.

Hierfür erhält der Datenkonvertierer vom Workflow-Manager einen request mit einem Convert-Befehl im Header. Das gewünschte Format wird diesem als Parameter übergeben. Die zu konvertierende Nachricht wird im Body eingefügt. Ähnlich den API-Schnittstellen sendet der Datenkonvertierer dem Workflow-Manager eine positive Response bei erfolgreicher Verarbeitung und eine weitere response (inklusive der IDs der Datenbankeinträge) bei erfolgreichem Eintragen der konvertierten Nachricht in die MongoDB. Für den Prozess der Datenkonvertierung stellt Apache Camel mehrere Funktionen zur Verfügung. Durch diese übernimmt Camel die Konvertierung automatisch und realisiert diese selbständig im Hintergrund. Als Ergebnis erhält man eine zweite Nachricht im neuen Format, welche zur Weiterverarbeitung in die MongoDB gespeichert wird.

Kommunikation

Für den Prototyp ist eine robuste Lösung beim Datenaustausch wichtig. Selbst bei erhöhter Belastung muss dieser Aufgaben zuverlässig erledigen. Die Herausforderung im Bereich Webentwicklung ist, auch bei rechenintensiven Prozessen den User schnell mit Inhalt zu bedienen. Mithilfe von http request und synchronen Sprachen wie PHP wird dies realisiert. Jedoch kann eine steigende Nutzerzahl zu Problemen führen. Aus diesem Grund wird für die betriebseigene Softwarelösung der open Source AMQP Broker RabbitMQ verwendet. Dieser unterstützt die asynchrone Kommunikation und bleibt auch bei höchster Belastung hochverfügbar. Mit asynchron ist der zeitlich versetzte empfang von Nachrichten

gemeint. Dadurch vermeidet man das Blockieren der Prozesse. Die Nachrichten werden in einer Warteschlange eingereiht und nacheinander abgearbeitet. Daten wie z.B. kleinere Textdateien müssen zunächst in eine Binärdatei umgewandelt werden. Diese werden in den Body der Nachricht eingefügt und über das Netzwerk verschickt [8].

Mithilfe der Spring AMQP API ist eine Anbindung von RabbitMQ an eine Java-Anwendung möglich. Im Folgenden werden die Aufgaben und Funktionen eines RabbitMQ Message Brokers erklärt. Das AMQP erzeugt eine Verbindung zwischen den Clients und dem Message Broker. Das Ziel ist die Interaktion mehrerer Systeme, die miteinander unabhängig von deren innerer Struktur kommunizieren. Hierfür kommen die oben beschriebenen Konnektoren zum Einsatz. Dieser Prozess wird für das ganze Enterprise Messaging standardisiert. Dabei werden die Routing Operationen und die Art der Nachrichtenverarbeitung definiert.

Das AMQP - Modell besteht aus mehreren Komponenten die Nachrichten empfangen, transportieren, speichern und in die Warteschlange einreihen.

Abbildung 2 beschreibt den Prozess der Nachrichtenübermittlung von einem Publisher zu einem Consumer. Der Producer schickt seine Nachricht an den RabbitMQ Server. Dort empfängt der Exchange die Nachricht und routet diese an die ausgewählte Message Queue. Hier werden die Nachrichten in eine Warteschlange eingereiht und warten auf das Auslesen und die Weiterverarbeitung vom Consumer.

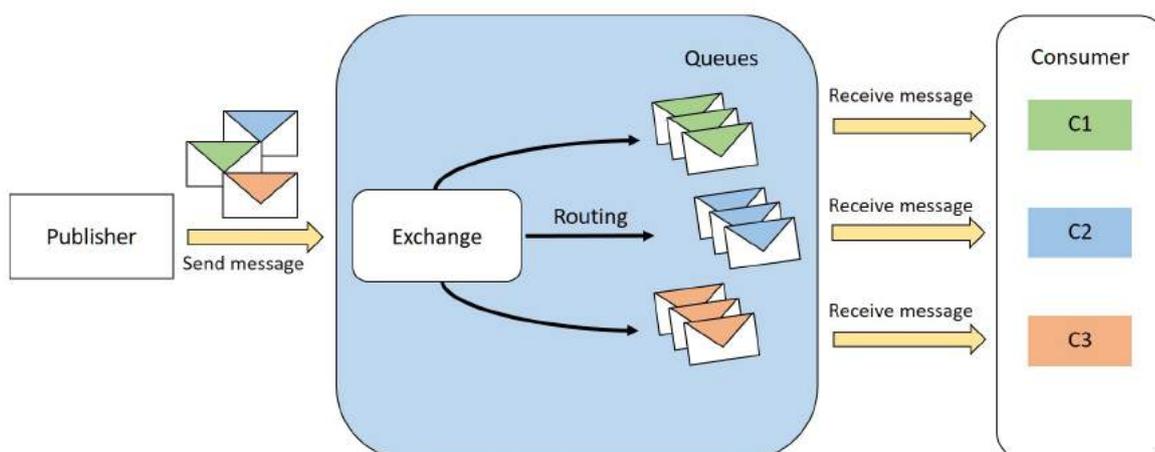


Abbildung 2: Funktionsweise des RabbitMQ Brokers im in der betriebseigenen Softwarelösung

RabbitMQ bietet mehrere Exchange Typen an, die das Routingverhalten definieren. In diesem Projekt wurde der Headers Exchange verwendet. Über den Header werden wichtige Informationen an den Consumer übergeben. Darin wird beschrieben welcher Befehl die verschiedenen Komponenten ausführen.

Headers Exchange: Jede RabbitMQ Message hat einen reservierten Speicher für Header Daten. Um auf diesen Bereich zuzugreifen, muss man die Properties einer RabbitMQ Message direkt ansprechen. Der Header wird in Form einer Key - Value Map eingefügt. Bei einem Headers Exchange werden die Routings Keys (falls angegeben) nicht berücksichtigt. Routing Keys sind zusätzliche Informationen die dem Exchange vorgeben, an welche Message Queue die Nachricht geroutet wird. Der Exchange liest die Header Daten einer Nachricht aus und leitet diese an die richtige Message Queue weiter.

Ausblick

Im weiteren Verlauf der Abschlussarbeit soll die betriebseigene Synchronisationslösung mit einer käuflichen Lösung verglichen werden. Hierfür wird die Synchronisationslösung der Firma Tasktop hinzugezogen. Kann der Prototyp zu betriebseigenen Zwecken Tasktop ersetzen und genauso effizient arbeiten? Wichtige Punkte wie Robustheit, Skalierbarkeit und Verfügbarkeit stehen besonders im Fokus. Auch die Frage nach dem Preis-Leistungsverhältnis der käuflichen Lösung soll analysiert werden. Am Ende soll darüber eine Entscheidung getroffen werden, ob es sich lohnt in eine käufliche Lösung zu investieren oder ob die billigere betriebseigene Variante ausreicht.

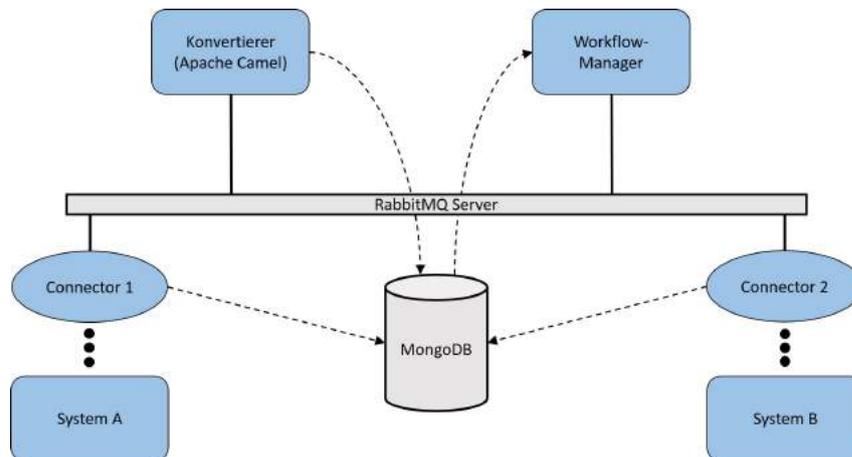


Abbildung 3: Architektur der betriebseigenen Softwarelösung mit allen Komponenten

[1] https://business-services.heise.de/software/enterprise-software/beitrag/einfuehrung-in-application-lifecycle-management-3027.html?tx_hbs_pi1%5Baction%5D=download&cHash=b0493e3b064caed87b0eb59406286914

[2] OnePager_BScThema Laszlo Pirooska

[3] <http://jukusoft.com/2016/10/24/tutorial-eine-einfuehrung-in-maven-teil-1/>

[4] <https://www.mongodb.com/de>

[5] <https://www.rabbitmq.com/tutorials/tutorial-one-java.html>

[6] <https://www.rabbitmq.com/protocol.html>

[7] <https://livebook.manning.com/#!/book/camel-in-action-second-edition/chapter-1/v-14/9>

[8] www.orkork.de/development/was-ist-eigentlich-rabbitmq

Bildquellen:

- Abbildung 1: Icons designed by Smashicons from Flaticon (www.flaticon.com) In Anlehnung an: <https://livebook.manning.com/#!/book/camel-in-action-second-edition/chapter-1/v-14/7>, Abrufdatum: 23.11.2017
- Abbildung 2: In Anlehnung an: <https://www.rabbitmq.com/tutorials/amqp-concepts.html>, Abrufdatum: 23.11.2017
- Abbildung 3: Eigene Darstellung

Konzept für die Integration pneumatischer und elektrischer Regler in eine Softwarearchitektur

Frieder Baumgratz*, Werner Zimmermann, Reiner Marchthaler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Die Firma Festo AG & Co. KG beschäftigt sich mit der Entwicklung von pneumatischen und zunehmend auch elektrischen Lösungen für industrielle Anwendungen. Damit ist sie in dem Bereich der Automatisierung tätig. Eines der neuesten Produkte der Firma Festo ist das so genannte *Motion Terminal VTEM*. VTEM ist eine Abkürzung für *Ventilbaugruppen mit Piezoventilen in der Baureihe M*. In einem solchen Ventil sind vier Piezo-Vorsteuerventile verbaut, die dazu dienen, Membran-Sitzventile exakt und proportional zu steuern. Die vier Membran-Sitzventile kann man getrennt ansteuern.

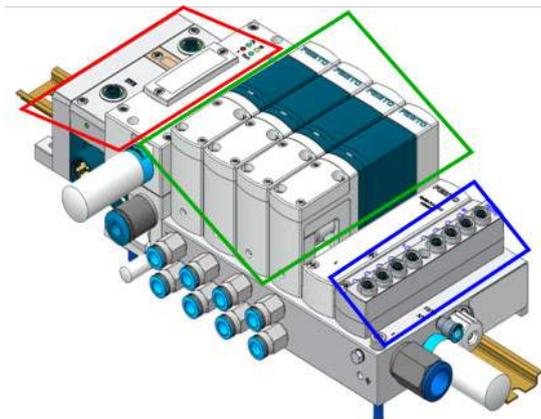


Abbildung 1: Darstellung einer Ventilinsel

Das Motion Terminal VTEM stellt so genannte *Motion-Apps* zur Verfügung. Diese Motion-Apps beinhalten diverse Bewegungsaufgaben, die auf einem Ventil die vorgefertigten Aufgaben erledigen. Teil dieser Motion-Apps sind Regelstrukturen, die die Aufgabe haben, die Drücke, die von einem Ventil bereitgestellt werden, zu regeln. Bei diesem Vorgang kann nicht nur ein Ventil, sondern eine ganze *Ventilinsel* (siehe Abbildung 1) verwendet werden. Unter einer Ventilinsel versteht man einen Zusammenschluss von mehreren Ventilen, sowie einem CPX-Modul. Mit einem „CPX [ist man] offen für viele Steuerungen und Endanwender-Spezifikationen sowie alle ty-

pischen digitalen und analogen E/A-Modulen“ [1]. Unter CPX versteht man ein *Modulares elektrisches Terminal*, auch *Controller mit Feldbusanbindung* genannt.

Die in Abbildung 1 dargestellte Ventilinsel besteht aus mehreren Komponenten. Der blau umrahmte Teil sind externe Anschlüsse wie zum Beispiel Drucksensoren. Der grün umrahmte Teil sind die eigentlichen Ventile. Hier sind vier Ventile verbaut. Die jeweilige Motion-App wird in dem rot umrahmten Teil (der CPX) ausgeführt.

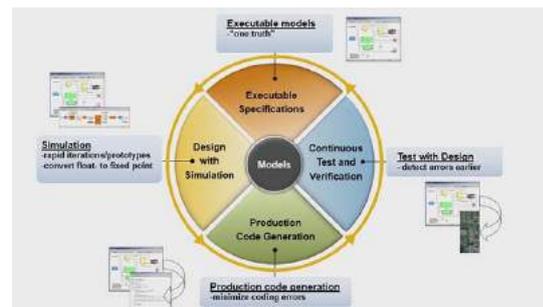


Abbildung 2: Entwicklungszyklus mit Matlab und Simulink

Die Motion-Apps sind in Matlab/Simulink entworfen. Dies hat den Vorteil, dass man eine Simulation durchführen und die entwickelten Regelstrukturen verifizieren kann. Unter anderem soll die automatische Codegenerierung im Rahmen der Bachelorarbeit untersucht werden. Mit Hilfe der automatischen Codegenerierung soll C++-Code erstellt werden, der in eine bestehenden Software integriert werden kann. Der Entwicklungszyklus für die Entwicklung mit Matlab/Simulink ist in Abbildung 2 dargestellt und besteht aus den Schritten:

- Produktspezifikationen
- Entwicklung unter der Verwendung von Simulationen
- Automatische Codegenerierung
- Kontinuierliches Testen und Verifizieren

*Diese Arbeit wurde durchgeführt bei der Firma Festo AG & Co. KG, Esslingen am Neckar

Aktuell werden die Motion-Apps als Programmable Logic Controller (PLC) Code generiert und dann in eine Software integriert.

Des Weiteren soll die Möglichkeit untersucht werden, mit dem Embedded Coder C++-Code zu generieren und in eine andere Software zu integrieren. Dabei spielt der Target Language Compiler (TLC) eine wichtige Rolle. Mit Hilfe des TLC kann man beispielsweise:

- Funktionsaufrufe
- Ausführungszeiten
- Die generelle Programmstruktur

vorgeben und optimieren. Unter genereller Programmstruktur wird verstanden, wie der TLC zum Beispiel die Klassenattribute setzt. Diese können *private*, *public* oder *global* gesetzt werden, je nach Anspruch und Aufgabenstellung. Die Bearbeitung des TLCs erfolgt mit einer eigens von Mathworks[©] entwickelten Skriptsprache. Der Ablauf der automatischen Codegenerierung ist in Abbildung 3 dargestellt.

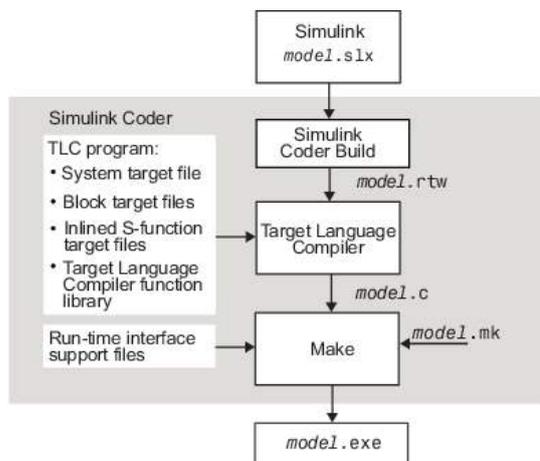


Abbildung 3: Schematischer Ablauf der automatischen Codegenerierung

Ein weiteres Ziel der Abschlussarbeit soll ein automatischer Prozess sein, der aus einem Matlab/Simulink-Modell einen C++-Code generiert. Dieser Prozess ist nicht gebunden an elektrische oder pneumatische Regelstrukturen. Daraus resultiert ein einheitlicher Code für die diversen Regelstrukturen. Im weiteren Verlauf der Arbeit soll der generierte Code dann in eine Software integriert werden, die bisher keine pneumatischen Regelstrukturen unterstützt.

Bevor dieser Code jedoch integriert werden kann, muss eine Voruntersuchung getätigt werden, um die Gemeinsamkeiten sowie die Unterschiede, in den Strukturen von elektrischen und pneumatischen Reglern zu identifizieren. Elementare Unterschiede sind zum Beispiel die Schnittstellen der Regler. Die elektrischen Regler benötigen zum Beispiel einen Strom i , eine Geschwindigkeit v oder eine Position x . Der pneumatische Regler hingegen benötigt einen Solldruck P . Dadurch entsteht eine einheitliche Plattform mit generischen Schnittstellen, die produktübergreifend eingesetzt werden kann.

Als Abschluss der Arbeit soll die angepasste Software mit dem generierten Code auf einem Mikrocontroller getestet werden. Bei dieser realen Implementierung ist nicht nur der Mikrocontroller involviert sondern auch noch ein Ventil, sowie ein angeschlossener Zylinder, der als *Volumen* dient. Die Kommunikation zwischen dem Ventil und dem Mikrocontroller ist über Serial Peripheral Interface (SPI) realisiert. Ein angeschlossenes Volumen wird benötigt, damit die Regelstrukturen gemäß den Matlab/Simulink Modellen getestet werden kann.

Die Abschlussarbeit soll also folgenden Ziele hervorbringen:

- Identifizierung und generische Auslegung der Reglerschnittstellen
- Automatische Codegenerierung der Regelstrukturen
- Testen der Software auf realem Testaufbau

[1] Digital Simplicity: Weltneuheit Festo Motion Terminal VTEM – Festo AG Co. KG

Bildquellen:

- Abbildung 1: Festo AG & Co. KG
- Abbildung 2, 3: Mathworks

Auf dem Weg zum offenen Ökosystem für intelligente, sichere und vertrauenswürdige Assistenzsysteme im Internet of Things. Implementierung und technologische Verknüpfung von Haushaltsgeräten mit digitalen Assistenten.

Björn Bieniakiewicz*, Kai Warendorf Dominik Schoop

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Einführung

Schon in den 60er Jahren des letzten Jahrhunderts wurde der Grundstein für den heutigen Begriff „Smart Home“ gelegt. Mit dem ECHO IV entwickelte James Sutherland im Jahr 1966 den ersten Computer, der heutzutage als „smart“ bezeichnet werden würde. Mit einer integrierten Haushaltsinventarverwaltung, Kalenderfunktion und Funktionen zum Klimatisierungsmanagement ging Sutherland den ersten Schritt eines Weges, den heutzutage viele Forschungseinrichtungen und kommerziell ausgerichtete Unternehmen gehen um technische Geräte im eigenen Haus zu vernetzen. [1] Nach heutigem Verständnis umfasst der Begriff dabei die Vernetzung und dabei eine gezielte Verknüpfung von Haushaltstechnik wie z.B. Lampen, Heizung und Steckdosen, aber auch von Haushaltsgroßgeräten wie Waschmaschinen, Kühlschränken und Unterhaltungselektronik.

Ziel der Vernetzung ist die Steigerung der Wohn- und Lebensqualität durch eine einfachere Bedienung von elektronischen Bauteilen. Gleichzeitig ein besseres Energiemanagement durch die intelligente Regulierung von Heizkörpern und Jalousien, aber auch die Erhöhung der Sicherheit durch Gefahren aus dem alltäglichen Leben, mithilfe von Rauchmeldern die automatisch die Feuerwehr alarmieren können, aber auch gegen Einflüsse von außen durch Sensoren die unbefugtes Betreten der eigenen Wohnung erkennen können. [2]

Der Umsatz im Smart Home Markt beträgt in Deutschland im Jahr 2017 etwa 1,8 Milliarden Euro. Bis zum Jahr 2022 wird der Umsatz in diesem Marktsegment alleine in Deutschland auf 5,2 Milliarden Euro geschätzt. Das mit 24% jährlich stark wachsende Umsatzpotential führt dazu, dass immer mehr Unternehmen eigene Smart Home Lösungen präsentieren. [3]

Neben Herstellern, die ihre Produkte mit smarten Funktionen aufwerten, wie die Medion AG oder der Philips GmbH steigen auch immer mehr – als digitale Konzerne bekannte – Unternehmen wie Google und Amazon in diesen Markt ein. Gerade letztere liefern, mit Google Home und der Echo Produktreihe Systeme, die einen hohen Komfortfaktor versprechen: digitale Assistenten – gesteuert per Sprachbefehl.

Durch die direkte Interaktion des Nutzers mit dem Smart Home über ein zentrales System per Sprache sollen Barrieren der Steuerung der integrierten Systeme minimiert werden. Ein Ansatz der Benutzer aller Altersklassen zur stärkeren Integration smarterer Geräte in ihren Haushalt überzeugen konnte.

Stand der Technik

Neben ZigBee, HomeMatic, EnOcean, WLAN und Bluetooth existieren zahlreiche Basistechnologien zur Vernetzung einzelner Geräte in einem Haushalt. Der Hersteller wählt dabei die für ihn passende Übertragungstechnologie und stattet seine Geräte mit entsprechender Schnittstellen aus. In der Vielfalt dieser Technologien, die untereinander zumeist inkompatibel sind, steckt ein Problem: Geräte von unterschiedlichen Anbietern können nur selten miteinander kommunizieren und verhindern so die Schaltung von vielen Komponenten zu einer vernetzten und intelligenten Hausautomation.

So werden von den Herstellern eigene, kleine Produktkreise generiert, die den Benutzer zwingen ein vom Hersteller zertifiziertes Produkt zu wählen um dieses in die eigene Hausautomation zu integrieren. Die Produktpalette eines Herstellers deckt dabei oft nur spezielle Lebensbereiche, wie z.B. eine Beleuchtungssteuerung oder ein Heiz- und Klimatisierungsmanagement ab. Eine Integration zusätzlicher Produktbereiche für weitere Lebensbereiche

*Diese Arbeit wurde durchgeführt bei der Firma Robert Bosch GmbH, Renningen

gestaltet sich dabei aufgrund inkompatibler Schnittstellen als schwierig.

Einen weiteren Problembereich stellt der Aspekt des Datenschutzes dar. Unternehmen mit der höchsten Marktdurchdringung sind häufig US-Unternehmen, folglich werden vorrangig amerikanische Gesetze zur Bemessung des Umgangs und des Schutzes persönlicher Daten der Benutzer herangezogen. Mit dem „Safe Harbor“-Abkommen wurde im Jahr 2000 ein Beschluss der Europäischen Kommission gefasst, der einen Austausch von personenbezogenen Daten mit Unternehmen in den USA – unter Einhaltung der europäischen Datenschutzrichtlinie – ermöglicht. Im Jahr 2016 wurde diese Regelung durch den „EU-US Privacy Shield“ ersetzt, der eine Anpassung an neue Datenschutzrichtlinien der Europäischen Union und den vereinigten Staaten von Amerika ermöglichte. [4] Trotz dieser Beschlüsse, die einen Mindeststandard an den Schutz und den Umgang mit personenbezogenen Daten bilden, herrschen noch immer weitreichende Unterschiede in den Datenschutzgesetzen beider Parteien. Diese Unterschiede sorgen – gerade auf europäischer Seite – für Unsicherheit und Misstrauen gegenüber Smart Home Produkten, die aufgrund ihrer Funktionsweise und Sensorik Bewegungs- und Verhaltensprofile der Benutzer anlegen können. Im Fall von digitalen Sprachassistenten liefern diese zusätzlich die technischen Möglichkeiten zum Abhören von Gesprächen in der eigenen Wohnung.

Vision

Um die gesetzten Ziele des Smart Home, die Steigerung von Komfort sowie Energieeffizienz und der Sicherheit im eigenen Zuhause zu erreichen ist eine effiziente Verknüpfung der einzelnen Komponenten des Smart Home notwendig. Durch eine herstellerübergreifende Kommunikation der einzelnen Geräte können die Ziele im Rahmen der zunehmenden Digitalisierung unter Berücksichtigung der wirtschaftlichen Interessen der Kunden und Unternehmen erreicht werden. Dadurch werden die einzelnen Produkte effizienter und leistungsfähiger. Eine übergreifende Plattform ermöglicht Kunden neue Freiheiten bei der Auswahl der Produkte für das eigene Smart Home, aber auch Herstellern die Konzentration auf die Neu- und Weiterentwicklung der Produkte, ohne stets neue Schnittstellen zur Integration von Produktpaletten anderer Hersteller entwickeln zu müssen. Ein offenes Ökosystem ermöglicht durch offene Schnittstellen und Standardisierung eine übergreifende Steuerung und Schaltung einzelner Komponenten durch den Kunden ohne auf multiple Smartphone Apps der Hersteller angewiesen zu sein. Dies gewährt

Aktualität, technischen Fortschritt und stetige Funktionssteigerung. Durch die gebündelte Reichweite der verschiedenen Smart Home Anbieter kann dabei eine Plattform mit standardisierten Schnittstellen entstehen, die als Standard für Unternehmen die dieses Marktsegment neu betreten gelten, wodurch weitreichendere Integrationsmöglichkeiten entstehen.

Durch die Kommunikation der einzelnen Produkte werden – funktionsbedingt – viele personenbezogene Daten gesammelt. Die Möglichkeiten der Aggregation dieser Daten befähigen zur umfassenderen Erstellung von Verhaltensprofilen der Benutzer. Ein auf die Europäische-Datenschutz-Grundverordnung (EU-DSGVO) abgestimmtes System kann dabei genutzt werden, dass der Benutzer zukünftig seine Rechte zur informationellen Selbstbestimmung umfassender wahrnehmen kann.

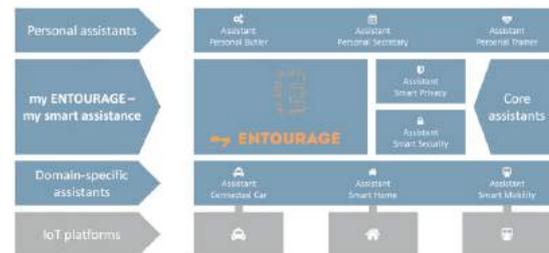


Abbildung 1: Geplante Architektur des ENTOURAGE Ökosystems

Die Entwicklung und Erforschung solcher Plattformen wurde im ENTOURAGE-Projekt gebündelt. Dieses Projekt umfasst neben der Robert Bosch GmbH weitere deutsche Unternehmen und Forschungseinrichtungen wie dem Fraunhofer IAO, der Universität Kassel und der technischen Universität Darmstadt. Das Projekt umfasst dabei nicht nur den Bereich Smart Home, sondern auch den Bereich Mobilität und Connected Car. Darstellung 1 zeigt dabei die geplante Architektur des ENTOURAGE Ökosystems. Ziel, des vom Bundesministerium für Wirtschaft und Energie, geförderten Projekts ist ein offenes Ökosystem für jegliche Assistenzsysteme in den genannten Bereichen. [5]

Abstrahierte Schnittstelle für Sprachinteraktionen

Als Komponente des ENTOURAGE-Projekts soll im Rahmen dieser Arbeit ein Teilbereich dieses offenen Ökosystems umgesetzt werden. Ziel ist die Entwicklung einer, von den Funktionen und Plattformen einzelner Hersteller abstrahierte Schnittstellenschicht, die eine personalisierte Sprachinteraktion eines belie-

bigen digitalen Assistenten mit einem Haushaltsgerät ermöglicht. Diese Schnittschicht soll die Implementierung von digitalen Assistenten in ein offenes Ökosystem zur Realisierung von integrierten und übergreifenden Steuerprozessen ermöglichen.

Dabei sollen mit dem „Alexa Skill Kit“ und dem „Actions on Google“ System bereits existierende Funktionen der digitalen Assistenten zur Bedienung und Steuerung von Smart Home Komponenten weiterhin, auch über die abstrahierte Schnittschicht, ermöglicht werden. Darstellung 2 zeigt dabei schemenhaft die Realisierung einer abstrahierten Schnittschicht zwischen digitalen Assistenten und Endgerät.

Die Herausforderung dabei besteht in der Vielfaltigkeit bisheriger Herstellerlösungen, die im Rahmen einer solchen Schnittstellenebene kombiniert werden müssen. Dabei muss auf zwei Seiten Hersteller- und Plattformunabhängig entwickelt werden: auf Seiten

der sprachgesteuerten, intelligenten digitalen Assistenten wie Google Home oder Amazon Alexa, aber auch auf Seiten der Küchengeräte, die aus Konsumentensicht im Smart Home Bereich bisher eine untergeordnete Rolle gespielt haben.

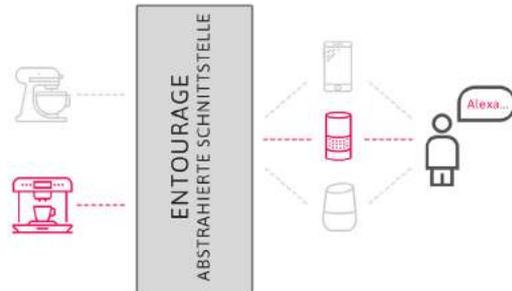


Abbildung 2: Schema einer abstrahierten Schnittschicht im Smart Home

[1] UBM Americas: <http://www.drdoobs.com/architecture-and-design/if-you-cant-stand-the-coding-stay-out-of/184404040> (Abgerufen am 21.11.2017)

[2] Meyer, Sibylle; Eva Schulze (2010): Smart Home für ältere Menschen. Handbuch für die Praxis. Stuttgart: Fraunhofer IRB Verlag

[3] Statista GmbH: <https://de.statista.com/outlook/279/137/smart-home/deutschland> (Abgerufen am 21.11.2017)

[4] Europäische Kommission: http://europa.eu/rapid/press-release_IP-16-216_de.htm (Abgerufen am 21.11.2017)

[5] ENX Association: <http://www.entourage-projekt.de/> (Abgerufen am 21.11.2017)

Bildquellen:

- Abbildung 1: ENX Association: [http://www.entourage-projekt.de/?file=files/downloads/ENTOURAGE %20 Merged%20Online%20Flyer%20Deutsch.pdf](http://www.entourage-projekt.de/?file=files/downloads/ENTOURAGE%20Merged%20Online%20Flyer%20Deutsch.pdf) (Abgerufen am 21.11.2017)
- Abbildung 2: Eigene Darstellung

Konzipierung einer Softwarearchitektur für das Produktdatenmanagement mit dem Fokus auf Flexibilität hinsichtlich der Integration heterogener Datenquellen

Benjamin Braun*, Reinhard Keller, Walter Lindermeir

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Im heutigen Zeitalter der digitalen Daten werden immer mehr Daten, vor allem im Bereich CAE, elektronisch erfasst, bearbeitet und dann gespeichert, damit sie zur Verbesserung von Prozessen und Produkten später herangezogen werden können. Um die gespeicherten Daten effizient verwenden zu können, müssen diese so verwaltet werden, dass jeder Benutzer in der Lage ist, diese Daten ohne umfangreiche Vorkenntnisse zu nutzen. Dafür zuständig sind sogenannte PDM-Systeme. Die vorliegende Arbeit beschäftigt sich mit dem Konzept zur Erstellung eines PDM-Systems unter Verwendung eines ASP.NET-Core-Servers.

PDM-System

Unter einem Produktdatenmanagementsystem versteht man ein zentrales System zur Verwaltung aller relevanten Produktdaten und deren Prozessinformationen. Das Ziel von PDM-Systemen ist es, die Produktdaten mit maximalem Nutzungspotenzial in den Entwicklungsprozess einzubringen. Hierfür stellt das PDM-System die Produktdaten und Prozesse für den Benutzer in gefilterter und aufbereiteter Form zur Verfügung. Dazu sollte keine weitere externe Software benötigt werden, sondern alle benötigten Funktionen im PDM-System integriert sein. Bei den auf dem Markt erhältlichen Systemen werden die Daten häufig in relationalen Datenbanken gespeichert.[1]

Startbedingungen und Anforderungen

Durch eine über die Zeit gewachsene Infrastruktur liegen die Produktdaten häufig in einer Vielzahl verschiedener Datenquellen jeweils in strukturierter Form vor. Durch das im Rahmen der vorliegenden Arbeit konzipierte PDM-System sollen die Daten der Produkte durch Objekte und deren Eigenschaften repräsentiert und durch eine Server-Anwendung zugänglich gemacht werden. Hierbei soll der Server über eine benutzerfreundliche Web-API als *Single Point of Access* für die Daten dienen

und auf Änderungen der Struktur der Datenquellen und Rückgabewerte flexibel anpassbar sein.

Konzept

Um aus den teilweise in relationalen Datenbanken gespeicherten Daten Objekte zu erzeugen, wird die Technik der objektrationalen Abbildung angewandt. Hierbei werden Tabellen der Datenbanken auf Klassen und die Entitäten der Datenbank auf die Objekte abgebildet. Die Abbildungslogik der Speicherquellen auf die Klassen der Objekte wird auf dem Server für jede Klasse in einem Microservice realisiert. Im Weiteren werden diese Microservices als Mapper bezeichnet. Die Mapper können, jeweils für ihre Klasse, alle relevanten Befehle, wie Erstellen, Lesen, Ändern und Löschen, umsetzen.

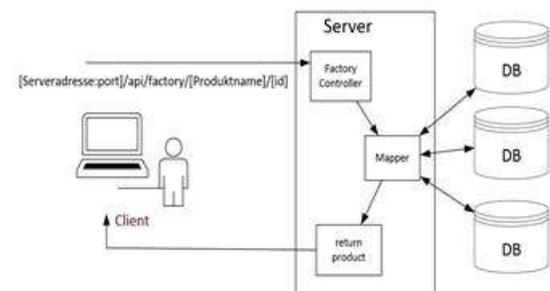


Abbildung 1: Konzept

Damit der Benutzer das gewünschte Objekt laden kann, muss er über eine Schnittstelle mit dem Server kommunizieren. Dafür wird in diesem Konzept das REST-Prinzip angewendet, die Mapper der einzelnen Klassen werden per URL kontaktiert. Die URL des Konzepts hat folgende Form:

```
<Serveradresse:port> /api/factory/  
<Produktname> /{id}
```

*Diese Arbeit wurde durchgeführt bei der Firma deutschen Accumotive GmbH & Co. KG, Nabern

Im ersten Teil der URL stehen die Adresse und der Port des Servers, auf dem sich der Mapper befindet. Die beiden nächsten Teile geben den Pfad auf dem Server an, mit dem die Mapper erreichbar sind. Die letzten beiden Teile beinhalten den Namen des Mappers und die ID des gewählten Objektes, wobei letztere optional ist. Mit diesen Informationen aus der URL kann der Server jeden benötigten Mapper auswählen. Dieser erzeugt nun ein Produkt, welches per JSON-String an den Clienten zurückgegeben wird. Hierfür werden, wie bei REST üblich, die HTTP-Methoden Get, Post, Put und Delete verwendet. Ein Produkt kann Objekte oder auch Informationen des Servers enthalten.

Da häufig auch sehr große Datenmengen gespeichert werden, müssen vor allem für das Herunterladen von Daten Strategien entwickelt werden, die die Antwortzeit des Servers verbessern. Da bei großen Objekten in der Regel nicht immer alle Daten benötigt werden, sollen die Daten nach dem hier vorliegenden Konzept mit dem Verfahren *Lazy Loading* geladen werden. Hierzu müssen die entsprechenden Properties mit dem Attribut *Lazy* versehen werden. Besitzt eine Property das Attribut *Lazy*, so wird sie nur bei einer expliziten Anforderung geladen. Das Attribut *Lazy* kann als Wert entweder einen leeren String oder aber eine URL enthalten. Wenn *Lazy* einen leeren String enthält, wurde die Property bereits geladen. Enthält sie aber eine URL, so wurde die Property noch nicht geladen. Die URL dient also dazu, den Mapper für den Inhalt der Property nach dem REST-Prinzip zu adressieren. Wird nun die Property nachgeladen, so wird die URL mit einem leeren String überschrieben, um zu verhindern, dass die Daten mehrmals geladen werden. Eine alternative Lösung des Problems wäre mit der Einbindung des Konzepts der *Hypermedia Application Language* in den Prozess möglich. Bei diesem Konzept werden anstelle der Werte Hyperlinks hinterlegt, welche auf den Speicherort des Wertes verweisen. Dies kann aber dazu führen, dass Daten mehrfach geladen werden, da es keinen Mechanismus gibt, der einen durchgeführten Ladevorgang speichert.

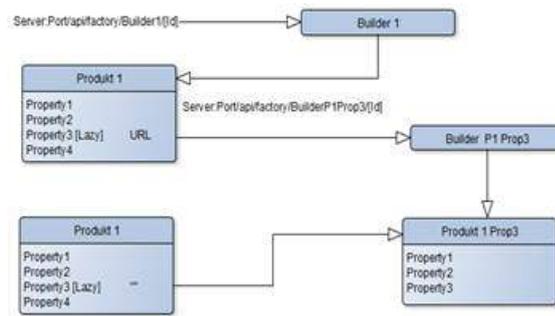


Abbildung 2: Lazy Loading

Das *Lazy Loading* kann auch beim Up-date vorhandener Objekte helfen, denn für ein Update muss zuerst das alte Objekt geladen und bearbeitet werden. Da aber hier nur die für die Änderung relevanten Daten geladen werden, beinhaltet das Objekt beim Hochladen nur Daten, die für die Änderung auch relevant sind. Es muss nur darauf geachtet werden, dass der Server beim Hochladen nicht relevante Daten auch nicht nachlädt. Hierfür darf der Mapper beim Upload die Lazy-Loading-Attribute nicht beachten, wenn diese noch nicht geladen sind.

Zusammenfassung und Ausblick

Mit der hier vorliegenden Arbeit wurde ein abstraktes PDM-Konzept erstellt, welches die Möglichkeit bietet, ein effizientes Software-Framework auf einem Server zu implementieren. Ein Vorteil dieses Konzeptes ist, dass mehrere schon vorhandene Datenbanken mit dem System verwaltet werden können und ein effizienter Zugriff über eine Web-API als *Single Point of Access* ermöglicht wird. Weiterführend wäre es von großem Nutzen, eine integrierte graphische Darstellung der Daten für den Benutzer zu ergänzen. Außerdem sollten bei der realen Umsetzung des Konzeptes die URLs von den Programmen oder von einem Webformular automatisiert versendet werden.

[1] Eigner, M.; Stelzer, R.: „3.2 PDM – Definition und Funktionsüberblick“, in Product Lifecycle Management, Berlin & Heidelberg, Springer-Verlag, 2009, pp. 31–39.

Bildquellen:

- Abbildung 1, 2: Eigene Darstellung

Mobile Development – Entwicklung einer webbasierten mobilen App

Nagihan Caliskan*, Reinhard Schmidt, Andreas Rößler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Die Kurzform App steht für den Begriff „Applikation“. Gemeint sind Software-Applikationen, die auf einem Computer, einem Smartphone oder Tablet laufen. Damit eine App auf unterschiedliche Plattformen ausgeführt werden kann, muss mit Multiplattform-Development gearbeitet werden. Die gängigen Betriebssysteme sind dabei iOS und Android. Bei Apps unterscheidet man zwischen nativen Apps, Web-Apps und Hybrid-Apps.

Native Apps

Native Apps sind speziell an die Zielplattform angepasst und laufen direkt auf dem Betriebssystem. Eine native App kann deswegen auf alle plattformspezifischen Hard- und Software-Funktionen zugreifen, wie Dateien, GPS, Beschleunigungssensoren, Mikrofon und Kamera. Ein für native App, die beispielsweise für das Android-Betriebssystem entwickelt wurde, läuft deswegen nicht mit einem iOS-Betriebssystem. Möchte man, dass eine Anwendung für beide Plattformen zur Verfügung steht, muss die Anwendung für beide Plattformen entwickelt werden. Für jede Plattform muss als ein eigenes Software Development Kit (SDK) mit integrierter Entwicklungsumgebung verwendet werden, entweder Android SDK oder Xcode [1].

Web Apps

Eine Web App ist eine in HTML5 programmierte Website, die durch den Browser des Endgerätes dargestellt wird und der den Inhalt optimiert darstellt. Im Gegensatz zu einer normalen Website, auf der ein Besucher durch einzelne Seiten navigiert, um Inhalte wie zum Beispiel Texte, Bilder und Videos abzurufen, gibt man bei einer Web App Informationen ein, die verarbeitet werden und ein Ergebnis ermittelt wird. So ist beispielsweise eine Suchmaschine wie Google mehr eine Web App als eine normale Website. Eine Web App lässt sich von einer Website man Layout und der Art der Bedienelemente unterscheiden. Die App beinhaltet Bedienelemente aus typischer Anwendungssoftware, wie beispielsweise Symbolleis-

ten, Buttons, Dialogfelder, Eingabefelder und Formulare und ist somit auch benutzerfreundlicher und einfacher zu bedienen [2].

Hybrid Apps

Hybrid-Apps stellen eine Sonderform einer mobilen App dar. Sie vereinen die Vorteile von nativen und Web-Apps, indem sie auf viele Hard- und Software-Komponenten des mobilen Endgeräts zugreifen und gleichzeitig unterschiedliche Plattformen bedienen können. Sie können sowohl auf unterschiedlichen Mobilgeräten (Smartphone & Tablet) und PCs als auch auf unterschiedlichen Betriebssystemen laufen [3].

Das UI-Design: Einheitlich, aber individuell

Die entwickelten Apps sollen auf allen Endgeräten und unabhängig vom Betriebssystem gleich aussehen. Außerdem sollte der Anwender sofort eine positive Nutzererfahrung und Wiedererkennung erleben. Er sollte intuitiv in der Lage sein die beabsichtigte Aktion durchzuführen.

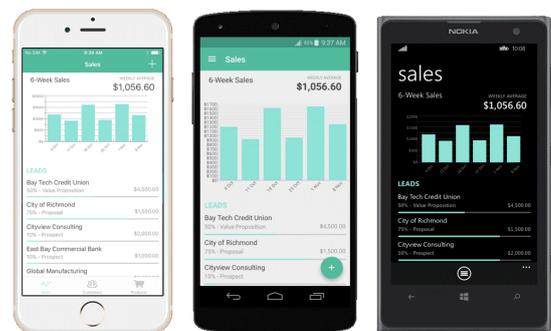


Abbildung 1: Darstellung der App für unterschiedliche Plattformen

Auf unterschiedliche Darstellungsformen, bedingt durch die verschiedenen Betriebssysteme, sollte rücksicht genommen werden. So ist beispielsweise die Darstellung eines Linkes unter Android etwas anders als unter iOS. Weitere Beispiel ist das Datum oder die Text-

*Diese Arbeit wurde durchgeführt bei der Firma Fraunhofer-Anwendungszentrum KEIM, Hochschule Esslingen

eingabe in Formularen. Ein Nutzer erwartet die ihm vertraute, gewohnte Darstellung. Die Abbildung 1 zeigt ein Beispiel für die unterschiedliche Darstellungsweise.

Konzeption der App

Ein Konzept ist notwendig, um den Erfolg einer mobilen Anwendung sicherzustellen. Ein Konzept ist die Phase zwischen Idee, Anwendung und tatsächlichem Start der Programmierung. Mit diesen Schritten erzielt man eine gute App [2][4]:

1. Definition der App
Eine App Definition ist eine kurze und schlüssige Erklärung des Hauptzweckes der App und deren Zielgruppen. Diese Definition ist hilfreich, um eine Liste an notwendigen Funktionalitäten zu erstellen.
2. Storyboard
Es ist wichtig, zu wissen, welche Nutzer diese App verwenden werden. Dazu wird ein Profil der späteren Nutzer erstellt. Folgende Fragestellungen sind in diesem Fall wichtig:
 - a) Sind das Jugendliche, Teenager oder ältere Leute?
 - b) Welcher Typ von Mensch sind diese?
 - c) Sind es eher Männer oder Frauen?
 - d) Welchen Beruf haben die Nutzer?
 - e) Welche Plattform oder welches Gerät werden die Nutzer vorwiegend verwenden?
 - f) Wie lange wird die durchschnittliche Nutzungsdauer sein?
3. Brainstorming
Für eine sinnvolle App benötigt man zu Beginn viele Ideen. Diese Ideen werden in Wörtern oder Sätzen formuliert. Dabei ist es zunächst nicht wichtig, ob alle Funktionalitäten am Ende umgesetzt werden können oder nicht.

4. Zeichnungen der App
In dieser Phase wird das Aussehen der App geplant. Dieses erfolgt mithilfe von Bleistiftzeichnungen, Farbschema, Wireframes und Mockups.
5. Nutzertest
Die App wird vor der technischen Umsetzung mit den erstellten Mockups, Wireframes oder Zeichnungen von unterschiedlichen Personen so lange getestet, bis eine gute Usability erreicht ist.
6. Details und Überarbeitung
Fehler und Schwächen des bisherigen Konzepts, die bei dem Nutzertest gefunden wurden, werden korrigiert und gegeben falls nochmals mit einem Nutzertest geprüft.
7. Start der Programmierung
Schließlich kann mit der Programmierung der App begonnen werden.

Die Abbildung 2 zeigt das prinzipielle Vorgehen bei einer App-Entwicklung.



Abbildung 2: Eine App konzipieren

[1] <https://www.computerwoche.de/a/der-richtige-weg-zur-multi-plattform-app,3062690>

[2] Franke, Florian & Ippen, Johannes (Hrsg.). (2013). Apps mit HTML5 und CSS3: für iPhone, iPad und Android. Bonn: Galileo Press

[3] <http://www.app-entwickler-verzeichnis.de/faq-app-entwicklung/11-definitionen/586-unterschiede-und-vergleich-native-apps-vs-web-apps-2>

[4] <http://www.yuhiro.de/app-konzeption/>

Bildquellen:

- Abbildung 1: <https://www.xamstatic.com/dist/images/pages/forms/crm-app-7KknaygL.png>
- Abbildung 2: <https://pixabay.com/de/ux-design-webdesign-app-mobil-787980/>

Erstellung eines Kennzahlenreports und Konzeption eines automatisierten Reporting-Systems im Themenbereich mobiler Betriebssysteme – Apple iOS

Zozan Ceyhan*, Harald Melcher, Astrid Beck

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Der Trend zur **App-Programmierung** und – **Nutzung** macht auch vor der Automobilbranche kein Halt. In Zeiten der digitalen Transformation steigt die Nachfrage nach entsprechenden mobilen Services für den Kunden. Diese Entwicklung zwingt Automobilhersteller, ihr entsprechendes Produktportfolio durch Applikationen, kurz Apps, zu erweitern und somit sowohl die **Kundenanbindung** als auch den **Kundenservice** zu verstärken und verbessern [1]. Hierauf basierend sind laufend unterschiedliche Apps wie beispielsweise interaktive Guides für das Produktportfolio, Produktneuheiten, aber auch Augmented Reality Apps zu entwickeln. Dadurch lässt sich interessierten Kunden das Produkterlebnis näher bringen. Diese Apps stehen dem Kunden über die gängigen Bereitstellungsplattformen wie dem Apple App Store oder dem Google Play Store zum Download zur Verfügung.

Aktuell stellt ein Automobilhersteller verschiedene Apps für iOS Endgeräte im **Apple App Store** zur Verfügung. Die jeweiligen Apps stammen aus verschiedenen Fachbereichen und bieten sowohl dem Kunden als auch dem Mitarbeiter der Organisation verschiedene Services an. Die **unternehmensintern** entwickelten Apps laufen an einer **zentralen Abteilung** im Unternehmen zusammen. Diese Abteilung unterstützt dabei den jeweiligen Product Owner aus dem Fachbereich in den verschiedenen Entwicklungsphasen einer App, bis hin zum finalen Rollout im App Store. Hierbei interessiert den Product Owner, inwieweit sich

der Produktlebenszyklus der jeweiligen App im Apple App Store durch **relevante Kennzahlen** beobachten lassen kann.

Hierzu bietet Apple Inc. bereits einen Reporting Service, das *App Analytics Tool*, an. Dieses Tool ist dabei nur durch den kostenpflichtigen iTunes-Connect Account, einem Developer Account von Apple, zugänglich. Das Reporting Tool bietet der Abteilung die Möglichkeit, bei entsprechendem Bedarf aus dem Fachbereich, die benötigten Kennzahlen in einem CSV-Format zu exportieren und an die Product Owner weiterzuleiten. Dieser Vorgang erfordert für die Abteilung derzeit einen **manuellen Aufwand**. Die generierte CSV-Datei enthält lediglich die entsprechenden Kennzahlen in einem Rohdatenformat. Dies bedeutet, dass die Kennzahlen aus dem Report in dem entsprechenden Fachbereich **zusätzlich aufzubereiten** sind. Daher sehen sich sowohl die Product Owner der verschiedenen Applikationen als auch die Abteilung selbst in der Notwendigkeit, ein **eigenes Kennzahlen Reporting-System** für unternehmensintern entwickelte Apps aufzusetzen. Dabei ist das Ziel zu verfolgen, die relevanten Kennzahlen in Zukunft automatisiert an die entsprechenden Product Owner aus dem Fachbereich zur Verfügung zu stellen.

