



Christoph Schlager, BSc

Interoperabler webbasierter Zugriff auf heterogene Sensordaten

MASTERARBEIT

zur Erlangung des akademischen Grades eines

Master of Science (MSc.)

in

Geospatial Technologies

[gemeinsames Studium im Rahmen von NAWI Graz]

eingereicht an der

Technischen Universität Graz

Betreuer:

Ao.Univ.-Prof. Dr.phil. tit.Univ.-Prof. Univ.Prof. Norbert Bartelme

Institut für Geoinformation der Technischen Universität Graz

Graz, September 2013

EIDESSTÄTTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am

.....

(Unterschrift)

STATUTORY DECLARATION

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

.....

(date)

.....

(signature)

ZUSAMMENFASSUNG

Die vorliegende Masterarbeit beschäftigt sich mit dem webbasierten Zugriff auf heterogene Sensordaten sowie dem einfachen und effizienten Austausch dieser über System- und Domänengrenzen hinweg. Den Kern der Arbeit zur Umsetzung neuer Geo-Sensor Web Ansätze bildet dabei die vom *Open Geospatial Consortium* (OGC) gegründete *Sensor Web Enablement* (SWE) Initiative, deren Ziel die Echtzeit-Integration heterogener Sensors-Webs in eine Informationsinfrastruktur ist. Anhand von standardisierten Schnittstellen und Metadatenkodierungen der Initiative sollen Sensoren über das Internet auffindbar, abfragbar und steuerbar gemacht werden.

Bereits heute zeigen bestehende Forschungsprojekte und Implementierungen die Möglichkeit eines operativen Einsatzes von standardisierten Geo-Sensor Netzwerken. Der vom Forschungsinstitut iSPACE gemeinsam mit der Firma SYNERGIS konzipierte *Live Geography* Ansatz integriert Messungen von Echtzeitdaten in eine voll standardisierte Infrastruktur und kombiniert diese mit anderen Systemen. Um eine umfangreiche Überwachung der Umwelt und ein hocheffizientes Krisenmanagement im Katastrophenfall zu gewährleisten, wurde im Zuge des Projekts *Open architecture for Smart an Interoperable networks in Risk management based on In-situ Sensors* (OSIRIS) eine auf Basis der OGC Standards realisierte serviceorientierte Architektur entwickelt. Das Sensor Web des Wupperverbands ist ein weiteres Beispiel für die Implementierung ausgewählter Komponenten der SWE Initiative. Eine Vielzahl an heterogenen Sensoren und Fachsystemen wird über standardisierte Schnittstellen und Datenformate in das webbasierte geographische Informationssystem FluGGS des Wupperverbands integriert.

Der praktische Teil der Arbeit widmet sich der Umsetzung eines standardisierten webbasierten Zugriffs auf Echtzeitdaten und Zeitreihen des vom Wegener Center betriebenen Klimamessstationsnetzes WegenerNet sowie auf aufbereitete meteorologische Stationsdaten der *Zentralanstalt für Meteorologie und Geodynamik* (ZAMG). Die Implementierung beider Instanzen ist über die *Sensor Observation Service* (SOS) Spezifikation der SWE Initiative realisiert. Zur Konvertierung der Datenformate des WegenerNets und der ZAMG Testdaten dient eine in C++ entwickelte Applikation je Instanz.

ABSTRACT

This master thesis deals with web based access to heterogeneous sensor data as well as simple and efficient exchange of this data across domain boundaries to establish new geo-sensor web realizations. Core of the thesis is the *Sensor Web Enablement (SWE)* initiative founded by the *Open Geospatial Consortium (OGC)*. The initiative enables – based on a set of interfaces and metadata encodings – finding, accessing and sharing of sensors and their data across different applications through the internet.

Already existing research projects and implementations are showing the possibility of an operative deployment of standardized geo-sensor networks. The *Live Geography* approach, set up by the research institute iSPACE in cooperation with the organization SYNERGIS, integrates real-time measurement data in a fully standardized infrastructure and allows a combination with other systems. To ensure a comprehensive monitoring of the environment and efficient crisis management in case of an environmental disaster, the project *Open architecture for Smart an Interoperable networks in Risk management based on In-situ Sensors (OSIRIS)* developed a service-oriented architecture mainly based on OGC standards. Another example for the implementation of selected components of the SWE initiatives is the sensor web of the *Wupperverband*. A large number of heterogeneous sensors and other specialized systems are integrated through standardized interfaces and data formats into the geographic information system *FlussGebietsGeoinformationsSystem (FluGGS)*.

The practical part of the thesis is devoted to the realization of a standardized web based access to real time data and time series from the climatological stations network WegenerNet provided by the WegenerCenter, as well as to processed meteorological data from stations operated by the *Zentralanstalt für Meteorologie und Geodynamik (ZAMG)*. The implementation of both instances is realized through the *Sensor Observation Service (SOS)* specification of the SWE. For converting data formats from the WegenerNet and the ZAMG test data into a SOS compliant format, two applications developed in C++ are used.

INHALTSVERZEICHNIS

ZUSAMMENFASSUNG	V
ABSTRACT	VI
ABKÜRZUNGSVERZEICHNIS	IX
ABBILDUNGSVERZEICHNIS	XII
TABELLENVERZEICHNIS	XV
DANKSAGUNG	XVII
1 EINLEITUNG	19
1.1 Zielsetzung	20
1.2 Methodik und Aufbau der Arbeit	21
2 GRUNDLAGEN	22
2.1 Open Geospatial Consortium (OGC)	22
2.2 Sensor Web Enablement (SWE)	23
2.3 Observations & Measurements (O&M)	26
2.3.1 Eigenschaften und Begriffe im Zusammenhang mit Observations	26
2.3.2 Konzeptionelles Modell einer Beobachtung	28
2.3.3 Klassifikation der Observations aufgrund des Ergebnisses	30
2.3.3.1 Observations mit konstantem Ergebnis	31
2.3.3.2 Observations mit variablem Ergebnis	32
2.3.4 Sampling Feature	33
2.4 Sensor Model Language (SensorML)	35
2.4.1 Allgemeines	35
2.4.2 Beziehung des Sensors zu einer Plattform	36
2.4.3 Konzeptionelles Modell für Prozesse	38
2.5 PUCK Protocol Standard	40
2.6 Sensor Observation Service (SOS)	42
2.6.1 SOS 2.0 Modellüberblick	42
2.6.2 Observation Offering	43
2.6.3 Kern-Operationen (Core Operations)	44
2.6.3.1 GetCapabilities Operation	44
2.6.3.2 DescribeSensor Operation	45
2.6.3.3 GetObservation Operation	46
2.6.4 Extensions	47
2.7 Ablauf zum Abfragen oder Einfügen von Observations oder Ergebnissen	48
2.7.1 Workflow zum Abfragen einer Observation	48
2.7.2 Workflow zum Einfügen einer Observation	49
2.7.3 Workflow zum Einfügen eines Ergebnisses	50
2.7.4 Workflow zum Abfragen eines Ergebnisses	51
2.8 Sensor Planning Service (SPS)	51
2.9 Sensor Alert Service (SAS)	53
2.10 Web Notification Services (WNS)	54
3 GEO-SENSOR WEBS (STATE OF THE ART)	57
3.1 Senseable City Lab	57

3.2	Live Geography	60
3.2.1	Embedded System Design	61
3.2.2	Implementierungen.....	62
3.3	Sensor Web im Wupperverband	68
3.4	Erhöhte Interoperabilität durch Sensor-Web-Profile	69
3.4.1	GEOWOW	70
3.4.1.1	Social Benefit Area Water	70
3.5	Integrated Ocean Observing System (IOOS)	73
3.6	OSIRIS.....	76
3.6.1	Architektur	77
3.6.2	Testimplementierungen des OSIRIS Systems	79
3.6.2.1	Waldbrand	80
3.6.2.2	Feuer in Industriegebäuden	82
3.6.2.3	Luftverschmutzung.....	83
3.6.2.4	Monitoring der Wasserqualität	86
4	SOS-IMPLEMENTIERUNG DER INSTANZEN WEGENERNET UND ZAMG.....	88
4.1	52°North SOS Server	88
4.2	WegenerNet SOS	94
4.2.1	Allgemeines	94
4.2.2	Implementierung im SOS Datenmodell.....	97
4.2.2.1	Beschreibung der Beziehungen anhand einer Basisstation.....	97
4.2.2.2	WegenerNet Offerings	100
4.2.2.3	Applikation zum Erzeugen der SensorML-Dateien sowie zum Abfragen der Observations ..	102
4.2.3	Ergebnisse	104
4.2.3.1	Kern-Operationen (Core).....	104
4.2.3.2	Zugriff mittels Clients	110
4.3	ZAMG SOS.....	113
4.3.1	Allgemeines	113
4.3.2	Implementierung in das SOS Datenmodell	114
4.3.2.1	Datengrundlage	114
4.3.2.2	Applikation zur Erzeugung der SensorML-Dateien und Erstellung des SOS Modells	116
4.3.2.3	ZAMG Offerings	117
4.3.3	Ergebnisse	118
4.3.3.1	SOS Operationen	118
4.3.3.2	Zugriff mittels Client	122
5	FAZIT UND AUSBLICK	125
6	LITERATURVERZEICHNIS.....	129

ABKÜRZUNGSVERZEICHNIS

AdF	Acquedotto del Fiora
BBOX	Bounding Box
CEP	Complex Event Processing
CI	Coastal Inundation
CODIS	Centre Opérationnel Départemental d'Incendie et de Secours
CO-OPS	Center for Operational Oceanographic Products and Services
DAP	Data Access Protocol
DGPS	Differential Global Positioning System
DIF	Data Integration Framework
ECM	European Centre for Mechatronics
ERDDAP	Environmental Research Division's Data Access Program
ER-Modell	Entity-Relationship-Modell
ETM	Enhanced Thematic Mapper
FluGGS	FlussGebietsGeoinformationsSystem
FP6	Sixth Framework Programme
FP7	Seventh Framework Programme
GDI	Geodateninfrastruktur
GENESIS	GENERIC European Sustainable Information Space for environment
GEOSS	Global Earth Observation System of Systems
GEOWOW	GEOSS Interoperability for Weather Ocean and Water
GIS	Geographischen Informationssystem
GMES	Global Monitoring for Environment and Security
GPRS	General Packet Radio Service
GPS	Global Positioning System
GRDC	Global Runoff Data Centers
GSM	Global System for Mobile Communications
HAB-FS	Harmful Algal Bloom Forecast System
HI	Hurricane Intensity
HWRF	Hurricane Weather Research And Forecast System
HYCOM	Hybrid Coordinate Ocean Model
ICT	Information and Communications Technologies

IDW	Inverse Distance Weighting
IEA	Integrated Ecosystem Assessments
IEEE	Institute of Eletrical and Electronic Engineers
IOOS	Integrated Ocean Observing System
JTS	Java Topology Suite
KiWIS	KISTERS Web Interoperability Solution
LIDAR	Laser detection and ranging
MC	Management Committee
MGI	Militärgeographisches Institut
MIT	Massachusetts Institute of Technology
MUC	Multi User Chat
NAS	Notification und Alert Service
NDBC	National Data Buoy Center
NOAA	National Ocean and Atmospheric Administration
NTRIP	Networked Transport of RTCM via Internet Protocol
O&M	Observations & Measurements
OBA	On Board Units
OGC	Open Geospatial Consortium
OPeNDAP	Open-source Project for a Network Data Access Protocol
ORDMS	objektrelationales Datenbankmanagementsystem
OSIRIS	Open architecture for Smart an Interoperable networks in Risk management based on In-situ Sensors
OWS	OpenGIS Web Services
ppb	parts per billion
QC-Flag	Quality Control Flag
RSS	Really Simple Syndication
RT	Region Toscana
RTGA	Real-time Geo-awareness
SAS	Sensor Alert Service
SBA	Social Benefit Areas
SensorML	Sensor Model Language
SES	Sensor Event Service
SIGs	Special Interest Groups
SOA	serviceorientierte Architektur
SPS	Sensor Planning Service

SWE	Sensor Web Enablement
TAWES	teilautomatische Wetterstationen
TC	Technical Committee
UAV	unmanned aerial vehicle
UML	Unified Modeling Language
URN	Uniform Resource Name
VSAT	very small aperture terminal
WCAS	Web Catalogue Service
WCM	Water Catastrophe Management
WCS	Web Coverage Service
WFS	Web Feature Service
WMS	Web Map Service
WNS	Web Notification Services
WPS	WegenerNet Processing System
WSDL	Web Services Description Language
XML	Extensible Markup Language
XMPP	Message Presence Protocol
ZAMG	Zentralanstalt für Meteorologie und Geodynamik

ABBILDUNGSVERZEICHNIS

Abbildung 1: Konzept des Sensor Webs (Reed, et al., 2013, S. 5).....	23
Abbildung 2: SWE-Standards - konzeptionelle Funktionsweise (Koch, et al. 2013 nach Botts et al. 2007).....	25
Abbildung 3: Beschreibung einer Beobachtung als UML-Diagramm (Cox, 2010, S. 10)	30
Abbildung 4: Stationsraster des WegenerNet (ca. 1,4km × 1,4km Gitter) mit den insgesamt 27 WegenerNet-Gemeinden (Kabas, 2011, S. 25)	33
Abbildung 5: Beziehung des lokalen Koordinatensystems des Sensors (Orange) zum lokalen Koordinatensystem der Plattform (schwarz) (Botts & Robin, 2007, S. 34)	37
Abbildung 6: Konzeptionelles Modell für Prozesse (Botts & Robin, 2007, S. 55).....	39
Abbildung 7: OGC SWE Services, SensorML und PUCK Protokoll (O'Reilly, 2012, S. xii).....	41
Abbildung 8: Abhängigkeiten zwischen Core, Extensions und der OGC Spezifikationsbasis (Bröring, et al., 2012, S. 10)	43
Abbildung 9: Konzeptionelles Modell der <i>GetCapabilities</i> Operation (Bröring, et al., 2012, S. 18).....	45
Abbildung 10: Sequenz der Operationen zum Abfragen einer Observation (Bröring, et al., 2012, S. 12).....	49
Abbildung 11: Sequenz der Operationen zum Einfügen einer Observation (Bröring, et al., 2012, S. 13).....	50
Abbildung 12: Sequenz der Operationen zum Einfügen eines Ergebnisses (Bröring, et al., 2012, S. 15)	50
Abbildung 13: Sequenz der Operationen zum Abrufen eines Ergebnisses (Bröring, et al., 2012, S. 15).....	51
Abbildung 14: Exemplarischer Workflow zur Steuerung eines unbemannten Luftfahrzeugs (52North, 2013)....	52
Abbildung 15: SAS Workflow (52North, 2013).....	54
Abbildung: 16: WNS Workflow (52North, 2013).....	56
Abbildung 17: Erfassung der Aktivitäten im Raum mittels einer Kamera montiert auf einem Gebäude (Massachusetts Institute of Technology, o.J.).....	58
Abbildung 18: Transportwege des Abfalls ausgehend von Seattle (Massachusetts Institute of Technology, o.J.)	59
Abbildung 19: Grundlegende Architekturkomponenten und standardisierte Schnittstellen der Live Geography Infrastruktur (Resch & Mittelböck, 2010)	61
Abbildung 20: Embedded Device für standardisierte Messung und Übertragung von Umweltparametern links und Service Stack am Embedded Device rechts (Mittelböck, et al., 2009).....	62
Abbildung 21: Darstellung der Verschmutzungsparameter als Zeitreihe mit ESRI ArcGIS (Resch & Mittelböck, 2010)	63
Abbildung 22: Echtzeit-Interpolation von Umgebungstemperaturwerten für das Monitoring von optimalen Umweltparametern der lokalen Fauna und Flora im Nationalpark Berchtesgaden (Resch & Mittelböck, 2010)	64
Abbildung 23: Illustration der <i>GeoHealth</i> Applikation: Historie biometrischer Parameter mit Darstellung der geographischer Lage implementiert in einer MS Silverlight Umgebung (Resch & Mittelböck, 2010)	65

Abbildung 24: Webbasiertes Geo-Processing in Echtzeit, für eine detaillierte Einschätzung der urbanen Luftqualität (Resch & Mittelböck, 2010)	66
Abbildung 25: Interpolation der Strahlungspunktdaten in Echtzeit (Resch & Mittelböck, 2010).....	67
Abbildung 26: Die Sensor Web Infrastruktur des Wupperverbands (Wupperverband, 2013)	69
Abbildung 27: Menü des 52°North Sensor Web Clients zur Auswahl anzuzeigender Zeitreihen (aktuell ausgewählt sind die hydrologischen Daten des GRDC-SOS) (52°North Open Source Client, 2013)	72
Abbildung 28: Observations des National Data Buoy Center (NDBC) dargestellt über Google Maps (NOOA, 2013)	75
Abbildung 29: Vorhersager einer tropischen Depression links und eines Hurrikans rechts (NOOA, 2013).....	76
Abbildung 30: Serviceorientierte Architektur (SOA) des OSIRIS (OSIRIS, 2009, S. 46)	78
Abbildung 31: OSIRIS Architektur für ein Waldbrand-Management (OSIRIS, 2009, S. 67)	82
Abbildung 32: Ein GIS zeigt ein detektiertes Feuer (OSIRIS, 2009, S. 146).....	83
Abbildung 33: Architektur des Monitoring-Szenarios (OSIRIS, 2009 S. 155)	85
Abbildung 34: Einsatz der OSIRIS Services im Wasser-Monitoring-Demonstrationsprojekt (OSIRIS, 2009, S. 189)	87
Abbildung 35: 52°North SOS Architektur (Hollmann, 2012)	90
Abbildung 36: Datenbankschema der SOS 52°North Version 3.5.0 (Hollmann, 2012, S. 14)	93
Abbildung 37: Anordnung der Klimastationen mit Erhebung der Windparameter und Verteilung der Bodenmesspunkte im WegenerNet. Darstellung auf Geländeschattierung des Untersuchungsgebiets (Kabas, 2011, S. 50).....	95
Abbildung 38: Schematische Darstellung des WegenerNet Prozessierungssystems: (a) Kernsystem der Datenprozessierung und (b) Verbindungsstelle zur Datennutzung (Kabas, 2011, S. 75).....	96
Abbildung 39: Installierte Sensorik einer Basisstation und umgesetzte Bezeichnungen der Klimastation 10 nach der SOS Spezifikation (nach Kabas, 2011 S. 39)	98
Abbildung 40: Beziehungen der procedure, proc_foi und feature_of_interest Tabellen	99
Abbildung 41: Abhängigkeit der Observations von den Procedures, Feature-Of-Interest und Phenomena.....	100
Abbildung 42: SOS 2.0 GetCapabilities Request.....	104
Abbildung 43: Ausschnitte des GetCapabilities Response	105
Abbildung 44: Offering Bereich der abgefragten SOS 1.0.0 Schnittstelle	106
Abbildung 45: SOS 2.0 DescribeSensor Request Operation des Sensors org:wegcenter:temperature_2m_Klimastation_8	107
Abbildung 46: Teil der SOS 2.0 DescribeSensor Antwort im SensoML Format	108
Abbildung 47: Teil des SOS 2.0 Requests zur Abfrage von Observations des Procedures org:wegcenter:temperature_2m_Klimastation_8 mit zeitlichem Filter	109
Abbildung 48: Teil der SOS 2.0 GetObservation Antwort	110
Abbildung 49: WegenerNet-Stationsübersicht des 52°North SensorWebClient.....	111
Abbildung 50: WegenerNet Stationen welche Messwerte zur Eigenschaft Windgeschwindigkeit liefern	112

Abbildung 51: Darstellung der Zeitreihen ausgewählter WegenerNet Stationen mit dem 52°North SensorWebclient	112
Abbildung 52: Zugriff auf die WegenerNet SOS Daten mittels ArcGIS SOS Extension (Kartengrundlage: OpenStreetMap)	113
Abbildung 53: Meteorologische Messstationen der ZAMG, des ACG und Ö3 (ZAMG, 2013).....	114
Abbildung 54: SOS 2.0 GetFeatureOfInterest Request mit räumlichen Filter (BBOX) (Datengrundlage: GIS Steiermark, ZAMG).....	119
Abbildung 55: Teil der SOS 2.0 GetFeatureOfInterest Antwort	120
Abbildung 56: SOS 2.0 GetObservation Request mit zeitlichem (During) und räumlichen Filter (BBOX)	121
Abbildung 57: Teil der SOS 2.0 GetObservation Antwort	122
Abbildung 58: ZAMG-Stationsübersicht des 52°North SensorWebClient, Auswahlfenster zur Station in Admont	123
Abbildung 59: Zeitreihen Darstellung der relativen Luftfeuchtigkeit mit dem 52°North SensorWebClient (ZAMG Daten kombiniert mit Daten des Wupperverbands).....	123
Abbildung 60: Auszug der PostgreSQL Tabelle phenomenon	127

TABELLENVERZEICHNIS

Tabelle 1: Ergebnistypen für Observations mit konstantem Ergebnis (nach Cox, 2010, S. 18)	31
Tabelle 2: Ergebnistypen für Observations mit variablen Ergebnis (Cox, 2010, S. 19)	32
Tabelle 3: Geometrietypen räumlicher <i>Sampling Features</i> (Cox, 2010, S. 27)	34
Tabelle 4: Beziehung zwischen In-Situ und Remote-Sensoren auf mobilen oder fixen Plattformen (Botts & Robin, 2007, S. 35)	38
Tabelle 5: Webservices und Datenkodierung des IOOS Data Integration Frameworks (DIF) (IOOS, 2010, S. 42)	74
Tabelle 6: Offerings der WegenerNet SOS-Implementierung	101
Tabelle 7: Strukturprogramm der WegenerNet-Applikation nach Nassi-Schneidermann	103
Tabelle 8: Auszug der meteorologischen Daten der Station Mariazell des Jahres 2008 (Datengrundlage: ZAMG)	115
Tabelle 9: Strukturprogramm der ZAMG-Applikation nach Nassi-Schneidermann	117
Tabelle 10: Offerings der ZAMG SOS-Implementierung	118

DANKSAGUNG

Als erstes möchte ich mich bei meinen Eltern, sowie meiner Schwester, meinen Freunden und StudienkollegInnen bedanken, die alle einen großen Anteil daran haben, dass die Studienzeit die bis jetzt beste Zeit in meinem Leben ist.

Für die Betreuung der Masterarbeit bedanke ich mich bei Ao.Univ.-Prof. Dr.phil. tit.Univ.-Prof. Norbert Bartelme.

Weiterer Dank gilt Herrn Univ.-Prof. Mag. Dr.rer.nat. Gottfried Kirchengast und Herrn DI Fuchsberger vom Wegener Center für die Ermöglichung eines Direktzugriffes auf benötigte Daten.

Außerdem bedanke ich mich bei Herrn Dr. Alexander Podesser für die Bereitstellung der ZAMG Testdaten.

1 EINLEITUNG

“In the next century, planet earth will do an electronic skin. It will use the Internet as a scaffold to support and transmit its sensations. This skin is already being stitched together. It consists of millions of embedded electronic measuring devices: thermostats, pressure gauges, pollution detectors, cameras, microphones, glucose sensors, EKGs, electroencephalographs. These will probe and monitor cities and endangered species, the atmosphere, our ships, highways and fleets of trucks, our conversations, our bodies – even our dreams.”

(Gross, 1999).

Um dieser bereits im Jahre 1999 von Neil Gross umfassend formulierten Vision gerecht zu werden, bedarf es einer steigenden Anzahl von Sensornetzwerk-Implementierung, die insbesondere raumbezogenen Daten zur Verfügung stellen. Besonders wichtig ist ein hoher Grad an Interoperabilität zwischen den Sensornetzwerken, auf Daten- als auch Serviceebene. Heute produzieren Sensoren in meteorologischen Messstationen, Mobiltelefonen, GPS-Empfängern, Autos u.v.m. Daten, die zunehmend in Echtzeit analysiert werden. Die meisten bestehenden Sensornetzwerke sind derzeit für den eigenen Bedarf in abgeschlossenen, monolithischen Systemen eingerichtet, wodurch ein Datenaustausch mit anderen Netzwerken nur sehr eingeschränkt möglich ist. Die größten Herausforderungen zur Realisierung der „digitalen Haut“ der Erde sind die Integration heterogener Daten aus verschiedensten Datenquellen und der Austausch von Informationen über System- und Domänengrenzen hinweg. Neue Geo-Sensornetzwerke enthalten die Eigenschaften *Interoperabilität*, *Skalierbarkeit* und *Intelligenz*. *Interoperabilität* ist die Eigenschaft der Kommunikation verschiedenster Sensoren untereinander bzw. das gemeinsame Erzeugen von Ergebnissen durch Verwendung heterogener Daten. Mit *Skalierbarkeit* ist die einfache Integration neuer Sensoren in ein bestehendes System gemeint, ohne aufwendige Veränderungen am System selbst vornehmen zu müssen. *Intelligenz* ist die Fähigkeit der Sensoren bis zu einem gewissen Grad autonom zu „denken“, z.B. durch Datenfilterung nach vorgegebenen Kriterien oder mit autonomen Softwareagenten. (Resch, 2010).

Aufgrund des Fortschritts in der Computertechnik und ständig steigender Rechnerleistungen werden immer komplexere und genauere Modelle zur Abbildung und Prognose

1 Einleitung

dynamischer Prozesse entwickelt. Um mittels Klimamodellen höhere räumliche und zeitliche Auflösungen zu erzielen oder größere geographische Bereiche abzudecken, werden größere Datenmengen als Modellinput benötigt. Eine einfache Integration und Kombination der Daten in ein übergeordnetes System ist aufgrund fehlender Standards und den damit einhergehenden unterschiedlichen Datenformaten, räumlichen Bezugssystemen, Übertragungsprotokollen, etc. nicht möglich oder mit hohem Aufwand verbunden.

Die *Sensor Web Enablement* (SWE) Spezifikationen und deren Datenformate wurden vom *Open Geospatial Consortium* (OGC) entwickelt, um jegliche Art von Sensoren mittels Standards übers Internet auffindbar, abfragbar und steuerbar zu machen (Resch, 2010).

1.1 ZIELSETZUNG

Ziel der Arbeit ist eine detaillierte Betrachtung und Beschreibung der wichtigsten Standards und Spezifikationen der komplexen generischen Datenformaten und Schnittstellen der SWE-Initiative.

Relevante Forschungsprojekte und Implementierungen zu Geo-Sensoren und Geo-Sensor-Netzwerken sollen unter besonderer Berücksichtigung von verwendeten Standards dargestellt werden. Ein weiterer Fokus wird auf die Möglichkeit einer Echtzeit-Integration bzw. -Analyse von heterogenen Sensordaten gelegt.

Der Hauptteil der Arbeit besteht aus der Anwendung des *Sensor Observation Services* (SOS) der SWE Initiative, basierend auf Echtzeitdaten des Klimastationsnetzes der Region Feldbach (WegenerNet) sowie auf Testdaten des österreichweiten meteorologischen Messnetzes der Zentralanstalt für Meteorologie und Geodynamik (ZAMG). Ziel des praktischen Teils ist die Illustration des Zugriffs auf heterogene Sensordaten. Durch Darstellung beispielhafter Ergebnisse der Implementierungen sollen die Vorteile der Verwendung von OGC Standards aufgezeigt werden. Des Weiteren soll der Zugriff mittels ausgewählter Client-Applikationen die Möglichkeiten einer Darstellung und Analyse der Daten in Echtzeit aufzeigen.

1 Einleitung

1.2 METHODIK UND AUFBAU DER ARBEIT

Eine umfassende Literaturrecherche dient als Basis zur Bearbeitung der Thematik und der Beschreibung der wichtigsten Standards und Spezifikationen zu den komplexen generischen Datenformaten und Schnittstellen der SWE-Initiative.

Im Kapitel *Grundlagen* (Kapitel 2) werden nach der Vorstellung des *Open Geospatial Consortium* (OGC), welches weltweit offene Standards auf Basis frei verfügbarer Spezifikationen bereitstellt, die Hauptkomponenten der SWE Initiative dargestellt. Genau beschrieben werden dabei die einzelnen standardisierten Schnittstellen und die in den Schnittstellen zur Messdatenkapselung verwendeten XML-Schemata.

Der inhaltliche Schwerpunkt des Kapitels *Geo-Sensor Webs* (Kapitel 3) widmet sich bestehenden Forschungsprojekten und Implementierungen zu Geo-Sensoren und Geo-Sensor Webs. Im ersten Teil wird die Forschungseinrichtung *Senseable City Lab* des *Massachusetts Institute of Technology* (MIT) vorgestellt, welche anhand von heterogenen Sensoren in Städten erzeugte Datenströme analysiert. Die folgenden Unterkapitel erläutern die Systemarchitektur und standardisierte Services aktueller Implementierungen von interoperablen Geo-Sensor Webs. Die Beispiele zeigen vor allem hinsichtlich Interoperabilität weit fortgeschrittene Projekte.

Kapitel 4 schildert die Umsetzung des WegenerNet SOS und des SOS angewendet auf aufbereitete Testdaten der ZAMG. Zu Beginn wird die von der Firma 52°North entwickelte Java basierte SOS Applikation erklärt. Nach Darstellung der nicht standardisierten Struktur beider Messstationsnetze wird auf die technische Umsetzung der Java basierten SOS Applikation näher eingegangen. Im Falle der WegenerNet SOS Implementierung erzeugt eine in C++ entwickelte Applikation automatisch benötigte Metadatenfiles zu den vorhandenen Sensoren. Die Messwerte der meteorologischen Daten des Wegner Centers werden zyklisch abgefragt und in eine SOS Datenbank gespeichert. Eine weitere C++ Applikation bringt die meteorologischen Testdaten der ZAMG Messstationen automatisch in die richtige SOS Struktur. Am Ende jedes Unterkapitels zu den Implementierungen werden die Ergebnisse anhand ausgewählter Beispiele dargestellt und der Zugriff auf Echtzeitdaten bzw. Testdaten mittels Clients illustriert.

2 GRUNDLAGEN

Zu Beginn des folgenden Kapitels wird die Standardisierungseinrichtung *Open Geospatial Consortium* (OGC) kurz vorgestellt. Die Unterkapitel widmen sich der vom OGC ins Leben gerufenen *Sensor Web Enablement* (SWE) Initiative, die den Kern der vorliegenden Arbeit darstellt. Dabei werden ausgewählte Standards des SWE genau beschrieben.

2.1 OPEN GEOSPATIAL CONSORTIUM (OGC)

Das OGC wurde 1994 als *OpenGIS Consortium* gegründet und 2004 in *Open Geospatial Consortium* umbenannt. Derzeit gehören dem OGC 475 Mitglieder aus Regierungsorganisationen, privater Industrie und Universitäten an, die im Konsensverfahren weltweit öffentlich verfügbare Standards zu Schnittstellen entwickeln. Diese Schnittstellen ermöglichen Technologieentwicklern die Realisierung eines einfachen Zugangs zu komplexen räumlichen Informationen und Services durch verschiedenste Applikationen (OGC, 2013).

Die in einen Standardisierungsprozess mit einbezogenen Komitees sind in drei Bereiche unterteilt: Strategieentscheidungen werden im *Management Committee* (MC) getroffen, die technische Arbeit wird vom *Technical Committee* (TC) koordiniert und *Working Groups* (WGs) sowie *Special Interest Groups* (SIGs) sind für individuelle Spezifikationen zuständig. Die Vision des OGC ist eine Welt, in der jeder von geographischen Informationen und Services profitiert, welche über diverse Arten von Netzwerken, Applikationen oder Plattformen verfügbar gemacht worden sind. Das Arbeitsprogramm des OGC wird in folgende zwei Bereiche eingeteilt (Kresse & Fadaie, 2004, S. 210):

1. **Specification Program** – Das *Specification Program* wird vom technischen Komitee gelenkt und ist in die Bereiche *Abstract Specification* und *Implementation Specification* unterteilt. Die *Abstract Specification* umfasst den Aufbau, die Komponenten und die Funktionsweise eines dienstbasierten GIS im Sinne des OGC, während die *Implementation Specification* eine detaillierte Beschreibung für die Implementierung der Dienste liefert.

2 Grundlagen

2. **Interoperability Program** – Das *Interoperability Program* besteht aus einer Reihe von Initiativen zur Beschleunigung der Entwicklung und Akzeptanz von OGC-Spezifikationen.

2.2 SENSOR WEB ENABLEMENT (SWE)

Die SWE-Initiative des OGC dient der Gewährleistung der Nutzung von Sensordaten über System- und Domänengrenzen hinweg, indem diese über das Internet auffindbar, abfragbar und steuerbar gemacht werden. Dieser neue Geo-Sensor Web Ansatz zeichnet sich im Unterschied zu herkömmlichen, meist geschlossenen monolithischen Systemen durch einen hohen Grad an Interoperabilität, Skalierbarkeit und Intelligenz aus und ermöglicht eine Kombination von Messdaten aus verschiedenen Geo-Sensornetzwerken. (Resch, 2010).

Abbildung 1 zeigt ein über das Internet zugängliches Sensornetz mit räumlich verteilten Sensoren, die an verschiedenen Positionen Bedingungen wie Temperatur, Geräusche, Vibrationen, Druck, Bewegungen, Schadstoffe, etc. messen. Diese Sensoren können wie über das WEB mittels standardisierten Protokollen und APIs aufgesucht, abgefragt und gesteuert werden.

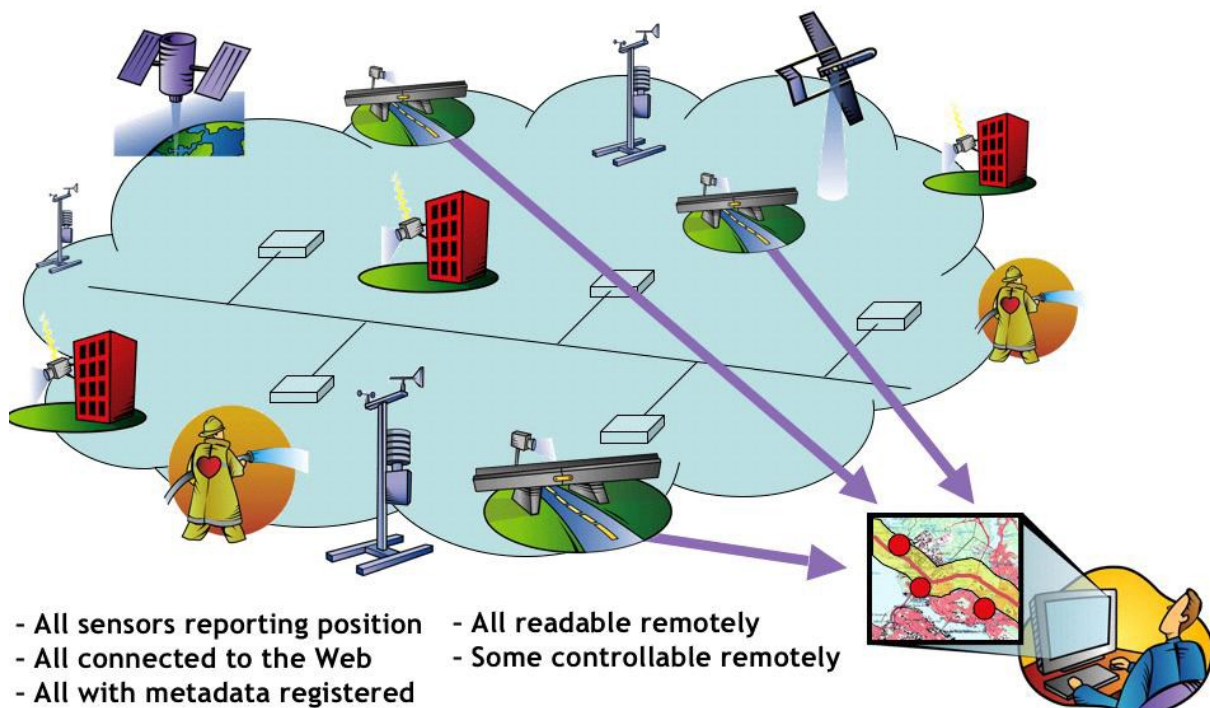


Abbildung 1: Konzept des Sensor Webs (Reed, et al., 2013, S. 5)

2 Grundlagen

Durch die Standards des SWE werden webbasiert über gemeinsame Schnittstellen und Metadatenkodierungen heterogene Daten auf einfache Art und Weise in eine Informationsinfrastruktur eingebunden und bilden damit ein Sensor-Web. Die entsprechenden Daten sind nicht nur Messwerte von Sensoren, sondern auch Ergebnisse von Simulationen oder numerischen Prozessen. Laut SWE Spezifikation müssen folgende Funktionen innerhalb eines Sensor-Webs gewährleistet sein (Reed, et al., 2013):

- Auffinden von Sensorsystemen, Beobachtungen und Prozessbeschreibungen
- Bestimmung der Fähigkeiten (*Capabilities*) eines Sensors und der Qualität der Messungen
- Zugriff auf Sensorparameter, die es einer Software ermöglichen Beobachtungen zu lokalisieren und zu prozessieren
- Abfragen von Beobachtungen in Echtzeit bzw. von Zeitreihendaten in normierter Kodierung
- Beauftragen von Sensoren, um Beobachtungen je nach Interesse zu akquirieren
- Abonnieren und Ausgeben von Alarmen in Abhängigkeit definierter Kriterien

Die Initiative beschreibt Sensorplattformen und Sensorbeobachtungen über ein XML-Schema und definiert Schnittstellen für Webservices wie beispielsweise eine Datenabfrage, eine Alarmierung bzw. eine Benachrichtigung von Benutzern oder Benutzergruppen. Abbildung 2 zeigt den Zusammenhang zwischen den Schritten eines wie oben beschriebenen Monitoring-Workflows und den Standards des SWE (Koch, et al., 2013, S. 261).

2 Grundlagen

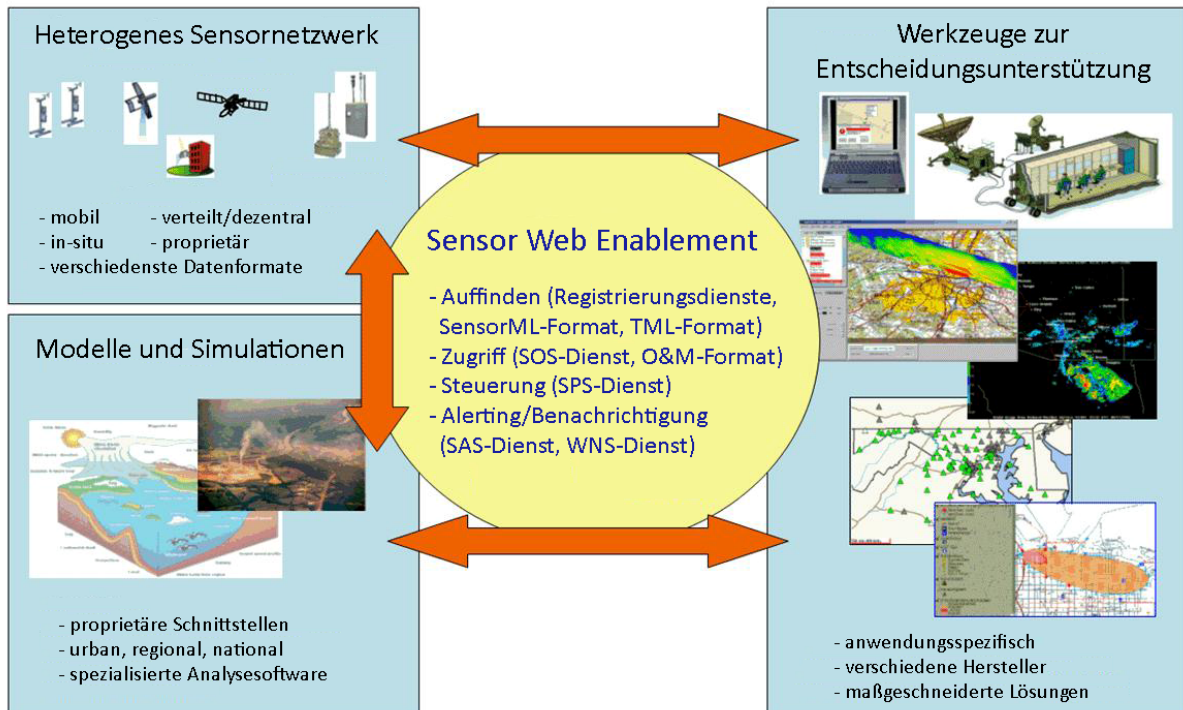


Abbildung 2: SWE-Standards - konzeptionelle Funktionsweise (Koch, et al. 2013 nach Botts et al. 2007)

Das SWE besteht aus folgenden Standards und *Best Practice*¹ Spezifikationen: (Open Geospatial Consortium, 2013):

1. **Observations & Measurements (O&M, Standard)** – Dieser Standard dient der Beschreibung von Beobachtungen und Messungen eines Sensors mittels Standard-Modellen und *Extensible Markup Language* (XML) Schemas. Die Beobachtungen und Messungen können sowohl in Echtzeit als auch als Zeitreihen vorliegen.
2. **Sensor Model Language (SensorML, Standard)** – Standard-Modelle und XML-Schemas beschreiben Sensorsysteme und Prozesse, die mit Sensorbeobachtungen in Verbindung stehen. Sie liefern zusätzlich Informationen, die zur Auffindung von Sensoren nötig sind. Des Weiteren werden Informationen zur Position und zur Prozessierbarkeit (Low-Level) der Sensorbeobachtungen sowie der bearbeitbaren Eigenschaften generiert.

¹ Best Practices bezeichnen bewährte, optimale Methoden oder Konzepte. OGC definiert unter anderem neue Konzepte, die potenziell Relevanz für die Zukunft haben, aber noch nicht als Standard freigegeben sind als Best Practice.

2 Grundlagen

3. **PUCK Protocol Standard (Standard)** – Die PUCK Protocol Spezifikation definiert ein standardisiertes Protokoll für Geräte, die mit Clients über ein RS232- oder Ethernet-Protokoll kommunizieren. Der Standard dient zum automatischen Speichern von Metadaten und stellt diese sowie weitere Informationen zum Gerät selbst dem Client zur Verfügung.
4. **Sensor Observations Service (SOS, Standard)** – Das SOS definiert ein standardisiertes Web Service Interface zum Anfragen, Filtern und Abrufen von Beobachtungen und Informationen zum Sensorsystem. Das Service vermittelt zwischen einem Client und einem Archiv mit Beobachtungen sowie einem Real-time Sensorkanal.
5. **Sensor Planning Service (SPS, Standard)** – Mittels SPS werden benutzergesteuert Akquisitionen und Beobachtungen abgefragt. Ein Web Service Interface bildet die Vermittlungsschicht zwischen einem Client und einem Managementsystem.
6. **Sensor Alert Service (SAS, Best Practice)** – Das SAS definiert ein standardisiertes Web Service Interface zum Abonnieren und Ausgeben von Alarmen.
7. **Web Notification Services (WNS, Best Practice)** – Das WNS definiert ein standardisiertes Web Service Interface zum asynchronen Übertragen von Benachrichtigung und Alarmen, die vom SAS, oder SPS generiert werden.

Diese Standards werden in den folgenden Kapiteln genauer beschrieben.

2.3 OBSERVATIONS & MEASUREMENTS (O&M)

Der O&M Standard spezifiziert ein konzeptionelles XML-Schema für *Observations* (Beobachtungen) und für sogenannte *Sampling Features*, die beide von Raum und Zeit abhängen (Cox, 2010, S. 1).

2.3.1 EIGENSCHAFTEN UND BEGRIFFE IM ZUSAMMENHANG MIT OBSERVATIONS

In diesem Kapitel werden die Begriffsdefinitionen und deren Bedeutung erläutert, die im Zusammenhang mit *Observations* verwendet werden (Cox, 2010; Bröring, et al., 2012).

Feature-Of-Interest:

Ein *Feature-of-Interest* repräsentiert ein Geobjekt der realen Welt, wie etwa eine meteorologische Messstation, welches Sensoren oder andere Prozeduren beinhaltet, die Ergebnisse durch *Observations* für Eigenschaften (*Properties*) liefern. Die in einem

2 Grundlagen

Anwendungsschema definierte Art eines *Feature-Of-Interest* legt die möglichen Eigenschaften des Geoobjekts fest. Die Werte der Eigenschaften sind z.B. der Name der Messstation, der Eigentümer oder Werte bestimmt durch eine *Observation*.

Observation:

Eine *Observation* bezeichnet jenen Vorgang, der einem Phänomen (*Phenomenon*) einen Wert, einen Begriff oder ein Symbol zuordnet. Zusätzlich wird der Erfassungszeitpunkt oder die Erfassungsperiode der Werte, Begriffe oder Symbole durch die *Observation* akquiriert. Liegt das Ergebnis als Quantität vor, so wird die *Observation* als Messung (*Measurement*) bezeichnet (Cox, 2010, S. 7).

Measurement:

Ein *Measurement* beschreibt Operationen zur Bestimmung des Werts einer Größe. Im Gegensatz zu *Observation* liefert ein *Measurement* nur eine numerische Größe (Cox, 2007).

Procedure:

Die Prozedur (*Procedure*), wie beispielsweise ein Sensor, ein Instrument, ein menschlicher Beobachter, ein Algorithmus, eine Wertschöpfungskette, eine Simulation, etc. erzeugt Werte, Begriffe oder Symbole einer *Observation* (Cox, 2010, S. 7).

Die *Procedure* beschreibt einen Prozess, der zur Erzeugung eines Ergebnisses benutzt wird.

Phenomenon:

Ein Phänomen stellt die Eigenschaft eines Geoobjekts dar. Die Ergebnisse einer *Observation* werden durch die *Procedure* dem Phänomen zugeordnet. Beispiele für Phänomene sind physikalische Größen wie Lufttemperatur, Niederschlag, Windgeschwindigkeit oder Luftfeuchtigkeit.

Observation Properties:

Eine *Observation* liefert das Ergebnis (*Result*) für eine Eigenschaft eines beobachteten Objekts (*Feature-Of-Interest*). Der Wert selbst wird durch die Prozedur erzeugt. *Observation Properties* sind Kontextinformationen und Metadaten für die Evaluierung, Interpretation und Benutzung der Ergebnisse, beschreiben also das *Phenomenon* (Cox, 2010, S. 7).

2 Grundlagen

Observation Location:

In vielen Fällen entspricht die *Location* der Sensorposition. Solange die Eigenschaft (*Property*) eines *Feature-Of-Interest* nicht die Position verändert, erfolgt die Verortung der Observations über das georeferenzierte *Feature-Of-Interest*. Schwieriger ist die Zuordnung der Position von Beobachtungen, die z.B. mittels mobilen Sensorplattformen wie Fernerkundungssatelliten akquiriert werden. Die Satellitendaten müssen prozessiert werden, bevor eine Geolokalisierung der Szene oder der Schwade möglich ist (Cox, 2010, S. 7-8).

Ein weiteres Beispiel für eine komplexe Bestimmung der Lage eines *Feature-Of-Interest* ist, wenn sich dieses auf eine Entität der realen Welt wie auf eine Region, ein Fahrzeug, ein Unwetter, etc. bezieht (Henson, et al., 2009).

2.3.2 KONZEPTIONELLES MODELL EINER BEOBACHTUNG

Die Kodierung der *Observations* und *Measurements* nach dem O&M Modell ermöglicht einen standardisierten Zugriff auf diese. Eine durch die *GetObservation* Operation (siehe Kapitel 2.6) abgefragte SOS Schnittstelle gibt die Antwort im O&M Format zurück.

