Gefördert durch:

Bundesministerium
für Wirtschaft
und Klimaschutz

aufgrund eines Beschlusses
des Deutschen Bundestages

Abschlussbericht

FlexHub - VERTEILTES FLEXIBILITÄTSDATENREGISTER FÜR STROMMÄRKTE DER ENERGIEWENDE

Bewilligungszeitraum: 01.01.2019 - 31.10.2022

FKZ: 0350056 A - F

Juni 2023

FGH

Herausgeber:
FGH e.V. (Konsortialführung), FKZ: 0350056A
Hochschule für angewandte Wissenschaften Hamburg (HAWH), FKZ: 0350056B
RWTH Aachen University, FKZ: 0350056C
Fraunhofer-Gesellschaft (FhG), FKZ: 0350056D
mit den Instituten
Fraunhofer Institut für Kommunikation, Informationsverarbeitung und Ergonomie (FKIE)
Fraunhofer Institut für Angewandte Informationstechnik (FIT)
Kiwigrid GmbH, FKZ: 0350056E
Mitteldeutsche Netzgesellschaft Strom mbH (Mitnetz Strom), FKZ: 0350056F

Autoren:
Jan Christoph Kahlen (FGH e.V.)
Prof. Wolfgang Renz (HAWH)
Florian Schmidtke (RWTH Aachen)
Immanuel Hacker (FhG)
Andrei Ionita (FhG)
Michael Rademacher (FhG)
Eugen Winter (FhG)
Lars Schwarzelt (Kiwigrid GmbH)
Steve Bahn (Mitnetz Strom)
Dr. Michael Lehmann (Mitnetz Strom)

Anschrift:
Forschungsgemeinschaft für Elektrische Anlagen und Stromwirtschaft e.V. (FGH e.V.)
Voltastraße 19 – 21
68199 Mannheim

Mannheim, im Juni 2023

Inhaltsverzeichnis

I. Kurze Darstellung

I.1 Hintergrund und Aufgabenstellung

Die Energiewende erfordert einen grundlegenden Umbau des Versorgungssystems und stellt Netzbetreiber und Energieversorger vor die Aufgabe, bei einer Vielzahl von volatilen, erneuerbaren Erzeugeranlagen weiterhin einen sicheren Netzbetrieb zu gewährleisten. Um dies in Zukunft erfüllen zu können, ist neben dem Netzausbau die größte Herausforderung, die Stromerzeugung und -nachfrage durch intelligente Steuerung und Anreizschaffung zu flexibilisieren.

Derzeitig sind die Märkte für netzdienliche Flexibilitäten nur begrenzt vorhanden. Sie beschränken sich weitestgehend auf den Einsatz der Regelenergieleistung und des Redispatchs auf Ebene der Übertragungsnetzbetreiber (ÜNB) sowie abschaltbarer Lasten auf Ebene der Verteilnetzbetreiber (VNB). Diese Instrumente sind zukünftig nicht mehr ausreichend, um Last- und Erzeugungsspitzen im Netz auszugleichen und den sicheren Netzbetrieb zu gewährleisten. Der BDEW fordert daher im Smart Grid Ampelkonzept die Nutzung netzdienlicher Flexibilität in der gelben Ampelphase. Die Umsetzung dieses Konzepts bedingt den Zugang von VNB und ÜNB zu netzdienlicher Flexibilität, um sie kontrahieren und im Bedarfsfall steuern zu können. Dies wiederum setzt die effiziente Vermarktung aller netzdienlichen Flexibilitäten von dezentralen Energieanlagen durch dynamische Aggregation, die auch Kleinstanlagen die Teilnahme an Märkten ermöglicht, zu einem Flexibilitätsportfolio eines Aggregators voraus. Diese Entwicklungen spiegelten sich im Laufe des Projektes insbesondere durch die Ausgestaltung der Paragrafen 14a und 14c des Energiewirtschaftsgesetz (EnWG) wider. §14a verpflichtet Netzbetreiber Lieferanten und Endverbrauchern ein reduziertes Netzentgelt zu berechnen, wenn ihnen im Gegenzug die netzdienliche Steuerung der Verbraucher gewährt wird. §14c regelt die marktgestützte Beschaffung solcher Flexibilitätsdienstleistungen im Verteilnetz und legt den Netzbetreibern auf, die notwendige Flexibilität in einem transparenten, diskriminierungsfreien und marktgestützten Verfahren durchzuführen.

Ziel dieses Vorhabens war es daher einen sogenannten „FlexHub“ zu entwickeln, der es ermöglicht alle im Netz angeschlossenen dezentralen Energieanlagen über eine sichere Infrastruktur anzubinden und deren verfügbare Flexibilitäten über einen Marktplatz zur Verfügung zu stellen. Damit soll die Kontrahierung von Flexibilität zwischen den beteiligten Marktrollen, wie z.B. Aggregator, VNB, und ÜNB ermöglicht werden. Der FlexHub ist somit eine verteilte, transparente, dynamische und diskriminierungsfreie Plattform, die als Marktplatz und Steuerungseinheit für intelligente Netze dient.

Konkret wurden Im FlexHub Projekt die folgende wissenschaftlichen/technischen Arbeitsziele verfolgt:

- Definition von Anwendungsfällen für den FlexHub und Ableitung von Geschäfts- und Anreizmodellen
- Testfall- und Testplanerstellung für die Durchführung von Tests des FlexHubs
- Entwicklung nach Security By Design

- Entwicklung eines Datenmodells für den FlexHub
- Konzipierung des FlexHubs und Entwicklung von Proof-of-Concept Demonstratoren
    - hierarchischer, auf Domänen basierender verteilter FlexHub
    - ein Blockchain-basierter FlexHub.
- Bewertung des FlexHub Konzepts aus Sicht der IT-Sicherheit
- Ermittlung der Anforderungen an die Kommunikationstechnologien und Analyse der Skalierbarkeit von IKT
- Entwicklung des Kommunikationsstacks für den FlexHub
- Aufbau und Inbetriebnahme einer Demonstrationsumgebung
- Durchführung von Verifikationsversuchen im Labor
- Durchführung eines Feldversuchs im Netz der MITNETZ
- Ableitung von Handlungsempfehlungen für den Einsatz des FlexHubs

I.2 Voraussetzungen, unter denen das Projekt durchgeführt wurde

Das Vorhaben wurde durch ein interdisziplinäres Konsortium aus Forschungseinrichtungen, Industrie und Anwendern des FlexHub bearbeitet. Die beteiligten Partner haben jeweils spezifische Vorkenntnisse und Ressourcen in das Vorhaben eingebracht:

- FGH:
  Die FGH hat ihr in zahlreichen deutschen und EU-Forschungsprojekten sowie Auftragsarbeiten erworbenes Wissen zum Bereich Smart Grids und dem IEC 61850 Standard eingebracht.

- HAW Hamburg:
  In dieses Projekt bringt die HAWH ihr in zahlreichen Auftragsarbeiten sowie in deutschen (z.B. BMWi-Projekte “Smart Power Hamburg”, „NEW4.0“ Teilprojekt IKT- Algorithmen und Standardisierung, Mitglied SINTEG-AG Standardisierung) und EU-Forschungsprojekten (z.B. EU FP7-ICT-2013-11 “Open System for Energy Services”) erworbenen Erkenntnisse in der Energieinformatik, insbesondere in den Bereichen Architektur, Entwicklung und Simulation verteilter Energiesysteme für zukünftige Energiemärkte, ein.

- RWTH Aachen:
  Das IAEW der RWTH Aachen hat ihr in zahlreichen Forschungsprojekten und Studien erworbenes Wissen eingebracht. Die Infrastruktur des Testzentrums (z.B. Ortsnetzstationen, Wechselrichter) verfügt darüber hinaus über umfangreiche Kommunikationstechnik. Eine flächendeckende Einbindung verschiedenster Informations- und Kommunikationstechnologien (z.B. PLC, Ethernet, Glasfaser, GSM) sowie eine vollständige Netzleitwarte (Server, Datenbank,

Fernwirkanbindung, Switche, Firewalls, Software), wie sie auch bei Verteilungsnetzbetreibern zum Einsatz kommt, stand für das Projekt FlexHub zur Verfügung.

- Fraunhofer FIT und FKIE:
  FIT besitzt langjährige Erfahrung in der Durchführung von Forschungsprojekten auf nationaler und internationaler Ebene und bringt Kompetenzen aus den Forschungsbereichen Kooperationssysteme und Prozessmanagement sowie Digitalisierungskonzepte aus Auftragsprojekten mit Industriepartnern als Vorarbeiten ein. Ferner nutzt FIT die Infrastruktur des in 2016 gegründeten Blockchain-Labs in dem Vorhaben. Das Fraunhofer Institut für Kommunikation, Informationsverarbeitung und Ergonomie FKIE erforscht und entwickelt anwendungsorientierte Sicherheitstechnologien für den zivilen und wehrtechnischen Bereich. Es zählt zu den führenden Instituten für anwendungsorientierte Forschung und praxisnahe Innovation in der Informations- und Kommunikationstechnologie und unterhält eine enge Kooperation mit der RWTH Aachen.

- Kiwigrid GmbH
  Kiwigrids strategische Partnerschaften mit Innogy, EnviaM und Mitnetz streben einen gemeinsamen und wirtschaftlichen Betrieb der Plattform an. Durch Kiwigrids laufende Projekte mit diesen Partnern in der Energiewirtschaft werden wichtige Kompetenzen und Know-How, die für dieses Forschungsprojekt wesentlich sind, eingebracht. Des Weiteren bringt Kiwigrid die IoT Plattform für die Einbindung des FlexHubs in eine Testumgebung, die Hardware für die Anbindung der steuerbaren Anlagen sowie Entwicklungsressourcen und Know-How zur Implementierung als Kernkompetenzen in das Projekt ein.

- MITNETZ Strom
  In das Projekt bringt Mitnetz seine langjährige Erfahrung mit Flexibilitätsmechanismen (Demand Side Management und Eingriffe in die Erzeugung) ein sowie ein Netz, das mit 80 % des Stroms aus EEG-Quellen für den Feldtest in diesem Projekt dient.

I.3 Planung und Ablauf

Das Projekt war – abgesehen vom Projektmanagement (Leitung FGH) - in die folgenden acht Arbeitspakete gegliedert:

- AP 1 Business- und Anreizmodelle für einen FlexHub (Leitung Kiwigrid)
- AP 2 Testfall- und Testplanerstellung (Leitung Kiwigrid)
- AP 3 Informationsflüsse und Datenmodell (Leitung FGH)
- AP 4 Architektur des FlexHubs und Entwicklung technischer Prototypen (Leitung HAWH)
- AP 5 Konzeptentwicklung zur IKT-Anbindung (Leitung FIT)
- AP 6 Implementierung und Entwicklung einer Plattform (Leitung Kiwigrid)

- AP 7 Betrieb des FlexHub und Demonstrationsversuche (Leitung MITNETZ)
- AP 8 Ableitung von Handlungsempfehlungen (Leitung MITNETZ)

Der Arbeitsstrukturplan (Abbildung 1) veranschaulicht die Beteiligung der Projektpartner an den Arbeitspaketen:

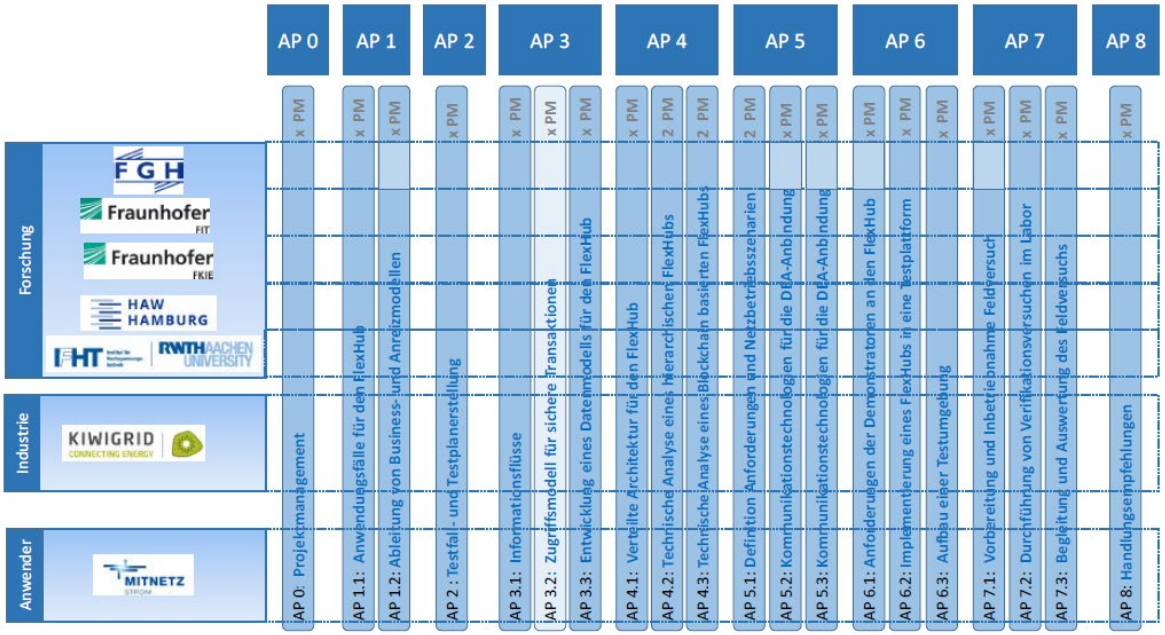


Abbildung 1: Arbeitsstrukturplan

Ursprünglich war die Bearbeitung innerhalb von 36 Monaten geplant. Aufgrund von Verzögerungen insbesondere infolge der Corona-Pandemie wurde die Laufzeit des Vorhabens zweimal um insgesamt 10 Monate verlängert.

Abbildung 2 zeigt die resultierende Zeitplanung des Projektes:

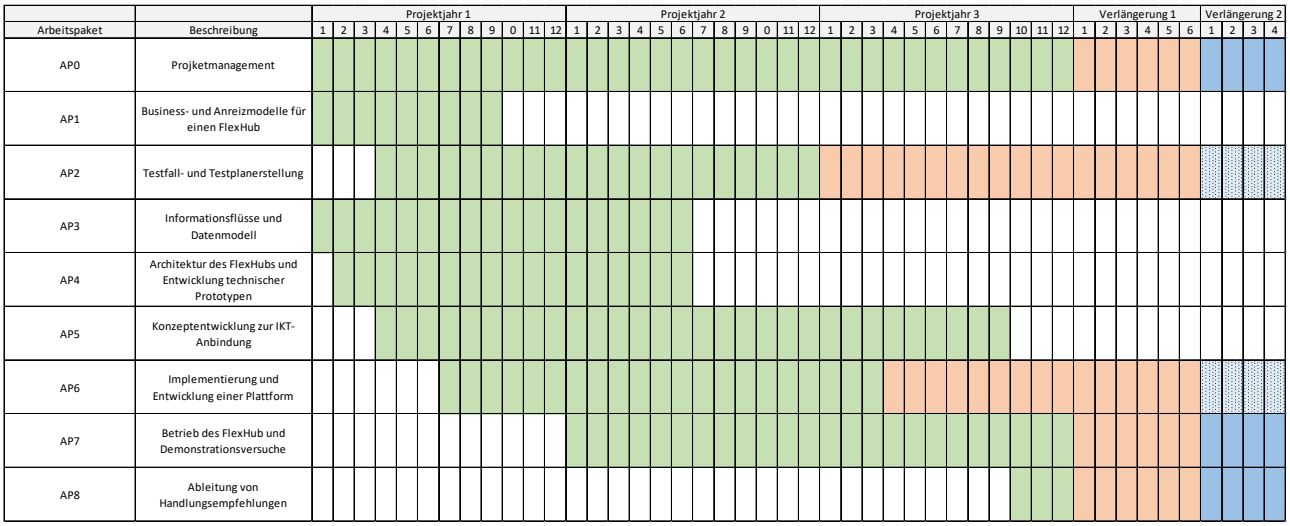


Abbildung 2: Zeitplanung

I.4 Wissenschaftlicher und technischer Stand

Erzeugungsanlagen werden bereits von Aggregatoren vertraglich gebunden, die gepoolte Anlagenleistung als Energieprodukt anbieten. Dabei wird allerdings das volle Anlagenpotential durch einzelne Aggregatoren nicht ausgeschöpft, da die Anlagen trotz freier Kapazitäten nur für eine Dienstleistung vermarktet werden.

Ein offenes System, welches die Livedaten von verteilten Anlagen bereitstellt, ermöglicht eine dynamische Kontaktierung durch Energieakteure und die Ausschöpfung des aktuellen Anlagenpotentials.

Die Schaffung einer Plattform für die marktliche Bereitstellung von flexibilitätsbasierten Energiedienstleistungen wurde als technologischer Grundstein für das FlexHub Projekt u.a. im Open System for Energy Services (OS4ES) Projekt gelegt, das im Rahmen des Framework Programm 7 der Europäischen Union gefördert wurde (Grant Agreement Nr. 619302). In diesem Rahmen wurden erste konzeptionelle Arbeiten an der Technologie vorgenommen. Diese waren inhaltlich jedoch stark an die Weiterentwicklung des IEC 61850 Standards zur Einbindung von DER Systemen in das Energienetz angelehnt. Ausgehend vom Konzept einer Verteilten Registry für servicebasierte Energiemanagementsysteme [3] wurden konzeptionellen Vorarbeiten zur Registry vorangetrieben, verblieben zunächst jedoch im Entwurfsstadium [4]. Es wurde eine Einordung des Konzeptes in das BSI-Schutzprofil durch, sodass bereits ein grundlegendes Verständnis über die sicherheitstechnischen Anforderungen an Energieaustausch- und Energiemarktplattformen wie dem FlexHub und den damit verbundenen Komponenten vorhanden ist [5]. Der Anwendungsprototyp der Registry, welcher grundlegende Verteilungsmechanismen und eine Untermenge der entworfenen Energiedienstleistungen enthielt, wurde vom MMLab und TNO entwickelt. Dieser Prototyp unterstützt neben einer prototypischen IEC 61850 Anbindung auch das REST-Protokoll, welches von den beteiligten Arbeitsgruppen als das derzeit vielversprechendste

Protokoll für den Markteintritt eingeschätzt wurde. Die Registry enthält nach derzeitigem Stand einen Teil der Sicherheitsmechanismen, die im BSI-Schutzprofil gefordert werden. Jedoch sind noch keine Authentifizierungsmechanismen und Mandantenzugriffe realisiert. Zuletzt sind die Funktionalitäten der Suche und Fahrplanvalidierung nur spezifisch auf dem Stand der Labor und Feldtests realisiert und nicht im Zustand eines Produktivsystems.

Die grundlegende Anwendbarkeit der Plattform und der Middleware wurde in Labor- und Feldtests nachgewiesen [6], [7]. Die MMLab Forschungsgruppe untersuchte dabei speziell die Anwendbarkeit in einem Marktumfeld zur Unterstützung von Aggregatoren. In Simulationen und hausinternen Studien konnte gezeigt werden, dass das System konzeptionell in der Lage ist, die Aggregation in Fehlerfällen marktnah zu unterstützen, um somit die Kosten zu senken und die Effizienz des elektrischen Gesamtsystems zu steigern.

Im FlexHub Projekt soll dieses Konzept netzdienlich umgesetzt werden und der Zugriff auf solch eine Datendrehscheibe für Flexibilitäten durch mehrere Aggregatoren in Abstimmung mit dem VNB erfolgen und damit die im OS4ES Projekt lediglich konzipierte Verteilung der Datendrehscheibe in dem FlexHub Projekt implementiert, funktional erweitert und in Labor- und Feldtests getestet werden.

Zurzeit gibt es zwar Softwarelösungen zur Steuerung von Flexibilitäten, wie z.B. die von Kiwigrid zur Steuerung von verteilten Stromerzeugern und Verbrauchern (Wärmepumpen, Speichersysteme und Elektrofahrzeuge) entwickelte Energy IoT Cloud1, doch basieren diese Lösungen meist auf proprietären Schnittstellen. Für die Akzeptanz einer deutschlandweiten Lösung ist jedoch nicht nur entscheidend, den Flexilitätsumfang auf alle DEA zu erweitern, sondern auch etablierte Normen für die Datenmodelle und die Kommunikation zur Modellierung und Aggregation von Flexibilität zu verwenden. Aktuell wird das im laufenden Normungsprozess der IEC 61850 befindliche Konzept der Energiedienste evaluiert (IEC 61850-7-420) und ein Web-basiertes Kommunikationsprotokoll (IEC 61850-8-2) standardisiert. Es wird deshalb im FlexHub Projekt geprüft, inwieweit das Datenmodell für Energiedienste für die in diesem Projekt definierten Anwendungsfälle des FlexHubs verwendet und an welchen Stellen Erweiterungen vorgenommen werden müssen. Weiterhin wird die Eignung des Web-basierten Kommunikationsprotokolls für den Einsatz in FlexHub untersucht und bei Eignung implementiert.

Das Netzflex Projekt von Kiwigrid mit MitNetz hat das Ziel, die technischen Voraussetzungen zu schaffen, damit eine netzdienliche Steuerung von Lasten und Einspeisern durch ein Anreizsystem mittels flexiblen Netzentgelten und direkter Steuerung von Kundenanwendungen in der Mittel- und Niederspannungsebene realisiert werden kann. Das Projekt ist ein wichtiger Schritt in Richtung netzdienliche Steuerung, berücksichtigt aktuell jedoch nicht, dass mit der Registrierung und Speicherung von Anlagenflexibilität in einem FlexHub auch Verbesserungen beim Monitoring und der Steuerung einhergehen und der Zugang zu Flexibilität auch anderen Marktteilnehmern ermöglicht wird, wie es im FlexHub Projekt realisiert werden soll. Dadurch wird ein automatisierbarer Markt zwischen Aggregatoren und DEA ermöglicht. Ein weiteres Forschungsprojekt mit EnviaM (sMobilityCom2) verfolgt das Ziel, einen bundeseinheitlichen, netzdienlichen Anschluss für E-Fahrzeuge mit einer dynamischen Leistungssteuerung und -optimierung am Netzanschluss zu entwickeln. Mit dem Vorhaben soll den VNB ein Instrument zur wirksamen und dauerhaften Steuerung von E-Fahrzeugen zur Verfügung gestellt werden, um

die Integration von Elektrofahrzeugen in das deutsche Energienetz zu vereinfachen und die Netzausbaufolgekosten aufgrund der Elektromobilität zu minimieren. Auch hier kann eine offene Plattform wie das FlexHub Projekt sie vorsieht zukünftig dazu beitragen, Flexibilitäten bereitzustellen sowie markt- und netzdienliche Transaktionen zu ermöglichen und somit die E-Mobilität in den Verteilnetzen zu befähigen. Im Projekt NEW4.0 und in anderen SINTEG-Projekten werden entweder Use-Case spezifische oder sehr allgemeine Plattformen konzipiert und entwickelt, der Flexhub dagegen ertüchtigt das OS4ES-Konzept für in Kapitel 1 genannten zwei Use-Cases, die gegenwärtig ganz oben auf der Agenda stehen und eine schnelle Umsetzung auf Basis eines zugeschnittenen offenen Konzepts ermöglichen.

Neben der Entwicklung einer Plattform und der Konzeption der Datenhaltung stellt insbesondere auch die informationstechnische Anbindung von Erzeugungsanlagen eine Herausforderung hinsichtlich der Verfügbarkeit, Daten- und IT-Sicherheit dar, die bei der Konzeptionierung Berücksichtigung finden muss. Mit dem „Gesetz zur Digitalisierung der Energiewende“ ist einer der zentralen Bausteine für die zukünftig sektorenübergreifende Vernetzung der am Energiemarkt beteiligten Akteure auf den Weg gebracht worden. Kern des Paketes bildet das Messstellenbetriebsgesetz (MsbG) mit Neuregelungen zum Messstellenbetrieb und der Datenkommunikation in intelligenten Energienetzen. Im Fokus steht dabei die Verbindung zwischen Netz und Kundenanlage, an dem Stromverbrauch bzw. -einspeisung gemessen wird. Bisher ist nicht ausreichend untersucht, inwieweit die Infrastruktur des intelligenten Messsystems (iMSys) geeignet ist, die Anbindung und insbesondere die Steuerung von dezentralen Energieanlagen über die vom Gesetzgeber geforderte Infrastruktur zu übernehmen. Ziel dieses Projekts ist es daher, die Erzeugungsanlagen über die iMSys Infrastruktur anzubinden und zu steuern als auch in Labor- und Feldtests zu erproben.

Das Projekt Flexhub verfolgt weiterhin das Ziel, ein Konzept für die IT-Sicherheit von Plattformen zu entwickeln und zu demonstrieren, welches das physikalische Risikopotential im Angriffsfall bereits während der Entwicklung berücksichtigt. Der Einsatz der Blockchain- Technologie [3] [4] als innovative Alternativlösung stellt hier eine vielversprechende Möglichkeit zur Erhöhung der Sicherheit und Vertrauenswürdigkeit dar. Im Anwendungsfeld Smart Grids werden bislang meist lokale Lösungen betrachtet und nicht wie im FlexHub-Projekt eine umfassende Modellierung von Energieerzeugungsanlagen einschließlich der Kommunikation zwischen den Smart Grid Akteuren. Das Konsortium verfügt über Erfahrungen mit ausreichend reifen Blockchain-Infrastrukturen wie Ethereum und Fabric aus Hyperledger. Die beiden Konzepte einer hierarchischen Plattform und einer Blockchainbasierten Plattform sind konzeptionell und praktisch durch die Evaluierung von geeigneten Anwendungsfällen und der Erstellung technischer Demonstratoren zu untersuchen. Damit sollen funktionale, aber auch nichtfunktionale Eigenschaften wie Skalierbarkeit und Robustheit, Vertraulichkeit und Integrität der Daten sowie Aspekte des Deployments und Betriebs bei Unternehmen bewertet werden.

Zusammengefasst möchte FlexHub die hier ausgeführten Kompetenzen einer offenen Plattform für Energiedienste zusammenführen und weiterentwickeln. Die Neuheit des FlexHubs kombiniert die Anforderungen des BDEW Ampelkonzepts und der verbesserten Bilanzkreistreue mit dem Ansatz einer offenen skalierbaren und diskriminierungsfreien Plattform, die dynamische Anlagendaten bereitstellt und den verschiedenen Akteuren am Energiemarkt den Zugriff und die Kontrahierung von Flexibilität

verteilter Anlagen über eine IT sicherheitskonforme Infrastruktur ermöglicht. Im FlexHub-Projekt fungiert der VNB als neutraler „Markt-Vermittler, wie dies von der Smart Grid Task Force 2012/2013 als eine der Möglichkeiten für den Betrieb von Flexibilitätsdatenregistern aufgeführt wird.

Auf technischer Ebene setzt sich gegenwärtig die sogenannte Containerisierung, insbesondere zur Realisierung von komplexen verteilten Systemen, immer weiter durch [8]. Containerisierung ist das Verpacken von Softwarecode in Pakete (Paketierung), die alle erforderlichen Komponenten wie Libraries, Frameworks und andere Abhängigkeiten enthalten und in ihrem eigenen Container isoliert sind. Auf diese Weise kann die Software oder Anwendung innerhalb ihres Containers in jede Umgebung verschoben und konsistent auf jeder Infrastruktur ausgeführt werden, unabhängig von der Umgebung oder dem Betriebssystem der Infrastruktur. Es handelt sich dabei um eine voll funktionsfähige tragbare Rechenumgebung. Dies ist insbesondere im Cloud Computing Umfeld von Bedeutung. Die Idee einer solchen Prozessisolation erreichte wirkliche Relevanz, als die Firma Docker 2013 die Docker Engine einführte, die zum Defacto Standard für die Verwendung von Containern wurde [9].

Im Gegensatz zu virtuellen Maschinen haben Container den Vorteil, dass sie schlanker und damit leichter portierbar sind, weil sie den Betriebssystem-Kernel des Host-Computers gemeinsam nutzen. Dadurch entfällt die Notwendigkeit eines separaten Betriebssystems für jeden Container, und die Anwendung kann auf jeder Infrastruktur ausgeführt werden.

Der Container-Ansatz eignet sich insbesondere gut für die Realisierung von Microservice-Architekturen, bei denen die Teile einer (komplexen) Anwendung in kleine, spezialisierte Services aufgeteilt wird [10]. Jeder Service hat damit eine klar definierte Aufgabe und interagiert über klar definierte Schnittstellen mit den anderen Services. Somit kann ein einzelner Service relativ einfach durch einen anderen Service ersetzt werden und Änderungen an einem Service ziehen keine Änderungen an anderen Services mit sich. Das Kapseln dieser Services in Container bildet diese Struktur entsprechend logisch auf die Infrastruktur ab.

Behavior Driven Development ist eine Technik der agilen Softwareentwicklung. Dabei werden während der Anforderungsanalyse die Aufgaben, Ziele und Ergebnisse der Software in einer bestimmten Textform festgehalten, die später als automatisierte Tests ausgeführt werden kann. Damit kann die Software auf ihre korrekte Implementierung getestet werden. Die Softwareanforderungen werden dabei meist in Wenn-dann-Sätzen verfasst. Damit soll der Übergang zwischen der Sprache der Definition der fachlichen Anforderungen und der Programmiersprache, mittels derer die Anforderungen umgesetzt werden, erleichtert werden [11]. Beim Behavior Driven Development werden die Anforderungen an die Software mittels Beispiele, sogenannten Szenarien beschrieben. Üblicherweise wird für die Beschreibung dieser Szenarien ein bestimmtes Format vorgegeben, damit später die automatisierte Überprüfung der Szenarien einfach umzusetzen ist. Eines dieser Formate ist die Beschreibungssprache *Gherkin* [12]. Diese Sprache gibt es sowohl mit englischen Schlüsselwörtern (Given, When, Then, And, …), deutschen (Gegeben, Wenn, Dann, Und, …) und in weiteren Sprachen.

Entsprechend der Use-Case-Methodik des Architekturframeworks „Smart Grid Architecture Model“ (SGAM) [13] hat sich die Nutzung des in IEC62559-2 standardisierten Use-Case Template etabliert [14].

In SGAM werden ausgehend von der Identifikation des Geschäftsmodells High-Level Use-Cases identifiziert, die im Falle des FlexHub-Projekts die Bereitstellung und Nutzung flexibler Ladekapazitäten von Elektromobilen umfassen (Geschäftsebene). Auf der Funktionsebene werden dann System-Use-Cases identifiziert und im o.g. Use-Case Template niedergelegt. In der Informationsschicht werden die Informationsmodelle identifiziert, welche auch die Identifikation von Standardisierungsbedarfen ermöglichen. Die Kommunikationsschicht beschreibt den Einsatz der erforderlichen Kommunikationsstandards. Schließlich umfasst die Komponentenschicht die physische und logische Beschreibung der beteiligten Einrichtungen, also IoT, Steuerbox, Smart Meter, Cloudsysteme usw.

Risk-assesssment nichtfunktionaler Risiken wie Systemsicherheit und Datenprivatheit wird im Anschluss an die SGAM-Modellierung in der Designphase und bei der Planung der Deployments durchgeführt [15]. Funktionale Risiken, die typischerweise durch seltene Ereignisse in der Systemumgebung also z.B. durch unerwartet koordiniertes Nutzerverhalten, Wetter- oder Marktgeschehen ausgelöst werden können, sollten allerdings im früheren Stadium des Requirementsengineering identifiziert werden und führen typischerweise zu weiterem Untersuchungs- und Handlungsbdarf [16].

Technisch wurde sowohl an Plattform- als auch damalige SMGW Entwicklungsstände der Kiwigrid angeknüpft. Hierbei handelt es sich um eine modulare serviceorientierte Cloud-Plattform. Im Rahmen des Flexhub Vorhabens wurde diese um die Middleware zu Nemo Spotmarket und der Bereitstellung der netz- und marktdienlichen Optimierung für Elektrofahrzeuge erweitert. Die Flexhub Komponenten sind nunmehr ein integraler Bestandteil dieser Kiwigrid-eigenen IoT Plattform geworden.

Arbeiten am SMGW wurden zum Ende der Projektlaufzeit seitens Kiwigrid, aus diversen Gründen, z.B. einer ständigen Verschiebung von BSI Direktiven und fehlender Anreize, wie z.B. Mehrwertdienste für Endkunden, eingestellt. Auch im Projekt selbst erwiesen sich ein Vorgehen und die geplante Infrastruktur mit SMGW als unvorteilhaft und nicht in die Praxis (Feldtest) übertragbar. Insofern wurden SMGW-relevante Arbeitspakete auch innerhalb des Projektes laborseitig ausgelagert.

Im Plattformbereich arbeitet die Kiwigrid GmbH sowohl nach DIN EN ISO 9001:2015 als auch DIN EN ISO/IEC 27001:2017 und ist entsprechend zertifiziert für die

“Entwicklung und Betrieb einer Energie-Service-Plattform bestehend aus einem Edge-Betriebssystem für Energy Manager und Cloud Services zur 3rd Party App-Entwicklung (PaaS) sowie von White-Label-Softwareprodukten (SaaS) für erneuerbare Energien und E-Mobilität. Vertrieb und Beratung zum Produktportfolio sowie Durchführung von Kundenprojekten.”

Alle notwendigen Informationen zu den Arbeitspaketen werden in etablierten Software-Tools, wie Jira und Confluence erarbeitet & dokumentiert. Hierbei findet bei Kiwigrid vor Allem die agile Softwareentwicklungsmethodik Verwendung, wobei Agile Ansätze sich auf Teile der Softwareentwicklung beziehen können oder auf den gesamten Softwareentwicklungsprozess. Das Ziel dabei ist, den Entwicklungsprozess flexibler und schlanker zu gestalten, als das bei klassischen Vorgehensmodellen der Fall ist. In diesem Vorhaben war ein agiler Ansatz, bedingt durch die Vorgaben eines klassischen Forschungs- und Konsortialprojektes nur innerhalb der Entwicklung selbst möglich. Mit dem

Konsortialpartner MITNETZ STROM wurde die agile Methodik über den gesamten Projektprozess hinweg aktiv gelebt, vom Anforderungsmanagement über die Organisation der Akzeptanzkriterien und Use Cases bis hin zur Entwicklung und schlussendlich zum Test.

I.5 Zusammenarbeit mit anderen Stellen

Das Projektkonsortium besteht aus den geförderten Partnern

- FGH (Forschungsgemeinschaft für Elektrische Anlagen und Stromwirtschaft e.V.)
- Hochschule für Angewandte Wissenschaften Hamburg (HAWH)
- RWTH Aachen University (RWTH), IAEW
- Fraunhofer-Institut für Angewandte Informationstechnik (FIT)
- Fraunhofer-Institut für Kommunikation, Informationsverarbeitung und Ergonomie (FKIE)
- Kiwigrid GmbH
- MITNETZ STROM

Darüber hinaus wurde während der Projektlaufzeit die EnergieDock GmbH eingebunden. Die EnergieDock GmbH wurde während der Projektlaufzeit im April 2020 von drei Wissenschaftlern der HAW Hamburg gegründet. Die drei Gründer haben vorher mehrere Jahre an der HAW Hamburg im energiewirtschaftlichen Umfeld zu Energiemärkten und die Integration flexibler Stromverbraucher ins Verteilnetz geforscht. Dabei haben sie auch die erste Projekthälfte des Flexhub-Projekts an der HAW Hamburg bestritten und waren maßgeblich an der Erstellung der Anwendungsfälle, Datenmodelle und Informationsflüsse der zu entwickelnden Plattform beteiligt und haben einen Prototyp des Flexibilitätenregisters mit Marktfunktion implementiert. Beginnend ab dem 01.09.2020 war die EnergieDock GmbH im Unterauftrag der HAW Hamburg in das Projekt eingebunden.

Als assoziierter Partner agierte im FlexHub Projekt die Verteilnetz Plauen GmbH. Sie ist als regionaler Verteilnetzbetreiber für Planung, Betrieb und Vermarktung des Elektrizitätsverteilnetzes in der Region Plauen verantwortlich und versorgt auf einer Fläche von 21 km² 65.201 Einwohner mit elektrischer Energie. Die Verteilnetz Plauen GmbH ist eine 100-prozentige Tochter der envia Mitteldeutsche Energie AG und hat ihren Geschäftssitz in Plauen. Das Netz von Plauen Netz hat dieselben IT-Systeme wie MITNETZ im Einsatz und ist dem MITNETZ Netz unterlagert.

II. Eingehende Darstellung

Die eingehende Darstellung gibt eine detaillierte Übersicht der entwickelten Verfahren, Tools und Ergebnisse der durchgeführten Simulationen und Messungen, welche abschließenden mit den angestrebten Zielen verglichen werden. Der Aufbau der eingehenden Darstellung erfolgt dabei anhand der bereits vorgestellten Arbeitspakete. Abschließend werden die erzielten Ergebnisse zusammengefasst und mit denen der Gesamtvorhabenbeschreibung verglichen.

II.1 Verwendung der Zuwendung, erzielte Ergebnisse und Gegenüberstellung mit den vorgegebenen Zielen

II.1.1 Arbeitspaket 1: Business- und Anreizmodelle für einen FlexHub

AP1 fokussiert sich auf die Entwicklung von Geschäfts- und Anreizmodellen für den FlexHub. Dazu werden in AP 1.1 Anwendungsfälle identifiziert und in AP 1.2 Geschäftsmodelle für die einzelnen FlexHub Akteure abgeleitet und im Business Model Canvas visualisiert. Ziel ist es damit den Nutzen für die einzelnen Akteure aufzuzeigen und schlussendlich Anreizmodelle für die Geschäftsmodelle auszuarbeiten.

Im Rahmen des AP1.1 zur Entwicklung der Anwendungsfälle für den FlexHub wurden die Rollen aller beteiligten Akteure definiert und drei verschiedene Nutzungsszenarien mithilfe von Prozessdiagrammen und Informationsflüssen (siehe AP3.1) im IEC 6259-2-Template definiert.

Auf Basis der erstellen Anwendungsfälle konnten die Projektpartner funktionale Anforderungen an das zu entwickelnde Flexibilitätenregister definieren. Die Beschreibung der drei Anwendungsfälle im IEC 6259-2-Template findet sich im Anhang dieses Berichts (A1 - A3). Diese Beschreibungen bilden wiederum die Grundlage für eine erste Vorbetrachtung welche IT-Sicherheitsanforderungen sich aus den Nutzungsszenarien ergeben (z.B. Vertraulichkeit gemeldeter System-Live-Daten). Ferner werden die nötige IKT-Anbindung und das Potential der Blockchain identifiziert.

II.1.1.1 AP 1.1 Anwendungsfälle für den FlexHub

Zu den konkreten Zielen für AP1.1 gehörten:

1. Identifikation von Nutzungsszenarien des FlexHubs (potenzielle Nutzer und deren Anforderungen)
2. Anwendungsfälle bzgl. Blockchain-Relevanz prüfen
3. Ableitung von IT-Sicherheitsanforderungen aus den Anwendungsfällen und den Anforderungen an die Kommunikationsanbindung.

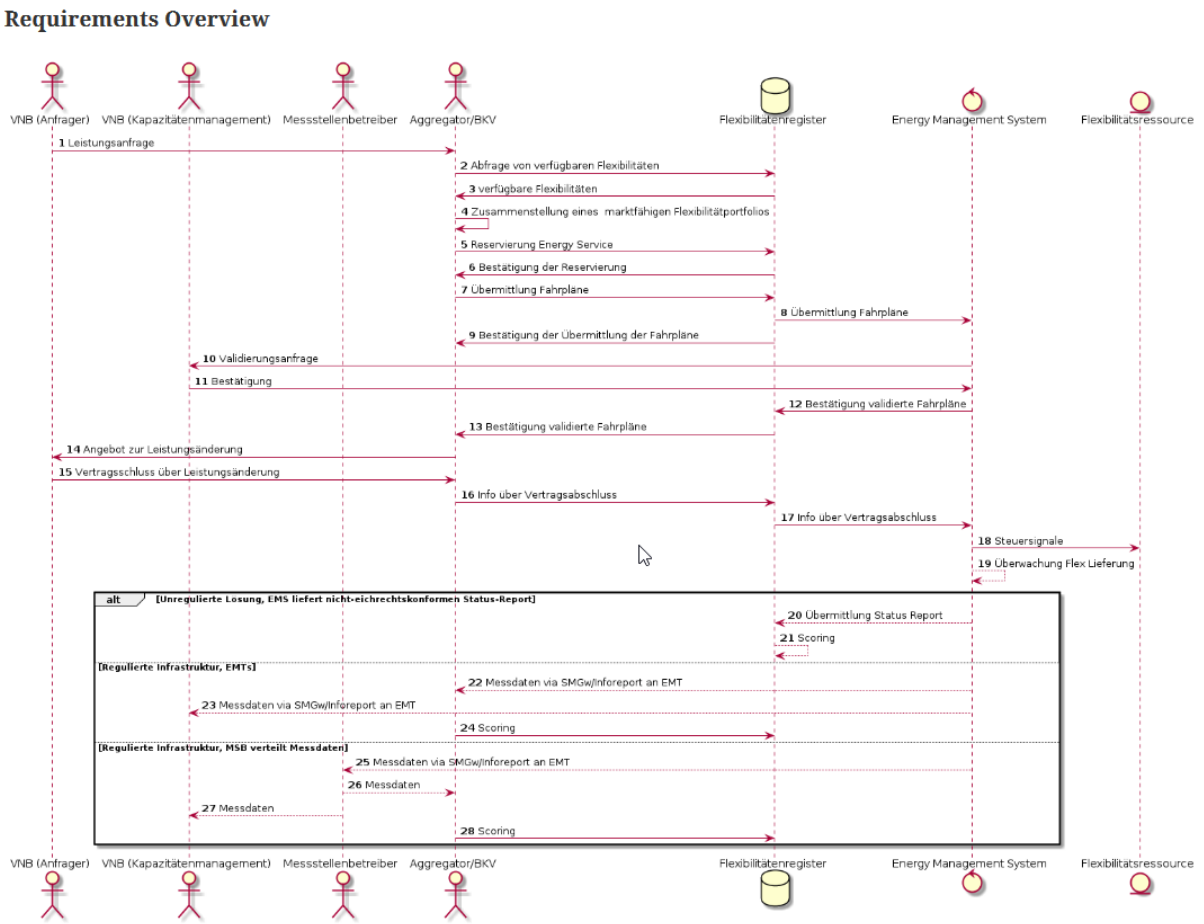
Ergebnis zu Ziel 1: Die in Zusammenarbeit mit den anderen Projekt-Partnern definierten Anwendungsfälle lassen sich wie folgt beschreiben:

1. Netzengpassmanagement: Um drohende Netzengpässe zu vermeiden, werden flexible Stromverbraucher mit signifikanter Bezugsleistung genutzt, um eine Umverteilung des

Leistungsbezuges zu erreichen. Im Verteilnetz kommt es durch hohe Einspeisungen auf Grund hoher Solarstrahlung bzw. starkem Wind zu Engpässen. Der VNB reagiert aktuell mit Netzsicherheitsmaßnahmen (NSM), bei denen dezentrale Erzeugungsanlagen runtergeregelt oder abgeschaltet werden. Die betroffenen Anlagenbetreiber erhalten eine Härtefallentschädigung. Die Nutzung von flexiblen Bezugskunden bietet einen alternativen Ansatz, einspeisebedingte Netzengpässe und somit auch Entschädigungszahlungen an die Anlagenbetreiber zu vermeiden.

2. Flex On Demand: Ein Aggregator bietet Flexibilitäten von DER Systemen in einem Flexibilitätsmarkt an. Diese Flexibilitäten können von Flexibilitätsanfragern gesucht und gebucht werden. Ein VNB kann diese Flexibilitäten z.B. zur Vermeidung von Netzengpässen verwenden. Ein Aggregator stellt unterschiedliche Flexibilitäten (Verbrauch und Erzeugung) von DER Systemen in einer Marktplattform (Flexiblitätenregister) ein. Die Flexibilitäten des DER Systems werden dabei (beispielsweise in der Domäne Customer Premises) über einen lokales DMS (DER Management System) ertüchtigt (Kommunikation/Steuerung). Flexibilitätsanfrager können nach Flexibilitäten suchen, sodass z.B. ein VNB flexible Verbraucher zum Netzengpassmanagement suchen kann. Der Flex Anfrager führt eine lokale Optimierung und Einsatzplanung auf Basis der gefundenen Flexibilitäten durch und erstellt daraus eine Buchung von n Flexibilitäten. Buchungen werden von einem VNB als Kapazitätsmanager validiert, damit die Netzrestriktionen eingehalten werden. Flex Anfrager und Aggregator werden über den Markt über das Ergebnis der Validierung informiert. Wurde die Buchung validiert, kann der Flex Anfrager Fahrpläne an den Markt senden, dieser validiert und authentifiziert die Fahrpläne und leitet sie an den Aggregator weiter, der die Anlage steuert. Das DER System sendet Messdaten über den tatsächlichen Leistungsabruf an einen Messstellenbetreiber. Dieser leitet die Ergebnisse an den Flex Markt weiter. Hier werden sie gespeichert und an den Aggregator, den Flexibilitätsanfrager und den VNB Kapazitätsmanager geschickt. Der Flex Anfrager kann somit die Plattform nutzen, um ein Monitoring der Maßnahme vorzunehmen.

- **Anwendungsfälle 1 und 2: Netzengpassmanagement**
- Der Standard-Ablauf dieser Anwendungsfalls ist der Abbildung 3 zu entnehmen



- *Abbildung 3: AF1 "Geradeausweg"*

- Im Falle einer Zurückweisung der Validierung wurde ein Fehlerfall modelliert. Der Fehlerfall beschreibt den Zustand, bei dem der VNB Kapa nach der Validierung der Fahrpläne deren Ausführung durch die Flexibilitätsressourcen aufgrund von Kapazitätsproblemen, die bei der Umsetzung der Fahrpläne im Netz auftreten würden, verweigert (vgl. Abbildung 4).

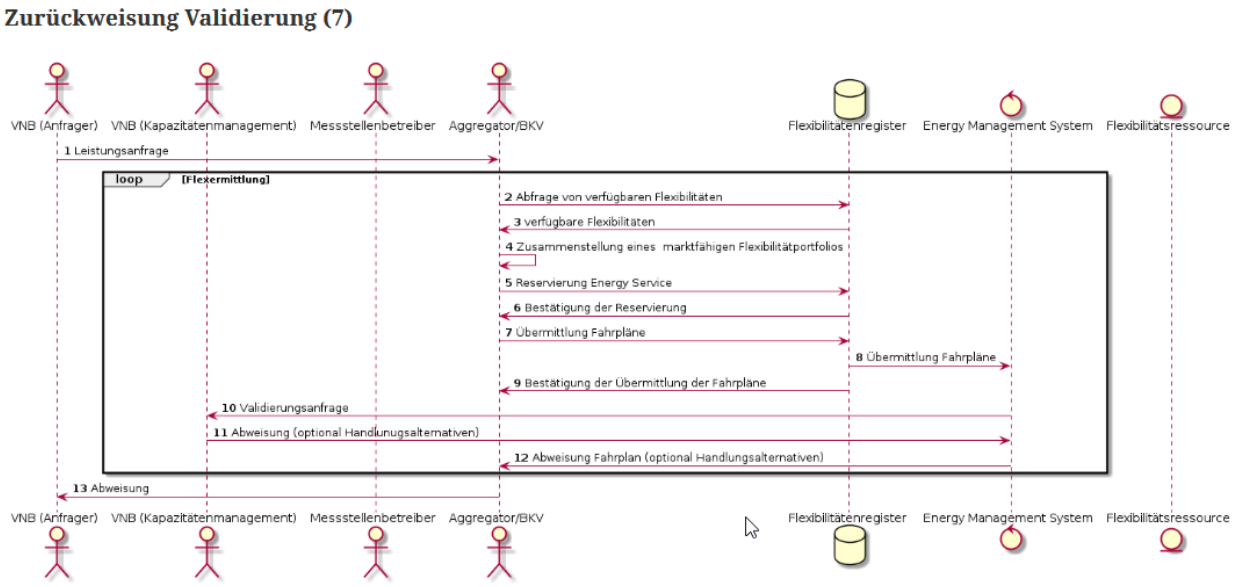


Abbildung 4: AF2 Zurückweisung Validierung

Bei einem Fehler während des Vertragsabschlusses nimmt der Ablauf einen leicht veränderten Weg. Dieser Fehlerfall beschreibt den Zustand, bei dem der VNB Anfrager das Angebot des Aggregators zur Leistungsänderung ablehnt (vgl. Abbildung 5).

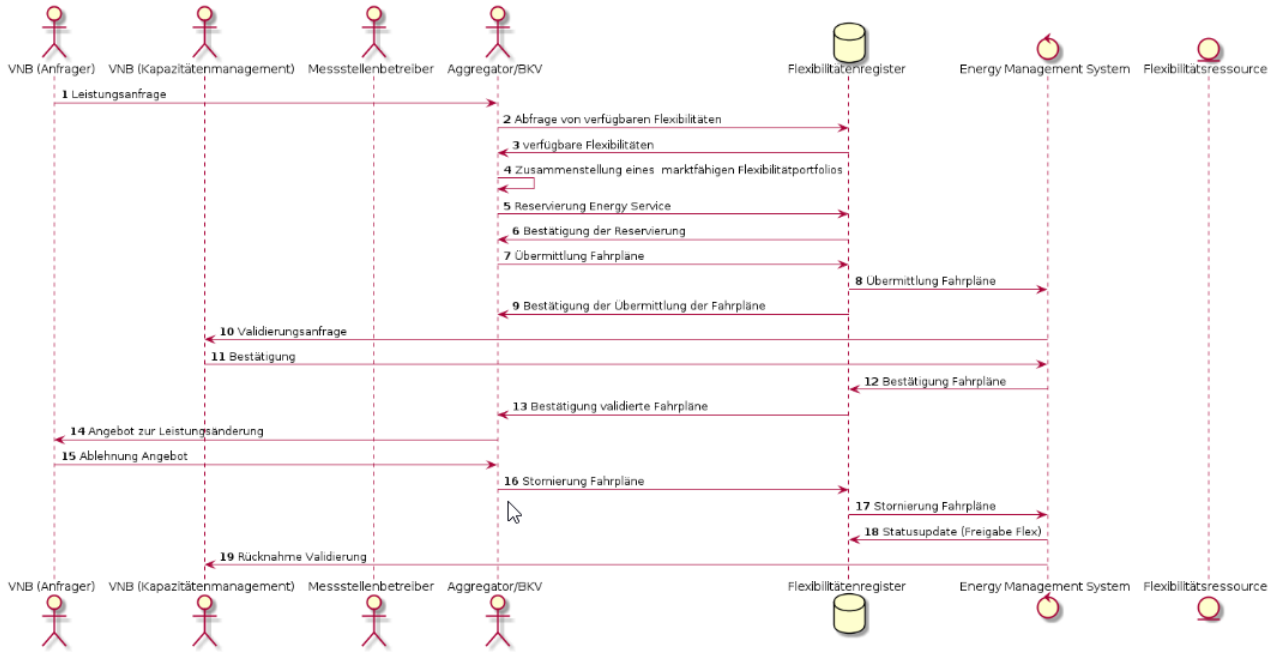


Abbildung 5: Fehlerfall Zurückweisung Vertragsabschluss

Die in den Sequenzdiagrammen vorkommenden Akteure sind in der Tabelle 1 gelistet.

Actor Name	Actor Type	Actor Description	Further information specific to this Use Case
Verteilnetzbetreiber (VNB) (Anfrager)	Rolle	Ein Betreiber, der ein oder mehrere Netze betreibt.	Die Rolle des VNB ist unterteilt in einen anfragenden VNB und den für das Kapazitätenmanagement verantwortlichen VNB
Verteilnetzbetreiber (VNB) (Kapazitätenmanagement)	Rolle	Ein Betreiber, der ein oder mehrere Netze betreibt.	
Aggregator	Rolle	A party that aggregates resources for usage by a service provider for energy market services.	Ein Unternehmen, das marktbezogene Informationen bereitstellt, die aus den Zahlen verschiedener Marktteilnehmer zusammengestellt wurden.
Bilanzkreisverantwortlicher (BKV)	Rolle	A party that has a contract proving financial security and identifying balance responsibility with the Imbalance Settlement Responsible of the Market Balance Area entitling the party to operate in the market. This is the only role allowing a party to nominate energy on a wholesale level.	Der BKV fasst alle bei ihm registrierten Einspeise- und Entnahmestellen zusammen, bilanziert und saldiert diese.
Energy Management System (EMS)	Rolle		Stellt Informationen zu lokal verfügbaren Flexibilitätsressourcen zur Verfügung und lässt eine Steuerung und Optimierung (mittels Fahrpläne) zu.
Flexibilitätenregister	Rolle		Zentrales Register für verfügbare Flexibilitätsressourcen

Messstellenbetreiber (MSB)	Rolle	Ein Betreiber, der für die Installation, Wartung, Prüfung, Zertifizierung und Außerbetriebnahme physikalischer Zähler verantwortlich ist.	
Flexibilitätsressource	Rolle		Eine Flexibilitätsressource (z.B. PV-Anlage mit Batterie oder ein Elektroauto) stellt eine Flexibilität für das Netz zur Verfügung und erbring dadurch eine Dienstleistung für das Energiesystem. So kann sie auf ein externes Signal hin ihr Einspeise- oder Verbrauchsverhalten verändern.

Tabelle 1: Auflistung der Akteure und deren Funktion bei AF1

Anwendungsfall 3: Flex on Demand

Die verschiedenen Abläufe dieses Anwendungsfalls sind der Abbildung 6 (Standard-Ablauf) und der Abbildung 7 (Fehlerfall) zu entnehmen.

