

# Sonderveröffentlichung

## **Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten**

Eine Heranführung an Sensor Web-Technologien

Mai 2018



Die Deutsche Vereinigung für Wasserwirtschaft, Abwasser und Abfall e. V. (DWA) setzt sich intensiv für die Entwicklung einer sicheren und nachhaltigen Wasser- und Abfallwirtschaft ein. Als politisch und wirtschaftlich unabhängige Organisation arbeitet sie fachlich auf den Gebieten Wasserwirtschaft, Abwasser, Abfall und Bodenschutz.

In Europa ist die DWA die mitgliederstärkste Vereinigung auf diesem Gebiet und nimmt durch ihre fachliche Kompetenz bezüglich Regelsetzung, Bildung und Information sowohl der Fachleute als auch der Öffentlichkeit eine besondere Stellung ein. Die rund 14 000 Mitglieder repräsentieren die Fachleute und Führungskräfte aus Kommunen, Hochschulen, Ingenieurbüros, Behörden und Unternehmen.

## Impressum

### Herausgeber und Vertrieb:

DWA Deutsche Vereinigung für  
Wasserwirtschaft, Abwasser und Abfall e. V.  
Theodor-Heuss-Allee 17  
53773 Hennef, Deutschland  
Tel.: +49 2242 872-333  
Fax: +49 2242 872-100  
E-Mail: [info@dwa.de](mailto:info@dwa.de)  
Internet: [www.dwa.de](http://www.dwa.de)

### Satz:

Christiane Krieg, DWA

### Druck:

XXXXXXXXXXXXXXXXXXXX

### ISBN:

XXX (Print)

XXX (E-Book)

Gedruckt auf 100 % Recyclingpapier

© DWA Deutsche Vereinigung für Wasserwirtschaft, Abwasser und Abfall e. V., Hennef 2018

Alle Rechte, insbesondere die der Übersetzung in andere Sprachen, vorbehalten. Kein Teil dieser Publikation darf ohne schriftliche Genehmigung des Herausgebers in irgendeiner Form – durch Fotokopie, Digitalisierung oder irgendein anderes Verfahren – reproduziert oder in eine von Maschinen, insbesondere von Datenverarbeitungsanlagen, verwendbare Sprache übertragen werden.

# Inhalt

<b>Bilderverzeichnis</b> .....	<b>5</b>
<b>Tabellenverzeichnis</b> .....	<b>5</b>
<b>Abkürzungen</b> .....	<b>7</b>
<b>Glossar</b> .....	<b>7</b>
<b>1 Einleitung</b> .....	<b>8</b>
1.1 Zusammenfassung .....	8
1.2 Einführung .....	9
1.3 Die Sensor Web-Technologie des OGC.....	10
1.4 Anwendungsfelder der Technologie.....	12
1.5 Technische Herausforderungen in der Umsetzung der Sensor Web-Technologie..	14
<b>2 Technologische Grundlagen</b> .....	<b>15</b>
2.1 OGC Sensor Observation Service.....	15
2.2 OGC SOS 2.0 Hydrology Profile.....	16
2.3 ISO/OGC Observations and Measurements.....	16
2.4 OGC WaterML 2.0.....	17
2.5 WaterML-WQ .....	18
2.6 OGC TimeseriesML .....	18
2.7 OGC Sensor Model Language .....	19
2.8 INSPIRE.....	19
2.8.1 Nutzung von OGC SOS-Servern als INSPIRE Download Service .....	19
2.8.2 O&M und WaterML 2.0 als INSPIRE Datenformat.....	20
2.9 Zusätzliche Funktionen über den SOS-Standard hinaus .....	21
2.9.1 Zustellung von Benachrichtigungen/Abonnieren von Datenströmen .....	21
1.1.1 Steuerung von Sensoren .....	22
<b>3 Anforderungen an SOS-Implementierungen in der Hydrologie</b> .....	<b>23</b>
3.1 Anforderungen an Server-Implementierungen .....	23
3.1.1 Allgemeine Anforderungen .....	23
3.1.2 Zu unterstützende SOS-Operationen .....	23
3.1.2.1 GetCapabilities.....	23
3.1.2.2 GetObservation .....	23
3.1.2.3 DescribeSensor .....	24
3.1.2.4 GetFeatureOfInterest.....	24
3.1.2.5 GetDataAvailability.....	24
3.1.2.6 Spatial Filtering Profile .....	25
3.1.2.7 Transaktionale SOS-Operationen.....	25
3.1.3 INSPIRE.....	25
3.1.4 Konformitätsprüfung .....	26
3.1.5 Weitere Funktionen/Operationen .....	26
3.1.6 Systemumgebung .....	26
3.1.7 Ansätze zur Datenhaltung .....	27
3.1.7.1 Verwendung bestehender Datenbanken .....	27

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

3.1.7.1.1	Entwicklung eines spezifischen Datenbankzugriffs-Moduls .....	28
3.1.7.1.2	Datenbank-Abstraktionsschicht.....	29
3.1.7.1.3	Datenbank-Views (Sichten).....	30
3.1.7.2	Erstellung separater Datenbanken für SOS-Server .....	31
3.1.7.2.1	Datenimport aus bestehender Datenbank .....	31
3.1.7.2.2	Verwendung transaktionaler SOS-Operationen zum Datenimport .....	32
3.1.7.3	Proxy-Lösungen.....	34
3.1.7.4	Fachanwendungen mit nativer Datenhaltung .....	35
3.1.7.5	Überblick.....	35
3.2	Anforderungen an Client-Implementierungen .....	38
3.2.1	Datenauswahl .....	38
3.2.2	Datendarstellung .....	39
3.2.3	Weitere funktionale Anforderungen.....	42
3.2.3.1	Metadaten-Darstellung .....	42
3.2.3.2	Generalisierung .....	42
3.2.3.3	Verlinkung.....	43
3.2.3.4	Datendownload .....	43
3.2.3.5	Einbindung externer SOS-Instanzen .....	44
3.2.4	Systemumgebungen .....	44
3.2.4.1	Client-Plattformen.....	44
3.2.4.2	Server.....	45
3.3	Nichtfunktionale Anforderungen.....	45
3.3.1	Dokumentation .....	45
3.3.2	Lizensierung .....	45
3.3.3	Leistungsanforderungen .....	45
3.3.4	Skalierbarkeit .....	46
3.3.5	Ausfallsicherheit.....	46
3.3.6	Software Erstellungs- und Supportkonzept.....	47
3.4	Zugriffskontrolle.....	47
3.4.1	Zugriffskontrolle auf Datenbankebene .....	47
3.4.2	Zugriffskontrolle auf Ebene der SOS-Implementierung.....	48
3.4.3	Zugriffskontrolle durch Fachanwendung.....	48
3.4.4	Separate Zugriffskontrolllösungen .....	48
<b>4</b>	<b>Anwendungsbeispiele.....</b>	<b>49</b>
4.1	Beispielhafte Sensor Web-Umsetzungen .....	49
4.1.1	Wupperverband – FluGGS .....	49
4.1.2	Bundesanstalt für Gewässerkunde (BfG) – Meeresumweltdatenbank .....	51
4.1.3	KISTERS .....	53
4.1.4	PEGELONLINE .....	53
4.1.5	Nachnutzung von Sensor Web-Diensten (Beispiel Hochwasser und Sumpfung).....	55
4.2	Beispiele für Sensor Web Software.....	56
4.2.1	Server-Implementierungen.....	56
4.2.1.1	KISTERS KiWIS .....	56
4.2.1.2	52°North SOS.....	57
4.2.2	Client Implementierungen .....	57

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

4.2.2.1	SOS.js .....	57
4.2.2.2	52°North Helgoland .....	58
4.2.2.3	KISTERS Timeseries Widget .....	59
4.2.2.4	con terra map.apps SOS-Erweiterung .....	60
4.2.2.5	WISKI .....	61
4.2.2.6	Esri ArcGIS GeoEvent Server .....	62
4.2.2.7	QGIS GML Application Schema Toolbox .....	63
4.2.2.8	sos4R .....	65
4.2.2.9	Weitere GIS-Anbindungen .....	65
<b>5</b>	<b>Möglichkeiten und Grenzen der SOS- und WaterML 2.0-Standards .....</b>	<b>66</b>
5.1	SOS-Antwortverhalten .....	66
5.2	Performance beim Rendering von Zeitreihen im Client .....	68
<b>Anhang A Textbausteinvorschläge zur Verwendung in Ausschreibungen .....</b>		<b>69</b>
<b>Anhang B Beispiele für Sensor Web-Datenformate .....</b>		<b>75</b>
B.1	SOS-Antwort: Messdaten als OGC O&M-Observation .....	75
B.2	SOS-Antwort: Messdaten als WaterML 2.0 Time Value Pair-Observation .....	76
B.3	SOS-Antwort: Messdaten als INSPIRE PointTimeSeriesObservation .....	78
<b>Anhang C Referenzen .....</b>		<b>82</b>
<b>Anhang D Autorenverzeichnis .....</b>		<b>83</b>

## Bilderverzeichnis

Abbildung 1:	Überblick über ausgewählte OGC SWE-Standards .....	10
Abbildung 2:	Beziehungen zwischen Client-Anwendungen und Datenquelle unter Nutzung der SOS-Standards .....	11
Abbildung 3:	Illustration der Grundlagen des O&M-Modells .....	16
Abbildung 4:	Familie der OGC WaterML 2.0-Standards .....	17
Abbildung 5:	Schematische Darstellung der Verwendung eines spezifischen Datenbankzugriffs-Moduls .....	28
Abbildung 6:	Schematische Darstellung der Verwendung einer Datenbank-Abstraktionsschicht .....	29
Abbildung 7:	Schematische Darstellung der Verwendung von Datenbank-Views .....	30
Abbildung 8:	Schematische Darstellung des Datenimports aus bestehender Datenbank per SQL-Skript .....	32
Abbildung 9:	Beispiel für das Konfigurationsmenü eines SOS-Feeders .....	33
Abbildung 10:	Schematische Darstellung der Verwendung transaktionaler SOS-Operationen zum Datenimport .....	33
Abbildung 11:	Schematische Darstellung der Nutzung eines Datenzugriffsdienstes .....	34
Abbildung 12:	Beispiel für die kartenbasierte Auswahl von Zeitreihen .....	38
Abbildung 13:	Beispiel für die listenbasierte Auswahl von Zeitreihen .....	39
Abbildung 14:	Beispiel zur Tabellenansicht .....	40
Abbildung 15:	Beispiel zur Diagrammansicht .....	41
Abbildung 16:	Beispiel zur Kartenansicht mit Trendanzeige .....	42

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Abbildung 17: Schematische Darstellung einer Zugriffskontrolllösung.....	48
Abbildung 18: Sensor Web Infrastruktur beim Wupperverband (Quelle: Dipl.-Geogr. Christian Förster, Wupperverband).....	50
Abbildung 19: Sensor Web Client des Wupperverbandes (online verfügbar unter <a href="http://fluggs.wupperverband.de/swc">http://fluggs.wupperverband.de/swc</a> ) .....	51
Abbildung 20: Datenfluss von der MUDAB in die SOS-Datenbank .....	52
Abbildung 21: Übersicht der Inhalte der MUDAB in einem Web-Client .....	53
Abbildung 22: Architektur des PEGELONLINE SOS-Servers .....	54
Abbildung 23: PEGELONLINE SOS-Viewer .....	55
Abbildung 24: Hochwasseralarme im LINEG Infoboard (Detail-Liste).....	56
Abbildung 25: SOS.js – Beispiel für einen SOS-Client, welcher direkt mit SOS-Servern interagiert.....	58
Abbildung 26: SOS.js – Diagrammdarstellung .....	58
Abbildung 27: Server-seitige Kapselung der Geschäftslogik zur Interaktion mit SOS-Servern .....	59
Abbildung 28: JavaScript SOS-Client – Menü zur Auswahl einer Zeitreihe .....	59
Abbildung 29: KISTERS Timeseries Widget mit serverseitig produzierten Grafiken.....	60
Abbildung 30: KISTERS WIDGET SOS-Implementierung für WorldWaterOnline .....	60
Abbildung 31: map.apps SOS-Erweiterung.....	61
Abbildung 32: WISKI Desktop – SOS Consumer .....	61
Abbildung 33: Esri ArcGIS GeoEvent Server-Workflow .....	62
Abbildung 34: Visualisierung von Wasserständen entlang des Mains (aus PEGELONLINE) in ArcGIS Online .....	62
Abbildung 35: QGIS-Darstellung von WaterML 2.0-Inhalten als XML-Baumstruktur .....	64
Abbildung 36: Zeitreihendarstellung in QGIS.....	65
Abbildung 37: SOS-Ladezeiten.....	67

## Tabellenverzeichnis

Tabelle 1: Beispielhafte Einsatzszenarien der Sensor Web-Technologie .....	13
Tabelle 2: WaterML 2.0 Observation-Typen.....	18
Tabelle 3: Umsetzung der verschiedenen Operationen eines INSPIRE Download Service durch entsprechende SOS-Funktionalität .....	20
Tabelle 4: Ausgewählte INSPIRE Observation-Typen .....	21
Tabelle 5: Überblick über Ansätze zur Datenhaltung eines SOS-Servers.....	36
Tabelle 6: Textbausteinvorschlage zur Verwendung in Ausschreibungen .....	69

## Abkürzungen

BfG	Bundesanstalt für Gewässerkunde
DWA	Deutsche Vereinigung für Wasserwirtschaft, Abwasser und Abfall e.V.
GIS	Geoinformationssystem
INSPIRE	Infrastructure for Spatial Information in the European Community
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
O&M	Observations and Measurements
OGC	Open Geospatial Consortium
REST	Representational State Transfer
SensorML	Sensor Model Language
SOS	Sensor Observation Service
SPS	Sensor Planning Service
SWE	Sensor Web Enablement
TSML	TimeseriesML
XML	Extensible Markup Language

## Glossar

### INSPIRE

Die europäische INSPIRE-Richtlinie, welche in Deutschland durch das Geodatenzugangsgesetz umgesetzt wurde, dient dem Aufbau einer interoperablen europäischen Geodateninfrastruktur. Neben Verpflichtungen zur Bereitstellung bestimmter Daten durch öffentliche Stellen umfasst INSPIRE auch umfassende technische Regelungen und Empfehlungen, in welcher Form die Bereitstellung von raumbezogenen Daten erfolgen soll bzw. kann.

### Observations and Measurements

Der Observations and Measurements-Standard (O&M) definiert ein konzeptionelles Modell für Messdaten sowie eine zugehörige Encoding-Spezifikation, welche vorschreibt, wie Messdaten in XML kodiert werden.

### Open Geospatial Consortium

Das Open Geospatial Consortium (OGC) ist eine internationale de-facto Standardisierungsorganisation mit Mitgliedern aus Forschung, Industrie und Verwaltung. Ein Schwerpunkt des OGC ist die Entwicklung von Standards zum Aufbau interoperabler Geodateninfrastrukturen.



## **Sensor Observation Service:**

Der OGC Sensor Observation Service-Standard (SOS) ist eine Schnittstellenspezifikation für Web-Dienste zum Zugriff auf Messdaten. Es wird also beispielweise festgelegt, welche Operationen zum Messdatenabruf genutzt werden können und welche Parameter diese Operationen bieten.

## **Sensor Web**

Der Begriff Sensor Web beschreibt die Integration von Sensoren und deren Messdaten in webbasierte Infrastrukturen. Dies kann als Erweiterung der Idee von Geodateninfrastrukturen angesehen werden, sodass Messdaten als neue Geoinformations-Quelle hinzukommen.

## **Sensor Web Enablement**

Unter Sensor Web Enablement (SWE) wird eine Reihe von OGC-Standards zusammengefasst, welche die Interoperabilität innerhalb von Sensor Web-Infrastrukturen sicherstellen. Dies umfasst sowohl einheitliche (webbasierte) Schnittstellen als auch Datenformate.

## **TimeseriesML**

TimeseriesML (TSMML) abstrahiert die Konzepte von WaterML 2.0 sodass diese auch auf die Modellierung und XML-Kodierung nicht-hydrologischer Zeitreihen angewendet werden können.

## **WaterML 2.0**

WaterML 2.0 umfasst mehrere OGC-Standards zur Modellierung und Kodierung hydrologischer Daten. Im Rahmen dieser Handlungsempfehlung ist insbesondere WaterML 2.0 Part 1 relevant. Darin wird, basierend auf dem Observations and Measurements-Standard, ein Datenmodell und ein XML-Encoding für hydrologische Zeitreihen spezifiziert.

# 1 Einleitung

## 1.1 Zusammenfassung

Die fortschreitende Digitalisierung unserer Welt betrifft auch die Bereitstellung von Daten wasserwirtschaftlicher Zeitreihen. Es gibt einen Trend, auch diese Daten webbasiert zugänglich zu machen und diese in Geo-Anwendungen einzubinden. Diese Entwicklung wurde vom Open Geospatial Consortium (OGC) aufgegriffen. In 2012 wurde für diesen Zweck die WaterML 2.0-Spezifikation zur Ergänzung der Sensor Observation Service-Spezifikation (SOS) als Datenaustauschstandard für hydrologische Messdaten veröffentlicht. Diese Standardisierung wird von diversen überregional tätigen Softwareentwicklern und hydrologischen Institutionen (z. B. WMO) unterstützt. Nicht zuletzt wird WaterML 2.0 als Standard zur Bereitstellung von Messdaten im Kontext der europäischen INSPIRE-Richtlinie empfohlen. Aufgrund dieser Entwicklung wird diesem neuen Standard eine gute Chance eingeräumt, sich sowohl als maßgebliche Web-Technologie als auch als domänenspezifisches Datenaustauschformat zu etablieren. Der Erfolg misst sich letztlich an der Verbreitung des Standards in der praktischen Anwendung.

Das vorliegende Dokument baut auf dem im Jahr 2016 in der Korrespondenz Wasserwirtschaft veröffentlichten Artikel „Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten – Eine Heranführung an die Sensor-Web-Technologien“ [8] auf. Es soll bei der Entscheidungsfindung, ob der OGC-Ansatz für die eigene Datenbereitstellung der geeignete Weg ist, unterstützen und bei einer hausinternen oder extern beauftragten technischen Umsetzung als Handlungsempfehlung helfen.

Die Handlungsempfehlung soll einen Wegweiser bieten, wie interoperable Standards, insbesondere die Sensor Web Enablement-Technologie (SWE) des Open Geospatial Consortiums (OGC) genutzt werden können um Messdaten als Bestandteil von Geodateninfrastrukturen bereitzustellen.



Zu diesem Zweck werden die zugrundeliegenden Spezifikationen kurz vorgestellt und erläutert. Darauf aufbauend werden grundlegende Funktionen des OGC Sensor Observation Service (SOS), einer Web-Dienst-Schnittstellenspezifikation zum Zugriff auf Messdaten, dargestellt. Zusätzlich werden typische Anforderungen an SOS-Server beschrieben sowie verschiedene Umsetzungsstrategien zur Integration in eine vorhandene IT-Infrastruktur diskutiert. Ergänzend dazu erfolgt im Anschluss daran die Betrachtung von Anforderungen an SOS-Client-Anwendungen (z. B. Web-Applikationen, Geoinformationssysteme (GIS)) und verfügbarer Technologien zur Implementierung solcher Anwendungen. Weiterhin erfolgt eine kurze Erläuterung des Themas der Zugriffskontrolle auf SOS-Server. Anschließend werden verschiedene Anwendungsbeispiele sowie Beispiele für verfügbare Software-Pakete vorgestellt. Abschließend erfolgt eine kurze Betrachtung von Möglichkeiten und Grenzen der Sensor Web-Technologie im Hinblick auf Datenmenge und Performance. Zusätzlich werden im Anhang dieses Dokuments Vorschläge und Beispiele zu Textblöcken bereitgestellt, welche für die Formulierung von Ausschreibungen und Anforderungen für Sensor Web-Infrastrukturen genutzt werden können. Weiterhin umfasst der Anhang ausgewählte XML-Beispiele, welche die Kodierung von Zeitreihen in verschiedenen Sensor Web-Repräsentationen illustrieren.

### 1.2 Einführung

Verantwortungsvolles Handeln braucht verlässliche Informationen, auf deren Grundlage Entscheidungen getroffen werden können. Der Komplexität gegenwärtiger Herausforderungen kann häufig nur noch mit einer globalen Herangehensweise angemessen begegnet werden. Ein einfacher (globaler) Informations- und Datenaustausch ist deswegen von großer Bedeutung für die Lösung vieler Probleme unserer Zeit. Insbesondere ein medienbruchfreier Austausch von Daten macht die Abstimmung hinsichtlich der Art und Weise, wie Informationen übermittelt werden, zwingend notwendig. Die rasante digitale Entwicklung hinterlässt uns eine kaum noch überschaubare Vielzahl an Datenaustauschformaten und -techniken. Auch der webbasierte Austausch von Messdaten wird zwischenzeitlich durch eine Vielzahl von verschiedenen Formaten und Techniken gekennzeichnet. In Zeiten einer fortschreitenden Digitalisierung und einer damit einher gehenden, technisch getriebenen, stetig wachsenden Datenflut hat sich dieses Problem für viele Nutzer von Daten verschärft. So ist die Erwartungshaltung vieler Entscheidungsträger in Zeiten von „Big Data“ und „Smart Data“ hoch. Doch ist den meisten Personen, die ein Endprodukt sehen, zumeist nicht der immense Aufwand, der im Datenmanagement zur Aufbereitung des Produktes betrieben werden muss, bewusst. So wird heute in der Regel bei der Projektierung von F&E-Projekten mit Datenauswertung ein maßgeblicher Anteil der Projektkosten der vorbereitenden Datenbereitstellung zugestanden. Substantieller Bestandteil dieser Vorarbeiten ist immer wieder die Homogenisierung von Daten durch inhaltliche Transformationen und Formatänderungen. Eine spezifische Folge des weiter wachsenden Portfolios an Datenaustauschformaten und -techniken ist die Notwendigkeit, Anwendungssoftware in der Hydrologie und Wasserwirtschaft stetig kostenintensiv weiter- bzw. neu zu entwickeln. Um diesem Problem zu begegnen, bietet sich der Einsatz von weithin akzeptierten Standards an. Durch die Verwendung dieser technischen Standards muss pro Datenquelle und pro Client-Anwendung jeweils (nur) einmal eine Anpassung an den Standard erfolgen: Datenquellen müssen ihre Daten entsprechend der Spezifikationen dieses Standards bereitstellen können und Client-Anwendung müssen diese Schnittstelle ansprechen und verstehen können. Sollen nun weitere Datenquellen oder Client-Anwendungen eingebunden werden, ergeben sich am bestehenden System keine Anpassungsbedarfe, da alle weiteren Komponenten über die abgestimmten Standards integriert werden können. Abbildung 2 macht dies deutlich: Verschiedene Datenbereitsteller stellen jeweils Server bereit, welche ihre Daten (diese liegen in individuellen Datenhaltungssystemen vor) über einen gemeinsamen Schnittstellenstandard zugreifbar machen. Gleichzeitig existiert eine Schicht von Client-Anwendungen, welche diese Daten über ein Netzwerk bzw. das WWW konsumiert und dabei den gemeinsamen Datenzugriffsstandard nutzt, sodass Daten unterschiedlicher Anbieter reibungslos geladen werden können.

## 1.3 Die Sensor Web-Technologie des OGC

Zum Umgang mit Mess- und Beobachtungsdaten wurde vom OGC, der globalen Institution zur Standardisierung im Bereich von Geodaten, das Konzept des Sensor Web für den Bereich der Sensordaten entwickelt. Die grundlegende Idee beim Sensor Web ist es, ähnlich wie bei anderen Standards der Geodateninfrastruktur, von Unterschieden in der zugrundeliegenden Infrastruktur zu abstrahieren und einheitliche (webbasierte) Schnittstellen und Datenformate anzubieten. Dies soll Interoperabilität in zwei verschiedenen Ausprägungen sicherstellen.

- Horizontale Interoperabilität: Systeme unterschiedlicher Hersteller können aufgrund gemeinsamer Schnittstellen, Datenmodelle und Datenformate reibungslos miteinander kommunizieren.
- Vertikale Interoperabilität: Software auf verschiedenen Schichten (z. B. Datenviewer und Datenquellen) kann im Sinne eines Plug-and-Play-Ansatzes mit einander interagieren ohne dass technische Anpassungen notwendig sind.

Das Ergebnis des Konzepts ist eine Familie von Standards, welche unter dem Begriff OGC Sensor Web Enablement (SWE) zusammengefasst werden (Abbildung 1). Einteilen lassen sich diese Standards in zwei Gruppen: Standards für Datenmodelle und -formate sowie Standards für (Web Service-) Schnittstellen, welche die Funktionalität des Sensor Web bereitstellen. Wichtig ist es darauf hinzuweisen, dass es sich bei diesen Standards um offene Spezifikationen handelt, welche von Software-Anbieter ohne Lizenzkosten oder sonstige Einschränkungen implementiert werden können. Hierbei gilt die folgende Unterscheidung im Hinblick auf die Terminologie:

- Spezifikation eines Standards: Definition von Datenformaten und/oder Operationen mit ihren Parametern
- Implementierung eines Standards: Software-Paket (Client oder Server), welches die Datenformate und/oder Schnittstellen eines Standards umsetzt.
- Installation einer Implementierung eines Standards: Ein Betreiber eines Servers kann ein Software-Paket, welches einen Standard implementiert, installieren, um Daten entsprechend der Spezifikationen des Standards zu veröffentlichen.

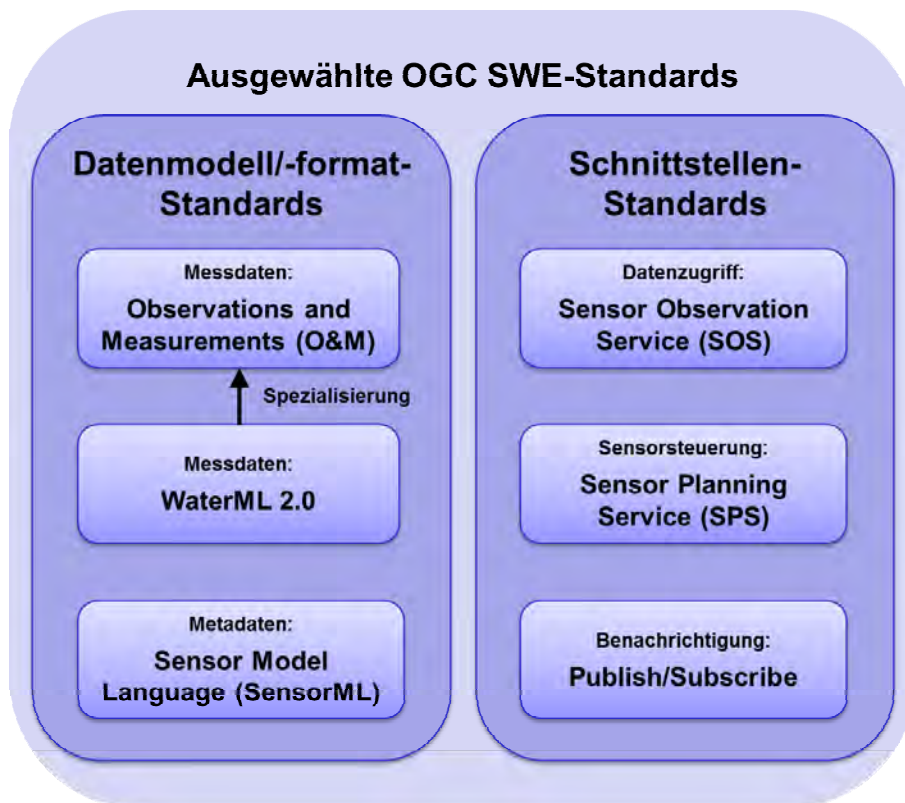


Abbildung 1: Überblick über ausgewählte OGC SWE-Standards

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Die SWE-Standards für Datenmodelle und -formate umfassen vor allem zwei wichtige Spezifikationen, welche einerseits die Modellierung und Kodierung der eigentlichen Messdaten beschreiben und andererseits ein Format für die zugehörigen Metadaten. Für die Messdaten besteht der Observations and Measurements-Standard (O&M). Dieser umfasst sowohl ein (UML-basiertes) Datenmodell (ISO-Standard) [18] und eine Spezifikation wie dieses Datenmodell auf XML-Dokumente abzubilden ist (OGC-Standard) [11]. Hinzuweisen ist hierbei weiterhin auf den OGC WaterML 2.0-Standard [22], welcher eine O&M-Spezialisierung für hydrologische Daten darstellt. Für die Metadaten zu Messdaten und Sensoren liefert der OGC Sensor Model Language-Standard (SensorML) [3] sowohl ein Datenmodell als auch eine Vorschrift zur XML-Kodierung.

Bei den SWE-Standards, welche die Funktionalität spezifizieren, d. h. primär Web Service-Schnittstellen, ist vor allem der OGC Sensor Observation Service-Standard (SOS) [7] zu nennen. Dieser Standard spezifiziert, wie Client-Anwendungen Messdaten anfragen und passende Abfragefilter an einen SOS-Server senden können, sodass dieser die entsprechenden Daten zurückliefert. Für die Inhalte dieser Antworten greift die SOS-Spezifikation auf die oben genannten SWE-Standards für Datenmodelle und -formate zurück. Die SOS-Schnittstelle ist auf die Bereitstellung punkt- und zeitbezogener Messdaten mit wiederkehrender Beobachtung zugeschnitten. Darüber hinaus existieren ein Standard zur Steuerung von Sensoren (OGC Sensor Planning Service (SPS) [21]) sowie Spezifikationen, um Messdaten bzw. Benachrichtigungen im Falle bestimmter Messwerte abonnieren zu können (OGC Publish/Subscribe Standard [4], [5]).

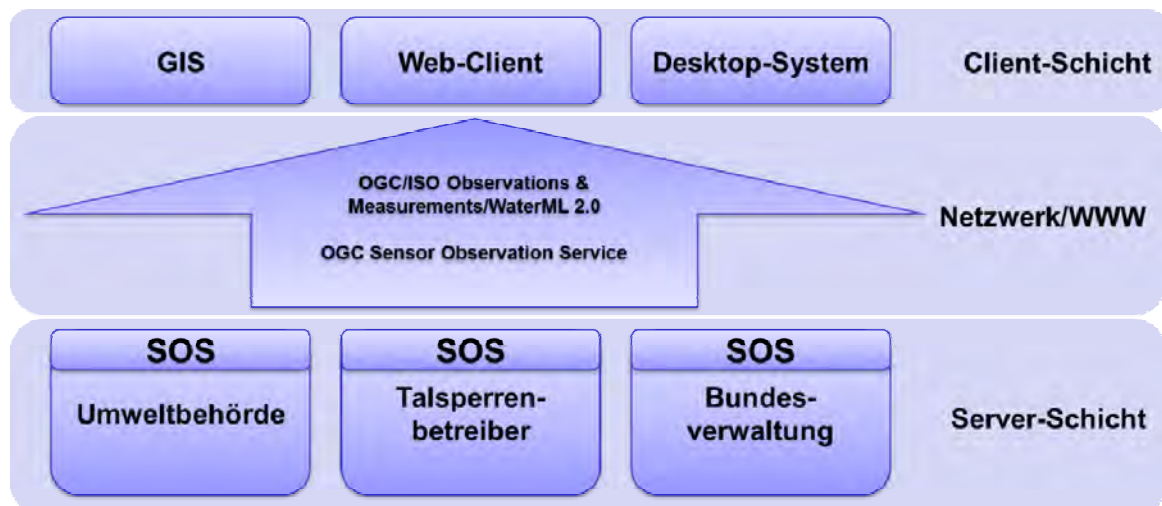


Abbildung 2: Beziehungen zwischen Client-Anwendungen und Datenquelle unter Nutzung der SOS-Standards

Die hier kurz beschriebene Sensor Web-Technologie birgt das reale Potential, hinsichtlich der zuvor skizzierten Probleme mittelfristig Abhilfe zu schaffen. Zum einen liefert der technische Ansatz offener, standardisierter, webbasierter Serverdienste direkt für jeden Interessenten eine maschinenlesbare Schnittstelle zur Entwicklung clientseitiger Anwendungssoftware. Zum anderen stellt WaterML 2.0 das erste international standardisierte Format für hydrologische Zeitreihen in diesem technischen Umfeld bereit. Aufgrund seiner hohen Relevanz konnten verschiedene nationale und internationale Organisationen diesen Standard in das Blickfeld und den Aktionsraum der WMO-Mitgliedsstaaten rücken. So erarbeitet die WMO Commission for Hydrology momentan eine Empfehlung für WaterML 2.0 als hydrologisches Datenaustauschformat<sup>1,2</sup>. Diese globale Aufmerksamkeit auf der einen Seite im Bereich der operativen hydrometeorologischen Dienste und auf der anderen Seite bei den führenden Softwareherstellern ist wegbereitend für eine Chance bisher ungekannter Dateninteroperabilität auf lokaler, regionaler und globaler Ebene.

1 [http://www.whycos.org/wordpress/?page\\_id=929](http://www.whycos.org/wordpress/?page_id=929)

2 [http://meetings.wmo.int/EC-69/English/2.%20PROVISIONAL%20REPORT%20\(Approved%20documents\)/EC-69-d05-2-WIS-approved\\_en.docx?Web=1](http://meetings.wmo.int/EC-69/English/2.%20PROVISIONAL%20REPORT%20(Approved%20documents)/EC-69-d05-2-WIS-approved_en.docx?Web=1)

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Eine besondere Rolle spielt hierbei auch die Aufnahme des OGC Sensor Observation Service-Standards in die INSPIRE Technical Guidance-Dokumente für Download Services [17]. Somit wird der SOS-Standard empfohlen um Messdaten in INSPIRE-konformer Weise zum Download anzubieten.

Ein weiterer wichtiger Aspekt ist die Idee der Wasserwirtschaft 4.0. Ausgehend vom Begriff der Industrie 4.0, also der Verknüpfung der industriellen Produktion mit modernen Informations- und Kommunikationstechniken, spielt hierbei die Vernetzung verschiedenster Elemente wie z. B. Kanal- und Leitungsnetzen, Pumpstationen, Sensorik usw. eine große Rolle. Die Erhebung und Verteilung von Echtzeit-Daten wie auch historischen Messungen wird in diesem Zusammenhang ein wichtiger Faktor sein. Berücksichtigt man die Vielfalt der Datenquellen und Datenkonsumenten in solchen Wasserwirtschaft 4.0-Systemen, ist die Nutzung einheitlicher Standards zur Sicherstellung der Interoperabilität unverzichtbar. Nur so lässt sich ein Plug-and-Play von Komponenten unterschiedlicher Hersteller aber auch der Datenaustausch zwischen Organisationen effizient ermöglichen. Die Sensor Web-Standards können hierfür einen wichtigen Baustein liefern.

Auch für Open Data-Aktivitäten, welche in Europa sowie auf Bundes- und Länderebene immer stärker vorangetrieben werden, spielen Datenzugriffsstandards eine große Rolle. Viele Aktivitäten setzen zunächst zwar auf die Veröffentlichung von Daten ohne spezifische technische Vorgaben, jedoch wird die Nutzbarkeit der Daten durch die Verwendung interoperabler Schnittstellen und Datenformate signifikant erhöht. Wenn Nutzer ihre Software nicht an spezifische Datenquellen anpassen müssen oder aufwendige Konvertierungsarbeiten vermeiden können, steigt der Wert der Daten für die Allgemeinheit. Auch hier können die in dieser Handlungsempfehlung beschriebenen Sensor Web-Standards eine wichtige Hilfe sein.

