



Projektgruppe Empower Generations

Abschlussbericht

Themensteller: Prof. Dr.-Ing. Jorge Marx Gómez, Jens Läkamp
Betreuer: Stefan Wunderlich M.Sc., Alexander Sandau M.Sc., Florian Fischer

Vorgelegt von: Andreas Meinke
Arne Christian Loose
Klaus Bruns
Marcel Saager
Maria Taut
Maximilian Meyer
Nikolas Nachtigall
Patrick Baber
Simone Schierloh
Steffen Gerdes

Abgabetermin: 13. Oktober 2017

Danksagung

An dieser Stelle möchten wir uns bei allen Personen bedanken, die uns über die gesamte Projektlaufzeit begleitet und unterstützt haben.

Wir bedanken uns besonders bei unseren Betreuern der Universität Oldenburg: Prof. Dr. Jorge Marx Gómez, Stefan Wunderlich und Alexander Sandau, die uns wöchentlich mit Rat und Tat zur Seite standen und uns auch in schwierigen Zeiten ein offenes Ohr geschenkt haben. Dank euch hatten wir die Möglichkeit, uns im Rahmen unseres Studiums mit innovativen Technologien, spannenden Themen auseinandersetzen und vielen Personen in Kontakt treten zu dürfen, die uns sonst verwehrt geblieben wären.

Außerdem wollen wir uns bei unseren Betreuern der the peak lab. GmbH Co. KG bedanken. Speziell Jens Läkamp und Florian Fischer haben uns stets zur Seite gestanden und uns mit neuen Impulsen vorangebracht. Auch eine ausgiebige Unterstützung aus dem Enera-Projekt konnten wir durch euch erhalten.

Vielen Dank für die Geduld, Mühe und Ressourcen, die ihr in dieses Projekt und in jeden von uns gesteckt habt.

Die Projektgruppe Empower Generations

Inhaltsverzeichnis

Abkürzungsverzeichnis	I
Abbildungsverzeichnis	V
Tabellenverzeichnis	VI
1. Einleitung	1
1.1. Motivation	1
1.2. Aufbau der Dokumentation	2
2. Energiewende	4
2.1. Gesetzliche Rahmenbedingungen	5
2.2. Dezentrale Erzeugung	6
2.3. Digitalisierung/Smart Meter	11
3. Projektmanagement	12
3.1. Projektgruppe	12
3.2. Projektablauf	16
3.3. Rollen innerhalb der Projektgruppe	19
3.3.1. Projektmanagement	19
3.3.2. Teilgruppenverantwortliche	20
3.3.3. Dokumentationsbeauftragte	20
3.3.4. System- und Serveradministrator	23
3.3.5. Websitebeauftragte	23
3.3.6. Event-Team	23
3.4. Meilensteinpläne	24
3.5. Projektphasen	27
3.5.1. Vorbereitung	27
3.5.2. Prototyp Phase 1	28
3.5.3. Prototyp Phase 2	28
3.5.4. Abschlussphase	28
3.6. Projektpartner	29
3.6.1. Rolle der Universität	29
3.6.2. Rolle der peak lab GmbH & Co. KG	29
3.6.3. Rolle des enera-Projektes	29
4. Energy of VALUE	30
4.1. Einleitung	30
4.1.1. Problemstellung	31
4.1.2. Beschreibung	32
4.1.3. Zielsetzung	37
4.2. Vorgehensweise	38
4.2.1. State of the Art	39

4.2.2.	Experteninterviews	41
4.3.	Konzeptionierung	42
4.3.1.	Anforderungserhebung	43
4.3.2.	Akteure	45
4.3.3.	Prognosen	47
4.3.4.	Prognosebasierter Energiehandel	49
4.4.	Implementierung	53
4.4.1.	Architektur	53
4.4.2.	Testumgebung	64
4.4.3.	Smart Contract	67
4.4.4.	Bewertung der Blockchain als Energiehandelstechnologie	69
4.5.	Schlussbetrachtung	71
5.	Energy of WOW	75
5.1.	Einleitung	75
5.1.1.	Motivation	75
5.1.2.	Problemstellung	77
5.1.3.	Aufgabenstellung	80
5.1.4.	Vorgehensweise	80
5.2.	Grundlagen	81
5.2.1.	Begrifflichkeiten	82
5.2.2.	State-of-the-Art / Bestehende Ansätze	85
5.2.3.	Einsatzmöglichkeiten	87
5.3.	Das Konzept	88
5.3.1.	Anwendungsszenario	89
5.3.2.	Business-Model-Canvas	90
5.3.3.	Grobarchitektur und Komponenten	92
5.3.4.	Anforderungen	94
5.3.5.	Zusammenfassung	96
5.4.	Umsetzung / Implementierung	97
5.4.1.	Gesamtarchitektur	97
5.4.2.	Komponente 01: Messstation	100
5.4.3.	Komponente 02: Server	103
5.4.4.	Komponente 03: Preprozessor	103
5.4.5.	Komponente 04: Datenbank	104
5.4.6.	Komponente 05: Converter	105
5.4.7.	Komponente 06: Disaggregation	109
5.4.8.	Komponente 07: Webservice	115
5.4.9.	Komponente 08: Visualisierung	121
5.4.10.	Zusammenfassung	126
5.5.	Empirisch-qualitative Studie	127
5.5.1.	Forschungsdesign und Methodik	127
5.5.2.	Auswertung der Interviews	128

5.5.3. Ergebnis der Studie	131
5.5.4. Zusammenfassung	132
5.6. Fazit und Ausblick	133
6. Energy of TOMORROW	136
6.1. Motivation	136
6.1.1. Problemstellung und Forschungsfrage	137
6.1.2. Zielsetzung	138
6.2. Vorgehensweise	138
6.2.1. Recherche	139
6.2.2. Data Mining	139
6.3. Konzept	142
6.3.1. Anwendungsfälle	142
6.3.2. Grobarchitektur	145
6.3.3. Analyse-/Prognose-Methoden	146
6.4. Technologieauswahl	147
6.5. Entwicklung Verbrauchs- und Produktionsmodell	147
6.5.1. Data Understanding	148
6.5.2. Data Preparation	158
6.5.3. Modelling	173
6.5.4. Evaluation	180
6.6. Technische Umsetzung	186
6.6.1. Problemstellung	186
6.6.2. SAP HANA	187
6.6.3. Microsoft R Server	190
6.6.4. Proof of Concept	193
6.6.5. Predictive Dashboard	195
6.7. Zusammenfassung und Ausblick	198
7. Gemeinsame Applikation	202
7.1. Motivation	202
7.2. Konzept und Design	202
7.2.1. Dashboard	203
7.2.2. Today / Devices	204
7.2.3. Trade / Transactions	206
7.2.4. Prediction	207
7.3. Implementierung	208
8. Fazit	210
Literaturverzeichnis	212
A. Anhang	222
A.1. Paper Energy of VALUE	222

A.2. Paper Energy of WOW	235
A.3. Paper Energy of TOMORROW	248
A.4. Interview-Leitfaden Energy of WOW	256

Abkürzungsverzeichnis

ACF	Autocorrelation Function
API	Application Programming Interface
ARIMA	Autoregressive-Moving Average
CER	Commission for Energy Regulation
CO₂	Kohlendioxid
CRAN	Comprehensive R Archive Network
CRISP-DM	Cross-Industry Standard Process for Data Mining
DBMS	Datenbankmanagementsystem
DWD	Deutschen Wetterdienst
GW	Gigawatt
GWp	Gigawatt Peak
HDF	Hierarchical Data Format
HMM	Hidden Markov Model
HW	Holt-Winters
InK	Informations- und Kommunikationstechnologien
kWh	Kilowattstunde
MA	Moving Average
MAPE	Mean Absolute Percentage Error
MCDA	Multi Criteria Decision Analysis
ML	Machine Learning
MSE	Mean Squared Error
MS SQL	Microsoft Structured Query Language
MWp	Megawatt Peak
NA	Not available
NILM	Nonintrusive load monitoring

NILMTK Nonintrusive load monitoring Toolkit

PA Predictive Analytics

PACF Partial Autocorrelation Function

PAL Predictive AnalyticsLibrary

PG Projektgruppe

PMML Predictive Model Markup Language

PV Photovoltaik

SAX Symbolic Aggregate ApproXimation

TUK Theilsche Ungleichheitskoeffizient

TWh Terawattstunde

UML Unified Modeling Language

PV Photovoltaik

Abbildungsverzeichnis

1.	Vision der Projektgruppe	1
2.	Entwicklung des Photovoltaikmarktes in Deutschland [Kon15]	8
3.	Stromerzeugung in Deutschland 2016 [Str17]	9
4.	Kick-Off-Meeting: Projektmitglieder, Praxispartner von the peak lab. und die universitären Projektbetreuer	13
5.	Statusfolie der Teilgruppen	18
6.	Statusfolie Projektmanagement	18
7.	Confluence Gruppenorganisation [Eigene Quelle]	22
8.	Meilensteinplan für die gesamte Projektlaufzeit	24
9.	Meilensteinplan für die erste Phase Team energy of VALUE (Blockchain)	24
10.	Meilensteinplan für die erste Phase Team energy of WOW (Lastkurvenanalyse)	25
11.	Meilensteinplan für die erste Phase Team energy of TOMORROW (Predictive Analytics)	25
12.	Meilensteinplan für die zweite Phase	26
13.	Meilensteinplan für die Abschlussphase	27
14.	Gartner Hype Cycle [Gar16]	30
15.	Entwicklungsstadien der Blockchain [DS17]	33
16.	Vergleich zentrale und dezentrale Netzwerkinfrastruktur [Eigene Quelle]	34
17.	Veränderung des Transaktionsmodells durch Blockchain [PWC16]	35
18.	Proof of Work vs. Proof of Stake	36
19.	Ablauf einer Transaktion in der Blockchain [PWC16]	36
20.	Akteulle Projekte zum Thema B2C/P2P [DS17]	40
21.	Energiemarkt und Energiehandel [Eigene Quelle]	43
22.	Marktakteure und Vertragsbeziehungen heute	45
23.	Marktakteure und Vertragsbeziehung in Zukunft [Eigene Quelle]	46
24.	Darstellungsweise eines Standardlastprofils [Quelle: Wikipedia]	49
25.	Prognosebasierter Energiehandel [Eigene Quelle]	50
26.	BPMN Energiehandel [Eigene Quelle]	52
27.	Architekturmodell [Eigene Quelle]	55
28.	Screenshot 2. Prototyp [Eigene Quelle]	62
29.	Schematische Darstellung der Testumgebung [Eigene Quelle]	65
30.	RaspberryPi's der Testumgebung [Eigene Quelle]	66
31.	Screenshot der Monitoring Software Ganglia [Eigene Quelle]	67
32.	Smart Meter Daten im Smart Contract [Eigene Quelle]	68
33.	Screenshot Blockchain Funktion in gemeinsamer Applikation [Eigene Quelle]	70
34.	Musterrechnung der EWE AG [EWE12]	78
35.	Überblick Vorgehen bei Disaggregation [Fra11]	83
36.	Verbrauchsdaten mit niedriger Auflösung [Fra11]	84
37.	Verbrauchsdaten mit hoher Auflösung [Fra11]	85
38.	Use Case Diagramm für das entwickelte Szenario [Eigene Quelle]	89
39.	Business Model Canvas [Eigene Quelle]	91

40.	Grobarchitektur Lastkurvenanalyse [Eigene Quelle]	92
41.	Komponentendiagramm Pyranha [Eigene Quelle]	98
42.	Datenstruktur für den Prototyp [Eigene Quelle]	99
43.	Aufbau Messstation [Eigene Quelle]	101
44.	Ablauf NILM-Toolkit [Nip14]	110
45.	Screenshots der Pyranha App Overview [Eigene Quelle]	123
46.	Screenshots der Pyranha App Details [Eigene Quelle]	124
47.	Datenstruktur der Pyranha App [Eigene Quelle]	124
48.	CRISP-DM-Modell	140
49.	Anwendungsfalldiagramm	143
50.	Grobarchitektur	146
51.	Monatskonsumdaten	149
52.	Wochenkonsumdaten	150
53.	Stundenkonsumdaten	151
54.	Marine Boje Stationen [Mar]	153
55.	Verbrauchswerte mit Markierung der Ferientage	153
56.	Verbrauchswerte mit Markierung der Feiertage und Nicht-Feiertage	154
57.	Lufttemperatur Marine Stationen und durchschnittliche Lufttemperatur	154
58.	Lufttemperatur Marine Stationen und durchschnittlicher Verbrauch	155
59.	Wetterstationen Marine Bojen farblich unterschieden	155
60.	Korrelationsmatrix des Produktionsmodells	157
61.	Moving Average von 7 Tagen und tatsächlicher Energieverbrauch	159
62.	Moving Average von 50 Tagen und tatsächlicher Energieverbrauch	159
63.	Moving Average von 100 Tagen und tatsächlicher Energieverbrauch	160
64.	Statistikausgabe in R - Moving Average von 7 Tagen	160
65.	Statistikausgabe in R - Moving Average von 50 Tagen	161
66.	Statistikausgabe in R - Moving Average von 100 Tagen	161
67.	Ergebnisse der Tests	163
68.	Ergebnisse Anomaly Detection	163
69.	Ergebnisse der Anomaly Detection in der Zeitreihe	164
70.	Stationäre Zeitreihe mit vier Segmenten	165
71.	Unterschiedliche Stromverbräuche in unterschiedlichen Haushalten	166
72.	Werte aller Haushalte nach Normalisierung	167
73.	Wertevertellung vor der Normalisierung	167
74.	Wertevertellung nach der Normalisierung	168
75.	Differenzierung des 1. Grades	169
76.	Differenzierung des 2. Grades	169
77.	Differenzierung des 3. Grades	169
78.	ACF und PACF	170
79.	Wertevertellung nach der Normalisierung und der saisonalen Anpassung	171
80.	Neuronales Netz mit den In- und Output Variablen	175
81.	Holt-Winters mit Niveaufaktor von 1	177
82.	Holt-Winters mit Niveaufaktor von 0,5	178

83.	Holt-Winters-Prognose für die nächsten 30 Tage	178
84.	Hidden Markov Modell (eigene Darstellung)	179
85.	Gewichtung der Kriterien	182
86.	Ergebnisse des Verbrauchsmodells	183
87.	Ergebnisse des Produktionsmodells	184
88.	Kommunikationsprozess [SAP17b]	188
89.	Predictive Analytics Library in SAP HANA [Cha13]	189
90.	Komponenten des Microsoft R Servers [Mic17c]	191
91.	Deployment Microsoft R Server [Mic17a]	192
92.	Operationalisierung von R WebServices [Mic17b]	193
93.	Dashboard	198
94.	Dashboard	203
95.	Ansicht: Today	205
96.	Ansicht: Devices	205
97.	Ansicht: Trade	206
98.	Ansicht: Transactions	207
99.	Ansicht: Prediction	208

Tabellenverzeichnis

1.	Übersicht Projektmanagement-Verantwortliche	19
2.	Teilgruppenverantwortlichkeiten	20
3.	Aufbau einer Prognose	48
4.	Datenbankauszug für die Messdaten	104
5.	Services des Converters	106
6.	Aufbau DataFrame konvertierter Daten	107
7.	Services der HTTP-Schnittstelle	116
8.	Anforderungen im Rahmen der Umfrage	132

1. Einleitung

Die Projektgruppe Empower Generations ist eine studentische Forschungsgruppe der Carl von Ossietzky Universität Oldenburg und dem designgetriebenen Softwareunternehmen the peak lab. Im Rahmen des einjährigen Masterprojekts galt es, aktuelle Herausforderungen, getrieben durch die Integration dezentraler Energieressourcen in das Top-Down designte Elektrizitätsnetz durch den Einsatz disruptiver Technologien anzugehen sowie aus Daten und Informationen wertschaffenden Nutzen zu generieren, sodass neue Geschäftsfelder untersucht und erschlossen werden können.

1.1. Motivation

Die Projektgruppe besteht aus elf Studierenden des Masterstudiengangs Wirtschaftsinformatik, welche sich drei innovativen Themen der Energiewende angenommen haben. Zu diesem Zweck wurden die folgenden drei Teilgruppen gebildet:

- energy of WOW (Lastkurvenanalyse)
- energy of TOMORROW (Prognosen)
- energy of VALUE (Blockchain)

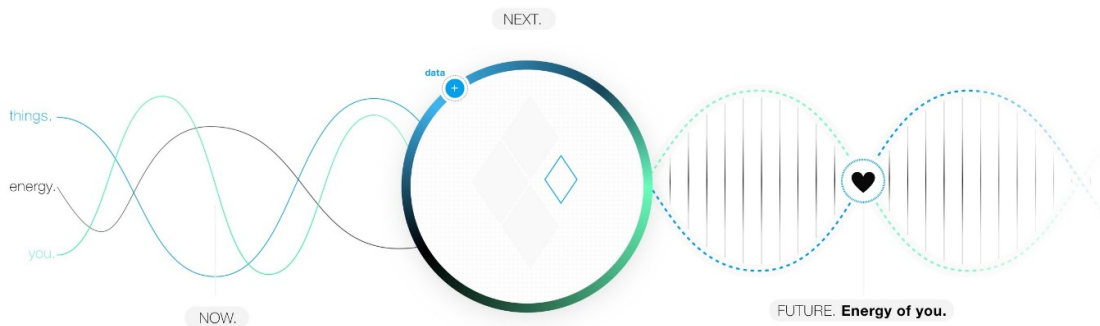


Abbildung 1: Vision der Projektgruppe

Getrieben durch den Leitspruch „Informationsströme, erzeugt von Energie, Menschen und Dingen sind unser Rohstoff“ untersuchen die drei Teilgruppen jeweils energiewirtschaftliche Fragestellungen der Zukunft. Aus Daten werden wertschaffende Informationen, sich ändernde Strukturen und Herausforderungen werden zu Potentialen und es entstehen neue Möglichkeiten durch Digitalisierung und der Energiewende. Die Projektgruppe

Empower Generations hat es sich zur Aufgabe gemacht, einige dieser Möglichkeiten zu untersuchen und mit wichtigen Erkenntnissen einen Beitrag für Forschung und Wirtschaft zu leisten.

Durch fortschreitende Veränderungen aufgrund der Energiewende und der Digitalisierung ergeben sich viele Potentiale, die schließlich auch ausschlaggebend für die Projektgruppe waren. Sowohl internationale als auch nationale Rahmenbedingungen erfordern eine Umstrukturierung alter Muster. Die Einführung intelligenter Zähler ermöglicht es, digitale Messwerte zu erheben und wertschöpfend weiterzuverarbeiten. Im Rahmen der Projektgruppe wurde untersucht, welchen Mehrwert diese neu entstandenen Möglichkeiten für Endkunden haben können und welche Entwicklungen des Energiemarktes ermöglicht werden.

Die Teilgruppe energy of WOW beschäftigte sich primär mit der Analyse von Lastkurven. Die Disaggregation von Energiedaten nach einzelnen Haushaltsgeräten stellte das Team vor eine große Herausforderung. Energy of TOMORROW beschäftigte sich mit Prognosen von Energieverbrauchern und -produzenten. Durch Verwendung spezieller Verfahren und Algorithmen wurde versucht, die Eintrittswahrscheinlichkeit der Prognose zu maximieren. Die Teilgruppe energy of VALUE beschäftigte sich mit der Einbindung der Blockchain-Technologie in den Energiemarkt von morgen. Dabei wurde ein Energiehandelsmodell, basierend auf einer Blockchain, konzipiert und prototypisch umgesetzt.

Im Rahmen des Projekts gelang es, wertvolle Erkenntnisse zu gewinnen und interessante Ergebnisse zu erzielen. Diese wurden durch Vorträge in universitären Veranstaltungen und auf der OLEC Jahrestagung 2017 vorgestellt und präsentiert. Das große Interesse für die Themengebiete sowie der großer Zuspruch des Publikums bestätigte das Projektteam auf dessen Weg. Um die Ergebnisse schriftlich festzuhalten und so auch für weitere Interessenten zugänglich zu machen, wurden drei wissenschaftliche Paper verfasst, die auf der Multikonferenz Wirtschaftsinformatik 2018 eingereicht wurden. Ebenso wurde die Projektarbeit und die wesentlichen Erkenntnisse in der vorliegenden Dokumentation festgehalten.

Wertvolle Unterstützung fand die Projektgruppe durch die Mitarbeiter des Lehrstuhls „Very Large Business Applications“ (VLBA) sowie der Mitarbeiter von the peak lab. Als Ansprechpartner, Berater und Impulsgeber standen diese der Projektgruppe stets zur Seite und leiteten das Projektteam in Richtung Zukunft.

1.2. Aufbau der Dokumentation

Diese Dokumentation dient dazu, den Ablauf der gesamten Projektgruppe Empower Generations besser nachvollziehen zu können. Dabei liegt der Fokus bei den Ergebnissen der

drei Teilgruppen, die im Laufe eines Jahres jeweils einen Prototyp entwickelt haben.

Nach den einleitenden Worten beginnt die Dokumentation inhaltlich mit Grundlagen zur Energiewende. Hierbei soll erläutert werden, wie dieses Projekt im Kontext der Energiewende eingeordnet ist. Außerdem wird ein Fokus auf die technologische Entwicklung des Smart Meters gelegt, die maßgebend für die einzelnen Teilprojekte ist. Das nächste Kapitel befasst sich mit dem Projektmanagement der Gruppe. Die Projektgruppe wird an dieser Stelle vorgestellt und der Projektablauf detailliert aufgezeigt. Am Anfang des Projektes wurden außerdem einige Rollen und Verantwortlichkeiten festgelegt. Ein weiterer wichtiger Punkt im Projektmanagement ist die allgemeine Meilensteinplanung, die an bestimmten Punkten im Projektablauf durchgeführt wurde, hier werden die Meilensteine dokumentiert und erläutert. Daraufhin folgen die sich aus den Meilensteinen ergebenden Projektphasen, die in drei Phasen unterteilt sind. Abgerundet wird dieses Kapitel mit der Vorstellung der Projektpartner und einer kurzen Zusammenfassung.

Das nächste große Kapitel handelt über die Blockchain-Technologie, die von der Teilprojektgruppe energy of VALUE auf innovative Weise in den Energiemarkt eingebracht werden kann. Hier wird die Technologie allgemein vorgestellt und der Prototyp der Projektgruppe von der Konzeptionierung bis zur Implementierung erläutert. Das Kapitel der Teilprojektgruppe energy of WOW widmet sich der Lastkurven-Erkennung von Endgeräten in privaten Haushalten. Hier wurde ebenfalls ein Prototyp konzipiert und implementiert. Darüber hinaus gibt es einen allgemeinen Teil und eine Evaluation. Die dritte Teilprojektgruppe energy of TOMORROW hat ein Modell entwickelt, welches es ermöglicht Vorhersagen über Energieverbräuche zu treffen. Dieser wird ebenfalls in seiner Konzeption und Implementierung erläutert und zum Schluss ausgewertet.

Diese drei Teilprojekte gipfeln in eine mobile Applikation, die in der dritten Projektphase entwickelt wurde. Es wird beschrieben, wie die App entwickelt und umgesetzt wurde. Zum Schluss der Dokumentation wird ein Fazit gezogen, in dem die Ergebnisse reflektiert werden. Im Anhang befinden sich die im Rahmen des Projektes angefertigten Paper.

2. Energiewende

Die Energiewende soll eine sichere, umweltverträgliche und wirtschaftlich erfolgreiche Zukunft erzielen [Bun17b]. Das heutige, auf fossil-nuklearer Erzeugung basierte Energiesystem ist ein Auslaufmodell und die Entwicklung des Energiemarktes im Bereich erneuerbarer Energie schreitet rasant voran [Har17]. Die fortlaufende Verbrennung fossiler Energieträger und nukleare Energiegewinnung führten zu einem drastischen Anstieg negativer externer Kosten, insbesondere der Emissionen des Treibhausgases Kohlendioxid (CO_2) sowie der Entsorgung ausgedienter Brennelemente und radioaktiver Abfälle [Sim08]. Der Abbau bzw. die Förderungen fossiler Energieträger wie Kohle, Erdöl und Erdgas ist durch regionale Verteilung und dem erschwerten Zugang zu den Ressourcen stark eingeschränkt. Laut Schätzungen beläuft sich die Zeitspanne bei derzeitigem Verbrauch von Erdöl auf ca. 44 Jahre bis hin zu Steinkohle mit der längsten Zeitspanne von 150 Jahren [Sim08].

Der ausschlaggebende Punkt ist dabei nicht nur die Begrenzung der Ressourcen, sondern die Verantwortung gegenüber der Nachhaltigkeit, also einer vorausschauenden Planung für die kommenden Generationen. Vor allem die Entstehung von CO_2 -Emissionen durch die Verbrennung von Kohle ist im Gegensatz zu allen anderen Energieträgern deutlich höher. Da bis dato keine Lösung gefunden wurde, Kohlenstoffdioxid in unproblematische Stoffe zu katalysieren, sind andere Maßnahmen gefragt. Zu diesen zählen einerseits die Reduktion fossiler Energieträger und andererseits die Ausweitung regenerativer Energieträger [Sim08].

Zu diesem Zweck hat die Europäische Union und auch die deutsche Bundesregierung Maßnahmen ergriffen, welche das Ziel haben, den Anteil der erneuerbaren Energien über die kommenden Jahre immer weiter zu erhöhen. In den vergangenen 15 Jahren hat sich der Anteil an erneuerbaren Energien bereits verfünffacht und liegt in Deutschland bereits bei über 30 % des gesamten Bruttostromverbrauchs. Darüber hinaus verfolgt die Bundesregierung das Ziel, dass 2050 bereits 80 % des Bruttostromverbrauchs aus erneuerbaren Energien stammen soll.

Neben dem Trend der erneuerbaren Energien wird auch der Einsatz von Informations- und Kommunikationstechnologien (im Folgenden IuK-Technologien) immer weiter verfolgt. Das Beherrschen großer Datenmengen (Big Data) und komplexer Prozesse macht die Unterstützung durch IuK-Technologien unabdingbar. Die bevorstehende Einführung intelligenter Stromzähler (Smart Meter) wird als entscheidendes Kriterium für die Digitalisierung in der Energiewirtschaft angesehen und bietet durch die Lieferung von Echtzeitdaten eine Grundlage für aufbauende Technologien, sodass neue Geschäftsmodelle entstehen können.

Eine weitere wesentliche Herausforderung für den Energiemarkt stellt die Dezentralisierung von wenigen großen Energieerzeugern auf viele kleinere Erzeugungseinheiten dar.

Betrachtet man allein die Solarbranche, so waren in Deutschland im Jahr 2015 PV-Module mit einer Nennleistung von ca. 40 GW installiert und auf ca. 1,5 Millionen Anlagen verteilt [Har17]. Dies zeigt deutlich, dass die Dezentralisierung des Strommarktes bereits fortgeschritten ist und in Zukunft weiter zunehmen wird.

Zuerst werden im Folgenden die gesetzlichen Rahmenbedingungen vorgestellt. Hier wird explizit auf das Erneuerbare-Energie-Gesetz (EEG) eingegangen. Des Weiteren werden die verschiedenen Techniken zur Erzeugung von erneuerbaren Energien vorgestellt. Zu diesen zählen Windkraft, Biomasse, Photovoltaik und die Wasserkraft. Abschließend wird das Thema Digitalisierung behandelt, welches eine zentrale Rolle in der Energiewende bildet, bedingt durch die Installation von intelligenten Stromzählern.

2.1. Gesetzliche Rahmenbedingungen

Das zentrale Gesetz für den Energiesektor ist das Erneuerbare-Energien-Gesetz (EEG), welches 2000 in Kraft getreten ist. Es verfolgt den Zweck, insbesondere im Interesse des Klima- und Umweltschutzes eine nachhaltige Entwicklung der Energieversorgung zu ermöglichen, die volkswirtschaftlichen Kosten der Energieversorgung auch durch die Einbeziehung langfristiger externer Effekte zu verringern, fossile Energieressourcen zu schonen und die Weiterentwicklung von Technologien zur Erzeugung von Strom aus erneuerbaren Energien zu fördern. Zudem soll sichergestellt werden, dass die Netzbetreiber den ökologisch produzierten Strom abnehmen und vorrangig in das Stromnetz einspeisen [Bun17c]. Um die Stromversorgung umweltfreundlicher zu gestalten und damit Deutschland Klimaschutzziele zu erreichen, sind gewisse Ziele definiert, die den Anteil der erneuerbaren Energien in Deutschland ansteigen lassen soll. Laut EEG soll bis spätestens 2050 dieser Anteil an erneuerbaren Energien an der Stromversorgung 80 % betragen. Zentraler Zweck des Gesetzes ist der Schutz von Klima und Umwelt [Pet12].

Ein wichtiges Ziel hat das EEG bereits erreicht, denn seit 2015 macht der Strom aus erneuerbaren Energien den größten Anteil am deutschen Stromverbrauch aus. Dieser Anteil liegt bei rund 30 % und ist somit höher als Strom aus Stein-, Braunkohle, Gas oder Atomkraft. [Bun17c] Durch das EEG wird ein Paradigmenwechsel bezüglich der Förderung vorgenommen. Seit Januar 2017 wird die Höhe der Vergütung für Strom aus nachhaltiger Erzeugung nicht mehr durch den Staat festgelegt, sondern durch eine Ausschreibung ermittelt. Der Grund dafür ist, dass die erneuerbaren Energien als wettbewerbsfähig eingestuft werden. Hierbei gilt, dass jene gefördert werden, die am Wenigsten für den wirtschaftlichen Betrieb einer Anlage fordert. Der Wunsch der Bundesregierung besteht darin, dass eine Vielzahl von verschiedenen Betreibern, von Firmen bis hin zu Bürgerenergiegesellschaften,

am Wettbewerb teilnehmen können [Bun17a]. Ausgenommen von den Ausschreibungen sind kleine Anlagen. Durch die beschriebenen Maßnahmen soll der Anteil der erneuerbaren Energien von rund 30 % bis 2020 auf mindestens 35 Prozent steigern und zeitgleich die Kosteneffizienz erhöhen [Bun17c].

Ein weiterer Punkt des EEG 2017 stellt der Netzausbau dar. Dieser soll künftig mit dem weiteren Ausbau der erneuerbaren Energien voranschreiten. Durch die wachsenden Anteile der erneuerbaren Energien sind neue Herausforderungen für die Netze zu bewältigen. Die Transportwege für den nachhaltig erzeugten Strom sind enorm. Um dieser Situation gerecht zu werden, muss eine leistungsfähige Netzstruktur geschaffen werden und der Ausbau von erneuerbaren Energien vorausschauend koordiniert werden. Im EEG 2017 ist daher geregelt, dass der Ausbau der Onshore-Windkraft in Norddeutschland künftig beschränkt wird, um die Netze zu entlasten [Bun17a].

2.2. Dezentrale Erzeugung

Durch die technologische Entwicklung gibt es derzeit viele verschiedenen Techniken, wie Energie dezentral erzeugt werden kann. In diesem Abschnitt werden vier Möglichkeiten vorgestellt, welche aktuell einen wichtigen Anteil am Strommix haben.

Windkraft Ein wichtiger Baustein im Bereich der erneuerbaren Energien bildet die Windkraft. Der Nord- und Ostseebereich bieten ein enormes Potential für die Möglichkeit der Energieerzeugung, welches mithilfe von Offshore Windkraftenergieanlagen genutzt werden soll. Auf dem Festland sind die Windanlagen keine Neuheit mehr, ob vereinzelt oder in Form eines Parks. Doch das große statische Anlagen im Bereich der Küsten errichtet werden sollen, ist eine völlig neue Situation. Besonders außerhalb des Hoheitsgebiets der Bundesrepublik Deutschland, außerhalb der 12-Seemeilen-Grenze, welche ausschließlich als Wirtschaftszone gilt, liegen keinerlei Erfahrungswerte für Offshore-Windparks vor. Weder auf dem technischen, noch auf dem rechtlichen oder ökologischen Sektor [Sim08].

Die geplanten Parks werden ein Gebiet von jeweils 100 km^2 einnehmen. Dabei ist es von hoher Bedeutung, dass diese sich in das vorhandene System, welches durch die Schifffahrt, Flugverkehr, Vogelzug besteht, einfügen. Bereits vorhandene Nutzungen, wie beispielsweise der Fischerei, Förderplattformen, Kabel- und Rohrleitungen und Ähnliches, müssen dabei berücksichtigt werden. Mit dem Ausbau der Offshore-Standorte müssen demnach wirtschaftliche und technische Herausforderungen gemeistert werden, sondern auch die sozialen und ökologischen Wechselwirkungen sowie die gesetzlichen Rahmenbedingungen beachtet bzw. erst noch geschaffen werden [Sim08].

Biomasse Unter dem „Begriff“Biomasse bzw. „Bioenergie“sind verschiedene Rohstoffe, welche sowohl aus dem regionalen Gebiet als auch über globale Handelsströme stammen können, Techniken und Anwendungsbereiche zu verstehen. Laut dem Umweltbundesamt kann Energie aus den folgenden Rohstoffen gewonnen werden [Umw16]:

- landwirtschaftliche angebaute Pflanzen, dazu zählen beispielsweise Mais, Weizen, Zuckerrübe, Raps, Sonnenblumen und Ölpalmen
- schnell wachsende Hölzer, welche auf landwirtschaftlich genutzten Flächen angebaut werden (Kurzumtriebsplantagen)
- Holz aus der Forstwirtschaft
- biogenen Abfall- und Rohstoffen, welche aus Land- und Forstwirtschaft, Haushalten oder der Industrie gewonnen werden können

Die Bioenergie kann in unterschiedlichen Zuständen zur Verfügung gestellt werden. Gasförmig als Biogas oder Biomethan, flüssig als beispielsweise reines Pflanzenöl für Heizkraftwerke oder als Kraftstoff oder in fester Form als Scheitholz, Holzreste und -pellets [Umw16]. Durch die Vielfalt der Rohstoffe und der Umwandlungstechniken kann die Bioenergie in allen energierelevanten Sektoren eingesetzt werden: als Treibstoff im Bereich der Mobilität, zur Erzeugung von Heizwärme in Haushalten, von Prozesswärme für die Industrie und zur Stromerzeugung [Umw16].

Im Bezug zur Umweltfreundlichkeit der Bioenergie sind geteilte Meinungen vorhanden. Jedoch ist es keinesfalls sinnvoll dieses pauschal zu bewerten, sondern nur unter der Betrachtung von Einzelfällen. Im Falle der Anbaubiomasse steigt die Nachfrage enorm und führt dazu, dass die globale Landnutzung verändert wird. Grund ist der Anbau auf fruchtbaren Ackerflächen, welche nicht mehr primär für die Erzeugung von Nahrungs- oder Futtermittel genutzt werden. Im Hinblick auf die steigende Bevölkerungszahl und der dadurch resultierenden größeren Nachfrage nach Lebensmitteln, ist die Nutzung von wichtigen Ackerflächen suboptimal [Umw16].

Im Jahr 2016 betrug der Anteil, aus Biomasse erzeugter Energie rund 7,9 % und belegt damit Platz zwei hinter der Windkraft [Str17]. Beim direkten Vergleich der verschiedenen Techniken zur Nutzung von erneuerbaren Energien belegt die Biomasse den dritten Platz. Zurückzuführen ist dieses auf die Berücksichtigung des Kriteriums der Flächeninanspruchnahme, denn fruchtbare Flächen sind zunehmend knappe Ressourcen. Diese Tatsachen führen dazu, dass die Biomasse auch in der Zukunft nur einen sehr geringen Teil zur Energieversorgung beitragen wird [Umw16].

Photovoltaik Die Photovoltaik nimmt innerhalb der erneuerbaren Energien eine besondere Rolle ein. Sie ermöglicht die direkte, emissionsfreie Umwandlung von Sonnenlicht in elektrische Energie und wird, aufgrund des großen Potentials, eine tragende Rolle des zukünftigen Energiesystems übernehmen [Kon15]. Das Potential der Technik, die zunächst nur im Bereich der Raumfahrt eingesetzt wurde und lange Zeit als Spielzeug galt, wurde in den 1980er Jahren erkannt. Länder wie die USA, Japan und Deutschland investierten in die Forschungsförderung, um Photovoltaik als Energiequelle zu nutzen. Es entstanden Förderprogramme, welche die Installation von netzgekoppelten Photovoltaikanlagen auf Einfamilienhäuser anregen sollten. Zwischen 1990 und 1995 startete in Deutschland das 1000-Dächer-Programm, das wertvolle Erkenntnisse über die Zuverlässigkeit von Modulen und Fragen zur Netzeinspeisung lieferte [Kon15].

Durch das Erneuerbare-Energien-Gesetz (EEG), welches für erneuerbare Energiequellen kostendeckende Vergütungssätze festlegte, kam es zu einem unerwarteten Boom im Bereich Photovoltaik. Das führte dazu, dass die in Deutschland kumulierte Leistung aller installierten Anlagen von rund 100 MWp im Jahr 2000 auf ca. 38 GWp im Jahr 2014 anstieg. Diese Steigerung entspricht einem jährlichen Wachstum von rund 50 %. In den Spitzenjahren 2010 bis 2012 erreichten die jährlichen Installationen rund 7,5 GWp pro Jahr und sanken, bedingt durch die drastische Verschlechterung der Förderbedingungen, auf unter 2 GWp ab. Die Entwicklung ist in der nachstehenden Abbildung 2 zu entnehmen. In blau dargestellt sind die neu installierten Anlagen in den jeweiligen Jahren und in rot die Leistung aller installierten Anlagen [Kon15].

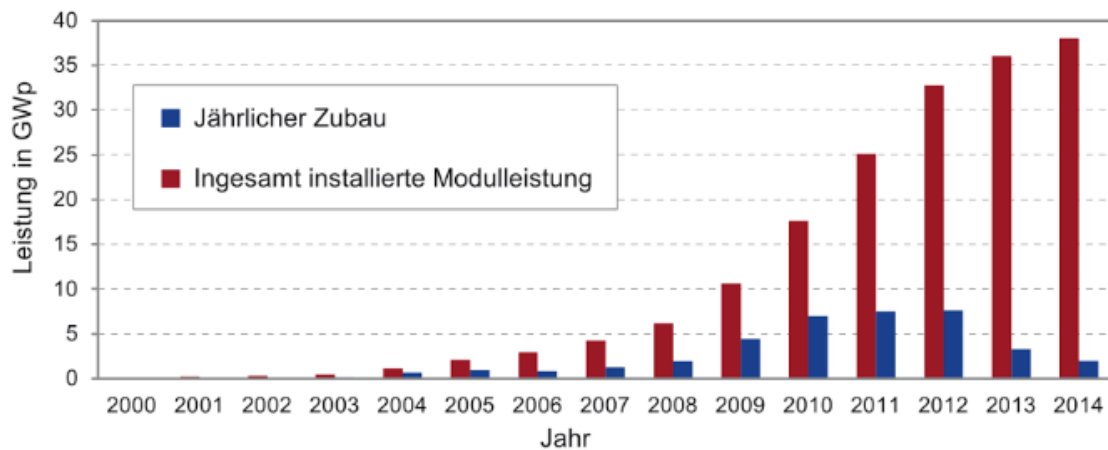


Abbildung 2: Entwicklung des Photovoltaikmarktes in Deutschland [Kon15]

Aktuell beträgt der Anteil des durch Photovoltaikanlagen erzeugten Stroms innerhalb

der erneuerbaren Energien 5,9 % und belegt damit den dritten Platz [Str17]. Im Bezug zur Flächeneffizienz schneiden die Photovoltaikanlagen, zusammen mit der Windkraft, gegenüber anderen Techniken sehr gut ab [Umw16].

Wasserkraft Die vierte und letzte Möglichkeit nachhaltige Energie zu erzeugen bildet die Wasserkraft. Diese Technik ist durchaus keine neue Erfindung, sondern wurde bereits vor der Industrialisierung, als Antrieb von Mühlen, Säge- oder Hammerwerken, genutzt. Die kinetischen und potentiellen Energie der Wasserströmung wird mittels eines Turbinenrads in mechanische Rotationsenergie umgewandelt, welche zum Antrieb von Maschinen oder Generatoren genutzt werden kann [Bun17d].

Für die Energieerzeugung mittels Wasserkraft bieten die südlichen Bundesländer die größten Potentiale, bedingt durch den Voralpenraum, welcher für ein optimales Gefälle sorgt. Auch die Wasserkraft selbst hat einige Potentiale, die vor allem im Bereich der Einsatzmöglichkeiten liegen. Sowohl bei der Modernisierung und Reaktivierung bestehender Anlagen sowie dem Neubau an bestehenden Querbauwerken. Hierbei müssen jedoch sämtliche Umwelanliegen berücksichtigt werden [Bun17d].

Aktueller Stand Im Jahr 2016 wurden in Deutschland rund 648 TWh Strom erzeugt, was eine Steigerung von 0,2 % gegenüber dem Vorjahr ausmacht. 29,5 % stammen aus erneuerbaren Energiequellen, welche damit den ersten Platz im deutschen Strommix einnehmen.[Str17].

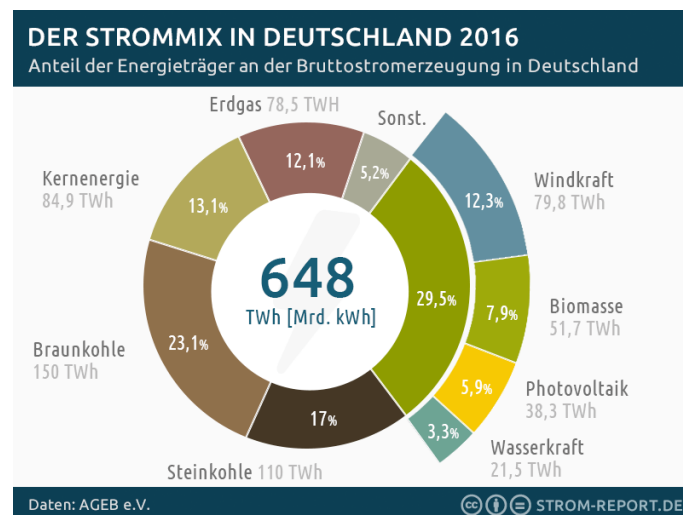


Abbildung 3: Stromerzeugung in Deutschland 2016 [Str17]

Der Begriff Strommix beschreibt die anteilige Zusammensetzung des in Deutschland

erzeugten Stroms nach den jeweiligen Energiequellen. Der Abbildung 3 ist die genaue Zusammensetzung aus dem Jahr 2016 zu entnehmen. In den letzten Jahren ist ein deutlicher Trend zu erkennen, welcher zugunsten der erneuerbaren Energien ausfällt und dazu führt, dass die Energieversorgung umweltfreundlicher wird [Str17].

Herausforderungen Innerhalb der Energiewende gibt es einige Herausforderungen zu bewältigen, welche im folgenden kurz erläutert werden.

- **Energieeffizienz:** Für die erfolgreiche Umsetzung der Energiewende ist es nicht nur wichtig sich auf die Umstellung auf erneuerbaren Energien zu fokussieren, wichtig ist ebenfalls deutlich mehr Energie einzusparen. Besonders bei dem Heizen von Wohnungen oder Häusern und bei der Mobilität sind enorme Einsparpotentiale vorhanden. Durch die Bundesregierung werden Gebäudesanierungen und Elektroautos zugunsten der Energieeinsparung gefördert. Mit der Einführung der Kaufprämie für Elektroautos wurden die Förderprogramme weiterentwickelt, um die Wende voran zu treiben [Bun17b].
- **Versorgungssicherheit:** Der Einsatz von erneuerbaren Energien führt dazu, dass viele Kohle- und Gaswerke weniger rentabel sind als früher, da der erzeugte Strom nicht immer gebraucht und infolgedessen auch nicht abgenommen wird. Dennoch sind sie für die Versorgungssicherheit noch unverzichtbar, beispielsweise, wenn die Wetterlage keine Energieproduktion zulässt. Zu diesen Zeiten sorgen sie permanent dafür, dass die Elektrizitätsversorgung sicher und stabil aufrechterhalten werden kann.
- **Netzausbau:** Zur Zeit sind die deutschen Stromnetze nicht flächendeckend in der Lage erneuerbare Energien zu transportieren. Besonders Problematisch ist der Transport aus dem Norden Deutschlands in den Süden. Der Bau von Offshore-Anlagen lässt sich besser im Norden realisieren, doch wie gelangt der Strom in den Süden? Aus diesem Grund muss der Ausbau der Stromtrassen dringend vorangetrieben werden. In dem Gesetz zum Ausbau von Energieleitungen aus dem Jahre 2009 hat die Bundesregierung bereits 22 Vorhaben als besonders dringend eingestuft. Bis heute wurde rund ein Drittel dieser Vorhaben fertiggestellt. Durch das neue Netzausbaubeschleunigungsgesetz soll die Verfahren für die Planung von Höchstspannungsleitungen deutlich verkürzt werden [Bun17b].
- **Stromspeicher:** Ein wichtiger Punkt bei der dezentralen Energieerzeugung bilden die Speicher. Im Gegensatz zu den fossilen Kraftwerken ist die Energieerzeugung durch Sonnen-, Wind- und Wasserkraft unbeständiger. Je nach Wetterlage stehen

diese oft über Wochen nicht zur Verfügung. Dabei reicht es nicht Lösungen zu entwickeln, welche im Stunden- und Tagesbereich dazu fähig sind überschüssigen dezentralen erzeugten Strom zu speichern. Es werden Speicher benötigt werden, welche ein Vielfaches eines Wochenbedarfs an Energie zur Verfügung stellen können. Da dieses noch ein Zukunftsszenario ist müsste es für den Notfall flexible Kraftwerke geben, welche bei Bedarf schnell hochgefahren werden können [Bun17b], [Die12].

2.3. Digitalisierung/Smart Meter

Aufgrund der Anforderungen im Bereich des Netzausbaus und vor allem der Netzauslastung werden intelligente Stromnetze (Smart Grids) und andere moderne Informationstechnologien notwendig sein. Ein zentrales Ziel der Energiewende, um die Digitalisierung voranzutreiben, ist die flächendeckende Implementierung von Smart Metern [Bun16c].

Als Smart Meter wird ein intelligenter Stromzähler bezeichnet, welcher auf langer Sicht die analogen Ferraris-Zähler ersetzen wird. Die bislang üblichen analogen Stromzähler (Ferraris-Zähler) haben einen großen Nachteil, denn diese sind nur in der Lage den absoluten Energieverbrauch zu messen, ohne die Zuordnung zu diskreten Zeitpunkten. Die absoluten Werte ergeben sich ausschließlich durch die Bilanzierung der Jahreswerte [Dan15]. Intelligente Stromzähler zeichnen Energiedaten in wesentlich kürzeren Intervallen auf und können durch Schnittstellen ausgelesen werden. Durch das Auslesen wird die Datenmenge drastisch ansteigen. Für die Daten muss ein Mehrwert geboten werden, wodurch eine Vielzahl von verschiedenen Anwendungsfällen entstehen [Kat13]. Im weiteren Verlauf der Dokumentation werden drei Geschäftsmodelle in dem Kontext näher vorgestellt.

3. Projektmanagement

In diesem Kapitel Projektmanagement werden alle Aufgaben beschrieben, die sich nicht direkt auf die fachliche Aufgabenstellung beziehen. Dazu werden die verschiedenen Teilgruppen und die einzelnen Mitglieder vorgestellt. Des Weiteren wird das allgemeine Vorgehen erläutert und dabei speziell auf die verschiedenen Verantwortungsbereiche der Teilnehmer eingegangen.

3.1. Projektgruppe

Am einjährigen Masterprojekt *Empower Generations* der Abteilung Very Large Business Applications (VLBA) an der Carl von Ossietzky Universität Oldenburg haben sich elf Masterstudenten der Wirtschaftsinformatik in Kooperation mit dem Praxispartners the peak lab. engagiert. Projektzeitraum lief vom 1. Oktober 2016 bis 30. September 2017. Die Abschlusspräsentation findet am 06. Oktober 2017 statt.

Zu Anfang des Projekts fand ein Kick-Off Meeting am 4. Oktober 2016 statt, an dem sich alle Projektmitglieder und -betreuer beim Praxispartner the peak lab. vor Ort kennengelernt haben. Es folgt das erst entstandene gemeinsame Foto (4).

Wöchentlich fand am Mittwoch ein protokollpflichtiges Projektgruppen-Meeting (variierend in der Universität und beim Praxispartner) statt, in dem alle Ergebnisse und allgemeine Statusinformationen präsentiert wurden. Anschließend wurden bei Bedarf interne Projektgruppen-Meetings abgehalten. Die Kommunikationsplattform Slack und ein E-Mail-Account wurden von the peak lab. zur Verfügung gestellt, über die die Projektkommunikation laufend stattfindet.

Das Projekt teilt sich in drei Teilgruppen auf: Team energy of VALUE (Blockchain), Team energy of WOW (Lastkurvenanalyse) und Team energy of TOMORROW (Predictive Analytics). Alle Teilgruppen haben Schnittmengen zum Thema Energiewende und befassten sich mit aktuellen Themen, Problemstellungen, Herausforderungen und Lösungen in der Umsetzung auf dem jeweiligen Gebiet, wie bereits im Abschnitt Einleitung dargestellt.

Im Folgenden werden die einzelnen Teilgruppen hinsichtlich ihres Themenbereiches genauer erläutert werden. Grundsätzlich hatten die Studierenden einen sehr hohen Freiheitsgrad und hatten keine konkreten Vorgaben seitens Praxispartner bzw. Universität. Lediglich die übergeordneten Themen wurden vorgegeben. Der konkrete Use Case sollte durch die Studierenden identifiziert und umgesetzt werden. Die Teilgruppen haben hierzu jeweils eine grobe Zusammenfassung der jeweiligen Themen formuliert.



Abbildung 4: Kick-Off-Meeting: Projektmitglieder, Praxispartner von the peak lab. und die universitären Projektbetreuer

energy of VALUE

"Wir ermöglichen den direkten Handel im Netz der Zukunft zwischen Konsumenten und Geräten. Hier wird die Grundlage für ein autonomes Energiesystem gelegt. In einer enorm mächtigen, global geteilten Infrastruktur der Blockchain werden Werte bewegt und repräsentieren Eigentum. Die Transaktionen werden in dieser angestoßen, übermittelt und gleichzeitig manipulationssicher dokumentiert."



Arne Christian Loose



Klaus Bruns



Nikolas Nachtigall

energy of WOW

Wir heben den Wert der Smart Meter-Daten für Konsumenten. Jede Energie nutzende Anwendung hat eine eigene Signatur, die wie ein wiederkehrender Herzschlag aus der gesamten Lastkurve extrahiert werden kann. Die Informationen ermöglichen dem Konsumenten den Energieverbrauch besser zu verstehen und neue Potenziale im Netz der Zukunft zu nutzen.



Patrick Baber

Kyra Bohnebeck
(bis März 2017)

Maximilian Meyer



Marcel Saager

energy of TOMORROW

Wir machen aus dem Konsumenten einen Akteur. Um den Herausforderungen zu begegnen, die durch den signifikanten Zuwachs von erneuerbaren Ressourcen gestellt werden, entsteht ein wachsendes Bedürfnis nach Transparenz im Verteilnetz. Lastprognosen privater Konsumenten über den nächsten Tag werden ein wichtiger Baustein im Netz der Zukunft.



Steffen Gerdes



Andreas Meinke



Simone Schierloh



Maria Taut

Diese Beschreibungen wurden als Ausgangspunkte für die folgenden Forschungen in den Teilgruppen genutzt. Der konkrete Anwendungsfall sollte sukzessive im Laufe des Projektes formuliert werden. Es fanden mehrere *Big Picture*-Workshops statt, in denen die Motivation, Ziele und Eigenschaften der abzuliefernden Prototypen je Teilgruppe im Rahmen eines Brainstormings beleuchtet und formuliert wurden. Jede Teilgruppe liefert zum Ende des Projekts jeweils einen Prototypen, eine zugehörige Dokumentation und ein wissenschaftliches Paper ab. Das Gesamtprojekt, wie z. B. Organisatorisches, Projektrollen, etc., ist ebenfalls zu dokumentieren und zu liefern.

Laufend fanden verschiedene Kreativ-Workshops sowohl Projektgruppen-intern als auch extern mit Praxispartnern zu verschiedenen Themen statt (z. B. Webseiten-Gestaltung, gemeinsame Applikation zum Ende des Projekts, usw.). Außerdem wurden verschiedene

projektinterne Bootcamps organisiert, in denen zuvor unbekannte Themen eingeführt wurden, wie z. B. in R, L^AT_EX und git.

3.2. Projektablauf

Im Projektteam hat die Projektmanagement-Methode SCRUM Anwendung gefunden. Aus projektspezifischen Gründen, wie z. B. aufgrund von Projektlaufzeit, Projektgröße und Projektumfang, wurde die Methode den Gegebenheiten etwas angepasst.

SCRUM-Master: Die SCRUM-Master-Rolle wird nicht an ein einzelnes Projektmitglied zugewiesen. Die Master-Rolle als Ganzes umfassen die Teilgruppenverantwortliche und das Projektmanagement-Team. Bei Koordinations-/ Problembehebungsbedarf, etc. wird gemeinsam mit den genannten Verantwortlichen oder bei kleinen Problemen Teilgruppenintern durch den Teilgruppenverantwortlichen eine Lösung dazu erarbeitet. Das Projektmanagement-Team und die Teilgruppenverantwortliche tauschen sich wöchentlich über den aktuellen Stand aus und sprechen ggf. Probleme an um die nächsten Schritte und Maßnahmen einzuleiten. Das Projektmanagement-Team holt sich zentral entsprechende Satis der Teilgruppen ein, um einen Gesamtprojektstatus zu identifizieren und ggf. bei Problemen einzugreifen und Lösungen zu erarbeiten. Das erste Treffen fand am 23. November 2016 um 13:30 Uhr im Besprechungsraum der Abteilung VLBA statt. Dieser Termin findet ab Juli 2017 alle zwei Wochen, in der sich keine Demo ereignet, statt.

Tool: Für die Sprint- und Ticketerstellung wird JIRA genutzt, welches von the peak lab. bereitgestellt wurde. Diese Anwendung stammt aus dem Hause Atlassian und richtet sich speziell an Entwickler-Teams, die eine genaue Planung und Koordination von Aufgaben verschiedener Teams benötigen. Die Funktionalität von JIRA reicht von Bug- und Issue-Tracking über die Verwaltung von Aufgaben, Code und zentrale Projekte bis hin zu detaillierten Berichten und Analytics-Werkzeugen. Diese bereitgestellten Möglichkeiten lassen sich mit Hilfe von zahlreichen Plugins und Integrationen erweitern. In der agilen Softwareentwicklung wird SCRUM häufig als bewährte Management-Methode eingesetzt. Daher gehend hat sich JIRA als eines der effizienten und anwenderfreundlichen Projektmanagement-Tools etabliert. [Die14]

Ticket-Status-Möglichkeiten (wie Backlog, Ready for Testing, Bug, etc.) und allgemein die Relevanz des JIRA-Tools sind Teilgruppenintern auszuwählen.

Sprint-Planung: Das Sprint-Planung soll innerhalb der einzelnen Teilgruppen stattfinden und wird von den Teilgruppenverantwortlichen umgesetzt. Der Sprint-Start bzw. die

erste Sprint-Planung findet am 17. November 2016 statt. Die Sprints sollen zweiwöchentlich andauern und beginnen immer am Donnerstag und enden am Mittwoch. Die genauen Termine für die Teilgruppen-Termine für Sprint-Planung, etc. sind jeweils Gruppen-intern festzulegen. Estimation-Meetings (z. B. Zeitschätzung, Komplexitätsschätzung und/oder allgemeine Aufwandsschätzung nach Fibonacci-Zahlen) sind nicht zwingend durchzuführen. Jede Gruppe kann selbst festlegen, ob ein Estimation-Meeting für deren Entwicklungsarbeit sinnvoll ist. Für die Sprint-Bezeichnung wurde folgende Namenskonvention in JIRA eingehalten:

S(Nr.)_(Gruppenkürzel); z. B.: S01_WOW (Sprint-Nummer 1, Team WOW)

Demo, Sprint-Review und -Retrospektive: Alle zwei Wochen findet eine Demo, ein Sprint-Review und eine Sprint-Retrospektive statt. User Stories, die erledigt sind, sollen anhand einer Demo im gemeinsamen PG-Treffen im Rahmen des Reviews vorgestellt werden. Eine detaillierte Sprint-Retrospektive findet nur im internen Teilgruppentreffen statt. Ein Review dient der konkreten Ergebnis-Vorstellung und eine Retrospektive erfüllt eine Reflektion der Sprintarbeit, in der ggf. Verbesserungsmöglichkeiten oder Probleme identifiziert werden können. Daraufhin können entsprechende Maßnahmen eingeleitet werden.

Weeklys: Jede Woche, in der keine Demo bzw. kein Review stattfindet, wird seitens aller Teilprojekte im PG-Treffen ein Zwischenstand der aktuellen Projektarbeiten und des allgemeinen Fortschritts kombiniert mit der Statusinformationsfolie (siehe Abbildung 5) an alle Projektmitglieder und die Betreuer kommuniziert (pro Gruppe ca. 5-10 min). Gleichzeitig werden ebenfalls identifizierte Risiken und die zugehörigen präventive und reaktive Maßnahmen vorgestellt.

Status – Teilgruppe				
Projektfortschritt			Status	
			Ist	Prognose
Nächste Schritte				
Was?	Bis wann?	Wer?	Prognose	
Risiken				
Was?	Maßnahmen			

Abbildung 5: Statusfolie der Teilgruppen

Ebenfalls werden dazu Informationen zum Gesamtprojekt und ein allgemeiner Projektstatus seitens Projektmanagement-Team mitgeteilt (siehe Abbildung 6).

PM-Statusinfo						
Termine				Gesamtstatus		
Was?	Wann?	Wo?			Ist	Prognose
Meilensteine				Offene Punkte, sonstiges		
	Meilenstein	Plan	Ist	Status	Text einfügen...	
<i>vorheriger</i>						
<i>aktuell</i>						
<i>nächster</i>						
Nächste Schritte					Stimmung	
Was?	Bis wann?	Wer?	Wird erreicht?			

Abbildung 6: Statusfolie Projektmanagement

User Stories & Tickets: Jede Teilgruppe erstellt in JIRA soweit sinnvoll die entsprechenden User Stories und die dazugehörigen Tickets (z. B. in JIRA).

Zu Anfang des Projekt wurde je Teilgruppe ein Grobkonzept erstellt, das die Ziele der Projektarbeit definiert hat. Im Laufe der Projektzeit haben sich weitere Ziele ergeben oder Teilziele wurden im Laufe des Projekts verändert.

3.3. Rollen innerhalb der Projektgruppe

Jeder Teilnehmer der Projektgruppe muss über das Jahr verteilt eine Verantwortung für einen Aufgabenbereich übernehmen. Diese Rollen gehen teilweise über das ganze Jahr oder wechseln in bestimmten Zyklen. Im folgenden Abschnitt werden die Rollen beschrieben und die entsprechenden Teilnehmer benannt

3.3.1. Projektmanagement

Das Projektmanagement-Team übernimmt Organisations-, Koordinations- und Kontrollaufgaben laufend im gesamten Projekt. Zusätzlich ergibt sich die Aufgabe einen wöchentlichen Projektstand an alle Projektmitglieder, Projektbetreuer und Praxispartner kontinuierlich zu kommunizieren. Außerdem ergaben sich Aufgaben wie das Vorbereiten und Koordinieren interner und zum Teil externer Termine. Konflikte, Probleme und interne Schwierigkeiten wurden dem Projektmanagement-Team übermittelt und Lösungsvorschläge gemeinsam mit entsprechenden Teilgruppen oder Projektmitgliedern erarbeitet. Da das Projekt in drei Phasen unterteilt war, ergibt sich für jede Projektphase ein anderes Projektmanagement-Team, das aus jeweils zwei Projektmitgliedern bestand.

Das Projektmanagement-Team, welches auch als Scrum-Master agierte, bestand zur Projektzeit laufend aus zwei Projektmitgliedern. Es folgt die Übersicht der Verantwortlichen und die Phasen, indem das Amt besetzt wurde (Tabelle 1).

Tabelle 1: Übersicht Projektmanagement-Verantwortliche

Phase	Verantwortliche/r
Erste Phase	Kyra Bohnebeck, Maria Taut
Zweite Phase	Patrick Baber, Nikolas Nachtigall
Dritte Phase	Simone Schierloh, Maria Taut

Fand ein Wechsel des Projektmanagements-Teams statt, dann erfolgte ein Abschluss-/Austauschtreffen mit dem alten und neuen Projektmanagement-Team und auch zum Teil mit den Praxisbetreuern statt. Im Rahmen der Projektmanagement-Aufgaben wurde ein Strafenkatalog eingeführt.

3.3.2. Teilgruppenverantwortliche

Jede der drei Teilgruppe verfügt in jeder Phase über einen Teilgruppenverantwortlichen. Jedes Teilgruppenmitglied übernimmt für die Dauer einer Phase die Verantwortung der Teilgruppe und koordiniert die Aufgaben im Rahmen der Sprints (inkl. Erstellung von Tickets, z. B. in JIRA) und Repräsentationsaufgaben für das Teil-Team. Zu den Aufgaben zählen auch die Erstellung von Demos und Statusfolien. Des Weiteren sind sie Ansprechpartner für die Betreuer bei organisatorischen und inhaltlichen Punkten. In dem Zuge findet regelmäßig ein Abstimmungstreffen mit den Betreuern und Praxispartnern statt, um Feedback einzuholen und Abstimmungen zu verschiedenen Themen vorzunehmen. In der folgenden Tabelle 2 sind die Namen und Phasen der Teilgruppenverantwortlichkeiten aufgeführt.

Tabelle 2: Teilgruppenverantwortlichkeiten

Teilgruppe	Verantwortliche/r
Erste Phase	
Energy of VALUE	Klaus Bruns
Energy of WOW	Patrick Baber
Energy of TOMORROW	Andreas Meinke
Zweite Phase	
Energy of VALUE	Klaus Bruns/Arne Christian Loose
Energy of WOW	Kyra Bohnebeck
Energy of TOMORROW	Steffen Gerdes
Dritte Phase	
Energy of VALUE	Arne Christian Loose/Nikolas Nachtigall
Energy of WOW	Maximilian Meyer
Energy of TOMORROW	Maria Taut
Vierte Phase	
Energy of VALUE	Nikolas Nachtigall
Energy of WOW	Marcel Saager
Energy of TOMORROW	Simone Schierloh

3.3.3. Dokumentationsbeauftragte

Zu den wesentlichen Aufgaben der Dokumentationsbeauftragten zählten folgende Punkte:

- Einrichtung der Kollaborationssoftware Confluence
- Erstellung von Vorlagen

- Koordination und Finalisierung der Dokumentation

Da ein gemeinsames Wiki als zentrale Informationsquelle im Rahmen eines Projekts sehr hilfreich ist, wurde die Kollaborationssoftware Confluence verwendet. Die Dokumentationsbeauftragten haben direkt zu Beginn des Projekts die entsprechende Seitenstruktur aufgebaut, sodass das Wiki über den gesamten Projektzeitraum die zentrale Informationsplattform darstellte und häufig verwendet wurde. Der Aufbau gliederte sich grundsätzlich in einen organisatorischen und einen Arbeitsbereich. Im Bereich „Gruppenorganisation“ teilte sich wiederum in Sitzungsangelegenheiten, Planungsangelegenheiten und Sonstiges, die im Folgenden aufgelistet sind. Zu den Sitzungsangelegenheiten gehören Moderationspläne, Sitzungsprotokolle und Teilgruppen-Protokolle. Die Planungsangelegenheiten umfassen die Projektplanung, Urlaubsplanung, PG-Raumplaner, Zeitplanung für Feedback-Gespräche und Projektabschlussfeier, die jeweils mit Hilfe des Confluence durchgeführt wurden. Weitere Punkte sind Strafenkatalog, Kontaktdaten, Anleitungen und Logo oder Grafiken.

Da die einzelnen Punkte selbsterklärend sind, wird nicht näher darauf eingegangen. Weiterhin war in diesem Bereich eine Reihenfolge der Protokollführung und der Sitzungsleitung für die wöchentlichen Meetings zu finden. Ebenso war eine Übersicht der aktuell besetzten Positionen vorhanden. Da die Positionen teilweise gewechselt wurden (Projektmanagement und Teilgruppenverantwortliche) bzw. aufwändige Positionen durch weitere Personen verstärkt wurden (Event-Team mit Maria Taut, Webseitenpflege mit Marcel Saager) waren somit die Zuständigkeiten und Verantwortlichkeiten stets transparent und zentral aufzufinden. In Abbildung 7 ist der soeben beschriebene Aufbau des Bereichs *Gruppenorganisation* dargestellt.

Weiterhin wurde neben dem organisatorischen Bereich ein *Gruppen-Arbeitsbereich* eingerichtet. In diesem war in der Übersichtsseite der aktuelle Projektplan bzw. Meilensteinplan zu finden sowie die teilgruppeninternen Seiten mit speziellen Themen der einzelnen Teilgruppen. So konnten bspw. Literatur, Erklärungen von Fachbegriffen oder teilgruppenspezifische Anleitungsartikel zentral abgelegt und von allen eingesehen werden.

Neben dem Gruppen-Arbeitsbereich im Confluence wurde zusätzlich Google-Drive zur gemeinsamen Erstellung von Dokumenten und das Versionsverwaltungsprogramm git zur gemeinsamen Anwendungsentwicklung verwendet. Nähere Informationen zum Git werden im folgenden Unterkapitel näher erläutert. Die Verwendung der Kollaborationssoftware Confluence war demnach eine äußerst hilfreiche und häufig verwendete Wiki-Plattform. Durch die zügige und übersichtliche Einrichtung der Plattform stellte diese vom Beginn bis zum Ende der Projektgruppe die zentrale Informationsplattform dar. Neben der Einrich-

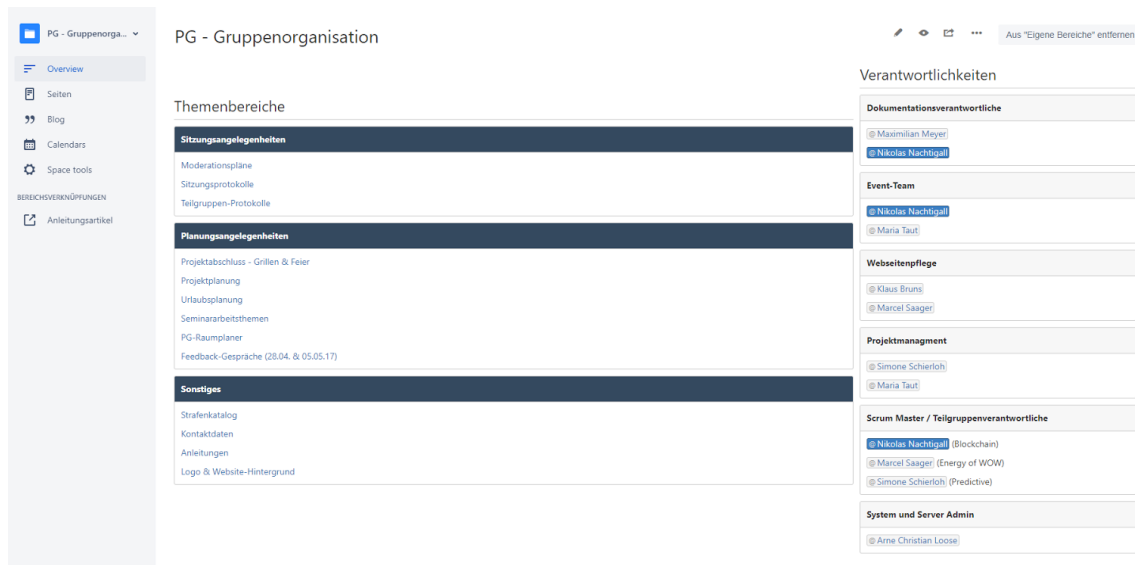


Abbildung 7: Confluence Gruppenorganisation [Eigene Quelle]

tung und Pflege der Wiki-Plattform Confluence zählte auch die Erstellung von notwendigen Vorlagen zu den Aufgaben der Dokumentationsbeauftragten. Ebenfalls zu Beginn des Projekts wurden Vorlagen für Word-Dokumente, Powerpoint-Dokumente, Confluence-Seiten (z.B. Sitzungsprotokolle) und \LaTeX -Dokumente erstellt. Da nicht jedes Teammitglied mit der Erstellung von \LaTeX -Dokumenten vertraut war, wurde zudem im Rahmen der Vorlagenerstellung auch eine Anleitungsseite im Confluence erstellt, sodass der Umgang mit dem Programm erleichtert wurde. Dadurch, dass es diese Vorlagen gab, wurden späterer Abstimmungsbedarf innerhalb der Teilgruppen überflüssig, redundante Arbeiten vermieden und ein gleichbleibendes Äußeres der Dokumente gewährleistet. Besonders in Vorträgen musste sichergestellt werden, dass im Rahmen der Projektgruppe die einzelnen Teilgruppen stets dieselben Folienvorlagen verwendet. So wurde sowohl intern als auch extern ein einheitliches Erscheinungsbild gewahrt. Um die Ergebnisse der Projektgruppe festzuhalten und für potentielle Interessenten transparent zu machen, wurde die vorliegende Dokumentation erstellt. Die einzelnen Kapitel wurden innerhalb der Projektgruppe verteilt, sodass jedes Teammitglied einen Teil zur Dokumentation beitragen konnte. Die Dokumentationsverantwortlichen koordinierten die Erstellung der Dokumentation, setzten Deadlines fest und gaben eine weitere Anleitung zur Verwendung von \LaTeX aus, sodass schließlich ein reibungsloser Ablauf bei der Erstellung der Dokumentation ermöglicht wurde. Das Zusammentragen der einzelnen Kapitel sowie die inhaltliche und formelle Endabnahme der Dokumentation und auch das Drucken zählten ebenfalls zu den Verantwortungsbereichen

der Dokumentationsbeauftragten. Die Dokumentationsbeauftragung lag über die gesamte Projektlaufzeit bei Maximilian Meyer und Nikolas Nachtigall.

3.3.4. System- und Serveradministrator

Ein wichtiger Posten in der Projektgruppe war die des Server- bzw. Systemadministrators. Tools wie Git mussten Server-seitig aufgesetzt und zur Verfügung gestellt werden. Ebenso nutzte die Teilgruppe energy of TOMORROW eine Microsoft R-Server Infrastruktur. Diese musste ebenfalls administrativ betreut werden. Die Aufgabe des Systemadministrators war daher die jeweilige Software aufzusetzen, instand zu halten und in Kontakt mit Verantwortlichen seitens der Universität und the peak lab. zu stehen. Arne Christian Loose hat die System- und Serveradministration in diesem Projekt verantwortet.

3.3.5. Websitebeauftragte

Es gab in der Projektgruppe zwei Websitebeauftragte, die sich um die Inhalte der Website gekümmert haben. Die Website (www.empowergenerations.de) wurde in enger Abstimmung mit dem Betreuer seitens the peak lab., mithilfe von wordpress entwickelt. Hier wurden alle allgemeinen Informationen über die Ziele des Gesamtprojektes und der Teilprojekte vorgestellt und Bilder der Projektteilnehmer und Betreuer veröffentlicht. Darüber hinaus wurde ein Blog gepflegt, in dem regelmäßig Informationen über den Fortschritt des Projektes bekannt gegeben wurden. Außerdem wurde über Social-Events und Besuche auf Messen informiert. Zudem ist eine Kontaktmöglichkeit zur Projektgruppe auf der Website bekanntgegeben. In diesem Projekt haben Marcel Saager und Klaus Bruns die Website-Arbeiten vorgenommen.

3.3.6. Event-Team

Das Event-Team besteht aus zwei Projektmitgliedern. Jeden Monat wurde ungefähr eine teambildende Veranstaltung oder Aktivität organisiert. Dazu zählen z. B. Fußballspiele im In- und Outdoor-Bereich, sowie Bowling-Spiele, Grill- und Spieleabende in Oldenburg. Die Aktivitäten und Termine für die teambildende Maßnahmen wurden über Doodle-Umfragen abgestimmt. Sowohl die Betreuer seitens der Universität und des Praxisunternehmens als auch Projektmitglieder anderer Projektgruppen haben sich an den Social-Events beteiligt. Falls sich finanzielle Angelegenheiten ergeben haben, wurde dies über das Social-Event-Team koordiniert. Auch Einkäufe, z. B. für Grillabende, wurden über das Event-Team getätigt. Das Social-Event-Team bildeten in diesem Projekt Nikolas Nachtigall und Maria Taut.

3.4. Meilensteinpläne

Jede Teilgruppe erstellte einen Meilensteinplan, in dem die abgestimmten Punkte zeitlich eingeplant und dargestellt wurden. Im Folgenden ist ein Meilensteinplan 8 dargestellt, der zu Anfang für das Gesamtprojekt zur Orientierung gedient hat.



Abbildung 8: Meilensteinplan für die gesamte Projektlaufzeit

Für jede Projektphase wurde folgend ein Meilensteinplan je Teilgruppe und schließlich ein Gesamtmeilensteinplan ab der zweiten Phase überblicksartig für alle Teilgruppen erstellt. Es fand also laufend eine Aktualisierung der Meilensteinpläne statt. Anhand der Deadlines im Meilensteinplan ließ sich der Fortschritt des Gesamtprojekts messen.

Es folgt der Meilensteinplan von Team energy of VALUE vom Oktober 2016 bis Januar 2017 (Darstellung 9).

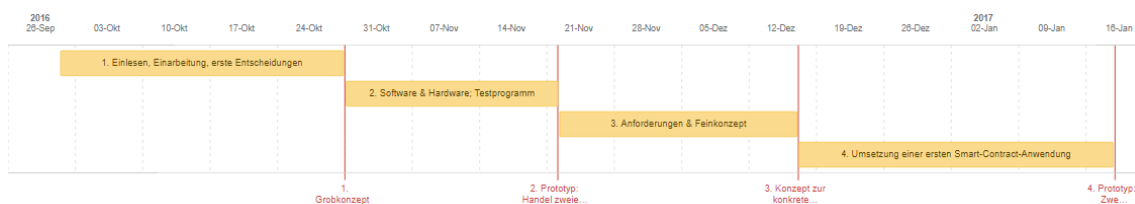


Abbildung 9: Meilensteinplan für die erste Phase Team energy of VALUE (Blockchain)

Anschließend ist der Meilensteinplan von Team energy of WOW vom Oktober 2016 bis Januar 2017 dargestellt (Darstellung 10).

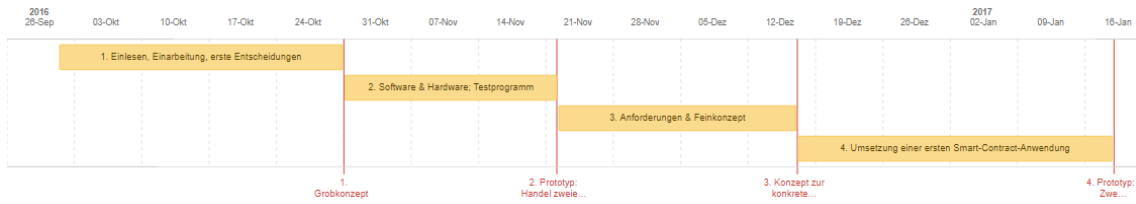


Abbildung 10: Meilensteinplan für die erste Phase Team energy of WOW (Lastkurvenanalyse)

Schließlich folgt der Meilensteinplan von Team energy of TOMORROW ebenfalls vom Zeitraum Oktober 2016 bis Januar 2017 (Darstellung 11).

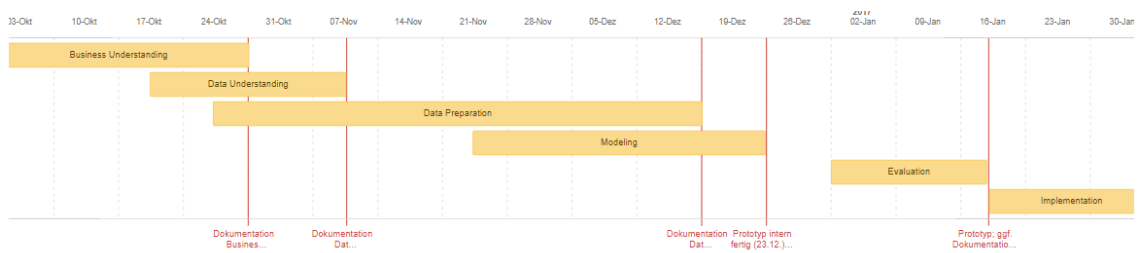


Abbildung 11: Meilensteinplan für die erste Phase Team energy of TOMORROW (Predictive Analytics)

In der zweiten Projektphase vom März bis Juni 2017 wurde ein gemeinsamer Meilensteinplan erstellt, der alle Teilgruppen-Meilensteine beinhaltet. In diesem Gesamtmeilensteinplan ist ebenfalls eingetragen, wann und wo welche Präsentationen und wo die wöchentlichen Projektgruppen-Meetings (the peak lab. oder in der Universität) stattfanden. Dieser ist folgend in Darstellung 12 abgebildet.

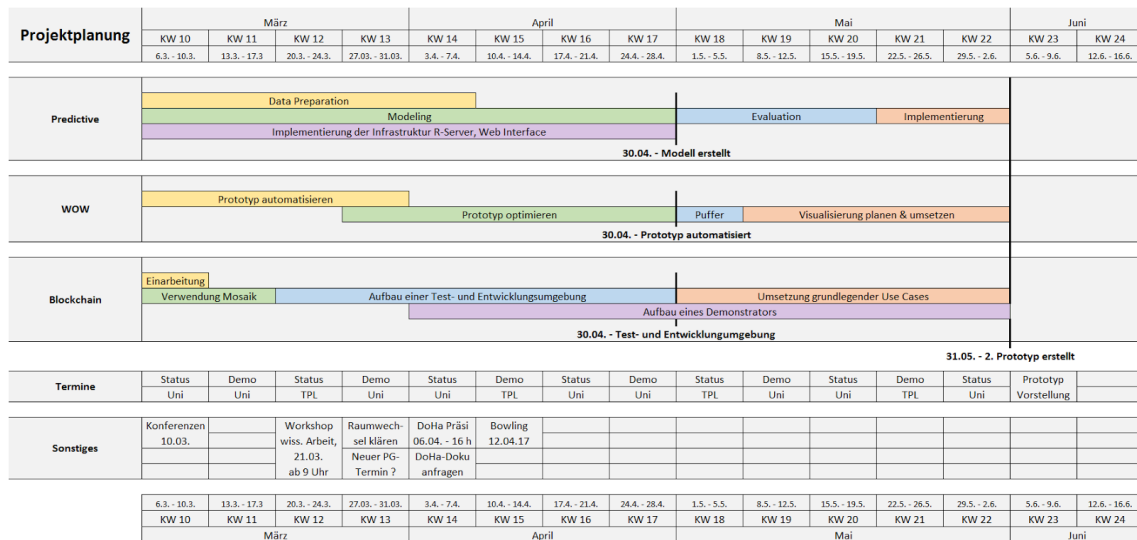


Abbildung 12: Meilensteinplan für die zweite Phase

Für die Abschlussphase vom Juni bis September 2017 wurde ebenfalls ein Meilensteinplan aufgestellt, der genau wie in der zweiten Phase alle Teilgruppen-Meilensteine vereint. In Darstellung 13 ist dieser zu entnehmen.

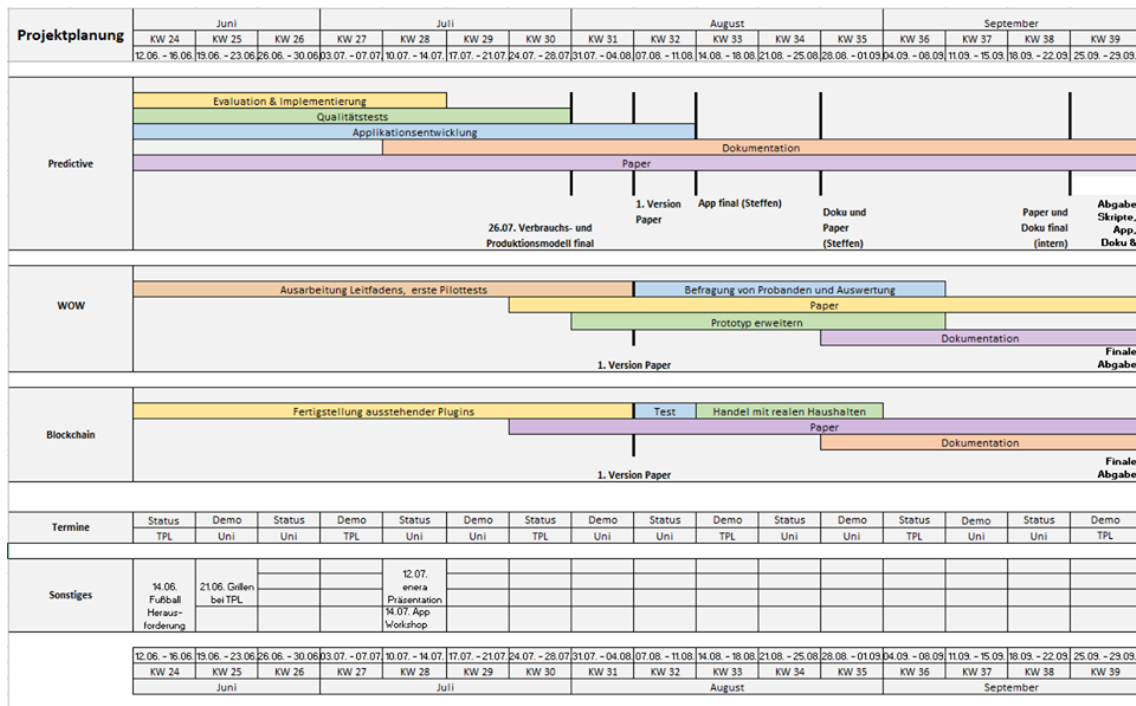


Abbildung 13: Meilensteinplan für die Abschlussphase

3.5. Projektphasen

Die folgenden Projektphasen wurden im Laufe der Projektgruppe von allen Teilbereichen durchlaufen.

3.5.1. Vorbereitung

Wie in Kapitel 3.1 bereits erläutert, wurde die Projektgruppe in Teilgruppen unterteilt. Die Themen der Teilgruppen wurden durch den Praxispartner the peak lab. definiert und beim Kick-Off Meeting präsentiert. Beginnend wurde das Know-How der Projektteilnehmer analysiert und notwendige Workshops geplant. Die Workshops wurden unter anderem zum Thema Git und L^AT_EX gehalten, damit alle Projektteilnehmer auf einem Level sind. Innerhalb der Teilgruppen wurde das Wissen themenbezogen analysiert. Da die Themengebiete in der Regel für die Studierenden unbekannt waren, war es notwendig sich vorerst in diese einzuarbeiten. Die Betreuer setzten insgesamt drei Termine für die Präsentation von Prototypen fest. Der erste Prototyp spiegelte die grundlegende Beschäftigung mit der zu untersuchenden Technik wieder. Der zweite Prototyp beinhaltet einen konkret ausgearbeiteten Use Case. Letztere Präsentation beinhaltet die Präsentation einer gemeinsamen

Applikation, in welcher die zuvor in den Teilgruppen erstellten Lösungen eingebunden wurden.

3.5.2. Prototyp Phase 1

Die erste Prototyp-Phase des Projekts dauerte insgesamt vier Monate. Im Vordergrund stand dabei zuerst die Grundlagen-Recherche. Dafür hat sich jede Teilgruppe ca. einen Monat eingeplant. Auf Grundlage dieser Recherchen, begann jede Gruppe damit Frameworks, Methoden und Programmiersprachen festzulegen, und gegebenenfalls aufbauend auf anderen verwandten Arbeiten zu arbeiten. Demzufolge konnte jede Teilgruppe somit auch Projekt-Meilensteine für sich erarbeiten. Die Abschlusspräsentation des ersten Prototypen fand am 2. Februar 2017 statt.

3.5.3. Prototyp Phase 2

Die Arbeit in der zweiten Prototyp-Phase baute auf den ersten Prototypen auf und dauerte insgesamt vier Monate. Weitere verschiedene Techniken wurden zusätzlich evaluiert und die Projekt-Meilensteine wurden zeitlich angepasst. Zum Ende der zweiten Prototyp-Phase wurde festgelegt, dass jede Teilgruppe ein Paper verfasst, welches dann bei einer passenden Konferenz eingereicht wird. Dazu hat sich jede Teilgruppe mit verschiedenen Konferenzen befasst und eine eigene Auswahl getroffen. Die Präsentation der zweiten Prototyp-Phase wurde am 7. Juni 2017 gehalten.

3.5.4. Abschlussphase

Die dritte Prototyp-Phase des Projekts, ist die Abschlussphase. In dieser Phase wurden primär die letzten Anpassungen an den Prototypen vorgenommen, die Paper bearbeitet und die Projektdokumentation geschrieben. Gleichzeitig wurde die MKWI (Multikonferenz Wirtschaftsinformatik) als Konferenz ausgewählt, bei der alle Teilgruppen ihre Paper einreichen. Eine fundamentale Entscheidung, die in dieser Phase gefallen ist, dass die Ergebnisse aller Teilgruppen in einer mobilen Applikation zusammengefasst werden sollen. Demzufolge hat sich jeweils ein Mitglied jeder Teilgruppe damit befasst, diese Applikation zu verwirklichen. Zusätzlich zu den Demonstrationen in dieser Phase, gab es die Möglichkeit, die bisherigen Ergebnisse beim HEdIS-Projekt (Hub for Education on ICT for Sustainability), im Rahmen der OLEC-Tagung und beim enera-Projekt vorzustellen und die Meinung von weiteren Leuten einzuholen. Die Abschlusspräsentation fand am 06.10.2017 statt.

3.6. Projektpartner

Im Laufe des Projektes haben verschiedene Projektpartner Einfluss auf die Projektgruppe genommen. Diese Rolle der verschiedenen Partner wird im folgenden Abschnitt genauer erläutert.

3.6.1. Rolle der Universität

Die Abteilung Very large Business Applications (VLBA) der Carl von Ossietzky Universität Oldenburg betreute die Projektgruppe während der zwölf Monate lang im Rahmen des Masterstudiums. Die zwei Betreuer standen mit Rat und Tat den Projektteilnehmern bei organisatorischen und auch inhaltlichen Fragestellungen zur Seite. Es fand ein von der Uni initiiertes wöchentliches Treffen am Mittwoch statt, in dem die einzelnen Teilgruppen ihre Fortschritte in der Abteilung dem Plenum vorstellten. Außerdem in regelmäßigen Abständen mit den einzelnen Projektteilnehmern Feedback geführt, in denen diskutiert und ein derzeitiger Leistungsstand dem Studenten offenbart wurde. Aus dieser Einzelleistung und der Gesamtleistung der Projektgruppe setzt sich die Endnote zusammen.

3.6.2. Rolle der peak lab GmbH & Co. KG

Die peak lab GmbH & Co. KG ist der Praxispartner und Aufgabensteller der Projektgruppe. Sie ist ein innovatives, designorientiertes Softwareunternehmen mit Sitz am Hafen von Oldenburg. Hierbei wurde der Projektgruppe der dritte Betreuer zur Verfügung gestellt, welcher sich wöchentlich Zeit für jede Teilgruppe des Projektes nahm. Dabei konnten die Wünsche und Anregungen beider Seiten unverbindlich ausgetauscht werden. Das Unternehmen hat den Studenten vielerlei Praxiskontakte im Bereich der Softwareentwicklung und Energiewirtschaft ermöglicht. Darüber hinaus stellte the peak lab. den Studierenden einige technische Arbeitsmittel zur Verfügung, um die Themen des Projektes bestmöglich bearbeiten zu können.

3.6.3. Rolle des enera-Projektes

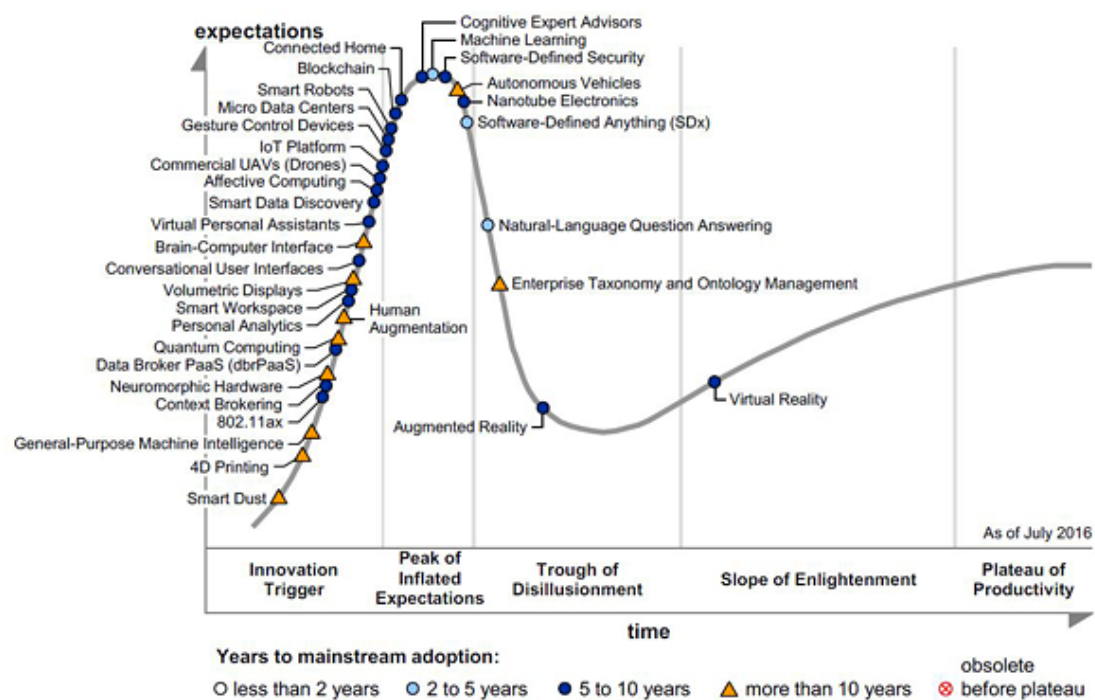
Das ENERA Projekt hat den Sinn, die Energiewende in Deutschland zu bewältigen. Ziel ist es durch dezentrale Energieerzeugung und disruptiven Technologien diese Vorhaben voranzutreiben. ENERA stellte den Studenten ebenfalls vielerlei Technik zur Verfügung und ermöglichte einen Austausch auf fachlicher Ebene.