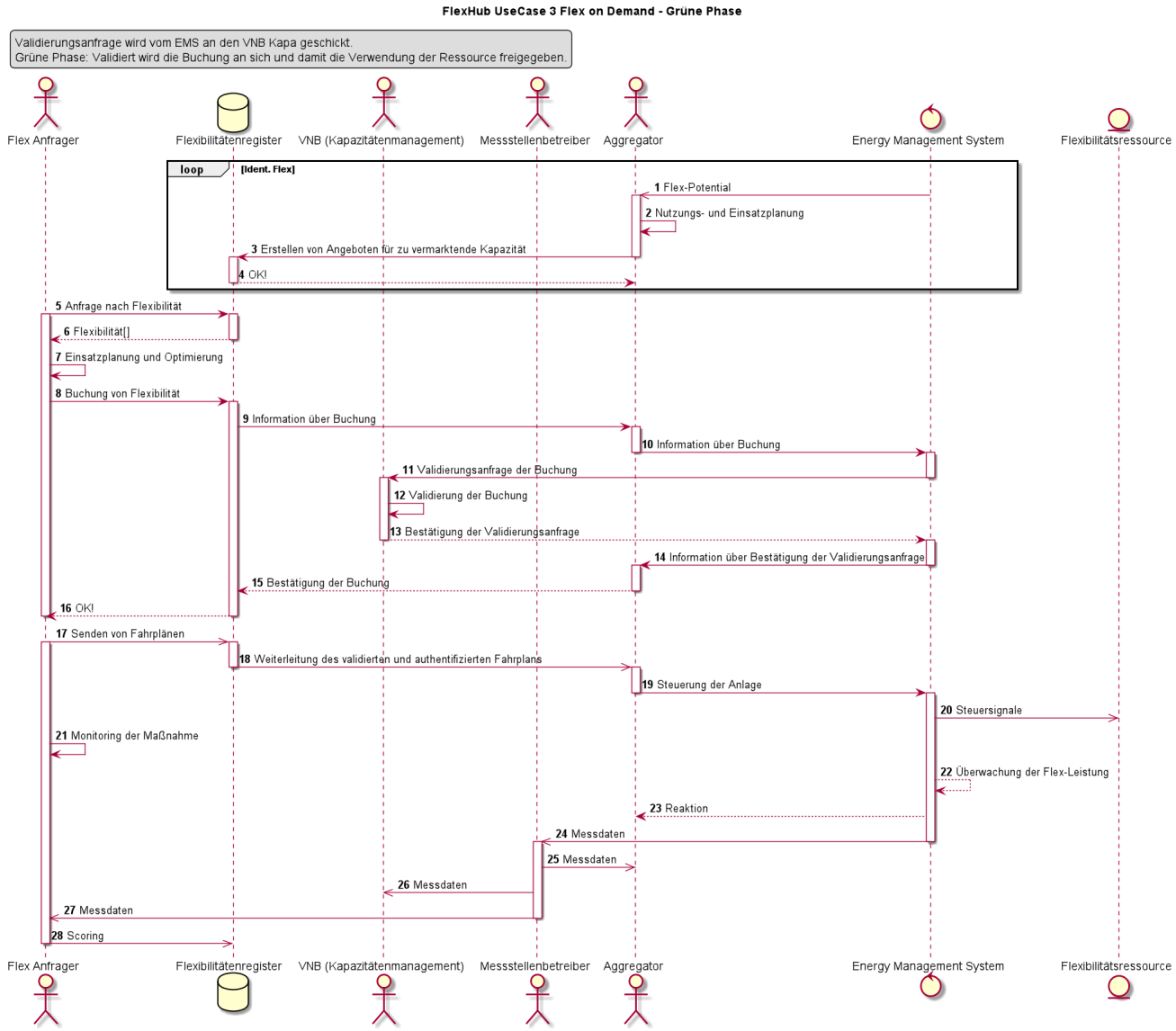


Abbildung 6: Flex on Demand Sequenzdiagramm – Vorwärtsweg, die Buchung wird Fahrplan-unabhängig bestätigt

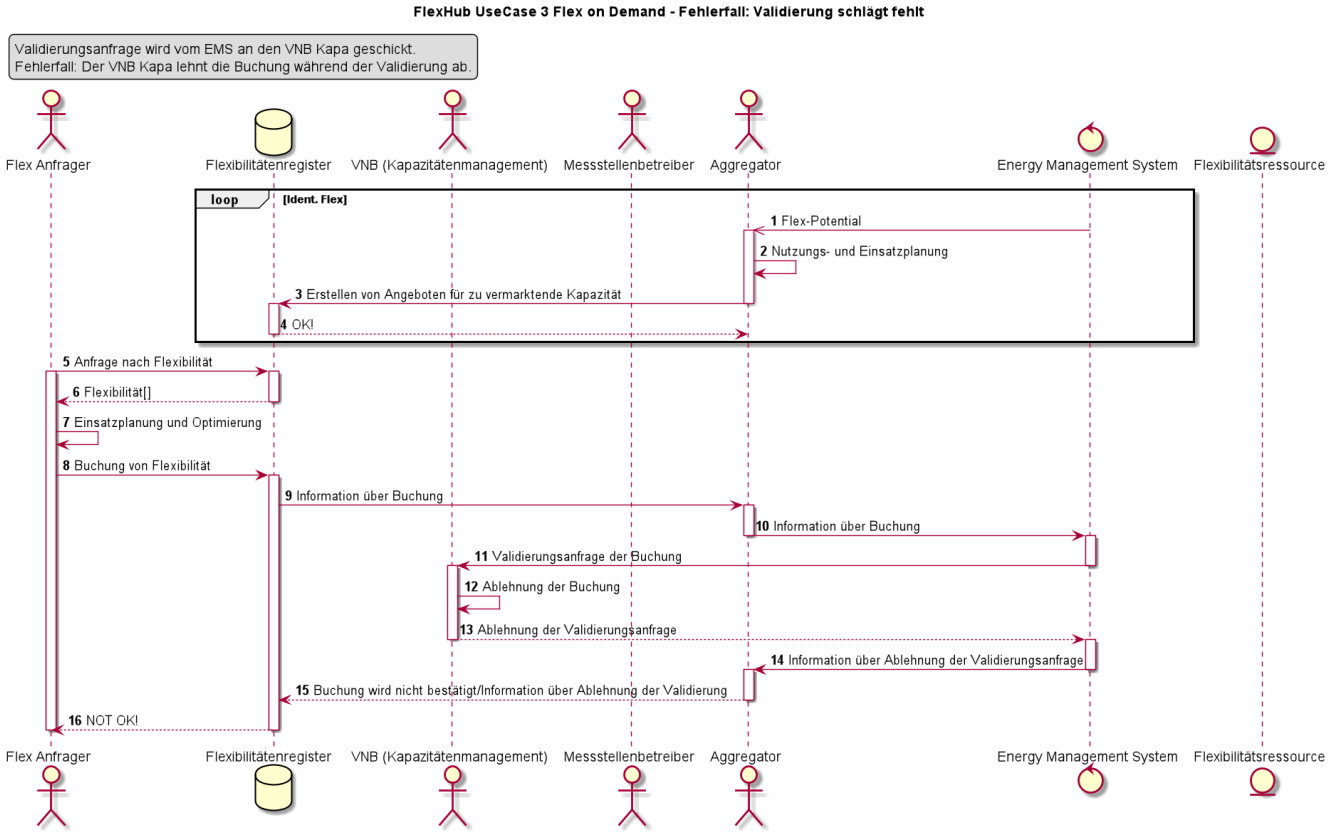


Abbildung 7: Flex on Demand – Fehlerfall, Validierungsabfrage wird abgelehnt

In Abbildung 8 ist ein Spezialfall für die Gelbe Ampelphase im Stromnetz geschildert. In diesem Fall gelten erhöhte Netzsicherungsanforderungen, die bislang mit einem freien und offenen Marktbetrieb nicht vereinbar waren. Durch eine feingranulare Kontrolle kann aber ein Marktbetrieb aufrechterhalten werden. Der VNB kontrolliert nun nicht nur die Buchung mit den intendierten Fahrplänen, oder die allgemeine Zulässigkeit nach Netzanschluss-Konditionen, sondern kontrolliert in einem zweiten Schritt auch den konkreten Fahrplan, sowie etwaige Änderungen an den Fahrplänen, ob diese mit dem Netzzustand vereinbar sind. Der Aggregator erhält somit ein detailliertes Feedback über die Zulässigkeit des durch den Flex-Anfrager angefragten und gebuchten Leistungsabrufs im Sinne der Netzstabilität.

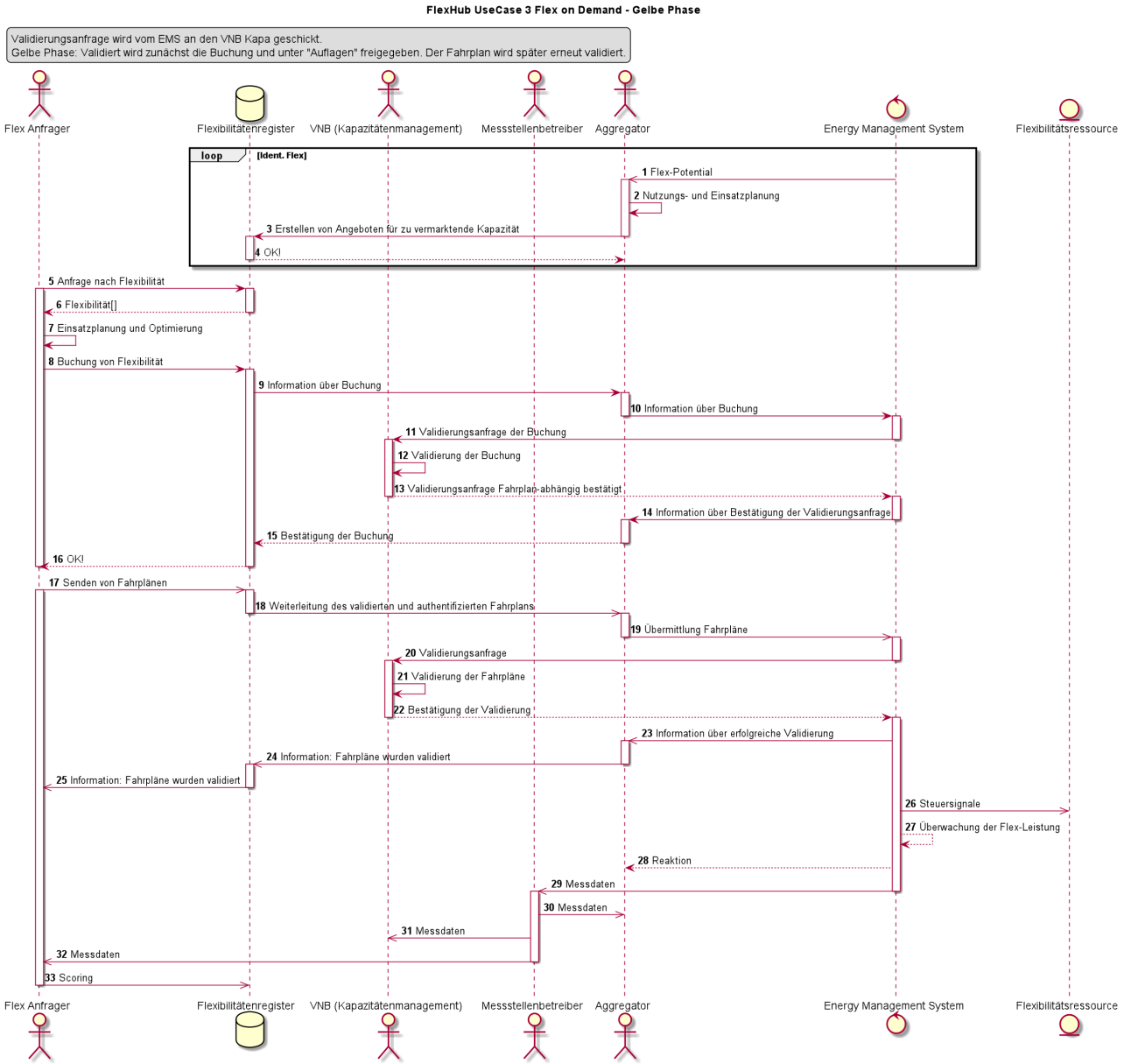


Abbildung 8: Flex on Demand - Gelbe Phase

Die in den Sequenzdiagrammen vorkommenden Akteure sind in der Tabelle 2 gelistet.

Actor Name	**Actor Type**	**Actor Description**	**Further information specific to this Use Case**
Verteilnetzbetreiber (VNB) (Anfrager)	Rolle (System Operator)	Ein Betreiber, der ein oder mehrere Netze betreibt.	Die Rolle des VNB ist unterteilt in einen anfragenden VNB (Anfrager) und den für

Verteilnetzbetreiber (VNB) (Kapazitäten-management)	Rolle (Nomination validator)	Ein Betreiber, der ein oder mehrere Netze betreibt.	das Kapazitätenmanagement verantwortlichen VNB (Kapa)
Aggregator	Rolle (Resource Aggregator)	Eine Partei, die Ressourcen zur Nutzung durch einen Service Provider für Energiemarktdienste aggregiert. Synonym zum Bilanzkreisverantwortlichen.	Ein Unternehmen, das marktbezogene Informationen bereitstellt, die aus den Zahlen verschiedener Marktteilnehmer zusammengestellt wurden.
Energiemanagement-system (EMS)	Anwendung	Energiemanagementsystem beim Aggregator/Bilanzkreisverantwortlichen.	
Bilanzkreisverantwortlicher (BKV)	Rolle (Balance Responsible Party, Resource Aggregator, Trader, Consumption and Production Responsible Party, Resource Provider)	Eine Partei, die einen Vertrag hat, der finanzielle Sicherheit bietet und die Bilanzierungsverantwortung mit der Verrechnungsstelle (Imbalance Settlement Responsible) des zu bilanzierenden Gebiets festlegt, in dem die Partei berechtigt ist, im Markt zu agieren.	Der BKV fasst alle bei ihm registrierten Einspeise- und Entnahmestellen zusammen, bilanziert und saldiert diese.
DER System	Rolle (Party Connected to the Grid)	Eine Entität, die an einem Netzanschlusspunkt vertraglich das Recht hat, Energie einzuspeisen oder zu beziehen.	Eine Sammlung von DER-Einheiten
DER Management System	Anwendung		z.B. HEMS von Kiwigrid
Home Energy Management System (HEMS)	Anwendung	Ein Home Energy Management System ist eine Implementation eines DER	Stellt Informationen zu lokal verfügbaren Flexibilitätsressourcen zur Verfügung und lässt eine Steuerung und

		Management Systems für Haushalte.	Optimierung (mittels Fahrpläne) zu.
Flex-Register	Rolle (Market Operator)	Eine Partei, die eine Dienstleistung anbietet, bei der angebotene Energie mit nachgefragter Energie abgeglichen wird.	Zentrales Register für verfügbare Flexibilitäten, Informations- und Handelssystem.
Messstellenbetreiber (MSB)	Rolle (Meter Operator)	Ein Betreiber, der für die Installation, Wartung, Prüfung, Zertifizierung und Außerbetriebnahme physikalischer Zähler verantwortlich ist.	
Flexibilitätsressource	Rolle (Party Connected to the Grid)	s. DER System	Eine Flexibilitätsressource (z.B. PV-Anlage mit Batterie, Wärmepumpe oder Elektroauto) stellt eine Flexibilität für das Netz zur Verfügung und erbringt dadurch eine Dienstleistung für das Energiesystem. Sie kann auf ein externes Signal hin ihr Einspeise- oder Verbrauchsverhalten verändern.

Tabelle 2: Auflistung der Akteure und deren Funktion bei AF3

Ergebnis zu Ziel 2: Aus Informationstechnischen-Sicht sind die zwei Kategorien von Anwendungsfällen sehr ähnlich. Flexibilitäten der DER Systemen werden in den Marktplattform zur Verfügung und von Flexibilitätsanfragern gesucht und gebucht. Der Blockchain Ansatz wird hierbei zwischen den zwei energietechnischen Herangehensweisen nicht unterscheiden. Nämlich wird der Flexibilitätsmarkt dezentral gestaltet, wobei die Flexibilitäten und die entsprechenden Anfragen in das Ledger geschrieben und nach optimal passenden Paarungen mittels Smart Contracts gesucht. Auf bestimmten Akteuren wird bei der Entwicklung der Blockchain Anwendung verzichtet. Die Logik des Aggregators und des Bilanzkreisverwalters wird von den Smart-Contracts übernommen. Die Funktion des VNB Kapa kann in der Blockchain nicht abgebildet werden, da es externe Informationen bedarf, um die entsprechenden Überprüfungen durchzuführen. Das gleiche gilt für den Messstellenbetreiber, deren Funktion ebenfalls durch ein externes Orakel nachgebaut werden könnte. Dies ist war nicht Teil der entwickelten Blockchain Prototyp und ist als Erweiterung in AP8 aufgelistet.

Ergebnis zu Ziel 3: Die in AP1 definierten Anwendungsfälle wurden in folgenden Versionen tabellarisch aufgelistet, mit Hinweisen zur IT-Sicherheit kommentiert und den Projektpartnern zur Verfügung gestellt:

Anwendungsfall 1 – Netzengpassmanagement (Version 0.20)

Anwendungsfall 2 – Netzengpassmanagement (Version 0.11)

Anwendungsfall 3 – Flex on Demand (Version 0.95)

Bei der Betrachtung der UML-Ablaufdiagramme der Anwendungsfälle aus IT-Sicherheitsperspektive wurden die Projektpartner auf eine Vielzahl von Events hingewiesen, bei denen unklar war, welche Daten wie erhoben und verarbeitet werden sollen. Des Weiteren wurden Empfehlungen zu nötigen Sicherheitsstandards und Hinweise auf potentielle Gefahrenstellen ausgesprochen.

Die entsprechenden Tabellen befinden sich in einem dedizierten Dokument „FlexHub (IT-Sicherheit)“ welches während der Projektlaufzeit als „lebendes Dokument“ geführt wurde. Aus Gründen der Übersichtlichkeit wird an einigen Stellen auf dieses Dokument referenziert. Es kann als Ergänzung zu diesem Abschlussbericht bei den Projektpartnern angefragt werden.

II.1.1.2 AP 1.2 Ableitung von Business- und Anreizmodellen

Auf Basis der der in AP1.1 definierten Anwendungsfälle wurden in diesem Arbeitspaket die Geschäfts- und Anreizmodelle für unterschiedlichen FlexHub Akteure abgeleitet. Dabei erfolgte zunächst eine Ausgestaltung im Rahmen eines Business Model Canvas für jeden der unterschiedlichen Akteure. Maßgebliche Erkenntnis aus diesem Arbeitspaket sind, dass für jeden der beteiligten Akteure durch den Einsatz und Betrieb des FlexHubs ein Mehrwert entsteht:

- Netzbetreiber (Rolle im Projekt Mitnetz) können den FlexHub nutzen um gemäß §14c EnWG transparent, diskriminierungsfrei und marktgestützt Endkundenflexibilität zu beschaffen, um dabei Engpasse im Verteilnetz auszugleichen. Dadurch reduzieren sich die Kosten für den Netzbetreiber, die dieser für die Abregelung von Erneuerbaren Energien bezahlen müsste. Nach §14a EnWG kann der Netzbetreiber für diese netzdienliche Steuerung ein reduziertes Netzentgelt gewähren. Im Rahmen des §14a und §14c EnWG sparen Netzbetreiber bis zu 50% der Kosten für übliches Einspeisemanagement ein.
- IoT-Anbieter und technische Aggregatoren, die die Endkundenflexibilität technisch erschließen (im Projekt Kiwigrid) können für ihre Kunden einen Mehrwert schaffen, in dem sie es den Endkunden ermöglichen ihre Flexibilität zur Lösung von Netzengpassproblemen zu vermarkten. So entstehen neue Erlösmodelle für sich und die Nutzer ihrer Home Energy Management Systeme und Alternativen zur Optimierung zum Eigenbedarf.
- Aggregatoren und die Betreiber des Flexibilitäten Markts (Rolle im Projekt EnergieDock) können entweder eine fixe Gebühr für die Bereitstellung und Nutzung dieser Plattform anbieten

oder nehmen einen prozentualen Anteil für jede Transaktion (Buchung und Steuerung von Flexibilität), die über die Plattform durchgeführt wurde.

- Endkunden können bei den gegenwärtigen Kosten für die Netzentgelte etwa 200€ pro Jahr durch die Vermarktung ihrer Flexibilität erzielen. Der Feldversuch im FlexHub hat gezeigt, dass dieses ohne Komfortverlust für die Endkunden möglich ist.

Weitere Geschäftsmodelle für die Vermarktung von Flexibilität im Day-ahead oder Intraday-Handel oder zum Redispatch 2.0 wurden im Projekt nicht untersucht, können aber für die genannten Akteure weitere Revenue-Ströme generieren, sodass entsprechend höhere Beträge für alle Teilnehmer zu erwirtschaften sind.

Abbildung 1 zeigt die Wertschöpfungskette für den FlexHub exemplarisch mit der Flexibilitätsplattform NEMO.spot, die im Rahmen des Projekts von der EnergieDock GmbH entwickelt wurde. Rechts dargestellt sind die Endkunden bzw. Endverbraucher, die Flexibilität bereitstellen. Dabei kann es sich um die Wallbox eines privat genutzten E-Autos handeln, es kann aber auch eine Wärmepumpe sein oder eine Ladesäule auf einem Betriebshof. Alle diese Flexibilitäten haben es gemeinsam, dass sie bis zu einem bestimmten Zeitpunkt eine bestimmte Menge an Energie beziehen wollen. Wann der Bezug dieser Energie passiert, ist für die Anbieter aber nicht relevant, solange der Bezug zum vorgegebenen Endzeitpunkt erfolgt ist. IoT-Anbieter wie Kiwigrid heben dieses Flexibilitätspotential, ermitteln die Verfügbarkeit und stellen die technische Anbindung zu den Verbrauchern her und können es dann auf einer Flexibilitätsplattform wie NEMO.spot anbieten. Hier kann diese Flexibilität dann von unterschiedlichen Akteuren für unterschiedliche Use-Cases verwendet werden. Im Fokus dieses Projekt lag der Anwendungsfall, dass ein Verteilnetzbetreiber die Flexibilität zum Netzengpassmanagement nach §14a EnWG nutzt. Wie bereits erwähnt können Netzbetreiber hier bis zu 50% der Kosten für übliches Einspeisemanagement einsparen. Diese Ersparnis kann dann die beteiligten Akteure, den Betreiber des Flexibilitätsmarktes, den IoT-Anbieter und Endkunden weitergegeben werden. Die weiteren Use Cases, die nicht im Rahmen dieses Projekts untersucht wurden, sind die Bereitstellung von Flexibilität für den Redispatch-Prozess durch Übertragungsnetzbetreiber oder einen Handel von Flexibilität im Intra-Day- oder auch Day-Ahead-Handel. Auch in diesen Fällen können die entsprechenden Erlöse an die beteiligten Akteure verteilt werden. Hier erfolgte bisher aber noch keine nähere Untersuchung über die möglichen Gewinnpotentiale.

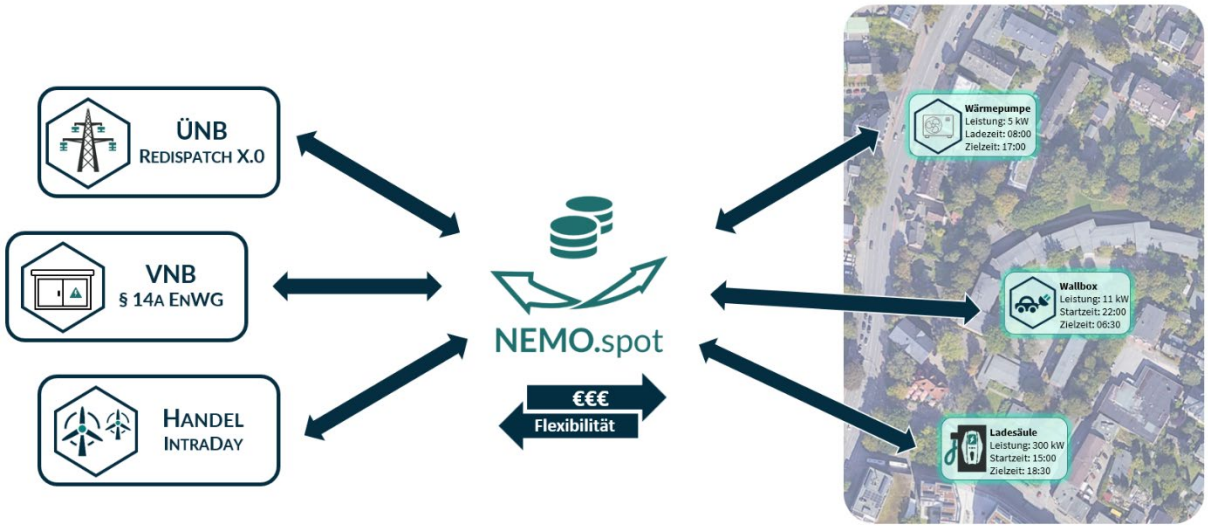


Abbildung 9: Wertschöpfungskette für Geschäftsmodelle mit dem FlexHub

II.1.2 Arbeitspaket 2: Testfall- und Testplanerstellung

Ziel dieses Arbeitspakets war die Definition von Testfällen und die Erstellung eines Testplans.

Basierend auf einer feingranularen Beschreibung der FlexHub Anwendungsfälle wurden Testfälle für die Komponenten des FlexHubs, wie z.B. dem Kommunikationsstack und den Proof-of-Concept-Demonstratoren, sowie für den abschließenden Acceptance Test der FlexHub Plattform definiert.

Es wurden 21 Client/Server Testfälle für die IEC61850-8-2 Simulationsumgebung implementiert und erfolgreich durchgeführt. Grundlage für die Testserie stellte das Lastenheft des FNN für die IEC 61850 Steuerbox dar. Konkret wurde dabei das gesamte IEC 61850 Datenmodell von Server über LogicalDevice und LogicalNode bis hin zu Data Objekten und deren Attributen abgeprüft.

Da die FNN Steuerbox während der Projektlaufzeit nicht lieferbar war, wurde ein 3-Stufen-Modell bis hin zur vollständigen Implementierung des Kommunikationsstacks entwickelt, um Auswirkungen der ausbleibenden Lieferung der Steuerbox auf das Gesamtprojekt zu minimieren.
In Stufe 1 wurde zur autonomen Cliententwicklung ein IEC 61850 SCL-Server (virtueller Server) mit Kommunikationsschnittstelle XMPP (Extensible Messaging and Presence Protocol) nach IEC 61850-8-2 entwickelt, so dass das Verhalten der späteren Hardware möglichst realgetreu simuliert werden konnte. Nach Implementierung aller Services auf der Ebene 61850-8-2 folgte in Stufe 2 der Stack-Entwicklung, die Integration der vorgegebenen Sicherheitsfeatures nach IEC 62351-4 (End2End Security) in die Clientanwendung. Mit der Entwicklungsstufe 3, d.h. mit der Einbindung der Hardware (Steuerbox) und mit der Integration es Client Stack Moduls in eine Java GUI Applikation sollte die

Stackentwicklung abgeschlossen werden. Dieser Schritt konnte nicht wie vorgesehen umgesetzt werden. Detaillierte Informationen zu diesem Konzept finden sich der Beschreibung zu AP 6.

Die APs 2 (Testpläne) und 6 (Entwicklung und Implementierung der Plattform) fokussierten sich auf die Ermöglichung der Demonstrationsversuche und Feldversuche auch ohne die FNN Steuerbox. Dementsprechend verlagerten sich die Aktivitäten u.a. der HAW Hamburg bzw. der EnergieDock in diesem Arbeitspaket in die Erstellung von Testfällen für im Rahmen von AP6 von EnergieDock entwickelte Plattform.

II.1.3 Arbeitspaket 3: Informationsflüsse und Datenmodell

II.1.3.1 Informationsflüsse

Die Informationsflüsse zwischen den Akteuren und Komponenten wurden in den IEC 6259-2-Templates für die Use Cases unter Federführung der HAW Hamburg erarbeitet und liegen dem Projektkonsortium vor. Sie finden sich in den Use Case Beschreibungen zu den Anwendungsfällen 1 – 3 (A1 -A3) und werden aus Gründen der Übersicht hier nicht erneut wiedergegeben. Grundsätzlich beinhalten die Informationsflüsse die beteiligten Akteure (Information-Producer und -Receiver), eine Beschreibung der Art der Informationen, die ausgetauscht werden und eine Einordnung in welchem Prozessschritt die Informationen in welchem Kontext und unter welchen Konditionen ausgetauscht werden. Daher gehören zu den Informationsflüssen auch immer ein UML-Sequenzdiagramm, welches die Ablauffolge bzw. den Prozess des Anwendungsfalls beschreibt.

Aufbauend auf der feingranularen Beschreibung der Anwendungsfälle aus AP1 (Aufbereitung durch Projektpartner) werden in diesem Arbeitspaket gemeinsam mit den Projektpartnern Informationsflüsse identifiziert und in UML-Diagrammen dargestellt. Für diese werden IT-Sicherheits-Anforderungen, erste IKT-Konzepte und ein Transaktionsmodell für die Blockchain abgeleitet.

Aus Fraunhofer-Sicht ergeben sich hierfür die folgenden Ziele:

1. Identifikation und Modellierung von Smart Contracts im Transaktionenmodell
2. Ableitung von IT-Sicherheitsanforderungen, die sich aus den Informationsflüssen ergeben.

Ergebnis zu Zielen 1-1:

Authentifizierung in der Blockchain

Ein blockchainbasierter Prototyp sieht die Interaktion mehrfachen Akteure innerhalb der Blockchain vor. Die im Flexibilitätsmarkt unternommenen Aktionen lösen Transaktionen aus, die mit der Absenderadresse gesendet und signiert werden. Eine Adresse in Quorum, dem für FlexHub gewählten Ethereum-basierten Blockchain-Netzwerk, identifiziert ein kryptografisches Schlüsselpaar, das zu einer agierenden Instanz in der Blockchain zugeordnet ist. Der private Schlüssel besteht aus 64 Hex-Zeichen und wird dazu verwendet, den 128 Hex-Zeichen langen öffentlichen Schlüssel mit dem Eliptic Curve

Digital Signature Algorithm (ECDAS oder DSA) zu generieren. Die Adresse entspricht den letzten 40 Hex-Zeichen des SHA-3 Hashwertes des öffentlichen Schlüssels [21] [22]. Der private Schlüssel wird dazu verwendet, Nachrichten zu signieren, während der öffentliche Schlüssel zur Überprüfung der digitalen Signatur benutzt wird.

Für den blockchainbasierten Prototyp ergeben sich zwei Authentifizierungsmodelle, wo die Benutzer auf unterschiedliche Art und Weise die Transaktionen auf Blockchain-Knoten initiieren können:

1. Token-basiertes Modell: Die kryptografischen Schlüsselpaare werden auf der API-Ebene erstellt und dort hinterlegt (vgl. Abbildung 10). Sie sind beim Aufbau und Signieren der Transaktionen verwendet, bevor die Transaktionen auf der Blockchain verarbeitet werden. Der Ablauf wird vom Benutzer initiiert, der ein Token bei der API-Instanz abfragt. Das Token dient zur Identifikation für den Benutzer und wird bei den darauffolgenden API-Anfragen mitgegeben, was als Anmeldung für den Benutzer fungiert, bevor Transaktionen mit seiner Adresse ausgeführt werden. Das Token wird mit dem öffentlichen Schlüssel des Benutzers verschlüsselt und kann bei der Anmeldung des Benutzers umgekehrt von der API wieder entschlüsselt werden. Auf API-Ebene wird somit ein „JSON Web Signature“ (JWS) Objekt erstellt.

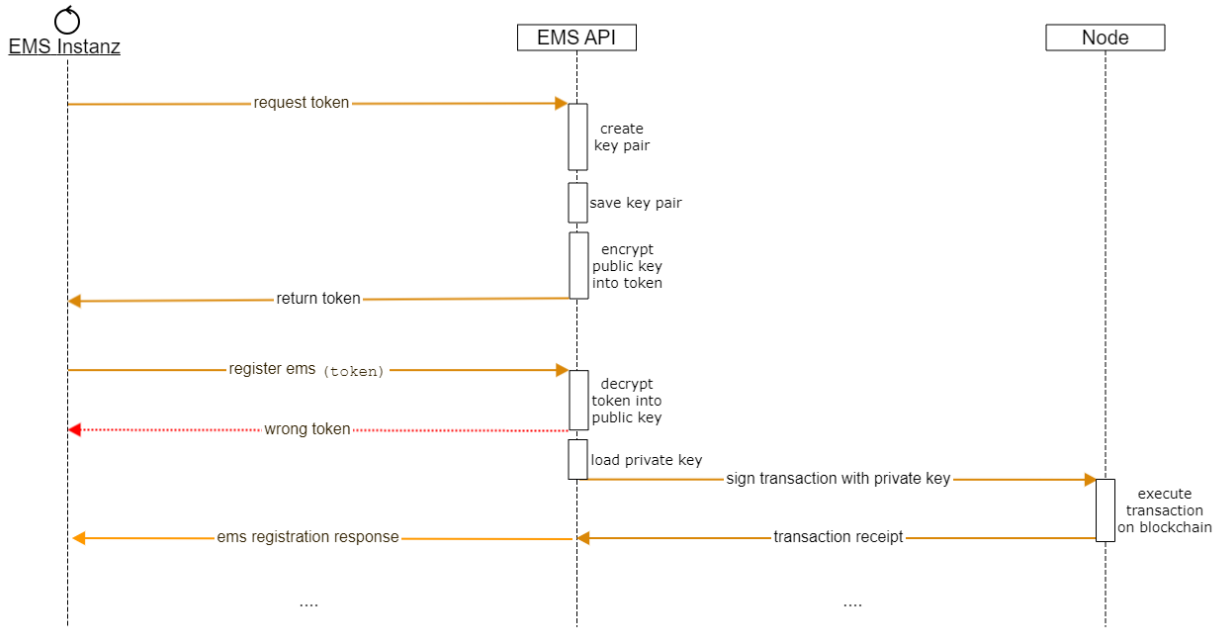


Abbildung 10: Anwendung mit Token-basierter Authentifizierung

2. dApp Modell: Der Name des Modells beruht auf dem Ansatz der dezentralen Applikation, die eine Web-Anwendung bezeichnet, die mit einem Blockchain-Netzwerk im Hintergrund verbunden ist. Ein digitales Wallet wird in diesem Fall auf der Client-Ebene integriert (vgl. Abbildung 11). Die Transaktionen entstehen auf der Client-Ebene und werden mit den privaten Schlüsseln des Wallets signiert. Die im Modell (a) auf API-Ebene umgesetzte Logik steht in diesem Fall auf der Client-Ebene. In Abbildung 11 ist die Kommunikation zwischen

dem im Browser installierten Wallet und im FlexHub Blockchain bereitgestellten Smart Contracts geschildert. Zunächst muss der Nutzer die Schlüsselpaare im Wallet generieren bzw. importieren, falls welche bereits vorhanden sind (Schritt 0). Die Verbindung zum Blockchain Knoten wird hergestellt, sobald die Adresse und Port des Knoten im Wallet eingetragen werden (Schritt 1). Danach kann der Nutzer Aktionen in der dApp mithilfe des Wallet durchführen. Eine Aktion, die den Blockchain-Zustand durch Schreiben neuer Informationen verändert, wird als Transaktion übertragen. Die Transaktion wird mit dem Privatschlüssel des im Wallet verwendeten Kontos signiert (Schritt 2). Einige Aktionen erfordern keine Änderungen in den Smart Contracts, sie lesen lediglich Daten aus dem Ledger. In dem Fall führt das Wallet Abrufe im Knoten aus, die zu einer Aktualisierung der User-Interface der dApp führt (Schritt 3).

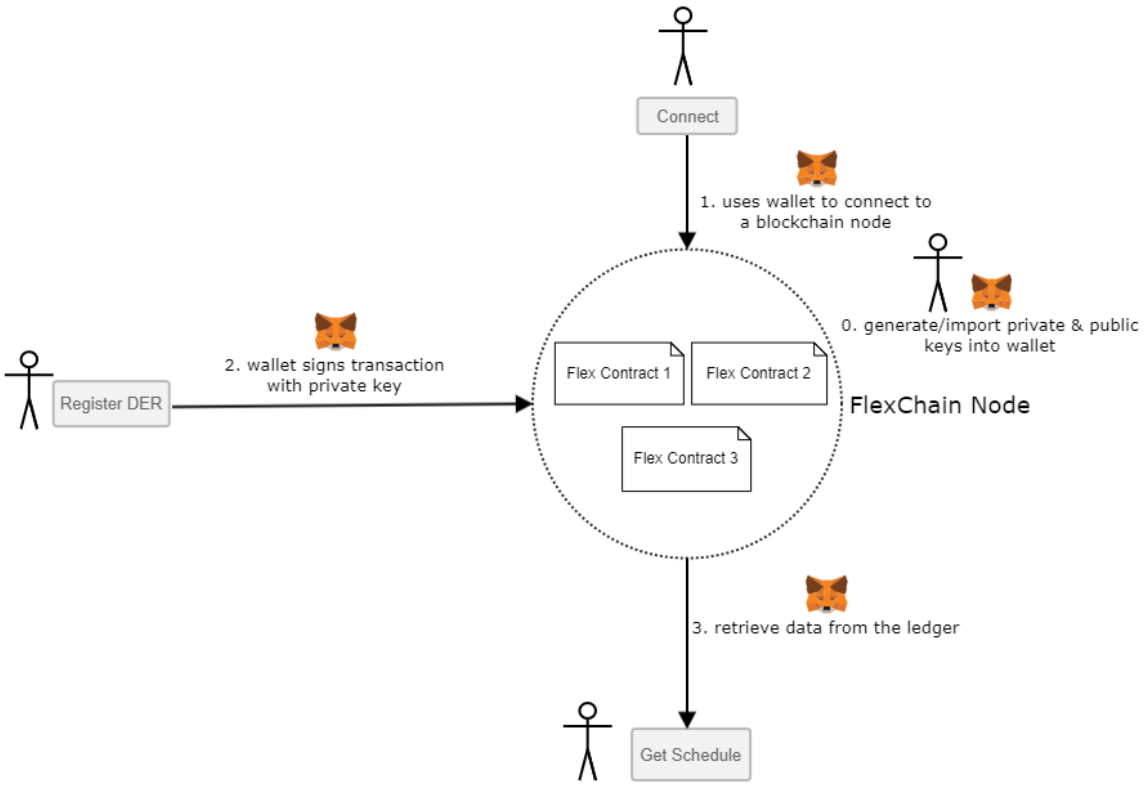


Abbildung 11: Ablauf des Authentifizierungsverfahrens mittels eines digitalen Wallet

Der Hauptnachteil von dem 1. Modell ist das Authentifizierungskonzept. Der Benutzer hat keine Kontrolle über die Adressen, die in den Transaktionen verwendet werden, denn sie sind auf der API-Ebene gespeichert. Zudem ist das Senden vom Token einem Man-in-the-Middle Angriff ausgesetzt, wo ein Angreifer auf das Token gelangen und es weiter bei der Anmeldung zur API-Instanz senden kann, ohne dass er als Angreifer erkannt werden kann. Darüber hinaus ist das Modell für dezentralisierte Denial-of-Service (DDoS) Attacken auf API-Instanzen anfällig. Weiterhin wird das 2. Authentifizierungsmodell weiter beschrieben und analysiert.

Digitale Wallets

Die digitalen Wallets sind ein wesentlicher Bestandteil der Interaktionen mit der Blockchain, denn sie verwalten die Adressen für den Benutzer. Wallets arbeiten mit Konten, die auf Basis von öffentlichen/privaten Schlüsselpaaren die Interaktionen mit der Blockchain ermöglichen. Die Hauptanwendungen der Wallets sind die Überweisung von Kryptowährungen zwischen Konten und das Halten des aktuellen Betrags an Währung, die in der Blockchain definiert ist. Zudem protokollieren die Wallets die von einem Konto initiierten Smart Contract Methodenaufrufe.

Die Wallets teilen sich in zwei Kategorien auf: Hot- und Cold-Wallets. Die Hot Wallets laufen als Anwendungen/Plug-Ins im Browser und sind mit dem Internet verbunden. Dadurch sind die Wallets von dieser Art verschiedenen Gefahren ausgesetzt, wie z.B. Phishing-Angriffen oder Angriffen, die Browser-Schwachstellen ausnutzen. Cold- oder Hardware-Wallets hingegen sind physischen Geräte, die sich zwar meist über den USB-Port mit dem Rechner verbinden, sind aber nicht mit dem Internet direkt verbunden. Diese Art von Wallets gilt als die sicherere von den beiden.

MetaMask

MetaMask ist ein Open-Source digitales Wallet, das von ConsenSys entwickelt wurde [23]. MetaMask unterstützt Ethereum-basierte Blockchain Netzwerke und ist primär als Browser Plug-in zu verwenden. MetaMask wird an dApps angebunden, die mit Smart Contracts in der Blockchain kommunizieren. Die Verbindung zum Blockchain-Netzwerk und Ausführung von Transaktionen läuft über die web3 API ab [24]. Die kryptografischen Schlüsselpaare können entweder vom Wallet selbst erstellt oder ins Wallet als bereits vorhandene JSON-Datei importiert werden.

Ein MetaMask Zugang ist passwortgeschützt. Für die Privatschlüssel-Generierung wird eine Seed Phrase (oder Mnemonic Phrase) verwendet, die aus 12 oder 24 Wörtern besteht. Die erforderliche Entropie für die Generierung der Seed-Phrase beträgt 128 bits bzw. 256 bits. Zudem wird die Seed Phrase zum Wiederherstellen des Zugangs verwendet, falls der Nutzer das Passwort vergisst.

MetaMask verbindet sich im Hintergrund mit einem Infura Dienst, wo die Daten vom Blockchain-Knoten als Mini-Knoten („Light-Node“) synchronisiert werden. Infura ist eine Plattform, worüber angebotene APIs die Mini-Knoten ansprechen [25]. Die von dApp entstandenen Transaktionen sind auf den Mini-Knoten ausgeführt, bevor die Ethereum-basierten Knoten vollständig synchronisiert werden, was eine längere Zeit in Anspruch nehmen kann.

Dezentrale Anwendung

Die Funktionalität des FlexChain-Clients als dApp ist in Abbildung 12 angezeigt. Vorausgesetzt, dass das Wallet im Browser installiert ist, fängt der Benutzer mit der Erstellung eines Kontos an. Anschließend wird das Wallet so eingestellt, dass es sich mit dem FlexChain-Netzwerk verbindet, d.h. durch Eintragen der entsprechenden URL und Port in die Wallet-Einstellungen.

Bei der Ausführung einer Aktion in der dApp durch den Benutzer wird eine Transaktion mit dem Absender als Konto-Adresse aufgebaut. Anschließend wird die Transaktion mit dem entsprechenden privaten Schlüssel signiert und an die Target-Blockchain gesendet.

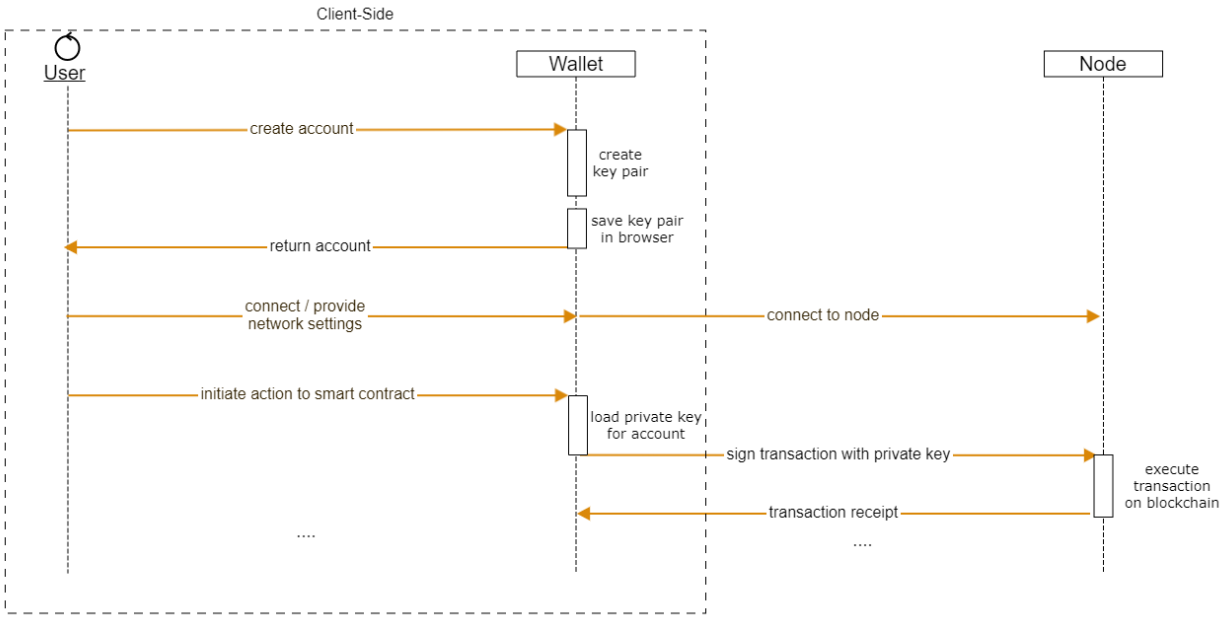


Abbildung 12: dApp mit Wallet Authentifizierung

Ergebnis zu Ziel 2: Ergänzend zu der tabellarischen Auflistung der Anwendungsfälle aus AP1 wurden noch Abschnitte zu den Informationsflüssen erarbeitet. Hierbei wurde dokumentiert, wo im UML-Diagramm Daten anfallen und verarbeitet werden. Falls bekannt, wurden auch die jeweiligen Protokolle und Sicherheitsmechanismen festgehalten. Die Projektpartner wurden darüber informiert, dass es noch zahlreiche Informationsflüsse gab, bei denen weder die Technologie noch ein Sicherheitsmechanismus bekannt waren. Dies galt es aufzubessern. Als Handreichung wurde hierfür der in AP3.2 erarbeitete Sicherheitskatalog zur Verfügung gestellt.

II.1.3.2 Zugriffsmodell für sichere Transaktionen

In diesem AP werden die Informationsflüsse zwischen dem FlexHub und dessen Nutzer erfasst. Aus den sich daraus ergebenden IT-Sicherheitsanforderungen werden Anforderungen für das Datenmodell abgeleitet. Dies berücksichtigt u.a. die Verwaltung der sich dynamisch ändernden Zugriffsberechtigungen in einer hochskalierten und verteilten Anwendungssituation. Ein weiterer Fokus liegt auf der Vertraulichkeit und Integrität der Daten.

Für die Fraunhofer Institute stellen sich folgende Ziele:

1. Ableiten von Sicherheitsaspekten aus der Perspektive Blockchain
2. Entwicklung eines IT-Sicherheitskatalog nach Best Practice

Ergebnis zu Ziel 1:

DApp Sicherheitsanalyse

Das dApp Authentifizierungsmodell weist den Vorteil eines bewährten kryptografischen Wallet auf. Dabei hat der Benutzer die Kontrolle über das von ihm direkt erstellte Konto. Die privaten Schlüssel sind lokal im Browser gespeichert. Ein Phishing-Angreifer muss deswegen an die Seed Phrase gelangen, bevor die privaten Schlüssel entschlüsselt werden können. Das Zugang-Passwort zum Wallet gewährt dem Angreifer die Möglichkeit, die aufbewahrten Konten frei zu benutzten. Darunter zählt auch die Option, die privaten Schlüssel zu exportieren und evtl. in ein anderes Wallet zu importieren und sie weiterzuverwenden.

Ein weiterer Angriffspunkt stellen dApp-Anfragen von falschen Benutzern dar, d.h. Benutzer, die weder zu VNB noch zu EMS gehören. Dagegen könnte ein Ansatz wie Whitelisting die Gefahr mindern. Dabei sind die Hauptadressen, d.h. die Adressen von den Instanzen, die Knoten betreiben, in eine Liste der zugelassenen Adressen hinzugefügt, sobald sich die entsprechenden Knoten-Betreiber über einen Kanal wie E-Mail anmelden und die dazugehörige Dokumentation zur Netzwerk-Teilnahme nachweisen [26]. Hinzu kommt der Genehmigungs-Mechanismus („Permissioning“), der Quorum für teilnehmende Blockchain Knoten eingebaut hat. Ein Knoten wird mit seiner IP-Adresse in einer Permissioning-Datei hinzugefügt, sobald dieser Teil des Netzwerks wird. Knoten, deren IP-Adressen nicht gelistet sind können weder Verbindungen mit anderen Knoten initiieren noch Verbindungsanfragen bekommen. Dadurch schütz Quorum vor unbekannten Knoten, die trotz laufenden Knoten-Clients Software sich nicht ans Netzwerk anschließen und Transaktionen nicht verarbeiten können [27]. Ein externer Kanal wie E-Mail oder das IP-Permissioning stellen keine Garantien für die verlässliche Verwendung des Wallet von den beabsichtigten Nutzern.

Eine weitere Option ist die Einführung von Tokens, wodurch einer Adresse der Zugang zum Netzwerk erteilt wird. Ein einzelnes Token reicht für eine begrenzte Anzahl von Transaktionen aus, während die Ausgabe eines Tokens für eine IP oder MAC-Adresse gedeckelt ist. Somit hält sich die Gefahr eines DoS Angriffs in Grenzen. Dezentralisierte DoS Angriffe sind jedoch weiterhin möglich. Es wird deswegen empfohlen, eine Wallet Seed Phrase mit 24 Wörtern einzustellen.

Ergebnis zu Ziel 2:

Im Rahmen dieses APs wurde ein Katalog mit IT-Sicherheitsanforderungen für den FlexHub erarbeitet. Dieser unterteilt sich in drei Abschnitte. Im Folgenden wird der grobe Inhalt der Abschnitte mit einigen Beispielen vorgestellt. Der vollständige IT-Sicherheitskatalog befindet sich in Kapitel 3 des Dokuments „FlexHub (IT-Sicherheit)“.

Im ersten Abschnitt werden zunächst IT-Sicherheitsanforderungen in die Kategorien *Allgemeine Sicherheitsanforderungen*, *Vertraulichkeit*, *Integrität* und *Verfügbarkeit* eingeteilt und formuliert. Zu den verschiedenen Anforderungen werden die Hintergründe kurz erläutert und Hinweise für eine Implementierung gegeben.

Allgemeine Sicherheitsanforderungen

Zu den allgemeinen Sicherheitsanforderungen zählt, dass Kommunikationsendpunkte immer beidseitig zu authentifizieren sind. Dadurch können nur berechtigte Nutzer den FlexHub nutzen und sich gleichzeitig sicher sein, mit einer validen FlexHub-Instanz zu kommunizieren. Durch beiderseitige Authentifizierung kann die Angriffsoberfläche selbst bei ins Internet exponierten Systemen erheblich reduziert werden. Alle Anfragen von anderen Systemen können so schnell verworfen werden. Auch bei internen Systemen wird die Angriffsoberfläche verringert. Denn selbst wenn Komponenten kompromittiert wurden, ist nur eine sehr eingeschränkte Kommunikation mit anderen Systemen möglich. Bei dieser Anforderung handelt sich um einen Aspekt der sogenannten Defense in Depth (mehrere Schichten von Sicherheitsmechanismen). Eine weit verbreitete Standardlösung ist der Einsatz von SSL/TLS in entsprechender Konfiguration. Zu beachten und zu bewerten ist jedoch, dass unter Umständen ein erhöhter Ressourcenbedarf entsteht, der einen Denial of Service (DoS) begünstigen könnte.

Eine Anforderung, die aus der Definition der Anwendungsfälle aus AP1 resultierte, ist eine Lösung für das Dining Philosophers Problem. Dieses könnte konkret in AF1 auftreten. Denn durch mehrere gleichzeitige Anfragen könnten Flexibilitätsressourcen so reserviert werden, dass keine der Anfragen positiv beantwortet wird. Deswegen ist es hier ggf. nötig die Flexibilitätsressourcen während des Buchungsprozesses zu reservieren, sodass diese Ressource anderen anfragenden Nutzern nicht zur Verfügung stehen. Eine mögliche Umsetzung einer solchen Reservierung könnte durch die Nutzung des Status-Felds eine FlexOffer erfolgen. Wichtig ist es auch Timeouts für entsprechende Reservierungen zu betrachten, um DoS Angriffe nicht zu erleichtern.

Auch Kommunikationsprotokolle werden in den allgemeinen Sicherheitsanforderungen betrachtet. Diese sind z.B. so zu entwickeln, dass sie vor Angriffen durch Wiedereinspielung (Replay-Attacken) sicher sind. Bei Replay-Attacken nutzt ein Angreifer den Umstand, dass Nachrichten immer identisch aufgebaut sind. So kann ein Angreifer auch in verschlüsselten Szenarien und ohne Möglichkeit die Verschlüsselung zu brechen, legitime Nachrichten an das System verschicken. Dies kann beispielsweise durch den Einsatz von Nonces, Sequenznummern oder Zeitstempeln verhindert werden. SSL/TLS z.B. nutzt zum Schutz gegen Replay-Attacken sogenannte Nachrichtenauthentifizierungscodes. Konkret im FlexHub Szenario sind gerätespezifische Protokolle potenziell gefährdet, die zur Kommunikation mit einer Flex-Ressource zum Einsatz kommen.

Im Rahmen der allgemeinen IT-Sicherheitsanforderungen wird auch empfohlen, eine Schutzbedarfsanalyse für alle Kommunikationsbeziehung und Daten im System durchzuführen. Das Ziel hierbei ist es den Schaden vorab einzuschätzen, bevor ein kritisches Ereignis eingetreten ist. Der Schutzbedarf wird in Bezug auf Vertraulichkeit, Integrität und Verfügbarkeit ermittelt. Eine Kategorisierung des Schutzbedarfes erfolgt in mindestens die drei Kategorien *normal*, *hoch* und *sehr hoch*:

- *Normal*: Die Schadensauswirkungen sind begrenzt und überschaubar.
- *Hoch*: Die Schadensauswirkungen können beträchtlich sein.
- *Sehr hoch*: Die Schadensauswirkungen können ein existentiell bedrohliches, katastrophales Ausmaß erreichen.

Ein sehr hoher Schutzbedarf in allen drei Bereichen wird beispielsweise für die Datenbank des Flex-Registers erwartet. Ziel der Schutzbedarfsfeststellung ist es, für die erfassten Daten zu entscheiden, welchen Schutzbedarf sie besitzen. Dieser orientiert sich an den möglichen Schäden, die mit einer Beeinträchtigung der betroffenen Anwendungen und damit der jeweiligen Geschäftsprozesse verbunden sind. Weitere Informationen und Beispiele hierzu können [28] entnommen werden.

Weitere allgemeine IT-Sicherheitsanforderungen betreffen die Nutzung kryptografischer Verfahren. Diese sollten jederzeit leicht ausgetauscht werden können, um bei neuen Erkenntnissen, die die Sicherheit des Systems gefährden, auf andere Verfahren umstellen zu können. Bei der Auswahl kryptografischer Verfahren sind aktuelle Empfehlungen zu beachten, z.B. die des BSI [29]. Durch die typischen langen Lebenszyklen von Produkten im Energiesektor sollten Parameter (z.B. Schlüssellängen) ausreichend lang gewählt werden. Die Länge sollte dabei derzeitige Empfehlungen ggf. übersteigen. Eine Prognose über die Eignung von verschiedenen kryptografischen Verfahren über einen Zeitraum von 10 Jahren ist schwierig, deshalb ist eine Austauschbarkeit der Parameter oder der gesamten Verfahren unbedingt im System-Design zu beachten. Um die Bedeutung dieser Empfehlung zu verdeutlichen sei auf die Verschlüsselung WEP im Bereich Wireless LAN verwiesen. Diese wurde erst nach Jahren im produktiven Einsatz kompromittiert.

Des Weiteren sollte das System so gestaltet werden, dass es Angreifern nicht möglich ist durch Abwärtskompatibilität Schwachstellen auszunutzen. Insbesondere kryptografische Funktionen könnten im Laufe der Zeit durch neue Erkenntnisse unsicher werden. In solchen Fällen wird meist ein Update der betroffenen Protokolle vorgenommen und für eine Übergangszeit ggf. die unsichere Funktion akzeptiert. Bei Downgrade-Angriffen versucht der Angreifer die Abwärtskompatibilität zu nutzen, um unsichere Funktionen auszuführen. Im Rahmen von FlexHub muss darauf geachtet werden, dass solche Angriffe nicht möglich sind und das Zeitfenster für den Übergang möglichst kurzgehalten wird.

Vertraulichkeit

Im Hinblick auf Vertraulichkeit empfiehlt der erstellte IT-Sicherheitskatalog, Übertragungen im Netzwerk durch geeignete Verschlüsselung vor dem Abhören zu schützen. Dabei sind alle in den Anwendungsfällen definierten Informationsflüsse zu beachten. Für die Verschlüsselung sollten Standardverfahren zum Einsatz kommen, die beispielsweise von dem BSI oder dem National Institute of Standards and Technology (NIST) empfohlen werden [29] [30].