Das konzeptionelle Modell zur Beschreibung einer *Observation* ist in Abbildung 3 als *Unified Modeling Language* (UML)-Diagramm dargestellt. Die Klasse *OM_Observation* ist eine Instanz der Metaklasse (<<metaclass>>) *GF_FeatureType*. *GF_FeatureType* repräsentiert den *Feature Typ*. *OM_Observation* unterstützt fünf Attribute, sechs Assoziationen und unterliegt vier Bedingungen, die im Folgenden beschrieben werden.

Attribute:

Das Attribut *resultTime* beschreibt die Zeit, zu der ein Ergebnis zur Verfügung steht, während die *phenomenonTime* den Zeitstempel des aktuellen Ergebnisses der Eigenschaft eines observierten Objektes (*Feature-of-Interest*) kennzeichnet. Für bestimmte *Observations* ist die *resultTime* gleich der *phenomenonTime*. Misst z.B. ein Thermometer die Temperatur und das Ergebnis ist augenblicklich verfügbar, so ist der Zeitstempel des Messwertes gleich der *resultTime* und der *phenomenonTime*. Werden mehrere Messpunkte über einen bestimmten Zeitraum erfasst und durch einen Algorithmus zur Kurvenglättung neue Werte erzeugt, dann entspricht der Zeitpunkt eines Messwertes aus den Rohdaten der *phenomenonTime*. Der Zeitpunkt eines neu generierten Wertes ist der *resultTime* gleichzusetzen. Die *validTime* kennzeichnet die Periode in der die Ergebnisse Gültigkeit haben (optionaler Parameter, der häufig für Anwendungen mit Vorhersagen verwendet wird). Mittels dem optionalen Para-

2 Grundlagen

meter *resultQuality* wird die Qualität der durch eine *Procedure* erzeugten *Observations* beschrieben. Das Attribut *parameter* soll ereignisspezifische Parameter beschreiben. Werden z.B. zur Bestimmung der Wasserqualität Proben aus einem Brunnen in verschiedenen Tiefen genommen, so kann die Beobachtung zusätzlich mit dem Parameter „Brunnentiefe“ beschrieben werden (Cox, 2010, S. 10-11).

Assoziationen:

Die Assoziation *Domain* ordnet *OM_Observation* die Klasse *GFI_Feature* zu. Das Feature entspricht entweder einem Objekt der realen Welt, dessen Eigenschaften unter Beobachtung stehen, oder es werden mit dem Feature wie in Kapitel 2.3.4 beschrieben, Objekte der realen Welt abgetastet. *Phenomenon* verlinkt *OM_Observation* mit *GFI_PropertyType*. Das Ergebnis einer *OM_Observation* ist die Schätzung des Wertes einer Eigenschaft. *Range* ordnet der *OM_Observation* den durch eine Prozedur erzeugten Wert zu. *OM_Process* erzeugt die Ergebnisse. Instanzen dieser Klasse sind Sensoren, menschliche Beobachter, Simulationen, Algorithmen, etc. Die Klasse *MD_Metadata* liefert die beschreibenden Daten (Cox, 2010, S. 12).

Constraints:

Das Modell definiert Einschränkungen bezüglich der *observedProperties*, *Procedures*, *resultTypes* und *Parameter*. So werden beispielsweise durch die in einem *Application Schema* definierte Art eines *Feature-Of-Interest* die Eigenschaften des Geoobjekts festgelegt (Cox, 2010, S. 13).

2 Grundlagen

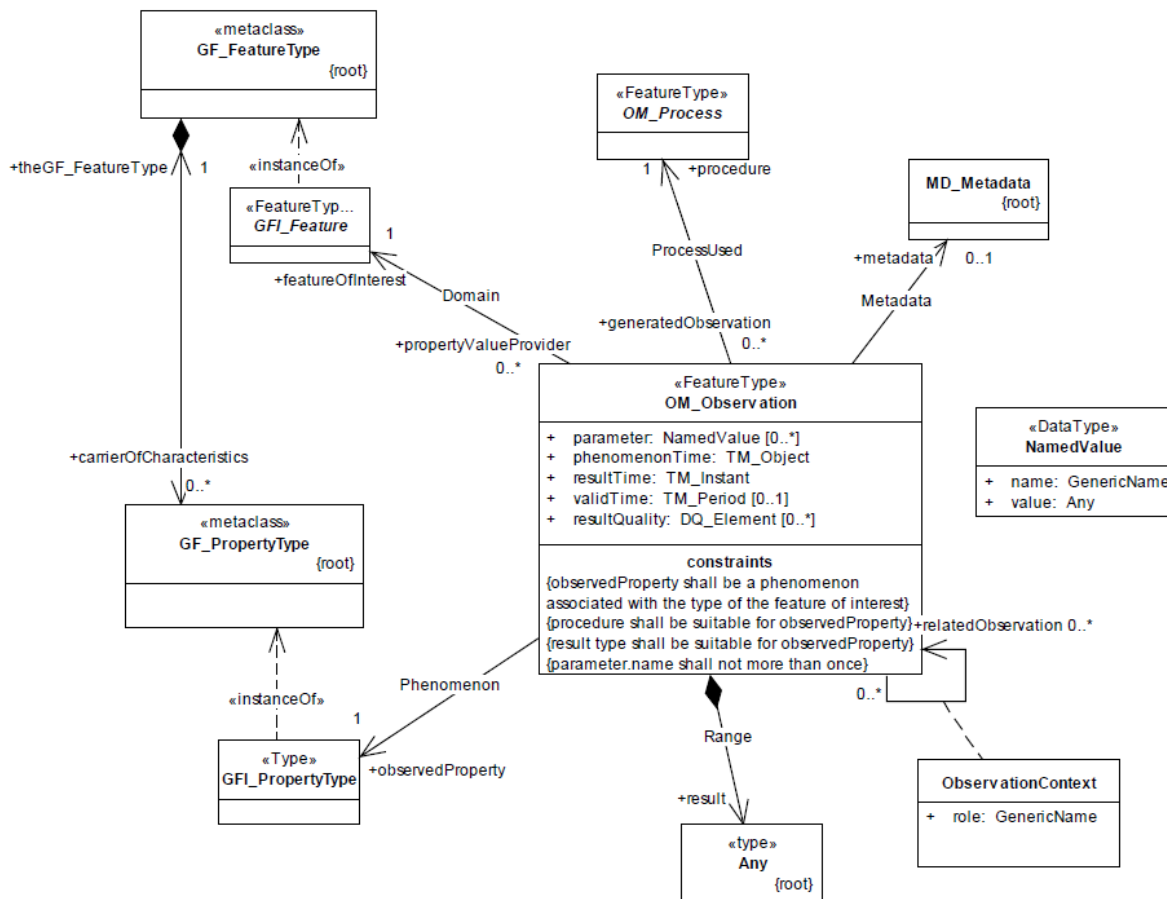


Abbildung 3: Beschreibung einer Beobachtung als UML-Diagramm (Cox, 2010, S. 10)

2.3.3 KLASSIFIKATION DER OBSERVATIONS AUFGRUND DES ERGEBNISSES

Observations können auf Basis der Ergebnisart in folgende zwei Gruppen eingeteilt werden (Cox, 2010, S. 15-16):

- **Observations mit konstantem Ergebnis** – *Observations*, deren Ergebnisse aus einem (*single-valued*) oder mehreren (*multi-valued*) Werten bestehen. Liefert das Ergebnis ein Datenfeld, so müssen die Werte dieses Feldes über Raum und Zeit konstant sein.
- **Observations mit variablem Ergebnis** – Einzelne *Observations* generieren Datenfelder, deren Werte sich über Raum und/oder Zeit ändern.

2 Grundlagen

2.3.3.1 OBSERVATIONS MIT KONSTANTEM ERGEBNIS

Bei *Observations* mit konstantem Ergebnis ist während der Dauer einer Observation die Eigenschaft eines *Feature-of-Interest* unveränderlich. Ist das Resultat ein Skalar, so wird die mathematische Größe (z.B. Temperatur, relative Feuchtigkeit, Länge, Geschwindigkeit, etc.) nur durch einen Zahlenwert charakterisiert. Im Falle eines Vektors (z.B. Komponenten eines Windgeschwindigkeitsvektors), Tensors oder Arrays (z.B. Bänder eines Satellitenbilds) besteht das Ergebnis aus mehreren Elementen. Eine Übersicht der Ergebnistypen ist in Tabelle 1 dargestellt. (Cox, 2010, S. 15-18).

Tabelle 1: Ergebnistypen für Observations mit konstantem Ergebnis (nach Cox, 2010, S. 18)

Observation-Klasse	Ergebnistyp	Beispiel
OM_Measurement	Measure	Eine Messung der „Temperatur“ (Property-Type) der „Station Mariazell“ (Feature-Of-Interest) mit dem „Temperaturfühler PT100“ (Procedure) lieferte das Ergebnis „23,5 °C“.
OM_CategoryObservation	ScopedName	Eine Kategorie-Beobachtung eines „Taxon ² “ (Property-Type) der „Probe 123“ (Feature-Of-Interest) durchgeführt von „Gerald Schweiger“ (Procedure) lieferte das Ergebnis „Eucalyptus caesia“
OM_CountObservation	Integer	Eine Count-Observation des „Verkehrs“ der „Autobahn XY“ (Feature-Of-Interest) mittels „viacount ³ “ (Procedure) lieferte als Ergebnis „5433“
OM_TruthObservation	Boolean	Eine <i>Truth-Observation</i> bezüglich der „Anwesenheit“ (Property-Type) eines „Einbrechers“ (Feature-Of-Interest) mittels „CCTV-Videoüberwachungsanlage“ (Procedure) lieferte den Wert „false“
OM_TemporalObservation	TM_Object	Eine temporale Observation der „Zeitdauer“ des „100 m Sprints von Usain Bolt“ mittels „Stoppuhr“ betrug „9,6 s“.
OM_GeometryObservation	GM_Object	Eine Geometry-Observation des „Umfangs“ (Property-Type) für das „Grundstück 987“ (Feature-Of-Interest) mit der „Feldvermessung GHJ“ (Procedure) ergab(...Beschreibung des Polygons)
OM_ComplexObservation	Record (ISO/TS 19103:2005)	Eine komplexe Observation der „Zusammensetzung der Hauptelemente“ (Property-Type) der Probe „h8j“ (Feature-Of-Interest) mittels „ICP-MS ⁴ “ (Procedure) ergab (...Array der Elementverhältnisse)

² Als Taxon wird in der Biologie eine bestimmte Kategorie von Lebewesen bezeichnet, die durch einen Namen gekennzeichnet sind.

³ VIACOUNT ist ein vom Unternehmen „via traffic control“ entwickeltest Verkehrszählgerät.

⁴ ICP-MS steht für inductively-coupled-plasma mass-spectrometry und ermöglicht die Bestimmung einer Vielzahl von Elementen. Es wird u. a. zur Spurenanalyse von Schwermetallen wie Quecksilber, Blei oder Cadmium benutzt.

2 Grundlagen

2.3.3.2 OBSERVATIONS MIT VARIABLEM ERGEBNIS

Bei Feature-Typen mit funktionalen Beziehungen hängen die durch *Observations* erzeugten Werte einer Eigenschaft von bestimmten veränderlichen Parametern ab. Tabelle 2 zeigt Observation-Klassen und deren Ergebnistypen von *Observations* mit variablem Ergebnis.

Tabelle 2: Ergebnistypen für Observations mit variablen Ergebnis (Cox, 2010, S. 19)

Observation-Klasse	Ergebnistyp
OM_DiscreteCoverageObservation	CV_DiscreteCoverage
OM_DiscretePointCoverageObservation	CV_DiscretePointCoverage
OM_DiscreteTimeSeriesObservation	CVT_DiscreteTimeInstantCoverage

DiscreteCoverageObservation:

Ändert sich der Wert eines Phänomens in Abhängigkeit der Zeit oder seiner geographischen Position, dann wird der Observation-Typ als *Coverage* betrachtet. Beispiele für Features mit Bezug zu *Coverages* sind der Ozean, die Oberfläche der Atmosphäre oder der Erde.

Laut DIN 19123 vereinen *Coverages* Konzepte der vektoriiellen und rasterorientierten Darstellung (Norm DIN EN ISO 19123, 2007).

DiscretePointCoverageObservation:

Ein Temperaturfeld einer Region kann z.B. als *DiscretePointCoverage* betrachtet werden, wenn Messstationen, die in einem regelmäßigen Raster angebracht sind, in einer bestimmten zeitlichen Auflösung Temperaturmesswerte erfassen. Abbildung 4 zeigt das Stationsnetz WegenerNet, bestehend aus 151 Messstationen, welche unter anderem die Hauptparameter Lufttemperatur, relative Luftfeuchtigkeit und Niederschlag messen. Dieses Netz kann als *DiscretePointCoverage* mit diskreten Punkten, die einen bestimmten räumlichen Bereich abdecken, modelliert werden. Das Datenarray hat dann drei Dimensionen, mit den Hauptparametern zu jedem Punkt (Cox, 2010, S. 19).

2 Grundlagen

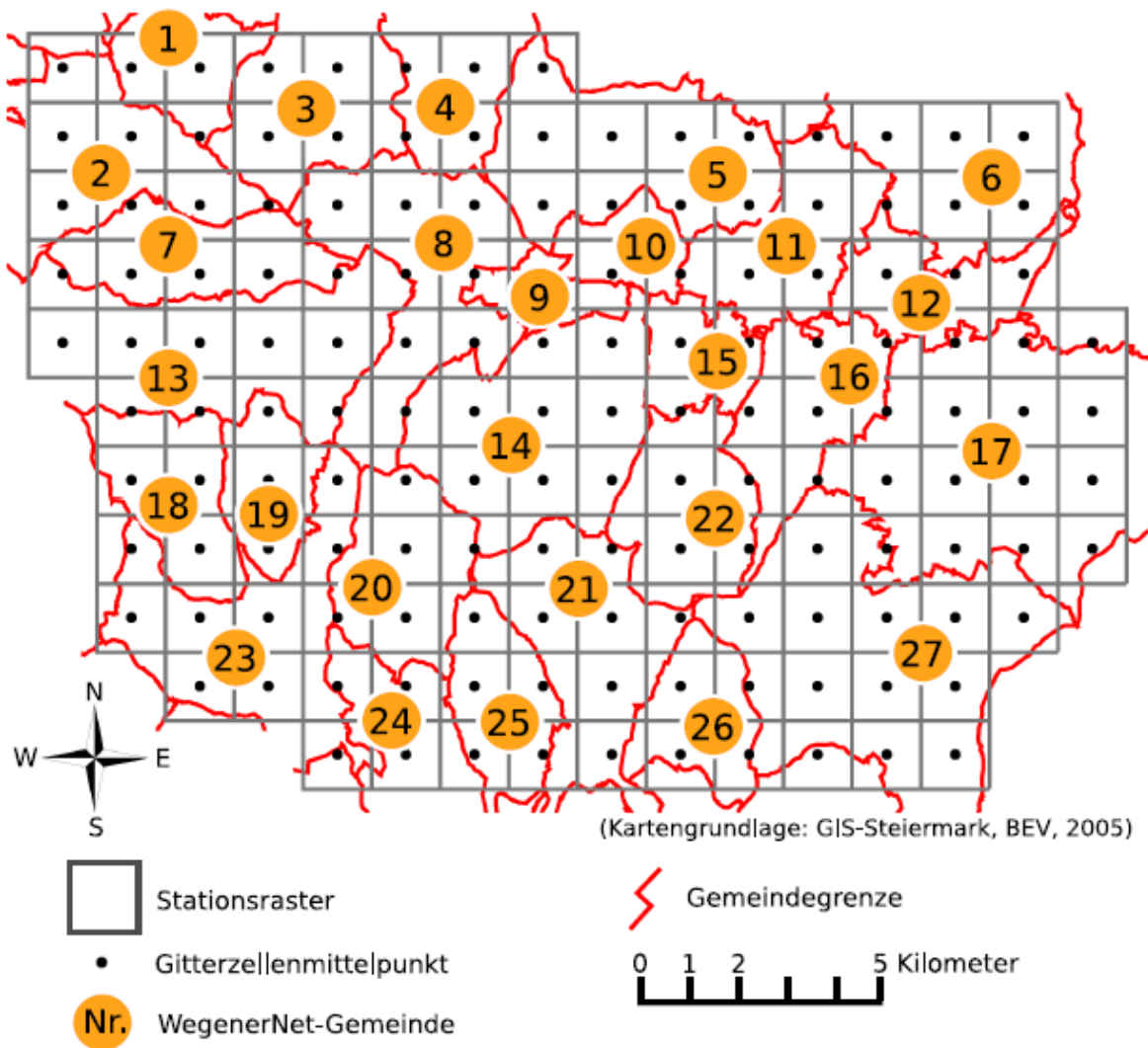


Abbildung 4: Stationsraster des WegenerNet (ca. 1,4km x 1,4km Gitter) mit den insgesamt 27 WegenerNet-Gemeinden
(Kabas, 2011, S. 25)

DiscreteTimeSeriesObservation:

Messtationen zur Überwachung der Luft- oder Wasserqualität liefern skalare zeitabhängige Konzentrations-, Ph-, Chlorid-Werte etc. Solche Zeitreihen werden für gewöhnlich als diskretes *Coverage* beschrieben (Cox, 2010, S. 19).

2.3.4 SAMPLING FEATURE

Proben (*Samplings*) an repräsentativen Stellen werden dann genommen, wenn es nicht möglich ist ein Objekt in seiner Gesamtheit zu untersuchen. Die Prozedur, um von den *Samplings* auf die Eigenschaft des gesamten Objekts zu schließen, ist abhängig vom Sample-Design. Das gesamte zu untersuchende Objekt wird als *ultimate Feature-Of-Interest*

2 Grundlagen

bezeichnet, während das mittels Probe untersuchte Objekt das *proximate Feature-Of-Interest* darstellt. Um beispielsweise auf den chemischen Zustand eines Grundwasserkörpers zu schließen, werden an einer oder mehreren Stellen Proben aus Bohrlöchern entnommen. Die Annäherung an die Eigenschaften des gesamten Grundwasserkörpers erfolgt durch ein Modell (Cox, 2010, S. 20).

Wird über eine *Procedure* die Eigenschaft eines *Feature-Of-Interest* indirekt bestimmt, so bezeichnet man das *Observed Property* als *Proxy*. In der Fernerkundung ist die observierte Eigenschaft ein *Proxy*. Die Observation beinhaltet je Kanal die reflektierte Strahlung in einem bestimmten Wellenlängenbereich, untersucht soll jedoch z.B. die Vegetation werden. Da das *ultimate Feature-Of-Interest* nicht die Eigenschaft des *Proxys* beinhaltet, muss ein *Proxy-Feature* durch geeignete prozessierende Schritte einbezogen werden (Cox, 2010, S. 21).

Tabelle 3 zeigt die *Sampling Feature* Klassen, welche auf Basis der Geometrie kategorisiert werden. Beispielsweise wird die Messung der Schadstoffkonzentration mittels mobilen Luftgütesensoren der Klasse *SF_SamplingCurve* zugeordnet, während ein vertikales Profil einer Meeresströmung, ermittelt durch einen Ultraschall-Doppler-Profil-Strömungsmesser ein *SF_SamplingSurface Feature* darstellt. Eine Zeitreihe eines dreidimensionalen Windfeldes wird als *SF_SamplingSolid Feature* modelliert.

Tabelle 3: Geometrietypen räumlicher *Sampling Features* (Cox, 2010, S. 27)

Sampling Feature	Geometrie
SF_SamplingPoint	GM_Point
SF_SamplingCurve	GM_Curve
SF_SamplingSurface	GM_Surface
SF_SamplingSolid	GM_Solid

2.4 SENSOR MODEL LANGUAGE (SENSORML)

Die *Sensor Model Language* (SensorML) dient der Beschreibung von Sensoren und stellt ein *Extensible Markup Language* (XML) Schema zur Definition von geometrischen, dynamischen und messungsbezogenen Charakteristika zur Verfügung. Durch das Modell sollen Sensoren im Web auffindbar sowie nutzbar gemacht werden und Sensorbeobachtungen erforscht werden können (OSGeo, 2013; Resch, 2010).

2.4.1 ALLGEMEINES

Die SensorML-Spezifikation definiert Prozesse und Prozesskomponenten in Bezug auf Messungen. Ein Sensor stellt einen Prozess dar, der Phänomene observiert und observierte Werte zurückgibt. Prozesskomponenten sind Eingänge, Ausgänge, Parameter und Prozessmethoden. Die nach der Spezifikation erzeugten Metadaten dienen der Beschreibung, Identifikation und Klassifikation eines Prozesses, als auch zur Beschreibung seiner Charakteristik (z.B. zeitliche oder räumliche Verfügbarkeit). Während des Verfassungszeitraums der vorliegenden Masterarbeit arbeitete das OGC am Upgrade des SensorML Standards der Version 1.0.0 auf 2.0. Zusätzliche Funktionalitäten der neuen Version sind beispielsweise Vererbungsmechanismen, die Redundanzen und die Größe von Sensormetadatenbeschreibungen reduzieren. Eine weitere Vision ist die on-the-fly Integration von Sensoren in ein beliebiges Sensor-Web (Bröring, et al., 2011).

SensorML beschreibt des weiteren Sensoren und Sensorsysteme, stellt Sensor- und Prozessinformationen bereit, unterstützt das Prozessieren und die Analyse von *Sensor-Observations*, fördert die räumliche Lokalisierung von observierten Werten, beschreibt den Leistungsumfang des Prozesses sowie den Prozess zu Erzeugung einer Observation, usw.

Prinzipiell unterscheidet man zwischen zwei grundlegenden Sensortypen (Botts & Robin, 2007):

1. In-Situ-Sensoren zur Messung der physikalischen Eigenschaft eines Mediums in seiner unmittelbaren Umgebung und
2. Remote-Sensoren (z.B. Fernerkundungssensoren), die physikalischen Eigenschaften der *Features-of-Interest* aus einer bestimmten Entfernung detektieren.

2 Grundlagen

Je nach Prozess-Modell werden die Sensoren in verschiedenen Komplexitätsstufen beschrieben. Ausgehend von einfachen Strukturen werden durch Aggregation komplexere Zusammensetzungen gebildet.

In der SensorML werden nicht zusammengesetzte und untrennbare Prozesse als *atomic processes* bezeichnet. *Detektoren (Components)* sind „einfache Geräte“, die aus einem Eingangssignal eines oder mehrere Ausgangssignale erzeugen. Die Software solcher Geräte arbeitet mit simplen Modellen (*Process Models*), die beispielsweise mittels Eichkurven kalibriert werden können. *Simple atomic processes* ohne Zeit-Raum-Bezug sind als *Process Models* definiert, während *Components* echte physikalische Größen messende Geräte kennzeichnen (z.B. Detektoren und Wandler).

Einfache Prozesse können über Kopplung der Ein- und Ausgänge zu komplexeren Prozessen zusammengefasst werden. Existieren diese nicht in der realen Welt, so werden sie als *Process Chains* (Prozessketten) bezeichnet. Prozesse mit Zeit- und Raumbezug werden *System* genannt. Typische *Systeme* bestehen aus eine Ansammlung von Sensoren, die zeitlich und räumlich in Bezug zueinander stehen (Botts & Robin, 2007, S. 33).

2.4.2 BEZIEHUNG DES SENSORS ZU EINER PLATTFORM

Ein Sensorsystem besteht aus Sensoren, die auf einer Plattform montiert sind. Beispielweise dient der Fernerkundungssatellit „LANDSAT 7“ als Trägerplattform des Sensors ETM+ (*Enhanced Thematic Mapper +*), welcher wiederum aus 100 Spektraldetektoren besteht, die durch einen oszillierenden Spiegel die reflektierte elektromagnetische Strahlung der Erde erfassen. Position und Orientierung des Satelliten selbst werden durch eigene Detektoren auf der Plattform bestimmt.

Innerhalb der SensorML-Spezifikation werden Sensor und Plattform als getrennte Entitäten behandelt. Ein Sensor besteht aus einem oder mehreren Detektoren. Die Plattform mit ihren Sensoren wird als System modelliert, welches entweder stationär ist oder seine Position in Bezug auf ein geodätisches Referenzsystem ändert. Die Beziehung der einzelnen Sensoren untereinander und zur Plattform selbst kann auch durch lokale Koordinatensysteme hergestellt werden. Die Verortung erfolgt anschließend über erdräumliche Koordinatensysteme. In Abbildung 5 ist ein Satellit dargestellt, dessen oranges lokales Koordinatensystem des Sensors gegenüber dem schwarzen Koordinatensystem der bewegenden Plattform verschoben und verdreht ist (Botts & Robin, 2007, S. 33-34).

2 Grundlagen

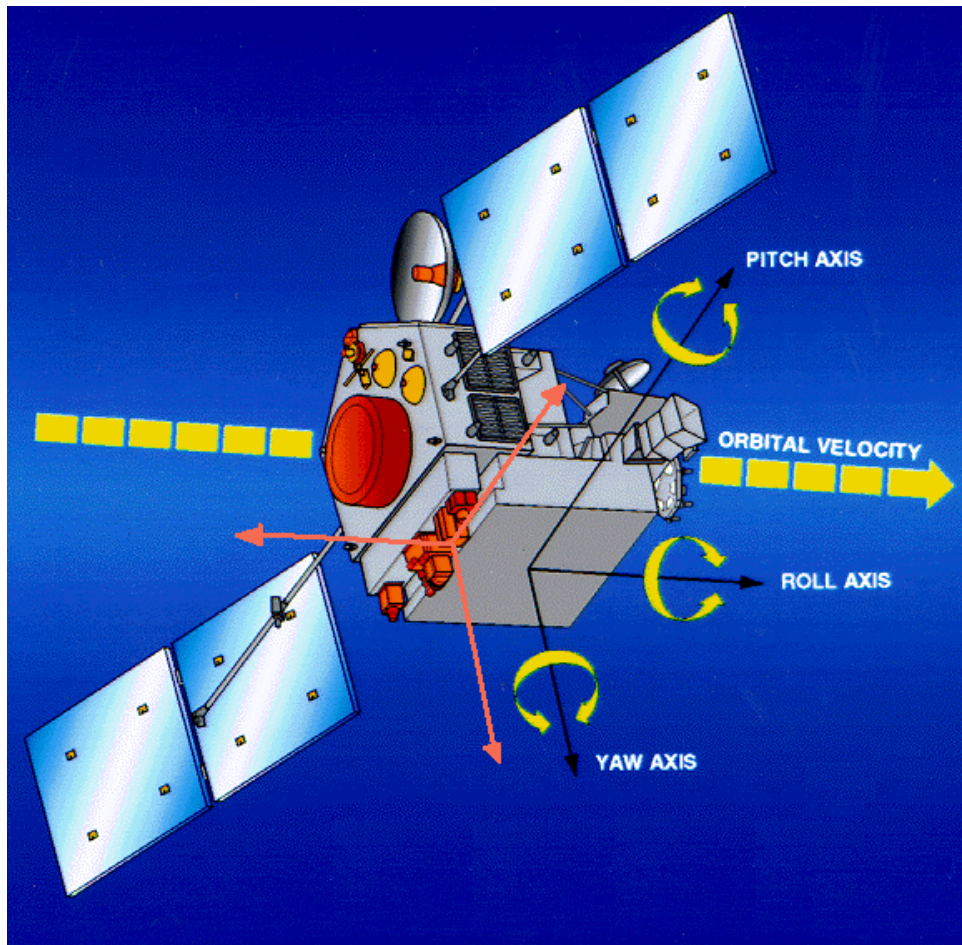


Abbildung 5: Beziehung des lokalen Koordinatensystems des Sensors (Orange) zum lokalen Koordinatensystem der Plattform (schwarz) (Botts & Robin, 2007, S. 34)

Tabelle 4 stellt Möglichkeiten zur Klassifikation von Sensorsystemen in Abhängigkeit des Sensortyps und der Dynamik der Plattform dar. So kann ein In-Situ-Sensor zur Messung der Wasserqualität an einer bestimmten Stelle fest montiert sein (Fixed In-Situ), oder es werden Messungen mit einer Sonde mobil etwa auf einem Boot durchgeführt (Mobil In-Situ). Beispiele für Remote-Sensoren sind fest montierte Doppler-Radar-Geräte zur Messung von Wegstrecken und Geschwindigkeiten oder mobile Airborne *Laser detection and ranging* (LIDAR) Systeme montiert auf Flugzeugen, u.a. zur Erstellung von digitalen Oberflächen- oder Geländemodellen.

2 Grundlagen

Tabelle 4: Beziehung zwischen In-Situ und Remote-Sensoren auf mobilen oder fixen Plattformen (Botts & Robin, 2007, S. 35)

	Sensortyp	In-Situ	Remote
Mobilität			
Fixed		Stationäre O2 Probe	Doppler-Radar-Station
Mobil		Salzgehaltprobe	Airborne LIDAR

2.4.3 KONZEPTIONELLES MODELL FÜR PROZESSE

Abbildung 6 zeigt die konzeptionelle Modellierung eines Prozesses bestehend aus nicht physikalischen und/oder physikalischen Subprozessen. Alle als Prozess modellierten Komponenten generieren unter Berücksichtigung der Eingänge und Parameter mittels Methoden Prozessaugänge (Abbildung 6, *AbstractProcess*).

Wie in Kapitel 2.4.1 beschrieben, unterscheidet die SensorML-Spezifikation zwischen nicht physikalischen und physikalischen Prozessen, die nachfolgend in Bezug auf das konzeptionelle Modell, dargestellt in Abbildung 6, erörtert werden. Nicht physikalische Prozesse sind beispielsweise mathematische Operationen, während Detektoren, Aktuatoren, Sensorsysteme, etc. physikalischen Prozessen zugeordnet werden.

Nicht physikalische Prozesse:

Das *ProcessModel* definiert nicht physikalische Prozesse, die innerhalb von Prozessketten benützt werden können. Die *ProcessMethod* eines *ProcessModels* erzeugt mittels eines Algorithmus Ausgänge in Abhängigkeit der Eingänge und Parameter, wodurch das Verhalten und die Schnittstelle des *ProcessModels* festgelegt werden. *ProcessChain* vereint mehrere Prozesskomponenten zu einem eigenen Prozess, um ein gewünschtes Ergebnis zu erzielen (Botts & Robin, 2007, S. 57).

Physikalische Prozesse:

Um physikalische Prozesse, die abhängig von Raum und Zeit sind, modellieren zu können, werden zeitliche und räumliche Referenzsysteme sowie die Beziehung dieser untereinander definiert (*spatialReferenceFrame*, *temporalReferenceFrame*). Ein einzelner Prozess, der nicht mehr in Subeinheiten unterteilt werden kann, wird als *Component* modelliert. Außerdem können mehrere zusammenhängende Subprozesse als einziger unteilbarer Prozess be-

2 Grundlagen

trachtet werden. Einzelne *Components* können Teil einer Prozesskette sein (Botts & Robin, 2007, S. 59-60).

Ein *System* besteht aus nicht physikalischen sowie physikalischen Prozessen, die miteinander interagieren und in Abhängigkeit der Systemeingänge und Parameter Ausgänge erzeugen. Eine Wetterstation kann z.B. aus mehreren Sensoren bestehen, die gemeinsam in bestimmten Zeitintervallen den Zustand der Atmosphäre bestimmen. Einen nicht physikalischen Prozess würde dabei die Berechnung des Windchills im Gerät selbst darstellen (Botts & Robin, 2007, S. 60).

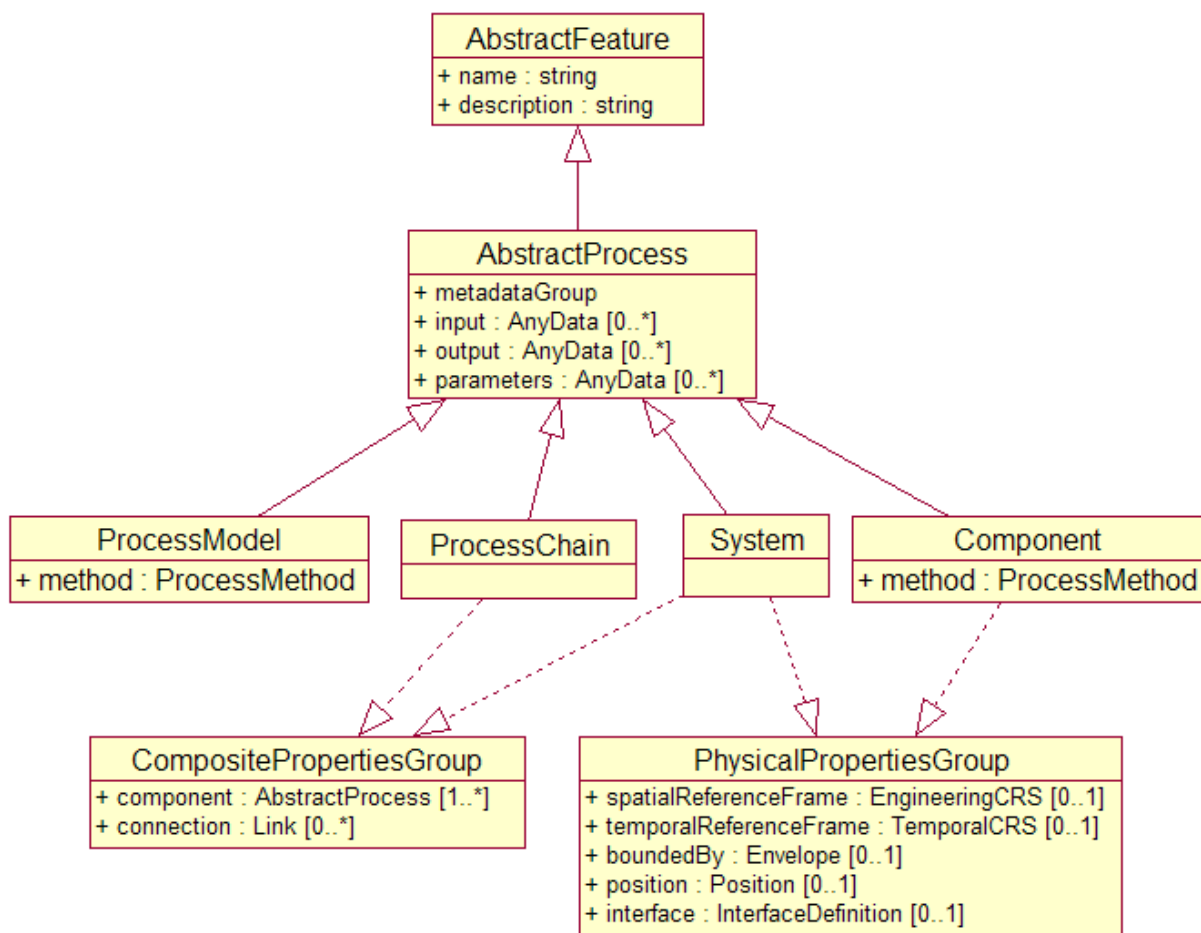


Abbildung 6: Konzeptionelles Modell für Prozesse (Botts & Robin, 2007, S. 55)

2.5 PUCK PROTOCOL STANDARD

Sensornetzwerke bestehen aus untereinander vernetzten Sensoren oder anderen Geräten sowie Hostmaschinen, welche die Sensoren kontrollieren und prozessieren. Die Sensoren selbst sind oftmals physikalisch an Datenlogger oder speicherprogrammierbaren Steuerungen angeschlossen und kommunizieren über Protokolle (z.B. RS232, Ethernet) mit einem Leitrechner. Der Leitrechner wiederum akquiriert die Daten der Sensoren über eine an das Protokoll der Datenlogger angepasste Driver-Software.

Der Hostrechner bzw. die Driver-Software benötigt Informationen über das Instrument und zum Communication-Port eines Rechners, an dem z.B. Datenlogger angeschlossen sind. Des Weiteren müssen die Befehle des Protokolls, und Metadaten, welche die Geräte und die gelieferten wissenschaftlichen Daten beschreiben, bekannt sein. Die meisten Sensoren oder Geräte bieten diese Informationen nicht automatisch an. Daher muss ein Benutzer die Software der Geräte und der Hostrechner manuell z.B. über Konfigurationsdateien oder mittels Benutzeroberfläche parametrisieren. Standards wie das OGC SWE oder der *Smart Transducer Interface Standard IEEE 1451* des *Institute of Electrical and Electronic Engineers* (IEEE) sollen das Integrieren von verschiedensten Instrumenten mit minimalem Aufwand und hoher Verfügbarkeit ermöglichen (O'Reilly, 2012, S. x).

Transducer sind Umsetzer, die sowohl Sensoren als auch Aktuatoren sein können. Als Aktuatoren bezeichnet man Bauteile, die ein elektrisches Signal in eine mechanische Bewegung oder andere physikalische Größe umsetzen und somit regulierend in ein System eingreifen. *Smart Transducers* sind Bauteile, die gleichzeitig physikalische Größen erfassen (Sensor), die Signalaufbereitung und –wandlung (Aktuator) übernehmen sowie Verarbeitungsfunktionen und Datenspeicher zur Verfügung stellen. Solche Funktionen sollen die Integration eines Umsetzers in ein Netzwerk vereinfachen (Reimer & Kampe, 2008).

Trotz dieser Standards müssen Softwarekomponenten inklusive Gerätetreiber, Web-Server und Dokumente mit Metadaten zur Beschreibung der Geräte manuell installiert werden. Das PUCK Protokoll definiert ein standardisiertes Geräteprotokoll um Metadaten automatisch in den PUCK-Speicher eines Gerätes zu schreiben und diese sowie andere Informationen abzufragen. Derartige Informationen liegen im OGC SWE SensorML Format oder als IEEE 1451 TEDS (elektronisches Datenblatt) vor und können vom Hostcomputer für eine automatische Installation der Komponenten genutzt werden (O'Reilly, 2012, S. xi).

2 Grundlagen

Abbildung 7 illustriert das Zusammenspiel der SWE Komponenten. Aus verschiedensten Gründen muss ein SWE manchmal direkt auf den Sensor zugreifen. So befiehlt beispielsweise das SPS einer an das Netz angeschlossenen Kamera in eine andere Richtung zu schwenken. Eine Vielzahl an Sensoren ist nicht direkt internetfähig und kommuniziert über spezielle Protokolle mit einer Hostmaschine (z.B. über die serielle RS232 Schnittstelle). Das SWE Service verschafft sich über eine am Hostrechner installierte Driver-Software, welche alle Anfragen und Befehle des Services übersetzt, Zugang zu den Sensoren und zum SensorML Dokument. Zusätzlich transformiert der Treiber die Rohdaten in ein O&M Objekt. Der Sensor-Treiber und das SensorML Dokument können direkt am Sensor selbst gespeichert und beim Anschließen des Sensors an ein Netzwerk dem Hostrechner automatisch über das PUCK Protokoll zur Verfügung gestellt werden (O'Reilly, 2012, S. xi).

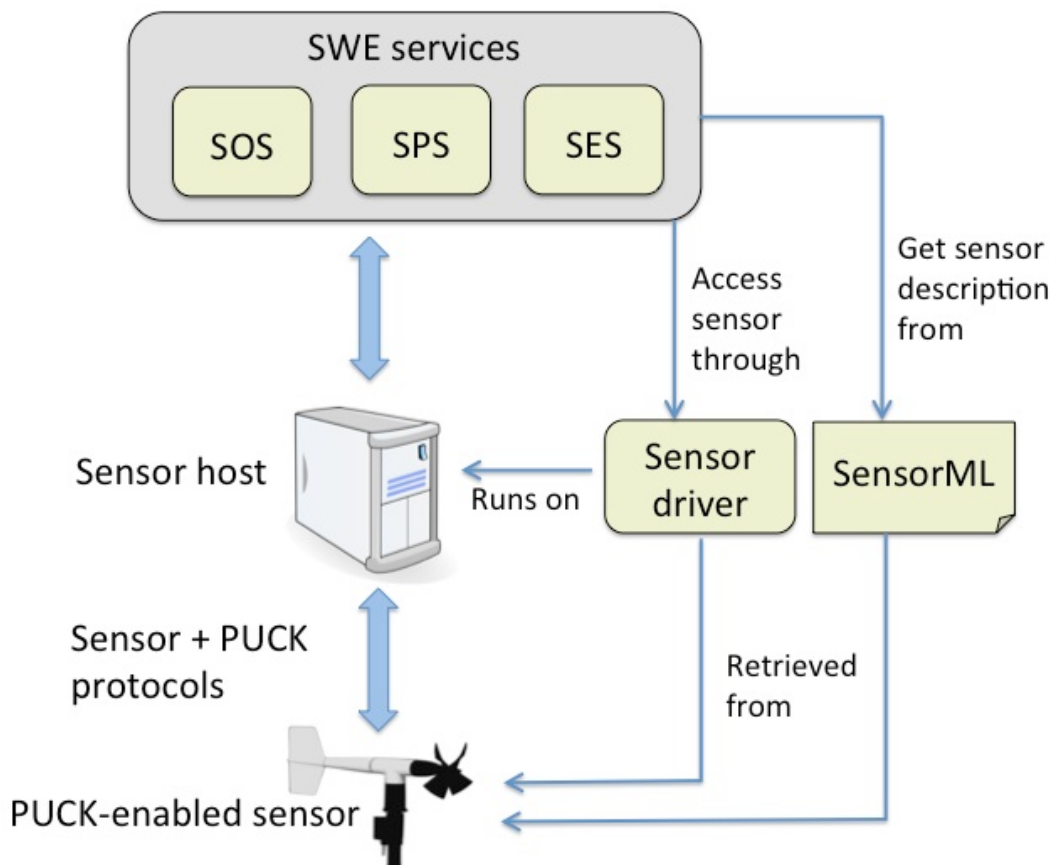


Abbildung 7: OGC SWE Services, SensorML und PUCK Protokoll (O'Reilly, 2012, S. xii)

2.6 SENSOR OBSERVATION SERVICE (SOS)

Das *Sensor Observation Service* (SOS) ist eine standardisierte webbasierte Schnittstelle mit der Metadaten und *Observations* von heterogenen Sensorsystemen verwaltet und abgefragt werden, wobei die in den vorhergehenden Kapiteln beschriebenen Standards O&M, PUCK Protocol und SensorML in der Spezifikation genutzt werden.

In einer SOS Schnittstelle werden Abfrageresultate im O&M Format modelliert und der SensorML Standard zur Beschreibung von Sensoren und Sensorsystemen genutzt. Das Service vermittelt zwischen einem Client und einem Sensordatenarchiv oder Real-Time-Sensorsystem. Die Schnittstelle des SOS ermöglicht einen Zugriff auf stationäre oder mobile heterogene Sensoren, wobei die Datenerfassung In-Situ oder remote erfolgen kann (vgl. Tabelle 4). Alle angeforderten Sensordaten werden als *Observations* zurückgegeben. Am 20. April 2012 wurde auf Basis der alten Version „SOS 1.0.0“ die teilweise veränderte und weiterentwickelte Version „SOS 2.0“ veröffentlicht (Bröring, et al., 2011, S. 2669).

2.6.1 SOS 2.0 MODELLÜBERBLICK

Die SOS 2.0 Spezifikation besteht aus dem *Core* (Kern), der aus vorhandenen OGC-Spezifikationen und ISO-Standards aufgebaut ist. Die Kernoperationen (*Core Operations*) sind bei einer SOS Implementierungen zwingend umzusetzen und können mittels *Extensions* funktional aufgewertet werden. Profile (*Profils*) dienen zur Anpassung des SOS an spezielle Themenbereiche (z.B. Hydrologie, Ozeanographie) sowie zur Erhöhung der Interoperabilität in den einzelnen Fachbereichen (Abbildung 8).

2 Grundlagen

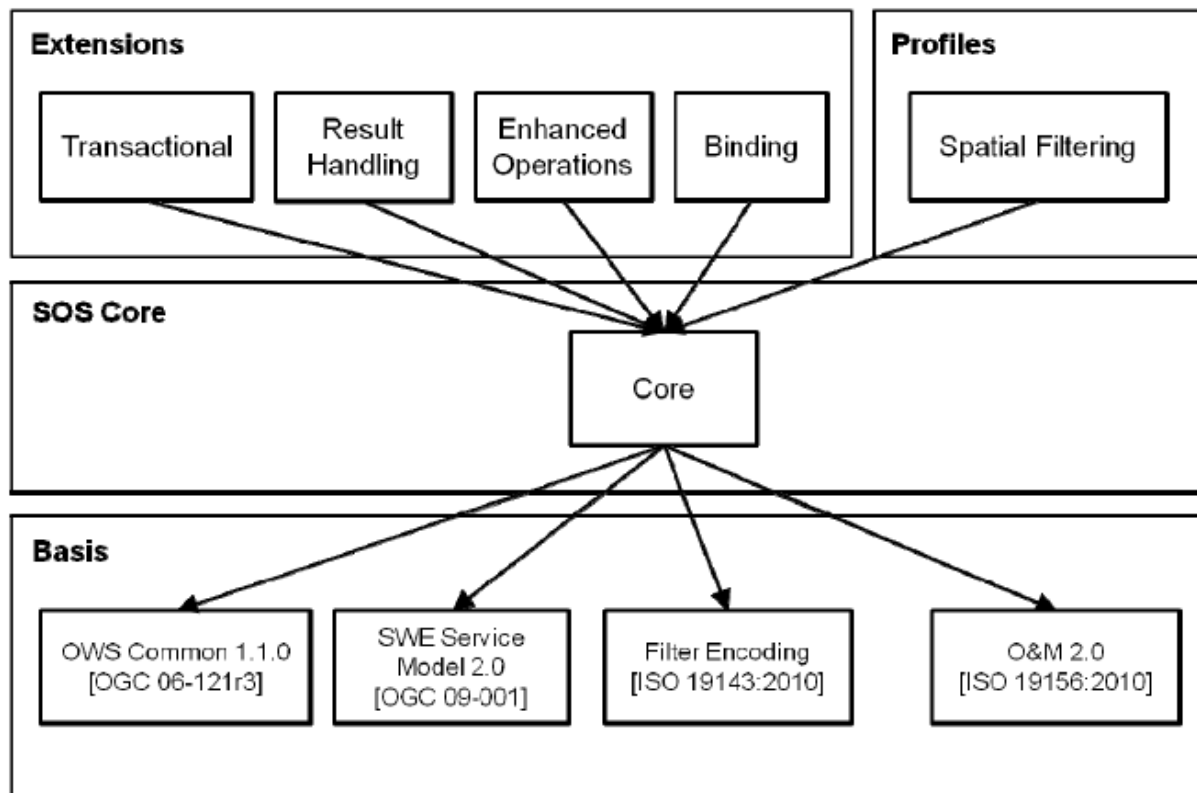


Abbildung 8: Abhängigkeiten zwischen Core, Extensions und der OGC Spezifikationsbasis (Bröring, et al., 2012, S. 10)

2.6.2 OBSERVATION OFFERING

Offerings gruppieren die durch eine Prozedur (z.B. ein Sensorsystem) erzeugten *Observations*. Ein Sensorsystem kann ein einzelnes Thermometer sein oder aus mehreren Sensoren bestehen. Ein Beispiel ist eine meteorologische Messstation, die mit mehreren Sensoren den augenblicklichen Zustand der Atmosphäre an einer bestimmten Position ermittelt (vgl. Kapitel 0). Eine weitere mögliche Gruppierung der *Observations* ist die Zusammenfassung anhand spezifischer thematischer Kriterien, etwa die Zusammenfassung aller *Observations* einer Region, eines Landes, oder einer Stadt (Open Geospatial Consortium, Inc., 2011).

Im Gegensatz zur SOS 1.0.0 Spezifikation erlaubt die neue SOS 2.0 Spezifikation die Zuordnung von nur einer Prozedur zu einem *Offering*. Dadurch wird die Mehrdeutigkeit des SOS 1.0.0 der Gruppierung von *Observations* von mehreren Prozeduren gelöst (Bröring, et al., 2012, S. x).

