

Landesamt für Geoinformation und Landentwicklung Baden-Württemberg, Hochschule für Technik Stuttgart

SMART VILLAGES – VERNETZUNG VON 3D-GEOINFORMATION UND UMWELTRELEVANTEN SENSORDATEN IM LÄNDLICHEN RAUM

Monique Vögele, Thunyathep Santhanavanich, Patrick Würstle, Gerald Graf, Volker Coors

Zusammenfassung: Das „Smart Cities“-Konzept wird in größeren Städten angewandt, jedoch aufgrund des Personal- und Strukturkostenbedarfs kaum in kleinen und mittleren Kommunen. Dieser Beitrag konzentriert sich auf die Übertragung des „Smart Cities“-Konzepts auf Kommunen im ländlichen Raum. Für den Anwendungsfall wird dieses Konzept im Rahmen des Projekts „Smart Villages – attraktive Orte im Ländlichen Raum“ in der Gemeinde Wüstenrot mittels einer 3D-Webplattform umgesetzt. Die Plattform integriert Sensordaten in das 3D-Stadtmodell und eröffnet der Gemeinde die Möglichkeit, vorhandene Sensordaten nachhaltiger Energiequellen über die 3D-Webplattform so zu präsentieren, dass sich daraus ein Mehrwert für die Anwender ergibt. Ein möglicher Weg zur Lösung dieser Herausforderung wird anhand einer Studie über die Vernetzung von 3D-Stadtmodellen und heterogenen Sensordaten unter Verwendung von OGC-Standards durchgeführt: CityGML und SensorThings-API anhand eines Konzepts namens CityThings.

Schlüsselwörter: Smart Cities, 3D-Webvisualisierung, Geoinformation, Sensordatenvernetzung, CityGML, SensorThings-API

SMART VILLAGES – INTEGRATION OF 3D-GEOINFORMATION AND SENSOR DATA WITH ENVIRONMENTAL RELEVANCE IN RURAL AREAS

Abstract: The Smart Cities concept is being deployed in larger cities, but hardly ever in small and medium-sized municipalities due to the personnel and structural cost requirements. This paper focuses on the transfer of the “Smart Cities” concept to rural municipalities. As for a scheme to tackle this challenge, a study on a link between heterogeneous sensor data and 3D city models is conducted by utilizing OGC standards: CityGML and SensorThings API with a novel concept called CityThings. For the use case, this concept is implemented in the context of the “Smart Villages” project in the area of Wüstenrot using a 3D web platform. The platform integrates sensor data into the 3D city model and gives the municipality the possibility to present existing sensor data of sustainable energy sources via the 3D web platform in a way that results in added value for the users. A possible way to solve this challenge is a study on the networking of 3D city models and heterogeneous sensor data using OGC standards: CityGML and SensorThings API using a concept called CityThings.

Keywords: Smart Cities, 3D web-visualisation, geoinformation, sensor data integration, CityGML, SensorThings API

Autoren

Dipl.-Geogr. Monique Vögele

Dipl.-Ing. (FH) Gerald Graf

Landesamt für Geoinformation und Landentwicklung Baden-Württemberg (LGL)

Büchsenstraße 54

D-70174 Stuttgart

E: monique.voegele@lgl.bwl.de

gerald.graf@lgl.bwl.de

M. Sc. Thunyathep Santhanavanich

M. Sc. Patrick Würstle

Prof. Dr.-Ing. Volker Coors

Hochschule für Technik (HFT) Stuttgart

Schellingstraße 24

D-70174 Stuttgart

E: thunyathep.santhanavanich@hft-stuttgart.de

patrick.wuerstle@hft-stuttgart.de

volker.coors@hft-stuttgart.de

1 EINLEITUNG

Das Projekt „Smart Villages – attraktive Orte im Ländlichen Raum“, bei dem das Landesamt für Geoinformation und Landentwicklung (LGL) und die Hochschule für Technik Stuttgart (HFT) zusammenarbeiten, ist Teil der Digitalisierungsstrategie des Landes Baden-Württemberg digital@bw¹.

Hauptziel ist es, Gemeinden im ländlichen Raum durch die Bereitstellung von dreidimensionalen Geobasisdaten über eine Webschnittstelle bei der Umsetzung von Smart-City-Konzepten zu unterstützen. In den beiden Teilprojekten „Architekturmodelle/BIM“ und „Verknüpfung 3D-Modell mit Fachdaten“ werden die Bedürfnisse der Gemeinden Niedernhall und Wüstenrot mit den zur Verfügung stehenden technischen Möglichkeiten abgeglichen.

Das LGL setzt für die 3D-Webpräsentation die Softwareprodukte der Firmen virtualcitySYSTEMS (VCS) und M.O.S.S. (VEGA-3D) ein, die als Webanwendung die Schnittstelle zum Benutzer bilden und auf Cesium² basieren. Die 3D-Webplattform besteht aus dem 3D-Geländemodell (DGM) und dem 3D-Gebäudemodell (LoD2) mit dreidimensionalen Geobasisdaten von hoher Qualität und Aktualität. Zusätzlich werden zahlreiche Dienste wie unter anderem das Digitale Orthophoto (DOP), die Digitale Topographische Karte (DTK) sowie die Katasterkarte (ALKIS) zur Darstellung der Oberfläche genutzt.

Die HFT stellt die Ergebnisse auf Basis des Cesium-Globus dar und realisiert die prototypische Implementierung mit dem Open-Geospatial-Consortium-(OGC-)³ Community-Standard 3D-Tiles⁴ zur performanten Übertragung und Visualisierung der 3D-Daten im Internet. Als Open-Source-Implementierung der SensorThings-API (STA)⁵ steht die Open-Source-SensorThings-Server-Software (FROST)⁶ des Fraunhofer Instituts für Optronik, Systemtechnik und Bildauswertung (Fraunhofer IOSB) zur Verfügung. Auf dieser Basis können technische Umsetzungen entwickelt und anschließend in die 3D-Webplattform des LGL integriert werden.

Für die Zielgruppe dieser 3D-Webplattform, den Bürgerinnen und Bürgern sowie kommunalen Entscheidungsträgern kleiner und mittlerer Gemeinden, kann ein klarer Mehrwert geschaffen werden, indem ein einfacher Zugang zum 3D-Gebäudemodell Aufwand und Kosten reduziert, und in-

novative Ansätze in Planung und Bürgerbeteiligung ermöglicht (Graf et al. 2016).

Ziel ist es, den Anwendern im ländlichen Raum die Möglichkeit zu bieten, sich aktiv an der Nutzung der 3D-Webanwendung zu beteiligen. Dazu gehören z. B. die integrierte Realisierung eigener webbasierter Planungen, die Unterstützung von Bürgerbeteiligungsverfahren, die automatische Übernahme von BIM-Modellen bei städtebaulichen Entwicklungen oder die Verknüpfung von Sensordaten mit der 3D-Webanwendung.

Das Teilprojekt „Architekturmodelle“ mit der Stadt Niedernhall ist überwiegend abgeschlossen.

Der Fokus dieses Beitrags liegt auf dem aktuell bearbeiteten Projektteil „Verknüpfung 3D-Modell mit Fachdaten“, wobei Lösungsansätze für innovative 3D-Nutzungsszenarien entwickelt werden, welche die Integration von 3D-Stadtmodellen mit Fachdaten und Echtzeit-Sensordaten ermöglichen.

In Kooperation mit der Gemeinde Wüstenrot, einer Gemeinde auf dem Weg zur Plus-Energiegemeinde, sollen die vorgesehenen Maßnahmen der innovativen und nachhaltigen Energiegewinnung über die 3D-Webanwendung und über die Darstellung der mit den Geobasisdaten vernetzten Sensordaten der Gemeinde für die Gemeindeverwaltung und für Laien zugänglich gemacht werden. Von Vorteil für die Anwendung ist ein einfacher Zugriff auf die vorhandenen Sensordaten, die durch zwei verschiedene IT-Dienstleister bereitgestellt werden. Die Datengrundlage bietet die Gemeinde durch ihr eigenes Energieversorgungsunternehmen „Energieversorgung Mainhardt Wüstenrot (EMW)“⁷.

Die Darstellung, die dem Bürger das Energiemanagement der Anlagen in Wüstenrot zeigt, wird über ein Dashboard erfolgen. Touristen und Interessierte können sich internetbasierend über eine Story Map den von der Gemeinde konzipierten „Energieerlebnispfad“ darstellen lassen. Der Energieerlebnispfad soll Touristen und Interessierten die Möglichkeit verschaffen, die Ergebnisse der realisierten Projekte der alternativen, nachhaltigen Energiegewinnung kennenzulernen.

