IAPG
Institut für
Angewandte Photogrammetrie
und Geoinformatik

VolkswagenStiftung

JADEHOCHSCHULE
Wilhelmshaven Oldenburg Elsfleth

Einsatz von SpatiaLite auf teilautonomen Unterwasserfahrzeugen

Tobias Werner, Thomas Brinkhoff

FOSSGIS 2018, Bonn

Institut für Angewandte Photogrammetrie und Geoinformatik (IAPG)
Jade Hochschule, Oldenburg, Germany

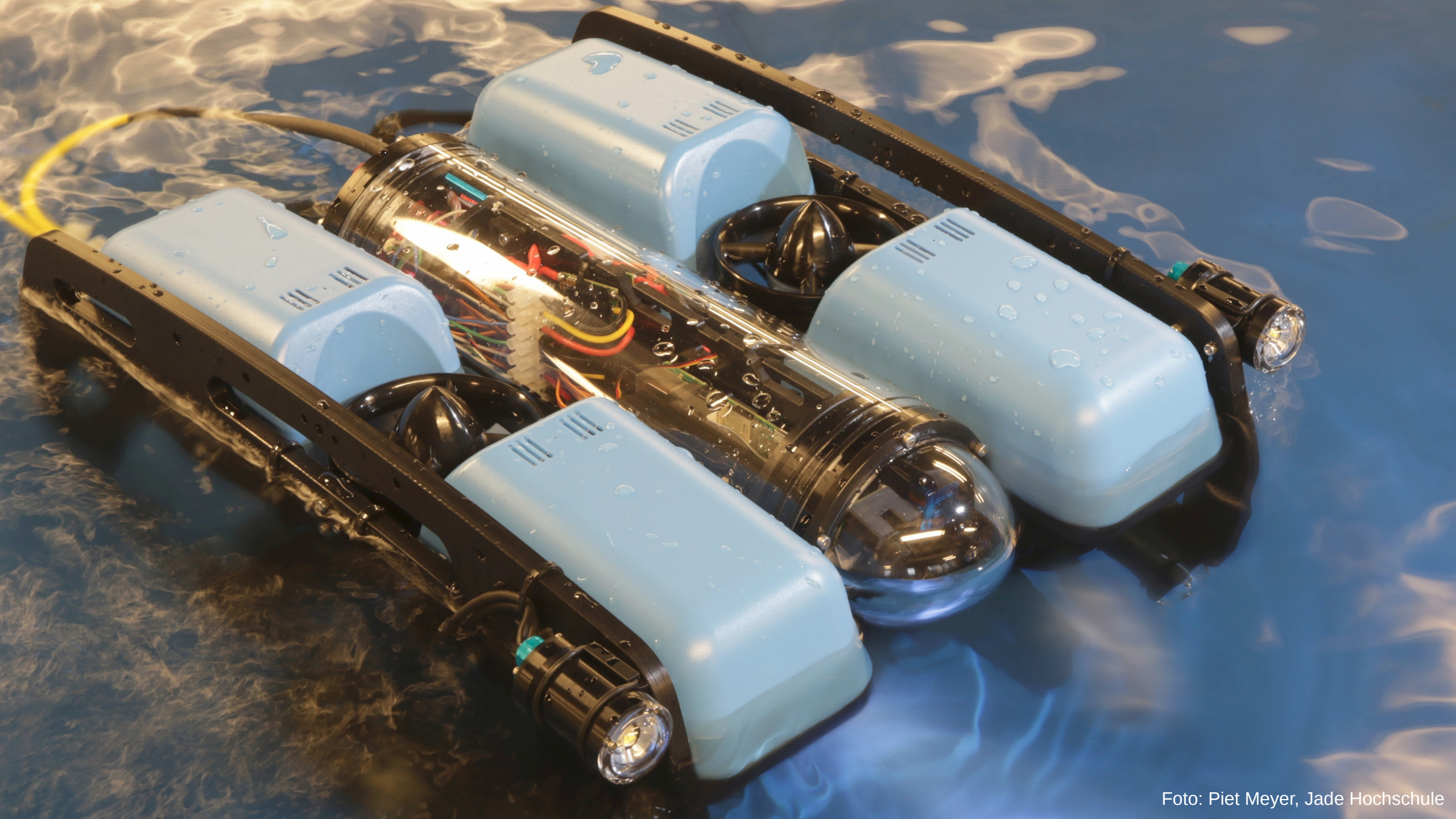


Foto: Piet Meyer, Jade Hochschule

Anwendungen Unterwasserfahrzeug

- Inspektion von Offshorebauwerken [1]
- Suche nach Kontaminationsquellen [2]
- Einsammeln von Messdaten aus einem Unterwassersensornetzwerk [3]

Allgemeine Entwicklung

- Kostengünstige Open-Source Hardware vereinfachen den Zugang zur Unterwasserwelt
- Technologien zur Unterstützung von Unterwassermissionen werden benötigt

