

# Upilio – Reallabor als Energiedatenmanagement

**Arlt, Stefan-Alexander; Mock, Markus U.**

Hochschule Landshut, University of Applied Sciences

<https://doi.org/10.57688/367>

## Kurzfassung

Energieeffizienz – Einsparpotenziale erkennen – CO<sub>2</sub>-Emissionen erfassen und reduzieren zu können. Das war bereits 2015 Anlass und Initiative der Beteiligten, für die Hochschule Landshut (HAW) ein gebäudebezogenes Energiedatenmanagementsystem (EnMS) zur Erfassung verbrauchsrelevanter Daten auf dem Campus zu planen und mit der Umsetzung zu beginnen.

Einer der wichtigsten Anforderungen für ein EnMS war, die unterschiedlichen Energiearten und ihre jeweiligen Verbräuche in den einzelnen Gebäuden der Hochschule zu erfassen. Bis dahin konnte nur der gesamte Energieverbrauch der Hochschule auf Basis der Abrechnungen des Energieversorgers beziffert werden. Die Daten des Campus wurden bis dahin meist noch händisch erfasst, vereinzelt gab es eine eingeschränkte automatische Erfassung. Insgesamt führte das zu einer asynchronen zeit- und personalintensiven Datenerfassung an den bis dahin existierenden Messstellen.

Um die Daten analysieren und bewerten zu können, müssten sie zusammengefasst und abgespeichert werden können.

Mit dem Team ist es gelungen, eine automatisierte und synchrone Erfassung der an der Hochschule anfallenden Energie- und anderer für Energieverbräuche relevanten Daten wie Wasserverbräuche und Wetterdaten aufzubauen.

In dem mittlerweile existierenden Reallabor werden gebäudeweise sämtliche verbrauchsrelevanten Daten, mit rund 100 Messstellen auf dem Hauptcampus, erfasst. Die Messwerte werden im Viertelstundentakt ausgelesen.

Ziel unseres Projektes ist der Aufbau eines herstellerunabhängigen EnMS, das sowohl für den technischen Dienst als auch für die Lehre, Forschung und Entwicklung genutzt werden kann. Als eine der ersten Hochschulen – wenn nicht sogar als erste Hochschule – in Bayern wird hier ein derartiges EnMS in dieser umfassenden Ausbaustufe und Projektkonstellation betrieben.

## 1. Beschreibung Upilio

Upilios praktischer Zweck war zunächst, dem Gebäudemanagement unserer Hochschule und in Unternehmen zu helfen, die Energieeffizienz zu verbessern. Dazu sammelt Upilio Energieverbrauchsdaten auf hochauflösender Ebene in Echtzeit und macht

die Daten zugänglich für halbautomatische und automatische Analyse und Visualisierung. Es enthält auch automatisierte Anomalie-Erkennungsmethoden, die erkennen ob z.B. außergewöhnliche Verbrauchs- oder Produktionsmuster vorliegen. Zur Unterstützung der Gebäudemanager bei der Konzeption und Bewertung der Wirksamkeit von Energiesparmaßnahmen ist Upilio so konzipiert, dass es typische OLAP (= Online Analytical Processing)-Operationen realisiert.

Eine von Upilios Hauptinnovationen aus Sicht der Informatik besteht in der Implementierung von OLAP-Operationen mit dem Serverless-Paradigma.

## 2. Echtzeit Datenerfassung

Seit 2018 wird mit dem Gebäudemanagement der Hochschule Landshut zusammengearbeitet, um nach und nach verschiedene Energieverbräuche in unseren Gebäuden durch Instrumentierung mit digital ablesbaren Messsensoren zu erfassen.

Der Hauptcampus in Landshut ist schon nahezu vollständig ausgerüstet. Dort werden Daten für den Stromverbrauch, regionale Fernwärme, an die die Hochschule angeschlossen ist, und Wasserverbrauch an verschiedenen Messpunkten erfasst. Auch die örtliche Wetterstation der Hochschule mit Daten über die Temperatur, relative Luftfeuchtigkeit und Globalstrahlung ist mit eingebunden und liefert ihre Daten auch an die offene Wetterdatenplattform Open Weather Map [1], von der auch Wetterdaten aus der Umgebung Landshuts für Upilio abgerufen werden.