Um das Reporting strukturiert und geeignet aufzubauen, ist der Prozess zur **Informationsversorgung** unabdingbar (Vgl. Abbildung 1) [2].

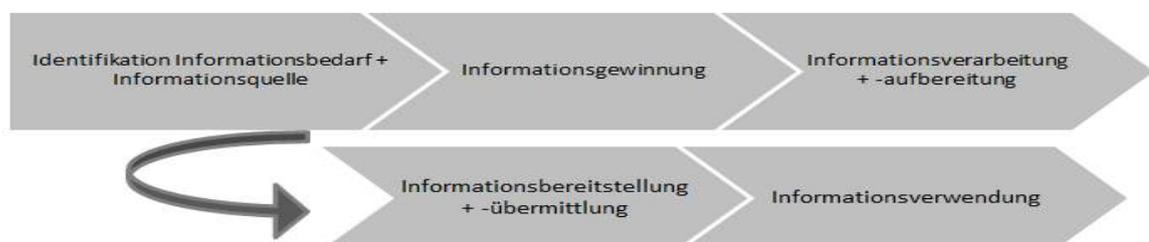


Abbildung 1: Informationsprozess

*Diese Arbeit wurde durchgeführt an der Fakultät Informationstechnik

Auf Grundlage dieses Prozesses sind die Anforderungen der Product Owner, welche nachfolgend als Kunden bezeichnet sind, zu analysieren. Die Ermittlung der **Anforderungen** basiert dabei auf den **Gestaltungsdimensionen** eines Reportings (siehe Abbildung 2). Dabei umfassen die Fragen zur Anforderungsanalyse die Themengebiete des Informationsbedarfs (Reportinhalt), der Informationsaufbereitung (Darstellung, Form), der Informationsbereitstellung und Informationsübermittlung (Zyklus, Termin) und der Informationsverwendung (Reportingzweck). Die Befragung der Kunden erfolgt durch **Experteninterviews**. Das Experteninterview setzt sich dabei aus zwei Teilen zusammen. Der erste Teil umfasst eine **quantitative Methode** zur Erfassung der ersten Anforderungen der Kunden. Hierzu ist in dieser Arbeit ein schriftlicher Fragebogen zu erstellen, welcher grundlegende Fragen zu den **Gestaltungsdimensionen** des Reports [3] (Abbildung 2) enthält und an den Kunden via E-Mail versendet wird.

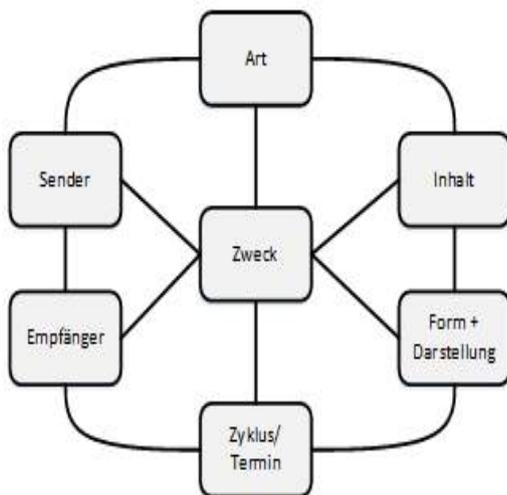


Abbildung 2: Gestaltungsdimensionen eines Reports

Der zweite Teil der Befragung umfasst ein persönliches Interview mit dem Kunden. Dieses erfolgt durch das **qualitative Vorgehen** des narrativen Interviews. Durch diese Methode sind weitere wichtige Anforderungen der Kunden in Form von **Erzählungen** zu erfahren. Im Anschluss an das qualitative Interview ist eine erneute Teilerhebung in Form eines schriftlichen Fragebogens vorzunehmen. Dieser Fragebogen enthält **ausgewählte Kennzahlen** für das Reporting, welche auf den Erkenntnissen des ersten Fragebogens basieren. Hierbei haben die entsprechenden Kunden die Möglichkeit, diese Zahlen nach ihrer **Wichtigkeit zu bewerten**. Durch diese aufwendige Vorgehensweise zur Ermittlung der Kundenanforderungen ist gewährleistet, dass sich das Reporting-System kundenorientiert aufbauen lässt. Sowohl die schriftlichen Fragebögen als auch das Interview sind durch geeignete Methoden auszuwerten. Der Fokus dieser Befragung liegt dabei in dem qualitativen Vorgehen. Daher ist zur Auswertung der Experteninterviews, die Methode der **qualitativen Inhaltsanalyse** nach Philipp Mayring gewählt [4]. Dieses Vorgehen erfordert eine Verschriftlichung der persönlichen Interviews und der Fragebögen. Dadurch lassen sich in der späteren Anforderungsanalyse sowohl die quantitative als auch die qualitative Methode gezielt miteinander **kombinieren** und somit alle Anforderungen der Kunden definieren.

Sind die Anforderungen der Kunden, nach diesem Prozess ermittelt, erfolgt anschließend eine **Interpretation** der Ergebnisse. In diesem Zusammenhang sind geeignete Lösungen zu finden, um das Reporting kundenorientiert umzusetzen. Mit der Interpretation der Ergebnisse aus den Interviews ist die theoretische Konzeption der Anforderungen an das Reporting erfolgreich abgeschlossen.

Zum bisherigen **Stand der Arbeit** existiert ebenfalls ein theoretischer Ansatz, um die entsprechenden Kennzahlen aus der vorhandenen Datenquelle zu beziehen. Auf Grundlage der theoretischen Konzeption, welches das primäre Ziel der Arbeit darstellt, gilt es anschließend das Reporting-System erfolgreich umzusetzen.

-
- [1] Aichele, Christian, et al. 2016. Auf dem Weg zur optimalen mobilen Anwendung. Wiesbaden : Springer Vieweg, 2016.
 [2] Taschner, Andreas. 2013. Management Reporting –Erfolgsfaktor internes Berichtswesen. Wiesbaden : Springer Fachmedien, 2013.
 [3] Küpper, Hans-Ulrich, et al. 2013. Controlling : Konzeption, Aufgaben, Instrumente (6. Auflage). Stuttgart : Schäffer-Poeschel Verlag, 2013.
 [4] Mayring, Philipp. 2010. Qualitative Inhaltsanalyse – Grundlagen und Techniken. Weinheim : Beltz Verlag, 2010.

Bildquellen:

- Abbildung 1: Eigene Darstellung in Anlehnung an (Taschner, 2013 S. 13)
- Abbildung 2: Eigene Darstellung in Anlehnung an (Küpper, et al., 2013 S. 237)

Konzipieren und Implementieren eines Release Management Prozesses mit dem Fokus auf Continuous Integration in einer Microsoft SharePoint Umgebung

Akif Cinar*, Thomas Rodach, Catharina Kriegbaum-Kling

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Mit zunehmender Digitalisierung der Umwelt ist auch die IT-Branche gezwungen seine Prozesse dynamischer zu gestalten, um weiterhin wettbewerbsfähig bleiben zu können. Aus diesem Grund ist es von großer Bedeutung die Zyklen der Release Prozesse in der Softwareentwicklung so kurz wie möglich zu halten. Jedoch ist eine Änderung an einer Produktivumgebung ein sehr langatmiger Prozess. Bei Änderungen in der Produktivumgebung handelt es sich keinesfalls um Routine und können nicht in Kürze umgesetzt werden. Der Zeitrahmen eines solchen Release Management Prozesses kann Wochen sogar Monate beanspruchen. Es ist nicht unüblich, dass innerhalb einer IT-Organisation die Entwicklung und der Betrieb sich in einem Zielkonflikt befinden. Die Entwicklung möchte auf den sich schnell ändernden Markt reagieren und dem Unternehmen damit einen Wettbewerbsvorteil verschaffen. Entgegengesetzt möchte der Betrieb stabile und sichere IT-Services für den Kunden bieten.

Um diesen Konflikt zwischen Entwicklung und Betrieb aufzubrechen und die daraus resultierenden langen Release-Zyklen möglichst auf ein Minimum zu reduzieren, müssen neue Vorgehensweisen implementiert werden. Die Lösung nennt sich DevOps. DevOps bietet mit seinen Verfahren und Techniken einen Ausweg aus diesem Konflikt.

Softwareentwicklung mit CI

Die kontinuierliche Integration von Code in ein Ziel-System ist unter dem Namen Continuous Integration bekannt geworden. Dabei handelt es sich um eine Softwareentwicklungspraktik des DevOps Konzeptes, bei der Entwickler ihre Code-Lösungen häufig integrieren. Jedoch muss dieser Prozess gut überwacht werden, da es zu riskant ist, fehlerhaften Code in die Produktionsumgebung zu integrieren. Aus diesem Grund durchläuft jede Code-Integration einen automatisierten Kompilierungsprozess und Test, um Fehler so schnell wie möglich

zu entdecken. Continuous Integration, auch als CI abgekürzt, beinhaltet eine umfassende Automatisierung typischer Arbeitsschritte eines Entwicklers, wie z.B. Kompilieren, Testen, Archivieren, Verteilen und Berichten. In Abbildung 1 wird ein Entwicklungsprozess in einem minimalen CI-System dargestellt.

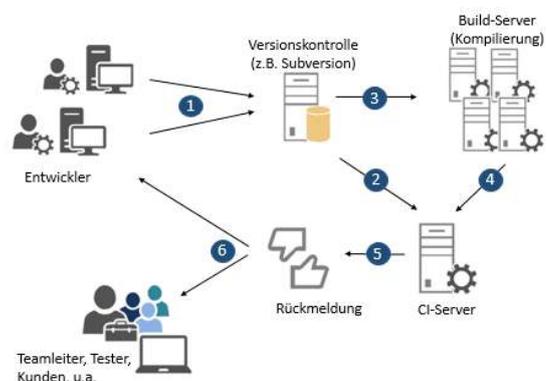


Abbildung 1: Entwicklungsprozess in einem minimalen CI-System.

Neue Code-Lösungen werden zunächst auf den Rechnern der Entwickler entwickelt. Nach einer lokalen Überprüfung des Entwicklers findet eine Übertragung der Codeänderungen in ein zentrales Versionskontrollsystem, wie z.B. Subversion statt (1). Der CI-Server überwacht das Versionskontrollsystem und stellt fest, dass eine Codeänderung vorgenommen wurde (2). Darauf wird die Kompilierung, d.h. ein Build auf den Build-Servern veranlasst (3). Ist der Build-Prozess abgeschlossen, werden Ergebnisse, wie Programme, Testberichte und Dokumentationen an den CI-Server zur Auswertung übertragen (4). Nach der Auswertung werden die Entwickler über geeignete Kommunikationskanäle vom CI-Server über das Resultat informiert, dies kann über E-Mail oder einen sogenannten Issue-Tracker geschehen (5), (6). Ist die Rückmeldung positiv ausgefallen kann der Entwickler an weiteren Komponenten der Software arbeiten. Sind jedoch

*Diese Arbeit wurde durchgeführt bei der Firma Vector Informatik GmbH, Stuttgart

Fehler aufgetreten, kann der Entwickler sofort mit der Fehlerbehebung beginnen und den CI-Zyklus erneut von vorne anstoßen. Des Weiteren können auch andere interessierte Parteien, wie Tester, Kunden oder Vorgesetzte durch den schnellen Feedback-Mechanismus über den aktuellen Fortschritt der Softwareentwicklung informiert werden.

Softwareentwicklung ohne CI

Ein Release, welches in einer Umgebung ohne CI-System verwirklicht wird, wird von den Entwicklern sowie den Release Managern i.d.R. als ein Ereignis mit gewissen Risiken betrachtet. Da es sich bei der Integration nicht tatsächlich um Programmieren handelt, wird der Integrationsprozess auch nicht als ein Teil des Entwicklungsprozesses gesehen. Diese Sichtweise der Entwickler kann dazu führen, dass sie den nahezu gesamten Release-Zeitraum zum Entwickeln investieren, obwohl in diesem Zeitrahmen das Deployment sowie die Integration stattfinden soll. Aus diesem Grund kann es zu Verzögerungen im Release-Lebenszyklus kommen und eine späte Integration der Code-Lösungen stattfinden. Die nicht automatisierten Tests reichen nicht aus um die gesamte Code-Lösung abzudecken und schlagen fehl. Die nur spärlich vorhandenen Tests werden kaum oder gar nicht dokumentiert. Diese menschlichen Nachlässigkeiten führen dazu, dass Fehler nur schwierig oder nicht reproduzierbar sind. Die von Hand erzeugte Dokumentation des Releases erfolgt häufig viel später als die vorgenommenen Änderungen an dem Softwareprojekt. Als Resultat erhält man i.d.R. eine Dokumentation in der ca. 80% der Änderungen erfasst sind, aber nicht abschätzen kann welche 20% der Dokumentation noch brauchbar ist. Diese Nachlässigkeiten können zu Flüchtigkeitsfehlern führen, die einfach nur lästig sind, im schlimmsten Fall allerdings fatale Auswirkungen haben. Die Abhängigkeiten zwischen mehreren Entwicklern und einzelnen Rechnern führen zu Single Point of Failures [1].

In Abbildung 2 wird ersichtlich, wie ein Entwicklungsprozess ohne ein integriertes CI-System aufgebaut ist. Der Entwickler entwickelt die Software entweder auf seinem Rechner oder auf einem Entwicklungsserver und überträgt diese in die Versionskontrolle (1). Darauf importiert der Tester die entwickelte Softwarekomponente manuell in die Testumgebung und führt manuelle, halbautomatisierte oder automatisierte Tests durch (2). Wird ein Fehler entdeckt, so beginnt der Entwickler mit der Behebung des Fehlers auf seiner Entwicklungsumgebung und überträgt sie erneut in die Versionskontrolle. Dieser Prozess wird durchlaufen bis kein Fehler in der jeweiligen Softwarekomponente auftritt (3). Konnten die Tests erfolgreich abgeschlossen werden, wird die Software-Lösung in die Produktivumgebung integriert (3). Die Integration wird häufig manuell durchgeführt. Dies geschieht in der Regel durch ein Script, das die Software kontrolliert in die Produktivumgebung einbettet.

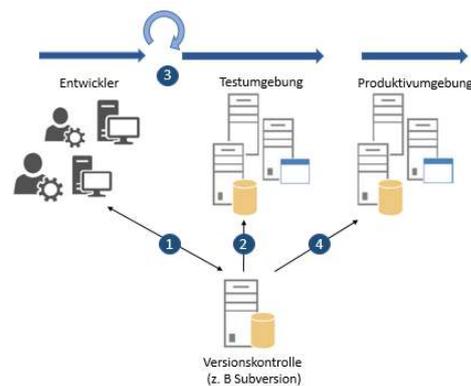


Abbildung 2: Entwicklungsprozess ohne CI-System.

[1] Wiest, Simon, 2011. Continuous Integration mit Hudson – Grundlagen und Praxiswissen für Einsteiger und Umsteiger; 1. Auflage. Heidelberg: dpunkt Verlag GmbH. ISBN 978-3-98864-690-1

Bildquellen:

- Abbildung 1: In Anlehnung an: Wiest, Simon, 2011. Continuous Integration mit Hudson – Grundlagen und Praxiswissen für Einsteiger und Umsteiger; 1. Auflage. Heidelberg: dpunkt Verlag GmbH. ISBN 978-3-98864-690-1
- Abbildung 2: Eigene Darstellung

Einführung in Microservices: Ein Architekturparadigma zur Entwicklung moderner Softwaresysteme

Sara Cizmic*, Manfred Dausmann, Kevin Erath

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Motivation

Die Softwareentwicklung ist ein kontinuierlicher Prozess. Alle erfolgreichen Softwareprodukte entwickeln sich im Laufe der Zeit weiter, um auf veränderte Anforderungen oder den technologischen Wandel zu reagieren. In der Praxis führt das oft dazu, dass Software immer größer und komplexer wird und die Anzahl der internen Abhängigkeiten zunimmt. Dieser Prozess führt ohne entsprechende Gegenmaßnahmen zu vielen Problemen, nicht zuletzt zu einer schlechteren Wartbarkeit.

In diesem Zusammenhang erfreuen sich Microservices immer größerer Beliebtheit. Die Microservice-Architektur soll den bekannten Problemen stetig größer und komplexer werdender Softwareprojekte entgegenreten. In diesem Artikel werden die grundlegenden Charakteristika und die daraus resultierenden Vor- und Nachteile von Microservices vorgestellt.

Definition

Obwohl es bis heute keine einheitliche Definition des Begriffs *Microservice* gibt, existiert ein konformes Grundverständnis hinsichtlich verschiedener Eigenschaften eines Microservices beziehungsweise dieses Architekturparadigmas [1].

Microservices sind ein Modularisierungsansatz [2]. Mit anderen Worten beschreiben Microservices eine Architektur zur Entwicklung von Serveranwendungen bestehend aus leichtgewichtigen, voneinander weitgehend entkoppelten Services. Dabei läuft jeder Service isoliert in einem eigenen Prozess und kommuniziert mit anderen Services über sprach- und plattformunabhängige Schnittstellen.

Größe und Umfang eines Microservices

Anders als das Präfix *Micro* möglicherweise suggeriert, muss ein Microservice nicht unbedingt klein sein bezüglich der Größe seiner Codebasis. Vielmehr bezieht sich das *Micro* auf den Funktionsumfang, den ein Microservice umfassen soll. Ein Microservice ist für eine

Geschäftsfunktion zuständig. Die grundlegende Idee eines Microservices ist somit unter anderem angelehnt an die Unix-Philosophie „Do one thing and do it well“ [3].

Lose Kopplung

Beim Design der Microservices muss immer darauf geachtet werden, dass möglichst keine direkten Abhängigkeiten zu anderen Microservices entstehen und dementsprechend eine lose Kopplung zwischen den Services bestehen bleibt [4]. Eine mögliche Herangehensweise, wie diese lose Kopplung geschaffen werden kann, beschreibt Robert C. Martin Single-Responsibility-Principle (SRP): „*Fasse Dinge zusammen, die aus demselben Grund geändert werden und trenne Dinge, die aus unterschiedlichen Gründen geändert werden*“ [5]. Durch das SRP werden direkte Abhängigkeiten minimiert und eine lose Kopplung unterstützt. Das SRP trägt außerdem maßgeblich zur Bestimmung der richtigen Servicegröße bei [6].

Autonomie

Jeder Microservice läuft in einem eigenen Prozess und ist dadurch von anderen Services isoliert. Aufgrund dieser Isolation und der losen Kopplung kann jeder Microservice unabhängig von anderen Services entwickelt, getestet und veröffentlicht werden. Außerdem kann ein Microservice vergleichsweise einfach durch eine Neuimplementierung ersetzt werden [2].

Da die Services miteinander über sprach- und plattformunabhängige Schnittstellen kommunizieren, kann jeder Microservice andere Technologien nutzen (Framework, Datenbank, usw.). Das vereinfacht die Einführung neuerer Technologien und verhindert das Veralten des Technologie-Stacks. Oft variiert auch die innere Architektur von Service zu Service, da für einen Microservice stets das Architekturmuster gewählt werden soll, das am Besten zu seinen Anforderungen passt [4].

Durch die Autonomie der Services sind sie unabhängig voneinander skalierbar. Es ist möglich, flexibel nur die Teile der Anwendung zu skalieren, die gerade ausgelastet sind oder

*Diese Arbeit wurde durchgeführt bei der Firma IT-Designers GmbH, Esslingen am Neckar

mehr Rechenleistung benötigen. Dadurch werden Hardware-Ressourcen gespart und Kosten reduziert.

Dezentrale Datenhaltung

Um direkte Abhängigkeiten zu vermeiden, ist zudem eine dezentrale Datenhaltung wichtig, das heißt, jeder Microservice hat seine eigenen Daten und teilt sich idealerweise keine Datenbank mit anderen Services. Bei dieser Art der verteilten Datenhaltung liegen einige Daten redundant in mehreren Microservices vor. Dementsprechend muss die Konsistenz durch bestimmte Mechanismen gewährleistet werden, sprich wenn sich die Daten in Microservice 1 ändern, so müssen diese Änderungen in allen anderen Microservices mit den selben Daten nachgezogen werden. Abbildung 1 zeigt eine Möglichkeit, wie redundante Daten in einer dezentralen Datenhaltung bei Microservices mit Hilfe der asynchronen Kommunikation und dem Publish-Subscribe-Muster konsistent geändert werden können.

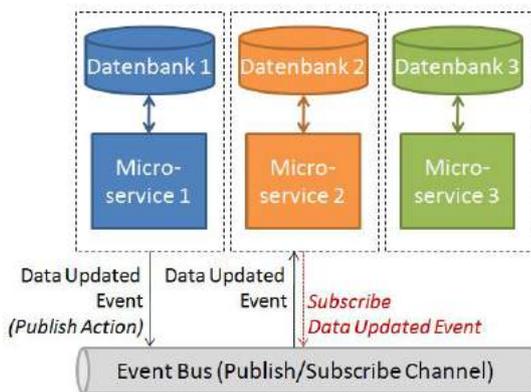


Abbildung 1: Dezentrale Datenhaltung und asynchrone Kommunikation bei Microservices

Kommunikation zwischen Client und Microservices

In der Microservice-Architektur verlagert sich die Komplexität der Anwendung auf das korrekte Zusammenspiel der Services [7]. Einerseits ist eine reibungslose interne Kommunikation für dieses Architekturparadigma unabdingbar. Andererseits ist der Aspekt der Kommunikation zwischen Client und Anwendung bei der Microservice-Architektur auch nicht trivial.

Zur Verdeutlichung der Problematik wird das Szenario einer Webanwendung zum Video-Streaming betrachtet. Dabei werden auf einer Webseite verschiedene Funktionen bereitgestellt, wie etwa ein Videoplayer, der das Video abspielt, Informationen über das Video (Titel, Aufrufe, Kategorie, ...), eine Kommentar- sowie eine Bewertungsfunktion.

Angenommen die Streaming-Plattform ist in der klassischen Schichtenarchitektur implementiert. Wenn ein Client nun die oben beschriebene Beispiel-Webseite aufrufen möchte, ist dazu ein einziger HTTP Aufruf nötig, beispielsweise `GET api.streamingplatform.com/~playvideo?v=videoId`. Die Anwendung führt anschließend die Logik inklusive Datenbankabfragen durch und sendet die Daten als Antwort an den Client.

Ist die Anwendung anhand von Microservices implementiert, so könnte jeder der oben gelisteten Funktionen der Webseite in einem anderen Microservice implementiert sein. Theoretisch könnte der Client beim Aufrufen der Seite nun Requests direkt an die verschiedenen Microservices senden. Da dieser Ansatz jedoch viele Probleme mit sich bringt und daher eher unüblich ist [8], wird stattdessen in der Regel ein *API Gateway* eingesetzt (siehe Abbildung 2).

Ein API Gateway ist ein Service, der einen zentralen Einstiegspunkt in das System zur Verfügung stellt [4]. Der Service nimmt alle Client-Requests entgegen und leitet dann Anfragen an die entsprechenden Microservices weiter. Wenn bei einem Request die Daten mehrerer Microservices abgefragt werden müssen, ruft das API Gateway die betreffenden Microservices auf, sammelt die Daten und sendet sie anschließend gebündelt als Response an den Client. Für das oben genannte Streaming-Plattform-Beispiel bedeutet das, dass der Client, genauso wie bei der monolithischen Anwendung, einen einzigen Request abschicken kann, der intern vom API Gateway zerlegt und an die einzelnen Microservices weitergeleitet wird. Dann kombiniert das API Gateway die Daten und schickt sie an den Client. Dadurch ist die innere Aufteilung in Microservices für den Client nicht ersichtlich.

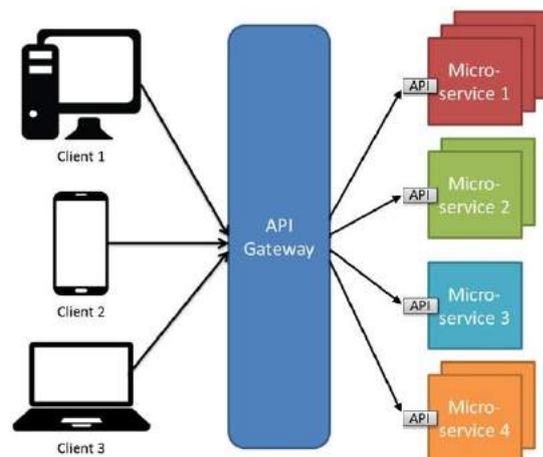


Abbildung 2: Prinzip des API Gateways

Fazit

Auf den ersten Blick bietet die Microservice-Architektur viele Vorteile. Software ist dynamisch skalierbar, weitgehend entkoppelt, Komponenten lassen sich unabhängig voneinander weiterentwickeln und über allem schwebt der Begriff "lightweight". Richtig umgesetzt können Microservices tatsächlich viele bekannte Probleme bei der Entwicklung großer, langwieriger Softwareprojekte minimieren.

Microservices sind jedoch kein Allheilmittel. Es gilt zu beachten, dass die Komplexität und der Aufwand der Softwareentwicklung durch Microservices definitiv nicht abnimmt, sondern sich bestenfalls nur verlagert. Die Schwierigkeiten liegen dann vor allem im Bereich der beträchtlichen Kommunikation zwischen den einzelnen Services

und der verteilten Datenhaltung. Auch Testen, Monitoring und Logging ist aufwendiger. Microservices stellen ein verteiltes System dar, deshalb gelten hier außerdem alle für verteilte Architekturen typischen Herausforderungen, wie Netzwerk-Latenzen, Lastverteilung oder Fehlertoleranz. Sollte ein API Gateway eingesetzt werden, so bringt das auch Nachteile mit sich. Da es die gesamte Routing-Logik und andere wichtige Aufgaben wie Microservice-übergreifende Datenabfragen übernimmt, kann das API Gateway sehr komplex werden. Es kann ein Engpass der Softwareentwicklung sein, weil es fortlaufend von Entwicklern aktualisiert werden muss, um die Schnittstellen jedes Microservices erreichbar zu machen. Außerdem besteht die Gefahr, dass es zum Flaschenhals der Anwendung wird.

-
- [1] Martin Fowler. Microservices. <https://martinfowler.com/articles/microservices.html>. [aufgerufen 21.09.2017]. 2014
 - [2] Eberhard Wolff. Microservices – Grundlagen flexibler Softwarearchitekturen. Heidelberg: dpunkt.verlag, 2015
 - [3] Irakli Nadareishvili u. a. Microservice Architecture – Aligning Principles, Practices, and Culture. Sebastopol: O'Reilly Media Inc., 2016, S. 98
 - [4] Cesar de la Torre, Bill Wagner und Mike Rousos. .NET Microservices. Architecture for Containerized .NET Applications. v1.0. Washington: Microsoft Corporation, 2017. URL: <https://docs.microsoft.com/en-us/dotnet/standard/microservices-architecture/>
 - [5] Sam Newman. Microservices – Konzeption und Design. Heidelberg: MITP-Verlags GmbH & Co. KG, 2015
 - [6] Masiar Ighani. Vom Monolithen zu Microservices – Eine Architektur Strategie. <http://www.skillbyte.de/vom-monolithen-zu-microservices-eine-architektur-strategie/>. [aufgerufen 25.09.2017]. 2016
 - [7] Lars Röwekamp und Arne Limburg. Der perfekte Microservice. <https://www.heise.de/developer/artikel/Der-perfekte-Microservice-3091905.html>. [aufgerufen 06.10.2017]. 2016
 - [8] Chris Richardson. Building Microservices: Using an API Gateway. <https://www.nginx.com/blog/building-microservices-using-an-api-gateway/>. [aufgerufen 02.10.2017]. 2015

Bildquellen:

- Abbildung 1, 2: Eigene Darstellung

Entwurf und Implementierung einer Diagnoseschnittstelle für einen Automotive-Datenlogger

Onur Demir*, Hermann Kull, Reiner Marchthaler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Die steigenden Anforderungen an Antrieb und Fahrsicherheit durch die Motor- und Getriebebesteuerung, sowie ABS, ESP, Fahrerassistenzsysteme und dem Airbag, sorgen für einen zunehmenden Austausch von Informationen zwischen den Steuergeräten der heutigen Fahrzeuge. Hinzu kommt, dass die Überwachung und Steuerung des Kraftstoffverbrauches und des Abgasverhaltens immer mehr in den Vordergrund rücken. [1] Moderne Fahrzeuge haben zudem dutzende Steuergeräte, die neben einfachen Aufgaben wie der Zentralverriegelung und dem Fensterheber, auch komplexe Aufgaben wie Spurwechsel- und Spurhalteassistenten erledigen. Zusammen mit einem umfangreichen Multimedia-System sorgt das alles für ein komplexes Zusammenspiel von Bussystemen und Steuergeräten. Die große Herausforderung in der Automobilindustrie ist es, diese Kommunikation genormt, sicher, schnell, leicht und günstig zu gestalten.

In der Abbildung 1 sind die Bussysteme abgebildet. Auffällig ist, dass verschiedene Anwendungsbereiche verschiedene Busse haben. Die Triebstrang- und Fahrwerk-Busse benötigen eine hohe Datenrate mit einer sehr kurzen Latenzzeit und eine extrem hohe Fehlersicherheit. Hier kommen High-Speed-CAN und/oder FlexRay zum Einsatz. Auf dem Infotainment-Bus sind die Anforderungen für die Fehlersicherheit und die Latenzzeit mäßig, jedoch sind sie für die Datenrate sehr hoch. Hier wird neben Ethernet der sogenannte MOST-Bus (Media Oriented Systems Transport) eingesetzt. Komponenten wie unter anderem die Tür-, Klima- und Lichtsteuerung sind an dem Karosserie-Bus angeschlossen. Hier ist eine hohe Fehlersicherheit gefordert. Jedoch ist die Datenrate gering, was die Kosten senkt. [2] Dazu eignet sich Low-Speed-CAN.

Die Verbindung dieser On-Board Systeme des Fahrzeugs nach außen erfolgt durch den Diagnosebus (siehe Abbildung 1).

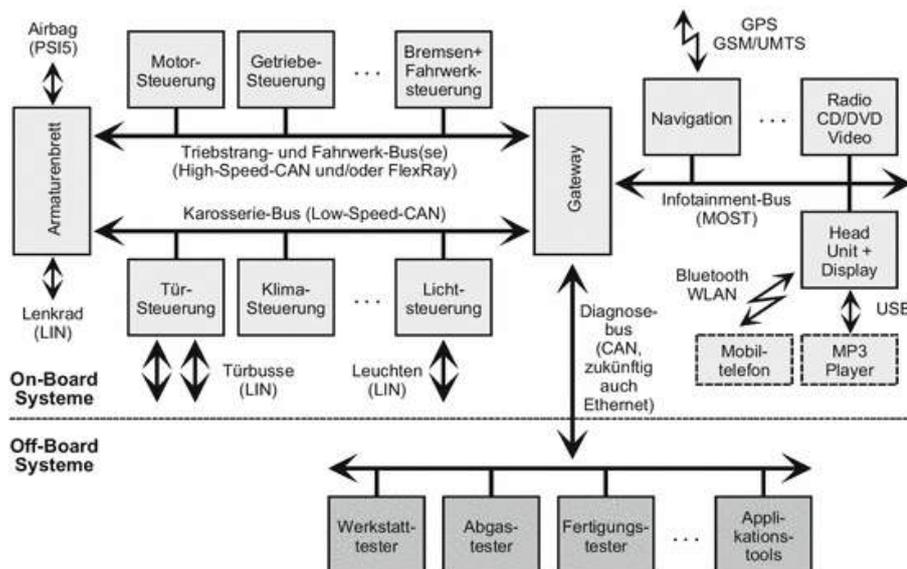


Abbildung 1: Bussysteme in modernen Fahrzeugen

*Diese Arbeit wurde durchgeführt bei der Firma Telemotive AG, Böblingen

Externe Geräte (Off-Board Systeme) können dann unter anderem folgende Aufgaben erfüllen: [2]

- Diagnose-Kommunikation in der Werkstatt und beim Abgastest
- Fertigungstest beim Kfz- und beim Steuergeräte-Hersteller
- Applikation am Prüfstand und im Fahrzeug in der Entwicklungsphase

Die Diagnosetestsysteme sind nach ISO 22900 bzw. ISO 22901 standardisiert. Sie besteht aus einem Diagnosetestechner und einem für verschiedene Bussysteme modular aufgebauten Businterface. Das Businterface wird als Modular Vehicle Communication Interface (MVCI) bezeichnet und übernimmt zwischen den Bussen und dem Diagnosetestechner die protokoll- und busunabhängige Programmierschnittstelle. Die Schnittstelle zwischen Software und Hardware wird als Diagnostic Protocol Data Unit (D-PDU API) bezeichnet. Die Diagnoseanwendung auf dem Diagnosetestechner kann dann über diese D-PDU API mit den Steuergeräten kommunizieren. In der Werkstatt wird dann der Diagnosetestechner über die ODB-2-Buchse (On-Board-Diagnose-2) mit dem Diagnosebus verbunden.

Motivation

Die im Fahrzeug verbauten Datenlogger (genannt blue PiraT) der Telemotive AG haben Schnittstellen für verschiedene Bussysteme. Sie werden unter anderem bei Testfahrten benutzt, um Messwerte und andere Daten aufzuzeichnen und anschließend auszuwerten. Da diese Datenlogger auch am Diagnosebus angeschlossen sind, will die Telemotive AG künftig die Fahrzeugdiagnose als weiteres Logger-Feature anbieten. Das ist möglich, indem man ein D-PDU API für den Datenlogger implementiert.

Die blue PiraT-Logger basieren auf Linux und können über Ethernet oder WLAN (künftig auch Mobilfunknetz) mit dem Internet verbunden werden. So ist es möglich, über die Cloud mit den Loggern Daten auszutauschen. Wird auf dem Logger ein sogenannter RPC (Remote Procedure Call) Server implementiert, ist es möglich, Funktionen über die Cloud auf dem Logger ausführen zu lassen. Das RPC Protokoll beschreibt einen Mechanismus, indem ein Prozess Funktionen aufruft, die in anderen Adressräumen (im Normalfall andere Rechner) ausgeführt werden. Die Verbindung kann über TCP oder UDP erfolgen. Dabei handelt es sich um eine Client/Server Interaktion zwischen den Rechnern. Der Client-Rechner ruft die Funktion auf und bittet den Server-Rechner, diese Funktion auszuführen (Request-Nachricht). Danach wartet der Client auf das Ergebnis vom Server (Response-Nachricht). Das RPC System kümmert sich darum, dem Server mitzuteilen, welche Funktion

mit welchen Parametern ausgeführt werden soll. Die Funktionen und die dazu benötigten Parameter werden mit der sogenannten Schnittstellenbeschreibungssprache (eng. Interface Definition Language) in einer Textdatei definiert.

Hat der Server auf dem Logger und ein beliebiger Client das RPC-Protokoll und die Funktionen der D-PDU-API implementiert, erschließt sich die Möglichkeit, eine Verbindung zu den Steuergeräten im Fahrzeug über das Internet aufzubauen. Das wird in der Abbildung 2 veranschaulicht.

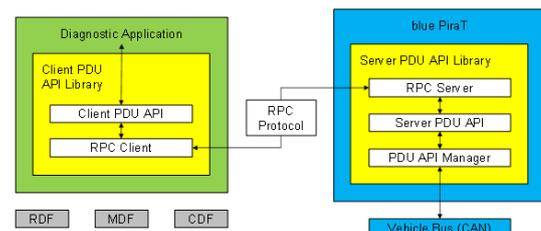


Abbildung 2: Kommunikation mit den Steuergeräten über das RPC-Protokoll

Root description file (RDF), module description file (MDF) und cable description file (CDF) sind Dateien, die für die Kommunikation zwischen der Anwendung und dem MVCI benötigt werden. Sie bestimmen unter anderem welche Bussysteme und Protokolle das Businterface unterstützt und wie die Pin-Belegung auf den Bussen ist. [2]

Analyse

Im Rahmen der Bachelorarbeit galt es im ersten Schritt verschiedene RPC-Varianten zu analysieren. Es gibt eine große Auswahl an quell-offenen RPC Protokollen wie unter anderem gRPC von Google, Apache Thrift und ONC RPC von Oracle. Diese unterscheiden sich in der Grundfunktionalität nicht groß. Alle bieten eine Möglichkeit, Funktionen auf entfernten Servern ausführen zu lassen und deren Ergebnisse zu empfangen. Zusätzliche Eigenschaften wie unter anderem Multithreaded-Servers, Unabhängigkeit von Betriebssystem und Sprache, Verschlüsselungen, Streaming-Eigenschaften und Regelungen der Timeouts bieten die meisten Frameworks auch an. Eine Kommunikation zwischen Server und Client ist also möglich, obwohl sie auf verschiedenen Betriebssystemen und Sprachen laufen. Kleine Unterschiede gibt es meist in den Datentypen, die als Funktionsparameter übergeben und als Ergebnis empfangen werden. Hinzu kommt, dass manche RPC-Bibliotheken mehr Programmiersprachen unterstützen als andere. Aus Sicht des Programmierers ist neben den Lizenzbedingungen auch zu beachten, wie umfangreich die Dokumentation ist und wie verständlich

die API aufgebaut ist. Unter Berücksichtigung dieser Eigenschaften wurde gRPC von Google am geeignetsten für dieses Projekt bewertet. Das High-Performance RPC-Framework von Google zeichnet sich aufgrund seiner für den Programmierer verständlichen und sauberen API aus. Sie bietet das Protokoll für viele Programmiersprachen an und hat eine umfangreiche Dokumentation für Entwickler. Speziell in Bezug auf die Funktionen der D-PDU API hat sie einfache Mechanismen, um Anforderungen zu bewältigen.

Realisierung

Im zweiten Schritt wurden die Funktionen und Parameter der D-PDU API in der gRPC definiert. gRPC benutzt als Interface Definition Language (IDL) die von Google selbst entwickelten Protocol Buffers. Optional können auch Formate wie JSON oder XML benutzt werden. Jedoch werden aus Gründen der Performance Protocol Buffers empfohlen. Protocol Buffers sind zuständig für die Serialisierung der strukturierten Daten. In einer Textdatei mit der Endung .proto werden dann die Funktionen und die Parameter in der von Protocol Buffers definierten IDL beschrieben. Der IDL-Compiler „protoc“ benutzt diese Datei um Code für die Serialisierung zu generieren. Mit dem gRPC-Plugin werden zudem noch sogenannte Stubs generiert. Diese Stubs definieren zudem die gewünschten Funktionen auf Client- und Serverseite und übernehmen die Kommunikation. Bindet man diesen generierten Code in der eigenen Anwendung ein,

kann man das RPC Protokoll nutzen. Die besondere Eigenschaft ist, dass es auf der Clientseite so aussieht, als ob die Funktionen lokal aufgerufen werden.

Die D-PDU API hat 30 Funktionen. Diese Funktionen haben zum Teil sehr komplexe und verschachtelte Strukturen als Parameterwerte. Das macht die Implementation in die IDL besonders komplex. In der Abbildung 3 wird der vereinfachte Ablauf einer Verbindung mit den Steuergeräten gezeigt. [2]

Zusammenfassung

Mit dieser Arbeit wurde eine Möglichkeit geschaffen, eine Kommunikation mit den Steuergeräten eines Fahrzeugs über das Internet aufzubauen. Dazu ist der Datenlogger bluePiraT der Telemotive AG notwendig. Dieser ist an dem Diagnosebus angeschlossen und kann mithilfe der D-PDU API Bibliothek Nachrichten mit den Steuergeräten austauschen. Um über das Internet die D-PDU API Funktionen auf dem bluePiraT aufzurufen, wurde das gRPC Protokoll benutzt. Dieses Protokoll ermöglicht über TCP oder UDP eine Client/Server Kommunikation. In diesem Fall ist der bluePiraT der Server, der die D-PDU API Funktionsaufrufe des Clients ausführt und das Ergebnis zurück sendet. Es ist also nicht mehr zwingend notwendig, die On-Board Systeme des Fahrzeugs über die ODB-2-Buchse zu erreichen.

| | |
|---|--|
| <i>Verbindungsaufbau</i> | |
| PDUConstruct() | Laden der D-PDU API Bibliothek, Erkennen der angeschlossenen Businterface-Module |
| PDURegisterEventCallback() | Registrieren von Rückruffunktionen (optional) |
| PDUCreateComLogicalLink() | Initialisierung einer logischen Verbindung |
| PDUGet/SetComParam() | Abfragen und Setzen von Protokollparametern |
| PDUConnect() | Herstellen der Verbindung vom Tester zum Steuergerät |
| <i>Senden und Empfangen von Daten mit Hilfe von ComPrimitives</i> | |
| PDUStartComPrimitive() | Senden einer Diagnosebotschaft |
| PDUGetEventItem() | Lesen der Antwortbotschaft |
| <i>Verbindungsabbau</i> | |
| PDUDisconnect() | Beenden der Verbindung zum Steuergerät |
| PDUDestroyComLogicalLink() | Löschen der logischen Verbindung |
| PDUDeconstruct() | Entladen der D-PDU API |

Abbildung 3: Vereinfachter Ablauf einer Verbindung mit den Steuergeräten

[1] WIESINGER, Johannes, 2017. Der CAN-BUS – Grundlagen von Automobil Bussystemen [online]. 13.11.2017 [Zugriff am: 20.11.2017].

Verfügbar unter: https://www.kfztech.de/kfztechnik/elo/can/can_grundlagen_1.htm

[2] ZIMMERMANN, Werner und Ralf SCHMIDGALL, 2014. Bussysteme in der Fahrzeugtechnik. 5. Auflage. Wiesbaden: Springer Fachmedien. ISBN 978-3-658-02419-2

Bildquellen:

- Abbildung 1: Aus [2] entnommen für Darstellungszwecke
- Abbildung 2: Telemotive AG, Betriebsinterne Bilderdatenbank
- Abbildung 3: Eigene Abbildung

Analyse der Auswirkungen von Bargeldabschaffung am Beispiel von Mobile Payment

Fabian Flaig*, Catharina Kriegbaum-Kling, Thomas Rodach

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Motivation und Zielsetzung

Am 4. Mai 2016 hat der Rat der Europäischen Zentralbank (EZB) beschlossen die Herstellung und Ausgabe des 500-Euro-Scheins einzustellen. Umsetzung dieses Beschlusses ist auf Ende 2018 datiert [1]. Infolgedessen kamen Spekulationen auf, dass dies als Startschuss für die umfassende Bargeldabschaffung in der Euro-Zone angesehen werden kann.

Durch die immer weiter fortschreitende technologische Entwicklung liegt es nahe, dass im Falle einer Bargeldabschaffung das Mobile Payment mehr in den Fokus rückt. Thomas Lerner definiert das Mobile Payment, in seinem 2013 erschienen Buch, als „das Zahlen mit einem mobilen Endgerät (Mobiltelefon, Smart Phone oder Tablet-PC)“ [2]. Auch die Prognose aus „Abbildung 1: Prognose zum weltweiten Bestand an Smartphones“ zeigt deutlich auf, weshalb das Thema Mobile Payment immer mehr an Aufmerksamkeit gewinnt. Laut dieser Prognose besitzen im Jahr 2020 circa 5,8 Milliarden Menschen auf der Welt ein Smart Phone.

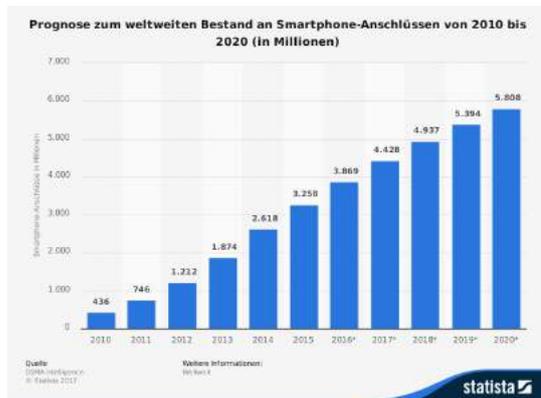


Abbildung 1: Prognose zum weltweiten Bestand an Smartphones

Ziel der Bachelorarbeit „Analyse der Auswirkungen von Bargeldabschaffung am Beispiel von Mobile Payment“ ist es mögliche Folgen, sowohl positive, als auch negative, der Bargeldabschaffung zu betrachten und zu bewerten, inwieweit das Mobile Payment als Sub-

stitutionsgut geeignet ist. Hierbei werden zunächst theoretische Grundlagen und Definitionen aufgezeigt. Daraufhin werden die wirtschaftlichen Aspekte durchleuchtet wie beispielsweise die aktuelle Lage in Deutschland in Bezug auf die Bargeldabschaffung, welche dann unter anderem mit den Ländern Schweden und Japan verglichen wird. Außerdem werden mögliche Geschäftsmodelle des Mobile Payments inklusive deren Interessengruppen analysiert. Ein weiteres Kapitel beschäftigt sich mit den Technologien hinter dem Mobile Payment, wie zum Beispiel Near Field Communication (NFC), Quick Response-Code (QR-Code) oder Bluetooth. Das Thema gipfelt in einer Gegenüberstellung der Chancen und Risiken, die die Abschaffung von Bargeld und das Mobile Payment mit sich bringen.

Geldtheoretische Grundlagen

Das Geld hat drei grundlegende Funktionen. Diese sind Tausch- und Zahlungsmittel-, Wertmess- und Recheneinheits- und Wertaufbewahrungsfunktion. In „Abbildung 2: Die Funktion des Geldes“ werden die einzelnen Funktionen genauer erläutert.

Die Funktionen des Geldes im Überblick

| Zahlungsmittel | Recheneinheit | Wertaufbewahrungsmittel |
|--|---|---|
| Geld erleichtert den Warentausch. Auch Finanztransaktionen wie die Vergabe von Krediten sind möglich. | Güterwerte lassen sich in einer Bezugsgröße ausdrücken und vergleichen. Geld fungiert als Wertmaßstab. | Geldverleih und Geldausgabe können zeitlich auseinanderfallen. Sparen ist möglich. |

Um diese Funktionen erfüllen zu können, muss der Gegenstand, der als Geld verwendet wird, gut teilbar, wertbeständig und allgemein akzeptiert sein.

Abbildung 2: Die Funktion des Geldes

Außerdem kann bei Geld noch zwischen Bargeld, Buchgeld (Giralgeld) und elektronischem Geld unterschieden werden. Bargeld beschreibt dabei sowohl das Münzgeld, als auch die Banknoten. Unter Buchgeld, oft auch

* Diese Arbeit wurde durchgeführt an der Fakultät Informationstechnik

Giralgeld genannt, wird verstanden, dass eine bargeldlose Buchung von Konto zu Konto erfolgt. Beispielhaft kann hierfür eine Überweisung, eine Lastschrift oder eine Kreditkarte angeführt werden. Digitales Geld oder auch elektronisches Geld definiert das Wirtschaftslexikon Gabler wie folgt: „elektronisch, darunter auch magnetisch, gespeicherter monetärer Wert in Form einer Forderung gegenüber dem Emittenten, der gegen Zahlung eines Geldbetrages ausgestellt wird“[3]. Beispiele für elektronisches Geld sind PayPal oder GiroGo.

Das Mobile Payment findet bei dieser Unterscheidung keine explizite Zuordnung, da die Zahlungsweise sowohl auf einem Prepaid-Prinzip wie bei der GiroGo basieren, als auch eine Kreditkarte hinterlegt werden kann.

Situation in Deutschland

Gemäß einer repräsentativen Studie der Deutschen Bundesbank aus dem Jahr 2014 ist Bargeld das Zahlungsmittel, welches überwiegend in Deutschland bevorzugt wird. Der Anteil der Barzahlungen lag dabei im Jahr 2014 bei 79,1% und veränderte sich nur leicht im Vergleich zu dem Wert im Jahr 2011 mit 82,0%. Dies stellt lediglich einen Rückgang von 2,9% dar und untermauert die Präferenz von Bargeld der Bevölkerung in Deutschland. Lediglich fünf der 19.247 getätigten Zahlungen erfolgte per Mobiltelefon [4]. Ebenfalls wurden die Teilnehmer der Studie der Deutschen Bundesbank, welche die Barzahlung präferieren, befragt, weshalb dies so ist. Die meist genannten Argumente waren: Gefühl bessere Ausgabenkontrolle, einfacher und sicherer (vgl. „Abbildung 3: Gründe für Barzahlung“)

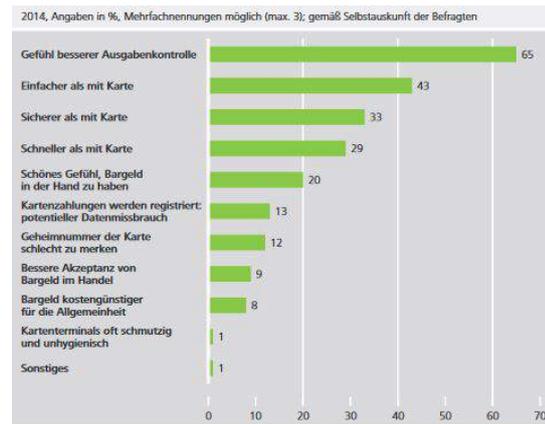


Abbildung 3: Gründe für Barzahlung

Allerdings hat das Bargeld auch seine negativen Seiten, trotz der hohen Beliebtheit bei der deutschen Bevölkerung. Mit dem Bargeld lässt sich einfacher Geld aus kriminellen Aktivitäten waschen, weil durch Bargeld anonyme Zahlungen getätigt werden können. Dadurch sinkt die Hemmschwelle zur Schwarzarbeit. Dies hat zur Folge, dass dem Staat Steuereinnahmen vorenthalten bleiben und die Last die Allgemeinheit zu tragen hat. Außerdem würde mit Bargeld ein Großteil des Terrorismus finanziert werden [5]. Allgemein lässt sich sagen, dass unter dem Schutz der Anonymität, die das Bargeld bietet, die Schattenwirtschaft agieren kann.