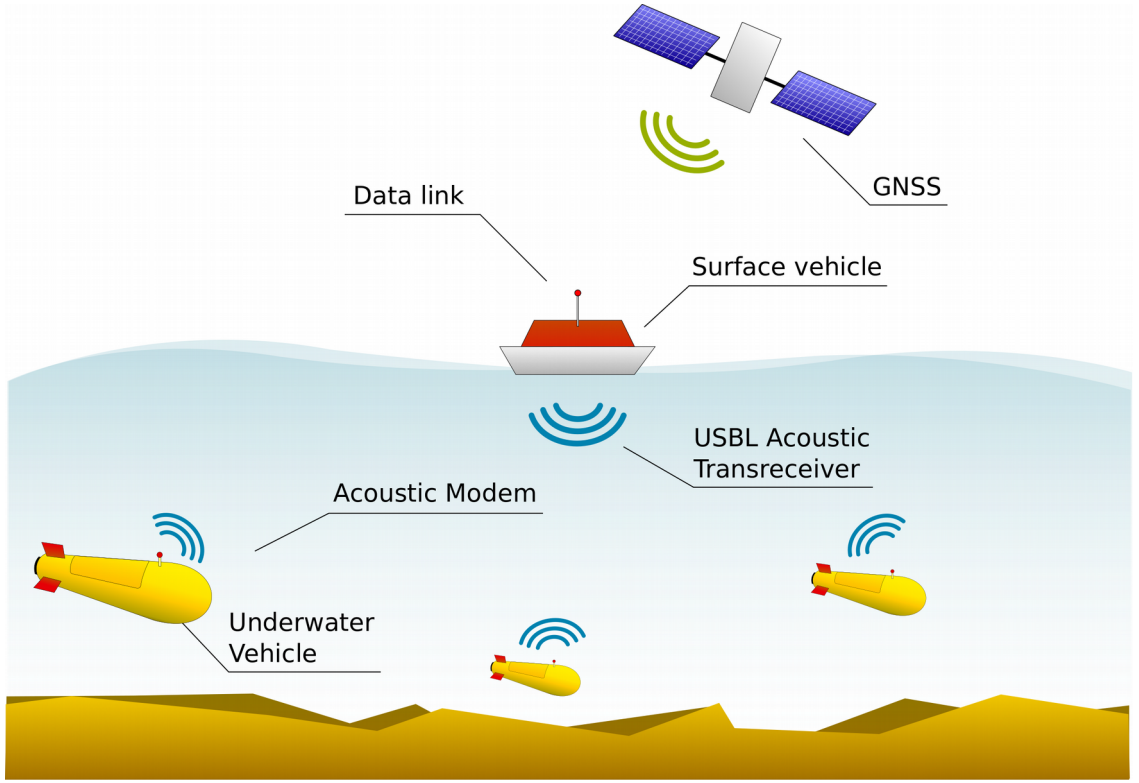


Abb. 1: Zusammenspiel von Über- und Unterwasserfahrzeugen

GLIEDERUNG

- Projekt EITAMS
- Zielsetzung
- Eingesetzte Hard-/Software
- Verwaltung Sensormessdaten
- Ergebnisse
- Fazit & Ausblick