Im weiteren Text soll es darum gehen, die Möglichkeiten und Grenzen der Technologie zu beleuchten, um dem Interessenten eine praxisnahe Hilfe bei seiner Entscheidung über eine Investition in diese Technologie zu geben.

### 1.4 Anwendungsfelder der Technologie

Der englischsprachige Titel des SOS-Standards lässt vermuten, dass die Technik auf einen Sensor als Messfühler ausgelegt ist. In der Hydrologie bzw. wasserwirtschaftlichen Anwendung wäre dies typischer Weise ein Messfühler – beispielsweise zur Wassertemperaturmessung im Gelände oder zur Messung von Wasserständen an Pegeln im Gewässer oder zur Überwachung von wasserwirtschaftlichen Bauwerken – der mit einem Internetzugang ausgestattet ist und in regelmäßigen Intervallen oder zu ereignisbezogenen Anlässen Daten nahe Echtzeit an eine Daten weiterverarbeitende Instanz übermittelt. Die Anwendungsszenarien sind so vielfältig wie die punkt- und zeitbezogenen operativen Messungen dieser fachlichen Domäne.

Interessanter Weise wächst die Anzahl möglicher Anwendungsfälle dieser Technologie noch um ein Vielfaches, wenn man den Schwerpunkt der Betrachtungen nicht allein auf den nahe Echtzeit wirkenden Sensor reduziert sondern den Aspekt der standardisierten, webbasierten Datenbereitstellung in den Fokus der Betrachtungen rückt. In diesem Fall kann der Sensor durch eine beliebig geartete Datenquelle ersetzt werden. Typischer Weise wäre dies eine datei- oder datenbankgestützte Datenhaltung, die wiederum nicht nur ungeprüfte nahe Echtzeitdaten bereitstellt sondern sehr wohl auch geprüfte und/oder historische Datenreihen vorhalten kann. Ebenso ist, wie in [1] beschrieben, sogar die Bereitstellung von Vorhersagedaten als Ergänzung zu beobachteten/gemessenen Daten möglich.

Will man die Technologie einsetzen, muss man als Datenbereiter zumindest einen entsprechenden SOS-Server einrichten. Will man hingegen die eigenen oder von anderen mit einem SOS-Server bereitgestellten Daten nutzen, benötigt man eine Anwendungs-/Clientsoftware, die diesen OGC-Standard unterstützt (vgl. Abbildung 2). Grundsätzlich lassen sich zwei Nutzungsszenarien unterscheiden. Das eine Szenario lässt sich unter dem Aspekt Maschine-Maschine-Kommunikation beleuchten. Hierunter soll die Nutzung des Datenstroms eines SOS-Servers zur unmittelbaren maschinellen Weiterverarbeitung verstanden werden. Typische Anwendungen wären die Speicherung

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

von Sensordaten in einem spezifischen Datenmodell für eine spätere Nutzung oder aber die unmittelbare Verarbeitung von Sensordaten in numerischen Modellen zur Berechnung abhängiger Parameter. Ein anderes Szenario lässt sich unter dem Aspekt der Mensch-Maschine-Kommunikation betrachten. Da wir uns in der Welt der Zeitreihen bewegen, stehen als Anwendung hier zumeist Vierwer/Betrachter für Zeitreihen im Fokus. Aber auch die Ableitung und Darstellung statistischer Angaben aus den Daten der Zeitreihen (z. B. Hauptwerte von Pegeln) sind typische Nutzungen. Im Kontext der Mensch-Maschine-Interaktion gewinnt der Faktor Zeit eine andere Bedeutung, geht es doch letztlich um die Performanz einer Clientsoftware. Zwischenzeitig gibt es sowohl kommerzielle als auch freie Desktop- (z. B. GIS-Software) und Web-Software zur Darstellung von SOS-Zeitreihen.

Zusammenfassend lassen sich folgende technische Funktionen der Sensor Web-Technologie identifizieren:

- Suche nach Beobachtungs- und Messdaten sowie nach Sensoren, Sensorsystemen und Messprozessen, welche diese Daten generiert haben
- Bereitstellung von Metadaten zu Beobachtungs- und Messdaten, z. B. Informationen über die zugrundeliegenden Messprozesse, die Datenqualität, die Kalibrierung von Sensoren oder zuständige Ansprechpartner
- Bereitstellung bzw. Abruf von Beobachtungs- und Messdaten über Web Service-Schnittstellen, welche präzise definierte Abfragefilter anbieten
- Abonnieren/Publizieren von Benachrichtigungen anhand von Messwerten (z. B. Vergleich aktueller Messwerte mit nutzerdefinierten Grenzwerten)
- Steuerung von Sensoren um bestimmte Messungen vorzunehmen

Innerhalb von Körperschaften die schon SOS-basierte Lösungen einsetzen (z. B. über bereits bestehende wasserwirtschaftliche Software-Pakete), können Synergien durch den Einsatz identischer Techniken gewonnen werden. Werden z. B. bereits einzelne Datensätze über SOS/WaterML 2.0-konforme Systeme bereitgestellt, so ist es sinnvoll im Zuge der Weiterentwicklung auch andere Systeme damit auszurüsten. Dies vereinfacht die Integration bislang getrennter Datenquellen und erlaubt somit potentiell die schnelle Ableitung bisher nicht direkt verfügbarer Informationen.

Tabelle 1 zeigt Beispiele möglicher Einsatzfelder. Für ausgewählte Einsatzszenarien werden detailliertere Beschreibungen von Umsetzungsbeispielen referenziert. Dort werden weitere Informationen über Motivation, Art der Umsetzung und Erfahrungen beschrieben.

**Tabelle 1: Beispielhafte Einsatzszenarien der Sensor Web-Technologie**

Anwendungsfall	Mögliche Daten-Bereitsteller	Mögliche Daten-Nutzer	Datenart	Stand der Implementierung 2017	Referenz
Hochwassermonitoring	Landesinstitutionen, benachbarter Wasserverband	Wasser- verband, Öffentlich- keit	Wasser- stände, Ab- flüsse, Wet- terdaten	Es gibt Beispiele in Produktivbetrieb	4.1.4
Grundwassermonitoring in Bergsenkungsgebieten	Wasserverband	Kommunen, Verbands- mitglieder	(Grund-) Wasser- stände, Wetterdaten	Es gibt Beispiele in Produktivbetrieb	4.1.5
Ökologisches Monitoring von Feuchtgebieten	Trinkwasser- versorger	Aufsichts- behörde Naturschutz	Grund- wasser- förder- mengen,	Nicht bekannt	

Anwendungsfall	Mögliche Daten-Bereitsteller	Mögliche Daten-Nutzer	Datenart	Stand der Implementierung 2017	Referenz
			(Grund-) Wasserstände		
Gewässergüte Monitoring	Betreiber	Aufsichtsbehörde Naturschutz	Gewässeranalytik	Erste Praxisbeispiele vorhanden	4.1.2
Gewässer-ökologisches Monitoring (WRRL)	Landesumwelt-amt	Benachbartes Landesumweltamt	Biologische Arten-erhebungen	Höhere Komplexität, Anwendungen noch in Entwicklung	

### 1.5 Technische Herausforderungen in der Umsetzung der Sensor Web-Technologie

Praktische Experimente mit der Sensor Web-Technologie zeigen, dass der OGC Sensor Observation Service-Standard geeignet ist lange Zeitreihen (z. B. 191 Jahre an Tagesmittelwerten, entsprechend ca. 70.000 Messwerten) in guter Geschwindigkeit auszuliefern. Auch die Darstellung großer Zeitreihen in Web-Clients ist gut möglich, wobei hier die im Browser zu verarbeitende Datenmenge zu beachten ist. Eine Anzeige von ca. 10.000 Werten stellt aber im Allgemeinen kein Problem dar. Mehr Details zu diesen Untersuchungen werden in Kapitel 5 beschrieben.

Die Anzahl der Betroffenen, die Messdaten aus der Hydrologie und Wasserwirtschaft bereitstellen möchten oder müssen ist relativ klein. Aufgrund der häufig individuell aufgebauten Systemarchitekturen der Datenbereitsteller, wird für den Aufbau von Sensor Web-Infrastrukturen oft nicht auf Standard-Software-Pakete zurückgegriffen sondern auf bestehende Software-Komponenten, die entsprechend der jeweiligen Anforderungen maßgeschneidert zusammengesetzt werden. Individuelle Implementierungen werden spätestens sicher dann notwendig, wenn die originäre Datenquelle eine Datenhaltung darstellt, die unabhängig von der zukünftig einzusetzenden Fachsoftware entstanden ist. In bestimmten Szenarien werden hier Softwareentwicklungen bzw. -anpassungen notwendig werden, die entweder mit den eigenen Kapazitäten im Hause oder mit Unterstützung externer Dienstleister zu realisieren sind. In beiden Fällen ist es hilfreich, die funktionalen und nicht-funktionalen Aspekte der zukünftigen Softwareanwendung festzuhalten und diese den Softwareentwicklern als erste Leistungsbeschreibung bereitzustellen. Als Lastenheft bildet es in unserem Anwendungsfall die Grundlage für die Erstellung eines spezifizierten Pflichtenhefts zur Umsetzung eines Sensor Web-Services, ggf. ergänzt durch einen Sensor Web-Client.

In dem Prozess der Erstellung eines Lastenheftes zur Umsetzung eines Sensor Web-Systems sind bereits diverse technische Weichenstellungen vorzunehmen. Um hier keine Fehlentwicklungen einzuleiten, ist ein gewisses Grundverständnis hinsichtlich der zum Einsatz kommenden Techniken notwendig. Diese werden im nachfolgenden Kapitel eingeführt.

## 2 Technologische Grundlagen

Im weiteren Verlauf soll diese Handlungsempfehlung einen Wegweiser bieten, wie interoperable Standards, insbesondere die Sensor Web Enablement-Technologie (SWE) des Open Geospatial Consortiums (OGC) genutzt werden können um Messdaten als Bestandteil von Geodateninfrastrukturen bereitzustellen.

Zu diesem Zweck werden die zugrundeliegenden Spezifikationen kurz vorgestellt und erläutert. Darauf aufbauend werden grundlegende Funktionen des OGC Sensor Observation Service (SOS), einer Web-Dienst-Schnittstellenspezifikation zum Zugriff auf Messdaten, dargestellt. Zusätzlich werden typische Anforderungen an SOS-Server beschrieben sowie verschiedene Umsetzungsstrategien zur Integration in eine vorhandene IT-Infrastruktur diskutiert. Ergänzend dazu erfolgt im Anschluss daran die Betrachtung von Anforderungen an SOS-Client-Anwendungen und verfügbarer Technologien zur Implementierung solcher Anwendungen. Abschließend erfolgt eine kurze Erläuterung des Themas der Zugriffskontrolle auf SOS-Server.

Nicht explizit vorgestellt werden die beiden Standards OGC SWE Common Data Model [20] und OGC SWE Service Model [13], da diese eher indirekt von Bedeutung sind. Während das OGC SWE Common Data Model grundlegende Datentypen spezifiziert, welche von allen weiteren SWE-Standards gemeinsam wiederverwendet werden, beschreibt das OGC SWE Service Model Grundsätze, welche von allen SWE-Schnittstellen-Standards befolgt werden (z. B. gemeinsamer Ansatz um Sensor-Metadaten über die DescribeSensor-Operation zugänglich zu machen).

### 2.1 OGC Sensor Observation Service

Der OGC Sensor Observation Service-Standard (SOS) ist eine Schnittstellenspezifikation für Web-Dienste zum Zugriff auf Messdaten. Es wird also beispielweise festgelegt, welche Operationen zum Messdatenabruf genutzt werden können und welche Parameter diese Operationen bieten. Auf Basis des SOS 1.0.0-Standards [19] wurde in den nachfolgenden Jahren anhand praktischer Anwendungserfahrungen eine optimierte Version 2.0 der SOS-Schnittstelle veröffentlicht [7].

Wichtige SOS-Operationen, welche die Grundfunktionalität des Dienstes bereitstellen sind:

- **GetCapabilities:** Download einer Selbstbeschreibung einer SOS-Instanz mit Metadaten zu unterstützten Operationen und verfügbaren Daten.
- **GetObservation:** Download von Beobachtungsdaten, die einem beim Operationsaufruf übergebenem Filter entsprechen.
- **DescribeSensor:** Download von Metadaten zu Beobachtungsprozessen.

Eine Übersicht, welche Operationen für die Hydrologie benötigt werden, befindet sich in Abschnitt 3.1.2.

Wichtig ist dabei jedoch festzuhalten, dass der SOS-Standard nur die Schnittstelle zum Abrufen von Messdaten festlegt. Die von einem SOS-Server zurück gelieferten Datenformate werden dagegen durch den O&M-Standard bzw. seine Spezialisierung in Forms des WaterML 2.0-Standards (für die eigentlichen hydrologischen Messdaten) und die Sensor Model Language/WaterML 2.0 (für die zugehörigen Metadaten zu Messprozesse) spezifiziert.



## 2.2 OGC SOS 2.0 Hydrology Profile

Um eine engere Festlegung zu treffen, wie der SOS-Standard innerhalb der Hydrologie zu nutzen ist und somit die Interoperabilität zwischen Datenanbietern zu erhöhen, wurde das SOS 2.0 Hydrology Profile entwickelt. Neben der Festlegung auf die SOS 2.0-Spezifikation als Schnittstelle und WaterML 2.0 als Datenformat, werden weitere Festlegungen getroffen, die sich aus praktischen Erfahrungen im Bereich der Hydrologie ergeben haben. Dies umfasst beispielsweise den Umgang mit (zu) umfangreichen Abfragen, die eindeutige Festlegung welche Begriffe aus der SOS-Spezifikation wie zu verwenden sind (z. B. wird festgelegt, dass sich der Begriff Sensor auf einen Sensortyp und nicht auf eine Sensorinstanz bezieht) und eine Anforderung, welche Operationen aus dem SOS-Standard mindestens zu unterstützen sind. Zusätzlich wird eine weitere Operation (GetDataAvailability) spezifiziert, mit der Informationen über die durch einen SOS-Server angebotenen Daten einfacher abgefragt werden können.

Das SOS 2.0 Hydrology-Profil wurde im Jahr 2015 als OGC Best Practice Paper angenommen [2] und besitzt somit den Charakter einer offiziellen Empfehlung des OGC.

## 2.3 ISO/OGC Observations and Measurements

Durch den Observations and Measurements (O&M)-Standard werden ein konzeptionelles Modell für Messdaten sowie eine zugehörige Encoding-Spezifikation, welche vorschreibt, wie Messdaten in XML kodiert werden, vorgegeben. So wird beispielsweise festgelegt, welche Attribute zu einer Messung angegeben werden müssen (z. B. verschiedene Arten von Zeitstempeln, Angaben zu Messprozessen, Maßeinheit, Georeferenzen). Abbildung 3 stellt das grundlegende Datenmodell des O&M-Standards grafisch dar. Wichtige Elemente sind hierbei:

- Observed Property: Der Parameter, welcher gemessen wird (z. B. Wasserstand).
- Feature of Interest: Das Geoobjekt, auf welches sich die Messung bezieht (z. B. Pegelstandort).
- Procedure: Der Messprozess, über welchen der Messwert generiert wurde (z. B. Sensor, Berechnungsmodell).
- Result: Der gemessene Wert mit zugehöriger Maßeinheit (Unit of Measurement)
- Phenomenon Time: Der Zeitpunkt, auf den sich der gemessene Wert bezieht.

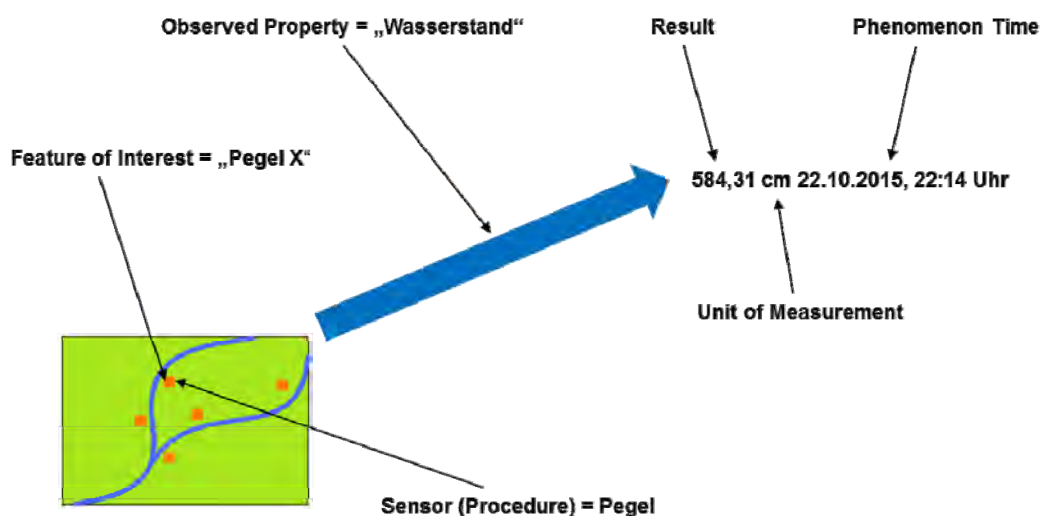


Abbildung 3: Illustration der Grundlagen des O&M-Modells

Der ursprüngliche O&M 1.0-Standard [9], [10] wurde auf Basis praktischer Erfahrungen weiterentwickelt, sodass das überarbeitete konzeptionelle O&M-Modell 2.0 als ISO-Standard angenommen wurde [18]. Das zugehörige O&M 2.0 XML-Encoding ist zudem als OGC-Standard veröffentlicht worden [11].

## 2.4 OGC WaterML 2.0

WaterML 2.0 ist eine Familie verschiedener OGC-Standards, welche die Modellierung und Kodierung verschiedener Arten hydrologischer Daten spezifizieren. Dies umfasst insbesondere folgende Spezifikationen (siehe auch Abbildung 4):

- WaterML 2.0 Part 1 – Timeseries [22]: Spezifikation zur Modellierung und Kodierung von hydrologischen Zeitreihen.
- WaterML 2.0 Part 2 – Ratings, Gaugings and Sections [23]: Spezifikation für Daten zur Wasserstands-Abfluss-Beziehung und Flussquerschnitten
- WaterML 2.0 Part 3 – Surface Hydrology Features (HY\_Features): Spezifikation eines Modells für Geobjekte im Bereich von Oberflächengewässern (Abstimmung zur Annahme des Standards wurde im Oktober 2017 erfolgreich abgeschlossen)
- WaterML 2.0 Part 4 – GroundWaterML 2 (GWML2) [6]: Spezifikation zur Modellierung und Kodierung von Grundwasser-Daten; für Messdaten wird ein Mapping auf den O&M-Standard beschrieben.

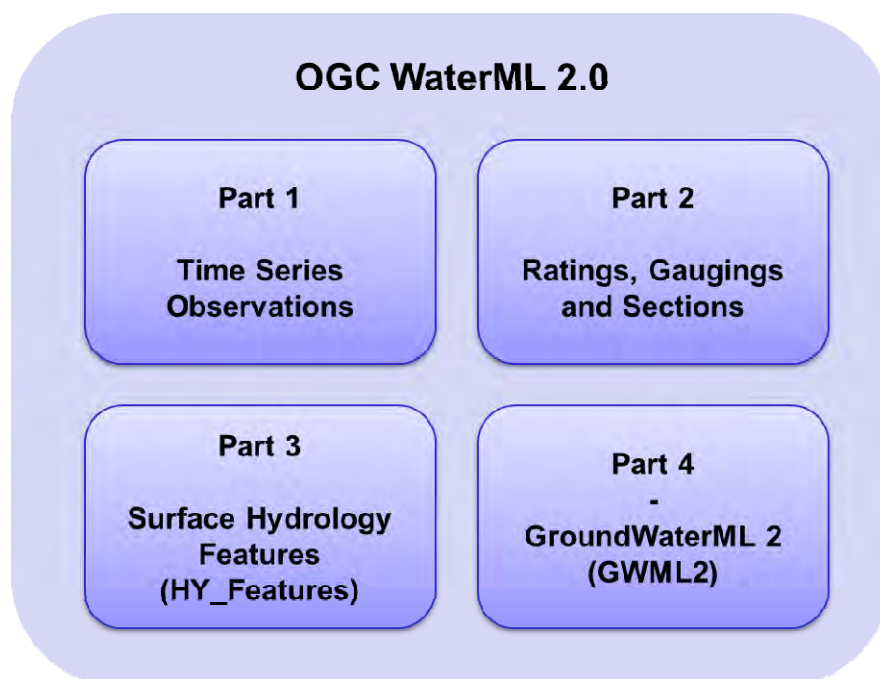


Abbildung 4: Familie der OGC WaterML 2.0-Standards

Da es in dieser Handlungsempfehlung um die Bereitstellung von Messdaten geht, ist im Folgenden WaterML 2.0 Part 1 im Fokus (es ist jedoch sinnvoll, die anderen WaterML 2.0-Teile zu berücksichtigen, wenn weitere Geodaten bereitgestellt werden sollen, auf die sich die Messdaten beziehen). WaterML 2.0 Part 1 basiert auf dem Observations and Measurements-Standard und bietet ein Datenmodell und ein XML-Encoding, welche für die Bedürfnisse hydrologischer Anwendungen optimiert wurden (z. B. explizite Felder um Metadaten zu Zeitreihen anzugeben, kompaktere Kodierung von Zeitreihen). WaterML 2.0 Part 1 (nachfolgend der Einfachheit halber als „WaterML 2.0“ bezeichnet) wurde auf internationaler Ebene durch Organisationen aus Europa, Nordamerika und Australien vorangetrieben und erfährt eine kontinuierlich steigende Akzeptanz. So wurde aus WaterML 2.0 der neue Standard TimeseriesML abgeleitet, welcher in weiteren Anwendungsbereichen über die Hydrologie hinaus genutzt werden kann.

Grundlage der Modellierung von Zeitreihen in WaterML 2.0 ist die sogenannte TimeSeriesObservation. Diese bietet die Möglichkeit eine Zeitreihe einschließlich ihrer Metadaten (z. B. Raumbezug, beobachtete Parameter, abgedeckte Zeitspanne) zu kodieren. Zur Kodierung der eigentlichen Messdaten und des zugehörigen Zeitbezugs bietet WaterML 2.0 zwei Untertypen der TimeSeriesObservation welche in Tabelle 2 erläutert sind.

Tabelle 2: WaterML 2.0 Observation-Typen

WaterML 2.0 Observation-Typ	Erläuterung
TimeseriesTVPObservation (Timeseries Time-Value-Pair Observation)	Darstellung von Messdaten als eine Liste von Paaren aus Zeitstempel und Messwert; für hydrologische Anwendungen der gebräuchlichere WaterML 2.0 Observation-Typ.
TimeseriesDomainRangeObservation	Darstellung von Messdaten als eine Liste von Zeitstempeln und eine Liste von Messwerten; angelehnt an die Handhabung von rasterbasierten Daten.

Hinweis: Bei WaterML 2.0 ist immer die Angabe der Version 2.0 wichtig, da WaterML 1.0 kein OGC-Standard ist und auch nicht für die Nutzung in Sensor Web-Infrastrukturen entwickelt wurde.

## 2.5 WaterML-WQ

Bei WaterML-WQ handelt es sich um ein Profil der OGC O&M und WaterML 2.0-Standards. Hierin wird spezifiziert, wie Messungen zur Wasserqualität anhand von OGC O&M und WaterML 2.0 kodiert werden können. Da ein großer Teil der notwendigen Grundlagen bereits in den O&M und WaterML 2.0-Standards spezifiziert ist, handelt es sich bei WaterML-WQ um eine relativ kompakte Spezifikation. Wichtige Elemente sind einerseits die Auswahl passender Datentypen aus O&M bzw. WaterML 2.0 um Einzelmessungen oder Zeitreihen zu kodieren. Weiterhin werden Empfehlungen zur Referenzierung von Maßeinheiten zu gemessenen Werten gemacht.

Die WaterML-WQ-Spezifikation hat den Status eines OGC Best Practice Papers [12]. Somit handelt es sich zwar nicht um einen bindenden Standard, aber um eine offiziell abgestimmte Empfehlung des OGC.

## 2.6 OGC TimeseriesML

Aufgrund der hohen Akzeptanz des WaterML 2.0-Standards zur Bereitstellung von Zeitreihen und der Attraktivität dieser Spezifikation für andere Fachgebiete (z. B. Luftqualitätsmessungen) bestand der Wunsch, die Ideen aus WaterML 2.0 in Domänen-unabhängiger Weise zu verallgemeinern. Das Resultat ist der OGC TimeseriesML-Standard, welcher im September 2016 veröffentlicht wurde. Durch TimeseriesML wird beschrieben, wie die Modellierung und XML-kodierung von Zeitreihen interoperabel erfolgen kann. Dieser Standard basiert auf den grundlegenden Elementen von WaterML 2.0. Die Hydrologie-spezifischen Charakteristika von WaterML 2.0 wurden dabei jedoch entfernt. TimeseriesML besteht aus zwei verknüpften Spezifikationen: Das sogenannte „Timeseries Profile of Observations and Measurements“ [24] beschreibt das konzeptionelle Modell von TimeseriesML als Spezialisierung des abstrakten O&M-Modells. Hierbei erfolgt insbesondere eine Fokussierung auf die Bestandteile der O&M-Spezifikation, welche für die Modellierung und Kodierung von Zeitreihen benötigt werden. Ergänzend dazu, liefert die Spezifikation „TimeseriesML 1.0 – XML Encoding of the Timeseries Profile of Observations and Measurements“ [25] die zugehörigen Vorschriften, wie die entsprechenden Zeitreihendaten in Form von XML-Dokumenten kodiert werden können.

Für hydrologische Anwendungen ist WaterML 2.0 die Spezifikation der Wahl. Dennoch ist es sinnvoll, sich der engen Verwandtschaft zwischen WaterML 2.0 und TimeseriesML bewusst zu sein.

### 2.7 OGC Sensor Model Language

Zur Bereitstellung von Metadaten zu den Messprozessen bzw. Sensoren mit denen Messdaten erzeugt wurden, wurde durch das OGC die Sensor Model Language (SensorML) spezifiziert. Dieser Standard liegt aktuell in der Version 2.0 vor [3]. SensorML bietet die Möglichkeit Metadaten wie z. B. Sensortypen, Kalibrierungsinformationen, Dokumentationsquellen, Kontaktdaten von Sensorbetreibern, etc.) zu modellieren und in XML zu kodieren.

Für hydrologische Anwendungen wird SensorML jedoch eher selten genutzt. Hier werden eher spezifische Metadaten-Kodierungen aus dem WaterML 2.0-Standard verwendet.

### 2.8 INSPIRE

Die europäische INSPIRE-Richtlinie, welche in Deutschland durch das Geodatenzugangsgesetz [14] umgesetzt wurde, dient dem Aufbau einer interoperablen europäischen Geodateninfrastruktur. Neben Verpflichtungen zur Bereitstellung bestimmter Daten durch öffentliche Stellen umfasst INSPIRE auch umfassende technische Regelungen und Empfehlungen, in welcher Form die Bereitstellung von raumbezogenen Daten erfolgen soll bzw. kann.

Zur Unterstützung der EU-Mitgliedsstaaten bei der Umsetzung der INSPIRE-Richtlinie, existieren neben den rechtlich verbindlichen Richtlinien und Verordnungen auch sogenannte Technical Guidance-Dokumente. Diese Dokumente haben keinen normativen sondern einen informativen Charakter. D. h. die dort empfohlenen Lösungsansätze wurden im Rahmen eines formalen Prozesses verschiedener INSPIRE-Akteure für sinnvoll und geeignet befunden, andere Umsetzungen in den Mitgliedsstaaten sind jedoch auch möglich. Die Technical Guidance-Dokumente bieten eine gute technische Erfahrungsgrundlage, sodass eine Verwendung der INSPIRE Technical Guidance-Dokumente empfohlen wird.

In diesem Zusammenhang ist es wichtig darauf hinzuweisen, dass die INSPIRE-Anforderungen in einzelnen Punkten über die jeweils entsprechenden OGC-Standards hinausgehen. Typische Beispiele hierfür die Unterstützung von Mehrsprachigkeit (z. B. Angabe welche Ausgabesprachen von einem Dienst unterstützt werden) sowie die Abfrage von Daten in mehreren Koordinatensystemen. Die Erfüllung dieser weitergehenden Anforderungen wird ebenfalls in den INSPIRE Technical Guidance Dokumenten beschrieben.

#### 2.8.1 Nutzung von OGC SOS-Servern als INSPIRE Download Service

Im Rahmen der INSPIRE-Richtlinie wird eine Reihe verschiedener Diensttypen beschrieben, welche durch EU-Mitgliedsstaaten bereitzustellen sind. Neben Funktionalitäten wie der Suche nach Daten und der kartographischen Repräsentation von Geodaten, spielt der Zugriff auf Geodaten, d. h. der Daten-Download, eine sehr große Rolle. Zu diesem Zweck sind im Rahmen von INSPIRE sogenannte Download Services spezifiziert.

Auch für INSPIRE Download Services bestehen Technical Guidance-Dokumente. Neben Diensten zum Download vektor- oder rasterbasierter Geodaten, wird auch die Nutzung von OGC SOS-Servern als Download Service für Messdaten empfohlen. Hierzu existiert ein eigenständiges Technical Guidance-Dokument, welches im Jahr 2016 veröffentlicht wurde [17]. In diesem Dokument wird einerseits beschrieben, welche in der INSPIRE Network Services-Verordnung geforderten Funktionalitäten über welche SOS-Operationen wie umgesetzt werden können (Tabelle 3). Andererseits werden notwendige Erweiterungen des SOS-spezifiziert, um Mehrsprachigkeit und multiple Raumbezugssysteme zu unterstützen (diese Erweiterungen sind vergleichbar mit den Erweiterungen, die für INSPIRE-konforme Web Feature Services und Web Coverage Services spezifiziert wurden).

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Möchte ein Datenanbieter Messdaten in INSPIRE-konformer Weise zum Download anbieten, ist der SOS-Standard somit die empfohlene Lösung.

**Tabelle 3: Umsetzung der verschiedenen Operationen eines INSPIRE Download Service durch entsprechende SOS-Funktionalität**

Operation aus der EU-Verordnung zu Netzdiensten (Network Services)	Entsprechende SOS-Operation	Erläuterung
Get Download Service Metadata	GetCapabilities	Zugriff auf die Metadaten (z. B. unterstützte Operationen, verfügbare Daten) einer SOS-Instanz
Get Spatial Data Set	GetObservation	Download kompletter Datensätze von Messdaten; über den sogenannten „Offering“-Abfrageparameter wird einem SOS-Server der Identifier des angefragten Datensatzes übergeben.
Describe Spatial Dataset	GetCapabilities	Die über einen SOS-Server angebotenen Datensätze werden in den über die GetCapabilities-Operation abfragebaren Metadaten beschrieben.
Link Download Service	-	Nicht über SOS-Operation umzusetzen; erfolgt durch den Upload von Metadaten einer SOS-Instanz in einen Katalog.
Get Spatial Object	GetObservation	Download von Teilmengen der Datensätze, die durch eine SOS-Instanz angeboten werden. Die Einschränkung auf Teilmengen erfolgt über die im SOS-Standard spezifizierten Abfrageparameter der SOS-Operation „GetObservation“
Describe Spatial Object Type	GetCapabilities	Die über einen SOS-Server angebotenen Typen von Messdaten werden in den über die GetCapabilities-Operation abfragebaren Metadaten beschrieben.

### 2.8.2 O&M und WaterML 2.0 als INSPIRE Datenformat

Neben den INSPIRE-Empfehlungen zum SOS-Standard als Schnittstelle zum Download von Messdaten, besteht ein weiteres Technical Guidance-Dokument, welches die Nutzung des ISO/OGC O&M 2.0-Standards zum Download von Messdaten empfiehlt [16]. In diesem Dokument werden insbesondere spezifische INSPIRE Observation-Typen definiert und Hinweise gegeben, in welchen Situationen welcher Typ geeignet ist. Ausgewählte INSPIRE Observation-Typen mit Relevanz für die Hydrologie werden in Tabelle 4 vorgestellt.

Tabelle 4: Ausgewählte INSPIRE Observation-Typen

INSPIRE Observation Typ	Erläuterung
pointObservation	Messung eines Einzelwertes eines Parameters an einem Ort.
pointTimeSeriesObservation	Messung einer Zeitreihe eines Parameters an einem Ort. Dieser Observation-Typ ist eng an WaterML 2.0 angelehnt.
multiPointObservation	Messung desselben Parameters zum selben Zeitpunkt an verschiedenen Orten.
profileObservation	Messung desselben Parameters zu einem Zeitpunkt an verschiedenen Positionen entlang eines vertikalen Profils.
specimenObservation	Messung eines Einzelwertes eines Parameters an einem Ort durch Probenahme. Im Vergleich zur pointObservation sind in diesem Fall zwei Zeitstempel notwendig: Zeitpunkt der Probenahme und Zeitpunkt des Vorliegens des Analyseergebnisses.
specimenTimeSeriesObservation	Messung einer Zeitreihe eines Parameters an einem Ort durch Probenahme. Im Vergleich zur pointObservation sind in diesem Fall zwei Zeitstempel notwendig: Zeitpunkt der Probenahme und Zeitpunkt des Vorliegens des Analyseergebnisses.

## 2.9 Zusätzliche Funktionen über den SOS-Standard hinaus

Die zuvor vorgestellten Standards spezifizieren vor allem die Modellierung und Kodierung von Messdaten bzw. entsprechenden Metadaten sowie den Zugriff auf diese Daten. Für die Zukunft zeichnen sich jedoch Entwicklungen ab, welche den funktionalen Umfang deutlich erweitern. Hierzu existieren zusätzliche Sensor Web-Standards, welche zurzeit teilweise im Rahmen erster Implementierungen evaluiert werden. Dies umfasst einerseits die Push-basierte Auslieferung von Daten (z. B. Benachrichtigungen, wenn neue Daten vorliegen oder die gemessenen Daten bestimmte Grenzwerte überschreiten). Andererseits besteht auch die Möglichkeit Messprozesse und Sensoren aktiv zu steuern.

### 2.9.1 Zustellung von Benachrichtigungen/Abonnieren von Datenströmen

In einigen Fällen ist es sinnvoll, neue Messdaten sofort in dem Augenblick an Nutzer auszuliefern, in dem sie verfügbar werden. Beispielsweise können Überwachungsanwendungen mit einem Datenstrom der neuesten Messwerte versorgt werden, sodass ggf. kritische Messwerte sofort erkannt werden können. Das klassische Pull-basierte Kommunikationsmodell des SOS-Standards bietet eine solche Funktion nicht, d. h. Nutzer müssen einen SOS-Server regelmäßig aktiv nach neuen Daten abfragen.

Um auch eine Push-basierte Kommunikation, d. h. Daten beim Vorliegen aktiv an Abonnenten auszuliefern, wurde durch das OGC der sogenannte Publish/Subscribe-Standard entwickelt. Dieser Standard umfasst einerseits eine abstrakte Spezifikation der Funktionalität [4] und andererseits eine Spezifikation, wie die Funktionalität über SOAP und XML umgesetzt werden kann [5]. Eine Spezifikation der Umsetzung über REST und JSON ist momentan noch im Diskussionsprozess.

Die OGC Publish/Subscribe-Spezifikationen sind kein Bestandteil des OGC SWE-Frameworks. Stattdessen handelt es sich hierbei um einen übergreifenden Standard, welcher auch auf andere Typen von OGC Diensten anwendbar ist. So könnte ein OGC Web Map Server über die Publish/Subscribe-

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Spezifikation so erweitert werden, dass Nutzer automatisch mit neuen Karten versorgt werden, sobald sich Aktualisierungen ergeben.