Eine Herausforderung bei der Webvisualisierung von 3D-Geodaten und der Vernetzung von Sensordaten ergibt sich aus der Vielzahl verschiedener Sensortypen

mit ihren unterschiedlichen Datenformaten. Daher bietet die HFT die technische Unterstützung an, um die verschiedenen Formate über die genormte Schnittstelle STA zusammenzuführen und somit eine Interoperabilität herzustellen. Dies soll ein Anstoß für Firmen sein, künftig nicht eigene produktspezifische Formate, sondern ein STA-kompatibles Format anzubieten. Weiterhin muss eine verteilte Datenhaltung berücksichtigt werden, denn Geobasisdaten und Sensordaten werden im Allgemeinen durch verschiedene Einrichtungen verwaltet und entsprechend auf unterschiedlichen Servern bereitgestellt.

Um eine übertragbare Lösung und damit einen breiten Nutzerkreis zu erreichen, basiert das Konzept auf den OGC-Standards CityGML⁸ und STA. STA bildet hierbei die Schnittstelle zu den Sensoren und Datenströmen, während CityGML – als urbanes Informationsmodell – für die Vernetzung von Sensor- und Geodaten sorgt (Coors & Schneider 2018). Nach Projektabschluss sollen die Vorteile einer standardisierten Einbindung von Sensordaten klar definiert sein.

2 STAND DER TECHNIK

In den letzten Jahren wurden verschiedene Forschungsarbeiten zur Integration von Sensordaten und eines 3D-Stadt- und Landschaftsmodells auf Basis von OGC-Standards durchgeführt (Chaturvedi & Kolbe 2015, Chaturvedi et al. 2019, Kasprzyk et al. 2019, Zhu et al. 2016, Jazayeri 2015). Für das semantische 3D-Stadtmodell wurde das CityGML-Modell häufig als Speicherung und Austausch von virtuellen 3D-Stadtmodellen verwendet (Gröger et al. 2012). Es wurden mehrere Ansätze zur Integration des CityGML-Modells mit den dynamischen Sensordaten verfolgt.

Dieser Beitrag konzentriert sich auf die Methoden und Werkzeuge, die den OGC-Standards entsprechen.

Chaturvedi & Kolbe 2015 schlagen mit dem Dynamizer ein Konzept zur Speicherung und zum Austausch von Sensor- oder Zeitseriendaten im CityGML-Modell vor. Es dient zur Modellierung und Implementierung der dynamischen Eigenschaften eines bestimmten Objekts in den semantischen 3D-Stadtmodellen. Dies ermöglicht die Darstellung dynamischer und zeitvariabler Attribute direkt im 3D-Stadtmodell im CityGML-Format.

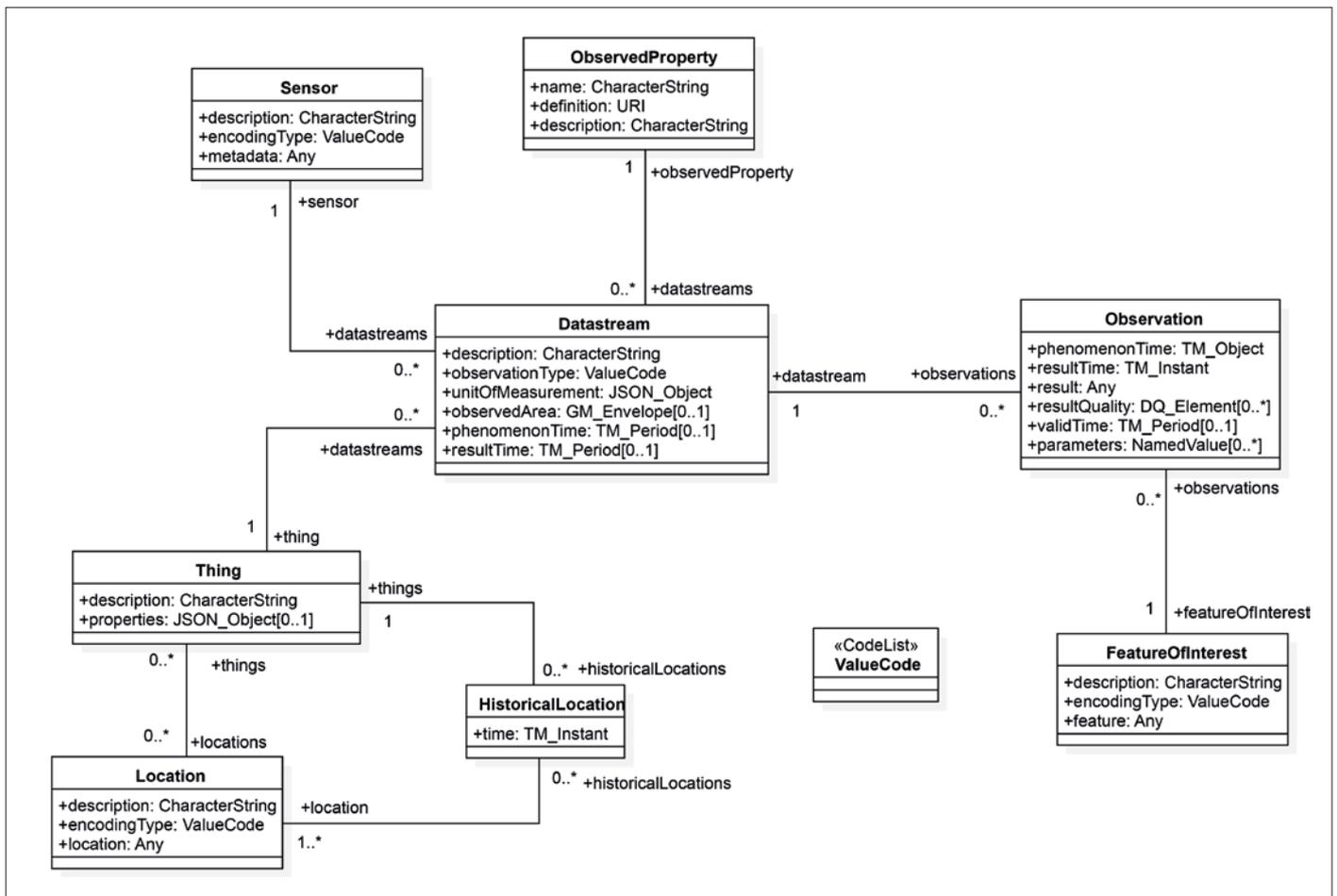


Abbildung 1: UML-Diagramm mit SensorThings-Elementen (Liang et al. 2016a)

Es unterstützt mehrere Codierungen der dynamischen Daten, darunter sind die Domain-Range und Time-Value Pair Encoding⁹. Eine weitere Möglichkeit ist ein Verweis auf externe Sensordienste wie OGC Sensor Observation Service (SOS). Jedes CityGML-Modell kann mehrere darstellbare Codierungen der dynamischen Daten enthalten.

Es wurden exemplarische Anwendungsfälle für den Einsatz von Dynamizern implementiert, um die dynamischen Eigenschaften des Gebäudes wie Wärmebedarf oder Energiegewinnung aus den angeschlossenen Solarmodulen zu verbinden.

Um diesen Ansatz in der 3D-Webanwendung zu nutzen, müssen die folgenden Anforderungen erfüllt sein. Zunächst sollte ein vollständiges CityGML-Modell vorhanden sein, mit allen Objekten oder Gebäuden, damit alle Sensoren ihrem Standort im 3D-Stadtmodell zugeordnet werden können. Des Weiteren müssen die Sensoradministratoren in der Lage sein, auf Sensorinformationen im CityGML-Modell zuzugreifen und diese zu aktualisieren. Letztlich sind

Parse-Tools, die alle Dynamizer-Codierungstypen unterstützen, erforderlich, damit Benutzer die Sensordaten erhalten können.

2019 wurde die Erweiterung der 3DCityDB zur Unterstützung des Dynamizer-Objekts in CityGML-Modellen vorgestellt (Chaturvedi et al. 2019). Mit dieser Methode kann der Webclient über die Web Feature Services (WFS) eine Information über das Gebäude erhalten und das System benötigt weiterhin externe Dienste wie Sensor-Web-Enablement-(SWE)-Standards oder InterSensor-Service¹⁰ (Chaturvedi et al. 2019), um den Webzugriff auf die Sensordaten zu erhalten und zu ermöglichen.

Ein alternativer Ansatz für eine interoperable und standardisierte Integration von Sensordaten und 3D-Stadtmodell verlinkt die CityGML-Objekte mit externen OGC SWE-Standards. Dieser Ansatz erfordert keine Erweiterung des ursprünglichen CityGML, und mehrere Parteien oder Anbieter von Sensoren können die Sensordaten dynamisch verwalten oder aktualisieren, ohne das CityGML direkt bearbeiten zu

müssen. Zhu et al. (2016) und Kasprzyk et al. (2019) haben mit dem SOS eine standardisierte Schnittstelle für die Verwaltung und den Abruf von Metadaten und Beobachtungen aus heterogenen Sensorsystemen bereitgestellt (Bröring et al. 2012).