Fazit

Abschließend kann festgehalten werden, dass das Misstrauen dem Mobile Payment gegenüber in Deutschland noch überwiegt und die Bevölkerung noch größtenteils auf das Bargeld vertraut. Im weiteren Verlauf der Bachelorarbeit wird noch genauer auf die Auswirkungen eingegangen, um eine Prognose für das Thema aufstellen zu können.

[1] FAZ. (2016). EZB schafft 500-Euro-Scheine ab [Online] [Zugriff am 22.November 2017]. <http://www.faz.net/aktuell/finanzen/devisen-rohstoffe/ezb-schafft-500-euro-schein-ab-14216175.html>

[2] Lerner Thomas. (2013). Mobile Payment – Technologien, Strategien, Standards und Fallstudien. Mainz: Springer Vieweg

[3] Wirtschaftslexikon Gabler. (2016). E-Geld [Online] [Zugriff am 22.November 2017]. <http://wirtschaftslexikon.gabler.de/Definition/e-geld.html>

[4] Deutsche Bundesbank. (2015). Zahlungsverhalten in Deutschland 2014 [Online] [Zugriff am 22. November 2017]. https://www.bundesbank.de/Redaktion/DE/Downloads/Veroeffentlichungen/Studien/zahlungsverhalten_in_deutschland_2014.pdf?__blob=publicationFile

[5] Hierl Ludwig. (2017). Mobile Payment – Grundlagen–Strategien–Praxis. Heilbronn: Springer Gabler

Bildquellen:

- Abbildung 1: <https://de.statista.com/statistik/daten/studie/312258/umfrage/weltweiter-bestand-an-smartphones/>
- Abbildung 2: https://www.bundesbank.de/Redaktion/DE/Dossier/Service/schule_und_bildung_kapitel_1.html?docId=153022¬First=true
- Abbildung 3: https://www.bundesbank.de/Redaktion/DE/Downloads/Veroeffentlichungen/Studien/zahlungsverhalten_in_deutschland_2014.pdf?__blob=publicationFile

Entwurf und Implementierung eines webbasierten Systems zur Identifikation von Fenstern über QR-Codes

Roland Fronczek*, Dominik Schoop, Kai Warendorf

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Im Zeitalter des mobilen Web ist es unheimlich wichtig, als Unternehmen nicht den Kontakt zu seinen Kunden zu verlieren und auch sein Produkt mit dem Web zu verbinden. Eine Möglichkeit, die Kundenzufriedenheit zu steigern und den Dialog aufrecht zu erhalten, kann hierbei sein, dem Kunden eine einfache Kontaktaufnahme mittels Web-Kontaktformular im Reklamationsfall anzubieten.

Auf diesen zwei Grundlagen entstand bei der WERU GmbH die Idee, die Produkte mit QR-Codes zu bestücken. Vor allem der Kontakt des Endkunden zum Kundendienst ist nach heutigem Stand der Technik überholt, da dieser bisher nur über die Telefonhotline möglich war. Auch die individuelle Identifikation der Produkte (Fenster und Türen) durch den Händler und den Kundendienst war bisher nur über mehrere Umwege möglich. Mittels des QR-Codes soll es dem Kunden nun möglich gemacht werden, Kontakt zum Fachhändler aufzunehmen. Auch soll der jeweilige Händler ebenso wie auch der Kundendienst die einfache Möglichkeit erhalten, schnell Informationen über ein bestimmtes Fenster zu erhalten. Dies soll mittels einer zu entwickelnder Software geschehen, mit der man dem Endkunden ein Kontaktformular zur Verfügung stellt und dem Fachhändler mittels eines Smartphones Informationen über das jeweilige Produkt darstellt.

Die Erstellung der QR-Codes und die Entwicklung einer Software, die diesen Zielen gerecht wird, ist Bestandteil des Projekts.

Grundlagen des QR-Codes

Gerade im Alltag sind QR-Codes allgegenwärtig. Sie haben nicht nur in der Industrie Einzug gehalten, um dort die Identifikation von Elementen im Arbeitsablauf zu erleichtern.

Auch im täglichen Leben begegnet man ihnen auf Werbeflächen, Lebensmittelverpackungen und auch auf Eintrittskarten. Gerade zu Marketingzwecken sind sie ein gern gesehenes Mittel.



Abbildung 1: Datenteil innerhalb eines QR-Codes

QR-Codes können zur Weiterleitung an eine URL genutzt werden, wie im Projekt verwendet. Ebenso können sie aber auch zur Weitergabe von Kontaktdaten mittels VCard, Plain Text oder auch Standorten verwendet werden. Ausgelesen werden können QR-Codes sowohl mit speziellen Lesegeräten als auch mit handelsüblichen Smartphones mit zusätzlich installierter Software.

Die gängige Version 2 des QR-Codes kann bis zu 7089 Ziffern oder 4296 alphanumerische Zeichen enthalten [1]. Hierbei werden die Ziffern binär in der schwarz-weißen Fläche des Codes codiert. Durch die drei Positionierungsmarken an den Ecken erkennt das Lesegerät, in diesem Fall das Smartphone, die Positionierung des QR-Codes. Zusätzlich ist in dem Code noch eine Marke zur Ausrichtung enthalten. Diese befindet sich in dem rechten unteren Viertel. Auch befindet sich eine Information über die Version des QR-Codes jeweils neben den Positionsmarken. Die eigentliche Information befindet sich größtenteils innerhalb des mittleren Teils des QR-Codes (rote Markierung in Abb. 1).

Ein großer Vorteil der Codes ist dessen Fähigkeit zur Fehlerkorrektur. Das heißt, je nach Stufe kann der Code auch noch gelesen und vollständig decodiert werden, wenn Teile davon beschädigt sind oder ganz fehlen. Die Fehlerkorrektur ist in vier Levels unterteilt. Innerhalb des Levels L (für Low) können nur sieben Prozent der Daten wiederhergestellt werden. Der Level M (Medium) kann 15 Prozent und der Level Q (Quartile) 25 Prozent der Daten wiederherstellen. Der höchste Level zur Fehlerkorrektur innerhalb eines QR-Codes ist der Level H (High), welcher bis zu 30 Prozent der Daten wiederherstellen kann.

*Diese Arbeit wurde durchgeführt bei der Firma WERU GmbH, Rudersberg

Umsetzung der Benutzeroberfläche

Der Umfang des Projekts umfasst in der ersten Stufe Grundlagen der zu erstellenden QR Codes und die Entwicklung eines Entwurfs der Oberfläche. Durch die Analyse der erstellten Personas und nach Gesprächen mit verschiedenen Kundengruppen wurde festgestellt, dass die Zielgruppe der Software eher wenig technikaffin ist. Aufgrund dessen wurde Wert auf eine sehr einfach benutzbare Informationsstruktur der Software gelegt.

Gerade durch die Vorgabe zur Benutzung von QR-Codes sollte die Benutzerschnittstelle nach den Grundsätzen des Designprinzips „mobile first“ entworfen werden. Dies gestaltet sich auch dadurch sinnvoll, dass die Fachhändler oftmals nur ein Smartphone mit zum Kunden nehmen. Zur Entwicklung der graphischen Oberfläche wurden anschließend zwei Anwendungsfälle ausgearbeitet.

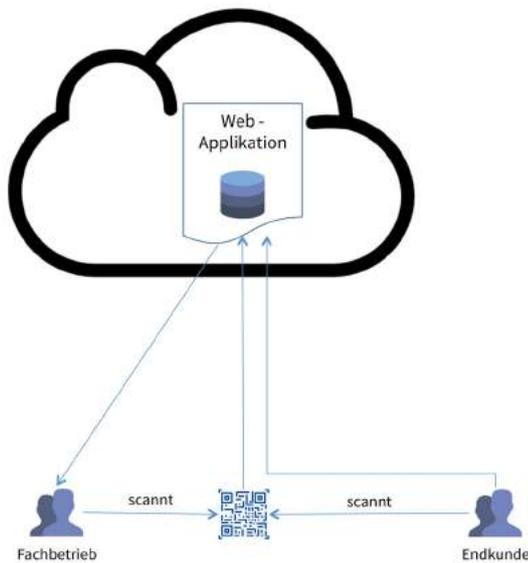


Abbildung 2: Anwendungsfall

Der erste Anwendungsfall zielt hierbei auf die Verwendung durch den Endkunden, also den Besitzer der Fenster ab. Hierbei wurde die Annahme getroffen, dass dieser sich nicht für die Daten des vorliegenden Produkts interessiert, sondern eher die Kontaktmöglichkeit zum Fachhändler sucht. Der zweite Anwen-

dungsfall hingegen bezieht sich auf die Zielgruppe der Fachhändler. Dieser braucht spezifische Daten zum Produkt, um bestimmte Entscheidungen zur Reparatur treffen zu können. Jedoch führt dieser Anwendungsfall zu dem Problem, dass hierbei sensible Kundendaten, wie zum Beispiel die Adresse des Kunden oder auch sicherheitsrelevante Daten über das Fenster, angezeigt werden müssen.

Wie in Abbildung 2 gezeigt, wird derselbe Barcode in beiden Anwendungsfällen per Smartphone gescannt. Im Anwendungsfall des Endkunden wird diesem nach kurzer Angabe von Daten wie Name und E-Mail-Adresse die Möglichkeit gegeben, per Kontaktformular, seinen zuständigen Fachhändler zu verständigen. Hierbei braucht er sich nicht um die Angabe des produktspezifischen Schlüssels zu kümmern, da dieser schon mittels des QR-Codes übergeben wurde.

Der Fachhändler kann nach erfolgreichem Login Daten über das vorliegende Fenster auslesen. Auch hierbei wurde der Schlüssel über den QR-Code übergeben. In diesem Anwendungsfall wurde auf eine sinnvolle Informationsarchitektur geachtet, um die Daten gerade auf dem Smartphone lesbar und gut verständlich darstellen zu können.

Ausblick

Die Implementierung des server-seitigen Backends ist nicht im Rahmen des Bachelor-Projekts enthalten. Diese soll nach dem Projekt geschehen, um die Software schlussendlich auch in den produktiven Betrieb übernehmen zu können. Auch soll hierzu eine automatisierte Übertragung der Daten von der WERU-eigenen Datenbank auf die Datenbankstruktur des Webservers entwickelt werden. Die Entwicklung zur manuellen Eingabe des Codes soll als zweiter Schritt entwickelt werden. Hierbei wird auf all die Produkte eingegangen, welche nicht die Möglichkeit der Anbringung eines QR-Codes bieten. Hierzu zählen vor allem Festverglasungen, welche keinen Platz zur unauffälligen Platzierung des QR-Codes haben. In diesen Fällen soll im Scheibenzwischenraum des Fensters der alphanumerische Schlüssel angebracht werden, welcher in ein Input-Feld eingeben werden kann.

[1] <http://www.qrcode.com/en/about/standards.html> (aufgerufen am 22.11.2017)

Bildquellen:

- Abbildung 1, 2: Eigene Abbildung

Möglichkeiten zur Effizienzsteigerung modellbasiert entwickelter Steuergerätesoftware für Kleinmotoren

Luis Garcia Barth*, Kai Warendorf, Dominik Schoop

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Ziel der Arbeit ist es Richtlinien festzulegen die die Effizienz von automatisch generiertem Quellcode erhöhen. Dabei sollte beachtet werden, dass die Struktur des Modells nicht unübersichtlich oder unnötig komplex wird. Verschiedene Bewertungskriterien helfen dabei. Diesen Kompromiss zu finden ist die Herausforderung.

Modellbasierte Softwareentwicklung

Die modellbasierte Softwareentwicklung ist heute Standard. Sie dient dazu automatisch Quellcode aus einem Modell zu generieren. So können komplexe Softwarestrukturen übersichtlich dargestellt werden. Des Weiteren können Simulationen und Tests wie Processor-in-the-Loop (PiL), Software-in-the-Loop (SiL) und Model-in-the-Loop (MiL) durchgeführt werden.

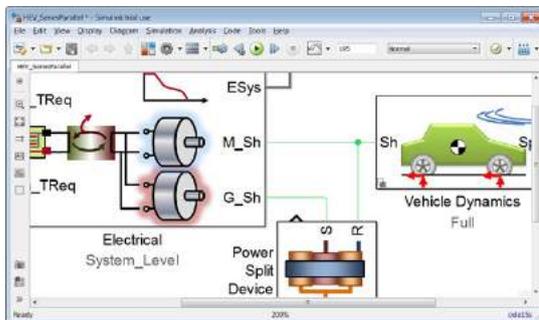


Abbildung 1: Modell eines Hybridelektroantriebs in Simulink

Bei der PiL- und der SiL-Simulation wird automatisch Quellcode generiert. Bei SiL wird dieser auf dem PC ausgeführt, bei PiL direkt auf dem Mikrocontroller. Dadurch lassen sich Laufzeit und Speicherverbrauch messen. Bei der MiL-Simulation handelt es sich um einen reinen Modell-basierten Test auf dem PC. Bei allen Simulationen wird ein "Testvektor" genutzt um den Betrieb eines Steuergerätes zu simulieren. Durch starten der Simulink-Simulation wird das Modell bzw. die Software getestet. Die Simulationsergebnisse lassen sich anschaulich in Graphen darstellen.

Durch die modellbasierte Softwareentwicklung sind Problembeschreibungen wesentlich klarer, einfacher und weniger redundant festgehalten. Der erhöhte Abstraktionsgrad der Modelle hilft die Software wesentlich klarer umzusetzen und vereinfacht die Wartung.

Um dennoch effizienten Quellcode zu erhalten muss sichergestellt werden, dass gewisse Richtlinien eingehalten werden. Beispielsweise gibt es die *TargetLink Modelling Guidelines* [1] oder die *MAAB Style Guidelines* [2]. Darin ist genau definiert wie ein Modell aufgebaut sein sollte um effizienten Quellcode daraus zu erzeugen.

Motivation

Bei dem Entwurf von Steuergerätesoftware ist effizienter Quellcode unerlässlich. Es müssen nicht nur viele, sondern auch teilweise komplexe Berechnungen durchgeführt werden für die nur ein gewisser Zeitraum (Slot-time) zur Verfügung steht. Das Ziel ist es, ein Modell zu entwerfen, das nicht nur funktional sondern auch praktisch umsetzbar ist.

Es stehen nur begrenzt Ressourcen zur Verfügung. Einer der Gründe ist, dass in Seriensteuergeräten leistungsschwächere Mikrocontroller verwendet werden, da diese günstiger sind. Bei effizienter Software versucht man demnach so wenig Ressourcen wie möglich zu verbrauchen. Darunter zählen Laufzeit, Codegröße und Speicherverbrauch auf dem Stack, RAM und ROM. Die Ressource Laufzeit kann man sich so vorstellen, dass die Berechnungen zweimal pro Arbeitsspiel durchgeführt werden, woraus eine sehr kurze Slot-time resultiert. Reicht diese nicht mehr aus, benötigt man einen Mikrocontroller mit höherer Taktfrequenz, diese sind wiederum kostenlastiger, verbrauchen mehr Energie und erzeugen mehr Wärme.

Da die Anforderungen an die Software ständig steigen, muss diese immer weiterentwickelt werden. Neue Funktionen kommen dazu und komplexere Berechnungen müssen durchgeführt werden. Dafür wird auch entsprechend mehr Zeit benötigt. Effizienter Quellcode hilft also dabei Kosten zu sparen und sorgt gleichzeitig für höhere Qualität.

*Diese Arbeit wurde durchgeführt bei der Firma ANDREAS STIHL AG & Co. KG, Waiblingen

Die Effizienz einer Software lässt sich anhand verschiedener Aspekte ermitteln (siehe Abschnitt Analyse). Damit bei diesem Vorgehen nicht die Vorteile der Modellbasierten Softwareentwicklung verloren gehen, muss die Komplexität so gering wie möglich bleiben.

Vorgehensweise

Als Tool zur Modellierung wird MATLAB/ Simulink eingesetzt. TargetLink bringt hauptsächlich einen Codegenerator mit. Dieser erlaubt SiL- und PiL-Simulationen. Zusätzlich ergänzt TargetLink die Simulink Blöcke um eine zusätzliche Oberfläche um Einstellungen für die Codegenerierung vorzunehmen. Die Hardware besteht aus einem Evaluation Board bestückt mit einem Mikrocontroller.

Als erster Schritt wird ein Teilmodell der Steuergerätesoftware ausgewählt, der Leerlaufregler. Dieser enthält zwei PI-Regler, komplexe Berechnungen, Stateflow-Diagramme, C-Include Blöcke und eine Parameter-Datenbank. Genug um die meisten Testfälle abzudecken und nicht zu komplex für die Analyse zu sein.

Im nächsten Schritt werden einfache Testmodelle erstellt, welche nur aus wenigen Blöcken und genau einem Kriterium bestehen. Kriterien wie die *Konsistenz von Skalierungen*, *Vermeidung von Stateflow Rekursionen* und *Vermeidung von 64-Bit Operationen* werden untersucht. Hierfür wird eine PiL-Simulation durchgeführt und die Messergebnisse mit und ohne Kriterium werden miteinander verglichen, wenn sich das Einhalten eines dieser Kriterien als positiv erweisen sollte wird es im

Leerlaufregler implementiert. Zusätzlich wurden die Änderungen im Quellcode analysiert, um Abweichungen bei den Messwerten zu erklären.

Analyse

Es wurden verschiedene Kriterien bei der Qualitätsanalyse betrachtet. Die "Lesbarkeit" des Modells wird in Form eines Komplexitätslevels dargestellt und mit dem Tool **MES M-XRAY®(MXRAY)** gemessen [3]. Dieses Tool führt eine ISO 26262-konforme Analyse der Modellarchitektur durch und garantiert somit eine effiziente Architektur.

Neben der Komplexität werden direkt in TargetLink folgende Werte durch die PiL-Simulation gemessen

- Laufzeit in μs
- Stackgröße in Byte
- RAM in Byte
- ROM in Byte

Das Resultat sind z. B. die Messwerte in Abbildung 2. Die Messwerte werden bei jeder Änderung am Modell erzeugt und mit denen des Modellausgangszustands verglichen. Da sich bei jeder Änderung am Modell auch der generierte Code ändert, erfolgt zusätzlich eine Analyse des Quellcodes. Dieser muss bestimmte Aspekte erfüllen. Eine große Rolle spielen Typwandlungen. Sie treten immer dann auf, wenn zum Beispiel der Wertebereich einer Variable nicht mehr ausreicht oder wenn die Skalierung mehrerer Festkommazahlen nicht einheitlich durchgeführt wurde.

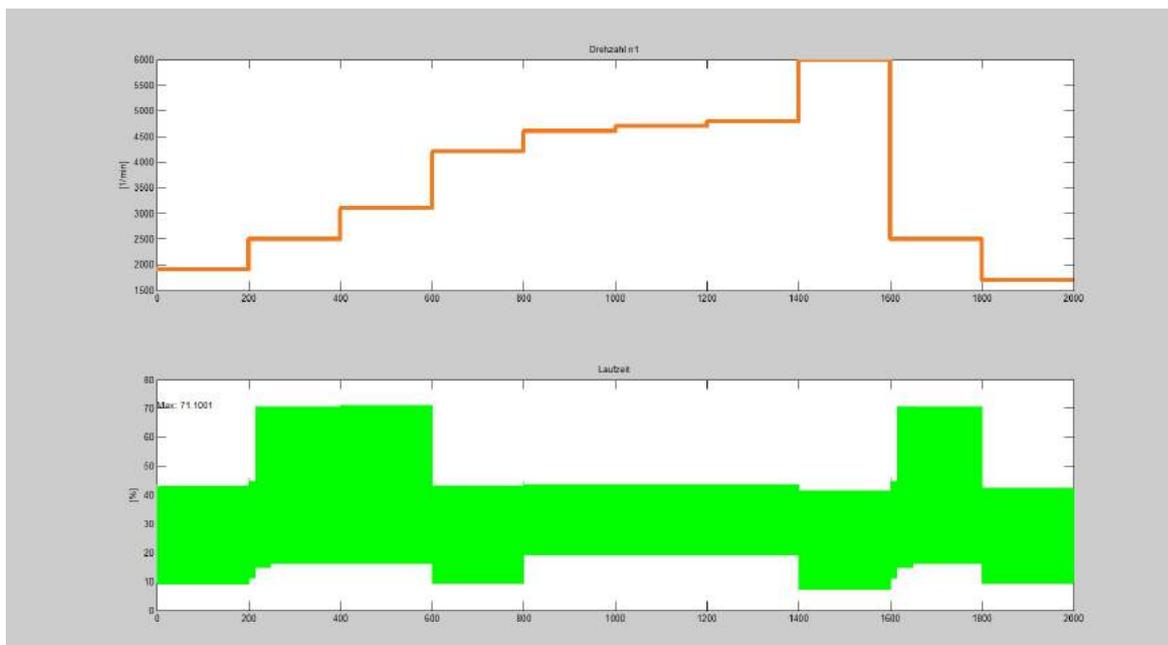


Abbildung 2: Messwerte Drehzahl und Laufzeit über Arbeitsspiele im PiL-Modus

Ein gutes Beispiel ist die Konsistenz von Skalierungen. Diese Richtlinie beschreibt das richtige Skalieren von Festkommazahlen. Festkommazahlen werden häufig genutzt wenn nur begrenzte Ressourcen zur Verfügung stehen, da mit ihnen schnell gerechnet werden kann und sie wenig Speicher benötigen. Auf leistungsfähigeren Prozessoren, wie sie bei PC's zum Einsatz kommen wird in der Regel mit Fließkommazahlen gerechnet. Diese liefern präzisere Ergebnisse. Der Nachteil an Festkommazahlen ist, dass skaliert werden müssen. Dabei nimmt man einen Faktor 2^{-n} mal der zu skalierenden Ganzzahl. Die Basis 2 wird gewählt, da so der Mikrocontroller mit Bits-hifts arbeiten kann. Hat man nun beispiels-

weise zwei dieser Festkommazahlen mit unterschiedlicher Skalierung und möchte diese adaptieren, muss zuerst die Skalierung gleichgestellt werden. Dieser Vorgang kostet Zeit und sollte vermieden werden.

Diese unerwünschten Effekte fallen nur auf, wenn man den Quelltext analysiert. Die modellbasierte Softwareentwicklung konzentriert sich allerdings mehr auf das Entwerfen von Software als auf die Überprüfung von Quelltext. Deshalb dienen Richtlinien und Vorschriften sowie Validierungs-Tools als Hilfestellung zum Erreichen von effizientem Quellcode und somit zu effizienter Software.

[1] https://www.dspace.com/de/gmb/home/support/kb/supkbspecial/kbtl/tlmodguide/tlapp_modelguide.cfm

[2] <https://de.mathworks.com/solutions/industrial-automation-machinery/standards/automotive-modeling-guidelines.html>

[3] <https://www.model-engineers.com/de/mxray.html>

Bildquellen:

- Abbildung 1: <https://de.mathworks.com/matlabcentral/mlc-downloads/downloads/submissions/28441/versions/22/screenshot.jpg>
- Abbildung 2: Eigene Abbildung

Development of a literature–software based on the snowflake method to support the planning, creation and structuring of novel content

Leonie Glanzer*, Kai Warendorf, Dominik Schoop

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

In the last decades our technology made a huge step forward, also in the book sells and distribution sector. Nowadays a lot of authors want to be independent from publishing companies to have more control over content, rights and marketing as stated in the European Self-publishing study 2016 [1]. This study also says that most of the books published are fiction, being due to complexity and length often an arduous genre to write. E-Books are getting more popular with time and are starting to take part in the competition. Today's authors want to have it easy and under control, including the way of writing books. Not many authors are using just pen and paper anymore to plan and write their stories, they are using programs to help and guide them through the process.

There are currently various applications, such as Bibisco [2], NovelFactory [3] and Papyrus Autor [4], helping authors write their stories on the market. While the outcome of this thesis, an application doing so too, has a lot in common with these, it has its special points, which includes being designed having the snowflake method as its model.

The snowflake method

There are many good approaches on writing a good story, but the most important part is detailed planning. The snowflake method by Ph.D. Randy Ingermanson[5] is a method for novel design, offering an easy to understand and simple approach on how to design and plan a novel. Ingermanson compares the writing process with the way an algorithm would create a simple snowflake, using fractal curves, known as the Koch snowflake [6]. The author starts with something simple and through the process he refines the shape in a structured way to get a complex shape still being rather easy to understand.

According to Ingermanson a good novel design is accomplished by writing the following documents in ten steps:

1. A one-sentence summary of the whole story, serving as a guideline
2. A one-paragraph summary describing the story, leading to splitting the story into acts, normally something like: Setup, one or two disasters and the finale
3. A one-page summary for each character, including its general information, motivation, goals, conflicts and learnings and the storyline
4. A one-paragraph summary of each act
5. A one-page summary for each character, describing its storyline
6. A four-page summary for each character, describing its storyline or rather expanding each paragraph of the one-page storyline into a whole page
7. A full-fledged character chart for each character, by expanding the four-page summary, describing everything there is to know about the character and defining the characters evolution
8. A one-sentence summary of each scene
9. A one-page summary of each scene
10. First draft

A crucial point is hereby the character driven storylines, telling the story from the perspective of a character [5].

Motive and goals

Not only hobby authors, also professionals and experts often have the problem that their stories get out of their hand and they lose the overview over what happens when and why and instead of progressing they remain static. Therefore an application will be implemented to assist in planning, creating and structuring novel content on an easy to understand level, without complex procedures and the demand for prior or advanced knowledge about writing novels. The application shall help guiding through the design process of a novel and at the same time give the author enough space to

*Diese Arbeit wurde durchgeführt an der Fakultät Informationstechnik

make own decisions and design choices. Not only creating novel content shall be offered, the author should be able to link contents to each other to picture even complex courses and plots. As an additional feature the application provides a built-in text editor, which makes it possible to gather statistics on writing behaviour and scene properties like letter count.

The snowflake method serves as a model for the application and defines which options are available and what the application can do in the end. The thesis main goal is to analyse the snowflake method and extract the resultant requirements for the application. From the requirements the applications architecture, models, logic and graphical interfaces will be derived and implemented. To improve the outcome some additional requirements are defined to enhance the user experience and to enable the user to increase a stories complexity.

Implementation and challenges

Derived from the snowflake method the user needs to manage a story, which can be spanned over multiple books. Therefore the structure needs to be as follows: A project consists of one or more books consisting of Acts, consisting of scenes, while these can be grouped into chapters. This gives the user a good overview and at the same time a good insight into what is happening on a coarser level. The latter is provided by filling out the summary property of the books, arcs and scenes, as described in the snowflake method.

The way the snowflake method approaches characters cannot be fully applied to the application, regarding the character driven storylines. The application contains instead general

storylines, being closely tied to characters, but not character dependent. This way also storylines concerning more characters are possible.

Besides managing characters, the application also includes locations and information, which is a format where the user can store important data on e.g. historical events, politics, tales and legends and other useful information, which can be linked to scenes, helping the user not to forget important background information.

The most important feature is the overview, which shows on changeable degrees of incivility the structure of the books and gives more or less detailed information on scenes.

Prospects

In the further course the implementation will be finished and put through its paces, to eliminated bugs and to optimize the application. The tests will be performed by several people, with different degrees of knowledge about the application and the writing process. As a final step a manual will be compiled and the application will be provided as an open source program on GitHub to potential users.

A plan is made to provide the application via web server to the users, but it is not an essential part of the thesis, which delays its realization until after the thesis is closed.

In the future this application could be transformed into a cloud application and run in a cloud to save resources. This way users could work whenever and wherever they want. Such a step would then afford more time and planning, which is way it is not part of the thesis.

-
- [1] The European Self-Publishing Study 2016, published by BoD – Books on Demand GmbH March 2016, executed by Prof. Dr. Vanessa Haselhoff, Dr. Joerg Pfuhl, Hochschule für angewandtes Management, Erding
 - [2] <http://www.bibisco.com/>
 - [3] <https://www.novel-software.com/>
 - [4] <https://www.papyrus.de/>
 - [5] Ph.D. Randy Ingermanson, The Snowflake Method For Designing A Novel, <https://www.advancedfictionwriting.com/articles/snowflake-method/>
 - [6] <http://ecademy.agnesscott.edu/~lriddle/ifs/ksnow/ksnow.htm>

Konzept und Implementierung für die Ermittlung und Anpassung von Maschinenstundensätzen in SAP

Nadine Haux*, Thomas Rodach, Catharina Kriegbaum-Kling

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Die Klingele Gruppe ist ein international tätiger Anbieter von Papier, Verpackungslösungen aus Wellpappe und Klebstoffen. Das 1920 gegründete Familienunternehmen mit Hauptsitz in Remshalden bei Stuttgart wird von Dr. Jan Klingele mittlerweile in dritter Generation geführt. In den Geschäftsfeldern Papier und Verpackung ist das Unternehmen inklusive Beteiligungen mit zwei Papierfabriken, zwölf Wellpappen- und acht Verarbeitungswerken in Europa, Afrika und Mittelamerika für seine Kunden präsent. Diese Werke beschäftigen insgesamt 2.400 Mitarbeiter und erzielten 2016 in Summe einen Umsatz in Höhe von 620 Mio. Euro. Das Produktspektrum umfasst Wellpappenrohpa-piere auf Basis von Altpapier sowie innovative und nachhaltige Verpackungslösungen (vorwiegend aus Wellpappe) für zahlreiche Branchen und Einsatzbereiche. Zudem ist Klingele Gründungsmitglied von Blue Box Partners, der größten europäischen Allianz für Verpackungslösungen [1].

Problemstellung

Aufgrund der steigenden Automatisierung wird auch im Unternehmen Klingele stets Personal durch Maschinen ersetzt. Kombiniert mit der Tatsache, dass in der Wellpappindustrie kostenintensive Maschinen zum Einsatz kommen, bekommt die Maschinenstundensatzrechnung eine immer größere Bedeutung. Denn dadurch vergrößert sich der Anteil maschinenabhängiger Gemeinkosten gegenüber den lohnabhängigen Gemeinkosten. Für eine genaue Kalkulation ist es erforderlich die Fertigungsgemeinkosten in maschinenabhängige und lohnabhängige Gemeinkosten aufzuteilen [2]. Die maschinenabhängigen Kosten sind in Abbildung 1 aufgelistet. Der Maschinenstundensatz ist ein wichtiger Bestandteil der Kundenauftragskalkulation, aber auch notwendig für die handelsrechtliche Bewertung. Der Unterschied beider Tarife liegt in den Kostenarten, die bei der Ermittlung, welche über eine Maschinenstundensatzrechnung erfolgt, berücksichtigt werden. Die Kostenarten, die in den handelsrechtlichen Tarif eingerechnet werden dürfen, sind im HGB[3] §255 definiert. Für den

kalkulatorischen Tarif gibt es lediglich eine Empfehlung, jedoch keine gesetzliche Vorgabe. Generell handelt es sich um Kosten, die für die Maschinenlaufzeit pro Stunde anfallen. Die Berechnung des Maschinenstundensatzes lässt sich ebenfalls aus Abbildung 1 entnehmen.

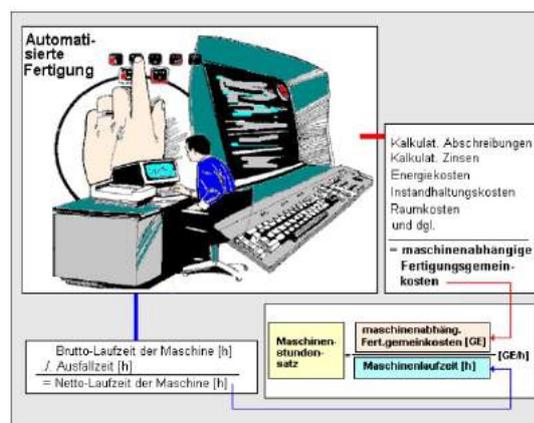


Abbildung 1: Darstellung der Ermittlung des Maschinenstundensatz

Das ERP System SAP bietet im Standard keine für Klingele ausreichende Möglichkeit für die Ermittlung der Maschinenstundensätze. Im Rahmen dieser Bachelorarbeit wird eine Anwendung für die vier deutschen Wellpappenwerke innerhalb des SAP Systems entwickelt. Die Ermittlung der Stundensätze wird an die Unternehmensanforderung angepasst.

Als weiterer Vorteil wird es möglich sein, über diese Anwendung auch neue Maschinenstundensätze zu pflegen ohne noch zusätzlich über die SAP Standard Anwendung gehen zu müssen, so dass alle Funktionalitäten über eine einzige Anwendung abgebildet werden.

Umsetzung

Für die Umsetzung einer Software Anwendung ist der Begriff „Softwaretechnik“ von großer Bedeutung, denn hinter der Erstellung einer Softwareanwendung steckt viel mehr als nur die Programmierung. Das Gebiet der Software-

*Diese Arbeit wurde durchgeführt bei der Firma Klingele Papierwerke GmbH & Co. KG, Remshalden

technik beschäftigt sich mit dem ganzen Paket der Herstellung und Entwicklung von Software und ist laut Helmut Balzert wie folgt definiert:

„Zielorientierte Bereitstellung und systematische Verwendung von Prinzipien, Methoden und Werkzeugen für die arbeitsteilige, ingenieurmäßige Entwicklung und Anwendung von umfangreichen Softwaresystemen.“ [4]

Die Bereiche der Softwaretechnik, welche in der Bachelorarbeit verwendet werden, sind in Abbildung 2 dargestellt:



Abbildung 2: Verwendete Bereiche der Softwaretechnik

Am Anfang einer jeden neuen Anwendung steht die Anforderung des Kunden. Hierfür wurde diese in einem kombinierten Lasten-/Pflichtenheft zwischen Entwickler und Kunde definiert. In der Definitionsphase wird zugleich ermittelt, woher die benötigten Daten genommen werden können bzw. wie sich diese errechnen lassen.

Als Vorgehensmodell wird das erweiterte Wasserfallmodell gewählt, da dieses für die Entwicklung am geeignetsten ist. Denn zum einen ist die Projektdauer auf einen kurzen Zeitraum beschränkt, nämlich die Dauer der Bachelorarbeit, und zum anderen erfolgt eine ständige Kommunikation mit dem Kunden, so dass man den einen oder anderen Schritt während der Entwicklung nochmals angehen und ggfs. abändern kann. Auch der Umfang der Anwendung ist überschaubar. Mit nur einem Entwickler sind mögliche Risiken gering zu bewerten. Ein Modell wie bspw. das Spiralmodell ist hier nicht angebracht.

Nach der Definition des vom Kunden gewünschten Produktes geht es in die Analyse des Ist- und Soll-Zustandes – was gibt es bereits und was wird benötigt? Während dieser Analyse wird zugleich aus Entwicklerseite auf die Softwarearchitektur eingegangen. Die Vorgaben der IT Unternehmensarchitektur wie System, Datenbank und Programmierrichtlinien werden hierbei berücksichtigt.

Im Anschluss beginnt die eigentliche Programmierung. In der hier geforderten Anwendung erfolgt diese innerhalb SAP in deren eigen entwickelten Programmiersprache ABAP[5]. Soweit erforderlich, werden objektorientierte Befehle von ABAP Objects verwendet. Da nahezu alle Daten, die für diese Anwendung benötigt werden in unterschiedlichen SAP Datenbanktabellen vorliegen, wird in der Programmierung viel mit Select-Statements und Aggregatfunktionen gearbeitet.

Herausfordernd ist dabei zum einen die Tatsache, dass die Klingele Papierwerke in den letzten Jahren viel außerhalb vom SAP Standard entwickelt hat und somit auch eigene Anwendertabellen berücksichtigt werden müssen und zum anderen die Erstellung einer ansprechenden grafischen Benutzeroberfläche für diese Anwendung, da SAP bekanntermaßen nicht sehr viel in diesem Bereich bietet.

Nachdem die Programmierung abgeschlossen ist, erfolgt die Validierung und Verifikation der erstellten Anwendung, bei der es sich zunächst um einen Prototyp handelt. Am Ende der Bachelorarbeit können fehlerfreie Systemtests am Prototyp durchgeführt werden.

Ausblick

Da die Ermittlung der Maschinenstundensätze bei Klingele zum Ende eines Kalenderjahres erfolgt, wird erst im Anschluss mit den Akzeptanztests begonnen und letztendlich folgt die produktive Implementierung der Anwendung zum Ende des dritten Quartals 2018.

[1] Klingele Papierwerke (2017): offizielles Firmenprofil

[2] Speth; Hahn; Waltermann (2016): Berufsfachliche Kompetenz Industrie, Steuerung und Kontrolle. Rinteln: Merkur Verlag

[3] Handelsgesetzbuch

[4] Balzert, H. (2000). Lehrbuch der Software-Technik. Bd.1. Software-Entwicklung. Heidelberg: Spektrum Akademischer Verlag

[5] Advanced Business Application Programming

Bildquellen:

- Abbildung 1: IWK. (2017). Maschinenstundensatz. Abgerufen am 20. November 2017 von <http://www.iwk-verlag.de/Demo/BwLex/html/M/Maschinenstundensatz.htm>
- Abbildung 2: Eigene Darstellung

Blockchain und Smart Contracts – Einführung, Analyse und Entwicklung eines Prototyps in der Automobilbranche

Fabian Högl*, Andreas Rößler, Reinhard Schmidt

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Einführung

Durch die Digitalisierung verändert sich unser tägliches Leben. Jeden Tag werden Start-ups gegründet, neue Technologien erforscht und veröffentlicht, branchenfremde Unternehmen drängen in für sich neue Märkte ein und vertreiben alt eingesessene Marktführer [1]. Vor allem die Automobilindustrie steht in dieser Zeit vor großen Herausforderungen. Connected Cars, Car Sharing, Industrie 4.0 und die Elektrifizierung der Fahrzeuge sind nur einige der Themen mit denen sich die Automobilhersteller befassen müssen. Auch große IT-Konzerne wie Google, Nvidia [2] und Intel sind mittlerweile neue Wettbewerber. In den letzten zwei bis drei Jahren wussten die Automobilhersteller nicht, wie mit diesen IT-Konzernen umgegangen werden soll. Mittlerweile lautet das Credo Partnerschaft statt Konkurrenz. Dies zeigen die jüngsten Kooperationen von Fiat-Chrysler und Google oder VW und Nvidia. Wie Friedrich Schiller sagte: „Wer nicht mit der Zeit geht, geht mit der Zeit“ [3]. Wie wichtig Digitalisierung in der Automobilbranche ist, zeigt eine Prognose des Marktforschungsunternehmens Gartner vom 26.01.2015, nach welcher bis 2020 rund 250 Millionen Connected Cars auf den weltweiten Straßen unterwegs sein werden [4]. Dies kommt auch zustande, da ab März 2018 der automatische Notruf eCall in der EU für jedes neu produzierte Fahrzeug Pflicht wird und somit jedes neue Fahrzeug ein Connected Car ist [5].

Die Automobilhersteller sollten deshalb einen Weg finden, wie sie von einem Automobilhersteller zu einem Anbieter für Mobilität werden, damit sie auch in Zukunft wettbewerbsfähig bleiben. Den jungen Menschen geht es nicht mehr um Besitz, sondern um den Nutzen [6]. Deshalb ist es sehr wichtig, dass sich die Automobilhersteller mit den neuen Technologien und Gegebenheiten auseinandersetzen.

Eine dieser neuen Technologien von der behauptet wird dass sie die Welt verändern kann, ist die Blockchain Technologie [7]. In der Regel wird sie mit der heutigen Kryptowährung

Bitcoin in Verbindung gebracht. Wie in Abbildung 1 zu sehen ist, gibt es aber weit mehr als nur den Bereich Kryptowährung.



Abbildung 1: Blockchain Ecosystem

Mit dem Aufstieg von Ethereum mit seiner Turing-vollständigen Scriptsprache und der Fähigkeit für Entwickler, den Status in jeden Block einzuschließen, wurde der Weg für die Entwicklung sogenannter Smart Contracts ebnet. Dies hat zu einem Zustrom von Teams geführt, die dezentralisierte Projekte aufbauen, um die wichtigste Eigenschaft von Blockchains zu nutzen – die Fähigkeit eine gemeinsame Wahrheit zu erreichen, die jeder ohne Intermediäre oder einer zentralen Autorität zustimmt [8].

Funktionsweise der Blockchain

Das folgende Beispiel zeigt die Funktionsweise einer Blockchain anhand einer Transaktion:

- Ein User fordert eine Transaktion an. Diese Transaktion wird via Broadcast in ein Peer-to-Peer Netzwerk bestehend aus Computern, auch bekannt als „nodes“ gesendet.
- Das Netzwerk aus „nodes“ validiert die Transaktion und den Status des Users mit bekannten Algorithmen. Eine verifizierte Transaktion kann Details über Kryptowährungen, Verträge, Dokumentationen oder andere Informationen beinhalten.

*Diese Arbeit wurde durchgeführt an der Fakultät Informationstechnik

- Einmal verifiziert, wird die Transaktion mit anderen Transaktionen kombiniert um einen neuen Block mit Daten für die „ledger“ zu kreieren. Die „ledger“ ist ein öffentliches, dezentral geführtes Konto-buch. Der neue Block wird dann an die bestehende Blockchain hinzugefügt, in einer Weise, die dauerhaft und unveränderlich ist.
- Die Transaktion ist abgeschlossen.

Aufbau einer Blockchain

Wie in Abbildung 2 zu sehen ist, besteht eine Blockchain aus einem Genesis Block, mehreren Blocks und den darin enthaltenen Informationen wie Transaktionen. Dies unterscheidet sich von Blockchain zu Blockchain und dessen Technologie.

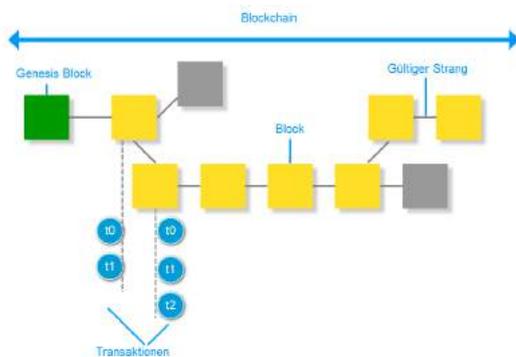


Abbildung 2: Blockchain Komponenten

Smart Contracts

Der Begriff „smart contract“ wurde von Nick Szabo im Jahr 1994 geprägt. Szabos Idee war es, eine rechtliche Vereinbarung in einem Programm zu kodifizieren und einen Computer die Bedingungen ausführen zu lassen, anstatt dass der Mensch sie interpretieren und danach handeln muss. In Verbindung mit der

Blockchain Technologie wird dies nun möglich, d.h. Smart Contracts werden nicht an einer zentralen Stelle von einer Instanz als Dienst bereitgestellt, sondern sie werden von vielen unterschiedlichen Teilnehmern im Netzwerk ausgeführt [9].

DAPP

Eine DAPP ist eine dezentralisierte Applikation, welche dezentralen Code benutzt. Somit hat eine DAPP ein dezentrales Backend und ein zentralisiertes Frontend. Das Backend einer DAPP wird mindestens teilweise Code und Speicher auf der Blockchain haben. Das Frontend wie das User Interface, Grafiken oder Webseiten werden normalerweise weiterhin in traditioneller Weise gehandhabt, mit zentralisierten Servern oder Cloud Services. In Abbildung 3 werden die Unterschiede verdeutlicht. Während bei einer traditionellen Applikation alle Daten auf einem Server liegen, ist in einer DAPP die „Data logic“ zum Beispiel in der Blockchain [10].

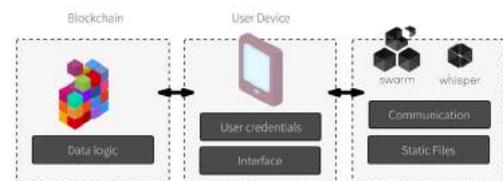


Abbildung 3: dezentrale Struktur einer DAPP

Ausblick

Ziel ist es, einen funktionsfähigen Prototyp zu entwickeln, der auf einer zuvor evaluierten Blockchain Technologie beruht. Dieser soll im Kontext Connected Car eingesetzt werden.

- [1] www.huffingtonpost.de/2014/04/29/nokia-gescheitert_n_5230806.html, (Zugriff: 11.10.2017)
- [2] www.wiwo.de/unternehmen/auto/autokonzerne-und-tech-giganten-wer-macht-es-mit-wem-und-warum/19234002.html, (Zugriff: 10.10.2017)
- [3] <https://www.evoila.de/unternehmen/philosophie/>, (Zugriff: 18.10.2017)
- [4] www.gartner.com/newsroom/id/2970017, (Zugriff: 09.10.2017)
- [5] <http://www.europarl.europa.eu/news/de/press-room/20150424IPR45714/ecall-automatisches-notrufsystem-in-allen-neuen-automodellen-ab-fruhling-2018>, (Zugriff: 09.10.2017)
- [6] <https://www.welt.de/wirtschaft/article132632751/Warum-der-jungen-Generation-das-Auto-egal-ist.html>, (Zugriff: 08.10.2017)
- [7] <https://www.mckinsey.com/industries/high-tech/our-insights/how-blockchains-could-change-the-world>, (Zugriff: 10.10.2017)
- [8] https://medium.com/@josh_nussbaum/blockchain-project-ecosystem-8940ababaf27, (Zugriff: 27.10.2017)
- [9] Diedrich, Henning (2016): ethereum. Wildfire Publishing
- [10] Diedrich, Henning (2016): ethereum. Wildfire Publishing

Bildquellen:

- Abbildung 1: https://medium.com/@josh_nussbaum/blockchain-project-ecosystem-8940ababaf27
- Abbildung 2: Eigene Darstellung
- Abbildung 3: <https://blog.ethereum.org/2016/07/12/build-server-less-applications-mist/>

Sichere und zuverlässige Aktualisierung linuxbasierter Firmware auf mobilen Telematiksystemen

Marcel Hubert*, Werner Zimmermann, Reiner Marchthaler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Firmware-Aktualisierungen auf vernetzten Embedded-Linux-Systemen sind sowohl wichtige als auch kritische Vorgänge. Sie umfassen Bugfixes und neue Funktionalitäten, vor allem aber auch Patches gegen Sicherheitslücken in Kernel, Treibern und Programmbibliotheken.

Updates können jedoch selbst zum Risiko für die Sicherheit und Zuverlässigkeit eines Systems werden. Ungesichert können sie Angreifern ermöglichen, Schadsoftware auf dem System zu platzieren. Scheitert ein Update, etwa durch unerwartetes Abschalten während des Installationsvorgangs, kann das System in einem Zustand verbleiben, in dem es nicht wieder hochfahren kann. Je nach Einsatzort muss es dann aufwendig ausgebaut und zur Wiederherstellung eingeschickt werden.

Ziel dieser Bachelorarbeit ist es, einen sicheren und zuverlässigen Aktualisierungsvorgang für die Embedded-Linux-Firmware auf den Telematikeinheiten der Trendfire Technologies GmbH zu konzipieren und implementieren.

Umfeld und Anforderungen

Die Trendfire Technologies GmbH ist Entwickler einer LKW-Telematiklösung für Flottenmanagement, Fahrzeugortung und Fahrerbewertung. Die Datenbasis für die Telematiklösung liefern die trendbee-Telematikeinheiten (siehe Abbildung 1). Sie werden in LKW verbaut und sammeln Positionsdaten über GPS, Fahrzeugdaten vom Fahrzeugbus und die Daten des digitalen Tachographen zu Arbeits-, Lenk- und Ruhezeiten des Fahrers. Diese Daten werden über Mobilfunk zur Speicherung und Verarbeitung auf einen Server hochgeladen.