Das Funktionsprinzip der Publish/Subscribe-Spezifikation basiert auf einem Abonnement-basierten Kommunikationsmuster. Ein Server (z. B. SOS) stellt Client-Anwendungen zunächst Informationen bereit, welche Datenströme verfügbar sind. Eine Client-Anwendung kann dann entsprechende Abonnement-Anfragen an den Server stellen. Der Server prüft dann für alle eingehenden Daten, ob sie zu einem zuvor registrierten Abonnement passen. Sofern dies der Fall ist, erfolgt automatisch eine Benachrichtigung bzw. Auslieferung der Daten an die zu den jeweiligen Abonnements gespeicherten Empfänger.

Zurzeit erfolgen Umsetzungen und Evaluierungen des Publish/Subscribe-Standards in verschiedenen Kontexten (vgl. z. B. 4.1.1). Weiterhin erfolgen Arbeiten zur Konzeption und Implementierung einer REST/JSON-Erweiterung des Publish/Subscribe-Standards. Aus diesem Grunde ist die Verbreitung des Publish/Subscribe-Standards nicht mit der der SOS- und WaterML 2.0-Spezifikationen vergleichbar. Dennoch ist es sinnvoll bei der Entwicklung von Benachrichtigungssystemen diese Standards zu berücksichtigen, da in der Zukunft von einer größeren Verbreitung auszugehen ist.

Eine weitere Anwendung, welche entsprechende Benachrichtigungsfunktionen bietet, ist der ArcGIS GeoEvent Server von Esri, welcher in Abschnitt 4.2.2.6 vorgestellt wird.

### 1.1.1 Steuerung von Sensoren

Zur Steuerung von Sensoren bzw. Messprozessen, wurde innerhalb der OGC Sensor Web Enablement-Architektur der OGC Sensor Planning Service (SPS) spezifiziert [21]. Hierbei handelt es sich um eine Web Service-Schnittstelle, welche verschiedene Operationen zur Definition und Verwaltung von Sensor-Aufgaben bietet. Typische Aufgaben, welche mit einem SPS-Server gesteuert werden können, reichen von einfachen Einstellungsänderungen (z. B. Änderung der Messfrequenz eines Sensors) bis hin zur Übermittlung komplexer Beobachtungsaufgaben (z. B. Übermittlung einer Route welche durch eine Drohne abgeflogen und fotografiert werden soll).

Die Grundsätzliche Funktionsweise des SPS wird über eine Reihe von Operationen realisiert. Wichtige Operationen aus der SPS-Spezifikation sind beispielsweise:

- **GetCapabilities:** Abfragen einer Selbstbeschreibung einer SPS-Instanz. Diese Metadaten umfassen u. a. Informationen, welche Sensoren über einen SPS gesteuert werden können und welche Aufgaben/Steuerungsfunktionen für diese Sensoren verfügbar sind.
- **DescribeTasking:** Abfragen, welche Parameter für die Ausführung einer bestimmten Aufgabe an den SPS-Server zu übergeben sind.
- **Submit:** Starten der Ausführung einer Aufgabe.
- **GetStatus:** Abfragen des Ausführungsstatus einer zuvor übermittelten Aufgabe.
- **DescribeResultAccess:** Der SPS ist nur dazu bestimmt, die Steuerung von Sensoren zu ermöglichen. Die Bereitstellung der im Rahmen einer Aufgabenausführung gesammelten Daten ist dagegen jedoch keine SPS-Funktion. Daher bietet die SPS-Schnittstelle die DescribeResultAccess-Operation an, sodass Nutzer ermitteln können, wie die resultierenden Daten abgerufen werden können (z. B. über einen SOS-Server).

Durch einzelne Organisationen erfolgte bereits eine erste Evaluierung des SPS-Standards. Allerdings sind die Interoperabilitätsvorteile beim SPS-Standard im Vergleich zu den vorher genannten Spezifikationen weniger ausgeprägt: Die Anzahl der Nutzer und Anwendungen, welche die Steuerung von Sensoren übernehmen ist deutlich geringer als die Menge der Daten-Konsumenten. Daher ist kritisch zu prüfen, welche Interoperabilitätsvorteile im Einzelfall bestehen, die für die Nutzung des SPS-Standards sprechen oder ob individuelle Lösungen effizienter umzusetzen sind.



## **3 Anforderungen an SOS-Implementierungen in der Hydrologie**

### **3.1 Anforderungen an Server-Implementierungen**

#### **3.1.1 Allgemeine Anforderungen**

Im Rahmen einer Implementierung ist festzulegen, welche Version des SOS-Standards zu unterstützen ist. Während in der Praxis häufig noch Installationen auf Basis des SOS 1.0.0-Standards zu finden sind, wird im Rahmen dieser Handlungsempfehlung nur die Version 2.0 betrachtet. Diese Version beinhaltet zahlreiche Verbesserungen gegenüber der Version 1.0.0 (z. B. besser strukturierte Metadaten zu SOS-Servern, erhöhte Interoperabilität), sodass die Version 1.0.0 höchstens zusätzlich zur Version 2.0 implementiert werden sollte.

Wichtig ist in diesem Zusammenhang der Begriff der „Conformance Class“. Dieser beschreibt einen zusammenhängenden Ausschnitt aus dem SOS-Standard, welcher einen bestimmten, fest definierten Funktionsumfang umfasst. Die Conformance Class „SOS Core“ stellt hierbei den Kern des SOS-Standards dar, welcher durch jede SOS-Implementierung zwingend unterstützt werden muss.

Neben der Unterstützung des SOS 2.0-Standards wird weiterhin empfohlen, die zusätzlichen Spezifikationen des OGC SOS 2.0 Hydrologie-Profiles (OGC SOS 2.0 Hydrology Profile) zu verpflichtenden Anforderungen zu machen. Hierdurch wird ein hohes Maß an Interoperabilität zu weiteren hydrologischen SOS-Servern und Clients sichergestellt.

#### **3.1.2 Zu unterstützende SOS-Operationen**

Nachfolgend werden wesentliche Operationen der SOS-Schnittstelle vorgestellt. Es wird einerseits die grundlegende Funktionalität dieser Operationen dargestellt und andererseits welche Bedeutung sie im Rahmen von Sensor Web-Anwendungen besitzen.

##### **3.1.2.1 GetCapabilities**

Über die GetCapabilities-Operation können die Metadaten zu einem SOS-Server, d. h. z. B. unterstützte Operationen und verfügbare Daten, abgefragt werden. Die GetCapabilities-Operation ist durch den SOS-Standard vorgeschrieben. Daher muss sie von jedem Server angeboten werden, sofern er den OGC Sensor Observation Service-Standard erfüllen soll.

Für die Entwicklung von SOS-Client-Anwendungen ist diese Operation grundlegend, da hierüber die notwendigen Metadaten zu einem SOS-Server bereitgestellt werden.

##### **3.1.2.2 GetObservation**

Mit Hilfe der GetObservation-Operation können Messdaten von einem SOS-Server abgerufen werden. Hierbei stehen verschiedene Abfragefilter zur Verfügung. Diese Filtermöglichkeiten umfassen z. B. die Auswahl der Messparameter und Messstationen sowie räumliche und zeitliche Einschränkungen.

Die GetObservation-Operation ist durch den SOS-Standard vorgeschrieben. Daher muss sie von jedem Server angeboten werden, sofern er den OGC Sensor Observation Service-Standard erfüllen soll. Sie stellt die grundlegende Operation dar, welche den Datenabruf von einem SOS-Server ermöglicht.

Weiterhin ist eine Festlegung der zu unterstützenden Antwort-Formate für die GetObservation-Operation sinnvoll. Im Bereich der Hydrologie ist die Nutzung von WaterML 2.0 empfohlen. Da WaterML 2.0 durch das OGC SOS 2.0 Hydrologie-Profil vorgegeben wird, wird dieses Format automatisch von jedem SOS-Server unterstützt, sofern er dieses Profil erfüllt.

### 3.1.2.3 DescribeSensor

Mit Hilfe der DescribeSensor-Operation können die Metadaten zu einem Messprozess oder Sensor abgefragt werden, welcher bestimmte Messdaten generiert hat, die über einen SOS-Server angeboten werden. Die DescribeSensor-Operation ist durch den SOS-Standard vorgeschrieben. Daher muss sie von jedem Server angeboten werden, sofern er den OGC Sensor Observation Service-Standard erfüllen soll. Wird WaterML 2.0 als Ausgabe-Format eines SOS-Servers genutzt, so werden wesentliche Metadaten zu Messprozessen bereits über die GetObservation-Antworten von SOS-Servern ausgeliefert. Um eine Standard-Konformität sicherzustellen, ist die Unterstützung dieser Operation jedoch dennoch erforderlich.

Weiterhin ist eine Festlegung der zu unterstützenden Antwort-Formate für die DescribeSensor-Operation sinnvoll. Im Bereich der Hydrologie ist die Nutzung des sogenannten WaterML 2.0 ObservationProcess empfohlen. Da der WaterML 2.0 ObservationProcess durch das OGC SOS 2.0 Hydrologie-Profil als Inhalt von DescribeSensor-Antworten vorgegeben wird, wird dieses Format automatisch von jedem SOS-Server unterstützt, sofern er dieses Profil erfüllt.

### 3.1.2.4 GetFeatureOfInterest

Die GetFeatureOfInterest-Operation erlaubt die Abfrage der Features (= Geoobjekte), welchen die über einen SOS-Server angebotenen Messungen zugeordnet sind. Typischerweise sind dies die Messstationen, an denen die Werte gemessen wurden. Es besteht die Möglichkeit, diese Features anhand verschiedener Abfrageparameter wie Raumausschnitt oder gemessene Parameter zu filtern.

Die GetFeatureOfInterest-Operation ist durch das OGC SOS 2.0 Hydrologie-Profil vorgeschrieben. Daher muss sie von jedem Server angeboten werden, sofern er zum OGC SOS 2.0 Hydrologie-Profil konform sein soll. Diese Operation besitzt insbesondere für die Client-Entwicklung eine große Bedeutung. Hierüber lassen sich beispielweise die notwendigen Daten abrufen, um eine kartenbasierte Datenauswahl zu ermöglichen.

### 3.1.2.5 GetDataAvailability

Die GetDataAvailability-Operation wurde als Erweiterung des SOS-Standards entwickelt um in einfacher Weise abfragen zu können, für welche Kombination der SOS-Abfrageparameter Offering, Procedure (entspricht dem Messprozess oder Sensortyp), FeatureOfInterest (entspricht der Messstation) und ObservedProperty (entspricht dem Messparameter) in welchen Zeiträumen Messdaten verfügbar sind. Somit ist diese Operation eine wichtige Grundlage um SOS-Clients zu entwickeln, welche komfortable Mechanismen zur Auswahl der anzuzeigenden Daten bieten.

Die GetDataAvailability-Operation wird momentan nur im OGC SOS 2.0 Hydrologie-Profil spezifiziert und nicht im OGC SOS 2.0-Standard selbst. Sie muss von jedem Server angeboten werden, sofern er zum OGC SOS 2.0 Hydrologie-Profil konform sein soll. Weiterhin wurde die GetDataAvailability-Operation von der aktualisierten INSPIRE Technical Guidance zu O&M- und SWE-Diensten aufgenommen. Auch für die Client-Entwicklung ist die GetDataAvailability-Operation ein sehr wichtiger funktionaler Baustein. Nur diese Operation ermöglicht differenzierte Abfragen, welche Zeitreihen an welchen Standorten, zu welchen Messparametern für welche Zeiträume von einem SOS-Server angeboten werden.

### 3.1.2.6 Spatial Filtering Profile

Das sogenannte Spatial Filtering Profile des SOS-Standards ist eine optionale Erweiterung um die Abfrage von Messungen anhand ihrer Geometrie zu vereinfachen. In üblichen SOS-Implementierungen wird die Geometrie einer Messung indirekt über die Station festgelegt, an der sie erzeugt wurde. Im Rahmen des Spatial Filtering Profiles werden dagegen auch alle Messwerte direkt mit einer Positionsangabe versehen. Dies ist insbesondere für die Fälle sinnvoll, in denen sich die Position eines Sensors ändern kann (z. B. mobile Sensor Plattformen). Werden dagegen nur Messdaten stationärer Sensoren bereitgestellt, so besitzt diese SOS-Erweiterung eine geringere praktische Bedeutung.

### 3.1.2.7 Transaktionale SOS-Operationen

Mit Hilfe der transaktionalen SOS-Operationen ist es möglich Messdaten sowie Metadaten über standardisierte Operationen auf einen SOS-Server zu laden und somit zu veröffentlichen. Der Upload von neuen Messdaten wird hierbei über die InsertObservation-Operation vorgenommen, während Metadaten der zugehörigen Messprozesse über die InsertSensor-Operation an den SOS-Server übermittelt werden.

Die transaktionalen SOS-Operationen sind ein optionaler Bestandteil des SOS 2.0-Standards. Abschnitt 3.1.7.2.2 enthält weitergehende Informationen zu diesen Operationen.

Erfahrungsgemäß werden hydrologische Daten eher selten über die transaktionalen SOS-Operationen veröffentlicht. Dies liegt zum einen an bereits existierenden Datenloggern und Importern, die ihre eigenen Formate verwenden, aber auch an den Sicherheitsaspekten beim Import über Web Services. Daher ist dieser Ansatz eher geeignet, wenn Datenarchive importiert werden sollen (z. B. um per Import-Werkzeug CSV-Dateien über einen SOS-Server bereitzustellen), während die Integration mit bestehenden Datenakquisitionsprozessen üblicherweise auf anderen Wege erfolgt (z. B. bestehende hydrologische Fachsoftware).

### 3.1.3 INSPIRE

Neben den OGC SWE-Standards ist in Europa für viele Datenbereitsteller auch das Thema der INSPIRE-Konformität von Bedeutung (siehe 2.8). Sofern ein Datenanbieter sich entscheidet, entsprechend der INSPIRE-Richtlinie bestimmte Daten anzubieten, so kann dies für Beobachtungsdaten über einen INSPIRE-konformen SOS-Server erfolgen. Hierzu ist einerseits eine Erweiterung des SOS-Interfaces zu unterstützen, sodass dieser als INSPIRE Download Service genutzt werden kann (siehe 2.8.1). Andererseits sind als Ausgabeformat auch entsprechende INSPIRE Observation-Typen zu unterstützen (siehe 2.8.2).

Die INSPIRE-Richtlinie regelt in erster Linie die Bereitstellung ausgewählter Geodaten, z. B. die Lage von Messstellen. Typische Themenbereiche, welche im Rahmen der INSPIRE Technical Guidance identifiziert werden und auch Messdaten adressieren, sind z. B. [16]:

- Geologie (Geology, GE)
- Atmosphärische Bedingungen und Meteorologisch-geografische Kennwerte (Atmospheric Conditions and Meteorological Geographical Features, AC-MF)
- Umweltüberwachung (Environmental Monitoring Facilities, EF)
- Ozeanografisch-geografische Kennwerte (Oceanographic Geographical Features, OF)
- Meeresregionen (Sea Regions, SR)
- Boden (Soil, SO)
- Verteilung der Arten (Species Distribution, SD)

Inwieweit Messdaten verpflichtend bereitzustellen sind, ist noch Gegenstand laufender Diskussionen. Allerdings ist die Bereitstellung von Messdaten über INSPIRE-konforme Dienste generell als sinnvoll zu bewerten, um eine nachhaltige und einfache Nutzung der bereitgestellten Daten zu unterstützen. Ebenso ist man durch die Bereitstellung eines INSPIRE-konformen SOS-Servers auch für mögliche zukünftige Anforderungen gerüstet, welche ggf. die Bereitstellung von Messdaten erfordern.

### 3.1.4 Konformitätsprüfung

In Zusammenhang mit der Implementierung von standardisierten Schnittstellen und Datenformate besitzt die Überprüfung der Konformität von Software zu den entsprechenden Spezifikationen eine hohe Bedeutung. Für Teile der oben genannten Standards bestehen Software-Pakete bzw. Webseiten, welche eine Konformitätsprüfung unterstützen.

Für die Konformitätsprüfung zum OGC SOS 2.0-Standard bietet das OGC eine entsprechende Test-Suite an. Diese Tests basieren auf der sogenannten TEAM Engine (Test, Evaluation, And Measurement Engine) des OGC<sup>3</sup>. Dieses Open Source-Produkt auf Java-Basis erlaubt die Ausführung von Test-Skripts zur Überprüfung von Web-Diensten. Für den SOS 2.0-Standard besteht eine eigene Test-Suite welche mit Hilfe der TEAM Engine ausgeführt werden kann<sup>4</sup>.

Für die Validierung von WaterML 2.0-Daten bietet die australische Forschungsorganisation CSIRO einen webbasierten Validierungsdienst an<sup>5</sup>. Dieser ermöglicht den Upload von WaterML 2.0-Dateien, welche auf dem Validierungsserver überprüft werden sollen.

Für das OGC SOS 2.0 Hydrologie-Profil bestehen zurzeit noch keine formalen Tests. Bei den bestehenden Tests zu SOS 2.0 und WaterML 2.0 ist darauf hinzuweisen, dass diese zwar nutzbar sind, jedoch einzelne Fehler in der Konformitätsprüfung nicht auszuschließen sind (insbesondere falsche Alarmer). In diesen Fällen ist eine manuelle Überprüfung von Fehlermeldungen erforderlich.

### 3.1.5 Weitere Funktionen/Operationen

Die zuvor aufgeführten OGC Spezifikationen sind jeweils das Resultat eines Harmonisierungs- bzw. Standardisierungsprozesses. Dies resultiert in der Tatsache, dass die genannten Operationen quasi einen „gemeinsamen Nenner“ zwischen den beteiligten Organisationen darstellen. Darüber hinaus sind viele weitere produkt-spezifische Funktionen/Operationen denkbar, die über die eigentliche Funktionalität des SOS-Standards hinausgehen können.

### 3.1.6 Systemumgebung

Damit sichergestellt werden kann, dass ein SOS-Server einer vorhandenen Systemumgebung lauffähig ist, sind insbesondere folgende Informationen zu prüfen:

- Betriebssystem
- Ggf. vorhandene Einschränkungen bzgl. notwendiger Software-Grundlagen (z. B. verfügbare Laufzeitumgebungen (beispielsweise Limitierungen von Java-Versionen), Beschränkung auf bestimmte Web-Server-Implementierungen).
- Hardware (CPU/Arbeitsspeicher/Festplatte)

---

3 <https://github.com/opengeospatial/teamengine>

4 <http://cite.opengeospatial.org/teamengine/about/sos/2.0.0/site/index.html>

5 <http://waterml2.csiro.au/part1-validator/>

Sofern Installation und/oder Wartung des SOS-Servers durch einen externen Auftragnehmer erfolgen soll, ist erfahrungsgemäß ein Internetzugriff, z. B. über eine Remote-Verbindung zum Rechner, auf dem der SOS-Server installiert werden soll, sinnvoll.

### 3.1.7 Ansätze zur Datenhaltung

Damit ein SOS-Server Messdaten zugreifbar machen kann, ist es erforderlich, eine solche Server-Implementierung mit einer entsprechenden Komponente zur Datenhaltung zu verbinden. Je nach Erfordernissen sind hierzu verschiedene Strategien möglich. Grundsätzlich ist der Fall zu unterscheiden, dass ein SOS-Server auf eine bereits bestehende Datenbank aufgesetzt werden soll und der Fall, dass eine separate, neu zu erstellende Datenbank als Datenquelle für einen SOS-Server genutzt werden soll. In den nachfolgenden Abschnitten werden beide Varianten diskutiert und näher vorgestellt. Darüber hinaus existiert ein weiterer Lösungsweg, der darauf beruht, dass SOS-Server keinen direkten Datenbank-Zugriff erhalten. Anstelle von direkten Datenbankabfragen nutzt der SOS-Server in diesem Fall andere, zwischengeschaltete Komponenten um die benötigten Daten aus einer Datenbank abzufragen. Typischerweise handelt es sich in diesem Fall um eine gekapselte Datenzugriffskomponente welche beispielsweise ein (proprietärer) Web-Dienst (z. B. SOAP-Service) sein kann.

Es ist zu beachten, dass je nach spezifischen Anforderungen weitere Varianten oder Kombinationen von Varianten zur Datenhaltung eines SOS-Servers in Frage kommen. Ggf. sind auch Kombinationen der nachfolgend vorgestellten Ansätze sinnvoll.

Generell geht es in diesem Kapitel überwiegend um Anwendungsfälle, bei denen der SOS-Server und die Datenhaltung getrennt betrachtet werden. Alternativ dazu besteht auch die Möglichkeit, dass die Datenhaltung durch eine Fachanwendung übernommen wird, die dann ihrerseits einen SOS-Server auf den Daten bereitstellt. Ist die Fachanwendung die ausschließliche Datenverwaltung, dürfte die Art der Importe ohnehin geklärt sein, ansonsten wären lediglich diese Importe oder die Datenreplikation in das Fachsystem zu klären. Unter der Annahme, dass der inkludierte SOS-Server alle geforderten Bedingungen erfüllt sind alle weiteren Funktionen dann bereits automatisch verfügbar.

Um eine Abschätzung des Aufwandes der Anbindung eines SOS-Servers an eine Datenquelle zu ermöglichen, sind insbesondere die folgenden Informationen notwendig:

- Schematische Darstellung des bestehenden Datenbankmodells (empfohlen: Entity-Relationship-Diagramm)
- Erklärung des Datenbankmodells (ggf. stichpunktartig), sodass verständlich wird, welche Bedeutung die einzelnen Tabellen und ihre Spalten besitzen.
- Im Falle einer Datenzugriffsschnittstelle (z. B. API): exakte Spezifikation der Schnittstelle.

#### 3.1.7.1 Verwendung bestehender Datenbanken

Beim Aufsetzen eines SOS-Servers ist es naheliegend bereits vorhandene Datenbanken mit dem SOS-Server zu koppeln und als Datenquelle zu verwenden. Hierzu ist es jedoch wichtig, dass der SOS-Server so angepasst bzw. konfiguriert wird, dass er die Datenstruktur der bestehenden Datenbank zugreifen und korrekt interpretieren kann. Zu diesem Zweck existieren verschiedene Strategien welche in den nachfolgenden Abschnitten näher erläutert werden. Die nachfolgenden Erläuterungen beziehen sich auf SQL-basierte Datenbanken.

Bei den in Abschnitt 3.1.7.1.1 und 3.1.7.1.2 vorgestellten Lösungen findet eine Anpassung des SOS-Servers an das vorhandene Datenbankmodell statt. Dagegen beruht der in Abschnitt 3.1.7.1.3 beschriebene Lösungsweg darauf, dass eine bestehende Datenbank dem SOS-Server in Form von Da-

tenbank-Views zugänglich gemacht wird, welche ein Default-Datenmodell nachahmen, das durch eine gewählte SOS-Implementierung bereits unterstützt wird.

### 3.1.7.1.1 Entwicklung eines spezifischen Datenbankzugriffs-Moduls

Bei diesem Ansatz wird für den SOS-Server als Datenquelle eine bereits bestehende Datenbank verwendet. Zur Anbindung des SOS-Servers an diese Datenbank erfolgt eine Anpassung (Programmierung) des in der SOS-Implementierung enthaltenen Datenzugriffsmoduls, also des Moduls welches die jeweiligen Messdaten aus der Datenbank abfragt. Diese Lösungsvariante ist in Abbildung 5 illustriert.

Im Gegensatz zu den in den Abschnitten 3.1.7.1.2 und 3.1.7.1.3 beschriebenen Lösungen, welche auf der Nutzung einer Abstraktionsschicht für die Datenhaltung beruhen, erfordert diese Lösung Programmieraufwände. Weiterhin entsteht durch die spezifische Entwicklung eines Datenbankzugriffs-Moduls eine Sonderlösung, welche bei zukünftigen Updates der SOS-Implementierung eventuell ebenfalls angepasst werden muss. Das bedeutet, dass in diesem Fall mit einem erhöhten Aufwand zur Pflege des SOS-Servers zu rechnen ist. Wichtig ist in diesem Zusammenhang auch die Lizenz unter welcher der SOS-Server steht: Im Falle einer Open Source-Lizenz können entsprechende Anpassungen des Datenzugriffs durch den Auftraggeber oder frei wählbare Dienstleister erbracht werden, während bei einer proprietären SOS-Server-Implementierung i.d.R. eine Bindung an den Hersteller der Software besteht, der allerdings möglicherweise bei Updates auch eine aktualisierte Zwischenkomponente mitliefern kann.

Aufgrund des höheren Entwicklungsaufwands wird dieser Ansatz in der Praxis nur noch in wenigen Fällen gewählt. Insbesondere wenn Datenbankstrukturen vorliegen, welche weit von typischen Sensor Web-Datenmodellen entfernt sind und somit eine entsprechende Konfiguration des SOS-Servers schwierig oder nur mit signifikanten Performance-Einbußen einhergehen würde, stellt die Entwicklung eines angepassten, spezifischen Datenbankzugriffs-Moduls jedoch eine sinnvolle Option dar.

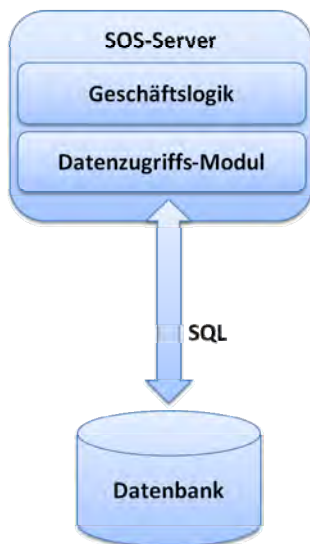


Abbildung 5: Schematische Darstellung der Verwendung eines spezifischen Datenbankzugriffs-Moduls

Um den Aufwand einer solchen Umsetzung abschätzen zu können, sind die folgenden Informationen notwendig:

- Schematische Darstellung des bestehenden Datenbankmodells (empfohlen: Entity-Relationship-Diagramm)
- Erklärung des Datenbankmodells (ggf. stichpunktartig), sodass potentielle Auftragnehmer verstehen können, welche Bedeutung die einzelnen Tabellen und ihre Spalten besitzen.

### 3.1.7.1.2 Datenbank-Abstraktionsschicht

Während der im vorherigen Abschnitt vorgestellte Ansatz Softwareentwicklungs-Arbeiten erfordert, wird in diesem Abschnitt eine flexiblere Lösung vorgestellt, welche es erlaubt anhand von Konfigurationsdateien eine Kopplung zwischen dem internen Datenmodell eines SOS-Servers und einer bestehenden Datenbank zu erreichen. Grundlage hierfür ist eine Datenbank-Abstraktionsschicht, wie sie beispielsweise durch Frameworks wie Hibernate<sup>6</sup> oder MyBatis<sup>7</sup> bereitgestellt wird. Die grundlegende Funktionsweise besteht hierbei darin, dass in einer Reihe von Konfigurationsdateien (sogenannten Mappings) festgelegt wird, in welchen Tabellen/Tabellenspalten ein SOS-Server welche der benötigten Informationen finden kann. Eine schematische Darstellung dieses Ansatzes wird in Abbildung 6 gezeigt.

Vorteil dieser Lösung ist vor allem der Wegfall von Softwareentwicklungs-Arbeiten, da eine Implementierung angepasster Datenbankzugriffs-Module nicht mehr notwendig ist. Somit besteht die Möglichkeit, einen SOS-Server durch Mitarbeiter aufzusetzen, welche mit der vorhandenen Datenbank vertraut sind, während Wissen über die Programmierung des SOS-Servers nicht mehr benötigt wird. Allerdings sind zusätzlich zwingend Kenntnisse der Datenbankabstraktionsschicht notwendig.

Die Grenzen dieser Lösung sind vor allem in den Fällen erreicht, in denen die zu nutzende Datenbank eine Struktur aufweist, welche die für einen SOS-Server notwendigen Abfragen nicht oder nur in umständlicher Form erlaubt. In diesem Fall können entsprechende Mappings evtl. keine zufriedenstellende Geschwindigkeit des SOS-Servers sicherstellen. Daher wird empfohlen vor der Umsetzung eines solchen Ansatzes eine kurze Prüfung durchzuführen, in wie weit das vorhandene Datenmodell für den in diesem Abschnitt beschriebenen Ansatz in Frage kommt.



Abbildung 6: Schematische Darstellung der Verwendung einer Datenbank-Abstraktionsschicht

Um den Aufwand einer solchen Umsetzung abschätzen zu können, sind die folgenden Informationen notwendig:

- Schematische Darstellung des bestehenden Datenbankmodells (empfohlen: Entity-Relationship-Diagramm)
- Erklärung des Datenbankmodells (ggf. stichpunktartig), sodass potentielle Auftragnehmer verstehen können, welche Bedeutung die einzelnen Tabellen und ihre Spalten besitzen.

6 <http://hibernate.org/>

7 <http://blog.mybatis.org/>



### 3.1.7.1.3 Datenbank-Views (Sichten)

Eine weitere Strategie, welche einerseits die bestehende Datenbank-Struktur unverändert lässt und andererseits keine Software-Entwicklungsarbeiten am SOS-Server benötigt, ist die Verwendung von Datenbank-Views. Bei diesem Ansatz kann durch den SOS-Server eine bestehende Datenbankzugriffsschicht weiter verwendet werden, ohne sie an ein anderes Datenmodell anzupassen. Gleichzeitig wird die vom SOS-Server erwartete Datenbankstruktur durch Datenbank-Views nachgebildet (siehe Abbildung 7).

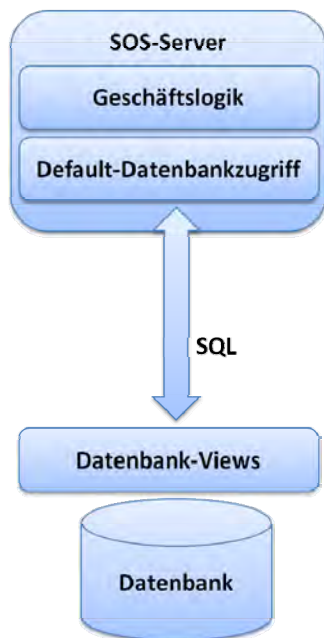


Abbildung 7: Schematische Darstellung der Verwendung von Datenbank-Views

Die Erstellung der Datenbank-Views wird üblicherweise durch Datenbank-Administratoren erfolgen. Allerdings ist hierbei eine Unterstützung (i.d.R. durch den Auftragnehmer) wichtig, um gemeinsam mit dem Datenbank-Administrator entsprechende Datenbank-Views zu definieren.

Wichtig bei der Verwendung von Views ist die Unterscheidung zwischen herkömmlichen Views, welche für eine eingehende Anfrage dynamisch erstellt werden und sogenannten Materialized Views. Bei einem solchen Materialized View besteht die Besonderheit darin, dass diese Views vorab durch die Datenbank erstellt und vorgehalten werden, sodass sie bei einer eingehenden Anfrage direkt zur Verfügung stehen. Wichtig ist hierbei, dass ein Ansatz festgelegt werden muss, wie diese Materialized Views aktualisiert werden. Möglich sind hierbei z. B. zeitliche Regeln (Update nach einer bestimmten Zeit) oder Datenbank-Trigger, z. B. falls neue Daten in die Datenbank eingefügt werden.

Vorteile dieses Ansatzes bestehen darin, dass keine Anpassung einer SOS-Server-Implementierung an ein spezifisches Datenbankmodell erfolgen muss und dass keine Änderung an einer bestehenden Datenbankstruktur nötig ist, sondern lediglich das Hinzufügen entsprechender Views.

Nachteile können darin bestehen, dass die anzulegenden Views gepflegt werden müssen und ggf. zusätzlicher Speicherplatz des Datenbank-Servers beansprucht wird um (Materialized) Views zu speichern. Weiterhin ist dieser Ansatz nur für einen lesenden Zugriff auf die Datenbank nutzbar. Schreibende Zugriffe (z. B. zur Implementierung der transaktionalen SOS-Operationen) sind dagegen nicht möglich.

Um den Aufwand einer solchen Umsetzung abschätzen zu können, sind die folgenden Informationen notwendig:

- Schematische Darstellung des bestehenden Datenbankmodells (empfohlen: Entity-Relationship-Diagramm)

- Erklärung des Datenbankmodells (ggf. stichpunktartig), sodass potentielle Auftragnehmer verstehen können, welche Bedeutung die einzelnen Tabellen und ihre Spalten besitzen.

### 3.1.7.2 Erstellung separater Datenbanken für SOS-Server

Alternativ zu den zuvor vorgestellten Lösungen, bei denen ein SOS-Server mit einer bestehenden Datenbank mit vorgegebenem Datenmodell gekoppelt wird, besteht auch die Möglichkeit, einen SOS-Server mit einer separaten Datenbank zu nutzen, welche gezielt als Datenquelle für den SOS-Server neu aufgesetzt wird.

Bei der Auswahl dieses Ansatzes spielen meist folgende Erwägungen eine Rolle:

- Verwendung eines optimierten Datenmodells, welches an die jeweilige SOS-Implementierung angepasst ist und eine hohe Abfrageperformance ermöglicht (z. B. falls bestehende Datenbankstrukturen keine effizienten Abfragen erlauben um die für SOS-Anfragen nötigen Daten zu erhalten).
- Vermeidung von Anpassungen der Implementierung des Datenbank-Zugriffs des SOS-Servers; stattdessen kann eine bestehende Datenbank-Zugriffsschicht ohne Anpassungen und umfangreichere Konfiguration wiederverwendet werden.
- Vermeidung, dass eine existierende Datenbank mit Abfragen, welche über den SOS-Server eingehen, stärker belastet wird.

Dem gegenüber stehen jedoch auch Nachteile, welche ggf. berücksichtigt werden müssen. Insbesondere zählt hierzu die Notwendigkeit eine weitere Datenbank zu betreiben und zu pflegen. Ggf. besteht jedoch die Möglichkeit, für den SOS-Server ein eigenes Datenbank-Schema zu nutzen und dadurch den Administrationsaufwand zu reduzieren.

Sofern der SOS-Server auch transaktionale Operationen anbietet, ist zusätzlich eine Strategie nötig, ob und wie die Inhalte der SOS-Datenbank in den ursprünglichen Datenbestand zurückgespielt werden sollen.

Weiterhin ist zu beachten, dass bei der Verwendung einer separaten Datenbank ein Mechanismus benötigt wird, mit dem die Messdaten in die Datenbank eingespielt werden. In den Abschnitten 3.1.7.2.1 und 3.1.7.2.2 werden zwei häufig angewendete Ansätze vorgestellt, mit denen die von einem SOS-Server genutzte Datenbank gefüllt werden kann.

Bei der Umsetzung dieses Ansatzes sind unterschiedliche Freiheitsgrade möglich, welche Lösungsoptionen in Frage kommen. Besteht im Hause bereits eine Festlegung auf ein spezifisches Datenbankmanagementsystem, so ist es sinnvoll auf dieses zurückzugreifen. Besteht keine Festlegung auf ein spezifisches Datenbankmanagementsystem, ergibt sich eine größere Vielfalt möglicher Lösungen. Grundsätzlich wird jedoch empfohlen, ein Datenbankmanagementsystem vorzugeben, welches durch die IT-Administratoren im Hause unterstützt werden kann.

Darüber hinaus ist festzulegen, über welchen Mechanismus, die durch den SOS-Server zu verwendende Datenbank mit Daten gefüllt wird. Hierzu existieren wiederum zwei verschiedene Lösungsansätze, welche nachfolgend vorgestellt werden.