Das SOS ist jedoch möglicherweise nicht für Echtzeit-Smart-Cities-Anwendungen geeignet, da Clients mit großen XML-Dokumenten arbeiten müssen, die für ressourcenbeschränkte Geräte zeitaufwendig sind. Aus diesem Grund haben Jazayeri et al. (2015) und Santhanavani (2018) vorgeschlagen, dass STA die Echtzeit-Sensordaten entsprechend ihrer JSON-Codierung, Ladeperformance, HTTP-REST- und MQTT-Unterstützung entwicklerfreundlich verwaltet.

Es lässt sich feststellen, dass die STA alle als wichtig definierten Kriterien erfüllt. Die STA wurde jedoch nur verwendet, um das Sensorsystem in der absoluten Position zu lokalisieren und wurde nie für eine direkte Verbindung mit dem CityGML-Datensatz verwendet. In dieser Arbeit wird mit „City-Things“ ein Konzept vorgeschlagen, um

die STA zur Verwaltung der dynamischen Zeitserien-Sensordaten zu verwenden und mit CityGML-Objekten im STA zu verlinken.

3 GRUNDLAGEN

3.1 CITYGML

Der OGC CityGML-Standard (Gröger et al. 2012) dient sowohl als urbanes Informationsmodell als auch als Datenaustauschformat für 3D-Stadtmodelle. In CityGML können die geometrischen und semantischen Attribute der Objekte auf Stadtebene sowie deren Wechselbeziehungen zu anderen Objekten abgebildet werden. Dementsprechend gibt es weltweit eine Vielzahl an Städten, die den CityGML-Standard zur Darstellung ihrer 3D-Stadtmodelle verwenden (Gröger et al. 2012).

Für das 3D-Stadtmodell werden Komponenten des Gebäudes im CityGML-Standard mit der eindeutigen *gml:id* identifiziert. Der CityGML-Standard findet Anwendung in vielen Smart-Cities-Bereichen.

Biljecki et al. (2015) beschreiben in ihrem Übersichtsartikel verschiedenste Use Cases für 3D-Stadtmodelle und CityGML. Sie können beispielsweise genutzt werden, um Wärmebedarfe von Gebäuden zu simulieren (Weiler et al. 2018) oder auch um Katastrophenmanagement durchzuführen (Schulte & Coors 2009). Eine neue Version (3.0) des CityGML-Standards befindet sich gerade in Entwicklung.

3.2 SENSORTHINGS-API

Der STA-Standard wurde 2016 vom OGC verabschiedet, um Daten aus verschiedenen Sensorquellen mit der Anwendungsschicht zu verbinden. Der Standard folgt den REST-Prinzipien, der JSON-Codierung und den URL-Konventionen. Die Grundlage dieses Standards basiert auf dem Modell OGC Observations and Measurements und ermöglicht es IoT-Geräten (Internet of Things) und -Anwendungen, IoT-Daten und -Metadaten in einem SensorThings-Dienst zu erstellen, zu lesen, zu aktualisieren und zu löschen (Liang 2016a). Der Vorteil dieses Standards gegenüber anderen IoT-Standards besteht darin, dass er ein umfassendes konzeptionelles Modell und Abfragefunktionalitäten unterstützt (Huang & Wu 2016). Zusätzlich beinhaltet der Standard „Tasking“, es bietet eine Möglichkeit zur Parametrierung von aufgabefähigen IoT-Gerä-

```
{
  "value": [
    {
      "name": "Room Sensor Monitoring System 1",
      "description": "Sensor system monitoring Humidity, Temperature, and light level",
      "properties": {
        "Deployment_Date": "2017-05-24",
        "Produced_Date": "2017-01-24",
        "CaseColor": "Red",
        "AliasName": "System 1"
      }
    }
  ]
}
```

Auflistung 1: Ein Beispiel für JSON, das eine Thing-Einheit mit mehreren Eigenschaften zeigt

ten, z. B. die Änderung der LED-Farbe der Sensoren entsprechend dem spezifizierten Sensorphänomen. Das bedeutet auch, dass die OGC STA erweiterbar ist und nicht nur auf einfache, sondern auch auf komplexe Anwendungsfälle angewendet werden kann (Liang 2016b). Trilles et al. (2015) und Soto et al. (2016) zeigen die Verwendung der STA, um interoperable Dienste anzubieten, die den Zugriff auf Daten in ihrer Anwendungsplattform erleichtern. Santhanavanich et al. (2018) hatten diesen Standard erfolgreich eingesetzt, um ein Sensordatenmanagementsystem zum Tracking von E-Bikes und Vitaldaten des Benutzers aufzubauen. Die genannten Arbeiten belegen, dass die STA ein effektives und effizientes Werkzeug zur Integration unregelmäßiger Zeitseriendaten von verschiedenen Sensortypen ist. Darüber hinaus zeigten Jazayeri et al. (2015) und Santhanavanich (2018), dass die STA in Bezug auf die Datenübertragung sehr effizient ist und somit für die Echtzeitanwendung besser geeignet ist als andere bestehende IoT-Standards.

Die STA sammelt die Informationen eines IoT-Geräts oder einer Sensorvorrichtung in verschiedenen Entitätstypen. Diese Entitätstypen sind *Thing*, *Sensor*, *Datastream*, *Location*, *HistoricalLocation*, *ObservedProperty*, *Observation* und *FeatureOfInterest*, wie in Abbildung 1 dargestellt. Das *Thing* definiert ein physikalisches Objekt oder Sensorsystem in der realen Welt. Eine *Location* repräsentiert die Position eines *Things*. Jedes *Thing* hat einen oder mehrere *Datastreams*. Jeder *Datastream* hat eine *ObservedProperty* und einen *Sensor*. Des Weiteren hat jede *Observation* einen *Datastream* mit einem spezifizierten *FeatureOfInterest* und wird verwendet, um die Echtzeit- oder historischen gemessenen Zeitseriendaten zu sammeln (Liang 2016a).

Im STA-Konzept werden die Eigenschaften der Sensoren oder physikalischen Objekte gesammelt. Diese werden im Properties-Objekt in der *Thing*-Entität abgelegt. Gemäß der Standarddefinition werden alle Daten im JSON-Format gepflegt. Dies gibt dem Entwickler die Flexibilität, mehrere Wertepaare oder Arrays hinzuzufügen und zu speichern. Beispielsweise zeigt Auflistung 1 die Entität *Things* des Raumsensor-Überwachungssystems, welche mehrere Properties enthält. Diese Properties dienen beispielsweise der Bestimmung des Sensoralters (*Produced_Date*) für Routine-Wartungsarbeiten. Die Property „CaseColor“ wird in diesem Beispiel genutzt, um zwischen mehreren Sensoren zu unterscheiden, falls diese sich im gleichen Gebäude oder im gleichen Gebiet befinden.

Im STA-Konzept werden die Eigenschaften der Sensoren oder physikalischen Objekte gesammelt. Diese werden im Properties-Objekt in der *Thing*-Entität abgelegt. Gemäß der Standarddefinition werden alle Daten im JSON-Format gepflegt. Dies gibt dem Entwickler die Flexibilität, mehrere Wertepaare oder Arrays hinzuzufügen und zu speichern. Beispielsweise zeigt Auflistung 1 die Entität *Things* des Raumsensor-Überwachungssystems, welche mehrere Properties enthält. Diese Properties dienen beispielsweise der Bestimmung des Sensoralters (*Produced_Date*) für Routine-Wartungsarbeiten. Die Property „CaseColor“ wird in diesem Beispiel genutzt, um zwischen mehreren Sensoren zu unterscheiden, falls diese sich im gleichen Gebäude oder im gleichen Gebiet befinden.

4 KONZEPT

4.1 CITYTHINGS

Das CityThings-Konzept bietet eine einfache Möglichkeit, Sensordatenströme mit dem 3D-Stadtmodell, basierend auf der STA und dem CityGML-Standard, zu verwalten.

In dem hier vorgeschlagenen CityThings-Konzept wird ein Sensor im SensorThings-Standard über eine eindeutige Identifikationsnummer (ID) mit dem 3D-Stadtmodell verknüpft.

tifikation mit einem CityObjekt verknüpft. Dies wird erreicht, indem die *gml-id* aus CityGML als eine der Eigenschaften in der *Thing*-Entität der STA gespeichert wird.

Anwendungsentwickler können die Sensordaten aus der STA mit der *gml-id* als Schlüssel abrufen, um zusätzliche Gebäudeinformationen aus dem CityGML-Modell zu erhalten. Entwickler können die STA auch verwenden, um eine Liste von Gebäudemodellen von einem bestimmten Sensorsystem abzufragen und umgekehrt. Es wird empfohlen, dieses Konzept mit einem CityGML-Modell zu verwenden, welches in einem Datenbanksystem gespeichert ist. Um dies zu realisieren, bietet die 3DCityDB (Yao et al. 2018) eine Methode, um ein CityGML-Modell auf einfache Weise in die PostgreSQL- oder Oracle-Datenbank zu importieren.