2 Grundlagen

2.6.3 KERN-OPERATIONEN (CORE OPERATIONS)

Wird eine Implementierung nach der SOS Spezifikation umgesetzt, so sind die Kernoperationen *GetCapabilities*, *DescribeSensor* und *GetObservation* zwingend umzusetzen.

GetCapabilities liefert Metadaten zum Service sowie Informationen zu den tatsächlich umgesetzten Operationen, *DescribeSensor* bietet die Möglichkeit, Metadaten zu einem bestimmten Sensor oder Sensorsystem abzufragen und mittels *GetObservation* wird direkt auf die *Observations* zugegriffen. In der Operation *GetObservation* können optional Methoden zum Filtern räumlicher, zeitlicher oder thematischer Daten realisiert werden. Die *GetCapabilities* Anfragen und Antworten folgen dem *OWS Common Standard* [OGC 06-121r3], alle anderen Anfragen (*ExtensibleRequest*) und Antworten (*ExtensibleResponse*) basieren auf dem *SWE Service Model Standard* [OGC 09-001].

2.6.3.1 GETCAPABILITIES OPERATION

Ein Request mittels einer *GetCapabilities* Operation besteht aus zwingenden und optionalen Parametern. Der *Service-Typ-Identifizier* (z.B. „SOS“ bei einer Anfrage an ein *Sensor Observation Service*) und der Operationsname (z.B. *GetCapabilities*) sind bei einer derartigen Anfrage zwingend anzugeben. Optionale Parameter sind *AcceptVersions* (z.B. SOS 1.0.0 oder SOS 2.0), *Sections*, *updateSequence* und *AcceptFormats*. Über die *Sections*-Parameter wird definiert, welche Metadaten zum Service zurückgegeben werden. *AcceptFormats* legt das Format der Antwort des Servers fest (Whiteside, 2007, S. 12 ff.).

Das SOS liefert als Antwort eine in XML kodierte Datei mit Metadaten zur angefragten Instanz und zu den gelieferten Daten selbst. Abbildung 9 zeigt die Datentypen der *GetCapabilities* Operation. *Contents* liefert Metadaten zu den *Observations* und *Prozeduren* die von der jeweiligen SOS-Instanz verwaltet werden, *Extension* ist ein Container, der Elemente, die durch die Extension-Spezifikation definiert sind enthält. Durch die *filterCapabilities* werden mit Metadaten die unterstützen Filtermöglichkeiten beschrieben. Der Bereich *ServiceIdentification* liefert Metadaten zum Service selbst, wie den Namen des SOS, Schlüsselwörter, etc. Die Sektion *ServiceProvider* schließt Informationen zur Identifizierung des Serviceanbieters und *OperationsMetadata* ein und listet die vom Server unterstützten Request-Typen auf (Bröring, et al., 2012, S. 20 ff.).

2 Grundlagen

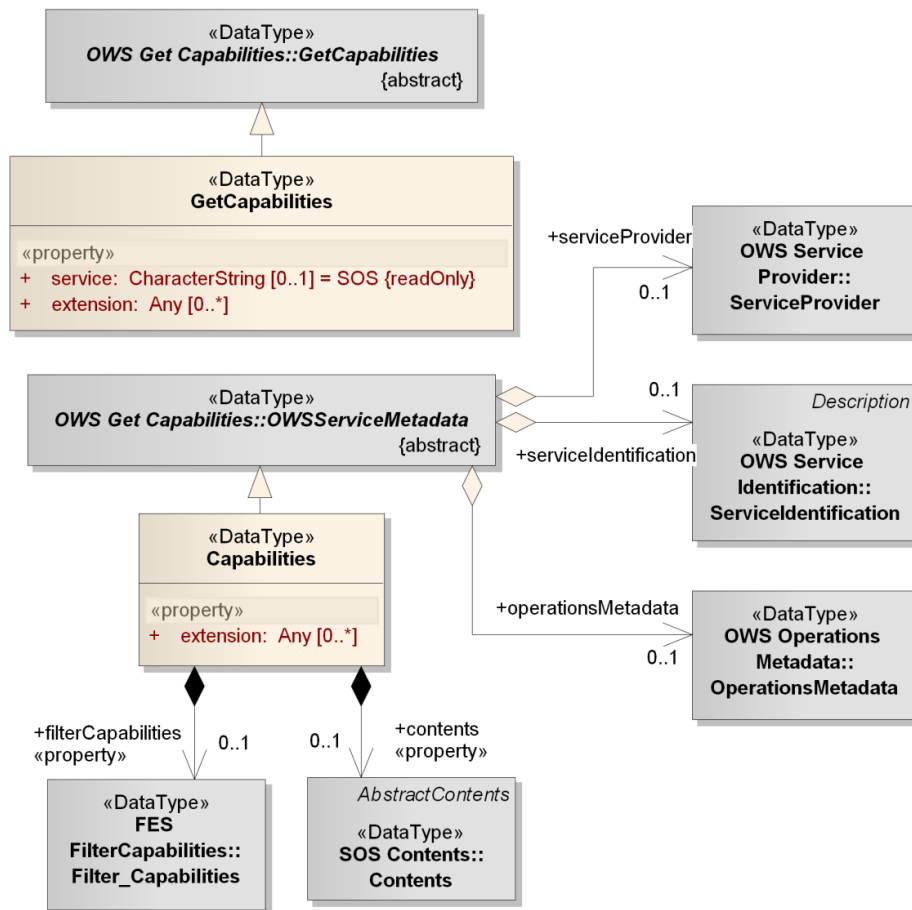


Abbildung 9: Konzeptionelles Modell der *GetCapabilities* Operation (Bröring, et al., 2012, S. 18)

2.6.3.2 DESCRIBESENSOR OPERATION

Die *DescribeSensor* Operation dient zum Abfragen von detaillierten Informationen zu Sensoren oder Prozeduren. Als Request-Parameter werden unter anderem das gewünschte Format der Antwort (z.B. *SensorML*, Kapitel 0) und die Version einer SOS Instanz übergeben (Echterhoff, 2011, S. 42).

Wird vom Server die Antwort im *SensorML* Format gefordert, so besteht das Dokument aus Metadaten zum Sensor oder Sensorsystem, wie z.B. die letzte gemessene Position des Sensors, eine Liste mit den Prozesseingängen und -ausgängen und eine Beschreibung der Komponenten des Sensorsystems.

2 Grundlagen

2.6.3.3 GETOBSERVATION OPERATION

Die *GetObservation* Operation dient zum Abfragen eines SOS und stellt Operationen in einer nach der O&M Spezifikation strukturierten Art und Weise bereit. Durch Parameter, die mit einer Anfrage an einen SOS Server übergeben werden, können die *Observations* gefiltert werden (Bröring, et al., 2012, S. 29 ff.).

Die Filterung einer Observation kann nach den unten dargestellten Eigenschaften erfolgen. Die Eigenschaften selbst können mit logischen Operationen verknüpft werden (Bröring, et al., 2012, S. 31-32):

- Procedure: Pointer auf *Observations* einer Prozedur
- Offering: Pointer auf ein *ObservationOffering*.
- ObservedProperty: Pointer auf *Observations* eines *ObservedProperty*.
- TemporalFilter: definiert einen zeitlichen Filter (z.B. Abfrage aller *Observations* die innerhalb einer bestimmten Periode akquiriert wurden)
- FeatureOfInterest: Pointer auf *Observations* eines *Feature-Of-Interest*
- SpatialFilter: definiert einen räumlichen Filter (z.B. Abfrage aller *Observations* innerhalb eines Polygons mit bestimmten Koordinaten)
- ResponseFormat: Abfrage aller *Observations* mit bestimmtem Format

Beispiel 1 zeigt eine *GetObservation* Abfrage, die alle *Observations* des gesamten Zeitraums und räumlichen Bereiches des *Feature-Of-Interest* „weatherstation_in_my_backyard“, welches das *observedProperty* „temperature“ mit der Prozedur „thermometerX“ oder „thermometerY“ misst, zurückgibt.

Beispiel 1: *GetObservation* Request (Bröring, et al., 2012, S. 31)

```
GetObservation( featureOfInterest:= weatherstation_in_my_backyard  
                AND observedProperty: = temperature  
                AND procedure: = thermometerX OR anemometerY )
```

2 Grundlagen

2.6.4 EXTENSIONS

Eine SOS-Implementierung kann durch den Einsatz von optionalen *Extensions* um zusätzliche Operationen erweitert werden. Die *Enhanced Operations Extension* bietet weitere Funktionen zum Abfragen der *Feature-Of-Interest* oder *Observations*, mit der *Transactional Extension* können Daten einer Datenbasis verändert werden, und die *Result Handling Extension* ermöglicht das direkte Einfügen oder Abfragen von Ergebnissen.

Enhanced Operations Extension:

Die Operationen der *Enhanced Operations Extension* sind *GetFeatureOfInterest* und *GetObservationById*. *GetFeatureOfInterest* dient zum Abfragen von Metadaten zu *Feature-Of-Interest*. Rückgabewerte sind beispielweise der Geometrietyt oder die Position (siehe Kapitel 4.3.3.1). Die *GetObservationById* Operation erlaubt dem Client die Abfrage von *Observations* mittels ID durch Übergabe eines Pointers auf die gewünschte Observation.

Transactional Extension:

Zu *Transactional Extension* Operationen zählen *InsertionCapabilities Section*, *InsertSensor*, *DeleteSensor* und *InsertObservation*. *InsertionCapabilities Section* liefert für das Einfügen von Daten Informationen zu den vom SOS unterstützten Feature-Typen, Observation-Typen und Result-Typen an den Client. Mit der *InsertSensor* Operation können neue Sensorsysteme im Service registriert und mittels *DeleteSensor* gelöscht werden (Bröring, et al., 2012, S. 47 ff.).

Result Handling Extension:

Die *Result Handling Extension* umfasst die *InsertTemplate*, die *InsertResult*, die *GetResultTemplate* und die *GetResult* Operation. Die *InsertResult* Operation ermöglicht dem Client ein Template für Ergebniswerte am Server zu erstellen. Ergebnisse, die mit dem Template übereinstimmen, können in das SOS eingefügt werden. Mit der *InsertResult* Operation werden nur die Ergebnisse einer Observation ohne Metadaten in die Datenbasis geschrieben. Diese Operation kann dann genutzt werden, wenn viele *Observations* aus denselben Metadaten bestehen. Durch das Einfügen der Ergebnisse alleine werden geringere Datenmengen übertragen, wodurch sich die Datenübertragungszeit verringert. *GetResultTemplate* liefert Informationen zur Struktur und Kodierung der Ergebnisse, die mit *InsertTemplate* erzeugt wurden. *GetResult* überträgt lediglich das Ergebnis der *Observations* vom Server zum Client (Bröring, et al., 2012, S. 57 ff.).

2 Grundlagen

Spatial Filtering Profile:

Eine Geometrie einer *Observation*, die beispielsweise einen bestimmten Bereich abtastet, kann dann abgefragt werden, wenn ein *Spatial Filtering Profile* am Server realisiert ist (Bröring, et al., 2012, S. 76).

2.7 ABLAUF ZUM ABFRAGEN ODER EINFÜGEN VON OBSERVATIONS ODER ERGEBNISSEN

Bestimmte Informationen müssen der Request-Operation als Parameter übergeben werden, um *Observations* oder Ergebnisse aus einer SOS Instanz abfragen und interpretieren bzw. in eine Datenbasis einfügen zu können. Der Ablauf zum Abfragen oder Einfügen von *Observations* oder Ergebnisse wird in den folgenden Kapiteln erklärt.

2.7.1 WORKFLOW ZUM ABFRAGEN EINER OBSERVATION

Abbildung 10 zeigt die Reihenfolge der Operationsaufrufe um *Observations* von einer SOS Instanz abzurufen und richtig interpretieren zu können. Optionale Operationen sind mit „opt“ gekennzeichnet. Um festzustellen, welche Daten, Prozeduren, Operationen, etc. die Instanz zur Verfügung stellt, sendet der Client eine *GetCapabilities* Anfrage an den Server. Benötigt der Client eine detaillierte Beschreibung der Prozeduren, so wird die *DescribeSensor* Operation ausgeführt. *GetFeatureOfInterest* stellt Information zu speziellen Features bereit, die der Client dem *GetObservation* Request als Parameter übergeben kann. *GetObservation* liefert schließlich die angefragten *Observations*.

2 Grundlagen

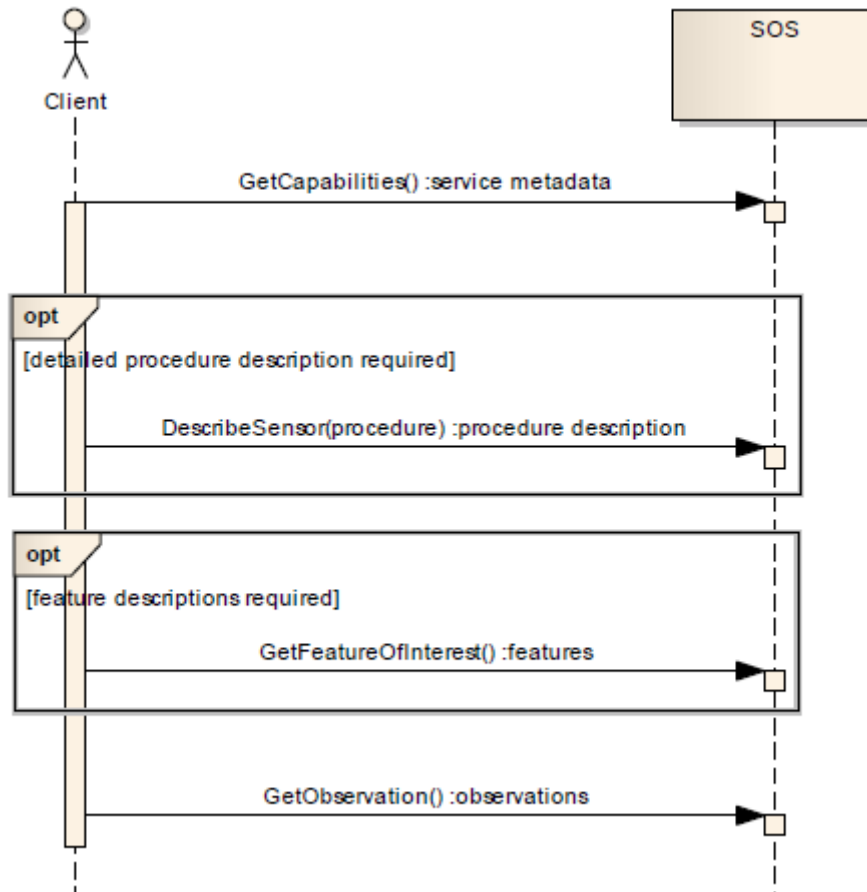


Abbildung 10: Sequenz der Operationen zum Abfragen einer Observation (Bröring, et al., 2012, S. 12)

2.7.2 WORKFLOW ZUM EINFÜGEN EINER OBSERVATION

Der Workflow zum Einfügen einer Observation ist in Abbildung 11 dargestellt. Der erste Schritt ist dabei das Ausführen der *GetCapabilities* Operation. Die vom Server gelieferte Antwort listet im *InsertionCapabilities* Bereich alle Observation- und Ergebnis-Typen, die von der Operation unterstützt werden, auf. Eine *Procedure*, für welche die *Observation* eingefügt werden soll, kann mit *InsertSensor* registriert werden, wenn diese noch nicht verfügbar ist. Die Operation *InsertObservation* fügt schließlich die *Observation* in die Datenbasis der SOS Instanz ein und ordnet diese, wenn gewünscht, einem *Offering* zu.

2 Grundlagen

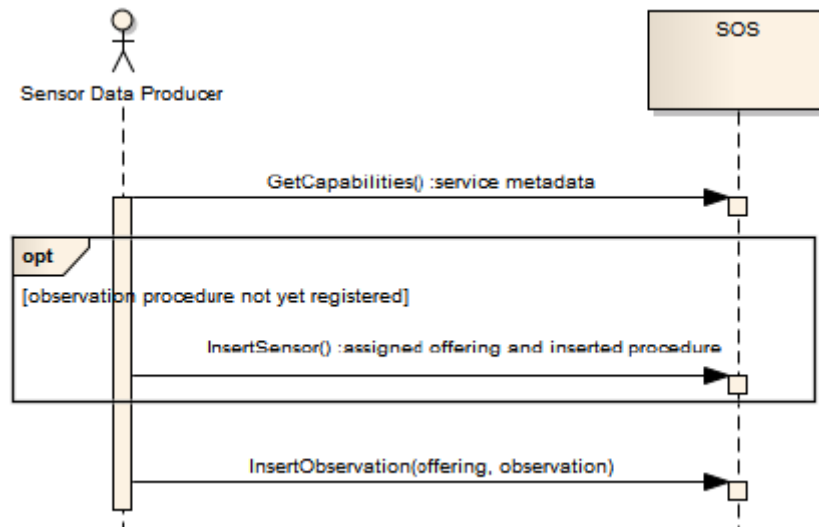


Abbildung 11: Sequenz der Operationen zum Einfügen einer Observation (Bröring, et al., 2012, S. 13)

2.7.3 WORKFLOW ZUM EINFÜGEN EINES ERGEBNISSES

Die Reihenfolge der Operationsaufrufe aus Abbildung 12 zeigt, dass vor Einfügen des Ergebnisses der Leistungsumfang der SOS Instanz mittels *GetCapabilities* abgefragt wird. *InsertSensor* registriert eine nicht vorhandene *Procedure*, welche die Ergebnisse erzeugt. Die *InsertResultTemplate* Operation sendet ein Template bestehend aus Metadaten der Observationen für die Ergebnisse und einer strukturierten Beschreibung dieser zum Server. Die erzeugten Sensordaten werden mit *InsertResult* ohne Metadaten zum SOS übertragen (Bröring, et al., 2012, S. 14).

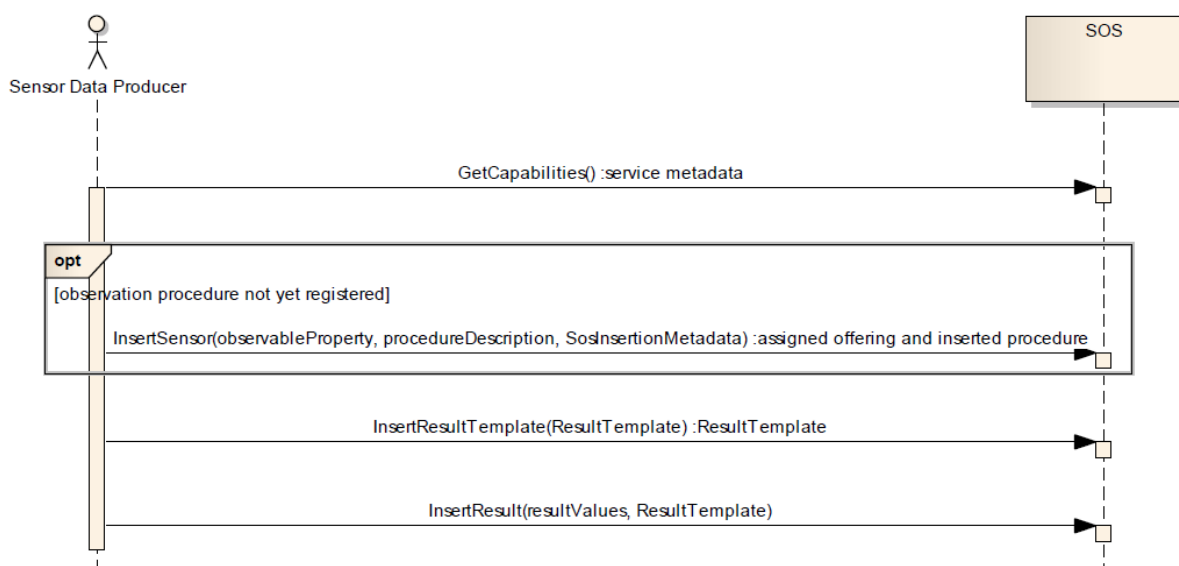


Abbildung 12: Sequenz der Operationen zum Einfügen eines Ergebnisses (Bröring, et al., 2012, S. 15)

2 Grundlagen

2.7.4 WORKFLOW ZUM ABFRAGEN EINES ERGEBNISSES

Die *GetResult* Operation wird angewendet, um kontinuierlich nur das Ergebnis ohne Meta-daten von *Observations* abzufragen. Die Kodierung und Struktur der Ergebnisse liefert die *GetResultTemplate* Operation (Bröring, et al., 2012, S. 15).



Abbildung 13: Sequenz der Operationen zum Abrufen eines Ergebnisses (Bröring, et al., 2012, S. 15)

2.8 SENSOR PLANNING SERVICE (SPS)

Die *Sensor Planning Service* Spezifikation dient unter anderem zum entfernten Steuern von Sensoren, also z.B. zur dynamischen Anpassung von Messintervallen oder zur Fernsteuerung von Überwachungskameras. Das SPS definiert webbasierte Schnittstellen, die benutzergesteuerte Akquisitionen und *Observations* ermöglichen und zielt auf die Automatisierung komplexer Informationsflüsse in großen Netzwerken ab (Resch, 2010, S. 7).

Im Gegensatz zum SOS, welches Informationen zu Beobachtungen und Sensorsystemen liefert, erlaubt das SPS durch Festlegung spezifischer Parameter eine Manipulation bestimmter Prozesse eines komplexen Systems. Welche Parameter zur Verfügung gestellt werden, wird vom Serviceanbieter festgelegt. Beispielsweise erlaubt eine SPS Schnittstelle die Veränderung des Zeitintervalls in der eine Webcam Bilder aufnimmt. Eine komplexere Anwendung ist die Steuerung eines Satelliten über verschiedenste Parameter wie z.B. die Region des Interesses (*Region-Of-Interest*), der Aufnahmezeitpunkt, die Bodenauflösung oder der Einfallswinkel mit Azimut und Höhe (Simonis & Echterhoff, 2011, S. 11).

Die standardisierte Schnittstelle des Services dient zum Verwalten eines *Asset Management Systems*, welches Informationsquellen identifiziert, benützt und managt.

2 Grundlagen

Assets bezeichnen verfügbare Mittel für das Sammeln von Informationen (z.B. Sensoren, Simulationen, Sensorplattformen, etc.).

Eine Client-Server-Interaktion basiert auf einer typischen Abfolge von Abfragen und Befehlen. Wie beim SOS (siehe Kapitel 2.6) identifiziert der Client den Funktionsumfang des SPS mit der *GetCapabilities* Anfrage und informiert sich über Sensordetails mit *DescribeSensor*. Die *DescribeTasking* Operation liefert dem Benutzer Informationen zu den Steuerparametern, die ihm zur Verfügung stehen und wie er diese einsetzen kann. Zusätzliche Operationen bieten Informationen zur Durchführbarkeit einer Anfrage (*GetFeasibility*) und zum Reservieren oder Aktualisieren eines Tasks (*Update/GetStatus/Cancel*). Weitere Methoden ermöglichen eine Abfrage des aktuellen Status oder das Abbrechen eines Tasks (Simonis & Echterhoff, 2011, S. 12 ff.).

Die oben beispielhaft erklärten Operationen, die ein Client an das SPS sendet, sind in Abbildung 14 dargestellt. Das *Web Notification Service* (WNS) dient zur asynchronen Übertragung von Benachrichtigungen oder Alarmen, die von komplexeren Service-schnittstellen erzeugt werden (siehe folgendes Kapitel 2.10).

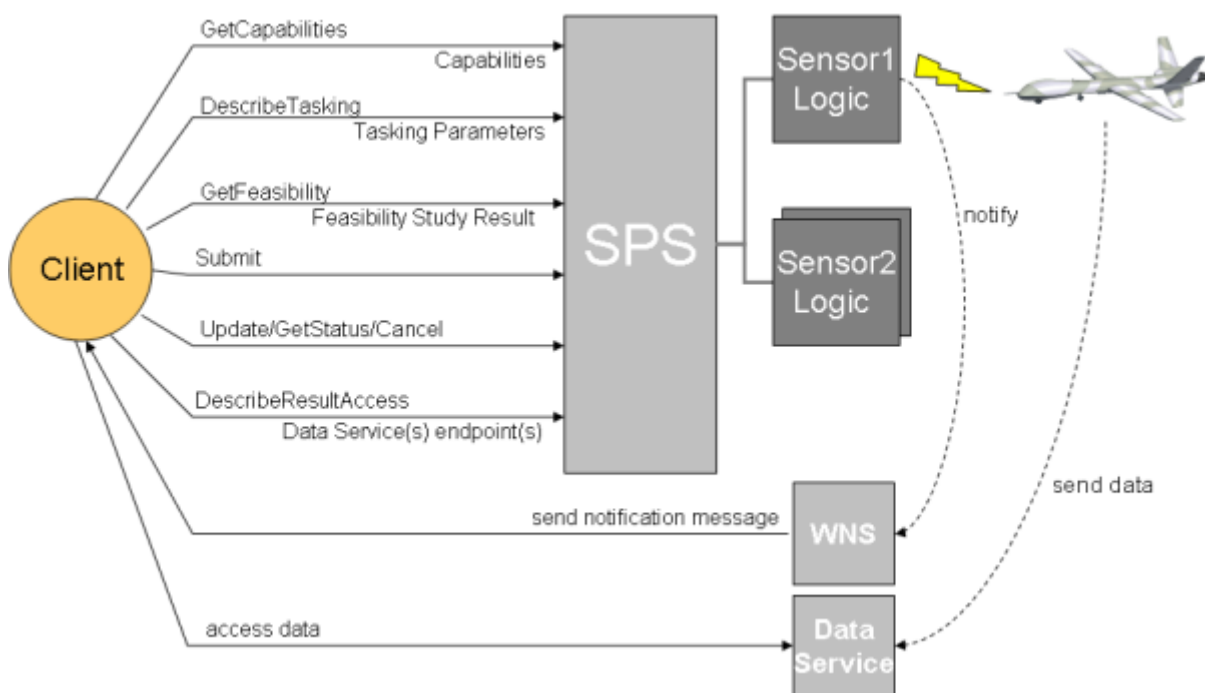


Abbildung 14: Exemplarischer Workflow zur Steuerung eines unbemannten Luftfahrzeugs (52North, 2013)

2.9 SENSOR ALERT SERVICE (SAS)

Das SAS ist noch nicht als OGC Standard freigegeben. Im OGC-Best-Practices-Dokument [OGC 06-028r3] wird eine webbasierte Schnittstelle zum Verteilen und Abonnieren von Alarmen beschrieben.

Ein SAS verteilt beim Eintreten bestimmter Messereignisse Benachrichtigungen an alle Clients, die für den jeweiligen Event registriert sind. Im Gegensatz zum pull-basierten SOS wird über das *Extensible Message Presence Protocol* (XMPP) ein push-basierter Datenübertragungsmechanismus definiert. Der Versand der Benachrichtigungen selbst erfolgt über das *Web Notification Service* (WNS) als E-Mail, HTTP, SMS, Telefon, Fax, usw. (Resch, 2010, S. 7).

Abbildung 15 zeigt den prinzipiellen Workflow eines SAS und die an den verschiedenen Stellen verwendeten Protokolle. Das Service besteht aus Operationen, die der Seite des Datenproduzenten (Sensor) und der Konsumentenseite (Client) zugeordnet sind. Die *Observations* der Sensoren werden mittels HTTP POST im SAS registriert, welches anschließend den Messaging Server zum Anlegen eines neuen *Multi User Chat* (MUC) beauftragt. Ist das MUC angelegt, kommuniziert der Sensor direkt über das XMPP mit dem MUC und ist in der Lage Daten zu veröffentlichen. Ein Client registriert sich für den Empfang von Benachrichtigungen mit der *Subscribe* Operation beim SAS. Die Registrierung hat einerseits HTTP- und andererseits XMPP-basiert zu erfolgen, wenn der Benutzer mittels XMPP benachrichtigt wird. Werden die Benachrichtigungen über das WNS verteilt, so endet die Registrierung nach der HTTP-basierten Anfrage (Simonis & Echterhoff, 2011, S. 6 ff.).

2 Grundlagen

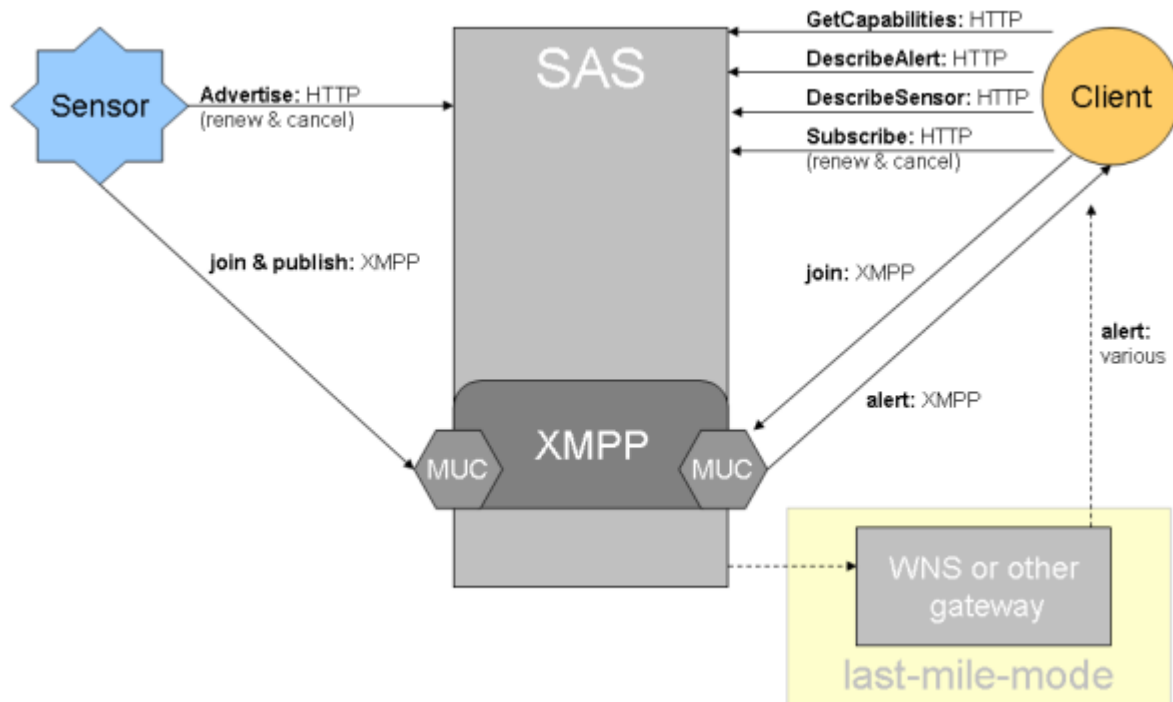


Abbildung 15: SAS Workflow (52North, 2013)

Ein Anwender kommuniziert mit dem Service über einen SAS Client und kann nach seiner Registrierung für den Empfang von Nachrichten, einen Standort, einen Sensor sowie ein Phänomen auswählen und Bedingungen definieren, für die er Benachrichtigungen erhalten möchte. Der Versand der Benachrichtigungen erfolgt über XMPP oder WNS.

2.10 WEB NOTIFICATION SERVICES (WNS)

So wie das SAS ist auch das *Web Notification Service* kein OGC Standard. Das OGC-Best-Practices-Dokument [OGC 06-095] beschreibt das Konzept des WNS-Modells, das einen asynchronen Datenaustausch mit anderen Services ermöglicht.

Das WNS transformiert die HTTP-basierten eingehenden Nachrichten in ein gewünschtes Protokoll (z.B. E-Mail, SMS, automatische Telefonanrufe) und leitet diese an die Benutzer weiter. So kann ein Client, der nicht ständig mit dem Internet verbunden ist, das WNS auch für eine telefonische Alarmierung verwenden. Dabei empfängt das WNS z.B. vom SAS seine generierten Alarme im HTTP-Format, transformiert das Protokoll und alarmiert die Teilnehmer. Prinzipiell wird zwischen zwei Arten der Benachrichtigung unterschieden: Im ersten Fall, der sogenannten *One-Way-Communication*, wird der Benutzer ohne Erwartung einer Antwort benachrichtigt, während die *Two-Way-Communication* eine asynchrone Ant-

2 Grundlagen

wort vom Client verlangt (z.B. das Quittieren eines Alarmes) (Simonis & Echterhoff, 2006, S. xi ff.).

Von den folgenden, in der WNS Schnittstelle (siehe Abbildung: 16) spezifizierten, acht Operationen sind vier zwingend umzusetzen (Simonis & Echterhoff, 2006, S. 6 ff.):

1. **GetCapabilities (zwingend)** - Die *GetCapabilities* Operation liefert Metadaten zum Service selbst und beschreibt den Funktionsumfang der spezifischen Serviceimplementierung.
2. **GetWSDL (optional)** – Die Antwort auf eine *GetWSDL* Operation ist ein *Web Services Description Language* (WSDL) Dokument welches die Schnittstelle des Services inklusive der umgesetzten Operationen der spezifischen Implementierung beschreibt.
3. **Register (zwingend)** – Diese Operation ermöglicht die Registrierung eines Clients.
4. **Unregister (zwingend)** – Mittels *Unregister* kann sich der Client vom Service abmelden.
5. **UpdateSingleUserRegistration (optional)** – *UpdateSingleUserRegistration* erlaubt einem Client eine Registrierung zu aktualisieren (z.B. Änderung der Email-Adresse oder der Telefonnummer).
6. **UpdateMultiUserRegistration (optional)** – *UpdateMultiUserRegistration* erlaubt einem Client eine Registrierung durch Hinzufügen oder Löschen von Gruppenmitgliedern zu aktualisieren.
7. **DoNotification (zwingend)** – Mit *DoNotification* werden Nachrichten zu registrierten Benutzern im definierten Protokoll gesendet.
8. **GetMessage (optional)** – *GetMessage* erlaubt das Abfragen von den in der Datenbank des WNS gespeicherten Nachrichten.

2 Grundlagen

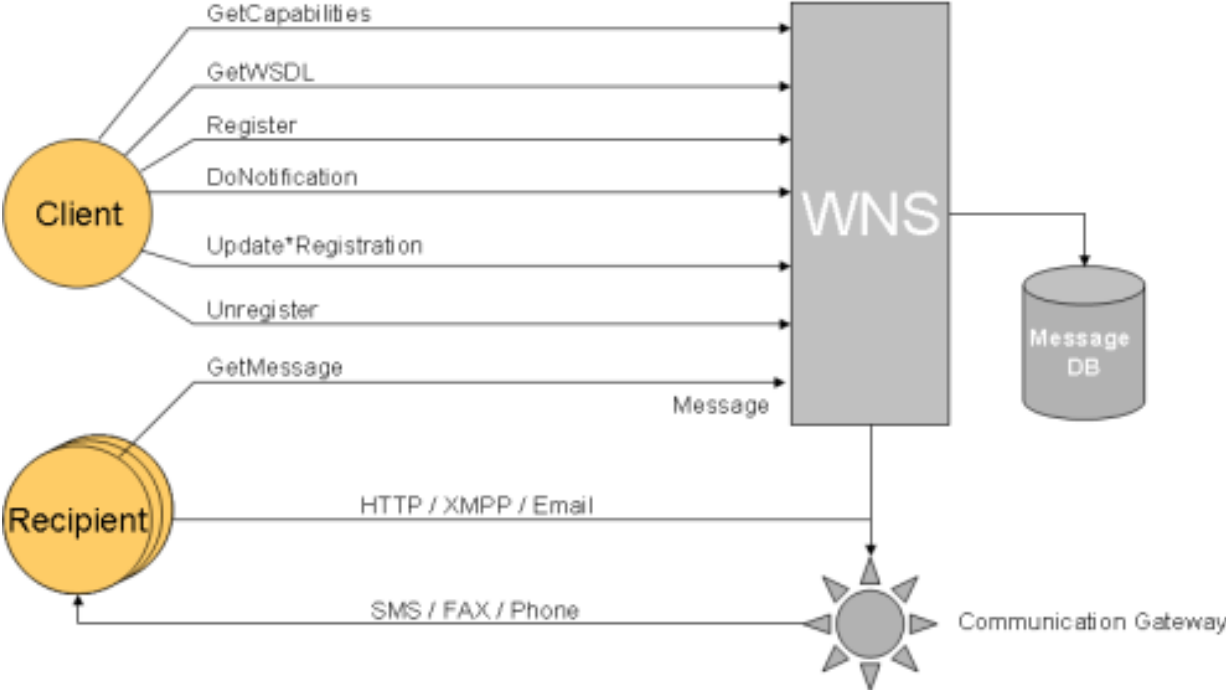


Abbildung: 16: WNS Workflow (52North, 2013)

3 GEO-SENSOR WEBS (STATE OF THE ART)

Dieses Kapitel beschreibt ausgewählte Forschungsprojekte zu Geo-Sensoren und Geo-Sensor Webs sowie deren Implementierung. Das Kapitel 3.1 beschäftigt sich mit Projekten der Forschungseinrichtung *Senseable City Lab*, dessen Ziel unter anderem eine Echtzeitanalyse von in Städten produzierten Daten ist. Der Datenzugriff auf die Sensoren und die Einbindung dieser in übergeordnete Systeme erfolgt hauptsächlich über OGC Standards. Die weiteren Kapitel beschreiben Forschungsprojekte und Implementierungen mit dem Schwerpunkt Interoperabilität durch verwendete Standards.

3.1 SENSEABLE CITY LAB

Der steigende Einsatz von Sensoren und Handheld-Geräten in den letzten Jahren ermöglicht neue Ansätze zur Analyse der vom Menschen erbauten Umgebung. *Senseable City Lab* bezeichnet eine Forschungseinrichtung am *Massachusetts Institute of Technology* (MIT), welches die in den Städten erzeugten Datenströme analysiert und auswertet. Smartphones, Internet, soziale Netzwerke, Smart Cards, Sensoren, etc. produzieren diese in den Städten konzentrierten Ströme von „Big Data“, die zunehmend in Echtzeit zu erfassen und zu analysieren sind. Das Ziel von *Senseable City Lab* ist die Untersuchung von Lebensgewohnheiten, Mobilitätsverhalten und Gesundheit der Bevölkerung. Diese Informationen dienen zur Erstellung von Konzepten um Lebensqualität, Energieeffizienz, Infrastruktur, etc. zu verbessern (ORF, 2013).

Ausgewählte Projekte des *Senseable City Labs* sind (Massachusetts Institute of Technology, 2013):

Road Frustration Index (2013) – Audi und *Senseable City Lab* bilden eine Arbeitsgruppe zur Entwicklung des *Road Frustration Index*, um emotionale Aspekte des Autofahrens zu untersuchen. Das Projekt kombiniert Echtzeit-Verkehrsdaten, Vorfälle im Straßenverkehr, Wetter und die Stimmung der Fahrer in 30 Ballungsräumen der USA, um das Ausmaß an straßeninduziertem Stress je Region annäherungsweise zu ermitteln.

3 Geo-Sensor Webs (State of the Art)

Local Warming (2013) – Das Projekt *Local Warming* durchleuchtet dynamisch gesteuertes lokales Heizen, um Energie zu sparen. Es soll nur dort geheizt werden, wo sich gerade Leute befinden. 20 Prozent des gesamten US Energieverbrauchs werden durch kommerzielle Gebäude verbraucht. Ein Prototyp des *Local Warmings* ist ein Infrarotenergiestrahler, welcher der Bewegung der Menschen folgt, wodurch ein gezieltes Wärmen der Menschen in einer kalten Umgebung ermöglicht wird.

Senseable Guggenheim (2012) – Im Workshop *Senseable Guggenheim* arbeitet das MIT zusammen mit dem *Austrian Institute of Technology* um zu untersuchen, wie und wann Passanten ihren Aufenthalt an öffentlichen Plätzen ändern. Der Workshop bot eine Einführung in die Theorie und Modelle zur Bewegung, Interaktion und Kommunikation von Menschen und wie diese Information für das Design öffentlicher Plätze genutzt werden kann. Die Analyse wurde mittels Image Processing, Umweltsensoren und der Analyse von sozialen Netzen durchgeführt. Abbildung 17 zeigt die durch eine auf einem Gebäude montierte Kamera erfassten Aktivitäten im Raum. Eine Bilderkennungssoftware detektierte die Bewegung und die Trajektorien der Körper.

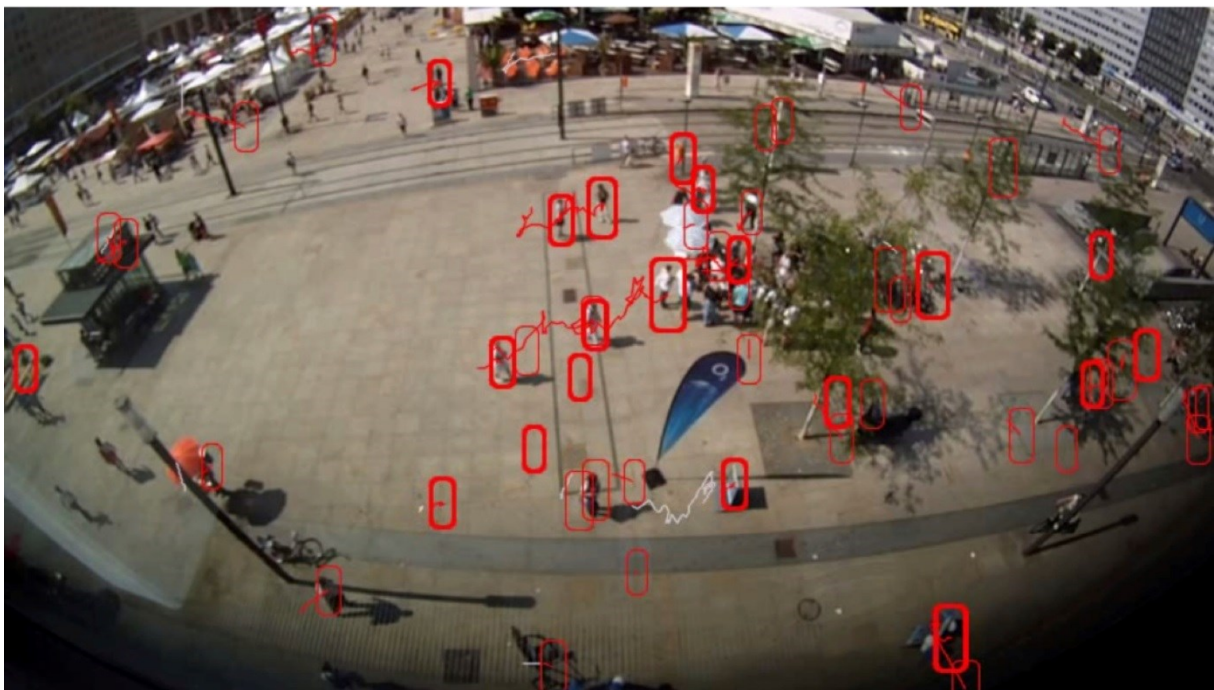


Abbildung 17: Erfassung der Aktivitäten im Raum mittels einer Kamera montiert auf einem Gebäude (Massachusetts Institute of Technology, o.J.)

3 Geo-Sensor Webs (State of the Art)

Health Infoscape (2011) – Das Projekt *Health Infoscape* dient zur Schaffung neuer Analysemöglichkeiten um räumliche Abhängigkeiten der menschlichen Gesundheit in den USA zu verstehen. Dabei wurden 7,2 Millionen anonymisierte elektronische Krankenakten statistisch untersucht. Eine erste Analyse ergab, dass Krankheiten keine isolierten Ereignisse sind und scheinbar nicht zusammenhängende Erkrankungen eng miteinander verbunden sind.

Trash Track (2009) – Trash Track bezeichnet ein Projekt zur Optimierung von Entsorgungsketten, indem Müll-Wege durch das Abfallsystem einer Stadt ermittelt werden. In Seattle dienten zur Ermittlung der aktuellen Positionen an 2.000 Abfallstücken (an wiederverwertbarem Müll, Sondermüll und Elektrogeräten) angebrachte elektronische Labels, die Informationen über das Mobilfunknetz an eine Datenzentrale sendeten. Einige Müllstücke durchquerten die Vereinigten Staaten und nicht alle Objekte wurden auf legalen Endpunkten entsorgt (Ratti & Townsend, 2012).

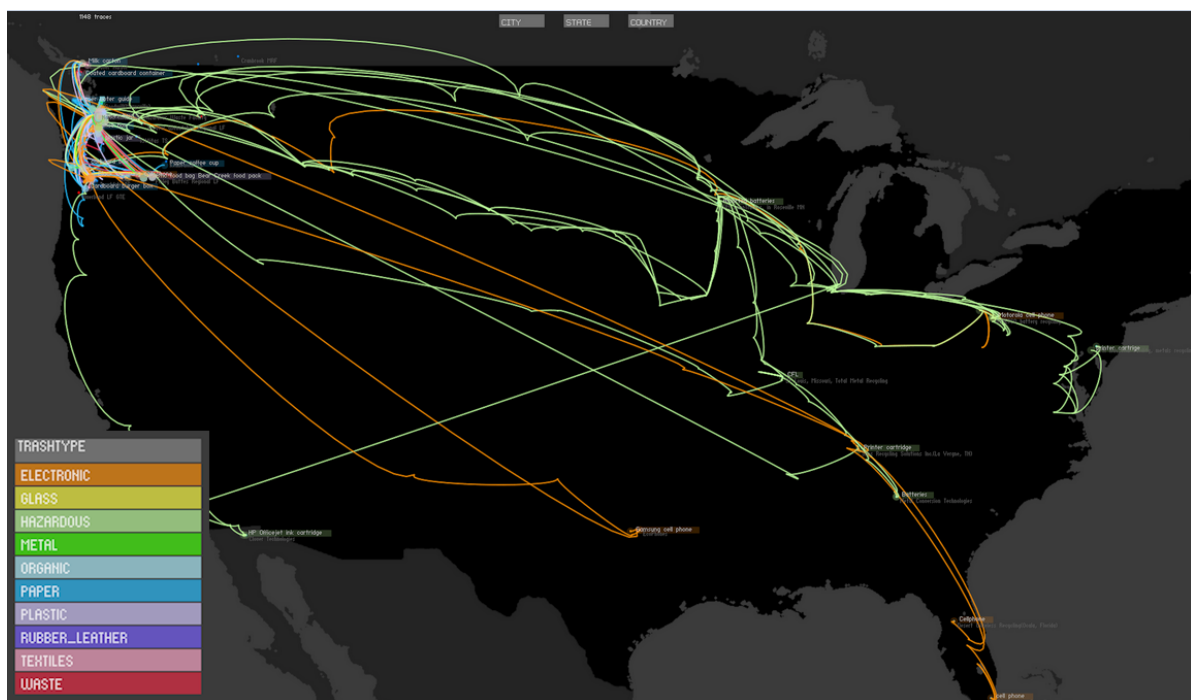


Abbildung 18: Transportwege des Abfalls ausgehend von Seattle (Massachusetts Institute of Technology, o.J.)

Der längste erfasste Weg war der einer Druckerpatrone, die 6.125 Kilometer bis zur Entsorgung zurücklegte. Eine Analyse der akquirierten Daten zeigt Möglichkeiten zur Verringerung der Transportwege und der damit einhergehenden Kohlendioxidemissionen.

3 Geo-Sensor Webs (State of the Art)

Seattle könnte außerdem mit den gewonnenen Informationen die Bürger zum verstärkten Recycling und zu einer sachgerechten Entsorgung des Mülls bewegen (Ratti & Townsend, 2012).