Abbildung 1: Telematikeinheit trendbeeTRUCK v4

Der Einsatz der Telematikeinheiten in Fahrzeugen in ganz Deutschland und Europa stellt besondere Voraussetzungen an die Zuverlässigkeit der Firmware-Aktualisierung. Ausgefallene Einheiten erzeugen Lücken in der Datenaufzeichnung und sind nur zu bestimmten Zeiten für Wartungsarbeiten erreichbar.

Sind die Fahrzeuge in Bewegung, kommen Schwankungen der Datenübertragungsrates und bei schlecht ausgebauten Mobilfunknetzen auch Verbindungsabbrüche vor. Eine stabile Mobilfunkverbindung kann also nicht gewährleistet werden. Mögliche Beschädigungen der Daten auf dem Übertragungsweg müssen von der Telematikeinheit erkannt und eine Installation beschädigter Dateien verhindert werden.

Schutz soll auch vor Manipulation der Daten gewährleistet sein. Dateien, deren Herkunft und Vertrauenswürdigkeit nicht festgestellt werden können, sollen nicht zur Installation angenommen werden.

Lösungsansätze

Zertifikatbasierte Authentizitäts- und Integritätsprüfung:

Um die Herkunft und Integrität eines Updates sicherzustellen, wird vor dem Download mithilfe des privaten Schlüssels eines RSA-Schlüsselpaares und dem Hash einer Datei deren digitale Signatur erstellt. Der Empfänger lädt die Datei zusammen mit der Signatur herunter. Er erhält mit dem öffentlichen Schlüssel des Schlüsselpaares aus der Signatur den Hash der gesendeten Datei und vergleicht ihn mit dem Hash der empfangenen Datei. Wird die Datei auf dem Übertragungsweg manipuliert oder beschädigt, stimmen die Hashes nicht überein. Die Verifizierung scheitert und die Datei wird abgelehnt. Da der öffentliche Schlüssel nur Signaturen seines privaten Gegenstücks öffnen kann, werden fremdsignierte Dateien ebenso abgelehnt.

Damit dieses Verfahren sicher bleibt, wird vorausgesetzt, dass der private Schlüssel auf Senderseite vor Fremdzugriff geschützt ist. Sonst können Dateien fremder Herkunft mit diesem Schlüssel signiert werden. Auch der öf-

*Diese Arbeit wurde durchgeführt bei der Firma Trendfire Technologies GmbH, Böblingen

fentliche Schlüssel auf Empfängerseite muss gegen Zugriff geschützt werden. Tauscht ein Angreifer den öffentlichen Schlüssel gegen den eigenen, kann er mit seinem privaten Schlüssel Daten signieren und von der Einheit verifizieren lassen. [1] (vgl. S. 253)

Durch ein Zertifikat kann das Schlüssel-paar an die Identität des Updateausstellers gebunden werden. Zur Signaturprüfung erfolgt dann eine Prüfung der Identität des Unterzeichners anhand des Signier-Zertifikats. Damit dessen Identität bestätigt werden kann, muss das Signier-Zertifikat von einem Root-Zertifikat ausgestellt sein, das der Empfänger als vertrauenswürdig erachtet. [1] (vgl. S. 299)

Paketerstellung und Download:

Im schlechtesten Fall besteht das Update aus großen Dateien, die mit langsamer Mobilfunkverbindung in einem fahrenden Fahrzeug heruntergeladen werden. Mit zunehmender Dauer eines Downloads erhöht sich die Wahrscheinlichkeit, dass er durch eine instabile Mobilfunkverbindung unterbrochen wird.

Um das Datenvolumen zu verringern, empfiehlt sich daher bei der Paketerstellung eine Komprimierung der Daten. Zusätzlich werden größere Pakete vor der Bereitstellung zum Download in kleinere Teilpakete aufgeteilt und Stück für Stück von der Telematikeinheit heruntergeladen. Eine Unterbrechung des Downloads oder beschädigte Pakete wirken sich damit nur auf einen kleinen Teil der Gesamtdatenmenge aus. Die beschädigten Teilpakete kann die Einheit erneut zum Download anfordern. Die bis dahin vollständig geladenen Dateien bleiben erhalten.

Die heruntergeladenen Teilpakete werden danach anhand ihrer Signatur überprüft und zusammengesetzt. Das zusammengesetzte Paket wird erneut geprüft und anschließend installiert (siehe Abbildung 2).

Redundantes Speicherlayout:

Die kritischste Phase der Firmware-Aktualisierung ist die Installation. Dabei wird die vorher heruntergeladene Firmware auf den Flash-Speicher geschrieben. Verliert die Einheit in dieser Zeit ihre Spannungsversorgung, wird der Schreibvorgang unvollständig abgebrochen. Die Installation ist gescheitert und das System wahrscheinlich nicht mehr bootfähig.

Auch unzureichend getestete Software, deren Fehler sich erst nach dem Aufspielen auf das Gerät zeigt, kann das System stilllegen.

Um die Auswirkung solcher Vorfälle aufzufangen, werden im Flash-Speicher zwei Speichersektionen für die Firmware reserviert. Der Bootloader bootet beim Systemstart von der aktiv gesetzten Sektion. Bei der Firmware-Aktualisierung wird die neue Firmware auf die inaktive Speichersektion geschrieben. Bei einem anschließenden Neustart versucht der Bootloader die neu aufgespielte Firmware zu booten. Verlieft der Bootvorgang erfolgreich, wird deren Speichersektion aktiv gesetzt. Andernfalls fällt der Bootloader auf die vorige Sektion zurück und bootet die zuletzt funktionstüchtige Firmware.

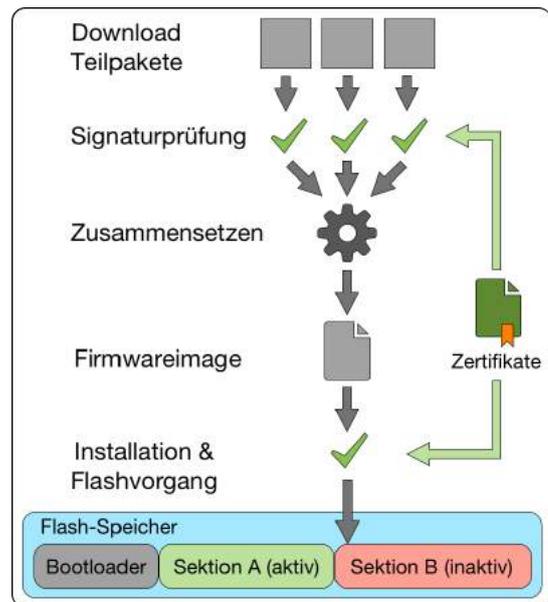


Abbildung 2: Verarbeitung eines Updates auf der Telematikeinheit

Verified Boot:

Eine weitere Sicherheitsmaßnahme ist die Prüfung der Signatur des Firmwareimages vor dem Booten des Betriebssystems. Die Signaturprüfung verhindert, dass Trendfire-fremde Firmware auf die Telematikeinheiten aufgespielt und betrieben werden kann. Unsignierte, fremdsignierte oder veränderte Firmwareimages können nicht mehr gebootet werden.

Beim Erstellen der Firmwareimages müssen diese mit einem privaten RSA-Schlüssel signiert werden. Auf dem Gerät muss der passende öffentliche Schlüssel für den Bootloader lesbar, aber gegen Austausch gesichert, gespeichert sein.

[1] BUCHMANN, Johannes, 2016: Einführung in die Kryptographie [online]. 6. Auflage. Berlin, Heidelberg: Springer Spektrum [Zugriff am: 20.11.2017]. PDF e-Book. ISBN 978-3-642-39775-2. Verfügbar unter: DOI: 10.1007/978-3-642-39775-2

Bildquellen:

- Abbildung 1: Trendfire Technologies GmbH
- Abbildung 2: Marcel Hubert

Die Blockchain-Technologie auf Basis dezentraler Strukturen. Von Bitcoin zur massentauglichen Technologie für Industrie und Wirtschaft.

Rukiye Kaya*, Catharina Kriegbaum-Kling, Thomas Rodach

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Einführung

Die technologische Grundlage für die Blockchain-Technologie wurde im Jahr 2008 mit der Entwicklung der digitalen Währung Bitcoin geschaffen. In diesem Jahr veröffentlichte eine Entwicklergruppe unter dem Pseudonym „Satoshi Nakamoto“ eine Fallstudie (ein so genanntes „White Paper“), in welcher ein Konzept einer dezentral verwalteten digitalen Währung vorgestellt wurde, das vollständig befreit vom bisher bekannten Bankensystem funktionieren kann. Die Idee dahinter ist der Austausch einer Währung zwischen zwei Parteien, anhand eines Systems das keine unabhängige dritte Instanz zur Verifikation der Transaktion benötigt. Im Jahr 2009 wurde dieses Konzept von Entwicklern umgesetzt und die digitale Währung Bitcoin zum ersten Mal auf einem digitalen Handelsplatz zur Verfügung gestellt, auf der diese gegen eine reale Währung umgetauscht werden konnte. Aufgrund des neuartigen Konzepts, der starken Transaktionsgeschwindigkeit und der starken Vereinfachung der Transaktionen gewann Bitcoin schnell an internationaler Beachtung und Wert.

Revolutionär an der Bitcoin Währung ist die technologische Basis, die Blockchain. Als Liste von Datensätzen gilt die Blockchain als digitales Register und ähnelt einem Datenbanksystem. Losgelöst von einer zentralen Instanz ermöglicht sie die Transaktionen zwischen den Teilnehmern ohne die Notwendigkeit einer zentralen Instanz. [1][2]

Die Blockchain-Technologie

Im Gegensatz zu einem zentralisierten System, bei der alle Informationen über Transaktionen auf einer zentralen Datenbank zu finden sind, ist Blockchain eine dezentrale Registerstruktur, die Transaktionen für jeden Teilnehmer transparent dokumentiert. Dadurch kann jeder Teilnehmer innerhalb des Netzwerks zu jeder Zeit alle Informationen einer jeden Transaktion abrufen. [2]

Dieses digitale Register besteht aus einer Kette von Blöcken, in der die Transaktionen nacheinander miteinander verknüpft werden. Mehrere Transaktionen werden dabei zu einem Block zusammengefasst, daraufhin auf ihre

Gültigkeit überprüft und innerhalb eines Algorithmus, der als „Proof-of-Work“ bekannt ist und die massenhafte Durchführung von Transaktionen einzelner Teilnehmer verhindert, an die bestehende Kette angefügt.

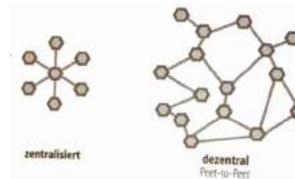


Abbildung 1: Links ein zentrales, rechts ein dezentrales Netzwerk-Modell

Zentrale Eigenschaften der Blockchain Technologie sind dabei die Dezentralität, Unveränderlichkeit und Transparenz. Mithilfe von Hashing und dem Proof-of-Work Algorithmus wird die nötige Transaktionssicherheit gewährleistet.

Dezentralität

Im Unterschied zur klassischen Transaktionsabwicklung basiert Blockchain auf vollständig dezentralisierten Strukturen. Dadurch können einzelne Teilnehmer einer Transaktion direkt miteinander kommunizieren und verhindern die Notwendigkeit einer zentralen, vertrauenswürdigen Instanz. Durch den Einsatz von kryptografischen Signaturen – also der Verschlüsselung von Informationen zur Verhinderung von Manipulation und unbefugtem Zugriff auf Informationen – sowie dem Proof-of-Work Algorithmus wird die notwendige Sicherheit und Wahrhaftigkeit der Transaktion garantiert.

Unveränderlichkeit

Sobald eine Information innerhalb der Blockchain Kette des Netzwerks abgelegt ist, können diese Informationen nicht mehr entfernt werden. Diese Unveränderlichkeit sichert die Dauerhaftigkeit der Transaktionen und eine unbefugte Rückabwicklung – welche speziell für finanzielle Transaktionen wichtig ist – wird verhindert. Zur Rückabwicklung müssten alle Teilnehmer des Netzwerks gemeinsam agieren und die Rückabwicklung bestätigen.

*Diese Arbeit wurde durchgeführt an der Fakultät Informationstechnik

Transparenz

Die Eigenschaft der Dezentralität ermöglicht neue Formen der Transparenz jeder Transaktion. Anstatt in einer zentralen Datenbank, auf die nur eine fremde, unabhängige Instanz wie eine Bank Zugriff hat, bekommt jeder Teilnehmer des Netzwerks Zugriff auf die gesamten Transaktionen. Dabei können die Transaktionen bis hin zum ersten generierten Block der Blockchain, dem so genannten Genesis Block, zurückverfolgt werden.

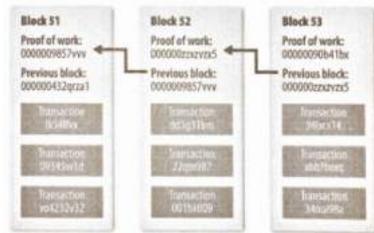


Abbildung 2: Mathematisch aufeinander aufbauende Blöcke

Besonderer Wert muss aufgrund der neuen, dezentralen Struktur, auf Konsistenz – also Beständigkeit – und Integrität der Daten gelegt werden. Um dies zu gewährleisten werden Konsens-Mechanismen wie das Hashing als kryptografische Signatur der gespeicherten Daten und den Proof-of-Work Algorithmus implementiert. [3]

Hashing

Ein Hash ist das Resultat einer komplexen mathematischen Aufgabe. Damit wird eine bestimmte Zahl – als arithmetisches Abbild der Information – gesucht, die festgelegten Anforderungen entspricht und ein festgelegtes Format aufweist. Mit dem Hashing wird sichergestellt, dass Blöcke nur am Ende einer Blockchain hinzugefügt werden können. Um dies zu gewährleisten, wird in jedem Block eine kryptografische Signatur des vorherigen Blocks gespeichert. Zur Bestimmung der Signatur eines Blocks, muss auch die Signatur des Vorgängerblocks beachtet werden, dadurch entsteht eine Kette aus Blöcken bei der die Veränderung eines Blocks sich auf die Signaturen aller auf ihn folgender Blöcke auswirkt. Dadurch kann eine Manipulation der Blockchain effektiv unterbunden werden. [2]

Proof-of-Work

Dieser Algorithmus ermöglicht eine gesteigerte Sicherheit und Funktionsfähigkeit der Blockchain. Innerhalb dieses Algorithmus muss der Teilnehmer komplexe mathematische Aufgaben lösen. Dieser Prozess stellt sicher, dass Transaktionen anderer Teilnehmer bestätigt werden und somit genug Aufwand in die Berechnung und Sicherung der Transaktionen investiert werden. [3]

Neue Einsatzgebiete

Blockchain war in seinen Anfängen ausschließlich als Technologische Basis der digitalen Währung Bitcoin gedacht. Mittlerweile entdecken mehr und mehr wirtschaftliche und industrielle Sektoren die Vorteile einer dezentralen Transaktionsabwicklung. Im Finanzsektor können mithilfe dieser Technologie Finanztransaktionen vereinfacht und beschleunigt werden. Ebenfalls kann die Technologie in weitere Bereiche des Finanzsektors ausgeweitet werden. Dies umfasst dabei jegliche Art von Transaktionen, bei der die Transaktionsgegenstände digital dargestellt werden können. So können sogenannte „Smart Contracts“, also automatisierte Verträge, die Aktionsbedingung in Kraft treten und keine menschliche Überwachung benötigen, in einer Blockchain abgelegt und somit zwischen zwei oder mehreren Parteien sicher und schnell abgeschlossen werden.

Auch im Smart Home Bereich stecken weitere Einsatzmöglichkeiten. Im Haus verbaute Sensoren können mögliche Schäden im Haus erkennen und das System könnte vollautomatisiert einen Handwerker beauftragen und – dank Blockchain Technologie – digital bezahlen. Der Handwerker würde sich dann um die Behebung des entstandenen Schadens kümmern und der Eigentümer hat keinen weiteren Aufwand. Durch diese weitreichenden Implementierungsmöglichkeiten bis hin in das alltägliche Leben von Endverbrauchern entwickelt sich ein hohes disruptives Potential für bestehende Geschäftsmodelle. [4][5][6]

Zielsetzung der Arbeit

Diese Arbeit soll die unterschiedlichen Einsatzmöglichkeiten der Blockchain Technologie in der Wirtschaft und Industrie durchleuchten und Antwort darüber geben, welche Vorteile und Risiken der Einsatz mit sich bringt, was sich mit dem Einsatz ändert und wie der Einsatz von Blockchain realisiert wird.

[1] Wikimedia Foundation Inc: <https://de.wikipedia.org/wiki/Blockchain> (Abgerufen 22.11.2017)

[2] Platzer, Joerg (2014): Bitcoin kurz & gut. Banking ohne Banken. Seite 20 Köln: O'Reilly Verlag

[3] Frankfurt School of Finance & Management gemeinnützige GmbH:

<http://blog.frankfurt-school.de/blockchain-technologien-konsens-mechanismen/> (Abgerufen 22.11.2017)

[4] Vogel IT-Medien GmbH: <https://www.dev-insider.de/was-ist-ein-smart-contract-a-585679/> (22.11.2017)

[5] BTC-Echo UG: <https://www.btc-echo.de/tutorial/was-ist-die-blockchain/> (Abgerufen 22.11.2017)

[6] CS2 AG: <http://www.zweiblog.com/2016/06/7-aufregende-anwendungen-der-blockchain-technologie> (Abgerufen 22.11.2017)

Bildquellen:

- Abbildung 1, 2: Joerg Platzer: Bitcoin kurz & gut. Banking ohne Banken. Köln: O'Reilly

Rule Based Methods for Object Detection

Tanja Kempfert*, Kai Warendorf, Dominik Schoop

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Motivation

Machines can assist us in many areas. Cameras automatically focus on faces. Image databases automatically tag photographs to make them easier to find again. Robots are able to work in areas inaccessible to humans. Driving assistance systems help to improve safety on roads.

These and many other applications require computers to interpret visual data. Surprising as it may seem to a human, this turns out to be quite a hard task for a machine.

Much research has been done on object detection (i.e. the determination of both presence and location of an object). For many applications deep learning approaches have turned out to be most successful [1]. Yet these approaches have two major drawbacks. First, they require large amounts of annotated training data. Second, although some methods for better interpretability exist [2], the behaviour of deep learning approaches is often hard to interpret – and therefore hard to predict.

For specialised applications and when training data is hard to obtain, rule based algorithms can therefore be preferable. Such a scenario will be covered in the author's bachelor thesis.

This article intends to give an overview over existing rule based approaches with a focus on current works applicable to 3d data.

Classification of approaches

In the following, the approaches are generalised into three classes: model based, feature geometry based and segmentation based approaches. Model based approaches take the appearance of the whole object into account. Feature geometry based approaches identify important parts and set them in geometric relation with each other. Segmentation based approaches subdivide the data into regions based on shared properties.

Model based approaches

Model based object detection uses an explicit model of an object that represents knowledge of the object's appearance [3]. The transformation necessary to align the model with the data is used to retrieve object position and orientation. The process of alignment is easier the more accurately a model describes a specific object. Yet the model also needs to describe all other objects of the same type. It is not always possible to find a single model that can be used both for robust and accurate detection.

Models can differ by properties such as

- Dimensionality: is the data two- or three-dimensional? Does it include colour data?
- Degrees of freedom: can the model be scaled, rotated, translated or deformed?
- Data representation: is the data represented as unordered points, a graph, voxels, a function or local descriptors?

A further challenge with model based detection is the dependency of object appearance on the point of view. This is especially true for two-dimensional data, but also applies in three dimensions because of self-occlusions. Most trivially it is impossible to see the back of an opaque object.

In general, a model based algorithm needs to perform the following tasks:

1. Measure the similarity between model and corresponding data for a given transformation
2. Find the transformation that maximises the similarity

One popular method for aligning model and data is the use of registration algorithms. The Iterative Closest Point (ICP) algorithm [4] is perhaps the best example for such a method. It is commonly used to finely align an already roughly aligned model. The algorithm first establishes correspondences between points of

*Diese Arbeit wurde durchgeführt bei der Firma Philips Medizin Systeme Böblingen GmbH, Böblingen

the model and the data by selecting the nearest neighbour. After rejecting outliers, the mean squared distance is taken as a dissimilarity measure. An example for the alignment using ICP is shown in figure 1. Also widely in use are RANSAC variants. RANSAC (Random Sampling Consensus) [5] works by repeatedly sampling as many data points as necessary to support a possible transformation (e.g. two points when trying to fit a line). The points are used to compute the transformation parameters for that hypothesis. Each hypothesis then scored by its similarity to the corresponding data. The transformation with the highest similarity score is chosen as result.

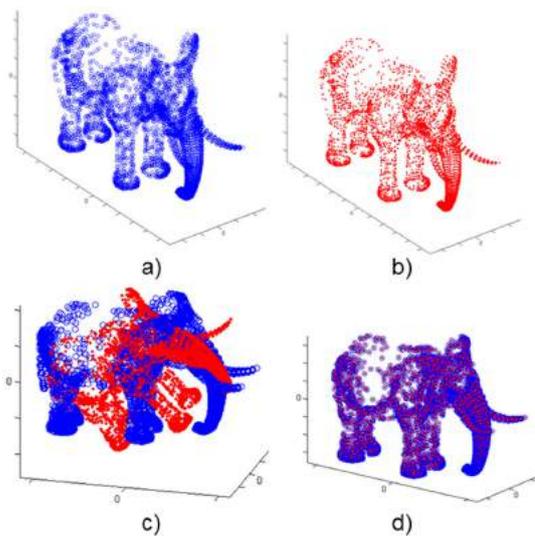


Figure 1: Example of the ICP algorithm. The model (a) is aligned to the data (b), starting from the initial alignment shown in (c). The alignment after application of ICP is shown in (d).

Feature geometry based approaches

Feature geometry based algorithms use important features and the geometric relationship between them to detect objects.

Features can be corners, edges, circles and other relatively simple structures in the data. In order to provide good result, features need to be

- invariant under geometric transformations that do not fundamentally change an object, such as translation and rotation, and
- robust to noise.

García-Garrido et al. [6] use a modified version of the Hough transform to detect outlines of traffic signs in an edge image (see figure 2). The Hough transformation in its original form maps all points on a line to one point in Hough space, enabling the detection of lines by finding maxima in Hough space. It

can be generalised for the detection of other parametrisable forms.

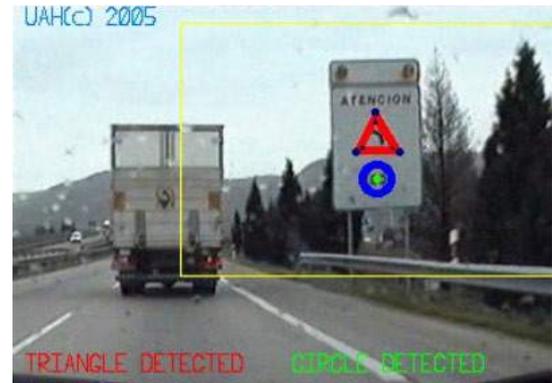


Figure 2: Traffic sign detection using a Hough based algorithm.

Early face detection algorithms also relied on geometric relationships between features like mouth, nose and eyebrows [7].

There is some overlap with model based detection since one way to obtain the geometric relations is to extract them from a model.

Segmentation based approaches

Segmentation based detection methods use local or global properties to divide the data into separate regions. The clusters can be used as object hypotheses. The similarity measures that have been described above in the context of model based detection can be used to assess whether a cluster belongs to an object category.

There are three general approaches to segmentation [8, ch. 4]:

- Thresholding
- Edge based segmentation
- Region based segmentation

Thresholding methods use global attributes such as intensity or colour to determine the category a data point belongs to. A category is defined by a lower and an upper threshold for the attribute value. The threshold values can be fixed or set depending on the values occurring in the data.

In edge based segmentation an edge filter is used to classify data points as either edge or non-edge points. All non-edge points which are not separated by an edge are assigned to the same region.

Region based segmentation groups together neighbouring data points with similar attribute values. This can be done by splitting (initially all points are in one region) or merging (initially each point forms its own region). A combined split-and-merge approach is possible by starting with a number of regions which

are either split or merged according to the homogeneity inside a region. Seeded methods start with a number of points from where the regions are grown. They can produce different results depending on the choice of seeds which can be a problem if the seeds are not carefully chosen.

For 3D data, often segmentation based only on Euclidean distances [9] is used. Another popular approach is the use of estimated surface normal vectors for extracting smooth [10] or convex [11] (see figure 3) regions.

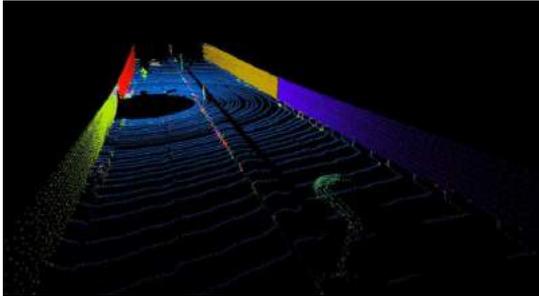


Figure 3: Segmentation of a street scene using a local convexity criterion.

Conclusion

Although machine learning has proved very successful for image processing tasks, rule based algorithms can still be a viable alternative. Their strengths lie in applications where little training data is available, where the detection task can be described by an easy set of rules or where predictability is of high value.

For a specific problem, the choice of algorithm has to be considered carefully so as to best take into account all available knowledge about the object type.

Model based approaches are useful if a specific object or objects with little intra-class variance need to be detected. Larger variances lead to more parameters to consider, which has a negative effect on both runtime and quality of results.

Feature geometry based approaches should be considered where the object is determined by a few well-known relations between prominent features.

With segmentation based approaches there is always the possibility of over- or under-segmentation. Small changes in a scene can require changes in parameters that lead to a successful segmentation. They can be applied if the scene is well controlled or if properties of object segments are more important than detecting whole objects.

-
- [1] Yann LeCun, Yoshua Bengio and Geoffrey Hinton. "Deep learning." *Nature* 521.7553 (2015): 436–444.
 - [2] Grégoire Montavon, Wojciech Samek and Klaus-Robert Müller. "Methods for interpreting and understanding deep neural networks." *Digital Signal Processing* (2017).
 - [3] Arthur R. Pope. "Model-based object recognition. A Survey of Recent Research." *Technical Report* (1994).
 - [4] Paul J. Besl and Neil D. McKay. "A method for registration of 3-D shapes." *IEEE Transactions on pattern analysis and machine intelligence* 14.2 (1992): 239–256.
 - [5] Martin A. Fischler and Robert C. Bolles. "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography." *Communications of the ACM* 24.6 (1981): 381–395.
 - [6] Miguel García-Garrido, Miguel Sotelo and Ernesto Martín-Gorostiza. "Fast road sign detection using Hough transform for assisted driving of road vehicles." *Computer Aided Systems Theory-EUROCAST 2005* (2005): 543–548.
 - [7] Constantine Kotropoulos and Ioannis Pitas. "Rule-based face detection in frontal views." *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP-97)*. Vol. 4. IEEE, 1997.
 - [8] Chris A. Glasbey and Graham W. Horgan. *Image analysis for the biological sciences*. Vol. 1. Chichester: Wiley, 1995.
 - [9] Radu Bogdan Rusu. "Semantic 3d object maps for everyday manipulation in human living environments." *KI-Künstliche Intelligenz* 24.4 (2010): 345–348.
 - [10] Tahir Rabbani, Frank Van Den Heuvel and George Vosselmann. "Segmentation of point clouds using smoothness constraint." *International archives of photogrammetry, remote sensing and spatial information sciences* 36.5 (2006): 248–253.
 - [11] Frank Moosmann, Oliver Pink and Christoph Stiller. "Segmentation of 3D lidar data in non-flat urban environments using a local convexity criterion." *2009 IEEE Intelligent Vehicles Symposium*. IEEE, 2009.

Image sources:

- Figure 1: Higinio Mora, Jerónimo M. Mora-Pascual, Alberto García-García and Pablo Martínez-González. "Computational Analysis of Distance Operators for the Iterative Closest Point Algorithm." *PLoS one* 11.10 (2016): e0164694.
- Figure 2: [6]
- Figure 3: [11]

Anwendbarkeit ausgewählter Lean Six Sigma Tools auf Verwaltungsprozesse

Dinesh Kishore ^{*}, Thomas Rodach, Catharina Kriegbaum-Kling

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Die Globalisierung und die damit verbundene Konkurrenz, sowie die Ressourcenknappheit und die kürzeren Lebenszyklen der Produkte sind Herausforderungen für die Unternehmen. Deswegen müssen Maßnahmen umgesetzt werden, damit Unternehmen immer effizienter und kostensparender operieren [1]. Die Wichtigkeit von Kernprozessen, sowie deren Optimierung und Verbesserung haben die Unternehmen schon erkannt. Zusätzlich zu den Kernprozessen gibt es unterstützende Prozesse wie zum Beispiel Verwaltungsprozesse, welche in allen Unternehmen vorhanden sind. Sie sind notwendig, damit die Mitarbeiter ihre jeweilige Arbeit verrichten können und damit alle Prozesse reibungsfrei laufen. Gerade in administrativen Bereichen gibt es laut einer Studie von Fraunhofer viel Verbesserungspotenzial. Viele Verbesserungsmöglichkeiten in den administrativen Bereichen hängen direkt mit der Produktion zusammen. Verzögerte Freigaben und wartende Aufträge kommen aus der

Verwaltung und wirken sich direkt auf die Produktion aus. Laut der Studie gibt es in der Verwaltung 30% Verschwendung [2]. In Rahmen der Bachelorthesis werden ausgewählte Tools von Lean Six Sigma auf den Onboarding Prozess von Capgemini angewendet. Das eigentliche Ziel ist es ist, mit den gewonnenen Erkenntnissen darüber Aufschluss zugeben, ob die ausgewählten Tools auf Verwaltungsprozesse anwendbar sind. Lean Six Sigma ist die Kombination der Methoden Six Sigma und Lean Management. Beide Methoden sind entstanden, um die Fertigungsproduktion zu optimieren bzw. zu verbessern. Im Lean Management ist das Ziel, Verschwendung zu eliminieren und damit die Durchlaufzeiten zu verkürzen. Bei Six Sigma geht es hauptsächlich darum, jegliche Fehler bzw. Fehlermöglichkeiten zu beseitigen [3]. Mithilfe der Tools soll die Prozessqualität von Verwaltungsprozessen erhöht werden.

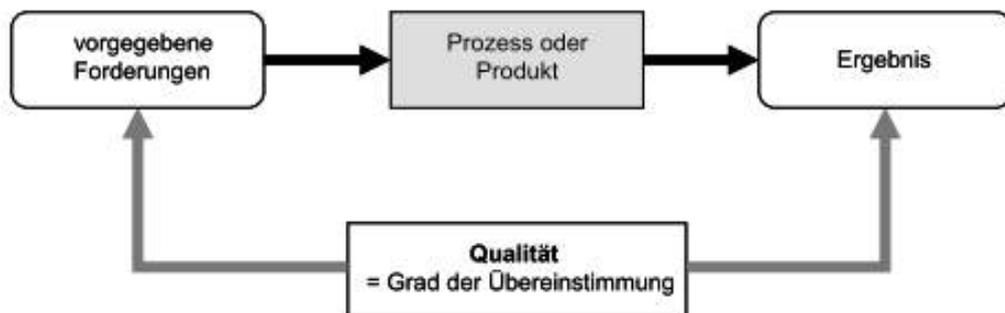


Abbildung 1: Definition von Qualität

Die Definition von Qualität ist in Abbildung 1 dargestellt und ist das Verhältnis zwischen Ist- und Soll- Merkmalen. Es gibt verschiedene Arten von Merkmalen an denen die Anforderungserfüllung gemessen werden kann, zum Beispiel funktionale Merkmale (Geschwindigkeit) oder ökonomische Merkmale (Kosten, Aufwand).

Die Auswahl der Tools erfolgte anhand des DMAIC-Zyklus von Six Sigma. In der Define Phase werden die zu erreichenden Ziele bzw. Leistungsmerkmale (Soll-Zustand) festgelegt. In der Measure Phase werden Daten zum Prozess gesammelt. In der Analyse Phase werden Ursachen für die Abweichung von den definierten Leistungsmerkmalen identifiziert.

^{*}Diese Arbeit wurde durchgeführt bei der Firma Capgemini, Stuttgart

In der Improve Phase werden Verbesserungsmaßnahmen gesucht und bewertet. In der Control Phase wird der Prozess fortdauernd überwacht. Es wird die Wirksamkeit der umgesetzten Verbesserungsmaßnahmen überprüft. Zusätzlich wird darauf geachtet, dass der abgeänderte Prozess zum neuen Standard wird.

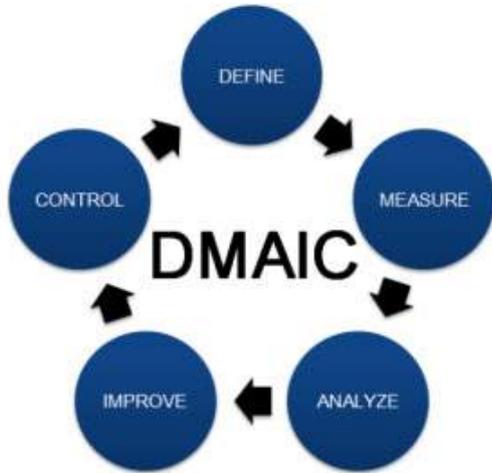


Abbildung 2: DMAIC-Zyklus

Es wurden verschiedene Tools, die zu jeweils unterschiedlichen Phasen des DMAIC-Zyklus gehören, ausgewählt. Zusätzlich zu der Process Map wird der Prozess mit der Methode „Value Stream Mapping“ abgebildet. Value Stream Mapping kommt aus dem Lean Management und wird zu den Phasen Define und Measure zugeordnet. Durch Value Stream Mapping werden zusätzlich zum Prozess einige wichtige Informationen wie Bearbeitungs- und Wartezeiten aufgezeigt. In Verwaltungsprozessen geht es hauptsächlich um Informationsverarbeitung, weswegen der Materialfluss, der im normalen Value Stream Mapping abgebildet ist, durch Input und Output der Prozessschritte ersetzt wurde. Der Prozessfluss

an sich und die Informationen zum Prozessfluss können als Basis für weitere Tools benutzt werden. Mithilfe des Value Stream Mapping wird Verschwendung erkannt und eliminiert. Diese unnötigen Prozessschritte sollten am Anfang entfernt werden, sodass im späteren Verlauf keine nutzlosen Aktivitäten optimiert werden. Zusätzlich wird anhand der Bearbeitungszeiten der Prozessschritt erkannt, der am meisten Aufwand verursacht. Dieser kann dann optimiert werden. Die zweite Methode ist die Cause und Effekt Matrix. Diese Methode gehört zu der Analyse Phase. Mit ihr werden die Inputs bewertet. Schlussendlich soll sie ein Ranking liefern, welche Inputs am wichtigsten für den Output sind. Mithilfe des Rankings können Verbesserungsmaßnahmen, welche die Inputs betreffen, priorisiert werden. Alternativ können für die wichtigsten Inputs dediziert Verbesserungsmaßnahmen gesucht werden. Die letzte Methode ist die Fehlermöglichkeitseinflussanalyse. Mithilfe dieser werden Verbesserungsmöglichkeiten bzw. Fehler und Maßnahmen zur Fehlervermeidung gefunden. Diese werden anhand der drei Kriterien Auftrittswahrscheinlichkeit, Entdeckungswahrscheinlichkeit und Bedeutung (Auswirkung des Fehlers) bewertet. Bei dieser Methode sind die Mitarbeiter eine wichtige Quelle für die in der Vergangenheit vorgekommenen Fehler. Die Fehler können in Verbindung mit dem Value Stream Mapping betrachtet werden, sodass der Prozessschritt der die meisten Fehler produziert, identifiziert werden kann. Im Gegensatz zu Produktionsprozessen liegen für Verwaltungsprozesse weniger Daten bereit. Die wichtigste Informationsquelle sind die Mitarbeiter, welche am Prozess beteiligt sind. Anhand ihrer Erfahrungen lassen sich Durchschnittswerte und weitere Information sammeln, die für die Anwendung der Tools notwendig sind.

-
- [1] SCHÖTZ, S., S. BUTZER, P. MOLEND, T. DREWS und R. STEINHILPER, 2017. An Approach Towards an Adaptive Quality Assurance [online]. *Procedia CIRP*, 63, 189–194. ISSN 22128271. Verfügbar unter: doi:10.1016/j.procir.2017.03.096
- [2] WITTENSTEIN, A.-K. und M. WESOLY, Hg., 2006. *Lean Office 2006*. Studie. Stuttgart: Fraunhofer-IRB-Verl. ISBN 3-8167-7146-7
- [3] ALPHADI. *Lean Six Sigma* [online] [Zugriff am: 1. November 2017]. Verfügbar unter: <http://www.alphadi.de/seminare/lean-six-sigma.html>

Bildquellen:

- Abbildung 1: BRÜGGEMANN, H. und P. BREMER, 2015. *Grundlagen Qualitätsmanagement. Von den Werkzeugen über Methoden zum TQM*. 2., überarb. und erw. Aufl. Wiesbaden: Springer Vieweg. Lehrbuch. ISBN 978-3-658-09221-4
- Abbildung 2: VILLANOVA UNIVERSITY. *Six Sigma: DMAIC Methodology* [online] [Zugriff am: 21. November 2017]. Verfügbar unter: <https://www.villanovau.com/resources/six-sigma/six-sigma-methodology-dmaic/>

Neuentwicklung eines Web-Frontends für das Benutzer Rollen-Rechte-System einer zentralen Administrationsplattform

Tobias Klingel*, Kai Warendorf, Dominik Schoop

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

3,58 Milliarden Menschen nutzen das Internet [1] und obwohl es sich dabei um unterschiedlichste Personen, jede mit ihren eigenen Anforderungen, handelt, sollen doch alle dieselben Webanwendungen bedienen können. Dies stellt eine große Herausforderung für jedes Frontend dar. In diesem Artikel soll auf die Arbeit an einem Frontend für ein Benutzer Rollen-Rechte-System eingegangen werden und worauf man bei der Entwicklung achten muss.

Das Benutzer Rollen-Rechte System ist Teil einer zentralen Administrationsplattform (ZAP). Gruppenadministratoren sollen neue Nutzer zu ihren Gruppen hinzufügen und diesen eine Auswahl an Rechten verleihen können die den Usern entweder erlauben andere Teile der ZAP zu benutzen (beispielsweise E-Mails zu versenden) oder auf andere Anwendungen zu zugreifen.

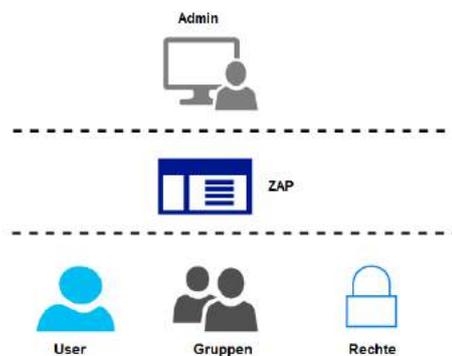


Abbildung 1: System Übersicht

Für ein System, das sich mit Benutzerrechten befasst, bieten sich besondere Herausforderungen. Ein Fehler kann schnell dazu führen, dass der falsche Benutzer das falsche Recht hat und damit wissentlich oder unwissentlich Schaden verursachen kann, da er eventuell Zugriff auf Funktionen und Informationen erhält, die nicht für ihn gedacht waren.

Damit solche Vorfälle nicht auftreten muss das User Interface besonders klar sein. Jeder Vorgang soll für die Administratoren verständlich und Nachvollziehbar ablaufen. Vor allem müssen die Administratoren stets das Gefühl haben die volle Kontrolle über das System (zumindest den Teil des Systems der aktiv von ihnen genutzt wird) zu haben.

Um dies zu erreichen kann eine Vielzahl an Richtlinien und Regeln beachtet werden, beispielsweise können die 10 Usability Heuristiken nach Jakob Nielsen [2] zurate gezogen werden, die Heuristiken können als Richtlinien verstanden werden die dabei helfen die Usability eines User Interfaces zu verbessern. Im Folgenden sollen nur ein paar der Heuristiken in Anbetracht auf das Frontend genannt werden.

Die erste Heuristik ist die „Sichtbarkeit des Systems“ der User soll stets darüber informiert sein, was das System gerade macht. In diesem Fall bedeutet dies beispielsweise die Administratoren darüber zu informieren, ob ihre Aktionen, beispielsweise das Hinzufügen eines Users zu einer Gruppe, erfolgreich waren.

Eine weitere Heuristik umfasst „Consistency and standards“, innerhalb einer Anwendung soll die gleiche Aktion immer das Gleiche bewirken und das Vokabular soll innerhalb der Anwendung konsistent bleiben (z. B. sollen Erstellen, Bearbeiten und Löschen nicht auf einmal Erzeugen, Ändern und Entfernen genannt werden).

Zuletzt wird die Heuristik „Help users recognize, diagnose and recover from errors“ betrachtet. Wenn ein Fehler im System auftritt, zum Beispiel ein Recht nicht vergeben oder neu erstellt werden konnte, soll dies den Gruppenadministratoren in klaren Worten erklärt und gegebenenfalls zusammen mit einem Lösungsvorschlag dargestellt werden.

*Diese Arbeit wurde durchgeführt bei der Firma DSV Service GmbH, Stuttgart

Dies ist nur ein Ausschnitt der Heuristiken und Design Prinzipien, die bei der Entwicklung eines Frontends für die Usability und User Experience beachtet werden können, wichtig ist zu erkennen, was das zu entwickelnde System für Prioritäten hat und den Fokus darauf zu setzen.

Technologie: Das Frontend wird mit dem Angular Framework in der Version Angular 5 entwickelt. Angular ist ein open-source Frontend-Webapplikationsframework welches auf Typescript basiert und von Google entwickelt wird.

Eine mit Angular entwickelte Anwendung besteht aus mehreren Komponenten, jede Komponente besitzt ihr eigenes Layout. Die Komponenten können modular ausgetauscht

und kombiniert werden und ermöglichen die Erstellung einer strukturierten, wartbaren Applikation.

Das Backend bietet über eine REST-Schnittstelle Zugriff auf die benötigten Daten, die in einer MySQL-Datenbank gespeichert sind, außerdem nutzt angular RxJS, eine ReactiveX Bibliothek für Javascript.

ReactiveX erweitert den "Observer pattern", Daten, die von der REST-Schnittstelle kommen, erhält die Anwendung in Form von Observables, auf diese muss ein „subscribe“ ausgeführt werden, um die Daten verarbeiten zu können. ReactiveX umfasst eine Vielzahl an Operatoren (map, retry, etc.), die das Arbeiten mit Observables erleichtern.

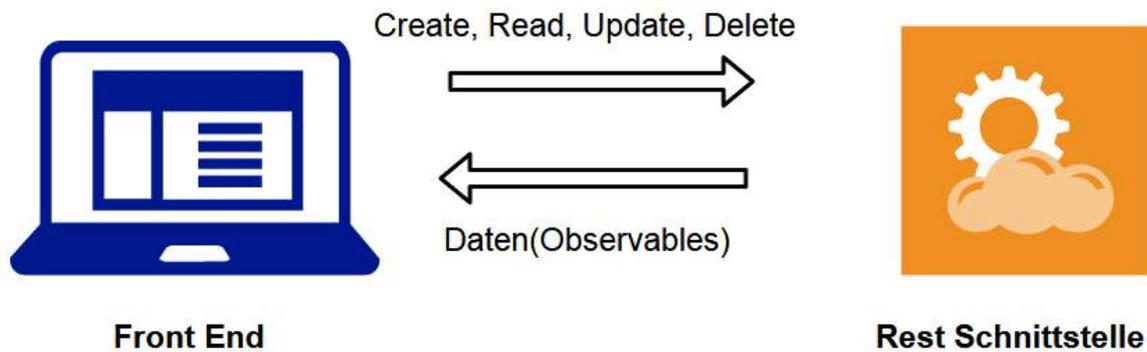


Abbildung 2: Verbindung von Front- und Backend

[1] Number of internet users worldwide from 2005 to 2017 (in millions)

<https://www.statista.com/statistics/273018/number-of-internet-users-worldwide/>

[2] Nielsen, J. (1994a). Enhancing the explanatory power of usability heuristics. Proc. ACM CHI'94 Conf. (Boston, MA, April 24-28), 152-158

Bildquellen:

- Abbildung 1, 2: Eigene Darstellung

Cloudgestütztes Devicemanagement – Infrastruktur und Wartungssystem für selbstständige Devices ohne Nutzereingriff

Daniel Kratzel*, Dominik Schoop, Kai Warendorf

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Herausforderung

In heutigen Fahrzeugen steckt eine Vielzahl von Sensoren mit unterschiedlichen Funktionen, um neuartige Fahrerassistenzsysteme zu ermöglichen, die den Autofahrer im Verkehr unterstützen. Damit wurden Autos über die vergangenen Jahre immer mehr zu rollenden Computern. Um an neuen Möglichkeiten der Unterstützung zu forschen und bestehende Funktionen zu verbessern, wird eine Vielzahl an Felddaten benötigt, die im Fahrzeug aufgezeichnet werden müssen. Folglich wächst die Menge an generierten Daten im Fahrzeug von Jahr zu Jahr zum Beispiel durch hochauflösende Kameras oder erweiterte Sensorik. Diese riesigen Datenmengen können mit heutigen Mobilfunkstandards nicht in Echtzeit in einen zentralen Speicher in der Cloud hochgeladen werden.

In Zukunft muss daher ein Weg gefunden werden, die zu übertragende Datenmenge auf ein überschaubares Maß zu reduzieren und gleichzeitig keine wichtigen Informationen zu verlieren.

Lösungsansatz

Eine lokale Sicherung der aufgezeichneten Daten auf Festplatten lokal im Fahrzeug und ein späteres Auslesen nach einer Versuchsfahrt ist grundsätzlich möglich. Im Hinblick auf die Entwicklung von autonomen Fahrsystemen ist das ein Rückschritt. Dynamische Entwicklungsprozesse werden dadurch ausgebremst. Um dieses Problem zu lösen, ist ein Computerdevice im Fahrzeug vorgesehen. Für einen Entwickler relevante Daten können während der Fahrt in die Cloud geladen werden und stehen ihm dort zur Verfügung. Das Device besitzt die Möglichkeit, bei Bedarf alle Sensor- und Kameradaten sowie Signale von Steuergeräten aufzuzeichnen. Es kann vor Ort im Fahrzeug eine Vorverarbeitung und Aufbereitung der Daten vornehmen. Die Menge an Daten wird auf die tatsächlich relevante reduziert. Ein Entwickler erhält nur genau die Daten und Informationen, die für seine Arbeit an einer bestimmten Assistenzfunktion von Nutzen sind.

Zu erkennen ist hier der Edge Computing Ansatz. Durch eine Vorverarbeitung, Analyse und Filterung der Daten direkt an den Netzwerkknoten, in diesem Fall lokal im Fahrzeug, wird der zentralisierten Verarbeitung von großen Datenmengen in der Cloud entgegengewirkt [1].

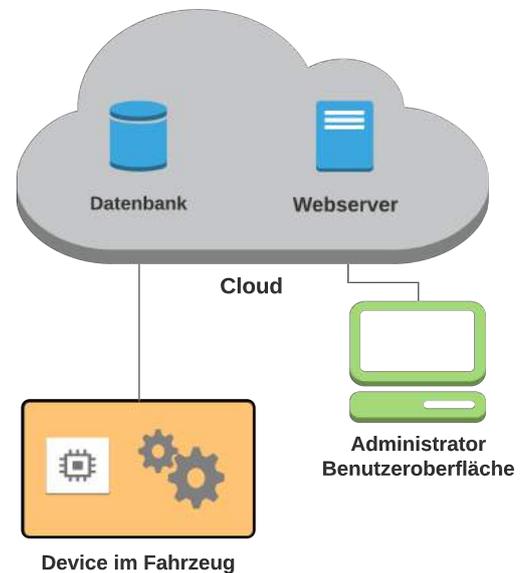


Abbildung 1: Zusammenspiel der Systemkomponenten

Das Device besteht aus unterschiedlichen Einheiten, die verschiedene Aufgaben erfüllen. Eine Powermanagementeinheit kümmert sich um eine konstante Energieversorgung. Die Kommunikationseinheit besitzt ein LTE-Modul und ist die Verbindung zur Cloud. Eine leistungsstarke Recheneinheit ist für das Sammeln und Verarbeiten der Daten im Fahrzeug zuständig. Aufgrund der hohen Rechen- und Grafikleistung können auch Aufgaben aus den Bereichen Computer Vision und Maschinelles Lernen übernommen werden.