#### 3.1.7.2.1 Datenimport aus bestehender Datenbank

Bestehen bereits eine oder mehrere Datenbanken, welche die zu veröffentlichenden Messdaten enthalten, so kann die für den SOS-Server zu nutzende Datenbank über Skripte oder Importe mit den Daten aus der vorhandenen Datenbank(en) befüllt werden (siehe Abbildung 8).

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

In diesem Fall ist für die Implementierung des Skripts oder des Importers zum Datentransfer eventuell eine Zusammenarbeit mit dem Administrator der Quelldatenbank nötig. Ggf. kann dieser sogar anhand der Dokumentation des SOS-Servers eigenständig entsprechende Skripte erstellen (ggf. mit Unterstützung des Anbieters des verwendeten SOS-Servers).

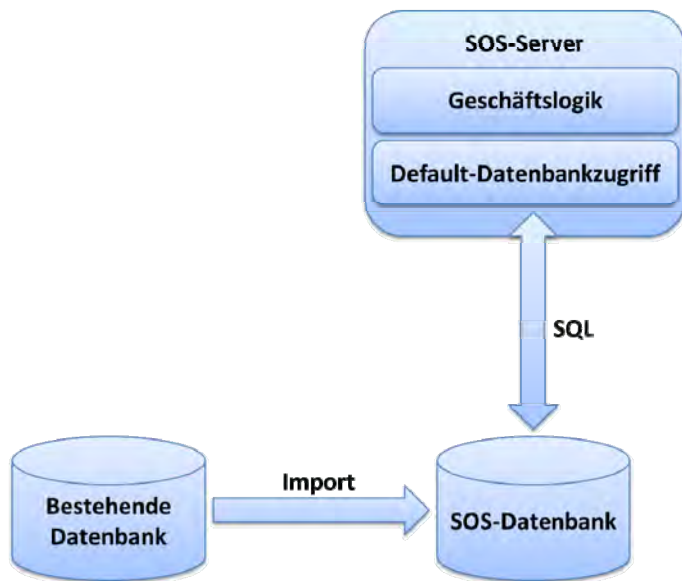


Abbildung 8: Schematische Darstellung des Datenimports aus bestehender Datenbank per SQL-Skript

Um den Aufwand einer solchen Umsetzung abschätzen zu können, sind die folgenden Informationen notwendig:

- Schematische Darstellung des bestehenden Datenbankmodells (empfohlen: Entity-Relationship-Diagramm)
- Erklärung des Datenbankmodells (ggf. stichpunktartig), sodass potentielle Auftragnehmer verstehen können, welche Bedeutung die einzelnen Tabellen und ihre Spalten besitzen.

### 3.1.7.2.2 Verwendung transaktionaler SOS-Operationen zum Datenimport

Ein weiteres mögliches Szenario ist, dass die über einen SOS-Server bereitzustellenden Daten noch nicht in einer Datenbank vorhanden sind und explizit über den SOS-Server selbst importiert werden sollen. Messdaten liegen dann zum Beispiel als Textdateien vor, in welchen die Messdaten (inkl. Informationen wie Zeitstempel, Messwert, usw.) durch Kommata getrennt enthalten sind (sogenannte CSV-Dateien). Dieser Fall tritt meist dann auf, wenn die von Datenloggern gesammelten Daten lediglich auf Festplatten oder anderen Speichermedien archiviert werden.

Um diese Daten für einen SOS-Server nutzbar zu machen ist es wichtig, die Daten in die vom SOS-Server zur Datenhaltung genutzte Datenbank zu importieren. Alternativ zu den bisher beschriebenen Möglichkeiten bietet der SOS-Standard zur Veröffentlichung von Messdaten die sogenannten transaktionalen Operationen `InsertSensor` und `InsertObservation` an. Diese Operationen können zum Beispiel von einer Einlese-Komponente („SOS-Importer“) verwendet werden um neue Daten und Metadaten auf den SOS-Server hochzuladen (siehe Abbildung 10).

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

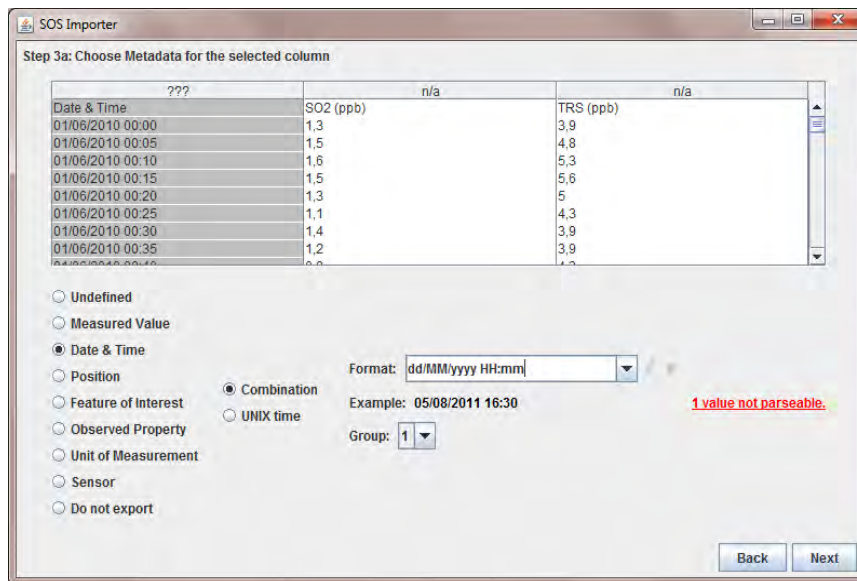


Abbildung 9: Beispiel für das Konfigurationsmenü eines SOS-Feeders

Je nach Implementierung bieten solche SOS-Feeder Funktionen an, um die Struktur einzulesender Dateien zu beschreiben, diese Dateien anhand der Strukturbeschreibungen zu parsen und die Daten anschließend über die transaktionalen SOS-Operation zu veröffentlichen. Abbildung 9 zeigt ein Beispiel für ein solches Werkzeug, mit welchem die Struktur einer CSV-Datei beschrieben werden kann. Wichtig ist hierbei, dass sowohl manuelle (d. h. ein Administrator startet den Import manuell) also auch automatische Import-Prozesse (d. h. der SOS-Feeder prüft regelmäßig das Vorhandensein neuer Daten) benötigt werden können.

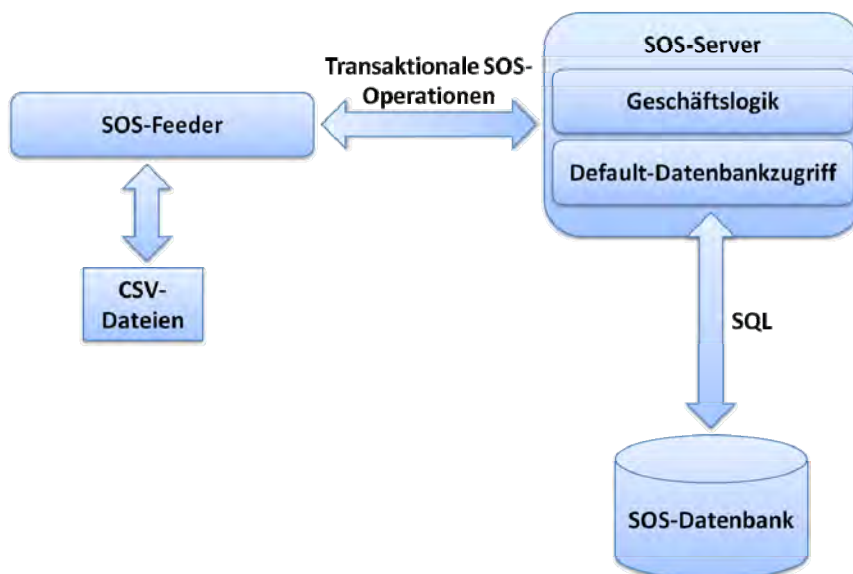


Abbildung 10: Schematische Darstellung der Verwendung transaktionaler SOS-Operationen zum Datenimport

Um den Aufwand einer solchen Umsetzung abschätzen zu können, sind die folgenden Informationen notwendig:

- Beschreibung der Datenquelle (z. B. Format und Struktur zu importierender Dateien)
- Beispieldaten, welche durch den Import Mechanismus zu laden sind
- Frequenz in der Datenaktualisierungen/neue Daten vorliegen

### 3.1.7.3 Proxy-Lösungen

Neben den beiden zuvor beschriebenen Varianten zur Datenhaltung eines SOS-Servers, existieren weiterhin auch Szenarien, in denen ein direkter Datenbankzugriff vermieden werden soll. Zu diesem Zweck wird der Datenbankzugriff durch eine eigenständige, vom SOS-Server unabhängige Komponente (Datenzugriffsdienst) gekapselt, welche durch den SOS-Server aber auch weitere Anwendungen, genutzt werden kann (siehe Abbildung 11). Beispiele sind IT-Infrastrukturen, in denen z. B. der Datenbank-Zugriff durch SOAP-Dienste gekapselt und Anwendungen bereitgestellt wird.

Der Hauptgrund für eine solche Architektur ist es, den SOS-Server unabhängig von Änderungen der internen Datenbankstruktur zu machen. D. h. selbst wenn sich das Datenbankmodell ändert, würde die Schnittstelle zum Datenzugriff gleich bleiben.

Die Vorteile liegen hierbei darin, dass bei einer Datenbankänderung lediglich die Datenzugriffskomponente angepasst werden muss, während alle Anwendungen, welche mit der Datenbank arbeiten sollen, weiterhin die unveränderte Datenzugriffsschnittstelle nutzen können. Dies reduziert die Wartungskosten bei Änderungen in der Datenbankstruktur. Zusätzlich kann die Zwischenkomponente bereits optimierte Zugriffe oder Caching anbieten.

Nachteile sind jedoch ggf. darin zu sehen, dass eine Performance-Optimierung eines SOS-Servers durch angepasste Datenbankabfrage-Strategien nicht ohne weiteres möglich ist. Weiterhin stellt die Kommunikation zwischen Datenzugriffskomponente und SOS-Server einen zusätzlichen Kommunikationsschritt dar, welcher zu Geschwindigkeitsverlusten führen kann. Dies gilt insbesondere für die Szenarien, in denen ein SOAP/XML-Dienst zum Datenzugriff genutzt wird und daher ein zusätzlicher Kodier-/Dekodier-Arbeitsschritt erforderlich wird.

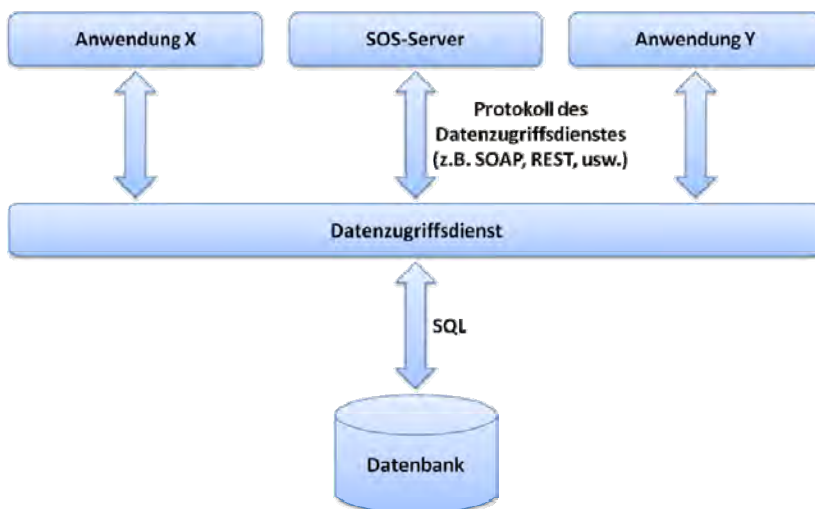


Abbildung 11: Schematische Darstellung der Nutzung eines Datenzugriffsdienstes

Wird dieser Ansatz zum Datenzugriff gewählt ist zu beachten, dass ggf. Caching-Mechanismen erforderlich werden können, mit denen der SOS-Server zwischenspeichert, welche Daten verfügbar sind. Dies ist insbesondere aus folgendem Grund notwendig: Der SOS-Standard schreibt die GetCapabilities-Operation vor, über welche ein Überblick über verfügbare Daten gegeben wird. Um diese Informationen zusammenzustellen, benötigt der SOS-Server einen Zugriff auf die entsprechenden Metadaten des zu nutzenden Datenzugriffsdienstes. Allerdings ist zu beachten, dass je nach Schnittstelle dieses Dienstes, diese Informationen ggf. über mehrere Aufrufe zusammengesetzt werden müssen. Sofern dies bei jeder GetCapabilities-Anfrage erneut erfolgen muss, kann dies negative Einflüsse sowohl auf die Antwortgeschwindigkeit des SOS-Servers als auch auf die Last auf den Datenzugriffsdienst haben. Daher ist es ggf. sinnvoll, dass je nach Schnittstelle des Datenzugriffsdienstes ein entsprechender Caching-Mechanismus für Metadaten über verfügbare Daten vorgehalten wird. Bei diesem Caching-Mechanismus ist weiterhin zu beachten, dass eine Update-Rate eingehalten wird, bei der sichergestellt ist, dass Informationen über neue Daten innerhalb einer

akzeptablen Geschwindigkeit in den Cache eingepflegt werden. Weiterhin wird empfohlen, dass ein Mechanismus bereitgestellt wird, um den Cache zu persistieren, sodass dieser nicht bei jedem Neustart des SOS-Servers neu aufgebaut werden muss.

Ist aufgrund der Datenmenge oder aufgrund der Schnittstelle des Datenzugriffsdienstes ein Mechanismus nötig, der die Anfragen an den Datenzugriffsdienst reduziert, so kann es sinnvoll sein, einen entsprechenden Cache vorzusehen.

Um den Aufwand einer solchen Umsetzung abschätzen zu können, sind die folgenden Informationen notwendig:

- Beschreibung der Schnittstelle und Datenformate des Datenzugriffsdienstes
- Falls verfügbar, Angabe wie der Anbieter den Datenzugriffsdienst im Zuge der Angebotserstellung zu Testzwecken zugreifen kann

### 3.1.7.4 Fachanwendungen mit nativer Datenhaltung

Vielfach betreiben Organisationen im Bereich der Hydrologie und Wasserwirtschaft bereits Fachanwendungen zur Speicherung, Verwaltung und Auswertung von Messdaten. In diesem Fall ist der Datenhaltung eng mit der bestehenden Fachanwendung gekoppelt und für die entsprechenden fachlichen Erfordernisse optimiert.

Aufgrund der zunehmenden Akzeptanz des SOS-Standards, wächst die Anzahl an Fachanwendungen, welche eine SOS-Schnittstelle anbieten. In diesem Fall kann die Bereitstellung der SOS-Schnittstelle ggf. direkt über die vorhandene Anwendung aktiviert oder durch die Installation entsprechender Erweiterungen bereitgestellt werden.

Vorteil dieses Ansatzes ist die direkte Einbindung in die bereits bestehende Software-Infrastruktur, sodass ggf. kein weiteres Produkt installiert werden muss. Weiterhin kann direkt die optimierte Datenhaltung der Fachanwendung genutzt werden, sodass keine Datenbankanpassungen notwendig sind. Updates am SOS-Server können direkt über Updates der jeweiligen Fachanwendung ausgeliefert werden.

Aus diesem Grunde ist diese Variante mit relativ geringen Aufwänden verbunden. Sofern die genutzten Fachanwendungen die SOS-Schnittstelle nicht unterstützen, sind jedoch eher aufwendige Erweiterungen nötig. In diesem Fall sind die vorhergehend genannten Alternativen ggf. effizienter umsetzbar.

### 3.1.7.5 Überblick

Tabelle 5 gibt einen Überblick über die Vor- und Nachteile der vorgestellten Strategien zur Datenhaltung eines SOS-Servers.

Tabelle 5: Überblick über Ansätze zur Datenhaltung eines SOS-Servers

	Verwendung bestehender Datenbanken			Für den SOS-Server neu zu erstellende Datenbank		Proxy-Lösungen	Fachanwendungen mit nativer Datenhaltung
	Entwicklung eines spezifischen Datenbankzugriffsmoduls	Datenbank-Abstraktionsschicht	Datenbank-Views (Sichten)	Datenimport aus bestehender Datenbank über SQL-Skripte	Verwendung transaktionaler SOS-Operationen zum Datenimport		
Möglichkeit zur Nutzung bestehender Datenbank	Ja	Ja	Ja (nur lesend)	Nein	Nein	Ja	Datenbank wird durch Fachanwendung mitgeliefert
Anpassungen an bestehender Datenbank	Nicht notwendig; Anpassungen an Datenbank könnten Integration jedoch vereinfachen	Nicht notwendig; Anpassungen an Datenbank könnten Integration jedoch vereinfachen	Anpassungen an bestehender Datenbank-Struktur nicht notwendig, es werden Datenbank-Views erstellt	Nicht notwendig; Anpassungen an Datenbank könnten Integration vereinfachen	Nicht notwendig	Nicht notwendig	Nicht notwendig
Erforderliche Anpassungen bei Änderungen an Datenbank	Programmierarbeiten erforderlich	Anpassung der Konfiguration/der Mappings	Anpassung der Datenbank-View- Erzeugung notwendig	Datenbank wird nur für SOS-Server genutzt, daher sind Änderungen unwahrscheinlich; Anpassungen können jedoch erforderlich werden, falls Synchronisierung zwischen SOS-Datenbank und Ursprungs-datenbank kontinuierlich erfolgen soll	Datenbank wird nur für SOS-Server genutzt, daher sind Änderungen unwahrscheinlich	Nicht notwendig, allerdings sind Anpassungen in den Fällen erforderlich, wenn sich die Schnittstelle des genutzten Zugriffsdienstes ändert	Da Datenbank direkt mit Fachanwendung gekoppelt ist, sind Anpassungen an der Datenbank nicht zu erwarten
Programmierarbeiten an SOS-Server	Erforderlich um Anpassung an vorhandene Datenbank vorzunehmen	Nicht notwendig, lediglich Konfiguration	Nicht notwendig	Nicht notwendig	Nicht notwendig	Erforderlich um Anpassung an Datenquelle vorzunehmen	Nicht notwendig, sofern bereits von Fachsoftware unterstützt



## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

	Verwendung bestehender Datenbanken			Für den SOS-Server neu zu erstellende Datenbank		Proxy-Lösungen	Fachanwendungen mit nativer Datenhaltung
	Entwicklung eines spezifischen Datenbankzugriffsmoduls	Datenbank-Abstraktionsschicht	Datenbank-Views (Sichten)	Datenimport aus bestehender Datenbank über SQL-Skripte	Verwendung transaktionaler SOS-Operationen zum Datenimport		
Möglichkeit von Updates der SOS-Server-Implementierung	Aufwendiger, da spezifisches Datenbankzugriffsmodul an neue SOS-Server-Version angepasst werden muss	Ggf. Anpassung der Konfiguration nötig, wenn sich bei Update des SOS-Servers das interne Datenmodell ändert	Ggf. Anpassung der Datenbank-Views nötig, wenn sich bei Update des SOS-Servers das interne Datenmodell ändert	Ggf. Anpassung der SQL-Skripte nötig, wenn sich bei Update des SOS-Servers das interne Datenmodell ändert	Nicht notwendig, da interne Struktur des SOS-Datenmodells hinter standardisierten Operationen gekapselt ist	Aufwendiger, da spezifisches Modul zur Kommunikation mit Datenzugriffsdienst an neue SOS-Server-Version angepasst werden muss	Einfach, da Updates durch Hersteller der Fachanwendung geliefert werden und Anpassungen der Datenbank automatisch vorgenommen werden können
Performance	Abhängig von bestehendem Datenbankmodell: Wie schnell können die benötigten Daten abgerufen werden?	Abhängig von bestehendem Datenbankmodell: Wie schnell können die benötigten Daten abgerufen werden?	Hoch, da Datenbank-Views so gestaltet werden können, dass optimale Abfrage-Geschwindigkeit erreicht wird	Hoch, da optimales SOS-Datenbank-Modell genutzt werden kann	Hoch, da optimales SOS-Datenbank-Modell genutzt werden kann	Abhängig von Geschwindigkeit des genutzten Datenzugriffsdienstes; zusätzlicher Geschwindigkeitsverlust durch zusätzlichen Kommunikations-Schritt; Caching-Mechanismen können diesen Geschwindigkeitsverlust allerdings ggf. ausgleichen	Hoch, da optimales SOS-Datenbank-Modell genutzt werden kann

## 3.2 Anforderungen an Client-Implementierungen

In diesem Kapitel werden Client-Applikationen zur Anzeige bzw. Verarbeitung von Daten, welche über SOS-Server bereitgestellt werden, betrachtet. Neben Beispielen für funktionale Anforderungen werden ebenso nicht-funktionale Anforderungen diskutiert, welche bei der Entscheidungsfindung zur Umsetzung eines SOS-Clients zu beachten sind.

Da die Entwicklung von SOS-Clients oft mit sehr unterschiedlichen Zielsetzungen erfolgt, ist es schwierig einen Überblick mit einem Anspruch auf Vollständigkeit zu erstellen. Stattdessen sind die nachfolgenden Anforderungen als Beispiele für Aspekte zu verstehen, welche für eine Entscheidung wichtig sein können. SOS-Konsumenten können in sehr unterschiedlicher Form vorliegen (z. B. Widgets, Apps für mobile Endgeräte, GIS, Zeitreihenmanagementsysteme, Web-Anwendungen oder Serverprozesse). Alle diese Komponenten können die SOS-Schnittstelle in gleicher Weise verwenden, sodass SOS-Server flexibel für eine große Bandbreite an Bausteinen als Datenquelle zur Verfügung steht. Die nachfolgende Darstellung fokussiert sich auf Web-Anwendungen als ein Beispiel für SOS-Clients.

### 3.2.1 Datenauswahl

Es existieren verschiedene Wege, wie Nutzer die in einer Anwendung anzuzeigenden Messdaten auswählen können. Im Wesentlichen kann hierbei zwischen einer kartenbasierten Auswahl von Daten und einer listenbasierten Datenauswahl unterschieden werden.

Eine intuitive Lösung zur Auswahl der darzustellenden Daten ist die Verwendung eines kartenbasierten Menüs. Hierbei hat der Nutzer einerseits die Möglichkeit aus einer Karte die Stationen auszuwählen, deren Daten angezeigt werden sollen. Andererseits besteht über Listen oder Drop-Down-Menüs die Möglichkeit anzugeben, für welche Parameter Daten gewünscht werden. Zur Erhöhung der Nutzerfreundlichkeit ist hierbei darauf zu achten, dass beide Auswahlmöglichkeiten gekoppelt werden (wird z. B. ein Parameter ausgewählt, so sollten auf der Karte nur noch die Stationen angezeigt werden, welche diesen Parameter messen).

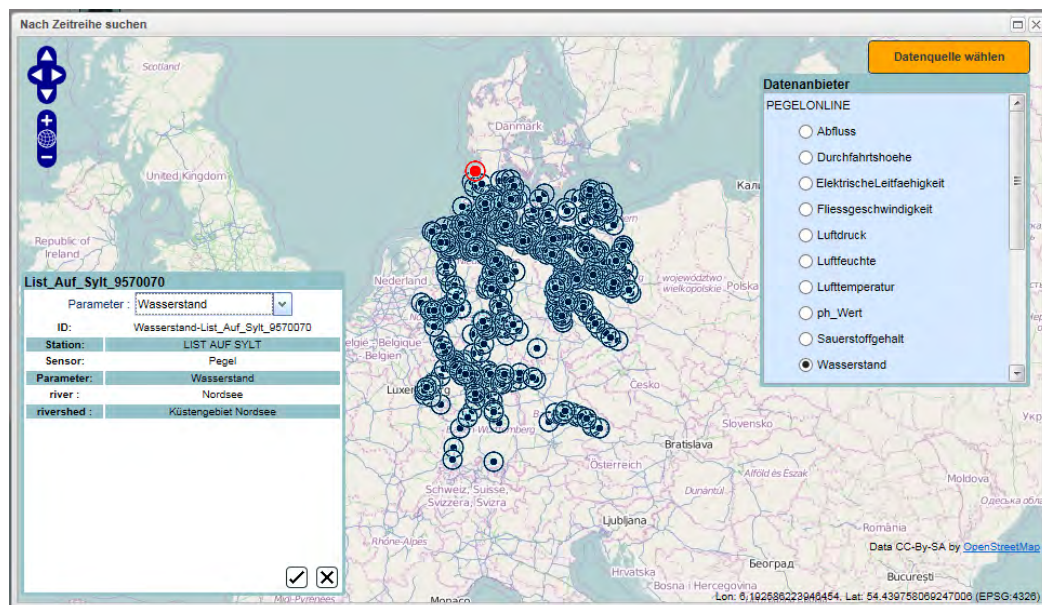


Abbildung 12: Beispiel für die kartenbasierte Auswahl von Zeitreihen

Bei der listenbasierten Datenauswahl (siehe Abbildung 13) erhält der Nutzer die Möglichkeit für verschiedene Abfrageparameter des SOS-Servers entsprechende Werte auszuwählen und somit zu spezifizieren, welche Daten (Zeitreihen) von Interesse sind. Sofern alle oben genannten Kriterien mit Werten belegt sind, sollte ein zu ladender Datensatz eindeutig identifiziert sein. Sinnvoll ist es hier-

bei, dass nach der Auswahl des Wertes für ein Kriterium die Auswahlfelder für die weiteren Abfrageparameter automatisch so angepasst werden, dass nur noch die Werte zur Auswahl stehen, welche zu den zuvor gewählten Einschränkungen passen. Beispiel: Es wurde bereits ein Parameter/ObservedProperty ausgewählt, dann sollte der Nutzer für das Feld Station nur noch die Stationen auswählen können, an denen dieser Parameter auch tatsächlich gemessen wird.

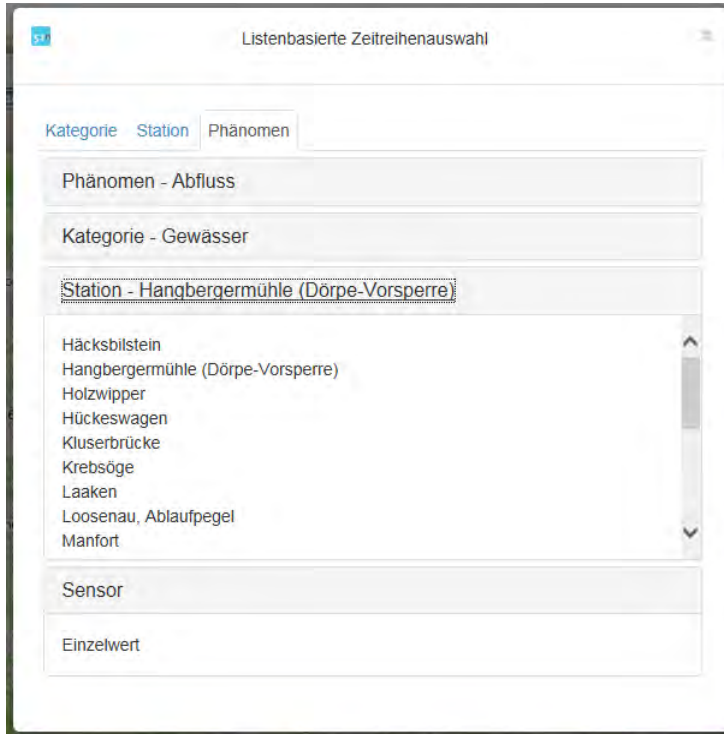


Abbildung 13: Beispiel für die listenbasierte Auswahl von Zeitreihen

### 3.2.2 Datendarstellung

Ein weiterer grundlegender Funktionsbereich ist die Darstellung von Messdaten. Auch hier existieren verschiedene Darstellungstypen:

- Tabellenansicht
- Diagrammansicht
- Kartenansicht

Bei der Tabellenansicht werden die verfügbaren Messdaten in einer Tabelle dargestellt. Neben einer Spalte, welche die verschiedenen Zeitstempel enthält werden in weiteren Spalten die jeweils zugehörigen Messwerte dargestellt. Zur genaueren Spezifikation der Tabellenansicht sind z. B. folgende Kriterien sinnvoll:

- Gewünschte Sortierungsmöglichkeiten, z. B. anhand von Zeitstempel und/oder Messwert
- Angabe ob ein Paging-Mechanismus gewünscht ist (d. h. Aufteilung der Tabelle auf mehrere Unterseiten um lange Tabellenansichten zu vermeiden; Hinweis: Paging wird durch die SOS-Schnittstelle nicht unterstützt, sodass dies Client-seitig zu lösen ist)
- Anzahl der Zeitreihen, welche in einer Tabellenansicht gleichzeitig darstellbar sein sollen

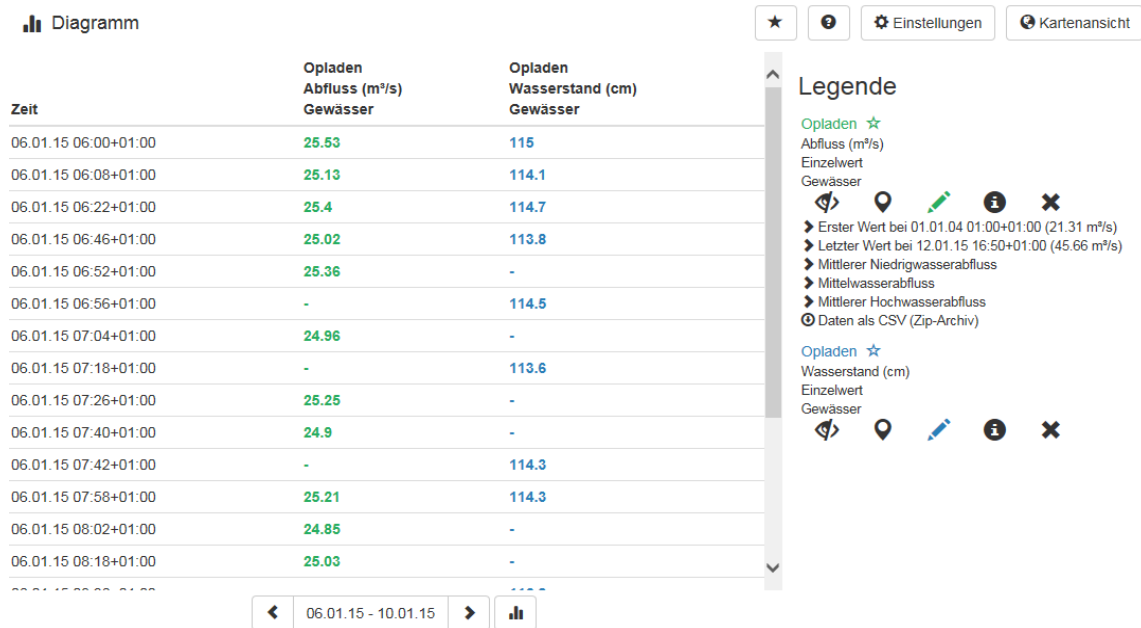


Abbildung 14: Beispiel zur Tabellenansicht

In der Diagrammansicht erfolgt eine graphische Darstellung der Messdaten in Form eines Diagramms. Hierdurch erfolgt eine anschauliche Präsentation der Messwerte und ihrer Änderungen im Laufe der Zeit.

Zur genaueren Spezifikation, welche Anforderungen an die Darstellung von Diagrammen bestehen, eignen sich die folgenden Kriterien. Hierbei ist zu beachten, dass dies nur ein exemplarischer Überblick ist und andere bzw. zusätzliche Anforderungen erforderlich sein können:

- Benötigte Diagrammtypen, z. B.:
  - Liniendiagramme (kumuliert und/oder nicht-kumuliert)
  - Balkendiagramme (kumuliert und/oder nicht-kumuliert)
- Anzahl der Zeitreihen, welche mindestens gleichzeitig im Diagramm darstellbar sein müssen
- Anpassungsmöglichkeiten um folgende Eigenschaften der Darstellung einstellen zu können, z. B.:
  - Farbe
  - Linienstärke
  - Skalierung des Diagramms: Null-basiert und/oder werte-abhängig angepasst
- Anzeige von durch den Nutzer auswählbaren Referenzwerten
- Navigationsmöglichkeiten, z. B.:
  - Datums- und Zeitangabe zur Auswahl des darzustellenden Zeitraums
  - Bereitstellung eines Übersichtsdiagramms zur Auswahl des darzustellenden Zeitraums
  - Panning innerhalb des Diagramms (z. B. per Maus, Tastatur oder Touch-Gesten)
  - Zooming innerhalb des Diagramms (z. B. per Mausekranz oder Touch-Gesten)

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

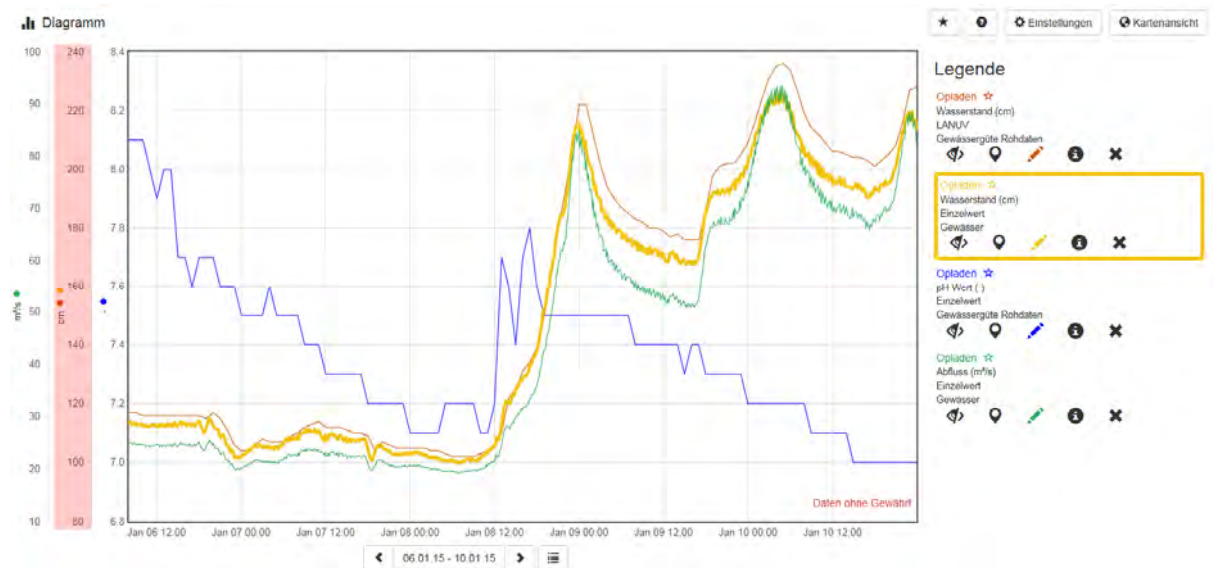


Abbildung 15: Beispiel zur Diagrammansicht

In den meisten aktuellen Client-Anwendungen wird die Kartendarstellung zur Visualisierung der Lage der Messstationen sowie zur Auswahl von darzustellenden Daten (siehe Abschnitt 3.2.1) verwendet.

Darüber hinaus existieren jedoch weitere Optionen, über welche reichhaltigere Informationen per Karte vermittelt werden können (wie z. B. in Abbildung 16 dargestellt, eine Visualisierung der aktuellen Trends wie „steigend“, „fallend“ und „gleichbleibend“). Allerdings ist zu beachten, dass für diese weitergehenden Anzeigevarianten entsprechende Komponenten benötigt werden, welche die notwendigen Informationen aus den SOS-Ausgaben generieren und mit dem Ziel einer guten Performance zwischenspeichern.

