

Forschungsbericht BWPLUS

Smart Grids und Speicher Sachsenhausen (SMASA)

von

Moritz Benninger, Florian Schirle, Fabian Zippel, Martina Hofmann

Hochschule Aalen
Lehrstuhl für Erneuerbare Energien

Förderkennzeichen: BWSGD 18012

Laufzeit: 13.11.2018 – 30.04.2021

Die Arbeiten des Baden-Württemberg-Programms Lebensgrundlage Umwelt und ihre Sicherung (BWPLUS) werden mit Mitteln des Landes Baden-Württemberg gefördert.

April 2021

Inhaltsverzeichnis

Inhaltsverzeichnis.....	2
1. Einleitung.....	3
2. AP 1 – Projektsteuerung und Öffentlichkeitsarbeit	7
3. AP 2 – Entwicklung des Online Monitoring Systems.....	13
4. AP 3 – Smart Grid Umsetzung	15
5. AP 4 – Geschäftsmodelle entwickeln und testen.....	25
6. AP 5 – Netzdienliche Nutzung der Kundenanlagen und der Biogasanlage mit Wärmenetz.....	62
7. Anschlussfähigkeit und Übertragbarkeit	63
Literaturverzeichnis.....	64
Anhang.....	67

1. Einleitung

Mit dem Projekt Smart Grids und Speicher Sachsenhausen (SMASA) wurden im Rahmen eines Demonstrationsprojekts im Ort Sachsenhausen, bei Giengen an der Brenz, auf Basis eines sogenannten Smart Grids und unter Einbeziehung unterschiedlicher Speicherlösungen neue Geschäftsmodelle für Stadtwerke entwickelt und getestet. Die Zielsetzung war, die größtmögliche Wirtschaftlichkeit für alle Beteiligten bei gleichzeitiger Netzdienlichkeit und Einbindung der regionalen Kunden zu erreichen. Grundlage der vorgesehenen Lösung war dabei der maximale Einsatz von erneuerbaren Energien vor Ort. Auf Basis der regionalen Gegebenheiten standen hier die Stromerzeugung durch Photovoltaik sowie die Wärmeerzeugung durch eine Biogasanlage im Fokus. Zur Erfassung der Lastgänge der lokalen Verbraucher und der Einbeziehung einer Sektorkopplung wurde auf eine kommerziell erhältliche Lösung (energybase von EnBW) zurückgegriffen und diese durch Einbindung der Wärmeseite ergänzt. Weiterhin wurde untersucht, wie die vorhandene und aktuell über EEG vergütete örtliche Biogasanlage eine wirtschaftliche und netzdienliche Lösung unterstützen kann. Als eine technische Besonderheit zur Unterstützung der neuen Geschäftsmodelle wurde ein Online Monitoring System für Komponenten der häuslichen Energieversorgung entwickelt und technisch implementiert. In Abbildung 1 sind die Einbindung der Haushalte in Sachsenhausen (linke Seite) sowie die Anbindung eines einzelnen Haushalts (rechte Seite) beispielhaft dargestellt.

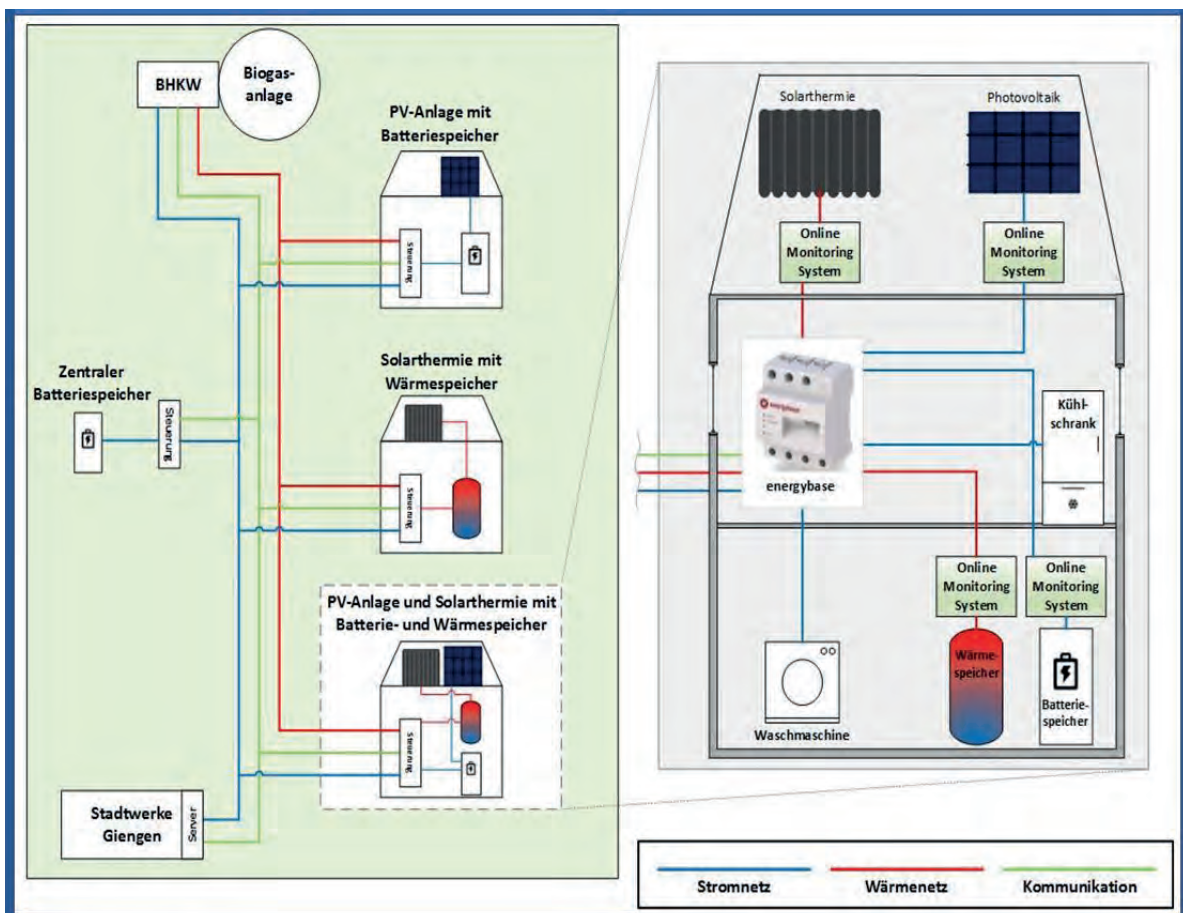


Abbildung 1 Übersicht der technischen Zusammenhänge im Projekt SMASA. Das bestehende Wärmenetz wird von dem BHKW der Biogasanlage gespeist. Die energybase-Steuerung wird hausintern mit sämtlichen Erzeugern und Verbrauchern verbunden. Die Anbindung an das Kommunikationsnetz erfolgt über den jeweils hauseigenen Router.

Vor diesem Hintergrund ergaben sich folgende Arbeitspakete, die anschließend im Zuge der Projektlaufzeit bearbeitet wurden:

AP 1 – Projektsteuerung und Öffentlichkeitsarbeit

AP 2 – Entwicklung des Online Monitoring Systems

AP 3 – Smart Grid Umsetzung

AP 4 – Entwicklung und Testen der Geschäftsmodelle

AP 5 – Netzdienliche Nutzung der Kundenanlagen und der Biogasanlage mit Wärmenetz

Die einzelnen Arbeitspakete gliedern sich weiterhin in Unterziele, in denen die entsprechenden Meilensteine und Zeitpläne des Projektantrags umgesetzt wurden. Eine Übersicht der jeweiligen Arbeitspakete mit den Unterpaketen sowie einen Stand der Umsetzung und eine stichwortartige Darstellung der entsprechenden Ergebnisse und Hindernisse bei der Umsetzung zeigt Tabelle 1. Neben den dargestellten Arbeitspaketen wurde eine Anschlussfähigkeit und Übertragbarkeit sowie eine Verwertung der Ergebnisse erreicht, indem die entwickelten Lösungen in anderen Projekten der Hochschule Aalen eingesetzt und weitergenutzt wurden. Dies war Teil der Projektziele und wird daher im vorliegenden Projektbericht kurz behandelt.

Tabelle 1: Darstellung der Arbeitspakete mit Unterpaketen: Stand der Umsetzung sowie Ergebnisse und Hindernisse der Umsetzung

	Umgesetzt / abgeschlossen	Ergebnisse	in Umsetzung	erste Ergebnisse/ weiteres Vorgehen/ Hindernisse	verworfen	Grund/ alternative Vorgehensweise
AP1 - Projektsteuerung und Öffentlichkeitsarbeit						
Statusmeeting alle 2 Monate	x	Aktueller Zeitplan				
Kick-Off Workshop	x	Finden von Pilothaushalten				
Ergebnisworkshop nach einem Jahr	x	Feedback der Partner				
Ergebnisworkshop zum Projektende	x	Ergebnispräsentation				
Publikation der Projektergebnisse	x	drei Veröffentlichungen				
Teilnahme "Smart Energy and Mobility Solutions"					x	Kongress fand nur 2018 statt
Teilnahme "Smart Grids Kongress"	x	Teilnahme als Besucher/Ziel Networking				
20 Pilotkunden finden	x	Datenaufnahme und Datenspeicherung sichergestellt				
Befragung der Pilotkunden						
zu Projektbeginn	x					
nach Erstellung der ersten Angebote					x	keine Befragung, da kein wirtsch. Geschäftsmodell vorhanden
nach Durchführung des Testlaufs der ersten Angebote					x	siehe vorherigen Punkt

AP 2 - Entwicklung Online Monitoring System						
Aufnahme der Rahmenbedingungen	x	erfolgreich				
Definition Schnittstelle energybase	x	Schnittstelle wurde erstellt				
Auswahl und Zusammenstellung der Hardware und Software	x	System einsatzfähig				
Integration des Online Monitoring Systems in die energybase Steuerung	x	Implementierung umgesetzt (Ethernet und Eingabe IP-Adresse)				
Test Online Monitoring System im Labor	x	Erfolgreich durchgeführt				
Einsatz Online Monitoring System bei Pilotkunden	x	an vier PV Anlagen installiert				
Auswertung der Messergebnisse und Erfahrung bei Kunden	x	Auswertungen sind kontinuierlich erfolgt				
AP 3 - Smart Grid Umsetzung						
Auslegung und Beschaffung der technischen Komponenten	x	erfolgreich				
Anschluss Pilotkunden	x	erfolgreich bei 20 Pilotkunden, zusätzliches Mess-equipment selbst entwickelt				
Durchführung von Messungen mit direkter Auswertung der Daten	x	erfolgreich				
AP 4 - Geschäftsmodelle entwickeln und testen						
Auswertung und Befragung sowie Entwicklung der Konzepte	x	Workshop mit Bürgern und Entwicklung von vier Geschäftsmodellen durchgeführt				
Umsetzung des Kundenfeedbacks in Lösungen	x	Ist erfolgt, kein wirtschaftliches Modell gefunden				
Abstimmung mit dem baden-württembergischen Genossenschaftsverband					x	Nicht möglich, da kein wirtschaftliches Geschäftsmodell vorhanden
Testen der entwickelten Geschäftsmodelle an Pilotkunden					x	Nicht möglich, da kein wirtschaftliches Geschäftsmodell vorhanden
AP 5 - Netzdienliche Nutzung der Kundenanlagen und der Biogasanlage mit Wärmenetz						
Datenaufnahme zur Abbildung des Strom- und Wärmenetzes im Simulationsmodell	x	Daten sind vorhanden				

Simulation des Gesamtsystems mit Stromerzeugung, Biogasanlage, Speichern und Wärmenetz	x	Modell vorhanden; wegen fehlender Geschäftsmodelle nicht weiter verfolgt				
Erarbeitung von Varianten zur wirtschaftlichen Nutzung der Biogasanlage und ggf. eines Quartiersspeichers im lokalen Energienetz					x	Wegen fehlender Geschäftsmodelle nicht abgeschlossen
Anschlussfähigkeit und Übertragbarkeit sowie Verwertung der Ergebnisse des Vorhabens						
Weiternutzung der Hardware			x	Weiternutzungskonzepte sind in Arbeit. Einsatz in Folgeprojekten SeLiG und KEFF+ möglich		
Inhalte Mehrwert für Stadtwerke, insbesondere Geschäftsmodelle					x	Erkenntnis: kein wirtschaftliches Geschäftsmodell
Weiterverbreitung in Vorlesungen	x	Fließt ein in Vorlesung Energiewirtschaft und Energiekonzepte				
Veröffentlichungen und Vorträge	x	Diverse umgesetzt				
Verbreitung durch Verband kommunaler Unternehmen					x	nicht sinnvoll
Weiterverbreitung über EnBW und energybase					x	nicht sinnvoll
Weiterverbreitung über Zusammenarbeit mit Genossenschaftsverband					x	nicht sinnvoll

2. AP 1 – Projektsteuerung und Öffentlichkeitsarbeit

Das Arbeitspaket 1 beinhaltete die Durchführung eines Statusmeetings alle zwei Monate. Als Dokumentation der Statusmeetings dienen die entsprechenden Protokolle und Präsentationen, in denen die aktuelle Entwicklung und die neu erarbeiteten Punkte vorgestellt wurden. Die Protokolle dokumentieren anhand einer entsprechenden Agenda (Top 1, Top 2, etc.) die im damaligen Zeitraum bearbeiteten Arbeitspakete bzw. die damit verbundenen Schwierigkeiten und Fragestellungen. Beispielsweise sind in Abbildung 2 der Meilensteinplan und die jeweiligen Maßnahmen aus der Präsentation eines Statusmeetings dargestellt. Der Meilensteinplan diente als Grundlage, um beispielsweise Zeitpläne anzupassen, offene Fragen zu klären und Zuständigkeiten zuzuweisen. Weiterhin wurden darin die jeweiligen Verantwortlichkeiten für die zu erledigenden Punkte festgelegt, was so auch entsprechend im Protokoll vermerkt wurde. Die jeweiligen Protokolle und Präsentationen sowie weitere Informationen zur Projektorganisation sind an der Hochschule Aalen archiviert.

Arbeitspakete	Jul	Aug	Sep	Okt	Nov	Dez	Jan	Feb	Mrz	Apr
Projektsteuerung und Öffentlichkeitsarbeit										
Kickoff und Abschlussveranstaltung mit allen Beteiligten inklusive Pilotkunden										
Befragungen der Pilotkunden										
Statusmeetings (je nach Projektphase, durchschnittlich alle 2 Monate)										
Entwicklung Online Monitoring System										
Aufnahme der Rahmenbedingungen, Definition der Schnittstelle zur EnergyBase Steuerung										
Auswahl und Zusammenstellung der Hardware und Software										
Integration des Online-Monitoring-Systems in die energybase Steuerung										
Test des Online-Monitoring-Systems im Labor										
Einsatz des Online-Monitoring-Systems bei ausgewählten Pilotkunden										
Auswertung der Messergebnisse und Erfahrungen bei Kunden										
Smart Grid Umsetzung										
Auslegung und Beschaffung der technischen Komponenten										
Anschluss Pilotkunden (Fremdvergabe)										
Durchführung von Messungen mit direkter Auswertung der Daten										
Anschluss zentraler Netzspeicher (im Rahmen von SELIG)										
Geschäftsmodell entwickeln und testen										
Auswertung der ersten Befragung und Entwicklung der ersten Konzepte										
Umsetzung des Kundenfeedbacks in Lösungen										
Abstimmung mit dem Baden-Württembergischen Genossenschaftsverband										
Testen der entwickelten Geschäftsmodelle an Pilotkunden										
Auswertung der Erfahrungen aus dem Test der Geschäftsmodelle										
Optimierung der Geschäftsmodelle										
Erneuter Test der optimierten Geschäftsmodelle										
Netzdienliche Nutzung der Kundenanlagen und der Biogasanlage mit Wärmenetz										
Datenaufnahme zur Abbildung des Strom- und Wärmenetzes im Simulationsmodell										
Simulation des Gesamtsystems mit Stromerzeugung, Biogasanlage, Speichern und Wärmenetz										
Erarbeitung von Varianten zur wirtschaftlichen Nutzung der Biogasanlage und ggf. eines Quartiersspeichers im lokalen Energienetz										

Abbildung 2: Ausschnitt des Meilensteinplans zur Verwendung im Statusmeeting als Teil der Projektorganisation

Weiterhin wurde im Rahmen des Arbeitspakets 1 ein Kick-Off Workshop durchgeführt. Hierzu wurde das Konsortium, bestehend aus dem Netzbetreiber Netze NGO GmbH, dem Energieversorgungsunternehmen SW Giengen sowie der energybase (Teil der EnBW Energie Baden-Württemberg AG) zu einer Auftaktveranstaltung eingeladen. Aufgrund der geplanten Einbindung der Second Life Speicher der Stadtwerke Aalen in das Projekt war auch ein Vertreter der Stadtwerke Aalen anwesend. Zum Auftakt des Workshops wurde eine Präsentation des Projekts sowie der jeweiligen Meilensteine und der zuständigen Personen an der Hochschule Aalen gezeigt. In diesem Zusammenhang wurde die energybase Hardware durch eine Vertreterin der EnBW Energie Baden-Württemberg AG vorgestellt. Weiterhin wurden die Verantwortlichkeiten für die einzelnen Meilensteine festgelegt.

Der nach dem ersten Projektjahr terminierte Ergebnisworkshop entspricht dem dritten Statusmeeting. Dieses Vorgehen wurde gewählt, um im Sinne eines effizienten Vorgehens den jeweiligen Projektpartnern einen zusätzlichen Termin zu ersparen. Im Statusmeeting 3 wurden den Teilnehmern die bisherigen Ergebnisse, wie die Darstellung der ersten gemessenen Daten über die Benutzeroberfläche Graphana sowie die Installation des Messequipments bei den Pilotkunden präsentiert (vgl. hierzu Abbildung 3, welche die Installation des Online Monitoring Systems bei einem Pilotkunden zeigt).



Abbildung 3: Darstellung des bei einem Pilotkunden installierten Online Monitoring Systems

Als weitere Aufgabenschwerpunkte in Arbeitspaket 1 wurde die Veröffentlichung von Projektergebnissen in wissenschaftlichen Publikationen und Konferenzen angeführt. Im Zuge des Projekts wurden bisher folgende drei Publikationen veröffentlicht:

- NEIS 2019 (Conference Proceedings ISBN: 978-3-8007-5152-5) – Peer Reviewed: Benninger, M.; Liebschner, M.; Hofmann, M., "Online Monitoring System for Photovoltaic Systems Using Anomaly Detection with Machine Learning", Schulz, D., NEIS 2019: Conference on Sustainable Energy Supply and Energy Storage Systems, VDE Verlag, Hamburg, 2019, 215 – 220, ISBN: 978-3-8007-5152-5
- NEIS 2020 (unpublished) – Peer Reviewed: Benninger, M.; Hofmann, M.; Liebschner, M., "Anomaly detection by comparing photovoltaic systems with machine learning methods"
- 24th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems 2020 (unpublished) – Peer Reviewed: Benninger, M.; Hofmann, M.; Liebschner, M., „Transferable and self-learning Online Monitoring System for electrical household appliances“

Im Projekt wurde auch eine Teilnahme an der Konferenz "Smart Energy and Mobility Solutions Kongress" erwogen. Diese Veranstaltung fand jedoch einzig 2018 statt und wurde in den Folgejahren nicht mehr angeboten. Als Ersatz diente die Veranstaltung "Smart Grids Kongress", bei der die Projektbearbeiter 2019 teilnahmen. Dabei lag das Ziel des Veranstaltungsbesuchs im Networking und im Austausch von Ideen und Anregungen. So konnte dort neuer Input für das eigene Forschungsprojekt gesammelt, interessante neue Kontakte etabliert und das Projekt auch bei anderen Teilnehmern der

Veranstaltung durch Kommunikation und Austausch im persönlichen Gespräch bekannt gemacht werden.

Im Anschluss an die bereits erwähnte Kick-Off Veranstaltung zu Beginn des Projekts wurden Anschreiben sowie Flyer für die Bürger in Sachsenhausen erstellt, um erste Pilotkunden für das Projekt zu gewinnen. Abbildung 4 und Abbildung 5 zeigen den zur Anwerbung von Pilotkunden entworfenen Flyer).

Zur Umsetzung des Projekts konnten schließlich 20 Pilotkunden gewonnen werden, bei denen die energybase Hardware installiert und zum Einsatz gebracht wurde. Abbildung 6 zeigt die gesamte Ortschaft Sachsenhausen. Dabei befinden sich die ausgewählten Pilotkunden alle innerhalb des Ausschnitts, werden jedoch zum Zweck des Datenschutzes nicht einzeln ausgewiesen. Die ermittelte technische Ausstattung der Pilotkunden umfasst in unterschiedlichen Kombinationen die an den Gebäuden installierte Photovoltaik sowie Solarthermie und Batteriespeicher. Weiterhin wurde der Anschluss an das Sachsenhausener Wärmenetz (sofern vorhanden) bzw. die installierten Wärmepumpen der Pilotkunden erfasst (weitere Details hierzu finden sich in Tabelle 2).

Hochschule Aalen - Technik und Wirtschaft
Beethovenstraße 1
73430 Aalen

In Kooperation mit: **SWG**

Baden-Württemberg
MINISTERIUM FÜR UMWELT, KLIMA UND ENERGIEWIRTSCHAFT

PTKA
Projektträger Karlsruhe
Karlsruher Institut für Technologie

Das Forschungsprojekt Smart Grids und Speicher Sachsenhausen (SMASA) entsteht im Rahmen der Förderung von Demonstrationsprojekten Smart Grids und Speicher Baden-Württemberg. Das Projekt wird in Zusammenarbeit mit dem Kooperationspartner Stadtwerke Giengen durchgeführt.

WERDEN SIE TEIL DER ENERGIEWENDE UND SPAREN SIE GLEICHZEITIG STROMKOSTEN!

Im beigelegten Brief finden Sie weitere Details zum Projekt. Bei Fragen wenden Sie sich an den Ansprechpartner auf der Rückseite. Wir freuen uns über Ihre Teilnahme am Projekt.

IHRE TEILNAHME IST FÜR SIE KOMPLETT KOSTENFREI!

IHR ANSPRECHPARTNER

BENJAMIN WAIBLE
MAIL: benjamin.waible@hs-aalen.de
TEL: +49 (0) 7361 576-4239

SMART GRIDS UND SPEICHER SACHSENHAUSEN


Hochschule Aalen

Abbildung 4: SMASA-Flyer erstellt zum Anwerben der Pilotkunden in Sachsenhausen – Vorderseite

ZUM PROJEKT: SMART GRIDS UND SPEICHER SACHSENHAUSEN (SMASA)

- Realisierung eines intelligenten Netzes (Smart Grid) in Sachsenhausen
- Installation eines Energiemanagers in Ihrem Zuhause
- Nutzung der erneuerbaren Energien vor Ort
- Testen von neuen Angeboten wie All-Inclusive bzw. Flatrates
- Einbindung des bestehenden Wärmenetzes in das Smart Grid
- Frühzeitiges Detektieren von Störungen Ihrer Geräte zur besseren Planbarkeit und Durchsetzung der Wartung

Mögliche Anzeige des Energiemanagers



IHRE VORTEILE

- ✓ **Kostenlose** Teilnahme
- ✓ Lassen Sie sich Ihren Energieverbrauch **digital** anzeigen!
- ✓ Individuelle Planung Ihrer Geräte
- ✓ Steigern Sie den Eigenverbrauch Ihrer Photovoltaik-Anlage!
- ✓ Reduzierung Ihrer Energiekosten
- ✓ Nehmen Sie **aktiv** an der Energiewende teil!
- ✓ Erhöhung der Autarkie

Abbildung 5: SMASA-Flyer erstellt zum Anwerben der Pilotkunden in Sachsenhausen – Rückseite



Abbildung 6: Darstellung der Bebauung in Sachsenhausen (Quelle GoogleEarth), innerhalb des Bildausschnitts befinden sich alle ausgewählten Pilotkunden. Die Biogasanlage, die das Nahwärmenetz versorgt, befindet sich in südlicher Richtung außerhalb des Bildausschnitts

Die bei den Pilotkunden installierte energybase Hardware sammelte über den gesamten Projektzeitraum vorab definierte Daten, die als Datengrundlage für unterschiedliche Arbeitspakete des Projekts verwendet wurden. Die bei den Pilotkunden aufgenommenen Daten wurden dabei auf einem

extra dafür bereitgestellten Server gespeichert, um anschließend von dort heruntergeladen und ausgewertet zu werden. Nach Abschluss der Messperiode, welche über den Projektzeitraum hinaus bis Dezember 2021 weitergeführt wird, werden die gesammelten Daten auf den Server der Hochschule Aalen geladen und stehen so nach einer Anonymisierung Folgeprojekten zur Verfügung. Die installierte energybase Hardware findet in dem Forschungsprojekt SeLiG der Hochschule Aalen Verwendung (SeLiG – Demonstrationsprojekt Second-Life Speicher in Smart Grids, Zuwendungsnummer BWSGD 19011). Dort ist geplant, den in SeLiG für das Reallabor Sachsenhausen vorgesehenen Speicher auf Basis der durch die energybase Hardware im SMASA-Projekt aufgenommenen, sekundengenauen Daten zu regeln. Da eine praktische Umsetzung hier aktuell nicht erfolgt, wird die Umsetzung theoretisch auf Basis von Simulationsmodellen betrachtet.

Weiterhin fand eine Befragung der Pilotkunden zu Projektbeginn statt. Diese diente dazu, die Grundlage für die Simulation der Geschäftsmodelle in Arbeitspaket 4 zu liefern. In die Befragung wurden nur 18 Haushalte miteinbezogen, da die fehlenden zwei Haushalte erst nach der durchgeführten Befragung als Pilotkunden gewonnen werden konnten. Die Ergebnisse der Befragung sind in Tabelle 2 dargestellt. So besitzen rund 72,2 Prozent der Befragten eine Photovoltaikanlage, im Besitz einer Solarthermieanlage sind nur 11,1 Prozent. Im Besitz eines Batteriespeichers ist nur ein einziger Pilotkunde. Beim primären Heizsystem besitzen 66,7 Prozent der Befragten einen Anschluss an das Nahwärmenetz der Biogasanlage, 16,7 Prozent heizen primär mit Biomasse (Holzhackschnitzel bzw. Pellets), 11,1 Prozent heizen mit Wärmepumpen und ein einziger Befragter nutzt eine Solarthermieanlage als primäres Heizsystem. Rund 66,7 Prozent der Befragten haben ein zweites Heizsystem installiert. Die durchschnittlich installierte Leistung der Photovoltaikanlage beträgt 16,6 kW (bei 13 angegebenen PV-Leistungsdaten), der durchschnittliche jährliche Netzbezug der Haushalte an elektrischer Energie beträgt rund 4.933 kWh (bei 13 genannten Verbrauchsangaben).

Auf Basis der erfassten Daten und der darauf aufbauenden Simulation konnte kein tragfähiges Geschäftsmodell abgeleitet werden (vgl. hierzu die Ergebnisse in Arbeitspaket 4). Daher wurde auch von einer ursprünglich laut Arbeitspaket geplanten, weiteren Befragung der Pilotkunden nach Erstellung des tragfähigen Geschäftsmodells sowie nach der Implementierung des tragfähigen Geschäftsmodells abgesehen.

Zum Ende des Projekts wurden die Pilotkunden mit einem Bürgerbrief über den aktuellen Stand des Projekts informiert. Zusätzlich fand am 29. April 2021 die Abschlussveranstaltung des Projekts über eine Onlineplattform statt. Hier wurden die Pilotkunden sowie die Projektpartner Stadtwerke Giengen, Hochschule Reutlingen und Netze NGO eingeladen, um über die erreichten Ziele und Meilensteine zu berichten. Hervorgehoben wurden in der Abschlussveranstaltung die Meilensteine Online Monitoring System, Datenaufzeichnung und Auswertung in den Haushalten und Umspannstationen, die Auslegung eines Netzspeichers auf Basis der Haushaltsdaten und die entwickelten Geschäftsmodelle, da hier die Pilotkunden die meisten Berührungspunkte hatten. Zusätzlich wurde angekündigt, die Messgeräte für eine weitere Messperiode bis Ende des Jahres 2021 in den Haushalten zu belassen und es wurde ein Ausblick gegeben, wie die aufgenommenen Daten verwendet werden. Die Abschlussveranstaltung endete mit einer offenen Runde für Fragen und Diskussion mit den Bürgern.

Tabelle 2: Ergebnisse der Befragung der Pilotkunden zu Projektbeginn in Sachsenhausen

Ifd. Nummer	PV	Solarthermie	Batteriespeicher	Heizungssystem	zweites Heizungssystem	Anschluss ans Wärmenetz	PV Eigenverbrauch	PV Leistung in kWp	Verbrauch Netz in kWh/a	Verbrauch PV in kWh/a
1	Ja	Nein	-	Wärme von Biogasanlage	-	Ja	Nein	8	4.700	-
2	Ja	Nein	Ja	Wärme von Biogasanlage	-	Ja	Ja	8,2	5.000	-
3	Ja	Nein	-	Wärme von Biogasanlage	-	Ja	Nein	29,9	13.000	-
4	Ja	Nein	-	Wärme von Biogasanlage	Holzofen	Ja	Ja	12,6	1.600	1.400
5	Nein	Nein	-	Holzpellets	-	Nein	Nein	-	5.600	-
6	Nein	Nein	-	Wärme von Biogasanlage	Holzofen	Ja	Nein	-	3.250	-
7	Ja	Nein	-	Wärme von Biogasanlage	Kachelofen	Ja	Ja	8,6	2.500	1.000
8	Nein	Nein	-	Wärme von Biogasanlage	Kaminofen	Ja	Nein	-	4.900	-
9	anderer Standort	Nein	-	Wärmepumpe	Elektroheizung	Ja	Nein	-	6.172	-
10	Ja	Nein	-	Holz hackschnitzelheizung	-	Nein	Nein	10,5	5.400	-
11	Ja	Nein	-	Wärme von Biogasanlage	Holzofen	Ja	Nein	8,9	-	-
12	Nein	Nein	-	Wärme von Biogasanlage	Kaminofen	Ja	Nein	-	1.800	-
13	Ja	Nein	-	Wärme von Biogasanlage	Elektroheizung	Ja	Ja	29,35	3.000	2.000
14	Ja	Ja	-	Solarthermie	Holzofen	Nein	Nein	10,58	4.700	-
15	Ja	Nein	Ja	Biomasse	Elektroheizung	Nein	Ja	9,9	1.500 + 2.400 (E-Heizung)	-
16	Ja	Nein	-	Wärmepumpe	Biomasse	Nein	Ja	26	3.200 + 3.600 Wärmepumpe	-
17	Ja	Ja	-	Wärme von Biogasanlage	Solarthermie	Ja	Nein	9,4	6.600	-
18	Ja	Nein	-	Wärme von Biogasanlage	-	Ja	Nein	44	-	-

3. AP 2 – Entwicklung des Online Monitoring Systems

Zur Entwicklung des an den PV-Anlagen eingesetzten Online Monitoring Systems wurden in einem ersten Schritt die Rahmenbedingungen aufgenommen. Diese ergaben sich teils aus technischen Vorgaben sowie aus dem Forschungsvorhaben selbst. So sollte die Einspeiseleistung der Photovoltaikanlage überwacht werden, ohne in die Anlage direkt eingreifen zu müssen. Der Anlagenzustand bzw. der Leistungsfluss wird daher allein aus der Strommessung (anhand einer Strommesszange) abgeleitet. Hierzu wird auf Basis der aufgenommenen Daten eine künstliche Intelligenz trainiert, die unzulässige Anlagenzustände sowie Leistungsflüsse erkennt. Die für das Vorhaben nötige Zusammenstellung an Hard- und Software erfolgte im Frühjahr 2019. Die Hardware besteht aus einem Raspberry Pi, mehreren Gleichstromsensoren sowie einem Analog-Digital-Wandler und ist in einem schützenden Gehäuse verbaut (ein entsprechend verbauter Sensor ist in Abbildung 7 dargestellt).

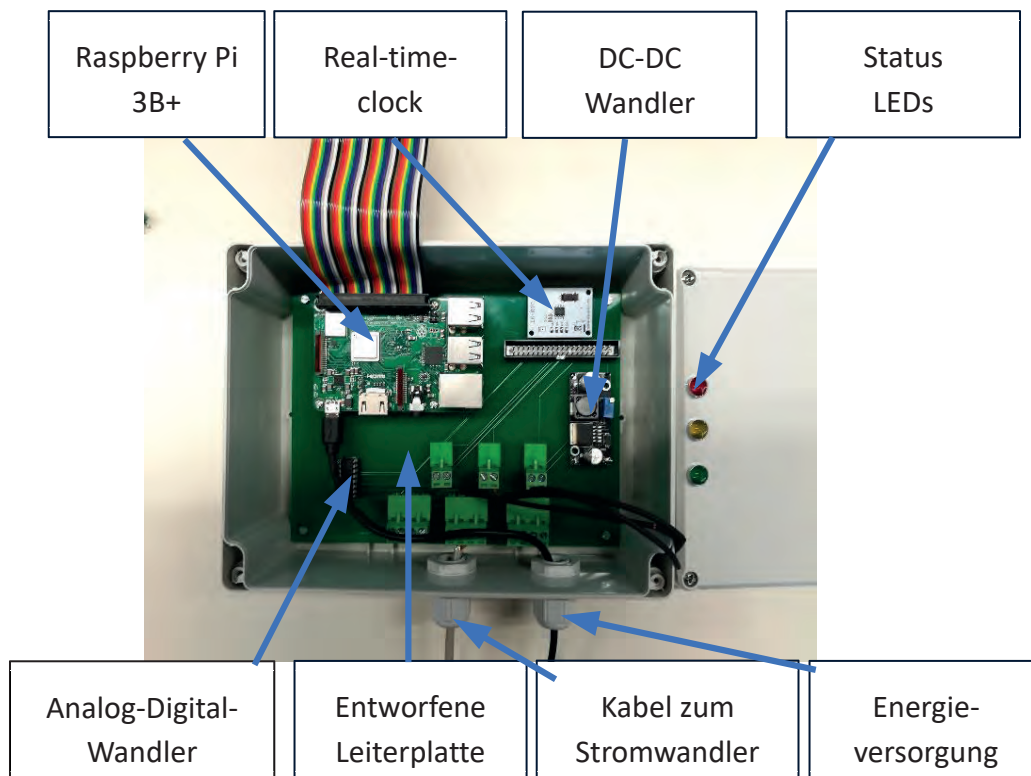


Abbildung 7 Aufbau und Hardware des Online-Monitoring Systems

Für die Auswertung der Messdaten wurde die künstliche Intelligenz in Form eines k-Means-Algorithmus für das Clustering sowie einer One-Class Support Vektor Maschine ausgelegt. Dabei erlernt die künstliche Intelligenz den normalen Zustand des untersuchten Geräts und kann in Folge ein Monitoring durchführen. Im späteren Betrieb ist es damit möglich, Anomalien in den Betriebsmodi und dem Verhalten des gemessenen Gerätes zu erkennen.

Abbildung 8 zeigt die unterschiedlichen Phasen des Algorithmus und die Verifizierung anhand von Messwerten. Weitere Details sind den Veröffentlichungen im Anhang zu entnehmen.

Nach der Entwicklung des Online Monitoring Systems wurde eine entsprechende Schnittstelle zur energybase Hardware aufgebaut. Hintergrund war hierbei die Einbindung der gemessenen und ausgewerteten Daten des Online Monitoring Systems (und des Wärmemengenzählers) in die energybase bzw. in die graphische Benutzeroberfläche (GUI) der energybase. Das Online Monitoring

System verschickt dabei nur die von der künstlichen Intelligenz ausgewerteten Daten, die gemessenen Werte verbleiben im Online Monitoring System.

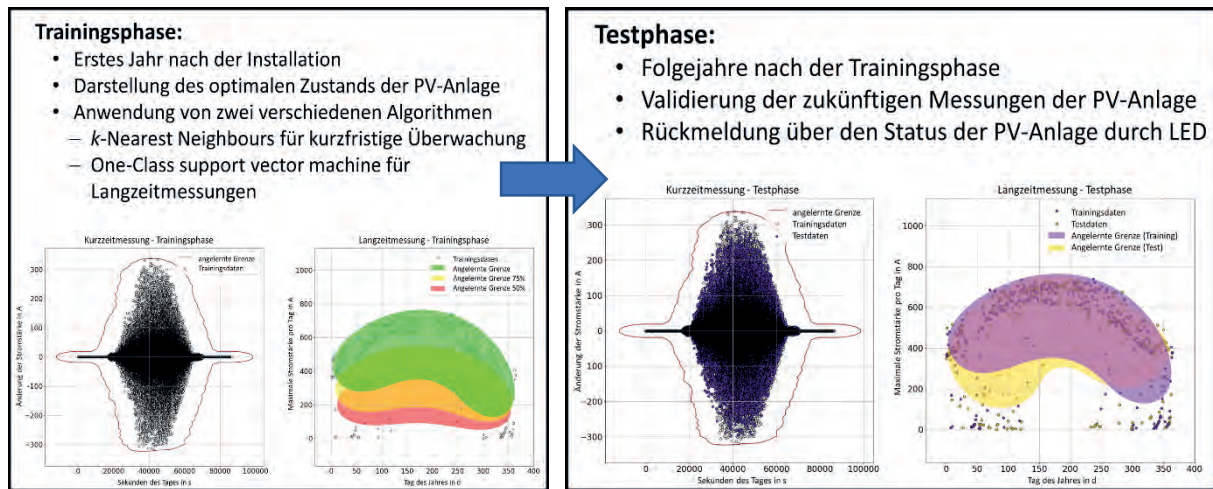


Abbildung 8: Auszug aus der Veröffentlichung zu den Methoden der künstlichen Intelligenz. Im linken Feld sind die Auswertungen einer Messung des Online Monitoring Systems in der Trainingsphase und im rechten Feld die Auswertungen aus dem Testbetrieb im Livebetrieb des Online Monitoring Systems dargestellt.

Die Parameter zur Einbindung des Online Monitoring Systems in die energybase sind wie folgt:

- Alle 10 Sekunden übermittelt der Raspberry Pi die Daten des Wärmemengenzählers und die Zustände des Online-Monitorings
- Die Daten werden über das Protokoll Modbus TCP/IP über die Ethernet-Schnittstelle gesendet
- Infos zu den Daten des Wärmemengenzählers:
 - o Verbrauchte Wärmeenergie (kWh)
 - o Speicherung der originalen Werte für 8 Tage
 - o Aggregation der Daten über einen Zeitraum von bis zu 2 Jahren
- Infos zu den Daten des Online-Monitorings:
 - o Alle 10 Sekunden erfolgt ein Update des Zustands
 - o Die Livedaten sind ausreichend, es ist keine Speicherung notwendig

Darauf aufbauend wurde eine zusätzliche Schnittstelle für die energybase entwickelt, um die Daten der externen PV-Anlagen (sowie der Wärmeübergabestationen, siehe hierzu die Ausführungen zu Arbeitspaket 3 in Kapitel 4) an die energybase zu senden.

Das Online Monitoring System kann somit auf Basis des Modbus TCP/IP Protokolls über die Ethernet-Schnittstelle in die energybase-Steuerung eingebunden werden. Die Integration des OMS in die energybase GUI erfolgt in einem ersten Schritt durch die hardwareseitige Einbindung (Verkabelung und Anschluss an Stromversorgung). Anschließend erfolgt die Einrichtung der Netzwerkverbindung über WLAN oder einen externen Hot Spot. Zuletzt wird die Kommunikation der energybase mit dem Online Monitoring System über die Vergabe und die Eingabe der IP-Adresse über die Raspberry Pi Oberfläche eingerichtet.

Nach einer erfolgreichen Labortestphase hinsichtlich der Installation und anschließenden Messaufzeichnung und Auswertung wurde das Online Monitoring System ab August 2019 bei ausgewählten Pilotkunden (an insgesamt 4 Anlagen) installiert. Ab Herbst 2019 wurde an einem weiteren Online-Monitoring-System gearbeitet, das entsprechend so angepasst werden soll, dass damit der Zustand unterschiedlicher Haushaltsgeräte überwacht werden kann. Dieses System befindet sich zum Zeitpunkt der Ausfertigung des Projektberichts in der Labortestphase und soll anschließend in weiteren Projekten der Hochschule Aalen Verwendung finden.

4. AP 3 – Smart Grid Umsetzung

Für die Umsetzung eines Smart Grids sollten der Energiebedarf und die Energieflüsse aufgenommen werden um diese später entsprechend steuern zu können. Hierfür konnte bei 20 Pilotkunden entsprechendes Messequipment installiert werden um den Netzbezug, die aus Photovoltaikanlagen gewonnene Energie, ins Netz gespeiste Energie, die tatsächlich genutzte Energie und die Wärmemenge, welche aus dem Wärmenetz entnommen wird, zu messen. Außerdem wurden an den beiden Umspannstationen von Sachsenhausen zwischen Mittel- und Niederspannungsnetz Messgeräte installiert, um auch hier Daten des Energieaustauschs aufzunehmen. Dies ermöglichte es, einen Gesamteindruck der energetischen Zusammenhänge von Sachsenhausen zu erhalten.

In den Haushalten wurden zur Messung des Energiebedarfs jeweils ein Smart Meter (im folgenden energybase genannt) der Firma energybase installiert. Die energybase misst den Stromfluss am Hausanschluss und kann Messwerte weiterer angeschlossener Sensorik wie z.B. Energiemessungen von PV-Anlagen auswerten und Haushaltsgeräte entsprechend steuern.

In jedem Haushalt konnte ein Raspberry Pi mit integrierter Datenbank implementiert werden. Dieser erhält über die REST-Schnittstelle (Representational State Transfer-Schnittstelle) der energybase die aufgenommenen Daten und legt diese zur Zwischenspeicherung in der Datenbank ab. Als Datenbank wurde SQLite verwendet. Diese ist für Rechner mit geringer Leistung wie dem Raspberry Pi ausgelegt und für diesen Einsatz geeignet, da die Daten lediglich zum Server übertragen werden sollen. Um die REST-Schnittstelle anzusprechen, wurde ein Python-Skript entwickelt, welches das Paket „requests“ [1] zur Abfrage nutzt. Mit dem Paket „persist-queue“ [2] werden die Daten in die Datenbank SQLite eingetragen. Ein weiteres Python-Skript sendet die Daten bei bestehender Internet-Verbindung zum Server. Ist eine Übertragung der Daten nicht möglich, werden diese in regelmäßigen Abständen gesendet. Sind die Daten erfolgreich übertragen, werden sie aus der Datenbank des Raspberry Pi gelöscht, um Speicherplatz freizugeben.

Auf dem Server wurde die Datenbank InfluxDB [3] implementiert. Sie wurde gewählt, da sie für Zeitreihen optimiert ist. In dieser Datenbank können Daten direkt eingetragen werden, wobei parallel ein Zeitstempel von InfluxDB erstellt wird. Alternativ hierzu können den Daten aber auch separate Zeitstempel zugewiesen werden. Mit dieser Funktion können die Daten bei einer verzögerten Übertragung, z.B. durch einen Internetausfall, dennoch korrekt übermittelt werden.

Für die 20 teilnehmenden Haushalte wurden 25 Raspberry Pis benötigt (20 zur Übertragung der Daten an den Server sowie fünf für Messungen an älteren Photovoltaikanlagen, bei welchen die Kommunikation mit den energybase-Smart-Metern nicht möglich war). Hierfür wurde jedes Gerät mit dem Raspbian-Betriebssystem bespielt, welches auf Linux basiert. Die Installation des Betriebssystems befindet sich jeweils auf einer microSD Karte, welche auch gleichzeitig den internen Speicher des Raspberry Pi darstellt. Um den Installationsprozess der Betriebssysteme zu beschleunigen, wurden Bash Skripte erstellt, die diesen vereinfachen. Weiterhin können mit solch einem Skript beim ersten Start die Passwörter automatisch gesetzt und alle nötigen Einstellungen und Software installiert werden.

Das Aufsetzen eines Raspberry Pi erfolgt nach den folgenden Schritten:

1. Raspbian auf der jeweiligen SD-Karte installieren.
2. Passwort des Raspberry Pi ändern.
3. Dateien des Ordners „Dateien für Raspberry Pi“ in den Ordner /home/pi/ verschieben, siehe Anhang.

Im Skript test.py muss zwingend die

- domain (IP-Adresse der energybase),

- die userID (Pilotkundennummer) sowie,
 - das Passwort geändert werden.
 - In den anderen Skripten muss die IP-Adresse der domain geändert werden.
4. Anydesk installieren und die Anydesk-ID notieren.
 5. Damit die Skripte funktionieren, muss SQLite installiert und die Service-Dateien in den richtigen Ordner bewegt sowie aktiviert werden.