Um die Devices nicht in jedem Fahrzeug einzeln konfigurieren zu müssen, ist ein Devicemanagement mit Fernzugriff über die Cloud vorgesehen. Im Rahmen dieses Device-

*Diese Arbeit wurde durchgeführt bei der Firma Robert Bosch GmbH, Stuttgart-Feuerbach

managements ist das Ziel dieser Arbeit die Entwicklung einer webbasierten Anwendung zur Steuerung und Konfiguration der Devices in den Fahrzeugen. Die zu entwickelnde Administratoroberfläche wird, wie in Abbildung 1 dargestellt, über einen Webserver in der Cloud bereitgestellt und hat Zugriff auf eine Datenbank, in der Informationen über die Devices gespeichert und bereitgestellt werden. Durch die Kommunikation von Frontend zu den Devices über die Cloud ist es möglich, diese zu verwalten und deren Status zu überwachen. Hierzu zeigt Abbildung 2 einen ersten Entwurf der Administratoroberfläche im Webbrowser. Des Weiteren soll es über das Frontend möglich werden, laufende Applikationen der Devices anzuzeigen, neue Applikationen auf ein Device aufzuspielen und diese starten und stoppen zu können.

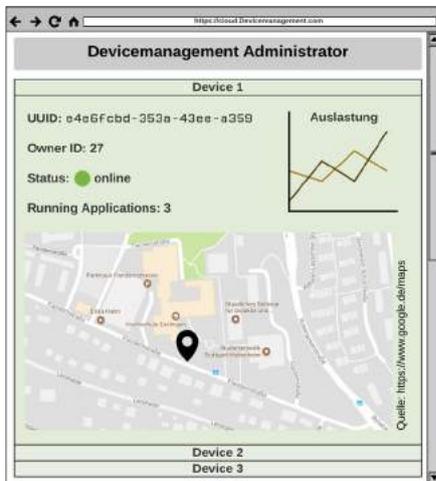


Abbildung 2: Entwurf der Devicemanagement Oberfläche

Umsetzung

Das Frontend wird als JavaScript-Applikation entwickelt. Es kommt die von Facebook entwickelte React-Bibliothek zum Einsatz. Diese basiert auf Komponenten, die zu größeren komplexen interaktiven Benutzeroberflächen zusammengesetzt werden. React verwendet hierbei verschiedene Konzepte der ES6 (ECMAScript 6) Syntax wie Arrow-Funktionen oder Klassen. Darüberhinaus verwendet React eine spezielle Syntaxerweiterung namens JSX. Mithilfe von JSX lassen sich React-Elemente erstellen, die in das Document Object Model (DOM) einer HTML-Seite gerendert werden.

Abbildung 3 zeigt die Kommunikation, das Anfragen und Übermitteln von Daten, der React-Oberfläche mit der Cloud über GraphQL. GraphQL ist eine von Facebook entwickelte Schnittstellendefinition mit großem Potential im Bereich der Kommunikation zwischen verschiedenen Services. Bei der Erstellung von Clientapplikationen unterstützt GraphQL mit Hilfe einer intuitiven und speziellen Syntax zur Anfrage und Veränderung von Daten. GraphQL ist dabei sehr flexibel und nicht an ein bestimmtes Datenbanksystem gebunden.

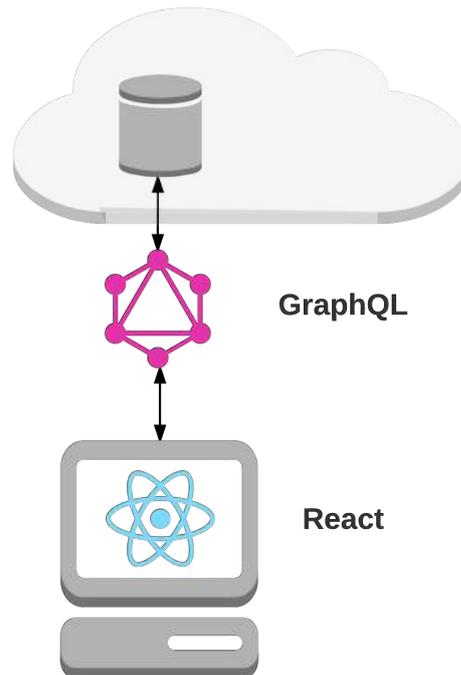


Abbildung 3: Einsatz der Technologien

Ausblick

In Zukunft wird in der IoT-Welt neben dem Vorantreiben von neuen Übertragungsstandards wie 5G auch eine effektive Reduktion der Daten entscheidend sein. Durch eine Verringerung der Datenmenge vor der Übertragung kann trotz der weiter steigenden Zahl an internetfähigen Geräten die Netzwerkbelastung im Rahmen gehalten werden. Auf diesem Weg wird auch in Zukunft die Stabilität von Netzwerken gewährleistet.

[1] Stefan Mutschler, Edge Computing – Randerscheinung mit zentraler Bedeutung, 18.9.2017, URL: www.golem.de/news/edge-computing-randerscheinung-mit-zentraler-bedeutung-1709-129248.html

Bildquellen:

- Abbildung 1-3: Eigene Darstellung

Projekt „DAISI“ – Entwicklungsmethoden und Erfolgsfaktoren zur Etablierung eines neuen Social Intranet bei Daimler

Alexandra Kühner*, Astrid Beck, Reinhard Schmidt

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Die Daimler AG befindet sich mitten in einem grundlegenden Wandel. Neue Mobilitätskonzepte, neue Geschäftsmodelle, neue Wettbewerber stellen das Unternehmen vor bislang unbekannte Herausforderungen. Daimler antwortet auf diese Herausforderungen mit neuen Technologien, neuen Strategien in der Mitarbeitergewinnung, neuen Initiativen und vor allem mit einem neuen Zusammenarbeitsmodell. Hier wird mit der Einführung eines Social Intranet ein wichtiger Beitrag zur Entwicklung der neuen Unternehmens- und Zusammenarbeitskultur geleistet. Vernetzte Produkte und Dienstleistungen benötigen ein vernetztes Unternehmen über Bereichs- und Hierarchiegrenzen hinweg. Ein Social Intranet bietet neue Möglichkeiten, um an genau diejenigen Informationen und Gesprächen teilzuhaben, die für den Mitarbeiter und sein Arbeitsfeld relevant sind. Hinzu kommt die Möglichkeit, im Team oder Bereich jenseits örtlicher oder zeitlicher Einschränkungen effizient zusammenzuarbeiten.

Neben den heute schon verfügbaren Funktionen des Daimler Intranets, kommen zusätzlich umfangreiche Möglichkeiten zur vernetzten Zusammenarbeit und Kommunikation hinzu – diese gehen noch über die Funktionen hinaus, die heute schon von dem Daimler Mitarbeiterportal, #onair (einer Nachrichtenplattform auf dem Mitarbeiterportal) und Daimler CONNECT (einer auf Sharepoint basierten Social-Lösung) abgedeckt werden.

Ebenso soll das neue Tool den Mitarbei-

terinnen und Mitarbeitern ihre tägliche Arbeit erleichtern. Dazu wird es die Stärken der drei derzeit existierenden Systeme auf einer Plattform bündeln. In einem Social Intranet werden die Benutzer, ähnlich wie im bisherigen Intranet, Unternehmensnachrichten und Anwendungen für ihren Arbeitsalltag finden. Gleichzeitig können Sie sich auf derselben Plattform mit Kollegen auf der ganzen Welt vernetzen und kommunizieren. Bei einem Social Intranet stehen, anders als bei einem herkömmlichen Intranet, die dynamische Wissensvermittlung und die gemeinsame kollaborative Zusammenarbeit im Vordergrund. In einem modernen Intranet werden Inhalte nicht (nur) zentral durch eine Redaktion zur Verfügung gestellt, sondern von Nutzern auch aktiv erstellt und mitgestaltet. Der User-Generated-Content wird von den Abteilungen direkt veröffentlicht und kann, je nach Freigabe und Zulassung, von Kollegen ergänzt oder geändert werden. Mitarbeiter können das Erscheinungsbild des Intranets nach ihren eigenen Interessen und Wünschen individuell gestalten. Sie können sich untereinander vernetzen, Blogbeiträge schreiben und sich aktiv in Wikis und Foren an der firmenweiten Kommunikation beteiligen.

Zugriff auf das Social Intranet haben alle internen Mitarbeiter (inkl. ANÜs) der Daimler AG und deren Tochtergesellschaften mit Anteil über 50% sowie ausgewählte Joint-Venture Gesellschaften in China.



Abbildung 1: Die Nachrichtenplattform #onair, Daimler Connect und das Mitarbeiterportal

*Diese Arbeit wurde durchgeführt bei der Firma Daimler AG, Untertürkheim

SOCIAL INTRANET  Apps  Erstellen

HOME NEWS PERSONEN COMMUNITIES MEIN STANDORT UNTERNEHMEN DAIMLER & ICH HILFE

Mercedes-Benz



Dr. Dieter Zetsche

Mercedes-Benz Marketing

Social Media Kampagne „A Guide to Growing Up“

287 117 53

2 | 5 >

Guide



Entdecke jetzt dein Social Intranet.

Lorem ipsum dolor sit amet, consetetur sadipscing elitr.

Später Loslegen

MY WORKSPACE MY NEWSFEED MY TIMELINE

+ Widget hinzufügen Workspace bearbeiten



Johanna Catharina Blomendahl
Senior Consultant, Marketing

MEINE NACHRICHTEN

 Deniel Desk 14:08 Uhr

Hello Again!

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquy ...

 Max Muster 9:00 Uhr

Meeting verschoben

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquy ...

 Lisa Muster Gestern

Mittagessen um 12:30?

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquy ...

MEINE APPS

TIME MANAGER BUSINESS TRIPS PROCUREMENT SYSTEMS CAR POOL WEATHER FORECAST

Mehr Apps anzeigen

NEWS

Sindelfingen



Cars & Coffee am Mercedes-Benz Museum

Zwölf Sommersonntage mit Benzingeräuschen

287 117 53

Daimler



DAS BESTE ODER NICHTS.

Gottlieb Daimler

287 117 53

Mercedes-AMG



Celebrating 50 Years of Driving Performance

Linkin Park Konzert powered by Mercedes-AMG

287 117 53

Abbildung 2: Open Beta Version des zukünftigen Daimler Social Intranet

Machbarkeitsanalyse zur technischen Realisierbarkeit der Systemerweiterung IO-Link Wireless des unter IEC 61131-9 standardisierten Kommunikationssystems IO-Link

Wolfram Ladurner*, Reinhard Keller, Walter Lindermeir

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

In Zeiten von zunehmender Fertigungsautomatisierung wächst die Zahl an Sensoren und Aktoren ständig. Vermehrt entsteht hierbei die Notwendigkeit, Sensoren und Aktoren auch an rotierenden oder beweglichen Maschinenteilen wie z.B. Roboterarmen anzubringen, wo eine kabelgebundene Anbindung oft schwierig zu realisieren und fehleranfällig ist. Hier kommt IO-Link Wireless ins Spiel:

Überblick über IO-Link Wireless

IO-Link Wireless wird mit dem Ziel entwickelt, eine Alternative zu der bisher notwendigen Verkabelung für die Anbindung von Sensoren/Aktoren in der Automatisierungstechnik zu bieten. Hierbei liegt der Schwerpunkt auf einer robusten Echtzeitkommunikation mit geringer Latenz. Wichtige Anwendungen für die Technologie sind insbesondere Sensoren und Aktoren an rotierenden oder beweglichen Maschinenteilen.[1]

Für die Funkverbindung wird das lizenzfreie ISM-Band bei 2,4 GHz verwendet. Wie auch beim kabelgebundenen IO-Link erfolgt die Kommunikation hierbei nach dem Master-Slave-Prinzip: Es kommuniziert immer ein W-Master (*wireless master*) mit einem oder mehreren W-Devices (*wireless devices*), die einzelnen W-Devices kommunizieren jedoch niemals untereinander.

Ein W-Master kann bis zu fünf Sendeempfänger (Transceiver) beinhalten, die jeweils über eine eigene Antenne sowie über eigene Frequenzkanäle, sogenannte Tracks, verfügen. Je Track können bis zu 8 W-Devices angesprochen werden, also kann ein W-Master insgesamt mit bis zu 40 W-Devices kommunizieren.[1] Der grundlegende Aufbau ist in Abbildung 1 dargestellt.

Nach erfolgreichem Verbindungsaufbau werden über unterschiedliche logische Kanäle Daten zwischen W-Master und W-Device übertragen.

Über den **Prozessdaten-Kanal** werden Eingangsprozessdaten und/oder Ausgangsprozessdaten übertragen.

Über den **ISDU-Kanal** (ISDU: Indexed Service Data Unit) können bei Bedarf Service-daten/Parameter übertragen werden. Hierüber kann beispielsweise eine Konfiguration/Parametrierung von Sensoren, also z.B. das Einstellen einer Schaltschwelle oder der Hysterese erfolgen.

Der **Diagnose-Kanal** ermöglicht es dem W-Device, Ereignisse (*events*) in Form von Fehlern, Warnungen und Meldungen an den W-Master weiterzuleiten.

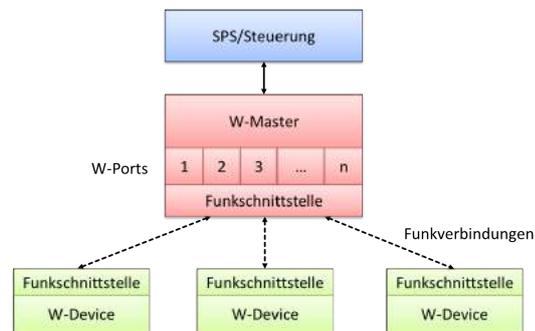


Abbildung 1: Grundlegender Aufbau IO-Link

Die IO-Link Wireless Spezifikation

Innerhalb der Spezifikation sind W-Master und W-Device in einzelne Module unterteilt, die gemäß der Schichtenarchitektur in Anlehnung an das OSI-Modell vertikal aufeinanderfolgend angeordnet sind. Der Aufbau ist in Abbildung 2 verdeutlicht.

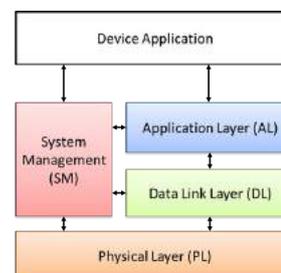


Abbildung 2: Das Schichtenmodell des IO-Link Wireless W-Device-Stacks

*Diese Arbeit wurde durchgeführt bei der Firma Balluff GmbH, Neuhausen a.d.F.

Der **Physical Layer** (PL) regelt den Zugriff auf das Übertragungsmedium. Er ist verantwortlich für die Ansteuerung des Sendeempfängers, über den die einzelnen Bits gesendet bzw. empfangen werden. Auch das Pairing fällt in den Zuständigkeitsbereich des Physical Layers.

Der **Data Link Layer** (DL) sorgt für eine fehlerfreie und zuverlässige Datenübertragung, indem Daten, deren Empfang von der Gegenseite nicht quittiert wurde, erneut gesendet werden.

Der **Application Layer** (AL) stellt der eigentlichen Anwendung eine Schnittstelle zum Senden bzw. Empfangen von Prozessdaten, Events und Parametern zu Verfügung. Für die Übertragung der Daten nutzt er die Dienste des darunterliegenden DL.

Das **System Management** (SM) nimmt eine Sonderrolle ein, da es sich nicht streng in die Hierarchie der einzelnen Schichten einordnet, sondern Dienste von mehreren Schichten (PL, DL, auf W-Master-Seite auch AL) nutzt. Seine Aufgabe ist im Wesentlichen die Koordination des Start- und Konfigurationsvorgangs innerhalb des W-Masters und innerhalb der einzelnen W-Devices.

Jede der Kommunikationsschichten bietet der darüberliegenden Schicht ihre Dienste (*services*) an. Für jeden dieser Dienste gibt es eine Funktionsbeschreibung sowie eine genaue Beschreibung der einzelnen Parameter, die den jeweiligen Dienstelementen zugeordnet sind. Alle Schichten enthalten zudem einen oder mehrere Zustandsautomaten (*state machines*), die die internen Zustände der jeweiligen Schicht sowie das Zusammenspiel der einzelnen Dienste beschreiben. Ein solcher Zustandsautomat ist beispielhaft in Abbildung 3 gezeigt.

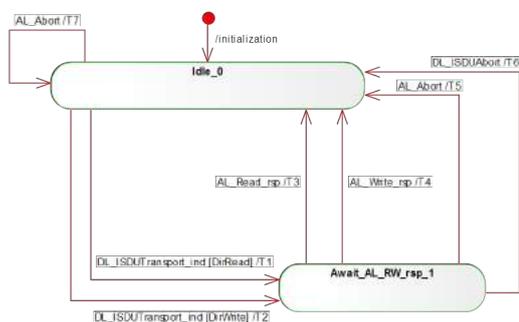


Abbildung 3: ISDU-Zustandsautomat W-Device AL

Aufgabenstellung und Lösungsansätze

Im Rahmen der Bachelorarbeit sollte eine Machbarkeitsanalyse zur technischen Realisierbarkeit der IO-Link Wireless System Extensions Spezifikation (Version 095) erstellt werden. Die Spezifikation sollte hierbei im Hinblick auf ihre Umsetzung in Hardware und Software auf Realisierbarkeit, Vollständigkeit und Konsistenz geprüft werden. Zudem sollten Testfälle extrahiert werden, um so die Vorarbeit für ein automatisiertes Testen der einzelnen Komponenten zu leisten. Der Schwerpunkt lag hierbei jeweils auf der Seite der W-Devices.

Die erste Teilaufgabe bestand darin, die kritischen Faktoren innerhalb der Spezifikation zu bestimmen, die für eine Umsetzung entscheidend sind. Als besonders erfolgsentscheidend haben sich hier die Zustandsautomaten sowie die einzelnen Dienste herausgestellt.

Anschließend wurden die Zustandsautomaten und die Dienste systematisch auf Vollständigkeit, Konsistenz und Plausibilität geprüft.

Hierzu wurden unter anderem auch einzelne Zustandsautomaten mit den zugehörigen Diensten prototypisch implementiert (*proof of principle*).

Um die Vorarbeit für ein späteres automatisiertes Testen zu leisten, wurden zunächst für die kritischen Fälle separate Testfälle definiert. Ein solcher Testfall für das AL sieht beispielsweise so aus: „Der Aufruf des Dienstes AL_SetInput muss einen Aufruf des Dienstes DL_PDInputUpdate auslösen.“

Anschließend wurden die Testfälle mit Hilfe des Test-Frameworks cmocka jeweils in vorläufige Test-Funktionen hinterlegt. Die Herausforderung bestand hierbei darin, dass jedes der Module bzw. jede der Schichten einzeln getestet werden können soll. Hierzu wurden jeweils die Schnittstellen zu den benachbarten Modulen durch Mock-up-Funktionen ersetzt, die dann als Platzhalter für die von dem zu testenden Modul jeweils verwendeten weiteren Module dienen.

Zusammenfassung und Ausblick

Viele der aus der Prüfung der Spezifikation auf Realisierbarkeit, Vollständigkeit und Konsistenz gewonnenen Erkenntnisse konnten als *change requests* in den Weiterentwicklungsprozess der Spezifikation einfließen und so zur Verbesserung beitragen.

Mit der Definition der Testfälle sowie mit deren Umsetzung innerhalb des Test-Frameworks sind die Voraussetzungen für eine spätere testgetriebene Entwicklung des Software-Stacks der W-Devices geschaffen.

[1] IO-Link Community (Hrsg.): IO-Link Wireless System Extensions Specification, Draft Version 095. IO-Link Community, 2017

Bildquellen:

- Abbildung 1, 2, 3: Eigene Darstellung nach [1], eigene Darstellung, [1]

Konsolidierung und Darstellung von Monitoringinformationen auf einem zentralen Web-Dashboard

Patrick Leibold*, Jürgen Koch, Kevin Erath

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

In der heutigen Zeit werden in der IT mehr und mehr Monitoring Informationen von verschiedensten IT-Monitoring-Tools gesammelt. Diese werden meist auf ihren eigenen Web-Dashboards dargestellt. Dadurch befinden sich die Informationen an vielen verschiedenen Orten verteilt. Um die Informationen an einem Ort zu bündeln, soll ein zentrales Web-Dashboard erstellt werden, das sich die Monitoring-Informationen von den IT-Systemen über bereitgestellte API-Schnittstellen beschafft, verarbeitet und grafisch aufbereitet darstellt.

Monitoring: Unter Monitoring versteht man die unmittelbare, systematische Erfassung, Messung oder Überwachung von Vorgängen oder Prozessen. Regelmäßig sich wiederholende Durchführungen sind ein zentrales Element um anhand von Ergebnisvergleichen, Probleme oder Trends erkennen zu können. Das Ziel aller Monitoring-Aktivitäten in der Informationstechnik ist es, einen reibungslosen Betrieb der IT-Infrastruktur zu gewährleisten. Grundsätzlich gibt es zwei Herangehensweisen: das Historical Monitoring und das Real Time Monitoring.

Beim Historical Monitoring werden die über einen Zeitraum zu IT-Systemen und IT-Diensten erfassten Daten, in Langzeitstatistiken zusammengefasst. Die daraus gewonnenen Erkenntnisse können angewendet werden, um zum Beispiel über Wachstumstrends bei der zukünftigen Ressourcenplanung zu helfen oder systematisch wiederkehrende Probleme zu diagnostizieren.

Beim Real Time Monitoring werden die von den durchgängig überwachten IT-Systemen und IT-Diensten gewonnenen Daten aktiv auf Auffälligkeiten oder Störungen hin überprüft, um im Notfall schnellstmöglich reagieren zu können. Dadurch können größere Probleme verhindert oder behoben werden, bevor es zu einer Beeinträchtigung der IT-Infrastruktur kommt. In den heute gängigen IT-Monitoring-Lösungen kommen beide Herangehensweisen zum Einsatz [1].

Bei der Entwicklung des Web-Dashboards wurde der MEAN Stack verwendet. Dieser besteht aus vier Technologien: MongoDB, Express.js, Angular und Node.js. Diese eignen

sich in Kombination miteinander hervorragend zur Entwicklung von Webapplikationen, sind daher weitverbreitet und werden oft zusammen genutzt.

MongoDB: Bei MongoDB, abgeleitet vom englischen humongous (riesig), handelt es sich um eine sogenannte dokumentenorientierte NoSQL Datenbank und grenzt sich dadurch von den traditionellen SQL-Datenbanken ab. MongoDB ist kostenlos, Open Source und funktioniert auf fast allen Betriebssystemen. Es wurde 2007 von der Firma 10gen entwickelt und 2009 zum ersten Mal veröffentlicht. Im Jahr 2013 hat sich die Firma in MongoDB Inc. umbenannt. Die Struktur einer MongoDB-Instanz lässt sich grundsätzlich wie folgt beschreiben, siehe Abbildung 1.

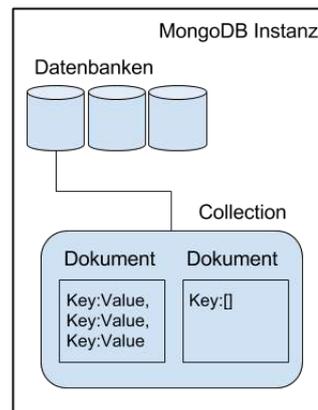


Abbildung 1: Aufbau einer MongoDB Instanz

Ein Datenbankserver verwaltet eine oder mehrere logische Datenbanken. Diese verwalten wiederum eine oder mehrere so genannte Collections. Eine Collection ist eine Ansammlung von einem oder mehreren Datensätzen, auch Dokumente genannt. Mit Dokumenten ist hier eine geordnete Menge von Key-Value-Paaren gemeint. Ein Value kann ein einfacher Datentyp, ein Array von Values oder ein eingebettetes Dokument sein. Es können für Collections Schemas erstellt werden, müssen aber nicht. Falls ein Betrieb ohne Schema gewählt wird, muss vorsichtig vorgegangen werden, da die Datenbank keine Validierung mehr

*Diese Arbeit wurde durchgeführt bei der Firma IT-Designers GmbH, Esslingen

für gespeicherte Dokumente und deren Inhalte übernehmen kann und diese Überprüfung in der Applikation selbst vorgenommen werden muss. Die Dokumente werden intern als BSON, ein binäres JSON-ähnliches Format, gespeichert [2].

Express.js: Bei Express.js handelt es sich um ein einfaches, flexibles Node.js-Framework für die Entwicklung von Webanwendungen. Es bietet eine Vielzahl von HTTP- und Middleware-Funktionen, um schnell und einfach robuste API-Schnittstellen für Webanwendungen zu erstellen.

Angular: Angular (Version 4) ist ein Frontend-Framework für Webapplikationen, basierend auf Typescript. Typescript ist eine von Microsoft entwickelte, auf dem ECMAScript-6-Standard basierende Programmiersprache. Jeder Javascript-Code ist auch gültiger Typescript-Code. Dadurch können alle gängigen Javascript-Bibliotheken, auch in Typescript, verwendet werden [3]. Bei Angular handelt es sich um eine Weiterentwicklung von AngularJS und wurde von Grund auf neu geschrieben. Daher ist in AngularJS geschriebener Code, mit den neueren Versionen nicht mehr kompatibel. Angular ist ein Open Source Produkt, das sowohl von einer Community aus Einzelpersonen, als auch von Unternehmen entwickelt wird, allem voran Google.

Node.js: Node.js ist eine Plattform zur Entwicklung von serverseitigen Webapplikationen, diese basiert auf der Chrome V8 Engine, um Javascript-Code serverseitig auszuführen. Die Chrome V8 Engine ist eine Open Source JavaScript Engine, die vom Chromium Project entwickelt wurde. Diese kompiliert Javascript-Code in nativen Maschinen-Code und führt diesen dann auf dem Dateisystem aus. Da-

durch sind Anwendungen, die mithilfe von Node.js entwickelt wurden, auf diversen Betriebssystemen lauffähig, darunter Linux, MacOS und Windows. Node.js stellt verschiedene Module bereit. Diese umfassen zum Beispiel Netzwerkfunktionalitäten wie HTTP, DNS, TCP, UDP, SSL/TLS, Dateisystem I/O Operationen und viele andere. Damit ist es nun möglich im Backend, wie im Frontend, Javascript zu verwenden ohne weitere Programmiersprachen erlernen zu müssen. Bei der Entwicklung von Node.js wurde besonders auf Datendurchsatz und Skalierbarkeit geachtet. Dies sieht man deutlich in der ereignisgesteuerten und auf nicht blockenden Prozessen aufgebauten Architektur. Dafür wurden blockende Prozesse, wie z.B. Zugriffe auf das Dateisystem oder auf das Netzwerk, umgeschrieben, um neben den synchron-ablaufenden, blockenden Prozessen eine asynchron-ablaufende, nicht-blockende Alternative anbieten zu können [4].

Systementwurf: Das Web-Dashboard besteht im Wesentlichen aus vier Komponenten, dem Frontend, dem Backend, der Datenbank und den Monitoring-Systemen, siehe Abbildung 2. Zur Implementierung einer Datenbank in das System hat man sich deshalb entschieden, um die angezeigten Daten an mehrere User konsistent zu halten und um die Anzahl der API-Anfragen an andere IT-Systeme so gering wie möglich zu halten. Die Kommunikation der Komponenten untereinander sieht wie folgt aus: Das Frontend sendet API-Requests an das Backend, dieses holt die Daten aus der Datenbank und leitet sie an das Frontend weiter. Das Backend sendet API-Requests an andere IT-Systeme, verarbeitet die Daten, bereitet die Ergebnisse auf und speichert sie in der Datenbank ab.

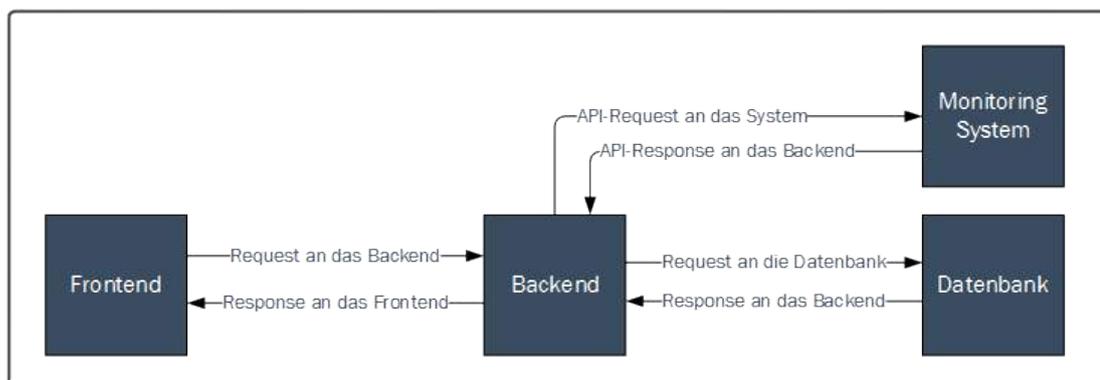


Abbildung 2: Systementwurf

[1] <http://www.it-administrator.de/themen/netzwerkmanagement/grundlagen/111034.html>

[2] <https://www.informatik-aktuell.de/betrieb/datenbanken/mongodb-fuer-software-entwickler.html>

[3] <https://en.wikipedia.org/wiki/TypeScript>

[4] <https://nodejs.org/en/docs/>

Bildquellen:

- Abbildung 1, 2: Eigene Abbildung

Methoden bereit, die es anderen Servern erlauben, sich bei diesen zu registrieren und *OPC UA Clients* ermöglicht, die an einem Discovery Server registrierten Server abzufragen.

Local Discovery Server (LDS) fassen mehrere *OPC UA Server* auf einem Gerät zusammen und agieren als Vermittler zwischen diesen Servern und anderen *OPC UA Devices* im Netzwerk.

Global Discovery Server (GDS) ermöglichen eine Netzwerkdomänen-übergreifende Verwaltung von *OPC UA Servern*, *Clients* und von Authentifizierungszertifikaten [2].

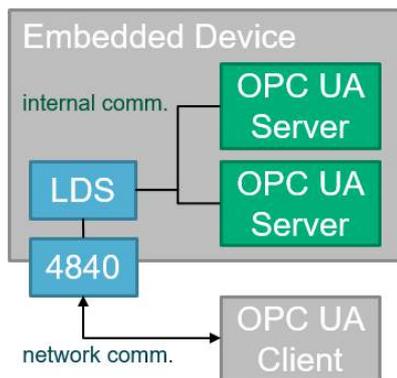


Abbildung 3: schematische Darstellung eines Embedded Devices mit Local Discovery Server

Sensorplattform Leuze BCL 300i

Für die vorliegende Arbeit war als Entwicklungsplattform das Barcodelesegerät *BCL 300i* von Leuze electronic gegeben.

Die Hauptfunktionalität des *BCL 300i* ist das Lesen von eindimensionalen Barcodes verschiedener Codierungen wie Code39, Code128 und CodeEAN. Zusätzlich kann der Sensor in Verbund mit einem Reflektor als Lichtschranke eingesetzt werden. Beispielsweise wartet das Gerät, an einem Paketförderband installiert, im Lichtschrankenmodus und beginnt den Scanvorgang bei Unterbrechung der Lichtachse durch ein Paket.

Konfiguration und Steuerung des Sensors sind über ein vom Gerät bereitgestelltes Webinterface und gängige Feldbus-Protokolle wie PROFIBUS, PROFINET, EtherCAT, und TCP/IP möglich [3].



Abbildung 4: Barcodelesegerät BCL 300i

Ergebnisse der Bachelorarbeit

Die Bachelorarbeit befasste sich mit der Analyse der *OPC UA Discovery Specifications* und deren Einsatzgebiete in vorhandenen Sensorprodukten von Leuze electronic.

Der bereits in der Firmware des *BCL 300i* vorhandene *OPC UA Server* wurde um die spezifizierten Funktionalitäten eines *Local Discovery Servers* erweitert, was durch die Integration der benötigten Funktionen aus einem kommerziellen OPC UA Software-Stack in die Firmware der Sensorplattform erreicht wurde.

Diese Erweiterung ermöglicht es dem Sensor zum einen, sich über mDNS (*multicast Domain Name System*) als *OPC UA Server* in einem lokalen Netzwerk bekannt zu machen, zum anderen, sich an anderen *Discovery Servern* zu registrieren.

Zur Bereitstellung einer Test-Infrastruktur für die Funktionalitäten von *Global Discovery* wurde zudem ein OPC UA GDS (*Global Discovery Server*) in Form eines von der OPC Foundation bereitgestellten Software-Prototypen im Leuze-Rechenzentrum auf einer virtuellen Maschine mit Windows-Server-Betriebssystem in Betrieb genommen.

Die Registrierung mehrerer in unterschiedlichen Gebäuden des Standorts Owen installierter Leuze-Sensoren ermöglichte den unkomplizierten Zugriff auf die verteilten Geräte von beliebigen *OPC UA Clients* an beliebigen Orten der globalen Netzwerkdomäne des Unternehmens.

Dem erfolgreichen Testaufbau für *Global Discovery* folgte die Integration eines GDS und eines *BCL 300i* auf dem für die Messe SPS IPC Drives 2017 in Nürnberg bestimmten Messeaufbau von Leuze electronic.

[1] OPC Foundation (Hrsg.): OPC Unified Architecture Specification Part 5: Information Model, Version 1.03, Release am 20.07.2015.

[2] OPC Foundation (Hrsg.): OPC Unified Architecture Specification Part 12: Discovery, Version 1.03, Release am 20.07.2015.

[3] Leuze electronic GmbH + Co. KG (Hrsg.): Produktinformationen zum Sensor BCL 300i, www.leuze.de, Stand 20.11.2017.

Bildquellen:

- Abbildung 1, 4: Leuze electronic, interne Grafikdatenbank
- Abbildung 2, 3: Eigene Darstellung

Evaluation of Time Responses from Video and Audio Transfer Paths in Automotive Systems

Mark Marszal*, Walter Lindermeir, Reinhard Keller

Faculty of Information Technology of the Hochschule Esslingen – University of Applied Sciences

Winter Semester 2017/2018

Motivation

Camera systems are used in data measurement acquisition within prototype systems during vehicle development to analyze the driving situation in order to facilitate a subjective assessment of the current situation in addition to the measurement data on the basis of signals. For legal purposes, the complete data measurement chain, particularly the time response, needs to be evaluated.

Such mechanisms already exist for today's signal based measurement values within bus systems like CAN, FlexRay and Ethernet. However, no such solution exists in order to determine the delay and the respective timing deviations within camera based systems, specifically the delay caused by IP/USB camera transfer paths, as well as the compression schemes used by such devices. In addition to the absent timing determination of image data, the same set of problems exists within the audio data domain.

Proposed solution

In order to determine the time response in both audio and video systems, a reference platform needs to be adopted within the system. In this Bachelor's Thesis, the Controller Area Network (CAN) bus system was used as a reference platform, due to the system's capability of meeting deterministic timing constraints [1]. With a stable reference platform in place, identical information needs to be encoded for each medium, and then transferred over all three physical mediums. Decoding and verification of the received analogue data then takes place on the receiver side, followed by a time response measurement in comparison to the received CAN data. With this timing information, the delay and jitter within the system can be determined.

Hardware and choice of measurement data

A PIC18F26K80 microcontroller serves as the basis for the CAN, Video and Audio data gen-

eration under realtime constraints. Such a microcontroller was chosen due to its simple Harvard RISC architecture, leading to its deterministic behaviour. The microcontroller generates black and white images using the International Telecommunication Union (ITU) System I signal standard. ITU System I defines an image consisting of 625 horizontal lines, which consist of two fields containing half images that are interlaced together to define a full picture [2]. Each line of video is $64\mu\text{s}$ long, leading to the microcontroller utilizing a $64\mu\text{s}$ interrupt as the basis of data generation. Within each image, that is during every 625 lines, CAN data and audio information is also generated.

The CAN information is sent during the generation of the first image line, while audio information is generated every 20 lines over a period of 305 lines starting at the second line of the image. Any small time deviation results in a television, or video capture device losing synchronization, resulting in either no image, or a very disfigured image being received. Figure 1 shows the hardware implementation of the thesis.



Figure 1: Hardware Setup with the barcode displaying the number 15 on the mini television

Considering analogue video signals have an industry defined standard, the amount of information to be sent across the audio path is limited to the timing requirements for video generation defined by the ITU. Each field within an image is 305 lines long, which results in a

*This thesis was conducted at the company XKrug GmbH, Stuttgart-Vaihingen

field timing length of $19.52ms$. Each field is followed by a vertical image synchronization before the next set of 305 lines are displayed. Due to the vertical synchronization signal complexity, this thesis further limits the audio data path to the first field time length of an image. Therefore any encoded picture information must be able to be encoded and transferred within the same field timing of $19.52ms$. It was decided that a very simple barcode depicting a overflowing counter should be the basis for the video signal generation. The Pharmacode standard offers the ability to depict any integer value from 3–131070 using a single track barcode with two varying black line thicknesses [3]. Along with the depicted hardware setup, Figure 1 shows the number 15 encoded as a barcode within the mini television.

For the audio signal data path, it was decided to use a method of amplitude modulation, specifically On–Off keying (OOK) in order to transfer digital information over sound waves. With OOK, the carrier wave is simply turned on and off to denote the presence of a binary one or a zero. Through empirical measurements, it was determined that five bits of digital information could be reliably transferred over the field timing length of $19.52ms$. Of these five bits, one bit has a unique signal form factor, and is used as a Start of Number marker. Therefore the amount of data that can be sent is limited to 4 bits, or 16 different values. Due

to the Pharmacode only being able to display values starting from three, the entire system is limited to transferring a counter ranging from numbers 3 to 15 on each physical medium. Figure 2 shows the number 15 in raw audio form, followed by a bandpass filtering of the signal. Next comes the envelope detection, following up with the signal demodulation process as shown in Figure 3. The time duration can then be measured in order to reconstruct the original number 15.

Offline data measurement

For the data recording of all three physical mediums, a standard PC was used to carry out the data acquisition. The Automotive Data and Time–Triggered Framework (ADTF) was the developer software used to reliably record the information, and assign timestamp information to each sample received from each data path [4]. Once the data was recorded, a set of filters were designed in order to filter and demodulate the recorded audio data, while OpenCV was used to perform image processing on each image to aid in the decoding of each barcode image from the microcontroller. Once the image and audio information has been successfully decoded, a comparison of the delay in each system can then be derived from our reference CAN data, therefore solving the problem of determining the absent timing information within video and audio data paths.

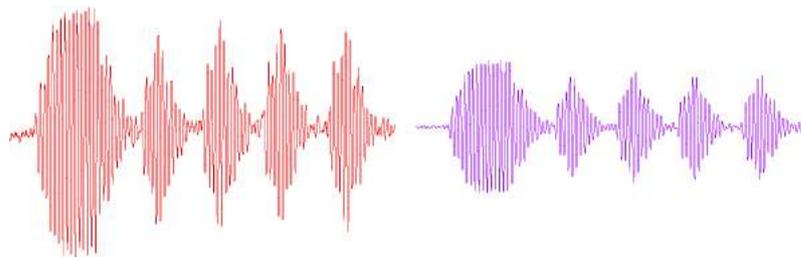


Figure 2: Raw audio signal passing through a bandpass filter



Figure 3: Filtered signal passing through an envelope detector, following with the demodulating process

[1] <http://www.ni.com/white-paper/2732/en/>

[2] https://www.itu.int/dms_pubrec/itu-r/rec/bt/R-REC-BT.470-6-199811-S!!PDF-E.pdf

[3] http://www.gomaro.ch/ftproot/Laetus_PHARMA-CODE.pdf

[4] <https://www.elektrobit.com/products/eb-assist/adtf/>

Figure sources:

- Figure 1–3: Author's own Figure

Digitalisierung und Optimierung der Einsatzplanung Vor-Ort Service

Vanessa Orendi*, Thomas Rodach, Catharina Kriegbaum-Kling

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

“Es ist nicht die stärkste Spezies, die überlebt, auch nicht die intelligenteste, sondern diejenige, die am besten auf Veränderungen reagiert.“

– Charles Robert Darwin

Es sind verschiedene Kräfte, die den Kampf um den Markt zunehmend beeinflussen: das veränderte Kundenverhalten, der Eintritt neuer Konkurrenten in den Markt, welche sinkende Margen und Umsätze im Produktgeschäft mit sich bringt, sowie die rapiden Einführungen neuer Technologien [1]. Schon Darwin hat erkannt, dass es wesentlich ist auf die Umwelt zu reagieren und sich selbst in Einklang zu bringen. Insbesondere Unternehmen werden dabei vor eine große Herausforderung gestellt. Heutzutage ist es die Digitalisierung und die damit einhergehende Veränderung, der es viel Aufmerksamkeit bedarf. Das Bundesministerium für Wirtschaft und Energie (kurz: BMWI) definiert die Digitalisierung als „die umfassende Vernetzung aller Bereiche von Wirtschaft und Gesellschaft sowie die Fähigkeit, relevante Informationen zu sammeln, zu analysieren und in Handlungen umzusetzen. Die Veränderungen bringen Vorteile und Chancen, aber sie schaffen auch ganz neue Herausforderungen“ [2].

Für die Unternehmen ist es im Zuge der Digitalisierung von hoher Bedeutung die bestehenden Prozesse anzupassen und intelligente IT Tools zur Unterstützung zu nutzen. Auch der technische Kundendienst bei der Festo Vertrieb GmbH & Co. KG in Esslingen ist von dieser Veränderung betroffen. Aus diesem Grund handelt die Bachelorthesis von einer Digitalisierung und Optimierung eines Ablaufprozesses im Kundendienst, genauer gesagt von dem Prozess „Einsatzplanung Vor-Ort Service“. Die Einsatzplanung Vor-Ort Service beinhaltet die Planung von Serviceeinsätzen der Servicetechniker beim Kunden. Dieser Planungsprozess ist aktuell intransparent und mit einer hohen Papierverschwendung verbunden. Aus diesem Grund liegt der Fokus im Laufe dieser Arbeit darin, eine Transparenz über die einzelnen Vorgänge mitarbeiterübergreifend

zu gestalten und gleichzeitig der Umwelt zugute das Verwenden von Formularen in Papierform zu reduzieren. Das Team im technischen Kundendienst, welches die Serviceeinsätze der Servicetechniker plant, wünscht sich aus den genannten Gründen das Abschaffen von Papierchecklisten und diverser analoger Arbeitsmappen um die Digitalisierung voranzutreiben. Dabei ist es wichtig, dass keine eingegangenen Informationen verloren gehen und der aktuelle Status der Serviceeinsatzplanung einsehbar ist.

Das erste definierte Ziel dieser Bachelorthesis ist es, die Checkliste in digitaler Form umzusetzen, sodass mindestens die Mitarbeiter des vierköpfigen Teams der Serviceeinsatzplanung diese nutzen können. Zu Beginn der Ist-Analyse stellte sich sehr früh heraus, dass ein größerer Mehrwert für das Team geschaffen wird, wenn eine digitale Übersicht über alle aktuellen Serviceeinsätze verfügbar wäre. Diese Übersicht beinhaltet ebenso die notwendigen Statusmeldungen der Checkliste. Folglich besteht das Projekt der Bachelorthesis darin, eine digitale Auflistung für den Arbeitsvorrat der Mitarbeiter zu erstellen. Ebenfalls erhofft man sich mögliche Prozessoptimierungen. Die Herausforderung besteht insbesondere darin, die eingesetzten Tools entsprechend anzupassen.

Zum vertieften Verständnis des Ablaufs der Einsatzplanung gilt es zunächst den Serviceprozess zu durchleuchten. Dafür ist es notwendig mit Einsatz verschiedener Methoden den Prozessablauf abzubilden.

Der erste Istzustand des Serviceeinsatzplanungsprozesses stellt sich wie folgt dar: Nach Eingang einer Serviceanfrage in das interne Ticketsystem entnimmt ein Mitarbeiter eine Mappe inklusive einer Checkliste, die zur Serviceeinsatzplanung abgearbeitet werden muss. Diese Checkliste unterstützt die Mitarbeiter darin, alle notwendigen Schritte durchzuführen und gewährleistet gleichzeitig die Sichtbarkeit von erledigten und offenen Punkten für einen Teamkollegen in Vertretung. Weitere enthaltene Dokumente und Informationen

*Diese Arbeit wurde durchgeführt bei der Firma Festo Vertrieb GmbH & Co. KG, Denkendorf

sind unter anderem die Servicebestellung, Informationen über notwendiges oder bestelltes Material, gegebenenfalls ein Schaltplan der Festo-Anwendung vor Ort etc. (s. auch Abbildung 1). Betrachtet man diese Aufzählung im Detail wird deutlich, dass diese Informationen zumeist digital übermittelt wurden und generell keine Notwendigkeit besteht, diese

auszudrucken. Ergo müssen die Mitarbeiter darauf vorbereitet werden, dass während des Projekts und der Umsetzung der gegenwärtige Ablauf der Serviceeinsatzplanung geändert werden muss.

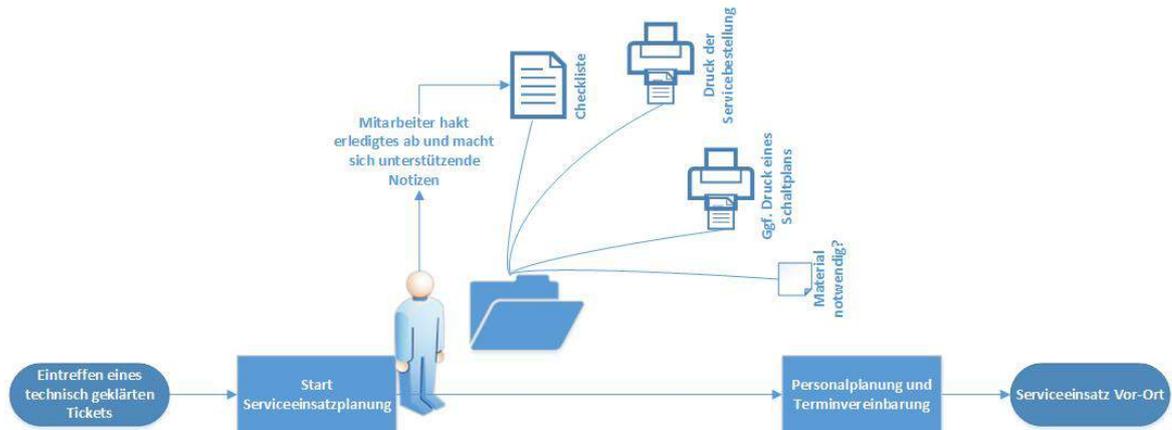


Abbildung 1: Stark vereinfacht dargestellte Serviceeinsatzplanung mit Fokus auf die Digitalisierung

Nachdem der Entschluss gefällt wurde, dass eine digitale Checkliste die Produktivität nur in geringem Maß steigert, sollte der optimale Sollzustand zum Abschluss der Bachelorthesis folgendermaßen aussehen: eine digitale Arbeitsvorratsübersicht liegt vor, die den Mitarbeitern zur Unterstützung der Serviceeinsatzplanung dient. Diese Übersicht ist für jeden Mitarbeiter des vierköpfigen Teams einsehbar und enthält alle laufenden Planungen der Serviceeinsätze, sowie deren zugehörigen Planungsstatus. Zudem ist erkennbar, welche Schritte nachfolgend noch zu erledigen sind. Der Zugang zu der Übersicht ist insbesondere dann wichtig, wenn Kunden und andere Mitarbeiter der gesamten Serviceabteilung (telefonisch) Details oder Statusinformationen erfragen.

Problemlösung

Zur Identifikation der notwendigen Informationen, die auf der Vorgangsliste enthalten sein müssen, wird zunächst ein Testlauf mit den Mitarbeitern durchgeführt. Dabei wird vollständig auf das Anlegen der analogen Mappen verzichtet. Unterstützende Dokumente gibt es keine. Demzufolge ist jedem Mitarbeiter die Planungsorganisation selbstüberlassen. Lediglich zur Dokumentation der Anzahl der Serviceeinsätze, potenzielle Informationen, für die eine Mappe angelegt worden wäre, und Kundenfragen, wird ein Dokument zur Verfügung gestellt. Dieses bildet die Basis für die Umsetzung des Projekts.

[1] Vgl. Hoong, V. (2013). Disruptive digital forces. (Deloitte, Hrsg.) The digital transformation of customer services. Our point of view. S. 6

[2] BMWI (Bundesministerium für Wirtschaft und Energie) (2015). Digitale Souveränität: Schlüssel für Wachstum und Beschäftigung in der digitalen Welt. (BMWi, Hrsg.) Industrie 4.0 und Digitale Wirtschaft – Impulse für Wachstum, Beschäftigung und Innovation. S. 3

Bildquellen:

- Abbildung 1: Eigene Darstellung

Konzepterstellung zur Treiberentwicklung für den Linux-Kernel

Gianluca Panetta*, Walter Lindermeir, Reinhard Keller

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Motivation

Aufgrund stetig steigender Anforderungen an elektronische Systeme in der Industrie und im Automobil werden die eingesetzten Steuerungen immer leistungsfähiger hinsichtlich Software und Hardware. Deshalb werden hier zunehmend leistungsfähigere und komplexere Betriebssysteme, wie Linux oder Windows eingesetzt. Diese Systeme bringen einen enormen Vorteil hinsichtlich Modularität, Parallelität, Datensicherheit und Konnektivität.

