

Forschungsbericht BWPLUS

Dezentrales Open-Source Energiemanagement

von

Thomas Brenner, Ole Langniß, Felix Förster

OLI Systems GmbH

Bernd Mauerer, Marcelo Emmerich

Conventic GmbH

Förderkennzeichen: BWSGD 17001 - 17002

Laufzeit: 01.04.2017 – 30.09.2020

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

Dezember 2020

DOSE

Dezentrales **O**pen-**S**ource **E**nergiemanagement

Ein Betriebssystem für die Energiewende

Ergebnisbericht aus dem Förderprojekt „DOSE“

BWSGD 17-001

Conventic GmbH

Burgstraße 69
53177 Bonn

Bernd Maurer
Marcelo Emmerich

OLI Systems GmbH

Silberburgstraße 112
70176 Stuttgart

Thomas Brenner, PhD
Ole Langniss, PhD
Felix Förster, M.Eng.

CONVENTIC



Inhalt

1	Was ist DOSE? Eine Kurzbeschreibung	4
2	Rollen	8
3	Funktionskomponenten	11
3.1	User-Management	11
3.2	Messen, Persistieren, Visualisieren	11
3.3	Energienutzung optimieren	11
3.4	Transparenz und Steuerung für und durch den Verteilnetzbetreiber	12
3.5	Lokale Energiemärkte und Community	13
4	Die DOSE Architektur und Komponenten	14
4.1	OLI Box	14
4.1.1	Hardware	15
4.1.2	Software	19
4.2	DOSE Cloud	23
4.2.1	IT-Architektur	25
5	Einsatz	29
5.1	Allgäu Microgrid	29
5.2	Canary Network zusammen mit der Grid Singularity	34
5.3	Windkraftanlage Edersleben als Teil der DOSE Community	35
5.4	ifeu Heidelberg	37
6	Ausblick und Transfer	40
7	Verzeichnisse	42
7.1	Quellenverzeichnis	42
7.2	Abbildungsverzeichnis	44

1 Was ist DOSE? Eine Kurzbeschreibung

DOSE ist ein dezentral organisiertes und frei verfügbares Energiebetriebssystem für die neuartigen Herausforderungen, die die Energiewende mit sich bringt. Von diesen seien insbesondere folgende genannt:

- Einbindung kleinteiliger und dezentraler Energieanlagen in Märkte
- Notwendigkeit der Verfügbarkeit und Bereitstellung von Flexibilität auf lokaler Ebene
- Anforderungen an Identitätsnachweise, Transparenz sowie Vertrauen
- Existierende proprietäre, „Silo“-artige Kommunikations- und Datenbankstrukturen

Mit den in DOSE entwickelten Komponenten möchten wir diesen Herausforderungen entgegentreten. Wie auch in anderen Bereichen, besteht auch in diesem Projekt das „große Ganze“ aus kleineren Teilen und ist modular aufgebaut. In DOSE spiegeln sich diese sowohl in den Rollen, die die User einnehmen können, als auch in den Hierarchien wider. So ist die kleinste Einheit, die in DOSE technisch mit einer OLI Box repräsentiert wird, beispielsweise eine PV-Anlage, ein Stromzähler oder eine Batterie. OLI Boxen können dann zusammengefasst und einem Teilnehmer zugeordnet werden. Bei dem Teilnehmer kann es sich um einen Haushalt, ein Büro oder eine andere Liegenschaft handeln. Eine Community, also ein Zusammenschluss an Teilnehmern auf organisatorischer, räumlicher und/oder logischer Ebene, bildet schließlich den Rahmen - und die Systemgrenze einer Instanz des DOSE Systems.

Dieser, für die Energiewirtschaft und -technik logische Rahmen, der beispielsweise u.a. in der Form einer „Zelle“ im C/sells Projekt näher untersucht wird, gewinnt an Bedeutung. So hat die Europäische Union eigene, rechtliche Rahmenbedingungen für sogenannte „citizen energy communities“ (CECs) (1) und „renewable energy communities“ (RECs) (2) geschaffen um deren Wachstum zu fördern.

Obwohl es – insbesondere bezüglich des Grades der Kommerzialität – Unterschiede zwischen beiden „Community“ Formen gibt, so sind auch Gemeinsamkeiten erkennbar:

- *Es handelt sich um Entitäten mit juristischer Person.*
- *Sie werden durch die Teilnehmer/Mitglieder kontrolliert.*
- *Das Hauptziel der Community ist die Erreichung bestimmter Umwelt- und Wirtschaftsziele unter Berücksichtigung sozialer Aspekte.*

Aus: European Union Electricity Market Glossary „Energy Community“ (3)

Rescoop.eu, der 2011 gegründete Dachverband für europäische Energiegenossenschaften¹, versteht unter dem Begriff „Community“ im Kontext der Energiewirtschaft folgendes:

„For Rescoop.eu, an energy community is a way to „organise“ citizens that want to cooperate together in an energy-sector related activity based on open and

¹ Engl: renewable energy cooperatives

democratic participation and governance, so that the activity can provide services or other benefits to the members or the local community. In this sense, energy communities represent an alternative type of market actor, and a different way/philosophy to do business, which is now acknowledged by the CEP. The primary purpose of energy communities is to create social innovation – to engage in an economic activity with non-commercial aims.“

- Aus: „Q&A: What are ‚citizen‘ and ‚renewable‘ energy communities?“ (4)

Im gleichen Dokument geht Rescoop.eu auch darauf ein, was konkret die Tätigkeiten von selbst-organisierten CECs und RECs sein können (4):

- Selbstversorgung mit erneuerbaren Energien
- Gemeinsame Erzeugung, gemeinsamer Bezug, gemeinsamer Speicher
- Strom-Sharing
- Aggregation

Dem Leser wird schnell klar, dass zur Umsetzung solcher Vorhaben Technologie notwendig ist, was auch Rescoop.eu feststellt. Aber natürlich ist Technologie nicht das Ziel von Communities, sondern nur das Mittel zum Zweck:

„Smart energy systems innovations can be used as one of the tools to achieve these aims – but they are not the aims in and of themselves.“

Der Markt hat die Idee von Energy Communities als Ganzes auch ohne europäische Initiativen bereits vor Jahren aufgegriffen und nutzt diese für Kundenbindung sowie Cross- und Upselling. Die **sonnen GmbH** bietet ein Komplettsset von Photovoltaik, Batterie, Stromversorgung und „community“ an (5). Grünstromdiscounter wie **eprimo** haben oft vergleichbare Produkte im Portfolio. (6)

Die **Heidelberger Energiegenossenschaft (HEG)** ermöglicht es den Teilnehmern gegen einen Mitgliedsbeitrag das Investieren in lokale Energieanlagen sowie den Bezug von „Bürgerstrom“. (7)

Das sogenannte **Brooklyn Microgrid** in New York ermöglichte erstmals im April 2016 das Handeln von lokal erzeugtem PV-Strom zwischen Haushalten mit der Blockchain-Technologie (8). Die technische Errungenschaft sowie die Art und Weise, wie hier mögliche neue Geschäftsmodelle aus dem Nichts entstanden sind, erzeugte in Industrie sowie Forschung für großes Aufsehen.

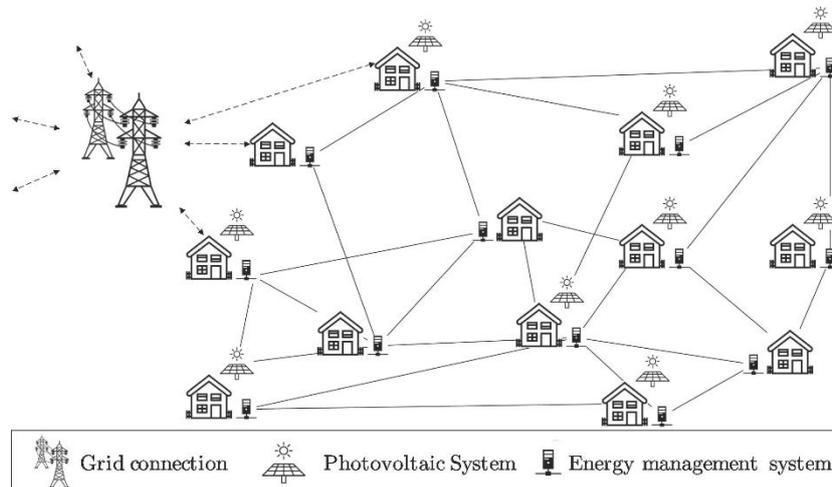


Abbildung 1 Einfache Systemansicht des Brooklyn Microgrid. (8)

Von den gerade eben genannten Beispielen ist die HEG die einzige „Community“, die tatsächlich der Definition der EU entspricht und „bottom-up“ entstanden ist. Gleichzeitig bietet die HEG ihren Kunden beziehungsweise Mitgliedern auch ein wesentlich statischeres Mehrwertpaket an und bleibt vermutlich unter ihren Möglichkeiten.

Ein wesentlicher Faktor hierfür ist, dass diesen Entitäten hohe Investitionskosten für die Informations- und Kommunikationstechnologie im Weg stehen. Um Echtzeit-Visualisierung, Daten- und Stromaustausch sowie Energiemanagement den Teilnehmern anzubieten, sind eine Vielzahl an Messungen, Geräten und Cloud-Servern notwendig.

Mit DOSE soll es dieser Zielgruppe ermöglicht werden, perspektivisch selbst ein IT-System aufzubauen, das nicht nur auf Augenhöhe mit den kommerziellen und proprietären Systemen der Konkurrenz ist, sondern auch genügend Freiheit für die Implementierung neuer Energiemanagement-Konzepte lässt und die Dynamik in der technischen Entwicklung respektiert.

An dieser Stelle soll auch noch auf das Thema Open Source eingegangen werden. Viele Energiemanagementsysteme und deren Ersteller sind mit proprietären Ansätzen gestartet und versuchten vor Allem den Verkauf von zentralen Hardwarebausteine inkl. Software als Geschäftsmodell zu etablieren. Dies scheitert jedoch aus verschiedenen Gründen. Zum Einen ist die Energiewirtschaft im Allgemeinen auf Interoperabilität angewiesen. Im Netz sind natürlich heute schon rein elektrisch die verschiedensten Typen von Erzeugern und Verbrauchern angeschlossen. Unterschiede in Spannung, Frequenz und anderen elektrotechnischen Größen wurden schon vor langer Zeit beseitigt, um die allgemeine Effizienz zu heben und keine lock-in Effekte und Abhängigkeiten (durch die Auswahl der langfristig unterlegenen Standards) zu erzeugen.

Heutzutage tun sich ähnliche Probleme auf der darüberliegenden Informations(austausch)-Ebene auf. Je nachdem wie Geräte kommunizieren, gibt es große Unterschiede bei den Datenformaten, -austauschprozessen und auch bei der -verarbeitung. Viele der verfügbaren Standards liegen hinter Paywalls und/oder Lizenzgebühren und erzeugen leicht Friktion bei den Nutzern. Nicht nur sind damit Gewerbekunden gemeint, die diese Standards in ihren Geräten implementieren, sondern

auch preissensitive Haushaltskunden, die oftmals vor einem Kauf zurückschrecken aus Angst, etwas nicht zukunftsfähiges über Jahre hinweg einzubauen.

Derzeit wird der Energiewirtschaft jedoch langsam bewusst, dass offene und individuell leicht anpassbare Standards der Weg nach vorn sind. Das **openEMS** ist sehr ähnlich aufgestellt wie DOSE und konnte einige Erfolge in den letzten Jahren verbuchen. Die Energy Web Foundation hat im Juni 2019 eine komplett offene Blockchain für den Energiesektor gestartet und verzeichnet mittlerweile weit über 100 Mitglieder, darunter Corporates als auch Start-Ups. DOSE folgt ebenfalls dem Open Source Modell. Deswegen ist der DOSE Code auf GitHub verfügbar.

Die folgende Dokumentation führt schrittweise durch das Thema DOSE. Zuerst wird die besondere Rollen- und Hierarchiestruktur von DOSE besprochen. Danach wird die Architektur hinter dem DOSE-System und die einzelnen Funktionskomponenten beleuchtet. Den Abschluss bildet ein Portfolio an Projekten und Einsatzzwecken, in denen die DOSE Komponenten erfolgreich eingesetzt wurden.

2 Rollen

Für DOSE wurden eigene Rollen- und Hierarchiesysteme angelegt. Klassische Admin-Nutzer oder Master-Slave Beziehungen sind für ein dezentrales Betriebssystem für ein dezentrales Energiesystem nur bedingt geeignet, da sie implizit Privilegien und Verantwortungen vorgeben.

Um hier ein Beispiel zu nennen: der klassische Administrator in einem jeden IT-Netzwerk ist für den ordnungsgemäßen Betrieb verantwortlich und erhält hierfür weitreichende Privilegien. Dazu gehört zum Beispiel das Anlegen und Bearbeiten von Nutzerdaten und -konten, sowie White- und Blacklisting. Übertragen auf ein IT-Netzwerk, das in Tandem mit dem „Energienetzwerk“² funktionieren soll, bedeutet dies, dass der Admin umfassende Zugriffsrechte auf Daten (Last- und Erzeugungsgänge, Verträge, Konfigurationen) aus der Infrastruktur sowie Steuerungsgewalt auf die Art und Weise, wie Energie verbraucht, verteilt und produziert wird.

Das „bottom-up“ Paradigma, das mit dem Ausbau der dezentralen Energieversorgung sowie Änderungen in der Policy immer mehr Einfluss erfährt, könnte daher durch eine konventionelle Digitalisierung sehr schnell untergraben werden. Der Admin, je nachdem welcher bestehende bzw. neue Marktteilnehmer diese Rolle übernimmt, erhält so schnell große Macht – und dies oftmals im unregulierten Bereich.

Die Herausforderung besteht nun darin, diese Privilegien als auch die Verantwortung durch eine geschickte Definition der Rollen und Hierarchien so einzuflechten, dass das dezentrale Paradigma zum Vorschein kommt – und auch bestehen bleibt.

Für DOSE wurde, wie bereits eingangs erwähnt, eine neue Rollenstruktur erstellt. Diese ist in Abbildung 2 dargestellt.

² „Energienetzwerk“ im Sinne der Gesamtheit der Erzeuger, Verbraucher, Zähler, Stromnetze, der Teilnehmerschaft sowie dahinterstehenden Wertschöpfungsketten.