3.2 LIVE GEOGRAPHY

Live Geography bezeichnet einen Workflow, der Messungen von Echtzeitdaten in eine voll standardisierte Infrastruktur einbindet und diese mit anderen Systemen, wie einem *Complex Event Processing* (CEP), verknüpft. Das Framework wurde vom Forschungsprojekt *Real-time Geo-awareness* (RTGA) des Forschungsinstitutes iSPACE gemeinsam mit der Firma SYNERGIS, gefördert von der österreichischen Forschungsgesellschaft (FFG) entwickelt, um der Vision von Al Gore aus dem Jahr 1999, der „digitalen Haut der Erde“ (siehe Kapitel 1), einen Schritt näher zu kommen. Wie in Abbildung 19 dargestellt, besteht das *Live Geography* Framework aus einer Kombination verschiedener Hard- und Softwarekomponenten und verwendet zum Zweck einer Interoperabilität die offenen Standards des OGC (siehe Kapitel 2). Serverseitig sind SOS als *Pull Services* und SAS als *Push Services* implementiert. Aufwendiger ist die Umsetzung der Softwaremodule auf der Client Seite. Hier werden mit den durch standardisierte Services abgefragte Daten in den verschiedensten Geoinformationssystemen (GIS) Analysen und sowie die Visualisierung der Ergebnisse in Echtzeit durchgeführt (Mittelböck & Resch, 2009).

Die Infrastruktur kann in fünf miteinander interagierende Komponenten unterteilt werden (siehe Abbildung 19) (Resch & Mittelböck, 2010):

- Komponente 1 ist das Geo-Netzwerk bestehend aus verschiedensten Sensoren
- Komponente 2 umfasst die Kommunikation des Sensornetzwerkes mit einer Datenzentrale über verschiedene Technologien (GPRS, Standleitung, GSM, etc.)
- Komponente 3 fusioniert und harmonisiert durch die Nutzung von OGC Standards Sensoren aus den unterschiedlichen Instanzen
- Komponente 4 führt serverseitige, anwendungsspezifische Analysen durch
- Komponente 5 stellt die analysierten Daten in Abhängigkeit der Bedürfnisse des Endnutzers oder der Benutzergruppen dar

3 Geo-Sensor Webs (State of the Art)

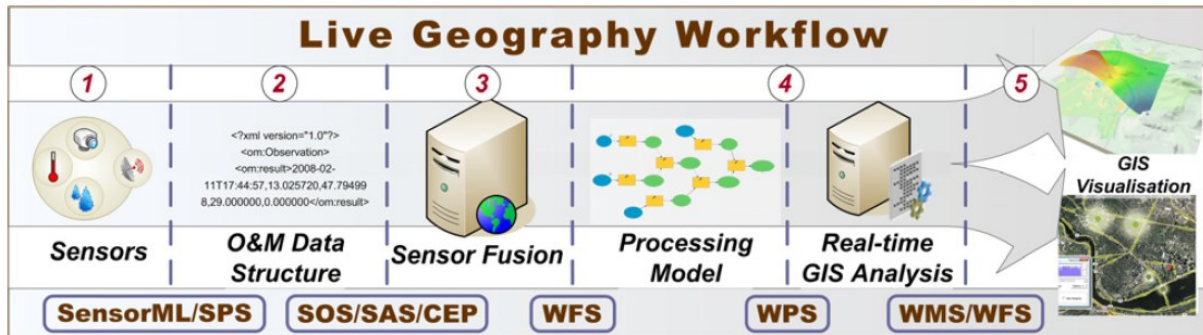


Abbildung 19: Grundlegende Architekturkomponenten und standardisierte Schnittstellen der Live Geography Infrastruktur (Resch & Mittelböck, 2010)

3.2.1 EMBEDDED SYSTEM DESIGN

Ein speziell entwickelter Software-Baukasten ermöglicht die stationäre Integration von Echtzeitmessungen, sowie die Einbindung mobiler GPS-gestützter Sensornetzwerke über die offenen Standards des SWE direkt von *Embedded Devices* in nahezu zeitliche geographische Analyseprozesse. Als Messinstrumente werden meteorologische Sensoren zur Messung von Lufttemperatur, Strahlungsparameter, relativer Luftfeuchtigkeit etc., sowie Biometrie-Sensoren, die unter anderem den Puls, die Sauerstoffsättigung, etc. messen und beliebige andere Sonden verwenden. Die miniaturisierten *Embedded Device* Computer sind im Gegensatz zu herkömmlichen PCs mit spezifischen Funktionalitäten wie etwa einem geringen Energieverbrauch, einem integrativen Design und einer Bereitstellung der Messdaten in Echtzeit ausgestattet. In Zusammenarbeit mit der Fachhochschule Salzburg entwickelte iSPACE ein Prototyp-Framework eines *Embedded Device* Computers, der Sensormessinformationen direkt am Gerät in einer *Embedded* Datenbank organisiert und über die standardisierten Schnittstellen des SOS und SAS zur Verfügung stellt. Das *Embedded Device* aus Abbildung 20 links hat ca. die Größe einer Kaugummischachtel. Die Sensoren können über verschiedenen Schnittstellen (GPIO, Seriell, USB, I2C und ZigBEE) an das Gerät angeschlossen werden (Mittelböck, et al., 2009).

Die *Observations* werden in einer SQLite-Datenbank gespeichert. *Sensor Daemons* konvertieren die gelesenen Sensordaten, bevor diese dem freien Webserver *Lighttpd* übergeben werden. Die geographischen Positionen, welche als Messparameter mitgeführt sind, werden über GPS/Galileo Empfänger bestimmt und mittels *Differential GPS* (DGPS) korrigiert. Die dazu nötigen Korrekturdaten werden über das in der Software eingebettete

3 Geo-Sensor Webs (State of the Art)

Networked Transport of RTCM via Internet Protocol (NTRIP) ermittelt (Resch & Mittelböck, 2010).

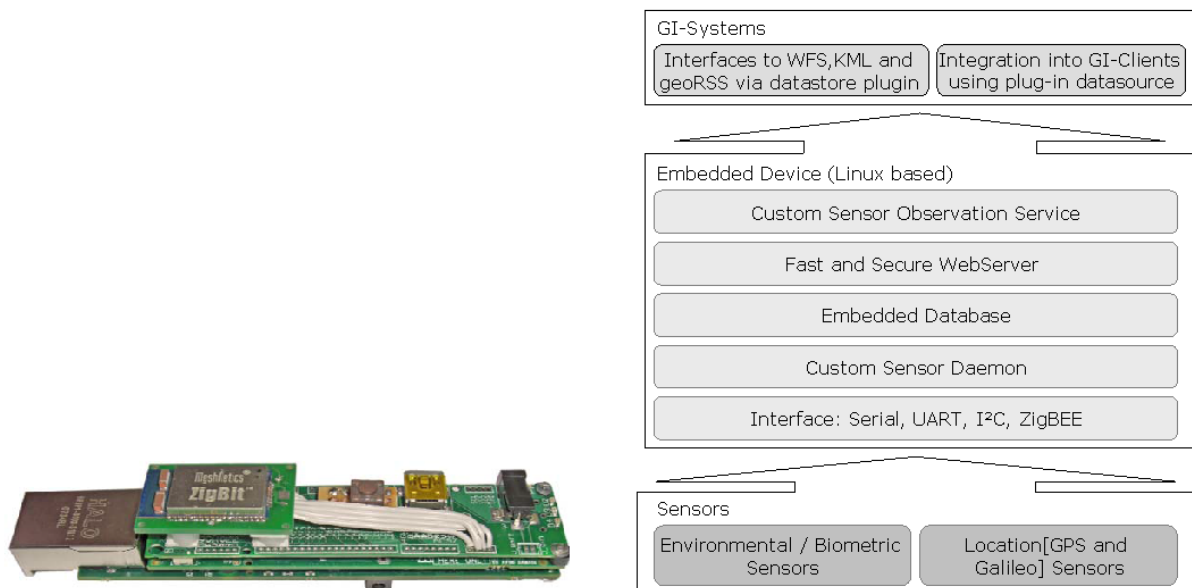


Abbildung 20: Embedded Device für standardisierte Messung und Übertragung von Umweltparametern links und Service Stack am Embedded Device rechts (Mittelböck, et al., 2009)

Zusätzlich ist ein in *Embedded C++* entwickeltes SAS zur Generierung von Alarmen implementiert, das Informationen über den MUC bereitstellt (siehe Kapitel 2.9). Der am Embedded Device installierte Open Source Server *Jabberd2* ermöglicht ein Streaming der SAS-Daten mittels XMPP-Protokoll in das Internet. Im SOS sind die drei zwingenden Operationen *DescribeSensor*, *GetCapabilities* und *GetObservation* implementiert (vgl. Kapitel 2.6) (Mittelböck & Resch, 2009).

3.2.2 IMPLEMENTIERUNGEN

Dieses Kapitel beschreibt den portablen Einsatz des *Live Geography* Ansatzes in fünf Anwendungsgebieten (Resch & Mittelböck, 2010).

Live Monitoring der Verschmutzung für die öffentliche Gesundheit – Das Ziel des Projekts *Common Scents* der Forschungseinrichtungen *Senseable City Lab*, *iSPACE*, des *Harvard University Sensor Network Lab*, sowie des *City of Cambridge's Public Health Department* und der *BBN Technologies* ist ein Monitoring der Luftverschmutzung in Echtzeit zum Zweck der öffentlichen Gesundheit. Das Netzwerk der *Common Cents* besteht dabei aus Sensoren zur

3 Geo-Sensor Webs (State of the Art)

Messung der CO₂-Konzentration und meteorologischer Parameter wie Windgeschwindigkeit, Lufttemperatur und relative Luftfeuchtigkeit. Das Ziel ist die Versorgung der Bürger mit aktuellen Daten, um kurzfristige Entscheidungen in naher Echtzeit zu treffen.

Die Implementierung, dargestellt in Abbildung 21, untersucht die zeitliche Korrelation zwischen Fluktuationen der erfassten Daten und der Verkehrsdichte, Wettersituation sowie anderen tageszeitlich abhängigen Bedingungen. Die Zeitreihe im linken unteren Teil der Abbildung 21 zeigt die Entwicklung der Sensorwerte, die in das O&M Format integriert worden sind. Wird die Zeitreihenanimation gestartet, ändert sich die Größe der grünen Symbole in Abhängigkeit des Messwertes.

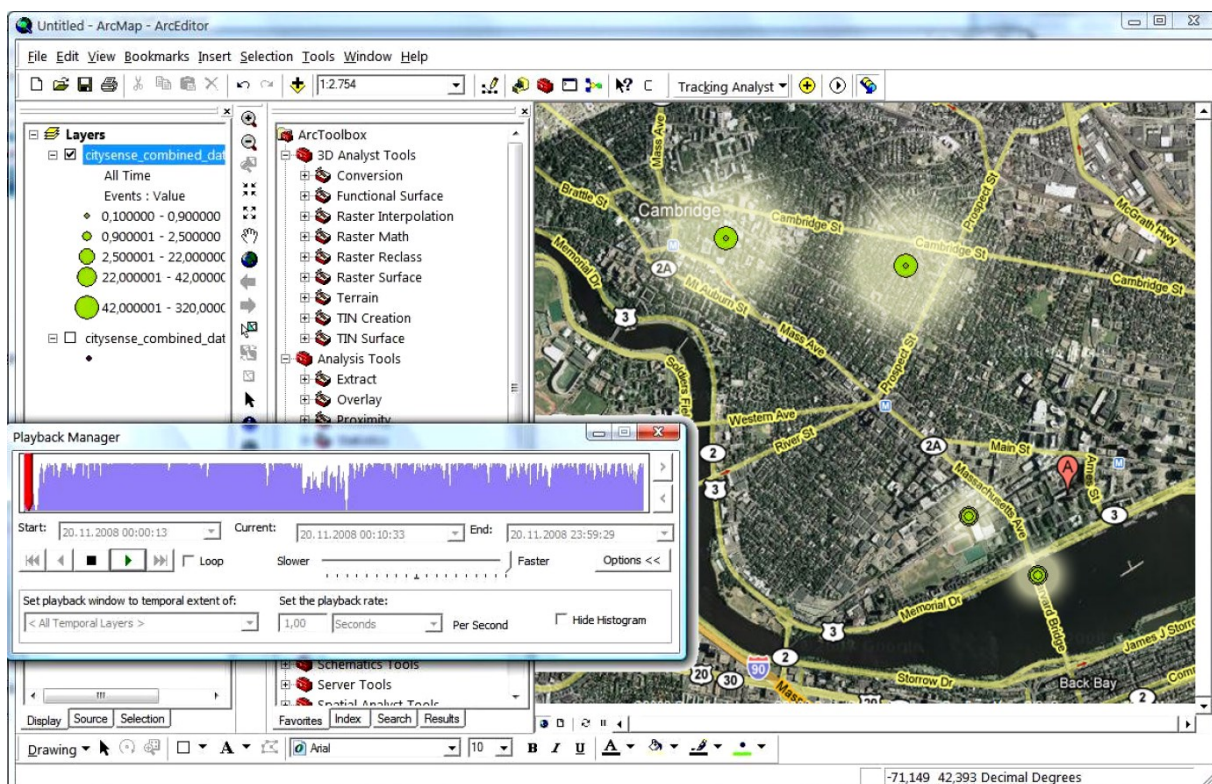


Abbildung 21: Darstellung der Verschmutzungsparameter als Zeitreihe mit ESRI ArcGIS (Resch & Mittelböck, 2010)

Detaillierte Erfassung von geringen Schwankungen der Lufttemperatur – Eine weitere Implementierung des *Live Geography* Ansatzes wurde im Zuge des *Real-time Geo-awareness* Projekts zur Beurteilung von Temperaturschwankungen durchgeführt. Ziel ist es, Hitzeinseln mit einer hohen räumlichen Auflösung in Echtzeit zu detektieren. Des Weiteren können Korrelationen der Temperatur mit anderen Umweltparametern wie Luftverschmutzung, Ozon oder verkehrsbedingten Emissionen analysiert werden. Eine Alarmierung über vor-

3 Geo-Sensor Webs (State of the Art)

definierten Events z.B. bei Überschreitung eines Grenzwertes, wird über das SAS gewährleistet. Ein CEP System kontrolliert Daten auf Ausreißer, führt raum-zeitliche Plausibilitätskontrollen durch und detektiert Events (Resch & Mittelböck, 2010).

Durch Transformation der Datenstruktur von SOS Antworten in die Ausgangsformate des *Web Feature Service (WFS)*, *Web Map Service (WMS)* und *Web Coverage Service (WCS)* mittels der Mapserver-Software *GeoServer* werden die SOS Antworten on-the-Fly in eine Geoinformations-Software eingebunden. Diese direkte Konvertierung ohne aufwendiges Zwischenspeichern in einer temporären Datenbank, ermöglicht eine einfache Integration der Sensordaten in GIS Applikationen und dadurch eine ubiquitäre Visualisierung und Datenanalyse. Abbildung 22 zeigt eine real-time *Inverse Distance Weighting (IDW)* Interpolation der gemessenen Lufttemperaturwerte, die aus den verschiedensten SOS Instanzen abgefragt werden. Die Analyse wurde im ESRI ArcGIS durchgeführt. Der Zugriff auf Echtzeitdaten erfolgt über die *Live Sensor Extension*, die eine direkte Integration der standardisierten Sensordaten in ArcGIS ermöglicht (Bernd, et al., 2009).

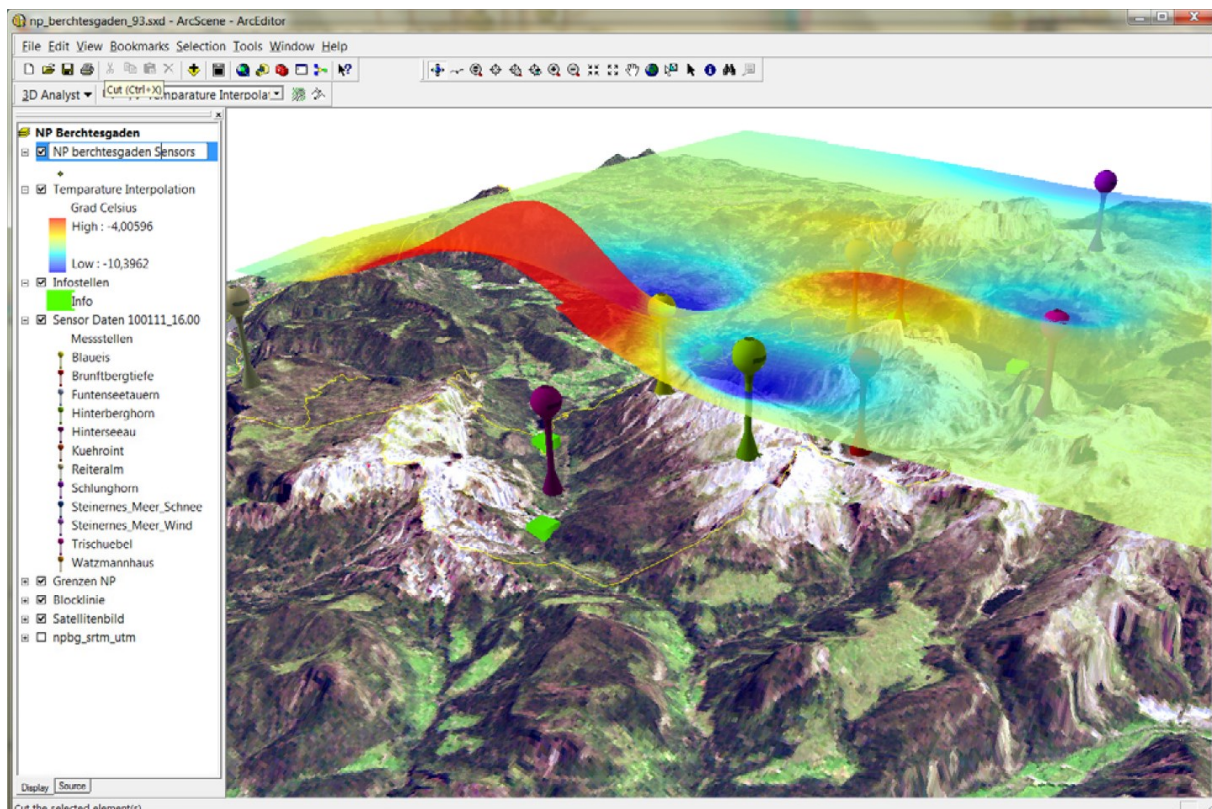


Abbildung 22: Echtzeit-Interpolation von Umgebungstemperaturwerten für das Monitoring von optimalen Umweltparametern der lokalen Fauna und Flora im Nationalpark Berchtesgaden (Resch & Mittelböck, 2010)

3 Geo-Sensor Webs (State of the Art)

Ubiquitäre Überwachung biometrischer Parameter – Das Projekt *geoHealth* wurde in Kooperation von iSPACE und der Fachhochschule Salzburg (*University of Applied Sciences*) ins Leben gerufen. Ein aus biometrischen Sensoren bestehendes System misst Puls und Sauerstoffsättigung zur Gesundheitsüberwachung einer Person. In Abbildung 23 ist das Web Interface der Applikation dargestellt. Der linke Konfigurationsbereich dient zur Auswahl des Messgerätes inklusive Messparameter, Abtastzeit und die Anzahl der maximalen Messpunkte, die in der Historie gespeichert werden. Der Zugriff und das Management der Daten basiert auf OGC SWE konformen Standards. In der Karte des rechten Bereiches werden die zuletzt gemessenen Positionen dargestellt. Das System ist nicht nur für eine Überwachung des Gesundheitszustandes von Patienten gedacht, sondern kann auch für andere Anwendungsbereiche wie beispielsweise das Tracking von Gegenständen, die Kontrolle der Waren einer Nahrungsmittelversorgungskette inklusive Messparameter zu deren Qualität, oder zur Verfolgung eines gestohlenen Autos verwendet werden.

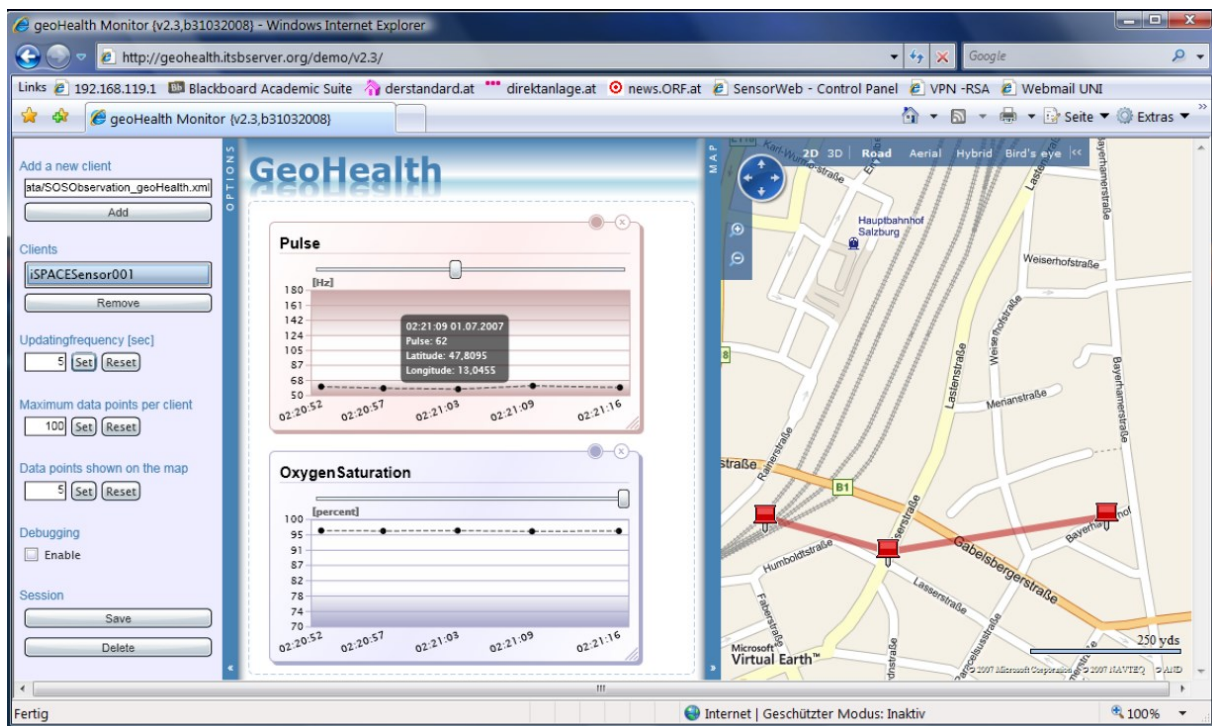


Abbildung 23: Illustration der *GeoHealth* Applikation: Historie biometrischer Parameter mit Darstellung der geographischer Lage implementiert in einer MS Silverlight Umgebung (Resch & Mittelböck, 2010)

Beurteilung der Luftqualität in Echtzeit – Das vom *Seventh Framework Programme* (FP7) der europäischen Kommission finanzierte Forschungsprojekt *GENeric European Sustainable Information Space for environment* (GENESIS) zielt einerseits auf den Einsatz einer

3 Geo-Sensor Webs (State of the Art)

standardisierten Infrastruktur zum Organisieren, Analysieren und Bereitstellen von Umweltinformationen und andererseits auf das Aufzeigen der Effizienz der Lösungen durch thematische Pilotprojekte in den verschiedenen Umweltbereichen ab. Der *Live Geography* Ansatz liefert durch seine standardisierten Mechanismen zum Messen und Bereitstellen der Daten (SOS), zur Datenfusionierung (GeoServer), zur Alarmierung (SAS und CEP) und durch die serverbasierten Datenanalysen (ARGIS Server-Applikationen) die Grundlage der Pilotprojekte. Das in Abbildung 24 dargestellte Webinterface zeigt flächenhafte Temperaturwerte im Rasterformat, die durch die geostatistische Interpolationsmethode *Kriging* erzeugt werden. Dabei werden dem Interpolationsmodell die gemessenen Punktdaten der Temperatur in Echtzeit übergeben. Ein spezieller Fokus des Projekts GENESIS ist eine Kopplung des SAS und CEP, sowie die Evaluierung des OGC *Sensor Event Service* (SES). Der Zugriff von GIS Softwareprodukten als Clients erfolgt über das standardisierte OGC *Web Processing Service* (WPS).

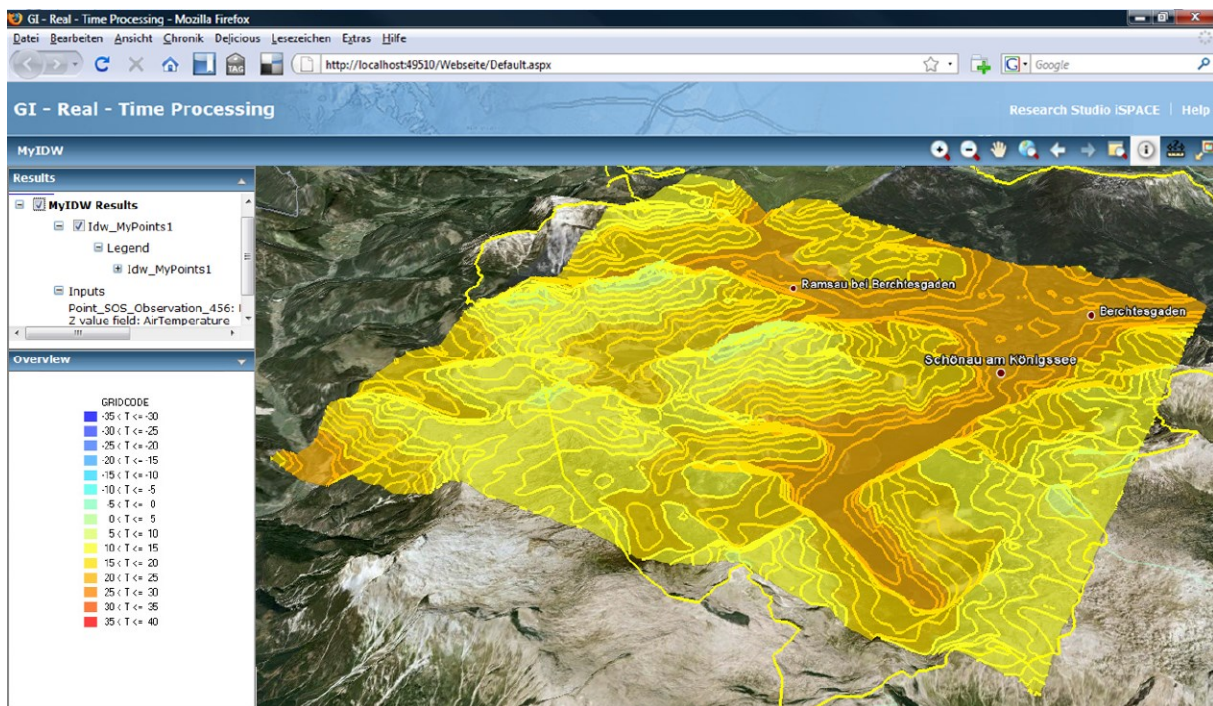


Abbildung 24: Webbasiertes Geo-Processing in Echtzeit, für eine detaillierte Einschätzung der urbanen Luftqualität (Resch & Mittelböck, 2010)

Echtzeit Decision Support für Strahlensicherheit – Im Rahmen des internationalen EU *Sixth Framework Programme* (FP6) Forschungsprojektes *G2real* wurde während der Strahlenspürübung *Shining Garden* in der Marktgemeinde Seibersdorf (Bezirk Baden) der *Live*

3 Geo-Sensor Webs (State of the Art)

Geography Ansatz angewendet und validiert. Testszenario war ein Satellitenabsturz, dessen verstrahlte Wrackteile mit einer Boden-Suchmannschaft möglichst effektiv aufzuspüren waren. Ziel des Feldtests war das Aufzeigen einer gemeinsamen Nutzung von bestehenden proprietären Softwaresystemen und Hardware-Entwicklungen in einem gemeinsamen ICT (*Information and Communications Technologies*) Framework. Im Mittelpunkt stand, wie in den zuvor beschriebenen Projekten, die Integration von Geodaten unterschiedlicher Instanzen mittels Standards der OGC SWE Initiative. Die Messwerte von Strahlenmesssonden sowie die DGPS (über NTRIP) und GALILEO ermittelten Positionsdaten wurden in Echtzeit akquiriert (Mittelböck, et al., 2011, S. 752 ff.).

Die Punkte in Abbildung 25 zeigen den zurückgelegten Weg des Sicherheitsexperten, der mit dem Strahlungsmessgerät die Strahlung in μ -Sievert und die Position mittel GPS/GALILEO Empfänger mit einer Abtastrate von einer Sekunde akquiriert hat. Die räumliche Interpolation wurde in Echtzeit unter Verwendung des *Inverse Distance Weighting* (IDW) Algorithmus durchgeführt. Die roten Bereiche stellen hohe Strahlungswerte dar.

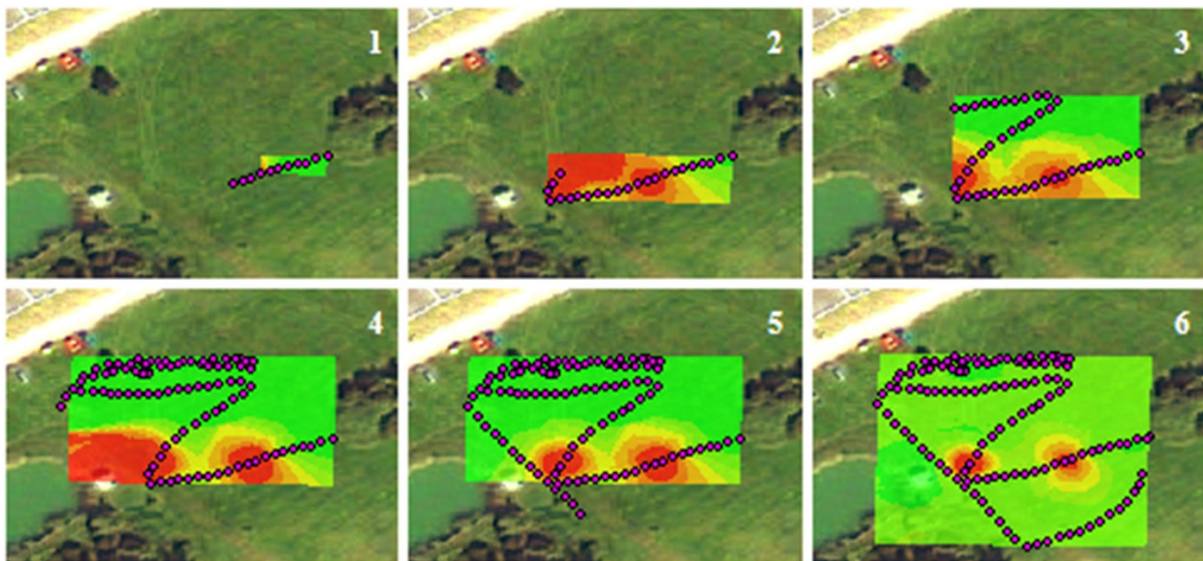


Abbildung 25: Interpolation der Strahlungspunktdaten in Echtzeit (Resch & Mittelböck, 2010)

3.3 SENSOR WEB IM WUPPERVERBAND

Die Wupper ist ein Fluss in Nordrhein-Westfalen (Deutschland) mit 113 Kilometer Länge und einem Einzugsgebiet von 813 Quadratkilometern, der in den Rhein mündet. Der 1930 gegründete Wupperverband umfasst eine Vielzahl an heterogenen Sensoren und Fachsystemen, die über die SWE Initiative in das webbasierte geographische Informationssystem *FlussGebietsGeoinformationssystem* (FluGGS) integriert und von standardisierten Services gesteuert werden. In den 22 Städten und Gemeinden des Einzugsgebietes werden 11 Klärwerke, zwei Trinkwasser-Talsperren, 10 Brauchwasser-Talsperren, 51 Pegelmessungen und 40 meteorologische Messstationen betrieben (Wupperverband, 2013).

Die OGC Sensor Web Komponenten sowie Prototypen wurden gemeinsam mit dem niederländischen Wasserverband *Aquo* im Rahmen des *Cross-Border-GDI* Forschungsprojektes *Water Catastrophe Management* (WCM) der EU Interreg-Initiative getestet. Vorhandene Geodateninfrastruktur (GDI) Ansätze werden für das Zusammenführen relevanter Datenbestände der Wasserwirtschaft genutzt, um eine grenzübergreifende GDI zu gestalten (X-border-GDI, 2010).

Abbildung 26 illustriert die Verwendung der OGC Services der Sensor Web Infrastruktur des Wupperverbands, die folgendermaßen eingesetzt werden (Wupperverband, 2013):

SOS – Die Bereitstellung von Sensorinformationen sowie Messreihen unterschiedlicher Institutionen erfolgt über das SOS. Ein browserbasierter SOS Client visualisiert die *Observations* als Diagramm oder stellt diese in tabellarischer Form dar.

SPS – Zur Überwachung kritischer Bereiche technischer Anlagen sowie für Plausibilitätsanalysen, der durch die Sensorik gemessenen Füllstände, Abschlags- und Durchflussmengen, sind Kameras installiert, die über ein SPS gesteuert werden können. Durch den Einsatz der SPS-Spezifikation können Kameras unterschiedlicher Hersteller mit nur einem Client angesprochen werden.

SES – Das *Sensor Event Service* (SES) dient zur Definition von Regeln, die von den Nutzern abonniert werden können. Ein Anwender kann sich beim SES für den automatischen Erhalt von Nachrichten registrieren. Er definiert für welchen Standort, für welchen Sensor und

3 Geo-Sensor Webs (State of the Art)

welches Phänomen er unter welchen Bedingungen (z.B. Überschreiten eines Grenzwertes) diese Nachrichten zu Events erhalten möchte.

WNS – Im SES ausgelöste Benachrichtigungen werden über das WNS versendet. Derzeit sind die Protokolle E-Mail und SMS realisiert.

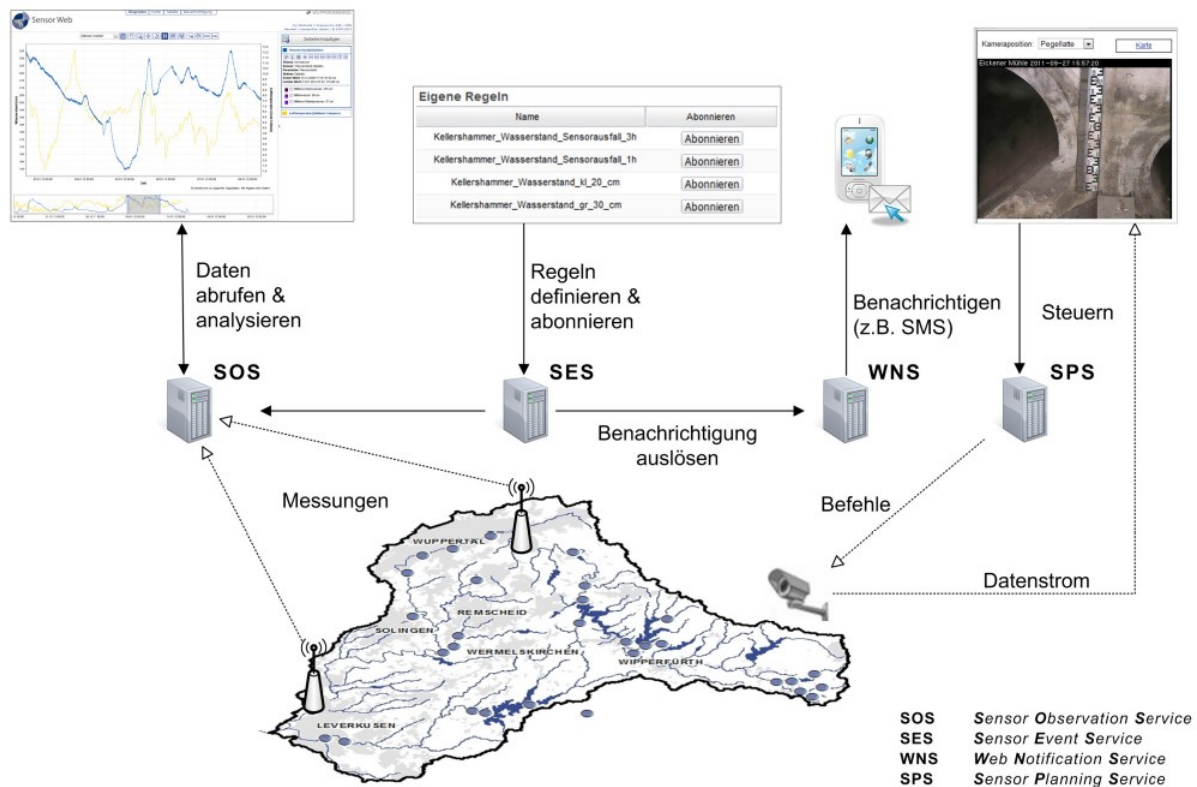


Abbildung 26: Die Sensor Web Infrastruktur des Wupperverbands (Wupperverband, 2013)

3.4 ERHÖHTE INTEROPERABILITÄT DURCH SENSOR-WEB-PROFILE

Aufgrund des steigenden Bedarfs an Interoperabilität für Sensor-Webs und der damit einhergehenden Notwendigkeit von Standards finden sich immer mehr Implementierungen wieder, welche die Standards der SWE-Initiative nutzen. Da die OGC Spezifikation generisch und nicht auf eine spezifische Fachdomäne zugeschnitten ist, können Unterschiede zwischen SWE-Implementierungen verschiedener Anbieter und Fachgebieten entstehen, wodurch eine beliebige Koppelung mit anderen Umsetzungen nicht immer möglich ist. Durch Verwendung von SWE-Datenmodellen und Profilen, wie beispielsweise die *WaterML 2.0* oder ein Hydrologieprofil für den SOS 2.0 Standard können Interoperabilitätsprobleme vermieden werden (Jirka, et al., 2013).

3 Geo-Sensor Webs (State of the Art)

3.4.1 GEOWOW

Das vom FP7 der europäischen Kommission teilfinanzierte Forschungsprojekt *GEOSS Interoperability for Weather Ocean and Water* (GEOWOW) zielt auf eine Vereinfachung der Bereitstellung und des Austausches von raumbezogenen Daten aus verschiedenen Fachgebieten über das *Global Earth Observation System of Systems* (GEOSS) ab. Eine Zusammenarbeit mit in den Bereichen Wetter, Ozeane und Wasser tätigen Organisationen und Unternehmen in sogenannten *Social Benefit Areas* (SBAs) dient einem benutzerorientierten Ansatz während der Systementwicklung (ESA, 2012).

GEOSS selbst verlinkt weltweit bestehende und geplante Beobachtungssysteme und unterstützt bei bestehenden Lücken die Entwicklung solcher Systeme. Durch die Verwendung technischer Standards sollen Daten verschiedenster Instrumente zu kohärenten Datensätzen kombiniert werden. Daten, Bilder und analytische Softwarepakete zu weltweit betreffenden Themen können über das Geoportal gesucht werden. Zugegriffen wird dabei auf bestehende Datenbanken und Portale (GEO, 2013).

3.4.1.1 SOCIAL BENEFIT AREA WATER

Die GEOWOW Arbeitsgruppe zum Austausch hydrologischer Messdaten (*SBA water*) besteht einerseits aus Organisationen, die im Bereich der Softwareentwicklung (52°North, Kisters) tätig sind und andererseits aus Anwendungspartnern (Universität Bonn, Bundesanstalt für Gewässerkunde, Global Runoff Data Center). Im Mittelpunkt der Arbeit stand die speziell auf hydrologische Messdaten zugeschnittene *WaterML 2.0* Spezifikation des OGC, welches ein Profil des O&M Standards darstellt. Aufgrund der generisch gehaltenen SOS Schnittstelle und der damit einhergehenden Anzahl an Freiheitsgraden bei einer Implementierung entwickelte die Arbeitsgruppe ein Hydrologieprofil für den SOS 2.0 Standard. Um die Interoperabilität zu erhöhen, wurden die Freiheitsgrade durch eine Anpassung des SOS 2.0 Standards an die *WaterML 2.0* Spezifikation reduziert. Der modifizierte Standard enthält folgende, teilweise modifizierte und ergänzte Operationen (vgl. Kapitel 2.6) (Jirka, et al., 2013):

1. **GetCapabilities (zwingend)** – Die verpflichtende Operation *GetCapabilities* ermöglicht den Aufruf von Metadaten und bleibt gegenüber der ursprünglichen SOS Definition im Hydrologieprofil unverändert.

3 Geo-Sensor Webs (State of the Art)

2. **DescribeSensor (zwingend)** – Der Abruf von Metadaten über Sensoren und Messprozessen über *DescribeSensor* ist auch verpflichtend und nicht modifiziert.
3. **GetObservations (zwingend)** – Der Abruf von Messdaten anhand räumlicher, zeitlicher und thematischer Kriterien mittels *GetObservations* ist von der Arbeitsgruppe im Hydrologieprofil so eingeschränkt, dass nur Dokumente in WaterML 2.0 Formaten ausgegeben werden.
4. **GetFeatureOfInterest (zwingend im Hydrologieprofil)** – *GetFeatureOfInterest* ist eine optionale Operation der SOS Implementierung, muss jedoch im Hydrologieprofil zwingend umgesetzt werden. Die Operation ermöglicht eine einfachere Suche nach Messpunkten eines SOS in Abhängigkeit von Geometrien, auf welche sich die bereitgestellten Messdaten beziehen.
5. **GetDataAvailability (zwingend, noch keine Standardoperation)** – Die *GetDataAvailability* Operation liefert Informationen in welcher Kombination aus Zeit, Geometrien, Sensoren und Parametern eine SOS Instanz Daten anbietet. Die Operation wird im Zuge des Standardisierungsprozess des OGC getestet und angepasst und ist noch nicht im Profil aufgenommen.

Zusätzlich zu den im Hydrologieprofil eingeschränkten Operationen sind die Bedeutungen der Begriffe an den Bereich der Hydrologie angepasst. Die Geometrie eines *Feature-Of-Interest* kann nur punktförmig sein und ist im *WaterML 2.0* Standard als *MonitoringPoint* definiert. Der Begriff *Procedure* gilt im SOS 2.0 Hydrologieprofil nur für Sensortypen, Zeitreihentypen und Algorithmen (vgl. Kapitel 2.3). Weitere Einschränkungen beziehen sich beispielsweise auf die Kommunikationsparameter zu den verschiedenen SOS Instanzen, wie die Limitierung der Größe von SOS-Antworten oder das Abrufen der neuesten Werte einer Zeitreihe. Eine Implementierung des Profils durch die Entwicklung unterschiedlicher Softwarekomponenten verschiedener Anbieter soll die Interoperabilität der Komponenten zeigen (Jirka, et al., 2013, S. 266-268).

Die Projektpartner 52°North und Kisters haben folgende client- und serverseitige Technologien entwickelt und diese gemeinsam getestet. Die Bundesanstalt für Gewässerkunde stellt für die Implementierung einen Auszug aus den Daten des *Global Runoff Data Centers* (GRDC) zur Verfügung (Jirka, et al., 2013, S. 268-270):

3 Geo-Sensor Webs (State of the Art)

52°North Implementierung – In der bestehenden Implementierung des SOS der Firma 52°North wurde einerseits die Ausgabe des WaterML 2.0 Formats und andererseits die Unterstützung des SOS 2.0 Hydrologieprofils adaptiert. Des Weiteren ist der bestehende Sensor Web Client um Funktionen zur Nutzung spezifischer Eigenschaften des SOS 2.0 Hydrologieprofils erweitert. Z.B. können Metadaten zu verfügbaren Zeitreihen von einer SOS-Instanz abgefragt werden. In Abbildung 27 sind als SOS Instanz die hydrologischen Daten des GRDC gewählt. Wird ein *Feature-Of-Interest* (blauen Kreise mit Punkt in der Mitte) gewählt, so können zu diesem die gemessenen Eigenschaften der Phänomene als Zeitreihe dargestellt werden.

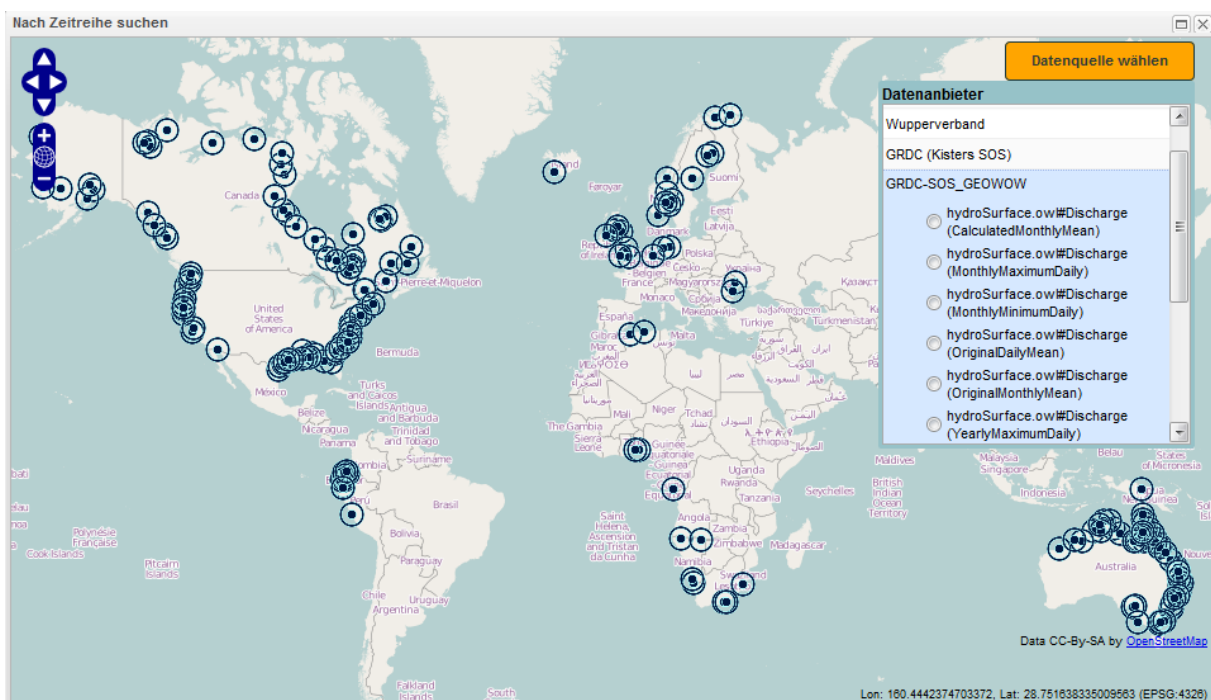


Abbildung 27: Menü des 52°North Sensor Web Clients zur Auswahl anzuzeigender Zeitreihen (aktuell ausgewählt sind die hydrologischen Daten des GRDC-SOS) (52°North Open Source Client, 2013)

Kister Implementierung – Serverseitig nutzt Kisters das kommerziell verfügbare Servicemodul *KISTERS Web Interoperability Solution* (KiWIS) um hydrologische Daten durch Nutzung offener Standards über das Web auszutauschen. In Kombination mit der Datenbanklösung WISKI zu Bereitstellung hydrologischer Daten kann KiWIS Messdaten in Echtzeit über das Web bereitstellen. Im Zuge des GEOWOW wurde das Modul um die Unterstützung des WaterML 2.0 in Verbindung mit dem SOS 2.0 Hydrologiemodul erweitert. Realisiert sind wieder die für das Profil zwingend umzusetzenden Operationen *GetCapabilities*,

3 Geo-Sensor Webs (State of the Art)

DescribeSensor, *GetObservation* und *GetFeatureOfInterest* sowie die sich noch im Standardisierungsprozess befindende Operation *GetDataAvailability*.