Zudem sollte die verschlüsselte Netzwerkkommunikation nach Möglichkeit Perfect Forward Secrecy (PFS) implementieren. Sollten kryptografische Schlüssel kompromittiert werden, so sichert die PFS vorherige Daten vor der nachträglichen Entschlüsselung.

Integrität

Um die Integrität von Daten zu schützen, sollten Übertragungen im Netzwerk vor absichtlichen und unabsichtlichen Änderungen geschützt werden. Die absichtliche Manipulation von Nachrichten ist eine gängige Methode, beispielsweise mithilfe von Man-in-the-Middle-Angriffen (MitM), um gefälschte

Daten in das Zielsystem einzuschleusen. Einfache Fehlerkorrekturmaßnahmen (z.B. Cyclic Redundancy Check (CRC) [31]), wie sie in Transportprotokollen eingesetzt werden, bieten in der Regel keinen ausreichenden Schutz. Nachrichten sind daher durch geeignete Verfahren (z.B. Keyed-Hash Message Authentication Code (HMAC) [32]) zu schützen. Hierfür sollten Standardverfahren zum Einsatz kommen, die beispielsweise von dem BSI oder dem NIST empfohlen werden [29] [30].

Verfügbarkeit

Die Verfügbarkeit spielt im FlexHub eine wichtige Rolle. Darum ist es umso wichtiger DoS-Angriffe zu verhindern. Bei einem solchen Angriff könnte ein Angreifer z.B. versuchen den Prozess der Leistungsanfrage zu initiieren und Flexressourcen zu reservieren. Anstatt das Ziel eines zeitnahen Vertragsabschlusses zu verfolgen, wird dieser möglichst lange hinausgezögert und schließlich abgebrochen. Durch viele solcher Anfragen könnten die Ressourcen unbrauchbar gemacht werden. DoS-Angriffe werden vereinfacht, wenn das Zielsystem Ressourcen reservieren oder Operationen durchführen muss, die aufwendiger sind als jene, die der Angreifer durchführt. Im FlexHub ist es unter Umständen unumgänglich Flexressourcen zu reservieren. In diesem Fall sind Mechanismen zu entwickeln, die eine hinreichende Flexibilität bieten, um einen DoS auszuschließen oder zu erschweren.

Ein solcher Mechanismus wäre z.B. die Definition von Timeouts. Das heißt, das System ist so zu entwerfen, dass ein Prozess zu jedem beliebigen Zeitpunkt abgebrochen werden kann und alle Ressourcen zeitnah wieder freigegeben werden. Diese Anforderungen geht über die Behandlung der definierten Fehlerfälle der Anwendungsfälle hinaus. Im FlexHub werden beispielsweise verschiedene Ereignisse durch eine Nachricht bestätigt. Sollte eine solche Bestätigung ausbleiben oder gar die Flex-Ressource selbst nach erfolgreicher Bestätigung nicht angefragt werden, muss diese nach einem Timeout wieder freigeben werden. Der Timeout-Wert sollte sich dabei an den zu erwartenden prozessbedingten Latenzen summiert mit den Latenzen durch den Kommunikationsweg orientieren.

Im zweiten Abschnitt des IT-Sicherheitskatalogs werden ausführlich konkrete IT-Sicherheitsanforderungen an ausgewählte Komponenten der FlexHub-Infrastruktur diskutiert. Die Auswahl dieser Komponenten erfolgte basierend auf den zu dem Zeitpunkt vorhandenen Ergebnissen der Arbeitspakete. Der Abschnitt ist zweigeteilt. Der erste Teil widmet sich dem Thema Logging/Protokollierung. Hierbei liegt der Fokus auf Logging im Kontext der IT-Sicherheit und der damit verbundenen Analyse von sicherheitsrelevanten Vorfällen. Im zweiten Teil werden das Message Queuing Telemetry Transport (MQTT) Protokoll und mögliche Sicherheitstechniken diskutiert. Zu dem Zeitpunkt der Diskussion sollte MQTT innerhalb der FlexHub-Infrastruktur als Protokoll für Push-Nachrichten ausgehend vom Flex-Register (IRES-Server) verwendet werden.

Logging

Log-Daten, auch Protokolldaten genannt, sind Ereignismeldungen, die von Betriebssystemen, Diensten oder anderen Software-/Firmwarekomponenten erzeugt werden. Zahlreiche Komponenten in der FlexHub-Infrastruktur stellen wichtige Log-Daten bereit. Der Bereich Log-Management beschäftigt sich mit der Sammlung, Verarbeitung, Speicherung und Auswertung dieser Log-Daten. Insbesondere vor

dem Hintergrund zunehmender Dezentralisierung komplexer Systeme, z.B. durch die Container-Virtualisierung im Flex-Register, aber auch wegen immer professioneller Angriffe mit geringer Sichtbarkeit ist ein gut abgestimmtes Log-Management ein essenzielles Element für die IT-Sicherheit. Darüber hinaus profitiert auch das Leistungsmonitoring von einem strukturierten Log-Management und das Debugging im Fehlerfall wird deutlich beschleunigt. Probleme können häufig schneller identifiziert werden, ohne sich von Teilsystem zu Teilsystem vorzuarbeiten. Nachfolgend werden einige Anforderungen an ein mögliches Log-Management für den FlexHub beschrieben. Innerhalb von Kapitel 3 des vollständigen IT-Sicherheitskatalogs (siehe Anhang) werden auch konkrete Umsetzungsempfehlungen zu den wichtigsten Punkten gegeben. Diese Umsetzungsempfehlungen sind so gestaltet, dass deren Realisierung einfach überprüft werden kann.

Zentrale Speicherung

Log-Daten, die innerhalb eines Bereichs (z.B. IRES-Server) anfallen, sollten in einem zentralen, besonders geschützten und bei Bedarf hochverfügbaren System gesammelt und gespeichert werden. Dadurch ist sichergestellt, dass Angriffe auch dann noch nachvollziehbar sind, wenn ein Angreifer Log-Daten auf Komponenten verändert oder gelöscht hat, um der Entdeckung aktiv entgegenzuwirken. Für den gesamten Übertragungsweg der Log-Daten sollten Anforderungen bezüglich Vertraulichkeit, Integrität und Verfügbarkeit formuliert werden. Diese sollten z.B. festlegen, dass Log-Daten von allen Systemen werden nur verschlüsselt und authentifiziert übertragen werden dürfen.

Zweckdienliche Auswahl und Konfiguration der Log-Quellen

Innerhalb der verschiedenen Bereiche des FlexHubs existieren zahlreiche Systeme, Komponenten und Architekturen, die Log-Daten erzeugen. Zur Verfügung stehende Log-Quellen sollten danach priorisiert werden, wie hilfreich sie bei der Erfüllung zuvor definierter Ziele sind wie z.B. der Erkennung von Cyberangriffen oder dem Monitoring von kritischen Komponenten im FlexHub. Da jede Anbindung einer neuen Log-Quelle mit Aufwand verbunden ist, sollten die Log-Quellen entsprechend ihrer Priorität nach und nach an das zentrale Log-Management-System angebunden werden. Jede Log-Quelle sollte so konfiguriert werden, dass sie möglichst viele zweckdienliche und wenige überflüssige Log-Daten liefert, um den Speicherbedarf des zentralen Log-Management-Systems effizient zu nutzen. Um zu entscheiden, wie zweckdienlich eine bestimmte Log-Meldung ist, sollten konkrete Szenarien formuliert werden. Idealerweise verfügt eine Quelle über ein strukturiertes, maschinenlesbares Format und die verschiedenen Typen von Log-Daten sind gut dokumentiert sowie mit einer Versionsnummer versehen.

Eine weitere Herausforderung sind mehrzeilige Log-Daten. Diese werden beim Weiterleiten oft wie getrennte Log-Daten betrachtet, was die weitere Verarbeitung deutlich erschweren kann. Falls die Möglichkeit existiert sollte auf einzeilige Logs umgestellt werden. Sollte dies nicht möglich sein, kann das zentrale Log-Management-System diesen Umstand ggf. beheben.

Jegliche Art von Audit-Logs sind besonders wertvoll. Audit-Logs beschreiben die Interaktionen eines Nutzers mit dem System oder zwischen Systemen. Im Kontext des FlexHubs sollten alle in den Anwendungsfällen beschriebenen Events als wichtige Audit-Logs betrachtet werden.

Negative Auswirkungen auf den Produktivbetrieb vermeiden

Die Erhebung und Weiterleitung von Logdaten sollte den Produktivbetrieb des FlexHubs nicht negativ beeinflussen. Es sollte daher auf die Auslastung von Prozessor, Arbeitsspeicher, Festplatte und Netzwerk geachtet werden.

Grundsätzlich sollten bereits vorhandene Mechanismen zur Logdaten-Sammlung und Logdaten-Weiterleitung gegenüber zusätzlichen Lösungen bevorzugt werden, um Kosten und ggf. sogar Sicherheitsrisiken zu vermeiden. Bei Linux-Systemen können die standardmäßig vorhandenen Syslog-Daemons (Syslog oder syslog-ng) so konfiguriert werden, dass sie Log-Daten per Netzwerk weiterleiten bzw. annehmen. Dabei kann auch eine Ziel-abhängige Filterung erfolgen. Ebenfalls besteht die Möglichkeit TCP und SSL/TLS zu verwenden [33].

Es hat sich bewährt, mithilfe der oben genannte Dienste Logs jeweils auf einem dedizierten Relay-Server pro Netzbereich bzw. Rechenzentrum zu sammeln. Zur Übertragung von diesen Relay-Servern zum zentralen Log-Management-System können dann dedizierte Agenten (z.B. Elastic Beats [34]) eingesetzt werden.

Zeitsynchronisation und Zeitstempel

Alle anliefernden Komponenten im FlexHub sowie das zentrale Log-Management-System sollten ihre Systemzeit synchronisieren. Zeitstempel sollten so präzise und vollständig wie möglich sein. Ein Negativbeispiel sind die Einträge von Linux-Log-Dateien. Diese enthalten standardmäßig kein Jahr, keine Sekundenbruchteile und keine Zeitzone. Sprünge in der Systemzeit sowie stark abweichende Entstehungs- und Empfangszeiten von Logs sollten ebenfalls geloggt werden.

Datenschutz

Gesetzliche und unternehmensspezifische Datenschutzanforderungen müssen beachtet werden. Personenbezogene Daten in zentralen Log-Management-Systemen sollten bei Bedarf pseudonymisiert werden und für die De-Pseudonymisierung muss ein Prozess festgelegt werden. Falls keine De-Pseudonymisierung erforderlich ist, können Daten auch anonymisiert werden. Dadurch dürfen Daten evtl. auch länger gespeichert werden. Im Falle des FlexHubs findet sich im Datenmodell der Punkt Adresse mit den Feldern Land, Stadt, Straße, Postleitzahl, Name und Koordination. Es ist unbedingt zu prüfen, wie mit den Log-Daten, die diese Felder enthalten, umzugehen ist.

Suche, Visualisierung, Auswertung und Alarmierung

Die Daten im zentralen Log-Management-System sollten flexibel und performant durchsuch- und visualisierbar sein. Im Open-Source-Bereich ist die Kombination von Elasticsearch, Logstash und Kibana (der sogenannte Elastic Stack [35]) zum Aufbau eines zentralen Log-Management-Systems weit verbreitet. Optional kann das System durch den Message-Broker und Stream-Prozessor Apache Kafka [36] erweitert werden. Dieser erlaubt ein flexibles Routing der Logs und ermöglicht eine redundante, persistente Pufferung.

Über die manuelle Suche und Visualisierung hinaus sollte es möglich sein, automatische Alarme zu generieren, die entweder beim Zutreffen manuell definierter (Korrelations-) Regeln oder bei statistischen Ausreißern ausgelöst werden.

MQTT

Das Protokoll MQTT wurde speziell für den Nachrichtenaustausch im Machine-to-Machine (M2M) Bereich entwickelt. In der FlexHub-Infrastruktur wird das Protokoll für die ausgehenden (Push) Nachrichten des Flex-Registers (IRES-Server) verwendet. Die aktuellste Version MQTT 5 wurde am 31.10.2018 veröffentlicht. Die derzeit weit verbreitetste Version ist 3.1.1.

Eine umfassende Einleitung in die Grundlagen und Prinzipien von MQTT findet sich in [37].

Im Vergleich zu alternativen Protokollen nutzt MQTT ein anderes Kommunikationsparadigma und stellt geringe Anforderungen an Ressourcen. Anstatt des weit verbreiteten „Request-Response“ bzw. „Client-Server“-Verfahrens (wie z.B. bei HTTP oder REST), dass eine zyklische Abfrage von Informationen voraussetzt (Polling), wird bei MQTT ein „Publish-Subscribe“ Verfahren eingesetzt. Bei diesem Verfahren wird der Initiator einer Nachricht (Publisher) von dem Adressaten oder den Adressaten (Subscriber) getrennt. Es gibt keinen direkten Kontakt zwischen Publisher und Subscriber, sondern es wird ein Mittelsmann namens Broker eingesetzt. Der Broker überträgt die Nachrichten an all jene Adressaten, die sich für den Empfang der Daten bereit erklärt haben. Diese Erklärung nennt man „Subscriben“. Damit ein Adressat nicht alle Nachrichten eines Gerätes empfängt, für welches er sich „subscribed“ hat, werden die Nachrichten durch Topics unterteilt.

Nachfolgend werden einige Anforderungen und Empfehlungen für MQTT beschrieben. Der Fokus liegt dabei auf der sicheren Konfiguration und der anschließenden Nutzung. Weitere Punkte und vertiefende Details sind Kapitel 3 des vollständigen IT-Sicherheitskatalogs (siehe Anhang) zu entnehmen.

Version

Das MQTT-Protokoll beschreitet zurzeit einen Versionssprung von Version 3.1.1 auf Version 5. Dabei sind in diesem Prozess nicht alle Systeme in der Lage MQTT 5 zu sprechen. Zur Lösung dieses Problems kann ein hybrider Broker verwendet werden. Hierbei sind die beiden wichtigsten Broker Mosquitto (Open Source) [38] und HiveMQ (kommerziell) [39] zu nennen. Bisher bietet jedoch nur HiveMQ eine vollständige Unterstützung aller in MQTT 5 verfügbaren Funktionen an. Wichtig ist es darauf zu achten,

dass neben dem Broker die jeweiligen Publisher und Subscriber ebenfalls in der Lage sind, die geforderte Protokollversion zu sprechen.

Quality of Service

Die Quality of Service (QoS) Stufe bestimmt die Abläufe zwischen einem Publisher/Subscriber und einem Broker, die beim Empfangen eines MQTT-Paketes zu absolvieren sind. Dabei können Publisher/Subscriber sowie Broker die Rolle des Empfängers annehmen. Falls eine bestimmte QoS-Stufe zwischen Publisher und Subscriber gewünscht ist, müssen beide diese in Richtung des Brokers angeben. Definiert sind die Stufen 0, 1 und 2.

Bei der QoS-Stufe 0 wird nach einer gesendeten Nachricht keine Bestätigung vom Empfänger zurückgeschickt. Dadurch wird bei einer verlorenen Nachricht auch kein erneutes Senden durch das MQTT-Protokoll veranlasst. Bei dieser QoS-Stufe wird die Zuverlässigkeit der Übertragung ausschließlich durch andere Protokolle wie TCP gewährleistet. Gerade bei einer Funkübertragung kann dies zu Problemen führen.

Die QoS-Stufe 1 erhöht die Zuverlässigkeit, indem sie eine Bestätigung seitens des Empfängers der Nachricht fordert. QoS-Stufe 1 garantiert, dass eine Nachricht mindestens einmal an den Empfänger zugestellt wird. Der Sender der Nachricht nutzt dazu eine bislang durch ihn unbenutzte Identifikationsnummer für das Paket. Die Bestätigungsnachricht durch den Empfänger muss dabei dieselbe Identifikation haben wie die des empfangenen Paketes. Kommt keine Bestätigung, kann der Sender die Nachricht erneut senden.

Die zuverlässigste QoS-Stufe ist 2. Jede gesendete Nachricht soll genau einmal ankommen und es darf keine Duplikate geben. Dazu werden zwei Bestätigungsnachrichten genutzt. Die erste bestätigt den Empfang der Nachricht. Die zweite Nachricht bestätigt, dass alle Subscriber die Nachricht einmal empfangen haben. Nicht alle MQTT-Bibliotheken unterstützen diesen Modus.

Topics

Topics sind ein zentrales Konzept in MQTT. Sie definieren einen Pfad, an den Nachrichten initiiert werden (Publish). Adressaten können sich anhand dieses Pfades für bestimmte Nachrichten registrieren (Subscribe). Dabei gelten für Topics einige grundlegende Regeln:

- Das Topic muss eindeutig sein und darf auf dem Broker nur einmal vorkommen,
- die Länge eines Topics muss mindestens ein Zeichen lang sein,
- die Topics sind case sensitive,
- Leerzeichen sind im Topic erlaubt,
- nur der NULL-Character darf nicht enthalten sein,
- $ zu Beginn eines Topics ist für Status Nachrichten reserviert,
- die maximale Länge eines Topics ist auf 65 kB begrenzt.

Die verschiedenen Ebenen des Topics werden mit dem Symbol / getrennt. Diese Trennung erlaubt den Aufbau einer Baumstruktur. Dabei sollte unbedingt darauf geachtet werden, vereinheitlichte und sinnvolle Namen zu verwenden. Diese sollten die Struktur des Gesamtsystems leicht verständlich machen und ggf. Raum für zukünftige Erweiterungen bieten.

Es sollte darauf geachtet werden, keine persönlichen Informationen in einem Topic zu verwenden, um den Datenschutz zu gewährleisten.

Filter

Statt sich beim Broker für den Empfang jedes einzelnen Topics zu registrieren, sollten Filter eingesetzt werden. Filter basieren auf Single- und Multilevel Wildcards in der Semantik einer Topic-Hierarchie.

Eine Single-Level-Wildcard wird mit dem Symbol + gekennzeichnet. Sie kann verwendet werden, um alle Elemente einer Stufe in der Topic-Hierarchie zu adressieren.

Eine Multi-Level-Wildcard wird mit dem Symbol # gekennzeichnet und häufig verwendet, um sich für viele Nachrichten gleichzeitig zu registrieren.

Authentifizierung

Die Authentifizierung ermöglicht es dem Broker, Publisher und Subscriber zu identifizieren. Es ist hiermit möglich nur bestimmte Topics für unterschiedliche Gruppen verfügbar zu machen. Der Zugriff auf empfindlichen Daten wird damit reglementiert. Die Authentifizierung erfolgt über einen Nutzernamen und ein Passwort. Ab MQTT 5 kann auch nur ein Passwort verwendet werden. Zusätzlich besteht die Möglichkeit eine Authentifizierung über Zertifikate oder Tokens zu implementieren.

Verschlüsselung

Der MQTT-Standard verweist beim Thema Verschlüsselung auf das TLS-Protokoll. Dieses schützt die MQTT-Verbindungen und sollte zu jeder Zeit aktiv sein. Aktuell wird die TLS-Version 1.2 am häufigsten verwendet. Der Nachfolger Version 1.3 bietet eine vereinfachte Konfiguration, einen kürzeren Handshake und sollte bevorzugt werden.

Skalierung

Mit MQTT 5 ist es möglich ``Shared Subscriptions'' zu verwenden. Ohne den Einsatz von Shared Subscriptions empfangen alle Subscriber eine Kopie der Nachricht vom Broker. Dies ist in der Regel gewollt, kann aber zu hohen Lasten bei bestimmten Anwendungsbereichen führen. Um Systeme horizontal zu skalieren, können Shared Subscriptions eingesetzt werden. Mit dieser Technik werden Nachrichten von Broker abwechselnd an die Subscriber geschickt. Gerade im Backend-Bereich ist dies eine interessante Lösung für Skalierungen ohne den Einsatz einer zusätzlichen Load-Balancing Applikation. Eine QoS-Stufe von 2 ist mit Shared Subscriptions allerdings nicht möglich.

Intervalle

Mit MQTT 5 kann Nachrichten ein Ablaufintervall zugeordnet werden. Dies kann dazu genutzt werden, ein System zu instruieren, Daten nur eine bestimmte Zeit zu persistieren. Für Echtzeitanwendungen sind alte Daten, die später als aktuelle Daten eintreffen, häufig nicht gewünscht. Auch die Steuerung eines Aktors erfolgt meist nur mit den aktuellsten Daten.

Der dritte Abschnitt des IT-Sicherheitskatalogs stellt Techniken vor, mit denen eine sichere Kommunikation über WebSockets realisiert werden kann. Eine Kommunikation über WebSockets wird in der FlexHub-Infrastruktur häufig verwendet. Um diese mit den vorgestellten Techniken effektiv absichern zu können, formuliert dieser Abschnitt auch Empfehlungen und Anleitungen zur Nutzung der Techniken. Weitere und ausführlichere Darstellungen finden sich im Dokument „FlexHub (IT-Sicherheit)“.

TLS

Bei TLS ehem. SSL handelt es sich um ein Verschlüsselungsprotokoll[1], dass eine sichere Übertragung von Informationen über TCP/IP-basierte Verbindungen ermöglicht. Für diese sichere Übertragung wird im ersten Schritt ein Handshake zwischen Client und Server vollzogen. Dieser ist ein wichtiger Bestandteil des Protokolls, durch den sich die Kommunikationspartner auf kryptografische Verfahren und ein Master Secret verständigen. Die kryptografischen Verfahren werden durch sogenannte Cipher-Suites festgelegt und definieren, wie die Verschlüsselung, die Integritätssicherung, die Schlüsseleinigung und die Authentifizierung vollzogen werden. Bei dem Master Secret handelt es sich um ein gemeinsames Geheimnis, aus dem die Kommunikationspartner Sitzungsschlüssel ableiten [29], [40].

Wie sicher der Informationsaustausch über TLS ist, hängt stark von der korrekten Einrichtung und Konfiguration der TLS-Verschlüsselung ab. In den folgenden Paragrafen werden hierfür grobe anleitende Schritte samt Empfehlungen beschrieben, durch die ein grundlegendes Maß an Sicherheit realisiert werden kann.

Erzeugen eines Certificate Signing Requests

Ein TLS-Zertifikat vermittelt die Sicherheit, dass hinter einer Website eine echte und vertrauenswürdige Person oder Institution steht. Dafür garantiert es, dass der Betreiber einer Website seine Identität eindeutig nachgewiesen hat und sensible Daten zwischen Server und Computer verschlüsselt ausgetauscht werden. Um ein solches TLS-Zertifikat zu erhalten, muss im ersten Schritt ein CSR erzeugt werden. Hierbei handelt es sich um eine Zertifikatsignierungsanfrage für den Betreiber der Website. Für die Erzeugung eines CSRs wird zunächst ein asymmetrisches kryptografisches Schlüsselpaar benötigt. Dieses besteht aus einem privaten und einem öffentlichen Schlüssel. Der öffentliche Schlüssel wird in das CSR und letztlich auch in das Zertifikat eingefügt und ist dadurch öffentlich zugänglich. Der zugehörige private Schlüssel muss jedoch geheim gehalten werden und dient der eindeutigen Identifikation des Betreibers.

[1] Ein Verschlüsselungsprotokoll besteht aus einem Schlüsselaustauschprotokoll und einem symmetrischen Verschlüsselungsverfahren.

Für die Erzeugung eines solches Schlüsselpaar können gängige TLS-Bibliotheken wie OpenSSL[2], LibreSSL[3] oder ähnliche genutzt werden. Diese bieten zudem verschiedene weitere Funktionen zur Nutzung von TLS.

Bei der Generierung des Schlüsselpaares besteht die Wahl zwischen verschiedenen Algorithmen und Schlüssellängen. Ein sicherer und starker privater Schlüssel ist wichtig, um Angreifer daran zu hindern Identitätsmissbrauch bzw. Impersonation-Angriffe durchzuführen.

Wie Abbildung 13 zeigt, empfiehlt das BSI für die Generierung eines asymmetrischen kryptografischen Schlüsselpaares den Rivest–Shamir–Adleman (RSA)-Algorithmus, den vom Digital Signature Standard (DSS) spezifizierten Digital Signature Algorithm (DSA) oder den auf elliptischen Kurven basierenden Elliptic Curve Digital Signature Algorithm(ECDSA) [29].

Da der ECDSA effizienter im Hinblick auf die Rechenzeit ist, sollte dieser bevorzugt werden [41].

Neben einer Empfehlung für Algorithmen enthält Abbildung 13 auch Angaben zur minimalen Schlüssellänge. Bei der Nutzung kürzerer Schlüssel besteht die Gefahr, dass diese durch „Brute Force“ Angriffe erraten werden könnten. Daher sollte die empfohlene Mindestlänge nicht unterschritten werden, um ein ausreichendes Sicherheitsniveau zu gewährleisten.

Algorithmus	Minimale Schlüssellänge	Verwendung spätestens ab	Verwendung bis
Signaturschlüssel für Zertifikate und Schlüsseleinigung			
ECDSA	250 Bit		2027+
DSS	2000 Bit		2022
	3000 Bit	2023	2027+
RSA	2000 Bit		2023
	3000 Bit	2024	2027+

Abbildung 13: Empfohlene Algorithmen und Mindest-Schlüssellängen für die Schlüsselgenerierung für das TLS-Protokoll laut BSI

Wie dort zu sehen ist, können sowohl der DSA- als auch der RSA-Algorithmus aktuell noch mit einer Schlüssellänge von 2000 Bit genutzt werden. Spätestens ab dem Jahr 2023 (DSA) bzw. 2024 (RSA) sollten diese jedoch auf 3000 Bit erweitert werden.

Bei dem ECDSA-Algorithmus reicht eine Schlüssellänge von 250 Bit aus, um dasselbe Sicherheitsniveau wie bei einem 3000 Bit RSA- oder DSA-Algorithmus zu erreichen [40].

Neben der Erstellung eines starken privaten Schlüssels ist auch der Schutz und die sichere Aufbewahrung dessen essenziell. Private Schlüssel sollten nur auf einem vertrauenswürdigen Computer

[2] https://www.openssl.org/
[3] https://www.libressl.org/

generiert und hinterher mit einem starken Passwort geschützt sicher abgespeichert werden. Eine sichere Speicherung kann z.B. durch die Nutzung von zertifizierter Hardware wie Chipkarten oder Hardware-Sicherheitsmodulen erfolgen [29].

Eine weitere Empfehlung ist das regelmäßige (d.h. jährlich oder häufiger) Erneuern von Zertifikaten und privaten Schlüsseln. Dies birgt zwar Aufwand, reduziert jedoch das Risiko einer Kompromittierung, da sich die Sicherheitsstandards stetig weiterentwickeln und nach bekanntgewordenen Angriffen angepasst werden. So kann sichergestellt werden, dass sowohl das genutzte Zertifikat als auch der Schlüssel ein hohes Sicherheitsmaß hat.

Wurde ein Schlüsselpaar nach den vorgestellten Empfehlungen erstellt, so kann ein CSR generiert werden. Eine solche Anfrage enthält Informationen zu dem anfragenden Betreiber (Person oder Institution) und dessen Webseite. Diese Informationen werden von der Certificate Authority zum Erstellen und Signieren des Zertifikats verwendet.

Konfigurieren von TLS auf dem Webserver

Sobald das Zertifikat durch eine vertrauenswürdige Instanz wie eine Certificate Authority signiert wurde, kann es bereits genutzt werden. Hierfür muss es auf dem Gerät abgespeichert und der Webserver so konfiguriert werden, dass er weiß, wo das Zertifikat liegt und es für TLS-Verbindungen nutzt. Neben dem Pfad zu dem erstellten Zertifikat können in der Konfiguration des Webservers noch viele andere Einstellungen getroffen werden. Eine korrekte Konfiguration ist wichtig, um zu gewährleisten, dass nur sichere kryptografische Verfahren verwendet und bekannte Schwachstellen vermieden werden.

So sollte bei der Konfiguration auf die Nutzung veralteter SSL/TLS-Versionen verzichtet werden, da diese bekannten Schwachstellen haben. Zu empfehlen sind die TLS Versionen 1.2 und 1.3. Diese gelten als sicher und bieten ein modernes authentisiertes Verschlüsselungsverfahren namens Authenticated Encryption with Associated Data (AEAD) [29] [40].

Abweichungen von dieser Empfehlung sollten nur in triftigen Ausnahmefällen und mit Vorsicht erfolgen.

Auch sollte sichergestellt werden, dass der Webserver während des TLS-Handshakes nur sichere und kryptografisch starke Cipher-Suites unterstützt und aus dem Angebot des Clients auswählt. Eine Liste der empfohlenen Cipher-Suites stellt das BSI in [29] vor.

Ein Kriterium nach dem die Cipher Suites ausgewählt werden sollten, ist die Forward Secrecy bzw. Perfect Forward Secrecy. Hierbei handelt es sich um ein Protokollmerkmal, das darauf abzielt, auch im Falle einer Kompromittierung eines privaten Schlüssels Sicherheit zu vermitteln. Konkret sorgt es dafür, dass für die Kommunikation ein Sitzungsschlüssel generiert wird, der auch dann nicht rekonstruiert werden kann, wenn der private Schlüssel eines Sitzungsteilnehmers kompromittiert werden sollte. Das heißt, selbst wenn der private Schlüssel kompromittiert werden sollte, können alle zuvor

aufgezeichneten Gespräche nicht entschlüsselt werden. Bei Cipher Suites, die keine Forward Secrecy bieten, könnte der Angreifer auch alle alten aufgezeichneten Gespräche entschlüsseln. Um Forward Secrecy zu unterstützen, sollte beim Schlüsselaustausch auf Elliptic-Curve Diffie-Hellman Ephemeral (ECDHE) oder Diffie-Hellman Ephemeral (DHE) Suites gesetzt werden. Der RSA-Schlüsselaustausch bietet diese Eigenschaft nicht [29].

Es sei angemerkt, dass es auch trotz Forward Secrecy im Falle einer Kompromittierung des privaten Schlüssels unabdingbar ist, einen neuen privaten Schlüssel und neue Zertifikate zu generieren und die alten sperren zu lassen.

Weiterhin sollten bei der Konfiguration des Webservers vollständige Zertifikatsketten verwenden werden. In der Regel sind heutige Zertifikate Teil einer Zertifikatskette, der sogenannten Chain of Trust [42]. Diese Kette erhöht die Sicherheit für den Fall der Kompromittierung eines privaten Schlüssels einer Certificate Authority. Um das Prinzip zu verstehen, veranschaulicht Abbildung 14 den Fall ohne Zertifikatskette.

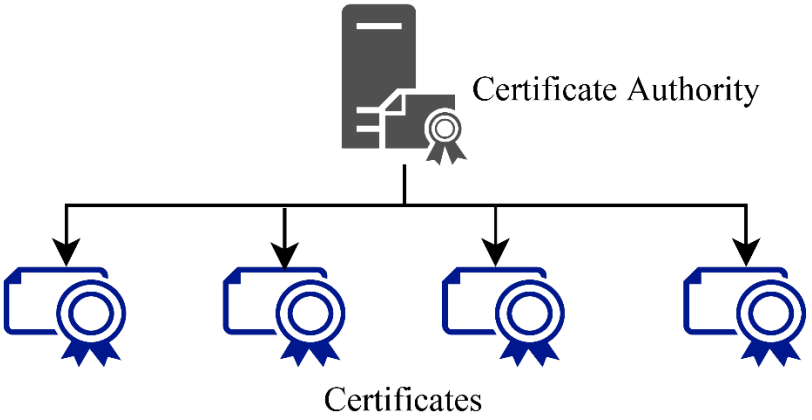


Abbildung 14: Hierarchische Vertrauensstruktur

Hier werden sämtliche Zertifikate von einer einzelnen Certificate Authority verteilt. Sollte in diesem Fall der private Schlüssel der Certificate Authority kompromittiert werden, so sind alle ausgestellten Zertifikate ungültig und müssen gesperrt werden.

Um diesem Fall vorzubeugen, führt die Chain of Trust sogenannte Intermediate Certificate Authorities ein. Wie in Abbildung 15 zu sehen ist, sitzen diese zwischen einer Root Certificate Authority und den Endanwendungen.

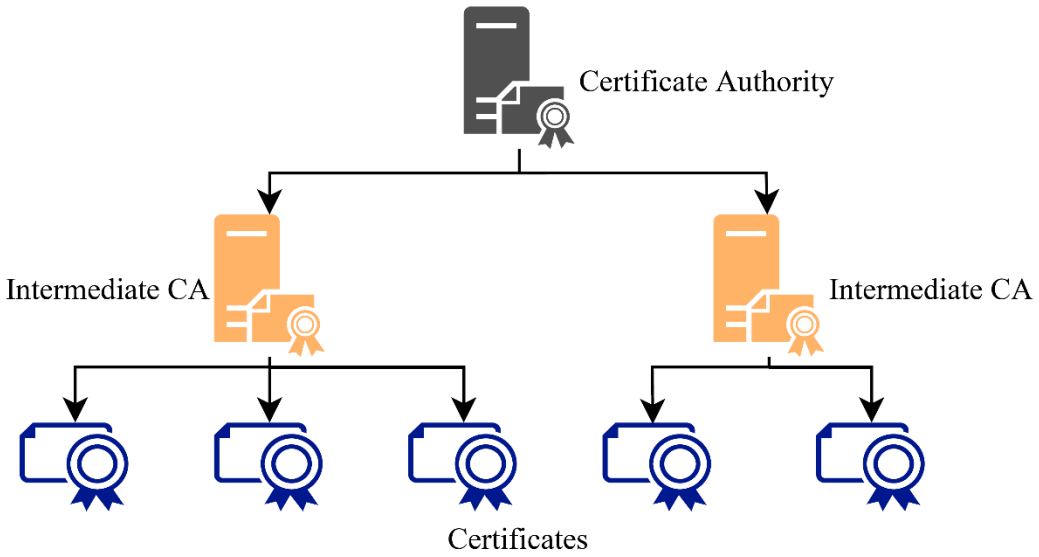


Abbildung 15: Verteilte Vertrauensstruktur --- Chain of Trust

Dadurch wird das Ausstellen von Zertifikaten auf mehrere Quellen verteilt. Dies sorgt dafür, dass im Falle einer Kompromittierung eines privaten Schlüssels nur die davon abhängigen Zertifikate ungültig sind. In diesem Fall wird diese Anzahl deutlich reduziert.

Heute öffentlich genutzten Zertifikate sind immer Teil einer Kette mit mindestens einer Intermediate Certificate Authority. Dementsprechend versenden ausstellende Certificate Authoritiies neben dem eigentlichen Server Zertifikat auch ein Intermediate Root-Zertifikat oder je nach Kettenlänge auch direkt ein ganzes Zertifikatsbündel. Um den Server korrekt zu konfigurieren, sollten alle erhaltenen Zertifikate hinterlegt werden [43].

DNSSEC

Das Domain Name System (DNS) ist ein zentrales Element in IP-basierten Netzwerken und dient im Wesentlichen der Namensauflösung. Das heißt, es liefert z.B. die zugehörige IP-Adresse zu einer Domain. Obwohl dieses System eine essenzielle Rolle hat, besitzt es keine Sicherheitsmechanismen. So ist es Angreifern mit Zugriff auf eine DNS-Kommunikation möglich mitzulesen, auf welche Dienste die Opfer zugreifen und gegebenenfalls sogar die DNS-Antworten zu manipulieren, wodurch Opfer auf maliziöse Seiten geführt werden könnten. So können Angreifer selbst die TLS-Verschlüsselung umgehen und z.B. private Daten, die vom Nutzer eingegeben werden, abgreifen. Solche Angriffe nennt man DNS-Spoofing oder DNS-Hijacking [44] [45].

Um dem entgegenzuwirken wurde Domain Name System Security Extensions (DNSSEC) eingeführt. Hierbei handelt es sich um eine Reihe von Technologien, die die Validierung von DNS-Antworten ermöglichen [44] [45]. Dadurch wird Datenintegrität und Authentizität gewährleistet und somit der Schutz gegen Spoofing Angriffe erhöht.

Der Unterschied zwischen dem normalen DNS und der Variante mit DNSSEC wird in Abbildung 16 vereinfacht dargestellt. Durch DNSSEC wird der grundlegende Mechanismus einer DNS-Anfrage und -Antwort nicht verändert. Allerdings werden DNS-Zonen nun signiert und ermöglichen so eine Validierung. Schickt ein DNS-Client nun eine DNS-Anfrage für eine DNSSEC-signierte Zone, so wird die DNS-Antwort zu nächst validiert. Hierfür wird die digitale Signatur entschlüsselt, ein Hashwert der Antwort berechnet und mit dem empfangenen und entschlüsselten Hashwert verglichen. Durch die erfolgreiche Entschlüsselung, wird sichergestellt, dass die Antwort von dem korrekten und erwarteten Absender stammt und durch den Abgleich der Hashwerte wird überprüft, ob sich die Antwort auf dem Weg vom Absender zum Empfänger verändert und somit manipuliert wurde.

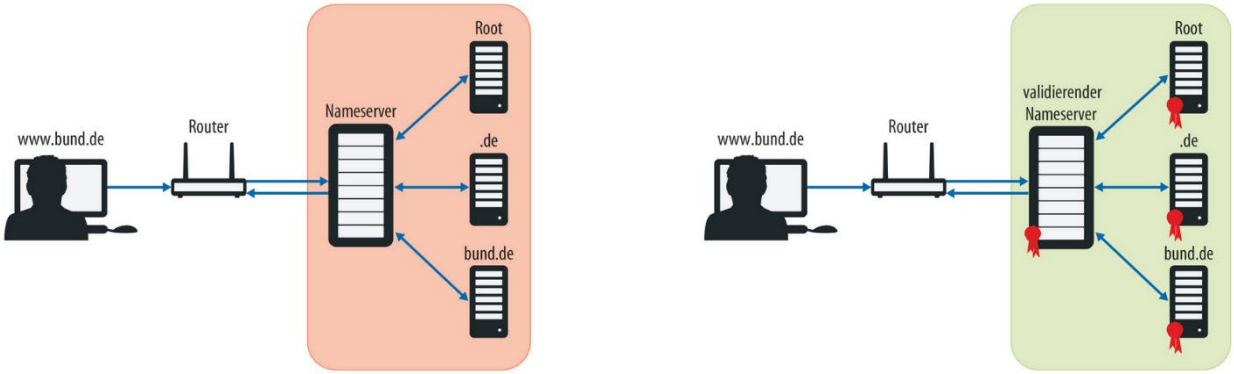


Abbildung 16: Unterschied zwischen DNS und DNSSEC [45]

Obwohl DNSSEC bereits für eine erhöhte Sicherheit bei der Nutzung von DNS sorgt, reicht dies noch nicht aus. Dies liegt im Wesentlichen an zwei Faktoren:

- Die Validierung der DNS-Antworten erfolgt in der Regel auf dem Nameserver. Eine weitere Validierung auf dem Router oder beim Client findet nicht mehr statt. Das heißt, sollte es einem Angreifer gelingen, sich an dieser Stelle in die Kommunikation zu hängen, so kann er weiterhin DNS-Pakete abfangen und manipulieren [45].
- DNSSEC sieht keine Vertraulichkeit vor. Das heißt, die Anfragen und Antworten werden weiterhin unverschlüsselt übertragen und ein Angreifer könnte z.B. mitverfolgen, auf welche Webseiten seine Opfer zugreifen [45].

Um diese beiden Mängel zu beheben, sollte DNSSEC mit einer Verschlüsselung wie DNS over TLS (DoT) oder DNS over HTTPS (DoH) kombiniert werden. Bei DoT und DoH handelt es sich um eine Verschlüsselung des DNS Verkehrs mittels TLS. Hier wird anstelle der für DNS üblichen ungesicherten UDP Kommunikation eine TCP- bzw. HTTP-Verbindung aufgebaut, die über TLS für Verschlüsselung und somit Vertraulichkeit und Authentifizierung sorgt [45]. So ist es möglich DNS Daten vom Nameserver zu beziehen ohne, dass Angreifer diese mitlesen können.

DANE

Das Konzept von TLS basiert auf dem Vertrauen in die Public-Key-Infrastruktur und somit in Certificate Authorities. Eine kompromittierte Certificate Authority hebelt die Sicherheit der Technologie aus, da

ein Angreifer so z.B. Domain-Zertifikate und somit die Serveridentität fälschen könnte. Um die Schwachstelle von Certificate Authorities zu umgehen, kann DNS-based Authentication of Named Entities (DANE) genutzt werden. Hierbei handelt es sich um einen eigenständigen Standard, der auf DNSSEC aufbaut und für eine konkrete Verknüpfung eines öffentlichen Schlüssels an den entsprechenden Domain-Namen sorgt [46] [47]. Das heißt, DANE ermöglicht eine von der Certificate Authority unabhängige Überprüfung der Zertifikate direkt beim Inhaber einer Domain. Hierdurch wird eine höhere Sicherheit gewährleistet, da Zertifikate nicht mehr unbemerkt ausgetauscht werden können. Auch können über DANE-Zertifikate erstellt und Certificate Authority somit komplett gemieden werden.

Konkret nutzt DANE sogenannte TLS Authentication (TLSA)-Einträge in den DNS-Zonen. Diese Einträge enthalten Informationen über das Zertifikat sowie wie einen eindeutigen Fingerabdruck in Form eines Hash-Wertes. Da nur der Domain-Besitzer berechtigt ist, diese Einträge zu verwalten, kann eine Manipulation der Informationen der TLSA-Einträge ausgeschlossen werden. Sobald ein Client ein Zertifikat verifizieren möchte, wendet er sich nicht mehr an die Certificate Authority, sondern fragt den TLSA-Eintrag der Domain mit dem Hash-Wert des Zertifikats ab. Dieser Austausch von DNS-Informationen wird mit DNSSEC abgesichert. Das heißt, die Integrität der Daten wird gewährleistet. Anhand eines Vergleichs des empfangenen Fingerabdrucks mit einem Hash-Wert, der mit dem öffentlichen Schlüssel selbst berechnet wurde, kann die Authentizität des Servers festgestellt werden. Ist die Überprüfung erfolgreich, so kann, wie gehabt, die TLS-verschlüsselte Verbindung aufgebaut werden [47].

II.1.3.3 Entwicklung eines Datenmodells für den FlexHub

In diesem Arbeitspaket wurden die Anforderungen aus den Modellen für die Anwendungsfälle aus AP1 parallel mit den Projektpartnern zusammen aufgenommen und, zusammen mit den Vorerfahrungen, die die HAW Hamburg aus dem OS4ES-Projekt gewonnen hat, zu einem weitgehend IEC61850-kompatiblem Datenmodell weiterentwickelt werden. Das Datenmodell basiert auf einem *White and Yellow Page Modell*. Die White Pages beinhalten dabei die Basisinformation eines DER-Systems und geben Auskunft über die Identität des Anbieters. Dazu gehören Informationen über den Standort der Anlage, den Besitzer sowie Kontaktdaten. Die Yellow Pages beschreiben die Energiedienstleistungen, die ein DER-System anbieten kann. Dabei besteht eine 1:n-Beziehung zwischen den White und Yellow Pages, was bedeutet, dass eine Anlage mehrere, unterschiedliche Energiedienstleistungen anbieten kann. Ausführliche Informationen über das Datenmodell finden sich in dem angehängten Dokument „Flexhub – White and Yellow Pages“ (A4), dort ist das Datenmodell u.a. als UML-Klassendiagramm beschrieben und es findet eine detaillierte Erläuterung der enthaltenen Daten und Felder statt.

Die erzielten Ergebnisse zu dem Datenmodell und das Marktdesign, welches vom Datenmodell abgebildet wird, wurde im ew – Magazin für die Energiewirtschaft im August 2020 publiziert [17].

Besonders hervorzugeben bei dem Datenmodell ist das Verständnis von Flexibilität, dass durch dieses abgebildet wird: Der FlexHub bildet einen opportunistischen Markt für die Flexibilität von Endkunden

ab. Dabei werden für das flexible Ladeverhalten der Endkunden die FlexOffer (Bezeichung der Yellow Page) erstellt. Ein FlexOffer beinhaltet im Kern folgende Informationen:

- Benötigte Menge an Energie.
- Minimale und maximale Ladeleistung des Anschlusses.
- Der Zeitraum in dem die Energie bezogen werden soll. Dieser Zeitraum beginnt mit dem frühestmöglichen Beginn des Ladevorgangs und endet mit dem Zeitpunkt, zu dem die komplette Ladung erfolgt sein muss.

Neben diesen Kerninformationen, die sich auf den eigentlichen Ladevorgang beziehen, beinhaltet ein FlexOffer u.a. noch den Angebotszeitraum. Also der Zeitraum, in dem das Angebot gültig ist und gebucht werden kann. Weitere wichtige Informationen sind der Preis für die Verschiebung bzw. Steuerung der Energie (in €/kWh), der Verweis auf die Flexibilitätsressource (White Page) und den Anbieter.

Wird ein FlexOffer von einem Netzbetreiber gebucht, so kann dieser den Verbrauch im Rahmen des Angebots steuern. Er kann also angeben, wann im Flexibilitätszeitraum, mit welcher Leistung geladen werden soll. Dabei ist sicherzustellen, dass die minimale Leistung nicht unterschritten und die maximale Leistung nicht überschritten wird. Die Ladung erfolgt, bis die benötigte Energiemenge bezogen wurden. Die Steuerung eines FlexOffers erfolgt über Fahrpläne, die der Buchende über den FlexHub zur Ressource sendet.

Wenn ein FlexOffer nach Ablauf des Angebotszeitraums nicht gebucht wurde, obliegt es dem Anbieter der Flexibilität (i.d.R. der IoT-Anbieter), dafür zu sorgen, dass entsprechend geladen wird. Die ungesteuerte Ladung kann dabei frühstens nach Ablauf des Angebotszeitraums beginnen und sollte spätestens bis zum Ende des angegeben Flexibilitätszeitraums erfolgt sein.

Flexibilität bedeutet im Kontext des FlexHubs als immer das Verschieben eines Ladevorgangs innerhalb der Parameter, die der Endkunde festgelegt hat (oder in seinem Auftrag von einer Software bestimmt wurde) und **nicht** das Abregeln oder Limitieren des Ladevorgangs.

Aufbauend auf den Informationsflüssen werden in diesem AP die zugehörigen Daten identifiziert, die sowohl im hierarchischen Modell als auch im Blockchain-Modell benötigt werden.

Das Ziel aus Sicht der Fraunhofer-Institute ist die Entwicklung eines Datenmodells für Transaktion in dem blockchain-basierten FlexHub.

Angelehnt an den vom HAW entwickelten White- und Yellow Pages wurden die Datenmodellen für die Energie Management System (EMS) / Dezentrale Energie Ressource (DER) und die Verteilnetzbetreiber (VNB) modelliert. Bedingt durch die Natur der Solidity Programmiersprache, insbesondere die verfügbaren Datentypen, gibt es Abweichungen vom ursprünglichen Modell. Die später in den Smart Contracts verwendeten EMS / DER und VNB-Modellen sind in Abbildung 17 und Abbildung 18 dargestellt.

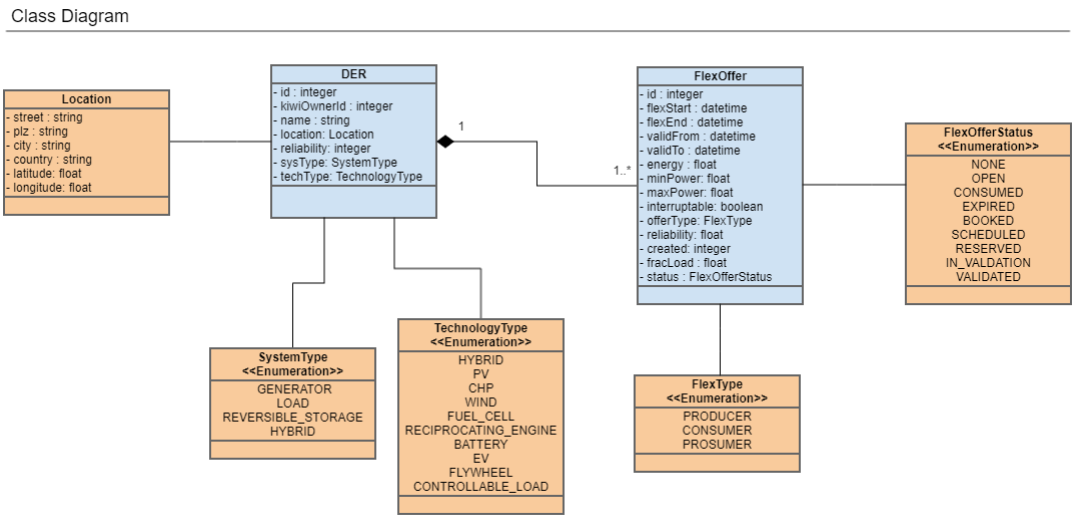


Abbildung 17: DER- und FlexOffer Klassendiagramm

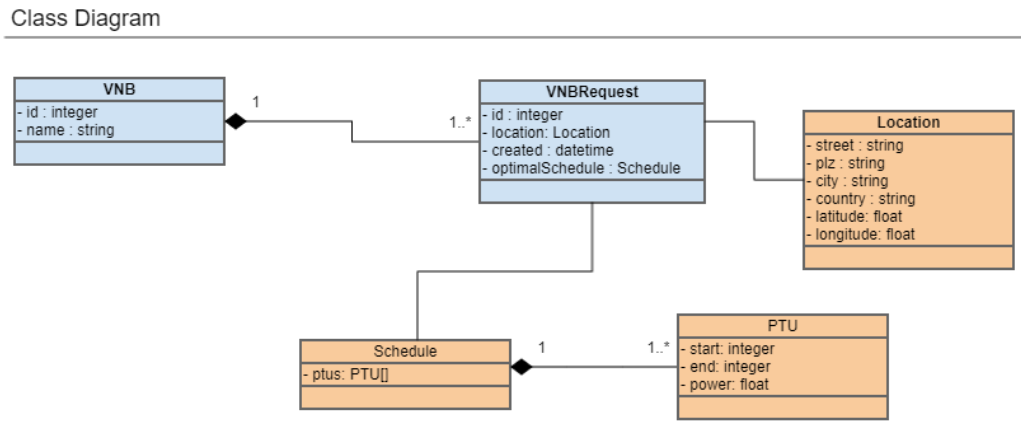


Abbildung 18: VNB- und VNBRequest Klassendiagramm

II.1.4 Arbeitspaket 4: Architektur des FlexHubs und Entwicklung technischer Prototypen

In AP 4 werden die beiden technischen FlexHub Architekturalternativen – die eines verteilten, hierarchischen, auf Domänen basierenden verteilten FlexHubs einerseits und die eines auf der Blockchain-Technologie basierenden FlexHubs andererseits - im Hinblick auf die in AP 1 definierten Anwendungsfälle und die Ergebnisse aus AP 3 untersucht und bewertet. Ziel war es zu entscheiden, welche der

beiden Architekturansätze für den FlexHub zu präferieren ist, weiterentwickelt und in der FlexHub Plattform implementiert wird.

II.1.4.1 Verteilte Architektur für einen FlexHub

II.1.4.1.1 Architektur für einen hierarchischen FlexHub

Im Rahmen dieses Arbeitspaketes wurde von der HAW Hamburg eine Architektur für ein verteiltes bzw. hierarchisches Flexibilitätenregister mit Marktfunktionen entwickelt. Hierarchische Architekturen sind eine spezielle Klasse von verteilten Systemen, die komplette Systeme als eine hierarchische Struktur betrachten, bei der ein Softwaresystem aus verschiedenen logisch unterteilten Modulen oder Subsystemen besteht, die hierarchisch auf unterschiedlichen Leveln angeordnet sind. Im Rahmen des OS4ES Projekts für ein hierarchisches, domänenbasiertes und verteiltes Datenregister konzipiert [16]. Dessen Implementierung allerdings aus zeitlichen Gründen nicht verteilt erfolgt ist. Die hierarchische Verteilung bei dieser Architektur ergab sich aus der hierarchischen Anordnung der Daten, der Energiedienstleistungen, die hierarchisch nach ihrem Erbringungsort in einer verteilten Datenstruktur abgelegt wurden. Die Struktur orientiert sich dabei an dem Domain Name Service (DNS) zur Adressauflösung im Internet. Eine Energiedienstleistung die z.B. in Hamburg angeboten wird, ist entsprechend hierarchisch in der Struktur EU-DE-NORTH-HAMBURG abgelegt. Eine Ressource in München findet sich entsprechend abgelegt in der Struktur EU-DE-SOUTH-MUNICH.

Erste Überlegungen bei der Antragsstellung orientierten sich daher an diesem Ansatz und planten eine ähnliche Struktur im Rahmen des FlexHubs umzusetzen. Dabei ging es insbesondere um die hierarchische Strukturierung der Daten und das entsprechende Abspeichern in hierarchisch organisierten Datenbanken. Während der Projektphase beschäftigten sich erste Architekturüberlegungen zunächst mit den Abgrenzungen der Domänen zwischen den unterschiedlichen Akteuren und Projektpartnern (siehe Abbildung 2.) Der FlexHub wurde hier als ein verteiltes System gesehen, dass die unterschiedlichen Komponenten, die zur Umsetzung der Anforderungen aus den Anwendungsfällen, verbindet.

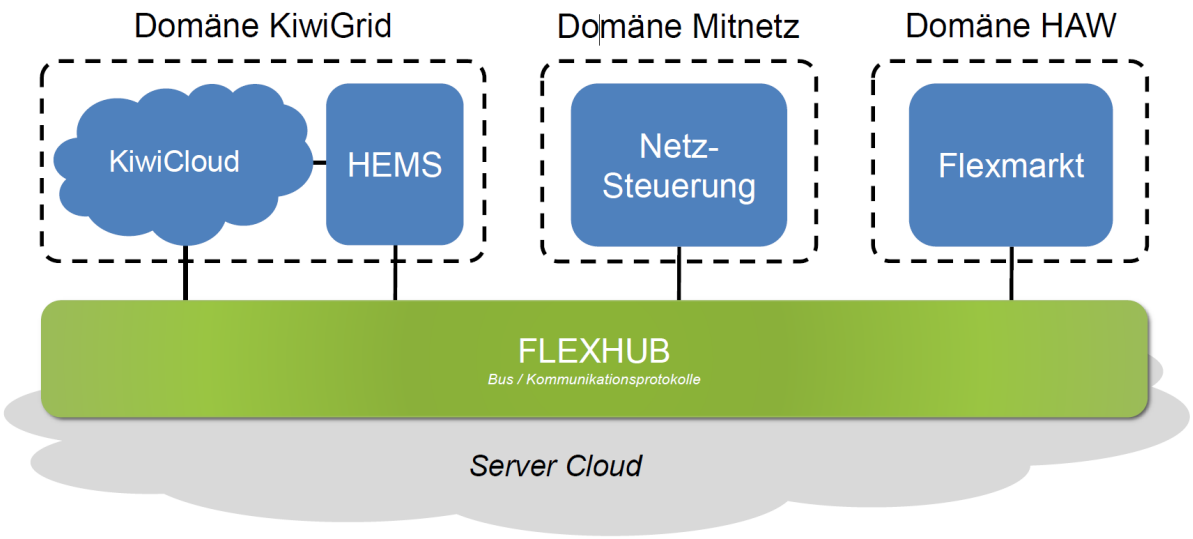


Abbildung 19: Erste monolithische Architektur zur Domänenabgrenzung.