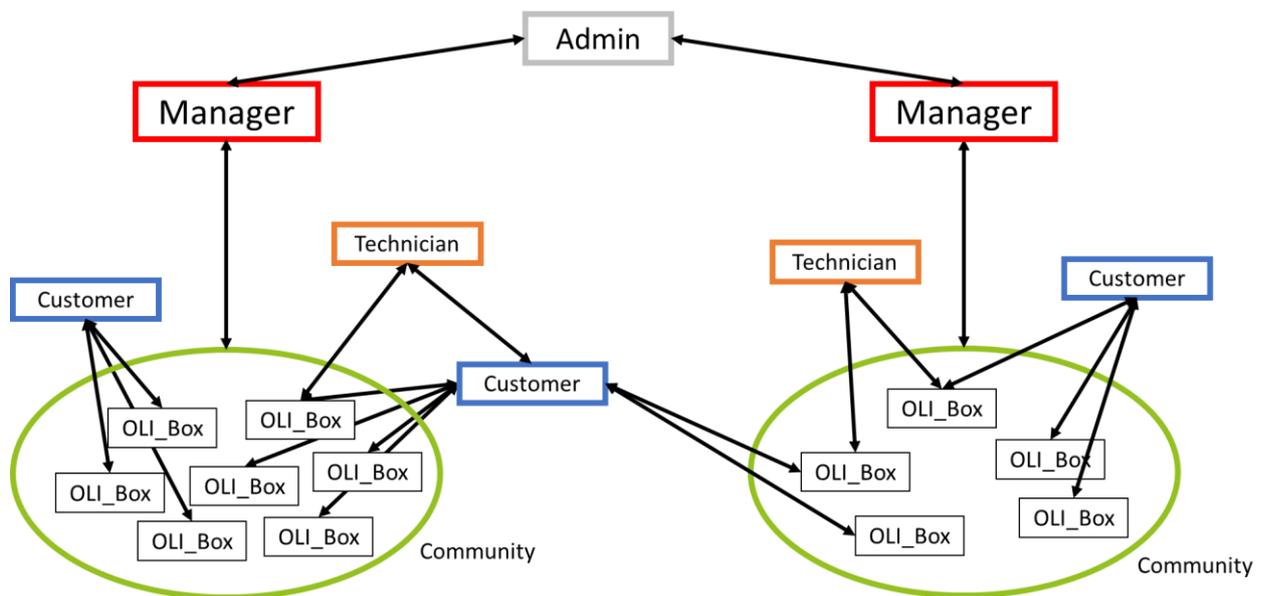


Abbildung 2 DOSE Rollenstruktur und Hierarchien

Es folgt eine kurze Beschreibung der einzelnen Rollen, Elemente und Gruppen sowie der Beziehungen untereinander.

Die **OLI Box** ist der Grundbaustein des DOSE Systems. Die OLI Box ist eine Hardware-Einheit und repräsentiert ein Energy-Device – das kann ein Zähler, ein Wechselrichter, eine steuerbare Last oder eine Batterie sein.

Der **Customer** ist die natürliche oder juristische Person, die eine Wohnung mietet, ein Einfamilien-/Mehrfamilienhaus bewohnt oder ein Gewerbe betreibt. Ein Customer verfügt über eine oder mehrere OLI Boxen.

Eine **Community** ist eine Gruppe aus **OLI Boxen**. Sie repräsentiert idealerweise eine räumlich, technisch-ökonomisch, organisatorisch oder anderweitig logisch abgrenzbare Gemeinschaft aus OLI Boxen. Beispiele hierfür sind lokale Energiecommunities in einer Ortschaft, einem Stadtbezirk, einem Quartier oder auch ein Strang in einem Verteilnetz. Auch organisatorische Einteilungen wie die Gruppierung aller OLI Boxen in den Filialen einer Supermarktkette oder den Produktionsstätten eines Autoherstellers sind denkbar. Die technisch-ökonomische Einteilung ist ebenfalls möglich: so könnten Bilanzkreise in DOSE gebildet werden oder Verbraucher und Erzeuger in Post-EEG oder EEG-befreite Gemeinschaften gruppiert werden.

Der **Manager** nimmt die „Vorsteher“ Rolle einer **Community** DOSE System ein. Hierbei kann es sich um einen teilnehmenden **Customer**, einen Dienstleister oder eine anderweitig befugte natürliche oder juristische Person handeln. Der Manager steht der Community vor und kann die ihm von den Customer individuell zugestandenen Befugnisse nutzen, um beispielsweise den Bilanzkreis zu bewirtschaften. Übertragen auf das Bilanzkreismodell, ist der Manager der *Bilanzkreisverantwortliche*.

Die Rolle des **Technikers** ergibt sich aus der Notwendigkeit, die vielen dezentral organisierten OLI Box Einheiten zu installieren, zu modifizieren und zu verwalten. Für diese kleinteilige Arbeit mit großem Volumen, benötigt man eine dedizierte Rolle die zwar die OLI Boxen der Customer (vor Ort) bearbeiten kann, jedoch nur limitierte Einsicht in die Datenströme erhält. Der Techniker ist damit Unterstützer des **Managers**, der technische Arbeiten an den Techniker delegieren kann.

Der **Admin** ist wie eingangs beschrieben eine Rolle aus klassischen Hierarchien in IT-Systemen. Auch für DOSE bildete sich ursprünglich der Admin heraus – schließlich muss jemand auch den **Managern** vorstehen und diese erstellen und verwalten und – im Zweifel – in den Betrieb als „Polizei“ eingreifen. Durch geschickten Einsatz des open-source Paradigmas in DOSE sowie der Art und Weise, wie DOSE keine einzige Cloud darstellt, sondern viele klein- bis mittelgroße Inseln in der Energiewirtschaft, kann der Admin entfernt werden.

3 Funktionskomponenten

Bevor DOSE als Technologie und Architektur in Kapitel 4 näher erläutert wird, soll in diesem Kapitel die einzelnen Funktionskomponenten beschrieben werden. Diese Bausteine sind teilweise voneinander abhängig und teilweise optional. Je nach Nutzung von DOSE können – solange die Komponente nicht obligatorisch ist – diese einzeln zugeschaltet werden.

3.1 User-Management

Das User-Management nimmt in nahezu jeder Software eine wichtige Position ein. Generell muss diese Funktionskomponente die folgenden Features abdecken:

- Erstellen, Bearbeiten und Löschen von Usern³
- Speichern und Bearbeiten verschiedener Metainformationen (Namen, Adressen, E-Mail Adressen)
- Rücksetzung der Passwörter via E-Mail durch die User
- Verknüpfung der User bzw. Tenant Accounts mit den dazugehörigen Rollen (siehe Kapitel 2)

Anhand der obigen Punkte ist zu erkennen, dass mit dem Betrieb dieser Komponente große Verantwortung sowie Privilegien verbunden sind. Die Datenübertragung, sowie -sicherung muss den Best Practices entsprechen.

3.2 Messen, Persistieren, Visualisieren

Die zielführende Umsetzung eines Energiemanagements kann nur auf Basis von Daten zur Erzeugung, Verbrauch und Steuerung elektrischer, thermischer und chemisch gebundener Energie erfolgen. Wie jedes andere Energiemanagementsystem benötigt DOSE daher auch eine Möglichkeit die Datenströme zu erfassen, persistieren und visualisieren.

Die beiden Hauptkomponenten, die diese Funktion ausüben sind die OLI Boxen zur Erfassung und „Versendung“ der Informationen in Richtung der DOSE-Cloud, die die Daten verarbeitet, abspeichert und dann schließlich für den Endnutzer über einen Browser visualisiert.

Nähere Informationen zu dieser Funktionskomponente und der OLI Box sowie der DOSE-Cloud sind in den Kapiteln 4.1 und 4.2 zu finden.

3.3 Energienutzung optimieren

In der DOSE-Cloud sind verschiedene Funktionen ablegbar. Dazu zählt auch die Optimierung einzelner Haushalte und Liegenschaften auf Basis von Nutzungsinformationen, Wetterdaten und daraus errechneten Last- und Erzeugungsprognosen. Während die DOSE-Cloud die Informationen

³ Der Begriff „Tenant“ ist ebenfalls üblich, besonders im User-Management.

aus verschiedenen Quellen sammelt und Handlungsempfehlungen ausspricht, findet die Umsetzung der Steuerung an den Anlagen über die OLI Boxen statt.

Eine Energieoptimierung kann grundsätzlich anhand vieler verschiedener Zielparameter erfolgen:

- ... nach größtmöglicher Eigennutzung des selbst erzeugten Stroms beispielsweise einer Photovoltaik-Anlage
- ... nach möglichst großer Nutzung von lokalen Energieanlagen
- ... nach möglichst großer Vermeidung von CO₂-Emissionen
- ... nach möglichst großer Kostenvermeidung

Im Rahmen des DOSE Projekts wurde ein Optimierer entwickelt, der nach der größtmöglichen Eigennutzung von selbst erzeugtem Photovoltaik-Strom für den Ladevorgang des Elektroautos des Eigentümers strebt. Die Optimierung kann vom Customer zu- und abgeschaltet werden. Außerdem steht es dem Manager frei, weitere Optimierer für die Customer in der Cloud anzubieten.

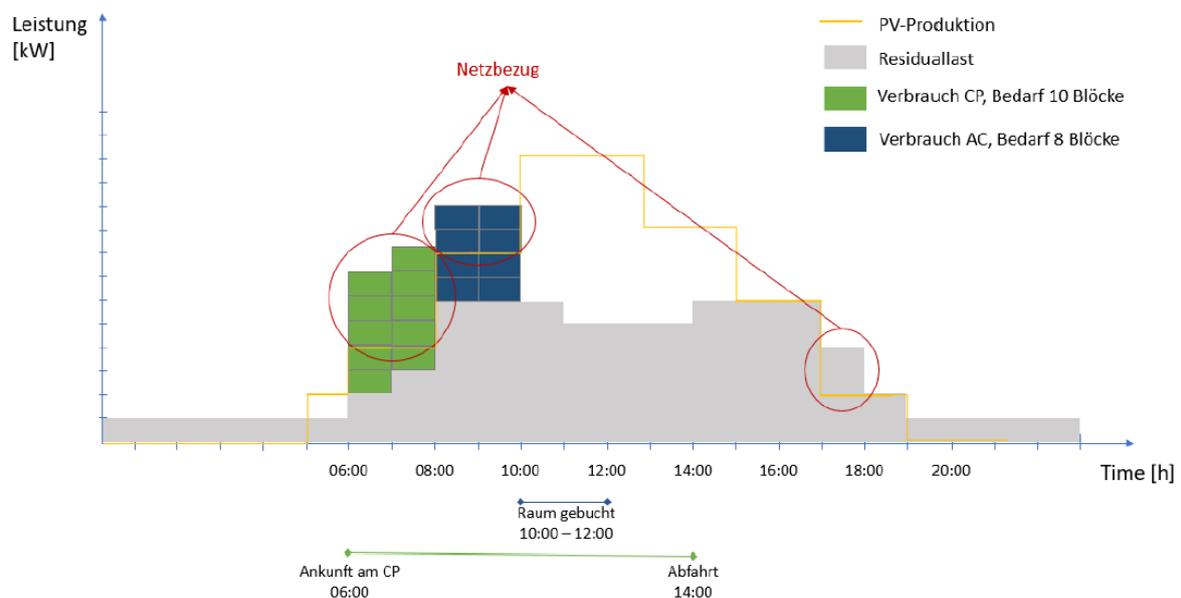


Abbildung 3 Beispiel Schemata des DOSE-Optimierers.

3.4 Transparenz und Steuerung für und durch den Verteilnetzbetreiber

Neben der Optimierung einzelner Haushalte und Liegenschaften ist die Kommunikation mit dem Verteilnetzbetreiber für ein modernes Energiemanagementsystem ebenfalls einzuplanen. In DOSE wurde ein Blockchain-basierendes Tool entwickelt, das die analogen, unidirektionalen und intransparenten Signale über Rundsteuerempfänger digitalisiert. Sowohl VNB als auch Customer erhalten dadurch genauere Einblicke über die Existenz und Wirksamkeit eines Steuerungsvorgangs, der beispielsweise im Rahmen des Einspeisemanagements durchgeführt werden könnte.

3.5 Lokale Energiemärkte und Community

Eine weitere Funktionskomponente stellen die lokalen Energiemärkte dar inklusive der Abstufung zur Community.

Im Rahmen von DOSE wurden die verschiedensten Modelle – sowohl aus organisatorischer wie auch technischer Sicht – erprobt. Neben der Umsetzung im Allgäu Microgrid (Kapitel 5.1) wurde auch eine Verbindung mit einem Drittsystem zur Umsetzung dieser Variante erprobt (Kapitel 5.2).

Prinzipiell geht es darum, dass es Prosumern und Consumern auch auf den niederen Netzebenen ermöglicht werden sollte, am Energiemarkt mit Ihren eigenen Energieanlagen teilzunehmen. In Kapitel 1 ist bereits auf dieses Thema eingegangen worden. Es kann sowohl ein genossenschaftlicher als auch ein Marktansatz gewählt werden. Neben der Erstellung einer Marktplattform muss auch der Biet- und Abrechnungsprozess durchdacht, sowie die Limitationen und Möglichkeiten der Hardwareeinbindung und der Nutzererfahrung einbezogen werden. In Kapitel 5 finden sich hier einige Beispiele.

4 Die DOSE Architektur und Komponenten

In diesem Kapitel werden die einzelnen Hardware- und Softwarekomponenten sowie die Architektur genauer beleuchtet. Dabei teilen wir die Komponenten hauptsächlich in die OLI Box sowie die DOSE-Cloud auf.

4.1 OLI Box

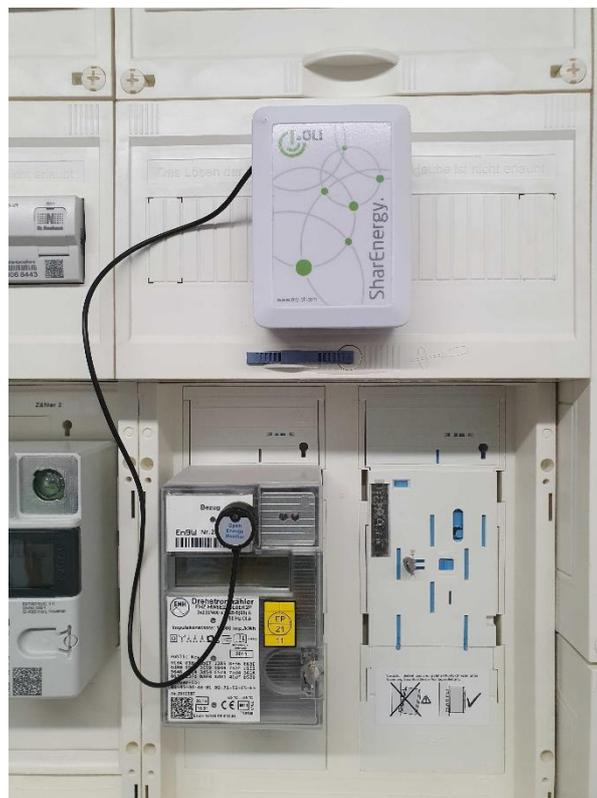


Abbildung 4 Beispielhafte Installation einer OLI Box an einer modernen Messeinrichtung

Wie bei den Rollen in Kapitel 2 beschrieben ist die OLI Box der Grundbaustein des DOSE Systems. Die OLI Box ist gleichzeitig eine Rolle, Software sowie Hardware.