3.5 INTEGRATED OCEAN OBSERVING SYSTEM (IOOS)

Im Rahmen des *Integrated Ocean Observing System* (IOOS) Programms der *National Ocean and Atmospheric Administration* (NOAA) der Vereinigten Staaten von Amerika wurde das *Data Integration Framework* (DIF) mit einer Laufzeit von drei Jahren konzipiert. Ziel des Projekts war die Nutzung gemeinsamer Daten und Informationen über Ozeane für vier entscheidungsunterstützende Modelle der NOAA zur Verbesserung der Genauigkeit und Effizienz von Modellberechnungen, Einschätzungen sowie Vorhersagen. Im DIF wurden die sieben Kernvariablen Meerestemperatur, Wasserstand, Ozeanfarbe, Ströme, Salzgehalt, Wind und Wellen definiert, die dann in einfacher Weise einem Modell als Eingangsparameter übergeben werden können. Als drei Hauptdatenanbieter für das Framework wurden das *National Data Buoy Center* (NDBC), *Center for Operational Oceanographic Products and Services* (CO-OPS) und die *CoastWatch* gewählt. Die Anbieter bieten gemeinsam eine signifikante Menge an Kernvariablen. Das *Open-source Project for a Network Data Access Protocol* (OPeNDAP) Service liefert die vom *CoastWatch* aus Satellitendaten ermittelte Ozeanfarbe. Die Abfrage der Messdaten der restlichen Kernvariablen erfolgt über von den Datenanbietern verwaltete SOS Instanzen (IOOS, 2010, S. 5-6).

Die *Observations* der SOS Instanzen dienen Modelle folgender Bereiche als Eingangsparameter (IOOS, 2010, S. 10):

1. **Harmful Algal Bloom Forecast System (HAB-FS)** – Basierend auf Satellitenbilddaten, Feldbeobachtungen, Modellen, Berichten aus dem Gesundheitswesen und Daten des NDBC liefert das HAB-FS Informationen zur Lage, zum Ausmaß, der Bewegung, sowie Entwicklung von schädlichen Algenblüten im Golf von Mexiko.
2. **Coastal Inundation (CI)** – Das CI Modell dient zur Unterstützung von Prognosen bezüglich der durch eine Sturmflut angerichteten Schäden. Es inkludiert Echtzeitdaten zu den Wasserständen und Winddaten vom NDBC und CO-OPS. Das Tool selbst wird als *SLOSH Display Program* (SDP) bezeichnet. Durch das DIF soll die

3 Geo-Sensor Webs (State of the Art)

Prognosefähigkeit mittels SDP von Hurrikans verursachte Überschwemmungen in Küstenbereichen verbessert werden.

3. **Hurricane Intensity (HI)** – In einer Studie wird bewertet, wie und ob die Berücksichtigung von Temperaturprofilen, ermittelt aus Satellitenaltimetriebeobachtungen, den Fehler der Vorhersage der Intensität eines Hurrikans des *Hurricane Weather Research And Forecast System* (HWRF) reduziert. Das HWRF wird dabei mit dem *Hybrid Coordinate Ocean Model* (HYCOM) gekoppelt. Der Vergleich der synthetischen Temperaturprofile mit In-Situ-Beobachtungen ergab einen geringeren Unterschied als 1°C.
4. **Integrated Ecosystem Assessments (IEA)** – Ziel des Projekts ist die Verbesserung des ERDDAP (*Environmental Research Division's Data Access Program*) Tools, welches Daten verschiedener DIF Standards in andere gewünschte Formate eines DIF transformiert. ERDDAP sammelt und übersetzt Daten für einen einheitlichen Zugriff.

Um Interoperabilität zwischen den einzelnen Modellen zu gewährleisten und einen einfachen Datenaustausch zwischen den Datenanbietern und entscheidungsunterstützenden Tools zu ermöglichen, hat das IOOS DIF standardisierte Webservices implementiert. In Tabelle 5 sind die verwendeten Webservices und deren unterstützte Datentypen sowie Rückgabeformate dargestellt. In-Situ Messungen werden über das SOS bereitgestellt, welches die Ergebnisse im O&M und die Metadaten zu Sensoren im SensorML Format bereitstellt. Das WFS oder DAP gibt Modellergebnisse, Level 3 Satellitenbilddaten und Oberflächenströme in Form von multidimensionalen Daten mit ihrer vollen Semantik für weitere Analysen zurück. Das WMS stellt die referenzierten Bilddaten für Kartendarstellungen bereit.

Tabelle 5: Webservices und Datenkodierung des IOOS Data Integration Frameworks (DIF) (IOOS, 2010, S. 42)

Datentyp	Webservice	Kodierung
Diskrete Features (z.B. In-Situ Punkt- oder Profildaten, Zeitreihen, Trajektorien)	Sensor Observation Service (SOS)	XML - Geography Markup Language (GML) basiert auf Observations & Measurements (O&M) für Daten; Sensor Model Language (SensorML) für Metadaten
Coverages (z.B. gerasterte Modellergebnisse, Level 3 Satellitendaten, HFR Oberflächenströme)	Data Access Protocol (DAP) oder Web Coverage Service (WCS)	Binär – Netzwerkdatenformate mit Konventionen für Klima & Vorhersagen (CF/NetCDF)
Georeferenzierte Bilddaten	Web Map Service (WMS)	Gebräuchliche Bildformate (PNG, TIFF, GIF, JPEG)

3 Geo-Sensor Webs (State of the Art)

Alle Standards, ausgenommen vom DAP, wurden vom OGC entwickelt. Das DAP wird vor allem für die Abfrage von komplexeren gerasterten Datensätzen mittels HTTP verwendet. Häufig verwendete Datensätze sind das NetCDF oder HDF, unterstützt wird jedoch jedes Format, inklusive vom Benutzer definierte Formate.

Daten des NDBC als Beispiel - Abbildung 28 zeigt die Stationen des NDBC dargestellt über Google Maps, dessen Messwerte über das SOS bereitgestellt und abgefragt werden. Die gelben Symbole kennzeichnen Stationen mit aktuellen Messwerten, für orange dargestellte Stationen sind historische Daten in einer Datenbank gespeichert und die roten Stationen haben zumindest in den letzten acht Stunden keine Daten akquiriert.

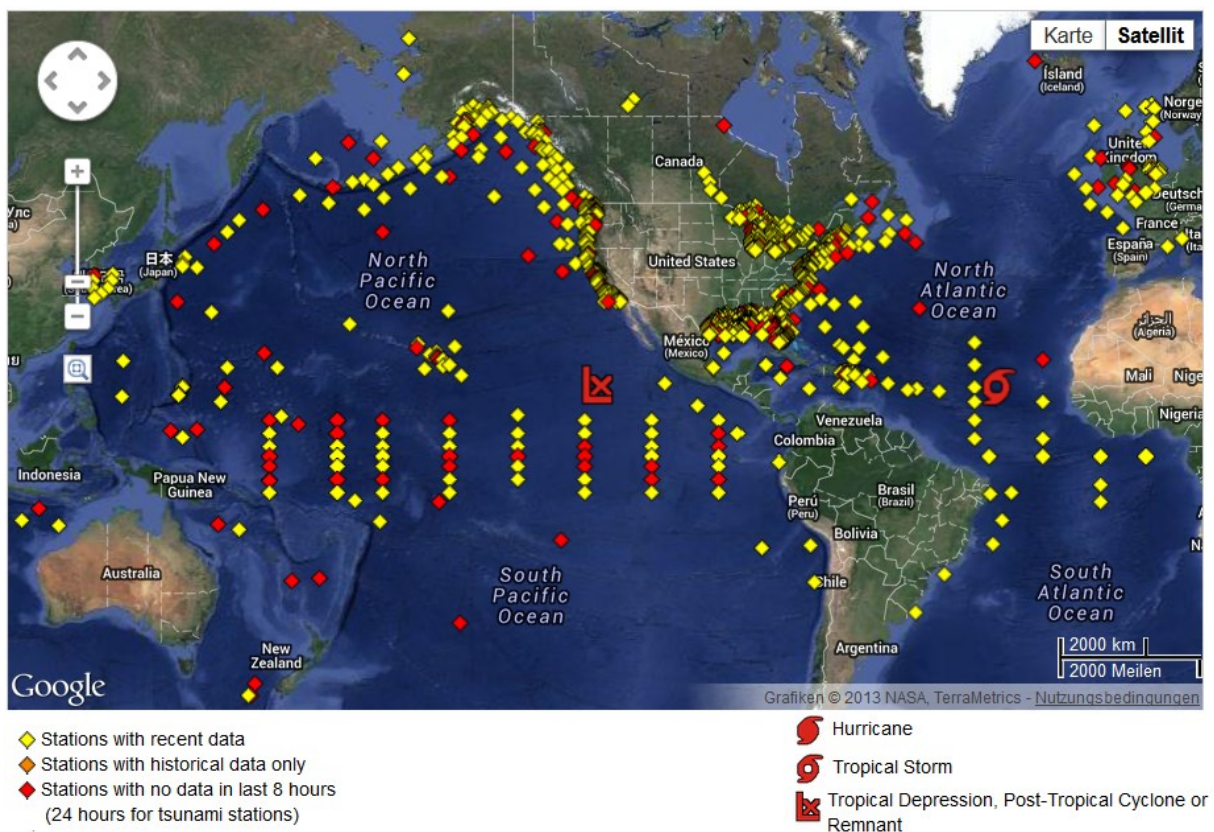


Abbildung 28: Observations des National Data Buoy Center (NDBC) dargestellt über Google Maps (NOAA, 2013)

Abbildung 29 links zeigt die 48 Stunden Vorhersage von Zugbahn, maximaler Windgeschwindigkeit, Windböen sowie Größe einer tropischen Depression des pazifischen Ozeans. Für den 26. Juli 2013 wird eine maximale Windgeschwindigkeit von 45 Knoten und Böen bis zu 55 Knoten vorhergesagt. Der rechte Teil des Bildes zeigt einen Hurrikan des

3 Geo-Sensor Webs (State of the Art)

Atlantiks, für den am 26. Juli 2013 Windgeschwindigkeiten bis zu 50 Knoten und Böen bis zu 60 Knoten prognostiziert werden.

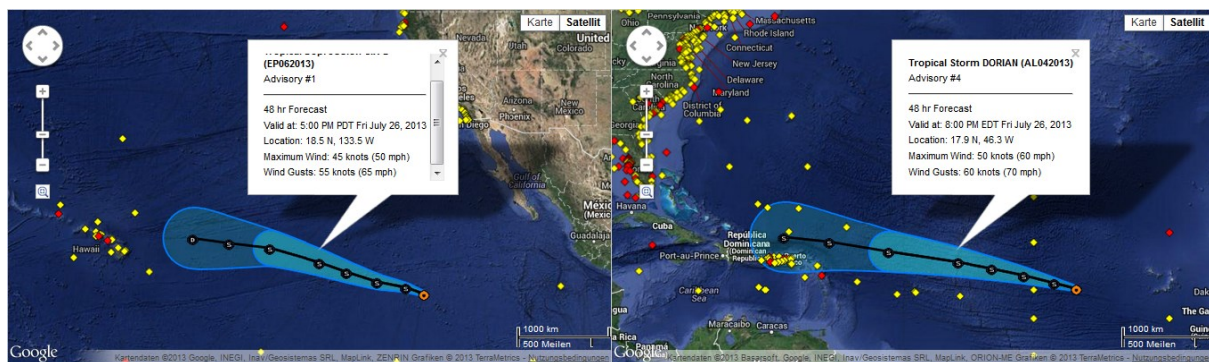


Abbildung 29: Vorhersager einer tropischen Depression links und eines Hurrikans rechts (NOAA, 2013)

3.6 OSIRIS

Das GMES (*Global Monitoring for Environment and Security*) Projekt *Open architecture for Smart and Interoperable networks in Risk management based on In-situ Sensors* (OSIRIS) ist durch die europäische Kommission im Rahmen des FP6 teilfinanziert und dient der Definition, Entwicklung und dem Testen von Services für eine Überwachung der Umwelt sowie dem Krisenmanagement im Falle einer Katastrophe. OSIRIS bietet über eine durch Standards des OGC realisierte serviceorientierte Architektur Funktionen zu In-Situ-Erdbeobachtungen und Benutzerservices. Das Konzept wird auf vier echte Projekte der Bereiche Waldbrände, industrielle Risiken, Wasserverschmutzung und Luftverschmutzung angewendet und hat folgende Ziele (OSIRIS, 2009, S. 23 ff.):

- Verbesserung der dauerhaften Überwachung der Umwelt und Krisenmanagement in Echtzeit.
- Umfangreiche Informationen zu den verschiedensten Umweltbereichen beinahe in Echtzeit bieten Endnutzern die Möglichkeit, gleichzeitig über Situationen unterschiedlicher Natur Bescheid zu wissen, diese zu evaluieren (Wettervorhersagen, Wasser, Luft, etc.) und Krisensituationen zu meistern.
- Verbesserung der Sicherheit der Bürger und der Rettungsteams.

3 Geo-Sensor Webs (State of the Art)

3.6.1 ARCHITEKTUR

Die in Abbildung 30 dargestellte SOA des OSIRIS dient zum Verwalten verschiedenster Sensortypen. Das dynamische Sensormangement erlaubt es, beliebige Sensoren ins System einzubinden und auftretende Fehler aufzuzeigen. Der Datenzugriff auf akquirierte Daten erfolgt einerseits über einen direkten pull-basierten Modus und andererseits über push-basierte Übertragungsmechanismen (z.B. Veröffentlichen und Abonnieren von Alarmen). Des Weiteren ist eine einfache Integration von OGC oder W3C konformen Webservice Spezifikationen möglich (OSIRIS, 2009, S. 45).

Die implementierten Services werden in folgende drei Gruppen unterteilt (OSIRIS, 2009, S. 46-47):

1. **Sensor Services** – Der Zugang zu den Messdaten der Sensoren und die Kontrolle der Sensoren basiert über webbasierte standardisierte Schnittstellen des SWE.
2. **System Services** – In der *System Service* Schicht ist eine Infrastruktur zur Datenprozessierung (*Sensor Management*) realisiert. Ein Datenspeicher bietet Möglichkeiten für einen standardisierten Zugriff auf die gespeicherten Daten und versteckt die Komplexität der zugrunde liegenden Datenquellen. Der Zugriff auf die Daten erfolgt für Zeitreihen über eine SOS Schnittstelle, für Rasterdaten wird ein WMS oder WCS genutzt. Das Alarmservice bietet Nutzern die Möglichkeit sich zu registrieren, gewünschte Ereignisse zu abonnieren und z.B. Schwellwerte für eine Alarmierung festzulegen. Dafür werden standardisierte Protokolle des OASIS oder andere Eventservices wie das SAS verwendet. Die Benachrichtigung erfolgt über SMS, E-Mail oder *Really Simple Syndication* (RSS). Als *System of System Service Layer* werden Gruppen von Services bezeichnet, die als Vermittlungsstellen zwischen den einzelnen Services fungieren. Diese Gruppe besteht aus dem *Sensor Instance Registry Service*, das Informationen zu den auf der Sensorebene eingesetzten Services liefert, der *Data Flow Organisation*, die ein Datenproviderservice (SWE oder Webservice) als Datenquelle nutzt, um den Datenspeicher zu füllen und aus *Integrationsmittel*, welche die Integration von zusätzlichen Diensten erleichtern sollen.
3. **Operational Services** – *Operational Services* dienen zur Darstellung von Sensordaten und Informationen aus den Systemdiensten. Bestimmte Applikationen werden zur

3 Geo-Sensor Webs (State of the Art)

Überwachung von bestimmten Umweltprozessen und zur Nutzbarmachung der Services (z.B. Datendarstellung, Alarmviewers, Steuerung der Sensoren) verwendet.

Die Services liegen in Schichten vor. Das *Sensor Service* befindet sich auf der Sensorseite und durch das *System Service* wird aus dem *Sensor Service* ein Mehrwert generiert. *Operational Services* haben Zugriff auf die beiden unteren Schichten.

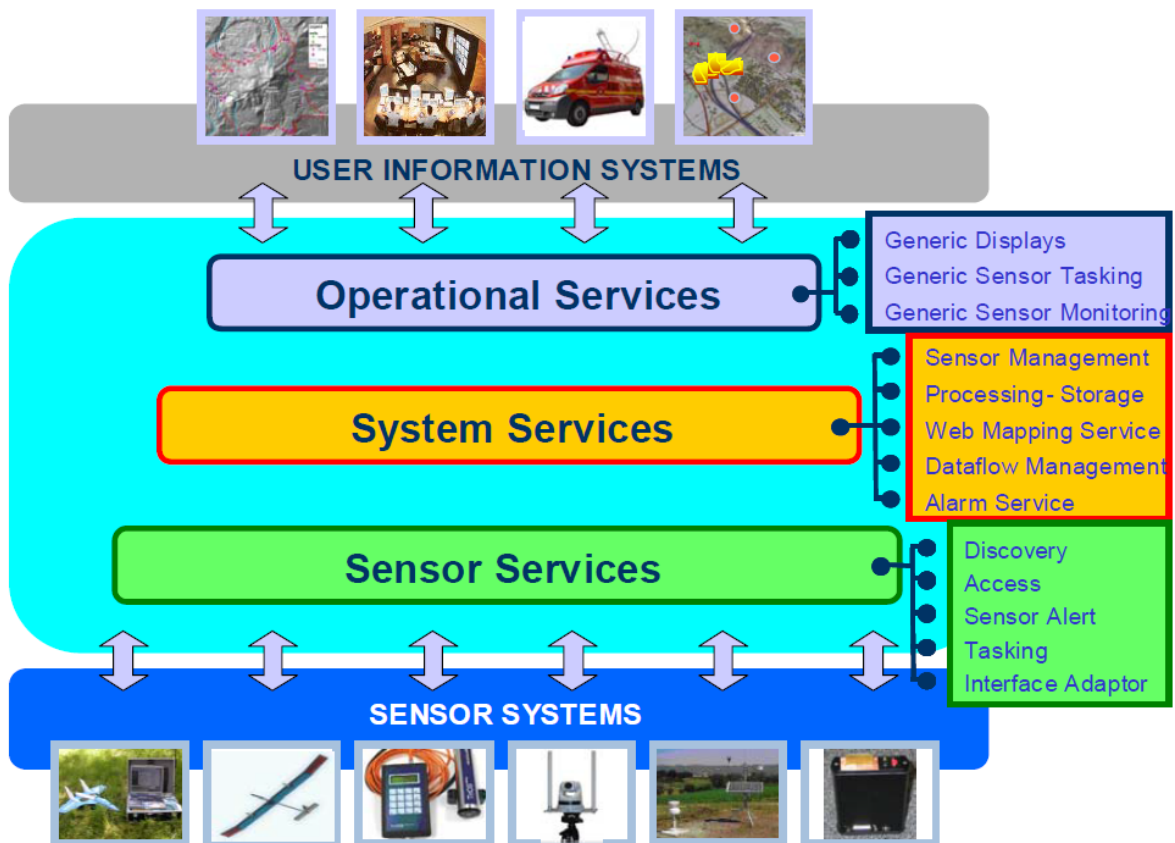


Abbildung 30: Serviceorientierte Architektur (SOA) des OSIRIS (OSIRIS, 2009, S. 46)

Durch die OSIRIS-Architektur sollen folgende Funktionalitäten gewährleistet sein (OSIRIS, 2009, S. 50-51):

Erreichbarkeit – Bestehende Sensornetzwerke sollen durch standardisierte Schnittstellen einen leichten Zugang zu deren Daten und Informationen ermöglichen. So kann ein Endnutzer über eine Service-Schnittstelle die Daten mehrerer Instanzen in sein Gesamtsystem integrieren. Solch ein Zugang wird durch die SWE Schnittstellen und Datenformate ermöglicht.

3 Geo-Sensor Webs (State of the Art)

Standardisierung – Aufgrund der Nutzung der verschiedensten Sensortypen im OSIRIS Projekt und der damit einhergehenden heterogenen Sensordaten, die verschiedenste Arten an Informationen liefern, ist die Notwendigkeit der Standardisierung gegeben. OSIRIS implementiert SWE Spezifikationen für verschiedenste Datentypen, modifiziert diese für die Nutzung mobiler Sensoren und fügt ein *Registry Service* hinzu.

Interoperabilität zwischen verschiedenen Sensoren und Systemen – Interoperabilität wird durch den Einsatz von Standards und durch eine Gestaltung sowie Umsetzung von Verbindungsmechanismen zwischen den Standards untereinander und den verschiedenen Protokollen gewährleistet.

Flexibilität – Flexibilität soll durch Einsatz der vielseitig einsetzbaren In-Situ-Sensoren gewährleistet werden. Dabei werden smarte Sensor-Netzwerke entwickelt, die neue Strategien zum Einsatz und zur Organisation dieser Sensoren (z.B. automatische Konfiguration, Skalierung) unterstützen. Des Weiteren werden Steuerprotokolle für mobile Sensoren basierend auf das SWE verwendet und ein In-Situ-Datenaustausch über das *Data Storage Service* ermöglicht.

Autonomie – Netzwerke, in den In-Situ-Sensoren eingesetzt werden, besitzen die Fähigkeit autonom, also in einer selbstorganisierenden Art und Weise zu arbeiten. Diese Netzwerke sollen ihre Konfiguration automatisch an Veränderungen anpassen können. Beispielsweise könnte bei Eintreten einer Katastrophe eines beobachteten Gebietes die Notwendigkeit gegeben sein, die Abtaste der Sensordaten, oder die räumliche Auflösung zu erhöhen.

3.6.2 TESTIMPLEMENTIERUNGEN DES OSIRIS SYSTEMS

Das OSIRIS System wurde erfolgreich durch Live-Demonstrationen in den vier Bereichen Waldbrände, industrielle Risiken, Wasserverschmutzung und Luftverschmutzung getestet. Das Ziel der Demonstration ist die Validierung des durch die OSIRIS Lösung generierten Mehrwertes im Monitoring- und Krisenbereich unter Einbezug der Endnutzer. Die eingesetzten Sensoren wurden vom Endnutzer oder vom Konsortium zur Verfügung gestellt.

3 Geo-Sensor Webs (State of the Art)

3.6.2.1 WALDBRAND

In Südfrankreich wurden im Jahr 2009 statische und mobile Sensoren zur Beobachtung eines kontrolliert gelegten Waldbrandes verwendet. Eine luftgestützte Plattform diente der Akquisition und georeferenzierten Prozessierung von Videos. Des Weiteren wurde eine Satelliteninternetverbindung eingerichtet sowie Wireless-Kameras und meteorologischen Messstationen installiert. Feuerwehrleute nutzten zur Positionsbestimmung tragbare Sensoren (OSIRIS, 2009, S. 65).

Ziel des Szenarios war es, eine effektive Waldbrandbekämpfung in Südfrankreich zu zeigen. Das Management eines Waldbrandes besteht dabei aus der Monitoring- und der Krisenphase.

In der Monitoring-Phase sollen kritische Bereiche (Risikozonen) und neu entflammte Bereiche identifiziert werden. Dabei wird das Gebiet mit einer am Flugzeug angebrachten hochauflösenden Kamera abgetastet, die ihre Daten zu einem Bodensegment sendet.

Die Krisenphase kennzeichnet die Hauptphase der Demonstration und dient der Benachrichtigung und Alarmierung der Entscheidungsträger und Einsatzkräfte, welche anhand von erarbeiteten Strategien feuerbekämpfende Maßnahmen ergreifen. Die Analysen und Strategien werden auf Basis der Daten, die während der Monitoring-Phase erfasst werden, durchgeführt und entwickelt.

In Abbildung 31 sind die verschiedenen Bereiche des Demonstrationsprojektes dargestellt. Für das Monitoring und Management im Falle eines Waldbrandes in Südfrankreich ist das *Centre Opérationnel Départemental d'Incendie et de Secours* (CODIS) zuständig. CODIS ist eine regulierende Behörde für die Koordinierung der Einsatzkräfte im Falle von Katastrophen.

Folgende Sensoren kamen zum Einsatz (OSIRIS, 2009, S. 66 ff.):

1. **Luftgestützte Sensoren** - Luftgestützte Sensoren dienen zur Verschaffung eines Überblicks zur Situation, Detektion und zeitlichen Entwicklung der Feuerfront, Ermittlung von Bereichen des Interesses (*Areas-Of-Interests*) und zur Erstellung hochauflösender Bilder von Bereichen, in denen Feuerwehrleute in potenzieller

3 Geo-Sensor Webs (State of the Art)

Gefahr sind. Die eingesetzten Standards waren einerseits das SPS zum Steuern der Kameras und andererseits das *Web Catalogue Service* (WCAS) sowie WMS zum Auffinden und Abfragen der rasterbasierten Daten.

2. **Sensoren zur Positionsbestimmung** –Positionssensoren dienten zur Ermittlung der Position der Feuerwehrleute und zum Senden von Alarmen im Notfall. Das Abfragen der Positionsdaten funktionierte über SOS Instanzen, alarmiert wurde über das SAS in Kombination mit dem WNS.
3. **Wireless Kameras** –Wireless Kameras wurden unter anderem zur Überwachung von entlegenen Gebieten, wie evakuierte Zonen, periurbane Zonen, oder strategische Verkehrspunkte verwendet. Als standardisiertes Service wurde das Video SOS verwendet.
4. **Mobile Wetterstationen** – Lokale Informationen zum Wetter wurden über mobile Wetterstationen ermittelt. Die Stationen lieferten über ein SOS die meteorologischen Daten Wind, Temperatur und relative Luftfeuchtigkeit.
5. **Satellitenkommunikationsverbindung** – Für eine Satelliteninternetverbindung sorgte ein auf einem LKW installiertes *very small aperture terminal* (VSAT⁵).

⁵ Das VSAT besteht aus Satellitenempfänger und –sender für eine mobile satellitengestützte Kommunikation.

3 Geo-Sensor Webs (State of the Art)

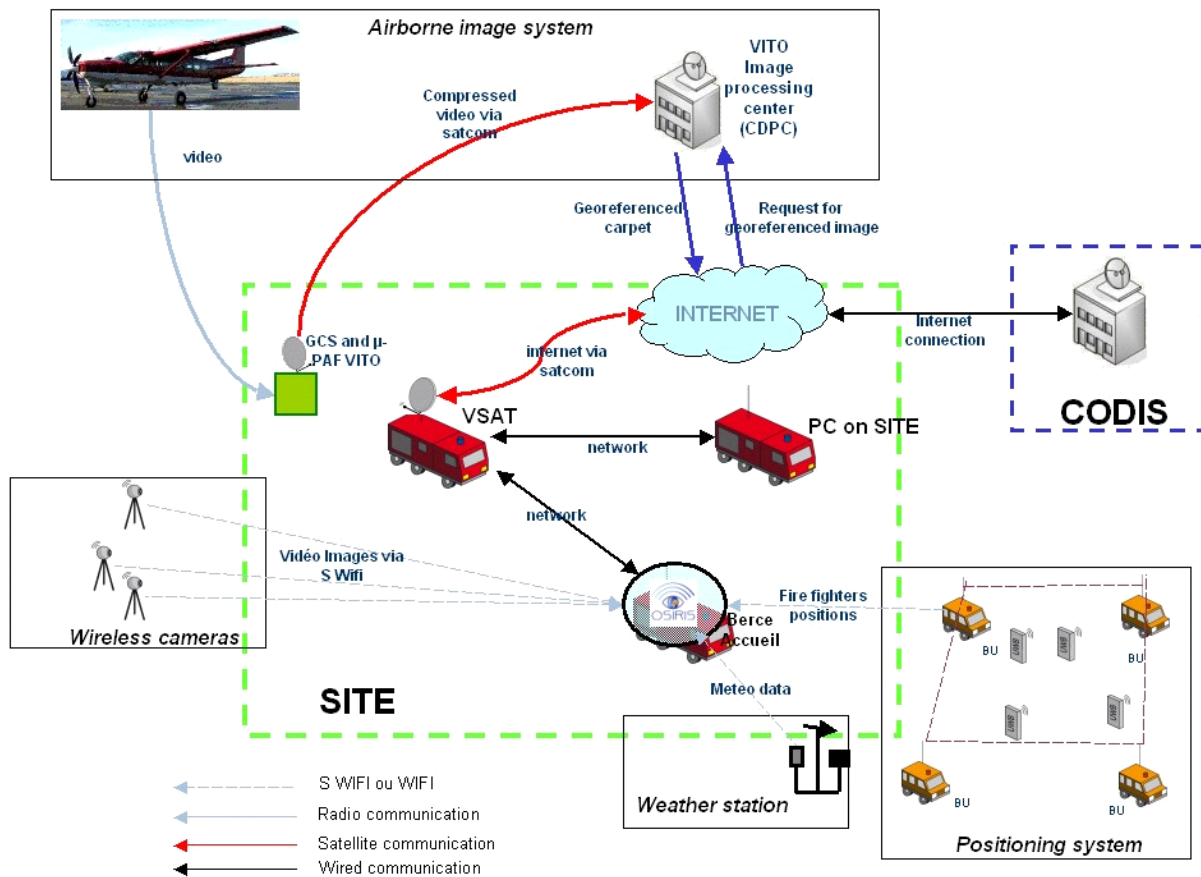


Abbildung 31: OSIRIS Architektur für ein Waldbrand-Management (OSIRIS, 2009, S. 67)

3.6.2.2 FEUER IN INDUSTRIEGEBÄUDEN

Im Szenario „Feuer in Industriegebäuden“ arbeiteten die Feuerwehr Aachen und das *APS European Centre for Mechatronics (APS-ECM)* zusammen, um die Anzahl an Fehlalarmen zu reduzieren und eine zuverlässige Feuerdetektion zu gewährleisten. Die Aachener Feuerwehr besitzt eine Schulungseinrichtung mit drei Räumen zur Brandbekämpfung und zur Simulation von Notfällen in Bezug auf Feuer in Industriegebäuden. Die Sensorknoten wurden mittels Propangas erzeugter Brände getestet.

Ein speziell entwickelter *multi-criteria* Sensor, bestehend aus einem Rauchmelder und einem Temperatursensor wird zur Erkennung von Bränden verschiedenen Typs verwendet. Der Temperatursensor erzeugt Alarme aufgrund eines Grenzwertes und eines Thermo-Differenzials (Temperaturanstieg innerhalb einer bestimmten Zeit). Die Sensorknoten bilden ein Wireless-Netzwerk und kommunizieren untereinander über das *Embernet Zigbee* Protokoll, ein Industriestandard für Funknetze. Für die Parametrierung der Sensorknoten wurde eine graphische Benutzeroberfläche entwickelt. Die Kommunikation zwischen Client

3 Geo-Sensor Webs (State of the Art)

und Service basiert auf dem SPS. Die Alarmparameter sind in der Datenbank des SPS gespeichert und können durch Wählen eines Sensorknoten abgerufen werden (OSIRIS, 2009, S. 144-145).

Die relevanten Sensordaten werden in einem GIS über eine interaktive Karte, welche die Lage der Sensoren und die gemessenen Werte enthält, in Echtzeit dargestellt. Die Web-Applikation in Abbildung 32 zeigt ein detektiertes Feuer inklusive der Adresse des betroffenen Gebäudes. Die Farbe der Marker kennzeichnet den Status der Sensorknoten. Der rot markierte Melder hat ein Feuer am Seffenterweg in Aachen detektiert.

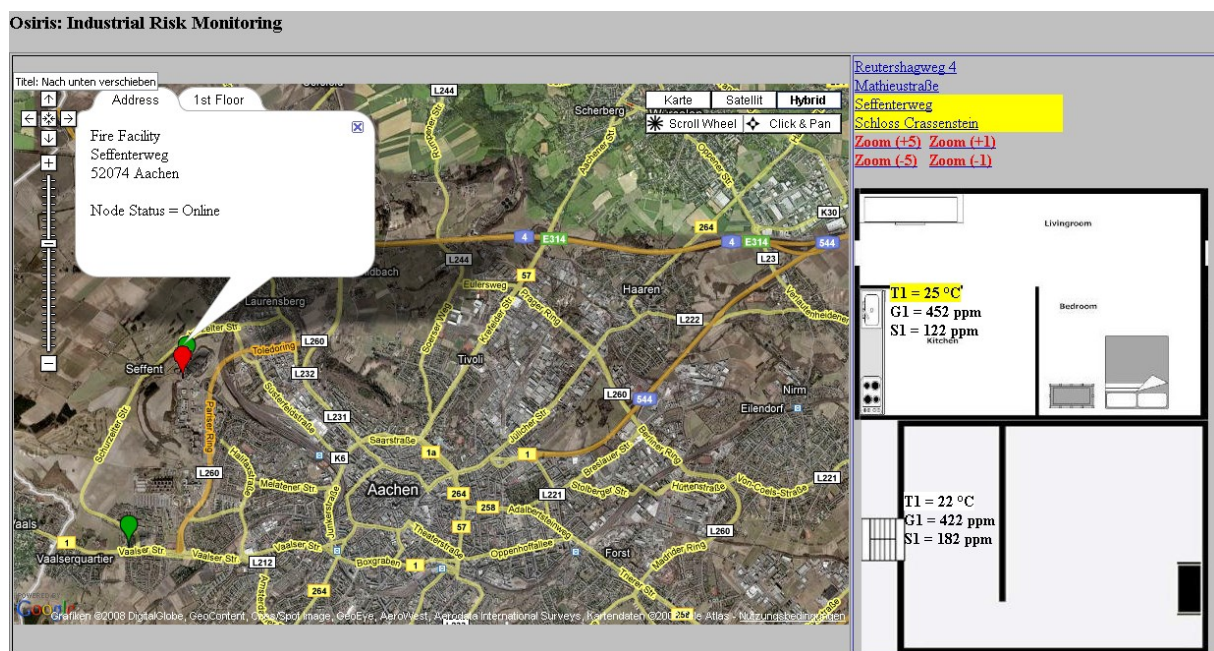


Abbildung 32: Ein GIS zeigt ein detektiertes Feuer (OSIRIS, 2009, S. 146)

3.6.2.3 LUFTVERSCHMUTZUNG

Ziel der OSIRIS Demonstration in Valladolid (Spanien) ist die Überwachung der Luftqualität, sowie eine Alarmierung und Krisenmanagement. Die von den Services des OSIRIS bereitgestellten Sensorinformationen werden zur Bestimmung des Verschmutzungsgrades sowie einer Überwachung der Entwicklung der Schadstoffausbreitung im Falle einer Krisensituation der Stadt Valladolid genutzt.

Aufgrund der schlechten Luftqualität in der Stadt sind fixe Messstationen zur Überwachung der Verschmutzung installiert, welche die Parameter CO, CO₂, NO, NO₂, O₃, SO₂, PM₁₀ und PM_{2,5}, Benzol, Toluol, Xylen und Lärm, sowie die meteorologischen Parameter Luftdruck,

3 Geo-Sensor Webs (State of the Art)

Temperatur, Taupunkt, Luftfeuchtigkeit, Windrichtung und Sonneneinstrahlung messen. Diese Beobachtungen werden in einer Datenbank der Abteilung für Umweltschutz in Valladolid gespeichert und automatisch in ein SOS konformes Format zur interoperablen Nutzung in die OSIRIS Plattform exportiert. Um eine ausreichende Genauigkeit der Modellsimulationen zu gewährleisten, werden die neun fixen Messstationen der Stadt um Datensätze ergänzt, die aus mobilen Messungen generiert werden

Das Szenario selbst wird in ein Air-Monitoring-Szenario und in ein Air-Hazard-Szenario unterteilt (OSIRIS, 2009, S. 149):

1. **Air-Monitoring-Szenario** – Mehrere auf einer Busflotte montierte Sensoren messen die NO und NO₂ Konzentration der Atmosphäre und spezielle Gerätschaften berechnen das Höhenintegral der Daten. Ein On-Board-Kommunikationssystem sendet diese Messergebnisse mit Zeitstempel inklusive der aktuellen Position zu einem Kontrollzentrum.

Abbildung 33 illustriert die Architektur des Monitoring-Szenarios. Alle ermittelten Messwerte der Phänomene der fixen Messstationen, welche in der Datenbank der Abteilung für Umweltschutz gespeichert sind, werden automatisch in die OSIRIS SOS-Datenbank geschrieben. Die in den Bussen montierten Sensoren besitzen eine eindeutige ID. Spezielle Routen werden vom Kontrollzentrum im Voraus geplant und sind so festgelegt, um für einen großen Bereich der Stadt Emissionswerte zu erhalten. Des Weiteren überwacht das Kontrollzentrum die GPS-Position und den Status der Busse und ist für die Funktionsfähigkeit der Flotten verantwortlich. Im Monitoring-Zentrum werden die Umweltdaten der fixen Messstationen mit den durch die mobilen Sensoren erzeugten Messwerten kombiniert. Diesen mobil akquirierten Daten werden in den *On Board Units* (OBA) die aktuelle GPS-Position sowie der Zeitstempel beigefügt und mittels *General Packet Radio Service* (GPRS) zum Kontrollzentrum gesendet (OSIRIS, 2009, S. 155 ff.).

3 Geo-Sensor Webs (State of the Art)

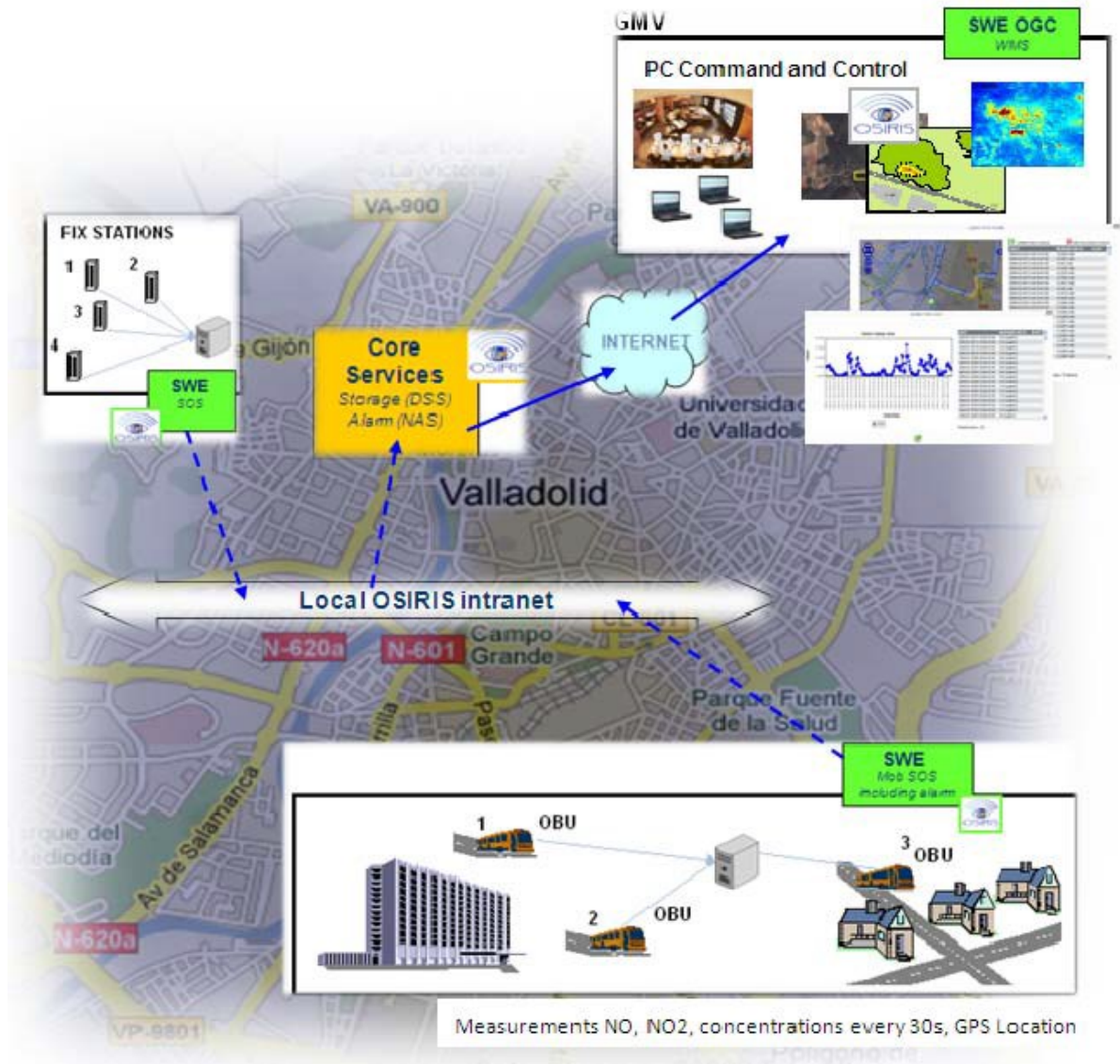


Abbildung 33: Architektur des Monitoring-Szenarios (OSIRIS, 2009 S. 155)

- Air-Hazard-Szenario** – Für das Hazard-Szenario wurde ein Unfall eines LKWs, der mit giftigen, sich in der Atmosphäre verbreitenden Stoffen beladen wurde, simuliert. Ein *unmanned aerial vehical* (UAV) ermittelte die Stabilitätsbedingungen der Atmosphäre im betroffenen Gebiet. Die Daten wurden als Modellinput zur Modellierung einer Schadstofffahne verwendet. Gleichzeitig führten Mitglieder des Katastrophenschutzes ein Krisenmanagement unter realitätsnahen Bedingungen durch. Das Testszenario wurde im Technologiepark von Boecillo, einem Nebenort ca. 15 Kilometer von Valladolid entfernt durchgeführt. Für das Krisenmanagement wurde ein Tool entwickelt, welches Informationen zur Krisensituation liefert und es

3 Geo-Sensor Webs (State of the Art)

ermöglicht, die Modellberechnungen in ein GIS zu importieren, um die vorhergesehene Verbreitung des Gases zu visualisieren (OSIRIS, 2009, S. 160 ff.).

3.6.2.4 MONITORING DER WASSERQUALITÄT

Das Ziel der Demonstration des Grundwassermanagementsystems basierend auf der OSIRIS Architektur ist die Sicherstellung der Versorgung des Trinkwassers mit guter Qualität. Als Untersuchungsgebiet wurde der Grundwasserleiter des Amiata, ein Berg vulkanischen Ursprungs in Italien gewählt, dessen Wasser einen erhöhten Gehalt an Spurenelementen wie Arsen, Bor und Quecksilber aufweist. Das Projekt wurde von der *Region Toscana* (RT), dem *Acquedotto del Fiora* (AdF) und dem *Institut für Biometeorologie des nationalen Research Council* (CNR-Inimet) finanziert. Im System sind entscheidungsunterstützende Werkzeuge für das Managen der Wasserverteilung über ein Aquädukt implementiert. Unterschiedlichste Sensoren mit heterogenen Eigenschaften sind entweder fix montiert oder werden während kritischen Verschmutzungsereignissen an speziellen Positionen eingesetzt (OSIRIS, 2009, S. 172 ff.).

Die Webservices des OSIRIS ermöglichen ein kontinuierliches Speichern und Visualisieren der Daten zur Durchführung von Offline-Analysen, um mittels Korrelationen das Verhalten des Grundwasserleiters zu verstehen. Des Weiteren können Endnutzer automatisch über standardisierte Services Alarme empfangen und entsprechend auf beispielsweise eine Wasserverschmutzung reagieren (OSIRIS, 2009, S. 173).

Im Zuge der Demonstration am 18. Februar 2009 wurden die folgende zwei Szenarien *Arsenic pollution* und *Tanker-track accident Szenario* für die Grundwasserapplikation angenommen (OSIRIS, 2009, S. 178 ff.):

1. **Arsenic pollution Szenario** – Das *Arsenic pollution Szenario* zählt zur Monitoring-Phase des Grundwassers und basiert auf einem System, welches den Arsengehalt im Wasser ermittelt und auf den gesetzlich vorgegebenen maximalen Grenzwert von 10 *parts per billion* (ppb) überwacht. Außerdem werden Variationen der Durchflussmengen und deren Auswirkungen auf die Arsenkonzentration vorhergesagt.

3 Geo-Sensor Webs (State of the Art)

2. **Tanker-track accident scenario** – Im *Tanker-track accident scenario* wird ein chemisches Krisenmanagement aufgrund eines Tankerunfalls in der Nähe einer Frischwasserquelle, die das Aquädukt versorgt, simuliert. Im Szenario wird einerseits ein Monitoring-System aus Sensoren zu Messung des Kohlenwasserstoffgehalts zur Bestimmung der Kontamination und andererseits ein Modell für Prognosen zum Schadstoffverhalten verwendet.

Abbildung 34 zeigt die Verwendung der einzelnen standardisierten Webservices der OSIRIS Architektur bezüglich des Demonstrationsprojektes zur Wasserqualität. Die Abfragen aller mittels Sensoren erzeugter Messwerte erfolgen über SOS Schnittstellen. Ein *Notification und Alert Service* (NAS) alarmiert bei Überschreitung des Arsen- oder Kohlenwasserstoffgehalts jene Nutzer, die Alarme abonniert haben. Sämtliche Messwerte zur Wasserqualität und -quantität werden im AdF Kontrollzentrum über verschiedenste Protokolle (GSM, FTP, VPN) abgefragt und gespeichert. Aufgrund der komplexen Dynamik des Wasserkreislaufes und des Schadstoffverhaltens ist im System ein Hydrologisches/Hydro-Geologisches Modell (siehe Model Unit aus Abbildung 34) für Nowcast und Forecast-Analysen integriert. Diese prozessierten Informationen werden zum Osiris-System gesendet, das die gesammelten Daten in Echtzeit in einer GIS basierten Web-Applikation darstellt.

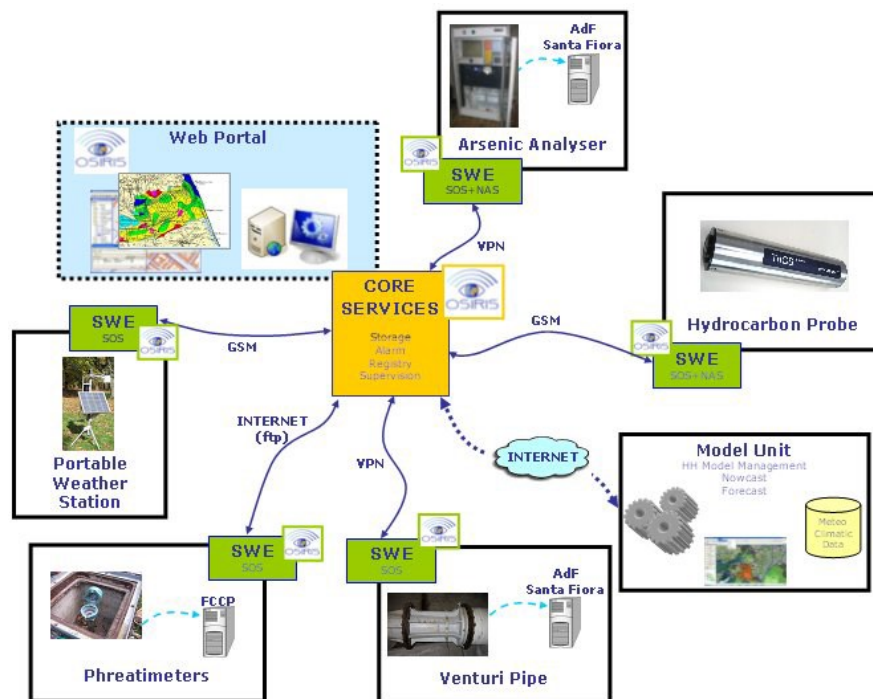


Abbildung 34: Einsatz der OSIRIS Services im Wasser-Monitoring-Demonstrationsprojekt (OSIRIS, 2009, S. 189)

4 SOS-IMPLEMENTIERUNG DER INSTANZEN WEGENERNET UND ZAMG

Dieses Kapitel widmet sich dem praktischen Teil der vorliegenden Arbeit und beschreibt die Umsetzung der Implementierung der SOS Instanzen *WegenerNet* des Wegener Centers und eine Implementierung mit Testdaten der Zentralanstalt für Meteorologie und Geodynamik (ZAMG).