Die Codes in Tabelle 3 müssen für das Installieren, die Aktivierung und zur Prüfung der Funktionalität des Raspberry Pi auf dem Gerät selbst ausgeführt werden.

Tabelle 3 Codes zur Installation, Aktivierung und Prüfung der Funktionalität des Raspberry Pi

Codes in Kommandozeile	Funktion
<code>sudo pip3 install persist-queue</code>	SQLite installieren
<code>sudo apt-get install python3-pandas</code>	Python3 Panda installieren
<code>sudo pip3 install mcp3208</code>	Implementiert Bibliothek für A/D-Wandler
<code>sudo mv energybase-influx.service /etc/systemd/system/</code> <code>sudo mv wmz.service /etc/systemd/system/</code> <code>sudo mv pv.service /etc/systemd/system/</code> <code>sudo mv wp.service /etc/systemd/system/</code>	Den benötigten Service (Energybase-influx, wmz, pv oder wp) in Ordner „system“ bewegen; für die Systemctl-Services, damit Skripte automatisch beim Restart gestartet werden / müssen im nächsten Schritt noch aktiviert werden
<code>sudo systemctl enable energybase-influx</code>	Aktiviert das Skript test.py
<code>sudo systemctl enable wmz</code>	Aktiviert das Skript Waerme.py
<code>sudo systemctl enable pv</code>	Aktiviert das Skript Strommessung.py
<code>sudo systemctl enable wp</code>	Aktiviert das Skript Strommessung.py
<code>systemctl status energybase-influx.service</code>	Zum Prüfen, ob der Service gestartet wurde.
<code>sudo systemctl restart energybase-influx</code>	Startet den Service neu
<code>sudo systemctl start energybase-influx</code>	startet den Service
<code>sudo systemctl stop energybase-influx</code>	stoppt den Service
Zum Debuggen <code>journalctl -u energybase-influx</code> <code>journalctl -u wmz / sudo python Waerme.py</code> <code>journalctl -u pv / sudo python3 Strommessung.py</code> <code>journalctl -u wp / sudo python3 Strommessung.py</code>	Logeintrag von test.py Logeintrag von Waerme.py Logeintrag von Strommessung.py Logeintrag von Strommessung.py

Die Raspberry Pis sind in den Haushalten an Routern ohne offene Ports (Weiterleitungen) angeschlossen. Dennoch wird eine Möglichkeit benötigt um auf die Systeme später per Fernzugriff

zugreifen zu können. Das Aufsetzen eines SSH (Secure Shell) Servers, ist eine Möglichkeit sich mit allen Teilnehmern zu verbinden. SSH erlaubt das Erstellen von Tunneln und damit Zugriff auf jeden Service im Netzwerk. Allerdings verzögerte sich die Bereitstellung des Servers über den Zeitpunkt, an welchem die Installation der Geräte in den Haushalten geplant war. Der Vorschlag, eine virtuelle Maschine von einem Cloud Anbieter für den Übergang zu nutzen, wurde abgelehnt.

Als Alternative wurde die Software Anydesk (ein Remote Desktop Service) eingesetzt. Anydesk bietet die Möglichkeit, Clients über die Anydesk Server zu vernetzen. Die Verbindung funktioniert hier auch hinter der NAT (Netzwerkadressübersetzung), da die Verbindung von innen heraus aufgebaut wird.

Ursprünglich war geplant, die Debian Server Version von Anydesk zu nutzen. Dies war leider nicht möglich, da sich im Test zeigte, dass sich Anydesk nicht ohne eine grafische Oberfläche (X-Server) startet. Aus diesem Grund wurde auf das über zwei Gigabyte große Desktop Abbild von Anydesk gewechselt und die Skripte entsprechend angepasst.

Für den Fernzugriff wird für jedes Gerät eine Anydesk ID (eine neunstellige Zahl) benötigt. Diese wurde nach dem ersten Start der Raspberry Pis für jeden Haushalt anonym in einer Excel Tabelle vermerkt, damit eine Zuordnung von Hardware, Software und der Haushalte gegeben ist.

Auslesen der energybase-Daten

Für das Auslesen der energybase Daten mithilfe des Raspberry Pi wurde ein Python Skript namens energybase.py geschrieben, siehe Anhang. Dieses Skript wird auf den jeweiligen Rasperry Pis ausgeführt. Es verwendet die sogenannte JSESSIONID mittels des Passwortes der energybase. Im Anschluss wird die JSESSIONID zurück zur energybase gesendet, woraufhin eine Subscription ID von der energybase gesandt wird. Aus der JSESSIONID und der Subscription ID können HTTP Long Polling Anfragen erstellt werden. Das Long Polling sorgt dafür, dass die Daten der energybase zusätzlich immer an den Raspberry Pi geschickt werden. Die Daten werden in einer SQLite Datenbank zwischengespeichert und daraufhin weiter an den Server geleitet. Somit ist gewährleistet, dass bei einem Internetausfall keine Daten verloren gehen. Über einen System Service wird dieses Skript auch bei einem Neustart des Raspberry Pi oder einem Absturz des Skriptes wieder automatisch gestartet.

Testaufbau mit energybase und Raspberry Pi

Für die Inbetriebnahme der energybase wurde ein Testaufbau inklusive einem Raspberry Pi in Betrieb genommen. Im Testaufbau wurde in einem ersten Schritt das Skript für die Datenübertragung von der energybase zum Raspberry Pi geschrieben und verbessert. Anschließend wurde am Testaufbau die Datenübertragung von der energybase über das Rasperry Pi zum Server realisiert.

Für die Datenübertragung wurde eine Warteschlange programmiert, damit bei einem Ausfall des Internets keine Daten verloren gehen. Das Skript schreibt hierbei die Daten in eine lokale SQLite Datenbank, falls der Server nicht verfügbar ist. Bei allen weiteren Sendeversuchen werden diese Daten aus der Datenbank ebenfalls gesendet. War das Senden der Daten erfolgreich, werden diese wieder aus der lokalen Datenbank gelöscht.

Nachdem im Testaufbau alle Komponenten über einen Zeitraum von einer Woche funktioniert haben, und der Raspberry Pi alle Daten der energybase erhält und an den Server weitergeben konnte, wurden die entwickelten Skripte auf alle weiteren Raspberry Pis gespielt.

Einbindung der Wärmeübergabestationen

Um die Sektorkopplung abzubilden, wurden die Wärmeübergabestationen mithilfe der Rasperry Pis eingebunden. Durch einen Defekt in der Datenkommunikation auf Seiten der Wärmeübergabestationen, wurden nur neun von 13 Wärmeübergabestationen eingebunden. Da die

Behebung des Defekts den zeitlichen und finanziellen Rahmen des Arbeitspakets unverhältnismäßig belastet hätte, wurde nach Rücksprache mit den Projektpartnern darauf verzichtet. Bei dieser Einschätzung war außerdem ausschlaggebend, dass bereits neun Wärmeübergabestationen Daten lieferten, so dass auf die vier fehlenden Stationen verzichtet werden konnte.

Messungen an den Umspannstationen

Messungen an den beiden Umspannstationen in Sachsenhausen sollten zusätzliche Daten über Verbrauch und Einspeisung zwischen Mittel- und Niederspannungsnetz liefern. Hierzu wurden Rogowski-Spulen an den Transformator-Kabeln installiert, siehe Abbildung 9 und mit Janitza UMG 96 RM-E Messgeräten verknüpft, siehe Abbildung 10. Die aufgenommenen Daten der installierten Messgeräte sollten mittels der Janitza GridVis Software an den Server gesendet werden. Diese können über Ethernet von der Hersteller Software angesprochen werden. Die Umspannstationen haben allerdings keine kabelgebundene Internetanbindung, daher war ein Router erforderlich, welcher über Mobilfunk eine Verbindung herstellt. Für die Mobilfunkanbindung wurde je eine SIM-Karte eingesetzt. Diese sind eigentlich nicht für diesen Zweck ausgelegt, da hier keine eigenen IPv4 Adressen vergeben werden. Stattdessen macht der Provider Carrier-grade NAT, um IP-Adressen einzusparen. Dementsprechend kann keine direkte Verbindung zu dem Gerät von außen aufgebaut werden. Aus diesem Grund wurde entschieden, eine Verbindung per OpenVPN zwischen SIM-Karte und Server aufzubauen. Dies war zum Zeitpunkt der Umsetzung noch nicht möglich, da dieser noch nicht Verfügung stand. Dies hatte unmittelbar zur Folge, dass der Meilenstein „Durchführung der Messungen mit direkter Auswertung“ stromseitig erst ab September/Okttober 2019 erfolgen konnte. Aus diesem Grund wurde vorübergehend ein Google Cloud Server erstellt, damit diese Verbindung getestet werden konnte. Google bietet ein OpenVPN Access Server Image an, welches für eine vereinfachte Installation genutzt wurde.



Abbildung 9: Installierte Rogowski-Spulen (siehe grüne Pfeile) zur Strommessung am Transformator in der Umspannstation Brühlstraße in Sachsenhausen

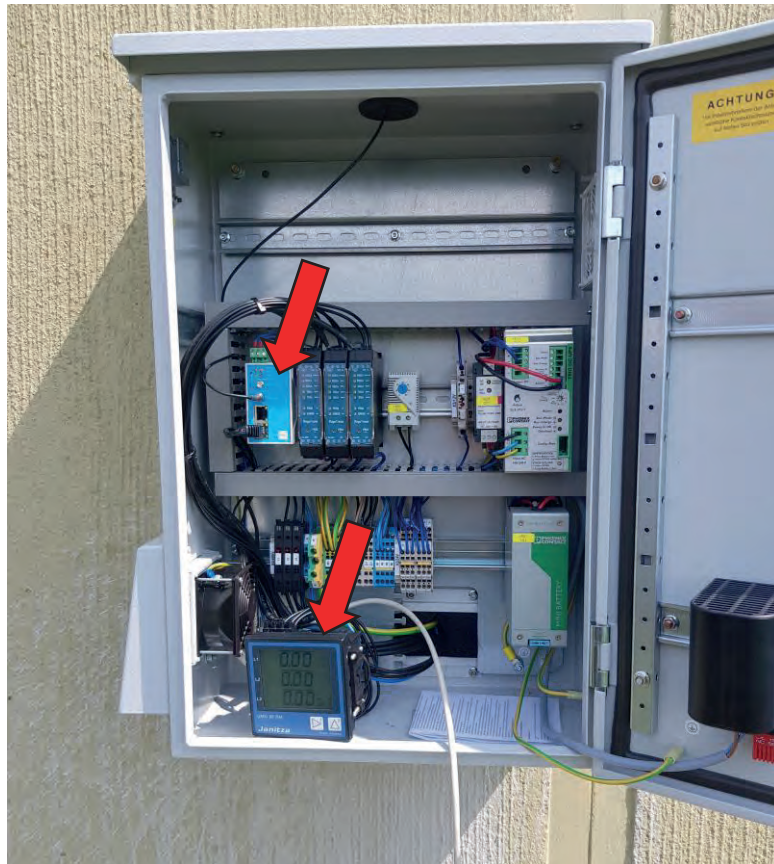


Abbildung 10: Installierte Messgeräte (siehe rote Pfeile) im Schaltkasten der Umspannstation Brühlstraße in Sachsenhausen: Janitza UMG 96 RM-E Messgerät (links unten) und INSYS icom Router EBW-L100 (links oben)

Da in der Anwendung ein Linux Server betrieben wurde, wurde auch für GridVis die Linux Variante verwendet, auch wenn hier kein Support geliefert werden kann. Dies führte folglich zu Problemen bei der Installation. Dabei werden bei der Datenübertragung alle Schritte korrekt ausgeführt, allerdings werden dabei automatisch alle zuvor installierten Skripte wieder entfernt. Als Ergebnis erscheint auf der Konsole die Meldung, dass der Service nicht gestartet werden konnte.

Aufsetzen des Servers

Der Server wurde mit Ubuntu Server 18.04 LTS vom Serverprovider aufgesetzt. Anfangs waren alle Ports (außer 22 - SSH) in der Firewall geschlossen und selbst ausgehende Verbindungen blockiert. Aus diesem Grund war es nicht möglich, Updates oder neue Pakete aus dem Ubuntu Repository zu installieren. Stattdessen wurden die InfluxDB-Datenbank und Grafana zur Visualisierung als .deb Paket manuell heruntergeladen und mit scp übertragen. Um Daten auf dem Server sinnvoll zu speichern, wurde die InfluxDB-Datenbank installiert und in Betrieb genommen. Um die Daten von externen Geräten zu visualisieren, wurde das Open Source Dashboard Grafana installiert und eingerichtet.

Nachdem der Serverprovider die benötigten Ports für die Serverdienste und ausgehende Verbindungen freigeschaltet hatte, konnte der HTTP Server konfiguriert werden. Um die Sicherheit des Servers zu erhöhen, wurde nginx als Reverse Proxy installiert [4]. Dieser terminiert die TLS Verbindung. Hiermit muss der InfluxDB und Grafana Service nur noch normales HTTP verarbeiten. Dies hat den Nutzen, dass beide Services nicht direkt im Internet sichtbar sind und somit die Sicherheit des Systems erhöht wird. Der Proxy leitet von der InfluxDB API nur den /write Endpunkt weiter. Damit können nur Daten auf eine vorhandene Datenbank geschrieben werden. Dies verhindert, dass Personen mit Zugriff auf einen Raspberry Pi die Daten von anderen Kunden auslesen könnte. Mit dieser Lösung sind selbst die eigenen Daten aus der Vergangenheit sicher (nicht auslesbar). Nur über Grafana (oder direkten

Datenbankzugriff) können sie gelesen werden. Insgesamt sind folgende Ports (eingehend) geöffnet: 443 (HTTPS, Grafana), 8086 (InfluxDB API), 943 (VPN Administrator webinterface) und 1194 (VPN). Um eine sichere HTTPS-Verbindung zu erhalten, muss der Server von einer offiziellen Zertifikatsautorität (engl. Certificate Authority oder kurz CA) zertifiziert werden. Dafür wurde auf dem Server ein Service namens certbot installiert [5], der regelmäßig den Server von Let's Encrypt [6], zertifizieren lässt. Für den Reverse-Proxy wird nginx [4] genutzt. In Abbildung 11 ist der komplette Aufbau der Kommunikationsinfrastruktur zwischen den einzelnen Teilnehmern und dem Server visualisiert.

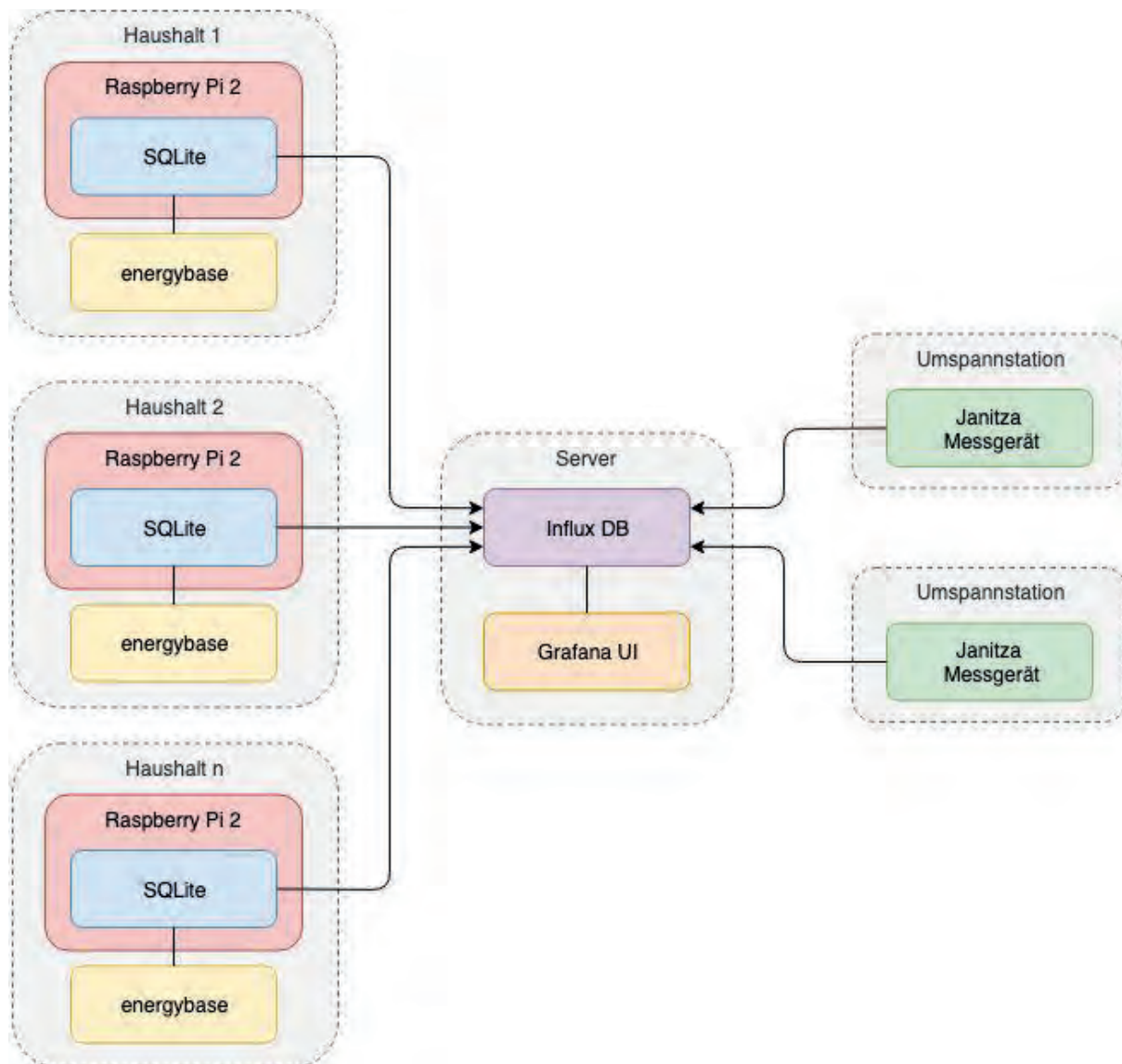


Abbildung 11 Aufbau der Kommunikationsinfrastruktur zwischen dem Server und den energybase Smart Metern sowie den Janitza-Messgeräten. Von Seiten der Haushalte erfolgt die Datenübertragung von der energybase mittels der Raspberry Pis. Für die Übertragung der Daten der Umspannstationen wurden SIM-Karten und ein VPN-Tunnel zur Übertragung eingesetzt.

Visualisierung

Die Daten der Pilotkunden wurden in einer Influx-Datenbank auf dem Server gespeichert. Für die Visualisierung der aufgenommenen Daten bietet sich die plattformübergreifende Open-Source-Anwendung zur grafischen Darstellung von Daten Grafana [7] an, da es InfluxDB offiziell unterstützt und eine benutzerfreundliche Oberfläche hat. Es können Nutzer angelegt werden, welche die visualisierten Daten ansehen und als CSV exportieren können. Eine weitere nützliche Funktion von

Grafana ist, dass auch verschiedene Datensätze aus frei wählbaren Zeiträumen überlagert werden können, wie beispielsweise die letzten zwei Wochen eines Haushalts und die letzten zwei Wochen einer Umspannstation. Abbildung 12 zeigt beispielhaft die auf dem Server gespeicherten Daten der Energieflüsse von Haushalt 2 bis Haushalt 4 am 20.06.2021. Dabei wird je nach Haushalt und installierten Geräten der Netzbezug, die PV-Erzeugung, die Netzeinspeisung sowie der Wärmebezug an der Wärmeübergabestation zum Nahwärmenetz angezeigt (die Nummerierung der Haushalte korrespondiert dabei nicht mit der laufenden Nummerierung aus Tabelle 2).

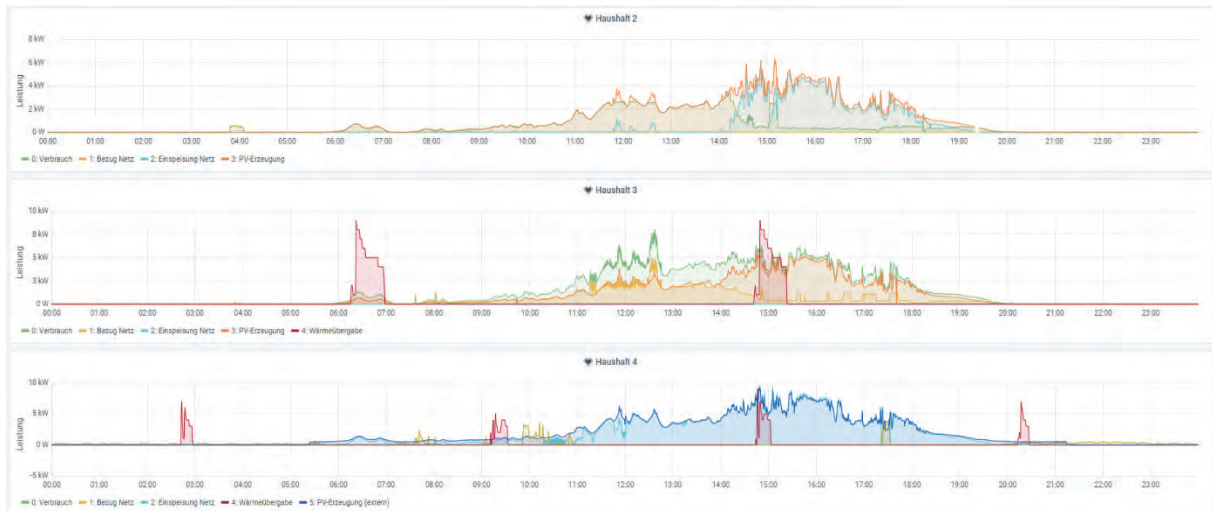


Abbildung 12 Darstellung der Energieflüsse am 20.06.2021 von 3 Haushalten basierend auf den auf dem Server gespeicherten Daten

Messfehler und Entwicklung einer Methode zum Schließen von Datenlücken durch Nutzung einer KI

Bei der Auswertung der Daten wurde festgestellt, dass diese teilweise lückenhaft auflaufen. Für diese Lücken gibt es unterschiedliche Ursachen. Teilweise führt der Ausfall des Netzwerks in den Haushalten dazu, dass eine Datenübermittlung nicht möglich ist. Weiterhin fallen zeitweise die Raspberry Pis kurzzeitig aus. Hierfür wurde eine Routine entwickelt, welche in diesem Fall eine Mail sendet. So kann per Fernzugriff das betroffene Raspberry Pi neu gestartet werden. In weiteren Fällen fielen die Raspberry Pis ganz aus und mussten durch neue Geräte ersetzt werden. Auch der Ausfall von SD-Karten in den Rasperry Pis führte zu lückenhaften Messreihen. Eine solche Datenlücke wird beispielhaft in Abbildung 13 dargestellt. Dabei variiert die Zeit einer solchen Lücke zwischen einigen Sekunden bis zu einigen Wochen.

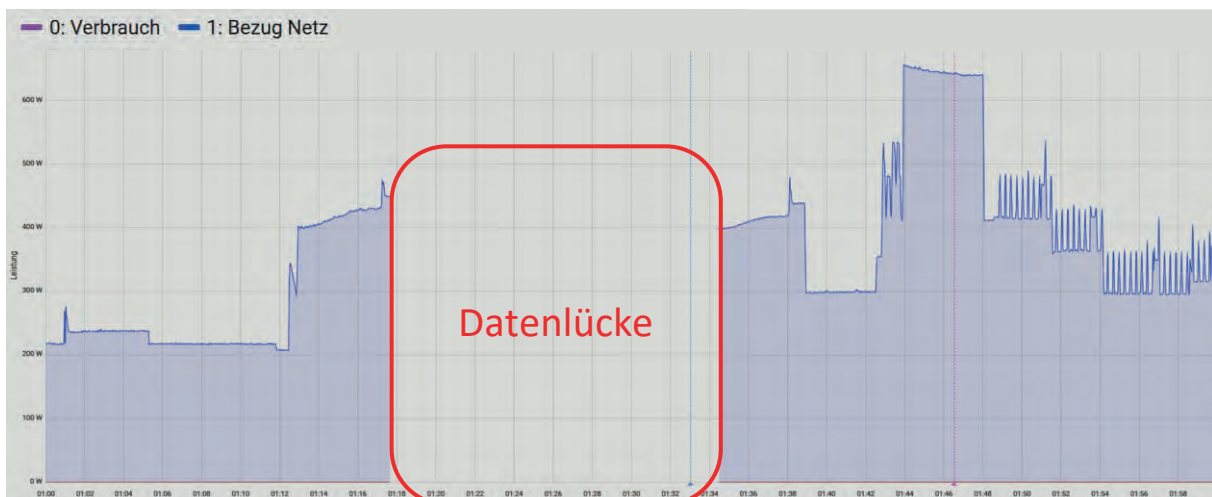


Abbildung 13 Darstellung des Netzbezugs eines Haushalts mit auftretender Datenlücke zwischen 01:17 Uhr und 01:34 Uhr.

Die Steuerung eines Netzspeichers sollte auf Basis der Pilotkundendaten erfolgen. Sind Überschüsse in den Haushalten der Pilotkunden zu verzeichnen, kann der Speicher Energie aus dem Netz von Sachsenhausen aufnehmen (wirkt als Verbraucher). Wird im Haushalt weniger Energie erzeugt als benötigt, kann der Speicher Energie abgeben und entsprechend diesen Bedarf decken (wirkt als Erzeuger). Wird in einem Haushalt ein Bedarf erkannt und in einem anderen Haushalt ein Überschuss, ist es sinnvoll, überschüssige Energie des einen Haushaltes dem Haushalt mit einem Bedarf zuzuführen.

Um solch eine Regelung umzusetzen, werden aktuelle Daten und eine echtzeitfähige Steuerung benötigt. Dies ist durch die aktuell auftretenden Datenlücken in den Aufzeichnungen nicht möglich. Weiterhin werden für Simulationen des Netzes von Sachsenhausen zusammenhängende Datensätze benötigt.

Um den Datenlücken entgegenzuwirken, wurde im ersten Schritt eine Routine entwickelt, die im Fehlerfall eine Mail versendet. Somit kann der betroffene Raspberry Pi zeitnah aus der Ferne neu gestartet werden. In den meisten Fällen konnte ein Neustart der Systeme den Fehler beheben. Allerdings gab es auch Fälle, in denen ein Neustart nicht die erneute Aufzeichnung von Daten starten konnte. In diesen Fällen wurden die Pilotkunden zum manuellen Neustart der Raspberry Pis oder zur Prüfung der internen Internetanbindung aufgefordert. Unter anderem führten aber auch der Ausfall von SD-Karten in den Raspberry Pis zu unvollständigen Messreihen. In diesen Fällen mussten neue SD-Karten mit neuen Betriebssystemen und der Software neu aufgesetzt und im jeweiligen Haushalt eingebracht werden. Weiterhin wurden in anderen Fällen die Raspberry Pis durch neue Geräte ersetzt. Die Länge einer auftretenden Datenlücke variiert je nach Art des Ausfalls zwischen Zeiträumen weniger Sekunden bis hin zu mehreren Wochen. Um dieses Problem zu lösen und lückenlose Datensätze zu erhalten, wurden verschiedene Interpolationslücken-Methoden mit verschiedenen Datensätzen getestet und die Ergebnisse ausgewertet.

Viele klassische Lösungen, die auf mathematischen und statistischen Methoden basieren, wie z. B. Mittelwertbildung, Verwendung des Modalwerts, lineare oder saisonale Interpolation, werden in Studien zum Füllen der Datenlücken verwendet [8]. Zusätzlich erscheinen Verfahren basierend auf künstlicher Intelligenz und Machine Learning vielversprechend, um eine zuverlässige und genaue Vorhersage großer Datenlücken zu erreichen [9]. Für das Füllen von Datenlücken werden üblicherweise verschiedene Methoden des maschinellen Lernens verwendet. Basierend auf Zeitreihenanalysen wie z.B. des Exponential Smoothing Forecast Algorithm (kurz: ES) oder dem Auto-Regressive Integrated Moving Average - Verfahren (kurz: ARIMA) ist die auf linearer Regression basierende Datenvorhersagemethode sehr genau [10] [11]. Auch Deep Learning basierte Methoden wie das Long-Short Term Memory - Verfahren (kurz: LSTM) können vielversprechende Lösungen für Vorhersageaufgaben liefern [12].

Hauptfaktoren für die Wahl einer bestimmten Methode zum Füllen fehlender Messwerte sind die Art der vorliegenden Daten. In diesem Fall beinhalten die verlorenen Daten sowohl Haushaltslastdaten als auch Daten von Photovoltaikanlagen, welche völlig unterschiedliche Muster haben. Es wird daher ein Algorithmus benötigt, welcher in beiden Datensätzen, eine Vorhersageleistung mit sehr hoher Genauigkeit und akzeptabler Laufzeit hat.

Um die beste Methode zum Schließen der Datenlücken wählen zu können, wurden unterschiedliche Ansätze gegenübergestellt und deren Genauigkeit miteinander verglichen. Die verglichenen Methoden sind ARIMA [13], ES, Lineare Regression [8] und LSTM, siehe Tabelle 4. Schließlich wurde aufgrund von Defiziten der einzelnen Methoden, wie einer geringen Genauigkeit, schlechte Erkennung auftretender Muster in Datensätzen oder langer Laufzeiten bei großen Datenlücken, eine umfassendere Methode

implementiert. In Tabelle 4 sind die Ergebnisse für die Genauigkeit der ausgewählten Methoden aufgeführt.

Zum Testen und Vergleichen der ausgewählten Methoden wurden Last- und Photovoltaikdaten eines Haushalts aus dem Jahr 2020 mit einem Zeitintervall von einer Minute verwendet. Zum Testen der Methoden wurden bewusst Lücken in die Datenreihen eingebaut. Durch den Vergleich der berechneten Werte mit den gemessenen, konnte die Genauigkeit mithilfe der mittleren quadratischen Abweichung (root mean square error - RMSE) und dem Determinationskoeffizienten (coefficient of determination - R^2) bestimmt werden, siehe Gleichungen (1) bis (4). Hierbei beziehen sich A_i und F_i auf den tatsächlichen und den vorhergesagten Wert der i -ten Daten. N stellt die Größe der Daten dar. \bar{A} bildet den Durchschnitt der tatsächlichen Daten. SSR steht für die Summenquadratregression und SST für die Gesamtsumme der Quadrate [14].

$$RMSE = \sqrt{\left(\frac{1}{N}\right) \sum_{i=1}^N (A_i - F_i)^2} \quad (1)$$

$$R^2 = 1 - \frac{SSR}{SST} \quad (2)$$

$$SSR = \sum_{i=1}^N (A_i - F_i)^2 \quad (3)$$

$$SST = \sum_{i=1}^N (A_i - \bar{A})^2 \quad (4)$$

Eine weitere Methode für das Füllen von Lücken ist eine Kombination aus linearer Interpolation und linearer Regression. Hierfür wurde eine hybride Methode auf Basis von Deep Learning entwickelt, welche ein LSTM-Netzwerk parallel zu einem Convolutional Neural Network (CNN oder ConvNet) verwendet, welches PLCNet genannt wird. Für das Training des neuronalen Netzwerks wurde ein vollständiger Datensatz ohne Lücken verwendet. Der Code des duplizierten Algorithmus ist im Anhang enthalten. Abbildung 16 zeigt den Aufbau des neuronalen Netzwerks PLCNet mit den implementierten Methoden LSTM und CNN [15]. Die Ersatzwertbildung von zwei auftretenden Datenlücken wird in Abbildung 15 anhand einer Photovoltaikkurve dargestellt.

Tabelle 4 Genauigkeit der verschiedenen Methoden ARIMA, ES und LSTM

Methode	Kleine Lücken (≤ 30 min)	Große Lücken (> 30 min)
ARIMA	97%	49%
ES	95,3%	53%
LSTM	91,1%	69%
Lineare Regression	95,2%	72,3%
PLCNet	95,5%	91,3%

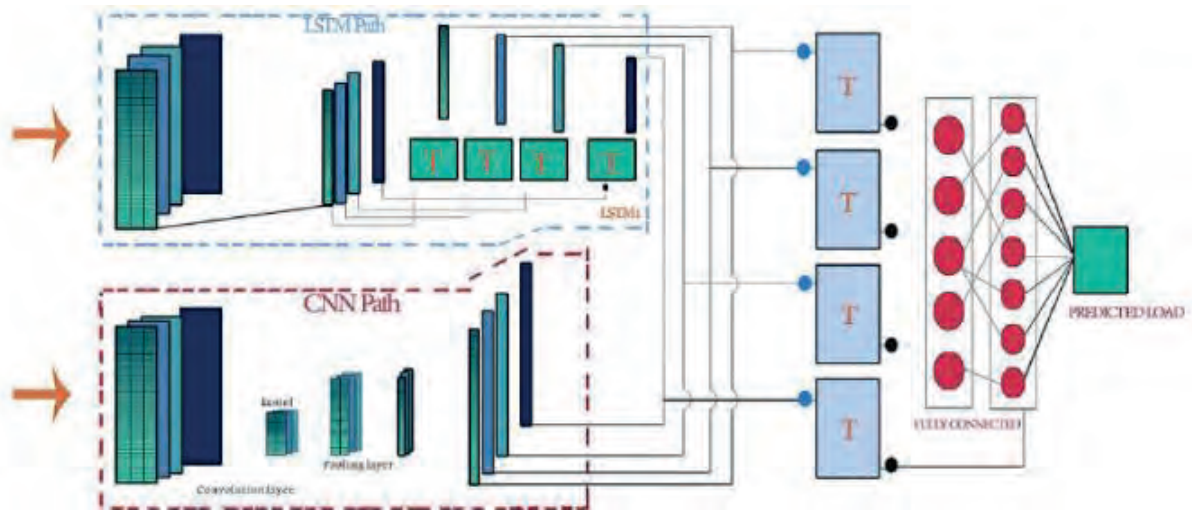


Abbildung 14 Framework von LSTM-CNN-Netzwerk (PLCNet) [15]

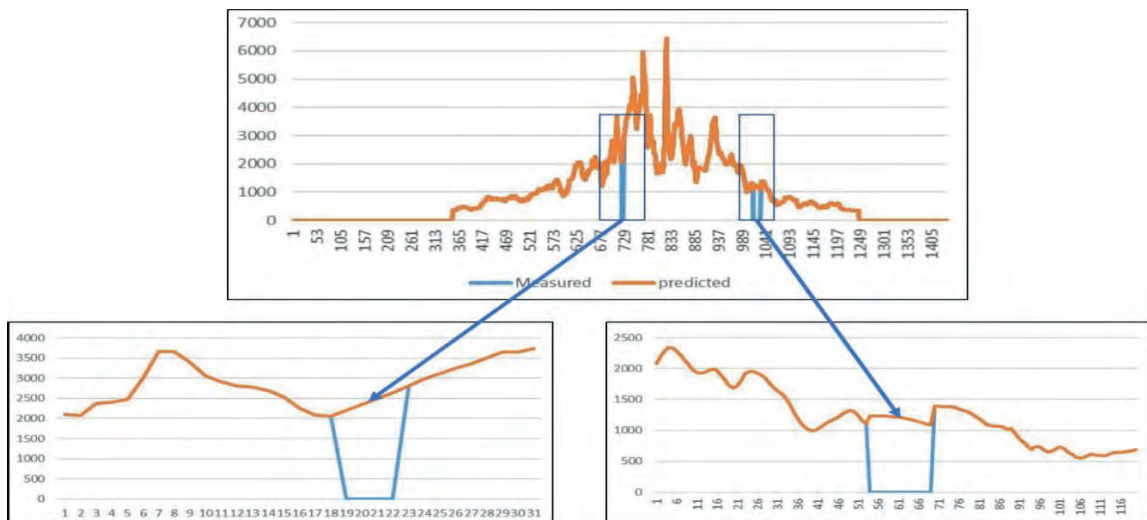


Abbildung 15 Ersatzwertbildung für fehlende Daten einer Photovoltaik-Kurve. Im Gesamtverlauf liegen die blaue und orangefarbene Linie übereinander. Nur bei fehlenden Messwerten wird die blaue Kurve sichtbar.

5. AP 4 – Geschäftsmodelle entwickeln und testen

In diesem Kapitel wird eine kurze Übersicht zu den beiden im Wesentlichen verwendeten Methoden zur Darstellung von Geschäftsmodellen in dieser Arbeit gegeben. Der BMC (Business Model Canvas) ist zur Entwicklung eines Geschäftsmodelles aus einer zentralen Sicht eines Unternehmens eine weltweit anerkannte Methode und wird in einer Vielzahl von Innovationsmethoden zur Darstellung von Geschäftsmodellen verwendet. Der am Herman-Hollerith-Zentrum (HHZ) entwickelte Geschäftsmodell-Konfigurator (BMConfig) basiert auf der e3-Value-Ontologie von Gordijn [16]. Diese wird zur Modellierung und Simulation und zur Validierung der ökonomischen Tragfähigkeit der Geschäftsmodelle verwendet.

5.1. Business Model Canvas

Der Business Model Canvas (BMC) [17] ist ein Modell zur Visualisierung und Strukturierung von Geschäftsmodellen. Dieser ist in 9 Building Blocks unterteilt. Im Kern steht dabei das Wertversprechen (engl. Value Proposition). Dieses beschreibt ein Produkt oder eine Dienstleistung, die für das Kundensegment einen Mehrwert bietet. Die rechte Seite des BMC visualisiert, wie die Werte für Kunden und das Unternehmen geschaffen werden. Auf der linken Seite des BMC werden Mittel dargestellt, welche ein Unternehmen für eine erfolgreiche Umsetzung des Wertversprechens benötigt. In Abbildung 16 ist der BMC grafisch dargestellt.

- Das Wertversprechen (engl. Value Proposition) beschreibt ein „Paket von Produkten und Dienstleistungen“, die für das Kundensegment einen gewissen Wert haben. Ein Wertangebot adressiert Probleme oder Bedürfnisse der Kunden. Es bietet dem Kunden einen „Zusammenschluss oder Paket“ an Nutzen an.
- Der Building Block Kundensegment (engl. Customer Segment) visualisiert die wichtigsten Kundengruppen, welche an den Wertversprechen interessiert sind bzw. die das Unternehmen erreichen möchte.
- Das Feld Kanäle (engl. Channels) beschreibt, wie die Kunden erreicht werden sollen.
- Die Kundenbeziehungen (engl. Customer Relationships) beschreibt, welche Art von Kundenbeziehung das Unternehmen pflegt. Als Beispiel pflegt das Unternehmen einen persönlichen Kontakt oder einen digitalen Kontakt z.B. durch ein Kundenportal.
- Der Building Block Einnahmequellen (engl. Revenue Stream) kann in zwei Einnahmenquellen unterschieden werden. Zum einen die einmaligen Kundenzahlungen und zum anderen die fortlaufenden Zahlungen.
- Das Feld Schlüsselressourcen (engl. Key Resources) beschreibt die essenziellen Wirtschaftsgüter, welche notwendig sind, damit ein Geschäftsmodell funktionieren kann. Zu den Ressourcen zählen physische, menschliche oder auch finanzielle Güter.
- Der nächste Baustein Schlüsselaktivitäten (engl. Key Activities) beschäftigt sich mit den fundamentalen Handlungen, welche vom Unternehmen erfüllt werden müssen.
- Der Block Schlüsselpartnerschaften (engl. Key Partners) zeigt die wichtigsten Unternehmen und Organisationen, mit denen eine Kooperation eingegangen wird.
- Das letzte Feld sind die Kostenstrukturen (engl. Cost Structure). Dabei werden alle Kosten aufgelistet, welche benötigt werden, damit das Geschäftsmodell ausgeführt werden kann. Hierbei müssen auch variable und fixe Kosten einfließen.

Key Partners	Key Activities	Value Propositions	Customer Relationships	Customer Segments
	Key Resources		Channels	
Cost Structure		Revenue Streams		

Abbildung 16: Business Model Canvas als Basis für die Entwicklung der Geschäftsmodelle im Projekt SMASA

5.2. e3-Value

Die e3value-Ontologie ermöglicht die Modellierung von Wertschöpfungsnetzwerken mit dem Ziel, ein gemeinsames Verständnis einer Geschäftsidee zu vermitteln. Diese Geschäftsidee wird von einem Netzwerk von Akteuren realisiert, die im Kontext von organisationsübergreifenden Geschäftsmodellen gemeinsam Werte schaffen, verteilen und konsumieren. Wertschöpfungsnetze setzen sich aus ergänzenden Akteuren, den absolvierten Aktivitäten und eingebrachten technischen, organisatorischen und sozialen Ressourcen zusammen, die miteinander in Verbindung stehen, um einen ökonomischen Mehrwert für einen (End-)Kunden in Form von Produkten und/oder Dienstleistungen zu schaffen. Der Ansatz des Business Ecosystem betrachtet im Gegensatz zu den Wertschöpfungsnetzen auch Akteure, die nicht direkt in die Wertschöpfung involviert sind. Ein Beispiel für solche Akteure sind Wettbewerber, Anbieter von komplementären Produkten, Finanzdienstleister oder Regulierungsbehörden. Entsprechend können sich Business Ecosystems auch über Branchengrenzen hinweg entwickeln. Der Ansatz des Business Ecosystem wird daher für die Darstellung von Geschäftsmodellen in der Energiewirtschaft als passender erachtet und sollte daher auch in diesem Vorhaben Anwendung finden.

In Abbildung 17 sind die einzelnen Konzepte von e3Value dargestellt. Dabei werden folgende Grundelemente unterschieden: Actor, Market Segment, Value Activity, Value Interface, Value Ports, Value Exchange und Value Object. Die Grundelemente von e3Value sind wie folgt definiert.

- Der Akteur (engl. Actor) wird als eine unabhängige wirtschaftliche Entität definiert.
- Eine Market-Segment Komponente ist die Generalisierung eines Actors und fasst mehrere Actors mit ähnlichen Eigenschaften zusammen.

- Value Activity ist eine operative Aktivität, deren Durchführung zur Erzeugung eines Werts dient und soll Transparenz über Vorgänge innerhalb eines Akteurs verdeutlichen.
- Value Ports ermöglichen das Anbieten bzw. Konsumieren von Wertobjekten.
- Value Interface fasst den Austausch von Werten zusammen. Es beinhaltet zu jedem Angebot von Wertobjekten an die Umgebung auch einen adäquaten Empfang, sodass ein Handel zwischen Akteuren entsteht. Die Value Ports sind ein Teil des Value Interfaces.
- Value Exchanges verbinden zwei Value Ports zum Transfer von Werten zwischen zwei Akteuren.
- Value Objects verdeutlichen, welcher Wert über die Value Exchanges ausgetauscht werden. Diese sind beispielsweise Güter, Services oder Geld, wobei ein Value Object für einen oder mehrere Actors einen Wert darstellen muss.
- Über Szenarien sollen verschiedene Business Cases des Geschäftsmodelles berechnet werden. Dazu werden über Connection Elemente Value Interfaces verknüpft. Ein Startstimulus kennzeichnet den Beginn eines Szenarios. Das Ende wird durch einen zugehörigen Stoppstimulus gekennzeichnet.
- AND/OR ermöglichen das Abbilden von logischen Operationen für ein Szenario.