Die Hauptaufgabe der OLI Box ist das Repräsentieren eines Energy-Device⁴. Die OLI Box wird physisch mit dem Device verbunden, liest es in bestimmten Intervallen über verschiedene Schnittstellen aus, digitalisiert das Signal und sendet es in einem vordefinierten Signal an die DOSE-Cloud der Community. Doch auch bidirektional – also Signale und Mitteilungen von der DOSE-Cloud zurück an die OLI Box – kann diese verarbeiten und an das eigentliche Gerät übermitteln.

Im tatsächlichen Einsatz muss die OLI Box folgende Anforderungen erfüllen:

⁴ Wie in Kapitel 2 beschrieben handelt es sich bei Energy-Devices um einen bewusst weiten Begriff, der unter anderem Wechselrichter, Zähler, Klimaanlage, Batterien und steuerbare Lasten umfasst.

- Genug Prozessorleistung und Arbeitsspeicher für das Erfassen, Verarbeiten und Versenden von Energiedaten
- Modulare Kommunikation mit einer Vielzahl von Energiegeräten und Sensoren über eine ebenso große Zahl an Kommunikationsschnittstellen und -protokollen
- Die Möglichkeit mit lokalen und externen (Internet-) Netzwerken zu kommunizieren
- Speicherplatz für lokale Energiedaten (Zwischenspeicherung)
- Freie (Open-Source) Verfügbarkeit der Komponenten

In den folgenden Kapiteln wird erklärt, wie das Projekt DOSE die Anforderungen analysiert und erfüllt hat.

4.1.1 Hardware

Aus den Anforderungen wird ersichtlich, dass nur eine Kombination aus modularer Hardware sowie möglichst flexibler Software die Anforderungen erfüllen kann. Im Laufe des Projekts wurden daher eine Reihe von Einplatinencomputer, Sensoren sowie Softwarelösungen analysiert. Es wurde dabei stets auf die Kompatibilität mit dem OpenSource Gedanken geachtet.

4.1.1.1 Einplatinencomputer

Schnell hat sich der Raspberry Pi 3B der Raspberry Pi Foundation als geeignetste Lösung für die Umsetzung innerhalb der Projektlaufzeit herauskristallisiert. Gründe hierfür sind neben den sehr guten technischen Eigenschaften, die in Tabelle 1 aufgelistet sind, auch die gute Verfügbarkeit an eigens angepassten Betriebssystemen. Für das Projekt wurde Raspbian ausgewählt, was das mit Abstand am meisten genutzte Betriebssystem für den Raspberry Pi 3 B ist und mit einer Vielzahl an frei verfügbaren, open-source Softwarepaketen erweiterbar ist.

Quick Facts	Raspberry Pi 3 Modell B
Entwickler / Stiftung	Raspberry Pi Foundation
Hersteller	Element 14
Prozessor	1,2 GHz 64-bit Quad-Core ARMv8 CPU
Arbeitsspeicher (RAM)	1 GB
WLAN Standard	802.11n
Konnektivität	MicroSD – Kartenleser 4 USB Ports 1 HDMI Port Ethernet Port 3.5mm Audio und Composite-Video Port Kamera (CSI) und Display (DSI) Anschluss Bluetooth 4.1 Bluetooth Low Energy (BLE) GPIO 40 Pins
Leistungsbedarf	Bis zu 5 Watt
Preis (mit MwSt.)	ca. 35 €

Tabelle 1 Raspberry Pi 3B Datenblatt

Der Raspberry Pi hat in der Vergangenheit bereits einige Iterationen durchlaufen und während der Projektlaufdauer sind dementsprechend weitere Modelle (9) (10) veröffentlicht worden. Diese verfügen dann meist über mehr Prozessorleistung, Arbeitsspeicher und Schnittstellen. Einige der Weiterentwicklungen wurden auch im Projekt eingesetzt (siehe auch Beschreibung des prototypischen Einsatzes in Kapitel 5).

4.1.1.2 Sensoren

Neben dem Raspberry Pi 3B wurde auch weitere Hardware verbaut. Beispielhaft soll hier der Infrarot-Sensor genannt werden, der zur Erfassung von Daten direkt aus sogenannten modernen Messeinrichtungen genutzt wird (siehe Abbildung 4 und Abbildung 5). Der Sensor ist über das ähnliche Projekt emonPI verfügbar (11). Moderne Messeinrichtungen verfügen – ähnlich wie die Drehscheiben in den klassischen Ferraris-Zählern, über eine optisch, vom Menschen interpretierbare Schnittstelle. Die LED in dieser Schnittstelle „blinkt“ schneller, desto mehr Leistung über den Zähler geleitet wird. Bei einem Haushaltszähler ist dies der Haushaltsverbrauch, bei einem Photovoltaikzähler die PV-Einspeisung. Der Sensor wurde mit einem Adapter versehen und ist dadurch mit der OLI Box kompatibel.



Abbildung 5 OLI Box kommuniziert mit einer modernen Messeinrichtung

Bei den Testinstallationen hat sich schnell gezeigt, dass die Infrarot-Sensor Methode zur leichten Beschaffung von Energiedaten von Haushalten zu bevorzugen ist. Der Sensor ist kostengünstig und lässt sich leicht anbringen. Da die Energiedaten direkt aus dem geeichten Zähler kommen, sind diese außerdem präzise. Jedoch ist sie nur anwendbar, wenn eine moderne Messeinrichtung (sprich: kein Ferraris-Zähler) installiert ist. Sollte in dem Haushalt keine mMe oder vergleichbare Geräte vorhanden sein oder die Messsituation⁵ anderweitig keinen Einsatz zulassen, so muss oftmals mit einer eigenen physikalischen Messung fortgefahren werden.



Abbildung 6 Der SmartPi mit Messwandler (12)

Innerhalb von DOSE wurde testweise der SmartPi der Firma enerserve erfolgreich eingesetzt (12). Dieser ermittelt über Messwandler die physikalischen Ströme an den Stromleitungen. Zusätzlich kann auch eine Spannungsmessung durchgeführt werden, deren Installation jedoch in jedem Fall einen Elektriker voraussetzt. Jedoch ist auch ohne Spannungsmessung die Installation für den Customer bzw. den Techniker (siehe Beschreibung Kapitel 2) zeitintensiv und mit Unsicherheiten behaftet. So kann ohne weitgehende Öffnung der Zählerschränke und der dahinterliegenden Verkabelung nicht immer eindeutig festgestellt werden, welche Stromleitungen für die Messung relevant sind.

4.1.1.3 Networking

Die Kommunikation der OLI Boxen mit der DOSE-Cloud der Community findet durch verschlüsselte Verbindungen über das Internet statt. Die OLI Box benötigt daher einen Internetzugang, der durch

⁵ Beispiele sind hier „organisch“ gewachsene Elektroinstallationen, die oftmals nur über einen Zähler verfügen. Dieser misst oftmals sowohl Erzeuger als auch Verbraucher unterschiedlicher Eigentümer, wodurch die direkte Bestimmung der positiven und negativen Energieflüsse schwierig und zeitintensiv wird.

das WLAN des Customers oder alternativ durch einen eigenen 3G/4G Router hergestellt werden kann. Vor- und Nachteile der Varianten sind in Tabelle 2 dargestellt.

	Customer WLAN	3G/4G Router
Vorteile	<ul style="list-style-type: none"> • Nahezu jeder Haushalt und die meisten Gewerbe verfügen über ein eigenes WLAN • Die Nutzung des WLANs durch die OLI Box im Haushalt ist mit keinen weiteren Kosten verbunden • Der Zugang zum WLAN kann bei Empfang mit einem einfachen Passwort durchgeführt werden 	<ul style="list-style-type: none"> • Der Router stellt eine eigene Verbindung zum Internet über das Mobilfunknetz her und ist damit komplett unabhängig von lokalen Netzwerken • Der Mobilfunkrouter kann eigens für die OLI Box konfiguriert werden.
Nachteile	<ul style="list-style-type: none"> • Die OLI Boxen müssen aufgrund der Sensorik in der Nähe der Zähler und Energy-Devices installiert werden. Diese befinden sich oft in wenig genutzten Räumen und Schränken (Beispiel: Keller). Das WLAN kann hier in manchen Gebäuden nur schlecht vordringen • Manche WLAN-Netzwerke und deren Router sind mit Zeitschaltuhren versehen, wodurch das WLAN zu bestimmten Zeiten ausgeschaltet wird und die OLI Box damit die Verbindung zur OLI Cloud verliert. Darüber hinaus verfügen manche WLAN-Router über verschiedene Sicherheitsmechanismen, die die Verbindung zusätzlich beeinträchtigen können. 	<ul style="list-style-type: none"> • Ähnlich zum Nachteil des „Customer WLAN“ gilt: das Mobilfunknetz ist oftmals in den Räumen und Schränken bei den Energy-Devices schwächer • Gerade in Deutschland sind Mobile Datentarife teuer. Eine Analyse der Tarife der Deutschen Telekom, Vodafone und O2/Eplus ergab grob einen Preis von 10€ für 3GB bei reinen Daten-SIM Karten • Auch die Investition in einen Mobilfunkrouter erzeugt Kosten. Einfache Modelle gibt es ab 50€, allerdings ohne Antennen für den stabilen Empfang. Professionellere Geräte kosten in der Regel über 150€ • Der Router muss eigens konfiguriert und instandgehalten werden, was Aufwände erzeugt

Tabelle 2 Vergleich der Anbindungsmöglichkeiten einer OLI Box an das Internet

Im Projektverlauf wurde klar, dass dem Networking eine große Rolle zuteil wird. Viele der Probleme lassen sich direkt auf Schwierigkeiten und Hindernisse im Networking zurückführen. Eine weitere Variante, die dem Customer WLAN sehr ähnlich ist, ist die Nutzung eines Powerline Adapters. Hier wird über das Haus- bzw. Gewerbestromnetz ein IP-Netzwerk aufgebaut.

4.1.2 Software

Nachdem in Kapitel 4.1.1 bereits die OLI Box als Hardwareeinheit beschrieben wurde, soll in diesem Kapitel nun auf die Software selbst eingegangen werden. In diesem Kapitel werden die Snaps und Fleet Management Tools beschrieben, die für einen ordnungsgemäßen Betrieb unerlässlich sind.

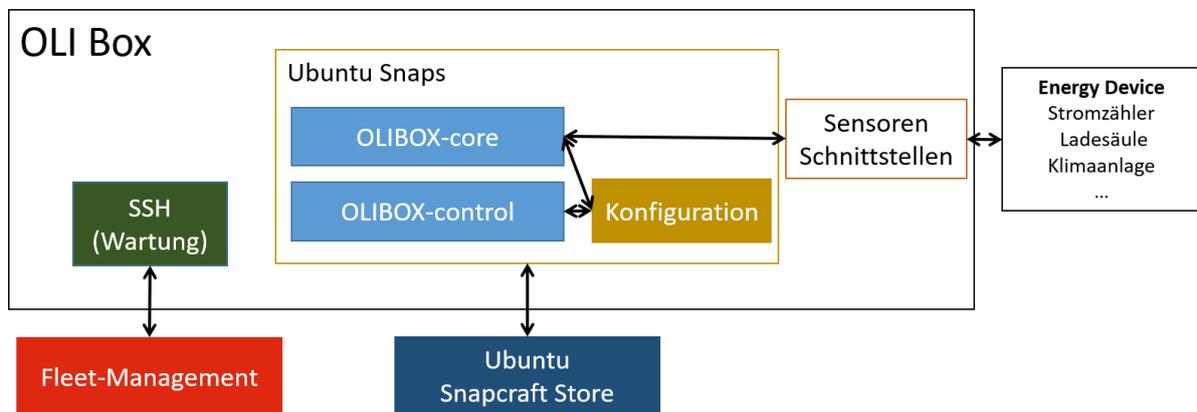


Abbildung 7 OLI Box Architektur

Generell bestehen folgende Anforderungen an die Software:

- **Universal und modular:** Die Software muss möglichst für eine Vielzahl an Anwendungsbereichen und -arten geeignet sein. Je nach Einsatzort kann Sie eigens konfiguriert werden.
- **Erweiterbar:** Die Software muss durch freie Verfügbarkeit modifizierbar sein
- **Einfaches Deployment und einfache Instandhaltung:** Die Software muss einfach zu beziehen und zu installieren sein. Updates werden automatisch bezogen und installiert.
- **Möglichkeit der (Fern)wartung:** Besonders während der Entwicklung muss ein Fernwartungskanal auf die Hardware vorhanden sein, um eventuelle Probleme zu beheben.

Zudem soll die Software konkret folgende Funktionen abbilden:

- Konfiguration der OLI Box
- Aufbau einer Verbindung zur DOSE-Cloud
- Verarbeitung der Sensor-Signale
- Transport der Messwerte zur DOSE-Cloud in einem vorgefertigten Format
- Verschlüsselung der Informationen bei der Übertragung
- Umsetzung der Steuersignale aus der DOSE Cloud zum Energy Device

Die Recherche ergab eine Liste an möglichen Plattformen und Technologien, die die Anforderungen erfüllt hätten. Darunter waren **Docker** (13) und **FIWARE** (14). Für die „Verpackung“ und Verteilung der Software auf den OLI Boxen wurde jedoch **Ubuntu Snaps** (15) als darunter liegende Technologie gewählt. Gründe hierfür sind unter anderem die bessere Eignung für IoT Umgebungen wie die OLI

Box sowie Features wie automatische Updates und die leichte Veröffentlichung und das Deployment des Codes über den Snapcraft Store.

Das Fleet-Management, also die Möglichkeit den Status der OLI Box aus der Ferne einzusehen sowie eine SSH-Verbindung zu der Box aufzubauen, wurde aus dem eigentlichen Code ausgelagert. Es ist sehr wahrscheinlich, dass je nach Kontext des Einsatzes von DOSE hier eigene Lösungen genutzt werden.

Kleinere Installationen in Gewerbeeinrichtungen benötigen oft kein Fleet-Management, da ein Systemadministrator ohnehin Zugriff auf die Geräte über das lokal verfügbare Netzwerk besitzt. Für eher kommerzielle Projekte auf größerer Ebenen mit vielen Teilnehmern können auch proprietäre Systeme in Frage kommen. Beispiele hierfür sind **ZeroTier** (16), **balena** (17), und **upswift** (18). Innerhalb des Projektrahmens fiel die Entscheidung auf **remote.it** (19), da es speziell in einer agilen und dynamischen Entwicklungsumgebung durch die leichte Konfiguration punktet.