Die weitere Ausgestaltung erfolgte nun insbesondere im Hinblick auf den zu realisierenden Flexmarkt. Hier wurde sich zunächst auf die Analyse einer effizienten Datenhaltung konzentriert. Dabei wurde sich für Einsatz von sogenannten NoSQL Datenbanken entschieden. Im Gegensatz zu den klassischen relationalen Datenbanken verfolgen sie einen nicht-relationalen Ansatz und benötigen daher keine festgelegten Tabellenschemata und versuchen Join-Operationen zu vermeiden. Dadurch ermöglichen sie eine horizontale Skalierung. D.h. sie sind insbesondere für datenintensive Applikationen geeignet und werden oft im Umfeld von Big Data verwendet. Viele NoSQL-Implementierungen unterstützen verteilte Datenbanken mit redundanter Datenhaltung auf vielen Servern, beispielsweise unter Nutzung einer verteilten Hashtabelle. Damit können die Systeme einfach erweitert werden sowie Ausfälle einzelner Server überstehen. Durch diese Eigenschaften von NoSQL Datenbanken werden die Anforderungen eine explizit hierarchischen Datenhaltung negiert. Im Folgenden wurde daher auf die explizit hierarchische Strukturierung der Datenhaltung verzichtet.

Stattdessen wurde sich für einen modernen Microservice-Ansatz entschieden, bei dem die unterschiedlichen Komponenten (Microservices) verteilt und ggf. auch hierarchisch skaliert ausgeführt werden können. Microservices sind ein Architekturmuster der Informationstechnik, bei dem komplexe Anwendungssoftware aus unabhängigen Prozessen generiert wird, die untereinander mit sprachunabhängigen Programmierschnittstellen kommunizieren. Die Dienste sind weitgehend entkoppelt und erledigen eine klar definierte Aufgabe, dabei folgen sie dem Unix-Prinzip: „Do One Thing and Do It Well“. So ermöglichen sie einen modularen Aufbau von Anwendungssoftware. Abbildung 3 stellt diese Architektur zentriert auf das Flexibilitätenregister dar. Aus Gründen der Übersicht werden dabei die Komponenten, die von Kiwigrid entwickelt werden zur KiwiCloud zusammengefasst. Eine detaillierte Darstellung und Beschreibung der enthaltenen Komponenten findet sich in der entsprechenden Komponentenbeschreibung von Kiwigrid.

In dieser Phase in der Entwicklung trug das verteilte Flexibilitätenregister den Namen IRES (Intelligent Registry for Energy Services). Das IRES-Backend ist die zentrale Komponente der Plattform und übernimmt die Aufgaben des Flexibilitätenregisters, wie sie in den Anwendungsfällen 1, 2 und 3 beschrieben sind. Es stellt eine über HTTPS abgesicherte REST-Schnittstellen zur Verfügung. Als Datenaustauschformat wird JSON verwendet. Das Backend ist zustandslos, sodass es entsprechend einfach horizontal skaliert werden kann. Das Backend stellt REST-Schnittstellen zur Verfügung über die ein Aggregator Flexibilitätsressourcen (DERSysteme) als White Pages im System registrieren kann. Auf Basis dieser White-Page-Informationen kann der Aggregator Flexibilitätsangebote (Yellow-Pages) zur Vermarktung der potenziellen Flexibilitäten der DERSysteme erzeugen. Dabei kann es sich z.B. um den flexiblen Ladezeitraum eines E-Autos handeln. Über weitere Schnittstellen kann der Aggregator seine Angebote anpassen oder löschen, sofern diese nicht bereits gebucht sind.

Nachfrager von Flexibilitäten (Flex-Anfrager in der Grafik, können z.B. Verteilnetzbetreiber zum Netzengpassmanagement sein) können über entsprechende REST-Schnittstellen nach Flexibilitäten suchen. Dabei können sie über Suchparameter, z.B. gezielt nach Flexibilitätsangeboten in einem bestimmten geographischen Bereich suchen. Weiterhin stellt das Backend eine REST-Schnittstelle zur Verfügung, um Flexibilitäten zu buchen. Diese Buchung kann optional von einem VNB als Kapazitätsmanager validiert werden (in diesem Fall erhält der VNB Kapa eine entsprechende Anfrage über den MQTT-Broker). Im Falle einer erfolgreichen Buchung wird der entsprechende Aggregator per MQTT-Push-Nachricht über die Buchung informiert. Für den VNB Kapa stellt das Backend in diesem Fall eine REST-Schnittstelle zur Verfügung, über die die Buchung bestätigt oder abgelehnt werden kann. Nach einer erfolgreichen Buchung kann der Flex-Anfrager per REST-Schnittstelle Fahrpläne für die gebuchten Anlagen an das Backend schicken. Diese Fahrpläne werden per MQTT an den entsprechenden Aggregator weitergeleitet. Optional können statt MQTT auch E-Mails als Push-Nachrichten versendet werden.

Die Datenbank hält den Zustand der IRES-Plattform. Hier sind die White- und Yellow-Pages, sowie die erfolgten Buchungen mit entsprechenden Statusinformationen hinterlegt.

MQTT-Broker - MQTT (Message Queuing Telemetry Transport) ist ein ISO standardisiertes (ISO/IEC PRF 20922) Publish/Subscribe Nachrichtenprotokoll. MQTT setzt auf den TCP/IP Protokollen auf und eignet sich aufgrund deines geringen *footprints* vor allem in Anwendungsfällen, in denen die Netzwerkbandbreite limitiert ist. MQTT wird daher oftmals in IoT-Szenarien eingesetzt. Ein MQTT-System besteht aus mehreren Clients, die mit einem Server kommunizieren. Im MQTT-Kontext wird der Server oftmals als Broker bezeichnet. Das IRES-Backend verbindet sich daher, ebenso wie der Aggregator und der VNB Kapa, als Client mit dem MQTT-Broker. Über diesen werden vom Backend unterschiedliche Arten von Push-Nachrichten verschickt (u.a. Buchungsbestätigungen für den Aggregator, Validierungsanfragen für den VNB Kapa und Fahrpläne für den Aggregator).

Bei dem Mailer handelt es sich um eine Komponente, die über das SMTP (Simple Mail Transfer Protocol) E-Mails als Push-Nachrichten verschickt. Der Mailer kann dabei alternativ zum MQTT-Broker verwendet werden.

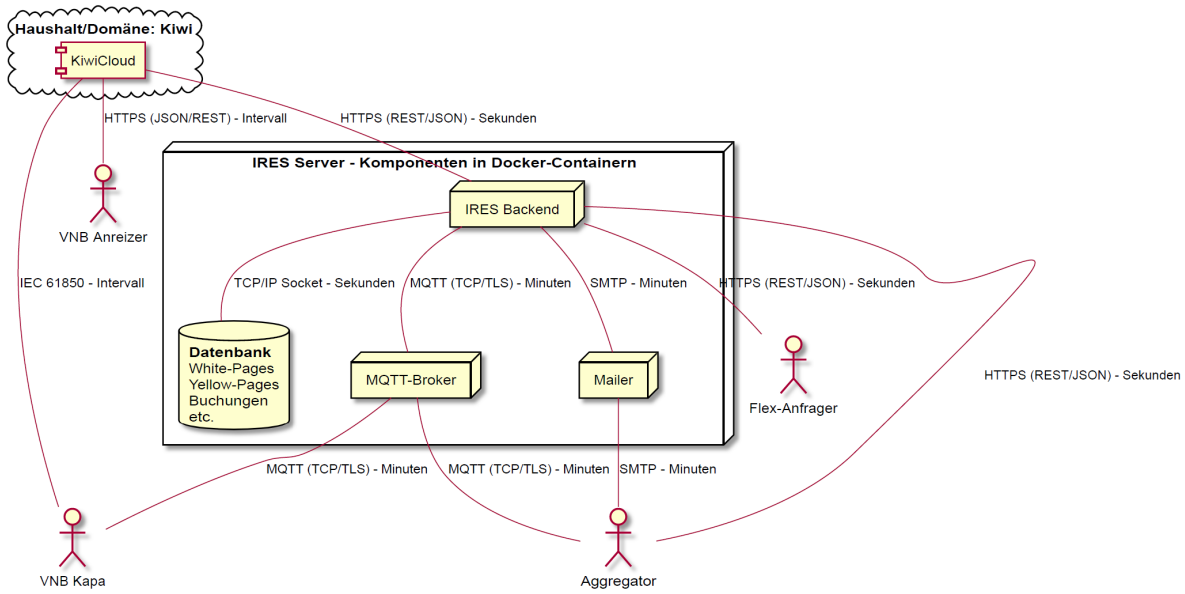


Abbildung 20: Architektur des Flexibilitätenregisters in einem verteilten Flexhub

II.1.4.1.2 Vergleich der Architekturalternativen

Die beiden Architekturalternativen für einen hierarchischen bzw. verteilten und einen Blockchain-basierten FlexHub wurden anhand von funktionalen und nichtfunktionalen Anforderungen bewertet und verglichen.

Die Auswahl der nichtfunktionalen Eigenschaften, die für den Vergleich der beiden Architekturalternativen herangezogen werden, basiert auf dem ISO 25010 Standard. Der ISO 25010 Standard [18] ist eine internationale Norm für Qualitätskriterien von Softwaresystemen. Dabei wird Software-Qualität anhand von Merkmalen definiert, die am fertigen System sichtbar werden. Der ISO 25010 Standard ist dabei ein direkter Nachfolger der ISO 9126 Norm. Die entscheidende Neuerung dabei ist die Hinzunahme von Sicherheit und Kompatibilität als Hauptkategorien, da diese seit der Definition der ISO 9126 an Bedeutung gewonnen haben.

Im Folgenden erfolgt tabellarische Übersicht der Qualitätsmerkmalen nach diesem ISO-Standard.

Tabelle 3: Qualitätsmerkmale nach ISO 25010

Qualitätsmerkmal	**Bezeichnung nach ISO**	**Beschreibung**	**Unterkategorien**
Funktionalität	Functional suitability	Die Art und Weise, wie die gewünschte Funktionalität geliefert wird	Vollständigkeit, Korrektheit, Angemessenheit
Effizienz	Performance efficiency	Stellt die Leistung des Systems in Relation zum	Zeitverhalten, Ressourcenverbrauch,

		Ressourcenverbrauch dar (Performance des Systems)	Kapazität
Kompatibilität	Compability	Fähigkeit zum Austausch von Daten mit anderen Systemen oder Komponenten	Ko-Existenz, Interoperabilität
Benutzbarkeit	Usability	Grad der Nutzbarkeit für definierte Benutzer, vorher bestimmte Ziele mit dem System effizient und zufriedenstellend erreichend	Erlenbarkeit, Bedienbarkeit, Schutz vor Fehlern des Benutzers und Barrierefreiheit
Zuverlässigkeit	Reliability	Erbringung der Leistung des Systems unter bestimmten Bedingungen über einen definierten Zeitraum	Reife, Verfügbarkeit, Fehlertoleranz, Wiederherstellbarkeit
Sicherheit	Security	Fähigkeit die Daten und das System so zu schützen, dass beabsichtigte und auch unbeabsichtigte Zugriffe erkannt und abgewehrt werden	Vertraulichkeit, Integrität, Nachweisbarkeit, Ordnungsmäßigkeit, Authentizität
Portabilität	Portability	Gibt an, wie einfach oder komplex es ist ein System in einer Umgebung zu betreiben und wie sehr das System in der Lage ist, sich auf wechselnde Gegebenheiten dort anzupassen	Anpassbarkeit, Installierbarkeit, Austauschbarkeit

Die funktionalen Eigenschaften ergeben sich aus den drei Anwendungsfällen, die im Rahmen des Flex-Hub Projekts umgesetzt werden sollen (siehe Anhänge A1 – A3).

Grundsätzlich haben die Analysen gezeigt, dass beide Architekturalternativen in der Lage sind die Logik aus den Anwendungsfällen abzubilden und somit die funktionalen Anforderungen erfüllen können. Dabei ist allerdings anzumerken, dass die Informationsflüsse, wie sie in den Anwendungsfällen beschrieben sind, so direkt nur mit der hierarchisch/verteilten Architektur abbildbar sind. Für den Peer-to-Peer Ansatz auf dem eine Blockchain basiert, sind gewisse Anpassungen an den Informationsflüsse erforderlich. Die grundsätzlichen Anforderungen, die sich durch die Bereitstellung und den Abruf von Flexibilität ergeben, sind aber auch mit der Blockchain-Architektur umsetzbar.

Auch die nichtfunktionalen Anforderungen konnten von beiden Architekturalternativen in einem ausreichenden Maß erfüllt werden. Dabei wurde das Kriterium der Benutzbarkeit nachrangig betrachtet, da der Fokus nicht auf der Benutzung durch Endanwender liegt, für die dieses Kriterium von größer Bedeutung ist, sondern um die Anbindung an die entsprechenden Systeme von Kiwigrid und Mitnetz. Die Anforderungen im Bereich der Sicherheit konnten, durch Umsetzung der Handlungsempfehlungen, wie sie vom FKIE entwickelt wurden, für beide Alternativen sichergestellt werden. Auch der zuverlässige und effiziente Betrieb konnte für beide Alternativen sichergestellt werden. Dem Blockchain-basierten Ansatz wohnt inhärent durch die Peer-to-Peer Architektur und dem damit verbundenen mehrfachten Auftreten der relevanten Knoten eine Zuverlässigkeit inne und für die verteilte/hierarchische Architektur kann diese durch eine Replikation der Daten und eine horizontale Skalierung und Verteilung der Zustandslosen Microservice-Komponenten sichergestellt werden. Im Bereich der Effizienz

erlaubt diese Möglichkeit der horizontalen Skalierung, das einfache und nahtlose Hinzufügen weiterer Komponente, wenn die Effizienz aufgrund zu viele Anfragen herabsinkt. Die Blockchain-basierte Lösung kann durch die Wahl eines geeigneten Konsensus-Algorithmus eine ähnliche Effizienz sicherstellen. Wichtig ist an dieser Stelle, dass hier kein Proof-of-Work (POW) Ansatz wie z.B. bei Bitcoin gewählt wird, sondern z.B. ein Proof-of-Stake Ansatz.

Abschließend konnte bewertet werden das beide Ansätze grundsätzlich für die Umsetzung eines verteilten FlexHubs geeignet sind. Der hierarchische Ansatz zeigte dabei jedoch eine höhere praktische Anwendungsrelevanz und ermöglichte insbesondere durch den Einsatz weit verbreiteter REST-Schnittstellen mit JSON-Datenmodelle eine schnellere Anbindung an die Systeme der Kiwigrid, sodass hier Vorteile im Bereich der Kompatibilität entstanden. Außerdem bildet die hierarchische Architektur die Prozesse und Datenhaltung aus regulatorischer Schicht einfacher und nachvollziehbarer ab, sodass diesem Ansatz im Hinblick auf eine zukünftige Regulation Vorteile eingeräumt wurden.

In diesem AP werden die Konzepte für den hierarchischen FlexHub sowie die blockchainbasierte Architektur mit Blick auf IT-Sicherheitsanforderungen untersucht und notwendige Maßnahmen vorgeschlagen. Die konzeptuelle Entwicklung der blockchainbasierten Architektur für FlexHub wird so gestaltet, dass sie sowohl die Anwendungsfälle als auch die Informationsflüsse und das Zugriffsmodell geeignet unterstützen.

Für die Fraunhofer Institute ergeben sich dabei die folgenden Aufgaben:

1. Verfahren zur Validierung, zum Mining, zum Konsensus-mechanismus und zur Anonymität von Daten konzipieren
2. Konzept zur Integration von externen Datenquellen erarbeiten
3. Auktionsmechanismen definieren und erproben
4. Struktur für Blockchain-Netzwerk entwickeln
5. Unterstützung bei der Konzeption beider Ansätze
6. Analyse zentraler Internetdienste sowie Recherche zu Konzepten von Hochverfügbarkeitslösungen mit dem Ziel diese im FlexHub einzusetzen.

Ergebnisse:

Zu 1-4:

Quorum als Distributed Ledger Technologie

Quorum hat sich für das FlexHub Vorhaben als geeignete open-source Blockchain Technologie erwiesen. Quorum ist ein auf Ethereum basiertes Distributed Ledger Protokoll, das für konsortiale Distributed Ledger Technologien (DLT) entwickelt wurde [48]. Bedingt durch die genehmigungsbasierte Natur

der privaten Netzwerke bietet Quorum Konsensalgorithmen an, die nicht auf leistungsbasierten Beweisen, e.g. Proof-of-Work, oder Besitz von Kryptowährungen, e.g. Proof-of-Stake, beruhen. Stattdessen stehen andere Konsensalgorithmen zur Verfügung, die Byzantine Fault Tolerance oder Crash Fault Tolerance gewährleisten. Zudem unterstützt Quorum private Transaktionen, wodurch ausschließlich der Absender und die jeweiligen Empfänger der Transaktion die Transaktionsinhalt unter sich teilen dürfen.

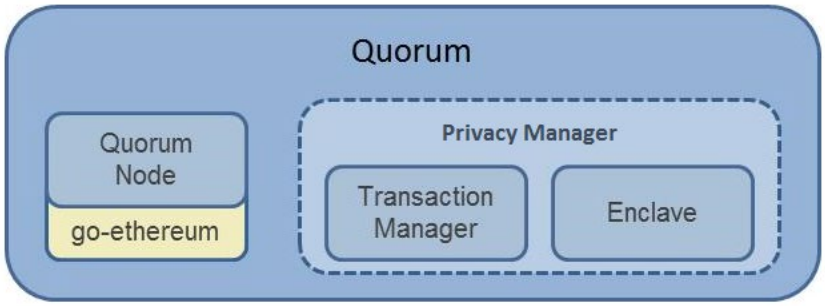


Abbildung 21: Darstellung der Komponenten eines Quorum Knotes

Da Quorum auf Ethereum basiert, hat Quorum die gleiche Grundfunktionalität wie Ethereum [49]. Wie in einem öffentlichen DLT, werden Transaktionen, die auf einem Knoten ausgeführt sind, je nach Konsensalgorithmus, in ein oder mehrere Blöcke zusammengeführt. Die neuen Blöcke erreichen die anderen Knoten im Netzwerk, die sich anschließend darüber einigen, ob die Blöcke einzeln in das bestehende Ledger (Blockkette) anhängen, vorausgesetzt, dass die Transaktionen gültig sind.

Der Genehmigung-Mechanismus („permissioning“) in Quorum bedeutet, dass neue Knoten nur dann im Netzwerk wirksam teilnehmen können, wenn sie als geprüfte Knoten von den Netzwerk-Teilnehmern zugelassen sind. Darum ist der darunterliegende Peer-to-Peer Layer so angepasst, dass Verbindungen nur zwischen genehmigten („permissioned“) Knoten erlaubt sind.

Aufgrund seiner privaten Natur gibt es in Quorum keine Kryptowährung und deshalb spielt der Gaspreis keine Rolle. Zudem gibt es in den konsortialen DLTs keine Miners, die rechnerische Arbeit leisten und deshalb fällt der Bedarf nach Belohnungen aus. Die Gas-Attributen in dem Protokoll bleiben aus Rückwärtskompatibilitätsgründen weiter bestehen.

Konsensalgorithmen

Quorum bietet aktuell drei Konsensalgorithmen an: Raft, Istanbul BFT und Clique POA.

Raft wird ausschließlich in privaten DLTs benutzt und wird in Netzwerken eingesetzt, wo eine beträchtliche Zahl von Knoten ausfallen können, d.h. eine Crash Fault Tolerance (CFT) erforderlich ist [50]. Tatsächlich ist das Netzwerk bei einer einfachen Mehrheit der Knoten robust. Zudem zeichnet sich Raft durch die schnellere Generierung von Blöcken (Größenordnung Millisekunden statt Sekunden bei

Ethereum) und die Finalität von Transaktionen, d.h. keine Forks dürfen in dem Ledger entstehen, aus. (Ein Fork entsteht, wenn zwei Knoten nahezu gleichzeitig Blöcke in das Ledger angehängt haben, wodurch eine Gabelung gebildet ist). Zudem werden keine leeren Blöcke generiert. Für weitere Details zur Funktionsweise von Raft siehe bitte die FlexHub Blockchain Spezifikation [26].

Istanbul BFT erbt von dem Practical Byzantine Fault Tolerance Algorithmus (PBFT). Byzantine Fault Tolerance (BFT) ist der Schutzmechanismus gegen böswillige Knoten, die gegen das gemeinsame Ziel des Netzwerks agieren wollen. Das System kann höchstens mit knapp einem Drittel fehlerhafter Knoten in einem Netzwerk umgehen [51]. Jede Runde wird ein Block generiert, indem die Validierungsknoten verschiedene Zustände durchlaufen. Wie im Falle von Raft gewährleistet Istanbul BFT die Finalität der Transaktionen, indem keine Forks in dem Ledger entstehen können und neue Blöcke in festen Zeitabständen (bekannt als Blockzeit) angehängt werden. Eine typische Blockzeit ist 5 Sekunden. Eine detaillierte Beschreibung des Modus Operandi von Istanbul BFT ist in [26] zu finden.

Clique Proof-of-Authority (PoA) ist ein Konsensalgorithmus, bei dem autorisierte Knoten den nächsten Block signieren dürfen. Die autorisierten Knoten entscheiden hierbei über ihre Nachfolger. Sie dürfen die Adresse eines Knotens für den Kreis den autorisierten Knoten vorschlagen oder ihn eben vom Kreis ausschließen. Clique PoA erfordert, dass die einfache Mehrheit der autorisierten Knoten vertrauenswürdig ist. Der Algorithmus weist eine ähnliche Sicherheit auf wie die PBFT Algorithmen, z.B. Istanbul BFT. Darüber hinaus skaliert Clique besser als die PBFT Algorithmen, da nur eine Runde Nachrichten ausreicht, um eine Entscheidung zu treffen, statt drei Runden bei PBFT.

Permissioning

Knoten lassen sich im Quorum Netzwerk nicht unbedingt frei untereinander verbinden. Dafür sorgt die Permissioning Einstellung. Für jeden Knoten lässt sich steuern, welche andere Knoten auf ihn Zugang haben und wiederum auf welchen anderen Knoten eine Verbindung möglich ist.
Zudem existiert ein fortgeschrittenes Modell zum Gestalten der Netzwerk-Erreichbarkeit, falls die Netzwerk-Partner in verschiedenen Bereichen aufgegliedert sind. Darin lassen sich die gegenseitigen Verbindungen in Form eines Baums spezifizieren. Der Netzwerk Admin steht oben in der Hierarchie und bestimmt über die Organisationen und Unterorganisationen. Diese wiederum entscheiden über die Zugriffsberechtigungen restlicher Knoten und deren Konten. Dieses Modell ist mittels Smart Contracts umgesetzt.

Governance Konsortialblockchains

Eine Konsortialblockchain benötigt eine Satzung, die das Ziel, den Zweck und die Verpflichtung der Mitglieder definiert. Zudem legt die Satzung die Beschlussverfahren fest, worüber die Mitglieder Off-Chain (physikalisch) oder On-Chain (online) abstimmen können. Die Verfahren zu Eintritt und Austritt der Mitglieder werden ebenfalls definiert, sowie die Pflichten eines Vorstands, der sich um die administrativen Aufgaben im Netzwerk kümmern muss.

Ein Mitglied betreibt einen Knoten, der als Validator oder Nicht-Validator in Netzwerk agiert. Validator-Knoten sind Mitglieder, die auf einem Server einen Quorum Knoten betreiben. Sie nehmen am

Konsensmechanismus teil, sie leisten rechnerische Arbeit, um Transaktionen zu validieren und hängen Blöcke in das Ledger an. Zudem dürfen sie über die Anträge in Konsortium abstimmen. Basierend auf den im Projekt definierten Anwendungsfällen, den unabhängigen Verteilnetzregionen und den Hardware-Anforderungen qualifizieren sich ausschließlich Verteilnetzbetreiber als Validator-Knoten. Nicht-Validator Knoten betreiben ebenfalls jeweils einen Quorum Knoten, der mit dem Netzwerk kommuniziert. Sie dürfen jedoch nicht beim Governance teilnehmen.

Das Governance Modell besteht aus Mitentscheidungsrechten, Verantwortung und Anreizen. Jedes Mitglied darf Anträge definieren und sie im Netzwerk als Vorschläge beantragen. Die Validatoren dürfen anschließend über die vorgeschlagenen Anträge abstimmen. Falls ein Validator-Knoten gegen die Satzung verstößt, werden ihm Sanktionen verhängt, die von einer Warnung bis zu einer temporären Sperrung reichen können. Bei wiederholten Verstößen ist der Ausschluss des Knoten zur Wahl gestellt und die Mitglieder dürfen darüber abstimmen. Folgende Aktionen können einen Verstoß darstellen:

1. Validator-Knoten ist offline für mehr als X Tage/Wochen/Monaten.
2. Validator-Knoten verbreitet Spam im Netzwerk.
3. Validator-Knoten verhält sich nicht im Einklang mit der Satzung.

Mögliche Anreize können folgende sein:

- Mitglieder gestalten das Netzwerk, indem sie Anträge zum Verwalten und Weiterentwicklung des Netzwerks einreichen und sie darüber abstimmen dürfen.
- Mitglieder setzen ein Zeichen für die Energiewende, indem sie zeigen, dass die Energiestrommärkte sich dezentral organisieren und verwalten lassen, ohne sich von zentralen Aggregatoren abhängig zu machen.
- Das Netzwerk ist demokratisch aufgebaut und die Mitglieder dürfen es weitergestalten.

Als Mitglieder des Netzwerks qualifizieren sich zunächst Verteilnetzbetreiber und Energie Management Systemen. Anträge zur Aufnahme eines neuen Mitgliedes sowie zum Ausschließen eines Mitglieds werden zur Wahl gestellt und die Mitglieder dürfen darüber abstimmen.

Der Vorstand wird bei einem physikalischen Treffen von den Mitgliedern gewählt. Das gewählte Mitglied wird das Amt ein halbes Jahr/1 Jahr innehaben. Der Vorstand verpflichtet sich, die abgestimmten Anträge umzusetzen. Das beinhaltet unter anderem die Aufnahme und Ausschluss Mitglieder, Verbesserungen/Aktualisierungen zum Netzwerk durchzuführen, Änderungen zum Governance Modell durchzuführen etc. Jedes Mitglied darf sich um den Posten des Vorstands bewerben, solange es im Netzwerk seit mehr als 1 Jahr Mitglied ist.

FlexChain Governance

Um ein ausgewogenes Governance-Modell zu finden, spielen folgende Faktoren eine entscheidende Rolle:

1. Handlungsspielraum eines Validators definieren

2. Anzahl der Validator-Knoten

3. Hardware-Anforderungen eines Validator-Knotens

4. Die Robustheit des Netzwerks im Falle ausfallender Knoten

5. Die Robustheit des Netzwerks im Falle böswilliger Knoten

6. Der Durchsatz des gewählten Konsensalgorithmus angesichts des entstehenden Transaktion Volumens

7. Die Skalierbarkeit des gewählten Konsensalgorithmus angesichts der wachsenden Anzahl des Knoten

Der Handlungsspielraum eines Validators ist die Zone, in der die definierten Anwendungsfälle abspielen. Da es in Projektrahmen um unabhängige Verteilnetz-Regionen handelt, innerhalb deren die definierten Abläufe stattfinden, ist es sinnvoll mindestens einen Validator-Knoten pro Verteilnetz zu wählen. Aktuell sind es ca. 900 Verteilnetzbetreiber in Deutschland aktiv (Stand 2019) [52], während die Anzahl der Dezentrale Energie Ressourcen beläuft sich auf über 2 Millionen (Stand Februar 2021) [53]. Eine Studie [54] hat gezeigt, dass ein Netzwerk mit 900 Knoten mit bestimmten Konsensalgorithmen (PoET auf Hyperledger Sawtooth) erfolgreich simuliert werden konnte.

Dezentrale Energie Ressourcen (DERs) müssen nicht zwingend eine direkte Vertretung im Validator-Kreis haben, denn Verteilnetzbetreiber berücksichtigen das Interesse der Energie Management Systemen (EMS), wenn Anreize wie Vermeidung von Netzengpässe gelten. Darüber hinaus sind die EMS i.d.R. nicht mit der erforderlichen Hardware ausgestatten, um Validator-Knoten zu betreiben.

Es ist vorteilhaft, wenn das entstehende Netzwerk nicht überlastet wird und damit die Hardware-Mindestanforderungen eines Validators verhältnismäßig gering bleiben, z.B. 2-kern CPU, 2 GB RAM und 100 GB Speicherplatz.

Es ist absehbar, dass Validator-Knoten ausfallen bzw. nicht immer verfügbar sein werden. Hier spielt das Crash Fault Tolerance eine wichtige Rolle, um das Netzwerk erhalten zu bleiben. Außerdem werden Verfügbarkeit-Mindestwerte festgelegt, um die Ausfallzeit der Validatoren zu minimieren, deren Überschreitung eine Strafe gegen die Validatoren bedeutet. Z.B. bei mehr als 3 Monaten Nichtverfügbarkeit würde ein Knoten mittels eines internen Punktesystems sanktioniert oder ggf. zum Ausschluss vorgeschlagen.

Darüber hinaus es ist nicht davon auszugehen, dass alle Validator-Knoten je nach Aktion das Interesse der Gemeinschaft immer unterstützen. Z.B. können manche Validatoren gegen eine Umformung des Netzwerks abstimmen, wenn das physikalische Netzwerk sich verändert hat. Deswegen wäre die Byzantine Fault Tolerance notwendig, um eine Marge bei der Durchsetzung der im Konsortium vorgeschlagenen Anträge zu bewahren.

Matching-Logik

Die Kern-Funktion des Flexibilitätsmarktes ist es, passende Flexibilitäts-Angebote für jeweils eine Flexibilitäts-Anfrage zu finden, die sich in der gleichen Region befinden. Da es sich um eine Optimierungsproblem handelt, wird es hier nicht versucht, die beste Lösung zu finden, die sonst eine exponentielle Komplexität erfordert, sondern wird es mit einem heuristischen Algorithmus mit quadratischer Komplexität vorangegangen, dessen gefundenen Lösung der Umstände entsprechend meistens zufriedenstellend ist.

Der Algorithmus verläuft wie folgt:

1. Flex-Angebote werden nach deren Start-Zeit ansteigend sortiert. Am besten werden die Flex-Angebote in eine Datenstruktur so hinzugefügt, dass sie vor der Anwendung der Matching-Funktion bereits sortiert sind und diese Operation zu keinem zusätzlichen Aufwand führt.
2. Die sortierte Liste der Flex-Angebote wird durchlaufen und überprüft, welche Flex-Angebot die folgenden Bedienungen erfüllt:
    (a) Nur Flex-Angebote werden berücksichtigt, die noch verfügbar sind, d.h. nicht bereits abgelaufen und nicht bereits gebucht sind.
    (b) Nur Flex-Angebote werden berücksichtigt, deren Zeitspanne mit der Zeitspanne der Flex-Anfrage überschneiden.
    (c) Falls die maximale Leistung des Flex-Angebots größer als die gewünschte Leistung der Flex-Anfrage ist, dann wird die Leistung der Flex-Anfrage für das Angebot verwendet. Wenn dies nicht der Fall ist, dann wird es mit der Leistung der Flex-Anfrage fortgeführt.
3. Nachdem ein Flex-Angebot die Bedienungen erfüllt hat, wird die vom Flex-Angebot abgedeckte Zeitspanne bei dem Flex-Anfrage als belegt markiert. Der Prozess geht ab dem 2. Schritt iterativ weiter bis kein Flex-Angebot mehr gefunden wird.

Für die ausgewählten Flex-Angeboten werden Fahrpläne fertiggestellt. Ein Fahrplan des Flex-Angebots ist eine oder mehrere Zeitspannen, die die obigen Auswahl-Bedienungen erfüllt hat, zusammen mit begleitenden Leistungswerten (kW).

Zu 5:

Die Projektpartner wurden bei ihren Entwicklungen im Bereich IT-Sicherheit bedarfsgerecht unterstützt. Der IT-Sicherheitskatalog aus AP3 enthält auch Best-Practices zum Thema IT-Sicherheit, die den Projektpartnern als Dokument zur Verfügung standen und genutzt werden konnten.

In den wöchentlichen Telefonkonferenzen wurden Herausforderungen für die Sicherheit eines verteilten FlexHubs durch Fraunhofer FKIE angesprochen. Basierend auf diesen Diskussionen und den definierten Anforderungen aus AP3 wurden weitere bilaterale Absprachen zwischen den Projektpartnern Fraunhofer FKIE sowie HAW (EnergieDock) und Kiwigrid durchgeführt. Das Ergebnis ist eine ausführliche Bedrohungsanalyse für den IRES-Flexibilitätsmarkt sowie eine tabellarische Auflistung von verschiedenen Bedrohungen für die von Kiwigrid entwickelten Softwarekomponenten und Schnittstellen. Beide Analysen sind dem Dokument „FlexHub (IT-Sicherheit)“ zu entnehmen.

Zwischen den beteiligten Fraunhofer Instituten wurden Herausforderungen in Hinblick auf Sicherheit und Privatsphäre für einen Blockchain basierten FlexHub abgestimmt. Der Fokus lag hierbei insbesondere auf einer sicheren Authentifizierung der Clients.

Zu 6:

Eine ausführliche Dokumentation zu Maßnahmen der Hochverfügbarkeit befindet sich in Abschnitt 4.4 des Dokuments „FlexHub (IT-Sicherheit)“. In diesem Abschnitt wird die Eigenschaft der Hochverfügbarkeit sowie grundlegende Konzepte zu dessen Umsetzung vorgestellt. Zudem wird eine Übersicht über Forschungsarbeiten in diesem Feld geschaffen. Die vorgestellten Forschungsarbeiten thematisieren die Realisierung von Hochverfügbarkeit durch virtuelle Maschinen, durch Container, und durch Hardware-Erweiterungen und Software-Lösungen. Zuletzt wird Hochverfügbarkeit in den Kontext des FlexHubs gerückt, in dem in Hinblick auf die Komponenten des IRES Flexibilitäts-marktes, Empfehlungen ausgesprochen werden, durch die eine höhere Verfügbarkeit geschaffen werden kann.

Die entsprechenden Empfehlungen lauten wie folgt:

Um eine hohe Verfügbarkeit der essenziellen Komponenten zu gewährleisten, sollte grundsätzlich die Mean Time To Failure (MTTF) – die Zeit bis zu einem Ausfall – erhöht und die Mean Time To Repair (MTTR) – die Zeit zwischen einem Ausfall und der Wiederaufnahme der Funktion – verringert werden [55]. Die Erhöhung der MTTF sollte im ersten Schritt durch eine Verstärkung der Fehlertoleranz bzw. der Robustheit erfolgen. Hierfür sollten Risikobewertungen des Systems herangezogen werden. Wenn bei den Bewertungen Angriffsmöglichkeiten auffallen, die die Verfügbarkeit beeinträchtigen, so sollte diesen durch geeignete Maßnahmen entgegnet werden. Neben einer Risikobewertung sind auch Penetration-Tests durch IT-Sicherheitsexperten möglich.

Weitere Punkte, durch die die Robustheit erhöht werden kann, sind die Nutzung von solider und hochwertiger Hardware. Da im Falle des IRES-Marktes nur das Host-System als Hardware-Komponente dient, sollte hier auf eine hohe Qualität gesetzt werden. Um diese zu verstärken, sollte das System regelmäßig gewartet und mit Updates versorgt werden.

Ein weiterer wichtiger Punkt zur Erhöhung der Robustheit ist eine gute Planung und Gewährleistung der Skalierbarkeit. Das heißt es sollte berücksichtigt werden, wie weit das System wachsen kann und wie sie sich die Auslastung erhöhen wird, um die nötigen Ressourcen jederzeit bereitzustellen.

Neben dem ersten Pfeiler – der Robustheit und Fehlertoleranz der Komponenten – sollte auch auf Redundanz gesetzt werden, um Ausfälle zu kompensieren und einen Single Point of Failure (SPoF) zu vermeiden. Das heißt, es sollte für alle essenziellen Komponenten mindestens ein Replikat existieren und genutzt werden. Hierbei wäre zum Beispiel bei dem Host-System die Nutzung eines Cluster-Systems denkbar.

Wird auch eine Verfügbarkeit unter extremen Umständen wie höherer Gewalt benötigt, so sollte die Redundanz auch geografisch getrennt erfolgen. So kann z.B. ein Server oder ein Cluster-System in

Finnland und einer/eines in Deutschland genutzt werden. Unabhängig wo die Server in Betrieb genommen werden, sollte auf eine redundante und separierte Stromversorgung geachtet werden, um den Ausfall eines der Stromkreise zu kompensieren.

Um die Software redundant zu gestalten, wird die Nutzung von Virtualisierung empfohlen, wie es auch in Forschungsarbeiten gezeigt wurde. Da sämtliche Software-Komponenten des IRES-Marktes in Docker Instanzen laufen und somit schon eine hohe Virtualisierung und Flexibilität vorhanden ist, bietet es sich an, auf eine Redundanz durch Container zu setzen. Hierbei kann Forschungsbeispielen gefolgt werden. So ist es möglich auf die Nutzung von Keepalived [56] oder CRIU [57] [58] zu setzen, um eine Checkpointing und Restoring Funktionalität in die Container-Umgebung zu integrieren.

Da es sich laut EnergieDock bei dem IRES-Markt um ein zustandsloses System handelt, sollte das Abspeichern und Wiederaufnehmen eines Container-Abbilds ohne größere Probleme funktionieren. Jedoch kann es unter Umständen nötig sein, eine Monitoring-Anwendung zu implementieren, die eine effiziente Fehlererkennung realisiert. Um die Implementierung einer solchen Anwendung zu umgehen, kann auf Docker Swarm [59] oder Kubernetes [60] zurückgegriffen werden.

Auch bei der Redundanz durch Virtualisierung sollte immer sichergestellt sein, dass mindestens eine Backup-Instanz für jede essenzielle Komponente vorhanden ist. Zudem sollte die Virtualisierung auch auf allen Hardware Servern erfolgen.

Da sich die Gesamtverfügbarkeit aus den Verfügbarkeiten aller beteiligten Komponenten ergibt, sollte keine essenzielle Komponente des Systems außer Acht gelassen und eine Kombination der oben aufgeführten Methodiken zur Erhöhung der Verfügbarkeit realisiert werden.

II.1.4.2 Technische Analyse eines hierarchischen FlexHubs und Proof-of-Concept Demonstrator

Die in AP4.1 konzipierte Architektur für einen hierarchischen FlexHub wurde in diesem Arbeitspaket als Prototyp für einen Proof-of-Concept Demonstrator umgesetzt und implementiert. Dabei floss insbesondere des Feedbacks der Konsortialpartner in die Umsetzung des technischen Prototyps ein. Der Prototyp konnte durch die Konsortialpartner erprobt werden und die Erfahrungen flossen wiederum als Ergebnisse in die Verfeinerung der Architektur und des technischen Prototyps ein. Der hierarchische Prototyp zeigte eine hohe praktische Anwendungsrelevanz und erhielt dabei von den Partnern für den Feldtest den technischen und regulatorischen Vorzug vor der Blockchain Lösung, sodass auf Grundlage dieser Architektur in AP6 eine Lösung implementiert und angebunden wurde.

Bei diesem AP geht es um die Begleitung der Proof-of-Concept-Entwicklung durch Prüfung sicherheitskritischer Code-Bereiche und Etablierung eines geeigneten Software-Entwicklungsprozess.

Die Begleitung umfasst die folgenden Aufgaben:

1. Abstimmung der Entwicklung
2. Bereitstellung von Coding-Guidlines und von Code-Reviews bei Bedarf der Partner
3. Unterstützung bei der Entwicklung sicherheitskritischer Komponenten

Ergebnisse:

Der IT-Sicherheitskatalog aus AP3 enthält ausführliche Best-Practices zum Thema IT-Sicherheit, die den Projektpartnern als Dokument zur Verfügung standen und genutzt werden konnten. Zudem fanden bilaterale Diskussionen über wichtige Maßnahmen zur IT-Sicherheit mit den Projektpartner Kiwigrid und EnergieDock statt. Die Ergebnisse dieser Diskussionen wurden partiell durch die Projektpartner selbst dokumentiert (siehe Bedrohungs bzw. Risikoanalysen im Dokument „FlexHub (IT-Sicherheit)“).

Zusätzlich und als Ersatz für die Code-Reviews erfolgte nach Absprache mit den Partnern und dem Projektträger eine IT-Sicherheitsanalyse von relevanten Kommunikationsprotokollen und -Komponenten. Eine detaillierte Beschreibung der Analysen und dem jeweiligen Vorgehen ist dem Dokument „FlexHub (IT-Sicherheit)“ zu entnehmen. Im Folgenden werden die Analysen und Ergebnisse nur grob vorgestellt.

61850-8-1

Die erste technische Analyse zielte auf das im Projekt FlexHub eingesetzte Protokoll IEC 61850 ab. Da eine prototypische Implementierung für die Spezifikation 61850-8-2 zum Zeitpunkt der Analyse nicht verfügbar war und sich noch in der Entwicklung durch die FGH befand, wurde auf eine offene Implementierung der Norm 61850-8-1 zurückgegriffen. Diese wurde mittels einer Sicherheitsanalyse namens Fuzzing untersucht. Bei einer Fuzzing-Analyse wird Software, die eine Eingabe erwartet, zahlreichen mutierten und randomisierten Testfällen unterzogen. Das Ziel ist es, Eingaben zu finden, welche die Software nicht verarbeiten kann und die somit einen Fehler aufzeigen. Diese Fehler können durch Angreifer beispielsweise dazu benutzt werden, die Software zum Absturz zu bringen oder durch das Überschreiben von Speicherbereichen Schadcode auf den Geräten auszuführen.

Konkret wurde das Open-Source-Projekt von „MZ-Automation“ [61] durch Fuzzing analysiert. Diese Implementierung des 61850 Protokolls ist öffentlich zugänglich und wird regelmäßig gepflegt. Deswegen besteht die Vermutung, dass sie auch in realen Hardware- oder kommerziellen Projekten eingesetzt wird.

In der wissenschaftlichen Literatur wurden bereits erfolgreich Fehler in IEC 61850-Implementierungen durch den Fuzzer Sulley [62] gefunden, die über das Netzwerk ausnutzbar sind. Da Sulley jedoch veraltet ist und nicht mehr gewartet wird [63], wurde für das Netzwerk-Fuzzing der Nachfolger Boofuzz [64] verwendet. Boofuzz vereinfacht das Fuzzing deutlich, indem das Framework wesentliche Aspekte wie die Absturzerkennung, das Zurücksetzen des Ziels nach einem Ausfall und die Aufzeichnung von Testdaten übernimmt und auch die einfache Erstellung Protokoll-spezifischer Pakete ermöglicht.

Bei der Analyse des Projekts wurden sowohl die Implementierung des MMS- als auch die des GOOSE- und SV Protokolls untersucht.

Die durchgeführte Analyse deckte gleich vier Fehler in der Implementierung auf – einen im MMS-Server, einen im GOOSE-Abonnenten und zwei im SV-Abonnenten. Die Entwickler wurden auf diese Fehler hingewiesen und die erzielten Ergebnisse auf der Konferenz „Energy Informatics“ im Jahr 2020 öffentlich präsentiert [65].

XMPP

Analog zur vorherigen Analyse erfolgte eine Analyse des XMPP Protokolls mit der Technik Fuzzing. Im Bereich der Energienetze wird XMPP vorrangig mit dem IEC 61850 Standard kombiniert [66] [67] [68]. Dies liegt insbesondere daran, dass die IEC 61850-8-2 Spezifikation den Austausch der Datenmodelle über XMPP definiert.

Im Rahmen des Projektes FlexHub soll die IEC 61850-8-2 Spezifikation und somit auch XMPP zur Kommunikation mit der Steuerbox genutzt werden. Da die Steuerbox eine bedeutende Rolle im FlexHub Konzept einnimmt, könnte eine Schwachstelle in dem XMPP Protokoll fatale Folgen haben. Denn je nach Schwere der Schwachstelle könnte ein Angreifer, die Steuerbox abschalten, Messwerte manipulieren oder sogar maliziöse Steuersignale verschicken. Somit ist die Sicherheit des Protokolls von hoher Bedeutung.Um diese zu testen, wurde eine Fuzzing-Untersuchung auf Openfire durchgeführt, einer weit verbreiteten Implementierung von XMPP.

Aufgrund der geringen Menge an Vorarbeiten, wurde für die Untersuchung auf eigens gesammelte Erfahrungen zurückgegriffen. Da bei dem IEC 61850 Standard Fehler mithilfe des Boofuzz Fuzzing Frameworks gefunden werden konnten. Kam dieses Framework auch für die Untersuchung von XMPP infrage. Allerdings existiert mit Fuzzowski [69] ein Fork von Boofuzz, der nutzerfreundlicher und interaktiver sein soll. Deswegen wurde dieser für die Untersuchung genutzt.

Die Fuzzing-Analyse führte zu einer Vielzahl an Testfällen, die vom Fuzzer als verdächtig eingestuft wurden und somit potentiell zu Fehlern führen könnten. Eine genauere Betrachtung zeigte jedoch, dass es sich dabei nur um Verbindungsabbrüche handelte. Das heißt, der Server brach zwar die Verbindung ab, blieb jedoch aktiv. Dies ist ein legitimes Verhalten, wenn der Server Pakete empfängt, die er nicht zuordnen kann. Somit konnten keine Abstürze des Openfire Servers herbeigeführt werden.

Diese Tatsache darf jedoch nicht gleich zu der Annahme führen, dass es keine Schwachstellen oder Bugs in der Implementierung gibt. Denn bei jeder Fuzzing-Untersuchung werden nur die Code-Regionen getestet, die von den generierten Inputs erreicht werden. In anderen Code-Regionen können dennoch Bugs existieren. Zudem ist es möglich, dass die Fuzzing-Untersuchung sogar Fehler gefunden hat, diese aber nicht feststellen konnte, weil sie zu keinem Absturz geführt haben.

Die in den beiden vorgestellten Analysen getätigten und beschriebenen Schritte können kombiniert und für den im Projekt entwickelten 61850-8-2 Softwarestack angewendet werden. Durch einen solchen Test kann die Sicherheit der Implementierung gesteigert werden.

PROLAN Steuerbox

Zuletzt erfolgte eine Analyse einer Steuerbox des Herstellers PROLAN. Die Analyse beinhaltete unter anderem das Fuzzing der Steuerbox. Ziel dabei war es zu prüfen, ob die Steuerbox Anfälligkeiten gegenüber unerwarteten und mutierten Paketen aufweist. Da eine Steuerbox für die Kommunikation zwischen SMGW und Erzeugungsanlagen und Lasten zuständig ist und auch Schalthandlung ausüben kann, hat sie eine wichtige Rolle in dem smarten Energienetz der Zukunft. Auch im Rahmen von Flex-Hub wird eine Steuerbox verwendet. Deshalb könnte jede gefundene Fehlfunktion kritisch sein.

Bei dem zu untersuchenden Produkt handelt es sich um eine PROLAN Steuerbox aus dem Jahr 2019 vom Typ STB-120N-2MD0-S0A0-R40S.

Der erste Schritt der Analyse der PROLAN Steuerbox bestand aus einem Nmap Scan. Bei Nmap handelt es sich um einen freien Netzwerkscanner zum Scannen und Auswerten von Hosts in einem Rechnernetz. Der Scan zeigte, dass ein SSH-Server auf der Steuerbox in Betrieb ist. Bei diesem SSH-Server handelt es sich um OpenSSH 7.9. Eine Recherche nach bekannten Schwachstellen dieser Version lieferte acht CVE-Einträge [70] [71] [72] [73] [74] [75] [76] [77]. Das zeigt, dass die Steuerbox Software betreibt, die bekannte und ausnutzbare Schwachstellen besitzt. Hier wird dringend empfohlen auf eine aktuelle Version der Software zu aktualisieren, bei der diese Sicherheitslücken bereits behoben wurden.

Auf den Nmap Scan folgte ein Versuch der Firmware Analyse. Diese war jedoch nicht möglich, da die Firmware der Steuerbox vermutlich verschlüsselt vorliegt und somit weder entpackt noch untersucht werden konnte.

Zuletzt erfolgte die Fuzzing Analyse der Steuerbox. Um während dieser nicht willkürlich zufällige Pakete an den Server zu senden, wurden die vorhandenen Steuerbox-Tools genutzt und die Kommunikation während der Nutzung mit Wireshark aufgezeichnet. Auf diese Weise wurden legitime Kommunikationssequenzen in Form von PCAP-Dateien erstellt. Diese PCAP-Dateien können genutzt werden, um den Datenaustausch in Fuzzern nachzubilden. Im Rahmen dieser Analyse wurde der Mutiny-Fuzzer der Cisco-Talos Gruppe [78] verwendet. Dieser liefert eine Funktionalität, durch die PCAP-Dateien automatisiert in für den Fuzzer geeignete Skripte umgewandelt werden können. Innerhalb dieser Skripte kann manuell festgelegt werden, welche der ausgehenden Pakete mutiert werden sollen und welche Antworten zu erwarten sind.

Die Fuzzing-Analyse zeigte auf, dass es durch gezielte Paketsequenzen möglich ist, die Steuerbox in einen (Fehl-)Zustand zu versetzen, in dem sie einmal durch alle Relais schaltet. Dieses Fehlverhalten kann je nach angeschlossener Last gravierend sein, da die Last dadurch vom Strom getrennt werden würde. Dieses Ergebnis muss dem Hersteller noch mitgeteilt werden.

II.1.4.3 Technische Analyse eines Blockchain basierten FlexHubs und Proof-of-Concept Demonstrator

In diesem AP erfolgt die Implementierung des Proof of Concept Demonstrators für den Blockchain basierten FlexHub nach einer Überprüfung der Machbarkeit auf Basis der entwickelten Konzepte. Diese Proof-of-Concept-Entwicklung wird durch Prüfung sicherheitskritischer Code-Bereiche und Etablierung eines geeigneten Software-Entwicklungsprozess begleitet.

Hieraus resultierende Aufgaben für die Fraunhofer Institute sind die folgenden:

1. Einrichten der Blockchain-Entwicklungsumgebung
2. Aufbau des Blockchain-Netzwerks
3. Implementation wichtiger blockchainspezifischen Konzepte
4. Implementation eines Auktionsmechanismus und mehrerer Smart Contracts
5. Zwischenevaluation mit den Projektpartnern
6. Abstimmung der Entwicklung
7. Bereitstellung von Coding-Guidlines und Code-Reviews bei Bedarf der Partner
8. Unterstützung bei der Entwicklung sicherheitskritischer Komponenten
9. Bewertungsmatrix von nichtfunktionalen Eigenschaften

Ergebnisse:

Zu 1-5:

FlexChain Sandbox: die Proof-of-Concept Blockchain
Das aktuelle Sandkasten System ist als Containers konzipiert, um die verschiedenen enthaltenden Dienste unbeeinflusst von der Host-Umgebung zu betreiben.

Das Netzwerk besteht aus folgenden Containern:

1. Drei bis fünf Quorum Knoten, die bis auf deren Adresse und verwendeten Schlüssel identisch konfiguriert sind. Die Knoten stellen die Verteilnetzbetreiber bzw. die Energie Management Systeme dar. Drei ist die Mindestanzahl an Knoten, womit der Raft Konsensalgorithmus wirkungsvoll arbeiten kann, bzw. Istanbul BFT benötigt 4 Knoten, um funktionieren zu können.
2. Ein Initialisierungs-Dienst, der auf die drei Quorum Knoten abwartet, bis sie vollständig gestartet sind. Dieser Dienst lässt sich per HTTP aufrufen, um den Zustand der Quorum Knoten leicht abzufragen.
3. Eins bis mehrere Schnittstellen jeweils für Verteilnetzbetreiber (VNB) und Energie Management Systeme (EMS). Diese sind als REST APIs umgesetzt und verschieben deren Start so

lange bis der Initialisierungsdienst am Laufen ist. Die EMS Clients verwenden die Quorum-Knoten, die als EMS Knoten bezeichnet sind, während die VNB Clients die VNB Quorum-Knoten verwenden.

Die drei Quorum Knoten verwenden Quorum in Version 21.10.2. Die Konfiguration in Einzelheiten ist der Tabelle 4 zu entnehmen. Von privaten Transaktionen wird zurzeit keinen Gebrauch gemacht. Neben der RPC Schnittstelle sind die Knoten über die 30303 Discovery Port erreichbar. Wenn Knoten auf verschiedenen Hosts laufen, brauchen sie den `rpccorsdomain` Flag aktiviert zu haben, um miteinander zu kommunizieren.

Eigenschaft	**Wert**
DLT / Version	Quorum / v20.10
Konsensalgorithmen	Raft (aktuell verwendet), Istanbul BFT, Clique
Knoten Schnittstellen	• Discovery • RPC API (admin, db, eth, debug, miner, net, shh, txpool, personal, web3, quorum, raft) • WebSockets • Inter-process (IPC)
Private Transaktionen	nicht verwendet
Permissioning	alle Knoten
Keystore Verschlüsselung Format	AES-128-CTR
Ethereum EVM Version	Istanbul
Ethereum Peer-2-Peer Protokolle	• RLPx Transport Protokoll • Ethereum Wire Protokoll
Rpccorsdomain	aktuell verwendet

Tabelle 4: Quorum Knoten Konfiguration im Sandkasten System

FlexChain Smart Contracts

Der Flex Smart Contracts setzen die Funktionalität des Flexibilitätsmarktes um. Die Smart Contracts verwenden die im AP3.3 für EMS und VNB beschriebenen Datenmodellen.

Die Interaktion zwischen VNB, EMS und dem Flex Smart Contract ist im Sequenzdiagramm von Abbildung 22 dargestellt. Hierbei nimmt der Flex Smart Contract die Rolle eines Flexibilitätsmarktes ein. Die einzelnen Interaktionen sind in Tabelle 5 beschrieben.