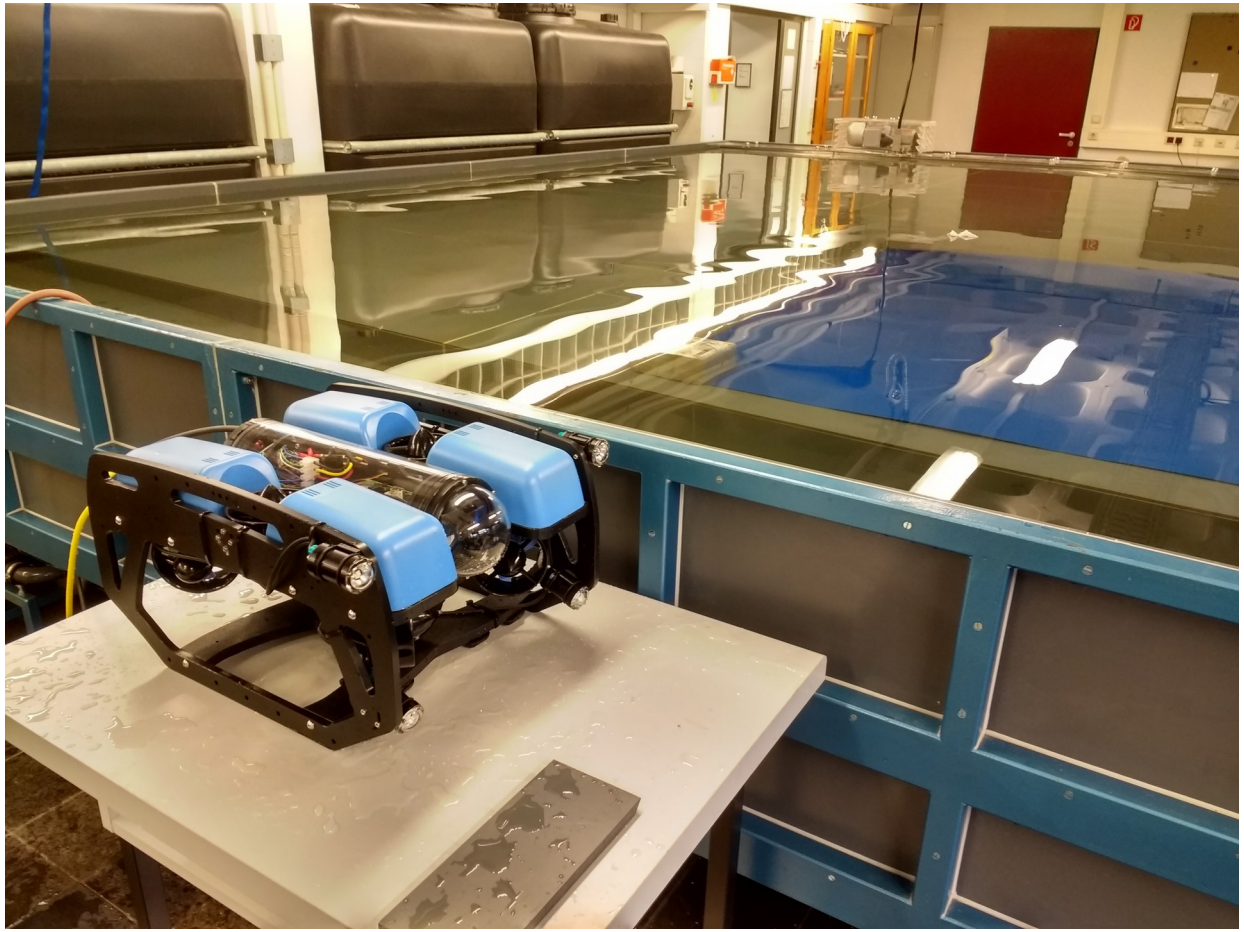


Abb. 2: Testumgebung für Unterwasserfahrzeug

PROJEKT „EITAMS“

- „**E**ntwicklung **I**nnovativer **T**echnologien für **A**utonome **M**aritime **S**ysteme“
  - Kognitive Systemmodellierung
  - Suchalgorithmen für kooperierende AUVs
  - Optische Unterwasser 3D-Messtechnik
  - **Datenmanagement**
  - Entwicklung eines Überwasserfahrzeugs
- Gefördert durch VW-Vorab
- 01/2017 bis 12/2020

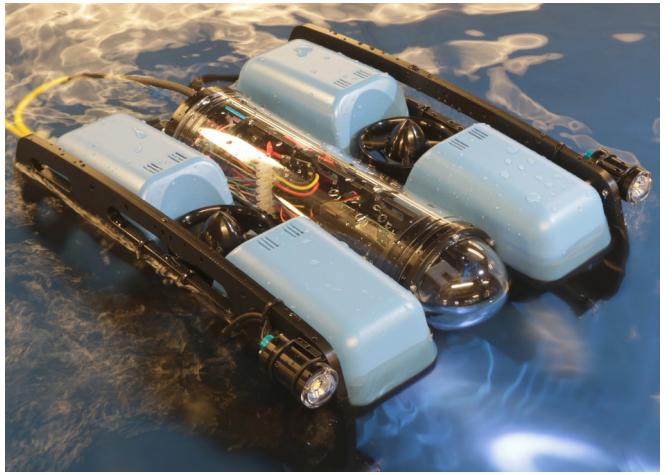


Abb. 3: Unterwasserfahrzeug

Abb. 4: Stereokamerasystem

Bereitstellung eines Datenmanagementsystems für Unterwasserfahrzeuge

- Speicherung und Bereitstellung erhobener Daten und Metadaten
- Definition und Bereitstellung raum-zeitlicher Abfragen
- Organisation gespeicherter Daten auf Basis raum-zeitlicher Kriterien