4. Energy of VALUE

Diese Teilgruppen-Dokumentation beinhaltet die Ergebnisse der Teilgruppe energy of Value. Einleitend wird die Ausgangslage und die damit verbundene Problemstellung erläutert. Die Vorgehensweise wird in Kapitel 4.2 beschrieben und beinhaltet eine Analyse der aktuellen Entwicklung in der Forschung und Wirtschaft sowie die Beschreibung der Arbeitsweise im Verlauf dieses Projekts. Der Kern der Projektarbeit stellt die Konzeptionierung mit folgender Implementierung dar. Der damit verbundene Proof of Concept mündet in einem Fazit und wird abschließend ausblickend betrachtet.

4.1. Einleitung

In Kapitel 2 wurde die aktuelle Entwicklung des Energiemarktes erläutert. Die Transformation zu einem dezentralen Erzeugersystem und die Möglichkeit, Energieinformationen digital über Smart Meter weiterverarbeiten zu können, bietet neue Lösungsmöglichkeiten für Informationssysteme.



Source: Gartner (July 2016)

Abbildung 14: Gartner Hype Cycle [Gar16]

Die Blockchain wurde für das Projekt aus zweierlei Gründen als technologische Basis gewählt. Erstens lässt die dezentrale Eigenschaft der Technologie eine passende Implementierung eines dezentralen Energiemarktes vermuten und zweitens wird die Blockchain sowohl in der Forschung als auch in der Wirtschaft hinsichtlich ihres disruptiven Potentials untersucht.

Abbildung 14 zeigt, dass das Buzzword Blockchain im Juli 2016 am Peak des Gartner Hype Cycles gewesen ist. Das Plateau der Produktivität ist also zu Beginn der Projektphase noch weit entfernt gewesen. Daraus folgt eine hohe Relevanz von Proof of Concepts, um die Technologie voranzutreiben.

4.1.1. Problemstellung

Die in Kapitel 2.2 dargestellte Ausgangslage wirft relevante Fragestellungen auf und bietet einen interessanten Forschungsrahmen zur Umsetzung verschiedener Szenarien auf Grundlage einer Blockchain-Anwendung. Die Verwendung von Blockchain-Technologie stößt im Finanzsektor bereits auf großes Interesse und auch im Energiesektor ergeben sich Potentiale und Einsatzmöglichkeiten [PWC16].

Durch die fortschreitende Entwicklung dezentraler Erzeugerstrukturen und den Einsatz digitaler Messtechnik, ergeben sich für den Energiemarkt Möglichkeiten hinsichtlich der Entwicklung neuer Geschäftsmodelle. Eine Anpassung an sich ändernde Umweltbedingungen ist netz- und marktseitig zwingend erforderlich, da elektrische Energie aus regenerativen Energiequellen nicht nach Bedarf gesteuert wird. Die zuvor nach Bedarf regelbaren konventionellen Kohlekraftwerke bieten die Möglichkeit, planbar Energie zu erzeugen und je nach Netzlast steuernd eingreifen zu können.

Erneuerbare Energien werden in Deutschland durch die EEG-Umlage weiter gefördert, sodass erneuerbare Energieanlagen weiter ausgebaut werden und die Probleme der Planungsunsicherheit sowie die Einhaltung der Netzstabilität in Zukunft weiter zunehmen werden. Die Planungsunsicherheit ist auf die erhöhte Anzahl dezentraler Erzeuger zurückzuführen. Die unbekannt Menge an Strom, die in das Netz eingespeist wird, erschwert die Erhaltung der Netzstabilität. Dem Endverbraucher wird kaum die Möglichkeit gegeben, auf die Volatilität Erneuerbarer Energien zu reagieren. Durch über Monate und Jahre statisch festgesetzte Preise ist es nicht möglich, die Netzlast an dynamisch schwankende Energieerzeugung anzupassen. Die sich ändernden Angebots- und Nachfragemengen an elektrischer Energie werden im Energiemarkt demnach unzureichend an die Endverbraucher herangetragen. Dieses Informationsdefizit gilt es zu beseitigen.

Der heutige Energiemarkt ist auf zentrale Erzeugungsanlagen und Verbrauchsprognosen

nach Standardlastprofilen ausgerichtet. Standardlastprofile sind standardisierte Prognosen über den zukünftigen Lastgang eines Energieverbrauchers. Die prognostizierte Energie wird am Terminmarkt bereits Wochen oder Jahre im Voraus gehandelt. Dazu werden feste Verträge mit Energielieferanten bzw. Aggregatoren vereinbart, die den Energiehandel übernehmen. An Energiebörsen kann weiterhin anhand von Verbrauchsprognosen auf einem Spotmarkt ein Tag sowie auf einem Intradaymarkt bis zu 45 Minuten vor der Energiebereitstellung gehandelt werden. Erneuerbare Energien können auf dem Spotmarkt am vorherigen Tag prognostiziert und entsprechend gehandelt werden.

Der Wandel zu einem dezentralen Energiemarkt bietet Potential für eine Umsetzung durch Blockchain Technologie, da sich diese Technologie durch ihre dezentrale Struktur auszeichnet. Zahlreiche Start-Ups befassen sich daher mit der Entwicklung von Geschäftsmodellen im Energiesektor in Verbindung mit Blockchain-Technologien [PWC16]. Das Brooklyn Microgrid ist ein Beispiel für eine isolierte Umsetzung einer Peer2Peer Handelsplattform mit dezentral erzeugtem Strom. Das PowerLedger Projekt verfolgt den Vertrieb und die Dokumentation von erneuerbaren Energien in einem offenen Stromnetz.¹ Die Analyse der bestehenden Projekte in Forschung und Wirtschaft ergab jedoch die Notwendigkeit der Berücksichtigung der Netzkomponente [HD17]. Dezentrale Erzeugerstrukturen führen im Rahmen der Energiewende zu neuen Anforderungen und erfordern Anpassungen des markt- und netzseitigen Energiesystems. Fehlende Informationen über prognostizierte Lasten und Energieerzeugung erschweren die Einhaltung der Netzstabilität. Der Blockchain wird Lösungspotential hinsichtlich der neuen Anforderungen im Energiemarkt prophezeit [PWC16]. Die vorliegende Arbeit untersucht daher, ob ein dezentraler Energiehandel mithilfe der Blockchain realisiert werden kann. Dabei stellt sich auch die Frage, ob die markt- und netzseitigen Komponenten des Energiemarktes auf einer Blockchain berücksichtigt werden können.

4.1.2. Beschreibung

Der Hype um die Blockchain-Technologie begann im Jahr 2008 durch ein White Paper, welches unter dem Pseudonym Satoshi Nakamoto veröffentlicht wurde [Nak08]. Auf Basis dieses Papers entstand unter anderem die Kryptowährung Bitcoin. Da die algorithmische Architektur der Blockchain durch das oben genannte White Paper frei zugänglich gemacht wurde, ist diese Idee von der Community weiterentwickelt worden. So entstanden mittlerweile zahlreiche Kryptowährungen und alternative Applikationen auf Basis der Blockchain. Ethereum zählt neben Bitcoin zu den bekanntesten Beispielen. Die Blockchain-Technologie

¹ Für weitere Informationen zu PowerLedger siehe: <https://powerledger.io/>

hat mittlerweile mehrere Entwicklungsstadien durchlebt und befindet sich aktuell in Phase 2.0. Abbildung 15 zeigt, dass eine Ausprägung der Blockchain 2.0 die Nutzung von Smart Contracts ermöglicht. Smart Contracts sind reaktive Programme, die auf der Blockchain ausgeführt werden können. Die Idee dahinter ist die Reduktion von Bürokratie durch Automatisierung und Dezentralisierung. Beispielsweise repräsentieren Smart Contracts einen Vertrag, der automatisch nach zuvor definierten Bedingungen abgewickelt wird.

Der Wandel zu einem dezentralen Energiemarkt bietet Potential für eine Umsetzung auf Grundlage der Blockchain-Technologie, da dezentrale Strukturen und transaktionsbasierte Erlösmodelle abgebildet werden können. Um das Konzept dieser Arbeit zu verstehen, ist es notwendig, die wesentliche Funktionsweise der Blockchain-Technologie zu begreifen. Die Blockchain stellt ein dezentrales, verteiltes Datenbanksystem dar.

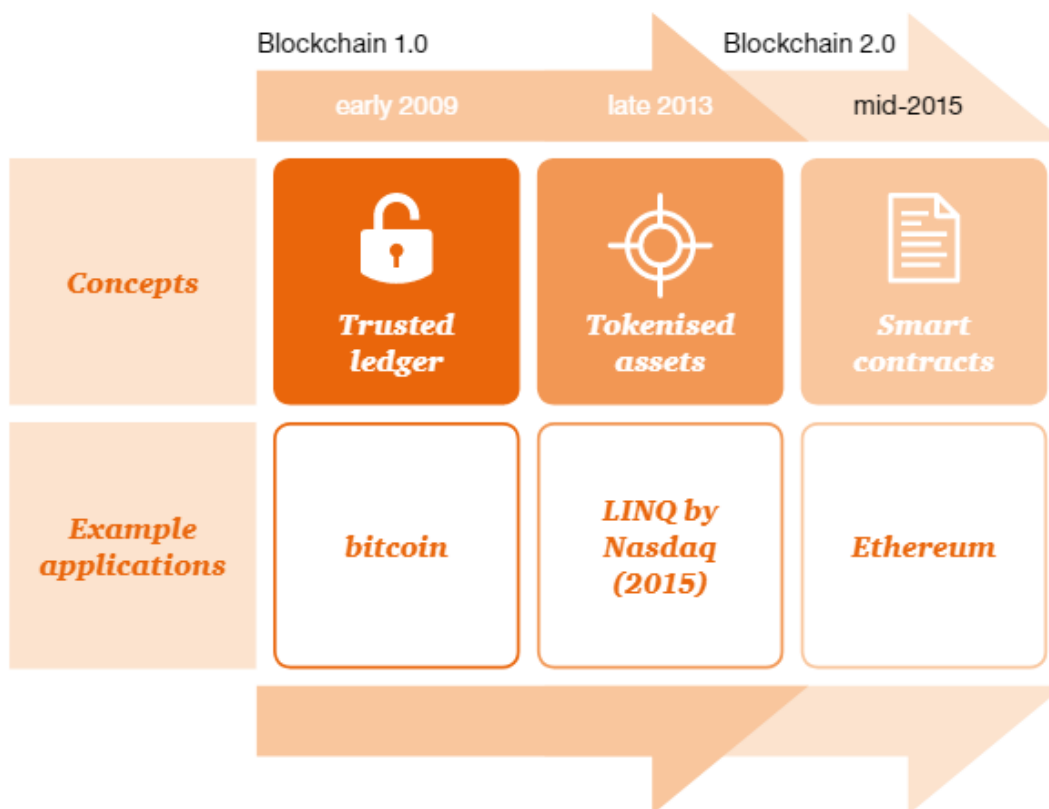


Abbildung 15: Entwicklungsstadien der Blockchain [DS17]

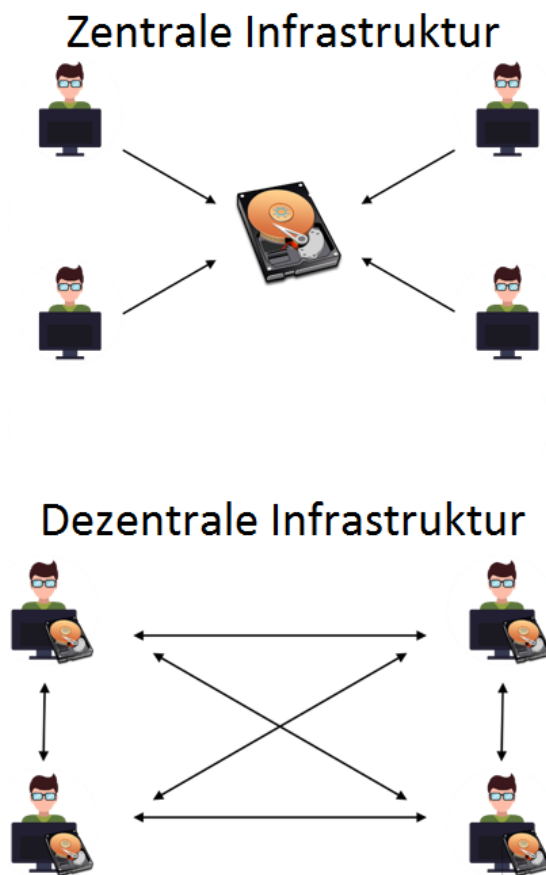


Abbildung 16: Vergleich zentrale und dezentrale Netzwerkinfrastruktur [Eigene Quelle]

Abbildung 16 illustriert den Unterschied zwischen einer zentralen und dezentralen Datenbankinfrastruktur. Bei einer zentralen Infrastruktur greifen viele Clients auf einen Datenbestand zu. Eine dezentrale Infrastruktur hat hingegen die Eigenschaft, dass jeder Client eine identische Datenkopie der Daten besitzt. Daraus folgt eine erhöhte Erreichbarkeit und Integrität. Die Rechenleistung kann im Netzwerk aufgeteilt werden. Diese Dezentralität lässt die Vermutung zu, dass der zukünftigen dezentrale Energiemarkt sich unter Verwendung einer Blockchain als technologische Basis realisieren lässt. Aus diesem dezentralen Ansatz folgt eine Veränderung im aktuellen Transaktionsmodell.

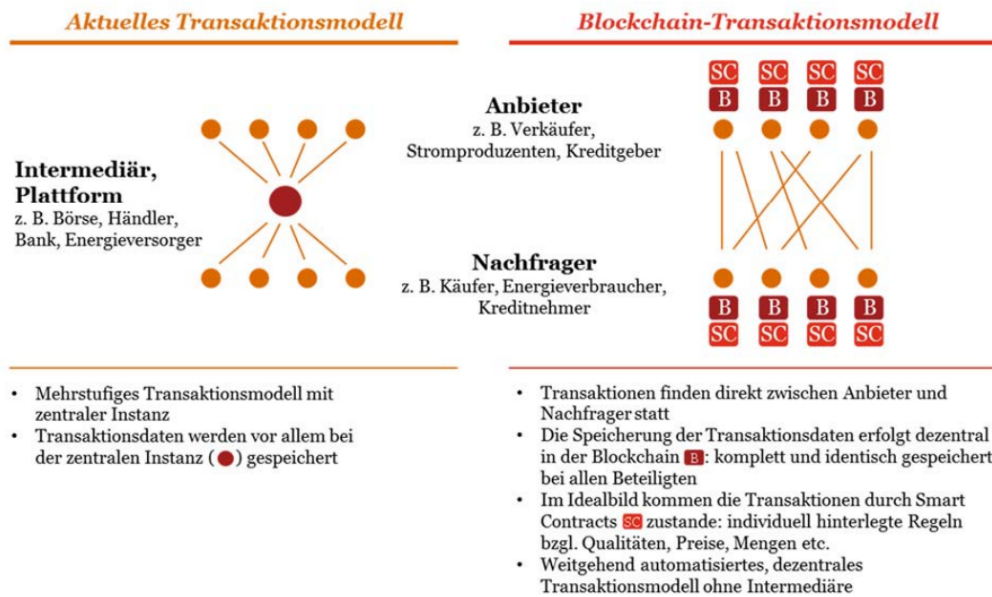


Abbildung 17: Veränderung des Transaktionsmodells durch Blockchain [PWC16]

Abbildung 17 stellt dar, wie ein Transaktionsmodell in einer Blockchain aussehen kann. Die aus einer PWC Studie stammende Abbildung bildet ein Transaktionsmodell ohne Intermediär ab. Wie in der Problemstellung bereits erläutert, gilt es ein Konzept zu entwickeln, in dem Marktakteure wie der Netzbetreiber und der Energieversorger im Transaktionsmodell berücksichtigt werden.

Neben der Infrastruktur stellt der Konsensmechanismus einen weiteren wichtigen Faktor dar. Dieser sorgt dafür, dass die Daten validiert und von allen Teilnehmern des Systems akzeptiert werden. In diesem Zusammenhang gibt es aktuell zwei wesentliche Ansätze. Der am häufigsten genutzte Validierungsmechanismus ist der Proof of Work. Neben diesem gibt es den Ansatz des Proof of Stake.

An dieser Stelle sei lediglich gesagt, dass der Proof of Work auf zu erbringende Rechenleistung basiert und sich die Transaktionszeit mit der Zeit verlängert. Dieser Aspekt wird in der Bewertung der Technologie aufgegriffen, da die Performance einen wichtigen Aspekt von Blockchain-Applikationen darstellt. Der Validierungsmechanismus auf Grundlage des Proof of Stake ist bspw. bei Ethereum aktuell noch in Entwicklung, verspricht aber deutlich bessere Performance als der Proof of Work.

Die Blockchain basiert algorithmisch auf einer erweiterbaren Liste (im Folgenden als Chain bezeichnet). In der Chain werden Datenrecords gespeichert, die Blocks genannt werden. Dadurch ist der Inhalt des jeweiligen Blocks eindeutig verschlüsselt. Ebenso ist

jeder Block mit einer eindeutigen Adresse versehen. Somit wird der Aspekt der Datenintegrität umgesetzt. Die Hauptoperationen in einer Blockchain stellen die Transaktionen dar. Diese dienen dazu, dass Daten von einem Account zu einem anderen übertragen werden können.

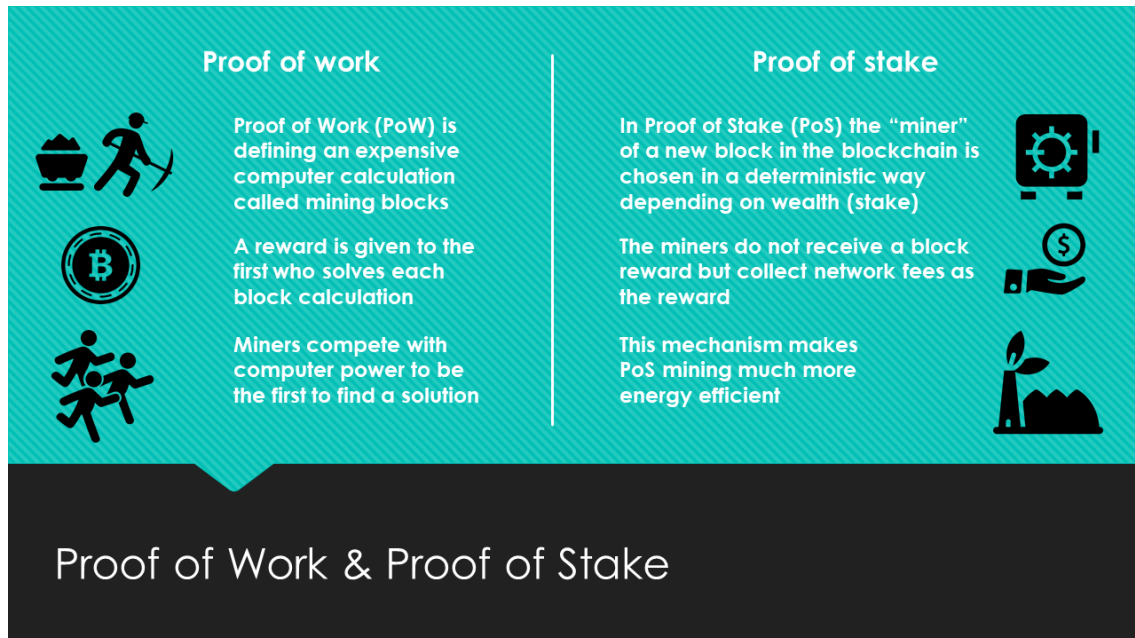


Abbildung 18: Proof of Work vs. Proof of Stake

Quelle:[Bad17]

Wie funktioniert die Blockchain?

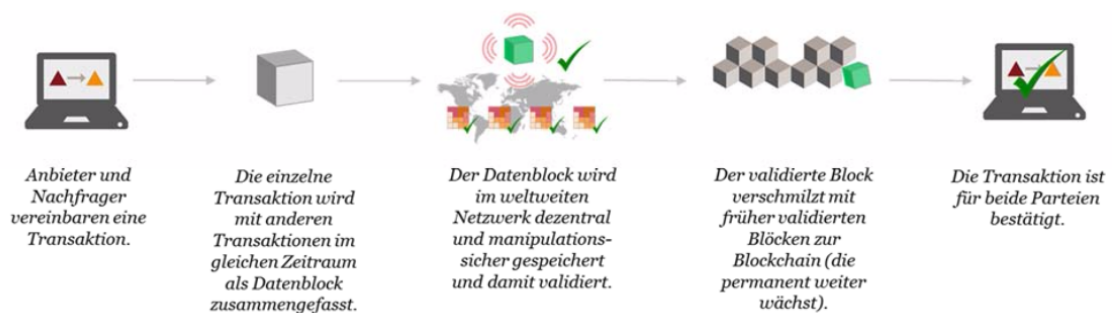


Abbildung 19: Ablauf einer Transaktion in der Blockchain [PWC16]

Abbildung 19 illustriert den logischen Ablauf einer Transaktion in der Blockchain. Eine

Transaktion wird zwischen zwei Parteien vereinbart. Diese Transaktion kann im Kontext von beispielsweise Kryptowährungen beinhalten, dass eine Währung von Partei A nach Partei B transferiert wird. Diese Transaktion wird mit anderen Transaktionen zusammengefasst und durch den Konsensmechanismus kryptografisch verifiziert und validiert. Nach diesem Vorgang wird der Block in die Blockchain aufgenommen und die Transaktion wird für beide Parteien bestätigt.

Blockchain-Technologien Mittlerweile sind zahlreiche Blockchain-Technologien und Ausprägungen vorhanden. Während der ersten Projektphase wurden verschiedene Implementierungen betrachtet.

R3 Das R3 (R3CEV LLC) Konsortium 2 wurde im Jahr 2014 gegründet. Mitglieder dieses Konsortiums sind Vertreter des Finanzsektors. Bisher wurde eine Open Source Plattform namens Corda erstellt. Diese Plattform legt ihren Fokus auf komplexe finanzielle Transaktionen.

Hyperledger Project Die Linux Foundation gründete im Dezember 2015 das Hyperledger Projekt. Dieses Projekt entwickelt Blockchain-Lösungen in sämtlichen Bereichen der Wirtschaft. Referenzen dieses Projektes sind der Blockchain Explorer, die Blockchains Fabric, Iroha und Sawtooth Lake. Besonders die autorisierte Blockchain-Infrastruktur Hyperledger Fabric, welche besonders von IBM und Digital Asset entwickelt wurde, stellt eine weit entwickelte Blockchain mit zahlreichen neuen Möglichkeiten dar.

Ethereum Ethereum wurde von Vitalik Buterin und Gawin Woods gegründet und fokussiert sich auf die Implementierung von Smart Contracts. Dieses System verfolgt die Vision eines "Weltcomputers". Das bedeutet, dass die öffentliche Ethereum Blockchain ihren Fokus auf das Ausführen von dezentralen Applikationen und Smart Contracts hat. Ethereum hat für Smart Contracts eine eigene Skriptsprache namens Solidity entwickelt. Mit einer Marktkapitalisierung von weit über einer Milliarde Dollar steht diese Blockchain direkt hinter Bitcoin auf Rang zwei hinsichtlich der Verbreitung.

Aufgrund dessen, dass Ethereum zu Beginn des Projekts (Ende 2016) die stabilste und verbreitetste Lösung war, wurde diese als Basis für die Implementierung genutzt.

4.1.3. Zielsetzung

Mit der initialen Idee, Energieverbraucher sowie kleine und mittelgroße Energieproduzenten besser in den Energiemarkt zu integrieren und einen Peer-To-Peer-Energiehandel

zwischen diesen, womöglich sogar benachbarten Haushalten, durchführen zu lassen, ist das Teilprojekt gestartet.

Die Verwendung der Blockchain-Technologie spielte eine zentrale Rolle, sodass die Überlegungen stets basierend auf einer Blockchain stattgefunden haben. Schließlich hat es sich die Teilgruppe *energy of Value* zum Ziel genommen, zu untersuchen, ob ein dezentraler Energiehandel mithilfe einer Blockchain realisiert werden kann. Im Rahmen dieser Kernfrage sollte ebenfalls erörtert werden, ob die markt- und netzseitigen Komponenten des Energiemarktes bei einem Energiehandel über die Blockchain berücksichtigt werden können.

Durch die Beantwortung der Kernfrage sollten ferner eine Reihe weiterer Fragen beantwortet werden. So war ein wichtiger Aspekt, die Potentiale, die mit der Blockchain-Technologie ermöglicht wurden, an einem konkreten Beispiel herauszuarbeiten. Ebenso sollte die Frage der Umsetzung einer Blockchain-Anwendung geklärt werden, sodass schließlich sowohl softwareseitig als auch hardwareseitig eine sinnvolle Lösung gefunden werden musste.

Da es sich bei der Untersuchung der o.g. Fragestellungen um eine erste Potentialanalyse mit prototypischer Implementierung gehandelt hat, musste eine Abgrenzung vorgenommen werden. Sicherheitsthemen, Berücksichtigung steuerlicher Aspekte oder normativer Vorgaben sowie auch Fragen der Skalierbarkeit würden den Rahmen sprengen und sind demnach kein fester Bestandteil der Untersuchungen gewesen.

4.2. Vorgehensweise

Um die Ziele zu erreichen und die o.g. Fragestellungen zu untersuchen, wurde neben einer Konzeptionierung eines Systems zum Energiehandel auch eine prototypische Implementierung vorgenommen. Dabei galt das strategische Ziel eines dezentralen Energiehandels über die Blockchain stets als Rahmen für die Konzeption des Energiehandels. Durch die Implementierung konnte deutlich gemacht werden, was durch die Blockchain-Technologie grundsätzlich möglich war und wie die Eigenschaften der Blockchain auf ein Energiemarktmodell übertragen werden konnten. Zu Beginn des Projekts wurde zunächst eine eigene Blockchain aufgesetzt, auf deren Grundlage auf zunächst manuelle Art und Weise Transaktionen und Calls ausgeführt werden konnten. Das anfänglich verwendete Entwicklungsframework *Truffle*² ermöglichte dies auf eine sehr intuitive Art, sodass durch die Entwicklungsplattform besonders der Aufbau des Smart Contracts in iterativen Schritten verbessert, getestet und diskutiert werden konnte. Die Entwicklung des zentralen Bausteins, des Smart Con-

² Weitere Informationen zu *Truffle*: <http://truffleframework.com/>

tracts, führte dazu, dass weitere Überlegungen bzgl. des Energiemarktmodells stattfinden mussten. Dazu wurden unterschiedliche Ansätze herausgearbeitet und miteinander verglichen. Durch iterative Optimierungen konnte schließlich ein sinnvolles Zielmodell konkretisiert werden, welches in Kapitel 4.3 näher erläutert wird. Besonders durch Recherchen ähnlicher Ansätze, kreative Ideenfindung und nicht zuletzt auch durch die Unterstützung externer Experten wurde das finale Energiehandelskonzept erarbeitet. An dieser Stelle gilt unser Dank besonders dem Energiemarktexperten Hendrik Brockmeyer.³ Da Truffle als Entwicklungsframework (zu der Zeit) noch sehr beschränkt und stark auf die Entwicklung eines Smart Contracts ausgerichtet gewesen ist, hat sich das Team für eine eigene Lösung entschieden und diese schließlich umgesetzt. Es wurde eine eigene Test- und Entwicklungsumgebung aufgebaut, welche basierend auf einer serverbasierten node.js-Anwendung durch einen Parity-Client mit der dauerhaft laufenden, privaten Blockchain interagieren konnte. Mithilfe der Testumgebung konnten schließlich weiterführende Anwendungsmöglichkeiten entwickelt werden, sodass ein automatisierter Energiehandel erfolgen konnte. Durch die Hardwarekomponente in Form eines RaspBerry Pi's wurde eine serielle Anbindung an einen Smart Meter ermöglicht und demnach ein realitätsnaher Prototyp mit den wesentlichen Funktionen, die software- und hardwareseitig für den Energiehandel notwendig waren, entwickelt.

4.2.1. State of the Art

Um sich die Eigenschaften und die Funktionsweise der Blockchain-Technologie anzueignen, wurde zum Beginn der Projektphase eine Einarbeitungsphase durchgeführt. In dieser wurden die Grundfunktionalitäten der Blockchain erörtert, erste Implementierungen vorgenommen und auch eine Analyse des State of the Art durchgeführt. Dabei galt neben der Einarbeitung in die Blockchain-Technologie, die Verschaffung eines Überblicks über aktuelle Projekte als wesentlicher Aspekt. So konnten die Erkenntnisse bei bestehenden Blockchain-Anwendungen und Projekten in die Konzeptionierung eingebunden werden. Die Literaturrecherche ergab zahlreiche praktische Projekte im Kontext der Blockchain, welche in folgender Abbildung dargestellt werden.

³ Weitere Informationen zu enera unter <http://www.energie-vernetzen.de/>.

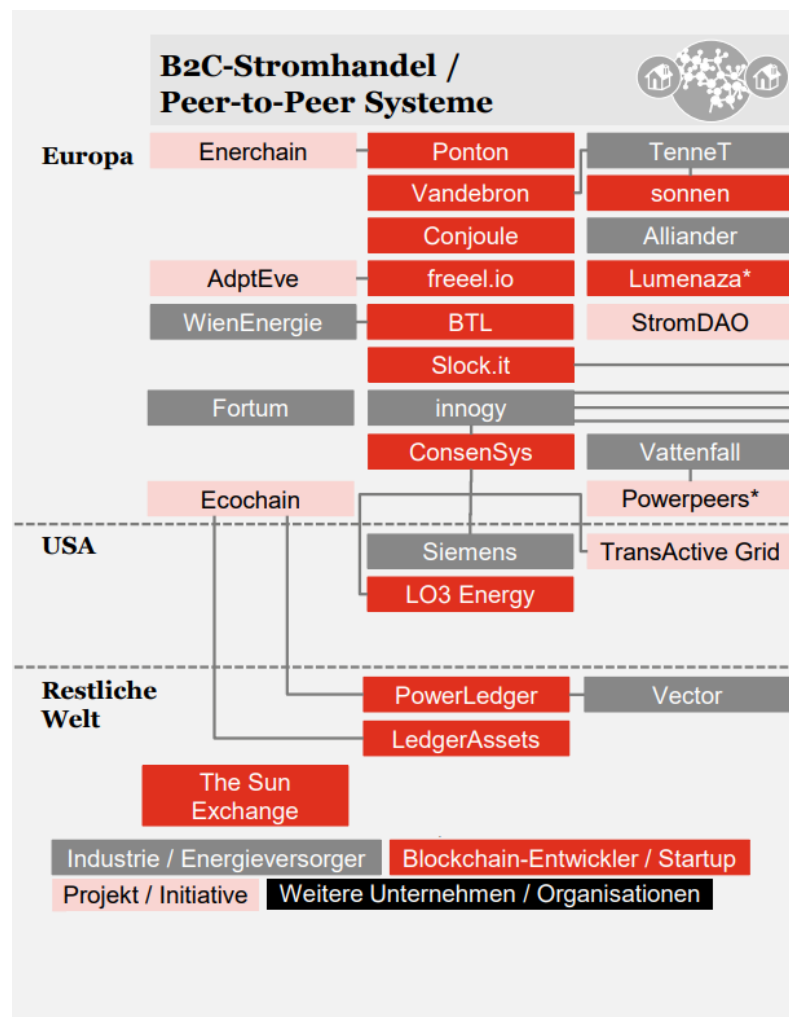


Abbildung 20: Aktuelle Projekte zum Thema B2C/P2P [DS17]

In Abbildung 20 werden aktuelle Projekte aufgezeigt, die sich mit B2C-Stromhandel bzw. mit P2P-Handel befassen. Zahlreiche Start-Ups befassen sich mit der Entwicklung von Geschäftsmodellen im Energiesektor in Verbindung mit Blockchain-Technologien[PWC16].

Das Brooklyn Microgrid, entwickelt von LO3 Energy, ist ein Beispiel für die isolierte Umsetzung einer Peer-to-Peer Handelsplattform mit dezentral erzeugtem Strom. Das PowerLedger-Projekt verfolgt den Vertrieb und die Dokumentation von erneuerbaren Energien in einem offenen Stromnetz[Leg17]. Die Analyse bestehender Projekte in Forschung und Industrie ergab jedoch die Notwendigkeit der Berücksichtigung der Netzkomponente als sowie auch aktueller Marktakteure[HD17].

Wissenschaftlich fundierte Arbeiten befassten sich weitestgehend mit theoretischen An-

sätzen bezüglich des Potentials der Blockchain-Anwendung. Der Erfolg der Blockchain hängt u.a. von den Lösungen derzeit bekannter technischer Probleme ab. Das Bewusstsein der Marktteilnehmer ist von ebenso großem Interesse. Ein Markt ohne Intermediär und somit ohne Regulator ist mit Skepsis verbunden. Die Studie *Where Is Current Research on Blockchain Technology a Systematic Review* von Yil-Huumo et al. untersucht schwerpunktmäßig die Probleme der Blockchaintechnologie.[YHKC⁺16] Laut der Studie befassen sich 80,5 Prozent der analysierten Forschungen mit der Bitcoin-Blockchain. Demnach muss die wissenschaftliche Betrachtung alternativer Blockchain-Technologien (wie beispielsweise Ethereum) vertieft werden. Daher ist eine weitere Motivation dieser Projektarbeit das Erforschen und Bewältigen der gegebenen Problematiken der Blockchain.

Ein großes Problemfeld stellt beispielsweise die Skalierbarkeit einer Blockchain-Anwendung dar. Softwarelösungen im Bereich des Energiemarktes verlangen nach angemessener Performance. Eine Abnahme der Performance durch Steigerung der Nutzerzahl wäre demnach als kritisch zu betrachten. Ein Problem bezüglich der Benutzerfreundlichkeit ist der nicht erneut generierbare Private Key des Accounts. Sobald der Nutzer diesen verloren hat, gibt es aus Sicht der Blockchain keine Möglichkeit diesen neu zu generieren.

4.2.2. Experteninterviews

Im Rahmen des Projekts wurden Gespräche mit Experten geführt, sodass Einschätzungen und Hinweise erfahrener Experten in das Energiehandelskonzept einfließen konnten. So konnten neue Impulse gesetzt werden und das Konzept wurde an die wesentlichen Eigenschaften des bestehenden Energiehandels ausgerichtet. Es fanden zwei Gespräche mit dem Senior Projektkoordinator Smart Market aus dem SINTEC-Schaufensterprojekt enera statt. In diesen sollte zum einen ein besseres Verständnis des heutigen Energiemarktes- und Energiehandels entwickelt werden, sodass ein Energiehandelskonzept entwickelt werden konnte, welches sich an dem heutigen Marktmodell orientiert. Zum anderen wurde auch das entwickelte Konzept vorgestellt und durch hilfreiche Diskussionen optimiert.

Im Rahmen der Gespräche wurden neben dem allgemeinen Smart Contract Design auch konkrete Fragestellungen wie die Einbindung des Konzepts in das heutige Energiemarktmodell oder verschiedene Szenarien beim Energiehandel diskutiert. Letztere werden in Kapitel 4.3.4 näher erläutert. So wurden besonders die zeitliche Aufteilung eines Tages in 96 15-Minuten-Pakete sowie die ex ante Prognoseplanung im Energiehandelskonzept berücksichtigt und im Nachgang der Diskussionen umgesetzt.

Weiterhin konnte das Team ein Gespräch mit dem ebenso erfahrenen Energiemarkt- und Blockchain-Experten Prof. Dr. Jens Strüker führen. In diesem wurden besonders die Her-

ausforderungen, die mit dem Einsatz der Blockchain-Technologie einhergehen, diskutiert und das Team wurde auf dem Weg der Konzeptentwicklung und Prototypentwicklung auf wichtige Hindernisse aufmerksam gemacht.

Ein wichtiger Aspekt war die Frage der Verantwortung zur Einhaltung gesetzlicher Rahmenbedingungen beim direkten Peer2Peer Energiehandel zwischen zwei privaten Haushalten. Ebenso wurde darauf hingewiesen, dass es sinnvoll ist, eine innovative Idee so zu konzipieren, dass diese bestenfalls vor Nachahmern geschützt werden kann und dass langfristig Profit erwirtschaftet werden kann. Diese Fragen waren es nämlich, die auch große IT-Dienstleister, die sich parallel mit ähnlichen Themen beschäftigten, zu lösen versuchen. Jedoch stand bei der Projektgruppe der Forschungsaspekt im Vordergrund und es sollte primär das Potential der Blockchain-Technologie untersucht werden statt ein marktreifes Produkt zu entwickeln.

Besonders die Frage, was mit dem überbleibenden „Reststrom“ passiert, wurde innerhalb des Teams intensiv diskutiert, sodass dieser Aspekt schließlich durch Anpassung des Energiehandelskonzepts und entsprechender Implementierungen Berücksichtigung fand. Ferner wurden weitere interessante Themen diskutiert, die sich jedoch zusammenfassend darauf beziehen, dass bei der Konzept- und Prototypentwicklung kritische Aspekte wie z. B. die Veröffentlichung vertraulicher Daten auf der Blockchain stets zu berücksichtigen und bestenfalls direkt mit einzubeziehen ist. So wurde basierend auf diesen Hinweisen stets Wert darauf gelegt, dass Marktmechanismen des heutigen Energiemarktes berücksichtigt wurden, Datenschutzaspekte nicht außer Betracht gelassen wurden und schließlich auch die Schnittstelle zum Smart Meter eine wesentliche Komponente des Prototyps darstellte. Dadurch konnten die Experten schließlich dazu beitragen, dass die Konzept- und Prototypentwicklung von den eingebrachten Erfahrungen und der Expertise profitierte.

4.3. Konzeptionierung

Die Konzeptionierung war für die Teilgruppe ein wichtiger Prozess. Dadurch konnte der bisherige Energiemarkt verstanden und Probleme identifiziert werden. Der Fokus auf die Blockchain-Technologie war beginnend zweitrangig, da vorerst das Marktdesign eines dezentralen Energiemarktes höher priorisiert wurde. Im Verlauf der Konzeptionierung zeigte sich, dass das entwickelte Marktdesign mit Blockchain-Technologie umgesetzt werden konnte.

4.3.1. Anforderungserhebung

Die Anforderungen ergeben sich aus der Problemstellung sowie der Zielsetzung: Es soll ein Energiehandel über die Blockchain ermöglicht werden, der sowohl Markt- als auch Netzkomponenten umfasst. Auf Grundlage dessen, hat sich die Teilgruppe einen grundlegenden Anwendungsfall erstellt. Dieser wird in Abbildung 21 abgebildet.

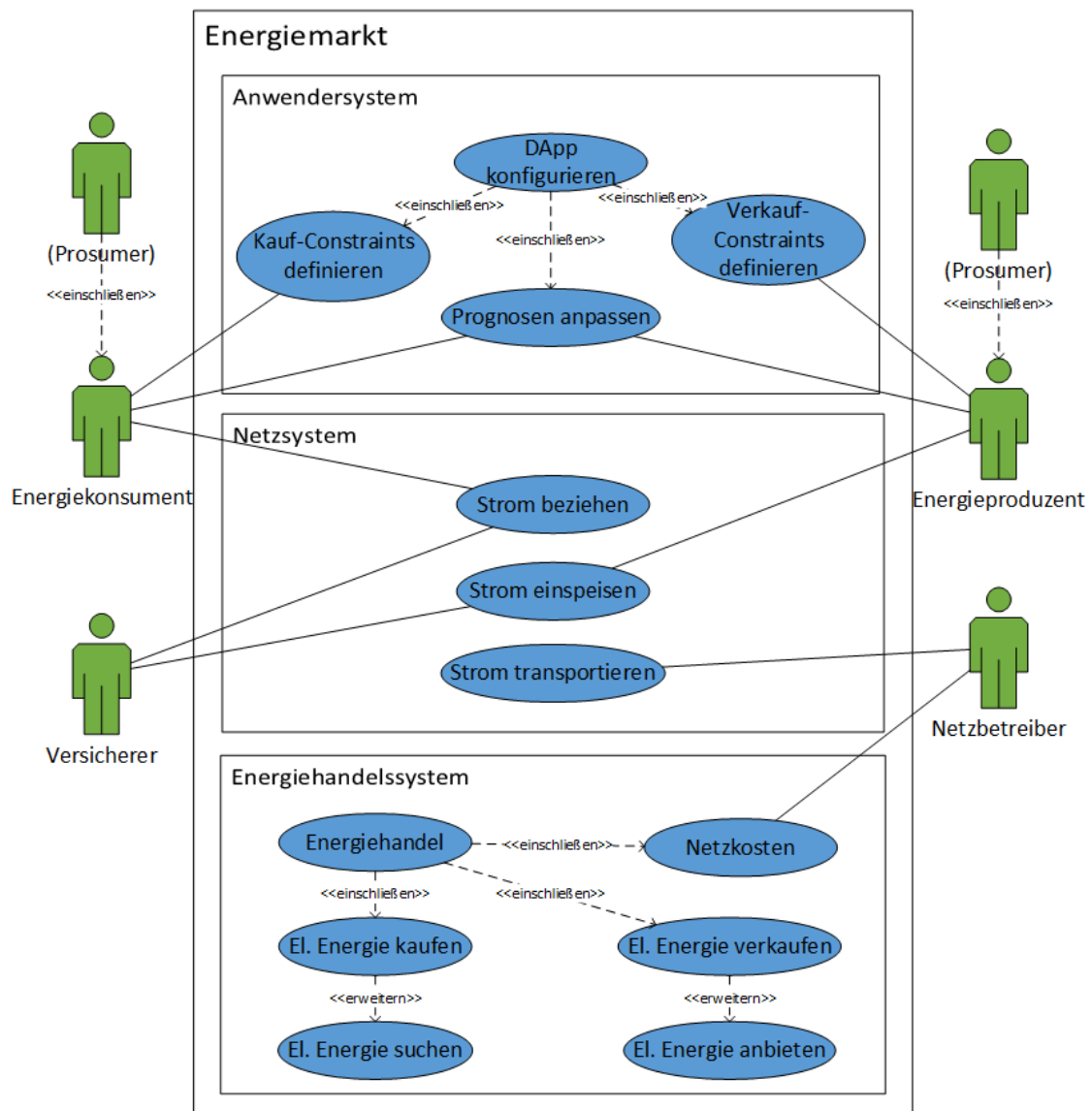


Abbildung 21: Energiemarkt und Energiehandel [Eigene Quelle]

Der Anwendungsfall beinhaltet zum einen die wesentlichen, im Folgenden Unterkapitel

näher erläuterten Akteure sowie deren grundlegenden Aufgaben. Die Aufgaben im Energiemarkt lassen sich in die drei Teilsysteme Anwendersystem, Energiehandelssystem und Netzsystem unterteilen.

Das Anwendersystem ermöglicht es dem Energieproduzenten bzw. dem Energieproduzenten die Decentralized Application (DApp) zu konfigurieren und damit grundlegende Parameter zum automatisierten Energiehandel zu setzen. Der Energiekonsument setzt die Kauf-Constraints nach individuellen Vorzügen. So ist es bspw. möglich, regionalen, "grünen" oder auch günstigen Strom zu bevorzugen und die Anwendung darauf auszurichten. Der Energieproduzent hat ebenfalls die Möglichkeit, Verkauf-Constraints zu definieren, sodass primär Verkaufspreise gesetzt werden können.

Weiterhin haben der Energiekonsument und der Energieproduzent die Möglichkeit, die Prognose manuell anzupassen. Grundsätzlich erfolgt der automatisierte Energiehandel auf Basis zugrundeliegender Prognosen. Diese werden im übernächsten Unterkapitel näher erläutert. Falls sich jedoch an den täglichen Last- bzw. Produktionsprofilen ändert, so wird eine manuelle Anpassung ermöglicht. Änderung der Profile können bspw. durch geplante Urlaubsreisen (und damit stark verringerter Energieverbrauch) oder Wartungsarbeiten an der Photovoltaikanlage (und damit kurzfristig ausgesetzte Energieproduktion) auslösen.

Die im Anwendersystem konfigurierten Parameter sowie die definierten Constraints fließen im Energiehandel ein. Die auf Basis prognostizierter Energieproduktion Energiemenge wird gemäß eingestellten Constraints auf der Blockchain angeboten. Ebenso wird auf Grundlage des prognostizierten Energieverbrauchs entsprechend im Angebotsportfolio gesucht und entsprechend der Constraint-Parameter gekauft. Individuelles Konsumentenverhalten kann durch Berücksichtigung von Constraints wie regionaler oder günstiger Energie berücksichtigt werden.

Neben den autonomen Kaufs- und Verkaufsmechanismen hat auch der Netzbetreiber die Möglichkeit, Netzkosten zu definieren und damit in den Energiehandel eingreifen zu können. Netzkosten können je nach Infrastruktur für bestimmte Netzregionen unterschiedlich ausfallen. Je nach Netzauslastung kann der Netzbetreiber die Kosten, die durch die Nutzung des Netzes anfallen, konfigurieren. Der Fokus der Implementierung liegt jedoch nicht auf der Konfiguration der Netzkosten über die DApp, sodass eine einmalige Angabe der Netzkosten zu Beginn einer Handelsperiode zunächst ausreicht und eine "hard-coded"-Variante ausreicht. Die Einbindung der Netzkosten ist demnach gegeben. Der Fokus des Prototyps wird auf den Energiehandel zwischen den Marktakteuren Energiekonsument und Energieproduzent gelegt.

Schließlich wird mit dem Netzsystem auch der physische Stromfluss berücksichtigt. Der Energiekonsument hat die Möglichkeit, Strom zu beziehen, der Energieproduzent die Mög-

lichkeit Strom einzuspeisen. Der Marktakteur Prosumer kann je nach Last- bzw. Produktionsprofil ein der beiden Rollen einnehmen. Der Netzbetreiber hat im Netzsystem die Aufgabe die Infrastruktur bereitzustellen, sodass der Strom transportiert werden kann. Als Versicherer muss je nach Prognoseabweichung oder Ungleichgewicht des Marktes Strom bezogen bzw. Strom eingespeist werden können.

4.3.2. Akteure

Der heutige Energiehandel erfolgt über diverse Zwischenhändler, bei denen der Endkunde einen Energielieferanten beauftragt, der oft wiederum verschiedene Dienstleister (z. B. Makler, Broker) beauftragt, Energie an der Börse zu kaufen. Zu den klassischen Marktakteuren des heutigen Energiemarktes zählen Erzeuger, Übertragungsnetzbetreiber, Verteilnetzbetreiber Bilanzkreisverantwortliche, Energielieferanten bzw. -händler, Messdienstleister sowie Endkunden. Zwischen diesen Marktrollen erstreckt sich ein komplexes Netzwerk aus unterschiedlichen bilateralen und multilateralen Vertragsbeziehungen, bei denen Beauftragungen und Verantwortungsübertragungen sowie infrastrukturelle und finanzielle Aspekte festgelegt und ausgehandelt werden. In Abbildung 22 sind die wesentlichen Marktakteure sowie deren Vertragsbeziehungen dargestellt. So führt dies schließlich dazu, dass zahlreiche Akteure am Energiemarkt mitwirken, nach profitablen Geschäften streben und aus einem dynamisch schwankenden Strompreis einen statischen Endkundenpreis festlegen können. In den Preisen sind neben Risikoprämien stets auch Gewinnaufschläge inbegriffen.

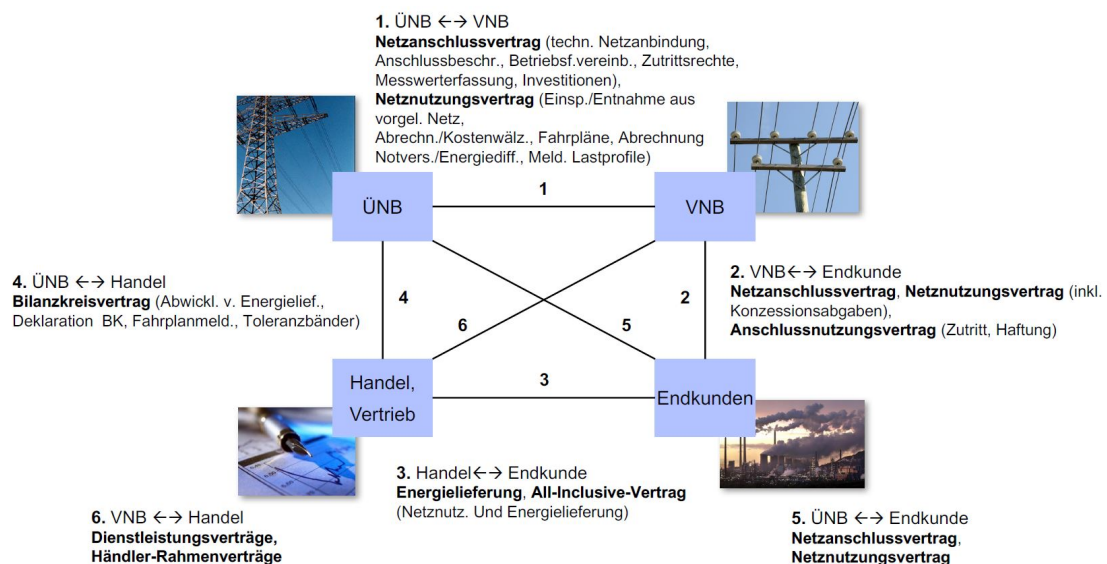


Abbildung 22: Marktakteure und Vertragsbeziehungen heute

Durch den Einsatz einer Blockchain als Energiehandelsplattform wird ein direkter Energiehandel zwischen den Energiekonsumenten und -produzenten ermöglicht. Der grundlegende Vertrag, in dem alle Marktakteure vertreten sind, ist der Smart Contract. Dieser bildet über die Blockchain die zentrale Schnittstelle zu allen relevanten Marktakteuren. In Abbildung 23 werden zum direkten Vergleich die Akteure des hier vorgestellten Energiehandelskonzepts sowie deren gemeinsamer Vertrag, dem Smart Contract, innerhalb der Blockchain dargestellt. Im Folgenden werden die Marktakteure vorgestellt, die am Blockchain-Energiehandel beteiligt sind.

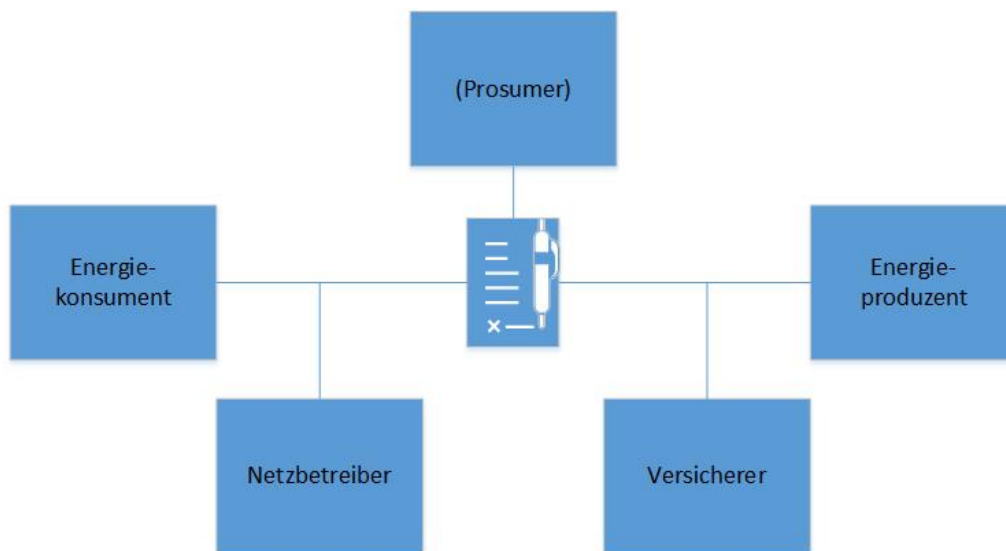


Abbildung 23: Marktakteure und Vertragsbeziehung in Zukunft [Eigene Quelle]

Energiekonsument Der Marktakteur Energiekonsument entnimmt elektrische Energie aus dem Stromnetz. Durch entsprechende Verbrauchsquellen stellt der Energiekonsument eine Netzlast dar. Dieser Marktakteur hat ein Interesse daran, günstig die individuell privilegierte Art von Energie (z. B. Grünstrom, regionalen Strom, etc.) zu beziehen und am Energiemarkt zu erwerben. Er selbst weist ein wiederkehrendes Verbrauchsmuster auf, welches durch Prognosen näherungsweise prognostiziert werden kann. Basierend auf Energiepreisen hat dieser teilweise die Möglichkeit, seine Last im Rahmen der individuellen Elastizität so anzupassen, dass dessen Lastprofil (prognostizierte Last) sich an den zeitlich variierenden Energiepreisen orientiert. Weiterhin versucht der Energiekonsument, je nach individueller Ausprägung, den prognostizierten Energieverbrauch an dessen tatsächlichen Verbrauch anzupassen. Beispiele für diesen Marktakteur sind private Haushalte, Unternehmen, Energiespeicher oder E-Tankstellen.

Energieproduzent Der Marktakteur Energie-

produzent speist Strom in das Netz ein. Mittels Stromerzeugungsanlagen wie Windkraft-, Photovoltaikanlagen oder Biomassekraftwerke produziert der Energieproduzent elektrische Energie. Dieser Marktakteur hat ein Interesse daran, den produzierten Strom zu möglichst hohen Preisen am Energiemarkt zu verkaufen. Je nach Stromerzeugungsanlage hat dieser die Möglichkeit, die Stromproduktion an die Energiepreise des Marktes anzupassen. Dabei versucht dieser, die prognostizierten Energiemengen an die tatsächlich produzierten Energiemengen anzugleichen. Prosumer Der Marktakteur Prosumer kann sowohl die Rolle des Energiekonsumenten als auch des Energieproduzenten einnehmen. Dieser hat die Möglichkeit, sich selbst mit Energie zu versorgen, überschüssige Energie am Markt anzubieten bzw. zusätzlich notwendige Energie am Energiemarkt zu erwerben. Die Interessensbestrebungen gleichen sich je nach eingenommener Marktrolle mit denen des Energiekonsumenten bzw. des Energieproduzenten. Netzbetreiber Unter dem Marktakteur Netzbetreiber werden die bestehenden Akteure Übertragungsnetzbetreiber und Verteilnetzbetreiber zusammengefasst. Der Netzbetreiber stellt mit dem Stromnetz die Infrastruktur bereit und ist zuständig für die Frequenzstabilität, Spannungshaltung, Versorgungszuverlässigkeit sowie die Betriebsführung. Demnach trägt dieser Marktakteur die Verantwortung für die Netzstabilität und strebt demnach einen Energiehandel an, der bilanziell ausgeglichen ist und an die infrastrukturellen Gegebenheiten angepasst ist, sodass u.a. die Wahrscheinlichkeit von Netzengpässen minimiert wird. Versicherer Der Marktakteur Versicherer tritt je nach Bedarf als Energiekonsument oder Energieproduzent auf, um ein Gleichgewicht zwischen Einspeisungen und Entnahmen von Energie zu gewährleisten. Dieser muss flexible Reserven elektrischer Energie (Regelleistung, Sekundärleistung und Minutenreserve) vorhalten bzw. aktivieren können und auch als Netzlast (Energiekonsument) agieren können. Der Versicherer tritt nur im unperfekten Markt auf, um Abweichungen von Prognosen oder unausgeglichenem Handel entgegenzuwirken.

4.3.3. Prognosen

Der heutige Energiemarkt basiert auf der Nutzung von aggregierten Standardlastprofilen für Privathaushalte. Sämtliche Verbraucher, die weniger als 100.000 kWh/a Strom verbrauchen, werden mittels dieser aggregierten Profile zusammengefasst. Aus dieser Aggregation resultiert eine Planungsunsicherheit bezüglich der Netzauslastung und des Börsenhandels für Energieunternehmen. Höhere Planungssicherheit lässt sich nur durch eine feinere Granularität der Datengrundlage über den Konsum bzw. die Produktion der Energie erzielen.

Infolge der fortschreitenden Verbreitung digitaler Stromzähler können detaillierte Verbrauchs- und Produktionsdaten einzelner Haushalte in Echtzeit erhoben werden. Mithilfe von Machi-

<i>Paket Nr.</i>	<i>Menge in wh</i>
0	21
1	41
2	42
.	.
.	.
23	233
24	232
.	.
95	22

Tabelle 3: Aufbau einer Prognose

ne Learning-Ansätzen (v.a. Regressionsanalysen) lassen sich anhand dieser Datengrundlage individuelle Lastprofile für einzelne Haushalte und Unternehmen erstellen.

Das nachfolgend erläuterte Marktmodell setzt das Vorliegen von individuellen Lastprofilen voraus, ist daher integraler Bestandteil dessen. Dies ist notwendig, da Strom basierend auf Verbrauchs- und Erzeugungsprognosen einzelner Marktakteure gehandelt wird. Neben höherer Planungssicherheit des Netzbetreibers können individuelle Lastprofile gehandelt werden. Das heißt, dass zu Konsum-Lastprofilen Produktionsprofile erworben werden können. Innerhalb dezentraler Erzeugerstrukturen können so einzelne Konsumenten und Erzeuger zusammengebracht werden und direkt untereinander handeln.

Exakte Prognosen setzen hochwertige Datenbestände voraus. Aktuelle Forschungen zeigen, dass hinreichend granulare Datenbestände mit digitalen Stromzählern gewonnen werden können, digitale Zähler teilweise jedoch noch fehlerhafte Werte liefern [Bri16]. Aus feingranularen, qualitativ hochwertigen Prognosen folgen für die Anwender niedrige Pönalzahlungen und für den Netzbetreiber eine effizientere Netzplanung, da die erforderliche Ausgleichsenergie minimiert wird. Ausgleichsenergie oder Regelenergie wird dann produziert, wenn weniger Strom verbraucht, als prognostiziert wurde.

Im Energiesektor wird ein Lastprofil eines Tages in 96 Gültigkeitszeiträumen à 15 Minuten unterteilt. Diese Zeitreihe bildet die Grundlage des blockchainbasierten Ansatzes. Die einzelnen 15-Minuten-Intervalle werden nachfolgend als Pakete bezeichnet und enthalten neben einer fortlaufenden Nummer für den jeweiligen Zeitpunkt (0-95) auch die Menge an Strom, die in diesem Zeitraum verbraucht, produziert, aber auch prognostiziert wurde, sowie die scharfe Zuordnung des jeweiligen Marktakteures. Beispielsweise entspricht das Paket mit der Nummer 0 dem Zeitraum 00:00-00:15 Uhr. Innerhalb des Marktmodells sind drei Gruppierungen der Prognosen relevant: der jeweilige Energieverbrauch, die Energieproduktion und die zugrundeliegende Netzauslastung. Der Aufbau der Prognosen

kann Tabelle 3 entnommen werden. Die übliche Darstellungsform von Prognosen wird als Lastprofil durch einen Linien Graph abgebildet.

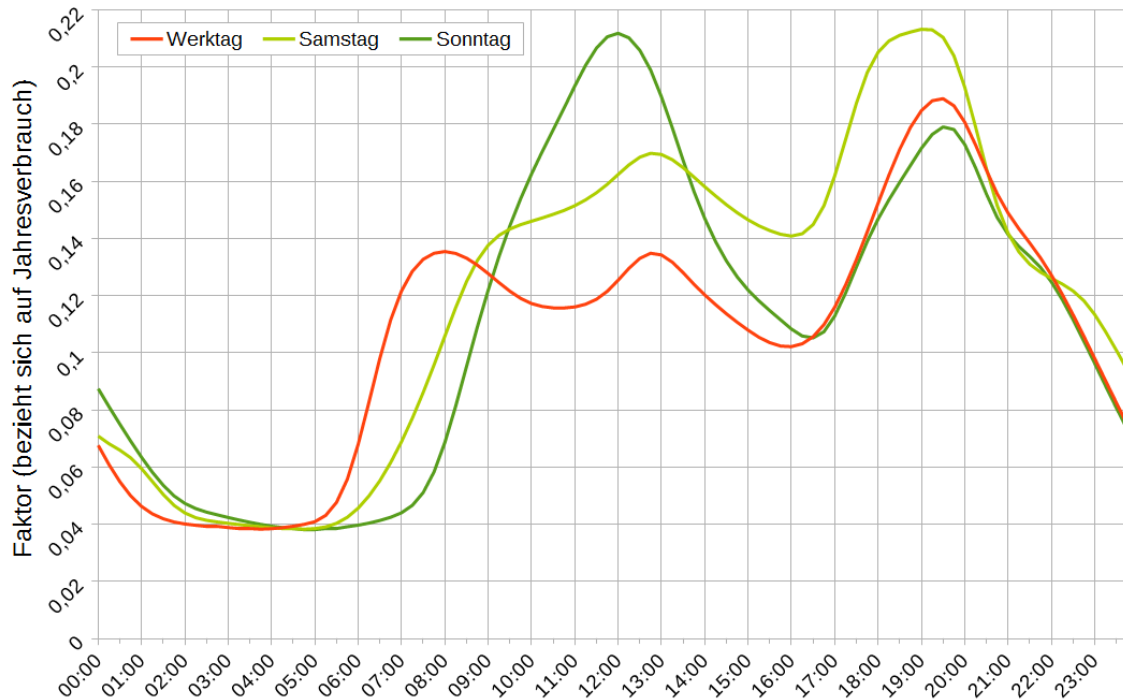


Abbildung 24: Darstellungsweise eines Standardlastprofils [Quelle: Wikipedia]

Durch die Berücksichtigung der Netzauslastung in dem Energiehandelsmodell werden Steuerungsmechanismen erzielt, die zur Vermeidung von möglichen Engpässen beitragen. Der Netzbetreiber veröffentlicht hierbei täglich die Auslastung der jeweiligen Trasse und kann unterschiedliche Preise für die Netznutzung bestimmen. Bei hoher Netzauslastung folgt ein dementsprechend höheres Netzentgelt. Ebenso können bspw. negative Netzkosten bei niedriger Netzauslastung veranlasst werden. Der Netzbetreiber „entlohnt“ den Konsumenten de facto für das Einschalten von elektronischen Geräten. Durch den Handelsmechanismus kann der Stromverkehr passiv kontrolliert werden, da neben dem Strompreis auch der Netzpreis Berücksichtigung findet.

4.3.4. Prognosebasierter Energiehandel

Neben der zuvor erläuterten Notwendigkeit von Prognosen sind sogenannte Pönalen von besonderer Relevanz. Pönalen sind Vertragsstrafen und wirken sich in diesem Modell durch

Gebühren für falsch prognostizierte Verbrauchs- und Produktionsmengen aus. Der Energiehandel erfolgt ex ante und wird täglich abgerechnet. So wird eine feingranulare Strompreiskontrolle und hohe Transparenz erzielt. Der Ablauf des Handelsprozesses wird in Abbildung 25 dargestellt. Dabei werden die Aktionen der einzelnen Akteure auf einer Zeitachse abgebildet. Dieser Abschnitt befasst sich zunächst ausschließlich mit der marktseitigen Konzeption. Die technische Umsetzung wird in Kapitel 4 erläutert.

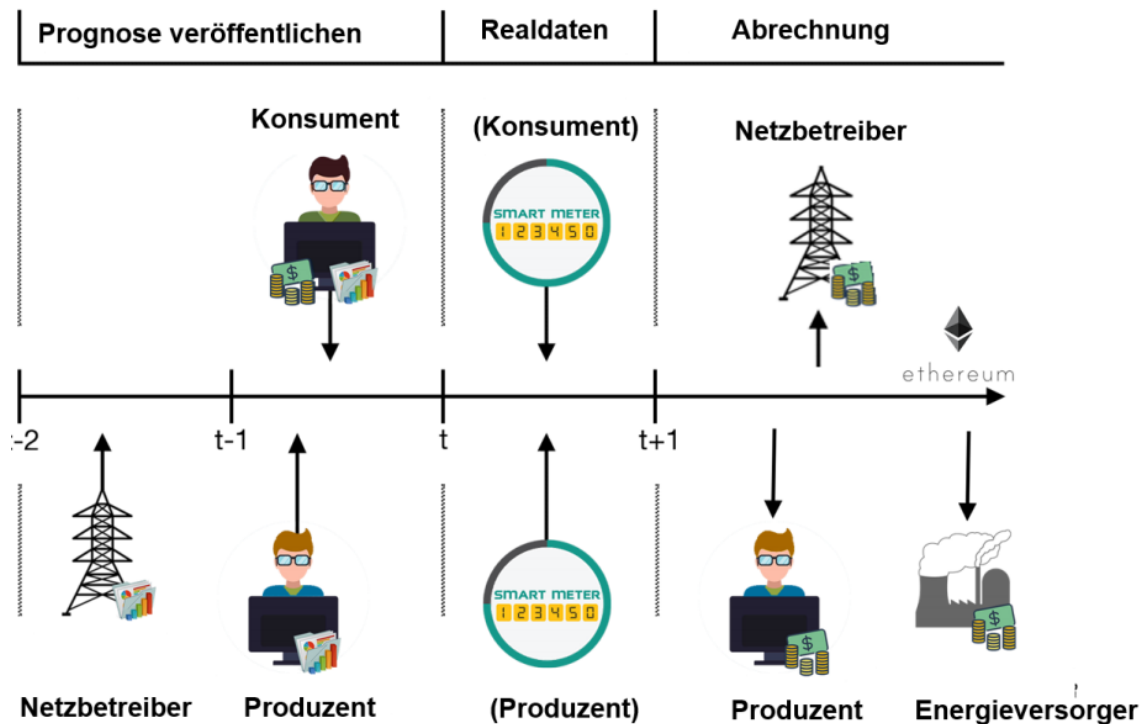


Abbildung 25: Prognosebasierter Energiehandel [Eigene Quelle]

Wie in Abbildung 25 dargestellt, unterteilt sich der prognosebasierte Energiehandel in drei logisch voneinander abgegrenzte Phasen. Die Phase Prognose veröffentlichen repräsentiert das Handelsfenster, welches vor dem Handelstag stattfindet. Hier werden Netzbetreiber, Konsumenten und Produzenten aktiv und veröffentlichen ihre Prognosen über Verbrauch und Produktion für die 96 Viertelstundenintervalle (insgesamt 24 Stunden). Der Zeitpunkt t bis $t+1$ beinhaltet die Erfassung der Realdaten. Hierbei wird über den digitalen Stromzähler die Menge an produziertem und konsumiertem Strom über eine Schnittstelle ausgelesen und in der Blockchain dokumentiert. Abschließend findet die Phase der Abrechnung statt. Diese Phase beginnt nach der Erfassung des 96. Strompaketes (23:45 –

00:00). Abschließend wird die Bilanzierung der Realwerte und der prognostizierten Werte durchgeführt.

Die Prognosen der jeweiligen Akteure werden in einen sogenannten Smart Contract geschrieben. Dies sind kleine Programme, die eigenständig auf Programmierlogik basierende Transaktionen auf Blockchains durchführen können. Der Netzbetreiber veröffentlicht eine Prognose über die Netzauslastung des Handelstages. Das Netz wird hierbei in verschiedene Regionen (Minigrids) unterteilt. Die Prognose beinhaltet die Angabe des jeweiligen Minigrids und eine Angabe zur Netzauslastung des jeweiligen Zeitpunktes. Der Produzent veröffentlicht ebenfalls die prognostizierte Energieproduktion des Handelstages. Der Produzent definiert zudem die Höhe des Kaufpreises der produzierten Energie. Das Netz wird hierbei in verschiedene Regionen (Minigrids) unterteilt.

Die Prognose beinhaltet die Angabe des jeweiligen Minigrids und eine Angabe zur Netzauslastung des jeweiligen Zeitpunktes. Der Produzent veröffentlicht ebenfalls die prognostizierte Energieproduktion des Handelstages. Der Konsument definiert seine Handelsstrategie und veröffentlicht seinen Strombedarf für den kommenden Handelstag in der Blockchain. Der implementierte Handelsmechanismus stellt sicher, dass der Bedarf gedeckt wird. Dabei werden Präferenzen des Konsumenten (Constraints wie z. B. Preis oder Herkunft) berücksichtigt. Der Einkaufspreis setzt sich aus dem Verkaufspreis der Produzenten und der konsumentenspezifischen Netzkosten zusammen. Der Handel kann also zwischen einem Konsumenten und mehreren Produzenten stattfinden (1:n-Beziehung). So wird sichergestellt, dass der Bedarf auch durch verschiedene Produzenten parallel gedeckt werden kann. Um finanzielle Sicherheit zu gewährleisten, bezahlt der Konsument den benötigten Strom unmittelbar mit der zugehörigen Währung des verwendeten Blockchain-Ökosystems, im vorliegenden wurde Ethereum eingesetzt, da Ethereum zum Zeitpunkt der Entwicklung des Prototyps das am weitesten entwickelte Blockchain-Ökosystem mit Unterstützung intelligenter Verträge (Smart Contracts) war.

Während der Phase Realdaten werden durch den digitalen Stromzähler die Daten über Produktion bzw. Verbrauch erhoben. Durch eine optische Schnittstelle werden diese ausgelesen und der Blockchain-Anwendung zur Verfügung gestellt. Nach dem letzten Zeitfenster (oder der Auslieferung des letzten Pakets) werden diese in der Blockchain dokumentiert, sodass die Abrechnung mithilfe eines Soll-Ist-Vergleichs (Bilanzierung) realisiert wird. Die Abrechnung wird vom Smart Contract nach Erfassung der Realdaten implementiert. Es ergeben sich Differenzen, so kommt es zu Pönalzahlungen. Die Höhe der jeweiligen Zahlungen ist ein Marktmechanismus und kann an dieser Stelle noch nicht abgeschätzt werden.

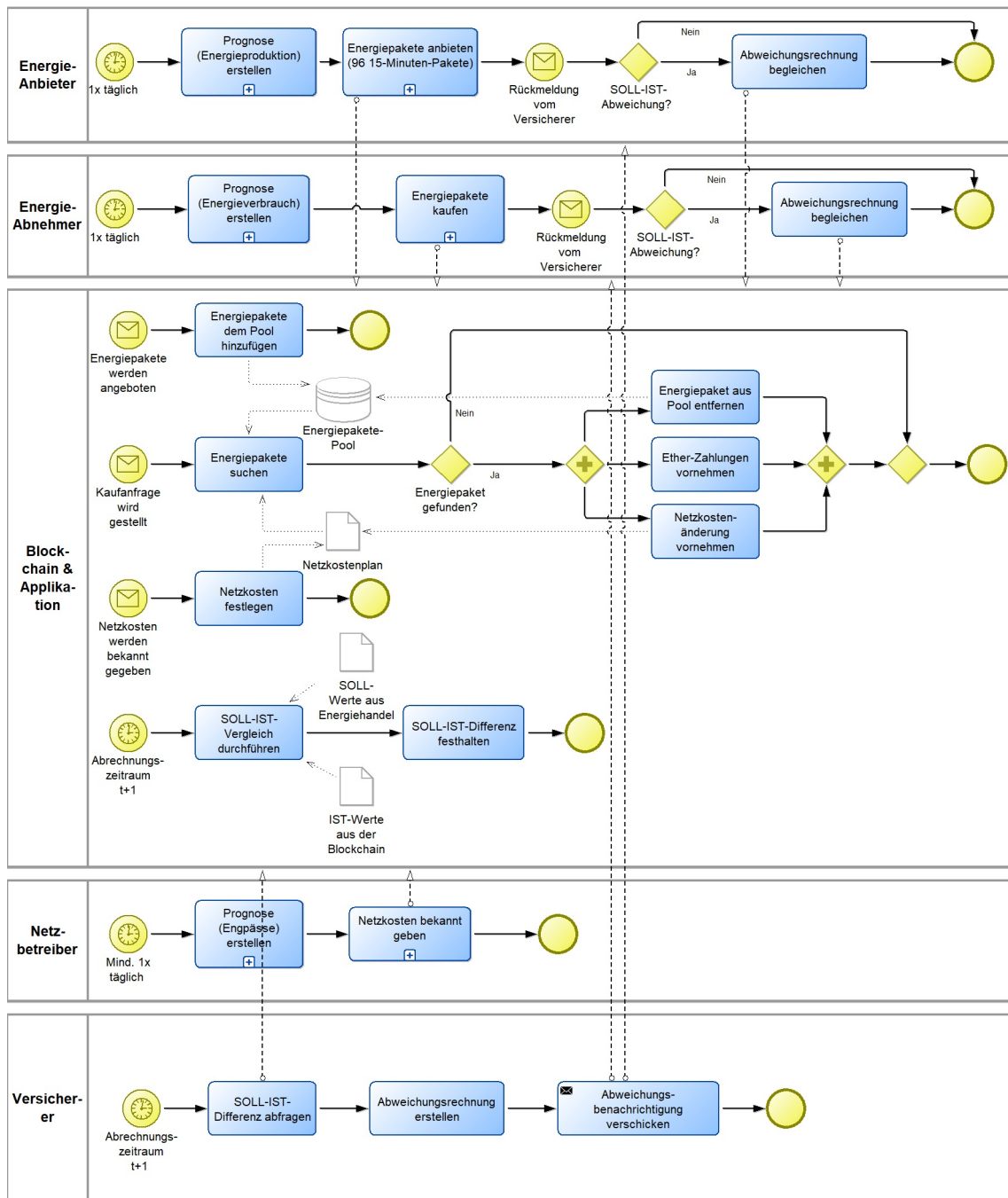


Abbildung 26: BPMN Energiehandel [Eigene Quelle]

Ebenso wird durch den Smart Contract der Netzbetreiber gemäß seiner Leistung entlohnt. Die Besonderheit des Marktmodells stellt die Position des Energieversorgers als Ver-

sicherer dar. Eine exakte Übereinstimmung der Prognose und der Realdaten ist unmöglich zu erzielen, daher müssen Differenzen ausgeglichen werden. Weiterhin ist eine Umstellung zu einem Peer2Peer Handelsmarkt ein langfristiger Prozess und im Zuge dessen weiterhin auf den Energieversorger angewiesen. Eine kurzfristige Umstellung erscheint aufgrund der aktuellen Marktdurchdringung der Energieversorger als sehr unrealistisch. Die Steuerungseffekte, die durch dieses Marktdesign erzielt werden, beziehen sich auf die Entlastung des Netzbetreibers durch die Minimierung von Netzengpässen. Ebenso werden durch die Angabe der Regionen der Akteure lange Transportstrecken vermieden, da diese mit hohen Netzkosten bepreist werden. Außerdem kann das Verbrauchsverhalten der Konsumenten beeinflusst werden, da eine Einhaltung der Prognose von wirtschaftlichem Interesse ist, weil Pönalzahlungen vermieden werden können.

In Abbildung 26 ist die BPMN Darstellung des Konzepts zu sehen, um die einzelnen Schritte visuell nachvollziehen zu können. Das Modell ist in fünf Swimlanes unterteilt. Die Energieabnehmer, Energieanbieter und Netzbetreiber lösen durch die Veröffentlichung der eigenen Prognosen den Mechanismus in der Blockchain und Applikations Swimlane aus. Daraus folgert, dass alle Akteure in diesem System aktiv werden müssen, damit der Energiehandel abgewickelt werden kann. Weiterhin wird deutlich, dass der eigentliche Energiehandel durch die veröffentlichte Prognose des Energie-Abnehmers ausgelöst wird. Im Energiekauf wird die Ether Zahlung direkt veranlasst. Dadurch wird den Produzenten eine Sicherheit gegeben, dass diese für ihre Angabe den festgelgten Betrag kommen. Ferner wird der Soll-Ist Vergleich zum Zeitpunkt $t+1$ automatisch angestoßen. Das System leitet die Ergebnisse an die Energieversorger weiter und werden den Produzenten und Konsumenten in Rechnung gestellt. Die Prozesse wurden so konzipiert, dass diese vollautomatisch ablaufen.

4.4. Implementierung

Die Implementierung stellt neben der Konzeptionierung den Kern der Forschungsarbeit der Teilgruppe dar. Wie einleitend formuliert, geht es darum die technische Machbarkeit eines dezentralen Energiemarktes mithilfe von Blockchain-Technologie zu zeigen. Dieses Kapitel erläutert daher die Umsetzung des zuvor definierten Konzepts.

4.4.1. Architektur

Parallel zur Konzeptentwicklung wurde eine Blockchain-Anwendung entwickelt, um den im vorherigen Kapitel vorgestellten Energiehandel prototypisch umzusetzen. Die Entwicklung des Prototyps hat Aufschluss darüber gegeben, wie ein Energiehandel mit Hilfe einer

Blockchain umgesetzt werden kann. Im Rahmen der Entwicklungsphase wurde nach Fertigstellung des ersten Prototyps eine grundlegende Architekturänderung notwendig. Der erste Prototyp basierte auf dem Ethereum-Entwicklungsframework Truffle. Das Truffle-Framework stellt zugleich eine Entwicklungsumgebung, ein Testframework und auch ein Verbindungsglied zur Ethereum-Blockchain dar. Die Nutzung des Frameworks erleichterte besonders die Smart Contract-Entwicklung, da es über einen integrierten Smart Contract Kompilierer und eine vordefinierte Deploymentfunktion besitzt. Weiterhin konnte durch die Verwendung einer sogenannten Testchain auf einfache Art und Weise eine lokale Blockchain aufgesetzt werden, sodass das Deployment von Smart Contracts und auch die Durchführung von Transaktionen ermöglicht wurden. Mit der Entwicklungsumgebung waren zahlreiche Funktionen bereits vorimplementiert, die das Testen des Smart Contracts enorm erleichterten und so zu schneller Entwicklung dieses wesentlichen Bausteins führten. Bestimmte, in der Konsole ausführbare Anweisungen führten dazu, dass aufwändige Prozesse im Hintergrund angestoßen wurden und eine Prototypentwicklung in kleinem Rahmen ohne hohe Programmieraufwände möglich war. Ebenso bot die Testchain die Möglichkeit, bis zu zehn Nodes zu simulieren. Die Testchain konnte sehr schnell aufgesetzt werden und war so lange aktiv, wie die entsprechende Konsole (z. B. Windows-Eingabeaufforderung, Powershell, Git Bash, o.ä.) geöffnet war. Neben dem initialen Aufsetzen der Blockchain, der Kompilierung und dem Contract-Deployment konnten ebenso die Transaktionen manuell über mehrere parallel geöffnete Konsolen ausgeführt werden. Ein manueller Energiehandel über die Blockchain war demnach bereits im ersten Prototyp im kleinen Stil möglich. Die zahlreichen Voreinstellungen und vordefinierten Funktionen beschränkten jedoch den entwicklungstechnischen Freiraum, der bei der Konzeptentwicklung notwendig war. Ebenso war die Entwicklung einer dauerhaft laufenden Applikation zum automatisierten Energiehandel sowie externe Schnittstellen kaum möglich. Die vielen Einschränkungen führten schließlich dazu, dass sich die Teilgruppe energy of VALUE dazu entschlossen hat, eine eigene Entwicklungsumgebung zu entwickeln und nach den eigenen Vorstellungen und Bedürfnissen einzurichten. Die Entscheidung führte dazu, dass die Komplexität des Prototyps wesentlich gesteigert wurde. Andererseits führte die Entscheidung, sich vom Truffle-Framework zu distanzieren und eine auf die eigenen Bedürfnisse ausgerichtete Umgebung zu erstellen dazu, dass sich ganz neue Möglichkeiten aufgetan haben. Das Ergebnis war schließlich der umfangreiche und vergleichsweise professionellere Prototyp mit zahlreichen zusätzlichen Funktionalitäten. Im Rahmen der Architektur-Umstrukturierung hat sich das Team auch für die Nutzung einer eigenen, dauerhaft verfügbaren Blockchain entschieden. Die für den End-Prototyp eingesetzte Blockchain basiert auf der zweiten Generation der Blockchain-Technologie, die neben Währungstransaktionen auch den Einsatz von Smart Contracts

ermöglicht [PWC16]. Es wird eine private Ethereum-Blockchain verwendet, welche die Grundlage für den Energiehandel bietet. Die Verwendung einer Ethereum-Blockchain war deshalb sinnvoll, da diese anders als Bitcoin neben dem Transfer der eigenen Kryptowährung auch den Einsatz von Smart Contracts ermöglicht. Smart Contracts werden in Kapitel 4.4.4 näher erläutert. Ebenso gilt Ethereum, gemessen an der Marktkapitalisierung, als die zweitgrößte Blockchain nach Bitcoin und ist demnach die bekannteste Blockchain, die den Einsatz von Smart Contracts zulässt.

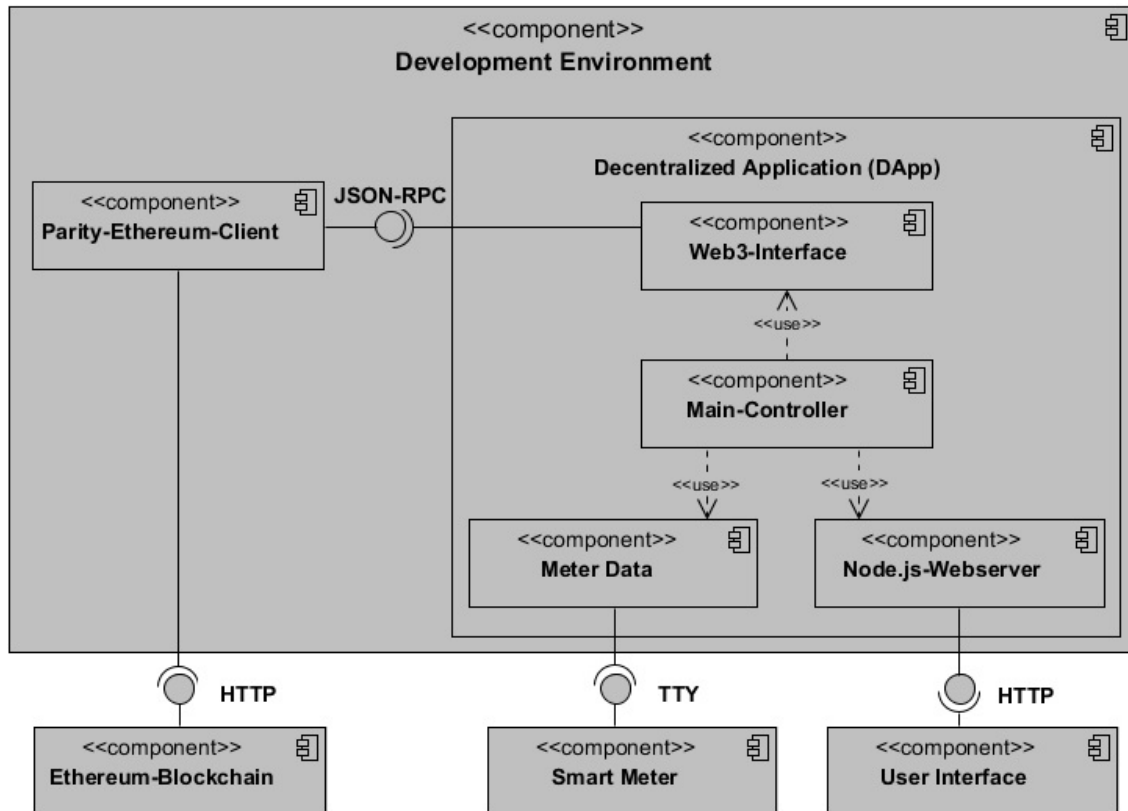


Abbildung 27: Architekturmodell [Eigene Quelle]

Die private Blockchain ermöglichte es, ohne entstehende Transaktionskosten die Entwicklung einer Applikation vorzunehmen und Testabläufe durchzuführen. Da bei einer öffentlichen Blockchain reale Kosten für Transaktionen entstehen, wurde im Rahmen der Anwendungsentwicklung auf eine private und damit kostenfreie Ethereum-Blockchain zurückgegriffen. Die Entwicklung des Prototyps auf Basis einer privaten Blockchain war demnach vollkommen ausreichend und gut geeignet. Da sich bzgl. der Prozesse keine Änderungen zu einer öffentlichen Blockchain ergeben, sind die Ergebnisse auf eine öffentliche Blockchain

direkt übertragbar und auf der öffentlichen Ethereum-Blockchain reproduzierbar.

Die aufgesetzte Blockchain war durch die dauerhaft angeschlossenen und aktiven Raspberry Pi's durchgehend online. Ein Zugriff konnte durch den Aufbau einer Verbindung zur Blockchain jederzeit hergestellt werden, sodass von jedem Standort, an dem Internetempfang vorhanden war, der Energiehandel angestoßen werden konnte bzw. Konfigurationen vorgenommen werden konnten. Durch das Aufsetzen einer eigenen Blockchain wurde die Grundlage für die Entwicklungsumgebung geschaffen. Der Energiehandel konnte prototypisch implementiert, getestet und iterativ weiterentwickelt werden. Das entsprechende Architekturmodell ist in Abbildung 27 dargestellt.

Die Hardware besteht aus fünf Raspberry Pi's der dritten Modellreihe. Diese verfügen über ausreichend Rechenkapazität sowie ausreichend Speicherkapazität für die prototypische Anwendung. Aufgrund der geringen Größe und den geringen Kosten eignen sich die Raspberry Pi's als Hardware-Komponenten für die Entwicklungsumgebung. Eine Reproduzierung bzw. Erweiterung der Entwicklungsumgebung ist demnach ohne hohe finanzielle Aufwände möglich. Einer der fünf Raspberry Pi's übernimmt die Aufgabe des Parity-Ethereum-Clients und ermöglicht somit den Zugriff auf die dauerhaft aktive Blockchain. Die weiteren vier Raspberry Pi's werden zum Energiehandel verwendet, sodass je nach aktivierter Marktrollenfunktion, ein Energiekonsument, Energieproduzent oder Prosumer dargestellt wird. Demnach ist die Komponente DApp (dezentralized Application) vierfach in der Entwicklungsumgebung vorhanden. Der Programmcode der vier DApps ist exakt identisch. Der entsprechende Marktakteur kann durch eine initiale Konfiguration festgelegt werden. Über einen Parity-Blockchain-Client kann mittels einer HTTP-Webschnittstelle auf die Blockchain zugegriffen werden. Der Parity-Client hat sich während der Entwicklungsphase gegen die konkurrierenden eth- und geth-Clients aufgrund einer recht guten Dokumentation, einem guten Entwicklungsstadium sowie fortschreitender Weiterentwicklung hervorgehoben. Die Konfiguration des Parity-Clients konnte demnach unter Verwendung sehr detaillierter Schritt-für-Schritt-Anleitungen implementiert werden. Der Parity-Client stellt eine sog. Node dar, enthält demnach eine Kopie der Blockchain und ermöglicht die Interaktion mit dieser. Die sog. DApp bietet die Möglichkeit, lokale oder serverseitige Prozesse ablaufen zu lassen und schließlich auch über die Node mit der Blockchain zu interagieren. Bei der hier gewählten DApp handelt es sich um eine serverseitige NodeJS-Anwendung, die sich in vier wesentliche Funktionsbereiche unterteilen lässt:

Web3-Interface Über eine JSON-RPC-Schnittstelle kann die eigentliche Applikation mit dem Parity-Client kommunizieren. Das Web3-Interface ist eine spezielle Ethereum-Javascript-API, die eine Bibliothek zur Verfügung stellt, um Informationen zwischen der Node und

der DApp über eine JSON-RPC-Schnittstelle auszutauschen. Dabei wird grundsätzlich zwischen Transaktionen und Calls unterschieden. Während Transaktionen schreibende Prozesse auslösen, die Blockchain mit Informationen erweitern und demnach Mining-Aufwand zur Folge haben, handelt es sich bei Calls um lesende Prozesse ohne Mining-Operationen. Eine Sonderform der Calls stellen Events dar. Diese ermöglichen es, durch Einsatz sog. Listener Veränderungen auf der Blockchain direkt zu erfassen. So können Events verwendet werden, um bestimmte Informationen stets auf dem aktuellen Stand zu halten und u.U. schnell reagieren zu können. Aufgrund dessen, dass npm (Node Package Manager) als Javascript Package Manager verwendet wurde, konnte das Web3-Package problemlos installiert und genutzt werden. Web3 verwendet grundsätzlich ein eth-Objekt (web3.eth) sowie ein ssh-Objekt (web3.ssh) zur Übertragung von Informationen. Transaktionen werden durch Nutzung des eth-Objekts ausgeführt, indem vordefinierte Methodenbezeichnungen angehängt werden. So wird eine Transaktion über den Ausdruck

Listing 1: Solidity Smart Contract Ausschnitt

```
web3.eth.sendTransaction(transactionObject [, callback])
```

angestoßen. Weiterhin können eine Vielzahl vordefinierter Transaktionen und Calls über das web3-Interface durchgeführt werden. Darunter sind eine Reihe von Ausdrücken, die essenziell für die Durchführung des Energiehandels sind. Einige wesentliche Operationen für den Blockchain-Energiehandel werden im Folgenden aufgelistet:

Listing 2: Solidity Smart Contract Ausschnitt

```
web3.eth.sendTransaction(transactionObject [, callback])
web3.eth.call(callObject [, defaultBlock] [, callback])
web3.eth.getBalance(addressHexString [, defaultBlock] [, callback])
web3.eth.getTransaction(transactionHash [, callback])
web3.eth.contract(abiArray)
web3.eth.contract
web3.eth.filter(options)
web3.eth.watch(changes)
```

Bei gewisser Affinität für programmiertechnische Vorgehensweisen wird anhand der Funktionsbezeichnungen deutlich, dass durch die Anwendung dieser Funktionen bereits viele Anwendungsfälle und Implementierungen ermöglicht werden. Die in der Übersicht aufgelisteten Operationen stellen dabei lediglich die am häufigsten verwendeten Funktionen dar.

Node.js-Webserver Die Bereitstellung eines Webservice, ermöglicht dem Anwender über einen Webbrowser aktiv auf die Applikation zuzugreifen. Mittels Web-UI kann sich der

Anwender relevante Informationen anzeigen lassen. Über den Webserver lassen sich konkret z. B. Verbrauchs- bzw. Produktionsdaten anzeigen oder der Energiehandel lässt sich verfolgen. Dabei sind übliche JavaScript-Frameworks wie AngularJS oder React verwendbar, sodass professionelle Darstellungen ermöglicht werden. Das im Prototypen verwendete JavaScript-Framework war aufgrund vorhandener Framework-Kompetenzen das von Google entwickelte Webanwendungs-Framework AngularJS. Neben reiner Anzeige ist ebenfalls eine direkte Interaktion über die Webanwendung möglich. So kann der Anwender über die Web-UI individuelle Constraints zum Energiehandel oder Einstellen von Modi, wie bspw. dem Urlaubsmodus, setzen, um Einfluss auf die Prognosezahlen zu nehmen und aktiv in den Energiehandel einzugreifen. Da es sich dabei um einen bestehenden Webserver handelt, ist eine Anwendung in einer gewöhnlichen App besonders gut realisierbar. Der Zugriff und die Konfiguration dieser Informationen kann in jedem Fall bedienungsfreundlich und ähnlich zu herkömmlichen User Interfaces umgesetzt werden.