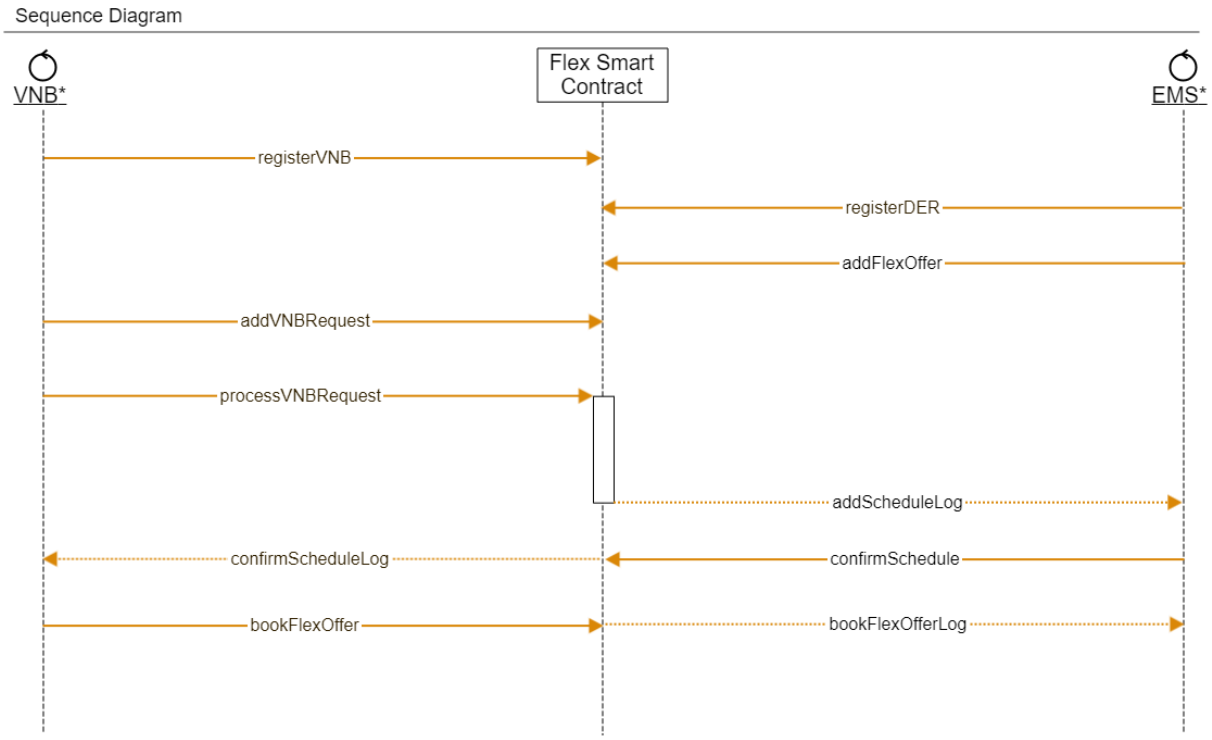


Abbildung 22: Sequenzdiagramm Interaktion mit dem Smart Contract gemäß Anwendungsfall 1

Name	Typ	Beschreibung
`registerVNB`	Transaktion	Meldet einen Verteilnetzbetreiber in die Blockchain an.
`registerDER`	Transaktion	Meldet eine Dezentrale Energie Ressource in die Blockchain an.
`addFlexOffer`	Transaktion	Fügt ein Flexibilitätsangebot zum entsprechenden DER hinzu.

`addVNBRequest`	Transaktion	Fügt eine Anfrage zum entsprechenden VNB hinzu.
`processVNBRequest`	Transaktion	Bearbeitet die VNB-Anfrage, indem dafür passende Angebote gefunden werden. Entsprechende Fahrpläne werden berechnet.
`addScheduleLog`	Event	Die berechneten Fahrpläne aus `processVNBRequest` werden als einzelne Events in der Blockchain verkündet.
`confirmSchedule`	Transaktion	Ein Fahrplan wird von EMS bestätigt.
`confirmScheduleLog`	Event	Die Bestätigung eines Fahrplans wird als Event verkündet.
`bookFlexOffer`	Transaktion	Ein Flexibilitätsangebot wird von VNB verbindlich gebucht.
`bookeFlexOfferLog`	Event	Die Buchung des Flexibilitätsangebots wird als Event verkündet.

Tabelle 5: Aktionen in der Interaktion zwischen VNB, EMS und dem FlexSmartContract

Die Matching-Logik hinter der `processVNBRequest` Anfrage stellt ein Optimierungsproblem dar, das passenden Flexibilitätsangebote gefunden werden, die den angekündigten Fahrplan der VNB-Anfrage möglichst gut abdecken. Der im Smart Contract implementierte Algorithmus ist im Detail in AP4.1 beschrieben. Der Smart Contract verwendet hierbei eine doppelt verlinkte Liste für die Flex-Angebote, die die sortierte Reihenfolge der Start-Zeiten beim Hinzufügen eines neuen Flex-Angebot in linearer Komplexität gewährleistet.

Anschließend resultieren Fahrpläne für die ausgewählten Flexibilitätsangebote, die mittels Solidity Events, in dem Falle `addScheduleLog`, in das Ledger angekündigt werden. Der zuständige EMS bekommt mittels eines Event-Filters von den Fahrplänen mit, der auf die entsprechenden Events abwartet. Der berechnete Fahrplan ist über den Call `getSchedule` abzufragen (mehr dazu in AP6.2).

Der EMS hat dabei die Möglichkeit, die Fahrpläne zu überprüfen. Falls der EMS oder der VNB Kapa mit dem Fahrplan nicht einverstanden ist, führt der Prozess in die Phase der Zurückweisung des Fahrplans. In dem aktuellen Sandkasten System wird aus Vereinfachungsgründen auf jegliche Prüfung verzichtet und sobald `addScheduleLog` Events aufgegriffen sind, wird die `confirmSchedule` Funktion unmittelbar für die jeweiligen Fahrpläne aufgerufen. Dies teilt die einverstandenen Fahrplandaten über den `confirmScheduleLog` Event in der Blockchain mit.

Der VNB erfährt über einen Event Filter von den bestätigten Fahrplänen und hat dabei noch die Möglichkeit die jeweiligen DER abzulehnen, was der Prozess in die Phase der Ablehnung des Vertrags führen würde. Aktuell wird der Einfachheit halber unmittelbar mit einer `bookOffer` Anfrage reagiert, wodurch die entsprechenden Flexibilitätsangebote als gebucht markiert werden.

Die Event Filters sind über einen Python-Listener umgesetzt, wobei die web3py Funktionen `events.eventName.createFilter` und `eventFilter.get_new_entries` in Einsatz kommen.

FlexChain APIs und Clients

Die Flex Smart Contract Funktionen werden über REST APIs für Verteilnetzbetreiber und Energie Ressource System zur Verfügung gestellt. Eine Flask API delegiert die Anfragen an jeweilige Python Clients, welche die VNB und EMS-Anfragen verarbeiten und sie anschließend mittels der web3py Bibliothek an den Flex Smart Contract weiterleiten.

Die Implementierung der Proof-of-Concept FlexHub in Form eines Sandkasten Systems ist in der Fraunhofer GitLab zu finden. Die Dokumentation der API befindet sich in dem Wiki innerhalb des GitLab Repository [79].

FlexHub Dapp

Die FlexHub Dapp bietet eine Benutzer-Oberfläche für die FlexChain Sandbox, um den modellierten Flexibilitätsmarkt visuell zu benutzen (vgl. Abbildung 23). Der Benutzer ist in der Lage, Aktionen zu starten, die in Transaktionen in dem FlexChain Sandbox umgewandelt sind. Die Anwendung ist als DApp bezeichnet, die Kurzform von Dezentralisierte Applikation, weil sie über das MetaMask Wallet die Verbindung zur Quorum-Netzwerk herstellt und anhand eines privaten Schlüssels die Transaktionen signiert. Die DApp ist mit Node.JS und Webpack entwickelt und verwendet Chart.js bei der Darstellung des Fahrplan-Diagramms. Als Docker Container hat sie Zugang zu dem Volume, das den deployten FlexRegister Smart Contract beinhaltet inkl. dessen Adresse in FlexChain, und ist auf dem gleichen Docker Netzwerk lokalisiert wie die FlexChain Anwendung.

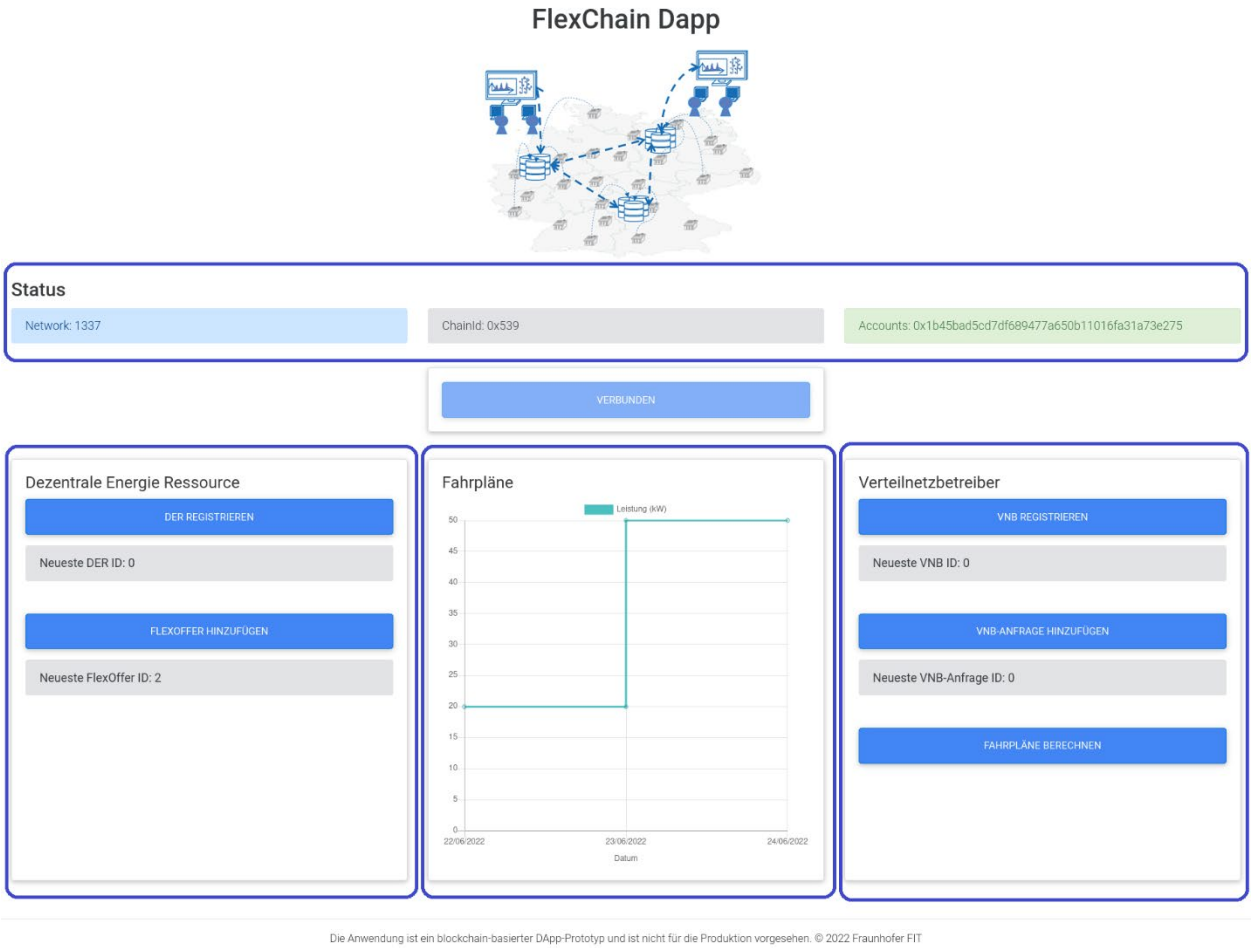


Abbildung 23: DApp Benutzeroberläche mit Eingabe-Funktionen zur Flexibilitätsmarkt

Der obere Bereich („Status“) in Abbildung 23 spiegelt die Blockchain-Verbindung wider, indem die Netzwerk-ID, ChainID und Konto (durch die Public Adresse) angezeigt werden. Zunächst sind diese Felder mit keinen Werten gefüllt. Erst nachdem der Benutzer auf „Verbinden“ klickt und MetaMask mit dem FlexChain Netzwerk und einem privaten Schlüssel eingestellt ist, werden die Status-Werte ausgefüllt. Falls das MetaMask-Wallet nicht als Browser-Erweiterung installiert ist, wird statt „Verbinden“ „Meta-Mask installieren“ angezeigt, wodurch der Benutzer auf die Erweiterungsseite des Wallets geleitet wird.

Auf der linken Seite lassen sich die Dezentrale Energie Ressource und deren Flex-Angebote in den Flexmarkt hinzufügen. Bei „DER registrieren“ handelt es sich um die Eingabe von Namen, Straße, PLZ, Stadt, Latitude, Longitude und Land, während bei „Flex-Angebot hinzufügen“ geht es um die Eingabe von Start- und Enddatum des Flex-Angebots, sowie um Offer-Typ (eine Auswahl zwischen „Producer“, „Consumer“ und „Prosumer“), die gesamte Energie (in kWh), die minimale und maximale Leistung in kW (vgl. Abbildung 24). Nach den betätigten Operationen erscheinen in den Textfeldern unterhalb der Knöpfe die zugewiesenen IDs der jeweiligen Objekte.

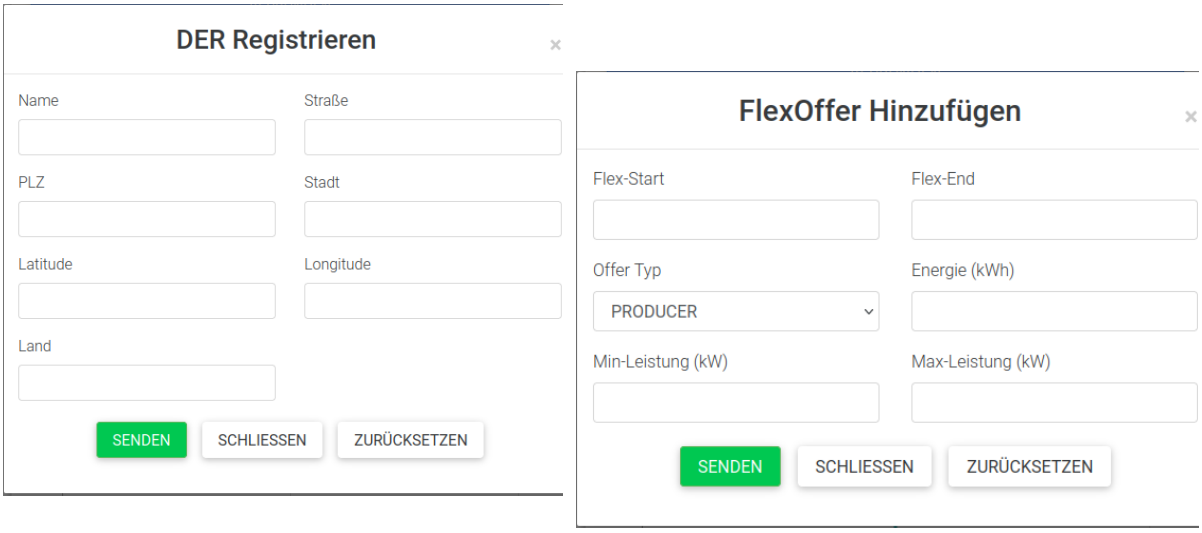


Abbildung 24: Benutzer-Dialoge mit der Registrierung einer DER bzw. Aufnahme eines Flex-Angebots

Auf der rechten Seite lassen sich die Verteilnetzbetreiber und deren Anfrage steuern. Um einen VNB zu registrieren, wird es lediglich einen Namen benötigt, während die Adresse und Ortsdaten in der VNB-Anfrage einzugeben sind. Zudem können eine oder mehrere Zeitspannen eingegeben werden sowie die gewünschte Leistung in kW (vgl. Abbildung 25). Im Anschluss an diesen Aktionen erscheinen in den Textfeldern unterhalb der Knöpfe die zugewiesenen IDs des VNB bzw. VNB-Anfrage.

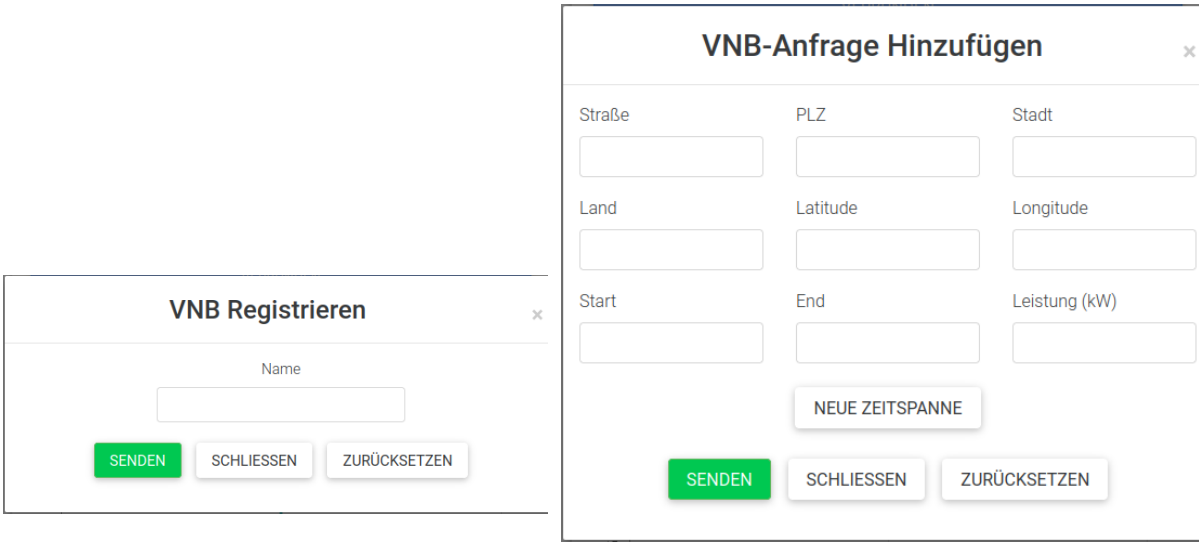


Abbildung 25: Benutzer-Dialoge mit der Registrierung eines VNB bzw. Aufnahme einer VNB-Anfrage

Mit einem Klick auf „Fahrpläne berechnen“ wird die Matching-Logik im Flexibilitätsmarkt ausgelöst, indem gültige DER-Angebote für eine ausgewählte VNB-Anfrage gefunden und Fahrpläne für diese

Angebote berechnet werden. Anschließend werden die Fahrpläne in mittlerem Bereich angezeigt (vgl. Abbildung 23). Alle betätigten Aktionen lösen Transaktionen in FlexChain aus, die in MetaMask zu bestätigen sind (vgl. Abbildung 26). Quorum ist ein gasfreies Netzwerk, d.h. die Gaspreise sind 0.

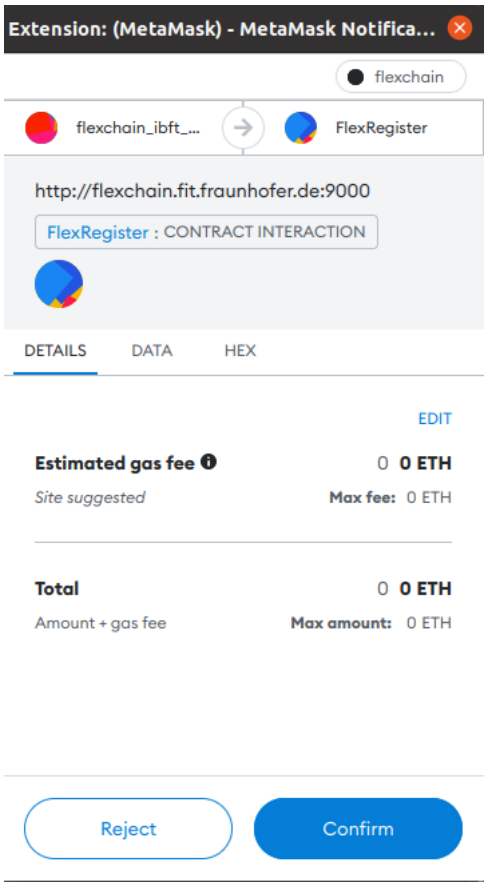


Abbildung 26: MetaMask fragt um Bestätigung einer Transaktion

Zu 6-8:

Aus IT-Sicherheitsperspektive wurde auf den IT-Sicherheitskatalog aus AP3 verwiesen, der ausführliche Best-Practices zum Thema IT-Sicherheit enthält. Zudem fand eine bilaterale Beratung in Form von Absprachen mit Fraunhofer FIT statt. Die Ergebnisse dieser Diskussionen wurden in den FlexChain Dokumenten festgehalten und den Projektpartnern zu Verfügung gestellt.

Zusätzlich und als Ersatz für die Code-Reviews erfolgte nach Absprache mit den Partnern und dem Projektträger die zuvor in AP4.2 vorgestellte IT-Sicherheitsanalyse von relevanten Kommunikationsprotokollen und -Komponenten.

Zu 9:
Eine Liste von nichtfunktionalen Eigenschaften, die auf die blockchainbasierte Architektur zutrifft, wurde verfasst und hierbei ausgewertet (vgl. Tabelle 6: Liste nichtfunktionaler Eingenschaften, die auf den blockchainbasierten Prototyp zutreffenTabelle 6). Die funktionalen Eigenschaften hingegen sind spezifisch zu jeder Architektur und wurden in der AP4.1 bzw. AP4.2 erläutert. Nichtfunktionale Eigenschaften beziehen sich auf die Qualitätsmerkmale des Software-Systems. Eine komplette Auflistung nichtfunktionaler Anforderungen ist in ISO 25010 definiert [80].

Eigenschaft-Kategorie	**Eigenschaft**	**Definition**	**Blockchain-basierte Anwendung**
Effizienz	Zeitverhalten	Grad, in dem die Reaktions-, Verarbeitungszeiten und Durchsatzraten eines Produkts oder Systems bei der Ausführung seiner Funktionen den Anforderungen entsprechen.	Siehe Reaktionszeiten in der Auswertung der Testergebnisse in AP 6.2.
	Ressourcenverbrauch	Grad, in dem die Mengen und Arten von Ressourcen, die von einem Produkt oder System bei der Ausübung seiner Funktionen verwendet werden, den Anforderungen entsprechen.	Siehe Anmerkungen bei der Grafana Server Monitoring (CPU, Arbeitsspeicher, Speicherplatz) bei der Testsuite Auswertung in AP 6.2.
Kompatibilität	Ko-Existenz	Grad, in dem ein Produkt seine erforderlichen Funktionen effizient erfüllen kann, wenn es eine gemeinsame Umgebung und Ressourcen mit anderen Produkten teilt, ohne	Die blockchainbasierterten Anwendungen (FlexChain Sandbox und Dapp) sind als Container-basierte Anwendungen implementiert worden. Die Container erfüllen jeweils eine einzige Funktion, z.B. ein Quorum Knoten pro Container, ein REST API pro Container. Der DApp Webserver ist in einem Container umgesetzt. Die

		nachteilige Auswirkungen auf ein anderes Produkt zu haben.	Anwendungen teilen einen Docker-Bridge-Netzwerk und Docker Volumen untereinander. Es wurden keine Nachteile beobachtet.
	Interoperabilität	Grad, in dem zwei oder mehrere Systeme Informationen austauschen und die ausgetauschten Informationen nutzen können.	Mit den Docker Volumen werden Informationen zwischen Containern ausgetauscht, z.B. die ABI und die Adresse des auf Quorum deployten FlexRegister Smart Contract sind mehreren Containern zur Verfügung gestellt. Das Docker Netzwerk ermöglicht die Kommunikation zwischen aller Container, z.B. die APIs haben Zugriff auf die Quorum Knoten und können Transaktionen senden.
Benutzbarkeit	Schutz vor Fehlern des Benutzers	Grad, in dem ein System die Benutzer vor Fehlern schützt.	Die Schnittstellen und die zugehörige Dokumentation sorgen dafür, dass die Benutzer vor Fehler des Systems geschützt werden.
Zuverlässigkeit	Reife	Grad, in dem ein System, ein Produkt oder eine Komponente die Anforderungen an die Zuverlässigkeit bei normalem Betrieb erfüllt.	Da es hier um eine Proof-of-Concept Demonstrator handelt, ist die Reife nicht von Bedeutung.
	Verfügbarkeit	Grad, bis zu dem ein System, ein Produkt oder eine Komponente funktionsfähig und zugänglich ist, wenn dies für den Gebrauch erforderlich ist.	Die Verfügbarkeit eines Blockchain-Netzwerks ist gewährleistet, falls ein oder mehrere Knoten ausfallen. Für die genaue Untergrenze bei der Anzahl der online Knoten siehe AP 4.2.
	Fehlertoleranz	Grad, in dem ein System, ein Produkt oder eine Komponente trotz	Da der FlexRegister Smart Contract umfangreich getestet wurde (siehe AP6.2) ist davon auszugehen, dass mögliche Fehler die

		vorhandener Hardware- oder Softwarefehler wie vorgesehen funktioniert.	Funktionalität der Systeme nicht beeinflussen.
Sicherheit	Vertraulichkeit	Grad, bis zu dem ein Produkt oder System sicherstellt, dass die Daten nur für zugriffsberechtigte Personen zugänglich sind.	Transaktionen in FlexChain Sandbox können nur über einen zugelassenen privaten Schlüssel initiiert werden. Der privaten Schlüssel kann mit Passwort geschützt werden. Die DApp kann nur über ein Wallet benutzt, das wiederum ein Konto auf Basis von dem privaten Schlüssel verwendet werden kann. Das Wallet ist ebenfalls passwortgeschützt.
	Integrität	Grad, in dem ein System, ein Produkt oder eine Komponente den unbefugten Zugriff oder die Änderung von Computerprogrammen oder Daten verhindert.	Betrifft die Authentifizierung, siehe Vertraulichkeit.
	Nachweisbarkeit	Grad, in dem Handlungen oder Ereignisse nachweislich stattgefunden haben, so dass die Ereignisse oder Handlungen später nicht zurückgewiesen werden können.	Die Quorum-Blockchain protokolliert die Benutzer-Aktionen und die Daten lassen sich nur einstimmig verändern. Durch ihre Notar-Funktion ist es sichergestellt, dass die einmal gespeicherten Daten zurückverfolgbar sind.
	Ordnungsmäßigkeit	Grad, bis zu dem die Handlungen einer Entität eindeutig auf die Entität zurückgeführt werden können.	Die Prozesse werden sowohl im FlexChain Sandbox als auch im Dapp im Hintergrund geloggt, damit Handlungen nachvollzogen werden können.
Wartbarkeit	Modularität	Grad, in dem ein System oder Computerprogramm aus	Die Containarisierung des FlexChain Sandbox und DApp stellt sicher, dass ein

		diskreten Komponenten zusammengesetzt ist, so dass eine Änderung an einer Komponente minimale Auswirkungen auf andere Komponenten hat.	Container/Modul mit der Ausnahme vom Kernel des hinterlegenden Systems keine Auswirkungen auf den anderen Container hat.
Portabilität	Anpassbarkeit	Grad, in dem ein Produkt oder System effektiv und effizient an unterschiedliche oder sich entwickelnde Hardware, Software oder andere Betriebs- oder Nutzungsumgebungen angepasst werden kann.	Die Systeme lassen sich anhand ihrer modularen Architektur leicht aktualisieren. Die Smart Contracts müssen von vorneherein so konzipiert, dass weitere Subklassen entwickelt werden und deren Blockchain-Adressen weiterhin unverändert bestehen.
	Installierbarkeit	Grad der Effektivität und Effizienz, mit dem ein Produkt oder System in einer bestimmten Umgebung erfolgreich installiert und/oder deinstalliert werden kann.	Die Einrichtung eines weiteren Quorum-Knotes ist im Rahmen dieses Proof-of-Concept Demonstrators nicht vorgesehen.
	Austauschbarkeit	Grad, in dem ein Produkt ein anderes spezifiziertes Softwareprodukt für den gleichen Zweck in der gleichen Umgebung ersetzen kann.	Die Quorum-Knoten lassen sich leicht ersetzen, indem die Konfigurationsdatei in GitLab gespeichert wird und die Daten sich über die anderen Knoten synchronisieren lassen.

Tabelle 6: Liste nichtfunktionaler Eingenschaften, die auf den blockchainbasierten Prototyp zutreffen

II.1.5 Arbeitspaket 5: Konzeptentwicklung zur IKT-Anbindung

Ziel des Arbeitspaketes 5 ist es die kommunikationstechnische Anbindung an den FlexHub genauer zu untersuchen. Hierfür wurden zuerst die Anforderungen welche sich aus den Netzbetriebsszenarien ergeben untersucht und die verschiedenen möglichen Technologie-Optionen gegenübergestellt. Basierend auf der Auswahl wurde die Breitband PLC-Technologie im realitätsnahen Einsatz in Laborversuchen vermessen und untersucht. Anschließend wurde die Anbindung der DEA an den FlexHub genauer untersucht. Hierfür wurden insbesondere die Gesamtinfrastruktur und die verschiedenen verwendeten Netzwerke und die üblichen Protokolle analysiert. In AP 5.3 wurden verschiedene Umgebungen für die Bewertung der IKT-Lösungen des FlexHub entwickelt.

II.1.5.1 Definition von Anforderungen und Netzbetriebsszenarien

Mithilfe von Informations- und Kommunikationstechnik (IKT) können Energieerzeugung und -verbrauch effizient verbunden und ausbalanciert werden. Ein zentrales Element sind intelligente Messsysteme. Diese sorgen für Verbrauchstransparenz und übermitteln sicher Messdaten. Zusätzlich können sie eine Plattform für die Steuerung elektrischer Verbrauchsgeräte bieten. Damit kann zukünftig das Last- und Erzeugungsmanagement im Verteilnetz optimiert werden. Ein intelligentes Messsystem besteht aus einer modernen Messeinrichtung und einem Smart Meter Gateway (SMGW). Die moderne Messeinrichtung verfügt über einen digitalen Zähler. Das SMGW mit integriertem Sicherheitsmodul ist die Kommunikationseinheit des Smart Meter (SM). [61] Der Smart Meter Gateway Administrator (SMGWA) ist eine zertifizierte, vertrauenswürdige Instanz, die das SMGW konfiguriert und den sicheren Betrieb sicherstellt. Dieser Administrator konfiguriert beispielsweise das kryptographische Schlüsselmaterial und die Regularien für die Tarifierung. Es werden verschiedene Schnittstellen durch das SMGW wie Lokales Metrologisches Netz (LMN- Local Metrological Network), Weitverkehrsnetz (WAN-Wide Area Network) und Lokales Heimnetz (HAN – Home Area Network) bereitgestellt.

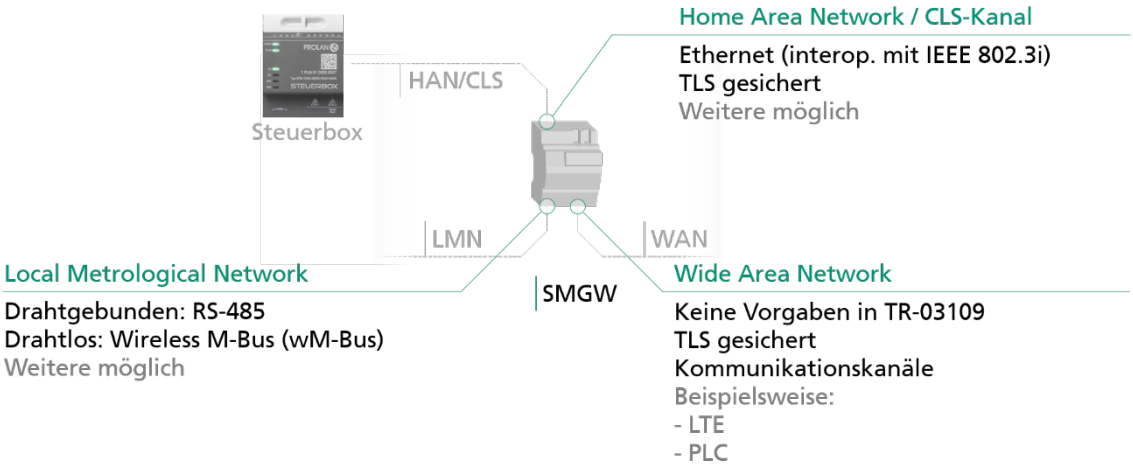


Abbildung 27: SMGW-Infrastruktur für den FlexHub

Lokale Metrologische Netze – LMN

Mit dem LMN werden die Messeinrichtungen mit dem SMGW verbunden. Das Gateway speichert und verarbeitet die erhobenen Einspeise- und Verbrauchswerte sowie die Netzzustandsdaten (Spannung, Frequenz, Phasenwinkel). Dabei werden die Daten je nach Tarif des Kunden unter Einbeziehung des Eichrechts und des Datenschutzes weiterverarbeitet.

Das Weitverkehrsnetz – WAN

Über das WAN kann mit externen Markteilnehmern kommuniziert werden. Dazu gehört auch der SMGWA. Sämtliche Kommunikationsverbindungen gehen aus Sicherheitsgründen von dem Gateway aus. Diese können zu variablen oder festgelegten Zeitpunkten erfolgen. Durch einen Wake-Up-Dienst für spontane Ereignisse kann der SMGWA das Gateway zu einem Verbindungsaufbau animieren. Hierbei wird ein vom SMGWA zeitbegrenztes, signiertes Datenpaket verwendet. Nach erfolgreicher Überprüfung des SMGW baut dieses eine gesicherte Verbindung auf.

Das Heimnetz – HAN

An die HAN-Schnittstelle werden steuerbare Geräte wie Wärmepumpen oder Photovoltaikanlagen angeschlossen. Hierrüber kann der Zugriff für Fernwartungs- oder Steuerungszwecke externer Markteilnehmer ermöglicht werden. Das SMGW stellt dafür einen sicheren, transparenten Kommunikationskanal bereit. Dieser Kanal kann nur durch den SMGWA konfiguriert werden. [81]

Im Folgenden wird die WAN-Schnittstelle des SMGW betrachtet. Diese muss nach der Technischen Richtlinie BSI TR-03109-1 für die Zeitsynchronisierung HTTPS-fähig sein. Für die Ausschöpfung des Potentials der SM und folglich damit des SMGW müssen weitere Anforderungen an die Kommunikationstechnologie gestellt werden. Damit eine fehlerfreie Datenübermittlung gewährleistet werden kann.

SM versenden und empfangen regelmäßig Datenpakete einiger Kilobyte. Damit werden unter anderem Messdaten, mögliche Steuerbefehle sowie Updates des SM-Status umfasst. Bei einer Messwerterfassung alle 0,5 Sekunden und einem Datentransfer an den Administrator alle 15 Minuten werden 154 Kilobyte erwartet [82]. Die Größe der Software-Updates eines SMs werden mit einigen Megabyte angegeben [82]. Um die Übermittlungszeit möglichst gering zu halten, sollte einem SMGW eine Anbindung mit mindestens einem 1 Mbit/s (Up- und Download) zur Verfügung stehen [83],[84]. Die Kommunikation der SMGWs ist bidirektional, sodass Messdaten des SMs übertragen und Steuerbefehle des SMGWAs empfangen werden können. Zusätzlich muss das SMGW direkt adressierbar (IP-fähig) sein. Eine geringe Latenz bei der Übertragung führt zur schnelleren Umsetzung der Steuerbefehle. Dafür ist eine Latenz von unter 100 ms angemessen. Aufgrund der Verwendung der SM von bis zu 20 Jahren [26] muss die verwendete Infrastruktur zur Datenübermittlung ohne weitere Hardware-Anpassungen in demselben Zeitrahmen zur Verfügung stehen.

Abbildung 28: Einordnung FlexHub Use Case in IKT Netzwerkinfrastruktur

WAN-Technologie	Einordnung	Realisierbare Latenzzeit	Direktionalität	Typische Übertragungsrate	Typische maximale Reichweite	IP fähig
	Kabelgebunden/drahtlos Öffentlich/dediziert	A: < 100 ms B: > 100 ms	BI: Bidirektional UN: Unidirektional	A: < 1 Mbit/s B: > 1 Mbit/s	(pro 1x Repeater)	Ja/Nein

Abbildung 29: Anforderung an die WAN IKT des FlexHubs

Auswahl IKT

Basierend auf den Anforderungen in 1.3 wurde eine Vorauswahl verfügbarer Technologien getroffen. Die Technologien können in drahtlos und kabelgebunden sowie öffentlich und dediziert eingeteilt werden. Die Mobilfunknetze UMTS, LTE und 5G erfüllen die geforderten Anforderungen. Die Unterschiede zwischen den Technologien ist der Kompromiss zwischen Reichweite und Übertragungsrate. Diese öffentlichen Netze stellen jedoch kein dediziertes Netzwerk dar. Für eine kabelgebundene Verbindung eignen sich Power Line Communication (PLC) und der häusliche DSL-Anschluss.

UMTS

Die Bundesnetzagentur (BNetzA) hat festgelegt, dass bundesweit 98 Prozent aller Haushalte mit einer Datenrate von mindestens 50 Mbit/s pro Antennensektor zu versorgen sind. [85] Mit UMTS-Technik lässt sich diese Datenrate nicht realisieren. Das öffentliche UMTS-Netz wird folglich 2021 in Deutschland abgeschaltet und wird deshalb in diesem Bericht nicht weiter betrachtet.

LTE

Die LTE-Technologie umfasst Frequenzen von 700 MHz bis 2600 MHz. Dabei sind die meist verwendeten Frequenzbänder 800 MHz, 1.800 MHz, 2.100 MHz und 2.600 MHz. Mit der 800-MHz-Frequenz kann eine Reichweite von bis zu 15 km erreicht werden. Diese wird folglich besonders in ländlichen Regionen eingesetzt. Die maximale Datenrate liegt dabei noch bei einigen MBit/s. Die 2,1-GHz- und die 2,6-GHz-Frequenzen erreichen eine maximale Reichweite von bis zu drei Kilometern. Dafür bieten diese eine Datenrate von über 100 MBit/s. LTE ist insbesondere in städtischen Regionen flächendeckend verfügbar. Wenig besiedelte Regionen werden jedoch noch nicht abgedeckt.

5G

Die 5G-Technologie umfasst einen Frequenzbereich von 600 MHz bis 40 GHz. Eine Erweiterung bis 60 oder 80 GHz ist zukünftig geplant. Funkfrequenzen unter sechs Gigahertz werden als Sub 6 5G bezeichnet. Bei Frequenzen ab 20 GHz wird von Millimeterwellen gesprochen. Dieser Bereich wird in diesem Bericht aufgrund der geringen Reichweite von einigen hunderten Metern und der geringen Gebäudedurchdringung (Line of Sight) nicht weiter betrachtet. 5G nutzt teilweise die gleichen Frequenzen wie LTE. Neue 5G-Techniken ermöglichen jedoch erhöhte Nutzerzahlen sowie Reichweiten. Beamforming bündelt Funkwellen, sodass diese zielgerichtet zu einem Empfangsgerät gelangen. Durch diese Techniken nähert sich die Reichweite des 3,6-GHz-Frequenzbereiches der Reichweite des 2-GHz-Bereiches an. Die Frequenzbänder 3,6 GHz und 2,1 GHz erreichen jeweils eine Reichweite von einem und drei Kilometer. Das 700-MHz-Band wird für die Versorgung ländlicher Regionen genutzt. Mit einer Reichweite von 15-20 km und einer Datenrate von 100 bis 200 Mbit/s eignet es sich auch für SM-Anwendungen. Aktuell befindet sich das 5G-Netz im Ausbau. Folglich ist noch kein flächendeckender Empfang möglich.

450 MHz

Auf der Basis der LTE-Technologie wird das 450 MHz Mobilfunknetz errichtet. Es soll eine sichere und hochverfügbare Kommunikation für Maschine-to-Maschine-Anwendungen bereitstellen. Das Netz garantiert die Systemverfügbarkeit durch Notstromversorgung der Infrastruktur und ist damit schwarzfallfest. Das Funknetz soll bis Ende 2024 mit rund 1600 Funkstandorten in Betrieb gehen. Aktuell sind rund 20 Prozent der Fläche Deutschlands abdeckt und erste Teilnetze sind bereits im Betrieb. Die neu errichtete Infrastruktur befindet sich im Eigentum der Energieversorger. Die 450 MHz Frequenz stellt den Kompromiss zwischen Flächenversorgung, Gebäudedurchdringung und Bandbreite dar. [30]

Satellit

Eine Datenübertragung mittels Satelliten ist ebenfalls möglich, jedoch wird für diese Verbindung zusätzliche, teure Hardware benötigt. Außerdem ist eine freie Sicht auf den Himmel notwendig. Im Rahmen des Berichtes wird eine Satelliten-Verbindung nicht weiter betrachtet.

Long Range Wide Area Network (LoRaWan)

Long Range Wide Area Network (LoRaWan) basiert auf einem offenen Industriestandard und zeichnet sich durch Energieeffizienz aus. Die Reichweite beträgt innerhalb dicht besiedelter Regionen 5 km und 16 km in ländlichen Regionen. Einzelne Endgeräte können jedoch nicht direkt adressiert werden. Die Technologie-Architektur LoRaWans sieht eine IP-basierte Kommunikation zu Gateways vor, welche die Brücke zu einem zentralen Server darstellen. Aufgrund der fehlenden direkten Adressierbarkeit wird LoRaWan nicht weiter betrachtet. [86]

xDSL

DSL erfüllt alle definierten Anforderungen, stellt jedoch kein dediziertes Netzwerk dar. DSL wird jedoch im Rahmen dieses Projekts als Übergangstechnologie bis zum Ausbringen der SMGW-Infrastruktur eingesetzt.

Power Line Communication (PLC)

Bei der PLC werden Datenpakte über die Leiter der Stromversorgung versendet. Für die Datenübertragung wird eine Frequenz im Kilo- oder Megaherz-Bereich gewählt. Narrowband Power Line Communication (NPL), welche im Kiloherzbereich betrieben wird, wird aufgrund der geringen Bandbreite (< 1 Mbit/s) nicht weiter betrachtet. Das Übertragungsband der Broadband Power Line Communication (BPL) liegt zwischen 2 MHz und 30 MHz. Damit können Datenübertragungsraten einiger Mbit/s erreicht werden. Die Reichweite der BPL ohne zusätzliche Repeater ist jedoch auf einige hundert Meter limitiert. [87],[84]

WAN-Technologie	Einordnung	Realisierbare Latenzzeit A: < 100 ms B: > 100 ms	Direktionalität BI: Bidirektional UN: Unidirektional	Typische Übertragungsrate A: < 1 Mbit/s B: > 1 Mbit/s	Typische maximale Reichweite ohne Geräteeinsatz (pro 1x Repeater)	Durchgängig IP-fähig
xDSL/ TV-Kabelnetz	Kabelgebunden öffentlich	A	BI	B	3 km	Ja
Mobilfunk GPRS	Drahtlos öffentlich	B	BI	A	10 km	Ja
Mobilfunk UMTS		B	BI	B	10 km	Ja
Mobilfunk LTE		A	BI	B	30 km	Ja
Satellit		B	BI	B	> 1000 km	Ja
BPL	Kabelgebunden dediziert	A	BI	B	0,5 km	Ja
NB-PLC		B	BI	A	3 km	Ja
TF-Rundsteuertechnik		B	UN	A	> 10 km	Nein
Eigene Leitungen (LWL oder Cu)		A	BI	B	> 10 km	Ja
e°Nergy	Drahtlos dediziert	B	UN	A	>10 km 800 Sender in Deutschland	Nein
EFR		B	UN	A	500 km 3 Sender in EU	Nein
LoRaWAN		A/B	BI	A	5 km	Nein
TETRA		B	BI	B	5 km	Nein
CDMA / LTE450	Drahtlos	A	BI	B	30 km	ja

Abbildung 30: Anforderungsanalyse IKT für den FlexHub [61]

IKT-Vermessung

Für die Bewertung der BPL-Technologie wurde im Smart Grid Labor der RWTH Aachen Untersuchungen in einer realitätsnahen Umgebung durchgeführt. Hierfür wurden ein NS-Netz aufgebaut wobei das Head-End des BPL-Netzwerkes in der Ortsnetzstation verbaut ist. Die einzelnen Modems wurden an den jeweiligen nachgestellten Hausanschlüssen verbaut. Das abgebildete Netzwerk bestand aus vier verschiedenen Modems. In Abbildung 31 ist der Aufbau mit den Modems an den Hausanschlussnachbildungen zu sehen.

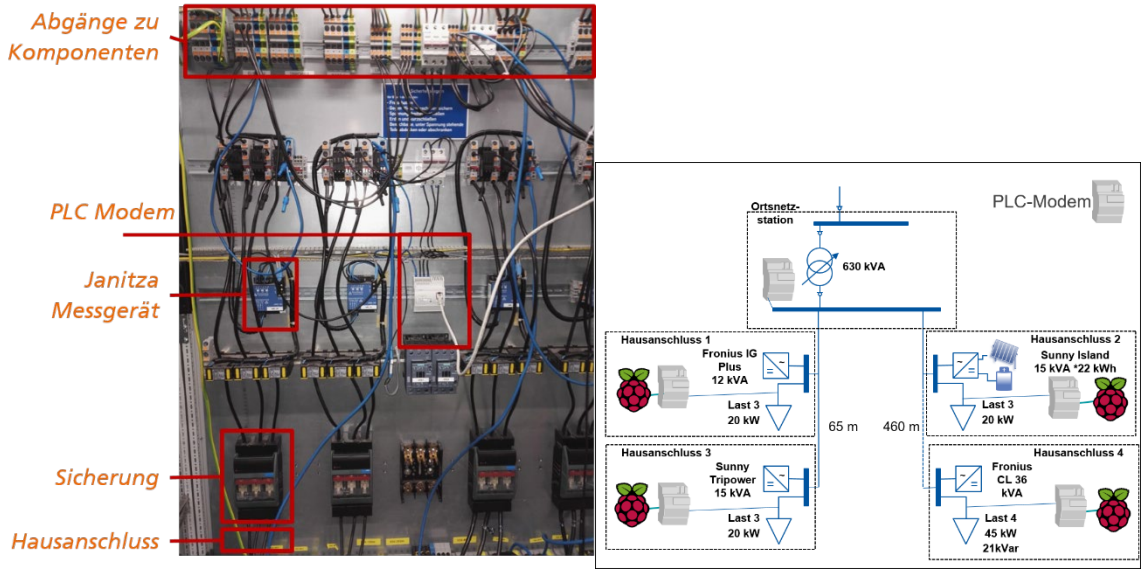


Abbildung 31: Aufbau BPL Vermessung

Die Vermessung der erreichten Bandbreite wurde mit Hilfe von iPerf3 und Wireshark durchgeführt. Hierbei wurden verschiedene Betriebsszenarien mit angeschlossenen Wechselrichtern nachgebildet bei allen Versuchen konnten im Labor die erwarteten Bandbreiten erreicht werden. Eine Gegenüberstellung verschiedener Messungen ist in Abbildung 32 zu finden. Außerdem ist die statistisch Auftrittshäufigkeit verschiedener Verbindungsgeschwindigkeiten für die UDP-Messung an Modem SMGW 2 exemplarisch dargestellt.

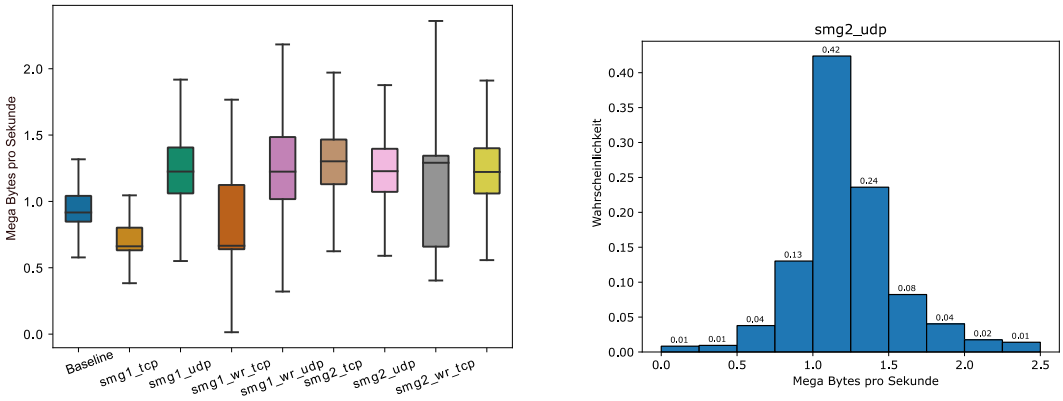


Abbildung 32: Vermessungsergebnisse BPL

II.1.5.2 Kommunikationstechnologien für die DEA-Anbindung

Für die Ansteuerung der verschiedenen Komponenten im Labor wurde das Modbus-Protokoll eingesetzt. Die große Verbreitung des Protokolls ermöglicht die Ansteuerung verschiedenster Geräte Die große Inhomogenität in der Umsetzung der Datenpunkte erfordert eine Anpassung an jeden Hersteller. Um die Anbindung im Rahmen dieses Projektes umzusetzen, wurde aus diesem Grund eine Middleware entwickelt welche die Ansteuerung der verschiedenen Geräte für die Simulationswerkzeuge und Prototypen abstrahiert. Dies bedeutet für die Hersteller der HEMS-Lösung auch immer einen entsprechenden Implementierungsaufwand, der notwendig ist um dies Anbindung umzusetzen. Die Reaktionszeit der einzelnen Komponenten wurden im Labor vermessen, hierbei konnten keine für den Anwendungsfall signifikanten Verzögerungen festgestellt werden. Eine Bewertung der Sicherheit möglicher Protokolle und Geräte mittels Fuzzing wurde in AP 4.2 bereits genauer beschrieben.

FNN Steuerbox

Eine alternative stellt zukünftig die FNN Steuerbox dar die am marktbefindlichen Steuerboxen konnten jedoch nicht für die Umsetzung der entwickelten Use-Cases im Projekt eingesetzt werden. Es fand jedoch eine intensive Untersuchung der vorhandenen Systeme und zukünftiger Anforderungen statt. Die Steuerbefehle des SMGWs werden zu einer Steuerbox weitergeleitet. Mithilfe dieser Box werden die Befehle für das netzdienliche Erzeugungs- und Lastmanagements umgesetzt. Zum Beispiel kann der Wechselrichter einer PV-Anlage abregelt werden oder Nachtspeicherheizungen sowie das Laden von Elektroautos reguliert werden. Das VDE FNN erarbeitet eine technologieneutrale Spezifikation für solche Steuer- und Schaltmodule. Die FNN Steuerbox besitzt vier analoge Kontakte, mit denen Leistungsstufen gesteuert werden können. Je nach Umsetzung der Steuerung können 16 (4 Bit) oder 4 Abstufungen realisiert werden. EEBUS wurde als erste digitale Schnittstelle im Dezember 2020 veröffentlich. Für September 2021 ist die Veröffentlichung einer weiteren digitalen Schnittstelle (KNX) vorgesehen. Zukünftig sollen weitere digitale Schnittstellen folgen.

II.1.5.3 Bewertung von IKT-Lösungen für den FlexHub

Für die Untersuchung und Bewertung der IKT-Lösungen im Rahmen des FlexHub-Projektes wurde eine flexible Simulationsumgebung für die Bewertung der verschiedenen Entwicklungsstufen einer Flexibilitätslösung geschaffen. Die Entwicklungsstufen beginnen dabei mit einem Flexibilitäts Use-Case, bei dem simulativ die Rückwirkung auf das Stromnetz untersucht wird bis hin zu einer Implementierung der notwendigen Plattformen und Home Energie Management Systeme. Um die Umgebung flexibel auf die verschiedenen Anforderungen anpassen zu können wurde eine Co-Simulationsumgebung aufgebaut. Eine genaue Vorstellung der gesamten Umgebung und der verschiedenen Anwendungsfälle wurden in [88] durchgeführt.

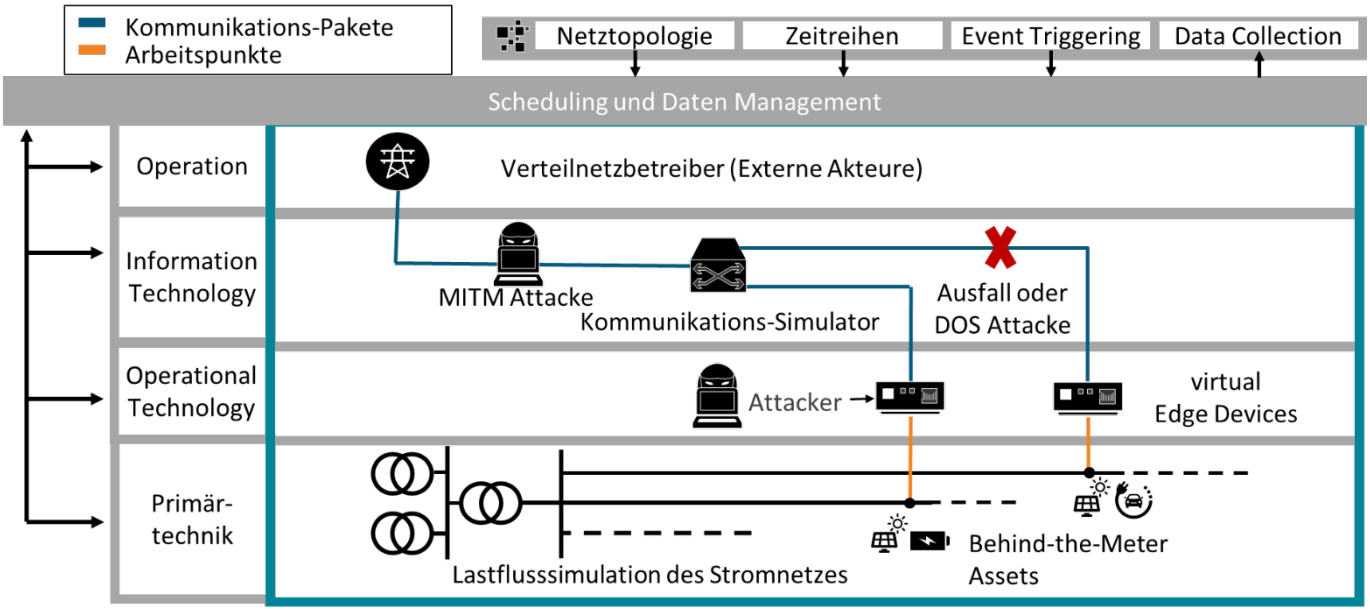


Abbildung 33: Aufbau Co-Simulation

Die verschiedenen Domänen die simuliert wurden, sind die Primärtechnik des Energienetzes, die Operational Technology, die Kommunikationsnetze und die Betriebsebene. Die Primärtechnik wird durch PandaPower simuliert oder alternativ wird die Co-Simulation an das SmartGrid-Labor der RWTH angebunden und das reale Netz wird anstelle einer Simulation genutzt (siehe Kapitel **Fehler! Verweisquelle konnte nicht gefunden werden.**). Die Operational Technology im Rahmen des Projektes sind die HEMS oder die Steuerboxen welche die Assets im Haushalt der Kunden steuern. Die Simulation und das Labor ermöglichten hier auch Kombinationen von Assets welche im Feldtest nicht möglich waren, zum Beispiel der Einsatz von Heimspeichern. Für diese Ebene wurde ein eigener Simulator entwickelt die „Virtual Edge Devices“ (VED). Die VED ermöglichen die Simulation und Emulation von EMS. Die Architektur der VED ist in Abbildung 34 dargestellt. Der Simulationskern liefert das zentrale Datenmanagement und steuert die einzelnen Interfaces und Logikblöcke. Die Interfaces abstrahieren das angebundene System gegenüber dem Simulationskern, so existieren drei mögliche Konfigurationen für Interfaces.

Entweder erfolgt die Anbindung an andere Simulatoren der Co-Simulation oder an reale Geräte wie Wechselrichter (z.B.: über Modbus) oder an Plattformen wie den FlexHub (z.B.: über REST). Außerdem können die Interfaces selbst als simulierte Geräte wie zum Beispiel als Heimspeicher genutzt werden. Dieses Design ermöglicht es flexibel die verschiedenen Komponenten in den verschiedenen Entwicklungsstufen auszutauschen. Die VEDs können hierbei entweder als Teil der Simulationsumgebung gestartet oder je nach Bedarf als Docker Container oder als dedizierte Hardware Komponenten auf Raspberry Pis deployt werden. Insbesondere der gemeinsame Einsatz mit dem Labor wurde in [89] genauer beschrieben. Die Laborversuche sind in AP 7.2 genauer beschrieben.