Das Wegener Center sowie die ZAMG betreiben jeweils ein Stationsnetz, welche über verschiedene heterogene Sensoren meteorologische Parameter zur Untersuchung der Wetter- und Klimaentwicklung messen. Die beiden voneinander getrennten Systeme arbeiten mit unterschiedlichen Datenformaten, Übertragungsprotokollen, Strukturen der Datenspeicherung, etc. wodurch keine Möglichkeit einer optimalen gemeinsamen Nutzung der Sensordaten gegeben ist. Aufgrund fehlender Standards ist eine Integration und Vernetzung der Sensoren, deren Zeitreihen und Metadaten nicht uneingeschränkt realisierbar bzw. verlangt es ein bestimmtes Maß an Aufwand. Durch die Nutzung der SOS Spezifikation können diese Daten über System- und Domänengrenzen hinweg verwendet werden und in einfacher Weise mit Daten anderer Geo-Netzwerke, die OGC Standards verfolgen, kombiniert werden.

4.1 52°NORTH SOS SERVER

Der in der Forschungsarbeit verwendete SOS Server wurde von der Open-Source-Software-Initiative 52°North entwickelt. Die Initiative besteht aus einem offenen internationalen Netzwerk aus Partnern der Forschung, Industrie und öffentlichen Verwaltung. Die verwendete SOS Version 3.5.0 wurde in JAVA entwickelt, unterstützt die SOS Implementierungsspezifikation 1.0.0, sowie Teile der SOS 2.0 Spezifikation. In der Applikation wird zusätzlich auf folgende technische Frameworks zurückgegriffen:

- **PostgreSQL inklusive PostGIS Extension** – Im objektrelationalen Datenbankmanagementsystem (ORDMS) PostgreSQL wird das Datenmodell

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

basierend auf dem O&M Schema abgebildet. Die PostGIS Erweiterung dient zur Verwaltung von Geodaten und unterstützt *Simple Features* gemäß OGC Spezifikation, den Import vieler GIS Formate sowie gängige GIS-Analysefunktionen.

- **Apache XMLBeans** – Apache XMLBeans bindet Daten aus einem XML-Schema heraus automatisch an Java-Klassen und wird als XML-Parser für Anfragen und Antworten eines Clients verwendet.
- **Java Topology Suite (JTS)** – JTS ist eine in Java geschriebene freie Programm-bibliothek, die räumliche 2D Funktionen zur Verfügung stellt.
- **Logback** – Logback erzeugt die Log-Files.
- **Apache Maven** – Apache Maven wird als Build-Management-Tool der SOS Java Applikation verwendet.

Das Design des 52°North Sensor Observation Service besteht aus einer vier-Schichten Web-Architektur, wie in Abbildung 35 dargestellt (Hollmann, 2012, S. 7-8):

1. **Data Layer** – Die unterste Schicht stellt die Verbindung zur Datenbank und anderen Datenquellen her. Die heterogenen Quellen der Sensordaten reichen von einer einfachen Textdatei bis hin zu komplexen Datenmodellen. Um den Zugriff auf unterschiedliche Datenquellen zu ermöglichen werden in der Applikation *Data Access Objects (DAO)* verwendet. Ein DAO kapselt die Datenquelle so, dass die angesprochene Datenquelle ausgetauscht werden kann, ohne den Source Code ändern zu müssen. Standardmäßig nutzt das 52°North SOS eine PostGIS Datenbank um *Observations* und die zugehörigen Metadaten zu speichern.
2. **Business Logic** – Der RequestOperator der Business Logic Schicht empfängt Anfragen von der Präsentationsschicht, validiert die Anfragen und leitet diese dem zugehörigen GetObservationListener weiter. Für jede unterstützte Operation liegt ein *Listener* vor, welcher in einem externen Konfigurationsfile definiert ist. Alle *Listeners* besitzen eine standardisierte Schnittstelle. Sollen zusätzliche Operationen unterstützt werden, so müssen der *OperationListener* sowie das zugehörige DAO implementiert und im Konfigurationsfile eingetragen werden. Weiter Komponenten des *Business Logic Tiers* sind beispielsweise Komponenten zum Parsen bzw. Kodieren der Serverantworten.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

3. **Web Tier** – Die HTTP Anfragen und Antworten werden über den Web Tier, welcher als Servlet implementiert ist verwaltet.
4. **Clients** – Alle Clients (Thin- und Thick-Clients), die das SOS Protokoll übersetzen können, können auch auf das SOS zugreifen.

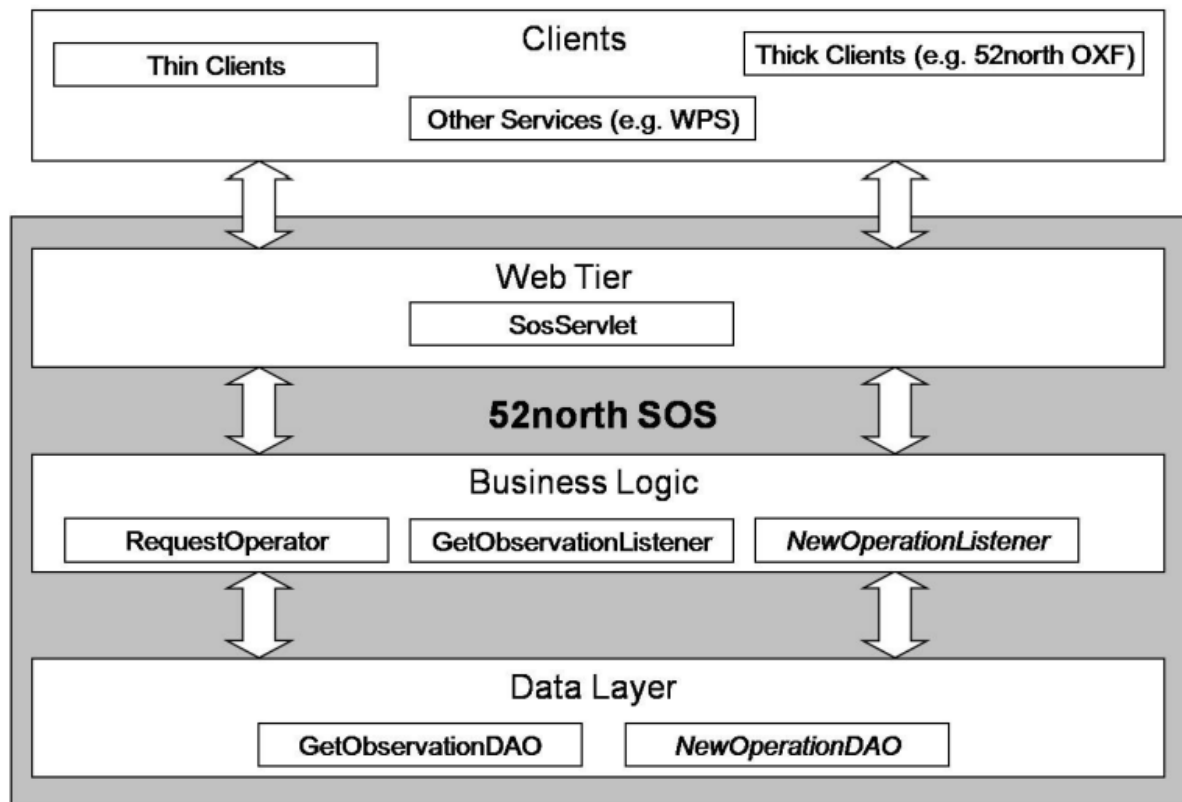


Abbildung 35: 52°North SOS Architektur (Hollmann, 2012)

Die Web-Applikation selbst wird im Open Source-Servlet-Container *Apache Tomcat* eingesetzt. Die von der SOS 3.5.0 Implementierung des 52°North unterstützten SOS 1.0.0 Operationen sind (siehe Kapitel 2.6):

- Das zwingend umzusetzende Kernprofil (*GetCapabilities*, *DescribeSensor* und *GetObservation*).
- *GetFeatureOfInterest*: Liefert Informationen zu Features. Der Abfrage wird unter anderem die ID des *FeatureOfInterest* übergeben.
- *GetResult*: Über *GetResult* kann ein Client aktuelle Daten desselben Sensors abfragen, ohne den ganzen Workflow der Abfragen ständig zu wiederholen (siehe Kapitel 2.7.3).

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

- *GetObservationById*: *GetObservationById* ermöglicht das Abfragen einer Observation über deren Identifier.

Die von der SOS 3.5.0 Implementierung des 52°North unterstützten SOS 2.0 Operationen sind:

- Das zwingend umzusetzende Kernprofil (*GetCapabilities*, *DescribeSensor* und *GetObservation*).
- *GetFeatureOfInterest*: Liefert Informationen zu Features. Der Abfrage wird unter anderem die ID des *FeatureOfInterest* übergeben.

Wie in Kapitel 4.1 beschrieben, wird das SOS Schema in einer PostgreSQL Datenbank mit der PostGIS Erweiterung abgebildet. Das *Entity-Relationship-Modell* (ER-Modell) aus Abbildung 36 zeigt die in der Datenbank erstellten Tabellen und die Beziehungen dieser untereinander. Alle Tabellen der grauen Box werden vom SOS zur Speicherung von Request-Parametern für spätere Anfragen mittels *getResult* Operationen genutzt. Die restlichen Datentabellen werden zur Beantwortung der eingehenden Anfragen oder zur Aktualisierung bestimmter Werte genutzt. Folgende Tabellen werden vom 52°North SOS 3.5.0 verwendet (Hollmann, 2012, S. 14-16):

1. **Feature_of_interest Tabelle** – In der *feature_of_interest* Tabelle sind Daten zu den *FeatureOfInterests* gespeichert. Die Geometriespalte beinhaltet die Koordinaten der Klimastationen im definierten Koordinatensystem.
2. **Foi_off Tabelle** – Die many-to-many Beziehung zwischen den *FeatureOfInterests* und den *Offerings* wird über die *foi_off* Tabelle realisiert.
3. **Proc_foi** – Über die *proc_foi* Tabelle wird die *many-to-many* Beziehung der *Prozedures* zu den *FeatureOfInterests* umgesetzt.
4. **Observation Tabelle** – In der *Observation* Tabelle werden die *Observations Events* mit Zeitstempel und den ID's der zugehörigen *Procedures*, *FeatureOfInterests* und *Phänomene* gespeichert. Die ID's sind als Fremdschlüssel definiert, müssen also in anderen Tabellen schon vorhanden sein.
5. **Quality Tabelle** – Die optional zu füllende Tabelle *quality* wird zum Speichern für Qualitätsattribute zu den *Observations* genutzt.
6. **Procedure Tabelle** – In der *Procedure* Tabelle wird die ID als *Uniform Resource Name* (URN), sowie eine Beschreibung der Prozedur gespeichert.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

7. **Proc_off Tabelle** – Über die *proc_off* Tabelle wird die *many-to-many* Beziehung der *Prozedures* zu den *Offerings* umgesetzt.
8. **Offering Tabelle** – Die *Offering* Tabelle beinhaltet alle *Offerings* eines SOS. Änderungen der Tabelle werden nur bei Neustart des Servers übernommen. Zugehörige *Procedures*, Phänomene und *Feature-Of-Interest* werden über Fremdschlüssel ermittelt.
9. **Phen_off Tabelle** – Die *phen_off* Tabelle realisiert *many-to-many* Beziehungen zwischen den Phänomenen und den *Offerings*.
10. **Composite_phenomenon Tabelle** – In der *composite_phenomenon* Tabelle werden zusammengesetzte Phänomene gespeichert.
11. **Com_phen_off Tabelle** – Die *com_phen_off* Tabelle setzt *many-to-many* Beziehungen zwischen zusammengesetzten Phänomenen und den *Offerings* um.
12. **Phenomenon Tabelle** – Die *Phenomenon* Tabelle beinhaltet Informationen zu observierten Eigenschaften, die im Kontext der SOS Spezifikation auch als *observedProperties* bezeichnet werden. Die ID eines Phänomens sollte den URN laut OGC Spezifikation beinhalten.
13. **Proc_phen Tabelle** – Über die *proc_phen* Tabelle werden die *many-to-many* Beziehungen der *Procedures* zu Phänomenen umgesetzt.
14. **Proc_off Tabelle** – Die *proc_off* Tabelle beinhaltet *many-to-many* Beziehungen der *Procedures* zu den *Offerings*.
15. **Request Tabelle** – Die Request Tabelle wird für die *getResult* Operation verwendet und realisiert *many-to-many* Beziehungen zwischen den *Procedures* und den *Requests*.
16. **Request_phenomenon Tabelle** – Die Request Tabelle wird nur für die *getResult* Operation verwendet und realisiert *many-to-many* Beziehungen zwischen den Phänomenen und den *Requests*.
17. **Request_composite_phenomenon Tabelle** – Die *request_composite_phenomenon* Tabelle wird zum Speichern der abgefragten, zusammengesetzten Phänomene für die *getResult* Operation verwendet.
18. **Observation_template Tabelle** – Die *observation_template* Tabelle realisiert die *many-to-many* Beziehung zwischen den *Procedures* und den *Requests* für die *getResult* Operation.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

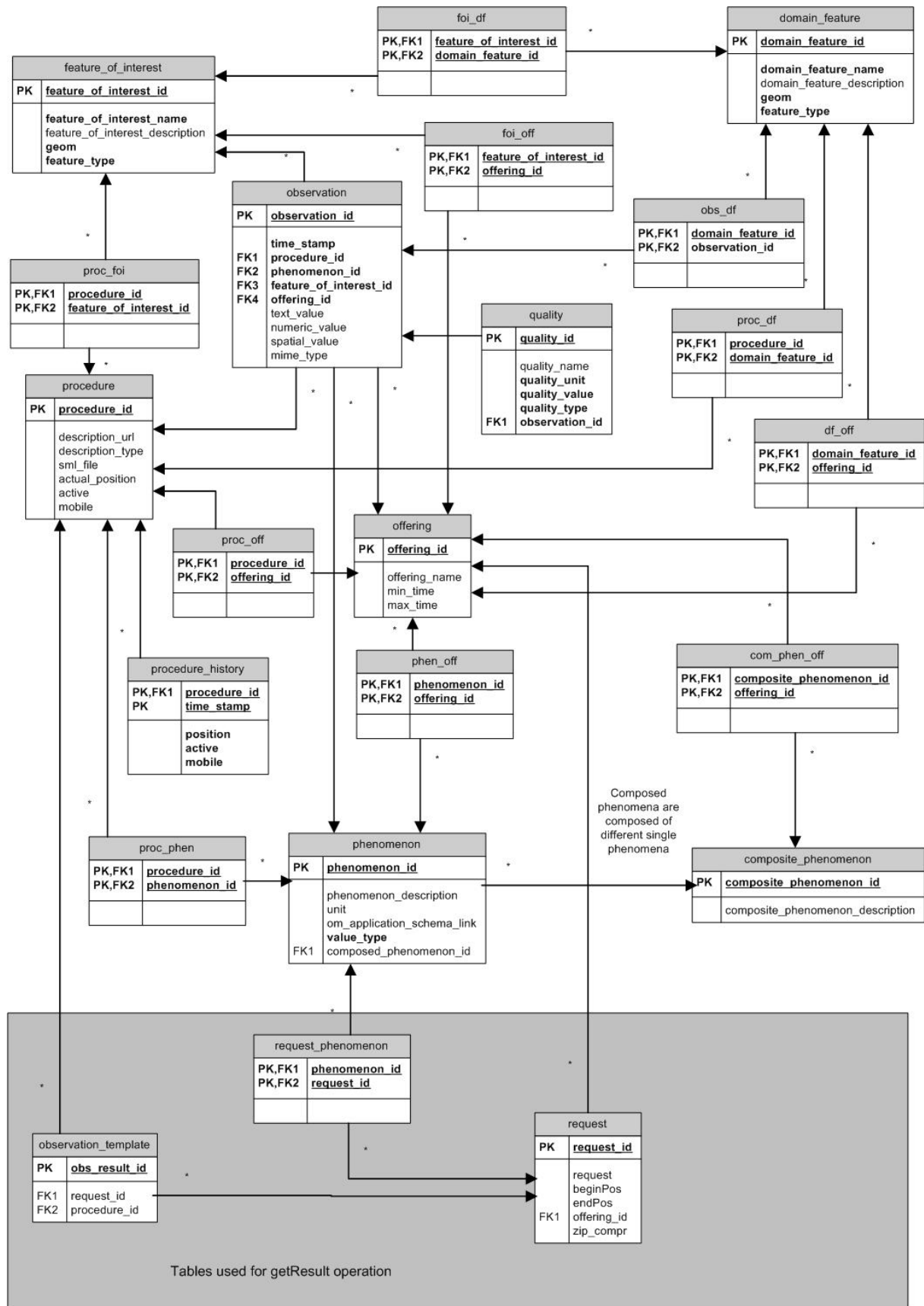


Abbildung 36: Datenbankschema der SOS 52°North Version 3.5.0 (Hollmann, 2012, S. 14)

4.2 WEGENERNET SOS

4.2.1 ALLGEMEINES

Das sich im südoststeirischen Bezirk Feldbach (Österreich) befindende Klimastationsnetz *WegenerNet* umfasst 151 Messstationen zur Langzeituntersuchung der kleinregionalen Wetter- und Klimaentwicklung. Die Stationen sind auf einer Fläche von ca. 20 km x 15 km im Abstand von ca. 1,4 km verteilt. Alle Messwerte der Parameter werden in einer zeitlichen Auflösung von fünf Minuten per Internet Logger über GPRS an den Datenserver übertragen und dort in einer PostgreSQL Datenbank gespeichert. An jeder der 151 Stationen sind Sensoren zum Messen der Hauptparameter Lufttemperatur, relative Luftfeuchtigkeit und Niederschlag montiert. Ausgesuchte Stationen messen zusätzlich Wind-, sowie Bodenparameter. Der Luftdruck und die Strahlungsbilanz werden zusätzlich in einer zentral gelegenen Klimastation beobachtet (Kabas, 2011, S. v).

In jeder der in Abbildung 37 dargestellten Rasterzellen ist eine Messstation positioniert. Explizit werden nur die Standorte mit Sensoren zur Messung der Wind- und Bodenparameter gezeigt.

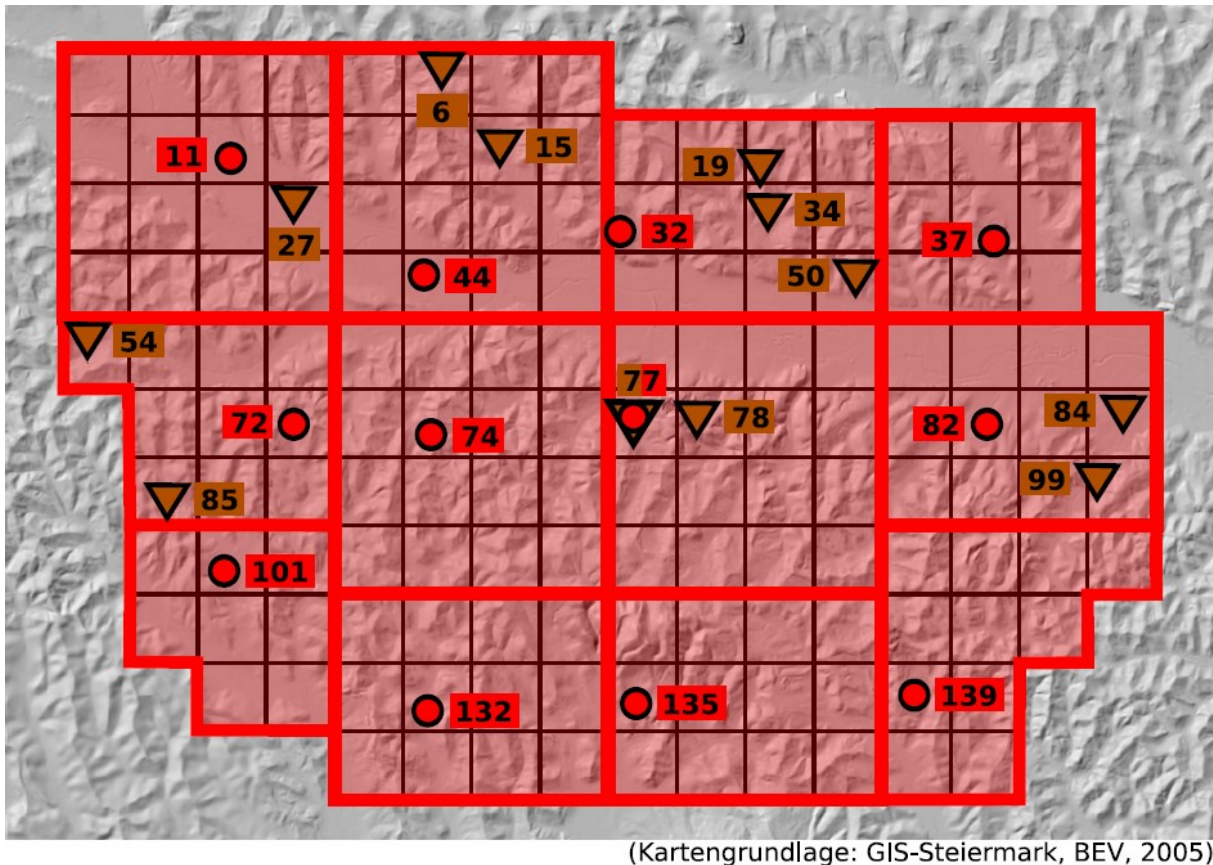
Um auf den Wartungszustand einer Station schließen zu können, werden die GSM-Signalstärke des GSM-Moduls, die Schrank-Feuchtigkeit sowie die Batteriespannung, Versorgungsspannung, Temperatur und die Speicherbelegung des Datenloggers zur Zentrale übertragen.

Aufgrund der Gruppierung der Stationen, die mittels Sensoren Grundparameter und gewählte Zusatzparameter messen, sind folgende vier Stationstypen definiert (Kabas, 2011, S. 38):

1. **Basisstation** – 128 Basisstationen messen mittels Kombisensor Lufttemperatur, relative Luftfeuchtigkeit und Niederschlag.
2. **Basisspezialstation** – Insgesamt sind 12 Klimastationen als Basisspezialstationen definiert, die wiederum in zwei Subtypen unterteilt sind. 11 Stationen des Subtyps A messen zusätzlich zu den Hauptparametern mittels Bodensensor den pH-Wert und die Bodentemperatur. Die zwölfte Basisspezialstation des Subtyps B misst im Gegensatz zu den Basisstationen nur die Lufttemperatur und die relative Luftfeuchtigkeit.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

3. **Primärstation** – Alle 11 Primärstationen führen zusätzlich zu den Hauptparametern Windmessungen durch.
4. **Referenzstation (ID 77)** – Die Referenzstation ist mit dem größten Umfang an Sensorik ausgestattet und misst als einzige Station die Strahlungsbilanz und den Luftdruck zusätzlich zu den Haupt-, Boden- sowie Windparametern.



- Standort mit Messung der Windparameter
- ▼ Standort mit Messung der Bodenparameter
- Teilsegment mit Standort zur Windmessung
- Stationsraster

0 1 2 5 Kilometer



Abbildung 37: Anordnung der Klimastationen mit Erhebung der Windparameter und Verteilung der Bodenmesspunkte im WegenerNet. Darstellung auf Geländeschattierung des Untersuchungsgebiets (Kabas, 2011, S. 50)

Für die Verarbeitung der Daten wurde ein eigenes *WegenerNet Processing System* (WPS), bestehend aus vier Subsystemen, entwickelt. Der Ablauf der Datenprozessierung ist in Abbildung 38 schematisch dargestellt. Die Level 0 Prozessierung beinhaltet die Übertragung

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

der Daten der Messstationen zum Datenserver, das Auslesen der in den Files gespeicherten Daten und die Übertragung dieser in die PostgreSQL Datenbank. Die erhobenen Rohdaten werden danach im Zuge des Level 1 Prozesses in sieben Kontrollschichten auf ihre Plausibilität überprüft und mit einem *Quality Control Flag* (QC-Flag) gekennzeichnet. Der *Data Product Generator* (Level 3) schafft unter Verwendung der mit höchster Qualität ausgezeichneten Daten durch Interpolation flächenhaft verteilte Felder, wobei hier Produkte mit unterschiedlicher zeitlicher Auflösung erzeugt werden (z.B. Halbstundenmittelwerte, Jahresmittelwerte). Der webbasierte Zugriff und die Darstellung der Daten erfolgt über das Visualisierungs- und Informationssystem (Kabas, 2011, S. 74).

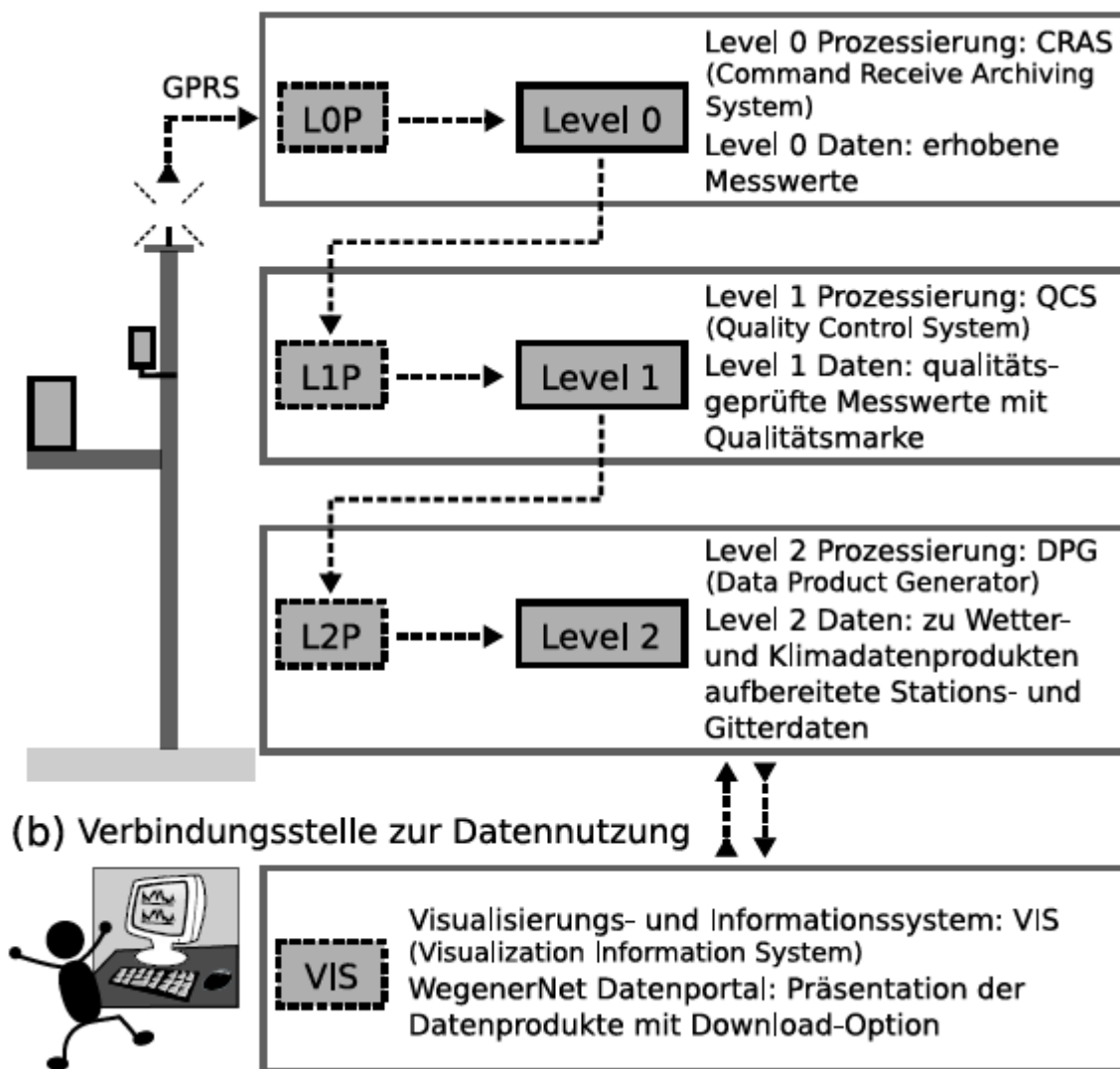


Abbildung 38: Schematische Darstellung des WegenerNet Prozessierungssystems: (a) Kernsystem der Datenprozessierung und (b) Verbindungsstelle zur Datennutzung (Kabas, 2011, S. 75)

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

4.2.2 IMPLEMENTIERUNG IM SOS DATENMODELL

Die Implementierung der WegenerNet Daten in die PostgreSQL Datenbank erfolgte nach dem in Abbildung 36 dargestellten 52°North SOS Datenbankschema. Bevor *Observations* eingefügt werden können, müssen alle zugehörigen *FeatureOfInterests*, *ObservedProperties*, *Procedures*, *Offerings* und ihre Beziehungen untereinander in der Datenbank vorhanden sein. Diese Beziehungen werden im folgenden Kapitel beschrieben.

4.2.2.1 BESCHREIBUNG DER BEZIEHUNGEN ANHAND EINER BASISSTATION

Abbildung 39 zeigt eine Basisstation mit einem Kombisensor zur Messung der Temperatur und Luftfeuchtigkeit sowie die unbeheizte Version eines Niederschlagsgebers, der die Daten mit dem Prinzip der Kippwaage erhebt. Des Weiteren sind die auf URN basierenden umgesetzten Bezeichnungen nach SOS Spezifikation der Klimastation 10 dargestellt. Die meteorologische Messstation kennzeichnet das *FeatureOfInterest* mit der ID *org:wegcenter:WegenerNet_Klimastation_10*. Dieses *Feature-Of-Interest* beinhaltet folgenden Sensoren (*Procedures*):

- *org:wegcenter:temperature_2m_Klimastation_10* – Temperaturmessung zwei Meter über Grund
- *org:wegcenter:relative_humidity_2m_Klimastation_10* – Messung der relativen Feuchtigkeit zwei Meter über Grund
- *org:wegcenter:precipitation_Friedr_Klimastation_10* – Niederschlagsmessung der Firma Friedrichs & CO
- *org:wegcenter:GSM_signal_strength_Klimastation_10* – Messung der GSM- Signalstärke
- *org:wegcenter:logger_battery_voltage_Klimastation_10* – Messung der aktuellen Batteriespannung des Datenloggers
- *org:wegcenter:logger_humidity_Klimastation_10* – Messung der Luftfeuchtigkeit des Schrankes in dem sich der Datenlogger befindet
- *org:wegcenter:logger_storage_occupancy_Klimastation_10* – Ermittlung des vorhandenen Speicherplatzes des integrierten Speichers im Datenlogger
- *org:wegcenter:logger_supply_voltage_Klimastation_10* – Messung der aktuellen Versorgungsspannung des Datenloggers

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

- org:wegcenter:logger_temperature_Klimastation_10 – Messung der Datenlogger-Temperatur

Die Sensoren der Klimastation 10 mit der ID *org:wegcenter:WegenerNet_Klimastation_10* liefern für folgende *ObservedProperties* Ergebnisse:

- urn:ogc:def:property:temperature – Lufttemperatur
- urn:ogc:def:property:relative_humidity – Relative Luftfeuchtigkeit
- urn:ogc:def:property:precipitation – Niederschlag
- urn:ogc:def:property:GSM_signal_strength – GSM-Signalstärke
- urn:ogc:def:property:logger_battery_voltage – Logger-Batteriespannung
- urn:ogc:def:property:logger_humidity – Logger-Schrankfeuchtigkeit
- urn:ogc:def:property:logger_storage_occupancy – Logger-Speicherplatz
- urn:ogc:def:property:logger_supply_voltage – Logger-Versorgungsspannung
- urn:ogc:def:property:logger_temperature – Logger-Temperatur

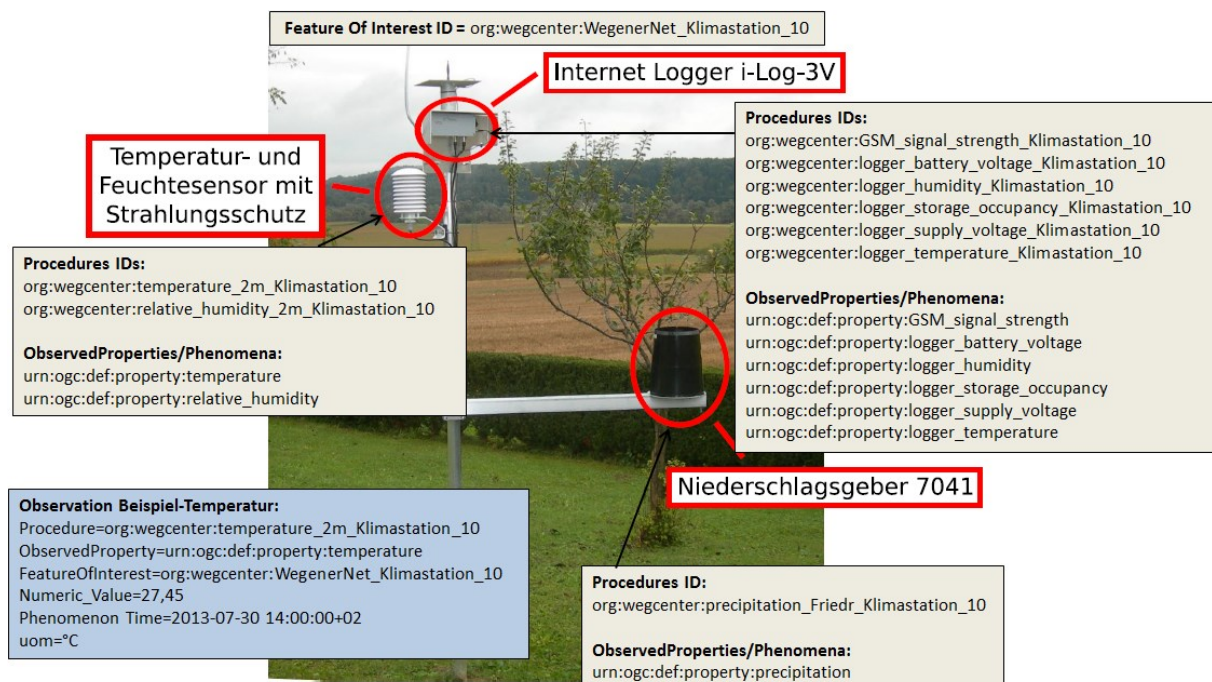


Abbildung 39: Installierte Sensorik einer Basisstation und umgesetzte Bezeichnungen der Klimastation 10 nach der SOS Spezifikation (nach Kabas, 2011 S. 39)

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

Beispiel einer Observation zur Ermittlung eines Temperaturwertes (blaue Box in Abbildung 39) – Die Observation zur Erfassung der Lufttemperatur ermittelte am 30. Juli 2012 um 14:00:00 (*Phenomenon Time*) für das Objekt der ID *org:wegcenter:WegenerNet_Klimastation_10* mit dem Temperatursensor der ID *org:wegcenter:temperature_2m_Klimastation_10* einen Temperaturwert von 27,45 (*Numeric_Value*) °C (*uom: Units of measurement*) für die Eigenschaft der ID *urn:ogc:def:property:temperature*.

Abbildung 40 zeigt die beispielhafte Realisierung einer *many-to-many* Beziehung zwischen den *Procedures* und den *Feature-Of-Interest* über die Zwischentabelle *proc_foi*. *Proc_foi* enthält die IDs der *Procedures* und den zugehörigen *Feature-Of-Interest* als Fremdschlüssel. Dem *Feature-Of-Interest* mit der ID *org:wegcenter:WegenerNet_Klimastation_10* (*foi* Tabelle) sind neun Sensoren zugeordnet (*procedure* Tabelle).

	procedure_id character varying(100)	procedure Tabelle	description_url character varying(200)
1	org:wegcenter:temperature 2m Klimastation 10		standard/temperature 2m Klimastation 10.xml
2	org:wegcenter:relative humidity 2m Klimastation 10		standard/relative humidity 2m Klimastation 10.xml
3	org:wegcenter:precipitation Friedr Klimastation 10		standard/precipitation Friedr Klimastation 10.xml
4	org:wegcenter:logger supply voltage Klimastation 10		standard/logger supply voltage Klimastation 10.xml
5	org:wegcenter:logger temperature Klimastation 10		standard/logger temperature Klimastation 10.xml
6	org:wegcenter:logger humidity Klimastation 10		standard/logger humidity Klimastation 10.xml
7	org:wegcenter:logger battery voltage Klimastation 10		standard/logger battery voltage Klimastation 10.xml
8	org:wegcenter:logger storage occupancy Klimastation 10		standard/logger storage occupancy Klimastation 10.xml
9	org:wegcenter:GSM signal strength Klimastation 10		standard/GSM signal strength Klimastation 10.xml

	procedure_id character varying(100)	proc_foi Tabelle	feature_of_interest_id character varying(100)
1	org:wegcenter:temperature 2m Klimastation 10		org:wegcenter:WegenerNet Klimastation 10
2	org:wegcenter:relative humidity 2m Klimastation 10		org:wegcenter:WegenerNet Klimastation 10
3	org:wegcenter:precipitation Friedr Klimastation 10		org:wegcenter:WegenerNet Klimastation 10
4	org:wegcenter:logger supply voltage Klimastation 10		org:wegcenter:WegenerNet Klimastation 10
5	org:wegcenter:logger temperature Klimastation 10		org:wegcenter:WegenerNet Klimastation 10
6	org:wegcenter:logger humidity Klimastation 10		org:wegcenter:WegenerNet Klimastation 10
7	org:wegcenter:logger battery voltage Klimastation 10		org:wegcenter:WegenerNet Klimastation 10
8	org:wegcenter:logger storage occupancy Klimastation 10		org:wegcenter:WegenerNet Klimastation 10
9	org:wegcenter:GSM signal strength Klimastation 10		org:wegcenter:WegenerNet Klimastation 10

	feature_of_interest_id [PK] character varying(100)	foi Tabelle	feature_of_interest_name character varying(100)	feature_of_interest_description character varying(200)	geom geome
1	org:wegcenter:Klimatschuetzinger_Station_32		Klimaschuetzinger Station 32	Der Standort befindet sich	01010C
2	org:wegcenter:Wasservers_Grenzl_SO_82		Wasserversorgung Grenzland Su	Die Station befindet sich a	01010C
3	org:wegcenter:WegenerNet_Klimastation_1		WegenerNet Klimastation 1	Der Standort befindet sich	01010C
4	org:wegcenter:WegenerNet_Klimastation_10		WegenerNet Klimastation 10	Die Station befindet sich a	01010C
5	org:wegcenter:WegenerNet_Klimastation_100		WegenerNet Klimastation 100	Station auf einem Huegel no	01010C

Abbildung 40: Beziehungen der procedure, proc_foi und feature_of_interest Tabellen

Die weiteren *many-to-many* Beziehungen werden auf dieselbe Art und Weise über Zwischentabellen mit Fremdschlüsseln realisiert. So werden beispielsweise über zusätzliche Tabellen dem jeweiligen *Feature-Of-Interest*, *Phenomenon*, der *Procedure* die definierten

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

Offerings zugewiesen (Tabellen *foi_off*, *phen_off*, *proc_off*). Den *Procedures* wird das zugehörige Phänomen zugeordnet (Tabelle *proc_phen*) (siehe Abbildung 36).

Wie zuvor beschrieben, müssen beim Einfügen einer *Observation* alle notwendigen Beziehungen und Abhängigkeiten bereits definiert sein. Die in Abbildung 41 rot markierten Einträge zeigen eine *Observation* der Klimastation 10 zur Messung der Lufttemperatur in zwei Meter Höhe über Grund. Bevor der durch eine *Observation* mittels Sensor erzeugte Messwert eingetragen werden kann, müssen die IDs der zugehörigen *Procedures*, des *Feature-Of-Interest* und des Phänomens in den jeweiligen Tabellen (*procedure* Tabelle, *feature_of_interest* Tabelle, *phenomenon* Tabelle) vorhanden sein.

time_stamp timestamp with time zone	procedure_id character varying(100)	feature_of_interest_id character varying(100)	phenomenon_id character varying(100)	text numeric, numeric
17	2013-07-30 14:00:00 org:wegcenter:logger_supply_voltage_Klimastation_10	org:wegcenter:WegenerNet_Klimastation_10	urn:ogc:def:property:logger_supply_voltage	2.911
18	2013-07-30 14:00:00 org:wegcenter:logger_temperature_Klimastation_10	org:wegcenter:WegenerNet_Klimastation_10	urn:ogc:def:property:logger_temperature	33.39
19	2013-07-30 14:00:00 org:wegcenter:logger_humidity_Klimastation_10	org:wegcenter:WegenerNet_Klimastation_10	urn:ogc:def:property:logger_humidity	24.1
20	2013-07-30 14:00:00 org:wegcenter:precipitation_Friedr_Klimastation_10	org:wegcenter:WegenerNet_Klimastation_10	urn:ogc:def:property:precipitation	250.8
21	2013-07-30 14:00:00 org:wegcenter:temperature_2m_Klimastation_10	org:wegcenter:WegenerNet_Klimastation_10	urn:ogc:def:property:temperature	27.45
22	2013-07-30 14:00:00 org:wegcenter:relative_humidity_2m_Klimastation_10	org:wegcenter:WegenerNet_Klimastation_10	urn:ogc:def:property:relative_humidity	39
23	2013-07-30 14:00:00 org:wegcenter:logger_supply_voltage_Klimastation_1	org:wegcenter:WegenerNet_Klimastation_1	urn:ogc:def:property:logger_supply_voltage	2.918
24	2013-07-30 14:00:00 org:wegcenter:logger_temperature_Klimastation_1	org:wegcenter:WegenerNet_Klimastation_1	urn:ogc:def:property:logger_temperature	32.7
25	2013-07-30 14:00:00 org:wegcenter:logger_humidity_Klimastation_1	org:wegcenter:WegenerNet_Klimastation_1	urn:ogc:def:property:logger_humidity	18.6
26	2013-07-30 14:00:00 org:wegcenter:precipitation_Friedr_Klimastation_1	org:wegcenter:WegenerNet_Klimastation_1	urn:ogc:def:property:precipitation	144.4

procedure_id PK character varying(100)	Procedure Tabelle	description_url character varying(200)	description_type character varying(100)
1297	org:wegcenter:soil_temperature_Klimastation_84	standard/soil_temperature_Klimastation_84.xml	text/xml;subtype="SensorML/1.0.1"
1298	org:wegcenter:soil_temperature_Klimastation_85	standard/soil_temperature_Klimastation_85.xml	text/xml;subtype="SensorML/1.0.1"
1299	org:wegcenter:soil_temperature_Klimastation_99	standard/soil_temperature_Klimastation_99.xml	text/xml;subtype="SensorML/1.0.1"
1300	org:wegcenter:temperature_2m_Klimastation_1	standard/temperature_2m_Klimastation_1.xml	text/xml;subtype="SensorML/1.0.1"
1301	org:wegcenter:temperature_2m_Klimastation_10	standard/temperature_2m_Klimastation_10.xml	text/xml;subtype="SensorML/1.0.1"
1302	org:wegcenter:temperature_2m_Klimastation_100	standard/temperature_2m_Klimastation_100.xml	text/xml;subtype="SensorML/1.0.1"
1303	org:wegcenter:temperature_2m_Klimastation_101	standard/temperature_2m_Klimastation_101.xml	text/xml;subtype="SensorML/1.0.1"

feature_of_interest_id PK character varying(100)	foi Tabelle	feature_of_interest_name character varying(100)	feature_of_interest_description character varying(200)	geom geometry	feature_type text	schema_link text
1	org:wegcenter:Klimaschuetzinger_Station_32	Klimaschuetzinger Station 32	Der Standort befindet sich	01010000207	sa:SamplingPoint	http://xyz.org/reference-ur12.html
2	org:wegcenter:Wasserversorgung_Grenz1_S0_82	Wasserversorgung Grenzland Sued	Die Station befindet sich	01010000207	sa:SamplingPoint	http://xyz.org/reference-ur12.html
3	org:wegcenter:WegenerNet_Klimastation_1	WegenerNet Klimastation 1	Der Standort befindet sich	01010000207	sa:SamplingPoint	http://xyz.org/reference-ur12.html
4	org:wegcenter:WegenerNet_Klimastation_10	WegenerNet Klimastation 10	Die Station befindet sich	01010000207	sa:SamplingPoint	http://xyz.org/reference-ur12.html
5	org:wegcenter:WegenerNet_Klimastation_100	WegenerNet Klimastation 100	Station auf einem Huelgel	01010000207	sa:SamplingPoint	http://xyz.org/reference-ur12.html
6	org:wegcenter:WegenerNet_Klimastation_101	WegenerNet Klimastation 101	Sende- und Speichereinheit	01010000207	sa:SamplingPoint	http://xyz.org/reference-ur12.html

phenomenon_id PK character varying(100)	phen. Tabelle	phenomenon_description character varying(200)	unit character varying(30)	valuetype character varying(40)	composite_phenomenon_application character varying(100)
11	urn:ogc:def:property:precipitation	precipitation	mm	numericType	
12	urn:ogc:def:property:relative_humidity	relative humidity	%	numericType	
13	urn:ogc:def:property:soil_moisture	soil moisture	m3/m3	numericType	
14	urn:ogc:def:property:soil_moisture_max	soil moisture max	m3/m3	numericType	
15	urn:ogc:def:property:soil_moisture_min	soil moisture min	m3/m3	numericType	
16	urn:ogc:def:property:soil temperature	soil temperature	*C	numericType	
17	urn:ogc:def:property:temperature	air temperature	*C	numericType	

Abbildung 41: Abhängigkeit der Observations von den Procedures, Feature-Of-Interest und Phenomena

4.2.2.2 WEGENERNET OFFERINGS

Die Gruppierungen der durch die Sensoren erzeugten *Observations* sind in Tabelle 6 dargestellt. Alle Sensoren zur Ermittlung der Lufttemperatur in zwei Meter Höhe über Grund sind z.B. dem Offering *temperature_2m* oder alle Prozeduren zur Ermittlung der GSM-Signalstärke dem Offering *GSM_signal_strength* zugeordnet. So können über nur einen Request alle *Observations* eines Offerings abgefragt werden.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

Tabelle 6: Offerings der WegenerNet SOS-Implementierung

Offering ID	Offering Beschreibung
air_pressure	The air pressure at the meteorological station
GSM_signal_strength	The GSM signal strength of the data logger at the meteorological station
logger_battery_voltage	The battery voltage of the data logger at the meteorological station
logger_humidity	The internal relative humidity of the cabinet with the data logger at the meteor. station
logger_storage_occupancy	The storage occupancy of the data logger at the meteorological station
logger_supply_voltage	The supply voltage of the data logger at the meteorological station
logger_temperature	The internal temperature of the cabinet with the data logger at the meteor. station
net_radiation	The net radiation at the meteorological station
peak_gust_10m	The peak gust at the meteorological station at 10 m height
peak_gust_14m	The peak gust at the meteorological station at 14 m height
peak_gust_18m	The peak gust at the meteorological station at 18 m height
peak_gust_55m	The peak gust at the meteorological station at 55 m height
peak_gust_direction_10m	The peak gust direction at the meteorological station at 10 m height
peak_gust_direction_14m	The peak gust direction at the meteorological station at 14 m height
peak_gust_direction_18m	The peak gust direction at the meteorological station at 18 m height
peak_gust_direction_55m	The peak gust direction at the meteorological station at 55 m height
pF_value	The pF value at the meteorological station
precipitation_Friedr	The precipitation at the meteorological station (sensor Friedrichs)
precipitation_Meteo	The precipitation at the meteorological station (sensor MeteoServis)
precipitation_Young	The precipitation at the meteorological station (sensor Young)
precipitation_Young_52m	The precipitation at the meteorological station at 52 m height (sensor Young)
relative_humidity_2m	The relative humidity at the meteorological station at 2 m height
relative_humidity_53m	The relative humidity at the meteorological station at 53 m height
soil_moisture	The soil moisture at the meteorological station
soil_moisture_max	The maximal moisture at the meteorological station
soil_moisture_min	The minimal soil moisture at the meteorological station
soil_temperature	The soil temperature at the meteorological station
temperature_2m	The air temperature at the at the meteorological station at 2 m height
temperature_53m	The air temperature at the at the meteorological station at 53 m height
wind_direction_10m	The wind direction at the meteorological station at 10 m height
wind_direction_14m	The wind direction at the meteorological station at 14 m height
wind_direction_18m	The wind direction at the meteorological station at 18 m height
wind_direction_55m	The wind direction at the meteorological station at 55 m height
wind_speed_10m	The wind speed at the meteorological station at 10 m height
wind_speed_14m	The wind speed at the meteorological station at 14 m height
wind_speed_18m	The wind speed at the meteorological station at 18 m height
wind_speed_55m	The wind speed at the meteorological station at 55 m height

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

In der dargestellten Konfiguration sind mehrere *Procedures* in einem *Offering* enthalten, was laut SOS 1.0.0 Spezifikation möglich ist. Aufgrund von dadurch erzeugten Mehrdeutigkeiten ist in der SOS 2.0 Spezifikation die Zuordnung zu einem *Offering* auf nur eine *Prozedur* limitiert. Wird die SOS 2.0 Schnittstelle abgefragt, so erzeugt der 52°North Server im Falle mehrere Prozeduren virtuelle *Offerings*.