Meter Data Über die serielle TTY-Schnittstelle des RaspBerry Pi können Messdaten vom intelligenten Zähler (Smart Meter) in die DApp übertragen werden. Das Smart Meter stellt demnach Messdaten in digitalisierter Form zur Verfügung, sodass diese in der Applikation verarbeitet oder an die Blockchain weitergeleitet werden können. Da die verwendeten RaspBerry Pi's bereits über eine GPIO-Steckerleiste verfügen, wurde die Datenübertragung mittels dieser gewählt. Vorausblickend auf eine etwaige Testinstallation in verschiedenen Haushalten wurde die Wahl des SmartMeters mit dem Energieversorger EWE abgestimmt. Die Wahl fiel dementsprechend auf das EasyMeter Q3D, welches sämtliche regulatorischen Anforderungen hinsichtlich eines normalen Energiezählers erfüllt. Darüber hinaus bietet der eingesetzte Zähler die Möglichkeit, den aktuellen Zählerstand sowohl für Verbrauchs- als auch Erzeugungsdaten über eine optische Schnittstelle auszulesen. Es sei an dieser Stelle angemerkt, dass der übermittelte Zählerstand seitens des Zählers nur alle zwei Sekunden aktualisiert wird. Optimal wäre natürlich eine Datenbereitstellung in Echtzeit. Die geplante Architektur bietet jedenfalls die Möglichkeit, gegebenenfalls auch Energiezähler anderer Hersteller anzubinden. Da zur Abbildung von Anwendungsfällen simulierte Werte sinnvoll waren, wurde zudem ein Simulator für Messwerte entwickelt. Dieser wird im folgenden Unterkapitel näher erläutert.

Main-Controller Der zentrale Funktionsbereich der Applikation ist der Main-Controller. Dieser steuert den Workflow der DApp. Der Main-Controller setzt sich aus den notwendigen Klassen und Methoden zusammen, die z. B. Daten verarbeiten, Prognosen bereitstellen und Transaktionen ausführen, sodass die Funktionalität der Anwendung ermöglicht wird.

Demnach stellt der Main-Controller die zentrale Steuereinheit der Applikation dar und führt autonom, gemäß der parametrisierten Constraints den automatisierten Energiehandel durch. Konkret erhält der Main-Controller in regelmäßigen Abständen Messdaten des Smart Meter und die relevanten Informationen von der Blockchain, um diese verarbeiten zu können. Die verarbeiteten Daten werden dem Anwender über den Webserver grafisch aufbereitet zur Verfügung gestellt, sodass dieser über die Web-UI darauf zugreifen kann. Constraint-Einstellungen oder Prognoseanpassungen werden im Main-Controller entgegengenommen und entsprechend verarbeitet. Weiterhin beinhaltet der Main-Controller den Code des Smart Contracts, welcher einmalig von einem Blockchain-Teilnehmer auf der Blockchain veröffentlicht werden muss. Je nach implementierter Funktionalität im Smart Contract lässt sich der Energiehandel durchführen. Zu den wesentlichen Funktionalitäten zählen die Angebots- und die Kauffunktion sowie die Möglichkeiten zur Ausführung von Calls oder dem direkten Abfangen von Events. Durch entsprechende Depots werden die für den Energiehandel relevanten Informationen ähnlich wie auch in einer herkömmlichen Datenbank gespeichert und sind abrufbar. Zur Identifizierung eines Marktteilnehmers werden die Identifikationsnummern der Stromzähler verwendet. Abhängig davon, ob der Anwender den Handel als Energieproduzent oder als Energiekonsument führt, wird die Energie über die Prognosekomponente des Main-Controllers prognostiziert und auf der Blockchain zu einem ermittelten Preis angeboten bzw. abhängig vom prognostizierten Konsum gekauft. Diese Aktionen erfolgen auf Basis eines zuvor konfigurierten Constraint-Netzes sowie einer künstlichen Intelligenz des Systems. Die dezentrale Applikation ist durch das npm-Package „Architect“ modular aufgebaut. Das zentrale Startmodul löst je nach initialer Konfiguration einen gewünschten Workflow unter Verwendung bestimmter Module durch. So ist es bspw. möglich die Applikation ohne UI oder auch ohne Backend zu starten. Ebenso werden die anfänglichen Konfigurationen im Rahmen des Startmoduls „startApp.js“ geladen.

Die modularen Bausteine der dezentralen Applikation sind in einzelne Plugins unterteilt.

- web3-provider
- web3-provider-mock
- contractAbi
- backend-utils
- smartmeter-mock
- r-client

- backendController
- event-bus
- web-ui
- web-ui-controller
- logger
- controller-buyer
- controller-seller
- controller-demo

web3-provider, web3-provider-mock Das Plugin web3-provider wird benötigt, um die o.g. Funktionen der Web3-Schnittstelle verwenden zu können. Es wird ein Web3-Objekt erzeugt, welches mit der eigenen web3-Bibliothek verwendet werden kann. Das Web3-Objekt besteht wiederum aus einem Eth-Objekt (zur Interaktion mit der Ethereum-Blockchain) und einem ssh-Objekt (zur sog. Whisper-Interaktion, der Kommunikation zwischen DApp und Node). Über den web3-provider kann demnach über einen sog. Remote Procedure Call (RPC) mit der lokalen Node kommuniziert und demnach auch mit der Blockchain interagiert werden.

Falls keine Verbindung zur Blockchain aufgebaut werden kann bzw. soll, kann auf den web3-provider-mock zurückgegriffen. Dieser löst keine Kommunikation mit der Blockchain aus, ermöglicht es jedoch ohne Interaktion mit der Blockchain den Workflow ohne Fehlermeldung fortzuführen und ist demnach lediglich zur Applikationsentwicklung notwendig.

contractAbi Die Funktionen, die durch den Smart Contract aufgerufen werden können, werden durch die contractAbi dargestellt. Die contractAbi stellt das sog. Application Binary Interface (ABI) dar. Die Schnittstelle ermöglicht das kodieren bzw. das dekodieren von Programmcode. So können die in der Programmiersprache Solidity vorhanden Smart Contract Methoden von der Controller-Anwendung gelesen werden. Es können demnach nur diejenigen Methoden (Transaktionen, Calls, Events) aufgerufen werden, die im Smart Contract enthalten und schließlich über die contractAbi zur Verfügung stehen.

backend-utils Im Plugin backend-utils sind für Demonstrationszwecke notwendige .csv-Parser implementiert. Diese können verwendet werden, um Testdaten verwenden zu können, falls keine Echtdateien zur Verfügung stehen. Im Wesentlichen werden das Datum, die Prognosen und die IST-Messwerte eingelesen. Auf Basis dieses Inputs, kann eine Demonstration der Funktionalitäten durchgeführt werden, sodass ein Energiehandel ermöglicht wird, obwohl kein Echtgeräte-Anschluss besteht.

smart-meter-mock Auch das Plugin smart-meter-mock ist notwendig, um Demonstrationen und einen Energiehandel ohne angeschlossenen Smart Meter durchzuführen. Falls ein Smart Meter angeschlossen ist, werden die Daten direkt verarbeitet. In Demonstrationen und zur Entwicklung wird das Plugin smart-meter-mock verwendet, um Testdaten bereitzustellen. Die Testdaten werden in demselben Format zur Verfügung gestellt, wie sie auch aus einem angeschlossenen Smart Meter bereitgestellt werden würden.

r-client Das Plugin r-client stellt die Schnittstelle zu einem Microsoft R-Server bereit und ermöglicht so die Kommunikation mit diesem. Auf dem R-Server werden vom Team energy of TOMORROW Prognosedaten bereitgestellt, welche für den Energiehandel verwendet werden können. Demnach ist dies die direkte Schnittstelle zu dieser Teilgruppe.

backendController Das Plugin backendController stellt die notwendigen Funktionen für den Energiehandel bereit. Die vom Smart Contract über den contractAbi zur Verfügung gestellten Funktionen werden im backendController zu vollwertigen, ausführbaren Funktionen. Die ggf. übergebenen Parameter werden entsprechend der Funktionsart verarbeitet und weitergegeben. Ein Beispiel ist die Funktion setSellerPredictedDay(): Diese Funktion erhält die notwendigen Parameter und führt 96 mal die Funktion setSellerPredictedPackage() aus, sodass mit einem Funktionsaufruf der gesamte Tag gesetzt werden kann. Dabei werden 96 Transaktionen durchgeführt, sodass schließlich 96 Energiepakete (ein gesamter Tag) für den Energiehandel zur Verfügung stehen.

event-bus Durch den event-bus können durch Transaktionen ausgelöste Änderungen der Blockchain gezielt abgefangen werden. Durch einen angesetzten Listener werden Transaktionen festgehalten und können über entsprechende Funktionen ausgelesen und weiterverarbeitet werden.

web-ui, web-ui-controller Die Plugins web-ui und web-ui-controller enthalten den Programmcode zur entsprechenden Anzeige und Input-Verarbeitung auf der Web-UI bzw.

Web-App. Die Informationen werden durch abfangen bestimmter Events aktuell gehalten und auch Konfigurationsparameter, die vom Anwender übergeben werden, können in die Anwendung und in den Energiehandelsprozess eingebunden werden.

Abbildung 28 zeigt einen Screenshot vom User Interface des zweiten Prototyps. Zum einen wird in einer Grafik der prognostizierte Energieverbrauch bzw. -produktion sowie die tatsächlichen Messwerte dargestellt. Die Differenz zwischen diesen Werten wird in einem farblich gekennzeichneten Bereich angezeigt. Zum anderen konnte manuell eine Anpassung der Prognose vorgenommen werden, sodass Prognoseanpassungen aufgrund von bspw. Urlaubsreisen o.ä. möglich sind. Das Auslösen der Transaktionen kann nach entsprechenden Änderungen auch manuell durch den Button “Save to Blockchain” erfolgen. Die Funktionalität des Kaufens wurde im Zuge der Einbettung in die gemeinsame Applikation prototypisch umgesetzt. Außerdem werden weitere Informationen in quantitativer Form dargestellt. Zu diesen gehören in der Energiekonsumenten-Ansicht z. B. der voraussichtliche Energieverbrauch an einem gesamten Tag, die voraussichtlichen Kosten für diesen sowie auch die voraussichtlich zu zahlenden Netzkosten.

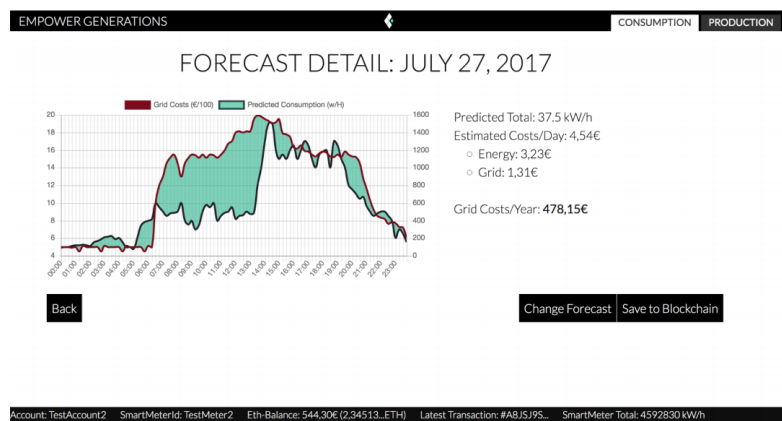


Abbildung 28: Screenshot 2. Prototyp [Eigene Quelle]

logger Das Plugin logger ermöglicht entsprechende Logs auf der Konsole. Diese sind besonders für Entwicklungsarbeiten von großem Vorteil, da somit bestimmt werden kann, welche Plugins und Funktionen problemlos durchlaufen wurden und bei welchen es zu möglichen Problemen kommt. So ermöglichte dieses Plugin neben einer Workflow-Kontrollmöglichkeit auch ein relativ unkompliziertes Bugfixing.

controller-buyer, controller-seller, controller-demo Die auch als Main-Controller bezeichneten Plugins controller-buyer, controller-seller und controller-demo stellen die Kern-

funktionalität der Applikation dar. Diese koordinieren und verwendeten die zuvor beschriebenen Hilfs-Plugins, um den prognosebasierten Energiehandel durchführen zu können.

Die Workflows zum Veröffentlichen von Prognosen, dem Suchen nach passenden Energiepaketen und schließlich auch dem Auslösen eines Kaufes werden in diesem Plugin angetriggert und koordiniert. Ein Energiehandel kann zur Demonstration der Funktionalitäten beschleunigt anhand von beispielhaften Input-Daten illustriert werden. Nachdem die Prognosen ermittelt bzw. aus den Beispieldatensätzen ausgelesen wurden, können die prognostizierten Energiepakete sowie die Netzkosten veröffentlicht d.h. durch Transaktionen auf die Blockchain geschrieben werden. Die wesentlichen Teile des entsprechenden Programmcodes werden in den Funktionen `writePrediction` und `writePrices` ausgelöst:

Listing 3: Ausschnitt Quellcode Applikation

```

async function writePrediction(prediction: number[], day: number, price:
  number, gridZone: number, meterId: string) {
  for (const [index, value] of prediction.entries()) {
    logger.info(##### Writing prediction ${index + 1} of ${
      prediction.length} to Blockchain #####);
    const pkg = new PredictedProduction(day, index, Number(value),
      meterId, price, gridZone);
    const res = await contract.setSellerPackage(pkg);
  }
}

async function writeGridPrices(prices: number[], day: number) {
  for (const [index, price] of prices.entries()) {
    logger.info(##### Writing gridPrice ${index + 1} of ${prices.
      length} to Blockchain #####);
    const res2 = await contract.setGridPrice(day, index, price);
  }
}

```

Nachdem die Prognosen des Energiekonsumenten, Energieproduzenten und des Netzbetreibers vorhanden sind, kann der Energiehandel durchgeführt werden. Dazu werden die angebotenen Energiepakete sowie die Netzkosten (in einem mapping) gesammelt und nach verschiedenen Kriterien sortiert. Die hier verwendeten Kriterien waren die Distanz zwischen den Netzregionen sowie der Preis.

Daraufhin können je nach konfigurierten Constraints die entsprechenden Energiepakete gekauft werden. Die wesentlichen Methodenköpfe des entsprechenden Programmcodes werden im Folgenden dargestellt:

Listing 4: Ausschnitt Quellcode Applikation

```
async function retrievePredictions()
async function retrieveSellers(day: number, pkg: number)
async function retrieveOffers(sellers: ISeller [], day: number, pkg: number)
async function retrieveGridCosts(offers: Offer [])
async function sortByDistance(offers: Offer [])
async function sortByPrice(offers: Offer [])
async function buy(offers: Offer [], buyerMeterId, amount)
async function verify(offers: Offer [], day: number, pkg: number)
```

Falls tatsächliche Messwerte (IST-Werte) bereitgestellt werden sollten, könnte nach dem Energiehandel ein SOLL-IST-Abgleich vorgenommen werden, sodass die Abrechnung erfolgen kann.

Grundsätzlich war es durch die Architektur mit den einzelnen Funktionsbereichen aufgrund der eigens entwickelten Anwendung möglich, vielfältige und individuelle Anwendungsfälle abzubilden. Die Funktionalitäten verdeutlichen, dass der wesentliche Rechenaufwand sowie die Kernfunktionalität nicht direkt auf der Blockchain, sondern auf der lokalen Applikation (DApp) stattfindet. Die Blockchain und der Smart Contract dienen lediglich als gemeinsame Schnittstelle für den Energiehandel, können diesen jedoch als reaktives System nicht eigenständig durchführen. Dies unterscheidet die Blockchain grundsätzlich auch von einem zentralen Rechner.

4.4.2. Testumgebung

Die Planung des zweiten Prototypen umfasste das Einrichten einer Test- und Entwicklungsumgebung um einen effizienten Workflow zu erzielen. Die bisherige Testumgebung basierte auf lokal emulierter Software. Nachdem der erste Prototyp fertiggestellt wurde, genügte eine lokale Umgebung nicht mehr. Die Untersuchung des Verhaltens der Software auf einem produktiv-ähnlichem System ist ein unumgänglicher Bestandteil einer fundierten Bewertung der Technologie. Ebenso konnte während der Erstellung der Testumgebung Erfahrung mit dem Umgang von verschiedenen Ethereum Clienten gesammelt werden. Die eingesetzten Komponenten sind so gewählt worden, dass ein praxisnahes Testen gewährleistet wird.

Die Test- und Entwicklungsumgebung ist beim Praxispartner The Peak Lab vor Ort installiert worden. Da das Unternehmen nach dieser Projektgruppe eine weitere Projektgruppe *Empower Generations Part II* zusammen mit der Universität betreut, kann die Testumgebung von diesen weiter genutzt werden.

Die Umgebung besteht aus fünf Raspberry Pi's. In Kapitel 4.4.1 wurde bereits erwähnt,

dass diese auch in der Praxis eingesetzt werden. Daher wurden die Pi's so konfiguriert, dass diese der Praxis entsprechen. Als Betriebssystem wurde Raspbian und der Ethereum Client Parity genutzt. Es wurden ebenso weitere Ethereum Clients getestet, jedoch gab es mit den meisten Komplikationen und Parity erwies sich als die stabilste Lösung, um eine private Blockchain bereitzustellen.

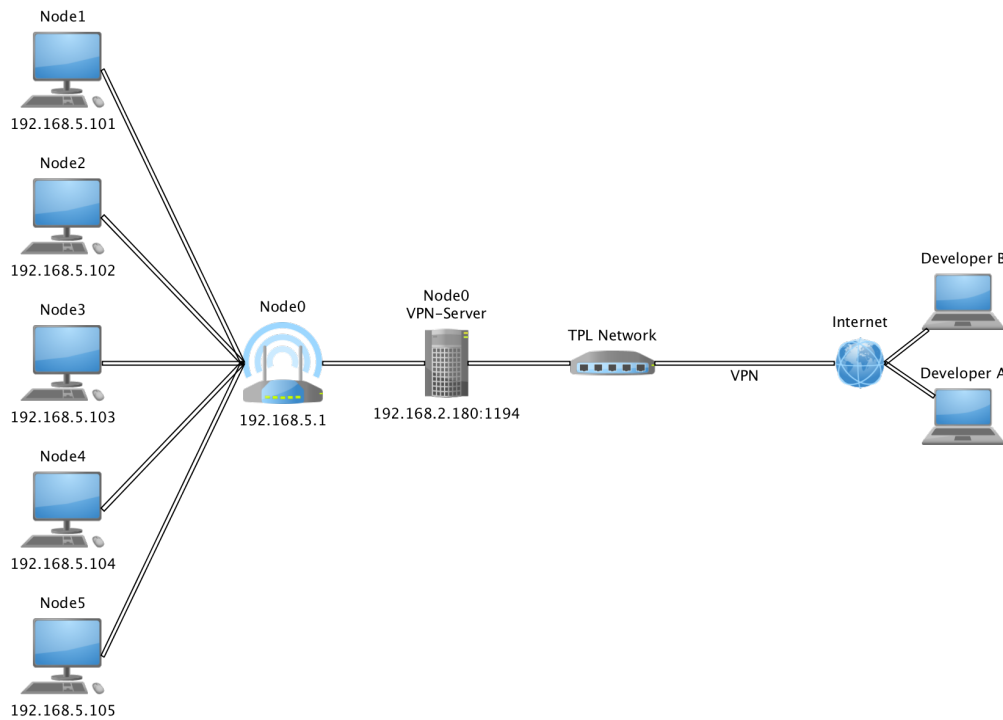


Abbildung 29: Schematische Darstellung der Testumgebung [Eigene Quelle]

Durch die Verwendung einer privaten Blockchain wurde die Grundlage für die Entwicklungsumgebung geschaffen, da hier im Vergleich zu einer öffentlichen Blockchain keine realen Kosten für Transaktionen entstehen. Weiterhin wurde Git auf den Clients installiert, um die zu entwickelnde dezentrale Applikation vom Git Server zu laden. Ein weiterer Entscheidungsgrund für die Pi's war der Kostenfaktor. Mit geringem Budget konnte ein Cluster aus fünf Clients aufgesetzt werden. Mittels W-Lan werden die jeweiligen Clients (in diesem Kontext auch Nodes genannt) mit einem Access Point verbunden. Als Access Point wurde ein weiterer Raspberry Pi verwendet, da für den externen Zugriff ein VPN Zugriff eingerichtet wurde. Dieser VPN Zugriff wurde mithilfe der Software Open VPN realisiert. Der Access Point mit installiertem VPN Server ist per Ethernet mit dem peak lab Netzwerk verbunden. Durch eine Portfreigabe im peak lab Netzwerk und weiteren Firewall

Einstellungen, ist ein Zugriff von außerhalb möglich. Das Team energy of VALUE musste hierzu lediglich einen VPN Clienten installieren und die vom VPN Server generierten Zertifikate installieren.



Abbildung 30: RaspberryPi's der Testumgebung [Eigene Quelle]

Nachdem die technologische Infrastruktur geschaffen wurde, galt es einen Entwicklungsworkflow zu definieren. Die Teilnehmer arbeiten lokal an der dezentralen Applikation. In der Applikation wird die Blockchain Adresse der Testumgebung hinterlegt. So kann die dezentrale Applikation vorerst lokal getestet werden, nutzt aber die Blockchain bzw. den Smart Contract aus der Testumgebung. Dadurch ist es nicht notwendig, dass jeder Nutzer lokal eine Blockchain installieren und der Smart Contract erneut deployed werden muss.

Im Falle der Smart Contract Entwicklung kann lokal der Smart Contract implementiert und auf die Blockchain der Testumgebung deployed werden. Die Adresse des Smart Contracts kann dann an andere Entwickler weitergegeben werden, um ein gemeinsames Testen bzw. Entwickeln zu ermöglichen. Sollte die dezentrale Applikation auf der Entwicklungsumgebung getestet werden, so muss der aktuelle Entwicklungsstand mittels Git an die Raspberry Pis verteilt werden. Dies kann durch eine SSH-Verbindung auf jedem Pi manuell oder durch ein hinterlegtes Script auf dem Pi erfolgen. Dieses Script ist je nach Projektstruktur jedoch individuell zu erstellen. Auf den Clienten wurden außerdem Smart Meter Emulatoren hinterlegt. Dadurch verhält sich jeder Raspberry Pi wie ein Haushalt mit Smart Meter. Dadurch kann die dezentrale Applikation auf emulierte Smart Meter

Daten zugreifen und das interagieren mit anderen Clienten untersucht werden.

Um das Cluster administrativ instand zu halten, wurde die Monitoring Software Ganglia zur Überwachung wichtiger Systemparameter wie CPU-, RAM- und Netzwerkauslastung eingesetzt.

Abbildung 31 zeigt einen Screenshot aus dem eingerichteten Monitoringtool zur Überwachung der Test- und Entwicklungsumgebung. Die Überwachung der Systemparameter dient dazu, die Aktivität und Auslastung der Raspberry Pi's beim Ausführen der dezentralen Applikation bei aktivem Blockchain-Clienten zu überwachen.

Die Erstellung der Testumgebung war für die Projektteilnehmer ein lehrreicher Vorgang im Umgang mit Blockchain bzw. Ethereum Software. In diesem Schritt war das frühe Stadium von Ethereum spürbar, da viele Funktionalitäten Fehler aufwiesen. Ebenso konnten Entwicklungskomplikationen, welche in der ersten Entwicklungsphase auftraten, vermieden werden.

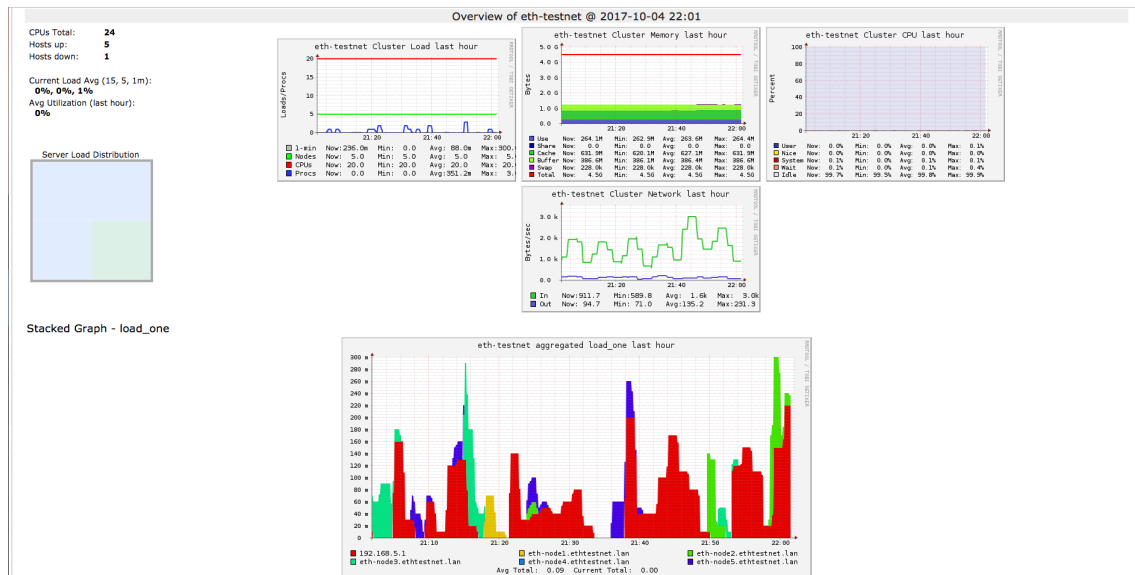


Abbildung 31: Screenshot der Monitoring Software Ganglia [Eigene Quelle]

4.4.3. Smart Contract

Der in der Blockchain gespeicherte Smart Contract muss in der von Ethereum eigens entwickelten Sprache Solidity implementiert werden. Er umfasst alle für die prototypische Anwendung notwendigen Funktionen um die Daten, welche von allen Teilnehmern des Marktes benötigt werden, zu verwalten (Abbildung 32). Der Smart Contract dient zusätzlich auch als Treuhänder – sämtliche Zahlungen werden zunächst hier hinterlegt und erst

im Zuge der Abrechnung an die entsprechenden Parteien freigegeben.

Das untenstehende Listing zeigt die Methode zum Abspeichern eines prognostizierten Angebotes. Die dargestellte Funktion entspricht dem Erstellen eines Inserates auf der Blockchain. So werden dem Smart Contract Daten zur eindeutigen Identifizierung des jeweiligen Verkäufers (`_ sellerSmartMeterId`), der prognostizierten Menge (`_ predictedAmount`) und des Preises pro Watt/h (`_ price`) übermittelt. Zusätzlich wird der 15 minütige Zeitraum, in dem die angegebene Menge produziert werden soll über die Kombination `_ day/_ packageNumber` definiert und im dreidimensionalen Array `SellerPackages[]` hinterlegt. Entsprechende Funktionen sind auch für das Hinterlegen der Netzkosten, das Handeln und das Abrufen der genannten Daten implementiert. Ferner wurden Fallback-Mechanismen für den Fall der Abweichung von Prognose und Ist-Wert implementiert.

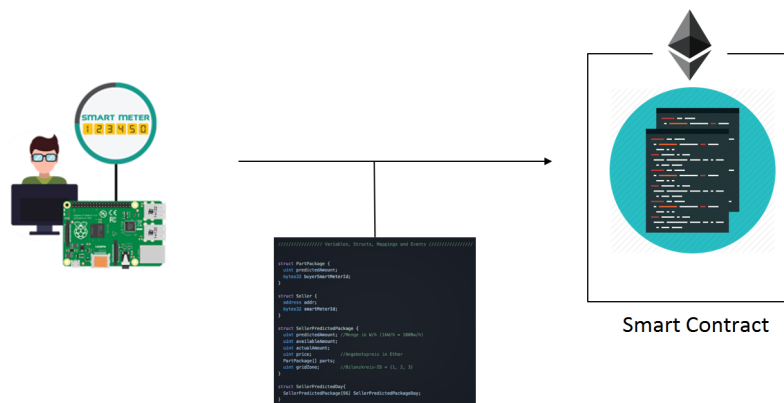


Abbildung 32: Smart Meter Daten im Smart Contract [Eigene Quelle]

Listing 5: Solidity Smart Contract Ausschnitt

```
// Is executed 96 times for one day, provides prediction
function setSellerPredicted(bytes32 _sellerSmartMeterId, uint _day, uint
    _packageNumber, uint _predictedAmount, uint _price) { Der Wandel zu einem
    dezentralen Energiemarkt
    SellerPackages[_sellerSmartMeterId][_day].SellerPackageDay[_packageNumber].
        predictedAmount = _predictedAmount;
    SellerPackages[_sellerSmartMeterId][_day].SellerPackageDay[_packageNumber].
        price = _price;
    SellerPackages[_sellerSmartMeterId][_day].SellerPackageDay[_packageNumber].
        availableAmount = _predictedAmount;
    SetSellerPredicted(_sellerSmartMeterId, _day, _packageNumber,
        _predictedAmount, _price, _gridZone); }
```



```
event SetSellerPredicted(  
bytes32 indexed _smartMeterId,  
uint indexed _day,  
uint indexed _packageNumber,  
uint _predictedAmount,  
uint _price,  
uint _gridZone);
```

Eine bedeutende Einschränkung bei der Implementierung mittels Solidity ist, dass ausschließlich primitive Datentypen beim Aufruf von Funktionen im Smart Contract verwendet werden können. So muss ein Anbieter zum Abspeichern der Prognosen für einen Tag im dargestellten Fall 96 rechen-, zeit- und kostenintensive Transaktionen durchführen (um 96 Integer-Werte abzulegen).

Gleiches gilt für die Käufer-Seite: Es ist nicht möglich, eine Liste oder ein Array aller Anbieter zu einem definierten Zeitraum (Tag/Paketnummer) abzufragen. Auf Clientseite muss hierzu erst über alle Transaktionen der Blockchain iteriert werden. In Solidity gibt es demnach sog. Events, die aufgerufen werden können, sobald eine bestimmte Funktion in einem Smart Contract ausgeführt wird. Auch lässt sich mit einer gefilterten Abfrage aller Events zur Funktion `_setSellerPredicted` die Ergebnismenge einschränken.

4.4.4. Bewertung der Blockchain als Energiehandelstechnologie

Abschließend soll die Blockchain als Basis einer Energiehandelstechnologie bewertet werden. Die Bewertung basiert auf den erlangten Erkenntnissen während der Implementierung des Energiehandels über die Blockchain. Für eine fundierte Bewertung anhand des erstellten Prototypen ist eine Untersuchung in echten Haushalten notwendig. Da der Fokus auf der prototypischen Entwicklung des Prototyps lag und bis zum Ende der Projektgruppe ständiger Verbesserungen unterzogen war, konnte im Zeitraum des Projekts keine umfangreiche Evaluation der Ergebnisse durchgeführt werden.

Der Prototyp hat jedoch gezeigt, dass dezentraler Energiehandel mithilfe einer Blockchain-Implementierung realisiert werden konnte. Ebenso konnte die marktseitige Komponente durch den Energiehandel zwischen Energieproduzenten und Energiekonsumenten abgebildet und die netzseitige Komponente durch Einbindung der Prognosen der Netzauslastung berücksichtigt werden. Während der Implementierung wurden jedoch die aktuellen Grenzen der Technologie deutlich. Die Programmentwicklung stellte sich zu Beginn des Projekts als äußerst schwierig heraus, da zu diesem Zeitpunkt kaum Best Practices vorlagen, Dokumentationen (z. B. Truffle-Dokumentation) nicht gepflegt wurden oder teilweise nicht vorhanden waren.

Da die Ethereum Blockchain durch eine Javascript API zugänglich ist, wurden Best Practices aus den Bereichen von Node.js und Angular angewendet. Die in Kapitel 4.4.2 erstellte Testumgebung zeigte, dass die Ethereum Clienten untereinander Differenzen hinsichtlich der Funktionalitäten aufwiesen. Ebenso wurde durch die Anzahl der Commits im jeweiligen Git Repository deutlich, dass die Clienten stetiger Entwicklung unterliegen.



Abbildung 33: Screenshot Blockchain Funktion in gemeinsamer Applikation [Eigene Quelle]

Es wurde die Erfahrung gemacht, dass verschiedene Versionen der Clienten Inkompatibilität aufwiesen, sodass ständige Anpassungen notwendig waren. Da entsprechende Dokumentation nicht direkt gepflegt und aktualisiert wurden, mussten die Anpassungen stets mit großem Aufwand nachimplementiert werden. Weiterhin wurde in Kapitel 4.4.4 verdeutlicht, dass Smart Contracts primitive Datentypen unterstützen. Somit sind große Datenmengen, die im Smart Contract verarbeitet wurden, unpraktikabel und aufwändig (bzgl. Transaktionskosten). Der Workaround, der in der Implementierung getroffen wurde, zeichnete sich durch die Aufteilung der Prognosen aus. Daraus folgert eine hohe Transaktionszahl, wel-

che mit Kosten in der Blockchain verbunden sind. Ebenso ist die Skalierbarkeit bei vielen Transaktionen mit Proof of Work zu untersuchen. Ein weiteres Indiz, welches die Blockchain als geeignete Technologie in Frage stellt, ist die Rechenleistung der Blockchain bzw. das ausführen komplexer Algorithmen in Smart Contracts. Die in Kapitel 4.4.1 dargestellte Architektur wurde bewusst so gewählt, dass die Operation des Kaufens auf die dezentrale Applikation ausgelagert wurde.

All diese Faktoren, die gegen die Blockchain als einzusetzende Technologie sprechen sind jedoch vor allem technischer Natur, die durch weitere Entwicklungen behoben werden können. Wie bereits erwähnt, arbeitet Ethereum bereits an der Proof of Stake Implementierung. Diese verspricht kürzere Transaktionszeiten.

Ein deutliches Potential konnte durch die Einarbeitung in eine gemeinsame Applikation identifiziert werden (siehe Abbildung 33). Der Prototyp wurde in eine mobile Web-Applikation eingebettet und Schnittstellen zu anderen Teilgruppen hergestellt. Synergieeffekte zwischen Predictive Analytics sowie Machine Learning konnten prototypisch erarbeitet werden. Die Blockchain ist ein enabler für die Entwicklung von Smart Devices. Geräte können so in der Lage sein durch definierte Parameter im Smart Contract eigenständig zu handeln. Im Kontext des Energiemarktes können sich Geräte beispielsweise selbst einschalten, wenn der Energiepreis zu einem bestimmten Zeitpunkt einen gewissen Wert unterschreitet.

Die Blockchain ist also vor allem eine unterstützende Technologie die auf unterster Schicht einer Architektur arbeitet. Die Daten- und Transaktionsspeicherung sowie das Auslösen von Prozessen über den Smart Contract zählen zu den tragenden Funktionalitäten der Blockchain.

4.5. Schlussbetrachtung

Zusammenfassung Die Teilgruppe energy of Value hat während der Projektzeit ein Proof of Concept eines Energiehandels unter Verwendung der Blockchain-Technologie erarbeitet. Während der Projektzeit wurden zwei wesentliche Ergebnisse erarbeitet. Zum einen wurde ein prognosebasiertes Konzept entwickelt, das die Anforderungen eines dezentralen Energiemarktes erfüllt. Zweitens wurde das Konzept mithilfe der Blockchain-Technologie durch prototypischer Implementierung umgesetzt.

Die Ausgangslage der Problemstellung offenbarte, dass die Netzkomponente in bisherigen Projekten kaum berücksichtigt wurde. Eben zu diesem Faktor muss die Blockchain-Technologie eine Lösung bieten, um die Problematiken der Energiewende zu bewältigen (Kapitel 2). Eine wichtige Erkenntnis, während der Konzeptionierung war die Tatsache,

dass ein Energiehandel ohne Intermediär nicht möglich sein wird. Damit ist ein reiner P2P Handel nur bedingt realisierbar. Der Netzbetreiber stellt stets die Netz-Infrastruktur bereit und ist demnach stets eine zentrale Instanz zwischen Energiekonsumenten und Energieverbrauchern. Ebenso ist der Versicherer als Instanz notwendig, um Ungleichgewichten entgegenzuwirken und bei Prognoseabweichungen aufzukommen.

Die Ausgangsfragestellung wurde daher von der reinen Realisierung eines P2P Handels zur Realisierung eines netzdienlichen, dezentralen Energiehandels umformuliert. Das Konzept basiert auf prognostizierten Energiemengen, da diese bereits im bisherigen Energiesystem Verwendung finden. Durch die technischen Möglichkeiten der digitalen Weiterverarbeitung mithilfe von Smart Metern wurde der marktübliche Ansatz weiterentwickelt.

Fazit Durch die anschließende Konzeption und Anwendungsentwicklung wurden sukzessiv wertvolle Erkenntnisse gewonnen, die zum vorgestellten Energiehandelssystem führten. Die grundsätzliche Machbarkeit auf IKT-Ebene wurde dabei gezeigt. Dezentraler Energiehandel kann mithilfe einer Blockchain-Implementierung realisiert werden. Ebenso konnte die marktseitige Komponente durch den Energiehandel zwischen Energieproduzenten und Energiekonsumenten abgebildet werden und die netzseitige Komponente durch Einbindung der prognostizierten Netzauslastung berücksichtigt werden.

Jedoch wurde auch festgestellt, dass ausschließlich auf Blockchain basierende Ansätze aus Performanzgründen in dieser Form zum jetzigen Zeitpunkt nicht in der Breite eingesetzt werden können. Dies liegt insbesondere an der Ineffizienz der Verifikationsprozesse (Proof of Work) und der Sicherheitsmechanismen der Blockchain-Datenstruktur selbst, bei der Daten, seien dies nun Prognosen oder monetäre Transaktionen, lediglich immer nur an das Ende angehängt werden können (append only).

Auch zeigte sich, dass die Möglichkeiten zur Datenverarbeitung stark limitiert bleiben. Somit zeigt sich, dass es sehr unwahrscheinlich ist, dass Anwendungen, die ausschließlich auf Blockchain zur Datenhaltung und -verarbeitung basieren, den dezentralen Energiemarkt beflügeln werden. Vielmehr wird eine weitere Verarbeitungsschicht implementiert werden müssen, die die individuellen Prognosen miteinander verknüpft, also die Rolle eines Brokers einnimmt. Als Alternative kommen natürlich auch Weiterentwicklungen im Bereich der Blockchain-Technologie infrage, bei der die Datenverarbeitungsgeschwindigkeit und die Verifikationsmechanismen deutlich effizienter gestaltet werden müssen.

Das Marktmodell basiert auf kurzfristigen Verbrauchs- und Konsumprognosen und ist bewusst an den bestehenden Energiemarkt angelehnt, da beide Energiesysteme zumindest mittelfristig parallel existieren müssen. Daher ist der Erfolg dieses Konzepts von Blockchain-Technologien und auch von der Güte der Prognosen abhängig. Der dezentrale

Energiehandel baut auf bestehenden Strukturen auf und einer Koexistenz mit dem derzeit praktizierten State of the Art steht grundsätzlich nichts im Wege. Aus diesem Modell erschließt sich jedoch der Wandel des klassischen Intermediäres vom Energieversorger zum Versicherer. Konsumenten, Produzenten und Prosumer nehmen aktiv am Energiehandel teil, ohne den klassischen Energieversorger damit zu beauftragen. Angebot und Nachfrage werden stark liberalisiert und deutlich transparenter als im aktuellen Energiesystem.

Ausblick Um das tatsächliche Potential vollends beurteilen zu können, muss die Blockchain-Technologie einen ausreichenden Reifegrad erlangen. Des Weiteren müssen wichtige Kriterien wie die Skalierbarkeit sowie weiterführende Sicherheitsaspekte hinsichtlich der Software als auch der Hardware erforscht werden. Hierbei sind u.a. Maßnahmen festzulegen, wie bei einem Hardwareausfall oder bei Verlust des Blockchain-Nutzerkontos (privater Schlüssel, öffentlicher Schlüssel) verfahren werden soll. Darüber hinaus sind viele (datenschutz-) rechtliche Fragen offen, die insbesondere durch die ab 2018 geltende EU-Datenschutzgrundverordnung nochmals verschärft werden. Die geltenden Rechtsvorschriften sind auf den heutigen Energiemarkt ausgelegt und erlauben derzeit keine Umsetzung des Konzepts in der Praxis. Dennoch ist eine Evaluation des Prototyps in realen Haushalten ein weiterer notwendiger Faktor, der untersucht werden muss.

Dezentraler Energiehandel über die Blockchain führt zu verschiedenen Vorteilen, so entstehen Steuerungseffekte durch die Berücksichtigung der Netzkosten während des Energiehandels. Hierbei kann das Konsumverhalten durch Preismechanismen beeinflusst werden. Das vorgestellte Marktdesign kann dazu beitragen, dass sich Konsumenten verstärkt netzdienlich verhalten. Die beschriebene Systemarchitektur ermöglicht zudem einen hohen Automatisierungsgrad, da der Energiehandel durch die individuell konfigurierbare Applikation abgewickelt wird, ohne dass tiefgreifende Eingriffe des Anwenders erforderlich werden. Weiterhin hat die Verwendung eines Blockchain-Ansatzes den positiven Effekt, dass Transaktionen prinzipbedingt manipulationssicher dokumentiert werden und gleichzeitig jederzeit transparent und nachprüfbar bleiben.

Das Kapitel der Implementierung zeigte, dass das erstellte Konzept auf die Blockchain abgebildet werden kann, jedoch noch deutliche technische Grenzen aufweist. Lösungsansätze, wie zum Beispiel die Einführung von Proof of Stake, sind jedoch bereits in der Entwicklung. Die einzige Grenze, die vermutlich bestehen bleiben wird, ist die Rechenleistung der Blockchain. Komplexe Operationen, wie zum Beispiel die implementierte Kauf-Funktion, werden auch langfristig nicht vollständig durch Smart Contracts implementiert werden können. Die Bedenken hinsichtlich der Performance werden durch die Einführung von Proof of Stake gemindert werden. Das Potential ist auf jeden Fall gegeben, dass die Blockchain den

bisherigen Energiemarkt beflügeln wird. Auch abseits des Energiemarkts ergeben sich viele Bereiche, in denen die Blockchain als disruptiv beschrieben wird. Viele große Unternehmen wie SAP, Microsoft oder IBM befassen sich aktuell mit diesem Thema. Vor allem in Kombination mit Machine Learning und Internet of Things wird die Blockchain ein wichtiger technischer Bestandteil werden.

5. Energy of WOW

Dieses Kapitel dokumentiert das Vorgehen und die Ergebnisse der Teilgruppe energy of WOW innerhalb der Projektgruppe Empower Generations. Thematisch handelt dieser Abschnitt von der Technologie der Lastkurvenanalyse, welche die Disaggregation von Lastkurven privater Haushalte vorsieht. Das Ziel ist die Erkennung von spezifischen Verbrauchsprofilen einzelner elektrischer Geräte im Haushalt, um damit bestimmen zu können, welche Geräte zu welchem Zeitpunkt angeschaltet sind und wie viel Strom diese verbrauchen. Mit Hilfe dieser Technologie ist es beispielsweise möglich dem Energieverbraucher seinen Gesamtverbrauch detailliert aufzuschlüsseln und so Verbräuche auf einzelne Geräte zurückzuführen. Diese Technologie kann in den verschiedensten privaten und gewerblichen Bereichen eingesetzt werden.

Nach einer kurzen Einleitung und einem Abschnitt zu den Grundlagen folgen Ausführungen zu Konzeption und Implementierung eines umgesetzten Prototyps. Dieser Prototyp spiegelt einen Ansatz für einen Gesamtprozess für die Nutzung von Disaggregation wider. Ein Anwendungsfall umfasst die Aufnahme der Daten über einen Smart Meter, die Verarbeitung der Daten auf einem Rechner und der Visualisierung der Informationen mit Hilfe eines Endgerätes. Ziel ist die Abbildung einer prototypischen Daten- und Informationsverarbeitung von der Messung der Daten bis hin zur Darstellung der Ergebnisse.

Bei der Visualisierung der Ergebnisse setzt auch der zweite Schwerpunkt der Teilgruppe ein. Im Rahmen des Projekts wurde eine mobile Anwendung zur Visualisierung von Verbrauchsdaten erstellt, diese bildet die Grundlage für die durchgeführte Forschungsarbeit. Die Forschungsarbeit fokussiert sich auf verschiedene Fragestellungen zum Thema User Experience beim privaten Energiemonitoring. Mit Hilfe von acht Interviews mit sogenannten Prosumern soll evaluiert werden, welche Informationen von dem entwickelten Prototyp tatsächlich für den Konsumenten nutzbringend sein können.

5.1. Einleitung

Im folgenden Abschnitt wird eine kurze Einleitung zur Thematik gegeben. Dazu wird eine kurze Motivation und die resultierende Problemstellung beschrieben. Diese ist die Grundlage für die Aufgabenstellung und die anschließende Vorgehensweise.

5.1.1. Motivation

Der Energiemarkt durchlebt derzeit einen starken Wandel. Erneuerbare Energien aus Windkraftanlagen, Photovoltaikanlagen und Biomasse lösen die konventionellen Energieträger

ab und treiben die Energiewende voran. In Deutschland machten die erneuerbaren Energien 31,6 % der gesamten Bruttostromerzeugung aus und bis zum Jahr 2025 soll dieser Anteil auf 40 - 45 % ansteigen [Zen16]. Eine Folge dieser Veränderungen ist die Dezentralisierung des gesamten Energienetzes: Neben den wenigen konventionellen Kraftwerken, die von den großen Stromerzeugern gesteuert werden, entstehen nun unzählige kleine Erzeuger. Durch diese neuen Umstände entstehen eine Vielzahl von neuen Anforderungen an das vorhandene Stromnetz, da jeder Konsument auch als potentieller Produzent wirken und kaum gesteuert werden kann [Bunc].

Um diesen neuen Anforderungen gerecht zu werden, sind intelligente Stromnetze, also die Smart Grids und verschiedene andere moderne Informationstechnologien notwendig. Ein wichtiger Baustein, um diese Digitalisierung voranzutreiben, sind die Smart Meter, die in jedem Haushalt den analogen Stromzähler ersetzen sollen. Mit diesem intelligenten Zähler entsteht in jedem Haushalt das Potential die Energieerzeugung, sowie den Energieverbrauch transparenter zu machen und diese miteinander zu verknüpfen. Mit dem Gesetz zur Digitalisierung der Energiewende hat die Bundesregierung den Einbau der intelligenten Messsysteme beschlossen. In diesem Gesetz sind ab 2017 erste Unternehmen bzw. größere Erzeuger und ab 2020 auch Privathaushalte eingeschlossen [Bun16c].

Etablierte Unternehmen in der Energiebranche, sind aufgrund der Veränderungen gefordert ihre traditionellen Geschäftsmodelle zu überdenken und den Fokus auf innovative und disruptive Technologien zu legen. Der Energieversorgungskonzern *RWE* beispielsweise bündelte im April 2016 alle Geschäfte im Bereich der erneuerbaren Energien in der Tochtergesellschaft *innogy SE* [RWE16]. Mit der Plattform *Innovation Hub* unterstützt *innogy SE* verschiedene Start-Ups, um neue digitale Geschäftsmodelle und Lösungen zu entwickeln [inn].

Im Jahr 2015 investierte *innogy SE* mit weiteren Finanzierungspartnern insgesamt 16.6 Millionen Dollar in die Firma *Bidgely Inc.* aus dem Silicon Valley [Cru15]. Dieses Unternehmen hat sich die Aufgabe gemacht cloud-basierte Lösungen zu entwickeln, die dabei helfen den Energieverbrauch eines Haushalts besser zu verwalten. Die Technologie, die sich dahinter verbirgt, trägt den Namen Disaggregation. Dabei werden Lastkurven eines Haushalts analysiert, um festzustellen welche Verbraucher im Haushalt zu einer bestimmten Zeit aktiv waren und wie viel Strom individuell verbraucht wurde [Bid16]. Die Daten für diese Lastkurven werden aus einem intelligenten Stromzähler gelesen. Die Disaggregation nutzt somit Möglichkeiten, die im Zuge der Digitalisierung der Energiewende eingeführt werden, um einen Mehrwert für verschiedene Akteure zu erzeugen.

Für den Netzbetreiber sind präzise Verbrauchsprofile wichtig, um bessere Planungen für das gesamte Energienetz durchführen zu können. Denn wie anfangs erwähnt kommt

dieser Aufgabe bei vielen dezentralen Erzeugern eine besondere Bedeutung zu. Für eine Privatperson hingegen entsteht durch die Disaggregation ein hoher Grad an Transparenz in Bezug auf sein Verbrauchsverhalten. Gegenüber intelligenten Haushaltsgeräten oder Plug-Systemen, die vergleichbare Informationen liefern, hat die Disaggregation kaum Anschaffungs- oder Betriebskosten und ist gut skalierbar [K. 12]. Dieses Verfahren wird in der Literatur auch als “Nonintrusive load monitoring“ bezeichnet, da die Systeme neben dem Smart Meter kaum bemerkbar sind [Rom12]. Ein Kritikpunkt der im Zuge dieses Verfahrens, auch besonders in Deutschland, immer wieder auftritt ist der Datenschutz. Denn es handelt sich, um eine Analyse von personenbezogenen Daten, die Rückschlüsse über Aufenthalt, Lebensweise und anderes Wissen ermöglicht [Bun15] [Buna].

Das gesammelte Wissen aus den Ergebnissen der Disaggregation kann in den verschiedensten Bereichen eingesetzt werden. Durch gezieltes Feedback kann der Hausbewohner sein Verbrauchsverhalten optimieren und so Strom sparen. Unternehmen dagegen könnten die Daten verwenden, um dem Kunden personalisierte Dienste anzubieten, wie beispielsweise Empfehlungen für den Kauf effizienterer Geräte. Die Möglichkeiten die sich aus den gesammelten Daten ergeben sind vielfältig und in den unterschiedlichsten Bereichen einsetzbar [K. 12]. Im Rahmen der Projektgruppe soll ein Konzept entwickelt und implementiert werden, wie der Datenstrom von den aufgezeichneten Daten eines intelligenten Stromzählers bis zur Visualisierung der Ergebnisse beim Endkunden aussehen kann. In einem weiteren Schritt gilt es dann den Mehrwert der Daten, die mit Hilfe der Disaggregation der Lastkurve ermittelt werden konnten, für weitere Akteure herauszustellen, um so auch mögliche Geschäftsfelder zu identifizieren.

Die Grundproblematik dieser Forschungsarbeit zeichnet sich durch mehrere Teilprobleme aus, die im Laufe dieses Abschnitts näher beleuchtet werden. Grundlage dieser Annahmen, dass diese Teilprobleme als solche identifiziert wurden, sind Literatur aus Praxis und Forschung sowie Impulse der Partner und Experten.

5.1.2. Problemstellung

In den meisten deutschen Privathäusern finden sich analoge Stromzähler, die als Grundlage für die Grundproblematik fungiert. Denn beim Einsatz dieser analogen Stromzähler ist der Stromverbrauch sowohl für den Stromanbieter als auch für den Kunden selbst nicht transparent.

Alle elektrischen Verbraucher sind an analoge oder digitale Stromzähler angeschlossen, welche die verbrauchte Energie messen und die verbrauchte Menge in kWh lediglich am Gerät widerspiegeln. Diese Anzeige wird jährlich vom Netzbetreiber abgelesen und bildet

die Grundlage für die Energieabrechnung, wodurch sich herausstellt, ob man nachzahlen muss oder eine Rückerstattung bekommt. Im Verlauf des Jahres muss der Kunde monatlich einen Abschlag an den Stromanbieter bezahlen [Kla10]. Als Beispiel einer Stromrechnung (Abbildung 34) wurde zwecks Praxistauglichkeit eine Rechnung vom Netzbetreiber *EWE* ausgewählt, der im Bereich Nord-West-Deutschlands zuständig ist. Zu sehen sind nur der Energieverbrauch in kWh, schwach aufgeschlüsselte Kosten, dazu die abgeführte Mehrwertsteuer mit Brutto- und Nettowerten, sowie der zu zahlende Stromabschlag für die Folgeperiode. Besonders in der Kritik steht die Intransparenz die diese Rechnung dem Endverbraucher liefert.

Rechnung 02. Oktober 2012

Guten Tag Herr Mustermann,

vielen Dank, dass Sie sich im zurückliegenden Abrechnungszeitraum für die EWE VERTRIEB GmbH als Ihren Partner für Energie und Dienstleistungen entschieden haben. Aus unseren Leistungen ergibt sich folgende Rechnung:

	Menge	netto Euro	MwSt Euro	brutto Euro
Strom	4.790 kWh	1.023,51	194,47	1.217,98
Erdgas	25.920 kWh	1.443,19	274,21	1.717,40
Wasser	65 m³	122,59	8,58	131,17
Abwasser <small>(im Auftrage Vekehr und Wasser GmbH)</small>	65 m³	117,00	0,00	117,00
Rechnungsbetrag		2.706,29	477,26	3.183,55
abzüglich Ihrer bis zum 25.09.2012 geleisteten Abschlagszahlungen		2.715,24	474,76	3.190,00
Ihr Guthaben				6,45

Das Guthaben wird mit Ihrem 1. Abschlagsbetrag verrechnet, so dass bei der 1. Abschlagsfälligkeit nur der verminderte Abschlagsbetrag zu zahlen ist. Bitte überweisen Sie den Betrag auf unser Bankkonto:
Oldenburgische Landesbank AG
Konto 1234567890
Bankleitzahl 28020050

Für den neuen Abrechnungszeitraum haben wir folgenden Abschlagsbetrag ermittelt:

Abschlagsbetrag	netto Euro	MwSt %	MwSt Euro	brutto Euro
Strom	89,92	19,00	17,08	107,00
Erdgas	126,05	19,00	23,95	150,00
Wasser	10,28	7,00	0,72	11,00
Abwasser	10,00	0,00	0,00	10,00
Abschlagsbetrag gesamt				278,00

Der Abschlagsbetrag ist erstmals am 01.10.2012 fällig und danach jeweils zum 1. eines jeden Monats bis einschließlich 01.09.2013.
Bitte überweisen Sie auf unser Bankkonto:
Oldenburgische Landesbank AG
Konto 1234567890
Bankleitzahl 28020050

Abbildung 34: Musterrechnung der *EWE AG* [EWE12]

Diese Rechnung veranschaulicht das Grundproblem: Das Versorgungssystem in einem Haushalt, insbesondere mit einem analogen Stromzähler, bietet keine Möglichkeit, um präzise Verbrauchsdaten zu ermitteln und erzeugt so Intransparenz. Mit der dann vorhandenen Datenqualität lassen sich keine Informationen aus den Daten gewinnen, wie zum Beispiel die Identifikation einzelner Geräte im Gesamtverbrauch.

Aus dieser Grundproblematik lassen sich auch die Problemstellungen für involvierte Akteure ableiten: Für Stromnetzbetreiber oder auch den Stromerzeuger können detaillierte Verbrauchsdaten helfen, präzise Planungen für das Energienetz aufzustellen oder auch die Energieerzeugung besser zu steuern. Wie bereits erwähnt sind etablierte Energieunternehmen aufgrund der Dezentralisierung des Energiemarktes dazu gezwungen neue Geschäftsmodelle zu entwickeln. Viele denkbare Dienstleistungen sind aufgrund der wenigen Daten, die nur asynchron eintreffen, schlichtweg nicht realisierbar.

Auch für den Kunden ergeben sich nicht genutzte Potentiale. Denn mit Hilfe von präzisen Informationen über den eigenen Stromverbrauch entsteht ein Optimierungspotential, zum Beispiel durch das Erkennen von Stromfressern oder das Aufdecken von Fehlverhalten. Mit der jährlichen Abrechnung, wie oben veranschaulicht, weiß der Kunde nicht, in welchem Zeitraum am meisten oder wenigsten Strom verbraucht wurde und schon gar nicht durch welche Geräte.

Aufbauend auf dem Grundproblem bietet die Verteilung von digitalen Stromzählern einen Ansatz zur Problemlösung, da digitale Stromzähler den aggregierten Stromverbrauch in einer deutlich höheren Frequenz aufzeichnen und diese präzisen Daten über eine Schnittstelle anbieten. Das Problem, nicht zugänglicher Verbrauchsdaten, wird damit angegangen, wodurch sich in diesem Zusammenhang eine neue Problemstellung ergibt. Wie sieht ein System aus, das die Daten eines Haushalts auslesen und so verarbeiten kann, um zur Lösung der oben beschriebenen Probleme der einzelnen Akteure beizutragen und wie können die Lastkurven, also die Verbrauchsdaten eines Haushalts, so disaggregiert werden, dass Rückschlüsse über den Stromverbrauch einzelner Geräte ermöglicht werden können? Genauer wird darauf im folgenden Abschnitt im Zuge der Untersuchungsobjekte eingegangen.

Teilbereiche der Untersuchungsobjekte Der erste Teilbereich befasst sich mit den technischen Anforderungen, sowie mit der technischen Umsetzung eines solchen Systems, das in einem Haushalt eine Lastkurvenanalyse durchführen kann. Auf unterster Ebene betrifft dies Entscheidungen zum Thema Einsatz der Hardware und zum Thema Schnittstellen. Für die Nutzung eines Servers beispielsweise müssen Anforderungen an Rechenleistung und ebenfalls Schnittstellen definiert und evaluiert werden. Vor allem auch die Schnittstellen zu den Akteuren haben eine große Bedeutung, da jeder Akteur andere Anforderungen und Ziele hat. Also wie können die Daten optimal für verschiedene Akteure bereitgestellt werden, um den Mehrwert zu maximieren? Ein weiterer wichtiger Punkt ist der Einsatz von Software und Anforderungen an die genutzten Daten. Gibt es Frameworks, Bibliotheken oder fertige Software, die zur Erstellung des Systems genutzt werden können und welchen Funktionsumfang haben diese? Mit welchem Algorithmus kann eine Disaggregation durchgeführt werden und gibt es dafür bestehende Software? Des Weiteren sind auch Anforderungen an die Daten relevant, da Qualität, Menge, Format und weitere Eigenschaften eine große Rolle spielen.

Der Zweite Teilbereich behandelt die Frage, welchen Nutzen verschiedene Akteure aus der Technologie der Lastkurvenanalyse ziehen können, beziehungsweise welche Mehrwerte die Umsetzung für Akteure, wie einen einzelnen Privathaushalt, Netzbetreiber oder Energieversorger und andere dritte Parteien, haben kann. Auch immer im Hinblick auf

die Nutzung der Ergebnisse innerhalb neuer Geschäftsmodelle weiterer externer Parteien. Aufgrund des Umfangs wird es hauptsächlich um die Konsumenten gehen. Dabei soll der Nutzen der bereitgestellten Informationen erarbeitet und überprüft werden.

5.1.3. Aufgabenstellung

Prototyp Obwohl die Technologie der Disaggregation in der Forschung bereits schon länger untersucht wird, gibt es nur wenige Lösungen von Unternehmen die diese Technologien einsetzen. In Zeiten der Energiewende und mit dem Ausrollen der Smart Meter werden die Voraussetzungen geschaffen, um auf dieser Basis neue Geschäftsmodelle zu entwickeln. Ein Ziel dieser Teilgruppe ist es, einen Prototyp für einen Disaggregationsprozess zu entwickeln. Damit soll die prinzipielle Durchführbarkeit für die Anwendung der Disaggregation herausgestellt werden, also ein Proof-of-Concept. Der Prozess begrenzt sich dabei nicht auf den Disaggregations-Algorithmus selbst. Es beginnt bei der Aufzeichnung und Verarbeitung der Verbrauchsdaten durch einen Stromzähler und der Speicherung der aufgezeichneten Daten. Im Anschluss folgt die Durchführung der Disaggregation und schließlich das Bereitstellen der Ergebnisse einem möglichen Kunden. Mit Hilfe eines Prototyps soll dieser Gesamtprozess im Rahmen des Projektes entworfen, implementiert und untersucht werden.

Paper Die Teilprojektgruppe hat es sich zur Aufgabe genommen, den Prototypen zu evaluieren. Es wurden semistrukturierte Interviews, die im Zusammenhang mit der User Experience im Energiemonitoring stehen, durchgeführt. Im ersten Schritt wurde ein Prototyp entwickelt, mit dessen Hilfe Lastkurven disaggregiert werden können. Dadurch ist das Nutzungsverhalten von Endgeräten in Haushalten erkennbar. Daraufhin soll eine mobile Applikation entwickelt werden, die dem User Energiedaten aus der Disaggregation anzeigen kann. Zu diesem Zweck werden Probanden hinsichtlich möglicher Informationen und Interaktionen in einer Applikation befragt. Zum Schluss werden diese Ergebnisse evaluiert, damit eine Empfehlung zu den wichtigsten Informationen und Funktionen gegeben werden kann.

5.1.4. Vorgehensweise

Zunächst wird eine systematische Literaturanalyse dazu genutzt, um den aktuellen Stand der Forschung zum Thema Lastkurvenanalyse herauszustellen. Der Fokus liegt bei der Erläuterung wichtigster Begrifflichkeiten und der Ermittlung des State-of-the-Art. Damit wird der aktuelle Stand der Technologie herausgestellt und ein Überblick über aktuelle Lösungen am Markt verschafft. Des Weiteren wird auch auf die verschiedenen privaten und

gewerblichen Einsatzmöglichkeiten der Disaggregation eingegangen. Zur Verdeutlichung dieser Möglichkeiten wird auch konkret ein regionales Unternehmen aufgezeigt, das an Lösungen mit dieser Technologie arbeitet.

Um die Anforderungen besser analysieren zu können, wird in einem weiteren Schritt ein globales Anwendungsszenario abgeleitet. Dieses wird mit Hilfe eines Use Case Diagramms nach UML-Vorgaben visualisiert. Dieses Szenario dient als Grundlage zur Erstellung einer Grobarchitektur, die alle nötigen Komponenten des Systems zur Umsetzung der Anwendungsfälle umfasst. Das Zusammenspiel dieser Komponenten wird dann mit Hilfe eines Gesamtbildes der Grobarchitektur verdeutlicht. Die einzelnen Komponenten selbst werden darauffolgend, mit dem Fokus auf den jeweiligen Aufgabenbereich, kurz beschrieben. Abschließend werden zu den Komponenten auch konkrete Anforderungen definiert, die als Anhaltspunkte für die folgende Implementation dienen.

Im nächsten Schritt folgte die hard- und softwaretechnische Umsetzung des Prototyps. Die Grundlage bildet ein Komponentendiagramm, welches die Beziehungen und Abhängigkeiten zwischen den verschiedenen Komponenten verdeutlicht. Anschließend werden Funktionalität, Umsetzung und Abhängigkeiten für jede einzelne Komponente erläutert. Dabei liegt ein Fokus auf den Entscheidungen für genutzte Technologien und die Lösung aufgetretener Probleme.

Eine weitere Komponente ist die Visualisierung der Ergebnisse und der Verbrauchsdaten in einer hybriden Applikation. Auf Basis dieses Prototyps wurde eine Forschungsarbeit zum Thema durchgeführt, welche Informationen in einer solchen Applikation für den energie-affinen Endanwender nutzbringend sind. Dafür werden erst die Voraussetzungen und Methodik der semistrukturierten Interviews vorgestellt, die zur Ermittlung der Ergebnisse eingesetzt wurden. Anschließend wird eine Auswertung des Interviews vorgenommen und abschließend werden die Ergebnisse der Forschungsarbeit erläutert.

Zum Abschluss wird ein kurzes Fazit und ein Ausblick zur Disaggregation allgemein, dem erstellten Prototyp und der durchgeführten Forschungsarbeit abgeleitet.

5.2. Grundlagen

Um die aktuelle Relevanz dieses Projekts aufzeigen zu können, ist es zunächst wichtig, sich einen Überblick über den aktuellen Stand zur Forschung zu verschaffen. In diesem Abschnitt soll der State-of-the-Art zu den Gegenständen mit Bezug zur Lastkurvenanalyse dargestellt werden. Im Folgenden werden dazu gewisse relevante Begrifflichkeiten geklärt, die rechtlichen Gegebenheiten untersucht, sowie auf vergleichbare bestehende Ansätze eingegangen.

5.2.1. Begrifflichkeiten

Energiemessgerät / Intelligente Steckdose Ein Energiemessgerät ist eine „intelligente“ Steckdose, welche die Funktion mit sich bringt, den Energieverbrauch angeschlossener Geräte zu messen und aufzuzeichnen. Ein genaues Energiemessgerät ist für die Teilprojektgruppe vor allem im Anfangsstadium hilfreich, da es die Möglichkeit bietet, einige wenige Geräte anzuschließen und auszuwerten. Vor allem zu Testzwecken ist dies ein Vorteil im Gegensatz zu Stromzählern, die den gesamten Energieverbrauch eines Haushalts messen.

Smart Meter Smart Meter sind intelligente, vernetzte Zähler für Wasser und Energien wie Gas und Strom, die in einem Kommunikationsnetzwerk eingebunden sind und dadurch den tatsächlichen Energieverbrauch und die Nutzungszeit anzeigen können [Pro]. Sie ermöglichen dadurch einen „bidirektionalen Informationsaustausch zwischen Energieversorger und Verbraucher“ [Wol16]. In dieser Teilgruppe werden Smart Meter zur Messung von Wasser und Gas nicht weiter betrachtet, da die Stromenergie im Zentrum steht.

Die Richtlinien der EU zur Energieeffizienz Die Bundesregierung hat im Jahr 2016 beschlossen die Richtlinien der EU, zur Energieeffizienz, in geänderter Form umzusetzen. In diesen Richtlinien steht geschrieben, dass die Mitgliedsstaaten verpflichtet sind, bis 2020 mindestens 80 Prozent der Verbraucher mit intelligenten Verbrauchserfassungssystemen auszustatten. Dies gilt allerdings nur, wenn die Einführung als kostenwirksam angesehen wird [Das12]. Die meisten Mitgliedsstaaten der EU gehen dieser Richtlinie und somit einem landesweiten Rollout von intelligenten Messsystemen nach [Bunb]. In Deutschland hingegen ergab eine Studie, dass der Austausch der Systeme nicht kostenwirksam ist, weshalb hier eine andere Vorgehensweise für die nächsten Jahre verfolgt wird [Ern13] [Bun16b]. Demnach beginnt bei Haushalten oder Firmen, welche einen jährlichen Stromverbrauch von mehr als 10.000 kWh verbuchen, ab 2017 stufenweise der Einbau von intelligenten Messsystemen. Ab einem jährlichen Stromverbrauch von 6.000 kWh wird mit dem Einbau 2020 begonnen.

Lastkurven Die Lastkurve bezeichnet den Verlauf des Energieverbrauchs in einer bestimmten Zeitperiode. In der Energiewirtschaft werden Lastkurven analysiert, um so Prognosen über die Bedarfe aufstellen zu können. So lässt sich herausfinden, zu welchen Zeitpunkten Lastspitzen erreicht werden, um somit die Energieerzeugung optimal zu steuern. Die tatsächlichen Lastkurven von Verbrauchern von mehr als 100.000 kWh im Jahr werden, im Gegensatz zu Verbrauchern mit geringem Stromverbrauch, im Viertelstunden-Takt gemessen. Da dies bei Geringverbrauchern zu aufwändig wäre, werden hier Standardlastpro-

file verwendet [Wol16]. Diese gemessenen Lastkurven zeichnen allerdings nur den gesamten Stromverbrauch eines Verbrauchers (z. B. Haushalt, Fabrik, Unternehmen) auf. Die Daten sind also aggregiert. Das bedeutet, dass sich anhand der Kurve nicht bestimmen lässt, welches Gerät für welchen Anteil des Stromverbrauchs verantwortlich ist. Auf dem Markt gibt es bereits einige wenige Ansätze, die versuchen, diese Kurven zu disaggregieren. Diese werden im folgenden Unterkapitel vorgestellt.

Disaggregation/NILM Im Fokus dieser Arbeit stehen die Disaggregation von Lastkurven privater Haushalte. Disaggregation bedeutet in diesem Zusammenhang das Aufschlüsseln einer Datenmenge anhand bestimmter Merkmale in unterschiedliche Einzelteile. Im Gegensatz dazu steht die Aggregation, die eine Zusammenführung von Einzelnen Daten zu einer gesamten Datenmenge beschreibt. Dies ist die absolute Grundlage der Teilprojektgruppe. NILM bedeutet übersetzt Nonintrusive Load Monitoring. Hierbei sind keine Zwischenmessstationen erforderlich und Geräte müssen über keine eigene Recheneinheit verfügen. Dies ermöglicht das Disaggregieren von Lastkurven lediglich anhand eines Smart Meters. Darüber hinaus ist eine Software die NILM benutzt einfach in ihrer Anschaffung und darüber hinaus skalierbar. Ableitend daraus lässt sich so feststellen, dass Lastkurven, die einen Stromverbrauch über einen gewissen Zeitraum widerspiegeln, aufgelöst werden sollen, und somit die konkreten Stromverbraucher erkennbar sind. Das bedeutet, dass aus der Gesamtlastkurve die einzelnen Lastkurven der Stromverbraucher herausgelöst werden.

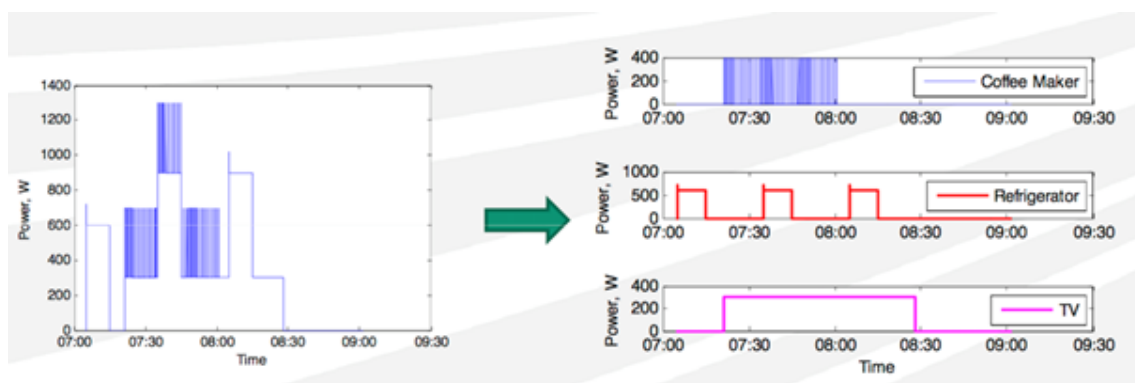


Abbildung 35: Überblick Vorgehen bei Disaggregation [Fra11]

Mit Hilfe der Abbildung 35 lässt sich das Prinzip verdeutlichen. Auf der linken Seite ist eine Gesamtlastkurve zu sehen, die auf der X-Achse die Uhrzeit zeigt und auf der Y-Achse den Watt-Wert als Verbrauchswert. Der Blaue Graph stellt dadurch den jeweiligen gesamten Stromverbrauch zu einer bestimmten Uhrzeit dar. Die disaggregierten Lastkurven sind

auf der rechten Seite zu sehen. Ebenfalls in einem Zeit-Watt-Diagramm sind die einzelnen Stromverbraucher erkennbar. Bei der vorliegenden Disaggregation kam heraus, dass die Kaffeemaschine, im oberen Diagramm zu sehen, von ca. 7:15 Uhr bis 8 Uhr eingeschaltet war. Der Eisschrank hatte um 7:10 Uhr um 7:40 Uhr und um 8:10 Uhr für jeweils 5 Minuten kurze Lastspitzen, was ebenfalls auf eine Nutzung zurückzuschließen ist. Das unterste Diagramm zeigt den TV, der von 7:20 bis 8:30 durchgehend genutzt wurde. Wenn nun wieder das linke Diagramm, also die Gesamtlastkurve betrachtet wird, ist erkennbar, dass diese die Aufsummierung, bzw. Überlagerung der einzelnen Lastkurven der eingeschalteten Geräte ist. Ein wichtiges Kriterium für die Disaggregation ist jedoch die Abtastrate der aggregierten Lastkurven, also die Frequenz der Daten. Dies wird ebenfalls in den folgenden Diagrammen 36 und 37 sichtbar.

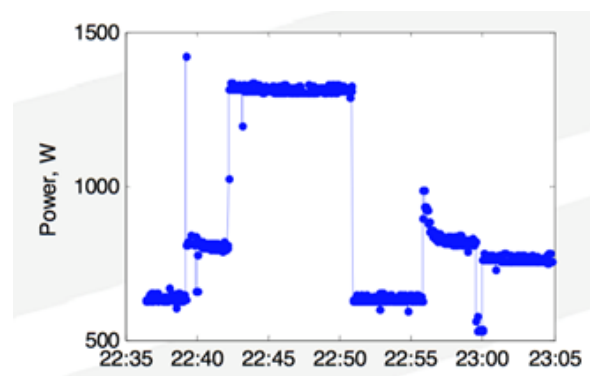


Abbildung 36: Verbrauchsdaten mit niedriger Auflösung [Fra11]

Das Schaubild 36, auch ein Zeit-Watt-Diagramm (der Aufbau wurde bereits beschrieben), zeigt eine aggregierte Lastkurve. Hierbei wird von einer geringen Abtastrate von ca. 1 Hertz gesprochen. Das bedeutet, dass die Messpunkte in diesem Diagramm eine Sekunde auseinanderliegen. So können eindeutige Profile von Geräten gut erkannt werden. Dies ermöglicht einen Ansatz die Endgeräte über die Veränderung in der Leistungsaufnahme zu erkennen. Das bedeutet, dass durch mehr eingeschaltete Geräte sich die Lastkurve nach oben vergrößert und so auf die einzelnen Endgeräte geschlossen werden kann.

Im Schaubild 37 ist eine Kurve mit einer hohen Abtastrate zu sehen. Hierbei ist die Abtastrate mit mehr als 5 Kilohertz sehr hoch, dies entspricht 5000 Messungen pro Sekunde. Damit lassen sich Ansätze über die Veränderungen von harmonischen Schwingungen verfolgen. Dies ist aber aufgrund technischer Grenzen innerhalb dieses Prototyps nicht möglich.

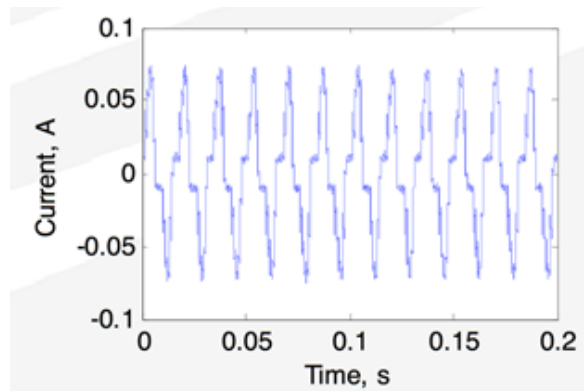


Abbildung 37: Verbrauchsdaten mit hoher Auflösung [Fra11]

NILM-Toolkit Zur Disaggregation von Lastkurven gibt es in der Literatur eine Vielzahl von Algorithmen und verschiedenen Herangehensweisen. Da diese nicht ohne Weiteres verglichen werden können, wurde im April 2014 von einer Gruppe amerikanischer Entwickler das sogenannte *NILM-Toolkit* entwickelt. Dies ist ein Toolkit für Wissenschaftler, um die Genauigkeit von NILM-Algorithmen zu untersuchen. Dieses Toolkit ermöglicht den Einsatz von verschiedenen öffentlichen Datensätzen, die Vorverarbeitung der Daten, die Disaggregation mit Hilfe verschiedener Algorithmen und den Einsatz von Metriken zur Messung der Genauigkeit [Jac16].

Innerhalb des Toolkits wird *NILM-Metadata* genutzt, ein Metadaten-Framework zur Beschreibung von Daten im Zusammenhang von *NILM*. Es dient also als Schema und generelle Struktur für Daten wie Messgeräte, Gebäude, Verbraucher und vielem mehr [Jac15].

5.2.2. State-of-the-Art / Bestehende Ansätze

Die Analyse von Lastkurven ist ein immer weiter wachsender Forschungsbereich. Ursprung findet diese Thematik in einem Bericht aus dem Jahre 1985 von George W. Hart am Massachusetts Institute of Technology im Bereich Electric Power Research der sich mit einem Prototyp für einen Nonintusive Appliance Load Monitor beschäftigt. Mitte Mai 2016 fand in Vancouver, Kanada der dritte internationale Workshop für NILM statt. Im Fokus stand hier der Austausch und die Vorstellung verschiedener Ansätze für die Lastkurvenanalyse [Ste16b].

Die vorhandenen Ansätze der Lastkurvenanalyse und die damit verbundenen Zielsetzungen werden oft hinsichtlich der zugrundeliegenden Datenfrequenz kategorisiert. Die erste wichtige Kategorie beinhaltet Daten mit einer Aufnahme­frequenz von 1 Mal pro Stunde bis hin zu alle 15 Minuten. Mit diesen Daten lassen sich noch keine individuellen Geräte aus

dem Gesamtverbrauch erkennen. Die nächste Kategorie hat eine Aufnahmefrequenz von 1 Mal pro Minute bis hin zu 1 Hertz. Diese Frequenz ermöglicht es etwa die größten zehn Verbraucher im Haushalt zu erkennen und ist zudem realistisch für die technische Umsetzung von Lastkurvenanalysen eines Haushalts, da die Frequenz auch mit herkömmlichen intelligenten Stromzählern erreicht werden kann. Die Algorithmen, die in diesem Bereich eingesetzt werden basieren auf der Analyse von sogenannten *Steady state steps/transitions*, also grundsätzlich den Veränderungen der Stromaufnahme der einzelnen Geräte. Die Aufnahme höherer Frequenzen bis hin zu mehr als 1 Megahertz werden in der Forschung untersucht. Da diese Anforderung in kaum einem Haushalt realisiert werden kann, kommt diese Aufnahme für die breite Nutzung nicht in Frage. Mit diesen Daten und entsprechenden Algorithmen können theoretisch alle Verbraucher identifiziert werden, sogar Geräte die ganztägig aktiv sind [K. 12].

Die Basis für alle vorhandenen Ansätze bilden die Datensätze von aufgezeichneten Verbrauchsdaten. Diese Daten werden dazu genutzt den Algorithmus zu trainieren und so die Disaggregation der Lastkurven zu optimieren. Für Forschungszwecke stehen eine Menge von öffentlichen Datensätzen zur Verfügung. Die Qualität dieser Datensätze hängt von verschiedenen Faktoren ab: Wichtig sind hier der Zeitraum in dem aufgezeichnet wurde, die Anzahl der Haushalte die aufgezeichnet wurden und die Aufnahmefrequenz. Einer der ersten und bekanntesten Datensätze ist das *REDD-Dataset*. Dies ist ein Datensatz aus dem Jahre 2011, welcher durch das *Massachusetts Institute of Technology* bereitgestellt wurde. Dieser Datensatz umfasst Verbrauchsdaten von sechs Haushalten mit einer Aufnahmefrequenz von 3 Sekunden über einen Zeitraum von 3 bis 19 Tage. Diese Daten wurden extra für die NILM-Forschung aufgezeichnet [Zic11].

Aber auch in deutschen Universitäten wird sich mit dem Thema der Lastkurvenanalyse auseinandergesetzt, eine bekannte Anlaufstelle zu diesem Thema ist das Institut für Signalverarbeitung und Systemtheorie an der Universität Stuttgart. Genauer im Forschungsgebiet Signalverarbeitung für Energiemonitoring (NILM) werden verschiedene Ansätze und Methoden zur Signalverarbeitung erforscht [Ren]. Zwei Forschungsschwerpunkte bilden die aktuellen NILM Ansätze, die sich zum einen mit dem *Hidden Markov Model* und zum anderen mit der Nutzung neuronaler Netze zur Lastkurvenanalyse beschäftigen [Luk16] [Kar16]. Diese beiden Ansätze waren auch Bestandteil des internationalen Workshops in Kanada.

Kommerzielle Lösungen Im Rahmen der Lastkurvenanalyse und der Disaggregation von Lastkurven existieren bereits Unternehmen, welche sich dieser Problemstellung annehmen. Das Unternehmen *Bidgely* bietet die Möglichkeit personalisierte Informationen, wie den

Stromverbrauch, direkt auf einem Endgerät darzustellen. Außerdem nutzt *Bidgely* diese Informationen, um den Bedarf der Verbraucher regulieren zu können. Der Verbraucher sieht anhand der App, wie hoch der einzelne Stromverbrauch seiner Endgeräte ist und kann damit seinen Bedarf verringern und anpassen. Weiter bietet *Bidgely* eine Statuserkennung der Endgeräte an, beispielsweise ob ein Gerät ein- bzw. ausgeschaltet ist. Der Verbraucher hat zusätzlich die Möglichkeit *Bidgely* so einzustellen, dass die Innenbeleuchtung der Behausung automatisch ein- bzw. ausgeschaltet wird beim Betreten bzw. Verlassen des Gebäudes. Abschließend bietet *Bidgely* also eine umfangreiche Bedarfsregulierung an [Bid16].

Smappee ist ein weiteres Unternehmen, welches sich der Erkennung des Stromverbrauchs widmet. Dieses Unternehmen nutzt eine direkte Ablesung an einem Stromkasten, welche Echtzeitdaten über den Stromverbrauch und dessen Kosten liefert. Weiter ist eine automatische Fernsteuerung möglich indem die entsprechende App verknüpft wird [Sma16]. Ähnlich wie *Smappee* beschäftigt sich das Unternehmen *Greenpocket GmbH* mit dem Speichern von Daten über einen längeren Zeitraum. Dies dient keiner Bedarfsbeurteilung, sondern einer Bedarfsberechnung [Gre].

5.2.3. Einsatzmöglichkeiten

Die Ergebnisse einer ausgereiften Disaggregation sind genaue Daten über Laufzeit und Stromverbrauch einzelner Geräte. Die Daten die hinter diesen Ergebnissen stehen, haben in den verschiedensten Bereichen großes Potential. Daraus ergibt sich auch ein großes Interesse an dieser Art Informationen für Privatleute, aber auch für Unternehmen aus den verschiedensten Bereichen.

Wie bereits in der Motivation ausgeführt liegt das Potential für eine Privatperson in der Transparenz. Auf Grundlage der Disaggregationsergebnisse ist es dem Kunden möglich ein besseres Bewusstsein für seinen eigenen Energieverbrauch zu schaffen. Entstandene Kosten und Energieverbräuche können einzelnen Geräten zugeordnet werden. Mit diesen Daten gibt man einer Privatperson das Werkzeug, um sein eigenes Verbrauchsverhalten zu analysieren und auch zu optimieren. Die Daten ermöglichen auch festzustellen, ob einzelne Geräte einen besonders hohen Stromverbrauch haben. Zum Beispiel, weil ein Gerät defekt ist oder da es aufgrund eines bestimmten Alters energietechnisch ineffizient ist. Dies stellt einen klaren Mehrwert zu Informationen dar, die man durch das Ablesen analoger Stromzähler bekommen kann [K. 12].

Aber auch im gewerblichen Bereich können die Daten in den verschiedensten Bereichen eingesetzt werden. Dabei besteht auch das Potential mit dieser Technologie alte Prozesse zu verbessern. Für einen Netzbetreiber beispielsweise könnten die genauen Daten genutzt

werden, um bessere Lastprognosen für die Netze zu erstellen. Weg von Standardlastprofil helfen diese präzisen Daten neue Modelle für die Lastberechnung zu erstellen. Aber auch im Bereich Forschung und Entwicklung kann diese Technologie eingesetzt werden, um neue Entwicklungen zu testen. Die Auswertung der Daten mit Disaggregation ermöglicht einen einfachen Ist- und Soll-Vergleich und erzeugt damit ein Feedback, ob sich die neuen Produkte energietechnisch so verhalten wie geplant. Damit kann die Disaggregation dabei helfen, energieeffizientere Produkte zu entwickeln [K. 12].

Auch in Richtung der Analyse von Verbrauchsdaten mehrerer Maschinen in Produktionen ist denkbar. Verändert sich das Verbrauchsverhalten einzelner Maschinen könnte das auf einen Defekt hinweisen. Der Einsatz ermöglicht sogar die Nutzung von Predictive Maintenance, also das Erkennen von Defekten an Maschinen, bevor diese aufgetreten sind.

An einem konkreten Use Case mit Disaggregation arbeitet zurzeit das Unternehmen Oldentec aus Oldenburg. Die Firma Oldentec⁴ entwickelt technische Assistenzprodukte für den Bereich der Pflege. Mit Hilfe der Disaggregation versucht das Unternehmen, die herkömmlichen Tagestaster für Senioren zu ersetzen. Der Tagestaster ist ein technisches Gerät, durch den Senioren signalisieren können, dass ihnen nichts zugestoßen ist. Dieser muss täglich betätigt werden, doch da Senioren wiederholt dies vergessen, entstehen hohe Kosten, da durch einen Pflegemitarbeiter sichergestellt werden muss, dass alles in Ordnung ist. Mit Hilfe der Disaggregation soll erkannt werden, ob bestimmte Geräte im Haushalt an einem expliziten Tag genutzt wurden. Wurde erkannt, dass ein Gerät genutzt wurde, ersetzt dies das Betätigen des Tagestasters. Dieser Prozess ist ein sehr simples Beispiel für den Einsatz von Disaggregation, aber verdeutlicht in welchen verschiedenen Bereichen diese Technologie eingesetzt werden kann.

5.3. Das Konzept

In den vorangegangenen Kapiteln wurde die vorliegende Problemstellung erläutert und diese über Ausführungen zum State-of-the-Art thematisch eingeordnet. Im folgenden Kapitel wird auf Basis dieser Grundlagen ein Anwendungsszenario für den Disaggregationsprozess definiert. Passend dazu werden daraufhin mit Hilfe eines Business-Model-Canvas mögliche Geschäftsmodelle visualisiert, die sich durch das Anwendungsszenario ergeben. Das entwickelte Szenario wird als Grundlage zur Erstellung einer Grobarchitektur des Prototyps verwendet. Die einzelnen Komponenten sollen dabei die Funktionalitäten abdecken, um die einzelnen Bestandteile des Anwendungsszenarios zu ermöglichen. Zum Abschluss werden zu diesen Komponenten verschiedene qualitative und quantitative Anforderungen definiert.

⁴ <http://www.oldntec.eu/>

Zusammen bilden diese Teile des Konzeptes die Grundlage für die Implementierung des Prototyps.

5.3.1. Anwendungsszenario

Um den Nutzen und die Funktion des zu entwickelnden Systems übersichtlicher zu gestalten, wurde dies mit Hilfe eines Use Case Diagramms visualisiert. Dieses Anwendungsszenario hilft dabei die Anforderungen an einzelne Systemteile und das Zusammenspiel einzelner Akteure besser zu analysieren. In Abbildung 38 wird dieses Diagramm gezeigt. Der Kunde hat verschiedene Verbraucher an einem intelligenten Stromzähler oder an eine intelligente Steckdose angeschlossen, welche die Verbrauchsdaten der angeschlossenen Geräte speichern. Der Einfachheit halber sind die elektrischen Verbraucher in diesem Use Case nicht explizit ausgewiesen, sondern gehören dem Akteure Messstation an.

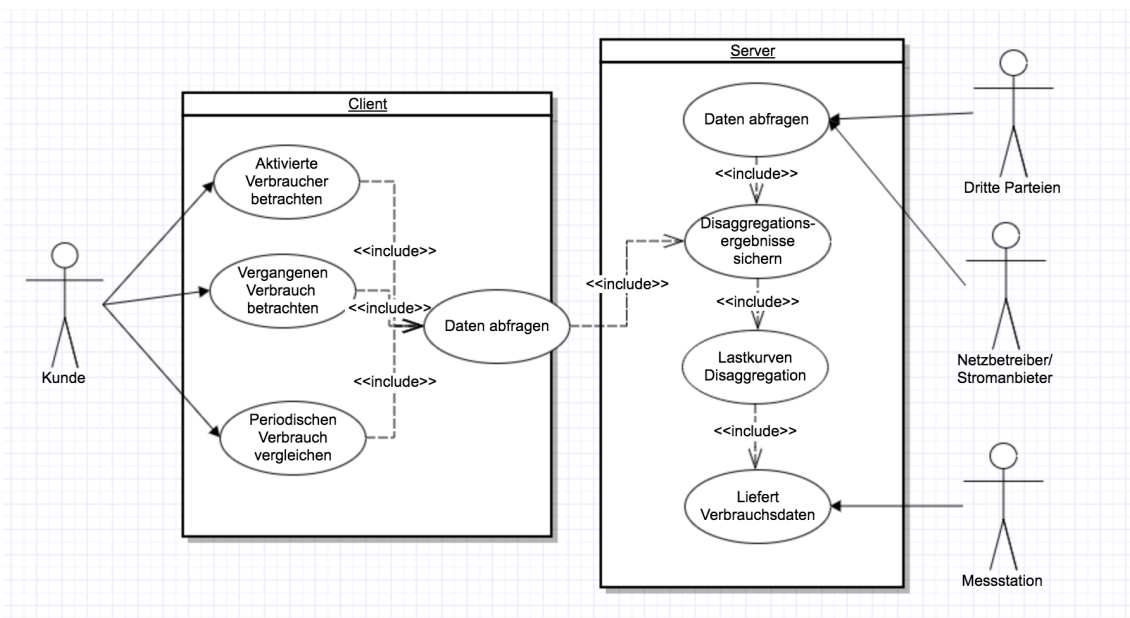


Abbildung 38: Use Case Diagramm für das entwickelte Szenario [Eigene Quelle]

Der Akteur der im Mittelpunkt dieses Anwendungsfalls steht, ist der Kunde. Der Kunde kann über einen Client bestimmte Informationen aufrufen. Beispielsweise kann der Client einfach eine native App auf dem Smartphone sein. Über diesen Client hat der Kunde die Möglichkeit drei verschiedene Use Cases aufzurufen, die im Diagramm durch drei Ellipsen dargestellt sind. Diese Use Cases spielen sich nur im Rahmen des Clients ab, der im Diagramm durch ein umfassendes Rechteck mit entsprechender Kennzeichnung dargestellt ist.

Diese Use Cases sind zum einen das Betrachten der aktivierten Verbraucher, das Visualisieren vergangener Verbrauchsdaten und das Vergleichen periodischer Verbrauchsdaten.

Die Grundlage dafür, dass diese Anwendungen vom Kunden durchgeführt werden können, ist das Abfragen der Daten vom Server. Diese Abhängigkeit wird im Diagramm durch eine Include-Verbindung gekennzeichnet. Diese Abfrage geschieht noch im Rahmen des Clients. Die nächste Abhängigkeit die von der Abfrage der Daten des Clients ausgeht, greift nun auf den eigentlichen Server zu. Dies wird auch durch ein Rechteck mit der Bezeichnung Server im Use Case Diagramm abgebildet. Die Bedingung für das Abfragen der Daten ist, dass die Ergebnisse der Disaggregation auf dem Server vorliegen.

Der Nächste Akteur der direkt mit dem Server kommuniziert ist die Messstation. Diese Messstation liefert aufgezeichnete Verbrauchsdaten an den Server. Die gelieferten Daten sind Bedingung für die Disaggregation der Lastkurven, dessen Ergebnisse auch selbst wieder Bedingungen für das Abspeichern der Disaggregationsergebnisse sind. Hier besteht ebenfalls eine Verbindung zwischen den beiden Akteuren Kunde und Messstation, da der Kunde über den Client auf die Daten zugreift, die von der Messstation geliefert werden.