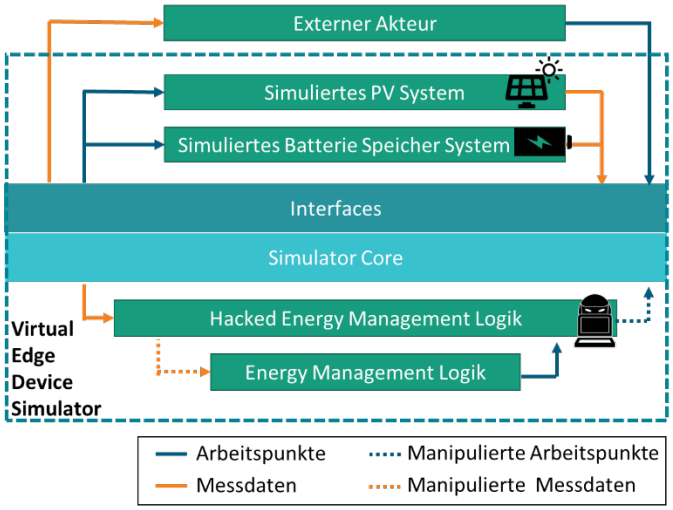


Abbildung 34: Aufbau Virtual Edge Device

Die Kommunikationsnetze können in der Ebene Information Technology auf drei Arten abgebildet werden. Auch hier können wieder reale Kommunikationsstrecken wie beispielsweise das aufgebaute PLC-Netz eingesetzt werden. Alternativ kann das Kommunikationsnetz mit Hilfe der Emulationssoftware Containernet nachgebildet werden. Hier werden die VED und die Plattform für den FlexHub als Docker Container deployt, die Kommunikationsverbindungen können mit definierter Bandbreite und Latenz emuliert werden. Für die Untersuchung der Auswirkungen von IT-Angriffen und Ausfällen wurde ein Kommunikationsnetz-Simulator entwickelt. Dieser ermöglicht die Abstraktion von konkreten Protokollen und Kommunikationsverbindungen um auch zukünftige unbekannte Angriffsoberflächen losgelöst von Implementierungsdetails zu untersuchen. Hierfür wurde die Kommunikation des *Application Layers* in der Simulation nachgebildet und Attacken auf die Kommunikationsverbindungen umgesetzt. So können Denial-of-Service und Maschine in the Middle Attacken untersucht werden. Um mögliche IoT-Angriffsvektoren auf die HEMS untersuchen zu können wurden die VED entsprechend erweitert. Die Logik-Blöcke können nun um eine durch den Angreifer verfälschte Logik erweitert werden. Der Kommunikations-Simulator und die Untersuchung von Cyber-Angriffen wurden in [88] genauer vorgestellt.

Für die Skalierbarkeitsuntersuchungen war insbesondere die neuartige Blockchain-Technologie von Interesse. Um die Skalierbarkeit der Plattformen untersuchen zu können, wurde die Umgebung entsprechend angepasst, um die Containerbasierte Umgebung losgelöst von der Energienetz Simulation zu nutzen. Hierbei konnten entsprechend Interaktionen für verschiedenen große Kundenanzahlen

untersucht werden. Eine detaillierte Auswertung der Ergebnisse findet sich in Kapitel **Fehler! Verweisquelle konnte nicht gefunden werden.**.

II.1.6 Arbeitspaket 6: Implementierung und Entwicklung einer Plattform

Die Entwicklungsarbeiten bei Kiwigrid basieren auf einer agilen, sowie iterativen und inkrementellen Vorgehensweise. Mehrere Teams waren im Laufe der Projektzeit an der Umsetzung des Vorhabens beteiligt. Arbeiten am SMGW fanden in einer Sicherheitszone im Kiwigrid Bürogebäude statt, zu dem nur entsprechend ausgewiesene Mitarbeiter Zutritt gewährt wurde.

Technologisch betrachtet handelt es sich bei der Kiwigrid Plattform um eine modulare serviceorientierte Cloud-Plattform. Telemetriedaten von DEA können auf dieser nahezu echtzeitfähig verarbeitet werden. Die hohe Reaktionsfähigkeit wird durch eine bidirektionale, stehende Verbindung zu Feldgeräten und Applikationen gewährleistet. Für die Speicherung der Momentanwerte aller angeschlossenen Geräte wird eine In-Memory-Datenbank genutzt. Zusätzlich existieren in allen Applikationen Datenströme für die Bewegungsdaten der Geräte. Die historischen Telemetrie-Daten aller DEA werden mit Hilfe einer Big-Data Technologie verarbeitet. Für die Verarbeitung der Datenströme wird eine asynchrone, nachrichtenbasierte Servertechnologie verwendet. Die Plattform beinhaltet horizontal skalierbare Komponenten in allen Architekturschichten, wodurch auch die gesamte Plattform skalierbar ist. Um eine maximale Portierbarkeit und Herstellerunabhängigkeit zu gewährleisten, wird Freie Open Source Software genutzt. Die Portierung ist sowohl auf eine Public-Cloud als auch auf Rechenzentrumsumgebungen möglich.

Das Vorhaben wurde seitens Kiwigrid zum größten Teil digital und softwareseitig umgesetzt, allerdings stets mit einer Umsetzung innerhalb eines Feldtests der Mitnetz Strom im Fokus. Tests wurden daher sowohl im Labor der Kiwigrid als auch im Feld mit echten Kunden durchgeführt.

Die Daten der DEA waren zum Zeitpunkt der Antragstellung ausschließlich vertraglich Nutzern der Kiwigrid Plattform zugänglich. Standardisierte Schnittstellen, die den Zugriff auf die DEA von externen Parteien zulassen, waren zum Start des Projektes noch nicht vorhanden. Dafür wurden im FlexHub Projekt von Kiwigrid standardisierte Schnittstellen entwickelt, welche es externen Parteien, wie Aggregatoren, BKVs und VNBs ermöglichen auf DEA zuzugreifen.

Die Testumgebung für die Labor- und Feldtests stand bei Kiwigrid zur Durchführung von Tests bereit. Bei den Aufbauarbeiten wurden die Funktionalität und die korrekte Verwendung der Kommunikationstechnologie sichergestellt.

II.1.6.1 Anforderungen der Demonstratoren an den FlexHub

Im Rahmen von AP 6 verschob sich der Fokus der HAW Hamburg von einer Gleichverteilung der Arbeitspakete von 6 PM in je AP 6.1 und AP 6.2 hin zu etwa 3 PM Aufwand in AP 6.1 und 9 PM Aufwand in AP 6.2. Begründen lässt sich dies dadurch, dass die Anforderungen an ein hierarchisches Flexibilitätenregister, nach den Arbeiten in AP 4 und der Entwicklung des Prototypen in diesem Arbeitspaket

schon sehr weitgehend erfasst waren. Daher wurde die Aufwände teilweise verlagert, sodass mehr Ressourcen für die Implementierung eines FlexHubs in eine Testplattform in AP 6.2 zur Verfügung standen und hier entsprechend eine Lösung entwickelt werden konnte, die technisch weiter ausgeprägt ist.

Der Fokus der HAW Hamburg in diesem Arbeitspaket lag dabei insbesondere in der Abbildung der Flexibilitätspotentiale und der Bepreisung der angebotenen Flexibilität. Daraus ergaben sich die folgenden Anforderungen insbesondere mit Hinblick auf die Markfunktionalitäten des Flexibilitätenregisters.

Tabelle 7: Kernanforderungen an die Flexibilitätsplattform

Anforderung	**Kategorie**	**Gewichtung**
Ein Anbieter von Flexibilität kann die Stammdaten seiner Anlage (Flexibilitätsressource) in der Flexibilitätsplattform anlegen (White Page Information). Diese Stammdaten beinhalten insbesondere den Ort der Erbringung der Leistung. Um Georeferenzierte Dienstleistungen anbieten zu können müssen diese mindestens die Anschrift und die GPS-Koordinaten enthalten.	Stammdaten	Must
Ein Anbieter kann die Stammdaten seiner Anlagen einsehen.	Stammdaten	Must
Ein Anbieter kann die Stammdaten seiner Anlagen verändern.	Stammdaten	Should
Ein Anbieter von Flexibilität kann als Energiedienstleistung die zeitliche Verschiebung seiner Ladung/seines Energiebezugs anbieten. Dabei definiert der Anbieter den geltenden Flexibilitätsrahmen. Er bestimmt also den Angebotszeitraum, als den eigentlichen Zeitraum, in dem der Energiebezug erfolgen kann und bis wann die Energie bezogen sein muss. Weiterhin definiert er die benötigte Menge an Energie, die er bei Buchung beziehen will und gibt die Leistungsparameter seiner Anlage vor. Außerdem bestimmt er den Preis in €/kWh, der er für das Verschieben seiner Ladung haben will. (Yellow Page Informationen)	Markt	Must
Ein Anbieter von Flexibilität kann neben einem fixen Preis auch einen flexiblen Preis definieren. Dabei kann er für jede Viertelstunde des Flexibilitätszeitraums einen anderen Preis ansetzen. Die Viertelstunden sind dabei auf „volle“ Viertelstunden terminiert. Damit kann der Anbieter präferierte Zeiträume definieren, in denen der es bevorzugt Energie zu beziehen und weniger präferierte, aber mögliche Zeiträume mit einem höheren Preis versehen	Markt	Should

Ein Nachfrager von Flexibilität kann anhand von Suchparametern, die den kompletten Datenraum der White- und Yellow Page Informationen abdeckt nach Flexibilitätsangeboten suchen.	Markt	Must
Ein Nachfrager von Flexibilität kann diese innerhalb des Angebotszeitraums der Angebote buchen.	Markt	Must
Ein Nachfrager von Flexibilität kann von ihm gebuchte Flexibilitäten über Fahrpläne, die er an die Plattform sendet, steuern. Die Plattform leitet diese dann an die Anlage weiter.	Markt	Must
Der Verteilnetzbetreiber in seiner Rolle als Kapazitätsmanager hat die Möglichkeit jede Buchung von Flexibilität optional zu validieren. Stimmt er der Buchung nicht zu, da diese die Stabilität in seinem Netz gefährdet, kommt die Buchung nicht zustande.	Validierung	Must
Die Plattform stellt einen Aggregationsfunktion zur Verfügung über die ein Flexibilitätsnachfrager als One-Click-Solution ein Problem bzw. einen Zielfahrplan definieren kann und die dann aus allen vorhandenen Flexibilitätsangebote, diejenigen auswählt, die die Anfrage am besten abbilden.	Optimierung	Should
Die Aggregationsfunktionalität erfüllt die oben aufgeführten Anforderungen und bevorzugt bei Flexibilitätsangeboten, die in gleich guter Weise zur Lösung beitragen, anhand der Preisinformationen das günstigere.	Optimierung	Nice to have

II.1.6.2 Implementierung eines FlexHubs in eine Testplattform

Mittels der zu entwickelnden Kommunikationslösung sollte der Datenaustausch zwischen einer Clientanwendungen (als steuernde Einheit) und einer FNN-konformen Steuerbox realisiert werden. Da die Box allerdings nicht verfügbar war, wurde abweichend von der ursprünglichen Projektplanung die Steuerbox bereits in der Entwicklungsphase nutzen zu können, ein 3-Stufen Modell bis hin zur vollständigen Implementierung umgesetzt.

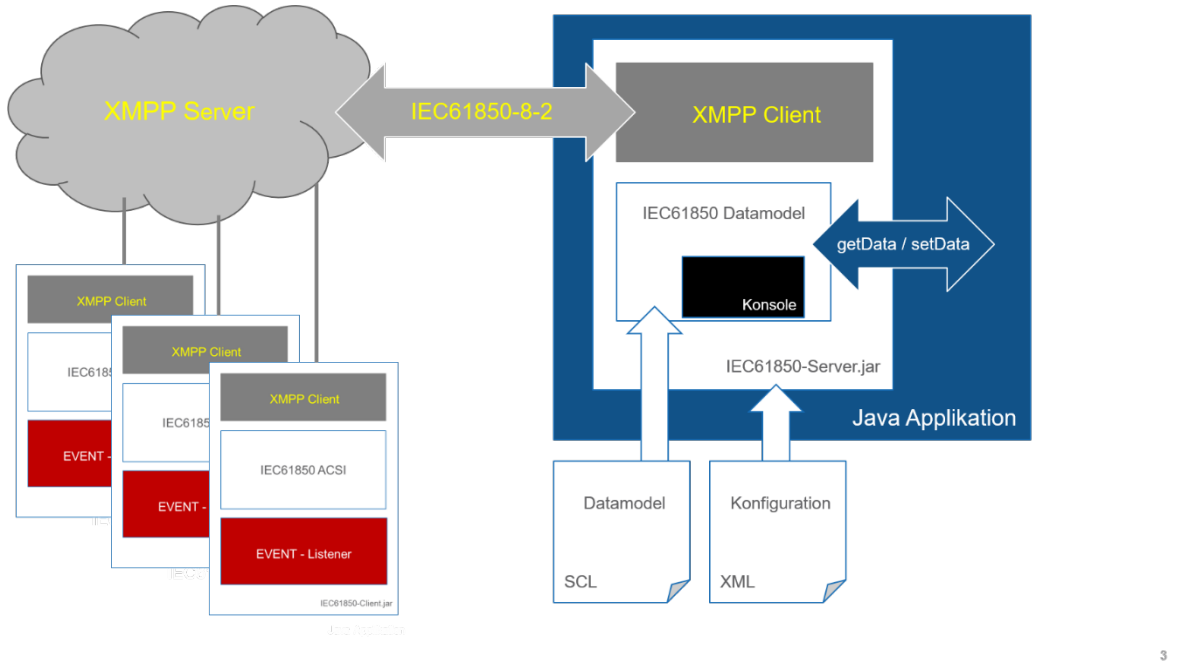
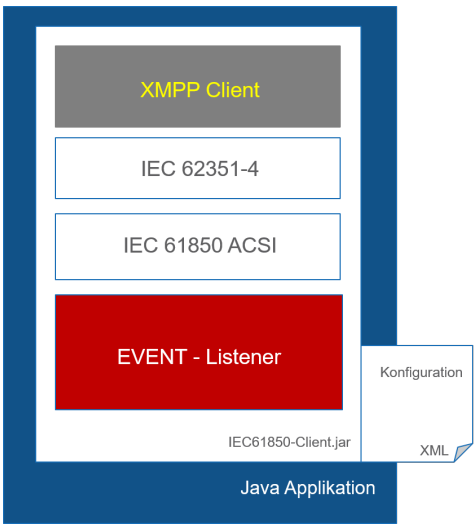


Abbildung 1

In Stufe 1 wurde zur autonomen Cliententwicklung ein IEC 61850 SCL-Server (virtueller Server) mit Kommunikationsschnittstelle XMPP (Extensible Messaging and Presence Protocol) nach IEC 61850-8-2 entwickelt. Das enthaltende Datenmodel konnte über eine SCL Datei (Substation Configuration Language - XML Dateiformat, Aufbau konform zu IEC 61850-6) geladen werden. Mittels Konsoleneingaben oder aufsetzende Java Applikationen konnten Änderungen von Status- und Messwerten zu Simulationszwecken gesetzt werden, so dass das Verhalten der späteren Hardware möglichst realgetreu simuliert werden konnte.

Die Kommunikation zwischen Client und Server wurde rein auf der Ebene XMPP nach Normenteil 8-2 durchgeführt. D. h. auf jegliche Form von Kommunikationssicherheit wurde hier bewusst verzichtet, da vordergründig die Entwicklung aller, für das FlexHub Projekt notwendigen, Kommunikationsservices war.

Nach Implementierung aller Services auf der Ebene 61850-8-2 folgte In Stufe 2 der Stack Entwicklung die Integrierung der vorgegebenen Sicherheitsfeatures nach IEC 62351-4 (End2End Security) in die Clientanwendung. Abbildung 2 veranschaulicht und beschreibt hier den grundsätzlichen Aufbau.



- XMPP Client zum Austausch von IQ und Empfang von Message „Stanzas“
- IEC 62351-4 Layer zur Verschlüsselung, Signierung und Authentifizierung (E2E)
- IEC 61850 Abstract Communication Service Interface (ACSI): Serviceimplementierung
- EVENT-Listener: Java Interface zum Empfang von Reject oder Report Nachrichten (unsolicited services)
- Client Konfiguration (XMPP Zugangsdaten, Zertifikate etc.) via XML Konfigurationsdatei

Abbildung 2

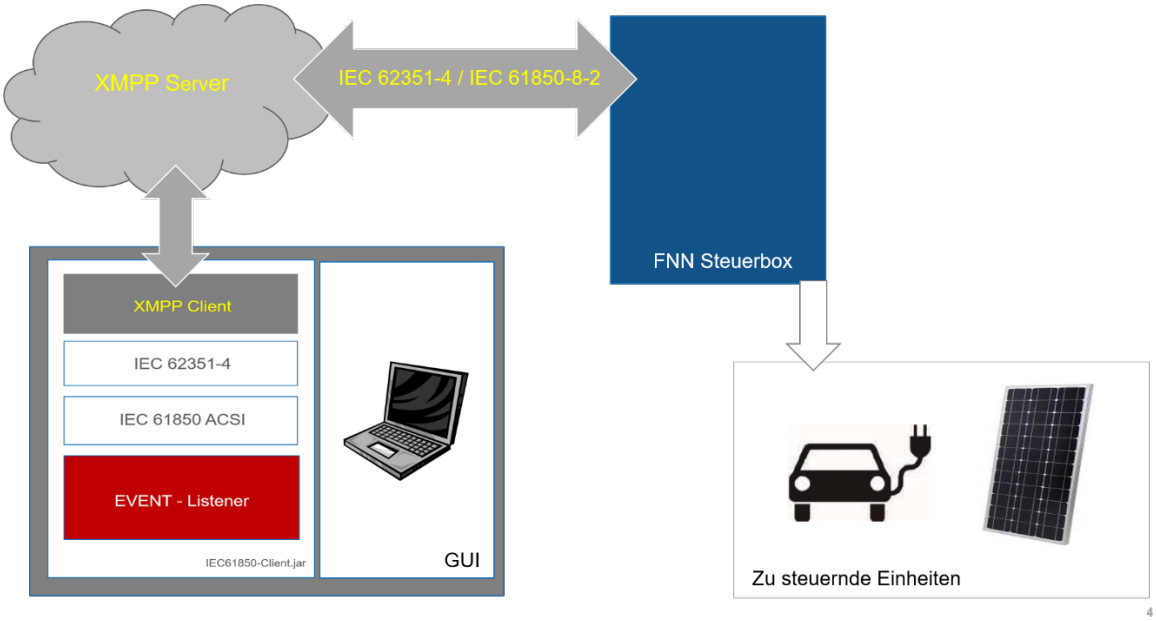


Abbildung 3

Abbildung 3 zeigt die finale Entwicklung, Grundlage der Laborversuche bei der RWTH Aachen wurde.

Das in AP 4.2 als Prototyp entwickelte hierarchische Flexibilitätsdatenregister mit angeschlossenem Markt wurde für die Weiterführung im Projekt als geeignet angesehen und an das Kiwigrid-System angebunden. Die Projektpartner konnten mithilfe von REST-Schnittstellen oder über ein grafisches Frontend mit einer Test- und Produktivinstallation des Registers interagieren, um erste Tests und

Integrationsversuche durchzuführen. Im Rahmen dieses Arbeitspakets wurden die Aufgaben der HAW Hamburg im Unterauftrag durch die EnergieDock GmbH übernommen. Der im Rahmen von AP4.2 entwickelte Prototyp für ein verteiltes Flexibilitätenregister mit Marktfunktion IRES (Intelligent Registry for Energy Services) wurde im Rahmen dieses Arbeitspaktes auf Grundlage der in AP4.1 entwickelten Architektur eigens neuentwickelt. Diese Neuentwicklung eines verteilten Flexibilitätenregisters mit Marktfunktionalität trägt den Namen NEMO.spot.

NEMO.spot bildet die White-/Yellow-Page Struktur des Datenmodells für einen verteilten FlexHub ab und teilt sich auf in ein Stammdatenregister, dass die White Page Informationen der verteilten Energieresourcen enthält und über eine API zur Verfügung stellt, den eigentlichen Flexibilitätenmarkt, der auf Basis der Yellow Page Informationen die eigentlichen Energiedienstleistungen (Angebot von Lade-Flexibilität) enthält und einen Optimierer, der die kleinteiligen Flexbilitätenangebote für den VNB aggregiert und so eine *One-Click* Lösung für Engpass-Anforderungen liefert. Zusätzlich hat der VNB über eine Validierungs-API die Möglichkeit jede Verschiebung von Flexibilität zu validieren und kann diese ggf. ablehnen, wenn dadurch neue Engpässe in seinem Netz entstehen würden. Abbildung 4 stellt die Architektur der NEMO.spot Plattform dar.

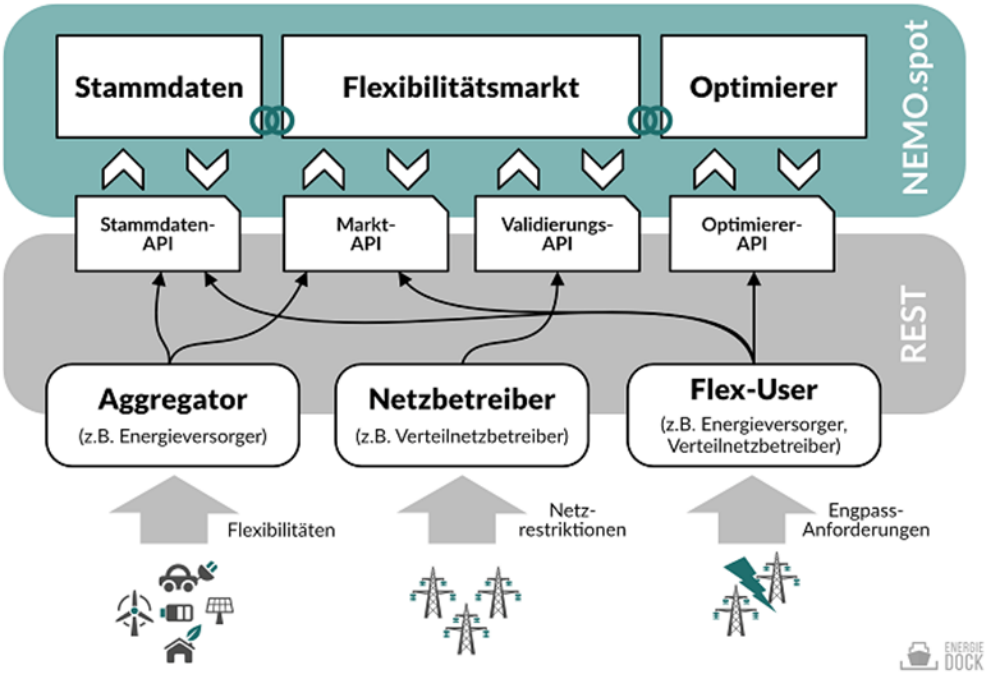


Abbildung 35: Architektur der NEMO.spot Plattform

Im Rahmen von Integrationstests wurde vor dem Beginn des Feldversuchs die erfolgreiche Anbindung des Kiwigrid-Systems an die NEMO.spot Plattform im Laborumfeld von Kiwigrid nachgewiesen. Dabei wurde das Einstellen, Suchen, Buchen und Steuern von Flexibilität anhand von E-Autos sowohl über einen manuellen Prozess als auch automatisiert durch Verwendung des Optimierers erfolgreich demonstriert. Außerdem wurde die Validierungsschnittstelle für den VNB erprobt und nachgewiesen,

wie dieser Flexibilitätsbuchungen validieren oder ablehnen kann, um die Entstehung von Netzengpässen in seinem Netzgebiet zu verhindern.

Um die korrekte Funktionalität aller Schnittstellen der NEMO.spot Plattform sicherzustellen wurden automatisierte API-Tests durchgeführt. Dazu wurden entsprechende Test-Cases für jeden Endpunkt und jede Schnittstelle entwickelt. Die Test-Cases decken dabei sowohl das korrekte Verhalten mit validen Eingabedaten als auch das korrekte Verhalten im Fehlerfall mit invaliden Eingabedaten ab. Die Spezifikation erfolgte dabei zur besseren Nachvollziehbarkeit in textueller Form gemäß dem BDD-Paradigma (Behavior Driven Development). Dazu wurde das Open-Source-Tool *Karate*[4] eingesetzt, dass sich insbesondere für das Testen von REST-Schnittstellen mit einem JSON-Datenmodell wie NEMO.spot eignet. Karate ermöglicht es dabei die Testfälle in einer menschenlesbaren BDD-Syntax zu beschreiben und automatisiert auszuführen. Eine Übersicht der erstellten Testfälle findet sich in Anhang 6. Die dort nach Endpunkt aufgelisteten Testfälle lassen sich wie folgt auf die in Abbildung 2 dargestellten Stammdaten-, Markt-, Validierungs- und Optimierer-API abbilden:

Tabelle 8: Mapping von REST-Endpunkten auf die APIs

Endpunkt	**API(s)**	**Enthaltene Test-Cases**
/bookings	Markt, Validierung	66
/businessentities	Stammdaten	22
/controls	Markt	39
/dersystems	Stammdaten	17
/flexoffers	Markt	74
/pricings	Markt	12
/rem	Optimierung	55
/reports	Markt	17
/sessions	(Authentifizierung und Autorisierung)	12
/settlements	Markt	25
/status	(Technisch)	1

[4] https://github.com/karatelabs/karate, Karate Webseite, zugegriffen am 22.11.2022

/transactions	Markt	1
/users	Stammdaten	19

Außerdem wurde im Rahmen dieses Arbeitspakets von der EnergieDock eine OpenAPI[5]-basierte technische Dokumentation erstellt. Diese Schnittstellenbeschreibung bildet detaillierte alle verwendeten Datenmodelle ab und listet alle Schnittstellen-Endpunkte auf, beschreibt deren erwartete Eingangsdaten, ihre Funktionalität, die Zugriffsberechtigungen, die erwarteten Rückgabedaten sowie die Fehler, die auftreten können. Die komplette Schnittstellenbeschreibung findet sich in Anhang 7.

In AP 6.2 wird der FlexHub auf Basis des blockchain-basierten Proof of Concept Demonstrators aus AP 4 in eine Testplattform für Labortest weiterentwickelt. Damit wird es ermöglicht den FlexHub unter Laborbedingungen zu erproben. Die Implementierung des blockchain-basierten Demonstrators erfolgt auf Basis des Proof of Concept Demonstrators mit Berücksichtigung der Anforderungen aus AP 6.1 zur Funktionstauglichkeit des FlexHub.

Hieraus ergeben sich folgende Aufgaben für die Fraunhofer Institute:

1. Erweiterung des Blockchain-Netzwerks um zusätzliche Knoten bei Projektpartnern
2. Implementation weiterer blockchain-spezifischer Konzepte
3. Implementation weiterer Smart Contracts und Auktionsmechanismen
4. Ergänzung und Sicherung der Implementation, so dass diese für Konsortiums- und öffentliche Blockchain-Plattformen geeignet ist

Ergebnisse:

Zu 1-4:

Die Funktionalität des entwickelten blockchain-basierten FlexHub wird mithilfe einer Testplattform auf die Probe gestellt. Die Testplattform umfasst zwei Anwendungen, die miteinander zusammenwirken:

1. eine Testsuite, die generierten Datensätze als Anfragen an die FlexChain Sandbox APIs sendet; hier ist zwischen einer „einfachen Testsuite“, die jede Runde ein einzige Flex-Angebot und VNB-Anfrage an die APIs sendet, und einer „fortgeschrittenen Testsuite“, die immer steigende Mengen an Anfragen sendet.
2. eine Monitoring-Plattform, die aus Prometheus und Grafana besteht. Prometheus ist eine Zeitreihendatenbank, die Daten aus dem FlexChain Netzwerk in festen zeitlichen Abständen abfragt. Grafana, eine grafische Monitoring Dashboard, die relevanten Metriken aus dem Prometheus als Grafiken anzeigt.

[5] Die OpenAPI Specification ist ein Standard zur Beschreibung von REST-konformen Programmierschnittstellen (API).

Testszenarien

Es wurden Testszenarien recherchiert und die Anzahlen an Stromnetzbetreiber und Flexibilitätsanbieter deutschlandweit, sowie der Energiebedarf bzw. die verfügbare Energie wurden identifiziert.

Mecklenborg [61] baut ein Testszenario mit dem Ziel auf, ein Worst-Case-Szenario darzustellen. Der Anzahl an Stromnetzbetreiber deutschlandweit wurde ermittelt und liegt bei 883 zum Stand Oktober 2020. Weitere abgeleiteten Annahmen zu diesem Anwendungsfall sind der Tabelle 9 zu entnehmen.

Annahme	Wert
Anzahl Verteilnetzbetrieber	883
Flexibilitäts-Energie von EMS Anbietern pro Tag (Durschschnitt)	42,17 kWh
Bedarf Flexibilitätsangebote pro Tag (Durschschnitt)	2,93 Mio
Buchungsrate von Flexibilitätsangeboten (Durchschnitt)	33,9 TPS
Flex-Angebote: Verteilnetzbetreiber	3,256

Tabelle 9: Annahme Worst-Case Szenario von Mecklenrung [81]

Schirrmacher [54] baut einen ähnlichen Testszenario für einen Flexibilitätsmarkt auf, in dem ein Hyperledger Sawtooth Blockchain Netzwerk mit Konsensalgorithmen Proof-of-Elapsed-Time und Practical-Byzantine-Fault-Tolerance getestet wird. Die deutschlandweiten Testszenario besteht aus den Daten in der Tabelle 10.

Annahme	Wert
Anzahl Verteilnetzbetreiber	883
Anzahl Flexangebote pro VNB pro Tag (durchschnitt)	3327
Anzahl DERs pro VNB (durchschnitt)	10k
Gesamte Anzahl DERs	8830k

Tabelle 10: Annahme Worst-Case Szenario von Schirrmacher [54]

Testsuites

Basierend auf den gewonnenen Erkenntnissen wurde eine Testsuite in Python entwickelt, die das FlexChain Sandbox mit Testszenarien von verschiedenen Größenordnungen testet. Dabei liegt der Fokus auf die Überprüfung der FlexChain APIs Funktionalität, sowie auf die Skalierbarkeit und Performance des FlexChain Netzwerks.

Die Testsuite simuliert den Flexibilitätsmarkt über mehrere Wochen hinweg, indem Anfragen an die verschiedene APIs in FlexChain Sandbox gesendet werden. Die dafür verwendeten Datensätze werden im Vorfeld generiert. Der Benutzer kann Listen als CSV-Dateien von DERs, Flex-Angebote, VNBs, VNB-Anfragen mithilfe von Python Skripten generieren und diese als Ausgangspunkt für die Testsuite nehmen.

Es gibt zwei Testsuites, eine „einfache“ und eine „fortgeschrittene“ Variante. Die einfache Variante läuft in einer Schleife und sendet jeweils ein einziges Flex-Angebot und VNB-Anfrage an die jeweiligen APIs. Die fortgeschrittene Testsuite fügt eine einstellbare Anzahl an Flex-Angebote bzw. VNB-Anfragen in die APIs hinzu, indem das Day-Ahead Verfahren simuliert wird.

Die Testsuite ist als Fraunhofer-internes GitLab Projekt vorhanden. Die Testsuite wurde auf einem Server durchführt, wo Debian Linux im Betrieb ist. Der Server besitzt 2 CPUs und 4GB RAM.

Einfache Testsuite

Bei der einfachen Testsuite werden folgende Anfrage an die REST APIs gesendet:

1. „VNB registrieren“ Anfrage an die VNB API (3. Quorum Knoten im FlexChain Sandbox)
2. „DER registrieren“ Anfrage an die EMS API (2. Quorum Knoten im FlexChain Sandbox)
3. „Flex-Angebot hinzufügen“ Anfrage an die EMS API
4. „VNB-Anfrage hinzufügen“ Anfrage an die VNB API
5. „Fahrpläne berechnen“ Anfrage an die VNB API
6. „Fahrplan abrufen“ Anfrage an die EMS API

Die Daten für die Anfragen 1-4 sind konstant, während die Anfrage 5-6 funktioniert mit den Angebot- und Anfrage-Ids, die im Laufe der Anfragen 3-4 zugewiesen werden. Die „einfache Testsuite“ wird in endloser Schleife ausgeführt bis der Benutzer sie explizit stoppt.

Fortgeschrittene Testsuite

Für die fortgeschrittene Testsuite sind Datensätze bereits vor der Ausführung der Testsuite vorberechnet. Es handelt sich hierbei um VNB-, DER-, Flex-Angebote- und VNB-Anfrage-Eingabedaten, die mithilfe von Python-Skripten generiert werden. Die Testsuite simuliert einen

Day-Ahead Flexibilitätsmarkt, indem Flex-Angebote und VNB-Anfragen für den nächsten Tag angemeldet werden. Anschließend werden die Anfragen zu dem entsprechenden simulierten Zeitpunkt an die APIs gesendet und verarbeitet. Die durch den Matching-Prozess gefundenen Flex-Angebote werden anschließend gebucht und dafür werden Fahrpläne berechnet. Die Testsuite registriert neue Flex-Angebote und VNB-Anfrage in einer Schleife bis der Benutzer sie explizit beendet.

Die Testsuite geht wie folgt vor:

1. Die Liste von DERs wird eingelesen und die DERs werden über die DER API im FlexChain Sandbox registriert.
2. Die Liste von VNBs wird eingelesen und die VNBs werden über die VNB API im FlexChain Sandbox registriert.
3. Während der reale Zeitverlauf simuliert wird, werden die Flex-Angebote und VNB-Anfragen zu den jeweiligen Zeitpunkten, einen Tag vor deren Aktivität, über die DER API bzw. VNB API in das FlexChain Sandbox hinzugefügt.
4. Zu den passenden Zeitpunkten werden die VNB-Anfragen über die VNB API im FlexChain Sandbox gesendet und verarbeitet.

Da die Istanbul BFT und Raft Konsensalgorithmen, durch deren unterschiedlichen Herangehensweise bei der Block-Schöpfung, die Ergebnisse der API-Anfrage beeinflussen, wurden die Testsuite in beiden Fällen ausgeführt und ausgewertet. Die resultierenden Ergebnisse sind unten präsentiert.

Ergebnisse nach der Anwendung von der „einfachen Testsuite“

Die einfache Testsuite bietet einen ersten Eindruck über die Performance vom FlexChain Sandbox. Es wurden zwei Metriken verwendet, um die Performance zu messen:

- Die Reaktionszeit der Anfragen: Hierbei handelt es sich um die Dauer einer Anfrage des Test-Clients, bis eine Antwort von dem FlexChain Sandbox APIs ankommt, d.h. die Latenzzeit der Transaktion zzgl. der API-Anfrage.
- Der entstandene Gas-Verbrauch bei den Quorum-Transaktionen, die durch die API-Anfragen ausgelöst wurden.

Um die Wirkung der beiden verfügbaren Konsensalgorithmen zu beobachten, wurde die „einfache Testsuite“ auf das FlexChain Sandbox angewendet, wenn Quorum mit Istanbul BFT und Raft verwaltet wurde. Darüber hinaus wurde die Art der Verbindung der Quorum Knoten ebenfalls variiert, indem die Knoten jeweils über HTTP-RPC und WebSockets angesprochen wurden. Der Verbindungstyp spielt eine grundlegende Rolle bei der Geschwindigkeit der Anfragen. Die Stabilität der entstandenen Verbindung zu den Knoten ist hierbei ebenfalls betroffen.

Ergebnisse der „einfachen Testsuite“ bei Istanbul BFT Konsensalgorithmus

Bei dem ersten Anlauf der Testsuite benötigten „VNB registrieren“ und „DER registrieren“ etwa 20,42 Sekunden bzw. 7,95 Sekunden, danach gab es Antworten zu den Anfragen nach jeweils circa 3 Sekunden. Dies ist darauf zurückzuführen, dass die VNB und DER APIs mehr Zeit benötigen bis sie die erste Anfrage beantwortet bekommen. Danach reguliert sich die Reaktionszeit auf circa 3 Sekunde, welche darauf zurückzuführen ist, dass die im Durchschnitt 3 Sekunden gewartet wird, bis einen neuen Block geschöpft wird (vgl. Abbildung 36)

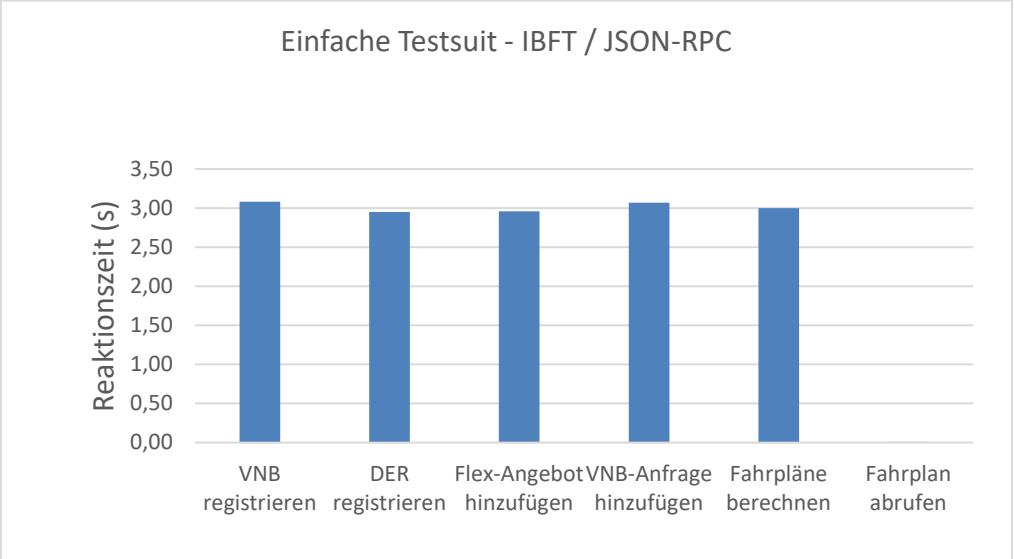


Abbildung 36: Die Reaktionszeiten der FlexChain Sandbox APIs (IBFT) auf die Anfragen der "einfachen Testsuite"

Die von API-Anfragen ausgelösten Transaktionen verbrauchten so viel Gas, wie es in der Abbildung 37 zu entnehmen ist. In der allerersten Schleife der Testsuite sind die Gas-Werte leicht höher, da extra Berechnungsaufwand in den Smart Contracts bei der Initialisierung entsteht, danach bleiben sie konstant bei den in der Abbildung angezeigten Werten. Auch wenn die VNB-Anfrage die höchste Berechnungsaufwand aufweist, welcher auf die genaue Implementation zurückzuführen ist, hat dies weiterhin keine weiteren Folgen auf die Komplexität der Transaktionen, wie es in den Ergebnissen der „fortgeschrittenen Testsuite“ zu sehen ist.

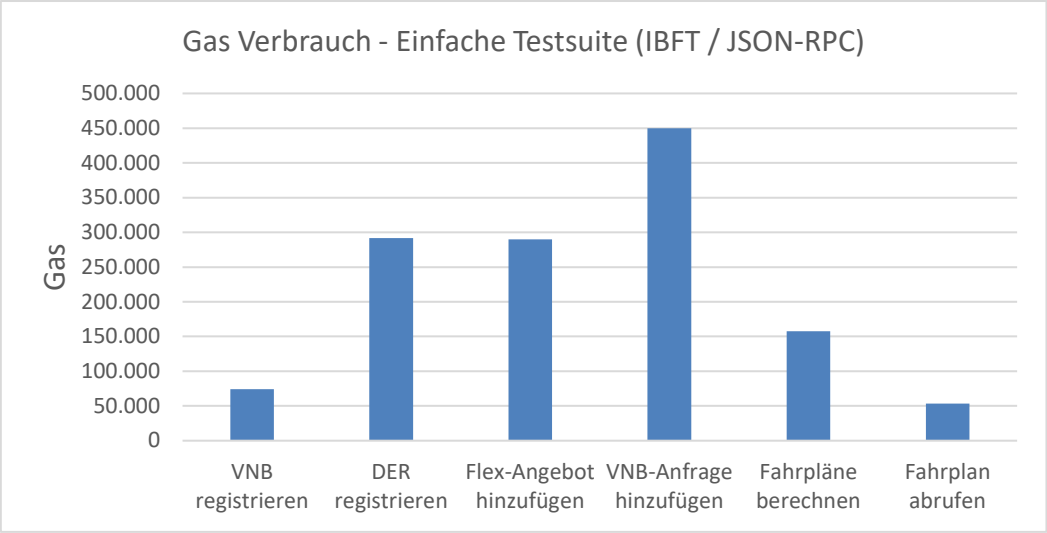


Abbildung 37: Gas Verbrauch der FlexChain Sandbox APIs (IBFT) auf die Anfragen der "einfachen Testsuite"

Ergebnisse der „einfachen Testsuite“ bei Raft

Wenn der Raft Konsensalgorithmus bei dem FlexChain Sandbox im Einsatz ist, sinken die Reaktionszeiten der API-Anfragen deutlich (vgl. Abbildung 38). Das ist auf die Funktionsweise vom Raft zurückzuführen, der Blöcke schöpft, sobald eine Transaktion im Netzwerk eingegangen ist. In diesem Falle wurden, wie auch bei Istanbul BFT, höhere Reaktionszeiten bei den ersten „VNB registrieren“ und „DER registrieren“ Anfragen vermerkt, da die jeweiligen APIs mehr Zeit bei deren ersten Anfrage benötigen. Sonst ist bei dieser Größenordnung keinen Unterschied zwischen den einzelnen Anfragen zu beobachten.

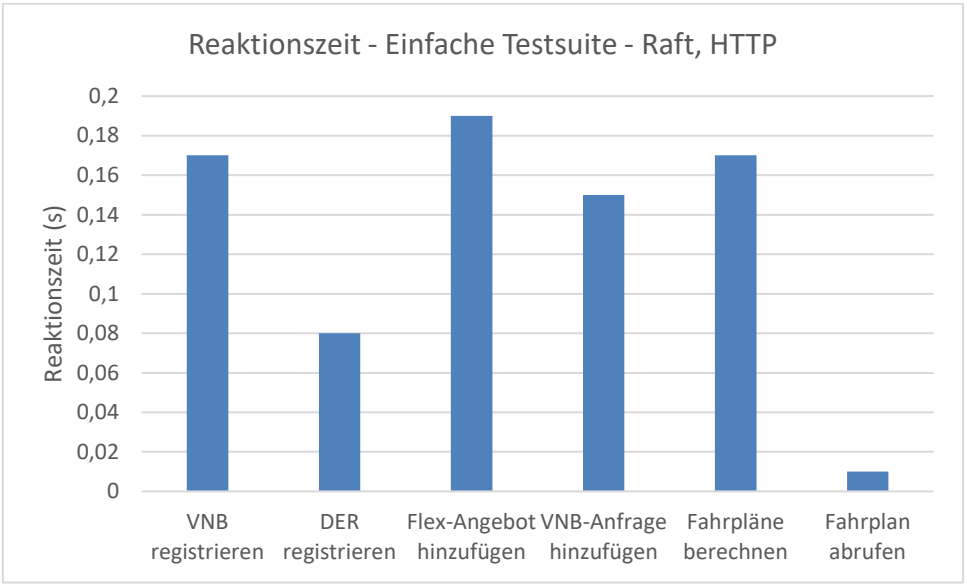


Abbildung 38: Reaktionszeiten der FlexChain Sandbox APIs (Raft) auf die Anfragen der "einfachen Testsuite"

Der Gas-Verbrauch hier fällt praktisch identisch aus wie im Falle von Istanbul BFT (vgl. Abbildung 39). Auf die Transaktionsebene haben die Konsensalgorithmen keinen Einfluss.

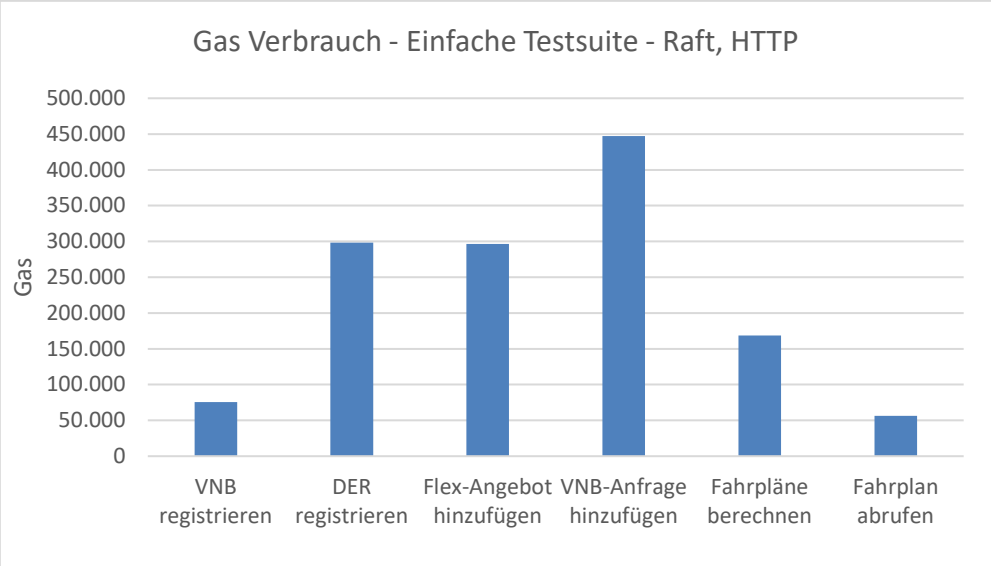


Abbildung 39: Gas Verbrauch der FlexChain Sandbox APIs (Raft) auf die Anfragen der "einfachen Testsuite"

Ergebnisse der „fortgeschrittenen Testsuite“

Anders als bei der einfachen Testsuite, wo die Matching-Funktion immer nur eine einzige Flex-Angebot berücksichtigen muss, werden bei der fortgeschrittenen Testsuite eine stetig ansteigende Anzahl an Flex-Angebote registriert und beim Matching mitberücksichtigt. Die anderen Anfragen verlaufen konstant, da es sich immer um das Verarbeiten von Eingabedaten geht, die konstant groß sind. Aus diesem Grund variiert bei der fortgeschrittenen Testsuite nur die „Fahrpläne berechnen“ Anfrage und deswegen wird ausschließlich diese Anfrage weiter ausgewertet.

Das Testen hat ergeben, dass die Reaktionszeit bei 38 Anfragen im Durschnitt 5,33 Sekunde misst, die Standardabweichung dabei liegt bei 1,11. Aufgrund der 5 Sekunden Blockzeit bestimmt die Reaktionszeit der API. In 35 der Fälle misst der Durchschnitt 5,05 (Standardabweichung liegt bei 0,17) und in drei weitere Fälle misst der Durchschnitt zwischen 6 und 9 Sekunden. Nach 38 „Fahrplan berechnen“ Anfragen wurde ein Timeout vermerkt, das aufgrund des hohen Gas-Verbrauchs bei den resultierenden Transaktionen entstand. Das benötigte Gas ist bei der Matching-Logik wegen der ansteigenden Anzahl an Flex-Angebote gewachsen. Der Timeout-Wert wurde auf 90 Sekunden gesetzt. In Abbildung 40 ist das Verhältnis zwischen diesen beiden Werten dargestellt. Der höchste erreichte Wert an Flex-Angebote war 311.

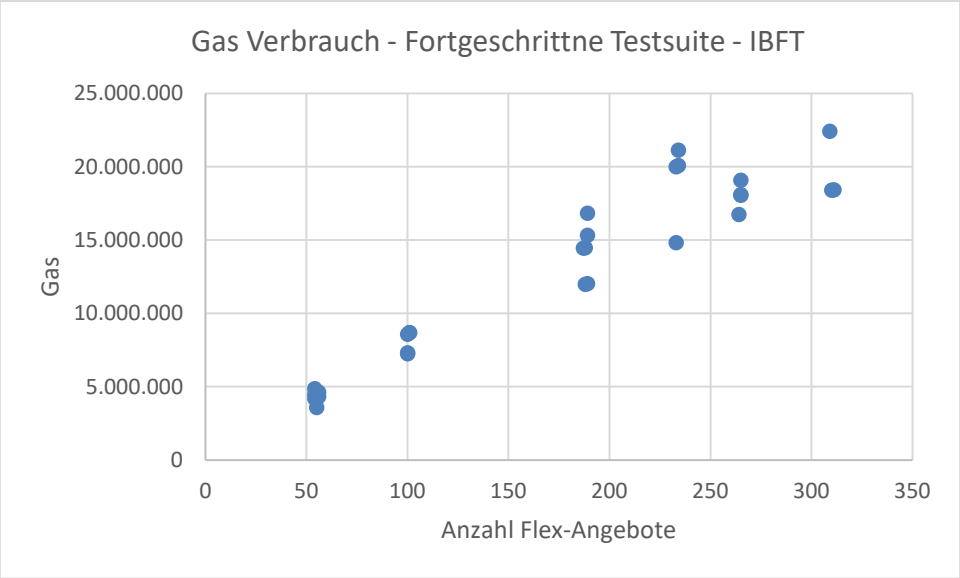


Abbildung 40: Der Gas Verbrauch gegen die Anzahl der Flex-Angebot bei der FlexChain Sandbox APIs (IBFT)

Im Falle von Raft wurde ebenfalls eine Obergrenze erreich, wo die Transaktionen den 90 Sekunden Timeout überschritten haben. Die letzte beantwortete Anfrage wurde bei 218 Flex-Angebote vermerkt (vgl. Abbildung 41):

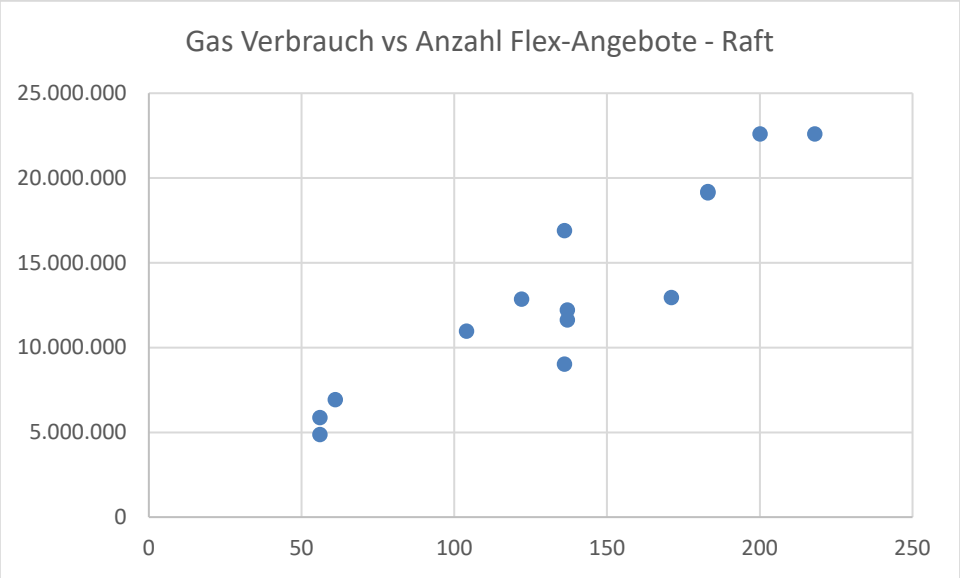


Abbildung 41: Der Gas Verbrauch gegen die Anzahl der Flex-Angebot bei der FlexChain Sandbox APIs (Raft)

Im Vergleich zu Istanbul BFT sind die Reaktionszeiten bei Raft deutlich niedriger. Die höchste Reaktionszeit wurde mit 2,28 Sekunde gemessen, bevor das Timeout von 90 Sekunden überschritten wurde. In Abbildung 42 ist der Verlauf der Reaktionszeit gegen die Anzahl der Flex-Angebote zu sehen.

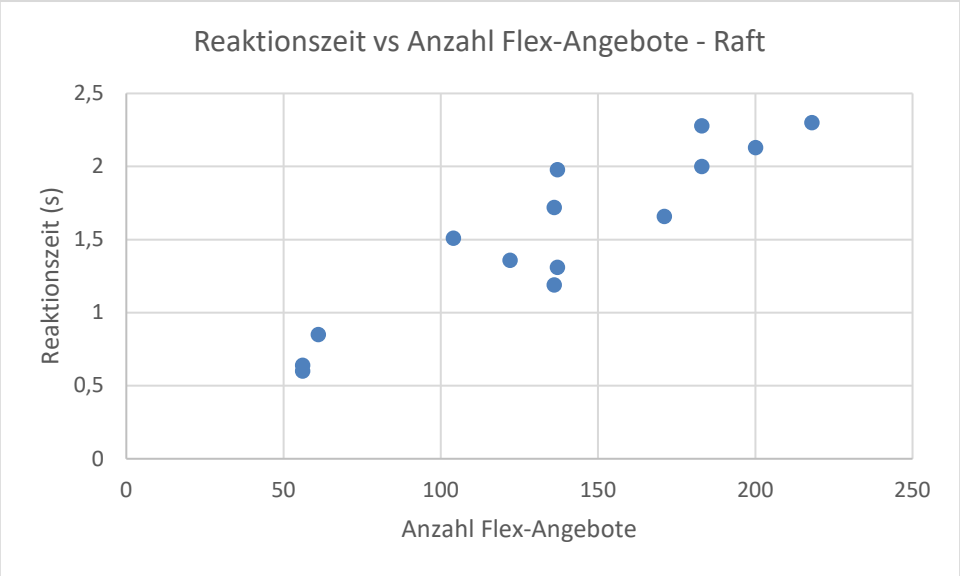


Abbildung 42: Die Reaktionszeit gegen die Anzahl der Flex-Angebot bei der FlexChain Sandbox APIs (Raft)

Ergebnisse bei WebSockets-Verbindung

Die Quorum-Knoten bieten die Möglichkeit, sich mit den Clients über HTTP-RPC oder WebSockets zu verbinden. Während HTTP-RPC das stabilere Protokoll ist, das gegen plötzliche Ausfälle geschützt ist, birg es jedoch einen Aufwand, der sich in den Kommunikationszeit ausschlägt. Die Verbindung über WebSockets ist hingegen unkomplizierter, verzichtet auf Handshakes und ist deswegen schneller für den verbundenen Client. Nach dem Testen mit dem fortgeschrittenen Testsuite, fielen die Reaktionszeiten deutlich geringer aus und der Punkt, wann das Timeout erreicht wurde, war verlängert.

Im Falle von Istanbul BFT betrugen die Reaktionszeiten weiterhin 5 Sekunden, die der eingestellten Blockzeit gleicht, allerdings gab es einige Anfragen, die bis zum knapp 10 Sekunden liefen (vgl. Abbildung 43). Darüber hinaus stieg der Obergrenze bei der Anzahl an Flex-Angebote bis zu 366, bevor das Timeout von 90 Sekunden überschritten wurde.