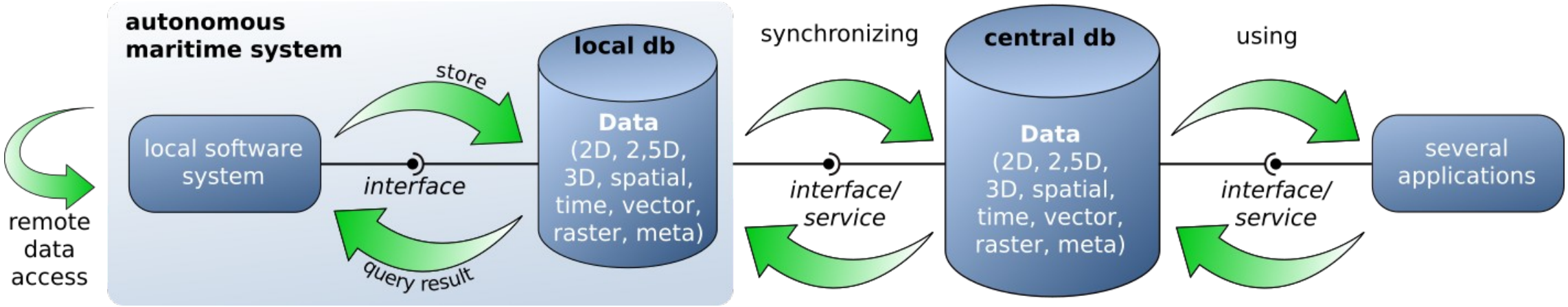


Abb. 5: Architektur des geplanten Gesamtsystems

Datenaustausch

- Luftbasierte Übertragungstechnologien wie WLAN nicht verwendbar

Begrenztes Gesamtgewicht

- Limitierung Energie
- Limitierung Hardware

Positionierung

- GNSS nicht verwendbar

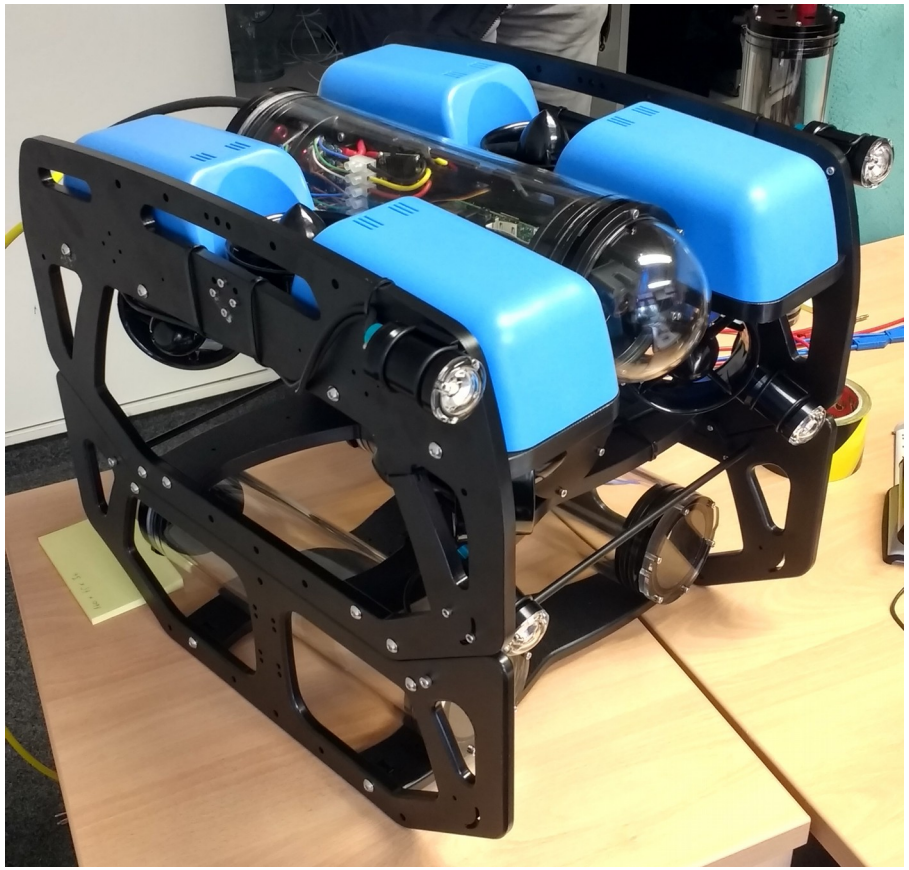


Abb. 6: Montiertes Payload Skid dient zur Anbringung zusätzlicher Hardware

EINGESETZTE HARD-/SOFTWARE

Trägerplattform

- BlueROV2 (Open-Source, ~3.500€) [4]
- Raspberry Pi 3 (4x 1,2GHz, 1GB RAM)

Kommunikation + Positionierung

- S2CR 18/34 USBL Akustik Modem [5]
- bis zu 3,5km mit max. 13,9 kbit/s

Steuerungssoftware

- DUNE: Unified Navigation Environment (Open-Source) [6]
  - https://github.com/LSTS/dune, Universität Porto
- Spezialisiert auf AUVs / UAVs + Simulationsmodus

Abb. 7: Akustik Modem (Antenne im Wasser)

Name	Abbreviation	Unit	Type
Latitude (WGS-84)	lat	*rad*	fp64_t
Longitude (WGS-84)	lon	*rad*	fp64_t
Height (WGS-84)	height	*m*	fp32_t
Offset north	x	*m*	fp32_t
Offset east	y	*m*	fp32_t
Offset down	z	*m*	fp32_t

Abb. 8: Publish/Subscribe von Vehikelinformationen

OGC SensorThings API

- Standard seit 2016
- „[…] interconnect IoT devices, data, and applications over the Web.“
- Teil des Sensor Web Enablements