Zunächst wurde nur der Energiebezug betrachtet. Interessant war und ist es für uns daher, die Betrachtung auf eine Bereitstellung von Energie auszuweiten. Deshalb wurde im Rahmen eines Neubaus an der Hochschule vom Staatsbauamt eine Photovoltaikanlage in Gebäude A errichtet, die bis zu 20 kW elektrische Leistung in der Spitze produzieren kann. Diese Energie ist wetter- und jahreszeitabhängig, also fluktuierend.

Mit Upilio wurde eine Plattform geschaffen, mit der Daten in einer Auflösung zwischen 1 Sekunde bis 15 Minuten gesammelt werden können. Die Frequenz ist für jeden Messpunkt flexibel konfigurierbar und kann dynamisch von der Online-Plattform geändert werden.

Alle Daten werden im Cloud-Speicher von Upilio gespeichert. Das erspart Aufwand und Kosten, weil eine direkte Kommunikation von den Datenerfassungsorten, d. h. vom Hochschulgelände zur Cloud ermöglicht wird. Das bedeutet, dass weder Kosten für Wartung und Pflege von Servern, noch Investitionen in zusätzliche Server anfallen.

Um die Anzahl der Datenverbindungen von jedem Standort zu begrenzen und eine Möglichkeit zu haben, um zwischen Datenaktualisierung und Netzbandbreite abzuwägen, kommunizieren Messgeräte nicht direkt mit dem Upilio Cloud-System in AWS.

Stattdessen verwendet jeder Standort einen – bei großen Standorten potenziell mehrere – Daten-Konzentratoren. Derzeit kommen Raspberry Pi 4 (kurz: RPI) Einplatinencomputer für diesen Zweck, wie in Abbildung 1 dargestellt, zum Einsatz.

Verwendet wird die AWS IoT-Plattform Greengrass, um das Management von den Edge-Geräten zu vereinfachen. Unter Edge versteht man in der Informatik Rechner, Sensoren und andere Komponenten, die nicht in der Cloud, sondern vor Ort eingesetzt werden. Dezentrale installierte Komponenten haben meist den Nachteil, nicht ohne direkten Zugang softwaretechnisch aktualisiert werden zu können. Upilio löst dieses Wartungsproblem dadurch, dass auf jedem RPI der AWS Greengrass-Core ausgeführt wird, mit dem unser Edge (= vor Ort) -Verarbeitungscode aus der AWS-Cloud aktualisiert werden kann. Häufig können sogar dieselben Funktionen, die im Cloud-Backend ausgeführt werden, direkt wiederverwendet werden.

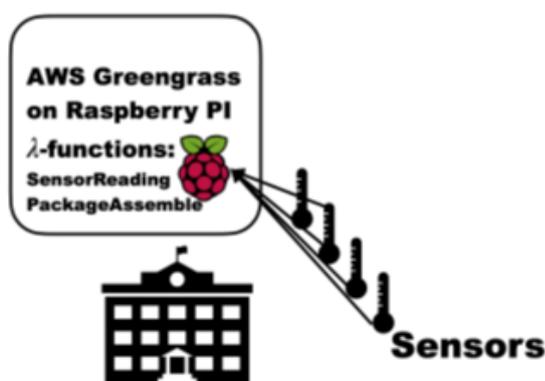


Abb. 1: Raspberry Pi Einplatinencomputer als Datensammler und Konzentrador. Mehrere Messungen werden zu Datenpaketen zusammengefasst und an unser Upilio-Backend gesendet.

Jedem Sensorgerät wird eine eindeutige Geräte-ID zugeteilt. Zur Anwendung kommt das NTP-Protokoll, um die Uhren unserer Konzentratorknoten so mit der offiziellen Atomzeit synchron zu halten, damit die Messungen korreliert werden können, die an verschiedenen Sensoren vorgenommen wurden. Eine Korrelation ist innerhalb der NTP-Synchronisationsgenauigkeit, die im Millisekundenbereich liegt, möglich, was mehr als ausreichend ist für die Ablesefrequenz, mit der der Energieverbrauch erfasst wird.