Das CityThings-Konzept unterstützt auch die Positionsänderung des Sensors, die auf der STA in der Entität *Thing* und *Location* durchgeführt wird. Wenn beispielsweise das Sensorsystem bewegt wird, wird der neue Standort in der Entität *Location* aktualisiert. Benutzer können ihren Pfad später in der Entität *HistoricalLocation* verfolgen.

Wenn beispielsweise das Sensorsystem in einem bestimmten Gebäudeteil im CityGML-Modell ist, wird die *gml-id* des Gebäudeteils in den Eigenschaften des *Things* in der STA gespeichert. Später wird die *gml-id* des CityGML-Gebäudes auf jedes *Thing* in der STA in der Anwendungsebene angepasst. Dies ermöglicht es dem Benutzer, Informationen über den Sensor

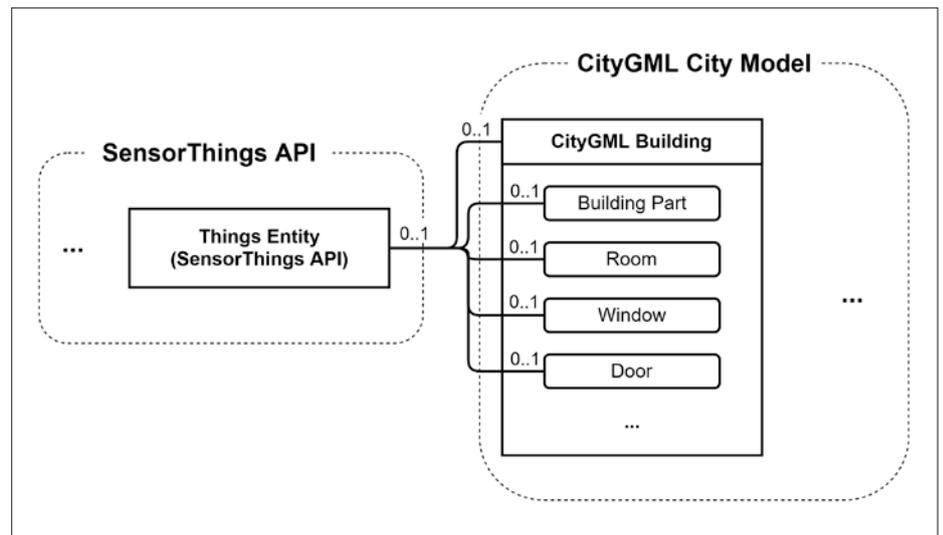


Abbildung 2: Ein Diagramm, das ein vereinfachtes Beispiel für die Verbindung zwischen der STA und dem CityGML-Stadtmodell zeigt (Santhanavanich & Coors 2019)

im CityGML-Modell auf der entwickelten Anwendung abzurufen.

Abbildung 2 veranschaulicht beispielsweise die Verbindung von der *Thing*-Entität in der STA zu verschiedenen Komponenten des Gebäudes im CityGML-Modell. Es ist auch möglich, dieses Konzept auf andere Teile des Stadtmodells wie Straßen, Vegetation und Gewässer anzuwenden. Bei diesem Konzept werden die Standards CityGML und STA zusammen verwendet, ohne dass eine Erweiterung erforderlich ist.

4.2 ERWEITERUNG SENSORTHINGS-API

CityThings bietet dem Anwender den Zugriff auf den Sensordatensatz über die STA als Zeitreihe im JSON-Format, die in

vielen Anwendungsbereichen eingesetzt werden kann. Der SensorThings-Standard enthält jedoch keine Aggregationsfunktion.

Insbesondere bei großen unregelmäßigen Zeitreihen besteht die Notwendigkeit, die bereinigten aggregierten Zeitreihen im angegebenen Zeitbereich anzufordern. Dementsprechend hat die HFT ein Tool „SensorThings Aggregator“ entwickelt, um aggregierte Sensordaten von der STA serverseitig zu verarbeiten und zu liefern (Santhanavanich 2019a). Dieses Tool baut auf der STA auf, indem die STA-Beobachtungs-URL, der Aggregationstyp und das Zeitintervall benötigt werden. Dieses Tool ermöglicht mehrere aggregierte Methoden, einschließlich *Mittelwert*, *Summe*,

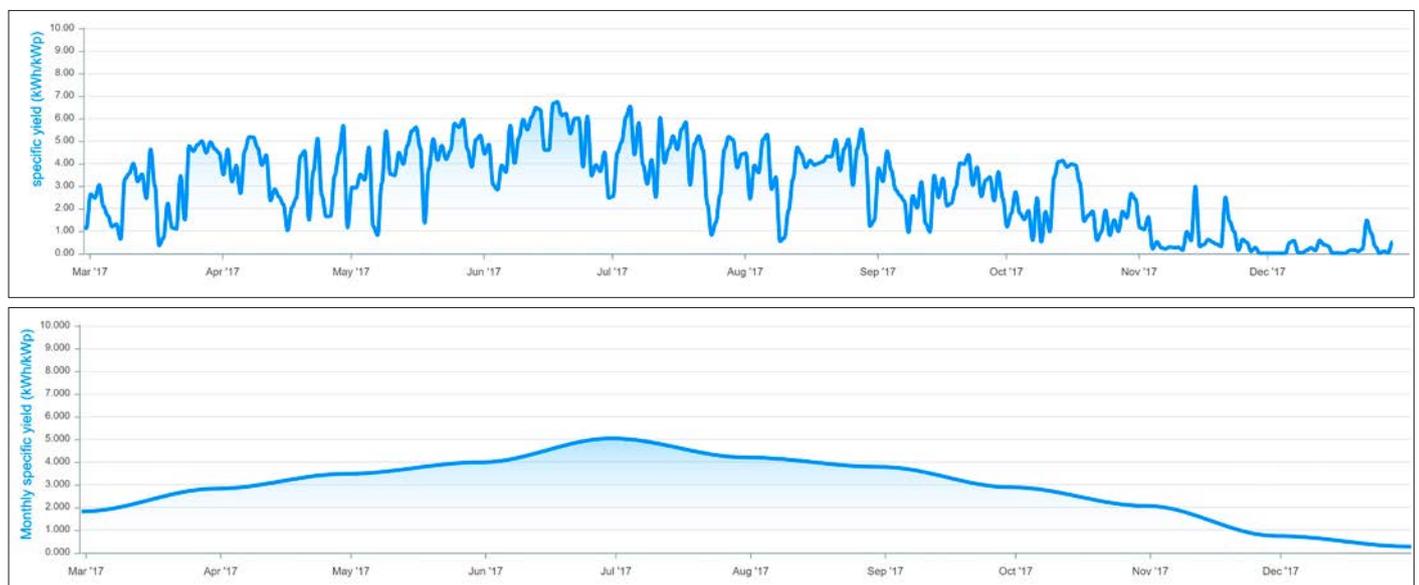


Abbildung 3: Aggregierte Zeitreihendaten

Min, Max, First und Last. Mit diesem Tool profitieren die Anwender davon, dass sie die verarbeiteten Zeitreihendaten im gewünschten Zeitintervall erhalten und so die Gesamtdatenmenge sowie die benötigte Bandbreite des Netzwerks reduzieren.

So zeigt beispielsweise das Liniendiagramm aus Abbildung 3 oben die Daten des spezifischen Ertrags eines Solarmoduls im Raum Wüstenrot im Tagesabstand. In der Anwendungsebene wird dieser Datensatz vom SensorThings-Server an die Benutzer-Clients übertragen, deren Leistung von der Größe der Datentransaktion maßgeblich abhängt. Aus diesem Grund kann die Datengröße stark reduziert werden, indem das aggregierte Ergebnis in einem monatlichen Intervall erzeugt und übertragen wird (Abbildung 3 unten). In vielen Anwendungen ist es nicht notwendig, alle Rohwerte zu verwenden und zu visualisieren. In diesem Beispiel geben die monatlich aggregierten Daten den Nutzern auch einen Einblick in die spezifische Rendite im Jahresverlauf. Dies zeigt, dass der SensorThings-Aggregator bei der Anwendungsentwicklung signifikant hilft, die Ausführung von Visualisierungs- und Analyseprozessen zu beschleunigen, indem er die übertragene Datenmenge reduziert und die Komplexität der Berechnungen verringert.

5 IMPLEMENTIERUNG

In diesem Abschnitt wird die Umsetzung des CityThings-Konzepts erläutert. Um dieses Konzept auf den realen Anwendungs-

fall zu übertragen, müssen das physische Sensorsystem und das 3D-Stadtmodell serverseitig nach dem in Abschnitt 5.1 erläuterten OGC-Standard vorbereitet werden.

In Abschnitt 5.2 wird die Verbindung zwischen Sensoren und 3D-Stadtmodellen erläutert.

In Abschnitt 5.3 wird dargestellt, wie Sensor- und Stadtmodell Daten aus CityThings abgerufen werden können. Abbildung 4 zeigt eine Systemarchitektur des CityThings-Konzepts.