Datenschema

- Grundlage zur Verwaltung von Sensormessdaten

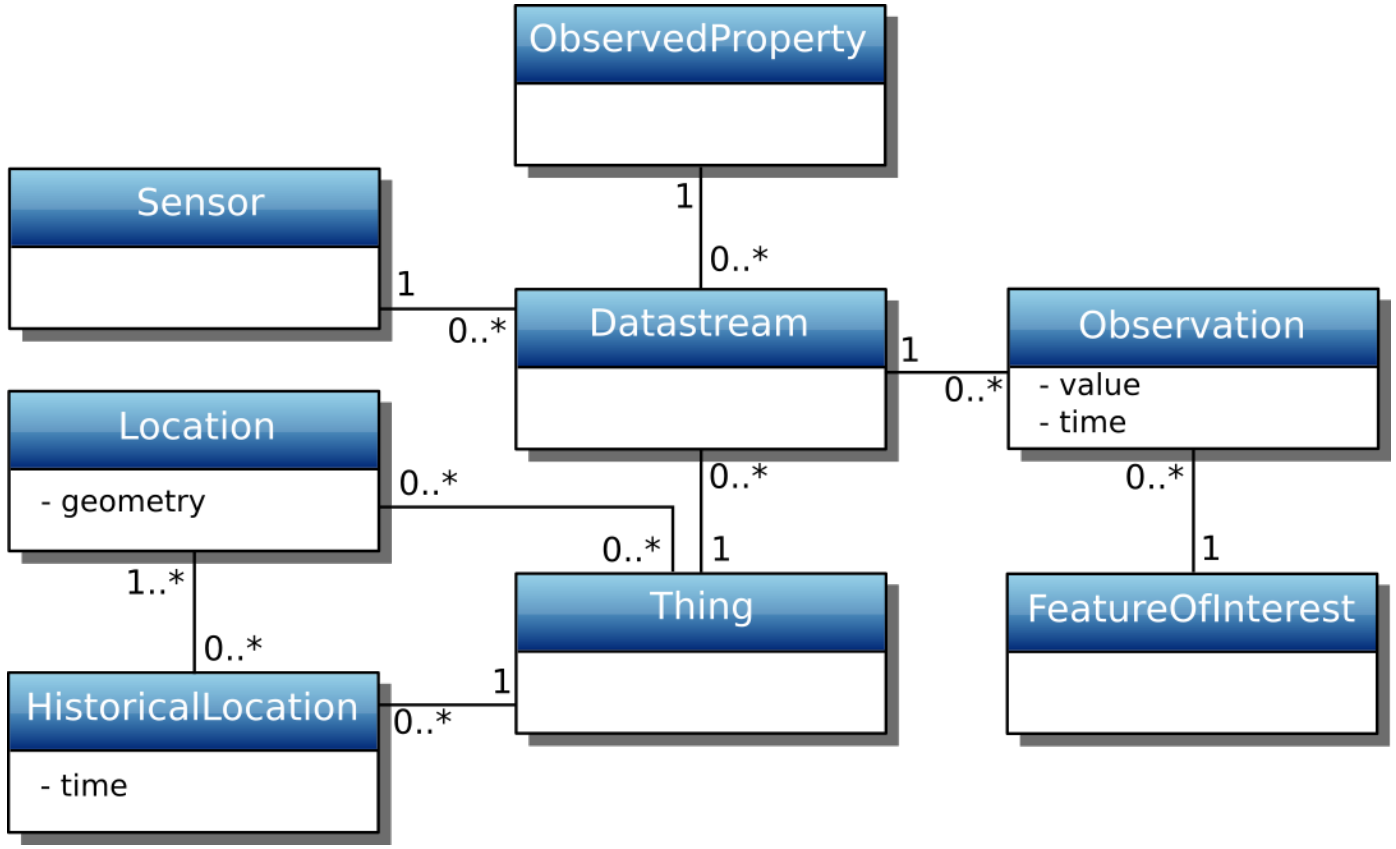


Abb. 9: Datenschema OGC SensorThings API [7]

SQLite

- Embedded Database Engine
- Bereits in DUNE integriert
- Cross plattform „One-file-based“

SpatiaLite

- Geo-Erweiterung für SQLite
- WKT, WKB, räumlicher Index, topologische Prädikate, ...

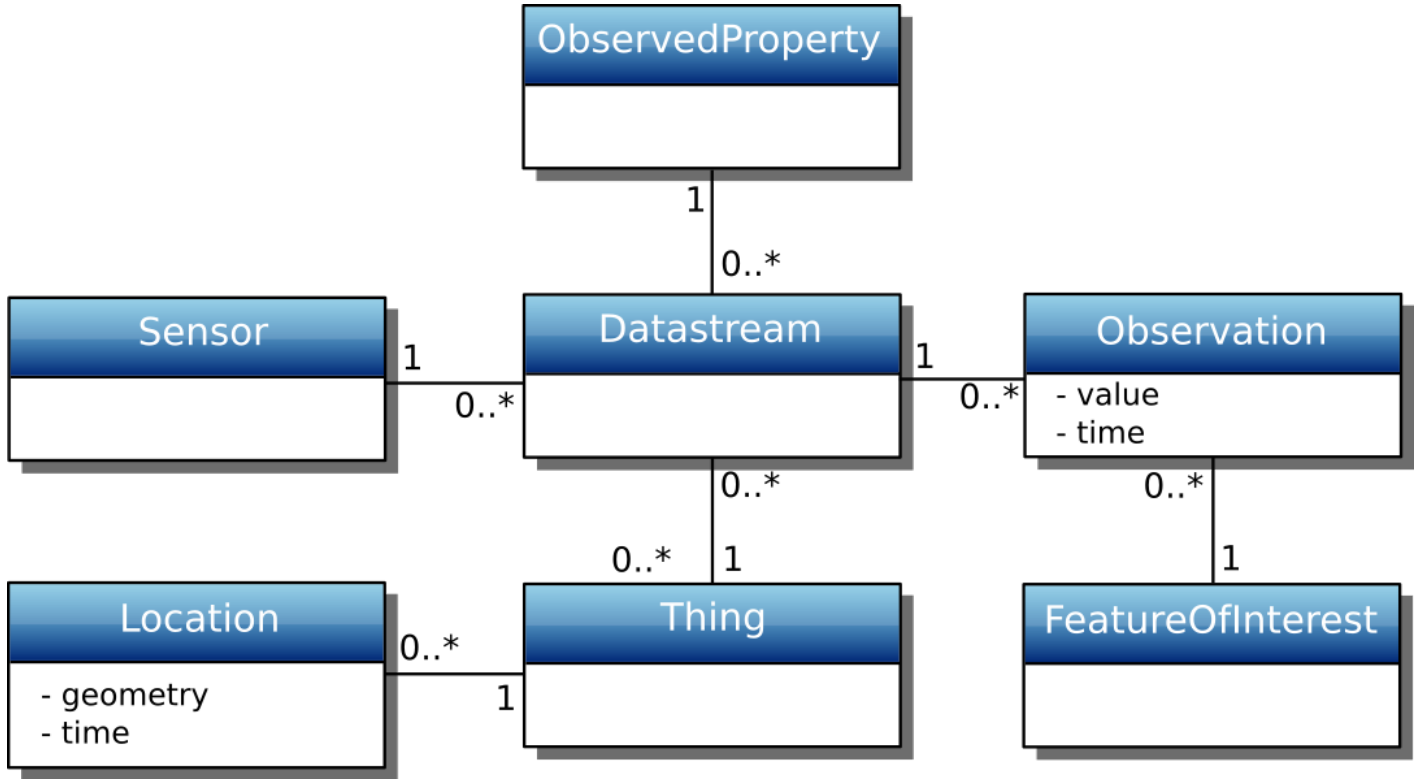
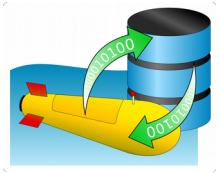


Abb. 10: Angepasstes Datenschema ohne HistoricalLocation



VERWALTUNG SENSORMESSDATEN

JADEHOCHSCHULE
Wilhelmshaven Oldenburg Elsfleth

Typsystem SQLite

- „Column type affinity“
- „Storage type affinity“

```
CREATE TABLE Observation {
    value REAL
};

INSERT INTO Observation(value) VALUES(21.1);
INSERT INTO Observation(value) VALUES(21);
INSERT INTO Observation(value) VALUES(„21“);
INSERT INTO Observation(value) VALUES(x‘1F‘);

SELECT * FROM Observation WHERE value > 20;

> 21.1
> 21.0
> 21.0
> ???

SELECT * FROM Observation WHERE value LIKE ‘%1%‘;

> 21.1
> 21.0
> 21.0
```