### 3. Optimierung der Datenerfassung

Neben der Echtzeit-Datenerfassung stellt Upilio auch einen dateibasierten Eingangspfad bereit. Dieser Eingangskanal funktioniert dadurch, dass eine CSV-Datei an einem vorkonfigurierten Upload-Pfad über eine URL im Amazon Simple Storage Service in einem so genannten S3-Bucket abgelegt wird. In Upilio ist eine AWS Lambda-Funktion konfiguriert, die diesen Bucket überwacht und von AWS automatisch aktiviert wird,

sobald neue Dateien in dem Bucket abgelegt werden. Um zu vermeiden, dass implementierte Verarbeitungsfunktionen dupliziert werden, liest diese Funktion einfach die Zeilen in der CSV-Datei und schickt die Daten unter Verwendung des Echtzeit-Eingangskanals von Upilio an das System, so als ob die Daten direkt von einem RPI-Datenkonzentrator stammen würden.

Obwohl der Dateieingangspfad konventionell gestaltet ist, hat dieser seine Daseinsberechtigung in der Praxis, da viele KMU-Unternehmen keine eigene IT-Abteilung haben oder über nicht ausreichende IT-Kenntnisse verfügen, um eine Echtzeit-Streaming-Lösung einzurichten.

Eine weitere Erkenntnis war, dass ein einfacher Datei-Push-Mechanismus ein nützlicher Schritt in Richtung eines flexibleren und häufigeren Dateneingangs sein kann, da aufgrund der gesetzlichen Bestimmungen viele Organisationen in Deutschland bereits verpflichtet sind, ihren Energieverbrauch zu erfassen. Leider wird in KMU-Unternehmen noch viel zu häufig die Erfassung umständlich, asynchron und kostenintensiv, weil fußläufig und manuell oder halbautomatisch durch Erstellen von Excel-Dateien durchgeführt [2,3].

#### 4. Die grafische Benutzeroberfläche von Upilio

Die folgende Abbildung zeigt eine schematische Darstellung der Realisierung der grafischen Benutzeroberfläche von Upilio (= Upilio Frontend) im Serverless-Paradigma, siehe Abbildung 2. Der Schlüssel zu einer serverlosen webbasierten grafischen Benutzeroberfläche ohne Betrieb eines permanent laufenden Servers basiert auf der Tatsache, dass das AWS Simple Storage Service (S3) es zulässt, Buckets für alle lesbar zu machen, und die AWS-Buckets mit Web-URLs adressierbar sind. Für die Verwaltung der zugelassenen Benutzer kommt der AWS Cognito Identity Service zum Einsatz. Mit der Erstellung eines Benutzerpools, um den Zugriff auf das Upilio-Frontend-System zu limitieren, konnte auch dieses Problem gelöst werden. Alle Zugriffe auf Frontend-Seiten (alle gehostet in S3) werden über dieses System authentifiziert, so dass außer der Login-Seite alle Seiten nur nach vorheriger Anmeldung zugänglich sind.



Abb. 2: Der Tornado-Anmeldebildschirm. Die Seite zeigt einen Login und verlangt, dass sich der Benutzer beim System anmeldet.

Sobald der Benutzer angemeldet ist, sieht er ein Dashboard wie in Abbildung 3 dargestellt.



**Abb. 3:** Auf der linken Seite sind die Auswahlelemente ersichtlich. Rechts sind der kumulierte elektrische Energiebezug (Zählerstände in zeitlicher Auflösung) des Gebäudes D und die Niederspannungsverteilung im Labor West-Bereich der Hochschule Landshut dargestellt. Parallel werden auch die entsprechenden 15-min-Leistungswerte der Gebäude bzw. der Gebäudekomplexe angezeigt.

Im Navigationsbereich links kann er zwischen einer Übersicht über alle verbundenen Standorte (Campus), einer Live-Daten-Ansicht (Abbildung 3) oder dem Analyse-Dashboard auswählen. Auch können die Kontoeinstellung eingesehen und geändert werden oder der Nutzer kann sich wieder ausloggen. Die Live-Datenansicht wird über AppSync generiert. AppSync ist ein AWS-Service, der eine Schnittstelle bereitstellt, die kompatibel mit der Open-Source-GraphQL Abfragesprache ist, die zum Abfragen und Anzeigen von Grafikdaten von Facebook entwickelt wurde. Upilio verwendet GraphQL, um Dynamo DB abzufragen und aktuelle Sensorwerte darzustellen.