Bei der Beschreibung, welche Anforderungen an die Kartenansicht bestehen, sind beispielsweise folgende Aspekte relevant:

- Darstellung der aktuellen (neuesten) Messwerte für die angezeigten Stationen (ggf. nach Auswahl eines darzustellenden Parameters)
- Darstellung von Trends über einen bestimmten Zeitraum (steigend, fallend, gleichbleibend) für die angezeigten Stationen (ggf. nach Auswahl eines darzustellenden Parameters)
- Farbkodierung der angezeigten Stationen in Abhängigkeit von den aktuellen Messwerten und zugehörigen Referenzwerten

Weiterhin ist im Rahmen einer Ausschreibung zu berücksichtigen, inwieweit die für eine Darstellung notwendigen Informationen von SOS-Servern abgerufen werden können oder ob ggf. eine Ableitung und Zwischenspeicherung dieser Informationen erforderlich ist.



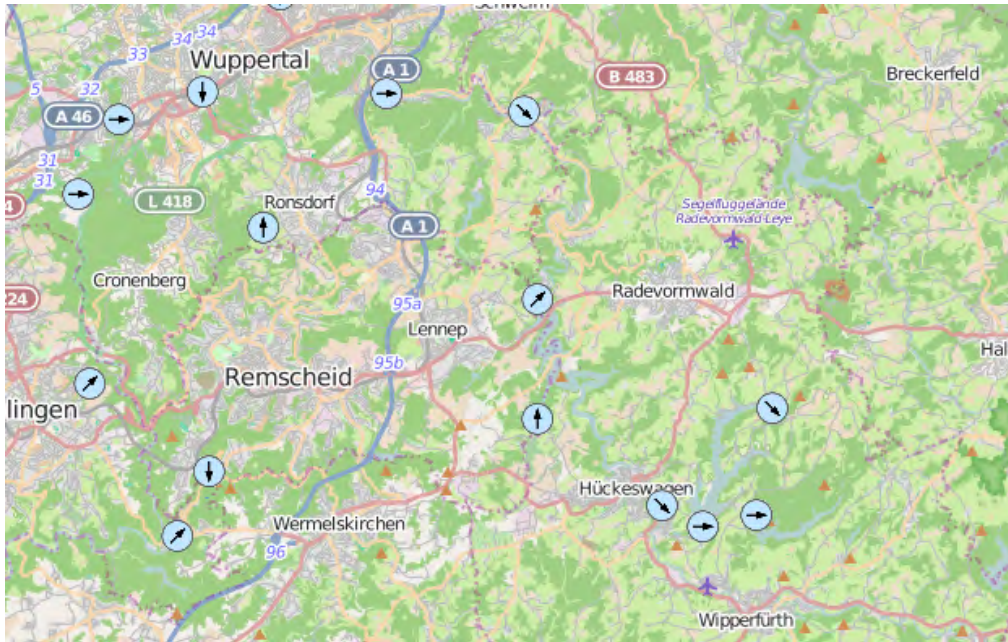


Abbildung 16: Beispiel zur Kartenansicht mit Trendanzeige

### 3.2.3 Weitere funktionale Anforderungen

Neben den zuvor vorgestellten grundlegenden Funktionen zur Datenauswahl und -anzeige sind ggf. weitere Anforderungen an Client-Anwendungen zu beachten. Diese werden nachfolgend dargestellt.

#### 3.2.3.1 Metadaten-Darstellung

Um den Nutzern eines SOS-Clients die Interpretation der angezeigten Daten zu ermöglichen, sind Metadaten von großer Bedeutung. Daher ist es wichtig, dass entsprechende Client-Anwendungen die Möglichkeit bieten, die verfügbaren Metadaten anzuzeigen. Da die Anzahl der denkbaren Metadaten-Elemente nahezu unbegrenzt ist, kann an dieser Stelle keine abschließende Aufzählung erfolgen. Stattdessen ist es wichtig in Abstimmung mit den zukünftigen Anwendern festzulegen, welche Anforderungen an die Metadaten bestehen. Weiterhin ist neben der Anzeige im Client auch zu berücksichtigen, dass die notwendigen Metadaten über den zu verwendenden SOS-Server zur Verfügung gestellt werden. Typische Metadaten-Elemente sind Angaben wie

- die Kategorie, zu der eine Zeitreihe gehört,
- die Organisation, welche die Daten erhoben hat,
- der für die Zeitreihe verantwortliche Ansprechpartner,
- der Typ/die Seriennummer des verwendeten Sensors,
- ...

#### 3.2.3.2 Generalisierung

Bei der Entwicklung von Client-Anwendungen ist zu beachten, dass oft eine wesentlich höhere Anzahl von Messwerten verfügbar ist, als in einem einzelnen Diagramm angezeigt werden können.

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Dies kann durch folgendes Beispiel illustriert werden: Eine Zeitreihe umfasst stündliche Werte. Bei einem Anzeigzeitraum von einem Jahr würde dies zu über 8500 Messwerten führen. Berücksichtigt man jedoch, dass ein Diagramm auf typischen Bildschirmen oft mit weniger als 1000 Pixeln dargestellt wird, wird deutlich, dass nur weniger als ein Achtel der Messwerte überhaupt abgebildet werden kann.

Um die zu übertragende Datenmenge zwischen Client und einer ggf. vorhandenen serverseitigen Komponente zu optimieren (dies betrifft insbesondere mobile Anwendungen mit Bandbreiteneinschränkungen), ist es sinnvoll, die Daten vor der Übertragung an den Client zu generalisieren.

Bei einer solchen Generalisierung ist es einerseits wichtig, dass die wesentlichen Charakteristika der Zeitreihe erhalten bleiben und gleichzeitig das Datenvolumen bestmöglich reduziert wird. Zu diesem Zweck sollte ein Vorschlag erarbeitet werden, welcher Generalisierungs-Algorithmus vorgeschlagen wird und wie dieser auf Client- und Serverseite eingebunden wird.

Weiterhin ist zu beachten, dass es verschiedene Wege gibt, derartige Generalisierungsmethoden in eine Sensor Web-Architektur zu integrieren. Daher ist es wichtig festzulegen, auf welcher Ebene eine Generalisierung erfolgen soll (z. B. beim Datenbankzugriff, in der Geschäftslogik eines SOS-Servers, oder innerhalb einer evtl. separat gekapselten Geschäftslogik einer Visualisierungskomponente).

### 3.2.3.3 Verlinkung

Hat ein Nutzer eines SOS-Clients eine Reihe verschiedener Zeitreihen zur Anzeige ausgewählt und ggf. sogar deren graphische Darstellung angepasst, so ist dies oft mit einem gewissen Aufwand verbunden gewesen. Daher liegt der Wunsch nahe, eine solche Darstellung zu einem späteren Zeitpunkt wieder aufrufen zu können, ohne die Daten erneut einzeln auswählen zu müssen. Aus diesem Grunde bieten viele Client-Anwendungen die Möglichkeit, die aktuelle Darstellung zum Beispiel über eine spezielle URL (sogenannter Perma-Link) wieder aufrufbar zu machen.

Typische Parameter, welche in solche Perma-Links aufgenommen werden können sind:

- Angezeigte Zeitreihen
- Angezeigter Zeitraum
- Graphische Darstellung (Farben, Linienstärken, Diagrammtypen) der angezeigten Zeitreihen

### 3.2.3.4 Datendownload

Oft besteht seitens der Anwender von Sensor Web-Clients der Bedarf, die angezeigten Daten zu speichern. Dies kann sowohl den Export in Form der reinen Messdaten (z. B. als CSV- oder Excel-Datei) betreffen um die Daten in anderen Programmen weiterzuverwenden, aber auch den Export der aktuellen Diagrammdarstellung in Form von Reports (PDF-Dateien) oder von Grafikdateien (z. B. JPG, PNG). Neben den neutralen Formaten (CSV und Microsoft Excel) ist auch eine Unterstützung der in der Wasserwirtschaft gängigen Formate eine wichtige Anforderung (siehe DWA-Merkblatt DWA-M 151 (August 2014): Messdatenmanagementsysteme (MDMS) in Entwässerungssystemen). Um im Rahmen eines Angebots eine präzise Aufwandsschätzung vornehmen zu können, ist es wichtig, genau anzugeben, welche der Datenformate gefordert werden.

Weiterhin ist es wichtig zu entscheiden, auf welcher Ebene ein Zeitreihenexport erfolgen soll: Soll es möglich sein die Werte einer einzelnen angezeigten Zeitreihe individuell zu exportieren oder sollen die Werte aller angezeigten Zeitreihen in Form mehrerer, in einem ZIP-Archiv zusammengefasster Dateien exportiert werden.

Abschließend ist darauf hinzuweisen, dass ggf. auch Einschränkungen beim Datendownload wichtig sind. So besteht z. B. die Frage, ob die Datenmenge, die ein Anwender herunterladen darf, beschränkt werden soll.

### 3.2.3.5 Einbindung externer SOS-Instanzen

Ein weiterer Aspekt ist die Einbindung externer SOS-Instanzen in Client-Applikationen. Hier ist zu prüfen, in wie weit bereits bestehende SOS-Server bekannt sind, welche eingebunden werden sollen. In diesem Zusammenhang ist insbesondere zu prüfen, um welche SOS-Version und ggf. SOS-Profile es sich bei diesen Instanzen handelt. Auf dieser Basis erhalten Client-Entwickler die Möglichkeit zu prüfen, in wie weit ggf. Anpassungen oder Erweiterungen am bestehenden Source-Code erforderlich sind oder ob die geforderten SOS-Versionen bereits vollständig unterstützt werden.

Ein weiterer wichtiger Aspekt ist die Frage, ob die Einbindung externer SOS-Server dynamisch erfolgen soll. Einerseits bestünde die Möglichkeit, einzubindende SOS-Server statisch in der Konfiguration des SOS-Clients festzulegen. Dieser Fall ist insbesondere für solche Client-Applikationen gegeben, welche Metadaten von SOS-Servern in einem internen Cache zwischenspeichern, um eine schnellere Interaktion zu ermöglichen. Andererseits ist auch der Fall zu finden, dass neue SOS-Server von Nutzern dynamisch während der Nutzung einer Client-Anwendung hinzugefügt werden sollen. In diesem Fall sind entsprechende Menüs zur Verwaltung von SOS-Servern und zum Testen der Verbindung zu diesen Servern notwendig.

## 3.2.4 Systemumgebungen

Nachfolgend werden verschiedene denkbare Systemumgebungen beschrieben, von denen aus Daten von einem SOS-Server konsumiert werden können. Allerdings handelt es sich dabei um keine erschöpfende Übersicht, sondern soll einige grundlegende Aspekte betrachten, welche bei der Entwicklung eines SOS-Clients eine Rolle spielen. Weiterhin ist es wichtig darauf hinzuweisen, dass die Wahl der Client-Plattform stark von der zu adressierenden Nutzergruppe abhängt. So kann es beispielweise unterschiedliche Anforderungen geben, je nachdem ob der SOS-Client durch Mitarbeiter einer Fachbehörde genutzt werden soll oder durch externe Nutzer. Somit ist es bei der Formulierung von Anforderungen an einen SOS-Client sinnvoll anzugeben, welche Nutzergruppen von der Software adressiert werden sollen.

### 3.2.4.1 Client-Plattformen

Je nach Zielplattform des benötigten SOS-Clients sind unterschiedliche Angaben notwendig um festzulegen, welche technischen Anforderungen und Rahmenbedingungen zu beachten sind. Im Falle eines webbasierten Clients sind insbesondere folgende Informationen erforderlich:

- Betriebssystem (z. B. Windows, Linux-Varianten, Mac OS, Android, Windows Phone, iOS, etc.) einschließlich Versionsangaben
- Browser einschließlich Versionsangaben

Ist dagegen ein SOS-Client als Erweiterung eines bestehenden Software-Pakets gewünscht, so sind dagegen folgende Informationen nötig:

- Betriebssystem (z. B. Windows, Linux-Varianten, Mac OS, Android, Windows Phone, iOS, etc.) einschließlich Versionsangaben
- Name und Version des um SOS-Client-Funktionalität zu erweiternden Software-Pakets



### 3.2.4.2 Server

Sofern es sich bei einer geplanten SOS-Client-Entwicklung um eine Web-Applikation handelt, ist wichtig festzulegen, in welcher Web-Server-Umgebung der Client gehostet werden soll. Hierzu sind insbesondere folgende Informationen nötig:

- Betriebssystem
- Ggf. vorhandene Einschränkungen bzgl. notwendiger Software-Grundlagen (z. B. verfügbare Laufzeitumgebungen (beispielsweise Limitierungen von Java-Versionen), Beschränkung auf bestimmte Web-Server-Implementierungen).
- Hardware (CPU/Arbeitsspeicher/Festplatte)

Sofern weitere Anforderungen an zu nutzende Technologien, Konzepte, Programmiersprachen, Komponenten bestehen, sind diese in diesem Zusammenhang ebenfalls zu definieren.

Werden externe Dienstleister mit der Entwicklung des Clients beauftragt oder wird eine bereits bestehende Software genutzt, so ist abzuklären, ob die Installation und/oder Wartung durch den Auftragnehmer erfolgen soll. Falls ja, kann es sinnvoll sein, diesem Dienstleister eine Remote-Verbindung zum Server, auf dem der SOS-Client installiert werden soll, bereitzustellen.

## 3.3 Nichtfunktionale Anforderungen

### 3.3.1 Dokumentation

Um die Installation und Wartung einer SOS-Server- und/oder SOS-Client-Installation sicherzustellen, ist eine ausreichende Dokumentation notwendig. Wichtig sind dabei insbesondere:

- Installations- und Betriebsanleitung einschließlich einer Dokumentation der einstellbaren Parameter des SOS-Servers
- Dokumentation der Systemarchitektur einschließlich der durch den SOS-Server genutzten [Datenbankanbindung/Datenbankmodelle/Mappings zwischen SOS-Modell und bestehender Datenbank].
- Benutzerdokumentation für einen ggf. zu liefernden SOS-Client (z. B. in Form eines PDF-Handbuchs, einer Online-Hilfe oder interaktiver Lernhilfen)

### 3.3.2 Lizenzierung

Es sollte angegeben werden, welche Vorgaben an die Lizenzierung der bereitzustellenden SOS-Implementierung bestehen. Hierbei kann zum Beispiel zwischen Open-Source und proprietärer Software unterschieden werden. Sofern eine Open Source-Lizenz gewünscht ist, ist es weiterhin wichtig festzulegen, ob aufgrund bereits vorhandener, programmatisch einzubindender Komponenten eine bestimmte Lizenz zu nutzen ist. Ebenso sind in diesem Zusammenhang unterschiedliche Modelle bzw. Kosten für Wartung, Pflege und Weiterentwicklung zu berücksichtigen, welche ggf. in Zusammenhang mit der Lizenzierung zu beachten sind.

### 3.3.3 Leistungsanforderungen

Im Hinblick auf die Leistungsanforderungen sind verschiedene Aspekte zu beachten. Zunächst ist es wichtig Angaben zu machen, mit welchen Datenmengen der zu liefernde SOS-Server arbeiten muss

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

bzw. welche Datenmengen ein SOS-Client verarbeiten muss. Hierbei sind insbesondere folgende Parameter von Interesse:

### ■ SOS-Server

- Ungefähre Anzahl der Messungen (Größenordnung)
- Ungefähre Anzahl der Zeitreihen (Größenordnung)
- Ungefähre Anzahl der Messstationen (Größenordnung)
- Ungefähre Anzahl der gemessenen Parameter (Größenordnung)

### ■ SOS-Clients

- Ungefähre Anzahl der maximal darzustellenden Messungen pro Zeitreihe (Größenordnung)
- Ungefähre Anzahl der Zeitreihen (Größenordnung)
- Ungefähre Anzahl der Messstationen (Größenordnung)

Gleichzeitig ist es weiterhin wichtig anzugeben, ob bestimmte Anforderungen an die Antwortgeschwindigkeit des SOS-Servers bzw. die Darstellungsgeschwindigkeit des SOS-Clients bestehen und welche Lasten zu erwarten sind:

- Ungefähre Abschätzung, wie viele Anfragen pro Zeitperiode (z. B. Anfragen pro Stunde) zu erwarten sind
- Anforderungen an die Antwortgeschwindigkeit (z. B. wie lange darf die Auslieferung einer bestimmten Anzahl an Messungen dauern)

### 3.3.4 Skalierbarkeit

Je nach Anwendungsfall und Datenmenge muss ein SOS-Server eine gewisse Skalierbarkeit aufweisen um höhere Last auffangen zu können. Dies bezieht sich einerseits auf den SOS-Server selbst, andererseits aber auch auf die Form der Datenhaltung. So kann ein einzelner Server nicht beliebig viele Clients bedienen, was entweder den Parallelbetrieb mehrerer Server oder einen Cluster voraussetzt.

Ähnliches gilt für die meist verwendeten SQL-Datenbanken in der Datenhaltung. Diese können zwar durch Aufstocken der Hardware Ressourcen oder Clustering bis zu einem gewissen Grad skaliert werden, dies ist jedoch oft, abhängig vom genutzten Datenbankmanagementsystem, verbunden mit hohen zusätzlichen Lizenzkosten. Alternativ dazu kann zum Beispiel die Nutzung einer besser skalierbaren Datenhaltung (wie z. B. verteilte noSQL Datenbanken) erwünscht sein.

### 3.3.5 Ausfallsicherheit

Der Betrieb öffentlicher Dienste erfordert in der Regel Ausfallsicherheit für alle beteiligten Komponenten. Diese kann redundante Zugriffswege und Datenhaltung erforderlich machen. Außerdem müssen möglicherweise alle Komponenten über aufrufbare Funktionen über die Betriebsbereitschaft Auskunft geben können, sodass zum Beispiel ein Network Load Balancer (NLB) jederzeit den Zustand aller Teilsysteme erfragen kann.

Weiterhin sind auch Aspekte wie Denial Of Service-Probleme durch häufige Anfragen nach Daten zu sehr langen Zeiträumen zu beachten. Ggf. sind hierzu Mechanismen nötig, wie sie im SOS 2.0 Hydrologie-Profil beschrieben werden. Dies können beispielsweise Limitierungen der Anfragegrößen und Ausgabedatenmengen sein.

### 3.3.6 Software Erstellungs- und Supportkonzept

Ein SOS-Server bzw. SOS-Client kann in der Dissemination von Messdaten und ihrer Visualisierung eine zentrale Rolle einnehmen. Durch einen Einblick in das Software Release- und Supportkonzept kann die Nachhaltigkeit eines angebotenen Sensor Web Software-Konzepts bewertet werden. Wichtig ist hierbei eine Betrachtung, wie die Investitions- und Produktstrategie zum weiteren Ausbau des SOS-Servers bzw. SOS-Clients, sowie zu dessen Wartung gestaltet ist.

Es wird weiterhin empfohlen, Antwortzeiten und Fehlerbehebungsgarantien über einen Wartungsvertrag abzusichern.

Um eine kontinuierliche Weiterentwicklung und Fehlerbehebung des SOS-Servers bzw. SOS-Clients zu garantieren, kann es darüber hinaus hilfreich sein, die Anbieter zu bitten, sofern möglich, ihren Software-Entwicklungsplan sowie die zugrunde gelegten Release-Zyklen darzustellen.

### 3.4 Zugriffskontrolle

Bei der Bereitstellung von Messdaten über einen SOS-Server sind ggf. Zugriffsbeschränkungen zu beachten. Das bedeutet, dass beispielweise bestimmte Daten nur von einem eingeschränkten Kreis von Nutzern zugegriffen werden dürfen. Hierzu existieren verschiedene Ansätze, welche nachfolgend kurz erläutert werden.

Da die Zugriffskontrolle nicht über den SOS-Standard abgedeckt ist, sind zusätzliche Lösungen erforderlich. Aufgrund der großen Bandbreite möglicher Lösungen, ist eine Entscheidung im Einzelfall sinnvoll. Grundsätzlich sind im Hinblick auf die Zugriffskontrolle folgende Fragen zu beachten.

- Nach welchen Kriterien ist der Zugriff auf Zeitreihen zu kontrollieren (z. B. zulässige Funktionen, räumliche Einschränkungen, zeitliche Einschränkungen, Limitierung auf bestimmte Messgrößen)?
- Sind bestimmte bereits vorhandene Nutzerdatenbanken zu verwenden?
- Sind bestimmte bereits vorhandene Komponenten zur Zugriffskontrolle zu nutzen und ggf. zur Unterstützung des SOS-Standards zu erweitern?
- Welche Leistungsdaten werden erwartet (wie viele gleichzeitige Nutzer, welche Datenvolumina und welcher Gesamtdatenbestand)?
- Welche Rollen sind zu berücksichtigen (z. B. Administrator, Fach-Nutzer, Gast-Nutzer)
- Gibt es weitere Anforderungen (z. B. Begrenzung des Volumens, das ein Nutzer herunterladen kann)

Auch wenn generische Lösungen für das Thema Zugriffskontrolle nicht gegeben werden können, so können die nachfolgenden Beispiele als eine Auswahl möglicher Ansätze aufgefasst werden.

#### 3.4.1 Zugriffskontrolle auf Datenbankebene

Liegen die über einen SOS-Server bereitzustellenden Daten in einer Datenbank vor, welche durch den SOS-Server als Datenquelle verwendet wird, ist hierdurch eine Zugriffskontrolle umsetzbar. In diesem Fall ist es möglich, dass der Datenbank-Nutzer, welcher für den Zugriff durch eine SOS-Instanz verwendet wird, nur die Zeitreihen abfragen kann, welche auch veröffentlicht werden dürfen. Alternativ ist auch die durch manche SOS-Implementierung unterstützte Verwendung eines „Flags“ möglich, welches anzeigt, ob ein Datenbankeintrag veröffentlicht werden darf oder nicht.

Wichtig ist es dabei anzumerken, dass dieser Ansatz nicht für eine feiner gegliederte, nutzerbasierte Zugriffskontrolle gedacht ist. Stattdessen kann hierbei auf relativ einfache Weise festgelegt werden, welche Daten veröffentlicht werden dürfen.

### 3.4.2 Zugriffskontrolle auf Ebene der SOS-Implementierung

Ein anderer Ansatz zur Zugriffskontrolle besteht auf der Ebene der SOS-Implementierung. Dieser Fall ist zumeist dann passend, wenn die SOS-Schnittstelle als Erweiterung eines bereits bestehenden hydrologischen Informationssystems entwickelt werden soll, welches bereits über Zugriffskontrollmechanismen verfügt.

### 3.4.3 Zugriffskontrolle durch Fachanwendung

Wird für die Datenverwaltung eine entsprechende Fachanwendung genutzt und bringt diese Anwendung ein eigenes Nutzer- und Berechtigungssystem mit, kann diese bereits intern Berechtigungen anwenden. Vorteil dieses Verfahrens ist, dass die Auswertung auf Datenbankebene oder in einer Zwischenschicht erfolgt, was einen Gewinn bei der Performance bedeutet. Berechtigungen und Sichtbarkeiten können dabei automatisch aus einem bereits bestehenden System genutzt werden oder auch für spezielle SOS-Benutzer eingerichtet werden.

### 3.4.4 Separate Zugriffskontrolllösungen

Weiterhin besteht auch die Möglichkeit, SOS-Server über zusätzliche Zugriffskontrolllösungen abzusichern. Hierzu existieren verschiedene Lösungen, welche auf dem in Abbildung 17 dargestellten Konzept basieren. Hierbei wird der Zugriff auf einen SOS-Server nur über einen vorgelagerten Proxy, den sogenannten Policy Enforcement Point (PEP) erlaubt (z. B. über IP-Filter oder dadurch dass der SOS-Server nur innerhalb eines Intranets erreichbar ist und somit der PEP der einzige Weg ist, den SOS-Server über das Internet anzusprechen). Der PEP hat die Funktion, eingehende Anfragen von SOS-Clients zu prüfen. Anhand einer vorher durchzuführenden Nutzeranmeldung oder anhand in der Anfrage mitgelieferter Nutzerdaten werden eingehende Anfragen gefiltert: Es wird geprüft, ob der Nutzer auf den SOS-Server zugreifen darf und ob nur für den Nutzer zulässige Daten abgefragt werden. Ggf. wird die eingehende Anfrage so modifiziert, dass nur noch zulässige Daten angefordert werden. Danach wird die Anfrage an den SOS-Server weitergeleitet und die Antwort in gleicher Weise gefiltert. Zur Unterstützung des PEP, welcher die Zugriffsregeln im Datenverkehr durchsetzt, existiert ein Policy Decision Point (PDP) welcher durch den PEP genutzt wird um abzufragen, ob ein bestimmter Nutzer auf eine gegebene Ressource zugreifen darf. Um diese Anfragen zu beantworten besitzt der PDP die Fähigkeit, Zugriffsregeln (sogenannte Policies) aus einem Bestand von Regeln abzufragen und auf dieser Basis eine Entscheidung zu treffen.

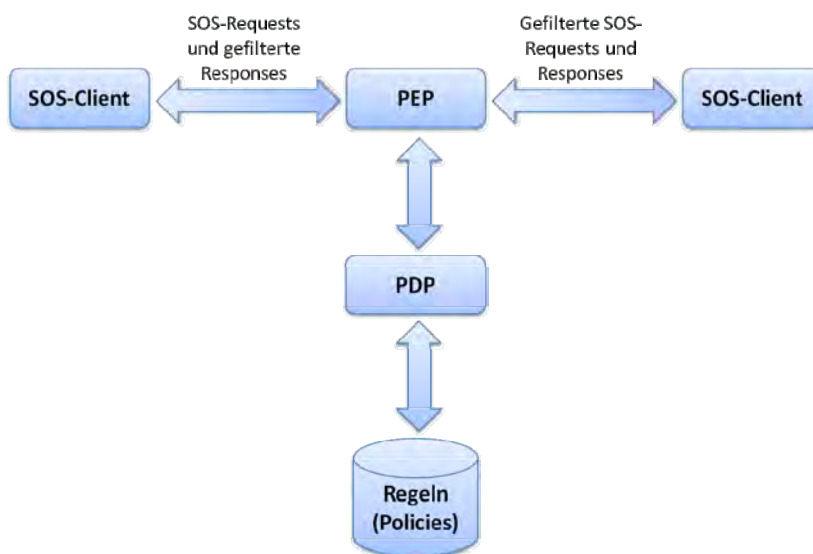


Abbildung 17: Schematische Darstellung einer Zugriffskontrolllösung

Vorteil dieses Lösungsansatzes ist, dass hiermit bestehende SOS-Installationen ohne Modifikation des SOS-Servers geschützt werden können. Allerdings ist als potentieller Nachteil darauf hinzuweisen, dass bei großen Datenmengen ggf. umfangreichere Operationen zur XML-Kodierung/Dekodierung erforderlich sind, da Anfragen an den SOS-Server sowie SOS-Antworten analysiert und ggf. geändert werden müssen. Weiterhin müssen weitere SOS-Konsumenten (z. B. externe Client-Applikationen) ebenfalls mit der Anmeldesystematik des PEP umgehen können müssen, sodass dies in verteilten Infrastrukturen zu Problemen führen kann.

## 4 Anwendungsbeispiele

Die nachfolgenden Abschnitte stellen verschiedene Beispiele für praktische SOS-Anwendungen und Implementierungen vor. Hierbei werden Lösungsmöglichkeiten für verschiedene Typen von Plattformen und Nutzern gezeigt. Dieser Überblick erhebt allerdings keinen Anspruch auf Vollständigkeit sondern dient dazu, verschiedene Möglichkeiten zur Nutzung von SOS-Servern als Datenquelle zu veranschaulichen.

### 4.1 Beispielhafte Sensor Web-Umsetzungen

#### 4.1.1 Wupperverband – FluGGS

Der Wupperverband unterhält als einer der sondergesetzlichen Wasserverbände in Nordrhein-Westfalen insgesamt rund 2.300 Kilometer Flüsse und Bäche. Außerdem betreibt er im Verbandsgebiet 14 Talsperren, 11 Klärwerke, eine Schlammverbrennungsanlage und zahlreiche weitere Anlagen wie Hochwasserrückhalte- oder Regenbecken. Verbandsmitglieder sind Städte und Gemeinden, Kreise, Wasserversorgungsunternehmen, Industrie sowie Gewerbe im Wupperebiet.

Für einen einheitlichen Zugriff auf die Geodaten im Verbandsgebiet betreibt der Wupperverband seit mehr als 15 Jahren eine verbandseigene Geodateninfrastruktur (GDI). Diese stellt für die Mitarbeiter ein wichtiges Werkzeug zur Analyse und Bearbeitung raumbezogener Fragestellungen dar. Um einen einfachen Austausch mit externen Infrastrukturen sicherzustellen, basiert die GDI im Wupperverband seit Beginn auf offenen Standards des OGC, zunächst auf den Standards für „klassische“ Geodatendienste wie z. B. dem Web Map Service (WMS) des OGC. Neben den Geobasisdaten werden zur fachlichen Analyse jedoch auch Messwerte von unterschiedlichen Sensoren benötigt, z. B. Wasserstände und Abflussmessungen, Gewässergüteparameter sowie weitere hydrologische Parameter. Ein grundlegendes Problem stellt hierbei die Speicherung und Bereitstellung der Messwerte in unterschiedlichen Fachsystemen dar, was zu einem hohen Aufwand bei der Integration zur fachlichen Analyse führt.

Daher gab der Wupperverband im Jahr 2005 zunächst eine Studie im Auftrag, um die Anwendbarkeit der Sensor Web Standards zu prüfen, die sich zu dem Zeitpunkt noch in einem frühen Entwurfsstadium befanden. Aufgrund des durchweg positiven Fazits entschloss sich der Wupperverband mit Verfügbarkeit der ersten offiziellen Sensor Web Standards im Jahr 2007, als einer der ersten Verbände die Sensor Web Standards zur Integration von Sensoren und Messwerte in die Geodateninfrastruktur zu nutzen [15]. Einen Überblick über die Sensor Web Infrastruktur des Wupperverbands zeigt Abbildung 18.

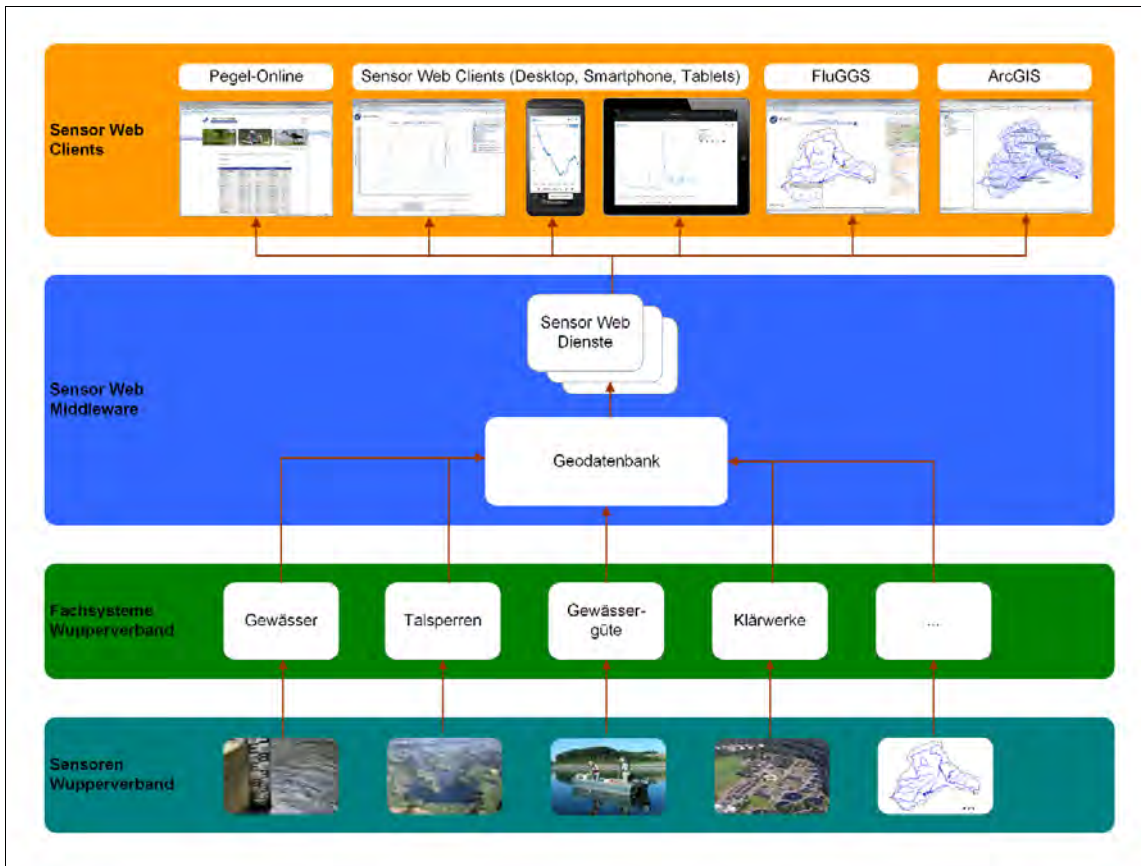


Abbildung 18: Sensor Web Infrastruktur beim Wupperverband (Quelle: Dipl.-Geogr. Christian Förster, Wupperverband)

Die zentrale Komponente zur Bereitstellung der Messwerte und Sensormetadaten in der Sensor Web Infrastruktur des Wupperverbandes ist der Sensor Observation Service, der auf dem 52°North Sensor Observation Service basiert (siehe auch Abschnitt 3.2.1.2). Allerdings werden die Messwerte der Sensoren nicht direkt über transaktionale Schnittstellen in den SOS eingefügt, sondern werden von den unterschiedlichen Fachsystemen mittels unterschiedlicher Erweiterungen in eine zentrale Geodatenbank überführt, die dann vom SOS genutzt wird. Der SOS fungiert somit als Middleware zwischen den Messreihen-Datenbanken in den Fachsystemen und der Geodateninfrastruktur bzw. externen Sensordatenklienten, indem er einen einheitlichen Zugriff auf die Messreihen und Sensorbeschreibungen bietet. Zur Analyse sind die Messreihen der Sensoren über den Sensor Web Client des Wupperverbandes für die Fachanwender und -sofern die Daten öffentlich zugänglich sind- auch für externe Nutzer verfügbar. Der Sensor Web Client basiert auf dem Helgoland Client von 52°North, der näher in Abschnitt 4.2.2.2 beschrieben ist. Abbildung 19 zeigt einen Screenshot des Sensor Web Clients des Wupperverbandes. Die Auswahl der Zeitreihen kann kartenbasiert oder über die Auswahl eines speziellen Sensors oder eines speziellen Messparameters erfolgen.

Neben dem Browser-basierten Web Client besteht der Client zusätzlich aus einer Server-Komponente die einen leichtgewichtigen Zugriff per REST API auf die Daten des Sensor Observation Services bietet. Die Daten sind dabei nicht in XML kodiert, sondern in JSON, was die Verwendung in Web Clients deutlich vereinfacht, da JSON die Default-Serialisierung von JavaScript Objekten ist und somit kein weiterer Aufwand zum Dekodieren der Daten notwendig ist. Neben den Daten im JSON-Format bietet die Server-seitige Client-Komponente auch weitere nützliche Zusatzfunktionen, wie z. B. das Rendern von Diagrammen auf Server-Seite oder das Caching von häufig verwendeten Zeitreihen.