Erkenntnis

- Speicherung heterogener Messdaten in einem Attribut möglich
- Da Datentyp eines Datenstroms immer gleich auch praktikabel

Prepared Statements

- Einmalige Kompilierung in Bytecode → Performance

```
INSERT INTO Observation(data, stream_id) VALUES (?,?)
```

Placeholder

Geometrieerzeugung durch WKT

- ST_GeomFromText('POINTZ(1.0 2.0 1.0)')
- Problem → ST_GeomFromText() kann nicht an Statement gebunden werden

```
INSERT INTO Location(time, geom) VALUES(datetime('now'),?);
```

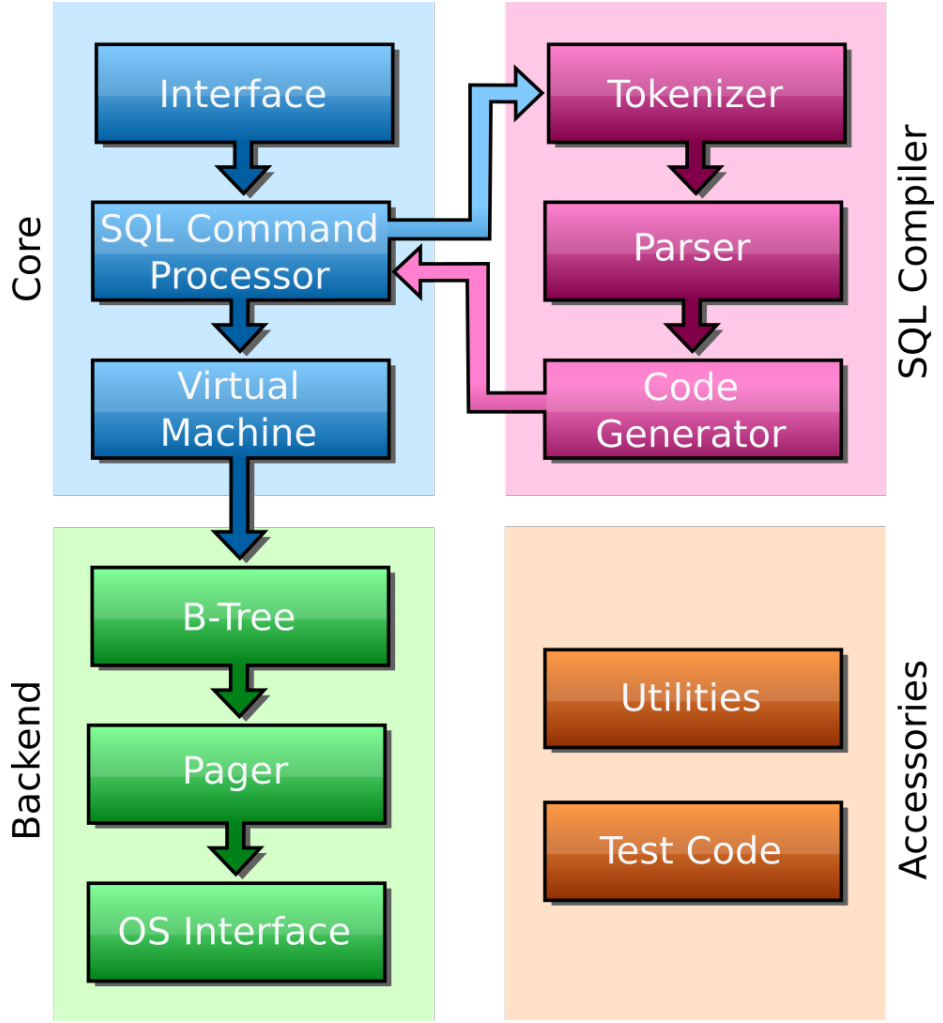
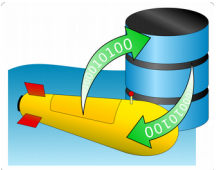


Abb. 11: Architektur SQLite [8]



VERWALTUNG SENSORMESSDATEN

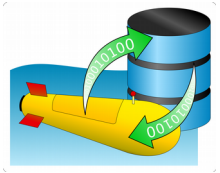

```
static const unsigned char point_geom_template[] = {
        0x00, 0x01, 0xE6, 0x10, 0x00, 0x00, 0x00, 0x00,
        0x00, 0x00, 0x00, 0x00, 0xF0, 0x3F, 0x00, 0x00,
        0x00, 0x00, 0x00, 0x00, 0x00, 0x40, 0x00, 0x00,
        0x00, 0x00, 0x00, 0x00, 0xF0, 0x3F, 0x00, 0x00,
        0x00, 0x00, 0x00, 0x00, 0x00, 0x40, 0x7C, 0xE9,
        0x03, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
        0x00, 0xF0, 0x3F, 0x00, 0x00, 0x00, 0x00, 0x00,
        0x00, 0x00, 0x40, 0x00, 0x00, 0x00, 0x00, 0x00,
        0x00, 0x08, 0x40, 0xFE
};
```

- Template zur Repräsentation von 3D Punkten (SpatiaLite internes Format)

- Aktualisierung Geometrie und MBR

```
// MBR
memcpy(geom + 6, lat_tmp, sizeof(lat));
memcpy(geom + 22, lat_tmp, sizeof(lat));
memcpy(geom + 14, lon_tmp, sizeof(lon));
memcpy(geom + 30, lon_tmp, sizeof(lon));

// Raw Geom
memcpy(geom + 43, lat_tmp, sizeof(lat));
memcpy(geom + 51, lon_tmp, sizeof(lon));
memcpy(geom + 59, height_tmp, sizeof(height));
```