Ein weiterer Akteur in diesem Anwendungsszenario ist der Stromanbieter bzw. der Netzbetreiber. Diese sind auch an den Informationen und den Verbrauchsprofilen der Nutzer interessiert. Der Use Case der vom Netzbetreiber ausgeht ist das Ablesen der Daten vom Server des Systems. Wie auch schon beim Zugriff durch den Client des Kunden ist die Bedingung, dass die Ergebnisse der Disaggregation auf dem Server gesichert sind. Parallel dazu können auch weitere Dienstleister auf die Daten zugreifen, um damit andere Services bereitzustellen.

5.3.2. Business-Model-Canvas

Neben dem erstellten Anwendungsszenario ist das Business Model Canvas eine Variante, um beabsichtigte Geschäftsmodelle tabellarisch zu visualisieren. Für die Teil-Projektgruppe energy of WOW wurde daher ein angepasstes Modell entwickelt.

Das *vorliegende Business-Modell-Canvas* ist in neun Bereiche aufgeteilt. Die Schlüsselpartner der Teilprojektgruppe sind vorwiegend die Partner, die der gesamten Projektgruppe zur Verfügung stehen. Das sind die Universität, the peak lab und Mitarbeiter von Enera. Speziell diese halfen dabei, benötigte Hardware für den Prototyp zu entwickeln. Darüber hinaus waren Schlüsselpartner für die Evaluation des Prototyps nötig, die hier noch einmal gesondert genannt werden, da diese wesentlichen Denkanstöße für die Entwicklung einer mobilen Applikation gegeben haben. Die Vorteile dieser Partner, was in dem Business-Model Canvas explizit aufgeführt wird, liegen darin, dass der Teilprojekt-

Schlüsselpartner	Schlüsselaktivitäten	Nutzenversprechen	Kundenbeziehung	Kundenarten
- Uni/VLBA - thepeaklab - diverse Experten	- Lastkurvenanalyse - - Datenvisualisierung - Datenverwaltung	- Überblick über seinen Energieverbrauch - Kundengerechte Beeinflussung des Energieverbrauchs - Bessere Kommunikation zwischen Netzbetreibern und Stromanbietern	- Bessere Kommunikation zwischen Kunden und Netzbetreibern/Strom-Anbietern - Automatisierte Dienstleistung	- private Haushalte - Stromanbieter - Netzbetreiber - Gerätehersteller ▪ externe Dienstleister
Vorteile von Partnern	Schlüsselressourcen		Vertriebs- und Kommunikationskanäle	
-Räume, Hard-/Software - Ressourcen, Expertenwissen, Bezug zur Realität Expertenwissen, Praxis	- Lastkurven privater Haushalte - Expertenwissen -Geräte zur Auswertung -Server zur Datenverwaltung		- Website - App (?) - Social Media	

Abbildung 39: Business Model Canvas [Eigene Quelle]

gruppe verschiedene Arten von Ressourcen zur Verfügung gestellt werden. Diese erstrecken sich von Räumen, Hard- sowie Software über Experten- und Fachwissen aus der Praxis.

Bei den Schlüsselaktivitäten wurde zwischen drei Begriffen unterschieden. Die Lastkurvenanalyse stellt den Kern dieser Arbeit da, da aus diesem die Informationen gewonnen werden. Diese werden für eine Datenvisualisierung bereitgestellt, da der Konsument über diese Informationen auf dem einfachsten Wege verfügen soll. Letztlich stellt die Datenverwaltung eine dritte, wichtige Schlüsselaktivität dar, da sich hier Gedanken um die Datenflüsse und Datenbanken gemacht werden muss. Die sogenannten Schlüsselressourcen lassen sich leicht von den Aktivitäten und Vorteilen von Partnern erkennen. Gerade zu Trainingszwecken werden Lastkurven benötigt, seien es Datensätze aus dem Internet oder tatsächliche Verbrauchskurven von privaten Haushalten. Darüber hinaus ist eine Plattform notwendig, um die Datenauswertung bzw. die Disaggregation durchführen zu können. Ein Server zur Datenverwaltung bildet das letzte Glied der Datenkette. Abgerundet werden die Schlüsselressourcen durch das bereits genannte Expertenwissen was der Projektgruppe von den Praxispartnern aber auch Fachliteratur zugetragen wird.

Das Nutzenversprechen für den Konsumenten liegt daher auf der Hand. Der Prototyp soll diesem einen Überblick über den Energieverbrauch bieten und darüber hinaus eine Kundengerechte Beeinflussung des Energieverbrauchs. Daraus könnte dann eine bessere

Kommunikation zwischen Netzbetreibern/ Stromanbietern und Konsumenten entstehen. Im Optimalfall ist die Kundenbeziehung eine automatisierte Dienstleistung, unterstützt von einer mobilen Applikation. Vorzugsweise richtet sich der Prototyp an die privaten Haushalte, also normale Konsumenten. Weitere Kundenarten, oder Personen/Unternehmen, die einen Nutzen von dem Prototyp haben könnten wären Stromanbieter, Netzbetreiber, Gerätehersteller und externe Dienstleister. Die konkreten Einnahmequellen könnten hier die neuen Absatzmöglichkeiten für die Stromanbieter/Netzbetreiber sein. Es können anonymisierte Daten an Gerätehersteller und Dienstleister weitergegeben werden, oder besondere Dienstleistungen dem Konsumenten angeboten werden. Realisiert kann dies durch eine Distribution über eine Website, mobile Applikation, Drittanbieter oder Social Media geschehen.

5.3.3. Grobarchitektur und Komponenten

Auf Basis des erstellten Anwendungsszenarios wurde ein Gesamtmodell erstellt, welches alle nötigen Komponenten beinhaltet, die zur Lösung der Teilprobleme notwendig sind. Abbildung 40 zeigt die Grobarchitektur dieses Modells, dessen Komponenten im Folgenden genauer beschrieben werden. Die Architektur zeigt abstrahiert den Datenfluss der von einer intelligenten Messstation bis hin zur Visualisierung auf einem Endgerät oder dem Ablegen von Daten in einer Cloud.

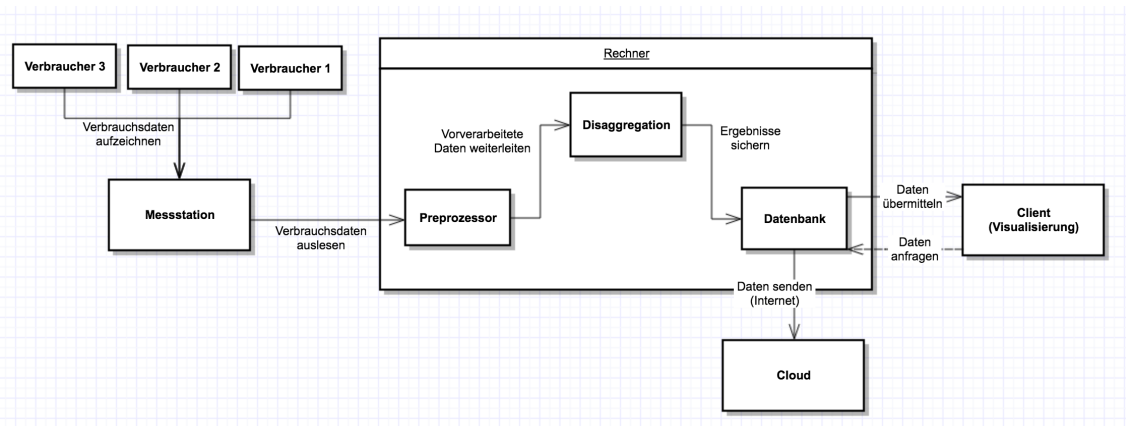


Abbildung 40: Grobarchitektur Lastkurvenanalyse [Eigene Quelle]

Komponente Messstation Die erste Komponente der Architektur stellt die Messstation dar. An so eine Messstation sind eine bestimmte Menge von elektronischen Verbrauchern angeschlossen und sie hat die Aufgabe, den aggregierten Stromverbrauch aller angeschlos-

senen Geräte in der Summe aufzuzeichnen. Dabei ist es nicht festgelegt, um welche Art von Messstation es sich handelt. Dies kann zum einen ein Stromzähler sein, der zum Aufzeichnen des Stromverbrauchs eines gesamten Haushaltes genutzt wird. Zum anderen kann dies aber auch ein einfaches Strommessgerät sein, das vor das Netzteil eines Gerätes oder vor eine Steckdosenleiste geschaltet wird.

Komponente Preprozessor Die zweite Komponente, der Preprozessor, ist ein Bestandteil eines Servers, der in Verbindung zu einer oder mehreren Komponenten Messstation steht. Der Preprozessor ist also eine Softwarekomponente und ist wie in Abbildung 40 zu sehen, ein Teil einer übergeordneten Komponente Server. Die Aufgabe des Preprozessors ist es, in einem ersten Schritt die aufgezeichneten Verbrauchsdaten über die vorhandene Schnittstelle von der Messstation in einer angemessenen Frequenz auszulesen. Die Daten, die über die Schnittstellen abgerufen werden, sind die Lastkurven, die den Stromverbrauch der einzelnen Verbraucher aggregiert anzeigen. In einem zweiten Schritt müssen die erhaltenen Daten für die nächsten Komponenten vorverarbeitet werden. Dazu werden die Daten aufbereitet und in einem entsprechenden Format an der vorgesehenen Stelle abgelegt.

Komponente Disaggregation Die vorverarbeiteten Daten, die von der Komponente Prozessor zur Verfügung gestellt werden, nutzt die Disaggregation für die weitere Verarbeitung. Wie auch der Preprozessor ist die Disaggregation eine Softwarekomponente, die im gleichen Server vorhanden ist. Die Daten liegen nun in aggregierter Form von den verschiedenen Messstationen vor und die Aufgabe der Disaggregation ist es, die gerätespezifischen Anteile verschiedener Geräte aufzuschlüsseln. Das Ergebnis dieses Schrittes sind also Daten, die Auskunft darüber geben, welche Geräte zu welcher Zeit eingeschaltet waren oder noch sind.

Komponente Datenbank Nach der Disaggregation sollen die Daten für die weitere Verwendung gesichert werden. Um diese Daten strukturiert abzulegen, werden diese nach der Disaggregation in eine Datenbank geschrieben oder gegebenenfalls in einem sinnvollen Format abgelegt. Diese Datenbank dient dann als Ankerpunkt für andere Systeme, die auf diese Daten zugreifen wollen. Auch die Datenbank ist auf dem gleichen Server vorhanden wie vorangegangenen Komponenten.

Komponente Visualisierung Die Komponente Visualisierung ermöglicht es dem Nutzer, Informationen über den Status der elektronischen Verbraucher im Haushalt auf einem Client aufzurufen. In einem ersten Schritt werden die notwendigen Informationen aus der

Komponente Datenbank abgerufen. Diese Daten werden dann vom Client in angemessener Weise visualisiert und bieten dem Nutzer verschiedene Möglichkeiten diese Informationen zu nutzen. Je nach Endgerät an dem die Daten aufgerufen werden, muss eine entsprechende Schnittstelle vorhanden sein. Für ein Smartphone müssen diese Daten über das Internet oder ein internes Netzwerk erreichbar sein.

Cloud Neben der Möglichkeit die Daten auf einem Endgerät zu visualisieren, bietet hier die Komponente Cloud einen weiteren Ansatz, die ermittelten Daten weiter zu nutzen. Die Daten, die in der Komponente Datenbank abgelegt sind, können über das Internet an eine Cloud gesendet werden. Welche Art von Cloud das ist und welche Services man auf Basis dieser Daten anbieten kann, gilt es zu erforschen.

5.3.4. Anforderungen

Im letzten Abschnitt wurden die einzelnen Komponenten und deren Beziehungen untereinander mit Hilfe einer Grobarchitektur erläutert und eingeordnet. In diesem Abschnitt sollen die Komponenten hinsichtlich ihrer funktionalen Anforderungen genauer beleuchtet werden. Diese Anforderungen ergeben sich aus dem erstellten Anwendungsszenario, das zu Anfang des Kapitels beschrieben wurde.

Komponente Server Für die Disaggregation, die Lagerung von Daten und die Bereitstellung von Daten wird ein Server benötigt. Dieser Server übernimmt die Aufgabe mehrerer Funktionalitäten und muss daher ein breites Spektrum an Anforderungen abdecken.

Zu Anfang einmal ist es die Aufgabe des Servers, die Daten von der Messstation entgegenzunehmen und diese in einer Datenbank zu speichern. Dafür muss der Server die benötigten Schnittstellen bereitstellen, um die Daten abfragen zu können. Je nachdem, wie die Daten von der Messstation angeboten werden, muss der Server entweder eine Hardwareschnittstelle oder eine Funkschnittstelle (z. B. für WLAN) besitzen. Da die Verbrauchsdaten dauerhaft aufgezeichnet werden sollen, muss der Server auch für den dauerhaften Betrieb ausgerichtet sein. Damit soll gewährleistet werden, dass auch langfristig die Daten aufgezeichnet und gespeichert werden können. Da die Verbrauchsdaten verschiedener elektrischer Verbraucher an verschiedenen Orten aufgezeichnet werden müssen, sollte der Server auch in gewissem Maße mobil sein. Diese Anforderung ist aber nur notwendig, wenn die Daten über eine Hardwareschnittstelle übertragen werden. Wird ein Netzwerk für die Übertragung genutzt, ist Mobilität und Standort nicht relevant.

Des Weiteren muss der Server auch Hard- und Softwarevoraussetzungen erfüllen, um alle softwaretechnischen Komponenten wie etwa Disaggregation und Datenbank ausführen

zu können. In Bezug auf Hardwarevoraussetzungen muss die Rechenleistung vorhanden sein, um die entwickelten Algorithmen auszuführen. Auf dem Server selbst müssen dazu noch ein entsprechendes Betriebssystem und passende Software verfügbar sein, die alle Funktionalitäten abdecken.

Komponente Messstation Die Funktionalität einer Messstation lässt sich in zwei grundlegende Teile unterteilen: Zum einen muss die Messstation die Verbrauchsdaten kontinuierlich und in einer angemessenen Frequenz aufzeichnen. Zum anderen müssen diese Daten über eine Schnittstelle synchron auszulesen sein. Dies kann nach Art der Messstation eine Hardware- oder eine Funkschnittstelle sein. Die Messstation selbst muss die Möglichkeit bieten Verbrauchsdaten aggregiert und für einzelne Verbraucher selbst aufzeichnen zu können. Da elektrische Verbraucher oft fest installiert sind, muss auch die Messstation zusammen mit dem Server mobil sein.

Komponente Preprozessor Der Preprozessor muss ein Service sein, der kontinuierlich Daten von der Messstation aufzeichnet und diesen Datenstrom weiterverarbeiten kann. Je nachdem in welcher Form die Daten aufgezeichnet werden, müssen diese aufbereitet werden. Der Preprozessor hat die Aufgabe die Rohdaten so zu Verfügung zu stellen, dass sie von weiteren Komponenten genutzt werden können.

Komponente Disaggregation In dieser Komponente muss ein Algorithmus umgesetzt werden, der auf Basis der Verbrauchsdaten eine Disaggregation durchführen kann, die durch die Messstation aufgenommen wurden. Mit den Daten die von einzelnen Verbrauchern aufgenommen wurde, soll der Algorithmus trainiert werden. Durch das trainierte Modell können aggregierte Lastkurven der Messstation disaggregiert werden. Die Frequenz der Daten sollte etwa 0,5 bis 1 Hertz betragen, da handelsübliche Stromzähler das Auslesen dieser Daten ermöglichen. Der Algorithmus soll damit die größten Verbraucher identifizieren können. Das Ergebnis der Disaggregation sollen wieder Lastkurven sein, die aber den einzelnen Gerätetypen zugeordnet sind. Diese Ergebnisse sollen auch langfristig wieder in einer Datenbank abgelegt werden.

Datenbank Die Datenbank der Prototypen muss vielseitig einsetzbar und ansprechbar sein. Es müssen große Datenmengen der Messstation, Ergebnisse der Disaggregation gespeichert werden. Ein weiterer Teil der Anforderung an die Datenbank ist eine Schnittstelle, über die die Daten für verschiedene andere Komponenten bereitgestellt werden können. Die Datenbank selber sollte auf dem Server liegen.

Komponente Visualisierung Ein Hausbewohner kann über diese Komponente auf einem Client Informationen über den Stromverbrauch aufrufen. Dazu gehört die Information, wie viel Strom zurzeit verbraucht wird. Hierzu soll sich der Nutzer weitere historische Daten über den Gesamtverbrauch anzeigen lassen können: Also wie viel Strom habe ich heute, letzte Woche oder auch das gesamte Jahr verbraucht. Falls entsprechende Daten vorliegen, kann der Nutzer den Verbrauch mehrerer Tage/ Wochen/ Jahre miteinander vergleichen. Neben den aggregierten Verbrauchern kann sich der Nutzer auch visualisieren, welche großen Verbraucher zurzeit aktiv sind und welchen Anteil des Verbrauchs diese ausmachen. Die Funktionalitäten auf Basis der gesammelten Daten lassen sich dann modular erweitern, diese könnten Empfehlungssysteme, ein Gamificationansatz und weitere Visualisierungsmöglichkeiten sein. Die Komponente soll also entsprechende Funktionalitäten bieten, um die Daten die in der Datenbank abgelegt sind für den Endnutzer visuell aufzuarbeiten.

5.3.5. Zusammenfassung

Durch die Konzeption des Projektes wurde ein Überblick über Use Cases, Grobarchitektur und mögliche Geschäftsfelder geschaffen. Das Konzept dient im weiteren Verlauf als wichtige Grundlage für die Implementierung des Prototyps, welche im nachfolgenden Kapitel näher erläutert wird.

Die Darstellung der Anwendungsfälle zeigt das Zusammenspiel der verschiedenen Systeme und Akteure des Teilprojekts. Die beiden elementaren Systeme des Prototyps bilden der Server und der Client. Die Server-Seite deckt die Datenerfassung, -berechnung und -speicherung ab. Die hierfür nötigen Daten stammen aus der Messstation, welche die Verbrauchsdaten der Verbraucher aufzeichnet. Außerdem werden auf dem Server Schnittstellen zur Datenfreigabe benötigt, auf welche Energieversorger, Dienstleister und vor allem der Client des Kunden zugreifen könnten. Der Privatkunde steht für diesen Prototyp im Mittelpunkt. Über einen Client, welcher eine mobile Applikation darstellen kann, kann dieser die Disaggregationsdaten des Servers aufrufen, sodass der Kunde sowohl aktuelle Verbräuche, als auch periodische vergangene Werte abrufen kann.

Durch das Business Model Canvas wird ein Überblick über die wichtigsten Schlüsselfaktoren für mögliche Geschäftsmodelle dargestellt. Dabei stellen sich die Partner der Universität, von the peak lab und des Enera-Projekts als wichtige Schlüsselpartner heraus. Diese sind teilweise auch maßgeblich an dem Zugang zu Schlüsselprodukten verantwortlich, wie Expertenwissen, Kontakten, Daten oder Hardwarekomponenten. Des Weiteren werden nicht nur Privatkunden, sondern auch Energieversorger, Netzbetreiber und externe

Dienstleister wie Gerätehersteller als Kundenarten identifiziert. So kann die Disaggregation von Lastkurven einen Überblick und Optimierungen über den Stromverbrauch bringen und somit verschiedenen Parteien Vorteile bringen sowie die Kommunikation zwischen diesen fördern.

Die Grobarchitektur und Komponenten machen deutlich, dass die Realisierung eines solchen Produktes auf üblichen Systemen lauffähig sein kann. Wichtige hardwareseitige Komponenten stellen ein üblicher Server, der für die Datenverarbeitung und -speicherung zuständig ist, und ein Client zum Abrufen der Visualisierung. Des Weiteren muss ein Smart Meter bzw. ein digitaler Stromzähler mit einer Schnittstelle zur Ausgabe von Daten verfügbar sein, der mit dem Server kommunizieren kann. Die nötigen softwareseitigen Komponenten lassen sich auf üblichen Betriebssystemen realisieren und stellen somit keinen großen Aufwand dar.

5.4. Umsetzung / Implementierung

In diesem Abschnitt wird die Implementierung der im Abschnitt Konzept definierten Komponenten beschrieben. Für jede Komponente wird die jeweilige technische Umsetzung in Bezug auf die gestellten Anforderungen erläutert. Der Fokus liegt auf der Erläuterung technischer Entscheidungen und der Lösung aufgetretener Problemstellungen.

5.4.1. Gesamtarchitektur

In den Anforderungen wurden verschiedene Komponenten des Prototyps und deren Aufgaben definiert. Setzt man diese Komponenten zusammen ergibt sich eine übergeordnete Gesamtarchitektur, die im Komponentendiagramm 41 zu sehen ist. In den folgenden Abschnitten wird auf Umsetzung und die Funktionalitäten der einzelnen Komponenten genauer eingegangen. Nach einem kurzen Abschnitt zum allgemeinen Vorgehen bei der Entwicklung des Prototyps, werden nacheinander die einzelnen Komponenten beschrieben. Die Reihenfolge der Komponenten mit der dazugehörigen Nummer ist: 01 Messstation (Appliance-Simulator); 02 Server; 03 Preprozessor; 04 Datenbank; 05 Converter; 06 Disaggregation; 07 Webservice; 08 Visualisierung.

Vorgehen Ein wichtiger Bestandteil der Implementierung des Prototyps ist das Vorgehen im Entwicklungsprozess. Zunächst sollen erstmal die grundlegende Arbeitsweise und die Strukturierung des Projektes erläutert werden.

Der Prototyp besteht aus verschiedenen Hard- und Softwarekomponenten, die zum Teil auch durch die Partner der Projektgruppe bereitgestellt wurden. Auf der anderen Seite

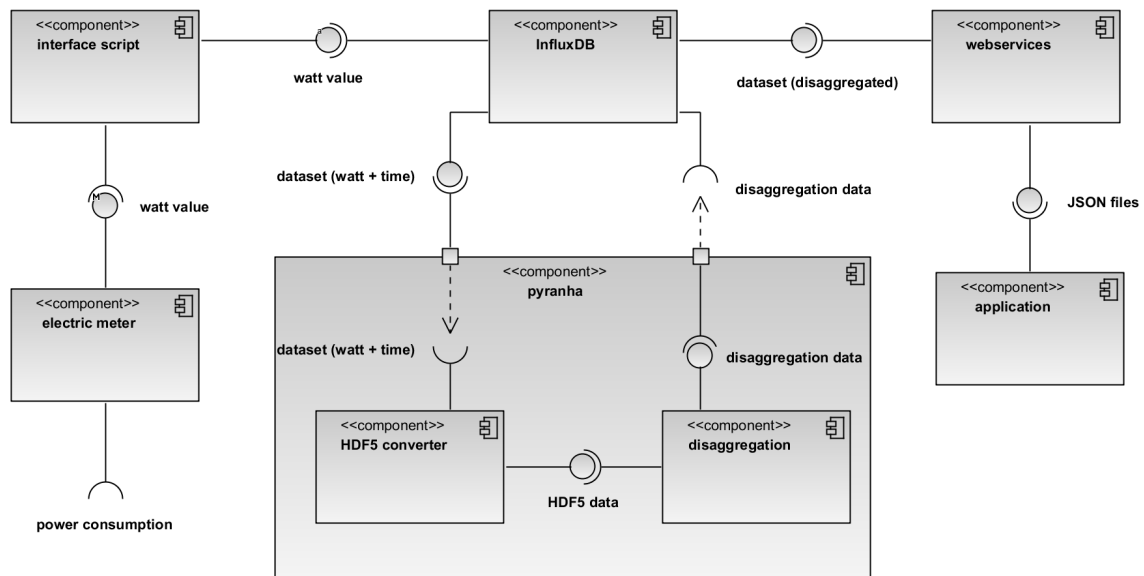


Abbildung 41: Komponentendiagramm Pyranha [Eigene Quelle]

sind viele Softwarebestandteile und Skripte grundlegend neu und für den Einsatz im Prototypen entwickelt worden. Alle Softwarekomponenten sind mit Hilfe der interpretierten Programmiersprache Python umgesetzt. Die Entscheidung für diese Sprache basiert auf verschiedenen Gründen: Zum einen ist die Programmiersprache kostenlos, leicht zu erlernen und universell einsetzbar. Zweitens hat Python sich in den letzten Jahren als eine der wichtigsten Sprachen im Bereich Data Science herausgestellt, da sie viele umfangreiche Module und Bibliotheken bietet. Diese Bibliotheken sind auch in der Umsetzung des Prototyps vielfach eingesetzt worden. Der dritte Grund ist die Kompatibilität zu anderer verwendeter Software. Einige Teile basieren auf Implementierungen aus verschiedenen Open-Source Projekten. Da diese auch in Python umgesetzt sind, ist auch dafür der Einsatz sinnvoll. Python kommt hier in der Version 2.7 zum Einsatz, da einige Module die für Disaggregation noch nicht mit einer höheren Version kompatibel sind.

Alle entwickelten Softwarekomponenten sind in einem Projekt untergebracht. Um eine Versionsverwaltung zu nutzen und am Code mit mehreren Leuten zu arbeiten, wurde dieses Projekt mit Hilfe der verteilten Versionsverwaltung Git⁵ entwickelt. Der entsprechende Server für die Nutzung wurde von der Universität zur Verfügung gestellt. Neben den konventionellen Vorteilen einer verteilten Versionsverwaltung, bietet der Einsatz von Git einen weiteren wichtigen Vorteil für die Entwicklung. Die entwickelten Softwaremodule werden

⁵ <https://git-scm.com>

auf einem externen Rechner ausgeführt, der mit einem Stromzähler verbunden und nicht ohne Aufwand zugänglich ist. Damit der Code synchron gehalten werden kann, wurde dort ein Git-Client installiert. Bei Änderungen kann der aktuellste Stand vom Git-Server heruntergeladen werden.

Das Projekt läuft unter dem internen Arbeitstitel Pyranha. Die Auswahl dieses Namens wurde von der Teilgruppe selbst ausgewählt und folgt einem inoffiziellen Muster für die Benennung von Projekten, die mit Python umgesetzt wurden. Der Name basiert auf dem tierischen Vorbild des Piranhas und stellt eine Metapher für die Verarbeitung von großen Mengen (in diesem Fall Daten) durch ein kleines Lebewesen dar. Das Ersetzen des Buchstaben y stellt eine Verbindung zum Einsatz der Programmiersprache Python her.

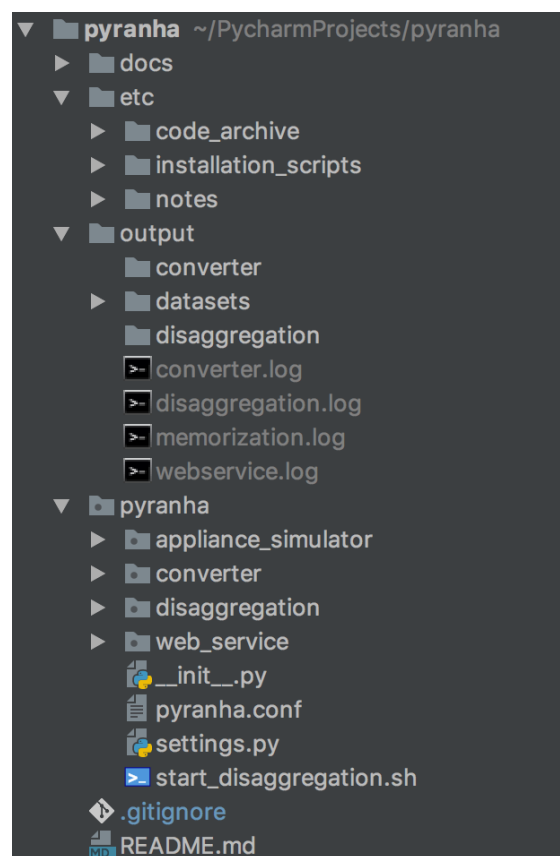


Abbildung 42: Datenstruktur für den Prototyp [Eigene Quelle]

In der Abbildung 42 ist die Datenstruktur aller Softwarekomponenten zu sehen, die zum Projekt gehören. Grundsätzlich ist die Software nach den vorhandenen Softwarekomponenten unterteilt:

- `pyranha.docs`: Enthält verschiedene Dateien zum Thema Dokumentation, aber auch teilweise Anleitungen zu verschiedenen Modulen
- `pyranha.etc`: Dieser Ordner dient der Sammlung sonstiger Sachen, darunter fall alte Code-Snippets, Installationskripte aber auch grundsächliche Notizen
- `pyranha.output`: Dies ist der Ausgabeordner für das gesamte Projekt, hier lagern verschiedene Ergebnisdokumente oder auch konvertierte Dateien. Des Weiteren sind hier auch die Logdateien für alle Module zu finden
- `pyranha.pyranha`: In diesem Ordner befinden sich alle Bestandteile des Codes des Projektes (Unterteilt nach den einzelnen Komponenten)
- `pyranha.pyranha.appliance_simulator`: Code zum softwaretechnischen Simulieren von elektrischen Verbrauchern
- `pyranha.pyranha.converter`: Module zum Konvertieren von Daten aus der Influx-Datenbank in ein HDF-Store
- `pyranha.pyranha.disaggregation`: Module zum Ausführen der Disaggregation auf konvertierten Datensätzen
- `pyranha.pyranha.web_service`: Module zur Bereitstellung des REST-Schnittstelle für die Verbrauchsdaten zur Darstellung auf einem Endgerät

In den kommenden Abschnitten wird die Funktionsweise der entwickelten Module genau erläutert. Dazu wird auch noch auf die Einordnung in das Gesamtkonzept eingegangen und das Zusammenspiel mit den anderen Modulen beschrieben. Grundsätzlich greifen die Module auf die Ergebnisse anderer Module zu oder nutzen aktiv Services anderer Module.

5.4.2. Komponente 01: Messstation

Damit Verbrauchsdaten von elektrischen Verbrauchern gesammelt werden können, muss dafür eine geeignete Vorrichtung vorhanden sein. Für den Prototyp kommt ein umgebauter digitaler Stromzähler zum Einsatz.

Da es leider keine Möglichkeit gab aggregierte Lastkurven eines realen Haushaltes in der richtigen Auflösung zu erhalten, wurde ein digitaler Stromzähler zur Nutzung an Steckdosen verwendet. Je nachdem, ob ein oder mehrere Geräte an dem Stromzähler angeschlossen sind, konnten Lastkurven einzelner Verbraucher oder aggregierte Lastkurven von mehreren Verbrauchern aufgezeichnet werden. Dieser Aufbau konnte also zum Erstellen eigener Test-

und Trainingsdaten genutzt werden. Es wurde so eine Simulationsumgebung für den Prototypen erstellt, der den Eigenschaften eines Stromzählers in einem Haushalt gleichkommt.

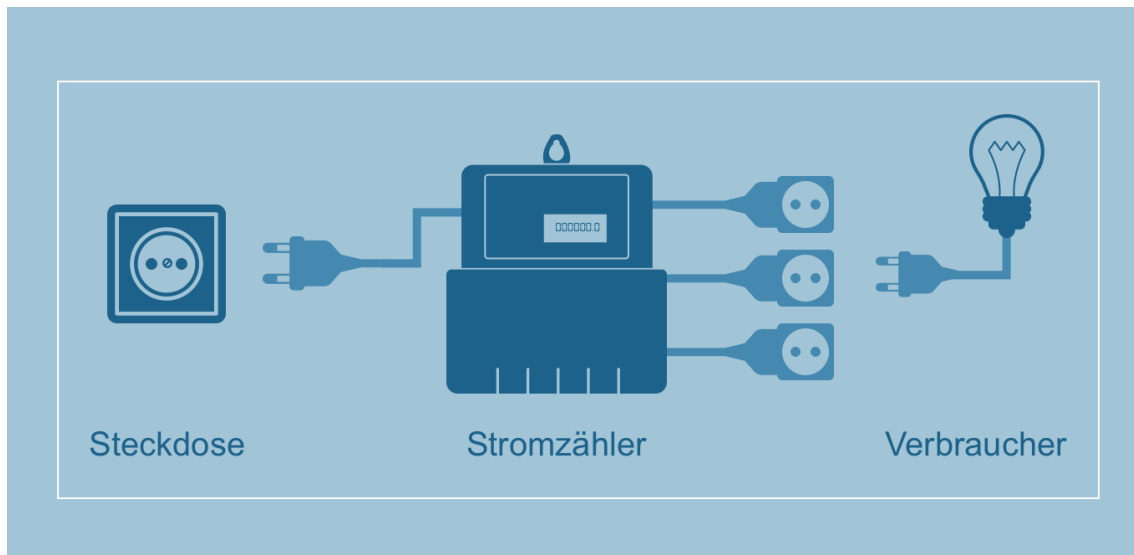


Abbildung 43: Aufbau Messtation [Eigene Quelle]

Das Konzept dieses umgebauten Stromzählers funktioniert nach einem simplen Prinzip und wird in der Abbildung 43 verdeutlicht. Der Stromzähler selbst besitzt ein eigenes Netzteil, der mit einer Steckdose verbunden werden kann. Den Strom leitet der Stromzähler weiter an drei eigene Steckdosen, die direkt an diesem befestigt sind. An diesen Steckdosen können dann elektrische Verbraucher wie an einer normalen Steckdose angeschlossen und benutzt werden. Damit ist der Stromzähler aufgebaut wie ein einfacher Doppelstecker, nur dass er die Möglichkeit bietet den verbrauchten Strom aufzuzeichnen. Jede einzelne dieser Steckdosen ist an eine eigene Phase des Stromzählers angeschlossen, was es ermöglicht zur gleichen Zeit aggregierte Verbräuche und die Verbräuche der einzelnen Geräte gleichzeitig aufzuzeichnen.

Ein wichtiges Merkmal ist die optische Schnittstelle des digitalen Stromzählers, über den die Verbrauchsdaten aufgezeichnet werden können. Zum Auslesen der Daten wurde durch das techlab eine entsprechende Platine gefertigt und mit dem Stromzähler zusammen bereitgestellt. Diese Platine ermöglicht es die Daten auszulesen und dieses Signal an einen angeschlossenen Raspberry Pi weiter zu leiten. Des Weiteren ist diese Platine auch auf die Passform des Stromzählers zugeschnitten und besitzt mehrere LEDs, die als Indikatoren für den Zustand des Raspberry Pis dienen.

Ein wichtiger Faktor in Bezug auf die Disaggregation, spielt die Frequenz mit der die

Leistungsaufnahme bei elektrischen Verbrauchern aufgezeichnet wird bzw. welche Frequenz ausgelesen werden kann. Generell ist zu sagen, dass je höher die Frequenz der Daten ist, desto besser sind die Möglichkeiten für die Disaggregation. Von der optischen Schnittstelle lassen sich die Daten mit einer Frequenz von 0,5 Hertz auslesen.

Appliance-Simulator Bei der Arbeit mit dem gesamten Prototyp ist es an vielen Stellen erforderlich Testszenarien aufzubauen. In den Szenarien sollen Daten von verschiedenen Verbrauchern aufgezeichnet werden, die zu einer definierten Zeit aktiviert waren. Versucht man Testszenarien durch das Anschließen und Aktivieren physischer Verbraucher zu erreichen, ist nicht praktikabel. Aus diesem Grund wurde der Appliance-Simulator entwickelt.

Der entwickelte Appliance-Simulator hat die Aufgabe den Verbrauch von verschiedenen elektronischen Geräten softwaretechnisch zu simulieren. Das Ziel dahinter ist es, das Aktivieren verschiedener Geräte nachzubilden, ohne diese Verbraucher wirklich an den Stromzähler anschließen zu müssen. Da Verbrauchsdaten einzelner Verbraucher im Laufe des Projekts bereits aufgezeichnet wurden, bietet es sich an diese Daten auch zum Simulieren zu verwenden.

Werden beispielsweise Verbrauchsdaten einer Spülmaschine aufgezeichnet, können diese in ein HDF-Store konvertiert werden. Dieser Vorgang wird im dedizierten Abschnitt über den Converter genauer erläutert. Es liegt für einen Lauf einer Spülmaschine ein Datensatz mit Timestamps und entsprechenden Wattzahlen vor. Dieser Datensatz kann dem Appliance-Simulator übergeben werden. Der Simulator entfernt die Timestamps und kann die aufgezeichneten Wattzahlen wieder in eine Datenbank schreiben. Das ist dieselbe Datenbank, in der auch die reale Messstation Daten ablegt.

Damit mit diesem Appliance-Simulator flexibel und auf die Disaggregation abgestimmt gearbeitet werden kann, werden noch weitere Funktionen angeboten. Es besteht die Möglichkeit Geräte zu einem bestimmten Zeitpunkt laufen zu lassen, also auch rückwirkend für Zeiträume in der Vergangenheit oder aber auch in der Zukunft. Des Weiteren können die Verbrauchsdaten mehrerer Geräte gleichzeitig simuliert werden. Überschneiden sich an einem Zeitpunkt die Wattwerte, werden diese aggregiert und bilden die Ausgangslage, für das Durchführen einer Disaggregation.

Zum Testen verschiedener Services, die keine echten Verbrauchszahlen von Geräten benötigen, können auch kontinuierlich zufälligen Verbrauchsdaten erzeugt werden. Der Appliance-Simulator bietet auch an, bereits aufgenommene Verbrauchszahlen wieder aus der Datenbank zu löschen.

5.4.3. Komponente 02: Server

Wie in den Anforderungen beschrieben wurde, stellt der Server eine zentrale und übergeordnete Komponente dar, die verschiedene Funktionen erfüllt. Durch die variablen Funktionen muss die ausgewählte Hardware verschiedenste Anforderungen erfüllen. Im folgenden Abschnitt werden die Funktionen einzelner Komponenten genauer erläutert, dazu gehört auch die Begründung zur Auswahl entsprechender Hard- und Software.

Vorab ist festzuhalten, dass die Aufgabe des Servers durch einen Einplatinencomputer Raspberry Pi Model 3 des Unternehmens Raspberry Pi Foundation⁶ erfüllt wird. In welcher Art der Raspberry Pi die gestellten Anforderungen erfüllt, wird in den einzelnen Abschnitten genau beleuchtet. Grundsätzlich ist der Raspberry Pi ein kleiner und mobiler Rechner, der etwa die Größe einer Kreditkarte besitzt. Dies macht den Rechner universell einsetzbar, da dieser auch für dauerhaften Betrieb geeignet ist und nur einen geringen Stromverbrauch hat. Das Betriebssystem ist eine Linux-Distribution, die auf Debian basiert und für den Raspberry Pi optimiert ist. Dies ist ein gängiges und stabiles Betriebssystem für den Rechner.

Zudem besitzt der Raspberry Pi eine große Auswahl an diversen Anschlüssen, darunter einen Ethernet-Anschluss. Über diesen Netzwerkzugang ist es durch verschiedene Netzwerkprotokolle möglich auf den Raspberry Pi zuzugreifen.

5.4.4. Komponente 03: Preprozessor

Im letzten Abschnitt wurde erläutert, wie der umgebaute digitale Stromzähler die Verbrauchsdaten elektronischer Geräte aufzeichnen kann. Über die optische Schnittstelle können diese Daten ausgelesen werden. Mit Hilfe von GPIO-Pins ist es möglich die Platine an den Raspberry Pi anzuschließen, um damit die Verbrauchsdaten auszulesen. Die Kombination zwischen dem Stromzähler, der speziell entwickelten Ausleseplatine und dem Raspberry Pi wurde zuvor als Prototyp durch das techlab genutzt. Ein weiterer Grund für den Einsatz des Raspberry Pi ist also die sichere Kompatibilität zur Hardware der Messstation.

Die nächste Anforderung an den Raspberry Pi ist es, die Daten von Messstation kontinuierlich entgegenzunehmen. Die Aufgabe, den ankommenden Datenstrom entgegenzunehmen und zu parsen, übernimmt ein C++-Programm, welches kontinuierlich als Systemjob gestartet wurde. Dieses C++-Programm konnte zusammen mit dem Stromzähler und der Platine vom techlab übernommen werden.

⁶ <https://www.raspberrypi.org/>

5.4.5. Komponente 04: Datenbank

Im nächsten Schritt sollen die Daten, die vom Datenstrom der Messstation geliefert werden, in einer Datenbank abgelegt werden. Die relevanten Daten die gesammelt werden, bestehen lediglich aus einem einzelnen Timestamp und der Leistungsaufnahme, die in diesem Moment in Watt gemessen wird. Diese Leistungsaufnahme in Watt ist auf die drei verschiedenen Phasen des Stromzählers aufgeteilt und in einem vierten Wert aggregiert.

Um diese Werte abzulegen, kommen verschiedene Datenbanken in Frage. Da es sich bei den Daten um Zeitreihendaten handelt und da auch der Timestamp eine übergeordnete Rolle spielt, wurde die eine Influx-Datenbank⁷ verwendet. Die Influx-Datenbank ist eine open-source Datenbank für die Speicherung von Zeitreihendaten. Diese Datenbank ist plattformübergreifend nutzbar und auch der Einsatz auf einem Raspberry Pi ist vorgesehen. Die Influx-Datenbank bietet eine simple HTTP-Schnittstelle an, um Daten in die Datenbank zu schreiben. Die Voraussetzung dafür ist aber, dass bereits eine Datenbank und eine entsprechende Tabelle angelegt wurde. Abstrahiert kann das Schema der Influx-Datenbank als einfache Tabelle angesehen werden, diese wird beispielhaft in der Tabelle 4 dargestellt.

time	L1	L2	L3	LSum	meterid
1502636833299171072	0	0	2048.47	2048.47	0273160104447
1502636837299171072	0	0	1983.99	1983.99	0273160104447
1502636839299171072	0	0	2041.66	2041.66	0273160104447
1502636841299171072	0	0	2113.16	2113.16	0273160104447

Tabelle 4: Datenbankauszug für die Messdaten

Die Daten werden in eine Datenbank mit dem Namen meterdata geschrieben. In der Tabelle 4 sind die relevanten Daten aufgezeigt. In dieser Datenbank ist der Timestamp der Schlüssel für den entsprechenden Datensatz. Dieser Schlüssel wird in der Tabelle in der Spalte time abgelegt und repräsentiert den Timestamp, wann die Leistung die die Messstation aufgenommen wurde. Entsprechend zum Timestamp spiegeln die Werte L1, L2, L3 und LSum die Leistung in Watt wieder. Die Spalte LSum ist nur eine Aggregation der drei Phasen des Stromzählers. In der Tabelle kann man sehen, dass alle Werte in den Spalten L1 und L2 null sind, dies impliziert das nur ein Verbraucher an der Phase L3 angeschlossen ist. Zu sehen ist, dass der elektrische Verbraucher der an Phase 3 angeschlossen ist, etwa um die 2000 Watt verbraucht.

Da Timestamps alleine keinen eindeutigen Schlüssel darstellen, besitzt die Influx-Datenbank

⁷ <https://www.influxdata.com/>

das Konzept von Tagsets. Geht man also davon aus, dass man die Verbrauchsdaten mehrerer Messstationen vereint, würden sich die Timestamps überschneiden. Die Tag-Sets sind Informationen, die den einzelnen Datensatz genauer definieren und unterscheidbar machen. Damit Daten aus mehreren Stromzählern unterschieden werden können, ist die ID des Stromzählers, die identifizierende Eigenschaft.

5.4.6. Komponente 05: Converter

Durch das Eintragen der Verbrauchsdaten in die Datenbank liegen alle Vergangenheitswerte in einer generischen, formatierten Form vor. Da die Daten für die Verarbeitung im Disaggregationsprozess in ein weiteres Format umgewandelt werden müssen, findet ein Converter Anwendung. Dieser Converter hat die Aufgabe bestimmte Verbrauchsdaten aus der Influx-Datenbank auszulesen und diese mit weiteren Metadaten zu verknüpfen. Zum Einsatz kommt diese Komponente nicht nur bei der Disaggregation, sondern auch bei der Berechnung der Daten für den Webservice. Der Converter ist ein selbst-entwickelter Softwarebaustein, der in der Programmiersprache Python entwickelt wurde. Dieser besteht aus verschiedenen Modulen, die verschiedene Schnittstellen bietet, um Verbrauchsdaten zu konvertieren. Dieser Vorgang ist in drei wichtige Phasen aufgeteilt:

1. Verbrauchsdaten werden über die Schnittstelle der Influx-Datenbank abgefragt und in ein passendes Format umgewandelt
2. Verbrauchsdaten werden mit relevanten Metadaten verknüpft
3. Verknüpfte Daten werden in ein HDF-Store geschrieben

Das Hauptmodul `influx_converter.py` verwaltet mehrere Methoden als Schnittstellen, um Daten zu konvertieren. Diese unterscheiden sich hauptsächlich durch die Art des Zeitfensters, welches abgefragt wird. In der Tabelle 5 sind diese verschiedenen Methoden aufgeführt.

Die erste Methode `convert_last_x_data(...)` der Klasse `InfluxDBConverter`, hat die Aufgabe Daten zu konvertieren, die eine gewisse Zeit vom aktuellen Zeitpunkt zurückliegen. Dieser Zeitraum kann mehrere Stunden, Tage aber auch Wochen und Monate betragen. Diese Funktionalität wird bei der Erstellung von Daten für die Disaggregation benötigt, denn für die Erkennung der Geräte werden immer wieder die Daten gebraucht, die kurz zuvor über einen bestimmten Zeitraum aufgenommen wurden. Die zweite Methode `convert_data_from_x_to_y(...)` ermöglicht es die aufgezeichneten Verbrauchsdaten auszulesen und zu konvertieren, die zwischen zwei Timestamps aufgenommen wurden. Diese Funk-

Methoden	Aufgabe
<code>convert_last_x_data(...)</code>	Konvertiert Daten, die eine gewisse Zeitspanne vom aktuellen Zeitpunkt in der Vergangenheit liegen. Zum Beispiel alle aufgezeichneten Lastkurven aus den letzten drei Stunden.
<code>convert_data_from_x_to_y(...)</code>	Konvertiert alle Daten, die zwischen zwei Timestamps liegen
<code>convert_data_for_specific_day(...)</code>	Konvertiert alle Daten, die an einem bestimmten Tag gesammelt wurden

Tabelle 5: Services des Converters

tionalität spielt eine besondere Rolle dabei, wenn Daten eines einzelnen Gerätes aufgezeichnet wurden und diese mit den entsprechenden Metadaten verknüpft werden sollen. Die dritte Methode macht es möglich Verbrauchsdaten für einen ganz bestimmten Tag zu konvertieren. Dies ist jedoch nur eine spezialisierte Form von `convert_data_from_x_to_y(...)`, welche die Timestamps auf den Anfang und das Ende des Tages legt.

In einem vorherigen Abschnitt wurde bereits angedeutet, wie das Konvertieren von Verbrauchsdaten funktioniert. Die drei verschiedenen Methoden durchlaufen dabei die gleichen Schritte, unterscheiden sich jedoch bei der Art der Zeitfenster.

Schritt 1: Verbrauchsdaten abfragen Im ersten Schritt geht es darum Daten aus der Datenbank abzufragen, die durch die Messstation aufgezeichnet wurden. Im Anschluss werden diese in ein Format konvertiert, mit dem weitergearbeitet werden kann. Analog zum Schreiben der Daten, besitzt die Influx-Datenbank eine HTTP-Schnittstelle, um die Daten aus der Datenbank zu lesen. Zur Umsetzung der Anfragen an die Schnittstelle gibt es ein Modul mit dem Namen `influxdb`. Dies ermöglicht es die Anfragen direkt aus der Python Umgebung an die Schnittstelle zu schicken und die Ergebnisse direkt weiterzuverarbeiten.

Eine Methode stellt eine Verbindung zur lokalen Influx-Datenbank her. Die Zugangsinformationen sind in einer externen Konfigurationsdatei abgelegt und werden bei der Herstellung der Verbindung neu geladen. Dies hat den Vorteil, dass diese Parameter einfach angepasst werden können, ohne in den eigentlichen Code eingreifen zu müssen. Nachdem die Verbindung hergestellt ist, wird eine entsprechende Anfrage an die Datenbank gestellt. Dabei werden alle Verbrauchsdaten, also Timestamp und gemessene Leistungsaufnahme, aus der Datenbank `meterdata` und der Tabelle `power` abgefragt. Diese Werte werden aber noch über den entsprechenden Zeitraum gefiltert, der über die Anfrage definiert wurde. Als Ergebnis wird eine Instanz der Klasse `ResultSet` geliefert, welche die angefragten Daten

beinhaltet. Mit diesem Objekt lassen sich die Daten nicht trivial verarbeiten, daher müssen die Daten in ein stärker generisches Format umgewandelt werden.

Die Daten werden mit Hilfe der Methode `convert_resultset_to_dataframe(...)` in die Form einer Tabelle konvertiert. Die Tabelle ist ein Objekt der Klasse `DataFrame` aus dem Modul `pandas`. Dieses Modul ist ein Toolkit, das auf die Verarbeitung und Analyse von Daten spezialisiert ist. In der Tabelle 6 ist der Aufbau dieser erzeugten Tabelle abgebildet.

id	physical_quantity: power powertype: active
2017-08-13 10:50:05.299171+02:00	3.23
2017-08-13 10:50:07.299171+02:00	2.44
2017-08-13 10:50:09.299171+02:00	2.32

Tabelle 6: Aufbau DataFrame konvertierter Daten

Zu sehen ist, dass die Tabelle hier über die Timestamps indiziert wird, welche auch in der ersten Spalte abgebildet sind. Die zweite Spalte spiegelt die gemessene Leistung wieder. Dafür wurde im Kopf der Spalte, die Art der physikalischen Größe definiert.

Schritt 2: Verknüpfung mit Metadaten Die Daten aus der Tabelle sind bisher mit keinerlei Kontextinformationen außer der Zeit verknüpft. Es ist also nicht zu erkennen, ob die gemessenen Werte aggregierte Verbrauchsprofile oder Verbrauchsprofile einzelner Geräte sind. Um die konvertierten Datensätze als Trainingsdaten oder auch als Datensatz zum Disaggregieren verwenden zu können, müssen die aktuellen Daten mit den Metadaten verknüpft werden.

Um die Datensätze und die Metadaten zu verknüpfen, werden die Daten in einem Hierarchical Data Format⁸ kurz HDF abgelegt. Dieses Format wurde ausgewählt, da es dafür ausgelegt ist auch in einer einzigen Datei große Datensätze zu speichern. Des Weiteren können neben den Datensätzen auch weitere Informationen wie die Metadaten abgelegt werden. Dazu kommt noch, dass die Vorgehensweise für die Disaggregation des NILM-Toolkit auch mit HDF-Daten arbeitet. Da auch im Laufe des Prototyps mit diesem Toolkit gearbeitet wird, ist es sinnvoll dieses Datenformat analog zu verwenden, da dies viel Konvertierungsarbeiten erspart. Intern werden konvertierte Daten auch auf das Datenmodell angepasst, damit diese in Verbindung mit dem NILM-Toolkit funktionieren. Dies definiert auch die Struktur der Metadaten, die mit den eigentlichen Messdaten verknüpft werden.

⁸ <https://www.hdfgroup.org/>

Die Metadaten befinden sich in einem Unterordner `metadata` des Converters und sind in YAML-Dateien abgelegt. Dies ist eine Auszeichnungssprache, die an XML angelehnt ist. Neben dieser Auszeichnungssprachen wären auch noch viele andere Dateiformate wie `Json` oder `XML` in Frage gekommen, jedoch ist das `YAML`-format sehr simpel und es kann ohne viel Aufwand direkt in der Textdatei bearbeitet werden.

An der obersten Stelle der Hierarchie steht der Datensatz, der eine umfassende Ebene für alle Daten bildet. In der `dataset.yaml` Datei finden sich wichtige Informationen zu Datensatz allgemein. Dazu gehört wer diese Datensätze aufgezeichnet hat, wo sie aufgenommen wurden und wann sie aufgenommen wurden. Diese Informationen bieten also die Möglichkeit alle Daten in einem gewissen Kontext einzuordnen. Ein Datensatz besteht aus verschiedenen Gebäuden, denen die eigentlichen Messsysteme untergeordnet sind. Zu den Informationen gehören Name und Instanznummer des Gebäudes, um die Daten verschiedener Gebäude unterscheidbar zu machen. Des Weiteren werden unter dem Schlüssel `elec_meters`, die Messsysteme eingetragen, die in diesem Haus genutzt werden. Da es sich bei diesem Prototyp auf einen Stromzähler beschränkt, findet sich in dieser Liste nur ein einziger Eintrag. Abschließend sind unter dem Schlüssel `appliances` alle Geräte aufgelistet, die in diesem Gebäude zu finden sind. Für die einzelnen Geräte werden Informationen wie Name, Gerätetyp, Instanznummer und weitere spezifische Informationen definiert. Die Datei `meter_devices.yaml` führt weitere spezialisierte Informationen auf, die den genutzten Stromzähler betreffen. Dieser Stromzähler wurde auch schon in den Metadaten zum Gebäude referenziert.

Damit die Tabelle mit den Verbrauchsdaten passend abgelegt werden kann, wird es an das Datenschema des NILM-Toolkits angepasst. Dieser Datensatz wird einer Messstation zugeordnet, die daraufhin einem Gebäude zugeordnet wird, welches zu einem Datensatz gehört. Innerhalb eines `HDF`-Stores werden die Daten mit Schlüsseln abgelegt. Der Aufbau dieser Schlüssel hat Analogien mit dem Aufbau von `Directories` in einem `Filesystem`, so wäre ein Beispiel `‘/building1/elec/meter1` der Schlüssel eines einzigen Messsystems. Die Datensätze aus der Tabelle werden unter dem Schlüssel gespeichert und die einzelnen speziellen Informationen werden auf der entsprechenden Ebene abgelegt.

Um eigene Trainingsdaten erstellen, ist es erforderlich Datensätze mit bestimmten Geräten mappen zu können. Das bedeutet, aufgezeichnete Verbrauchsdaten, die von einem einzelnen Gerät verursacht wurden, auch diesem zuweisen zu können. Konvertiert man die Daten von einem bestimmten Zeitraum und weiß, dass zu dieser Zeit nur ein bestimmtes Gerät gelaufen ist, kann dies als Parameter übergeben werden. Nutzt man eine der oben angegebenen drei Konvertierungsmethoden, ist es möglich die Art des Gerätes als `Key-Word-Argument` mit zu übergeben. Ein Methodenaufruf könnte folgendermaßen aussehen:

`convert_last_x_data(..., appliance="dish_washer")`). Dabei werden die Verbrauchsdaten der Spülmaschine zugewiesen, die Voraussetzung ist aber, dass die Metadaten für dieses Gerät in der `building1.yaml`-Datei eingetragen sind. Es kommt also darauf an, dass der Parameter und der Wert `original_name` übereinstimmen. Beabsichtigt man nun Daten undefiniert zu lassen, wird kein Parameter `appliance` im Methodenaufruf definiert. Standardmäßig werden dann die Daten mit dem Gerät `unknown_measurements` verknüpft. Die Daten gelten also als nicht klassifiziert und stellen damit einfach unbekannte oder aggregierte Verbrauchsdaten dar.

Schritt 3: Speichern des HDF-Stores Unabhängig davon, ob die Verbrauchsdaten einem Gerät zugewiesen wurden oder nicht, werden diese im letzten Schritt in einer einzigen Datei gespeichert und an einem definierten Ort abgelegt. Diese Dateiablage definiert somit eine Schnittstelle mit verschiedenen HDF-Stores, die von anderen Services genutzt werden kann.

5.4.7. Komponente 06: Disaggregation

Die Speicherung und Konvertierung der Verbrauchsdaten bildet eine Basis für die Durchführung der Disaggregation der Verbrauchskurven. Da jedoch die Implementierung eines eigenen Ansatzes zur Disaggregation sehr aufwendig ist, liegt im Rahmen dieses Prototyps darauf kein Fokus. Um jedoch auf einen bestehenden Ansatz zurückgreifen zu können, wird die Open-Source Implementierung des NILM-Toolkits⁹ für eigene Zwecke angepasst. Das NILM-Toolkit ist eine Sammlung von verschiedenen Softwaretools, um die Genauigkeit eines selbst-entwickelten Disaggregationsansatzes evaluieren zu können. Des Weiteren besteht die Möglichkeit den Ansatz mit anderen bestehenden Techniken zu vergleichen, daher umfasst das Toolkit auch Implementierungen bestehender Disaggregationsalgorithmen. Eine dieser Implementierungen wird für den Prototyp so angepasst, dass die Disaggregation durchgeführt werden kann. Neben diesen Implementierungen besitzt das Toolkit Parser für bestehende Datensätze, verschiedene statistische Methoden zur Beschreibung von Datensätzen und eine Menge von Metriken zur Messung der Genauigkeit. Das NILM-Toolkit baut auf dem NILM-Metadata-Framework¹⁰ auf. Dies ist ein Schema für die Daten rund um das Thema Disaggregation, welches von den gleichen Entwicklern wie von dem NILM-Toolkit stammt. Im Rahmen des Prototyps wird auch auf die Verwendung zurückgegriffen, um die Kompatibilität zum NILM-Toolkit sicherzustellen.

⁹ <https://github.com/nilmk/nilmk>

¹⁰ https://github.com/nilmk/nilm_metadata

Funktionsweise NILM-Toolkit Bevor die Anpassung des NILM-Toolkits erläutert werden kann, wird im folgenden Text die grundlegende Funktionsweise erläutert. Das ist notwendig, da auch das Vorgehensmodell des Prototyps an diese Funktionsweise angepasst ist.

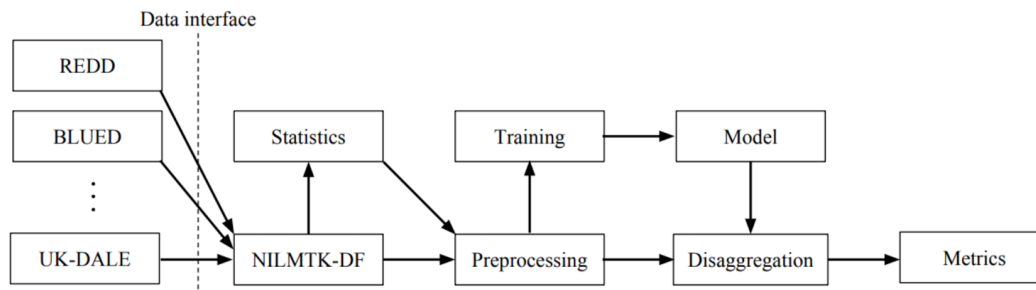


Abbildung 44: Ablauf NILM-Toolkit [Nip14]

Die Abbildung 44 visualisiert das Vorgehen bei Einsatz des NILM-Toolkits. Die Grundlage für eine Disaggregation bilden immer Test- und Trainingsdaten, also Datensätze, die zum einen aggregierte Verbrauchskurve von Haushalten besitzen, an denen die Disaggregation durchgeführt werden kann. Um das Ergebnis evaluieren zu können, werden passend dazu die Verbrauchsdaten der einzelnen Geräte, die aktiviert waren gebraucht. Diese Art von Daten ist in einigen öffentlichen Datensätzen frei verfügbar. Dazu gehört unter anderem der REDD-Datensatz¹¹ aus den USA, welcher im Jahr 2011 aufgezeichnet wurde und Daten von sechs Haushalten mit einer Auflösung von 3 Hertz. Um mit diesem und anderen Datensätzen arbeiten zu können besitzt das NILM-Toolkit entsprechende Parser für mehrere öffentlich verfügbare Datensätze. Die Aufgabe dieser Parser ist es die Daten in das vorgegebene Datenmodell zu überführen und mit den entsprechenden Metadaten zu verbinden. Des Weiteren wird auch mit Hilfe von NILM-Metadaten die Namensgebung der verschiedenen Datensätze vereinheitlicht. Dies entspricht dem Vorgehen des Converters dieses Prototyps, welches vorher bereits erläutert wurde.

Überführt werden die Datensätze in ein Format, das hier als NILMTK-DF bezeichnet wird. Dabei handelt es sich um ein HDF-Store, welcher auch schon im Rahmen des Converters genauer erläutert wurde. Ausgehend von diesem Datenformat bietet das Toolkit zum einen verschiedene Methoden aus der Statistik an, um die Daten zu untersuchen. Zum anderen lassen sich die Daten durch verschiedene Methoden im Paket Preprocessing zur Wiederverwendung vorbereiten. Darunter fällt auch das Herunterrechnen vorhandener Leistungsdaten auf eine geringere Frequenz.

¹¹ <http://redd.csail.mit.edu/>

Der erste Schritt der Disaggregation sieht vor, mit den vorhandenen Verbrauchsdaten der Geräte ein Modell zu trainieren. Die Lastprofile jener Geräte die später erkannt werden sollen, werden dem Algorithmus antrainiert. Der zweite Schritt ist dann das Erkennen der Geräte aus dem aggregierten Lastprofil, die vorher antrainiert wurden. Dafür besitzt das Toolkit die Implementierung der Algorithmen mit dem Namen *Factorial Hidden Markov Model* und *Combinatorial Optimisation*.

Das Ergebnis dieser Algorithmen ist dann eine Prognose, wann die einzelnen Geräte aktiviert waren und wie viel Strom diese verbraucht haben. Das Toolkit bietet im letzten Schritt dann die Funktion an, die Prognosen mit den tatsächlichen Verbräuchen zu vergleichen. Mit verschiedenen Metriken bekommt man so eine Aussage über die Güte des Algorithmus.

Anpassung im Rahmen der Implementierung Die Implementierung des NILM-Toolkits ist ausgerichtet auf die Evaluierung eines Disaggregations-Algorithmus. Damit dies jedoch für die Disaggregation von Verbrauchsdaten aus einem Stromzähler genutzt werden kann, wurde dieses entsprechend angepasst.

Die wichtigsten Schritte bei der Disaggregation sind:

1. Aufbau der Pyranha Datenbank mit Gerätesignaturen
2. Trainieren eines Modells
3. Testdaten aus aktuellem Verbrauch
4. Disaggregation der Testdaten mit Trainingsdaten
5. Schreiben der Ergebnisse
6. Extra: Dauerhafter und kontinuierlicher Einsatz

1. Aufbau der Datenbank Wie auch beim NILM-Toolkit ist eine Datenbank mit Verbrauchsdaten einzelner Geräte die Voraussetzung für die Disaggregation von Lastkurven, die mit einem intelligenten Stromzähler kurz vorher aufgezeichnet werden. Der intelligente Stromzähler selbst liefert die Daten mit einer Frequenz von 0,5 Hertz. Je höher die Verbrauchsdaten aufgezeichnet sind, umso größer ist die Möglichkeit die einzelnen Gerätesignaturen in der Gesamtlastkurve zu unterscheiden. Mit dieser Art von Daten lassen sich nur Geräte mit einem eindeutigen Profil erkennen, wie etwa Waschmaschinen, Trockner

oder auch Geschirrspüler. Generell lassen sich bei dieser Qualität der Daten, nur wenige Geräte (etwa fünf Großverbraucher) erkennen und diese müssen auch mit ausreichend Daten trainiert werden.

Das Training der Daten muss auf Basis der spezifischen Geräte stattfinden, die auch später durch die Disaggregation erkannt werden sollen. Da diese aggregierten Aufzeichnungen aus den Messungen des umgebauten Stromzählers stammen, mussten die Daten auch von den gleichen Geräten aufgezeichnet werden. Wie die Daten von der Messstation aufgezeichnet werden, wurde im entsprechenden Kapitel zur Funktionsweise erläutert. Die Messstation selbst kann nicht erkennen, wenn ein einzelnes Gerät angeschlossen wurde und damit auch nicht automatisch Datensätze erstellen.

Das Nachtragen des Gerätetyps für die Aufzeichnungen muss also manuell vorgenommen werden. Für den Aufbau des Datensatzes wurde ein Haushalt eines Mitglieds der Teilgruppe ausgewählt. Hierfür stand die Aufzeichnung der großen Verbraucher wie Waschmaschine, Geschirrspüler und weiterer kleiner Verbraucher im Vordergrund. Der digitale Stromzähler wurde dabei eine bestimmte Zeit an einen elektrischen Verbraucher angeschlossen, sodass mehrere Aktivierungen aufgezeichnet werden konnten. Diese Zeiträume wurden notiert und mit Hilfe des selbst-entwickelten Converters mit den entsprechenden Geräten verbunden.

Das Konvertieren vieler Durchläufe führte zur Erstellung ebenso vieler HDF-Stores, die einzelne oder mehrere Aktivierungen eines Verbrauchers vereint. Zur Vereinfachung wurden diese einzelnen Teile des Datensatzes zu einem großen HDF-Store zusammengefasst, welches einen großen Datensatz repräsentiert. Dieser Datensatz trägt den Namen `pyranha.h5` und kann mit Hilfe des Moduls `dataset_creator.py` um weitere Teile von Verbrauchsdaten erweitert werden.

2. Trainieren des Modells Mit Hilfe des Datensatzes `pyranha.h5` wird das Modell für den Disaggregations-Algorithmus trainiert. Wie dieses Modell aufgebaut ist und welche Daten hier erzeugt werden, hängt vom Algorithmus selbst ab. Auf diesen Algorithmus wird speziell im Punkt Disaggregation der Testdaten mit Trainingsdaten eingegangen.

Das Trainieren des Modells nimmt je nach Menge der Trainingsdaten eine gewisse Zeit in Anspruch. Sollen jedoch viele Disaggregationsversuche hintereinander laufen macht dieser Vorgang den Prozess ineffizient. Kommen jedoch keine neuen deklarierten Verbrauchsdaten hinzu, ändert sich das Modell nicht. Daher soll das Modell nach der Berechnung gespeichert werden und bei neuer Disaggregation geladen werden. Das Laden dieser Daten erspart dem System eine Menge Rechenzeit.

Das trainierte Modell stellt nach dem Vorgehen des NILM-Toolkit ein Python-Objekt aus dem Datensatz her. Dieses Objekt kann nach der Erstellung serialisiert werden. Das bedeu-

tet die gesamten Werte, die im Objekt enthalten sind, werden persistent auf der Festplatte abgelegt. Diese Aufgabe ermöglicht in Python das Standardmodul Pickle. In dem Modul `serialisation.py` werden zwei Methoden angeboten, welche die Serialisierung und Deserialisierung übernehmen. Sind beispielsweise neue Daten zum Datensatz hinzu gekommen oder liegt noch kein gespeichertes Modell vor, wird ein neues Modell trainiert und mit der Methode `pickle_train_data(...)` serialisiert. Das Modell wird dann unter der Datei `model.save` gespeichert. Wird dann eine neue Disaggregation durchgeführt, deserialisiert die Methode `unpickle_train_data(...)` die gespeicherte Datei und erzeugt ein nutzbares Python Objekt.

3. Testdaten aus aktuellem Verbrauch Das Ziel ist es, die aggregierten Verbrauchsdaten zu analysieren, die kurz vorher vom intelligenten Stromzähler aufgezeichnet wurden. Diese Daten werden hier als Testdaten bezeichnet. Diese Testdaten sind die Daten aus einem definierten Zeitintervall aufgezeichnet wurden, wie etwa die letzten zwei Stunden. Der erstellte Converter bietet die Funktion, die Daten eines solchen Intervalls abzufragen. Vor der Durchführung der Disaggregation werden die nötigen Testdaten geladen und gelten damit auch als eine Art kleiner Datensatz mit unbekanntem Verbrauchern.

4. Die Disaggregation Im vierten Schritt wird die eigentliche Disaggregation der Daten durchgeführt. Wie zuvor schon erwähnt basiert der Vorgang auf der Implementierung aus dem NILM-Toolkit. Das Toolkit bietet zwei etablierte Implementierungen von Disaggregationsalgorithmen an: Das erste ist das Factorial Hidden Markov Model, dies wurde auch anfangs eingesetzt, doch im Verlauf des Projektes traten verschiedene Kompatibilitätsprobleme auf und auch die Serialisierung des trainierten Modells führte zu Problemen. Daher wurde grundsätzlich auf die Nutzung von Combinatorial Optimisation (CO) umgestellt. Dies war ohne viel Aufwand möglich, da an vielen Stellen mit generischen Schnittstellen gearbeitet wird.

Bei der *Combinatorial Optimisation* wird versucht die optimale Kombination der verschiedenen Stati der Geräte zu finden. Dabei wird der Fehler zwischen prognostizierten Energieverbrauch aller Geräte und dem tatsächlichen Verbrauch minimiert. Die Komplexität bei dieser Optimierung steigt exponentiell zu der Anzahl möglicher Geräte und vorhandener Stati der Geräte. Dies führt dazu, dass die Berechnungen auch nur für eine kleine Anzahl von Geräten durchgeführt werden kann [Nip14].

Das Modul `disaggregator.py` ist verantwortlich für die Durchführung der Disaggregation, dabei werden die folgenden Schritte durchlaufen. Die Methode `train_model(...)` trainiert das Modell bzw. deserialisiert das Modell, falls dieses schon vorhanden ist. Die zweite Methode `init_measurement_data(...)` konvertiert die Daten, die durch den intelligenten

Stromzähler in die Datenbank geschrieben wurden. Über die Methode `disaggregate(...)` wird die *Combinatorial Optimisation* mit Hilfe des Modells durchgeführt. Das Ergebnis sind Prognosen für den getesteten Zeitraum, wie viel Strom von den einzelnen Geräten verbraucht wurde.

Aus diesen Daten muss in einem letzten Schritt extrahiert werden, welche Geräte mit einer Plausibilität wirklich aktiviert waren. Für einige Geräte ermittelt der Algorithmus nur eine sehr geringe Prognose für die aufgenommene Leistung. Jedoch der Lauf einer Waschmaschine mit sehr kleinen Verbrauchszahlen weist eher darauf hin, dass dieses Gerät nicht aktiviert war. Eine weitere Aufgabe ist es zu identifizieren, welche Prognosen plausibel anzeigen, dass ein Gerät wirklich gelaufen ist.

Ob ein Gerät wirklich gelaufen ist, wird mit Hilfe der Trainingsdatenbank ermittelt, die verschiedene Vergangenheitswerte zu den einzelnen Geräten gespeichert hat. Die Faktoren die dabei beachtet werden, sind die Dauer wie lange ein Gerät aktiviert war und wie viel Strom in dieser Zeit verbraucht werden. Wird beispielsweise ein Lauf einer Waschmaschine in den letzten beiden Stunden prognostiziert, wird dieser Lauf mit alten Waschmaschinen-Durchläufen in der Datenbank abgeglichen. Wird der Stromverbrauch oder die Laufzeit weiter unterschritten ist eine Aktivierung des Gerätes nicht plausibel. Passen jedoch die Werte zu den Vergangenheitsdaten, kann davon ausgegangen werden, dass dieses Gerät aktiviert wurde.

5. Schreiben der Ergebnisse Wurde ein Gerät erkannt und der Lauf dieses Gerätes ist plausibel, werden diese Werte in einer Datenbank abgelegt. Das Ergebnis der Disaggregation sind Prognosen für den Verbrauch eines einzelnen Gerätes im Verlauf der Zeit. Auch eine Liste von Timestamp mit der Leistungsaufnahme zu diesen Zeitpunkten. Diese Daten können in eine neue Tabelle der Influx-Datenbank geschrieben werden. Die disaggregierten Daten werden analog zu den Daten des Stromzählers in die Datenbank geschrieben, nur diese sind durch die Information des Gerätetyps zu den Daten erweitert.

6. Dauerhafter und kontinuierlicher Einsatz Grundsätzlich soll der Prototyp kontinuierlich im Einsatz sein und wiederholt versuchen die aktuellen Verbrauchsdaten zu disaggregieren. Die Erzeugung einer solchen Situation mit dem Simulationsaufbau ist nicht praktikabel, da die Verbraucher manuell an den Stromzähler angeschlossen werden mussten. Daher sind theoretische Konzepte und erste Implementierungen für einen dauerhaften Betrieb entwickelt worden, aber konnten nicht konkret umgesetzt werden. Viele Komponenten sind bereits für den kontinuierlichen Betrieb vorgesehen. Zu Anfang wäre da die Aufnahme der Daten mit der Messstation und das Eintragen der Daten in die Datenbank.

Das Skript, welches die Daten von der optischen Schnittstelle ausliest und speichert, wird durch den ankommenden Datenstrom gesteuert. Dieser ist so umgesetzt, dass schon dauerhaft die Daten entgegengenommen werden können. Auch der Webservice läuft dauerhaft und liefert die Daten zurück, sobald diese von einem Client angefragt wurden. Das Erstellen eines eigenen Datensatzes kann nicht ohne einen weiteren automatisiert werden, da die Geräte noch manuell eingetragen werden müssen.

Der Schritt der automatisiert werden muss, ist die Durchführung der Disaggregation. Dafür wurde auch ein entsprechendes Konzept erarbeitet. Dies ist nicht komplett praktisch im Prototypen umgesetzt, daher wird hier das theoretische Konzept kurz erläutert:

Um Geräte zu erkennen werden die Daten die vom Stromzähler als Testdaten bei der Disaggregation verwendet. Beim kontinuierlichen Betrieb handelt es sich dabei um einen potentiell unendlichen Datenstrom. Daher wird beim Einsatz von Machine-Learning auf sogenannte Zeitfenster zurückgegriffen. Das bedeutet es werden nicht alle aufgezeichneten Daten als Testdatensatz verwendet, sondern nur die Daten die einen bestimmten Zeitraum zurückliegen. Und dieses Vorgehen wird nach bestimmter Zeit immer wiederholt. Für die Art von Geräten die untersucht werden sollen, kommt etwa ein Zeitfenster von zwei bis drei Stunden in Frage. Damit kontinuierlich die Erkennung der Geräte vorgenommen werden soll, wird nicht erst bis zum Ablauf des nächsten Zeitfensters abgewartet, sondern wird nach kurzer Zeit immer wieder neu gestartet. Dies führt dazu, dass sich die Zeitfenster bei verschiedenen Durchläufen der Disaggregation überschneiden. Dieses Prinzip wird als *Sliding-Time-Window* bezeichnet.

5.4.8. Komponente 07: Webservice

Die Daten, die auf dem Raspberry Pi gespeichert und verarbeitet worden sind, sollen für einen eventuellen Nutzer visualisiert werden. Um eine möglichst generische Schnittstelle zu schaffen, wird dafür eine HTTP-Schnittstelle mit Hilfe des Web Application Framework Flask genutzt. Dieses Framework ist selbst auch in Python geschrieben und ermöglicht es in Python eine stabile und flexible Schnittstelle zu implementieren.

Die Tabelle 7 zeigt eine Übersicht über die einzelnen Services mit den entsprechenden URIs für die Anfragen. Es werden ausschließlich GET-Methoden als HTTP-Method verwendet, da nur Daten abgerufen werden. Von der Client-Seite werden in keinem Fall Daten am Server verändert, gelöscht und auch nicht erstellt.

Die ersten drei Services geben relevante Daten für einen bestimmten Tag, einen Monat, ein Jahr wieder, dafür können über die URL-Parameter die entsprechenden Argumente übergeben werden. Durch das Angeben der genauen Parameter bleibt die Anfrage der

HTTP-Method	URI	Action
GET	http://[hostname]/api/day?date=2017-08-13	Gibt relevante Daten für einen Tag wieder
GET	http://[hostname]/api/month?month=2017-08	Gibt relevante Daten für einen Monat wieder
GET	http://[hostname]/api/year?year=2017	Gibt relevante Daten für ein Jahr wieder
GET	http://[hostname]/api/overview	Gibt definierte Daten für eine erste Übersicht wieder
GET	http://[hostname]/api/recommendations	Gibt eine Reihe von Verhaltensempfehlungen wieder

Tabelle 7: Services der HTTP-Schnittstelle

Daten generisch und kann immer die passenden Daten aufrufen. Es gibt keinen Service der nur Daten für das aktuelle Jahr widerspiegelt, sondern der Client muss dies selber übergeben. Im folgenden Abschnitt wird genauer darauf eingegangen welche Daten die verschiedenen Services zurückliefern.

Tag Die kleinste Einheit, von dem die Daten abgefragt werden können, ist ein einzelner Tag. Mit dem URL-Parameter *day* kann der Client im Format YYYY-MM-DD einen speziellen Tag definieren. Auf Basis der Anforderungen, die für die Visualisierung der Daten definiert wurden, werden über die Schnittstelle verschiedene Daten wiedergegeben. Der Code-Abschnitt 6 zeigt beispielhaft diese Daten im JSON-Format:

Listing 6: Json-Response von der HTTP-Schnittstelle

```
{
  "appliances": [
    {
      "appliance": "toaster",
      "cent": 37.00795839599999,
      "kwh": 1.4535725999999998,
      "percentage": 25.030526308756944
    },
    {
      "appliance": "dish washer",
      "cent": 37.007958396,
      "kwh": 1.4535726,
      "percentage": 25.03052630875695
    }
  ]
}
```



```
    },
    {
      "appliance": "Others",
      "cent": 73.83538261536698,
      "kwh": 2.900054305395404,
      "percentage": 49.938947382486106
    }
  ],
  "date": "2017-08-13T00:00:00",
  "kwh_day": [
    5.8071995053954035,
    147.85129940736698
  ],
  "kwh_days_mean": [
    0.5592649785612124,
    14.238886354168468
  ],
  "kwh_hourly": [
    0.12961270637637085,
    0.055983259794447494,
    "NO_DATA",
    0.0010455915488888888,
    1.044856761717775,
    0.3270277833333333,
    0.07643791581258057,
    "NO_DATA",
    0.0007479636469141666,
    "NO_DATA",
    0.00105211743,
    1.04765745,
    0.3272167393876194,
    0.13948567930680555,
    0.07731562777777778,
    0.0003934222222222222,
    1.04765745,
    0.3270277833333333,
    0.6613778111111112,
    0.3270277833333333,
    0.07731562777777778,
    "NO_DATA",
    0.03977475683546361,
    0.09317332617964974
  ],
  "kwh_price": 25.46
```

```
}
```

Unter dem Schlüssel *appliances* ist eine Liste mit Geräten aufgeführt, die durch die Disaggregation erkannt wurden. Der Gerätetyp *Others* steht für Stromverbräuche die keinem Gerät zugeordnet werden konnten. Der Wert zu *kwh_day* gibt Verbrauchsdaten und Kosten für den definierten Tag an und *kwh_days_mean* die Durchschnittsverbrauch- und kosten der letzten fünf Tage. Über *kwh_hourly* wird eine Liste von 24 Werten übergeben, die jeweils den Stromverbrauch in kWh für die entsprechende Stunde widerspiegeln. So eine Zeitreihe der Daten über den ganzen Tag ermöglicht einen tieferen Detaillierungsgrad der Verbräuche, der zur Erstellung einer grafischen Darstellung genutzt werden kann. Zum Abschluss wird mit *kwh_price* der aktuelle Preis für eine kwh an diesem Tag übergeben, mit dem auch alle Preisberechnungen durchgeführt wurden.

Monat Die nächste Schnittstelle bietet die Möglichkeit, Daten für einen bestimmten Monat abzufragen, dies kann über den URL-Parameter *month* im Format YYYY-MM durchgeführt werden. Der Aufbau der Daten der folgenden Services ist äquivalent zum Tag, nur das eine andere Abstraktionsebene der Daten vorliegt. Da die Art der Daten für den Tag schon ausführlich beschrieben wurden, wird dies für den Monat und das folgende Jahr nicht mehr genauer erläutert. Es werden also Verbräuche der einzelnen Geräte für den Monat in der Liste wiedergegeben, dazu der Gesamtverbrauch des Monats und der Durchschnittsverbrauch für die letzten fünf Monate. Für den Monat werden auch keine einzelnen Verbräuche stundenweise aufgelistet, sondern alle Werte des Monats pro Tag.

Jahr Über eine dritte Schnittstelle können Daten für ein ganzes Jahr abgefragt werden. Dabei werden jeweils Verbräuche der Geräte über das ganze Jahr betrachtet. Die Auflistung spiegelt hier die 12 Werte für die einzelnen Monatsverbräuche im Jahr wieder.

Overview Diese Schnittstelle ergab sich aus den Anforderungen für die Visualisierung der Daten. Denn neben den konkreten Anfragen für Daten eines Jahres, eines Monats oder für einen bestimmten Tag, besteht auch das Bedürfnis für eine Abfrage von Daten unterschiedlichster Art. Bei dieser URL werden aber keine Parameter für einen bestimmten Zeitpunkt angenommen, sondern die Daten beziehen sich immer auf den aktuellen Zeitpunkt. Über den Schlüssel *energy_today* wurden Energieverbrauch und Kosten für den aktuellen Tag berechnet, dies gilt für den Tagesverbrauch bis zum Zeitpunkt, an dem die Daten abgefragt wurden. Wird diese Schnittstelle beispielsweise um 14 Uhr abgefragt, wird der Energieverbrauch von 0:00 bis 14:00 an diesem Tag wiedergegeben. Das gleiche gilt auch

für die Daten aus dem aktuellen Monat, die mit dem Schlüssel *energy_this_month* angegeben werden. Des Weiteren wird der Verbrauch für den letzten Monat mit übergeben, um eine Vergleichbarkeit herzustellen.

Recommendations Ein weiterer Ansatz der für die Repräsentation der Verbrauchsdaten auf einem Endgerät entwickelt wurde, sind sogenannte Handlungsempfehlungen für den Nutzer. Auf Basis der aufgezeichneten Verbräuche und der berechneten Laufzeiten von Geräten lassen sich Handlungsempfehlungen ableiten. Diese Handlungsempfehlungen können über den letzten Service angesprochen werden. Mit dem Schlüssel *recommendations* wird eine Liste mit diesen Handlungsempfehlungen übergeben, die durch das System berechnet wurden.

Berechnung der Verbrauchsdaten Die Services für Tag, Monat, Jahr und Overview fragen grundsätzlich ähnliche Daten ab, auf denen die Berechnungen durchgeführt werden. Abgefragt wird die aufgezeichnete Leistung für eine bestimmte Stunde, einen bestimmten Tag, einen bestimmten Monat oder auch ein bestimmtes Jahr. Wie im Abschnitt zu der Komponente Datenbank erläutert, werden diese Daten in eine Influx-Datenbank geschrieben. Der Webservice muss mit seinen Methoden die Daten zur verbrauchten Leistung direkt aus der Influx-Datenbank lesen.

Über die HTTP-Schnittstelle der Datenbank greift der Webservice auf die Daten in der Influx-Datenbank zu. Dafür werden unterschiedliche Methoden genutzt, um verschiedene Zeiträume abzufragen. Die Empfehlungen die an den Kunden übergeben werden, ist bei Schnittstelle für eine mögliche Visualisierung auf einem Endgerät beachtet worden. Doch die tatsächlichen Services sind noch theoretisch und sind im aktuellen Prototypen nicht praktisch umgesetzt. Daher basieren Handlungsempfehlungen die durch die Schnittstelle abgefragt werden können nicht auf realen Berechnungen, sondern dienen als Platzhalter.

Memorization Bei der Umsetzung der Webservices ist das Problem aufgetreten, dass Abfragen und Berechnungen von Gesamtverbräuchen gerade bei längeren Zeiträumen, eine gewisse Zeit in Anspruch nimmt. Bei den Aufzeichnungen über einen Monat mit einer Frequenz von 0,5 Hertz entstehen maximal etwa 1,3 Millionen Datensätze, die dann bei der Berechnung für einen einzelnen Monat verarbeitet werden müssen. Dies kann zu einer gewissen Dauer bei der Abfrage der Daten über die REST-Schnittstelle führen. Gerade bei der Darstellung auf einem mobilen Endgerät sind solche Wartezeiten unerwünscht.

Da die Daten wiederholt mit gleichen Parametern abgefragt werden, bietet sich hier das Konzept der Memorization einzusetzen. Das Konzept der Memorization verfolgt den

Ansatz die Ergebnisse für bereits durchgeführte Berechnungen zu speichern. Sollen diese Berechnungen erneut durchgeführt werden, werden die Daten aus dem Speicher geladen. Das Laden spart redundante Rechenzeiten und beschleunigt dadurch die Prozesse. Ein bekanntes Beispiel ist der Einsatz dieser Technik in Verbindung mit der Berechnung von rekursiven Funktionen wie Fibonacci-Zahlen [Pau08].

Für den Webservice wird die Memorization dazu genutzt, um die Berechnungen von Energieverbräuchen über bestimmte Zeiträume zwischen zu speichern. Die Voraussetzung, um diese Art von Berechnungen zwischenspeichern ist, dass die Funktionen referentiell transparent sind, also die Resultate bei gleichen Parametern identisch sind. Das trifft zu, da sobald der Zeitraum eines Tages, Monats oder Jahres abgeschlossen ist, auch die Aufzeichnungen der Messstation und das Eintragen in die Datenbank final sind. Mit dem Ziel auch langfristig auf die Werte zugreifen zu können, werden diese nicht nur im Hauptspeicher zwischengespeichert, sondern persistent auf der Festplatte abgelegt. Hierfür wurde auf die einfache Möglichkeit der Serialisierung zugegriffen, dabei werden vorliegende Python-Objekte in einer einzelnen Datei serialisiert. Dies ist eine schnelle und simple Möglichkeit die Daten in ihrer ursprünglichen Struktur zu speichern und zu laden. Dafür wird das standardmäßige Tool mit dem Namen Pickle zurückgegriffen.

Wird versucht auf Verbrauchsdaten eines Tages zuzugreifen, wird kontrolliert ob diese Daten bereits berechnet wurden. Die wichtigen Parameter sind dabei die Kategorie (z. B. Tag), der Zeitraum (z. B. 07.07.2017) und ein mögliches Geräte, dem die Verbrauchsdaten zugewiesen sind. Sind keine Daten berechnet worden, wird die Abfrage und Berechnung der Daten neu gestartet. Nach der Berechnung werden die Daten gespeichert, falls diese bei einer späteren Abfrage wieder benötigt werden. Werden die Daten mit den gleichen Parametern noch einmal angefragt, können diese aus der serialisierten Datei geladen werden.

Automatisierte Berechnung Bei der Einführung der Memorization zeigt sich eine neue Problemstellung: Denn die Zeiteinsparung die durch die Einführung des Konzeptes möglich ist, wird nur erreicht, wenn die Berechnung der Daten bereits einmal stattgefunden hat. Also bei der ersten Abfrage der Daten entstehen durch die Berechnung weiterhin Wartezeiten.

Diesem Problem wird mit der automatisierten Berechnung von den Verbrauchswerten Abhilfe verschafft. Die Daten für den Verbrauch eines bestimmten Zeitraumes werden vorgenommen, sobald die Daten in der Datenbank vorliegen. Die Berechnungen werden automatisch gestartet, bevor der Webservice auf diese Daten zugegriffen hat. Am 21.09.2017 um 00.00 Uhr finden also beispielhaft verschiedene Berechnungen automatisch statt. Zuerst einmal wird der Verbrauch der 23. Stunde des 20.09.2017 statt, zudem findet auch die

Berechnung des gesamten Energieverbrauches des Tages statt. Diese Stundenwerte werden dann zum aktuellen Verbrauch des Monats und des Jahres hinzuaddiert, um auch hier den aktuellsten Stand zu berechnen.

Die Ausführung der Berechnungen findet über einen Cron-Daemon der genutzten Linux-Distribution statt. Dieser macht es möglich Methoden automatisch zu festen Uhrzeiten wiederholt auf dem System auszuführen. Auf eine detaillierte Beschreibung der Cron-Implementierung wird an dieser Stelle abgesehen, das entsprechende Modul trägt den Namen `cron_memorization.py`.

Der Einsatz von Memorization und automatisierten Skripten stellt nicht die optimale Lösung dar, da immer ein Zeitversatz bei der Berechnung vorliegt. Für den Einsatz innerhalb des Prototyps ist die Lösung mehr als ausreichend. Eine effizientere Lösung wäre die Berechnung des Verbrauches direkt am Datenstrom der Leistungsaufnahme über die Messstation. So würde eine Berechnung der Verbrauchsdaten in Echtzeit vorgenommen werden können.

5.4.9. Komponente 08: Visualisierung

Zur Visualisierung der Ergebnisse aus der Disaggregation der Energiedaten, wurde der Einsatz einer auf Smartphones angepassten Applikation gewählt. Die Entscheidung darüber erfolgte bewusst, da der Einsatz einer Anwendung über ein mobiles Endgerät weitaus flexibler ist. Dass die Darstellung von Anwendungen auf die Nutzung des Smartphones angepasst werden sollte, zeigt auch eine Studie von Adobe Digital Insights. Demnach ist der entstandene Traffic in Europa für die Nutzung von Webseiten von Januar 2014 bis Dezember 2016 auf Smartphones um 156 % gestiegen, wohingegen das Aufrufen von Internetseiten auf Desktops um -25 % zurückgegangen ist [CMO].

ionic Bei der Auswahl nach einem Framework zur Erstellung einer mobilen Applikation, fiel die Entscheidung auf das Ionic-Framework¹². Ionic ist ein kostenloses Open-Source Software Development Kit zur Entwicklung von nativen und progressiven Web-Apps. Dabei bildet es für verschiedene mobile Betriebssysteme eine gemeinsame Code-Basis, welche leicht in eine native App transferiert werden kann. Dafür nutzt Ionic wiederum die Frameworks Apache Cordova bzw. Adobe PhoneGap. Dies ermöglicht, dass die App native Geräte-gebundene Features und Schnittstellen unterstützen kann, wie beispielsweise Bluetooth oder den Fingerabdrucksensor. Die Applikation an sich kann jedoch ebenfalls im Browser als übliche Web-App verwendet werden.

¹² <https://ionicframework.com/>

Zur Erstellung einer Ionic Applikation wird, wie in vielen mobilen Anwendungen im Front-End, die Kombination von HTML5, CSS3 und TypeScript (bzw. JavaScript) eingesetzt. Der Aufbau funktioniert als eine Art Aufsatz für Googles Webapplikationsframework *Angular*. Hierdurch ist eine gemeinsame Code-Basis für verschiedene Plattformen möglich. Des Weiteren bietet Ionic den Vorteil, dass es viele Design-Elemente vorgibt. Vorgegebene Komponenten, wie beispielsweise Listen, Karten, Tabs oder Icons, können so auf einfache Art erzeugt werden und bedienen sich automatisch den Android- bzw. iOS-Designelementen. Dadurch entsteht ein simples, aufgeräumtes und funktionales Design [Dri].

Die Visualisierung der Teilgruppe energy of WOW erfolgte bereits vor der Entscheidung, die drei Systeme der drei Teilgruppen in einer gemeinsamen Applikation zu kombinieren. Daher wird im Folgenden nicht auf alle Funktionen, die bereits in vorhergehenden Kapiteln genannt wurden, in der Ionic-App eingegangen. Der Grund ist, dass diese Funktionen zu diesem Zeitpunkt noch nicht existierten.