5.1 SERVERSEITIGE VORBEREITUNG

Das Sensordatenmanagement erfolgt mit einer PostGIS-Datenbank und der STA-Implementierung des Fraunhofer Open Source SensorThings (FROST). Nachdem der Sensorserver eingerichtet ist, müssen alle Sensorsysteme in allen erforderlichen Einheiten, wie in Kapitel 3 beschrieben, auf dem SensorThings-Server registriert sein. Im Fall des Solarmoduls auf einem bestimmten Gebäude muss der Benutzer beispielsweise den Namen und die Metadaten jedes Sensors im System in der Sensorentität eintragen. Die allgemeinen Informationen des Systems und erweiterte Informationen, wie ein Link zum CityGML-Modell, müssen bei der Thing-Entität registriert werden. Anschließend müssen die Standortkoordinaten dieses Systems in der Location-Entität und jede beobachtete Eigenschaft, wie die Energieerzeugung, in der ObservedProperty-Entität eingetragen werden. In diesem

Fall ist die Entität *FeatureOfInterest* identisch mit der Entität *Location*.

Anschließend werden die Messdaten der Sensorsysteme an den SensorThings-Server übertragen. Dazu muss ein Programmierskript auf den Sensorsystemen oder Zwischenservern mit folgenden drei Funktionalitäten erstellt werden.

Erstens werden die Sensordaten im JSON-Format nach dem SensorThings-Standard in der spezifischen Struktur verarbeitet. Zweitens werden die Messdaten von jedem Sensorsystem mit der richtigen *Datastream-ID* abgeglichen. Drittens werden die vorverarbeiteten Daten über die HTTP- oder MQTT-Verbindung an die STA übertragen.

Das „SensorThings Importer“ genannte Tool dient dem Importieren von Daten aus historischen Zeitreihen in verschiedenen Formaten wie CSV, JSON und GPX in die STA auf einfache Weise (Santhanavanich 2019b). Die Software wird im Rahmen des Smart-Villages-Projekts als Open Source bereitgestellt. Zusammenfassend werden die Zeitreihen-Sensordaten vom SensorThings-Server und unabhängig von der 3D-Stadtmodell Datenbank gehandhabt. Die Anbindung des Sensorsystems an das Stadtmodell wird in Abschnitt 5.2 erläutert.

5.2 ANSCHLUSS VON SENSOR-THINGS UND CITYGML

Dieser Abschnitt zeigt ein Verfahren zum Anschluss der Sensorsysteme der STA zum CityGML-basierten 3D-Stadtmodell. Da das CityGML-Stadtmodell auf dem

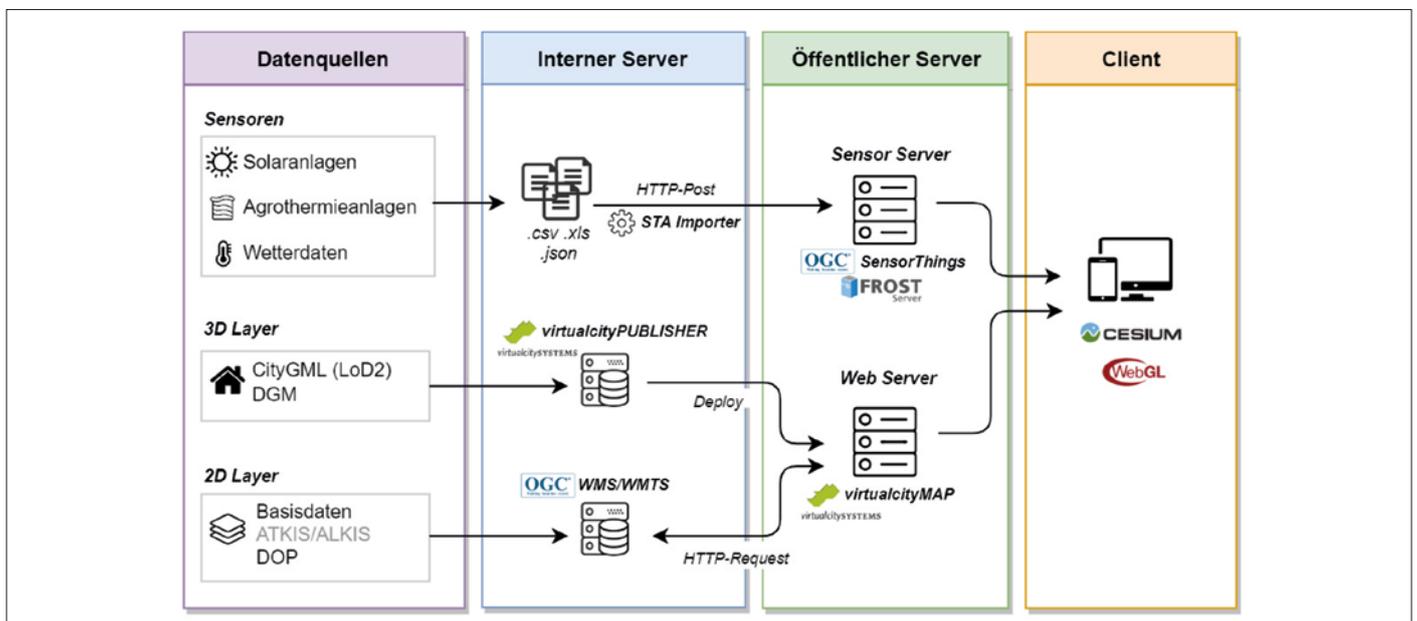


Abbildung 4: Systemarchitektur des CityThings-Konzepts

```

1: <core:CityModel ...>
2:   <gml:name>Sample-GML CityThings</gml:name>
3:   <gml:boundedBy>...</gml:boundedBy>
4:   <core:cityObjectMember>
5:     <bldg:Building gml:id="DENW24ALG00003YF">
6:       <gml:description>Sample-Building</gml:description>
7:       <gml:name>Building-a</gml:name>
8:       <core:creationDate>2019-02-01</core:creationDate>
9:       ...
10:      <bldg:boundedBy>
11:        <bldg:RoofSurface gml:id="GML_50155-aa175">
12:          <gml:name>RoofSurface South West</gml:name>
13:          ...
14:        </bldg:RoofSurface>
15:      </bldg:boundedBy>
19:    ...
20:  </bldg:Building>
21: </core:cityObjectMember>
22: </core:CityModel>

```

Auflistung 2: Ein Beispiel für ein Gebäude im CityGML-Format

```

1: {
2:   "value": [{
3:     "name": "System HFT-566",
4:     "description": "PV Sensor System",
5:     "properties": {
6:       "property a": "xxx",
7:       "property b": "yyy",
8:       "citythings": {
9:         "gml_id": "DENW24ALG00003YF"
10:      }
11:     }
12:   }]
13: }

```

Auflistung 3: Ein Beispiel JSON, das die Entität einer Sache in der STA zeigt, die mit dem CityGML-Gebäude in Auflistung 2 verbunden ist

GML-Format basiert, enthält jedes Element des Stadtmodells die *gml-id*. Dieses Attribut kann zusammen mit den Sensorinformationen auf die Eigenschaft der *Thing*-Entität in der STA registriert werden. Dazu wird ein Array namens *Citythings* in den *Thing*-Eigenschaften erzeugt. Dieses Array enthält das Schlüssel-Wert-Paar der *gml-id* des *CityObjects* aus dem Stadtmodell. Um dies zu demonstrieren, wird in Auflistung 2 ein Beispiel für ein Gebäude im Stadtmodell im CityGML-Format gezeigt. Die angegebenen Gebäude *gml-id* (Auflistung 2: Zeile 5) ist "DENW24ALG000000003YF". Außerdem enthält es die Dachfläche mit

der angegebenen *gml-id* (Auflistung 2: Zeile 11) "GML_50155-aa175".

Wenn ein neues Sensorsystem im Gebäude aus der Auflistung 2 installiert wird und kein dazugehöriges Gebäudeteil spezifiziert ist, wird die Gebäude-GML-ID "DENW24ALG000000003YF" verwendet, um den Sensor in der STA zu registrieren (Auflistung 3: Zeile 9). In einem anderen Fall, wenn der Sensor eine spezifizierte Position auf der Dachseite Richtung Südwesten hat, muss die *gml-id* "GML_50155-aa175" registriert werden, da die Details auf der äußeren Ebene der CityGML-Hierarchie liegend später direkt im CityGML-Modell

oder in der 3DCityDB abgerufen werden können.