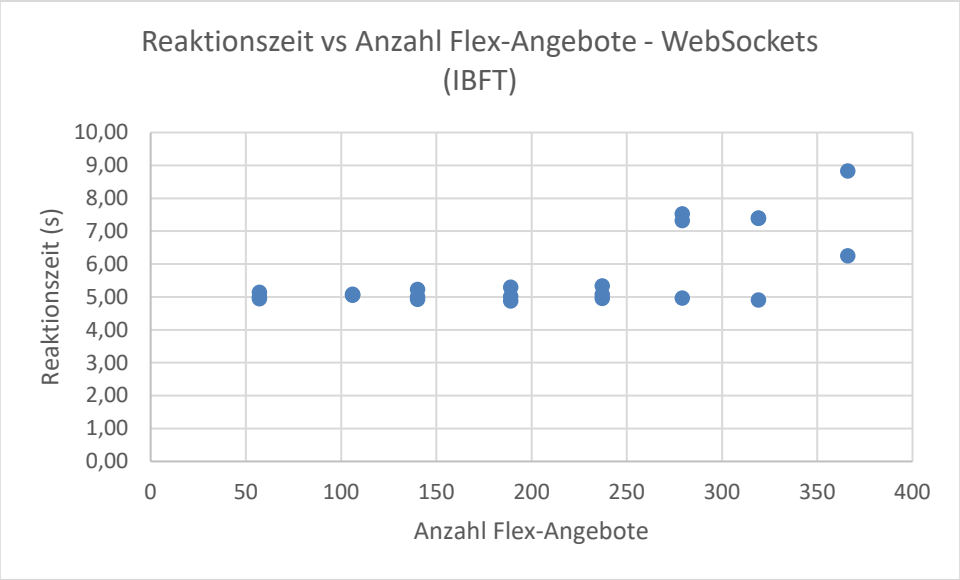


Abbildung 43: Die Reaktionszeit gegen die Anzahl der Flex-Angebot bei der WebSockets Verbindung (IBFT)

Währenddessen sind die Reaktionszeiten bei Raft über die WebSockets-Verbindung weiterhin deutlich geringer gewesen. Die höchste gemessene Reaktionszeit dabei war 2,21 Sekunden. Die Anzahl der Flex-Angebote stieg in diesem Falle bis zum 189 Flex-Angebote, nachdem die Anfragen das Timeout-Zeit überschritt (vgl. Abbildung 44).

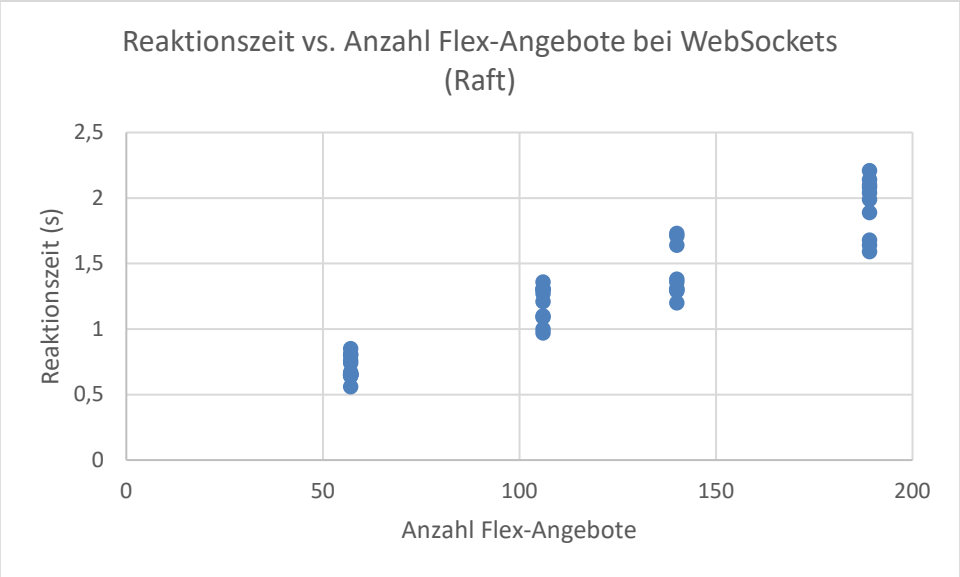


Abbildung 44: Die Reaktionszeit gegen die Anzahl der Flex-Angebot bei der WebSockets Verbindung (Raft)

Der Gas-Verbrauch im Falle der WebSockets-Verbindung gleicht den Werten, die über die HTTP-RPC Verbindung entstanden, denn die Transaktionen haben in beiden Fällen den gleichen Inhalt.

Die höchsten Werte an Reaktionszeit und Gas-Verbrauch sind in der Tabelle 11 zusammengefasst. Die Werte wurden bei einem erfolgreichen Abschluss der Anfrage (ohne Timeout) gemessen. Die Gas-

Verbrauchswerte sind, wie erwartet, sehr ähnlich, weil sich der Inhalt der Transaktionen nicht ändert. Die Werte, die über die WebSockets erreicht wurden, sind hierbei etwas höher. Die Transaktionsaufbau unterscheidet sich sonst bei den beiden Konsensalgorithmen wohl nicht. Einen Unterschied ist hingegen bei der Anzahl der Flex-Angebote zu merken. Demzufolge erreicht Istanbul BFT deutlich höhere Werte als Raft. Die WebSockets-Verbindung bei Istanbul BFT erreicht die beste Leistung mit 366 Flex-Angeboten, die bei der „Fahrpläne berechnen“-Anfrage verarbeitet wurden.

Konsensalgorithmus	Verbindungstyp	Höchste erreichte Anzahl an Flex-Angebote	Höchster entstandener Gas-Verbrauch
Istanbul BFT	HTTP-RPC	311	22.422.051
Istanbul BFT	WebSockets	366	23.068.893
Raft	HTTP-RPC	218	22.613.617
Raft	WebSockets	189	23.355.079

Tabelle 11: Übersich der höchsten erreichten Flex-Angebote und die bei den "Fahrplän berechnen"-Anfrage entstandenen Gas-Verbrauch

Ausfall von Knoten

Es wurde ein Testszenario simuliert, wo Knoten explizit gestoppt wurden, um die Auswirkung dieses Ereignisses zu beobachten. Wenn die Anzahl der verbleibenden Knoten größer oder gleich als die Mindestanzahl ist (für Istanbul BFT 4 Knoten, für Raft 3 Knoten), dann kann das Netzwerk den Konsens weiterhin erreichen und die Knoten synchronisieren nach wie vor die verarbeiteten Transaktionsdaten. In der Praxis sind die Knoten so eingestellt, dass sobald ein Fehler auftritt, wodurch ein Knoten zwingend herunterfahren muss, dann startet der Knoten automatisch so bald wie möglich wieder.

Prometheus und Grafana

Prometheus ist eine Zeitreihendatenbank, die Metriken bei dem FlexChain Sandbox in festen Zeitintervallen abfragt, d.h. Block, Transaktion und Netzwerk-Metriken (vgl. Tabelle 12). Der Zeitabstand zwischen den Abfragen ist bei 5 Sekunden eingestellt. Als Monitoring Plattform bietet Grafana die Visualisierung von Messwerten als Grafiken. Prometheus eignet sich hervorragend als Quelle für die Metriken, die hier in verschiedenen Dashboards dargestellt werden.

Prometheus Metrik	Bedeutung
eth_block_number	Aktuelle Block Nummer

eth_latest_block_transactions	Die Zahl der Transaktionen beinhaltet im letzten Block
net_peers	Anzahl der Knoten, mit denen der Knoten verbunden ist
eth_pending_block_transactions	Anzahl der Transaktionen, die darauf abwarten, in einen Block aufgenommen zu werden

Tabelle 12: Von FlexChain abgefragten Metriken

Das für FlexChain eingesetzte Grafana besteht aus zwei Dashboards:

1. Die Eth Dashboard, die Metriken zum Quorum Netzwerk in verschiedenen Grafiken anzeigt, vgl. Abbildung 45.

Abbildung 45: Das ETH Grafana Dashboard mit Metriken zum FlexChain Quorum Netzwerk

2. Die Server Dashboard, die Metriken zum Server, der einen Knoten hostet, darstellt wie z.B. Prozessor- und Arbeitsspeicher Auslastung, Speicherplatzverbrauch, Netzwerkverkehr, etc. (vgl. Abbildung 46). Mögliche Abhängigkeiten des FlexChain Sandbox mit der Serverleistung können hier beobachtet werden.

Abbildung 46: Das Grafana Server-Dashboard beinhaltet Metriken zu Prozessor, Arbeitsspeicher, Speicherplatz, Netzauslastung u.v.m.

II.1.6.3 Aufbau einer Testumgebung

Ein Teilziel des Projektes ist die Ertüchtigung einer bestehenden Laborinfrastruktur der RWTH Aachen University. Es soll die Möglichkeit geschaffen werden in einer definierten Testumgebung Konzepte für Marktplattformen wie den FlexHub zu integrieren und zu untersuchen. Hierzu ist es insbesondere notwendig adäquate Schnittstellen der Laborkomponenten zur Marktplattform – hier dem FlexHub – umzusetzen.

Die Ertüchtigung des Labors hat auf mehreren Ebenen stattgefunden. Einerseits die Konzeptionierung und Ausgestaltung der Hardware im Labor. Andererseits die kommunikationstechnische Anbindung der Komponenten. Die Anbindung erfolgt auf der einen Seite für die Einprägung energietechnischer Szenarien im Sinne von Kontrollschleifen und Zeitreihenbasierte Anlagensteuerung für Erzeuger und Einspeiser. Auf der anderen Seite erfolgt die Anbindung des FlexHub für die Marktkommunikation. Das allgemeine Konzept basieret auf einer modularen Hard- und Softwareumgebung. Diese eröffnet zum einen die Möglichkeit im Rahmen des Projektes unterschiedliche Konfigurationen zu evaluieren und

bietet zum anderen die Möglichkeit der Weiterverwendung bzw. Erweiterung im Rahmen weiterer Forschungsvorhaben. Das Gesatmkonzept wird in Abbildung II-47 schematisch dargestellt.

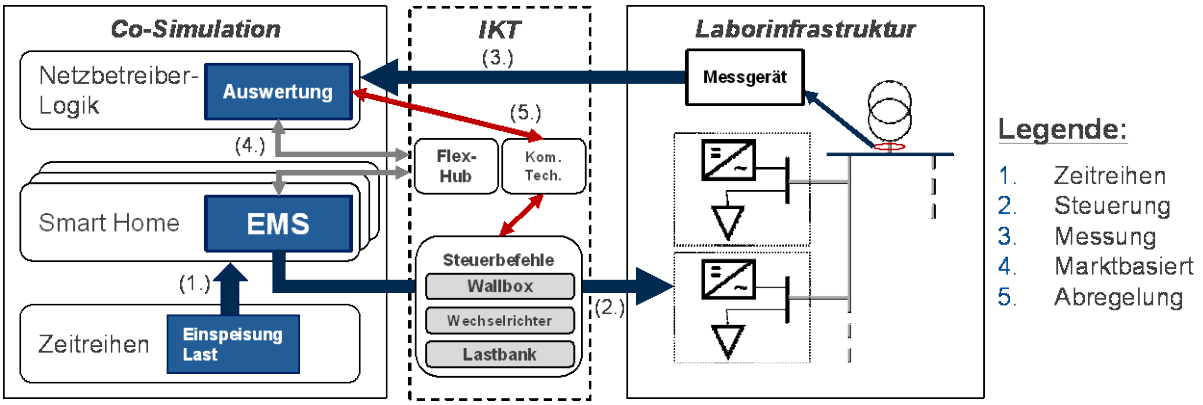


Abbildung II-47: Schematischer Aufbau der gesamten Laborinfrastruktur (Hard- und Software) [1]

Die Kommunikationsschnittstellen sind ebenfalls als modulare Adapter ausgelegt, sodass unterschiedliche Kommunikationsstrecken zwischengeschaltet werden können. Damit auch die Steuereinheit als wohldefinierte Komponente mit expliziten Schnittstellen im Labor integriert werden kann, wurde die Steuerlogik und die Kommunikationsadapter auf dem Einplatinencomputer RaspberryPi ausgebracht. Dabei erfolgt die Kommunikation mit der FlexHub-Plattform und die Übertragung der Steuerbefehle für die Anlagen über getrennte Netzwerkschnittstellen. Im Rahmen des Projektes wurde unter anderem ein Kommunikationsadapter für das ModbusTCP Protokoll implementiert, der insbesondere für die Anlagensteuerung und das Auslesen von Messdaten genutzt werden kann. Das Scheduling der gesamten Testumgebung inklusive der Einprägung der Zeitreihen erfolgt als zeitabhängiges Co-Simulations-Konzept mittels der Opensource Software *mosaik*. Sowohl die Software für das Scheduling als auch der FlexHub laufen auf dedizierten Servern. In Abbildung II-48 wird der schematische Aufbau der kommunikationstechnisch angebundenen Komponenten zusammengefasst dargestellt.

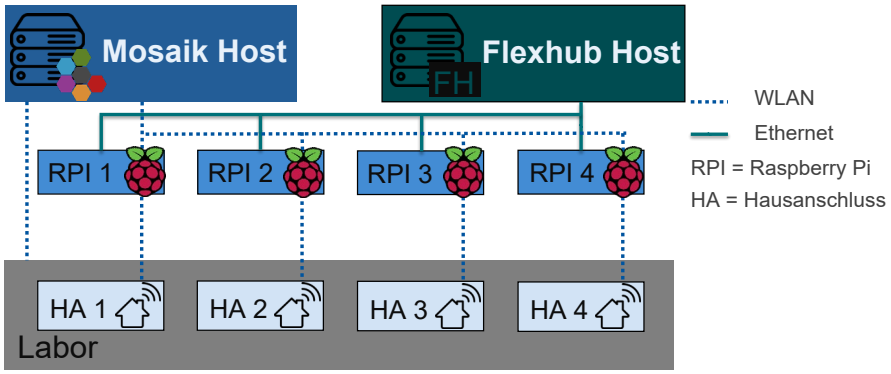


Abbildung II-48: Schematischer Aufbau der Kommunikationsschnittstellen

Die Zusammenstellung untersuchter Hausanschlusskonfigurationen ist im Rahmen der verfügbaren technischen Geräte (Inverter, Lasten, Schaltmöglichkeiten) prinzipiell frei wählbar. Für die Vergleichbarkeit ist im Rahmen des Projektes eine exemplarische Netzkonfiguration mit den zugehörigen

Behind-the-Meter Analgen definiert, die für alle Versuchsreihen verwendet wird. Die Netzkonfiguration bestehend aus vier Hausanschlüssen ist in Abbildung II-49 skizziert. An zwei Abgängen werden je zwei Hausanschlüsse angeschlossen und von einem 630 kVA Transformator gespeist.

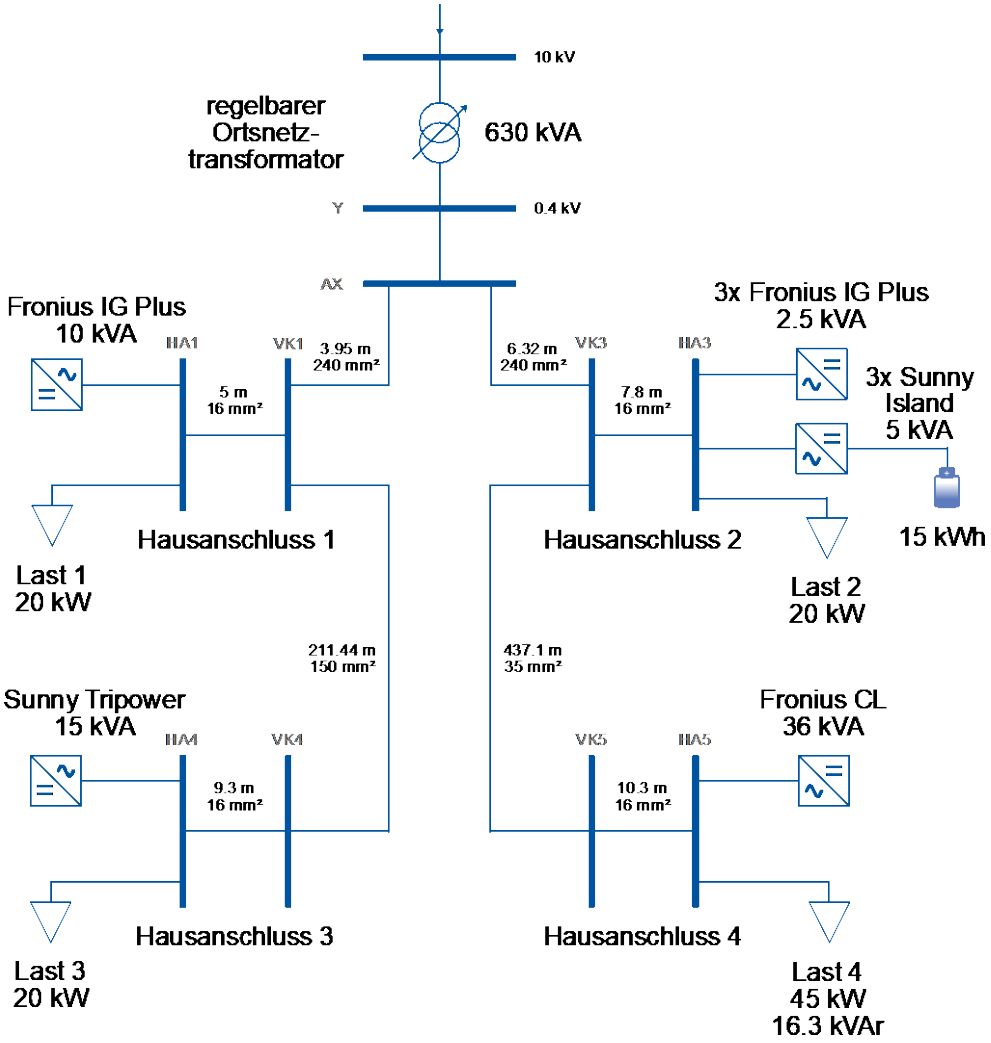


Abbildung II-49: Exemplarische Netztopologie [2]

II.1.7 Arbeitspaket 7: Betrieb des FlexHub und Demonstrationsversuche

Über einen Zeitraum von ca. 4 Monaten mit 20 Kunden wurde der Use Case getestet. Grundsätzliche wurde die Komplexität im Verlauf des Demonstrators erhöht und in Monatsscheiben ausgewertet.

II.1.7.1 Vorbereitung und Inbetriebnahme Feldversuch

Grundsätzlich lässt sich die Vorbereitung des Feldversuchs in fünf Themenschwerpunkte unterteilen.

- Geeignete Technik wurde ausgewählt, getestet und beschafft.
- Kriterien für geeignete Pilotkunden wurden definiert, Kunden identifiziert und der zu testende Use Case kundenorientiert adressiert.
- Identifizieren und schulen von Rahmenvertragsfirmen und Servicepartnern.
- Die Installation und Konfiguration der Technik vor Ort beim Pilotkunden wurde dann durch geeignete Rahmenvertragsfirmen in die vorhandene Kundeninstallation gebracht.
- Festlegung von geeigneten Testszenarien.

Bei den technischen Hauptbestandteilen handelt es sich um eine Wallbox und ein Energiemanagementsystem. Hier kam eine KEBA Wallbox X-Series P30 zum Einsatz. Unsere Wahl fiel auf dieses Modell, da wir bereits im Labor und bei friendly costumer vor Ort positive Erfahrungen mit der KEBA-Wallbox gesammelt hatten. Grundvoraussetzung war, dass es sich um eine intelligent Wallbox handelt, welche LAN, als auch WLAN-Kommunikation unterstützt. Die Bestellung der Wallboxen erfolgte in dem Zeitraum, in dem Wallboxen durch die KfW gefördert wurden. Daraus resultierte eine erhöhte Nachfrage, welche mit längeren Lieferzeiten verbunden waren.

Des Weiteren kam ein Energiemanagementsystem zum Einsatz. Dieses ist notwendig, um die Backendlogik mit der Wallbox bei dem Kunden vor Ort zu verbinden. Hier viel die Wahl auf das Energiemanagementsystem der Firma KiwiGrid, da wir den Use Case mit diesem Modell bereits vorab erfolgreich getestet hatten.

Eine Besonderheit hierbei war, dass das Energiemanagementsystem bei dem Kunden im Zählerschrank auf die Hutschiene montiert und dann via LAN-Kabel in das Kundennetzwerk integriert wird. Die Abfrage der technischen Rahmenbedingungen bei dem Kunden haben jedoch ergeben, dass die meisten Kunden weder Platz auf der Hutschiene noch einen LAN-Anschluss in Reichweite zum Zählerschrank hatten. Es musste eine Möglichkeit gefunden werden, das Energiemanagementsystem in Reichweite des Kundenrouters und nicht in Nähe des Zählerschranks zu installieren. Folglich haben wir das Energiemanagementsystem und das zugehörige Netzteil in eine handliche Box verbaut. Wir verwendeten eine transparente Box, um die Sichtbarkeit auf die Status-LEDs weiterhin zu gewährleisten und führten LAN-Kabel und Schuko Stecker aus der Box heraus. Somit war es möglich das Energiemanagementsystem direkt via LAN und Schuko Stecker in Reichweite des Routers zu verbinden. Ohne diese Lösung wären 90% der interessierten Kunden nicht für den Feldtest geeignet gewesen. Wichtige

Erkenntnis an dieser Stelle ist die Tatsache, dass die wenigsten Kunden über eine LAN-Verbindung in Nähe des Zählerschranks verfügen und größtenteils auch keinen Platz auf Ihrer Hutschiene für ein Zusatzgerät wie ein Energiemanagementsystem hatten. Die Praxis hat außerdem gezeigt, dass es ratsam ist das Energiemanagementsystem vorab auf den aktuellen Softwarestand zu bringen, da dieses Update vor Ort beim Kunden unnötig Zeit beansprucht hätte.

Um das Interesse der Kunden zu wecken, wurde vorab ein kundenorientierter Flyer mit den notwendigen Informationen zum Use Case und ein Anschreiben erstellt. Es wurden ca. 100 Kunden kontaktiert, von den ca. 30 Kunden ihr Interesse an einer Pilotteilnahme bekundet hatten. Mit einer Rückmeldequote von ungefähr 30% waren wir durchaus zufrieden.

Für die interessierten Kunden wurden dann Informationsveranstaltungen organisiert. Bei diesen Veranstaltungen wurde den interessierten Kunden der Feldtest im Detail erläutert, der energiewirtschaftliche Gesamtkontext nähergebracht und die Zielstellung besprochen. An dieser Stelle der Kundenakquise wurden auch erstmalig Benefits für den Kunden durch eine Teilnahme am Piloten kommuniziert. Neben der Möglichkeit die Wallbox nach dem Feldtest günstig zu erwerben, bekam jeder Kunde eine Prämie in Höhe von 100 €. Nachdem alle offenen Fragen der Kunden geklärt wurden, waren von den ursprünglichen Interessierten 20 Kunden weiterhin am Feldtest interessiert.

Neben der Technik und den Pilotkunden als solches, werden Rahmenvertragsfirmen und Servicepartner benötigt. Diese bringen die Technik ins Feld und betreuen den Kunden während der Testdauer. Daher wurden Schulungsunterlagen, für die die Kolleginnen und Kollegen der Service-Hotline erstellt und Use Case bezogene Schulungen durchgeführt, um Fragen der Pilotkunden während der Feldtestphase innerhalb kundenorientierter Servicezeiten kompetent beantworten zu können.

Als Verteilnetzbetreiber arbeitet man mit zahlreichen Rahmenvertragsfirmen langfristig zusammen. Dennoch stellte es eine Herausforderung dar geeignete Rahmenvertragsfirmen zu identifizieren, die Interesse haben an solch einer Pilotierung inkl. Feldtest mitzuwirken. Wir haben uns dafür entschieden, zwei Rahmenvertragsfirmen zu beauftragen und diese entsprechend ihrem regionalen Firmensitz auf die Netzregionen der MITNETZ sinnvoll zu verteilen. Für die Rahmenvertragsfirmen wurde eine detaillierte technische Dokumentation bezogen auf Technik und zu konfigurierender Software erstellt. Das weiteren wurden die Kolleginnen und Kollegen entsprechend von uns geschult.

Nachdem nun alle Akteure den notwendigen Wissenstand hatten, konnten die Termine beim Pilotkunden vor Ort vereinbart werden. Wir haben uns gemeinsam mit den Rahmenvertragsfirmen entschieden zwei Termine mit dem Kunden zu vereinbaren. Im Fokus des ersten Termins stand der Vorab-Check. Dabei wurden durch die Rahmenvertragsfirmen die technischen Rahmenbedingungen vor Ort geprüft. Die Vorab-Checks führten zu dem Ergebnis, dass einige Kundeninstallationen vor Ort als für den Feldtest nicht geeignet bewertet wurden.

Gründe dafür waren bspw.:

- Sicherung befand sich in der Wallbox und nicht im Zählerschrank. Da die Keba-Wallbox keine eigene Sicherung innerhalb der Wallbox besitzt, hätte der Zählerschrank um eine Sicherung erweitert werden müssen. Bei den betroffenen Kunden war dafür jedoch kein Platz vorhanden.
- Trotz vorab angefragter Qualität des WLAN-Empfangs, stellt sich die Situation vor Ort als unzureichend dar.
- Kundenanlagen wurde im Vorabcheck als nicht ordnungsgemäß bewertet. Aus Gründen der Haftung wurde hier von den Rahmenvertragsfirmen keine Veränderung der Anlage herbeigeführt
- Die Wallbox und das Energiemanagementsystem müssen sich im selben Netzwerk befinden müssen. Aufgrund von Unwissenheit der Kunden über ihre eigene Netzwerkkonfiguration war die erfolgreiche Konfiguration in einigen Fällen nicht möglich.

Schlussendlich konnten bei 12 Pilotkunden Technik und Software erfolgreich ins Feld gebracht werden.

Im AP 7 wurde die in AP6.2 entwickelte Flexibilitätenplattform zur Anbindung im Feldtest bereitgestellt. Im Rahmen der Plattformlösung wurde eine Aggregator-Applikation zur Anbindung und Nutzung durch Netzbetreiber entwickelt, sodass im Feldversuch mögliche Lösungsstrategien für Netzengpassszenarien identifiziert und evaluiert werden können. Über die entwickelte Weboberfläche konnte der Netzbetreiber Netzengpassszenarien entweder als Microsoft Excel Datei beschreiben und hochladen oder direkt in der Weboberfläche das Szenario definieren. Über eine georeferenzierte Eingrenzung der zu berücksichtigen Flexibilitäten konnte sichergestellt werden, dass nur solche Flexibilitäten herangezogen werden, die im Gebiet des potenziellen Netzengpasses liegen.

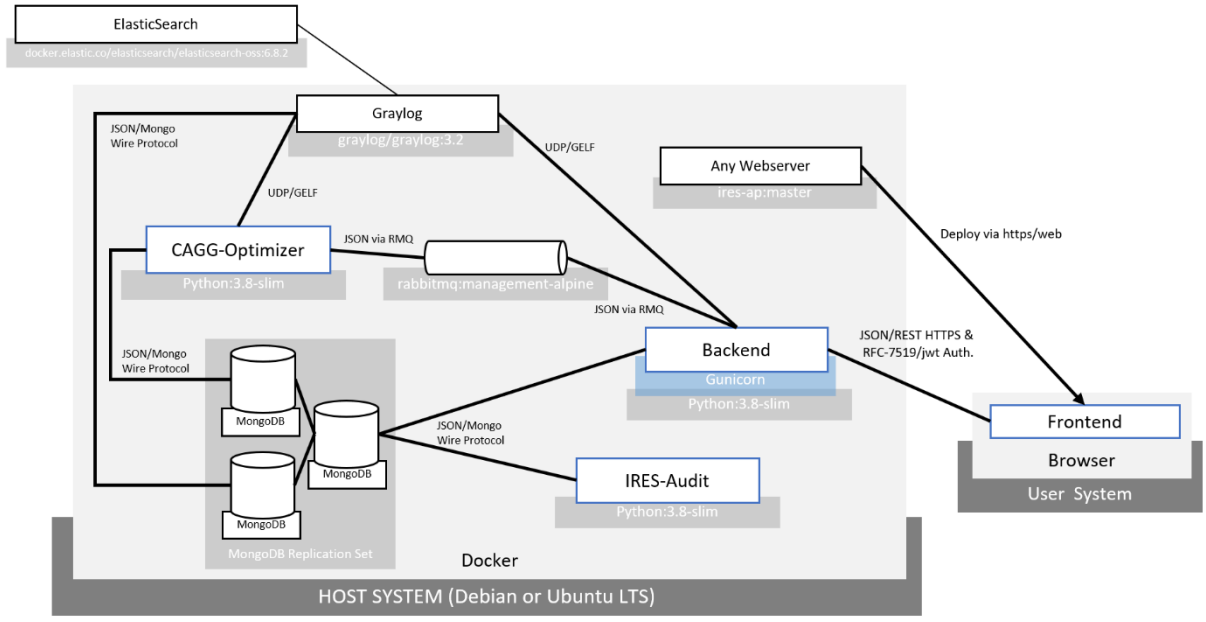


Abbildung 50: Deployment NEMO.spot Plattform Feldtest

Abbildung 5 zeigt die Deployment-Situation der NEMO.spot Plattform für den Feldtest. Alle entwickelten Komponenten werden als Docker Images bereitgestellt und zur Ausführungszeit entsprechend als

Docker Container auf einem Linux-Hostsystem ausgeführt. Kern der Plattform ist die Backend-Komponente, die zustandslos zentraler Access-Point für die Funktionen des Systems ist. Die API bietet dabei REST-Schnittstellen mit JSON-Datenmodellen abgesichert über HTTPS an. Diese Schnittstellen werden vom Kiwigrid System genutzt, um Flexibilitätenangebote aus der Kiwi-Cloud für die HEMS von Kiwigrid zu erstellen und im Falle einer Buchung, die entsprechenden Fahrpläne abzurufen. Die eigentlichen Daten (White- und Yellow-Page Informationen) über die Flexibilitäten liegen sicher repliziert in einem MongoDB Replication Set. An dieser Stelle ist auch eine hierarchische Organisation der Daten möglich. Für den Feldtest wurde aufgrund der überschaubaren Datenmengen auf eine solche hierarchische Organisation verzichtet. Alle Änderungen an den Daten werden in einer zusätzlichen Audit-Komponente festgehalten, um jegliche Änderungen und Prozesse nachvollziehen zu können. Über eine Message-Queue kommuniziert der Aggregator (CAGG-Optimizer) mit dem Backend und bedient so Aggregationsanfragen des Verteilnetzbetreibers. Das Frontend stellt die eingangs erwähnte Weboberfläche zur Verfügung, mit der Anbieter und Nachfrager von Flexibilität mit dem System interagieren können. Dabei benutzt es in dem Fall der manuellen Interaktion die gleichen Schnittstellen des Backends, wie sie im Falle einer automatisierten Interaktion z.B. im Zusammenspiel mit dem Kiwigrid-System verwenden werden.

Außerdem wurde im Rahmen dieses AP eine Bedrohungsanalyse für die Plattform erstellt und die entsprechenden Maßnahmen zur Absicherung des Feldtests vorgenommen. Die ausführliche Bedrohungsanalyse, die eine vollständige Systembeschreibung inkl. Der Prozesse und Assets beinhaltet, das Bedrohungsmodell vorstellt, Gegenstrategien entwirft und eine Risikobewertung vornimmt findet sich im Anhang dieses Berichts (A5).

II.1.7.2 Durchführung von Verifikationsversuchen im Labor

Die im Rahmen des Projektes entwickelte Testumgebung wird insbesondre für die Untersuchung des blockchain-basierten FlexHub genutzt. Die durch das Fraunhofer FIT entwickelte Blockchain-Lösung wird als Service auf der Serverinfrastruktur im Labor der RWTH Aachen ausgebracht. Die eingeführten Netzbetriebsszenarien werden im Standardversuchsaufbau, der ebenfalls zuvor eingeführt wurde, eingeprägt. Der mit den Expert*innen des Fraunhofer FIT entwickelte Versuchsaufbau unter Berücksichtigung der Blockchain, die über eine REST-API an die IT-Infrastruktur angebunden ist, ist in Abbildung II-51 für ein exemplarisches Untersuchungsszenario dargestellt.

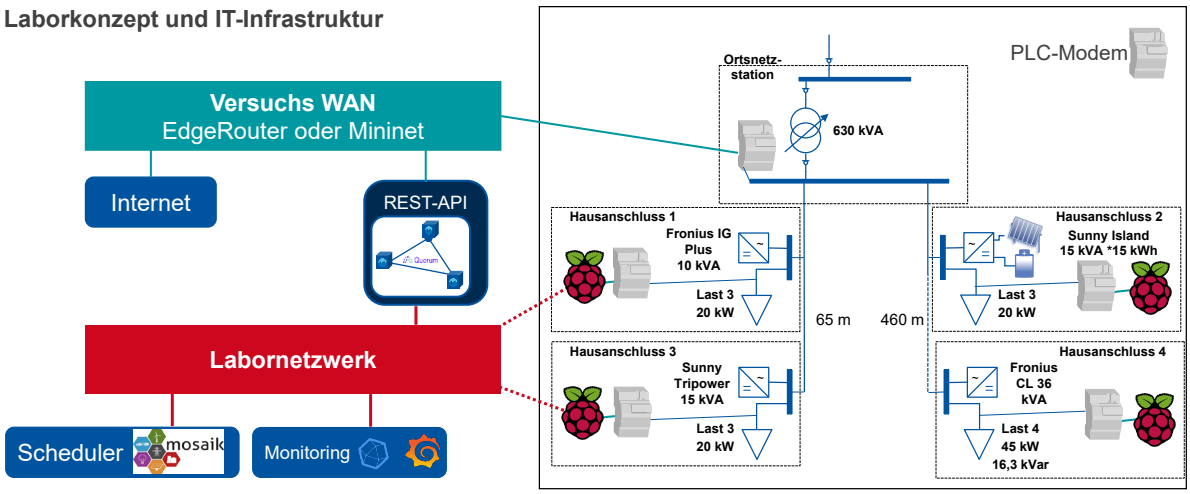


Abbildung II-51: Versuchsaufbau zur Untersuchung des blockchain-basierten FlexHub

Zunächst wird der Normalbetrieb als Referenzszenario „Benchmark“ untersucht. Anschließend wird unter Einsatz des FlexHub eine Flexibilitätskoordination durchgeführt. Dabei findet für den dargestellten Hausanschluss ein Abruf von 10 kW über 3 Stunden angefragt über den FlexHub statt (vgl. Abbildung II-52). Der Leistungsabruf wirkt sich entsprechend auf die lokale Auslastung – in diesem Fall die ONS – aus (vgl.

Abbildung II-53). Damit konnte gezeigt werden, dass unter realen Rahmenbedingungen in einer Laborumgebung der über den FlexHub koordinierte Flexibilitätsabruf im Echtzeitbetrieb möglich ist. Dazu gehört die kommunikationstechnische Anbindung der Anlagen und des abrufenden Netzbetreibers sowie die fahrplangetreue Umsetzung des Leistungsabrufes durch die Anlagen selbst.

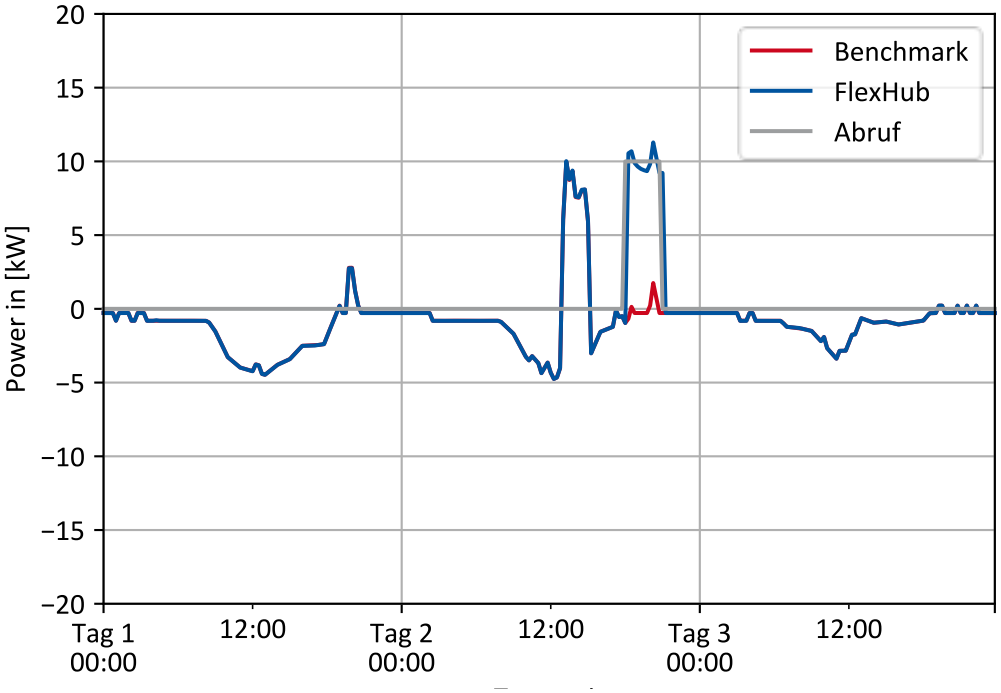


Abbildung II-52: Leistungsverlauf eines exemplarischen Hausanschlusses

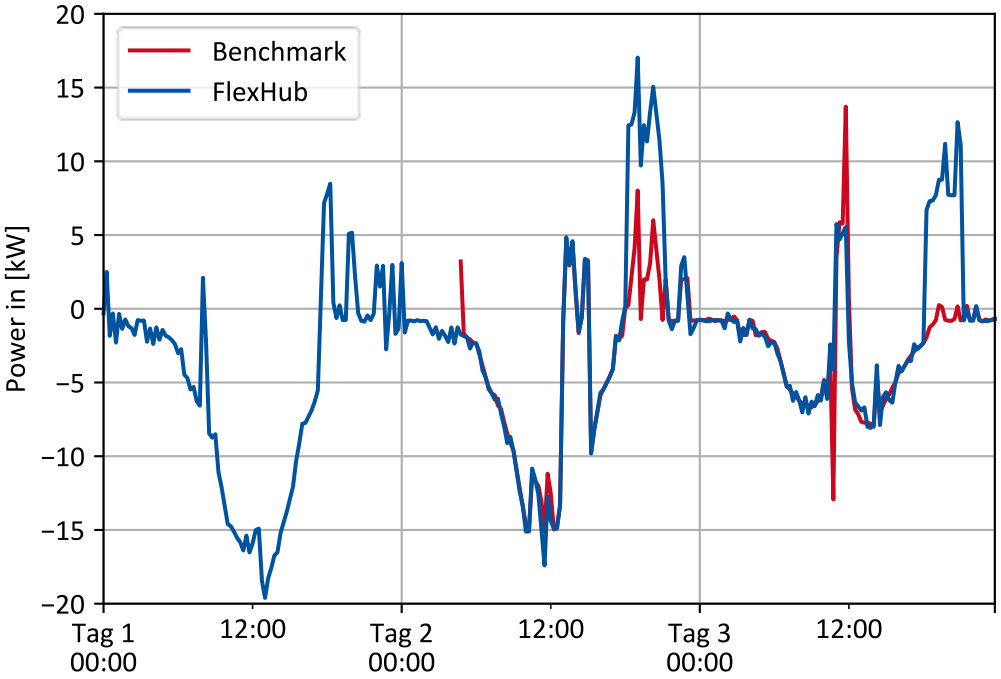


Abbildung II-53: Leistungsverlauf der ONS mit vier unterlagerten Hausanschlüssen

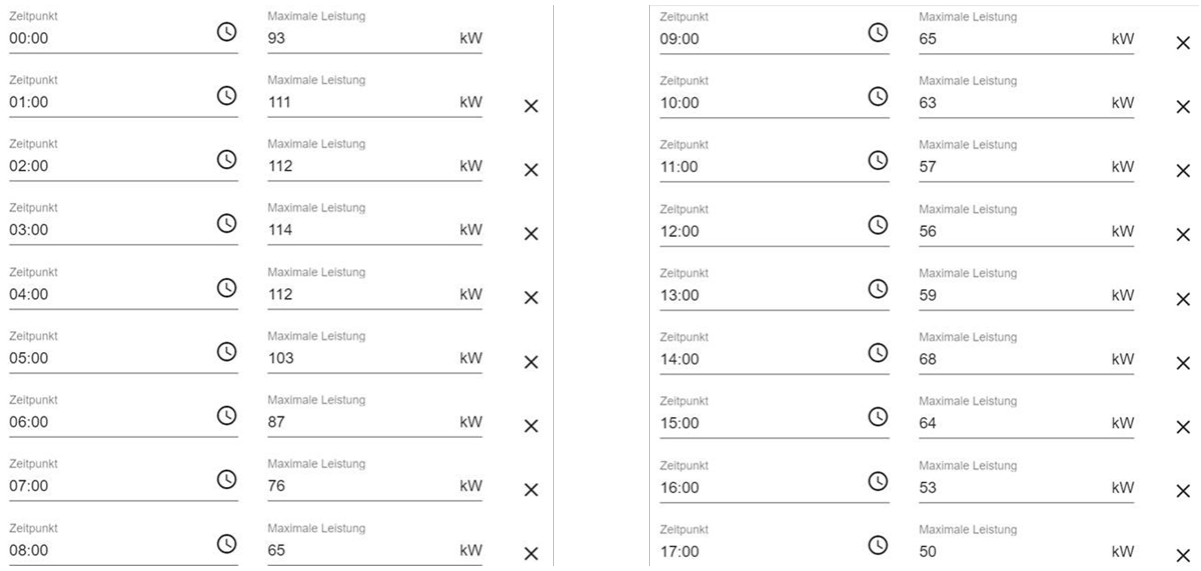
II.1.7.3 Begleitung und Auswertung des Feldversuchs

Während der gesamten Dauer des Feldversuchs gab es einen regelmäßigen Austausch via Mail und Telefon mit den Pilotkunden. Dabei stand vor allem die Lösung von technischen Problemen und die Erläuterung der zu testenden Szenarien im Vordergrund.

Folgende Testszenarien wurden dabei durchlaufen.

- Szenario 1 – 3 Wochen (02.05.22 bis 20.05.22)
  - Netz-Check-In only (zeitvariable Netzentgelte) ohne Engpass Niederspannung
- Szenario 2 – 7 Wochen (14.06.22 bis 30.06.22)
  - Netz-Check-In only (zeitvariable NNE) mit Engpass Niederspannung
- Szenario 3 – 4 Wochen (01.07.22 bis 31.07.22)
  - Netz-Check-In mit Optimierung Flexibilitätsmarkt ohne Engpass in der Niederspannung
  - VNB-Anfragen
    - Erste Woche: VNB-Nachfrage >= komplettes Flexangebot (alle Mutual Flexibility Agreements)
    - Zweite Woche: VNB-Nachfrage < komplettes Flexangebot (ausgewählte MFAs)
    - Flexmarktprämie für Pilotkunden abstimmen (FP1: 2 Cent/kWh; FP2: 3 Cent/kWh; FP3: 4 Cent/kWh)
- Szenario 4 – 4 Wochen (01.08.22 bis 31.08.22)
  - Netz-Check-In mit Optimierung Flexibilitätsmarkt mit Engpass in der Niederspannung
  - VNB-Anfragen definieren
    - Erste Woche: VNB-Nachfrage >= komplettes Flexangebot (alle MFAs)
    - Zweite Woche: VNB-Nachfrage < komplettes Flexangebot (ausgewählte MFAs)
    - Flexmarktprämie für Pilotkunden abstimmen (FP1: 2 Cent/kWh; FP2: 3 Cent/kWh; FP3: 4 Cent/kWh)

Hier die zugrundliegende Engpasssituation am Niederspannungsabgang der Pilotkunden. Grundlage für diese abgangsscharfe Restriktionskurve in der Niederspannung waren Messungen, welche wir bei 20 vergleichbare Ortsnetzstation verbaut hatten.



Nachdem die Vorbereitung bereits aufschlussreiche Erkenntnisse geliefert hat, war auch die Begleitung des Feldtests an sich von interessanten Erkenntnissen geprägt.

So reagieren unterschiedle Hersteller/Modelle von Elektroautos auch andersartig auf Ladepläne und Steuerbefehle. Das Thema Komfortladen, also das Vorwärmen der Batterie vor dem eigentlichem Ladeplan stellte uns in der Optimierung und Steuerung ebenfalls vor Herausforderungen. Diese wurden gelöst, indem dem Elektroauto neben dem optimalen Ladeplan auch außerhalb des Zeitfensters zum Laden ein minimaler Ladestrom ermöglicht wurde. Jedoch ist der Minimalladestrom zwischen den unterschiedlichen Modellen von Elektroautos verschieden und eine individuelle Konfiguration wurde notwendig. Ein weiterer interessanter Punkt war die Erkenntnis, dass zahlreiche Kunden nachts den Router vom Netz trennen oder Wallbox und EMS ausschalten, um Strom zu sparen. Da die Geräte dann offline sind, war es nicht in jedem Fall möglich erfolgreich zu kommunizieren.

Grundsätzlich wurde bei der Begleitung des Feldtests durch Kundengespräche deutlich, dass die Kunden an einer gesamthaften Optimierung ihres Hauses interessiert sind (PV optimiertes Laden, V2G, etc.)

Es konnte im Testfeld gezeigt werden, dass sich die Ladevorgänge der Elektroautos an den Anreizen in Sinne der zeitvariablen Netzentgelte als auch an den Anreizen aus dem Markt für Flexibilitäten orientieren. Neben der Anreizlogik wurde ebenfalls der Mechanismus des Netz-Check-In angewendet. Dabei wurde deutlich, dass Lasten in der Niederspannung entsprechend der auf Anreize optimierten Ladepläne der Elektroautos erfolgreich gegen die Niederspannungsrestriktion validiert werden konnten. Technik als auch Software haben sich dabei bewährt und als praxistauglich sowie skalierbar erwiesen. Auch die Pilotkunden haben Technik als auch Software als geeignet und benutzerfreundlich bewertet.

II.1.7.3.1 Feldversuch NEMO.spot KPIs

Der Feldtest unter Beteiligung der NEMO.spot Plattform (Flexibilitätenregister mit Marktfunktionalität) erstreckte sich in einem Zeitraum von 8 Wochen vom 03.08.2022 bis zum 25.09.2022. In diesem Zeitraum wurden:

- **11** E-Autos als verteilte Energieressourcen, die Flexibilitäten anbieten (White Page) erstellt, …
- die wiederum **414** Flexibilitätsangebote zum Verschieben ihres Ladeverhaltens erstellt haben und in …
- **51** Aggregationsanfragen zur Lösung eines Netzengpasses vom Verteilnetzbetreiber über die Weboberfläche von NEMO.spot verwendet wurden.
- Daraus resultierten **48** Buchungen, die je nach Anfrage **1 bis 10** Flexibilitätsangebote enthalten haben.
- (Für 3 Aggregationsanfragen konnte der Aggregator der NEMO.spot Plattform keine passende Lösung auf Basis der vorhandenen Flexibilitätsangebote finden, sodass es in diesem Fällen zu keiner Lösung und damit auch keiner Buchung kam.)
- In Summe wurden dabei **9,55 MWh** an flexiblen Ladeverhalten verschoben (ohne das die Endnutzer dabei in ihrem Komfort eingeschränkt wurden).

II.1.7.3.2 Feldversuch NEMO.spot Abbildung auf die Use Cases

Der Feldversuch unter Beteiligung der Flexibilitätsplattform NEMO.spot bildet unterschiedliche Teile des ersten und dritten Anwendungsfalls ab. Im zweiten Anwendungsfall findet keine Buchung und Steuerung der Flexibilitäten über eine Flexibilitätenplattform statt, sondern der Verteilnetzbetreiber sendet Preissignale direkt an die Energy-Management-Systeme, umso über diese Signale indirekt Einfluss auf das Lastverhalten der Flexibilitätsressourcen zu nehmen.

Anwendungsfall 1 und Anwendungsfall 3 entscheiden sich primär dadurch, dass in AF1 eine Flexibilität direkt mit einem spezifischen Fahrplan gebucht wird, der bereits zu diesem Zeitpunkt genau festlegt, wie sich die Flexibilitätsressource geladen wird. In AF3 wird während der Buchung zunächst nur das Recht auf Steuerung dieser Flexibilitätsressource im Rahmen des Flexibilitätsangebots erworben. Der Flexibilitätsnachfrager hat hier also die Möglichkeit nach Buchung der Ressource einen Fahrplan an diese Ressource zu senden und kann den Fahrplan bis zum Erbringungszeitraum auch noch anpassen. In AF1 muss sich der Flexibilitätsnachfrager bereits zum Buchungszeitpunkt auf einen Fahrplan festlegen. In AF1 erfolgt daher die Validierung durch den Verteilnetzbetreiber in seiner Rolle als Kapazitätsmanager während der Buchung und er kann die Buchung einer Ressource bereits zum Buchungszeitraum ablehnen, sodass die Buchung nicht zu Stande kommt (siehe Anhang 1 – Diagramm „Geradeausweg“). AF3 unterscheidet bei der Validierung von Buchungen zwischen der grünen und gelben Ampelphase im Netz. Während der grünen Ampelphase validiert der Kapazitätsmanager die Buchung an sich, ohne dass ein expliziter Fahrplan vorliegt (vgl. Anhang 3 – Figure 3). Wenn er die Buchung so frei gibt, kann der Flexibilitätsnachfrager die Ressource frei im Rahmen des Angebots steuern. In der gelben Ampelphase hingegen findet eine zusätzliche Validierung statt, wenn nach der Buchung vom Flexibilitätsnachfrager der konkrete Fahrplan für die Ressource hinterlegt wurde. Wenn ein Fahrplan hier

abgelehnt wird, bleibt die Buchung an sich bestehen und der Flexibilitätsnachfrager hat die Möglichkeit einen aktualisierten Fahrplan zu senden. Optimalerweise liefert der Kapazitätsmanager daher bei der Ablehnung eines Fahrplans eine Handlungsempfehlung bzw. einen Rahmen, in dem ein aktualisierter Fahrplan nicht abgelehnt werden würde (vgl. Anhang 3 – Figure 5).

Im Feldversuch wurde eine abgewandelte Validierungslogik aus Anwendungsfall 3 in der gelben Ampelphase umgesetzt, wobei auf die Validierung der Buchung an sich verzichtet wurde. Die Anfrage-Logik entsprecht dabei dem in AF1 abgebildeten Verhalten bei dem ein VNB als Flexibilitätsnachfrager eine Anfrage an einen Aggregator stellt, der diese Anfrage an das Flexibilitätenregister weiterleitet. Der Aggregator wurde dabei von Optimierungskomponente in NEMO.spot abgebildet (AF1, Schritte 1 – 6). Aus AF3 wurde die Aufteilung zwischen der eigentlichen Buchung und dem späteren Übersenden der Fahrpläne übernehmen. Allerdings wurde diese Logik vom NEMO.spot Optimierer automatisiert abgebildet. Der Optimierer agierend als Aggregator hat entsprechend der Anfrage vom VNB ein optimales Flexibilitätenportfolio zusammengestellt, diese Flexibilitäten gebucht und in einem zweiten Schritt automatisiert die Fahrpläne für diese Ressourcen auf der Plattform hinterlegt (entspricht AF3, Schritte 5 – 18). Die EMS der Ressource haben diese Fahrpläne vom Markt erhalten und sie an den VNB in seiner Rolle als Kapazitätsmanager geschickt. Hier erfolgte dann eine Validierung der Fahrpläne (AF3, Schritte 19 – 25). Im Rahmen des Feldversuchs konnte somit erfolgreich demonstriert werden, dass die Logik von AF1 und AF3 erfolgreich angewendet werden kann. Insbesondere die technische Integration der unterschiedlichen Systeme und die erfolgreiche Steuerung, ausgehend von der Anfrage des VNB, über den Aggregator/Optimierer der NEMO.spot Plattform, im Zusammenspiel mit der Marktkomponente der Plattform, weiter zu den Home Energy Systemen von Kiwigrid über die Kiwi-Cloud bis hin zu den tatsächlichen Flexibilitätsressourcen konnte erfolgreich demonstriert werden. Auch die Untersuchung, inwieweit der Ansatz der marktbasierten Bereitstellung von Flexibilität zum Netzengpassmanagement genutzt werden kann, konnte erfolgreich dargestellt werden. Allerdings mussten die entsprechenden Problemstellungen aufgrund der Beschränkung auf 11 Endkunden mit Flexibilität entsprechend herunterskaliert werden. Aus technischer wie auch konzeptioneller Sicht spricht allerdings nichts dagegen, dass der Ansatz nicht auch mit wesentlich mehr Teilnehmer funktioniert. Gerade im Bereich der Optimierung und Aggregation können mit mehr Freiheitsgeraden durch mehr verfügbare Ressourcen noch bessere Ergebnisse erwartet werden.

II.1.7.3.3 Feldversuch NEMO.spot Beispiel Aggregationsanfrage

Beispielhaft wird an dieser Stelle die Aggregationsanfrage vom 25.08.2022 beschrieben, die in Abbildung 6 dargestellt ist. Zwischen 18:00 Uhr am 25.08.2022 und 06:00 Uhr am 26.08.2022 hat der Verteilnetzbetreiber hier 22 kW an Leistung (entspricht 264 kWh Energie) angefragt (blaue Linie). In diesem Bereich sollte also optimalerweise die Ladung der E-Autos erfolgen, wobei die 22 kW Leistungsabnahme aus Gründen der Netzstabilität nicht überschritten werden sollten. Mit der Abnahme der benötigten Energie in diesem Zeitraum kann z.B. verhindert werden, dass die Einspeisung von Erneuerbaren Energien in diesem Zeitraum abgeregelt werden müsste. (Wobei natürlich zu beachten ist, dass der Feldversuch prototypenhaft mit einer kleinen Benutzerbasis und damit eingeschränkten Flexibilitätspotential durchgeführt wurde.)

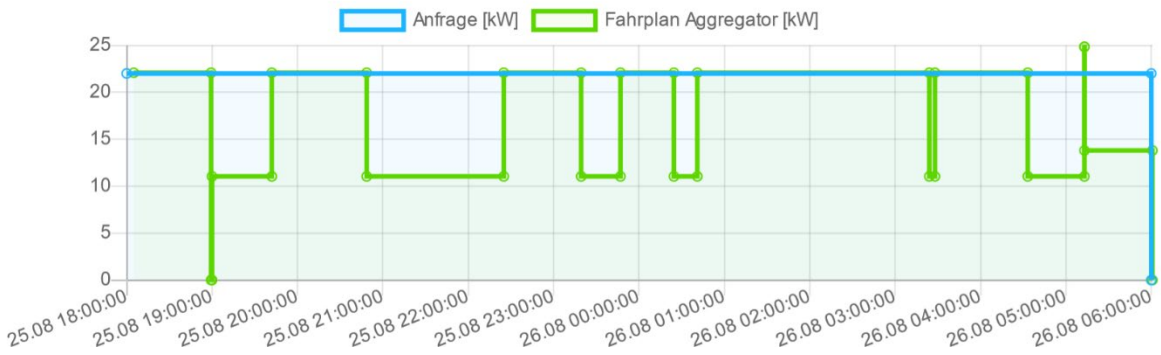


Abbildung 54: Screenshot NEMO.spot Weboberfläche Aggregationsanfrage und Lösung vom 25.08.2022

Im Rahmen des Feldversuchs wurde die Aggregationsanfrage manuell über die Weboberfläche von NEMO.spot erstellt und löste dann den automatisierten Aggregationsprozess aus, bei dem der Aggregator die Flexibilitäten auswählt, die aufgrund der vorgegebenen Parameter (z.B. Geolokation, Leistungs- und Energiebezug, Angebots- und Flexibilitätszeitraum) geeignet sind. Der Aggregator optimiert diese dann so, dass die Auswahl, die gewählten Ladezeitpunkte und die bezogene Leistung eine möglichst optimale Annäherung an den Zielfahrplan sind. Erfüllen dabei mehrere Flexibilitätsangeboten die gleichen Anforderungen wählt der Aggregator das günstigere von beiden aus. Über die NEMO.spot Plattform und erfolgt dann eine automatisierte Buchung der ausgewählten Flexibilitäten und es werden jeweils die optimalen Einzelfahrplänen für die entsprechenden Ressourcen hinterlegt, die dann von den technischen Systemen des Flexibilitätsproviders (in diesem Fall Kiwigrid) abgerufen werden können. Die grüne Linie in Abbildung 6 zeigt die Lösung des Aggregators für diese Anfrage, die 8 Flexibilitätsangebote aggregiert hat. Abbildung 7 zeigt die einzelnen Fahrpläne für diese Flexibilitätsangebote (MFA – Mutual Flexability Agreement). Diese laden ab dem abgebildeten Startzeitpunkt mit der abgebildeten Leistung, bis die aufgeführte Energiemenge erreicht ist.