Aufbau und Funktionen der App Grundsätzlich war das Ziel der Visualisierung in erster Linie, die Ergebnisse aus der Disaggregation der Energiedaten zu veranschaulichen. Aus den Funktionen des Prototyps ergaben sich hieraus folgende Komponenten, die präsentiert werden können:

- Auswahl einer bestimmten Zeitperiode (Tag, Woche, Monat, Jahr)
- Stromverbrauch in kWh für eine Periode
- Kosten für den Stromverbrauch in Euro für eine Periode
- Verbrauchskurve für den gesamten Stromverbrauch einer Periode
- Disaggregierte Verbrauchskurve für einzelne Geräte

Hierfür wurden für die Applikation zwei verschiedene Screens bereitgestellt. Zum einen die Startseite mit dem Namen „Overview“. Hier werden generelle Informationen zum Stromverbrauch dargestellt. Dazu gehören der Verbrauch in kWh und die dazugehörigen Kosten in Euro. Außerdem folgt ein Diagramm, welches eine zeitliche Kurve des Verbrauchs der ausgewählten Periode anzeigt. Durch eine Auswahl, die unter dem Graphen platziert ist, kann die gewünschte Zeitperiode ausgewählt werden. In Abbildung 45 wird die „Overview“-Seite dargestellt. Der Unterschied zwischen den Ansichten von Android und iOS sind die plattformspezifische Designelemente. Diese sind hier an den Buttons, den Tabs sowie an den Icons erkennbar.

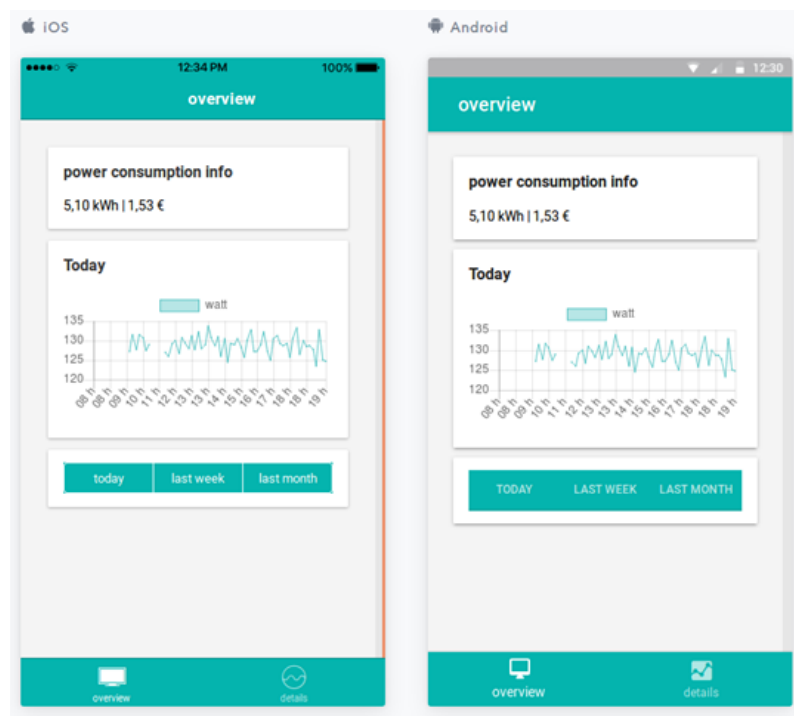


Abbildung 45: Screenshots der Pyranha App Overview [Eigene Quelle]

Des Weiteren existiert die Seite „Details“, auf die mit einem Klick auf den Tab unten rechts zugegriffen werden kann. Der Details-View gibt die Informationen, welche sich aus der Disaggregation der Lastkurve ergeben wieder. Hauptsächlich besteht die Seite hierbei aus verschiedenen Graphen der Verbrauchskurve der erkannten Geräte, wobei oben der zusammengefasste Graph der Periode nochmals zum Vergleich angezeigt wird. Abbildung 46 zeigt die Detail-Seite als iOS- und als Android-Version.

Implementierung und Struktur Zur Erstellung der Funktionen und Seiten der Ionic-App, fand die Implementierung hauptsächlich in den zwei Bereichen der Struktur statt, dem „pages“- sowie dem „providers“-Ordner. In „pages“ werden die jeweiligen Views erstellt, die in der App angezeigt werden sollen. Dieser Ordner erhält pro Komponente einen Unterordner, welcher jeweils eine HTML5, CSS3 und eine TypeScript-Datei enthält. Unter „providers“ wird die Kommunikation mit den Schnittstellen zum Backend gesteuert. Hierfür wurde für die verschiedenen Abfragen jeweils eine TypeScript-Datei erstellt. Abbildung 47 zeigt einen Auszug aus der Projekt-Struktur der Applikation, bei dem die Sichten „details“ und „home“ („overview“) sowie die die Provider zu sehen sind.

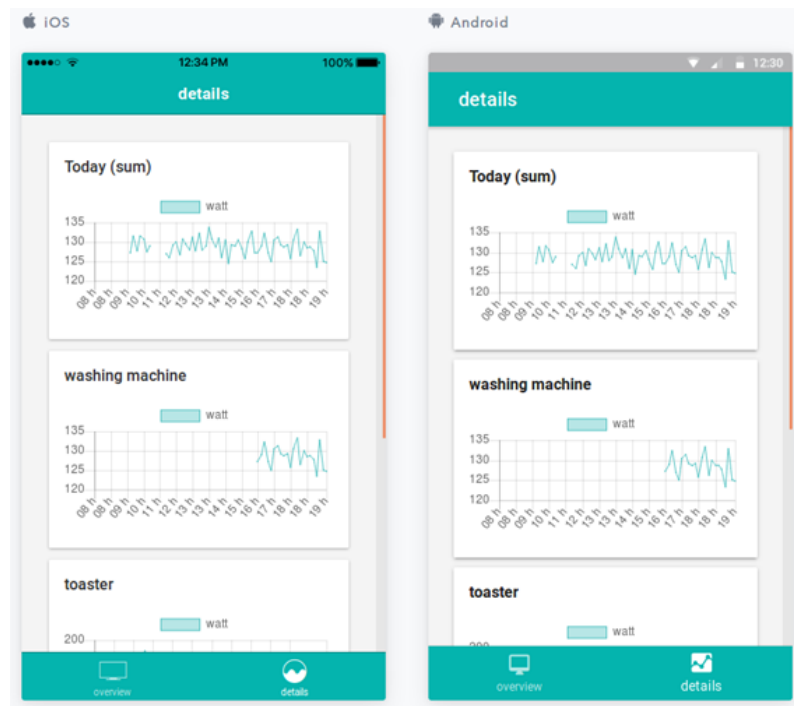


Abbildung 46: Screenshots der Pyranha App Details [Eigene Quelle]

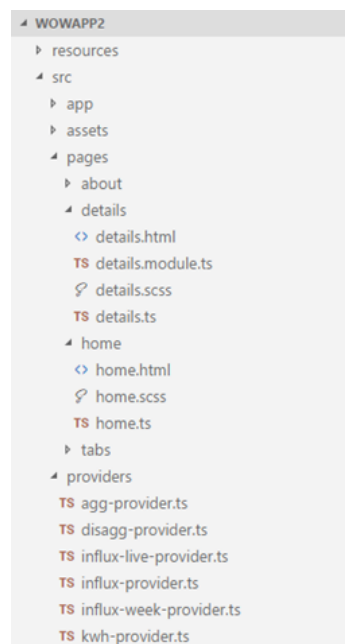


Abbildung 47: Datenstruktur der Pyranha App [Eigene Quelle]

Die wichtigsten Funktionen der Provider, welche die Schnittstellen zu der Datenbank im Backend bilden, werden in folgendem Listing 7 beschrieben. Anzumerken ist, dass hierbei noch nicht die eigenen Webservices über die Python Flask genutzt wurden, sondern ein integrierter Service der InfluxDB zum Einsatz kam. Dieser ermöglicht es eine Abfrage bei der Datenbank durchzuführen, um an die angeforderten Daten zu kommen.

Listing 7: Auszug der Schnittstelle

```
// Initialisiert die Variable influxUrl mit der http-Anfrage zu den
// gewünschten Daten.
// Kann nach verschiedenen Attributen selektiert werden.

influxUrl : string =
    'http://' + this.ipaddress + ':' + this.port +
    '/query?pretty=true&db=' + this.db +
    '&q=SELECT mean(LSum) FROM ' + this.measurement +
    ' where time > ' + this.starttime +
    ' and time < ' + this.endtime +
    ' group by time(10m)';

// gibt das Ergebnis der http-GET-Anfrage an die Datenbank wieder
getInfluxData() {
    return this.http.get(this.influxUrl)
        .do(this.logResponse)
        .map(this.extractData)
        .catch(this.catchError);
}

// extrahiert die angegebenen Daten des JSON-Files (watt value und timestamp
// ),
// welche die Datenbank zuvor ausgegeben hat
private extractData(res: Response) {
    return res.json().results[0].series[0].values;
}

// uebernimmt die Daten aus den dazugehoerigen Providern ,
// um sie an die Methoden zur Erstellung des Graphen weiterzugeben
getInfluxData() {
    this.aggProvider.getInfluxData().subscribe(data => {this.aggChartData = data
        ;
        this.createAggChart()});
}
```

Die Pages beinhalten die Darstellung der Applikation. Hier wird sichergestellt, dass die

Daten in der App erscheinen, welche in den Providern abgerufen werden. Wie üblich werden das Markup und das Design der Oberfläche mit HTML und CSS realisiert. Der Einsatz von CSS ist bei Ionic-Applikationen jedoch nicht zwingend notwendig, um eine ansprechende Visualisierung zu erhalten, da die Ionic Komponenten bereits eigene Styles haben, welche an die Design-Elemente von Android und iOS gerichtet sind.

In den TypeScript-Files der Pages werden die Daten aus den Providern übernommen, um sie hier in ein Diagramm zu überführen. Für die Darstellung der Graphen wurde die JavaScript Bibliothek Chart.js genutzt.

Aufgrund der Tatsache, dass die Entscheidung getroffen wurde, eine gemeinsame Applikation zu entwerfen, welche alle drei Teilprojekte miteinander vereint, wurde die Weiterarbeit an der Ionic-App nicht mehr fortgesetzt. Fehlende interessante Features, wie eine Auflistung der zuletzt aktiven Geräte und deren Verbräuche, sowie Handlungsempfehlungen wurden erst in das Design der neuen Applikation berücksichtigt. Auch in Hinblick auf die Ergebnisse der durchgeführten qualitativen Studie, welche im folgenden Abschnitt thematisiert wird, erfüllt die kollaborative Anwendung mehrere wichtige Aspekte, welche zu einer positiveren Nutzererfahrung beitragen könnten.

5.4.10. Zusammenfassung

In diesem Abschnitt wurden die wichtigsten Schritte der Implementierung des Prototypen erläutert. Auf Basis des entwickelten Grobkonzeptes wurde ein Gesamtkonzept entworfen, das verschiedene Komponenten umfasst, die für die Umsetzung der Anforderungen notwendig sind. Die erste wichtige Komponente stellt die Messstation dar, dieser umgebaute Stromzähler zeichnet die Verbrauchsdaten elektrischer Verbraucher auf und bietet diese Werte über eine optische Schnittstelle an. Diese Daten werden mit Hilfe einer Preprocessing Software vom Stromzähler aufgezeichnet und in eine Influx-Datenbank geschrieben. Diese Datenbank befindet sich auf einem Einplatinencomputer, dem Raspberry Pi Model 3. Eine weitere Softwarekomponente mit dem Namen Appliance-Simulator ermöglicht es Verbrauchsdaten von bestimmten elektrischen Verbrauchern zu simulieren.

Die Komponente Converter konvertiert Datensätze zu einem HDF-Format und verknüpft dabei vorhandene Metadaten mit den Verbrauchsdaten. Mit diesen Datensätzen werden sowohl Trainingsdatensätze für die Disaggregation aufgebaut. Als auch die Datensätze erstellt, die disaggregiert werden sollen. Die Komponente Disaggregation nutzt diese Datensätze und führt die Disaggregation durch, basierend auf der Implementierung aus dem NILM-Toolkit. Die Ergebnisse werden wieder zurück in die Influx-Datenbank geschrieben. Ein Webservice bietet über eine HTTP-Schnittstelle diese Ergebnisse an. Eine hybride Ap-

plication die mit Ionic umgesetzt wurde, greift auf diese Schnittstelle zu und visualisiert diese Daten.

Im Zusammenspiel ergeben alle Komponenten einen Gesamtprozess, der zur Analyse der Verbrauchsdaten die Disaggregation einsetzt.

5.5. Empirisch-qualitative Studie

Zum Zweck einer realistischen Abschätzung von Nutzungsmöglichkeiten des Prototyps wurde eine qualitative Studie in Form von Interviews durchgeführt. Im Prototyp werden Verbrauchsdaten und Ergebnisse der Disaggregation in einer App visualisiert. Ziel dieser Interviews war es zu ermitteln, welche Informationen aus Sicht eines Nutzers interessant sind. Im folgenden Abschnitt werden Forschungsdesign und Ergebnisse der Studie vorgestellt.

5.5.1. Forschungsdesign und Methodik

Für die Evaluation wurden semistrukturierte Interviews durchgeführt. Es wurden Probanden ausgesucht, die sich mit dem Thema Energie intensiver beschäftigen als der durchschnittliche Konsument. Insgesamt wurden acht Probanden befragt und mittels eines strukturierten Interviewleitfadens nach Mayring durch das Gespräch geführt. Dabei werden die Probanden aufgenommen und das Gespräch anschließend transkribiert. Der Interviewleitfaden ist im Anhang dieser Dokumentation zu finden und ist wie folgt strukturiert: Der Fragebogen verfügt über 18 Leitfragen. Zu fast jeder Fragestellung existieren eine oder mehrere Fragen, um gezielt Informationen aus dem Probanden zu eruieren. Deshalb wird zur Erklärung der Leitfaden in fünf Abschnitte aufgeteilt.

1. Im ersten Abschnitt werden Angaben zur Person abgefragt. Der Proband gibt an, welchen Beruf dieser ausübt oder in welchen Bereichen dieser forscht. Außerdem soll der Proband über seinen Werdegang erzählen. Dies hat den Zweck, den Probanden in eine Sprechsituation zu bringen und ihn für die nachfolgenden Themen kommunikativer zu machen.
2. Im zweiten Abschnitt zum Energiemarkt sollen Informationen zum Bewusstsein und zum privaten Umgang mit Energie des Probanden gewonnen werden.
3. Im dritten Abschnitt folgen generelle Fragen zum Nutzungsverhalten von mobilen Endgeräten. Passend dazu werden Präferenzen für die Aufbereitung energiebezogener Daten abgefragt.

4. Der vierte Abschnitt ist der Kern des Fragebogens, da dieser in Verbindung mit dem Prototyp zur Disaggregation von Lastkurven steht. Der Proband soll abschätzen, wie er die Geräteerkennung privat am liebsten einsetzen würde und welche Informationen wie nutzbar sein sollen.
5. Zum Schluss des Interviews soll der Proband speziell die Disaggregation von Lastkurven kritisch reflektieren und andere Nutzungsmöglichkeiten aufzeigen.

Die Ergebnisse der Befragungen sollen einen gewissen qualitativen Standard aufweisen. Den Aussagen von Prosumern und Personen mit professionellen Ansprüchen, wird unterstellt, diesen Anforderungen gerecht zu werden. Zu diesem Zweck werden Prosumer gesucht, die sich mit dem Energiebereich entweder beruflich/forschend befassen oder privat mit Energie bewusst auseinandersetzen. Zudem ist dieser Beitrag getrieben von einem interdisziplinären Forschungsansatz.

5.5.2. Auswertung der Interviews

Innerhalb dieser Auswertung wird die Analyse und Interpretation der Interview-Studie durchgeführt. Als unterstützende Software hat sich MAXQDA bewährt. Dafür wurde ein Kodierleitfaden erstellt, anhand dessen die Interviews systematisch analysiert werden. Die Ergebnisse des Kodierleitfadens werden anhand von drei Oberpunkten beschrieben und interpretiert. Dabei wurde ermittelt, welche Informationen in einer App dem User am wichtigsten sind. Diese wurden in zwei Kategorien aufgeschlüsselt, die die Informationen und die Funktionen abbilden. Außerdem gibt es eine allgemeine Kategorie, die die Forschungsfrage thematisiert.

Abgesehen von der Vorbildung und dem Beschäftigungsverhältnis der Probanden wurde ermittelt, wie diese in Verbindung zur Energie und zum Energiemarkt stehen. Es ist festzuhalten, dass sieben von acht Probanden ein in ihren Augen sehr hohes Bewusstsein für Energie aufweisen. Sie legen Wert auf Netzstabilität, sparen aktiv Strom und ihnen sind energieeffiziente Geräte sehr wichtig. Lediglich drei Probanden legen weniger Wert auf ihren Stromverbrauch, relativieren dies aber direkt, da ihnen Anreize verwehrt bleiben und sie Strom ohnehin verbrauchen. Sechs Probanden kontrollieren ihren Stromverbrauch stetig, zwei von ihnen passen den Stromverbrauch aktiv an. Signifikant ist, dass vier Probanden bereits als Produzenten tätig sind, da sie eine Photovoltaikanlage besitzen. Dies hat vorrangig finanzielle Gründe.

Anschließend wurde die Erwartung an Smart Meter-Daten der Probanden erfasst. Alle erwarten einen persönlichen Vorteil von den gesammelten Daten. Dennoch können diese Daten eine Sensibilisierung der Menschen ermöglichen. Für Energieversorger könnten derartig

aufbereitete Daten bessere Planungsmöglichkeiten bedeuten. Mögliche Einzelverbräuche in Haushalten zu identifizieren, um Sparpotenziale aufzudecken. Schlussendlich könnten diese Daten an Unternehmen veräußert werden, die diese für ihre Zwecke weiterverarbeiten können.

Im Anschluss sollten die Probanden ihre die Nutzungsaffinität mobiler Endgeräte einschätzen. Im Kontext einer energiebasierten mobilen Applikation sind zwei Probanden skeptisch. Dennoch kann ein Interesse entstehen, wenn diese zur Verfügung stehen würde. Als Alternative kann eine festinstallierte Station, anhand des Vorbilds der Photovoltaik-Anlagen, oder auch dezentere Informationen an einer Küchenuhr geeignet sein. Eine mobile Website mit Login-Funktion hingegen wurde von einigen Befragten abgelehnt. Auf einem Smartphone soll die Applikation kontextsensitiv und schnell sein, wobei eine gute grafische Aufbereitung wichtig ist. Es lassen sich kaum Nutzungsintervalle prognostizieren, jedoch würde mit der Zeit das Interesse abflachen.

Im Zentrum der Interviewstudie stehen Fragen nach der Funktion einer energiebasier-ten Applikation. Interessant ist die Darstellung des Gesamtverbrauchs sowohl in aktueller Form als auch auf Tages-, Wochen- und Monatsbasis. Hierbei sind Auswertungsmöglichkeiten zum Energieverhalten, Konsum und auch Produktion denkbar. Dabei sind Empfehlungen zur Reduzierung des Energieverbrauchs wünschenswert. Des Weiteren stehen für alle Probanden die Kosten im Vordergrund. Es ist denkbar, dass Kosten am Ende eines bestimmten Zeitraums einsehbar sind. Außerdem wünschen sich zwei Probanden Amortisationsrechnungen, was eine Kopplung mit der Stromerzeugung oder der Anschaffung von energieeffizienten Geräten bedingt. Eine dritte Funktion ist die Kopplung von Strompreisen und Tarifen. Dabei wäre es interessant, die Tarife flexibel zu gestalten.

Da die Disaggregation von Lastkurven und der konzipierte Prototyp im Mittelpunkt dieser Arbeit stehen, wird ein Hauptaugenmerk auf die Nutzbarkeit der Erkennung von Endgeräten gelegt. Eine Echtzeitdarstellung des Verbrauchs eines Gerätes wird von allen Probanden als nicht notwendig angesehen. Dennoch wurden durch die Probanden verschiedene gewünschte Funktionsmöglichkeiten aufgezeigt. Ganz nach dem Vorbild des Gesamtverbrauchs wünschen sich die Probanden, Geräte im Zeitverlauf, auf Tages- Monats- oder Jahresbasis betrachten zu können. Ein Proband wünscht sich eine Betrachtung nach Tageszeit. Infolgedessen ist es jedoch interessanter, dass Anomalien betrachtet werden können. Durch die Erkennung einzelner Geräte im gesamten Stromverbrauch ist es möglich „Energiefresser“ zu identifizieren. Des Weiteren wäre es möglich, defekte Geräte zu erkennen. Dies wird als enormes Potenzial zur Stromeinsparung gesehen. Hierbei könnten sich die Probanden außerdem Tipps zu Optimierungen in Form von Benachrichtigungen vorstellen.

Darüber hinaus ist die Integration der Applikation in einer Smart-Home-Anwendung interessant. Neben der Analyse könnte die Funktionalität, um die Steuerung der Geräte erweitert werden. Auch Prognoserechnungen über zu erwartende Stromverbräuche sind denkbar. Um Anreize zu schaffen sind eine Community-Funktion interessant, um sich mit den Nachbarn oder einer Gruppe vergleichen zu können. Inhaltlich könnte es um die Vergleichbarkeit von einzelnen Geräten gehen oder auch die Gesamtverbräuche der Haushalte zu bestimmten Zeiten.

Abgesehen von den Informationen ist auch die Aufbereitung der Daten ein wichtiger Aspekt einer mobilen Applikation. Die Probanden fänden es interessant über verschiedene, frei wählbare Ansichten zu verfügen, die ganz an die Bedürfnisse der einzelnen Personen angepasst werden können. Dabei sollen die Informationen in Relation zur Zeit und zum Geld stehen. Trotzdem ist es wichtig am Anfang über ein übersichtliches Dashboard zu verfügen, dass die wichtigsten Informationen aggregiert bereitstellt. Die bereits angesprochenen Benachrichtigungen stellen sich als überaus sinnvolle Funktion heraus. Auf Basis einer Geräteerkennung könnten Benachrichtigungen erfolgen, falls Gerätedurchläufe beendet sind, Geräte zu lange laufen, Anomalien auftreten oder andere Optimierungshinweise vorliegen. Diese Benachrichtigungen sollten moderat erfolgen. Deshalb sollte die Applikation automatisch Informationen filtern, oder dem Nutzer sollten Konfigurationsmöglichkeiten dazu gegeben werden. Als alternative Form der Benachrichtigungen können E-Mail oder SMS versendet werden.

Am Ende eines jeden Interviews wurde der Proband darum gebeten, sich Gedanken über die negativen Folgen eines derartigen Energiemonitoring zu machen. Allgemein lässt sich festhalten, dass eine hohe Sensibilität der Datensicherheit vorliegt. Die Probanden haben eine kritische Haltung gegenüber Dritten, die Interesse an derartig fein aufbereiteten Daten haben könnten. Es sind tiefe Einblicke in das Leben eines jeden Menschen möglich, insbesondere falls die Disaggregation von Lastkurven und die daraus resultierende Geräteerkennung umgesetzt wird. Lösungsansätze gibt es mehrere: Grundlegend sollte die Datensicherheit gesetzlich reguliert sein. Dabei sollten mindestens allgemeine IT-Sicherheitsstandards erfüllt sein. Darüber hinaus dürfen nur eingeschränkt Informationen in die Hände Dritter weitergeleitet werden. Dies kann durch eine lokale Datenhaltung und Verarbeitung gewährleistet werden. Dafür ist jedoch eine Recheneinheit im Haushalt notwendig. Im Idealfall kann der Nutzer freiwillig entscheiden, was mit seinen persönlichen Daten geschieht.

5.5.3. Ergebnis der Studie

Diese Studie soll zeigen, wie die Geräteerkennung sinnvoll eingesetzt werden kann und welche daraus resultierenden Informationen für Konsumenten wichtig sind. Die Konsumenten in dieser Studie sind Menschen, die sich auch von beruflicher Seite mit dem Thema Energie beschäftigen. Dies lässt Rückschlüsse ziehen, welche Informationen durch ein Medium bereitgestellt werden sollten, damit auch der durchschnittliche Konsument erhöhtes Interesse für Energie zeigt.

Zum einen wird deutlich, dass eine mobile Applikation zur Darstellung der Informationen geeignet ist. Denn Smartphones bieten üblicherweise die Möglichkeit, ort- und zeitunabhängig auf die Informationen zugreifen zu können. Außerdem werden umständliche Logins vermieden, da die Applikation nur beim ersten Aufruf eingerichtet werden muss und die Informationen daraufhin jederzeit schnell einsehbar sind. Der Nutzer hat dann ortsunabhängig die Möglichkeit sich mit dem Thema der Energie zu befassen. Darüber hinaus haben die Konsumenten eine anfangs häufigere Nutzung beschrieben, welche mit der Zeit wohl abnehmen würde. Dieses Nutzerverhalten ließe sich sehr gut mit einer mobilen Applikation decken. Dies trägt zur angesprochenen Sensibilisierung der durchschnittlichen Konsumenten an. Deshalb kann festgehalten werden, dass die Informationen durch ein mobiles Medium zu dem Konsumenten geführt werden sollten.

Zum anderen wird von den Probanden die visuelle Aufbereitung der Informationen stark thematisiert. In Bezug auf den Durchschnittskonsumenten wäre eine einfache und verständliche Darstellung sinnvoll. Ein Dashboard als Startseite einer mobilen Applikation, das die wichtigsten Informationen direkt aufführt, wird von allen Nutzern präferiert. Des Weiteren kann auf dem Dashboard optional auf verschiedene, detaillierte Seiten innerhalb der App verwiesen werden. Die Daten müssen grafisch sinnvoll aufbereitet sein und in Relation zu anderen Größen, wie Zeit oder Währung, gebracht werden. Gleiches gilt für die Anzeige einzelner Verbraucher. Für Konsumenten ist es wichtig zu sehen, welches Gerät wie viel Strom verbraucht. Damit werden Laien in den Zustand versetzt Anomalien zu erkennen und den Stromkonsum zu optimieren. Generell ist diese Funktion, nach Informationen der Studie, nur geeignet, um Anomalien auf Geräteebene erkennen zu können. In Kombination mit einer Benachrichtigungsfunktion erscheint die Geräteerkennung noch nützlicher. Hierbei darf der Konsument jedoch nicht zu sehr gestört werden. Eine einfache Benachrichtigung per Push Funktion bei defekten und bei angelassenen Geräten, sollte frei anpassbar sein. Nicht ausgeschaltete Geräte lassen sich durch die Verbindung mit der Standortfunktion eines Mobiltelefons umsetzen. Dabei sind auch Betrachtungen der Vergangenheitswerte interessant, damit der Konsument sein Energieverhalten selbst reflektieren kann. Im Folgenden

wird eine Auflistung mit Anforderungen gezeigt, die umgesetzt werden.

Anforderung	Beschreibung
Mobile Applikation	Informationen werden dem Konsumenten durch eine mobile Applikation angeboten.
Dashboard	Ein Dashboard, das Informationen zum Gesamtverbrauch anzeigt, wahlweise in Relation zum Strompreis.
Unterseite mit Stromverbrauch einzelner Verbraucher	Einzelne Verbraucher werden in Relation zu den Stromkosten gesetzt und historische Verbrauchsprofile angezeigt.
Benachrichtigungsfunktion	Benachrichtigungsfunktion auf Basis des Push-Prinzips mit Filterfunktionen
Unterseite mit Gesamtstromverbrauch	Vergangenheitswerte des Gesamtstromverbrauchs
Lokale Datenverarbeitung	Die Datenverarbeitung durch die Disaggregation muss lokal gehalten sein und Daten müssen lokal gespeichert werden.

Tabelle 8: Anforderungen im Rahmen der Umfrage

5.5.4. Zusammenfassung

Im Rahmen der Prototyp-Entwicklung wurde entschieden, eine empirisch-qualitative Studie durchzuführen, um Anforderungen für eine Visualisierungsmöglichkeit der Informationen aus dem Prototypen zu ermitteln. In diesem Kapitel wurde zuerst die Bestimmung der Probanden und der Interviewleitfaden vorgestellt. Während die Interviews durchgeführt wurden, wurden die Ergebnisse transkribiert und mit Hilfe des Programms MAXQDA ausgewertet. Dabei entstand ein Kodierleitfaden der die wichtigsten Eckpunkte der Gespräche sinnvoll aufbereitet. Abschließend wurden die Ergebnisse interpretiert und in den Kontext der Energiewende gebracht. Auf Basis der Ergebnisse konnten Anforderungen abgeleitet werden, die bei der Entwicklung eines Prototypen eingesetzt werden können.

5.6. Fazit und Ausblick

Zu Anfang wurden die Grundlagen zum Thema Disaggregation genauer beleuchtet, die notwendig sind, um die Schritte dieses Teilprojektes nachvollziehen zu können. Dazu gehört auch eine Einordnung der Technologie durch ein State-of-the-Art und die Erläuterung der vielfältigen privaten und gewerblichen Einsatzmöglichkeiten. Denn die Disaggregation kann unter anderem bei den Themen Forschung und Entwicklung, Erstellung von Lastprognosen und bei der Wartung technischer Anlagen dienlich sein.

Um die Umsetzbarkeit dieser Technologie zu untersuchen wurde ein Anwendungsszenario für den Einsatz eines Disaggrationsansatzes als Proof-of-Concept entwickelt. Dieses Anwendungsszenario umfasst das Aufnehmen von Daten, die Durchführung der Disaggregation und das Speichern der Ergebnisse. Dazu kommt noch die Darstellung dieser Ergebnisse über einen Client für einen möglichen Kunden. Dieses Szenario wurde als Vorlage für die Erstellung eines Konzeptes verwendet, die mit konkreten Anforderungen verknüpft wurden.

Auf Basis des entwickelten Anwendungsszenarios und dem entsprechenden Konzept wurde ein Prototyp implementiert. Dieser Prototyp besteht aus verschiedenen Komponenten, die jeweils in Verbindung stehen und die Ergebnisse anderer Module nutzen. Die erste Komponente ist die Messstation, die durch einen umgebauten digitalen Stromzähler umgesetzt ist. Dieser hat die Aufgabe Verbrauchsdaten elektrischer Verbraucher aufzuzeichnen und bereitzustellen. Diese Daten werden dann von einem Raspberry Pi abgefragt und einer Datenbank gespeichert. Die Komponente Converter hat die Aufgabe die aufgezeichneten Daten in ein HDF-Format zu konvertieren und mit Metadaten zu verknüpfen. Diese Dateien im HDF-Format werden von der Komponente Disaggregation genutzt, sowohl, um den Algorithmus zu trainieren, als auch die Disaggregation selbst auszuführen. Hinter der Disaggregation verbirgt sich ein Machine Learning Algorithmus, der aggregierte Verbrauchsdaten analysiert und die Aktivierung einzelner Geräte identifiziert. Diese Ergebnisse werden anschließend wieder in die Datenbank geschrieben. Ein Webservice ermöglicht die Abfrage der Verbrauchsdaten über eine HTTP-Schnittstelle, die entsprechend aus der Datenbank geladen werden. Mit Hilfe des Ionic Frameworks wurde eine hybride mobile Anwendung entwickelt, die auf diese Schnittstelle zugreift und die Daten für einen möglichen Endanwender visualisiert.

Zusammenfassend kann man sagen, dass die Technologie der Disaggregation großes Potential in den unterschiedlichsten Themenbereichen hat. Mit den Ergebnissen lassen sich sowohl bestehende Prozesse verbessern, aber auch neue Prozesse und Geschäftsmodelle entwickeln. Doch obwohl diese Technologie in der Forschung schon länger behandelt wird, ist noch keine allgemeingültige Lösung entwickelt worden. Das ist auch der Grund, warum

auf dem Markt wenige ausgereifte Lösungen zu finden sind, die Disaggregation einsetzen. Begünstigt wird die Entwicklung zurzeit durch das Ausrollen der Smart Meter und damit dem Bereitstellen der Daten, die die Voraussetzung für diese Technologie sind.

Die Disaggregation ist aber auch von verschiedenen Faktoren abhängig, die für die Entwicklung der Technologie hinderlich sind. Die Ergebnisse und damit auch das Potential der Technologie ist direkt an die Frequenz der Verbrauchsdaten gekoppelt. Je höher die Auflösung ist, umso mehr Geräte lassen sich erkennen und umso besser sind die Ergebnisse allgemein. Aktuelle Stromzähler liefern nur Daten in einem Frequenzbereich von etwa 0,5 bis 1 Hertz, was nur die Erkennung weniger großer Verbraucher ermöglicht. Diese Zahl wird maßgeblich durch Datenschutzregelungen bestimmt und stellt damit einen klaren Hemmfaktor für diese Technologie dar.

Der Prototyp zeigt, dass es möglich ist ein Anwendungsszenario mit Disaggregation in einem Prototyp umzusetzen. Dieser Prototyp funktioniert ohne teure Anschaffungen und setzt auf vorhandenen Schnittstellen von herkömmlichen digitalen Stromzählern an. Doch es zeigt sich auch, dass Ergebnisse die mit einem solchen Modell erreicht werden können, stark durch die vorhandene Datenbasis eingeschränkt werden. Zum einen sorgt die niedrige Aufnahmefrequenz dafür, dass nur große Verbraucher mit klarem Profil erkannt werden können. Die Voraussetzung ist dabei auch das Trainieren des Algorithmus mit spezifischen Verbrauchsdaten für eine geringe Zahl von Geräten. Zum anderen ist das Erstellen und Spezifizieren von Datensätzen unter den gegebenen Voraussetzungen aufwendig und langwierig.

Die dazugehörige Forschungsarbeit setzt bei der Frage an, welche der Informationen die durch einen solchen Prototypen erzeugt wurden, für die Visualisierung bei einem Endkunden nutzbringend sind. Dafür wurden semistrukturierte Interviews mit verschiedenen Personen durchgeführt, die eine gewisse Affinität zum Thema Energie besitzen. Dabei zeigte sich, dass Informationen einfach und strukturiert in einer mobilen Applikation dem Konsumenten zugänglich gemacht werden sollten.

Ausblick Der Fokus dieser Arbeit lag auf der Umsetzung eines Prototyps für einen gesamten Disaggregationsprozess von Stromzähler bis zum Kunden. Aufgrund der eingeschränkten Datenbasis sollten in einem weiteren Schritt nun eine Evaluierung des Prototyps mit veränderter Grundlage durchgeführt werden. Dazu gehört eine hohe Datenfrequenz, Daten eines ganzen Tages von einem realen Stromzähler, mehr Daten von verschiedenen Geräten und Haushalten. Des Weiteren müssen noch weitere Schritte umgesetzt werden, damit der Prototyp kontinuierlich aktiv sein kann. Die entsprechenden Konzepte wurden zwar entwickelt, aber konnten im Rahmen des Projektes noch nicht umgesetzt werden.

In der Forschung gibt es viele Ansätze, die gute Ergebnisse bei der Disaggregation von Lastkurven liefern. Diese brauchen eine hohe Frequenz der Verbrauchsdaten. Um die Technologie weiter voranzutreiben muss an der Datenbasis generell gearbeitet werden. Entweder müssen die Voraussetzungen mit höheren Abstraten an den Smart Metern geschaffen werden, sodass die Algorithmen diese hochfrequenten Daten einsetzen können. Oder es müssen weitere Lösungen entwickelt werden, die es ermöglichen die anfängliche Hürde der fehlenden Datenbasis zu überwinden, um auch mit geringen Frequenzen gute Ergebnisse erzielen zu können.

6. Energy of TOMORROW

In diesem Kapitel werden das Vorgehen und die Ergebnisse der Teilgruppe *energy of TOMORROW* im Rahmen der Predictive Analytics dokumentiert. Nach einleitenden Worten, welche die Motivation, Problemstellung und Zielsetzung beinhalten, erfolgt die Vorstellung des Konzepts zur Vorgehensweise und die Auswahl der Technologiethoden. Der Kern dieses Kapitels befasst sich mit der Entwicklung, Implementierung und Evaluation der Prognosemodelle, sowohl für den Energieverbrauch als auch für die -produktion und die technische Umsetzung. Abschließend erfolgt ein Fazit und ein Ausblick für weitere mögliche Arbeiten.

6.1. Motivation

Bereits im Kapitel 2 wurde das Thema Energiewende ausführlich behandelt. Aus diesem Grund ist bekannt, dass aufgrund der Installation der intelligenten Stromzähler und der dadurch feingranularen Datenaufzeichnung, neue Anwendungsfälle im Bereich der Informationstechnologie entstehen.

In diesem Kontext gewinnt in der Energiebranche das Thema *Data Analytics*, bei dem die Datenmodellierung, Transformation und Gewinnung von neuen Zusammenhängen mit dem Ziel, werttreibende und nützliche Informationen zu finden, an zunehmender Bedeutung. Z. B. lassen sich Energieverbräuche und -handelsgeschäfte durch genauere Informationen optimieren. In diesem Kontext spielt die Begrifflichkeit *Predictive Analytics* eine wichtige Rolle. Predictive Analytics ist ein Oberbegriff, der statistische und analytische Modellierungsmethoden umfasst, mit deren Hilfe datengetriebene Vorhersagemodelle entwickelt werden können [Chr15].

Prosumer spielen aktuell ebenfalls eine wichtige Rolle in der Energiebranche. Zentraler Treiber ist die Weiterentwicklung von Systemen, die es Haushalten ermöglichen, regenerativ selbstständig Strom zu erzeugen. Somit würden diese nicht nur als Konsumenten gelten, sondern auch als Produzenten, die sogenannten Prosumern [Ins]. Nicht nur die Energiewende und das Erneuerbare-Energien-Gesetz bestimmen den Umgang mit produzierter Energie [Kla16]. Der Einbau intelligenter Stromzähler(Smart Meter) in allen Haushalten, bei denen das Einverständnis zur Nutzung von personenbezogenen Energiedaten benötigt wird [Kon12], oder die Verpflichtung, dass erneuerbare Energien im Wärmebereich eines Neubaus eingesetzt werden müssen [Bun16a], sind wesentliche Aspekte.

6.1.1. Problemstellung und Forschungsfrage

Der allgemeine Konsens im deutschen Datenschutz zielt darauf ab, die informationelle Selbstbestimmung zu gewährleisten [BMI16]. In diesem Zusammenhang werfen zukünftige Szenarien mit intelligenten Stromzählern Fragen auf.

Wie bereits aus dem Kapitel 2 bekannt wird der Zählerstand derzeit manuell erfasst, sei es durch die elektronische Übermittlung vom Verbraucher oder durch das Ablesen eines Mitarbeiters des Stromversorgers. Jedoch ist die Ablesefrequenz zu gering, um Informationen mit Mehrwert aus den Verbrauchszahlen zu gewinnen. Durch die intelligenten Stromzähler, die in einer sehr hohen Frequenz Verbrauchsdaten sowie Erzeugungsdaten, z. B. durch Solaranlagen auf dem Dach, erfassen können und digital speichern, kann dieser Mehrwert geschaffen werden [Oli16]. Für den Prosumer stellt sich die Frage, welche Vorteile er aus dem hochfrequentierten Messen seines Energiekonsums und seiner Energieproduktion ziehen kann [Ste16a]. Durch die intelligenten Stromzähler und die Übermittlung dieser Daten an den Prosumer können Kostentransparenz sowie eine Steigerung der Energieeffizienz geschaffen werden, beispielsweise durch das Identifizieren von Lastspitzen [Oli16]. Ebenso können aus den Daten Prognosen über den zukünftigen Stromverbrauch und -erzeugung getroffen werden.

Auf Grundlage der erzielten Ergebnisse könnte der Verbraucher seine Energiehaushaltsplanung verbessern. So können hypothetisch anhand einer Prognose über den Stromverbrauch energieintensive Geräte identifiziert werden, welche beispielsweise in der Nacht für hohen Verbrauch sorgen. Zudem kann die Prognose des Stromverbrauchs für Prosumer interessant werden, um zu überprüfen, ob der Verbrauch durch die eigene Erzeugung gedeckt werden kann. Das Delta vom Verbrauch und der Erzeugung stellt damit den Strom dar, der verkauft oder hinzugekauft wird. Jedoch kommt auch Kritik auf, denn durch die Daten können Lebensmuster des Konsumenten theoretisch identifiziert werden [Oli16].

Aufgrund der stark wachsenden Bedeutung von erneuerbaren Energien und die Erstellung von Prognosen, ergibt sich die Herausforderung diese entsprechend zu berechnen und eine Planungsgrundlage für den Verbrauch und die Erzeugung von Eigenenergie beim Prosumer zu schaffen. Anhand des vorab gegebenen Kontextes lassen sich folgende Forschungsfragen ableiten:

- **FF1:** Mithilfe welcher Methodologie und Datengrundlage lässt sich der Stromverbrauch auf Monats-, Wochen-, Tages- und Stundenbasis vorhersagen?
- **FF2:** Wie lassen sich mehrere Verbrauchsdaten von Haushalten zusammenfügen und mithilfe welcher Herangehensweise und welchen externen Datenquellen lässt sich die

Energieverbrauchs- und -erzeugungsprognose optimieren?

6.1.2. Zielsetzung

Ziel des Teilprojektes *energy of TOMORROW* ist es, auf Basis identifizierter Trends geeignete Energieverbrauchs- und -erzeugungsprognosen auf Tagesbasis für private Haushalte zu erstellen. Zusätzlich zu dieser Verbrauchsprognose soll dem Haushalt, beispielsweise mit vorhandener Photovoltaik-Anlage, mitgeteilt werden, ob der Verbrauch mit der eigens erzeugten Energie gedeckt werden kann. Dafür sollen zunächst die durch den Smart Meter aufgezeichneten Daten aggregiert werden. Das bedeutet, dass Daten, welche beispielsweise im 15-Minuten-Takt ausgelesen wurden, zu einer größeren Einheit zusammengefasst werden. Im Anschluss daran werden die Prognosen mithilfe statistischer Modelle und Methoden des Data Minings optimiert. Faktoren wie Feiertags-, Ferien- oder Wetterdaten, die die Prognose beeinflussen, sollen bei der Analyse berücksichtigt werden.

Der Nutzen der genannten Ziele besteht darin, es dem Verbraucher zu ermöglichen, effizienter mit der Ressource Strom umzugehen. Durch diese Umsetzung sind weitere Möglichkeiten vorhanden. Beispielsweise kann der Verbraucher Strom bestellen, wenn die Prognose für den nächsten Tag nicht positiv ausfällt, also der Verbrauch nicht durch die eigen erzeugte Energie gedeckt werden kann. Hier könnte es die Möglichkeit geben, dass Strom bei einem Anbieter erworben werden kann. Die Transaktion könnte über eine Blockchain abgewickelt werden.

Zusammenfassend lassen sich folgende Ziele aufzählen:

- Energieverbrauchs und -erzeugungsprognosen für Haushalte erstellen
- Berücksichtigung verschiedener Faktoren bei der Prognoseerstellung
- Optimierung der Prognosen anhand linearer Regression und Methoden des Data Minings (z. B. nach Shearer [Col00])

6.2. Vorgehensweise

Um die im vorherigen Abschnitt beschriebenen Ziele zu erreichen, wurden allgemein folgende Schritte eingeleitet:

1. Recherche
2. Data Mining
3. Einbindung eines Servers

4. Erstellung eines Dashboards

6.2.1. Recherche

Bei der Recherche waren die Ziele, sich in die Thematik einzulesen und für die meisten Projektmitglieder in die neue Programmiersprache R und zugehörigen Tools einzuarbeiten. Dazu gehörte auch die Einarbeitung in statistische Auswertungs- und Analysemethoden.

State of the Art / Verwandte Arbeiten In der Energiebranche gewinnt das Thema *Predictive Analytics* an zunehmender Bedeutung. Wissenschaftliche Arbeiten befassen sich insbesondere mit der Optimierung der Prognosemethoden und Modellierungstechniken für den Energiebedarf.

Das Ziel einer Zusammenarbeit zwischen der MONASH University und verschiedenen australischen Unternehmen, war die Verbesserung der Prognosemethoden für den Energiebedarf. Neben den üblichen Verbrauchswerten der Nutzer wurden zusätzlich noch Wetterdaten herangezogen, um so die Prognosen noch präziser zu gestalten. Zunächst wurde ein Bedarfsmodell erstellt, in welchem die folgenden Faktoren mit einbezogen wurden: Temperatur, ökologische und demographische Variablen. [Rob15] Der Ansatz, Wetterdaten bei der Prognose mit einzubeziehen, wird für die Erstellung der Prognosemodelle übernommen. Die Aussicht besteht darin, dass dadurch bessere Prognosen für den Konsum und die Erzeugung erzielt werden können. In beiden Fällen könnten zwei verschiedene Annahmen getroffen werden: Der Konsum ist höher, wenn das Wetter schlecht ist bzw. niedriger, wenn es gut ist. An sonnigen Tagen kann über Photovoltaik-Anlagen mehr Energie erzeugt werden bzw. an nicht sonnigen Tagen weniger.

In [Geo07] beschreiben Tso und Yau verschiedene Modellierungstechniken, welche zur Vorhersage des Stromverbrauchs genutzt werden können. Neben der klassischen Regressionsanalyse wurden ebenfalls Entscheidungsbäume und neuronale Netze näher betrachtet. Das Ergebnis ist, dass beide Alternativen für das Verständnis von Energieverbrauchsmustern und Vorhersagen des Energieverbrauchs brauchbar sind. Aufgrund dieser Tatsache werden für das Teilprojekt auch Algorithmen aus dem Bereich der Künstlichen Intelligenz implementiert und evaluiert.

6.2.2. Data Mining

Als Vorgehensweise für den Data Mining-Prozess wurde das Vorgehensmodell CRISP-DM (Cross-Industry Standard Process for Data Mining) gewählt. Grund für diese Entscheidung war, dass dieses Modell eine leitende und agile Vorgehensweise bietet und somit durch neue

und laufende Erkenntnisse gleichzeitig neue Fragestellungen und Probleme identifiziert werden können. Der Lebenszyklus eines Data Mining-Prozesses besteht aus sechs verschiedene Phasen, welche in der Abbildung 48 zu sehen sind. Zwischen den einzelnen Phasen ist eine gewisse Dynamik zu erkennen. Das bedeutet, dass die Phasen nicht starr nach einer bestimmten Reihenfolge bearbeitet werden, sondern ein Zurückspringen zu einer vorherigen Phase möglich und vorgesehen ist. Die Pfeile stellen die Abhängigkeiten der verschiedenen Phasen dar [Col00].

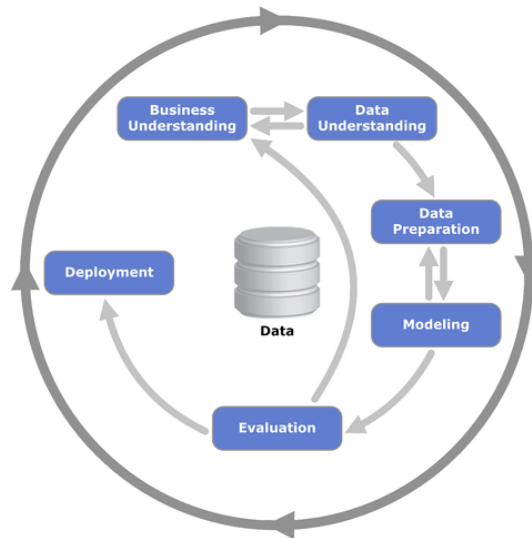


Abbildung 48: CRISP-DM-Modell nach [Wol12]

Business Understanding Bei der Phase Business Understanding geht es primär um die Festlegung von Zielen und Anforderungen. Aus diesen lassen sich die konkrete Aufgabenstellung und eine grobe Vorgehensweise ableiten. Letzteres lässt sich durch einen Projektplan, welcher Meilensteine enthält, darstellen [Col00]. Diese Phase wird in dieser Dokumentation mit der vorangegangenen Zielsetzung und Kozeptionierung der Vorgehensweise behandelt.

Data Understanding Die zweite Phase beginnt, sobald mit der ersten Datenerhebung begonnen wurde bzw. mit der Sichtung dieser. Dabei werden die Daten hinsichtlich Menge, Datenformat, Typen, Werte und Anzahl der Attribute untersucht. Wichtigster Aspekt ist die Ermittlung der Datenqualität und daraus resultierende Probleme, welche erkannt und behoben werden müssen [Col00].

Data Preparation Die Aufgaben dieser Phase umfassen die Auswahl der Tabellen, Datensätzen und Attribute, sowie Umwandlung und Reinigung von Daten für die Modellierung [Col00].

Bei der Data Preparation werden die ursprünglichen Rohdaten, entsprechend der Zielsetzung, um die benötigten Daten erweitert, indem Daten aus verschiedenen Tabellen zusammengeführt werden. Der erste Schritt ist die Auswahl der wichtigsten Variablen. Danach erfolgt die Bereinigung der Daten, wobei Ausreißer erkannt und fehlende Werte behandelt werden sollen [Col00].

Diese Phase ist sehr zeitintensiv und kann häufig bis zu 90 % der Gesamtzeit ausmachen. Daher ist mit hoher Wahrscheinlichkeit davon auszugehen, dass die Datenaufbereitung mehrmals durchgeführt werden muss bis ein optimales Ergebnis (entsprechend der Zieldefinition) erzielt wird [Kat08].

Modelling Um mit dem Modelling zu beginnen, werden zunächst verschiedene Techniken untersucht, ausgewählt und angewandt. Die benötigten Parameter sind dabei auf optimale Werte kalibriert. Es gibt eine Vielzahl von verschiedenen Modellierungstechniken, welche für identische Probleme angewendet werden können. Die Unterschiede der einzelnen Techniken bestehen in den speziellen Anforderungen an die Form der Daten. Hierbei ist es oft der Fall, dass nochmals mit der Daten Vorbereitung begonnen wird und Anpassungen vorgenommen werden [Col00].

Evaluation Nachdem in der vorherigen Phase das Modell erstellt wurde, wird dieses nun im Hinblick auf die Qualität der Datenanalyse und deren Perspektive evaluiert.

Bevor das endgültige Modell eingeführt wird, ist zu untersuchen, ob alle zuvor definierten Ziele mit den erlangten Resultaten erreicht wurden. Im Anschluss daran werden alle Schritte im Hinblick auf die Daten begutachtet. Wichtig hierbei ist festzustellen, ob nur Daten verwendet wurden, welche auch in Zukunft zur Verfügung stehen und ob wichtige Aspekte übersehen wurden. Zum Schluss werden die nächsten Schritte definiert. Das kann sowohl das Zurückgehen zu einer vorherigen Phase oder der Übergang zur nächsten Phase, der Einführung des Modells, bedeuten [Col00].

Deployment Die letzte Phase des Vorgehensmodells beschreibt den Einsatz des erarbeiteten Modells. Das Projektende wird nicht durch das endgültige Modell markiert, welches in Phase fünf erstellt und in der vorangegangenen Phase getestet wurde, auch wenn die Erstellung das primäre Ziel war.

Hierbei geht es um die Anwendung der Modelle in der Praxis. Dazu zählt beispielsweise die möglichen Implementierungsstrategien innerhalb einer Organisation oder Ähnliches und gegebenenfalls auch Mitarbeiterschulungen. Zudem wird permanent die Gültigkeit des Modells in der Realität überwacht und Strategien für die Überarbeitung ausgearbeitet.

Im letzten Schritt wird zunächst ein vollständiger Bericht angefertigt, welcher das komplette Modell beschreibt. Im Anschluss daran erfolgt das sogenannte Review. Hierbei wird das gesamte Projekt reflektiert, um herauszufinden was positiv oder negativ verlief und welche Erfahrungen für das nächste Projekt mitgenommen werden können [Col00]. Da es sich bei den erstellten Modellen ausschließlich um Prototypen handelt, wird diese Phase im weiteren Verlauf nicht betrachtet werden.

6.3. Konzept

Nachdem die Grundlagen geschaffen wurden, geht es um die Erstellung eines Konzepts für die weiteren Arbeiten. Zu Beginn mussten verschiedene Anwendungsfälle ausgearbeitet werden.

6.3.1. Anwendungsfälle

Hauptakteure für die Teilgruppe sind die Energiekonsumenten bzw. einzelne private Haushalte, in denen Energie verbraucht wird. Diese sollen zukunftsbezogene Verbrauchsdaten anhand von Prognosen erhalten. In der folgenden Abbildung 49 werden identifizierte Anwendungsfälle anhand eines Use Case-Diagramms in UML (Unified Modeling Language) veranschaulicht.

Die identifizierten Anwendungsfälle werden im Folgenden beschrieben.

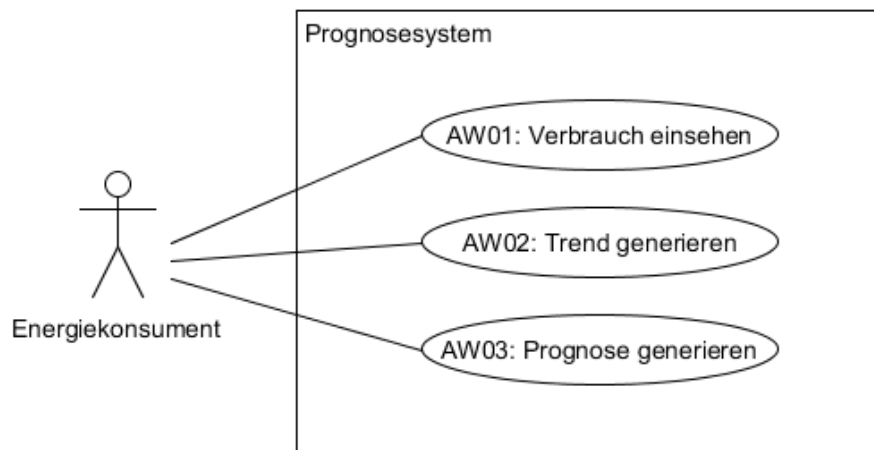


Abbildung 49: Anwendungsfalldiagramm

AW01: Verbrauch einsehen

Anwendungsfall	AW01: Verbrauch einsehen
Akteure	Energiekonsument
Ziel	Die Energieverbrauchsdaten werden dem Energiekonsumenten angezeigt.
Vorbedingung	Historische Energieverbrauchsdaten sollen im Prognosesystem geladen werden.
Nachbedingung im Erfolgsfall	Der Energiekonsument sieht die Energieverbrauchsdaten ein.
Fehler-/ Ausnahmefälle	<ol style="list-style-type: none"> 1 Daten können nicht geladen werden 2 Verbindungsfehler zum Smart-Meter 3 Fehlende Werte

Nachbedingung im Fehler-/Ausnahmefall	<ul style="list-style-type: none"> - zu 1 & 2: Daten werden erneut in das Prognosesystem geladen. - zu 3: Fehlende Werte werden ergänzt.
Varianten	Verbrauchsdaten pro Stunde, Tag, Woche, Monat oder Jahr

AW02: Trend generieren

Anwendungsfall	AW02: Trend generieren
Akteure	Energiekonsument
Ziel	Die Energieverbrauchsdaten werden in aggregierter Form anhand eines Trends dargestellt.
Vorbedingung	Historische Energieverbrauchsdaten sollen im Prognosesystem geladen werden.
Nachbedingung im Erfolgsfall	Der Energiekonsument hat sich einen Trend anhand der Verbrauchsdaten generieren lassen.
Fehler-/ Ausnahmefälle	<ol style="list-style-type: none"> 1 Daten können nicht geladen werden 2 Verbindungsfehler zum Smart-Meter 3 Fehlende Werte
Nachbedingung im Fehler-/Ausnahmefall	<ul style="list-style-type: none"> - zu 1 & 2: Daten werden erneut in das Prognosesystem geladen. - zu 3: Fehlende Werte werden ergänzt.
Varianten	Trend pro Stunde, Tag, Woche, Monat oder Jahr

AW03: Prognose generieren

Anwendungsfall	AW03: Prognose generieren
Akteure	Energiekonsument
Ziel	Der Energieverbraucher soll sich anhand der Verbrauchsdaten eine Verbrauchs-Prognose generieren können.
Vorbedingung	Historische Energieverbrauchsdaten sollen im Prognosesystem geladen werden.
Nachbedingung im Erfolgsfall	Der Energiekonsument hat sich eine Prognose anhand der Verbrauchsdaten generieren lassen.
Fehler-/ Ausnahmefälle	<ol style="list-style-type: none"> 1 Daten können nicht geladen werden 2 Verbindungsfehler zum Smart-Meter 3 Fehlende Werte
Nachbedingung im Fehler-/ Ausnahmefall	<ul style="list-style-type: none"> - zu 1 & 2: Daten werden erneut in das Prognosesystem geladen. - zu 3: Fehlende Werte werden ergänzt.
Varianten	Prognose für die nächste Stunde, die nächste Woche oder den nächsten Monat

6.3.2. Grobarchitektur

Um den Gesamtbedarf an Strom des nächsten Tages vorherzusagen, erfolgt nach der Erstellung des statistischen Modells, die performante Implementierung. Dazu wird in unserem Anwendungsfall die SAP HANA-Appliance genutzt, die zum einen durch das DBMS, als auch durch die R-Integration die nötigen Voraussetzungen bietet.

Zu Beginn werden die Daten von den Smart Metern, die bei den Konsumenten installiert sind, zentral im DBMS der SAP HANA gespeichert. Im Anschluss werden die Daten durch die R-Integration präpariert und für das statistische Modell in eine entsprechende Struktur gebracht und zusätzliche Kennzahlen berechnet. Das Modell ist via PMML-Datei (Predictive Model Markup Language) in der R-Integration gespeichert und wird zum Vor-

hersagen der nötigen Kennzahl genutzt. Die Vorhersagen werden zurück zum Gerät des Konsumenten geschickt, sodass dieser die Vorhersage für heterogene Anwendungsszenarien einsetzen kann. In der folgenden Abbildung 50 wird die Grobarchitektur veranschaulicht.

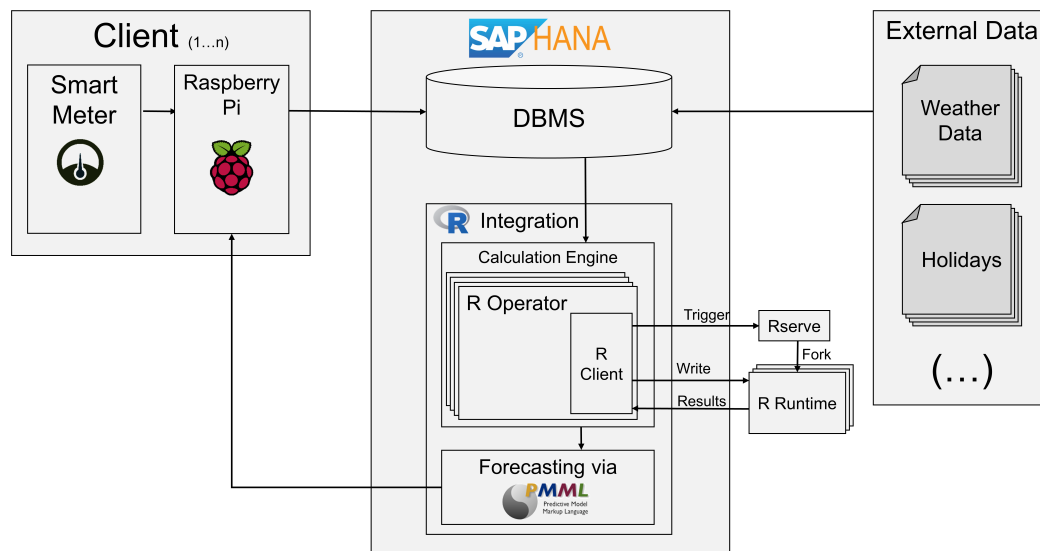


Abbildung 50: Grobarchitektur

Weitere Beschreibungen zur SAP HANA und Implementierung erfolgen im Kapitel 6.6 Technische Umsetzung.

6.3.3. Analyse-/Prognose-Methoden

Bevor mit der Analyse der Daten begonnen wird, muss zunächst eine geeignete Prognose-Methode erarbeitet werden. Da in den Anforderungen eine Prognose eines numerischen Wertes ohne Klassifikationen verlangt wird, ist in diesem Fall die multiple lineare Regression eine passende Methode. Diese kann einen Wert auf Grundlage von mehreren Faktoren vorhersagen.

Diese Faktoren können zum einen Daten sein, die in den Verbrauchsdaten enthalten sind, wie zum Beispiel Zeitstempel und somit den Wochentag und den Monat, außerdem statistisch abgeleitete Faktoren, wie der gleitende Durchschnitt über die letzten sieben Tage, oder auch externe Daten. Zu den externen Daten zählen beispielsweise Wetterdaten oder Ferien- und Feiertage. Voraussetzung für die Einbindung dieser Daten ist jedoch, dass in den Verbrauchsdaten Informationen über den Standort enthalten sind.

Um Interferenzen zwischen dem zu untersuchenden Faktor und anderen Faktoren zu überprüfen, werden zunächst mit dem Ansatz der explorativen Datenanalyse graphische

Darstellungen erstellt, die unter anderem Trends verdeutlichen. Somit können Saisonalitäten innerhalb einer Periode ermittelt werden, die mithilfe von einer Sinus- oder Cosinus-Kurve abgebildet werden können.

Nach der Evaluation der Faktoren, unter anderem auch durch die Residualanalyse, die Aufschluss über die Richtigkeit des linearen Zusammenhangs gibt, wird die Auswahl der Faktoren zum Erstellen des Modells genutzt.

Das Ergebnis der Evaluation des statistischen Modells ist ein R^2 -Wert, der die Quote der richtig vorhergesagten Werte angibt. Dementsprechend ist der beste Wert hierbei eins.

6.4. Technologieauswahl

Für die Erstellung der Prognosen wurde die Open-Source Software bzw. Programmiersprache *R* ausgewählt. Diese bietet im Rahmen von Big Data eine gute Möglichkeit zur Erstellung statistischer Analysen und lässt sich größtenteils von Anwendern erweitern. Diese Erweiterungen, welche als *packages* bezeichnet werden, sind für jeden im *Comprehensive R Archive Network (CRAN)* frei zugänglich. Die Vorteile von *R* im Überblick [Seb10]:

- kontinuierliche Verbesserung von *R* durch ein festes Team von Entwicklern
- eine Vielzahl von Erweiterungen, welche durch eine ständig wachsende *R*-Community erstellt werden, sind frei zugänglich
- Erweiterungen können von dem Nutzer selbst erstellt werden

Neben der Auswertung der Daten bietet *R* auch eine Visualisierung der Daten. Hierbei können qualitativ hochwertige Grafiken direkt in der *R*-Syntax erstellt werden und sind somit reproduzierbar. Dabei wird eine Vielzahl von verschiedenen Grafiktypen geboten [Seb10].

6.5. Entwicklung Verbrauchs- und Produktionsmodell

In diesem Kapitel steht die Entwicklung des Verbrauchs- und Produktionsmodells im Vordergrund. Für die Erstellung der statistischen Modelle wird auf verschiedene Algorithmen aus der Bereich der künstlichen Intelligenz zurückgegriffen.

Des Weiteren werden verschiedene Machine Learning-Ansätze beschrieben, die für die Prognosemodelle umgesetzt wurden. Ebenfalls erfolgt die Erläuterung, wie diese bewertet werden können und wie ein intelligenter Algorithmus implementiert werden könnte. Abschließend folgt die Beschreibung der Implementierung eines Servers, auf dem die Daten gesammelt und Prognosen erstellt werden.

Für die Erstellung des Verbrauchs- und Produktionsmodells wurde nach den bereits vorgestellten CRISP-DM-Phasen vorgegangen. Die Phase Business Understanding wurde bereits im Kapitel 6.1.2 Zielsetzung beschrieben und daher mit der Phase *Data Understanding* begonnen.

6.5.1. Data Understanding

Datensammlung Nach der Phase des Business Understandings werden die zur Verfügung gestellten Daten inspiziert und verstanden. Aufgezeichnete Smart Meter-Daten verschiedener Haushalte wurden untersucht.

Zielformulierung Ziel ist es in dieser Phase des CRISP-DM-Modells die Daten so weit zu verstehen, dass im nächsten Schritt eine genaue Vorbereitung der Datensätze erfolgen kann, um die darauffolgenden Schritte der genauen Analyse der Daten bzw. der Modellierung auszuführen. Folgende Punkte sollen anhand der Datensätze herausgefunden werden:

- Welche Daten liegen genau vor?
- Wie viele und welche Datentypen?
- Können schon Aussagen aus diesen abgeleitet werden?
- Welche Potentiale bieten die Daten?

Datenbeschreibung - deutsche Smart Meter-Daten Zu Beginn der Projektphase lagen lediglich die Energieverbrauchs- und -produktionsdaten eines Haushaltes aus Niedersachsen im 15-Minuten-Takt vor. Dieser beinhaltet insgesamt 36.510 Datensätze im Zeitraum vom 16. September 2015 bis einschließlich dem 30. September 2016. Im weiteren Projektverlauf kamen weitere neun Energiedaten von Haushalten hinzu. Genauere Hintergrundinformationen lagen zu diesem Zeitpunkt nicht vor.

Die Datensätzen enthalten die folgenden Informationen: Smart Meter IDs, Zeitstempel, Energieverbräuche in kWh und Energieproduktionen in kWh. Der Energieverbrauch wurde im 15-Minuten-Takt aufgezeichnet. Folgende Datentypen sind vorhanden: numeric-, date-, time-, int- und factor-Datentypen. Allgemein beschreiben die Datensätze den elektrischen Jahresverbrauch und -produktion eines Haushaltes in Niedersachsen im 15-Minuten-Takt. Für die Datenqualitätssicherung wurden die vorliegenden Datensätze auf Korrektheit, Vollständigkeit und Redundanz geprüft.

Anhand verschiedener statistischer Darstellungsmöglichkeiten, wurden die Daten für ein übersichtliches und anschauliches Datenverständnis aufgezeigt. Es folgt eine beispielhafte

Darstellung (Abbildung 51) des Monatskonsums in einem Haushalt innerhalb eines Jahres (ohne Ausreißerdatensätze).

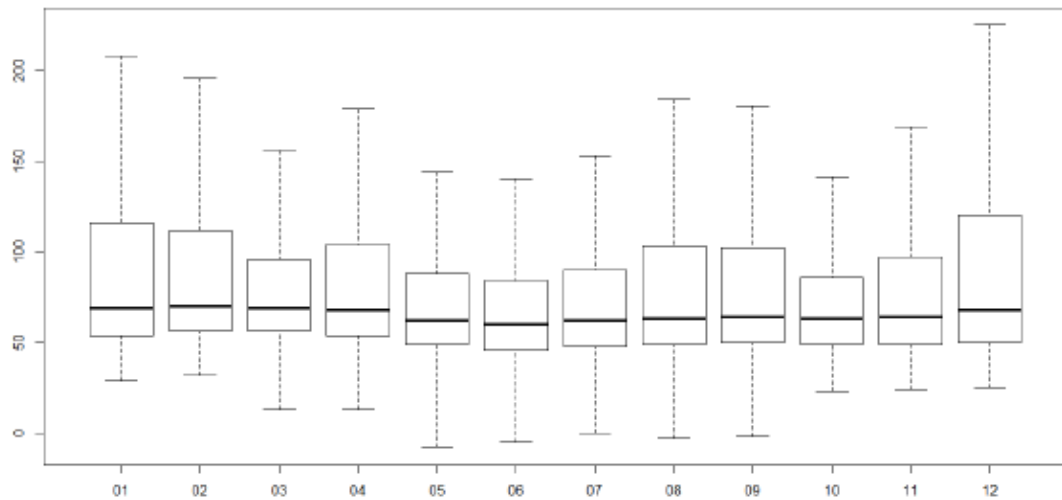


Abbildung 51: Monatskonsumdaten

Im folgenden Box-Plot sind die Wochenkonsumdaten (Abbildung 52) in einem Haushalt innerhalb eines Jahres (ohne Ausreißerdatensätze) zu entnehmen.

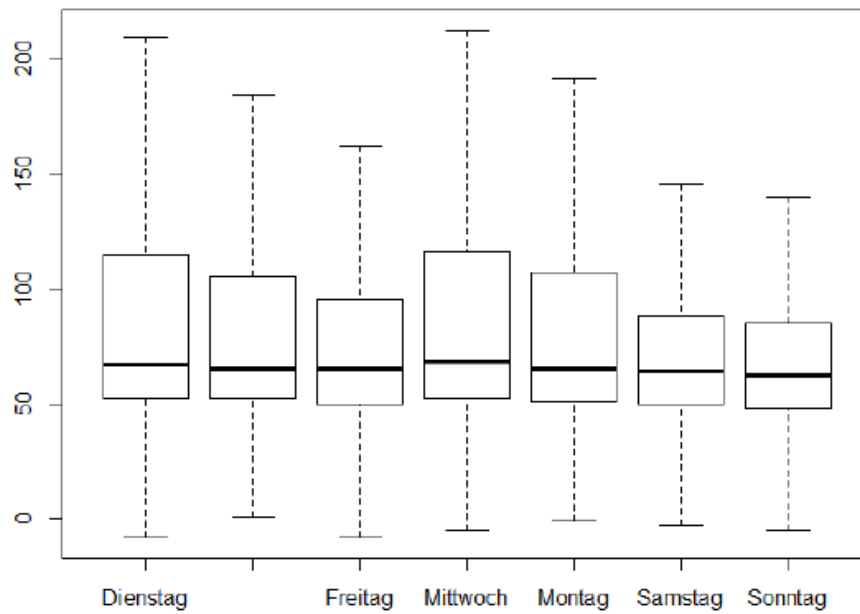


Abbildung 52: Wochenkonsumdaten

Schließlich folgt die Darstellung (Abbildung 53) zum Stundenkonsum in einem Haushalt innerhalb eines Jahres (ohne Ausreißerdatensätze).

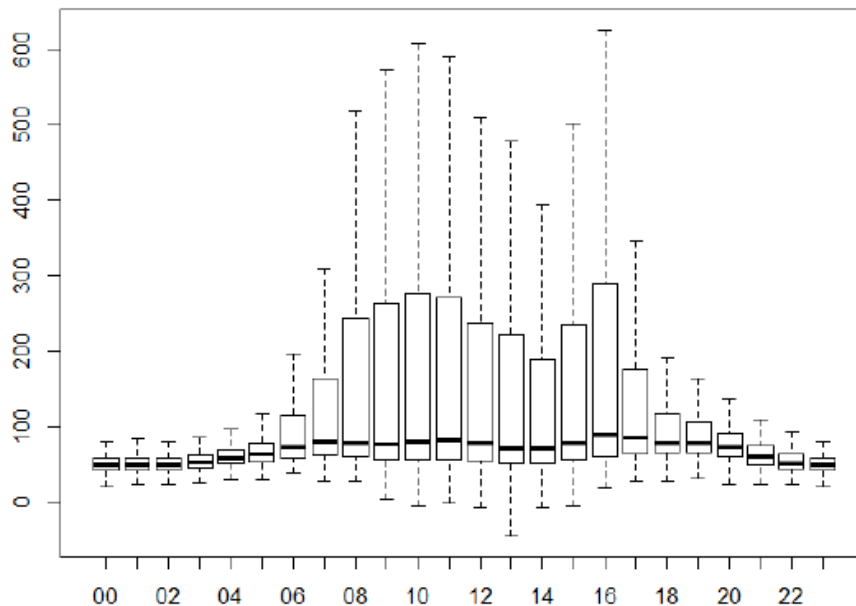


Abbildung 53: Stundenkonsumdaten

Weiter wurden die Daten bezüglich ihrer Länge (`length()`), Mittelwerte (`mean()`), Mediane (`median()`), Quartile (`quartile()`), usw. untersucht.

CER - Commission for Energy Regulation Ab der Hälfte der Projektlaufzeit wurden Smart Meter-Daten vom CER (Commission for Energy Regulation) aus Irland (Dublin) zur Verfügung gestellt, welche ausschließlich für das Verbrauchsmodell genutzt wurden. CER ist eine unabhängige Organisation für die Regulierung elektrischer Energie, Gas und Wassersystemen in Irland. CER bringt verschiedene Paper zu Forschungs- und Praxisarbeiten zum Thema Energie und Energiewende heraus. Diese Organisation führt verschiedene Projekte im Rahmen der Einführung des Smart Meterings durch [CER] [ISS].

Im Rahmen des CER-Projekts Smart Metering Electricity Customer Behaviour Trials (CBTs) wurden 5.000 Smart Meter Installationen von verschiedenen anonymen Haushalten in Irland gesammelt (größtenteils Gewerbe). Die genauen Haushaltseigenschaften sind entsprechend nicht bekannt. Die Datenaufzeichnung fand zwischen 2009 und 2010 statt. Insgesamt liegen 3.291.252 Smart Meter-Messwerte als .csv vor (vom 14. Juli 2009 bis 31. Dezember 2010). Zu den Teilnehmern gehörten z. B. Sustainable Energy Authority of Ireland (SEAI), Department of Communications, Energy and Natural Resources (DCENR), ESB Networks, Bord Gáis Networks Electric Ireland und andere Energielieferanten.

Mit der Datenaufnahme bzw. dieses Projekts soll herausgefunden werden, wie Smart Metering dabei helfen kann, das Verhalten von Energieverbräuchen zu modellieren (verschiedene demographische Aspekte, Lebensstile und Haushaltsgrößen) [CER] [ISS].

Externe Daten Neben den eigentlichen Verbrauchs- und Produktionsdaten wurden zusätzlich Wetterdaten hinzugezogen werden. Für das **Verbrauchsmodell Deutschland**, welches die deutschen Daten als Datenbasis nutzt, wurden neben den Wetterdaten, welche vom DWD¹³ stammen, noch zusätzlich Ferien- und Feiertagsdaten mit eingebunden. Insgesamt werden die folgenden Faktoren für das Verbrauchsmodell genutzt:

- Temperatur
- Luftfeuchtigkeit
- Ferien- und Feiertagsdaten

Bei dem **Irland Verbrauchsmodell** sollen ebenfalls diese Faktoren genutzt werden. Dazu wurden die **irischen Ferien- und Feiertagsdaten** verwendet. Datenquelle für die irischen Ferientage war der internationale Feiertagskalender im Internet¹⁴. Die irischen Feiertage wurden ebenfalls aus dem internationalen Feiertagskalender¹⁵ entnommen. Die Daten wurden anschließend in einer csv-Datei manuell zusammengestellt.

Da es keine zentral frei zugängliche Datenquelle der Wetterstationen für irisches Festland gab, wurden irische Marine Boje-Stationen für die **irischen Wetterdaten** in Betracht gezogen. Diese verfügen über die Jahre aufgezeichnete Wetterdaten. Diese stehen im Internet¹⁶ zum Download unter Angabe des gewünschten Zeitraums als csv-Datei bereit.

In der folgenden Abbildung 54 sind die Marine-Stationen kenntlich gemacht. Einige Stationen waren jedoch nicht auswertbar, weil diese für den zu untersuchenden Zeitraum keine Messwerte aufgewiesen haben (M1, M4 Original, FS1, AMETS und M6).

¹³ ftp://ftp-cdc.dwd.de/pub/CDC/observations_germany/climate/hourly

¹⁴ <https://www.feiertagskalender.ch/ferien.php?geo=3446&jahr=2010>

¹⁵ <https://www.feiertagskalender.ch/index.php?geo=3446&jahr=2009&klasse=4>

¹⁶ <http://data.marine.ie/Dataset/Details/20972>

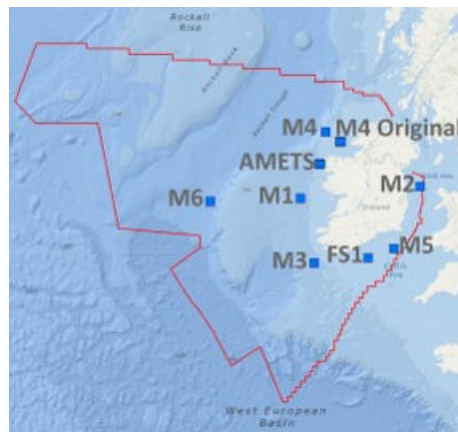


Abbildung 54: Marine Boje Stationen [Mar]

Es folgt in Abbildung 55 ein Plot der Irland-Verbrauchsdaten mit Markierungen zu Ferientagen. Die Ferien sind farblich unterschieden und der Legende zu entnehmen.

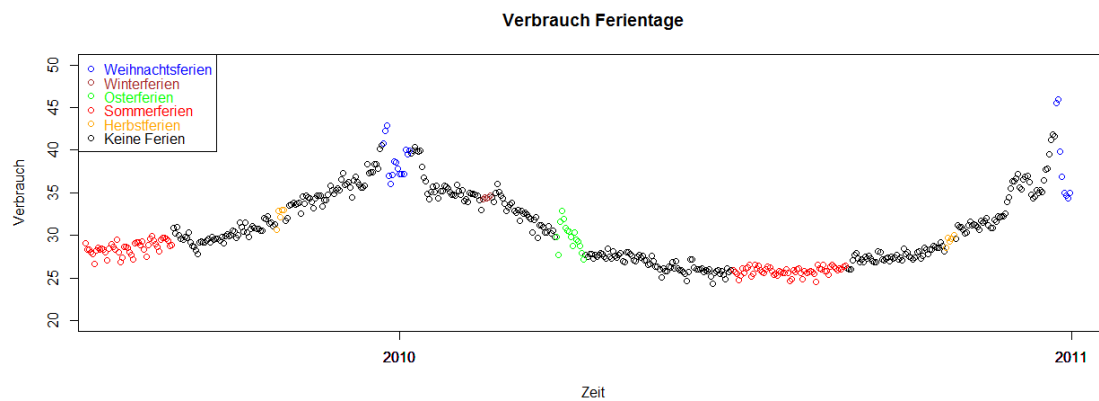


Abbildung 55: Verbrauchswerte mit Markierung der Ferientage

In der folgenden Abbildungen 56 ist der Irland-Verbrauch mit roter Markierung zu den Feiertagen dargestellt.

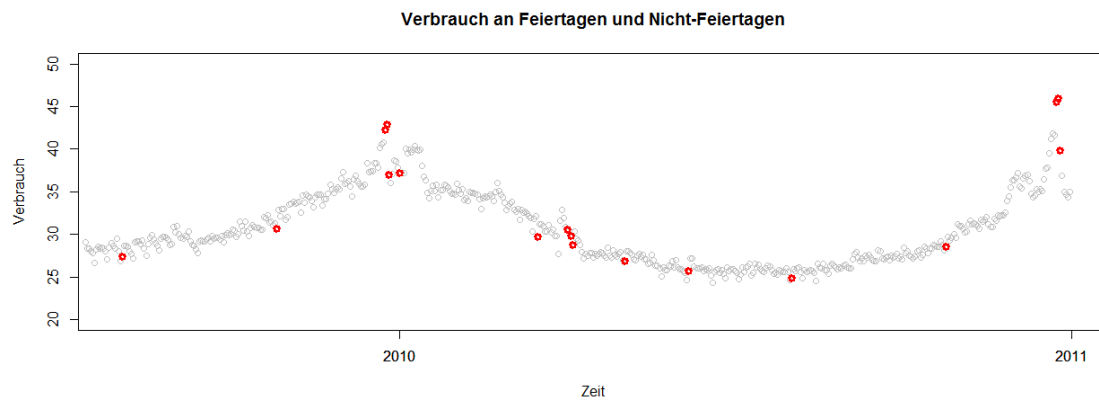


Abbildung 56: Verbrauchswerte mit Markierung der Feiertage und Nicht-Feiertage

Im folgenden Plot sind die irischen Wetterdaten und der zugehörige Durchschnitt der Temperaturen dargestellt (Abbildung 57).

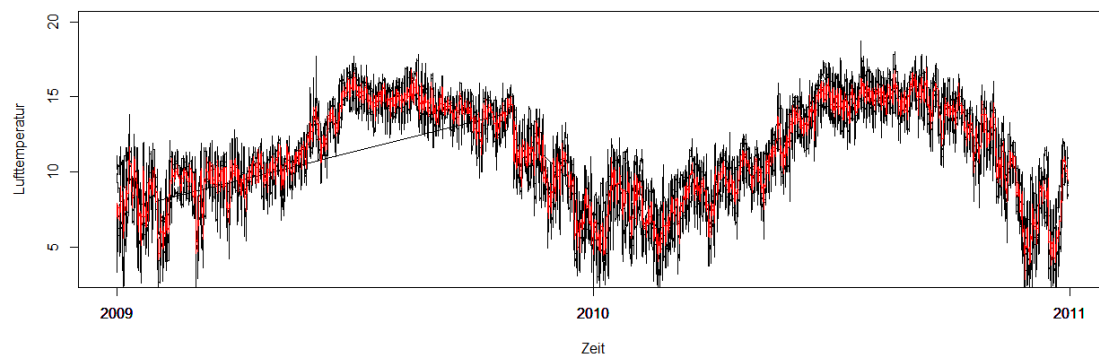


Abbildung 57: Lufttemperatur Marine Stationen und durchschnittliche Lufttemperatur

In der folgenden Darstellung (Abbildung 58) ist die durchschnittliche Lufttemperatur gegenüber dem durchschnittlichen Energieverbrauch aller irischen Haushalte abgebildet. Man erkennt dabei, dass bei niedrigen Temperaturen entsprechend mehr Energie verbraucht wird und bei einer hohen Lufttemperatur weniger.

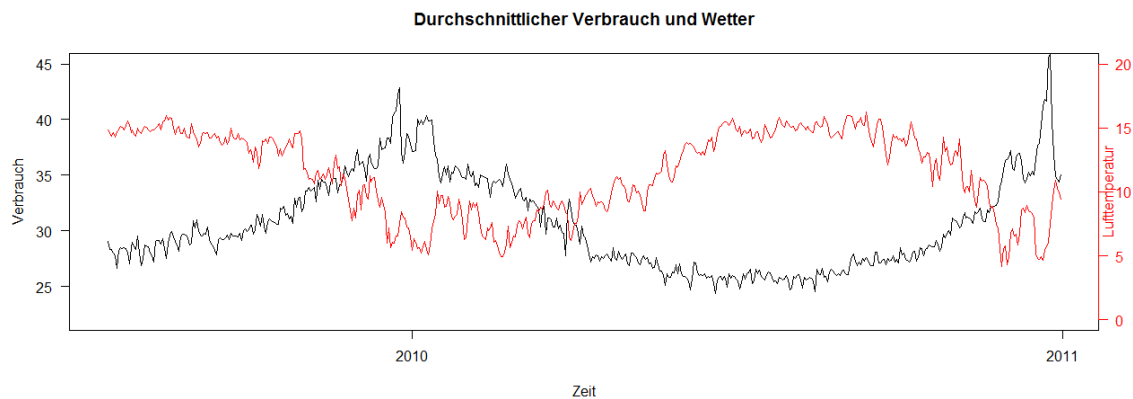


Abbildung 58: Lufttemperatur Marine Stationen und durchschnittlicher Verbrauch

Letztlich soll in der folgenden Darstellung (Abbildung 59) eine farbliche Unterscheidung aller Marine-Stationen gezeigt werden. Die Darstellung soll besagen, dass alle Stationen keine drastisch erkennbaren Ausreißer-Stationen beinhalten.

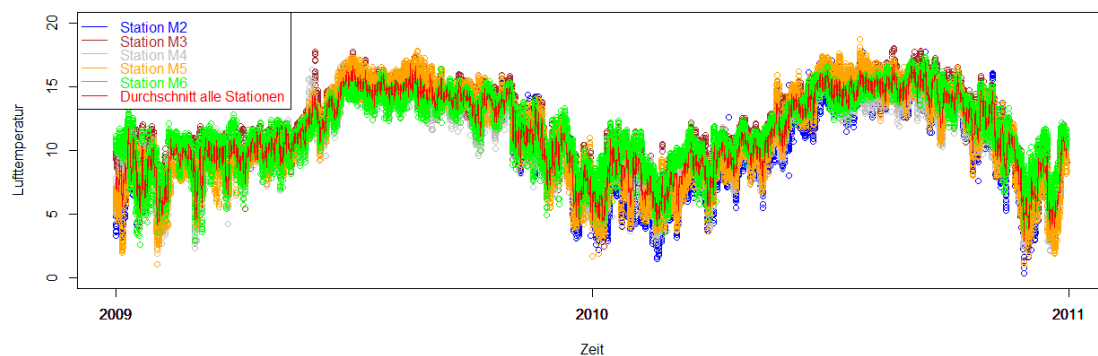


Abbildung 59: Wetterstationen Marine Bojen farblich unterschieden

Für das zu erstellende **Produktionsmodell**, welches ausschließlich auf die Energieverbrauchsdaten aus Deutschland zurückgreifen wird, sollen ebenfalls externe Daten hinzugezogen werden. Im Gegensatz zu dem Verbrauchsmodell, welches durch menschliches Handeln beeinflusst wird und dauerhaft über einen fixen Verbrauchswert verfügt, wird die Energieproduktion hauptsächlich vom Wetter beeinflusst. Daher ist die Anzahl von möglichen externen Daten begrenzt und Faktoren wie Feier- und Ferientage können nicht genutzt werden. Daher die folgenden drei Faktoren ausgewählt, dessen Datensätze ebenfalls

von dem Server des DWD abgerufen wurden:

- **Sonnenstrahlung:** Die Sonnenstrahlung ist im Zusammenhang der Energieproduktion, der wichtigste Faktor. Durch diesen lässt sich exakt bestimmen in welchem Zeitintervall die Sonne schien/scheinen wird.
- **Diffuse Strahlung:** Dieser Faktor gibt die Intensität der Strahlung an. Auf dem Weg zur Erde verliert die Strahlung an Intensität, beispielsweise durch das Durchdringen von Wolken.
- **Bedeckungsgrad:** Dieser liefert Informationen über den Grad der Bedeckung durch Wolken. Dieser Wert wird in verschiedenen Kategorien unterteilt von acht, was dem Grad stark bewölkt angibt, bis eins, freier Himmel

Nachdem die Faktoren bestimmt und eingebunden wurden, galt es den Einfluss dieser auf die bereits vorhandenen Faktoren zu untersuchen. Zu diesem Zweck wurde eine Korrelationsmatrix erstellt. Hierbei geht man von einem positiven Wert aus, wenn eine blaue Färbung zu erkennen ist. Ein neutraler Einfluss wird mit der Farbe weiß ausgewiesen und ein negativer mit einer roten Färbung.

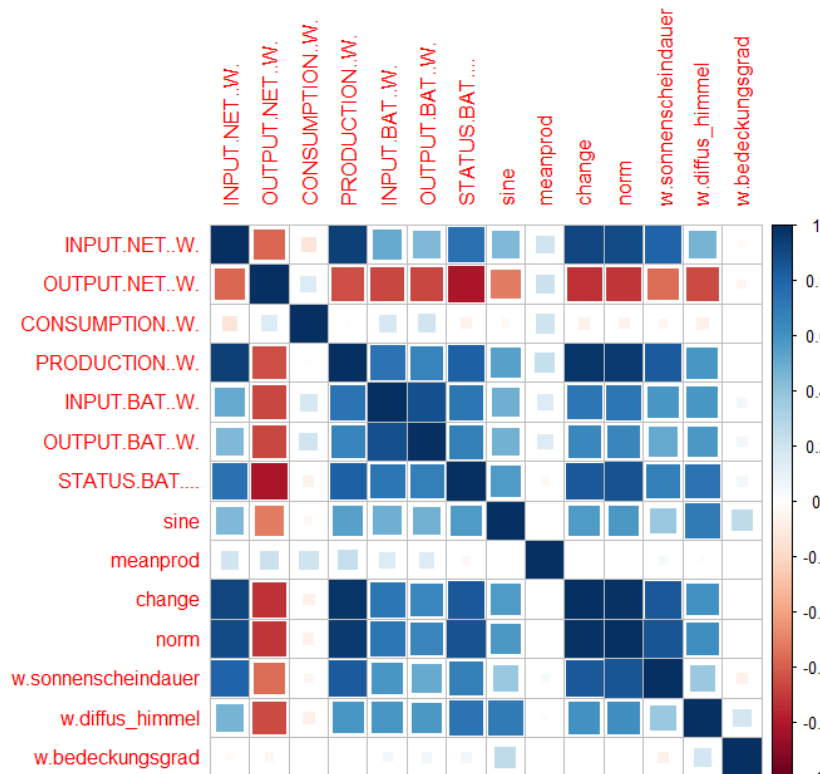


Abbildung 60: Korrelationsmatrix des Produktionsmodells

In der Abbildung 60 sind die Einflüsse der zuvor ausgewählten Faktoren zu erkennen (letzten drei Zeilen von rechts aus gesehen), woraus sich die folgende Bewertung ableiten lässt.

Die Sonnenscheindauer kristallisiert sich als hervorragender Faktor heraus. Dieser hat auf mehrere Faktoren einen guten Einfluss. Dazu zählen vor allem der Faktor *INPUT.NET..W.* (Einspeisung in das Stromnetz), der Faktor *PRODUCTION..W.* (Produktion) und *STATUS.BAT* (Batteriestatus/Akku). Auch der Faktor *diff_himmel* hat einen guten Einfluss auf die anderen Faktoren. Erwähnenswert sind hierbei die Faktoren *STATUS.BAT* und *sine*. Letzterer hilft bei der Erkennung von Saisonalitäten, aufgrund des Verlaufs. In diesem Zusammenhang ein logischer Faktor, welcher beeinflusst wird. Aber auch auf alle anderen Faktoren, mit Ausnahme des Verbrauchs, ist ein positiver Einfluss zu erkennen. Der Faktor *Bedeckungsgrad* hat einen neutralen Einfluss auf alle anderen Faktoren. Einzig und allein auf den Faktor *sine* ist ein geringer positiver Einfluss zu verzeichnen.

6.5.2. Data Preparation

Moving Average Anhand der vorliegenden Datensätze wurde der Moving Average (gleitender Durchschnitt) der Verbrauchsdaten (CONSUMPTION) der vergangenen sieben Tage erstellt. Dabei wurde jeder Datensatz der letzten sieben Tage (aktueller Tag nicht einbezogen) aufsummiert und durch die sieben Tage dividiert. Eine zusätzliche Spalte `mma` (Moving Average) wird im Datensatz für diese Angabe erstellt. Es gibt verschiedene Möglichkeiten den Moving Average in R zu berechnen:

1. Man nutzt aus dem `zoo`-Package in R die Funktion `rollmeanr()`. Der aktuelle Tag wird bei der Berechnung miteinbezogen, weswegen eine weitere Spalte zur Verschiebung der Daten notwendig ist, um den Moving Average in korrekter Zeile anzugeben (`lag`-Funktion), also eine Zeile höher setzen.
2. Man erstellt durch `function()` eine Moving Average-Funktion manuell.