In den folgenden Unterkapiteln soll noch einmal näher auf die Bedeutung der Softwarekomponenten eingegangen werden.

4.1.2.1 OLIBOX-core

Der OLIBOX-core Snap ermöglicht die Kern- bzw. Basisfunktionalitäten der OLI Boxen:

- Konfiguration der OLI Box
- Aufbau einer Verbindung zur DOSE-Cloud
- Verarbeitung der Sensor-Signale
- Transport der Messwerte zur DOSE-Cloud in einem vorgefertigten Format
- Verschlüsselung der Informationen bei der Übertragung

Umgesetzt sind die Funktionalitäten in Python 3.8 (20). Neben Python wären auch andere Programmiersprachen wie C++ und Java denkbar gewesen. Diese bieten ähnliche Funktionalität und Flexibilität wie Python, jedoch hat die sehr gute Unterstützung von Raspbian⁶ (21) und dem gesamten Raspberry Pi Ökosystem schließlich zu dieser Entscheidung geführt. Besonders hervorzuheben ist hier die gute Unterstützung für die GPIO Sensoren auf dem Raspberry Pi (siehe Kapitel 4.1.1.1) mit einer eigenen Python-Bibliothek bzw. Package (22). Damit ist Python die erste Wahl für die verschiedenen Applikationen, die direkt auf dem Raspberry Pi ausgeführt werden.

Der Code von OLIBOX-core selbst ist in verschiedene Module aufgeteilt. Die wichtigsten sind in Tabelle 3 aufgelistet und beschrieben.

⁶ Bei Raspbian handelt es sich um das am weitest verbreitete Betriebssystem für den Raspberry Pi und wurde deshalb im Rahmen des Projektes für die OLI Boxen genutzt. Das Betriebssystem basiert auf der Linux Distribution Debian.

Tabelle 3 Auswahl der wichtigsten Module in OLIBOX-core

Name des Moduls	Funktion
Olibox_core.py	Aufrufen der in der Konfiguration festgelegten Programmbestandteile; Verbindung zum MQTT Broker
Config.py	Abfrage der Konfiguration über ein Command-Line-Interface beziehungsweise eine vorkonfigurierte TOML Datei ⁷ (23). Darunter sind Fragen wie „Welche ID hat diese OLI Box?“, „Zu welchem Projekt gehöre ich?“, „Mit was für einem Energy Device bin ich verbunden?“
Oli_mqtt.py	Umsetzung der Verbindung zum Broker sowie dem Versenden der verarbeiteten Daten des Energy Device.
Oli_ir.py	Die Messung mithilfe des in Kapitel 4.1.1.2 vorgestellten Infrarot Lesekopfs wird hier programmatisch umgesetzt.

Der Code für OLIBOX-core ist in einem öffentlich zugänglichen Github Repository hinterlegt.⁸ (24) Wie in Kapitel 4.1.2 beschrieben, wurde der Code in Ubuntu Snaps überführt, und ist somit leicht über den Snapcraft Store beziehbar.⁹ (25) Snapcraft übernimmt dabei auch die Updates: sollte eine neue Version des OLIBOX-core Snaps verfügbar sein, so werden diese automatisiert heruntergeladen.

4.1.2.2 Fleet Management

In Kapitel 4.1.1.3 und 4.1.2 wurde bereits auf die Relevanz des Networkings, insbesondere auf das WAN¹⁰ und die Verbindung der OLI Boxen zur DOSE-Cloud eingegangen. Doch auch die Verbindung zu den OLI Boxen für Wartungszwecke ist eine wichtige Funktion, die besonders im prototypischen Einsatz notwendig ist.

⁷ TOML ist ein menschenlesbares Datenformat, das einfach per Hand oder maschinell erstellt und bearbeitet werden kann. TOML Dateien werden meist für Konfigurationsdateien genutzt. (27)

⁸ https://github.com/olisystems/olibox_core

⁹ <https://snapcraft.io/install/olibox-core/ubuntu>

¹⁰ Wide Area Network, praktisch gleichbedeutend mit „Internet“ im DOSE Kontext.

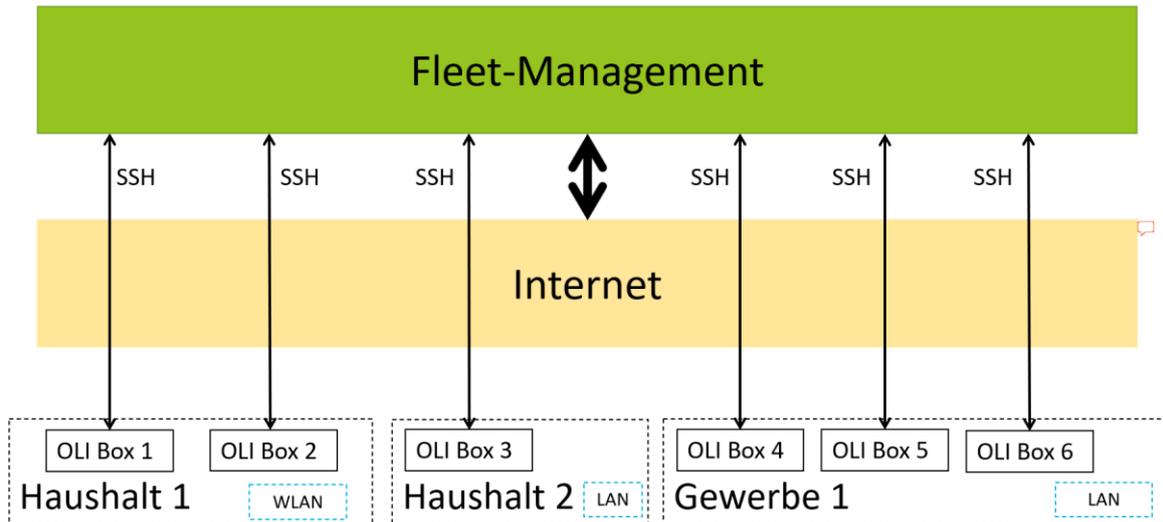


Abbildung 8 Fleet-Management von verteilten Geräten über das Internet

Das Fleet-Management Tool agiert technisch gesehen als Reverse-Proxy, zu dem die OLI Boxen selbstständig „bottom-up“ Verbindung aufnehmen. Dadurch werden Verbindungsprobleme umgangen, die normalerweise entstehen würden, wenn der Prozess „top-down“ durchgeführt wird:

- **Dynamische IPs** sind insbesondere bei privaten Internetzugängen Standard. Dadurch ändert sich auch die IP-Adresse der OLI Boxen in unregelmäßigen Abständen, was diese im schlimmsten Fall nach einer Weile im Internet unauffindbar macht.
- **Vorkonfigurierte Firewalls** in Routern oder eigenen Hardware-Firewalls blockieren verschiedene Ports, die die Verbindung erheblich erschweren und oftmals nur mit zeitintensivem Port-Forwarding behoben werden können.

In DOSE wurde, wie bereits in Kapitel 4.1.2 benannt, das Tool **remot3.it** genutzt (19). Es ermöglicht sowohl die Einsicht in den Status der OLI Box (u.a. online/offline, lokale IP ...) als auch die Verbindung zur OLI Box via SSH und einen private key. In Abbildung 9 ist ein Screenshot des remot3.it Dashboards abgebildet.

<input type="checkbox"/>	Status	Device Name	Owner	Shared	HWID	Internal IP	External IP
<input type="checkbox"/>	CONNECT	OLI_25	me	No	80:00:00:00:01:0		
<input type="checkbox"/>	CONNECT	OLI_31	me	No	b8:27:eb:27:bb:6		
<input type="checkbox"/>	CONNECT	OLI_32	me	No	b8:27:eb:c0:a5:5		
<input type="checkbox"/>	CONNECT	OLI_33	me	No	b8:27:eb:8f:a8:7		
<input type="checkbox"/>	CONNECT	OLI_42	me	No	b8:27:eb:9b:d1:c		

Vertrauliche Information

Abbildung 9 remot3.it Dashboard mit einer Auswahl an OLI Boxen

4.2 DOSE Cloud

Ein weiterer Grundbaustein des DOSE-Systems sind die verschiedenen Komponenten auf der Open Telekom Cloud (OTC), welche dem Paradigma des Cloud Computing zuzuordnen ist. Zu den bekanntesten Cloudanbietern weltweit zählen Amazon – AWS, Google – GCP oder Microsoft – Azure. Als deutscher Platzhirsch gilt allerdings die in DOSE eingesetzte Lösung der Deutsche Telekom mit der OTC und deren Umsetzung der europäischen Datenschutzrichtlinien (DSGVO). (26).

Nach Baun, Kunze, Nimis, Tai verfolgt das Cloud Computing den Ideen des Utility Computing. Das heißt, es werden nur Ressourcen bezahlt, die genutzt werden, was die wirtschaftliche Bedeutung unterstreicht. Die Ressourcen sind zudem skalierbar und es kann auf sehr umfangreiche Kapazitäten zugegriffen werden, wodurch eine große Economy of Scale ausgenutzt werden kann (27). Des Weiteren bietet das Cloud Computing zahlreiche nutzbare integrierte Services für einen schnelleren Aufbau und hierdurch die Förderung von bestehenden Best Practices - beispielsweise ein robuster CI/CD¹¹-Prozess. Gerade für neu aufkommende Projekte wie es DOSE ist, bietet das Cloud Computing eine flexible Alternative zu dem klassischen Aufbau von Serverstrukturen.

Um zu verstehen, wie die OTC in DOSE genutzt wird, bietet es sich an die Clouddefinition des National Institute of Standards and Technology zu nutzen:

„Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models.“ (28)

Die essenziellen Charakteristika wurden im obigen Abschnitt bereits angeschnitten und sollen hier nicht weitere im Detail erörtert werden. In Zusammenhang mit den erwähnten drei Service Modellen fungiert die OTC im DOSE-System als Infrastructure as a Service (IaaS). Im Detail heißt das, dass in der OTC Kapazitäten aufgebaut wurden, auf denen der Quellcode bereitgestellt wird und im Endeffekt für die Nutzer über eine Webapplikation zugänglich ist (siehe Kapitel 4.2.2). Die Modellierung als IaaS wurde gewählt, da für den Quellcode ein Open-Source-Ansatz gilt, der im Anschluss von anderen Nutzern/Anbietern zum Aufbau einer eigenen Infrastruktur genutzt oder in diese integriert werden kann. Zuletzt muss ein geeignetes Deploymentmodell festgelegt werden. Derzeit wird der Code in einer privaten Cloud verwaltet, doch kann dieses auch im weiteren Entwicklungsverlauf angepasst werden.

¹¹ Continuous Integration / Continuous Deployment. Hierbei handelt es sich um ein Paradigma im Software Development, das die kontinuierliche Verbesserung von Software sowie deren Einsatz über sogenannte Deployment-Pipelines umsetzt.

Technisch wird innerhalb der OTC die Infrastruktur für DOSE als Infrastructure as Code (IaC) mit Hilfe des Werkzeugs Terraform aufgebaut und verwaltet. Dies dient zusätzlicher Sicherheit sowie einem immer gleichen Prozess und erhöht so den Automatisierungsgrad. Die Microservices (vgl. nächstes Kapitel) werden in einem containerisierten Kubernetes-Cluster orchestriert. Der Einsatz von Kubernetes ist allerdings kein Zwang, sondern könnte auch durch beispielsweise Docker Swarm ersetzt werden.

Ausgehend von der grundlegenden Einführung in die DOSE-Cloud, die die Basis für das gesamte System bildet, wird nachfolgenden die IT-Architektur dargestellt und beschrieben, um dann die einzelnen Teilaspekte, wie die Webapplikation, das Authentifizierungstool, die eingesetzten Services, die zugrundeliegende Datenbank sowie die MQTT-Schnittstelle zu den OLI Boxen und schließlich zu den endgültigen Peripheriegeräten näher zu beleuchten.

4.2.1 IT-Architektur

Die aufgebaute IT-Infrastruktur wird anhand des nachfolgenden Bildes beschrieben (vgl. Bild) und paragraphenweise beschrieben. Ausgegangen wird hierbei von den Nutzerrollen und dem hauptsächlichen Interaktionspunkt mit dem DOSE-System – der Webapplikation.

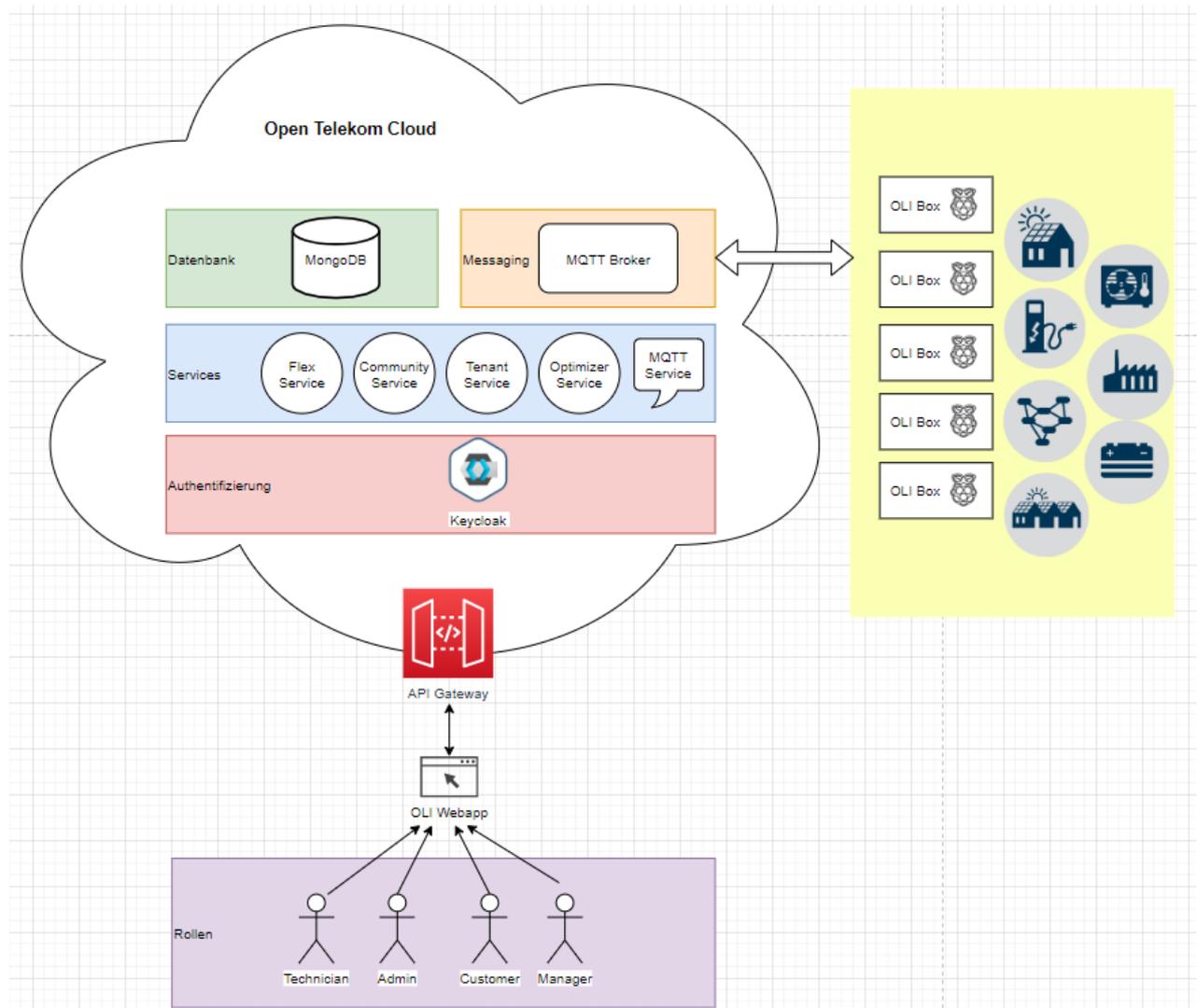


Abbildung 10 Übersicht über die DOSE Cloud

Webapplikation

Für die Entwicklung des Front-Ends wurde das zeitgemäße JavaScript-Framework Angular genutzt, welches für das Erstellen von Single-Page-Applikationen (SPA) eingesetzt wird und nativ auf der Programmiersprache TypeScript basiert.