Abbildung 19: Sensor Web Client des Wuppertalverbandes (online verfügbar unter <http://fluggs.wuppertalverband.de/swc>)

Durch den Einsatz der Sensor Web Standards des OGC wurde die Integration von Sensoren und Messreihen in die Geodateninfrastruktur des Wuppertalverbandes wesentlich erleichtert und wirtschaftlicher. So können Mitarbeiter und externe Nutzer heute mit einer intuitiv zu bedienenden Webanwendung auf aktuelle Messdaten aus vielen Fachbereichen des Wuppertalverbandes zugreifen, ohne dass sie sich mit z.T. komplexen Fachsystemen auskennen müssen. Auch die Verteilung aktueller Messdaten wurde durch den Einsatz der Sensor Web Technologie optimiert. Die Daten können direkt aus der Sensor Web Anwendung heraus für eigene, weitergehende Analysen exportiert werden, dies entlastet die datenerhebenden Fachbereiche erheblich. Wie bei einem Aufbau einer Geodateninfrastruktur müssen allerdings neben den technischen Herausforderungen auch rechtliche, organisatorische und fachliche Regelungen berücksichtigt werden. Generell ist die Sensor Web-Technik in fast allen Bereichen der Wasserwirtschaft einsetzbar. Die Technik bietet daher großes Potenzial für die Unterstützung der Fachanwender bei der Analyse, aber auch für Sensornetzbetreiber zur Überwachung eines Sensornetzwerkes. Die Sensor Web Entwicklung des Wuppertalverbandes stehen als Open Source-Software auch Dritten zur Verfügung, die die Technologien für ihre Infrastrukturen nutzen wollen.

## 4.1.2 Bundesanstalt für Gewässerkunde (BfG) – Meeresumweltdatenbank

Die Meeresumweltdatenbank (MUDAB), welche seit 1987 betrieben wird, dient der Sammlung und Bereitstellung verschiedenster Meeresumwelt-Parameter (z. B. hydrochemische und biologische Messungen/Beobachtungen). Neben dem fachlichen Wert dieser Daten, können hierüber verschiedene nationale und internationale Berichtspflichten im maritimen Bereich erfüllt werden. Die Bundesanstalt für Gewässerkunde ist für den Betrieb dieser Datenbank zuständig.

Um den interoperablen Zugriff auf die Inhalte der Meeresumweltdatenbank zu ermöglichen, hat die Bundesanstalt für Gewässerkunde eine SOS-Schnittstelle auf die Meeresumweltdatenbank aufgesetzt.

Technisch basiert die Meeresumweltdatenbank auf einer Oracle-Datenbank mit einer fachlich gewachsenen Struktur. Da bestimmte Abfragen zur Datenverfügbarkeit (d. h. insbesondere zur Generierung von SOS-GetCapabilities-Antworten) in diesem Modell nicht ohne weiteres mit hoher Geschwindigkeit durchführbar waren, wurde der in Abschnitt 3.1.7.2.1 beschriebene Ansatz gewählt. Der umgesetzte

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Datenfluss ist in Abbildung 20 dargestellt: Auf Basis einer Analyse, welche Daten der MUDAB über den SOS zu veröffentlichen sind, wurde ein Mapping zwischen den Konzepten der MUDAB und der SWE-Standards entwickelt. So entspricht z. B. der Begriff „Parameter“ in der MUDAB dem Begriff „ObservedProperty“ in der SWE-Welt. Dieses Mapping wurde genutzt, um mithilfe eines Modelltransformations-Werkzeugs (in diesem Fall FME von der Firma Safe Software) einen Transfer in das Standard-Datenmodell eines SOS-Servers (hier 52°North SOS 4.x) umzusetzen.

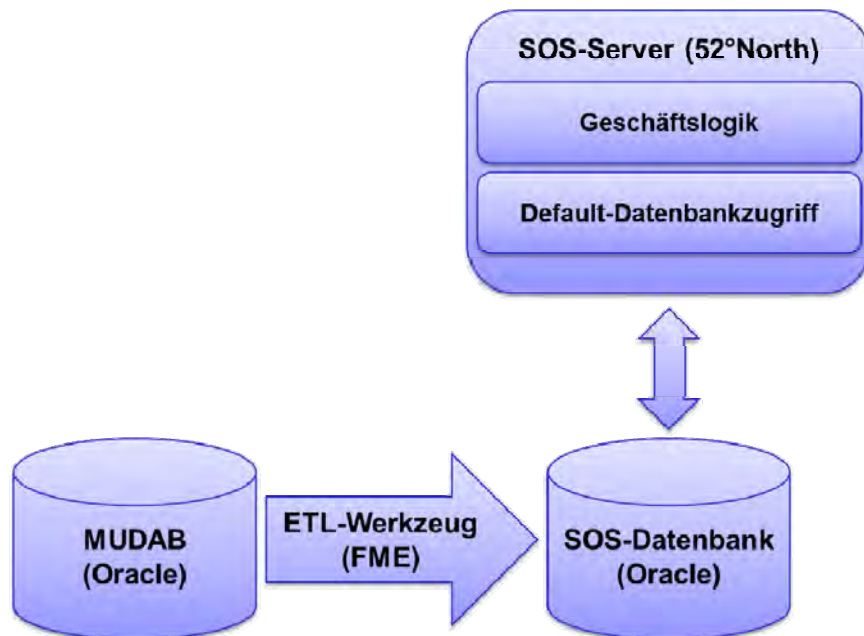


Abbildung 20: Datenfluss von der MUDAB in die SOS-Datenbank

Diese Befüllung der SOS-Datenbank kann kontinuierlich erfolgen, sodass die Daten der MUDAB über den SOS-Server veröffentlicht werden. Auf dieser Basis können dann verschiedene Client-Anwendungen erstellt werden, die in der Lage sind, SOS-Daten zu interpretieren. Die Abbildung 21 zeigt beispielsweise einen solchen Client, der im Kontext einer MUDAB-Anwendung genutzt wird.

Ein direkter Nutzer der MUDAB ist das Vorhaben „Marine Dateninfrastruktur Deutschland (MDI-DE)“. Ziel dieses Vorhabens ist der Nachweis von Daten und Informationen aus dem Küstenbereich und Meeresschutz. Hier werden die MUDAB-Daten direkt über die SOS-Schnittstelle abgezogen, um Informationen zu Messstationen und den erhobenen Parametern in verschiedenen (Karten)Produkten darzustellen. Darüber hinaus sind weitere Verbindungen zwischen den Instanzen MUDAB und MDI-DE auf Basis einer Maschine-Maschine-Kommunikation geplant.



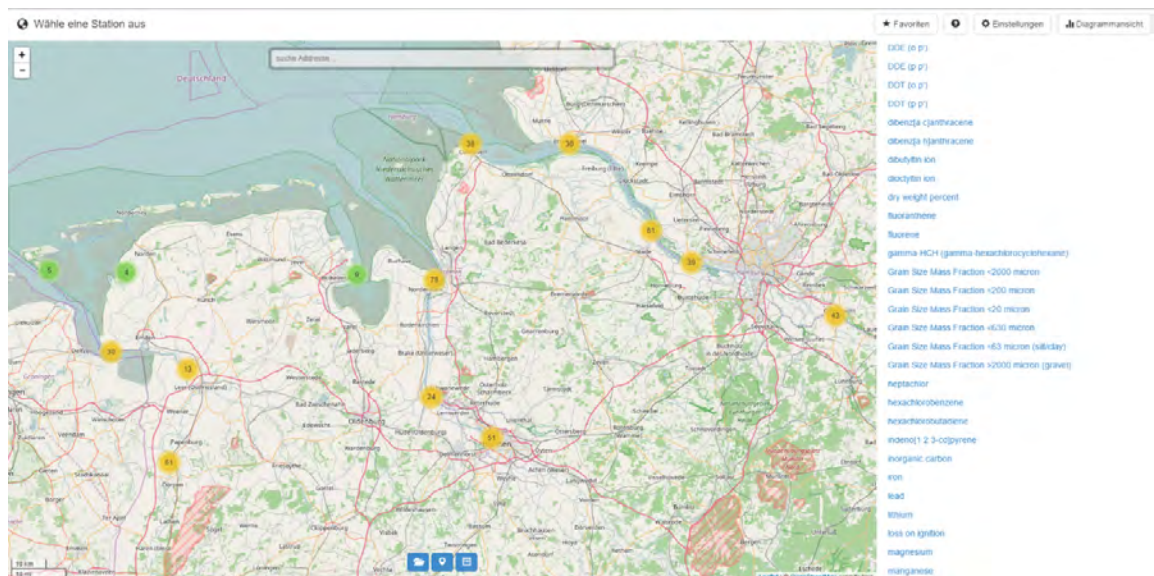


Abbildung 21: Übersicht der Inhalte der MUDAB in einem Web-Client

## 4.1.3 KISTERS

Auch die Firma KISTERS bietet Lösungen zur Unterstützung des SOS-Standards an. In den meisten Fällen wird die SOS 2.0-Schnittstelle von Kunden als zusätzlicher Dienst (zu den nativen Endpunkten) zum standardisierten Datenaustausch genutzt. Dies kann rein zum Austausch in internen Netzwerken sein, als Datengrundlage für Modellberechnungen dienen oder auch um Daten standardisiert im Internet bereitzustellen.

Beispiele für extern verfügbare SOS-Server Installationen:

1. WaterInfo.be / VMM (Vlaamse Milieumaatschappij, Belgien)

<http://tsmpub.waterinfo.be/KiWIS/KiWIS?datasource=1&request=getcapabilities&service=sos&type=queryServices&version=2.0>

2. BOM (Bureau of Meteorology, Australia, older version)

<http://www.bom.gov.au/waterdata/services?service=SOS&version=2.0&request=GetCapabilities>

3. KISTERS Demo System

<http://kiwis.kisters.de/KiWIS2/KiWIS?datasource=0&service=SOS&version=2.0&request=GetCapabilities>

## 4.1.4 PEGELONLINE

Ein weiteres Beispiel für die langjährige Anwendung der Sensor Web-Technologie ist das PEGELONLINE-System. Bereits im Jahr 2008 wurde eine erste Sensor Observation Service-Implementierung für PEGELONLINE entwickelt. Diese wurde im weiteren Verlauf kontinuierlich gepflegt und weiterentwickelt. Dieser SOS-Server stellt eine Vielzahl verschiedener Messdaten entlang der Bundeswasserstraßen für die jeweils zurückliegenden 30 Tage bereit.

Da eine Entkopplung zwischen der eigentlichen PEGELONLINE-Datenhaltung und der SOS-Entwicklung angestrebt wurde, um beispielsweise von Anpassungen im Datenbankmodell unabhängig zu werden, wurde der in Abschnitt 3.1.7.3 beschriebene Architektur-Ansatz gewählt: Der Zugriff

auf die Daten des PEGELONLINE-Systems wird nicht direkt über die Datenbank gewährt sondern über einen SOAP-basierten Web Service. Dieser Web Service wird von einem SOS-Server konsumiert, welcher als Übersetzer (Proxy) zwischen dem interoperablen SOS-Standard und der spezifischen REST-Schnittstelle fungiert (siehe Abbildung 22).

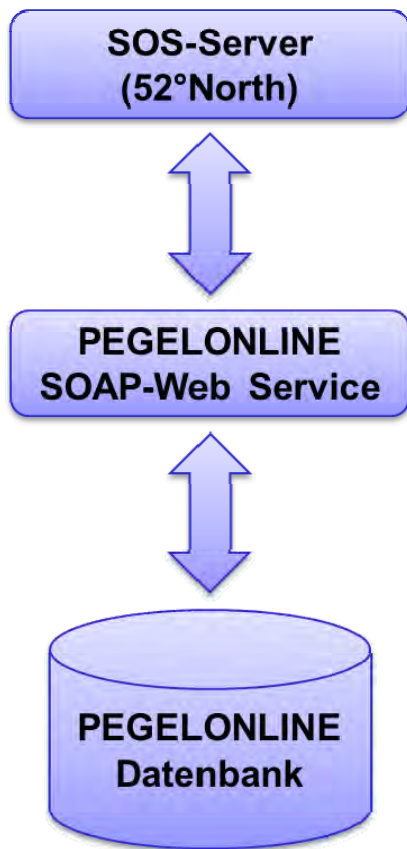


Abbildung 22: Architektur des PEGELONLINE SOS-Servers

Zum Betrieb des SOS-Servers, überprüft die SOS-Implementierung regelmäßig den PEGELONLINE SOAP Web-Service auf verfügbare Datensätze. Diese Metadaten werden gesammelt und in einem Cache des SOS-Servers abgelegt. Dies hat den Vorteil, dass der SOS-Server eigenständig Anfragen zu verfügbaren Daten beantworten kann, ohne dafür bei jeder Anfrage des PEGELONLINE SOAP Web-Service abfragen zu müssen. Trifft nun ein GetObservation-Request zu einer bestimmten Zeitreihe beim SOS-Server ein, so wird diese dynamisch vom PEGELONLINE SOAP-Web Service abgefragt, in das OGC O&M- oder WaterML 2.0-Format konvertiert (je nach Anfrage) und ausgeliefert. Im Anschluss daran kann dann z. B. eine Visualisierung in einem SOS-Client erfolgen (vgl. Abbildung 23).

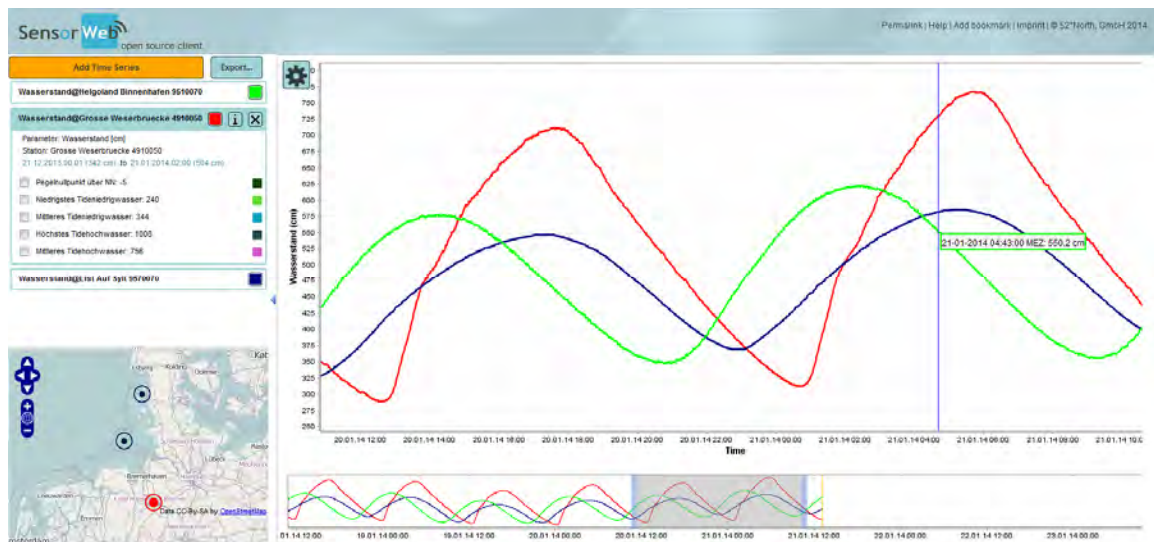


Abbildung 23: PEGELONLINE SOS-Viewer

Der PEGELONLINE SOS-Server ist öffentlich zugreifbar<sup>8</sup> und kann zur Entwicklung von Sensor Web-Anwendungen genutzt werden. Insbesondere aufgrund der Verfügbarkeit von Nah-Echtzeit-Daten bietet sich der PEGELONLINE SOS auch zur darauf aufbauenden Entwicklung von Benachrichtigungssystemen an.

## 4.1.5 Nachnutzung von Sensor Web-Diensten (Beispiel Hochwasser und Sumpfung)

Nicht immer ist erforderlich, Sensor Observation Services selbst zu implementieren. Oftmals ist es ausreichend, bestehende (ggf. externe) Sensor Web-Dienste für die optimierte Ausgestaltung eigener Geschäftsprozesse zu integrieren. Ein Beispiel zum Thema Hochwasserwarnung zeigt das Infoboard im Prozessleitstand der LINEG (Linksniederrheinische Entwässerungs-Genossenschaft), zu deren Hauptaufgaben die Sumpfung des bergbaubeeinflussten linken Niederrheins gehört.

Der Hochwasserfall tritt im Zuständigkeitsgebiet der LINEG ein, wenn hohe Wasserstände vom Rhein kommend kritische Marken im Verbandsgebiet verursachen. Das heißt die Wasserstände am Rheinpegel Ruhrort (WSA Duisburg-Rhein) können von PEGELONLINE abgefragt und als Alarmindikator für anstehende Hochwässer im Verbandsgebiet weiterverarbeitet werden.

Den einzelnen Hochwasserschutzanlagen sind spezifische Alarmstufen für Warnung und Entwarnung mit entsprechenden Schutzmaßnahmen und Benachrichtigungsempfängern zugeordnet. Werden Warn- oder Entwarnungsstufen an den Anlagen über- bzw. unterschritten, so gehen diese in der Leitzentrale ein und werden parallel als mobile Benachrichtigungen an das verantwortliche Personal und Gemeindevertreter versendet. Damit ist gewährleistet, dass alle erforderlichen Maßnahmen zur Abwehr von Hochwasserschäden getroffen werden können. Fluttore werden geschlossen, Pumpanlagen werden in Betrieb gesetzt.

Die zeitliche Entwicklung der Wasserstände am Rheinpegel wird graphisch visualisiert. Die Liste aller bestehenden Überschreitungen ist für alle Verbandsmitarbeiter über das Portal abrufbar, so dass die aktuelle hydraulische Situation im Hochwasserfall für alle einsehbar ist.

<sup>8</sup> <http://www.pegelonline.wsv.de/webservices/gis/sos?service=SOS&request=GetCapabilities>

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Pegelstand	Datum Beginn	Datum Ende	Überschreitung	Richtwert	Anlagenbezeichnung	Maßnahme
481 cm	04.03.2017 (05:00)	26.03.2017 (05:00)	X	480 cm		
532 cm	05.03.2017 (05:00)	18.03.2017 (05:00)	X	500 cm		
561 cm	10.03.2017 (05:00)	16.03.2017 (05:00)	X	550 cm		
561 cm	10.03.2017 (05:00)	16.03.2017 (05:00)	X	552 cm		
624 cm	12.03.2017 (05:00)	15.03.2017 (05:00)	X	600 cm		
624 cm	12.03.2017 (05:00)	15.03.2017 (05:00)	X	610 cm		
648 cm	13.03.2017 (05:00)	15.03.2017 (05:00)	X	630 cm		
648 cm	13.03.2017 (05:00)	17.03.2017 (05:00)	X	631 cm		
573 cm	15.03.2017 (05:00)	15.03.2017 (05:00)		590 cm		
573 cm	15.03.2017 (05:00)	15.03.2017 (05:00)		605 cm		
573 cm	15.03.2017 (05:00)	15.03.2017 (05:00)		585 cm		
524 cm	16.03.2017 (05:00)	16.03.2017 (05:00)		527 cm		
485 cm	17.03.2017 (05:00)	17.03.2017 (05:00)		500 cm		

Abbildung 24: Hochwasseralarme im LINEG Infoboard (Detail-Liste)

Die Implementierung des Infoboards erfolgte in einer verbandsweiten Portal-Anwendung auf der Grundlage von Liferay durch die ahu AG. Die Hochwasserwarnung ist eines der Portlets, die kritische Abläufe rund um die Leitzentrale implementiert. Die überwachten Abläufe dienen beispielsweise dazu, frühzeitig Hinweise auf mögliche andere Gefahren für Infrastruktur und Personal zu bekommen, um so geeignete Maßnahmen einleiten zu können.

## 4.2 Beispiele für Sensor Web Software

### 4.2.1 Server-Implementierungen

Dieser Abschnitt stellt Beispiele für verfügbare Lösungen zur Bereitstellung eines SOS-Servers vor. Wichtig ist hierbei der Hinweis, dass dies keine umfassende Auflistung ist, welche einen Anspruch auf Vollständigkeit erhebt.

#### 4.2.1.1 KISTERS KiWIS

KISTERS KiWIS (KISTERS Web Interoperability Solution) ist ein Produkt um Stamm- und Zeitreihendaten aus den kommerziellen WISKI und Hydstra Systemen über mehrere Schnittstellen gleichzeitig verfügbar zu machen. Eine dieser Schnittstellen ist der OGC SOS 2.0-Standard mit dem OGC SOS 2.0 Hydrology Profile und WaterML 2.0 als Ausgabeformat für Zeitreihendaten.

Der KiWIS Server ist in Java umgesetzt und lässt sich einfach in einem sogenannten Servlet-Container wie z. B. Apache Tomcat betreiben. Nach der Einrichtung des entsprechenden Backend-Systems als Datenquelle in KiWIS kann der SOS 2.0-Dienst einfach auf der entsprechenden Datenquelle aktiviert werden. Optional können außerdem zahlreiche Stamm- und Metadaten zur Ausgabe im SOS 2.0-Server hinzukonfiguriert werden.

Durch Streaming-Technologie und effizientes Caching kann auch auf große Datenmengen in kurzer Zeit zugegriffen werden, dazu ist die Software mit weiteren Servern skalierbar. Beispielinstallation auf einem KISTERS Demo-Server:

<http://kiwis.kisters.de/KiWIS/KiWIS?datasource=0&service=SOS&version=2.0&request=GetCapabilities>

### 4.2.1.2 52°North SOS

Beim 52°North Sensor Observation Service 4.x handelt es sich um eine Java-basierte SOS-Implementierung. Dieser SOS-Server ist unter der Open Source-Lizenz GPL v2 veröffentlicht<sup>9</sup>.

Diese SOS-Implementierung ist als eigenständiger SOS-Server konzipiert, d. h. es handelt sich um eine Applikation welche unabhängig von anderen Software-Paketen in einem Web Service-Container wie z. B. Apache Tomcat installiert werden kann.

Wichtiges Merkmal dieser SOS-Implementierung ist eine modulare Architektur welche es erlaubt, einzelne Bestandteile der SOS-Implementierung anzupassen bzw. spezifische zusätzliche Anforderungen umzusetzen. Zur Datenhaltung des 52°North SOS 4.x können verschiedene Datenbank-Managementsysteme wie z. B. Oracle, Microsoft SQL Server, PostgreSQL oder MySQL genutzt werden. Über das Hibernate-Framework, welches als Datenbank-Abstraktionsschicht genutzt wird, ist es weiterhin möglich, Mappings zu definieren, wie das interne Datenmodell des SOS-Servers auf andere relationale Datenbanken abgebildet werden kann. Dies vereinfacht die Nutzung der SOS-Implementierung mit bestehenden Messdatenbanken.

Die Konfiguration und Wartung des 52°North SOS-Server erfolgt über eine Web-Oberfläche in welcher alle wichtigen Einstellungen zum Betrieb des SOS-Servers vorgenommen werden können.

Neben den SOS-Standard-Versionen 1.0.0 und 2.0 unterstützt diese SOS-Implementierung auch das OGC SOS 2.0 Hydrology Profile und somit auch die Ausgabe von WaterML 2.0 als weiteres Datenformat neben einfachen O&M-Dateien. Somit ist die 52°North SOS-Implementierung gut für hydrologische Anwendungen geeignet.

Darüber hinaus ist die 52°North SOS-Implementierung als INSPIRE Download-Service nutzbar. Da die notwendigen Erweiterungen um einen SOS-Server als INSPIRE Download-Service nutzen zu können in einem Kooperationsprojekt mit dem JRC erarbeitet und umgesetzt wurden, erfüllt diese SOS-Implementierung bereits die Anforderungen, welche durch die sogenannte INSPIRE Technical Guidance für Download Services definiert werden. Eine aktualisierte Version dieses INSPIRE Technical Guidance-Dokuments, welche auch den SOS-Standard einschließt, wurde im Dezember 2016 veröffentlicht.

## 4.2.2 Client Implementierungen

### 4.2.2.1 SOS.js

Ein Beispiel für eine SOS-Client-Anwendung ist SOS.js<sup>10</sup>. Diese vom British Antarctic Survey entwickelte Open Source-Anwendung beruht auf JavaScript und kann innerhalb von Web Browsern genutzt werden. Hierbei handelt es sich um einen SOS-Client welcher sowohl eine Kartendarstellung der verfügbaren Messstationen (Abbildung 25) als auch eine Diagrammansicht der gemessenen Daten (Abbildung 26) bietet.

SOS.js basiert auf einer direkten Interaktion zwischen Client und SOS-Server. D. h. SOS.js verwendet direkt die SOS-Operationen, sodass die Logik zur Interaktion mit SOS-Servern sowie zur Kodierung von Anfragen/Dekodierung von Server-Antworten im Client, d. h. also im Web Browser, umgesetzt ist. Dies hat den Vorteil, dass die Interaktion zwischen Client und Server keine weiteren Komponenten erfordert und eine direkte Kommunikation aufgebaut wird. Damit ist allerdings z. B. auch verbunden, dass die Verarbeitung von SOS-Antworten (d. h. insbesondere das Parsen von XML-Dokumenten) innerhalb der Client-Anwendung zu leisten ist.

---

<sup>9</sup> <http://52north.org/sos>

<sup>10</sup> <https://github.com/paul-breen/sos-js>



## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

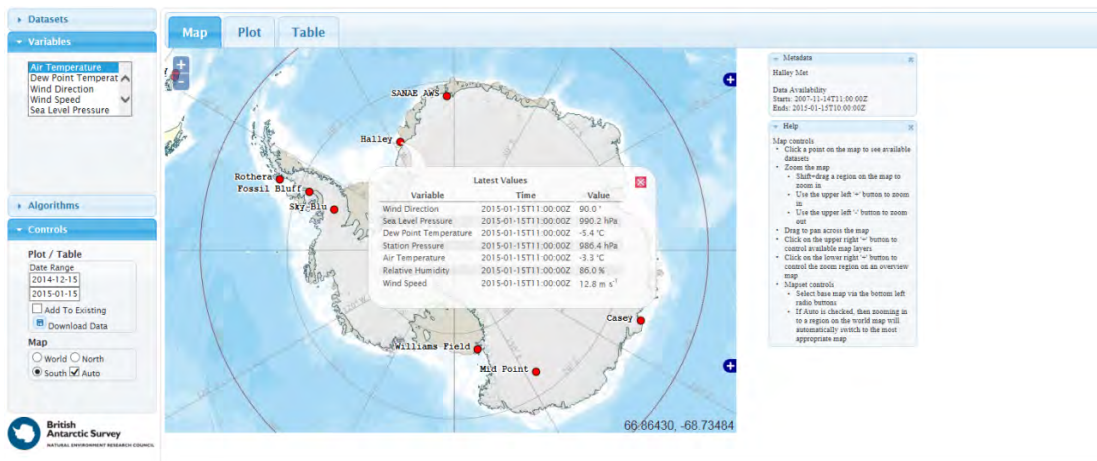


Abbildung 25: SOS.js – Beispiel für einen SOS-Client, welcher direkt mit SOS-Servern interagiert

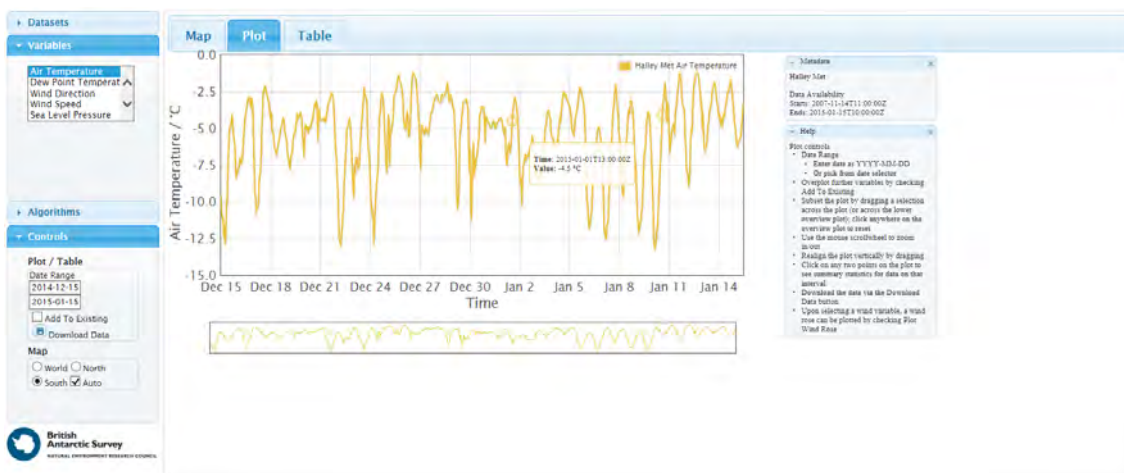


Abbildung 26: SOS.js – Diagrammdarstellung

### 4.2.2.2 52°North Helgoland

Der 52°North Helgoland, ein JavaScript SOS-Client, ist ein weiteres Beispiel für einen webbasierten SOS-Viewer. Dieser Client folgt einem sogenannten Responsive Design-Ansatz, d. h. die Erscheinung des Clients in einem Web Browser wird automatisch entsprechend des Geräts (z. B. Mobiltelefon, Tablet, Desktop-Browser) auf dem er genutzt wird angepasst. Neben einer Kartenansicht (Abbildung 28), bietet dieser Client verschiedene Formen der Messdatendarstellung wie z. B. Tabellen (Abbildung 14) und Diagramme (Abbildung 15).

Wie bereits dargestellt, ist die direkte Interaktion zwischen SOS-Server und -Client mit gewissen Nachteilen verbunden, wenn z. B. Hardware mit eingeschränkter Leistungsfähigkeit oder langsamer Internet-Anbindung genutzt wird. Um dieser Herausforderung zu begegnen wird in diesem Fall die Geschäftslogik zur Interaktion mit SOS-Servern in einer serverseitigen Komponente gekapselt. Wie in Abbildung 27 dargestellt, wird zu diesem Zweck eine Server-Komponente zwischengeschaltet, welche die Kommunikation mit dem SOS-Server übernimmt, Metadaten ggf. zwischenspeichert und dem SOS-Client eine leichtgewichtige Schnittstelle (z. B. über REST und JSON) bietet, über welche die Client-Funktionalität implementiert werden kann. Eine solche Server-Komponente kann so ausgelegt werden, dass mehrere SOS-Server gleichzeitig verwaltet werden können und die Client-Applikation die Möglichkeit hat, gezielt die Informationen abzufragen, welche für eine bestimmte Funktion erforderlich sind.

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Die Vorteile dieses Ansatzes liegen in einem deutlich geringeren Daten- und Metadatenvolumen, welches an den Client zu übertragen ist, in der Möglichkeit rechenaufwändigere Operationen wie das Parsen von (großen) XML-Dokumenten auf leistungsfähigerer Hardware durchzuführen und in der Option Metadaten über verfügbare Daten zu cachen, sodass von der Client-Anwendung schneller ermittelt werden kann, welche Daten zur Verfügung stehen. Potentielle Nachteile liegen dagegen darin, dass eine weitere Komponente erforderlich wird, dass zusätzliche, neue SOS-Server nicht direkt im Client ausgewählt werden können, sondern serverseitig konfiguriert werden müssen und dass eine Zugriffskontrolle für Messdaten nicht direkt auf dem SOS-Server aufgesetzt werden kann, sondern dass dies in Kombination mit der zusätzlichen Server-Komponente gelöst werden muss.



Abbildung 27: Server-seitige Kapselung der Geschäftslogik zur Interaktion mit SOS-Servern

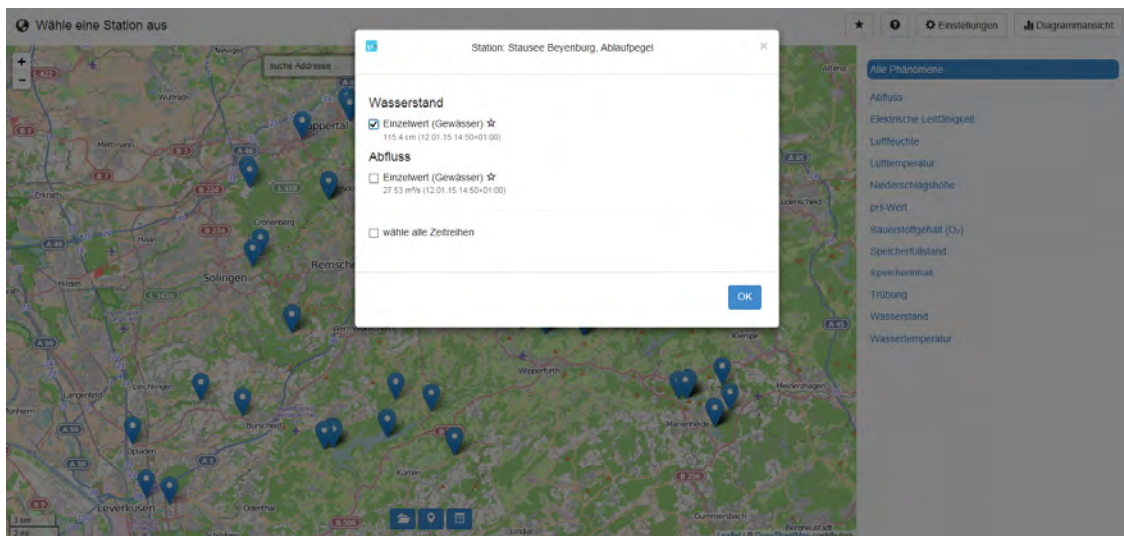


Abbildung 28: JavaScript SOS-Client – Menü zur Auswahl einer Zeitreihe

### 4.2.2.3 KISTERS Timeseries Widget

Das KISTERS Timeseries Widget ist ein Beispiel für einen Ansatz wie Zeitreihendaten eines SOS-Servers auf einfache Weise in Web-Seiten eingebunden werden können. So besteht die Möglichkeit, serverseitig vorgeordnete Grafiken (Diagrammdarstellungen) zu verwenden (Abbildung 29).

Ein Anwendungsbeispiel für diese Technologie ist das WorldWaterOnline-Portal (Abbildung 30).