Damit eigene Hardware mit diesen Betriebssystemen vertrieben werden kann, müssen dazu eigene Treiber entwickelt werden. Da Linux im Bereich der eingebetteten Systeme ein großes Wachstum erfährt, wird in dieser Arbeit die Treiberentwicklung für dieses Betriebssystem näher betrachtet.

Rahmenbedingungen

Damit ein Linux-Treiber im Zusammenspiel mit dem Rest des Betriebssystems funktioniert, müssen Standards eingehalten werden. Diese Standards fordern unter anderem, dass jeder Treiber eine Initialisierungs- und eine Deinitialisierungsroutine besitzt, eine Anmeldung dieser Routinen an den Kernel durchführt und eine korrekte Fehlerbehandlung anbietet. Für Treiber, die eine Schnittstelle zur Anwendungsebene, dem sogenannten User Space (siehe Abbildung 1), benötigen, muss eine Anbindung an diese, über eine Kommunikations-API implementiert werden. Die Kommunikation mit der Anwendungsebene erfolgt somit ausschließlich über die angebundene Kommunikations-API.

Für den Datenaustausch zwischen den zwei Ebenen stellt der Kernel die Funktionen `copy_to_user()` und `copy_from_user()` bereit. Diese Befehle sind notwendig, da zum einen Programme der Anwendungsebene keinen Zugriff auf die Kernebene haben und zum anderen mit virtuellen Adressen anstatt physikalischen Adressen arbeiten. Aus diesem Grund darf hier auf keinen Fall mit direkten Kopier-Befehlen, wie `memcpy()`, gearbeitet werden.

Bei der Entwicklung des Treibers muss stets beachtet werden, dass der Code nebenläufig ist, was bedeutet, dass er von mehr als einem Prozessor ausgeführt und zu jeder Zeit unterbrochen werden kann. Dies erfordert eine hohe Sorgfalt bei der Konzepterstellung und Programmierung des Treibers. So müssen zum Beispiel kritische Abschnitte, bei denen mehrere Zeilen Code in jedem Fall hintereinander ununterbrochen ausgeführt werden müssen, als solche kenntlich gemacht werden [1].

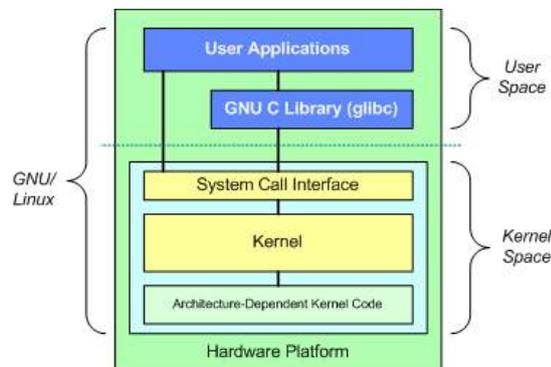


Abbildung 1: Architektur eines Linux-Betriebssystems

Lizenzierung

Auch bei der Lizenzierung des Treibers muss aus Entwicklersicht einiges beachtet werden. Linux bezeichnet genau genommen nicht das Betriebssystem, auf dem eine bestimmte Anwendung ausgeführt wird, sondern nur den Kern des Betriebssystems, auch Kernel genannt. Dieser Kernel steht unter der GNU General Public License (GPL), welche dem Nutzer gestattet den Quellcode einzusehen und ihn Berechtigten, diesen zu modifizieren und diese Modifikation anschließend weiter zu verteilen oder zu verkaufen.

Trotz dieser weitreichenden Rechte handelt es sich bei der GPL um eine sehr restriktive Open-Source-Lizenz, da sie fordert, dass alle abgeleiteten Werke des unter ihr stehenden Produkts, ebenfalls unter der GPL veröffentlicht werden müssen [2].

*Diese Arbeit wurde durchgeführt bei der Firma Vector Informatik GmbH, Weilimdorf

Aufgrund der monolithischen Architektur von Linux ist ein Gerätetreiber, welcher speziell für Linux geschrieben wurde, laut Linux-Erfinder Linus Torvalds als abgeleitetes Werk zu betrachten und unter der GPL zu veröffentlichen. Ein Kernelmodul, das für ein anderes Betriebssystem, wie zum Beispiel UNIX, entwickelt und für Linux portiert wurde, könne allerdings nicht als abgeleitetes Werk betrachtet werden und dürfe unter seiner ursprünglichen Lizenz veröffentlicht werden [3].

Eine Ausnahme bilden Gerätetreiber, die auf der Anwendungsebene ausgeführt werden. Anwendungsprogramme, die auf Linux-Betriebssystemen laufen, werden nicht als abgeleitetes Werk betrachtet und sind somit frei in der Lizenzgebung.

Treibermodelle

Um einen Treiber in den Linux-Kernel einzubinden, gibt es zwei Möglichkeiten. Zum einen kann der Treiber direkt beim Erstellen des Kernels in diesen einkompiliert werden und zum anderen kann er als Kernelmodul erstellt werden. Wenn der Treiber in den Kernel mit einkompiliert wird, hat das den Vorteil, dass der Treiber beim Bootvorgang direkt nach dem Laden des Kernels startet, während das Modul dynamisch zur Laufzeit ge- und entladen werden kann, was vor allem bei Hot-Plug-fähigen Geräten von Vorteil sein kann. Damit wird ein unnötiges Aufblähen des Kernels verhindert. Wenn ein Kernelmodul erstellt wird, ist darauf zu achten, dass das Modul nur garantiert mit der Linux-Version funktioniert, dessen Header-Dateien beim Kompilieren des Moduls mit eingebunden wurden.

Bei der Wahl eines Treibermodells gibt es verschiedene Möglichkeiten zur Auswahl hinsichtlich Anbindung an die Anwendungsebene und Kernfunktionalität. Zur Kernfunk-

tionalität existieren diverse Subsysteme, wie Character-Devices, Block-Devices, Netzwerk, SCSI, PCI, SPI, I2C und viele mehr. Für den Zugriff aus der Anwendungsebene existieren verschiedene Applikationsschnittstellen, wie das Standard-API, das Kommunikations-API oder das Multimedia-Interface [6].

Das Character-Device ist das am häufigsten verwendete Treibermodell. Es basiert auf einfacher zeichenweiser Ein- und Ausgabe. Im Dateisystem wird eine Gerätedatei erstellt, über die auf das Gerät mittels der Standard-API durch Lese- und Schreibzugriffe (read() und write()) zugegriffen werden kann. Das Block-Device basiert ebenfalls auf Ein- und Ausgabe, wird auch über Dateizugriffe gesteuert, liest und schreibt jedoch blockweise, anstatt zeichenweise. Netzwerkgeräte unter der Verwendung der Kommunikations-API werden hingegen nicht als Datei dargestellt. Sie erscheinen im System als Netzwerkinterface, das mittels des Programmes ifconfig einsehbar ist.

Für Treiber, die auf der Anwendungsebene ausgeführt werden sollen, gibt es seit Linux 2.6.23 das sogenannte Userspace I/O-Framework. Dieses Framework ist gut geeignet für spezielle Hardware, die sich nicht den üblichen Treibersubsystemen, wie USB, Netzwerk oder PCI, zuordnen lässt. Darunter fallen zum Beispiel industrielle IO-Karten oder Feldbus-schnittstellen [4]. Wichtig ist, dass die Hardware Speicher hat, der gemappt werden kann und sich die Hardware ausschließlich über Zugriffe auf diesen steuern lässt. Dazu wird ein Kernel-Modul minimaler Größe geschrieben, welches den Speicher mappt, alle Zugriffe darauf verarbeitet und der Anwendungsebene ein Character Device zur Verfügung stellt, auf das der eigentliche Treiber als Benutzerprogramm zugreifen kann [5]. Diese Architektur ist in Abbildung 2 dargestellt.

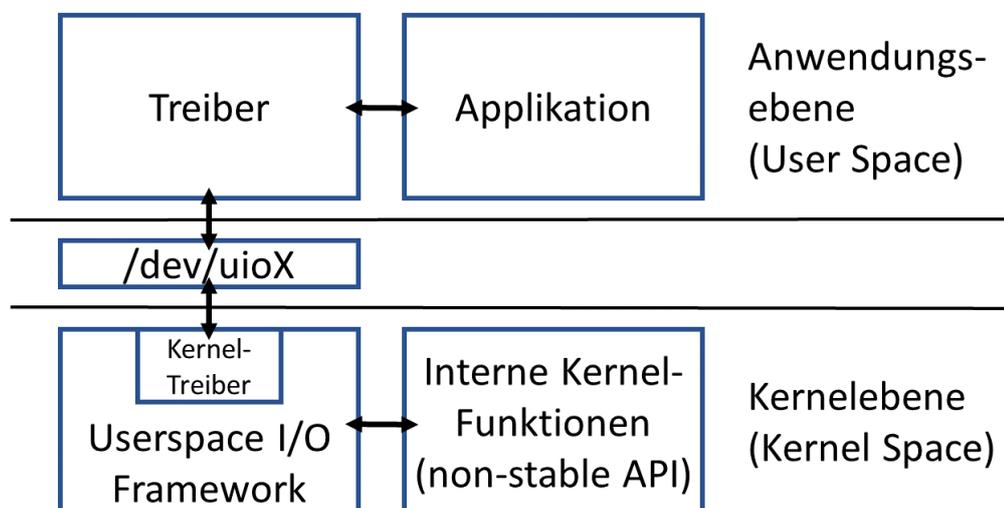


Abbildung 2: Architektur eines Userspace I/O Treibers

Debugging

Zum Debuggen eines Linux-Treibers hat der Entwickler ebenfalls unterschiedliche Mittel zur Verfügung, die sich jedoch von der klassischen Vorgehensweise zum Debuggen eines Anwendungsprogrammes unterscheiden. So gibt es keine Möglichkeit, den Code eines Treibers mit einem klassischen Software-Debugger anzuhalten oder Schritt für Schritt durchzugehen, da die Programme der Anwendungsebene keinen Zugriff auf die Kernebene haben.

Stattdessen wird meist auf Debugging via Datenausgabe zurückgegriffen. Dazu stellt der Linux-Kernel den Befehl `printk` bereit, mit dem es möglich ist, Nachrichten auf den Kernel-Log auszugeben. Mit dem Befehl `dmesg` können diese Nachrichten anschließend über die Kommandozeile angezeigt werden. Dadurch ist es möglich, durch einfaches Ausgeben von Daten interne Zustände zu beobachten.

Eine weitere Methode besteht in der Ausgabe von Daten über das `/proc`-Dateisystem. In `/proc` können durch den Kernel beliebig Dateien und Ordner erzeugt werden, die aus der Anwendungsebene mit herkömmlichen Programmen ausgelesen werden können. Die Ausgabe über das `/proc`-Dateisystem hat den Vorteil, dass eine Ausgabe erst dann erfolgt, wenn von der jeweiligen Datei innerhalb des `/proc`-Dateisystems gelesen wird, was weniger Performanceeinbußen zur Folge hat, als die Ausgabe über `printk`.

Eine weitere Möglichkeit einen Treiber zu testen, der nicht oder noch nicht auf Hardware zugreifen muss, ist User Mode Linux. Dabei wird ein Linux-Kernel als Anwendungsprozess mit Benutzerrechten auf einem laufenden Linux-System ausgeführt und kann deshalb von Debugging-Tools, wie `gdb`, angesteuert werden.

Fazit

Ziel der Arbeit ist die Analyse der Herangehensweise zur Entwicklung eines Treibers für das Betriebssystem Linux. Dabei sollte vor allem auf die Wahl des Treibermodells, die Interaktion zwischen Anwendungs- und Kernebene, das Debugging und die lizenztechnischen Bedingungen eingegangen werden.

Hinsichtlich des Treibermodells kann je nach Anbindung und Art der Hardware zwischen verschiedenen Subsystemen und Kommunikations-APIs gewählt werden, zum Datenaustausch zwischen Kernel- und Anwendungsebene muss beachtet werden, dass dies nur über bestimmte Befehle geschehen darf, das Debuggen eines Treibers erfolgt hauptsächlich durch Ausgabe von Daten und hinsichtlich der Lizenz muss bei einem Kernel-Treiber die GPL beachtet werden, sofern der Herausgeber sich nicht in einer Grauzone befinden will.

-
- [1] CORBET, Jonathan, RUBINI, Alessandro, KROAH-HARMAN, Greg, 2005. Linux Device Drivers, Third Edition. Sebastopol, USA: O'Reilly. ISBN 0-596-00590-3
- [2] FREE SOFTWARE FOUNDATION, INC., 2007. GNU General Public License. Verfügbar unter: <http://www.gnu.org/licenses/gpl.html>
- [3] TORVALDS, Linus, 2003. Re: Linux GPL and binary module exception clause? [E-Mail]. 04.12.2003, 00:03:11 GMT. Verfügbar unter: http://yarchive.net/comp/linux/gpl_modules.html
- [4] KOCH, Hans-Jürgen, [kein Datum]. Userspace I/O drivers in a realtime context. Uhldingen: Linutronix GmbH.
- [5] KOCH, Hans-Jürgen, 2006. The Userspace I/O HOWTO. Uhldingen: Linutronix GmbH, 12.11.2006 [Zugriff 12.11.2017]. Verfügbar unter: www.kernel.org/doc/html/v4.12/driver-api/uio-howto.html
- [6] QUADE, Jürgen, KUNST, Eva-Katharina, 2016. Linux-Treiber entwickeln: Eine systematische Einführung in die Gerätetreiber- und Kernelprogrammierung. 4., aktualisierte und erweiterte Auflage. Heidelberg: dpunkt.verlag. ISBN 978-3-86490-288-8
- [7] JONES, M., 2007. Anatomy of the Linux Kernel. Armonk, USA: IBM, 06.06.2007 [Zugriff 12.11.2017]. Verfügbar unter: www.ibm.com/developerworks/linux/library/l-linux-kernel

Bildquellen:

- Abbildung 1: [7]
- Abbildung 2: Eigene Abbildung nach [4]

Entwicklung einer mobilen, kamerabasierten Zweipunktstrahlerdetektion und -vermessung zur Lagebestimmung

Patrick Pawlowski*, Reiner Marchthaler, Werner Zimmermann

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Einleitung

Durch die steigende Verbreitung von Smartphones eröffnen sich stetig neue Möglichkeiten bildverarbeitende Algorithmen und Augmented Reality (AR) in das alltägliche Leben einzubinden. Zudem sorgt die steigende Leistung der Smartphoneprozessoren in den letzten Jahren dafür, dass diese Algorithmen auf mobilen Geräten echtzeitfähiger werden. Mithilfe bekannter Eigenschaften des Objektivs sowie des Kamerasensors lassen sich Aufgaben, welche für einen Menschen nicht oder nur zum Teil lösbar sind, auf dem Smartphone bearbeiten. Beispiele hierfür sind die Bestimmung von Größen oder Farbwerten bestimmter Objekte. Eine Problemstellung welche für Algorithmen schwieriger lösbar ist als für Menschen, ist die Klassifizierung von Objekten in Bildern. Fortschritte in diesem Bereich gab es in dem letzten Jahren vor allem durch die effiziente Nutzung von Neuronalen Netzen. Im Verlauf der Bachelorarbeit soll ein System entwickelt werden, welches über die Smartphonekamera Lichtquellen um Bild detektiert, diese einem Objekt zuordnet und somit den Abstand und die Lage des Smartphones zu diesem Objekt bestimmen kann.

Grundlagen

Um Lichtquellen in einem Bild zu detektieren muss das Bild segmentiert werden. Die Segmentierung von Bilddaten ist einer der wichtigsten Schritte der digitalen Bildverarbeitung. Diese hat zum Ziel Teile der Bildinformation welche eine starke Korrelation mit den gesuchten Objekten besitzen zu finden und zu extrahieren und dabei den Hintergrund auszublenden. Hierbei steht die Bildsegmentierung im Ablauf vor der Bildanalyse, nach der Segmentierung soll gesagt werden können welche einzelnen Pixel des Bildes zu den gesuchten Objekten gehört. Die Segmentierung lässt sich einteilen in Schwellwertsegmentierung, Kantensegmentierung und Regionensegmentierung [1]. Für Lichtquellen eignet sich speziell eine Schwellwertsegmentierung nach den Helligkeitsinformationen im Bild da Lichtquellen

meist überblendet dargestellt werden [1].

Der Segmentierung folgt eine Filterung der detektierten Lichtquellen. Hierzu werden Eigenschaften der elliptischen Form der gesuchten Lichtquellen genutzt. Hierdurch können Spiegelungen und fremde Lichtquellen entfernt werden, zudem soll dabei ein Lichtquellenpaar gefunden werden bei dem die größtmögliche Überlappung an Ellipseneigenschaften gilt [2].

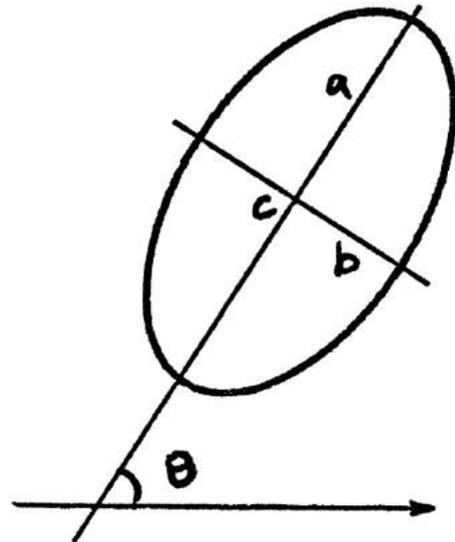


Abbildung 1: Ellipsen Eigenschaften

Als weiterer Anhaltspunkt gilt die Lage der Lichtquellen im Bild, da sich die gesuchten Lichtquellen auf einer Ebene parallel zum Boden befinden. Diese Filterung wird durch die Inertialsensorik des Smartphones unterstützt und filtert alleinstehende Lichtquellen aus. Ein Modell einer Perspective Camera wird genutzt um Koordinaten der zweidimensionalen Bildebene auf den dreidimensionalen realen Raum umzurechnen. Hierbei werden die Punkte mithilfe einer Projektionsmatrix in den gewünschten Raum transformiert [3].

*Diese Arbeit wurde durchgeführt bei der Firma Daimler AG, Sindelfingen

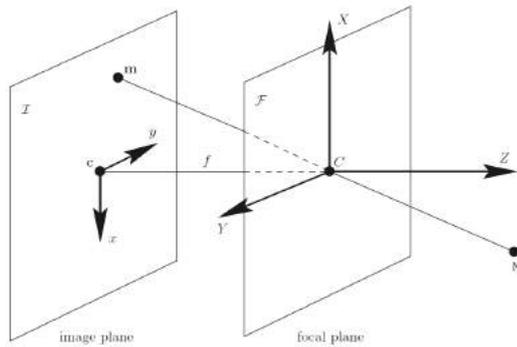


Abbildung 2: Lochkamera Modell

Um die Laufzeit der Detektion zu verringern wird nach einer erfolgreichen Bestimmung des Abstandes der Bildbereich auf die Position der Lichtquellen eingeschränkt. Werden bei einem Folgedurchlauf keine Lichtquellen in diesem Bildausschnitt detektiert wird dieser Schrittweise vergrößert bis eine Detektion erfolgreich ist.

Als Erweiterung des Systems wurde zudem die Verwendung eines Convolutional Neural Networks (CNN) untersucht. Dieses schränkt bereits vor der ersten Detektion von Lichtquellen den Bildbereich durch Klassifizierung ein und spart somit Rechenleistung bei der Detektion. Hierzu wurde die mobile Variante der Tensorflow API für Android und ein vortrainiertes Netz zur Klassifikation von verschiedenen Objekten genutzt.

Programmablauf

Als Testobjekt wurden zwei LEDs an einem dem Klassifikator bekannten Objekt in einem definierten Abstand befestigt. Im normalen Programmablauf wird der gesamte Bildbereich der Kamera als erstes segmentiert und nach dem bestmöglichen Lichtquellenpaar gesucht

welche zudem die Eigenschaften für die gesuchten Lichtquellen besitzen. Durch die Koordinaten und elliptischen Eigenschaften der Lichtquellen sowie der kameraspezifischen Daten wird im folgenden Schritt ein genauer Abstand sowie Winkel zum Objekt bestimmt und der Bildbereich für die nächste Detektion eingeschränkt.

Im erweiterten Programmablauf wird der Bildbereich im Vorfeld durch das Neuronale Netz eingeschränkt. Dieser Bildbereich wird im Bild verfolgt um die Dauer der Klassifizierung zu kompensieren. Nach der Bildbereichseinschränkung wird der Programmablauf normal fortgesetzt aber keine Bildbereichseinschränkung nach dem erfolgreichen Bestimmen des Abstandes vorgenommen.

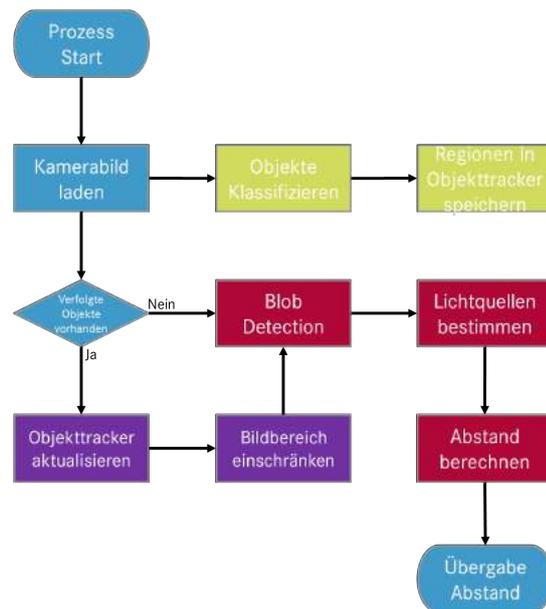


Abbildung 3: Programmablauf

[1] Bernd Jähne. Digitale Bildverarbeitung. Springer-Verlag, 2013.

[2] Tim Ellis, Ahmed Abbood, and Beatrice Brillault. Ellipse Detection and Matching with Uncertainty, pages 136–144. Springer London, London, 1991.

[3] Zhengyou Zhang. Computer Vision: A Reference Guide. Springer US, Boston, MA, 2014

Bildquellen:

- Abbildung 1: [2, S.139]
- Abbildung 2: [3, S.591]
- Abbildung 3: Eigene Abbildung

Entwicklung einer Software für den Betrieb eines Radardemonstrators und Entwurf eines digitalen Filters zur Bildstabilisierung

Patrik Reiske*, Karlheinz Höfer, Reiner Marchthaler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Motivation

Ziel dieser Bachelorarbeit war die Entwicklung eines ersten prototypischen Radardemonstrators auf Basis einer Evaluationshardware. Diese Entwicklung sollte neben einer Software für den Betrieb der verwendeten Hardware auch den Entwurf eines digitalen Filters zur Bildstabilisierung umfassen.

Elektromagnetische Wellen

Allgemein können elektromagnetische Felder, die als periodische Funktionen von Zeit und Raum beschrieben werden, als elektromagnetische Wellen oder Überlagerung elektromagnetischer Wellen angenommen werden. Diese elektromagnetischen Wellen breiten sich mit Lichtgeschwindigkeit aus und werden dabei von metallischen und dielektrischen Materialien reflektiert. Wie stark die elektromagnetischen Wellen von einem solchen Material reflektiert werden hängt dabei von dessen elektrischen und magnetischen Materialkonstanten ab [1, 2].

Ebenso wie andere Wellen unterliegen auch elektromagnetische Wellen dem Dopplereffekt. Dieser beschreibt den Zusammenhang der Frequenzverschiebung zwischen Sender und Empfänger einer Welle in Abhängigkeit von der relativen Radialgeschwindigkeit zwischen diesen [3].

Radar

Unabhängig vom verwendeten Radarverfahren gliedert sich der Messzyklus einer industriellen Radaranwendung stets in die folgenden fünf Phasen [3]:

1. Das Radarsignal wird entsprechend des verwendeten Radarverfahrens erzeugt und in Form einer elektromagnetischen Welle abgestrahlt.
2. Die elektromagnetische Welle breitet sich mit Lichtgeschwindigkeit vom Radarsensor aus.
3. Metallische und dielektrische Materialien reflektieren die elektromagnetische Wel-

le entsprechend ihrer elektrischen und magnetischen Materialeigenschaften.

4. Ein Teil der rückgestreuten elektromagnetischen Welle breitet sich in Richtung des Radarsensors aus.
5. Der Radarsensor wertet die empfangene elektromagnetische Welle entsprechend des verwendeten Radarverfahrens aus.

Dem frequenzmodulierten Dauerstrichradar (engl. „frequency modulated continuous wave radar“) liegt dabei ein Chirp-Signal zugrunde. Ein solches Chirp-Signal ermöglicht sowohl eine genaue Messung der Entfernung als auch der relativen Radialgeschwindigkeit von Objekten zum Sender [3].

Kalman-Filter

Das Kalman-Filter ist ein Verfahren zur Zustandsschätzung zeitdiskreter linearer Systeme auf Basis verrauschter und teils redundanter Messungen. Abbildung 1 veranschaulicht eine derartige Schätzung eines verrauschten gemessenen Signals [4].

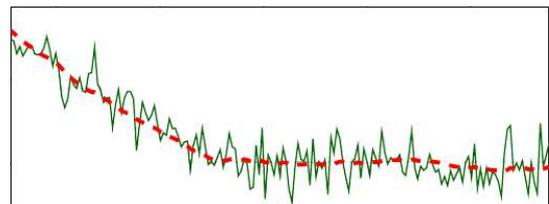


Abbildung 1: Beispielhafte Gegenüberstellung des Verlaufs einer verrauschten gemessenen Zustandsgröße und einer mittels Kalman-Filter korrigierten geschätzten Zustandsgröße.

Grundlage für die Zustandsschätzung ist die Beschreibung des zu schätzenden Systems im Zustandsraum, wie in Abbildung 2 dargestellt. Die Zustandsschätzung eines solchen Systems mittels eines Kalman-Filters ist jedoch nur unter der Voraussetzung möglich, dass seine zeitdiskretisierte Zustandsraumbeschreibung beobachtbar ist [4].

*Diese Arbeit wurde durchgeführt bei der Firma Balluff GmbH, Neuhausen auf den Fildern

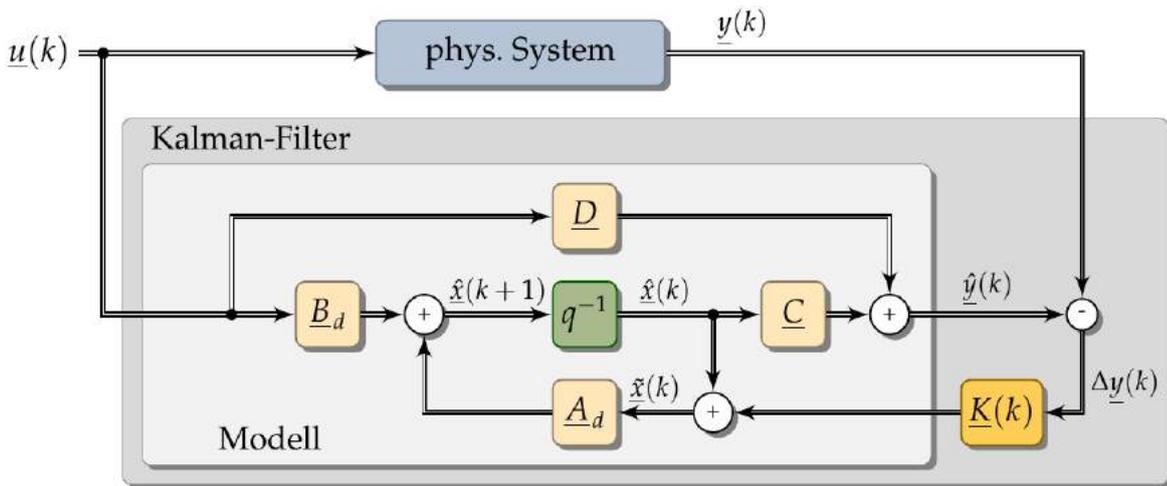


Abbildung 2: Allgemeine Darstellung eines Kalman-Filters bei vernachlässigtem System- und Messrauschen.

Evaluationshardware

Als Grundlage für die angestrebte Entwicklung wurde die IWR1443BOOST Evaluationshardware von Texas Instruments verwendet. Diese beinhaltet bereits die gesamte Radarfunktionalität und stellt die Messdaten in vorverarbeiteter Form zur Verfügung.

Softwareentwicklung

Da sich das verfügbare C/C++-Softwareentwicklungskit der Evaluationshardware nicht für die schnelle Prototypenentwicklung eignet, wurde zunächst eine Python-Programmierschnittstelle implementiert. Deren Funktionalität umfasst neben der Kommunikation mit der Evaluationshardware auch ein Modul zur Visualisierung der Messdaten.

Filterentwurf

Bevor das Filter zur Bildstabilisierung entworfen werden konnte, musste ein Beispielszenario für dessen Anwendung festgelegt werden.

Basierend auf diesem Szenario wurde das System zunächst im Zustandsraum beschrieben. Anschließend konnte die Beschreibung in den zeitdiskreten Zustandsraum übertragen und seine Beobachtbarkeit festgestellt werden. Auf Grundlage dieser beobachtbaren zeitdiskreten Zustandsraumbeschreibung des Systems war es möglich, das Kalman-Filter zu dimensionieren.

- [1] Roger F. Harrington. Time-Harmonic Electromagnetic Fields. The IEEE Press Series on Electromagnetic Wave Theory. New York: Wiley-IEEE Press, 2001. ISBN: 978-0-471-20806-8.
- [2] Uwe Siart. Formelsammlung für die Hochfrequenztechnik. Version 0.55. 8. Juni 2017. URL: <http://siart.de/lehre/hf-formeln.pdf> (besucht am 06. November 2017).
- [3] Jürgen Göbel. Radartechnik. Grundlagen und Anwendungen. Berlin: VDE VERLAG, 2001. ISBN: 3-8007-2582-7.
- [4] Reiner Marchthaler und Sebastian Dingler. Kalman-Filter. Einführung in die Zustandsschätzung und ihre Anwendung für eingebettete Systeme. Wiesbaden: Springer Vieweg, 2017. ISBN: 978-3-658-16727-1.

Bildquellen:

- Abbildungen 1, 2: Reiner Marchthaler und Sebastian Dingler. Kalman-Filter. Einführung in die Zustandsschätzung und ihre Anwendung für eingebettete Systeme. Wiesbaden: Springer Vieweg, 2017. ISBN: 978-3-658-16727-1.

Videobasierte Objekterkennung unter Verwendung von Bildverarbeitungsalgorithmen für autonom fahrende Modellfahrzeuge

Eugen Rudel*, Jürgen Koch, Reiner Marchthaler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Mit dem technologischen Wandel in der Automobilindustrie und der Forschung durch Unternehmen wie Google und Tesla, gewinnen autonome Fahrsysteme zunehmend an Bedeutung. Für eine selbstständige und vor allem kollisionsfreie Navigation durch dicht besiedelte Städte, ist eine Objekterkennung in Echtzeit zwingend erforderlich. Objekte sind im Kontext des autonomen Fahrens unter anderem weitere Verkehrsteilnehmer, Straßenschilder oder Ampeln. Neben Laserscannern, Radar- und Ultraschallsystemen, finden optische Sensoren in Form von Kameras Verwendung in Fahrassistenzsystem und im autonomen Fahren. Tesla Gründer Elon Musk setzt bei seinem Tesla Model S vor allem auf Kamerasysteme und verbaute acht Kameras, welche dem Fahrzeug das Sehen ermöglichen [1]. Die Erkennung von Objekten in von den Kameras aufgezeichneten Video-Streams erfolgt meist durch eine Abfolge von Bildverarbeitungsschritten, dargestellt in Abbildung 1.

Bildvorverarbeitung

Im ersten Schritt der Bildvorverarbeitung werden die Bilder aufbereitet. Es werden Störungen wie Rauschen oder Verzerrungen entfernt und gegebenenfalls wird das Ausgangsbild in einen anderen Farbraum konvertiert. Es wird bei der Bildvorverarbeitung zwischen lokalen, globalen und Punktoperatoren unterschieden

[2]. Globale Operatoren umfassen die Rotation, Translation und die Inversion eines Bildes. Punktoperatoren beziehen sich jeweils auf einzelne Pixel des Eingabebildes und werden bei Helligkeits- oder Kontrastkorrekturen verwendet. Lokale Operatoren, wie beispielsweise der Median-Filter, nutzen zusätzlich den Bereich um den Eingabepixel.

Segmentierung

Segmentierung ist im Prozess des maschinellen Sehens meist der erste Schritt der Bildanalyse. Dabei wird unter anderem das Schwellwert-Verfahren angewandt. Dieses Verfahren (im englischen thresholding) ist eine einfache Methode um Graustufenbilder zu Binärbildern umzuwandeln. Es wird für jedes Pixel eine Vergleichsoperation durchgeführt. Durch das Vergleichen des Grauwerts mit einem Schwellwert wird entschieden, ob das Pixel weiß oder schwarz wird. Ein weiteres Verfahren ist das Multiband Thresholding, welches es ermöglicht nach Farben zu segmentieren. Die Grenzwerte sind hier je nach Farbraum durch mehrere Parameter beschrieben. Im RGB-Farbraum wird nach Minimum und Maximum eines Farbanteils gefiltert. Die jeweiligen Farbanteile werden in diesem Verfahren separat behandelt und im Anschluss mit dem logischen UND-Operator verbunden.

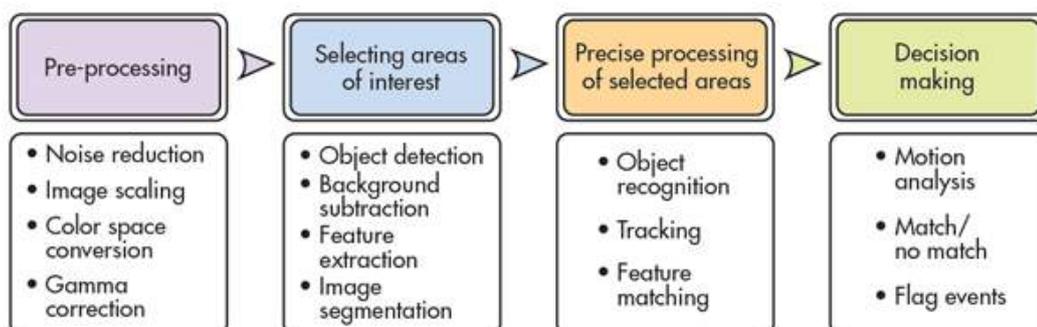


Abbildung 1: Ablauf der Objekterkennung

*Diese Arbeit wurde durchgeführt bei der Firma IT-Designers GmbH, Esslingen

Merkmalerkennung

Oft reichen die Segmentierungsschritte nicht aus, um ein Objekt klar zu erkennen. Deshalb ist es meist notwendig, gezielt nach einzelnen Merkmalen eines Objekts zu suchen. Merkmale eines Objekts beschreiben dieses so, dass es sich klar von anderen unterscheidet. Sehr markante Objektmerkmale sind Kanten, welche oft die Konturen eines Objekts bilden. Diese Kanten werden anhand eines starken Umbruchs des Farb- oder Grauwerts erkannt. Die Funktionsweise eines Kantendetektors umfasst in der Regel die Berechnung der Farbwertgradienten jedes einzelnen Bildpunkts durch Untersuchung des umliegenden Bereichs. Dies erfolgt durch diskrete Faltung des Bildes mit einer Faltungsmatrix. Die resultierende Matrix der Gradienten an jedem Bildpunkt wird Kantenbild genannt. Für die Kantenerkennung wird meist der Canny Edge Algorithmus verwendet [3].



Abbildung 2: Canny Edge Algorithmus

Erkennungsalgorithmen die sogenannte Schlüsselpunkte erkennen und beschreiben sind beispielsweise SIFT (Scale Invariant Feature Transform) oder SURF (Speeded-Up Robust Features). Diese sind im Gegensatz zum

Canny Edge Algorithmus Invariant gegenüber Skalierung, Translation und Rotation. Wie in Abbildung 3 zu sehen werden die Objekte (Trainingsbilder links) hier unabhängig von ihrer Größe und Orientierung im Bild rechts lokalisiert. Die kleineren Rechtecke stellen die erkannten Merkmale da und das jeweils Große umschließt das gefundene Objekt.



Abbildung 3: Objekterkennung mit SIFT

Entscheidungsfällung

Im letzten Schritt der Objekterkennung wird entschieden, wie mit dem erkannten Objekt weiterhin umgegangen wird. Das Objekt wird eindeutig identifiziert und klassifiziert, dessen Bewegung analysiert und daraufhin weitere Schritte eingeleitet. Im Bereich des autonomen Fahrens würde das Fahrzeug auf ein erkanntes Verkehrsschild reagieren oder bei einem erkannten Hindernis ausweichen oder bremsen.

Im Rahmen dieser Bachelorarbeit soll ein System für ein autonom fahrendes Modellfahrzeug entwickelt werden, welches Objekte auf der Fahrbahn erkennt und diese als Hindernisse identifiziert. Bei einer Erkennung eines Hindernisses soll das Modellfahrzeug eine Kollision mit diesem vermeiden.

[1] Tesla autopilot. https://www.tesla.com/de_DE/autopilot. Eingesehen am 14.11.2017

[2] Reinhard Klette. Handbuch der Operatoren für die Bildbearbeitung. Vieweg+ Teubner Verlag, 1992

[3] John Canny. A computational approach to edge detection. IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, Vol. PAMI-8,(Nr. 6):S.679-698, 1986

Bildquellen:

- Abbildung 1: Conquer the challenge of integrating efficient embedded vision. <http://www.electronicdesign.com/embedded/conquer-challenge-integrating-efficient-embedded-vision>. Eingesehen am 19.09.2017.
- Abbildung 2: The usc-sipi image database. <http://sipi.usc.edu/database/database.php?volume=misc&image=12#top>. Eingesehen am 09.09.2017.
- Abbildung 3: David G. Lowe. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, Vol. 60(Nr. 2):S.91-110, 2004.

Die Entwicklung elektronischer Glücksspiele

Phillip Sailer*, Kai Warendorf, Dominik Schoop

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Der heutige Glücksspielmarkt ist in fast allen Ländern heute nicht mehr wegzudenken. Bis auf ein paar Ausnahmen wie China, Indien und weitere Länder ist das Glücksspiel um Geld unter bestimmten Auflagen erlaubt. Es kommt ganz darauf an von wem die Spiele bereitgestellt werden.

Dabei wurde schon bereits 3000 v. Chr. mit Hilfe des Zufalls um Wertsachen gespielt. Viele waren sogar so sehr in ihrem Wahn verfallen, dass sie um ihr eigenes Leben gespielt haben.

Umso wichtiger ist es heute das Glücksspiel in gewisser Weise einzudämmen, damit Menschen vor dem eigenen Ruin geschützt werden können. [1]

In der folgenden Arbeit sollen viele Hintergründe und Informationen aus der Vergangenheit aufgearbeitet werden, um eine Idee davon zu bekommen welche Lösungen heutige Spielehersteller anbieten, um Menschen noch heute für das Glücksspiel zu gewinnen.

Gäbe es heute keine technischen Innovationen im Glücksspielmarkt, wären wir bei weitem nicht so erfolgreich wie heute.

Wenn man sich den Zufall von Würfelspielen ansieht, erkennt man, dass der Spieler einen Einfluss auf den Zufall hat. Er kann die direkte Bewegung physikalisch nachvollziehen. Die notwendige Präzision ist natürlich nicht gegeben, um die exakten Positionen zu bestimmen.

Anders verhält es sich bei Glücksspielen, welche vom Computer simuliert sind. All diese Bewegungen können nicht vom Spieler beeinflusst werden, da sie nur virtuell existieren. Daher ist es umso wichtiger den Spielern das Gefühl zu vermitteln, dass sich in den Bildschirmen reale Objekte bewegen. Jede Kombination muss für sie aussehen, als hätte sie direkten Einfluss auf das Spielgeschehen.

Ein weiterer Faktor spielt die Auszahlungsquote. Diese kann in virtuellen Spielen beliebig definiert werden. Wo hingegen das Roulette Spiel nicht beeinflusst werden kann, da man schlecht noch ein blaues Feld in das vorhandene Zahlenfeld hinzufügen kann.

Es ist für die Spieler weniger interessant, wenn sie beispielsweise innerhalb einer Stunde immer wieder kleinere Gewinne haben und am Ende dennoch nur 95% zurückbekommen. Unschön für Spieler wäre auch, wenn sie in ei-

ner Stunde nur einmal etwas größeres gewinnen. Spiele müssen von den Chancen her so entwickelt werden, dass sie den realen Spielen ähnlich sind und von der Gewinnhöhe sehr variieren.



Abbildung 1: Zeigt die ersten Würfel aus Ägypten.

In der nachfolgenden Arbeit soll herausgefunden werden, welche Mittel und vor allem Probleme eine Spieleindustrie für Glücksspiele hat.

Wie man in den letzten Jahren feststellen konnte, haben sich die Spiele in diesen Bereichen stetig verbessert. Sowohl technisch als auch vom Spielablauf her.

Waren es zu Beginn der Glücksspielzeit, wie oben beschrieben, einfache Würfel, gab es nach und nach Fortschritte. Beispielsweise kamen später im Mittelalter die ersten Spielkarten wie wir sie heute kennen hinzu. Mit diesen ergaben sich völlig neue Spielmöglichkeiten. Es war nun möglich mit den Karten selbst neue Spieleideen zu erschaffen. Neben nichts gewinnen, aber in der letzten Minute doch einen Hauptgewinn haben, welcher ihnen 95% zurückzahlt.

Dies würde vielleicht den gesetzlichen Regelungen entsprechen, sich an eine gewisse Auszahlungsquote zu halten, nicht aber den Spielspaß fördern. [2]

Am Ende möchten Spieler unterhalten werden und dennoch viel gewinnen. Theoretisch könnte man auch eine Art Geldautomat aufstellen, welcher den Gewinnbetrag für eine eingegebene Spielzeit nach Ablauf des Spieles sofort ausbezahlt.

Viele Spieler würden dann wahrscheinlich sagen, dass das nicht der Wahrheit entspre-

*Diese Arbeit wurde durchgeführt an der Fakultät Informationstechnik

chen würde und der Automat nicht wahrheitsgemäß arbeitet.

Somit spielt die Art und Aufbau des Spiels eine sehr wichtige Rolle und soll in dieser Arbeit genauer untersucht werden.

Die Arbeit soll durch ein selbstentwickeltes Spiel ergänzt werden, um die Effekte in heutigen Spielen ein wenig zu veranschaulichen und nachvollziehen zu können.

Nachfolgend einer der ersten Automaten, welche dem „Einarmigen Bandit“ aus dem Jahre 1889 ähneln.



Abbildung 2: Zeigt die ersten Spielautomaten der Firma Gauselmann aus dem Jahre 1977.

Grundsätzlich gibt es vier unterschiedliche Arten von Glücksspielen. Neben den Spielautomaten, welche in dieser Arbeit behandelt werden, gibt es die bekannten Kartenspiele wie Poker oder Black Jack. Ein weiteres Spiel, welches hauptsächlich in Casinos zu finden ist, wäre Roulette.

Zu guter Letzt und was sich immer größerer Beliebtheit erfreut, sind die Sportwetten. Mittlerweile gibt es viele unterschiedliche Sportarten, auf die man wetten kann.

Allerdings ist es nicht immer legal Glücksspiele anzubieten. Dazu gibt es in Deutschland mehrere Verordnungen, welche in der Arbeit zusätzlich erläutert werden.

Beispielsweise existiert ein großes Angebot an Wettbüros und Online Casinos in Deutschland nur, weil sie sich in einer gesetzlichen Grauzone befinden.

Viele Glücksspiellizenzen werden heutzutage nicht mehr in Deutschland beantragt, sondern in Malta. Damit ist es für die EU bzw. Deutschland nicht möglich, Dienstleistungen, die in Deutschland angeboten werden, zu verbieten.

So agierte auch einer der bekanntesten Wettanbieter wie Tipico bis zum Jahr 2012. Mittlerweile verfügen sie über eine gültige Sportwetten-Lizenz in Schleswig-Holstein und verzeichnen Umsätze um bis zu 500 Millionen Euro. [3]



Abbildung 3: Zeigt den Hauptsitz der Firma Tipico in Malta.

[1] <https://www.casinoerdiener.com/wissenswertes/geschichte-des-gluecksspiels>

[2] <https://www.gamblejoe.com/ratgeber/spielautomaten-auszahlungsquote>

[3] http://www.focus.de/finanzen/experten/christian_edler/branchenkonzentration-nimmt-fahrt-auf-milliardenschwerer-markt-wer-schnappt-sich-den-sport-wetten-anbieter-tipico_id_5365777.html

Bildquellen:

- Abbildung 1: <https://www.casinoerdiener.com/wp-content/uploads/2016/08/image07.jpg>
- Abbildung 2: <https://www.casinoerdiener.com/wp-content/uploads/2016/08/image12.jpg>
- Abbildung 3: https://c1.staticflickr.com/4/3239/2814071370_f4cd83724d_b.jpg

Maschinelles Lernen und Edge Computing im Automobilbereich auf dem TX2

Fabian Schilling*, Reinhard Schmidt, Andreas Rößler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Herausforderung

In Fahrzeugen fallen mehr und mehr Messdaten an. Diese Daten werden für die Automobilbranche immer interessanter, da mit deren Hilfe Fahrfunktionen verbessert werden können. Idealerweise müssten alle diese Daten in die Cloud geladen und dort verarbeitet werden. Das ist bei einem Fahrzeug nicht möglich, da während einer Fahrt die Upload-Geschwindigkeit nicht garantiert werden kann. Neben der Geschwindigkeit mit denen die Daten hochgeladen werden können, ist mit Blick auf das autonome Fahren auch die Reaktionszeit des Fahrzeugs wichtig. Die Funktionen, die das autonome Fahren ermöglichen, können nicht in der Cloud berechnet werden, da das Fahrzeug auch ohne Online-Verbindung sicher fahren muss.

Edge Computing

In Bereichen in denen die Cloud, aufgrund von ihrer Abhängigkeit und Limitationen wie ständigem Internetzugriff und hohen Latenzen, nicht einsetzbar ist, kann Edge Computing eingesetzt werden. Bei Edge Computing werden Funktionen, die normalerweise in der Cloud angeboten werden, im lokalen Netzwerk angeboten. Sensordaten werden beispielsweise an das Edge Gerät geschickt, welches diese auswertet. Das System kann daraufhin schnell auf Events reagieren. Zusätzlich können die Daten in der Edge gefiltert und nur die relevanten Daten in die Cloud geschickt werden, um den Netzwerkverkehr nur auf das Nötigste zu reduzieren. Ein Anwendungsbereich für Edge Computing sind Fahrzeuge, da diese nicht ständig mit der Cloud verbunden sein können und viele Fahrfunktionen eine sehr niedrige Latenz benötigen. Beim autonomen Fahren erhöht sich dabei nochmal die Datenmenge. Allein durch Kameras, Radarsensoren und Lidar kommen mehrere Gigabyte pro Minute an Daten zustande, die mit der heutigen Technik allein für ein Fahrzeug nur schwer und

für mehrere hundert Fahrzeuge unmöglich in die Cloud hochgeladen werden können. Diese Daten müssen demnach im Fahrzeug verarbeitet werden, gleichzeitig relevante Daten herausgefiltert und hochgeladen werden.

Maschinelles Sehen

Neben den klassischen Reduktionsmethoden, wie der Komprimierung oder der Selektion von Grenzparametern für Variablen, eröffnet das maschinelle Sehen eine weitere Möglichkeit Daten zu Filtern. Das maschinelle Sehen ermöglicht es, einfache Situationen zu erkennen wie beispielsweise eine rote Ampel, Verkehrsschilder oder die Erkennung der Fahrbahnmarkierung. Damit ist es möglich, nur in bestimmten Fahrsituationen relevante Fahrzeugdaten zu speichern.