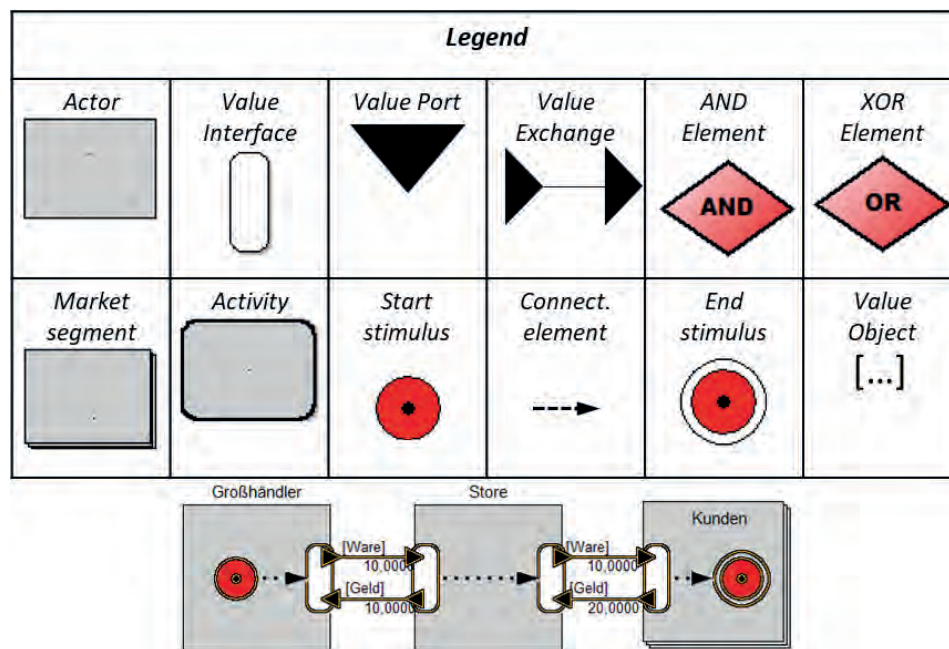


Abbildung 17: Grundelemente des e3 Values Modells

5.3. Methodik

Vor dem Hintergrund der Energiewende, der Digitalisierung und der dynamischen Märkte müssen Unternehmen in der Energiedomäne laufend neue Geschäftsmodelle entwickeln und diese an neue Marktbedingungen adaptieren. Die Entwicklung von Geschäftsmodellen findet oft im Rahmen von Innovationsworkshops statt. Hierbei kommen Methoden wie Design Thinking oder Value Proposition Design von A. Osterwalder et al. [17] zum Einsatz. Die Geschäftsmodellentwicklung basiert auf einer Vielzahl von Annahmen und ist daher mit einem gewissen Risikofaktor behaftet. Um die ökonomische Tragfähigkeit eines Geschäftsmodells zu evaluieren, ist es wichtig, die zu Grunde gelegten Annahmen zu identifizieren, in Hypothesen zu überführen und mit entsprechenden Experimenten/Methoden zu evaluieren. Durch Business Model Evaluation werden die erforderlichen Informationen für die Evaluierung generiert und dadurch die Unsicherheiten verringert, um die Risiken des neuen Geschäftsmodells zu managen [18]. Bei der Geschäftsmodellentwicklung muss der Nachweis erbracht werden, dass das entwickelte Geschäftsmodell technisch und rechtlich realisierbar sowie ökonomisch tragfähig ist. Darüber hinaus ist die Zahlungsbereitschaft von möglichen Kunden nachzuweisen. Nach Ende eines Geschäftsmodell-Innovationsworkshops stellt sich daher oft die Frage, wie sich die kreativ entwickelten und mittels Business Model Canvas (BMC) visualisierten Geschäftsmodelllösungen [17] in stärker formalisierte betriebswirtschaftlich auswertbare Methoden überführen lassen und dadurch in einem weiteren Schritt zur Implementierung in ein Geschäftsmodell eignen.

Die Geschäftsmodelle in SMASA wurden in verschiedenen Geschäftsmodell- Innovationsworkshops mit unterschiedlichen Methoden entwickelt. In einem ersten Workshop mit Beteiligungen von Bürgern aus Sachsenhausen wurde versucht, verschiedene Geschäftsmodellideen zu evaluieren. Dabei stand die Einholung von Feedback zu den einzelnen Geschäftsmodellideen im Vordergrund. Am HHZ wurde in einem Workshop mittels einer Geschäftsmodellinnovationsmethode, welche im Kapitel „Geschäftsmodell-Innovationsworkshops“ näher beschrieben wird, zwei Geschäftsmodelle entwickelt. Diese dienten als Grundlage für die in diesem Bericht vorgestellten Geschäftsmodelle. Anschließend wurden die einzelnen Geschäftsmodellideen in Form eines BMC beschrieben und bei einem Workshop bei den Stadtwerken Gingen, wurden auf Basis der zuvor durchgeführten Workshops drei Geschäftsmodelle identifiziert, welche weiter untersucht werden sollten. An diesem Workshop waren Teilnehmer der HS Aalen, der Stadtwerke Gingen und der HS Reutlingen beteiligt.

Die Geschäftsmodelle in der Domäne der erneuerbaren Energien werden meist in Form eines komplexen Business Ecosystems realisiert [19]. Daher reicht eine Beschreibung in der Darstellung eines BMC nicht aus, um ein holistisches Verständnis über das Geschäftsvorhaben zu entwickeln. Zusätzlich ermöglicht die Darstellung eines Geschäftsmodells in verschiedenen Modellierungssprachen eine Analyse unterschiedlicher Perspektiven. Hierdurch kann das Geschäftsmodell in unterschiedlichen Kontexten analysiert werden, welches wieder neue Erkenntnisse über ein neues Geschäftsvorhaben ermöglicht. Als Grundlage der Modellierung im BMConfig dient ein BMC, welches als Ergebnis eines Geschäftsmodell-Innovationsworkshops oder einer Analyse eines bestehenden Geschäftsmodells entwickelt wurde. Ein Mapping zwischen der e3Value-Ontologie und dem BMC ist ebenfalls in Abbildung 18 dargestellt.

Das Mapping von Caetano et al. [20] wurde erweitert und auf die erweiterte BMO von Meertens [21] übertragen. Die beiden Konzepte des BMC Customer Segments und Key Partners lassen sich 1:1 in ein e3Value-Modell überführen. Für die Modellierung können Akteure oder Market Segments eingesetzt werden. Das Konzept Key Activities sind die zentralen Tätigkeiten, welche ein Unternehmen zur Erbringung der Value Proposition ausüben muss. Dieses lässt sich in e3Value durch eine Value Activity

beschreiben, die ein Akteur zur Steigerung seines Nutzens ausführt, oder durch Value Exchanges mit Value Objects, welche die Beschaffung benötigter Ressourcen mit einem Key-Partner darstellen. Eine Key-Ressource wird mittels eines Value Objects in e3Value abgebildet, welche von einem Key Partner erworben wird. Die Value Proposition ist das Wertangebot, das ein Unternehmen seinen Kunden anbietet. In e3Value wird die Value Proposition mittels eines Value Interface und den dazugehörigen Value Ports dargestellt. Die Customer Relationships und Channels werden durch Value Exchanges und Value Objects in e3Value abgebildet. Dabei wird über dieses die Value Proposition an den Kunden gebracht. Revenue Streams und Cost Streams stehen im Zusammenhang mit den Value Exchanges, und Value Objects und müssen entsprechend mit Preis und Stückzahl versehen werden. Der BMC ermöglicht die Darstellung eines Geschäftsmodells auf einer höheren Granularität und nimmt dabei eine eher strategische Perspektive ein [20]. Mittels des BMConfig wird eine Modellierung, welche näher an der eigentlichen Operationalisierung des Geschäftes ist, ermöglicht. Deshalb müssen bei der Transformation eines BMC zum e3Value-Modell weitere Informationen über das Geschäftsvorhaben identifiziert werden. Die Überführung des BMC in eine e3Value-Ontologie ermöglicht eine Analyse auf Basis des BE-Ansatzes und durch die Simulation eine Untersuchung der ökonomischen Tragfähigkeit des Geschäftsmodells.

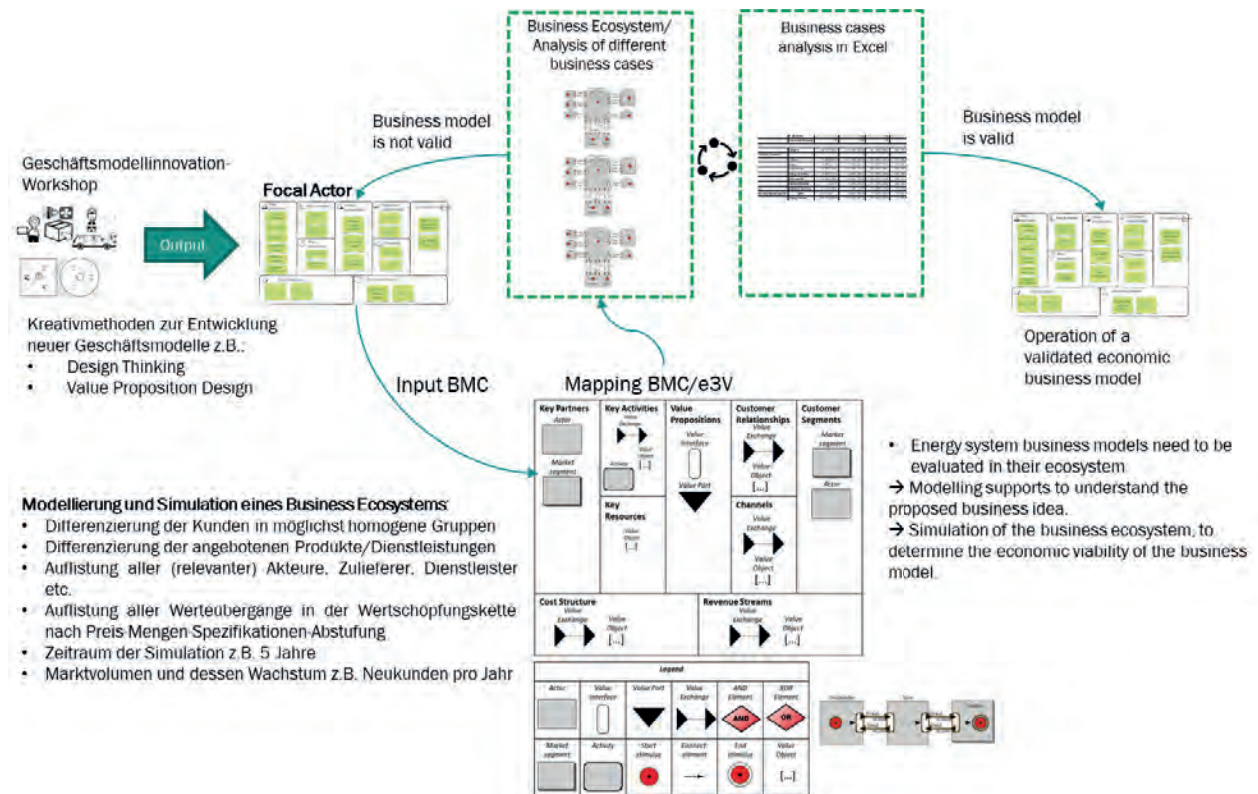


Abbildung 18: Methodik der Erarbeitung von neuen Geschäftsmodellen im Projekt SMASA

Nachdem die Geschäftsmodelle, welche in Form eines BMC vorlagen, in den BMConfig übertragen wurden, wurden die benötigten Daten für die Simulation mittels Literatur- und Internetrecherche erhoben. Anhand mehrerer Webkonferenzen wurden zusätzliche Daten zu den Geschäftsmodellen erhoben und weitere Szenarien diskutiert. Hierbei stand im Fokus, ob alle relevanten Akteure und Value Exchanges identifiziert wurden. Anschließend wurden die einzelnen Value Exchanges quantifiziert. Da die Software den einzelnen Partnern aufgrund eines komplexen Installationsprozesses nicht zur Verfügung gestellt werden konnte, wurde die Simulation in Excel mit derselben Logik wie im

BMConfig nachgebaut und konnte dadurch für einzelnen Partner nochmals nachvollzogen werden. Die zugrundeliegende Methodik dieses Berichts ist in Abbildung 18 dargestellt.

5.4. Geschäftsmodell-Innovationsworkshops

Zur Entwicklung der Geschäftsmodelle wurde die im Projekt Digitrans [22] entwickelte Geschäftsmodell-Innovationsmethode verwendet, welche speziell für die Bedürfnisse von KMUs entwickelt wurde. Digitrans steht dabei als Akronym für „Digitale Transformation in der Donau-Region“. Teilnehmer des Workshops waren der Geschäftsentwicklungsleiter der Stadtwerke Gingen, der Vertriebsleiter der Stadtwerke Aalen sowie ein Mitarbeiter und insgesamt sechs wissenschaftliche Mitarbeiter der Hochschulen Aalen und Reutlingen. Der Workshop wurde in dem Kreativlabor der Hochschule Reutlingen am HHZ durchgeführt. Ziel des Workshops war es, neue Geschäftsmodelle für Stadtwerke im ländlichen Raum zu entwickeln. Die Digitrans-Innovationsmethode ist in Abbildung 19 dargestellt. Das DIGITRANS-Framework besteht aus zwei Hauptphasen. Die erste Phase umfasst die Innovationsprozesse (Analyse, Design und Test). Die zweite Phase enthält die Transformationsprozesse (Implementierung und Evaluation). Die Konzeptionierung und Entwicklung der Geschäftsmodelle erfolgt in der Innovationsphase. Die Transformationsprozesse wurden im Projekt Digitrans nur konzeptionell ausgearbeitet, aber nicht empirisch validiert. Aus diesem Grund nutzten wir nur die Innovationsphase der Methode.

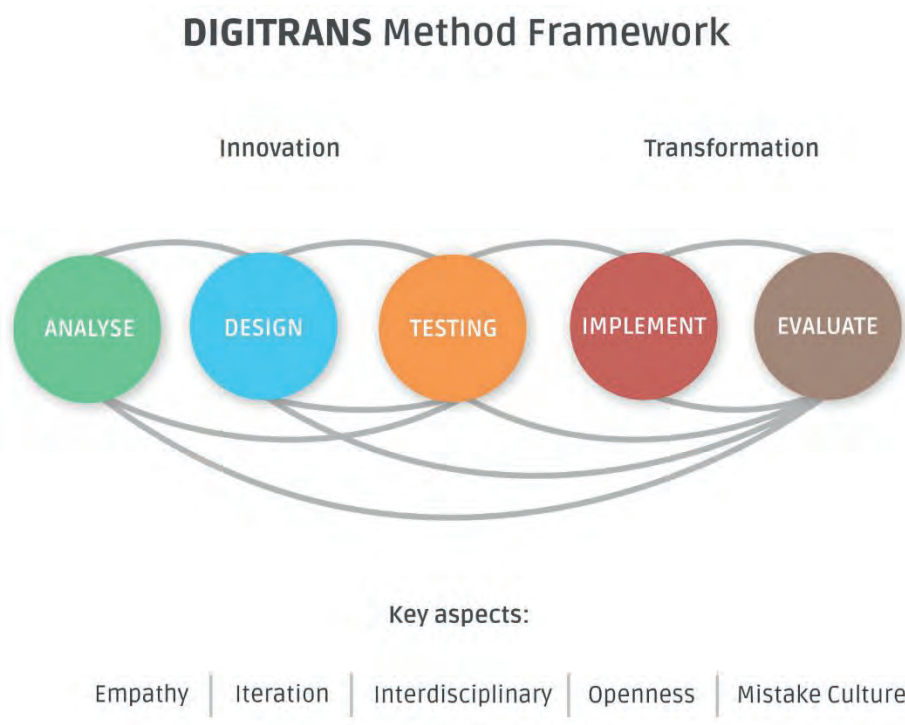


Abbildung 19 Digitrans Vorgehensmodell zur Entwicklung von Geschäftsmodellen

Die erste Phase der Methode ist die Analysephase, welche wiederum in zwei Sub-Phasen unterteilt ist. Die Insight-Phase und die Customer and Value Definition Phase. Dabei zielen die Methoden in der Insight-Phase darauf ab, die aktuelle strategische Landschaft, die Zukunftsaussichten des Unternehmens in Bezug auf den Markt, Trends, Kunden, Nichtkunden und Wettbewerber zu identifizieren. Dadurch kann eine klare Vorstellung gewonnen werden, wie das aktuelle Geschäftsmodell in Bezug auf Wettbewerbstrends, Kundentrends, Technologie- und Rechtstrends einzuordnen ist und zusätzlich können relevante Veränderungen im Markt identifiziert werden. Die

Ergebnisse der strategischen Landkarte des Workshops sind in Abbildung 20 dokumentiert. Eine zweite Methode, welche in der Insight-Phase betrachtet wurde, ist die Stakeholder-Map. Diese ist ebenfalls in Abbildung 5 dargestellt. Dadurch sollen die verschiedenen Stakeholder, die an einem bestimmten Produkt oder einer Dienstleistung beteiligt sind, z.B. Kunden; Partner, Lieferanten usw., identifiziert werden.



Abbildung 20 Strategische Landkarte und Stakeholder Map: Ergebnisfoto aus dem Geschäftsmodell-Innovationsworkshops in SMASA

In der zweiten Sub-Phase *Customer and Value Definition* werden relevante Informationen über die Kunden, deren Wünsche und Bedürfnisse des beteiligten Unternehmens gesammelt. Ziel ist es, eine Value Proposition zu entwickeln, welche die Bedürfnisse der Kunden in den Fokus rückt. Hierbei kam der Value Proposition Canvas (VPC) von Osterwalder et al. [23] zum Einsatz. Dieser dient zur systematischen Analyse des Kundensegmentes und veranschaulicht die Bedürfnisse sowie Probleme der Kunden. Hierbei müssen zuerst die vom Kunden zu bewältigenden Aufgaben definiert und darauf aufbauend unangenehme Situationen und Hindernisse beschrieben werden, welche die Zielgruppe beim Ausführen der Aufgaben hindert bzw. die Ausführung erschwert (= Pains). Im Gegenzug werden auch positive Aspekte betrachtet, welche die Zielgruppe beim Ausführen der Aufgaben unterstützen und erleichtern (= Gains). Bei diesem Prozess soll sich der Anwender in die Situation des Kunden versetzen und Empathie mit diesem aufbauen. Dies ermöglicht die Entwicklung einer kundenzentrierten Value Proposition. Im Workshop wurden zwei Gruppen gebildet, welche zu einem gewählten Kundensegment parallel eine Brainstorming-Session durchführten. Dabei fokussierte sich eine Gruppe auf das Kundensegment „Ü50-Paare“ und die andere Gruppe betrachtete das Kundensegment „junge Paare/Familien“. In Abbildung 21 und Abbildung 22 sind die beiden VPCs in Form von Fotos dokumentiert. Dabei fokussierte sich die Ü50-Paare Gruppe auf Kunden, welche eine PV-Anlage besitzen und die 20 Jahre Einspeisevergütung für den erzeugten PV-Strom ausläuft. In diesem Kontext stellte sich die Frage, wie diese Kunden unterstützt werden können und wie sie den nicht-verbrauchten Strom vermarkten können. In der „junge Paare/Familien“-Gruppe, wurde untersucht, wie Paare oder junge Familien beim Bau oder Umbau ihres Eigenheims bzw. bei der

Konzipierung eines neuen Energiesystems unterstützt werden können. In Anbetracht der hohen Immobilienpreise, stellen die hohen Investitionskosten dabei eine zentrale Barriere dar.

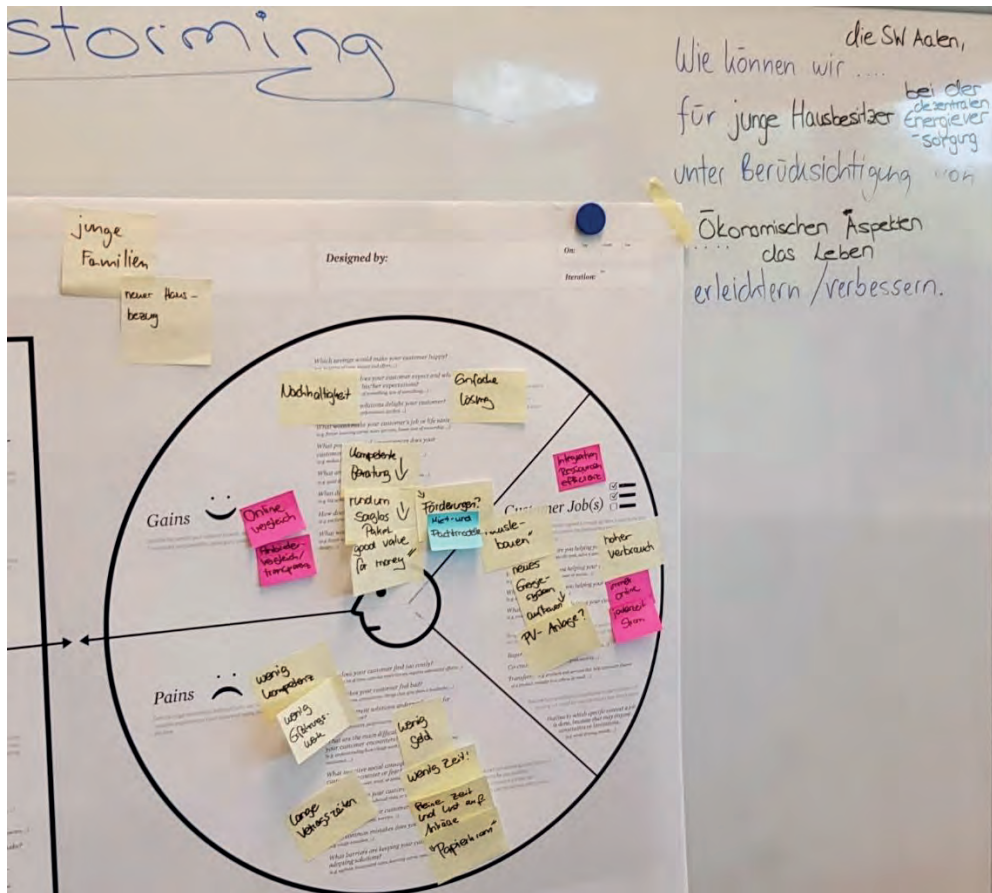


Abbildung 21 Ergebnisdokumentation der Gruppe „Junge Paare/Familien“ im SMASA Workshop

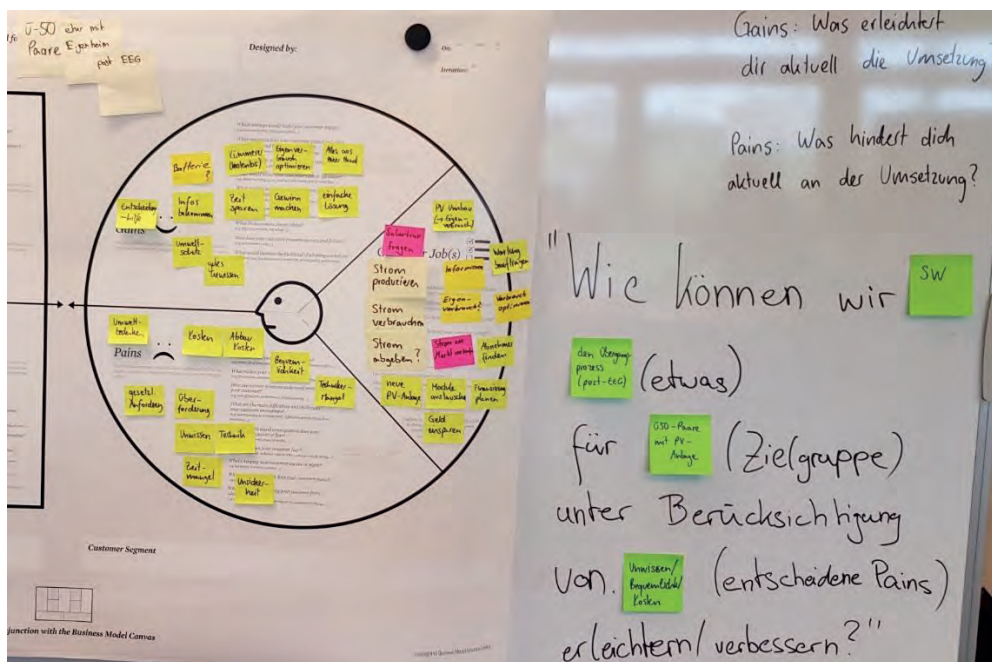


Abbildung 22 Ergebnisdokumentation der Gruppe „Ü50 Paare Post EEG“ im SMASA Workshop

Im nächsten Schritt der Methode erfolgte die Design-Phase, welche aus den Schritten Ideation und Prototype besteht. Nachdem das Customer Segment beschrieben wurde, sollte eine konkrete

Dienstleistung oder ein Produkt konzipiert werden, welches eine hohe Vielzahl der Kundenbedürfnisse adressiert. In der Ideation Phase werden in einer Brainstorming-Session Ideen entwickelt, die das zuvor definierte Kundenprofil adressieren. Anschließend werden die besten Ideen für ein innovatives Geschäftsmodell ausgewählt. In einem nächsten Schritt wird ein Slogan entwickelt, welcher die Idee der Geschäftsmodelle zusammenfasst.

Um den Fokus für die Brainstorming-Session zu erhöhen, werden zuerst in einen Satz die Zielgruppe und entscheidenden zu adressierenden Pains in der Lösung zusammengefasst. Dieser Satz ist in der Abbildung 21 und Abbildung 22 für die jeweiligen Gruppen auf der rechten Seite dargestellt. In Abbildung 23 und Abbildung 24 sind die Ergebnisse der Brainstorming-Session dokumentiert.

Der Slogan für die Gruppe junge Paare/Familien lautet: „Ihr smartes Home. Wir begleiten Sie bei der Planung ihres Eigenheims.“

Die Gruppe Ü50-Paare entwickelte den folgenden Slogan: „Sonnen-Strom-Coach: Wir kümmern uns um Ihre sonnige Zukunft.“

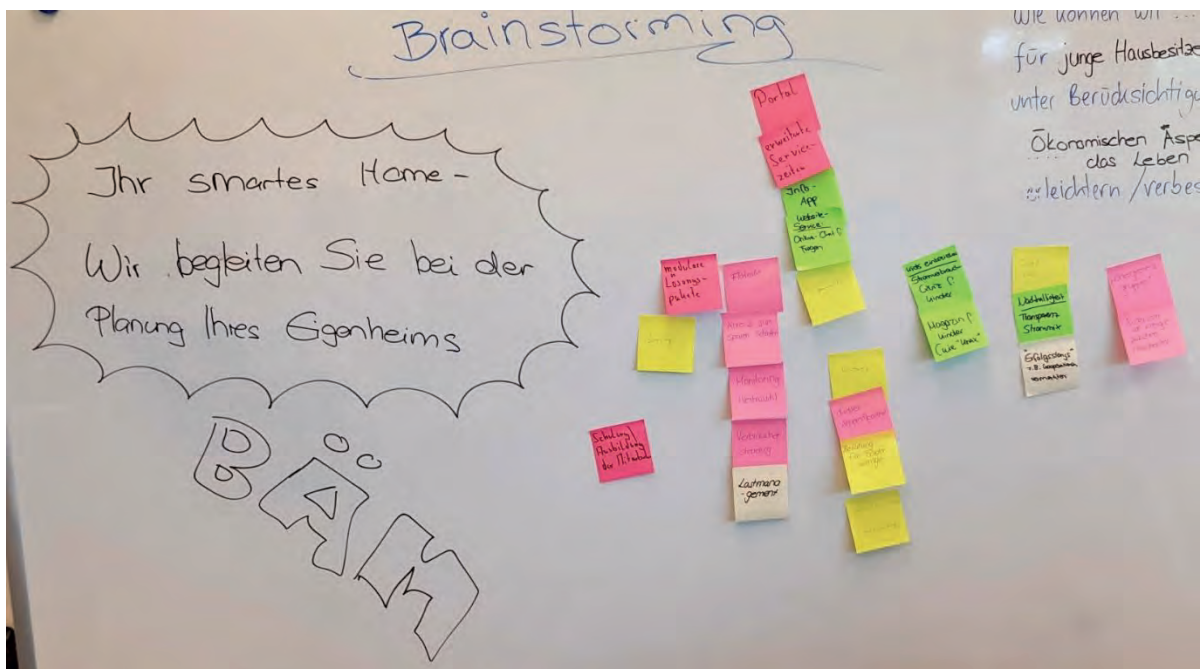


Abbildung 23 Lösungsentwicklung und Slogan der Gruppe „Junge Paare/Familien“ im SMASA Workshop

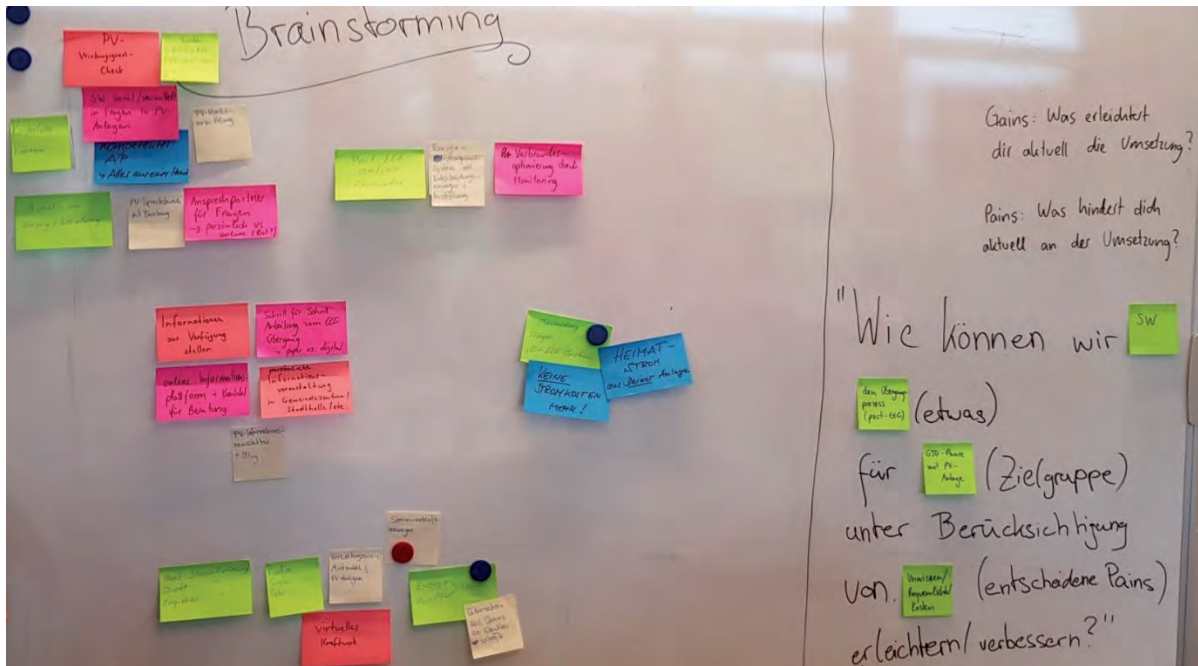


Abbildung 24 Lösungsentwicklung und Slogan der Gruppe „Ü50 Paare“ im SMASA Workshop

In der zweiten Sub-Phase der Methode steht die Entwicklung eines Prototyps im Mittelpunkt. Dabei soll der zuvor definierte Slogan in dem Prototyp widerspiegelt werden. Zur Entwicklung der Prototypen werden SAP-Scenes eingesetzt. SAP-Scenes ist eine Methode zur Entwicklung von Storyboards. Mittels Storytelling kann eine Art von Prototyping aufgebaut werden, bei dem eine Idee erzählerisch dargestellt wird, indem die Informationen visualisiert werden. Mithilfe von Materialien wie Pappfiguren oder Comics können Szenen erschaffen und spielerisch und lebendig eine Geschichte erzählt werden. In einem weiteren Schritt wurden die Ergebnisse der Storyboards mit SAP-Scenes in einem BMC übertragen. Alle erarbeiteten Ergebnisse in den Workshops wurden während des Workshops gegenseitig präsentiert, um eine erste Validierung der Ergebnisse vorzunehmen und so die Diskussion anzuregen und eine Qualitätssicherung zu implementieren. Die Storyboards des Workshops sind in Abbildung 25 und Abbildung 26 dargestellt.

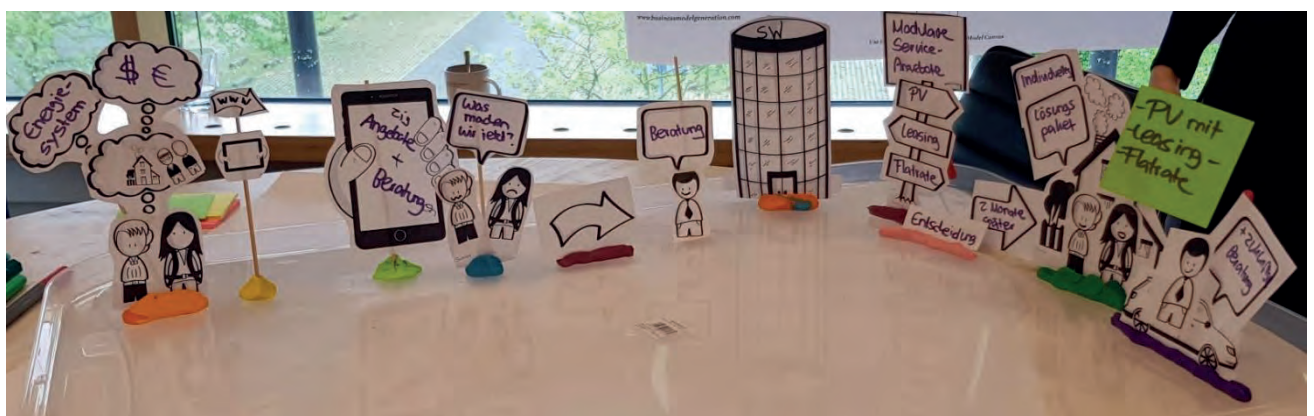


Abbildung 25 Storyboards der Gruppe „Junge Paare/Familien“ im SMASA Workshop



Abbildung 26 Storyboards der Gruppe „Ü50 Paare“ im SMASA Workshop

In Abbildung 27 und Abbildung 28 sind die BMCs für die beiden Geschäftsmodelle dargestellt.

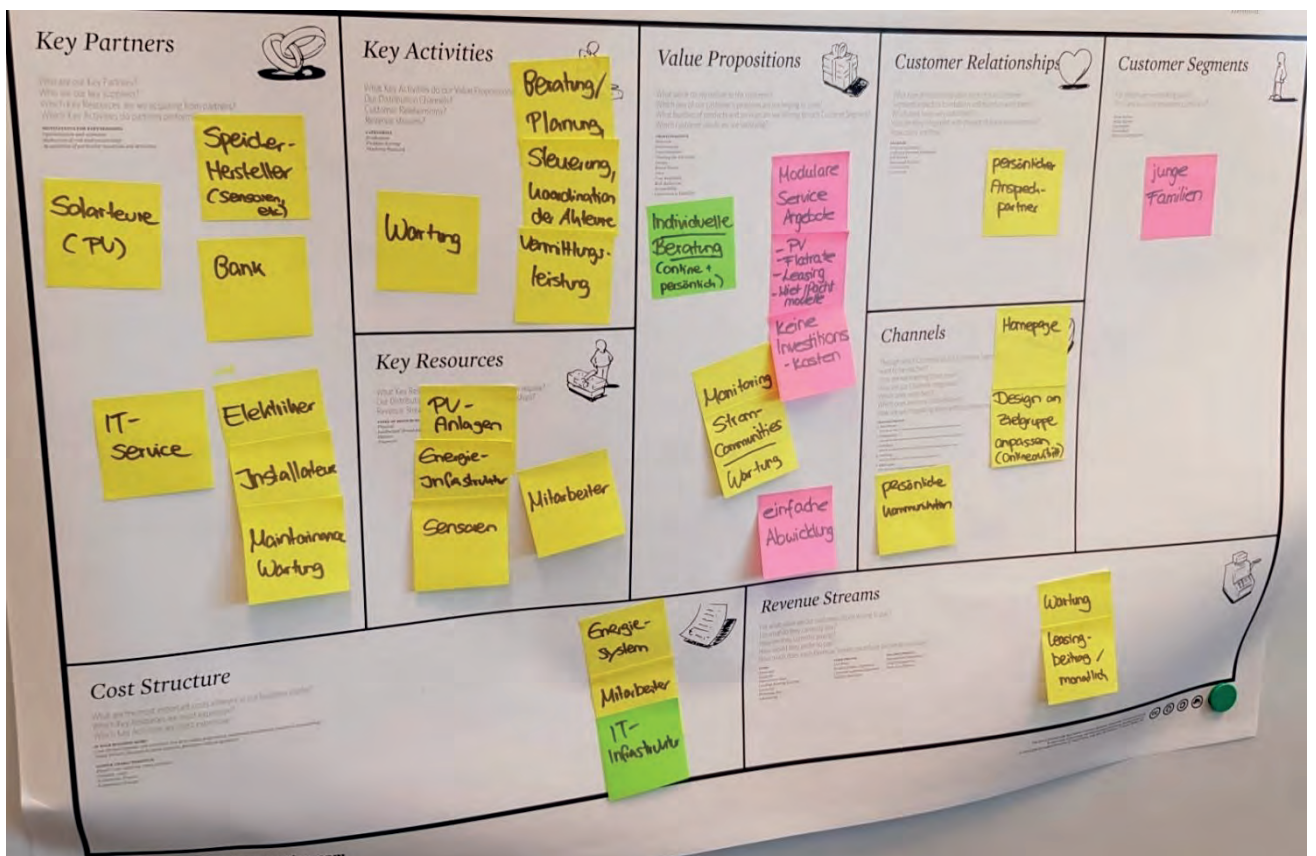


Abbildung 27 BMC der Gruppe „Junge Paare/Familien“ im SMASA Workshop

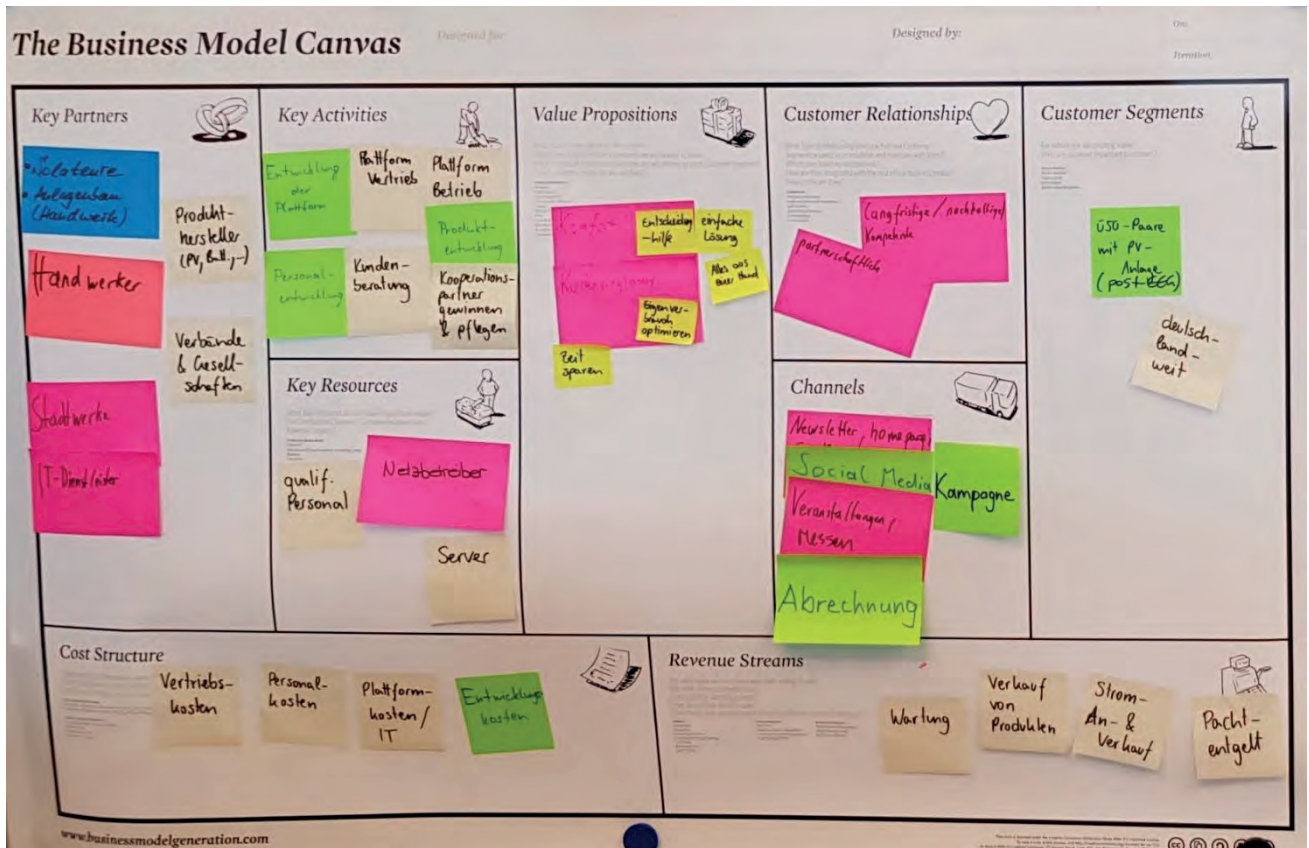


Abbildung 28 BMC der Gruppe „Ü50 Paare“ im SMASA Workshop

5.4.1. Strom-Community

Eine Strom-Community soll verschiedene Prosumenten und Stromkonsumenten miteinander verbinden und das Gemeinschaftsgefühl untereinander stärken. Dies kann ab einer bestimmten Größe der Strom-Community zu einem Netzwerkeffekt führen. Dabei stellen die Community-Mitglieder ihren Strom dem Strom-Community-Betreiber zur Verfügung und beziehen ihren Reststrom auch wieder aus der Community. Zusätzliche Dienstleistungen werden in einen Mitgliedsbeitrag inkludiert. Dieser kann an den Vorjahresverbrauch gekoppelt werden. Für Eigenheimbesitzer wird beispielsweise unter der Angabe des Vorjahresstromverbrauchs der jährliche Reststrombedarf errechnet. Daraus folgt der monatliche Beitrag. Technisch wird das Modell mithilfe von Smart-Metern umgesetzt. Diese geben grundsätzlich zwei wichtige Informationen an die Steuerzentrale ab: (i) es herrscht ein Stromüberschuss in einem Haushalt und Energie kann abgegeben werden oder (ii) der Strombedarf eines Haushaltes kann momentan nicht selbst gedeckt werden und es muss zusätzliche Energie aus der Community bezogen werden.

Das Geschäftsmodell für die Stadtwerke wird anhand des folgenden BMC in Abbildung 29 dargestellt und erläutert. Kunden sind alle Mitglieder, die eine eigene Erzeugungsanlage wie PV, Wasser- und Windkraft oder BHKW besitzen und betreiben. Zusätzlich können auch sensible Stromkunden ohne Erzeugeranlagen mit diesem Angebot angesprochen werden. Der Kundennutzen besteht aus dem Angebot der Strom-Community und dem Strom-Sharing. Durch den monatlichen Mitgliedsbeitrag besteht zudem eine Kostensicherheit, da keine Nachzahlungen anfallen. Die Anlagen werden mithilfe einer Mess- und Steuerbox angeschlossen und sind somit zu einem virtuellen Kraftwerk verbunden. Dadurch kann der Anbieter eine intelligente, sichere und nachhaltig dezentrale Energieversorgung

gewährleisten. Der Kunde hat volle Transparenz über seinen Stromverbrauch. Eine Webseite, Veranstaltungen, Flyer und Energieberatung dienen hierbei als klassische Vertriebskanäle. Auch das Unterbreiten des Angebots an Bestandskunden mittels eines Informationsflyers oder einen Newsletter wäre möglich. Da die Mitglieder durch eine zugehörige Software stets einen guten Überblick über die erzeugte sowie verbrauchte Energie erhalten, ist die Beziehung eher automatisiert und erfordert keine großen zusätzlichen Ressourcen. Durch die Genehmigung, die Speicher der Kunden be- und entladen zu dürfen, sind hier zusätzliche Einnahmen im Rahmen der Regelenergievermarktung erzielbar. Damit die Strom-Community funktionieren kann, benötigt das Modell ausreichend Mitglieder. Daher gehören Energieerzeuger zu den unentbehrlichen Schlüsselressourcen. Eine intelligente Schnittstelle und Messsysteme sind ebenso wichtig, um das Modell bilanzierungsfähig zu machen. Der Betreiber der Strom-Community ist für den Ausgleich und die störungsfreie Abwicklung von Angebot und Nachfrage der benötigten Energie zuständig. Durch die Anbindung mehrerer Anlagen baut der Anbieter ein virtuelles Kraftwerk auf. Für diese technische Umsetzung sind Dienstleister und Hersteller von Mess- und Steuerboxen notwendige Schlüsselpartner. Einge kaufte Dienstleistungen sowie Hard- und Softwarelösungen sind die Kosten, welche zur Erbringung des Wertangebotes entstehen und in der Kostenstruktur berücksichtigt werden müssen. Auch die Bereitstellung von elektrischer Energie und deren Lieferung ist mit Kosten verbunden. Daher sind alle steuerlichen Abgaben, Umlagen sowie die entstehenden Kosten für die Beschaffung, den Vertrieb und Netzentgelte in der Preisgestaltung zu berücksichtigen.