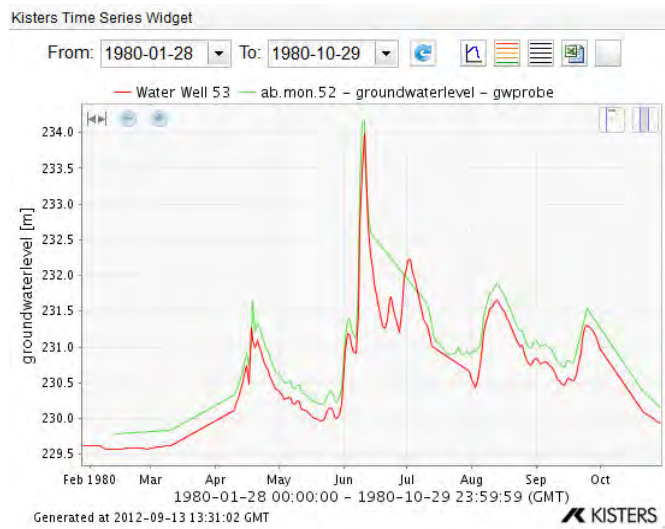


Abbildung 29: KISTERS Timeseries Widget mit serverseitig produzierten Grafiken

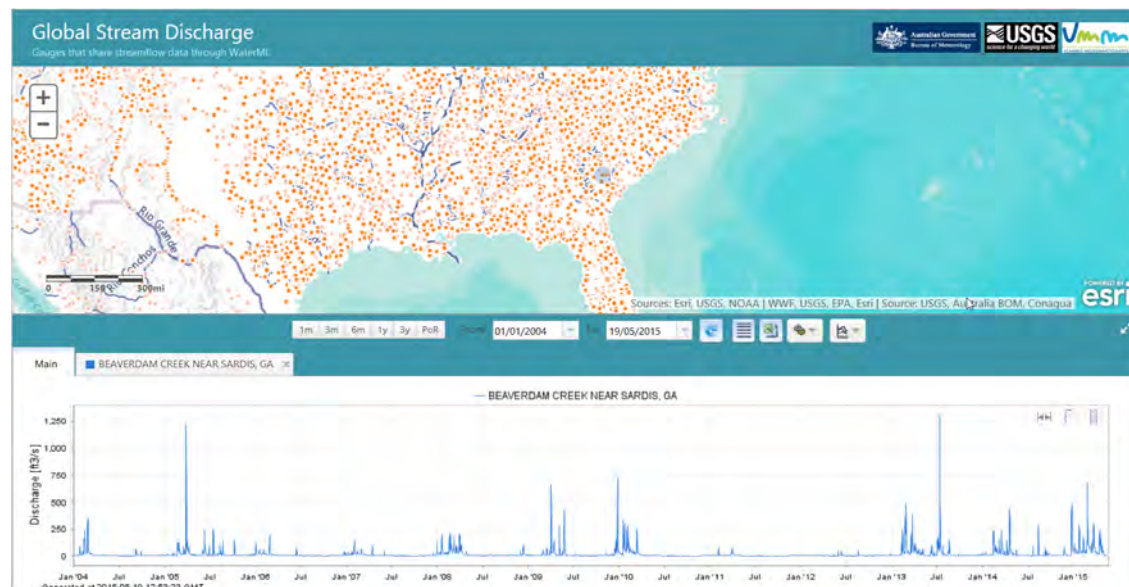


Abbildung 30: KISTERS WIDGET SOS-Implementierung für WorldWaterOnline

## 4.2.2.4 con terra map.apps SOS-Erweiterung

Das Produkt map.apps<sup>11</sup> der Firma con terra ermöglicht die einfache, fokussierte Erstellung webbasierter GIS-Funktionalitäten. Für dieses Produkt existiert eine erste Version einer SOS-Erweiterung, sodass Daten von SOS-Servern konsumiert werden können. Diese SOS-Erweiterung bietet neben Funktionen wie Karten- und Diagrammdarstellung (Abbildung 31) auch die Möglichkeit aktuelle Trends anzuzeigen, welche an Messstellen vorliegen (Abbildung 16).

Technologisch basieren sowohl map.apps als auch die SOS-Erweiterung auf JavaScript. Zur Anbindung von SOS-Servern wird die in Abschnitt 4.2.2.2 beschriebene REST-API zur Kapselung der Geschäftslogik zur Interaktion mit SOS-Servern genutzt.

11 <https://www.conterra.de/mapapps>



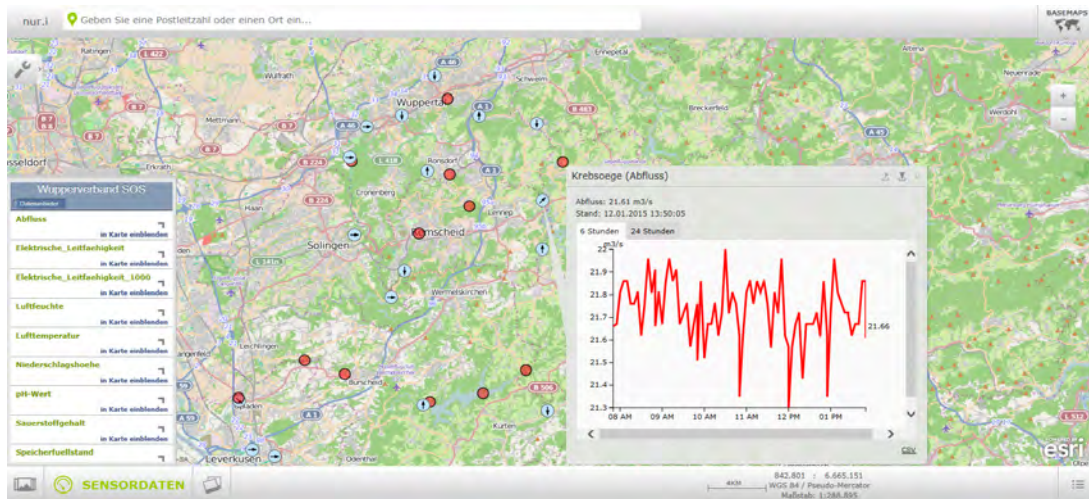


Abbildung 31: map.apps SOS-Erweiterung

## 4.2.2.5 WISKI

Das wasserwirtschaftliche Informationssystem WISKI ist mit sogenannten SOS-Consumer Klassen ausgestattet, die es ermöglichen, Zeitreihendaten von SOS-Servern direkt zu laden und im System zu speichern. Hierdurch werden qualifizierte wasserwirtschaftliche statistische Analysen möglich, die durch die Applikation WISKI bereitgestellt werden (z. B. Hochwasser-/Niedrigwasseranalysen, Mehrjahreswerte, Extremwertstatistik). Zusätzlich können SOS-Daten auch zur Ableitung weiterer Zeitreihen verwendet werden.

WISKI unterstützt die Konfiguration beliebig vieler SOS-Endpunkte und bietet darüber hinaus die Möglichkeit der interaktiven Auswahl von Datenpunkten.

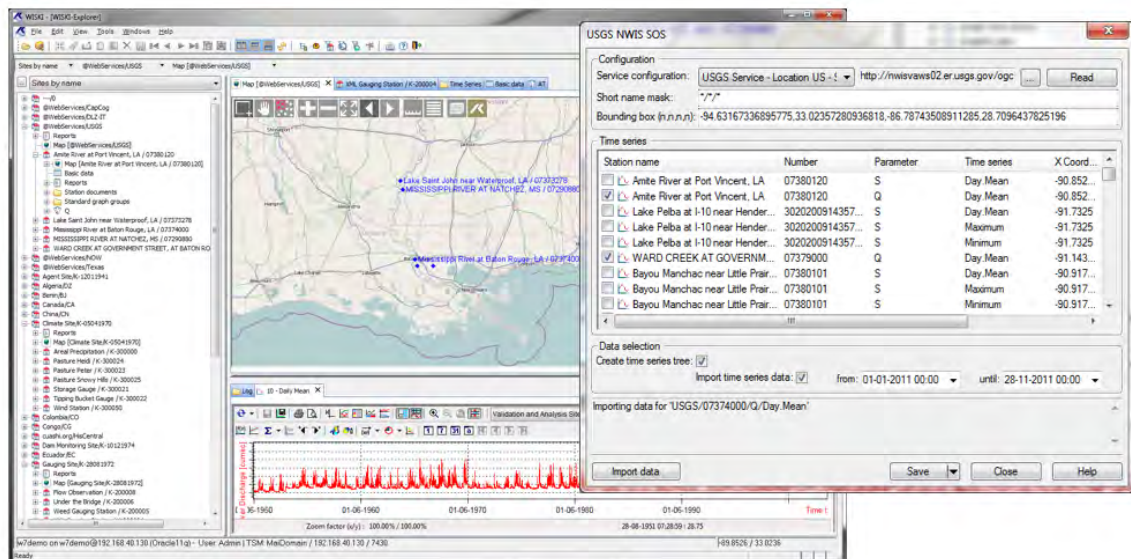


Abbildung 32: WISKI Desktop – SOS Consumer

#### 4.2.2.6 Esri ArcGIS GeoEvent Server

Auch für den Esri ArcGIS GeoEvent Server besteht eine erste, wenn auch prototypische Erweiterung um Daten von SOS-Servern zu konsumieren. Der Esri ArcGIS GeoEvent Server bietet primär die Funktionalität eingehende (Echtzeit-) Datenströme zu verarbeiten und die enthaltenen Daten (räumlich) zu analysieren. Beispielsweise können Ereignisse wie Grenzwertüberschreitungen oder räumliche Abhängigkeiten erkannt werden.

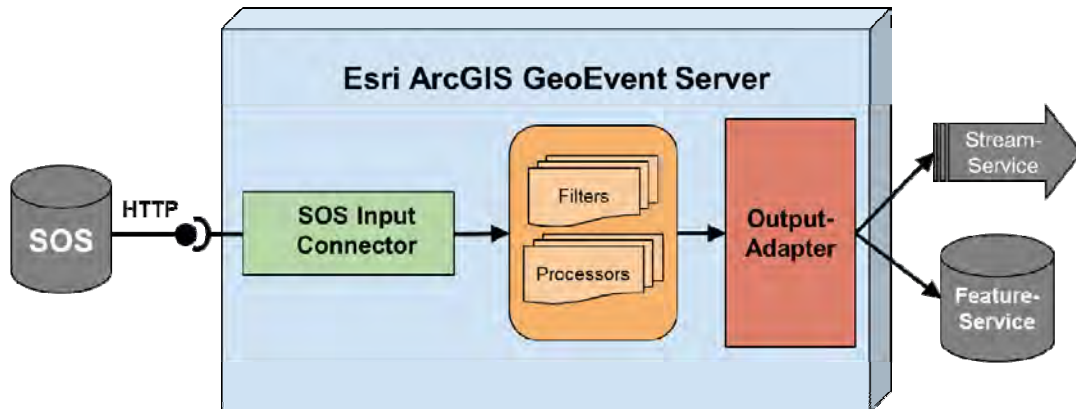


Abbildung 33: Esri ArcGIS GeoEvent Server-Workflow

Abbildung 33 zeigt den Workflow, welcher durch den Esri ArcGIS GeoEvent Server bereitgestellt wird. Über einen SOS Input Connector kann der GeoEvent Server regelmäßig einen SOS-Server auf neue Daten überprüfen. Werden neue Daten vorgefunden, so werden diese abgerufen, dekodiert und an einen Satz von Filtern und Prozessoren weitergegeben, welche entsprechend der Nutzeranforderungen konfiguriert sind (z. B. Erkennung der Überschreitung eines bestimmten Grenzwertes). Wird ein entsprechendes Ereignis erkannt, so können Output-Adapter genutzt werden um dieses Ereignis zur weiteren Nutzung weiterzugeben. Dies können entweder Datenströme zur Benachrichtigung (z. B. SMS, E-Mail) sein oder Feature-Layer (Feature-Services, Stream-Service), die von Visualisierungsapplikationen wie z. B. ArcGIS Online (vgl. Abbildung 34) verwendet werden können. Ebenso sind weitere räumliche Analysen, gerade in Kombination mit weiteren Geodatenquellen möglich, um im Falle eines bestimmten Ereignisses weitere Informationen zu generieren (z. B. Ermittlung welche Flächen von der Überschreitung eines Wasserstands-Grenzwertes betroffen sind).

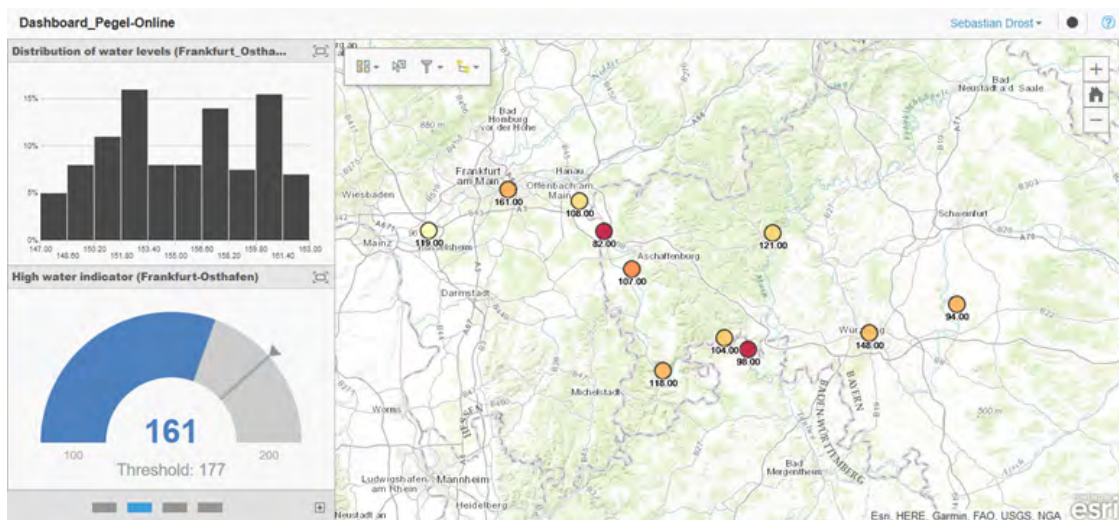


Abbildung 34: Visualisierung von Wasserständen entlang des Mains (aus PEGELONLINE) in ArcGIS Online

### 4.2.2.7 QGIS GML Application Schema Toolbox

Voraussetzung für die Nutzung von SOS-Servern als Datenquelle in einem Desktop GIS ist die Konsumierbarkeit der auf diese Weise bereit gestellten Messdaten. Wie in Kapitel 2 dargestellt, werden diese Daten in Form von XML-Dokumenten repräsentiert. Aufgrund der hierarchischen Struktur von XML kann jedes XML-Dokument als Ableitung aus einem umfangreicheren Objektmodell bzw. der zugehörigen Schemata verstanden werden. Ausgehend von der Grundidee diese Strukturiertheit und Schema-Konformität direkt für Zugriff auf und die Interaktion mit diesen Daten auszunutzen, wurde das QGIS Plugin „GML Application Schema Toolbox“<sup>12, 13</sup> entwickelt. Anstatt für bestimmte Inhalte vorprogrammiert zu sein (d. h. bestimmte vorgegebene Typen von XML-Dokumenten), sollen XML-Daten generisch und anpassbar auf Grundlage jedweden Schemas gelesen werden. Dies wurde u. a. schon erfolgreich für WaterML 2.0, GroundWaterML 2.0 sowie die INSPIRE PointTimeSeriesObservation umgesetzt und getestet.

Die QGIS GML Application Schema Toolbox bietet zwei Herangehensweisen mit unterschiedlicher Datendarstellung in QGIS. Zum einen können die Daten direkt in ihrer normalen, hierarchischen XML-Baumstruktur verwendet werden. Dabei wird jedes sogenannte „Complex Feature“ als eine Zeile eines Vektor-Layers gespeichert. Mit den üblichen QGIS Werkzeugen kann auf jeden Unterabschnitt der Baumstruktur zugegriffen werden, zudem können abhängige, externe Informationen über die „xlink:href“-Attribute abgerufen werden. Informationen aus Codelist-Registries oder zusätzliche, durch Referenzen verknüpfte Features, wie die „Observations“ von einer „Monitoring Facility“ wären Beispiele dafür.

Der andere Ansatz besteht darin, den Inhalt eines XML-Dokuments zunächst in ein relationales SQL-Modell zu überführen, welches in einer Datenbank wie PostgreSQL/PostGIS oder SpatialLite gespeichert wird, sodass QGIS auf diese Daten zugreifen kann. Dazu werden die im XML angegebenen XSD-Schemata durch den OGR GMLAS Treiber<sup>14</sup> analysiert, um aus dem zugrunde liegenden Objektmodell alle Elementtypen und Attribute sowie die Verknüpfungen zwischen den Elementen abzuleiten. Letztere werden in Beziehungen zwischen den Datenbanktabellen der Elemente umgesetzt, die mit den entsprechenden Werten aus dem XML-Dokument gefüllt werden. In QGIS lassen sich 1:N-Beziehungen zwischen Vektorlayern deklarieren, die in der Formularansicht eines Layers das Navigieren in der Modellstruktur ermöglichen. Abschließend erzeugt das Plug-In automatisch ein QGIS-Projekt mit allen geladenen Layern und Beziehungen.

Der OGR GMLAS Treiber, welcher Bestandteil der weit verbreiteten Geospatial Data Abstraction Library (GDAL) ist und die beschriebene Konvertierung übernimmt, wird anhand einer XML-Datei<sup>15</sup> konfiguriert, deren Struktur und Inhalt im zugehörigen Schema gmlasconf.xsd dokumentiert sind. Dies eröffnet weitreichende Konfigurationsmöglichkeiten durch Anpassung dieser Dateien.

---

12 [https://github.com/BRGM/gml\\_application\\_schema\\_toolbox](https://github.com/BRGM/gml_application_schema_toolbox)

13 [https://planet.qgis.org/plugins/gml\\_application\\_schema\\_toolbox](https://planet.qgis.org/plugins/gml_application_schema_toolbox)

14 [http://www.gdal.org/drv\\_gmlas.html](http://www.gdal.org/drv_gmlas.html)

15 <https://svn.osgeo.org/gdal/trunk/gdal/data/gmlasconf.xml>





Abbildung 35: QGIS-Darstellung von WaterML 2.0-Inhalten als XML-Baumstruktur

Wichtig für die Nutzung von Zeitreihendaten ist natürlich eine Diagrammdarstellung, die jedoch normalerweise nicht Teil kartenbasierter GIS ist. Für diesen Zweck ist in der QGIS GML Application Schema Toolbox ein einfaches Betrachtungsmodul für Zeitreihen von Messwerten enthalten, die der Spezifikation in WaterML 2.0 Part 1 (vgl. Abschnitt 2.4) entsprechen. Bei Vorhandensein eines „wml2:MeasurementTimeseries“-Elementes wird im XML-Modus automatisch ein Knopf zum Aufruf des Viewers angezeigt. Die Idee hinter dem Viewer ist vor allem eine API anzubieten, die von Dritten zur Erstellung von Widgets für spezielle Elementtypen genutzt werden kann. Neben weiteren, fachspezifischen Benutzeroberflächen ist jedoch insbesondere auch ein SOS-Download-Client als weitere Entwicklung vorgesehen, ähnlich dem schon vorhandenen WFS 2-Client, um nicht nur (lokal vorhandene oder via URL erreichbare) Dateien, welche Sensordaten enthalten, konvertieren zu können.

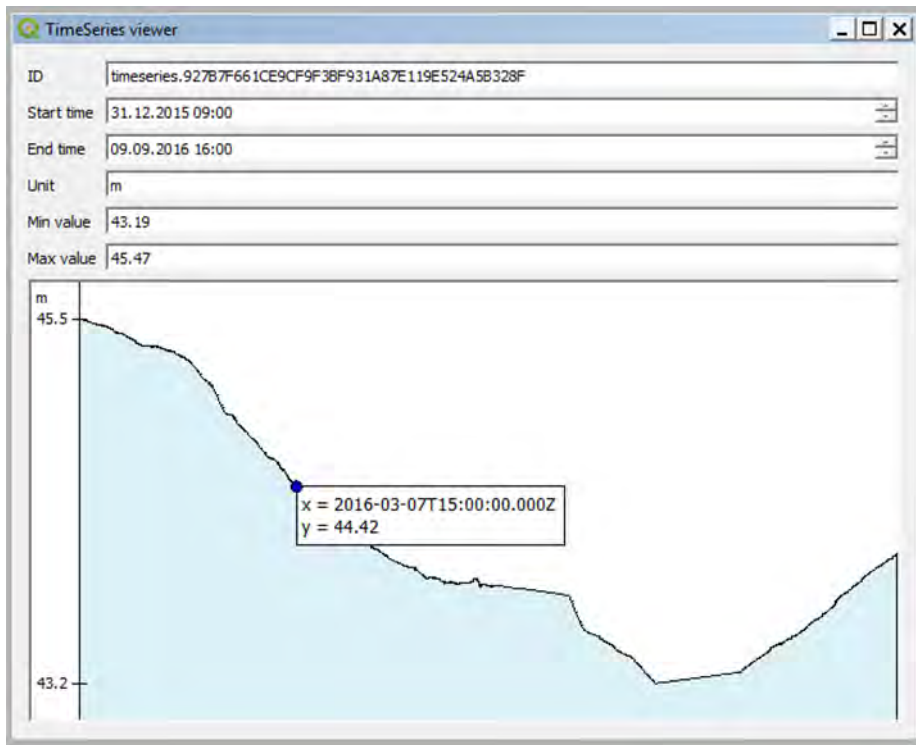


Abbildung 36: Zeitreihendarstellung in QGIS

Die hier beschriebene Software ist frei nutzbar, als Open Source-Software lizenziert und verfügt zudem über eine umfangreiche Dokumentation. Das Plug-In GML Application Schema Toolbox wurde durch BRGM (den französischen Geologischen Dienst) finanziert und durch Oslandia entwickelt. Der OGR GMLAS Treiber ist von Spatialys entwickelt worden.

### 4.2.2.8 sos4R

Neben Desktop-GIS existieren weitere Analyse-Werkzeuge mit denen Sensordaten analysiert werden können. Beispielsweise ist hier die Statistik-Softwareumgebung R<sup>16</sup> zu nennen. Während in diesem Software-Segment die Anbindung an Geodateninfrastrukturen und insbesondere an das Sensor Web bislang nur eine untergeordnete Rolle spielt, ist für das Softwarepaket R eine vergleichsweise fortgeschrittene Open Source-Entwicklung, sos4R<sup>17</sup>, verfügbar, welche den Import von Daten eines SOS-Servers ermöglicht, so das hierauf statistische Analysen durchgeführt werden können.

### 4.2.2.9 Weitere GIS-Anbindungen

Auch für weitere Desktop-GIS-Anwendungen existieren Bestrebungen, die Nutzung von SOS-Servern als Datenquelle zu ermöglichen. Somit haben GIS-Nutzer die Möglichkeit, ihre Geodaten mit Messdaten anzureichern und diese Messdaten ebenfalls in Analysen mit einzubeziehen.

Für die Desktop GIS ArcGIS Desktop, GRASS GIS sowie uDig laufen bzw. liefern bereits entsprechende Entwicklungen. Allerdings ist anzumerken, dass diese Entwicklungen bislang einen eher prototypischen Stand erreicht haben, der für eine reguläre, produktive Nutzung noch ausgebaut werden sollte.

<sup>16</sup> <http://www.r-project.org/>

<sup>17</sup> <http://cran.r-project.org/web/packages/sos4R/index.html>

# 5 Möglichkeiten und Grenzen der SOS- und WaterML 2.0-Standards

Dieses Kapitel soll einige beispielhafte, detailliertere Informationen bereitstellen, welche Datenmengen mit welcher Geschwindigkeit im Rahmen von Sensor Web-Anwendungen bereitgestellt und visualisiert werden können.

Die Möglichkeiten und Grenzen eines Web-Services zur Umsetzung des WaterML 2.0-Standards möglichst objektiv auszuloten, wurden neben theoretischen Erwägungen verschiedene IT-Teststellungen aufgebaut und in diesen Testkonstellationen technisch relevante Parameter analysiert. Als systemkritische Parameter wurden zum einen die Systemstabilität im Zusammenhang mit dem übertragenden Datenvolumen betrachtet und zum anderen die Reaktionszeit des Systems in Abhängigkeit vom zu übertragenden Datenvolumen. Die Reaktionszeit ist insbesondere im Hinblick auf die Systemakzeptanz im Zusammenspiel von Mensch und Maschine – also der Anwendungssoftware- bzw. Client-Server-Performanz – eine kritische Größe.

Im Folgenden werden ausgewählte Ergebnisse der Analyse vorgestellt. Unter anderem wurde untersucht, welche Grenzen bei der Verarbeitung langer Zeitreihen in Sensor Web-Umgebungen bestehen. Die erste Untersuchung geht der Frage nach dem SOS-Antwortverhalten nach, also wie schnell liefert ein SOS-Server nach dem WaterML 2.0-Standard in Abhängigkeit von der Länge der Zeitreihe Daten, sodass diese beim Empfänger für die Darstellung/Verarbeitung zur Verfügung stehen?

Hinzuweisen ist darauf, dass sich die Untersuchungen auf Laufzeiten konzentriert haben. Aspekte wie der Speicherverbrauch der einzelnen Komponenten wurden nicht betrachtet. Dies ist jedoch für bestimmte Aspekte ein sinnvoller Bestandteil zukünftiger Analysen (z. B. die Untersuchung des Speicherverbrauchs bei der Verarbeitung von XML-Dateien).

## 5.1 SOS-Antwortverhalten

Für die Durchführung des Tests wurde ein Auszug aus der Datenbank des Global Runoff Data Centres (GRDC) verwendet. Um die Verarbeitung möglichst langer Zeitreihen zu untersuchen, wurde eine Zeitreihe ausgewählt (Abfluss-Tagesmittelwerte für den Pegel Rees) und um 150 Jahre mit fiktiven Messdaten verlängert.

Als Software wurden folgende Komponenten eingesetzt:

- 52°North GRDC SOS 4.0.2-SNAPSHOT mit GRDC-Datenbankmodell
- Datenbank: Oracle XE 11.2.0.1.0.x86\_64
- Java 1.7\_76\_b13 (Oracle)
- Apache Tomcat 6.0.41

Der SOS-Server wurde auf folgendem System betrieben:

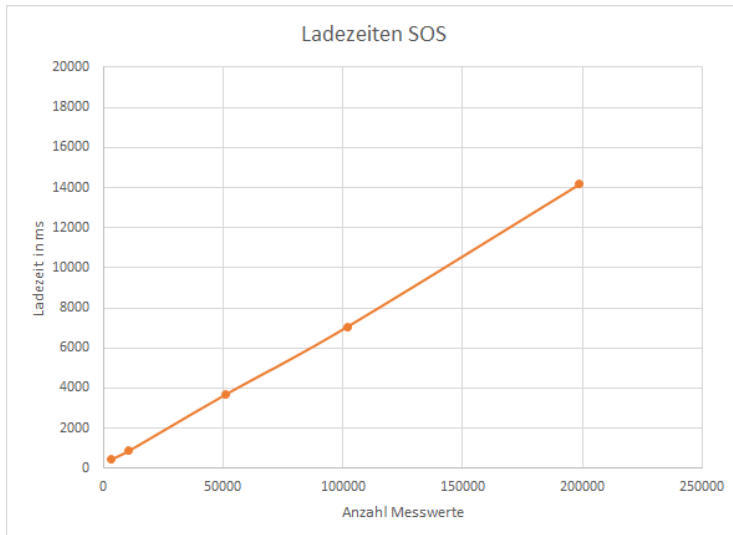
- Virtuelle Maschine mit 2 CPU-Kernen
- Betriebssystem: Debian 6 Squeeze (Linux)
- Arbeitsspeicher: 4 Gigabyte
- Festplatte: 50 Gigabyte

Die Abfragen an den SOS-Server erfolgen nicht lokal sondern über das Internet (Breitbandanbindung). Daher sind die nachfolgend beschriebenen Ergebnisse als Beispiele zu verstehen. Weitere Tests über verschiedene Internetanbindungen (z. B. auch Mobilfunknetze (GRPS, UMTS, LTE)) sind

jedoch sinnvoller Bestandteil zukünftiger Arbeiten. Insbesondere ist es wichtig, darauf hinzuweisen, dass Kriterien wie das Datenvolumen bei geringeren Bandbreiten kritischer werden.

Die Abfragen an den SOS-Server wurden automatisiert für unterschiedliche Zeiträume abgesendet und nach einer Anzahl von 10 bis 15 Wiederholungen ein Mittelwert für die Antwortzeit gebildet.

Abbildung 37 zeigt die Ergebnisse dieses Test-Szenarios.



**Abbildung 37: SOS-Ladezeiten**

Es ist zu erkennen, dass die Download-Dauer von WaterML 2.0-kodierten Daten im betrachteten Bereich tendenziell linear mit der Anzahl der Messwerte steigt. Grundsätzlich scheint mit WaterML 2.0 eine effiziente Möglichkeit bereitzustehen, Messwerte von einem SOS-Server abzurufen. Betrachtet man verschiedene Limits von Antwortzeiten und der innerhalb dieser Zeiten zu übertragenden Messwerte, so ergibt sich folgendes Bild:

- In 1 Sekunde übertragbare Messwerte:
  - WaterML 2.0: ca. 10.000 Messwerte (entsprechend ca. 27 Jahren an Tagesmittelwerten)
- In 5 Sekunden übertragbare Messwerte:
  - WaterML 2.0: ca. 70.000 Messwerte (entsprechend ca. 191 Jahren an Tagesmittelwerten)
- In 10 Sekunden übertragbare Messwerte:
  - WaterML 2.0: ca. 145.000 Messwerte (entsprechend ca. 397 Jahren an Tagesmittelwerten)

Auch wenn der Download von WaterML 2.0-Dokumenten ggf. etwas länger dauert als der Download einer anderweitig kodierten Zeitreihe, ist dennoch der Einsatz von WaterML 2.0 sehr sinnvoll, da im Gegensatz zu vielen anderen Methoden zusätzliche Metadaten für hydrologische Anwendungen geliefert werden und durch die Kodierung der Messwerte als einzelne XML-Elemente eine höhere Interoperabilität sichergestellt wird. Zudem muss man sich der Frage stellen, welche zeitkritische Anwendung benötigt mehr als 50.000 Messwerte in einem Verarbeitungsschritt. Ein einfacher Zeitreihen-Viewer wird in der Regel bereits bei deutlich weniger Daten ein graphisches Darstellungs-/Auflösungsproblem bekommen.

### 5.2 Performance beim Rendering von Zeitreihen im Client

Auf Basis des in Abschnitt 5.1 beschriebenen Versuchsaufbaus wurde zusätzlich die Darstellung von Zeitreihendaten eines SOS-Servers in einem graphischen Web-Client untersucht. Für die Durchführung der Messungen wurde die nur.i App mit SensorAccess-Bundle der con terra (auf Basis von map.apps 3.0) verwendet. Der Zugriff auf die Messdaten erfolgt über die 52°North Sensor Web Client REST-API im SOS-Proxy-Modus, sodass Metadaten über verfügbare Zeitreihen des SOS-Servers dort zwischengespeichert werden und die Dekodierung der XML-Antworten des SOS-Servers ebenfalls dort und nicht im Browser erfolgt (stattdessen erfolgt eine REST/JSON-basierte Auslieferung der Daten an den Browser).

Bei der Nutzung der Sensordaten durch einen Menschen via Grafikclient gewinnt die Software-Performance eine wesentliche Rolle. Der Fokus liegt hierbei auf der Anzeigegeschwindigkeit, d. h. der Dauer von der Auswahl einer Zeitreihe bis diese grafisch angezeigt wird. Anzeigegeschwindigkeiten von wenigen Sekunden werden als akzeptabel empfunden. Zeiten von mehr als 5 Sekunden werden bereits als kritisch eingestuft. Die Gesamtdauer bis zur Anzeige einer Zeitreihengrafik setzt sich im Wesentlichen aus der Datenladezeit, dem Datendecoding, der Datenformatierung und dem Diagramm-Rendering zusammen. Die Verwendung des WaterML 2.0-Standards nimmt maßgeblich Einfluss auf die Ladezeit und das Datendecoding. Tests haben gezeigt, dass die Ladezeit in Abhängigkeit der Länge der Zeitreihe für ca. 95% bis 99% der Gesamtdauer bis zur Grafikanzeige verantwortlich ist. Das bedeutet im Umkehrschluss zu den zuvor genannten Übertragungsraten eines SOS-Servers, dass ohne weiteres 5-stellige Werte bei den darzustellenden Messdaten in akzeptabler Anzeigegeschwindigkeit erreicht werden.

Abschließend soll auf folgende Entwicklung hingewiesen werden, welche zurzeit zwar noch nicht implementiert ist, aber zukünftig zusätzliches Potential bietet. Im Bereich von Map-Clients ist das dynamische, vorausschauende Nachladen von Daten bereits bekannt. Auch im Rahmen von Sensor-Daten ist eine solche Strategie denkbar und vorteilhaft. Werden in einem Diagramm z. B. nur wenige Werte angezeigt, kann der Client im Hintergrund bereits Daten aus benachbarten Zeiträumen (oder sofern Generalisierungsmechanismen genutzt werden für höhere Zoomstufen) laden und cachen, sodass ein flüssigeres Navigieren in der Zeitreihe möglich wird. D. h. der Client würde einen intelligenten Cache nutzen um (noch) unsichtbare Daten vorzuhalten. Dies würde eine schnelle Ansicht, Exploration und Interaktion mit den Daten ermöglichen.



## Anhang A Textbausteinvorschlage zur Verwendung in Ausschreibungen

Die nachfolgende Tabelle 6 enthalt Vorschlage fur Textbausteine, welche in Ausschreibungen im Bereich Sensor Web wiederverwendet werden konnen. Die Tabelle enthalt neben den eigentlichen Textbausteinen auch einen Verweis auf entsprechende Erklarungen sowie ggf. Angaben zur Wichtigkeit des Textbausteines.