Maschinelles Lernen

Damit Fahrzeuge in Zukunft autonom fahren können, müssen Verkehrssituationen richtig erkannt werden. Dafür ist maschinelles Lernen eine essentielle Schlüsseltechnologie. Mit maschinellem Lernen gelingt es Computern Zusammenhänge in Daten zu erkennen. Im Fall des autonomen Fahrens bekommt das maschinelle Lernmodell sowohl Daten von der Umgebung über Video, Rader oder Lidar Sensoren, aber auch die fahrzeugeignen Daten, wie Geschwindigkeit und Fahrriechtung. Das Modell verarbeitet alle diese Daten und berechnet, welche Aktion das Fahrzeug tätigen soll. Damit das Modell richtige Entscheidungen treffen kann, muss es trainiert werden. Ein Ansatz dafür ist das überwachte Lernen, bei dem das System ein Set an Trainingsdaten bekommt, die aufgeteilt sind in Eingangswerte und der gewünschten Reaktion. Beim Trainieren versucht das Modell anschließend seine Parameter so zu verändern, dass seine Reaktion möglichst oft und genau mit der gewünschten übereinstimmt [1].

*Diese Arbeit wurde durchgeführt bei der Firma Robert Bosch GmbH, Feuerbach

Jetson TX2

Das Jetson TX2 ist ein von Nvidia entwickelter Einplatinencomputer, der im Gegensatz zu den meisten Andern einen leistungsstarken Grafikprozessor besitzt [2]. Von diesem Prozessor profitieren Anwendungen des maschinellen Lernens und Sehens, da die dafür verwendeten Algorithmen für das Rechnen auf dem Grafikprozessor geeignet und optimiert sind. Aufgrund seiner Größe von nur 50 x 87 Millimetern und relativ geringem Stromverbrauch von maximal 15W, ist das TX2 außerdem ein guter Kandidat für ein Edge Gerät, da es einfach in bereits bestehende Systeme integriert werden kann.

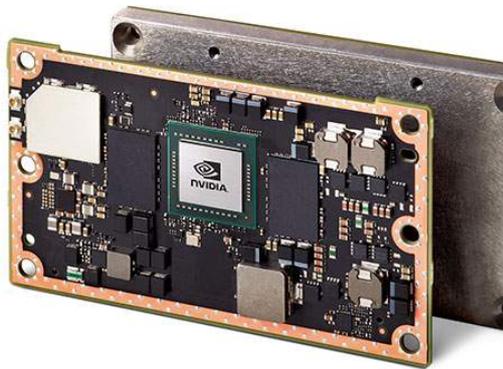


Abbildung 1: Jetson TX2

Container-Technologie

Mithilfe von Container lassen sich Anwendungen, ähnlich wie in virtuellen Maschinen, voneinander isolieren. Diese sind allerdings performanter als virtuelle Maschinen. Abhängigkeiten wie Softwarebibliotheken müssen innerhalb des Containers installiert werden und sind für das restliche System nicht sichtbar. Container verwenden dazu Funktionen des Linux Kernels und sind somit nur unter Linux wirklich effizient. Container können unter Windows dennoch durch den Einsatz einer virtuellen Linux Maschine verwendet werden. Die Anwendungen für das maschinelle Lernen und Sehens sollen auf dem TX2 einfach installierbar sein. Um dieses zu erreichen, werden

die verschiedenen Anwendungen in Container gepackt. Damit hat jede Anwendung seine eigene Laufzeitumgebung und ist somit unabhängig von allen anderen. Neben der einfacheren Installation der Software, wird das Updaten und Entfernen ebenfalls wesentlich vereinfacht. Bei einem Update kann der alte Container durch den neuen ausgetauscht werden und auf alte oder nicht mehr benötigte Abhängigkeiten muss nicht mehr geachtet werden. Um die Software und alle ihre Abhängigkeiten zu entfernen, muss nur der Container gelöscht werden. Container helfen also dabei das System frei von unnötigen Abhängigkeiten zu halten.

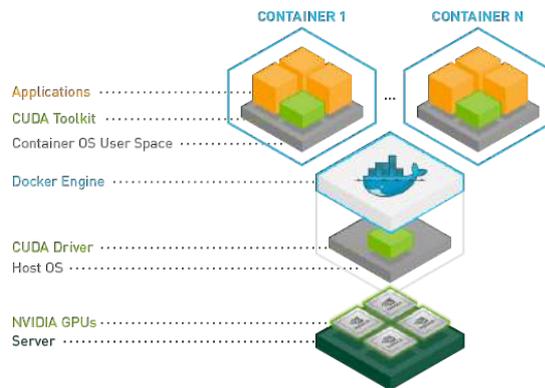


Abbildung 2: Nvidia Docker

Inhalt der Bachelorarbeit

Um die Integration von neuer Software zu vereinfachen sollen Container Umgebungen erstellt werden, welche bereits bestimmte Abhängigkeiten mit sich bringen. Ein konkretes Beispiel ist eine Umgebung, in der alle gängigen Abhängigkeiten vom Tensorflow, einem Framework für maschinelles Lernen, enthalten sind. Ein Entwickler kann seinem Tensorflow-Code in diesen Container integrieren und muss sich nicht um das Bereitstellen der Laufzeitumgebung kümmern. Getestet wird die Umgebung mit einem bereits erstellten und trainierten Modell zur Ampelerkennung.

[1] Ian Goodfellow, Yoshua Bengio, Aaron Courville: "Deep Learning". MIT Press, 2016, S.136

[2] Nvidia, NVIDIA® Jetson(tm) Die embedded Plattform für selbstständige Systeme (2017), <http://www.nvidia.de/object/embedded-systems-dev-kits-modules-de.html> (Stand:20.11.2017)

Bildquellen:

- Abbildung 1: images.nvidia.com/content/tegra/embedded-systems/images/jx10-jetson-tx2-170203.jpg
- Abbildung 2: images.nvidia.com/content/products/hpc/gpu-enabled-docker-containers.png

Konzipierung und Realisierung einer Erweiterung des Engineering-Tools TX-Open der Siemens AG für das Erstellen einer XML-basierten Konfiguration zur Anbindung von Komponenten über den Feldbus Modbus an die Gebäudeleittechnik

Tobias Schmadl*, Reinhard Keller, Walter Lindermeir

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

In den technischen Bereichen gilt bereits seit dem 19. Jahrhundert der Grundsatz, dass die Produktivität und die Effizienz der Prozesse kontinuierlich gesteigert werden muss. Mit entsprechenden Erfindungen wie der Fließbandfertigung von Henry Ford zu Beginn dieser Entwicklung über die Einführung rechnerintegrierter Produktionen bis hin zur Umsetzung von Industrie 4.0 nehmen die Mechanisierung von technischen Arbeitsschritten und die Automatisierung ganzer Prozessabläufe bis heute immer weiter zu [1].

Mit zunehmender Automatisierung steigt daher auch der Bedarf an Vernetzung der einzelnen, meist nicht kompatiblen Systeme [2]. Für den Informationsaustausch vernetzter Systeme werden Kommunikationsprotokolle benötigt. Eines der am häufigsten eingesetzten Protokolle ist Modbus, entwickelt 1979 von der Firma Gould-Modicon (heute: Schneider Electric), welches heutzutage als einer der Industriestandards gilt [3].

Unabhängig von dem im Feld eingesetzten Protokoll kann mittels eines Protokollübersetzers (Gateway) die Aufschaltung auf eine Gebäudeleittechnik (GLT) erfolgen. Die GLT ging aus dem zunehmenden Bedarf an Automatisierung in Gebäuden hervor. Die Grundlage hierfür wurde in den 60er-Jahren gelegt, als die ersten Gebäudeleitsysteme aufkamen. Sie dient der Energieeinsparung unterschiedlichster Anlagen und gleichzeitig einer Erhöhung des Komforts für die Nutzer. Unter der GLT wird eine zentrale Leittechnik in Gebäuden verstanden, die technische Anlagen steuert und überwacht. Hierzu zählen beispielsweise Elektroanlagen, heizungs- und raumlufttechnische Anlagen, sanitärtechnische Anlagen, fördertechnische Anlagen, sowie Ver- und Entsorgungsanlagen. Der Trend geht zu einer weiteren Vernetzung mit Anlagen aus der Brandmelde- und Sicherheitstechnik. Die GLT wurde sowohl in Hinblick auf Komfort als auch auf Energiesparmaßnahmen kontinuier-

lich weiterentwickelt, sodass heute komplexe Regelalgorithmen existieren [4].

Bei der Siemens AG werden Modbus-unterstützende Geräte verwendet. Ein solches Gerät ist in Abbildung 1 dargestellt. Dabei handelt es sich um ein Multifunktions-Messgerät, welches zur Anzeige, Speicherung und Überwachung der Netzparameter in Niederspannungs-Energieverteilungen dient.



Abbildung 1: Siemens SENTRON PAC4200

Für die Aufschaltung solcher Geräte auf das Desigo Gebäudeautomationsystem ist entsprechendes Engineering von Nöten, wobei hier als Feldbus der Modbus verwendet wird, der Daten über ein Master-/Slave Protokoll überträgt.

Die Hauptanforderungen an Feldbussysteme sind die Verfügbarkeit und die Störuneempfindlichkeit der Datenübertragung, unabhängig von den Umgebungsbedingungen. Weiterhin wichtig sind eine großflächige und flexible Vernetzung sowie eine einfach gehaltene Technik in Bezug auf Anschluss und Handhabung. Außerdem spielen sowohl harte als auch weiche Echtzeitfähigkeit und die Übertragungsgeschwindigkeit eine große Rolle in der Anwendung. Die Einbindung des Modbus in das Gesamtsystem der Desigo ist in Abbildung 2 dargestellt.

Bisher wurden zur Interpretation der Nutzdaten des Protokolls sowohl Freeware als auch lizenzpflichtige Tools eingesetzt. Diese boten jedoch nicht die notwendigen Schnittstellen für die firmeneigene Software Desigo XWORKS, so dass Siemens-Techniker manuell jeweils spe-

*Diese Arbeit wurde durchgeführt bei der Firma Siemens AG, Stuttgart

zielle Punktvorlagen im XML-Format für jedes Gerät im firmeneigenen Desigo TX-Open erstellen mussten. Hierbei war jedoch zum einen die Fehleranfälligkeit, beispielsweise in Bezug auf die Formatierung von Mehrbyte-Datenwerten (Little- bzw. Big-Endian) und Datenformate wie SINT/UINT, sehr hoch. Zum anderen ist diese Vorgehensweise sehr zeitaufwändig und wenig bedienerfreundlich, da jeder Hersteller für seine Geräte eigene Speicheradressen vergibt.

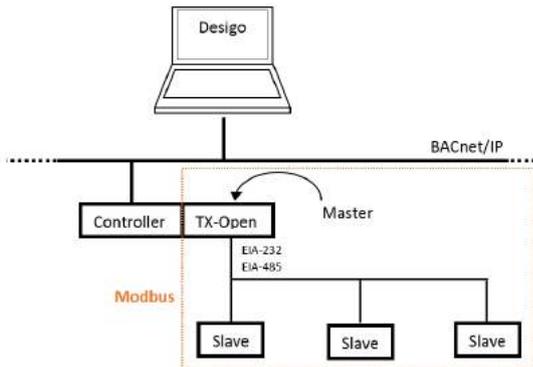


Abbildung 2: Schematische Darstellung Desigo Gebäudeautomationsystem

Erstellung eines protokollkonformen Software-Tools

Da bisher kein Programm in der Firma vorhanden war, das alle gewünschten Anforderungen erfüllte und intuitiv bedienbar war, bestand die Notwendigkeit, ein entsprechendes Software-Tool zu entwickeln, das auch dem Modbus-Standard genügt. Dieses sollte dann schrittweise um alle anwendungsspezifischen Anforderungen erweitert werden, die im Vorfeld von Siemens definiert worden waren.

Im Rahmen der vorliegenden Arbeit wurde ein Tool konzipiert und entwickelt, das diese Anforderungen berücksichtigt und über die serielle Kommunikationsschnittstellen EIA-232/485 eine Verbindung zu Modbus-Slaves herstellt (siehe Abbildung 2). Als Funktionen wurden zum einen das Auslesen der entsprechenden Werte durch Angabe von Funktionscodes und Register und zum anderen eine For-

matierung der Werte implementiert.



Abbildung 3: Siemens TX1-Open Koppelmodul zur Anbindung über EIA-232/485

Analyse der Firmensoftware-Bibliothek

In einem weiteren Schritt wurde die Struktur der Firmensoftware-Bibliothek analysiert, die Daten für alle im Unternehmen verwendeten Protokolle bereithält. Dieser Schritt war zur vollständigen Zusammenstellung und Bereitstellung aller auszuwertenden Daten für das Tool erforderlich. Hierfür mussten insbesondere die Modbus-relevanten Daten mittels eines dafür erstellten Hilfsprogramms gefiltert werden. In Vorbereitung für das Hauptprogramm wurde darauf geachtet, dass auf der Datenstruktur aufbauend eine graphische Benutzeroberfläche erzeugt werden konnte.

Finalisierung des Tools

Abschließend musste die zur Software TX-Open kompatible XML-Struktur ermittelt werden, damit das finale Tool automatisiert die gewünschte Punktvorlage erstellen konnte.

Als Ergebnis dieser Arbeit entstand eine Software, die anhand festgelegter Parameter eine sogenannte Punktvorlage im XML-Format erstellen kann. Diese wird unabdingbar für den weiteren Kopplungsprozess benötigt. Das entstandene Software-Tool ermöglicht eine grafische Benutzeroberfläche und ist ohne aufwändige Installation auf den von den Technikern eingesetzten Systemen ausführbar.

- [1] Schnell, G.; Wiedemann, B.: Bussysteme in der Automatisierungs- und Prozesstechnik, Vieweg, 2006.
 [2] Berns, K.; Schürmann, B.; Trapp, M.: Eingebettete Systeme: Systemgrundlagen und Entwicklung eingebetteter Software, Vieweg+Teubner, 2010.
 [3] Kunbus GmbH (Hrg.): Modbus – www.kunbus.de/modbus.html Stand 30.10.2017 21:12
 [4] M. Ellrich: Gebäudeleittechnik und Energiemanagement in Nichtwohnbauten, Springer, 1991.

Bildquellen:

- Abbildung 1, 3: Siemens AG
- Abbildung 2: Eigene Abbildung

Die Prinzipien von Scrum

Daniel Fabian Seitz*, Reinhard Schmidt, Andreas Rößler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Probleme wie zum Beispiel die verspätete Auslieferung der Software, falsch interpretierte oder implementierte Anforderungen der Software, sind seit der Einführung der agilen Softwaremethoden wie Scrum und Kanban deutlich seltener der Fall. Agile Entwicklerteams arbeiten heute fast schon täglich mit dem Kunden zusammen und geben ihm somit ein Gefühl der Mitbestimmung und der direkten Einflussnahme auf das Softwareprojekt. Dadurch können Fehler oder falsche Entscheidungen schnell erkannt und behoben werden und dadurch eine hohe Kundenzufriedenheit erzielt werden.

Scrum

Scrum stellt ein Rahmenwerk dar, um innerhalb eines Projektteams kooperativ, selbstverantwortlich und selbstorganisiert zu arbeiten. Die hohe Einbeziehung des Kunden ist hierbei essenziell [2]. Zum einen soll flexibel auf Kundenwünsche reagiert werden. Zum anderen soll der größtmögliche Kundennutzen geniert werden. Zusätzlich wird eine schnelle, effektive und kostenorientierte Entwicklung ermöglicht [3].

Scrum ist unterteilt in drei interne Rollen sowie drei externe Rollen:

Zu den internen Rollen zählen:

- **Product Owner:** Er steht in ständigem Kontakt zum Kunden und ist für den wirtschaftlichen Erfolg des Projektes verantwortlich.
- **Entwicklerteam:** Das Entwicklerteam ist für die Umsetzung der zu entwickelnden Features zuständig.
- **Scrum Master:** Er dient der Unterstützung des Scrum Teams in technischen sowie persönlichen Konflikten [2].

Die externen Rollen sind:

- **Kunde:** Stellt den Empfänger der Software dar. Ist der Auftraggeber.
- **Anwender:** Ist der Endnutzer der Software.
- **Management:** Das Management stellt die Rahmenbedingungen des Projektes sicher [1].

Der Scrum Prozess stellt sich wie folgt dar. Zu Beginn wird vom Product Owner der Product Backlog initial befüllt, indem sich alle zum Projektstart geforderten Anforderungen des Kunden bezüglich der Software befinden, anschließend folgt das Sprint Planning Meeting. Daraus resultiert der Sprint Backlog indem sich alle gesammelten Anforderungen des Kunden an die gewünschte Software befinden. Im Anschluss folgt der Sprint indem die aus dem Sprint Backlog eingetragenen Anforderungen in einzelnen Sprints iterativ abgearbeitet werden. Täglich wird dazu ein Daily Scrum abgehalten. Die Sprint Retrospektive und das Sprint Review bilden den Abschluss. Nach Abschluss eines jeden Sprints steht eine auslieferfähige Softwareversion [2].

Im Product Backlog befinden sich alle gesammelten Anforderungen des Kunden an die gewünschte Software. Im Sprint Planning Meeting wird geplant, was im nächsten Sprint umgesetzt werden kann. Das wird dann im Sprint Backlog festgehalten und das Sprint Ziel wird formuliert. Im anstehenden Sprint sind nur diese Anforderungen vom Entwicklerteam zu bearbeiten. Der Sprint Backlog dient dem Team als Leitfaden und Organisation des Sprints. Unter dem Sprint Ziel versteht man die verbale Beschreibung dessen was der Product Owner dem Kunden nach Ablauf des Sprints präsentieren möchte.

Über den anschließenden gesamten Sprint Verlauf wird der Fortschritt auf einem Scrum-Board festgehalten. Dieses besteht aus User Stories und Tasks die in den Daily Scrums, anhand des Scrum Boardes, in ihrem Bearbeitungsstatus ständig aktualisiert werden.

Das Sprint Review stellt dann die Überprüfung der zu entwickelten Teilsoftware aus dem jeweiligen Sprint dar. Der Product Owner muss die Software anschließend prüfen und abnehmen. Innerhalb der Sprint Retrospektive soll abschließend der vergangene Sprint analysiert werden. Wo hatte das Entwicklerteam Probleme oder was lief besonders gut. Unter dem Working Increment versteht man die auslieferfähige Softwareversion [2].

* Diese Arbeit wurde durchgeführt bei der Firma IT-Designers GmbH, Esslingen Neckar

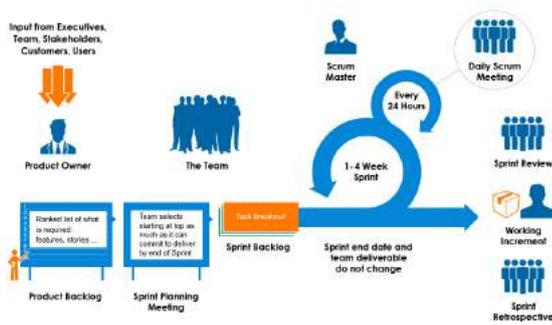


Abbildung 1: Scrum Prozess

Unterstützt wird Scrum durch folgende Techniken:

User Stories

Eine User Story dient der Kommunikation zwischen Entwicklern und Kunden. Notiert werden diese User Stories in der Regel auf Karteikarten oder in Computerprogrammen. Es wird auch von einer „Geschichte erzählen“ gesprochen, bei der die Fragen Wer, Was und Warum beantwortet werden [4]. User Stories werden im Product Backlog eingetragen.

Taskboard/Scrum Board

Heutzutage wird bei der Verwendung von Taskboards, im Einsatz von Scrum, in eine analoge und digitale Variante unterschieden. Beide Variationen bringen entsprechende Vorteile und Nachteile mit sich.

Die analoge Verwaltung zeichnet sich durch eine schnelle Projektaufsetzung und eine kostengünstige Durchführung aus da keine zusätzlichen Rechner und Bildschirme benötigt werden die dauerhaft laufen und überwachen. Somit kann der Projektfortschritt dauerhaft überwacht werden. In dem Daily Scrum kann somit auch bildlich gemacht werden an welcher Anforderung gearbeitet wurde und ein größeres Grundverständnis im gesamten Team geschaffen werden.

Nachteil ist, dass durch viele Tickets das Board schnell unübersichtlich werden kann, und bei handschriftlichen Tickets die Lesbarkeit nicht immer gegeben ist. Ebenso ist ein Verlust von Tickets und ein dadurch resultierender Informationsverlust ein mögliches Problem.

Bei den digitalen Boards entfallen die Probleme der Lesbarkeit sowie des Informationsverlustes bei Verlust von Tickets. Zusätzlich wird das verteilte Arbeiten zum Beispiel im Homeoffice oder Remote Teams ermöglicht. Nachteil der Variante stellt die digitale Verarbeitung dar. Bei einem Ausfall der Internetverbindung sind die Daten bis zur Problembhebung nicht mehr, bei online, oder nur noch lokal zu erreichen. Des Weiteren kann es durch eine Störung passieren das durch Probleme mit dem verwendeten Programm Tickets falsch angezeigt oder sich in falscher Position befinden. Aufgrund dessen verwenden viele Scrum Teams heute eine Kombination aus einem analogen und einem digitalen Scrum-Board. Somit sind die Daten in doppelter Hinsicht gesichert, es erfordert jedoch auch eine gegenseitige Synchronisation beider Modelle. Dies erfolgt in der Regel per Hand und erfordert somit einen zusätzlichen Zeitaufwand.

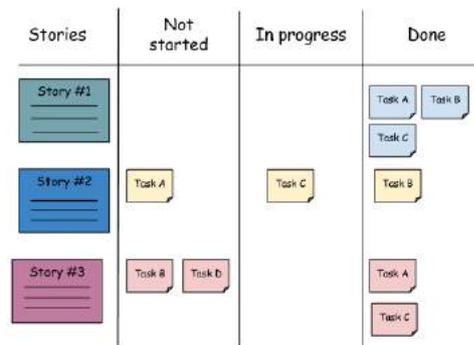


Abbildung 2: Analoges Scrum Board

Meine Bachelor Arbeit beschäftigt sich daher mit der Verknüpfung eines analogen Scrum Boardes mit einem Issue Tracker unter der Verwendung von bekannten Bildverarbeitungsalgorithmen und OpenCV.

- [1] Eckhart Hanser. Agile Prozesse: Von XP über Scrum bis MAP. Springer Verlag, 2010.
 [2] Daniel Hommel Joachim Goll. Mit Scrum zum gewünschten Erfolg. Springer Verlag, 2015.
 [3] Carsten Sahling Rolf Dräther, Holger Koschek. Scrum Kurz und Gut. O Reilly, 2013.
 [4] Andreas Wintersteiger. Scrum Schnelleinstieg. entwickler.press, 2012.

Bildquellen:

- Abbildung 1: InterWorks Tome Hristovski. Scrum prozess, Januar, 2017. <http://interworks.com.mk/why-scrum-simply-works/> [Zugriff am 08.11.2017].
- Abbildung 2: Manifesto Jim Bowes. Scrum taskboard, 02.10.2014. <https://manifesto.co.uk/agile-concepts-scrum-task-board/> [Zugriff am 16.10.2017].

Jump'n'run Spiel für Touch-Tables in Unity

Armin Tausch*, Reinhard Schmidt, Andreas Rößler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Große Touch-Displays sind sehr unhandlich und die Interaktion mit einem Display an der Wand nicht immer das Beste. Aus diesem Grund werden große Displays, mit denen interagiert werden soll, oftmals in Form eines Touch-Tables in einen Tisch integriert. Der Touch-Table ist ein Gerät, das bisher keine weite Verbreitung erfahren hat, weder in privaten Haushalten, noch im öffentlichen Sektor. Oftmals werden Touch-Tables im Rahmen von Konferenzräumen oder Messen als Präsentationsmedium genutzt. Die Verbreitung von Geräten hängt aber auch immer von ihren Möglichkeiten und Einsatzzwecken ab.



Abbildung 1: Ein Touch-Table

Für Touch-Tables bieten sich sehr viele Einsatzmöglichkeiten, die bislang kaum bis gar nicht genutzt werden. Denkbar wäre ein Touchtable beispielsweise bei gesellschaftlichen Abenden als Basis für digitale Brettspiele oder andere Videospiele. Das Display wäre hierbei (je nach Touch-Table) ausreichend groß um für 4–12 Spieler Platz zu bieten, wobei man je nach Spielart das Display in mehrere Abschnitte unterteilen müsste. Hierbei gibt es jedoch ungewohnte Anforderungen an die Spieleentwickler, da das Display unter Umständen viele Berührungen gleichzeitig verarbeiten muss. Multitouch nennt man die

Technologie, die die Erkennung mehrerer Berührungen auf einmal ermöglicht, jedoch ist die Anzahl der maximalen Berührungspunkten begrenzt und von Display zu Display unterschiedlich.

Diese Bachelorthesis hat das Ziel ein Jump'n'run Spiel zu entwickeln, das mit bis zu 4 Spielern auf einem Touch-Table gespielt werden kann. Selbst bei gerade mal 4 Spielern, schränkt die maximale Anzahl an Berührungspunkten des Touch-Tables die Funktionalität des Spiels ein. Das Testgerät erfasst gerade mal 6 Berührungspunkte gleichzeitig. Wenn jeder der 4 Spieler seine Spielfigur gleichzeitig nach rechts oder links steuern und dabei noch springen lassen möchte sind bis zu 8 Berührungspunkte nötig. Das Jump'n'run lässt daher die eigenständige Steuerung der Spielfigur in die Seiten nicht zu. Die Spieler können ihre Figuren also springen lassen und ducken lassen, ansonsten nichts. Da man das Spiel aber nicht derart statisch belassen kann, laufen die Figuren automatisch nach rechts durch die Welt und der Spieler muss nahenden Hindernissen rechtzeitig ausweichen.

Das Jump'n'run Spiel wird mit der Unity Engine entwickelt. Die Unity Engine ist eine Grafik-Engine, die die Entwicklung von Videospielen soweit wie möglich abstrahieren, vereinheitlichen und vereinfachen soll.[1] In der Engine sind einfache und realistische Spielphysiken für 3D und 2D Objekte standartmäßig integriert. Spielobjekte können nach Belieben hinzugefügt und entfernt werden, ihre Interaktionsweisen miteinander simpel beschrieben werden, aber auch komplexe Verhaltensweisen können mit Skripten in C# oder Javascript den Objekten beigebracht werden. Die Objekte können in Form und Farbe verändert werden, animiert werden und in verschiedene Szenen platziert werden. Das Spiel wird für Windows 8 entwickelt, da die meisten Touch-Table zurzeit auf Windows setzen und daher eine möglichst große Kompatibilität erzielt wird, doch auch eine Portierung für Android Tablets wäre denkbar.

*Diese Arbeit wurde durchgeführt an der Fakultät Informationstechnik

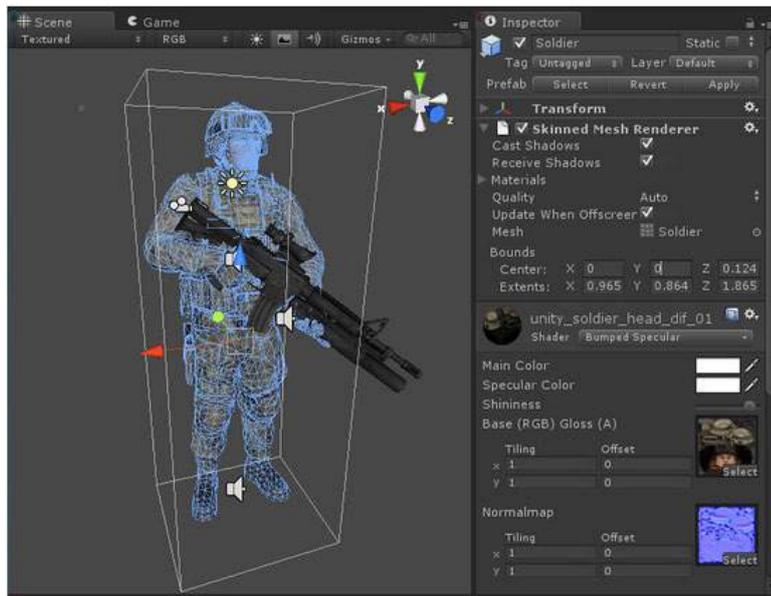


Abbildung 2: Der Editor für die Unity Engine

Das Jump'n'run Spiel ist in 2D Grafik gehalten, was für die Funktionalität des Spiels ausreichend ist und durch die Reduzierung der Grafik- und Rechenleistung den Touch-Table entlastet, wodurch das Spiel auch auf schwächerer Hardware noch flüssig läuft. Das Spiel soll für 1–4 Spieler funktionieren, dabei wird der Bildschirm auf die Spieler verteilt. Wer alleine spielt bekommt den ganzen Bildschirm für sich, bei 2 Spielern bekommt jeder eine Hälfte, bei 3–4 Spielern bekommt jeder ein Viertel des Bildschirms. Hierzu erscheint zu Beginn des Spiels ein Menü, in dem die Spieleranzahl und die gewünschte Spielfigure ausgewählt werden kann. Die Steuerung der Spielfiguren ist einfach gehalten. Die Figuren laufen von selbst durch die Welt, die sich von rechts nach links durchs Bild bewegt, dem Spieler erscheint es also so, als würde sich seine Figur konstant nach rechts bewegen. In der Welt sind verschiedene Hindernisse platziert, über die der Spieler springen muss oder sich unter ihnen hindurch ducken muss. Das passiert durch eine einfache Berührung an einer von 2 Stellen auf seinem Teil des Bildschirms, die symbolisch gekennzeichnet sind. Wer eines der Hindernisse berührt, scheidet aus, die anderen Spieler spielen weiter. Dabei wird jedoch auch stets geprüft, ob ein Spieler den Bildschirm mehr als einmal berührt, denn wer beispielsweise alle 10 Finger auf den Bildschirm legt, könnte so seinen Mitspielern die Berührungspunkte und damit die Kontrolle über ihre

Figuren wegnehmen. Am Ende heißt es dann: wer schafft es am weitesten, ohne ein Hindernis zu berühren? Wer am längsten durchhält wird zum Sieger erklärt. Die besten Ergebnisse werden in einer Statistik aufgelistet.

Weitere Spielmöglichkeiten wären mit maximal 3 Spielern denkbar. Dann wäre abgesehen von springen und ducken auch eine eigenständige Steuerung der Seitwärtsbewegung realisierbar, da hierfür jeder Spieler nur 2 Berührungspunkte benötigt, was bei den maximalen 6 Berührungspunkten des Touch-Tables gerade aufgeht. Beispielsweise könnte man dann einen Spielmodus einführen, in dem alle Spielfiguren auf dem ganzen Bildschirm zusammen durch die Welt laufen und sich gegenseitig beim Erreichen des Ziels helfen oder behindern können. In diesem Spielmodus könnten dann auch feindliche Computergegner im Spiel eingesetzt werden, die sich nicht konstant durch das Bild bewegen, sondern aktiv versuchen, die Spieler zu berühren und damit aus dem Spiel ausscheiden zu lassen. Hierfür müssten die Steuerflächen aller Spieler zum unteren Bildschirmrand wandern und werden im Gegensatz zum anderen Spielmodus nicht gleichmäßig auf die Fläche des Touch-Tables verteilt, was bei maximal 3 Spielern jedoch keine Platzprobleme zur Folge haben sollte.

[1] Unity Technologies: Unity Documentation. <https://docs.unity3d.com/Manual> (2017)

Bildquellen:

- Abbildung 1: http://www.marveltechgroup.com/product/Touch-Table_1
- Abbildung 2: docs.unity3d.com/430/Documentation/Components/class-SkinnedMeshRenderer.html

Entwicklung einer Vorgehensweise zur Testautomatisierung der Mercedes me-App mit Hilfe von HP Mobile Center und HP LeanFT

Teodora Todorova^{*}, Thomas Rodach, Catharina Kriegbaum-Kling

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Aufgrund der fortschreitenden Digitalisierung im Bereich der Automobilindustrie werden immer mehr Apps angeboten, die einen Remotezugriff auf Fahrzeuge ermöglichen.

Die Daimler AG bietet die Mercedes me-App an, die unter anderem einen Zugang zu personalisierten Mobilitäts-, Konnektivitäts- und Finanzdienstleistungen ermöglicht.

Dieser Artikel befasst sich mit den Grundlagen der Testautomatisierung für die Mercedes me-App.

Warum wird getestet

Die Mercedes me-App hat im Laufe der Zeit einen großen Funktionsumfang erhalten. Um die Funktionalität, Qualität und Benutzerfreundlichkeit der App zu überprüfen und eine Beurteilung der Softwareeife für den Go-Live neuer Versionen zu ermöglichen muss diese in regelmäßigen Abständen getestet werden.

Motivation

Zwei der größten Herausforderungen in der Schnittstelle zur Software-Entwicklung sind die Umsetzungs-Geschwindigkeit und Besonderheiten der agilen Entwicklungsmethoden. Die Herausforderung der manuellen Tests bei Mercedes me connect besteht darin, mögliche Fehler im Rahmen der Abnahmetests auf der Ende-Zu-Ende-Teststufe zu finden. Da der manuelle Test aufgrund des hohen Funktionsumfangs der App sehr zeitaufwendig ist, wird der Einsatz automatisierter Tests benötigt, damit sich die Testzeiträume verkürzen und die Kosten für das Testen nicht linear mit dem Funktionsumfang ansteigen.

Vorteile der Testautomatisierung

Durch die Verwendung automatisierter Skripte kann man Veränderungen in der App identifizieren, um mögliche Fehler aufzudecken. Automatisierte Tests bieten außerdem die Möglichkeit, die Kundenanforderungen an die App rechtzeitig zu überprüfen und wenn die Anforderungen nicht erfüllt worden sind, Fehlerursachen früh aufzuspüren.

Die Testautomatisierung der App lohnt sich langfristig gesehen, weil sie ohne großen Aufwand an verschiedene Märkte angepasst werden kann. Zudem kann die Testautomatisierung auch in der Entwicklung eingesetzt werden, um Fehler bereits in der Teststufe Systemtest zu identifizieren und damit den Aufwand im E2E-Abnahmetest zu reduzieren.

Außerdem können automatisierte Testskripte:

- häufiger und schneller ausgeführt werden,
- mit unterschiedlichen Testdaten ausgeführt werden,
- den manuellen Testaufwand reduzieren,
- mit anderen Tools kombiniert werden.

HP Mobile Center

HP Mobile Center ist eine Testumgebung, die verschiedene Aktionen auf realen Mobile Devices simulieren kann (Abbildung 1), um die Verfügbarkeit und Leistung einer Anwendung zu gewährleisten. Die Vorteile von HP Mobile Center sind [1]:

- Ausführung paralleler Prozesse
- Testen mehrerer Apps in einem einzigen Testlauf
- iOS Automatisierung ohne Macbook
- volle Kontrolle über das Mobile Device
- Reservierung von Geräten
- Security Scans

^{*}Diese Arbeit wurde durchgeführt bei der Firma Daimler TSS GmbH, Leinfelden-Echterdingen



Abbildung 1: iOS Gerät mit installiertem Mobile Center

HP LeanFT

HP LeanFT ist ein Tool, das die Entwicklung von stabilen, automatisierten Tests ermöglicht. Die Änderungen in einer App können mithilfe von LeanFT schnell erkannt und durch die Reporting-Funktion übersichtlich dargestellt werden.

Für die Testautomatisierung der App ist auch die Kombinierbarkeit mit anderen Tools von Bedeutung (Abbildung 2). In Kombination mit verschiedenen Tools lässt sich das Testframework optimieren.



Abbildung 2: LeanFT Integration mit anderen Tools

Testkonzept

Für die Machbarkeitsstudie wurden Testfälle aus der Regression ausgewählt. Dies sind u.A. Testfälle, die Grundfunktionalitäten der App (z.B. „Login“) abbilden. Die einzelnen Testfälle wurden in sogenannte Features aufgeteilt. Bereits erstellte Features können für beliebig viele Testfälle verwendet werden. Testskripte wurden mithilfe von HP Mobile Center einmalig manuell aufgezeichnet und in Eclipse mit der Programmiersprache Java parametrisiert. Für das Testframework wurde das PageObject-Pattern verwendet. Dieses stellt sicher, dass jede App-Seite eine eigene Java-Klasse besitzt. Damit wird sowohl die Übersichtlichkeit als auch eine gute Struktur des Testframeworks gewährleistet. Durch das Pattern wurden Wiederholungen im Code reduziert, weil die Features ausschließlich einmal in der Testklasse existieren und mehrmals in verschiedenen Testfällen benutzt werden können. Dies erleichtert außerdem die Anpassung von Veränderungen am Code.

Ausblick

Im weiteren Verlauf dieser Arbeit sollen neben den repräsentativen Testfällen weitere funktionale Testvorgänge automatisiert werden.

Außerdem soll die Anpassbarkeit des Testframeworks an unterschiedliche Märkte untersucht werden. Das Ziel ist es zu zeigen, dass Testergebnisse im Vergleich zu manuellen Tests schneller verfügbar sind.

[1] Martin, D. HPE Mobile Center im Einsatz. 2017. <https://www.proficom.de/unternehmen/aktuelles/artikel/hpe-mobile-center-im-einsatz.html> (Zugriff am 16. Oktober 2017)

Bildquellen:

- Abbildung 1: Eigene Darstellung
- Abbildung 2: <http://automated-360.com/devops/leanft-new-monster-automation-town/> (letzter Zugriff am 16. Oktober 2017)

Algorithmen zur Fahrspurerkennung bei autonom fahrenden Modellfahrzeugen mit Hilfe von maschinellem Sehen und OpenCV

Manuel Tutsch*, Jürgen Koch, Reiner Marchthaler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Ein modernes Fahrzeug ohne integrierte Computer ist in der aktuellen Zeit nicht mehr vorstellbar. Sei es für unterstützende Fahrassistenzsysteme oder komplett autonom fahrende Verkehrsmittel. Spätestens als Tesla seine pilotierten Fahrzeuge auf den Markt brachte, wurde klar, dass sich die Technologie insbesondere im Bezug zu autonomen Fahrzeugen rasant weiterentwickelt. Eine der wichtigsten Aufgaben eines autonomen Fahrzeuges ist es, die Umgebung zu erfassen und daraus Rückschlüsse für das geplante Fahrverhalten zu ziehen. Dabei ist es nicht nur relevant zu wissen, wo sich das Fahrzeug auf der Fahrbahn befindet, sondern auch, wie sich der Fahrbahnverlauf ändert und ob es zu beachtende Hindernisse gibt. Während zur Umwelterfassung sehr häufig Sensorsysteme wie Radar oder Lidar eingesetzt werden, liefern die Hauptinformationen für das Umgebungsbild und die Fahrspuren jedoch Kameras. Je nach Ansatz werden verschiedene Arten und Objektive von Kameras verwendet. Tesla nutzt für die vorwärts gerichteten Kameras zum Beispiel eine Hauptfeldkamera, sowie eine mit Teleskopobjektiv für weite Entfernungen. Abbildung 1 zeigt die in den Fahrzeugen von Tesla verbauten Kameras, mit welchen die komplette Umgebungserfassung erfolgt.

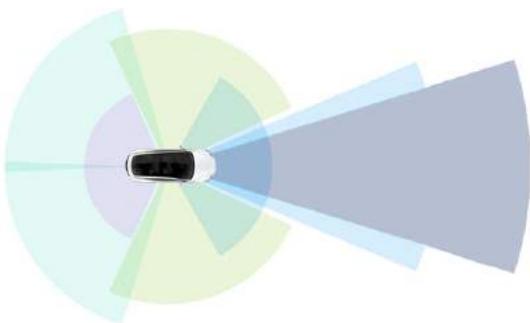


Abbildung 1: Kameras für die Umgebungserfassung in Tesla Modellen

Der erste Schritt um Fahrspuren in einem Bild erkennen zu können, ist die **Kantenerkennung**. Kanten bieten für die Analyse von Bildern eine hohe Informationsquel-

le, da mit ihnen Strukturen und Konturen erfasst und somit Markierungen auf der Fahrbahn erkannt werden können. In einem Bild stellen sich Kanten durch einen Intensitätsanstieg der Bildfunktion dar. Um diese zu lokalisieren kann man die Bildfunktion ableiten. Beim einmaligen Ableiten findet man Intensitätsanstiege durch das Suchen der Extremwerte der Ableitungsfunktion, beim zweimaligen Ableiten hat die Funktion an den Stellen Nulldurchgänge, bei welchen sich eine Kante befindet. Abbildung 2 zeigt eine eindimensionale Grauwertkante mit deren ersten und zweiten Ableitung. Ein bewährter Algorithmus zum Finden der Kantenpixel ist der Canny-Algorithmus. Mittels Faltung des Bildes mit einem Sobel-Operator wird die erste Ableitung angenähert und durch weitere Operationen anschließend Nicht-Kantenpixel aussortiert. Als Ergebnis erhält man ein binäres Bild, welches die gefundenen Kanten von einem Pixel breite enthält.

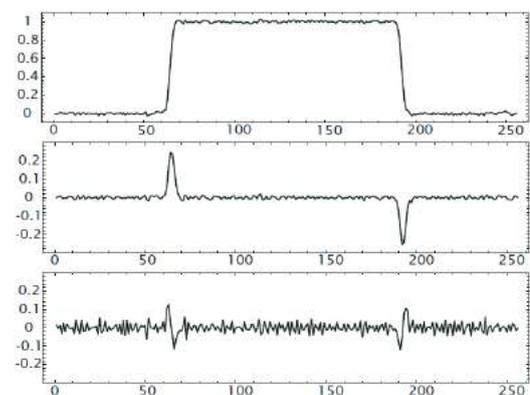


Abbildung 2: Grauwertkante mit erster und zweiter Ableitung

Die **Hough-Transformation** ist ein Verfahren zum Finden von Geraden in einer Menge von Punkten. Alle gefundenen Kantenpixel aus dem Bild werden mittels einer Transformation in den Houghraum abgebildet, wo sie als Sinusfunktion dargestellt werden. Schneiden sich im Houghraum die Sinuskurven, bedeutet dies, dass die Kantenpunkte im Bild auf

*Diese Arbeit wurde durchgeführt bei der Firma IT-Designers GmbH, Esslingen

einer Geraden liegen. Es werden nun diejenigen Schnittpunkte im Houghraum gesucht, durch welche die meisten Sinuskurven gehen. Bei einer Rücktransformation dieses Punktes in den Bildraum ist dies dann die Gerade, auf welcher die meisten Punkte liegen. Mittels dieser Transformation können zum einen gerade Fahrbahnmarkierungen erkannt werden, zum anderen dienen diese Geraden dazu Fluchtpunkte im Bild zu bestimmen, welche bei Krümmungsberechnungen genutzt werden können [1].

Krümmungsschätzung durch Parabeln

Aus den gewonnenen Kanteninformationen kann ein Modell gebildet werden, das den Fahrspurverlauf möglichst genau schätzt. Dazu werden häufig, unter der Annahme, dass die Fahrbahn eben ist, Krümmungsschätzungen in horizontaler Richtung durchgeführt. Eine einfache Form dessen ist die Repräsentation der Fahrbahn durch Parabeln im Fernbereich des Fahrzeuges [2]. Für jede Fahrbahnmarkierung wird für alle Kantenpixel jeweils eine quadratische Gleichung aufgestellt und das daraus resultierende Gleichungssystem gelöst. Die dadurch erhaltenen Koeffizienten der Parabel bieten dann einen Schätzwert für die Krümmung der Kurve. Im Nahbereich vor dem Fahrzeug kann in den meisten Fällen die Fahrbahn durch Geraden geschätzt werden. Dies kann zum Beispiel durch die Hough-Transformation erfolgen. Abbildung 3 zeigt die erkannte Fahrspur sowie den Fluchtpunkt der Hough-Linien eines Kamerabildes.

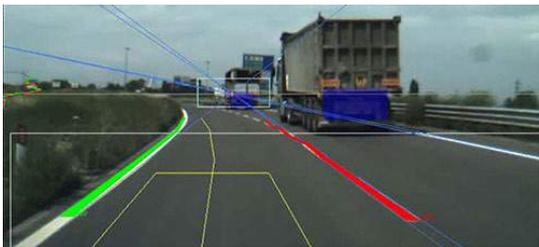


Abbildung 3: Fahrspurerkennung in einem Kamerabild

Krümmungsschätzung durch B-Splines

Splines sind Kurven, die aus mehreren Stücken von Polynomen von höchstens k -ten Grades zusammengesetzt sind. Sie werden häufig für Interpolation und Näherungen verwendet, um möglichst glatte Kurven zu erhalten. Für die Stellen, an denen sich zwei Polynomstücke treffen, sogenannten Knoten, werden zusätzliche Bedingungen gestellt. So muss der Spline an diesen Knoten zum Beispiel bis zu einem bestimmten Grad differenzierbar sein [3]. Da die Funktion und die Ableitungen dann stetig sind, gibt es zwischen den Polynomen keine Sprünge und damit keine sprunghaften hektischen Lenkbewegungen im Fahrzeug. Mit Hilfe dieser Technik lässt sich im Gegensatz zu Parabel-Modellen außerdem eine größere Menge an Fahrspurmodellen abschätzen, da nun für jeden Kurvenabschnitt ein passendes Polynom gewählt werden kann. Als Beispiel lassen sich dadurch die in Abbildung 4 gezeigten Kurven berechnen.

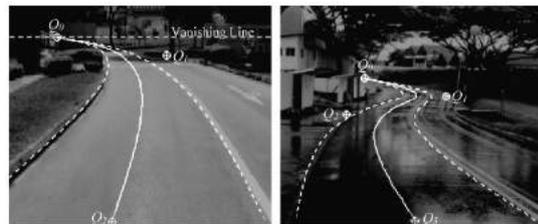


Abbildung 4: Krümmungsschätzung des Fahrbahnverlaufs durch B-Splines

Durch die Position des Fahrzeuges und dem Wissen, wo die Fahrbahn seitlich endet und wie der weitere Verlauf dieser ist, können nun Informationen für weitere Quer- und Längssteueranweisungen gewonnen werden. Die Fahrspurerkennung ist für das autonome Fahren daher unerlässlich und lässt sich in der heutigen Zeit durch die rasante Entwicklung im Soft- und Hardwarebereich schnell und einfach realisieren. Im Rahmen dieser Bachelorarbeit soll ein autonom fahrendes Modellfahrzeug entwickelt werden, welches die Fahrspur durch eine Monokamera ermittelt.

-
- [1] B. Yu und A. K. Jain. Lane boundary detection using a multiresolution houghtransform. In Proceedings of International Conference on Image Processing, volume 2, pages 748–751 vol.2, 1997.
 [2] C. R. Jung and C. R. Kelber. A robust linear-parabolic model for lane following. In Proceedings. 17th Brazilian Symposium on Computer Graphics and Image Processing, pages 72–79, 2004.
 [3] Hartmut Prautzsch und Wolfgang Boehm und Marco Paluszny. Bezier and BSpline Techniques. Springer-Verlag Berlin Heidelberg GmbH, 2002.

Bildquellen:

- Abbildung 1: https://www.tesla.com/de_DE/autopilot
- Abbildung 2: Bernd Jähne. Digital Image Processing, 6th revised and extended edition. Springer, 2005.
- Abbildung 3: <https://www.ptgrey.com/case-study/id/10393>
- Abbildung 4: Yue Wang und Eam Khwang Teoh und Dinggang Shen. Lane detection and tracking using b-snake. 22:269–280, 04 2004.

Vergleich von Anomalie Erkennungssoftware

Thuy Vi Uong*, Gabriele Gühring, Dirk Hesse

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Big Data ist sicher eines der größten Schlagworte in der IT heutzutage, denn Daten gibt es überall. Daten werden aus den verschiedensten Themenbereichen mit speziellen Lösungen extrahiert, gespeichert, verarbeitet und ausgewertet. Die Problematik hierbei ist die riesige Menge an Datenvolumen und ihre mit sich bringende Anomalien [1]. Durch das ständige Wachstum von Daten, steigt auch das Risiko von immer mehr werdenden Ausreißern, den „Outliern“.

Nun stellt sich die Frage, was genau Anomalien sind und wie sie zu beheben sind. Dafür gibt es zahlreiche Anomalie Erkennungssoftwares. Das Ziel ist es die Erkennungssoftwares zu analysieren und miteinander zu vergleichen. Doch zunächst wird geklärt, was Anomalien sind.

Was sind Anomalien, bzw. Outlier?

Hawkings definierte Outlier wie folgt: “An outlier is an observation which deviates so much from the other observations as to arouse suspicions that it was generated by a different mechanism.” [2]. Outliers, die auch unter Abnormalitäten, Anomalien und Diskordanten bekannt sind, sind Daten bzw. Datenpunkte, die sich anders als alle anderen Daten in einem Datensatz verhalten, und somit direkt auffallen.