SPAs und klassische Webanwendungen unterscheiden sich insofern, dass bei SPAs lediglich ein einzelnes HTML-Dokument existiert, welches Änderungen dynamisch lädt. Klassische Webanwendungen hingegen bestehen aus mehreren durch Hyperlinks verbundenen Dokumenten. Der Vorteil von SPAs gegenüber klassischen Webanwendungen liegt darin, dass die UI-Logik und die

benötigten Daten überwiegend clientseitig gelagert sind und verarbeitet werden. Dies führt zu einer Entlastung der Serverlast sowie eine flüssigeren Handhabung der Webapplikation – was heute von Anwendern erwartet wird.

TypeScript erweitert den Sprachumfang von JavaScript um Variablentypen und trägt so zur Fehlerminimierung während des Entwicklungsprozesses bei. Dies führt infolge zu einer sichereren und schnelleren Entwicklung sowie einem sichereren Produkt.

Neben Angular existieren weitere verbreitete SPA-Frameworks bzw. SPA-Bibliotheken, wie React oder Vue.js. Zum Zeitpunkt des Projektbeginns war allerdings nur React ein vergleichbares und zukunftssträchtiges Produkt. Technisch hätte die Wahl auf beide Tools fallen können, ohne die Entwicklung signifikant zu beeinträchtigen.

API-Gateway / NGINX

Die Notwendigkeit ein API-Gateway-Produkt einzusetzen basiert hauptsächlich auf der Microservicearchitektur im Back-End. Von zentraler Bedeutung des API-Gateway-Produkt ist hier der Reverse Proxy, der eine Kapselung der Microserviceendpunkte ermöglicht. Das heißt, dass Nutzeranfrage aus dem Front-End serverseitig dem richtigen Service bzw. den richtigen Services zugeordnet werden sowie eine Antwort erstellt wird. Des Weiteren ermöglicht ein Reverse Proxy einen evolutionären Architekturansatz, da Services im Hintergrund neu aufgebaut, zerschlagen oder verschmolzen werden können, ohne dass der Client diesen Änderungen direkt ausgesetzt ist.

Als Alternative zu einem Reverse Proxy existiert der Aufbau von Endpunktlisten. Die Verwaltung dieser wird aber bei einer zunehmenden Anzahl an Services unübersichtlich und somit fehleranfällig, weshalb von dem Einsatz solcher Listen abgesehen werden sollte.

Innerhalb des Projekts wurde der Reverse Proxy von NGINX genutzt, welcher derzeit eines der am meist verwendeten Produkte ist. Dies kann zum einen als Qualitätsmerkmal gesehen werden und gewährleistet des Weiteren einen langfristigen Support.

Authentifizierung – Keycloak

Keycloak ist ein Open Source Tool, welches zur Verwaltung von Authentifizierung und Autorisierung genutzt wird. Es bietet zum einen die derzeit notwendigen Features zur Gestaltung eines Loginprozess in Abhängigkeit von bestimmten Rollen sowie eine Vielzahl an weiteren Features, welche perspektivisch von Relevanz sein können. Zu nennen sind beispielsweise Zwei-Faktor-Authentifizierung oder Third-Party-Integration (Authentifizierung über Social Media).

Derzeit erfolgt der Authentifizierungsprozess wie folgt: Eine Nutzeranfrage in der Webapplikation wird über den Reverse Proxy NGINX an den richtigen Microservice weitergeleitet. Nachfolgend prüft der Microservice ob der User die notwendigen Berechtigungen besitzt und für in Abhängigkeit davon die geforderte Operation aus.

Services

Ein Microservice ist ein „leichtes“ Programm, das in eigenem Prozess läuft und über eigene Ressourcen verfügt. Die Kommunikation nach außen erfolgt über eine Schnittstelle, meistens über HTTP. Eine Software Architektur die aus Microservices aufgebaut ist, nennt man eine „Microservice Architecture“]. Dieser Ansatz erlaubt ein großes System aus kleinen, leichten und problemorientierten Programmen zu erstellen. Jeder Microservice kann unabhängig entwickelt, deployt und skaliert werden. Das spart Ressourcen, erhöht die Entwicklungsgeschwindigkeit und erlaubt die Entwicklung des Systems auf mehrere Teams zu verteilen. Die kleine Größe eines Microservices vereinfacht Entwicklung, Refactoring und Debugging. Obwohl die Entwicklung eines einzelnen Microservices einfacher wird, wird die Verwaltung (DevOps) schwieriger. Jeder einzelne Service muss deployt und monitort werden. Eine weitere Herausforderung ist die Kommunikation zwischen den Services. Obwohl es oftmals durch den Innovationsgrad bedingt Herausforderungen gibt, ist eine Microservice Architektur gut geeignet um große, komplexe und skalierbare Systeme zu entwickeln.

Datenbank

Eine der grundlegendsten Anwendungen ist die Speicherung, Verwaltung und Manipulation von Informationen innerhalb eines Systems, weshalb der Auswahl geeigneter Datenbanken aus IT-architektonischer Sicht ein besonderer Stellenwert zukommt. Die Möglichkeit unterschiedliche Datenbanken auszuwählen ist durch die eingesetzte Mikroservice-Architektur gegeben, in welcher es nicht unüblich ist, dass für verschiedene Services verschiedenen Datenbanken eingesetzt werden. Für den Optimizer-Service wurde beispielsweise eine SQLite-Datenbank und für die restlichen Services eine gemeinsame MongoDB-Datenbank genutzt. Der Optimizer-Service wird an anderer Stelle beschrieben, weshalb hier nur auf den Einsatz der dokumentenbasierten MongoDB eingegangen wird.

Historisch gesehen wurden überwiegend relationale Datenbanken eingesetzt, welche auch heutzutage noch weite Verbreitung finden. Mit den immer wachsenden Ansprüchen an die moderne Informationstechnologie ist jedoch zu hinterfragen, ob die Funktionalitäten weiterhin genügen. Im Zuge der Notwendigkeit Applikation für eine Größenordnung „web-scale“ verfügbar zu machen, wurden verschiedene NoSQL-Ansätze („Not only SQL“) entwickelt.

Eine der meistgenutzten Datenbanken ist hierbei die NoSQL-Datenbanken MongoDB, welche einen dokumenten-orientierten Ansatz nutzt. Die Datenspeicherung geschieht in BSON-Dokumenten (Binary JSON). Des Weiteren basiert sie auf einem schemalosen Speicherungsformat, was dazu führt, dass von Beginn an keine feste Datenstruktur notwendig ist sowie keine fixen Datentypen notwendig sind. Sie bietet somit ein hohes Maß an Flexibilität, welches in diesem Projekt als Vorteil gesehen wurde.

Dem DOSE-System können zudem Schlagwörter wie „Big Data“ oder „IoT“ zugeordnet werden, da durch die Boxen eine Vielzahl an sensorartigen Daten gesammelt werden. So werden über die Boxen in Intervallen von wenigen Sekunden Daten bereitgestellt, was für das System eine hohe Anzahl an Schreib- und Leseprozessen bedeutet.

Alles in allem haben die Flexibilität und mögliche Skalierung sowie die lange Beständigkeit der Datenbank zu der Auswahl geführt.

5 Einsatz

In den vorangegangenen Kapiteln wurde DOSE sowohl auf der theoretischen als auch auf der praktischen (Deployment) Ebene besprochen und Zusammenhänge aufgezeigt. Innerhalb dieses Abschnitts werden nun anhand einiger Projekte die Einsatzmöglichkeiten vorgestellt.

5.1 Allgäu Microgrid

Im Jahre 2018 wurde ein lokaler Energiemarkt mit DOSE Komponenten im Allgäu in der Zusammenarbeit mit der AÜW erstellt. Zentrales Ziel des Projekts war es, möglichst viele Komponenten des lokalen Energiemarktes on-chain, d.h. direkt auf der Blockchain zu platzieren und diese zu erproben. Auch die Interaktion des Users über eine App wurde analysiert um die Ecken und Kanten, die ein oft LEM mit sich bringt, zu erkennen. Es folgen an dieser Stelle Auszüge zu dem Projekt aus dem Buch „DLT im Energiesektor – Wie blockchainbasierte Werkzeuge und maschinelles Lernen ein dekarbonisiertes Energiesystem möglich machen“ (29).

Das Ziel des „Allgäu Microgrid“ Projektes ist die Implementierung eines lokalen Strommarktes unter dem Dach des Allgäuer Überlandwerkes als Stromlieferant, Verteilnetzbetreiber und Bilanzkreisverantwortlichem. Das Pilotprojekt dient einerseits dazu, Erfahrungen mit DLT-Anwendungen im Realbetrieb zu sammeln, andererseits aber auch dazu, eine Gesamtarchitektur zu entwerfen, die aus regulatorischer und technischer Sicht in den Realbetrieb übertragbar ist. Fünf Teilnehmer wurden für den Testbetrieb ausgewählt. Dabei war für den Betrieb des lokalen Marktes wichtig, dass deren Erzeugungs- und Verbrauchsprofile möglichst vielfältig sind. Für das Allgäu Microgrid wurden deshalb Haushalte mit Photovoltaikanlage, mit flexiblen Verbrauchern (Wärmepumpe), reine Verbraucherhaushalte und auch ein Gewerbebetrieb in die „Solar Community“ integriert. Das Erzeugungs- und Verbrauchsverhalten der Teilnehmer sollte unterschiedlich sein, um einen aktiven Handel im lokalen Markt zu ermöglichen und Clustereffekte zu vermeiden. Ein weiteres Ziel des Projektes ist es, den lokalen Markt inklusive der dazu notwendigen Hardware dezentral mittels eines Smart Contracts zu gestalten und ohne zentrale Plattform zu realisieren.

Im Hinblick auf die Umsetzung im Realbetrieb bietet eine dezentrale Architektur mehrere Vorteile:

- Durch den Verzicht auf einen zentralen Server wird das Gesamtsystem wesentlich ausfallsicherer. Selbst wenn mehrere Knoten des Blockchainnetzwerkes ausfallen, sind einerseits die Marktdaten mehrfach redundant vorhanden, andererseits gewährleisten die verbleibenden Knoten den reibungslosen Weiterbetrieb der Anwendung.
- Der Betreiber des lokalen Marktes kann gleichzeitig als Anbieter und Nachfrager auftreten, ohne in einen Interessenskonflikt zu geraten. Der implementierte Smart Contract ist für alle Marktteilnehmer derselbe, eine Manipulation zugunsten des Betreibers ist ausgeschlossen.

- Durch die Schaffung eines neutralen, standardisierten Marktplatzes wird die Einbindung von weiteren Teilnehmern – insbesondere von Erzeugungsanlagen unterschiedlicher Betreiber - wesentlich erleichtert.

Die Marktlogik wird im Projekt komplett auf Basis des Smart Contracts abgebildet. Konkret bedeutet dies, dass sowohl die Regeln bezüglich Bietverfahren, Preisbildung und Markträumung dort festgelegt sind, als auch die automatisierte Ausführung des Verfahrens komplett über den Contract erfolgt. Wesentlich für das Projekt ist es ebenfalls, den Endkunden eine einfach zugängliche und intuitiv verständliche Schnittstelle zum lokalen Markt anzubieten. Über eine Smartphone-App hat der Endkunde Zugang zu seinen Erzeugungs- und Verbrauchsdaten. Die aktive Interaktion mit dem lokalen Markt findet ebenfalls über die Smartphone-App statt. Die Benutzeroberfläche unterscheidet dabei nicht zwischen Daten, die aus konventionellen Datenbanken bereitgestellt werden und solchen, die aus der Blockchain kommen.



Abbildung 11 Beispielinstallation einer OLI Box an einer modernen Messeinrichtung

Architektur

Die Erfassung von Leistungs- und Stammdaten erfolgt über ein sogenanntes Blockchain-Gateway (sprich: OLI Box). Dieses hat die Aufgabe, auf der einen Seite Daten aus Stromzählern und Erzeugungsanlagen in hoher Qualität zu erfassen und auf der anderen Seite die Verbindung zwischen den Teilnehmern über ein Kommunikationsnetzwerk sicherzustellen. Dies kann über das

nutzereigene WLAN, einen kabelgebundenen LAN-Anschluss oder das Mobilfunknetz erfolgen, je nach Verfügbarkeit vor Ort. Das Blockchain-Gateway dient gleichzeitig als Light Node innerhalb des Blockchainnetzwerkes. Dies bedeutet, dass eine lokale Kopie der Blockchainheader zur Prüfung vorgehalten wird, aber der Teilnehmer keine aktive Validierungsfunktion übernehmen kann. Dabei wird berücksichtigt, dass das Blockchain-Gateway nur über eine begrenzte Rechenleistung verfügt und selbst nicht aktiv zur Aufrechterhaltung des Netzwerkes beitragen muss. Als Konsensmechanismus kommt eine Proof-of-Authority (PoA) Variante („Aura“) zum Einsatz. Dies sorgt für einen dramatisch

geringeren Energieverbrauch pro Transaktion im Vergleich zu öffentlichen Blockchains, die auf Proof-of-Work basieren. Im Projekt validieren sechs Authority Nodes die Transaktionen. Diese sind wiederum verteilt auf drei Unternehmen. Es ist möglich, Authority Nodes auf eigener Hardware zu betreiben oder in einen Clouddienst (z.B. Open Telekom Cloud, Amazon AWS, Strato, etc.) auszulagern. Der dafür notwendige administrative Aufwand ist überschaubar, Authority Nodes sind typischerweise innerhalb weniger Stunden einsatzbereit.