4.2.2.3 APPLIKATION ZUM ERZEUGEN DER SENSORML-DATEIEN SOWIE ZUM ABFRAGEN DER OBSERVATIONS

Eine in C++ geschriebenen Applikation erzeugt automatisch Files mit Metadaten zur jeweiligen Prozedur im SensorML Format. Außerdem werden mittels der PostgreSQL C++ Bibliothek "libpq-fe.h" die *Observations* der WegenerNet Datenbank abgefragt, in eine SOS konforme Struktur gebracht und in die SOS Datenbank geschrieben. Derzeit werden nur die Level 0 Daten des WegenerNets konvertiert.

In Tabelle 7 ist die grobe Struktur des C++ Programms dargestellt, wobei die Realisierung auf folgende drei Threads unter Verwendung des *POSIX Threads API* basiert:

Thread 1 (writeSMLFile) – Nach dem Starten der Applikation wird der Nutzer aufgefordert, den Startzeitpunkt zum Lesen der Daten aus der WegenerNet Datenbank einzugeben. Anschließend werden zu den *Feature-Of-Interest* die ID, eine Beschreibung sowie die Koordinaten der Messstationen aus einer Datei ausgelesen und auf interne Variablen geschrieben. Ein weiteres File beinhaltet Informationen zu den *Procedures*, wie z.B. die ID, die Zugehörigkeit zum jeweiligen *Feature-Of-Interest*, die ermittelte Eigenschaft und den Dateinamen für die DescribeSensor Operation. Diese Informationen werden in eine SOS konforme Struktur gebracht und der Methode zum Erzeugen der SensorML Dateien als Eingangsparameter übergeben. Die Methode liest eine DescribeSensor-Template-Datei ein, ersetzt die Inhalte definierter XML-Tags mit den Eingangsparametern und erzeugt das File zur jeweiligen Prozedur.

Thread 2 (readWegenetDB) – Sind alle SensorML-Dateien erstellt worden, so liest der Driver die Daten der WegenerNet Datenbank vom übergebenen Startzeitpunkt bis zum aktuellen Datum aus, bringt die Daten in die standardisierte Struktur und schreibt sie in die SOS Datenbank. Der Driver prüft zusätzlich, ob bereits Daten innerhalb des gewünschten

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

Zeitraums in der SOS Datenbank gespeichert sind und fragt nur die nicht vorhandenen Daten aus der WegenerNet Datenbank ab.

Sind alle *Observations* bis zum aktuellen Zeitraum in der SOS Datenbank gespeichert, so wird die WegenerNet Datenbank in einem definierten Zeitintervall auf neue Daten überprüft. Im Falle neuer Datensätze werden diese wieder ausgelesen, in eine SOS konforme Struktur konvertiert und in die SOS Datenbank geschrieben.

Thread 3 (exitProg) – Die Applikation kann jederzeit mit der Escape-Taste beendet werden.

Tabelle 7: Strukturprogramm der WegenerNet-Applikation nach Nassi-Schneidermann

Thread 1	Thread 3 Programmabbruch mit Escape
Eingabe des Startzeitpunktes zum Lesen der Daten aus der WegenerNet Db	
Solange das Ende der Datei mit Informationen zu den <i>Feature-Of-Interests</i> nicht erreicht ist, Dateiinhalte auf interne Variablen schreiben	
Solange das Ende der Datei mit Informationen zu den <i>Procedures</i> nicht erreicht ist, Daten auf interne Variablen schreiben Daten in die richtige SOS-Struktur bringen Die Methode zum Erzeugen der SensorML Files aufrufen und die SensorML Dateien in den Sensorkonfigurationsordner des SOS-Servers speichern	
Thread 3	
Freigabe zum Lesen der Daten aus der WegenerNet DB	
Solange der Endzeitpunkt der gespeicherten Messwerte der WegenerNet DB nicht erreicht ist, Daten lesen und auf interne Variablen schreiben (Methode Fetch Table) Daten in die SOS konforme Struktur konvertieren Schreiben der Daten in die SOS-DB	
Solange die Applikation nicht abgebrochen wird, Die WegenerNet DB auf neue Messwerte überprüfen Wenn neue Messwerte vorhanden sind, Daten in die SOS konforme Struktur konvertieren und in die SOS-DB schreiben	

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

4.2.3 ERGEBNISSE

Dieses Kapitel beschreibt die erzielten Ergebnisse anhand ausgewählter Beispiele. Einerseits werden Ergebnisse der Kernoperationen dargestellt und andererseits der Zugriff mittels Webclient und SOS ArcGIS Extension illustriert. Prinzipiell können HTTP-GET und HTTP-POST-Anfragen an den 52°North SOS Server gesendet werden.

4.2.3.1 KERN-OPERATIONEN (CORE)

In den folgenden Beispielen wird die SOS Schnittstelle durch die Kernoperationen abgefragt. Die auf die SOS 1.0.0 sowie SOS 2.0 angewendete GetCapabilities Operation zeigt die Unterschiede der Service Antwort bezüglich der *Offerings*. Auf die GetObservation Operation wird zusätzlich ein zeitlicher Filter angewendet.

GetCapabilities:

Abbildung 42 zeigt eine nach der *OpenGIS Web Services (OWS)* Spezifikation strukturierte HTTP-POST-Anfrage um Metadaten zum SOS-Server zu erhalten. Im Tag *Sections* ist definiert zu welchen Bereichen der Server eine Antwort liefert.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope http://www.w3.org/2003/05/soap-envelope"
5   xmlns:sos="http://www.opengis.net/sos/2.0" xmlns:ows="http://www.opengis.net/ows/1.1">
6   <env:Body>
7     <sos:GetCapabilities xmlns:sos="http://www.opengis.net/sos/2.0"
8       xmlns:ows="http://www.opengis.net/ows/1.1"
9       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
10      xsi:schemaLocation="http://www.opengis.net/sos/2.0
11      http://schemas.opengis.net/sos/2.0/sosGetCapabilities.xsd"
12      service="SOS">
13       <ows:AcceptVersions>
14         <ows:Version>2.0.0</ows:Version>
15       </ows:AcceptVersions>
16       <ows:Sections>
17         <ows:Section>OperationsMetadata</ows:Section>
18         <ows:Section>ServiceIdentification</ows:Section>
19         <ows:Section>ServiceProvider</ows:Section>
20         <ows:Section>FilterCapabilities</ows:Section>
21         <ows:Section>Contents</ows:Section>
22       </ows:Sections>
23     </sos:GetCapabilities>
24   </env:Body>
25 </env:Envelope>
```

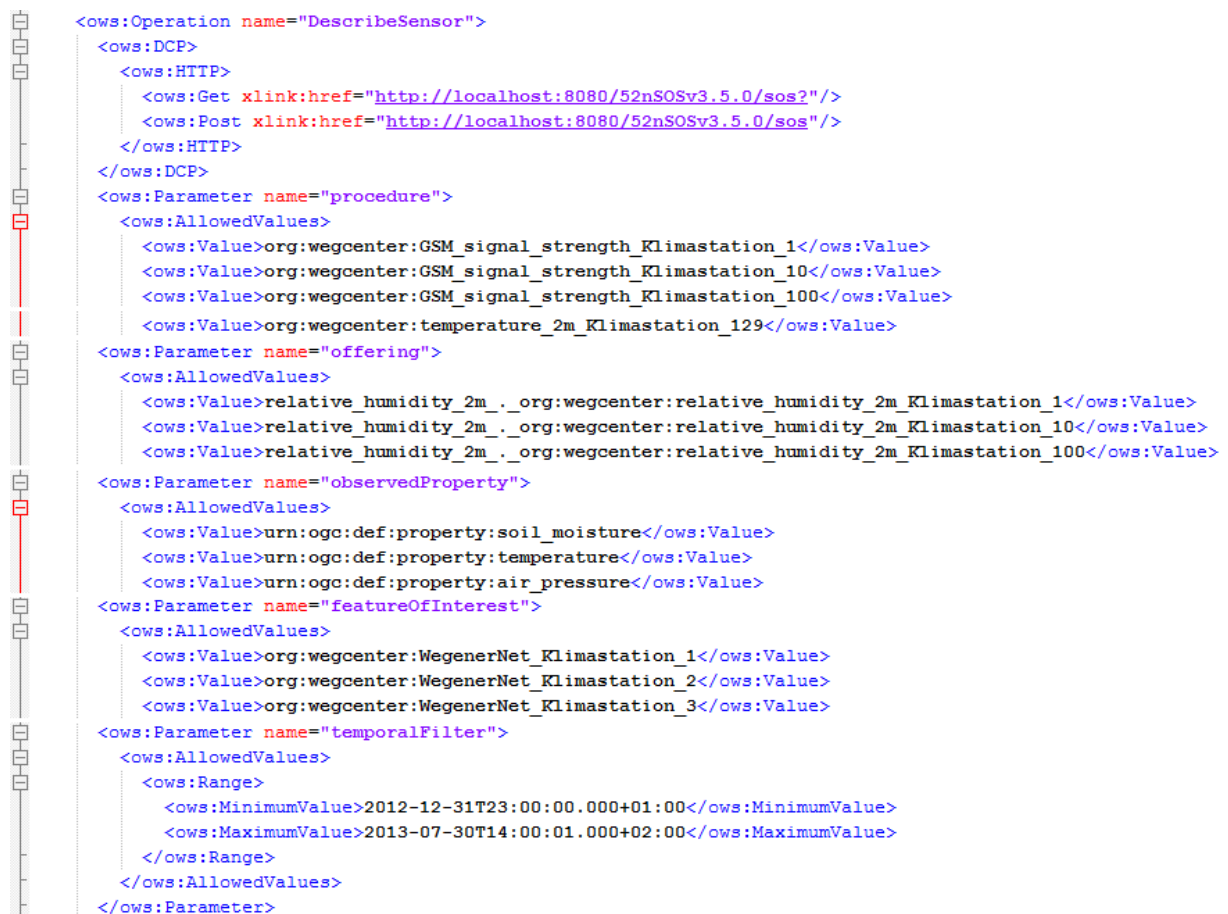
Abbildung 42: SOS 2.0 GetCapabilities Request

Die Antwort der *GetCapabilities* Operation besteht aus Informationen zum Service-Anbieter, zur unterstützten Service-Version, zum Namen des Anbieters, zur Telefonnummer, zu den

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

unterstützte Operationen sowie vorhandenen *Procedures*, *Feature-Of-Interest*, *Offerings*, *ObservedProperties* usw.

In Abbildung 43 sind Teile der Antwort der abgefragten SOS 2.0 Schnittstelle zu den *Procedures*, *Feature-Of-Interest*, *Offerings* und *ObservedProperties* dargestellt. Die Sektion *temporalFilter* beschreibt den Zeitraum, für den *Observations* vorhanden sind. Für diesen Zeitraum können innerhalb einer *GetObservation* oder einer *GetResult* Operation zeitliche Filter zur Abfrage der *Observations* und Ergebnisse angewendet werden. Da im dargestellten Beispiel die SOS 2.0 Schnittstelle abgefragt wurde und einem *Offering* mehrere *Procedures* zugeordnet sind, generiert der 52°North Server virtuelle *Offerings*, deren Identifier aus der eigentlichen *Offering* ID und der *Procedure* ID besteht.



```
<ows:Operation name="DescribeSensor">
  <ows:DCP>
    <ows:HTTP>
      <ows:Get xlink:href="http://localhost:8080/52nSOSv3.5.0/sos?"/>
      <ows:Post xlink:href="http://localhost:8080/52nSOSv3.5.0/sos"/>
    </ows:HTTP>
  </ows:DCP>
  <ows:Parameter name="procedure">
    <ows:AllowedValues>
      <ows:Value>org:wegcenter:GSM_signal_strength_Klimastation_1</ows:Value>
      <ows:Value>org:wegcenter:GSM_signal_strength_Klimastation_10</ows:Value>
      <ows:Value>org:wegcenter:GSM_signal_strength_Klimastation_100</ows:Value>
      <ows:Value>org:wegcenter:temperature_2m_Klimastation_129</ows:Value>
    </ows:AllowedValues>
  </ows:Parameter name="offering">
    <ows:AllowedValues>
      <ows:Value>relative_humidity_2m_._org:wegcenter:relative_humidity_2m_Klimastation_1</ows:Value>
      <ows:Value>relative_humidity_2m_._org:wegcenter:relative_humidity_2m_Klimastation_10</ows:Value>
      <ows:Value>relative_humidity_2m_._org:wegcenter:relative_humidity_2m_Klimastation_100</ows:Value>
    </ows:AllowedValues>
  <ows:Parameter name="observedProperty">
    <ows:AllowedValues>
      <ows:Value>urn:ogc:def:property:soil_moisture</ows:Value>
      <ows:Value>urn:ogc:def:property:temperature</ows:Value>
      <ows:Value>urn:ogc:def:property:air_pressure</ows:Value>
    </ows:AllowedValues>
  <ows:Parameter name="featureOfInterest">
    <ows:AllowedValues>
      <ows:Value>org:wegcenter:WegenerNet_Klimastation_1</ows:Value>
      <ows:Value>org:wegcenter:WegenerNet_Klimastation_2</ows:Value>
      <ows:Value>org:wegcenter:WegenerNet_Klimastation_3</ows:Value>
    </ows:AllowedValues>
  <ows:Parameter name="temporalFilter">
    <ows:AllowedValues>
      <ows:Range>
        <ows:MinimumValue>2012-12-31T23:00:00.000+01:00</ows:MinimumValue>
        <ows:MaximumValue>2013-07-30T14:00:01.000+02:00</ows:MaximumValue>
      </ows:Range>
    </ows:AllowedValues>
  </ows:Parameter>
</ows:Operation>
```

Abbildung 43: Ausschnitte des GetCapabilities Response

Im Vergleich zu den virtuellen *Offerings*, generiert von der SOS 2.0 Schnittstelle, zeigt Abbildung 44 alle *Offerings* der SOS 1.0.0 Schnittstelle, die erlaubt, dass mehrere *Procedures*

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

in einem *Offering* enthalten sind. Die in Tabelle 6 dargestellte Konfiguration der *Offerings* bleibt also unter Verwendung der SOS 1.0.0 Spezifikation unverändert.

```
<ows:Parameter name="offering">
  <ows:AllowedValues>
    <ows:Value>relative_humidity_2m</ows:Value>
    <ows:Value>peak_gust_18m</ows:Value>
    <ows:Value>GSM_signal_strength</ows:Value>
    <ows:Value>soil_moisture_max</ows:Value>
    <ows:Value>relative_humidity_53m</ows:Value>
    <ows:Value>soil_moisture_min</ows:Value>
    <ows:Value>logger_battery_voltage</ows:Value>
    <ows:Value>precipitation_Young</ows:Value>
    <ows:Value>logger_supply_voltage</ows:Value>
    <ows:Value>wind_direction_18m</ows:Value>
    <ows:Value>logger_humidity</ows:Value>
    <ows:Value>peak_gust_direction_55m</ows:Value>
    <ows:Value>wind_speed_14m</ows:Value>
    <ows:Value>peak_gust_direction_14m</ows:Value>
    <ows:Value>peak_gust_55m</ows:Value>
    <ows:Value>wind_speed_18m</ows:Value>
    <ows:Value>pF_value</ows:Value>
    <ows:Value>precipitation_Meteo</ows:Value>
    <ows:Value>peak_gust_direction_10m</ows:Value>
    <ows:Value>precipitation_Friedr</ows:Value>
    <ows:Value>net_radiation</ows:Value>
    <ows:Value>precipitation_Young_52m</ows:Value>
    <ows:Value>soil_temperature</ows:Value>
    <ows:Value>wind_direction_55m</ows:Value>
    <ows:Value>temperature_2m</ows:Value>
    <ows:Value>soil_moisture</ows:Value>
    <ows:Value>peak_gust_14m</ows:Value>
    <ows:Value>peak_gust_direction_18m</ows:Value>
    <ows:Value>wind_direction_10m</ows:Value>
    <ows:Value>wind_speed_55m</ows:Value>
    <ows:Value>wind_direction_14m</ows:Value>
    <ows:Value>air_pressure</ows:Value>
    <ows:Value>logger_storage_occupancy</ows:Value>
    <ows:Value>logger_temperature</ows:Value>
    <ows:Value>wind_speed_10m</ows:Value>
    <ows:Value>peak_gust_10m</ows:Value>
    <ows:Value>temperature_53m</ows:Value>
  </ows:AllowedValues>
```

Abbildung 44: Offering Bereich der abgefragten SOS 1.0.0 Schnittstelle

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

DescribeSensor:

Die DescribeSensor Operation gibt die durch die C++-Applikation erzeugte SensorML-Datei mit Metadaten zum abgefragten Sensor bzw. zur abgefragten *Procedure* zurück. Der Abfrage werden als Parameter das Format der Ausgabedatei, die SensorID, das verwendete Service, sowie die Version des SOS übergeben.

Abbildung 45 zeigt ein Beispiel einer DescribeSensor Request Operation zur Abfrage der Metadaten des Sensors der ID `org:wegcenter:temperature_2m_Klimastation_8`. Im oberen Teil der Abbildung ist der Datenbankeintrag der Tabelle *procedure* mit der ID, der URL und des Typs der Rückgabe dargestellt. Bei Abfrage der Metadaten des Sensors `org:wegcenter:temperature_2m_Klimastation_8` durch die im unteren Teil der Abbildung dargestellten Request Operation werden die Inhalte der Datei „temperature_2m_Klimastation_8.xml“ zurückgegeben.

	procedure_id [PK] character varying(100)	description_url character varying(200)	description_type character varying(100)
1427	org:wegcenter:temperature_2m_Klimastation_79	standard/temperature_2m_Klimastation_79.xml	text/xml;subtype="SensorML/1.0.1"
1428	org:wegcenter:temperature_2m_Klimastation_8	standard/temperature_2m_Klimastation_8.xml	text/xml;subtype="SensorML/1.0.1"


```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope
5     http://www.w3.org/2003/05/soap-envelope"> <env:Body>
6
7   <swes:DescribeSensor service="SOS" version="2.0.0"
8     xmlns:swes="http://www.opengis.net/swes/2.0"
9     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
10    xsi:schemaLocation="http://www.opengis.net/swes/2.0
11      http://schemas.opengis.net/swes/2.0/swes.xsd">
12     <swes:procedure>org:wegcenter:temperature_2m_Klimastation_8</swes:procedure>
13
14     <swes:procedureDescriptionFormat>http://www.opengis.net/sensorML/1.0.1
15     </swes:procedureDescriptionFormat>
16   </swes:DescribeSensor>
17 </env:Body>
18 </env:Envelope>
```

Abbildung 45: SOS 2.0 DescribeSensor Request Operation des Sensors `org:wegcenter:temperature_2m_Klimastation_8`

Abbildung 46 zeigt Teile der Antwort der DescribeSensor Operation im SensorML Format. Durch die kodierten sensorspezifischen Informationen kann der Benutzer Charakteristiken zum Sensor und benötigte Informationen wie die Sensorposition, den EPSG-Code, die beobachtbare Eigenschaft (*observableProperty*), etc. abrufen. Der erste Teil der Abbildung bietet Informationen zum Rechtswert und Hochwert der Sensorposition, inklusive EPSG Code (EPSG 32633 = WGS 84 / UTM zone 33N). Die Sektion *InputList* listet die durch die *Procedure* `org:wegcenter:temperature_2m_Klimastation_8` beobachtbare Eigenschaft `urn:ogc:def:property:temperature` auf. Die *OutputList* zeigt, dass die Temperatur als

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

Quantität der Einheit °C vorliegt. Im Bereich *validTime* wird die Zeitperiode mit vorhandenen *Observations* angegeben.

Die in der Abbildung nicht dargestellte *Capabilities* Sektion beinhaltet einen Link zum *Feature-Of-Interest* `org:wegcenter:WegenerNet_Klimastation_8`. Diese Klimastation ist unter anderem mit der *Procedure* `org:wegcenter:temperature_2m_Klimastation_8` zur Messung der Lufttemperatur in zwei Metern über Grund ausgestattet.

```
<sml:position name="stationPosition">
  <swe:Position fixed="true" referenceFrame="urn:ogc:def:crs:EPSG::32633">
    <swe:location>
      <swe:Vector gml:id="SYSTEM_LOCATION">
        <swe:coordinate name="easting">
          <swe:Quantity axisID="y">
            <swe:uom code="degree"/>
            <swe:value>567384.02</swe:value>
          </swe:Quantity>
        </swe:coordinate>
        <swe:coordinate name="northing">
          <swe:Quantity axisID="x">
            <swe:uom code="degree"/>
            <swe:value>5205115.67</swe:value>
          </swe:Quantity>
        </swe:coordinate>
      </swe:Vector>
    </swe:location>
  </swe:Position>
</sml:position>
<sml:inputs>
  <sml:InputList>
    <sml:input name="urn:ogc:def:property:temperature">
      <swe:ObservableProperty definition="urn:ogc:def:property:temperature"/>
    </sml:input>
  </sml:InputList>
</sml:inputs>
<sml:outputs>
  <sml:OutputList>
    <sml:output name="urn:ogc:def:property:temperature">
      <swe:Quantity definition="urn:ogc:def:property:temperature">
        <swe:uom code="°C"/>
      </swe:Quantity>
    </sml:output>
  </sml:OutputList>
</sml:outputs>

<sml:validTime>
  <gml:TimePeriod>
    <gml:beginPosition>2012-12-31T23:00:00.000+01:00</gml:beginPosition>
    <gml:endPosition>2013-07-30T13:10:00.000+02:00</gml:endPosition>
  </gml:TimePeriod>
</sml:validTime>
```

Abbildung 46: Teil der SOS 2.0 DescribeSensor Antwort im SensoML Format

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

GetObservation:

Mit der GetObservation Operation aus Abbildung 47 werden *Observations* des Sensors der ID `org:wegcenter:temperature_2m_Klimastation_8` abgefragt. Zusätzlich wird die SOS Antwort durch einen zeitlichen Filter eingeschränkt, der verursacht, dass nur *Observations* vom 7. Juli 2013 zwischen 17:44:15 und 18:00:15 Uhr zurückgegeben werden.

Neben dem zeitlichen Filter besteht die Möglichkeit der Anwendung räumlicher Filter.

```
15
16 <sos:procedure>org:wegcenter:temperature_2m_Klimastation_8</sos:procedure>
17
18 <sos:temporalFilter>
19   <fes:During>
20     <fes:ValueReference>phenomenonTime</fes:ValueReference>
21     <gml:TimePeriod gml:id="tp_1">
22       <gml:beginPosition>2013-07-01T17:44:15.000+02:00</gml:beginPosition>
23       <gml:endPosition>2013-07-01T18:00:15.000+02:00</gml:endPosition>
24     </gml:TimePeriod>
25   </fes:During>
26 </sos:temporalFilter>
27
```

Abbildung 47: Teil des SOS 2.0 Requests zur Abfrage von Observations des Procedures `org:wegcenter:temperature_2m_Klimastation_8` mit zeitlichem Filter

Teile der SOS 2.0 GetObservation Response sind in Abbildung 48 dargestellt. Die Antwort ist nach dem O&M Standard kodiert und besteht für das Beispiel aus vier zurückgegebenen Werten (siehe Abbildung 48, Sektion *elementCount*) mit Zeitstempel und anderen Informationen zur *Procedure*. Die Sektion *encoding* definiert das Rückgabeformat der Werte. Im Bereich *values* sind die Ergebnisse mit Zeitstempel und zugehörigem Ergebnis aufgelistet.

Um nur das Ergebnis als Antwort zu erhalten kann beispielsweise für Systeme mit eingeschränkter Bandbreite die *getResult* Operation angewendet werden.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

```
27 <om:result xmlns:swe="http://www.opengis.net/swe/2.0" xsi:type="swe:DataArrayPropertyType">
28   <swe:DataArray>
29     <swe:elementCount>
30       <swe:Count>
31         <swe:value>4</swe:value>
32       </swe:Count>
33     </swe:elementCount>
34     <swe:elementType name="Components">
35       <swe>DataRecord>
36         <swe:field name="phenomenonTime">
37           <swe:Time definition="http://www.opengis.net/def/property/OGC/0/PhenomenonTime">
38             <swe:uom xlink:href="http://www.opengis.net/def/uom/ISO-8601/0/Gregorian"/>
39           </swe:Time>
40         </swe:field>
41         <swe:field name="temperature">
42           <swe:Quantity definition="urn:ogc:def:property:temperature">
43             <swe:uom code="°C"/>
44           </swe:Quantity>
45         </swe:field>
46       </swe>DataRecord>
47     </swe:elementType>
48     <swe:encoding>
49       <swe:TextEncoding blockSeparator="," decimalSeparator="." tokenSeparator=","/>
50     </swe:encoding>
51     <swe:values>2013-07-01T17:45:00.000+02:00,23.38;2013-07-01T17:50:00.000+02:00,23.25;
52     2013-07-01T17:55:00.000+02:00,23.21;2013-07-01T18:00:00.000+02:00,23.55;</swe:values>
53   </swe:DataArray>
54 </om:result>
```

Abbildung 48: Teil der SOS 2.0 GetObservation Antwort

4.2.3.2 ZUGRIFF MITTELS CLIENTS

Folgende Beispiele illustrieren den Zugriff auf das SOS, einerseits mittels einer webbasierten Client-Applikation zur Darstellung von Graphen und Erstellung von Tabellen und andererseits durch eine ArcGIS Extension zur Integration der Daten ins ArcGIS.

52°North SensorWebClient:

Der Thin Client des 52°North bietet die Möglichkeit eines Zugriffs auf Zeitreihendaten, die innerhalb eines SOS gespeichert sind. Die Daten können von einer oder mehreren SOS Instanzen der Version 1.0.0 und 2.0 abgefragt und miteinander kombiniert werden, ohne im Detail über die Datenmodellierung und der SOS Struktur Bescheid wissen zu müssen. Ist ein SES oder WNS am Server implementiert, so können benutzerdefinierte Regeln für das automatische Erhalten von Benachrichtigungen über die Benutzeroberfläche des Clients definiert werden. Des Weiteren unterstützt der Client, wie in Kapitel 3.4 erklärt, verschiedene Profile zur Erhöhung der Interoperabilität (Bredel, 2013).

Im Auswahlfenster „Datenquelle wählen“ der Abbildung 49 sind alle im Client konfigurierten SOS Instanzen aufgelistet. Wird die gewünschte Quelle gewählt, so

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

erscheinen zu dieser Instanz die vorhandenen Phänomene der *Feature-Of-Interest*. In einer Übersichtskarte sind die Positionen der *Feature-Of-Interest* dargestellt.

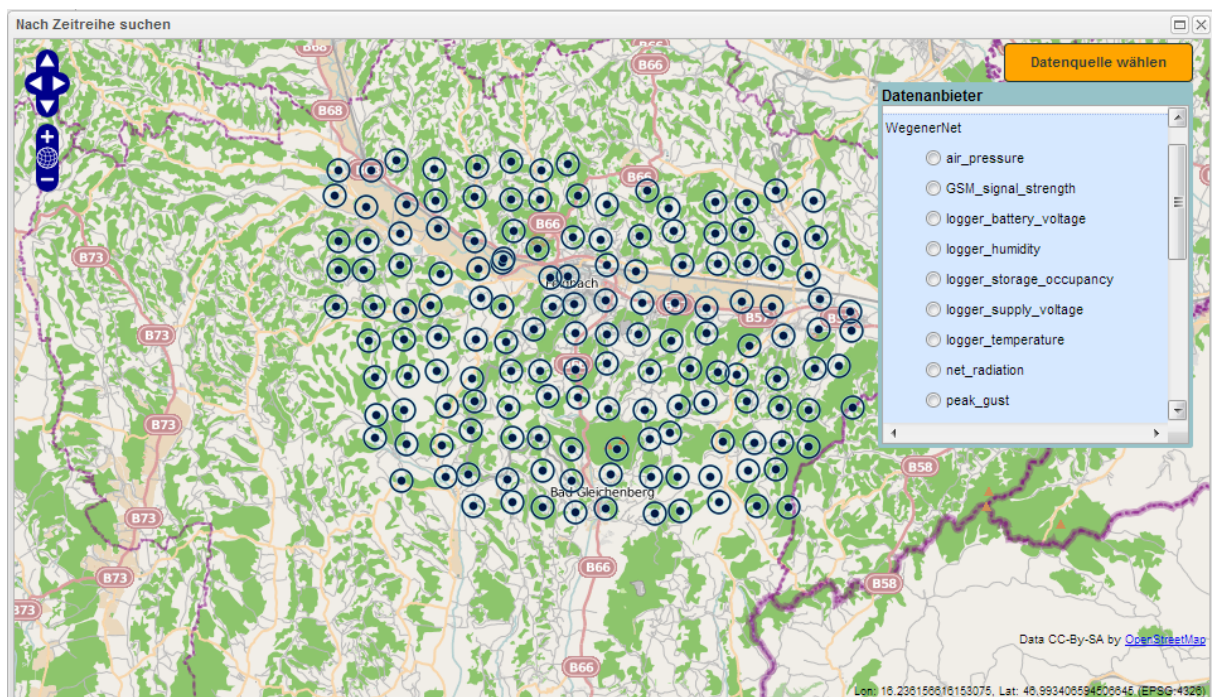


Abbildung 49: WegenerNet-Stationsübersicht des 52°North SensorWebClient

Wird ein Phänomen im Fenster „Datenquelle wählen“ markiert, werden alle *Feature-Of-Interest*, die über *Procedures* Messwerte zur gewählten Eigenschaft liefern, dargestellt. Abbildung 50 zeigt alle Stationen mit der Eigenschaft Windgeschwindigkeit, wobei die rot markierte Station die Referenzstation mit den meisten Sensoren darstellt. Durch Anwählen eines *Feature-Of-Interest* können Daten zur Zeitreihe hinzugefügt werden.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

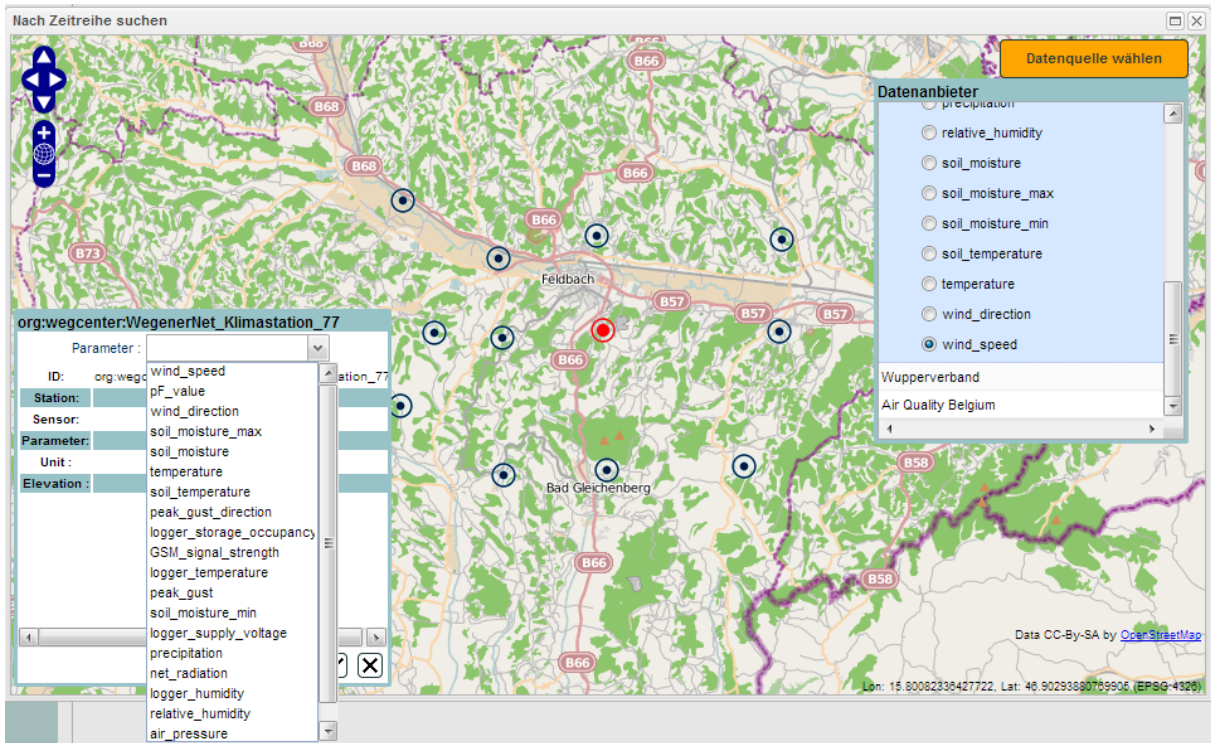


Abbildung 50: WegenerNet Stationen welche Messwerte zur Eigenschaft Windgeschwindigkeit liefern

Abbildung 51 zeigt Zeitreihen der Temperatur sowie eine Zeitreihe mit Windgeschwindigkeitsmesswerten (blaue Trendlinie) ausgewählter Messstationen (*Feature-Of-Interest*).

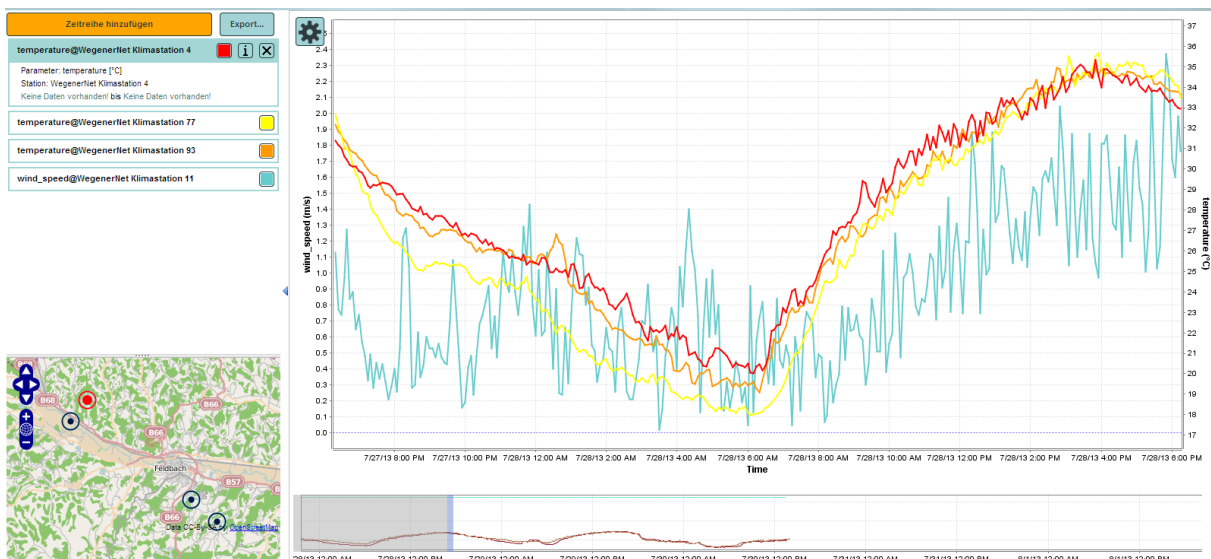


Abbildung 51: Darstellung der Zeitreihen ausgewählter WegenerNet Stationen mit dem 52°North SensorWebclient

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

52°North ArcGIS SOS Extension:

Durch die ArcGIS SOS Extension können Daten in Echtzeit sowie historische Zeitreihen dem ArcGIS hinzugefügt werden. Werden die Daten vom Server z.B. in eine File-Geodatabase heruntergeladen, kann der volle Analyse- und Darstellungsumfang des ArcGIS auf diese angewendet werden.

Im rechten Teil der Abbildung 52 wird mittels ArcGIS SOS Extension auf die Daten der WegenerNet SOS Datenbank zugegriffen. Die Daten können nach *Offerings*, *Properties* oder *Procedures* geordnet werden und für einen gewünschten Zeitraum in eine File-Geodatabase gespeichert werden. Die roten Punkte kennzeichnen die mit der ID beschrifteten *Feature-Of-Interest* des WegenerNets.

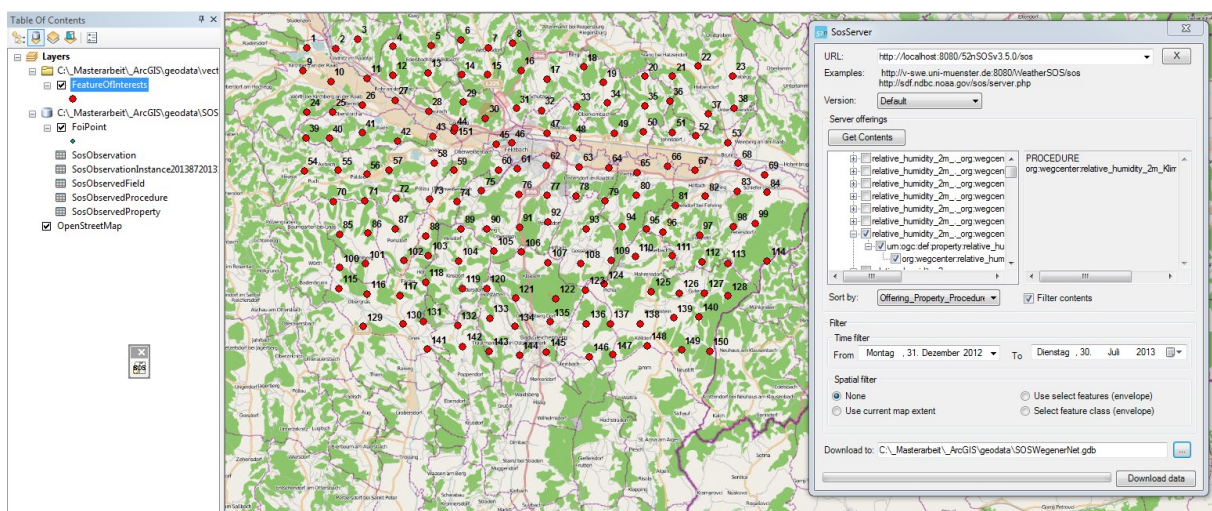


Abbildung 52: Zugriff auf die WegenerNet SOS Daten mittels ArcGIS SOS Extension (Kartengrundlage: OpenStreetMap)

4.3 ZAMG SOS

4.3.1 ALLGEMEINES

Die Wetterstationen der ZAMG bilden ein umfassendes meteorologisches Messnetz in Österreich, dessen über Sensoren produzierte Messwerte als Eingangsparameter für diverse Wetter- oder Klimamodelle verwendet werden. Die Stationen sind so positioniert, dass sie möglichst gleichmäßig alle Höhenstufen, sowie Klimaregionen abdecken. Die grundlegenden Wetterelemente werden hauptsächlich über sogenannte teilautomatische Wetterstationen (TAWES) aufgezeichnet und in Echtzeit an die Hohe Warte in Wien übertragen, wo eine

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

Qualitätskontrolle sowie Archivierung der Daten stattfindet. Zusätzlich erfolgt ein Datenaustausch mit österreichischen Messstationen anderer Anbieter (ZAMG, 2013).

Die gelben TAWES-Stationen in Abbildung 53 sind als SYNOP-Stationen klassifiziert. Bei den orangen Minitakles-Stationen handelt es sich um das Vorgängermodell der TAWES-Stationen, rote Kreise kennzeichnen Ö3-Stationen und blaue Kreise sind Stationen der *Austro Control* (ACG).

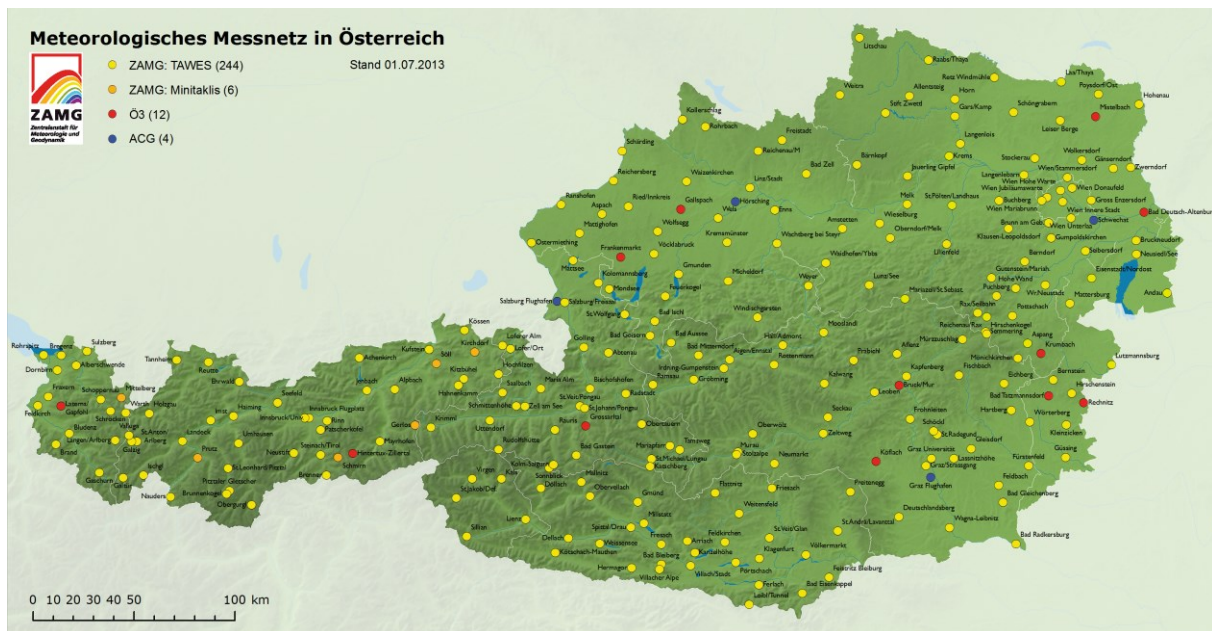


Abbildung 53: Meteorologische Messstationen der ZAMG, des ACG und Ö3 (ZAMG, 2013)

4.3.2 IMPLEMENTIERUNG IN DAS SOS DATENMODELL

Im Gegensatz zur WegenerNet Implementierung erfolgt der Datenzugriff nicht auf Echtzeitdaten, sondern auf aufbereitete Offline-Testdaten der Stationen des Jahres 2008.

4.3.2.1 DATENGRUNDLAGE

Die Datengrundlage besteht aus 265 Dateien, mit von der Universität Graz aufbereiteten und korrigierten meteorologischen Messwerten. Jedes Datenfile beinhaltet als Header den Stationsnamen, die Seehöhe und die Koordinaten des Aufstellungsortes. Die Messwerte der Parameter sind in einer zeitlichen Auflösung von einer Stunde in den Dateien gespeichert, und die Koordinaten der Messstationen liegen in MGI- (Militärgeographisches Institut) Lambert-konformer Kegelprojektion vor.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

Tabelle 8 zeigt einen Auszug der aufbereiteten meteorologischen Daten der sich in St. Sebastian am Flugfeld auf einer Seehöhe von 862 Metern befindenden Messstation. Neben dem Header beinhaltet jedes File einen Zeitstempel (Monat, Tag, Jahr, Stunde) und Messwerte zu folgenden meteorologischen Parametern:

1. **sIn** – Kurzwellige Einstrahlung in Watt pro Quadratmeter
2. **sOut** – Kurzwellige Ausstrahlung in Watt pro Quadratmeter
3. **lIn** – Langwellige Einstrahlung in Watt pro Quadratmeter
4. **lOut** – Langwellige Ausstrahlung in Watt pro Quadratmeter
5. **hum** – Relative Luftfeuchtigkeit in Prozent
6. **temp** – Lufttemperatur in zwei Metern über Grund in Kelvin
7. **speed** – Windgeschwindigkeit in Kilometer pro Stunde
8. **dir** – Windrichtung in Grad
9. **precip** – Niederschlag in Millimeter
10. **phase** – Phasenzustand des Niederschlags, fest oder flüssig

Alle mit NaN (not a number) gekennzeichneten Einträge stehen für nicht erfasste Daten. Die Parameter sOut, lIn, lOut, dir und phase sind in den Dateien nur vorbereitet, die Felder enthalten jedoch keine Einträge.

Tabelle 8: Auszug der meteorologischen Daten der Station Mariazell des Jahres 2008 (Datengrundlage: ZAMG)

Station		MARIAZELL/ST.SEBASTIAN-FLUGFELD												
Hoehe		862												
RW		547505												
HW		434067												
month	day	year	hour	sIn	sOut	lIn	lOut	hum	temp	speed	dir	precip	phase	
1	1	2008	0	0	NaN	NaN	NaN	93	270.25	1.9	NaN	0.2	NaN	
1	1	2008	1	0	NaN	NaN	NaN	95	270.05	3	NaN	0.7	NaN	
1	1	2008	2	0	NaN	NaN	NaN	90	269.95	2.1	NaN	0	NaN	
1	1	2008	3	0	NaN	NaN	NaN	93	269.75	1	NaN	0	NaN	
1	1	2008	4	0	NaN	NaN	NaN	99	269.65	1.5	NaN	0.8	NaN	
...	
12	31	2008	17	0	NaN	NaN	NaN	64	271.05	0.5	NaN	0	NaN	
12	31	2008	18	0	NaN	NaN	NaN	67	271.15	2.1	NaN	0	NaN	
12	31	2008	19	0	NaN	NaN	NaN	78	268.65	1.2	NaN	0	NaN	
12	31	2008	20	0	NaN	NaN	NaN	90	267.15	0.4	NaN	0	NaN	
12	31	2008	21	0	NaN	NaN	NaN	96	266.05	0.8	NaN	0	NaN	
12	31	2008	22	0	NaN	NaN	NaN	95	265.85	0.4	NaN	0	NaN	
12	31	2008	23	0	NaN	NaN	NaN	96	265.25	0.5	NaN	0	NaN	

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

4.3.2.2 APPLIKATION ZUR ERZEUGUNG DER SENSORML-DATEIEN UND ERSTELLUNG DES SOS MODELLS

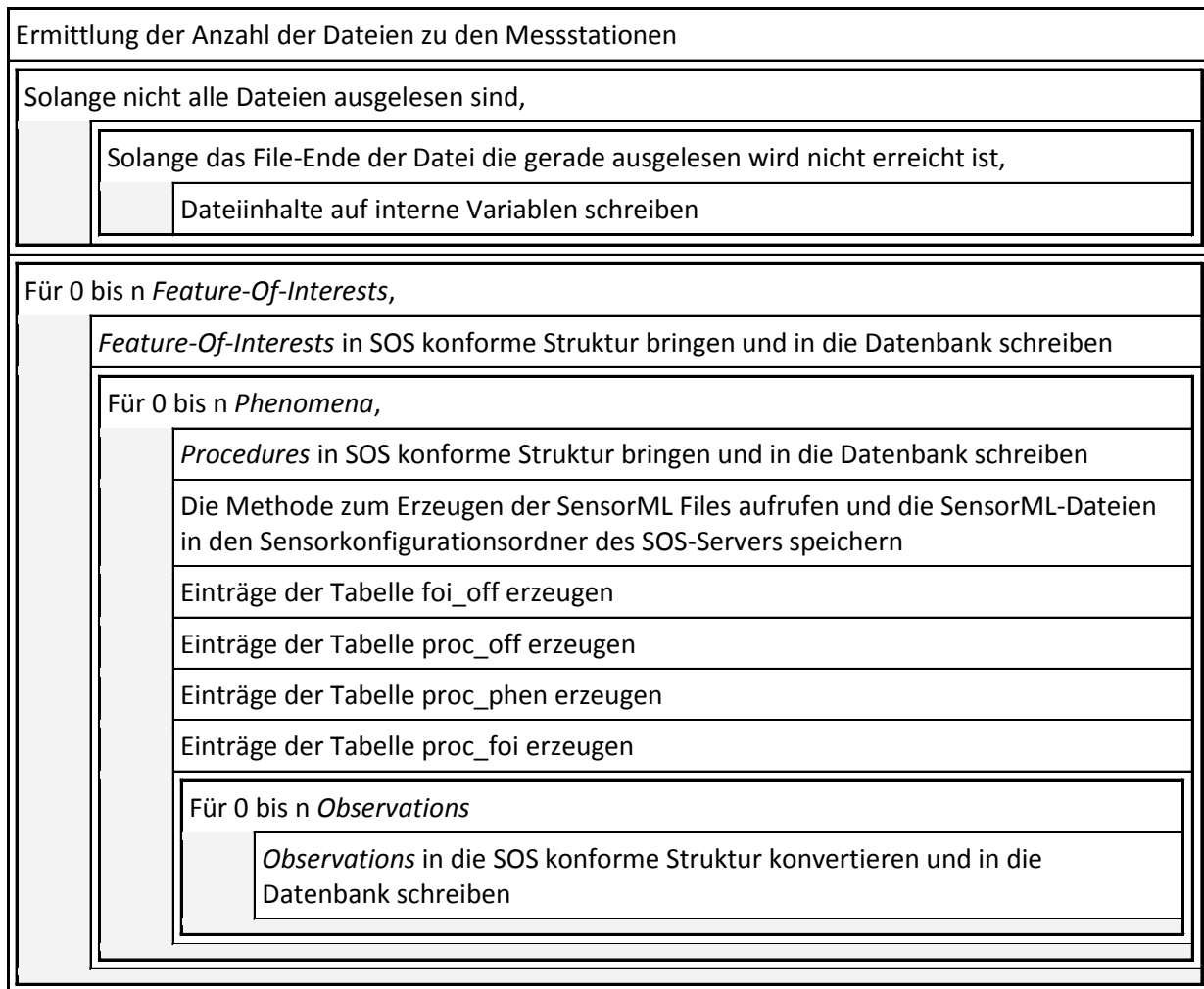
Eine weitere in C++ entwickelte Applikation liest die Dateiinhalte der Files (siehe Tabelle 8) zu den Messstationen aus und erzeugt automatisch ein SOS konformes Modell für den standardisierten Zugriff auf die Messwerte der Parameter der ZAMG Stationen. Für den Zugriff auf die Datenbank wird wie im Falle der WegenerNet Implementierung (siehe Kapitel 4.2) die PostgreSQL C++ Bibliothek "libpq-fe.h" verwendet.

Der Ablauf des C++-Programms ist als Strukturprogramm in Tabelle 9 dargestellt. Nach dem Starten der Applikation ermittelt diese die Anzahl der zu lesenden Dateien und schreibt die Dateiinhalte auf internen Variablen. Da pro Messstation ein File vorliegt entspricht die Anzahl der Dateien jener der *Feature-Of-Interest*. Die *Feature-Of-Interest* werden in ein SOS konformes Format konvertiert und in die PostgreSQL Tabelle geschrieben. Für jedes *Feature-Of-Interest* werden die in Kapitel 4.3.2.1 zehn beschriebenen *Phenomena* durchlaufen. Die *Procedures* werden anschließend in der richtigen Struktur in die Datenbank geschrieben. Außerdem werden die Einträge der Tabellen *foi_off*, *proc_off*, *proc_phen*, *proc_foi*, welche die *many-to-many* Beziehungen umsetzen, erstellt. Für die anschließende Erzeugung der SensorML-Dateien wird die für die WegenerNet Implementierung bereits erstellte C++-Methode verwendet. Sofern keine NaN-Werte vorliegen, sind für das Jahr 2008 aufgrund der zeitlichen Auflösung von einer Stunde pro *Phenomenon* 8.784 durch *Observations* erzeugte Messwerte in den Files gespeichert. Die *Observations* werden durchlaufen, in eine SOS konforme Struktur gebracht und in die Datenbank geschrieben.

Die Einträge der Phänomene, der *Offerings* und der *many-to-many* Beziehungen zwischen den Phänomenen und *Offerings* wurden vorab manuell erstellt.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

Tabelle 9: Strukturprogramm der ZAMG-Applikation nach Nassi-Schneidermann



4.3.2.3 ZAMG OFFERINGS

Die Gruppierung der in Tabelle 10 dargestellten *Observations* ist wie im Falle der WegenerNet Implementierung auf Basis der Phänomene realisiert. Da auch hier mehrere *Procedures* in einem *Offering* enthalten sind, werden bei der Abfrage der SOS 2.0 Schnittstelle virtuelle *Offerings* erzeugt (siehe Kapitel 4.2.3.1).

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

Tabelle 10: Offerings der ZAMG SOS-Implementierung

Offering ID	Offering Beschreibung
in_shortwave_radiation	The incoming shortwave radiation at the meteorological station
out_shortwave_radiation	The outgoing shortwave radiation at the meteorological station
in_longwave_radiation	The incoming longwave radiation at the meteorological station
out_longwave_radiation	The outgoing longwave radiation at the meteorological station
relative_humidity	The relative humidity at the meteorological station
temperature	The air temperature at the meteorological station
wind_speed	The wind speed at the meteorological station
wind_direction	The wind direction at the meteorological station
precipitation	The precipitation at the meteorological station
phase	The phase at the meteorological station

4.3.3 ERGEBNISSE

Die Ergebnisse der ZAMG Implementierung werden anhand ausgewählter Operationen und durch Zugriff mittels 52°North SensorWebClient illustriert.

4.3.3.1 SOS OPERATIONEN

Durch die folgenden dargestellten Operationen `GetFeatureOfInterest` und `GetObservation` wird die SOS 2.0 Schnittstelle des ZAMG SOS unter zusätzlicher Verwendung von Filter Operationen abgefragt. Für die `GetFeatureOfInterest` Operation wird ein räumlicher, für die `GetObservation` Operation ein räumlicher und zeitlicher Filter angewendet.

GetFeatureOfInterest:

Die `GetFeatureOfInterest` Operation ist der *Enhanced Operation Extension* (siehe Abbildung 8) zugeordnet und gibt *Feature-Of-Interest* anhand bestimmter Abfragekriterien zurück.

Abbildung 54 illustriert die Verwendung des räumlichen Filters *Bounding Box* (BBOX) innerhalb der `GetFeatureOfInterest` Operation. Durch Parameterübergabe der Koordinaten des Hochwertes (HW) und Rechtswertes (RW) der BBOX und des zugehörigen EPSG-Codes werden alle *Feature-Of-Interest* innerhalb dieser Box zurückgegeben.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

Parameterübergabe für die Realisierung des räumlichen Filters BBOX innerhalb der Sektion *spatialFilter* (siehe Abbildung 55):

Parameter: Koordinaten der linken unteren Ecke der Box; HW=386884, RW=534340.
Koordinaten der rechten oberen Ecke der Box; HW=439113, RW=557676.
EPSG-Code; 31287 für MGI-Austria-Lambert

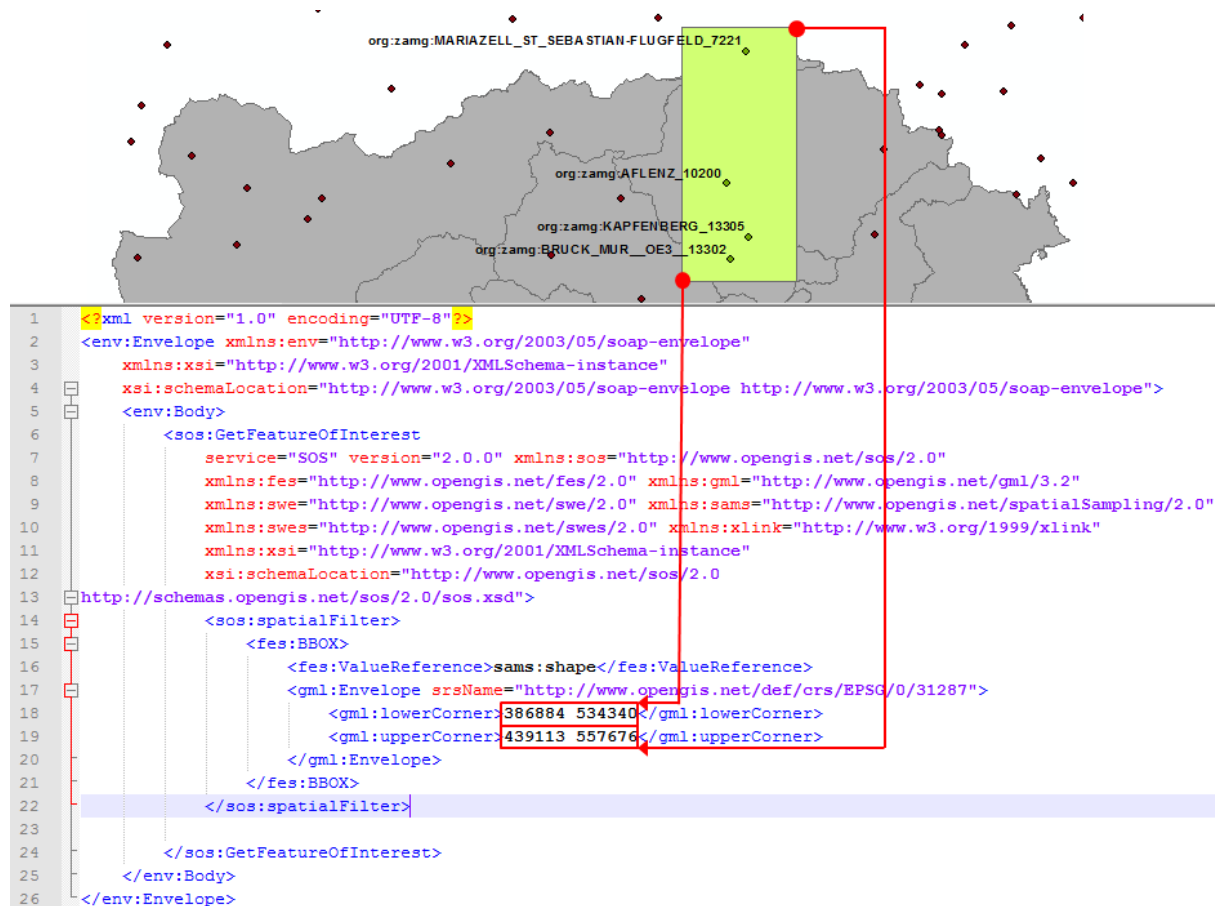


Abbildung 54: SOS 2.0 GetFeatureOfInterest Request mit räumlichen Filter (BBOX) (Datengrundlage: GIS Steiermark, ZAMG)

Folgende *Feature-Of-Interest* liegen innerhalb der BBOX (siehe Abbildung 55):

- org:zamg:BRUCK_MUR_OE3_13302
- org:zamg:MARIAZELL_ST_SEBASTIAN-FLUGFELD_7221
- org:zamg:AFLENZ_10200
- org:zamg:KAPFENBERG_13305

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