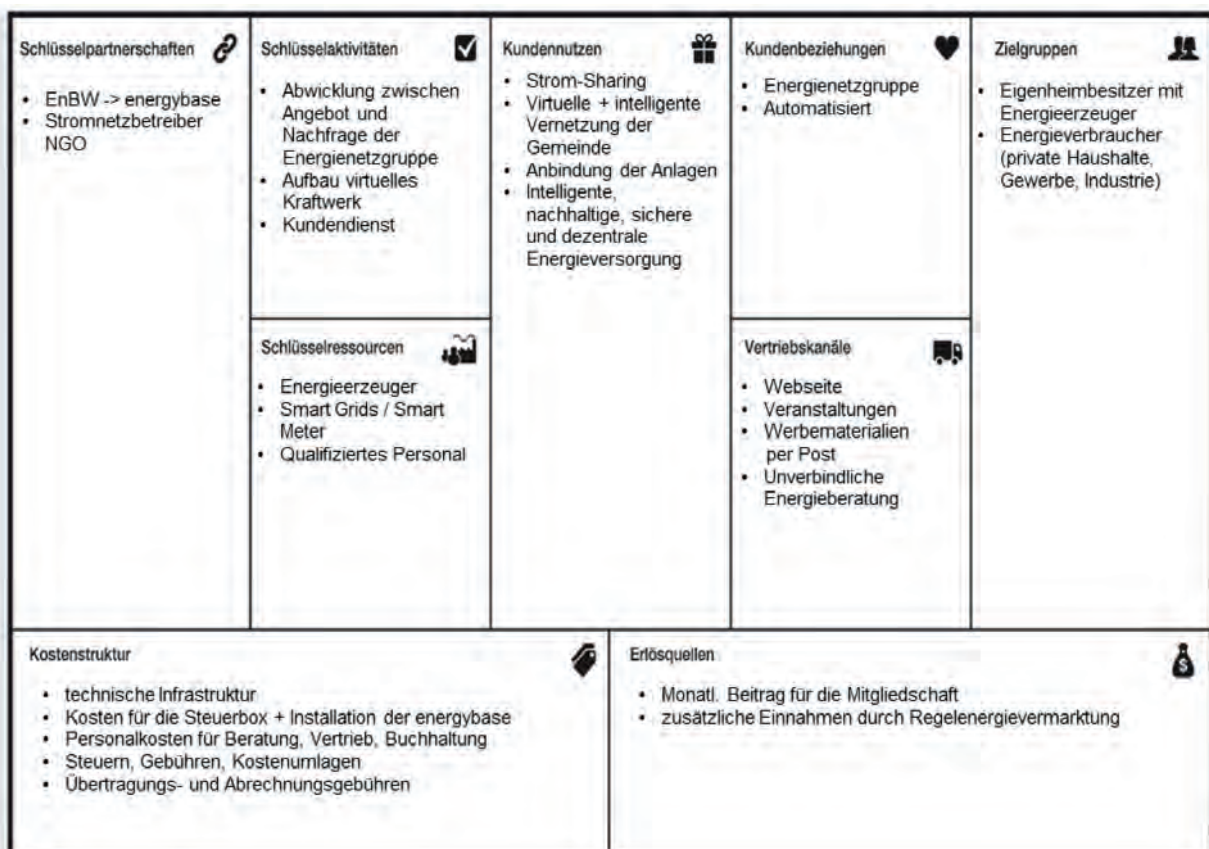


Abbildung 29 BMC des möglichen Geschäftsmodells „Strom-Community“

Je nach Auslegung dieses Community-Modells kann der Anbieter auch als Vermittler bzw. „Trader“ agieren. Hier würde der Anbieter die überschüssige Energie eines Haushaltes aufkaufen und an einen anderen verkaufen. Mit diesem Ansatz kann der technische Anschluss an das Netz eventuell umgangen

werden, sodass Netzentgelte entfallen würden. Grundsätzlich lässt sich das Strom-Sharing Modell gut vermarkten. Durch das Gemeinschaftsgefühl hat der Kunde eine direkte und stärkere Bindung zum Anbieter.

In diesem Abschnitt wird das Geschäftsmodell für eine Strom-Community, welches die Stadtwerke Gingen an der Brenz realisieren möchten, untersucht. Dabei wurde das Strom-Community-Angebot der Firma Albwerk betrachtet, welches als White- Label-Geschäftsmodell speziell für die Kundschaft von Stadtwerken im ländlichen Raum konzipiert wurde. Der BMC hierzu ist in Abbildung 30 dargestellt. Dabei liefert der Anbieter für die Strom-Community-Lösung den Stadtwerken ein White-Label-Produkt mit eigener App, ein Modul zur Eigenstromoptimierung und die Komponenten zum Aufbau einer eigenen Strom-Community. Die zentrale Value Proposition ist der Verkauf einer PV-Anlage und eventuell eines Batteriespeichers. Zusätzlich soll eine App dem Kunden über den aktuellen Verbrauch und die erzeugte Energie der PV-Anlage informieren. Durch ein Monitoringsystem (Energiemanagementsystem) kann der Eigenverbrauch erhöht und das Bilanzkreismanagement optimiert werden. Durch die Integration thermischer Erzeugungsanlagen kann eine Sektorkopplung zwischen thermischer und elektrischer Energie realisiert werden. Durch die Integration eigener Anlagen der Stadtwerke und den Kauf von Strom an der Börse ist die Versorgungssicherheit der Kunden in der Strom-Community gewährleistet. Die Strom-Community kann selbst ein virtuelles Kraftwerk sein oder versuchen, den überschüssigen Strom an einen virtuellen Kraftwerksbetreiber zu veräußern. Die PV-Anlagen der Kunden werden mit einer hohen Eigenverbrauchsquote betrieben, den überschüssigen Strom bieten die Kunden zur Vermarktung an. Die Kundengruppe sind Hauseigentümer ohne PV-Anlage und Speicher. Bezüglich der hohen Investitionssumme für einen Kunden muss ein persönlicher Ansprechpartner vorhanden sein. Über die Homepage können Informationen zum Angebot oder sonstige Informationen, z.B. zur Energiewende, bereitgestellt werden. Durch die Strom-Community könnte ab einer bestimmten Größe an Kunden ein Netzwerkeffekt entstehen. Durch die regionale Strom-Community soll der Kunde auf einer emotionalen Ebene angesprochen werden und sich einer Bewegung zugehörig fühlen.

Key Partners <ul style="list-style-type: none"> Anbieter für Community Lösung und Solarteur VNB/ ÜNB Marketing Agentur Batterie-lieferant 	Key Activities <ul style="list-style-type: none"> Vertrieb Marketing Customer-Relationship-Management 	Value Propositions <ul style="list-style-type: none"> Regionale Community Strom Strom-Sharing Eigenstrom-optimierung Kunden App Batterie PV-Anlage Teil der Energiewende Virtuelles Kraftwerk Feste Vergütung für Einspeisung Sektorkopplung 	Customer Relationships <ul style="list-style-type: none"> Beratung Self-Service Energy Community 	Customer Segments <ul style="list-style-type: none"> Hauseigentümer ohne PV-Anlage und Speicher Eventuell Post-EEG Kunde
Key Resources <ul style="list-style-type: none"> Energieerzeuger CRM Vertriebspersonal Energiemanagementsystem Kunden App Abrechnungssystem Verkaufsfleifaden IT-Infrastruktur 			Channels <ul style="list-style-type: none"> Homepage App Flyer Veranstaltungen 	
Cost Structure <ul style="list-style-type: none"> Anbieter Community Lösung FrontUp 29.000 € Lizenzen 6.000 € PV-Anlagen Speicher Vertrieb 			Revenue Streams <ul style="list-style-type: none"> PV-Anlage Batterie Servicegebühr Reststromlieferung/ Community Fee 	

Abbildung 30 BMC des möglichen Geschäftsmodells „Strom-Community Albwerk“

Mittels des vorgestellten Mappings wurde der entwickelte BMC in eine e3Value-Modellierung in den BMConfig überführt. Im ersten Schritt wurden die Kunden und Key-Partner aus dem BMC in die e3Value-Modellierung überführt. Anschließend wurde erörtert, welche Akteure in Relation stehen und welche Werte diese austauschen. Dazu wurden die Value Exchanges und Value Objects definiert. Durch die feingranulare Modellierung wurden anschließend weitere Akteure und Austauschrelationen identifiziert, welche ein noch realitätsnäheres Business-Szenario ermöglichen. Für die Simulation wurden im nächsten Schritt Kosten und Preise der einzelnen Leistungen identifiziert. Zusätzlich wurden das Marktvolumen, die Absatzmengen und die gewünschte Dauer ermittelt, mit der das Geschäftsmodell simuliert werden soll. In Abbildung 31 ist ein vereinfachtes Modell der Strom-Community dargestellt. Der größte Kostentreiber in diesem Geschäftsmodell ist der Vertrieb von Photovoltaikanlagen (PV-Anlagen). Deshalb wurde für die Analyse und aus Darstellungsgründen ein vereinfachtes Geschäftsmodell modelliert und simuliert. Der Vertrieb wird für eine bessere Übersicht als ein eigenständiger Akteur modelliert. Der Zeitraum für die Simulation wurde in Absprache mit den Stadtwerken auf 7 Jahre festgelegt. Für die Akquise von 20 Neukunden im Jahr rechnen die Stadtwerke, dass ca. 67 Angebote geschrieben werden müssen. Die Aufwände pro Angebot belaufen sich auf ca. 400 €. Die Front-up-Kosten für die Lösungen werden über fünf Jahre abgeschrieben. Zusätzlich fallen in den sieben Jahren Lizenzkosten in Höhe von insgesamt 42 000 € an. Es werden 140 PV-Anlagen und 560 Services über die sieben Jahre veräußert. Der untersuchte Business Case weist einen Verlust von ca. 76 000 € nach sieben Jahren auf.

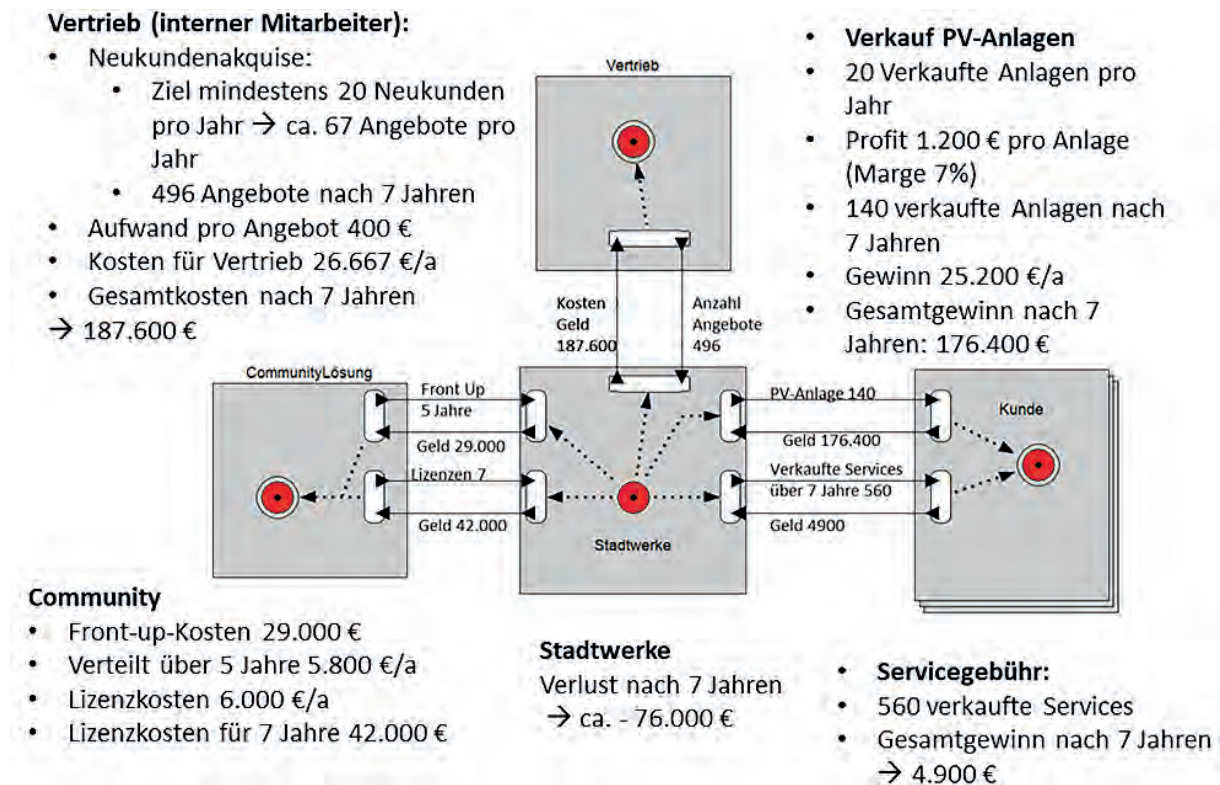


Abbildung 31 Simulation des Geschäftsmodells Albwerk Variante 1

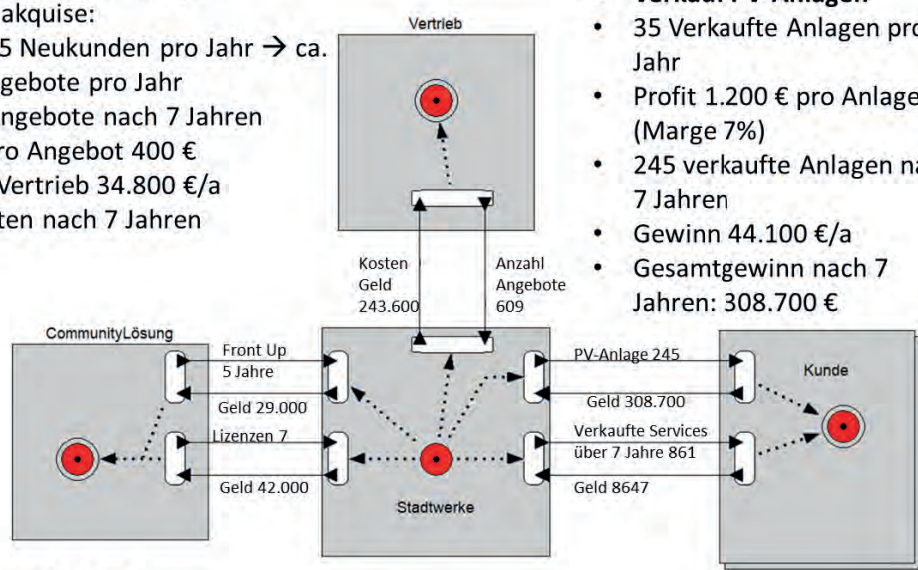
In einem zweiten Szenario wurde versucht, den Break-even für das Geschäftsmodell zu ermitteln. Dabei müssten im Jahr ca. 35 Neukunden akquiriert werden. Zusätzlich müsste die Conversion-Rate bei der Neukundenakquise von 30% auf 40% gesteigert werden. Dafür müssten ca. 87 Angebote im Jahr geschrieben werden. Die restlichen Informationen zu den Simulationsergebnissen können der Abbildung 32 entnommen werden.

Die Überführung des Geschäftsmodells in den BMConfig ermöglicht, die Darstellung verschiedener Ausprägungen des Geschäftsmodells zu vergleichen und dadurch Design-Entscheidungen zu reflektieren und zur weiteren Umsetzung des Geschäftsmodells relevante Komponenten zu identifizieren. Durch die Konkretisierung verschiedener Parameter und in Hinblick auf die Simulation werden neue Erkenntnisse gewonnen. Im Zusammenhang mit dem Geschäftsmodell Strom-Community wurden, durch die Modellierung und Simulation und die sich daraus ergebende Diskussion, die folgenden drei zentralen Barrieren identifiziert: die hohen Vertriebskosten, der höhere Preis für PV-Anlagen gegenüber der Konkurrenz und die geringe Skalierung durch die regionale Verwurzelung der Stadtwerke. Die Analyse zeigt, dass der Verkauf von PV-Anlagen für die Stadtwerke zum jetzigen Zeitpunkt unwirtschaftlich ist und dass weitere Untersuchungen zu diesem Thema notwendig sind.

Vertrieb (interner Mitarbeiter):

- Neukundenakquise:
 - Bei 35 Neukunden pro Jahr → ca. 87 Angebote pro Jahr
 - 609 Angebote nach 7 Jahren
- Aufwand pro Angebot 400 €
- Kosten für Vertrieb 34.800 €/a
- Gesamtkosten nach 7 Jahren → 243.600 €

- **Verkauf PV-Anlagen**
- 35 Verkaufte Anlagen pro Jahr
- Profit 1.200 € pro Anlage (Marge 7%)
- 245 verkaufte Anlagen nach 7 Jahren
- Gewinn 44.100 €/a
- Gesamtgewinn nach 7 Jahren: 308.700 €



Community

- Front-up-Kosten 29.000 €
- Verteilt über 5 Jahre 5.800 €/a
- Lizenzkosten 6.000 €/a
- Lizenzkosten für 7 Jahre 42.000 €

Stadtwerke

Gewinn nach 7 Jahren
→ ca. 1000 €

- **Servicegebühr:**
- 861 verkaufte Services
- Gesamtgewinn nach 7 Jahren → 8647 €

Abbildung 32 Simulation des Geschäftsmodells Albwerk Break-Even

5.4.2. Geschäftsmodell Betreibermodell

In Sachsenhausen ist eine Vielzahl unterschiedlicher Erzeugeranlagen vorhanden. Insgesamt sind 21 PV-Anlagen bei verschiedenen Privathaushalten installiert. Zwei Biogas-BHKW gehören einem lokalen Biobauern und versorgen 50% der Haushalte mit Fernwärme. Das Fernwärmenetz wurde von dem lokalen Biobauern aufgebaut und ist in dessen Besitz. Die thermische Energie wird mit 0,04 bis 0,08 € pro kWh berechnet. Der erzeugte Strom der BHKWs wird ins Netz eingespeist und mit der KWK-Umlage vergütet. Aufgrund der vertraglichen Konstellation aus dem EEG werden die BHKWs nicht dazu genutzt, lokale Spitzen aus PV zu puffern. Da in Zukunft immer mehr verfügbare Batterien aus Elektrofahrzeugen zur Verfügung stehen werden, wurde versucht, ein Geschäftsmodell zum Betreiben der Second-Life-Batterien für den Ausgleich von Erzeugungsspitzen aus PV zu entwickeln. Durch Simulationen der Hochschule Aalen konnte als netzdienstliches Szenario der Betrieb der Speicher im Verbund als Community-Speicher identifiziert werden. Dabei werden die Batteriespeicher nicht auf

einzelne Haushalte aufgeteilt, sondern die dezentral erzeugte Energie wird in einem Quartierspeicher gespeichert. Dieser besteht aus einem Zusammenschluss mehrerer Batteriespeicher und soll einen Ausgleich von Erzeugung und Verbrauch zwischen den einzelnen Akteuren schaffen. Eine Skizze eines Quartierspeichers ist in Abbildung 33 visualisiert. Heimspeicher können von Community-Speichern mit derselben Kapazität ersetzt werden. Laut [24] kann durch die Verwendung eines Community-Speichers für fünf Haushalte die Levelized Costs of Energy Storage um 29% bis 44% reduziert werden.

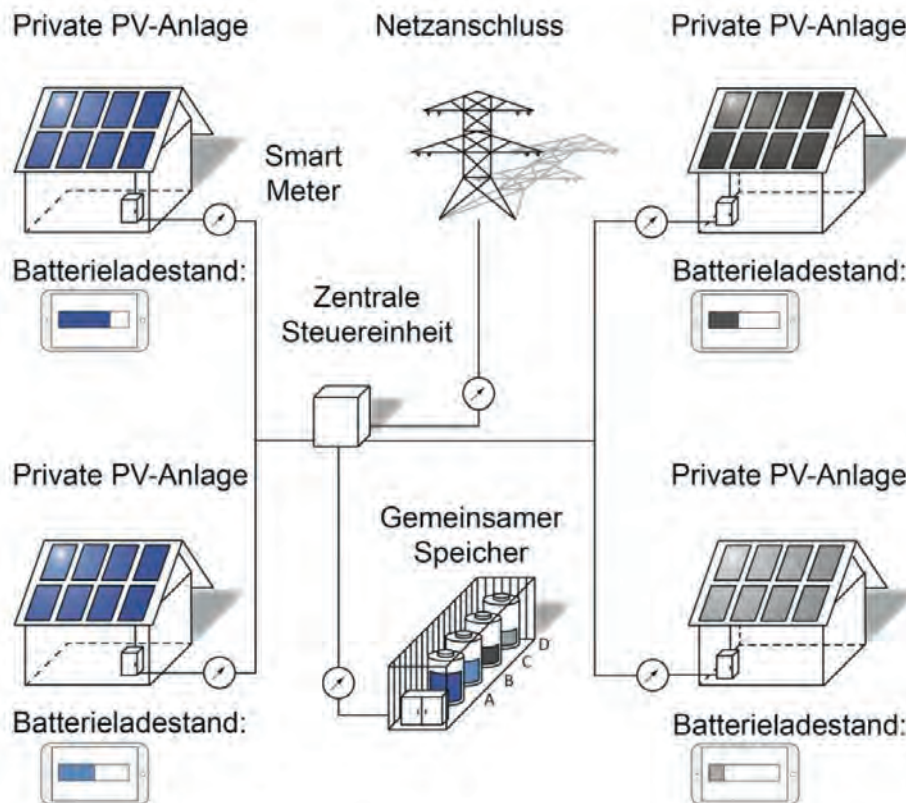


Abbildung 33: Skizze der Einbindung eines Quartierspeichers [25]

Das Geschäftsmodell, welches im Projekt SMASA, für Quartierspeicher-Lösungen entwickelt wurde, ist auf Neukunden ausgerichtet, die noch keine PV-Anlage besitzen. Die grundlegende Idee des Geschäftsmodells ist es, dass die Stadtwerke freie Dachflächen pachten, um auf diesen PV-Anlagen zu errichten. Die Batterien nehmen den PV-Überschuss oder Energie anderer Erzeugungsanlagen auf, um diesen zu puffern und an Dritte weiterzugeben. Dadurch wird ein Ausgleich zwischen Produktion und Verbrauch zwischen den einzelnen Haushalten geschaffen. Dabei sollen die Energiekosten für die Dachverpächter für Jahre konstant und unabhängig vom eigenen Stromverbrauch bleiben. Es fallen dabei keine Investitionskosten für die Dachverpächter an, diese können dennoch Teil der Energiewende sein.

In Abbildung 34 ist der entwickelte BMC des Betreiber Geschäftsmodells dargestellt. Das Wertangebot des Geschäftsmodells, welches den Kunden offeriert wird, adressiert eine Vielzahl von Barrieren bei EE-Geschäftsmodellen. Dabei sind meist die hohen Investitionskosten eine zentrale Barriere für eine Vielzahl von Kunden bei der Anschaffung einer PV-Anlage [26]. Die Kunden können Teil der Energiewende sein und erhalten einen garantierten niedrigen Strompreis durch die zur Verfügung gestellten Dachflächen. Zur Realisierung des Geschäftsmodells werden Waren und Dienstleistungen von mehreren Key Partnern bezogen. Hierbei werden von einem Solarteur PV-Anlagen gekauft und von diesem auf den Dächern der Kunden installiert. Für den Quartierspeicher werden Batterien eines

Batterielieferanten bezogen. Zur Absicherung des Vorhabens werden Versicherungen für die PV-Anlage bei einem Versicherungsunternehmen abgeschlossen. Zusätzlich wird ein Monitoringsystem eingesetzt, welches die einzelnen Komponenten überwacht und bei Unregelmäßigkeiten im System die entsprechenden Handwerker benachrichtigt und einen Support-Prozess auslöst. Der Netzbetreiber ist ebenfalls ein Key Partner in diesem Vorhaben. Hier sind mehrere rechtliche Rahmenbedingungen abzuklären und das Anmelden der PV-Anlagen vorzunehmen. Des Weiteren wird der Quartiersspeicher an das öffentliche Netz angeschlossen. Key Activities werden im Unternehmen erbracht und sind zentraler Bestandteil zur Erbringung der Value Proposition und haben ebenfalls eine Relation zu den Customer Relationships. Zentrale Key Activities sind Vertrieb und Marketing. Dabei ist die Planung der PV-Anlage nicht Bestandteil der Dienstleistung des Vertriebes, sondern die Angebotserstellung sowie die Koordination der einzelnen Akteure zur Installation der PV-Anlage und der Anschluss der Anlage am Quartiersspeicher und Anmelden der PV-Anlage beim Netzbetreiber. Als Key Resources sind zum einem der Quartiersspeicher, die PV-Anlagen, Mitarbeiter und das Monitoringsystem essenziell. Zusätzlich werden Key Resources benötigt zu dem im Building Block Channel aufgeführten Punkten. Für die Homepage wird die entsprechende IT-Infrastruktur benötigt. Als Kundensegment werden Hauseigentümer mit freien Dachflächen angesprochen. Diese können Teil der Energiewende werden, ohne die hohe Investitionssumme aufzubringen. Die Kundenbeziehung erfordert aufgrund der Komplexität des Projektes eine persönliche Beratung. Als Channel zur Ansprache der Kunden werden Informationen über eine Homepage bereitgestellt, die über das Angebot informieren sollen. Zusätzlich soll versucht werden, über einen Flyer aktuelle Kunden der Stadtwerke zu informieren. Bei Veranstaltungen kann ebenfalls auf das Angebot aufmerksam gemacht werden. Für das Geschäftsmodell fallen die folgenden Kosten an: PV-Anlagen, Quartiersspeicher, Vertrieb und Personalkosten, Versicherungen, Ausspeichern von elektrischer Energie, Monitoringsystem. Umsätze können durch die Einnahmen der EEG-Umlage und den Verkauf des Überschussstromes erzielt werden. Des Weiteren fällt eine einmalige Einrichtungsgebühr in Höhe von 600 € an. Mit den Kunden werden Reststromlieferverträge zu einem fixen Preis pro Jahr abgeschlossen, die flexibel an die aktuelle Marktsituation angepasst werden können.

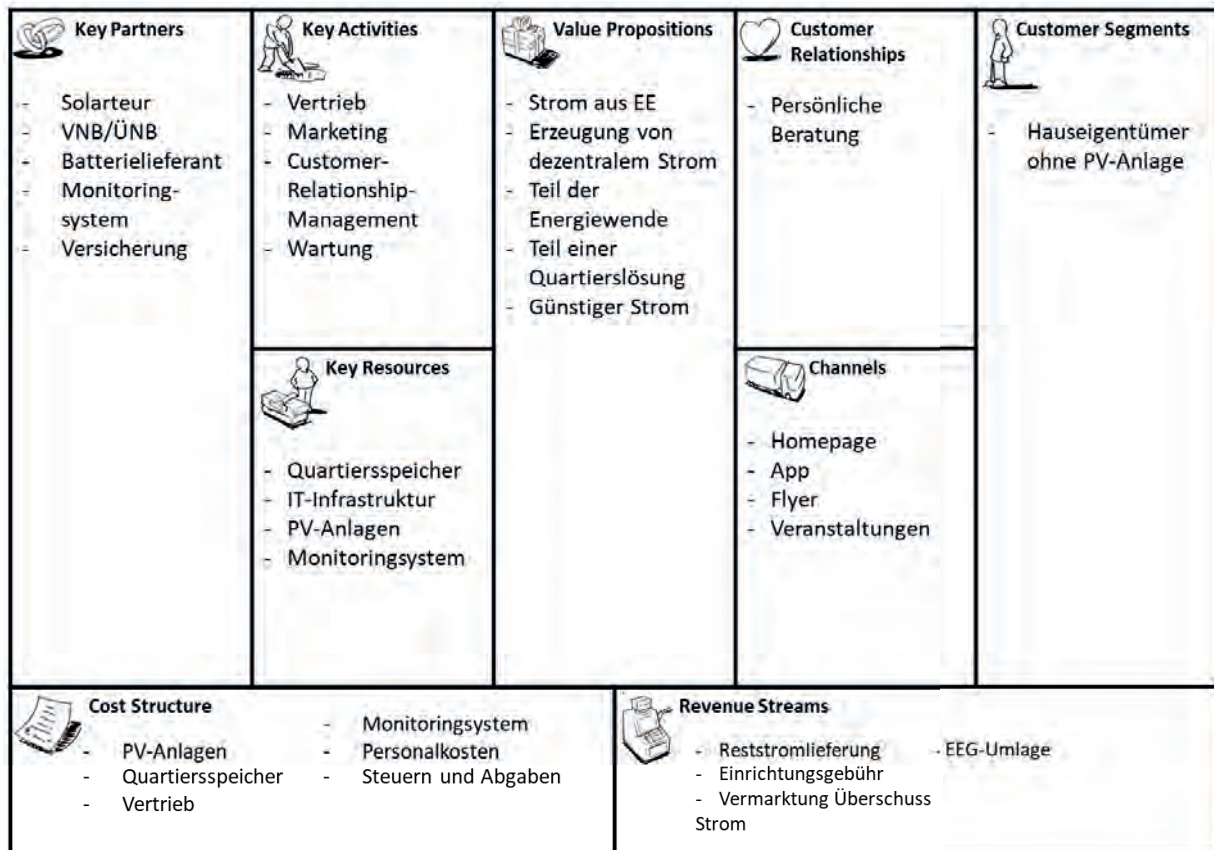


Abbildung 34: BMC des möglichen Geschäftsmodells „Betreibermodell“

Das Geschäftsmodell, welches in Form des BMC vorliegt, wurde anhand des in Kapitel 3 beschriebenen Mappings in dem BMConfig modelliert und anschließend zur besseren Kalkulation in Excel überführt. In Abbildung 35 ist das Geschäftsmodell im BMConfig dargestellt.

Die Grundlegenden Parameter für die Berechnung der Szenarien sind.

- Das Geschäftsmodell wird für einen Zeitraum von 20 Jahren simuliert.
- Es wird mit 20 Neukunden pro Jahr gerechnet.
 - Dabei wird für die Berechnung von Serviceleistungen eine Korrektur mit dem Faktor 0,5 vorgenommen, weil nicht alle Verträge am 01.01. abgeschlossen werden.
- Insgesamt wird das Marktvolumen auf 140 Kunden gedeckelt.
- Die PV-Anlagen werden von einem Solarteur bezogen. Dabei fallen Kosten in Höhe von 1200 € pro kWp bei einer 10 kWp PV-Anlage an. Die Investitionssumme beträgt für 20 Neukunden pro Jahr 240.000 €. Die Anlagen werden auf 20 Jahre linear abgeschrieben. Mit dem Solarteur wird ein Servicevertrag für Wartung und Reinigung geschlossen. Für den Wartungsvertrag fallen Kosten in Höhe von 60 € für die Wartung und 50 € für die Reinigung pro PV-Anlage an. In Tabelle 5 ist eine vereinfachte Kalkulation für den „Key Partner“ Solarteur dargestellt.

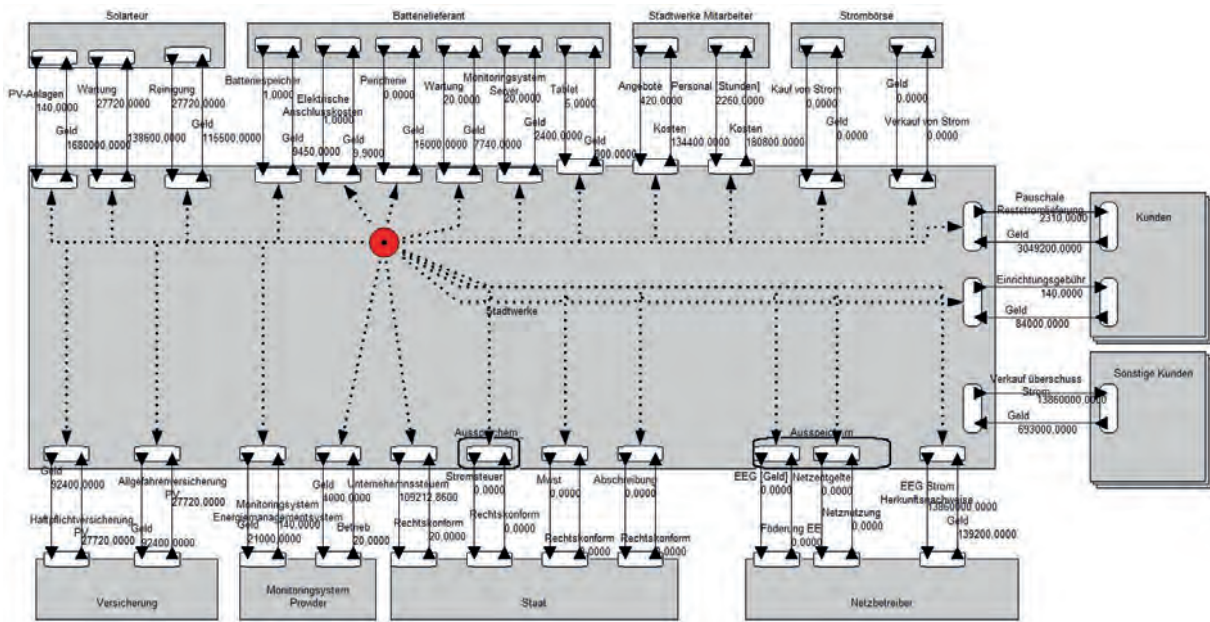


Abbildung 35: Darstellung des Geschäftsmodells „Betreibermodell“ in BMConfig

Tabelle 5 Kalkulation der Leistungen eines Solarteurs im Geschäftsmodell „Betreibermodell“ für einen Zeitraum von 20 Jahren. Die Tabelle wird bis Jahr sieben dargestellt, die übrigen Simulationsjahre sind durch die drei Punkte dargestellt.

Jahre		PV-Anlage 10 kWp	Abschreibung PV- Anlage	Wartung	Reinigung
	Preis	12.000 €	12.000 €	60 €	50 €
	Marktvolumen	140		140	140
	Stückzahl	20 Neukunden			
1		240.000 €	12.000 €	600 €	500 €
2		240.000 €	24.000 €	1.800 €	1.500 €
3		240.000 €	36.000 €	3.000 €	2.500 €
4		240.000 €	48.000 €	4.200 €	3.500 €
5		240.000 €	60.000 €	5.400 €	4.500 €
6		240.000 €	72.000 €	6.600 €	5.500 €
7		240.000 €	84.000 €	7.800 €	6.500 €
...		240.000 €		8.400 €	7.000 €
	Gesamt	1.680.000 €	1.344.000 €	138.600 €	115.500 €

Für den Aufbau des Quartierspeichers werden Batteriespeicher von einem Lieferanten bezogen. Der Preis für Second-Life-Batteriespeicher liegt bei 35 € bis 55 € pro kWh. Die Größe des Speichers wird wegen der lokalen Verfügbarkeit eines bestimmten Typs auf 210 kWh festgelegt. Der Preis pro kWh für den Batteriespeicher wurde mit 45 € angenommen. Somit kostet der Speicher in der Beschaffung 9.450 €. Der Batteriespeicher wird über 10 Jahre linear abgeschrieben. Die elektrischen Anschlusskosten für den Speicher belaufen sich auf 9.900 € und werden ebenfalls über 10 Jahre abgeschrieben. Die Kosten für die Peripherie (z.B. PV-Wechselrichter) liegen nach Rücksprache mit Experten bei ca. 15.000 €. Diese werden ebenfalls auf 10 Jahre linear abgeschrieben. Für die Wartung des Quartierspeichers werden 2% der Summe aus Beschaffungskosten und den elektrischen Anschlusskosten veranschlagt. Auf dem Monitoringssystem-Server wird eine Monitoring-Software zur Überwachung und eine Abrechnungssoftware installiert, welche die ein- und ausgespeiste Energie dem jeweiligen Kunden zuordnet. Das dafür notwendige Tablet wird zu Wartungszwecken im Schaltschrank des Quartierspeichers angebracht und alle 5 Jahre ersetzt.

Zur Absicherung der PV-Anlagen wird eine PV-Haftpflicht und Allgefahrenversicherung eingeplant. Beide Versicherungen werden mit einem Jahresbeitrag von 40 € angenommen. Im ersten Jahr ergeben sich Kosten für die Versicherung von 400 €. Dies liegt an der bereits erwähnten Korrektur bezüglich der Neukunden, die nicht alle am 01.01. einen Vertrag abschließen. Es ergeben sich jeweils Gesamtkosten von 92.400 € über 20 Jahre. Ein Auszug aus der Kalkulation ist in Tabelle 6 dargestellt.

Für die Ausspeicherung der elektrischen Energie aus dem Quartiersspeicher fallen Kosten in Höhe von ca. 0,24 €/kWh an. Dabei wird angenommen, dass ca. 2.000 kWh elektrische Energie pro Haushalt ein- oder ausgespeichert werden. Bei dieser Berechnung wird ebenfalls die Anzahl der Neukunden mit dem Faktor 0,5 berichtigt. In Tabelle 6 ist der Sachverhalt dargestellt. Für die Ausspeicherung der Energie sind die folgenden Kosten zu entrichten: EEG-Umlage, Stromsteuer und Netzentgelte. Zur Speicherung der Energie fallen keine Steuern und Umlagen an.

Mittels eines Monitoringsystems sollen die einzelnen PV-Anlagen überwacht und der Solarteur bei eventuellen Störungen informiert werden. Die Kosten für das Monitoringsystem werden mit 150 € pro Haushalt angenommen. Der Betrieb des Monitoringsystems mit Software-Updates etc. wird auf 200 € im Jahr geschätzt. Zusätzlich verfügt das Monitoringsystem noch über eine Energiemanagement-Komponente, welche die erzeugte Energie der PV-Anlage aufzeichnet und Verbesserungsvorschläge im Bereich Energieeffizienz auf Grundlage der einzelnen Verbrauchströme macht.

Der Vertriebsmitarbeiter der Stadtwerke wird als eigenständiger Akteur dargestellt. Die Erstellung eines Angebotes wird mit 4 Stunden à 80 € kalkuliert. Es wird davon ausgegangen, dass aus drei erstellten Angeboten ein Auftrag resultiert. Daher müssen für die Akquise von 20 Neukunden 60 Angebote erstellt werden. Dadurch werden Kosten in Höhe von 19.200 €/a verursacht. Damit ist der Vertrieb einer der größte Kostentreiber in diesem Geschäftsmodell. In Tabelle 6 ist die Kalkulation für den Vertrieb dargestellt.

Für Verwaltung, Marketing, Abrechnungen und sonstige Aufgaben werden Personalkosten von 1 Stunde pro Kunde/Jahr à 80 € veranschlagt. Hierbei wird der Aufwand immer nur für die Bestandskunden angenommen. Im 8. Jahr werden Personalkosten von 11.200 € für 140 Kunden fällig. In Tabelle 6 ist die Kalkulation dargestellt.

Tabelle 6 Kalkulation der Personalkosten im Geschäftsmodell „Betreibermodell“

Jahre		PV-Haftpflicht	Allgefahrenversicherung PV	Strom aus Speicher	Monitoring-system Energie-management-system	Betrieb	Vertrieb Personal-kosten	Verwaltung, Marketing, Abrechnung Personal-kosten
	Preis	40 €	40 €	0,24 €	150 €	200 €	320 €	80 €
	Marktvolumen	140	140					
	Stückzahl			2000 kWh				1
1		400 €	400 €	4.800 €	3.000 €	200 €	19.200 €	1.600 €
2		1.200 €	1.200 €	14.400 €	3.000 €	200 €	19.200 €	1.600 €
3		2.000 €	2.000 €	19.200 €	3.000 €	200 €	19.200 €	3.200 €
4		2.800 €	2.800 €	28.800 €	3.000 €	200 €	19.200 €	4.800 €
5		3.600 €	3.600 €	38.400 €	3.000 €	200 €	19.200 €	6.400 €
6		4.400 €	4.400 €	48.000 €	3.000 €	200 €	19.200 €	8.000 €
7		5.200 €	5.200 €	57.600 €	3.000 €	200 €	19.200 €	9.600 €
...		5.600 €	5.600 €	67.200 €		200 €	19.200 €	11.200 €
	Gesamt	92.400 €	92.400 €	1.084.800 €	21.000 €	4.000 €	134.400 €	180.800 €

Als Erlöse für das Geschäftsmodell stehen zum einen die Stromlieferung für die einzelnen Haushalte zur Verfügung. Dabei wird von einem Betrag von 110 € pro Monat pro Haushalt ausgegangen. Dies generiert Umsätze in Höhe von 1.320 €/a pro Kunde. Zusätzlich wird eine einmalige Einrichtungsgebühr von 600 € fällig. Der Überschuss an elektrischer Energie wird für 0,05 € veräußert. Hierbei wird von 6.000 kWh im Jahr pro Kunde ausgegangen. Eine zusätzliche Einnahmequelle sind Herkunftsnachweise für EE-Strom. Dabei wird von einem Wert von 0,01 €/kWh ausgegangen. Die Kalkulation ist in Tabelle 7 dargestellt.

Tabelle 7 Kalkulation der Erlöse im Geschäftsmodell „Betreibermodell“

Jahre		Stromlieferung	Einrichtungsgebühr	Verkauf Überschussstrom	Herkunftsnachweise
	Preis	1.320 €	600 €	0,05 €	0,01 €
	Marktvolumen				
	Stückzahl	1		6000 kWh	6000 kWh
1		13.200 €	12.000 €	3.000 €	1.200 €
2		39.600 €	12.000 €	9.000 €	1.800 €
3		66.000 €	12.000 €	15.000 €	3.000 €
4		92.400 €	12.000 €	21.000 €	4.200 €
5		118.800 €	12.000 €	27.000 €	5.400 €
6		145.200 €	12.000 €	33.000 €	6.600 €
7		171.600 €	12.000 €	39.000 €	7.800 €
...		184.800 €		42.000 €	8.400 €
	Gesamt	3.049.200 €	84.000 €	693.000 €	139.200 €

Für den Kunden belaufen sich die Kosten auf 27.000 € in 20 Jahren. Bei einem Jahresverbrauch von 4.500 kWh und einem Strompreis von 0,29 €/kWh würden 26.100 € für den Kunden für elektrische Energie anfallen. Dabei ist der stetig ansteigende Strompreis nicht berücksichtigt. Durch verringerte Abgaben- und Steuerlast könnte ein ökonomisch noch sinnvollerer Angebot für den Endkunden geschaffen und dadurch ein zentraler Beitrag zur Energiewende geleistet werden.

In Abbildung 36 ist die grafische Aufbereitung der einzelnen Ein- und Ausgabenströme dargestellt. Nach 20 Jahren wird ein Gewinn von 254.830,00 € erzielt. Dabei verringert der ausbleibende Kundenzuwachs nach dem siebten Jahr den Gewinn ab Jahr 8. Auch mit einem Sicherheitszuschlag von 5%, für unvorhergesehene Kosten wird ein Gewinn von 188.370,60 € nach 20 Jahren erzielt. Dieser Sachverhalt ist in Abbildung 37 dargestellt. In Tabelle 8 ist das Ergebnis nochmals tabellarisch erfasst.