Die Ressourcen, die ausgewählt werden können, sind nicht im Frontend fest verdrahtet, sondern werden aus der serverlosen Aurora-Datenbank abgerufen. Die Datenbank enthält Metadaten, in der die verfügbaren Sensoren, die Art der Ressource, die sie darstellen, ihre für Menschen lesbaren Namen usw. verankert sind. Sie verwendet diese Informationen, um die Dropdown-Menüs zu befüllen, also z.B. "HAW" für den Hauptcampus.

Abbildung 4 zeigt die Dashboard-Ansicht, wenn der Benutzer die Option zum Ausführen von Analysen im OLAP-Stil ausgewählt hat. In der Abbildung wurden zwei Sensoren, die PV-Produktion und die Globalstrahlung für ein 24-Stundenintervall ausgewählt.

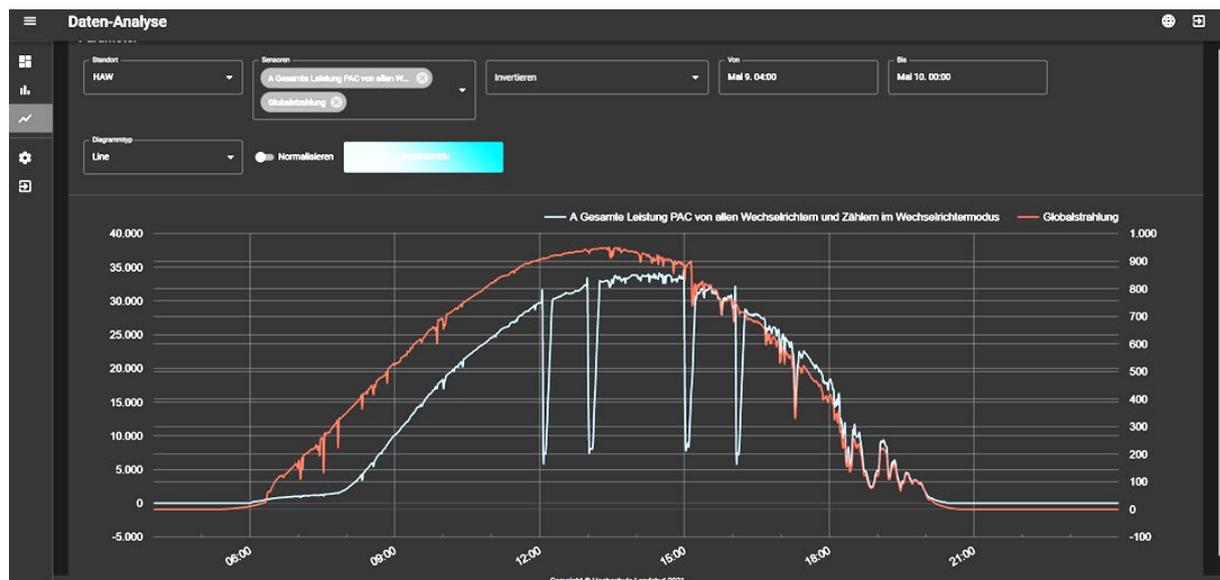


Abb. 4: Darstellung der Gesamtenergie, die von der Solaranlage in Gebäude A erzeugt wurde (linke Skalierung), gemeinsam mit der von der lokalen Wetterstation gemessenen Globalstrahlung, rechte Skalierung.

## 5. Perspektiven von Upilio

Wie mit diesem Projekt gezeigt werden konnte, können zukünftige Energiedatenmanagement-Systeme in der Lage sein, herstellerübergreifend die unterschiedlichsten Arten von Messgeräten und Sensoren diverser Medien (z.B. Gas, Wasser, Strom, Wärme, diverse Gase, Druckluft sowie Wetterdaten) abzubilden. Die Systeme können intervallflexibel die Messdaten diverser Messgeräte empfangen, validieren und verarbeiten. Zusammen mit diesem System können auch die bekannten Prozesse im Rahmen des Gerätelebenszyklus (Einbau, Ausbau, Wartung und Prüfung) mit den stammdatenführenden Systemen unterstützt werden.