Die im Projekt erhobenen Daten werden nicht vollständig auf der Blockchain gespeichert. Wie in den meisten Fällen, in denen Blockchainkomponenten zum Einsatz kommen, werden diese nur dort verwendet, wo sie einen konkreten Mehrwert bieten. Dies bedeutet im Fall des Allgäu Microgrid, dass wie in Abbildung 12 skizziert, die Blockchain zur Nutzerregistrierung, dem Sammeln und Aggregieren von Geboten für Stromein- und -verkäufe, der individuellen Berechnung der Markträumungsergebnisse für jeden Nutzer sowie zur Kommunikation der Marktergebnisse verwendet wird. Zur Speicherung hochauflöser Verbrauchsdaten hingegen kommt zum Beispiel eine konventionelle Datenbanklösung zum Einsatz.

Die eingesetzte permissioned Blockchain mit PoA-Konsensmechanismus genügt bezüglich der Transaktionsgeschwindigkeit und Skalierbarkeit voll den Anforderungen und lässt auch eine größere Zahl an Teilnehmern zu. Soll die Anwendung auf das ganze Netzgebiet ausgewertet, beziehungsweise deutschlandweit angeboten werden bietet es sich an, auf eine weltweit verfügbare Alternative wie die Blockchain der Energy Web Foundation zurückzugreifen. Diese ist technisch der im Allgäu realisierten Lösung sehr ähnlich, d.h. ein Umzug auf die Infrastruktur der Energy Web Foundation wäre mit vergleichsweise geringem Aufwand möglich.

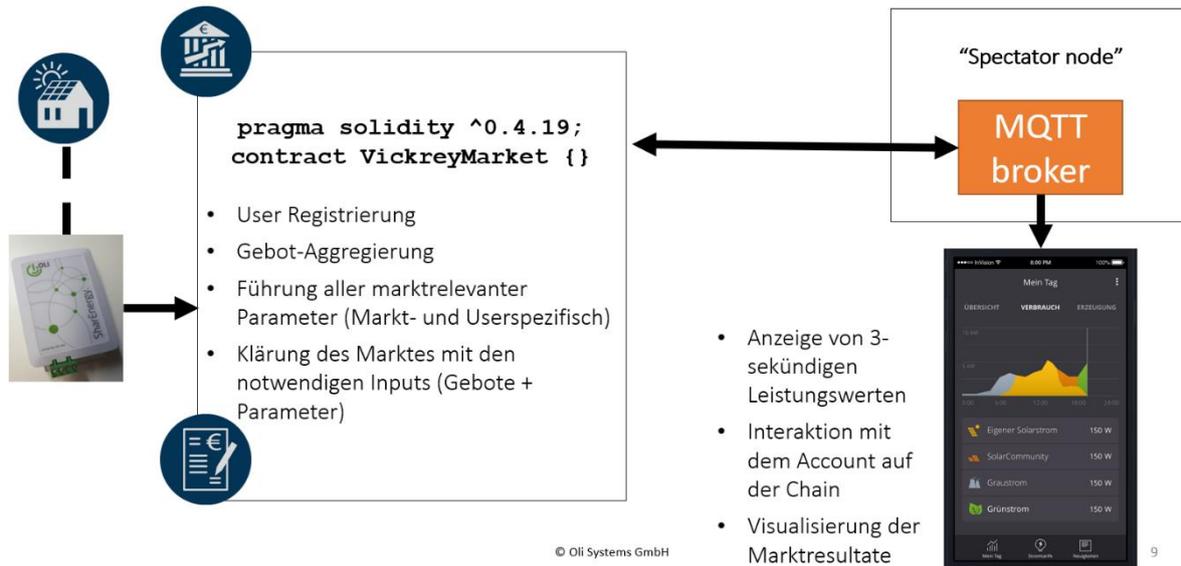


Abbildung 12 Features im Allgäu Microgrid

Nach Eingabe von Benutzername und Passwort (automatisierbar), gelangt der Nutzer auf die Startseite, auf der alle relevanten Parameter zu Erzeugung, Verbrauch und Kosten des jeweiligen Tages einsehbar sind. Dabei werden bereits Daten aus der Blockchain bezüglich des Handels in der Solar Community berücksichtigt, da der Preis pro Kilowattstunde hier je nach Ergebnis der Auktion alle 15 Minuten ein anderer ist.

Über die App kann der Nutzer aktiv seine Gebote für Strom aus der Solar Community abgeben, den Stromtarif zur Reststrombelieferung wählen und eine genaue Übersicht über die Bezugsquellen seines Stromes erhalten (Abbildung 13, Mitte). Eine Anpassung des Gebotspreises oder des Tagesbudgets kann theoretisch alle 15 Minuten erfolgen. Ebenso kann der Tarifwechsel – in diesem Fall zwischen „Grünstrom“ und „Graustrom“ innerhalb des Lieferanten theoretisch alle 15 Minuten erfolgen.

Die Liefermengen und die dazugehörigen Preise pro Kilowattstunde werden transparent in der Blockchain dokumentiert und können im Falle einer Überführung in den Realbetrieb dazu genutzt werden, jedem Kunden eine detaillierte Verbrauchsrechnung zu erstellen, analog zum heute im Telekommunikationsbereich üblichen Einzelverbindungs nachweis.

Für den aktuellen Tag erhält der jeweilige Kunde über die App eine Auswertung zu Herkunft und Kosten des bezogenen Stromes. Dort lässt sich viertelstundengenau die Zusammensetzung grafisch und numerisch ablesen. Die dort angezeigten Daten kommen direkt aus der Blockchain. Außerdem dient die App als neuer Kundenkanal für das Allgäuer Überlandwerk, da über die Nachrichtenfunktion der App Neuigkeiten aus dem Unternehmen direkt auf das Smartphone des Kunden kommen.



Abbildung 13 Einsichten in die App

5.2 Canary Network zusammen mit der Grid Singularity

Zusammen mit Grid Singularity wurde ein Canary¹² Network zur Umsetzung eines lokalen Energiemarktes für den Odyssey Hackathon im November 2020 erfolgreich umgesetzt. (30)

Drei Teams (darunter auch OLI Systems) „kämpften“ in diesem Wettbewerb mit Ihren eigenen Bietagenten um die beste Performance¹³ in einem lokalen Energiemarkt. Dieser wurde durch einen digitalen Zwilling dargestellt, in dem OLI Boxen, SMGWs sowie weitere Drittpartei-APIs genutzt wurden (Abbildung 14).

Insgesamt waren 14 Messpunkte eingebunden. Darunter (Prosumer) Haushalte, Gewerbeeinheiten sowie die Windkraftanlage aus Kapitel (5.3).

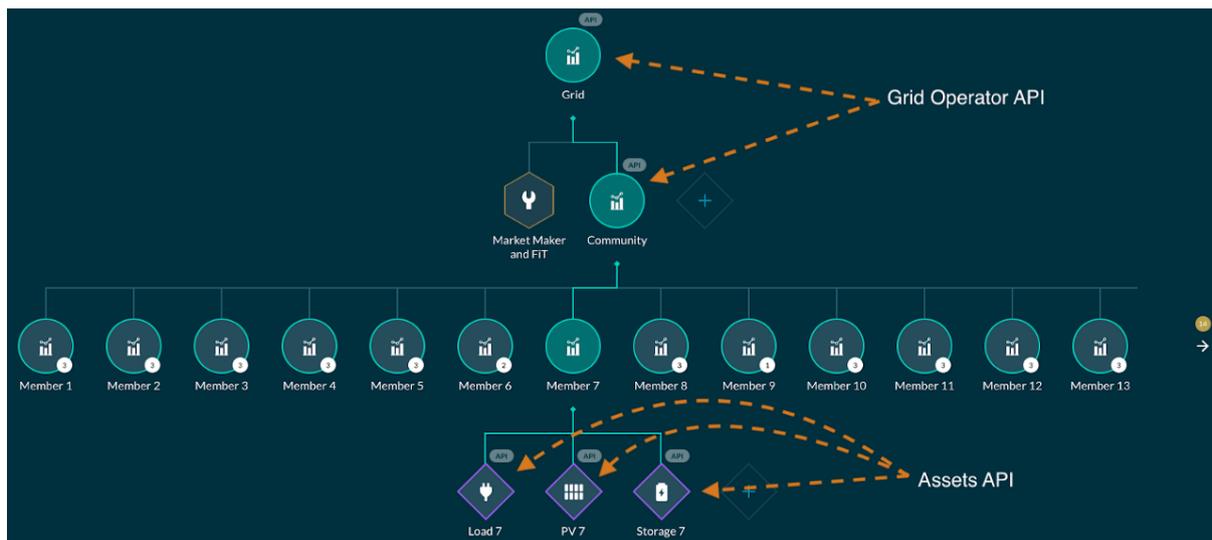


Abbildung 14 Digitaler Zwilling des Canary Networks in D3A

Das Handeln auf dem lokalen Energiemarkt läuft dabei im sogenannten double-sided pay-as-bid ab. Dies bedeutet, dass sowohl Angebot als auch Nachfragegebote (Sprich: Erzeugung und Verbrauch) auf dem Markt aufeinandertreffen. Dabei sind Preis pro kWh und Volumina die wichtigsten Parameter in jedem Gebot, aber auch die Herkunft des Stroms wird dargestellt. Sehr effektiv gestaltet sich auch die Einflussnahme des Verteilnetzbetreibers auf das Handelsgeschehen. Je nach Netzzustand kann der VNB die Preise für die Netznutzung erhöhen und verringern, und dabei auch die Netztopologie mit einbeziehen. Befindet sich zum Beispiel das Netz derzeit in der „gelben Phase“ laut der BDEW Ampel, so könnte der VNB aktiv eine Art Strafgebühr für das Trading über die Netzebenen hinweg erheben.

¹² Der Begriff „Canary“ bezieht sich auf die Kanarienvögel, die früher im Untertagebau eingesetzt wurden und deren Zustand die Bergarbeiter vor gefährlichen Gasen warnte. Analog soll das Canary Network uns frühzeitig davor warnen, welche Herausforderungen bei der Umsetzung von LEMs auftreten können.

¹³ Die beste Performance wird in diesem Kontext hauptsächlich durch einen möglichst hohen Umsatz für die vom Agenten verwalteten Energieanlagen im LEM definiert. Prinzipiell sind aber auch möglichst hohe Deckungsgrade, die Vermeidung von Netzentgelten oder andere Metriken als Orientierungsgröße möglich.

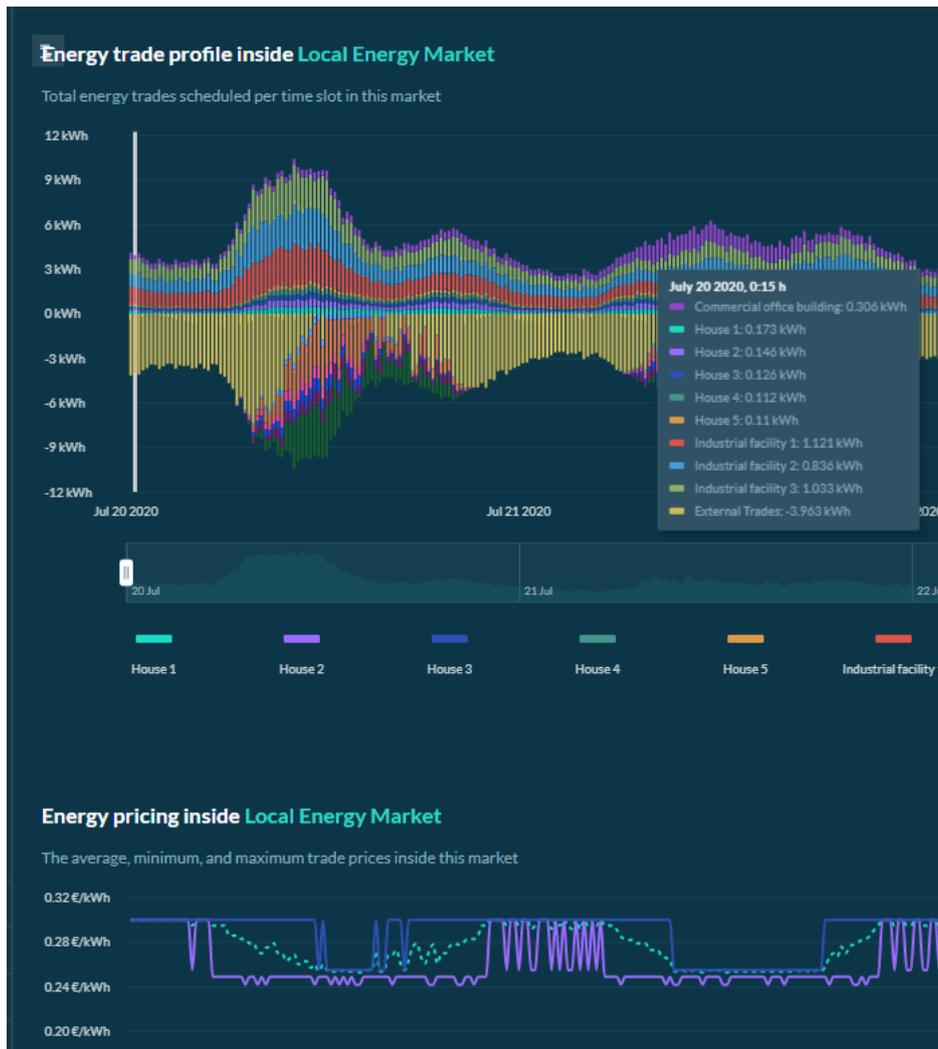


Abbildung 15 Tradingprofile im Canary Network

5.3 Windkraftanlage Edersleben als Teil der DOSE Community

OLI Boxen wurden auch in einem Projekt zusammen mit der Firma EXAMESH eingesetzt. EXAMESH platziert Server-Racks innerhalb von Windkraftanlagen und ermöglicht es hiermit den Kunden, energieintensive Berechnungen direkt mit Strom aus einer EE-Anlage durchzuführen. Besonders interessant war hier der Einsatz von DOSE als Transparenztool, um die Erzeugung viertelstundenscharf darzustellen. Die von uns angebundene Windkraftanlage befindet sich in Edersleben in Sachsen-Anhalt.