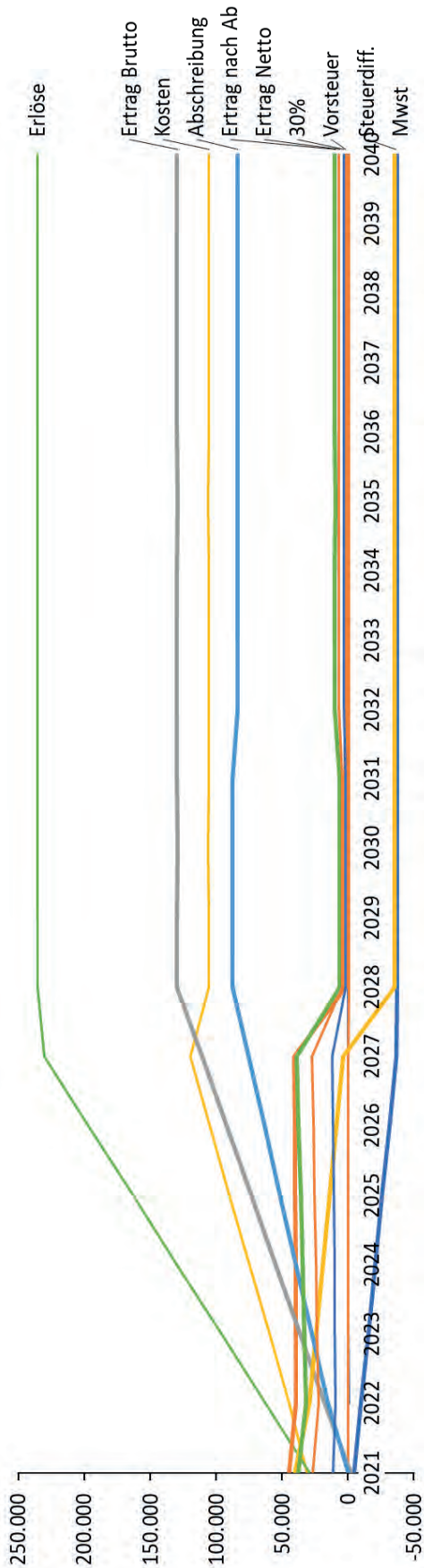


Abbildung 36: Ergebnis der wirtschaftlichen Simulation des Geschäftsmodells „Betreibermodell“

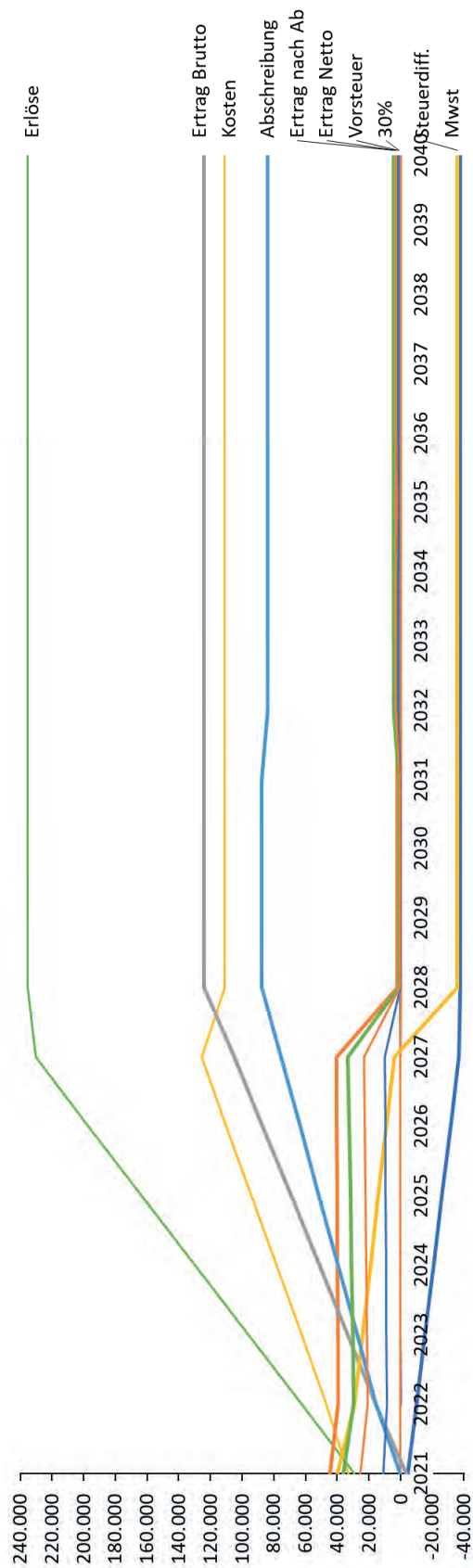


Abbildung 37: Ergebnis der wirtschaftlichen Simulation des Geschäftsmodells „Betreibermodell“ mit 5% Sicherheitszuschlag

Tabelle 8: Detailliertes Ergebnis der wirtschaftlichen Simulation des Geschäftsmodells „Betreibermodell“

Jahr	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035	2036	2037	2038	2039	2040	Gesamt	
Jahr in Betrieb	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20		
Neuansbringung	20	20	20	20	20	20	20	0	0	0	0	0	0	0	0	0	0	0	0	0		
Kosten	31.407,00	44.607,00	59.607,00	74.607,00	89.807,00	104.607,00	119.607,00	105.707,00	105.707,00	105.907,00	105.707,00	105.707,00	105.707,00	105.707,00	105.707,00	105.707,00	105.707,00	105.707,00	105.707,00	105.707,00	105.707,00	1.898.840,00
Erlöse	29.400,00	62.400,00	96.000,00	129.600,00	163.200,00	196.800,00	230.400,00	235.200,00	235.200,00	235.200,00	235.200,00	235.200,00	235.200,00	235.200,00	235.200,00	235.200,00	235.200,00	235.200,00	235.200,00	235.200,00	235.200,00	3.965.400,00
Mwst	4.694,12	9.963,03	15.327,73	20.692,44	26.057,14	31.421,85	36.786,55	37.552,94	37.552,94	37.552,94	37.552,94	37.552,94	37.552,94	37.552,94	37.552,94	37.552,94	37.552,94	37.552,94	37.552,94	37.552,94	37.552,94	-633.131,09
Vorsteuer	44.555,32	39.294,39	39.549,86	39.805,32	40.092,71	40.316,24	40.571,71	1.901,12	1.901,12	1.933,05	1.901,12	1.901,12	1.901,12	1.901,12	1.933,05	1.901,12	1.901,12	1.901,12	1.901,12	1.901,12	1.901,12	308.963,95
Ertrag Brutto	2.007,00	17.793,00	36.393,00	54.993,00	73.393,00	92.193,00	110.793,00	129.493,00	129.493,00	129.293,00	129.493,00	129.493,00	129.493,00	129.493,00	129.293,00	129.493,00	129.493,00	129.493,00	129.493,00	129.493,00	129.493,00	2.066.560,00
Steuerdiff.	39.861,20	29.331,37	24.222,13	19.112,88	14.035,57	8.894,39	3.785,15	35.651,82	35.651,82	35.619,89	35.651,82	35.651,82	35.651,82	35.651,82	35.619,89	35.651,82	35.651,82	35.651,82	35.651,82	35.651,82	35.651,82	-324.167,14
Abschreibung	0,00	15.435,00	27.435,00	39.435,00	51.435,00	63.435,00	75.435,00	87.435,00	87.435,00	87.435,00	87.435,00	87.435,00	87.435,00	87.435,00	87.435,00	87.435,00	87.435,00	87.435,00	87.435,00	87.435,00	87.435,00	1.378.350,00
Ertrag nach Ab	37.854,20	31.689,37	33.180,33	34.670,88	35.993,57	37.652,39	39.143,15	6.406,18	6.406,18	6.238,11	6.406,18	6.406,18	6.406,18	6.406,18	6.238,11	6.406,18	6.406,18	6.406,18	6.406,18	6.406,18	6.406,18	364.042,86
30%	11.356,26	9.506,81	9.954,04	10.401,26	10.798,07	11.295,72	11.742,95	1.921,85	1.921,85	1.871,43	1.921,85	1.921,85	1.921,85	1.921,85	1.871,43	1.921,85	1.921,85	1.921,85	1.921,85	1.921,85	1.921,85	109.212,86
Ertrag Netto	26.497,94	22.182,56	23.226,09	24.269,62	25.195,50	26.356,68	27.400,21	4.484,32	4.484,32	4.366,68	4.484,32	4.484,32	4.484,32	4.484,32	4.366,68	4.484,32	4.484,32	4.484,32	4.484,32	4.484,32	4.484,32	254.830,00

5.4.3. Geschäftsmodelle Post-EEG

Die beiden BMCs aus dem Workshop sind in Abbildung 27 und Abbildung 28 dargestellt. Dabei wurde das Geschäftsmodell Post-EEG aufgrund seiner Wichtigkeit für die Stadtwerke auch nach dem Workshop weiter untersucht.

Im Rahmen des Projektes stehen Privatbesitzer mit PV-Anlagen, welche aus der Vergütung fallen im Fokus. Es werden die folgenden Modelle unterschieden: Sonstige Direktvermarktung, Umrüstung auf Eigenverbrauch mit Überschusseinspeisung im Rahmen der sonstigen Direktvermarktung und Umrüstung auf Eigenverbrauch ohne Überschusseinspeisung (Optional: zusätzliche Nutzung von Speichern oder steuerbaren Verbrauchern)

Geschäftsmodell kleine Direktvermarktung

Die Idee der kleinen Direktvermarktung entspricht zwar nicht den aktuellen gesetzlichen Rahmenbedingungen, dennoch ist dieser ein marktorientierter Ansatz. Dabei ermöglicht es die kleine Direktvermarktung den Post-EEG PV-Anlagenbetreibern, diese noch wirtschaftlich zu betreiben. Im nachfolgenden wird ein Geschäftsmodell für die kleine Direktvermarktung vorgestellt. Dies ist in Abbildung 39 in Form eines BMC dargestellt. Die Erörterung der Probleme des Kundensegments wurde bereits in dem durchgeführten Workshop eruiert und ist in Kapitel 4 beschrieben. Die Autoren der kleinen Direktvermarktung unterscheiden zwischen zwei Kundensegmenten: PV-Anlagen > 7 kWp und < 7 kWp. Dies hängt mit den zu erwartenden Einspeisevergütungen und Kosten zusammen. Die Kosten für eine viertelstündliche Bilanzierung sind für PV-Anlagen < 7 kWp unwirtschaftlich. Weshalb eine Ausnahmeregelung von der Pflicht zu den Viertelstundenscharfen Messungen und einer jährlichen Abrechnung der eingespeisten Strommenge in Form von Standard-Einspeiseprofilen ausgegangen wird. Dabei soll ein Standard-Einspeiseprofil, welches speziell auf das Verhalten von Prosumern angepasst ist, verwendet werden. Hierdurch fallen geringere Kosten für Post-EEG PV-Anlagen < 7 kWp an, weil diese keinen teuren RLM-Zähler anschaffen müssen. Dieser kostet nach Recherchen zwischen 200 € und 1000 €. Für Post EEG PV-Anlagen > 7 kWp ist eine viertelstundenscharfe Messung erforderlich. Ein zentrales Leistungsversprechen der Stadtwerke gegenüber dem Kunden ist die Beratung zu den Möglichkeiten im Zeitalter nach der EEG-Vergütung. Ein zentraler Punkt, welcher während der durchgeführten Internetrecherche zu Post-EEG Geschäftsmodellen und welcher auch im Rahmen der durchgeführten Workshops identifiziert wurde, ist die Umstellung auf Eigenverbrauch. Der größte Anteil der Anfang 2000er Jahre in Betrieb genommenen PV-Anlagen speist aufgrund der hohen EEG-Vergütung zu diesem Zeitpunkt ihren Strom zu 100% ein. Zur Umstellung der PV-Anlagen auf ein Eigenverbrauchsmodell mit Überschusseinspeisung sind in der Regel kleinere Umbauten im Zählerschrank notwendig [27]. Durch die Umstellung der PV-Anlage auf Eigenverbrauch muss kein teurer Strom mehr aus dem Netz bezogen werden. Dabei soll im Rahmen der kleinen Direktvermarktung keine EEG-Umlage auf den Eigenverbrauch anfallen. Zur weiteren Steigerung des Eigenverbrauches beraten die Stadtwerke die Kunden zum Kauf eines Speichers. Ebenfalls werden mit Vertragsabschluss auch die Reststromlieferung an den Kunden übernommen. Zusätzlich könnten die Leistungen im Rahmen von Wartungs- und Reparaturvertrag erweitert werden. Vor Abschluss des Vertrages wird die PV-Anlage und die dazugehörigen Komponenten von einem Solarteuer auf ihre Funktionsweise überprüft. Die wesentlichen Schlüsselaktivitäten sind: Vertrieb und Beratung zur Gewinnung neuer Kunden, Koordination der Handwerker zur Überprüfung und Umrüstung der Anlage auf Eigenverbrauch sowie Vermarktung und Verkauf des Stroms. Die Key Resources sind in erster Linie das intelligente Messsystem zur Bilanzierung und Monitoring. Schlüsselpartner ist der Batterielieferant, welcher den Batteriespeicher verkauft und für den Anschluss und die Wartung und

Reparaturen zuständig ist. Hierbei soll ein Servicevertrag mit dem Batterielieferanten geschlossen werden. Mit dem VNB müssen Daten zur Anlage und die zuvor spezifizierten Informationen ausgetauscht werden. Der Solarteur überprüft die Anlage und nimmt die Umbaumaßnahmen zur Umstellung auf Eigenverbrauch vor. Zusätzlich erbringt der Solarteur Leistungen im Rahmen der Wartung und Reparatur der PV-Anlagen. Zur Realisierung der kleinen Direktvermarktung wird eine Softwarelösung eingekauft. Diese beinhaltet alle wesentlichen Komponenten, die benötigt werden, um die kleine Direktvermarktung zu betreiben. Durch Auswertung der Monitoring-Daten des iMSys durch einen IT-Serviceprovider sollen Wartungen oder eventuelle Schäden frühzeitig erkannt werden. Die Ansprache an den Kunden erfolgt durch eine persönliche Beratung. Aufmerksamkeit für das Angebot kann durch Flyer, entsprechende Einträge auf der Homepage oder durch Veranstaltungen generiert werden.

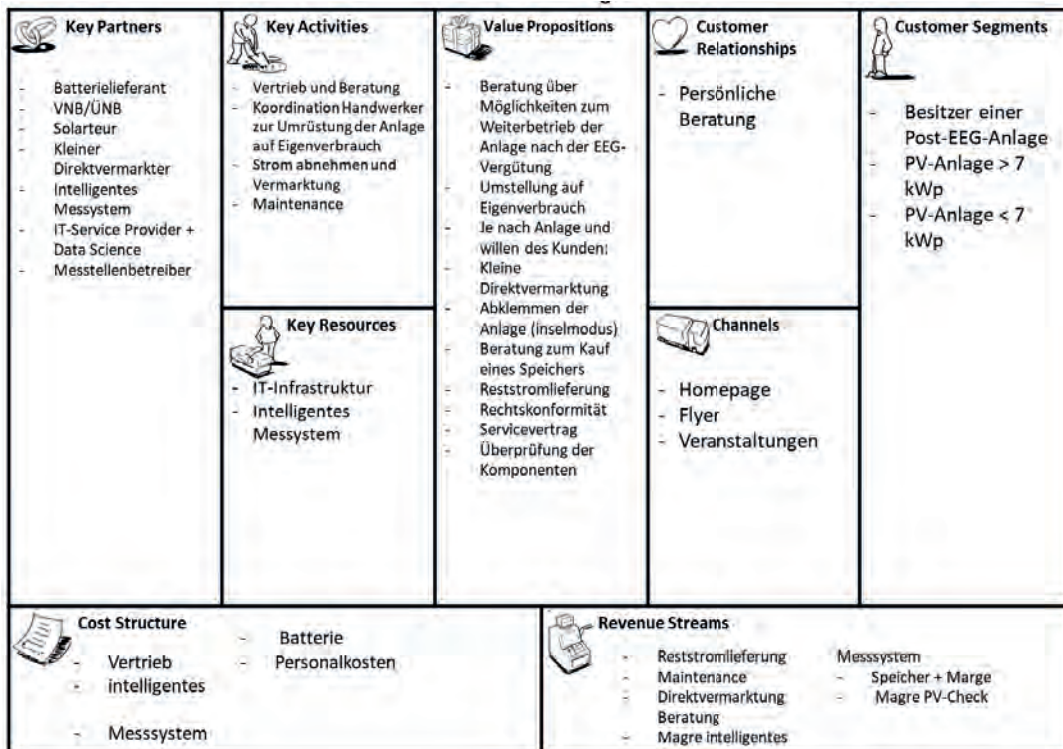


Abbildung 38: BMC des möglichen Geschäftsmodells „Kleine Direktvermarktung“

Szenario 1: Volleinspeisung

Das Geschäftsmodell, welches in Form des BMC vorliegt, wurde anhand des in Kapitel 3 beschriebenen Mappings in den BMConfig überführt. In Abbildung 39 ist die Gesamtübersicht über das Modell dargestellt. Das Modell im BMConfig gibt die angepasste Konfiguration des Geschäftsmodells wieder, welches auf Grundlage der Simulation und Diskussion mit den Partnern erstellt wurde. Im weiteren Verlauf wurden die einzelnen Austauschbeziehungen dargestellt und beschrieben. Die grundlegenden Parameter für die Berechnung der Szenarien sind:

- Das Geschäftsmodell wird für einen Zeitraum von 10 Jahren simuliert.
- Es werden insgesamt 175 Neukunden in 7 Jahr akquiriert → 25 Neukunden pro Jahr.
- Nicht alle Kunden unterschreiben den Vertrag zum 01.01. eines Jahres. Aus diesem Grund werden Werte mit einem Faktor von 0,5 korrigiert.
- Zur Vereinfachung werden nur PV-Anlagen > 7kWp betrachtet, weil diese kein iMSys benötigen.

- Zu Unternehmenssteuern und eventuellen Abschreibungen werden die Value Exchanges nicht explizit beschrieben. Diese können der **Fehler! Verweisquelle konnte nicht gefunden werden.** entnommen werden.

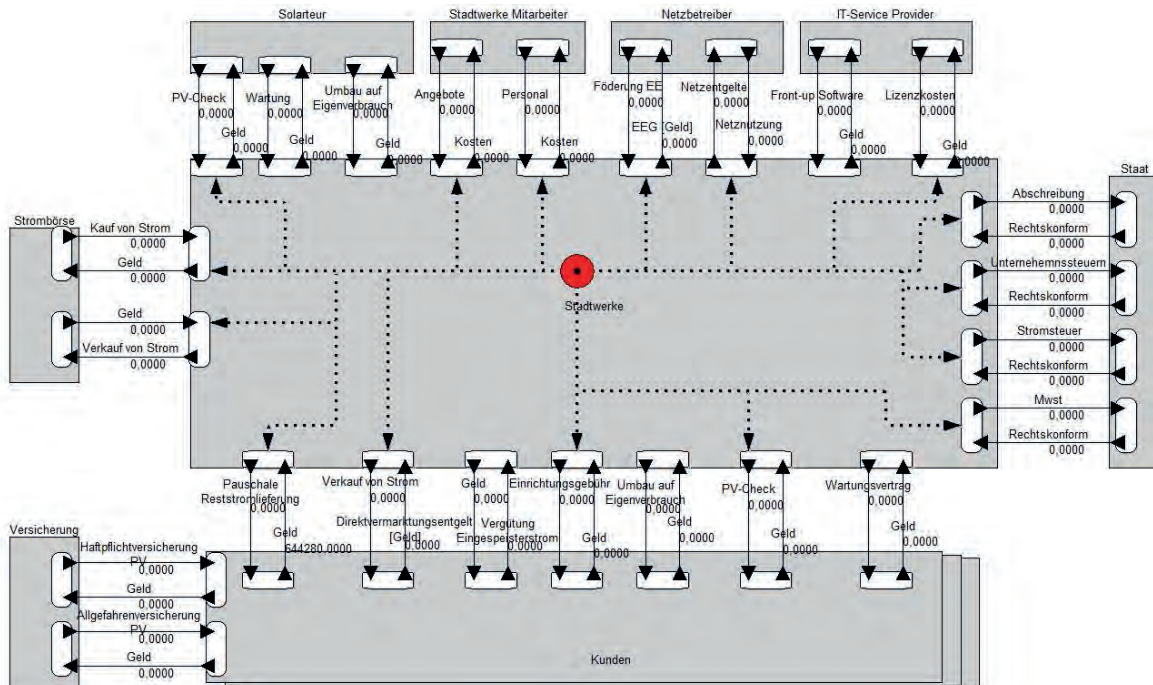


Abbildung 39: Volleinspeisung kleine Direktvermarktung

Der Akteur Solarteur ist einer der Schlüsselpartner und führt eine Überprüfung der PV-Anlage durch. Der PV-Check wird vom Solarteur eingekauft und wird mit einer Marge von 10% an einen Kunden verkauft. In Tabelle 9, Spalte 3 ist der PV-Check mit konkreten Zahlen nochmals dokumentiert.

Für die Durchführung der kleinen Direktvermarktung wird eine entsprechende Software eingekauft. Hierbei werden Lizenzkosten von 4.000 €/a angenommen. Dabei sind noch weitere Recherchen notwendig, um eine tatsächlich den Bedarf einer Software zu eruieren und eine verbesserte Kosteneinschätzung zu ermöglichen.

An den Staat fließen zur Vereinfachung alle Kosten pro kWh Strom, die nicht an den Netzbetreiber abgeführt werden müssen. Dabei werden vom Strompreis von 0,29 € pro kWh, 0,18 € pro kWh an den Staat in Form von Steuern und Abgaben abgeführt. In Tabelle 9 sind die einzelnen Posten dargestellt. Pro Haushalt wird mit einem durchschnittlichen Strombezug von 4.000 kWh/a gerechnet.

Aus der Beschreibung der kleinen Direktvermarktung konnten keine Informationen zu Austauschbeziehungen mit Geld zwischen Kunde oder Stadtwerk und dem Netzbetreiber gezogen werden. Der Kunde wird von den Stadtwerken beim Netzbetreiber angemeldet. Die Stadtwerke führen den Betrag von 0,06 € an den Netzbetreiber für Netzentgelt im Rahmen der Reststromlieferung ab. Die Kommunikation zwischen Netzbetreiber und Stadtwerke ist digitalisiert. Der Netzbetreiber hat Zugriff auf die relevanten Stammdaten. Die anfallenden Kosten über die einzelnen Jahre sind in Tabelle 9 dargestellt.

Die Kosten für die Beschaffung der elektrischen Energie auf Seiten der Stadtwerke wird mit 0,02 €/kWh angenommen. In Tabelle 9 sind die einzelnen Parameter spezifiziert. Somit belaufen sich die Gesamtkosten für eine kWh auf 0,26 €/kWh.

Für den Vertrieb fallen Personalkosten und Kosten für eine Homepage an. Über die Homepage kann ein Vertrag via Self-Service abgeschlossen werden. Hierbei wird von einem Betrag von 80 € pro Kunde ausgegangen. Die Personalkosten entstehen durch zusätzliche Beratungen etc. Es wird angenommen, dass aus drei Angeboten ein Auftrag resultiert. Die Kosten für den Vertrieb belaufen sich somit auf 6.000 € im Jahr. In Tabelle 9 ist der Sachverhalt nochmals dargestellt.

Im Rahmen der Kundenbetreuung oder sonstiger Aktivitäten werden Personalkosten von 15 Minuten pro Kunde angenommen. Für eine Stunde werden Kosten von 80 € veranschlagt. In Tabelle 9 sind die Personalkosten nochmals aufgeführt.

Der Kunde erhält von den Stadtwerken, die in diesem Fall als Direktvermarkter auftreten, die Erlöse aus dem verkauften Strom. Zur Vereinfachung der Simulation wird von einem konstanten Ertrag von durchschnittlich 0,04 € pro kWh ausgegangen. Die Größe der PV-Anlage wird auf 7 kWp festgelegt. Es wird angenommen, dass 7.000 kWh zum Verkauf durch den Direktvermarkter zur Verfügung stehen.

Tabelle 9 Kleine Direktvermarktung Volleinspeisung. Die Tabelle wird bis Jahr sieben dargestellt, die übrigen Simulationsjahre sind durch die drei Punkte dargestellt.

Jahre		Solarteure PV-Check	IT-Service-Provider Lizenzkosten	Staat Abgabe Strom Strompreis	Abgabe Netzbetreiber	Stadtwerke Strombeschaffung Einkauf Börse; Mitarbeiter, etc. 0,02 € / kWh	Vertrieb Personalkosten + Homepage	Eingespeicherter Strom ca. 0,04 € bei 7 kWp Anlage
	Preis	270 €	4.000 €	0,18 €	0,06 €	0,02 €	80 €	0,04 €
	Marktvolumen	175					175	
	Stückzahl	1	1	4000 kWh	4000	4000 kWh	6.000 €	7000 kWh
1		6.750 €	4.000 €	9.000 €	3.000 €	1.000 €	6.000 €	3.500 €
2		6.750 €	4.000 €	27.000 €	9.000 €	3.000 €	6.000 €	10.500 €
3		6.750 €	4.000 €	45.000 €	15.000 €	4.000 €	6.000 €	17.500 €
4		6.750 €	4.000 €	63.000 €	21.000 €	6.000 €	6.000 €	24.500 €
5		6.750 €	4.000 €	81.000 €	27.000 €	8.000 €	6.000 €	31.500 €
6		6.750 €	4.000 €	99.000 €	33.000 €	10.000 €	6.000 €	38.500 €
7		6.750 €	4.000 €	117.000 €	39.000 €	12.000 €	6.000 €	45.500 €
...			4.000 €	126.000 €	42.000 €	14.000 €		49.000 €
	Gesamt	47.250 €	40.000 €	919.000 €	273.000 €	86.000 €	42.000 €	318.500 €

Die Stadtwerke erzielen Erlöse durch die Reststromlieferung an die Kunden. Dabei wird von 4.000 kWh pro Haushalt pro Jahr ausgegangen. Die Akquise der 25 Neukunden verteilt sich über ein Jahr. Daher wird die Berechnung mit einem Korrekturfaktor von 0,5 multipliziert. Die Direktvermarktungspauschale wird auf 90 € pro Jahr festgelegt. Hierbei erfolgt ebenfalls eine Korrektur der Anzahl der Neukunden um 0,5. Als Einrichtungsgebühr, z.B. die Anmeldung der Anlage beim Netzbetreiber etc. fällt einmalig ein Betrag von 30 € für den Kunden an. Zusätzlich können Einnahmen durch die Vermittlung des Solarteurs zur Überprüfung der PV-Anlage (PV-Check) generiert werden. Der von den PV-Anlagen zur Verfügung gestellte Strom wird mit 0,045 €/kWh im Durchschnitt an der Strombörse verkauft. Die einzelnen Einnahmen sind in Tabelle 10 dargestellt.

Tabelle 10 Erlöse kleine Direktvermarktung Volleinspeisung. Die Tabelle wird bis Jahr sieben dargestellt, die übrigen Simulationsjahre sind durch die drei Punkte dargestellt.

Jahre		Reststromlieferung	Direktvermarktung	Einrichtungsgebühr	PV-Check	Verkauf von Strom
	Preis	0,3 €	90 €	30 €	297 €	0,045 €
	Marktvolumen					
	Stückzahl	4000 kWh				7000 kWh
1		14.750 €	1.125 €	750 €	7.425 €	3.937,5 €
2		44.250 €	3.375 €	750 €	7.425 €	11.812,5 €
3		73.750 €	5.625 €	750 €	7.425 €	19.687,5 €
4		103.250 €	7.875 €	750 €	7.425 €	27.562,5 €
5		132.750 €	10.125 €	750 €	7.425 €	35.437,5 €
6		162.250 €	12.375 €	750 €	7.425 €	43.312,5 €
7		191.750 €	14.625 €	750 €	7.425 €	51.187,5 €
...		206.500 €	15.750 €			55.120 €
	Gesamt	1.342.250 €	102.375 €	5.250 €	51.975 €	358.312,5 €

Für den Kunden fallen bei diesem Geschäftsvorhaben einmalig die Überprüfung der PV-Anlage in Form des PV-Checks und die Einrichtungsgebühr an. Die Versicherung und das Direktvermarktungsentgelt sind jährlich anfallende Kosten. Die Vergütung des Stromes liegt bei 280 € jährlich. Damit hätte der Kunde nach drei Jahren einen Break-even erreicht und kann seine Post-EEG PV-Anlage ökonomisch sinnvoll weiterbetreiben. In Tabelle 11 sind die Ergebnisse eines exemplarischen Kunden nochmals zusammengefasst.

Tabelle 11: Übersicht Kosten und Erlöse Kunde kleine Direktvermarktung Volleinspeisung

Jahr	1	2	3	Gesamt:
Kosten:				
PV-Check	297 €	0	0	
Einrichtungsgebühr	30 €			
Versicherung	66 €	66 €	66 €	
Direktvermarktungsentgelt	90 €	90 €	90 €	
Gesamt:	483 €	156 €	156 €	795 €
Erlöse:				
Vergütung Strom	280 €	280 €	280 €	840 €

In Abbildung 40 und Tabelle 12 sind die Ergebnisse der kleinen Direktvermarktung zusammengefasst. Bei der aktuellen Konfiguration erzielen die Stadtwerke ein Plus von ca. 39.000 € nach 10 Jahren. Diese Konfiguration benötigt weitere Untersuchungen, z.B. inwieweit organisatorische Veränderungen notwendig sind. Auch müsste die Form der organisatorischen Verankerung des Geschäftsmodelles diskutiert werden. Hierbei gibt es prinzipiell zwei Möglichkeiten der Ausgründung oder der internen Verankerung in der Organisation. Diese Untersuchungen benötigen mehr Zeit und konnten nicht mehr im Projektzeitraum untersucht werden.

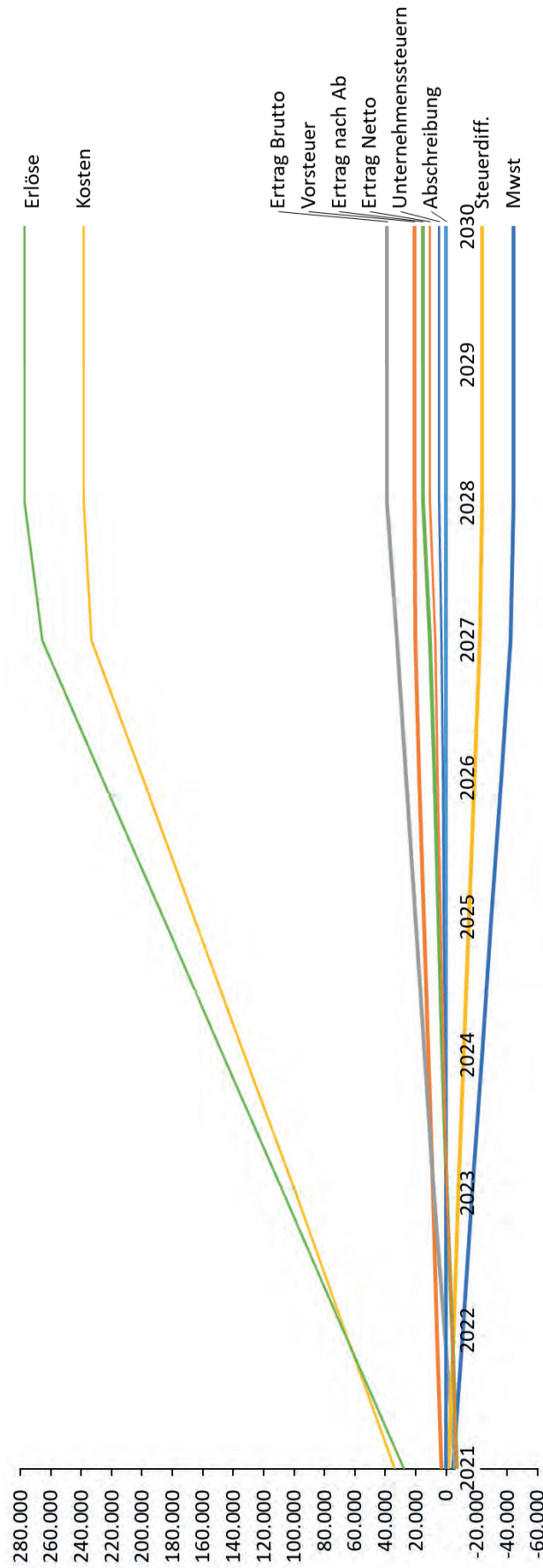


Abbildung 40 Übersicht Kosten und Erlöse Stadtwerke kleine Direktvermarktung Volleinspeisung

Tabelle 12 Übersicht Stadtwerke Kosten und Erlöse kleine Direktvermarktung Volleinspeisung

	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	Gesamt
Jahr	1	2	3	4	5	6	7	8	9	10	
Jahr in Betrieb	25	25	50	75	100	125	150	175	175	175	
Neuausbringung	25	25	25	25	25	25	25				
Kosten	33.750,00 €	66.750,00 €	99.250,00 €	132.750,00 €	166.250,00 €	199.750,00 €	233.250,00 €	238.500,00 €	238.500,00 €	238.500,00 €	1.647.250,00 €
Erlöse	27.987,50 €	67.612,50 €	107.237,50 €	146.862,50 €	186.487,50 €	226.112,50 €	265.737,50 €	277.375,00 €	277.375,00 €	277.375,00 €	1.860.162,50 €
Mwst	-4.468,59 €	-10.795,27 €	-17.121,95 €	-23.448,63 €	-29.775,32 €	-36.102,00 €	-42.428,68 €	-44.286,76 €	-44.286,76 €	-44.286,76 €	-297.000,74 €
Vorsteuer	3.153,36 €	6.027,31 €	8.901,26 €	11.775,21 €	14.649,16 €	17.523,11 €	20.397,06 €	20.756,30 €	20.756,30 €	20.756,30 €	144.695,38 €
Ertrag Brutto	-5.762,50 €	862,50 €	7.987,50 €	14.112,50 €	20.237,50 €	26.362,50 €	32.487,50 €	38.875,00 €	38.875,00 €	38.875,00 €	212.912,50 €
Steuerdiff.	-1.315,23 €	-4.767,96 €	-8.220,69 €	-11.673,42 €	-15.126,16 €	-18.578,89 €	-22.031,62 €	-23.530,46 €	-23.530,46 €	-23.530,46 €	-152.305,36 €
Abschreibung	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €
Ertrag nach Ab 30%	-7.077,73 €	-3.905,46 €	-233,19 €	2.439,08 €	5.111,34 €	7.783,61 €	10.455,88 €	15.344,54 €	15.344,54 €	15.344,54 €	60.607,14 €
Ertrag Netto	0,00 €	0,00 €	0,00 €	731,72 €	1.533,40 €	2.335,08 €	3.136,76 €	4.603,36 €	4.603,36 €	4.603,36 €	21.547,06 €
	-7.077,73 €	-3.905,46 €	-233,19 €	1.707,35 €	3.577,94 €	5.448,53 €	7.319,12 €	10.741,18 €	10.741,18 €	10.741,18 €	39.060,08 €

Szenario 2 Eigenverbrauch mit Überschusseinspeisung

Aus einer Beschreibung eines Geschäftsmodells in Form eines BMC können eine Vielzahl unterschiedlicher Ausprägungen des Geschäftsvorhabens orchestriert und verschiedene Business Cases abgeleitet werden. Das Geschäftsmodell, welches in Form des BMC vorliegt, wurde anhand des in Kapitel 3 beschriebenen Mappings in den BMConfig überführt. In Abbildung 41 ist die Gesamtübersicht über das Modell in einem alternativen Szenario dargestellt. Im weiteren Verlauf werden die einzelnen Austauschbeziehungen dargestellt und beschrieben. Das Modell im BMConfig gibt die angepasste Konfiguration des Geschäftsmodells wieder, welches auf Grundlage der Simulation und Diskussion mit den Partnern erstellt wurde. Die grundlegenden Parameter für die Berechnung der Szenarien sind.

- Das Geschäftsmodell wird für einen Zeitraum von 10 Jahren simuliert.
- Es werden insgesamt 175 Neukunden in 7 Jahr akquiriert → 25 Neukunden pro Jahr.
- Nicht alle Kunden unterschreiben den Vertrag zum 01.01. eines Jahres. Aus diesem Grund werden Werte mit einem Faktor von 0,5 korrigiert.
- Zur Vereinfachung werden nur PV-Anlagen > 7kWp betrachtet, weil diese kein iMSys benötigen.
- Zu Unternehmenssteuern und eventuellen Abschreibungen werden die Value Exchanges nicht explizit beschrieben. Diese können Abbildung 41 entnommen werden.

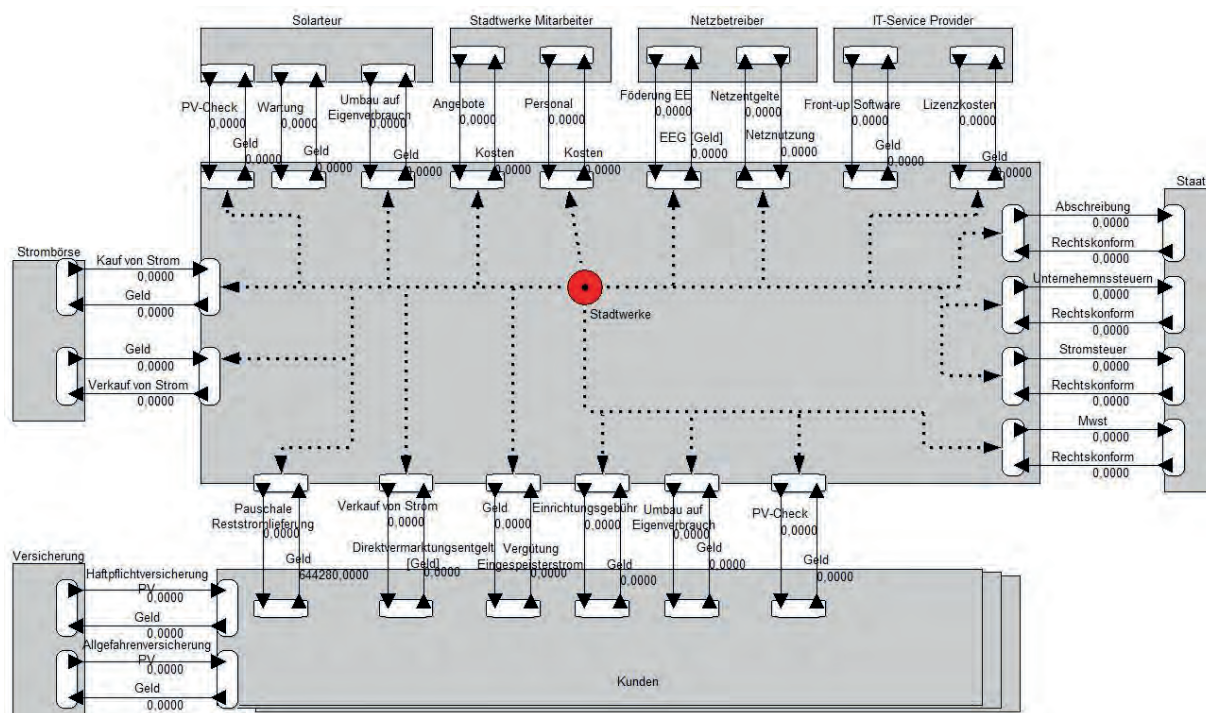


Abbildung 41: Eigenverbrauch kleine Direktvermarktung

Der Akteur Solarteure ist für den Umbau und die Überprüfung der Anlage zuständig. Zur Umrüstung auf Eigenverbrauch wird den Kunden ein Solarteure vermittelt. Die Stadtwerke kaufen die Leistung für 400 € bei einem Solarteure im Durchschnitt ein. In Tabelle 13 sind die einzelnen Posten nochmals dargestellt. Auch die einzelnen Value Exchanges zwischen den Akteuren und den Stadtwerken sind in Tabelle 13 dargestellt. Sie sind identisch zu den in Szenario 1 beschriebenen Value Exchanges.

An den Staat fließen zur Vereinfachung alle Kosten pro kWh Strom, die nicht an den Netzbetreiber abgeführt werden müssen. Dabei werden vom Strompreis von 0,30 € pro kWh 0,18 € pro kWh an den Staat in Form von Steuern und Abgaben abgeführt. Die Reststromlieferung beträgt dabei durchschnittlich 2.400 kWh pro Haushalt im Jahr. Dieser Wert ergibt sich bei einem Ertrag von 7.000 kWh bei einer 7 kWp PV-Anlage und einem Eigenverbrauch von 30 %. Dies entspricht einem Eigenverbrauch von 2.100 kWh. Bei einem jährlichen Strombedarf von 4.500 kWh abzüglich des Eigenverbrauchs von 2.100 kWh ergibt sich eine Reststromlieferung von 2.400 kWh. In Tabelle 13 sind die Kosten gegenüber dem Staat für Steuern und Abgaben nochmals aufgelistet.

Aus der Beschreibung der kleinen Direktvermarktung konnten keine Informationen zu den Austauschbeziehungen mit Geld zwischen Kunde oder Stadtwerk und dem Netzbetreiber gezogen werden. Der Kunde wird von den Stadtwerken beim Netzbetreiber angemeldet. Die Stadtwerke führen den Betrag von 0,06 € an den Netzbetreiber für Netzentgelte im Rahmen der Reststromlieferung ab. Die Kommunikation zwischen Netzbetreiber und Stadtwerke ist digitalisiert. Der Netzbetreiber hat Zugriff auf die relevanten Stammdaten. Die anfallenden Kosten über die einzelnen Jahre sind in Tabelle 13 dargestellt.

Die Kosten für die Beschaffung der elektrischen Energie auf Seiten der Stadtwerke (interne Kosten der Stadtwerke) wird mit 0,02 €/kWh angenommen. Auch diese Parameter sind in Tabelle 13 spezifiziert. Somit belaufen sich die Gesamtkosten für eine kWh auf 0,26 €/kWh.

Der Kunde erhält von den Stadtwerken, die in diesem Fall als Direktvermarkter auftreten, die Erlöse aus dem verkauften Strom. Zur Vereinfachung der Simulation wird von einem konstanten Ertrag von durchschnittlich 0,04 € pro kWh ausgegangen. Die Größe der PV-Anlage wird auf 7 kWp festgelegt. Es wird angenommen, dass 30% des Stromes selbst verbraucht wird. Somit stehen den Stadtwerken ca. 4.000 kWh/a zum Verkauf zu Verfügung.

Tabelle 13 Kunden kleine Direktvermarktung. Die Tabelle wird bis Jahr sieben dargestellt, die übrigen Simulationsjahre sind durch die drei Punkte dargestellt.

Jahre		Solarteure PV-Check	Umbau PV-Anlage auf Eigenbedarf	Staat Angaben Strompreis	Abgaben Netzbetreiber	Stadtwerke Strombeschaffung Einkauf Börse; Mitarbeiter, etc. 0,02 €/kWh	Eingespeister Strom ca. 0,04 € mit 7 kWp Anlage bei 30% Eigenverbrauch
	Preis	270 €	400 €	0,18 €	0,06 €	0,02 €	0,04 €
	Marktvolumen	175	175				
	Stückzahl	1	1	2400 kWh	2400 kWh	2400 kWh	4.000 kWh
1		6.750 €	10.000 €	5.400 €	1.800 €	600 €	2.000 €
2		6.750 €	10.000 €	16.200 €	5.400 €	1.800 €	6.000 €
3		6.750 €	10.000 €	27.000 €	9.000 €	2.400 €	10.000 €
4		6.750 €	10.000 €	37.800 €	12.600 €	3.600 €	14.000 €
5		6.750 €	10.000 €	48.600 €	16.200 €	4.800 €	18.000 €
6		6.750 €	10.000 €	59.400 €	19.800 €	6.000 €	22.000 €
7		6.750 €	10.000 €	70.200 €	23.400 €	7.200 €	26.000 €
...				75.600 €	25.200 €	8.400 €	28.000 €
	Gesamt	47.250 €	70.000 €	491.400 €	163.800 €	51.600 €	182.000 €

Die Stadtwerke erzielen Erlöse durch die Reststromlieferung an die Kunden. Dabei wird von 2.400 kWh pro Haushalt im Jahr ausgegangen. Die Akquise der 25 Neukunden verteilt sich über ein Jahr. Daher wird die Berechnung mit einem Korrekturfaktor von 0,5 multipliziert. Die

Direktvermarktungspauschale wird auf 90 € pro Jahr festgelegt. Hierbei erfolgt ebenfalls eine Korrektur der Anzahl der Neukunden um 0,5. Als Einrichtungsgebühr z.B. die Anmeldung der Anlage beim Netzbetreiber etc. fällt einmalig ein Betrag von 30 € für den Kunden an. Zusätzlich können Einnahmen durch die Vermittlung des Solarteurs zur Überprüfung der PV-Anlage (PV-Check) generiert werden. Der von den PV-Anlagen zur Verfügung gestellte Strom wird mit 0,045 €/kWh im Durchschnitt an der Strombörse verkauft Die einzelnen Einnahmen sind in Tabelle 14 dargestellt.

Tabelle 14 Erlöse kleine Direktvermarktung kleine Direktvermarktung Eigenverbrauch.

Jahre		Reststrom-lieferung	Direkt-vermarktungs-entgelt	Einrichtungs-gebühr	PV-Check	Verkauf von Strom	Umbau auf Eigen-verbrauch
	Preis	0,3 €	90 €	30 €	297 €	0,045 €	440 €
	Markt-volumen						
	Stückzahl	2400 kWh				4000 kWh	
1		8.850 €	1.125 €	750 €	7.425 €	2.250 €	11.000 €
2		26.550 €	3.375 €	750 €	7.425 €	6.750 €	11.000 €
3		44.250 €	5.625 €	750 €	7.425 €	11.250 €	11.000 €
4		61.950 €	7.875 €	750 €	7.425 €	15.750 €	11.000 €
5		79.650 €	10.125 €	750 €	7.425 €	20.250 €	11.000 €
6		97.350 €	12.375 €	750 €	7.425 €	24.750 €	11.000 €
7		115.050 €	14.625 €	750 €	7.425 €	29.250 €	11.000 €
...		123.900 €	15.750 €			31.500 €	
	Gesamt	805.350 €	102.375 €	5.250 €	51.975 €	204.750 €	77.000 €

Für den Kunden fallen bei diesem Geschäftsvorhaben einmalig die Überprüfung der PV-Anlage in Form des PV-Checks, der Umbau der PV-Anlage auf Eigenverbrauch und die Einrichtungsgebühr an. Die Versicherung und das Direktvermarktungsentgelt sind jährlich anfallende Kosten. Die Vergütung des Stromes liegt bei 160 € jährlich. Die Stromvermeidung durch den Eigenverbrauch liegt bei 790 €. In der nachfolgenden Tabelle 15 ist der Sachverhalt nochmals zusammengefasst.