Ferner können alle Rohdaten und aufbereiteten Informationen in der Datenbank auch von anderen Applikationen verarbeitet werden, sodass die Informationen von nachgelagerten Systemen korreliert werden können. Der modulare Aufbau der Cloud-Anwendungen besteht aus einer Vielzahl von diesen individuell, unabhängig voneinander einsetzbaren Lambda-Funktionen. Upilio kann jedwede Art von Messdaten – unabhängig von Dimension, Maßeinheit und Volumen – in Empfang nehmen, validieren und persistieren [4]. Darüber hinaus werden die Informationen aufbereitet, damit Applikationen als Konsumenten diese Daten zeitnah, qualitativ hochwertig und in der gewünschten Granularität nutzen können, um Entscheidungen zu treffen, Geschäftsprozesse zu betreiben und Informationen den Mitarbeitern sowie anderen Stakeholdern zu kommunizieren.

Auch die kaufmännischen Systeme können bei Bedarf die relevanten Informationen für Prozesse wie Abrechnung, Energiemengenbilanzierung, Marktkommunikation etc. entsprechend abfragen.

## 6. Betriebskosten von Upilio

Neben der einfachen Bereitstellung sind die Betriebskosten in der Praxis von entscheidender Bedeutung. Upilio ist seit einigen Monaten betriebsbereit und noch sind nicht alle neu geplanten und im vergangenen Jahr errichteten Hochschulgebäude angebunden, daher konnten auch die bisher separat gespeicherten Daten integriert werden, um die Betriebskosten für verschiedene Energiemanagement-Installationsgrößen abzuschätzen. Die Kosten, die hier Berücksichtigung fanden, sind die Kosten, die für die anfängliche Datenverarbeitung (Schnittstellen-Aufrufe, SNS-, SQS- und Lambda-Funktionen) und für die Datenspeicherung anfallen. Kosten, die durch die Durchführung zusätzlicher Datenanalysen entstehen, müssen separat berechnet werden. Unsere Schätzungen basieren auf der Verarbeitung der von uns aktuell erfassten Sensordaten.

Dies führt zu Datenpaketgrößen von etwa 8,2 KB und 1.440 Datenpaketen pro Sensor pro Tag im 24-Stunden-Betrieb. Von der Annahme ausgehend, dass jeder Sensor jede Sekunde ausgelesen wird und diese Daten jeweils nach einer Minute gemeinsam übertragen werden.

Bei einer kleinen Anzahl von Geräten fallen für SNS und SQS keine Kosten an, da der Datenverkehr bei diesem Volumen vollständig unter die kostenlose Stufe von AWS fällt. Selbst für eine große Anzahl von Messgeräten sind die Kosten mit weniger als 30 Cent pro Gerät und Monat durchaus sehr günstig.

Für das Umfeld der Hochschule, mit der Größenordnung von 100 bis 200 Messpunkten, werden Betriebskosten von weniger als 25 USD pro Monat [5] erwartet, sobald das System vollständig bereitgestellt ist. Außerdem werden nicht alle Messungen in Intervallen von einer Sekunde durchgeführt, beispielsweise bei der Erfassung von Fernwärmeeinsatz. Ein Intervall von einer Minute ist völlig ausreichend, damit die Schätzungen als Obergrenze für die Kosten behandelt werden können.

Upilio ist nicht nur kosteneffizient, sondern weist auch hervorragende Eigenschaften für die Kostenskalierung auf: mit weniger erfassten Sensoren sinken die Kosten fast proportional, und aufgrund des bei AWS verfügbaren Freikontingents sind kleine Upilio-Instanzen extrem kostengünstig. Auf der anderen Seite sind Quartierlösungen mit Zehntausenden von Messpunkten auch zu vernünftigen Kosten realisierbar, da mehrere Tausend Euro monatliche Betriebskosten für z.B. kommunale Betreiber nahezu „aus der Portokasse“ bezahlbar sein dürften.