Sie liegen außerhalb einer vom Benutzer definierten Grenze. Der Benutzer entscheidet selbstständig, ab welcher Abweichung Daten als Ausreißer definiert werden sollen. Dabei werden die Ausreißer in zwei Kategorien unterteilt. Die erste Ausreißer Kategorie beinhaltet Schreibfehler oder Extremwerte. Diese beiden Anomalien treten auf, wenn sich die Werte außerhalb der validen Grenze aufhalten. Die zweite Ausreißer Kategorie betrifft Daten, die sich ebenfalls nicht mehr in dem validen Bereich aufhalten. Der unterschied hier ist, dass es einen mehrdimensionalen Zusammenhang zwischen den Werten gibt [3].

Häufig werden die Datensätze mit Hilfe von Prozessen zur Datengenerierung erfasst. Meist werden folgende Prozesse angewendet, um die Daten zu erhalten. Die erste Möglichkeit wäre, die gesammelten Entitäten, die durch Observationen entstanden sind oder die verschiedenen Aktivitäten, welche mit dem System durchgeführt wurden. Tritt ein außergewöhnliches Verhalten während des Datengenerierungsprozess auf, werden diese Daten als Anomalien identifiziert.

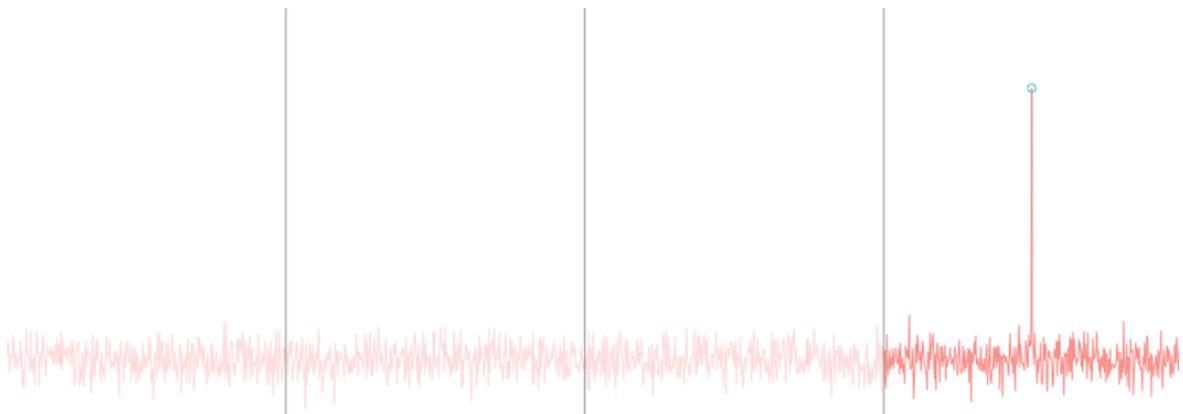


Abbildung 1: Lokaler Ausreißer

*Diese Arbeit wurde durchgeführt an der Fakultät Informationstechnik

Anomalien können auch durch inkorrekte Eingabe der Daten oder auch durch eine fehlerhafte Durchführung eines beispielsweise Tests entstehen. Falls der Anomalie einen Fehler zugewiesen werden kann, sollte dieser Fehler im besten Fall korrigiert werden, ansonsten sollte der Ausreißer entfernt werden [3].

Die Anomaliedetektion kann auf viele verschiedene Datentypen angewendet werden. Diese betreffen hochdimensionale Daten, Stromdaten, Netzwerkdaten, Zeitreihendaten, räumliche Daten und räumlich-Zeitlicher Daten und unsichere Daten.

Anomalien sind für das System von hoher Bedeutung, denn sie können wichtige Informationen beinhalten. Da sich Anomalien durch Unstimmigkeiten im System kennzeichnen, können anhand der Anomalien Ursachen für diese auffälligen Verhalte ermittelt werden, die möglicherweise das System und die Entitäten bei der Datengenerierung beeinträchtigen.

Wie nützlich Anomalie seien können, lässt sich unter folgenden diversen Anwendungsbereiche aufzeigen.

Anwendungsgebiete

- **Kreditkartenbetrug:** Heutzutage ist der Betrug mit Kreditkarten kein seltenes Thema mehr. Die fortgeschrittene Informatik ermöglicht es Betrügern Kreditkarten anhand einfacher Informationen, wie z.B. der Kreditkartennummer zu kompromittieren. Durch die ungewöhnlichen Kaufmuster bzw. Kaufverhalten, kann ein Kreditkarten-Missbrauch ermittelt werden.
- **Entdeckung von Messfehlern:** Auffällige Messwerte die beispielsweise durch Sensoren aufgenommen wurden, können auch Fehler enthalten. Dabei können die Sensoren zu unterschiedlichsten Zwecken eingesetzt werden. Am häufigsten werden sie eingesetzt, um Umwelt- und Standortparameter zu sammeln.
- **Medizin:** Gesundheitliche Probleme können anhand von außergewöhnlichen Testergebnissen oder Symptomen diagnostiziert werden. Diese Testergebnisse spiegeln ein ungewöhnliches Muster wieder.
- **Sport:** Auch im Sport werden Statistiken geführt, um sich die Leistungen der jeweiligen Sportler zu verfolgen. Durch die ungewöhnlichen Werte, lassen sich dabei besonders gute wie auch die etwas schwächeren Sportler rausfiltern.
- **Angriffserkennungssysteme:** Der Computer ist voll mit Daten, die beispielsweise bei Betriebssystemaufruf, Netzwerkverkehr oder andere Aktionen entstehen. „Intrusion detection“ wird der Vorgang, welcher ungewöhnliche Muster entdeckt, die durch böses Verhalten ausgelöst werden, bezeichnet.
- **Strafverfolgung:** In der Straferkennung kann die Anomaliedetection auch sehr von Nutzen sein. Durch das wiederholte, auffällige Verhalten einer Entität, können Betrugsfälle in Finanzaktionen, Handelsaktivitäten und Versicherungsansprüchen aufgedeckt werden.
- **Geowissenschaften:** Mit Hilfe von Satelliten oder Fernerkundungen, werden Daten aufgelesen. Diese Daten sagen einiges über Klimawandel, Wettermuster und Landbedeckungsmuster aus. Entdeckte Anomalien, sagen hier einiges über das menschliche Verhalten oder die Umweltrends aus, die möglicherweise Grund der Anomalie seien können. [4]

Fazit

Anomalie Detektion ist besonders wichtig geworden, da Fehler gemacht werden und es daher umso wichtiger ist, die Ausreißer zu erkennen, zu wissen woher sie stammen und sie zu beseitigen. Deshalb ist die Anomalie Erkennung vor allem in den Bereichen Big Data und Data Mining wichtig.

[1] Vgl. Bendel, Prof. Dr. Oliver. 2017. Gabler Wirtschaftslexikon. [Online] Springer Gabler Verlag, 18. November 2017. [Zitat vom: 24. November 2017.]

<http://wirtschaftslexikon.gabler.de/Archiv/-2046774198/big-data-v5.html>.

[2] Hawkins, D. 1980. Identification of Outliers. s.l. : Springer Netherlands, 1980. ISBN 978-94-015-3994-4.

[3] Vgl. NIST SEMATECH. 2013. [Online] Engineering statistics handbook, 30. Oktober 2013. [Zitat vom: 24. November 2017.] <http://www.itl.nist.gov/div898/handbook/index.htm>.

[4] Vgl. Aggarwal, Charu C. 2013. Outlier Analysis. New York : Springer New York Heidelberg Dordrecht London, 2013. ISBN 9781461463955.

Bildquellen:

- Abbildung 1: <https://anomaly.io/anomaly-detection-twitter-r>

Evaluation und Optimierung eines Continuous Delivery Prozesses

Fabian Weber*, Thomas Rodach, Catharina Kriegbaum-Kling

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Was ist DevOps ?

Das Wort DevOps ist ein Kunstwort und setzt sich aus „development“ (Entwicklung) und „operation“ (Betrieb) zusammen. Historisch gesehen sind, vor allem in großen Unternehmen, diese zwei Bereiche voneinander getrennt. Dabei entstand oft die sogenannte „Wall of Confusion“ welche den Konflikt, während der Auslieferung einer Anwendung, zwischen diesen zwei Parteien beschreibt. Oft ist ein langwieriger Prozess entstanden bei dem sich zwei Bereiche die Schuld, beim Auftreten eines Fehlers, gegenseitig in die Schuhe geschoben haben. Die Ziele der beiden Bereiche beeinflussen sich gegenseitig negativ. Ziel der Entwicklung ist es möglichst schnell neue Features / Code dem Kunden bereit zu stellen während der Betrieb auf die Stabilität der Anwendung Wert legt.

DevOps bezeichnet eine Reihe von Vorgehensweisen zur Automatisierung des Auslieferungsprozesses in der Softwarebereitstellung. Dadurch kann die Software schneller entwickelt, getestet und freigegeben werden wobei die Qualität des Codes nicht darunter leiden darf. Die Basis auf der DevOps aufbaut ist die

Zusammenarbeit bzw. das Vertrauen zwischen den an dem Prozess beteiligten Parteien. Durch verschiedene Tools kann ein hoher Grad an Automatisierung erreicht werden, der für die Umsetzung von DevOps förderlich ist. Allerdings sollte DevOps nicht mit der Automatisierung durch Tools gleichgesetzt werden. Des Weiteren vereint DevOps das Beste aus der Welt der Entwicklung und des Betriebs und bringt somit die Agilität die sich über die letzten Jahre in der Entwicklung etabliert hat in den Betrieb. Dieser wurde als Flaschenhals angesehen da hier viele Schritte noch manuell und statisch ausgeführt worden sind. Mit neuen Tools wie Docker oder Chef soll dieses Problem gelöst werden [1].

Continuous Delivery und DevOps

Continuous Delivery ist besonders gut umsetzbar im Rahmen einer DevOps Kultur. Oft werden diese sogar gleichgesetzt. Das ist allerdings nicht der Grundgedanke des Ganzen. DevOps hebt das Silo denken auf und wandelt die Unternehmenskultur. Der Unterschied zu Continuous Delivery liegt darin, dass hierbei der Betrieb in den Bereitstellungsprozess mit einbezogen wird (siehe Abbildung 1).

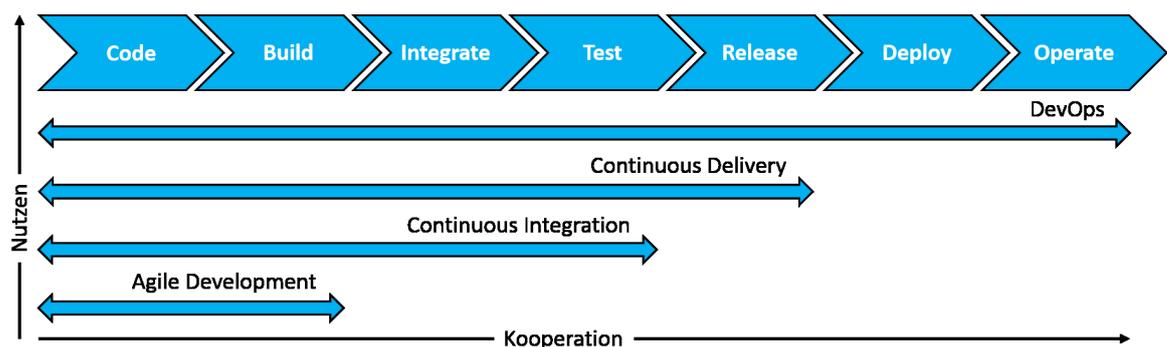


Abbildung 1: Integration von DevOps über den Bereitstellungszyklus

*Diese Arbeit wurde durchgeführt bei der Firma Robert Bosch GmbH, Feuerbach

Die Problemstellung

Die Bachelorarbeit beschäftigt sich mit der Evaluation und Optimierung des Continuous Delivery Prozesses in einem Softwareteam der Robert Bosch GmbH.

Das Team entwickelt und betreut ein Controlling System für den zentralen Einkauf des Unternehmens mit dessen Hilfe die Unternehmensweite Preisplanung für direktes Material gemacht wird.

Im Rahmen dieses Projektes werden die Auswirkungen von DevOps auf die Bereitstellungs- und Entwicklungsprozesse analysiert. Zudem wird die Unterstützung durch mögliche Tools bewertet und getestet.

Als erstes werden die vorhandenen Prozesse dokumentiert und anschließend bewertet. Nach der Identifikation von Optimierungspotenzial wird anhand eines Beispielprojektes die Umsetzung von ausgewählten Automatisierungstools aufgezeigt.

Bisher ermittelte Optimierungspotenziale sind unter anderem bei der Bereitstellung von Artefakten für die einzelnen Umgebungen gefunden worden, sowie bei der Versionierung der Datenbank. Dabei gibt es ein breites Angebot an Tools die versprechen diese Verbesserungspotenziale umzusetzen. Die Herausforderung dabei ist es das passende Tool für jeden dieser Optimierungspotenziale zu finden.



Abbildung 2: Beliebte Tools im DevOps Umfeld

Ausblick

Durch die Implementierung in einem Beispielprojekt kann konzeptionell aufgezeigt werden, nach der Marktanalyse, wie das ausgewählte Tool zur Optimierung eines Prozesses einbinden lässt. Nach Beendigung dieser Arbeit können die gewonnenen Erkenntnisse von dem Team in die Anwendung integriert werden. Ein weiterer Ansatzpunkt ist eine Studie die bis Ende November bei der Robert Bosch GmbH

durchgeführt worden ist. Hierbei wird die Anwendung mit einer zweiten zusammengeführt. Dafür wird eine neue IT-Infrastruktur aufgebaut in der sich neueartige Tools und Techniken z.B. Docker oder Microservices implementieren lassen. Dazu bedarf es Untersuchungen und Bewertungen die im voraus durchgeführt werden. Hierbei können diese auch in bezug auf Continuous Delivery bzw. DevOps bewertet werden.

[1] Atlassian: Was ist DevOps? <https://de.atlassian.com/devops> (23.11.2017)

Bildquellen:

- Abbildung 1: <http://security-architect.com/discovering-agile-cloud-security/>
- Abbildung 2: <https://pocfarm.wordpress.com/2016/03/31/popular-devops-tools/>

Evaluierung der Integration des PPM-Protokolls in den Engineering-Prozess der OPCON Automatisierungslösungen von Bosch für Anwendungen mit Bezug auf Industrie 4.0

Paris Wegener*, Reinhard Keller, Dominik Schoop

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Die vierte industrielle Revolution, kurz auch Industrie 4.0 genannt, in der sich die Produktion mit modernster Informations- und Kommunikationstechnik verzahnt, hat längst begonnen. Durch intelligente digital vernetzte Systeme soll dabei eine weitestgehend selbstorganisierte Produktion möglich werden [1]. Eine wichtige Rolle spielt dabei das von der OPC Foundation spezifizierte Protokoll OPC UA. „In der OPC Foundation haben sich führende Steuerungshersteller und Softwareproduzenten zusammengeschlossen, um OPC immer auf dem Stand der Technik zu halten, insofern ist es nur folgerichtig, dass OPC UA auch als Standard für die Industrie 4.0 hoch gehandelt wird“ [2]. Eine wichtige Eigenschaft von OPC UA ist es, Daten in ihrer semantischen Struktur beliebig beschreiben zu können. Dies hat jedoch auch zur Folge, dass unterschiedliche Hersteller ihre Daten auch unterschiedlich strukturieren. An diesem Punkt setzt das PPM-Protokoll an, welches die semantische Struktur von Daten spezifiziert, um die Überwachung und Datenanalyse von Anlagen herstellerübergreifend zu ermöglichen.

Aufgabenstellung

Im Rahmen der vorliegenden Arbeit wurde untersucht, inwiefern sich das Production Performance Management Protocol (PPMP) in den Engineering-Prozess der Automatisierungsplattform OPCON Control plus von Bosch integrieren lässt. Dazu wurde zunächst das PPM-Protokoll und der Production Performance Manager (PPM) untersucht, um anschließend eine Gegenüberstellung des PPM-Protokolls und der bereits bestehenden proprietären OPC-UA-Schnittstelle der OPCON-Plattform anfertigen zu können. Unter Berücksichtigung des Engineering-Prozesses war es das Ziel, einen Prototyp zu entwickeln, der als Windows-Service zu Verfügung stehen sollte und als Gateway zwischen der OPC-UA-Schnittstelle des OPCON-Systems und dem PPM-Protokoll angesehen werden kann.

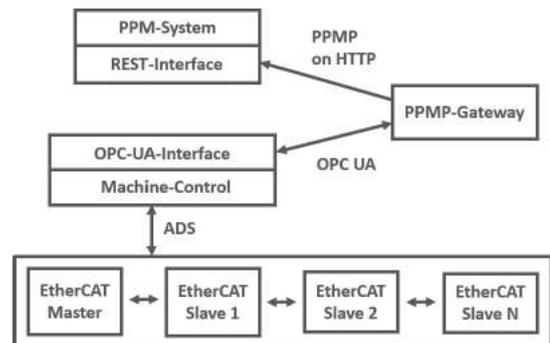


Abbildung 1: Systemkontext des PPMP-Gateways

PPM-Protokoll und PPM-System

Das PPM-Protokoll ist ein von der Robert Bosch GmbH spezifiziertes Protokoll, das mittlerweile als Open-Source-Projekt bei der Eclipse Foundation verfügbar ist. Dabei konzentriert sich dieses auf die semantische Darstellung von Maschinendaten, die im JSON-Format über unterschiedliche Transportprotokolle wie HTTP, MQTT oder AMQP übertragen werden können. Grundsätzlich spezifiziert das PPM-Protokoll drei unterschiedliche Nachrichtentypen für Maschinen-, Messwert- und Prozessdaten. Jede dieser Nachrichten besteht dabei aus einer Vielzahl von Entitäten. Die Spezifikation legt darüber hinaus auch fest, welche dieser Entitäten und ihrer Eigenschaften zwingend erforderlich sind und welche optional genutzt werden können. So ist bspw. das Attribut deviceID der Device Entität zwingend erforderlich, da für diesen Wert ein Globally Unique Identifier (GUID) vorgesehen ist, über den die Zuordnung im Backend stattfindet. Das PPM-System ist ein Backend-System von Bosch, das Daten einer Maschine über das PPM-Protokoll aufnehmen kann. Neben der Funktionalität zur Datenauswertung und Darstellung enthält es auch ein Modul zur Verbesserung der Qualität von Prozessen und ein Ticket-Management-System. Auf das PPM-System kann über ein Webinterface, über mobile Apps und ein Application Programming Interface (API) zuge-

*Diese Arbeit wurde durchgeführt bei der Firma Robert Bosch Manufacturing Solutions GmbH, Stuttgart

griffen werden. Diese API ist dabei eine HTTP-basierte REST-Schnittstelle. Für die vorliegende Arbeit wurde das Webinterface genutzt, um den Aufbau des Systems zu untersuchen und um zu verifizieren, ob gesendete Daten auch angekommen sind. Darüber hinaus war die API von zentralem Interesse, da das PPMP-Gateway auch befähigt sein sollte, ein Backend-System wie das PPM konfigurieren zu können. Dies bedeutet, dass die Stationshierarchie einer Anlage zuerst im PPMP-Gateway abgebildet werden muss, um diese dann im PPM abbilden zu können.

Das OPC-UA-Interface von OPCON

Das OPC-UA-Interface ist eine einheitliche Bosch-interne Schnittstelle, über welche Informationen einer Anlage ausgelesen werden können. Diese enthält Definitionen, wie Daten repräsentiert werden, sogenannte Profildefinitionen. Ein Bearbeitungsprozess besitzt z.B. ein definiertes Profil über welches Informationen und Ergebnisse des Prozesses über OPC UA dargestellt werden. Das Open Platform Communications Unified Architecture (OPC UA) Protokoll ist ein plattformübergreifendes zustandsbasiertes M2M-Protokoll, welches das Client/Server-Prinzip verwendet. Ein Client kann Daten dabei über ein Pull/Push-Mechanismus lesen bzw. schreiben oder über das Publish/Subscribe-Pattern bei Wertänderungen informiert werden. Eine Besonderheit von OPC UA ist dabei, dass die ankommenden Daten Meta-Informationen bereitstellen. So können im Eventhandler viele zusätzliche Informationen aus den so genannten DataChangedEventArgs ausgelesen werden. Dies sind neben dem eigentlichen Wert, Informationen wie z.B. Zeitstempel, Server-Adresse oder Nodename. Die Nodename ist dabei ein eindeutiger Endpunkt auf einem OPC-UA-Server.

OPCON Engineering System

Das OPCON Engineering System (OES) ist ein von Bosch entwickeltes Software-Tool das ermöglicht nach dem Baukastenprinzip Software für Sondermaschinen zu erstellen. Es setzt dabei auf Echtzeit-SPS-Systeme auf und erweitert diese um Baukastendenken, Wieder-

verwendbarkeit, Maschinenkonzepte, strukturiertes Datenmanagement, Diagnosekonzepte und eine einheitliche Anlagenvisualisierung. Die Baukastenelemente werden in den Anlagenprojekten als Interfaces referenziert, so dass bspw. ein Objekt eines Antriebs leicht durch einen Antrieb eines anderen Herstellers ausgetauscht werden kann [3]. Weitere integrierbare Erweiterungen können Taktzeit- oder Verbrauchsanalysen, historische Produktionsdaten oder Schnittstellen zu anderen Systemen sein. Auch das PPMP-Gateway stellt in dieser Architektur eine Erweiterung dar. Das OES ist dabei das Herz aller Informationen für die Projekterstellung. Es generiert eine Vielzahl von Konfigurationsdateien, unter denen sich auch die XML-Konfigurationsdatei des OPC-UA-Interfaces von OPCON befindet. Diese wird vom PPMP-Gateway genutzt, da sie Informationen über die Stationshierarchie sowie die über OPC UA freigegebenen Daten enthält.

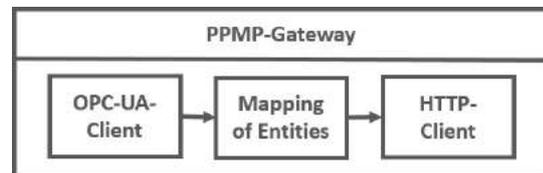


Abbildung 2: Funktionsprinzip des PPMP-Gateways

Entwicklung des Prototyps

Der Prototyp wurde in C# und gegen die .NET Standard 2.0 Spezifikation implementiert, welche eine Schnittmenge von .NET-Framework, .NET-Core, Mono und Xamarin darstellt. Dadurch kann die Applikation sowohl in einem klassischen Windows-Service als auch auf Linux und anderen Betriebssystemen genutzt werden [4]. Auch Szenarien in einem Docker Container auf unterschiedlichen Plattformen oder in der Cloud sind somit möglich. Die Architektur des PPMP-Gateways nutzt Inversion of Control und Dependency Injection. Durch diese Architektur ist es leicht möglich die Anwendung zu erweitern, um so bspw. weitere PPMP-Backend-Systeme oder Transportprotokolle wie MQTT oder AMQP zu unterstützen.

[1] Plattform Industrie 4.0: Was ist Industrie 4.0, <http://www.plattform-i40.de>

[2] Hanjo Schlüter, Inray Industriesoftware <https://www.produktion.de/trends-innovationen/opc-ua-wird-neben-anderen-industrie-4-0-standard-334.html>

[3] Robert Bosch GmbH: Schulungsunterlage Op120 OpCon Control plus

[4] Maira Wenzel: .NET-Standard, <https://docs.microsoft.com/de-de/dotnet/standard/net-standard>

Bildquellen:

- Abbildung 1, 2: Eigene Darstellung

Projektvorgehensmodell für Cloud Business Applications im Rahmen des RB IT-PEP Prozesses

Marc Wendel*, Catharina Kriegbaum-Kling, Thomas Rodach

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Wenn man sich als Firma einmal dafür begeistert hat Cloud Applikationen einzuführen, wird einem schnell bewusst, welche Vorteile es gegenüber On-Premise Lösungen mit sich bringt.

Als größter Vorteil ist zu nennen, dass die bezogenen Leistungen nach Nutzung abgerechnet werden. Das bedeutet, dass man bei geringer Nutzung auch weniger bezahlt. Weiter kümmert sich der Anbieter der Cloud Applikation um Updates und Weiterentwicklungen. Außerdem hat man einen Ansprechpartner bei Fragen und Problemen.[1]

Aber wie führt man Cloud Software am Besten in ein Unternehmen ein? Es sind dabei zwei Aspekte zu beachten. Das Projektvorgehensmodell und die Verschiebung der Schwerpunkte bei der Einführung gegenüber einer On-Premise Lösung.

1. Das Projektvorgehensmodell

Zum einen kann man keine klare Aussage treffen, ob ein phasenorientiertes Vorgehensmodell wie das Wasserfallmodell oder ein agiles Vorgehensmodell wie Scrum geeigneter ist, um eine Cloud Applikation einzuführen. Beide haben ihre Vor- und Nachteile. Dennoch muss man eines von beiden für die gesamte Dauer wählen.

Aber warum passt kein phasenorientiertes Modell? Am Anfang werden in der Phase „Spezifikation“ Anforderungen erstellt, die ein System in Zukunft abbilden soll. Schließlich schreibt ein phasenorientiertes Modell vor, dass möglichst alle Anforderungen am Anfang beschrieben werden. Dafür wird Zeit und Geld benötigt. Erst danach wird ein Anbieter einer Cloud Software gesucht, der diese Anforderungen erfüllt.

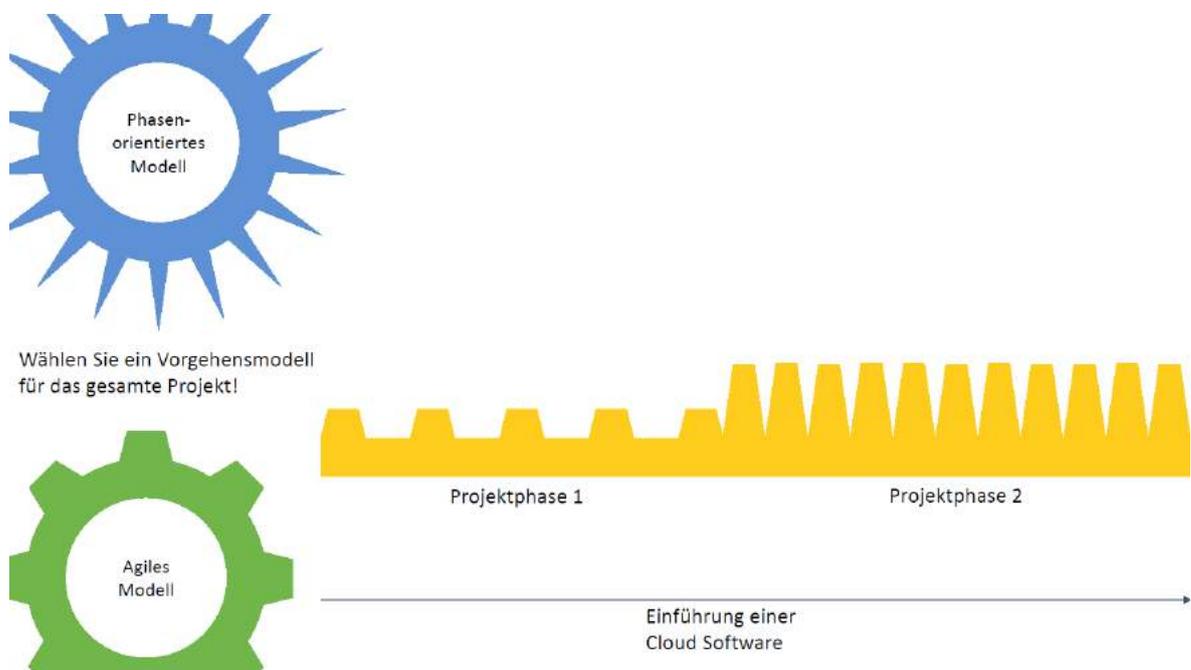


Abbildung 1: Problemstellung bei einer Cloud Einführung

*Diese Arbeit wurde durchgeführt bei der Firma Robert Bosch GmbH, Feuerbach

Möglicherweise kann es aber kein Anbieter geben, der eine passende Cloud Software zu diesen Anforderungen entwickelt hat. Alle erstellten Anforderungen und somit auch die Ressourcen müssen unter Einsatz vieler Ressourcen überarbeitet werden.

Warum passt kein agiles Modell? In einem agilen Modell wie Scrum werden aus einem Sprint Backlog Themen und Arbeitspakete entnommen, die im nächsten Sprint bearbeitet werden sollen. Sobald man sich für einen Anbieter der Cloud Software entschieden hat, sinken die Vorteile bei diesem Modell, da die Cloud Software bereits programmiert ist und läuft.

Die Entscheidung, ob ein agiles oder ein phasenorientiertes Modell benutzt wird, sollte außerdem nicht der Käufer der Cloud Software treffen (müssen), sondern der Anbieter. Er hat bei der Einführung seiner eigenen Software natürlich viel mehr Erfahrung. Dies ist der

Fall, weil er die Software wahrscheinlich schon bei vielen anderen Unternehmen eingeführt hat und er die Software bis ins kleinste Detail kennt.

2. Die Verschiebung der Schwerpunkte bei der Einführung

Zum anderen verschieben sich die Themen der Einführung von der Bereitstellung der IT-Infrastruktur und dem eigentlichen Programmieren in Richtung von Datenschutzthemen und der Auftragsdatenverarbeitung von Partnerunternehmen. Hier müssen Verträge und Service Level Agreements ausgehandelt werden. Diese Schwerpunkte kommen in den bisherigen Vorgehensmodellen nur ungenügend, an der falschen Stelle oder gar nicht zur Geltung.

[1] Janßen, Annika (2013): Wie Big Pharma von der Datenwolke profitiert [online]
<http://www.manager-magazin.de/unternehmen/it/cloud-computing-welche-strategischen-vorteile-die-wolke-bietet-a-933795>. [23.11.2017]

Bildquellen:

- Abbildung 1: Eigene Darstellung

Untersuchung einer Blockchain im Kontext der IT-Sicherheit bei Transaktionen mit IoT-Daten anhand einer Beispielanwendung

Marvin Wiegand*, Dominik Schoop, Andreas Rößler

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Motivation

Blockchain, Distributed Ledger Technologies und Kryptowährungen bekommen immer mehr Aufmerksamkeit von den Unternehmen. Während sich die Bankenbranche hauptsächlich auf Kryptowährungen konzentrieren, interessieren sich auch andere Branchen für Distributed Ledger, einem Überbegriff für Blockchain-Lösungen. Das hängt in der Verbindung mit der Digitalisierung von Firmen und der fortschreitenden Vernetzung im Internet der Dinge (IoT, Internet of Things). Dabei entstehen neue Herausforderungen, insbesondere im Bereich der IT-Sicherheit, die sich mit Distributed Ledger lösen lassen. IoT-Daten sollen in Zukunft nicht mehr zentral gesammelt und analysiert werden, sondern in einem Netzwerk mit Geschäftspartnern ausgetauscht und mit Informationen erweitert werden können.

Das Konzept der Frachtüberwachung mittels Sensoren bei Transport- und Logistikunternehmen verdeutlicht den Ansatz. Sensoren überwachen ob temperaturempfindliche Waren durchgehend mit vorgeschriebener Temperatur transportiert wurden. Bei Auslieferung können anhand der Daten alle Geschäftspartner benachrichtigt werden, dass der Vertrag zwischen Transportunternehmen und Auftraggeber hinsichtlich der vereinbarten Temperaturwerte eingehalten wurde. [1]

Aber durch den Austausch dieser IoT-Informationen zwischen den Geschäftspartnern entstehen wichtige Sicherheitsfragen:

- Wer kann garantieren, dass die Daten von dem angegebenen Geschäftspartner kommen?
- Wie kann abgesichert werden, dass die Informationen nicht verfälscht wurden?
- Wie kann festgelegt werden, Wer, Wann, welche Informationen lesen und schreiben darf?

Je nach Anwendungsfall besitzen die ausgetauschten Informationen einen sehr hohen Wert. Die Informationen könnten sich, wie im

Beispiel, auf eine sehr teure und empfindliche Fracht beziehen und im Schadensfall Rückschlüsse auf die Ursache bieten. Ohne Distributed Ledger muss ein großes Vertrauen zwischen den Geschäftspartnern oder zu einer übergeordneten zentralen Instanz etabliert sein. Die Geschäftspartner müssen darauf vertrauen können, dass die Informationen über die Kühlkette korrekt verteilt werden. Distributed Ledger sollen diesen Austausch der Informationen besser lösen können, ohne eine zentrale Instanz. Durch die Blockchain-Technologie, ein Konsensus-Mechanismus und Verschlüsselungen soll dieser Austausch so sicher sein, dass die Geschäftspartner dieser Lösung ausreichend vertrauen können.

Besonders die Blockchain Hyperledger Fabric der Linux Foundation, bei der IBM stark mitarbeitet, soll unter anderem eine Blockchain für IoT-Netze und Geräte sein. Aus diesem Grund wird in dieser Studienarbeit der Fokus auf die IT-Sicherheit bei Transaktionen von IoT-Daten in Hyperledger Fabric gelegt.

Blockchain Grundlagen

Blockchain ist die Technologie im Zentrum einer Distributed Ledger Technologie (DLT). Distributed Ledger bedeutet wörtlich „verteiltes Kontobuch“. Eine Kryptowährung wie Bitcoin ist eine Spezialanwendung von Distributed Ledger Technologies. Ein Distributed Ledger ist ein Netzwerk in dem eine Art Kontobuch unter den Teilnehmern verteilt und aktualisiert wird. Das Kontobuch einer DLT ist die Blockchain und jeder Eintrag in das Kontobuch ist ein Block der Blockchain. Beim Verteilen jedes neuen Eintrags in das Kontobuch muss darauf geachtet werden, dass alle das gleiche aufschreiben. [2] Jeder Block einer Blockchain speichert den Hash-Wert des vorherigen Blockes, Status-/Geldtransaktionen und den eigenen Hash-Wert. Der Hash-Wert eines Blockes dient als ID und wird der aus den Transaktionen und dem vorherigen Hash-Wert berechnet. [3] (siehe Abbildung 1: Blockchain)

*Diese Arbeit wurde durchgeführt bei der Firma IBM Deutschland GmbH, Ehningen

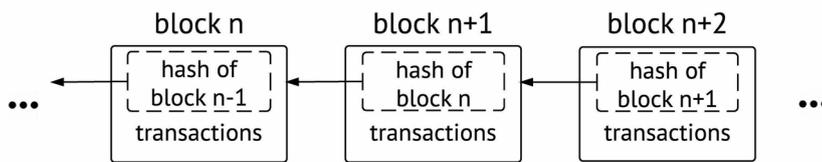


Abbildung 1: Blockchain

Zielsetzung

In der Studienarbeit soll geklärt werden, wie Blockchain als Bestandteil von IoT-Anwendungen zur IT-Sicherheit beiträgt. Zur einfacheren Erklärung werden diese IT-Sicherheitsthemen anhand einer eigens für diese Studienarbeit erstellte Beispielanwendung veranschaulicht.

Vorstellung der Beispielanwendung

Die Beispielanwendung soll es Mitarbeitern/innen vereinfachen ihr privates Fahrzeug für Geschäftsreisen zu nutzen. Dafür muss die Anwendungen garantieren können, dass die Geschäftsreise genehmigt ist, die Fahrt getätigt wurde und melden ob während der Geschäftsreise ein Schaden am Fahrzeug entstand. Wenn Mitarbeiter das private Fahrzeug für eine genehmigte Geschäftsreise verwenden, soll der Mitarbeiter eine Vergütung bekommen können. Falls auf der Geschäftsreise das Fahrzeug beschädigt wurde, soll der Mitarbeiter einen Versicherungsanspruch nutzen können. Dieser wird ebenfalls in der Blockchain gespeichert. Versicherungsansprüche werden dann von der Versicherung der Firma geprüft, statt der privaten Versicherung des Mitarbeiters.

Architektur der Beispielanwendung

Das Fahrzeug eines Mitarbeiters sendet die gefahrenen Kilometer und eventuelle Schadensdaten an eine IoT-Plattform. Die IoT-Plattform sammelt diese Daten und sendet sie an den Server der Firma des Mitarbeiters diese mit der zugehörigen Geschäftsreise. Der Server des Unternehmens ist über einen eigenen Blockchain-Peer mit der Blockchain verbunden und sendet bzw. empfängt Geschäftsreisen und Versicherungsansprüche. Dort angekommen werden die Ansprüche von Prüfern der Versicherung genehmigt oder abgelehnt und an die Blockchain zurück gesendet.

Diese Architektur in Verbindung mit den Anforderungen an den Demonstrator bietet einige interessante Möglichkeiten eine Blockchain auf IT-Sicherheit zu untersuchen. Zunächst haben die Firma und die Versicherung eigene Peers im Blockchain-Netzwerk,

so dass die Verbindung zwischen den Knoten des Netzwerks am besten Verschlüsselt, oder nicht von außen erreichbar sind. Zusätzlich wird eine Schnittstelle zur Übertragung der Daten aus den Fahrzeugen der Mitarbeiter benötigt. Dafür eignet sich eine IoT-Plattform am besten, da sie die IoT-Daten von vielen Fahrzeugen entgegen nehmen muss. Außerdem wechseln die Fahrzeuge ständig ihre Position, haben keine statische IP-Adresse und besitzen unter Umständen keine oder nur eine schlechte Internetverbindung.

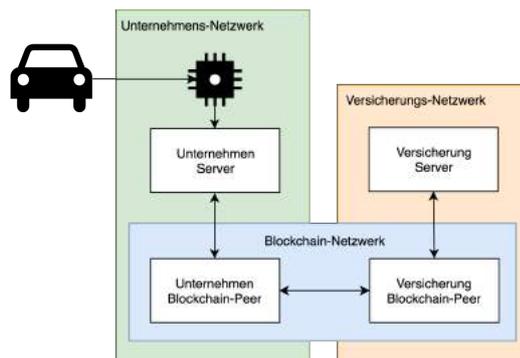


Abbildung 2: Architektur Beispielanwendung

IT-Sicherheits-Aspekte der Beispielanwendung

In der Beispielanwendung werden u.a. personenbezogene Daten verarbeitet die einen hohen Schutz erfordern. Dies sind zum Beispiel: Versicherungsansprüche, Reisekostenabrechnungen und Reisedaten der Mitarbeiter. Zusätzlich sollte das allgemeine Vertrauen zwischen Mitarbeitern, Unternehmen und Versicherung geschützt werden. Für Angreifer sind diese Informationen sehr interessant und könnten die Beispielanwendung auf verschiedene Weisen attackieren. Die verschiedenen Angriffsmöglichkeiten bilden den Attack Tree, die alle zum selben Ziel führen. In der Beispielanwendung wäre es für Angreifer interessant personenbezogene Daten zu stehlen, oder Versicherungsansprüche zu sich umzuleiten. Jeder Angreifer besitzt Eigenschaften, die beschreiben welche Einstiegspunkte und Knoten im Attack Tree verwenden das Ziel zu erreichen. Ein anpassungsfähiger Insider hätte einen anderen Einstiegspunkt, würde an-

dere Sicherheitslücken nutzen, als ein passiver Außenstehender. Da kein System absolut sicher sein kann, muss entschieden werden welche Angreifer und Angriffsmöglichkeiten vom Attack Tree im Fokus stehen. Im Kontext der Beispielanwendung werden die wichtigsten Angriffsszenarien in der Studienarbeit erleutert und beschrieben welche Gegenmaßnahmen die verwendeten Technologien bieten.

Ausblick auf das Ergebnis

Distributed Ledger mit Blockchain und die Integration von IoT-Daten bieten einerseits für IoT wichtige IT-Sicherheitsmöglichkeiten. IoT bringt andererseits interessante Aspekte für die Blockchain.

So entfällt bei der autonomen Generierung von Daten wie bei IoT, ein Teil der möglichen Bedienfehler. Außerdem ist es bei IoT nicht notwendig, dass jedes Gerät zu jeder Zeit online ist, wie es bei Blockchain-Peers der Fall ist, sodass eine gewisse Asynchronität möglich ist, ohne das System einzuschränken.

[1] <https://www.computerwoche.de/a/die-iot-trends-in-deutschland,3330602> (23.11.2017)

[2] http://www.bafin.de/SharedDocs/Veroeffentlichungen/DE/Fachartikel/2016/fa_bj_1602_blockchain (23.11.2017)

[3] <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7467408> (23.11.2017)

Bildquellen:

- Abbildung 1: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7467408> (23.11.2017)
- Abbildung 2: Eigene Darstellung

Kamerabasierte Detektion von Umwelteinflüssen mit Machine Learning Algorithmen

Reinhard Wolfmaier*, Reiner Marchthaler, Werner Zimmermann

Fakultät Informationstechnik der Hochschule Esslingen – University of Applied Sciences

Wintersemester 2017/2018

Einleitung

Über die letzten Jahre hat sich in der Fahrzeugindustrie der Technologiefokus für Sicherheit deutlich verschoben. Wurde früher hauptsächlich im Bereich der passiven Bau-sicherheit eines Fahrzeugs entwickelt, so ist heutzutage das Hauptaugenmerk auf die softwarebasierte Fahrzeugsicherheit gelegt. Mit steigender Rechenperformance von Steuergeräten steigen auch die Möglichkeiten, diese Ressourcen für früher unmöglich umsetzbare Fahrzeugfunktionen zu nutzen. Ein wesentlicher Aspekt ist hierbei die sensorgestützte Erfassung der Umwelt. Das Ziel dieser Bachelorthesis ist es für Sicherheitsfunktionen des Fahrzeugs entsprechende Situationen mit den

vorhandenen Kamerasystemen zu detektieren. Dabei soll ein möglichst robuster Klassifikator mittels Machine Learning entwickelt werden. Im Folgenden wird erläutert welche Schritte vom aufgenommenen Bild zu dessen Klassifizierung unternommen werden müssen.

Bildklassifizierung

Das Klassifizieren von Bildern bedeutet, dass anhand bestimmter, übereinstimmender Merkmale und Muster die jeweiligen Bilder abgegrenzt und eingeordnet werden können. Die Erkennung von Mustern, kann in Teilschritte zerlegt werden, die prinzipiell in Abbildung 1 dargestellt sind.

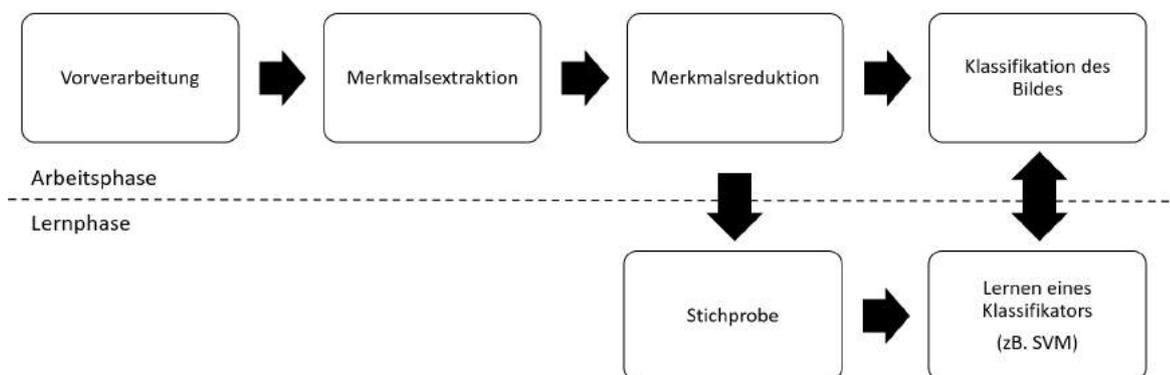


Abbildung 1: Prinzipieller Ablauf einer Bildklassifizierung.

Der wichtige erste Schritt ist stets die Vorverarbeitung der Daten. Der Sinn ist es die Menge der zu verarbeitenden Daten zu reduzieren, indem unerwünschte bzw. irrelevante Bildbestandteile eliminiert werden. Dazu gehört unter anderem die Umrechnung der Bilder in Grauwerte, deren Segmentierung und die Anwendung morphologischer Operatoren [1, Seiten 59–160].

Auf die Vorverarbeitung folgt die Merkmalsextraktion. Hier werden aus dem verarbeiteten Bild Merkmale gewonnen, die später zur Differenzierung verwendet werden können. Die Merkmale können unzählige Gestalten und Formen annehmen, wie beispielswei-

se die Verteilung von Grauwerten, dargestellt durch ein Histogramm, oder durch die Gewinnung von Strukturmerkmalen aus einem anderen Darstellungsraum, beispielsweise durch Fourier-Transformation [1, Seiten 161–302].

Nach der Merkmalsextraktion werden die Merkmale möglichst reduziert, damit für die Klassifikation nur relevante Merkmale miteinbezogen werden. Heutzutage wird versucht für ein gesuchtes Muster eine Wahrscheinlichkeit herauszufinden in welche Klasse es eingeordnet werden kann. Dazu werden unter anderem sogenannte Support Vector Machines eingesetzt [1, Seiten 303–481].

*Diese Arbeit wurde durchgeführt bei der Firma Daimler AG, Sindelfingen

Support Vector Machine

Eine Support Vector Machine (kurz: SVM) dient dazu Eigenschaften, bzw. Merkmale von Daten, bzw. Bildern mit Hilfe eines Lerndatensatzes räumlich voneinander zu trennen und sie somit klassifizieren zu können.

Hat ein Bild die Anzahl N an extrahierten Eigenschaften, so können diese als Punkt in einem N -dimensionalen Koordinatensystem dargestellt werden. In diesem Koordinatensystem repräsentiert jede Achse eine Eigenschaft eines Bildes. Der Lerndatensatz, bestehend aus klassifizierten Bildern, kann somit in solch ein Koordinatensystem eingepflegt werden. Jeder Punkt repräsentiert ein klassifiziertes Bild. Die SVM versucht im folgenden nun eine Hyperebene in den Raum zu legen, welche die Punkte der sich unterscheidenden Klassen räumlich voneinander trennt. Dabei wird darauf geachtet, dass die Ebene den größtmöglichen Abstand zu den unterschiedlichen Klassen einhält. Eine Hyperebene beschreibt eine Ebene im N -dimensionalen Raum mit einem Stützvektor und $N - 1$ Richtungsvektoren. Eine Hyperebene ist somit eine Verallgemeinerung des Prinzips einer klassischen Ebene im dreidimensionalen Raum mit einem Stützvektor und zwei Richtungsvektoren [2, Seiten 75–146][3].

Das Ergebnis einer Klassifizierung mit einer SVM von Daten mit zwei Eigenschaften ist in Abbildung 2 links dargestellt. Die Daten werden dabei in zwei Klassen (hell und dunkel) eingeteilt. Die Klassen lassen sich hier nicht direkt durch eine Hyperebene einteilen, denn eine Hyperebene lässt sich nicht krümmen. Hier-

zu wendet man den sogenannte Kernel-Trick an, welcher die Daten in eine höhere Dimension transformiert. Abbildung 2 zeigt das Prinzip dieses Kernel-Tricks. Es werden die vorhandenen Lerndaten künstlich durch eine Kernel-Funktion in eine höhere Dimension transformiert, in der die Daten durch eine Hyperebene getrennt werden können. Für diese Transformation gibt es unterschiedliche Funktionen und Herangehensweisen. Dazu gehören unter anderem lineare, oder polynomiale Kernel-Funktionen [4].

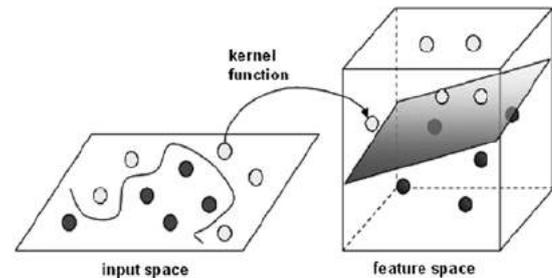


Abbildung 2: Kernel Trick: Linear nicht separierbare Daten können Transformiert in einer höheren Dimension linear separierbar sein.

Die Punkte, die den Verlauf der Hyperebene bestimmen werden Stützvektoren (Support Vectors) genannt. Durch die räumliche Einteilung können weitere Datensätze klassifiziert werden, indem lediglich überprüft wird auf welcher Seite der Hyperebene sie sich mit ihren Eigenschaften einordnen.

[1] Niemann, H., Klassifikation von Mustern. Springer Verlag Berlin, 2003

[2] Burg, K. ; Haf, H. ; Wille, F. ; Meister, A. : Höhere Mathematik für Ingenieure Band II. Springer Verlag Wiesbaden, 2012

[3] Steinwart, I. ; Christmann, A. : Support Vector Machines. Springer Science & Business Media New York, 2008

[4] Zararsiz, G. ; Elmali, F. ; Ozturk, A. : Bagging support vector machines for leukemia classification. In: IJCSI International Journal of Computer Science Issues Bd. 9. IJCSI, 2012, S. 355–358

Bildquellen:

- Abbildung 1: Klassifikation von Mustern, dargestellt nach Niemann [1]
- Abbildung 2: Kerneltrick bei SVM aus Zararsiz et al. [4]