Tabelle 15: Übersicht Kosten und Erlöse Kunde kleine Direktvermarktung Eigenverbrauch

Jahr	1	2	3	Gesamt:
Kosten:				
PV-Check	297 €	0	0	
Einrichtungsgebühr	30 €	0		
Versicherung	66 €	66 €	66 €	
Direktvermarktungsentgelt	90 €	90 €	90 €	
Umbau auf Eigenverbrauch	440 €			
Gesamt:	913 €	156 €	156 €	1235 €
Erlöse:				
Vergütung Strom	160 €	160 €	160 €	
Stromvermeidung	630 €	630 €	630 €	
Gesamt:	790 €	790 €	790 €	2370 €

In Abbildung 42 und in Tabelle 16 sind die Ergebnisse der kleinen Direktvermarktung zusammengefasst. Bei der aktuellen Konfiguration erzielen die Stadtwerke ein Plus von ca. 25.000 € nach 10 Jahren. Im Gegensatz zu Szenario 1 Volleinspeisung, bei welchem ein Gewinn von 39.000 € erzielt werden konnte. Für den Kunden scheint das Eigenverbrauchsmodell mit Überschusseinspeisung von Vorteil zu sein.

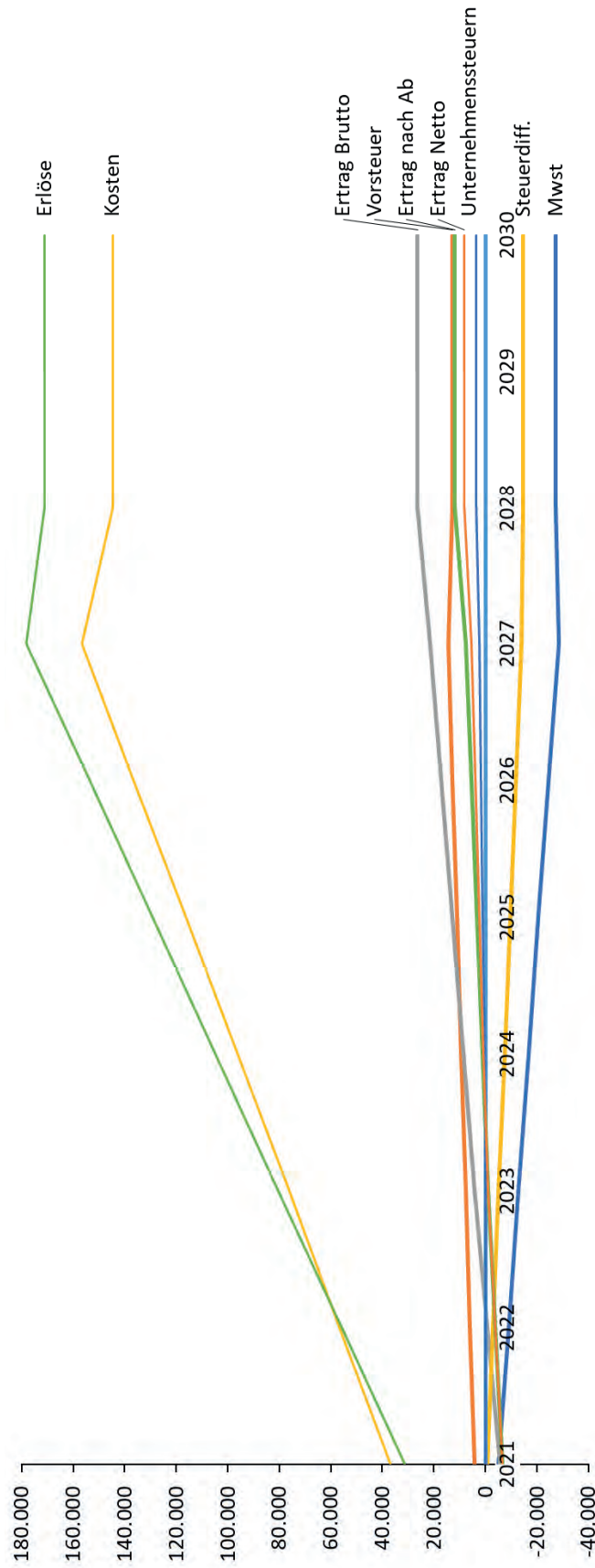


Abbildung 42: Übersicht Stadtwerke kleine Direktvermarktung Eigenverbrauch

Tabelle 16 Übersicht kleine Direktvermarktung Eigenverbrauch

	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	Gesamt
Jahr											
Jahr in Betrieb	1	2	3	4	5	6	7	8	9	10	
Neuausbringung	25	25	50	75	100	125	150	175	175	175	
Kosten	37.050,00 €	56.650,00 €	76.150,00 €	96.250,00 €	116.350,00 €	136.450,00 €	156.550,00 €	144.700,00 €	144.700,00 €	144.700,00 €	1.109.550,00 €
Erlöse	31.400,00 €	55.850,00 €	80.300,00 €	104.750,00 €	129.200,00 €	153.650,00 €	178.100,00 €	171.150,00 €	171.150,00 €	171.150,00 €	1.246.700,00 €
Mwst	-5.013,45 €	-8.917,23 €	-12.821,01 €	-16.724,79 €	-20.628,57 €	-24.532,35 €	-28.436,13 €	-27.326,47 €	-27.326,47 €	-27.326,47 €	-199.052,94 €
Vorsteuer	4.175,21 €	5.899,58 €	7.623,95 €	9.348,32 €	11.072,69 €	12.797,06 €	14.521,43 €	12.709,24 €	12.709,24 €	12.709,24 €	103.565,97 €
Ertrag Brutto	-5.650,00 €	-800,00 €	4.150,00 €	8.500,00 €	12.850,00 €	17.200,00 €	21.550,00 €	26.450,00 €	26.450,00 €	26.450,00 €	137.150,00 €
Steuerdiff.	-838,24 €	-3.017,65 €	-5.197,06 €	-7.376,47 €	-9.555,88 €	-11.735,29 €	-13.914,71 €	-14.617,23 €	-14.617,23 €	-14.617,23 €	-95.486,97 €
Abschreibung	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €	0,00 €
Ertrag nach Ab 30%	-6.488,24 €	-3.817,65 €	-1.047,06 €	1.123,53 €	3.294,12 €	5.464,71 €	7.635,29 €	11.832,77 €	11.832,77 €	11.832,77 €	41.663,03 €
Ertrag Netto	-6.488,24 €	-3.817,65 €	-1.047,06 €	786,47 €	2.305,88 €	3.825,29 €	5.344,71 €	8.282,94 €	8.282,94 €	8.282,94 €	25.758,24 €

6. AP 5 – Netzdienliche Nutzung der Kundenanlagen und der Biogasanlage mit Wärmenetz

Aus Zeit- und Kapazitätsgründen erfolgt dieser Teil im Folgeprojekt SeLiG. Die Dokumentation der Simulation wird entsprechend im Endbericht von SeLiG dokumentiert.

7. Anschlussfähigkeit und Übertragbarkeit

Es wurden durch zahlreiche Maßnahmen die Anschlussfähigkeit und Übertragbarkeit sowie eine Verwertung der Ergebnisse gewährleistet. Hierbei wurden die entwickelten Lösungen in anderen Projekten der Hochschule Aalen eingesetzt bzw. weitergenutzt. Die nachfolgende Tabelle 17 gibt hierzu einen Überblick.

Tabelle 17: Zusammenstellung der Maßnahmen zur Anschlussfähigkeit und Übertragbarkeit sowie der Verwertung der Ergebnisse

Maßnahme	Umsetzung	Potenzieller Mehrwert (ggf. durch weitere Verwertung)
Einsatz in der Lehre an der Hochschule Aalen	Einbringung der Erfahrungen und Ergebnisse in die Vorlesungen „Energiewirtschaft“ und „Energiekonzepte“	Studierende lernen aktuelle Forschungsfragen kennen, sie verstehen welche Probleme im Zuge der Energiewende auftreten
Einsatz in der Öffentlichkeitsarbeit an der Hochschule Aalen	Postererstellung (vgl. hierzu Abbildung 43) und Präsentation der Poster bei Schulbesuchen (insbesondere in 2019) sowie öffentlichen Veranstaltungen (z.B. Reichstädter Tage in 2019).	Breite Bekanntmachung der Forschungsfragen, da viel Laufpublikum bei öffentlichen Veranstaltungen; Schaffung von Interesse zu den Themen Energieerzeugung, Energieübertragung, Smart Grids
Öffentlichkeitsarbeit in lokalen Zeitungen	Veröffentlichungen von Pressemeldungen	Weitere Bekanntmachung bei Pilotkunden, lokalen Akteuren und Entscheidern sowie politischen Entscheidungsträgern
wissenschaftliche Veröffentlichungen	Publikation auf Konferenzen und in Fachzeitschriften (siehe zitierte Quellen in AP 1)	Bekanntmachung der Forschungsfragen in der internationalen wissenschaftlichen Community
Einsatz in SeLiG (Demonstrationsprojekt Second-Life Speicher in Smart Grids, Zuwendungsnummer BWSGD 19011)	Nutzung der Daten der Pilotkunden als Grundlage einer Steuerung des Netzspeichers	Datenbasis und Blaupause für weiterführende Fragestellungen und Lösungsansätze in SeLiG
Einsatz in KEFF und RePho	Einsatz des Online Monitoring Systems bei Photovoltaikanlagen bei entsprechenden Fragestellungen der KEFF- und RePho-Kunden (z.B. Monitoring der PV-Anlage bei Unstimmigkeiten bei der Einspeisung)	Mehrwert und Näherbringung der Thematik bei Industriekunden und kommunalen Unternehmen (z.B. Stadtwerke); mögliche Grundlage für weitere Forschungsprojekte, wie beispielsweise ZIM

Literaturverzeichnis

- [1] „Requests: HTTP for Humans,“ [Online]. Available: <https://2.python-requests.org/en/master/>. [Zugriff am 22 Juli 2021].
- [2] P. Wangxu, „Github: Nextpertise / persist-queue,“ Github, [Online]. Available: <https://github.com/Nextpertise/persist-queue>. [Zugriff am 22 Juli 2021].
- [3] influxdata, „influxdata,“ influxdata, [Online]. Available: <https://www.influxdata.com/>. [Zugriff am 22 Juli 2021].
- [4] NGINX, „NGINX,“ NGINX, 2021. [Online]. Available: <https://www.nginx.com/>. [Zugriff am 22 Juli 2021].
- [5] certbot, „certbot,“ certbot, 2021. [Online]. Available: <https://certbot.eff.org/>. [Zugriff am 22 Juli 2021].
- [6] Let's Encrypt, „Let's Encrypt,“ Let's Encrypt, 2021. [Online]. Available: <https://letsencrypt.org/>. [Zugriff am 22 Juli 2021].
- [7] Grafana, „Grafana,“ Grafana, 2021. [Online]. Available: <https://grafana.com/>. [Zugriff am 22 Juli 2021].
- [8] S. M. Mostafa, „Imputing missing values using cumulative linear regression,“ *CAAI Transactions on Intelligende Technology*, 2019.
- [9] J. Jerez, I. Molina, P. García Laencina, E. Alba, N. Ribelles, M. Martin und L. Franco, „Missing Data Imputation Using Statistical and Machine Learning Methods in a Real Breast Cancer Problem,“ *Artificial Intelligence in Medicine 50*, pp. 105-115, October 2010.
- [10] J. Bulhoes, A. Assis, C. Martins, G. Furriel, B. Silva, L. Rodrigues, M. Reis, D. Calheiros, M. Oliveira und W. Calixto, „Gap filling in time series: A new methodology applying spectral analysis and system identification,“ in *CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, 2017.
- [11] J. Peirong, X. Di, W. Peng und C. Juan, „A Study on Exponential Smoothing Model for Load Forecasting,“ in *Conference: Power and Energy Engineering Conference (APPEEC)*, Asia-Pacific, March 2012.
- [12] H. Verma und S. Kumar, „An accurate missing data prediction method using LSTM based deep learning for health care,“ in *Proceedings of the 20th International Conference on Distributed Computing and Networking.*, 2019.

- [13] W. Velicer und S. Colby, „A Comparison of Missing-Data Procedures for Arima Time-Series Analysis,“ *Educational and Psychological Measurement* 65, pp. 596-615, 2005.
- [14] <https://www.datatechnotes.com/regression-model-accuracy>.
- [15] B. F. Farsi, M. A. Amayri, N. B. Bouglulia und Eicker, „On Short-Term Load Forecasting Using Machine Learning Techniques and a Novel Parallel Deep LSTM-CNN Approach,“ *IEEE Access*, 2021.
- [16] J. Akkermans und J. Gordjin, „Value-based requirements engineering: exploring innovative e-commerce ideas,“ Springer, 2003.
- [17] A. Osterwalder, „Business Model Generation,“ John Wiley & Sons, New Jersey, 2010.
- [18] A. Thompson und I. MacMillan, „Business models: Creating new markets and societal wealth,“ Elsevier Ltd., 2010.
- [19] C. Koppenhoeffler, J. Fauser und D. Hertweck, „'Multi-Model-Approach Towards Decentralized Corporate Energy Systems', in Otjacques, B. et al. (eds) From Science to Society: New Trends in Environmental Informatics,“ Springer International Publishing, Böblingen, 2018.
- [20] A. Caetano, „Representation and analysis of enterprise models with semantic techniques: an application to ArchiMate, e3value and business model canvas,“ Springer, London, 2017.
- [21] L. Meertens, „Relating business modelling and enterprise architecture,“ University of Twente, Twente, 2013.
- [22] Digitrans, „Digitrans,“ MFG Medien- und Filmgesellschaft Baden-Württemberg mbH, 2018. [Online]. Available: <https://digitrans.me/psm/home> (Accessed: 10 February 2020). [Zugriff am 22 Juli 2021].
- [23] A. Osterwalder, „Value Proposition Design: How to Create Products and Services Customers Want,“ John Wiley & Sons, 2014.
- [24] D. Parra, „Optimum community energy storage system for PV energy timeshift,“ Elsevier Inc., Nottingham, 2015.
- [25] S. C. Müller, Welppe und I.M., „Quartierspeicher: Überblick über mögliche Geschäftsmodelle und Barrieren,“ TU München, München, 2017.
- [26] M. Engelken, „Comparing drivers, barriers and opportunities of business models for renewable energies: A review,“ 2016.

- [27] EnBausa.de, „Eigenverbrauch ist eine option für ältere PV-Anlagen,“ EnBausa, 20 Januar 2020. [Online]. Available: <https://www.enbausa.de/solarenergie/aktuelles/artikel/eigenverbrauch-ist-option-fuer-aeltere-pv-anlagen-6792.html>. [Zugriff am 22 Juli 2021].

Anhang

NEIS 2019: „Online monitoring system for photovoltaic systems using anomaly detection with machine learning“

Online monitoring system for photovoltaic systems using anomaly detection with machine learning

Moritz Benninger*, Martina Hofmann, Marcus Liebschner
 Aalen University of Applied Sciences, Beethovenstr. 1, 73430 Aalen, Germany
 *Moritz.Benninger@hs-aalen.de

Abstract

This paper presents a novel method from the field of machine learning for monitoring photovoltaic systems by detecting anomalies. Self-learning algorithms considerably reduce the measuring effort and at the same time offer reliable monitoring of occurring faults. As a prototype, a Raspberry Pi is used in combination with a contactless current sensor.

Keywords – photovoltaic systems, online monitoring system, machine learning, anomaly detection

1 Introduction

In recent years, the global photovoltaic (PV) market has grown steadily. As the fastest growing energy source, PV also offers great potential for the future [1]. The use of PV systems has many advantages, such as the use of freely available solar radiation, practically unlimited resources for the production of silicon-based cells and good scalability of electrical power [2]. However, the systems also have problems because they are susceptible to faults such as glass breakage, delamination or hotspots and also to degradation with increasing lifetime [3]. Therefore, it is very important to ensure stable and continuous operation for reliable power generation in power grids with high levels of volatile power generation. Online monitoring of PV is technically advantageous to help operators maintain their facilities and ensure economic integration into smart grids.

2 Online monitoring

Online monitoring is already widespread in the field of electrical systems. High-voltage bushings are an example of electrical components that are successfully equipped with monitoring systems [4]. Various approaches are currently available for the online monitoring of PV systems: For example, *Ahrary et al.* include the values of voltage and current in combination with irradiance in their monitoring [5]. The additional light sensor enables a comparison to be made between the power actually generated by the PV system and the power to be expected due to solar radiation. Larger differences between these two values indicate an existing problem with the PV system.

The use of artificial intelligence to monitor PV systems is state of the art. It enables the system-specific recognition and classification of individual fault causes: *Liu et al.* use an optimized backpropagation neural network that searches certain types of faults in PV systems [6]. The inputs to the neural network are electrical parameters such as current, voltage and power (especially the short circuit

current and the open circuit voltage). The output detects the state of the PV system as normal behaviour or as different types of faults such as short circuit, open circuit, partial shading and abnormal ageing. *Dhimish* tests different methods based on neural networks and fuzzy logic systems for monitoring [7]. This approach uses the electrical parameters in combination with the actual irradiance value and temperature to distinguish between normal operation, partial shading and defect modules. Commercial suppliers are also currently trying to monitor photovoltaic systems with self-learning algorithms and artificial intelligence. In Germany alone, seven companies are already known to be involved with artificial intelligence in their product development. However, there is still no satisfactory software on the market [8].

All available solutions require a considerable effort, for example for the installation of measurement technology and data processing. In the new approach presented here, online monitoring uses anomaly detection models from the field of machine learning. The enormous advantage of these special methods is that only the measurement of one electrical quantity (here the generated direct current) is required. Thus the detection of a deviation from the initial conditions of the system can be carried out with very little effort. In addition, this method works automatically and learns the differences between normal and abnormal values by itself and without supervision. This aspect allows it to be used in any PV system, because the healthy and faulty states are recognized directly from the behaviour of the system. For full functionality, however, online monitoring should be integrated into new PV systems. The artificial intelligence learns the initial state of the system best from the beginning of its life cycle.

3 Method

3.1 The behaviour of the electrical parameters of a photovoltaic system

Due to the strong dependence from solar radiation, the generated electrical current is the most important

parameter for monitoring PV systems. During a single day as well as over a whole year, different values of electric current are generated.

Looking at the current flow over a period of a few seconds, it becomes apparent that the change in the position of the sun and the movement of the clouds only have a slow effect on the current intensity generated. The time derivative of the current over the day leads to a characteristic picture of data points, as shown in **Figure 1**. The largest changes in current intensity occur at noon, when the irradiance is greatest. At times with less daylight, the changes are smaller. The direction of the changes can be either positive or negative, depending on whether the solar irradiance increases or decreases.

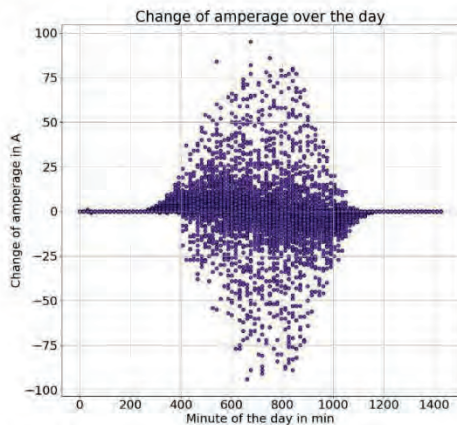


Figure 1 Time derivatives of the current over the minutes of all days of one year for a PV system with an output of 40.56 kW

The course of the maximum current strength per day over the period of one year shows the influence of the different seasons. **Figure 2** shows the differences between winter and summer. Typical for regions above the equator is the lower power generation in the months September to March compared to the months April to August. These two characteristic curves of the current intensity allow the detection of faults in different time resolutions.

3.2 Anomaly detection with machine learning

Machine learning is a part of artificial intelligence. It is the application and science of algorithms that can extract knowledge from data [9]. With such self-learning algorithms, it is possible to identify the faults of PV systems based on electrical measurement data. In the first year of operation of the PV system, data is measured to generate a set of training data. As this is the initial year of operation, it is highly probable that these measured data reflect the optimal behaviour of the system. In the training step, the self-learning algorithms learn boundaries from these data. The boundaries are used to monitor future

behaviour and divide the measurement data into normal and abnormal values. Thus, deviations from the initial state of the system can be recognized and a corresponding warning issued. The so-called short-term faults can be detected with the "*k*-Nearest-Neighbours (*k*-NN) method" and the so-called long-term faults with the "one-class support vector machine (OCSVM)" from the *PyOD Python toolbox* [10].

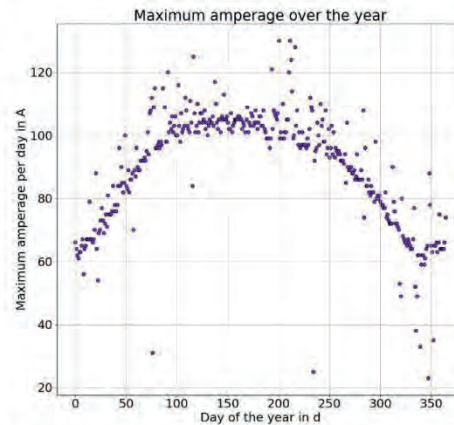


Figure 2 Maximum current per day over the days of one year for a PV system with an output of 40.56 kW

3.2.1 Short-term faults with *k*-Nearest Neighbours (*k*-NN)

The *k*-Nearest Neighbours method uses a non-parametric model that can classify individual data points in data sets. In principle, this method is used to divide data into several classes. In this approach, however, the *k*-Nearest Neighbours algorithm is applied to only one class in order to detect deviations from this class, so-called anomalies, after the training phase. [11]

For this purpose, the *k*-NN algorithm searches the next *k* data points for each data point in the training data set, using a distance metric. The Euclidean metric serves as the distance metric *d* and is described by the following equation with the test sample $x^{(i)}$ and the *k* nearest neighbours $x_k^{(j)}$ [9]:

$$d(x^{(i)}, x^{(j)}) = \sqrt{\sum_k |x_k^{(i)} - x_k^{(j)}|^2}$$

The following classification is then a majority decision between the *k* nearest neighbours of the data point under consideration. This results in a so-called decision boundary (**Figure 3**), which allows a separation into normal and abnormal observations for future measurement data. [11] In addition, a tolerance band is introduced to avoid too strict decisions by the anomaly detection. For this purpose, the values of the training data are increased by a further 10%. This leads to increased robustness. Otherwise, the

anomaly detection would already be triggered if the test sample were only slightly outside the training values. A recognition of short-term faults can be carried out with the time derivatives of the current. A short-term fault could be caused by hail or overheating of a single module. In this case, a part of the system would suddenly fail and lead to a larger current change than in normal operation. In such an error case, the measurement data is outside the learned boundary and the k -NN detects a fault.

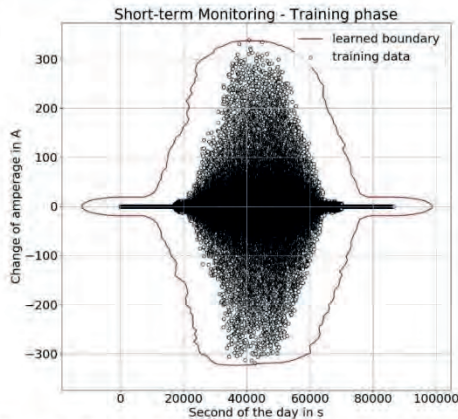


Figure 3 Time derivatives of the current over the seconds of the day as training data and the learned boundary for short-term monitoring with the k -Nearest Neighbours algorithm.

3.2.2 Long-term faults with one-class support vector machine (OCSVM)

The one-class support vector machine is a classifier that is often used to detect anomalies in data sets. The unique feature is that the algorithm can independently distinguish between normal and abnormal observations from previously unknown data points of a single class. The main goal is to achieve the largest possible margin between the class boundaries. [12]

In order to separate the anomalies from the normal observations, the corresponding data set is projected into a higher dimension. There, the algorithm searches for an optimal hyperplane, which separates a previously defined proportion of data points from the data set at a maximum distance from the origin. This hyperplane is represented by the following equation, where x are the data values, w is the weight vector and b is the bias [11]:

$$w^T x + b = 0$$

This hyperplane is then projected back into the original dimension together with the data set, resulting in a nonlinear curve for the decision boundary like shown in **Figure 4**. [12]

In order to be able to detect long-term faults, online monitoring uses the maximum current values per day as training data. By applying the one-class SVM to the

training data, an area is found that defines the desired operational status and serves as a reference. After the training phase, a second data set is created as a copy of the training data, which is continuously supplemented with new measurement data and permanently compared with the test data set. After another year of operation, the second data set consists exclusively of new measurement data. In order to classify the actual condition of the system, two additional limit values are calculated during the training phase.

The first limit value is surpassed if the PV system has lost a substantial part of its original output and may no longer be economically efficient to operate. The second limit value is exceeded if the power loss is so great that repowering of the PV system becomes necessary. In order to determine these limit values, the training data is manipulated by multiplying it by factors 0.5 and 0.75. The resulting areas are shown in **Figure 4**. In addition to the actual classification process, the percentage of overlap between the reference area and the area of the modified training data is calculated. On the other hand, the displacement of the areas on the y-axis to each other is also considered. The results of these two calculations are stored as limit values.

If the maximum current per day continuously decreases compared to the respective seasonal reference value, the overlay between the area from the current period and the reference area exceeds the defined limit values and a long-term fault is detected. Such a reduction of the generated power can be caused by pollution or ageing, for example. If the current values lie within the yellow area shown in **Figure 4**, only a simple warning of a power drop is issued. If the values are within the range of the red area, a request for a repowering of the plant appears.

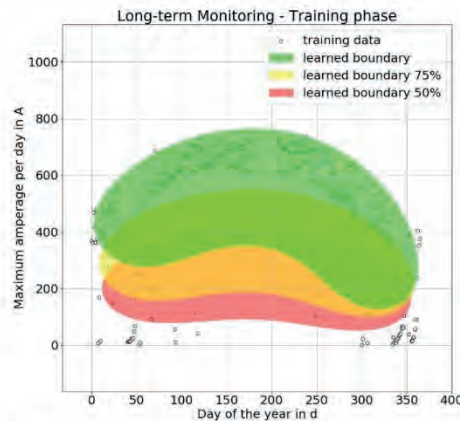


Figure 4 Maximum current per day over the days of the year as training data and learned areas for long-term monitoring with a one-class support vector machine.

3.3 Simulation

3.3.1 Training and test phase

The data set used for the simulation originates from the PV Data Acquisition Program (PVDAQ) of the National Renewable Energy Laboratory (NREL). The PV system is located on the NREL visitor parking lot and has an installed power of 524.16 kW. The direct current is measured on two strings of the PV system at intervals of 15 seconds. The measured data of the first year of operation are used as training data. The test data consists of the measurements of the following year.

Figure 3 shows the training data with the time derivatives of the electric current over the time together with the learned boundary. The decision boundary follows the outer data points of the training data set and encloses them all. The validation of the test data is visible in **Figure 5**. All time derivatives of the current in the following year are included within the boundary. Therefore, the *k*-Nearest Neighbours method does not detect any anomalies.

For long-term monitoring, the maximum amperage per day is decisive. **Figure 4** shows these training data over one year, as well as the area within the learned decision boundary (in green). The two areas of the limit values of the state of health are marked in yellow and red below the reference area.

The validation with the test data is shown in **Figure 6**. The areas of the reference and the test year are not congruent, but the overlap is still large. The online monitoring system has detected neither short- nor long-term anomalies and therefore the condition of the PV system in its second year of operation can be classified as very good.

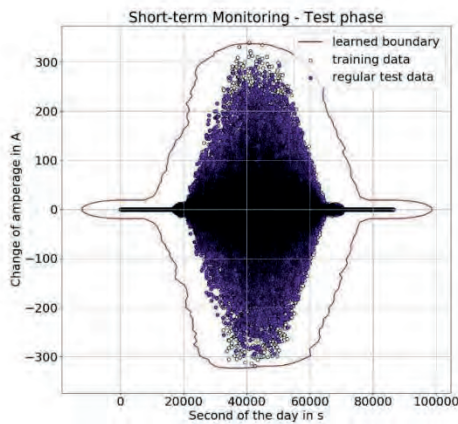


Figure 5 Time derivatives of the current over the seconds of the day as training and test data and the learned boundary for short-term monitoring with the *k*-Nearest Neighbours algorithm. All test data is within the boundary and is therefore classified as normal values.

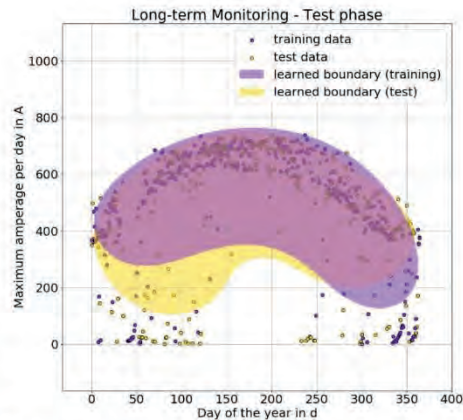


Figure 6 Maximum current per day over the days of the year as training and test data and learned areas for long-term monitoring with a one-class support vector machine. The overlap between the areas is not congruent, but still large enough for classification as normal values.

3.3.2 Fault tests

After the successful validation in regular PV operation, tests with deliberately provoked faults will follow. In order to simulate a short-term fault of the PV system, a complete breakdown is imitated. For this purpose, some values in the test data are set to zero on an exemplary summer day. This current collapse is simulated at evenly distributed times over the course of the day. **Figure 7** shows the sensitivity of the anomaly detection as a function of the magnitude of the current drop. The lower the irradiance of the sun, the more difficult it is to detect a complete power failure.

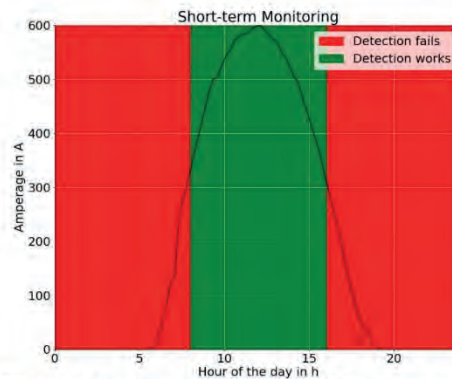


Figure 7 Amperage over the hours of the day and the dependent detection of the short-term monitoring with the *k*-Nearest Neighbours algorithm. If the current suddenly drops to zero within the red range, no detection takes place. Faults are only detected in the green range.

For the validation of the long-term monitoring, the test data are manipulated by multiplication with factors from 0 to 1, so that the simulated test data appear degraded to varying degrees. This makes it possible to determine at which points in time the monitoring system detects a significant fault. The first detection takes place at degradation values above 15 % and issues a warning. The second detection is triggered by values above 35 % degradation and recommends repowering. The monitoring reacts successfully to the decreasing maximum current of the test data.

3.4 Construction of a prototype

A Raspberry Pi was selected as the main component for the implementation of the online monitoring system. Its advantage lies in the good combination capability between software and hardware. It offers both the computing power for machine learning and a simple way to connect a current sensor and other electrical components. The measured values and the evaluations of the anomaly detection are stored in CSV files locally on the SD card. This data can be copied to an external device via the Ethernet interface and the SSH protocol parallel to online monitoring operation.

The current measurement is contactless, so that a monitoring system can be implemented without interfering with the existing cabling. Therefore, a split-core current transformer based on the Hall Effect is used. The installation is carried out by attaching the sensors to the DC cables, which lead from the PV modules to the inverters (Figure 8). The output of the current transformer is connected to an analog-to-digital converter, which transmits the measured current as a digital value to the Raspberry Pi via the SPI protocol.



Figure 8 Online monitoring system in grey housing with traffic light display and cables for connection to the current transformer (grey wire) and the power supply (black wire) as well as the current transformer in red housing, installed on a DC cable between PV module and inverter

In addition, a real-time clock with battery is added to the system. This makes it possible to store the actual time if the power supply to the Raspberry Pi is temporarily interrupted. After a restart, the Raspberry Pi obtains the time from the real time clock via the I²C protocol to store the following readings with the correct time stamp.

The current sensor and the Raspberry Pi with further electronics are both enclosed in a suitable housing in order to set up a convenient online monitoring system. A traffic light display consisting of a green, yellow and red LED indicates the actual status of the PV system directly on the housing. To protect the electronics from external influences, the cables for connecting the current sensor and the power supply are inserted into the housing with cable entries (Figure 9).



Figure 9 Printed circuit board with Raspberry Pi, real-time clock, analog-to-digital converter and other electronics in grey housing

4 Transferability of the approach

A big advantage of the approach of the online monitoring system is its transferability. In principle, it is possible to apply anomaly detection to any PV system. Due to the individual data acquisition and learning phase, it is possible to monitor the electrical behaviour independent of the system size and module type.

The approach of this online monitoring is not only suitable for photovoltaic systems. The idea of learning the behaviour of electrical systems by measuring specific parameters at the beginning of the life cycle or at a later upgrade, if the operation is still intact, can be applied to any electrical component. A predictive maintenance in different applications requires only a small adaptation of the algorithm to the specific temporal behaviour and suitable sensors to measure the relevant parameters. The essential part of the online monitoring of photovoltaic systems is therefore completely transferable. In other applications, the use of predictive maintenance can be even more useful if, for example, higher maintenance costs are to be expected or the component to be monitored is part of a critical infrastructure.

5 Results

The novel approach of the online monitoring algorithm, which is implemented on a Raspberry Pi, was successfully tested in the simulation. The validation of the anomaly detection was carried out with a data set from real measurements over a period of two years. A training phase enables the self-learning algorithms to determine the decision boundaries for the classification using the data from the first year. No anomalies could be detected for the test data from the second year. Under the pretext of various fault cases, however, the models react as desired and detect abnormal behaviour.

The comparison of this new approach with other online monitoring systems for photovoltaics shows some advantages. The main advantage is the low effort of data recording. While other monitoring systems require many sensors to measure parameters such as current, voltage, temperature and luminescence, this system is only equipped with a contactless current sensor. Due to the low measuring effort, only the installation of the current sensor on the DC line and a power supply are required. This means that the entire system can be installed in just a few minutes. In addition, the system is scalable to different PV system sizes. It is possible to apply anomaly detection to the entire system by measuring the total current of the system, or even to monitor each string individually.

Ultimately, it is possible to use the monitoring system regardless of the type of module or technology used. In addition, this online monitoring system approach can be applied to any other electrical component with minor adaptations.

6 Acknowledgement

The authors gratefully acknowledge financial support of this work by the Ministry of the Environment, Climate Protection and the Energy Sector Baden-Württemberg (Ministerium für Umwelt, Klima und Energiewirtschaft Baden-Württemberg BWSGD18012).

7 Literature

- [1] SolarPower Europe: Global Market Outlook For Solar Power / 2018 - 2022, Brüssel, 2017.
- [2] V. Wesselak und S. Voswinckel: Photovoltaic - How sun becomes electrical current (in German: Photovoltaik - Wie Sonne zu Strom wird), Berlin Heidelberg: Springer, 2016.
- [3] M. Köntges et al.: IEA-PVPS T13-01 2014 Review of Failures of Photovoltaic Modules Final, 2015.
- [4] M. Liebschner, K. Böhm, A. Reumann, K. A., R. Krump und J. Titz: On-line Monitoring of Capacitance and Dissipation Factor of High Voltage Bushings at Service Temperature, in 15th Symposium on High Voltage Engineering, Ljubljana, 2007.
- [5] A. Ahrary, M. Inada und Y. Yamashita: Solar Power Monitoring System "SunMieru", 2018.
- [6] Y. Liu, X. Zhu und J. Yang: Fault diagnosis of PV array based on optimised BP neural network by improved adaptive genetic algorithm, The Journal of Engineering, pp. 1427-1431, 2017.
- [7] M. Dhimish: Fault Detection and Performance Analysis of Photovoltaic Installations, Huddersfield: University of Huddersfield, 2018.
- [8] M. Sieg: Local technician vs. artificial intelligence (in German: Ortskundiger Techniker vs. künstliche Intelligenz), pv magazine, pp. 75-76, November 2018.
- [9] S. Raschka und V. Mirjalili: Machine learning with python and scikit-learn and tensorflow (In German: Machine Learning mit Python und Scikit-learn und Tensorflow), Frechen: mitp, 2018.
- [10] Y. Zhao, Z. Nasrullah und Z. Li: PyOD: A Python Toolbox for Scalable Outlier Detection, 2019.
- [11] C. Bishop: Pattern recognition and machine learning, New York: Springer, 2006.
- [12] B. Schölkopf et al.: Support vector method for novelty detection, Advances in neural information processing systems, pp. 582-588, 2000.

NEIS 2020: „Anomaly detection by comparing photovoltaic systems with machine learning methods“

Anomaly detection by comparing photovoltaic systems with machine learning methods

Moritz Benninger^{1,*}, Martina Hofmann¹, and Marcus Liebschner¹

¹University of Applied Sciences Aalen, 73430 Aalen, Germany

*moritz.benninger@hs-aalen.de

Abstract

The digital interconnection of several photovoltaic systems enables a comprehensive monitoring and detection of failures. The concept of this work is based on the similar behaviour of individual photovoltaic systems under same conditions. By recording electrical data, individual strings can be compared with each other by machine learning methods. For this purpose, a k -Nearest-Neighbours algorithm and a Multi-layer Perceptron are used for processing the data and detecting divergent characteristics of the electrical current. With this approach, a reliable and reproducible fault monitoring can be guaranteed without large measuring effort.

Keywords— *online monitoring, photovoltaic system, machine learning, anomaly detection, artificial intelligence*

1 Introduction

The establishment of photovoltaics (PV) is already well advanced in Germany. In 2018, 7.2 % of the electricity primarily generated was produced by photovoltaics [1]. The further expansion of photovoltaics in Germany could generate an additional 226 GW of capacity [2]. These numbers illustrate the current and future role of photovoltaics in the energy turnaround in Germany. This makes it all the more important to maintain the quality of the photovoltaic systems by monitoring.

For many existing plants, failures and problems with individual modules or inverters become more and more probable over time. As a result, monitoring and maintenance of photovoltaic systems will also become more important. These services will probably be offered and used more frequently in the future. For various players such as installers, plant operators and municipal utilities, a cheap and reliable monitoring system will be very attractive. With the monitoring system presented in this paper, the monitoring of several systems can be implemented without great effort in terms of measurement technology and hardware.

Digitalization offers the possibility of connecting decentrally distributed systems and thus performing comprehensive monitoring from a central location. This allows an overview over a large number of plants. The early detection of faults allows maintenance or repair to be initiated as required and in a targeted manner, thus significantly reducing costs and making PV systems even more attractive.

2 Online monitoring

Overall, there are many different possibilities and approaches to monitor photovoltaic systems. These often depend on the given circumstances. Madet and

Singh [3] have already compared different approaches and methods in detail in an overview. Monitoring is performed for different configurations, such as grid connected, standalone or hybrid PV systems. The use of parameters such as current, voltage, temperature and irradiance and the measurement technology vary in the numerous approaches. This also applies to the topics of data processing and storage. The crucial component of monitoring is finally the data analysis. Statistical evaluations with MATLAB[®], LabVIEW[®] and Excel[®] are common there.

For example, Kekre and Gawre [4] use an Arduino with voltage, current and temperature sensors for their monitoring system. The measured data are transmitted via Internet and can thus be viewed remotely by the plant operator. In addition, an SMS notification can be sent in case of implausible measured values. Xiaoli, Tao and Yunbo [5] monitor PV systems and diagnose faults using a multi-layer system. In total, a large amount of data and information about the individual solar cells, inverters, irradiation and weather conditions is collected. A current and voltage analysis can be used to diagnose any faults that occur.

Also artificial intelligence methods such as k -Nearest-Neighbours (k NN) algorithms, Support Vector Machines and neural networks are state of the art. In this field, there have recently been some new developments in connection with the monitoring of PV systems. Harrou, Taghezouit and Sun [6] use different approaches of k -Nearest-Neighbours algorithms for monitoring PV systems. Several meteorological and electrical quantities are recorded and additionally an electrical modelling of PV systems is applied. The differences between real and simulated output variables are monitored with the k NN model and depending on the deviation a corresponding error message (short circuit, open circuit, partial shading) is output. Wang et al. [7] use multi-class Support Vector Machines (SVM) for the diagnosis of errors. With the help

of electrical modeling, further data sets are generated from historical measured values. These include the electrical current and voltage at the maximum power point, the fill factor and an error factor. This data set is used to train three Support Vector Machines, which can then be used to classify the plant condition. Akram and Lotfifard [8] also use an electrical model of a solar cell. In this approach, the electrical parameters (current, voltage, power) are utilized to train a probabilistic neural network for the classification of defects. Samara and Natsheh [9] use a similar methodology for defect detection. A neural network and modelling are also applied here. The difference lies in the parameters used for the input data of the neural network. In the approach presented in this paper, the parameters are limited to irradiation, temperature and electrical voltage.

3 Method

The difference between this approach and other monitoring systems for photovoltaics is the interconnectivity of the systems. Here, the information and data can be linked together across the individual plants and evaluated centrally. In the case of comprehensive interconnection, the geographic proximity of the systems to each other plays a decisive role. If the distance is small, similar irradiation values can be assumed for the different photovoltaic systems. Since the irradiation values are proportional to the electricity generation, the systems can be related to each other via the parameter of the electric current. This has the big advantage over other approaches that the required measurement technology is highly reduced and only one single current sensor needs to be installed per monitored string [10].

The analysis of the measured data is done by machine learning. Two different methods, a k -Nearest-Neighbours algorithm and a Multi-layer Perceptron, are used and their performance in the application is compared. During the training phase, the artificial intelligence learns the specific behaviour of the different strings by comparing the measured data. The behaviour of the strings to each other is monitored in the later test phase. If deviations occur in the measurement data of a specific string, anomalies are detected and an error message is issued. The analysis of the data is implemented with the k -Nearest-Neighbours algorithm and the Multi-layer Perceptron. The methods are applied at the same real data set.

3.1 k -Nearest-Neighbours

In general, the k -Nearest-Neighbours (k NN) algorithm allows the classification of individual samples of a data set. A data set can be divided into any number of classes. To train the k NN algorithm, the k nearest data points are determined for each individual sample of the data set. This is done using the Euclidean metric, which calculates the distance d between the sample and the k nearest neighbours [11]. This methodology, as shown in **Figure 1**, allows the division of a data set X_{train} into normal observations and anomalies. For this purpose,

k -Nearest-Neighbours

- Learns decision boundary for separation of classes based on distance metric
- Allows classification in normal values and anomalies

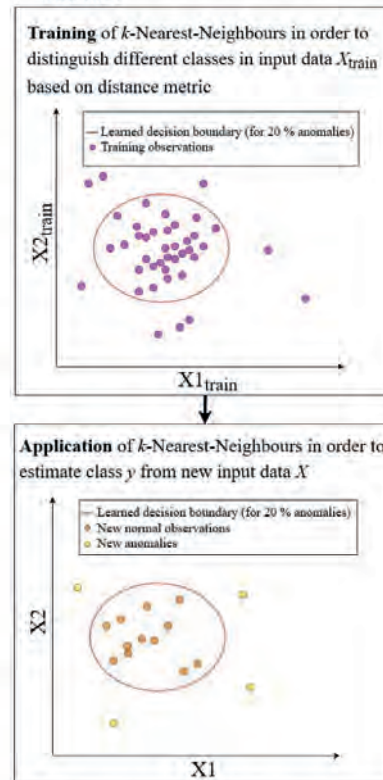


Figure 1 Summary of k -Nearest-Neighbours algorithm.

The k NN algorithm learns in the training phase to distinguish different classes and estimates the class from new input data in the application phase.

the data points with the largest distances d to their k nearest neighbours are classified as anomalies. The classification of new samples is then a majority decision between the corresponding classes (normal observation, anomaly) of the k nearest neighbours. This results in a decision boundary, which separates the different classes from each other and allows the classification of new data observations. [12]

The k -Nearest-Neighbours algorithm is used to compare the measurement data of two strings. For the training of the k NN algorithm it is specified that 20 % of all data points should be classified as anomalies. **Figure 1** shows how the anomalies (yellow) are separated from the normal values (orange) by a decision boundary in red. The decision boundary of the trained algorithm is used to classify the measured data in later operation. To monitor a specific

string, a separate comparison is made with all other strings. The average value is calculated from the individual shares of anomalies and used as a basis for anomaly detection. In the case of a faulty string, the data points will increasingly be outside the decision boundary, which increases the number of anomalies. A warning is issued if the share of anomalies rises well above the defined reference value of 20 %.