5.3 ABRUFEN VON SENSOR- UND 3D-STADTMODELLDATEN

Die Sensordaten können im 3D-Stadtmodell komfortabel mit der HTTP-GET-Anfrage an die SensorThings-Schnittstelle abgerufen werden. Mit dem Request-URI unterstützt STA Abfrageoptionen, die es Benutzern ermöglichen, eine bestimmte Sammlung von Entitäten zu filtern, einschließlich Gebäudeinformationen und Sensorinformationen wie in Tabelle 1. Das Beispiel a) (Tabelle 1) zeigt eine einfache Anfrage, um eine Liste von Sensorsystemen zu erhalten, die in einem bestimmten Gebäude verortet sind. In Beispiel b) (Tabelle 1) wird eine komplexere Abfrage zum Durchsuchen eines Sets von *Datastreams* dargestellt, der eine bestimmte *ObservedProperty* misst und in einem bestimmten Gebäude ausgestattet ist. Das Ergebnis aus dem Beispiel b) ist in der Auflistung 4 dargestellt. Bei der Abfrageoption *expand* zeigt die Antwort den Benutzern die Stadtmodell *gml-id* (Auflistung 4: Zeile 7) und das Ergebnis der *Observation*-Zeitreihe (Auflistung 4: Zeile 13:29) direkt aus einer einzigen Anfrage.

Auf diese Weise können die Beobachtungsdaten vom Sensorserver basierend auf einer *gml-id* aus CityGML auf einfache Weise abgerufen werden. Umgekehrt können die Informationen des Stadtmodells auch aus der SensorThings-Antwort der *gml-id* aus den Eigenschaft des *Things* bezogen werden. Das CityThings-Konzept unterstützt auch die Anfrage nach den Sensordaten basierend auf den Attributen des Stadtmodells. Dazu kann der Benutzer die *gml-id* der Gebäude mit den angegebenen Attributen aus der Stadtmodellatenbank einfach suchen und abfragen. Anschließend kann die resultierende *gml-id* verwendet werden, um die Sensordaten von der STA abzurufen.

5.4 WEBBASIERTER 3D-VISUALISIERUNG

Um die Inhalte des 3D-Stadtmodells einfach verfügbar zu machen, spielt die webbasierte 3D-Visualisierung eine wichtige Rolle, welche immer relevanter wird. Um dies zu unterstützen, gibt es einige Open-Source-Anwendungsprogrammierschnittstellen (APIs) wie CESIUM und NASA WorldWind. Sie erlauben das Rendern und Bereitstellen der 3D-Inhalte auf einem

| Beispiel | Request URI | Beschreibung |
|----------|---|---|
| a) | <code>http://<STA-Base>/Things? \$filter=Things/properties/citythings/gml_id eq 'xx1'</code> | Abfrage aller physikalischen Systeme (Things) vom Gebäude mit der <i>gml-id</i> 'xx1' |
| b) | <code>http://<STA-Base>/Datastreams? \$expand=Thing& \$expand=ObservedProperty& \$filter=Things/properties/ citythings/gml_id eq 'xx1'& \$filter=ObservedProperty/name eq 'electrical-power'& \$filter=Datastream/unitOfMeasurement eq 'kWp'</code> | Anforderung aller Datenströme des Sensorsystems, das auf das Gebäude <i>gml-id xx1</i> registriert ist und die Stromertrag in kWp misst. |

Tabelle 1: Beispiele für HTTP-GET-Anfragen zum Abrufen der Sensor- und der 3D-Stadtmodelldaten

virtuellen 3D-Globus durch das Web. Es gibt jedoch keinen direkten Weg der Visualisierung des CityGML-basierten Modells auf diesen Web-Globus-Framework aus der Datenbankebene. Nach Angaben des OGC wurde neben dem I3S auch das 3D-Tiles-Format als Standard eingeführt welches eine Geodatenstruktur definiert und für die Optimierung von 3D-Streaming und -Rendering entworfen wurde (Cozzi 2019).

In der Entwicklung dieses Projekts wurde eine Gesamtlösung genutzt. Der virtualcityPUBLISHER beinhaltet einen CESIUM-Viewer als Visualisierungskomponente. Die Datenbankkomponente ist eine 3DCityDB und zusätzlich beinhaltet sie einen Konverter von CityGML nach 3D-Tiles. Diese Softwarelösung wurde der HFT unter einer akademischen Lizenz von virtualcitySYSTEMS zur Verfügung gestellt.

6 EVALUIERUNG

6.1 STAND DER GEMEINDE WÜSTENROT

Das hier vorgestellte Konzept wird anhand einer Anwendung der Gemeinde Wüstenrot evaluiert. Die Gemeinde hat zur Strom- und Wärmeerzeugung verschiedene Energiegewinnungsanlagen errichtet, für die von den Systembetreibern bereits Sensordaten zur Verfügung gestellt werden. Weitere Energiegewinnungsanlagen, wie zur Produktion von Energie aus Biomasseanlagen, befinden sich im Aufbau. Der Testlauf der Vernetzung von Sensordaten mit der 3D-Webplattform am Beispiel der Gemeinde Wüstenrot hat gezeigt, dass die in Ka-

```

1: {
2:   "name": "Room Temperature Datastream",
3:   "...": "...",
4:   "Thing": {
5:     "...": "...",
6:     "citythings": {
7:       "gml_id": "DENW24ALG00003YF"
8:     }
9:   },
10:  "unitOfMeasurement": {
11:    "...": "...",
12:  },
13:  "Observations": [
14:    {
15:      "phenomenonTime": "2019-01-01T02:00:00.000Z",
16:      "resultTime": "2018-04-01T06:46:37.507Z",
17:      "result": 18,
18:      "@iot.id": 49265986
19:    },
20:    {
21:      "phenomenonTime": "2019-01-01T01:00:00.000Z",
22:      "resultTime": "2018-04-01T06:46:42.508Z",
23:      "result": 17.5,
24:      "@iot.id": 49266986
25:    },
26:    {
27:      "...": "...",
28:    }
29:  ]
30: }

```

Auflistung 4: Ein Beispiel JSON, das ein Anfrageergebnis von Beispielanforderung b) aus Tabelle 1 zeigt

pitel 4 beschriebene Vorgehensweise umsetzbar und auch replizierbar ist. Dies kann derzeit an den Beispielen der Photovoltaikanlagen auf öffentlichen Gebäuden (PV) sowie der Agrothermieanlage präsentiert werden. Die Verknüpfung zwischen Gebäude und Sensoren erfolgt über die *gml-id* der Gebäudedaten.



Abbildung 5: Fachspezifische Daten wie umweltrelevante Sensordaten, z. B. der Stromertrag der PV-Anlagen, werden zum Monitoring in die Plattform integriert und können für weitere Simulationen genutzt werden

Der Energieerlebnispfad besteht aus zehn beschilderten Stationen, die u. a. nachhaltige Energiegewinnungsverfahren darstellen und erlebbar machen sollen. Vor Ort soll das über eine App mit Augmented-Reality-Technologie erfolgen, im Rahmen dieses Teilprojekts als Story Map über das Internet. Die unterirdisch verbaute Agrothermieanlage soll dadurch sichtbar gemacht werden. Mit dem Energieerlebnispfad können die Bürgerinnen und Bürger multimedial dafür sensibilisiert werden, wie und wie viel Strom erzeugt und wie viel benötigt wird, um beispielsweise ein E-Bike zu laden oder eine Glühbirne zum Leuchten zu bringen bzw. wie viel Heizwärme benötigt wird, um ein Wohnzimmer zu heizen.

Stand der Visualisierung von Sensordaten aus PV-Anlagen in der 3D-Webanwendung

Energieflusserträge historischer Ertragsdaten von PV-Anlagen auf öffentlichen Gebäuden können für vorgegebene Zeitabschnitte über einen Sensordatenaustausch im 3D-Modell von CESIUM dargestellt werden (Sini 2019). Das Ergebnis, ergänzt durch Informationen zu Gebäude, PV-Anlage und zu den daraus resultierenden Werten der Energieproduktion sowie Witterungsbedingungen, wird in die 3D-Webplattform des

LGL übertragen (vgl. Abbildung 5). Hierbei handelt es sich um die Implementierung von historischen Messdaten und noch nicht um einen Echtzeitzugriff.

Stand der Visualisierung von Messwerten einer Agrothermieanlage in der 3D-Webanwendung

Oberflächennahe Geothermie durch einen Agrothermiekollektor dient der Plusenergiesiedlung „Vordere Viehweide“ als Wärmequelle, die gleichzeitig eine Oberflächennutzung als Futterwiese ermöglicht (Peschet al. 2018). Die Anbindung des Flächenkollektors an dezentrale Wärmepumpen mit Pufferspeichern, die den Wärmebedarf der Liegenschaften decken, erfolgt über ein kaltes Nahwärmenetz (KNW) ohne zentrale Umwälzpumpe. Selbst bei niedriger Lufttemperatur kann dem Boden Energie entzogen werden und für die Haushalte über Wärmepumpen nutzbar gemacht werden, da die Bodentemperatur relativ geringen Schwankungen unterliegt. Im Sommer kann hingegen die Wärme aus den Häusern in den Boden abgeführt werden.