Die erste Variante wurde gewählt, da die Berechnung in R damit effizienter gestaltet werden kann. Manuelle Funktionen können wegen mehrerer Variablenberechnungen ein Ergebnis langsamer generieren. Der Moving Average wurde folgendermaßen in R implementiert:

Listing 8: Moving Average

```
df3.day$lag <- lag(df3.day$CONSUMPTION..W., shift = -1)
df3.day$am <- rollmeanr(df3.day$CONSUMPTION..W., 7, fill = NA)
```

Die ersten sieben Datensätze sind von der Berechnung nicht betroffen und ergäben somit NULL. Sinnvolle Lösungen dafür wären:

1. Diese komplett gar nicht berechnen (mit NA befüllen), da es sonst zu Datenverfälschung bzw. letztendlich willkürliche Datenauswahl und fehlerhaften Modellen kommen kann.
2. Leere Datensätze mit vorhanden Daten berechnen, z. B. im dritten Datensatz wählt man den ersten und zweiten Verbrauchswert und teilt ihn durch die Anzahl vorhandener Datensätze, also in diesem Fall 2: $\frac{x_1+x_2}{2}$.

Je nach Durchschnittszeitraum (z. B. Betrachtung letzter 3, 7, 10, 50 oder 100 Tage, usw.) ist jeweils ein anderer Prognosewert zu erwarten. Im Folgenden sind beispielhaft die annähernd übereinstimmenden Energieverbrauchswerte des tatsächlichen Verbrauchs

und der Prognose der letzten 7, 50 und 100 Tagen mithilfe der Residualanalyse in R gegenübergestellt (Residuals vs. Fitted).

Im folgenden Plot (Abbildung 61) ist die Gegenüberstellung der Moving Average-Werte von 7 Tagen zu dem tatsächlichen Verbrauch dargestellt.

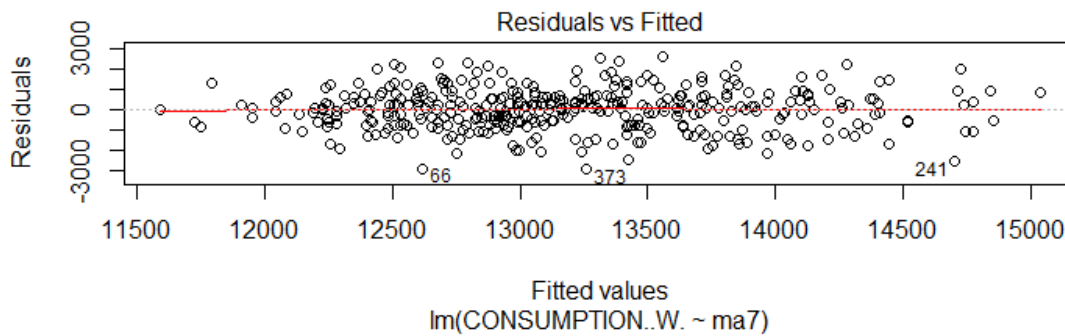


Abbildung 61: Moving Average von 7 Tagen und tatsächlicher Energieverbrauch

Es folgt die Gegenüberstellung mit den Moving Average-Werte von 50 Tagen (Abbildung 62).

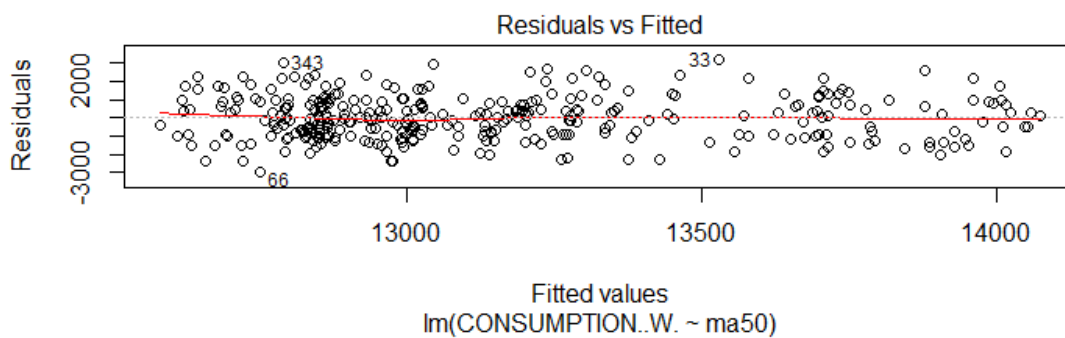


Abbildung 62: Moving Average von 50 Tagen und tatsächlicher Energieverbrauch

Schließlich ist folgend die Gegenüberstellung der Moving Average-Werte von 100 Tagen abgebildet (Abbildung 63).

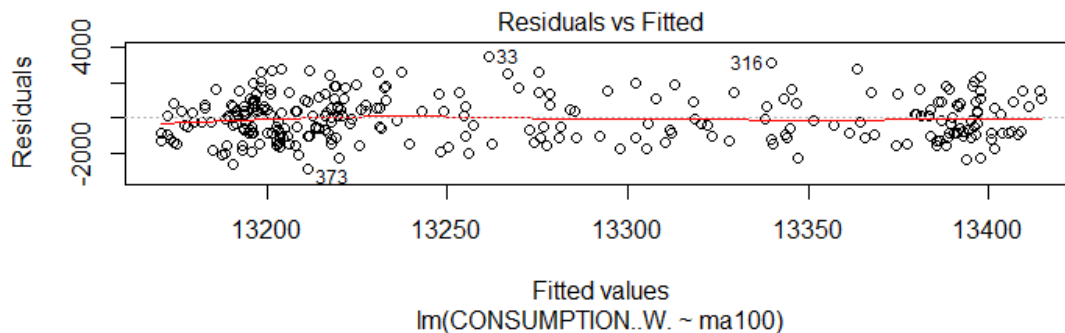


Abbildung 63: Moving Average von 100 Tagen und tatsächlicher Energieverbrauch

Dazu wurden entsprechende Statistiken in R ausgegeben, in denen insbesondere das R^2 untersucht wurde. Es folgen die entsprechenden Ausgaben.

Zunächst folgt die Statistikausgabe in R mit dem Moving Average für 7 Tage (Abbildung 64).

```
Call:
lm(formula = CONSUMPTION..w. ~ ma7, data = df1.day)

Residuals:
    Min       1Q   Median       3Q      Max
-2943.53  -652.33    54.77   607.25  2557.08

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  954.33406   947.44995    1.007   0.314
ma7          0.92870     0.07189   12.919 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 997.3 on 369 degrees of freedom
(6 observations deleted due to missingness)
Multiple R-squared:  0.3114,    Adjusted R-squared:  0.3096
F-statistic: 166.9 on 1 and 369 DF,  p-value: < 2.2e-16
```

Abbildung 64: Statistikausgabe in R - Moving Average von 7 Tagen

In der folgenden Konsolenausgabe ist die statistische Ausgabe mit Moving Average-Werten von 50 Tagen zu entnehmen (Abbildung 65).

```

Call:
lm(formula = CONSUMPTION..w. ~ ma50, data = df1.day)

Residuals:
    Min       1Q   Median       3Q      Max
-3015.7  -931.6   -38.1    826.7   3164.7

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 2745.3238  1762.2340   1.558   0.12
ma50         0.7910    0.1339   5.909 8.66e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1179 on 326 degrees of freedom
(49 observations deleted due to missingness)
Multiple R-squared:  0.09675, Adjusted R-squared:  0.09398
F-statistic: 34.92 on 1 and 326 DF, p-value: 8.66e-09

```

Abbildung 65: Statistikausgabe in R - Moving Average von 50 Tagen

Schließlich folgt die Statistikausgabe mit dem Moving Average-Werten von 100 Tagen (Abbildung 66).

```

Call:
lm(formula = CONSUMPTION..w. ~ ma100, data = df1.day)

Residuals:
    Min       1Q   Median       3Q      Max
-2893.4  -930.4   -80.5    802.5   3432.8

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.045e+04  2.570e+03   4.065 6.28e-05 ***
ma100       2.139e-01  1.947e-01   1.099  0.273
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1227 on 276 degrees of freedom
(99 observations deleted due to missingness)
Multiple R-squared:  0.004353, Adjusted R-squared:  0.000746
F-statistic: 1.207 on 1 and 276 DF, p-value: 0.2729

```

Abbildung 66: Statistikausgabe in R - Moving Average von 100 Tagen

Es lässt sich feststellen, dass ab ca. 50 Durchschnittstagen das R^2 unter 10 % liegt und sogar in den negativen Bereich (Durchschnitt letzte 200 Tage) fällt. Heißt also, dass der Durchschnitt über mehr Durchschnittstagen eine schlechtere Prognose liefert. Eine Prognose z. B. über 50 Tagen wäre also mit der Moving Average-Methode nicht sinnvoll.

In dem Zusammenhang lässt sich auch die Stichprobengröße bzw. Menge der Zeitreihenwerte erwähnen. Nach Bühner [Mar06] ist die Stichprobengröße sehr gut solange $n \Rightarrow 1000$ beträgt. In diesem Fall könnte sich dies aufgrund der Beobachtungen (vor allem ab 50 Durchschnittstagen) nicht bestätigen. Trend- und Saisonkomponenten werden in diesem

Fall erstmal nicht berücksichtigt.

Anomaly Detection Das Ziel in diesem Schritt war es, unterstützt durch eine Anomaly Detection, basierend auf gleitendem exponentiellen Durchschnitt und gleitender exponentiellen Standard Abweichung, Ausreißer zu identifizieren und somit große Energieverbraucher im Haushalt zu identifizieren. Diese Informationen sollen für höhere Aggregationsstufen behilflich sein. Zu Beginn der Tests, wurden Verbrauchsdaten in 15-minütiger Auflösung, aus einem Haushalt, über einen Zeitraum von einer Woche zur Verfügung gestellt. Zusätzlich wurden Notizen über die Laufzeiten der einzelnen Geräte in diesem Zeitraum zur Verfügung gestellt. Für die Anomaly Detection wurde folgender Ansatz benutzt. Hierbei steht EMA für den exponentiellen gleitenden Durchschnitt und EMS für die exponentielle Standardabweichung:

$$\begin{aligned}
 EMA_t &= w \times EMA_t + (1 - w) \times x \\
 EMS_t &= \sqrt{w \times EMS_t^2 + (1 - w) \times (x \times EMA_t)^2} \\
 &if(|x - EMA_t| > t \times EMS_t \text{ AND } t \times EMS_t > x) \\
 &\quad then(Anomaly = TRUE)
 \end{aligned} \tag{1}$$

In dieser Gleichung haben wir zwei Unbekannte, w und t. w ist das Gewicht und liegt zwischen 0,01 bis 1,0 und t beschreibt den Zeitraum der vergangenen Daten, die man für die jeweiligen Messungen benutzt. Ausgehend von den Notizen, testen wir für w von 0,01 bis 1,0 und für t von 1 bis 50 und überprüfen hierbei die größten Raten für richtig erkannte Großverbraucher und für richtig klassifizierte Messpunkte in den kein neues Gerät eingeschaltet wurde. Das Ergebnis wird in der nächsten Abbildung dargestellt:

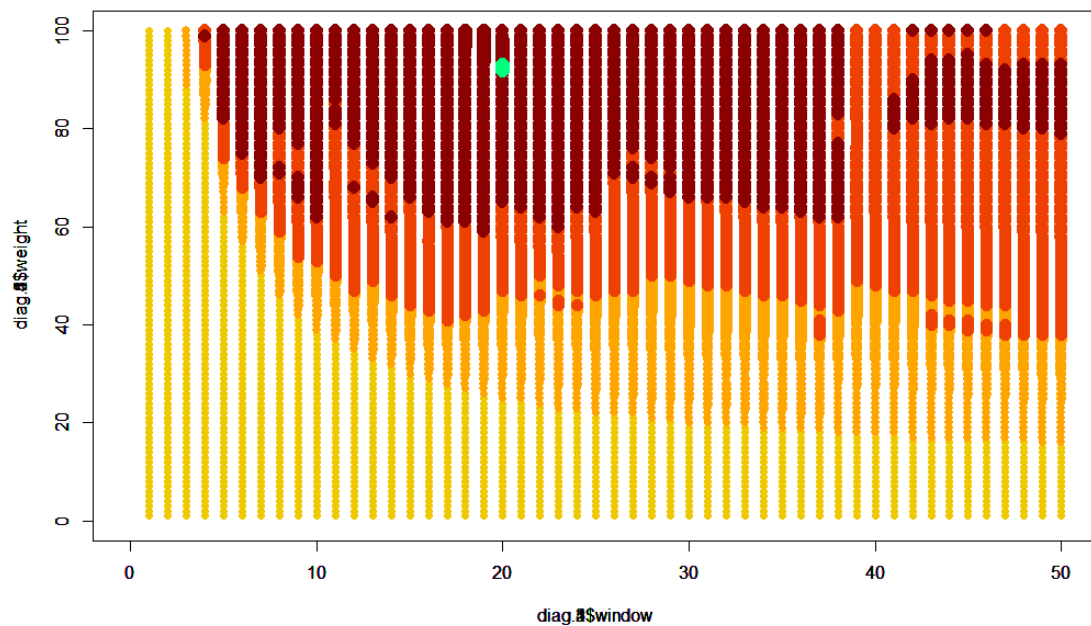


Abbildung 67: Ergebnisse der Tests

Auf der Y-Achse ist das Gewicht w in Prozent dargestellt und auf der X-Achse das gleitende Fenster für t . Je dunkler die Farbe und je größer der Punkt in diesem Diagramm, desto höher ist der Wert der richtig klassifizierten Messpunkte. Das Maximum bildet $t=20$ und $w=0,92$ oder $w=0,93$ (grün gekennzeichnet). Die Confusion Matrix für die Werte, zeigt die Ergebnisse detailliert:

Klassifiziert Klasse	TRUE	FALSE
TRUE	83,33%	16,66%
FALSE	8,97%	91,03%

Abbildung 68: Ergebnisse Anomaly Detection

In der Zeitreihe sehen die Ergebnisse wie folgt aus:

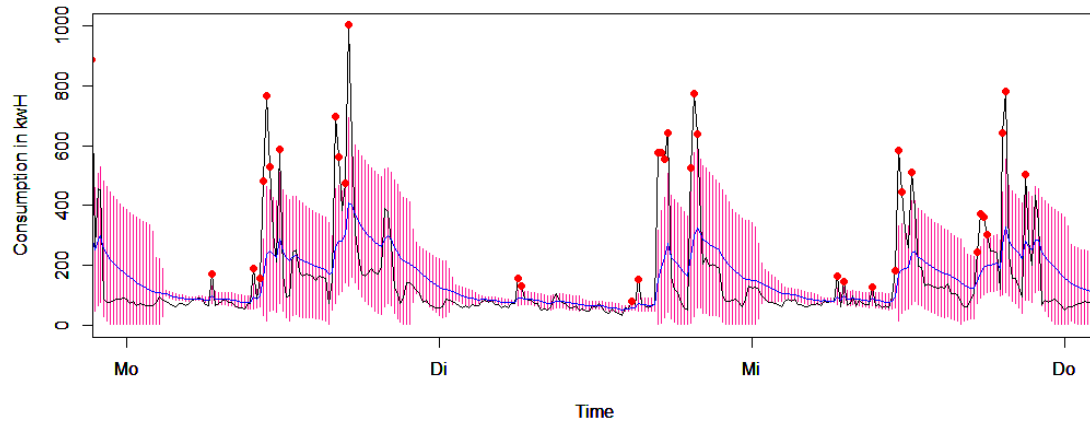


Abbildung 69: Ergebnisse der Anomaly Detection in der Zeitreihe

In schwarz ist die eigentliche Zeitreihe, in blau ist der exponentielle gleitende Durchschnitt, in rosa ist die exponentielle Standardabweichung und in rot sind die großen Energieverbraucher dargestellt.

SAX - Symbolic Aggregate Approximation Symbolic Aggregate Approximation ist eine Methode, mit der Zeitreihen-Daten in n Segmente aufgeteilt werden. Dabei werden die Daten der Zeitreihen in alphabetische Werte kodiert. Die Aufteilung der Daten passiert anhand der Verteilung, der Werte. Ein Beispiel: eine Zeitreihe wird in vier Segmente, a , b , c und d aufgeteilt. Die Bereiche der Segmente werden somit definiert durch:

- $a < \mu - \sigma$
- $\mu - \sigma < b < \mu$
- $\mu < c < \mu + \sigma$
- $\mu + \sigma < d$

In dem Anwendungsfall, der im Rahmen der Arbeit behandelt wurde, ist die saisonale Anpassung vorangestellt.

Nachdem die die Zeitreihen-Daten zuvor saisonal angepasst wurden, und somit stationär sind, können die vier Segmente durch die Verteilung definiert werden:

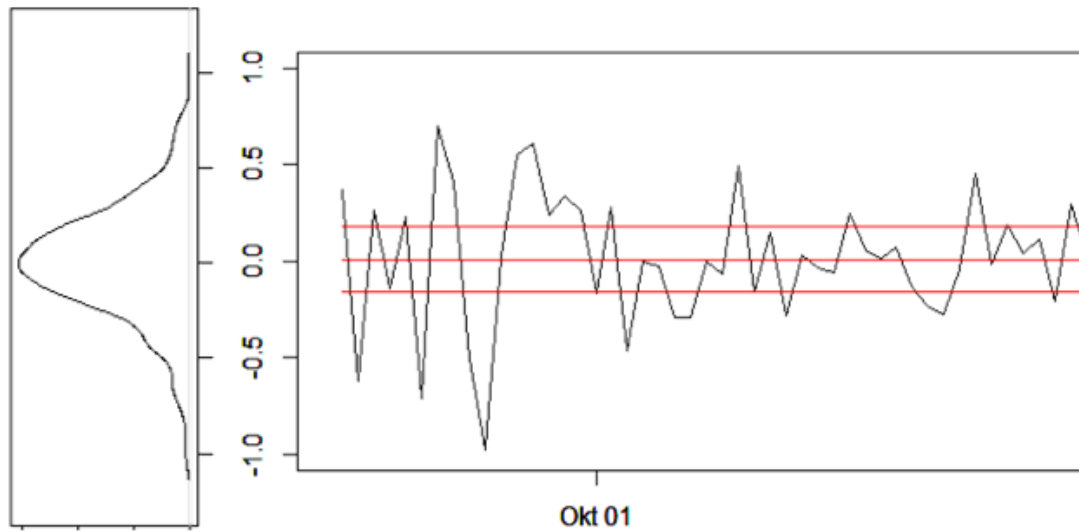


Abbildung 70: Stationäre Zeitreihe mit vier Segmenten

Das Kodieren der stationären Zeitreihen, ermöglicht es neue Modelle auf die Daten anzuwenden und mithilfe von Mustererkennung den nächsten Wert vorherzusagen.

Normalisierte Daten Nachdem die ersten Modelle mit den absoluten Zahlen entworfen wurden, kam man zu dem Ergebnis, dass die Modelle keine korrekten Resultate abbilden. Der Ursprung dafür liegt in den unterschiedlichen Ebenen des mindest und maximalem Stromverbrauchs, siehe folgende Abbildung.

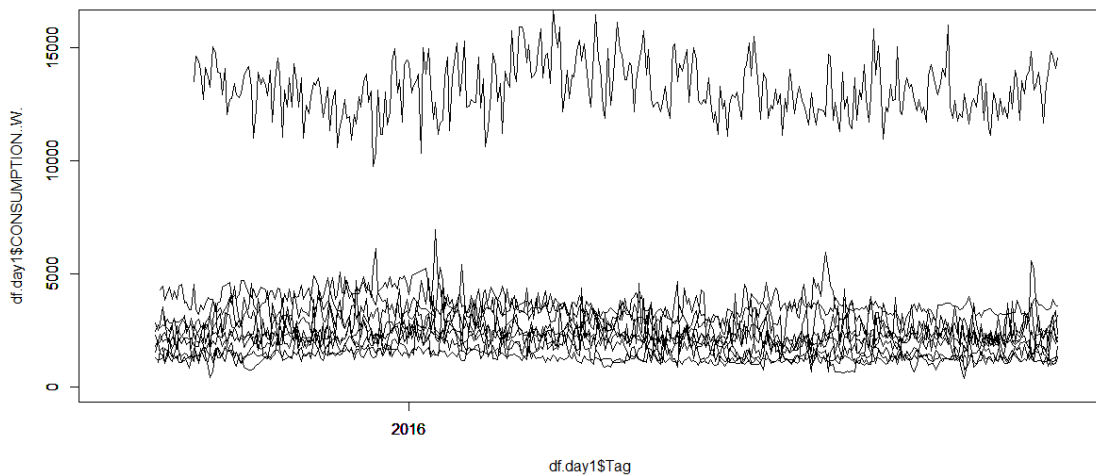


Abbildung 71: Unterschiedliche Stromverbräuche in unterschiedlichen Haushalten

Diese Erkenntnis führte dazu, dass zunächst alle Werte der Haushalte normalisiert werden müssen. Dies passierte in den folgenden Schritten:

1. Der Minimalwert jedes Haushalts wurde von jedem Messpunkt des Haushalts subtrahiert.
2. Das arithmetische Mittel wurde von jedem Haushalt berechnet.
3. Jeder Wert eines Haushalts wurde nach der Subtraktion des Minimalwerts mit der arithmetischen Mittel dividiert.

Die Berechnung ist wie folgt, wobei x_n der normalisierte Wert von x ist:

$$x_n = \frac{x - x_{min}}{x_{mean} - x_{min}} \quad (2)$$

Durch die Normalisierung anhand des arithmetischen Mittels anstatt des Maximums, verhindert man, dass ein besonders hoher Stromverbrauch an wenigen Tagen, die Normalisierung verzerrt. Da man davon ausgeht, dass so ein besonders hoher Stromverbrauch nur an vereinzelt Tagen anfällt, kann man anhand des arithmetischen Mittels berechnen, welches dafür bekannt ist, sensibel bezüglich Ausreißer zu sein. Nach der Normalisierung sieht die Verteilung der Daten wie folgt aus:

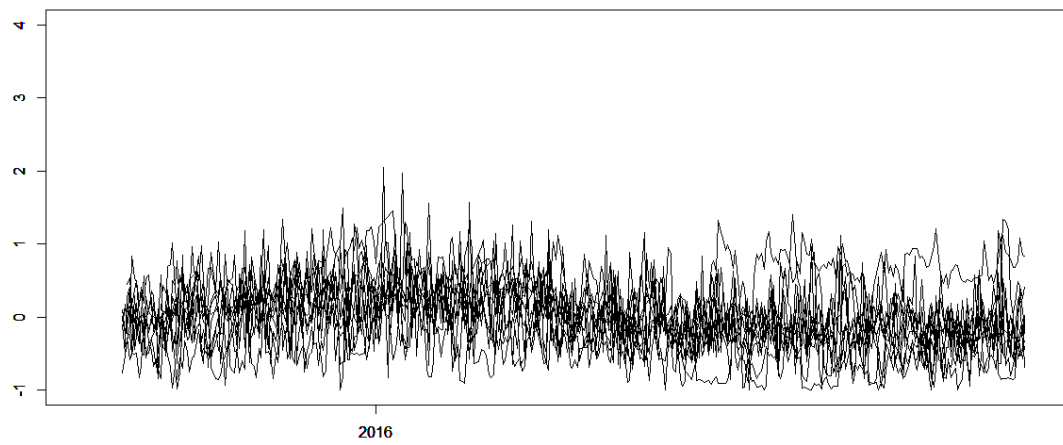


Abbildung 72: Werte aller Haushalte nach Normalisierung

Zudem kann man in der nächsten Abbildung 73 den Vergleich der Werteverteilung vor der Normalisierung.

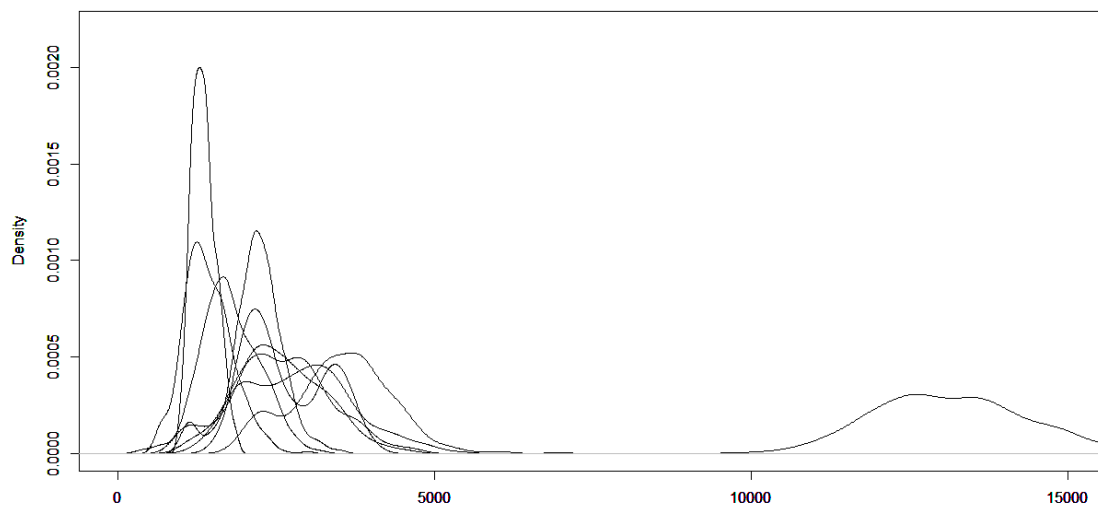


Abbildung 73: Werteverteilung vor der Normalisierung

Es folgt die Abbildung 74 nach der Normalisierung.

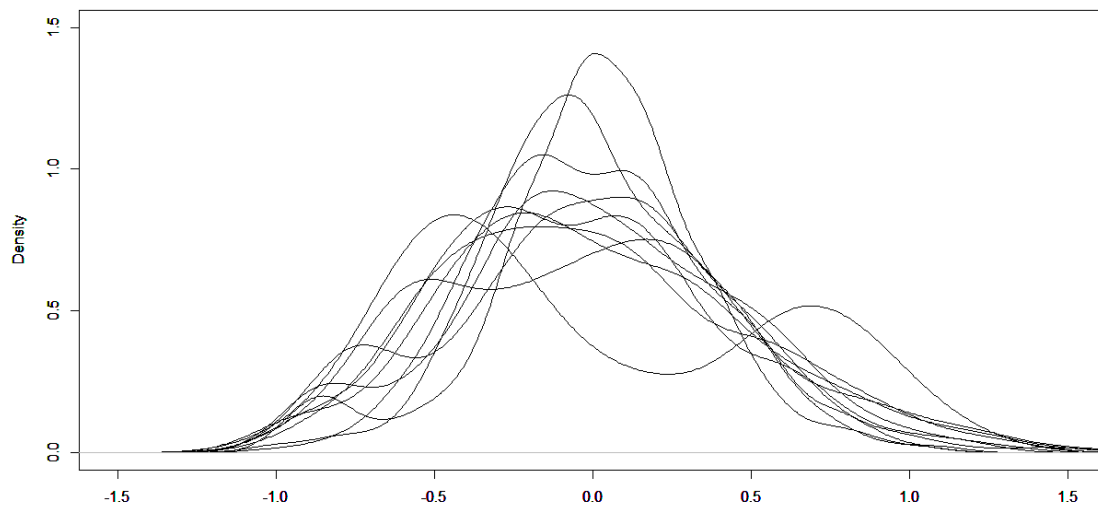


Abbildung 74: Werteverteilung nach der Normalisierung

Schließlich erkennt man, dass nach dem Verfahren die Haushalte miteinander verglichen werden können. Mit dieser Normalisierung ist die Grundlage für weitere Modellbildungen geschaffen.

ARIMA Das ARIMA-Modelling (Autoregressive-Moving Average) untersucht zyklische Verhalten in Zeitreihen und erstellt für solch einen Anwendungsfall das passende Modell. Im ersten Schritt werden dafür die Differenzen zum vorherigen Tag gebildet, also:

$$x'_t = x_t - x_{t-1}hi \quad (3)$$

In den nächsten Schritte, wird derselbe Prozess wiederholt, mit den Daten der Differenzierung des 1. Grades und im dritten Schritt mit den Daten der Differenzierung des 2. Grades, also:

$$x_t^* = x'_t - x'_{t-1} \quad (4)$$

Die Zeitreihen der Differenzierung des 1., 2. und 3. Grades sehen wie folgt aus:

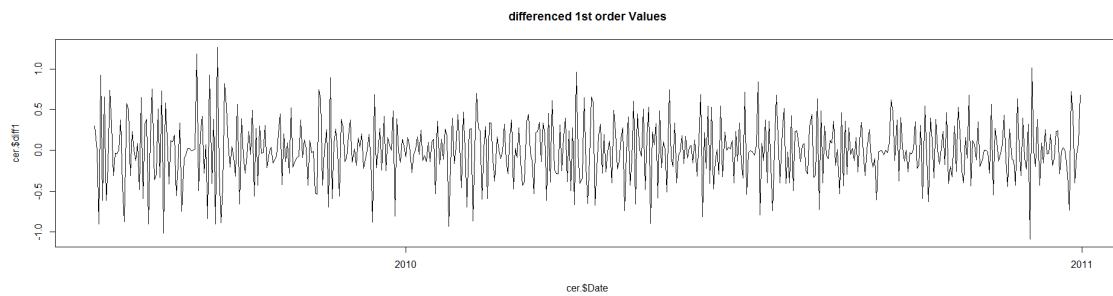


Abbildung 75: Differenzierung des 1. Grades

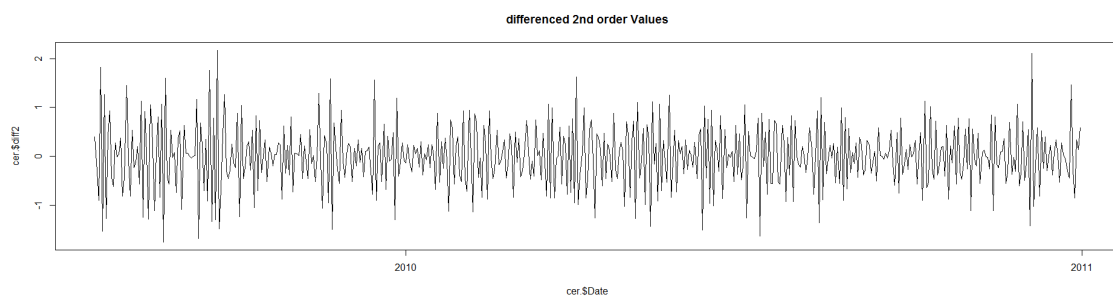


Abbildung 76: Differenzierung des 2. Grades

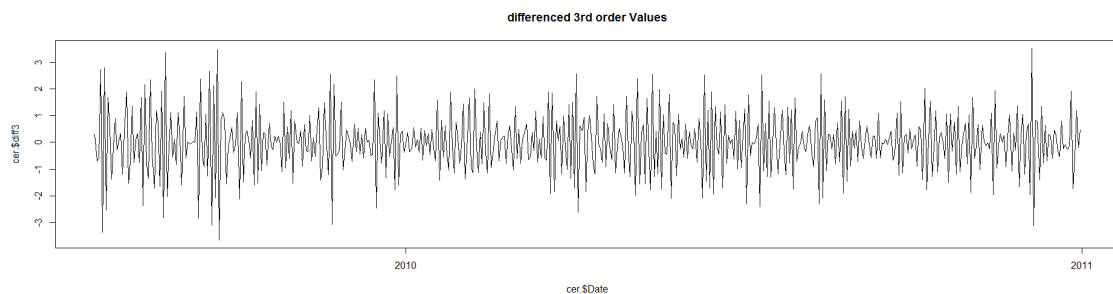


Abbildung 77: Differenzierung des 3. Grades

Im nächsten Schritt wird die Varianz der drei Zeitreihen berechnet und die Zeitreihe, mit der geringsten Varianz, im nächsten Schritt auf das zyklische Verhalten untersucht. In dem Fall hat die Zeitreihe mit der Differenzierung des 1. Grades die geringste Varianz und wird mit der ACF (Autocorrelation Function) und der PACF (Partial Autocorrelation Function) untersucht. Die Ergebnisse der beiden Funktionen sehen wie folgt aus:

In beiden Graphen erkennt man, dass die Daten dauerhaft nach 7 Tagen die höchste

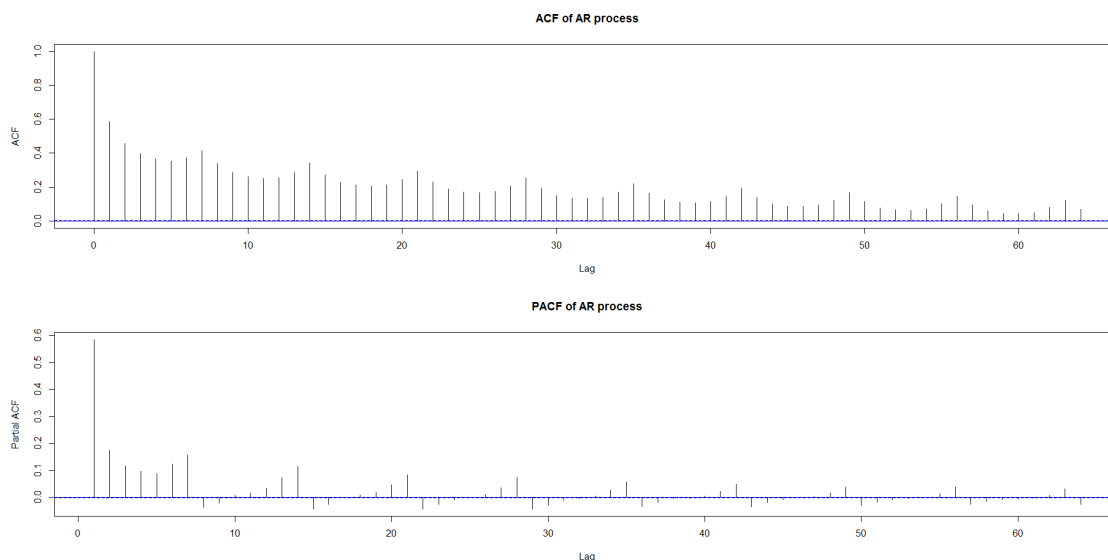


Abbildung 78: ACF und PACF

Korrelation besitzen und somit ein klarer Zyklus besteht. Nach weiteren Test wurde dennoch sichtbar, dass nicht jeder Haushalt den gleich Grad des Zyklus? besitzt und somit auch in dem Fall nicht ein Modell für alle Haushalte gebildet werden können.

Saisonale Anpassung Die saisonale Anpassung ist die Fortsetzung der Normalisierung und hat das Ziel, die Daten der verschiedenen Haushalte weiterhin vergleichbar zu machen, da weiterhin eine großer Unterschied in der Saisonalität der Haushalte liegt. Der Ansatz, der hier verfolgt wird ist, dass der gleitende Durchschnitt mit der Länge eines Zyklus, in dem Fall eine Woche, gebildet wird und der gleitende Durchschnitt von den relativen Werten abgezogen wird. Der gleitende Durchschnitt ist hierbei definiert durch:

$$m_{MA(7)}(t) = \frac{1}{7} (x(t-1) + x(t-2) + x(t-3) + x(t-4) + x(t-5) + x(t-6) + x(t-7)) \quad (5)$$

Nachdem der gleitende Durchschnitt von den Werten subtrahiert wurde, wurde eine Stichprobe der Daten entnommen und die jeweiligen Verteilungen graphisch dargestellt:

In dieser Abbildung sieht man, dass die Werte der Haushalte nun vergleichbar und um 0,0 fast gleich verteilt sind.

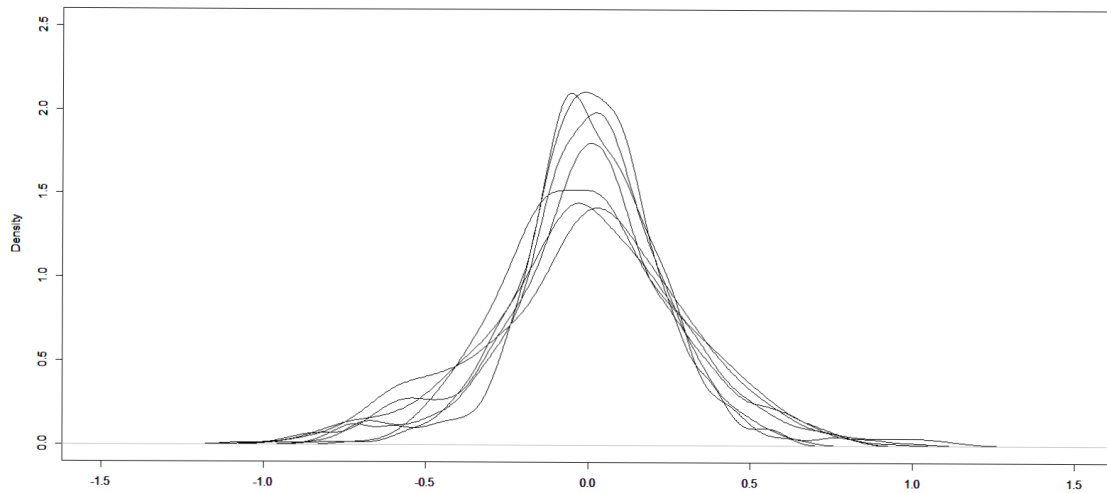


Abbildung 79: Werteverteilung nach der Normalisierung und der saisonalen Anpassung

Moving Average Berechnung der Moving Average 7-Werte und Zusammenführung der irländischen Verbräuche mit den Wetter, Ferien- und Feiertagen und die dabei identifizierte Probleme Für die Berechnung der Moving Averages von 7 Tagen und die Zusammenführung der irländischen Verbräuche mit den Wetter, Ferien- und Feiertagen wurden zuerst die normalisierten Verbrauchsdaten mit den Wetterdaten (Temperaturangaben pro Tag über den entsprechenden Zeitraum) zusammengefasst. Die Verbrauchs-Datensätze wurden pro Tag aggregiert, in dem der durchschnittliche Verbrauch pro Tag für die Aggregation eingefügt wurde (Funktion `aggregate()`).

Die aggregierten Datensätze wurden im nächsten Schritt mit den Feiertagen, in der identifiziert wird, ob es sich um einen oder keinen Feiertag handelt zusammengefasst. Dazu dienten zunächst zwei Tabellen, damit die Moving Average-Werte für Feiertage und Nicht-Feiertage berechnet werden können (mit der Funktion `rollmeanr()`). Bei den Feiertagen wurde der an dem Tag verbrauchte Wert für den Moving Average-Wert übernommen, da eine Berechnung eines Moving Averages von 7 Tagen an der Stelle keinen Sinn machen würde, da einzelne Feiertage bei der Berechnung einbezogen werden. Bei den Nicht-Feiertagen wurde die Liste für den Moving Average 7 erstellt. Anschließend wurden die beiden Tabellen zusammengefasst und nach Datum sortiert.

Bei den Ferientagen wurde so ähnlich vorgegangen, außer dass beim Splitting, der Moving Average je Ferienart (z. B. Weihnachtsferien, Sommerferien, usw.) und Nicht-Ferientag in separaten Tabellen berechnet wurde. Bei Ferienzeiträumen, die unter 7 Tagen betragen, wurde der MA7-Wert als NA-Wert (not available) gesetzt. Anschließend wurden alle Ta-

bellern dann wieder zusammengefügt und nach Datum sortiert.

Allgemein besteht hier das Problem, das eine Lücke bei der Berechnung der MA7-Werte besteht, da die Zeiträume mit und ohne Ferien unbeachtet bleiben, da ein Splitting von Ferien- und Nicht-Ferientagen bzw. Feiertagen und eine fortlaufende Berechnung stattfindet. Beispielsweise für den MA7-Wert für den August (keine Sommerferien mehr) werden die Verbräuche vom Juni (vor den Sommerferien) herangezogen. Mögliche Lösung wäre hierbei Intervalle/Phasen zwischen den Stichdaten (Sprung Ferien- und Nicht-Ferientag) zu betrachten, die allerdings eine komplexere Struktur und längere Laufzeit der Berechnung der Moving Average-Werte verursachen würden.

Zum Schluss wurde die gesamte Feriendaten Tabelle mit den Feiertagen (positiv) zusammengefasst und nach Datum sortiert. An dieser Stelle besteht dann das Problem, dass man Datensätze doppelt hat, die einen Feiertag aufweisen und man an dieser Stelle entscheiden müsste, welchen MA7-Wert man hinzuzieht. Lösung hierfür wäre, dass die Feiertage keinen separaten Berechnungen unterliegen sollen, sondern lediglich nur markiert werden, z. B. anhand einer weiteren Spalte im Datensatz, die mit 0 oder 1 zu befüllen ist.

Schließlich wurde die Differenz zwischen realem Verbrauch und MA7-Wert gebildet. An der Stelle, wo die Differenz exakt gleich 0 betrug, also entsprechend ein Feiertag, wurde der Verbrauch des letzten Tages als MA7-Wert übertragen. Zum Schluss wurde noch überprüft, bei welchem Tag noch allgemein ein NA-Wert steht. Ist dies der Fall wurde der vorherige tatsächliche Tages Verbrauchswert als MA7-Wert übernommen, um keine Datensätze verlieren zu müssen. Letztendlich bleibt nur noch ein NA-Wert (erste Zeile), der einzig eliminiert wurde.

Zusammenfassend lassen sich allgemein bei diesem Zusammenführungsprozess Genauigkeitslücken identifizieren, für die vollständigermaßen Lösungen ausstehen:

- Man müsste für die MA7-Berechnung Intervalle betrachten, z. B. zwischen Sommerferien und Herbstferien, usw. damit keine Ungenauigkeiten entstehen. Dadurch gehen aber mehr Daten verloren (je Intervall 7 Tage (da der Moving Average von 7 Tagen berechnet werden soll; es sei denn, man wählt einen anderen Moving-Average-Wert, z. B. von 3 Tagen))
- Die Feiertage kommen doppelt vor. Lösung hierfür wäre, dass die Feiertage keinen separaten Berechnungen unterliegen, sondern lediglich nur markiert werden sollen, z. B. anhand einer weiteren Spalte die mit 0 oder 1 zu befüllen ist.

6.5.3. Modelling

In der Phase Modelling stand die Untersuchung verschiedener Machine Learning-Ansätze im Fokus. Ziel ist es die Modelle performanter und einen besseren Prognosewert aufweisen. Für jedes Modell wurden neben der linearen Regression jeweils zwei weitere Algorithmen ausgewählt, umgesetzt und evaluiert. Zunächst einmal wird mit dem Vergleichsmodell der linearen Regression begonnen, dieses dient zur Gegenüberstellung der anderen Algorithmen. Für das Verbrauchsmodell wurden der Bayes-Klassifikator und die Methode nach Holt-Winters ausgewählt und für das Produktionsmodell Random Forest und künstlich neuronale Netze.

Lineare Regression Die lineare Regression war der erste Machine Learning-Ansatz welcher verfolgt wurde. Um das Erstellte Modell zu verbessern, wurde die Kreuzvalidierung gewählt. Der erste Schritt bei der Durchführung bildet die Partitionierung eines Datensatzes in Test- und Trainingsdaten. [Mat17] Wie im nachstehenden Code zu sehen, wurde ein Verhältnis von 70 % Trainingsdaten und 30 % Testdaten gewählt.

Listing 9: Splitten der Trainings- und Testdaten

```
split.lm <- sample.split(mod.lm.x$norm, SplitRatio = 0.7)
train.lm <- subset(mod.lm.x, split.lm == T)
test.lm <- subset(mod.lm.x, split.lm == F)
```

Der Trainingsdatensatz wird für das Training des Algorithmus genutzt, um dieses für überwachtes Machine Learning zu trainieren. Dazu wird zunächst einmal eine lineare Regression erstellt.

Listing 10: Training des Algorithmus

```
fit.lm <- lm(norm ~ sine + w.sonnen Scheindauer
            + w.diffus_himmel + w.bedeckungsgrad, data = train.lm)
train.fit.lm <- fit.lm
```

Durch den Testdatensatz wird der Algorithmus nun einer Leistungsbewertung unterzogen. Im Anschluss daran erfolgt die Berechnung der Differenz zwischen dem tatsächlichen und dem prognostizierten Wert, um die erfolge des Training zu verfolgen.

Listing 11: Berechnung der Differenz

```
test.lm$predicted <- predict(train.fit.lm, test.lm, level = 0.95)
train.lm$dif <- train.lm$norm - train.lm$predicted
test.lm$dif <- test.lm$norm - test.lm$predicted
```

Für jeden Durchlauf der Kreuzvalidierung, in diesem Fall erfolgten fünf Iterationen, wird eine zufällige Partitionierung der Datensätze in Trainings- und Testdatensatz gewählt. Der Testdatensatz dient zur anschließenden Leistungsbewertung genutzt. Dieser Vorgang wird mehrere Male wiederholt, bis alle möglichen Kombinationsmöglichkeiten angewendet und keine Verbesserung mehr zu erkennen ist [Mat17].

Random Forest Bei Random Forest handelt es sich um ein Verfahren, welches zur Klassifikation und Regressionen von Daten angewendet wird. Die Idee dahinter ist, dass eine Vielzahl von Entscheidungsbäumen (trees) erstellt werden und somit ein Forest entsteht. Der Begriff Random lässt sich herleiten, da diese Bäume zufällig generiert werden [Fab15].

Die Durchführung liefert interessante Kenntnisse darüber, welche Variablen einen besonderen Einfluss auf die Zielgröße haben. Bei jedem Objekt findet sowohl eine Klassifikation, als auch eine Regression statt. Im Klassifikationsfall fällt die Entscheidung auf die Klasse, für die sich der Großteil der Bäume entschieden hat. Entsteht ein Gleichstand, dann erfolgt eine zufällige Zuweisung. Im Regressionsfall wird die Entscheidung auf Grundlage des Durchschnitts aller Bäume getroffen [Fab15].

Mithilfe des Packages randomForest konnte der Algorithmus in der R Umgebung erstellt werden. Wie zuvor bei der Kreuzvalidierung wurden auch hier Test- und Trainingsdaten erstellt. Wie im folgenden Listing 12 zu erkennen, wurde als Output der normalisierte Wert vorgegeben und als Faktoren die zuvor gewählten Wettereinflüsse.

Listing 12: Berechnung Wettereinflüsse

```
fit.rf <- randomForest(norm ~ sine + w.sonnenscheindauer + w.diffus_himmel  
+ w.bedeckungsgrad, data = train.rf, ntree=500)
```

Die Vorteile dieses Algorithmus liegen in der robusten Verhaltensweise im Bezug zur Überanpassungen, welche aufgrund der hohen Anzahl an Variablen nicht ausgeschlossen werden kann. Im Vergleich mit anderen Methoden besitzt Random Forest eine gute Performance und schneidet im direkten Vergleich mit einem einzelnen Entscheidungsbaum, bei dem eine hohe Varianz vorzufinden ist (Fehlentscheidung an einem relativ hohen Knoten zieht sich durch die weiteren Knoten) deutlich besser ab [Fab15].

Neuronales Netz Ein neuronales Netz besteht aus einer Anzahl von Neuronen, welche die Informationsverarbeitungseinheit bezeichnet werden. Diese sind durch gewichtete Verbindungen verbunden, welche das Austauschen von Nachrichten ermöglichen. Neuronale Netze gelten als sehr lernfähig, denn sie lernen aus Erfahrungen, welche sie durch ausgewählte Trainingsdaten erlangen. Unter lernen versteht man in diesem Zusammenhang die Anpas-

sung der Gewichtungen zwischen den einzelnen Neuronen des Netzes. Es wird zwischen drei verschiedenen Lernmethoden unterschieden: überwacht, bestärkt und nicht-überwacht. In diesem Fall wird ausschließlich mit der überwachten Lernmethoden gearbeitet [Kla15].

Beim überwachten Lernen wird dem Netz ein Input und ein Output vorgegeben. Anhand dieser kann das Netz eigenständig Muster erkennen und so die Wichtigkeit einzelner Faktoren korrekt einstufen [Kla15].

Um ein neuronales Netz in R zu erstellen stehen verschiedene Packages zur Verfügung. In diesem Fall wurde das Package *neuralnet* genutzt, womit auch eine grafische Ausgabe erfolgen kann. Wie in der nachstehenden Abbildung 80 zu erkennen wurden als Input die ausgewählten Faktoren *sine*, *sonnenscheindauer*, *diff_himmel* und *bedeckungsgrad* und als Output der normalisierte Produktionswert vorgegeben.

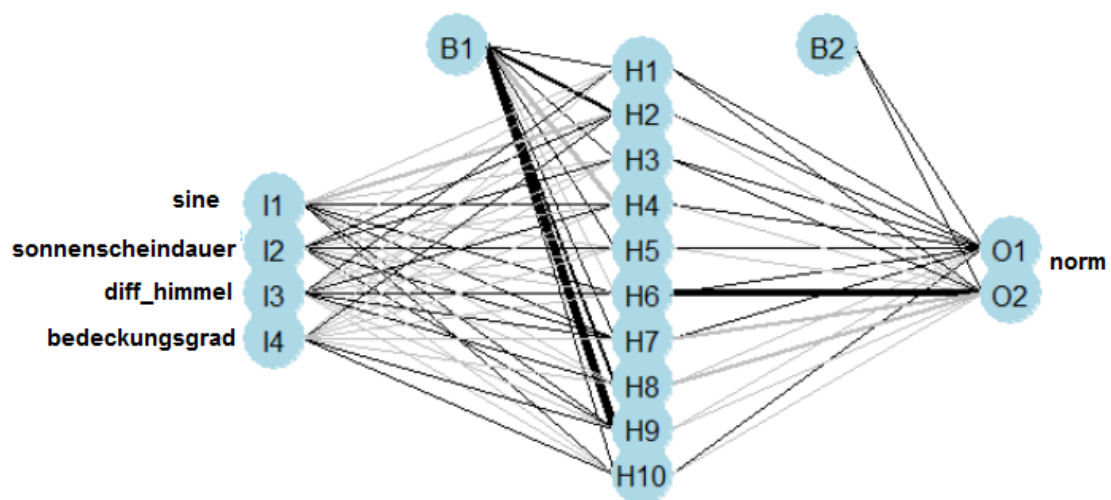


Abbildung 80: Neuronales Netz mit den In- und Output Variablen

Bayes-Klassifikator Für eine Klassifizierung der vorliegenden Daten, wie z. B. eines niedrigen, mittleren oder hohen Energieverbrauchs, wurde die Bayes Klassifikator-Methode angewandt. Die Bayes-Methode ermöglicht die Klassifikation und Vorhersage einer Klasse. Alle Attribute werden statistisch unabhängig betrachtet (Unabhängigkeitsannahme teilweise jedoch verletzt; z. B. im Winter gilt nicht automatisch ein hoher Energieverbrauch, wenn z. B. Urlaub besteht, Temperaturschwankungen sind, etc.). Bei fehlenden Werten wird bei der Klassifikation das Attribut bei der Berechnung ausgelassen. In R stellt das Package *e1071* die notwendigen Funktionen der Bayes-Methode zur Verfügung [Wol07] [edu15].

In den untersuchten Datensätzen wurden die Bayes-Klassen im R-Skript folgendermaßen (am Normalisierungswert orientiert) eingeteilt.

Listing 13: R-Skript Bayes-Klassen

```
mod.nb$class <- "hoch"
mod.nb$class[which(mod.nb$norm < 1)] <- "mittel"
mod.nb$class[which(mod.nb$norm < 0.3)] <- "niedrig"
```

Mit der R-Funktion `naiveBayes()` wurde im nächsten Schritt die Klasse prognostiziert.

Listing 14: R-Skript Prognose

```
mod.nb$class <- as.factor(mod.nb$class)
nb <- naiveBayes(class ~ ., data = mod.nb)
```

Holt-Winters-Verfahren Einfache exponentielle Glättungsverfahren eignen sich nicht gut für trend- und saisonbehaftete Zeitreihen. Die Holt-Winters-Methode berücksichtigt den lokalen Trend durch eine laufende Niveaueinstellung. Die Anpassung wird ebenfalls laufend korrigiert, also rekursiv berechnet. Bei dem exponentiellen Glättungsverfahren handelt es sich um eine Erweiterung des doppelten exponentiellen Glättens. Für die Geschwindigkeit der Niveaueinstellung werden Parameter α und β gesetzt. Nach Holt-Winters (HW) besteht eine noch genauere Prognosemöglichkeit. Es folgt die mathematische Formel dazu.

$$g_t^H = \beta (g_{t-1}^H + b_t - 1) + (1 - \beta) x_t \quad (6)$$

$$b_t = \alpha b_{t-1} + (1 - \alpha) (g_t^H - g_{t-1}^H) \quad (7)$$

α und β sind dabei zwischen 0 und 1 zu wählen, jedoch sind Werte zwischen 0,5 und 0,9 gängig. Je größer α , desto langsamer wird die Steigung b_t verändert. Für $\alpha = 1$ besteht das einfache exponentielle Glätten. Unbekannte Parameter werden durch das Minimieren des quadratischen Vorhersagefehlers festgelegt. Alpha stellt die exponentielle Glättung dar (Niveauschätzung = aktueller Messwert, den man benutzt, um den nächsten Messwert abzuschätzen), Beta die Trendkomponente (Trendschätzung = langfristiger Trend, z. B. 10 % Steigung jeden Monat) und Gamma die Saisonkomponente (Saisonalitätsschätzung = Zunahme von Erfahrungswerten (Saisonalitäten), z. B. je Woche, Monat, Jahr) dar.

Listing 15: Definition der HW-Funktion

```
HoltWinters(x, alpha = NULL, beta = NULL, gamma = NULL,
seasonal = c("additive", "multiplicative"),
start.periods = 2, l.start = NULL, b.start = NULL,
```

```
s.start = NULL,
optim.start = c(alpha = 0.3, beta = 0.1, gamma = 0.1),
optim.control = list())
```

[Jos16] [Rai12]

Bei der Modellierung im R-Skript wurde die abgekürzte Form mit der Gewichtung der einzelnen Komponenten verwendet:

Listing 16: Gewichtung der einzelnen Komponenten

```
HW <- HoltWinters(df1.day$CONSUMPTION..W., alpha = 0.5, beta = 0.5, gamma =
F)
```

In diesem Fall ist $\gamma = \text{False}$, da an dieser Stelle die Saisonkomponente bei der Berechnung nicht einbezogen werden soll.

Es folgt ein Plot mit $\alpha = 1$. Der Holt-Winters-Werteverlauf ist in rot markiert; der tatsächliche Energieverbrauch in schwarz.

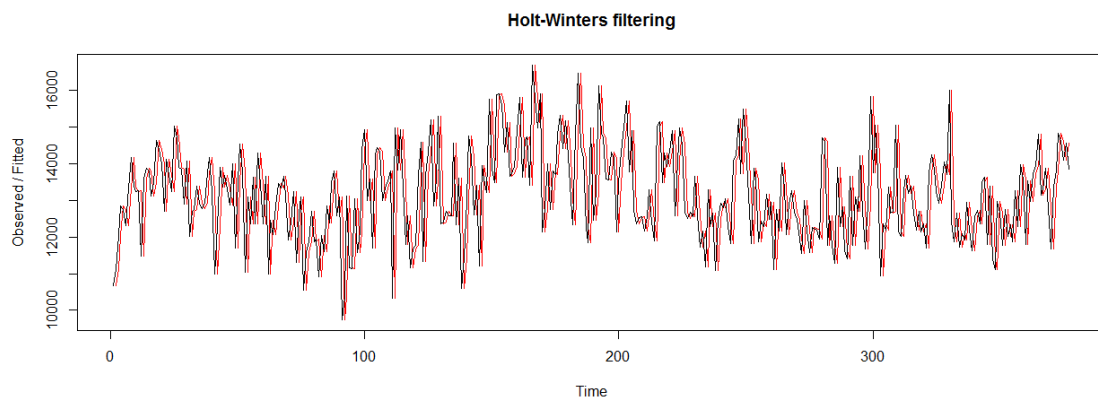


Abbildung 81: Holt-Winters mit Niveaufaktor von 1

Es folgt ein Plot mit $\alpha = 0,5$:

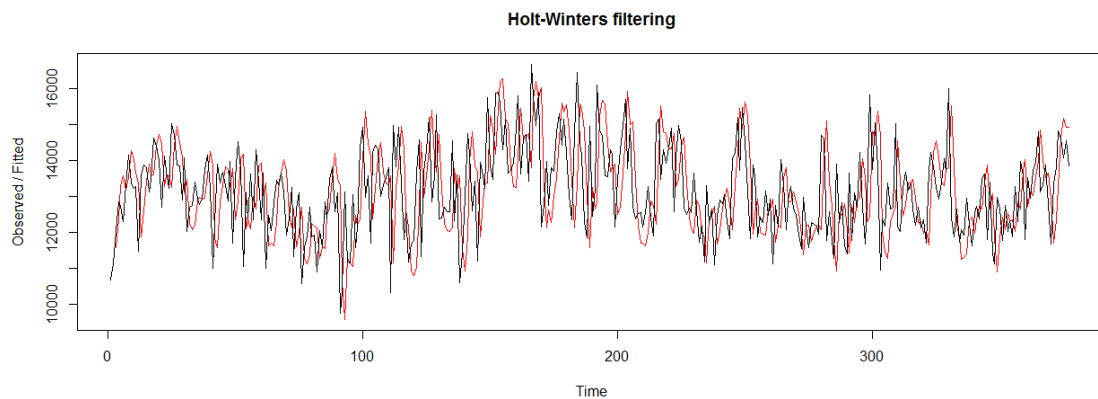


Abbildung 82: Holt-Winters mit Niveaufaktor von 0,5

Mit diesem ersten Holt-Winters-Schritt wird eine Trainingsgrundlage der aktuell vorliegenden Energieverbräuche dargestellt. Auf Basis dieser wird dann ein Prognosewert geliefert. Eine beispielhafte Prognose für die nächsten 30 Tage mit einem Konfidenzintervall von 95 % ist folgend anhand des Holt-Winters-Verfahrens in R berechnet und im Plot 83 dargestellt worden:

Listing 17: Holt-Winters-Verfahrens in R

```
forecast <- predict(HW, n.ahead = 30, prediction.interval = T, level = 0.95)
plot(HW, forecast)
plot(HW, forecast)
```

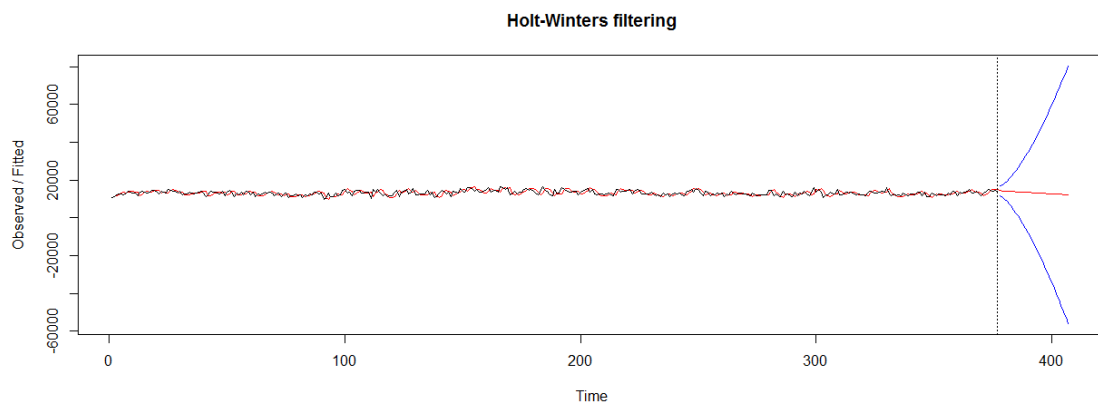


Abbildung 83: Holt-Winters-Prognose für die nächsten 30 Tage

Am Ende, also nach 365 Tagen, sieht man, wie die Prognose sein könnte, dennoch aber

im Extremfall große Abweichungen (siehe Intervall in blau) möglich sind [Reg09] [Dav] [Fre03].

Hidden Markov Model In R werden mit dem Package *depmixS4* (dependent mixture model) die Funktionalitäten der Hidden Markov Model-Methode (HMM) zur Verfügung gestellt. Folgende Ziele werden mit diesem Package verfolgt:

1. Parameter abschätzen, die allgemeinen linearen (Un)Gleichheitsbeschränkungen unterliegen
2. Übergangsmodelle mit Kovariaten anpassen, d. h. einen zeitabhängigen Übergangsmatrizen haben
3. Kovariaten in die vorherigen oder anfänglichen Zustandswahrscheinlichkeiten aufnehmen
4. Den Nutzern erlauben, einfach uni- oder multivariate Antwortverteilungen und neue Übergangsmodelle hinzuzufügen, z. B. Modelle der kontinuierlichen zeitlichen Beobachtungen

Man spricht bei einem Hidden Markov Model auch von einem Dynamischen bayesschen Netz. Bei dieser Methode bestehen zufällige Zustandsübergänge (Übergangswahrscheinlichkeiten hängen nur vom aktuellen Zustand ab). Die inneren Zustände und der aktueller Zustand sind verborgen. Allerdings sind beobachtbare Ausgabesymbole (Emissionen) mit bestimmter Auftrittswahrscheinlichkeit zu verzeichnen. HMM bietet die Möglichkeit eine Vorhersage des am wahrscheinlichsten eintretenden Zustands zu treffen [Ing10] [Ing16] [Luk16] [Gek14]. Es folgt eine beispielhafte Darstellung 84 eines Hidden Markov Modells:

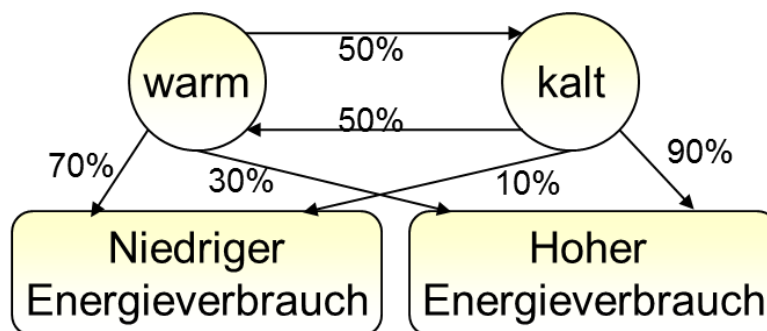


Abbildung 84: Hidden Markov Modell (eigene Darstellung)

Auf eine Energieprognose bezogen, benötigen wir nicht nur die Zustände, die vorherzusagen sind (z. B. warm und kalt), sondern in einem zweiten Schritt auch den entsprechenden Energieverbrauch (wenig/viel Energieverbrauch bzw. konkret prognostizierbare Energieverbrauchszahlen). Es besteht die Herausforderung entsprechende Variablen und die Übergangswahrscheinlichkeiten der Zustände zu Beginn sinnvoll zu wählen. In weiteren Forschungsarbeiten kann dieser Ansatz genauer untersucht und auf Umsetzbarkeit im Rahmen eines Nachfolge-Projekts geprüft werden.

6.5.4. Evaluation

Für die implementierten Modelle wurde ein Konzept entwickelt, welches sich mit der Bewertung von Modellen, auf Basis von Machine Learning-Algorithmen befasst. Darüber hinaus ein konzeptioneller Ansatz für die Erstellung eines intelligenten Algorithmus zur Identifikation der besten Prognosezahl. In diesem Abschnitt wird die Entwicklung und Umsetzung/mögliche Umsetzung näher erläutert.

Ansatz für die Bewertung der Machine Learning-Algorithmen Für die Bewertung der untersuchten Machine Learning-Algorithmen wird ein Konzept benötigt. In diesem muss festgelegt werden, unter welchen Aspekten und vor allem, mithilfe welcher Kennzahlen diese bewertet werden können, um damit eine Entscheidung für den besten Algorithmus treffen zu können.

Auswahl der Methode Zunächst musste eine Methode ausgewählt werden, welches es erlaubt eine Entscheidung für oder gegen die Machine Learning-Algorithmen zu treffen. Hierbei wurde für die Multikriterielle Entscheidungsanalyse (MCDA - Multi Criteria Decision Analysis) entschieden, welche zur Entscheidungsfindung in komplexen Situationen dient. Die Vorgehensweise bei der Entwicklung eines solchen Entscheidungsmodells besteht aus mehreren Schritten [swo16].

- **Identifizierung und Festlegung eines Ziels**

In diesem Fall ist das Ziel das beste Prognosemodell auszuwählen. Wichtig dabei ist, dass die Prognose eine hohe Zuverlässigkeit aufweist und wenige Fehler, zwischen prognostizierten und dem tatsächlichen Wert, aufweist [swo16].

- **Strukturierung des Entscheidungsproblems**

Hier gilt es, dass entscheidungsrelevante Kriterien identifiziert werden. Anhand dieser kann im späteren Verlauf die Auswahl erfolgen. Die Erläuterung der ausgewählten Kriterien erfolgt zu einem späteren Zeitpunkt [swo16].

- **Zerlegung des Entscheidungsproblems**

Die zu Beginn festgelegten Ziele und ausgewählten Kriterien müssen logisch miteinander Verknüpft und eine qualitative Gewichtung ermöglicht werden [swo16].

- **Auswahl der besten Alternative**

Nachdem das Ziel, die geeigneten Kriterien und die Gewichtung festgelegt wurden, kann nun eine Entscheidung erfolgen, welche auf die vorherigen Punkte aufbaut [swo16].

Im folgenden Verlauf werden die aufgezeigten Schritte für ein Entscheidungsmodell näher und Fall bezogen erläutert.

Entscheidungsrelevante Kriterien Als entscheidungsrelevante Kriterien stehen eine Vielzahl von verschiedenen Kennzahlen zur Verfügung. In diesem Fall wurden vier verschiedene Kennzahlen ausgewählt, welche aus den Bereichen Fehlermaße und Prognosegüte stammen. Hierbei muss erwähnt werden, dass es sich bei dieser Entscheidung um eine rein objektive handelt.

1. R^2 : Bei dem R^2 handelt es sich um ein Gütemaß, welches angibt wie viele Daten durch das Modell erklärt werden können. Je höher der Wert ist, desto besser sind die prognostizierten Werte. Dabei ist ein Wert von 1 das beste und 0 entspricht keinem Modell. Berechnet wird der R^2 -Wert durch die folgende Formel [Lei17]:

$$R^2 = r^2 = \frac{\sum (\hat{y}_i - \bar{y})^2}{\sum (y_i - \bar{y})^2} = \frac{d_{\hat{y}}^2}{d_y^2} = \frac{\hat{b}^2 d_x^2}{d_y^2} = 1 - \frac{\sum_{i=1}^n \hat{u}_i^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (8)$$

2. **Mean Squared Error (MSE)**: Durch den Mean Squared Errors wird angegeben, inwieweit die Prognose von dem tatsächlichen Wert abweicht. Dabei wird die Abweichung quadriert, aufsummiert und durch die Anzahl der Beobachtungen geteilt. Dadurch wird verhindert, dass sich positive und negative Abweichungen gegenseitig aufheben [Sö09].

Berechnet wird der Mean Squared Error mit der folgenden Formel [Lei17]:

$$d_x^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 = \frac{1}{n} \sum_{i=1}^n x_i^2 - \left(\frac{1}{n} \sum_{i=1}^n x_i \right)^2 = \bar{x}^2 - \bar{x}^2 \quad (9)$$

$$d_x = \sqrt{d_x^2}$$

3. **Mean Absolute Percentage Error (MAPE):** Der Mittlere absoluter prozentualer Fehler drückt die Genauigkeit als Prozentsatz des Fehlers aus. Da diese Zahl einen prozentualen Wert darstellt, kann dieser einfacher nachvollzogen werden. Wenn der MAPE-Wert beispielsweise fünf beträgt, weicht die Prognose im Durchschnitt um 5 % vom tatsächlichen Wert ab. Dabei wird nicht berücksichtigt, ob es sich dabei um einen positiven oder negativen Wert handelt [Sö09]. Berechnet wird der MAPE folgendermaßen [Lei17]:

$$MAPE = \frac{100}{n} \sum_{t=1}^n \left| \frac{A_t - F_t}{A_t} \right|, \quad (10)$$

4. **Theilsche Ungleichheitskoeffizient (TUK):** Der Theilsche Ungleichheitskoeffizient dient zur Beurteilung der Prognosegüte. Treffen alle Prognosen zu, dann nimmt der Koeffizient den Wert null an. Je größer der Wert, desto schlechter ist die Prognosequalität. Nimmt der TUK den Wert eins an, dann ist die Prognose nicht besser als eine sogenannte naive Prognose, was bedeutet, dass die Veränderungsrate der betrachteten Variablen unverändert sind. Als naive Prognose wird die Konstanz der Niveauewerte bezeichnet [Sö09], [Lei17].

$$TUK = \sqrt{\frac{\sum (\Delta P_v - \Delta R_v)^2}{\sum \Delta R_v^2}} = \sqrt{\frac{\sum [\hat{y}_v - y_v]^2}{\sum [y_v - y_{v-1}]^2}} \quad (11)$$

Gewichtung Nachdem die vier ausgewählten Kriterien ausgewählt und berechnet wurden, gilt es nun diese zu gewichten. Hierbei muss betont werden, dass es sich ausschließlich um eine objektive Entscheidung handelt.

Durch die Gewichtung der einzelnen Kennzahlen besteht die Möglichkeit ein Punktesystem anzuwenden. Dadurch wird die Entscheidungsfindung vereinfacht und alle ausgewählten Kriterien werden in dem Maße berücksichtigt, wie sie in den Zielen definiert wurden.

R^2	MSE	MAPE	TUK
$\frac{4}{12}$	$\frac{3}{12}$	$\frac{2}{12}$	$\frac{3}{12}$

Abbildung 85: Gewichtung der Kriterien

Die Darstellung 85 enthält die von uns festgelegte Gewichtung der Kennzahlen. Für die Auswahl des Prognosemodells steht die Prognosequalität deutlich im Vordergrund, deshalb bekommt diese Kennzahl die höchste Gewichtung. Sowohl der MSE als auch der TUK sind ebenfalls zwei wichtige Kennzahlen und werden daher mit jeweils mit $\frac{3}{12}$ gewichtet. Die geringste Gewichtung erhält der MAPE. Die Gewichtung der Kennzahlen ist der Tabelle zu entnehmen.

Nachdem die Gewichtung festgelegt wurde, rückt das Punktesystem in den Fokus. Hierbei wird so verfahren, dass der beste Wert der einzelnen Kennzahlen vier Punkte bekommt und der schlechteste einen. Diese vergebenen Punkte werden im Anschluss mit den Gewichtungen multipliziert. Mithilfe dieser berechneten Zahl soll das geeignete Modell ausgewählt werden.

Ergebnis Die Ergebnisse der berechneten Kennzahlen für die jeweiligen Methoden werden in tabellarischer Form dargestellt.

Für das Verbrauchsmodell wurden die Methoden lineare Regression, Holt-Winters und Naive Bayes untersucht. Die Berechnung der vier ausgewählten entscheidungsrelevanten Kennzahlen erfolgte für jede einzelne Methode. Die Ergebnisse sind der nachstehenden Abbildung 86 zu entnehmen. Mit einem Kreis markiert sind die besten Ergebnisse für die jeweilige Kennzahl.

Prognose- gütemaß Prognosemethode	R ²	MSE	MAPE	TUK
Lineare Regression	0,3866	0,0708	Inf	1,99
Holt-Winters	0,8859	2,8003	9,37 %	0,13
Naive Bayes	0,9007	3,5483	5,16 %	5,37

Abbildung 86: Ergebnisse des Verbrauchsmodells

In der Gesamtheit, aufgrund der berechneten Kennzahlen und deren Gewichtung, schneidet das Verfahren *Naive Bayes* am besten ab.

Für das Produktionsmodell wurden die Machine Learning-Algorithmen lineare Regres-

sion, Random Forest und Neuronale Netz untersucht. Die Kennzahlen wurden mit den bereits beschriebenen Formeln berechnet. Die jeweiligen Ergebnisse der einzelnen Kennzahlen sind der nachstehenden Grafik 87 zu entnehmen.

Prognose- gütemaß Prognosemethode	R^2	MSE	MAPE	TJK
Lineare Regression	0,8908	0,0593	4,19 %	0,51
Random Forest	0,9122	0,0526	5,83 %	0,29
Neuronales Netz	0,9318	0,0992	0,03 %	0

Abbildung 87: Ergebnisse des Produktionsmodells

Hervorzuheben ist in diesem Fall das Neuronale Netz, welches mit einem Prognosewert von 93 % und einer durchschnittlichen Abweichung der Prognose von 0,03 % vom tatsächlichen Wert die beste Methode für die Prognose der Energieproduktion darstellt.

Konzept zur Erstellung eines intelligenten Algorithmus Um lediglich eine zuverlässige Prognosezahl anhand der im Verbrauchs- und Produktionsmodell implementierten Prognosemethoden zu ermitteln und zu liefern, soll mithilfe eines oder mehrerer entscheidenden Gütemaße ein Prognosewert identifiziert werden. Dazu wurde ein Konzept für eine Art intelligenten Algorithmus erstellt, welcher die zu dem Zeitpunkt beste Prognosezahl bestimmen soll und somit ein zuverlässiges Ergebnis durch den Vergleich der verschiedenen Prognosemethoden (Lineare Regression, Holt-Winters, etc.) zustande kommt.

Tag 1:

- Basis: historische Daten
- Verschiedene Prognosemethoden liegen vor: HW, Naive Bayes und Lineare Regression
- Verschiedene Prognosegüteregebnisse liegen vor: Bestimmtheitsmaß R^2 , MAPE, MSE und Theilsche Ungleichheitskoeffizient

- R^2 als bestimmende Prognosegüte → die am besten aus den Prognosemethoden resultierende Prognosegüte anhand des R^2 soll gewählt werden (Prognosezahl) nach der zugrundeliegenden Prognosemethode)

Alle weiteren Tage:

- Im Laufe der Zeit kommt die intelligente Algorithmik zustande
- Resultat aus dem Test: Naive Bayes als beste Prognosemethode herausgestellt → allerdings handelt es sich dabei um den Durchschnittswert des Prognoseintervalls → eher ungeeignet (nur 3 Klassen)
- Prognosegütemaß: Tendenz zu Theilsche Ungleichheitskoeffizient: Laut Andres und Spiwok [Pet00] gilt der Theilsche Ungleichheitskoeffizient als gutes Prognosegütemaß, der für viele Fragestellungen richtige Einschätzungen abgibt. Eigenschaften, die im Gegensatz zu den anderen Prognosegütemaßen nicht vorhanden sind:
 - impliziter Vergleich mit der naiven Prognose
 - Keine gegenseitige Aufhebung der positiven und negativen Abweichungen des Schätzers
 - Große Fehler werden überproportional gewichtet
 - Prognosen mit unterschiedlichen Prognosevariablen werden durch die Normierung miteinander vergleichbar gemacht
 - Henschel: seit 1979 TUK wohl bekannteste Prüfmaß der Prognosebewertung [Hen]

Für das Produktionsmodell wird ebenfalls ein Konzept für einen intelligenten Algorithmus für die beste Prognosezahl konzeptionell erstellt.

Tag 1:

- Basis: historische Daten
- Verschiedene Prognosemethoden liegen vor: Neuronale Netze, Random Forest und Lineare Regression
- Verschiedene Prognosegüteergebnisse liegen vor: Bestimmtheitsmaß R^2 , MAPE, MSE und Theilsche Ungleichheitskoeffizient
- R^2 als bestimmende Prognosegüte → die am besten aus den Prognosemethoden resultierende Prognosegüte anhand des R^2 soll gewählt werden (Prognosezahl nach der zugrundeliegenden Prognosemethode)

Alle weiteren Tage:

- So wie beim Verbrauchsmodell soll der Theilsche Ungleichheitskoeffizient als Prognosegüte-Basis gewählt werden

Dieser Ansatz soll als Entwurf für eine mögliche Vorgehensweise für die Identifikation der besten Prognosezahl dienen. Darüber hinaus sind natürlich weitere Ansätze möglich, wie z. B. der Wahl des entscheidenden Prognosegütemaßes (z. B. MAPE, MSE, usw.) oder einer Kombination verschiedener Gütemaße, die zu verschiedenen Zeitpunkten und je Zustand (Wertintervalle, Wertsprünge, etc.) am besten geeignet sind.

6.6. Technische Umsetzung

Wie bereits im Grobkonzept beschrieben sollte die technische Umsetzung zunächst mithilfe der SAP HANA-Datenbank realisiert werden. Aufgrund von Einschränkungen bei der Nutzung, welche in diesem Kapitel aufgezeigt werden, wurde dieser Ansatz verworfen und sich für den Microsoft R-Server entschieden. Daher befasst sich dieses Kapitel insbesondere mit der Einbindung der Technologie des Microsoft R Servers für die Berechnung von Prognosen. Microsoft R Server fungiert dabei als zentrale Anlaufstelle für die Datenanalyse auf Basis der statistischen Programmiersprache R. Die auf dem R Server veröffentlichten Modelle wurden danach in eine Applikation integriert, in der mit den Ergebnissen der Prognose weitergearbeitet werden kann. Zunächst wurde dafür lediglich die Funktionalität in einer .NET Konsolenanwendung getestet, anschließend sollten die Modelle in einem Dashboard auf Basis von Razor MVC5 genutzt werden. Schließlich wurden von allen Teilgruppen beschlossen, eine gemeinsame Applikation mit React Javascript zu erstellen.

6.6.1. Problemstellung

Um die Prognosemodelle in einem möglichen Produktivszenario zu nutzen, ist es praktikabler die Prognosen per Remotecall abzurufen als sie lokal beim Nutzer durchführen zu lassen. Zudem steigt mit den Datenmengen auch die Auslastung der Hardware, was auf Systemen mit wenig Rechenleistung zu Abstürzen führen kann. Eine Lösung, welche die Rechenarbeit auf einem Server mit entsprechenden Kapazitäten durchführen kann, wurde daher gesucht. Der Nutzer einer App für Prognosen soll diese auf Basis seiner Daten von unserem Modell erhalten, aber nicht in direktem Kontakt mit der Berechnung über R stehen. Um das zu realisieren, müssen die Daten vorher gespeichert werden und an den Server geschickt werden, welcher anschließend eine Prognose als Antwort liefern soll. Dafür wurden zwei Technologien für die Berechnung dieser Prognose untersucht.

6.6.2. SAP HANA

Der erste Ansatz verfolgte eine Realisierung innerhalb der SAP HANA. Im Folgenden wird auf die Funktionsweise der SAP HANA und die Realisierung der R-Integration näher eingegangen.

Funktionsweise Mit der SAP HANA wird eine ACID-kompatible (atomicity, consistency, isolation und durability) Datenbank mit Anwendungsservices, Hochgeschwindigkeitsanalysen und flexiblen Werkzeugen für die Datenerfassung auf einer einheitlichen In-Memory-Plattform kombiniert. Die In-Memory-Technologie ist auf dem besten Weg, die altbewährten Datenbanksysteme zu verdrängen, so Ayfer Özdem von Crisp Research. Relationale Datenbanken haben den bisherigen Standard gebildet. Oftmals wurden MS SQL Server oder Oracle Datenbanken in Unternehmen eingesetzt. Mit SAP HANA sollen In-Memory-Datenbanken eine optimierte Datenverarbeitung liefern [Jü16].

Die SAP HANA-Datenbank dient als Speicherort für die in Anwendungen genutzte Daten und als ausführendes Element. Zudem kann die Datenbank als modernes Data Warehouse genutzt werden, welches Daten aus einer Vielzahl von Quellen mit Live-Transaktionsdaten integriert und dadurch die Möglichkeit bietet, immer die aktuellsten Daten einzusehen. Mithilfe von integrierten Anwendungsservices wird sowohl die Entwicklung, als auch die Einführung neuer Geschäftsanwendungen, welche auf die Daten zurückgreifen und verschiedene Datenverarbeitungsfunktionen nutzen, vereinfacht [SAP17b].

R-Integration Der Vorteil im Bezug zu unserem Anwendungsfall ist das vorhandene DBMS und die mögliche R-Integration. Damit der R-Code in der SAP HANA ausgeführt werden kann, wird dieser als Prozedur eingebettet. Für die Ausführung wird auf die R-Umgebung zurückgegriffen. Damit R unterstützt werden kann, muss innerhalb der Berechnungsmaschine innerhalb von SAP HANA eine Erweiterung stattfinden. Die Berechnungsmaschine unterstützt Datenflussdiagramme, welche logische Datenbankausführungspläne beschreiben. Innerhalb dieses Diagramms kann jeder Knoten unter anderem für eine benutzerdefinierte Operation stehen, beispielsweise einem R Operator [SAP17a].

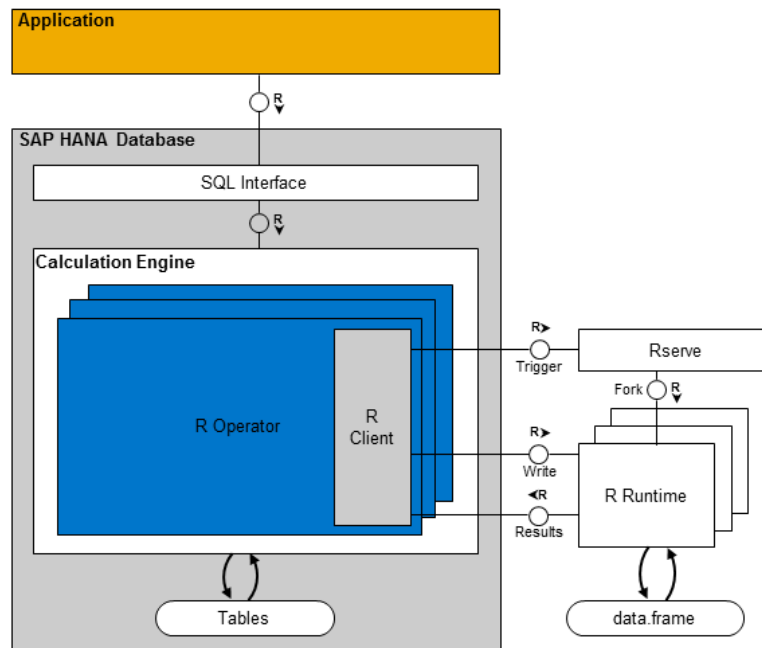


Abbildung 88: Kommunikationsprozess [SAP17b]

Die Abbildung 88 zeigt die drei Hauptkomponenten der Integration und den Kommunikationsprozess: die Applikation, welche SAP HANA basiert ist, die Datenbank und die R-Umgebung [SAP17a].

Wird ein R-Operator bei der Ausführung erreicht, wird eine Anfrage über den Rserve-Mechanismus an die Calculation Engine versendet, um einen Prozess auf dem R-Host zu erstellen. Im nächsten Schritt überträgt der Client den R-Code und die Eingabetabellen auf einen R-Prozess und löst somit eine Ausführung aus. Sobald diese abgeschlossen ist, wird der resultierende Datenrahmen an die Calculation Engine zur Umwandlung zurückgegeben [SAP17a].

Im Bezug auf das Thema Predictive Analytics, kann in der SAP HANA die Bibliothek PAL (Predictive Analytics Library) eingebunden werden. Mit dieser Bibliothek ist es möglich analytische Algorithmen (über 60) direkt auszuführen und in der SAP HANA-Anwendung einbinden zu lassen (über SQL Script-Prozeduren). Die Analysealgorithmen sind in neun Data Mining-Kategorien unterteilt (Clustering, Classification, usw.). Die R-Integration ist dabei eine separate Komponente in der SAP HANA Platform. Die folgende Abbildung 89 verdeutlicht den entsprechenden Aufbau [Cha13].

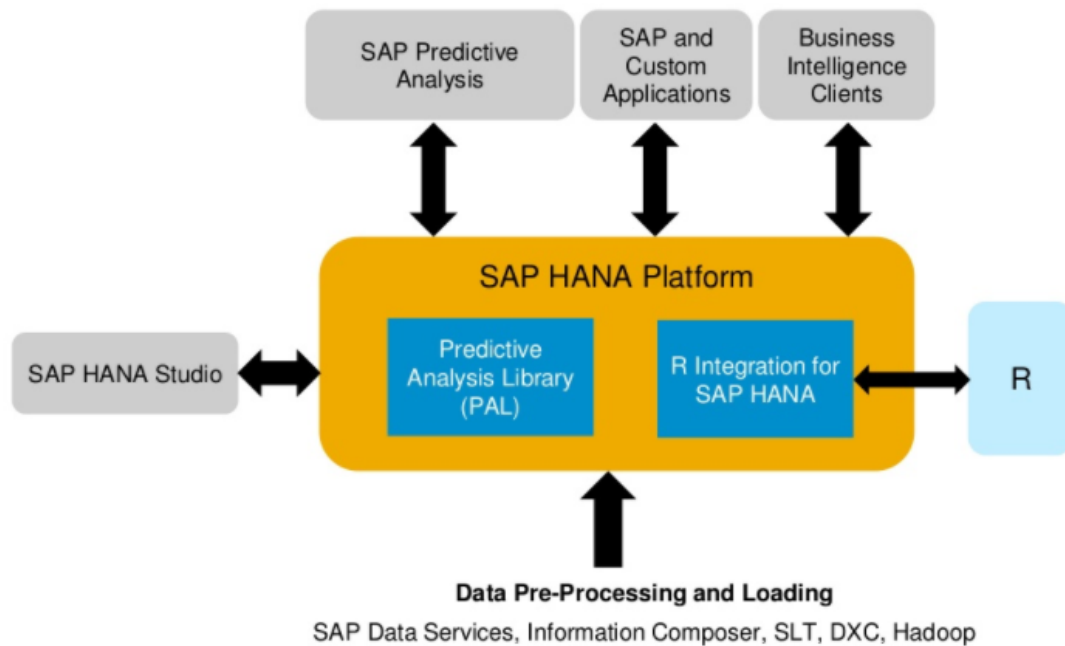


Abbildung 89: Predictive Analytics Library in SAP HANA [Cha13]

Da die In-Memory-Technologie verwendet wird, bestehen bei der Berechnung eine schnellere Zugriffszeit und der Datentransfer zwischen Datenbank und Applikationsserver ist dabei deutlich reduziert. Nur das Ergebnis wird an den Applikationsserver übergeben [Cha13].

Auswahl Die Nutzung von R und der SAP HANA ist sehr eingeschränkt. Im Folgenden werden einige Einschränkungen genannt [SAP17a]:

- Die Integration von R wurde von SAP nicht ausreichend getestet, daher kann nicht auf vorhandenes Wissen zurückgegriffen werden.
- Auf der Datenbank kann ausschließlich mit SQL Scripten gearbeitet werden. Was dazu führt, dass jegliche Tabellenarten dementsprechend manuell angepasst werden müssen.
- Die Benennung von Variablennamen muss nach festgelegten Regeln erfolgen. Bspw. dürfen keinerlei Großbuchstaben genutzt werden.
- R-Funktionen müssen mindestens über ein Ergebnis in Form eines Dataframes enthalten.

- Faktorspalten können nur als Zeichenvektoren abgerufen werden. Einige Funktionen erfordern unter anderem String-Spalten von Dataframes als Faktor, so dass diese zwingend umgewandelt werden müssen.

Aus den genannten Gründen und aufgrund des Ziels eine eigene plattformunabhängige Applikation zu entwickeln, wurde keine Integration auf der SAP HANA-Datenbank realisiert und der Microsoft R Servers wurde in Betracht gezogen. Zudem stellte der Microsoft R Server zum Zeitpunkt der Entscheidungsfindung eine State of the Art-Lösung dar, was für die Auswahl dieses Servers spricht. Im folgenden Kapitel wird die Funktionalität des Microsoft R Servers näher erläutert.

6.6.3. Microsoft R Server

Der R Server bietet eine Plattform um R-Modelle mit besserer Performance laufen zu lassen. Des Weiteren kann durch die Operationalisierung die Arbeit effizienter gestaltet werden, da mit dem R Server eine zentrale Stelle geschaffen wird. In der nachfolgenden Abbildung 90 werden die verschiedenen Komponenten des R Servers dargestellt:

Components of R Server

Components	Description
Microsoft R Open	Microsoft's distribution of open source R. This distribution ships standalone and as a component of Microsoft R Client and Microsoft R Server.
Operationalization	An operationalization engine only in R Server used to deploy R script or code as a web service with support for remote runtime execution. Console users can exercise the functions in mrsdeploy package . Developers can use the Swagger-APIs to create programmatic solutions.
ScaleR	ScaleR is a high performance computing and analytical engine used to partition massively large datasets into smaller chunks, distributed and analyzed in parallel, often on multiple nodes or on database platforms like SQL Server and Teradata. ScaleR is an R Server feature, but it also ships in R Client with limits on data size and processor utilization. ScaleR functions are provided by the RevoScaleR package .
Machine learning algorithms	State-of-the-art machine learning algorithms are now available in Microsoft R. You can use these functions in R code or script for performing machine learning on a standalone R Server. Machine learning algorithms are also available in R Client, subject to data size limits (in-memory only) and processor limits (2). Functions are provided by the MicrosoftML package .
Other packages	Additional packages are distributed with R Client and R Server, such as RevoPemaR . For the complete list, see Package reference on MSDN .
Platform-specific Components	Windows, Linux, and Hadoop components are only available in R Server. Cloud services, like Azure HDInsight, integrate R Server internally so that you don't have to provision or manage the server manually. Platform-specific components are available when you install R server on that platform. For more information, see installation links below.
Rgui.exe and Rexe	R Server includes console applications for command line execution in a local session.