3.2 Multi-Layer Perceptron

The Multi-layer Perceptron allows the recognition of relationships between input and output data. A perceptron is one of the simplest architectures of neural networks and is based on a slightly modified artificial neuron called Linear Threshold Unit (LTU) with input X and output y . The input X has additional weightings w . The Linear Threshold Unit calculates the weighted sum z of the input X . An activation function is applied to the result, which ultimately produces the output y . For learning the relationships between input X_{train} and output y_{train} of the training data, the weights w of the perceptron are tuned by an algorithm. With new input data X , the expected output data y can be determined, as shown in **Figure 2**. A perceptron consists of only a single layer of Linear Threshold Units. A Multi-layer Perceptron, on the other hand, consists of an input layer, several hidden layers and the output layer. This multi-layer structure allows some of the limitations of a single perceptron to be overcome. Thus, various classification and regression tasks can be solved. [13]

The Multi-layer Perceptron does not compare individual strings with each other, but includes the data of all strings. With the MLP, the current power production of a single string is forecast based on the current power production of the remaining strings. This allows a single string to be monitored in relation to the remaining strings. The coefficient of determination is calculated to assess the agreement between the forecast and measurement. In the training phase there is a high agreement with values close to 100 % due to the very similar behaviour of the strings and the same irradiation conditions. For later operation, the coefficient is used to determine a possible deviation between the value forecasted by the Multi-layer Perceptron and the measured value. Due to increasing anomalies the measured data deviate from the forecasted values. The coefficient of determination therefore decreases, which results in the detection of an error in the respective string.

4 Data

For obtaining the data set, measurements were performed on different plants with a total of eight strings in the period between August 2019 and February 2020. The plants are distributed over an area of about 1 km². The direct current at the monitored string was measured in a time interval of three seconds using a split-core DC sensor. Before a more detailed examination and analysis of the data can be carried out, the data must first be processed accordingly. For this

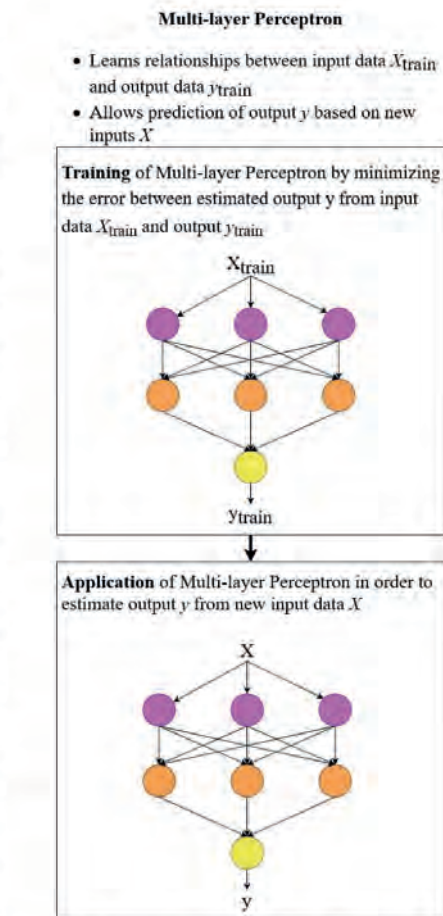


Figure 2 Summary of the Multi-layer Perceptron. The Multi-layer Perceptron learns the relationships between the training data and estimates the output from new input data in the application phase.

purpose the time stamps are synchronized and the electrical current values are normalized to a range between 0 and 1. It should be noted that measurement failures also occur. These appear for different periods of time and also vary between the respective plants. The failures are removed from the data set for the following executions. In addition, the sensors have a certain scattering and inaccuracy in the de-energized state. However, this has no influence on the functionality of the monitoring.

The measurement data are shown in **Figure 3** as an example for one week in January 2020. The course of the generated, standardized direct current for the different strings of the PV systems is for the most part very similar. The influence of the slightly different orientations and performance variables of the different systems plays only a minor role.

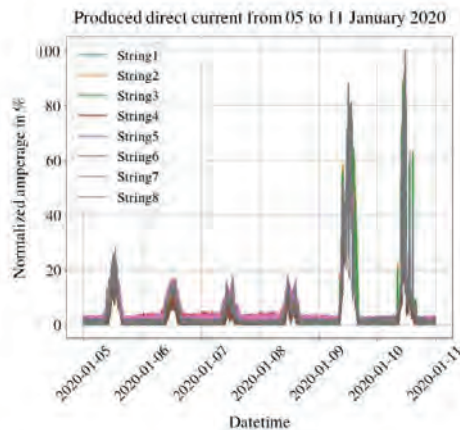


Figure 3 Course of the measured and normalized direct current of several PV systems in % over the period from 05 to 11 January 2020. The measured data originate from the recorded data set with different strings and synchronized time stamps, which is used for the validation of the methodology.

5 Validation and simulation

The validation of both methods is done with data sets from real measurements. The data is split into a training and a test data set. A share of 75 % of all data is used for training the machine learning algorithms. The k NN algorithm learns the decision boundary and the weights of the Multi-layer Perceptron are tuned based on this training data. The remaining 25 % of the data serve as test data and are used to check the functionality of the trained models. Over the measurement period, the systems behaved normally and without any abnormalities, so that no faulty behaviour is assumed in the measured data. Therefore, the algorithms cannot detect any anomalies in the test data at first. With the k NN algorithm, the share of anomalies remains at the predefined value of 20 %. Even with the Multi-layer Perceptron, the values of the coefficient of determination are close to 100 % agreement, as expected.

In order to be able to validate the algorithms now also for faulty behaviour, two different faults are artificially generated. For this purpose, the original measurement data are digitally manipulated. In the first case, a short-term drop in the current of the respective string is simulated, as it could occur, for example, due to external mechanical damage or a short circuit. For this purpose, the power production of the plants in the data is suddenly set to zero. In the second case, a creeping, long-term deterioration is simulated, as it could occur, for example, due to pollution or aging. To do this, the data is multiplied by a factor corresponding to the degradation.

5.1 Short-term fault

In the simulation of the short-term fault, the normal current values suddenly and permanently drop to zero. From this point, the behaviour of the system deviates from the normal state and the algorithms can detect anomalies.

Figure 4 and **Figure 5** show the progression of the respective fault rates (k NN: share of anomalies, MLP: coefficient of determination) over the number of monitored samples for the simulation of the respective faulty string. For the k -Nearest-Neighbours algorithm, the share of anomalies changes significantly from the time of failure, as shown in **Figure 4**. The share of anomalies is at the expected value of 20 % before the failure and then increases over time. For the Multi-layer Perceptron, the coefficient of determination before the failure is also at the expected value close to 100 %. With the abrupt failure of the respective system, the coefficient decreases further and further (see **Figure 5**). The fault rate of the respective method becomes larger and larger over time, as the anomalies continue to accumulate.

5.2 Long-term fault

For the simulation of the long-term fault, the current values are multiplied by a factor to simulate degradation. This is done for degradation rates between 0 and 40 % over 700.000 test samples, and the respective fault rates of the two methods are calculated.

Figure 6 and **Figure 7** show these fault rates (k NN: share of anomalies, MLP: coefficient of determination) versus the degree of degradation. In the k -Nearest-Neighbours algorithm, the share of anomalies increases as the degradation rate increases, starting at 20 % (**Figure 6**). Degradation is also successfully detected with the Multi-layer Perceptron. The coefficient of determination decreases steadily starting at nearly 100 % as degradation increases (**Figure 7**).

5.3 Faults of several plants

In addition, further simulations are used to investigate what effects the faulty behaviour of several plants simultaneously has on the two methods. With the k NN algorithm, the detection of anomalies continues to work well and the faulty strings are correctly identified. In this method, all strings are included, but only two strings at a time are directly compared. Therefore the method is very robust. With the Multi-layer Perceptron, on the other hand, several failures lead to incorrect results. This is due to the fact that in this model several strings are used as input variables for the Multi-layer Perceptron. If these input variables are already faulty, no meaningful prediction can be made for the examined string. However, it should be noted that in the present test the number of input variables is relatively small with seven other strings. With a larger data set with more photovoltaic systems, the robustness of the model will automatically increase.

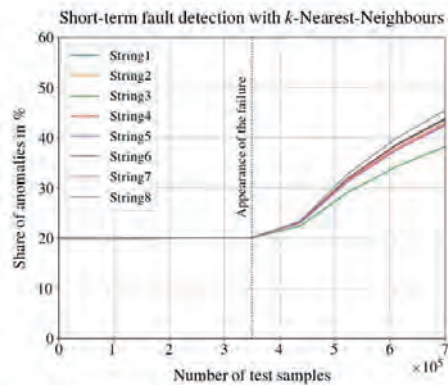


Figure 4 Averaged share of anomalies of the k -Nearest-Neighbours algorithm in % over the number of test samples checked for the respective faulty string of the PV system. The dotted line marks the start of the short-term fault behaviour by simulating an abrupt failure of the plant. From this point on, the share of anomalies increases and a fault can be detected.

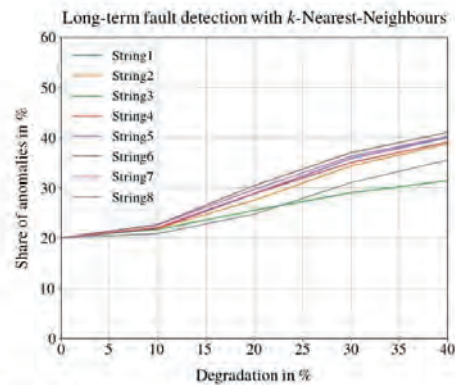


Figure 6 Averaged share of anomalies of the k -Nearest-Neighbours algorithm in % over the percentage degradation for the respective faulty string of the PV system. A long-term fault behaviour due to a degradation of the electrical power (0 to 40 %) is simulated. As degradation increases, the share of anomalies increases and a fault can be detected.

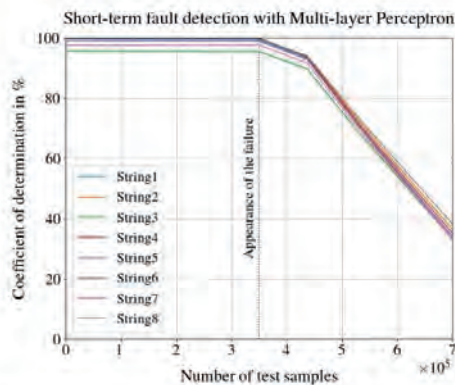


Figure 5 Coefficient of determination of the Multi-layer Perceptron in % over the number of test samples checked for the respective faulty string of the PV system. The dotted line marks the start of the short-term fault behaviour by simulating an abrupt failure of the plant. From this point on, the determination coefficient decreases and a fault can be detected.

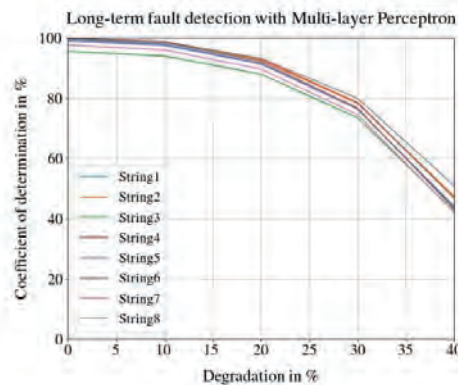


Figure 7 Coefficient of determination of the Multi-layer Perceptron in % over the percentage degradation for the respective faulty string of the PV system. A long-term fault behaviour due to a degradation of the electrical power (0 to 40 %) is simulated. As degradation increases, the coefficient of determination decreases and a fault can be detected.

5.4 Results

Generally, it can be stated that with both methods an anomaly detection is possible. The functionality of both methods could be demonstrated with the measured data sets. A short-term fault in the simulated form is easier to identify than a long-term faulty behaviour due to the larger deviation. The long-term fault is more difficult to identify, especially at small degradation rates. Nevertheless, with a sufficiently large data set, deviations from the normal

condition can be determined and a reliable statement about the state of the respective string can be made. When detecting several parallel failures of systems, however, the Multi-layer Perceptron produces faulty results. In comparison, the k NN algorithm is much more robust and can reliably detect faulty systems. In general, a larger data set also leads to more reliable results. For this method it is also advantageous if a new photovoltaic system is included in the data set at regular intervals. In this way, the degradation of existing systems can always be checked.

This method is specially adapted for use by institutions and companies that will in future take on various maintenance and repair tasks for photovoltaic systems. These include, for example, installers, municipal utilities, manufacturers of PV systems, but also the operators of several or very large PV systems. For these players it is important to be able to monitor many systems efficiently and with as little effort as possible. Compared to other methods, this approach therefore offers specific advantages. On the one hand, the hardware required to measure suitable parameters is significantly less than for other solutions. A single current sensor is required per string. Since this sensor can be installed non-destructively, the main advantage is that it can be retrofitted. Moreover, the method can be applied regardless of the size or model of the PV system and without the knowledge of any parameters. These aspects are often missing completely with other approaches.

The disadvantage of this method is generally the problem of data availability. Stable interconnection and data transmission must be ensured to enable reliable detection. A further disadvantage is that the systems must all be located in a geographic limited area. At greater distances, the weather conditions and irradiance values change and a reliable comparison of the individual systems is no longer possible. As an alternative, the reference values, which are absolutely necessary for this method, can also be determined using a luminescence sensor or from actual weather data.

6 Conclusion

The digital interconnection of several photovoltaic systems enables comprehensive monitoring of many systems. By referencing the different strings of the photovoltaic systems, statements about the condition of the respective system can be derived. Two methods from machine learning, a k -Nearest-Neighbours algorithm and a Multi-layer Perceptron, are used in combination with measurement data from a DC sensor. The application of the methods with real data works successfully and anomalies are detected. The k NN algorithm is much more robust than the Multi-layer Perceptron, as it can detect several faults simultaneously. The advantage over other approaches for monitoring is the low measurement and installation effort, while stable data transmission must be ensured. In addition, a geographic proximity between the systems is required. The approach is particularly suitable for practical and large-scale use in connection with regional players who are responsible for the maintenance of photovoltaic systems.

7 Acknowledgement

The authors acknowledge the financial support for this work by the Ministry of the Environment, Climate Protection and the Energy Sector of Baden-Württemberg (BWSGD18012).

8 References

- [1] Bundesministerium für Wirtschaft und Energie, *Energy data: Complete edition*, Berlin (In German), 2019.
- [2] Bundesministerium für Verkehr und digitale Infrastruktur, *Spatially differentiated area potentials for renewable energies in Germany*, Berlin (In German), 2015.
- [3] S. R. Madet and S. N. Singh, "Monitoring system for photovoltaic plants: A review," in *Renewable and Sustainable Energy Reviews*, 2017, pp. 1180-1207.
- [4] A. Kekre and S. K. Gawre, "Solar photovoltaic remote monitoring system using IOT," *2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE)*, Bhopal, 2017, pp. 619-623.
- [5] X. Xiaoli, C. Tao and Z. Yunbo, "Condition monitoring and diagnosis for grid-connected photovoltaic power system," *2013 IEEE 11th International Conference on Electronic Measurement & Instruments*, Harbin, 2013, pp. 807-811.
- [6] F. Harrou, B. Taghezouit and Y. Sun, "Improved kNN-Based Monitoring Schemes for Detecting Faults in PV Systems," in *IEEE Journal of Photovoltaics*, vol. 9, no. 3, pp. 811-821, May 2019.
- [7] L. Wang, J. Liu, X. Guo, Q. Yang and W. Yan, "Online fault diagnosis of photovoltaic modules based on multi-class support vector machine," *2017 Chinese Automation Congress (CAC)*, Jinan, 2017, pp. 4569-4574.
- [8] M. N. Akram and S. Lotfifard, "Modeling and Health Monitoring of DC Side of Photovoltaic Array," in *IEEE Transactions on Sustainable Energy*, vol. 6, no. 4, pp. 1245-1253, Oct. 2015.
- [9] S. Samara and E. Natsheh, "Intelligent Real-Time Photovoltaic Panel Monitoring System Using Artificial Neural Networks," in *IEEE Access*, vol. 7, pp. 50287-50299, 2019.
- [10] M. Benninger, M. Hofmann and M. Liebschner, "Online Monitoring System for Photovoltaic Systems Using Anomaly Detection with Machine Learning," in *NEIS Conference on Sustainable Energy Supply and Energy Storage Systems 2019*, 2019, pp. 215-220.
- [11] S. Raschka and V. Mirjalili, *Machine learning with python and scikit-learn and tensorflow*, Frechen: mitp (In German), 2018.
- [12] C. Bishop, *Pattern recognition and machine learning*, New York: Springer, 2006.
- [13] Aurélien Géron, *Hands-on machine learning with Scikit-Learn and TensorFlow*, Heidelberg: O'Reilly (In German), 2018.

KES 2020: „Transferable and self-learning Online Monitoring System for electrical household appliances”



Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2020) 000–000

Procedia

Computer Science

www.elsevier.com/locate/procedia

24th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

Transferable and self-learning Online Monitoring System for electrical household appliances

Moritz Benninger*, Martina Hofmann, Marcus Liebschner

University of Applied Sciences Aalen, 73430 Aalen, Germany

Abstract

In this paper an Online Monitoring System is presented, which enables a universal condition monitoring of electrical household appliances. The approach involves recording the power consumption of the respective device and evaluating the measured data using machine learning methods and tools. Thereby an artificial intelligence learns the normal condition of the examined device and can subsequently perform a monitoring. A *k*-Means algorithm is used for clustering and a One-class Support Vector Machine for classification. This makes it possible to detect anomalies in the operating modes and behavior of the device during subsequent operation. In addition to the methodology, the design and real application of a prototype is presented, in which a suitable sensor technology and a software with artificial intelligence are implemented.

© 2020 The Author(s). Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Peer-review under responsibility of KES International.

Keywords: online monitoring; machine learning; artificial intelligence; clustering; classification; anomaly detection

1. Introduction

With the advancing digitalization and the spread of the Internet of Things, the influence and importance of data is increasing. Through the increased use of sensors, a multitude of processes and procedures can be monitored by measuring technology. The evaluation and analysis of measurement data is often carried out with tools from machine learning. Thanks to high computing capacity in compact designs, the use of artificial intelligence (AI) is easily possible for many applications. The combination of inexpensive sensors and artificial intelligence creates a powerful tool. Thus, complex challenges can be mastered with the help of the most simple components. The evaluation of measurement data can open up many advantages for humans. Fitness wristbands and watches serve as a simple example. With this equipment, people can measure and monitor various parameters of their physical condition, such as the current heart

* Corresponding author. Tel.: +49-7361-576-5715

E-mail address: moritz.benninger@hs-aalen.de

1877-0509 © 2020 The Author(s). Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Peer-review under responsibility of KES International.

rate, calories burned or steps taken. If this results in deviations from the desired parameters, it is possible to react by adjusting behaviour, such as more sporting activity or a healthier diet.

Despite these developments, there are many examples of technical equipment where no monitoring is carried out. Due to a lack of know-how, some industries are still behind the current state of digitalization and artificial intelligence. Especially existing devices are not yet integrated. Often only a few steps are missing to turn an analog component into a smart device. The prototype presented in this paper enables a smart upgrade and thus the exploitation of new potentials for different devices. The tools for such an implementation are methodically explained and prototypically demonstrated in the paper. After the introduction, Chapter 2 first compares some approaches for online monitoring. Chapter 3 explains the structure of the hardware and software in detail. This is followed in Chapter 4 by the results from real applications and in Chapter 5 by the final conclusion.

2. Online monitoring

Online monitoring is already being used sensibly in various areas. These include sectors such as chemistry and biology [1], but also various areas in the medical and health care industry, e.g. for monitoring patients [2][3][4]. This paper deals specifically with the condition monitoring. There are many technical applications for the condition monitoring, which are mainly used on machines and industrial plants. These include, for example, the condition monitoring of nuclear power plants [5], photovoltaic systems [6], transformers [7] or high-voltage bushings [8].

The state of a plant is defined by one or more parameters. Measurement data is recorded from these parameters, which can vary greatly depending on the device. The data is then analysed and evaluated [9]. The numerous approaches differ in the methods. For example, when monitoring photovoltaic systems, these range from statistical evaluations with MATLAB®, LabVIEW® and also Microsoft Excel® to models and simulations and artificial intelligence in various forms such as artificial neural networks or genetic algorithms [10]. In connection with condition monitoring of electric drives, knowledge-based approaches are increasingly used [11]. These are based on methods from machine learning, which represent a subarea of artificial intelligence. For example, many monitoring systems for electric drives are based on artificial neural networks [12][13], support vector machines [14][15] or a combination of different tools [16].

Monitoring in the household sector is a less prominent topic, which is also reflected in the state of technology. Nevertheless, literature on the monitoring of individual appliances can be found [17]. These approaches are, among other things, also linked to methods from machine learning [18][19], but primarily refer to monitoring of energy consumption. Artificial intelligence and monitoring are also increasingly used in industry for household applications. For example, refrigerators [20] and washing machines [21] can be found with built-in intelligence. In this context, there are also refrigerators with their own smart diagnostics function, which perform a kind of condition monitoring [22].

3. Method

The basic idea of monitoring in this approach is to learn the normal state of the device and to refer to it in the further course of the process. Monitoring can be used universally for new and existing devices. The individual steps of the entire approach are shown in Figure 1. The procedure can basically be divided into a training phase and an operating phase. In the first step of the training phase, measurement data is recorded using suitable sensors for measuring alternating current. The measured current data is stored internally in the monitoring system, so that it can be accessed quickly and easily after a sufficiently long measurement phase. After completion of the measurement phase, the data is processed in two steps by the algorithms from Machine Learning. First, the data is classified into different operating modes using the clustering procedure *k*-Means. This classification is then passed on to the One-class Support Vector Machine for further classification. Thus, for each operating mode a boundary is calculated inside which the data values are in the normal state. The training phase is now completed. During the operating phase, the determined boundaries serve as a decisive tool for classifying the measured data into normal and abnormal values. If the data are within the boundaries, standard operation shall be assumed. If, however, an above-average number of data points are outside the limits, an anomaly is detected and the behavior is classified as faulty.

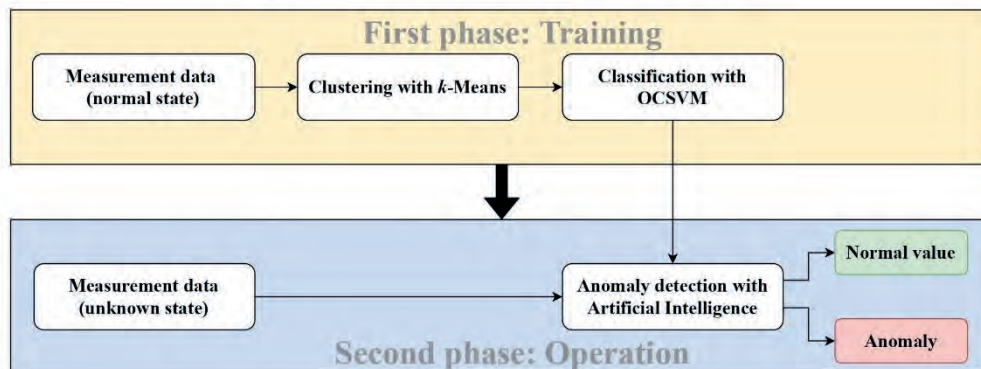


Fig. 1: Flow chart of online monitoring divided into a training phase with k -Means algorithm and One-class Support Vector Machine (OCSVM) and an operating phase with anomaly detection by artificial intelligence.

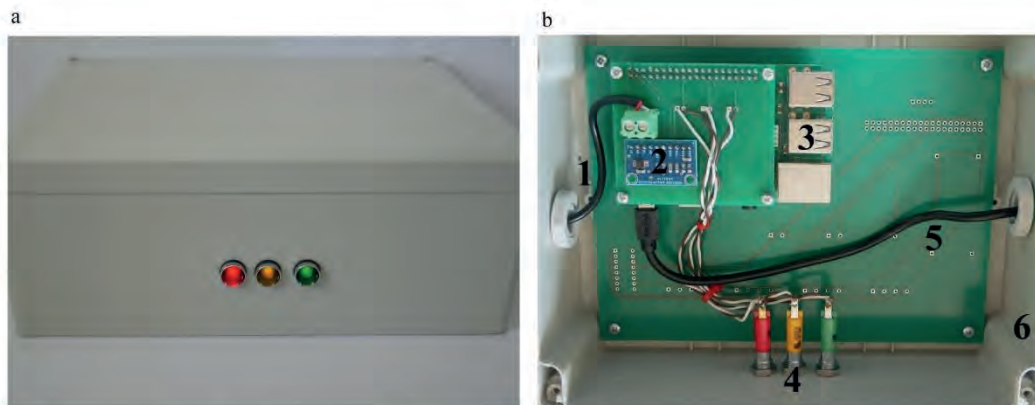


Fig. 2: (a) Outside view of the Online Monitoring System prototype in the training phase (all LED turned on); (b) Internal set-up of the Online Monitoring System with cable of the current sensor (1), analog-digital converter (2), Raspberry Pi (3), status light consisting of light-emitting diodes (4), cable of the supply voltage (5) and housing (6).

3.1. Hardware

The hardware consists of several individual components, including a sensor for alternating current measurement, an analog-to-digital converter, a Raspberry Pi, a housing and three light-emitting diodes (see Figure 2). The hardware is designed to allow household appliances to be monitored flexibly. The monitoring can be applied to all devices operated with a single-phase power connection. For the implementation of a monitoring system, the measurement technology is of great importance. A current sensor is used, which enables a contactless and damage-free measurement of the alternating current. The output signal of the sensor is passed to the analog-to-digital converter ADS1115 with a precision of 16-bit. There the analog signals are converted into digital values. These digital values are then communicated to the Raspberry Pi via an I²C interface.

The analog-to-digital converter and the Raspberry Pi are housed in a standard case. In addition, status lights are installed at the housing, which can be used to output the status of the monitored device (see Figure 2). The current sensor, which is connected to the device to be examined, is located separately.

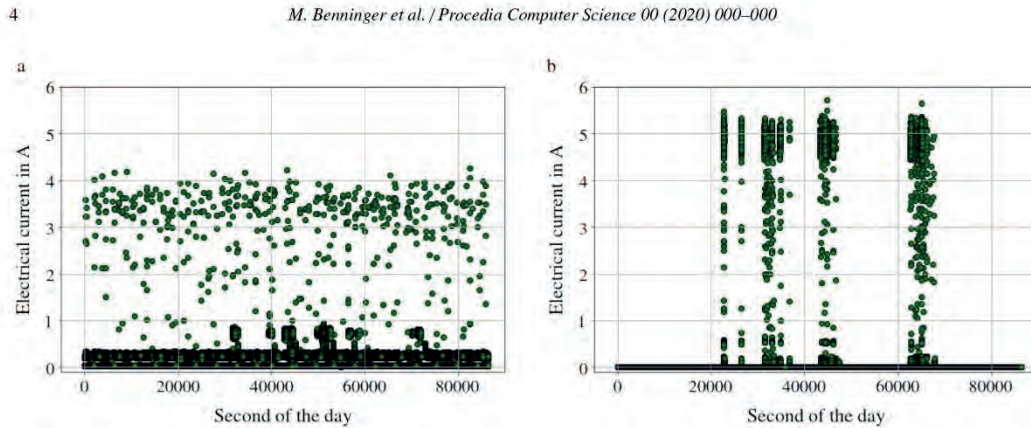


Fig. 3: Measured values of the electrical current, reflecting the power consumption of a refrigerator (a) and a coffee machine (b) with a resolution of one second over one day. These data sets serve as a starting point for artificial intelligence training.

3.2. Software

With the described hardware the recording of the measurement data is performed. As soon as the respective electrical device is connected to power, the Raspberry Pi automatically starts measuring and storing the data. In the Raspberry Pi, the digital values obtained via the I²C interface are stored directly in an SQLite database together with the current time stamp. From the database they can be loaded and processed in applications later. The measurements result in data sets as shown in Figure 3 as an example. In the following, the machine learning based on the data for the purpose of monitoring is shown in more detail.

3.2.1. Clustering

The k -Means algorithm can be assigned to the non-probabilistic techniques. It can be used to identify a number of k clusters in a data set. First, vectors μ_k are introduced that represent the center of the respective cluster. The goal of the algorithm is to assign the data points to the clusters and to select the centroids μ_k (center of a cluster) in such a way that the sum of the squares of the distances from each data point x_n to the next centroid μ becomes minimal. A common measure of distance in clustering is the squared Euclidean distance of two points in m -dimensional space. In addition, an indicator variable r_{nk} is introduced which is equal to 1 if the data point x_n belongs to the cluster, otherwise equal to 0. This allows the target function J to be declared, which is iteratively minimized to convergence in an optimization task [23]:

$$J = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \|x_n - \mu_k\|^2 \quad (1)$$

The disadvantage of the k -Means algorithm, however, is that the number of clusters k must be determined in advance. An incorrectly selected value of k can reduce the performance of clustering. To be able to make a suitable selection of k centroids, the silhouette analysis is used. With its help, the quality of a clustering can be assessed and the optimal number of clusters k can be determined. The higher the coefficient of the silhouette, the higher the quality of the clustering. To perform the silhouette analysis, calculations are necessary in three steps. First, the variable a is calculated as the mean value of the distances between the object and all other objects x in the same cluster. Then the distance b to the nearest cluster is calculated as the mean distance between the object and all other objects x of the nearest cluster. Finally, the silhouette coefficient s can be calculated by dividing the difference between the variables b and a by the larger of the two values [24]:

$$s^{(i)} = \frac{b^{(i)} - a^{(i)}}{\max\{b^{(i)}, a^{(i)}\}} \quad (2)$$

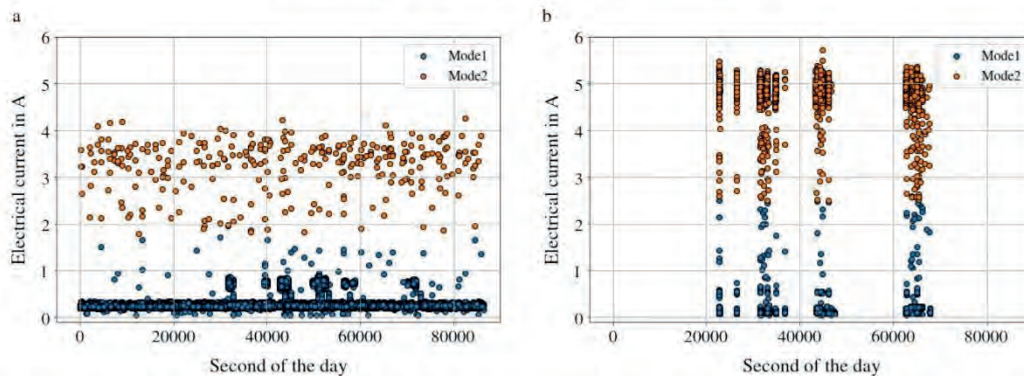


Fig. 4: Measured values of the electrical current, reflecting the power consumption of a refrigerator (a) and a coffee machine (b) with a resolution of one second over one day. The data are divided into different operating modes by the k -Means algorithm.

The k -Means algorithm is now applied to the measurement data collected in the database. The values for the silhouettes are calculated to automatically select the optimal number of k centroids. The data is then divided into k clusters as shown in Figure 4. These clusters represent the different operating modes of each device. If the associated values of a cluster represent less than 0,1 % of all data points, they are considered irrelevant and this operating mode is excluded.

3.2.2. Classification

The One-class Support Vector Machine (OCSVM) is a classifier often used to detect anomalies in records. Its special feature is that the algorithm can distinguish between normal and abnormal observations of previously unknown data points. The main goal of this method is to maximize the distance between the anomalies and normal values. [25] For this purpose the corresponding data set is projected into a higher dimension. In this dimension, the algorithm searches for an optimal hyperplane that separates a previously defined share of anomalies from the data set with the maximum distance from the origin. This hyperplane is represented by the following equation, where x represents the data values, w the weight vector and b the distortion [23]:

$$w^T x + b = 0 \quad (3)$$

The hyperplane is then projected back into the original dimension together with the data set, resulting in a nonlinear curve for the decision boundary, as shown in Figure 5. [25]

The OCSVM algorithm is used to monitor the individual operating modes for anomalies. The OCSVM algorithm is trained to classify 20 % of all data points as anomalies. The anomalies are separated from normal values by a decision boundary, as shown in Figure 5. Although these data points do not represent anomalies in reality, this procedure is used to define a threshold. This threshold value is important for the later determination of the device state. For the classification of the measured data in the unknown state, the decision boundary of the trained algorithm is used. During normal operation, anomalies occur with a share of 20 %. In case of a faulty device the data points will be outside the boundary. As a result, the share of anomalies will increase above the 20 % threshold. Thus, the device is clearly identified as faulty and a warning is issued.

4. Transferability of the method

The transferability and universality of this basic approach will be investigated below. The method in this paper is derived from an already existing concept. With the original approach, photovoltaic systems are monitored for short-term faults and anomalies in the system behaviour are detected using a k -Nearest-Neighbours algorithm [6].

6

M. Benninger et al. / Procedia Computer Science 00 (2020) 000–000

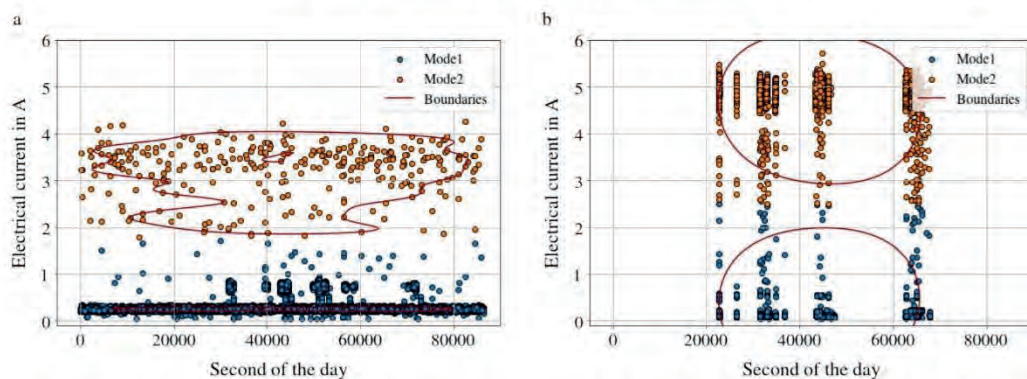


Fig. 5: Measured values of the electrical current, reflecting the power consumption of a refrigerator (a) and a coffee machine (b) with a resolution of one second over one day. For the already clustered data, decision boundaries for the classification of anomalies are learned with the One-class Support Vector Machine.

The methodology of this paper is based on a One-class SVM instead of a k -Nearest-Neighbours algorithm. As an additional component, the original approach for better transferability is extended by clustering for the detection of different operation modes. This makes it easier to detect anomalies in the case of unsteady behaviour of the device. Nevertheless, the method remains comparable with its origin, since photovoltaic systems usually have only one operating mode.

To extend the method to broader smart applications in a next step, anomaly detection would have to work on the basis of longer time scales. This would ensure more reliable and efficient monitoring overall. The algorithm then considers not only individual values, but a group of many measurement points over a longer period of time. Such an array of data offers a higher information content than a single data point and allows clearer conclusions about the reason for the deviation. This broader approach can also be used to monitor routines associated with household appliances. Anomaly detection of household appliances in this form will become increasingly important for future applications in the smart home or in elderly care.

5. Results

In a first series of tests, the data of a refrigerator and a coffee machine were recorded using the monitoring system and the methods of machine learning were applied. The raw data, the clustering by the k -Means algorithm and the classification by the One-class Support Vector Machine are shown in Figures 3, 4 and 5. The measured data of the refrigerator and the coffee machine were each divided into two operating modes. The artificial intelligence trained with the measured data was then used for further monitoring of the refrigerator and the coffee machine. The results in Figure 6 show the average share of anomalies over the samples of the test data. Overall, the share of anomalies in the two test series (refrigerator and coffee machine) varies by 20 % with small deviations (see Figure 6, green curve). This means that the share of anomalies is equal to the expected threshold value. The state of the refrigerator and coffee machine is therefore considered normal.

At this early stage, it was not yet possible to make a full statement on the functionality of the algorithm and the monitoring system. In order to be able to represent the behavior of the software in case of an error, an abnormal behavior had to be simulated. For this purpose, it was assumed that aging is accompanied by wear and contamination, which increases the overall power consumption. The data sets were manipulated with a higher power consumption and the monitoring of the two household devices were tested again. From an additional power consumption of 10 and 20 % respectively, the share of anomalies increased with values of over 25 % well above the threshold value (see Figure 6, orange and red curve). The operation of the refrigerator and coffee machine was therefore classified

- [5] J. H. Min, D.-W. Kim, and C.-Y. Park, "Demonstration of the validity of the early warning in online monitoring system for nuclear power plants," *Nuclear Engineering and Design*, vol. 349, pp. 56-62, 2019.
- [6] M. Benninger, M. Hofmann, M. Liebschner "Online Monitoring System for Photovoltaic Systems Using Anomaly Detection with Machine Learning," in D. Schulz *NEIS 2019: Conference on Sustainable Energy Supply and Energy Storage Systems*, VDE Verlag: Hamburg, pp. 215-220, 2019.
- [7] J. Chow, R. Lee, E. Wong, K. Leung, and G. Lai, "Novel engineering techniques to overcoming traditional challenges in online condition monitoring systems for power transformer," *HKIE Transactions*, vol. 25, no. 4, pp. 248-254, 2018.
- [8] M. Liebschner, K. Boehm, A. Reumann, A. Kuechler, R. Krump and J. Titzte (2007) "On-line Monitoring of Capacitance and Dissipation Factor of High Voltage Bushings at Service Temperature," in *15th Symposium on High Voltage Engineering*, Ljubljana, 2007.
- [9] ISO 17359:2018(E) "Condition monitoring and diagnostics of machines - General guidelines"
- [10] S. R. Madet and S.N. Singh, "Monitoring system for photovoltaic plants: A review," *Renewable and Sustainable Energy Reviews*, pp. 1180-1207, 2017.
- [11] X. Liang and K. Edomwandekhoe, "Condition Monitoring Techniques for Induction Motors," *2017 IEEE Industry Applications Society Annual Meeting*, Cincinnati, OH, 2017, pp. 1-10.
- [12] R. R. Schoen, B. K. Lin, T. G. Habetler, J. H. Schlag and S. Farag, "An unsupervised, on-line system for induction motor fault detection using stator current monitoring," *Proceedings of 1994 IEEE Industry Applications Society Annual Meeting*, Denver, CO, USA, 1994, pp. 103-109 vol.1.
- [13] J. F. Martins, V. Ferno Pires and A. J. Pires, "Unsupervised Neural-Network-Based Algorithm for an On-Line Diagnosis of Three-Phase Induction Motor Stator Fault," in *IEEE Transactions on Industrial Electronics*, vol. 54, no. 1, pp. 259-264, Feb. 2007.
- [14] J. Kurek and S. Osowski, "Support vector machine for fault diagnosis of the broken rotor bars of squirrel-cage induction motor," *Neural Computing and Applications*, vol. 19, pp. 557-564, 2009.
- [15] J. S. L. Senanayaka, S. T. Kandukuri, Huynh Van Khang and K. G. Robbersmyr, "Early detection and classification of bearing faults using support vector machine algorithm," *2017 IEEE Workshop on Electrical Machines Design, Control and Diagnosis (WEMDCD)*, Nottingham, 2017, pp. 250-255.
- [16] A. A. Silva, A. M. Bazzi and S. Gupta, "Fault diagnosis in electric drives using machine learning approaches," *2013 International Electric Machines & Drives Conference*, Chicago, IL, 2013, pp. 722-726.
- [17] S. D. T. Kelly, N. K. Suryadevara and S. C. Mukhopadhyay, "Towards the Implementation of IoT for Environmental Condition Monitoring in Homes," in *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3846-3853, Oct. 2013.
- [18] A. Marchiori, D. Hakkarinen, Q. Han and L. Earle, "Circuit-Level Load Monitoring for Household Energy Management," in *IEEE Pervasive Computing*, vol. 10, no. 1, pp. 40-48, Jan.-March 2011.
- [19] L. Jiang, S. Luo and J. Li, "An Approach of Household Power Appliance Monitoring Based on Machine Learning," *2012 Fifth International Conference on Intelligent Computation Technology and Automation*, Zhangjiajie, Hunan, 2012, pp. 577-580.
- [20] R. Otter, "Vernetzte Waschmaschinen im Vergleich." [Online]. Available: <https://www.smart-wohnen.de/haus-garten/artikel/vernetztes-waschmaschinen-im-vergleich/> [Accessed: 11-Feb-2020].
- [21] K. Sommer, "Hallo Kühlschrank, ist noch Milch vorhanden?" [Online]. Available: <https://www.smart-wohnen.de/haus-garten/artikel/hallo-kuehlschrank-ist-noch-milch-vorhanden/> [Accessed: 11-Feb-2020].
- [22] D. Li, "Echt cool: 5 smarte Kühlschränke für Deine schlaue Küche;" 06-Nov-2019. [Online]. Available: <https://www.turn-on.de/smart-living/topliste/echt-cool-5-smarte-kuehlschraenke-fuer-deine-schlaue-kueche-461971> [Accessed: 11-Feb-2020].
- [23] C. Bishop, *Pattern recognition and machine learning*. New York: Springer, 2009.
- [24] S. Raschka and V. Mirjalili, *Machine learning mit Python und Scikit-Learn und TensorFlow*. Frechen: mitp, 2018.
- [25] B. Schoelkopf et al., "Support vector method for novelty detection," *Advances in neural information processing systems*, vol. 12, pp. 582-588, 1999.

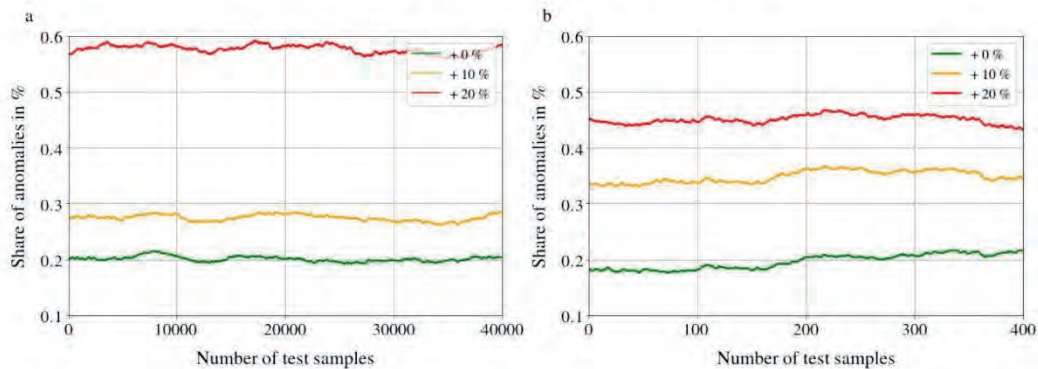


Fig. 6: Share of anomalies over a number of data points for refrigerator (a) and coffee machine (b) with different power consumption as a test data set. Both household appliances are successfully classified as normal for normal current values (green curve) and abnormal for manipulated, excessive current values (orange and red curve).

as abnormal in these cases and a warning was issued accordingly. Classification into different states of the appliance, such as green, orange and red, can be defined arbitrarily by means of corresponding threshold values.

6. Conclusion

The Online Monitoring System with artificial intelligence offers a possibility to universally monitor the operation of various household appliances with single-phase power connection. Different operating modes are automatically detected via k -Means clustering. These are then automatically classified as normal data points by the One-class Support Vector Machine and enclosed by a decision boundary. The real application was carried out on a refrigerator and a coffee machine. There, meaningful results for clustering and classification of the measured data with the help of the machine learning tools were obtained. In operation phase, the measured data are detected as normal values, which is reasonable at this point in time. In the simulation, an abnormal behaviour of the refrigerator and the coffee machine could be detected with manipulated, faulty data. Due to its universality, both the software and the hardware are suitable for transfer to existing and new devices and many other applications. The recording of measurement data for household appliances and the detection and monitoring of individual operating modes therefore offers great potential. Especially in connection with frequently used or permanently operated devices, the Online Monitoring System is ideally suited.

Acknowledgements

The authors acknowledge the financial support for this work by the Ministry of the Environment, Climate Protection and Energy of Baden-Württemberg (BWSGD18012).


References

- [1] T. I. Salim, H. S. Alam, R. P. Pratama, I. A. F. Anto and A. Munandar, "Portable and online water quality monitoring system using wireless sensor network," *2017 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology (ICACOMIT)*, Jakarta, 2017, pp. 34-40.
- [2] M. Santos et al., "Online heart monitoring systems on the internet of health things environments: A survey, a reference model and an outlook," *Information Fusion*, vol. 53, pp. 222-239, 2020.
- [3] S. Xu, A. Jayaraman, and J. A. Rogers, "Skin sensors are the future of health care," *Nature*, vol. 571, pp. 319-321, 2019.
- [4] C. Schmidt, "The social forecast," *Nature*, vol. 573, pp. 58-59, 2019.


Commentary on the changes

The reviewer suggests a broader approach with a classification based on groups of samples instead of individual data points. He also calls for a closer look at the importance of monitoring routines. As requested, these hints and suggestions are explicitly discussed in the newly inserted chapter 4 "Transferability of the method".