Für die Bürgerinnen und Bürger sind die Agrothermieanlage und damit die Wärme-gewinnung bis auf ein kleines Trafohäuschen weitestgehend unsichtbar. Mithilfe der Messwertvisualisierung in der 3D-Darstellung können die unterirdisch verlaufen-

de Anlage und deren Anbindung an die Liegenschaften der Plusenergiesiedlung „Vordere Viehweide“ sichtbar gemacht sowie die Boden- und Außentemperatur der Agrothermieanlage zur Darstellung der Temperaturdifferenz in der 3D-Webanwendung visualisiert werden, um Rückschlüsse auf Grenzwerte der Wärme der thermisch-aktivierten Fläche ziehen zu können (vgl. Abbildung 6). Hier sind ein Echtzeitdatenzugriff sowie ein Zugriff auf historische Messdaten möglich.

6.2 EVALUIERUNG DES CITYTHINGS-KONZEPTS

Anhand des CityThings-Konzepts wird die Vernetzung von Sensordaten und dem Geobasismodell nach OGC-Standards ermöglicht. Hierzu werden zum einen für das Sensordatenmanagement die Einrichtung des STA-Servers sowie die Aufbereitung der Messdaten der Sensorsysteme mit dem Open-Source-Tool „SensorThings Importer“ zum Importieren historischer Zeitreihen aus verschiedenen Formaten benötigt. Zum anderen ist die webbasierte 3D-Visualisierung notwendig, um über Open-Source-APIs, hier CESIUM, auf einem virtuellen Globus zu Rendern und 3D-Inhalte bereitzustellen. CityGML wird hierbei in 3D-Tiles als OGC-Standard-Format konvertiert und in CESIUM bereitgestellt.



Abbildung 6: Echtzeitdaten der Bodentemperatur sowie Außentemperatur der Agrothermieanlage

Über die *gml-id* wird die *Thing*-Entität in STA vernetzt, beispielsweise wie in Kapitel 5.1 anhand des Gebäudedachs und einer PV-Anlage. Es besteht die Möglichkeit, mit einer HTTP-GET-Anfrage an den STA-Server-Sensordaten über die *gml-id* angegebene Attribute in der 3D-Webanwendung abzurufen. Der Zugriff anhand CityThings über STA auf einen Sensordatensatz als Zeitreihe in JSON-Format erfolgt jedoch ohne Zeitreihenaggregation.

Die HFT hat das Tool „SensorThings Aggregator“, das auf STA aufbaut, zur Aggregation unsortierter Datenmengen über unterschiedliche Methoden entwickelt, um die Abfrage eines gewünschten Zeitintervalls zu ermöglichen. Die Vorteile dieses Tools sind eine beschleunigte Ausführung von Visualisierungs- und Analyseprozessen durch Reduktion der übertragenen Datenmenge sowie eine Vereinfachung der Komplexität der Berechnungen.

Die Anbindung der Schnittstellen, welche direkt in CESIUM erfolgt ist, kann auf virtualcityPUBLISHER und Wega3D Publisher übertragen werden.

6.3 SCHLUSSFOLGERUNGEN

Wie bereits in den vorherigen Kapiteln erläutert, wurde das grundlegende Ziel, Sensordaten mit einem 3D-Stadtmodell über die offene Standardschnittstelle STA zu vernetzen, erreicht. Dies wurde an einer prototypischen Anwendung der Gemeinde Wüstenrot gezeigt.

Generell können alle Arten von Sensortypen implementiert werden. Daten sollten direkt von den Erzeugern bereitgestellt werden, um Redundanzen zu vermeiden. Zur Vereinheitlichung werden derzeit Daten aus unterschiedlichen Formaten zunächst über den „SensorThings Importer“ in das JSON-Format konvertiert. Ein wesentlicher Vorteil der Vernetzung von Echtzeitdaten mit der 3D-Webplattform liegt darin, dass die aktuellen Energieflüsse bestehender Anlagen jederzeit abgerufen werden können. Auf die Echtzeitdaten der PV-Anlagen von öffentlichen Gebäuden ist derzeit noch kein Zugriff möglich, weshalb die Entwicklung hier noch nicht abgeschlossen ist. Dies ist nicht von der technischen Lösung abhängig, sondern von der Datenbereitstellung durch den Anbieter. Die Gemeinde gibt vor, welche Daten in welcher Form in der 3D-Webplattform visualisiert werden sollen. Prinzipiell ist die zukünftige Digitalisierung und damit die zuverlässige Nutzung von webbasierten Anwendungen im ländlichen Raum grundlegend abhängig von schnellen Internetverbindungen.

Zu berücksichtigen ist, dass sich im Zuge der „Wüstenroter Energiewende“ noch weitere Energiegewinnungsanlagen in unterschiedlichen Baustadien befinden. Da diese nicht immer mit dem Zeitplan der technischen Umsetzungen im Teilprojekt korrelieren, müssen Vorhaben gegebenenfalls angepasst werden. Verschiedene Sensortypen weiterer Energiegewinnungsan-

gen sollen an die STA angehängt werden können. Es ist vorgesehen, die Produktions- und die Verbrauchsdaten über die 3D-Webplattform darzustellen. In welchem Maße dies realisierbar ist, hängt vom Bedarf der Gemeinde sowie von datenschutzrechtlichen Grundsatzfragen ab.

Die Gemeindeverwaltung selbst sieht den Mehrwert der Integration der Sensordatenauswertung zunächst in der Datenbereitstellung und schließlich in der Möglichkeit zur Weiterverarbeitung der Daten, insbesondere für das kommunale Energiemanagement zur Überwachung des Energie- und damit des Sanierungsbedarfs. Überdies stellt die 3D-Webanwendung ein wichtiges Präsentationstool in politischen Gremien und für die Öffentlichkeitsarbeit dar. Ein weiteres Anliegen ist die mögliche Anwendung in der Georg-Kropp-Schule, um das Thema „Digitalisierung“ im Unterrichtsstoff einzubringen, wobei die Schule mit ihrem Arealnetz als Beispiel visualisiert werden soll. Die Evaluierung der Sensorschnittstellen soll im Energieerlebnispfad erfolgen, um die Ergebnisse auch dem Bürger zugänglich zu machen. Hierfür soll eine Darstellung des Energieerlebnispfades in der 3D-Webanwendung als Story Map mit den verschiedenen Stationen der Energiegewinnungsanlagen erfolgen.

7 FAZIT UND AUSBLICK

Das LGL bietet eine 3D-Webplattform mit amtlichen Daten als Grundlage für Pla-

nungsvorhaben und zur Datenvernetzung, um die Gemeinde zu unterstützen. Die HFT ermöglicht die Implementierung der Anwendungen von Sensordaten durch externe Anbieter über die offene Schnittstelle STA in ebendiese 3D-Webplattform. Das Teilprojekt der „Vernetzung von Sensordaten“ befindet sich noch am Anfang, daher wurden zu Testzwecken derzeit verfügbare Daten zur Vernetzung genutzt.

Hier wurden zum einen am Beispiel von Photovoltaikanlagen mit Stromproduktion und zum anderen am Beispiel einer Agrothermieanlage mit Wärmeproduktion die Vernetzung der aktuell verfügbaren Sensordaten mit den 3D-Geodaten der 3D-Webplattform inklusive der technischen Umsetzung aufgezeigt. Weitere Inhalte und Lösungsmöglichkeiten können mithilfe der Gemeinde realisiert werden.

Derzeit steht das LGL im Austausch mit der Gemeinde, damit deren Bedarf gemeinsam mit den Projektpartnern erörtert und entsprechend deren Wünschen in der 3D-Webanwendung abgebildet wird. Dies soll am Beispiel der Visualisierung der Energieproduktion (Strom und Wärme) sowie des Haushaltsverbrauchs am Beispiel eines öffentlichen Gebäudes, der Schule, realisiert werden.

In Zukunft sollen die Daten für Simulationen nutzbar gemacht werden, zum Beispiel bei der Planung weiterer PV- oder Agrothermie-Anlagen an geeigneten Standorten unter Einbeziehung des Ertrags bereits existierender Anlagen sowie unter Berücksichtigung der besonderen Gegebenheiten vor Ort wie Topographie und Mikroklima. Ziel ist es überdies, Information und Transparenz für Bürgerinnen und Bürger zu bieten. Mithilfe der virtuellen Begehung des Energieerlebnispfads können sie für einen bewussten Umgang mit den natürlichen Ressourcen sensibilisiert werden.

Derzeit handelt es sich um ein in sich geschlossenes System. Für die Zukunft soll die Basis dafür geschaffen werden, welches Anwender (Gemeindeverwaltung, Dienstleister) durch die eigenständige Registrierung von Sensordaten Energiequellen ergänzen können, beispielsweise aus einem Biomassewärmernetz mit dezentraler Solarwärme zur Strom- und Wärmeproduktion.