Abbildung 90: Komponenten des Microsoft R Servers [Mic17c]

Um Probleme bei der Verarbeitung großer Datenmengen zu verhindern, wie z. B. Kapazitäts- oder Zeitengpässe, bietet es sich an den R Server zu nutzen. Mit der Infrastruktur des R Server bekommen statistische Analysen die notwendige Kapazität und können besser skaliert werden. Zusätzlich wurden in dem neuesten Release Funktionen zum Machine Learning und zur Operationalisierung beigefügt. Ein Data Scientist kann so z. B. in der IDE Visual Studio ein Skript erstellen und an den Server schicken. Der Server verarbeitet das Skript und schickt ein Ergebnis zurück. Dazu werden, wie in der Grafik zu den Komponenten dargestellt, andere Packages verwendet. Mit dem ScaleR Package kann eine bessere Performance erzielt werden als mit den üblichen Open Source Packages. In der folgenden Grafik wird die Enterprise Architektur eines R Servers dargestellt:

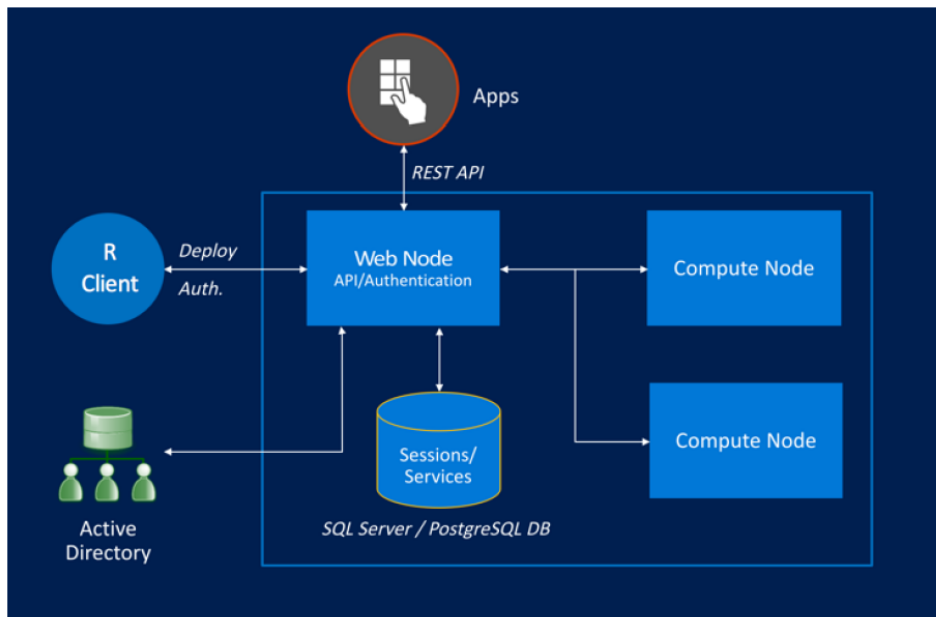


Abbildung 91: Deployment Microsoft R Server [Mic17a]

Der R Client stellt z. B. einen Data Scientist dar, der lokal in einer IDE arbeitet und per Remote Skripte auf dem R Server ausführen können. Die Web Node ist für die Authentifizierung des Clients für den R Server, z. B. in Zusammenarbeit mit dem Active Directory, zuständig und stellt die API für den Client dar. Wenn die Web Node hochskaliert wird, können mehrere API Requests ermöglicht werden. Außerdem ist bei mehr als einer Web Node ein remote SQL Server oder eine PostgreSQL Datenbank erforderlich, um die Daten untereinander verfügbar zu machen. Unabhängig davon gibt es Compute Nodes für das Berechnen der R Skripte, diese können dabei hochskaliert werden, damit mehrere Skripte ausgeführt werden können.

Ein weiteres wichtiges Feature des neuesten R Releases ist die Operationalisierung:

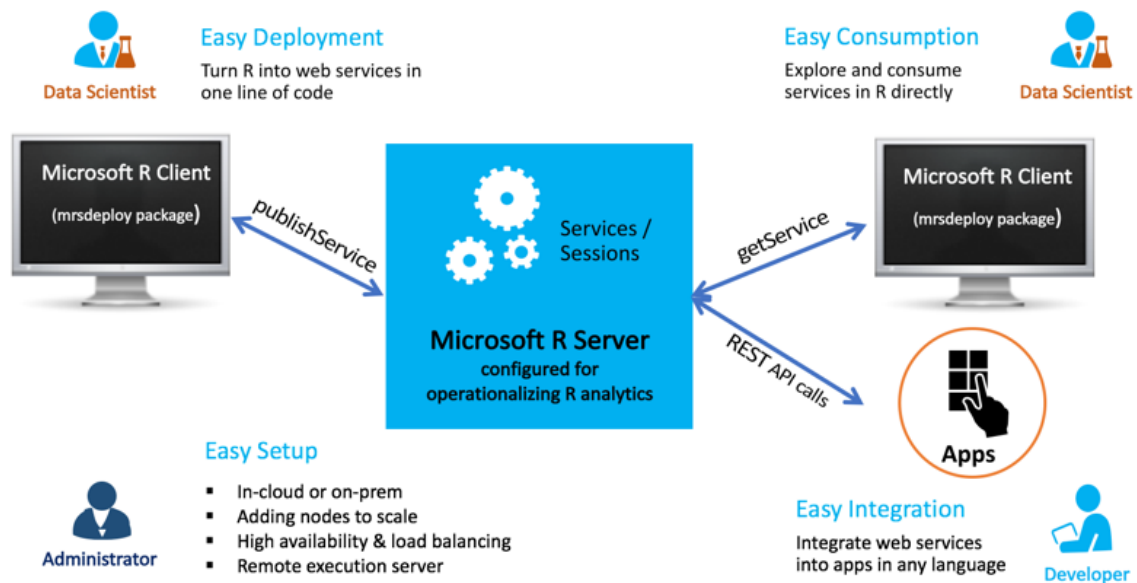


Abbildung 92: Operationalisierung von R WebServices [Mic17b]

Mithilfe dieses Features können Data Scientists Skripte als Web Services veröffentlichen. Dieser Web Service kann dann von anderen Data Scientists über die R Client IDE oder über Swagger-based APIs mit nahezu jeder Programmiersprache genutzt werden. Somit können R Skripte bzw. Services über remote ausgeführt werden.

6.6.4. Proof of Concept

Für das *Proof of Concept* wurde das erste Modell der linearen Regression zum Verbrauch verwendet. Das Modell wurde auf dem R Server als Webservice veröffentlicht. Mittels Login-Daten kann eine Verbindung zum R Server aus der Entwicklungsumgebung hergestellt werden.

Im nächsten Schritt konnte das zuvor erstellte Modell auf dem R Server veröffentlicht werden. Als Parameter für den Service müssen Input Variablen hinzugefügt werden, welche zuvor 6.5 als wichtige Indikatoren bezüglich der Güte der Prognose des Modells identifiziert wurden. Nachfolgend wird der Code zum Veröffentlichen des WebServices dargestellt:

Listing 18: Veröffentlichen des WebServices

```
api <- publishService(
  "prototyp",
  code = prototyp,
```

```

model = fit ,
inputs = list(date = "character", avg = "numeric",
min = "numeric", max = "numeric", ma= "numeric"
feuchte= "numeric"),
outputs = list(answer = "numeric"),
v = "v1.0.0",
alias = "prototyp"
)

```

Anschließend wurde der Webservice in der R Entwicklungsumgebung per Remotesteuerung ausgeführt, dabei kam das Ergebnis von 7796 kWh-Verbrauch zustande, wie dem nachfolgendem Code zu entnehmen ist.

Listing 19: Webservice

```

api <- getService("prototyp")
result <- api$prototyp("2015-24-12", 7800, 7100, 8900, 7900, 80)
> api <- getService("prototyp")
> result <- api$prototyp("2015-24-12", 7800, 7100, 8900, 7900, 80)
+ print(result$output("answer"))

[1] 7796

```

Als Entwicklungsumgebung wurde Visual Studio 2015 ausgewählt. Dazu wurde eine C# Konsolenanwendung mit dem Namen PrototypeProject erstellt. Damit auf die Rest Schnittstelle des Webservices zuzugreifen werden kann, musste zunächst eine Rest Client Library, mithilfe des NuGet Packages AutoRest erstellt werden. Neben diesem Package mussten dem Projekt noch die Packages Newtonsoft.Json sowie Microsoft Clients Active Directory hinzugefügt werden. Zusätzlich wird die swagger.json Datei des Webservices benötigt. Die Swagger.json Datei des zu konsumierenden Services wird schließlich im Projektordner hinterlegt. Nach Aufruf der NuGet Package Manager Console kann mit dem folgenden Befehl die Client Library erzeugt werden, welche es ermöglicht den REST Service zu erstellen:

Listing 20: REST Service

```

AutoRest.exe -CodeGenerator CSharp -Modeler Swagger -Input swagger.json -
Namespace MyNamespace

```

Anschließend befindet sich im Visual Studio Projekt die Library im Ordner *Generated*. Dieser Ordner muss nun dem Projekt hinzugefügt werden. Anschließend wurde das Ergebnis in einer Windows-Form dargestellt. Dieses Resultat bildet die Grundlage für weitere Entwicklungen mit dem Microsoft R Server.

6.6.5. Predictive Dashboard

Um die Ergebnisse der Prognosen angemessen darzustellen, wurde entschieden ein Dashboard für den Nutzer zu erstellen. Auf diesem Dashboard sollen die Prognosen für Verbrauch und Produktion sowie weitere Infos (z. B. Trendchart des Verbrauchs) angezeigt werden. Im Folgenden werden die verschiedenen Schritte, um dieses zu erstellen, näher erläutert.

Konzept zur Auslesung von Smart Meter Daten für das Modell Dieses Kapitel stellt die ursprüngliche Kozeptionierung des Datenflusses der Rohdaten vom Smart Meter bis hin zur Applikation dar. Des Weiteren wurden Schritte für die Berechnung der Prognosen für den Folgetag festgelegt. Aufgrund weiterer Entwicklungen des Modells sind Wetterdaten für die Prognose nicht mehr von Relevanz.

Notwendige Daten sollen vom Smart Meter ausgelesen, verarbeitet und anschließend über den Web Service vorhergesagt werden.

Alternative: Skript schreiben um Daten auszulesen

- Smart Meter Daten zunächst als Schleife simulieren
- Serielle Schnittstelle des Pis nutzen um Daten ab zugreifen
- Parameter in der jeweiligen Programmiersprache berechnen
- Daten um 00:00 für den Folgetag berechnen/Moving Average verschieben
- Daten jeweils für 24 Stunden speichern um daraus Parameter zu berechnen
- Moving Average verschieben

Umsetzung Die Umsetzung soll mittels einer C# Applikation, welche für das Auslesen der Daten über die serielle Schnittstelle und anschließende Berechnung der Parameter zuständig ist, realisiert werden. Mit der auf einem Raspberry Pi integrierte Entwicklungsumgebung MonoDevelop können .Net Sprachen plattformübergreifend kompatibel gemacht werden. Via http Post werden die Daten an die Web Applikation gesendet, um den Service mit den Parametern zu konsumieren. Dazu ist es notwendig, dass eine API eingebaut wird, welche die Verarbeitung der Daten ermöglicht. Die Berechnung der Parameter soll durch die Programmiersprache C# erfolgen.

Alternativ könnte ein R Webservice erstellt werden, um die Parameter in der R-Umgebung zu berechnen und von dort aus wieder zurück zu schicken. Die Daten würden für 24 Stunden gespeichert werden und werden in Form einer Liste an den Webservice übergeben. Die

Rückgabeparameter bilden die Werte, welche für das Vorhersagemodell benötigt werden. Daraufhin müssten die Daten an die WebApp zurückgesendet werden, um die Vorhersage zu berechnen und auszugeben.

Fragen bezüglich der Performance und der Umsetzung wären die Folgenden:

- In welchem Format sollen die ausgelesenen Daten aus der C# Applikation an den Server gesendet werden?
- Welche Variante ist performanter?
- Welche Variante kann die Integrität der Daten sicherstellen?

Für die Berechnung der Luftfeuchtigkeit soll die OpenWeatherMap API genutzt werden, welche einen einfachen API Zugriff ermöglicht. Darüber hinaus soll die Flächenbedeckung überprüft werden, um auch möglichst die relative Feuchte (basierend auf GPS Koordinaten) abzufragen.

Aus den vorangegangenen Beschreibungen ließe sich folgender Prozess ableiten:

Auslese der Daten über serielle Schnittstelle mit C# → Daten 24h speichern → Um 00:00 Parameter berechnen (in C# oder über Webservice) → Wetterdaten z. B. mit OpenWeatherApi auslesen (WebAPI Eingabe Option von PLZ oder Ort für Vorhersage) → Modell/Vorhersage berechnen für den übernächsten Tag)

Anforderungen Innerhalb der Gruppe wurden die Anforderungen an ein Dashboard besprochen. Nachfolgend werden die Anforderungen an das Predictive Dashboard dargestellt:

- Sprache: C# Razor MVC5
- Wofür soll das Dashboard genutzt werden?
 - Übersicht über aktuelle Prognosen für den Verbrauch und die Produktion der installierten Anlage
 - Grafiken/Charts über Verbrauchs und Produktionszahlen der vergangenen Tage
 - Implementierung von Kennzahlen (z. B. monetäres Delta aus Verbrauch und Produktion)
- Schnittstellen zum Raspberry Pi/ Microsoft R Server → Verarbeitung der Daten
- Datenbank zum Speichern der Prognose und der Rohdaten
- Lokale Applikation

- Microsoft R Server Datafile Upload

Die Sprache C# mit dem Razor MVC5 Framework wurde deshalb ausgewählt, weil innerhalb der Gruppe bereits Erfahrung in der Programmiersprache vorhanden war. Um dem Nutzer einen Mehrwert zu bieten, sollten neben den Prognosen weitere Informationen angeboten werden. Dazu gehört beispielsweise ein Trendchart über den Verbrauch, welcher es dem Nutzer ermöglicht das Verbrauchsverhalten der letzten Wochen zu reflektieren. Zusätzlich sollten Kennzahlen implementiert werden, wodurch beispielsweise das monetäre Delta aus Verbrauch und Produktion dargestellt werden kann. Ebenso kann der aktuelle Stand der Batterie der PV-Anlage angezeigt werden. Des Weiteren sollte die Applikation lokal ausgeführt werden, um gesetzliche Vorgaben bezüglich des Datenschutzes nicht zu verletzen. Aus diesem Grund sollen die Rohdaten und Prognosen ausschließlich in einer lokalen Datenbank gespeichert werden. Zudem war diese Anforderung notwendig, damit das Modell die Vorhersage auf einer größeren Datenbasis treffen kann, um somit die Güte der Prognose zu erhöhen. Die in der Datenbank gespeicherten Prognosen und Rohdaten sind auch für die Generierung eines Mehrwerts durch das Dashboard notwendig, da nur auf einer größeren Datenbasis Trendcharts des Verbrauchs und der Produktion über die letzten Wochen angezeigt werden können. Da sich im Verlauf des Testens die Berechnung der Kennzahlen für das Modell innerhalb der Sprache C# als problematisch herauskristallisiert hat, wurde ebenfalls festgelegt den gesamten Datensatz als Liste an den Microsoft R Server zu schicken. Die Berechnung des Moving Average für verschieden Zeiten z. B. drei oder sieben Tage, konnte in C# nicht umgesetzt werden. Zudem war die Berechnung auf dem R Server performanter als lokal auf dem Rechner. Unter dem Aspekt, dass die Applikation auf Geräten wie dem Raspberry Pi laufen soll, wurde daher beschlossen die gesamte Berechnung auf dem Microsoft R Server zu vollziehen.

Umsetzung Für die Umsetzung des geplanten Dashboards wurde mit C# im Razor MVC5 Framework programmiert. Da Razor MVC5 die Bootstrap Technologie nutzt, wurde nach einem Template gesucht, welches den Anforderungen entsprechend modifiziert werden kann. Dabei wurde nach Recherche im Internet das AdminLTE¹⁷ Dashboard 2 ausgewählt. Dieses Template kann kostenlos im Internet bezogen und konnte daher in das Razor MVC5 Projekt integriert werden. Zunächst wurde die grundlegende Design-Struktur für das Dashboard modifiziert. Des Weiteren wurde die grundlegende Modell-Struktur auf Basis der Smart Meter-Daten erstellt, dazu wurde mit dem Entity Framework das Backend der Applikation erstellt. Damit die Smart Meter-Daten aus einer Konsolenapplikation empfangen

¹⁷ <https://adminlte.io/themes/AdminLTE/>

werden können, wurden Schnittstellen geschaffen. Die Daten werden dazu in JSON serialisiert und nach dem Empfang der Daten im Dashboard deserialisiert. Aus den im Backend gespeicherten Smart Meter-Daten werden dann der Trendchart und das Live-Monitoring Histogramm angereichert. Die Graphen basieren auf der Javascript-Library ChartJS. Ebenso werden die Kennzahlen, wie das monetäre Delta aus den Daten im Backend, und einem statischen Preis für die Kilowattstunde berechnet. Es folgt die zugehörige Darstellung (Abbildung 94).

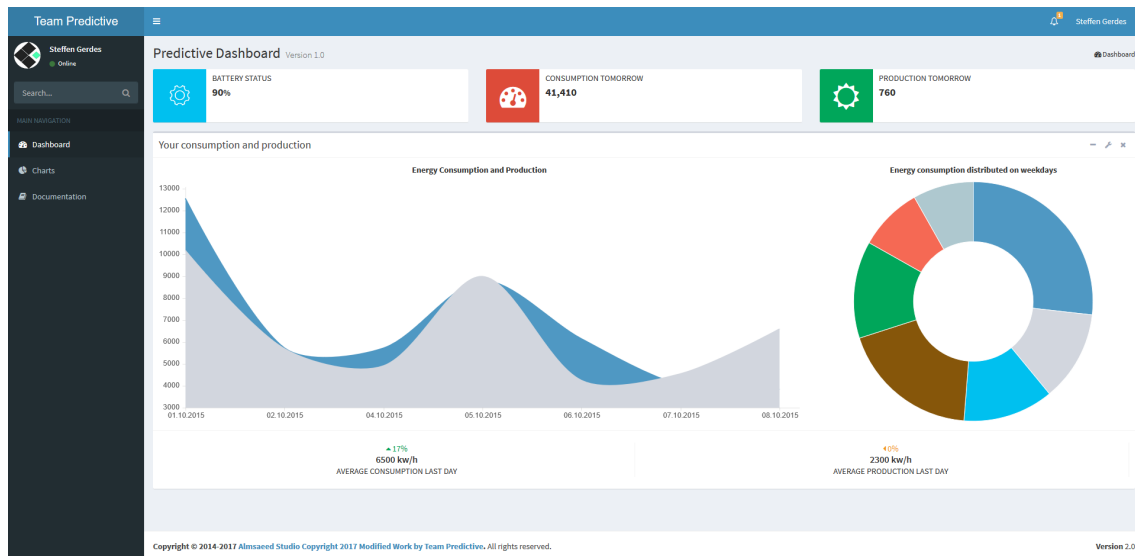


Abbildung 93: Dashboard

6.7. Zusammenfassung und Ausblick

Ziel dieses Teilprojekts war die Erstellung von Vorhersagemodellen für den Energieverbrauch und -erzeugung auf privater Ebene. Als Datengrundlage fungierten aufgezeichnete Daten eines Smart Meters und externe Daten, wie Wetter-, Feiertags- und Feriendaten. Für die Umsetzung dieser Aufgabenstellung wurden Methoden aus dem Bereich Data Mining und Machine Learning bzw. allgemein statistische Modelle genutzt.

Das entwickelte **Verbrauchsmodell** beinhaltet für die Berechnungen der Verbrauchsdaten neben der Datennormalisierung die Bereinigung saisonaler Komponenten. Wie bereits beschrieben, wurden verschiedene Ansätze zur Generierung einer geeigneten Prognosezahl umgesetzt und durch verschiedenen Kennzahlen evaluiert. Ebenfalls wurde bewiesen, dass die Erstellung eines **Produktionsmodell** möglich ist. Das Produktionsmodell wurde erstellt, um Prosumern nicht nur aufzuzeigen, wie viel Energie sie verbrauchen, sondern auch

erzeugen können. Als Schnittstelle zu den Haushalten wurde mit dem Microsoft R Server gearbeitet. An dieser Stelle wurden die Smart Meter-Daten abgelegt, die Prognosen berechnet und deren Ergebnisse an den Haushalt übermittelt.

Neben der Erstellung der Prognosemodelle und der technischen Umsetzung, wurde während der Projektlaufzeit ein Paper angefertigt, welches bei der Multikonferenz Wirtschaftsinformatik 2018 (MKWI), in Form eines Research in Progress Papers, eingereicht wurde. In dem Paper ging es um die Schwierigkeiten bei der Erstellung von Prognosen, bedingt durch saisonale Effekte. Der Grund für die Problematik liegt in der Heterogenität der Datenquellen für die Verbrauchsdaten, bedingt durch die verschiedenen Energieverhaltensweisen. Die Lösung für die Problematik ist die Bereinigung von Saisonalitäten und Trends. Dieses führt dazu, dass die Daten vergleichbar gemacht werden können und sich so bessere Prognosemodelle erstellen lassen. Das vollständige Paper ist im Anhang zu finden.

Zusammenfassend lässt sich sagen, dass das Thema Predictive Analytics allgemein hilf- und aufschlussreiche, aber auch ausbaufähige Ansätze liefert und einen relevanten Bestandteil im Rahmen der Energiewende ausmacht. Ziel der Prognosen ist es ein energieeffizientes Verhalten bei Energie-Prosumern im Haushalten zu erzielen und entsprechende Maßnahmen anzuregen. Das in diesem Teilprojekt entwickelte Verbrauchs- und Produktionsmodell und die darin umgesetzten Algorithmen und Prognosemethoden sind relevante Forschungsansätze. Diese Modelle stellen eine gute Grundlage für weitere potentielle Forschungs- und Entwicklungsarbeiten dar. Auch der Bereich Big Data nimmt an der Stelle eine wichtige Rolle ein. Die Verarbeitung großer Datenmengen erfordert in dem Zuge eine schnelle Verarbeitungslaufzeit. Diese kann mit entsprechender Logik und intelligenter Algorithmik eingebaut werden. Dazu kann für den Ausblick formuliert werden, dass eine permanente Berechnung der Prognosezahlen erfolgen kann, wenn gegenwärtig aktuelle Smart Meter-Daten (auch über mehrere Jahre hinweg) eingelesen werden können. Damit die Berechnung auf Basis dieser aktuellen Daten und die Übermittlung der aktuellen Zahlenwerte geschehen kann, ist die kontinuierliche Ausführung des Skripts notwendig. Dieses Teilprojekt hat sich auf die private Haushaltsebene fokussiert. Denkbar wäre die Ebene zu erweitern und Industrien insbesondere z. B. für regionale Verteilnetzbetreiber Prognosemodelle zu entwickeln und Prognosezahlen zu liefern, damit diese entsprechend den Strom kalkulieren und über den Energiehandel organisieren können. In Hinblick auf den konzeptionellen Energiehandel der Teilgruppe *energy of VALUE*, welcher auf Basis von Prognosen realisiert werden kann, könnten sich weitere Arbeiten mit dem Erstellen solcher Prognosen befassen. In dem vorliegenden Fall wurden die Prognosen auf Tagesbasis erstellt. Für den Handel werden jedoch 96 a 15-Minütigen Prognosen benötigt. Eventuell lassen sich auch weitere Szenarien zur Einbindung von Prognosezahlen identifizieren. An der Stelle könnte die Ebene noch-

mals erweitern und internationale Prognosemodelle entwickeln, die allerdings eine sehr hohe Komplexität ausmachen würden. Das Einspielen externer Daten, wie Wetterdaten, Feiertags- und Feriendaten ist dabei ebenfalls zu beachten.

Da es sich bei den Erstellten Modellen um prototypische Umsetzungen handelt, können auch in diesem Zusammenhang weitere Arbeiten erfolgen:

Ein wichtiger Aspekt stellt dabei die Optimierung der Prognosemodelle dar. In dieser Dokumentation wurden fünf verschiedene Machine Learning-Algorithmen (lineare Regression, Hidden Markow, Naives Bayes, Random Forest und Neuronale Netze) beschrieben und im Verbrauchsmodell bzw. Produktionsmodell umgesetzt. Die Evaluation dieser erfolgt durch vier verschiedene Prognosegütemaße (R^2 , MAPE, MSE und TUK). Jedoch gibt es noch eine Vielzahl weiterer Algorithmen bzw. allgemein Prognose-Methoden und Gütemaße. Weitere Arbeiten könnten sich daher mit der Untersuchung und Implementierung weiterer Prognose-Algorithmen und Gütemaße befassen und einen Vergleich zu den hier dargestellten Machine Learning-Methoden aufstellen. Auch das in diesem Zusammenhang aufgestellte Konzept zur Erstellung eines intelligenten Algorithmus, welcher nach bestimmten Faktoren die beste Prognosezahl auswählt, könnte in weiteren Arbeiten umgesetzt werden.

Bei den hier ausgewählten externen Daten, für die Erstellung des Verbrauchs- und Produktionsmodells, handelt es sich lediglich um historische Daten. Überlegung wäre z. B. einen Wetterdienst einzubinden, um somit laufend mit aktuellen Wetterdaten arbeiten zu können. Besonders im Hinblick auf den Einsatz in der Praxis sind aktuelle Daten von hoher Bedeutung.

Da über die Smart Meter-Geräte die Energieverbrauchsdaten eines Haushalts in dem Verbrauchs- und Produktionsmodell über einen R Server eingespielt werden, handelt es sich im übertragenen Sinne um datenschutzrechtliche Informationen die verarbeitet werden. Zusätzlich können Daten mit Hintergrundinformationen für eine genauere und zuverlässigere Berechnung der Prognosen einbezogen werden. An dieser Stelle kommt das Thema Datenschutz auf, das ebenfalls als wichtiges Thema in weiteren Arbeiten separat betrachtet werden kann.

Über die Verbrauchsprognosen hinaus gibt es die Möglichkeit, dass entscheidungsrelevante Prognosen erstellt werden, das Prescriptive Analytics bezeichnet wird. Im Gegensatz zu Predictive Analytics, welches prognostizierte zukünftige Ereignisse und Abschätzungen der Wahrscheinlichkeit des Eintretens eines solchen Ereignisses darstellt, kann durch Prescriptive Analytics genaue Vorschläge basierend auf Predictive Analytics-Ergebnissen erstellt werden [Wol15]. Es können dabei zusätzlich Erklärungen, warum ein bestimmtes Ereignis eintreffen wird und Empfehlungen, wie man auf ein bestimmtes Ereignis reagieren

sollte, generiert [Wol15].

Mit dieser Methode könnte ein Haushalt dabei unterstützt werden effizienter und kostengünstiger mit der Ressource Energie umzugehen. Das führt dazu, dass beispielsweise vorgegeben wird, dass es sinnvoll ist, wenn die Waschmaschine erst gegen 18 Uhr abends eingeschaltet wird und nicht wie gewöhnlich um 10 Uhr morgens. In Verbindung mit einem Smart Home (Bereich: Internet der Dinge) könnte dieses Vorgehen automatisiert ablaufen und bestimmte Geräte, werden nur zu den prognostizierten bestmöglichen Zeitpunkten eingeschaltet.

In Bezug auf den Energiehandel lässt sich diese Methode auf den Einkauf und Verkauf von Energie übertragen, mit dem sich die Teilgruppe *energy of VALUE* beschäftigt hat. Auf Basis bestimmter Prognosezahlen, die durch Predictive Analytics zustande kommen, können Entscheidungen für den Ein- oder Verkauf von Energie (Prescriptive Analytics) (ggf. automatisch) getroffen werden. Mit Prescriptive Analytics werden also Auswirkungen zukünftiger Entscheidungen abgeschätzt, um Entscheidungen zu bewerten bevor sie getroffen werden. Ein Energie-Prosumer selbst kann in dem Zusammenhang den vorgeschlagenen Handlungsempfehlungen zustimmen oder ablehnen. An der Stelle sei angemerkt, dass das Verbrauchsverhalten, aufgrund der vorliegenden Energie-Prognosezahlen, beeinflusst werden kann. Da sich die Energieverbraucher bzw. -Prosumer durch solche Prognosen im Energieverbrauch beeinflussen lassen, können sich entsprechend Lebensgewohnheiten verändern. Hierbei könnte man eine empirische Untersuchung durchführen, die dieses Verhalten untersucht.

Bezüglich der Applikation könnten weitere Tests in einem größeren Szenario in Verbindung der WebServices stattfinden. Ursprünglich sollten im Dashboard auch die Microsoft R Server WebServices für Verbrauch und Produktion integriert werden. Zu der Zeit war jedoch der neue Webservice, welcher ebenso auf einem neuen Modell basiert, in der Entwicklung. Während der Entwicklung des neuen WebServices wurde auch festgelegt, dass eine gemeinsame Empower Generations-Applikation erstellt wird. Aus diesem Grund wurde die Entwicklung am Predictive Dashboard nicht weiter fortgeführt. Im Rückblick konnten mit dem Dashboard die Ergebnisse der Prognosen angemessen visualisiert werden, jedoch fehlt der Applikation der Mehrwert für den Nutzer. Bis auf das Betrachten der Charts und Kennzahlen bietet das Dashboard keine Interaktion mit dem Nutzer. Dementsprechend können in einer gemeinsamen Applikation aller Teilgruppen mehr Funktionen gebündelt werden um dem Nutzer eine besser User Experience zu gewährleisten. Das gesamte Projekt ist im Git¹⁸ der Projektgruppe vorzufinden.

¹⁸ <https://empower.informatik.uni-oldenburg.de/empower/predictive/tree/master/PredictiveDashboard>

7. Gemeinsame Applikation

Während der dritten und letzten Projektphase fasste die Projektgruppe den Entschluss, die einzelnen Ergebnisse der Teilprojekte in einer gemeinsamen Applikation zu vereinen. Zuvor hatte jede Teilgruppe ihre eigenen Webanwendungen, welche durch die neue Applikation abgelöst werden sollte. Für die Erstellung der gemeinsamen App wurde aus jeder Teilgruppe ein Verantwortlicher bestimmt, wodurch sich das Team von nun an mit der Implementierung einer solchen Anwendung beschäftigte. Das Projekt *Gemeinsame Applikation* selbst wurde ebenfalls in verschiedene Phasen unterteilt: Konzeption, Design und Implementierung.

7.1. Motivation

Zu Beginn des Projekts bestand bereits der Wunsch, am Ende des Projektes alle drei Teilprojekte zusammenzubringen. Dies erwies sich jedoch zu Anfang als schwierig, da die einzelnen Projektgruppen zuerst selbst Erfahrungen, Informationen und Wissen ansammeln mussten, bevor eine Entscheidung über die Art des Zusammenspiels getroffen werden konnte. Da mit Abschluss der zweiten Projektphase drei lauffähige Prototypen bereitstanden, welche jeweils eine eigene Visualisierung mitbrachten, konnte schlussendlich der Schritt in die Richtung der gemeinsamen Applikation gegangen werden.

Der Gedanke hierbei ist es, ein einziges System zu schaffen, mit dem der Nutzer die verschiedenen Funktionen *Lastkurvendisaggregation*, *Energiehandel über die Blockchain* und *die Prognose der Verbrauchswerte* nutzen kann. Außerdem sollen auf der Startseite auf einer Ansicht die wichtigsten Informationen zum eigenen Energiekonsum dargestellt werden.

7.2. Konzept und Design

Die ersten Überlegungen für das Konzept der gemeinsamen Applikation wurden von allen Mitgliedern und Betreuern der Projektgruppe gemeinsam vorgenommen. Hierbei wurde in einem ersten Schritt von jeder Teilgruppe alle Funktionen gesammelt, die in die App übertragen werden könnten. In einem weiteren Schritt wurden die Funktionen priorisiert, sodass sich auf die wichtigsten Aspekte konzentriert werden konnte. Im weiteren Verlauf hat sich ergeben, dass die Startseite der Applikation ein Dashboard darstellen sollte, auf dem die wichtigsten Informationen abgebildet werden. Von diesem Dashboard aus gelangt man auf die Seiten *Today* (energy of WOW), *Trade* (energy of VALUE) und *Prediction* (energy of TOMORROW). Diese Teilbereiche der Applikation bieten einen bis zwei Views, die spezielle Funktionen der Teilgruppen beinhalten.

7.2.1. Dashboard

Bubble-Chart Das Dashboard, welches in der Abbildung 94 dargestellt ist, besteht hauptsächlich aus vier Teilen. Im oberen und größten Abschnitt befindet sich ein *Bubble-Chart*. Dieses besteht aus drei Feldern. In der mittleren Bubble *Consumption* werden die Tageswerte für den bereits verbrauchten Energiekonsum in kWh angezeigt. Ähnlich verhält sich das weitere Feld *Produced*. Hier wird die eigens erzeugte Energie angezeigt, ebenfalls in kWh. Die dritte Bubble *Balance* gibt die Differenz der über die Blockchain gehandelten Energie in Euro an. Die einzelnen Bubbles verhalten sich dynamisch und ändern ihre Größe, sobald ein Wert im Vergleich zu vorherigen Tagen über- bzw. unterdurchschnittlich ist.

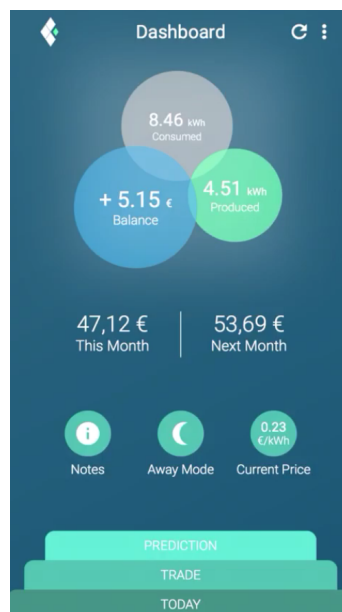


Abbildung 94: Dashboard

Monatliche Kosten In dieser Zeile des Dashboards befinden sich zwei Felder, welche jeweils prognostizierte Kosten für einen Monat anzeigen. Das linke Feld zeigt an, wie hoch der Stromverbrauch für den aktuellen Monat voraussichtlich werden wird. Das rechte Feld zeigt den prognostizierten Wert für den nächsten Monat an. Diese Daten sollen anhand des Prognosemodells berechnet werden.

Informationen und Away Mode In dieser Komposition befinden sich drei runde Buttons mit verschiedenen Funktionen. Der Button *Notes* zeigt an, wenn eine Benachrichtigung für

den Nutzer verfügbar ist. Möglich wären hier Optimierungshinweise, Anomalien im Energieverhalten oder weitere Erinnerungen. Der *Away Mode*-Button lässt eine Art Ruhemodus aktivieren. Dieser kann zum einen für das Prognosemodell hilfreich sein, da so vermittelt wird, dass in nächster Zeit weniger Energie konsumiert wird. Außerdem ist dieser Modus sinnvoll, um zu erkennen, ob sich die Grundlast der Wohnung bzw. des Hauses verändert, während man selbst nicht zu Hause ist, oder Geräte ungenutzt aktiv sind. Der dritte Kreis *Current Price* zeigt den aktuellen Strompreis pro kWh an, der bei einer Transaktion über die Blockchain momentan zu Stande kommen würde.

Karten-Header Am Unteren Bereich des Dashboards befinden sich die Header der Karten, die zu den bereits genannten Teilbereichen der Applikation führen. Diese besitzen unterschiedliche Farben, welche von den Views der Teilgruppen übernommen wird. Auf den Inhalt und die Funktionen der Bereiche der verschiedenen Teilprojekte wird im nächsten Kapitel eingegangen.

7.2.2. Today / Devices

Die Sichten *Today* und *Devices* decken die Funktionen der Teilgruppe *energy of WOW* ab. Hier geht es hauptsächlich um vergangene Verbrauchsdaten, um diese Vergleichen zu können. Dazu zählt der Gesamtverbrauch des Haushalts, sowie die Verbräuche der einzelnen Geräte in diesem. Diese Informationen sollen durch die Lastkurvendisaggregation gewonnen werden.

Today In der View *Today* (Abbildung 95) werden Stromverbrauchsdaten angezeigt, welche in der Vergangenheit liegen. Durch einen *Selector* kann zwischen der Anzeige von Tages-, Monats- und Jahreswerten gewechselt werden. Am oberen Bereich der Karte wird angezeigt, welcher Zeitraum gerade dargestellt wird. Zusätzlich werden hier Informationen über den Verbrauch in kWh und die dazugehörigen Kosten abgebildet. Im Mittelteil der Karte wird ein Graph in Form einer Verbrauchskurve dargestellt, wodurch auf einen Blick nachvollzogen werden kann, zu welcher Zeit mehr bzw. weniger elektrische Energie verbraucht wurde. Am unteren Teil der Karte befindet sich der Übergang zur Sicht *Devices*.

Devices Mit dem *Devices*-View (Abbildung 96) wird eine Übersicht über die größten Verbraucher im Haushalt gegeben, die vom Disaggregationsalgorithmus erkannt worden sind. Hierfür wird eine Liste ausgegeben, in der die einzelnen Verbraucher aufgeführt sind. Wie bekannt, werden auch hier der Stromverbrauch und die dadurch entstandenen Kosten der

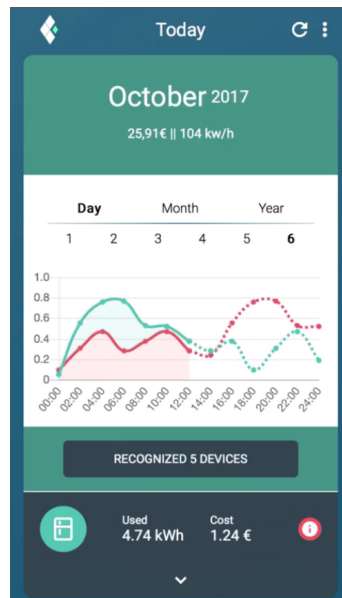


Abbildung 95: Ansicht: Today

einzelnen Geräte dargestellt. Außerdem kennzeichnet ein Informationssymbol an der rechten Seite, welches der aufgeführten Geräte gerade aktiv ist oder ob Benachrichtigungen zu einem Gerät vorliegen. Die Liste ordnet sich nach den größten Verbrauchern im Haushalt.

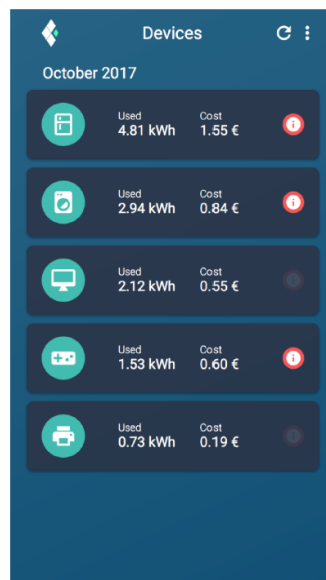


Abbildung 96: Ansicht: Devices

7.2.3. Trade / Transactions

Die beiden Views *Trade* und *Transactions* verhalten sich analog zu den eben beschriebenen Sichten, allerdings für das Teilprojekt, das sich mit dem *Energiehandel über die Blockchain* beschäftigt. Hierbei soll die Applikation sich hauptsächlich mit getätigten Transaktionen befassen.

Trade Die *Trade*-Karte ist nach demselben Prinzip aufgebaut wie die *Today* und *Prediction* Views, siehe Abbildung 97.

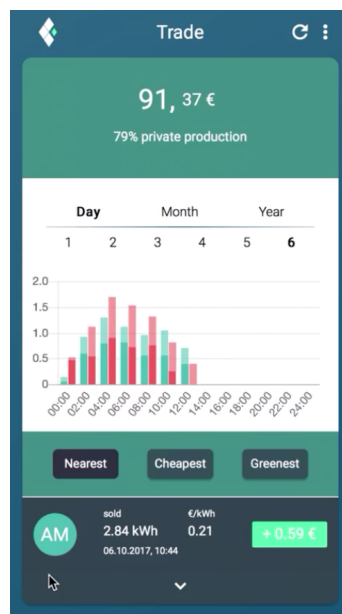


Abbildung 97: Ansicht: Trade

Am oberen Teil der Karte befinden sich Informationen über Einnahmen bzw. Ausgaben über die Blockchain in Form einer Differenz-Anzeige. Diese Anzeige kann ebenfalls durch einen *Selector* für die gewünschte Zeit angezeigt werden. Das Diagramm in der Mitte der Karte zeigt zum einen eine Kurve des Stromverbrauchs und zum anderen in Form eines Balkendiagramms, wie viel des Stromkonsums über Energie, welche über den Blockchain-Handel gekauft wurde, abgedeckt wurde. Am unteren Teil der Karte befindet sich ein Übergang zur zweiten Sicht der Blockchain-Funktionen.

Transactions Die *Transactions* bilden alle zuletzt getätigten Transaktionen ab, die über die Blockchain abgeschlossen wurden. Diese sind chronologisch geordnet, mit der neuesten Transaktionen am oberen Teil der Liste (Abbildung 98).



Abbildung 98: Ansicht: Transactions

Jeder Listeneintrag zeigt die Person mit der gehandelt wurde an, sowie Informationen zur Transaktion selbst. Dazu gehören die gehandelte Anzahl an Strom in kWh, der dazugehörige Preis pro kWh und die dementsprechende Finanzsumme, die sich aus der Transaktion ergibt. Diese kann positiv oder negativ ausfallen, je nachdem ob Strom ver- oder gekauft wurde. Letzteres wird in der Liste deutlich, indem die Transaktionssumme farblich rot bzw. grün hinterlegt ist. Des Weiteren wird der genaue Zeitpunkt der Transaktion dargestellt. Am Kopf der gesamten Sicht wird der durchschnittliche Preis pro Kilowattstunde angegeben.

7.2.4. Prediction

Prediction Das Teilprojekt *energy of TOMORROW*, das sich mit der Vorhersage von Energieverbräuchen beschäftigt, findet in der Applikation ebenfalls einen Platz in der *Prediction*-Sicht, welche in der nachstehenden Abbildung 99 zu sehen ist.

Auch hier kann der Nutzer einen bestimmten Zeitbereich selektieren. Hierdurch wird oben angezeigt, wie viel Energie voraussichtlich für diese Periode benötigt wird und welche Kosten dadurch am Ende auf den Nutzer zukommen könnten. Diese Anzeige wird durch ein Diagramm ergänzt, welches sowohl den Konsum an Energie als auch die voraussichtliche Produktion prognostiziert. Dadurch lässt sich abschätzen, wie viel Strom etwa aus der eigenen Produktion kommen wird und wie viel vom Energieversorger bzw. über die



Abbildung 99: Ansicht: Prediction

Blockchain dazugekauft werden muss, um den Bedarf zu decken. Am unteren Teil der Karte werden die Summen beider Kurven dargestellt, sodass auf einem Blick gesehen werden kann, wie viel voraussichtlich produziert und konsumiert wird.

7.3. Implementierung

Für den Design-Prozess wurde eine Kombination aus der Template-Engine Pug und der Stylesheet-Sprache Sass gewählt. Diese stellen eine Vereinfachung zur Arbeit mit HTML und CSS dar, wodurch die Komplexität von größeren Anwendung sinkt und die Lesbarkeit steigt.

Die Ordnerstruktur sah dabei wie folgt aus: Im src-Ordner befinden sich die für das Design maßgeblich verantwortlichen Dateien. Hierbei sind besonders die beiden Ordner *components* und *compositions* wichtig. Die Compositions beinhalten alle wiederkehrenden Bereiche der Applikation, welche öfter in der App in ähnlicher Form wiederkehren. Dies können beispielsweise die Navigationsleiste, Listeneinträge oder Kartenabschnitte sein. Diese Compositions wiederum enthalten einzelne Components. Diese bilden die einzelnen Elemente ab, die auf den Teilbereichen der Applikationen zu sehen sind, wie z.B. Labels, Kreise oder Buttons.

Beide Ordner beinhalten für jede Composition bzw. component einen Unterordner, welcher jeweils eine .pug- und eine .scss-Datei enthält. Hier werden der Aufbau, das Ausse-

hen und das Zusammensetzen der jeweiligen Bereiche gestaltet.

Damit die Applikation Daten abbilden und dynamisch ablaufen kann, soll das Design in eine State-basierte Umgebung umgesetzt werden. Dabei sollten Funktionen wie das Aktualisieren von Zählerwerten und Graphen in Echtzeit ohne "Reload" realisiert werden können. In die Auswahl hierfür kamen die beiden Frameworks *Angular* und *React*.

Dabei fiel die Entscheidung auf Angular, was mehrere Gründe hat. Zum einen nutzt Angular die Programmiersprache *TypeScript*, welche diverse Vorteile gegenüber JavaScript bietet. Des Weiteren müssen keine weiteren Metasprachen genutzt werden und der Aufbau von Angular-Webseiten nutzt eine Komponenten-Struktur, wodurch das Design der Applikation leichter integriert werden kann. Außerdem bietet Angular mit dem *Ionic Framework* einen kurzen Weg zu einer nativen App für iOS und Android. Ein weiterer Grund für die Entscheidung für Angular war, dass das Team bereits Erfahrungen im Einsatz mit Angular und der Integration von Graphen besaß.

Für die Überführung in Angular wurden die Pug-Components angepasst. Für mehrfach verwendete Komponenten sind parametrisierte pug Dateien erstellt worden, wodurch Code-Redundanzen vermieden wurden und die Möglichkeit zu globalen Änderungen geschaffen werden.

Die Aufteilung der Applikation in Angular erfolgt in Dumb- und Smart-Components. Dabei wird bei den Components differenziert, ob diese lediglich Informationen aus anderen Components erhalten und darstellen, oder ob sie an einen Service gebunden sind und in der Lage sind Daten zu laden sowie fortbestehend Änderungen dieser zu erkennen. Derzeit arbeitet die Applikation lediglich mit einer Simulation von Energiedaten. Eine Anbindung über eine Schnittstelle wie *SocketJS* ist aber möglich. Webservices der einzelnen Teilgruppe auf die zugegriffen werden kann, liegen bereits vor.

8. Fazit

In diesem Kapitel wird der gesamte Projektablauf resümiert und kritisch reflektiert. Dazu wird als erstes eine grobe Zusammenfassung formuliert auf die eine Beurteilung zum organisatorischen und dem allgemeinen Ablauf folgt. Abschließend wird Stellung zur Zielerreichung seitens der Projektgruppe genommen.

Zusammenfassung Die Projektgruppe Empower Generations hat im Rahmen eines einjährigen Projekts neue Möglichkeiten und Potentiale untersucht, die durch die Energiewende und die Digitalisierung entstehen. Durch den Einsatz neuer Technologien und innovativer Ansätze sollte wertschöpfender Nutzen für Endverbraucher generiert werden. Damit ist das Aufbrechen veralteter Energiemarktstrukturen unvermeidlich. Die Ansätze haben gezeigt, dass digitale Messwerte die Ausgangsbasis für neue Ideen und Konzepte bieten. Die Projektmitglieder haben es gewagt, sich von bestehenden Ansätzen zu distanzieren, vorauszuschauen und stets mit Blick auf den Konsumenten neue Geschäftsfelder zu untersuchen.

Organisatorisches und allgemeiner Ablauf Die Projektgruppe wurde am Anfang Oktober 2016 in drei Teilprojekte unterteilt. Es fand über die ersten beiden Projektphasen relativ wenig Interaktion unter den einzelnen Gruppen statt. Das wurde als weniger positiv empfunden. Nach anfänglichen Startschwierigkeiten konnten sich alle Projektgruppen durch die Grobkonzepte, die jede Teilgruppe anfertigte, wieder in ihre Spur begeben. Durch die Grobkonzepte haben wir uns Ziele formuliert und Anforderungen spezifiziert, die wir im Laufe des Projekts zu erfüllen versucht haben. Die Grobkonzepte wurden später Teil dieser Dokumentation und konnten erfolgreich eingearbeitet werden. Die Implementierung des ersten Prototyps der drei Teilgruppen kollidierte indes stark mit der Klausurenphase des ersten Semesters, was sich als sehr fordernd herausstellte, aber dennoch gut überwunden wurde. Nach einer erfolgreichen Präsentation und durchweg gutem Feedback seitens aller Betreuer und Partner ging es darum, die Prototypen weiterzuentwickeln. Besser war hierbei, dass man sich durch Phase 1 auf dieses Niveau eingestimmt hat und so gelernt hat, besser zu arbeiten. Nachdem die Weiterentwicklung der Prototypen vorgestellt wurde, startete im Juni die letzte Projektphase. In dieser Phase fand eine Zusammenführung der drei Teilprojekte statt. Dafür erschien die Visualisierung über eine App als die sinnvollste Variante. Zudem bekam jede Teilprojektgruppe den Auftrag ein Paper zu ihrem Thema anzufertigen, sodass schließlich neben den Prototypen und wichtigen Erkenntnissen auch drei Paper zu den Endergebnissen dieser Projektgruppe zählen. Beendet wurde das Pro-

jekt mit einer Abschlusspräsentation, der Abgabe der Prototypen und der vorliegenden Dokumentation.

Generell ist positiv anzumerken, dass die Rollenverteilung zu jedem Zeitpunkt klar war und jeder wusste, was zu tun ist. Speziell das Projektmanagement und der Wechsel zwischen dem Projektmanagement-Team nach jeder Phase lief sehr gut ab und ist hervorzuheben. Die Betreuer haben dabei auch ihre Rolle voll ausgefüllt und konnten der Gruppe zu jedem Zeitpunkt helfen. Auch dass keine Seminararbeiten geschrieben wurden, kann im Nachhinein als die richtige Entscheidung festgehalten werden, und sollte auch für die nachfolgenden Projektgruppen so beibehalten werden. Ein anfängliches Grobkonzept über den zu entwickelnden Prototypen, am besten zu Beginn der Projektzeit, und ein Paper, welches in Phase drei angefertigt wird ist eine optimale Lösung. Das stärkte das Vertrauen in die Studenten und hat zugleich die Gruppendynamik gestärkt.

Zielerreichung Die Teilgruppen der Projektgruppe haben sich thematisch mit verschiedenen Einsatzbereichen im Kontext der Energiewirtschaft beschäftigt. Hierbei wurden in den drei Projektphasen prototypisch Ergebnisse erzielt. Die erste Projektphase wurde genutzt, um Know-How in den jeweiligen Themengebieten zu generieren. Die zweite Projektphase fokussierte sich auf die Umsetzung eines konkreten Anwendungsfalles. Abschließend sollten in der finalen Projektphase die Ergebnisse der Teilgruppen in einer gemeinsamen Applikation münden. Die gemeinsame Applikation zeigt, welche Synergieeffekte zwischen den Teilgruppen entstanden sind. Die Teilgruppe energy of VALUE benutzt im erstellten Energiehandelskonzept Prognosen, welche durch Predictive Analytics Methoden der Teilgruppe energy of TOMORROW bereitgestellt werden. Weiterhin bietet das Ergebnis der Teilgruppe energy of WOW einen Ausblick auf die Möglichkeiten von Machine Learning im Energiesektor. Mit der Disaggregation von Lastkurven wurde ein Use Case umgesetzt, der ein Beispiel für neue Geschäftsmodelle und Technologien ist. Die Voraussetzung für diese Möglichkeiten sind die Digitalisierung und die beschriebenen Entwicklungen des Energiesektors.

Unter der Idee “Interface der Energiewende” wurden die drei Themenbereiche abschließend in einer App zusammengeführt. Mit dieser App wird die Vision der konsum-zentrischen Energiewende fokussiert, in welcher der Kunde selbst als Marktakteur teilnehmen kann. Der Kunde kann detaillierte Informationen über sein Verbrauchsverhalten entnehmen und bekommt damit das Werkzeug, um sein Verhalten zu optimieren. Durch den Einsatz von Predictive Analytics erhält der Kunde Prognosen für seine Energieerzeugung und seinen Energieverbrauch. Es wird so eine Transparenz erzeugt, die in Verbindung mit Blockchain Technologie, einen dezentralen Energiehandel ermöglicht.

Literatur

- [Bad17] Kripi Badonia. Proof of work v/s proof of stake : Mining 101, 07 2017.
- [Bid16] Bidgely, Inc. THE HEART OF THE PLATFORM. <https://www.bidgely.com/technology/>, 2016. Zugegriffen am 20.11.2016.
- [BMI16] BMI. Der Schutz des Rechts auf informationelle Selbstbestimmung. http://www.bmi.bund.de/DE/Themen/Gesellschaft-Verfassung/Datenschutz/Informationelle-Selbstbestimmung/informationelle-selbstbestimmung_node.html, 2016. Zugegriffen am 25.11.2016.
- [Bri16] Miles Brignal. Smart meters: an energy-saving revolution or just plain dumb?, 2016.
- [Buna] Bundesamt für Sicherheit in der Informationstechnik. Smart Metering – Datenschutz und Datensicherheit auf höchstem Niveau. Technical report, Bundesministerium für Wirtschaft und Energie (BMWi).
- [Bunb] Bundesministerium für Wirtschaft und Energie (BMWi). Digitalisierung der Energiewende. <http://www.bmwi.de/DE/Themen/Energie/Netze-und-Netzausbau/digitalisierung-der-energiewende,did=726780.html>. Zugegriffen am 29.11.2016.
- [Bunc] Bundesministerium für Wirtschaft und Energie (BMWi). Stromnetze. <http://www.bmwi.de/DE/Themen/Energie/Energieforschung-und-Innovationen/foerderschwerpunkte,did=455512.html>. Zugegriffen am 21.11.2016.
- [Bun15] Bundesamt für Sicherheit in der Informationstechnik. Das Smart-Meter-Gateway. Technical Report BSI-Bro15/332, Bundesministerium für Wirtschaft und Energie (BMWi), Oktober 2015.
- [Bun16a] Bundesministerium für Wirtschaft und Energie (BMWi). Erneuerbare-Energien-Wärmegesetz. http://www.erneuerbare-energien.de/EE/Navigation/DE/Recht-Politik/Das_EEWaermeG/das_eewaermeg.html, 2016. Zugegriffen am 25.11.2016.

- [Bun16b] Bundesregierung. Entwurf eines Gesetzes zur Digitalisierung der Energiewende. <http://dip21.bundestag.de/dip21/btd/18/075/1807555.pdf>, feb 2016. Zugegriffen am 29.11.2016.
- [Bun16c] Bundesregierung für Wirtschaft und Energie. Gesetz zur Digitalisierung der Energiewende, September 2016. Zugegriffen am 01.10.2017.
- [Bun17a] Bundesregierung für Wirtschaft und Energie. Die nächste Phase der Energiewende: Das EEG 2017. <https://www.bmwi.de/Redaktion/DE/Artikel/Energie/eeg-2017-start-in-die-naechste-phase-der-energiewende.html>, 2017. Zugegriffen am 01.10.2017.
- [Bun17b] Bundesregierung für Wirtschaft und Energie. Die nächste Phase der Energiewende kann beginnen. <https://www.bundesregierung.de/Content/DE/StatischeSeiten/Breg/Energiekonzept/0-Buehne/ma%C3%9Fnahmen-im-ueberblick.html;jsessionid=189B046ECE4B76B423A60DE6C46FE952.s6t2>, 2017. Zugegriffen am 01.10.2017.
- [Bun17c] Bundesregierung für Wirtschaft und Energie. Warum wurde das EEG eigentlich eingeführt? <https://www.bmwi.de/Redaktion/DE/FAQ/EEG-2017/warum-eeg.html>, 2017. Zugegriffen am 01.10.2017.
- [Bun17d] Bundesregierung für Wirtschaft und Energie. Wasserkraft. <http://www.erneuerbare-energien.de/EE/Navigation/DE/Technologien/Wasserkraft/wasserkraft.html;jsessionid=673C582EFB3E295DFE5D34F4B9570583>, 2017. Zugegriffen am 01.10.2017.
- [CER] CER - Commission for energy Regulation. About CER. <http://www.cer.ie/about-cer>. Zugegriffen am 28.09.2017.
- [Cha13] Charles Gadalla. SAP Predictive Analysis - Transform Your Future with Predictive Insight. <https://www.slideshare.net/timoelliott/predictive-analysis-showcase-apj>, 2013. Zugegriffen am 30.09.2017.
- [Chr15] Christoph Meinel, Norbert Koppenhagen. Smart Data in der Energiewirtschaft. http://deutschland-intelligent-vernetzt.org/app/uploads/sites/4/2015/12/151116_FG2_018_PG_Smart_Data_Thesenpapier_SmartData_Thesenpapier_Schwerpunkt_Energie_und_SmartData.pdf, 2015. Zugegriffen am 25.11.2016.

- [CMO] CMO by Adobe. ADI: Is Europe In The Middle Of A Smartphone Divide? <http://www.cmo.com/adobe-digital-insights/articles/2017/2/20/adi-is-europe-in-the-middle-of-a-smartphone-divide.html>. Zugriffen am 02.10.2017.
- [Col00] Colin Shearer. The CRISP-DM model: the new blueprint for data mining. *Journal of Data Warehousing*, 5, 2000.
- [Cru15] Crunchbase Inc. Bidgely. <https://www.crunchbase.com/funding-round/7af88fb85753c3f5936eb6535c472d68>, November 2015. Zugriffen am 21.11.2016.
- [Dan15] Daniel Thomas Roy. *Intelligente Energiesysteme der Zukunft: Die Entwicklung von Digitaler Stromzählering und Smart Grid im Jahre 2025*. Deplomica Verlag GmbH, Hamburg, 2015.
- [Das12] Das Europäische Parlament und der Rat der Europäischen Union. Richtlinie 2012/27/EU des Europäischen Parlaments und des Rates vom 25. Oktober 2012. <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2012:315:0001:0056:DE:PDF>, oct 2012. Zugriffen am 29.11.2016.
- [Dav] David Meyer. Holt-Winters Filtering. <https://stat.ethz.ch/R-manual/R-devel/library/stats/html/HoltWinters.html>. Zugriffen am 29.09.2017.
- [Die12] Dieter Attig. Kraft-Wärme-Kopplung als Ergänzung der erneuerbaren Energien. *Energiewirtschaftliche Tagesfragen*, pages 8 – 9, 2012.
- [Die14] Diego Wyllie. Projektmanagement für Profis. *t3n digital pioneers Magazin*, (37):118–121, 2014.
- [Dri] Drifty Co. Ionic Framework. <https://ionicframework.com/>. Zugriffen am 02.10.2017.
- [DS17] Gunther Dütsch and Neon Steinecke. Use cases for blockchain technology in energy & commodity trading. *PricewaterhouseCoopers GmbH*, 2017.
- [edu15] edureka! Naive Bayes Classifier Tutorial - Naive Bayes Classifier in R - Naive Bayes Classifier Example. <https://www.youtube.com/watch?v=-SeyrC4yZF4>, März 2015. Zugriffen am 28.09.2017.

- [Ern13] Ernst & Young. Kosten-Nutzen-Analyse für einen flächendeckenden Einsatz intelligenter Zähler. <http://www.bmwi.de/BMWi/Redaktion/PDF/Publikationen/Studien/kosten-nutzen-analyse-fuer-flaechendeckenden-einsatz-intelligenterzaehler,property=pdf,bereich=bmwi2012,sprache=de,rwb=true.pdf>, 2013. Zugegriffen am 29.11.2016.
- [EWE12] EWE Vertrieb GmbH. Musterrechnung. <https://www.ewe.de/~ /media/ewe/internet/images/grafiken-unbestimmtes-format/privatkunden/energierechnung/energierech1.jpg?la=de-de&h=881&w=624>, oct 2012.
- [Fab15] Fabian Knorre. Random Forests - Angewandte Biostatistik mit R. https://www.statistik.tu-dortmund.de/fileadmin/user_upload/Lehrstuehle/Genetik/BS1213/Biostatistik_mit_R_Knorre.pdf, 2015. Zugegriffen am 28.09.2017.
- [Fra11] Fraunhofer USA. What Is NIALM? IEEE Transactions on Consumer Electronics, 2011.
- [Fre03] Freie Universität Berlin, Center für Digitale Systeme. Zeitreihenanalyse: Das Holt-Winters-Verfahren. http://www.fernuni-hagen.de/ksw/neuestatistik/content/files/modul_24703.pdf, 2003. Zugegriffen am 29.09.2017.
- [Gar16] Gartner. Gartner's 2016 hype cycle for emerging technologies identifies three key trends that organizations must track to gain competitive advantage. Press Release, 08 2016.
- [Gek14] GekkoQuant. Hidden Markov Models - Model Description Part 1 of 4. <http://gekkoquant.com/2014/05/18/hidden-markov-models-model-description-part-1-of-4/>, Mai 2014. Zugegriffen am 25.03.2017.
- [Geo07] Geoffrey Tso, Kelvin Yau. Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks. *Energy*, 32:1761–1768, 2007.
- [Gre] GreenPocket GmbH. GreenPocket Website. <http://www.greenpocket.de/>. Zugegriffen am 22.11.2016.

- [Har17] Harry Wirth. Aktuelle Fakten zur Photovoltaik in Deutschland. <https://www.ise.fraunhofer.de/content/dam/ise/de/documents/publications/studies/aktuelle-fakten-zur-photovoltaik-in-deutschland.pdf>, August 2017. Zugegriffen am 29.09.2017.
- [HD17] Louise Hagström and Olivia Dahlquist. Scaling blockchain for the energy sector. ., 2017.
- [Hen] Henri Theil. Who Forecasts Best? In *Henri Theil's Contributions to Economics and Econometrics*, volume 3 of *Economic Policy and Forecasts, and Management Science*, pages 1115–1120. Springer Science+Business Media, B.V., Dordrecht. Zugegriffen am 29.09.2017.
- [Ing10] Ingmar Visser, Maarten Speekenbrink. depmixS4: An R Package for Hidden Markov Models. *Journal of Statistical Software*, 36(7), 2010. Zugegriffen am 25.03.2017.
- [Ing16] Ingmar Visser, Maarten Speekenbrink. Package depmixS4. <https://cran.r-project.org/web/packages/depmixS4/depmixS4.pdf>, Februar 2016. Zugegriffen am 25.03.2017.
- [inn] innogy SE. innogy Innovation Hub. <https://www.innogy.com/web/cms/de/3181722/ueber-innogy/innogy-innovation-technik/innovation-hub/>. Zugegriffen am 22.11.2016.
- [Ins] Institut für ökologische Wirtschaftsforschung. Haushalte werden relevante Akteure auf dem Energiemarkt. <http://www.prosumer-haushalte.de/das-projekt/ausgangslage.html>. Zugegriffen am 22.07.2017.
- [ISS] ISSDA - Irish Social Data Archive. Commission for Energy Regulation (CER). <http://www.ucd.ie/issda/data/commissionforenergyregulationcer/>. Zugegriffen am 28.09.2017.
- [Jac15] Jack Kelly, Nipun Batra, Oliver Parson, Graham Wood. NILM METADATA. https://github.com/nilmtnk/nilm_metadata, September 2015. Zugegriffen am 20.11.2016.
- [Jac16] Jack Kelly, Nipun Batra, Oliver Parson, et al. NILMTK: Non-Intrusive Load Monitoring Toolkit. <https://github.com/nilmtnk/nilmtnk/blob/master/README.md>, August 2016. Zugegriffen am 20.11.2016.

- [Jos16] Josef Schira. *Statistische Methoden der VWL und BWL: Theorie und Praxis*. Pearson, München, 2016.
- [Jü16] Jürgen Frisch. In-Memory-Datenbank soll SAP-Systeme optimieren. *is report - Informationsplattform für Business Solutions*, pages 20–23, 2016.
- [K. 12] K. CARRIE ARMEL, ABHAY GUPTA , GIREESH SHRIMALI, ADRIAN ALBERT. IS DISAGGREGATION THE HOLY GRAIL OF ENERGY EFFICIENCY? THE CASE OF ELECTRICITY. Technical Report PTP-2012-05-1, Precourt Energy Efficiency Center, 2012.
- [Kar16] Karim Said Barsim, Lukas Mauch, and Bin Yang. Neural Network Ensembles to Real-time Identification of Plug-level Appliance Measurements. Technical report, Institute of Signal Processing and System Theory, University of Stuttgart, 2016.
- [Kat08] Katharina Morik. Der CRISP-DM Prozess für Data Mining. http://www-ai.cs.uni-dortmund.de/LEHRE/VORLESUNGEN/KDD/SS08/02_CRISP_4p.pdf, 2008. Zugegriffen am 14.11.2016.
- [Kat13] Katherine Portilla. Will smart meters really help homeowners save energy? <https://www.theguardian.com/environment/blog/2013/jun/28/smart-meters-homeowners-save-energy>, 2013. Zugegriffen am 01.10.2017.
- [Kla10] Klaus Heuck, Klaus-Dieter Dettmann, Detlef Schulz. *Elektrische Energieversorgung*. Vieweg +Teubner Verlag, Springer Fachmedien, Wiesbaden, eighth edition, 2010.
- [Kla15] Klaus Backhaus, Bernd Erichson, Rolf Weiber. *Fortgeschrittene Multivariate Analysemethoden*. Springer Gabler, Berlin Heidelberg, 2015.
- [Kla16] Klaus Müller. Energiewende heißt: Umstecken im Kopf. <http://www.beegy.com/2016/03/energiewende-heisst-umstecken-im-kopf-2/>, 2016. Zugegriffen am 25.11.2016.
- [Kon12] Konferenz der Datenschutzbeauftragten des Bundes und der Länder und Düsseldorf Kreis . Orientierungshilfe datenschutzgerechtes Smart Metering. https://www.datenschutz-bayern.de/technik/orient/oh_smartmeter.pdf, 2012. Zugegriffen am 25.11.2016.

- [Kon15] Konrad Mertens. *Photovoltaik - Lehrbuch zu Grundlagen Technologie und Praxis*. Hansen, München, 2015.
- [Leg17] Power Leger. Power ledger white paper. 2017.
- [Lei17] Leibniz Universität Hannover. Formelsammlung: Methoden der empirischen Wirtschaftsforschung. <https://www.wiwi.uni-hannover.de/fileadmin/emp/Hauptstudium/ewifo/fs-2.pdf>, 2017. Zugegriffen am 29.09.2017.
- [Luk16] Lukas Mauch, Karim Said Barsim and Bin Yang. How well can HMM model load signals. Technical report, Institute of Signal Processing and System Theory, University of Stuttgart, 2016.
- [Mar] Marine Institute. Weather Buoy Network Real Time Data. <http://data.marine.ie/Dataset/Details/20972>. Zugegriffen am 28.09.2017.
- [Mar06] Markus Bühner. *Einführung in die Test- und Fragebogenkonstruktion*. Pearson, München, 2006.
- [Mat17] MathWorks. Kreuzvalidierung. <https://de.mathworks.com/discovery/kreuzvalidierung.html>, 2017. Zugegriffen am 28.09.2017.
- [Mic17a] Microsoft. Configuration Initial. <https://msdn.microsoft.com/en-us/microsoft-%20r/operationalize/configuration-initial>, 2017. Zugegriffen am 20.08.2017.
- [Mic17b] Microsoft. Operationalize. <https://msdn.microsoft.com/en-us/microsoft-r/operationalize/about>, 2017. Zugegriffen am 20.08.2017.
- [Mic17c] Microsoft. What is Microsoft R Server. <https://docs.microsoft.com/en-us/r-server/what-is-microsoft-r-server>, 2017. Zugegriffen am 20.08.2017.
- [Nak08] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Www.Bitcoin.Org*, page 9, 2008.
- [Nip14] Nipun Batra, Jack Kelly, Oliver Parson. NILMTK: An Open Source Toolkit for Non-intrusive Load Monitoring. Technical report, Indraprastha Institute of Information Technology Delhi, India, 2014.

- [Oli16] Oliver Bendel. Stichwort: Smart Metering. <http://wirtschaftslexikon.gabler.de/Archiv/1097117097/smart-metering-v4.html>, 2016. Zugegriffen am 25.11.2016.
- [Pau08] Paul E. Black. memorization. <https://xlinux.nist.gov/dads/HTML/memoize.html>, August 2008. Zugegriffen am 21.07.2017.
- [Pet00] Peter Andres, Markus Spiwox. Prognosegütemaße - State of the Art der statistischen Ex-post-Beurteilung von Prognosen, 2000.
- [Pet12] Peter Salje. *EEG 2012 - Gesetz für den Vorrang Erneuerbarer Energien*. Carl Heymanns Verlag, Köln, 2012.
- [Pro] Prof. Dr. Oliver Bendel. Smart Metering. <http://wirtschaftslexikon.gabler.de/Archiv/1097117097/smart-metering-v4.html>. Zugegriffen am 29.11.2016.
- [PWC16] PWC. Blockchain - chance für energieverbraucher? ., 2016.
- [Rai12] Rainer Schlittgen. *Angewandte Zeitreihenanalyse mit R (Lehr- und Handbücher der Statistik)*. Oldenbourg, München, 2012.
- [Reg09] Regina Tüchler. Beispiel in R: Verfahren zur Modellierung von ZR mit Saison und Trend. <http://statmath.wu.ac.at/courses/multverf1/SaisonTrendinR.pdf>, November 2009. Zugegriffen am 29.09.2017.
- [Ren] Rene Troeger. Institut für Signalverarbeitung und Systemtheorie, Universität Stuttgart. <http://www.uni-stuttgart.de/cgi-bin/mail.cgi?rene.troeger=iss.uni-stuttgart.de>, oct. Zugegriffen am 24.11.16.
- [Rob15] Rob Hyndman, Shu Fan. Monash Electricity Forecasting Model. <http://robjhyndman.com/papers/MEFMR1.pdf>, 2015. Zugegriffen am 25.11.2016.
- [Rom12] Roman Streubel and Bin Yang. Identification of Electrical Appliances via Analysis of Power Consumption. Technical report, Institute for Signal Processing and System Theory, University of Stuttgart, 2012.
- [RWE16] RWE AG. RWE startet Zukunftsgeschäft mit neuer Tochtergesellschaft. <http://www.rwe.com/web/cms/de/37110/rwe/presse-news/pressemitteilungen/pressemitteilungen/?pmid=4014733>, März 2016. Zugegriffen am 22.11.2016.

- [SAP17a] SAP. R and SAP HANA. <https://help.sap.com/viewer/a78d7f701c3341339fafe4031b64f015/2.0.02/en-US/d867885b48b0480a81f0e61818eafcf1.html>, 2017. Zugegriffen am 28.09.2017.
- [SAP17b] SAP. Was ist SAP HANA? <https://www.sap.com/germany/products/hana.html>, 2017. Zugegriffen am 28.09.2017.
- [Seb10] Sebastian Jeworutzki. Visualisierung statistischer Daten mit R. <http://www.stat.ruhr-uni-bochum.de/teaching/vis/visu.pdf>, 2010. Zugegriffen am 23.09.2017.
- [Sim08] Simone Thiele. *Elektrizitätserzeugung durch Windenergie - Von Onshore- zu Offshore-Standorten*. Diplomica, Hamburg, 2008.
- [Sma16] Smappee n.v. Smappee Website. <http://www.smappee.com/de/energiemonitor>, 2016. Zugegriffen am 22.11.2016.
- [Ste16a] Stefan Krempel. Bundestag beschließt Zwangsbeglückung mit intelligenten Stromzählern. <http://www.heise.de/newsticker/meldung/Bundestag-beschliesst-Zwangsbeglueckung-mit-intelligenten-protect\discretionary{\char\hyphenchar\font}{\}\Stromzaehlern-3248056.html>, 2016. Zugegriffen am 25.11.2016.
- [Ste16b] Stephen Makonin. 3rd International Workshop on Non-Intrusive Load Monitoring. <http://nilmworkshop.org/2016/>, 2016. Zugegriffen am 23.11.2016.
- [Str17] Strom-Report. Der deutsche Strommix: Stromerzeugung in Deutschland. <https://1-stromvergleich.com/strom-report/strommix/>, Oktober 2017. Zugegriffen am 01.10.2017.
- [swo16] swot-analyse.bizi. Multikriterielle Entscheidungsanalyse. <http://www.swot-analyse.biz/multikriterielle-entscheidungsanalyse>, 2016. Zugegriffen am 23.09.2017.
- [Sö09] Sönke Albers, Daniel Klapper, Udo Konradt, Achim Walter, Joachim Wolf. *Methodik der empirischen Forschung*. Gabler, Wiesbaden, 2009.
- [Umw16] Umweltbundesamt. Bioenergie. <https://www.umweltbundesamt.de/themen/klima-energie/erneuerbare-energien/bioenergie#textpart-1>, Februar 2016. Zugegriffen am 29.09.2017.

- [Wol07] Wolfgang Konen. Naive Bayes - Ein einfacher Klassifikator. <http://www.gm.fh-koeln.de/~konen/WPF-DM-Cup/04-Naive-Bayes.pdf>, 2007. Zugegriffen am 28.09.2017.
- [Wol12] Wolf Riepl. CRISP-DM: Ein Standard-Prozess-Modell für Data Mining. <http://statistik-dresden.de/archives/1128>, 2012. Zugegriffen am 25.11.2016.
- [Wol15] Wolfgang Martin. Predictive Analytics – Von Big Data zu Smart Customer Data. *is report - Informationsplattform für Business Solutions - Business Guide & ERP 2015*, 20:8–11, 2015.
- [Wol16] Wolfgang Schellong. *Analyse und Optimierung von Energieverbundsystemen*. Springer Vieweg, Heidelberg, 2016.
- [YHKC⁺16] Jesse Yli-Huumo, Deokyoon Ko, Sujin Choi, Sooyong Park, and Kari Smolander. Where is current research on blockchain technology?—a systematic review. *Plos One*, 11:e0163477, 2016.
- [Zen16] Zentrum für Sonnenenergie- und Wasserstoff-Forschung. Erneuerbare Energien in Zahlen. Technical report, Bundesministerium für Wirtschaft und Energie (BMWi), September 2016.
- [Zic11] Zico Kolter. The Reference Energy Disaggregation Data Set. <http://redd.csail.mit.edu/>, 2011. Zugegriffen am 19.11.2016.

A. Anhang

A.1. Paper Energy of VALUE

Energiemarkt mit Blockchain-Technologie: Ein Marktmodell unter Berücksichtigung bestehender Netzkomponenten und Marktakteure

First Author¹, Second Author¹, Third Author^{1,2}, and Fourth Author²

¹ Institution, Department, City, Country

{first.author, second.author, third.author}@example.com

² Other Institution, Other Department, Other City, Other Country

{third.author, fourth.author}@anotherexample.com

Abstract. Die Energiebranche befindet sich in einem signifikanten Veränderungsprozess. Angesichts des starken Wachstums dezentraler Energieressourcen experimentieren Regierungen, Versorger und weitere Stakeholder mit neuen Lösungen um Effizienz des Elektrizitätsnetzes zu optimieren und einen fortschreitenden Kapazitätsausbau erneuerbarer Energien zu ermöglichen. Nach Jahren, in denen die Energiewende vorrangig mit dem Aufbau erneuerbarer Energie assoziiert wurde, rücken zunehmend systemische Fragen in den Vordergrund, insbesondere zur Integration neuer Marktakteure. Die bestehenden Geschäftsmodelle unterliegen einem grundlegenden Wandlungsprozess, jedoch lassen aktuelle Forschungsarbeiten bestehende Netzkomponenten und Marktakteure unbetrachtet. Die breite Einsatzfähigkeit dieser Ansätze bleibt daher fraglich. Dieser Beitrag präsentiert eine Machbarkeitsstudie eines dezentralen Energiemarktes unter Berücksichtigung des State of the Art des Energieversorgungsnetzes.

Keywords: Blockchain, Energiemarkt, Netzkomponenten, Ethereum

1 Einleitung

Große Veränderungen auf dem Energiemarkt, politische Vorgaben und Ziele im Hinblick auf die Energiewende sowie neue, teilweise bereits ausgereifte Energietechnik führen zu einem Umschwung in der Energiewirtschaft. Die Akteure am Energiemarkt stehen vor der Herausforderung, sich auf die wandelnden dezentralen Erzeugerstrukturen einzustellen und Auswirkungen auf den Energiemarkt der Zukunft abzuschätzen [1]. Die Dezentralisierung des Energiemarktes von wenigen großen Stromerzeugern zu vielen kleinen- und mittelgroßen Erzeugern erfordert eine datenbasierte Umstrukturierung des Energiemanagements. Dies bietet eine Chance durch neue Informations- und Kommunikationstechnologien (IKT) effizientere Steuermechanismen zu etablieren und neue Geschäftsmodelle zu erschließen. Die Einführung intelligenter Stromzähler (Smart Meter) wird als Schlüsselfaktor für die Digitalisierung in der Energiewirtschaft angesehen und bietet durch die Lieferung von Echtzeitdaten eine Grundlage für datenbasierte Technologien und Mehrwertdienste. Der heutige Energiehandel erfolgt über

diverse Wertschöpfungsstufen, bei denen der Endkunde einen Energieversorger beauftragt, der wiederum verschiedene Dienstleister (z. B. Makler, Broker) beauftragt, Energie an der Strombörse (z. B. EEX) zu erwerben. Durch den Einsatz der Blockchain-Technologie als Energiehandelsplattform wird ein direkter Energiehandel zwischen den Energiekonsumenten und -produzenten ohne zusätzliche Intermediäre ermöglicht. Eine Disintermediation ist die direkte Folge.

Politisch fokussieren auf internationaler Ebene das Kyoto-Protokoll sowie das Pariser Klimaabkommen die Reduktion von schädlichen Emission und wirken somit als Treiber für die Energiewende [2], [3]. Auf nationaler Ebene legt das Erneuerbare-Energien-Gesetz (EEG) einen richtungsweisenden Rahmen, der zu einer Umstrukturierung des Energiemarktes führt [4]. Neben politischen Regularien sowie dem gesellschaftlichen Drang nach „sauberem Strom“ tritt auch der ökonomische Aspekt stärker in den Vordergrund. Durch effizientere und kostengünstigere Technologien regenerativer Energieanlagen werden finanzielle Anreize geschaffen, in erneuerbare Energien zu investieren oder eigenständig als Marktakteur teilzunehmen.

Das heutige, auf fossil-nuklearer Erzeugung basierende Energiesystem ist ein Auslaufmodell und die Entwicklung des Energiemarktes im Bereich erneuerbarer Energie schreitet rasant voran [5]. In den vergangenen 15 Jahren hat sich der Anteil an erneuerbaren Energien bereits verfünffacht und liegt in Deutschland bei über 30% des gesamten Bruttostromverbrauchs [6].

Die Dezentralisierung des Energiemarktes von wenigen großen Stromerzeugern zu vielen kleinen- und mittelgroßen Erzeugern verlangt nach einer Umstrukturierung des Energiemanagements. Dies bietet eine Chance durch neue Informations- und Kommunikationstechnologien (IKT) effizientere Steuermechanismen und neue Geschäftsmodelle zu etablieren [7]. Die Einführung intelligenter Stromzähler (Smart Meter) wird als entscheidendes Kriterium für die Digitalisierung in der Energiewirtschaft angesehen und bietet durch die Lieferung von Echtzeitdaten eine Grundlage für zahlreiche darauf aufbauende Technologien [8-12].

In dem vorliegenden Paper wird erläutert, wie ein Blockchain-Ansatz im dezentralen Energiemarkt eingesetzt werden kann. Der Fokus liegt dabei auf realistischen Marktszenarien, indem aktuelle Marktakteure wie Energieversorger und Netzbetreiber berücksichtigt werden. Nachdem in Kapitel 2 die Ausgangssituation vorgestellt wurde, wird in Kapitel 3 das entwickelte Marktmodell des prognosebasierten Energiehandels erläutert. Dabei werden sowohl Aspekte des Energiemarktes als auch des Energienetzes berücksichtigt. In Kapitel 4 wird die prototypische Umsetzung beschrieben. Schließlich wird die vorliegende Arbeit mit einem Fazit sowie einem Ausblick geschlossen.

2 Ausgangssituation

Der in der Einleitung dargestellte Rahmen wirft relevante Fragestellungen auf und bietet den Forschungskontext zur Umsetzung verschiedener Szenarien auf Grundlage einer Blockchain-Anwendung. Die Verwendung der Blockchain-Technologie stößt im Finanzsektor bereits auf großes Interesse und auch im Energiesektor ergeben sich Potentiale und Einsatzmöglichkeiten [13].

Durch die fortschreitende Entwicklung dezentraler Erzeugerstrukturen und den Einsatz digitaler Messtechnik, ergeben sich für den Energiemarkt Chancen hinsichtlich der

Entwicklung neuer Geschäftsmodelle. Eine Anpassung an die sich ändernden Umweltbedingungen ist netz- und marktseitig zwingend erforderlich, da elektrische Energie aus regenerativen Energiequellen nicht nach Bedarf gesteuert werden kann. Die zuvor nach Bedarf regelbaren konventionellen bspw. Kohle- oder Gaskraftwerke bieten die Möglichkeit, planbar Energie zu erzeugen und je nach Netzauslastung steuernd eingreifen zu können.

Erneuerbare Energien werden in Deutschland durch die EEG-Umlage weiter gefördert, sodass erneuerbare Energieanlagen weiter ausgebaut werden und die Herausforderungen in der Planungsunsicherheit sowie die Einhaltung der Netzstabilität in Zukunft weiter zunehmen werden [4]. Die Planungsunsicherheit ist auf die erhöhte Anzahl dezentraler Erzeuger zurückzuführen. Die unbekannte Menge an Strom, die in das Netz eingespeist wird, erschwert die Erhaltung der Netzstabilität. Dem Endverbraucher wird kaum die Möglichkeit gegeben, auf die Volatilität erneuerbarer Energien zu reagieren. Durch über Monate und Jahre statisch festgesetzte Preise ist es nicht möglich die Netzauslastung an dynamisch schwankende Energieerzeugung anzupassen. Die sich ändernden Angebots- und Nachfragemengen an elektrischer Energie werden im Energiemarkt demnach unzureichend an die Endverbraucher herangetragen. Dieses Informationsdefizit gilt es zu beseitigen.

Das heutige Energiesystem ist auf zentrale Erzeugungsanlagen ausgerichtet und wird, abgesehen von Großverbrauchern (>100.000 kWh/a) mithilfe von Standardlastprofilen gesteuert. Standardlastprofile sind standardisierte Prognosen über den zukünftigen Lastgang eines Energieverbrauchers [18]. Die prognostizierte Energie wird am Terminmarkt bereits Wochen oder Jahre im Voraus gehandelt. Dazu werden feste Verträge mit Energielieferanten bzw. Aggregatoren vereinbart, die den Energiehandel übernehmen. An Energiebörsen kann weiterhin anhand von Verbrauchsprognosen auf dem Spotmarkt ein Tag sowie auf einem Intraday-Markt bis zu 45 Minuten vor der Energiebereitstellung gehandelt werden [19]. Erneuerbare Energien können auf dem Spotmarkt am vorherigen Tag prognostiziert und entsprechend gehandelt werden.

Der Wandel zu einem dezentralen Energiemarkt bietet Potential für eine Umsetzung durch Blockchain-Technologie, da diese Technologie dezentrale Strukturen und transaktionsbasierte Erlösmodelle abbilden kann. Zahlreiche Start-Ups befassen sich daher mit der Entwicklung von Geschäftsmodellen im Energiesektor in Verbindung mit Blockchain-Technologien [13]. Das *Brooklyn Microgrid* ist ein Beispiel für die isolierte Umsetzung einer Peer-to-Peer Handelsplattform mit dezentral erzeugtem Strom. Das *PowerLedger-Projekt* verfolgt den Vertrieb und die Dokumentation von erneuerbaren Energien in einem offenen Stromnetz [14]. Die Analyse bestehender Projekte in Forschung und Industrie ergab jedoch die Notwendigkeit der Berücksichtigung sowohl der Netzkomponenten, als auch aktueller Marktakteure [15].

Dezentrale Erzeugerstrukturen führen im Rahmen der Energiewende zu neuen Anforderungen und erfordern Anpassungen des markt- und netzseitigen Energiesystems. Fehlende Informationen über exakt prognostizierte Lasten und Energieerzeugung erschweren die Einhaltung der Netzstabilität. Der Blockchain wird Lösungspotential hinsichtlich der neuen Anforderungen im Energiemarkt prophezeit. Die vorliegende Arbeit untersucht folglich, ob dezentraler Energiehandel mithilfe einer Blockchain-Implementierung realisiert werden kann. Dabei steht die Frage im Mittelpunkt, ob die markt- und

netzseitigen Komponenten des Energiemarktes innerhalb einer Blockchain-Anwendung berücksichtigt werden können. Zur Untersuchung der Frage wurde zunächst ein theoretisches Marktmodell entwickelt. Anschließend wurde die technische Machbarkeit auf IKT-Ebene anhand eines Prototyps evaluiert.

3 Prognosebasierter, direkter Energiehandel

Dezentrale Erzeugungsstrukturen führen im Rahmen der Energiewende zu einer Umstrukturierung der Wertschöpfungskette. Die klassischen Marktakteure, insbesondere der Energieversorger, weichen den dezentralen Kleinerzeugern. Gleichzeitig können prognosebasierte Energiehandelsszenarien Indikatoren für die zukünftige Entwicklung des Energiemarktes darstellen. In diesem Abschnitt werden zunächst die Akteure vorgestellt sowie die Relevanz von individuellen Lastprognosen beschrieben. Anschließend wird ein hierauf basierendes Marktmodell vorgestellt. Bisherige blockchainbasierte Energiehandelsansätze betrachten den Energieversorger und Netzbetreiber (sowie deren Wertschöpfung) als Marktakteure bisher nicht.

3.1 Akteure

Die heutige Energiehandelssystematik erfolgt über diverse Intermediäre, bei denen der Endkunde einen Energielieferanten beauftragt, der wiederum verschiedene Dienstleister (z. B. Makler, Broker) beauftragt, Energie an der Strombörse zu erwerben. Durch den Einsatz einer Blockchain als dezentrale Energiehandelsplattform wird direkter Energiehandel zwischen Energiekonsumenten und -produzenten ermöglicht, da die Bilanzierung und Abrechnung der Transaktionen automatisiert werden können und durch dezentrale Sicherheitsmechanismen (insbesondere Proof of Work) auf einen klassischen vertriebsseitigen Energieversorger, der die jährliche Abrechnung für Energiekunden durchführt, verzichtet werden kann [13]. Im Folgenden werden die Marktakteure vorgestellt, die im Marktmodell berücksichtigt werden.

Energiekonsument: Der Marktakteur *Energiekonsument* entnimmt elektrische Energie aus dem Stromnetz. Durch entsprechende Verbrauchsquellen stellt der Energiekonsument eine Netzlast dar. Beispiele für diesen Marktakteur sind private Haushalte, Unternehmen, Energiespeicher oder E-Tankstellen.

Energieproduzent: Der Marktakteur *Energieproduzent* speist Strom in das Netz ein. Mittels Stromerzeugungsanlagen wie Windkraft-, Photovoltaikanlagen oder Biomassekraftwerken produziert der Energieproduzent elektrische Energie, um diese an Energiekonsumenten verkaufen zu können.

Prosumer: Der Marktakteur *Prosumer* ist sowohl Energiekonsument als auch Energieproduzent, der bspw. durch Solaranlagen auf Firmengebäuden oder Privathaushalten elektrische Energie in das Stromnetz einspeist.

Netzbetreiber: Unter dem Marktakteur *Netzbetreiber* werden die bestehenden Akteure Übertragungsnetzbetreiber und Verteilnetzbetreiber zusammengefasst. Der Netzbetreiber stellt mit dem Stromnetz die Infrastruktur bereit und ist zuständig für Frequenzstabilität, Spannungsstabilität und Versorgungszuverlässigkeit.

Versicherer: Der Marktakteur *Versicherer* tritt nach Bedarf als Energiekonsument oder Energieproduzent auf, um ein Gleichgewicht zwischen Einspeisungen und Entnahmen zu gewährleisten. Dieser muss flexible Reserven (Regelleistung, Sekundärleistung, Minutenreserve, etc.) elektrischer Energie vorhalten und auch als Netzlast (Energiekonsument) agieren können. Der Versicherer tritt nur im unperfekten Markt (Angebot und Nachfrage gleichen sich nicht exakt aus) auf, um Abweichungen der Prognosen oder unausgeglichenem Handel entgegenzuwirken.

3.2 Notwendigkeit von Energieverbrauchs- und Erzeugungsprognosen

Der heutige Energiemarkt basiert auf der Nutzung von aggregierten Standardlastprofilen für Privathaushalte. Sämtliche Verbraucher, die weniger als 100.000 kWh/a Strom verbrauchen, werden mittels dieser aggregierten Profile zusammengefasst [18]. Aus dieser Aggregation resultiert eine Planungsunsicherheit bezüglich der Netzauslastung und des Börsenhandels für Energieunternehmen. Höhere Planungssicherheit lässt sich nur durch eine feinere Granularität der Datengrundlage über den Konsum bzw. die Produktion der Energie erzielen. Infolge der fortschreitenden Verbreitung digitaler Stromzähler können detaillierte Verbrauchs- und Produktionsdaten einzelner Haushalte in Echtzeit erhoben werden. Mithilfe von Machine Learning-Ansätzen (v.a. Regressionsanalysen) lassen sich anhand dieser Datengrundlage individuelle Lastprofile für einzelne Haushalte und Unternehmen erstellen [16].

Das nachfolgend erläuterte Marktmodell setzt das Vorliegen von individuellen Lastprofilen voraus. Dies ist notwendig, da Strom basierend auf Verbrauchs- und Erzeugungsprognosen einzelner Marktakteure gehandelt wird. Neben höherer Planungssicherheit des Netzbetreibers können individuelle Lastprofile gehandelt werden. Das heißt, dass zu Konsum-Lastprofilen Produktionsprofile erworben werden können. Innerhalb dezentraler Erzeugerstrukturen können so einzelne Konsumenten und Erzeuger zusammengebracht werden und direkt untereinander handeln. Exakte Prognosen setzen hochwertige Datenbestände voraus. Aktuelle Forschungen zeigen, dass hinreichend granulare Datenbestände mit digitalen Stromzählern gewonnen werden können, digitale Zähler teilweise jedoch noch fehlerhafte Werte liefern [17]. Aus feingranularen, qualitativ hochwertigen Prognosen folgen für die Anwender niedrige Pönalzahlungen und für den Netzbetreiber eine effizientere Netzplanung, da die erforderliche Ausgleichsenergie minimiert wird. Ausgleichsenergie oder Regelenergie wird dann produziert, wenn weniger Strom verbraucht, als prognostiziert wurde [18]. Im Energiesektor wird ein Lastprofil eines Tages in 96 Gültigkeitszeiträumen à 15 Minuten unterteilt. Diese Zeitreihe bildet die Grundlage des blockchainbasierten Ansatzes. Die einzelnen 15-Minuten-Intervalle werden nachfolgend als *Pakete* bezeichnet und enthalten neben einer fortlaufenden Nummer für den jeweiligen Zeitpunkt (0-95) auch die Menge an Strom, die in diesem Zeitraum verbraucht, produziert, aber auch prognostiziert wurde, sowie die scharfe Zuordnung des jeweiligen Marktakteures. Beispielsweise entspricht das Paket mit der Nummer 0 dem Zeitraum 00:00-00:15 Uhr. Innerhalb des Marktmodells sind drei Gruppierungen der Prognosen relevant: der jeweilige Energieverbrauch, die Energieproduktion und die zugrundeliegende Netzauslastung.

Durch die Berücksichtigung der Netzauslastung in diesem Energiehandelsmodell werden Steuerungsmechanismen erzielt, die zur Vermeidung von möglichen Engpässen beitragen. Der Netzbetreiber veröffentlicht hierbei täglich die Auslastung der jeweiligen Trasse und kann unterschiedliche Preise für die Netznutzung bestimmen. Bei hoher Netzauslastung folgt ein dementsprechend höheres Netzentgelt. Ebenso können bspw. negative Netzkosten bei niedriger Netzauslastung veranlasst werden. Der Netzbetreiber „entlohnt“ den Konsumenten de facto für das Einschalten von elektronischen Geräten. Durch den Handelsmechanismus kann der Stromverkehr passiv kontrolliert werden, da neben dem Strompreis auch der Netzpreis Berücksichtigung findet.