Tabelle 6: Textbausteinvorschlage zur Verwendung in Ausschreibungen

Titel	Siehe Kapitel	Textbaustein	Bedeutung
<b>Anforderungen zur Standard-Konformitat</b>			
Konformitat zum SOS 2.0-Standard	3.1.1	Der SOS-Server muss die Conformance Class „SOS Core“ des OGC SOS 2.0-Standards erfullen.	Erforderlich
Konformitat zum OGC SOS 2.0 Hydrology Profile	3.1.1	Der SOS-Server muss die Festlegungen des OGC SOS 2.0 Hydrology Profile erfullen.	Erforderlich
Konformitat zum OGC SOS 2.0 Spatial Filtering Profile	3.1.2.6	Der zu liefernde SOS-Server muss das „Spatial Filtering Profile“ des OGC SOS 2.0-Standards unterstutzen.	Empfohlen
INSPIRE-Konformitat	3.1.3	Der zu liefernde SOS-Server muss die Empfehlungen der INSPIRE Technical Guidance Dokumente zur Nutzung des SOS-Interfaces als Download Service sowie zur Anwendung von O&M umsetzen (vgl. Technical Guidance for implementing download services using the OGC Sensor Observation Service and ISO 19143 Filter Encoding – Version 1.0 und Guidelines for the use of Observations & Measurements and Sensor Web Enablement-related standards in INSPIRE – Version 3.0).	Empfohlen
Beschreibung von Funktionen uber die Funktionalitat des SOS-Standards hinaus	3.1.5	Bitte beschreiben Sie zudem weitere Funktionen und Moglichkeiten der angebotenen Software, die uber die hier beschriebenen Anforderungen hinausgehen.	Optional
<b>Anforderung zur Datenhaltung</b>			
Keine Vorgaben zur Datenhaltung	3.1.6	Fur den zu liefernden SOS-Server wird keine spezielle Form der Datenhaltung vorgeschrieben. Der Auftragnehmer stellt sicher, dass alle benotigten Daten zeitnah und performant bereitstehen. Dies kann z. B. durch Replikation oder Importe existierender Daten geschehen. Informationen zum vorhandenen Datenbank-Modell und/oder -Schnittstelle sind in Anhang [Verweis auf Anhang] beigefugt.	Abhangig von existierender Infrastruktur

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Titel	Siehe Kapitel	Textbaustein	Bedeutung
Entwicklung eines spezifischen Datenbankzugriffs-Moduls	3.1.7.1.1	Der zu liefernde SOS-Server muss die bestehende Datenbank [Produktbezeichnung und Version] sowie das vorhandene Datenmodell als Datenquelle nutzen. Informationen zum vorhandenen Datenbank-Modell sind in Anhang [XX] beigefügt.	Abhängig von existierender Infrastruktur
Datenbank-Abstraktionsschicht	3.1.7.1.2	<p>Der zu liefernde SOS-Server muss die bestehende Datenbank [Produktbezeichnung und Version] sowie das vorhandene Datenmodell als Datenquelle nutzen. Informationen zum vorhandenen Datenbank-Modell sind in Anhang [Verweis auf Anhang] beigefügt.</p> <p>Um spätere Änderungen des Datenbankmodells flexibel, ohne Anpassungen des Quellcodes des SOS-Servers berücksichtigen zu können, ist eine Abstraktionsschicht vorzusehen, welche die Anpassung des SOS-Servers an das vorhandene Datenbank-Modell über Konfigurationsdateien (Mappings) ermöglicht.</p>	Abhängig von existierender Infrastruktur
Datenbank-Views (Sichten)	3.1.7.1.3	<p>Der zu liefernde SOS-Server muss die bestehende Datenbank [Produktbezeichnung und Version] als Datenquelle nutzen. Zur Anbindung des SOS-Servers an diese Datenbank werden in Abstimmung mit dem Auftragnehmer durch den Datenbank-Administrator des Auftraggebers Datenbank-Views erzeugt, welche die vom angebotenen SOS-Server benötigten Daten in einer durch den SOS-Server nutzbaren Struktur bereitstellen.</p> <p>Zur Definition der zu nutzenden Datenbank-Views muss der Auftragnehmer in Abstimmung mit dem Auftraggeber eine Struktur der Datenbank-Views entwickeln und dokumentieren, sodass sowohl das Erzeugen/Befüllen der Datenbank-Views als auch das Abrufen von Daten aus der Datenbank durch den SOS-Server sichergestellt werden können.</p>	Abhängig von existierender Infrastruktur

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Titel	Siehe Kapitel	Textbaustein	Bedeutung
Erstellung separater Datenbanken für SOS-Server	3.1.7.2	<p>Zur Datenhaltung ist eine separate Datenbank zu verwenden, welche als Datenquelle für den SOS-Server genutzt wird. In dieser Datenbank ist ein durch den Auftragnehmer zu spezifizierendes Datenbankmodell anzulegen. Dieses Datenmodell ist als ER-Diagramm zu dokumentieren und mit entsprechenden textuellen Erklärungen zu beschreiben.</p> <p>[und einer der beiden folgenden Textbausteine]</p> <p>Als Datenbankmanagementsystem ist [Produktbezeichnung und Version] in Verbindung mit folgenden Erweiterungen [ggf. vorhandene Erweiterungen, z. B. zur Handhabung von Geodaten] zu verwenden.</p> <p>[oder]</p> <p>Das durch den SOS-Server zu nutzende Datenbankmanagementsystem ist durch den Bieter im Angebot zu benennen (Produktname, Version, ggf. benötigte Erweiterungen).</p>	Abhängig von existierender Infrastruktur
Datenimport aus bestehender Datenbank	3.1.7.2.1	<p>Im Rahmen des Auftrags ist durch den Auftragnehmer in Abstimmung mit dem Datenbank-Administrator ein Skript oder Importer zu entwickeln, welches den automatischen Datentransfer aus der bestehenden Datenbank [Verweis auf Anhang] in die SOS-Datenbank ermöglicht.</p> <p>Dieser Datentransfer soll [automatisch beim Vorliegen neuer Daten]/[regelmäßig im Zeitabstand von [Zeitangabe]] erfolgen.</p> <p>[oder]</p> <p>Im Rahmen des Projekts wird durch den Datenbank-Administrator des Auftraggebers ein Skript oder Importer entwickelt, welches den automatischen Datentransfer aus der bestehenden Datenbank [Verweis auf Anhang] in die SOS-Datenbank ermöglicht. Um diese Entwicklung zu unterstützen ist durch den Auftragnehmer neben der Dokumentation des SOS-Datenbankmodells eine Beratung des Datenbank-Administrators vorzusehen.</p>	Abhängig von existierender Infrastruktur

Titel	Siehe Kapitel	Textbaustein	Bedeutung
Verwendung transaktionaler SOS-Operationen zum Datenimport	3.1.7.2.2	<p>Durch den Auftragnehmer ist ein Mechanismus zu entwickeln, über welchen die SOS-Datenbank gefüllt wird. Dieser Mechanismus muss die transaktionalen Operationen des SOS 2.0-Standards (InsertSensor und InsertObservation) verwenden. Die durch den zu entwickelnden Mechanismus in die SOS-Datenbank zu transferierenden Daten sind aus folgender Quelle zu beziehen: [Verweis auf Anhang].</p> <p>Der Import-Mechanismus soll [manuell durch einen Mitarbeiter des Auftraggebers erfolgen/automatisch im Abstand von [Zeitangabe] erfolgen, sofern neue Daten vorliegen].</p>	Abhängig von existierender Infrastruktur
Proxy-Lösungen	3.1.7.3	<p>Der zu liefernde SOS-Server muss den bestehenden Datenzugriffsdienst als Datenquelle nutzen. Daher ist ein Datenzugriffsmodul für den SOS-Server zu entwickeln, welches die zur Beantwortung eingehender SOS-Anfragen notwendigen Daten vom Datenzugriffsdienst abfragt.</p> <p>Die Schnittstelle des zu nutzenden Datenzugriffsdienstes ist in [Referenz auf Anhang] beschrieben.</p> <p>[und optional]</p> <p>Um die geforderten Leistungsdaten zu erreichen und die Last auf den bestehenden Datenzugriffsdienst zu beschränken, ist durch den Auftraggeber zu prüfen, ob ein entsprechender Caching-Mechanismus zu entwickeln ist, welcher Daten- bzw. Metadaten aus dem Datenzugriffsdienst zur schnelleren Beantwortung von SOS-Anfragen zwischenspeichert.</p> <p>Dieser Caching-Mechanismus soll gleichzeitig so ausgelegt werden, dass neue Zeitreihen nach spätestens [Zeitangabe] im Cache des SOS-Servers verfügbar sind und bei SOS-Anfragen ausgeliefert werden können.</p> <p>Weiterhin ist eine Persistierung des Caches zu prüfen, um einen Neuaufbau des Caches beim Neustart des SOS-Servers nicht zwingend erforderlich zu machen.</p>	Abhängig von existierender Infrastruktur
<b>Nicht-funktionale Anforderungen</b>			

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Titel	Siehe Kapitel	Textbaustein	Bedeutung
Systemumgebung	3.1.6	<p>Das Sensor Web-System ist in folgender Umgebung zu betreiben:</p> <ul style="list-style-type: none"> <li>– [Betriebssystem]</li> <li>– [Ggf. vorhandene Einschränkungen bzgl. notwendiger Software-Grundlagen (z. B. verfügbare Laufzeitumgebungen (beispielsweise Limitierungen von Java-Versionen), Beschränkung auf bestimmte Web-Server-Implementierungen).]</li> <li>– [Hardware (CPU/Arbeitsspeicher/ Festplatte)]</li> </ul>	Erforderlich
Remote-Verbindung	3.1.6	<p>[Für die Installation/Wartung des Sensor Web-Systems steht dem Auftragnehmer eine Remote-Verbindung zum Rechner, auf dem der Sensor Web-Server betrieben werden soll, bereit.]/[Eine Remote-Verbindung zum Rechner, auf dem der Sensor Web-Server betrieben werden soll, kann nicht bereitgestellt werden. Daher sind Installation/Wartung vor Ort durchzuführen.]</p>	Empfohlen
Dokumentation	3.3.1	<p>Die Sensor Web-Komponenten sind mit folgender Dokumentation bereitzustellen:</p> <ul style="list-style-type: none"> <li>– [Installations- und Betriebsanleitung einschließlich einer Dokumentation der einstellbaren Parameter des SOS-Servers]</li> <li>– [Dokumentation der Systemarchitektur einschließlich der durch den SOS-Server genutzten [Datenbankanbindung/ Datenbankmodelle/Mappings zwischen SOS-Modell und bestehender Datenbank].]</li> <li>– [Benutzerdokumentation für einen ggf. zu liefernden SOS-Client (z. B. in Form eines PDF-Handbuchs, einer Online-Hilfe oder interaktiver Lernhilfen)]</li> </ul>	Erforderlich

Titel	Siehe Kapitel	Textbaustein	Bedeutung
Leistungsanforderungen	3.3.3	<p>Die über den SOS-Server bereitzustellenden Daten haben folgenden Umfang:</p> <ul style="list-style-type: none"> <li>– [Ungefähre Anzahl der Messungen (Größenordnung)]</li> <li>– [Ungefähre Anzahl der Zeitreihen (Größenordnung)]</li> <li>– [Ungefähre Anzahl der Messstationen (Größenordnung)]</li> <li>– [Ungefähre Anzahl der gemessenen Parameter (Größenordnung)]</li> </ul> <p>Weiterhin bestehen folgende Leistungsvorgaben:</p> <ul style="list-style-type: none"> <li>– Ungefähre Abschätzung, wie viele Anfragen pro Zeitperiode (z. B. Anfragen pro Stunde) zu erwarten sind</li> <li>– Anforderungen an die Antwortgeschwindigkeit (z. B. wie lange darf die Auslieferung einer bestimmten Anzahl an Messungen dauern)</li> </ul>	Erforderlich
Skalierbarkeit	3.3.4	<p>Der SOS-Server und die zugehörige Datenthaltung müssen mindestens [xx] Anfragen pro Stunde mit einer mittleren Wertezahl von [yy] beantworten können. Bitte beschreiben Sie das Skalierungskonzept. Falls vorhanden, benennen Sie existierende Systeme mit ihren jeweiligen Lastkennzahlen.</p>	Empfohlen
Ausfallsicherheit	3.3.5	<p>Ein lastfallgerechter Betrieb des SOS-Servers soll ohne größere Ausfallzeiten kontinuierlich erfolgen. Es wird eine Ausfallsicherheit von [n]% erwartet. Bitte stellen Sie das Konzept zur Ausfallsicherheit dar und benennen Sie beispielsweise die Mechanismen zur Selbstüberwachung.</p>	Optional
Software Erstellungs- und Supportkonzept	3.3.6	<p>Der Bieter wird aufgefordert sein Supportkonzept darzustellen, aus dem das Antwortzeitverhalten und die Erreichbarkeit des Bieters im Unterstützungsfall hervorgehen. Je nach Auswirkung von Softwarefehlern werden Fehlerbehebungszeiten vereinbart:</p> <ul style="list-style-type: none"> <li>– Schweregrad [1]: Fehlerbehebung in [m] [Stunden/ Tage]</li> <li>– Schweregrad [2]: Fehlerbehebung in [n] [Stunden/ Tage]</li> <li>– Schweregrad [x]: Fehlerbehebung in [o] [Stunden/ Tage]</li> </ul>	Empfohlen

## Anhang B Beispiele für Sensor Web-Datenformate

### B.1 SOS-Antwort: Messdaten als OGC O&M-Observation

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:sos="http://www.opengis.net/sos/2.0"
  xmlns:om="http://www.opengis.net/om/2.0"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.opengis.net/om/2.0
    http://schemas.opengis.net/om/2.0/
      observation.xsd
    http://www.opengis.net/gml/3.2
    http://schemas.opengis.net/gml/3.2.1/
      gml.xsd
    http://www.w3.org/2003/05/soap-envelope
    http://www.w3.org/2003/05/soap-envelope
    http://www.opengis.net/sos/2.0
    http://schemas.opengis.net/sos/2.0/
      sosGetObservation.xsd">
  <soap:Header/>
  <soap:Body>
    <!--Alle Messungen, welche von einer Sensor Observation Service-
      Instanz im Rahmen einer Antwort geliefert werden, sind in einem
      GetObservationResponse-Element verpackt.-->
    <sos:GetObservationResponse>
      <!--Ein observationData-Element enthält eine einzelne Messung.-->
      <sos:observationData>
        <!--Der Identifier der kodierten Messung.-->
        <om:OM_Observation gml:id="o_18858013">
          <!--Der Typ der Messung. OM_Measurement steht hierbei für
            eine Messung die einen skalaren Wert als Ergebnis
            liefert.-->
          <om:type xlink:href="http://www.opengis.net/def/
            observationType/OGC-OM/2.0/
            OM_Measurement"/>
          <!--Der Zeitpunkt, für welchen die Messung Daten enthält.-->
          <om:phenomenonTime>
            <gml:TimeInstant gml:id="phenomenonTime_18858013">
              <gml:timePosition>
                2013-06-02T00:00:00.000Z
              </gml:timePosition>
            </gml:TimeInstant>
          </om:phenomenonTime>
          <!--Der Zeitpunkt, an dem die Messung veröffentlicht wurde.
            Da dieser Wert meist nicht explizit bekannt ist, wird
            hier auf die phenomenonTime verwiesen.-->
          <om:resultTime xlink:href="#phenomenonTime_18858013"/>
          <!--Der Messprozess (z. B. Sensor, Algorithmus) über
            welchen, der Messwert erzeugt wurde. Einzelwert bedeutet
            in diesem Fall, dass es sich um einzelne Messwerte und
            keine aggregierten Messungen handelt.-->
          <om:procedure xlink:href="Einzelwert"/>
          <!-- Die beobachtete Messgröße, hier "Wasserstand".-->
          <om:observedProperty xlink:href="Wasserstand"/>
          <!--Der Standort, an dem die Messung vorgenommen wurde. In
            diesem Fall wird nur auf den Namen der Station
            verwiesen. Alternativ könnten an dieser Stelle auch die
            zugehörigen Koordinaten mitgeliefert werden.-->
          <om:featureOfInterest xlink:href="Krebssoege"
            xlink:title="Krebsöge"/>
          <!--Der gemessene Wert zusammen mit der Maßeinheit.-->
          <om:result xmlns:ns="http://www.opengis.net/gml/3.2"
            uom="cm" xsi:type="ns:MeasureType">
```

```

        83.428788
      </om:result>
    </om:OM_Observation>
  </sos:observationData>
  <!-- Ab hier kann eine beliebige Anzahl weiterer observationData-
    Elemente in der gleichen Struktur wie zuvor folgen. -->
</sos:GetObservationResponse>
</soap:Body>
</soap:Envelope>

```

## B.2 SOS-Antwort: Messdaten als WaterML 2.0 Time Value Pair-Observation

```

<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:sos="http://www.opengis.net/sos/2.0"
  xmlns:om="http://www.opengis.net/om/2.0"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:wml2="http://www.opengis.net/waterml/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.opengis.net/om/2.0
    http://schemas.opengis.net/om/2.0/
      observation.xsd
    http://www.opengis.net/waterml/2.0
    http://schemas.opengis.net/waterml/2.0/
      waterml2.xsd
    http://www.opengis.net/gml/3.2
    http://schemas.opengis.net/gml/3.2.1/
      gml.xsd
    http://www.w3.org/2003/05/soap-envelope
    http://www.w3.org/2003/05/soap-envelope
    http://www.opengis.net/sos/2.0
    http://schemas.opengis.net/sos/2.0/
      sosGetObservation.xsd">
  <soap:Header/>
  <soap:Body>
    <!--Alle Messungen, welche von einer Sensor Observation Service-
      Instanz im Rahmen einer Antwort geliefert werden, sind in einem
      GetObservationResponse-Element verpackt.-->
    <sos:GetObservationResponse>
      <sos:observationData>
        <!--Ein observationData-Element enthält in diesem Fall eine
          Zeitreihe.-->
        <om:OM_Observation gml:id="o_18858013">
          <om:phenomenonTime>
            <!--Zeitperiode, für welchen die Zeitreihe Daten
              enthält.-->
            <gml:TimePeriod gml:id="phenomenonTime_18858013">
              <gml:beginPosition>
                2013-06-02T00:00:00.000Z
              </gml:beginPosition>
              <gml:endPosition>
                2013-06-03T00:00:00.000Z
              </gml:endPosition>
            </gml:TimePeriod>
          </om:phenomenonTime>
          <!--Der Zeitpunkt, an dem die Messung veröffentlicht wurde.
            In diesem Fall wird auf den Zeitstempel der letzten
            Messung der Zeitreihe verwiesen.-->
          <om:resultTime>
            <gml:TimeInstant gml:id="ti_B41829D0775852CC1CF1FC8D0B33B
              F6C728A3773">
              <gml:timePosition>
                2013-06-03T00:00:00.000Z
              </gml:timePosition>
            </gml:TimeInstant>

```



```

</om:resultTime>
<!--Der Messprozess (z. B. Sensor, Algorithmus) über
welchen, der Messwert erzeugt wurde. Einzelwert bedeutet
in diesem Fall, dass es sich um einzelne Messwerte und
keine aggregierten Messungen handelt.-->
<om:procedure>
  <wml2:ObservationProcess gml:id="process.CE870EA18969FA52
    D9401BB3CA4A7E564FFED2F
    9">
    <!--Identifizier des Messprozesses, der den Wert
    geliefert hat.-->
    <gml:identifizier codeSpace="http://www.opengis.net/def/
      nil/OGC/0/unknown">
      Einzelwert
    </gml:identifizier>
    <!--Typ des Messprozesses, der den Wert geliefert hat.
    In diesem Fall handelt es sich um einen Sensor.-->
    <wml2:processType xlink:href="http://www.opengis.net/
      def/waterml/2.0/
      processType/
      Sensor"/>
  </wml2:ObservationProcess>
</om:procedure>
<!-- Die beobachtete Messgröße, hier "Wasserstand".-->
<om:observedProperty xlink:href="Wasserstand"/>
<!--Der Standort, an dem die Messung vorgenommen wurde. In
diesem Fall wird nur auf den Namen der Station
verwiesen. Alternativ könnten an dieser Stelle auch die
zugehörigen Koordinaten mitgeliefert werden.-->
<om:featureOfInterest xlink:href="Krebssoege"
  xlink:title="Krebsöge"/>
<om:result>
  <!--Das result-Element enthält eine Zeitreihe kodiert als
  WaterML 2.0 MeasurementTimeseries-Element.-->
  <wml2:MeasurementTimeseries gml:id="timeseries.18858013">
    <!--Allgemeine Metadaten zur kodierten Zeitreihe.-->
    <wml2:metadata>
      <!--Zeitliche Ausdehnung der Zeitreihe. In diesem
      Fall wird auf das oben dargestellte
      phenomenonTime-Element verwiesen werden.-->
      <wml2:MeasurementTimeseriesMetadata>
        <wml2:temporalExtent xlink:href="#phenomenon
          Time_
          18858013"/>
      </wml2:MeasurementTimeseriesMetadata>
    </wml2:metadata>
    <!--Informationen, die für alle Elemente der kodierten
    Zeitreihe gleichermaßen gelten.-->
    <wml2:defaultPointMetadata>
      <wml2:DefaultTVPMeasurementMetadata>
        <!--Die Maßeinheit, in welcher die Messwerte
        angegeben sind.-->
        <wml2:uom code="cm"/>
        <!--Angabe zur Art des beobachteten Phänomens
        (z. B. augenblickliche Messung, Durchschnitt
        über Zeitperiode)-->
        <wml2:interpolationType
          xlink:href="http://www.opengis.net/def/
            timeseriesType/
            WaterML/2.0/
            continuous"
          xlink:title="Instantaneous"/>
      </wml2:DefaultTVPMeasurementMetadata>
    </wml2:defaultPointMetadata>
    <!--Ein Paar aus Zeitpunkt und Messwert wird als
    WaterML 2.0 point-Element kodiert.-->
    <wml2:point>
      <wml2:MeasurementTVP>

```

```

        <!--Der Zeitpunkt, auf den sich der Messwert
             bezieht.-->
        <wml2:time>2013-06-02T00:00:00.000Z</wml2:time>
        <!--Der Messwert.-->
        <wml2:value>83.428788</wml2:value>
    </wml2:MeasurementTVP>
</wml2:point>
<!-- Ab hier kann eine beliebige Anzahl weiterer
     point-Elemente in der gleichen Struktur wie zuvor
     folgen. -->
<wml2:point>
    <wml2:MeasurementTVP>
        <wml2:time>2013-06-02T03:20:00.000Z</wml2:time>
        <wml2:value>83.49418</wml2:value>
    </wml2:MeasurementTVP>
</wml2:point>
<wml2:point>
    <wml2:MeasurementTVP>
        <wml2:time>2013-06-02T03:40:00.000Z</wml2:time>
        <wml2:value>83.198565</wml2:value>
    </wml2:MeasurementTVP>
</wml2:point>
    <!--...-->
</wml2:MeasurementTimeseries>
</om:result>
</om:OM_Observation>
</sos:observationData>
</sos:GetObservationResponse>
</soap:Body>
</soap:Envelope>

```

### B.3 SOS-Antwort: Messdaten als INSPIRE PointTimeSeriesObservation

```

<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:sos="http://www.opengis.net/sos/2.0"
  xmlns:om="http://www.opengis.net/om/2.0"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:wml2="http://www.opengis.net/waterml/2.0"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.opengis.net/om/2.0
    http://schemas.opengis.net/om/2.0/
    observation.xsd
    http://www.opengis.net/waterml/2.0
    http://schemas.opengis.net/waterml/2.0/
    waterml2.xsd
    http://www.opengis.net/gml/3.2
    http://schemas.opengis.net/gml/3.2.1/
    gml.xsd
    http://www.w3.org/2003/05/soap-envelope
    http://www.w3.org/2003/05/soap-envelope
    http://inspire.ec.europa.eu/schemas/
    omso/3.0
    http://inspire.ec.europa.eu/schemas/
    omso/3.0/SpecialisedObservations.xsd
    http://www.opengis.net/sos/2.0
    http://schemas.opengis.net/sos/2.0/
    sosGetObservation.xsd">
  <soap:Header/>
  <soap:Body>
    <!--Alle Messungen, welche von einer Sensor Observation Service-
         Instanz im Rahmen einer Antwort geliefert werden, sind in einem
         GetObservationResponse-Element verpackt.-->
    <sos:GetObservationResponse>

```

```

<!--Ein observationData-Element enthält in diesem Fall eine
Zeitreihe.-->
<sos:observationData>
  <!--PointTimeSeriesObservation ist der im Rahmen von INSPIRE
empfohlene Typ zur Kodierung von Zeitreihen-->
  <omso:PointTimeSeriesObservation xmlns:xlink="http://www.w3.org
/1999/xlink"
xmlns:om="http://www.opengis.
net/om/2.0"
xmlns:gml="http://www.opengis.
net/gml/3.2"
xmlns:wml2="http://www.opengis.
net/waterml/2.0"
xmlns:omso="http://inspire.ec.
europa.eu/schemas/
/omso/3.0"
gml:id="o_1">
  <!--Identifizier zur Referenzierung der Zeitreihe-->
  <gml:identifizier codeSpace="http://www.opengis.net/def/nil/
OGC/0/unknown">
    http://example/obs/WaterLevel/1258154984/51545
  </gml:identifizier>
  <!--Angabe des INSPIRE Observation-Typs-->
  <om:type xlink:href="http://inspire.ec.europa.eu/
featureconcept/
PointTimeSeriesObservation"/>
  <om:phenomenonTime>
    <!--Zeitperiode, für welchen die Zeitreihe Daten
enthält.-->
    <gml:TimePeriod gml:id="phenomenonTime_18858013">
      <gml:beginPosition>
        2013-06-02T00:00:00.000Z
      </gml:beginPosition>
      <gml:endPosition>
        2013-06-03T00:00:00.000Z
      </gml:endPosition>
    </gml:TimePeriod>
  </om:phenomenonTime>
  <!--Der Zeitpunkt, an dem die Messung veröffentlicht wurde.
In diesem Fall wird auf den Zeitstempel der letzten
Messung der Zeitreihe verwiesen.-->
  <om:resultTime>
    <gml:TimeInstant gml:id="ti_B41829D0775852CC1CF1FC8D0B33B
F6C728A3773">
      <gml:timePosition>
        2013-06-03T00:00:00.000Z
      </gml:timePosition>
    </gml:TimeInstant>
  </om:resultTime>
  <!--Der Messprozess (z. B. Sensor, Algorithmus) über
welchen, der Messwert erzeugt wurde. In diesem Fall eine
Referenz über welche die Sensorbeschreibung zugegriffen
werden kann.-->
  <om:procedure xlink:href="http://example/sensors/
1r38z92.xml"
xlink:title="Water gauge"/>
  <!--Mit Hilfe dieses Parameters wird auf die zugehörige
INSPIRE Environmental Monitoring Facility oder das
INSPIRE Environmental Monitoring Network verwiesen,
welche/welches den Messwert geliefert hat.-->
  <om:parameter>
    <om:NamedValue xmlns:om="http://www.opengis.net/om/2.0"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance">
      <om:name xlink:href="RelatedMonitoringFeature"/>
      <om:value xmlns:ns="http://www.opengis.net/gml/3.2"
xlink:href="http://example/data/stations/

```

```

        125847521"
        xsi:type="ns:ReferenceType"/>
    </om:NamedValue>
</om:parameter>
<!-- Die beobachtete Messgröße, hierbei wird ein Verweis
auf ein entsprechendes Vokabular mit angegeben.-->
<om:observedProperty xlink:href="http://example/def/
486145151.xml"
xlink:title="WaterLevel"/>
<!--Der Standort, an dem die Messung vorgenommen wurde.-->
<om:featureOfInterest>
    <sams:SF_SpatialSamplingFeature
        xmlns:sams="http://www.opengis.net/
samplingSpatial/2.0"
        xmlns:sf="http://www.opengis.net/
sampling/2.0"
        gml:id="ssf_DE9504E6002BADERHA5
4D4F0BA568728A9">
        <gml:description>
            SF_SamplingPoint representation of the measurement
            location.
        </gml:description>
        <!--Identifizier der Messstelle.-->
        <gml:identifizier codeSpace="">http://example/stations/
984z98zj4
        </gml:identifizier>
        <!--Lesbarer Name der Messstelle.-->
        <gml:name codeSpace="http://www.opengis.net/def/nil/
OGC/0/unknown">
            Station XYZ
        </gml:name>
        <!--Feature-Typ mit dem die Messstelle repräsentiert
wird.-->
        <sf:type xlink:href="http://www.opengis.net/def/
samplingFeatureType/OGC-OM/2.0/
SF_SamplingPoint"/>
        <!--Geoobjekt, welches beobachtet wurde.-->
        <sf:sampledFeature xlink:href="River"/>
        <!--Geometrie der Messstelle-->
        <sams:shape>
            <gml:Point gml:id="Point_ssf_DE9504E6002BADERHA
54D4F0BA568728A9">
                <!--Koodinaten mit Angabe des Raumbezugs-
systems-->
                <gml:pos srsName="http://www.opengis.net/def/
crs/EPSSG/0/4326">
                    52.134 2.359760343 8.154871697
                </gml:pos>
            </gml:Point>
        </sams:shape>
    </sams:SF_SpatialSamplingFeature>
</om:featureOfInterest>
<om:result>
    <!--Das result-Element enthält eine Zeitreihe kodiert auf
Basis von WaterML 2.0-->
    <wml2:MeasurementTimeseries gml:id="timeseries.1">
        <wml2:metadata>
            <!--Zeitliche Ausdehnung der Zeitreihe. In diesem
Fall wird auf das oben dargestellte
phenomenonTime-Element verwiesen.-->
            <wml2:MeasurementTimeseriesMetadata>
                <wml2:temporalExtent
                    xlink:href="#phenomenonTime_1"/>
            </wml2:MeasurementTimeseriesMetadata>
        </wml2:metadata>
        <!--Informationen, die für alle Elemente der kodierten
Zeitreihe gleichermaßen gelten.-->
        <wml2:defaultPointMetadata>
            <wml2:DefaultTVPMeasurementMetadata>

```

```

        <!--Die Maßeinheit, in welcher die Messwerte
        angegeben sind.-->
        <wml2:uom code="cm"/>
        <!--Angabe zur Art des beobachteten Phänomens
        (z. B. augenblickliche Messung, Durchschnitt
        über Zeitperiode)-->
        <wml2:interpolationType
            xlink:href="http://www.opengis.net/def/
            timeseriesType/
            WaterML/2.0/
            continuous"
            xlink:title="Instantaneous"/>
        </wml2:DefaultTVPMeasurementMetadata>
    </wml2:defaultPointMetadata>
    <!--Ein Paar aus Zeitpunkt und Messwert wird als
    WaterML 2.0 point-Element kodiert.-->
    <wml2:point>
        <wml2:MeasurementTVP>
            <!--Der Zeitpunkt, auf den sich der Messwert
            bezieht.-->
            <wml2:time>2013-06-02T00:00:00.000Z</wml2:time>
            <!--Der Messwert.-->
            <wml2:value>83.428788</wml2:value>
        </wml2:MeasurementTVP>
    </wml2:point>
    <!-- Ab hier kann eine beliebige Anzahl weiterer
    point-Elemente in der gleichen Struktur wie zuvor
    folgen. -->
    <wml2:point>
        <wml2:MeasurementTVP>
            <wml2:time>2013-06-02T03:20:00.000Z</wml2:time>
            <wml2:value>83.49418</wml2:value>
        </wml2:MeasurementTVP>
    </wml2:point>
    <wml2:point>
        <wml2:MeasurementTVP>
            <wml2:time>2013-06-02T03:40:00.000Z</wml2:time>
            <wml2:value>83.198565</wml2:value>
        </wml2:MeasurementTVP>
    </wml2:point>
    <!--...-->
    </wml2:MeasurementTimeseries>
    </om:result>
    </omso:PointTimeSeriesObservation>
    </sos:observationData>
    </sos:GetObservationResponse>
    </soap:Body>
</soap:Envelope>

```

## Anhang C Referenzen

- [1] Andres, Volker, Henning Bredel, Ralf Busskamp, Simon Jirka, Ulrich Looser, Manuela Schlummer, Adrian Strauch und Michael Utech (2014). Interoperability Between GRDC's Data Holding and the GEOSS Infrastructure. HIC 2014 – 11th International Conference on Hydroinformatics, New York City, NY, USA, CUNY Academic Works.
- [2] Andres, Volker, Simon Jirka und Michael Utech (2014). OGC Best Practice: OGC Sensor Observation Service 2.0 Hydrology Profile [OGC 14-004r1]. Wayland, MA, USA, Open Geospatial Consortium Inc.
- [3] Botts, Mike und Alexandre Robin (2014). OGC Implementation Specification: Sensor Model Language (SensorML) 2.0.0 (12-000). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [4] Braeckel, Aaron, Lorenzo Bigagli und Johannes Echterhoff (2016). OGC Implementation Standard: Publish/Subscribe Interface Standard 1.0 – Core (13-131r1). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [5] Braeckel, Aaron und Lorenzo Bigagli (2016). OGC Implementation Standard: Publish/Subscribe Interface Standard 1.0 – SOAP Protocol Binding Extension (13-133r1). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [6] Brodaric, Boyan (2017). OGC Implementation Specification: WaterML 2.0: Part 4 – GroundWaterML 2 (GWML2) (16-032r2). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [7] Bröring, Arne, Christoph Stasch und Johannes Echterhoff (2012). OGC Implementation Specification: Sensor Observation Service (SOS) 2.0 (12-006). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [8] Busskamp, Ralf, Christoph Elvert, Dennis Prangenberg, Christian Förster, Simon Jirka, Wolfgang Kappler, Michael Utech und Thomas Schüttenberg (2016). Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten – Eine Heranführung an die Sensor-Web-Technologien. KW – Korrespondenz Wasserwirtschaft. Hennef, GFA. 9: 357-364.
- [9] Cox, Simon (2007). OGC Implementation Specification: Observations and Measurements (O&M) – Part 1 – Observation Schema 1.0.0 (07-022r1). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [10] Cox, Simon (2007). OGC Implementation Specification: Observations and Measurements (O&M) – Part 2 – Sampling Features 1.0.0 (07-002r3). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [11] Cox, Simon (2011). OGC Implementation Specification: Observations and Measurements (O&M) – XML Implementation 2.0 (10-025r1). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [12] Cox, Simon and Bruce A Simons (2014). OGC Best Practice: WaterML-WQ – An O&M and WaterML 2.0 Profile for Water Quality Data 1.0 [OGC 14-003]. Wayland, MA, USA, Open Geospatial Consortium Inc.
- [13] Echterhoff, Johannes (2011). OGC Implementation Specification: SWE Service Model 2.0.0 (09-001). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [14] Gesetz über den Zugang zu digitalen Geodaten (Geodatenzugangsgesetz, GeoZG)
- [15] Heier, Christian und Karl-Heinz Spies (2009). Einsatz der OGC Sensor Web Standards in der Wasserwirtschaft. KW – Korrespondenz Wasserwirtschaft. Hennef, GFA. 3: 369-373.
- [16] INSPIRE MIG sub-group MIWP-7a (2016). Guidelines for the use of Observations & Measurements and Sensor Web Enablement-related standards in INSPIRE – Version 3.0. Ispra, Italy, INSPIRE Maintenance and Implementation Group (MIG).
- [17] INSPIRE MIG sub-group MIWP-7a (2016). Technical Guidance for implementing download services using the OGC Sensor Observation Service and ISO 19143 Filter Encoding – Version 1.0. Ispra, Italy, INSPIRE Maintenance and Implementation Group (MIG).
- [18] ISO TC 211 (2011). ISO 19156:2011 – Geographic information -- Observations and measurements – International Standard. Genf, Schweiz, International Organization for Standardization.
- [19] Na, Arthur und Mark Priest (2007). OGC Implementation Specification: Sensor Observation Service (SOS) 1.0.0 (06-009r6). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [20] Robin, Alexandre (2011). OGC Implementation Specification: SWE Common Data Model 2.0.0 (08-094r1). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [21] Simonis, Ingo und Johannes Echterhoff (2011). OGC Implementation Specification: Sensor Planning Service (SPS) 2.0.0 (09-000). Wayland, MA, USA, Open Geospatial Consortium Inc.

- [22] Taylor, Peter (2014). OGC Implementation Specification: WaterML 2.0.1: Part 1 – Timeseries (10-126r4). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [23] Taylor, Peter (2016). OGC Implementation Specification: WaterML 2.0: Part 2 – Ratings, Gaugings and Sections (15-018r2). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [24] Tomkins, James und Dominic Lowe (2016). OGC Implementation Specification: Timeseries Profile of Observations and Measurements 1.0 (15-043r3). Wayland, MA, USA, Open Geospatial Consortium Inc.
- [25] Tomkins, James und Dominic Lowe (2016). OGC Implementation Specification: TimeseriesML 1.0 – XML Encoding of the Timeseries Profile of Observations and Measurements 1.0.0 (15-042r3). Wayland, MA, USA, Open Geospatial Consortium Inc.

## Anhang D Autorenverzeichnis

Der Autorenkreis setzt sich aus den Mitgliedern der DWA-Arbeitsgruppe BIZ 12.2 GIS & GDI zusammen:

Dr. Ralf Busskamp  
Referatsleiter „Geoinformation & Fernerkundung, GRDC“  
Bundesanstalt für Gewässerkunde  
Am Mainzer Tor 1  
56068 Koblenz

Christoph Elvert  
Stadtentwässerungsbetriebe Köln, AöR  
Geodaten, Vermessung und Liegenschaften  
Ostmerheimer Straße 555  
51109 Köln

Christian Förster  
Sachgebiet GIS und GDI  
Wupperverband – Körperschaft des öffentlichen Rechts  
Untere Lichtenplatzer Straße 100  
42289 Wuppertal

Dr. Simon Jirka  
52°North Initiative for Geospatial Open Source Software GmbH  
Martin-Luther-King-Weg 24  
48155 Münster

Wolfgang Kappler  
ahu AG  
Wasser Boden Geomatik  
Kirberichshofer Weg 6  
52066 Aachen

Dennis Prangenberg  
Esri Deutschland GmbH  
Niederlassung Köln  
Konrad-Adenauer-Ufer 41-45  
50668 Köln

Thomas Schüttenberg  
Technische Werke Burscheid AöR  
Geodatenmanagement  
Pastor-Löh-Straße 12  
51399 Burscheid

## Standardisierte Bereitstellung und Nutzung von wasserwirtschaftlichen Messdaten

Christoph Stasch  
52°North Initiative for Geospatial Open Source Software GmbH  
Martin-Luther-King-Weg 24  
48155 Münster

Michael Utech  
KISTERS AG  
Pascalstraße 8+10  
52076 Aachen





**Deutsche Vereinigung für Wasserwirtschaft, Abwasser und Abfall e. V. (DWA)**

Theodor-Heuss-Allee 17 · 53773 Hennef

Telefon: +49 2242 872-333 · Fax: +49 2242 872-135

info@dwa.de · www.dwa.de