Von Vorteil wäre die Nutzung einer offenen standardisierten Schnittstelle, wie dem

3D Portrayal Service, um die unterschiedlichsten firmenspezifischen Ausgabeschnittstellen zu bündeln. Dadurch könnten die Sensordaten mit den 3D-Geodaten vernetzt und herstellerunabhängig in die 3D-Webanwendung integriert werden. Die verallgemeinerbaren Ergebnisse des Teilprojekts Wüstenrot sollen letztendlich für andere Gemeinden nutzbar gemacht werden.

DANKSAGUNG

Das in diesem Beitrag zugrunde liegende Vorhaben „Smart Villages – attraktive Orte im ländlichen Raum“ wird mit Mitteln der Landesregierung Baden-Württemberg im Rahmen der Initiative digital@bw gefördert und erfolgt in Zusammenarbeit mit der Hochschule für Technik (HFT) Stuttgart. In diesem Zusammenhang seien dem Ministerium für Ländlichen Raum und Verbraucherschutz (MLR) sowie den Partnerkommunen, der Stadt Niedernhall sowie der Gemeinde Wüstenrot, gedankt.

¹ <https://www.digital-bw.de/-/smarter-geoinformation-geo-goes-digital-bw>

² <https://cesium.com/cesiumjs>

³ <https://www.opengeospatial.org/>

⁴ <https://www.opengeospatial.org/standards/3DTiles>

⁵ <https://www.opengeospatial.org/standards/sensorthings>

⁶ <https://www.iosb.fraunhofer.de/servlet/is/80113/>

⁷ <http://www.emw-energie.de>

⁸ <https://www.opengeospatial.org/standards/citygml>

⁹ Domain-Range und Time-Value Pair Encoding sind Codierungen zum Speichern und Austauschen von Zeitseriendatensätzen (Tomkins & Lowe 2016).

¹⁰ Der InterSensor-Service ist eine Open-Source-Java-Anwendung, die auf dem Spring-Framework basiert. Es ermöglicht die Interoperabilität von heterogenen Sensor- und IoT-Plattformen und anderen Zeitreihendaten auf standardisierte Weise (<http://www.intersensorservice.org/>).

Literatur

- Biljecki, F.; Stoter, J.; Ledoux, H.; Zlatanova, S.; Çöltekin, A. (2015): Applications of 3D city models: state of the art review. In: ISPRS International Journal of Geo-Information, 4 (4), S. 2842-2889.
- Bröring, A.; Stasch, C.; Echterhoff, J. (2012): OGC Sensor Observation Service Interface Standard. Version 2.0.0, OGC 12-006. <http://www.opengis.net/doc/IS/SOS/2.0>.
- Chaturvedi, K.; Kolbe, T. H. (2015): Dynamizers – Modeling and Implementing Dynamic Properties for Semantic 3D City Models. In: Biljecki, F.; Tourre, V. (Eds.): Eurographics Workshop on Urban Data Modelling and Visualisation: The Eurographics Association. DOI:10.2312/udmv.20151348.
- Chaturvedi, K.; Kolbe, T. H. (2017): Future City Pilot 1 Engineering Report. Lehrstuhl für Geoinformatik, TU München. <http://www.opengis.net/doc/PER/FCP1-ER>.
- Chaturvedi, K.; Yao, Z.; Kolbe, T. H. (2019): Integrated Management and Visualization of Static and Dynamic Properties of Semantic 3d City Models. In: International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, XLII-4/W17, S. 7-14. <https://doi.org/10.5194/isprs-archives-XLII-4-W17-7-2019>.
- Coors, V.; Andrae, A.; Böhm, K.-H. (2016): 3D-Stadtmodelle – Konzepte und Anwendungen mit CityGML. Wichmann, Berlin/Offenbach.
- Coors, V.; Schneider, S. (2018): Smart Cities benötigen Sensoren und smarte Geodaten. In: Zeitschrift für das Öffentliche Vermessungswesen des Landes Sachsen-Anhalt, 24 (1), S. 7-16.
- Cozzi, P.; Liley, S.; Getz, G. (2019): 3D Tiles Specification. Version: 1.0. <http://www.opengis.net/docs/CS/3DTiles/1.0>, Zugriff 2/2019.
- Graf, G.; Holuba, K.-H.; Blessing, E. (2016): 3D-Webvisualisierung von Planungsdaten. In: Kolbe, T. H. et al. (Hrsg.): Geoinformationssysteme 2016. Beiträge zur 3. Münchner GI-Runde. Wichmann, Berlin/Offenbach, S. 187-195.
- Gröger, G.; Kolbe, T. H.; Nagel, C.; Häfele, K.-H. (2012): OGC City Geography Markup Language (CityGML) Encoding Standard. Version 2.0.0, OGC 12-019. <http://www.opengis.net/spec/citygml/2.0>, Zugriff 11/2018.
- Huang, C. Y.; Wu, C. H. (2016): Design and Implement an Interoperable Internet of Things Application Based on an Extended OGC SensorThings API Standard. In: International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, 41.
- Jazayeri, M. A.; Liang, S. H. L.; Huang, C.-Y. (2015): Implementation and Evaluation of Four Interoperable Open Standards for the Internet of Things. In: Sensors, 15 (9), S. 24343-24373. DOI:10.3390/s150924343.
- Kasprzyk, J. P.; Nys, G. A.; Billen, R. (2019): Integration of multiple sensor data into a 3D GIS for cities monitoring. Belgium Geographers Days, Ghent University. <https://orbi.uliege.be/handle/2268/240420>.
- Liang, S. H. L.; Huang, C.-Y.; Khalafbeigi, T. (2016a): OGC SensorThings API Part 1: Sensing. Version 1.0 <http://www.opengis.net/doc/is/sensorthings/1.0>, Zugriff 11/2018.
- Liang, S. H. L.; Khalafbeigi, T. (2016b): OGC SensorThings API Part 2: Tasking Core. Version 1.0. <http://docs.opengeospatial.org/is/17-079r1/17-079r1.html>, Zugriff 2/2019.
- Pesch, R.; Zeh, R.; Pietruschka, D.; Stockinger V. (2018): Quervergleich oberflächennaher Geothermiesysteme in EnVisaGePlus. In: Tagungsband „Der Geothermiekongress DGK 2018“, 27.-29.11.2018, Essen, S. 1-15.
- Santhanavanich, T. (2018): Visualization and Analysis of E-bike Usage in 3D City Model by Integration of Heterogeneous Sensor Data. Masterthesis, Hochschule für Technik Stuttgart. DOI:10.13140/RG.2.2.14606.00324.
- Santhanavanich, T. (2019a): SensorThings Importer: A Tool for Importing Data from several Formats to SensorThings Server: Zenodo. DOI: 10.5281/ZENODO.3237680.
- Santhanavanich, T. (2019b): SensorThings-Aggregator: A Tool for Aggregating Time-series Data from SensorThings API Observation Entity: Zenodo. DOI:10.5281/ZENODO.323-8443.
- Santhanavanich, T.; Coors, V. (2019): City-Things: An Integration of the Dynamic Sensor Data to the 3d City Model. In: The 2nd International Conference on Urban Informatics, The Hong Kong Polytechnic University.
- Santhanavanich, T.; Schneider, S.; Rodrigues, P.; Coors, V. (2018): Integration and Visualization of Heterogeneous Sensor Data and Geospatial Information. In: ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences, IV-4/W7, S. 115-122. DOI:10.5194/isprs-annals-IV-4-W7-115-2018.
- Sini, S. K. (2019): Sensor Data Integration for Smart Villages – A Case Study of a Plus Energy District in Wüstenrot. Masterthesis, Hochschule für Technik Stuttgart.
- Soto, J. A. C.; Werner-Kytölä, O.; Jahn, M.; Pullmann, J.; Bonino, D.; Pastrone, C.; Spirito, M. (2016): Towards a federation of smart city services. In: Proceeding of International Conference on Recent Advances in Computer Systems. DOI:10.2991/racs-15.2016.28.
- Tomkins, J.; Lowe, D. (2016): TimeseriesML 1.0 – XML Encoding of the Timeseries Profile of Observations and Measurements. OGC 15-042r3, Version 1.0, Sept. 2016. <http://www.opengis.net/doc/is/tsml/1.0>.
- Trilles, S.; Luján, A.; Belmonte, Ó.; Montoliu, R.; Torres-Sospedra, J.; Huerta, J. (2015): SEnviro: a sensorized platform proposal using open hardware and open standards. In: Sensors, 15 (3), S. 5555-5582. DOI:10.3390/s150305555.
- Yao, Z.; Nagel, C.; Kunde, F.; Hudra, G.; Willkomm, P.; Donaubaauer, A. et al. (2018): 3DCityDB – a 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML. In: Open Geospatial Data, Software and Standards, 3 (1), S. 5. DOI:10.1186/s40965-018-0046-7.
- Zhu, W.; Simons, A.; Wursthorn, S.; Nierschu, A. (2016): Integration of CityGML and Air Quality Spatio-Temporal Data Series via OGC SOS. In: Proceedings of the Geospatial Sensor Web Conference (GSW), Münster, S. 29-31.