```
4 <sams:SF_SpatialSamplingFeature gml:id="sf_1">
5   <gml:identifier codeSpace="">org:zamg:BRUCK_MUR_OE3_13302</gml:identifier>
6   <sf:type xlink:href="http://www.opengis.net/def/samplingFeatureType/OGC-OM/2.0/SF_SamplingPoint"/>
7   <sf:sampledFeature xlink:href="urn:ogc:def:nil:OGC:unknown"/>
8   <sams:shape>
9     <gml:Point gml:id="point_sf_1">
10      <gml:pos srsName="http://www.opengis.net/def/crs/EPSSG/0/31287">391318.0 544564.0</gml:pos>
11    </gml:Point>
12  </sams:shape>
13 </sams:SF_SpatialSamplingFeature>
14 </sos:featureMember>
15 <sos:featureMember>
16 <sams:SF_SpatialSamplingFeature gml:id="sf_2">
17   <gml:identifier codeSpace="">org:zamg:MARIAZELL_ST_SEBASTIAN-FLUGFELD_7221</gml:identifier>
18   <sf:type xlink:href="http://www.opengis.net/def/samplingFeatureType/OGC-OM/2.0/SF_SamplingPoint"/>
19   <sf:sampledFeature xlink:href="urn:ogc:def:nil:OGC:unknown"/>
20   <sams:shape>
21     <gml:Point gml:id="point_sf_2">
22      <gml:pos srsName="http://www.opengis.net/def/crs/EPSSG/0/31287">434067.0 547505.0</gml:pos>
23    </gml:Point>
24  </sams:shape>
25 </sams:SF_SpatialSamplingFeature>
26 </sos:featureMember>
27 <sos:featureMember>
28 <sams:SF_SpatialSamplingFeature gml:id="sf_3">
29   <gml:identifier codeSpace="">org:zamg:AFLENZ_10200</gml:identifier>
30   <sf:type xlink:href="http://www.opengis.net/def/samplingFeatureType/OGC-OM/2.0/SF_SamplingPoint"/>
31   <sf:sampledFeature xlink:href="urn:ogc:def:nil:OGC:unknown"/>
32   <sams:shape>
33     <gml:Point gml:id="point_sf_3">
34      <gml:pos srsName="http://www.opengis.net/def/crs/EPSSG/0/31287">406949.0 543593.0</gml:pos>
35    </gml:Point>
36  </sams:shape>
37 </sams:SF_SpatialSamplingFeature>
38 </sos:featureMember>
39 <sos:featureMember>
40 <sams:SF_SpatialSamplingFeature gml:id="sf_4">
41   <gml:identifier codeSpace="">org:zamg:KAPFENBERG_13305</gml:identifier>
42   <sf:type xlink:href="http://www.opengis.net/def/samplingFeatureType/OGC-OM/2.0/SF_SamplingPoint"/>
43   <sf:sampledFeature xlink:href="urn:ogc:def:nil:OGC:unknown"/>
44   <sams:shape>
45     <gml:Point gml:id="point_sf_4">
46      <gml:pos srsName="http://www.opengis.net/def/crs/EPSSG/0/31287">395917.0 548200.0</gml:pos>
47    </gml:Point>
48  </sams:shape>
```

Abbildung 55: Teil der SOS 2.0 GetFeatureOfInterest Antwort

GetObservation:

In der GetObservation Operation aus Abbildung 56 wird zusätzlich zur Abfrage des Phänomens `urn:ogc:def:property:relative_humidity` ein zeitlicher und räumlicher Filter angewendet. Der räumliche Filter BBOX beinhaltet wieder dieselben Stationen wie im GetFeatureOfInterest Beispiel in Abbildung 54 und begrenzt die Ergebnisse auf die Stationen in Bruck an der Mur, St. Sebastian, Aflenz und Kapfenberg. Der zeitliche Filter *During* vom 1. April 2008 von 15:00:00 bis 18:00:00 Uhr beschränkt das Ergebnis aufgrund der zeitlichen Auflösung von einer Stunde auf zwei Messwerte pro *Feature-Of-Interest*.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope http://www.w3.org/2003/05/soap-envelope">
5   <env:Body>
6     <sos:GetObservation
7       service="SOS" version="2.0.0" xmlns:sos="http://www.opengis.net/sos/2.0"
8       xmlns:fes="http://www.opengis.net/fes/2.0" xmlns:gml="http://www.opengis.net/gml/3.2"
9       xmlns:swe="http://www.opengis.net/swe/2.0" xmlns:swes="http://www.opengis.net/swes/2.0"
10      xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
11      xsi:schemaLocation="http://www.opengis.net/sos/2.0
12      http://schemas.opengis.net/sos/2.0/sos.xsd">
13
14     <sos:observedProperty>urn:ogc:def:property:relative_humidity
15     </sos:observedProperty>
16     <sos:temporalFilter>
17       <fes:During>
18         <fes:ValueReference>phenomenonTime</fes:ValueReference>
19         <gml:TimePeriod gml:id="tp_1">
20           <gml:beginPosition>2008-04-01T15:00:00.000+02:00
21           </gml:beginPosition>
22           <gml:endPosition>2008-04-01T18:00:00.000+02:00</gml:endPosition>
23         </gml:TimePeriod>
24       </fes:During>
25     </sos:temporalFilter>
26     <sos:spatialFilter>
27       <fes:BBOX>
28         <fes:ValueReference>om:featureOfInterest/sams:SF_SpatialSamplingFeature/sams:shape
29         </fes:ValueReference>
30         <gml:Envelope srsName="http://www.opengis.net/def/crs/EPSG/0/31287">
31           <gml:lowerCorner>386884 534340</gml:lowerCorner>
32           <gml:upperCorner>439113 557676</gml:upperCorner>
33         </gml:Envelope>
34       </fes:BBOX>
35     </sos:spatialFilter>
36     <sos:responseFormat>http://www.opengis.net/om/2.0</sos:responseFormat>
37   </sos:GetObservation>
38 </env:Body>
39 </env:Envelope>
```

Abbildung 56: SOS 2.0 GetObservation Request mit zeitlichem (During) und räumlichen Filter (BBOX)

Abbildung 57 zeigt einen Ausschnitt der nach dem O&M Standard kodierten Antwort, welche aufgrund des zuvor beschriebenen angewendeten räumlichen Filters vier Klimastationen beinhaltet. So wurde beispielsweise am 1 April 2008 durch die *Procedure* der ID org:zamg:relative_humidity_AFLENZ_10200 um 16:00:00 Uhr eine relative Luftfeuchtigkeit von 49 Prozent und um 17:00:00 Uhr von 58 Prozent gemessen (swe:values). Im Bereich phenomenonTime ist die Zeitperiode entsprechend des definierten zeitlichen Filters angegeben, in der die aktuellen Ergebnisse der Eigenschaft urn:ogc:def:property: relative_humidity der vier Klimastationen zurückgegeben werden. Die Ergebnisse liegen als Quantität der Einheit Prozent (%) vor.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

```
7 <gml:TimePeriod gml:id="phenomenonTime_o_1376062551651">
8   <gml:beginPosition>2008-04-01T16:00:00.000+02:00</gml:beginPosition>
9   <gml:endPosition>2008-04-01T17:00:00.000+02:00</gml:endPosition>
10 </gml:TimePeriod>
11 </om:phenomenonTime>
12 <om:resultTime xlink:href="#phenomenonTime_o_1376062551651"/>
13 <om:procedure xlink:href="org:zamg:relative_humidity_AFLENZ_10200"/>
14 <om:observedProperty xlink:href="urn:ogc:def:property:relative_humidity"/>
15 <om:featureOfInterest xlink:title="org:zamg:AFLENZ_10200">
16   <sams:SF_SpatialSamplingFeature gml:id="sf_0">
17     <gml:identifier codeSpace="">org:zamg:AFLENZ_10200</gml:identifier>
18     <sf:type xlink:href="http://www.opengis.net/def/samplingFeatureType/OGC-OM/2.0/SF_SamplingPoint"/>
19     <sf:sampledFeature xlink:href="urn:ogc:def:nil:OGC:unknown"/>
20     <sams:shape>
21       <gml:Point gml:id="point_sf_0">
22         <gml:pos srsName="http://www.opengis.net/def/crs/EPSSG/0/31287">406949.0 543593.0</gml:pos>
23       </gml:Point>
24     </sams:shape>
25   </sams:SF_SpatialSamplingFeature>
26 </om:featureOfInterest>
27 <om:result xmlns:swe="http://www.opengis.net/swe/2.0" xsi:type="swe:DataArrayPropertyType">
28   <swe:DataArray>
29     <swe:elementCount>
30       <swe:Count>
31         <swe:value>2</swe:value>
32       </swe:Count>
33     </swe:elementCount>
34     <swe:elementType name="Components">
35       <swe:DataRecord>
36         <swe:field name="phenomenonTime">
37           <swe:Time definition="http://www.opengis.net/def/property/OGC/0/PhenomenonTime">
38             <swe:uom xlink:href="http://www.opengis.net/def/uom/ISO-8601/0/Gregorian"/>
39           </swe:Time>
40         </swe:field>
41         <swe:field name="relative_humidity">
42           <swe:Quantity definition="urn:ogc:def:property:relative_humidity">
43             <swe:uom code="%">
44           </swe:Quantity>
45         </swe:field>
46       </swe:DataRecord>
47     </swe:elementType>
48     <swe:encoding>
49       <swe:TextEncoding blockSeparator=";" decimalSeparator="." tokenSeparator=","/>
50     </swe:encoding>
51     <swe:values>2008-04-01T16:00:00.000+02:00,49.0;2008-04-01T17:00:00.000+02:00,58.0;</swe:values>
```

Abbildung 57: Teil der SOS 2.0 GetObservation Antwort

4.3.3.2 ZUGRIFF MITTELS CLIENT

Abbildung 58 zeigt den Zugriff des 52°North SOS Clients auf die Testdaten der ZAMG SOS Instanz, wobei die blauen Kreise mit dem Punkt in der Mitte alle implementierten *Feature-Of-Interest* darstellen. Für die rot markierte Station in Admont ist das Fenster zum Hinzufügen einer Zeitreihe geöffnet. Das Fenster zeigt den ausgewählten Parameter *precipitation* (Niederschlag), der mit dem Ombrometer gemessen wird. Zusätzlich werden die ID der *Procedure* und des *Feature-Of-Interest* sowie die Seehöhe der Station gezeigt.

4 SOS-Implementierung der Instanzen WegenerNet und ZAMG

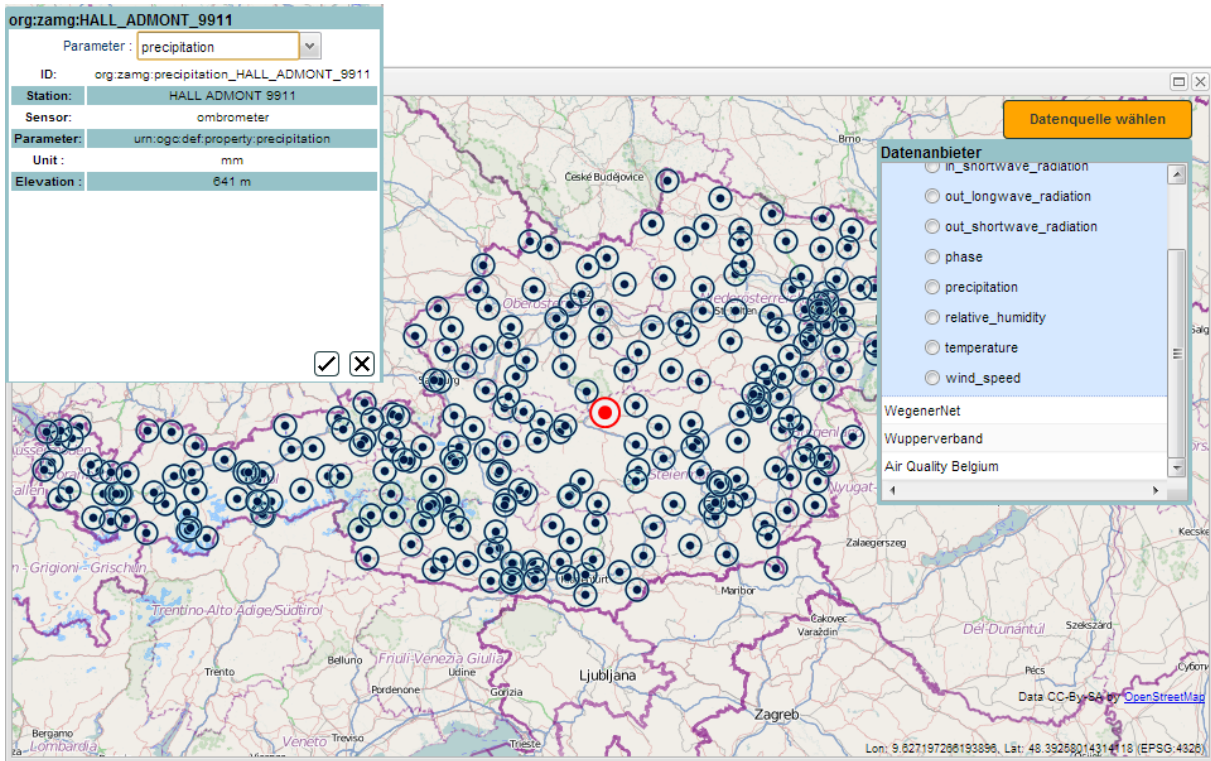


Abbildung 58: ZAMG-Stationsübersicht des 52°North SensorWebClient, Auswahlfenster zur Station in Admont

Abbildung 59 stellt Zeitreihen der relativen Luftfeuchtigkeit ausgewählter Stationen dar. Die im Grünnton dargestellten Trendlinien werden von der ZAMG SOS Instanz abgefragt, die Rot-Orange-Töne stellen abgefragte Daten der Wupperverband SOS Instanz dar.

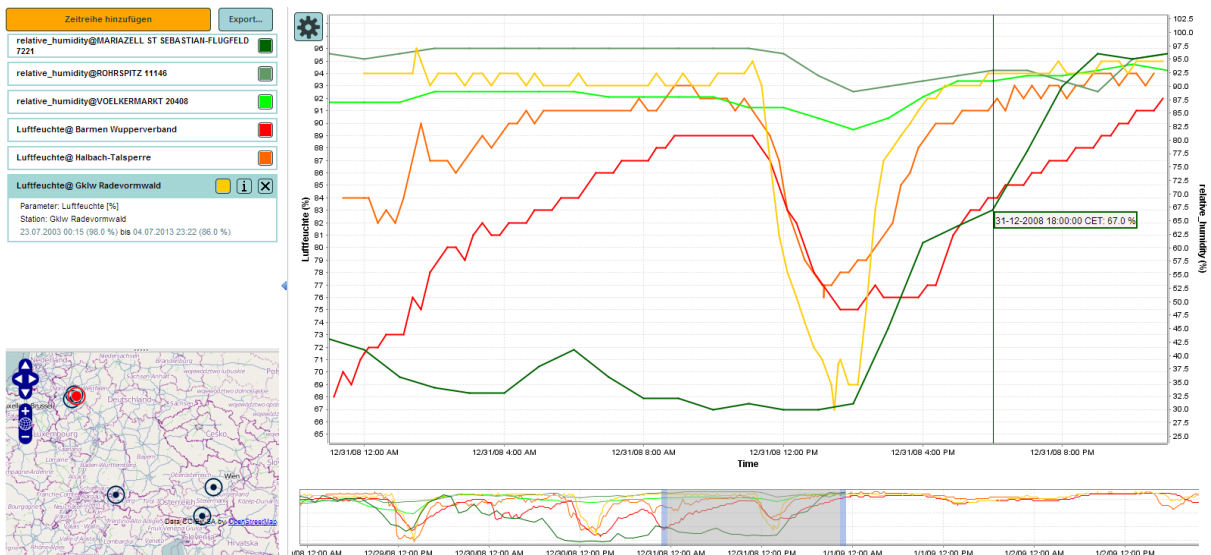


Abbildung 59: Zeitreihen Darstellung der relativen Luftfeuchtigkeit mit dem 52°North SensorWebClient (ZAMG Daten kombiniert mit Daten des Wupperverbands)

5 FAZIT UND AUSBLICK

Standardisierte Services und Datenformate müssen eingesetzt werden, um einen einfachen Datenaustausch zwischen verschiedensten Sensornetzwerken und Datenanbietern zu gewährleisten. Die SWE Initiative führte 2003 aufgrund von Datenaustausch- und Datenintegrationsschwierigkeiten generische und funktional umfangreiche Standards zu Datenformaten und Services ein. Systemarchitekturen auf Basis standardisierter Schnittstellen sind im Gegensatz zu monolithischen Sensornetzwerkimplementierung modular und verteilt aufgebaut. Ein Systemanbieter kann so seine Systemanforderungen durch Kombination einzelner standardisierter Services realisieren. Neben dem einfachen Datenaustausch ist ein weiterer großer Vorteil der Zugriff auf Echtzeitdaten und die damit einhergehende Möglichkeit einer Real-Time-Analyse durch Integration dieser Daten z.B. in ein GIS. Außerdem können Sensordaten in Echtzeit bzw. in naher Echtzeit auf ihre Qualität überprüft und gegebenenfalls korrigiert und prozessiert werden.

Die in Kapitel 3 vorgestellten Real-World-Implementierungen der Forschungsprojekte zeigen bereits die produktive Umsetzbarkeit solcher standardisierten interoperablen Geo-Sensornetzwerke. Vor allem die Architekturen des *Live Geography* und OSIRIS Ansatzes zur Messung und Integration von Echtzeitdaten in eine voll standardisierte Infrastruktur ermöglichen den portablen und flexiblen Einsatz dieser Architekturen in verschiedensten Anwendungsbereichen. Durch ein *Embedded System Design* können Sensoren praktisch über *Plug and Play* in ein Sensornetzwerk integriert werden. Aufgrund der auf den *Embedded Computers* vorinstallierten SOS und SAS können *Observations* und abonnierte Alarmer auf einfache Weise über die standardisierten Schnittstellen verteilt werden.

Für eine domänenunabhängige Integration heterogener Sensoren beliebiger Art, von Temperatursensoren bis hin zu Satelliten, basieren die SWE Spezifikationen auf einem generischen Ansatz.

Die SWE Spezifikationen basieren auf einem generischen Ansatz, um heterogene Sensoren beliebiger Art, von Temperatursensoren bis hin zu Satelliten, domainunabhängig in ein System integrieren zu können. Eine bestehende Herausforderung der Initiative ist die Gewährleistung einer vollständigen Interoperabilität, trotz des flexiblen Charakters solch

Fazit und Ausblick

eines generischen Ansatzes. Fachschalen spezifische Profile wie beispielsweise die *Water Markup Language 2.0 Spezifikation* sollen diesem Problem entgegenwirken (Bröring, et al., 2011; Jirka, et al., 2013).

Die Umsetzung zweier SOS Instanzen im praktischen Teil der Arbeit zeigt die Möglichkeit eines Direktzugriffes auf Echtzeitdaten oder Datenarchive. Verschiedenste SOS Clients können durch die realisierten Standards auf die Datenbasis zugreifen und die aktuellen Messwerte der Sensoren sowie Zeitreihen darstellen oder analysieren. Die gemessenen Parameter der WegenerNet und ZAMG Instanzen sind durch die realisierten XML-Schemata zur Messdatenkapselung und standardisierte Schnittstellen einfach untereinander und mit anderen, mittels SOS zur Verfügung gestellten Sensordaten, kombinierbar. Da jetzt alle Beziehungen SOS konform realisiert sind, ist die Möglichkeit der einfachen Umsetzung weiterer Services, wie z.B. ein SES oder WNS, gegeben.

Die WegenerNet Implementierung ist derzeit so umgesetzt, dass die *Observations* aus der PostgreSQL Datenbank des Wegener Centers in eine weitere PostgreSQL Datenbank, in der die Beziehungen nach dem O&M Modell realisiert sind, kopiert werden. Die C++-Applikation bringt als Zwischenschritt die *Observations* in eine SOS konforme Struktur. Die zwei unterschiedlichen Systeme verhindern eine nahtlose Integration neuer Sensoren und damit die Eigenschaft der Skalierbarkeit. Beispielsweise erlaubt die *InsertSensor* Operation der *Transactional Extension* das Registrieren neuer Sensoren in das SOS. Wird diese Operation angewendet, so sind die neuen Informationen nur seitens der SOS Datenbank gespeichert, das System des Wegener Centers weiß jedoch nichts vom neuen Sensor. Selbiges Problem besteht bei der *DeleteSensor* Operation, welche das Löschen neuer Sensoren und deren zugehörige *Observations* aus der Datenbasis verursacht. Beim Ausführen dieser Operation werden die registrierten Sensoren und zugehörigen *Observations* aus der SOS Datenbank entfernt, im ursprünglichen WegenerNet System bleiben diese jedoch vorhanden. Als Problemlösung kann wieder eine C++ Applikation entwickelt werden, die eine Konvertierung der SOS Datenformate in die WegenerNet konforme Struktur, also in die andere Richtung vornimmt. Zielführender wäre jedoch das direkte Speichern der von den Datenloggern über GPRS gesendeten *Observations* am SOS Server. Eine derartige Implementierung erspart einerseits den aufwendigen Konvertierungsschritt der Datenformate zwischen zwei Systemen und andererseits wird die Rechen- sowie Datenübertragungszeit verringert.

Fazit und Ausblick

Im WegenerNet SOS sind derzeit nur die Level 0 Rohdaten implementiert. Zukünftig könnte zusätzlich der standardisierte Zugriff auf die Level 1 Daten (qualitätsgeprüfte Messdaten mit Qualitätsmarken) sowie auf die Level 2 Daten (Wetter-, Klimaprodukte und aufbereitete Stations- und Gitterdaten) umgesetzt werden (siehe Kapitel 4.2.1 Allgemeines). Um dabei eine standardisierte SOS Struktur zu gewährleisten, können *Procedures* definiert werden, die sowohl eine Qualitätsprüfung zur Erzeugung der Level 1 Daten durchführen, als auch Wetter- und Klimaprodukte sowie aufbereitete Stations- und Gitterdaten erzeugen. Zur Darstellung der rasterbasierten Gitterdaten als Kartenbild kann ein *Web Map Service* (WMS) realisiert werden. Ein *Web Coverage Service* (WCS) bietet die Möglichkeit, die verfügbaren interpolierten Gitterdaten mit Metadaten zur weiteren Auswertung bereitzustellen.

Da von jeder Klimastation Parameter zur Überwachung des Messbetriebes gesendet werden, wäre eine push-basierte Übertragung von Wartungsmeldungen und Alarmen mittels SES und WNS eine weitere sinnvolle Ergänzung der Implementierung. Ein Benutzer könnte so Wartungsregeln definieren, um durch ein bestimmtes Protokoll (z.B. SMS, E-Mail, Telefon) über den Wartungszustand einer Station automatisch benachrichtigt zu werden. Abbildung 60 zeigt die Eigenschaften der Messstationen, aus deren Messwerte sinnvolle Wartungsmeldungen generiert werden können.

	phenomenon_id [PK] character varying(100)	phenomenon_description character varying(200)	unit character va	valuetype character varying(40)
2	urn:ogc:def:property:GSM_signal_strength	GSM signal strength	''	numericType
3	urn:ogc:def:property:logger_battery_voltage	logger battery voltage	V	numericType
4	urn:ogc:def:property:logger_humidity	logger humidity	%	numericType
5	urn:ogc:def:property:logger_storage_occupancy	logger storage occupancy	''	numericType
6	urn:ogc:def:property:logger_supply_voltage	logger supply voltage	''	numericType
7	urn:ogc:def:property:logger_temperature	logger temperature	°C	numericType

Abbildung 60: Auszug der PostgreSQL Tabelle *phenomenon*

Im Falle der ZAMG Instanz wurden Testdaten aus dem Jahr 2008 in das SOS integriert. Um eine Abfrage, Darstellung und Analyse der Daten in Echtzeit zu gewährleisten, müsste ein Direktzugriff auf die Sensordaten erfolgen. Derzeit können nur die In-Situ-Messwerte des Jahres 2008 mittels SOS 1.0.0 oder SOS 2.0 Schnittstelle abgefragt und weiterverwendet werden.

6 LITERATURVERZEICHNIS

- 52North, 2013. *52°North Initiative for Geospatial Open Source Software GmbH*. [Online]
URL: <http://52north.org/communities/sensorweb/sps/1.0.0/>
[Zugriff am 14 Juli 2013].
- 52North, 2013. *52°North Initiative for Geospatial Open Source Software GmbH*. [Online]
URL: <http://52north.org/communities/sensorweb/sas/1.0.0/index.html>
[Zugriff am 14 Juli 2013].
- 52North, 2013. *52°North Initiative for Geospatial Open Source Software GmbH*. [Online]
URL: <http://52north.org/communities/sensorweb/wns/1.0.0/>
[Zugriff am 14 Juli 2013].
- Bernd, R. et al., 2009. Real-time Geo-awareness – Sensor Data Integration for Environmental Monitoring in the City. *The International Conference on Advanced Geographic Information Systems & Web Services (GEOWS 2009)*, 1-7 Februar, S. 92-97.
- Botts, M. & Robin, A., 2007. *OpenGIS: Sensor Model Language (SensorML)*. [Online]
URL: http://portal.opengeospatial.org/files/?artifact_id=21273
[Zugriff am 4 Mai 2013].
- Bredel, H., 2013. *52°North Initiative for Geospatial Open Source Software GmbH*. [Online]
URL: <https://wiki.52north.org/bin/view/SensorWeb/SensorWebClient>
[Zugriff am 7 August 2013].
- Bröring, A. et al., 2011. New Generation Sensor Web Enablement. *Open Access*, 1 März, S. 2652-2699.
- Bröring, A., Stasch, C. & Johannes, E., 2012. *Sensor Observation Service Interface Standard*.
[Online]
URL: https://portal.opengeospatial.org/files/?artifact_id=47599
[Zugriff am 12 Februar 2013].

Literaturverzeichnis

Cox, S., 2007. *Observations and Measurements - Part 1 - Observation schema*. [Online]

URL: http://portal.opengeospatial.org/files/?artifact_id=22466

[Zugriff am 05 Juli 2013].

Cox, S., 2010. *Geographic Information: Observations and Measurements*. [Online]

URL: http://portal.opengeospatial.org/files/?artifact_id=41579

[Zugriff am 25 Juni 2013].

Echterhoff, J., 2011. *OpenGIS® SWE Service Model*. [Online]

URL: http://portal.opengeospatial.org/files/?artifact_id=38476

[Zugriff am 07 Juli 2013].

ESA, 2012. *GEOWOW*. [Online]

URL: <http://www.geowow.eu>

[Zugriff am 23 Juli 2013].

GEO, 2013. *Group on Earth Observation*. [Online]

URL: <http://www.earthobservations.org/geoss.shtml>

[Zugriff am 23 Juli 2013].

Gross, N., 1999. *BloombergBusinessweek, Businessweek Archives*. [Online]

URL: <http://www.businessweek.com/stories/1999-08-29/14-the-earth-will-don-an-electronic-skin>

[Zugriff am 14 Mai 2008].

Henson, C. A., Pschorr, J. K., Sheth, A. P. & Thirunarayan, K., 2009. SemSOS: Semantic Sensor Observation Service. *Proc. of the 2009 International Symposium on Collaborative Technologies and Systems (CTS 2009)*, 22 Mai.

Hollmann, C., 2012. *Sensor Web Enablement, Installation Guide for 52°North Sensor Observation Service*, s.l.: 52°North.

IOOS, 2010. *Integrated Ocean Observing System*. [Online]

URL: http://www.ioos.noaa.gov/library/ioos_dif_assmnt_report_final.pdf

[Zugriff am 24 Juli 2013].

Literaturverzeichnis

- Jirka, S., Bredel, H., Utech, M. & Andres, V., 2013. Erhöhte Interoperabilität durch Sensor Web-Profile - Praxisbeispiele aus der Hydrologie. In: *Geoinformationssysteme 2013, Beiträge zum 18. Münchner Fortbildungsseminar*. neue Ausg Hrsg. Berlin: Wichmann H, S. 265-271.
- Kabas, T., 2011. *WegenerNet Klimastationsnetz Region Feldbach: Experimenteller Aufbau und hochauflösende Daten für die Klima- und Umweltforschung*. Graz: UNI-Graz.
- Koch, A., Bill, R. & Donaubaue, A., 2013. *Geoinformationssysteme 2013, Beiträge zum 18. Münchner Fortbildungsseminar*. neue Ausg Hrsg. Berlin: Wichmann H.
- Kresse, W. & Fadaie, K., 2004. *ISO Standards for Geographic Information*. Berlin Heidelberg: Springer.
- Massachusetts Institute of Technology, 2013. *Senseable City Lab*. [Online]
URL: <http://senseable.mit.edu/>
[Zugriff am 17 Juli 2013].
- Massachusetts Institute of Technology, o.J.. *Senseable City Lab*. [Online]
URL: <http://senseable.mit.edu/trashtrack/visualizations.php?id=2>
[Zugriff am 18 Juli 2013].
- Massachusetts Institute of Technology, o.J.. *Senseable City Lab*. [Online]
URL: <http://senseable.mit.edu/guggenheim/>
[Zugriff am 17 Juli 2013].
- Mittelböck, M. & Resch, B., 2009. Live Geography - Real-time Analyse mit GI-Systemen. *GISBusiness*, Issue 5, S. 36-39.
- Mittelböck, M. et al., 2011. Geomonitoring – Kernbaustein für räumliche Wissensgenerierung in Geoinformationsinfrastrukturen. In: *Angewandte Geoinformatik 2011, Beiträge zum 23. AGIT-Symposium Salzburg*. Salzburg: Wichmann, S. 750-755.
- Mittelböck, M. et al., 2009. Standardisierte Embedded Sensornetze zur Integration Lokationsbezogener „live“ Information in GI-Systeme. In: *Angewandte Geoinformatik 2009*. Heidelberg: Wichmann Verlag, S. 178-183.

Literaturverzeichnis

NOAA, 2013. *National Data Buoy Center*. [Online]

URL: <http://www.ndbc.noaa.gov/>

[Zugriff am 25 Juli 2013].

Norm DIN EN ISO 19123, 2007. *Geoinformation – Coverage Geometrie- und Funktionsschema (ISO 19123:2005); Englische Fassung EN ISO 19123:2007*. Berlin: Deutsches Institut für Normung e.V..

O'Reilly, T., 2012. *OGC® PUCK Protocol Standard*. [Online]

URL: https://portal.opengeospatial.org/files/?artifact_id=47604

[Zugriff am 16 Juli 2013].

OGC, 2013. *About OGC*. [Online]

URL: <http://www.opengeospatial.org/ogc>

[Zugriff am 17 Juli 2013].

Open Geospatial Consortium, Inc., 2011. *OGC Network*. [Online]

URL: http://www.ogcnetwork.net/sos_2_0/tutorial/capabilities

[Zugriff am 12 Juli 2013].

Open Geospatial Consortium, 2013. *Sensor Web Enablement DWG*. [Online]

URL: <http://www.opengeospatial.org/projects/groups/sensorwebdwg>

[Zugriff am 26 Juni 2013].

ORF, 2013. *OE1.ORF.at*. [Online]

URL: <http://oe1.orf.at/programm/327022>

[Zugriff am 17 Juli 2013].

OSGeo, 2013. *OSGeoLive*. [Online]

URL: http://live.osgeo.org/de/standards/sensorml_overview.html

[Zugriff am 20 August 2013].

OSIRIS, 2009. *Open architecture for Smart and Interoperable networks In Risk management based on In-situ Sensors*. [Online]

URL: <http://tws01.terranovaws.it/osiris/doc/A2%20OSIRIS-WP8000-DEL-0054->

Literaturverzeichnis

[Publishable%20final%20activity-Cons_Report-revB.pdf](#)

[Zugriff am 25 Juli 2013].

Ratti, C. & Townsend, A., 2012. Die smarte Stadt der Zukunft. *Spektrum der Wissenschaft*, Issue 4, S. 62-67.

Reed, C., Botts, M., Percivall, G. & John, D., 2013. *OGC® Sensor Web Enablement: Overview And High Level Architecture. White Paper OGC 07-165r1*. [Online]

URL: http://portal.opengeospatial.org/files/?artifact_id=25562

[Zugriff am 24 Juni 2013].

Reimer, O. & Kampe, J., 2008. *plug & play - fähige Signalaufbereitung und –wandlung von Messdaten IEEE 1451.4 kompatibler Sensoren unter Verwendung eines programmierbaren mixed-signal Schaltkreises*. Siegen, ANALOG '08.

Resch, B., 2010. Geo-sensor Web - Echtzeitmessungen für Ubiquitäre Monitoring-Systeme. *HMD Journal Praxis der Wirtschaftsinformatik*, S. 50-58.

Resch, B., 2013. Standardisierte Geosensornetzwerke für Umweltbeobachtung in naher Echtzeit. In: *Geoinformationssysteme 2013, Beiträge zum 18. Münchner Fortbildungsseminar*. neue Ausg Hrsg. Berlin: Wichmann H, S. 261-263.

Resch, B. & Mittelböck, M., 2010. Live Geography: Interoperable Geo-Sensor Webs Facilitating the Vision of Digital Earth. *International Journal on Advances in Networks and Services*, Issue 3, S. 323-332.

Simonis, I. & Echterhoff, J., 2006. *Web Notification Service*. [Online]

URL: http://portal.opengeospatial.org/files/?artifact_id=18776

[Zugriff am 14 Juli 2013].

Simonis, I. & Echterhoff, J., 2011. *Sensor Planning Service (SPS)*. [Online]

URL: http://portal.opengeospatial.org/files/?artifact_id=38478

[Zugriff am 06 Juli 2013].

Literaturverzeichnis

Whiteside, A., 2007. *OGC Web Services Common Specification*. [Online]

URL: http://portal.opengeospatial.org/files/?artifact_id=38867

[Zugriff am 07 Juli 2013].

Wupperverband, 2013. *FluGGS*. [Online]

URL: <http://fluggs.wupperverband.de/web/fluggs/start>

[Zugriff am 22 Juli 2013].

X-border-GDI, 2010. *CROSS-BORDER GEODATA INFRASTRUCTURE*. [Online]

URL: <http://www.x-border-gdi.org/de/projekte/details/project/wcm/index.html>

[Zugriff am 22 Juli 2013].

ZAMG, 2013. *Zentralanstalt für Meteorologie und Geodynamik*. [Online]

URL: <http://www.zamg.ac.at/cms/de/klima/messnetze/wetterstationen>

[Zugriff am 8 August 2013].