Abbildung 16 Eine der Windkraftanlagen in Edersleben

DOSE-Code wurde auf einem industrieharten Gerät platziert, das über einen Impulsrecorder den Zähler des Messstellenbetreibers ausliest. Durch die Systeme vor Ort (Abbildung 17) können die Informationen im viertelstündlichen Intervall an die jeweiligen Plattformen weitergegeben werden. Die Daten werden für Visualisierungs- und Transparenzzwecke eingesetzt. Insgesamt handelt es sich um drei WKA mit einer Gesamtleistung von 1,8 MW an diesem Standort. Die Anlage wurde außerdem in die DOSE Cloud und die Community eingebunden und vermarktete virtuell in nahezu Echtzeit im Canary Network (siehe Kapitel 5.2) direkt Energie an Haushalte und Gewerbe.

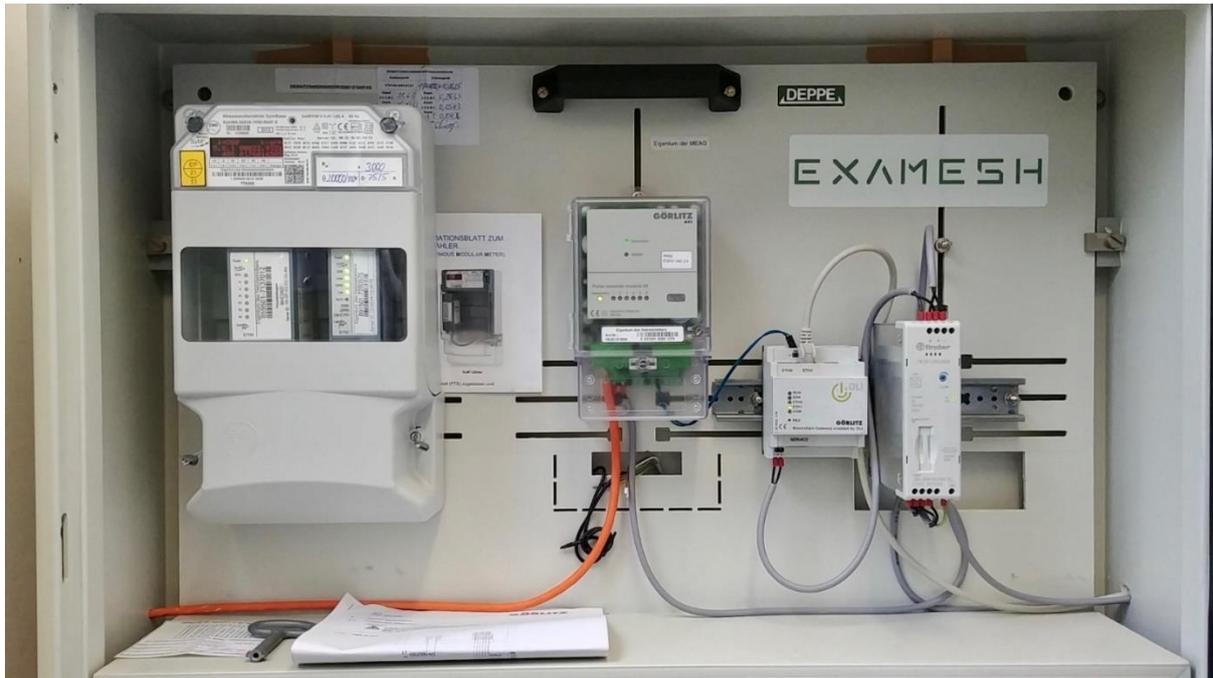


Abbildung 17 Installation an der EXAMESH Windkraftanlage

5.4 ifeu Heidelberg

Im Institut für Energie- und Umweltforschung in Heidelberg wurde DOSE ebenfalls installiert. Neben einer komplett eigenen Netzwerkinfrastruktur inkl. 3G/4G Router und Lichtwellenwandler wurden OLI Boxen an folgenden Energieanlagen bzw. deren Zähler platziert:

- Hauptzähler
- PV-Anlagen Zähler
- Wallbox

Die OLI Box liest die Geräte über verschiedene Varianten (IR, REST-Schnittstelle) aus und sendet die Messwerte an die DOSE Cloud. Dort können die Informationen über das DOSE-Dashboard visualisiert werden.

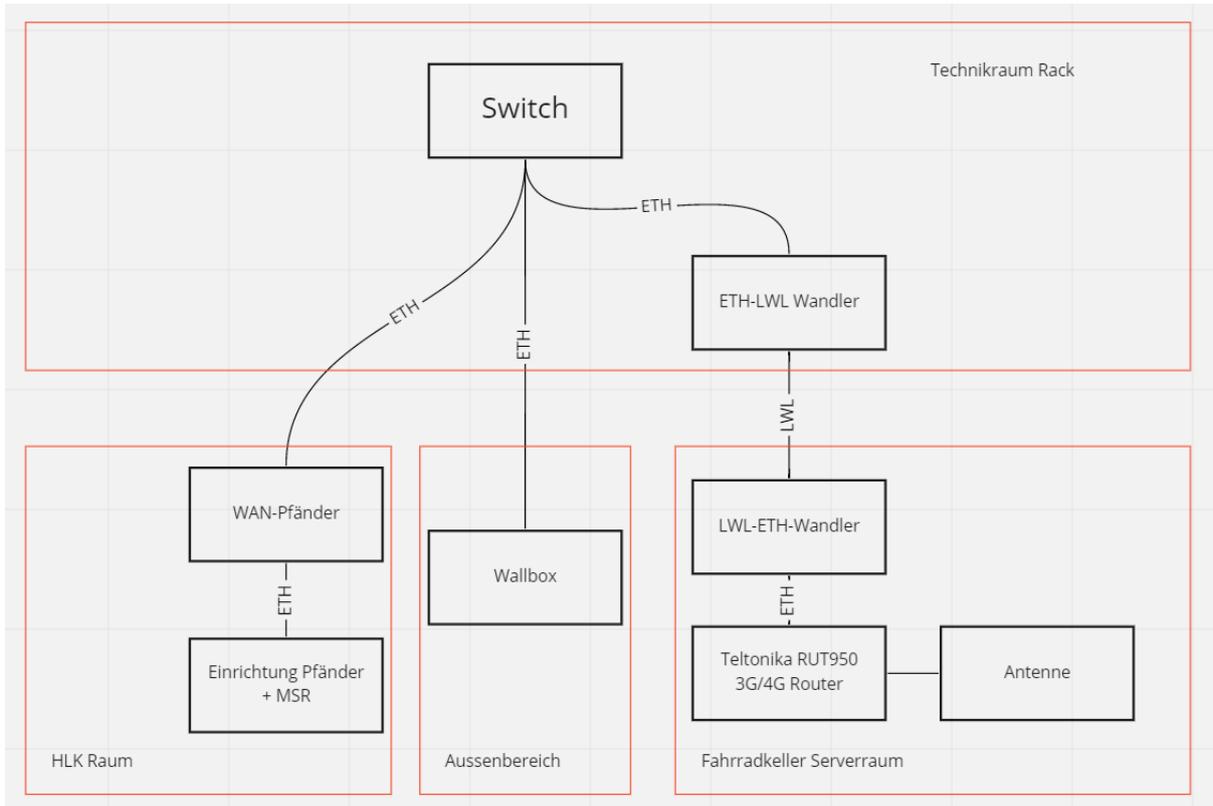


Abbildung 18 Eigene Netzwerkinfrastruktur für DOSE am ifeu

Zudem wurde ein Optimierer entwickelt, der auf Basis der Raumbuchungen eines Konferenzsaals, der prognostizierten PV-Leistung und der Belegung der Wallbox möglichst die Kühlleistung und Ladeleistung in Zeiten verlegt, an denen ein Überschuss an eigenem PV-Strom vorliegt (Abbildung 19). Durch die Erhöhung des Eigenstromverbrauchs wird Strom aus erneuerbaren Energien effizient und netzdienlich genutzt (Abbildung 20).

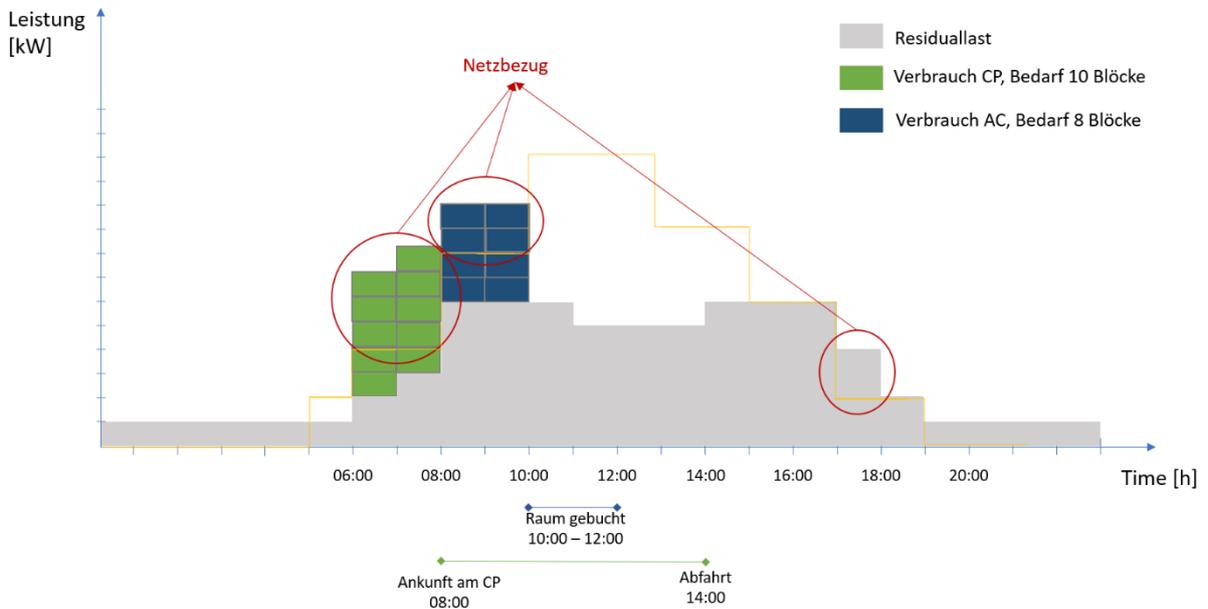


Abbildung 19 DOSE-Optimierer Prozess bei Prognose

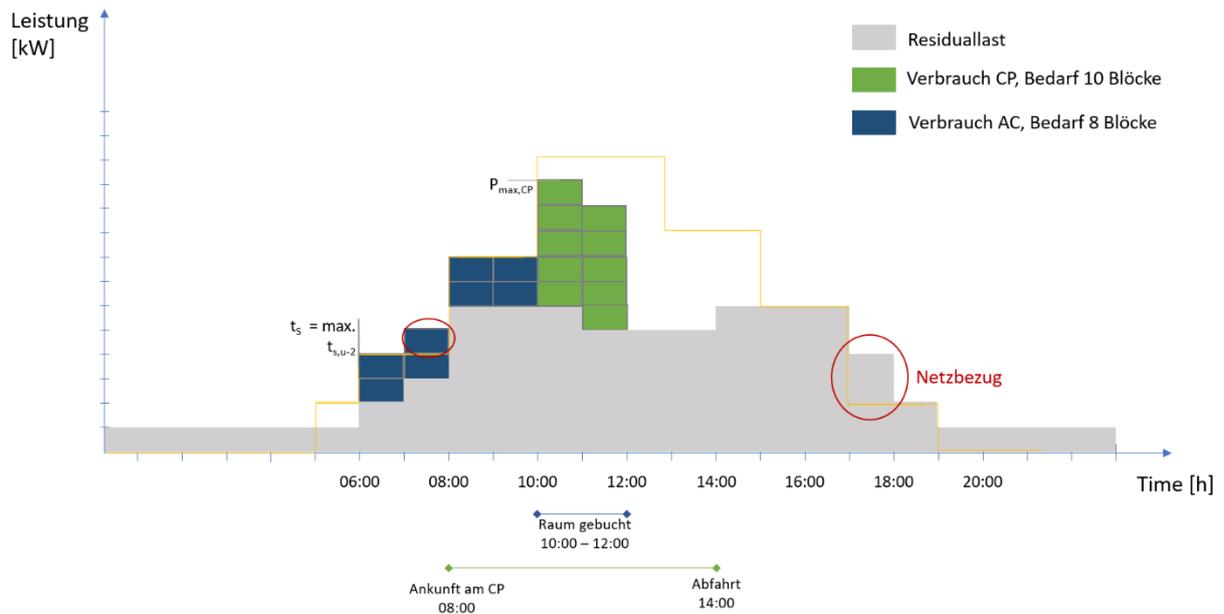


Abbildung 20 DOSE-Optimierer nach Durchführung der eigentlichen Optimierung



Abbildung 21 Zwei OLI Boxen mit PoE Modulen. Die untere OLI Box ist mit dem Hauptzähler verbunden.

6 Ausblick und Transfer

Wie in Kapitel 5 bereits beschrieben, kommen die in DOSE entwickelten Komponenten in einer ganzen Reihe von verschiedenen Projekten aktiv zum Einsatz. Dieser Prozess setzt sich derzeit auch fort und die meisten Projekte werden weitergeführt. So wird das Canary Network inklusive der OLI Boxen aktiv betrieben und erweitert. Eine Ausnahme stellt hier das ifeu dar. Anstatt von DOSE wird langfristig das vom eigentlichen HLK-Unternehmen bereitgestellte EMS weiter genutzt. Jedoch wurde die Netzwerkinfrastruktur übernommen und die Erfahrungen bei der Inbetriebnahme, Instandhaltung und Nutzung ausgetauscht.

Einen weiteren neuen Einsatzzweck und -ort für OLI Boxen stellt das von der deutschen Energieagentur (DENA) ausgeschriebene Projekt „Blockchain Machine Identity Ledger“ dar (31). In diesem Projekt werden OLI Boxen mit einem Hardware Security Module („Krypto-Chip“) ausgerüstet und mit einer Energieanlage wie z.B. einem PV-Wechselrichter verbunden. Die OLI Boxen verbinden sich über ein SMGW der Firma PPC mit einem externen Marktteilnehmer und übermitteln mit kryptographischen Signaturen eine Art Stammzertifikat der Anlage sowie dynamische Messwerte in bestimmten Zeitabständen.