## 7. Zusammenfassung

Im Rahmen dieser Arbeit wurde ein funktionsfähiges serverless Energiedatenmanagement-System implementiert und vorgestellt. Aus diesem geht eine für den Produktiv-einsatz geeignete und kostengünstige Anwendung hervor.

Das System dient als Ausgangsbasis für eine alltagstaugliche Umsetzung. Für individuelle Anpassungen und besondere Adaptionen wären bei Bedarf weitere Entwicklungs- und Implementierungsarbeiten erforderlich.

Die aus dem Projekt hervorgehende gesamte Anwendung konnte in einem der derzeit größten Reallabore erprobt werden. Unter Realbedingungen konnten bisher alle Anforderungen umgesetzt werden, welche wiederum in die Weiterentwicklung des Energiedatenmanagementsystems eingeflossen sind. Darüber hinaus sollte dieses System unter realistischen Bedingungen und von verschiedenen Anwendern eingehenden Alltagsprüfungen unterzogen werden.

Ebenfalls im Rahmen der Anwendung konnte geprüft werden, welche Anreize den Dateneignern geboten werden müssten, damit sie bereit wären, ihre Daten in definierten Abstufungen von unterschiedlicher Aussagekraft zu teilen. Ein entsprechendes Energiedatenmanagement-System dient als Sammelstelle für alle im Rahmen dieses Projektes angefallenen Energiedaten.

Aktuell wird Forschung im Umgang mit Big Data betrieben. Bis zum Projektende werden die anfallenden Daten zum Zwecke der Forschung und Entwicklung weiter genutzt. Nach dem Projektende soll das System für die Lehre und zur konsekutiven Ausbildung eingesetzt werden.

Die im ursprünglichen Projekt beteiligten Wissenschaftler werden die Weiternutzung der Daten koordinieren. Diese Arbeit leistet einen Beitrag und stellt ein Konzept vor, das wie in Upilio gezeigt, in der Lage ist, herstellerübergreifend die unterschiedlichsten Arten von Messgeräten und Sensoren diverser Medien abzubilden, intervallflexibel deren Daten zu empfangen, zu validieren und zu verarbeiten sowie Prozesse im Rahmen stammdatenführender Systeme zu unterstützen. Ferner sollten alle Rohdaten und aufbereiteten Informationen in der Datenbank von anderen Applikationen verarbeitet werden können, sodass die Informationen von nachgelagerten Systemen korreliert werden können.

## Literatur

- [1] <https://openweathermap.org/>, Zugriff am 15.12.2021.
- [2] Zahner, Christian; Arlt, Stefan-Alexander: Aspekte der automatischen Zählerstandsfernübertragung in Versorgungsunternehmen. GWF Erdgas, Essen, Jg. 138, 1997, Nr. 6, S. 290-298.

- [3] Arlt, Stefan-Alexander: Assigning Costs for Energy Management. Metering international, Scarborough Publications International Steenberg – South Africa, Issue 1, 1999, p. 39.
- [4] Mock, M.; Huber, F.; Arlt, St.-A.; Energiedatenmanagement in einer im Frontend Cloud Computing Umgebung. GWF Gas+Energie, Essen, Jg. 162, 2021, Nr. 6, S. 42-53.
- [5] Huber, Florian; Mock, Markus: Toci: Computational Intelligence in an Energy Management System. 2020 IEEE Symposium Series on Computational Intelligence; Canberra, Australia, December 1-4, 2020.

## Kontakt

Prof. Dr. rer. nat. Stefan-Alexander Arlt  
Hochschule Landshut  
Fakultät Elektrotechnik und Wirtschaftsingenieurwesen  
Am Lurzenhof 1  
D-84036 Landshut  
E-Mail: [Stefan-Alexander.Arlt@haw-landshut.de](mailto:Stefan-Alexander.Arlt@haw-landshut.de)

Prof. Dr. Markus U. Mock  
Hochschule Landshut  
Fakultät Informatik  
Am Lurzenhof 1  
D-84036 Landshut  
E-Mail: [Markus.Mock@haw-landshut.de](mailto:Markus.Mock@haw-landshut.de)