VERWALTUNG SENSORMESSDATEN

JADEHOCHSCHULE
Wilhelmshaven Oldenburg Elsfleth

```
CREATE VIEW gis_temperature AS
    SELECT o.id,o.data,l.geom FROM observation o
    JOIN location l ON o.time >= l.time
    GROUP BY o.id
    ORDER BY o.time;

INSERT INTO views_geometry_columns (
    view_name,
    view_geometry,
    view_rowid,
    f_table_name,
    f_geometry_column,
    read_only)
VALUES (
    'gis_temperature',
    'geom',
    'id',
    'location',
    'geom',
    1);
```

- View verortet Beobachtung
- Left join zum nächsten Zeitstempel

- Anlegen einer Spatial View (für QGIS)

Simulationsfahrt „Roter Sand“

- Beobachtung von Wassertemperatur
- 113 Messwerte auf 107m
- Direkt in QGIS auswertbar

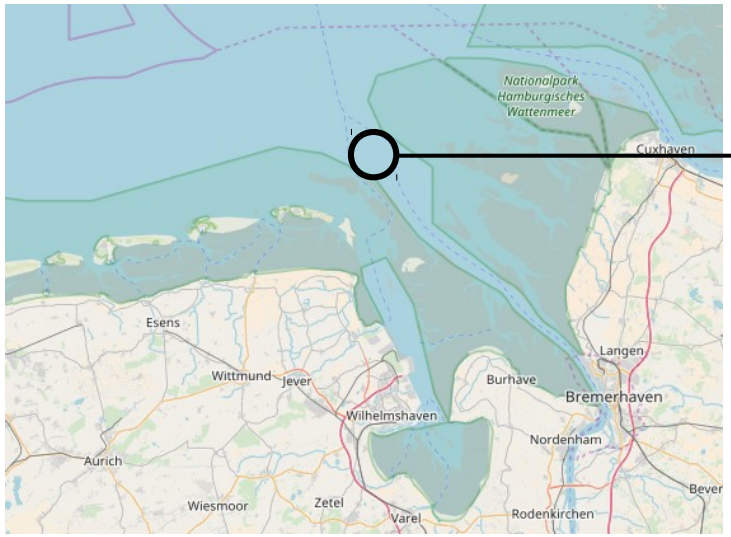


Abb. 12: Ausschnitt Simulationsgebiet

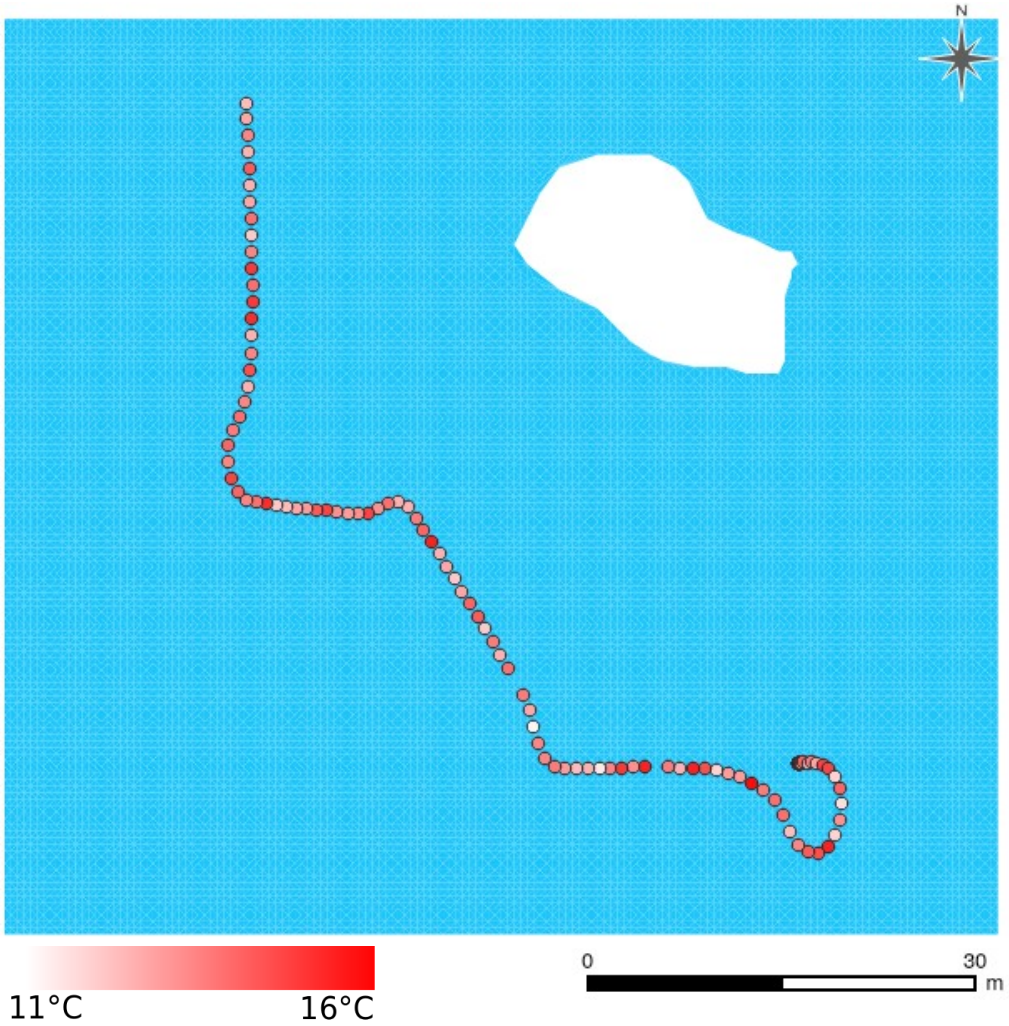


Abb. 13: Simulierte AUV-Temperaturmessung

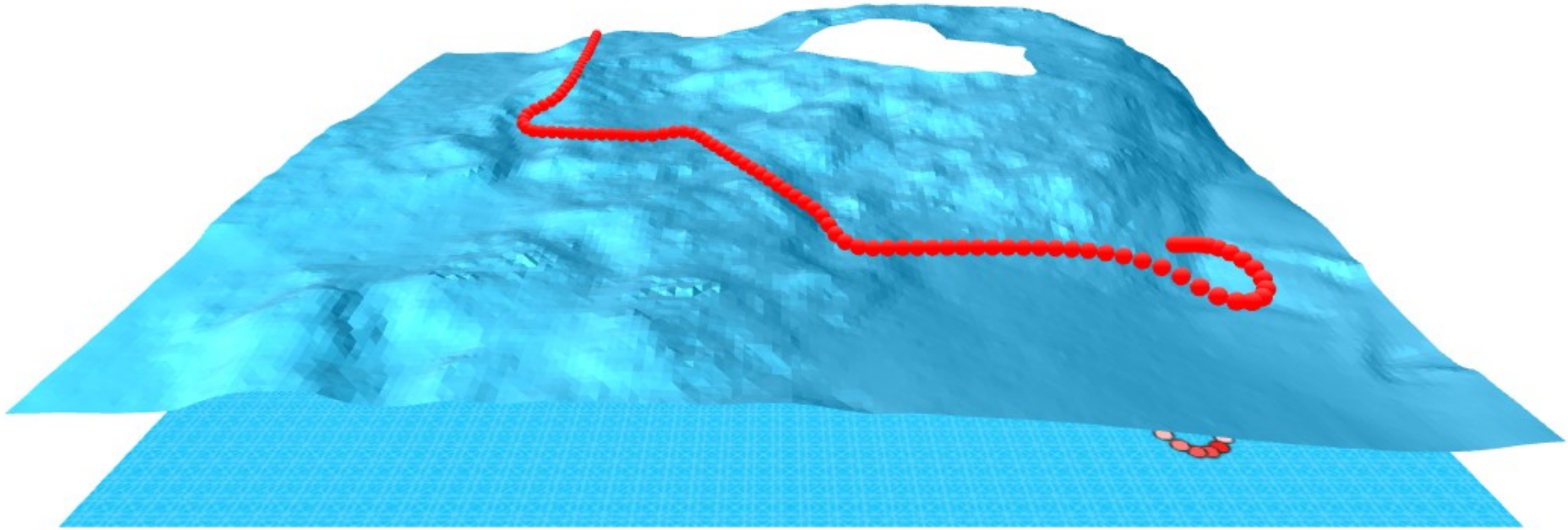
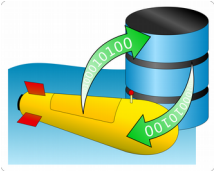


Abb. 14: Simulierte AUV-Temperaturmessung in 3D-QGIS-Darstellung



FAZIT & AUSBLICK

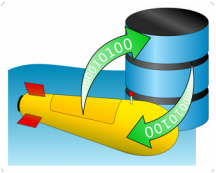
JADEHOCHSCHULE
Wilhelmshaven Oldenburg Elsfleth

Fazit

- SensorThings API Schema + SpatiaLite erlauben Verwaltung heterogener Sensormessdaten auf Unterwasserfahrzeugen
- Geometrieblobs für Prepared Statements müssen applikationsseitig definiert werden
- JOIN auf Timestamps erfordert individuelle Lösungen

Ausblick