Aggregierte MFAs

MFA	Startzeitpunkt	Leistung [kW]	Energie [kWh]	Preis [€]
6305dc3b90b460dec4a4ff8f	2022-08-25 18:05:00	22,08	20	0,6000
6305dc3290b460dec4a4ff8e	2022-08-25 19:00:00	11,04	20	1,4000
6305dc3c90b460dec4a4ff90	2022-08-25 19:42:00	11,04	40	2,8000
6305dc3f90b460dec4a4ff92	2022-08-25 22:25:00	11,04	22	0,6600
6305dc4090b460dec4a4ff93	2022-08-25 23:47:00	11,04	60	1,8000
6305dc3754d64ac7de2e8d2b	2022-08-26 00:41:00	11,04	30	0,9000
6305dc44f537e6db4a23db2a	2022-08-26 03:28:00	11,04	12	0,3600
6305dc3d90b460dec4a4ff91	2022-08-26 05:13:00	13,8	11	0,3300

8 total

Abbildung 55: Screenshot NEMO.spot Weboberfläche Darstellung der einzelnen Fahrpläne als Teil der Lösung einer Aggregationsanfrage.

II.1.8 Arbeitspaket 8: Ableitung von Handlungsempfehlungen

Der Feldversuch konnte belegen, dass eine Kombination von Lastvalidierung und Flexibilitätsmarkt hinsichtlich der Integration und Nutzbarmachung von flexiblen Lasten in der Niederspannung zwei grundlegende Probleme löst. Zum einen sorgt die Validierung dafür, dass die Niederspannung vor Überlastungen geschützt werden, zum anderen kann ein Flexibilitätsmarkt davon weitgehend unbehelligt dafür Sorge tragen, dass die flexiblen Verbraucher dazu eingesetzt werden können überschüssige Einspeisungen der höheren Spannungsebenen sinn voll zu nutzen und damit insgesamt einen erheblichen Beitrag zu Energiewende zu leisten.

Exemplarisch für die IEC 61850 wurde gezeigt , wie die Validierungsfunktion in einen weit verbreiteten Kommunikationsstandard implementiert werden kann. Während im Feldtest noch auf einem proprietären Webservice (REST, JSON) aufgesetzt wurde, kann davon ausgegangen werden, dass die Kommunikation ohne Weiteres auf einen Standard gehoben werden kann.

Der Implementierungsaufwand auf Seiten des Verteilnetzes beschränkt sich dabei im Wesentlichen auf die Installation repräsentativer Messungen sowie die Entwicklung einer Validierungslogik jeweils für jeden einzelnen Netzstrang. Konkret wären im Fall der MITNETZ Strom von den ca. 60.000 Ortsnetzten nur eine Teilmenge mit Sensorik auszustatten. Die so ermittelten Daten würden nach unserer Auffassung dazu ausreichen, auch für die ungemessenen Ortsnetzten eine für die Validierungslogik hinreichend genaue Zustandsschätzung und Prognose zu liefern. Die im Projekt entwickelte Schnittstelle setzt kann ohne Weiteres auf diese Größenordnung skaliert werden. Des Weiteren ist noch erforderlich, einen Kommunikationsstandard zu implementieren sowie die Kommunikation vom kundeneigenen Internetzugang auf die Infrastruktur der intelligenten Messsysteme zu übertragen. Insgesamt ist damit der Aufwand durchaus vergleichbar mit einer direkten Steuerung von Netzanschlüssen bzw. Verbrauchseinrichtungen durch Verteilnetzbetreiber. Die im Pilotprojekt gefundenen Ergebnisse zeigen jedoch deutlich die Vorteile der genutzten Validierungslogik:

- es besteht nur in wirklichen Grenzfällen die Notwendigkeit zum ad hoc-Eingriff durch den Verteilnetzbetreiber (z.B. bei IKT-Störungen oder unerwartet hohen Lastbedarfen)
- für die Beteiligten Marktakteure ist im Vorfeld ersichtlich, wenn eine Lastverlagerung erforderlich ist
- der Kunde weiß in kürzester Zeit welche Zeitfenster für seinen Lastbedarf verfügbar ist
- Im Sinne der von der Politik geforderten besseren Auslastung der Niederspannungsnetze [EU ...] sowie als kurzfristige Maßnahme, um die schnell wachsende Elektromobilität sowie die Zunahme von Wärmepumpen ohne Einschränkungen zu ermöglichen, wird die flächendeckende Einführung einer Validierungslogik empfohlen. Neben den oben genannten technischen Voraussetzungen, die durch Verteilnetzbetreiber umzusetzen wären, bedarf es noch politischer Weichenstellungen.

Seit diesem Jahr gilt ein novellierter §14a EnWG, der zum Ziel hat, flexible Verbraucher möglichst schnell und reibungslos in die Niederspannung zu integrieren. Der Fokus liegt dabei zu Recht auf einer

Anschlusspflicht durch den Verteilnetzbetreiber sowie einer Ausbauverpflichtung im Falle von Netzengpässen. Im Gegenzug erhalten Netzbetreiber die Möglichkeit im Falle nachgewiesener Engpässe eine kurative Lastabsenkung anzuweisen [BNetzA: Eckpunkte]. Die Herausforderungen für die Umsetzung dieser Regelung liegen jedoch auf der Hand: auf der einen Seite ist mit einer rasanten Verbreitung der Elektromobilität zu rechnen, die sich mit der raschen Installation von privaten Wallboxen auf die Netze auswirken wird und auf der anderen Seite stehen dem Verteilnetzbetreiber erhebliche Aufwände für die Verstärkung sowie die Ausrüstung mit Sensorik der Ortsnetze ins Haus. Dort werden die Umsetzungszeiträume von der Beauftragung von Tiefbauunternehmen, Bestimmungen der örtlichen Gebietskörperschaften sowie der Verfügbarkeit von Material und Fachkräften bestimmt. Es ist aktuell bereits absehbar, dass die Lastnachfrage deutlich schneller steigen wird als die Schaffung von örtlichen Netzkapazitäten.

In diesem Kontext bietet sich die Validierungslogik als Lösungsoption an. Würde zum Beispiel in einer Übergangsphase z.B. statt auf Sensorik in der Niederspannung State Estimation und Prognose eingesetzt und als Kommunikationsweg auf TLS-Verbindungen über das kundeneigene Internet zurückgegriffen, so könnte eine Validierungslogik sehr schnell implementiert werden und damit den rasanten Anstieg der Lastnachfrage abfangen. Besser noch: durch die Information über die konkreten Lastbedarfe wären Verteilnetzbetreiber kurzfristig in der Lage, Ortsnetze mit sehr starker Nachfrage zu identifizieren und somit als Grundlage für die Netzausbauplanung heranziehen. Eine solche Übergangslösung kann dann beim Rollout intelligenter Messsysteme auf die CLS-Kommunikation über Smart Meter Gateways ohne Weiteres umgestellt werden. Auch die nachfolgende Ausstattung der Ortsnetze ließe sich in einen solchen Umsetzungsfahrplan integrieren. Sobald ein Niederspannungsabgang nachgerüstet wird, sehen genauere Daten für die Ermittlung einer Restriktionskurve zur Verfügung und damit kann sofort mehr freie Kapazität für die Validierungslogik zur Verfügung gestellt werden.

Eine rechtliche Verankerung der Validierungslogik in der Festlegung der Bundesnetzagentur zum §14a EnWG würde somit die Integration der flexiblen Verbraucher in die Niederspannung erheblich beschleunigen und dabei insbesondere die Belange von Kunden und deren Stromlieferanten explizit berücksichtigen. Sie stellt auch die Basis für eine Standardisierung der Datenkommunikation zwischen Liegenschaft und Verteilnetzbetreiber dar. Hier ist mitunter die meiste Arbeit zu leisten. Sowohl Verteilnetzbetreiber müssen neben der Validierungslogik auch eine zugehörige Kommunikationsnorm zur Verfügung stellen. MITNETZ Strom hat dazu die entsprechenden Anwendungsfälle in der DKE AR-2829-6 vorangetrieben. Damit ist prinzipiell die Umsetzung auf der Herstellerseite (Energie Management Systeme, einzelne Geräte wie smarte Wallboxen oder Wärmepumpen) möglich.

Aus Sicht der HAW Hamburg und der EnergieDock GmbH war die Umsetzung des FlexHubs und insbesondere die Entwicklung des Flexibilitätsplattform NEMO.spot, die Register, Marktplatz und Aggregationssoftware vereint ein voller Erfolg. Am Beispiel des Ladens von E-Autos konnte gezeigt werden, dass Flexibilität von Endverbrauchern, sinnvoll eingesetzt werden kann, um Netzengpässe proaktiv zu vermeiden und das Abregeln von Erneuerbaren Energieanlagen zu verhindern. Die erarbeitete Lösung ist ebenfalls einsetzbar für andere Assets wie z.B. Wärmepumpen. Dabei sind sowohl der Markt als auch die Regulatorik bereit für eine solche Lösung:

2021 hat die scheidende Bundesregierung eine Novelle des Energiewirtschaftgesetztes (EnWG) beschlossen, die transparente, diskriminierungsfreie Flexibilitätsmärkte als zwingenden Bestandteil der zukünftigen Beschaffungsmodelle von Netzbetreibern festlegt (§14c EnWG). §14a EnWG besagte schon länger das steuerbare Verbrauchseinrichtungen, z.B. Elektroautos, gegen ein reduziertes Netzentgelt von Verteilnetzbetreibern gesteuert werden können. Zunächst wurde dort allerdings eine restriktive Abregelung in Form einer Abschaltung der Ladung über 2 Stunden bevorzugt. Dieses wurde allerdings später vom Bundeswirtschaftsministerium zurückgezogen und stattdessen §14c im Sinne einer marktdienlichen Lösung weiter ausgestaltet. Die dort gemachten Vorgaben werden aktuell bereits von NEMO.spot umgesetzt.

Aufgrund der stark gestiegenen Energiepreise und er zunehmenden Volatilität der Preise bedingt durch den gegenwärtigen Krieg in der Ukraine, bietet Flexibilität und der Handel bzw. die Optimierung auf einer Plattform wie NEMO.spot auch für Energieversorger ein erhebliches Einsparungspotential (von ca. 25-33%)[6].

Eine zukünftige Verzahnung dieser Maßnahmen und die Etablierung von Flexibilitätsmärkten wie NEMO.spot bietet hier einen großen volkswirtschaftlichen Nutzen, der:

- Energiebeschaffungskosten erheblich senken,
- eine bessere Auslastung/Ausnutzung der Erzeugung von Erneuerbaren Energien sicherstellen,
- Netzengpässe proaktiv verhindern und bei auftretenden Netzengpässen, gerade in der Niederspannung, diese durch Verschiebung der flexiblen Lasten verhindern kann,
- zusätzlich durch eine bessere Ausnutzung der Erneuerbaren Energien und der Unterstützung der Mobilitätswende, hin zu mehr E-Autos, und der Wärmewende, Einbindung der wachsenden Zahl an Wärmepumpen, CO2 Emissionen senken kann.

Eine Methodik zum Assessment funktionaler Risiken, wie sie typischerweise durch seltene Ereignisse in der Systemumgebung also z.B. durch unerwartet koordiniertes Nutzerverhalten, Wetter- oder Marktgeschehen ausgelöst werden können, wurde erfolgreich entwickelt [19]. Ihre Anwendung auf die hier untersuchten Use-Cases erfordert die zukünftige Entwicklung spezialisierter simulationsbasierter Werkzeuge zur Erfassung von Systemanforderungen unter Berücksichtigung funktionaler Risiken zur Ermöglichung eines robusten Betriebs flexibler Ressourcen im Rahmen von Flexibilitätsmärkten.

Zusammenfassung FlexChain Entwicklung und Testplattform Ergebnisse

Das im FlexHub-Projekt entwickelte FlexChain bietet einen blockchainbasierten Ansatz zum Flexibilitätsmarkt Konzept und zu den entworfenen Anwendungsfällen. Die folgenden Punkte fassen die Vorteile des FlexChain Prototyps zusammen.

1. Die auf Ethereum-basierte Quorum-Blockchain dient als Infrastruktur für die Flexibilitätsmarkt und deren Akteuren. Quorum ersetzt die zentrale Speicherstelle durch ein Netzwerk von Knoten, wo Daten redundant gespeichert und die zentralisierte Logik zu Smart Contracts umgewandelt wird.

[6] Diese Daten basieren auf der Auswertung eines Pilotprojekts, dass die EnergieDock GmbH im Frühjahr 2022 u.a. zusammen mit Green Planet Energy (Energieversorger) im Rahmen des FlexHafen Projektes vorgenommen hat. Die Publikation des entsprechenden Projektberichtes steht aktuell noch aus.

2. Die Notar-Funktion von Quorum ermöglicht das Speichern der Eingabedaten und Ergebnisse der Matching-Logik in das Ledger. Die Daten lassen sich somit zurückverfolgen und sind unveränderbar.
3. Die Smart Contracts sind für allen Partner bzw. Validator-Knoten einsehbar und die von denen produzierten Ergebnisse sind nachvollziehbar.
4. Die dezentrale Natur von Quorum bietet keine Single-Point-of-Failure. Zudem können Daten nicht verloren gehen und die Sicherheit des Netzwerks ist nicht durch einen Angriff an einen Knoten kompromittiert.
5. Die Konsensalgorithmen sorgen dafür, dass das Netzwerk bei Ausfällen oder bösartigen Knoten erhalten bleibt. Dafür gibt es großzügigen Margen, wobei Schaden in Netzwerk ohne Auswirkungen bleiben. Solange über 50% der Validator-Knoten nicht ausfallen, im Falle von Raft, oder 67% der Knoten nicht bösartig sind, im Falle vom Istanbul BFT, dann verkraftet das Netzwerk problemlos die Schäden.
6. Neue Partner als Validator-Knoten aufzunehmen oder Partner vom Netzwerk zu entfernen ist im Netzwerk-Konsortium frei einstellbar. Governance-Modellen, in denen über diese Entscheidungen gestimmt wird, sind in konsortialen Blockchain verbreitet [82],[83].
7. Wallets wie MetaMask ermöglichen den Benutzern einen reibungslosen Zugang zu den Smart Contracts anhand von kryptographischen Schlüsselpaaren.

Als festgestellte Einschränkung ist die erhöhte Berechnungs-Komplexität in Smart Contracts zu benennen. Die implementierte Matching-Logik nimmt beim Ausführen viel Zeit in Anspruch und lässt sich durch die relativ weniger flexibel Solidity Programmiersprache nur bedingt optimieren. Eine Verbesserung könnte durch die Auslagerung der Matching-Logik außerhalb der Blockchain gelangen und sie in eine Programmiersprache wie Python oder Java implementieren. Der Nachteil hierbei ist schlicht der Verlust der Transparenz bei der Programmcode.

Zudem gibt es einige Erweiterungsmöglichkeiten zur aktuellen Version des FlexChains:

1. Aktualisierung der Smart Contracts: Sobald den Clients die Adresse der in Quorum deployten Smart Contracts mitgeteilt wird, lässt sich diese Adresse nicht ohne weiteren Aufwand ändern. Neue Versionen von den Smart Contracts sind allerdings unerlässlich, etwa um neue Features einzuführen oder Bugs zu beheben. Deswegen können Proxy Smart Contracts die Rolle der unveränderbare Smart Contracts übernehmen. Diese speichern die Adresse der neuesten Version der „inhaltlichen“ Smart Contracts, die sich im Laufe der Zeit verändern. Wichtig dabei ist, dass das Proxy-Muster von Anfang an verwendet wird. Damit lassen sich Updates bei Smart Contracts problemlos einspielen.
2. Währungen einführen: Anfragen unbegrenzt an FlexChain senden kann zu langen Latenzzeiten bei der Verarbeitung der Transaktion führen. Deswegen ist eine Einschränkung der Anfragen erforderlich, welche durch die Einführung einer Kryptowährung (FlexCoin) möglich wäre. Für jede Transaktion muss das Gas in FlexCoin bezahlt und jeder Validator-Knoten bekommt die Transaktionsgebühren für seine ausgeführten Transaktionen. FlexCoin hat nicht zwingend einen realen finanziellen Wert, vielmehr begrenzt sie die Anzahl und Komplexität der ausgeführten Transaktionen pro Client.

3. Identitäten und Zertifikate einführen: Nicht nur die Quorum Clients (etwa die FlexChain APIs) können von den Vorteilen der kryptographischen Schlüsselpaar profitieren, sondern alle Assets in den Flexibilitätsmarkt. DER-Objekte, Flex-Angebote, VNB-Objekte, VNB-Anfragen können sich als Schöpfung einer der Clients nachweisen, indem die Clients deren erstellen Objekte mit kryptographischen Signaturen versehen. Eine Identität-Bündel (auch als DID bekannt) kann durch eine seiner Aktionen Verifiable Credentials erstellen, eine Art Zertifikat, die Signaturen und das Autor-DID beinhalten. Somit lassen sich die erstellen Objekten leicht nachweisen, sind auf ihren Autor zurückverfolgen und sind fälschungssicherer.

Zu IT-Sicherheit: Das IT-Sicherheitskonzept ergibt sich aus den Handlungsempfehlungen welche an die Projektpartner während der Projektlaufzeit herausgegeben wurden. Eine finale Bewertung ist aufgrund der mehrfachen Verlängerungen der Projektlaufzeit nicht möglich. Fraunhofer FKIE nimmt während des Feldversuches nur noch beratend am Projekt teil. Als Handlungsempfehlung ergibt sich, vor einem Produktiveinsatz des Systems, ein ausführliches Sicherheitsaudit der Komponenten durchzuführen. Als Basis für dieses Audit können direkt die Handlungsempfehlungen verwendet werden, welche durch das Fraunhofer FKIE abgeleitet wurden.

II.2 Erfolgte oder geplante Veröffentlichungen im Rahmen der Projektlaufzeit

Das Flexhub Vorhaben wurde in folgenden Artikeln etc. veröffentlicht. Weitere Veröffentlichungen sind derzeit in Arbeit.

- **Schmidtke, Florian; Hacker, Immanuel; Vertgewall, Chris Martin; Ulbig, Andreas: Evaluation of multi-use charging strategies in a time-dependent co-simulation environment for behind-the-meter flexibility; 2022; CIRED Workshop**

- **Schmidtke, Florian; Fatemi, Armin; Offergeld, Thomas; Immanuel, Hacker; Georgiev, Borislav; Ulbig, Andreas: Evaluating multi-use applications in a co-simulation environment; 2022;MEDPOWER**

- **FlexHub Pilottest: "Kiwigrid, EnergieDock und Mitnetz Strom demonstrieren netzdienliche Steuerung von Elektroautos" - veröffentlicht, s. https://kiwigrid.com/de/artikel/kiwigrid-energiedock-und-mitnetz-strom-demonstrieren-netzdienliche-steuerung-von-elektroautos**

- **FlexHub Feldtest: "Kiwigrid ermöglicht günstiges und netzdienliches Laden von Elektroautos" – veröffentlicht, s. https://kiwigrid.com/de/artikel/kiwigrid-ermoeglicht-guenstiges-und-netzdienliches-laden-von-elektroautos**

- **W. Renz, J. Sudeikat, J. Backes und K. Eger, „Assessment of functional risks for engineering adaptive smart grid applications," in *IEEE 10th International Conference on Smart Energy Grid Engineering*, 2022.**

- **T. Dethlefs, A. Schröder und C. Kahlen, „Marktdesign und Datenmodelle im Projekt Flexhub," ew - Magazin für die Energiewirtschaft, Nr. 6, August 2020.**

- **Eugen Winter, Michael Rademacher: Fuzzing of SCADA Protocols used in Smart Grids Poster: DACH+ Energy Informatics Conference 2020 [65]**
  A successful deployment and operation of smart grids depends on the reliability and security of the protocols used to gather data from the various components. This work evaluates a technique called fuzzing to investigate the security of smart grid communication protocols. Based on a structured process for fuzzing in this specific domain we develop a fuzzer that has been made publicly available to ensure repeatability of the results and ease further security assessments of protocols and implementations. By applying this process to a well-known implementation of the IEC 61850 protocol, several bugs have been found and reported to the developers.

- **Immanuel Hacker , Florian Schmidtke , Dennis van der Velde, Toni Czyrnik: A framework to evaluate multi-use flexibility concepts simultaneously in a co-simulation environment and a cyber-physical laboratory.; Konferenzbeitrag Cired Main Conference 2021**
  In this paper, we present an approach for a framework to test multi-use concepts for the coordination of flexibility resources in distribution grids. The framework allows testing in a large scale co-simulation environment, but also our cyber-physical smart grid laboratory. At the core of the proposed framework is the newly proposed design of a Virtual Edge Device (VED) to control either real or simulated flexibility resources. The VED is also provided for the interchangeable use of simulated and physical components with the same coordination logic. To tackle the questions of cyber resilience and cybersecurity, not only the energy system is taken into account but also the underlying Information and Communication Technology (ICT). Therefore, the ICT components are emulated in the co-simulation and mapped by corresponding components in the laboratory. We validate the environment by simulating and analysing three different use-cases, comprising local congestion management and battery dispatch optimisation. Our tests prove that our laboratory is suitable for a future application of the analysis and development of multi-use flexibility concepts.

- **Immanuel Hacker; Omer Sen; Dennis van der Velde; Florian Schmidtke; Andreas Ulbig: Towards more realistic co-simulation of cyber-physical energy distribution systems; Konferenzbeitrag NecSys22**
  The increased integration of information and communications technology at the distribution grid level offers broader opportunities for active operational management concepts. At the same time, requirements for resilience against internal and external threats to the power supply, such as outages or cyberattacks, are increasing. The emerging threat landscape needs to be investigated to ensure the security of supply of future distribution grids. This extended abstract presents a co-simulation environment to study communication infrastructures for the resilient operation of distribution grids. For this purpose, a communication network emulation and a power grid simulation are combined in a common modular environment. This will

provide the basis for cybersecurity investigations and testing of new active operation management concepts for smart grids. Exemplary laboratory tests and attack replications will be used to demonstrate the diverse use cases of our co-simulation approach.

- **Immanuel Hacker, Johannes Lenzen, Florian Schmidtke, Dennis Van der Velde, Andreas Ulbig: A Co-Simulation Environment to Evaluate Cyber Resilience in Active Distribution Grids Utilising Behind-the-Meter Assets; Konferenzbeitrag MedPower22**
  The increasing utilisation of behind-the-meter assets in an active distribution grid leads to an ever more complex cyber-physical system, producing a greater dependency on active communication and increasing vulnerability to cyber-attacks. Consequently, there is a need to simulate relevant cyber-physical systems and incidents and their impact to assess the overall system`s cyber resilience. This paper aims to present a simulation environment that can be used to analyse the cyber resilience of different system designs. This simulation environment incorporates all relevant domains, including the energy grid, the information and communication infrastructure and the used operational technology. Furthermore, we propose a new way to abstract the communication infrastructure inside the co-simulation, including the simulation of incidents like cyber-attacks and communication failures. The basis of this work is a requirements analysis that considers the events to be simulated in the form of threat modelling based on the STRIDE framework. The functioning of the entire environment is demonstrated using an example concept for voltage control in the low-voltage grid, for which a cyber-attack and its effects are simulated, and finally, a more resilient alternative is presented.

- **Immanuel Hacker, Florian Schmidtke, Dennis van der Velde, Steve Bahn, Andreas Ulbig :: Full Stack Development Process for Demand Side Flexibility Solutions from Cyber-Physical Testbeds to Field Operation; Konferenzbeitrag ETG Kongress 2023**
  Autoren: Immanuel Hacker, Florian Schmidtke, Dennis van der Velde, Steve Bahn, Andreas Ulbig
  The ongoing energy transition necessitates a comprehensive transformation of the electricity supply system. This presents grid operators and utilities with the challenge of maintaining stable and reliable grid operation in the presence of a large number of variable renewable energy sources. Demand-side management solutions, especially for behind-the-meter assets, will play a crucial role in future grid concepts by providing demand-side response to support system stability. These solutions often rely on centralized or decentralized platforms that operate outside of the supervisory control and data acquisition systems currently used by distribution system operators. To effectively utilize the operational flexibility offered by demand-side management solutions in real-world settings, several technical domains must be addressed, including power systems, information and communication technology, and operational processes. This paper outlines the steps taken to develop and analyze demand-side management solutions, including the simulation of operational processes, testing in a smart grid laboratory, and field tests with real-world customers.

III. Literaturverzeichnis

[1] Dethlefs, Tim; Schröder Andrea; Kahlen Christoph; Schmidtke, Florian; Hacker, Immanuel; Conrad, Michael: Betriebliche und technishce Herausforderungen der Einbindung von Endkundenflexibiliät

[2] Hacker, Helge, Immanuel; Schmidtke Florian; van der Velde, Dennis; Czyrnik, Sascha Toni: A framework to evaluate multi-use flexibility concepts in ca co-simulation envrionment and a cyber-physical laboratory

[3] T. Dethlefs und W. Renz, „A distributed registry for service-based energy management," in *Proc. 39th Annual Conference of the IEEE Industrial Electronics Society*, 2013.

[4] T. Dethlefs, T. Preisler, O. Renke, W. Renz, A. Lang, A. Pawils, A. Morillon, A.-K. Peschka und R. Wagner, „D4.1 The OS4ES Security and Privacy Concept and the Distributed DER," 2015.

[5] T. Dethlefs, C. Brunner, T. Preisler, O. Renke, W. Renz und A. Schröder, „Energy Service Description for Capabilities of Distributed Energy Resources," *Lecture Notes in Computer Science,* Nr. 9424, pp. 24-35, 2015.

[6] T. Dethlefs, T. Preisler und W. Renz, „Dynamic Aggregation of Virtual Power Plants with a Registry System for Distributed Energy Resources," *Smart Energy Research. At the Crossroads of Engineering, Economics, and Computer Science,* 2017.

[7] T. Dethlefs, A. Schröder, C. van den Broek, M. Fernandez, A. Papanikolaou und W. Renz, „Demonstration of an OS4ES based Virtual Power Plant Aggregation Process," in *VDE - ETG Kongress*, Mannheim, 2017.

[8] I. Bilgin, „Principles of Container-based Application Design," Red Hat Consulting, 2019.

[9] B. Burns und D. Oppenheimer, „Design patterns for container-based distributed systems," in *Proceedings of the 8th USENIX Conference on Hot Topics in Cloud Computing*, 2016.

[10] A. Bucchiarone, N. Dragoni, S. Dustdar, P. Lago, M. Mazzara und V. Rivera, Microservices - Science and Engineering, Springer Book, 2020.

[11] D. North, „Introducing Behaviour Driven Development," [Online]. Available: https://dannorth.net/introducing-bdd/.

[12] „Cucumber - Gherkin," [Online]. Available: https://github.com/cucumber/common.

[13] CEN-CENELEC-ETSI, „Smart grid reference architecture," Smart Grid Coordination Group, 2012.

[14] M. Gottschalk, M. Uslar und C. Delfs, The Use Case and Smart Grid Architecture Model Approach, Springer International Publishing, 2017.

[15] C. Neureiter, D. Engel und M. Uslar, „Domain specific and model based systems engineering in the smart grid as prerequesite for security by design," *Electronics,* Bd. 5, Nr. 2, 2016.

[16] W. Renz und A. Lang, „OS4ES Deliverable 4.1: The OS4ES Security and Privacy Concept and the Distributed DER Registry System Architecture," 2015.

[17] T. Dethlefs, A. Schröder und C. Kahlen, „Marktdesign und Datenmodelle im Projekt Flexhub," *ew - Magazin für die Energiewirtschaft,* Nr. 6, August 2020.

[18] ISO/IEC250010, „Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models," 2011.

[19] W. Renz, J. Sudeikat, J. Backes und K. Eger, „Assessment of functional risks for engineering adaptive smart grid applications," in *IEEE 10th International Conference on Smart Energy Grid Engineering*, 2022.

[20] B. Ditzel, J. Dahlkemper, K. Landenfeld und W. Renz, „Integratives Grundstudium in den Ingenieurwissenschaften durch Themenwochen - vom Konzept zur Umsetzung," *Zeitschrift für Hochschulentwicklung,* Bd. 9, Nr. 4, pp. 191-, 2014.

[21] „Common Terms | MetaMask Docs," [Online]. Available: https://docs.metamask.io/guide/common-terms.html. [Zugriff am 28 9 2021].

[22] „Ethereum accounts | ethereum.org," [Online]. Available: https://ethereum.org/en/developers/docs/accounts/. [Zugriff am 28 9 2021].

[23] „MetaMask - A crypto wallet & gateway to blockchain apps,“ [Online]. Available: https://metamask.io/. [Zugriff am 29 9 2021].

[24] „web3.js - Ethereum JavaScript API — web3.js 1.0.0 documentation,“ [Online]. Available: https://web3js.readthedocs.io/en/v1.5.2/. [Zugriff am 26 10 2021].

[25] „Infura Documentation | Infura Documentation,“ [Online]. Available: https://infura.io/docs. [Zugriff am 28 7 2021].

[26] W. P. Andrei Ionita, „FlexHub Blockchain Spezifikation,“ 2021.

[27] „Basic permissions - GoQuorum,“ [Online]. Available: https://docs.goquorum.consensys.net/en/stable/HowTo/Configure/Permissioning/BasicPermissions/. [Zugriff am 28 9 2021].

[28] Bundesamt für Sicherheit in der Informationstechnik, „BSI-Standard 100-2 IT-Grundschutz-Vorgehensweise,“ 2008. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/ITGrundschutzstandards/BSI-Standard_1002.pdf. [Zugriff am 03. 05. 2022].

[29] Bundesamt für Sicherheit in der Informationstechnik, „Kryptographische Verfahren: Empfehlungen und Schlüssellängen , https://www.,“ 2021. [Online]. Available: https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Standards-und-Zertifizierung/Technische-Richtlinien/TR-nach-Thema-sortiert/tr02102/tr02102_node.html. [Zugriff am 03. 05. 2022].

[30] Q. Dang, „NIST Special Publication 800-107 Revision 1, Recommendation for Applications Using Approved Hash Algorithms,“ 2012. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-107r1.pdf. [Zugriff am 03. 05. 2022].

[31] A. J. M. P. C. V. O. S. A. V. J. Katz, Handbook of applied cryptography., CRC press, 1996.

[32] M. B. R. C. H. Krawczyk, „ Hmac: Keyed-hashing for message authentication, Internet Requests for Comments, RFC Editor, RFC 2104,,“ 02. 1997. [Online]. Available: https://www.rfc-editor.org/rfc/rfc2104.txt. [Zugriff am 03. 05. 2022].

[33] One Identity LLC, „"syslog-ng Open Source Edition 3.16 - Administration Guide",“ 2019. [Online]. Available: https://www.syslog-ng.com/technical-documents/doc/syslog-ng-open-source-edition/3.16/administration-guide/56. [Zugriff am 03. 05. 2022].

[34] Elasticsearch B.V, „Beats,“ [Online]. Available: https://www.elastic.co/de/beats/. [Zugriff am 03. 05. 2022].

[35] Elasticsearch B.V., „The Elastic Stack,“ [Online]. Available: https://www.elastic.co/de/products/elastic-stack. [Zugriff am 03. 05. 2022].

[36] Apache Software Foundation, „Apache Kafka: A distributed streaming platform,“ [Online]. Available: https://kafka.apache.org/. [Zugriff am 03. 05. 2022].

[37] HiveMQ, „MQTT Essentials,“ 2019. [Online]. Available: https://www.hivemq.com/mqtt-essentials/. [Zugriff am 03. 05. 2022].

[38] Eclipse Foundation, „Eclipse Mosquitto,“ 2019. [Online]. Available: https://mosquitto.org//. [Zugriff am 03. 05. 2022].

[39] HiveMQ, „HiveMQ MQTT Broker,“ 2019. [Online]. Available: https://www.hivemq.com/hivemq/. [Zugriff am 03. 05. 2022].

[40] Bundesamt für Sicherheit in der Informationstechnik, „Mindeststandard des BSI zur Verwendung von Transport Layer Security,“ 2021. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Mindeststandards/Mindeststandard_BSI_TLS_Version_2_2.pdf?__blob=publicationFile&v=5. [Zugriff am 04. 05. 2022].

[41] Bundesamt für Sicherheit in der Informationstechnik, „Technische Richtlinie BSI TR-03116 Kryptographische Vorgaben für Projekte,“ 24. 01. 2022. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03116/BSI-TR-03116-4.pdf?__blob=publicationFile&v=2. [Zugriff am 04. 05. 2022].

[42] SSL.com - Conor Wilson, „What is a chain of trust?,“ 2020. [Online]. Available: https://www.ssl.com/faqs/what-is-a-certificate-authority/. [Zugriff am 04. 05. 2022].

[43] Qualys, Inc., „SSL and TLS Deployment Best Practices," 2020. [Online]. Available: https://github.com/ssllabs/. [Zugriff am 04. 05. 2022].

[44] Microsoft Documentation, „ Overview of DNSSEC," 31. 08. 2016. [Online]. Available: https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/jj200221(v=ws.11). [Zugriff am 04. 05. 2022].

[45] Heise Magazine - Carsten Strotmann und Jürgen Schmidt, „Private Auskunft - DNS mit Privacy und Security vor dem Durchbruch," 2018. [Online]. Available: https://www.heise.de/select/ct/2018/14/1530492966691096. [Zugriff am 04. 05. 2022].

[46] APNIC Blog - Geoff Huston, „DNSSEC 'and' DNS over TLS," 20. 08. 2018. [Online]. Available: https://blog.apnic.net/2018/08/20/dnssec-and-dns-over-tls/. [Zugriff am 04. 05. 2022].

[47] Security Insider - Dipl.-Ing. (FH) Stefan Luber, „Definition DNS-based Authentication of Named Entities (DANE)," 02. 09. 2019. [Online]. Available: https://www.security-insider.de/was-ist-dane-a-860135/. [Zugriff am 04. 05. 2022].

[48] „GitHub, Quorum," [Online]. Available: https://github.com/ConsenSys/quorum. [Zugriff am 14. 04. 2022].

[49] „Ethereum Developer Documentation," [Online]. Available: https://ethereum.org/en/developers/docs/. [Zugriff am 14. 04. 2022].

[50] D. Ongaro und J. Ousterhout, „In Search of an Understandable Consensus Algorithm," p. 16.

[51] H. Moniz, „The Istanbul BFT Consensus Algorithm," 05. 2020. [Online]. Available: http://arxiv.org/abs/2002.03613. [Zugriff am 14. 04. 2022].

[52] Bundesnetzagentur, „Monitoringbericht 2019," 2019, p. 543.

[53] Bundesnetzagentur Marktstammdatenregister, „Öffentliche Einheiten Stromerzeugung," [Online]. Available: https://www.marktstammdatenregister.de/MaStR/Einheit/Einheiten/OeffentlicheEinheitenuebersicht#stromerzeugung. [Zugriff am 14. 04. 2022].

[54] T. Schirrmacher, „Bewertung von Konsensus-Algorithmen für die Verwendung in Blockchain-Lösungen für intelligente Energieverteilnetze,“ 2020, p. 78.

[55] Bundesamt für Sicherheit in der Informationstechnik, „Hochverfügbarkeitskompendium,“ [Online]. Available: https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Informationen-und-Empfehlungen/Empfehlungen-nach-Angriffszielen/Hochverfuegbarkeit/HVKompendium/hvkompendium_node.html. [Zugriff am 17. 05. 2022].

[56] Y. Liu, „High availability of network service on docker container,“ in *5th International Conference on Measurement, Instrumentation and Automation*, Atlantis Press, 2016.

[57] CRIU, „Docker,“ [Online]. Available: https://criu.org/Docker. [Zugriff am 17. 05. 2022].

[58] D. Z. a. Y. Tamir, „Fault-tolerant containers using nilicon,“ in *2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS).*, IEEE, 2020, pp. 1082 - 1091.

[59] Docker Inc., „Swarm mode overview,“ [Online]. Available: https://docs.docker.com/engine/swarm/. [Zugriff am 17. 05. 2022].

[60] Kubernetes, „Was ist Kubernetes?,“ [Online]. Available: https://kubernetes.io/de/docs/concepts/overview/what-is-kubernetes/. [Zugriff am 17. 05. 2022].

[61] M. Zillgith, „Open source libraries for IEC 61850 and IEC 60870-5-104,“ 2020. [Online]. Available: https://libiec61850.com/. [Zugriff am 18. 05. 2022].

[62] H. Z. B. Q. a. Z. C. T. Tu, „A vulnerability mining system based on fuzzing for IEC 61850 protocol,“ in *2017 5th International Conference on Frontiers of Manufacturing Science and Measuring Technology (FMSMT 2017)*, Atlantis Press, 2017.

[63] OpenRCE, „Sulley,“ 2020. [Online]. Available: https://github.com/OpenRCE/sulley. [Zugriff am 18. 05. 2022].

[64] J. Pereyda, „Boofuzz,“ 2020. [Online]. Available: https://github.com/jtpereyda/boofuzz. [Zugriff am 18. 05. 2022].

[65] M. R. Eugen Winter, „Fuzzing of SCADA Protocols used in Smart Grids,“ in *Abstracts from the 9th DACH conference on energy informatics*, Springer Nature, 2020, pp. 1 - 3.

[66] M. A. A. a. I. A. S. S. Hussain, „IEC 61850 modeling of dstatcom and xmpp communication for reactive power management in microgrids,“ in *IEEE Systems Journal, vol. 12, no. 4*, IEEE, 2018, pp. 3215-3225.

[67] S. S. H. I. A. a. T. S. U. M. A. Aftab, „IEC 61850 and xmpp communication based energy management in microgrids considering electric vehicles,“ in *IEEE Access, vol. 6*, IEEE, 2018, pp. 35657 - 35668.

[68] M. A. A. S. H. I. A. P. K. T. A. K. G. a. T. S. U. F. Nadeem, „Virtual power plant management in smart grids with xmpp based iec 61850 communication,“ in *Energies, vol. 12, no. 12*, 2019, p. 2398.

[69] M. Rivas, „Fuzzowski,“ 2020. [Online]. Available: https://github.com/nccgroup/fuzzowski. [Zugriff am 18. 05. 2022].

[70] Mitre CVE-List, „CVE-2018-20685,“ 2018. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-20685. [Zugriff am 18. 05. 2022].

[71] Mitre CVE-List, „CVE-2019-16905,“ 2019. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-16905. [Zugriff am 18. 05. 2022].

[72] Mitre CVE-List, „CVE-2019-6109,“ 2019. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-6109. [Zugriff am 18. 05. 2022].

[73] Mitre CVE-List, „CVE-2019-6110,“ 2019. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-6110. [Zugriff am 18. 05. 2022].

[74] Mitre CVE-List, „CVE-2019-6111,“ 2019. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-6111. [Zugriff am 18. 05. 2022].

[75] Mitre CVE-List, „CVE-2020-14145,“ 2020. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-14145. [Zugriff am 18. 05. 2022].

[76] Mitre CVE-List, „CVE-2020-15778," 2020. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-15778. [Zugriff am 18. 05. 2022].

[77] Mitre CVE-List, „CVE-2021-28041," 2021. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-28041. [Zugriff am 18. 05. 2022].

[78] Cisco-Talos, „Mutiny-Fuzzer," 2020. [Online]. Available: https://github.com/Cisco-Talos/mutiny-fuzzer. [Zugriff am 18. 05. 2022].

[79] „Fraunhofer GitLab - flexchain sandbox Projekt," [Online]. Available: https://gitlab.fit.fraunhofer.de/coop/project/flexhub/flexchain-sandbox/-/wikis/home. [Zugriff am 27 6 2022].

[80] „ISO/IEC 25010," [Online]. Available: https://iso25000.com/index.php/en/iso-25000-standards/iso-25010. [Zugriff am 27 6 2022].

[81] A. Mecklenborg, „Bewertung eines Blockchain-basierten Plattformkonzeptes für die netzdienliche Koordination und Steuerung dezentraler Flexibilitäten," RWTH, Aachen, 2021.

[82] „bloxbeg - The Trusted Research Infrastructure," 2 2020. [Online]. Available: https://bloxberg.org/wp-content/uploads/2020/02/bloxberg_whitepaper_1.1.pdf. [Zugriff am 27 6 2022].

[83] R. C. M.-B. a. J. L. K. Beck, „Governance in the blockchain economy: A framework and research agenda," in *Journal of the Association for Information Systems 19.10 (2018): 1*, 2018.

[84] G. Liu, *Analysis of Switching Transients during Energization in Large Offshore Wind Farms,* 2018.

[85] CIGRE, *Network modelling for harmonic studies,* 2019.

[86] T. Ohno, *Derivation of Theoretical Formulas of Sequence Currents on Underground Cable System,* 2011.

[87] K. Leuven, *Resonance and Transient Behaviour of Extensive Cable Grids,* 2020.

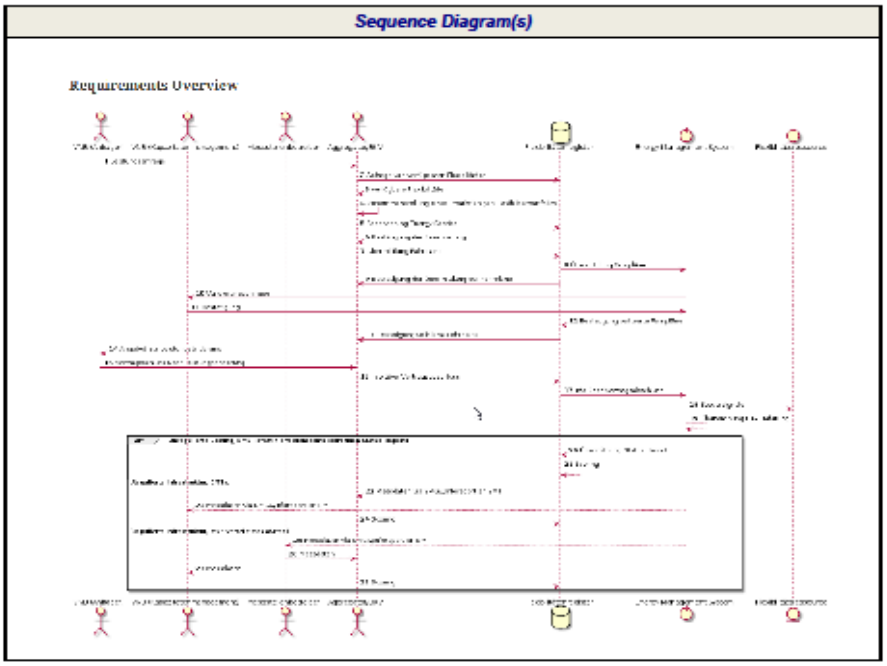
[88] Das, *Power System Analysis: Short Circuit, Load Flow and Harmonics,* 2002.

[89] G. Ye, *Downstream Network Modeling With GeneralizedDistribution Networks for Harmonic Studies,* 2020.

IV. Anlagen

A1 Anwendungsfall 1: Netzengpassmanagement

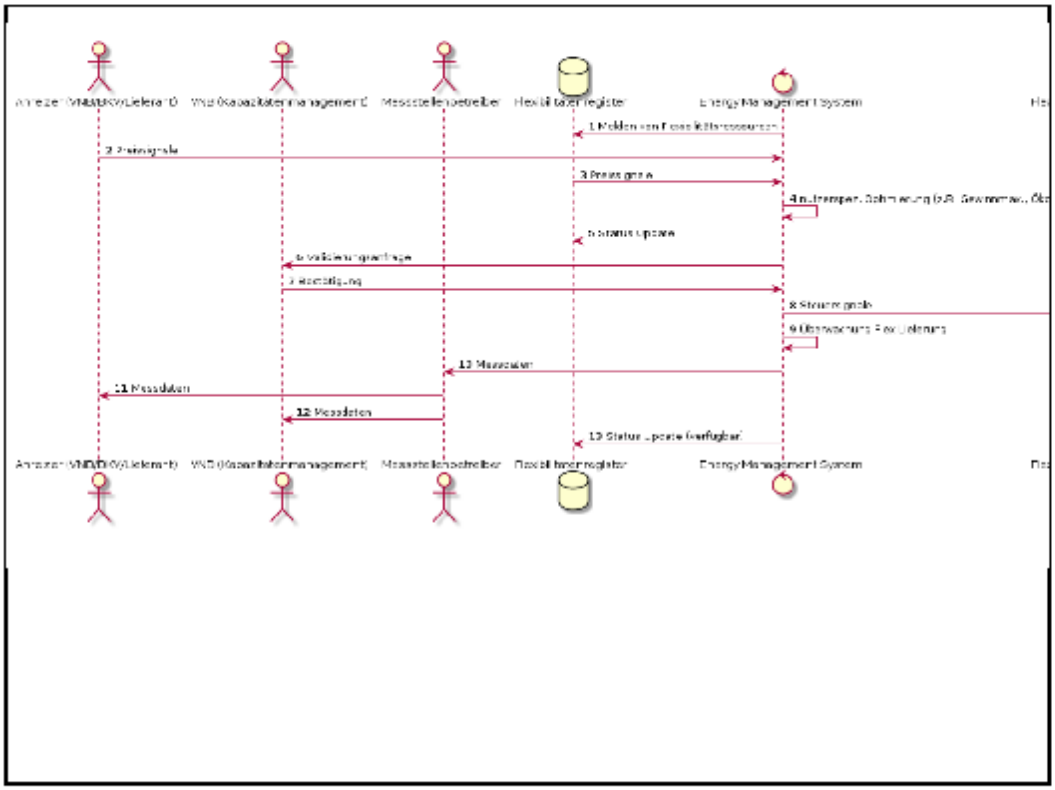
Diagramm AF1 "Geradeausweg"



Fehlerfall Zurückweisung Validierung

Der Fehlerfall beschreibt den Zustand, bei dem der VNB Kapa nach der Validierung der Fahrpläne deren Ausführung durch die Flexibilitätsressourcen aufgrund von Kapazitätsproblemen, die bei der Umsetzung der Fahrpläne im Netz auftreten würden, verweigert.

A2 Anwendungsfall 2: Netzengpassmanagement ohne Markt



Fehlerfall Validierung

A3 Anwendungsfall 3: Flex on Demand

4 Step by Step Analysis of Use Case

4.1 Overview of Scenarios

Scenario Conditions						
No.	**Scenario Name**	**Scenario description**	**Primary Actor**	**Triggering Event**	**Pre-Condition**	**Post-Condition**
1	Identifikation von Flexibilitäten	Ein Aggregator erstellt Angebote für die Vermarktung von Flexibilitäten von DER Systemen im Flexibilitäten-register.	Aggregator	-	Der Aggregator verfügt über DER-Systeme mit vermarktbaren Flexibilitäten	Angebote von zu vermarktenden Flexibilitäten sind im Flexibilitäten-register vorhanden
2	Suchen und Buchen von Flexibilitäten	Ein Flex-Anfrager sucht im Register nach buchbaren Flexibilitäten, führt eine lokale Einsatzplanung und Optimierung durch und bucht dann ausgewählte Flexibilitäten im Flexibilitätenregister. Das Register leitet die Buchungsanfrage an den Aggregator weiter, der die gebuchte Ressource verwaltet.	Flex Anfrager	Suche nach Flexibilitäten durch den Flex Anfrager	Angebote von zu vermarktenden Flexibilitäten sind im Flex-Register vorhanden	Die Buchungsanfrage ist bei dem Aggregator, der die gebuchte Flexibilität verwaltet, eingegangen
3	Validierung einer Buchung (grüne Phase)	Der Aggregator informiert das EMS über die Buchung, das EMS schickt daraufhin eine Validierungsanfrage an den VNB Kapa. In der grünen Phase wird die Buchung validiert.	Aggregator	Buchungsanfrage erhalten	Der Aggregator hat eine Buchungsanfrage für eine oder mehrere verwaltete Flexibilitäten erhalten	Die Buchungsanfrage wurde von dem VNB Kapa validiert und der Aggreator und das Register wurden über die erfolgreiche Validierung informiert.
4	Steuerung von Flexibilitäten	Der Flex Anfrager schickt Fahrpläne für von ihm gebuchten Flexibilitäten an das Flex-Register. Dieses validiert und authentifiziert die Fahrpläne und leitet sie an den Aggregator weiter, der die Anlagen steuert.	Flex Anfrager	Senden von Fahrplänen durch den Flex Anfrager	Mindestens eine Flexibilität wurde erfolgreich vom Flex Anfrager gebucht	Die Flexibilität wurde gemäß Fahrplan gesteuert.

A4 FlexHub White and Yellow Pages

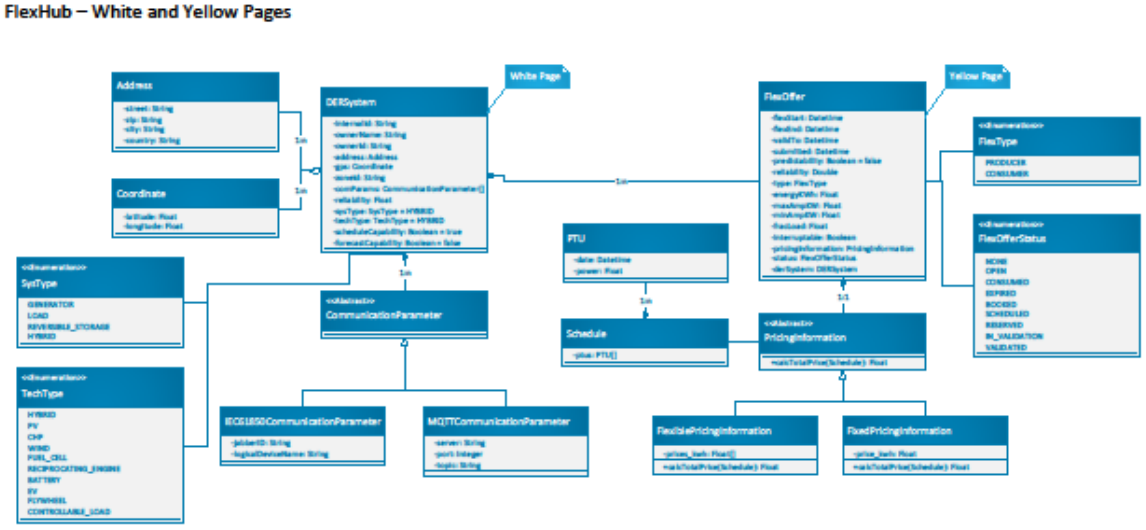


Abbildung 2: UML Klassendiagramm White und Yellow Pages

A5 Bedrohungsanalyse IRES Flexibilitätsmarkt

EnergieDock

Bedrohungsanalyse IRES Flexibilitäts-markt

A6 Test Dokumentation NEMO.spot

	status.feature	1
/transactions		8
	transactions.feature	8
/users		19
	users.feature	19

Im diesem Dokument beiliegenden Verzeichnis *test-reports* befindet sich die sowohl die schrittweise Dokumenten der einzelnen Test-Cases als auch die jeweiligen Test-Ergebnisse. Die Test-Cases sind dabei jeweils als Test-Feature strukturiert. Ein Feature beinhaltet dabei ein Test-Szenario. In einem Test-Szenario wird zunächst die benötigte Test-Umgebung erzeugt. Das bedeutet, es werden die Datenobjekte erzeugt, die für diesen Test-Case benötigt werden. Dann finden entsprechend dem Szenario die Anfragen an das NEMO.spot-Backend statt und es wird gegen das erwartete Verhalten im Erfolgs- oder Fehlerfall getestet. Die Unterteilung in unterschiedliche Features erfolgt primär, um die Tests logisch zu strukturieren und im Fehlerfall ein übersichtlicheres Debugging zu ermöglichen.

Den HTML-Dateien im *test-reports* Verzeichnis kann die detaillierte, schrittweise Beschreibung der Test-Cases in der BDD-Syntax entnommen werden. Gleichzeitig wird die Ausführungszeit eines jeden Schrittes in Millisekunden angegeben und eine grüne Zeile weist auf eine fehlerfrei ausgeführte Zeile hin. (Graue Zeilen sind ergänzende Kommentare und gelbe Zeilen Szenario-Beschreibungen.) Dazu ist die Datei *karate-summary.html* in einem Web-Browser zu öffnen.

A7 Schnittstellenbeschreibung NEMO.spot

NEMO.spot

Overview

REST API Beschreibung des NEMO.spot-Flexibilitätsmarkts.

Autorisierung:

Die Autorisierung erfolgt über ein JWT das im Auth Header des HTTP Requests als Bearer Token übergeben wird. Das JWT kann über die Login-Schnittstelle (siehe `POST /sessions/login`) mit den Anmeldedaten (E-Mail und Passwort) des Benutzers erhalten werden. JWTs werden vom Flexmarkt aus drei möglichen Gründen devalidiert. In diesem Fällen ist eine erneute Anmeldung über die Anmeldedaten nötig, um ein neues JWT zur Autorisierung zu erhalten. 1. Nach 60 Minuten läuft das JWT ab (expired). 2. Nach 15 Minuten Inaktivität läuft die Session des Benutzers ab. 3. Ein erneuter Login mit den gleichen Anmeldedaten hat den JWT devalidiert.

Tritt einer dieser Fälle auf, antwortet der Flexmarkt mit einem 401 und gibt den Grund für das Ablaufen der Autorisierung an. In all diesen Fällen kann durch ein erneutes Anmelden wieder ein gültiger JWT bezogen werden.

Benutzergruppen

Der Flexmarkt kennt drei Benutzergruppen, die über die hier beschriebenen Schnittstellen mit dem Markt interagieren können: 1. FlexProvider: FlexProvider sind Benutzer die Flexibilitäten anbieten. Sie erstellen DERSysteme für ihre Ressourcen, die dann Flexibilitäten in Form von FlexOffern anbieten. Dabei können sie ihre Ressourcen und Flexibilitätsangebote einsehen und verwalten, sowie ihre Buchungen und zugehörige Fahrpläne einsehen. FlexProvider können außerdem nach Lieferung einer Flexibilität einen Report zur Leistungserfüllung schicken und ihre monatlichen Settlements einsehen. 2. FlexConsumer: FlexConsumer sind Benutzer die Flexibilitäten buchen (z.B. als Verteilnetzbetreiber zum Netzpassmanagement). Sie können nach Flexibilitäten suchen und diese Angebote buchen. Sie können ihre Buchungen einsehen und Fahrpläne zu gebuchten Angeboten schicken (und einsehen). Außerdem können sie ihre monatlichen Abrechnungen abrufen und die Optimierungs/Aggregator Schnittstelle nutzen. Über diese Schnittstelle können sie nach aggregierten Lösungen für Zielfahrpläne suchen, die mehrere Flexibilitäten (FlexOffern) zur Lösung benötigen. 3. VNB Kapazitätsmanager: Verteilnetzbetreiber die für einen Netzbereich als Kapazitätsmanager verantwortlich sind und bestimmte Buchungen im Hinblick auf ihre Netzverträglichkeit hin untersuchen und validieren.

Paths