Abbildung 22 Die OLI Box in Verbindung mit PPC SMGW und CLS-GW im derzeitigen BMIL Labor

Die Identitäten werden passend zum DOSE Gedanken nicht über zentralisierte, Silo-artige Dienste umgesetzt, sondern über den verifiable credentials und decentralized identifiers Standard der W3C.

Dieser Ansatz unterstützt auf dezentrale Art und Weise mit dem Einsatz verschiedener Blockchain-Netzwerke das Marktstammdatenregister und soll letztendlich den Plug-and-Play Betrieb von Energieanlagen in u.a. Flex-Märkten, Herkunftsnachweisprozessen und in der Präqualifikation ermöglichen. Das Projekt begann im Herbst 2020 und ein finaler Bericht wird zum DENA Kongress im November 2021 veröffentlicht. Das BMIL Projekt ist ein sehr gutes Beispiel für die Nutzung von DOSE in proof-of-concepts und agilen Entwicklungsprozessen.

Über das Projekt hinausgehend beschäftigen uns derzeit noch folgende Fragestellungen:

- Wie sieht die optimale User-Experience für lokale Energiemärkte und Communities aus? Welche „Tiefe“ an Management ist für ein lokales Energiesystem wirklich gefragt?
- Wie kann die Verbindung der iMSys Architektur mit Open-Source Energiemanagementsystem langfristig gestaltet werden?
- Wie verhält sich langfristig die Stabilität der Hardware- und Softwarekomponenten?
- Wie sehen die Schnittstellen zum Netzbetreiber in der Zukunft aus? Sind dynamische Netzentgelte wie in unserem Testnetzwerk interessant und eventuell sogar ausreichend, um Netzengpässe dauerhaft zu vermeiden?
- Schafft es der Open Source Gedanke langfristig in der Energiewirtschaft Fuß zu fassen? Ist der offene Charakter von Blockchains und anderen Distributed-Ledger Technologien bereits ausreichend von den großen Stakeholdern (Netzbetreiber, Energieversorger etc.) verstanden?

7 Verzeichnisse

7.1 Quellenverzeichnis

1. **DAS EUROPÄISCHE PARLAMENT UND DER RAT DER EUROPÄISCHEN UNION . RICHTLINIE (EU) 2019/944 DES EUROPÄISCHEN PARLAMENTS UND DES RATES vom 5. Juni 2019 mit gemeinsamen Vorschriften für den Elektrizitätsbinnenmarkt und zur Änderung der Richtlinie 2012/27/EU.** Brüssel : Europäische Union, 2019.
2. **DAS EUROPÄISCHE PARLAMENT UND DER RAT DER EUROPÄISCHEN UNION. RICHTLINIE (EU) 2018/2001 DES EUROPÄISCHEN PARLAMENTS UND DES RATES vom 11. Dezember 2018 zur Förderung der Nutzung von Energie aus erneuerbaren Quellen.** Brüssel : Europäische Union, 2018.
3. **Michal Glowacki.** Website EUTS Market Glossary . *Website EUTS Market Glossary* . [Online] 03. 02 2020. [Zitat vom: 03. 02 2020.] <https://www.emissions-euets.com/internal-electricity-market-glossary/2095-energy-community>.
4. **REScoop.eu vzw.** Rescoop.eu Homepage: Q&A: what-are-citizen-and-renewable-energy-communities. *Rescoop.eu Homepage: Q&A: what-are-citizen-and-renewable-energy-communities*. [Online] 3. 7 2019. [Zitat vom: 08. 05 2020.] <https://www.rescoop.eu/blog/what-are-citizen-and-renewable-energy-communities>.
5. **Sonnen GmbH.** Sonnen GmbH Homepage - Erläuterungen zur Stromcommunity. *Sonnen GmbH Homepage - Erläuterungen zur Stromcommunity*. [Online] 05. 04 2020. [Zitat vom: 05. 04 2020.] <https://sonnen.de/sonnencommunity/>.
6. **eprimo GmbH.** eprimo Homepage und Erläuterungen zur Grünstromcommunity. *eprimo Homepage und Erläuterungen zur Grünstromcommunity*. [Online] 08. 05 2020. [Zitat vom: 08. 05 2020.] <https://www.eprimo.de/oekostrom/gruenstromcommunity/>.
7. **HEG Heidelberger Energiegenossenschaft eG.** HEG Homepage: Bürgerstrom beziehen und investieren. *HEG Homepage: Bürgerstrom beziehen und investieren*. [Online] 01. 03 2020. [Zitat vom: 01. 03 2020.] <https://www.heidelberger-energiegenossenschaft.de/mitmachen/buergerstrom-beziehen-investieren>.
8. **Mengelkamp, Esther, et al.** *Designing microgrid energy markets: A case study: The Brooklyn Microgrid.* Karlsruhe : Applied Energy, 2018.
9. **Raspberry Pi Foundation.** Raspberry Pi 3B+ Product Brief. [Online] 2019. [Zitat vom: 02. 05 2020.] <https://static.raspberrypi.org/files/product-briefs/200206+Raspberry+Pi+3+Model+B+plus+Product+Brief+PRINT&DIGITAL.pdf>.
10. —. Raspberry Pi 4 Product Brief. [Online] 1. 6 2019. [Zitat vom: 02. 05 2020.] <https://static.raspberrypi.org/files/product-briefs/200521+Raspberry+Pi+4+Product+Brief.pdf>.
11. **Open Energy Monitor.** Open Energy Monitor Shop: IR Sensor. [Online] Open Energy Monitor / EmonPi, 01. 06 2020. [Zitat vom: 01. 06 2020.] <https://shop.openenergymonitor.com/optical-utility-meter-led-pulse-sensor/>.
12. **nd-enerserve GmbH.** Smart PI Beschreibung auf der enerserve Homepage. *Smart PI Beschreibung auf der enerserve Homepage*. [Online] 03. 05 2020. [Zitat vom: 03. 05 2020.] <https://www.enerserve.eu/de/smartpi.html>.

13. **Docker Inc.** Docker Homepage. *Docker Homepage*. [Online] 20. 06 2020. [Zitat vom: 20. 06 2020.] <https://www.docker.com/>.
14. **FIWARE Foundation e.V.** FIWARE Homepage. *FIWARE Homepage*. [Online] 22. 06 2020. [Zitat vom: 22. 06 2020.] <https://www.fiware.org/>.
15. **Canonical Ltd.** Ubuntu Snap/Snapcraft Homepage. *Ubuntu Snap/Snapcraft Homepage*. [Online] 22. 06 2020. [Zitat vom: 22. 06 2020.] <https://snapcraft.io/>.
16. **ZeroTier, Inc.** Zero Tier Website. *Zero Tier Website*. [Online] 22. 06 2020. [Zitat vom: 22. 06 2020.] <https://www.zerotier.com/>.
17. **Balena Inc.** Balena Homepage. *Balena Homepage*. [Online] 22. 06 2020. [Zitat vom: 22. 06 2020.] <https://www.balena.io/>.
18. **Upswift Inc.** Upswift Homepage. *Upswift Homepage*. [Online] 22. 06 2020. [Zitat vom: 22. 06 2020.] <https://www.upswift.io/>.
19. **remot3.it Inc.** Remote.it Homepage. *Remote.it Homepage*. [Online] 22. 06 2020. [Zitat vom: 22. 06 2020.] <https://remote.it/>.
20. **Python Software Foundation.** Python 3.8 Download und Infopage. *Python 3.8 Download und Infopage*. [Online] 22. 06 2020. [Zitat vom: 22. 06 2020.] <https://www.python.org/downloads/release/python-380/>.
21. **Raspbian Developers.** Raspbian Homepage. *Raspbian Homepage*. [Online] 22. 06 2020. [Zitat vom: 22. 06 2020.] <https://www.raspbian.org/>.
22. **Python Software Foundation.** Pip Package für Raspberry Pi GPIO. [Online] 19. 06 2020. [Zitat vom: 19. 06 2020.] <https://pypi.org/project/RPi.GPIO/>.
23. **TOML Developers.** TOML Github Homepage. *TOML Github Homepage*. [Online] 08. 05 2020. [Zitat vom: 08. 05 2020.] <https://github.com/toml-lang/toml>.
24. **OLI Systems GmbH.** Code-Repository OLIBOX-Core. *Code-Repository OLIBOX-Core*. [Online] 22. 06 2020. [Zitat vom: 22. 06 2020.] https://github.com/olisystems/olibox_core.
25. —. OLIBOX-core Snapcraft Homepage. *OLIBOX-core Snapcraft Homepage*. [Online] 03. 05 2020. [Zitat vom: 03. 05 2020.] <https://snapcraft.io/install/olibox-core/ubuntu>.
26. **Riti, Pierluigi.** *Pro DevOps with Google Cloud Platform: With Docker, Jenkins, and Kubernetes*. New York : Apress, 2018. ISBN 978-1-4842-3896-7.
27. **Baun, Christian, et al.** *Cloud Computing: Web-basierte dynamische IT-Services*. Heidelberg : Springer, 2011. ISBN 978-3-642-18435-2.
28. **Mell, Peter und Grance, Timothy.** *The NIST Definition of Cloud Computing*. Gaithersburg : US Department of Commerce, 2011.
29. **Brenner, Thomas.** *DLT im Energiesektor – Wie blockchainbasierte Werkzeuge und maschinelles Lernen ein dekarbonisiertes Energiesystem möglich machen*. Stuttgart : Springer, 2019. DOI 10.1007/978-3-662-60408-3_6.
30. **Grid Singularity GmbH.** Energy Singularity Challenge 2020: Testing Novel Grid Fee Models and Intelligent Peer-to-Peer Trading Strategies. [Online] 10. 12 2020. [Zitat vom: 10. 12 2020.] <https://gridsingularity.medium.com/energy-singularity-challenge-2020-testing-novel-grid-fee-models-and-intelligent-peer-to-peer-6a0d715a9063>.
31. **Deutsche Energie Agentur (dena).** *Information zu BMIL*. [Online] 20. 01 2021. [Zitat vom: 20. 01 2021.] <https://future-energy-lab.de/piloten/machine-id-ledger/>.
32. **ZigBee Alliance.** ZigBee Homepage. *ZigBee Homepage*. [Online] 03. 05 2020. [Zitat vom: 03. 05 2020.] <https://zigbeealliance.org/de/>.

33. **KNX Association.** KNX Homepage. *KNX Homepage*. [Online] 02. 05 2020. [Zitat vom: 02. 05 2020.] <https://www.knx.org/knx-en/for-professionals/index.php>.
34. **EEBus Initiative e.V.** EEBus Initiative Homepage. *EEBus Initiative Homepage*. [Online] 01. 05 2020. [Zitat vom: 01. 05 2020.] <https://www.eebus.org/>.
35. —. Mitgliederliste EEBus Initiative e.V. *Mitgliederliste EEBus Initiative e.V.* [Online] 04. 06 2020. [Zitat vom: 04. 06 2020.] <https://www.eebus.org/about-us/#members>.
36. **Holthuis, Jan.** *Der Aufbau von Warehouse-Systemen, Konzept, Datenmodellierung, Vorgehen*. Wiesbaden : Deutscher Universitätsverlag, 1999. 978-3-322-92336-3.
37. **SMA Solar Technology AG.** ennexOS Homepage. *ennexOS Homepage*. [Online] 03. 04 2020. [Zitat vom: 03. 04 2020.] <https://www.ennexos.com/>.
38. —. SMA Developer Landing Page. *SMA Developer Landing Page*. [Online] 03. 05 2020. [Zitat vom: 03. 05 2020.] <https://developer.sma.de>.
39. **OLI Systems GmbH.** Code-Repository OLIBOX-Control. *Code-Repository OLIBOX-Control*. [Online] 03. 05 2020. [Zitat vom: 03. 05 2020.] https://github.com/olisystems/olibox_control.
40. —. OLIBOX-control Snapcraft Homepage. *OLIBOX-control Snapcraft Homepage*. [Online] 05. 06 2020. [Zitat vom: 05. 06 2020.] <https://snapcraft.io/install/olibox-control/raspbian>.
41. **FHEM e.V.** FHEM Homepage. *FHEM Homepage*. [Online] 03. 05 2020. [Zitat vom: 03. 05 2020.] <https://fhem.de/>.
42. **OpenEMS Association e.V.** OpenEMS Homepage. *OpenEMS Homepage*. [Online] 05. 02 2020. [Zitat vom: 05. 02 2020.] <https://openems.io/>.
43. **openHAB Foundation e.V.** openHAB Homepage. *openHAB Homepage*. [Online] 05. 06 2020. [Zitat vom: 05. 06 2020.] <https://www.openhab.org/>.
44. **Allerding, Florian.** Organic Smart Home - Energiemanagement für intelligente Gebäude. *Organic Smart Home - Energiemanagement für intelligente Gebäude*. [Online] 25. 7 2013. [Zitat vom: 02. 05 2020.] <https://d-nb.info/1051753791/34>.

7.2 Abbildungsverzeichnis

Abbildung 1 Einfache Systemansicht des Brooklyn Microgrid. (8).....	6
Abbildung 2 DOSE Rollenstruktur und Hierarchien	9
Abbildung 3 Beispiel Schemata des DOSE-Optimierers.	12
Abbildung 4 Beispielhafte Installation einer OLI Box an einer modernen Messeinrichtung.....	14
Abbildung 5 OLI Box kommuniziert mit einer modernen Messeinrichtung.....	16
Abbildung 6 Der SmartPi mit Messwandler (12)	17
Abbildung 7 OLI Box Architektur	19
Abbildung 8 Fleet-Management von verteilten Geräten über das Internet.....	22
Abbildung 9 remot3.it Dashboard mit einer Auswahl an OLI Boxen	22
Abbildung 10 Übersicht über die DOSE Cloud	25
Abbildung 11 Beispielinstallation einer OLI Box an einer modernen Messeinrichtung	30
Abbildung 12 Features im Allgäu Microgrid	32
Abbildung 13 Einsichten in die App	33
Abbildung 14 Digitaler Zwilling des Canary Networks in D3A	34

Abbildung 15 Tradingprofile im Canary Network	35
Abbildung 16 Eine der Windkraftanlagen in Edersleben	36
Abbildung 17 Installation an der EXAMESH Windkraftanlage	37
Abbildung 18 Eigene Netzwerkinfrastruktur für DOSE am ifeu	38
Abbildung 19 DOSE-Optimierer Prozess bei Prognose	38
Abbildung 20 DOSE-Optimierer nach Durchführung der eigentlichen Optimierung	39
Abbildung 21 Zwei OLI Boxen mit PoE Modulen. Die untere OLI Box ist mit dem Hauptzähler verbunden.	39
Abbildung 22 Die OLI Box in Verbindung mit PPC SMGW und CLS-GW im derzeitigen BMIL Labor..	40