- Portierung von Mechanismen aus Datenstrommanagementsystemen auf Unterwasserplattform
- Anwendung topologischer Prädikate und geometrischer Funktionen im 3D-Raum



LITERATUR

[1] Micha l Ciszewski et al. “Robotic system for off-shore infrastructure monitoring”. In: Journal of Marine Engineering & Technology 16.4 (2017), pp. 310–318.

[2] Xiang Cao, Hongbing Sun, and Gene Eu Jan. “Multi-AUV cooperative target search and tracking in unknown underwater environment”. In: Ocean Engineering 150 (2018), pp. 1–11.

[3] Guangjie Han et al. “Probabilistic Neighborhood-Based Data Collection Algorithms for 3D Underwater Acoustic Sensor Networks”. In: Sensors 17.12 (2017), p. 316.

[4] Blue Robotics Inc. (2018), BlueROV2, Retrieved from https://www.bluerobotics.com/store/rov/bluerov2/

[5] EvoLogics GmbH (2018), Products / Underwater USBL Positioning Systems / S2CR 18/34 USBL | EvoLogics GmbH , Retreived from https://www.evologics.de/en/products/usbl/s2cr_18_34_usbl.html

[6] J. Pinto, Dias, P. Sousa, Martins, R., Fortuna, J., Marques, E. R. B., and de Sousa, J. Borges, “The LSTS toolchain for networked vehicle systems”, 2013 MTS/IEEE OCEANS - Bergen, pp. 1–9, 2013.

[7] Liang, Steve H.L., Chih-Yuan Huang, and Tania Khalafbeigi. "OGC SensorThings API Part I:Sensing" OGC® Implementation Standard (2016)

[8] SQLite Team (2018), Architecture of SQLite, Retreived from https://sqlite.org/arch.html

Vielen Dank für Eure Aufmerksamkeit