3.3 Marktmodell des dezentralen, direkten Energiehandels

Neben der erläuterten Notwendigkeit von Prognosen sind sogenannte Pönalen von besonderer Relevanz. Pönalen sind Vertragsstrafen und wirken sich in diesem Modell durch Gebühren für falsch prognostizierte Verbrauchs- und Produktionsmengen aus. Der Energiehandel erfolgt *ex ante* und wird täglich abgerechnet. So wird eine feingranulare Strompreiskontrolle und hohe Transparenz erzielt. Der Ablauf des Handelsprozesses wird in Abbildung 1 dargestellt. Dabei werden die Aktionen der einzelnen Akteure auf einer Zeitachse abgebildet. Dieser Abschnitt befasst sich zunächst ausschließlich mit der marktseitigen Konzeption. Die technische Umsetzung wird in Kapitel 4 erläutert.

Wie in Abbildung 1 dargestellt, unterteilt sich der prognosebasierte Energiehandel in drei logisch voneinander abgegrenzte Phasen. Die Phase *Prognose veröffentlichen* repräsentiert das Handelsfenster, welches vor dem Handelstag stattfindet. Hier werden Netzbetreiber, Konsumenten und Produzenten aktiv und veröffentlichen ihre Prognosen über Verbrauch und Produktion für die 96 Viertelstundenintervalle (insgesamt 24 Stunden). Der Zeitpunkt t bis $t+1$ beinhaltet die Erfassung der *Realdaten*. Hierbei wird über den digitalen Stromzähler die Menge an produziertem und konsumiertem Strom über eine Schnittstelle ausgelesen und in der Blockchain dokumentiert. Abschließend findet die Phase der *Abrechnung* statt. Diese Phase beginnt nach der Erfassung des 96. Strompaketes (23:45 – 00:00). Abschließend wird die Bilanzierung der Realwerte und der prognostizierten Werte durchgeführt.

Die Prognosen der jeweiligen Akteure werden in sogenannte Smart Contracts geschrieben. Dies sind kleine Programme, die eigenständig auf Programmierlogik basierende Transaktionen auf Blockchains durchführen können. Der Netzbetreiber veröffentlicht eine Prognose über die Netzauslastung des Handelstages. Das Netz wird hierbei in verschiedene Regionen (Minigrids) unterteilt. Die Prognose beinhaltet die Angabe des jeweiligen Minigrids und eine Angabe zur Netzauslastung des jeweiligen Zeitpunktes. Der Produzent veröffentlicht ebenfalls die prognostizierte Energieproduktion des Handelstages. Der Produzent definiert zudem die Höhe des Kaufpreises der produzierten Energie. Das Netz wird hierbei in verschiedene Regionen (Minigrids) unterteilt. Die Prognose beinhaltet die Angabe des jeweiligen Minigrids und eine Angabe zur Netzauslastung des jeweiligen Zeitpunktes. Der Produzent veröffentlicht ebenfalls die prognostizierte Energieproduktion des Handelstages. Der Konsument definiert seine Handelsstrategie und veröffentlicht seinen Strombedarf für den kommenden Handelstag in der Blockchain. Der implementierte Handelsmechanismus stellt sicher, dass der Bedarf gedeckt wird. Dabei werden Präferenzen des Konsumenten (Einschränkungen

wie z.B. Preis oder Herkunft) berücksichtigt. Der Einkaufspreis setzt sich aus dem Verkaufspreis der Produzenten und der konsumentspezifischen Netzkosten zusammen. Der Handel kann also zwischen einem Konsumenten und mehreren Produzenten stattfinden (1:n-Beziehung). So wird sichergestellt, dass der Bedarf auch durch verschiedene Produzenten parallel gedeckt werden kann. Um finanzielle Sicherheit zu gewährleisten, bezahlt der Konsument den benötigten Strom unmittelbar mit der zugehörigen Währung des verwendeten Blockchain-Ökosystems, im vorliegenden Beitrag wurde *Ethereum* eingesetzt, da Ethereum zum Zeitpunkt der Entwicklung des Prototyps, das am weitesten entwickelte Blockchain-Ökosystem mit Unterstützung intelligenter Verträge (Smart Contracts) war.

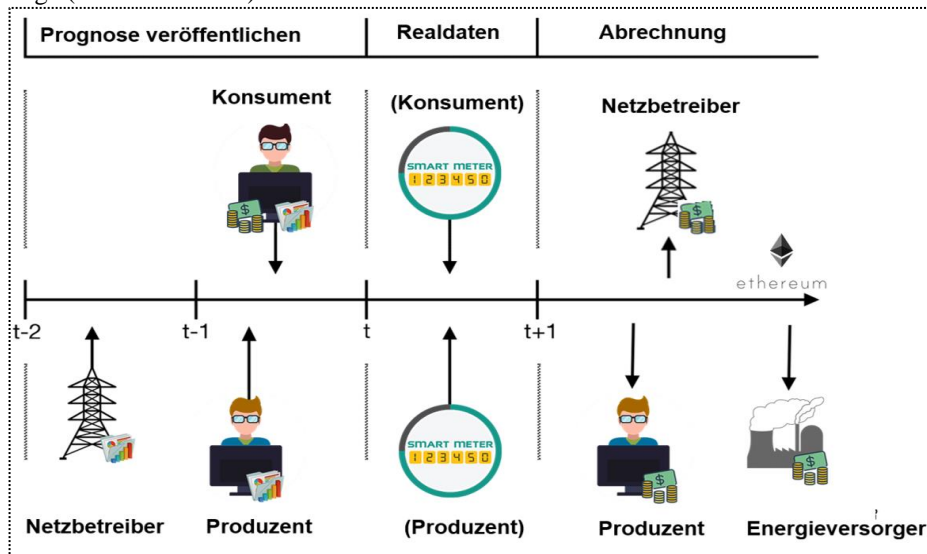


Abbildung 1: Prognosebasierter Energiehandel

Während der Phase *Realdaten* werden durch den digitalen Stromzähler die Daten über Produktion bzw. Verbrauch erhoben. Durch eine optische Schnittstelle werden diese ausgelesen und der Blockchain-Anwendung zur Verfügung gestellt. Nach dem letzten Zeitfenster (oder der Auslieferung des letzten Pakets) werden diese in der Blockchain dokumentiert, sodass die Abrechnung mithilfe eines Soll-Ist-Vergleichs (Bilanzierung) realisiert wird. Die Abrechnung wird vom Smart Contract nach Erfassung der Realdaten implementiert. Ergeben sich Differenzen, so kommt es zu Pönalzahlungen. Die Höhe der jeweiligen Zahlungen ist ein Marktmechanismus und kann an dieser Stelle noch nicht abgeschätzt werden. Ebenso wird durch den Smart Contract der Netzbetreiber gemäß seiner Leistung entlohnt. Die Besonderheit des Marktmodells stellt die Position des Energieversorgers als *Versicherer* dar. Eine exakte Übereinstimmung der Prognose und der Realdaten ist unmöglich zu erzielen, daher müssen Differenzen ausgeglichen werden. Weiterhin ist eine Umstellung zu einem Peer-to-Peer Handelsmarkt ein langfristiger Prozess und dementsprechend weiterhin auf den Energieversorger angewiesen. Eine kurzfristige Umstellung erscheint aufgrund der aktuellen Marktdurchdringung der Energieversorger als sehr unrealistisch. Die Steuerungseffekte, die durch dieses Marktdesign erzielt werden, beziehen sich auf die Entlastung des Netzbetreibers

durch die Minimierung von Netzengpässen. Ebenso werden durch die Angabe der Regionen der Akteure lange Transportstrecken vermieden, da diese mit hohen Netzkosten bepreist werden. Außerdem kann das Verbrauchsverhalten der Konsumenten beeinflusst werden, da eine Einhaltung der Prognose von wirtschaftlichem Interesse ist, weil Pönalzahlungen vermieden werden können.

4 Prototypische Implementierung

Das dargestellte Marktmodell in der Blockchain erfordert die irreversible Sicherung bzw. Speicherung der dem Energiehandel zugrundeliegenden Daten (Prognosen, tatsächliche Verbrauchs/Erzeugungsdaten, Preise). Für die prototypische Umsetzung wurde eine Ethereum-Blockchain verwendet. Diese basiert auf der zweiten Generation der Blockchain-Technologie, die neben reinen Währungstransaktionen auch das Bereitstellen von dezentralen Applikationen (Smart Contracts) ermöglicht [13].

Hardware: Um Energiehandel mit mehreren Marktakteuren über die Blockchain realitätsnah zu simulieren wurde als Test- und Entwicklungsumgebung auf fünf Raspberry Pi Einplatinencomputern (Pi) der dritten Modellreihe zurückgegriffen. Diese verfügen über ausreichend Rechen- und Speicherkapazität sowie notwendiger Schnittstellen für die prototypische Anwendung. Durch die geringen Kosten und gute Verfügbarkeit der Hardware ist zudem eine einfache Reproduktion der Ergebnisse sichergestellt. Die Pis wurden in einem abgeschlossenen, gemeinsamen Netzwerk als eigenständige Topologie konfiguriert, um die Kommunikation zwischen den einzelnen Geräten zu ermöglichen.

Software: Als Grundlage für die Softwarearchitektur kommen ausschließlich kostenfreie, frei verfügbare Komponenten zum Einsatz. So wird als Betriebssystem auf den einzelnen Pis raspbian verwendet. Zusätzlich wurde ganglia zur zentralen Überwachung wichtiger Systemparameter wie CPU-, RAM- und Netzwerkauslastung eingesetzt. Im Rahmen der Anwendungsentwicklung unter den fünf Pis wurde eine private Blockchain mit Hilfe des Parity-Ethereum-Clients aufgesetzt. Durch die Verwendung einer privaten Blockchain wurde die Grundlage für die Entwicklungsumgebung geschaffen, die im Vergleich zu einer öffentlichen Blockchain keine realen Kosten für Transaktionen verursacht. Ein Pi samt Parity-Client stellt hierbei eine sog. Node dar, sichert stetig eine Kopie der gesamten privaten Blockchain und ermöglicht die Interaktion mit dieser. Die prototypische Anwendung selbst (siehe Abbildung 2) wurde unter NodeJS realisiert.

Die prototypische Applikation lässt sich in zwei Teile unterteilen: Einer clientseitigen NodeJS-Anwendung, die auf den Pis installiert ist und einem Smart Contract, der in der Blockchain hinterlegt ist und somit gemeinsam von allen Teilnehmern genutzt wird. Dabei ermöglicht die clientseitig ausgeführte Applikation unterschiedliche Konfigurationen um die Handlungen eines Käufers, Verkäufers oder Netzbetreibers simulieren zu können. Das Architekturmodell ist in Abbildung 2 dargestellt.

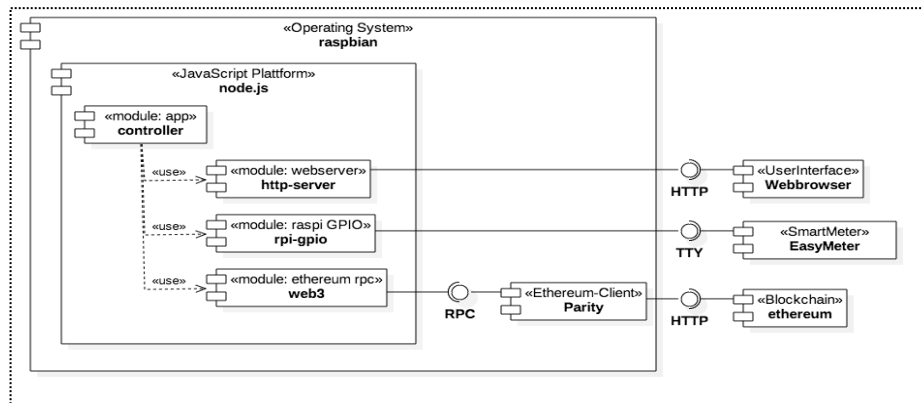


Abbildung 2: Architekturmodelle / Komponentendiagramm der Anwendung

Der Controller dient als Steuereinheit der einzelnen Module. Seine Aufgabe ist es, Daten automatisiert zu verarbeiten und so – je nach Anwendungsszenario – bspw. Prognosen in die Blockchain zu schreiben, automatisch eine Liste von präferierten Verkäufern zu erstellen um anschließend die notwendigen Transaktionen zu tätigen. Je nach Konfiguration kann er als Käufer, Verkäufer, Prosumer oder Netzbetreiber agieren. Der Energiehandel und die Interaktion mit der Blockchain wird also autonom, gemäß parametrisierter Constraints, durchgeführt.

Die Bereitstellung eines Webservice über das Standardmodul http-server ermöglicht dem Anwender mit dem Webbrowser auf die Applikation zuzugreifen. Über eine grafische Oberfläche werden alle relevanten Informationen wie Verbrauchs-, Produktions- und Handelsdaten angezeigt. Ferner lässt sich die Applikation über das Webinterface parametrisieren, um etwa Einschränkungen oder Prognosezahlen zum Energiehandel zu pflegen und so selbst aktiv in den Energiehandel einzugreifen. Über das Modul rpi-gpio wird auf die serielle Schnittstelle des Pi zugegriffen und so aktuelle Verbrauchs- und Erzeugungswerte vom Smart Meter in digitalisierter Form zur Verfügung gestellt. Diese werden in Folge automatisiert in die Blockchain geschrieben bzw. auf dem Web-Interface angezeigt.

Die Verbindung zur Blockchain wird über das Modul web3 realisiert – es kommuniziert über eine standardisierte JSON-RPC-Schnittstelle mit dem Parity-Client. Hierdurch ist es möglich, auch andere Ethereum-Clients als Parity anzubinden. Durch den modularen Aufbau der gesamten Applikation könnte an dieser Stelle auch ein Modul zum Zugriff auf andere/zukünftige Blockchain-Implementierungen eingesetzt werden.

Der in der Blockchain gespeicherte Smart Contract muss in der von Ethereum eigens entwickelten Sprache Solidity implementiert werden. Er umfasst alle für die prototypische Anwendung notwendigen Funktionen um die Daten, die von allen Teilnehmern des Marktes benötigt werden, zu verwalten. Der Smart Contract dient zusätzlich auch als Treuhänder – sämtliche Zahlungen werden zunächst hier hinterlegt und erst im Zuge der Abrechnung an die entsprechenden Parteien freigegeben. Das untenstehende Listing zeigt die Methode zum Abspeichern eines prognostizierten Angebotes. Die dargestellte Funktion entspricht dem Erstellen eines Inserates auf der Blockchain. So wer-

den dem Smart Contract Daten zur eindeutigen Identifizierung des jeweiligen Verkäufers (*_sellerSmartMeterId*), der prognostizierten Menge (*_predictedAmount*) und des Preises pro Watt/h (*_price*) übermittelt. Zusätzlich wird der 15 minütige Zeitraum, in dem die angegebene Menge produziert werden soll über die Kombination *_day/_packageName* definiert und im dreidimensionalen Array *SellerPackages[]* hinterlegt. Entsprechende Funktionen sind auch für das Hinterlegen der Netzkosten, das Handeln und das Abrufen der genannten Daten implementiert. Ferner wurden Fallback-Mechanismen für den Fall der Abweichung von Prognose und Ist-Wert implementiert.

```
// Is executed 96 times for one day, provides prediction
function setSellerPredicted(bytes32 _sellerSmartMeterId, uint
_day, uint _packageName, uint _predictedAmount, uint
_price) {
    SellerPackages[_sellerSmartMeterId][_day].SellerPackageDay[_p
ackageNumber].predictedAmount = _predictedAmount;
    SellerPackages[_sellerSmartMeterId][_day].SellerPackageDay[_p
ackageNumber].price = _price;
    SellerPackages[_sellerSmartMeterId][_day].SellerPackageDay[_p
ackageNumber].availableAmount = _predictedAmount;
    SetSellerPredicted(_sellerSmartMeterId, _day,
_packageNumber, _predictedAmount, _price, _gridZone); }
event SetSellerPredicted(
    bytes32 indexed _smartMeterId,
    uint indexed _day,
    uint indexed _packageName,
    uint _predictedAmount,
    uint _price,
    uint _gridZone);
```

Eine bedeutende Einschränkung bei der Implementierung mittels Solidity ist, dass ausschließlich primitive Datentypen beim Aufruf von Funktionen im Smart Contract verwendet werden können. So muss ein Anbieter zum Abspeichern der Prognosen für einen Tag im dargestellten Fall 96 rechen-, zeit- und kostenintensive Transaktionen durchführen (um 96 Integer-Werte abzulegen). Gleiches gilt für die Käufer-Seite: Es ist nicht möglich, eine Liste oder ein Array aller Anbieter zu einem definierten Zeitraum (Tag/Paketnummer) abzufragen. Auf Clientseite muss hierzu erst über alle Transaktionen der Blockchain iteriert werden. In Solidity gibt es zwar sog. *Events*, die aufgerufen werden können, sobald eine bestimmte Funktion in einem Smart Contract ausgeführt wird. Auch lässt sich mit einer gefilterten Abfrage aller Events zur Funktion *_setSellerPredicted* die Ergebnismenge einschränken.

5 Fazit und Ausblick

Im Rahmen des vorliegenden Papers wurde ausgehend von der Frage, wie ein dezentrales Marktmodell sowie eine prototypische Machbarkeitsstudie auf Basis einer Blockchain-Anwendung entwickelt werden kann, vorgestellt. Dezentraler Energiehandel

kann mithilfe einer Blockchain-Implementierung realisiert werden. Ebenso konnte die marktseitige Komponente durch den Energiehandel zwischen Energieproduzenten und Energiekonsumenten abgebildet werden und die netzseitige Komponente durch Einbindung der Prognosen der Netzauslastung berücksichtigt werden.

Das Marktmodell basiert auf kurzfristigen Verbrauchs- und Konsumprognosen und ist bewusst an den bestehenden Energiemarkt angelehnt, da beide Energiesysteme mittelfristig parallel existieren müssen. Daher baut der dezentrale Energiehandel auf bestehenden Strukturen auf und einer Koexistenz mit dem derzeit praktizierten State of the Art steht grundsätzlich nichts im Wege. Aus diesem Modell erschließt sich jedoch der Wandel des klassischen Intermediäres vom Energieversorger zum Versicherer. Konsumenten, Produzenten und Prosumer nehmen aktiv am Energiehandel teil, ohne den klassischen Energieversorger damit zu beauftragen. Angebot und Nachfrage werden stark liberalisiert und deutlich transparenter, als im aktuellen Energiesystem.

Dezentraler Energiehandel über die Blockchain führt zu verschiedenen Vorteilen, so entstehen Steuerungseffekte durch die Berücksichtigung der Netzkosten während des Energiehandels. Hierbei kann das Konsumverhalten durch Preismechanismen beeinflusst werden. Das vorgestellte Marktdesign kann dazu beitragen, dass sich Konsumenten verstärkt netzdienlich verhalten. Die beschriebene Systemarchitektur ermöglicht zudem einen hohen Automatisierungsgrad, da der Energiehandel durch die individuell konfigurierbare Applikation abgewickelt wird, ohne dass tiefgreifende Eingriffe des Anwenders erforderlich werden. Weiterhin hat die Verwendung eines Blockchain-Ansatzes den positiven Effekt, dass Transaktionen prinzipbedingt manipulationssicher dokumentiert werden und gleichzeitig jederzeit transparent und nachprüfbar bleiben.

Durch die anschließende Konzeption und Anwendungsentwicklung wurden wertvolle Erkenntnisse gewonnen, die zum vorgestellten Energiehandelssystem führten. Die grundsätzliche Umsetzbarkeit auf IKT-Ebene wurde dabei gezeigt. Jedoch wurde festgestellt, dass ausschließlich auf Blockchain basierende Ansätze aus Performanzgründen in dieser Form zum jetzigen Zeitpunkt nicht in der Breite eingesetzt werden können. Dies liegt insbesondere an der Ineffizienz der Verifikationsprozesse (Proof of Work) und der Sicherheitsmechanismen der Blockchain-Datenstruktur, bei der Daten, seien dies nun Prognosen oder monetäre Transaktionen, lediglich immer nur an das Ende angehängt werden können (append only). Auch zeigte sich, dass die Möglichkeiten zur Datenverarbeitung stark limitiert bleiben. Somit ist es sehr unwahrscheinlich, dass Anwendungen, die ausschließlich auf Blockchain zur Datenhaltung und -verarbeitung basieren, den dezentralen Energiemarkt beflügeln werden. Vielmehr wird eine weitere Verarbeitungsschicht implementiert werden müssen, die die individuellen Prognosen miteinander verknüpft, also die Rolle eines Brokers einnimmt. Als Alternative kommen natürlich auch Weiterentwicklungen im Bereich der Blockchain-Technologie infrage, bei der die Datenverarbeitungsgeschwindigkeit und die Verifikationsmechanismen deutlich effizienter gestaltet werden müssen.

Um das tatsächliche Potential abschließend beurteilen zu können, muss die Blockchain-Technologie einen ausreichenden Reifegrad erlangen. Des Weiteren müssen wichtige Kriterien wie die Skalierbarkeit sowie weiterführende Sicherheitsaspekte hinsichtlich der Software als auch der Hardware erforscht werden. Hierbei sind u.a. Maßnahmen festzulegen, wie bei einem Hardwareausfall oder bei Verlust des Blockchain-Nutzerkontos (privater Schlüssel, öffentlicher Schlüssel) verfahren werden soll. Darüber hinaus sind viele (datenschutz-)rechtliche Fragen ungelöst, die insbesondere durch

die ab 2018 geltende EU-Datenschutzgrundverordnung [20] nochmals verschärft werden.

References

1. Hecker, W.: Wandel antizipieren–Zukunft gestalten. In: Zukunftsorientierte Unternehmenssteuerung in der Energiewirtschaft. Springer Fachmedien Wiesbaden. 3-39 (2015)
2. Protocol, Kyoto. United Nations framework convention on climate change. Kyoto Protocol, Kyoto, 19. Jg. (1997)
3. Dröge, S.: Das Pariser Abkommen 2015: Weichenstellung für das Klimaregime. (2015)
4. Erneuerbare-Energien-Gesetz (EEG) vom 21. Juli 2014 (BGBl. I S.1066), Berlin (2014)
5. Wirth, H., Schneider, K.: Aktuelle Fakten zur Photovoltaik in Deutschland. Fraunhofer ISE (2012)
6. Energiebilanzen EV, Arbeitsgemeinschaft. Auswertungstabellen zur Energiebilanz Deutschland. Berlin (2015)
7. Aichele, C. und Dolseski, O. D. (Hg.). Smart Market: vom Smart Grid zum intelligenten Energiemarkt. Springer-Verlag (2014)
8. Krickel, F.: Digitalisierung in der Energiewirtschaft. In: Zukunftsorientierte Unternehmenssteuerung in der Energiewirtschaft. Springer Fachmedien. 41-73. Wiesbaden (2015)
9. Brakhage, J., Hollerbach, K.: Die „smarte“ Revolution der Energiemärkte. In: Zukunftsorientierte Unternehmenssteuerung in der Energiewirtschaft. Springer Fachmedien. 75-96. Wiesbaden (2015)
10. Europäisches Parlament und der Rat der Europäischen Union. Richtlinie 2009/72/EG des Europäischen Parlaments und des Rates vom 13. Juli 2009, Brüssel (2009)
11. Bundesregierung der Bundesrepublik Deutschland. Energiewirtschaftsgesetz vom 7. Juli 2005 (BGBl. I S. 1970, 3621), Artikel 6 des Gesetzes vom 13. Oktober 2016 (BGBl. I S. 2258), Berlin (2016)
12. Ernst & Young. Kosten-Nutzen-Analyse für einen flächendeckenden Einsatz intelligenter Zähler. Technical report, Ernst & Young GmbH, London (2013)
13. PriceWaterhouseCoopers, Wirtschaftsprüfungsgesellschaft.: Blockchain – Chance für Energieverbraucher?, PWC, Frankfurt am Main (2016)
14. PowerLedger: PowerLedger Project, <https://powerledger.io> (Abgerufen am: 22.09.2017)
15. Hagström, L., Dahlquist, O.: Scaling blockchain for the energy sector, University of Uppsala, Uppsala (2017)
16. Martin, W.: Predictive Analytics –Von Big Data zu Smart Customer Data. In: isreport –Informationsplattform für Business Solutions –Business Guide CRM & ERP 2015, 8-11 (2015)
17. Brignal, M.: Smart meters: an energy-saving revolution or just plain dumb?, Guardian, <https://www.theguardian.com/money/2016/oct/01/smart-meter-energy-saving-revolution-cut-bills-gas-electricity> (2016) (Zugriff am 28.09.2017)
18. Müller-Kirchenbauer, J. und Zenke, I.: Wettbewerbsmarkt für Regel-und Ausgleichsenergie, In: Energiewirtschaftliche Tagesfragen, 51(11), 696 -702, Essen (2001)
19. EPEXSPOT: Intraday-Markt mit Lieferung in einer der deutschen Regelzonen, <https://www.epexspot.com/de/produkte/intradaycontinuous/deutschland> (2017) (Abgerufen am 28.09.2017)
20. Bundesministerium des Innern: Entwurf eines Gesetzes zur Anpassung des Datenschutzrechts an die Verordnung (EU) 2016/679 (Datenschutz-Grundverordnung) und zur Umsetzung der Richtlinie (EU) 2016/680 (Datenschutz-Anpassungs- und -Umsetzungsgesetz EU – DSAnpUG-EU). Berlin (2016)

A.2. Paper Energy of WOW

Disaggregation von Lastkurven privater Haushalte im Kontext der Energiewende – Informationsgewinn für den Energiekonsumenten?

First Author¹, Second Author¹, Third Author^{1,2}, and Fourth Author²

¹ Institution, Department, City, Country

{first.author,second.author,third.author}@example.com

² Other Institution, Other Department, Other City, Other Country

{third.author,fourth.author}@anotherexample.com

Abstract. Dieser Beitrag umfasst eine empirisch-qualitative Erhebung, die im Zusammenhang mit User Experience im Energiemonitoring, durchgeführt wurde. Dafür wurde ein Prototyp entwickelt, der auf Basis des NILM-Toolkits eine Disaggregation von Lastkurven privater Haushalte durchführen kann. Für den Endkunden wurde eine mobile Applikation entwickelt, die die Nutzerdaten aus der Disaggregation anzeigen kann. Zu diesem Zweck wurden Probanden hinsichtlich möglicher Informationen und Interaktionen in einer Applikation befragt. Abschließend wurden diese Ergebnisse evaluiert und Empfehlungen zu den wichtigsten Informationen und Funktionen abgeleitet.

Keywords: User Experience, Energiemonitoring, Machine Learning, Energieinformatik, Disaggregation

1 Motivation

Die Energiebranche unterliegt einem massiven Veränderungsprozess. Im Zuge des starken Wachstums dezentraler Energieversorgung [1] müssen Politik, Unternehmen und Konsumenten neue Lösungsansätze partizipativ entwickeln, um den Stromverbrauch zu minimieren und das Elektrizitätsnetz besser regulieren bzw. entlasten zu können. Der Fokus der großen Energieversorger, mit Bezug auf die Energiewende, galt vorrangig der flächendeckenden Einführung und der Inbetriebnahme von erneuerbaren Energiequellen [2]. Doch durch die EEG-Umlage müssen private Konsumenten einen Teil der Subventionen der erneuerbaren Energien tragen [3] und nehmen somit eine aktive Schlüsselfunktion als Akteur in der Gestaltung der Energiewende ein. Die Einbindung der Konsumenten durch die zeitnahe Bereitstellung von umfassenden Informationen mit denen der Handel präziser gesteuert werden kann um den Energieverbrauchs besser regulieren, kann als vielversprechender Ansatz verfolgt werden. Durch einen höheren Informationsgrad über den eigenen Verbrauch müssen den Konsumenten Hilfestellungen gegeben werden, damit sie optimal an der Energiewende teilnehmen können. Dabei gilt es sowohl informative als

auch technische Herausforderungen zu bewältigen: Dem Konsumenten stehen die aggregierten Informationen über seinen Energieverbrauch bereits zur Verfügung. Mit der Nutzung von digitalen Stromzählern können diese Informationen in variablen Intervallen digital abgerufen, gespeichert und als Lastkurven abgebildet werden [19]. Jedoch bleibt der Informationsgehalt beschränkt. Wird nicht vor jedem Energieverbraucher ein eigener Messpunkt installiert, steht auch bei digitalen Stromzählern nur eine aggregierte Lastkurve aus allen Verbrauchern zur Verfügung. Durch Methoden der Disaggregation von Lastkurven, zur Ermittlung der Lastprofile von Verbrauchern in privaten Haushalten, kann diese Informationslücke geschlossen und detaillierte Informationen für den Konsumenten bereitgestellt werden. Jedoch bleibt die Fragestellung offen, ob die Disaggregation von Lastkurven einen Mehrwert im Sinne einer Wissensgenerierung und Selbstreflektion für den Konsumenten bietet. Um ein besseres Verständnis der Rahmenbedingungen zu erhalten wird in Abschnitt 2 die Energiewende in Deutschland näher beschrieben. In Abschnitt 3 ist die Problemstellung und Forschungsfrage mit Bezug auf die erläuterten Herausforderungen der Energiewende formuliert. Abschnitt 4 beschreibt die prototypische Umsetzung einer Software Applikation zur Disaggregation von Lastkurven. Abschnitt 5 beschreibt die Anforderungserhebung für die Visualisierung der Informationen unter Anwendung von Interviews. Abschnitt 6 schließt mit einem kurzen Fazit und Ausblick.

2 Energiewende in Deutschland

Der Begriff Energiewende steht für die Realisierung einer nachhaltigen Energieversorgung in den Bereichen Energie, Mobilität und Wärme [5]. Die Energiegewinnung aus nuklearen und fossilen Brennstoffen, also Braunkohle und Erdöl soll reduziert werden, sodass bis zum Jahre 2025 40 bis 45 % und bis 2050 mindestens 80 % der gewonnenen Energie aus regenerativen Quellen erzeugt wird [6], [7]. Dies ermöglicht neue Geschäftsmodelle und Beschäftigungen, da innovative Dienste für die Energiewende umgesetzt und zahlreiche neue Arbeitsplätze geschaffen werden können [8]. Ein ähnlicher Trend ist bei der Erstellung von Nachhaltigkeitsberichten zu erkennen. Große Unternehmen sind verpflichtet, ihren Umgang mit Energie transparent offenzulegen [9]. Deutschland bemüht sich flächendeckend Windparks zu errichten, die einen Atomausstieg begünstigen sollen [10]. Eine weitere Möglichkeit grüne Energie zu erzeugen sind Photovoltaikanlagen. Sichtbar entwickelt sich ein Trend, funktionierende Marktmechanismen in diesen Bereichen aufzubauen [11]. Aktuell sind 41,3 Megawatt Photovoltaikleistung installiert und im Jahr 2016 wurden 38 Mrd. kWh Solarstrom produziert. Allein in Rheinland-Pfalz und dem Saarland wurden 2016 knapp 115 MW in Betrieb genommen [12].

Die Marktbarrieren sind indes noch immer vorhanden. Ein großes Problem ist die mangelnde Akzeptanz der Kunden [13], [14]. Des Weiteren ist davon das Vertrauen in die Technik abhängig, dass die Grundlage für die prekäre Rechtssituation darstellt. Konsumenten, oder in diesem Fall Prosumer, dürfen nicht als normale Stromlieferanten zugelassen werden [15].

Darüber hinaus zeichnet sich im privaten Bereich eine neue Entwicklung ab. Aufgrund der Verordnung über energiesparenden Wärmeschutz und energiesparender Anlagentechnik bei Gebäuden, kurzum die Energieeinsparverordnung EnEV, werden gerade private Haushalte und Hausbauer in die Pflicht genommen [16]. Gemäß EnEV darf Strom, der aus erneuerbaren Energien gewonnen wird, bei der sich die Quellen in einem bestimmten Haus befinden, von dem zu berechneten Energiebedarf abgezogen werden. Dies ist der Fall, wenn der Strom im unmittelbaren räumlichen Zusammenhang zu dem Gebäude erzeugt wird und das Resultat der Erzeugung direkt, oder nach vorübergehender Speicherung, selbst genutzt und die überschüssige Energiemenge in ein öffentliches Netz eingespeist wird. Die Berechnungsgrundlage richtet sich nach Art und Umfang der jeweiligen regenerativen Erzeugungsanlage des elektrischen Stroms. Daraus können Informationen gewonnen werden, um die Notwendigkeit einer Energiesenkung neuer Wohngebäude zu forcieren. Eine beim Bau, oder nach dem Bau installierte Photovoltaikanlage führt unmittelbar dazu, dass der gewonnene Strom vom Gesamtenergiebedarf des Hauses abgezogen wird. So sinkt der Primärenergiebedarf des Gebäudes. Diese Energiesparverordnung zur dezentralen Energieerzeugung dient als zentrales Anreizsystem, um über den eigenen Energiebedarf und -verbrauch nachzudenken [17]. An dieser Stelle kann dem Prosumer und Konsumenten durch die Disaggregation von Lastkurven der eigene Bedarf und Verbrauch aufgeschlüsselt als Hilfestellung dienen.

Eine weitere wichtige Entwicklung ist der Austausch von Ferraris-Zähler der privaten Haushalte durch digitale Stromzähler [18]. Die schon seit den 90er Jahren bei Großkunden in Betrieb genommenen Stromzähler speichern digital den Stromverbrauch und können diesen in ein Kommunikationsnetz weiterleiten. Dabei können diese Daten auch aus der Ferne ausgelesen werden, sodass ein Ablesen des Zählerstandes vor Ort nicht mehr notwendig ist. Des Weiteren ist die Frequenz der Übermittlung der Daten höher, wodurch es möglich ist, dem Kunden Informationen auf Monats- oder Tagesbasis anzuzeigen [19]. Dies lässt verschiedene Anwendungsfälle zu, die in disruptive Technologien und neuen Geschäftsmodellen gipfeln können. Je höher die Frequenz dieser Daten ist, desto besser kann zum Beispiel die Disaggregation von Lastkurven einzelne Verbraucher durchgeführt werden. Deshalb sollte die bisherige Entwicklung, intelligente Stromzähler in private Haushalte zu verbauen, weiter fortgeführt werden. Es wird angestrebt intelligente Energiemärkte und Netze zu errichten, um das Potential dieses Technologiewandels auszunutzen [22].

Ein gesellschaftlicher Trend, der sich aus den aktuellen Umweltereignissen ergibt, fordert das Land, beziehungsweise die Gesellschaft dazu auf, ihr Handeln und den Konsum von Energie zu überdenken. Die negativen Auswirkungen der Umweltbelastung durch fossile Brennstoffe und nukleare Kernenergie, nimmt in der gesellschaftlichen Diskussion einen höheren Stellenwert ein. Daraus folgt, dass die Bundesregierung den Erzeugern und Konsumenten Anreize bietet, den Markt neu zu gestalten. Dies führt zu neuartigen Technologien, die den Energiemarkt bestimmen. Zum einen durch den Einbau intelligenter Stromzähler und zum anderen, dass Endverbraucher in die Pflicht genommen werden, sich aufgrund von Gesetzen und Verordnungen, am Ausbau eines intelligenten sowie nachhaltigen Marktes zu beteiligen.

3 Problemstellung

Da jeder Konsument grundsätzlich auf seinen Vorteil bedacht ist, müssen Anreize etabliert werden, diesen Trends und Bestimmungen Folge zu leisten. Dieser Beitrag fokussiert die Identifikation dieser Anreizmechanismen, um eine gesellschaftliche Akzeptanz zu fördern. Es ist davon auszugehen, dass das Vorwissen, die Pflichten und die Bedürfnisse der für die Energiewende im Zentrum stehenden Konsumenten, individuell sind.

Es wird deutlich, dass der Konsument im Zentrum der Energiewende steht. Der Informationszugang des Konsumenten zu seinem Energieverbrauch, um diesen bestmöglich im Sinne der Umwelt und Gesellschaft anpassen zu können, muss aufgrund der verschiedenen Vorkenntnisse einfach gestaltet sein. Deshalb wird der softwarebasierte Zugang in diesem Zusammenhang eine Schlüsselrolle eingeräumt. Die Informationen, die für den Kunden gewonnen werden, sollen nicht durch unsichere Sensorik gesammelt werden, die an verschiedenen Stellen im Haushalt angebracht werden können [20]. Um z. B. Informationen über den Stromverbrauch einzelner Geräte zu sammeln, müssen mehrere Strommessgeräte angeschafft und eingerichtet werden, was Kosten und Aufwand verursacht. In dieser Arbeit wird ein Softwarekonzept vorgestellt, das Lastkurven privater Haushalte disaggregiert. Die Software arbeitet ohne zusätzliche Sensorik und basiert ausschließlich auf den Daten des intelligenten Stromzählers.

Damit der Konsument ein höheres Bewusstsein für Energie bekommen kann, muss die Datenbasis um zusätzliche Informationen ergänzt werden. Über das Anzeigen von energiebezogenen Daten des eigenen Haushalts, kann das Energiekonsumbewusstsein des Anwenders erhöht werden. Diese Informationen könnten durch die Disaggregation von den Lastkurven des Gesamtstromverbrauchs des Haushalts gewonnen werden. Hierbei muss ermittelt werden, wie diese Informationen so einfach wie möglich für den Konsumenten aufbereitet werden können. Hierfür sind insbesondere Informationen über Verbräuche, Kosten oder den aktuellen Strompreis von Bedeutung.

Da eine Software möglichst den Prosumer dazu motivieren soll, mehr Verantwortung bei der Energiewende zu tragen, ist die Frage nach der benötigten Informationsgrundlage zu beantworten. Dahingehend ist die Frage zu beantworten, welche Gewohnheiten ein Konsument preisgeben möchte und inwieweit dies mit der Rechtsprechung vereinbar ist.

Daraus leitet sich folgende Fragestellung ab: Welche Informationen könnten auf Grundlage eines exakt aufgeschlüsselten Energieverbrauchs durch Disaggregation, eingebettet in einer Applikation, dem Konsumenten und somit auch der Energiewende förderlich sein?

4 Prototyp

Das Konzept der Disaggregation von Lastkurven wurde durch einen Software-Prototypen umgesetzt und visualisiert. Der Prototyp verwendet das NILMTK¹ (Non-intrusive Load Monitoring Toolkit), das aufgrund seiner Bedeutung detailliert erklärt wird. Entwickelt wurde NILM, als Basis des intelligenten Energiemonitoring. Das Toolkit basiert auf Plausibilitätskennziffern, die überprüfen, zu wie viel Prozent ein Gerät, auf Basis eines Datensatzes, aus einer Lastkurve erkannt werden kann. Das Ziel war es diesen Prozess im Hintergrund durchzuführen ohne dem Anwender einzubinden. Ein weiteres Ziel war es keine Zwischenmessstationen elektrischer Energie einzusetzen.

Für diesen Prozess wurde das NILMTK 2014 verwendet. Die Funktionsweise des Toolkits wurde im Laufe der Implementierung angepasst, da das NILMTK vom Design, nicht auf eine Echtzeitauswertung ausgelegt ist.

Mithilfe von Metadaten, die das Framework beschreiben und Informationen zu den Daten liefern (Art der Geräte und Aufnahmeparameter), konnte ein Testdatensatz generiert werden. Aus diesen Informationen wird unter Verwendung eines Converters eine Datei im HDF5-Format erstellt. Diese bildet die Basis zur Erstellung einer Trainingskurve und einer Testkurve. Abschließend kann eine Disaggregation durchgeführt werden, die beide Kurven miteinander verarbeitet. Die Testkurve eines einzelnen Endgerätes wird über die Trainingskurve eines Datensatzes gelegt und prozentuale Wahrscheinlichkeiten ermittelt. Das Ergebnis ist die Plausibilitätskennziffer, die die Güte der durchgeführten Disaggregation widerspiegelt. Der Prototyp setzt sich aus verschiedenen Komponenten zusammen. In Abbildung 1 ist ein Komponentendiagramm des Prototyps abgebildet. Die linke Seite der Abbildung stellt die Datenerhebung dar. Mithilfe einer Stromquelle erfasst der digitale Stromzähler Wattwerte, die weiterverarbeitet werden. Durch eine optische Schnittstelle, die an dem digitalen Stromzähler angebracht ist, werden die Wattwerte in eine InfluxDB geschrieben. Die InfluxDB ist eine spezielle Datenbank, die die Messwerte mit einem dazugehörigen eindeutig identifizierbaren Zeitstempel speichert. Dieser Zeitstempel dient somit als Primärschlüssel des Tupels. Aus der Datenbank werden die Tupel (Zeitstempel + Wattwerte) mit einem Converter in das HDF5 (H5) Format umgewandelt. Dieses Format benötigt das NILMTK um die Disaggregation durchzuführen. Die Lastkurve wird disaggregiert und die gewonnenen Informationen werden in die Datenbank auf einen neuen Tabellenabschnitt gelegt.

¹ <https://github.com/nilmk/nilmk>

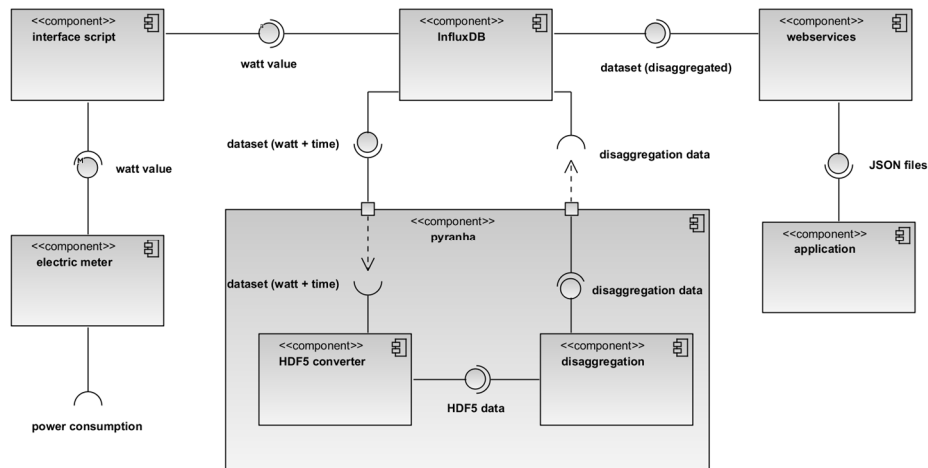


Abbildung 1. Komponentendiagramm des Prototyps zur Disaggregation von Lastkurven.

Für die Visualisierung wurde eine Applikation entwickelt, die konzeptionell und inhaltlich auf dieser wissenschaftlichen Arbeit beruht. Technisch wurde ein Webservice erstellt, der die disaggregierten Werte der InfluxDB im JSON-Format abrufen und für die mobile Applikation bereitstellt.

5 Empirisch-qualitative Erhebung

Für die Evaluation wurden semistrukturierte Interviews durchgeführt. Es wurden Probanden ausgewählt, die sich mit dem Thema Energie intensiver als der durchschnittliche Konsument beschäftigen. Insgesamt wurden acht Probanden befragt und unter Verwendung eines strukturierten Interviewleitfadens nach Mayring [21] durch das Gespräch geführt. Dabei werden die Probanden aufgenommen und das Gespräch anschließend transkribiert.

5.1 Vorgehen

Insgesamt verfügt der Fragebogen über 18 Leitfragen. Zu jeder Fragestellung existieren eine oder mehrere Fragen, um gezielt Informationen aus dem Probanden zu eruiere. Deshalb wird zur Erklärung der Leitfaden in fünf Abschnitte aufgeteilt.

1. Im Abschnitt zur Person wird der berufliche und biografische Hintergrund des Probanden erhoben. Dies hat den Zweck den Probanden an die Sprechsituation zu gewöhnen und ihn für die nachfolgenden Themen kommunikativer zu machen.
2. Der zweite Abschnitt umfasst Fragen zum Energiemarkt. Insbesondere das Bewusstsein zur Energie und dem privaten Verhalten mit Energie soll erhoben werden.

3. Im dritten Abschnitt folgen Fragen zum Nutzungsverhalten von mobilen Endgeräten und wie die Aufbereitung energiebezogener Daten bevorzugt wird.
4. Der vierte Abschnitt ist der Kern des Fragebogens, da dieser direkt in Verbindung mit dem Prototyp zur Disaggregation von Lastkurven steht. Der Proband soll abschätzen, wie die Geräteerkennung im privaten Umfeld eingesetzt würde und welche Informationen nützlich seien.
5. Zum Schluss des Interviews soll der Proband die Disaggregation von Lastkurven kritisch reflektieren und andere Nutzungsszenarien aufzeigen.

5.2 Auswertung

Zur Auswertung wird eine Analyse und Interpretation der Interview-Studie durchgeführt. Als unterstützende Software wird MAXQDA angewendet. Dafür wurde ein Kodierleitfaden erstellt, der die zur systematischen Analyse der Interviews verwendet wird. Die Ergebnisse des Kodierleitfadens werden anhand von drei Oberpunkten beschrieben und interpretiert. Dabei wurde ermittelt, welche primären Informationen in einer App visualisiert und dem Konsumenten zur Verfügung gestellt werden müssen. Diese wurden in zwei Kategorien aufgeschlüsselt, die die Informationen und die Funktionen abbilden. Außerdem gibt es eine allgemeine Kategorie, die die Forschungsfrage thematisiert.

Abgesehen von der Vorbildung und dem Beschäftigungsverhältnis der Probanden, wurde ermittelt, wie diese in Verbindung mit Energie und dem Energiemarkt stehen. Bei 7 von 8 Probanden ist die Selbsteinschätzung, über ihr Bewusstsein für Energie sehr hoch. Sie legen Wert auf Netzstabilität, und sparen aktiv Strom, dabei sind auch energieeffiziente Geräte sehr wichtig. Drei der 7 Probanden legen trotzdem weniger Wert auf ihren Stromverbrauch. Sie relativieren die Relevanz da ihnen Anreize verwehrt bleiben und sie Strom ohnehin verbrauchen. 6 Probanden kontrollieren ihren Stromverbrauch stetig, zwei von ihnen passen den Stromverbrauch aktiv an. Signifikant ist, dass vier Probanden bereits als Produzenten tätig sind, da sie eine Photovoltaikanlage besitzen. Dies hat vorrangig finanzielle Gründe.

Anschließend wurde die Erwartung an Daten aus digitalen Stromzählern der Probanden erfasst. Alle erwarten einen persönlichen Vorteil von den gesammelten Daten, dennoch können diese Daten eine Sensibilisierung der Gesellschaft ermöglichen. Für Energieversorger könnten derartig aufbereitete Daten bessere Planungsmöglichkeiten bedeuten. Die Abbildung von einzelnen Verbrauchsgeräten kann helfen Einsparpotenziale in Haushalten zu identifizieren. Eine Zeitvermarktung ermöglicht die Weitergabe an Unternehmen, um defekte Geräte zu erkennen.

Im Anschluss sollten die Probanden ihre Nutzungsaffinität mobiler Endgeräte einschätzen. Im Kontext einer energiebasierten mobilen Applikation sind zwei Probanden skeptisch. Dennoch kann ein Interesse entstehen, wenn eine Applikation zur Verfügung stehen würde. Als Alternative kann eine festinstallierte Station, anhand des Vorbilds der Photovoltaik-Anlagen, oder auch dezentere Informationen z. B. an einer Küchenuhr, geeignet sein. Eine mobile Website mit Login-Funktion hingegen wurde von einigen Befragten abgelehnt. Auf einem Smartphone soll die Applikation kontextsensitiv und schnell sein, wobei eine gute grafische Aufbereitung wichtig ist. Es

lassen sich kaum Nutzungsintervalle prognostizieren, jedoch würde mit der Zeit das Interesse abnehmen.

Im Zentrum der Interviewstudie stehen Fragen nach der Funktion einer energiebasierten Applikation. Interessant ist die Darstellung des Gesamtverbrauchs sowohl in aktueller Form, als auch auf Tages-, Wochen- und Monatsbasis. Hierbei sind Auswertungsmöglichkeiten zum Energieverhalten, Konsum und auch der Produktion denkbar. Dabei sind Empfehlungen zur Reduzierung des Energieverbrauchs gewünscht. Des Weiteren stehen für alle Probanden die Verbrauchskosten im Vordergrund. Kosten sollen über verschiedene Intervalle einsehbar sein. Außerdem wünschen sich zwei Probanden Amortisationsrechnungen, was eine Kopplung mit der Stromerzeugung oder der Anschaffung von energieeffizienten Geräten benötigt. Eine dritte Funktion ist die Koppelung von Strompreisen und Tarifen. Dabei wäre es interessant, die Tarife flexibel zu gestalten.

Da die Disaggregation von Lastkurven im Fokus dieser Arbeit steht, wird das Hauptaugenmerk auf die Nutzbarkeit der Erkennung von Endgeräten gelegt. Eine Echtzeitdarstellung des Verbrauchs eines Gerätes wird von allen Probanden als unnötig angesehen. Dennoch lassen sich erweiterte Szenarien aus dem Monitoring einzelner Geräte entwickeln. Ganz nach dem Vorbild des Gesamtverbrauchs wünschen sich die Probanden, Geräte im Zeitverlauf, auf Tages- Monats- oder Jahresbasis betrachten zu können. Ein Proband wünscht sich eine Betrachtung nach Tageszeit, um Anomalien entdecken zu können. Durch die Identifikation einzelner Geräte im gesamten Stromverbrauch ist es möglich „Energiefresser“ zu erkennen. Des Weiteren ist es möglich, defekte Geräte zu bestimmen. Dies wird als enormes Potenzial zur Stromeinsparung gesehen. Hierbei könnten sich die Probanden Tipps zu Optimierungen in Form von Benachrichtigungen vorstellen.

Darüber hinaus ist die Integration der Applikation in einer Smart-Home-Anwendung interessant. Dies erweitert die Analyse um eine aktive Steuerung der Geräte. Auch Prognoserechnungen über zu erwartende Stromverbräuche sind denkbar. Um Anreize zu schaffen sind Community-Funktionen interessant, um sich mit den Nachbarn oder einer Gruppe vergleichen zu können. Inhaltlich könnte es um die Vergleichbarkeit von einzelnen Geräten gehen oder auch die Gesamtverbräuche der Haushalte zu bestimmten Zeiten.

Abgesehen von den Informationen, ist auch die Visualisierung und Strukturierung der Informationen ein wichtiger Aspekt. Die Probanden fänden es interessant, über verschiedene, frei wählbare Oberflächen zu verfügen, die ganz nach den Bedürfnissen individualisiert werden können. Dabei sollen die Informationen in Relation zu der Zeit und zum Geld stehen. Trotzdem ist es wichtig am Anfang über ein übersichtliches Dashboard zu verfügen, dass die wichtigsten Informationen aggregiert bereitstellt. Die bereits angesprochenen Benachrichtigungen stellen sich als überaus sinnvolle Funktion heraus. Auf Basis einer Geräteerkennung könnten Benachrichtigungen erfolgen, falls Gerätedurchläufe fertig sind, Geräte zu lange laufen, Anomalien auftreten oder Optimierungshinweise vorliegen. Diese Benachrichtigungen sollten moderat erfolgen. Deshalb sollte die Applikation automatisch Informationen filtern, oder dem Nutzer sollten Konfigurationsmöglichkeiten gegeben werden. Als alternative Form der Benachrichtigungen können E-Mail oder SMS versendet werden.

Am Ende eines jeden Interviews wurde der Proband gebeten, sich Gedanken über die negativen Folgen eines derartigen Energiemonitoring zu machen. Allgemein lässt sich festhalten, dass eine hohe Sensibilität der Datensicherheit vorliegt. Die Probanden haben eine kritische Haltung gegenüber Dritten, die Interesse an derartig fein aufbereiteten Daten haben könnten. Es sind tiefe Einblicke in das Leben eines jeden Menschen möglich, insbesondere falls die Disaggregation von Lastkurven und die daraus resultierende Geräteerkennung umgesetzt wird. Lösungsansätze gibt es mehrere: Grundlegend sollte die Datensicherheit gesetzlich reguliert sein. Dabei sollten mindestens allgemeine IT-Sicherheitsstandards erfüllt sein. Darüber hinaus dürfen nur eingeschränkt Informationen an Dritte weitergeleitet werden. Dies kann durch eine lokale Datenhaltung und Verarbeitung gewährleistet werden. Dafür ist jedoch eine Recheneinheit im Haushalt notwendig. Im Idealfall kann der Nutzer freiwillig entscheiden, was mit seinen persönlichen Daten geschieht.

5.3 Ergebnis und Interpretation der Studie

Diese Studie soll zeigen, wie die Disaggregation von Lastkurven sinnvoll eingesetzt werden kann und welche daraus resultierenden Informationen für Konsumenten Mehrwerte bieten.

Zum einen wird deutlich, dass eine mobile Applikation geeignet ist, da diese zeit- und ortsunabhängig Zugriff auf die Informationen bietet. Zudem werden umständlichen Kontrollmechanismen bei der Authentifizierung vermieden. Der Nutzer hat ortsunabhängig die Möglichkeit sich mit dem Thema der elektrischen Energie zu befassen. Darüber hinaus haben die Konsumenten eine anfangs häufigere Nutzung beschrieben, die mit der Zeit wohl abnehmen würde. Dieses Nutzerverhalten ließe sich sehr gut mit einer mobilen Applikation abdecken und trägt zur angesprochenen Sensibilisierung der Gesellschaft bei. Deshalb kann festgehalten werden, dass die Informationen durch ein mobiles Medium zu dem Konsumenten geführt werden sollten.

Zum anderen wird von den Probanden die Aufbereitung der Informationen stark thematisiert. Insbesondere beim Fokus auf die Durchschnittskonsumenten, wird auf eine einfache Darstellung Wert gelegt. Ein Dashboard als Startseite einer mobilen Applikation, das die wichtigsten Informationen direkt aufzeigen kann, wird von allen Nutzern präferiert. Des Weiteren kann auf dem Dashboard optional auf verschiedene, detaillierte Seiten innerhalb der App verwiesen werden. Die Daten müssen grafisch sinnvoll aufbereitet sein und in Relation zu anderen Größen, wie Zeit oder Währung, gebracht werden. Gleiches gilt für die Analyse einzelner Verbraucher. Anhand variabler Zeitintervalle muss der Konsument einen Einblick in den Verbrauch erhalten. Unerfahrene Konsumenten werden somit Möglichkeiten eröffnet, die Anomalien zu erkennen und den Stromkonsum zu optimieren. In Kombination mit einer Benachrichtigungsfunktion erscheint die Geräteerkennung noch nützlicher. Hierbei darf der Konsument jedoch nicht zu sehr gestört werden, sodass eine einfache Benachrichtigung per Push-Funktion, bei defekten und bei angelassenen Geräten, individuell angepasst werden sollte. Die Benachrichtigung bei nicht ausgeschalteten Geräten, lässt sich durch die Verbindung mit der Standortfunktion eines Mobiltelefons,

umsetzen. Dabei sind auch Betrachtungen der Vergangenheitswerte interessant, damit der Konsument sein Energieverhalten selbst reflektieren kann.

Es lassen sich folgende Kernanforderungen an eine Softwareanwendung zur Optimierung des Energiebedarfs (Tabelle 1) stellen:

Table 1. Anforderungen an die Applikation

<i>Anforderung</i>	<i>Beschreibung</i>
Mobile Applikation	Informationen werden dem Konsumenten durch eine mobile Applikation angeboten.
Dashboard	Ein Dashboard, das Informationen zum Gesamtverbrauch anzeigt, wahlweise in Relation zum Strompreis.
Unterseite mit Stromverbrauch einzelner Verbraucher	Einzelne Verbraucher werden in Relation zu den Stromkosten gesetzt und historische Verbrauchsprofile angezeigt.
Benachrichtigungsfunktion	Benachrichtigungsfunktion auf Basis des Push-Prinzips mit Filterfunktionen
Unterseite mit Gesamtstromverbrauch	Vergangenheitswerte des Gesamtstromverbrauchs
Lokale Datenverarbeitung	Die Datenverarbeitung durch die Disaggregation muss lokal gehalten sein und Daten müssen lokal gespeichert werden.

6 Reflexion und Ausblick

Die Funktionsweise der Prototypen bietet den Konsumenten eine Vielzahl an Informationen die durch die ermittelten Anforderungen sinnvoll in dessen Alltag eingebunden werden können. Ob diese Information einen Beitrag zur Energiewende leisten können muss über einen längeren Zeitraum getestet werden. Fest steht jedoch, dass der Prototyp, anders als bei Sensorik oder intelligenten Steckdosen, störend in das Leben des Konsumenten eingreift. Problematisch ist, dass derartig sensible Daten für negative Zwecke missbraucht werden können. Dem Konsumenten muss klar sein, dass der Energielieferant viele Informationen über das Verhalten sammeln kann, deshalb müssen im Rahmen der Rechtsprechung Regulierungen festgelegt werden.

Weiterführend ist die Entwicklung der mobilen Applikation mit den Anforderungen angestrebt. Anschließend sind umfangreiche Feldtest notwendig, um individuelle Bedürfnisse abzudecken. Deshalb muss die Stichprobengröße der Interviews erweitert werden um genauere Informationen zu den Anforderungen zu erhalten.

Die Disaggregation von Lastkurven ist eine von vielen Möglichkeiten disruptive Technologien, wie den intelligenten Stromzählern, in der Energiewirtschaft nutzbar zu machen. Erkennbar wichtig ist jedoch, dass die Informationen einfach gehalten und für den Konsumenten zugänglich sind.

References

1. Energietechnische Gesellschaft im VDE: Dezentrale Energieversorgung 2020. www.vde.com/etg (2007)
2. Synwoldt, C.: Dezentrale Energieversorgung mit regenerativen Energien. Springer Vieweg, Wiesbaden (2016)
3. Prognose der EEG-Umlage 2017 nach AusglMechV. <https://www.netztransparenz.de/EEG/EEG-Umlage/EEG-Umlage-2017> (2016)
4. Giacobelli, S: Die Energiewende aus wirtschaftssoziologischer Sicht. Springer VS, Wiesbaden (2017)
5. Maubach, K.: Energiewende, Wege zu einer bezahlbaren Energieversorgung. Springer VS, Wiesbaden (2013)
6. Bundesministerium für Energie und Wirtschaft <http://www.bmwi.de/Redaktion/DE/Dossier/energiewende.html> (Accessed (05.09.17))
7. Bundesministerium für Wirtschaft und Energie: Ein Strommarkt für die Energiewende, Berlin (2015)
8. Bardt, H., Kempermann, H.: Folgen der Energiewende für die Industrie. Institut der deutschen Wirtschaft Köln Medien GmbH, Köln (2013)
9. Franz, P.: Nachhaltigkeitsberichterstattung: Empfehlungen für eine gute Unternehmenspraxis. Bundesministerium für Umwelt, Naturschutz und Reaktorsicherheit (BMU), Berlin (2009)
10. Landeszentrale für politische Bildung Baden-Württemberg: Die Energiewende 2011. <http://www.lpb-bw.de/energiewende.html> (Access 05.09.17)
11. Erneuerbare Energien: Der Photovoltaikmarkt in Deutschland wird 2017 wieder wachsen. <https://www.erneuerbareenergien.de/der-photovoltaikmarkt-in-deutschland-wird-2017-wieder-wachsen/150/436/99880/> (Access 05.09.17)
12. Quaschnig, V.: Beitrag erneuerbarer Energien am Stromaufkommen in Deutschland. <http://www.volker-quaschnig.de/datserv/ren-Strom-D/index.php> (Access 05.09.17)
13. Statista: Inwieweit sind Sie für oder gegen den Gebrauch von Solarenergie in Deutschland? <https://de.statista.com/statistik/daten/studie/196147/umfrage/meinung-zum-gebrauch-von-solarenergie-in-deutschland/> (Access 05.09.17)
14. Langer, E.: Akzeptanz für erneuerbare Energien. CARMEN e.V. Straubing (2014)
15. Marx, E., Der Tagesspiegel: Zum Nachbarn für eine Tasse Strom. <http://www.tagesspiegel.de/politik/energiewende-zum-nachbarn-fuer-eine-tasse-strom/13666956.html>
16. Bundesministerium für Umwelt, Naturschutz, Bau und Reaktorsicherheit: Energieeinsparverordnung. <http://www.bmub.bund.de/themen/bauen/energieeffizientes-bauen-und-sanieren/energieeinsparverordnung/> (Access 05.09.17)
17. Giacobelli, S: Die Energiewende aus wirtschaftssoziologischer Sicht. Springer VS, Wiesbaden (2017) pp?
18. Greveler, U.: Die Smart-Metering-Debatte 2010-2016 und ihrer Ergebnisse zum Schutz der Privatsphäre. Datenbankspektrum 16, 137-145 (2016)

19. Maubach, K.D.: Energiewende – Wege zu einer bezahlbaren Energieversorgung. Springer VS, Wiesbaden (2014)
20. Heinrich, B., Linke, P., Glöckler, M.: Grundlagen der Automatisierung - Sensorik, Regelung, Steuerung. Springer Fachmedien, Wiesbaden (2015)
21. Mayring, P.: Qualitative Inhaltsanalyse: Grundlagen und Techniken. Beltz Verlag, Weinheim und Basel (2015)
22. Aichele, C., Doleski, O.: Smart Market: Vom Smart Grid zum intelligenten Energiemarkt. Springer Verlag, Wiesbaden (2014)

A.3. Paper Energy of TOMORROW

Bereinigung von saisonalen Effekten zur Relativität von homogenen Energieverbrauchsdatenbeständen

Abstract. Durch die Energiewende und die flächendeckende Installation digitaler Stromzähler ergeben sich für die Energiebranche neue Marktherausforderungen. Die kontinuierlich wachsende Menge an Daten ermöglicht gleichzeitig Verbrauchern ein feingranulares Bild über eigenen Verbrauch und Produktion. Die Erstellung von Prognosen auf Energieverbrauchs- und Erzeugungsdatenbeständen bildet die Grundlage verschiedener Mehrwertdienste. Durch exaktere Prognosen können Energiekosten ganzheitlich durch Minimierung von Ausgleichsenergie gesenkt werden.

Keywords: Prognosen, Energieverbrauch, Saisonalität, Stationarität, Ausgleichsenergie

1 Einleitung und Motivation

Daten erhalten einen immer höher werdenden Stellenwert und werden in Industrie und Forschung immer mehr als „Rohstoff“ für heterogene Mehrwerte betrachtet. Die massenhafte Produktion und Verfügbarkeit von Daten aus heterogenen Quellen ermöglicht neue Impulse für die Entwicklung von innovativen Produkten, Diensten und Geschäftsmodellen und findet auch immer stärker Eingang in den Energiesektor, insbesondere auch in Privathaushalte [1].

Im Bezug zur Energiewende wurde die Bedeutung von Daten erkannt. Wichtige Themen sind die erneuerbaren Energien und deren Einbindung in das Energienetz. Ein zentraler Aspekt im Energiebereich ist die technologische Entwicklung. In den nächsten Jahren ist die flächendeckende Implementierung von digitalen Stromzählern ein zentrales Ziel der Energiewende [2], [3]. Die bislang üblichen analogen Stromzähler (Ferraris-Zähler) haben einen großen Nachteil, denn diese sind nur in der Lage den absoluten Energieverbrauch zu messen, ohne die Zuordnung zu diskreten Zeitpunkten. Die absoluten Werte ergeben sich ausschließlich durch die Bilanzierung der Jahreswerte [3]. Digitale Zähler zeichnen Energiedaten in wesentlich kürzeren Intervallen auf und können durch Schnittstellen ausgelesen werden. Aufgrund der feingranularen Aufzeichnung ist die Erstellung von Verbrauchsprognosen, auf historischer Daten ein möglicher Anwendungsfall. Daraus ergibt sich eine visuelle Darstellung der Verbräuche und kurze Reaktionsintervalle des Verbrauchers [4].

Der Schwerpunkt dieser Arbeit liegt auf der Prognose von Energieverbräuchen. Besonders die Komplikation durch von saisonale Effekte sowohl bei Energieverbrauchsbeständen, als auch bei Prognosen. Die saisonalen Verläufe eines ganzen Jahres ergeben sich im Wesentlichen aus den Jahreszeiten. Der Energieverbrauch in den Sommermonaten ist geringer als in den Wintermonaten. Auch Feiertage können dazu führen,

dass der Energieverbrauch steigt oder sinkt. Die dargestellten Zeitreihen (Fig. 1 und 2) zeigen dies exemplarisch.

In der Vergangenheit waren Haushalte auf dem Energiemarkt ausschließlich als Nachfrager repräsentiert, doch seit einiger Zeit können sie auch als Energieproduzent auf dem Markt agieren [5]. Zentraler Treiber ist die Weiterentwicklung von Systemen, die es Haushalten ermöglichen, regenerativ selbstständig Strom zu erzeugen. Somit würden diese nicht nur als Konsumenten gelten, sondern auch als Produzenten, sogenannten *Prosumern* [5]. Für diese Haushalte können Prognosen über Produktion und Verbrauch eine wichtige Rolle einnehmen.

2 Forschungsansatz

Im Jahr 2016 hat der Bundestag verabschiedet, dass bis zum Jahr 2032 sämtliche mechanischen Ferraris-Zähler durch digitale Stromzähler ersetzt werden sollen [6]. Für den Verbraucher entfällt durch die Installation eines digitalen Stromzählers das manuelle Ablesen des Verbrauchs, da der Zählerstand automatisch übermittelt wird. Durch das ständige Monitoring können die Stromkosten zudem feingranularer betrachtet werden als auf herkömmlichen Stromzählern. Weiterhin können in Echtzeit Entscheidungen über das eigene Handeln getroffen werden, um bspw. große Verbraucher bei günstigen Konditionen einzuschalten [7]. Digitale Stromzähler zeichnen Energieverbrauchs- und Erzeugungsdaten auf, mithilfe derer Prognosen auf Basis größerer Datenmengen durchgeführt werden können. Realitätsnahe Prognosen über den Stromverbrauch können für Prosumer einen Mehrwert darstellen und sein Handeln im Energienetz sowie im privaten Haushalt beeinflussen.

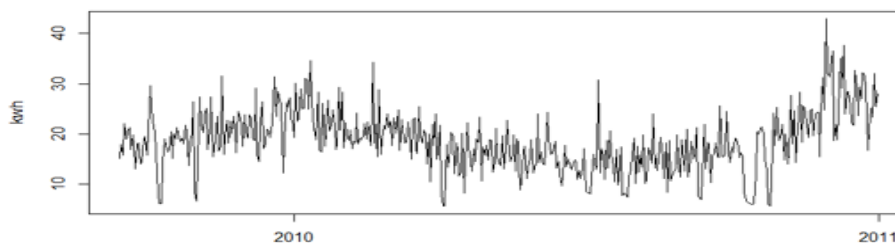


Figure 1: Energieverbrauch eines Haushalts über ein Jahr

Heterogene Lebensweisen und unterschiedliche Bewohnerzahlen innerhalb der Haushalte führen zu unterschiedlichen Lastprofilen. Eine Saisonalität in den Lastprofilen wird durch die Jahreszeiten determiniert, jedoch sorgen Ereignisse, wie z.B. Urlaubszeiten, Wetterumstände oder Feiertage für Abweichungen. Diese Umstände erschweren das Anwenden herkömmlicher Modellierungsmethoden auf solchen nicht-homogenen Datenquellen. Als Datengrundlage dienen Energieverbrauchsdaten, die durch installierte digitale Stromzähler in verschiedenen Haushalten aufgezeichnet wurden. Im Rahmen explorativer Datenanalyseansätze wurde deutlich, dass diese Daten keine Homogenität aufweisen, sondern dass sich innerhalb dieser Datenbestände starke heterogene Schwankungen zeigen. Diese Beobachtung wird mit Hilfe der Grafiken

(Fig. 1 und 2) verdeutlicht. Dargestellt sind zwei Zeitreihen, die aus zwei verschiedenen Haushalten entstammen. In der oberen Zeitreihe (Fig. 1) ist gut zu erkennen, dass in den Sommermonaten weniger und in den Wintermonaten mehr Strom verbraucht wird, dazu kommen steigende Verbräuche bzw. Lastspitzen an Feiertagen. Die untere Zeitreihe unterscheidet sich deutlich von der oberen. Hierbei ist auffällig, dass der Verbrauch bei diesem Haushalt im Sommer sehr gering ist. Das lässt darauf schließen, dass sich die Bewohner möglicherweise nicht zuhause befanden und zu diesem Zeitpunkt nur von einem fixen Verbrauch ausgegangen werden kann, welcher dauerhaft durch gewisse Großgeräte entsteht. Das gleiche gilt für große Feiertage. Diese Abbildungen stehen stellvertretend für eine Menge von verschiedenen eigens durchgeführten Verbrauchsaufzeichnungen und sollen verdeutlichen, dass es in diesem Sinne kein typisches Verbrauchsverhalten gibt, sondern eine Vielzahl von verschiedenen Verhaltensweisen, weshalb jeder Haushalt individuell betrachtet werden muss.

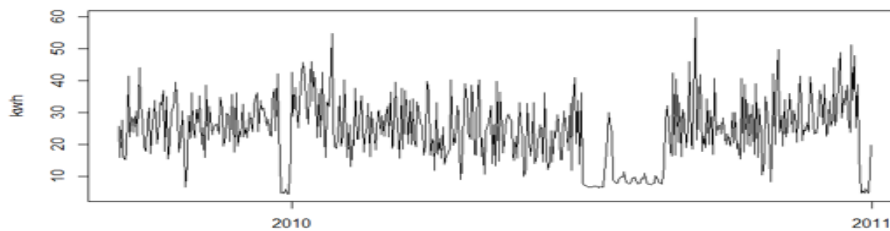


Figure 2: Energieverbrauch eines Haushalts über ein Jahr

Die Vorbereitung der Daten erfolgte in mehreren Schritten. Zunächst wurden verschiedene Hilfsvariablen gebildet. Dazu zählen die stündlichen Werte, die aufzeigen, in welcher Stunde des Tages die Daten aufgezeichnet wurden und Tageswerte (1-365). Im nächsten Schritt wurden die Daten, die aus 15-Minuten-Werten bestanden, auf Stundenbasis aggregiert, um diese wiederum auf Tagesbasis zu aggregieren, da es sich um Tagesprognosen da Tagesprognosen Ziel waren. Für die Aggregation sind die zuvor erstellten Hilfsvariablen von hoher Bedeutung, denn durch diese wird sichergestellt, dass nur vollständige Tage in das Modell übernommen werden. Im Anschluss daran erfolgte die Auswahl von geeigneten Faktoren. Dafür wurden Feiertags-, Ferien- und Wetterdaten des Deutschen Wetterdienst (DWD) berücksichtigt.

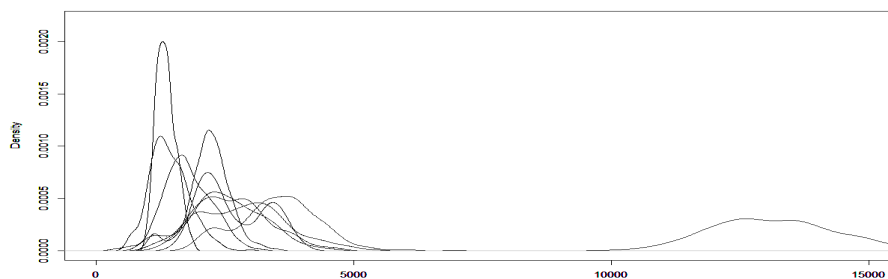


Figure 3: Werteverteilung mittels Dichtefunktion

Die ersten Modelle wurden ausschließlich mit absoluten Zahlen entworfen und führte zu inkorrekten Resultaten. Der Ursprung für dieses Problem lag in den unterschiedlichen Ebenen des Mini- und Maximums der Stromverbräuche. In der Abbildung 3 (Fig. 3) ist die aktuelle Werteverteilung mittels Dichtefunktion zu erkennen: Deutlich werden die unterschiedlichen Stromverbräuche der verschiedenen Haushalte. Dies erschwert die Vergleichbarkeit der Haushalte.

Um dieses Problem, das durch die heterogenen Datenquellen resultiert zu lösen, wurden die Werte aller Haushalte normalisiert. Der erste Schritt bestand darin, dass das Minimum jedes Haushalts von jedem Messpunkt des Haushalts subtrahiert wurde. Anschließend erfolgt die Berechnung des arithmetischen Mittels von jedem Haushalt. Abschließend wurde jeder Wert (Wert nach der Subtraktion des Minimums) mit dem arithmetischen Mittel dividiert. Die Berechnung erfolgte mit Hilfe der folgenden Formel, wobei x_n der normalisierte Wert von x darstellt:

$$x_n = \frac{x - x_{\min}}{x_{\text{mean}} - x_{\min}} \quad (1)$$

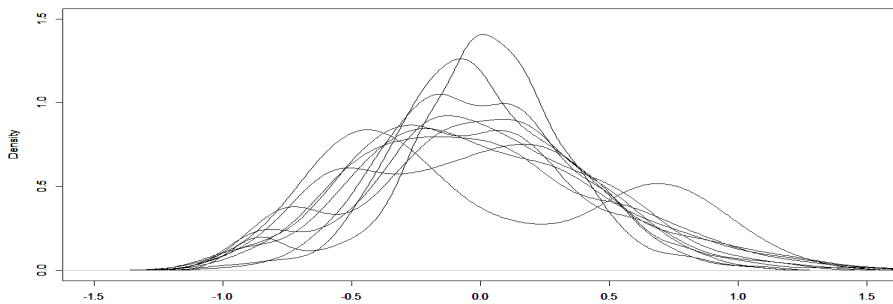


Figure 4: Normalisierte Werteverteilung

Durch die Normalisierung anhand des arithmetischen Mittels, an Stelle des Maximums, wird verhindert, dass ein besonders hoher Stromverbrauch an vereinzelten Tagen, die Normalisierung verzerrt. Die Berechnung erfolgt unter Verwendung des arithmetischen Mittels, da dieses als sensibel gegenüber Ausreißern ist. Nach der Durchführung der Normalisierung ergibt sich die Werteverteilung wie in Abbildung 4 (Fig. 4) dargestellt. Die Haushalte liegen dichter beieinander und können miteinander verglichen werden. Es kann von einem fixen und variablen Verbrauch ausgegangen werden. Nachdem die Daten vorbereitet wurden, konnte dass auf einer linearen Regression basierende Modell erstellt werden. Für die Prognose der zukünftigen Energiewerte wird der gleitende Durchschnitt einbezogen, der auf die bereits bekannten Daten der letzten sieben Tage zurückgreift. Ein Vorteil des gleitenden Durchschnitts ist die Reduzierung von Prognosefehlern [8].

Eine Zeitreihe besteht aus drei Komponenten: dem Trend, den saisonalen Schwankungen und den irregulären Schwankungen [9]. Zum einen sollen die verwendeten Daten von saisonalen Komponenten befreit werden. Saisonale Komponenten beinhalten

sowohl klimatische Einflüsse und institutionelle Faktoren [10], die an dieser Stelle bereinigt werden sollen. Zum anderen soll auch kein Trend in die Berechnung der zu prognostizierenden Verbräuche einfließen. Somit soll auf eine stationäre Zeitreihe geführt werden [11], [12]. Für Saisonbereinigungen lassen sich verschiedene Methoden anwenden, wie z. B. Berechnung eines gleitenden Durchschnitts, Berechnungen durch exponentielles Glätten, Bereinigung der additiv oder multiplikativ überlagerten konstanten Saisonfigur oder variablen Saisonfigur. Eine saisonbereinigte Zeitreihe würde man entsprechend dann erhalten, wenn von der unbereinigten Zeitreihe die Saisonveränderungszahl abgezogen wird [11]. Auf der anderen Seite müssen die Daten, die im Verbrauchsmodell genutzt werden, normalisiert sein. Dabei soll der Betrag der Verbräuche, der in jedem Haushalt in vollem Umfang vorhanden ist, abgezogen werden. Stationarität bezeichnet Eigenschaften, die nicht nur für einzelne Zeitpunkte gelten, sondern Invarianten über die Zeit hinweg sind. Die Zeitreihe hat zu allen Zeitpunkten den gleichen Erwartungswert und die gleiche Varianz.

$$\mathbf{Erwartungswert (Mittelwert) = Varianz} \quad (2)$$

$$E[X_t] = \text{const} = X = \text{var}[X_t] = \text{const} = \sigma_X^2 \quad (3)$$

Stationäre Zeitreihen enthalten keine periodischen Schwankungen und weisen somit keinerlei Veränderungen im Gesamtbild auf. Das bedeutet, dass Zeitreihen nicht mit der Jahresperiode schwanken und keine zyklischen Verläufe enthalten. Es werden sowohl positive, als auch negative Steigungen über eine lange Zeitdauer erkennbar [11], [12]. Für die Überprüfung der Stationarität können verschiedene Methoden angewendet werden. In dieser Arbeit wird der Augmented Dickey-Fuller Test, der Ljung-Box Test und der Kwiatkowski-Philips-Schmidt-Shin Test durchgeführt.

Des Weiteren gilt als Anforderung, dass das im nächsten Schritt erstellte Verbrauchsmodell mit einer möglichst hohen Wahrscheinlichkeit die Werte prognostizieren soll. Dazu können verschieden Kennzahlen formuliert werden, die solche Prognosewerte revidieren. Dies kann zum Beispiel mithilfe von Prognosegütemaßen wie *Mean Square Error* (MSE), *Mean Absolute Percentage Error* (MAPE), *Theilsche Ungleichheitskoeffizient* (TUK), Bestimmtheitsmaß R^2 oder weitere durchgeführt werden. Auf Basis dieser Maße kann festgestellt werden, welcher Prognosewert am besten ausfällt. Anhand bestimmter Faktoren muss entschieden werden, welches Maß bei der Berechnung des besten Prognosewertes zur Zeit der Berechnung angewandt werden soll [13], [14]. Nach Andres und Spiwoks gilt jedoch, dass der TUK neuer Art wesentliche Vorteile im Vergleich zu anderen Prognosegütemaße vereinigt und häufig als "Goldstandard" der Prognosegütemessung betrachtet wird [14]. Demzufolge kann ein intelligenter Algorithmus formuliert werden, der genau diesen besten Prognosewert auf Basis der Prognosegütemaße identifiziert. Um die Stationarität der transformierten Daten zu zeigen, wird aus der Grundgesamtheit der Daten, eine Stichprobe entnommen. Die Stichprobe setzt sich aus zehn zufällig ausgewählten Aufzeichnungen von Haushalten zusammen. Zunächst wurden alle Messwerte, wie zuvor beschrieben, normalisiert und danach drei Tests zur Stationarität unterzogen: Augmented Dickey-Fuller Test, Ljung-Box Test und dem Kwiatkowski-Philips-Schmidt-Shin Test. Der Ergebnisse sind in Tabelle 1 dargestellt.

Tabelle 1. Testergebnisse

Meter ID	Augmented Dickey-Fuller (p-value)	Ljung-Box (p-value)	Kwiatkowski-Philips-Schmidt-Shin (p-value)
2848	0,01	0,000988492	0,1
6078	0,01	0	0,1
3622	0,01	6,95E+08	0,1
6684	0,01	0,001520242	0,1
7054	0,01	0,009192771	0,1
1290	0,01	5,71E+08	0,1
4394	0,01	3,89E+03	0,1
6740	0,01	0	0,1
4552	0,01	0	0,1
3924	0,01	1,48E+04	0,1

Die geringen P-Werte in den drei Tests bei allen Stichproben zeigt, dass die transformierten Daten stationär sind. Die Berechnung der P-Werte können beim Augmented Dickey-Fuller Test und beim Kwiatkowski-Philips-Schmidt-Shin Test kein Ergebnis unter 0,1 ausgeben. Nachdem im ersten Schritt die Daten modelliert und normalisiert wurden, galt es diese im nächsten zu bereinigen. Dabei wurden bei sämtlichen Daten sowohl die Saisonalität, als auch der Trend entfernt. Anhand dieser Daten konnte gezeigt werden, dass es sich um stationäre Daten handelt. Anhand der geringen P-Werte bei allen drei Testdurchläufen wurde gezeigt, dass es sich bei den transformierten Daten um stationäre Daten handelt. Die Saisonalitäten der verschiedenen Haushalte konnten bei der Modellierung von Prognosemodellen berücksichtigt werden.

3 Fazit und Ausblick

Energiedatenbestandsprognosen werden in naher Zukunft einen immer höheren Stellenwert einnehmen, da diese eine Möglichkeit bieten, die gesammelten Daten der digitalen Stromzähler verwertbar zu machen. Da diese Daten aus heterogenen Datenquellen stammen, wird die Erstellung von Prognosemodellen deutlich erschwert. Dies bezieht sich sowohl auf das tagtägliche Verhalten eines Haushalts, als auch im Vergleich mit anderen Haushalten. Die Lösung des Problems ist die Bereinigung von Saisonalitäten und Trends. Dies führt zu höherer Vergleichbarkeit der Daten und daher zu performanteren Prognosemodellen. Wichtig ist, dass die Daten eine Stationarität aufweisen, welche mit Hilfe von drei verschiedenen Tests gezeigt werden konnte. In Bezug auf den dezentralen Energiehandel lässt sich diese Methode auf den Einkauf und Verkauf von Energie übertragen. Auf Basis bestimmter Prognosezahlen, die durch Vorhersagemodelle berechnet werden, können Entscheidungen für den Ein- oder Verkauf von Energie getroffen werden. Mit Blick auf aktuelle Forschungsarbeiten insbesondere im Bereich dezentraler Energieversorgung stellt diese Arbeit eine wichtige Grundlage dar. Das heutige Energiesystem ist auf zentrale Erzeugungsanlagen ausgerichtet und wird, abgesehen von Großverbrauchern (>100.000 kWh/a) mithilfe von Standardlastprofilen gesteuert. So lassen Energieverbrauchs- und Erzeugungsprognosen auf Haushaltsebene

eine sehr feingranulare Netzplanung zu. Aus feingranularen, qualitativ hochwertigen Prognosen folgen für den Kunden geringere Energiekosten und für den Netzbetreiber eine effizientere Netzplanung, da die erforderliche Ausgleichsenergie minimiert wird. Geringere Produktion von Ausgleichsenergie führt zu niedrigeren Pönlzahlungen und daher zu insgesamt geringeren Energiekosten [17]. Im nächsten Schritt werden basierend auf dieser Arbeit Prognosemodelle auf Haushaltsebene weiterentwickelt und für verschiedene Anwendungsfälle bereitgestellt.

4 Referenzen

1. Fraunhofer Institut: Jahresbericht 2016: Chancen zur Digitalisierung, <https://www.fraunhofer.de/content/dam/zv/de/publikationen/Jahresbericht/jb2016/Fraunhofer-Jahresbericht-2016.pdf> (zugegriffen am: 13.09.2017)
2. Bundesregierung: Energiewende im Überblick, <https://www.bundesregierung.de/Content/DE/StatistischeSeiten/Breg/Energiekonzept/0-Buehne/ma%C3%9Fnahmen-im-ueberblick.html> (zugegriffen am: 24.06.2017)
3. Roy, D. T.: Intelligente Energiesysteme der Zukunft: Die Entwicklung von Digitaler Stromzählung und Smart Grid im Jahre 2025. Hamburg, Diplomica Verlag GmbH (2015)
4. Portilla, K.: Will smart meters really help homeowners save energy?, <https://www.theguardian.com/environment/blog/2013/jun/28/smart-meters-homeowners-save-energy> (zugegriffen am: 24.06.2017)
5. Institut für ökologische Wirtschaftsforschung: Haushalte werden relevante Akteure auf dem Energiemarkt, <http://www.prosumer-haushalte.de/das-projekt/ausgangslage.html> (zugegriffen am: 22.07.2017)
6. Aichele, C., Doleski, O. D.: Digitaler Stromzähler Rollout – Praxisleitfaden zur Ausbringung intelligenter Zähler. Springer, Wiesbaden (2013)
7. McHenry, M. P.: Technical and governance considerations for advanced metering infrastructure/digitaler Stromzählers: Technology, security, uncertainty, costs, benefits, and risks (2013)
8. Herrman, F.: Operative Planung in IT-Systemen für die Produktionsplanung und -steuerung – Wirkung, Auswahl und einstellhinweise. Vierweg+Teuber Verlag, Wiesbaden (2011)
9. Newbold, P., Bos, T.: Introductory Business Forecasting. South-Western Pub (1990)
10. Linde, P.: Seasonal Adjustment. Statistics Denmark (2005)
11. Schlittgen, R.: Angewandte Zeitreihenanalyse. Oldenbourg, München (2001)
12. Schlittgen, R., Streitberg, B.: Zeitreihenanalyse. Oldenbourg, München (1994)
13. Feindt, M., Kerzel, U.: Prognosen bewerten - Statistische Grundlagen und praktische Tipps. Springer-Verlag, Berlin Heidelberg (2015)
14. Andres, P., Spiwoks, M.: Prognosegütemaße. State of the Art der statistischen Ex-post-Bewertung von Prognosen. Sofia-Studien zur Institutionenanalyse Nr. 00-1. Darmstadt (2000)
15. Martin, W.: Predictive Analytics – Von Big Data zu Smart Customer Data. In: isreport – Informationsplattform für Business Solutions – Business Guide CRM & ERP 2015, S. 8-11 (2015)
16. Gartner: Gartner's Top 10 Strategic Technology Trends for 2017, <http://www.gartner.com/smarterwithgartner/gartners-top-10-technology-trends-2017/> (zugegriffen am: 27.07.2017)
17. Müller-Kirchenbauer, J. und Zenke, I.: Wettbewerbsmarkt für Regel- und Ausgleichsenergie. In: Energiewirtschaftliche Tagesfragen, 51(11), S. 696 - 702 (2001)

A.4. Interview-Leitfaden Energy of WOW

Leitfaden für das Interview zur Forschungsfrage

Leitfrage	Check	Konkrete Fragen	Aufrechterhaltungs- und Steuerungsfragen
1. Wer sind Sie und was machen Sie?	<ul style="list-style-type: none"> -Beruf -Forschungsschwerpunkt -Studium -Unternehmen 	<p>Welchen Beruf üben Sie in welchem Unternehmen aus? Welchen Forschungsgegenstand? Wie ist die Vorbildung?</p> <p>Wie kommen Sie in Ihrem Beruf damit in Berührung? Welches private Interesse haben Sie daran genau? Welche Forschungsfrage wollen Sie beantworten?</p>	<p>Nonverbale Aufrechterhaltung</p> <p>Können Sie das genauer beschreiben? Haben Sie Beispiele?</p>
2. Welchen Bezug haben Sie zum Energiemarkt?	<ul style="list-style-type: none"> -Beruflich -Privates Interesse -Forschungsschwerpunkt 		
3. Wie ist Ihr eigenes Bewusstsein zur Energie?	<ul style="list-style-type: none"> -Ich besitze XY -Ich spare Strom, Gas, Wasser -Ich Handel mit Energie -Ich Handel mit Wertpapieren im Energiekontext 		
4. Was wird von Ihnen aus Smart Meterdaten erwartet (ggf. Smart Meter erklären)	<ul style="list-style-type: none"> -Genauigkeit der Daten/Frequenz -Diversifizierung der Daten -Proaktive Nutzung -Passiv im Hintergrund 	<p>Welche Meinung haben sie über ein derartiges Energiemonitoring im privaten oder geschäftlichen Bereich?</p>	
5. Besitzen Sie mobile Endgeräte?	<ul style="list-style-type: none"> -Mobiltelefon -Tablet -Organizer 	<p>Um welches Gerät handelt es sich genau? Besitzen sie keine? Besitzen Sie einen PC?</p>	<p>Stellen Sie sich vor sie hätten XY noch nicht. Könnten Sie sich vorstellen, ein Gerät dieser Art in Ihr Leben zu integrieren?</p>
6. Wie und in welchem Kontext nutzen sie ihre Endgeräte?	<ul style="list-style-type: none"> -Planung -Spiele spielen -Nachrichten -Informationsaustausch 	<p>Nutzen Sie Messenger? Nutzen Sie Apps? Nutzen sie die Notizen?</p>	
7. Welche Informationen fänden Sie Interessant, bzw. würden Sie gerne sehen wollen?	<ul style="list-style-type: none"> -Stromverbrauch Echtzeit -aktuelle Netzbelastung -Stromverbrauch der Phasen -Kosten 		<p>Fänden sie wirklich nichts interessant? Weshalb finden sie keine Informationen interessant?</p>

Leitfrage	Check	Konkrete Fragen	Aufrechterhaltungs- und Steuerungsfragen
8. Wo wollen sie diese Informationen am liebsten sehen wollen	<ul style="list-style-type: none"> -App auf dem Handy / Tablet -Am PC -An einer fest installierten Station -Schriftlich per Post 	Was genau spricht dafür? Was genau spricht dagegen?	
9. Stellen Sie sich vor sie haben eine Schnittstelle zum Energiekonsum/Verbrauch und es wäre möglich, einzelne Stromverbraucher zu erkennen, was würde Sie interessieren?	<ul style="list-style-type: none"> -Prozentualer Anteil der einzelnen Geräte am Gesamtverbrauch -Kosten der einzelnen Geräte über eine bestimmte Zeit/ Durchgang -Überblick über Häufigkeit/Dauer der Nutzung einzelner Geräte 	Wie würden Sie es bewerten, dass man auf das einzelne Gerät in Relation zum Gesamtverbrauch schließen kann? Was könnte man daraus ableiten?	Können Sie sich das nicht vorstellen?
10. Wie lange / wie oft würden Sie sich Zeit für das anschauen der Daten nehmen?	<ul style="list-style-type: none"> -selten -oft -Kurz -lang 	Wieso nur so selten / kurz? Warum so oft / lang?	
11. Sehen Sie einen tieferen Sinn in derartiges Energiemonitoring?			
12. Welche Bedenken haben Sie dazu?		Wie könnte eine solche Technologie missbraucht werden? Gibt es ein Bewusstsein hinsichtlich der Kritik an so einer Technik?	
13. Welche Nutzungsmöglichkeiten könnten privat, sowie geschäftlich noch interessant sein?			
14. Würde es Sie interessieren, welche Geräte an welchem Tag / Monat / Jahr wie viel verbraucht haben?		Wissen Sie, wie viel ein Gerät gekostet hat? Wissen Sie, welche Geräte am meisten verbrauchen?	

<p>15. Würden Sie Tagen / Monat / Jahres Daten zu Ihrem Durchschnittsverbrauch vergleichen wollen?</p>			
<p>16. Würden Sie sich Handlungsempfehlungen für die Unterstützung des Verbrauchs wünschen?</p>		<p>z.B. – ungewöhnlicher Verbrauch -Geräte laufen unbenutzt -Urlaubsmodus und Geräte laufen trotzdem -Veränderter Geräteverbrauch</p>	

Abschließende Erklärung

Wir versichern hiermit, dass die vorliegende Arbeit selbständig und ohne fremde Hilfe angefertigt wurde und, dass alle von anderen Autoren wörtlich übernommenen Stellen wie auch die sich an die Gedankengänge anderer Autoren eng anliegenden Ausführungen dieser Arbeit besonders gekennzeichnet und Quellen zitiert sind.

gezeichnet

Meinke, Loose, Bruns, Saager, Taut, Meyer, Nachtigall, Baber, Schierloh, Gerdes

Oldenburg, den 13.10.2017