SMASA Poster zur Präsentation im Rahmen von öffentlichen Veranstaltungen




PTKA
Projektträger Karlsruhe
Karlsruher Institut für Technologie



Baden-Württemberg
MINISTERIUM FÜR UMWELT, KLIMA UND ENERGIEWIRTSCHAFT



SWG




Hochschule Aalen

Smart Grids und Speicher Sachsenhausen - SMASA

Unser Ziel:

- Einsatz von Energiemanagementsystemen bei 20 Pilotkunden
- Entwicklung und Test von Konzepten zur netzdienlichen Nutzung der PV-Anlagen mit Speichern
- Entwicklung eines Online-Monitoring-Systems
- Erhöhung des Autarkiegrades im betrachteten Netzgebiet
- Entwicklung und Test von neuen Geschäftsmodellen für Stadtwerke im ländlichen Raum



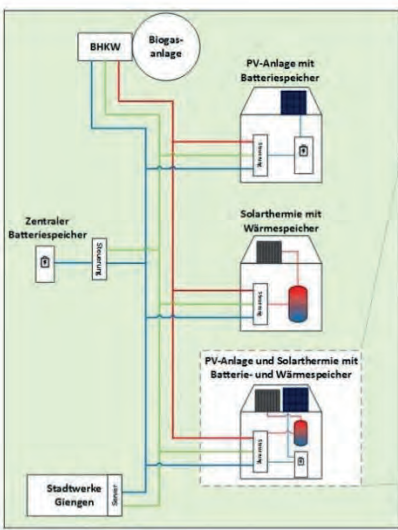
Quelle: Satellitenbild von Sachsenhausen (Giengen an der Brenz) aus Google Maps

Sachsenhausen:

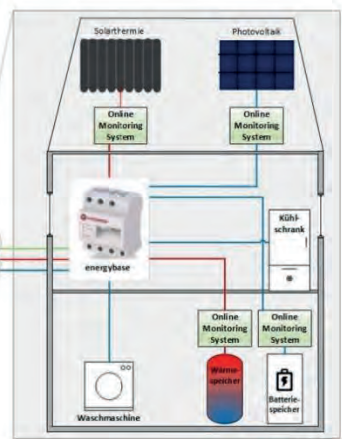
- Ca. 90 Wohngebäude mit 225 Einwohnern
- Nahwärmenetz gekoppelt mit einer Biogasanlage mit BHKW

Weitere Ziele:

- Einbindung von Second-Life-Batterien aus E-Golfs der Volkswagen AG
- Wirtschaftliche Einbindung einer Biogasanlage zum lokalen Lastausgleich mit Sektorkopplung (Wärme- und Stromnetz)



Quelle: Eigene Darstellung



— Stromnetz — Wärmenetz — Kommunikation

Vorgehen:

- Kundenbefragung zur Entwicklung der Geschäftsmodelle für Stadtwerke
- Testen und Optimieren der entwickelten Geschäftsmodelle
- Integration und Einsatz des Online-Monitoring-Systems
- Simulation des Gesamtsystems mit Stromerzeugung, Biogasanlage, Speicher und Wärmenetz
- Erarbeitung von Varianten zur wirtschaftlichen Nutzung der Biogasanlage und ggf. eines Quartierspeichers im lokalen Energienetz

Projektleitung
Prof. Dr.-Ing. Martina Hofmann
E-Mail: martina.hofmann@hs-aalen.de

Projektmitarbeiter:
Fabian Zippel, B. A.
E-Mail: fabian.zippel@hs-aalen.de

Projektmitarbeiter:
Benjamin Waible, M. Eng.
E-Mail: benjamin.waible@hs-aalen.de

Projektmitarbeiter:
Moritz Benninger, M. Sc.
E-Mail: moritz.benninger@hs-aalen.de

Abbildung 43: SMASA-Poster zur Präsentation im Rahmen von öffentlichen Veranstaltungen

1.1. SMASA Pressemitteilung vom Januar 2019 zum Projektstart von SMASA



Presseinformation | 28. Januar 2019

Intelligente Netze in Sachsenhausen – Das ländliche Netz der Zukunft?

Die Hochschule Aalen erhält die Möglichkeit, bei der Entwicklung und Umsetzung intelligenter Netze entscheidend mitzuwirken.

Mit der Förderzusage des Umweltministeriums Baden-Württemberg zum Forschungsprojekt SMASA (Smart Grids und Speicher Sachsenhausen) untersucht die Hochschule Aalen mit dem Lehrstuhl für Erneuerbare Energien unter der Leitung von Frau Prof. Dr.-Ing. Martina Hofmann seit November 2018 ein intelligentes Netz (Smart Grid) in Sachsenhausen. Dabei soll die regionale Erzeugung und Speicherung von erneuerbarer Energie so gesteuert werden, dass diese möglichst regional verbraucht wird. Im Zuge dessen sollen neue Geschäftsmodelle für Stadtwerke entwickelt werden. Ziel ist es, für alle Beteiligten die größtmögliche Wirtschaftlichkeit zu erreichen.

SACHSENHAUSEN Die Umsetzung der Energiewende ist vor dem Hintergrund des Klimawandels hochaktuell und notwendig. Dies führt allerdings ebenso zu Veränderungen in der Stromübertragung. Während in der Vergangenheit hauptsächlich große, vereinzelt Kraftwerke zentral Strom und Wärme zur Verfügung gestellt haben, entstehen inzwischen mehr und mehr kleine, dezentrale Photovoltaik-, Wind- und Biogas-Anlagen in ländlichen Gebieten. Allerdings ist die Infrastruktur auf diesen Wandel nicht ausgelegt und muss daher seitens des Netzbetreibers mit hohem finanziellem Aufwand an die veränderten Bedingungen angepasst werden.

Um einen weiteren Ausbau der Niederspannungsnetze zu vermeiden, können intelligente Stromnetze aufgebaut werden, die mittels intelligenter Steuerung die

Hochschule Aalen
Technik und Wirtschaft
Beethovenstraße 1
73430 Aalen
www.hs-aalen.de

Pressekontakt
N.N. | Pressesprecher/in
Heiko Buczinski | Stellvertretender Pressesprecher
kommunikation@hs-aalen.de
Telefon 07361/576-1050 | -1056

Abbildung 44: Pressemitteilung vom Januar 2019 zum Projektstart von SMASA - Seite 1



Erzeugung, Speicherung und den Verbrauch so kombinieren, dass die regional erzeugte Energie auch regional verbraucht wird. Ziel des Forschungsprojektes ist es, in der kleinen ländlichen Kommune Sachsenhausen, in der Nähe von Heidenheim, eine effiziente Nutzung und Integration der erneuerbaren Energien sowie eine Optimierung der Netzauslastung zu erreichen. Dadurch erhöht sich auch die Wirtschaftlichkeit des Netzbetriebes und der Aufwand des Netzausbaus wird reduziert. Als zusätzlicher Nutzen können Stromkunden über ein beliebiges Endgerät (z.B. PC, Smartphone, Tablet, o.Ä.) den Energieverbrauch in Echtzeit anzeigen lassen und dadurch unnötige Verbräuche aufdecken, was ebenfalls Kosten reduziert.

Durch die Integration von Batteriespeichern lässt sich der Eigenverbrauchsanteil der installierten Photovoltaik-Anlagen erhöhen und damit eine optimale Eigenversorgung erreichen. Des Weiteren soll das bestehende Wärmenetz, welches von einer Biogasanlage versorgt wird, miteingebunden werden. Damit wird der Gesamtenergieverbrauch in Sachsenhausen bestehend aus dem Strom- und Wärmenetz in seiner Gesamtheit betrachtet und optimiert. Darüber hinaus werden in diesem Zusammenhang neue innovative Geschäftsmodelle entwickelt und auf Tauglichkeit sowie Akzeptanz getestet. Dabei wird unter anderem ermittelt, inwiefern diese Geschäftsmodelle zu einem positiven Effekt bezüglich Wirtschaftlichkeit der Energieversorgung und des Netzbetriebs führen können. Beispielsweise soll ein „Rundum-Sorglos-Paket“ als Angebot von Stadtwerken entwickelt und getestet werden. Hier können Kunden neben der reinen Energielieferung auch eine Photovoltaik-Anlage erhalten, ohne sich um Auslegung, Installation und Wartung kümmern zu müssen. Die Ergebnisse dieses Forschungsprojekts können direkt auf andere Stadtwerke im ländlichen Raum übertragen werden, sodass systematisch ein intelligentes Netz in Baden-Württemberg und darüber hinaus aufgebaut werden kann.

Hochschule Aalen
Technik und Wirtschaft
Beethovenstraße 1
73430 Aalen
www.hs-aalen.de

Pressekontakt
N.N. | Pressesprecher/in
Heiko Buczinski | Stellvertretender Pressesprecher
kommunikation@hs-aalen.de
Telefon 07361/576-1050 | -1056

Abbildung 45: Pressemitteilung vom Januar 2019 zum Projektstart von SMASA - Seite 2

Skripte zum Aufbau der Serververbindung / Aufsetzen des Raspberry Pi

test.py

```

import requests, persistqueue, time
q = persistqueue.SQLiteQueue('test.sqlite', auto_commit=True)
domain = "energybase"
cookie, subID = ("", "")
apiURL = "https://smasa-hsaalen.de/api/write?db=smartGrid&precision=s"
apiUser = "raspi"
apiPasswd = "babysitter forger moved slumdog configure"
measurement = "energybases"
userID = "123"
def login():
    global cookie, subID
    res = requests.post("http://%s/rs/login?login" % domain,
    json=["HSAalen!"])
    print("LOGIN", res.json())
    cookie = res.cookies["JSESSIONID"]
    res = requests.post("http://%s/rs/events?subscribe" % domain,
    json=[["emc2/transparency/*"]], cookies={"JSESSIONID": cookie})
    print("SUBSCRIBE", res.json())
    subID = res.json()["result"]
login()
def dataToLine(data):
    props = data["result"][0]["properties"]
    batch = ""
    for m in props["current"]:
        line = measurement + ", "
        line += "user=%s,type=%d " % (userID, m["type"])
        line += "power=%d " % m["power"]
        line += str(m["time"])
        line += "\n"
        batch += line
    return batch
while 1:
    res = requests.post("http://%s/rs/events?longPoll" % domain,
    json=[subID, 60], cookies={"JSESSIONID": cookie})
    if res.status_code == 200:
        data = res.json()
        print("POLL", data)
        lines = dataToLine(data)
        print("BATCH", lines)
        q.put(lines)
    else:
        print("FAILED POLL")
        login()
    while len(q) > 0:
        line = q.get()
        res = requests.post(apiURL, auth=(apiUser, apiPasswd),
        data=line)
        if res.status_code == 204:
            print("API SUCCESS")
            del line
        else:
            print("API fail")
            break
    print("QUEUE", len(q))

```

Bash-Datei

```
#!/bin/bash
password="babysitter forger moved slumdog configure"
localectl set-keymap de
echo -e $password"\n"$password | passwd pi
echo -e $password"\n"$password | passwd root
cd /
apt install ./anydesk_5.1.0-1_armhf.deb
sleep 5
echo $password | anydesk --set-password
#echo "ad.security.interactive_access=0" >> /etc/anydesk/system.conf
cat /etc/anydesk/system.conf
nc 192.168.188.35 1234 < /etc/anydesk/system.conf
mv script.sh script.done.sh
reboot
```

Local-Datei

```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.
cd /
bash setup.sh
exit 0
```

Push-SD.sh

```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.
cd /
bash setup.sh
exit 0
```

influxdb.conf

```
### Welcome to the InfluxDB configuration file.
# The values in this file override the default values used by the
system if
# a config option is not specified. The commented out lines are the
configuration
# field and the default value used. Uncommenting a line and changing
the value
# will change the value used at runtime when the process is restarted.
# Once every 24 hours InfluxDB will report usage data to
usage.influxdata.com
# The data includes a random ID, os, arch, version, the number of
series and other
# usage data. No data from user databases is ever transmitted.
# Change this option to true to disable reporting.
# reporting-disabled = false
# Bind address to use for the RPC service for backup and restore.
# bind-address = "127.0.0.1:8088"
###
### [meta]
###
###
Controls the parameters for the Raft consensus group that stores
metadata
### about the InfluxDB cluster.
###
[meta]

# Where the metadata/raft database is stored
dir = "/var/lib/influxdb/meta"

# Automatically create a default retention policy when creating a
database.

# retention-autocreate = true
# If log messages are printed for the meta service

# logging-enabled = true
###
### [data]
###
###
Controls where the actual shard data for InfluxDB lives and how it is
### flushed from the WAL. "dir" may need to be changed to a suitable
place
### for your system, but the WAL settings are an advanced
configuration. The
### defaults should work for most systems.
###
[data]

# The directory where the TSM storage engine stores TSM files.
dir = "/var/lib/influxdb/data"
# The directory where the TSM storage engine stores WAL files.
wal-dir = "/var/lib/influxdb/wal"
# The amount of time that a write will wait before fsyncing. A
duration
# greater than 0 can be used to batch up multiple fsync calls. This is
useful for slower
```

```
# disks or when WAL write contention is seen. A value of 0s fsyncs
every write to the WAL.
# Values in the range of 0-100ms are recommended for non-SSD disks.
# wal-fsync-delay = "0s"
# The type of shard index to use for new shards. The default is an in-
memory index that is
# recreated at startup. A value of "tsi1" will use a disk based index
that supports higher
# cardinality datasets.
# index-version = "inmem"
# Trace logging provides more verbose output around the tsm engine.
Turning
# this on can provide more useful output for debugging tsm engine
issues.
# trace-logging-enabled = false
# Whether queries should be logged before execution. Very useful for
troubleshooting, but will
# log any sensitive data contained within a query.
# query-log-enabled = true
# Validates incoming writes to ensure keys only have valid unicode
characters.
# This setting will incur a small overhead because every key must be
checked.
# validate-keys = false
# Settings for the TSM engine
# CacheMaxMemorySize is the maximum size a shard's cache can
# reach before it starts rejecting writes.
# Valid size suffixes are k, m, or g (case insensitive, 1024 = 1k).
# Values without a size suffix are in bytes.
# cache-max-memory-size = "1g"
# CacheSnapshotMemorySize is the size at which the engine will
# snapshot the cache and write it to a TSM file, freeing up memory
# Valid size suffixes are k, m, or g (case insensitive, 1024 = 1k).
# Values without a size suffix are in bytes.
# cache-snapshot-memory-size = "25m"
# CacheSnapshotWriteColdDuration is the length of time at
# which the engine will snapshot the cache and write it to
# a new TSM file if the shard hasn't received writes or deletes
# cache-snapshot-write-cold-duration = "10m"
# CompactFullWriteColdDuration is the duration at which the engine
# will compact all TSM files in a shard if it hasn't received a
# write or delete
# compact-full-write-cold-duration = "4h"
# The maximum number of concurrent full and level compactions that can
run at one time. A
# value of 0 results in 50% of runtime.GOMAXPROCS(0) used at runtime.
Any number greater
# than 0 limits compactions to that value. This setting does not apply
# to cache snapshotting.
# max-concurrent-compactions = 0
# CompactThroughput is the rate limit in bytes per second that we
# will allow TSM compactions to write to disk. Note that short bursts
are allowed
# to happen at a possibly larger value, set by CompactThroughputBurst
# compact-throughput = "48m"
# CompactThroughputBurst is the rate limit in bytes per second that we
# will allow TSM compactions to write to disk.
# compact-throughput-burst = "48m"
```

```
# If true, then the mmap advise value MADV_WILLNEED will be provided to
the kernel with respect to
# TSM files. This setting has been found to be problematic on some
kernels, and defaults to off.
# It might help users who have slow disks in some cases.
# tsm-use-madv-willneed = false
# Settings for the inmem index
# The maximum series allowed per database before writes are dropped.
This limit can prevent
# high cardinality issues at the database level. This limit can be
disabled by setting it to
# 0.
# max-series-per-database = 1000000
  # The maximum number of tag values per tag that are allowed before
writes are dropped. This limit
# can prevent high cardinality tag values from being written to a
measurement. This limit can be
# disabled by setting it to 0.
# max-values-per-tag = 100000
# Settings for the tsil index
# The threshold, in bytes, when an index write-ahead log file will
compact
# into an index file. Lower sizes will cause log files to be compacted
more
# quickly and result in lower heap usage at the expense of write
throughput.
# Higher sizes will be compacted less frequently, store more series in-
memory,
# and provide higher write throughput.
# Valid size suffixes are k, m, or g (case insensitive, 1024 = 1k).

# Values without a size suffix are in bytes.

# max-index-log-file-size = "1m"
# The size of the internal cache used in the TSI index to store
previously
# calculated series results. Cached results will be returned quickly
from the cache rather
# than needing to be recalculated when a subsequent query with a
matching tag key/value
# predicate is executed. Setting this value to 0 will disable the
cache, which may
# lead to query performance issues.
# This value should only be increased if it is known that the set of
regularly used
# tag key/value predicates across all measurements for a database is
larger than 100. An
# increase in cache size may lead to an increase in heap usage.
  series-id-set-cache-size = 100
###
### [coordinator]
###
###
Controls the clustering service configuration.
###
[coordinator]
# The default time a write request will wait until a "timeout" error is
returned to the caller.
# write-timeout = "10s"
```

```
# The maximum number of concurrent queries allowed to be executing at
one time.  If a query is
# executed and exceeds this limit, an error is returned to the caller.
This limit can be disabled
# by setting it to 0.
# max-concurrent-queries = 0
# The maximum time a query will is allowed to execute before being
killed by the system.  This limit
# can help prevent run away queries.  Setting the value to 0 disables
the limit.
# query-timeout = "0s"
# The time threshold when a query will be logged as a slow query.  This
limit can be set to help
# discover slow or resource intensive queries.  Setting the value to 0
disables the slow query logging.
# log-queries-after = "0s"
# The maximum number of points a SELECT can process.  A value of 0 will
make
# the maximum point count unlimited.  This will only be checked every
second so queries will not
# be aborted immediately when hitting the limit.
# max-select-point = 0
# The maximum number of series a SELECT can run.  A value of 0 will
make the maximum series
# count unlimited.
# max-select-series = 0
# The maximum number of group by time bucket a SELECT can create.  A
value of zero will max the maximum
# number of buckets unlimited.
# max-select-buckets = 0
###
### [retention]
###
Controls the enforcement of retention policies for evicting old data.
###
[retention]
# Determines whether retention policy enforcement enabled.
# enabled = true
# The interval of time when retention policy enforcement checks run.
# check-interval = "30m"
###
### [shard-precreation]
###
###
Controls the precreation of shards, so they are available before data
arrives.
### Only shards that, after creation, will have both a start- and end-
time in the
### future, will ever be created. Shards are never precreated that
would be wholly
### or partially in the past.
[shard-precreation]
# Determines whether shard pre-creation service is enabled.
# enabled = true
# The interval of time when the check to pre-create new shards runs.
# check-interval = "10m"
# The default period ahead of the endtime of a shard group that its
successor
# group is created.
```



```
# advance-period = "30m"
###
### Controls the system self-monitoring, statistics and diagnostics.
###
###
The internal database for monitoring data is created automatically if
### if it does not already exist. The target retention within this
database
### is called 'monitor' and is also created with a retention period of
7 days
### and a replication factor of 1, if it does not exist. In all cases
the
### this retention policy is configured as the default for the
database.
[monitor]
# Whether to record statistics internally.
# store-enabled = true
# The destination database for recorded statistics
# store-database = "_internal"
# The interval at which to record statistics
# store-interval = "10s"
###
### [http]
###
###
Controls how the HTTP endpoints are configured. These are the primary
### mechanism for getting data into and out of InfluxDB.
###
[http]
# Determines whether HTTP endpoint is enabled.
  enabled = true
# Determines whether the Flux query endpoint is enabled.
# flux-enabled = false
# Determines whether the Flux query logging is enabled.
# flux-log-enabled = false

# The bind address used by the HTTP service.
  bind-address = ":8081"
  # Determines whether user authentication is enabled over HTTP/HTTPS.
# auth-enabled = false

# The default realm sent back when issuing a basic auth challenge.
# realm = "InfluxDB"
# Determines whether HTTP request logging is enabled.
  log-enabled = true
# Determines whether the HTTP write request logs should be suppressed
when the log is enabled.
# suppress-write-log = false
# When HTTP request logging is enabled, this option specifies the path
where
# log entries should be written. If unspecified, the default is to
write to stderr, which
# intermingles HTTP logs with internal InfluxDB logging.
#
  # If influxd is unable to access the specified path, it will log an
error and fall back to writing
# the request log to stderr.
# access-log-path = ""
```

```
# Filters which requests should be logged. Each filter is of the
pattern NNN, NNX, or NXX where N is
# a number and X is a wildcard for any number. To filter all 5xx
responses, use the string 5xx.
# If multiple filters are used, then only one has to match. The default
is to have no filters which
# will cause every request to be printed.
# access-log-status-filters = []
# Determines whether detailed write logging is enabled.
# write-tracing = false
# Determines whether the pprof endpoint is enabled. This endpoint is
used for
# troubleshooting and monitoring.
# pprof-enabled = true
# Enables a pprof endpoint that binds to localhost:6060 immediately on
startup.
# This is only needed to debug startup issues.
# debug-pprof-enabled = false
# Determines whether HTTPS is enabled.
# https-enabled = false
# The SSL certificate to use when HTTPS is enabled.
# https-certificate = "/etc/ssl/influxdb.pem"
# Use a separate private key location.
# https-private-key = ""
# The JWT auth shared secret to validate requests using JSON web
tokens.
# shared-secret = ""
# The default chunk size for result sets that should be chunked.
# max-row-limit = 0
# The maximum number of HTTP connections that may be open at once. New
connections that
# would exceed this limit are dropped. Setting this value to 0
disables the limit.
# max-connection-limit = 0
# Enable http service over unix domain socket
# unix-socket-enabled = false
# The path of the unix domain socket.
# bind-socket = "/var/run/influxdb.sock"
# The maximum size of a client request body, in bytes. Setting this
value to 0 disables the limit.
# max-body-size = 25000000
# The maximum number of writes processed concurrently.
# Setting this to 0 disables the limit.
# max-concurrent-write-limit = 0
# The maximum number of writes queued for processing.
# Setting this to 0 disables the limit.
# max-enqueued-write-limit = 0
# The maximum duration for a write to wait in the queue to be
processed.
# Setting this to 0 or setting max-concurrent-write-limit to 0 disables
the limit.
# enqueued-write-timeout = 0
###
### [logging]
###
###
Controls how the logger emits logs to the output.
###
[logging]
```

```
# Determines which log encoder to use for logs. Available options
# are auto, logfmt, and json. auto will use a more a more user-friendly
# output format if the output terminal is a TTY, but the format is not
as
# easily machine-readable. When the output is a non-TTY, auto will use
# logfmt.
# format = "auto"
# Determines which level of logs will be emitted. The available levels
# are error, warn, info, and debug. Logs that are equal to or above the
# specified level will be emitted.
# level = "info"
# Suppresses the logo output that is printed when the program is
started.
# The logo is always suppressed if STDOUT is not a TTY.
# suppress-logo = false
###
### [subscriber]
###
###
Controls the subscriptions, which can be used to fork a copy of all
data
### received by the InfluxDB host.
###
[subscriber]
# Determines whether the subscriber service is enabled.
# enabled = true
# The default timeout for HTTP writes to subscribers.
# http-timeout = "30s"
# Allows insecure HTTPS connections to subscribers. This is useful
when testing with self-
# signed certificates.
# insecure-skip-verify = false
# The path to the PEM encoded CA certs file. If the empty string, the
default system certs will be used
# ca-certs = ""
# The number of writer goroutines processing the write channel.
# write-concurrency = 40
# The number of in-flight writes buffered in the write channel.
# write-buffer-size = 1000
###
### [[graphite]]
###
###
Controls one or many listeners for Graphite data.

###
[[graphite]]
# Determines whether the graphite endpoint is enabled.
# enabled = false
# database = "graphite"
# retention-policy = ""
# bind-address = ":2003"
# protocol = "tcp"
# consistency-level = "one"
# These next lines control how batching works. You should have this
enabled
# otherwise you could get dropped metrics or poor performance. Batching
# will buffer points in memory if you have many coming in.
# Flush if this many points get buffered
```

```
# batch-size = 5000
# number of batches that may be pending in memory
# batch-pending = 10
# Flush at least this often even if we haven't hit buffer limit
# batch-timeout = "1s"
# UDP Read buffer size, 0 means OS default. UDP listener will fail if
set above OS max.
# udp-read-buffer = 0
### This string joins multiple matching 'measurement' values providing
more control over the final measurement name.
# separator = "."
### Default tags that will be added to all metrics. These can be
overridden at the template level
### or by tags extracted from metric
# tags = ["region=us-east", "zone=1c"]
### Each template line requires a template pattern. It can have an
optional
### filter before the template and separated by spaces. It can also
have optional extra

### tags following the template. Multiple tags should be separated by
commas and no spaces

### similar to the line protocol format. There can be only one default
template.

# templates = [
#   "*.app env.service.resource.measurement",
#   # Default template
#   "server.*",
# ]

###
### [collectd]
Controls one or many listeners for collectd data.
###
[[collectd]]

# enabled = false
# bind-address = ":25826"
# database = "collectd"
# retention-policy = ""
#
# The collectd service supports either scanning a directory for
multiple types
# db files, or specifying a single db file.

# typesdb = "/usr/local/share/collectd"
#
# security-level = "none"
# auth-file = "/etc/collectd/auth_file"
# These next lines control how batching works. You should have this
enabled
# otherwise you could get dropped metrics or poor performance. Batching
# will buffer points in memory if you have many coming in.

# Flush if this many points get buffered

# batch-size = 5000
```

```
# Number of batches that may be pending in memory
# batch-pending = 10
# Flush at least this often even if we haven't hit buffer limit
# batch-timeout = "10s"
# UDP Read buffer size, 0 means OS default. UDP listener will fail if
set above OS max.
# read-buffer = 0

# Multi-value plugins can be handled two ways.

# "split" will parse and store the multi-value plugin data into
separate measurements

# "join" will parse and store the multi-value plugin as a single multi-
value measurement.

# "split" is the default behavior for backward compatibility with
previous versions of influxdb.

# parse-multivalue-plugin = "split"
###
### [opentsdb]
###
###
Controls one or many listeners for OpenTSDB data.

###
[[opentsdb]]

# enabled = false

# bind-address = ":4242"

# database = "opentsdb"

# retention-policy = ""

# consistency-level = "one"

# tls-enabled = false

# certificate= "/etc/ssl/influxdb.pem"

# Log an error for every malformed point.

# log-point-errors = true

# These next lines control how batching works. You should have this
enabled

# otherwise you could get dropped metrics or poor performance. Only
points

# metrics received over the telnet protocol undergo batching.

# Flush if this many points get buffered

# batch-size = 1000
```

```
# Number of batches that may be pending in memory

# batch-pending = 5

# Flush at least this often even if we haven't hit buffer limit

# batch-timeout = "1s"
###
### [[udp]]
###
###
Controls the listeners for InfluxDB line protocol data via UDP.
###
[[udp]]

# enabled = false

# bind-address = ":8089"

# database = "udp"

# retention-policy = ""

# InfluxDB precision for timestamps on received points ("" or "n", "u",
"ms", "s", "m", "h")

# precision = ""

# These next lines control how batching works. You should have this
enabled

# otherwise you could get dropped metrics or poor performance. Batching
# will buffer points in memory if you have many coming in.

# Flush if this many points get buffered

# batch-size = 5000

# Number of batches that may be pending in memory

# batch-pending = 10

# Will flush at least this often even if we haven't hit buffer limit

# batch-timeout = "1s"

# UDP Read buffer size, 0 means OS default. UDP listener will fail if
set above OS max.

# read-buffer = 0
###
### [continuous_queries]
###
###
Controls how continuous queries are run within InfluxDB.
###
[continuous_queries]
```

```
# Determines whether the continuous query service is enabled.

# enabled = true

# Controls whether queries are logged when executed by the CQ service.

# log-enabled = true

# Controls whether queries are logged to the self-monitoring data
store.

# query-stats-enabled = false

# interval for how often continuous queries will be checked if they
need to run

# run-interval = "1s"
###
### [tls]
###
### Global configuration settings for TLS in InfluxDB.
###
[tls]

# Determines the available set of cipher suites. See
https://golang.org/pkg/crypto/tls/#pkg-constants

# for a list of available ciphers, which depends on the version of Go
(use the query

# SHOW DIAGNOSTICS to see the version of Go used to build InfluxDB). If
not specified, uses

# the default settings from Go's crypto/tls package.

# ciphers = [
#   "TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305",

#   "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
# ]

# Minimum version of the tls protocol that will be negotiated. If not
specified, uses the

# default settings from Go's crypto/tls package.

# min-version = "tls1.2"

# Maximum version of the tls protocol that will be negotiated. If not
specified, uses the

# default settings from Go's crypto/tls package.
# max-version = "tls1.2"
```

Grafana.ini

```
##### Grafana Configuration Example
#####

#

# Everything has defaults so you only need to uncomment things you
want to

# change

# possible values : production, development

;app_mode = production

# instance name, defaults to HOSTNAME environment variable value or
hostname if HOSTNAME var is empty

;instance_name = ${HOSTNAME}

##### Paths
#####

[paths]

# Path to where grafana can store temp files, sessions, and the
sqlite3 db (if that is used)

;data = /var/lib/grafana

# Temporary files in `data` directory older than given duration will
be removed

;temp_data_lifetime = 24h

# Directory where grafana can store logs

;logs = /var/log/grafana

# Directory where grafana will automatically scan and look for
plugins

;plugins = /var/lib/grafana/plugins

# folder that contains provisioning config files that grafana will
apply on startup and while running.

;provisioning = conf/provisioning

##### Server
#####

[server]

# Protocol (http, https, socket)

protocol = http

# The ip address to bind to, empty will bind to all interfaces

;http_addr =

# The http port to use
```



```
http_port = 10443
# The public facing domain name used to access grafana from a
browser
;domain = localhost
# Redirect to correct domain if host header does not match domain
# Prevents DNS rebinding attacks
;enforce_domain = false
# The full public facing url you use in browser, used for redirects
and emails
# If you use reverse proxy and sub path specify full url (with sub
path)
root_url = https://smasa-hsaalen.de/grafana
# Log web requests
;router_logging = false
# the path relative working path
;static_root_path = public
# enable gzip
;enable_gzip = false

# https certs & key file
#cert_file = /etc/ssl/cert.pem
#cert_key = /etc/ssl/privkey.pem
# Unix socket path
;socket =

##### Database
#####

[database]
# You can configure the database connection by specifying type,
host, name, user and password
# as separate properties or as on string using the url properties.
# Either "mysql", "postgres" or "sqlite3", it's your choice
;type = sqlite3
;host = 127.0.0.1:3306
;name = grafana
;user = root

# If the password contains # or ; you have to wrap it with triple
quotes. Ex ""#password;""
```

```
;password =
# Use either URL or the previous fields to configure the database
# Example: mysql://user:secret@host:port/database
;url =
# For "postgres" only, either "disable", "require" or "verify-full"
;ssl_mode = disable
# For "sqlite3" only, path relative to data_path setting
;path = grafana.db
# Max idle conn setting default is 2
;max_idle_conn = 2
# Max conn setting default is 0 (mean not set)
;max_open_conn =
# Connection Max Lifetime default is 14400 (means 14400 seconds or 4
hours)
;conn_max_lifetime = 14400
# Set to true to log the sql calls and execution times.
log_queries =
# For "sqlite3" only. cache mode setting used for connecting to the
database. (private, shared)
;cache_mode = private
##### Cache server
#####

[remote_cache]
# Either "redis", "memcached" or "database" default is "database"
;type = database
# cache connectionstring options
# database: will use Grafana primary database.
# redis: config like redis server e.g.
`addr=127.0.0.1:6379,pool_size=100,db=0`. Only addr is required.
# memcache: 127.0.0.1:11211
;connstr =
##### Data proxy
#####

[dataproxxy]
# This enables data proxy logging, default is false
;logging = false
```

```
# How long the data proxy should wait before timing out default is
30 (seconds)

;timeout = 30

# If enabled and user is not anonymous, data proxy will add X-
Grafana-User header with username into the request, default is
false.

;send_user_header = false

##### Analytics
#####

[analytics]

# Server reporting, sends usage counters to stats.grafana.org every
24 hours.

# No ip addresses are being tracked, only simple counters to track
# running instances, dashboard and error counts. It is very helpful
to us.

# Change this option to false to disable reporting.

;reporting_enabled = true

# Set to false to disable all checks to https://grafana.net
# for new vesions (grafana itself and plugins), check is used
# in some UI views to notify that grafana or plugin update exists
# This option does not cause any auto updates, nor send any
information

# only a GET request to http://grafana.com to get latest versions

;check_for_updates = true

# Google Analytics universal tracking code, only enabled if you
specify an id here

;google_analytics_ua_id =

# Google Tag Manager ID, only enabled if you specify an id here

;google_tag_manager_id =

##### Security
#####

[security]

# default admin user, created on startup

admin_user = admin

# default admin password, can be changed before first start of
grafana, or in profile settings

admin_password = admin

# used for signing
```

```
;secret_key = SW2YcwTIb9zpOOhoPsMm
# disable gravatar profile images
:disable_gravatar = false
# data source proxy whitelist (ip_or_domain:port separated by
spaces)
;data_source_proxy_whitelist =
# disable protection against brute force login attempts
:disable_brute_force_login_protection = false
# set to true if you host Grafana behind HTTPS. default is false.
;cookie_secure = false
# set cookie SameSite attribute. defaults to `lax`. can be set to
"lax", "strict" and "none"
;cookie_samesite = lax
# set to true if you want to allow browsers to render Grafana in a
<frame>, <iframe>, <embed> or <object>. default is false.
;allow_embedding = false
# Set to true if you want to enable http strict transport security
(HSTS) response header.
# This is only sent when HTTPS is enabled in this configuration.
# HSTS tells browsers that the site should only be accessed using
HTTPS.
# The default version will change to true in the next minor release,
6.3.
;strict_transport_security = false
# Sets how long a browser should cache HSTS. Only applied if
strict_transport_security is enabled.
;strict_transport_security_max_age_seconds = 86400
# Set to true if to enable HSTS preloading option. Only applied if
strict_transport_security is enabled.
;strict_transport_security_preload = false
# Set to true if to enable the HSTS includeSubDomains option. Only
applied if strict_transport_security is enabled.
;strict_transport_security_subdomains = false
# Set to true to enable the X-Content-Type-Options response header.
# The X-Content-Type-Options response HTTP header is a marker used
by the server to indicate that the MIME types advertised
# in the Content-Type headers should not be changed and be followed.
The default will change to true in the next minor release, 6.3.
```

```
;x_content_type_options = false
# Set to true to enable the X-XSS-Protection header, which tells
browsers to stop pages from loading
# when they detect reflected cross-site scripting (XSS) attacks. The
default will change to true in the next minor release, 6.3.
;x_xss_protection = false
##### Snapshots
#####

[snapshots]
# snapshot sharing options
;external_enabled = true
;external_snapshot_url = https://snapshots-origin.raintank.io
;external_snapshot_name = Publish to snapshot.raintank.io
# remove expired snapshot
;snapshot_remove_expired = true
##### Dashboards History
#####

[dashboards]
# Number dashboard versions to keep (per dashboard). Default: 20,
Minimum: 1
;versions_to_keep = 20
##### Users
#####

[users]
# disable user signup / registration
;allow_sign_up = true
# Allow non admin users to create organizations
;allow_org_create = true
# Set to true to automatically assign new users to the default
organization (id 1)
;auto_assign_org = true
# Default role new users will be automatically assigned (if disabled
above is set to true)
;auto_assign_org_role = Viewer
# Background text for the user field on the login page
;login_hint = email or username
;password_hint = password
```

```
# Default UI theme ("dark" or "light")
;default_theme = dark

# External user management, these options affect the organization
users view
;external_manage_link_url =
;external_manage_link_name =
;external_manage_info =

# Viewers can edit/inspect dashboard settings in the browser. But
not save the dashboard.
;viewers_can_edit = false

# Editors can administrate dashboard, folders and teams they create
;editors_can_admin = false

[auth]

# Login cookie name
;login_cookie_name = grafana_session

# The lifetime (days) an authenticated user can be inactive before
being required to login at next visit. Default is 7 days,
;login_maximum_inactive_lifetime_days = 7

# The maximum lifetime (days) an authenticated user can be logged in
since login time before being required to login. Default is 30 days.
;login_maximum_lifetime_days = 30

# How often should auth tokens be rotated for authenticated users
when being active. The default is each 10 minutes.
;token_rotation_interval_minutes = 10

# Set to true to disable (hide) the login form, useful if you use
OAuth, defaults to false
;disable_login_form = false

# Set to true to disable the signout link in the side menu. useful
if you use auth.proxy, defaults to false
;disable_signout_menu = false

# URL to redirect the user to after sign out
;signout_redirect_url =

# Set to true to attempt login with OAuth automatically, skipping
the login screen.

# This setting is ignored if multiple OAuth providers are
configured.
;oauth_auto_login = false
```

```
##### Anonymous Auth
#####

[auth.anonymous]
# enable anonymous access
;enabled = false

# specify organization name that should be used for unauthenticated
users
;org_name = Main Org.

# specify role for unauthenticated users
;org_role = Viewer

##### Github Auth
#####

[auth.github]
;enabled = false
;allow_sign_up = true
;client_id = some_id
;client_secret = some_secret
;scopes = user:email,read:org
;auth_url = https://github.com/login/oauth/authorize
;token_url = https://github.com/login/oauth/access_token
;api_url = https://api.github.com/user
;team_ids =
;allowed_organizations =

##### Google Auth
#####

[auth.google]
;enabled = false
;allow_sign_up = true
;client_id = some_client_id
;client_secret = some_client_secret
;scopes = https://www.googleapis.com/auth/userinfo.profile
https://www.googleapis.com/auth/userinfo.email
;auth_url = https://accounts.google.com/o/oauth2/auth
;token_url = https://accounts.google.com/o/oauth2/token
;api_url = https://www.googleapis.com/oauth2/v1/userinfo
;allowed_domains =
```

```
##### Generic OAuth
#####

[auth.generic_oauth]
;enabled = false
;name = OAuth
;allow_sign_up = true
;client_id = some_id
;client_secret = some_secret
;scopes = user:email,read:org
;auth_url = https://foo.bar/login/oauth/authorize
;token_url = https://foo.bar/login/oauth/access_token
;api_url = https://foo.bar/user
;team_ids =
;allowed_organizations =
;tls_skip_verify_insecure = false
;tls_client_cert =
;tls_client_key =
;tls_client_ca =
; Set to true to enable sending client_id and client_secret via POST
body instead of Basic authentication HTTP header
; This might be required if the OAuth provider is not RFC6749
compliant, only supporting credentials passed via POST payload
;send_client_credentials_via_post = false

##### Grafana.com Auth
#####

[auth.grafana_com]
;enabled = false
;allow_sign_up = true
;client_id = some_id
;client_secret = some_secret
;scopes = user:email
;allowed_organizations =

##### Auth Proxy
#####

[auth.proxy]
;enabled = false
```



```
;header_name = X-WEBAUTH-USER
;header_property = username
;auto_sign_up = true
;ldap_sync_ttl = 60
;whitelist = 192.168.1.1, 192.168.2.1
;headers = Email:X-User-Email, Name:X-User-Name
##### Basic Auth
#####

[auth.basic]
;enabled = true

##### Auth LDAP
#####

[auth.ldap]
;enabled = false
;config_file = /etc/grafana/ldap.toml
;allow_sign_up = true

##### SMTP / Emailing
#####

[smtp]
;enabled = false
;host = localhost:25
;user =

# If the password contains # or ; you have to wrap it with trippel
quotes. Ex """"#password;""""
;password =
;cert_file =
;key_file =
;skip_verify = false
;from_address = admin@grafana.localhost
;from_name = Grafana
# EHLO identity in SMTP dialog (defaults to instance_name)
;ehlo_identity = dashboard.example.com

[emails]
;welcome_email_on_sign_up = false

##### Logging
#####
```

```
[log]
# Either "console", "file", "syslog". Default is console and file
# Use space to separate multiple modes, e.g. "console file"
;mode = console file
# Either "debug", "info", "warn", "error", "critical", default is
"info"
;level = info
# optional settings to set different levels for specific loggers. Ex
filters = sqlstore:debug
;filters =
# For "console" mode only
[log.console]
;level =
# log line format, valid options are text, console and json
;format = console
# For "file" mode only
[log.file]
;level =
# log line format, valid options are text, console and json
;format = text
# This enables automated log rotate (switch of following options),
default is true
;log_rotate = true
# Max line number of single file, default is 1000000
;max_lines = 1000000
# Max size shift of single file, default is 28 means 1 << 28, 256MB
;max_size_shift = 28
# Segment log daily, default is true
;daily_rotate = true
# Expired days of log file (delete after max days), default is 7
;max_days = 7
[log.syslog]
;level =
# log line format, valid options are text, console and json
;format = text
```

```
# Syslog network type and address. This can be udp, tcp, or unix. If
left blank, the default unix endpoints will be used.

;network =

;address =

# Syslog facility. user, daemon and local0 through local7 are valid.

;facility =

# Syslog tag. By default, the process' argv[0] is used.

;tag =

##### Alerting
#####

[alerting]

# Disable alerting engine & UI features

;enabled = true

# Makes it possible to turn off alert rule execution but alerting UI
is visible

;execute_alerts = true

# Default setting for new alert rules. Defaults to categorize error
and timeouts as alerting. (alerting, keep_state)

;error_or_timeout = alerting

# Default setting for how Grafana handles nodata or null values in
alerting. (alerting, no_data, keep_state, ok)

;nodata_or_nullvalues = no_data

# Alert notifications can include images, but rendering many images
at the same time can overload the server

# This limit will protect the server from render overloading and
make sure notifications are sent out quickly

;concurrent_render_limit = 5

# Default setting for alert calculation timeout. Default value is 30

;evaluation_timeout_seconds = 30

# Default setting for alert notification timeout. Default value is
30

;notification_timeout_seconds = 30

# Default setting for max attempts to sending alert notifications.
Default value is 3

;max_attempts = 3

##### Explore
#####
```

```
[explore]
# Enable the Explore section
;enabled = true

##### Internal Grafana Metrics
#####

# Metrics available at HTTP API Url /metrics

[metrics]
# Disable / Enable internal metrics
;enabled = true

# Publish interval
;interval_seconds = 10

# Send internal metrics to Graphite

[metrics.graphite]
# Enable by setting the address setting (ex localhost:2003)
;address =

;prefix = prod.grafana.%(instance_name)s.

##### Distributed tracing
#####

[tracing.jaeger]
# Enable by setting the address sending traces to jaeger (ex
localhost:6831)
;address = localhost:6831

# Tag that will always be included in when creating new spans. ex
(tag1:value1,tag2:value2)
;always_included_tag = tag1:value1

# Type specifies the type of the sampler: const, probabilistic,
rateLimiting, or remote
;sampler_type = const

# jaeger samplerconfig param
# for "const" sampler, 0 or 1 for always false/true respectively
# for "probabilistic" sampler, a probability between 0 and 1
# for "rateLimiting" sampler, the number of spans per second
# for "remote" sampler, param is the same as for "probabilistic"
# and indicates the initial sampling rate before the actual one
# is received from the mothership
;sampler_param = 1
```

```
##### Grafana.com integration
#####

# Url used to import dashboards directly from Grafana.com

[grafana_com]
;url = https://grafana.com

##### External image storage
#####

[external_image_storage]

# Used for uploading images to public servers so they can be
included in slack/email messages.

# you can choose between (s3, webdav, gcs, azure_blob, local)
;provider =

[external_image_storage.s3]
;bucket =
;region =
;path =
;access_key =
;secret_key =

[external_image_storage.webdav]
;url =
;public_url =
;username =
;password =

[external_image_storage.gcs]
;key_file =
;bucket =
;path =

[external_image_storage.azure_blob]
;account_name =
;account_key =
;container_name =

[external_image_storage.local]
# does not require any configuration

[rendering]

# Options to configure external image rendering server like
https://github.com/grafana/grafana-image-renderer
```

```
;server_url =
;callback_url =
[enterprise]
# Path to a valid Grafana Enterprise license.jwt file
;license_path =
[panels]
# If set to true Grafana will allow script tags in text panels. Not
recommended as it enable XSS vulnerabilities.
;disable_sanitise_html = false
[plugins]
;enable_alpha = false
;app_tls_skip_verify_insecure = false
```

Energybase-influx.service

```
[Unit]
Description=Simple energybase send service
[Service]
Type=simple
ExecStart=/usr/bin/python3 /home/pi/test.py
Restart=always
[Install]
WantedBy=multi-user.target
```